



ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO

FACULTAD DE INFORMÁTICA Y ELECTRÓNICA

CARRERA DE INGENIERÍA EN SISTEMAS

“DESARROLLO DE UN SISTEMA WEB PARA LA AUTOMATIZACIÓN DEL PROCESO DE CALENDARIZACIÓN DE RIEGO PARA EL CENTRO EXPERIMENTAL DEL RIEGO DE LA ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO”

Trabajo de titulación

Tipo: Proyecto Técnico

Presentado para optar el grado académico de:

INGENIERO EN SISTEMAS INFORMÁTICOS

AUTOR: ISAAC DAVID TORRES PAREDES

DIRECTORA: Ing. MAYRA ALEJANDRA OÑATE ANDINO

Riobamba – Ecuador

2020

©2020, Isaac David Torres Paredes

Se autoriza la reproducción total o parcial, con fines académicos, por cualquier medio o procedimiento, incluyendo la cita bibliográfica del documento, siempre y cuando se reconozca el Derecho de autor.

Yo, Isaac David Torres Paredes, declaro que el presente trabajo de titulación es de mi autoría y los resultados del mismo son auténticos. Los textos en el documento que provienen de otras fuentes están debidamente citados y referenciados.

Como autor asumo la responsabilidad legal y académica de los contenidos de este trabajo de titulación; el patrimonio intelectual pertenece a la Escuela Superior Politécnica de Chimborazo

Riobamba, 05 de Agosto de 2020

Isaac David Torres Paredes.

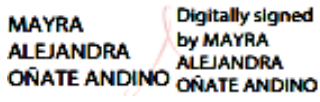
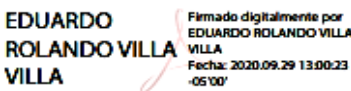
080249715-6

ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO

FACULTAD DE INFORMÁTICA Y ELECTRÓNICA

CARRERA DE INGENIERÍA EN SISTEMAS

El Tribunal del Trabajo de Titulación certifica que: El trabajo de titulación; tipo: Proyecto Técnico, “**DESARROLLO DE UN SISTEMA WEB PARA LA AUTOMATIZACIÓN DEL PROCESO DE CALENDARIZACIÓN DE RIEGO PARA EL CENTRO EXPERIMENTAL DEL RIEGO DE LA ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO**”, realizado por el señor: **ISAAC DAVID TORRES PAREDES**, ha sido minuciosamente revisado por los Miembros del Trabajo de Titulación, el mismo que cumple con los requisitos científicos, técnicos, legales, en tal virtud el Tribunal Autoriza su presentación.

	FIRMA	FECHA
Ing. Jorge Ariel Menéndez Verdecia PRESIDENTE DEL TRIBUNAL	 Firmado digitalmente por: JORGE ARIEL MENENDEZ VERDECIA	2020-09-28
Ing. Mayra Alejandra Oñate Andino DIRECTOR DE TRABAJO DE TITULACIÓN	 Digitally signed by MAYRA ALEJANDRA OÑATE ANDINO	2020-09-28
Ing. Eduardo Rolando Villa Villa MIEMBRO DEL TRIBUNAL	 Firmado digitalmente por EDUARDO ROLANDO VILLA VILLA Fecha: 2020.09.29 13:00:23 -05'00'	2020-09-28

DEDICATORIA

El esfuerzo y dedicación invertida en realizar este trabajo de titulación esta dedico primeramente para exaltar la gloria y honra de mi Dios mi creador y mi padre él es: *“Jehová de los ejércitos”*, también a toda mi familia, empezando por mi mami, Teresa Paredes que ha sido más que una madre una amiga un apoyo incondicional en todo momento. Mi papi Alciviades Torres que ha sido el pilar tanto económico como consejero, mi segunda madre y hermana Sonia Torres que siempre ha estado cuando más lo he necesitado conjuntamente con su esposo mi cuñado Fernando Tubón y mis queridos sobrinos Génesis y David, a mi hermana menor mi querida Barbarita Torres que ha sido un motivo bien fuerte para querer ser un profesional para poder ser un ejemplo de bien para ella, a mi querida mujer, compañera, confidente, amiga Katherine Chalá (Lu) que me ha dado ánimos a continuar en esta larga trayectoria de formación profesional, y de una manera muy especial a mi pequeño hijo Aarón Caleb que con sus travesuras e inquietudes me ha dado motivación y ganas de ser mejor persona, y de esta manera poder ser un mejor padre para él.

Isaac

AGRADECIMIENTOS

Realmente no me alcanzaría las palabras para agradecerle a Dios por el amor y misericordia que ha tenido para conmigo, porque a pesar de mi ingratitud e indiferencia con él, su voluntad me ha permitido llegar a terminar esta etapa de mi vida. ***De todo corazón muchas Gracias Señor Jesucristo.*** También agradezco a mis queridos e incondicionales padres que a pesar de mis errores siempre estuvieron ahí para ayudarme en todo sentido, sin juzgarme, a mis hermanas por sus consejos y ánimos. Le agradezco también a mi querida mujer por su comprensión y apoyo sincero que me ha brindado. A todos y cada uno de ustedes un enorme “*Dios le pague*”.

A mí estimada Directora del trabajo de titulación Ing. Alejandra Oñate, muchas gracias por sus sabias palabras, consejos y orientaciones.

Isaac

TABLA DE CONTENIDO

ÍNDICE DE TABLAS.....	xi
ÍNDICE DE FIGURAS.....	xii
ÍNDICE DE GRÁFICOS.....	xiii
ÍNDICE DE ANEXOS.....	xiv
ÍNDICE DE ABREVIATURAS.....	xv
RESUMEN.....	xvi
SUMMARY	xvii
INTRODUCCIÓN	1

CAPÍTULO I

1. MARCO TEÓRICO.....	6
1.1. Centro experimental del riego de la ESPOCH.....	6
1.2. Calendario de riego	6
1.2.1. <i>Beneficios de un calendario de riego</i>	6
1.3. Métodos para el proceso de la elaboración del calendario de riego.....	7
1.3.1. <i>Método analítico</i>	7
1.3.2. <i>Método gráfico</i>	13
1.4. Sistemas informáticos utilizados para elaborar un calendario de riego.....	16
1.4.1. <i>Aquadaia</i>	18
1.4.2. <i>Croptwat</i>	18
1.5. Aplicaciones Web.....	18
1.5.1. <i>Características de las aplicaciones web</i>	18
1.5.2. <i>Ventajas y desventajas de las aplicaciones web</i>	19
1.6. Arquitectura de la aplicación web.....	19
1.6.1. <i>Elementos</i>	21
1.6.2. <i>Características</i>	21
1.6.3. <i>Ventajas y desventajas</i>	21
1.7. Patrón de diseño Modelo – Vista – Controlador MVC	21
1.7.1. <i>Características del MVC</i>	22
1.7.2. <i>Elementos del MVC</i>	23
1.8. Herramientas de desarrollo	23
1.8.1. <i>Lenguaje de programación</i>	23

1.8.2.	<i>Lenguaje de programación JAVA</i>	24
1.8.3.	<i>Herramientas de desarrollo para la capa cliente</i>	25
1.8.4.	<i>Entorno de desarrollo integrado</i>	25
1.8.5.	<i>Entorno de desarrollo integrado Netbeans</i>	26
1.8.6.	<i>Características de Netbeans</i>	26
1.8.7.	<i>Frameworks para desarrollo de interfaces</i>	26
1.8.8.	<i>Funcionamiento de AJAX</i>	28
1.8.9.	<i>Tecnologías implicadas en el mecanismo de AJAX</i>	28
1.8.10.	<i>Herramientas para el desarrollo de la capa Servidor</i>	29
1.8.11.	<i>Características de JSP</i>	30
1.9.	Bases de datos	30
1.10.	PostgreSQL	31
1.10.1.	<i>Características del sistema gestor de base de datos SGBD PostgreSQL</i>	31
1.11.	Metodología para gestión de proyecto	32
1.11.1.	<i>Definición de SCRUM</i>	32
1.11.2.	<i>Perspectivas de SCRUM</i>	33
1.12.	Evaluación de calidad en aplicaciones web	35

CAPÍTULO II

2.	MARCO METODOLÓGICO	38
2.1.	Diseño de la investigación.	38
2.1.1.	<i>Tipo de investigación</i>	38
2.1.2.	<i>Método de investigación.</i>	38
2.1.3.	<i>Técnicas de Investigación</i>	39
2.2.	Determinación del proceso de la elaboración del calendario de riego.	39
2.2.1.	<i>Gestión de los datos: Relación – Agua – Suelo – Planta – atmósfera (RASPA)</i>	39
2.2.2.	<i>Gestión del Calendario de riego</i>	39
2.3.	Desarrollo del sistema informático para la automatización del proceso de elaboración del calendario de riego	44
2.3.1.	<i>Fase de inicio</i>	45
2.3.2.	<i>Personas Involucradas</i>	45
2.3.3.	<i>Tipos de usuarios y sus roles</i>	45
2.4.	Requerimientos solicitados del usuario para el sistema	46
2.5.	Product Backlog	46
2.6.	Fase de planificación	48

2.6.1.	<i>Sprint Backlog</i>	49
2.7.	Fase de desarrollo	51
2.8.	Arquitectura del sistema	51
2.8.1.	<i>Capa de cliente</i>	51
2.8.2.	<i>Capa de Servidor</i>	52
2.9.	Estándar de codificación	53
2.10.	Estándar de interfaces	53
2.11.	Diseño de la base de datos	54
2.11.1.	<i>Reuniones y entregables</i>	55
2.11.2.	<i>Documentación</i>	55
2.11.3.	<i>Historias de usuario</i>	56
2.11.4.	<i>Manual de usuario</i>	57
2.11.5.	<i>Diagrama de despliegue</i>	57
2.12.	Fase de cierre	58
2.12.1.	<i>Gestión del proyecto</i>	58
2.12.2.	<i>Burndown Chart</i>	58
2.13.	Método utilizado para evaluar la eficiencia del desempeño de la aplicación web	59
2.13.1.	<i>Análisis comparativo del proceso de manera manual vs automático</i>	59
2.13.2.	<i>Población</i>	59
2.13.3.	<i>Muestra</i>	59
2.13.4.	<i>Comportamiento del tiempo</i>	60
2.14.	Utilización de recursos	60
2.14.1.	<i>Recolección de datos</i>	60

CAPÍTULO III

3.	MARCO DE RESULTADOS, DISCUSIÓN Y ANÁLISIS DE RESULTADOS.	62
3.1.	Estadística Descriptiva	62
3.2.	Eficiencia	62
3.2.1.	<i>Tiempo de respuesta</i>	62
3.3.	Estadística Inferencial	66
3.3.1.	<i>Grupo 1</i>	66
3.3.2.	<i>Grupo 2</i>	68
3.3.3.	<i>Grupo 3</i>	70
3.4.	Recursos Utilizados	73
3.4.1.	<i>Estadística descriptiva</i>	74

CONCLUSIONES.....	75
RECOMENDACIONES.....	76
GLOSARIO	
BIBLIOGRAFÍA	
ANEXOS	

ÍNDICE DE TABLAS

Tabla 1-1:	Formato del calendario de riego	13
Tabla 3-1:	Ventajas y desventajas de las aplicaciones web	19
Tabla 4-1:	Ventajas y desventajas de las aplicaciones web.	19
Tabla 5-1:	Ventajas y desventajas de arquitectura Cliente/Servidor.....	21
Tabla 6-1:	Cuadro comparativo de los lenguajes de programación.	24
Tabla 7-1:	Cuadro comparativo entre entornos de desarrollo integrado.	25
Tabla 8-1:	Frameworks de desarrollo para interfaces.	27
Tabla 9-1:	Cuadro comparativo entre tecnologías de desarrollo web.	29
Tabla 10-1:	Sistemas gestores de bases de datos.	31
Tabla 11-1:	Cuadro comparativo metodologías ágiles.....	32
Tabla 12-1:	Roles, artefactos y actividades dentro de la metodología SCRUM.....	34
Tabla 13-1:	Normas y estándares para medir la calidad en un software.	36
Tabla 14-1:	Elementos de la calidad del producto software.	37
Tabla 1-2:	Descripción de pasos para la elaboración del calendario de riego manual.....	42
Tabla 2-2:	Descripción de los pasos de elaboración del calendario de riego automático.	44
Tabla 3-2:	Roles y personas involucradas en el proyecto.	45
Tabla 4-2:	Descripción de las actividades que puede realizar un usuario.....	45
Tabla 5-2:	Descripción de técnica de estimación T-Shirt.	47
Tabla 6-2:	Lista de actividades priorizadas. (Product Backlog)	47
Tabla 7-2:	Sprint Backlog.	49
Tabla 8-2:	Características de computadora evaluada.	60
Tabla 1-3:	Descripción de las muestras tomadas y valores calculados del nivel básico.	63
Tabla 2-3:	Descripción de las muestras tomadas y valores calculados del nivel intermedio	64
Tabla 3-3:	Descripción de las muestras tomadas y valores calculados del nivel avanzado. .	65
Tabla 4-3:	Resultado Saphrio Wilk nivel básico.....	66
Tabla 5-3:	Prueba T pareada nivel básico	67
Tabla 6-3:	Resultado Saphrio Wilk nivel intermedio.	68
Tabla 7-3:	Prueba T pareada nivel intermedio.....	69
Tabla 8-3:	Resultado Saphrio Wilk nivel avanzado.....	70
Tabla 9-3:	Prueba T pareada nivel básico	71
Tabla 10-3:	Descripción hardware y software de computadora evaluada.	73
Tabla 11-3:	Descripción hardware y software de computadora evaluada.	74

ÍNDICE DE FIGURAS

Figura 1-1:	Cuadro de referencia para determinación del Kp.....	9
Figura 2-1:	Datos para cálculo de calendario de riego de manera gráfica.....	14
Figura 3-1:	Representación del calendario de riego de manera gráfica.....	16
Figura 4-1:	Arquitectura Cliente/Servidor.....	20
Figura 5-1:	Patrón de diseño MVC.....	22
Figura 6-1:	Funcionamiento del lenguaje JAVA.....	25
Figura 7-1:	Funcionamiento una JSP.....	30
Figura 8-1:	Funcionamiento de framework Hibernate JPA.....	33
Figura 9-1:	Resumen de actividades y artefactos SCRUM.....	35
Figura 1-2:	Estándar de colores de interfaz de usuario formulario ingreso datos.....	53
Figura 2-2:	Estándar de colores de interfaz de usuario reporte.....	54

ÍNDICE DE GRÁFICOS

Gráfico 1-2:	Diagrama de proceso de elaboración del calendario de riego.	41
Gráfico 2-2:	Diagrama de proceso de elaboración del calendario de riego.	43
Gráfico 3-2:	Arquitectura del sistema	51
Gráfico 4-2:	Modelo vista controlador con sus componentes.....	52
Gráfico 5-2:	Base de datos del sistema.	55
Gráfico 6-2:	Diagrama de despliegue.	57
Gráfico 7-2:	Puntos estimados vs reales de los Sprints.....	58
Gráfico 1-3:	Tiempo empleado antes y después nivel básico (minutos).....	63
Gráfico 2-3:	Tiempo empleado antes y después intermedio (minutos)	64
Gráfico 3-3:	Tiempo empleado antes y después nivel avanzado (minutos).....	65
Gráfico 4-3:	Tiempos generales invertidos en la elaboración del calendario de riego.	72
Gráfico 5-3:	Reducción de tiempo representado en porcentaje.	72

ÍNDICE DE ANEXOS

ANEXO A: Manual técnico.

ANEXO B: Diagramas de casos de usos con sus tablas descriptivas.

ANEXO C: Estándar de Programación.

ANEXO D: Manual de usuario.

ANEXO E: Tiempos de respuesta y utilización de recursos.

ÍNDICE DE ABREVIATURAS

CER:	Centro Experimental del Riego.
ISO:	Organización Internacional de Normalización.
MVC:	Modelo vista controlador.
JSP:	Java Server Page.
HU:	Historia de Usuario.
HT:	Historia Técnica.
TI:	Tarea de Ingeniería.
PA:	Prueba de aceptación.
SCRUM:	Significa melé, un tipo de jugada de Rugby.
PDF:	Formato de documento portátil.

RESUMEN

El presente trabajo de investigación tuvo como objetivo principal desarrollar un sistema informático web para la automatización del proceso de calendarización de riego para el Centro Experimental del Riego (CER) de la Escuela Superior Politécnica del Chimborazo. Para obtener la información necesaria para comprender el proceso, se utilizó las siguientes técnicas de investigación: observación y revisión de documentación. Se implementó este software con la ayuda de la metodología ágil SCRUM gestionando las actividades y tareas por medio de la herramienta Target Process, metodología que permitió implementar la mayoría de requisitos iniciales del software. Para el desarrollo del sistema informático se utilizó herramientas de software de libre acceso como por ejemplo: Entorno de desarrollo integrado (IDE) Netbeans 8.2, Bootstrap 4 como gestor de base de datos PostgreSQL 9.5 y Payara Server. El software implementado se lo evaluó por medio de la métrica tiempo de respuesta y utilización de recursos, correspondiente a la eficiencia según la norma ISO 25010. Al evaluar la eficiencia de la aplicación web y luego de cuantificar el tiempo que se demoraban los usuarios en elaborar un calendario de riego de manera manual y después de manera automática, se obtuvo los siguientes resultados: en el nivel básico se redujo el tiempo en un 93.2%, en el nivel intermedio un 89.71% y en el nivel avanzado un 90%, teniendo un promedio de reducción del tiempo de un 90.97%. El software realizado en esta investigación ayuda a reducir los tiempos empleados en desarrollar un calendario de riego. Se recomienda agregar más cultivos para ampliar el alcance de la aplicación web además de ejecutarla en el navegador Internet Explorer dado que es el que consume 245.85MB de memoria RAM y el 19.85% de capacidad del procesador, cantidades menores en relación a otros navegadores evaluados.

Palabras clave: <INGENIERÍA DE SOFTWARE>, <AGRICULTURA>, <DESARROLLO DE APLICACIONES WEB>, <METODOLOGÍA DE DESARROLLO ÁGIL (SCRUM)>, <CALENDARIO DE RIEGO>, <DATOS CLIMATOLÓGICOS>, <EFICIENCIA DE DESEMPEÑO>, <OPTIMIZACIÓN DEL TIEMPO>.



0293-DBRAI-UPT-

2020

SUMMARY

The main objective of this research work was to develop a web computer system for the automation of the irrigation scheduling process for the Experimental Irrigation Center (CER) of the Escuela Superior Politécnica del Chimborazo. To obtain the necessary information to understand the process, the following investigation techniques were used: observation and documentation review. This software was implemented with the help of the agile SCRUM methodology, managing activities and tasks through the Target Process tool, a methodology that allowed the implementation of most of the initial software requirements. For the development of the computer system, free access software tools were used, such as: Integrated development environment (IDE) Netbeans 8.2, Boostrap 4 as PostgreSQL 9.5 database manager and Payara Server. The implemented software was evaluated by means of the metric response time and resource utilization, corresponding to the efficiency according to the ISO 25010 standard. When evaluating the efficiency of the web application and after quantifying the time it took users to elaborate a manual irrigation schedule and then automatically, the following results were obtained: at the basic level the time was reduced by 93.2%, at the intermediate level by 89.71% and at the advanced level by 90%, having a average time reduction of 90.97%. The software made in this research helps reduce the time spent developing an irrigation schedule. It is recommended to add more crops to expand the scope of the web application in addition to running it in the Internet Explorer browser since it consumes 245.85MB of RAM memory and 19.85% of processor capacity, smaller amounts in relation to other browsers evaluated.

Keywords: <SOFTWARE ENGINEERING>, <AGRICULTURE>, <WEB APPLICATION DEVELOPMENT>, <AGILE DEVELOPMENT METHODOLOGY (SCRUM)>, <IRRIGATION CALENDAR>, <CLIMATOLOGICAL DATA>, <PERFORMANCE EFFICIENCY>, <OPTIMIZATION OF TIME>.

INTRODUCCIÓN

La calendarización busca proporcionar de una manera fácil el comprender y acceder a un plan de trabajo establecido para una tarea, de la misma manera la calendarización en el riego busca crear un cronograma para saber cuándo y cuánto hay que regar de agua a un cultivo que está en proceso de desarrollo. Si bien se ha avanzado mucho en la elaboración de este calendario de riego los involucrados en el tema como ingenieros agrónomos, agricultores, estudiantes de carreras afines, etc. se enfrentan a nuevos inconvenientes: a la falta de una herramienta tecnológica (software) que ayude a mejorar el tiempo invertido en desarrollar esta tarea, cabe mencionar que sí existen software's e incluso aplicaciones móviles que realizan este proceso, pero no tienen datos propios como por ejemplo; condiciones climáticas o tipos de cultivos de la zona (maíz, cebolla, papa, arveja, frejol, zanahoria, brócoli). Se puede lograr un progreso significativo si un programa informático automatiza la tarea de crear un calendario de riego y lo más importante aplicar datos que son tomados del territorio mismo de la siembra, para una mayor precisión en cantidades y fechas de riego.

La importancia de crear un programa viene del interés de hacerlo fácil entendible y manipulable por las personas que lo van a utilizar, a la vez para que este sirva de guía para la creación de un calendario de riego el cual ayuda mucho en el proceso de siembra de un cultivo.

Al desarrollar el programa para la elaboración de un calendario de riego se analizaron dos herramientas tecnológicas relacionadas a esta área como lo son: Cropwat en su versión 8.0 de la FAO (Organización de las Naciones Unidas para la Agricultura y la Alimentación) y Aquadaia que es una aplicación móvil, ambos programas sirven para determinar la necesidad hídrica de la planta. Además, durante la creación del presente programa informático se ha contado con la asesoría del Ing. Juan León director del CER (Centro Experimental del Riego de la Escuela Superior Politécnica de Chimborazo) quien ha contribuido con su conocimiento y experiencia en el desarrollo de calendarios de riego y necesidades hídricas de los cultivos de la zona.

Para implementar el programa para la automatización del proceso de calendarización de riego, se utilizó el lenguaje de programación JAVA y herramientas de software libre como lo son: el entorno de desarrollo integrado IDE netbeans en su versión 8.2, el framework para interfaces de usuario denominado BOOSTRAP, el framework para persistencia de bases de datos JPA Hibernate, el sistema gestor de base de datos PostgreSQL Server en su versión 9.5, un IDE de conexión con el gestor de base de datos PgAdmin en su versión 4, como servidor informático se utilizó a Payara Server en su versión 4.1, además para la gestión de tareas y actividades involucradas en el desarrollo de este sistema se hizo uso de la herramienta Target Process la cual

se ajusta muy bien a la metodología ágil SCRUM, misma que administro las etapas de desarrollo de este sistema informático de tipo web.

Al desarrollar el programa se completaron tres módulos, el primero relacionado a los datos y condiciones climatológicas, el segundo está comprendido por los datos de los cultivos y tercero el módulo de reportes el cual se encarga de la generación de reportes en manera gráfica y física, cabe aclarar que el sistema está orientado elaborar calendarios de riego de los cultivos de la zona de la provincia de Chimborazo y más que todo a los cultivos que ya han sido objeto de estudio e investigaciones en el CER, limitando por el momento a otros tipo de cultivos que no estén en los anteriormente mencionados.

El sistema informático esta evaluado para comprobar la calidad de software en el ámbito de eficiencia de desempeño según la norma ISO/IEC 25010 con la subcaracterística **comportamiento temporal** la cual evalúa los tiempos de respuesta y procesamiento de datos de un sistema cuando lleva a cabo sus funciones en condiciones determinadas en relación a un banco de pruebas previamente establecido y la **utilización recursos** la cual evalúa a nivel de hardware la cantidad de memoria RAM y porcentaje de procesador utilizado al ejecutarse la aplicación.

ANTECEDENTES

El centro experimental del riego, inicia sus actividades en el año 2010; dentro de estas está la investigación relacionada al riego y la capacitación a agricultores de la zona, como parte de estas actividades está la elaboración de los calendarios de riego para los diferentes tipos de cultivos (papa, cebolla, zanahoria, arveja, maíz). Durante una visita a este centro de investigación se pudo observar que la elaboración del calendario de riego la realizan los técnicos de manera manual, es decir toman los valores de las diferentes variables involucradas en este proceso y van realizando cada uno de los cálculos matemáticos paso a paso, hasta llegar a obtener el calendario de riego según los datos propios de cada cultivo. Este proceso demanda de un tiempo promedio de 4 horas para concretarlo completamente. Una de las causas que dan origen a este problema, es que el proceso de elaboración del calendario de riego se lo realiza una y otra vez para cada siembra repitiendo pasos (gráficos de comportamiento del cultivo, tablas de tiempo-cantidad de riego, cuadros comparativos, etc.). Las secuelas que generan estas causas es que se debe de volver a invertir una cantidad de tiempo que ya se la utilizó anteriormente en la elaboración de un calendario de riego, ocupando tiempos que se pueden invertir en otras áreas del proceso del cultivo. Estos efectos hacen que este proceso se vuelva repetitivo y laborioso para la persona que lo realiza.

Al conversar con los técnicos del centro investigativo mencionaron que si existen algunos programas informáticos que ayudan en la elaboración de este proceso, pero el inconveniente está

en que dichos programas tienen modelos matemáticos basados en parámetros de otros países como por ejemplo CROPWAT, dicho software se basa en datos de cultivos y climas de California - Estados Unidos. Otro programa es AQUADAIA el cual permite determinar la cantidad de agua necesaria en un cultivo para su crecimiento óptimo, el inconveniente que existe con ese programa es que utiliza el método de coeficiente dual para el cálculo de la evapotranspiración del cultivo el mismo que trabaja con datos meteorológicos aproximados del lugar no son los valores exactos de donde se va a realizar la siembra. Todo esto hace que exista la necesidad de crear un sistema informático basado en modelos e investigaciones ya realizadas en el CER, con los valores meteorológicos tomados en la zona. Hay que mencionar también que los cultivos con los que trabaja el CER en los cuales ya tiene información son (papa, maíz, cebolla, arveja, frejol, zanahoria, lechuga y brócoli), conocida esta información el sistema debe ser enfocado a los cultivos anteriormente mencionados.

FORMULACIÓN DEL PROBLEMA

¿De qué manera el desarrollo de un sistema web para la automatización del proceso de calendarización de riego, para el centro experimental del riego de la ESPOCH, contribuirá a la reducción del tiempo invertido en la elaboración del calendario de riego por cada siembra de manera manual?

JUSTIFICACIÓN DEL TRABAJO DE TITULACIÓN

JUSTIFICACIÓN TEÓRICA

Teóricamente, el presente proyecto técnico justifica su emprendimiento, porque se pretende aplicar la automatización al proceso de cobranza para de esta manera ayudar a optimizar los tiempos empleados en realizar esta tarea. Para lograr automatizar este proceso se va a combinar las tecnologías informáticas como JAVA según el sitio campusMVP sostiene que JAVA es un lenguaje de código robusto, Gracias a su orientación a objetos es más fácil crear aplicaciones modulares y reutilizar partes de las mismas, además de que permite implementar mejores patrones de diseño probados que contribuyen a la robustez. (campusMVP, 2019)

Sistemas gestores de bases de datos (PostgreSQL), entornos de desarrollo (Netbeans), todas estas herramientas se las ha seleccionado porque el sitio GENBETA enuncia lo siguiente acerca de estas tecnologías:

Netbeans suele dar soporte a casi todas las novedades en el lenguaje Java. Cualquier previa del lenguaje es rápidamente soportada por Netbeans.

Es un asistente para la creación y configuración de distintos proyectos, incluida la elección de algunos frameworks. (Suárez, 2013)

Para gestionar las actividades y tareas se propone utilizar la metodología ágil de gestión de proyectos SCRUM porque según Ramiro Moreno Barcia trabajador de la empresa de comunicaciones y marketing “METODO” dice que Scrum es actualmente la metodología ágil más popular. Se basa principalmente en la premisa de ejecutar un proyecto en entregas parciales y regulares del producto. El desarrollo del producto se realiza de forma incremental y evolutiva, lo que resulta ideal en entornos dinámicos y cambiantes. (Barcia, 2016).

Se pretende utilizar el framework de desarrollo para interfaces Bootstrap en su versión 4.1.3 lanzada al mercado en Julio 25, 2018. (RubyGems, 2018). Según el sitio informático DevCode, Bootstrap facilita la maquetación de sitios web, además de ser compatible con preprocesadores como Less y Saas, nos ofrece las herramientas para que nuestro sitio web se vea bien en toda clase de dispositivos, ahorrándonos así el trabajo de tener que rediseñar un sitio web. (DevCode, 2013). Es por esta razón que se pretende usar esta tecnología para desarrollar interfaces intuitivas e amigables con el usuario final del sistema.

JUSTIFICACIÓN APLICATIVA

En la parte práctica, el presente proyecto técnico justifica su realización, ya que se pretende desarrollar un sistema informático web para que lo utilicen usuarios en general que reciben capacitaciones en el centro experimental del riego perteneciente a la Escuela Superior Politécnica, éste sistema va a ayudar a mejorar los tiempos empleados en la elaboración de calendarios de riego. A continuación se detallan los módulos que se pretende realizar:

Módulo calendario en el cual se puede gestionar la información relacionada a la siembra (fecha, área, cultivo, tipo de suelo) para la cual se está elaborando el calendario de riego.

Módulo de gestión de datos climatológicos en el cual se va a poder administrar las variables involucradas para la elaboración del calendario de riego conjuntamente con sus valores sus valores.

Módulo de gestión de cultivo, el cual va a contener la información característica de cada cultivo (tiempo de vida, kc, lámina de riego, tamaño aprox, condiciones bioambientales).

Este trabajo está alineado a las líneas transversales de investigación de la ESPOCH las cuales son: las tecnologías de la información y la comunicación conjuntamente con el programa desarrollo de software.

OBJETIVOS

OBJETIVO GENERAL

Desarrollar un sistema informático web para la automatización del proceso de calendarización de riego para el Centro Experimental del Riego (CER) de la ESPOCH.

OBJETIVOS ESPECÍFICOS

- Analizar el proceso de elaboración de un calendario de riego, para comprender el algoritmo del mismo.
- Determinar los métodos que existen para el proceso de la elaboración del calendario de riego.
- Evaluar la eficiencia de desempeño del proceso automatizado para la elaboración del calendario de riego bajo los lineamientos de la norma ISO 25010.

CAPÍTULO I

1. MARCO TEÓRICO

1.1. Centro experimental del riego de la ESPOCH

La Secretaria de Riego y Drenaje determinó la necesidad de realizar investigaciones relacionadas al riego, en respuesta a esa necesidad el Centro Experimental del Riego (CER) de la Escuela Superior Politécnica de Chimborazo empezó sus actividades en el año del 2010, tras suscribirse el convenio ESPOCH- MAGAP. A partir de esto, el CER realiza investigaciones y brinda capacitaciones a nivel nacional, para personal docente, técnicos y agricultores, realizando actividades como diseño de sistemas de riego así como la operación y mantenimiento de los mismos, entre otras actividades. (ESPOCH-Comunicaciones, 2010)

Dentro de los diseños de sistemas de riego que realiza el CER está la elaboración de los calendarios de riego, tarea que se la realiza de manera manual, requiriendo un tiempo promedio de dos horas para su elaboración total, según lo establecido por el Ing. Juan León Director del CER.(ESPOCH-Comunicaciones, 2010)

1.2. Calendario de riego

Un calendario de riego es una tabla donde se indican los meses y el año conjuntamente con las cantidades de agua que se debe suministrar en una siembra específica dependiendo del tipo de cultivo (León, 2012).

Por otra parte, de acuerdo a Cisneros en su libro RIEGOS Y DRENAJES: “Las estrategias de riego son, unos criterios generales que se concretan elaborando un calendario medio de riegos en el que se precisa el momento del riego y la cantidad de agua”. (Cisneros, 2003)

1.2.1. Beneficios de un calendario de riego

El impacto que tiene la creación de un calendario de riego al inicio de una siembra es muy alto, puesto que permite realizar una planificación y programación de riegos necesarios para ayudar a desarrollarse de una manera adecuada. (Cisneros, 2003)

Al crear un calendario para el cultivo se obtiene los siguientes beneficios:

- Número de riegos por aplicar al cultivo.
- Intervalos entre riegos.

- Láminas de requerimiento de riego.
- Láminas netas de riego.

1.3. Métodos para el proceso de la elaboración del calendario de riego.

Dada la importancia y los beneficios que tiene elaborar un calendario de riego existen dos métodos que permiten elaborar el mismo, estos métodos ayudan a establecer los tiempos y cantidades de riego que necesita el cultivo de una siembra, estos son: el método gráfico y el analítico, los cuales se detalla a continuación:

1.3.1. Método analítico

El método analítico está basado en fórmulas y cálculos matemáticos, este consiste en ir calculando los valores de las variables de manera consecutiva, una en consecuencia de otra, en total son 10 pasos, se empieza por la evapotranspiración (ET_o), con este valor se procede a calcular el coeficiente del tanque de cultivo (K_p), con estos valores se pasa a calcular la pérdida de humedad en el suelo (ET_c), luego se calcula el coeficiente de cultivo (K_c), con estos valores se procede a calcular el agua útil (A_u), luego se calcula la lámina neta, después se pasa a calcular la frecuencia de riego (Fr), con estos valores se calcula lámina bruta, después se calcula el volumen de riego, con estos valores se calcula el tiempo de riego, luego de tener todas las variables calculadas se procede a agruparlas conjuntamente con el tiempo que demora en desarrollarse la planta, es decir por cada mes se calcula todas las variables que pertenecen el calendario del riego. (León, 2012)

Para empezar con el proceso de cálculo se detalla los datos necesarios para la elaboración del calendario de riego.

- Profundidad radicular media en distintas fases del cultivo.
- Intervalo de humedad disponible del suelo.
- Caudal del sistema de riego.
- Medida del umbral.
- Velocidad del viento
- Distancia del terreno de siembra al barlovento
- Medida de humedad relativa
- Medida de capacidad de campo
- Medida de punto de marchitamiento permanente
- Tabla de referencia del libro 56 de la FAO

A continuación se describe el paso a paso del proceso:

Según el doctor Juan León en su libro “RIEGOS Y DRENAJES” el proceso para realizar un calendario de riego, se debe de realizar de la manera que se describe a continuación:

Paso 1.

DETERMINACIÓN DE LA EVAPOTRANSPIRACIÓN (ETO)

Se empieza por determinar la evapotranspiración, misma que se define como la cantidad de humedad que pierde el suelo debido a la evapotranspiración directa, está dada en milímetros sobre una unidad de tiempo (mm/t).(León, 2012).

La fórmula de la ETO es:

$$ETO = \text{evapotranspiración} * k_p$$

Ecuación 1: Cálculo de la evapotranspiración del suelo

Fuente:(León, 2012)

Realizado por: Torres, I. 2020

Donde:

ETO = evapotranspiración medida en el día.

Kp = Es un coeficiente que va a depender de tres factores (humedad relativa, velocidad del viento y la distancia del barlovento a la siembra) este coeficiente se lo toma de la tabla establecida por la FAO.

Paso 2.

DETERMINACIÓN DEL COEFICIENTE DEL TANQUE EVAPORIMETRO (KP)

Es un valor que se lo toma directamente de la tabla que ha establecido la FAO en su libro número 56 llamado Estudio riego y drenaje. En la página 121 está la tabla de la que podemos sacar el valor del Kp en función de la velocidad del viento (m/s^2), la humedad relativa (%) y la distancia del barlovento (m), y del tipo de suelo en el que se encuentre el tanque tipo A, puede ser con suelo con pasto o suelo desnudo. Estos tres valores que definen el Kp se los toma de la estación meteorológica, más cercana al área de siembra. (León, 2012)

Tanque Clase A	Caso A: Tanque situado en una superficie cultivada			Caso B: Tanque situado en un suelo desnudo				
	HR media	baja	media	alta	baja	media	alta	
Velocidad del viento (m s ⁻¹)	Distancia del cultivo a barlovento (m)	< 40	40-70	> 70	Distancia del barbecho a barlovento (m)	< 40	40-70	> 70
Baja	1	,55	,65	,75	1	,7	,8	,85
< 2	10	,65	,75	,85	10	,6	,7	,8
	100	,7	,8	,85	100	,55	,65	,75
	1 000	,75	,85	,85	1 000	,5	,6	,7
Moderada	1	,5	,6	,65	1	,65	,75	,8
2-5	10	,6	,7	,75	10	,55	,65	,7
	100	,65	,75	,8	100	,5	,6	,65
	1 000	,7	,8	,8	1 000	,45	,55	,6
Alta	1	,45	,5	,6	1	,6	,65	,7
5-8	10	,55	,6	,65	10	,5	,55	,65
	100	,6	,65	,7	100	,45	,5	,6
	1 000	,65	,7	,75	1 000	,4	,45	,55
Muy alta	1	,4	,45	,5	1	,5	,6	,65
> 8	10	,45	,55	,6	10	,45	,5	,55
	100	,5	,6	,65	100	,4	,45	,5
	1 000	,55	,6	,65	1 000	,35	,4	,45

Figura 1-1: Cuadro de referencia para determinación del Kp.

Fuente: (León, 2012)

Paso 3.

DETERMINACIÓN DE LA PERDIDA DE HUMEDAD EN EL SUELO (Etc)

La evapotranspiración del cultivo ETc es la pérdida de humedad en el suelo debido a dos situaciones la primera la evapotranspiración del agua contenida en el suelo y segundo la transpiración del agua contenida en las plantas. (León, 2012)

Se calcula de la siguiente manera:

$$ETc = ETO * kc$$

Ecuación 2: Cálculo de la evapotranspiración del cultivo

Fuente:(León, 2012)

Realizado por: Torres, I. 2020

Donde:

ETc es la evapotranspiración del cultivo (mm/día)

ETO = Evapotranspiración del suelo.

Kc = coeficiente del cultivo.

Paso 4.

DETERMINACIÓN DEL COEFICIENTE DEL CULTIVO (Kc)

Para determinar el coeficiente de cultivo Kc se tiene varios métodos pero para el caso de este estudio lo realizaremos con MacGrawn. El cual nos indica la siguiente formula:

$$Kc = 0.01335 + 0.04099 * C - 0.00040 * C^2$$

Ecuación 3: Cálculo de coeficiente cultivo

Fuente:(León, 2012)

Realizado por: Torres, I. 2020

Donde:

Kc = coeficiente de cultivo.

C = cantidad de días transcurridos del cultivo.

La variable C se calcula de la siguiente manera:

$$C = \frac{\text{cantidad de dias} * 100}{\text{dias de siembra cosecha}}$$

Ecuación 4: Cálculo de cantidad de días transcurridos de la siembra

Fuente:(León, 2012)

Realizado por: Torres, I. 2020

Paso 5.

DETERMINACIÓN DEL AGUA ÚTIL PARA EL RIEGO (AU)

$$AU = (CC - PMP) * Da * Z$$

Ecuación 5: Cálculo del agua útil

Fuente:(León, 2012)

Realizado por: Torres, I. 2020

Donde:

AU = Agua útil o lámina total de agua, disponible para las plantas, en la zona radicular.(León, 2012).

Z = Profundidad radicular de la planta (mm).

CC = Contenido de humedad a capacidad de campo (m/m³).

PMP = Contenido de humedad a punto de marchitamiento permanente (m^3/m^3).

Da = Densidad aparente del suelo (gr/cm^3).

DETERMINACIÓN DE LA LÁMINA NETA DE RIEGO (LN)

Con el fin de considerar la cantidad de agua disponible para la planta, que puede ser fácilmente extraída por las plantas, se determina la denominada lámina neta por medio de la siguiente ecuación:

$$LN = umbral * AU$$

Ecuación 6: Cálculo de la lámina neta de riego

Fuente:(León, 2012)

Realizado por: Isaac Torres

Donde:

LN = Lámina neta.

Umbral = es la sensibilidad del cultivo a la reducción del agua disponible en el suelo (fracción).(León, 2012)

AU = Agua útil (mm).

Paso 6.

DETERMINACIÓN DE LA FRECUENCIA DE RIEGO (FR)

Después de haber determinado la lámina neta de riego que el suelo es capaz de retener definido el umbral de riego según el tipo de cultivo, se continúa con la determinación de la frecuencia de riego por medio de la siguiente ecuación:

$$FR = \frac{LN}{ETc}$$

Ecuación 7: Cálculo de la frecuencia de riego

Fuente:(León, 2012)

Realizado por: Isaac Torres

Donde:

FR = Frecuencia de riego.

LN = Lámina neta.

ETc= Evapotranspiración del cultivo.

Paso 7.

DETERMINACIÓN DE LA LÁMINA BRUTA DE RIEGO (LB)

$$LB = \frac{ETc * FR}{Ef}$$

Ecuación 8: Cálculo de la lámina bruta de riego

Fuente:(León, 2012)

Realizado por: Isaac Torres

Donde:

LB = Lámina bruta de riego

ETc = Evapotranspiración del cultivo.

FR = Frecuencia de riego.

Ef = Eficiencia de aplicación del agua

Paso 8.

DETERMINACIÓN VOLUMEN DE RIEGO (VR)

$$VR = Lb * \text{área}$$

Ecuación 9: Cálculo del volumen de riego

Fuente:(León, 2012)

Realizado por: Isaac Torres

Donde:

VR = Volumen de riego a aplicar (litros)

Lb = Lámina bruta de riego (mm).

Área= espacio de terreno sembrado del cultivo (m²).

Paso 9.

DETERMINACION DEL TIEMPO DE RIEGO (TR)

$$TR = \frac{VR}{caudal}$$

Ecuación 10: Cálculo del tiempo de riego

Fuente:(León, 2012)

Realizado por: Isaac Torres

Donde:

TR = Tiempo de riego a proporcionar al cultivo (minutos)

VR = Volumen de riego a aplicar (litros)

Caudal = cantidad de agua del sistema de riego (litros/h).

Paso 10.

CREACIÓN DEL CALENDARIO DE RIEGO

Una vez que se ha calculado todas las variables necesarias para definir el calendario de riego se las organiza en una tabla, agregando la columna del mes, partiendo de la fecha de siembra se incrementa para cada calculo 30 días que equivalen un mes. Es decir por cada mes se debe de calcular todas las variables hasta que se acabe el tiempo desarrollo del cultivo. (León, 2012) Como se indica en la tabla 1-1 a continuación detallada:

Tabla 1-1: Formato del calendario de riego

Mes	ETO	KP	ETC	KC	AU	LN	FR	Lb	VR	TR
Marzo										
Abril										
Mayo										
Junio										
Julio										
Agosto										

Fuente: (León, 2012)

Realizado por: Torres, I. 2020

Como se observa el proceso de cálculo del calendario de riego por este método es secuencial, involucra varios pasos para determinar las variables una en función de la otra, utilizando fórmulas matemáticas, esto hace que los valores encontrados sean precisos, al final hay que volver a repetir el proceso por cada mes hasta que se complete el tiempo de desarrollo del cultivo,

1.3.2. Método gráfico

En este método se construye un gráfico de coordenadas, en el eje “X” se coloca el tiempo de duración del ciclo de la planta en días y en el eje “Y” los valores del requerimiento de riego del cultivo. (Cisneros, 2003), de igual manera que en el método analítico se debe de tener datos de entrada para realizar los cálculos necesarios, los mismos que se describen en el **gráfico 3-1**, para comprender este proceso se detalla a continuación un ejemplo:

Datos:

Cultivo: trigo

Duración desarrollo: 1 de diciembre al 15 de abril

MES	Duración	°T	Et' (cm)	Et' acum (cm)	P observ. (cm)	Pe* (cm)	Rr (cm)	Rr acum (cm)
DIC	1	16.8	3.35	3.35	2.47	1.6	1.75	1.75
ENE	1	15.8	8.81	12.16	2.98	2.16	6.65	8.40
FEB	1	16.8	13.75	25.91	0.25	0	13.75	22.15
MAR	1	18.4	15.70	41.61	0.68	0	15.70	37.85
ABR	0.5	21.2	5.96	47.57	0.0	0	5.96	43.81

* Pe fue calculada con el método de Doorenbos y Pruitt

Datos del suelo:

PROFUNDIDAD (cm)	CC (%)	PMP (%)	Da (gr/cm ³)	La (cm)
00 - 30	33.90	17.40	1.21	5.99
30 - 60	34.30	17.20	1.20	6.16
				Σ= 12.15

Figura 2-1: Datos para cálculo de calendario de riego de manera gráfica.

Fuente:(Cisneros, 2003)

Proceso de cálculo:

Paso 1.

APLICACIÓN DE LÁMINA DE RIEGO INICIAL

La lámina que se aplica en el primer riego es la necesaria para humedecer toda la profundidad radicular hasta capacidad de campo y que será la lámina que tenga que consumirse después del último riego.

Para el ejemplo:

Lámina de siembra (Ls) = 12.15 cm

Paso 2.

BALANCEO DE CARGAS EN LÁMINAS DE RIEGO

Los riegos subsecuentes tendrán que proporcionar una lámina total igual al requerimiento de riego menos la lámina aplicada en el primer riego.

Así sería:

$Rr - Ls = Lt$ $43.81 - 12.15 = 31.66 \text{ cm}$

Paso 3.

ESPECIFICACIÓN DE LÁMINA EN FUNCIÓN DE LA ETÁPA DE LA PLANTA

Está lámina total no puede proporcionarse en un sólo riego, ya que es necesario tomar en cuenta el desarrollo del cultivo, por lo tanto cada riego se aplicará cuando se haya llegado al nivel de humedad más bajo aceptable (punto crítico = Pc) (Pc fluctúa entre el 30-60% consumo de humedad de la La)

Para el ejemplo será: 60% consumo humedad

Así:

$$\begin{aligned} L_s * P_c &= L_r \\ 12.15 * 0.60 &= 7.30 \text{ cm} \end{aligned}$$

Paso 4.

DIVISIÓN DE LÁMINAS DE RIEGO

Para estimar el número de riegos subsecuentes al primer riego, se divide la lámina total obtenida en el 2º inciso entre la lámina de riego.

Así:

$$\begin{aligned} L_t / L_r &= R_1 \\ 31.66 / 7.30 &= 4.33 \text{ riegos*} \end{aligned}$$

Paso 5.

LÁMINA AJUSTADA

Calculado el número de riegos subsecuentes al primer riego se obtendrá la lámina ajustada que se aplicará a cada uno de ellos.

Se tiene:

$$\begin{aligned} L_t / R_a &= L_{rt} \\ 31.66 / 4 &= 7.91 \text{ cm} \end{aligned}$$

Paso 6.

GRAFICAR VALORES

Con estos valores se procede hacer la gráfica del programa de riego.

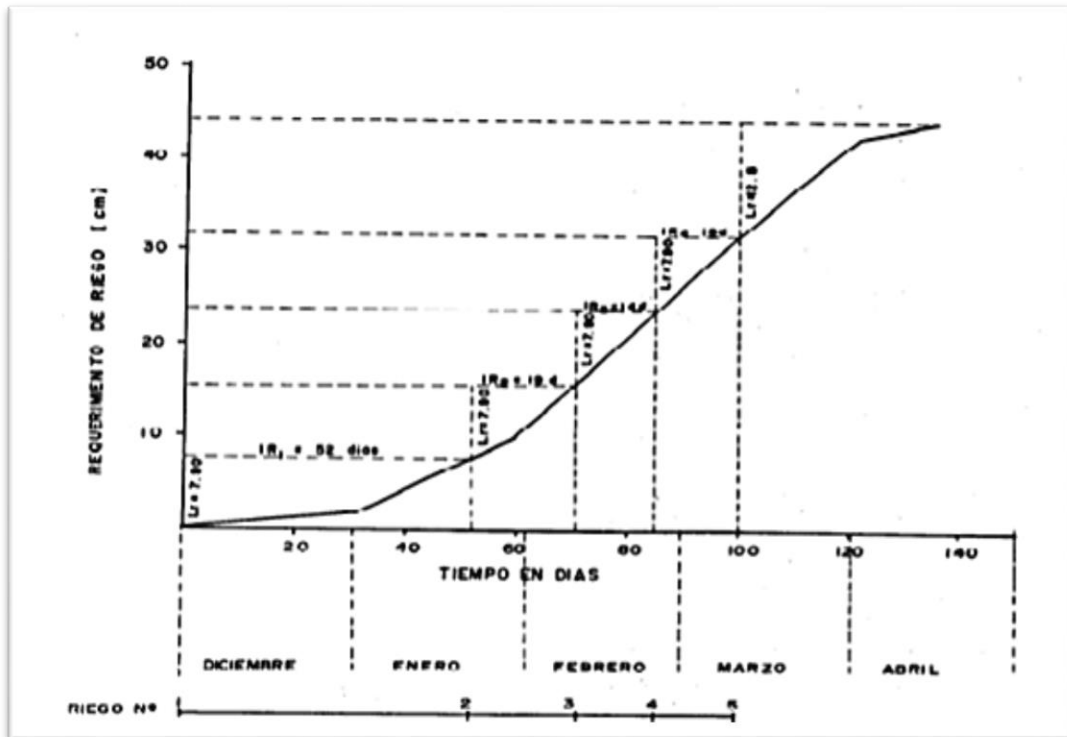


Figura 3-1: Representación del calendario de riego de manera gráfica.

Fuente:(Cisneros, 2003)

Este método resulta útil cuando no se dispone de herramientas de cálculo como aplicaciones o programas informáticos, o se desconoce el procedimiento para hacerlo de manera analítica. En general, es un método muy aproximado.(Cisneros, 2003)

El método gráfico permite elaborar el calendario de riego de manera aproximada, puede existir errores en los valores de la cantidad de agua y tiempo a regar, mientras que el método analítico por trabajar con fórmulas y cálculos matemáticos tiene un grado de exactitud mayor, por esta razón la mayoría aplicaciones informáticas que facilitan la realización del calendario de riego trabajan con este método como se describe a continuación.

1.4. Sistemas informáticos utilizados para elaborar un calendario de riego

Existen varias aplicaciones informáticas para calcular el calendario de riego, basándose tanto en el método analítico como el en método gráfico, los datos climáticos con los que trabajan son por lo general de la zona de donde se creó el software y para trabaja en otros entornos trabajan con datos meteorológicos aproximados. A continuación se describe dichas aplicaciones:

Aplicación / Sistema	Descripción	Método de trabajo
App del IVIA	Funciona como una herramienta para el cálculo de volúmenes de aplicación de agua y calibración de turboatomizadores. Cuenta también con un sistema de estación de avisos para el control de alternancia.	Gráfico
Cultivapp	Funciona como un cuaderno de campo y base de datos, en el cual se registran las actividades diarias, la aplicación de tratamientos, la administración de parcelas, y precios de productos en origen y destino.	Gráfico
Croptwat	Es un programa informático creado por la FAO (organización de naciones unidas para la agricultura y la alimentación), este programa trabaja en función de datos de suelo, clima y cultivos de los Estados Unidos, permite elaborar un calendario de riego para poder estimar cuanto y cuando regar, utiliza el método analítico para la elaboración del calendario de riego.	Analítico
PlantCare Pro	La misma proporciona información sobre dosis y aplicaciones necesarias dependiendo de las características particulares de la plaga y la enfermedad.	Gráfico
Agrobío	Es un manual interactivo que permite a los agricultores conocer los efectos secundarios que el uso de fitosanitarios puede causar en la fauna auxiliar del huerto o invernadero.	Gráfico
Yara CheckIt	Muestra los efectos de las deficiencias nutricionales en los cultivos a través de una biblioteca de imágenes. Una vez identificada la deficiencia, esta app proporciona bastante información al respecto: sus efectos, los suelos propensos, factores de riesgo y alternativas para el tratamiento	Gráfico
Agronómico Calculadora	Esta herramienta permite calcular más precisamente la cantidad de insumos que es necesario aplicar en una determinada área sembrada.	Gráfico
Aquadaia	Una app gratuita que permite crear calendarios de riego en donde se especifica la cantidad de agua necesaria en un cultivo para asegurar su crecimiento óptimo. Se trata de una app muy interesante para los agricultores debido a la precisión en sus cálculos, combinado con la sencillez de su uso	Analítico
Appgro	El escritorio virtual de Appgro le permite contar con toda la información de su producción, así como toda la información relevada a campo, para luego generar informes y tomar la mejor decisión en el menor tiempo posible y desde cualquier lugar.	Gráfico

Fuente: (Icaza & Rúaes, 2009)

Realizado por: Torres. I. 2020

Como se observa la mayoría de aplicaciones trabajan con el método gráfico es decir sus valores de respuesta están expuestos a errores, las aplicaciones Aquadaia y Croptwat trabajan con el método analítico por lo que sus valores de respuesta son más acertados, a continuación se describen estas aplicaciones:

1.4.1. Aquadaia

Es una aplicación móvil creada por la empresa Daia, sirve para varios cultivos, incluidos entre estos los jardines frutales del norte de Estados Unidos. El sitio web oficial de la empresa que desarrollo esta aplicación dice lo siguiente: La empresa Daia Intelligent Solutions acaba de lanzar Aquadaia, “una app gratuita que permite crear calendarios de riego en donde se especifica la cantidad de agua necesaria en un cultivo para asegurar su crecimiento óptimo. Se trata de una app muy interesante para los agricultores debido a la precisión en sus cálculos, combinado con la sencillez de su uso”.(iAgua, 2014)

1.4.2. Croptwat

Es un programa informático creado por la FAO (organización de naciones unidas para la agricultura y la alimentación), este programa trabaja en función de datos de suelo, clima y cultivos de los Estados Unidos, permite elaborar un calendario de riego para poder estimar cuanto y cuando regar, utiliza el método analítico para la elaboración del calendario de riego.

Después de haber revisado estos dos programas se puede notar que son creados con datos climatológicos del país en donde se los desarrolló. Para usarlos estos programas en cultivos de la provincia de Chimborazo trabaja con datos aproximados lo cual no garantiza que el calendario de riego tenga valores reales, por esta razón es necesario crear una aplicación web para la elaboración del calendario de riego que trabaje con datos y valores de los cultivos propios de la zona en la que el CER realiza sus actividades.

1.5. Aplicaciones Web

Se denomina aplicación web al software que reside en un ordenador, denominado servidor web, que los usuarios pueden utilizar a través de Internet o de una intranet, con un navegador web, para obtener los servicios que ofrezca.(Zofío Jiménez, 2013)

1.5.1. Características de las aplicaciones web

Las principales características con las que cuentan las aplicaciones web son las detalladas a continuación:

- La facilidad de acceso a la aplicación, esto se da porque solo es necesario un navegador web como cliente.

- No importa en qué sistema operativo se esté utilizando para navegar.
- La facilidad de actualización y mantenimiento, sin tener que redistribuir y reinstalar el software a miles de usuarios potenciales.(Zoffio Jiménez, 2013)

1.5.2. Ventajas y desventajas de las aplicaciones web

A continuación la **tabla 2-1** describe las ventajas y desventajas de una aplicación web:

Tabla 2-1: Ventajas y desventajas de las aplicaciones web

Ventajas	Desventajas
No es necesario ninguna instalación	Menos funcionalidades que las aplicaciones de escritorio
Compatibilidad	La disponibilidad depende de un tercero
Multiplataforma	
Consumo bajo de recursos	
Alta disponibilidad	

Fuente:(Cisneros, 2003)

Realizado por: Torres, I. 2020)

1.6. Arquitectura de la aplicación web

La arquitectura de software es la forma de trabajar en un sistema para desarrollar la aplicación web es muy importante, es lo primero que se debe en definir. (Zoffio Jiménez, 2013)

Una arquitectura de desarrollo de software debe de tener las siguientes características:

- Independencia de los frameworks.
- Testeable.
- Independencia de las interfaces de usuario.
- Independencia de las bases de datos.
- Independencia de algún componente externo (librerías)

Existen varias arquitecturas para el desarrollo de software entre las más utilizadas se las detalla a continuación en **la tabla 4-1:**

Tabla 3-1: Ventajas y desventajas de las aplicaciones web.

Arquitectura	Definición	Ventajas	Desventajas
Monolítica	Es la arquitectura de los primeros sistemas operativos constituidos por un solo programa compuesto as u vez por un	Eficiente ya que se producen pocos cambios de contexto	Difícil de depurar, un error en una función se puede dar a notar en otra distinta. Complejo a la hora de ampliar el sistema

	conjunto de rutinas enlazadas de tal forma que la una invocaba a la otra.		
Cliente / Servidor	Las cargas del software son distribuidas en dos partes independientes, los proveedores de los servicios (servidores) y los demandantes (cliente)	Centralización del control. Escalabilidad. Fácil mantenimiento. Tecnologías maduras y robustas.	Dependencia de respuesta en función de la calidad el servidor.
Arquitectura 3 capas	El objeto principal de esta arquitectura es separa la lógica de negocios de la lógica del diseño.	Escalabilidad. Tecnologías maduras.	Recursos a nivel de hardware muy altos. Altos costos de implementación.

Fuente:(Ramírez, 2009)

Elaborado: Torres, I. 2020

La aplicación debe ser una calculadora de calendarios de riego en línea, por esta razón como se puede observar la arquitectura Cliente /Servidor es la que más se ajusta a las necesidades del presente proyecto, por tener centralización del control, tecnologías maduras y robustas, escalabilidad y fácil mantenimiento, por estas razones se utiliza esta arquitectura para desarrollar la aplicación web.

En el grafico 4-1 se detalla de manera general la arquitectura Cliente/Servidor.

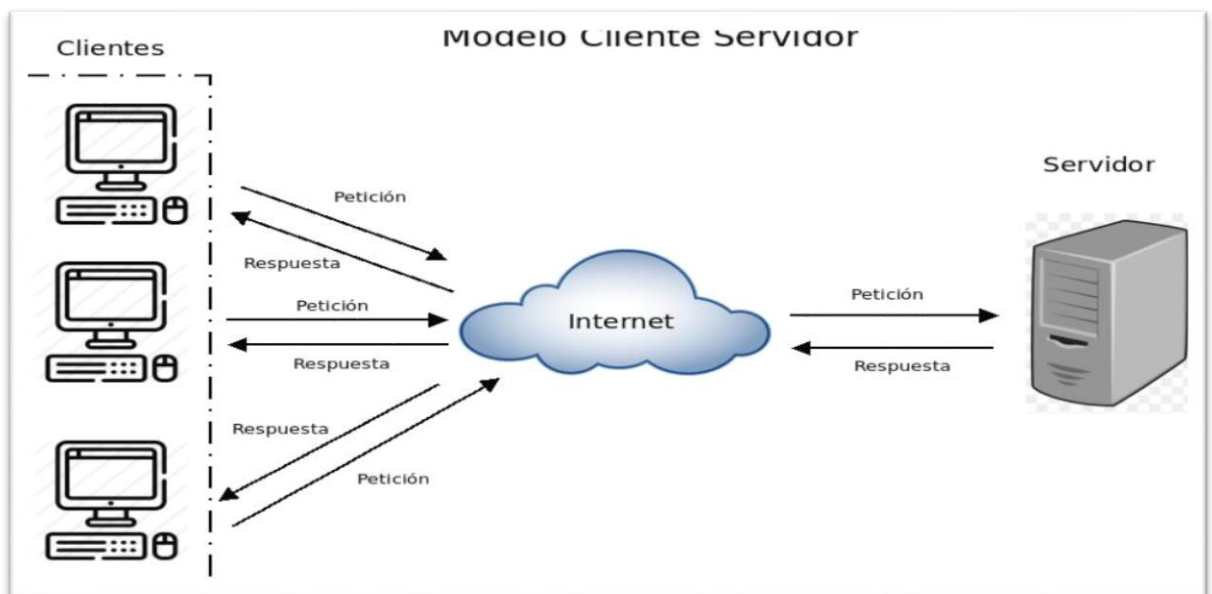


Figura 4-1: Arquitectura Cliente/Servidor.

Fuente:(Ramírez, 2009)

Realizado por: Torres, I. 2020

1.6.1. Elementos

Básicamente son dos elementos:

Cliente: Son los dispositivos electrónicos (computadoras, celulares, tablets) conectados a una red de internet.

Servidor: Es un equipo electrónico con mayores capacidades de procesamiento que se encarga de dar respuesta a las peticiones recibidas de los clientes.

1.6.2. Características

Según (Ramírez, 2009), las características principales de una arquitectura cliente/servidor son:

- Recursos unificados, dentro del servidor están todos los componentes y servicios que se ofrecen al cliente.
- Control de acceso a usuarios incrementando el nivel de seguridad.
- Escalabilidad.
- Gestión del comportamiento a nivel de servidor.

1.6.3. Ventajas y desventajas

Los beneficios e inconvenientes según (Ramírez, 2009) de la arquitectura Cliente/Servidor se detallan en la **tabla 5-1**.

Tabla 4-1: Ventajas y desventajas de arquitectura Cliente/Servidor

VENTAJAS	DESVENTAJAS
Centralización del control.	Dependencia de respuesta en función de la calidad el servidor.
Escalabilidad.	
Tecnologías maduras y robustas	
Fácil mantenimiento	

Fuente:(Ramírez, 2009)

Elaborado: Torres, I. 2020

La arquitectura cliente/servidor brinda una robusta base de desarrollo que se basa en brindar respuestas rápidas y precisas los clientes se decide utilizarla para el desarrollo del presente trabajo.

1.7. Patrón de diseño Modelo – Vista – Controlador MVC

Según la publicación de la revista digital de las tecnologías de la información y de las telecomunicaciones el patrón de diseño MVC tuvo sus orígenes en el año 1979 por Trygve Reenskaug. (Y. D. González & Romero, 2012). El objetivo de crear este patrón de diseño fue reducir el esfuerzo de programación necesario en la implementación de sistemas múltiples y sincronizados de los mismos datos. Sus características principales están dadas por el hecho de

que, el Modelo, las Vistas y los Controladores se tratan como entidades separadas; esto hace que cualquier cambio producido en el Modelo se refleje automáticamente en cada una de las Vistas.(Y. D. González & Romero, 2012)

Para representar de mejor manera la estructura de este patrón de diseño se ha elaborado la siguiente ilustración.

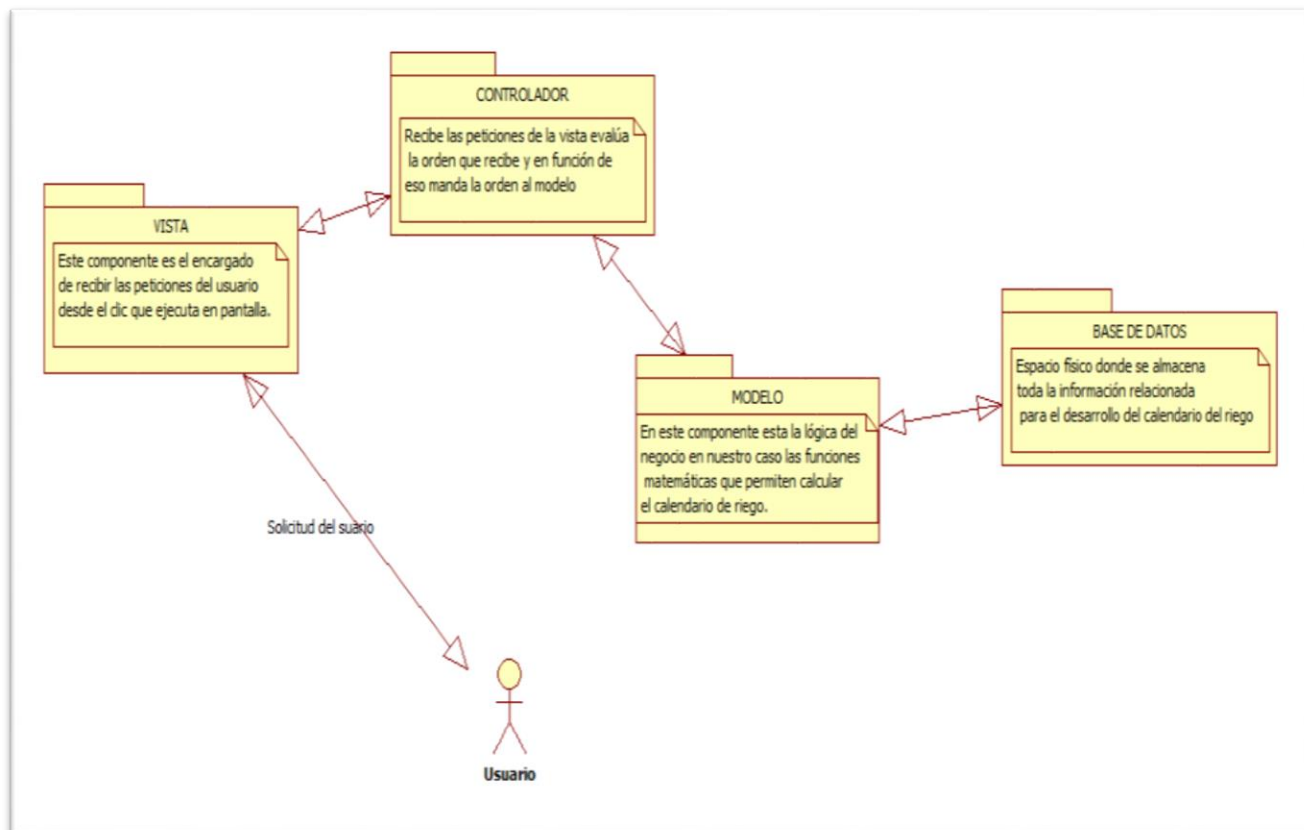


Figura 5-1: Patrón de diseño MVC.

Fuente:(Y. D. González & Romero, 2012)

Realizado por: Torres, I. 2020

1.7.1. Características del MVC

Este patrón de diseño se diferencia de otros patrones de modelo por contar con las siguientes características:

- Separación clara entre los componentes de un programa; lo cual permite su implementación por separado. (Y. D. González & Romero, 2012).
- Interfaz de Programación de Aplicaciones API (Application Programming Interface) muy bien definida; cualquiera que use el API, podrá reemplazar el Modelo, la Vista o el Controlador, sin aparente dificultad. (Y. D. González & Romero, 2012).

- Conexión entre el Modelo y sus Vistas dinámica; se produce en tiempo de ejecución, no en tiempo de compilación.(Y. D. González & Romero, 2012).

1.7.2. Elementos del MVC

Este patrón de diseño de software está compuesto por tres elementos, a continuación se enuncia los conceptos según (Y. D. González & Romero, 2012).

El **modelo** se encarga de:

- Acceder a la capa de almacenamiento de datos. Lo ideal es que el modelo sea independiente del sistema de almacenamiento. (Y. D. González & Romero, 2012)
- Define reglas de negocio (la funcionalidad del sistema).(Y. D. González & Romero, 2012).

Este elemento es el que se encarga del acceso a datos para facilitar esta tarea se utilizará el framework mencionado a continuación:

La **vista** es la encargada de:

- Recibir datos procesados por el controlador o del modelo y mostrarlos al usuario.(Y. D. González & Romero, 2012)
- Tienen un registro de su controlador asociado.(Y. D. González & Romero, 2012)

El **Controlador** se encarga de:

- Comunicar la vista y el modelo (Velasco et al., 2018, p. 57).
- Encargado de gestionar eventos estos pueden ser aquellos que se producen cuando un usuario.

1.8.Herramientas de desarrollo

Para desarrollar la aplicación web es necesario contar con las herramientas que permiten crear, probar y ejecutar las funcionalidades de la aplicación web.

1.8.1. Lenguaje de programación

Los lenguajes de programación se clasifican por niveles como son: nivel máquina, ensamblador y alto nivel. Los lenguajes de alto nivel tienen una sintaxis parecida al lenguaje entendido por el humano o conocido también como lenguaje natural (Sicilia, 2015, p. 12).

Existen varios lenguajes de programación web, los más utilizados para el desarrollo de aplicaciones web son PHP, Java, Python y Ruby, en la siguiente tabla se detalla cada uno de ellos con sus principales características.

Tabla 5-1: Cuadro comparativo de los lenguajes de programación.

Lenguaje	Paradigma	Características	Ventajas	Desventajas
Python	Orientado a objetos	Permite la creación de todo tipo de programas incluso sitios web, no requiere de compilaciones, un código interpretado.	Libre y código fuente abierto, lenguaje de propósito general, portable.	Los lenguajes interpretados suelen ser relativamente lentos.
Java	Orientado a objetos	Simple, Orientado a Objetos., distribuido, Interpretado, robusto, seguro, de arquitectura neutral, multihilo, con recolector de basura, portable, dinámico	Al ser orientado objetos permite su modularización. Permite la creación de aplicaciones de escritorio. Tiene soporte a desarrollo de aplicaciones móviles y web	Interpretado así que es relativamente lento en comparación con otros lenguajes
Ruby	Orientado a objetos	Se caracteriza por preocuparse más por el desarrollador que por la máquina, esto se traduce en dos cosas: a) La sintaxis es muy amigable y se lee casi como lenguaje natural. b) El performance no es igual que en lenguajes de más bajo nivel.	Diferencia entre mayúsculas y minúsculas, maneja excepciones, puede cargar librerías si el sistema operativo lo permite, portátil, desarrollo de bajo costo, software libre	Es relativamente nuevo y no cuenta con mucha documentación en comparación con otros lenguajes de programación. No está muy difundido en relación a otros lenguajes
PHP	Multiparadigma , imperativo, orientado a objetos, procedural y reflexivo	Utilizado para generar páginas web dinámicas, se ejecuta en el servidor, no se necesita la instalación de PHP en el lado del cliente versiones resiente permiten la POO, lenguaje de alto nivel.	Su sintaxis es muy similar a otros lenguajes, fácil, es un lenguaje muy popular tiene una comunidad muy grande, rápido, bastante documentado, libre y gratuito, no requiere definición de variables puede ser combinado junto a HTML	Como es un lenguaje que se interpreta en ejecución, para ciertos usos puede resultar un inconveniente que el código fuente no pueda ser ocultado. La ofuscación es una técnica que puede dificultar la lectura del código pero no la impide y, en ciertos casos, representa un costo en tiempos de ejecución.

Fuente:(Ramírez, 2009)

Realizado por: Torres, I. 2020

Se observa que el lenguaje de programación JAVA es simple, orientado a objetos, distribuido, interpretado, robusto, seguro, de arquitectura neutral, multihilo, portable y dinámico por estas características se lo ha elegido para desarrollar la aplicación web en el presente trabajo.

1.8.2. Lenguaje de programación JAVA

Java es un lenguaje de alto nivel interpretado y orientado a objetos en un principio fue creado por la empresa Sun Microsystems en 1995, su sintaxis es muy parecida al del lenguaje C++ y C, cuenta con protocolos de red estándar, además ayuda en la reutilización de código, es multiplataforma y multitarea permitiendo de esta manera la escalabilidad de la aplicación.(Prieto Sáez & Casanova Faus, 2016).

1.8.2.1. Características

Las principales características de Java según (Prieto Sáez & Casanova Faus, 2016).

- Diseñado para facilitar el trabajo en la WWW, mediante el uso de los programas navegadores.
- Inclusión en el lenguaje de un entorno para la programación gráfica (AWT y Swing) y el hecho de que su ejecución es independiente de la plataforma.
- Uso de lo que se denomina Máquina Virtual Java (Java Virtual Machine, JVM).

1.8.2.2. Funcionamiento de JAVA

A continuación se describe en el **grafico 6-1**, el funcionamiento de un programa desarrollado en lenguaje JAVA.

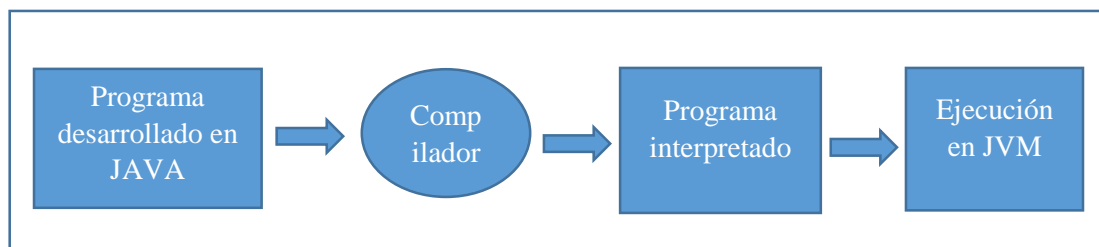


Figura 6-1: Funcionamiento del lenguaje JAVA.

Fuente:(Ramírez, 2009)

Realizado por: Torres, I. 2020

1.8.3. Herramientas de desarrollo para la capa cliente.

Para desarrollar los elementos de la capa del cliente se requiere un entorno de desarrollo, frameworks de interfaces de usuarios- Estos se los detalla a continuación:

1.8.4. Entorno de desarrollo integrado

Es un programa que permite editar y en algunos casos compilar y ejecutar el código del programa que se está desarrollando, a continuación en la tabla 6-1 se hace una comparativa entre los programas de desarrollo más comunes con sus características.

Tabla 6-1: Cuadro comparativo entre entornos de desarrollo integrado.

ENTORNO DE DESARROLLO INTEGRADO			
	Eclipse	Netbeans	Jcreator
Características	Posee un editor de texto con resaltado de sintaxis, la compilación es en tiempo real, refactorización	Soporte para usar estructuras Spring de tipo web, integración con diferentes sistemas gestores de bases de datos, mejor manera de compartir librerías entre proyectos, soporte completo para java.	Incluye un gestor de proyectos, vistas de clases, resaltado de sintaxis, interfaz personalizable.
Tamaño	150MB	79.5MB	5.04MB
Tiempo de ejecución	Pc. 1.8GHz y 6Gram (13.15 segundos)	Pc. 1.8GHz y 6Gram (11.22 segundos)	Pc. 1.8GHz y 6Gram (1.75 segundos)

Fuente:(Ramírez, 2009)

Realizado por: Torres, I. 2020

Se puede observar que el IDE netbeans tiene soporte para usar estructuras Spring de tipo web, integración con diferentes sistemas gestores de bases de datos, mejor manera de compartir librerías entre proyectos, soporte completo para java, por estas razones se lo ha elegido para ser el entorno de desarrollo del sistema web.

1.8.5. Entorno de desarrollo integrado Netbeans:

Es un conjunto de instrumentos que permite la edición, compilación, y ejecución de un programa todo esto en un mismo lugar. Vamos a citar la definición que nos da el autor Jesús Sánchez en su libro “Programación en JVA 2” el cual enuncia lo siguiente: NetBeans es un proyecto de creación de un entorno de libre distribución profesional patrocinado por Sun Microsystems.(Sánchez Allende et al., 2005)

1.8.6. Características de Netbeans

Las principales características del IDE netbeans según (Sánchez Allende et al., 2005) son las siguientes:

- Multiplataforma.
- Multilenguaje.
- Código abierto y gratuito.
- Apoyo de la comunidad con el soporte
- Recursos como documentación, video tutoriales, traductores de plugies o herramientas que se utilizan.
- Módulos.
- Fácil de usar y adaptable.
- Puglin que son herramientas extra para poder agregar algún componente, funciones extras.(M. González, s. f.)

1.8.7. Frameworks para desarrollo de interfaces

Un framework es una estructura de software conformada por diversos elementos personalizables para el desarrollo de una aplicación; permitiendo acelerar el proceso de desarrollo y reutilizar código (Valbuena, 2014, p. 15).

Un framework de maquetado es una herramienta que está conformado por elementos personalizables como son CSS, HTML y JavaScript, estos ayudan a reducir el tiempo en el desarrollo de tareas complejas de diseño de sitios web para el desarrollador.

Entre las características más importantes que poseen son de código abierto, poseen compatibilidad con varios navegadores, integración de librerías y diseño web adaptable (Jácome, 2016, p. 3) En la actualidad existen varios framework de maquetado entre los más usados para el desarrollo de interfaces de usuario se encuentran Bootstrap, Bulma, Picnic CSS y Primer CSS.

En la tabla 7-1, se detalla cada uno de ellos con sus principales características:

Tabla 7-1: Frameworks de desarrollo para interfaces.

Frameworks de desarrollo para interfaces	
Framework	Definición y características
Bootstrap	Permite crear interfaces que corren en los distintos navegadores, ya sea en una Tablet, en un celular o en una computadora a diferentes escalas y resoluciones de pantalla. Utiliza unas sólidas bases en el diseño de interfaces basado en los estándares CSS3/HTML5. Se levanta en absolutamente todos los navegadores hasta en Internet Explorer usando HTML Shim esto es para que reconozca las tags HTML5.
Bulma	Bulma es un framework CSS orientado para el desarrollo de páginas webs muy legibles y agradables, tanto para el visitante, como para el desarrollador. Tiene una documentación muy detallada y digna de alabanza, ya que están todas sus características muy bien documentadas, empezando por sus características más básicas y elementales, hasta llegar a las más complejas o las que incorporan mayor cantidad de código.
Picnic CSS	Tiene como objetivo poner a nuestra disposición una alternativa a Bootstrap ligera, sólo con CSS, y lo más visualmente agradable posible. Y lo consigue bastante bien. Echando un vistazo a su documentación, vemos que tiene un gran número de elementos para utilizar con este framework que ocupa apenas 38KB.
Primer CSS	No es sólo un framework, sino una guía de estilo y un conjunto de herramientas para crear páginas con el «look and feel» de GitHub. Desde recomendaciones HTML hasta fragmentos de código, pasando por componentes al estilo de los que existen en Bootstrap, siempre manteniendo la misma apariencia de la famosa plataforma de repositorios de código.

Fuente:(Prieto Sáez & Casanova Faus, 2016)

Realizado por: Torres, I. 2020

En la tabla 7-1 se puede observar que entre los frameworks evaluados el que brinda y tiene mayores ventajas al momento de desarrollar interfaces de usuario amigables y llamativas, es Bootstrap por estas razones se lo ha elegido para crear las interfaces en el presente trabajo.

Bootstrap

Es un framework front-end muy famoso creado por el grupo de Twitter (Tovar, 2013, p. 6) dedicado a la maquetación de sitios web este proporciona un conjunto de soluciones fáciles de usar, aplicar y ejecutar en muchos exploradores web los cuales están basados en los últimos estándares de desarrollo web siendo estos HTML5, CCS3 y JavaScript/Jquery, permitiendo acelerar el desarrollo de interfaces agradables e interactivas para el usuario. Este framework manipula un grid responsive de 12 columnas y trae complementos, plugins de JavaScript, tipografía y mucho más. (Cochran y Whitley, 2014, p. 7).

Características

Las características que posee el Framework Bootstrap web según (González y Galarza, 2016, p. 19), son las siguientes:

- Open Source y se puede encontrar en el repositorio de código GitHub.
- Optimización de tiempo en el desarrollo de interfaces adaptativas (responsive).
- Basado en el sistema de rejilla esto quiere decir que todos los elementos se adaptan al ancho de 12 columnas variando su tamaño acorde al terminal que se esté usando.
- Posee animaciones agregadas para diferentes componentes.
- A partir de Bootstrap 3.1 se incorporó módulos para CSS3 para 4 tamaños de equipos diferentes.

AJAX (Asynchronous JavaScript And XML)

Es un conjunto de tecnologías que permiten recargar solo una parte de la interfaz que visualiza el usuario.

1.8.8. Funcionamiento de AJAX

Cuando se carga una página web en un navegador esta viene cargada de toda la información que trae del backEnd, dividida en diferentes bloques que se observa visualmente, para comprender mejor imaginemos un formulario de consulta de datos personales, al ejecutar la consulta se vuelve a dibujar toda la página desde Headen hasta el Footer, y si queremos hacer una nueva consulta pues tendrá que dibujarse otra vez todo eso. De ahí nace la fabulosa idea de crear un mecanismo el cual permite volver a dibujar en la vista solamente el bloque en el que se visualiza la información. Este mecanismo se denomina AJAX nombre que viene dado por las iniciales de las palabras Asynchronous JavaScript and XML. Pero para entender mejor vamos a ver la definición que nos da Antonio Martín Sierra en su libro “AJAX en J2EE” en donde dice: El término AJAX hace referencia a un mecanismo de combinación de tecnologías y estándares de cliente, consistente en la solicitud asíncrona de datos al servidor desde una página Web y la utilización de éstos para actualizar una parte de la misma, sin obligar al navegador a realizar una recarga completa de toda la página.(Martín Sierra, 2014)

1.8.9. Tecnologías implicadas en el mecanismo de AJAX

AJAX es un mecanismo que une varias tecnologías para poder hacer una petición asíncrona desde el navegador hasta el servidor entonces el protocolo que sobresale en esta acción es el HTTP. Para comprender mejor esta tecnología citaremos lo que dice el autor Martín Sierra en el mismo libro anteriormente nombrado en el que mencionan lo siguiente acerca de XHTML y CSS: La interfaz gráfica de una aplicación AJAX es una página Web cargada en un navegador. XHTML y CSS son los dos estándares definidos por el W3C para la construcción de páginas Web; mientras que XHTML se basa en la utilización de un conjunto de marcas o etiquetas para la construcción de la

página, el estándar CSS define una serie de propiedades de estilo que pueden aplicarse sobre las etiquetas XHTML, a fin de mejorar sus capacidades de presentación.(Martín Sierra, 2014).

Otra tecnología involucrada en el mecanismo de AJAX es JavaScript. Las peticiones HTTP que lanza la página Web en modo asíncrono al servidor son realizadas por un bloque de script implementado con cualquier lenguaje capaz de ser interpretado por el navegador; este lenguaje es, en la gran mayoría de los casos, JavaScript.(Martín Sierra, 2014).

1.8.10. Herramientas para el desarrollo de la capa Servidor

Para el desarrollo de las aplicaciones web del lado del servidor se utiliza Java Server Page también conocidas como JSP, es una tecnología que permite representar de manera abstracta un servlet, cumpliendo exactamente las mismas funciones. De manera general los servlets son módulos escritos en Java que se ejecutan del lado del servidor para extender sus capacidades de respuesta a los clientes al utilizar las potencialidades de Java. A continuación se hace una comparación entre las tecnologías que existen para desarrollar aplicaciones web.

Tabla 8-1: Cuadro comparativo entre tecnologías de desarrollo web.

Tecnologías de desarrollo web	
Tecnología	Características
CGI	Es un mecanismo de comunicación entre el servidor web y una aplicación externa, esta aplicación puede estar desarrollada en casi cualquier lenguaje, este solo debe cumplir la condición de ser soportado por el servidor http, es común encontrar que la mayoría de las aplicaciones CGIs se encuentren desarrolladas con el lenguaje PERL. (Pérez Martínez, 2015)
Paginas dinámicas en servidor	Este nuevo enfoque consiste en insertar pequeños fragmentos de lógica de programación en la estructura HTML de la página, (Pérez Martínez, 2015)
Java Server Pages	JSP provee a los desarrolladores de aplicaciones web un entorno de desarrollo para crear contenidos dinámicos en el servidor usando código Java, encapsulando la lógica que genera el contenido de las páginas. Cuando se ejecuta una página JSP es traducida a una clase de Java, la cual es compilada para obtener un servlet. (Pérez Martínez, 2015)
Applets de Java	Un applet es un componente de software que corre en el contexto de otro programa, por ejemplo un navegador web. (Pérez Martínez, 2015)

Fuente:(Pérez Martínez, 2015)

Realizado por: Torres, I. 2020

Como se puede observar entre algunas de las tecnologías que permiten desarrollar aplicaciones web del lado del servidor, la que brinda mejores opciones es Java Server Page, la razón más importante por la que se utiliza esta tecnología es que se ejecuta en una máquina virtual de java JVM, es decir son archivos realizados en lenguaje java el cual se está utilizando para desarrollar el presente trabajo.

A continuación en **figura 7-1**, se describe el comportamiento de los JSP.

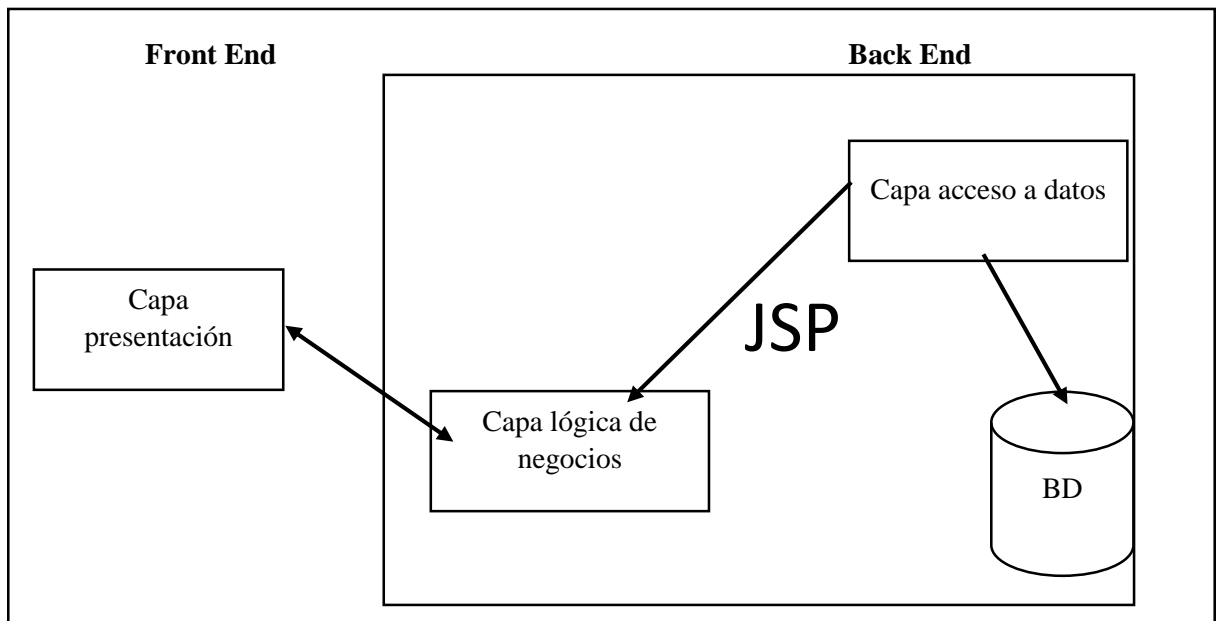


Figura 7-1: Funcionamiento una JSP.

Fuente:(Pérez Martínez, 2015)

Realizado por: Torres, I. 2020

1.8.11. Características de JSP

Las paginas server de java JSP, cuentan con las siguientes características:

- Orientado al desarrollo de páginas web en el lenguaje Java.
- Lenguaje Multiplataforma.
- Se ejecuta en el lado del servidor.
- Similar a ASP.NET en la potencia de los desarrollos.
- Necesita para su funcionamiento un servidor Web.
- Permite separar el código dinámico del estático.
- Las páginas se compilan en la primera petición.
- La extensión de archivos es JSP.
- El código JSP se puede incrustar en HTML.
- Utiliza el motor de servlet de Java.(Y. D. González & Romero, 2012)

1.9. Bases de datos

Una base de datos es una colección de información almacenada de manera electrónica a la que se puede acceder por medio de un programa gestor, esta información está clasificada por archivos, registros, ficheros, etc. (Ginesta & Pèrez, 2006)

A continuación en la **tabla 10-1** se presenta los diferentes sistemas gestores de bases de datos (SGBD) conjuntamente con sus características.

Tabla 9-1: Sistemas gestores de bases de datos.

SGBD	Características	Ventajas	Desventajas
MySql	Es de la propiedad de Oracle, licencia GLP/licencia comercial.	Integración de transacciones, diferentes motores de almacenamiento, instalación sencilla.	No tiene soporte, capacidad de almacenamiento limitada.
PostgreSQL	Posee la extensión POSTGIS para bases de datos espaciales.	Código abierto y gratuito, multiplataforma, almacena una gran cantidad de datos, permite transacciones, disparadores y afirmaciones.	Respuesta lenta, requiere un hardware potente, no es intuitivo al momento escribir las ordenes.
Oracle	Tiene su propio lenguaje PL/SQL, puede alojar bases de datos de gran tamaño.	Es el SGBD más usado a nivel mundial, es intuitivo y fácil de usarse.	Precio muy elevado, es manipulado por trabajadores de Oracle
Access	Pertenece a Microsoft, métodos simples y directos, interfaz amigable y gráfica, formularios para trabajar con la información.	Accesible para personas con poco conocimiento en bases de datos, crea varias vistas para una misma información.	No es multiplataforma, no funciona con bases de datos de gran tamaño ni a nivel de registros ni a nivel de usuarios.

Fuente: (Ginesta & Pèrez, 2006)

Realizado por: Torres, I. 2020

Como observa en la tabla comparativa, PostgreSQL es el que presenta mayores beneficios por las razones que es de código abierto y gratuito, multiplataforma, almacena una gran cantidad de datos, permite transacciones, disparadores y afirmaciones, es por estas razones que se ha elegido a PostgreSQL como sistema gestor de base de datos para el desarrollo de la aplicación web.

1.10. PostgreSQL

Es un gestor de base de datos de código abierto, manipula un modelo cliente/servidor, utiliza multiprocesos lo cual sirve de apoyo para la escalabilidad de una aplicación web, se encuentra disponible en manera libre, tiene un costo bajo para su mantenimiento y posee licencia BSD

1.10.1. Características del sistema gestor de base de datos SGBD PostgreSQL

Las principales características de PostgreSQL según (Ginesta & Pèrez, 2006) son las siguientes:

- Cuenta con un rico conjunto de tipos de datos, permitiendo además su extensión mediante tipos y operadores definidos y programados por el usuario.(Ginesta & Pèrez, 2006).
- Sus opciones de conectividad abarcan TCP/IP, sockets Unix y sockets NT, además de soportar completamente ODBC.(Ginesta & Pèrez, 2006)
- Puede extenderse con librerías externas para soportar encriptación, búsquedas por similitud fonética (soundex), etc.(Ginesta & Pèrez, 2006)

1.11. Metodología para gestión de proyecto

El control de las tareas y avances del proyecto se lo ha llevado acabo con una metodología ágil, ya que el proyecto tenía tendencia a cambiar de requerimientos en cualquier momento y había que ajustarse a dichos cambios.

Tabla 10-1: Cuadro comparativo metodologías ágiles.

XP	SCRUM
Se centra en la programación y creación del producto	Está enfocada a la administración del proyecto.
Se sigue estrictamente el orden de la prioridad de las tareas definidas por el cliente	Puede modificar el orden de las prioridades establecidas por el Product Owner (cliente)
Los miembros del trabajo deben de obligatoriamente trabajar en parejas.	Cada miembro del trabajo puede trabajar de manera individual
Estructura más cambiante y menos organizada	Posee una estructura jerárquica y organizada
Las iteraciones se entregan en 1-3 semanas	Las iteraciones se entregan en 2-4 semanas

Fuente: (Monte Galiano, 2016)

Realizado por: Torres, I. 2020

En relación a la metodología ágil Xtreme Programing XP, que exige la programación en parejas SCRUM de su parte la acepta pero no la exige, es decir se puede trabajar de manera individual, además SCRUM se adapta fácilmente a cambios o sugerencias de último momento ya que como sus Sprints son de una duración de tiempo corto. Algo adicional que nos brinda SCRUM es que al finalizar las tareas o en si el proyecto hay una retroalimentación que nos permite aprender de las experiencias transcurridas durante el desarrollo del proyecto.

1.11.1. Definición de SCRUM

Es un conjunto de técnicas que permite gestionar de una manera más ordenada las actividades y tareas relacionadas en un proyecto. Scrum está basado, por un lado, en la teoría del control empírico de procesos para la gestión de sistemas adaptativos complejos.(Monte Galiano, 2016)

Los pilares que son fundamentales en el concepto de esta metodología los detalla este mismo autor a continuación:

Transparencia: los aspectos significativos del proceso tienen que ser conocidos por todo aquel que participa, lo cual conlleva que estos aspectos estén definidos mediante un estándar común, de forma que todo el mundo tenga la misma percepción de las características de cada aspecto (por ejemplo, la definición de acabado).(Monte Galiano, 2016)

Inspección: todo proceso persigue un objetivo y, para llegar a ese objetivo, hace falta que los participantes en el proceso evalúen de manera continua sus resultados, y el proceso mismo, para detectar posibles desviaciones tan pronto como sea posible.(Monte Galiano, 2016)

Adaptación: cuando se detecta una desviación, la respuesta debe ser la adaptación; es decir, la adopción de acciones o planes que, o bien ayuden a corregir la desviación, o bien reconfiguren el objetivo.(Monte Galiano, 2016)

Para poder lograr un producto de calidad aceptable es necesario cumplir con un proceso, el objetivo del mismo es tener un mejoramiento continuo de la tarea que se esté realizando, las fases involucradas en este proceso los detallamos a continuación:

Planificar.- En esta fase se crea un cronograma en el que establecemos que vamos a hacer y como lo vamos a hacer.

Ejecución.- Hacer lo que se ha planificado anteriormente.

Verificación.- En esta parte se evalúa si se ha alcanzado el objetivo de la tarea que se ha realizado.

Retroalimentación.- Aquí se hace una recopilación de la tarea realizada, para poder mejorar en otra ocasión que tengamos que hacer una tarea similar.

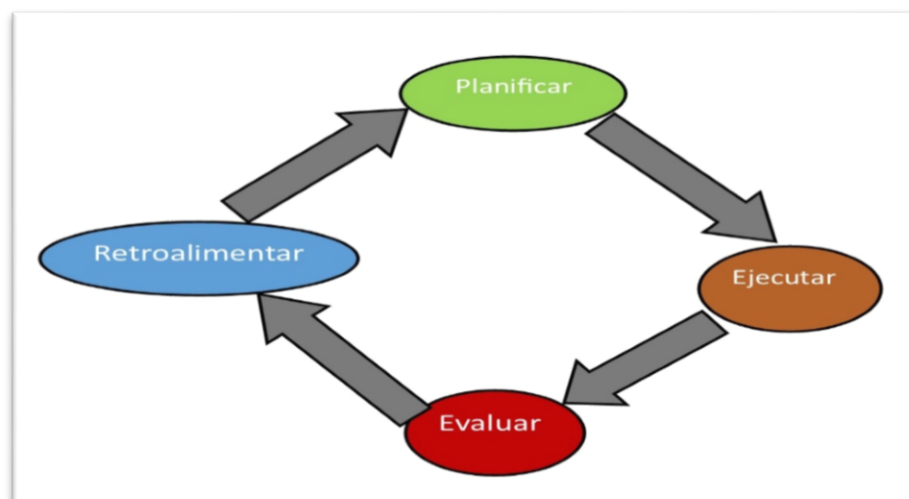


Figura 8-1: Funcionamiento de framework Hibernate JPA

Fuente:(Monte Galiano, 2016)

Realizado por: Torres, I. 2020

1.11.2. Perspectivas de SCRUM

A continuación enlistamos según Josep Monte los principales objetivos que persigue SCRUM con su propuesta:

Satisfacción del cliente: el objetivo último es la satisfacción del cliente. El cliente tiene que obtener lo que quiere y tiene que sentir que el producto que le damos es útil.(Monte Galiano, 2016).

Receptividad ante el cambio de requerimientos: los proyectos no son estáticos, cambian cada día. Nuestro trabajo diario tiene que prever y asumir este hecho.(Monte Galiano, 2016)

Trabajar enfocado en el producto, proyecto o servicio: la finalidad es la creación de un producto útil, por encima del método empleado.(Monte Galiano, 2016)

Desarrollo sostenible: la creación del producto tiene que ser posible dentro de un marco de trabajo que sea favorable a todo el mundo.(Monte Galiano, 2016)

Comunicación directa entre personas: debe favorecer la comunicación cara a cara por encima de otros medios de comunicación porque, si hay compromiso de todas las partes, se favorece la adopción de responsabilidades.(Monte Galiano, 2016)

Orientación a la excelencia: el objetivo no es crear productos porque sí, sino crear productos incrementales que mejoren en calidad cada día.(Monte Galiano, 2016)

Adaptabilidad: como los proyectos cambian, es necesario adaptarse a este cambio y hacer propuestas que adapten el proyecto a la nueva situación. La adaptabilidad sólo es posible si el equipo es adaptable.(Monte Galiano, 2016)

Tabla 11-1: Roles, artefactos y actividades dentro de la metodología SCRUM.

Rol	Artefactos	Actividades
Equipo de trabajo (Scrum team) <ul style="list-style-type: none"> • Product owner • Scrum Master • Development team • Stakeholders 	Pertenecientes al proyecto <ul style="list-style-type: none"> • Product backlog • Sprint backlog • Graphs • Impediment Backlog 	<ul style="list-style-type: none"> • First Sprint • Sprint • Sprint planning (planificación de tareas) • Daily meeting (reunión diaria) • Sprint review (revisión de lo ejecutado) • Sprint retrospective
	Pertenecientes al Sprint <ul style="list-style-type: none"> • Scrum board • Incidence backlog • Parking backlog 	

Fuente:(Monte Galiano, 2016)

Realizado por: Torres, I. 2020

Luego de enlistar los roles, actividades y artefactos principales que están inmersos dentro de SCRUM, podemos ahora definir estos.

Los roles determinan el reparto de responsabilidades en un proyecto. Así vemos reflejado el equipo de trabajo (Scrum team) que lleva el peso del desarrollo del producto. El negocio, representado por el product owner, que junto al Scrum master conforman el organismo de gobierno del proyecto. Y, por último, los usuarios clave (stakeholders), conocedores del negocio y la necesidad. (Monte Galiano, 2016)

Los **artefactos** conforman el equipo de herramientas con el cual el equipo de trabajo puede ejecutar y controlar el proyecto. Basado esencialmente en listas de control y el omnipresente Scrum board. (Monte Galiano, 2016)

Las **actividades** conforman los tiempos de ejecución del proyecto. La división de este en ciclos auto organizados y evaluables, con el objetivo de proporcionar en todo momento una visión sobre el producto que se está construyendo.(Monte Galiano, 2016)

Finalmente podemos resumir todas las fases, roles, actividades y artefactos de SCRUM en el siguiente ilustración.

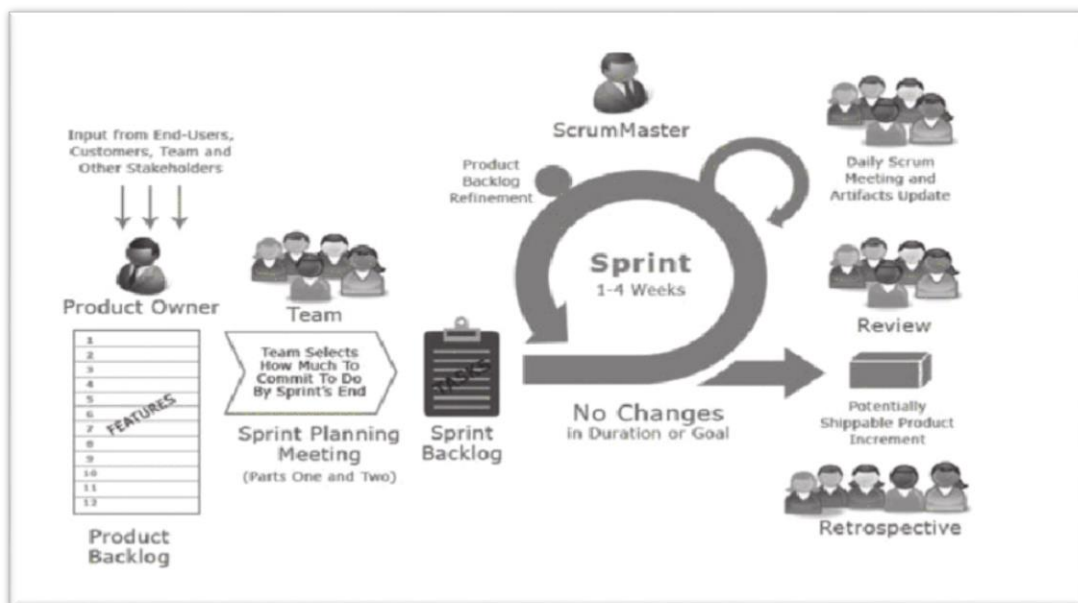


Figura 9-1: Resumen de actividades y artefactos SCRUM

Fuente:(Monte Galiano, 2016)

Realizado por: Torres, I. 2020

1.12. Evaluación de calidad en aplicaciones web

La calidad de un producto software se define como el grado que alcanza el producto en satisfacer las necesidades iniciales del cliente. Para alcanzar lo dicho se toma en cuenta la funcionalidad, rendimiento, seguridad, mantenibilidad, etc. Estas son llamadas categorías y dentro de las mismas

hay más métricas o parámetros a considerar denominadas subcategorías (Coral Calero et al., 2010).

Existen varias normas y estándares para medir la calidad de un producto software, las más utilizadas se las presenta en siguiente cuadro comparativo donde se indica el organismo que regula la norma, el objetivo que persigue.

Tabla 12-1: Normas y estándares para medir la calidad en un software.

Estándares y Normas	Organismo regulador	Orientación
CMMI	(SEI) Software Engineering Institute	Mejora de procesos de construcción de software y proyectos TI.
PSP	ISO	Permite estimar cuánto se tarda un individuo en realizar una aplicación de software
PSP-TSP	ISO	Predice el tiempo y tamaño del software Administración de calidad.
ISO 25000	ISO	Establecen un modelo de calidad para el producto del software, además de definir la evaluación de la calidad del producto.
IEEE	IEEE	Serie de documentación para el desarrollo de software y proyectos de TI
TSP	Team Software Process	Es un método de establecimiento y mejora del trabajo en equipo para procesos de software
SPICE	Programa de simulación con énfasis en circuitos integrados	Es una importante iniciativa internacional para apoyar el desarrollo de una Norma Internacional para la Evaluación de procesos del software
MOPROSOFT	ISO	Norma mexicana, basada en procesos para las industrias de software, la cual sirve para estandarizar operaciones y prácticas en gestión de ingeniería de software

Fuente: (Arcieniega, 2016)

Realizado por: Torres, I. 2020

Como se observa la norma 25000 permite establecer un modelo de calidad para el producto del software, además de definir la evaluación de la calidad del producto que es lo que se va a evaluar en el presente trabajo.

Para medir la calidad de la aplicación web que permite determinar un calendario de riego y la eficiencia en el proceso de cálculo, se ha utilizado la norma ISO/IEC 25010.

El modelo de la calidad de la norma ISO/IEC está constituido por ocho características como se detalla a continuación en la tabla 14-1:

Tabla 13-1: Elementos de la calidad del producto software.

CALIDAD DEL PRODUCTO SOFTWARE NORMA ISO/IEC 25010	
Categoría	Sub-categoría
Adecuación Funcional	Compleitud, corrección, pertinencia.
Eficiencia de desempeño	Comportamiento temporal, Utilización de recursos, capacidad.
Compatibilidad	Coexistencia, interoperabilidad.
Usabilidad	Inteligibilidad, aprendizaje, operatividad, protección, estética, accesibilidad.
Fiabilidad	Madurez, disponibilidad, tolerancia a fallos, capacidad de recuperación.
Seguridad	Confidencialidad, integridad, no repudio, autenticidad, responsabilidad.
Mantenibilidad	Madurabilidad, reusabilidad, analizabilidad, capacidad de ser modificado, capacidad de ser probado.
Portabilidad	Adaptabilidad, facilidad de instalación, capacidad de ser reemplazado.

Fuente: (Coral Calero et al., 2010)

Realizado por: Torres, I. 2020

Para medir la calidad del producto software en el presente trabajo se toma como referencia a la característica eficiencia de desempeño, la cual se define a continuación.

Eficiencia de desempeño: Esta característica está representando el desempeño en función de los recursos utilizados bajo específicas condiciones. (Coral Calero et al., 2010)

Las sub-características o métricas que pertenecen a esta característica son: utilización de recursos, comportamiento temporal y capacidad. A continuación se describe cada una:

El comportamiento temporal: son los tiempos de respuesta y procesamiento de un sistema informático cuando está realizando alguna determinada tarea.(Coral Calero et al., 2010)

Utilización de recursos: Es la cantidad y los tipos de recursos físicos que se han utilizado mientras el sistema desarrolla una tarea específica.(Coral Calero et al., 2010).

Capacidad: Grado en que los límites máximos de un parámetro de un producto o sistema software cumplen con los requisitos.(Coral Calero et al., 2010)

De estas sub-características, se mide el comportamiento temporal y la utilización de recursos la capacidad no se mide por la razón que la aplicación controla a nivel de base de datos los valores ingresados en los campos del formulario y no ejecuta el proceso si existe un valor fuera de lo establecido que no sea coherente para el cálculo.

CAPÍTULO II

2. MARCO METODOLÓGICO

Luego de analizar las aplicaciones web existentes para el cálculo del calendario de riego y el contexto para el cual fueron creados, se evidencia que es necesario desarrollar una aplicación web para el Centro Experimental del Riego de la ESPOCH, la misma que tiene como propósito mejorar el proceso de elaboración del calendario del riego mediante la reducción de tiempos.

2.1. Diseño de la investigación.

2.1.1. Tipo de investigación

Para comprender el proceso de la elaboración del calendario de riego se utiliza la investigación aplicada, ya que el objetivo de este tipo de investigación es aplicar los conocimientos adquiridos durante el tiempo de formación académica, combinando estos conocimientos con el área tecnológica.

2.1.2. Método de investigación.

En el presente trabajo en el área de investigación se utilizan los métodos Análisis – Síntesis e Inductivo – Deductivo, mismos que se definen a continuación:

- Método Análisis – Síntesis

En esta etapa de investigación se analiza cómo el Centro Experimental del riego realiza el proceso de la elaboración del calendario de riego, para comprender el cómo se desarrollan uno a uno los pasos involucrados en el mismo. Luego se emplea la síntesis para enlistar la información necesaria para elaborar el marco teórico en el cual se reflejan las herramientas utilizadas para el desarrollo de la aplicación web.

- Método Inductivo – Deductivo

Se parte de que el método inductivo inicia evaluando el funcionamiento de la aplicación web, es decir va de lo particular a lo general esto orienta a establecer las conclusiones del proyecto técnico que se redacta en el capítulo 3 denominado Marco de resultados, discusión y análisis de resultados.

Por el otro lado el método deductivo inicia en el conocimiento general y termina en el particular esto orienta a enunciar las recomendaciones del presente proyecto, mismas que surgieron en función de las conclusiones establecidas por medio del método inductivo.

2.1.3. Técnicas de Investigación

Las técnicas empleadas para la recolección de la información para poder desarrollar el presente proyecto técnico, es la observación y revisión de la documentación, como se menciona a continuación:

- Entrevista al Doctor Juan Eduardo León Ruiz director del Centro Experimental del Riego para determinar los requerimientos de la aplicación web.
- Observación de los pasos del proceso para la elaboración del calendario de riego.
- Observación y registro del tiempo que requiere cada paso, por medio de un cronómetro.
- Revisión del libro “Riegos y Drenajes” de la autoría del Doctor León en donde se describe de manera más detalla el proceso de la elaboración de calendario de riego.

2.2. Determinación del proceso de la elaboración del calendario de riego.

A continuación se detalla el proceso que realiza en Centro Experimental del Riego para elaborar un calendario de riego.

2.2.1. Gestión de los datos: Relación – Agua – Suelo – Planta – atmósfera (RASPA)

En la actualidad el Centro Experimental del Riego de la ESPOCH, realiza de manera manual el cálculo de las variables del calendario de riego, es decir va generando variable por variable una en consecuencia de otra con la ayuda de una calculadora, el técnico tiene en cuenta la gestión de la relación-agua-suelo-planta-atmósfera, empieza calculando la evapotranspiración (ETo), con este valor se procede a calcular el coeficiente del tanque de cultivo (Kp), con estos valores se pasa a calcular la pérdida de humedad en el suelo (ETc), luego se calcula el coeficiente de cultivo (Kc), con estos valores se procede a calcular el agua útil (Au), luego se calcula la lámina neta, después se pasa a calcular la frecuencia de riego (Fr), con estos valores se calcula lámina bruta, después se calcula el volumen de riego, con estos valores se calcula el tiempo de riego, luego de tener todas las variables calculadas se procede a agruparlas conjuntamente con el tiempo que demora en desarrollarse la planta, es decir por cada mes se calcula todas las variables que pertenecen el calendario del riego.

2.2.2. Gestión del Calendario de riego

Anotando todos estos valores obtenidos en hojas físicas, luego de esto pasa a la gestión del calendario de riego, para esta actividad se crea una tabla en las que se agrupa las variables teniendo en cuenta la fecha que se realiza la siembra, realizando el cálculo de cada una de las variables por cada mes dentro del tiempo del desarrollo del cultivo.

Estas actividades las realizan los técnicos de investigación, los estudiantes, agricultores (capacitados) o docentes del CER, hay que mencionar que este proceso se realiza para cada siembra diferente.

A continuación se representa de manera gráfica el proceso del cálculo del calendario de riego:

En la **gráfico 1-2** se representa de manera gráfica los procesos que se describen en la **tabla 1-2**, el modelado de este proceso se lo ha realizado con herramienta de procesos Bizagi Modeler.

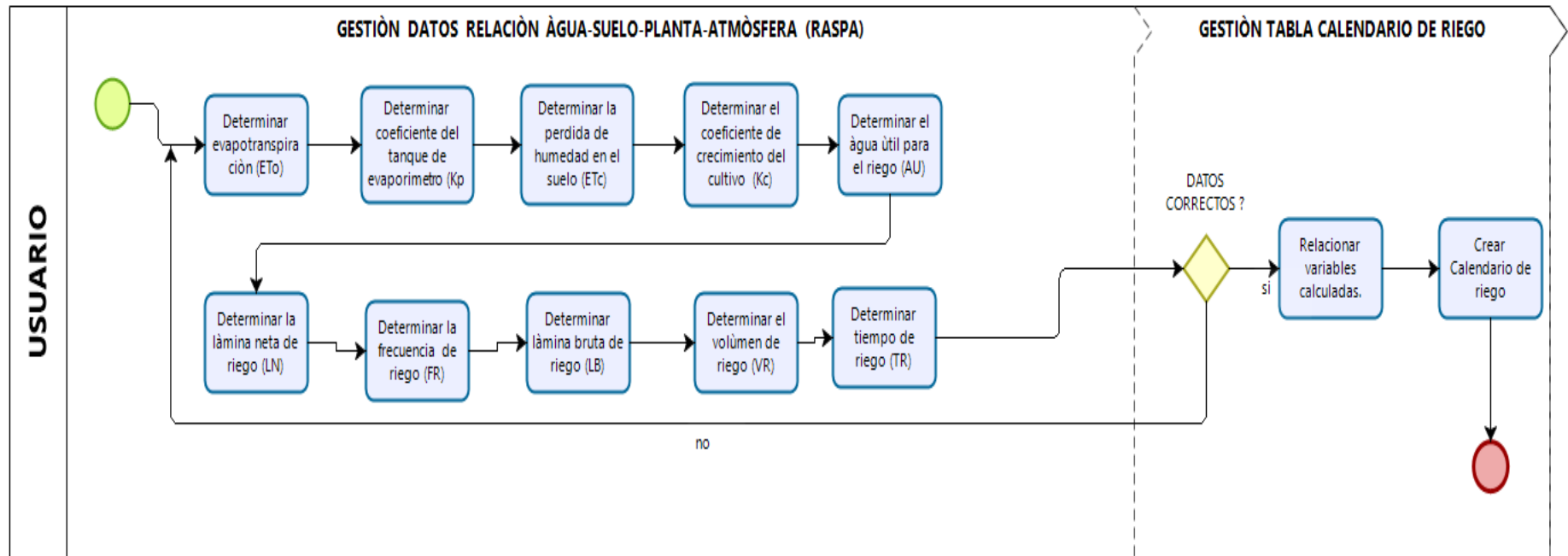


Gráfico 1-2: Diagrama de proceso de elaboración del calendario de riego.

Realizado por: Torres, I. 2020

En la tabla 1-2: se describe el paso a paso del proceso de la elaboración del calendario de riego:

Tabla 1-2: Descripción de pasos para la elaboración del calendario de riego manual

Nº	Actividades	Descripción	Responsable	Resultado
1	Determinar Evapotranspiración (ETo)	El investigador debe de calcular matemáticamente la ETo	Investigador	Establecer el valor de la (ETo)
2	Determinar el coeficiente del tanque evaporímetro (KP)	El investigador debe de calcular matemáticamente el KP	Investigador	Establecer el valor del (KP)
3	Determinar la pérdida de humedad en el suelo (ETc)	El investigador debe de calcular matemáticamente el (ETc)	Investigador	Establecer el valor de la (ETc)
4	Determinar el coeficiente de crecimiento del cultivo (Kc)	El investigador debe de calcular matemáticamente el (Kc)	Investigador	Establecer el valor del (Kc)
5	Determinar el agua útil para el riego (AU)	El investigador debe de calcular matemáticamente la (AU)	Investigador	Establecer el valor de la (AU)
6	Determinar la lámina neta de riego (LN)	El investigador debe de calcular matemáticamente la (LN)	Investigador	Establecer el valor de la (LN)
7	Determinar la frecuencia de riego (FR)	El investigador debe de calcular matemáticamente la (FR)	Investigador	Establecer el valor de la (FR)
8	Determinar la lámina bruta de riego (LB)	El investigador debe de calcular matemáticamente la (LB)	Investigador	Establecer el valor de la (LB)
9	Determinar el volumen de riego (VR)	El investigador debe de calcular matemáticamente el (VR)	Investigador	Establecer el valor del (VR)
10	Determinar el tiempo de riego (TR)	El investigador debe de calcular matemáticamente la (TR)	Investigador	Establecer el valor del (TR)
11	Relacionar variables calculadas.	Unir las variables calculadas conjuntamente con el tiempo de desarrollo del cultivo	Investigador	Calendario de riego

Realizado por: Torres, I. 2020

A partir de este proceso, se propone realizar estas mismas actividades pero de manera automática, es decir el usuario solo ingresa en un formulario los datos necesarios para calcular el calendario de riego, atendiendo a la relación-agua-suelo-planta-atmósfera RASPA.

Luego de esto presiona un botón y la aplicación web automáticamente gestiona estos datos ingresados y calcula todas las variables, después se gestiona la generación del calendario de riego, presentándole al usuario por medio de interfaz la tabla con las variables calculadas y agrupadas, dando la opción de descargar e imprimir dicho reporte.

A continuación se representa de manera gráfica el proceso de generare un calendario de riego de manera automatizada:

En la **gráfica 2-2** se representa de manera gráfica los procesos que se describen en la **tabla 2-2**, el modelado de este proceso se lo ha realizado con herramienta de procesos Bizagi Modeler.

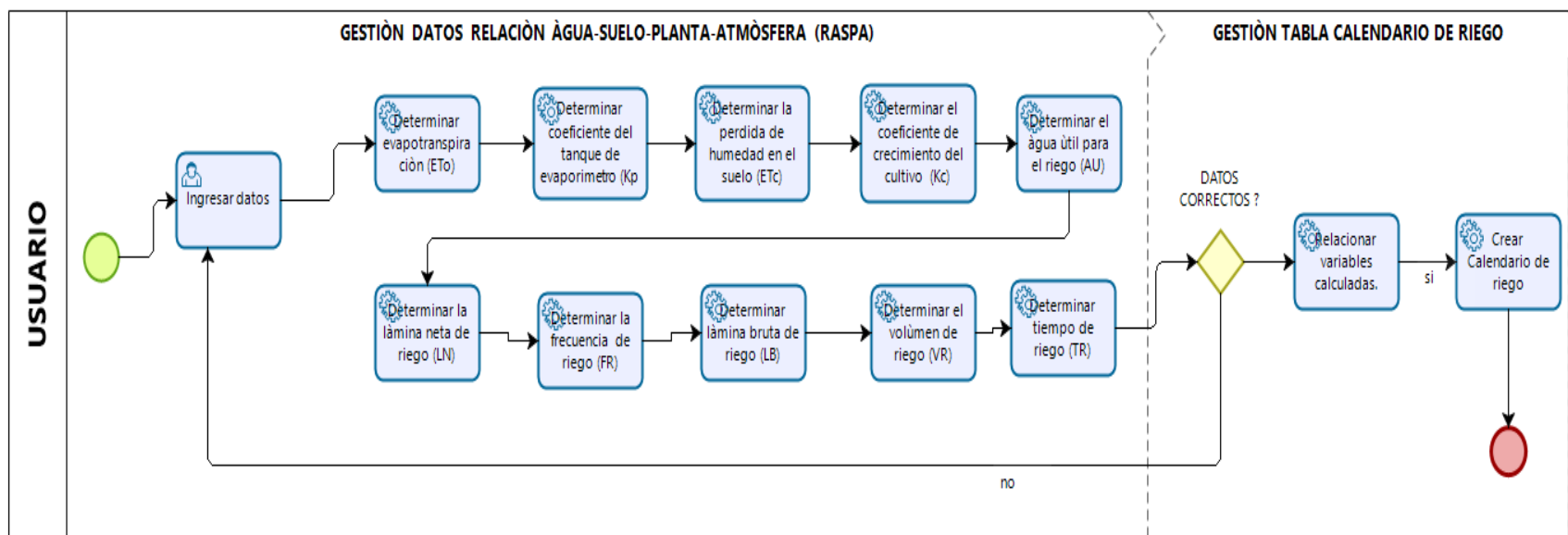


Gráfico 2-2: Diagrama de proceso de elaboración del calendario de riego.

Realizado por: Torres, I. 2020

Como se observa con la ayuda de la aplicación web el usuario solo tiene que ingresar los datos necesarios para calcular el calendario de riego, y el sistema automáticamente va calculando cada variable, finalmente uniéndolas en un reporte a modo de tabla el cual es el calendario de riego.

En la siguiente tabla se enlista las actividades que realizan el usuario y las que realizan la aplicación web.

En la tabla 2-2: se describe el paso a paso del proceso de la elaboración del calendario de riego automático:

Tabla 2-2: Descripción de los pasos de elaboración del calendario de riego automático.

Nº	Actividades	Descripción	Responsable	Resultado
1	Ingresar datos	El usuario ingresa todos los datos necesarios para el cálculo del calendario de riego	Investigador	Ingresar los datos RASPA
2	Determinar Evapotranspiración (ETo)	El investigador debe de calcular matemáticamente la ETo	Aplicación web	Establecer el valor de la (ETo)
3	Determinar el coeficiente del tanque evaporímetro (KP)	El investigador debe de calcular matemáticamente el KP	Aplicación web	Establecer el valor del (KP)
4	Determinar la pérdida de humedad en el suelo (ETc)	El investigador debe de calcular matemáticamente el (ETc)	Aplicación web	Establecer el valor de la (ETc)
5	Determinar el coeficiente de crecimiento del cultivo (Kc)	El investigador debe de calcular matemáticamente el (Kc)	Aplicación web	Establecer el valor del (Kc)
6	Determinar el agua útil para el riego (AU)	El investigador debe de calcular matemáticamente la (AU)	Aplicación web	Establecer el valor de la (AU)
7	Determinar la lámina neta de riego (LN)	El investigador debe de calcular matemáticamente la (LN)	Aplicación web	Establecer el valor de la (LN)
8	Determinar la frecuencia de riego (FR)	El investigador debe de calcular matemáticamente la (FR)	Aplicación web	Establecer el valor de la (FR)
9	Determinar la lámina bruta de riego (LB)	El investigador debe de calcular matemáticamente la (LB)	Aplicación web	Establecer el valor de la (LB)
10	Determinar el volumen de riego (VR)	El investigador debe de calcular matemáticamente el (VR)	Aplicación web	Establecer el valor del (VR)
11	Determinar el tiempo de riego (TR)	El investigador debe de calcular matemáticamente la (TR)	Aplicación web	Establecer el valor del (TR)
12	Relacionar variables calculadas.	Unir las variables calculadas conjuntamente con el tiempo de desarrollo del cultivo	Aplicación web	Calendario de riego

Realizado por: Torres, I. 2020

Como se observa de las 12 actividades solo 1 realiza el usuario mientras que las otras 11 realiza el sistema, reduciendo el trabajo y esfuerzo de una manera considerable.

Una vez planteada la propuesta del funcionamiento de la aplicación web se pasa al desarrollo de la misma.

2.3. Desarrollo del sistema informático para la automatización del proceso de elaboración del calendario de riego.

La primera fase de la metodología SCRUM es la de Inicio la cual se detalla a continuación:

2.3.1. Fase de inicio

En la fase de inicio se realizaron varias reuniones con las personas directamente involucradas en el desarrollo del calendario de riego, definiendo los roles que cada uno de ellos va a cumplir, además de hacer un análisis de los riesgos que podrían afectar el desarrollo del mismo, y se ha hecho la recolección de requisitos con los que debe de cumplir el producto final para satisfacer las necesidades del cliente.

2.3.2. Personas Involucradas

La metodología ágil SCRUM define diferentes roles a las personas involucradas en el desarrollo del proyecto, a continuación se define las personas conjuntamente con sus respectivos roles:

Tabla 3-2: Roles y personas involucradas en el proyecto.

Persona	Rol	Institución / Facultad
Dr. Juan León	Project Owner	ESPOCH - FRN
Ing. Alejandra Oñate	Scrum Master	ESPOCH - EIS
Ing. Eduardo Villa	Team Member	ESPOCH - EIS
Dr. Omar Gomez	Project Management Office	ESPOCH - EIS
Isaac Torres	Team Member	ESPOCH - EIS

Fuente: (Romero, 2017)

Realizado por: Torres, I. 2020

2.3.3. Tipos de usuarios y sus roles

El cliente ha solicitado que la aplicación web sea de libre acceso y que todos los datos se ingresen a un principio del desarrollo de la aplicación, es decir que sea como una especie de calculadora en línea que no tenga necesidad de autenticación.

Por esta razón se ha creado un solo tipo de usuario para la aplicación web que se detalla a continuación en la tabla 4-2.

Tabla 4-2: Descripción de las actividades que puede realizar un usuario.

Tipo de usuario	Descripción	Rol
Usuario	El usuario puede visualizar los cultivos del sistema, los datos meteorológicos, las constantes matemáticas, generar un calendario de riego, y sacar un reporte en formato pdf del mismo.	Genera un calendario de riego con los datos previamente ingresados en el sistema.

Fuente: (Romero, 2017)

Realizado por: Torres, I. 2020

2.4. Requerimientos solicitados del usuario para el sistema

A continuación se enlista los requerimientos con los que debe de cumplir la aplicación web:

- Ingresar en la base de datos los valores de las constantes matemáticas.
- Ingresar en la base de datos los cultivos estudiados del CER.
- Ingresar los valores del coeficiente del cultivo determinado por la FAO.
- Ingresar en la base de datos los tipos de suelo existentes.
- Ingresar en la base de datos los métodos de cálculo para el K_c .
- Ingresar en la base de datos los tipos de superficie del tanque tipo A.
- Calcular la evapotranspiración E_{To} .
- Calcular coeficiente de cultivo K_c .
- Calcular evapotranspiración del cultivo E_{Tc} .
- Calcular coeficiente de varianza K_p .
- Calcular constante de tiempo de calendario C.
- Calcular lámina neta de riego L_n .
- Calcular la cantidad de agua útil A_u .
- Calcular lamina bruta de riego L_b .
- Calcular volumen de agua de riego V_r .
- Calcular tiempo de riego T_r .
- Visualizar las constantes ingresadas en el sistema
- Visualizar los datos meteorológicos de referencia
- Visualizar los cultivos ingresados en el sistema
- Listar todas las variables (calendario de riego)
- Generar calendario en formato pdf.

Luego de definir los requerimientos se pasa a crear el artefacto Product Backlog en el cual se han enlistado los requerimientos de manera jerárquica. Obedeciendo las prioridades de cada funcionalidad.

2.5. Product Backlog

Dentro de la SCRUM tenemos un artefacto denominado Product backlog que es la lista de las tareas, que permitirán cumplir con los requerimientos acordados con el Dr. Juan León director del Centro Experimental de Riego, estos requerimientos se los ha enlistado priorizando las actividades con una escala cualitativa de alta, media y baja, según se haya considerado en el desarrollo del sistema.

Los requerimientos se los va a denominar como Historias Técnicas (HT) y de usuario (HU). Para realizar la estimación de cada tarea se utiliza el método de la talla de la camiseta o T-shirt. Las tallas o estimaciones del método son S, M, L y XL, como se lo presenta en la Tabla 5-2, a continuación:

Tabla 5-2: Descripción de técnica de estimación T-Shirt.

Talla	Puntos estimados	Horas de trabajo
XS	4	4
S	8	8
M	16	16
L	20	20
XL	40	40

Fuente: (Romero, 2017)

Realizado por: Torres, I. 2020

Un punto estimado es igual a una hora de trabajo; un día de trabajo es de 4 horas realizado por una sola persona, por consiguiente, una semana de trabajo (5 días) equivale a 20 puntos estimados.

En la siguiente tabla se describe las tareas a desarrollar del sistema informático para la automatización del proceso de la elaboración del calendario de riego para el CER de la ESPOCH, en la que se especifica:

ID: Es la columna del identificador para la funcionalidad.

Tareas: La es la columna de la funcionalidad.

Estimación: La estimación está dada por tiempo-hombre, el tiempo está dado por horas acotando que un día laborable es de 4 horas y el trabajo es realizado por una persona.

HT: Historias Técnicas.

HU: Historias de Usuario.

Tabla 6-2: Lista de actividades priorizadas. (Product Backlog)

ID	TAREA	ESTIMACIÓN	TALLA
PRIORIDAD ALTA			
HT-01	Investigar, en que consiste el proceso de la elaboración del calendario de riego.	16	M
HT-03	Definir los requerimientos funcionales y no funcionales del proyecto a realizarse.	8	S
HT-05	Analizar y diseñar de manera preliminar de la base de datos.	16	M
HT-07	Desarrollar e implantar de la base de datos en un Sistema Gestor de Bases de Datos (PostgreSQL 9.4).	16	M

HT-08	Desarrollar la interfaz del formulario de ingreso.	20	XS
HT-09	Desarrollar el formato de la tabla del calendario de riego.	40	XL
HU-01	Ingresar en la base de datos los valores de las constantes.	40	XL
HU-02	Ingresar en el sistema la información de los cultivos.	20	L
HU-03	Ingresar los valores del coeficiente del cultivo determinado por la FAO.	40	XL
HU-04	Ingresar en el sistema los tipos de suelo existentes.	20	L
HU-07	Ingresar en el sistema los métodos de cálculo para el Kc.	40	XL
HU-08	Ingresar en el sistema los tipos de superficie del tanque tipo A.	20	XL
PRIORIDAD MEDIA			
HU-09	Calcular la evapotranspiración ETo	8	S
HU-10	Calcular coeficiente de cultivo Kc	8	S
HU-11	Calcular evapotranspiración del cultivo ETc	16	M
HU-12	Calcular coeficiente de varianza Kp.	40	XL
HU-13	Calcular constante de tiempo de calendario C	16	X
HU-14	Calcular lámina neta de riego Ln	40	XL
HU-15	Calcular la cantidad de agua útil Au	20	L
HU-16	Calcular lamina bruta de riego Lb	20	L
HU-17	Calcular volumen de agua de riego Vr	20	L
HU-18	Calcular tiempo de riego Tr	20	L
HU-19	Listar todas las variables (calendario de riego)	20	L
PRIORIDAD BAJA			
HT-04	Definir el estándar de programación.	4	XS
HU-20	Generar calendario en formato pdf.	4	XS
HT-13	Documentar cada uno de los procesos desarrollados en las prácticas mediante un manual técnico.	4	XS
HU-21	Capacitar a los usuarios en el manejo del sistema informático para su adecuado funcionamiento.	4	XS

Realizado por: (Torres, 2020)

En total se obtuvieron 8 historias técnicas y 19 historias de usuario, de las cuales 7 tienen una talla L, 4 tienen una talla M, 3 tienen una talla S y 5 son de talla XS.

2.6. Fase de planificación

Dentro de los artefactos de SCRUM está el Sprint Backlog el cual permite crear un cronograma de las actividades que se desarrollaran en los diferentes sprint's con el objetivo de ir presentado entregables que denoten el avance del proyecto. Para programar estas actividades se ha tenido en

cuenta la tabla anteriormente detallada que prioriza las tareas, es decir según como según es la necesidad de realizar la funcionalidad se lo ha agregado al primer sprint.

2.6.1. *Sprint Backlog*

Sprint o iteración es la unidad básica de trabajo para un equipo dentro de la metodología Scrum, esta es la característica principal que marca la diferencia entre Scrum y otros modelos para el desarrollo ágil.

Para el desarrollo de la aplicación web, se establecieron 7 sprints que se detallan en la Tabla 6-2 a continuación presentada:

La ejecución de los Sprints se realizó con un total de 340 puntos de esfuerzo.

Tabla 7-2: Sprint Backlog.

SPRINT	ID TAREA	FECHA INICIO	FECHA FIN	RESPONSABLE	ESFUERZO
Sprint 1 Total Esfuerzo 56	HT-01	09-09-2019	18-09-2019	Isaac Torres	32
	HT-03	19-09-2019	20-09-2019	Isaac Torres	8
	HT-05	23-09-2019	25-09-2019	Isaac Torres	8
	HT-13	26-09-2019	26-09-2019	Isaac Torres	4
	HT-04	27-09-2019	27-09-2019	Isaac Torres	4
Sprint 2 Total Esfuerzo 60	HT-07	30-09-2019	04-10-2019	Isaac Torres	20
	HT-08	07-10-2019	11-10-2019	Isaac Torres	20
	HT-09	14-10-2019	17-09-2019	Isaac Torres	16
	HT-13	18-09-2019	18-09-2019	Isaac Torres	4
Sprint 3 Total Esfuerzo 40	HU-01	21-10-2019	25-10-2019	Isaac Torres	20
	HU-02	28-10-2019	31-10-2019	Isaac Torres	16

	HT-13	01-11-2019	01-11-2019	Isaac Torres	4
Sprint 4 Total Esfuerzo 40	HU-03	04-11-2019	08-11-2019	Isaac Torres	20
	HU-04	11-11-2019	14-11-2019	Isaac Torres	16
	HT-13	15-11-2019	15-11-2019	Isaac Torres	4
Sprint 5 Total Esfuerzo 40	HU-07	18-06-2019	22-06-2019	Isaac Torres	20
	HU-08	25-11-2019	28-11-2019	Isaac Torres	16
	HT-13	29-11-2019	29-11-2019	Isaac Torres	4
Sprint 6 Total Esfuerzo 40	HU-09	02-12-2019	02-12-2019	Isaac Torres	4
	HU-10	03-12-2019	03-12-2019	Isaac Torres	4
	HU-11	04-12-2019	04-12-2019	Isaac Torres	4
	HU-12	05-12-2019	05-12-2019	Isaac Torres	4
	HU-13	06-12-2019	06-12-2019	Isaac Torres	4
	HU-14	09-12-2019	10-12-2019	Isaac Torres	8
	HU-15	11-12-2019	11-11-2019	Isaac Torres	4
	HU-16	12-12-2019	12-11-2019	Isaac Torres	4
Sprint 7 Total Esfuerzo 24	HT-13	13-12-2019	13-12-2019	Isaac Torres	4
	HU-17	16-12-2019	16-12-2019	Isaac Torres	4
	HU-18	16-12-2019	16-12-2019	Isaac Torres	4

	HU-19	17-12-2019	17-12-2019	Isaac Torres	4
	HU-20	18-07-2019	18-07-2019	Isaac Torres	4
	HU-21	19-07-2019	19-07-2019	Isaac Torres	4
	HT-13	20-12-2019	20-12-2019	Isaac Torres	4

Realizado por: (Torres, 2020)

2.7. Fase de desarrollo

SCRUM en esta fase describe las tareas y actividades conforme se han ido desarrollando, para completar la creación del proyecto. Lo primero en realizarse fue las historias técnicas mismas que se han contemplado los sprints de inicio, para luego continuar con las historias de usuario de acuerdo al orden que se estableció en el product backlog.

2.8. Arquitectura del sistema

El desarrollo del sistema web para la automatización de proceso de la elaboración del calendario de riego, está basado en la arquitectura cliente/servidor.

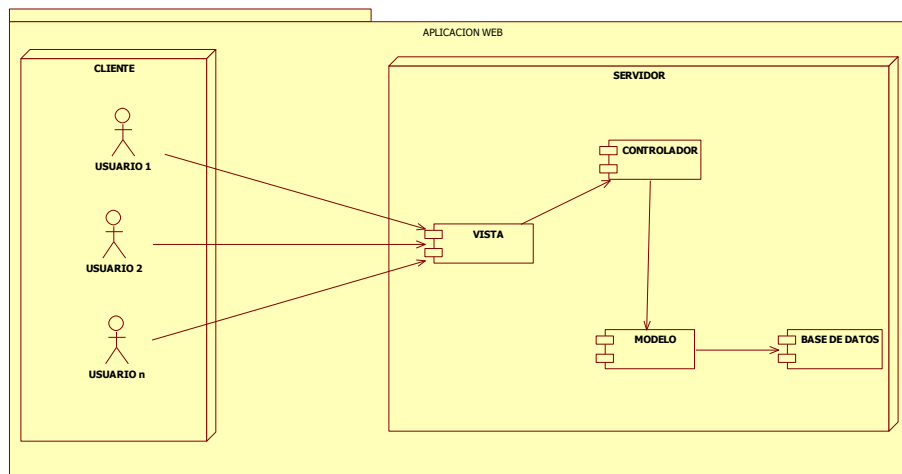


Gráfico 3-2: Arquitectura del sistema

Realizado por: (Torres, 2020)

2.8.1. Capa de cliente

Dentro de esta capa se han desarrollado los archivos en los cuales está construida la interfaz que el usuario va a visualizar en pantalla cuando esté trabajando en el sistema. Estas interfaces se las ha realizado con el framework que permite crear interfaces amigables e interactivas con el usuario,

BOOSTRAP en su versión 4.0. En esta capa se ha utilizado el patrón de diseño modelo vista controlador (MVC), en el modelo está el archivo modelo.jsp que interactúa con el acceso a datos, en la vista está el archivo index.jsp, menú.jsp, y html.jsp y en el controlador está el archivo controlador.jsp que recibe las solicitudes de la vista y se las envía al modelo. Todo esto permite separar de manera ordenada los archivos que se comunican entre sí, dando como resultado que el trabajo de dar mantenimiento o a futuro aumentar funcionalidades sea una tarea sencilla.

A continuación se ilustra el patrón MVC con sus componentes:

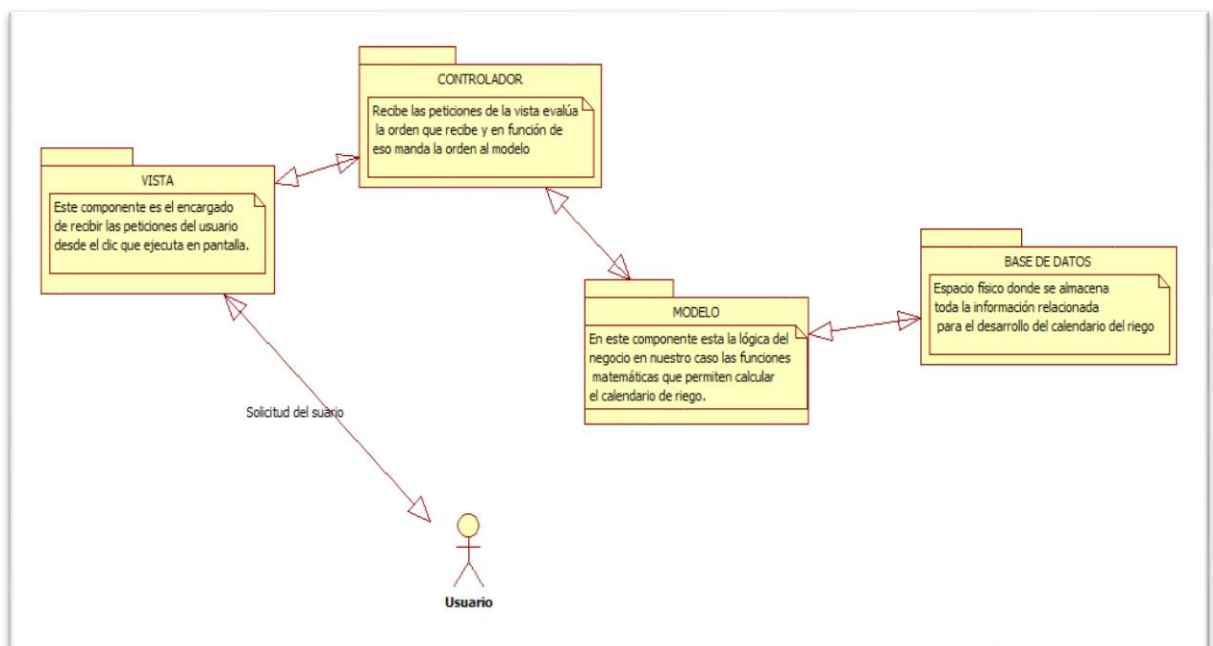


Gráfico 4-2: Modelo vista controlador con sus componentes.

Realizado por: Torres, I. 2020

2.8.2. Capa de Servidor

En esta capa se ejecutan las peticiones que recibe de la capa de presentación, aquí se establecen lo que se conoce como las reglas del negocio, el cómo va a desempeñarse el sistema. Aquí es donde está el controlador el cual de acuerdo a la tarea realiza un procedimiento y devuelve su resultado.

En esta capa están las paginas server de java archivos JSP, que son los encargados de realizar ejecutar la consultas del lado del servidor.

La información necesaria como valores de constantes, métodos, cálculos matemáticos, etc. se encuentran alojados en la base de datos, desde los archivos JSP se puede acceder a ellos para realizar las consultas. Toda esta información antes nombrada está administrada por el sistema

gestor de base de datos PostgreSQL en su versión 9.4, en esta capa se recibe las peticiones de la capa de lógica de negocios como guardar, modificar o eliminar las diferentes entidades.

2.9. Estándar de codificación

Con el propósito de tener un mismo orden de codificación de las funciones, métodos, procedimientos almacenados, de las interfaces, variables, objetos, etc. Se ha obedecido a un estándar de programación para que al momento de integrar funcionalidades, dar mantenimiento o agregar código se lo pueda hacer sin problemas de uniformidad o legibilidad. Para mayor detalle del Estándar de programación dirigirse al **ANEXO A**

2.10. Estándar de interfaces

El objetivo de tener un estándar en las interfaces es brindar al usuario una experiencia agradable y que sea amigable la iteración con el sistema mientras el usuario lo usa. Con el cliente del sistema se ha llegado al acuerdo de que el sistema tendrá las siguientes interfaces de ingreso de datos para el cálculo de calendario de riego, los mismo colores y formas se las reutilizará en los otros formularios y componentes gráficos que se desarrollen el sistema. A continuación se presenta el modelo y colores de las interfaces a utilizar en el sistema.

Figura 1-2: Estándar de colores de interfaz de usuario formulario ingreso datos.

Realizado por: Torres, I. 2020

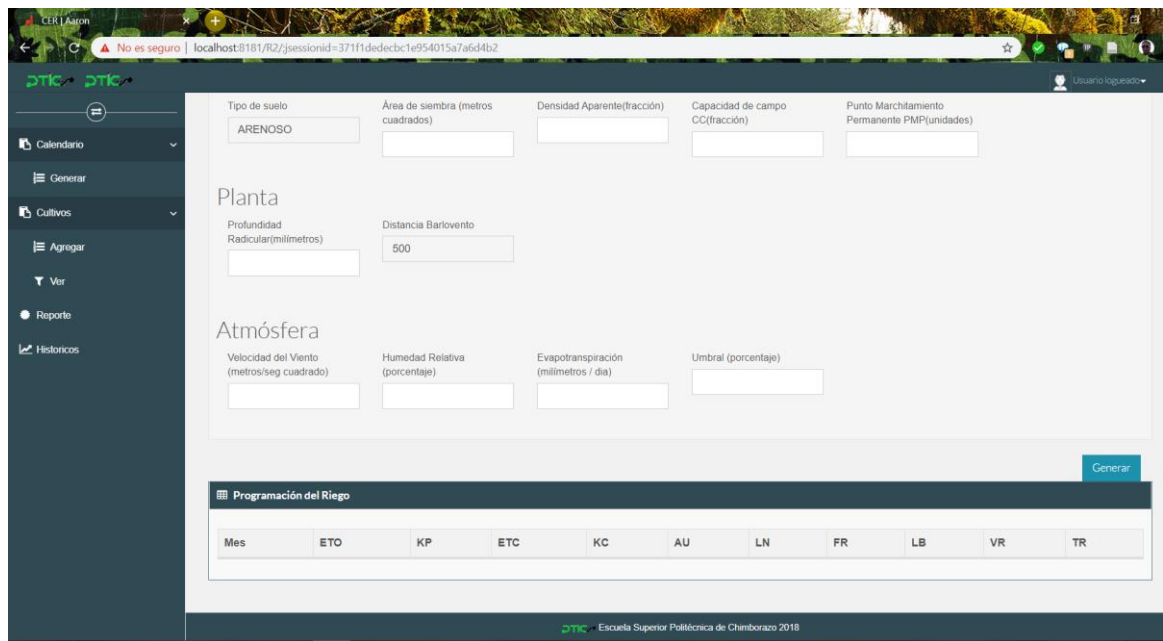


Figura 2-2: Estándar de colores de interfaz de usuario reporte.

Realizado por: Torres, I. 2020

2.11. Diseño de la base de datos

La Base de Datos para el Sistema Informático para la automatización del proceso de la elaboración del calendario de riego para el CER de la ESPOCH está elaborado en PostgreSQL en su versión 9.4 y cuenta con 12 tablas para realizar las acciones de ingreso, modificación, visualización, búsquedas y procesos de cálculo matemático.

En la siguiente ilustración se muestra el esquema de la base de datos, se observan todas las tablas ocupadas en el sistema, y los campos que cada tabla contiene con sus respectivos tipos de dato (modelo lógico). PgAdmin en su versión 4.2 es la herramienta que se utilizó para diseñar la base de datos.

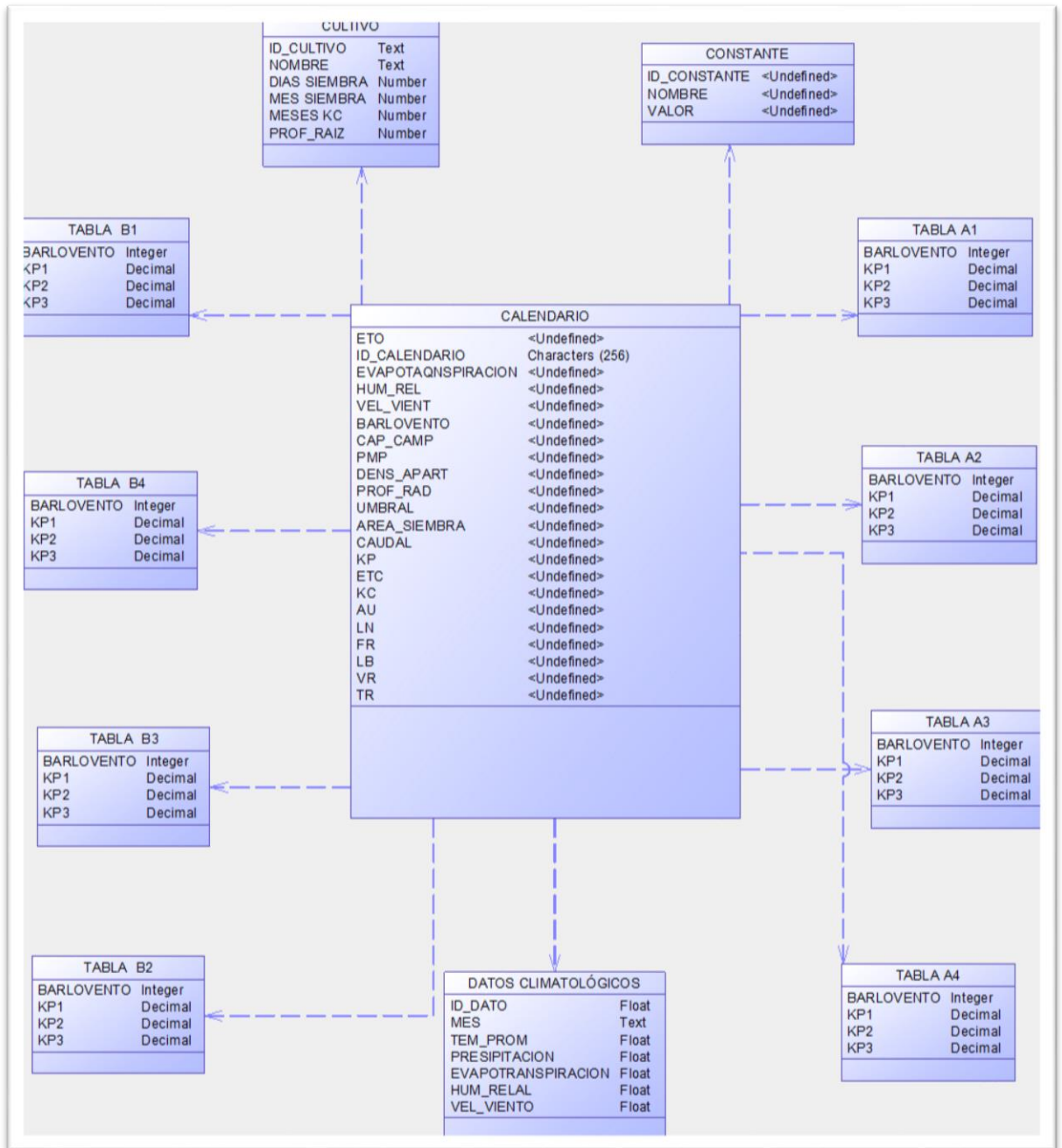


Gráfico 5-2: Base de datos del sistema.

Realizado por: Torres, I. 2020

2.11.1. Reuniones y entregables

En la metodología SCRUM se debe de tener reuniones para mostrar al cliente los avances en el desarrollo de la aplicación web, estas reuniones han sido cada dos semanas con el Dr. Juan León director de CER. En cada entregable se ha presentado de 3 a 4 funcionalidades.

2.11.2. Documentación

Todas las tareas ya sean tareas de usuario o técnicas que se ha realizado han sido documentadas para evidenciar el trabajo realizado y para una guía del proceso de desarrollo de la aplicación web.

Toda la documentación de las historias de usuario y técnicas conjuntamente con sus tareas de ingeniería y pruebas de aceptación se encuentra en el ANEXO C correspondiente al manual técnico.

2.11.3. Historias de usuario

Empezando el 16 de septiembre del 2019 hasta el 21 de febrero del 2010, se fueron creando una a una las historias de usuario necesarias para lograr terminar el proyecto, tal como se había planificado en Sprint Backlog, cada una de estas historias de usuario se las fue documentando en su respectiva tabla descriptiva.

La evidencia de todo esto se la refleja en el ANEXO E. Todas las historias de usuario tienen de dos a tres tareas de ingeniería dependiendo la complejidad de la misma, tanto la historia de usuario como la tarea de ingeniería tiene su prueba de aceptación validada.

Fueron 27 historias de usuario, 45 tareas de ingeniería y 52 pruebas de aceptación, el número de entidades en la base de datos fueron 12, procedimientos almacenados 86, interfaces para iteración del usuario fueron 8, contabilizando las líneas de código que se escribieron dieron un total de 25634.

Cada Historia de usuario está descrita por:

ID: Es el identificador de la Historia de Usuario está descrita para las Historias técnicas el prefijo.

HT-Historias Técnicas

HU- Historia de Usuario.

Nombre: Es el nombre descriptivo de la Historia de Usuario.

Descripción: Es una descripción resumida de la Historia de Usuario.

Responsable: Muestra el nombre de la persona encargada de la Historia de Usuario.

Esfuerzo: Es la evaluación del coste de implementación en unidad de desarrollo. Esta unidad representa el tiempo teórico (desarrollo/hombre) estimada.

Historias Técnicas: Estas historias mantienen la misma estructura que una historia de Usuario a diferencia que la primera da solución a las necesidades que tiene el desarrollador mientras que la segunda corresponde a dar solución a un requerimiento funcional.

Pruebas de Aceptación: Son las características con las que debe cumplir una funcionalidad para ser aceptada. Cada historia de Usuario tiene una o más pruebas de aceptación. Las pruebas de aceptación fueron evaluadas por el cliente al finalizar con el desarrollo de la historia, posterior a

la evaluación se estableció su aceptación o a su vez su modificación para que cumpla con las expectativas del cliente.

Tareas de ingeniería: Son cada una de las tareas que se realizaron para cumplir con el requerimiento de una historia de usuario especificando el esfuerzo que se empleó para cumplir con la misma.

Al terminar las historias de usuarios que responden a los requerimientos funcionales del sistema web para la automatización del proceso de elaboración del calendario de riego, terminado la fase de desarrollo, todo en cuanto se ha hecho se lo ha documentado en el manual técnico del sistema mismo que se encuentra en el ANEXO G.

EL código fuente del sistema se lo ha almacenado en la nube en la carpeta de acceso libre TESIS_ISAAC.TORRES-AARON versión 1.0, en link de esta carpeta es:”
<https://github.com/isaactores/aaron>”

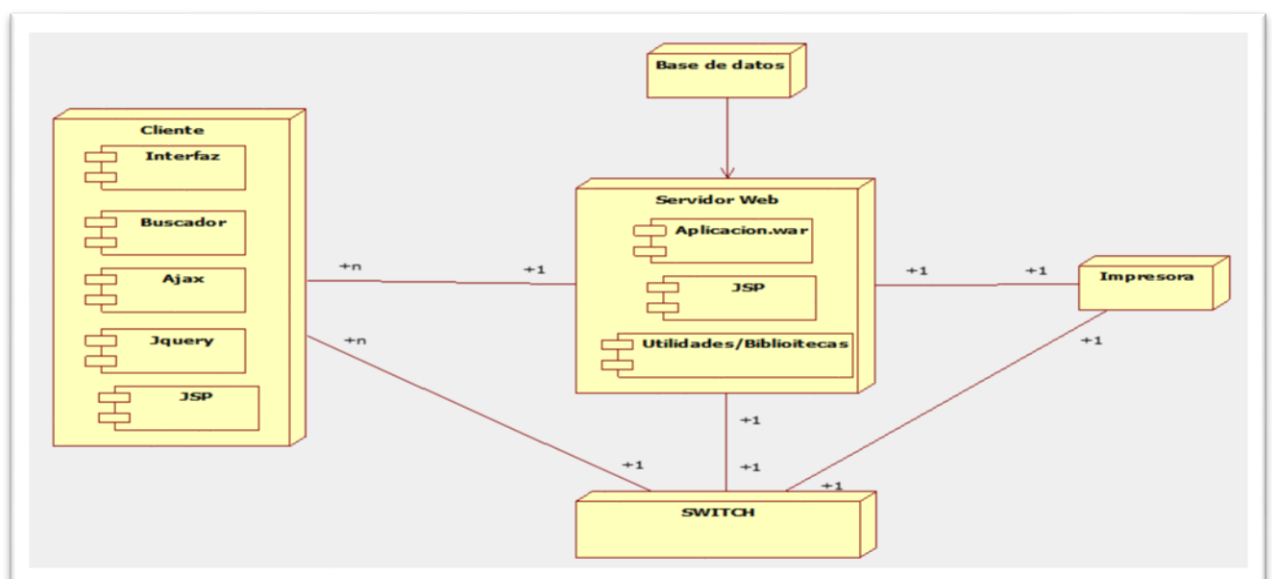
2.11.4. Manual de usuario

En este manual se detalla cada uno de los componentes de las interfaces, para que el usuario tenga una guía de cómo utilizar la aplicación de manera correcta. En dicho manual se detalla los íconos de manera gráfica conjuntamente con la acción que realiza, para mayor detalle del manual de usuario dirigirse al ANEXO E.

2.11.5. Diagrama de despliegue

A continuación en el **gráfico 16-2** se muestra el diagrama de despliegue en el que se muestra la forma en la que está instalada la aplicación web a nivel de hardware.

Gráfico 6-2: Diagrama de despliegue.



Realizado por: (Torres, 2020)

2.12. Fase de cierre

En esta fase se pone en evidencia el artefacto de SCRUM denominado Burndown Chart en cual podremos observar el cómo fue el ritmo de avance de la ejecución del proyecto.

2.12.1. Gestión del proyecto

A continuación en la fase de cierre se describe el comportamiento de los sprint`s de desarrollo de la aplicación web.

2.12.2. Burndown Chart

A continuación en el gráfico podemos observar que la línea de color azul evidencia los puntos estimados y la línea de color naranja describe los puntos reales en ambos casos en función de los sprints.

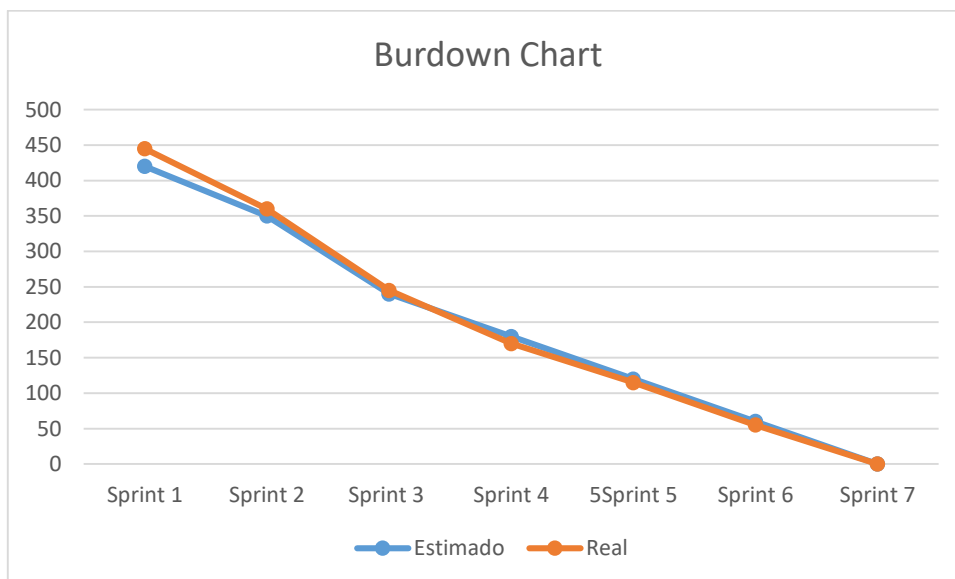


Gráfico 7-2: Puntos estimados vs reales de los Sprints.

Realizado por: Torres, I. 2020

Existen unos leves desfases entre los puntos estimados y los puntos reales de los sprints 1, 2, 3 y 4, estas diferencias se dieron por la razón de que existieron cambios de última hora en las funcionalidades lo que llevaba a ocupar más tiempo en el desarrollo de la misma. También se puede evidenciar que en algunos sprints como el número 5 y 6 el tiempo de desarrollo fue menor debido a que ya se estaba familiarizado con las herramientas de desarrollo.

2.13. Método utilizado para evaluar la eficiencia del desempeño de la aplicación web

A continuación se evalúa los tiempos de respuesta en elaborar un calendario de riego, haciendo una comparativa entre el tiempo invertido en hacer este proceso de manera manual y de forma automatizada, además de medir los recursos a nivel de hardware (cantidad de memoria RAM y porcentaje de utilización del procesador) que se emplean en realizar esta tarea, para lograr lo antes dicho se define la población y la muestra.

Las técnicas que se utilizan para describir los valores de las muestras tomadas es la estadística descriptiva mientras que para establecer una conclusión en función de los resultados se utiliza la estadística inferencial.

2.13.1. Análisis comparativo del proceso de manera manual vs automático

Se definen tres grupos de usuarios para el análisis de la eficiencia al utilizar la aplicación web, estos grupos se los ha dividido por nivel de experticia que tienen en la elaboración del calendario de riego. Clasificándolos como se denota a continuación:

- Nivel Básico
- Nivel Intermedio
- Nivel Avanzado

A estos tres grupos han realizado el calendario de riego de manera manual tomando el tiempo que demoran en su elaboración con la ayuda de un cronómetro, y luego ha realizado el mismo calendario con la aplicación web.

Con estas mediciones se puede hacer el comparativo del proceso manual vs el proceso automático.

2.13.2. Población

La población para evaluar la calidad desde la perspectiva de la eficiencia es de 168 usuarios, los cuales se tomaron del 6to, 7mo y 8vo semestre de la escuela de Agronomía de la facultad de recursos naturales de la ESPOCH, adicional a ellos docentes e investigadores del CER.

Teniendo en cuenta que los de 6to y 7mo semestre son del nivel básico, del 8vo semestre son de nivel intermedio y los docentes e investigadores pertenecen al nivel avanzado.

Para medir la utilización de recursos a nivel de hardware, la población es de 38 computadoras pertenecientes al laboratorio de cómputo del CER.

2.13.3. Muestra

Del total de la población se ha elegido 22 usuarios para hacer el estudio del proceso en el nivel avanzado, 25 usuarios para hacer el estudio del nivel intermedio y 32 usuarios para el estudio del nivel básico.

Para la medir la utilización de recursos se ha tomado una de las 38 computadoras del laboratorio de cómputo con las siguientes características:

Tabla 8-2: Características de computadora evaluada.

Característica	Descripción
HARDWARE	
Procesador	Intel Pentium E5700 2MB caché a 3.00GHz
Memoria RAM	4GB DDR1 de 266MTps.
Almacenamiento	Disco duro de 320GB sigate.
SOFTWARE	
Sistema Operativo	Windows 10 de 32 bits en versión mínima.
Navegadores	Opera, Chrome, internet Explorer, Edge

Realizado por: Torres, I. 2020

2.13.4. Comportamiento del tiempo

Consiste en medir la cantidad de tiempo utilizado en realizar un calendario de riego, antes y después del desarrollo de la aplicación web para automatizar este proceso.

2.14. Utilización de recursos

Para medir la cantidad de memoria RAM y el porcentaje de utilización del procesador que invierte una computadora en la elaboración del calendario de riego se ha utilizado el medidor de recursos del administrador de tareas del propio sistema operativo.

Tomando las muestras en los diferentes navegadores que tiene la computadora y comparando cual navegador requiere de menor cantidad.

2.14.1. Recolección de datos

Para obtener los datos a analizar se ha evaluado el proceso de manera general desde que empieza el cálculo de las variables, hasta unirlos en una tabla conjuntamente con el tiempo de desarrollo del cultivo, para obtener el calendario de riego.

Para tomar los tiempos invertidos en la elaboración del calendario de riego de manera manual se ha realizado un examen al grupo de estudiantes de los semestres mencionados anteriormente, mientras que los docentes e investigadores lo han hecho en forma individual por separado.

La toma de mediciones de los recursos se lo ha realizado con el administrador de tareas, capturando la pantalla en donde se muestra los valores de memoria y de CPU utilizados por cada navegador, los navegadores evaluados son: Chrome, Edge, Opera, Mozilla Firefox e Internet

Explorer, en cada uno se ha ejecutado la aplicación web de manera individual para tomar valores más exactos, es decir no todos o más de uno al mismo tiempo.

Todos los valores de los datos tomados para el análisis están detallados en el ANEXO D Tiempos de respuesta.

CAPÍTULO III

3. MARCO DE RESULTADOS, DISCUSIÓN Y ANÁLISIS DE RESULTADOS

En este capítulo se realizó el análisis y evaluación de la eficiencia de la aplicación para el cálculo del calendario de riego, utilizando la norma ISO/IEC 25010, dentro de la eficiencia se evaluó las sub-características tiempo de respuesta en realizar el proceso y la utilización de recursos.

Se tomó tres grupos de usuarios para realizar las mediciones, estos grupos se los clasificó de acuerdo a la experticia que tienen para elaborar un calendario de riego, estos son: Básico, intermedio, avanzado.

Los datos se los recolectó desde el 03-01-2020 hasta 19-05-2020, las muestras tomadas de manera manual se las obtuvo por medio de un test, mientras que los datos del proceso automatizado por medio de un cronómetro digital.

Se han aplicado las herramientas de la estadística descriptiva para detallar los valores de las muestras tomadas, luego de eso se aplica la estadística inferencial para determinar si existe o no mejoramiento en la eficiencia del proceso de la elaboración del calendario de riego.

La unidad de medida con la que se ha trabajado para el tiempo de respuesta es en minutos (min), para la cantidad de memoria en Megabytes (MB) y la cantidad de procesador utilizado en porcentaje (%).

3.1. Estadística Descriptiva

A continuación, se realizó el análisis descriptivo de los tiempos de respuesta tomados en los diferentes grupos.

3.2. Eficiencia

Se empezó con el análisis de la sub-característica de la eficiencia:

3.2.1. *Tiempo de respuesta*

3.2.1.1. *Grupo 1: Nivel Básico*

En la tabla 1-3 se detalla los tiempos que emplearon los usuarios del grupo de nivel básico en la elaboración del calendario de manera manual y automática.

Tabla 1-3: Descripción de las muestras tomadas y valores calculados del nivel básico.

	Número de muestras	Media	Desviación Estándar	Mediana	Máximo	Mínimo
Antes	32	214.93	8.27	214	232	200
Después	32	15.33	1.17	14.45	17.3	12.9

Realizado por: Torres, I. 2020

Como esta detallado en la tabla 1-3, se ha evaluado a 32 usuarios, los mal realizar el calendario de riego de manera manual se demoraban un promedio de 214.94 min, mientras que de manera automática obtuvieron un promedio de 15.33 min, notándose una clara disminución de tiempos, la desviación estándar antes de la automatización es de 8.27 y después 1.17, las medianas también evidencian una clara diferencia antes 214 y después 14.45, el valor máximo de demora antes fue de 232 min y después de automatizar fue de 17.3, el valor mínimo de tiempo que se utilizó antes de la automatización fue de 200 min y después se encontró un mínimo de 12.9 min. Como se observa todos los valores antes de automatizar el proceso eran mayores a los tiempos de después de realizar la automatización.

A continuación, en el **gráfico 1-3** se detalla el promedio de los tiempos que se emplean antes y después de la automatización del proceso



Gráfico 1-3: Tiempo empleado antes y después nivel básico (minutos)

Realizado por: Torres, I. 2020

Se observa en el gráfico 1-3 los tiempos promedios utilizados en la elaboración del calendario de riego antes y después de la automatización, siendo muy evidente la diferencia de los valores, antes con 214 min frente a un después con 15.45 min.

3.2.1.2. Grupo 2: Nivel Intermedio

En la tabla 2-3 se detalla los tiempos que emplearon el grupo de usuarios del nivel intermedio en la elaboración del calendario de manera manual y automática.

Tabla 2-3: Descripción de las muestras tomadas y valores calculados del nivel intermedio.

	Número de muestras	Media	Desviación Estándar	Mediana	Máximo	Mínimo
Antes	25	103.04	8.96	102	123	90
Después	25	10.60	1.16	10.7	12.8	8.7

Realizado por: Torres, I. 2020

Como esta detallado en la tabla 2-3, se ha evaluado a 25 muestras, los usuarios al realizar el calendario de riego de manera manual se demoraban un promedio de 103.04 min, mientras que de manera automática obtuvieron un promedio de 10.60 min, notándose una clara disminución de tiempos, la desviación estándar antes de la automatización es de 8.96 y después 1.16, las medianas también evidencian una clara diferencia antes 102 y después 10.7, el valor máximo de demora antes fue de 123 min y después de automatizar fue de 12.8, el valor mínimo de tiempo que se utilizó antes de la automatización fue de 90 min y después se encontró un mínimo de 8.7 min.

A continuación, en el **gráfico 2-3** se detalla el promedio de los tiempos que se emplean antes y después de la automatización del proceso

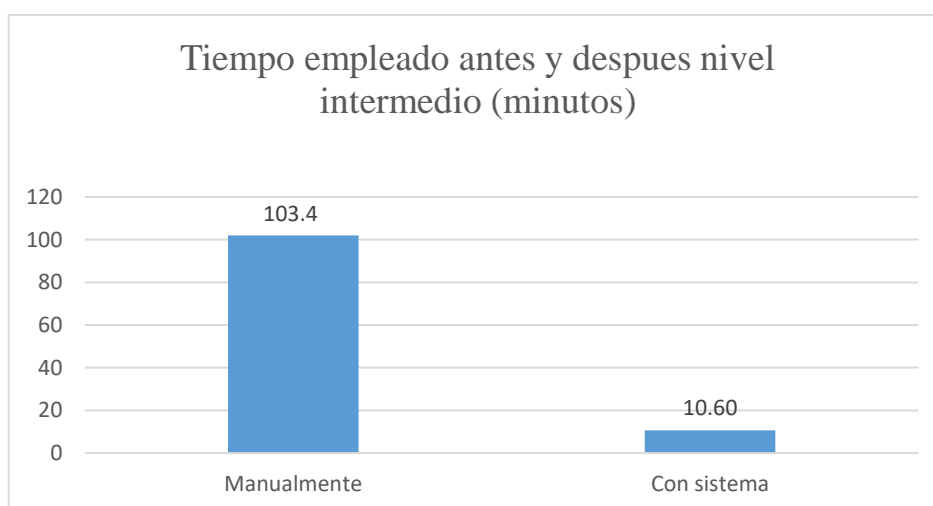


Gráfico 2-3: Tiempo empleado antes y después intermedio (minutos)

Realizado por: Torres, I. 2020

Se observa en el gráfico 2-3 los tiempos promedios utilizados en la elaboración del calendario de riego antes y después de la automatización, siendo muy evidente la diferencia de los valores, antes con 103.4 min frente a un después con 10.60 min.

3.2.1.3. Grupo 3: Nivel Avanzado

En la tabla 3-3 se detalla los tiempos que emplearon el grupo de usuarios del nivel avanzado en la elaboración del calendario de manera manual y automática.

Tabla 3-3: Descripción de las muestras tomadas y valores calculados del nivel avanzado.

	Número de muestras	Media	Desviación Estándar	Mediana	Máximo	Mínimo
Antes	22	48.86	3.77	49	55	42
Después	22	4.88	0.45	4.9	5.7	4.1

Realizado por: Torres, I. 2020

Como esta detallado en la tabla 3-3, se ha evaluado a 22 muestras, los usuarios al realizar el calendario de riego de manera manual se demoraban un promedio de 48.86 min, mientras que de manera automática obtuvieron un promedio de 4.88 min, notándose una clara disminución de tiempos, la desviación estándar antes de la automatización es de 3.77 y después 0.45, las medianas también evidencian una clara diferencia antes 49 y después 4.9, el valor máximo de demora antes fue de 55 min y después de automatizar fue de 5.7, el valor mínimo de tiempo que se utilizó antes de la automatización fue de 42 min y después se encontró un mínimo de 4.1 min. Como se observa todos los valores antes de automatizar el proceso son mayores a los tiempos después de la automatización.

A continuación, en el **gráfico 3-3** se detalla el promedio de los tiempos que se emplean antes y después de la automatización del proceso.

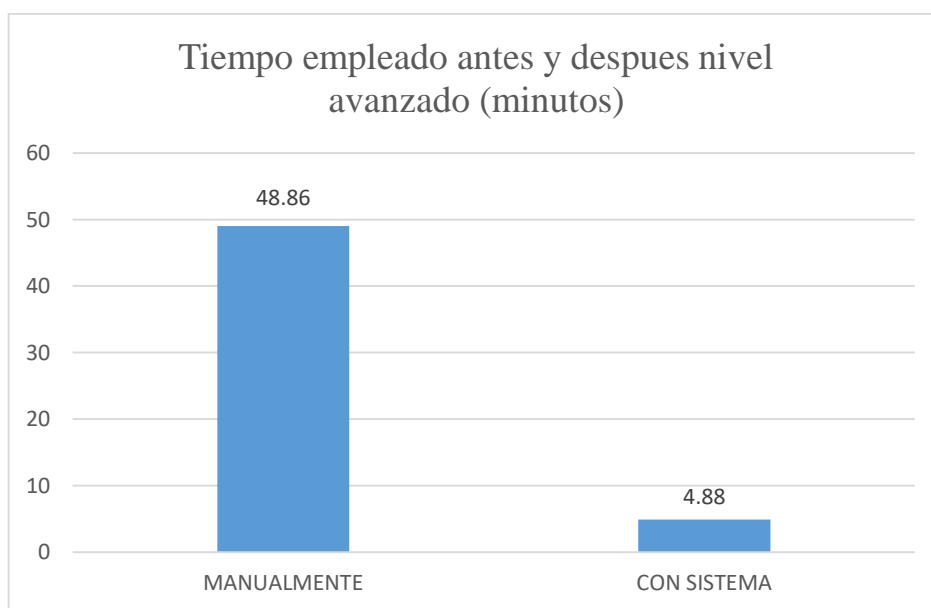


Gráfico 3-3: Tiempo empleado antes y después nivel avanzado (minutos)

Realizado por: (Torres, 2020)

Se observa en el gráfico 3-3 los tiempos promedios utilizados en la elaboración del calendario de riego antes y después de la automatización, siendo muy evidente la diferencia de los valores, antes con 48.86 min frente a un después con 4.88 min.

3.3. Estadística Inferencial

Una vez descritas las muestras tomadas en el antes y después de la automatización del proceso se procede a aplicar la estadística inferencial para determinar el mejoramiento en la eficiencia del proceso de elaboración del calendario de riego.

3.3.1. Grupo 1

Para determinar si los datos de las diferencias de tiempos se ajustan a una distribución normal se aplica el Test de Shapiro Wilk. Este test nos indica si los datos a analizar obedecen una distribución normal.

Antes de aplicar el test se plantean las posibles hipótesis.

Hipótesis Nula (H0) = Los datos obedecen a una distribución normal.

Hipótesis alternativa (H1) = Los datos no obedecen una distribución normal.

En la tabla 4-3 se detalla los resultados del test de Shapiro Wilk, aplicados a los datos de las muestras del nivel básico.

Para establecer si existe o no normalidad en los datos se define la regla:

Si el valor de p-value es mayor que 0.05 existe una distribución normal, caso contrario no existe una distribución normal.

Tabla 4-3: Resultado Saphrio Wilk nivel básico

Shapiro-Wilk Test	
	<i>DIFERENCIA</i>
W-stat	0,97262461
p-value	0,57477848
alpha	0,05
normal	yes
d'Agostino-Pearson	
DA-stat	1,51631723
p-value	0,46852837
alpha	0,05
normal	yes

Realizado por: Torres, I. 2020

En la **tabla 4-3**, se presenta los resultados del test Shapiro Wilk, observando que el valor de p-value es de 0.57477848, por este valor se confirma que los datos correspondientes a la diferencia de tiempos utilizados en el proceso, obedece a una distribución normal.

Se acepta la hipótesis nula del test en favor de la normalidad de los datos. Como se ha determinado que existe una distribución normal en los datos, se procede a aplicar una prueba estadística de normalidad.

La prueba T pareada, se aplica a un conjunto de datos que tiene una distribución normal, a continuación, en el tabla 5-3 se presenta los resultados de la prueba T pareada aplicada al grupo de usuarios de nivel básico, las dos columnas de datos que se evalúan son los tiempos utilizados para desarrollar un calendario de riego antes y después de la automatización, al mismo número de usuarios.

Para realizar el análisis inferencial se debe de establecer el punto de referencia o regla de decisión:

Hipótesis Nula (H_0)= Los tiempos de antes SI son iguales a los tiempos de después de la automatización. ($TA=TD$).

Hipótesis alternativa (H_1)= Los tiempos de antes NO son iguales a los tiempos de después de la automatización. ($TA\neq TD$).

Si $P < 0.05$; Si se rechaza la hipótesis nula

Si > 0.05 ; No se rechaza la hipótesis nula

Tabla 5-3: Prueba T pareada nivel básico

Variables	Tiempo antes	Tiempo después
Media	214,9375	15,330625
Varianza	67,41532258	1,392206048
Observaciones	32	32
Coefficiente de correlación de Pearson	-0,088766076	
Diferencia hipotética de las medias	0	
Grados de libertad	31	
Estadístico t	134,4532796	
P($T \leq t$) una cola	9,35268E-45	
Valor crítico de t (una cola)	1,695518783	
P($T \leq t$) dos colas	1,87054E-44	
Valor crítico de t (dos colas)	2,039513446	

Realizado por: Torres, I. 2020

Se observa que en las 32 muestras la media en antes es de 214.93 mientras que en el tiempo del después es de 15.33, los grados de libertad son 31, con un estadístico t de 134.4532 y un valor P (para dos colas) de 1.87045E-44. Es notorio que la probabilidad que indica el valor P es significativamente baja por lo tanto se rechaza absolutamente la hipótesis nula a favor de la hipótesis alternativa, mientras más pequeño es el valor de P más fuerte es la contra de la hipótesis nula.

La cantidad de tiempo reducido es de 199.59 min con la automatización del proceso. En este caso los tiempos del “después” siempre van a ser significativamente menores de los tiempos que se utilizan “antes” de la automatización. Al representar estos valores en términos de porcentaje estamos hablando de una reducción de tiempo en un 93.2%. Es decir del 100% del tiempo que se utilizaba para elaborar un calendario de riego de manera manual, ahora con la aplicación web se emplea solo el 6.8% de ese tiempo inicial.

3.3.2. Grupo 2

Para determinar si los datos de las diferencias de tiempos se ajustan a una distribución normal se aplica el Test de Shapiro Wilk.

Antes de aplicar el test se plantean las posibles hipótesis.

Hipótesis Nula (H_0) = Los datos obedecen a una distribución normal.

Hipótesis alternativa (H_1) = Los datos no obedecen una distribución normal.

En la tabla 5-3 se detalla los resultados del test de Shapiro Wilk, aplicados a los datos de las muestras del nivel intermedio.

Para establecer si existe o no normalidad en los datos se define la regla:

Si el valor de p-value es mayor que 0.05 existe una distribución normal, caso contrario no existe una distribución normal.

Tabla 6-3: Resultado Saphrio Wilk nivel intermedio.

Shapiro-Wilk Test	
	<i>DIFERENCIA</i>
W-stat	0,967478237
p-value	0,605279726
alpha	0,05
normal	yes
d'Agostino-Pearson	
DA-stat	0,802052546
p-value	0,669632468
alpha	0,05
normal	yes

Realizado por: Torres, I. 2020

En la **tabla 6-3**, se presenta los resultados del test Shapiro Wilk, observando que el valor de p-value es de 0.605279726, por este valor se confirma que los datos correspondientes a la diferencia de tiempos utilizados en el proceso, obedece a una distribución normal.

Se acepta la hipótesis nula del test, en favor de la normalidad de los datos. Como se ha determinado que existe una distribución normal en los datos, se procede a aplicar una prueba de normalidad.

La prueba T pareada, se aplica a un conjunto de datos que tiene una distribución normal, a continuación, en el tabla 6-3 se presenta los resultados de la prueba T pareada aplicada al grupo de usuarios de nivel intermedio, las dos columnas de datos que se evalúan son los tiempos utilizados para desarrollar un calendario de riego antes y después de la automatización, al mismo número de usuarios.

Para realizar el análisis inferencial se debe de establecer el punto de referencia o regla de decisión:

Hipótesis Nula (H_0)= Los tiempos de antes SI son iguales a los tiempos de después de la automatización. ($T_A=TD$).

Hipótesis alternativa (H_1)= Los tiempos de antes NO son iguales a los tiempos de después de la automatización. ($T_A\neq TD$).

Si $P < 0.05$; Si se rechaza la hipótesis nula

Si > 0.05 ; No se rechaza la hipótesis nula

Tabla 7-3: Prueba T pareada nivel intermedio

<i>Variables</i>	<i>Antes</i>	<i>Después</i>
Media	103,04	10,6004
Varianza	80,45666667	1,363754
Observaciones	25	25
Coeficiente de correlación de Pearson	0,037667895	
Diferencia hipotética de las medias	0	
Grados de libertad	24	
Estadístico t	51,34541786	
P($T \leq t$) una cola	2,3516E-26	
Valor crítico de t (una cola)	1,71088208	
P($T \leq t$) dos colas	4,7032E-26	
Valor crítico de t (dos colas)	2,063898562	

Realizado por: Torres, I. 2020

Se observa que en las 25 muestras la media en antes es de 103.04 mientras que en el tiempo del después es de 10.60, los grados de libertad son 24, con un estadístico t de 51,34541786 y un valor P (para dos colas) de 4,7032E-26. Es notorio que la probabilidad que indica el valor P es significativamente baja por lo tanto se rechaza absolutamente la hipótesis nula a favor de la hipótesis alternativa, mientras más pequeño es el valor de P más fuerte es la contra de la hipótesis nula.

La cantidad de tiempo reducido es de 92.44 min con la automatización del proceso. En este caso los tiempos del “después” siempre van a ser significativamente menores de los tiempos que se utilizan “antes” de la automatización. Al representar estos valores en términos de porcentaje estamos hablando de una reducción de tiempo en un 89.71%. Es decir del 100% del tiempo que se utilizaba para elaborar un calendario de riego de manera manual, ahora con la aplicación web se emplea solo el 10.29% de ese tiempo inicial.

3.3.3. Grupo 3

Para determinar si los datos de las diferencias de tiempos se ajustan a una distribución normal se aplica el Test de Shapiro Wilk.

Antes de aplicar el test se plantean las posibles hipótesis.

Hipótesis Nula (H_0) = Los datos obedecen a una distribución normal.

Hipótesis alternativa (H_1) = Los datos no obedecen una distribución normal.

En la tabla 8-3 se detalla los resultados del test de Shapiro Wilk, aplicados a los datos de las muestras del nivel avanzado.

Para establecer si existe o no normalidad en los datos se define la regla:

Si el valor de p-value es mayor que 0.05 existe una distribución normal, caso contrario no existe una distribución normal.

Tabla 8-3: Resultado Saphrio Wilk nivel avanzado

Shapiro-Wilk Test	
	<i>DIFERENCIA</i>
W-stat	0,961285562
p-value	0,515698928
alpha	0,05
normal	yes
d'Agostino-Pearson	
DA-stat	0,351437276
p-value	0,838853972
alpha	0,05
normal	yes

Realizado por: Torres, I. 2020

En la tabla 8-3, se presenta los resultados del test Shapiro Wilk, observando que el valor de p-value es de 0. 515698928, por este valor se confirma que los datos correspondientes a la diferencia de tiempos utilizados en el proceso, obedece a una distribución normal.

Se acepta la hipótesis nula del test en favor de la normalidad de los datos. Como se ha determinado que existe una distribución normal en los datos, se procede a aplicar una prueba de normalidad.

La prueba T pareada, se aplica a un conjunto de datos que tiene una distribución normal, a continuación, en el tabla 9-3 se presenta los resultados de la prueba T pareada aplicada al grupo de usuarios de nivel avanzado, las dos columnas de datos que se evalúan son los tiempos utilizados para desarrollar un calendario de riego antes y después de la automatización, al mismo número de usuarios.

Para realizar el análisis inferencial se debe de establecer el punto de referencia o regla de decisión:

Hipótesis Nula (H_0)= Los tiempos de antes SI son iguales a los tiempos de después de la automatización. ($TA=TD$).

Hipótesis alternativa (H_1)= Los tiempos de antes NO son iguales a los tiempos de después de la automatización. ($TA\neq TD$).

Si $P < 0.05$; Si se rechaza la hipótesis nula

Si > 0.05 ; No se rechaza la hipótesis nula

Tabla 9-3: Prueba T pareada nivel básico

<i>Variables</i>	<i>Antes</i>	<i>Después</i>
Media	49,22727273	4,886363636
Varianza	8,755411255	0,208852814
Observaciones	22	22
Coefficiente de correlación de Pearson	-0,265229662	
Diferencia hipotética de las medias	0	
Grados de libertad	21	
Estadístico t	66,84098053	
P($T \leq t$) una cola	2,96197E-26	
Valor crítico de t (una cola)	1,720742903	
P($T \leq t$) dos colas	5,92395E-26	
Valor crítico de t (dos colas)	2,079613845	

Realizado por: Torres, I. 2020

Se observa que en las 22 muestras la media en antes es de 49.22 mientras que en el tiempo del después es de 4.88, los grados de libertad son 21, con un estadístico t de 66.84 y un valor P (para dos colas) de 5,92395E-26. Es notorio que la probabilidad que indica el valor P es significativamente baja por lo tanto se rechaza absolutamente la hipótesis nula a favor de la hipótesis alternativa, mientras más pequeño es el valor de P más fuerte es la contra de la hipótesis nula.

La cantidad de tiempo reducido es de 44.34 min con la automatización del proceso. En este caso los tiempos del “después” siempre van a ser significativamente menores de los tiempos que se

utilizan “antes” de la automatización. Al representar estos valores en términos de porcentaje estamos hablando de una reducción de tiempo en un 90%. Es decir del 100% del tiempo que se utilizaba para elaborar un calendario de riego de manera manual, ahora con la aplicación web se emplea solo el 10% de ese tiempo inicial.

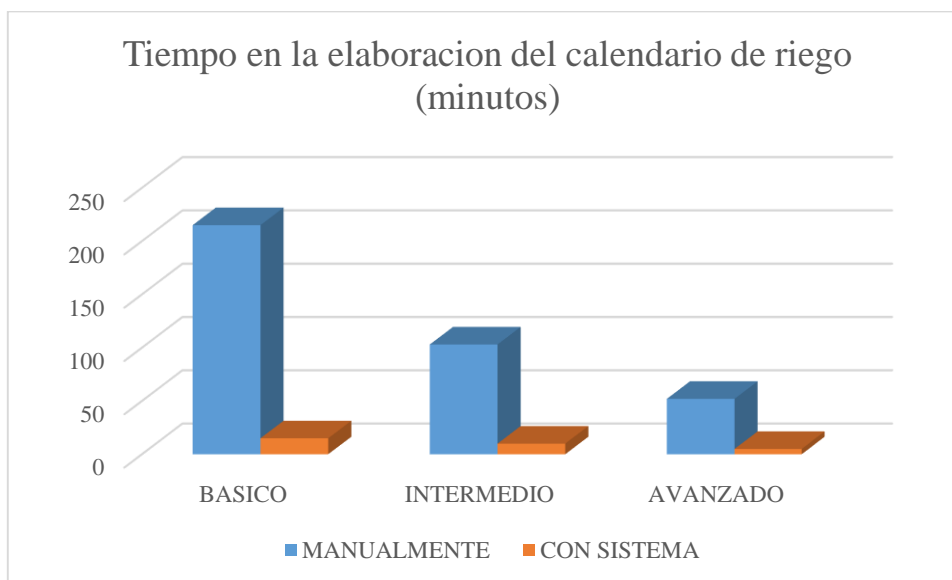


Gráfico 4-3: Tiempos generales invertidos en la elaboración del calendario de riego.

Realizado por: Torres, I. 2020

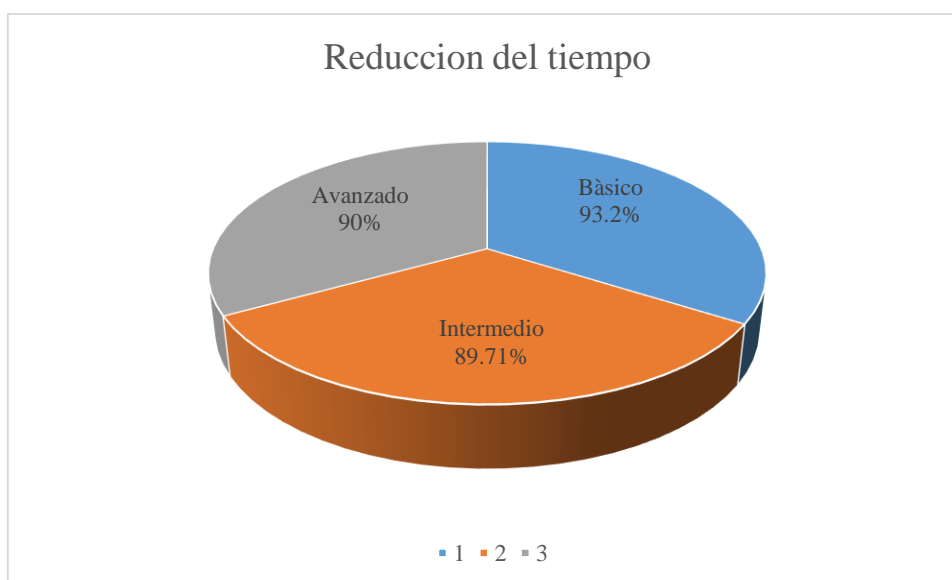


Gráfico 5-3: Reducción de tiempo representado en porcentaje.

Realizado por: Torres, I. 2020

Como se puede observar en el gráfico 5-3 la reducción de tiempos representados en porcentaje tienen un nivel alto muy considerable, similares en los tres tipos de usuarios en los que se ha

clasificado (básico, intermedio y avanzado) notando una leve mayoría en el nivel básico y casi una igualdad en el nivel intermedio y avanzado.

Los resultados mostrados en el gráfico 4-3 se puede observar que el sistema ha logrado reducir considerablemente los tiempos en la elaboración del calendario de riego independientemente del tipo de usuario que esté realizando esta tarea. En el nivel básico en principio de manera manual se demoraba un promedio de 214.93 minutos, con el sistema se ha realizado la misma tarea en 15.33 minutos, logrando una reducción de 199.6 minutos expresando ese resultado en porcentaje se denota como el 93.2%, en el nivel intermedio se ha reducido 92.44 minutos equivalente al 89.71% del valor inicial y por último el nivel avanzado ocupaba 49.22 minutos de manera manual frente a los 4.88 minutos que ocupa con el sistema lo que se pronuncia como una reducción del 90%.

3.4. Recursos Utilizados

Para realizar las mediciones de los recursos empleados en la elaboración del calendario de riego se ha utilizado el administrador de tareas del propio sistema operativo para tomar los diferentes valores de las cantidades de memoria RAM y porcentaje de procesador utilizado, en los diferentes navegadores que tiene instalado la computadora, luego de esto se hace una comparativa entre los diferentes valores de los recursos que utiliza cada navegador.

Todas estas mediciones se las ha tomado de una de las treinta similares computadoras del laboratorio de cómputo del CER, la cual cuenta con las siguientes características:

Tabla 10-3: Descripción hardware y software de computadora evaluada.

Característica	Descripción
HARDWARE	
Procesador	Intel Pentium E5700 2MB caché a 3.00GHz
Memoria RAM	4GB DDR1 de 266MTps.
Almacenamiento	Disco duro de 320GB sigate.
SOFTWARE	
Sistema Operativo	Windows 10 de 32 bits en versión mínima.
Navegadores	Opera, Chrome, internet Explorer, Edge

Realizado por: Torres, I. 2020

La cantidad de memoria RAM viene dada en MB, y la utilización del procesador en unidades de porcentaje, a continuación se muestra los valores de los recursos utilizados en cada uno de los navegadores.

3.4.1. Estadística descriptiva

A continuación en la **tabla 11-3** se describe los valores de cantidad de memoria y porcentaje de procesador utilizados en cada uno de los navegadores.

Tabla 11-3: Descripción hardware y software de computadora evaluada.

Navegador	Memoria RAM	Porcentaje de procesador
Edge	282.32	20.15
Opera	465.35	23.65
Google Chrome	568.90	32.98
Mozilla Firefox	376.91	27.35
Internet Explorer	245.85	19.85

Realizado por: Torres, I. 2020

Se observa que el navegador que menos recursos utiliza es Internet Explorer con 245.85MB de memoria RAM y un 19.85% de procesador, mientras que el más recursos utiliza es el Google Chrome con 568.90 MB y un 32.98% de procesador.

CONCLUSIONES

- Luego de analizar y comprender el proceso para elaborar un calendario de riego utilizando la observación y la revisión de documentación como técnicas de investigación se concluye que: Realizar ésta tarea manualmente requiere un tiempo promedio de 214.93 minutos.
- Para elaborar un calendario de riego existen dos métodos: el método gráfico y el método analítico, siendo este último el más utilizado para desarrollar esta tarea. Este método se basa en fórmulas y cálculos matemáticos, lo cual permite que el resultado obtenido tenga menor grado de incertidumbre en relación al método gráfico, por lo cual se aplicó dentro de la aplicación web para obtener un calendario de riego.
- Al evaluar la eficiencia de la aplicación web basados en la norma ISO 25010 y luego de haber cuantificado el tiempo que se demoran los usuarios en elaborar un calendario de riego de manera manual y después de manera automática, se obtuvieron los siguientes resultados: En el nivel básico se redujo el tiempo empleado en desarrollar esta tarea un 93.2%, en el nivel intermedio un 89.71% y en el nivel avanzado un 90.0%, teniendo un promedio de reducción del tiempo de un 90.97%.
- La aplicación web está limitada a calcular calendarios de riego para los 8 cultivos prioritarios que han sido previamente investigados en el centro experimental del riego siendo estos (papa, maíz, cebolla, arveja, brócoli, fréjol, zanahoria, lechuga).

RECOMENDACIONES

- Al empezar el desarrollo de una aplicación ya sea de tipo web o de escritorio, se recomienda establecer el diagrama de procesos el cual permite comprender de mejor manera el objetivo que pretende alcanzar dicha aplicación, además de establecer el alcance y limitaciones del mismo.
- En el proceso de generación del calendario de riego se recomienda agregar una funcionalidad que permita el cálculo del coeficiente del cultivo (K_c) de manera diaria, esto para que los tiempos de riego sean expresados en intervalos de 60 minutos.
- Se recomienda ejecutar la aplicación web en el navegador Internet Explorer dado que consume 245.85MB de memoria RAM y el 19.85% de capacidad del procesador, cantidades menores en relación a otros navegadores evaluados.
- Implementar en la aplicación web la funcionalidad que permita el ingreso de nuevos cultivos diferentes a los 8 establecidos por defecto, con el fin de ampliar el alcance y la generación de otros calendarios de riego.

GLOSARIO

Clase: Colección encapsulada de datos y operaciones que actúan sobre los datos. El concepto de clase es fundamental en programación orientada a objetos. Una clase consta de métodos y datos. Los métodos de una clase definen el conjunto de operaciones permitidas sobre los datos de una clase (sus atributos). Una clase puede tener muchas instancias de la clase u objetos.

Clase abstracta: (abstract class) Superclase que contiene características comunes compartidas por las subclasses. Se declaran utilizando la palabra reservada `abstract`. Las clases abstractas pueden contener datos y métodos, pero no se pueden instanciar (crear objetos); es decir, no se pueden crear objetos de esta clase.

Compilación: (compilation) Proceso de traducción de un lenguaje de programación. Normalmente este proceso implica la traducción de un lenguaje de programación de alto nivel a lenguaje de programación de bajo nivel, o el formato binario de un conjunto de instrucciones específicas. La traducción se realiza con un programa denominado compilador. Un compilador java traduce los programas en bytecodes.

Depuración: (debugging) Proceso de encontrar, fijar y eliminar errores en un programa. Para estas tareas se suele utilizar una herramienta de programación conocida como depurador.

Excepción: (exception) Un suceso (evento) no previsto que indica que un programa ha fallado en alguna forma. Las excepciones se representan por objetos excepción en java. Las excepciones se manejan con un bloque de sentencias `try/catch`.

HTML (Hypertext Markup Language): Lenguaje de 'script' o de marcas para diseñar páginas Web para creación y compartición de documentos electrónicos integrados preparados para multimedia e Internet.

JVM, Máquina Virtual Java: (Java Virtual Machine) Una emulación de software de una máquina que puede ejecutar bytecodes de Java. Proporciona una implementación del procesador, sistema de memoria e interfaces a dispositivos hardware. Todos los programas Java se compilan a bytecodes que se ejecutan por una JVM.

Sobrecarga de un método: (method overloading) La sobrecarga de n método significa que se puede definir los métodos con el mismo nombre de una clase siempre que haya diferencia en sus parámetros.

Patrón: (pattern) Disposición avanzada de clases y objetos.

Sentencia: (statement) Una unidad de código que representa una acción o una secuencia de acciones. Las sentencias se ejecutan en el orden en que están escritas y siempre terminan en un punto y coma.

Lenguaje de programación: (programming language) Notación utilizada por los programadores para escribir programas. Un lenguaje tiene una sintaxis (las palabras y símbolos utilizadas para escribir códigos de programa), una gramática (las reglas que definen una secuencia de palabras y símbolos significativos y correctos) y semántica. Java es un lenguaje de programación.

UML: (UML) Lenguaje unificado de modelado que proporciona notación estándar visual para documentar el análisis y diseño de sistemas orientados a objetos.

BIBLIOGRAFÍA

CISNEROS ALMAZÁN, Rodolfo. *Riego y Drenaje* [en línea]. (Trabajo de titulación). (Apuntes) UNIVERSIDAD AUTÓNOMA DE SAN LUIS POTOSI. Facultad de Ingeniería. 2003 [Consulta: 11 octubre 2019]. Disponible en: <http://www.ingenieria.uaslp.mx/Documents/Apuntes/Riego%20y%20Drenaje.pdf.html>

DIRECCIÓN DE COMUNICACIÓN Y RELACIONES PÚBLICAS. *Centro Experimental del Riego de la ESPOCH un referente del Desarrollo Nacional* [blog]. [Consulta: 2 enero 2020]. Disponible en: <https://www.espoch.edu.ec/index.php/component/k2/item/1673-centro-experimental-del-riego-de-la-espoch-un-referente-del-desarrollo-nacional.html>

CALERO, C; MÓRAGA, M. & PIATTINI, M. *Calidad del Producto y Proceso de Software*. RA-MA editorial, 2010, pp. 1-688

GINESTA, G. & PÉREZ, O. *Base de Datos en PostgreSQL* [en línea]. 2006. [Consulta: 5 enero 2020]. Disponible en: [https://s3.amazonaws.com/academia.edu.documents/35811174/P06_M2109_02152.pdf?response-content-disposition=inline%3B%20filename%3DBases de datos en PostgreSQL.pdf&X-Amz.html](https://s3.amazonaws.com/academia.edu.documents/35811174/P06_M2109_02152.pdf?response-content-disposition=inline%3B%20filename%3DBases+de+datos+en+PostgreSQL.pdf&X-Amz.html)

GONZÁLEZ, M. (s. f.). *Herramienta de Desarrollo Netbeans*. pp. 1-5

GONZÁLEZ, Y. & ROMERO, Y. “Patrón Modelo – Vista Controlador”. *Revista Telemática* [en línea], 2012, 11(1), pp. 47-57. [Consulta: 10 octubre 2019]. Disponible en: <http://revistatelematica.cujae.edu.cu/index.php/tele/article.view/15>

IAGUA, redacción. *Aquadaia, la app que permite conocer la cantidad de agua necesaria en un cultivo para su crecimiento óptimo* [blog]. [Consulta: 11 octubre 2019]. Disponible en: <https://www.iagua.es/noticias/agricultura/14/06/13/aquadaia-la-app-que-permite-conocer-la-cantidad-de-agua-necesaria-en-un-cultivo-para-su-crecimiento>

ICAZA, P. & RUALES, C. *Ingeniería de Riego Moderno*. 3^oed. Editorian Santa Anam, 2009.

LEÓN, J. *Riegos y Drenajes*. Riobamba. Ediorial E-Copycenter. 2012. Pp. 1-66

MARTÍN, A. *Ajax en Java EE [en línea]*, 2^oed. Madrid-España: Editorial RA-MA, 2014.
[Consulta: 8 octubre 2019]. Disponible en:
<http://ebookcentral.proquest.com/lib/epochsp/detail.action?docID=5758900>

MONTE GALIANO, J. *Implantar scrum con éxito [en línea]*. Barcelona: Editorial UOC, 2016.
[Consulta: 11 octubre 2019]. Disponible en:
<http://ebookcentral.proquest.com/lib/epochsp/detail.action?docID=4795177>

PÉREZ MARTÍNEZ, E. *Hibernate: Persistencia de objetos en JEE, venciendo las diferencias entre el modelo relacional y el orientado a objetos [en línea]*. Madrid, RA-MA Editorial, 2015.
[Consulta: 10 octubre 2019]. Disponible en:
<http://ebookcentral.proquest.com/lib/epochsp/detail.action?docID=5758975>

PRIETO SÁEZ, N., & CASANOVA FAUS, A. *Empezar a programar usando Java [en línea]*. 3^o Ed. Valencia: Editorial de la Universidad Politécnica de Valencia, 2016. [Consulta: 8 octubre 2019].
Disponible en:
<http://ebookcentral.proquest.com/lib/epochsp/detail.action?docID=479508>

RAMÍREZ, P. *Lenguajes de Programación*. 2009.

SÁNCHEZ ALLENDE, J., HUECAS FERNÁNDEZ-Toribio, G., & MORENO DÍAZ, P. *Programación en Java 2 [en línea]*. Madrid: McGraw-Hill, 2005. [Consulta: 10 octubre 2019].
Disponible en: <http://ebookcentral.proquest.com/lib/epochsp/detail.action?docID=3195078>

ZOFÍO JIMÉNEZ, J. *Aplicaciones web [en línea]*. Madrid: Macmillan Iberia, S.A., 2013.
[Consulta: 11 octubre 2019]. Disponible en:
<http://ebookcentral.proquest.com/lib/epochsp/detail.action?docID=3217129>

ANEXOS

ANEXO A Manual técnico

**ESCUELA SUPERIOR POLITÉCNICA DE
CHIMBORAZO**

**FACULTAD DE INFORMÁTICA Y ELECTRÓNICA
ESCUELA DE INGENIERÍA EN SISTEMAS
CARRERA DE INGENIERÍA EN SISTEMAS**



MANUAL TÉCNICO

**“DESARROLLO DE UN SISTEMA WEB PARA LA
AUTOMATIZACIÓN DEL PROCESO DE CALENDARIZACIÓN
DE RIEGO PARA EL CENTRO EXPERIMENTAL DEL RIEGO DE
LA ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO”**

Presentado por:

ISAAC DAVID TORRES PAREDES

Riobamba - Ecuador

MANUAL TÉCNICO

En el presente manual técnico se detalla el proceso de desarrollo de la aplicación web que se encarga de generar de manera automática un calendario de riego para el centro experimental del riego de la ESPOCH, aplicando la metodología ágil SCRUM.

Para la definición de las historias de usuario se siguió un proceso ordenado, basado en la característica principal de la metodología SCRUM que es el Sprint, partiendo de una planificación inicial, determinando: tareas, actividades y pruebas de aceptación, siendo esta información útil para el supervisor del desarrollo del proyecto mediante la cual se puede llevar una constancia del trabajo realizado.

El desarrollo del proyecto denominado: “DESARROLLO DE UN SISTEMA WEB PARA LA AUTOMATIZACIÓN DEL PROCESO DE CALENDARIZACIÓN DE RIEGO PARA EL CENTRO EXPERIMENTAL DEL RIEGO DE LA ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO”, el cual se encarga de generar los calendarios de riego de manera automática a partir de la información inicial del cultivo.

El sistema consiste en llenar el formulario de las variables necesarias para calcular el calendario de riego como por ejemplo: cultivo, tipo de suelo, área de siembra, fecha de siembra, umbral, etc. Estas variables son procesadas por la aplicación calculando cada una de las sub-variables necesarias para crear el calendario de riego final. Los reportes generados por el sistema se lo realizaran en formato PDF el cual permitirá visualizar e imprimir el calendario de riego final.

A continuación se detalla cada una de las historias técnicas, de usuarios con sus respectivas tareas de ingenierías y pruebas de aceptación.

Historias Técnicas Realizadas en el Sprint 1

HISTORIA TÉCNICA	
Número: HT-01	Nombre: Investigar en que consiste el proceso de la elaboración del calendario de riego.
Usuario: Desarrollador	Spring Asignado: 1
Puntos Estimados: 16	Puntos Reales: 20
Descripción: Como desarrollador se necesita investigar en que consiste el proceso desde el inicio hasta el fin de la elaboración de un calendario de riego y las herramientas que se utilizan para elaborar el mismo.	
Pruebas de aceptación:	
<ul style="list-style-type: none"> • Verificar que se ha comprendido el proceso para elaborar un calendario de riego. 	

PRUEBA DE ACEPTACIÓN	
Código: HT-01. PA-01	Nombre Historia: Investigar en que consiste el proceso de la elaboración del calendario de riego.
Nombre prueba: Verificar que se hay comprendido todo el proceso para elaborar un calendario de riego.	
Responsable: Ing. Juan León	Fecha: 18/09/2019
Descripción: Se revisa y verifica que se haya comprendido todo el proceso de elaborar un calendario de riego por parte del desarrollador.	
Condiciones de ejecución:	
Pasos de ejecución: <ul style="list-style-type: none"> • Dirigirse al Centro Experimental del Riego. • Estar presente en la toma de datos necesarios para el cálculo del calendario de riego. • Estar presente en la elaboración del calendario del riego. • Anotar el paso a paso del proceso de elaboración. 	
Resultado Esperado: Comprensión del proceso de elaboración de un calendario de riego.	
Evaluación de la prueba: Exitosa	

TAREA DE INGENIERÍA	
Sprint: 1	Número de tarea: 01
Nombre historia: Investigar en que consiste el proceso de la elaboración del calendario de riego.	
Nombre Tarea: Comprender el proceso de la elaboración del calendario de riego.	
Responsable: Isaac Torres	Tipo de Tarea: Desarrollo
Fecha Inicio: 09/09/2019	Fecha Fin: 18/09/2019
Descripción: Se debe de comprender como realizar un calendario de riego desde la toma de datos del campo hasta los cálculos matemáticos y la elaboración del reporte final.	
Pruebas de aceptación: <ul style="list-style-type: none"> • Verificar que se logre presentar el reporte de un calendario de riego elaborado. 	

PRUEBA DE ACEPTACIÓN	
Código: HT-01.PA-02	Nombre Historia: Investigar en que consiste el proceso de la elaboración del calendario de riego.

Nombre prueba: Verificar que se logre presentar el reporte de un calendario de riego elaborado.	
Responsable: Ing. Juan León	Fecha: 18/09/2019
Descripción: Se revisa que el desarrollador sea capaz de realizar un calendario de riego sin errores.	
Condiciones de ejecución: Debe de contar con todos los datos necesarios para realizar un calendario de riego.	
Pasos de ejecución: <ul style="list-style-type: none"> • Tomar los datos del campo • Realizar cálculos. • Realizar el reporte final. 	
Resultado Esperado: Calendario de riego elaborado.	
Evaluación de la prueba: Exitosa	

HISTORIA TÉCNICA	
Número: HT-03	Nombre: Definir los requerimientos funcionales y no funcionales del proyecto a realizarse.
Usuario: Desarrollador	Spring Asignado: 1
Puntos Estimados: 8	Puntos Reales: 10
Descripción: Como desarrollador se definir los requerimientos funcionales y no funcionales del proyecto a realizarse.	
Pruebas de aceptación: <ul style="list-style-type: none"> • Verificar que se exista la lista de requerimientos funcionales y no funcionales del sistema. 	

PRUEBA DE ACEPTACIÓN	
Código: HT-03. PA-01	Nombre Historia: Definir los requerimientos funcionales y no funcionales del proyecto a realizarse.
Nombre prueba: Verificar que se exista la lista de requerimientos funcionales y no funcionales del sistema.	
Responsable: Ing. Alejandra Oñate	Fecha: 20/09/2019
Descripción: Se revisa y verifica que se haya establecido la lista de requerimientos tanto funcionales como no funcionales.	

Condiciones de ejecución:
Pasos de ejecución: <ul style="list-style-type: none"> • Reunirse con el director del Centro Experimental del Riego. • Enlista de requerimientos solicitados por el director.
Resultado Esperado: Lista de requerimientos funcionales y no funcionales.
Evaluación de la prueba: Exitosa

HISTORIA TÉCNICA	
Número: HT-05	Nombre: Analizar y diseñar de manera preliminar de la base de datos.
Usuario: Desarrollador	Spring Asignado: 1
Puntos Estimados: 16	Puntos Reales: 20
Descripción: Como desarrollador se necesita analizar y diseñar de manera preliminar la base de datos para la aplicación web.	
Pruebas de aceptación: <ul style="list-style-type: none"> • Verificar que se haya diseñado de forma preliminar la base de datos del sistema. 	

PRUEBA DE ACEPTACIÓN	
Código: HT-05. PA-01	Nombre Historia: Analizar y diseñar de manera preliminar de la base de datos.
Nombre prueba: Verificar que se haya diseñado de forma preliminar la base de datos del sistema.	
Responsable: Ing. Alejandra Oñate	Fecha: 25/09/2019
Descripción: Se revisa y verifica que se ha elaborado el borrador de las entidades y sus relaciones de la base de datos para la aplicación web.	
Condiciones de ejecución:	
Pasos de ejecución: <ul style="list-style-type: none"> • Definir entidades fuertes. • Definir entidades débiles. • Crear relaciones entre entidades. 	
Resultado Esperado: Diseño preliminar de la base de datos para la aplicación web.	
Evaluación de la prueba: Exitosa	

TAREA DE INGENIERÍA	
Sprint: 1	Número de tarea: 01
Nombre historia: Analizar y diseñar de manera preliminar de la base de datos.	
Nombre Tarea: Desarrollar el modelo entidad relación de la base de datos.	
Responsable: Isaac Torres	Tipo de Tarea: Desarrollo
Fecha Inicio: 23/09/2019	Fecha Fin: 25/09/2019
Descripción: Se debe de identificar las entidades débiles y fuertes conjuntamente con sus relaciones y diseñar en modelo entidad relación.	
Pruebas de aceptación:	
<ul style="list-style-type: none"> • Verificar que se haya desarrollado el modelo de las entidades con sus relaciones de la base de datos para la aplicación web. 	

PRUEBA DE ACEPTACIÓN	
Código: HT-05.PA-02	Nombre Historia: Analizar y diseñar de manera preliminar de la base de datos.
Nombre prueba: Verificar que se haya desarrollado el modelo de las entidades con sus relaciones de la base de datos para la aplicación web.	
Responsable: Ing. Alejandra Oñate.	Fecha: 25/10/2019
Descripción: Se revisa que el desarrollador sea capaz de realizar un calendario de riego sin errores.	
Condiciones de ejecución: Debe de contar con todos requerimientos establecidos por el director del CER (cliente).	
Pasos de ejecución:	
<ul style="list-style-type: none"> • Comparar entre la lista de requerimientos y las entidades del modelo. • Realizar una secuencia de las entidades para comprender la lógica del modelo. • Observar el modelo entidad relación tenga todos los atributos. 	
Resultado Esperado: Modelo entidad de la relación de la base de datos.	
Evaluación de la prueba: Exitosa	

HISTORIA TÉCNICA	
Número: HT-13	Nombre: Documentar cada uno de las historias técnicas y actividades realizadas hasta la presente fecha en el proyecto.
Usuario: Desarrollador	Spring Asignado: 1
Puntos Estimados: 4	Puntos Reales: 4

Descripción: Como desarrollador se necesita documentar cada uno de las historias técnicas realizadas hasta la presente fecha.
Pruebas de aceptación: <ul style="list-style-type: none"> • Verificar que exista la documentación de las historias realizadas.

PRUEBA DE ACEPTACIÓN	
Código: HT-13. PA-01	Nombre Historia: Documentar cada uno de las historias técnicas y actividades realizadas hasta la presente fecha en el proyecto.
Nombre prueba: Verificar que exista la documentación de las historias realizadas.	
Responsable: Ing. Alejandra Oñate	Fecha: 26/09/2019
Descripción: Se verifica que exista la documentación donde se evidencie la realización de las historias técnicas desarrolladas con anterioridad.	
Condiciones de ejecución: Tener el manual técnico del proyecto.	
Pasos de ejecución: <ul style="list-style-type: none"> • Observar que existan las tablas en donde se describe las historias técnicas del proyecto. 	
Resultado Esperado: Existe la documentación de las historias técnicas.	
Evaluación de la prueba: Exitosa	

Historias Técnicas Realizadas en el Sprint 2

HISTORIA TÉCNICA	
Número: HT-04	Nombre: Definir el estándar de programación.
Usuario: Desarrollador	Spring Asignado: 2
Puntos Estimados: 4	Puntos Reales: 4
Descripción: Como desarrollador se necesita definir el estándar de programación.	
Pruebas de aceptación: <ul style="list-style-type: none"> • Verificar que se haya establecido y documentado el estándar de programación. 	
PRUEBA DE ACEPTACIÓN	
Código: HT-04. PA-01	Nombre Historia: Definir el estándar de programación.
Nombre prueba: Verificar que se haya establecido y documentado el estándar de programación.	
Responsable: Ing. Alejandra Oñate	Fecha: 27/09/2019

Descripción: Se verifica que se ha establecido el estándar de programación y se ha realizado la respectiva documentación.
Condiciones de ejecución: Conocer la arquitectura de la aplicación web. Conocer las herramientas que se va a utilizar para el desarrollo de la aplicación web.
Pasos de ejecución: <ul style="list-style-type: none"> • Observar el documento del estándar de programación. • Comparar si el estándar propuesto está acorde con los estándares predefinidos.
Resultado Esperado: Se ha establecido un estándar para el código de desarrollo de la aplicación.
Evaluación de la prueba: Exitosa

HISTORIA TÉCNICA	
Número: HT-07	Nombre: Desarrollar e implantar la base de datos en un Sistema Gestor de Bases de Datos.
Usuario: Desarrollador	Spring Asignado: 2
Puntos Estimados: 16	Puntos Reales: 22
Descripción: Como desarrollador se necesita desarrollar e implantar de la base de datos en un Sistema Gestor de Bases de Datos.	
Pruebas de aceptación: <ul style="list-style-type: none"> • Verificar que se haya creado la base de datos. 	

PRUEBA DE ACEPTACIÓN	
Código: HT-07. PA-01	Nombre Historia: Desarrollar e implantar de la base de datos en un Sistema Gestor de Bases de Datos.
Nombre prueba: Verificar que se haya creado la base de datos.	
Responsable: Ing. Alejandra Oñate	Fecha: 04/10/2019
Descripción: Se verifica que se haya creado la base de datos del sistema en el gestor de bases de datos.	
Condiciones de ejecución: Tener el modelo entidad relación.	
Pasos de ejecución: <ul style="list-style-type: none"> • Ingresar al sistema gestor de base de datos. • Desplegar el contenedor de bases de datos. 	

<ul style="list-style-type: none"> • Ingresar con las credenciales a la base de datos.
Resultado Esperado: Ingreso a la base de datos.
Evaluación de la prueba: Exitosa

TAREA DE INGENIERÍA	
Sprint: 2	Número de tarea: 01
Nombre historia: Desarrollar e implantar de la base de datos en un Sistema Gestor de Bases de Datos.	
Nombre Tarea: Crear las tablas en las bases de datos	
Responsable: Isaac Torres	Tipo de Tarea: Desarrollo
Fecha Inicio: 30/09/2019	Fecha Fin: 02/10/2019
Descripción: Se debe crear las tablas que representan a las entidades fuertes y débiles del modelo entidad relación.	
Pruebas de aceptación:	
<ul style="list-style-type: none"> • Verificar que se haya creado las tablas con todos los campos correspondientes a cada una. 	

PRUEBA DE ACEPTACIÓN	
Código: H7-01.PA-02	Nombre Historia: Desarrollar e implantar de la base de datos en un Sistema Gestor de Bases de Datos.
Nombre prueba: Verificar que se haya creado las tablas con todos los campos correspondientes a cada una.	
Responsable: Ing. Alejandra Oñate	Fecha: 04/10/2019
Descripción: Se revisa que estén creadas todas las tablas correspondientes a cada entidad del MER.	
Condiciones de ejecución: Debe de estar creada la base de datos	
Pasos de ejecución:	
<ul style="list-style-type: none"> • Ingresar al gestor de base de datos. • Ingresar a la base de datos del sistema. • Ingresar al esquema público de la base. • Desplegar el paquete de tablas. 	
Resultado Esperado: Tablas de la base de datos creadas.	
Evaluación de la prueba: Exitosa	

TAREA DE INGENIERÍA	
Sprint: 2	Número de tarea: 02
Nombre historia: Desarrollar e implantar de la base de datos en un Sistema Gestor de Bases de Datos.	
Nombre Tarea: Crear las relaciones y llaves primarias en las tablas.	
Responsable: Isaac Torres	Tipo de Tarea: Desarrollo
Fecha Inicio: 03/10/2019	Fecha Fin: 04/10/2019
Descripción: Se debe crear las relaciones y las llaves primarias y foráneas que están relacionadas entre sí.	
Pruebas de aceptación:	
<ul style="list-style-type: none"> • Verificar que se haya creado las llaves primarias y secundarias en las tablas de las bases de datos. 	

PRUEBA DE ACEPTACIÓN	
Código: H7-01.PA-03	Nombre Historia: Desarrollar e implantar de la base de datos en un Sistema Gestor de Bases de Datos.
Nombre prueba: Verificar que se haya creado las llaves primarias y secundarias en las tablas de las bases de datos.	
Responsable: Ing. Alejandra Oñate	Fecha: 04/10/2019
Descripción: Se revisa que estén creadas todas las llaves primarias y foráneas en tablas de la base de datos.	
Condiciones de ejecución: Debe de estar creada las tablas.	
Pasos de ejecución:	
<ul style="list-style-type: none"> • Ingresar al gestor de base de datos. • Ingresar a la base de datos del sistema. • Ingresar al esquema público de la base. • Desplegar el paquete de tablas. • Cambiar las propiedades de las tablas. 	
Resultado Esperado: Llaves primarias y foráneas creadas en todas las tablas de la base de datos.	
Evaluación de la prueba: Exitosa	

HISTORIA TÉCNICA

Número: HT-08	Nombre: Desarrollar la interfaz del formulario de ingreso.
Usuario: Desarrollador	Spring Asignado: 2
Puntos Estimados: 20	Puntos Reales: 25
Descripción: Como desarrollador se necesita desarrollar la interfaz del formulario de ingreso de datos para el cálculo del calendario de riego.	
Pruebas de aceptación:	
<ul style="list-style-type: none"> • Verificar que se haya creado la interfaz de ingreso de datos. 	

PRUEBA DE ACEPTACIÓN	
Código: HT-08. PA-01	Nombre Historia: Desarrollar la interfaz del formulario de ingreso.
Nombre prueba: Verificar que se haya creado la interfaz de ingreso de datos.	
Responsable: Ing. Alejandra Oñate	Fecha: 11/10/2019
Descripción: Se revisa y verifica que se haya creado la interfaz de usuario de ingreso de datos a la aplicación web.	
Condiciones de ejecución:	
Pasos de ejecución:	
<ul style="list-style-type: none"> • Entrar en el servidor local de la computadora. (localhost:4040) 	
Resultado Esperado: Visualizar la interfaz de ingreso de datos a la aplicación web.	
Evaluación de la prueba: Exitosa	

TAREA DE INGENIERÍA	
Sprint: 2	Número de tarea: 01
Nombre historia: Desarrollar la interfaz del formulario de ingreso.	
Nombre Tarea: Desarrollar el archivo index.JSP con las etiquetas HTML de la interfaz de ingreso de datos.	
Responsable: Isaac Torres	Tipo de Tarea: Desarrollo
Fecha Inicio: 07/10/2019	Fecha Fin: 09/10/2019
Descripción: Se debe realizar la estructura de la interfaz de ingreso de datos a la aplicación.	
Pruebas de aceptación:	
<ul style="list-style-type: none"> • Verificar que se logre presentar en pantalla el formulario de ingreso de datos necesarios para el cálculo del calendario de riego. 	

PRUEBA DE ACEPTACIÓN	
Código: HT-08.PA-02	Nombre Historia Desarrollar la interfaz del formulario de ingreso.
Nombre prueba: Verificar que se logre presentar en pantalla el formulario de ingreso de datos necesarios para el cálculo del calendario de riego	
Responsable: Ing. Alejandra Oñate	Fecha: 11/10/2019
Descripción: Se revisa que exista el archivo index.jsp en el que está el formulario de ingreso de datos.	
Condiciones de ejecución: <ul style="list-style-type: none"> • Debe de haberse creado el proyecto en netbeans. • Estar levantado el servidor Payara Server. • Estar desplegado el proyecto en el servidor. 	
Pasos de ejecución: <ul style="list-style-type: none"> • Ingresar en el servidor local. • Visualizar el formulario en pantalla. 	
Resultado Esperado Visualizar el formulario en pantalla.	
Evaluación de la prueba: Exitosa	

TAREA DE INGENIERÍA	
Sprint: 2	Número de tarea: 02
Nombre historia: Desarrollar la interfaz del formulario de ingreso.	
Nombre Tarea: Desarrollar las hojas de estilo para el formulario de ingreso de datos.	
Responsable: Isaac Torres	Tipo de Tarea: Desarrollo
Fecha Inicio: 09/10/2019	Fecha Fin: 11/10/2019
Descripción: Se debe realizar la estructura de la interfaz de ingreso de datos a la aplicación.	
Pruebas de aceptación: <ul style="list-style-type: none"> • Verificar que el formulario tenga los estilos determinados en las CSS. 	

PRUEBA DE ACEPTACIÓN	
Código: HT-08.PA-03	Nombre Historia Desarrollar la interfaz del formulario de ingreso.
Nombre prueba: Verificar que el formulario tenga los estilos determinados en las CSS.	
Responsable: Ing. Alejandra Oñate	Fecha: 11/10/2019

Descripción: Se revisa que existan los archivos correspondientes a las hojas de estilo para el formulario de ingreso.
Condiciones de ejecución: <ul style="list-style-type: none"> • Debe de haberse creado el proyecto en netbeans. • Estar levantado el servidor Payara Server. • Estar desplegado el proyecto en el servidor.
Pasos de ejecución: <ul style="list-style-type: none"> • Ingresar en el servidor local. • Visualizar el formulario en pantalla con los estilos de las CSS.
Resultado Esperado Visualizar el formulario en pantalla con los estilos establecidos.
Evaluación de la prueba: Exitosa

HISTORIA TÉCNICA	
Número: HT-09	Nombre: Desarrollar el formato de la tabla del calendario de riego.
Usuario: Desarrollador	Spring Asignado: 2
Puntos Estimados: 16	Puntos Reales: 20
Descripción: Como desarrollador se desarrollar el formato de la tabla del calendario de riego.	
Pruebas de aceptación: <ul style="list-style-type: none"> • Verificar que se haya creado la tabla en la que se va a mostrar el calendario de riego. 	

PRUEBA DE ACEPTACIÓN	
Código: HT-09. PA-01	Nombre Historia: Desarrollar el formato de la tabla del calendario de riego.
Nombre prueba: Verificar que se haya creado la tabla en la que se va a mostrar el calendario de riego.	
Responsable: Ing. Alejandra Oñate	Fecha: 17/10/2019
Descripción: Se revisa y verifica que se ha creado la tabla en la que se va a visualizar el calendario de riego.	
Condiciones de ejecución:	
Pasos de ejecución: <ul style="list-style-type: none"> • Ingresar al sistema. • Seleccionar generar calendario. 	

Resultado Esperado: Visualizar el formato de la tabla en donde se muestra el calendario del riego.
Evaluación de la prueba: Exitosa

TAREA DE INGENIERÍA	
Sprint: 2	Número de tarea: 01
Nombre historia: Desarrollar el formato de la tabla del calendario de riego.	
Nombre Tarea: Desarrollar el formato de la tabla donde se muestra el calendario de riego.	
Responsable: Isaac Torres	Tipo de Tarea: Desarrollo
Fecha Inicio: 14/10/2019	Fecha Fin: 17/10/2019
Descripción: Se debe desarrollar una tabla que contenga todas las columnas que se muestran en un calendario de riego.	
Pruebas de aceptación:	
<ul style="list-style-type: none"> • Verificar que se presente en pantalla el formato de la tabla del calendario del riego. 	

PRUEBA DE ACEPTACIÓN	
Código: HT-09.PA-02	Nombre Historia: Desarrollar el formato de la tabla del calendario de riego.
Nombre prueba: Verificar que se presente en pantalla el formato de la tabla del calendario del riego.	
Responsable: Ing. Alejandra Oñate	Fecha: 17/10/2019
Descripción: Se revisa que se muestre en pantalla el formato del calendario de riego en una tabla.	
Condiciones de ejecución:	
<ul style="list-style-type: none"> • Estar desplegado el proyecto en el servidor local. 	
Pasos de ejecución:	
<ul style="list-style-type: none"> • Ingresar al sistema. • Seleccionar generar calendario. 	
Resultado Esperado: Visualizar en pantalla el formato del calendario de riego en pantalla.	
Evaluación de la prueba: Exitosa	

HISTORIA TÉCNICA	
Número: HT-13	Nombre: Documentar cada uno de las historias técnicas y actividades realizadas hasta la presente fecha en el proyecto.
Usuario: Desarrollador	Spring Asignado: 2
Puntos Estimados: 4	Puntos Reales: 4
Descripción: Como desarrollador se necesita documentar cada uno de las historias técnicas realizadas hasta la presente fecha.	
Pruebas de aceptación:	
<ul style="list-style-type: none"> • Verificar que exista la documentación de las historias realizadas. 	

PRUEBA DE ACEPTACIÓN	
Código: HT-13. PA-01	Nombre Historia: Documentar cada uno de las historias técnicas y actividades realizadas hasta la presente fecha en el proyecto.
Nombre prueba: Verificar que exista la documentación de las historias realizadas.	
Responsable: Ing. Alejandra Oñate	Fecha: 26/09/2019
Descripción: Se verifica que exista la documentación donde se evidencie la realización de las historias técnicas desarrolladas con anterioridad.	
Condiciones de ejecución: Tener el manual técnico del proyecto.	
Pasos de ejecución:	
<ul style="list-style-type: none"> • Observar que existan las tablas en donde se describe las historias técnicas del proyecto. 	
Resultado Esperado: Existe la documentación de las historias técnicas.	
Evaluación de la prueba: Exitosa	

Historias Técnicas Realizadas en el Sprint 3

HISTORIA USUARIO	
Número: HU-01	Nombre: Ingresar en la base de datos los valores de las constantes.
Usuario: Desarrollador	Spring Asignado: 3
Puntos Estimados: 20	Puntos Reales: 18
Descripción: Como desarrollador se necesita ingresar en la base de datos los valores de las constantes.	
Pruebas de aceptación:	

<ul style="list-style-type: none"> • Verificar que se estén ingresadas los valores de las constantes numéricas en la base de datos.
--

PRUEBA DE ACEPTACIÓN	
Código: HU-01. PA-01	Nombre Historia: Ingresar en la base de datos los valores de las constantes.
Nombre prueba: Verificar que se estén ingresadas los valores de las constantes numéricas en la base de datos.	
Responsable: Ing. Alejandra Oñate	Fecha: 25/10/2019
Descripción: Se revisa y verifica que estén ingresadas en la base de datos los valores de las constantes numéricas necesarias para la elaboración del calendario de riego.	
Condiciones de ejecución:	
<ul style="list-style-type: none"> • Debe de estar creada la tabla constante. 	
Pasos de ejecución:	
<ul style="list-style-type: none"> • Ingresar el sistema gestor de la base de datos. • Ingresar en la base de datos. • Realizar una consulta de selección a la tabla constante. 	
Resultado Esperado: Retornar el valor de las constantes numéricas ingresadas.	
Evaluación de la prueba: Exitosa	

TAREA DE INGENIERÍA	
Sprint: 3	Número de tarea: 01
Nombre historia: Ingresar en la base de datos los valores de las constantes.	
Nombre Tarea: Crear tabla constantes en la base de datos.	
Responsable: Isaac Torres	Tipo de Tarea: Desarrollo
Fecha Inicio: 21/10/2019	Fecha Fin: 23/10/2019
Descripción: Se debe de crear la tabla con todos los campos y los tipos de datos correspondientes a la tabla constante.	
Pruebas de aceptación:	
<ul style="list-style-type: none"> • Verificar que se cree la tabla constante en la base de datos. 	

PRUEBA DE ACEPTACIÓN	
Código: HU-01.PA-02	Nombre Historia: Ingresar en la base de datos los valores de las constantes.

Nombre prueba: Verificar que se cree la tabla constante en la base de datos.	
Responsable: Ing. Alejandra Oñate	Fecha: 25/10/2019
Descripción: Se revisa que este creada la tabla donde se almacena los datos numéricos de las constantes necesarias para la elaboración del calendario de riego.	
Condiciones de ejecución: Existencia de la base de datos.	
Pasos de ejecución: <ul style="list-style-type: none"> • Ingresar al SGBD. • Ingresar en la base de datos. • Ejecutar la consulta que crea la tabla constante. 	
Resultado Esperado: Creación de la tabla constante en la base de datos.	
Evaluación de la prueba: Exitosa	

TAREA DE INGENIERÍA	
Sprint: 3	Número de tarea: 02
Nombre historia Ingresar en la base de datos los valores de las constantes.	
Nombre Tarea: Ingresar las tuplas en la tabla constantes.	
Responsable: Isaac Torres	Tipo de Tarea: Desarrollo
Fecha Inicio: 23/10/2019	Fecha Fin: 25/10/2019
Descripción: Se debe ingresar los datos de cada una de las constantes.	
Pruebas de aceptación: <ul style="list-style-type: none"> • Verificar que se haya ingresado los datos en la tabla. 	

PRUEBA DE ACEPTACIÓN	
Código: HU-01.PA-03	Nombre Historia: Ingresar en la base de datos los valores de las constantes.
Nombre prueba: Verificar que se haya ingresado los datos en la tabla.	
Responsable: Ing. Alejandra Oñate	Fecha: 25/10/2019
Descripción: Se revisa que este ingresadas las tuplas en la tabla donde se almacena los datos numéricos de las constantes necesarias para la elaboración del calendario de riego.	
Condiciones de ejecución: Existencia de la base de datos. Existencia de la tabla constante.	
Pasos de ejecución: <ul style="list-style-type: none"> • Ingresar al SGBD. 	

<ul style="list-style-type: none"> • Ingresar en la base de datos. • Ejecutar la consulta que recupera los datos de la tabla constantes.
Resultado Esperado: Visualizar los datos de la tabla constantes.
Evaluación de la prueba: Exitosa

HISTORIA USUARIO	
Número: HU-02	Nombre: Ingresar en la base de datos los valores de los cultivos
Usuario: Desarrollador	Spring Asignado: 3
Puntos Estimados: 16	Puntos Reales: 20
Descripción: Como desarrollador se necesita ingresar en la base de datos los valores de los cultivos.	
Pruebas de aceptación:	
<ul style="list-style-type: none"> • Verificar que se estén ingresadas los valores de los cultivos en la base de datos. 	

PRUEBA DE ACEPTACIÓN	
Código: HU-02. PA-01	Nombre Historia: Ingresar en la base de datos los valores de los cultivos.
Nombre prueba: Verificar que se estén ingresadas los valores de los cultivos en la base de datos.	
Responsable: Ing. Alejandra Oñate	Fecha: 31/10/2019
Descripción: Se revisa y verifica que estén ingresadas en la base de datos los valores de los cultivos necesarios para la elaboración del calendario de riego.	
Condiciones de ejecución:	
<ul style="list-style-type: none"> • Debe de estar creada la tabla cultivo. 	
Pasos de ejecución:	
<ul style="list-style-type: none"> • Ingresar el sistema gestor de la base de datos. • Ingresar en la base de datos. • Realizar una la inserción de los datos del cultivo en la tabla. 	
Resultado Esperado: Ingresar los datos de los cultivos en las tablas.	
Evaluación de la prueba: Exitosa	

TAREA DE INGENIERÍA	
Sprint: 3	Número de tarea: 01

Nombre historia Ingresar en la base de datos los valores de los cultivos.	
Nombre Tarea: Ingresar los datos en la tabla cultivo en la base de datos.	
Responsable: Isaac Torres	Tipo de Tarea: Desarrollo
Fecha Inicio: 28/10/2019	Fecha Fin: 31/10/2019
Descripción: Se debe ingresar todos los valores correspondientes a las tablas cultivos.	
Pruebas de aceptación:	
<ul style="list-style-type: none"> • Verificar que hayan ingresado los valores de los cultivos en la tabla. 	

PRUEBA DE ACEPTACIÓN	
Código: HU-02.PA-02	Nombre Historia: Ingresar en la base de datos los valores de los cultivos.
Nombre prueba: Verificar que hayan ingresado los valores de los cultivos en la tabla	
Responsable: Ing. Alejandra Oñate	Fecha: 31/10/2019
Descripción: Se revisa que está ingresada la información en la tabla cultivo.	
Condiciones de ejecución:	
Existencia de la base de datos.	
Existencia de la tabla cultivo.	
Pasos de ejecución:	
<ul style="list-style-type: none"> • Ingresar al SGBD. • Ingresar en la base de datos. • Ejecutar la consulta que recupera la información de la tabla cultivo. 	
Resultado Esperado: Visualizar los datos de los cultivos.	
Evaluación de la prueba: Exitosa	

HISTORIA TÉCNICA	
Número: HT-13	Nombre: Documentar cada uno de las historias técnicas y actividades realizadas hasta la presente fecha en el proyecto.
Usuario: Desarrollador	Spring Asignado: 3
Puntos Estimados: 4	Puntos Reales: 4
Descripción: Como desarrollador se necesita documentar cada uno de las historias técnicas realizadas hasta la presente fecha.	
Pruebas de aceptación:	
<ul style="list-style-type: none"> • Verificar que exista la documentación de las historias realizadas. 	

PRUEBA DE ACEPTACIÓN	
Código: HT-13. PA-01	Nombre Historia: Documentar cada uno de las historias técnicas y actividades realizadas hasta la presente fecha en el proyecto.
Nombre prueba: Verificar que exista la documentación de las historias realizadas.	
Responsable: Ing. Alejandra Oñate	Fecha: 26/09/2019
Descripción: Se verifica que exista la documentación donde se evidencie la realización de las historias técnicas desarrolladas con anterioridad.	
Condiciones de ejecución: Tener el manual técnico del proyecto.	
Pasos de ejecución:	
<ul style="list-style-type: none"> • Observar que existan las tablas en donde se describe las historias técnicas del proyecto. 	
Resultado Esperado: Existe la documentación de las historias técnicas.	
Evaluación de la prueba: Exitosa	

Historias Técnicas Realizadas en el Sprint 4

HISTORIA TDE USUARIO	
Número: HU-03	Nombre: Ingresar los valores del coeficiente del cultivo determinado por la FAO.
Usuario: Desarrollador	Spring Asignado: 4
Puntos Estimados: 16	Puntos Reales: 20
Descripción: Como desarrollador se necesita ingresar los valores del coeficiente del cultivo determinado por la FAO.	
Pruebas de aceptación:	
<ul style="list-style-type: none"> • Verificar que se ha Ingresado los valores del coeficiente del cultivo determinado por la FAO. 	

PRUEBA DE ACEPTACIÓN	
Código: HU-03. PA-01	Nombre Historia: Ingresar los valores del coeficiente del cultivo determinado por la FAO.
Nombre prueba: Verificar que se ha ingresado los valores del coeficiente del cultivo determinado por la FAO.	
Responsable: Ing. Juan León	Fecha: 15/10/2019

Descripción: Se revisa y verifica que se ha ingresado los valores del coeficiente del cultivo determinado por la FAO.
Condiciones de ejecución:
Pasos de ejecución: <ul style="list-style-type: none"> • Dirigirse el Centro Experimental del Riego. • Estar presente en la toma de datos necesarios para el cálculo del calendario de riego. • Estar presente en la elaboración del calendario del riego. • Anotar el paso a paso del proceso de elaboración.
Resultado Esperado: Ingreso de los valores del coeficiente del cultivo determinado por la FAO.
Evaluación de la prueba: Exitosa

TAREA DE INGENIERÍA	
Sprint: 4	Número de tarea: 01
Nombre historia: Ingresar los valores del coeficiente del cultivo determinado por la FAO.	
Nombre Tarea: Comprender el proceso de la elaboración del calendario de riego.	
Responsable: Isaac Torres	Tipo de Tarea: Desarrollo
Fecha Inicio: 15/10/2019	Fecha Fin: 20/10/2019
Descripción: Se debe de ingresar los valores del coeficiente del cultivo determinado por la FAO.	
Pruebas de aceptación: <ul style="list-style-type: none"> • Verificar que se logre haya Ingresado los valores del coeficiente del cultivo determinado por la FAO. 	

PRUEBA DE ACEPTACIÓN	
Código: HU-03.PA-02	Nombre Historia: Ingresar los valores del coeficiente del cultivo determinado por la FAO.
Nombre prueba: Verificar que se haya ingresado los valores del coeficiente del cultivo determinado por la FAO.	
Responsable: Ing. Juan León	Fecha: 20/10/2019
Descripción: Se revisa que el sistema tenga la información de los cultivos determinados por la FAO.	
Condiciones de ejecución: Debe de contar con todos los datos necesarios para realizar un calendario de riego.	

Pasos de ejecución: <ul style="list-style-type: none"> • Tomar los datos del campo • Realizar cálculos. • Realizar el reporte final.
Resultado Esperado: Datos de la FAO ingresados.
Evaluación de la prueba: Exitosa

HISTORIA DE USUARIO	
Número: HU-04	Nombre: Ingresar en el sistema los tipos de suelo existentes.
Usuario: Desarrollador	Spring Asignado: 4
Puntos Estimados: 16	Puntos Reales: 20
Descripción: Como desarrollador se necesita ingresar en el sistema los tipos de suelo existentes.	
Pruebas de aceptación: <ul style="list-style-type: none"> • Verificar que se haya ingresado en el sistema los tipos de suelo existentes. 	

PRUEBA DE ACEPTACIÓN	
Código: HU-04. PA-01	Nombre Historia: Ingresar en el sistema los tipos de suelo existentes.
Nombre prueba: Verificar que se haya ingresado en el sistema los tipos de suelo existentes.	
Responsable: Ing. Juan León	Fecha: 15/10/2019
Descripción: Se revisa y verifica que se haya ingresado en el sistema los tipos de suelo existentes.	
Condiciones de ejecución:	
Pasos de ejecución: <ul style="list-style-type: none"> • Dirigirse el Centro Experimental del Riego. • Estar presente en la toma de datos necesarios para el cálculo del calendario de riego. • Estar presente en la elaboración del calendario del riego. • Anotar el paso a paso del proceso de elaboración. 	
Resultado Esperado: Ingreso en el sistema los tipos de suelo existentes.	
Evaluación de la prueba: Exitosa	

TAREA DE INGENIERÍA	
Sprint: 4	Número de tarea: 01
Nombre historia: Ingresar en el sistema los tipos de suelo existentes.	
Nombre Tarea: Comprender el proceso de la elaboración del calendario de riego.	
Responsable: Isaac Torres	Tipo de Tarea: Desarrollo
Fecha Inicio: 15/10/2019	Fecha Fin: 20/10/2019
Descripción: Se debe de ingresar en el sistema los tipos de suelo existentes.	
Pruebas de aceptación:	
<ul style="list-style-type: none"> • Verificar que se logre ingresar en el sistema los tipos de suelo existentes. 	

PRUEBA DE ACEPTACIÓN	
Código: HU-04.PA-02	Nombre Historia: Ingresar en el sistema los tipos de suelo existentes.
Nombre prueba: Verificar que se logre ingresar en el sistema los tipos de suelo existentes.	
Responsable: Ing. Juan León	Fecha: 20/10/2019
Descripción: Se revisa que se haya ingresado en el sistema los tipos de suelo existentes.	
Condiciones de ejecución: Debe de contar con todos los datos necesarios para realizar un calendario de riego.	
Pasos de ejecución:	
<ul style="list-style-type: none"> • Tomar los datos del campo • Realizar cálculos. • Realizar el reporte final. 	
Resultado Esperado: Ingreso en el sistema los tipos de suelo existentes.	
Evaluación de la prueba: Exitosa	

HISTORIA TÉCNICA	
Número: HT-13	Nombre: Documentar cada uno de las historias técnicas y actividades realizadas hasta la presente fecha en el proyecto.
Usuario: Desarrollador	Spring Asignado: 3
Puntos Estimados: 4	Puntos Reales: 4
Descripción: Como desarrollador se necesita documentar cada uno de las historias técnicas realizadas hasta la presente fecha.	
Pruebas de aceptación:	

- Verificar que exista la documentación de las historias realizadas.

PRUEBA DE ACEPTACIÓN	
Código: HT-13. PA-01	Nombre Historia: Documentar cada uno de las historias técnicas y actividades realizadas hasta la presente fecha en el proyecto.
Nombre prueba: Verificar que exista la documentación de las historias realizadas.	
Responsable: Ing. Alejandra Oñate	Fecha: 26/09/2019
Descripción: Se verifica que exista la documentación donde se evidencie la realización de las historias técnicas desarrolladas con anterioridad.	
Condiciones de ejecución: Tener el manual técnico del proyecto.	
Pasos de ejecución: <ul style="list-style-type: none"> • Observar que existan las tablas en donde se describe las historias técnicas del proyecto. 	
Resultado Esperado: Existe la documentación de las historias técnicas.	
Evaluación de la prueba: Exitosa	

Historias Técnicas Realizadas en el Sprint 5

HISTORIA USUARIO	
Número: HU-07	Nombre: Ingresar el sistema los métodos de cálculo para el Kc.
Usuario: Desarrollador	Spring Asignado: 5
Puntos Estimados: 16	Puntos Reales: 20
Descripción: Como desarrollador se necesita ingresar el sistema los métodos de cálculo para el Kc.	
Pruebas de aceptación: <ul style="list-style-type: none"> • Verificar que se haya ingresado el sistema los métodos de cálculo para el Kc. 	

PRUEBA DE ACEPTACIÓN	
Código: HU-07. PA-01	Nombre Historia: Ingresar el sistema los métodos de cálculo para el Kc.
Nombre prueba: Verificar que se haya comprendido todo el proceso para elaborar un calendario de riego.	
Responsable: Ing. Juan León	Fecha: 15/10/2019

Descripción: Se revisa y verifica que se haya comprendido todo el proceso de elaborar un calendario de riego por parte del desarrollador.
Condiciones de ejecución:
Pasos de ejecución: <ul style="list-style-type: none"> • Dirigirse el Centro Experimental del Riego. • Estar presente en la toma de datos necesarios para el cálculo del calendario de riego. • Estar presente en la elaboración del calendario del riego. • Anotar el paso a paso del proceso de elaboración.
Resultado Esperado: Ingreso en el sistema los métodos de cálculo para el Kc.
Evaluación de la prueba: Exitosa

TAREA DE INGENIERÍA	
Sprint: 5	Número de tarea: 01
Nombre historia: Ingresar el sistema los métodos de cálculo para el Kc.	
Nombre Tarea: Comprender el proceso de la elaboración del calendario de riego.	
Responsable: Isaac Torres	Tipo de Tarea: Desarrollo
Fecha Inicio: 15/10/2019	Fecha Fin: 20/10/2019
Descripción: Se debe ingresar el sistema los métodos de cálculo para el Kc.	
Pruebas de aceptación: <ul style="list-style-type: none"> • Verificar que se logre ingresar el sistema los métodos de cálculo para el Kc. 	

PRUEBA DE ACEPTACIÓN	
Código: HU-07.PA-02	Nombre Historia: Ingresar el sistema los métodos de cálculo para el Kc.
Nombre prueba: Verificar que se ingresar el sistema los métodos de cálculo para el Kc.	
Responsable: Ing. Juan León	Fecha: 20/10/2019
Descripción: Se revisa que el desarrollador haya ingresado el sistema los métodos de cálculo para el Kc.	
Condiciones de ejecución: Debe de contar con todos los datos necesarios para realizar un calendario de riego.	
Pasos de ejecución: <ul style="list-style-type: none"> • Tomar los datos del campo • Realizar cálculos. • Realizar el reporte final. 	

Resultado Esperado: Ingreso en el sistema los métodos de cálculo para el Kc.
Evaluación de la prueba: Exitosa

HISTORIA USUARIO	
Número: HU-08	Nombre: Ingresar en el sistema los tipos de superficie del tanque tipo A.
Usuario: Desarrollador	Spring Asignado: 5
Puntos Estimados: 16	Puntos Reales: 20
Descripción: Como desarrollador se necesita ingresar en el sistema los tipos de superficie del tanque tipo A.	
Pruebas de aceptación:	
<ul style="list-style-type: none"> • Verificar que se haya ingresado en el sistema los tipos de superficie del tanque tipo A. 	

PRUEBA DE ACEPTACIÓN	
Código: HU-08. PA-01	Nombre Historia: Ingresar en el sistema los tipos de superficie del tanque tipo A.
Nombre prueba: Verificar que se haya ingresado en el sistema los tipos de superficie del tanque tipo A.	
Responsable: Ing. Juan León	Fecha: 15/10/2019
Descripción: Se revisa y verifica que se haya ingresado en el sistema los tipos de superficie del tanque tipo A.	
Condiciones de ejecución:	
Pasos de ejecución:	
<ul style="list-style-type: none"> • Dirigirse el Centro Experimental del Riego. • Estar presente en la toma de datos necesarios para el cálculo del calendario de riego. • Estar presente en la elaboración del calendario del riego. • Anotar el paso a paso del proceso de elaboración. 	
Resultado Esperado: Ingreso en el sistema los tipos de superficie del tanque tipo A.	
Evaluación de la prueba: Exitosa	

TAREA DE INGENIERÍA	
Sprint: 5	Número de tarea: 01
Nombre historia Ingresar en el sistema los tipos de superficie del tanque tipo A.	

Nombre Tarea: Crear la tabla de los tipos de superficie del tanque tipo A.	
Responsable: Isaac Torres	Tipo de Tarea: Desarrollo
Fecha Inicio: 15/10/2019	Fecha Fin: 20/10/2019
Descripción: Se debe de ingresar en el sistema los tipos de superficie del tanque tipo A.	
Pruebas de aceptación:	
<ul style="list-style-type: none"> • Verificar que se logre ingresar en el sistema los tipos de superficie del tanque tipo A. 	

PRUEBA DE ACEPTACIÓN	
Código: HU-08.PA-02	Nombre Historia: Ingresar en el sistema los tipos de superficie del tanque tipo A.
Nombre prueba: Verificar que se logre ingresar en el sistema los tipos de superficie del tanque tipo A.	
Responsable: Ing. Juan León	Fecha: 20/10/2019
Descripción: Se revisa que se haya ingresado en el sistema los tipos de superficie del tanque tipo A.	
Condiciones de ejecución: Debe de contar con todos los datos necesarios para realizar un calendario de riego.	
Pasos de ejecución:	
<ul style="list-style-type: none"> • Tomar los datos del campo • Realizar cálculos. • Realizar el reporte final. 	
Resultado Esperado : Ingreso en el sistema los tipos de superficie del tanque tipo A.	
Evaluación de la prueba: Exitosa	

HISTORIA TÉCNICA	
Número: HT-13	Nombre: Documentar cada uno de las historias técnicas y actividades realizadas hasta la presente fecha en el proyecto.
Usuario: Desarrollador	Spring Asignado: 5
Puntos Estimados: 4	Puntos Reales: 4
Descripción: Como desarrollador se necesita documentar cada uno de las historias técnicas realizadas hasta la presente fecha.	
Pruebas de aceptación:	
<ul style="list-style-type: none"> • Verificar que exista la documentación de las historias realizadas. 	

PRUEBA DE ACEPTACIÓN	
Código: HT-13. PA-01	Nombre Historia: Documentar cada uno de las historias técnicas y actividades realizadas hasta la presente fecha en el proyecto.
Nombre prueba: Verificar que exista la documentación de las historias realizadas.	
Responsable: Ing. Alejandra Oñate	Fecha: 26/09/2019
Descripción: Se verifica que exista la documentación donde se evidencie la realización de las historias técnicas desarrolladas con anterioridad.	
Condiciones de ejecución: Tener el manual técnico del proyecto.	
Pasos de ejecución:	
<ul style="list-style-type: none"> • Observar que existan las tablas en donde se describe las historias técnicas del proyecto. 	
Resultado Esperado: Existe la documentación de las historias técnicas.	
Evaluación de la prueba: Exitosa	

Historias Técnicas Realizadas en el Sprint 6

HISTORIA USUARIO	
Número: HU-09	Nombre: Calcular la evapotranspiración ETo.
Usuario: Desarrollador	Spring Asignado: 6
Puntos Estimados: 16	Puntos Reales: 20
Descripción: Como desarrollador se necesita calcular la evapotranspiración ETo.	
Pruebas de aceptación:	
<ul style="list-style-type: none"> • Verificar que se haya calcular la evapotranspiración ETo. 	

PRUEBA DE ACEPTACIÓN	
Código: HU-09. PA-01	Nombre Historia: Calcular la evapotranspiración ETo
Nombre prueba: Verificar que se hay comprendido todo el proceso para elaborar un calendario de riego.	
Responsable: Ing. Juan León	Fecha: 15/10/2019
Descripción: Se revisa y verifica que se haya calculado la evapotranspiración ETo.	
Condiciones de ejecución:	
Pasos de ejecución:	
<ul style="list-style-type: none"> • Dirigirse el Centro Experimental del Riego. • Estar presente en la toma de datos necesarios para el cálculo del calendario de riego. 	

<ul style="list-style-type: none"> • Estar presente en la elaboración del calendario del riego. • Anotar el paso a paso del proceso de elaboración.
Resultado Esperado: Calcular la evapotranspiración ETo.
Evaluación de la prueba: Exitosa

TAREA DE INGENIERÍA	
Sprint: 6	Número de tarea: 01
Nombre historia Calcular la evapotranspiración ETo	
Nombre Tarea: Calcular la evapotranspiración ETo.	
Responsable: Isaac Torres	Tipo de Tarea: Desarrollo
Fecha Inicio: 15/10/2019	Fecha Fin: 20/10/2019
Descripción: Se debe de calcular la evapotranspiración ETo.	
Pruebas de aceptación:	
<ul style="list-style-type: none"> • Verificar que se logre calcular la evapotranspiración ETo. 	

PRUEBA DE ACEPTACIÓN	
Código: HT-01.PA-02	Nombre Historia: Calcular la evapotranspiración ETo.
Nombre prueba: Verificar que se logre calcular la evapotranspiración ETo.	
Responsable: Ing. Juan León	Fecha: 20/10/2019
Descripción: Se revisa que el desarrollador sea capaz de calcular la evapotranspiración ETo.	
Condiciones de ejecución: Debe de contar con todos los datos necesarios para realizar un calendario de riego.	
Pasos de ejecución:	
<ul style="list-style-type: none"> • Tomar los datos del campo • Realizar cálculos. • Realizar el reporte final. 	
Resultado Esperado: Calcular la evapotranspiración ETo.	
Evaluación de la prueba: Exitosa	

HISTORIA USUARIO	
Número: HU-10	Nombre: Calcular coeficiente de cultivo Kc
Usuario: Desarrollador	Spring Asignado: 6
Puntos Estimados: 16	Puntos Reales: 20

Descripción: Como desarrollador se necesita calcular coeficiente de cultivo Kc
Pruebas de aceptación: <ul style="list-style-type: none"> • Verificar que se haya calculado coeficiente de cultivo Kc

PRUEBA DE ACEPTACIÓN	
Código: HU-10. PA-01	Nombre Historia: Calcular coeficiente de cultivo Kc
Nombre prueba: Verificar que se hay comprendido todo el proceso para elaborar un calendario de riego.	
Responsable: Ing. Juan León	Fecha: 15/10/2019
Descripción: Se revisa y verifica que se haya calculado coeficiente de cultivo Kc	
Condiciones de ejecución:	
Pasos de ejecución: <ul style="list-style-type: none"> • Dirigirse el Centro Experimental del Riego. • Estar presente en la toma de datos necesarios para el cálculo del calendario de riego. • Estar presente en la elaboración del calendario del riego. • Anotar el paso a paso del proceso de elaboración. 	
Resultado Esperado: Cálculo coeficiente de cultivo Kc	
Evaluación de la prueba: Exitosa	

TAREA DE INGENIERÍA	
Sprint: 6	Número de tarea: 01
Nombre historia: Calcular coeficiente de cultivo Kc	
Nombre Tarea: Calcular coeficiente de cultivo Kc	
Responsable: Isaac Torres	Tipo de Tarea: Desarrollo
Fecha Inicio: 15/10/2019	Fecha Fin: 20/10/2019
Descripción: Se debe de calcular coeficiente de cultivo Kc	
Pruebas de aceptación: <ul style="list-style-type: none"> • Verificar que se logre calcular coeficiente de cultivo Kc. 	

PRUEBA DE ACEPTACIÓN	
Código: HU-10.PA-02	Nombre Historia: Calcular coeficiente de cultivo Kc
Nombre prueba: Verificar que se logre calcular coeficiente de cultivo Kc	

Responsable: Ing. Juan León	Fecha: 20/10/2019
Descripción: Se revisa que el desarrollador sea capaz calcular coeficiente de cultivo Kc	
Condiciones de ejecución: Debe de contar con todos los datos necesarios para realizar un calendario de riego.	
Pasos de ejecución: <ul style="list-style-type: none"> • Tomar los datos del campo • Realizar cálculos. • Realizar el reporte final. 	
Resultado Esperado: Calcular coeficiente de cultivo Kc	
Evaluación de la prueba: Exitosa	

HISTORIA USUARIO	
Número: HU-11	Nombre: Calcular evapotranspiración del cultivo ETc.
Usuario: Desarrollador	Spring Asignado: 6
Puntos Estimados: 16	Puntos Reales: 20
Descripción: Como desarrollador se necesita calcular evapotranspiración del cultivo ETc.	
Pruebas de aceptación: <ul style="list-style-type: none"> • Verificar que se haya calcular evapotranspiración del cultivo ETc. 	

PRUEBA DE ACEPTACIÓN	
Código: HU-11. PA-01	Nombre Historia: Calcular evapotranspiración del cultivo ETc
Nombre prueba: Verificar que se haya calculado evapotranspiración del cultivo ETc.	
Responsable: Ing. Juan León	Fecha: 15/10/2019
Descripción: Se revisa y verifica que se haya calculado evapotranspiración del cultivo ETc.	
Condiciones de ejecución:	
Pasos de ejecución: <ul style="list-style-type: none"> • Dirigirse el Centro Experimental del Riego. • Estar presente en la toma de datos necesarios para el cálculo del calendario de riego. • Estar presente en la elaboración del calendario del riego. • Anotar el paso a paso del proceso de elaboración. 	
Resultado Esperado: Comprensión del proceso de elaboración de un calendario de riego.	
Evaluación de la prueba: Exitosa	

TAREA DE INGENIERÍA	
Sprint: 6	Número de tarea: 01
Nombre historia: Calcular evapotranspiración del cultivo ETc.	
Nombre Tarea: Comprender el proceso para calcular evapotranspiración del cultivo ETc.	
Responsable: Isaac Torres	Tipo de Tarea: Desarrollo
Fecha Inicio: 15/10/2019	Fecha Fin: 20/10/2019
Descripción: Se debe de comprender como realizar calcular la evapotranspiración del cultivo ETc.	
Pruebas de aceptación:	
<ul style="list-style-type: none"> • Verificar que se logre calcular evapotranspiración del cultivo ETc. 	

PRUEBA DE ACEPTACIÓN	
Código: HT-01.PA-02	Nombre Historia: Calcular evapotranspiración del cultivo ETc.
Nombre prueba: Verificar que se calcular evapotranspiración del cultivo ETc.	
Responsable: Ing. Juan León	Fecha: 20/10/2019
Descripción: Se revisa que el desarrollador sea capaz de calcular evapotranspiración del cultivo ETc.	
Condiciones de ejecución: Debe de contar con todos los datos necesarios para realizar un calendario de riego.	
Pasos de ejecución:	
<ul style="list-style-type: none"> • Tomar los datos del campo • Realizar cálculos. • Realizar el reporte final. 	
Resultado Esperado Cálculo evapotranspiración del cultivo ETc.	
Evaluación de la prueba: Exitosa	

HISTORIA USUARIO	
Número: HU-12	Nombre: Calcular coeficiente de varianza Kp.
Usuario: Desarrollador	Spring Asignado: 6
Puntos Estimados: 16	Puntos Reales: 20
Descripción: Como desarrollador se necesita calcular coeficiente de varianza Kp.	
Pruebas de aceptación:	

- Verificar que se haya calculado coeficiente de varianza Kp.

PRUEBA DE ACEPTACIÓN	
Código: HU-12. PA-01	Nombre Historia: Calcular coeficiente de varianza Kp.
Nombre prueba: Verificar que se haya calculado el coeficiente de varianza Kp.	
Responsable: Ing. Juan León	Fecha: 15/10/2019
Descripción: Se revisa y verifica que se haya calculado el coeficiente de varianza Kp.	
Condiciones de ejecución:	
Pasos de ejecución:	
<ul style="list-style-type: none"> • Dirigirse el Centro Experimental del Riego. • Estar presente en la toma de datos necesarios para el cálculo del calendario de riego. • Estar presente en la elaboración del calendario del riego. • Anotar el paso a paso del proceso de elaboración. 	
Resultado Esperado: Comprensión del proceso de calculo del coeficiente de varianza Kp.	
Evaluación de la prueba: Exitosa	

TAREA DE INGENIERÍA	
Sprint: 6	Número de tarea: 01
Nombre historia Calcular coeficiente de varianza Kp.	
Nombre Tarea: Comprender el proceso de calculo coeficiente de varianza Kp.	
Responsable: Isaac Torres	Tipo de Tarea: Desarrollo
Fecha Inicio: 15/10/2019	Fecha Fin: 20/10/2019
Descripción: Se debe de comprender como realizar el cálculo coeficiente de varianza Kp.	
Pruebas de aceptación:	
<ul style="list-style-type: none"> • Verificar que se logre calcular coeficiente de varianza Kp. 	

PRUEBA DE ACEPTACIÓN	
Código: HU-12.PA-02	Nombre Historia: Calcular coeficiente de varianza Kp.
Nombre prueba: Verificar que se logre calcular coeficiente de varianza Kp.	
Responsable: Ing. Juan León	Fecha: 20/10/2019
Descripción: Se revisa que el desarrollador sea capaz de realizar el cálculo del coeficiente de varianza Kp.	

Condiciones de ejecución: Debe de contar con todos los datos necesarios para realizar un calendario de riego.
Pasos de ejecución: <ul style="list-style-type: none"> • Tomar los datos del campo • Realizar cálculos. • Realizar el reporte final.
Resultado Esperado: Calcular coeficiente de varianza Kp.
Evaluación de la prueba: Exitosa

HISTORIA USUARIO	
Número: HU-13	Nombre: Calcular constante de tiempo de calendario C
Usuario: Desarrollador	Spring Asignado: 6
Puntos Estimados: 16	Puntos Reales: 20
Descripción: Como desarrollador se necesita Calcular la constante de tiempo de calendario C.	
Pruebas de aceptación: <ul style="list-style-type: none"> • Verificar que se haya calculado la constante de tiempo de calendario C 	

PRUEBA DE ACEPTACIÓN	
Código: HU-13. PA-01	Nombre Historia: Calcular constante de tiempo de calendario C.
Nombre prueba: Verificar que se haya calculado la constante de tiempo de calendario C	
Responsable: Ing. Juan León	Fecha: 15/10/2019
Descripción: Se revisa y verifica que se haya calculado constante de tiempo de calendario C	
Condiciones de ejecución:	
Pasos de ejecución: <ul style="list-style-type: none"> • Dirigirse el Centro Experimental del Riego. • Estar presente en la toma de datos necesarios para el cálculo del calendario de riego. • Estar presente en la elaboración del calendario del riego. • Anotar el paso a paso del proceso de elaboración. 	
Resultado Esperado: Comprensión del proceso de cálculo constante de tiempo de calendario C	
Evaluación de la prueba: Exitosa	

TAREA DE INGENIERÍA	
Sprint: 6	Número de tarea: 01
Nombre historia: Calcular constante de tiempo de calendario C	
Nombre Tarea: Comprender el proceso de cálculo constante de tiempo de calendario C.	
Responsable: Isaac Torres	Tipo de Tarea: Desarrollo
Fecha Inicio: 15/10/2019	Fecha Fin: 20/10/2019
Descripción: Se debe de comprender como realizar el cálculo de la constante de tiempo de calendario C	
Pruebas de aceptación:	
<ul style="list-style-type: none"> • Verificar que se logre calcular constante de tiempo de calendario C. 	

PRUEBA DE ACEPTACIÓN	
Código: HU-13.PA-02	Nombre Historia: Calcular constante de tiempo de calendario C
Nombre prueba: Verificar que se logre calcular constante de tiempo de calendario C	
Responsable: Ing. Juan León	Fecha: 20/10/2019
Descripción: Se revisa que el desarrollador sea capaz de calcular constante de tiempo de calendario C	
Condiciones de ejecución: Debe de contar con todos los datos necesarios para realizar un calendario de riego.	
Pasos de ejecución:	
<ul style="list-style-type: none"> • Tomar los datos del campo • Realizar cálculos. • Realizar el reporte final. 	
Resultado Esperado: Calcular constante de tiempo de calendario C.	
Evaluación de la prueba: Exitosa	

HISTORIA USUARIO	
Número: HU-14	Nombre: Calcular lámina neta de riego Ln
Usuario: Desarrollador	Spring Asignado: 6
Puntos Estimados: 16	Puntos Reales: 20
Descripción: Como desarrollador se necesita realizar el procedimiento almacenado para calcular la lámina neta de riego Ln	
Pruebas de aceptación:	

- Verificar que se haya desarrollado el procedimiento almacenado que sirve para Calcular lámina neta de riego Ln.

PRUEBA DE ACEPTACIÓN	
Código: HU-14. PA-01	Nombre Historia Calcular lámina neta de riego Ln.
Nombre prueba: Verificar que se haya desarrollado el procedimiento almacenado para Calcular lámina neta de riego Ln.	
Responsable: Ing. Juan León	Fecha: 15/10/2019
Descripción: Se revisa y verifica que se haya calculado lámina neta de riego Ln.	
Condiciones de ejecución:	
Pasos de ejecución:	
<ul style="list-style-type: none"> • Dirigirse el Centro Experimental del Riego. • Estar presente en la toma de datos necesarios para el cálculo del calendario de riego. • Estar presente en la elaboración del calendario del riego. • Anotar el paso a paso del proceso de elaboración. 	
Resultado Esperado: Desarrollo del procedimiento almacenado que permite calcular la lámina neta de riego Ln.	
Evaluación de la prueba: Exitosa	

TAREA DE INGENIERÍA	
Sprint: 6	Número de tarea: 01
Nombre historia: Calcular lámina neta de riego Ln	
Nombre Tarea: Desarrollar el procedimiento almacenado para calcular lámina neta de riego Ln.	
Responsable: Isaac Torres	Tipo de Tarea: Desarrollo
Fecha Inicio: 15/10/2019	Fecha Fin: 20/10/2019
Descripción: Se debe de desarrollar el procedimiento almacenado para calcular lámina neta de riego Ln.	
Pruebas de aceptación:	
<ul style="list-style-type: none"> • Verificar que se logre calcular lámina neta de riego Ln, por medio de un procedimiento almacenado. 	

PRUEBA DE ACEPTACIÓN

Código: HU-14.PA-02	Nombre Historia: Calcular lámina neta de riego Ln
Nombre prueba: Verificar que se logre calcular lámina neta de riego Ln	
Responsable: Ing. Juan León	Fecha: 20/10/2019
Descripción: Se revisa que el desarrollador haya creado un procedimiento almacenado que permita calcular lámina neta de riego Ln.	
Condiciones de ejecución: Debe de contar con todos los datos necesarios para realizar un calendario de riego.	
Pasos de ejecución: <ul style="list-style-type: none"> • Tomar los datos del campo • Realizar cálculos. • Realizar el reporte final. 	
Resultado Esperado: Procedimiento almacenado para calcular lámina neta de riego Ln desarrollado.	
Evaluación de la prueba: Exitosa	

HISTORIA USUARIO	
Número: HU-15	Nombre: Calcular la cantidad de agua útil Au.
Usuario: Desarrollador	Spring Asignado: 6
Puntos Estimados: 16	Puntos Reales: 20
Descripción: Como desarrollador se necesita investigar en que consiste el proceso desde el inicio hasta el fin de la elaboración de un calendario de riego y las herramientas que se utilizan para elaborar el mismo.	
Pruebas de aceptación: <ul style="list-style-type: none"> • Verificar que se hay comprendido todo el proceso para elaborar un calendario de riego. 	

PRUEBA DE ACEPTACIÓN	
Código: HU-15. PA-01	Nombre Historia: Calcular la cantidad de agua útil Au.
Nombre prueba: Verificar que se haya calculado la cantidad de agua útil Au	
Responsable: Ing. Juan León	Fecha: 15/10/2019
Descripción: Se revisa y verifica que se haya calculado la cantidad de agua útil Au.	
Condiciones de ejecución:	

Pasos de ejecución:
<ul style="list-style-type: none"> • Dirigirse el Centro Experimental del Riego. • Estar presente en la toma de datos necesarios para el cálculo del calendario de riego. • Estar presente en la elaboración del calendario del riego. • Anotar el paso a paso del proceso de elaboración.
Resultado Esperado: Comprensión del proceso calculo la cantidad de agua útil Au.
Evaluación de la prueba: Exitosa

TAREA DE INGENIERÍA	
Sprint: 6	Número de tarea: 01
Nombre historia: Calcular la cantidad de agua útil Au.	
Nombre Tarea: Comprender el proceso para calcular la cantidad de agua útil Au.	
Responsable: Isaac Torres	Tipo de Tarea: Desarrollo
Fecha Inicio: 15/10/2019	Fecha Fin: 20/10/2019
Descripción: Se debe de comprender como realizar el cálculo de la cantidad de agua útil Au.	
Pruebas de aceptación:	
<ul style="list-style-type: none"> • Verificar que se logre calcular la cantidad de agua útil Au. 	

PRUEBA DE ACEPTACIÓN	
Código: HU-15.PA-02	Nombre Historia: Calcular la cantidad de agua útil Au
Nombre prueba: Verificar que se logre calcular la cantidad de agua útil Au.	
Responsable: Ing. Juan León	Fecha: 20/10/2019
Descripción: Se revisa que el desarrollador sea capaz de calcular la cantidad de agua útil Au	
Condiciones de ejecución: Debe de contar con todos los datos necesarios para realizar un calendario de riego.	
Pasos de ejecución:	
<ul style="list-style-type: none"> • Tomar los datos del campo • Realizar cálculos. • Realizar el reporte final. 	
Resultado Esperado: Calcular la cantidad de agua útil Au	
Evaluación de la prueba: Exitosa	

HISTORIA USUARIO	
Número: HU-16	Nombre Calcular lamina bruta de riego Lb.
Usuario: Desarrollador	Spring Asignado: 6
Puntos Estimados: 16	Puntos Reales: 20
Descripción: Como desarrollador se necesita calcular lamina bruta de riego Lb.	
Pruebas de aceptación:	
<ul style="list-style-type: none"> • Verificar que se haya Calcular lamina bruta de riego Lb. 	

PRUEBA DE ACEPTACIÓN	
Código: HU-16. PA-01	Nombre Historia: Calcular lamina bruta de riego Lb.
Nombre prueba: Verificar que se hay comprendido todo el proceso para calcular lamina bruta de riego Lb.	
Responsable: Ing. Juan León	Fecha: 15/10/2019
Descripción: Se revisa y verifica que se haya calculado la lámina bruta de riego Lb.	
Condiciones de ejecución:	
Pasos de ejecución:	
<ul style="list-style-type: none"> • Dirigirse el Centro Experimental del Riego. • Estar presente en la toma de datos necesarios para el cálculo del calendario de riego. • Estar presente en la elaboración del calendario del riego. • Anotar el paso a paso del proceso de elaboración. 	
Resultado Esperado: Comprensión del proceso para ccalcular la lámina bruta de riego Lb.elaboración de un calendario de riego.	
Evaluación de la prueba: Exitosa	

TAREA DE INGENIERÍA	
Sprint: 6	Número de tarea: 01
Nombre historia: Calcular lamina bruta de riego Lb.	
Nombre Tarea: Comprender el proceso para calcular lamina bruta de riego Lb.	
Responsable: Isaac Torres	Tipo de Tarea: Desarrollo
Fecha Inicio: 15/10/2019	Fecha Fin: 20/10/2019
Descripción: Se debe de comprender como realizar el cálculo de la lámina bruta de riego Lb.	
Pruebas de aceptación:	
<ul style="list-style-type: none"> • Verificar que se logre calcular lamina bruta de riego Lb. 	

PRUEBA DE ACEPTACIÓN	
Código: HU-16.PA-02	Nombre Historia: Calcular lamina bruta de riego Lb
Nombre prueba: Verificar que se calcular lamina bruta de riego Lb	
Responsable: Ing. Juan León	Fecha: 20/10/2019
Descripción: Se revisa que el desarrollador sea capaz de calcular lamina bruta de riego Lb	
Condiciones de ejecución: Debe de contar con todos los datos necesarios para realizar un calendario de riego.	
Pasos de ejecución:	
<ul style="list-style-type: none"> • Tomar los datos del campo • Realizar cálculos. • Realizar el reporte final. 	
Resultado Esperado: Calcular lamina bruta de riego Lb	
Evaluación de la prueba: Exitosa	

Historias Técnicas Realizadas en el Sprint 7

HISTORIA USUARIO	
Número: HU-17	Nombre: Calcular volumen de agua de riego Vr.
Usuario: Desarrollador	Spring Asignado: 6
Puntos Estimados: 16	Puntos Reales: 20
Descripción: Como desarrollador se necesita calcular volumen de agua de riego Vr.	
Pruebas de aceptación:	
<ul style="list-style-type: none"> • Verificar que se haya calculado volumen de agua de riego Vr. 	

PRUEBA DE ACEPTACIÓN	
Código: HU-17. PA-01	Nombre Historia Calcular volumen de agua de riego Vr
Nombre prueba: Verificar que se hay comprendido todo el proceso para Calcular volumen de agua de riego Vr.	
Responsable: Ing. Juan León	Fecha: 15/10/2019
Descripción: Se revisa y verifica que se haya comprendido todo el proceso para calcular volumen de agua de riego Vr.	
Condiciones de ejecución:	

Pasos de ejecución:
<ul style="list-style-type: none"> • Dirigirse el Centro Experimental del Riego. • Estar presente en la toma de datos necesarios para el cálculo del calendario de riego. • Estar presente en la elaboración del calendario del riego. • Anotar el paso a paso del proceso de elaboración.
Resultado Esperado: Calcular volumen de agua de riego Vr.
Evaluación de la prueba: Exitosa

TAREA DE INGENIERÍA	
Sprint: 7	Número de tarea: 01
Nombre historia Calcular volumen de agua de riego Vr.	
Nombre Tarea: Comprender el proceso de la elaboración del cálculo del volumen de agua de riego Vr.	
Responsable: Isaac Torres	Tipo de Tarea: Desarrollo
Fecha Inicio: 15/10/2019	Fecha Fin: 20/10/2019
Descripción: Se debe de comprender como realizar el cálculo volumen de agua de riego Vr.	
Pruebas de aceptación:	
<ul style="list-style-type: none"> • Verificar que se logre presentar ccálculo del volumen de agua de riego Vr. 	

PRUEBA DE ACEPTACIÓN	
Código: HU-17.PA-02	Nombre Historia Calcular volumen de agua de riego Vr.
Nombre prueba: Verificar que se logre presentar Calcular volumen de agua de riego Vr.	
Responsable: Ing. Juan León	Fecha: 20/10/2019
Descripción: Se revisa que el desarrollador sea capaz de realizar el cálculo del volumen de agua de riego Vr.	
Condiciones de ejecución: Debe de contar con todos los datos necesarios para realizar un calendario de riego.	
Pasos de ejecución:	
<ul style="list-style-type: none"> • Tomar los datos del campo • Realizar cálculos. • Realizar el reporte final. 	
Resultado Esperado: Calcular volumen de agua de riego Vr.	
Evaluación de la prueba: Exitosa	

HISTORIA USUARIO	
Número: HU-18	Nombre: Calcular tiempo de riego Tr
Usuario: Desarrollador	Spring Asignado: 7
Puntos Estimados: 16	Puntos Reales: 20
Descripción: Como desarrollador se necesita calcular el tiempo de riego Tr	
Pruebas de aceptación:	
<ul style="list-style-type: none"> • Verificar que se haya logrado calcular tiempo de riego Tr 	

PRUEBA DE ACEPTACIÓN	
Código: HU-18. PA-01	Nombre Historia: Calcular tiempo de riego Tr.
Nombre prueba: Verificar que se haya comprendido todo el proceso para calcular tiempo de riego Tr	
Responsable: Ing. Juan León	Fecha: 15/10/2019
Descripción: Se revisa y verifica que se haya comprendido todo el proceso de calcular tiempo de riego Tr.	
Condiciones de ejecución:	
Pasos de ejecución:	
<ul style="list-style-type: none"> • Dirigirse el Centro Experimental del Riego. • Estar presente en la toma de datos necesarios para el cálculo del calendario de riego. • Estar presente en la elaboración del calendario del riego. • Anotar el paso a paso del proceso de elaboración. 	
Resultado Esperado: Calcular tiempo de riego Tr	
Evaluación de la prueba: Exitosa	

TAREA DE INGENIERÍA	
Sprint: 7	Número de tarea: 01
Nombre historia: Calcular tiempo de riego Tr.	
Nombre Tarea: Comprender el proceso para calcular tiempo de riego Tr	
Responsable: Isaac Torres	Tipo de Tarea: Desarrollo
Fecha Inicio: 15/10/2019	Fecha Fin: 20/10/2019
Descripción: Se debe de comprender como realizar el ccálculo del tiempo de riego Tr	
Pruebas de aceptación:	
<ul style="list-style-type: none"> • Verificar que se logre Calcular tiempo de riego Tr. 	

PRUEBA DE ACEPTACIÓN	
Código: HU-18.PA-02	Nombre Historia: Calcular tiempo de riego Tr
Nombre prueba: Verificar que se logre calcular tiempo de riego Tr	
Responsable: Ing. Juan León	Fecha: 20/10/2019
Descripción: Se revisa que el desarrollador sea capaz de realizar el cálculo del tiempo de riego Tr	
Condiciones de ejecución: Debe de contar con todos los datos necesarios para realizar un calendario de riego.	
Pasos de ejecución: <ul style="list-style-type: none"> • Tomar los datos del campo • Realizar cálculos. • Realizar el reporte final. 	
Resultado Esperado: Calcular tiempo de riego Tr	
Evaluación de la prueba: Exitosa	

HISTORIA USUARIO	
Número: HU-19	Listar todas las variables (calendario de riego)
Usuario: Desarrollador	Spring Asignado: 7
Puntos Estimados: 16	Puntos Reales: 20
Descripción: Como desarrollador se necesita listar todas las variables anteriormente calculadas, agrupándolas en una tabla creando de esta manera del calendario de riego.	
Pruebas de aceptación: <ul style="list-style-type: none"> • Verificar que se enlisten todas las variables anteriormente calculadas para la creación de la tabla del calendario del riego. 	

PRUEBA DE ACEPTACIÓN	
Código: HU-19. PA-01	Listar todas las variables (calendario de riego)
Nombre prueba: Verificar que se hayan calculado todas las variables necesarias para crear el calendario del riego.	
Responsable: Ing. Juan León	Fecha: 15/10/2019

Descripción: Se revisa y verifica que se haya calculado el agua útil, lámina bruta, lámina neta, tiempo de riego, etc. Variables necesarias para formar la tabla del calendario de riego.
Condiciones de ejecución:
Pasos de ejecución: <ul style="list-style-type: none"> • Abrir el SGBD. • Ingresar los datos para ejecutar el procedimiento almacenado. • Ejecutar el procedimiento almacenado del calendario. • Visualizar todas las variables necesarias para formar el calendario de riego.
Resultado Esperado: Visualizar todas las variables necesarias para calcular el calendario de riego.
Evaluación de la prueba: Exitosa

TAREA DE INGENIERÍA	
Sprint: 7	Número de tarea: 01
Nombre historia: Listar todas las variables (calendario de riego)	
Nombre Tarea: Crear procedimiento almacenado para unir todas las variables del calendario de riego.	
Responsable: Isaac Torres	Tipo de Tarea: Desarrollo
Fecha Inicio: 15/10/2019	Fecha Fin: 20/10/2019
Descripción: Se debe desarrollar el procedimiento almacenado que agrupe a todas las variables que forman el calendario de riego.	
Pruebas de aceptación: <ul style="list-style-type: none"> • Verificar que haya creado el procedimiento almacenado que agrupe a todas las variables que conforman el calendario de riego. 	

PRUEBA DE ACEPTACIÓN	
Código: HU-19.PA-02	Nombre Historia: Listar todas las variables (calendario de riego)
Nombre prueba: Crear procedimiento almacenado para unir todas las variables del	
Responsable: Ing. Juan León	Fecha: 20/10/2019
Descripción: Se revisa que el desarrollador haya creado el procedimiento almacenado que permite unir a todas las variables anteriormente calculadas para crear el calendario de riego.	
Condiciones de ejecución: Deben de estar calculadas todas las variables que comprenden el calendario de riego.	
Pasos de ejecución:	

<ul style="list-style-type: none"> • Abrir el SGBD. • Escribir el Script del procedimiento almacenado. • Ejecutar el procedimiento almacenado.
Resultado Esperado: Calendario de riego elaborado.
Evaluación de la prueba: Exitosa

HISTORIA USUARIO	
Número: HU-20	Nombre: Generar calendario en formato PDF.
Usuario: Desarrollador	Spring Asignado: 7
Puntos Estimados: 16	Puntos Reales: 20
Descripción: Como desarrollador se necesita crear la función que permita crear en formato PDF la tabla del calendario de riego.	
Pruebas de aceptación:	
<ul style="list-style-type: none"> • Verificar que se haya creado el archivo en formato PDF del calendario de riego previamente calculado. 	

PRUEBA DE ACEPTACIÓN	
Código: HU-20. PA-01	Nombre Historia: Generar calendario en formato PDF.
Nombre prueba: Verificar que se haya creado el archivo en formato PDF del calendario de riego previamente calculado.	
Responsable: Ing. Juan León	Fecha: 15/10/2019
Descripción: Se revisa y verifica que se haya creado descargado y que se pueda abrir el archivo del calendario de riego.	
Condiciones de ejecución: Debe de estar calculado el calendario de riego	
Pasos de ejecución:	
<ul style="list-style-type: none"> • Dirigirse a www.cer.esPOCH.edu.ec • Ingresar en calendario de riego. • Seleccionar calendario • Ingresar todos los datos en el formulario • .Seleccionar en generar calendario. • Seleccionar en el botón que indica PDF. 	
Resultado Esperado: Descarga del calendario generado en formato PDF.	

Evaluación de la prueba: Exitosa

TAREA DE INGENIERÍA	
Sprint: 7	Número de tarea: 01
Nombre historia: Generar calendario en formato pdf.	
Nombre Tarea: Verificar que se descargue y se abra el archivo.	
Responsable: Isaac Torres	Tipo de Tarea: Desarrollo
Fecha Inicio: 15/10/2019	Fecha Fin: 20/10/2019
Descripción: Se debe de verificar que el archivo del calendario de riego se haya descargado y que se lo pueda abrir en visualizador de archivos PDF.	
Pruebas de aceptación:	
<ul style="list-style-type: none"> • Verificar que se logre presentar y abrir el reporte de un calendario de riego elaborado. 	

PRUEBA DE ACEPTACIÓN	
Código: HU-20.PA-02	Nombre Historia: Generar calendario en formato PDF.
Nombre prueba: Verificar que se logre presentar y abrir el reporte de un calendario de riego elaborado.	
Responsable: Ing. Juan León	Fecha: 20/10/2019
Descripción: Se revisa que el archivo generado se haya descargado y que se pueda abrir en un visualizador de archivos PDF.	
Condiciones de ejecución: Debe de estar calculado el calendario de riego	
Pasos de ejecución:	
<ul style="list-style-type: none"> • Seleccionar el botón de generar PDF. • Ir al sitio de descargas. • Abrir el archivo. • Visualizar el calendario de riego en un visualizador de archivos PDF. 	
Resultado Esperado: Calendario de riego elaborado en formato PDF.	
Evaluación de la prueba: Exitosa	

HISTORIA USUARIO	
Número: HU-21	Capacitar a los usuarios en el manejo de la aplicación web para su adecuado funcionamiento.

Usuario: Desarrollador	Spring Asignado: 7
Puntos Estimados: 16	Puntos Reales: 20
Descripción: Se necesita enseñar a los usuarios el manejo de la aplicación web.	
Pruebas de aceptación:	
<ul style="list-style-type: none"> • Verificar que los usuarios puedan manejar la aplicación web. 	

PRUEBA DE ACEPTACIÓN	
Código: HT-21. PA-01	Capacitar a los usuarios en el manejo del sistema informático para su adecuado funcionamiento.
Nombre prueba: Verificar que los usuarios puedan manejar la aplicación web.	
Responsable: Ing. Juan León	Fecha: 15/10/2019
Descripción: Se revisa y verifica que se haya comprendido el manejo de la aplicación web.	
Condiciones de ejecución: Aplicación montada en un servidor web.	
Pasos de ejecución:	
<ul style="list-style-type: none"> • Dirigirse a www.cer.esPOCH.edu.ec • Abrir la aplicación web. • Indicar los campos del formulario de ingreso. • Elaborar un calendario de riego demostrativo para su mejor comprensión. . 	
Resultado Esperado: Comprensión del proceso de elaboración de un calendario de riego.	
Evaluación de la prueba: Exitosa	

TAREA DE INGENIERÍA	
Sprint: 7	Número de tarea: 01
Nombre Tarea: Verificar que se haya comprendido el manejo de la aplicación web.	
Responsable: Isaac Torres	Tipo de Tarea: Desarrollo
Fecha Inicio: 15/10/2019	Fecha Fin: 20/10/2019
Descripción: Se debe de comprobar que el usuario haya comprendido el manejo de la aplicación web, solicitando que desarrolle un ejemplo desde el principio hasta el final.	
Pruebas de aceptación:	
<ul style="list-style-type: none"> • Verificar que se logre presentar el reporte de un calendario de riego elaborado. 	

PRUEBA DE ACEPTACIÓN	
Código: HU-21.PA-02	Capacitar a los usuarios en el manejo del sistema informático para su adecuado funcionamiento.
Nombre prueba: Verificar que se logre presentar el reporte de un calendario de riego elaborado.	
Responsable: Ing. Juan León	Fecha: 20/10/2019
Descripción: Se revisa que el usuario sea capaz de generar un calendario de riego desde el inicio hasta el final lograr abrir el archivo PDF.	
Condiciones de ejecución: Debe de haber recibido la inducción del uso de la aplicación.	
Pasos de ejecución: <ul style="list-style-type: none"> • Capacitar al usuario. • Solicitar que realice un ejemplo de cálculo de calendario de riego. • Visualizar el archivo PDF final. 	
Resultado Esperado: Calendario de riego Realizado por un usuario.	
Evaluación de la prueba: Exitosa	

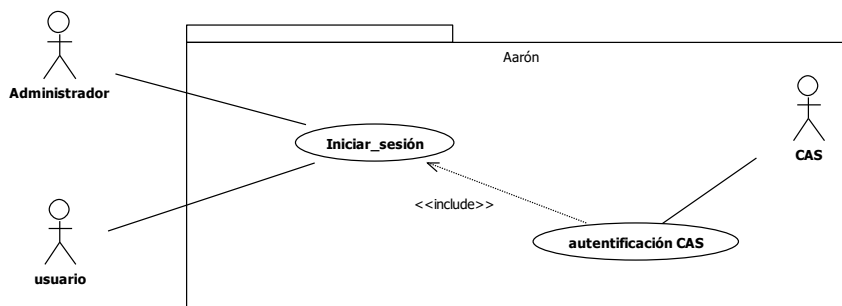
HISTORIA TÉCNICA	
Número: HT-13	Nombre: Documentar cada uno de las historias técnicas y actividades realizadas hasta la presente fecha en el proyecto.
Usuario: Desarrollador	Spring Asignado: 7
Puntos Estimados: 4	Puntos Reales: 4
Descripción: Como desarrollador se necesita documentar cada uno de las historias técnicas realizadas hasta la presente fecha.	
Pruebas de aceptación: <ul style="list-style-type: none"> • Verificar que exista la documentación de las historias realizadas. 	

PRUEBA DE ACEPTACIÓN	
Código: HT-13. PA-01	Nombre Historia: Documentar cada uno de las historias técnicas y actividades realizadas hasta la presente fecha en el proyecto.
Nombre prueba: Verificar que exista la documentación de las historias realizadas.	
Responsable: Ing. Alejandra Oñate	Fecha: 26/09/2019

Descripción: Se verifica que exista la documentación donde se evidencie la realización de las historias técnicas desarrolladas con anterioridad.
Condiciones de ejecución: Tener el manual técnico del proyecto.
Pasos de ejecución: <ul style="list-style-type: none"> • Observar que existan las tablas en donde se describe las historias técnicas del proyecto.
Resultado Esperado: Existe la documentación de las historias técnicas.
Evaluación de la prueba: Exitosa

ANEXO B **Diagramas** de casos de usos con sus tablas descriptivas.

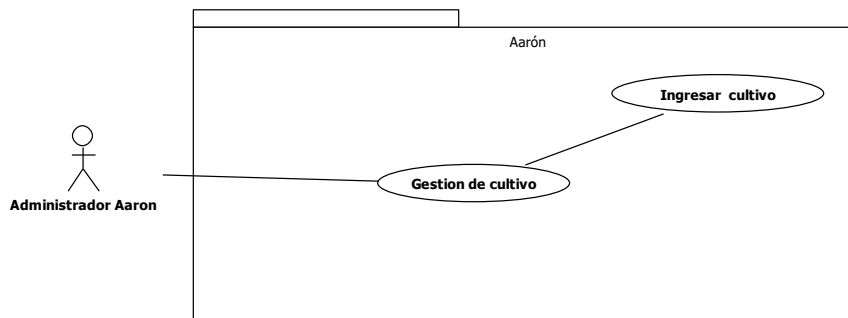
Diagrama representativo del caso de uso: Iniciar sesión.



Caso de uso	INICIAR SESIÓN	
Descripción	El desenvolvimiento del sistema debe de presentar el paso a paso del caso de uso cuando el usuario pretende iniciar sesión.	
Precondición	El usuario debe de contar con una cuenta institucional activa y estar registrado en el sistema.	
Secuencia Normal	Paso	Acción
	1	El usuario ingresa Al sistema para iniciar sesión.
	2	El sistema único de autenticación (CAS) muestra la pantalla de ingreso de credenciales.
	3	El usuario ingresa su nombre y su contraseña.
Post Condición	El sistema muestra la página principal de inicio, con el estado de usuario logueado.	
Excepciones	Pasos	Acción

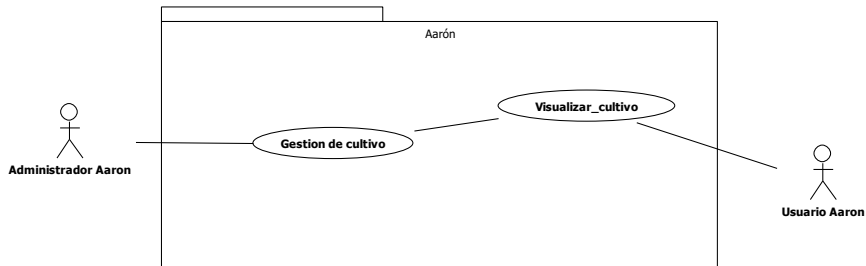
	1	Si el usuario no cuenta con usuario y contraseña institucionales, no podrá ingresar al sistema. Mostrando el sistema de autenticación CAS un error.
	2	Si el usuario cancela el proceso de ingreso, termina el proceso de ingreso al sistema.

Diagrama representativo del caso de uso denominado: Agregar cultivo.



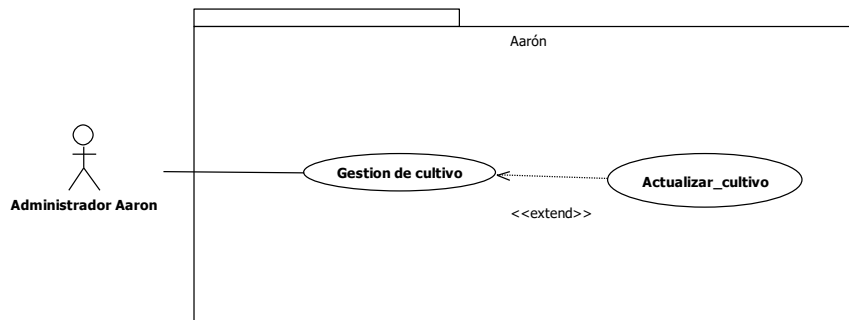
Caso de uso	INGRESAR CULTIVO	
Descripción	El desenvolvimiento del sistema debe de presentar el paso a paso del caso de uso cuando el usuario agrega un nuevo cultivo en el sistema.	
Precondición	El cultivo no debe de estar registrado en el sistema. El usuario debe de haber iniciado sesión en el sistema.	
Secuencia Normal	Paso	Acción
	1	El usuario ingresa al sistema para agregar un nuevo cultivo
	2	El sistema presenta en pantalla el formulario para registro de un nuevo cultivo.
	3	El usuario ingresa los datos del nuevo cultivo.
	4	El sistema guarda la información ingresada por el usuario y confirma su almacenamiento con un mensaje de éxito.
Post Condición	El sistema mostrará el listado de los cultivos incluido el último registro que se ingresó.	
Excepciones	Pasos	Acción
	1	Si todos los campos que son obligatorios no son llenados por el usuario el usuario, no se podrá almacenar el nuevo cultivo.
	2	En caso de que el usuario cancele el proceso de ingresar un nuevo cultivo, el sistema terminara con esta tarea.

Diagrama representativo del caso de uso: Visualizar cultivo.



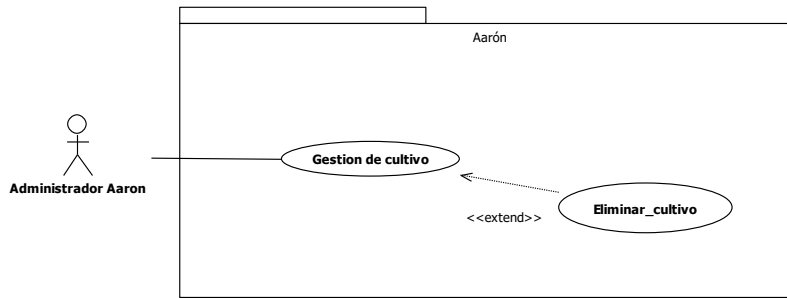
Caso de uso	VISUALIZAR CULTIVO	
Descripción	El desenvolvimiento del sistema debe de presentar el paso a paso del caso de uso cuando el usuario vaya a visualizar un cultivo.	
Precondición	El cultivo debe de estar registrado en el sistema. El usuario debe de haber iniciado sesión en el sistema.	
Secuencia Normal	Paso	Acción
	1	El usuario selecciona en el menú los cultivos.
	2	El usuario da clic en el ícono del ojo o vista, el cual significa visualizar.
Post Condición	El sistema recupera y muestra en pantalla los datos del cultivo.	
Excepciones	Pasos	Acción
	1	En caso de que el usuario cancele el proceso de visualizar cultivo, el sistema terminara con esta tarea.

Diagrama representativo del caso de uso: Actualizar cultivo



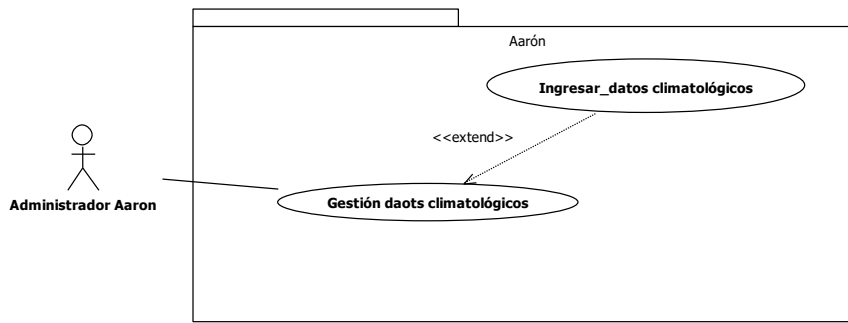
Caso de uso	ACTUALIZAR CULTIVO	
Descripción	El desenvolvimiento del sistema debe de presentar el paso a paso del caso de uso cuando el usuario vaya a actualizar un cultivo.	
Precondición	El cultivo debe de estar ingresado en el sistema. El usuario debe de haber iniciado sesión en el sistema.	
Secuencia Normal	Paso	Acción
	1	El usuario selecciona el cultivo que desea actualizar.
	2	El usuario da clic en el ícono del lápiz, el cual significa editar la información.
	3	El sistema muestra en pantalla la información del cultivo para modificarla.
	4	El usuario modifica la información que desee.
	5	El sistema almacena la información modificada confirmando con un mensaje de éxito.
Post Condición	El sistema muestra en pantalla el cultivo con los nuevos datos modificados.	
Excepciones	Pasos	Acción
	1	Si el usuario no llena los campos obligatorios, no se podrá terminar con el proceso de actualización.
	2	En caso de que el usuario cancele el proceso de modificar el cultivo, el sistema terminara con esta tarea.

Diagrama representativo del caso de uso: Eliminar cultivo.



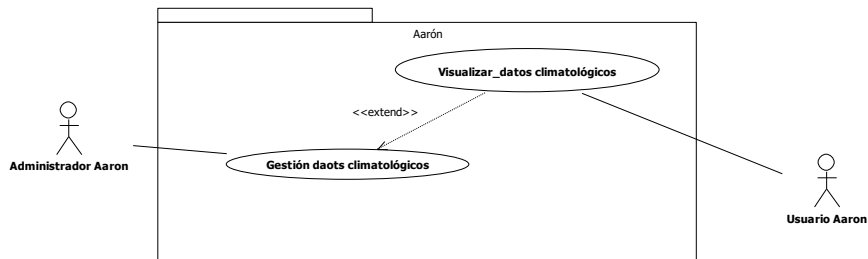
Caso de uso	ELIMINAR CULTIVO	
Descripción	El desenvolvimiento del sistema debe de presentar el paso a paso del caso de uso cuando el usuario vaya a eliminar un cultivo.	
Precondición	El cultivo debe de estar ingresado en el sistema. El usuario debe de haber iniciado sesión en el sistema.	
Secuencia Normal	Paso	Acción
	1	El usuario selecciona el cultivo que desea eliminar.
	2	El usuario da clic en el ícono del bote de basura, el cual significa eliminar.
	3	El sistema advierte si desea eliminar el cultivo.
	4	El usuario confirma la elminación del cultivo.
	5	El sistema elimina todos los datos correspondientes a ese archivo.
Post Condición	El sistema enlista los cultivos menos al que se acabo de eliminar.	
Excepciones	Pasos	Acción
	1	Si el usuario en el mensaje de advertencia escoge la opcion NO, el cultivo no se elimina.
	2	En caso de que el usuario cancele el proceso de eliminar el cultivo, el sistema terminara con esta tarea.

Diagrama del caso de uso: Agregar datos climatológicos.



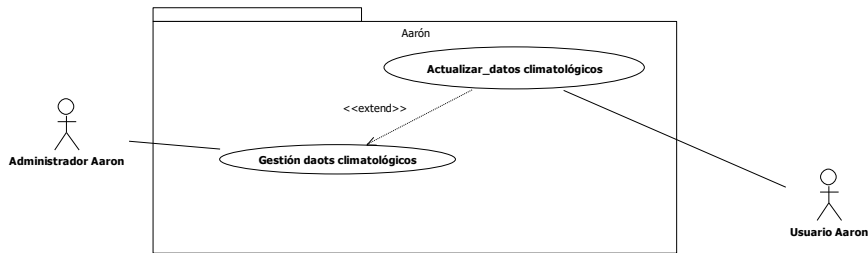
Caso de uso	INGRESAR DATOS CLIMATOLÓGICOS	
Descripción	El desenvolvimiento del sistema debe de presentar el paso a paso del caso de uso cuando el usuario agrega un nuevo dato climático en el sistema.	
Precondición	El dato climático no debe de estar registrado en el sistema. El usuario debe de haber iniciado sesión en el sistema.	
Secuencia Normal	Paso	Acción
	1	El usuario ingresa al sistema para agregar un nuevo dato climático.
	2	El sistema presenta en pantalla el formulario para registro de un nuevo dato climático.
	3	El usuario ingresa la información del nuevo dato climático.
	4	El sistema guarda la información ingresada por el usuario y confirma su almacenamiento con un mensaje de éxito.
Post Condición	El sistema mostrará el listado de los datos climáticos, incluido el último registro ingresado.	
Excepciones	Pasos	Acción
	1	Si todos los campos que son obligatorios no son llenados por el usuario el usuario, no se podrá almacenar el nuevo dato climático.
	2	En caso de que el usuario cancele el proceso de ingresar un nuevo dato climático, el sistema terminará con esta tarea.

Diagrama representativo del caso de uso: Visualizar datos climatológicos.



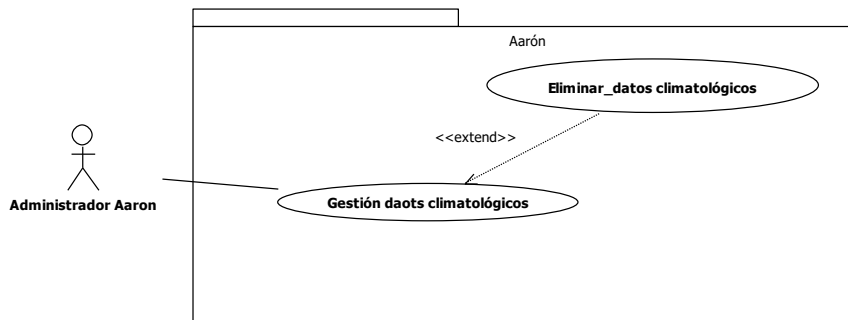
Caso de uso	VISUALIZAR DATO CLIMATOLÓGICO	
Descripción	El desenvolvimiento del sistema debe de presentar el paso a paso del caso de uso cuando el usuario vaya a visualizar un dato climatológico.	
Precondición	El dato climatológico debe de estar registrado en el sistema. El usuario debe de haber iniciado sesión en el sistema.	
Secuencia Normal	Paso	Acción
	1	El usuario selecciona en el menú dato climatológico.
	2	El usuario da clic en el ícono del ojo o vista, el cual significa visualizar.
Post Condición	El sistema recupera y muestra en pantalla los datos climatológicos.	
Excepciones	Pasos	Acción
	1	En caso de que el usuario cancele el proceso de visualizar dato climatológico, el sistema terminara con esta tarea.

Diagrama representativo del caso de uso: Actualizar datos climatológicos.



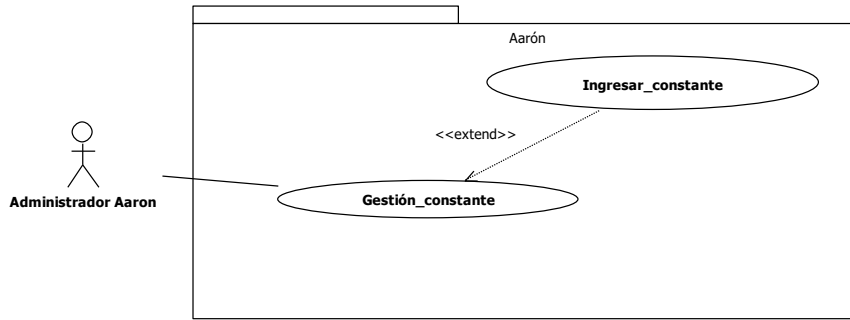
Caso de uso	ACTUALIZAR DATO CLIMATOLÓGICO	
Descripción	El desenvolvimiento del sistema debe de presentar el paso a paso del caso de uso cuando el usuario vaya a actualizar un dato climatológico.	
Precondición	El dato climatológico debe de estar ingresado en el sistema. El usuario debe de haber iniciado sesión en el sistema.	
Secuencia Normal	Paso	Acción
	1	El usuario selecciona el dato climatológico que desea actualizar.
	2	El usuario da clic en el ícono del lápiz, el cual significa editar la información.
	3	El sistema muestra en pantalla la información del dato climatológico para modificarla.
	4	El usuario modifica la información que desee.
	5	El sistema almacena la información modificada confirmando con un mensaje de éxito.
Post Condición	El sistema muestra en pantalla el dato climatológico con la nueva información modificada.	
Excepciones	Pasos	Acción
	1	Si el usuario no llena los campos obligatorios, no se podrá terminar con el proceso de actualización.
	2	En caso de que el usuario cancele el proceso de modificar el dato climatológico, el sistema terminara con esta tarea.

Diagrama representativo del caso de uso: Eliminar datos climatológicos.



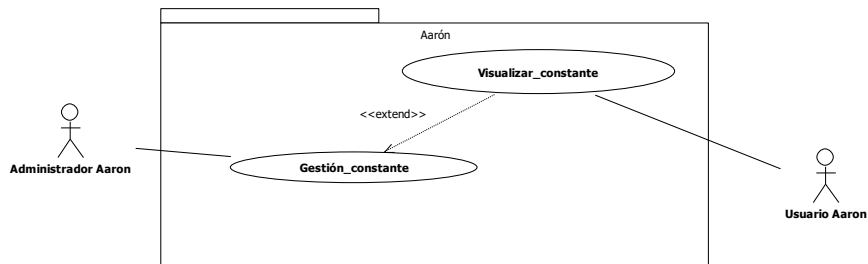
Caso de uso	ELIMINAR DATO CLIMATOLÓGICO	
Descripción	El desenvolvimiento del sistema debe de presentar el paso a paso del caso de uso cuando el usuario vaya a eliminar un dato climatológico.	
Precondición	El dato climatológico debe de estar ingresado en el sistema. El usuario debe de haber iniciado sesión en el sistema.	
Secuencia Normal	Paso	Acción
	1	El usuario selecciona el dato climatológico que desea eliminar.
	2	El usuario da clic en el ícono del bote de basura, el cual significa eliminar el dato climatológico.
	3	El sistema advierte si desea eliminar el dato climatológico.
	4	El usuario confirma la elminación del dato climatológico.
	5	El sistema elimina todos los datos correspondientes a ese dato climatológico.
Post Condición	El sistema lista los datos climatológicos, menos al que se acabo de eliminar.	
Excepciones	Pasos	Acción
	1	Si el usuario en el mensaje de advertencia escoge la opcion NO, el dato climatológico no se elimina.
	2	En caso de que el usuario cancele el proceso de eliminar el dato climatológico, el sistema terminara con esta tarea.

Diagrama representativo del caso de uso: Ingresar constante.



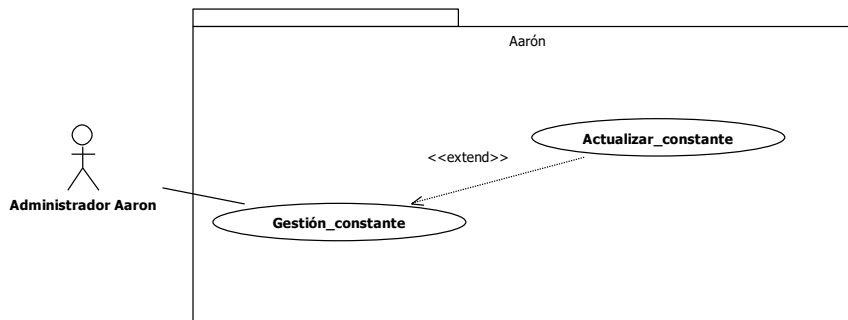
Caso de uso	INGRESAR CONSTANTE	
Descripción	El desenvolvimiento del sistema debe de presentar el paso a paso del caso de uso cuando el usuario desee agregar una nueva constante en el sistema.	
Precondición	La constante no debe de estar registrada en el sistema. El usuario debe de haber iniciado sesión en el sistema.	
Secuencia Normal	Paso	Acción
	1	El usuario ingresa al sistema para agregar una nueva constante.
	2	El sistema presenta en pantalla el formulario para registro de una nueva constante.
	3	El usuario ingresa la información de la nueva constante.
	4	El sistema guarda la información ingresada por el usuario y confirma su almacenamiento con un mensaje de éxito.
Post Condición	El sistema mostrará el listado de las constantes, incluido el último registro que se ingresó.	
Excepciones	Pasos	Acción
	1	Si todos los campos que son obligatorios no son llenados por el usuario el usuario, no se podrá almacenar la nueva constante.
	2	En caso de que el usuario cancele el proceso de ingresar una nueva constante, el sistema terminará con esta tarea.

Diagrama representativo del caso de uso: Visualizar constante.



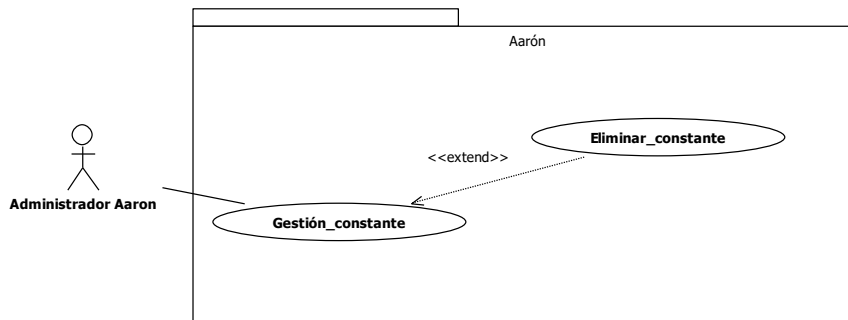
Caso de uso	VISUALIZAR CONSTANTE	
Descripción	El desenvolvimiento del sistema debe de presentar el paso a paso del caso de uso cuando el usuario vaya a visualizar una constante.	
Precondición	La constante debe de estar registrado en el sistema. El usuario debe de haber iniciado sesión en el sistema.	
Secuencia Normal	Paso	Acción
	1	El usuario selecciona en el menú, constantes.
	2	El usuario da clic en el ícono del ojo o vista, el cual significa visualizar.
Post Condición	El sistema recupera y muestra en pantalla los datos de las constantes.	
Excepciones	Pasos	Acción
	1	En caso de que el usuario cancele el proceso de visualizar constante, el sistema terminara con esta tarea.

Diagrama representativo del caso de uso: Actualizar constante.



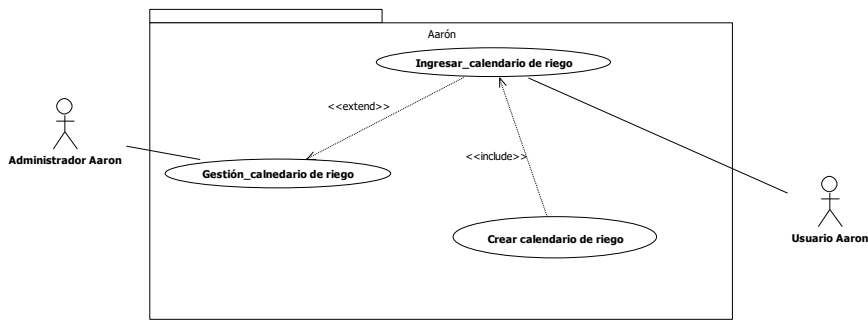
Caso de uso	ACTUALIZAR CONSTANTE	
Descripción	El desenvolvimiento del sistema debe de presentar el paso a paso del caso de uso cuando el usuario vaya a actualizar una constante.	
Precondición	La constante debe de estar ingresado en el sistema. El usuario debe de haber iniciado sesión en el sistema.	
Secuencia Normal	Paso	Acción
	1	El usuario selecciona la constante que desea actualizar.
	2	El usuario da clic en el ícono del lápiz, el cual significa editar la información.
	3	El sistema muestra en pantalla la información de la constante para modificarla.
	4	El usuario modifica la información que desee.
	5	El sistema almacena la información modificada confirmando con un mensaje de éxito.
Post Condición	El sistema muestra en pantalla la constante con la información modificada.	
Excepciones	Pasos	Acción
	1	Si el usuario no llena los campos obligatorios, no se podrá terminar con el proceso de actualización.
	2	En caso de que el usuario cancele el proceso de modificar la constante, el sistema terminara con esta tarea.

Diagrama representativo del caso de uso: Eliminar constante.



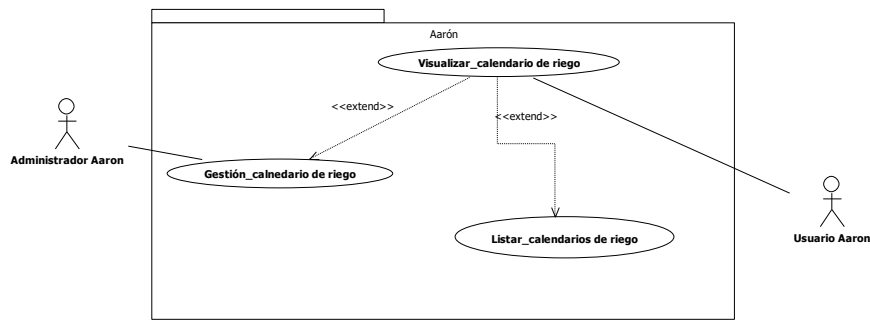
Caso de uso	ELIMINAR CONSTANTE	
Descripción	El desenvolvimiento del sistema debe de presentar el paso a paso del caso de uso cuando el usuario vaya a eliminar una constante.	
Precondición	La constante debe de estar ingresado en el sistema. El usuario debe de haber iniciado sesión en el sistema.	
Secuencia Normal	Paso	Acción
	1	El usuario selecciona la constante que desea eliminar.
	2	El usuario da clic en el ícono del bote de basura, el cual significa eliminar.
	3	El sistema advierte si desea eliminar la constante.
	4	El usuario confirma la eliminación de la constante.
	5	El sistema elimina todos los datos correspondientes a esa constante.
Post Condición	El sistema lista las constantes, menos la que se acaba de eliminar.	
Excepciones	Pasos	Acción
	1	Si el usuario en el mensaje de advertencia escoge la opción NO, la constante no se elimina.
	2	En caso de que el usuario cancele el proceso de eliminar constante, el sistema terminara con esta tarea.

Diagrama representativo del caso de uso: Ingresar calendario de riego.



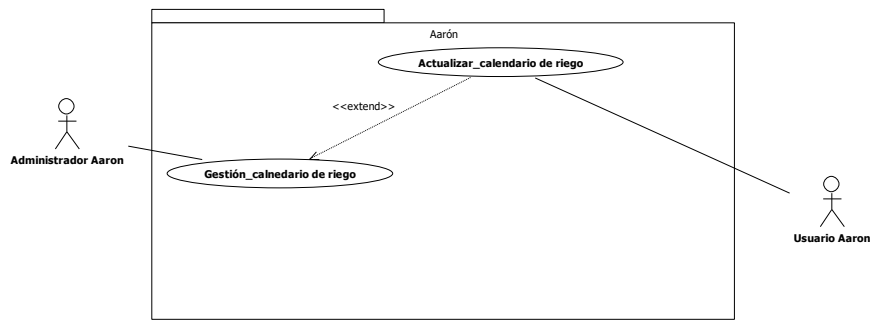
Caso de uso	INGRESAR CALENDARIO DE RIEGO	
Descripción	El desenvolvimiento del sistema debe de presentar el paso a paso del caso de uso cuando el usuario pretenda crear un nuevo calendario de riego.	
Precondición	El calendario de riego no debe de estar registrado en el sistema. El usuario debe de haber iniciado sesión en el sistema.	
Secuencia Normal	Paso	Acción
	1	El usuario ingresa al sistema para crear un nuevo calendario de riego.
	2	El sistema presenta en pantalla el formulario para la creación de un nuevo calendario de riego.
	3	El usuario ingresa la información necesaria para crear un nuevo calendario de riego.
	4	El sistema procesa la información ingresada por el usuario y crea el calendario de riego.
Post Condición	El sistema mostrará el calendario de riego generado.	
Excepciones	Pasos	Acción
	1	Si todos los campos que son obligatorios no son llenados por el usuario el usuario, no se podrá crear el calendario de riego.
	2	En caso de que el usuario cancele el proceso de crear un nuevo calendario de riego, el sistema terminará con esta tarea.

Diagrama representativo del caso de uso: Visualizar calendario de riego.



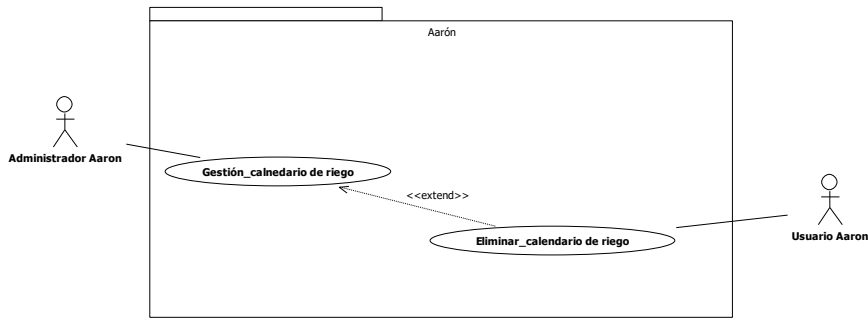
Caso de uso	VISUALIZAR CALENDARIO DE RIEGO	
Descripción	El desenvolvimiento del sistema debe de presentar el paso a paso del caso de uso cuando el usuario vaya a visualizar un calendario de riego.	
Precondición	El calendario de riego debe de estar registrado en el sistema. El usuario debe de haber iniciado sesión en el sistema.	
Secuencia Normal	Paso	Acción
	1	El usuario selecciona en el menú, calendario de riego.
	2	El usuario da clic en el ícono del ojo o vista, el cual significa visualizar.
Post Condición	El sistema recupera y muestra en pantalla los datos de los calendarios de riego.	
Excepciones	Pasos	Acción
	1	En caso de que el usuario cancele el proceso de visualizar calendario de riego, el sistema terminara con esta tarea.

Diagrama representativo del caso de uso: Actualizar calendario de riego.



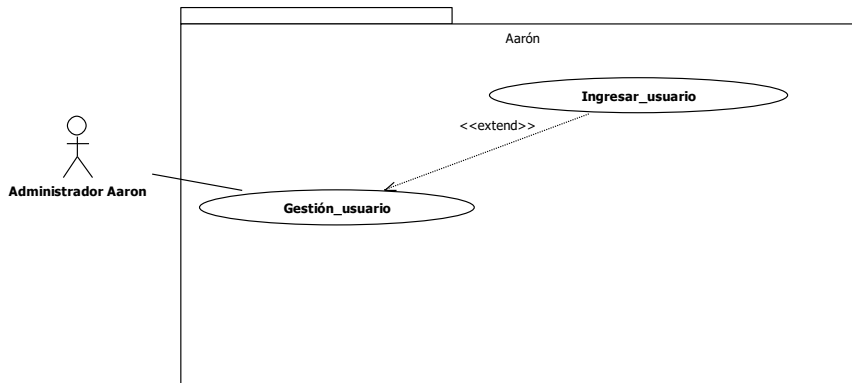
Caso de uso	ACTUALIZAR CALENDARIO DE RIEGO	
Descripción	El desenvolvimiento del sistema debe de presentar el paso a paso del caso de uso cuando el usuario vaya a actualizar un calendario de riego.	
Precondición	El calendario de riego debe de estar ingresado en el sistema. El usuario debe de haber iniciado sesión en el sistema.	
Secuencia Normal	Paso	Acción
	1	El usuario selecciona el calendario de riego que desea actualizar.
	2	El usuario da clic en el ícono del lápiz, el cual significa editar la información.
	3	El sistema muestra en pantalla la información del calendario de riego para modificarlo.
	4	El usuario modifica la información que desee.
	5	El sistema almacena la información modificada confirmando con un mensaje de éxito.
Post Condición	El sistema muestra en pantalla el calendario de riego con la información modificada.	
Excepciones	Pasos	Acción
	1	Si el usuario no llena los campos obligatorios, no se podrá terminar con el proceso de actualización.
	2	En caso de que el usuario cancele el proceso de modificar calendario de riego, el sistema terminara con esta tarea.

Diagrama representativo del caso de uso: Eliminar calendario de riego.



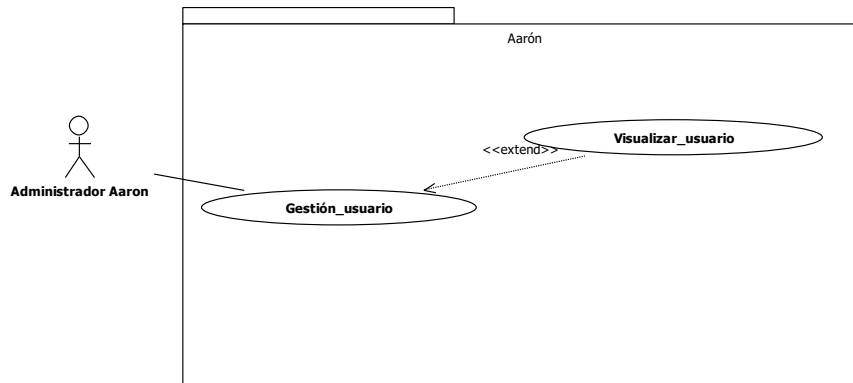
Caso de uso	ELIMINAR CALENDARIO DE RIEGO	
Descripción	El desenvolvimiento del sistema debe de presentar el paso a paso del caso de uso cuando el usuario vaya a eliminar un calendario de riego.	
Precondición	El calendario de riego debe de estar ingresado en el sistema. El usuario debe de haber iniciado sesión en el sistema.	
Secuencia Normal	Paso	Acción
	1	El usuario selecciona el calendario de riego. que desea eliminar.
	2	El usuario da clic en el ícono del bote de basura, el cual significa eliminar.
	3	El sistema advierte si desea eliminar el calendario de riego.
	4	El usuario confirma la eliminación del calendario de riego.
	5	El sistema elimina todos los datos correspondientes a ese calendario de riego.
Post Condición	El sistema lista los calendarios de riego, menos al que se acaba de eliminar.	
Excepciones	Pasos	Acción
	1	Si el usuario en el mensaje de advertencia escoge la opcion NO, el calendario de riego no se elimina.
	2	En caso de que el usuario cancele el proceso de eliminar calendario de riego, el sistema terminara con esta tarea.

Diagrama representativo del caso de uso: Ingresar usuario.



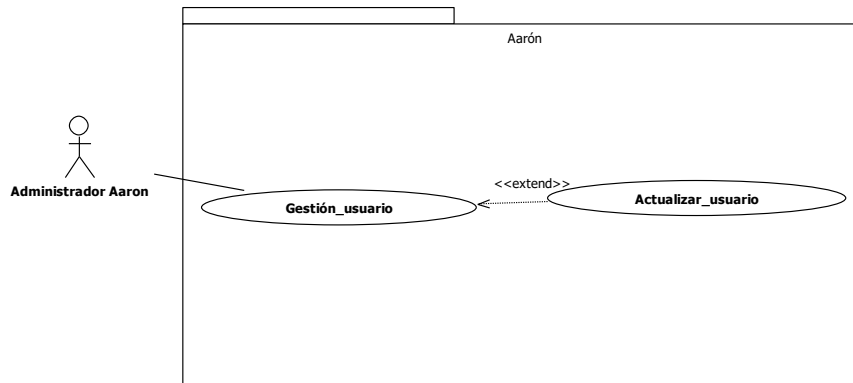
Caso de uso	INGRESAR USUARIO	
Descripción	El desenvolvimiento del sistema debe de presentar el paso a paso del caso de uso cuando el administrador pretenda crear un nuevo usuario.	
Precondición	El usuario no debe de estar registrado en el sistema. El administrador debe de haber iniciado sesión en el sistema.	
Secuencia Normal	Paso	Acción
	1	El administrador ingresa al sistema para crear un nuevo usuario.
	2	El sistema presenta en pantalla el formulario para la creación de un nuevo usuario.
	3	El administrador ingresa la información necesaria para crear un nuevo usuario.
	4	El sistema registra la información ingresada.
Post Condición	El sistema mostrará la lista de usuarios registrados en el sistema.	
Excepciones	Pasos	Acción
	1	Si todos los campos que son obligatorios no son llenados por el administrador del sistema, no se podrá crear el usuario.
	2	En caso de que el administrador cancele el proceso de ingresar un nuevo usuario, el sistema terminará con esta tarea.

Diagrama representativo del caso de uso: Visualizar usuario.



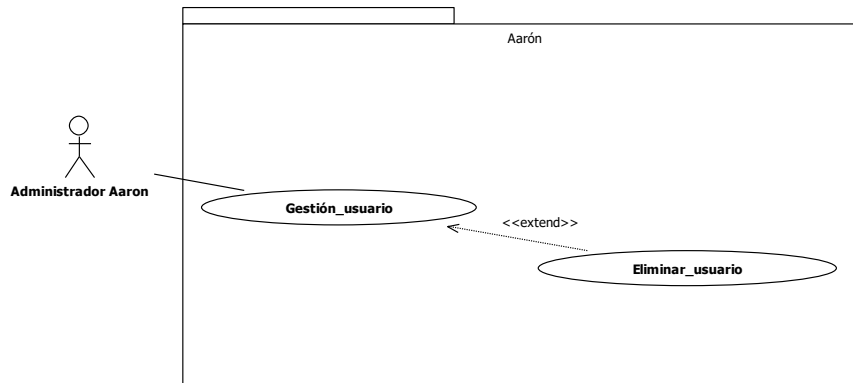
Caso de uso	VISUALIZAR USUARIO	
Descripción	El desenvolvimiento del sistema debe de presentar el paso a paso del caso de uso cuando el administrador vaya a visualizar los usuarios.	
Precondición	El usuario debe de estar registrado en el sistema. El administrador debe de haber iniciado sesión en el sistema.	
Secuencia Normal	Paso	Acción
	1	El administrador selecciona en el menú, usuarios.
	2	El administrador da clic en el ícono del ojo o vista, el cual significa visualizar.
Post Condición	El sistema recupera y muestra en pantalla los datos de los usuarios.	
Excepciones	Pasos	Acción
	1	En caso de que el administrador cancele el proceso de visualizar usuarios, el sistema terminara con esta tarea.

Diagrama representativo del caso de uso: Actualizar usuario.



Caso de uso	ACTUALIZAR USUARIO	
Descripción	El desenvolvimiento del sistema debe de presentar el paso a paso del caso de uso cuando el administrador vaya a actualizar un usuario.	
Precondición	El usuario debe de estar ingresado en el sistema. El administrador debe de haber iniciado sesión en el sistema.	
Secuencia Normal	Paso	Acción
	1	El administrador selecciona el usuario que desea actualizar.
	2	El usuario da clic en el ícono del lápiz, el cual significa editar la información.
	3	El sistema muestra en pantalla la información del usuario para modificarlo.
	4	El administrador modifica la información que desee.
	5	El sistema almacena la información modificada confirmando con un mensaje de éxito.
Post Condición	El sistema muestra en pantalla el usuario con la información modificada.	
Excepciones	Pasos	Acción
	1	Si el usuario no llena los campos obligatorios, no se podrá terminar con el proceso de actualización.
	2	En caso de que el administrador cancele el proceso de modificar usuario, el sistema terminara con esta tarea.

Diagrama representativo del caso de uso: Eliminar usuario.



Caso de uso	ELIMINAR USUARIO	
Descripción	El desenvolvimiento del sistema debe de presentar el paso a paso del caso de uso cuando el administrador vaya a eliminar un usuario.	
Precondición	El usuario debe de estar ingresado en el sistema. El administrador debe de haber iniciado sesión en el sistema.	
Secuencia Normal	Paso	Acción
	1	El administrador selecciona el usuario que desea eliminar.
	2	El administrador da clic en el ícono del bote de basura, el cual significa eliminar.
	3	El sistema advierte si desea eliminar el usuario.
	4	El administrador confirma la eliminación del usuario.
	5	El sistema elimina todos los datos correspondientes a ese usuario.
Post Condición	El sistema lista los usuarios, menos al que se acaba de eliminar.	
Excepciones	Pasos	Acción
	1	Si el administrador en el mensaje de advertencia escoge la opcion NO, el usuario no se elimina.
	2	En caso de que el administrador cancele el proceso de eliminar usuario el sistema terminara con esta tarea.

ANEXO C Estándar de Programación

Los programadores deben implementar un estándar de forma prudente, éste debe tender siempre a lo práctico. Un código fuente completo debe reflejar un estilo armonioso, como si un único programador hubiera escrito todo el código de una sola vez.

Al comenzar un proyecto de software, se debe establecer un estándar de codificación para asegurarse de que todos los programadores del proyecto trabajen de forma coordinada y ordenada.

Para poder obtener una programación ordenada y coordinada se definió el estándar de programación CamelCase.

Objetos de ejecución.

Tipo de Objeto	Extensión
Documento HTML	.html
Archivo Include	.inc
Java Server Pages	.jsp
Clase Java	.class
Librería de clases empaquetadas	.jar o .zip
Hoja de estilos	csc
Java Scrips	.js

Tabla 12: Especificaciones de los objetos de ejecución. Fuente: Isaac Torres.

OBJETOS MULTIMEDIA

Tipo de Objeto	Extensión
Imágenes JPG	.jpg
Imágenes GIF	.gif
Archivos MIF	.mif
Archivos de audio	.aud
Archivos flash	.swf

Tabla 13: Especificaciones de los objetos multimedia. Fuente: Isaac Torres.

OBJETOS DE FUENTES

Tipo de Objeto	Extensión
Código Java	.java
Código XML	.xml
Archivo de propiedades	.properties

Tabla 14: Especificaciones de los objetos de fuentes. Fuente: Isaac Torres.

Estándar para Variables

Para la definición de variables debemos tener presente algunas consideraciones:

- La elección de un nombre de variable debe ser fácil de recordar para demostrar el uso a cualquier observador.
- Los nombres de variables de un solo carácter se deben utilizar únicamente para variables temporales, como i, j o k en ciclos repetitivos.

Tipo de Dato	Prefijo	Ejemplo
Números Enteros		
byte	byt	bytRasterData
short	shr	shrNumber
int	int	intQuantity
long	lon	lonDistance
Números Reales		
float	flt	fltPrice
double	dbl	dblTotalSalary
Otros Tipos		
char	chr	chrOption
boolean	bol	bolFound
String	str	strName

Tabla 15: Especificaciones de los tipos de variables. Fuente: Isaac Torres.

Estándar para Constantes

Para la definición de constantes debe tener en cuenta las siguientes:

- Los nombres de variables declaradas como constante deben ser todas en mayúsculas con palabras separadas por guion abajo (“_”).
- Se debe seguir los mismos estándares que se usan para variables en cuanto a los prefijos para tipo de dato.

Ejemplos

```
Public static final int IMIN_WIDTH = 4;
public static final int IMAX_WIDTH = 999;
public static final int IGET_THE_CPU = 1;
```

Estándar para Clases

Reglas para la definición de clases

- Usar palabras completas, evite abreviaturas o acrónimos, a menos que sean mucho más usadas que el nombre completo.
- Los nombres de clase deben empezar con una letra mayúscula y el resto de letras deben estar escritas en minúscula. En el caso que pueda tener más de una palabra, las primeras letras de cada palabra interna deben ser con mayúscula.
- Los nombres de clase no pueden contener espacios ni caracteres especiales, sólo son permitidas las letras de la “a” a la “z” y los números del 0 al 9.

Estándar para Atributos de Clases

Reglas para definición de atributos de clases

- Los nombres de atributos deben empezar con una letra mayúscula y el resto de letras deben estar escritas en minúsculas.
- Los nombres de atributo no pueden contener espacios ni caracteres especiales, sólo son permitidos las letras de la “a” a la “z” y los números del 0 al 9.
- Los nombres de atributo no pueden ser verbos.

Estándar para Métodos

Reglas para definir los métodos

- Los nombres de atributos deben empezar con una letra mayúscula y el resto de letras deben estar escritas en minúsculas.
- Los nombres de los métodos deben ser verbos o palabras que identifiquen de manera general el objetivo del método.
- Los nombres de los métodos no pueden contener espacios ni caracteres especiales, sólo son permitidas las letras de la “a” a la “z” y los números del 0 al 9.

- Si el nombre de método requiere estar compuesto por más de una palabra, cada palabra adicional debe empezar con mayúscula.

Estándar para Paquetes

Los nombres de los paquetes sólo se escriben en letras minúsculas, por ejemplos:

com.empresa.apli.nombremodulo.subparte

Donde

- empresa, por ejemplo, cibertec, integra, vcp, uni, etc.
- apli, es el código de la aplicación según la tabla de aplicaciones.
- nombremodulo, es el nombre del módulo, por ejemplo: mantenimiento, reporte, etc.
- subparte, cada módulo ordena sus clases en subpartes según el framework que utiliza.

Estándares de Diseño y Programación

Diseño Visual

Para el diseño Web debe ser el establecido ya previamente por el departamento de desarrollo del DTIC por tratarse de un sistema institucional de la ESPOCH.

Entorno de Programación

El entorno de desarrollo en el que se trabaja es Netbeans en su versión 8.0.2 se eligió este por la razón que facilita el mapeo de las tablas de la base de datos.

Con respecto a la INDENTACION

- Usar cuatro espacios o Tab como unidad de indentación.
- Alinear la nueva línea al principio de la expresión al mismo nivel de la línea anterior.

Ejemplo:

```
someMethod(longExpression1, longExpression2, longExpression3,  
           longExpression4, longExpression5);
```

Con respecto a las DECLARACIONES

- Se debe declarar cada variable en su propia línea.
- Es válido inicializar variables al momento de su declaración.

Ejemplos válidos:

```
int inumero = 9;
```

```
int icantidad;
```

NO utilizar:

```
int inumero, icantidad
```

Con respecto a las SENTENCIAS

- Usar sentencias simples, cada sentencia debe ser escrita en su propia línea.
- No utilizar el operador coma para agrupar declaraciones múltiples. Evitar:

```
if(err) {  
Format.print(System.out, "error"), exit(1); }
```

Sentencias de retorno

- Una sentencia return con un valor no se debe usar.
- Intentar hacer que la estructura del programa se ajuste a su intención.

Con respecto a las SENTENCIAS IF

- Las sentencias if siempre utilizan paréntesis.
- Deben tener la siguiente forma :

```
if (condition) {  
    statements;  
}
```

```
if (condition) {  
    statements;  
} else {  
    statements;  
}
```

Con respecto a las SENTENCIAS FOR

Deben tener la siguiente forma:

```
for (initialization; condition; update) {  
    statements;  
}
```

Con respecto a las SENTENCIAS WHILE

Deben tener la siguiente forma:

```
while (condition) {  
    statements;
```

```
}
```

Una sentencia do – While debe tener la siguiente forma:

```
do {  
statements;  
} while (condition);
```

Con respecto a las SENTENCIAS SWITCH

- Deben tener la siguiente forma:

```
switch (condition) {  
case ABC:  
statements;  
/* no se concreta */  
case DEF:  
statements;  
break;  
case XYZ:  
statements;  
break;  
default:  
statements;  
}
```

- Cada sentencia switch debe incluir un case por default.

Con respecto a las SENTENCIAS TRY – CATCH

- Deben tener el siguiente formato:

```
try {  
statements;  
} catch (ExceptionClass e) {  
statements;  
}
```

- Se recomienda usar la sentencia finally, cuando como parte de los statements se maneja conexiones a BD, para asegurar que la conexión a BD sea cerrada.

```
try {
    statements;
} catch (ExceptionClass e) {
    statements;
} finally {
    statements;}
```

Con respecto a las CONCATENACIONES

- La concatenación de cadenas String no debe realizarse con la unión de varias cadenas, pues esto consume recursos de memoria innecesariamente.
- Este es un modo incorrecto de concatenar cadenas:

```
String cadenaFinal = "Esta es " + "una " + "concatenación " + "incorrecta";
```

- Este es un modo correcto de concatenar cadenas:

```
StringBuffer sb = new StringBuffer();
sb.append("Esta es ");
sb.append("una ");
sb.append("concatenación ");
sb.append("correcta");
String cadenaFinal = sb.toString();
```

PRUEBA DE ACEPTACIÓN	
Código: PA_01	Historia de Usuario: HT_04 Definición del estándar de programación
Nombre: Documentación del estándar de codificación	
Responsable: Isaac Torres	Fecha: 29/03/2019
Descripción: Documentar detalladamente el estándar a implementar	
Condiciones de Ejecución: Tener el estándar ya establecido y analizado	
Pasos de ejecución: Verificar que los componentes del estándar Java en el que se basará contiene: métodos, clases, objetos, variables, constantes, paquetes.	
Resultado esperado: El estándar a implementar sea entendible	

Evaluación de la prueba: Exitosa.
--

Tabla 16: Prueba 1 de aceptación. Fuente Isaac Torres.

PRUEBA DE ACEPTACIÓN	
Código: PA_02	Historia de Usuario: HT_04 Definición del estándar de programación
Nombre: Verificar que el estándar de codificación cumpla con las expectativas requeridas por los desarrolladores.	
Responsable: Elías Rivera	Fecha: 29/03/2019
Descripción: El estándar deberá contener especificaciones claras de cómo crear: clases, objetos, métodos, variables, constantes, paquetes, todo el formato de escritura.	
Condiciones de Ejecución: El estándar de codificación para el desarrollo del sistema debe estar especificado.	
Pasos de ejecución: <ul style="list-style-type: none"> • Analizar el estándar que se ha especificado para el desarrollo del proyecto. • Verificar que es un estándar fácil de seguir y que permitirá desarrollar código homogéneo y comprensible. 	
Resultado esperado: Obtener un único estándar para la codificación que se realizara en el proyecto.	
Evaluación de la prueba: Exitosa.	

Tabla 17: Prueba 2 de aceptación. Fuente Isaac Torres.

Fuente:

- <https://medium.com/@alonsus91/convencción-de-nombres-desde-el-camelcase-hasta-el-kebab-case-787e56d6d023>
- <https://www.neumoytoraxpanama.org/cloud/resources/documentos/camelcase.pdf>
- <http://www.itnuevolaredo.edu.mx/takeyas/Apuntes/POO/Apuntes/02.-%20NomenclaturaComponentesProyecto.pdf>
- <https://javadesdecero.es/fundamentos/convenciones-nomenclatura-java/>

ANEXO D Manual de usuario

**ESCUELA SUPERIOR POLITÉCNICA DE
CHIMBORAZO**

FACULTAD DE INFORMÁTICA Y ELECTRÓNICA

ESCUELA DE INGENIERÍA EN SISTEMAS

CARRERA DE INGENIERÍA EN SISTEMAS



MANUAL DE USUARIO

**“DESARROLLO DE UN SISTEMA WEB PARA LA
AUTOMATIZACIÓN DEL PROCESO DE CALENDARIZACIÓN
DE RIEGO PARA EL CENTRO EXPERIMENTAL DEL RIEGO DE
LA ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO”**

Presentado por:

ISAAC DAVID TORRES PAREDES

Riobamba – Ecuador

A continuacion en el manual de usuario se detalla las diferentes funcionalidades con las que cuenta la aplicacion web para desarrollar un calendario de riego, describiendo los botones en las pantalla conjuntamente con la accion que realiza.

REQUISITOS PARA EJECUTAR LA APLICACION WEB

A nivel de hardware se debe contar con:

- Un computador portatil o de escritorio.
- Conexión a internet mediante ethernet o conexión inalámbrica.

A nivel de software debe de contar con:

- Sistema operativo (de preferencia Windows)
- Navegadores de internet (Mozilla firefox, google chrome, internet explorer, opera, etc.).

Descripcion de pantalla inicial

La pantalla principal de inicio de la aplicacion web, esta distribuida en tres zonas principales:


Número de zona	Descripcion de zona
1	Zona de titulos
2	Zona de menú
3	Zona de contenido dinámico

En la parte superior derecha está el menú desplegable “EMPEZAR” desde ahí podemos desplegar el menú de todas las funcionalidades de la aplicación web.




Descripción de barras y botones

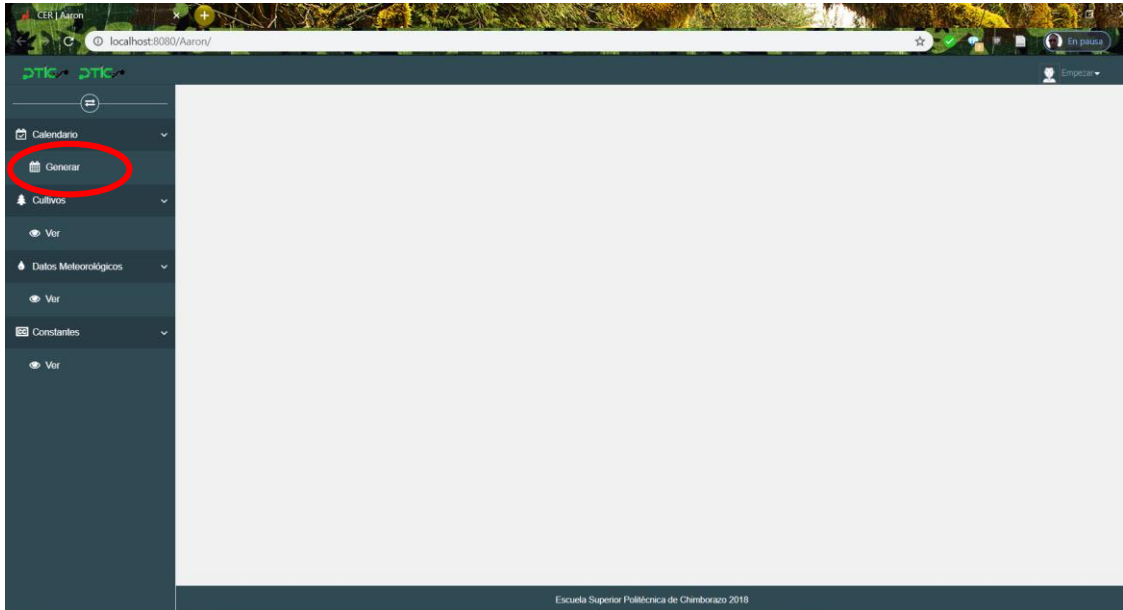
Ícono	Descripción
	Calendario: Despliega las opciones del calendario.
	Cultivo: Despliega las opciones de los datos climatológicos
	Constantes: Despliega las opciones de las constantes matemáticas.
	Cultivo: Despliega las opciones de los cultivos
	Desplegar: Minimiza el menú de las opciones y las vuelve a mostrar
	Inicio: Permite regresar a la página principal.
	Empezar: Desliza las principales opciones de la aplicación web
	Generar: Muestra en formulario de ingreso de datos
	Generar: Ejecuta permitiendo calcular y visualizar el calendario de riego
	Inicio: Permite regresar a la página principal.
	PDF: Permite generar y descargar el calendario en formato pdf
	Salir: Permite salir de la aplicación web
	Tareas: Muestra las tareas que se puede realizar en el sistema

	Visualizar: Permite visualizar los diferentes contenidos según la opción elegida.
---	--

Funcionalidades del sistema

Generar calendario

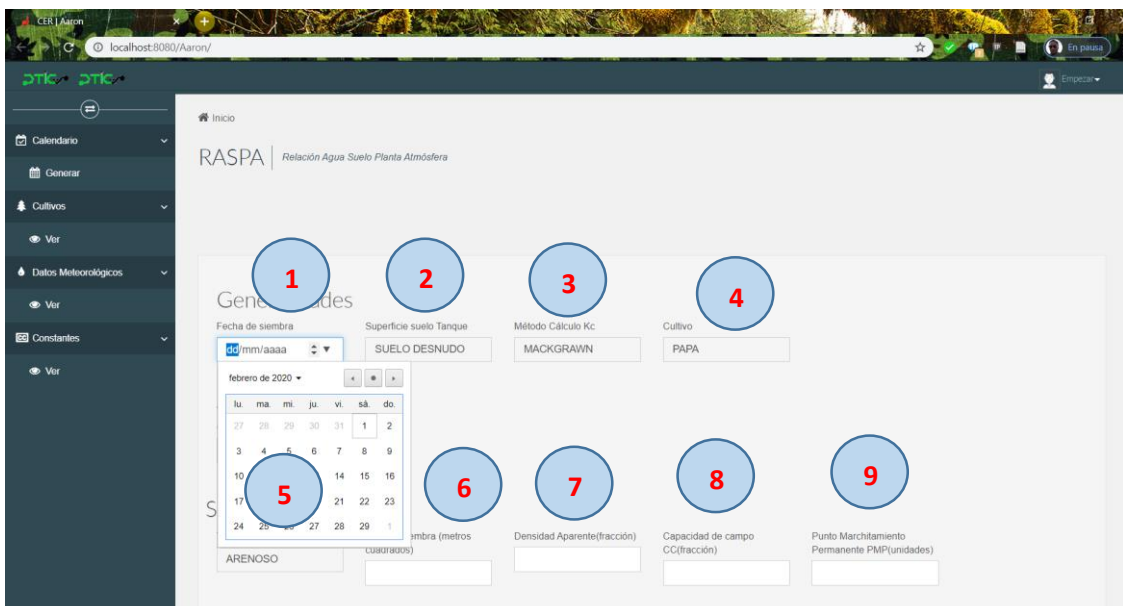
Presionar el botón  calendario se desplaza en la zona de contenido dinámico el formulario de ingreso de datos.



Formulario de ingreso de datos para el cálculo del calendario de riego.

Descripción de campos de ingreso de datos.

A continuación se describe cada uno de los campos en donde debe ingresar la información requerida para calcular el calendario de riego.



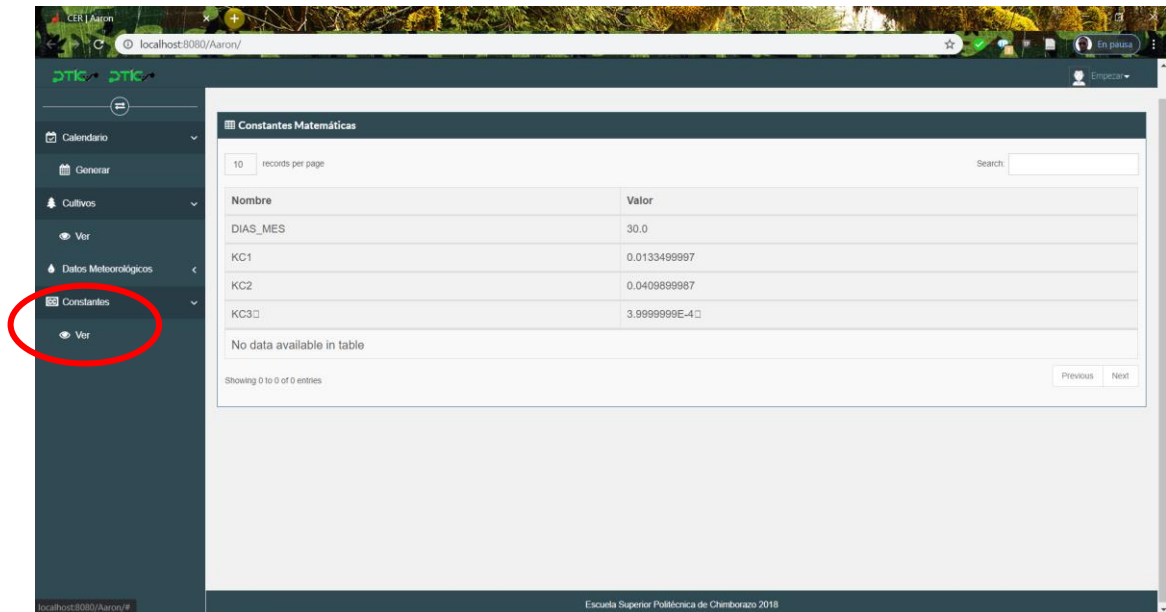
A continuación se detalla cada uno de los campos del formulario para el ingreso de datos.

Número	Descripción
1	Campo para ingreso de la fecha de la siembra
2	Campo para selección de tipo de suelo del tanque evaporímetro
3	Campo para selección de tipo de método de cálculo del Kc
4	Campo de selección de cultivo
5	Campo para selección de tipo de suelo
6	Campo para ingreso de área de siembra
7	Campo para ingreso de densidad aparente
8	Campo para ingreso de capacidad de campo
9	Campo para ingreso de punto marchitamiento permanente
10	Campo para ingreso de caudal de agua
11	Campo para ingreso de área de siembra
12	Campo para ingreso de profundidad radicular
13	Campo para ingreso de área de siembra
14	Campo para ingreso de humedad relativa
15	Campo para ingreso de velocidad de viento
16	Campo para ingreso de evapotranspiración
17	Campo para ingreso de umbral
18	Campo para selección de distancia a barlovento

Una vez que haya ingresado todos los campos requeridos dé clic en **generar**, la tabla del calendario de riego se generará.

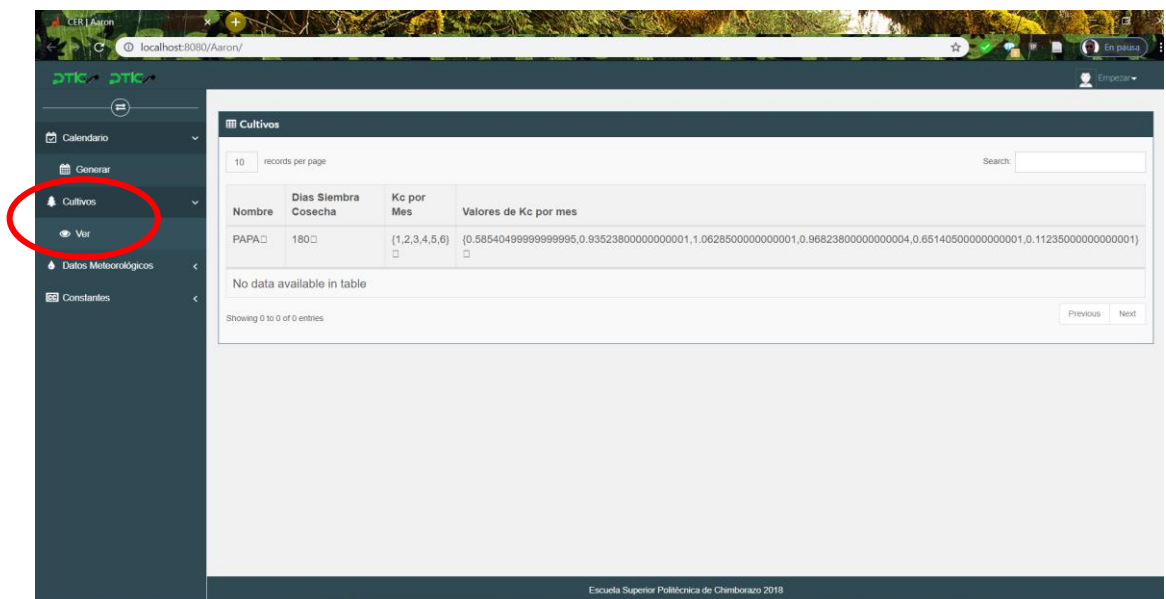
Visualizar datos de constantes con las que trabaja la aplicación web

Esta opción le permite visualizar los datos de las constantes matemáticas con los que trabaja la aplicación web.



Visualizar datos de cultivos con los que trabaja la aplicación web

Esta opción le permite visualizar los datos de los cultivos con los que trabaja la aplicación web.



Visualizar datos climatológicos de referencia con los que trabaja la aplicación web

Esta opción le permite visualizar los datos climatológicos con los que trabaja la aplicación web.

Mes	Temperatura	Precipitación	Evaporación	Humedad Relativa	Velocidad Viento
ENERO	18.5	19.0	94.0	56.5	2.3
FEBRERO	20.75	7.0	121.0	55.6	2.9
MARZO	22.75	3.0	182.0	52.4	2.8
ABRIL	24.5	0.0	221.0	48.8	3.4
MAYO	27.0	1.0	253.0	45.9	3.5
JUNIO	28.0	7.0	264.0	49.7	3.8
JULIO	31.75	53.0	239.0	55.9	3.9
AGOSTO	31.25	72.0	206.0	60.5	3.5
SEPTIEMBRE	28.5	82.0	117.0	62.9	3.6
OCTUBRE	27.5	39.0	163.0	56.8	2.7
NOVIEMBRE	22.5	24.0	120.0	56.4	2.9
DICIEMBRE	19.25	30.0	8.0	61.2	3.1

ANEXO E Tiempos de respuesta y utilización de recursos

A continuación se muestra la toma de tiempos en la elaboración del calendario de riego, de manera manual y de manera automática al grupo de usuarios de nivel básico.

TIEMPO EMPLEADO EN LA ELABORACION DEL CALENDARIO DE RIEGO							
# DATO	(MANUALMENTE)			(CON SISTEMA)			DIFERENCIA
	HORA INICIO	HORA FINALIZACIÓN	TIEMPO INVERTIDO (horas)	TIEMPO INVERTIDO (minutos)	TIEMPO INVERTIDO (minutos)	TIEMPO INVERTIDO (horas)	
1	09H20	13H05	3,75	225	15,5	0,258	209,5
2	09H20	12H53	3,55	213	14,3	0,238	198,7
3	09H20	12H50	3,5	210	12,9	0,215	197,1
4	09H20	13H09	3,82	229	15,6	0,260	213,4
5	09H20	12H48	3,47	208	14,7	0,245	193,3
6	09H20	12H47	3,45	207	13,5	0,225	193,5
7	09H20	12H51	3,52	211	13,9	0,232	197,1
8	09H20	13H00	3,67	220	15,2	0,253	204,8
9	09H20	12H48	3,47	208	17,2	0,287	190,8
10	09H20	12H47	3,45	207	16,5	0,275	190,5
11	09H20	12H50	3,50	210	16,2	0,270	193,8
12	09H20	12H49	3,48	209	15,8	0,263	193,2
13	09H20	12H42	3,37	202	14,7	0,245	187,3
14	09H20	13H01	3,68	221	14,58	0,243	206,42
15	09H20	13H05	3,75	225	13,7	0,228	211,3
16	09H20	12H58	3,63	218	14,6	0,243	203,4
17	09H20	13H07	3,78	227	15,3	0,255	211,7
18	09H20	12H59	3,65	219	15,9	0,265	203,1
19	09H20	13H12	3,87	232	16,3	0,272	215,7

20	09H20	12H51	3,52	211	16,9	0,282	194,1
21	09H20	12H55	3,58	215	14,6	0,243	200,4
22	09H20	12H48	3,47	208	13,4	0,223	194,6
23	09H20	12H40	3,33	200	16,8	0,280	183,2
24	09H20	13H09	3,65	219	15,4	0,257	203,6
25	09H20	12H51	3,52	211	16,8	0,280	194,2
26	09H20	12H49	3,48	209	15,7	0,262	193,3
27	09H20	13H04	3,73	224	14,6	0,243	209,4
28	09H20	13H00	3,67	220	14,4	0,240	205,6
29	09H20	12H59	3,65	219	15,6	0,260	203,4
30	09H20	12H56	3,60	216	17,3	0,288	198,7
31	09H20	12H43	3,38	203	16,8	0,280	186,2
32	09H20	13H02	3,70	222	15,9	0,265	206,1

Aplicación de estadística descriptiva y test de Shapiro Wilk a las muestras.

Descriptive Statistics	
<i>DIFERENCIA</i>	
Mean	199,606875
Standard Error	1,48458167
Median	198,7
Mode	198,7
Standard Deviation	8,39806216
Sample Variance	70,527448
Kurtosis	-0,79990109
Skewness	0,0829386
Range	32,5
Maximum	215,7
Minimum	183,2
Sum	6387,42
Count	32
Geometric Mean	199,435863
Harmonic Mean	199,265034
AAD	7,10730469
MAD	5,45
IQR	12,275

Shapiro-Wilk Test	
<i>DIFERENCIA</i>	
W-stat	0,97262461
p-value	0,57477848
alpha	0,05
normal	yes
d'Agostino-Pearson	
DA-stat	1,51631723
p-value	0,46852837
alpha	0,05
normal	yes

Resultados de la prueba pareada aplicada a las muestras.

Prueba t para medias de dos muestras emparejadas

	<i>Variable 1</i>	<i>Variable 2</i>	<i>Antes</i>		<i>Después</i>	
<i>Media</i>	214,9375	15,330625				
<i>Varianza</i>	67,41532258	1,392206048	<i>Media</i>	214,9375	<i>Media</i>	15,330625
<i>Observaciones</i>	32	32	<i>Error típico</i>	1,451457485	<i>Error típico</i>	0,20858197
<i>Coefficiente de correlación de Pearson</i>	-0,088766076		<i>Mediana</i>	214	<i>Mediana</i>	15,45
<i>Diferencia hipotética de las medias</i>	0		<i>Moda</i>	208	<i>Moda</i>	14,6
<i>Grados de libertad</i>	31		<i>Desviación estándar</i>	8,210683442	<i>Desviación estándar</i>	1,17991781
<i>Estadístico t</i>	134,4532796		<i>Varianza de la muestra</i>	67,41532258	<i>Varianza de la muestra</i>	1,39220605
<i>P(T<=t) una cola</i>	9,35268E-45		<i>Curtosis</i>	-0,765845657	<i>Curtosis</i>	-0,77711488
<i>Valor crítico de t (una cola)</i>	1,695518783		<i>Coefficiente de asimetría</i>	0,205127807	<i>Coefficiente de asimetría</i>	-0,16054129
<i>P(T<=t) dos colas</i>	1,87054E-44		<i>Rango</i>	32	<i>Rango</i>	4,4
<i>Valor crítico de t (dos colas)</i>	2,039513446		<i>Mínimo</i>	200	<i>Mínimo</i>	12,9
			<i>Máximo</i>	232	<i>Máximo</i>	17,3
			<i>Suma</i>	6878	<i>Suma</i>	490,58
			<i>Cuenta</i>	32	<i>Cuenta</i>	32

A continuación se muestra la toma de tiempos en la elaboración del calendario de riego, de manera manual y de manera automática al grupo de usuarios de nivel intermedio.

TIEMPO EMPLEADO EN LA ELABORACION DEL CALENDARIO DE RIEGO							
# DATO	(MANUALMENTE)			(CON SISTEMA)			DIFERENCIA
	HORA INICIO	HORA FINALIZACIÓN	TIEMPO INVERTIDO (horas)	TIEMPO INVERTIDO (minutos)	TIEMPO INVERTIDO (minutos)	TIEMPO INVERTIDO (horas)	
1	09H20	13H05	3,75	225	15,5	0,258	209,5
2	09H20	12H53	3,55	213	14,3	0,238	198,7
3	09H20	12H50	3,5	210	12,9	0,215	197,1
4	09H20	13H09	3,82	229	15,6	0,260	213,4
5	09H20	12H48	3,47	208	14,7	0,245	193,3
6	09H20	12H47	3,45	207	13,5	0,225	193,5
7	09H20	12H51	3,52	211	13,9	0,232	197,1
8	09H20	13H00	3,67	220	15,2	0,253	204,8
9	09H20	12H48	3,47	208	17,2	0,287	190,8
10	09H20	12H47	3,45	207	16,5	0,275	190,5
11	09H20	12H50	3,50	210	16,2	0,270	193,8
12	09H20	12H49	3,48	209	15,8	0,263	193,2
13	09H20	12H42	3,37	202	14,7	0,245	187,3
14	09H20	13H01	3,68	221	14,58	0,243	206,42
15	09H20	13H05	3,75	225	13,7	0,228	211,3
16	09H20	12H58	3,63	218	14,6	0,243	203,4
17	09H20	13H07	3,78	227	15,3	0,255	211,7
18	09H20	12H59	3,65	219	15,9	0,265	203,1
19	09H20	13H12	3,87	232	16,3	0,272	215,7
20	09H20	12H51	3,52	211	16,9	0,282	194,1
21	09H20	12H55	3,58	215	14,6	0,243	200,4
22	09H20	12H48	3,47	208	13,4	0,223	194,6
23	09H20	12H40	3,33	200	16,8	0,280	183,2
24	09H20	13H09	3,65	219	15,4	0,257	203,6
25	09H20	12H51	3,52	211	16,8	0,280	194,2

Aplicación de estadística descriptiva y test de Shapiro Wilk a las muestras.

Descriptive Statistics	
<i>DIFERENCIA</i>	
Mean	199,606875
Standard Error	1,48458167
Median	198,7
Mode	198,7
Standard Deviation	8,39806216
Sample Variance	70,527448
Kurtosis	-0,79990109
Skewness	0,0829386

Range	32,5
Maximum	215,7
Minimum	183,2
Sum	6387,42
Count	32
Geometric Mean	199,435863
Harmonic Mean	199,265034
AAD	7,10730469
MAD	5,45
IQR	12,275

Resultados de la prueba pareada aplicada a las muestras.

Shapiro-Wilk Test	
<i>DIFERENCIA</i>	
W-stat	0,97262461
p-value	0,57477848
alpha	0,05
normal	yes
d'Agostino-Pearson	
DA-stat	1,51631723
p-value	0,46852837
alpha	0,05
normal	yes

Prueba t para medias de dos muestras emparejadas

	Variable 1	Variable 2	Antes		Después	
Media	214,9375	15,330625				
Varianza	67,41532258	1,392206048	Media	214,9375	Media	15,330625
Observaciones	32	32	Error típico	1,451457485	Error típico	0,20858197
Coficiente de correlación de Pearson	-0,088766076		Mediana	214	Mediana	15,45
Diferencia hipotética de las medias	0		Moda	208	Moda	14,6
Grados de libertad	31		Desviación estándar	8,210683442	Desviación estándar	1,17991781
Estadístico t	134,4532796		Varianza de la muestra	67,41532258	Varianza de la muestra	1,39220605
P(T<=t) una cola	9,35268E-45		Curtosis	-0,765845657	Curtosis	-0,77711488

Valor crítico de t (una cola)	1,695518783		Coficiente de asimetría	0,205127807	Coficiente de asimetría	-0,16054129
$P(T \leq t)$ dos colas	1,87054E-44		Rango	32	Rango	4,4
Valor crítico de t (dos colas)	2,039513446		Mínimo	200	Mínimo	12,9
			Máximo	232	Máximo	17,3
			Suma	6878	Suma	490,58
			Cuenta	32	Cuenta	32

A continuación se muestra la toma de tiempos en la elaboración del calendario de riego, de manera manual y de manera automática al grupo de usuarios de nivel avanzado.

TIEMPO EMPLEADO EN LA ELABORACION DEL CALENDARIO DE RIEGO							
# DATO	(MANUALMENTE)			(CON SISTEMA)			DIFERENCIA
	HORA INICIO	HORA FINALIZACIÓN	TIEMPO INVERTIDO (horas)	TIEMPO INVERTIDO (minutos)	TIEMPO INVERTIDO (minutos)	TIEMPO INVERTIDO (horas)	
1	09H20	13H05	3,75	225	15,5	0,258	209,5
2	09H20	12H53	3,55	213	14,3	0,238	198,7
3	09H20	12H50	3,5	210	12,9	0,215	197,1
4	09H20	13H09	3,82	229	15,6	0,260	213,4
5	09H20	12H48	3,47	208	14,7	0,245	193,3
6	09H20	12H47	3,45	207	13,5	0,225	193,5
7	09H20	12H51	3,52	211	13,9	0,232	197,1
8	09H20	13H00	3,67	220	15,2	0,253	204,8
9	09H20	12H48	3,47	208	17,2	0,287	190,8
10	09H20	12H47	3,45	207	16,5	0,275	190,5
11	09H20	12H50	3,50	210	16,2	0,270	193,8
12	09H20	12H49	3,48	209	15,8	0,263	193,2
13	09H20	12H42	3,37	202	14,7	0,245	187,3
14	09H20	13H01	3,68	221	14,58	0,243	206,42
15	09H20	13H05	3,75	225	13,7	0,228	211,3
16	09H20	12H58	3,63	218	14,6	0,243	203,4
17	09H20	13H07	3,78	227	15,3	0,255	211,7
18	09H20	12H59	3,65	219	15,9	0,265	203,1
19	09H20	13H12	3,87	232	16,3	0,272	215,7
20	09H20	12H51	3,52	211	16,9	0,282	194,1
21	09H20	12H55	3,58	215	14,6	0,243	200,4
22	09H20	12H48	3,47	208	13,4	0,223	194,6

Aplicación de estadística descriptiva y test de Shapiro Wilk a las muestras.

Descriptive Statistics	
<i>DIFERENCIA</i>	
Mean	199,606875
Standard Error	1,48458167
Median	198,7

Mode	198,7
Standard Deviation	8,39806216
Sample Variance	70,527448
Kurtosis	-0,79990109
Skewness	0,0829386
Range	32,5
Maximum	215,7
Minimum	183,2
Sum	6387,42
Count	32
Geometric Mean	199,435863
Harmonic Mean	199,265034
AAD	7,10730469
MAD	5,45
IQR	12,275

Prueba t para medias de dos muestras emparejadas

Shapiro-Wilk Test	
<i>DIFERENCIA</i>	
W-stat	0,97262461
p-value	0,57477848
alpha	0,05
normal	yes
d'Agostino-Pearson	
DA-stat	1,51631723
p-value	0,46852837
alpha	0,05
normal	yes

Elaboración del calendario de riego en el navegador Edge

	<i>Variable 1</i>	<i>Variable 2</i>	<i>Antes</i>		<i>Después</i>	
<i>Media</i>	214,9375	15,330625				
<i>Varianza</i>	67,41532258	1,392206048	<i>Media</i>	214,9375	<i>Media</i>	15,330625
<i>Observaciones</i>	32	32	<i>Error típico</i>	1,451457485	<i>Error típico</i>	0,20858197
<i>Coefficiente de correlación de Pearson</i>	-0,088766076		<i>Mediana</i>	214	<i>Mediana</i>	15,45
<i>Diferencia hipotética de las medias</i>	0		<i>Moda</i>	208	<i>Moda</i>	14,6
<i>Grados de libertad</i>	31		<i>Desviación estándar</i>	8,210683442	<i>Desviación estándar</i>	1,17991781
<i>Estadístico t</i>	134,4532796		<i>Varianza de la muestra</i>	67,41532258	<i>Varianza de la muestra</i>	1,39220605
<i>P(T<=t) una cola</i>	9,35268E-45		<i>Curtosis</i>	-0,765845657	<i>Curtosis</i>	-0,77711488
<i>Valor crítico de t (una cola)</i>	1,695518783		<i>Coefficiente de asimetría</i>	0,205127807	<i>Coefficiente de asimetría</i>	-0,16054129
<i>P(T<=t) dos colas</i>	1,87054E-44		<i>Rango</i>	32	<i>Rango</i>	4,4

Valor crítico de t (dos colas)	2,039513446		Mínimo	200	Mínimo	12,9
			Máximo	232	Máximo	17,3
			Suma	6878	Suma	490,58
			Cuenta	32	Cuenta	32

El proceso de la elaboración del calendario de riego ejecutado en el navegador Edge utiliza una cantidad de 123.9MB de memoria RAM y el 0.1% del procesador, como se puede observar en el siguiente gráfico.

Gráfico 6-3: Valores de recursos utilizados en el navegador Edge.

Nombre	Estado	CPU	Memoria	Disco	Red	GPU	Motor de GPU	Consumo de e...	Tendencia de ...
Aplicaciones (6)									
> Administrador de tareas		2,0%	25,2 MB	0 MB/s	0 Mbps	0%		Muy baja	Muy baja
> Microsoft Edge (9)	🟢	0%	123,9 MB	0 MB/s	0 Mbps	0%	GPU 0 - 3D	Muy baja	Muy baja

Realizado por: (Torres, 2020)

Elaboración del calendario de riego en el navegador Opera

El proceso de la elaboración del calendario de riego ejecutado en el navegador Edge utiliza una cantidad de 123.9MB de memoria RAM y el 0.1% del procesador, como se puede observar en el siguiente gráfico.

Gráfico 6-3: Valores de recursos utilizados en el navegador Edge.

Nombre	Estado	CPU	Memoria	Disco	Red	GPU	Motor de GPU	Consumo de e...	Tendencia de ...
Aplicaciones (6)									
> Administrador de tareas		2,0%	25,2 MB	0 MB/s	0 Mbps	0%		Muy baja	Muy baja
> Microsoft Edge (9)	🟢	0%	123,9 MB	0 MB/s	0 Mbps	0%	GPU 0 - 3D	Muy baja	Muy baja

Realizado por: (Torres, 2020)

Elaboración del calendario de riego en el navegador Chrome

El proceso de la elaboración del calendario de riego ejecutado en el navegador Edge utiliza una cantidad de 123.9MB de memoria RAM y el 0.1% del procesador, como se puede observar en el siguiente gráfico.

Gráfico 6-3: Valores de recursos utilizados en el navegador Edge.

Nombre	Estado	CPU	Memoria	Disco	Red	GPU	Motor de GPU	Consumo de e...	Tendencia de ...
Aplicaciones (6)									
> Administrador de tareas		2,0%	25,2 MB	0 MB/s	0 Mbps	0%		Muy baja	Muy baja
> Microsoft Edge (9)	🟢	0%	123,9 MB	0 MB/s	0 Mbps	0%	GPU 0 - 3D	Muy baja	Muy baja

Realizado por: (Torres, 2020)

Elaboración del calendario de riego en el navegador Internet Explorer

El proceso de la elaboración del calendario de riego ejecutado en el navegador Edge utiliza una cantidad de 123.9MB de memoria RAM y el 0.1% del procesador, como se puede observar en el siguiente gráfico.

Gráfico 6-3: Valores de recursos utilizados en el navegador Edge.

Nombre	Estado	CPU	Memoria	Disco	Red	GPU	Motor de GPU	Consumo de e...	Tendencia de ...
Aplicaciones (6)									
> Administrador de tareas		2,0%	25,2 MB	0 MB/s	0 Mbps	0%		Muy baja	Muy baja
> Microsoft Edge (9)	⊕	0%	123,9 MB	0 MB/s	0 Mbps	0%	GPU 0 - 3D	Muy baja	Muy baja

Realizado por: (Torres, 2020)

Elaboración del calendario de riego en el navegador Mozilla Firefox

El proceso de la elaboración del calendario de riego ejecutado en el navegador Edge utiliza una cantidad de 123.9MB de memoria RAM y el 0.1% del procesador, como se puede observar en el siguiente gráfico.

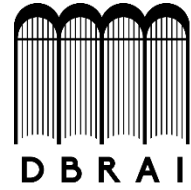
Gráfico 6-3: Valores de recursos utilizados en el navegador Edge.

Nombre	Estado	CPU	Memoria	Disco	Red	GPU	Motor de GPU	Consumo de e...	Tendencia de ...
Aplicaciones (6)									
> Administrador de tareas		2,0%	25,2 MB	0 MB/s	0 Mbps	0%		Muy baja	Muy baja
> Microsoft Edge (9)	⊕	0%	123,9 MB	0 MB/s	0 Mbps	0%	GPU 0 - 3D	Muy baja	Muy baja

Realizado por: (Torres, 2020)




**ESCUELA SUPERIOR POLITÉCNICA DE
CHIMBORAZO**



**DIRECCIÓN DE BIBLIOTECAS Y RECURSOS
PARA EL APRENDIZAJE Y LA INVESTIGACIÓN**

**UNIDAD DE PROCESOS TÉCNICOS
REVISIÓN DE NORMAS TÉCNICAS, RESUMEN Y BIBLIOGRAFÍA**

Fecha de entrega: 11 / 10 / 2020

INFORMACIÓN DEL AUTOR	
Nombres – Apellidos: ISAAC DAVID TORRES PAREDES	
INFORMACIÓN INSTITUCIONAL	
Facultad: INFORMÁTICA Y ELECTRÓNICA	
Carrera: INGENIERÍA EN SISTEMAS	
Título a optar: INGENIERO EN SISTEMAS INFORMÁTICOS	
f. Analista de Biblioteca responsable:	



0293-DBRAI-UPT-2020