

**INSTITUTO NACIONAL DE COMERCIO
FEDERICO ALVAREZ PLATA – NOCTURNO**



**SISTEMA DE MONITOREO Y CONTROL DE UN
INVERNADERO A TRAVÉS DE UNA APLICACIÓN WEB**

**PROYECTO DE GRADO PARA OPTAR
EL TÍTULO DE TÉCNICO SUPERIOR EN
SISTEMAS INFORMÁTICOS**

**ALBARADO SÁNCHEZ EMILIANA MIRNA
ENCALADA CAMARGO JOSÉ CRISTIAN**

TUTOR: ING. PAVEL FRANZ CÁCERES TORREZ

COCHABAMBA –BOLIVIA

Diciembre, 2017

AGRADECIMIENTOS

Un especial agradecimiento a nuestro tutor Ingeniero Pavel Franz Cáceres Torrez, por la excelente enseñanza y apoyo brindado.

Al Instituto Nacional de Comercio Federico Álvarez Plata – Nocturno, por la oportunidad de obtener conocimientos muy valiosos.

Al Ingeniero Jorge Villarroel Salvatierra por la calidad de enseñanza y paciencia durante nuestra permanencia en el instituto.

A los Ingenieros: David Escalera, Edson Flores, Raúl Vera, Rodrigo Fontanilla, Joel Alanez y todo el grupo de docentes por brindarnos sus conocimientos para nuestra formación académica.

¡MUCHAS GRACIAS!

RESUMEN

El presente documento muestra el análisis, diseño e implementación de un “Sistema de Control y Monitoreo a través de una Aplicación Web”.

Un invernadero ayuda a mejorar la producción de plantas o cultivos brindando un control del ambiente para aumentar la calidad de los productos. Son utilizados en zonas donde las condiciones climáticas son muy variadas y no favorecen los cultivos al aire libre.

El proyecto está dirigido a propietarios de invernaderos y agricultores, tiene por objetivo brindar un servicio que permita al usuario controlar y monitorear en tiempo real la temperatura, humedad relativa e iluminación dentro del invernadero desde cualquier dispositivo móvil o computador con acceso a internet, además de obtener reportes de las lecturas realizadas.

La metodología utilizada para el desarrollo del sistema fue SCRUM. Es un proceso en el que se aplican de manera regular un conjunto de buenas prácticas para trabajar colaborativamente, en equipo, y obtener el mejor resultado posible de un proyecto.

En la etapa inicial del proyecto se realizó el análisis de las necesidades de los usuarios, identificando los requerimientos tanto funcionales como no funcionales del sistema.

Durante la segunda etapa se realizó el diseño del software, para posteriormente definir los diferentes módulos del sistema, además de la construcción de la estructura física del prototipo de invernadero.

La tercera etapa fue el desarrollo del proyecto, durante el cual para el control se utilizó la placa de desarrollo Raspberry Pi Modelo B V1.1, la cual, a través de los puertos de entrada y salida que posee permitió activar o desactivar los diferentes componentes para controlar el ambiente, además de trabajar como servidor del sistema. Se utilizó el lenguaje de programación PHP para el desarrollo de la página web, el motor de base de datos MySQL, y en la parte del servidor se utilizó el lenguaje Phyton, para la programación de sensores para la obtención de datos.

Finalmente se realizaron las pruebas para evaluar la calidad del software y hardware con la ayuda de distintas herramientas.

El proyecto cumplió con el objetivo de contar con un Sistema de Control y Monitoreo a través de una Aplicación Web, su diseño consideró la eficiencia, la reducción de costos y la correcta selección de componentes, brindando una herramienta de gran ayuda para el sector de la agricultura.

ABSTRACT

This document shows the analysis, design and implementation of a "Control and Monitoring System through a Web Application".

A greenhouse helps to improve the production of plants or crops by providing control of the environment to increase the quality of the products. They are used in areas where the climatic conditions are very varied and do not favor outdoor crops.

The project is aimed at owners of greenhouses and farmers, aims to provide a service that allows the user to control and monitor in real time the temperature, relative humidity and lighting inside the greenhouse from any mobile device or computer with Internet access, in addition to obtain reports of the readings made.

The methodology used for the development of the system was SCRUM. It is a process in which a set of good practices are applied on a regular basis to work collaboratively, as a team, and obtain the best possible result from a project.

In the initial stage of the project, the analysis of the needs of the users was carried out, identifying both the functional and non-functional requirements of the system.

During the second stage the software design was carried out, to later define the different modules of the system, in addition to the construction of the physical structure of the greenhouse prototype.

The third stage was the development of the project, during which the Raspberry Pi Model B V1.1 development board was used for the control, which, through its input and output ports, enabled the different components to be activated or deactivated. to control the environment, in addition to working as a server of the system. The PHP programming language was used for the development of the web page, the MySQL database engine, and in the server part the Phyton language was used for the programming of sensors to obtain data.

Finally tests were carried out to evaluate the quality of software and hardware with the help of different tools.

The project fulfilled the objective of having a Control and Monitoring System through a Web Application, its design considered efficiency, cost reduction and the correct selection of components, providing a tool of great help for the sector of the farming.

ÍNDICE GENERAL

1	PLANTEAMIENTO DEL PROBLEMA.....	1
1.1	TEMA.....	1
1.2	DIAGNOSTICO Y JUSTIFICACION.....	1
1.2.1	Diagnóstico.....	1
1.2.2	Justificación.....	2
1.2.2.1	Justificación Social.....	2
1.2.2.2	Justificación Económica.....	2
1.2.2.3	Justificación Técnica.....	2
1.3	PLANTEAMIENTO DEL PROBLEMA.....	3
1.4	OBJETIVOS.....	3
1.4.1	OBJETIVO GENERAL.....	3
1.4.2	OBJETIVOS ESPECÍFICOS.....	3
1.5	ALCANCE.....	3
1.5.1	Prototipo del cultivo hidropónico.....	4
1.5.2	Módulo de Gestión de Usuarios.....	4
1.5.3	Módulo Web de Monitoreo.....	4
1.5.4	Módulo Web de Control.....	4
1.5.5	Módulo de reportes.....	4
1.6	Metodología y herramientas.....	5
1.6.1	Metodología.....	5
1.6.2	Fases de la metodología SCRUM.....	5
1.6.2.1	Recogida de Requisitos.....	6
1.6.2.2	Gestión del Backlog.....	6
1.6.2.3	Sprint Planning Meeting.....	6
1.6.2.4	Ejecución de sprint.....	7
1.6.2.5	Inspección e iteración.....	7
1.6.3	Herramientas.....	8
1.6.3.1	Hardware.....	8
1.6.3.2	Software.....	9
2	MARCO TEORICO.....	6

2.1 INVERNADERO.....	6
2.2 LA HIDROPONIA.....	6
2.2.1 Perspectivas y futuro de la hidroponía.....	7
2.2.2 Ventajas y desventajas del cultivo hidropónico.....	8
2.2.2.1 Ventajas del cultivo Hidropónico.....	8
2.2.2.2 Desventajas del cultivo Hidropónico.....	8
2.2.3 Tipos de sistemas hidropónicos.....	8
2.2.4 Sistema NFT.....	9
2.2.4.1 Ventajas del Sistema NFT.....	9
2.2.4.2 Desventajas del Sistema NFT.....	9
2.2.4.3 Componentes del sistema NFT.....	10
2.2.4.4 Solución nutritiva en Sistemas Hidropónicos.....	11
2.2.5 Sistemas NFT implementados en Bolivia.....	12
2.3 HERRAMIENTAS PARA EL DESARROLLO DEL SISTEMA.....	13
2.3.1 Componentes Lógicos (Software).....	13
2.3.1.1 Sistema de información (SI).....	13
2.3.1.2 Sistema informático.....	13
2.3.1.3 Base de datos (DB).....	14
2.3.1.4 SCRUM – Metodología Ágil de Desarrollo.....	15
2.3.1.5 Aplicación Web.....	16
2.3.1.6 Python.....	17
2.3.1.7 PHP.....	18
2.3.1.8 HTML (<i>HyperText Markup Language</i>).....	18
2.3.1.9 Bootstrap.....	19
2.3.1.10 HighCharts.....	19
2.3.2 Componentes físicos (Hardware).....	19
2.3.2.1 Hardware de Adquisición de datos (DAQ).....	19
2.3.2.2 Sensor.....	20
2.3.2.3 Sensor de Intensidad Luminosa LDR.....	21
2.3.2.4 Sensor de Temperatura y Humedad DHT11.....	22
2.3.2.5 Actuador.....	23
2.3.2.6 Placa Arduino.....	23
2.3.2.7 Raspberry Pi.....	24
2.3.2.8 NodeMCU.....	28
3 DESARROLLO DEL PROYECTO.....	30

<i>3.1 PLAN DE TRABAJO</i>	30
<i>3.2 PLANEACIÓN DEL PROCESO SCRUM</i>	30
3.2.1 Definición del Equipo.....	31
<i>3.3 ARTEFACTOS</i>	31
3.3.1 Definición de Objetivos del Producto - Product Backlog.....	31
3.3.2 Definición de Objetivos de las Iteraciones - Sprint Backlog.....	33
3.3.3 Lista de Tareas de Sprints (1 al 4).....	35
3.3.4 Planeación de las iteraciones – Sprint Planning.....	37
3.3.5 Definición de Terminado.....	37
3.3.6 Burndown Charts.....	37
3.3.7 Burndown Chart – Product Backlog.....	37
<i>3.4 REUNIÓN DE PLANIFICACIÓN DEL SPRINT 1 - 2017/06/05</i>	38
3.4.1 Asistentes.....	38
3.4.2 ¿Qué se completará en este Sprint? (Sprint Backlog).....	38
3.4.2.1 Actores del Sistema.....	38
3.4.2.2 Diagrama de Casos de Uso.....	39
3.4.2.3 Refinamiento de Casos de Uso.....	40
Caso de Uso Gestionar Usuarios.....	40
Caso de Uso Controlar Dispositivos.....	40
3.4.2.4 Diccionario de Casos de Uso.....	41
3.4.3 Diagrama de Clases.....	46
3.4.3.1 Requerimientos Funcionales del Sistema.....	47
3.4.3.2 Requerimientos No Funcionales del Sistema.....	48
3.4.4 Diagrama de Capas del Sistema.....	49
3.4.5 Diagrama de Despliegue.....	49
3.4.6 Burndown Chart Sprint 1.....	50
<i>3.5 REUNIÓN DE PLANIFICACIÓN DEL SPRINT 2 - 2017/06/21</i>	51
3.5.1 Asistentes.....	51
3.5.2 ¿Qué se completará en este Sprint? (Sprint Backlog).....	51
3.5.2.1 Herramientas para el Desarrollo de Software.....	51
Motor de Base de Datos.....	51
3.5.2.2 Lenguaje de Programación de la Plataforma Web PHP.....	52
3.5.2.3 Dispositivo de Control y Almacenamiento de Datos - Raspberry Pi.....	52
3.5.2.4 Dispositivo de Adquisición de Datos - Arduino.....	55
3.5.2.5 Selección de Sensores.....	56

3.5.2.6	Construir el prototipo hidropónico.....	58
3.5.3	Burndown Chart Sprint 2.....	59
3.6	REUNIÓN DE PLANIFICACIÓN DEL SPRINT 3 - 2017/07/07.....	60
3.6.1	Asistentes.....	60
3.6.2	¿Qué se completará en este Sprint? (Sprint Backlog).....	60
3.6.2.1	Definir los roles de Usuario.....	60
3.6.2.2	Crear la Interfaz de Usuario (MVC).....	61
3.6.2.3	Crear la Interfaz de Administrador (MVC).....	61
3.6.2.4	Realizar la Validación de Usuarios.....	62
3.6.2.5	Realizar el Registro de Usuarios.....	63
3.6.2.6	Interfaz de Monitoreo de Variables Ambientales.....	64
3.6.3	Burndown Chart Sprint 3.....	64
3.7	REUNIÓN DE PLANIFICACIÓN DEL SPRINT 4 - 2017/07/25.....	65
3.7.1	Asistentes.....	65
3.7.2	¿Qué se completará en este Sprint? (Sprint Backlog).....	66
3.7.2.1	Instalación del SO Raspbian.....	66
3.7.2.2	Interfaz de Control Individual.....	67
3.7.2.3	Programación del Arduino para envío de variables ambientales.....	67
3.7.2.4	Reporte de Temperatura.....	68
3.7.2.5	Reporte de Humedad Relativa.....	69
3.7.2.6	Reporte de Iluminación.....	69
3.7.3	Burndown Chart Sprint 4.....	70
3.8	Arquitectura General del Sistema.....	33
3.9	33
3.9.1	Arquitectura del Hardware.....	72
3.9.2	Arquitectura del Software.....	73
Módulo de Gestión de Usuarios. Se tendrán los siguientes roles:.....	73	
Módulo de reportes. Se generará reporte de parámetros ambientales (temperatura, humedad relativa e iluminación).....	73	
3.9.3	Arquitectura Modelo Vista Controlador MVC.....	74
4	ANÁLISIS DE COSTOS.....	75
4.1	Recursos Humanos.....	75
4.2	Recursos Tecnológicos.....	75

4.2.1 Recursos de Hardware.....	75
4.2.2 Recursos de Software.....	76
4.3 <i>Material de Apoyo</i>	77
5 ANÁLISIS DE COSTOS PROYECTADOS A CINCO AÑOS.....	78
5.1 <i>Inversión Activo Fijo</i>	78
5.2 <i>Inversión Activo Diferido</i>	78
5.2.1 Amortización Activo Diferido.....	79
5.3 <i>Inversión Total</i>	79
5.4 <i>Costos Operativos</i>	79
5.4.1 Costo de Administración.....	79
5.4.2 Costo de Producción.....	79
5.4.3 Costo de Comercialización.....	80
5.5 <i>Estado de Resultados Proyectados para cinco años</i>	80
5.5.1 Esquema para cálculo de Rentabilidad.....	81
6 CONCLUSIONES Y RECOMENDACIONES.....	76
6.1 <i>CONCLUSIONES</i>	76
6.2 <i>RECOMENDACIONES</i>	77
6.2.1 Recomendaciones con el Hardware.....	77
6.2.2 Recomendaciones a Futuro.....	77
7 BIBLIOGRAFIA.....	78
ANEXO A.....	80
INTERFACES DEL SISTEMA.....	80
Interfaz inicio sesión.....	80
Interfaz de Registro de Usuarios.....	80
Interfaz Principal Administrador.....	81
Interfaz Principal Usuario.....	81
Interfaz de Monitoreo de variables Ambientales.....	82
Interfaz de Control Rápido de Dispositivos.....	82
Interfaz de Control manual de Dispositivos.....	83
Interfaz de Reporte de Temperatura.....	83
Interfaz de Reporte de Iluminación.....	84

Interfaz de Reporte de Humedad Relativa.....	84
ANEXO B.....	85
<i>ACTAS DE REUNIÓN DE SPRINTS.....</i>	<i>85</i>
Acta Sprint 1.....	85
Acta Sprint 2.....	86
Acta Sprint 3.....	87
Acta Sprint 4.....	88

ÍNDICE DE FIGURAS

Figura 1.1 Fases de la Metodología Scrum.....	5
Figura 2.1 Invernadero Tradicional.....	6
Figura 2.2 Sistema NFT básico.....	10
Figura 2.3 Empresa Hidroponía Bolivia.....	12
Figura 2.4 Empresa Hidroponía Tiquipaya – Cochabamba.....	12
Figura 2.5 Sistemas de Gestión de Bases de Datos.....	14
Figura 2.6 Diagrama de bloques DAQ.....	20
Figura 2.7 Tipos de Sensores.....	21
Figura 2.8 Fotorresistencia o LDR.....	22
Figura 2.9 Sensor de Temperatura y Humedad DHT11.....	22
Figura 2.10 Arduino UNO Rev 3.....	24
Figura 2.11 Raspberry Pi Modelo B Versión 1.1.....	25
Figura 2.12 Raspberry Pi Modelo A.....	25
Figura 2.13 Raspberry Pi Modelo A +.....	26
Figura 2.14 Raspberry Pi Modelo B.....	26
Figura 2.15 Raspberry Pi Modelo B+.....	27
Figura 2.16 Raspberry Pi 2 Modelo B.....	27
Figura 2.17 Raspberry Pi 3 Modelo B.....	28
Figura 2.18 ESP8266 – NodeMCU.....	28
Figura 3.1 Product Backlog.....	31
Figura 3.2 Sprint 1 Backlog.....	33
Figura 3.3 Sprint 2 Backlog.....	33
Figura 3.4 Sprint 3 Backlog.....	34
Figura 3.5 Sprint 4 Backlog.....	35
Figura 3.6 Burndown Chart – Product Backlog.....	37
Figura 3.7 Esquema de la Raspberry Pi Modelo 2 v1.1.....	54
Figura 3.8 Prototipo del Cultivo Hidropónico.....	58
Figura 3.9 Interfaz Web de Usuario.....	61
Figura 3.10 Interfaz Web de Administrador.....	61
Figura 3.11 Validación de Inicio de Sesión.....	62

Figura 3.12 Validación de Usuario y Contraseña.....	62
Figura 3.13 Interfaz Registro nuevo Usuario.....	63
Figura 3.14 Interfaz de Monitoreo de Variables Ambientales.....	64
Figura 3.15 SO. Raspbian instalado en la Raspberry Pi 2.....	66
Figura 3.16 Interfaz de Control Individual.....	67
Figura 3.17 Programación de Arduino Uno.....	68
Figura 3.18 Reporte de Temperatura.....	68
Figura 3.19 Interfaz de Control Individual.....	69
Figura 3.20 Interfaz de Control Individual.....	69
Figura 3.21 Arquitectura General del Sistema.....	33

ÍNDICE DE TABLAS

Tabla 1.1 Recursos de Hardware Utilizados.....	9
Tabla 1.2 Recursos de Software Utilizados.....	10
Tabla 3.4 Plan de Trabajo.....	30
Tabla 3.5 Equipo de trabajo de Scrum.....	31
Tabla 3.6 Product Backlog.....	32
Tabla 3.7 Lista de Tareas de Sprints.....	35
Tabla 3.8 Planeación de Iteraciones – Sprint Planning.....	37
Tabla 3.9 Caso de Uso Registrar Usuario.....	41
Tabla 3.10 Caso de Uso Eliminar Usuario.....	42
Tabla 3.11 Caso de Uso Iniciar Sesión.....	42
Tabla 3.12 Caso de Uso Encender Bomba de Agua.....	43
Tabla 3.13 Caso de Uso Apagar Bomba de Agua.....	43
Tabla 3.14 Caso de Uso Encender Luces.....	44
Tabla 3.15 Caso de Uso Apagar Luces.....	44
Tabla 3.16 Caso de Uso Encender Ventilador.....	45
Tabla 3.17 Caso de Uso Apagar Ventilador.....	45
Tabla 3.18 Caso de Uso Encender Calefactor.....	46
Tabla 3.19 Caso de Uso Apagar Calefactor.....	46
Tabla 3.20 Requerimientos Funcionales del Sistema.....	48
Tabla 3.21 Tabla Burndown Chart – Sprint 1.....	51
Tabla 3.22 Comparativa de Modelos Raspberry Pi.....	54
Tabla 3.23 Especificaciones Raspberry Pi Modelo B.....	55

Tabla 3.24 Comparativa de placas Arduino.....	56
Tabla 3.25 Características de la placa Arduino UNO.....	57
Tabla 3.26 Características del sensor LDR.....	58
Tabla 3.27 Comparativa de Sensores de Humedad.....	58
Tabla 3.28 Características del Sensor DHT11.....	59
Tabla 3.29 Tabla Burndown Chart – Sprint 2.....	60
Tabla 3.30 Tabla Burndown Chart – Sprint 3.....	66
Tabla 3.31 Tabla Burndown Chart – Sprint 4.....	72
Tabla 4.32 Costo Recursos Humanos.....	75
Tabla 4.33 Costo Recursos de Hardware.....	76
Tabla 4.34 Recursos de Software.....	77

ÍNDICE DE DIAGRAMAS

Diagrama 2.1 Elementos de un Sistema Informático.....	13
Diagrama 3.1 Diagrama General de Casos de Uso del Sistema.....	39
Diagrama 3.2 Caso de Uso Gestionar Usuarios.....	40
Diagrama 3.3 Caso de Uso Controlar Dispositivos.....	40
Diagrama 3.4 Diagrama de Clases.....	46
Diagrama 3.5 Diagrama de Capas.....	49
Diagrama 3.6 Diagrama de Despliegue.....	49
Diagrama 3.7 Burndown Chart – Sprint 1.....	50
Diagrama 3.8 Burndown Chart – Sprint 2.....	59
Diagrama 3.9 Burndown Chart – Sprint 3.....	65
Diagrama 3.10 Burndown Chart – Sprint 4.....	70
Diagrama 3.11 Arquitectura General del Hardware.....	72
Diagrama 3.12 Subsistemas de Hardware del Sistema.....	72
Diagrama 3.13 Arquitectura MVC.....	74

INTRODUCCION

CAPÍTULO I

**PLANTEAMIENTO DEL
PROBLEMA**

1 PLANTEAMIENTO DEL PROBLEMA

1.1 TEMA

Sistema de monitoreo y control de un invernadero a través de una aplicación web.

1.2 DIAGNOSTICO Y JUSTIFICACION

1.2.1 Diagnóstico

En la gestión de un invernadero, existen actividades como el mantenimiento general, el riego y la atemperación del ambiente, tareas que se dificultan por diversos motivos, tales como el tamaño del invernadero o la necesidad de manipulación in-situ de la ventilación y calefacción, estos factores combinados con el desperdicio de energía y agua hacen necesaria la creación de un sistema eficiente y económicamente viable que asegure una adecuada gestión de recursos.

La Hidroponía es una metodología que permite el cultivo de plantas sin tierra mucho anterior al cultivo en tierra.

Actualmente en Bolivia, este sistema es utilizado por algunos productores, tal es el caso de la empresa "Hidroponía Bolivia" e "Hidroponía Tiquipaya", ubicadas en el departamento de Cochabamba.

Los factores principales para incursionar en el área de la hidroponía en Bolivia son:

- Carencia de productos frescos para el consumo de la población, puesto que gran parte de la producción agrícola en Bolivia es dada por la temporalidad, que ha sido afectada por los constantes cambios climáticos.
- Aprovechamiento de espacios para el cultivo y escases de agua en varios departamentos del país.
- Deficiencias en el control de la seguridad alimentaria.

Hoy en día la integración con la tecnología da una variedad infinita de posibilidades para facilitar las actividades de gestión de un cultivo hidropónico, como por ejemplo realizar el monitoreo y control de un cultivo que se encuentre fuera del alcance de una persona.

El uso de la tecnología es importante para el desarrollo de la agricultura, por ello, países desarrollados como Japón, Alaska, Corea entre otros, utilizan sistemas automatizados para el control de los procesos de cultivo, obteniendo un ambiente óptimo y maximizando la producción, realizando el reemplazo por ejemplo de la luz solar por iluminación LED en algunos cultivos.

En tal sentido, se plantea la pregunta que motiva el presente proyecto: ¿Se puede integrar una tecnología móvil con una microcomputadora Raspberry Pi 2 modelo B capaz de activar automáticamente los mecanismos de riego, calefacción, ventilación e iluminación para mantener un ambiente óptimo para cultivos dentro de un invernadero, monitorearlo de forma remota y ejecutar acciones en tiempo real?

1.2.2 Justificación

1.2.2.1 Justificación Social

El sistema está orientado a fomentar el cultivo de alimentos de necesidad común, tales como vegetales y otros, los cuales son elementos básicos en la alimentación de la población en general.

Los usuarios contarán con una herramienta que facilitará la gestión de algunas tareas manuales y la ejecución de las mismas en los cultivos.

1.2.2.2 Justificación Económica

El sistema será relativamente económico siendo accesible a todo público interesado en el área de la agricultura.

Se utilizará el material exacto necesario para las plantas, evitando desperdicios sobretodo en el recurso hídrico, además del ahorro de tiempo, ya que el sistema

tendrá un control sobre las variables más importantes para el cultivo de las plantas, como la temperatura, la humedad relativa e iluminación.

1.2.2.3 Justificación Técnica

El sistema permitirá monitorear las variables ambientales del cultivo, para toma de decisiones, evitando algún incidente por falta de control.

Se podrán obtener productos fuera de temporada, por medio de la aplicación de tecnologías de control y automatización.

1.3 PLANTEAMIENTO DEL PROBLEMA

En la actualidad, en Bolivia los invernaderos existentes con cultivos tradicionales e hidropónicos no cuentan con un control y monitoreo del cultivo, que permita optimizar la productividad y mejorar la calidad de los productos.

1.4 OBJETIVOS

1.4.1 OBJETIVO GENERAL

- ✓ Desarrollar un software que permita el control y monitoreo de un cultivo hidropónico mediano de lechuga para agricultores a través de una aplicación web.

1.4.2 OBJETIVOS ESPECÍFICOS

- ✓ Seleccionar los Dispositivos Electrónicos (hardware).
- ✓ Diseñar e implementar el Prototipo del Invernadero.
- ✓ Desarrollar un Módulo de Gestión de Usuarios.
- ✓ Desarrollar un Módulo Web de Monitoreo
- ✓ Desarrollar un Módulo Web de Control.
- ✓ Desarrollar un Módulo Web de Reportes.

1.5 ALCANCE

El sistema estará destinado al área agrícola, en nuestro caso para el cultivo hidropónico de lechuga en un invernadero mediano, siendo accesible remotamente por el usuario para realizar el monitoreo y control del cultivo.

Se realizará un estudio de las tecnologías existentes disponibles en el mercado para el área de automatización, priorizando el aspecto técnico y económico de los componentes.

Se realizará la selección de dispositivos, en función del alcance del proyecto, según catálogos de fabricante (motor, contactor, disyuntor y temporizador), sin olvidar el componente para el almacenamiento de datos y control a distancia (Raspberry Pi).

Se contará con los siguientes módulos descritos a continuación:

1.5.1 Prototipo del cultivo hidropónico.

El prototipo del Invernadero hidropónico, tendrá capacidad para un cultivo de 41 lechugas, ocupando un espacio de 1.3 [m] de alto por 0.93 [m] de largo y 0.6 [m] de profundidad.

1.5.2 Módulo de Gestión de Usuarios.

Se tendrán los siguientes roles:

- o **Administrador:** Podrá crear, visualizar y dar de baja a otros usuarios del sistema.
- o **Usuario:** Tendrá acceso a la información y datos del sistema, además de poder controlar los dispositivos de forma remota.

1.5.3 Módulo Web de Monitoreo.

Se visualizará los parámetros ambientales como temperatura, humedad relativa e iluminación en tiempo real a través de la aplicación web.

1.5.4 Módulo Web de Control.

Se realizará el control de los dispositivos (motor, ventilador, calefactor y luces Led) a través de una página web accesible desde cualquier lugar con conexión a internet.

1.5.5 Módulo de reportes.

Se generarán reportes de los parámetros ambientales (temperatura, humedad relativa e iluminación) según requerimientos del usuario.

Se mostrarán los datos de temperatura, humedad relativa e iluminación en una tabla, recuperados del servidor, para información del usuario.

1.6 Metodología y herramientas

1.6.1 Metodología

La metodología seleccionada para el desarrollo del proyecto es SCRUM. Es una metodología en la que se aplican de manera regular un conjunto de buenas prácticas para trabajar colaborativamente, en equipo, y obtener el mejor resultado posible de un proyecto. Con esta forma de trabajo se realizan entregas parciales y regulares del producto final, priorizadas por el beneficio que aportan al receptor del proyecto.¹

1.6.2 Fases de la metodología SCRUM

SCRUM considera cinco fases de trabajo. Todas estas etapas están definidas por tiempos máximos de ejecución y las reuniones se cronometran para no extenderlas innecesariamente. De esta manera se garantiza que funcione como una metodología ágil.

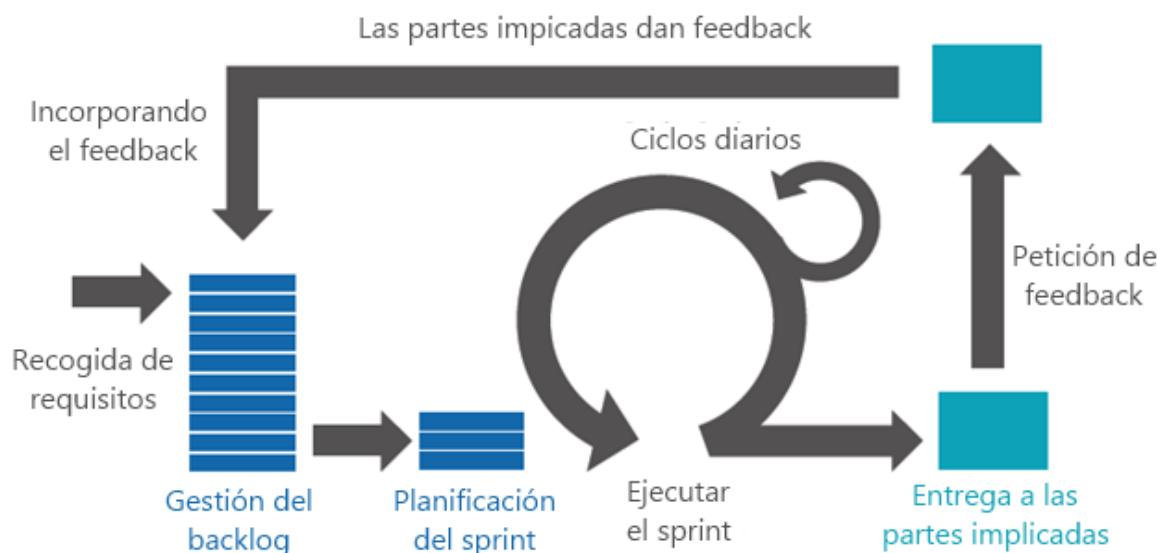


Figura 1.1 Fases de la Metodología Scrum

Fuente: www.ida.cl/blog/estrategia-digital/metodologia-scrum-en-proyectos-digitales/, 2017

¹ proyectosagiles.org, 2017

Las fases de la metodología Scrum son las siguientes:

1.6.2.1 Recogida de Requisitos

El proceso comienza con la generación de la lista de objetivos o requisitos priorizada, que **actúa como plan del proyecto** y que es entregada por el cliente o dueño del producto al equipo. La lista de objetivos/requisitos priorizada representa la visión y expectativas del cliente respecto a los objetivos y entregas del producto o proyecto. Es importante comprender que el cliente es el responsable de crear y gestionar la lista con ayuda del líder del proceso, el **Scrum Master**, que es el director del proyecto y encargado de eliminar los obstáculos que impiden que el equipo de desarrollo alcance el objetivo del sprint.

Esta etapa sería la “planificación” del proyecto, en un marco no ágil de trabajo.

1.6.2.2 Gestión del Backlog

Es el conjunto de funcionalidades y tareas a realizar. Para cada objetivo/requisito se indica el valor que aporta al cliente y el costo estimado de completarlo, velando por un equilibrio entre ambos en pos del Retorno de Inversión (ROI).

1.6.2.3 Sprint Planning Meeting

Un **sprint** es una unidad de trabajo que agrupa un conjunto de tareas en un periodo de tiempo. La primera iteración es de planificación y está compuesta por dos partes:

- **Selección de requisitos:** Es la iteración entre cliente y equipo, el momento en que el equipo pregunta al cliente las dudas que surgen y se seleccionan los requisitos más prioritarios que se comprometen a completar en la iteración. Tiene una duración máxima de cuatro horas.
- **Planificación de la iteración:** Se elabora la lista de tareas o acciones necesarias para desarrollar los requisitos a los que se han comprometido. La estimación de esfuerzo se hace de manera conjunta, siempre con el scrum master como facilitador, y los miembros del equipo se autoasignan las tareas. La duración de este ejercicio no debe superar las cuatro horas.

1.6.2.4 Ejecución de sprint

En la metodología Scrum un proyecto se ejecuta en bloques temporales cortos y fijos, llamados sprint, que son iteraciones de 2 semanas. Si se sobrepasa este tiempo, como máximo un sprint puede tomar 4 semanas.

Daily Scrum Meeting: Todos los días, una vez comenzado el sprint, el equipo realiza una reunión de coordinación. En estas sesiones diarias, cada miembro del equipo revisa el trabajo que el resto está realizando.

En la reunión cada integrante debe responder a tres preguntas:

- **¿Qué he hecho desde la última reunión de sincronización?**
- **¿Qué voy a hacer a partir de este momento?**
- **¿Qué impedimentos tengo o voy a tener?**

Estas reuniones son fundamentales en el proceso, ya que son instancias para avanzar desde los procesos individuales que desarrolla cada miembro del equipo a la colaboración de todos en el desarrollo.

1.6.2.5 Inspección e iteración

El último día de la iteración se realiza la reunión de revisión de la iteración, y se compone de dos partes:

- **Sprint Review:** El equipo desarrollador presenta al cliente los requisitos completados en la iteración, en forma de incremento de producto preparado para ser entregado. El cliente revisa el entregable y se adaptan las mejoras necesarias.
- **Sprint Retrospective:** En esta fase el equipo analiza cómo ha sido su manera de trabajar y cuáles son los problemas que podrían impedirle progresar adecuadamente, enfocando el proceso a la mejora continua del equipo.

Todas las instancias de reunión se deben cronometrar y respetar en el marco de tiempos establecidos. Esta variable es fundamental para mantener los esfuerzos enfocados en el desarrollo del producto.

1.6.3 Herramientas

Para el desarrollo del sistema se utilizarán las siguientes herramientas:

1.6.3.1 Hardware

Para el desarrollo del sistema se utilizarán los recursos de hardware, mencionados en la Tabla 1.1 Recursos de Hardware Utilizados.

Tabla 1.1 Recursos de Hardware Utilizados

Item	Cantidad	Función
Raspberry Pi 2 Modelo B	1	Servidor
Arduino Uno Rev3	1	Envía datos ambientales para almacenamiento
NodeMCU	1	Envía datos ambientales para monitoreo
Sensor DHT11	3	Capta variables de temperatura y humedad relativa
Sensor LDR	3	Capta variable de intensidad luminosa
Calefactor	1	Eleva la temperatura del ambiente
Ventilador	1	Reduce la temperatura del ambiente
Bomba Hidráulica	1	Impulsa la solución nutritiva desde el tanque colector hasta la tubería de distribución del sistema.

Fuente: Elaboración Propia, 2017

1.6.3.2 Software

Para el desarrollo del sistema se utilizarán los recursos de software, mencionados en la Tabla 1 .2 Recursos de Software Utilizados.

Tabla 1.2 Recursos de Software Utilizados

Item	Cantidad	Tipo
Modelamiento (ArgoUml)	1	Software Libre
Elaboración de Diagramas EDraw Max 8.4	1	EDraw Max 8.4
Geany Text Editor	1	Software Libre
Programación: Lenguaje Html 5	1	Software Libre
Programación: Lenguaje PHP	1	Software Libre
Programación: Lenguaje Python	1	Software Libre
Motor de Base de Datos: MySql	1	Software Libre
IDE Arduino	1	Software Libre

Fuente: Elaboración Propia, 2017.

CAPÍTULO II

MARCO TEÓRICO

2 MARCO TEORICO

2.1 INVERNADERO

Un invernadero (o invernáculo) es un lugar cerrado, estático y accesible a pie, que se destina a la horticultura, dotado habitualmente de una cubierta exterior translúcida de vidrio o plástico, que permite el control de la temperatura, la humedad y otros factores ambientales para favorecer el desarrollo de las plantas.²



Figura 2.2 Invernadero Tradicional

Fuente: <http://visionagropecuaria.com/estudiantes-innovan-invernaderos-automaticos>, 2017

2.2 LA HIDROPONIA

La hidroponía o agricultura hidropónica es un método utilizado para cultivar plantas usando disoluciones minerales en vez de suelo agrícola. Un sistema hidropónico es un sistema aislado del suelo utilizado para cultivar diversos tipos de plantas.

El crecimiento de las plantas es posible por un suministro adecuado de todos sus requerimientos nutricionales a través del agua o solución nutritiva.³

² <http://fibras.com.mx/uncategorized/que-es-un-invernadero/>

³ (Rodriguez, 2001)

2.2.1 Perspectivas y futuro de la hidroponía

Hoy en día la hidroponía es el método más intensivo de producción hortícola; generalmente es de alta tecnología y de fuerte capital, y viene siendo aplicada exitosamente con fines comerciales en países desarrollados.

En los últimos diez años, el área mundial destinada a la producción hidropónica se ha incrementado notablemente. En 1996 el área mundial era de 12,000 hectáreas (ISOSC; Sociedad Internacional de Cultivo Sin Suelo) y, según las últimas proyecciones, habrían más de 35,000 hectáreas, de las cuales alrededor del 80% (24,000 hectáreas) son cultivadas sólo por 10 países (Holanda, España, Canadá, Francia, Japón, Israel, Bélgica, Alemania, Australia y Estados Unidos). Los únicos países latinoamericanos considerados como países hidropónicos son México y Brasil.⁴

El crecimiento futuro de la hidroponía en Latinoamérica dependerá mucho del desarrollo y adaptación de sistemas menos sofisticados de producción que sean competitivos en costos con respecto a la tecnología sofisticada generada en países desarrollados. Los cultivos hidropónicos más rentables son el tomate, el pepino, el pimiento, la lechuga y las flores cortadas.

Una considerable disminución de las áreas de tierras agrícolas en países en vías de desarrollo, hace de la hidroponía una interesante alternativa de producción en zonas urbanas y periurbanas. Dentro del contexto de la agricultura urbana, la hidroponía puede ser muy bien aplicada en las ciudades con tecnologías más sencillas y de bajo costo, principalmente en zonas de extrema pobreza, como una manera de incentivar el autoconsumo de hortalizas y de apoyar el ingreso familiar a través del autoempleo en las propias viviendas o en los centros comunales.⁵

2.2.2 Ventajas y desventajas del cultivo hidropónico

2.2.2.1 Ventajas del cultivo Hidropónico.

- Aprovechar las tierras o suelos no aptos para la agricultura tradicional.

⁴ (Carrasco, 1996)

⁵ (Rodriguez, 2001)

- Los rendimientos obtenidos con hidroponía superan significativamente a la producción tradicional en suelo.
- Menor consumo de agua y fertilizantes. La técnica es muy apropiada en zonas donde hay escasez de agua.
- Reducción en la aplicación de agroquímicos.
- No contamina el medio ambiente.
- Crecimiento más rápido y vigoroso de las plantas.

2.2.2.2 Desventajas del cultivo Hidropónico.

- Costo inicial elevado.
- Se requieren conocimientos de agricultura y sales nutritivas.
- El desconocimiento del sistema hidropónico apropiado para producir un determinado cultivo.
- La falta de experiencia en el manejo de las soluciones nutritivas puede alterar su composición y afectar la apariencia y calidad de las plantas.

2.2.3 Tipos de sistemas hidropónicos.

Los sistemas hidropónicos se pueden dividir en dos grupos:

- **Sistemas hidropónicos puros (en agua).** Son sistemas hidropónicos por excelencia; las raíces de las plantas están en contacto directo con la solución nutritiva.
- **Sistemas en agregados o sustratos.** En este tipo de sistemas, las raíces de las plantas crecen y desarrollan en sustratos inertes; la solución nutritiva fluye entre las partículas del sustrato humedeciendo las raíces.

2.2.4 Sistema NFT

La técnica de la solución nutritiva recirculante conocida como NFT (Nutrient Film Technique), fue desarrollada en la década del sesenta por el Dr. Allan Cooper, en Inglaterra.

Desde esa época, este sistema de cultivo, destinado principalmente a la producción de hortalizas de alta calidad, se ha desarrollado y difundido por un gran número de países, donde existen condiciones limitantes de suelo y un mercado promisorio para suplir con hortalizas frescas de alta calidad y sanidad.⁶

El sistema NFT se basa en el flujo permanente de una pequeña cantidad de solución a través de caños o tuberías de los que el cultivo toma los elementos necesarios para su nutrición.

2.2.4.1 Ventajas del Sistema NFT

La ventaja que destaca la técnica NFT en relación a otros sistemas hidropónicos, es la alta calidad obtenida de diferentes productos hortícolas, en un corto período de cultivo como también en rendimiento. La constante oferta de agua y elementos minerales permite a las plantas crecer sin estrés y obtener el potencial productivo del cultivo.⁷

2.2.4.2 Desventajas del Sistema NFT

Las desventajas de este sistema radican en el costo de uso de energía, la formulación y chequeo frecuente del pH y salinidad de la solución.

Si bien la principal desventaja de este sistema es alto costo de implementación, este resulta ser rentable considerando una inversión a largo plazo.

2.2.4.3 Componentes del sistema NFT

Los componentes del sistema NFT o Recirculante se muestran en la figura 2.4 y son los siguientes:

- Tanque de la Solución Nutritiva
- Electrobomba
- Canales de Cultivo (Tuberías de PVC)

⁶ (Rodríguez, 2001)

⁷ (J. C. Gilsanz, 2007)

- Solución Nutritiva

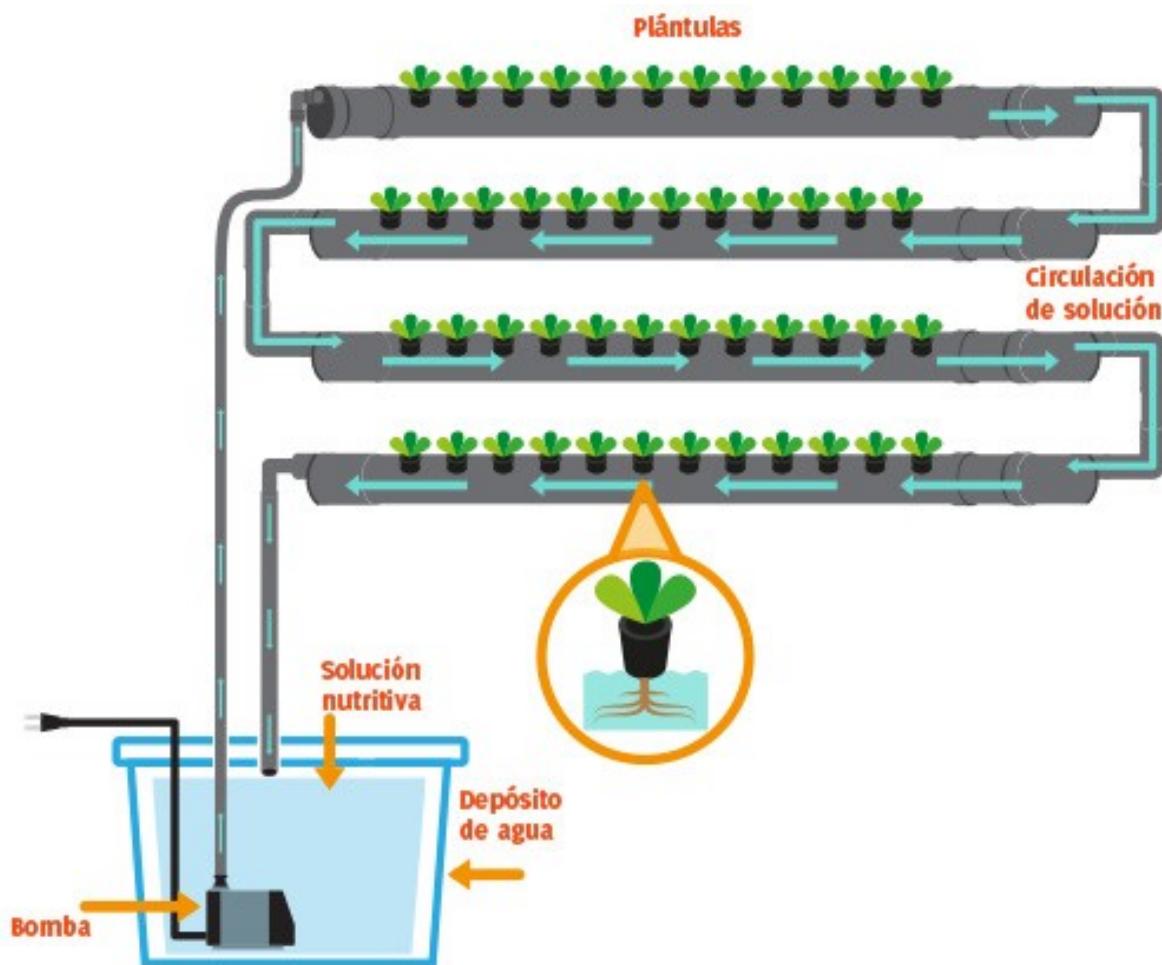


Figura 2.3 Sistema NFT básico

Fuente: <https://www.pinterest.es/pin/392516923754594075/>, 2017

2.2.4.4 Solución nutritiva en Sistemas Hidropónicos.

La solución nutritiva (SN), es la mezcla de agua con nutrientes minerales, que se añaden a través de fertilizantes comerciales, en cantidades y proporciones adecuadas, de manera que cubran las necesidades de las plantas para su crecimiento y desarrollo.⁸

⁸ (Rodríguez, 2001)

Entre los elementos esenciales de las soluciones nutritivas tenemos los macronutrientes: nitrógeno, fosforo, potasio, calcio, magnesio, azufre y los micronutrientes: boro, cloro, cobre, hierro, manganeso, molibdeno, zinc y níquel.

En la tabla 2.1 se muestran las sales fertilizantes utilizadas en hidroponía, que está diseñado para cubrir las necesidades de todos los elementos que se necesitan en el desarrollo de las plantas.⁹

Tabla 2.1 Sales fertilizantes utilizadas en hidroponía

Nombre químico	Fórmula química	Solubilidad(gr. / litro)
Nitrato de calcio	Ca(NO ₃) ₂	1220
Nitrato de potasio	KNO ₃	130
Nitrato de magnesio	Mg(NO ₃) ₂ , 6H ₂ O	279
Fosfato monopotásico	KH ₂ PO ₄	230
Sulfato de magnesio	MgSO ₄ , 6H ₂ O	710
Sulfato de potasio	K ₂ SO ₄	111
Sulfato de manganeso	MnSO ₄	980
Ácido bórico	H ₃ BO ₃	60
Sulfato de cobre	CuSO ₄ , 5H ₂ O	310
Sulfato de zinc	ZnSO ₄ , 7H ₂ O	960
Molibdato de amonio	(NH ₄) ₆ Mo ₇ O ₂₄ , 4H ₂ O	430

Fuente: Elaboración propia, 2017

2.2.5 Sistemas NFT implementados en Bolivia.

La eficiencia de los cultivos hidropónicos está generando un crecimiento acelerado de productores hidropónicos en nuestro país, donde el consumo de hortalizas ha crecido considerablemente principalmente debido al aumento de la calidad de vida de la población. A continuación, se muestran algunos de los productores hidropónicos más importantes de Bolivia, proveedores de grandes cadenas de supermercados:

⁹ (Favela, 2006)



Figura 2.4 Empresa Hidroponía Bolivia

Fuente: www.hidroponiabol.com, 2012



Figura 2.5 Empresa Hidroponía Tiquipaya – Cochabamba

Fuente: www.actiweb.es/hidrotiqui/, 2015

2.3 HERRAMIENTAS PARA EL DESARROLLO DEL SISTEMA

2.3.1 Componentes Lógicos (Software)

2.3.1.1 Sistema de información (SI).

“Un sistema de información es un conjunto de componentes interrelacionados que recolectan (o recuperan), procesan, almacenan y distribuyen información para apoyar los procesos de toma de decisiones y de control en una organización”.¹⁰

2.3.1.2 Sistema informático

La ISO ¹¹ define sistema informático como “El sistema compuesto de equipos y de personal pertinente que realiza funciones de entrada, proceso, almacenamiento, salida y control con el fin de llevar a cabo una secuencia de operaciones con datos”.

Un sistema informático está constituido por tres elementos: Hardware (recursos físicos), Software (recursos lógicos) y Usuarios (recursos humanos), como se muestra en el diagrama 2.1

Diagrama 2.1 Elementos de un Sistema Informático



Fuente: Elaboración propia

2.3.1.3 Base de datos (DB)

Una base de datos o banco de datos es un conjunto de datos pertenecientes a un mismo contexto y almacenados sistemáticamente para su posterior uso.

¹⁰ (Laudon, 2012)

¹¹ Organización Internacional de Normalización

Existen programas denominados sistemas gestores de bases de datos, abreviado SGBD (del inglés *Database Management System* o DBMS), que permiten almacenar y posteriormente acceder a los datos de forma rápida y estructurada.¹²

Existen gran cantidad de sistemas de gestión de bases de datos, cada una de ellas posee una forma diferente de manejar la información, algunos de ellos son: Oracle, MySQL, SQL Server, entre otras que se muestran en la figura 2.5



Figura 2.6 Sistemas de Gestión de Bases de Datos

Fuente: devcode.la/blog/que-es-sql/

2.3.1.4 SCRUM – Metodología Ágil de Desarrollo

Scrum es un proceso de la Metodología Ágil que se usa para minimizar los riesgos durante la realización de un proyecto, pero de manera colaborativa.

Entre las ventajas se encuentran la productividad, calidad y que se realiza un seguimiento diario de los avances del proyecto, logrando que los integrantes estén unidos, comunicados y que el cliente vaya viendo los avances.

¹² (Silberschatz, 2002)

A continuación, se definen algunos de los conceptos más importantes de esta metodología:

Product backlog: Es una “lista de deseos” sobre las funcionalidades del producto. Es elaborado por el cliente (Product Owner) y las funciones están priorizadas según lo que es más y menos importante para el negocio. El objetivo es que el Product Owner responda la pregunta “¿Qué hay que hacer?”.

Sprint backlog: Es un subconjunto de ítems del Product Backlog, que son seleccionados por el equipo para realizar durante el Sprint sobre el que se va a trabajar. El equipo establece la duración de cada Sprint.

Sprint planning meeting: Esta reunión se hace al comienzo de cada Sprint y se define cómo se va a enfocar el proyecto que viene del Product Backlog las etapas y los plazos. Cada Sprint está compuesto por diferentes características (features).

Daily scrum o Stand-up meeting: Es una reunión breve que se realiza a diario mientras dura el periodo de Sprint. Se responden individualmente tres preguntas: ¿Qué hice ayer?, ¿Qué voy a hacer hoy?, ¿Qué ayuda necesito?

El Gestor de Proyecto (Scrum Master) debe tratar de solucionar los problemas u obstáculos que se presenten.

Sprint review: Se revisa el sprint terminado, y ya debería haber un avance claro y tangible para presentárselo al cliente.

Sprint retrospective: El equipo revisa los objetivos cumplidos del Sprint terminado. Se anota lo bueno y lo malo, para no volver a repetir los errores. Esta etapa sirve para implementar mejoras desde el punto de vista del proceso del desarrollo.

Participantes

Product Owner: Habla por el cliente, y asegura que el equipo cumpla las expectativas. Es “el jefe” responsable del proyecto.

Scrum Master: Lidera las reuniones y ayuda al equipo si es que tienen problemas. Además, minimiza los obstáculos para cumplir el objetivo del Sprint, es un “facilitador” pero no es un gestor.

Scrum Team: Son los encargados de desarrollar y cumplir lo que les asigna el Product Owner.

Cliente: Recibe el producto y puede influir en el proceso, entregando sus ideas o comentarios respecto al desarrollo.¹³

2.3.1.5 Aplicación Web.

Se denomina aplicación web a aquellas *herramientas* que los usuarios pueden utilizar accediendo a un *servidor web* a través de *Internet* mediante un *navegador*.

Las aplicaciones web son populares debido a lo práctico del *navegador web* como *cliente ligero*, a la independencia del sistema operativo, así como a la facilidad para actualizar y mantener aplicaciones web sin distribuir e instalar software a miles de usuarios potenciales.¹⁴

Ventajas de las Aplicaciones Web

- **Ahorran tiempo:** se pueden realizar tareas sencillas sin necesidad de descargar ni instalar ningún programa.
- **No existen problemas de compatibilidad:** basta tener un navegador actualizado para poder utilizarlas.
- **No ocupan espacio** en nuestro disco duro.
- **Actualizaciones inmediatas:** como el software lo gestiona el propio desarrollador, cuando nos conectamos estamos usando siempre la última versión que haya lanzado.
- **Consumo bajo de recursos:** dado que toda (o gran parte) de la aplicación no se encuentra en nuestra computadora.
- **Multiplataforma:** se pueden usar desde cualquier sistema operativo porque solamente es necesario tener un navegador.

¹³ proyectosagiles.org/base-conocimiento-agil/

¹⁴ (Luján, 2001)

- **Portables:** es independiente de la computadora donde se utilice (PC de sobremesa, portátil) porque se accede a través de una página web (solamente es necesario disponer de acceso a Internet).
- **Alta disponibilidad** porque el servicio se ofrece desde múltiples localizaciones para asegurar la continuidad del mismo.

2.3.1.6 Python

Python es un lenguaje de scripting independiente de plataforma y orientado a objetos, preparado para realizar cualquier tipo de programa, desde aplicaciones Windows a servidores de red o incluso, páginas web. Es un lenguaje interpretado, lo que significa que no se necesita compilar el código fuente para poder ejecutarlo, lo que ofrece ventajas como la rapidez de desarrollo e inconvenientes como una menor velocidad.

Python puede ser utilizado en diversas plataformas y sistemas operativos, entre los que podemos destacar los más populares, como Windows, Mac OS X y Linux. Pero, además, Python también puede funcionar en SmartPhones.

Python es de código abierto, es decir, cualquiera puede contribuir a su desarrollo y divulgación. Además, no es necesario pagar ninguna licencia para distribuir software desarrollado con este lenguaje. Hasta su intérprete se distribuye de forma gratuita para diferentes plataformas.¹⁵

2.3.1.7 PHP

PHP (acrónimo recursivo de PHP: Hypertext Preprocessor) es un lenguaje de código abierto muy popular especialmente adecuado para el desarrollo web y que puede ser incrustado en HTML.

Lo que distingue a PHP de algo del lado del cliente como Javascript es que el código es ejecutado en el servidor, generando HTML y enviándolo al cliente.

El cliente recibirá el resultado de ejecutar el script, aunque no se sabrá el código subyacente que era.

¹⁵ <https://desarrolloweb.com/articulos/1325.php>

El servidor web puede ser configurado incluso para que procese todos los ficheros HTML con PHP, por lo que no hay manera de que los usuarios puedan saber qué se tiene debajo de la manga.

Lo mejor de utilizar PHP es su extrema simplicidad para el principiante, pero a su vez ofrece muchas características avanzadas para los programadores profesionales.¹⁶

2.3.1.8 HTML (*HyperText Markup Language*)

HTML o Lenguaje de Formato de Documentos para Hipertexto, se encarga de desarrollar una descripción sobre los contenidos que aparecen como textos y sobre su estructura, complementando dicho texto con diversos objetos (como fotografías, animaciones, etc), es decir, sirve para crear páginas web y darles estructura y contenido

Es un lenguaje muy simple y general que sirve para definir otros lenguajes que tienen que ver con el formato de los documentos. El texto en él se crea a partir de *etiquetas*, que permiten interconectar diversos conceptos y formatos.¹⁷

2.3.1.9 Bootstrap

Es un framework o conjunto de herramientas de Código abierto para diseño de sitios y aplicaciones web. Contiene plantillas de diseño con tipografía, formularios, botones, cuadros, menús de navegación y otros elementos de diseño basado en HTML y CSS, así como, extensiones de JavaScript opcionales adicionales.

Bootstrap fue desarrollado por Mark Otto y Jacob Thornton de Twitter, como un marco de trabajo (framework) para fomentar la consistencia entre las herramientas internas. Antes de Bootstrap, se usaban varias librerías para el desarrollo de interfaces de usuario, las cuales llevaban a inconsistencias y a una gran carga de trabajo en su mantenimiento.¹⁸

¹⁶ <http://php.net/manual/es/intro-whatis.php>

¹⁷ <https://developer.mozilla.org/es/docs/Web/HTML>

¹⁸ puntoabierto.net/blog/que-es-bootstrap-y-cuales-son-sus-ventajas, 2017

2.3.1.10 HighCharts

Es un producto que fue creado por la empresa con sede en Noruega, Highsoft. Highcharts fue lanzado en 2009, y es una biblioteca de gráficos escrito en puro JavaScript. El producto fue demostrado por primera vez y pensó en 2006.

En una entrevista con Finansavisen, habló sobre la necesidad de un programa para crear gráficos y tablas, lo que permite a los usuarios publicar gráficos directamente en una página web.¹⁹

2.3.2 Componentes físicos (Hardware)

2.3.2.1 Hardware de Adquisición de datos (DAQ)

La adquisición de datos (DAQ) es el proceso de medir con una PC un fenómeno eléctrico o físico como voltaje, corriente, temperatura, presión o sonido.

Los dispositivos DAQ son instrumentos fundamentales para aplicaciones diseñadas para el registro de datos externos ya sean analógicos y/o digitales.

El hardware DAQ actúa como la interfaz entre una PC y señales del mundo exterior. Funciona principalmente como un dispositivo que digitaliza señales analógicas entrantes para que una PC pueda interpretarlas. Los tres componentes clave de un dispositivo DAQ usado para medir una señal son el circuito de acondicionamiento de señales, convertidor analógico-digital (ADC) y un bus de PC.²⁰

Una PC de escritorio se puede utilizar en un laboratorio por su poder de procesamiento, una laptop se puede utilizar por su portabilidad o una PC industrial se puede utilizar en una planta de producción por su robustez²¹.

La figura 2.6 muestra el diagrama de bloques de la adquisición de datos basada en un computador.

¹⁹ www.highcharts.com/products/highcharts/, 2017

²⁰ (Instruments, 2017)

²¹ (Instruments, 2017)



Figura 2.7 Diagrama de bloques DAQ

Fuente: www.ni.com/data-acquisition/what-is/esa/

2.3.2.2 Sensor

La medida de un fenómeno físico, como la temperatura de una habitación, la intensidad de una fuente de luz o la fuerza aplicada a un objeto, comienza con un sensor. Un sensor, también llamado un transductor, convierte un fenómeno físico en una señal eléctrica que se puede medir.

Dependiendo del tipo de sensor, su salida eléctrica puede ser un voltaje, corriente, resistencia u otro atributo eléctrico que varía con el tiempo. Algunos sensores pueden requerir componentes adicionales y circuitos para producir correctamente una señal que puede ser leída con precisión y con toda seguridad por un dispositivo DAQ.²²

En la figura 2.7 se muestran algunos tipos de sensores más utilizados.

²² (Instruments, 2017)



Figura 2.8 Tipos de Sensores

Fuente: sensoricamaria.blogspot.com

2.3.2.3 Sensor de Intensidad Luminosa LDR

Un LDR (Resistor Dependiente de la Luz) es un resistor que varía su valor de resistencia eléctrica dependiendo de la cantidad de luz que incide sobre él. Es llamado también, fotorresistor o fotoresistencia. Los sensores de luz se usan para producir una señal de salida que represente la cantidad de luz detectada.

El valor de resistencia eléctrica de un LDR es bajo cuando hay luz incidiendo en él y muy alto cuando está a oscuras. El LDR es un elemento muy útil para aplicaciones en circuitos donde se necesita detectar la ausencia de luz de día.

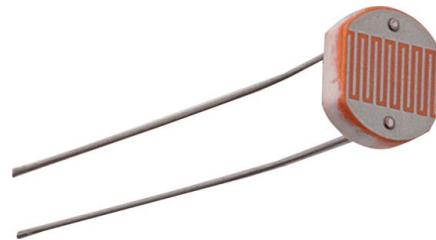


Figura 2.9 Fotorresistencia o LDR

Fuente: saber.patagoniatec.com/ldr-arduino-argentina-ptec/

2.3.2.4 Sensor de Temperatura y Humedad DHT11

El DHT11 es un sensor de temperatura y humedad digital de bajo costo. Utiliza un sensor capacitivo de humedad y un termistor para medir el aire circundante, y muestra los datos mediante una señal digital en el pin de datos (no hay pines de entrada analógica).²³

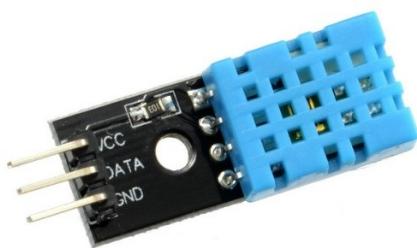


Figura 2.10 Sensor de Temperatura y Humedad DHT11

Fuente: electronilab.co/tienda/sensor-de-temperatura-y-humedad-dht11/

Su manejo es sencillo, pero requiere sincronización cuidadosa para tomar datos. El único inconveniente de este sensor es que sólo se puede obtener nuevos datos una vez cada 2 segundos, así que las lecturas que se pueden realizar serán mínimo cada 2 segundos.

En comparación con el DHT22, este sensor es menos preciso, menos exacto y funciona en un rango más pequeño de temperatura / humedad, pero su empaque es más pequeño y más económico.

2.3.2.5 Actuador

Un actuador es un dispositivo capaz de transformar energía hidráulica, neumática o eléctrica en la activación de un proceso con la finalidad de generar un efecto sobre un proceso automatizado. Se puede decir que un actuador realiza la función inversa de un sensor.²⁴

Existen multitud de dispositivos que pueden considerarse como actuadores, entre los más importantes podemos mencionar:

²³ electronilab.co/tienda/sensor-de-temperatura-y-humedad-dht11/

²⁴ www.slideshare.net/ErendiraAvalosMorales/sensores-y-actuadores-11723590

- Los **relés** que actúan como un interruptor controlado por un circuito eléctrico y son capaces de conmutar un circuito de salida de mayor potencia que el de entrada, permitiendo encender y apagar bombillas u otros equipos.
- Los **dimmers** son dispositivos que regulan la potencia que llega a una carga, se usan para regular la intensidad de las bombillas, fluorescentes, etc.
- Las **electroválvulas** se utilizan para regular el fluido de líquidos y gases.
- También **motores eléctricos, contactores**, etc. se consideran actuadores.

2.3.2.6 Placa Arduino

Arduino es una plataforma de prototipos electrónica de código abierto (open-source) basada en hardware y software flexibles y fáciles de usar.

Arduino puede tomar información del entorno a través de sus pines de entrada de toda una gama de sensores y puede afectar aquello que le rodea controlando luces, motores y otros actuadores.

El microcontrolador en la placa Arduino se programa mediante el lenguaje de programación Arduino (basado en Wiring) y el entorno de desarrollo Arduino (basado en Processing).²⁵

²⁵ (Arduino, 2017)

En la figura 2.8 se muestra el modelo de Arduino más utilizado actualmente.



Figura 2.11 Arduino UNO Rev 3

Fuente: store.arduino.cc/usa/arduino-uno-rev3

2.3.2.7 Raspberry Pi

La Raspberry Pi es un ordenador de bajo costo aproximadamente del tamaño de una tarjeta de crédito que se conecta a un monitor de ordenador o un televisor, y utiliza un teclado y un ratón estándar para su manejo.

Se trata de un pequeño dispositivo capaz que permite a las personas de todas las edades explorar el área informática, y aprender a programar en lenguajes como Scratch y Python. Es capaz de hacer todo lo que hace un ordenador de sobremesa, desde la navegación por Internet, reproducción de videos de alta definición, procesamiento de hojas de cálculo hasta jugar algunos videojuegos.

La Raspberry Pi tiene la capacidad de interactuar con el mundo exterior, y se ha utilizado en una amplia variedad de proyectos informáticos gracias a su costo.²⁶

²⁶ www.raspberrypi.org/help/what-%20is-a-raspberry-pi/, 2015

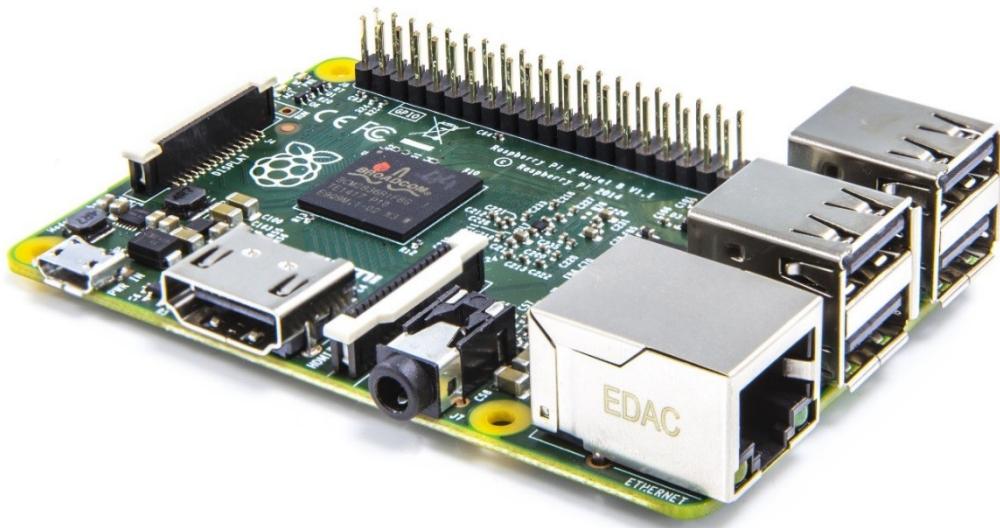


Figura 2.12 Raspberry Pi Modelo B Versión 1.1

Fuente: [/www.raspberrypi.org/blog/raspberry-pi-2-on-sale/](http://www.raspberrypi.org/blog/raspberry-pi-2-on-sale/), 2015

Existen muchos modelos diferentes de Raspberry Pi, como ser:

Raspberry Pi Modelo A. Se trata de la versión más básica. Con un tamaño de 85.6 x 56.5 mm, tiene un consumo de 1.5W y pesa 45 gramos. El procesador gráfico es compatible con OpenGL 2.0 y es capaz de decodificar vídeo a 1080p.



Figura 2.13 Raspberry Pi Modelo A

Fuente: www.raspberrypi.org/products/, 2017

Raspberry Pi Modelo A+. Esta variante lanzada en Noviembre de 2014, también ofrece el mismo hardware que el modelo A pero añade más conectores GPIO, hasta llegar a los 17, soporta tarjetas microSD, posee un sistema de audio mejorado, y además es más pequeña y consume tan solo 1W. Mide 65 x 56.5 mm y pesa 23 gramos.



Figura 2.14 Raspberry Pi Modelo A +

Fuente: www.raspberrypi.org/products/, 2017

Raspberry Pi Modelo B. Fue el primer modelo que salió al mercado en Febrero de 2015. Tiene el mismo hardware que el modelo A, pero incluye 512 MB de memoria RAM, así como un segundo puerto USB. Además, una conexión de red Ethernet 10/100 que permite conectar la Raspberry Pi a Internet.



Figura 2.15 Raspberry Pi Modelo B

Fuente: www.raspberrypi.org/products/, 2017

Raspberry Pi Modelo B+. El Modelo B + es la revisión final de la original Raspberry Pi. Reemplazó el Modelo B en julio de 2014 y fue reemplazado por el Raspberry Pi 2 Modelo B. Tiene el mismo hardware que el modelo B, pero se añade dos puertos adicionales USB 2.0, tarjeta microSD, audio mejorado un menor consumo de 3W.



Figura 2.16 Raspberry Pi Modelo B+

Fuente: www.raspberrypi.org/products/, 2017

Raspberry Pi 2 Modelo B. Es la segunda generación de Raspberry Pi. Reemplazó el original Raspberry Pi 1 Modelo B + en febrero de 2015. Conserva la misma arquitectura, así que mantiene la estabilidad de la Raspberry Pi, pero añade una nueva CPU ARM Cortex-A7 de cuatro núcleos a 900 MHz, así como 1 GB de memoria RAM a 450 Mhz. Es 6 veces más potente que los modelos anteriores, posibilitando la ejecución de Windows 10.



Figura 2.17 Raspberry Pi 2 Modelo B

Fuente: www.raspberrypi.org/products/, 2017

Raspberry Pi 3 Modelo B. Entre sus principales novedades la Raspberry Pi 3 cuenta con un procesador ARM Cortex A53, cuatro núcleos a 1.2 GHz de 64 bits, el cual da un rendimiento hasta 10 veces mejor que el Raspberry original y hasta un 50% mejor que el modelo Raspberry Pi 2. Además, incluye conectividad Bluetooth y Wi-fi.



Figura 2.18 Raspberry Pi 3 Modelo B

Fuente: www.raspberrypi.org/products/, 2017

2.3.2.8 NodeMCU

El NodeMcu es un kit de desarrollo de código abierto basado en el chip ESP8266 (ESP-12E), que utiliza el lenguaje de programación Lua²⁷ para crear un ambiente de desarrollo propicio para aplicaciones que requiera conectividad Wifi de manera rápida.



Figura 2.19 ESP8266 – NodeMCU

Fuente: electronilab.co/tienda/nodemcu-board-de-desarrollo-con-esp8266-wifi-y-lua/, 2017

El ESP8266 es un chip altamente integrado diseñado para las necesidades de proyectos del Internet de las Cosas (IoT). Ofrece una solución completa y

²⁷ Lua. Lenguaje de programación que carga scripts al controlador para ejecutarlos en el arranque.

autónoma de redes Wi-Fi, lo que le permite alojar la aplicación o servir como puente entre Internet y un microcontrolador.

El ESP8266 tiene potentes capacidades de procesamiento y almacenamiento que le permiten integrarse con sensores y dispositivos específicos a través de sus GPIOs²⁸ con un desarrollo mínimo y carga mínima durante el tiempo de ejecución.

²⁸ GPIO: General Port Input Output – Puerto General de Entrada o Salida.

CAPÍTULO III

DESARROLLO DEL PROYECTO

3 DESARROLLO DEL PROYECTO

En este capítulo se lleva a cabo el proceso de análisis, diseño, programación, documentación y pruebas al sistema a desarrollar, este proceso se realiza siguiendo los lineamientos de la metodología de desarrollo de procesos ágiles SCRUM, que permite por medio de una serie de reuniones mayormente cortas realizar entregas funcionales llamadas incrementos.

3.1 PLAN DE TRABAJO

El plan de trabajo elaborado para el desarrollo del proyecto se puede observar en la tabla 3.1:

Tabla 3.3 Plan de Trabajo

Descripción	Fecha Inicio	Fecha Fin
Inicio del proyecto	05/06/2017	05/06/2017
Análisis y Requerimientos	05/06/2017	20/06/2017
Selección de Componentes	21/06/2017	23/06/2017
Construcción del prototipo de cultivo hidropónico	24/06/2017	06/07/2017
Diseño de la Aplicación	07/07/2017	24/07/2017
Creación de Base de Datos	08/07/2017	08/07/2017
Desarrollo de la Aplicación	25/07/2017	09/08/2017
Documentación	05/06/2017	15/11/2017
Fin del proyecto	15/11/2017	15/11/2017

Fuente: Elaboración Propia, 2017

3.2 PLANEACIÓN DEL PROCESO SCRUM

La metodología SCRUM requiere una planeación bien definida, deberá establecerse un equipo de trabajo, unos objetivos y estos objetivos serán divididos en ciclos (iteraciones) de máximo 20 días, realizando reuniones cortas de 10 minutos cada día, para verificar el avance, analizar los problemas y determinar rápidamente un plan de acción para solucionarlos.

3.2.1 Definición del Equipo

El equipo SCRUM estará conformado por:

Tabla 3.4 Equipo de trabajo de Scrum

Persona	Contacto	Rol
Ing. Pavel Cáceres Torrez	70719630	Scrum Master
Sr. Sebastián Alcalá	72289564	Product Owner
Albarado Mirna / Encalada Cristian	72205308-72258956	Development Team

Fuente: Elaboración Propia, 2017

3.3 ARTEFACTOS

3.3.1 Definición de Objetivos del Producto - Product Backlog

Los objetivos del producto (Product BackLog) son:

The screenshot shows a software interface for managing a product backlog. The top navigation bar includes links for Dashboards, Code, Work, Build and Release, Test, Wiki, and a search bar. Below the navigation, there are tabs for Work Items*, Backlogs, and Queries, with Backlogs being the active tab. A sidebar on the left lists categories: Features, Backlog items, Current, Sprint 1, Future, Sprint 2, Sprint 3, Sprint 4, Sprint 5, and Sprint 6. Under the Backlog items section, there are buttons for New, Create query, Column options, and Email. The main area displays a table titled 'Backlog items' with columns for Order, Work Item Type, Title, State, and Effort. Six backlog items are listed, each with a small icon and a brief description. The items are ordered by priority (Order 1 to 6). The first item is 'Seleccionar los Dispositivos Electrónicos (hardware)'.

Order	Work Item Type	Title	State	Effort
1	Product Backl...	Seleccionar los Dispositivos Electrónicos (hardware).	New	2
2	Product Backl...	Diseñar e implementar el Prototipo del Invernadero.	New	2
3	Product Backl...	Desarrollar un Módulo de Gestión de Usuarios.	New	1
4	Product Backl...	Desarrollar un Módulo Web de Monitoreo	New	3
5	Product Backl...	Desarrollar un Módulo Web de Control	New	4
6	Product Backl...	Desarrollar un Módulo Web de Reportes	New	2

Figura 3.20 Product Backlog

Fuente: Elaboración propia, 2017

Tabla 3.5 Product Backlog

Id	Título	Estimación (1-4)	Descripción
1	Analizar y diseñar la aplicación web	4	Elaborar diagramas de casos de uso, clases y definir requerimientos del sistema.
2	Seleccionar los Dispositivos Electrónicos (hardware).	2	Seleccionar dispositivos, en función del alcance del proyecto
3	Diseñar e implementar el Prototipo del Invernadero.	2	Construir el prototipo de cultivo hidropónico.
4	Desarrollar un Módulo de Gestión de Usuarios.	1	Realizar la gestión, validación y auditoría de usuarios de la aplicación.
5	Desarrollar un Módulo Web de Monitoreo	3	Visualizar parámetros ambientales en tiempo real a través de la aplicación web.
6	Desarrollar un Módulo Web de Control.	4	Controlar los dispositivos eléctricos a través de la aplicación web accesible desde cualquier lugar con conexión a internet.
7	Desarrollar un Módulo Web de Reportes.	2	Obtener reportes de los parámetros ambientales en rangos de tiempo.

Fuente: Elaboración Propia, 2017

3.3.2 Definición de Objetivos de las Iteraciones - Sprint Backlog

Sprint 1: En el primer Sprint, se realiza el análisis y diseño de la aplicación.

The screenshot shows the 'Backlogs' tab selected in the navigation bar. On the left, a sidebar lists 'Features', 'Backlog items', and 'Current' sections for 'Sprint 1', 'Future', 'Sprint 2', 'Sprint 3', and 'Sprint 4'. The main area displays the 'SISTEMA DE MONITOREO Y CONTROL DE UN INVERNADERO Team Sprint 1' backlog. It includes tabs for 'Backlog', 'Board', and 'Capacity'. Below is a table with columns: Order, Reason, Title, and State. Three items are listed:

Order	Reason	Title	State
1	Approved by the Product Owner	Elaborar diagramas de análisis	Approved
2	Approved by the Product Owner	Definir requerimientos	Approved
3	Approved by the Product Owner	Elaborar diagramas de diseño	Approved

Figura 3.21 Sprint 1 Backlog

Fuente: Elaboración propia, 2017

Sprint 2: En el segundo Sprint, se atienden los ítems del producto:

- Seleccionar los Dispositivos Electrónicos (hardware).
- Diseñar e implementar el Prototipo del Invernadero.

The screenshot shows the 'Backlogs' tab selected in the navigation bar. On the left, a sidebar lists 'Features', 'Backlog items', and 'Current' sections for 'Sprint 1', 'Future', 'Sprint 2', 'Sprint 3', and 'Sprint 4'. The main area displays the 'SISTEMA DE MONITOREO Y CONTROL DE UN INVERNADERO Team Sprint 2' backlog. It includes tabs for 'Backlog', 'Board', and 'Capacity'. Below is a table with columns: Order, Reason, and Title. Six new backlog items are listed:

Order	Reason	Title
1	New backlog it...	Seleccionar los Dispositivo de Adquisición de Datos
2	New backlog it...	Seleccionar Dispositivo de Control y Almacenamiento de Datos
3	New backlog it...	Seleccionar Sensores
4	New backlog it...	Seleccionar Actuadores
5	New backlog it...	Seleccionar materiales para el prototipo de cultivo hidropónico
6	New backlog it...	Construir el prototipo hidropónico

Figura 3.22 Sprint 2 Backlog

Fuente: Elaboración propia, 2017

Sprint 3: En el tercer Sprint, se atienden los ítems del producto:

- Desarrollar un Módulo de Gestión de Usuarios.
- Desarrollar un Módulo Web de Monitoreo.

The screenshot shows the 'Backlogs' tab selected in the Azure DevOps interface. The title bar reads 'SISTEMA DE MONITOREO Y CONTROL DE UN INVERNADERO Team Sprint 3'. The backlog table has columns for Order, Reason, and Title. The backlog items are as follows:

Order	Reason	Title
1	Approved by t...	Definir los roles de usuario
2	New backlog it...	Crear tablas de usuario y roles en base de datos
3	New backlog it...	Crear Modelo Usuario
4	New backlog it...	Crear Controlador Usuario
5	New backlog it...	Crear Vista Usuario
6	New backlog it...	Crear Modelo Administrador
7	New backlog it...	Crear Controlador Administrador
8	New backlog it...	Crear Vista Administrador
9	New backlog it...	Realizar Autenticación de Usuarios
10	New backlog it...	Realizar Validación de Usuarios
11	New backlog it...	Realizar Registro de Usuarios
12	New backlog it...	Realizar la conexión de sensores y NodeMCU
13	New backlog it...	Programación de NodeMCU para obtener datos de monitoreo
14	New backlog it...	Recuperar y mostrar datos de monitoreo en la aplicación Web

Figura 3.23 Sprint 3 Backlog

Fuente: Elaboración propia, 2017

Sprint 4: En el cuarto Sprint, se atienden los ítems del producto:

- Desarrollar un Módulo Web de Control.
- Desarrollar un Módulo Web de Reportes.

The screenshot shows a Microsoft Azure DevOps interface. At the top, there's a navigation bar with links for Dashboards, Code, Work, Build and Release, Test, Wiki, and a gear icon. Below the navigation bar, there are tabs for Work Items*, Backlogs, and Queries, with 'Backlogs' being the active tab. On the left, a sidebar lists 'Features' and 'Backlog items' under 'Current', with 'Sprint 4' selected. The main area displays a backlog titled 'SISTEMA DE MONITOREO Y CONTROL DE UN INVERNADERO Team Sprint 4'. It includes tabs for Backlog, Board, and Capacity. The backlog table has columns for Order, Reason, and Title. There are 11 items listed, each with a small icon and a link to its details. The items are:

Order	Reason	Title
1	New backlog it...	Instalar el SO Raspbian en la Raspberry Pi 2
2	New backlog it...	Instalar y configurar PHP, MySQL, Phyton y Sensores en Raspberry Pi 2
3	New backlog it...	Crear Modelo Control Independiente
4	New backlog it...	Crear Controlador Control Independiente
5	New backlog it...	Crear Vista Control Independiente
6	New backlog it...	Realizar la conexión de sensores a Arduino UNO
7	New backlog it...	Programar el Arduino Uno para almacenar variables ambientales
8	New backlog it...	Recuperar Datos de variables ambientales de la Raspberry Pi 2
9	New backlog it...	Crear Reporte de Temperatura
10	New backlog it...	Crear Reporte de Humedad
11	New backlog it...	Crear Reporte de Iluminacion

Figura 3.24 Sprint 4 Backlog

Fuente: Elaboración propia, 2017

3.3.3 Lista de Tareas de Sprints (1 al 4)

Tabla 3.6 Lista de Tareas de Sprints

Id	Sprint	T/h	Item	Asignado a:
1	1	32	Elaborar diagramas de análisis	Mirna Albarado
2	1	24	Definir requerimientos	Cristian Encalada
3	1	40	Elaborar diagramas de diseño	Mirna Albarado
4	2	4	Seleccionar los Dispositivo de Adquisición de Datos	Cristian Encalada
5	2	4	Seleccionar Dispositivo de Control y Almacenamiento de Datos	Cristian Encalada
6	2	4	Seleccionar Sensores	Cristian Encalada
7	2	4	Seleccionar Actuadores	Cristian Encalada
8	2	8	Seleccionar materiales para el prototipo de cultivo	Mirna Albarado
9	2	72	Construir el prototipo hidropónico	Mirna Albarado / Cristian Encalada

1 0	3	2	Definir los roles de usuario	Mirna Albarado
1 1	3	2	Crear tablas de usuario y roles en base de datos	Mirna Albarado
1 2	3	8	Crear Modelo Usuario	Cristian Encalada
1 3	3	8	Crear Controlador Usuario	Cristian Encalada
1 4	3	4	Crear Vista Usuario	Mirna Albarado
1 5	3	8	Crear Modelo Administrador	Cristian Encalada
1 6	3	8	Crear Controlador Administrador	Cristian Encalada
1 7	3	4	Crear Vista Administrador	Mirna Albarado
1 8	3	12	Realizar Autenticación de Usuarios	Cristian Encalada
1 9	3	12	Realizar Validación de Usuarios	Cristian Encalada
2 0	3	12	Realizar Registro de Usuarios	Mirna Albarado
2 1	3	4	Realizar la conexión de sensores y NodeMCU	Mirna Albarado
2 2	3	4	Programación de NodeMCU para obtener datos de monitoreo	Cristian Encalada
2 3	3	8	Recuperar y mostrar datos de monitoreo en la aplicación Web	Cristian Encalada
2 4	4	8	Instalar el SO Raspbian en la Raspberry Pi 2	Mirna Albarado
2 5	4	24	Instalar y configurar PHP, MySql, Phyton y Sensores en Raspberry Pi 2	Mirna Albarado
2 6	4	8	Crear Modelo Control Independiente	Cristian Encalada
2 7	4	8	Crear Controlador Control Independiente	Cristian Encalada
2 8	4	8	Crear Vista Control Independiente	Mirna Albarado
2 9	4	4	Realizar la conexión de sensores a Arduino UNO	Cristian Encalada
3 0	4	24	Programar Arduino Uno para almacenar variables ambientales en la Raspberry Pi 2	Mirna Albarado
3 1	4	12	Recuperar Datos de variables ambientales de la Raspberry Pi 2	Cristian Encalada
3	4	8	Crear Reporte de Temperatura	Mirna Albarado

2				
3	4	8	Crear Reporte de Humedad	Mirna Albarado
3	4	8	Crear Reporte de Iluminación	Mirna Albarado

Fuente: Elaboración propia, 2017

3.3.4 Planeación de las iteraciones – Sprint Planning

Las Iteraciones (Sprints) tendrán una duración de 12 días laborales de 8 horas diarias, y se han distribuido de la siguiente manera:

Tabla 3.7 Planeación de Iteraciones – Sprint Planning

Sprint	Fecha Inicio	Fecha Fin
Sprint 1	05/06/2017	20/06/2017
Sprint 2	21/06/2017	06/07/2017
Sprint 3	07/07/2017	24/07/2017
Sprint 4	25/07/2017	09/08/2017

Fuente: Elaboración propia, 2017

3.3.5 Definición de Terminado

Todo el equipo Scrum dará por terminado un elemento de la lista de pendientes del Sprint (Sprint Backlog), cuando realice las actividades que se ha planeado que realice y esté listo para ir a pruebas, las modificaciones se realizarán una vez terminadas las pruebas, y un incremento se dará por terminado cuando las pruebas sean aprobadas correctamente y en su totalidad.

3.3.6 Burndown Charts

3.3.7 Burndown Chart – Product Backlog

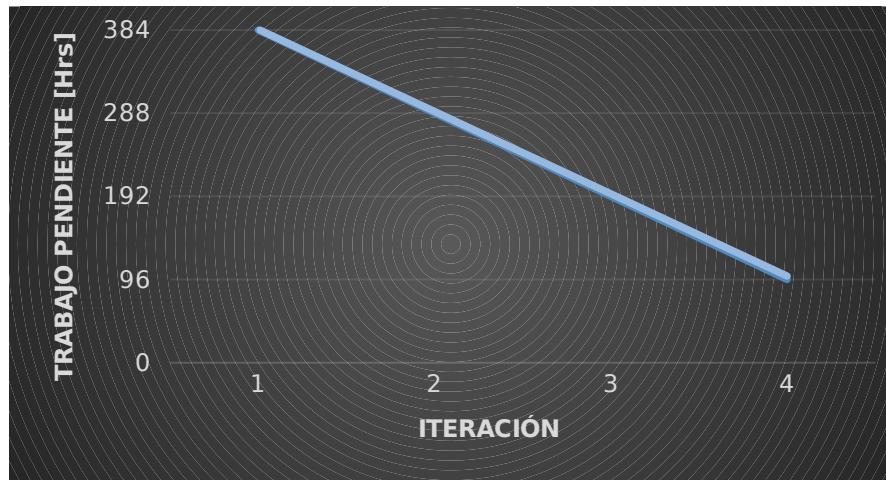


Figura 3.25 Burndown Chart – Product Backlog

Fuente: Elaboración Propia, 2017

3.4 REUNIÓN DE PLANIFICACIÓN DEL SPRINT 1 – 2017/06/05

El objetivo de la reunión es identificar los objetivos que deberán cumplirse en la iteración y establecer un procedimiento para llevarlos a cabo.

3.4.1 Asistentes

Sebastián Alcalá – *Product Owner*

Mirna Albarado / Cristian Encalada – *Equipo de Desarrollo*

Ing. Pavel Cáceres Torrez – *Scrum Master*

3.4.2 ¿Qué se completará en este Sprint? (Sprint Backlog)

En el primer Sprint, se realiza el análisis y diseño de la aplicación, esto es:

- Definir actores del sistema
- Diagramas de Casos de Uso
- Refinamiento de Casos de Uso

- Diccionario de Casos de Uso
- Diagramas de Clases para el Diseño
- Definición de los requerimientos funcionales
- Definición de los requerimientos no funcionales
- Diagrama de Capas
- Diagrama de Despliegue

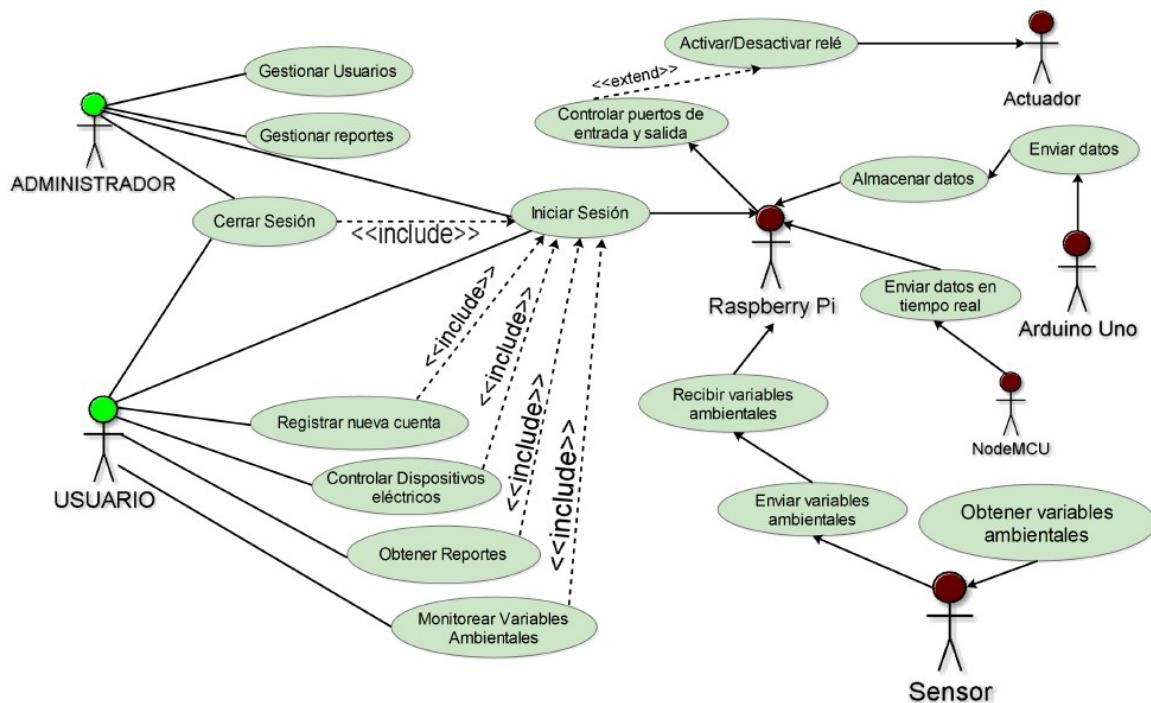
3.4.2.1 Actores del Sistema

- **Administrador.** Tiene acceso total al sistema, puede gestionar usuarios, reportes y variables ambientales.
- **Usuario.** Tiene un acceso más restringido al sistema, puede controlar dispositivos, obtener reportes y monitorear variables ambientales.
- **Raspberry.** Es el hardware principal del sistema, almacena datos, recibe variables ambientales y controla actuadores y sensores.
- **Arduino UNO.** Realiza el envío de variables ambientales para su almacenamiento en el servidor.
- **NodeMCU.** Realiza el envío de variables ambientales para el monitoreo en tiempo real.
- **Sensor.** Obtiene variables ambientales.
- **Actuador.** Generar un efecto sobre un proceso automatizado

3.4.2.2 Diagrama de Casos de Uso

En el diagrama 3.6 se observa el diagrama general de casos de uso del sistema, en el que interactúan los actores del sistema y que más adelante se detallarán casos más específicos en el refinamiento de casos de uso.

Diagrama 3.2 Diagrama General de Casos de Uso del Sistema

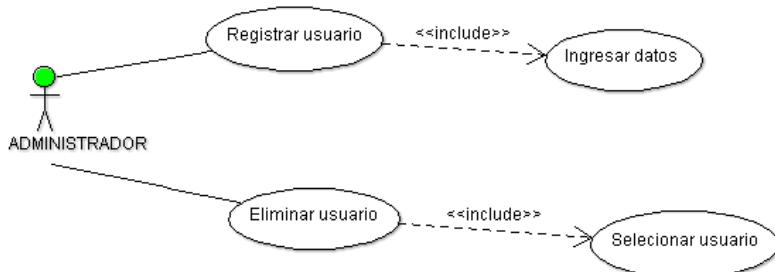


Fuente: Elaboración Propia, 2017

3.4.2.3 Refinamiento de Casos de Uso

Caso de Uso Gestionar Usuarios

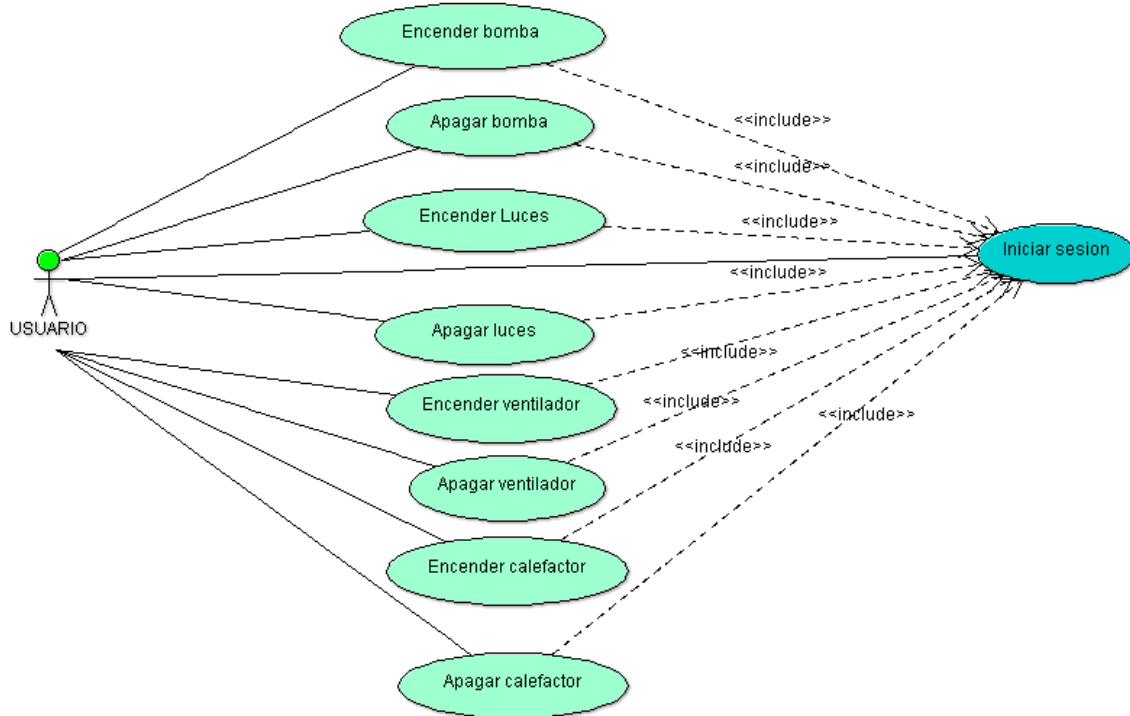
Diagrama 3.3 Caso de Uso Gestionar Usuarios



Fuente: Elaboración Propia, 2017

Caso de Uso Controlar Dispositivos

Diagrama 3.4 Caso de Uso Controlar Dispositivos



Fuente: Elaboración Propia, 2017

3.4.2.4 Diccionario de Casos de Uso

Tabla 3.8 Caso de Uso Registrar Usuario

CASO DE USO: Registrar usuario	
Caso de uso	Registrar usuario
Resumen	Se realiza el registro de usuario
Actor	Administrador
Precondiciones	El administrador debe contar con datos del usuario.
Post condiciones	Ninguna
Flujo Básico	1.- El administrador inicia sesión. 2.-El administrador realiza el registro en la tabla de usuarios.

--	--

Fuente: Elaboración propia, 2017

Tabla 3.9 Caso de Uso Eliminar Usuario

CASO DE USO: Eliminar usuario	
Caso de uso	Eliminar usuario
Resumen	Elimina usuarios registrados
Actor	Administrador
Precondiciones	Tener usuarios registrados
Post condiciones	Ninguna
Flujo Básico	<ol style="list-style-type: none"> 1.-El administrador inicia sesión. 2.-El administrador selecciona usuario a eliminar. 3.-El sistema muestra mensaje de eliminación de usuario. 4.-El administrador realiza la eliminación.

Fuente: Elaboración propia, 2017

Tabla 3.10 Caso de Uso Iniciar Sesión

CASO DE USO: Iniciar sesión	
Caso de uso	Iniciar sesión
Resumen	El usuario debe iniciar sesión para acceder a las plataformas de control y monitoreo.
Actor	Usuario
Precondiciones	Contar con internet
Post condiciones	Ninguna
Flujo Básico	<ol style="list-style-type: none"> 1.-El usuario debe iniciar sesión para acceder al sistema 2.- El usuario debe ingresar nombre usuario y contraseña al sistema.

	3. El sistema verifica datos de usuario. 4.-El sistema muestra plataforma principal.
--	---

Fuente: Elaboración Propia, 2017

Tabla 3.11 Caso de Uso Encender Bomba de Agua

CASO DE USO: Encender Bomba de Agua	
Caso de uso	Encender bomba de Agua
Resumen	El usuario enciende la bomba de agua a través del botón de encender.
Actor	Usuario
Precondiciones	El usuario debe iniciar sesión.
Post condiciones	Ninguna
Flujo Básico	<p>1.- El usuario accede a la plataforma principal del sistema, desde cualquier punto del mundo.</p> <p>2.- El usuario selecciona la pestaña de control.</p> <p>3.-El sistema muestra la plataforma de control.</p> <p>4.-El usuario selecciona el botón de encender motor.</p> <p>5.-El sistema muestra el estado de encender de la bomba.</p>

Fuente: Elaboración propia, 2017

Tabla 3.12 Caso de Uso Apagar Bomba de Agua

CASO DE USO: Apagar bomba de agua	
Caso de uso	Apagar bomba de agua
Resumen	El usuario apaga la bomba de agua a través del botón de apagado desde la plataforma de control.
Actor	Usuario
Precondiciones	El usuario debe iniciar sesión.
Post condiciones	Ninguna
Flujo Básico	1.- El usuario accede a la plataforma principal del sistema.

	<p>2.- El usuario selecciona la pestaña de control.</p> <p>3.-El sistema muestra la plataforma de control.</p> <p>4.-El usuario selecciona el botón de apagar motor.</p> <p>5.-El sistema muestra el estado de motor apagado.</p>
--	---

Fuente: Elaboración propia, 2017

Tabla 3.13 Caso de Uso Encender Luces

CASO DE USO: Encender luces	
Caso de uso	Encender luces
Resumen	El usuario enciende las luces a través del botón de encender.
Actor	Usuario
Precondiciones	El usuario debe iniciar sesión.
Post condiciones	Ninguna
Flujo Básico	<p>1.- El usuario accede a la plataforma principal del sistema.</p> <p>2.- El usuario selecciona la pestaña de control.</p> <p>3.-El sistema muestra la plataforma de control.</p> <p>4.-El usuario selecciona el botón de encender de luces.</p> <p>5.-El sistema muestra el estado de encendido de luces.</p>

Fuente: Elaboración propia, 2017

Tabla 3.14 Caso de Uso Apagar Luces

CASO DE USO: Apagar luces	
Caso de uso	Apagar luces
Resumen	El usuario apaga las luces a través del botón de apagar desde la plataforma de control.
Actor	Usuario

Precondiciones	El usuario debe iniciar sesión.
Post condiciones	Ninguna
Flujo Básico	<p>1.- El usuario accede a la plataforma principal del sistema.</p> <p>2.- El usuario selecciona la pestaña de control.</p> <p>3.-El sistema muestra la plataforma de control.</p> <p>4.-El usuario selecciona el botón de apagar de luces.</p> <p>5.-El sistema muestra el estado de apagado de luces.</p>

Fuente: Elaboración propia, 2017

Tabla 3.15 Caso de Uso Encender Ventilador

CASO DE USO: Encender ventilador	
Caso de uso	Encender ventilador
Resumen	El usuario enciende ventilador a través del botón de encender.
Actor	Usuario
Precondiciones	El usuario debe iniciar sesión.
Post condiciones	Ninguna
Flujo Básico	<p>1.- El usuario accede a la plataforma principal del sistema.</p> <p>2.- El usuario selecciona la pestaña de control.</p> <p>3.-El sistema muestra la plataforma de control.</p> <p>4.-El usuario selecciona el botón de encender de ventilador.</p> <p>5.-El sistema muestra el estado de encendido de ventilador.</p>

Fuente: Elaboración propia, 2017

Tabla 3.16 Caso de Uso Apagar Ventilador

CASO DE USO: Apagar ventilador	
Caso de uso	Apagar ventilador

Resumen	El usuario apaga ventilador a través del botón de apagado desde la plataforma de control.
Actor	Usuario
Precondiciones	El usuario debe iniciar sesión.
Post condiciones	Ninguna
Flujo Básico	<ol style="list-style-type: none"> 1.- El usuario accede a la plataforma principal del sistema. 2.- El usuario selecciona la pestaña de control. 3.-El sistema muestra la plataforma de control. 4.-El usuario selecciona el botón de apagar ventilador. 5.-El sistema muestra el estado de apagado del ventilador.

Fuente: Elaboración propia, 2017

Tabla 3.17 Caso de Uso Encender Calefactor

CASO DE USO: Encender calefactor	
Caso de uso	Encender calefactor
Resumen	El usuario enciende calefactor a través del botón de encender.
Actor	Usuario
Precondiciones	El usuario debe iniciar sesión.
Post condiciones	Ninguna
Flujo Básico	<ol style="list-style-type: none"> 1.- El usuario accede a la plataforma principal del sistema. 2.- El usuario selecciona la pestaña de control. 3.-El sistema muestra la plataforma de control. 4.-El usuario selecciona el botón de encender de calefactor. 5.-El sistema muestra el estado de encendido de calefactor.

Fuente: Elaboración propia, 2017

Tabla 3.18 Caso de Uso Apagar Calefactor

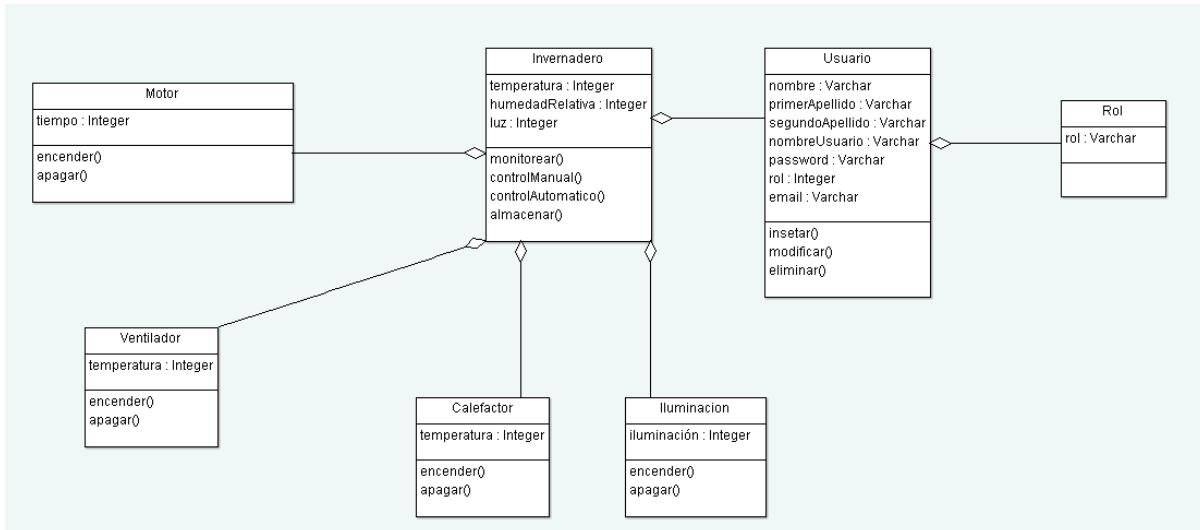
CASO DE USO: Apagar calefactor	
Caso de uso	Apagar calefactor

Resumen	El usuario apaga calefactor a través del botón de apagado desde la plataforma de control.
Actor	Usuario
Precondiciones	El usuario debe iniciar sesión.
Post condiciones	Ninguna
Flujo Básico	<p>1.- El usuario accede a la plataforma principal del sistema.</p> <p>2.- El usuario selecciona la pestaña de control.</p> <p>3.-El sistema muestra la plataforma de control.</p> <p>4.-El usuario selecciona el botón de apagar calefactor.</p> <p>5.-El sistema muestra el estado de apagado del calefactor.</p>

Fuente: Elaboración propia, 2017

3.4.3 Diagrama de Clases

Diagrama 3.5 Diagrama de Clases



Fuente: Elaboración Propia

3.4.3.1 Requerimientos Funcionales del Sistema

Muchos de los problemas de la ingeniería de software provienen de la imprecisión en la especificación de requerimientos.

Por esta razón, la especificación de requerimientos funcionales del sistema debe ser completa, es decir, que todos los servicios solicitados por el usuario estén definidos y no tengan definiciones contradictorias.

Los requerimientos funcionales del proyecto se observan en la siguiente tabla:

Tabla 3.19 Requerimientos Funcionales del Sistema

Código	Nombre del requerimiento	Caso de uso	Actor
RF1	La aplicación web debe permitir controlar el acceso al sistema a través de usuario y password.	Inicio de sesión	Usuario y Aplicación
RF2	El sistema debe permitir al usuario monitorear el estado de las variables ambientales de: temperatura, humedad relativa e iluminación del invernadero de forma remota a través de internet.	Monitoreo de Variables Ambientales	Usuario
RF3	El sistema debe permitir visualizar a través de la aplicación web las diferentes lecturas registradas por los sensores.	Datos Históricos	Raspberry y Usuario
RF4	El sistema debe generar un archivo log donde se registre las acciones de gestión de usuarios..	Archivo Log de Aplicación	Aplicación
RF5	El sistema debe almacenar en la base de datos las diferentes lecturas a partir de los sensores de temperatura, humedad y luz.	Almacenar datos	Arduino
RF6	El sistema debe permitir accionar manualmente los mecanismos de encendido de motor,ventilación, calefacción e iluminación por parte del usuario de forma remota.	Control individual	Usuario
RF7	El sistema permitirá obtener reportes de parámetros ambientales en un periodo de tiempo	Módulo de Reportes	Usuario

Fuente: Elaboración Propia, 2017

3.4.3.2 Requerimientos No Funcionales del Sistema

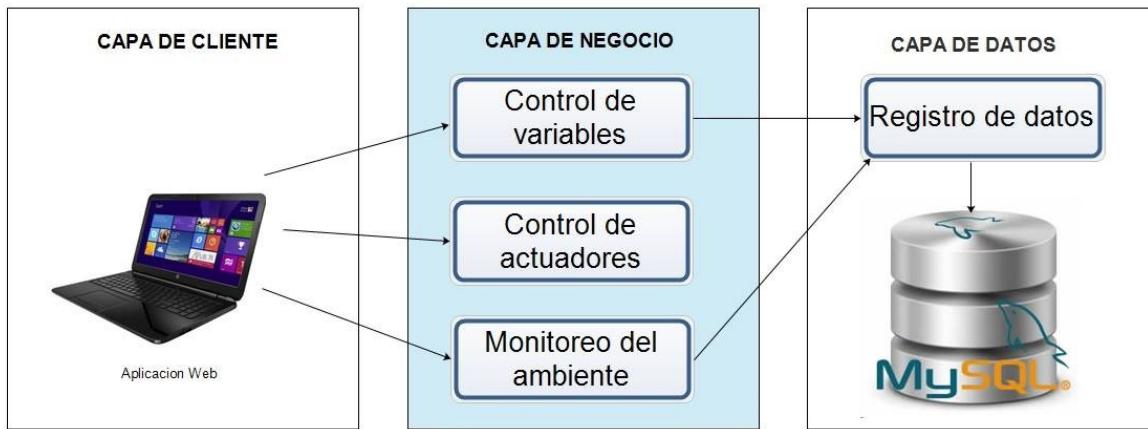
Los requerimientos no funcionales son restricciones de los servicios o funciones ofrecidos por el sistema, es decir, son aquellos que no se refieren directamente a las funciones específicas que proporciona el sistema, sino a las propiedades emergentes de éste como la fiabilidad, el tiempo de respuesta y la capacidad de almacenamiento.

Los requerimientos no funcionales del proyecto son los siguientes:

- a).** El sistema debe tener una interfaz amigable para el usuario y así permitir un fácil uso.
- b).** El sistema debe permitir el acceso desde cualquier sitio con acceso a internet.
- c).** La aplicación web debe ocupar la mayor parte de la pantalla para poder visualizar las diferentes funciones.
- d).** El sistema debe procesar los datos en línea y de forma rápida
- e).** El sistema debe mostrar mensajes de errores de una forma clara, que permita al usuario comprenderlos fácilmente.
- f).** La aplicación web debe tener compatibilidad de visualización con las últimas versiones de los navegadores.

3.4.4 Diagrama de Capas del Sistema

Diagrama 3.6 Diagrama de Capas

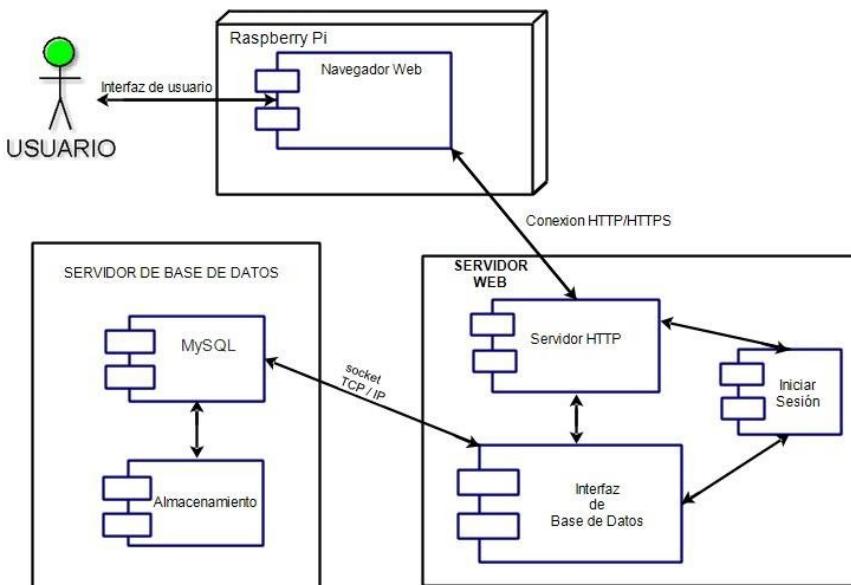


Fuente: Elaboración Propia, 2017

3.4.5 Diagrama de Despliegue

En el diagrama 3.4 se detalla el diagrama de despliegue del sistema, el cual describe físico de la información generada por el programa de software en los componentes de hardware del sistema.

Diagrama 3.7 Diagrama de Despliegue



Fuente: Elaboración propia, 2017

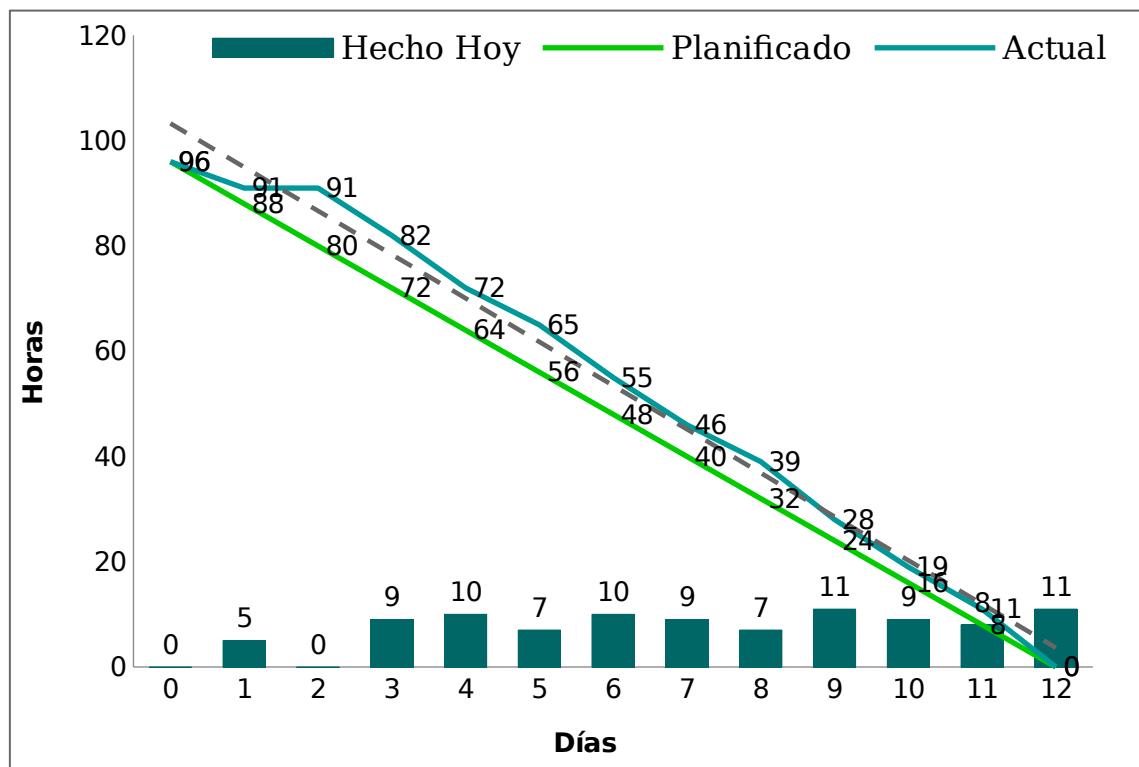
3.4.6 Burndown Chart Sprint 1

Tabla 3.20 Tabla Burndown Chart – Sprint 1

Día	Burned down		Balance		Hecho Hoy
	Planificado	Actua 1	Planificado	Actua 1	
0			96	96	N/A
1	8	4	88	92	4
2	8	6	80	86	6
3	8	0	72	86	0
4	8	4	64	82	4
5	8	6	56	76	6
6	8	4	48	72	4
7	8	4	40	68	4
8	8	12	32	56	12
9	8	10	24	46	10
10	8	12	16	34	12
11	8	11	8	23	11
12	8	12	0	11	12

Fuente: Elaboración propia, 2017

Diagrama 3.8 Burndown Chart – Sprint 1



Fuente: Elaboración propia, 2017

3.5 REUNIÓN DE PLANIFICACIÓN DEL SPRINT 2 – 2017/06/21

El objetivo de la reunión es identificar los objetivos que deberán cumplirse en la iteración y establecer un procedimiento para llevarlos a cabo.

3.5.1 Asistentes

Sebastián Alcalá – *Product Owner*

Mirna Albarado / Cristian Encalada – *Equipo de Desarrollo*

Ing. Pavel Cáceres Torrez – *Scrum Master*

3.5.2 ¿Qué se completará en este Sprint? (Sprint Backlog)

En el segundo Sprint, se realiza la selección de componentes y construcción del prototipo de cultivo, esto es:

- Seleccionar Herramientas para el Desarrollo de Software
- Seleccionar Dispositivo de Control y Almacenamiento de Datos
- Seleccionar los Dispositivo de Adquisición de Datos
- Seleccionar Sensores
- Construir el prototipo hidropónico

3.5.2.1 Herramientas para el Desarrollo de Software.

Motor de Base de Datos.

A la hora de realizar una aplicación, una parte importante para el desarrollo son las bases de datos. Existen multitud de base de datos como Oracle, MySQL, PostGresSQL, NoSQL (entre ellas MariaDB), etc.

A continuación, se mencionan las razones por las que se utiliza MySQL como Motor de Base de Datos:

- Es Multiplataforma: Soportado en S.O. como Windows, Linux y Mac.
- Existe gran cantidad de documentación.

- El aprendizaje es sencillo: Simplemente con conocer el estándar de SQL podemos manejar la base de datos MySQL si ningún problema.
- MySQL es una base de datos ampliamente probada por distintos usuarios y empresas con alto éxito.
- El coste total de inicio es ampliamente inferior al de Oracle o Microsoft SQL Sever.
- MySQL es escalable.

3.5.2.2 Lenguaje de Programación de la Plataforma Web PHP

Se utiliza PHP como lenguaje de lado del servidor debido a que forma parte del software libre y tiene varias ventajas, como ser:

- **Velocidad:** No solo la velocidad de ejecución, la cual es importante, sino además no crear demoras en la máquina. PHP se integra muy bien junto a otro software, especialmente bajo ambientes Unix, cuando se configura como módulo de Apache, está listo para ser utilizado.
- **Estabilidad:** PHP utiliza su propio sistema de administración de recursos y dispone de un sofisticado método de manejo de variables, conformando un sistema robusto y estable.
- **Seguridad:** El sistema debe poseer protecciones contra ataques. PHP provee diferentes niveles de seguridad, estos pueden ser configurados desde el archivo *.ini*
- **Simplicidad:** Se debe permitir a los programadores generar código productivamente en el menor tiempo posible. Usuarios con experiencia en C, C++ y C# pueden utilizar PHP rápidamente.
- **Conectividad:** PHP dispone de una amplia gama de librerías, y agregar extensiones es muy sencillo.

3.5.2.3 Dispositivo de Control y Almacenamiento de Datos - Raspberry Pi

Debido al alto costo de los Dispositivos de Almacenamiento de Datos (Servidores), Dispositivos de Control y Automatización como PLCs, el computador de placa reducida Raspberry Pi se torna en una alternativa atractiva debido a su bajo coste, su plataforma de hardware abierta y a la posibilidad de ser utilizado como servidor

para la plataforma web del proyecto. En la tabla 3.18 se comparan diferentes modelos de Raspberry Pi:

Tabla 3.21 Comparativa de Modelos Raspberry Pi

Modelos de Raspberry Pi	Raspberry Pi A	Raspberry Pi A+	Raspberry Pi B	Raspberry Pi B+	Raspberry Pi 2 B
SoC (Chip)	Broadcom BCM 2835	Broadcom BCM 2835	Broadcom BCM 2835	Broadcom BCM 2835	Broadcom BCM 2836
CPU	ARM1176JZF-S a 700 MHz	ARM Cortex-A7 cuatro núcleos a 900 Mhz			
GPU	VideoCore IV a 250 Mhz				
Memoria RAM	256 MB a 400 Mhz	256 MB a 400 Mhz	512 MB a 400 Mhz	512 MB a 400 Mhz	1 GB a 450 Mhz
Entradas de vídeo	Cámara CSI				
Salidas de vídeo	HDMI 1.4, conector RCA	HDMI 1.4, conector TRRS			
Salidas de audio	Jack de 3.5 mm (auriculares), HDMI				
Conectores USB 2.0	1	1	2	4	4
Tarjetas de almacenamiento	SD	microSD	SD	microSD	microSD
Conexión a red	No	No	Ethernet 10/100 Mbit/sg	Ethernet 10/100 Mbit/sg	Ethernet 10/100 Mbit/sg
Interfaz periféricos (GPIO)	8	17	8	17	17
Tamaño	85.6 x 56.5 mm	65 x 56.5 mm	85.6 x 56.5 mm	85.6 x 56.5 mm	85.6 x 56.5 mm
Peso	45 gramos	23 gramos	45 gramos	45 gramos	45 gramos
Consumo	1.5W/5V	1W/5V	3.5W/5V	3W/5V	5V
Precio	25€	20€	30€	35€	35€

Fuente: computerhoy.com/noticias/hardware/raspberry-pi-que-modelo-me-compro-23811, 2017

El modelo que se consideró para el desarrollo del proyecto fue la Raspberry Pi 2 Modelo B. Este modelo representa la gama alta en lo que a Raspberry Pi se refiere. Ofrece una capacidad cómputo muy aceptable gracias a sus 4 núcleos ARM y un rendimiento notable gracias a su memoria RAM de 1GB. La Figura 3.2 muestra el esquema de este modelo:

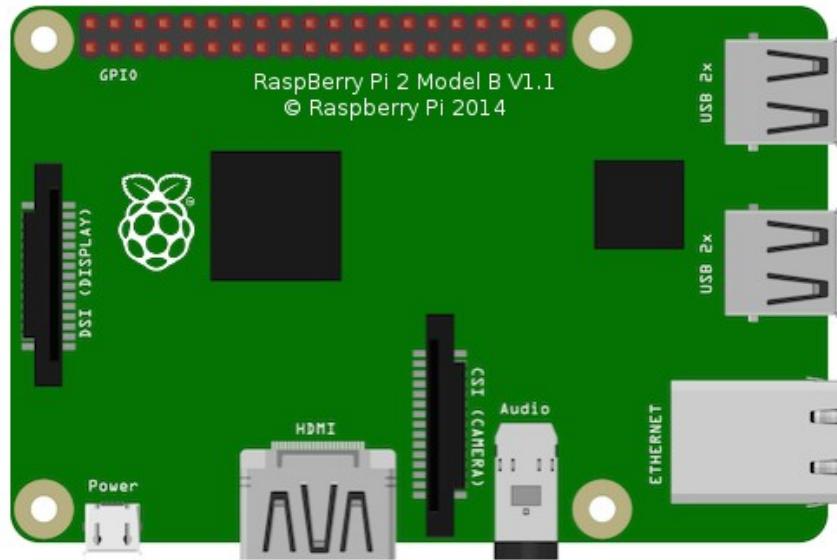


Figura 3.26 Esquema de la Raspberry Pi Modelo 2 v1.1.

Fuente: <https://www.hackster.io/idreams/boost-usb-current-in-raspberry-pi-a1531d>, 2017

Las especificaciones de este modelo se listan en la Tabla 3.19:

Tabla 3.22 Especificaciones Raspberry Pi Modelo B

Especificaciones Raspberry Pi Modelo B	
CPU	Broadcom BCM2836
Velocidad CPU (estándar)	900Mhz
RAM	1 GB
Capacidad de memoria estable	Full SD
Ethernet	10/100 Mbps
HDMI	Sí
Vídeo compuesto	Sí (en jack)
Puertos USB	2.0 (x4)
GPIO Pins	40
3.5mm audio jack	Sí
Camera Interface Port (CSI-2)	Sí (x1)
LCD Display Interface Ports (DSI)	Sí (x1)
Consumo	5V, 1,2 - 2A, 6 -10W
Dimensiones	85 x 49 x 17mm

Fuente: Elaboración Propia, 2017

3.5.2.4 Dispositivo de Adquisición de Datos - Arduino

Al igual que los dispositivos de Control y Almacenamiento de Datos, los Dispositivos de Adquisición de Datos (DAQ), la placa Arduino es una gran alternativa para el desarrollo de prototipos.

Debido a su bajo coste, su plataforma de hardware abierta y a la posibilidad de ser usado como un dispositivo versátil, que junto al computador de placa reducida Raspberry Pi utilizado como servidor para la plataforma web satisfacen de forma muy aceptable los requerimientos del proyecto.

Para el proyecto se elige la placa Arduino por las siguientes razones:

- Es libre y extensible.
- Tiene una gran comunidad.
- Tiene un entorno de programación es multiplataforma.
- Es una placa económica.
- Es una placa reutilizable y versátil.

La tabla 3.20 muestra las características principales de placas Arduino más utilizadas:

Tabla 3.23 Comparativa de placas Arduino

Característica de Arduino	UNO	Mega 2560	Leonardo	DUE
Tipo de microcontrolador	Atmega 328	Atmega 2560	Atmega 32U4	AT91SAM3 X8E
Velocidad de reloj	16 MHz	16 MHz	16 MHz	84 MHz
Pines digitales de E/S	14	54	20	54
Entradas analógicas	6	16	12	12
Salidas analógicas	0	0	0	2 (DAC)
Memoria de programa (Flash)	32 Kb	256 Kb	32 Kb	512 Kb
Memoria de datos (SRAM)	2 Kb	8 Kb	2.5 Kb	96 Kb
Memoria auxiliar (EEPROM)	1 Kb	4 Kb	1 Kb	0 Kb

Fuente: www.arduino.cc/en/Main/Products, 2017

Para la selección de la placa Arduino UNO se tomó en cuenta la memoria de datos y programa además de la cantidad de Entradas y Salidas Analógicas y Digitales que posee.

Las características de la placa Arduino UNO se muestran a continuación en la Tabla 3.21:

Tabla 3.24 Características de la placa Arduino UNO

Características de la Placa Arduino UNO	
Microcontrolador	Atmega328
Voltaje de operación	5V
Voltaje de entrada (Recomendado)	7 – 12V
Voltaje de entrada (Límite)	6 – 20V
Pines para entrada- salida digital.	14 (6 pueden usarse como salida de PWM)
Pines de entrada analógica.	6
Corriente continua por pin IO	40 Ma
Corriente continua en el pin 3.3V	50 mA
Memoria Flash	32 KB (0,5 KB ocupados por el bootloader)
SRAM	2 KB
EEPROM	1 KB
Frecuencia de reloj	16 MHz

Fuente: www.iescamp.es/miarduino/2016/01/21/placa-arduino-uno/ 2017

3.5.2.5 Selección de Sensores

Sensor de Intensidad Luminosa

Para este sensor se utiliza un LDR con la finalidad de medir la intensidad de luz usando la variación de su resistencia. Para la selección del LDR se tomó en cuenta su rango de variación de resistencia, ya que al tener un rango amplio de resistencia se puede diferenciar de mejor manera las variaciones de la incidencia de luz.

La Tabla 3.22 muestra las características principales del LDR seleccionado:

Tabla 3.25 Características del sensor LDR

Características del sensor LDR		
Resistencia Mínima	12.6 ohm	
Resistencia Máxima	1 Mohm	

Fuente: Elaboración Propia, 2017

Sensor de Humedad y Temperatura

El sensor de humedad es un dispositivo electrónico que mide la humedad relativa en un área determinada. La humedad relativa es la cantidad de agua que contiene un gas expresada en porcentaje de la cantidad que el gas tendría en estado de saturación, a la misma temperatura y presión absoluta. Para la selección del sensor de humedad relativa se consideraron distintos sensores existentes en el mercado como se muestra en la Tabla 3.23:

Tabla 3.26 Comparativa de Sensores de Humedad

PARÁMETRO ELEMENTO	Voltaje de operación	Corriente de Consumo	Factor de Escala	Rango	Tipo de Salida
DHT11	3-5V	2.5mA	1 %	20%-95%	DIGITAL
SHT15	2.4-5V	1mA	0.05%	20%-60%	DIGITAL
SHT25	2.1-3.6 V	0.15uA	0.04%	10%-90%	I2C
HIH-400	4-5.8V	200uA		0%-100%	ANALOGICA

Para esta etapa se utiliza el sensor de temperatura y humedad relativa DHT11, que presenta una salida digital que permite integrarlo de forma sencilla a los dispositivos de control y adquisición de datos seleccionados anteriormente.

Las características del sensor se muestran a continuación en la Tabla 3.24:

Tabla 3.27 Características del Sensor DHT11

Características Sensor DHT11	
Alimentación	3Vdc ≤ Vcc ≤ 5Vdc
Señal de Salida	Digital
Rango de medida Temperatura	De 0 a 50 °C
Precisión Temperatura	±2 °C
Resolución Temperatura	0.1°C
Rango de medida Humedad	De 20% a 90% RH
Precisión Humedad	4% RH
Resolución Humedad	1%RH
Tiempo de respuesta	1s
Tamaño	12 x 15.5 x 5.5mm

Fuente: omniblug.com/sensor-temperatura-humedad-DHT11-DHT22.html

3.5.2.6 Construir el prototipo hidropónico

En la siguiente figura se muestra el prototipo del cultivo hidropónico construido.



Figura 3.27 Prototipo del Cultivo Hidropónico

Fuente: Elaboración Propia, 2017

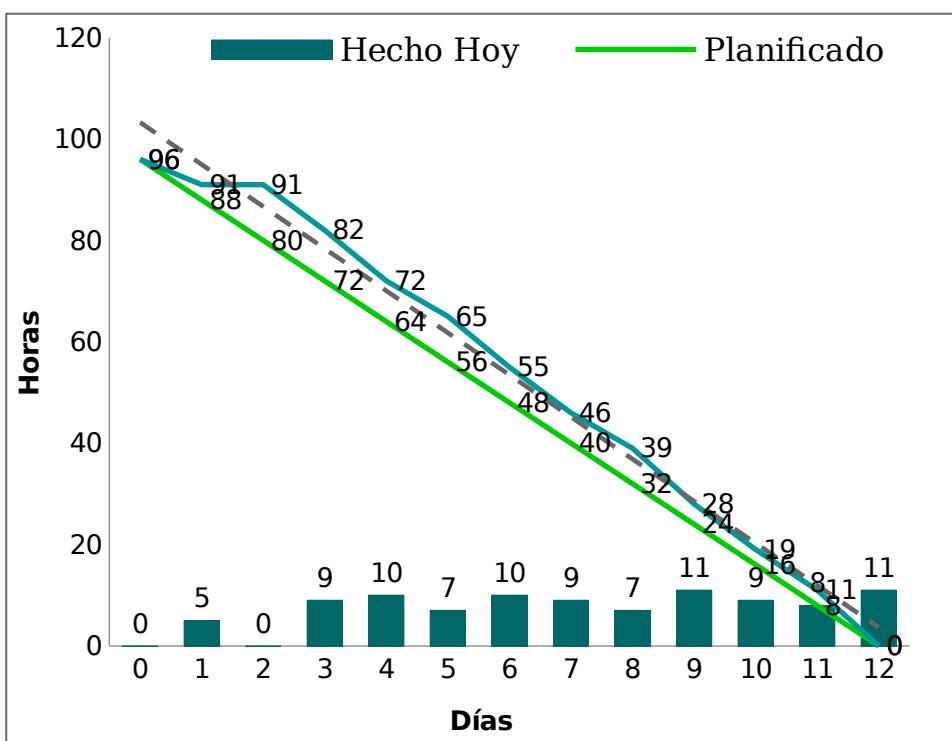
3.5.3 Burndown Chart Sprint 2

Tabla 3.28 Tabla Burndown Chart – Sprint 2

Día	Burned down		Balance		
	Planificado	Actua l	Planificado	Actua l	Hecho Hoy
0			96	96	N/A
1	8	5	88	91	5
2	8	1	80	90	1
3	8	11	72	79	11
4	8	7	64	72	7
5	8	9	56	63	9
6	8	6	48	57	6
7	8	7	40	50	7
8	8	9	32	41	9
9	8	9	24	32	9
10	8	11	16	21	11
11	8	9	8	12	9
12	8	12	0	0	12

Fuente: Elaboración propia, 2017

Diagrama 3.9 Burndown Chart – Sprint 2



Fuente: Elaboración propia, 2017

3.6 REUNIÓN DE PLANIFICACIÓN DEL SPRINT 3 – 2017/07/07

El objetivo de la reunión es identificar los objetivos que deberán cumplirse en la iteración y establecer un procedimiento para llevarlos a cabo.

3.6.1 Asistentes

Sebastián Alcalá – *Product Owner*

Mirna Albarado / Cristian Encalada – *Equipo de Desarrollo*

Ing. Pavel Cáceres Torrez – *Scrum Master*

3.6.2 ¿Qué se completará en este Sprint? (Sprint Backlog)

En el tercer Sprint, se realiza el desarrollo de la aplicación, esto es:

- Definir los roles de usuario del sistema
- Crear Modelo Usuario
- Crear Controlador Usuario
- Crear Vista Usuario
- Crear Modelo Administrador
- Crear Controlador Administrador
- Crear Vista Administrador
- Realizar Autenticación de Usuarios
- Realizar Validación de Usuarios
- Realizar Registro de Usuarios
- Recuperar y mostrar datos de monitoreo en la aplicación Web

3.6.2.1 Definir los roles de Usuario

Se tienen los siguientes roles de Usuario para la aplicación web:

- **Administrador:** Podrá crear, dar de baja y asignar roles a otros usuarios del sistema.
- **Cliente:** Tendrá acceso a la información y datos del sistema, además de poder controlar los dispositivos de forma remota.

3.6.2.2 Crear la Interfaz de Usuario (MVC)

A continuación, se muestra la página principal de Usuario desarrollada en base al Modelo Vista Controlador MVC



Figura 3.28 Interfaz Web de Usuario

Fuente: Elaboración Propia, 2017

3.6.2.3 Crear la Interfaz de Administrador (MVC)

A continuación, se muestra la página principal de Administrador desarrollada en base al Modelo Vista Controlador MVC

La captura de pantalla muestra la página principal del sistema de gestión de usuarios. En la parte superior izquierda, hay un menú lateral oscuro con las siguientes opciones: Bienvenido, Principal, Nuevo Usuario/Administrador, Monitoreo, Datos históricos, Control, Reportes y Acerca de... En la parte superior derecha, se encuentran los botones 'Salir' y 'Página Principal / Gestión de Usuarios'. El centro de la pantalla muestra una sección titulada 'Lista de Usuarios' con un formulario para 'Mostrar' (selección de 10 registros) y un campo de búsqueda 'Buscar: Introduzca los datos'. Una tabla detallada muestra la lista de usuarios registrados. Los datos de la tabla son:

Figura 3.29 Interfaz Web de Administrador

Fuente: Elaboración Propia, 2017

3.6.2.4 Realizar la Validación de Usuarios

En la siguiente figura se muestra la validación de inicio de Sesión de Usuario desarrollada en base al Modelo Vista Controlador MVC.

The screenshot shows a login form titled "Iniciar sesión". It contains two text input fields: "Nombre de Usuario" (Username) and "Contraseña" (Password), both with placeholder text "Introduzca su nombre de Usuario" and "Introduzca su contraseña" respectively. Below the inputs is a blue "Iniciar Sesión" (Login) button. At the bottom of the form is a link "Registrar una cuenta" (Create account). Above the form, a red error message reads "Por Favor, Inicie Sesión para acceder a la plataforma" (Please log in to access the platform).

Figura 3.30 Validación de Inicio de Sesión

Fuente: Elaboración Propia, 2017

En la siguiente figura se muestra la validación de usuario y contraseña desarrollada en base al Modelo Vista Controlador MVC.

This screenshot displays a login form with identical layout and fields to Figure 3.30. The "Nombre de Usuario" field has the placeholder "Introduzca su nombre de Usuario". The "Contraseña" field has the placeholder "Introduzca su contraseña". A red error message above the form states "Usuario o Contraseña Incorrecta. Intente nuevamente" (User or Password Incorrect. Try again). The "Iniciar Sesión" button and "Registrar una cuenta" link are also present.

Figura 3.31 Validación de Usuario y Contraseña

Fuente: Elaboración Propia, 2017

3.6.2.5 Realizar el Registro de Usuarios

A continuación, se muestra la página de Registro de Usuario normal desarrollada en base al Modelo Vista Controlador MVC

The screenshot shows a user registration form titled "Registrar nueva cuenta". The form fields are as follows:

- Nombre: A text input field containing "Introduzca su nombre".
- Apellido Paterno: A text input field containing "Introduzca su apellido paterno".
- Apellido Materno: A text input field containing "Introduzca su apellido materno".
- Dirección de correo electrónico: A text input field containing "Introduzca su correo electrónico".
- Usuario: A text input field containing "nombreUsuario".
- Contraseña: A text input field containing "Contraseña".
- Confirmar contraseña: A text input field containing "Confirmar Contraseña".
- Registrarse: A blue button labeled "Registrarse".
- Página de inicio de sesión: A link labeled "Página de inicio de sesión".

Figura 3.32 Interfaz Registro nuevo Usuario

Fuente: Elaboración Propia, 2017

El registro de usuario también se puede realizar desde la cuenta del administrador, permitiendo registrar un nuevo administrador en caso de ser necesario.

3.6.2.6 Interfaz de Monitoreo de Variables Ambientales

A continuación, se muestra la página de Monitoreo de Variables Ambientales enviadas a través de la placa NodeMCU.



Figura 3.33 Interfaz de Monitoreo de Variables Ambientales

Fuente: Elaboración Propia, 2017

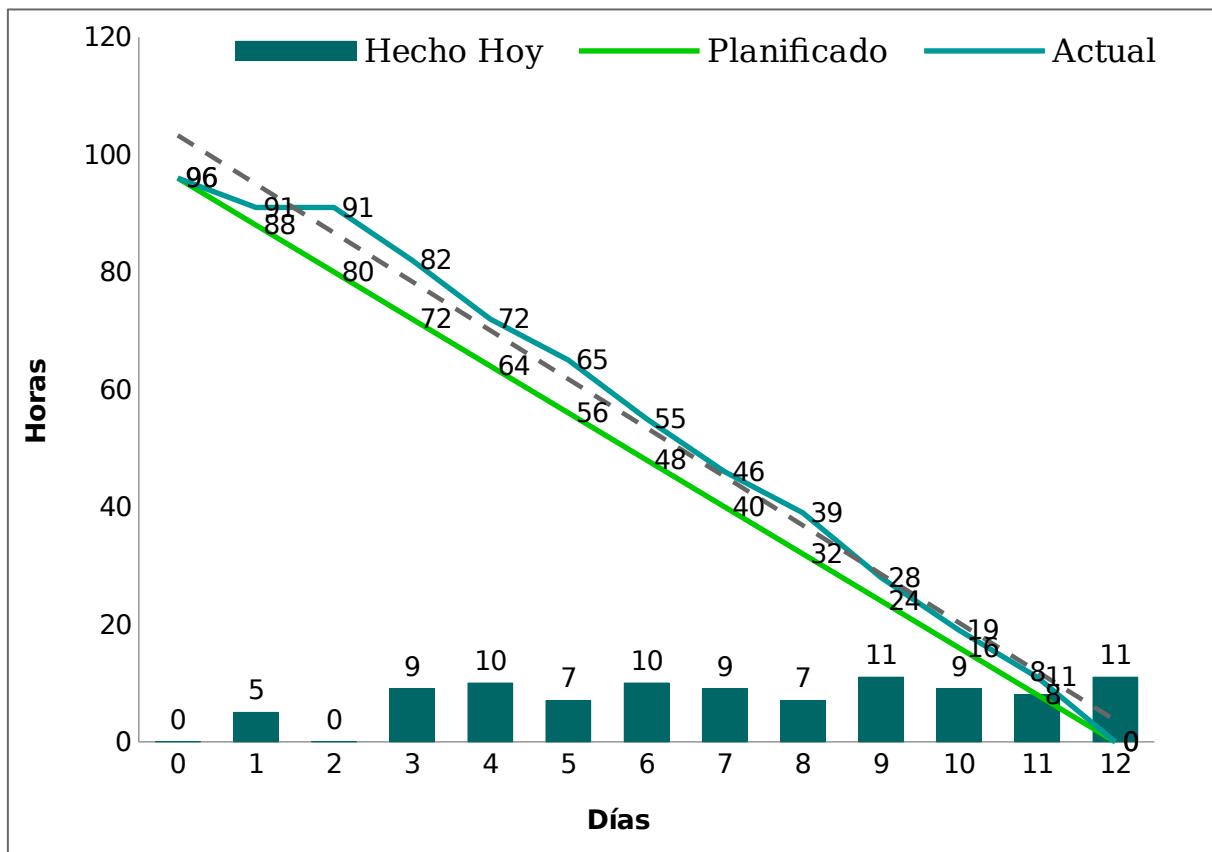
3.6.3 Burndown Chart Sprint 3

Tabla 3.29 Tabla Burndown Chart – Sprint 3

Día	Burned down		Balance		
	Planificado	Actua 1	Planificado	Actua 1	Hecho Hoy
0			96	96	N/A
1	8	6	88	90	6
2	8	9	80	81	9
3	8	3	72	78	3
4	8	11	64	67	11
5	8	7	56	60	7
6	8	9	48	51	9
7	8	4	40	47	4
8	8	11	32	36	11
9	8	11	24	25	11
10	8	9	16	16	9
11	8	7	8	9	7
12	8	3	0	6	3

Fuente: Elaboración Propia, 2017

Diagrama 3.10 Burndown Chart – Sprint 3



Fuente: Elaboración Propia, 2017

3.7 REUNIÓN DE PLANIFICACIÓN DEL SPRINT 4 – 2017/07/25

El objetivo de la reunión es identificar los objetivos que deberán cumplirse en la iteración y establecer un procedimiento para llevarlos a cabo.

3.7.1 Asistentes

Sebastián Alcalá – *Product Owner*

Mirna Albarado / Cristian Encalada – *Equipo de Desarrollo*

Ing. Pavel Cáceres Torrez – *Scrum Master*

3.7.2 ¿Qué se completará en este Sprint? (Sprint Backlog)

En el cuarto Sprint, se realiza el desarrollo de la aplicación, esto es:

- Instalar el SO Raspbian en la Raspberry Pi 2
- Crear Modelo Control Individual
- Crear Controlador Control Individual
- Crear Vista Control Individual
- Programar Arduino Uno para envío de variables Ambientales
- Crear Reporte de Temperatura
- Crear Reporte de Humedad
- Crear Reporte de Iluminación

3.7.2.1 Instalación del SO Raspbian

En la siguiente imagen se muestra el SO. Raspbian Instalado en la Raspberry Pi 2.

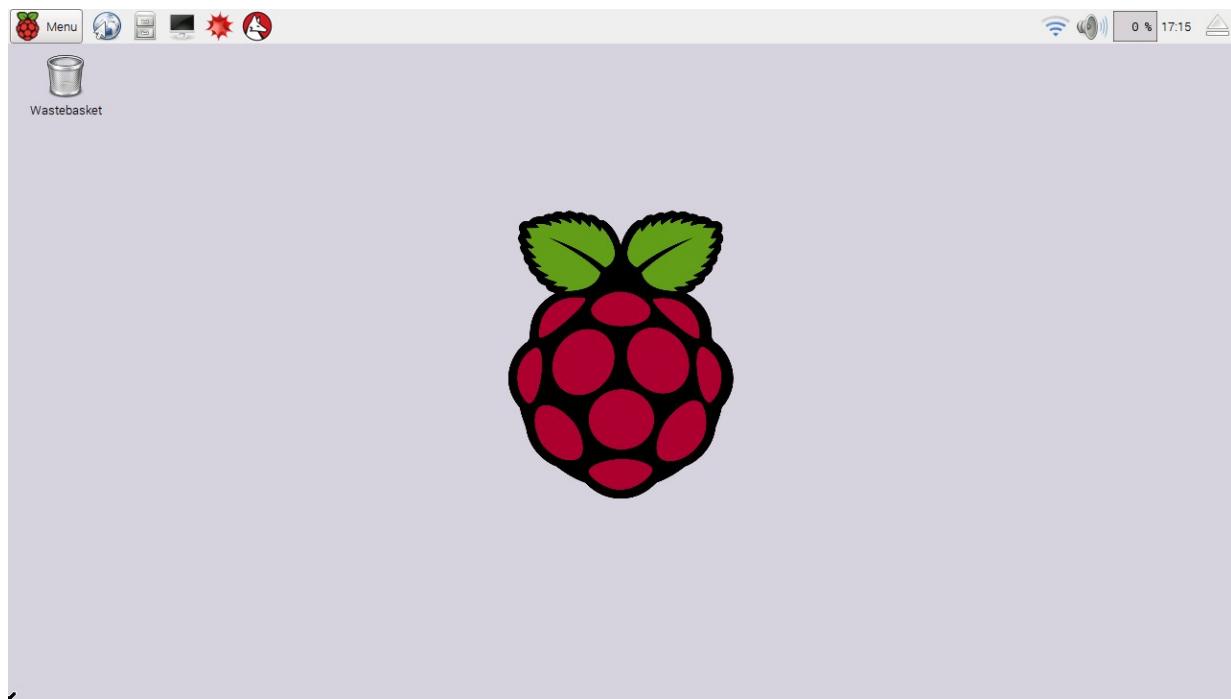


Figura 3.34 SO. Raspbian instalado en la Raspberry Pi 2

Fuente: Elaboración Propia, 2017

3.7.2.2 Interfaz de Control Individual

En la siguiente imagen se muestra la interfaz de control individual de dispositivos eléctricos a través de la aplicación web

The screenshot shows a web-based monitoring and control system for agricultural cultivation. On the left, a sidebar menu includes 'Principal', 'Monitoreo', 'Historial', 'Control', 'Reportes', and 'Contactos'. The main content area is titled 'Control Individual' and contains four cards: 'Motor' (blue), 'Iluminación' (yellow), 'Ventilación' (green), and 'Calefacción' (red). Each card has 'Encender' and 'Apagar' buttons. Below the cards is a table titled 'Estado de los Dispositivos' with columns 'Dispositivo' and 'Estado'. The table lists: Calefactor (Apagado), Ventilador (Apagado), Motor (Apagado), and Iluminación (Apagado). At the bottom is a section titled 'Estado Actual Modo Manual'.

Figura 3.35 Interfaz de Control Individual

Fuente: Elaboración Propia, 2017

3.7.2.3 Programación del Arduino para envío de variables ambientales

En la siguiente imagen se muestra la programación del Arduino Uno para el envío de variables ambientales a la Raspberry Pi 2.

```
mysql_temp_hum_rasp
#include <Ethernet.h>
#include <SPI.h>
#include <DHT.h>

// Configuración del Ethernet Shield
byte mac[] = {0xDE, 0xAD, 0xBE, 0xEF, 0xFF, 0xEE}; // Dirección MAC
byte ip[] = {192,168,1,100}; // Dirección IP del Arduino
byte server[] = {192,168,1,10}; // Dirección IP del servidor
IPAddress serverDns(8, 8, 8);
IPAddress gateway(192, 168, 1, 1);
EthernetClient client;
// Definimos variables
int luz;
int analog_pin = 0;
int pin_DHT=2; //El pin digital 2 será usado para el sensor DHT11 (temperatura y humedad)
int LED_pin=3; //El LED esta en el pin 3
int contador = 0; //Variable para guardar el conteo de los pulsos
DHT11 dht11(pin_DHT);

void setup(void)
{
  Serial.begin(9600); //Inicializamos el puerto serial
  pinMode(LED_pin,OUTPUT); // El LED esta en el pin 3
  Ethernet.begin(mac, ip, serverDns, gateway); // Inicializamos el Ethernet Shield
  delay(1000); // Esperamos 1 segundo de cortesía
}

void loop(void) {
  // En el loop iremos leyendo continuamente del sensor LDR
  int err;
  float temp, hum;
  //Mostramos en monitor Serial
  luz = analogRead(analog_pin);
  if((err = dht11.read(hum, temp)) == 0) // Si devuelve 0 es que ha leido bien
  {
    Serial.print("Temperatura: ");
    Serial.print(temp);
    Serial.print(" Humedad: ");
    Serial.print(hum);
    Serial.print("Luz = ");
    Serial.println(luz);
  }
  else
  {
    Serial.println();
    Serial.print("Error Num :");
    Serial.print(err);
    Serial.println();
  }
  if (luz>300) // Si hay luz, apago el led
  {
    digitalWrite(LED_pin, LOW); // Apagamos el LED
  }
  else if (luz<=300) // Si no hay luz, enciendo el led
  
```

```

        // Proceso de envío de muestras al servidor
if (client.connect(server, 80)>0)
{
    // Conexión con el servidor
    // http://192.168.1.10//arduino/temp1.php?valor1=15.5&valor2=9.5
    Serial.println("Conectando al servidor...");
    String x = "GET /datos.php";      // Enviamos los datos por GET
    x.concat("?valor1=");
    x.concat(temp);
    x.concat("&valor2=");
    x.concat(hum);
    x.concat("&valor3=");
    x.concat(luz);
    client.print(x);
    client.println(" HTTP/1.0");
    client.println("User-Agent: Arduino 1.0");
    client.println();
    Serial.println("Enviado al servidor");
}

if (!client.connected()) {
    Serial.println("Disconnected!");
}
client.stop();
client.flush();
// delay(3600000); // Espero 10 min antes de tomar otra muestra
delay(60000); // Espero 10 min antes de tomar otra muestra
}

```

Figura 3.36 Programación de Arduino Uno

Fuente: Elaboración Propia, 2017

3.7.2.4 Reporte de Temperatura

En la siguiente figura se muestra el reporte de temperatura utilizando HigCharts.



Figura 3.37 Reporte de Temperatura

Fuente: Elaboración Propia, 2017

3.7.2.5 Reporte de Humedad Relativa

En la siguiente figura se muestra el reporte de humedad relativa utilizando la librería HigCharts.

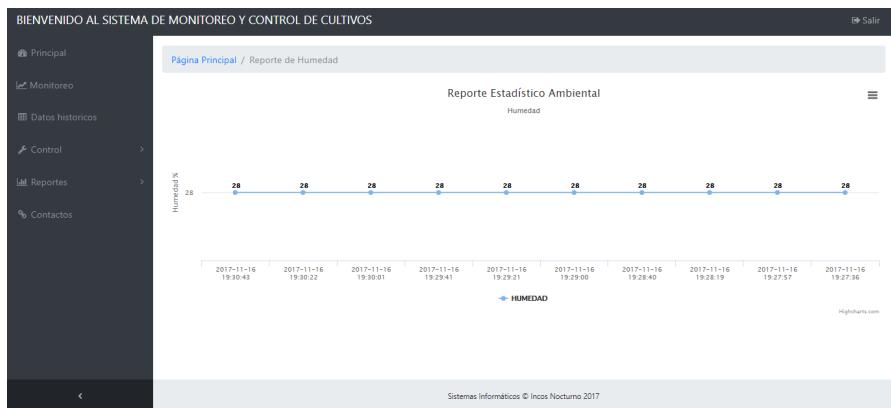


Figura 3.38 Interfaz de Control Individual

Fuente: Elaboración Propia, 2017

3.7.2.6 Reporte de Iluminación

En la siguiente figura se muestra el reporte de iluminación utilizando HigCharts.



Figura 3.39 Interfaz de Control Individual

Fuente: Elaboración Propia, 2017

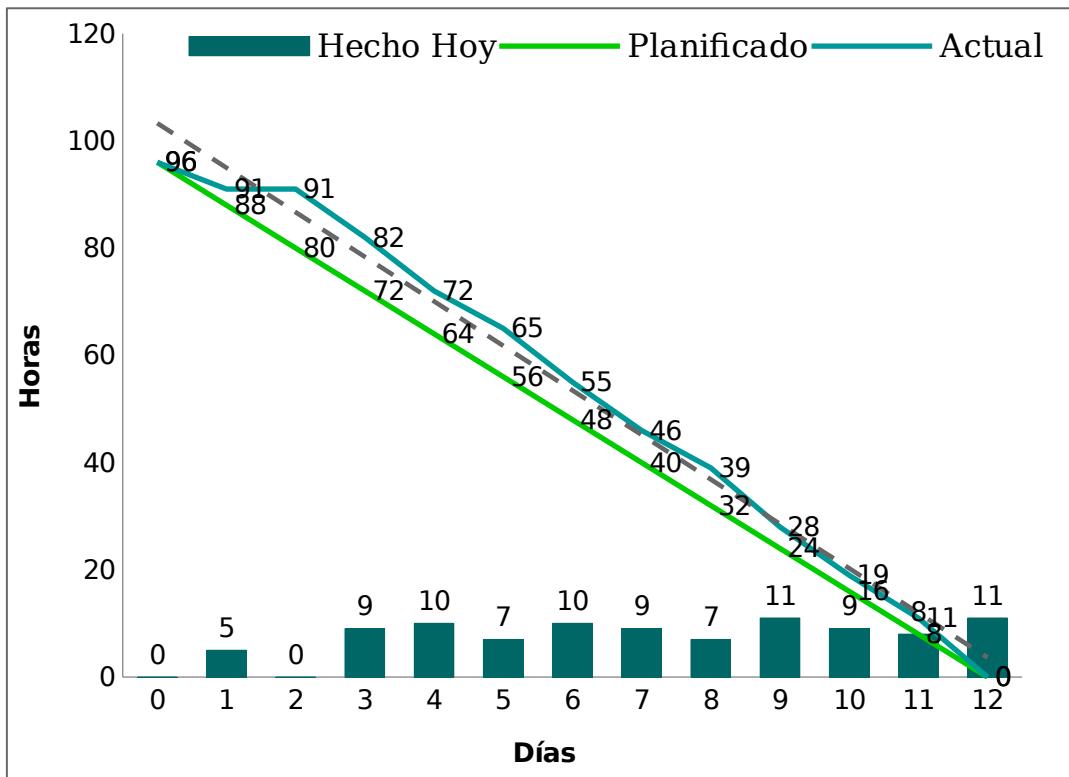
3.7.3 Burndown Chart Sprint 4

Tabla 3.30 Tabla Burndown Chart – Sprint 4

Día	Burned down		Balance		
	Planificado	Actua l	Planificado	Actua l	Hecho Hoy
0			96	96	N/A
1	8	5	88	91	5
2	8	0	80	91	0
3	8	9	72	82	9
4	8	10	64	72	10
5	8	7	56	65	7
6	8	10	48	55	10
7	8	9	40	46	9
8	8	7	32	39	7
9	8	11	24	28	11
10	8	9	16	19	9
11	8	8	8	11	8
12	8	11	0	0	11

Fuente: Elaboración Propia, 2017

Diagrama 3.11 Burndown Chart – Sprint 4



Fuente: Elaboración Propia, 2017

3.8 Arquitectura General del Sistema

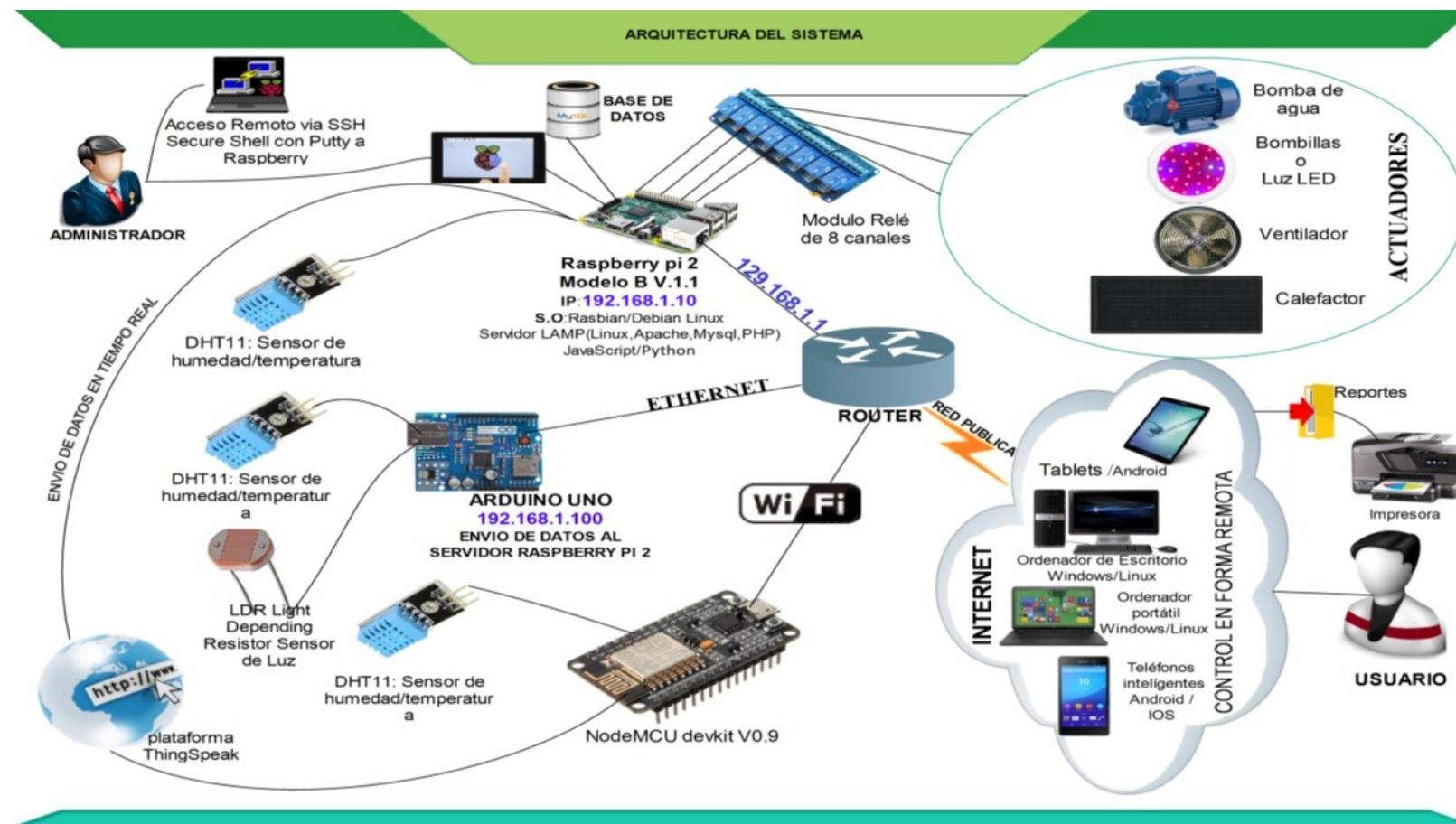


Figura 3.40 Arquitectura General del Sistema

Fuente: Elaboración Propia, 2017

3.8.1 Arquitectura del Hardware

Para el desarrollo del sistema se utiliza una arquitectura de hardware híbrida o mixta. En el diagrama 3.6 se muestra la arquitectura del hardware del proyecto.

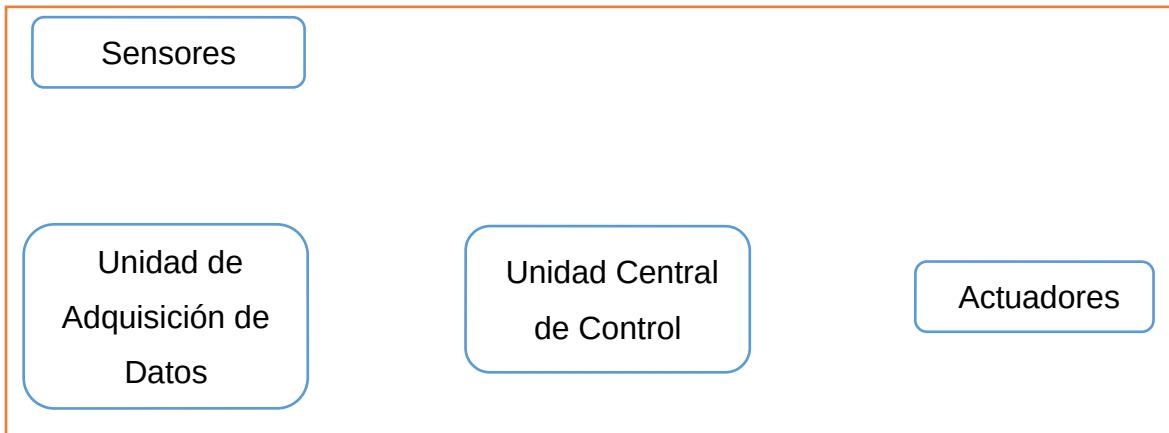


Diagrama 3.12 Arquitectura General del Hardware

Fuente: Elaboración Propia, 2017

En el diagrama 3.7 se muestra con mayor detalle los dispositivos que componen los diferentes subsistemas:

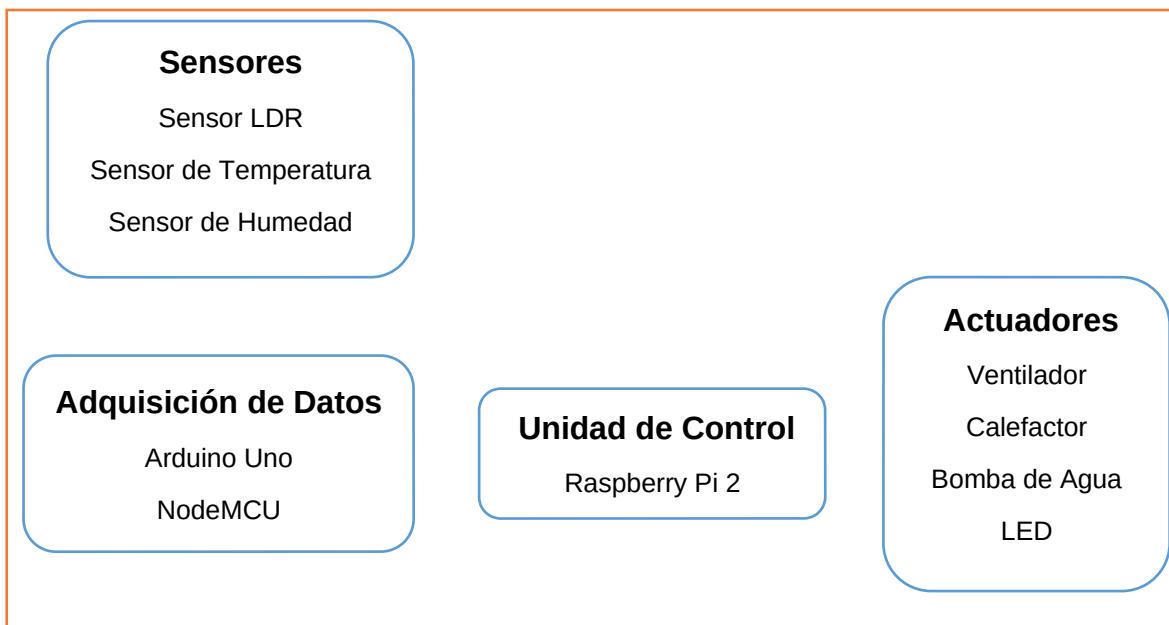


Diagrama 3.13 Subsistemas de Hardware del Sistema

Fuente: Elaboración Propia, 2017

3.8.2 Arquitectura del Software

El sistema dispone de una unidad de adquisición de datos que envía la información obtenida de los sensores a la unidad central de control, que realiza el control de los actuadores del sistema, además de almacenar los datos adquiridos de la unidad de adquisición de datos.

El proyecto utiliza una plataforma Web que contiene los diferentes módulos.

Módulo de Gestión de Usuarios. Se tendrán los siguientes roles:

- **Administrador:** Podrá crear, dar de baja y asignar roles a otros usuarios del sistema.
- **Cliente:** Tendrá acceso a la información y datos del sistema, además de poder controlar los dispositivos de forma remota.

Módulo Web de Monitoreo. Se llevará a cabo el monitoreo de variables ambientales como temperatura, humedad relativa e iluminación desde una página web accesible desde cualquier lugar con conexión a internet.

Módulo Web de Control. Se llevará a cabo el control de los actuadores como motor, ventilador, calefactor y luces led a través de una página web accesible desde cualquier lugar con conexión a internet. Además, se visualizará el estado de los actuadores (motor, ventilador, calefactor y luces LED).

Módulo de reportes. Se generará reporte de parámetros ambientales (temperatura, humedad relativa e iluminación).

3.8.3 Arquitectura Modelo Vista Controlador MVC

En el diagrama 3.8 se muestra el diseño arquitectónico de tres capas (MVC), que se utilizó para la elaboración del proyecto.

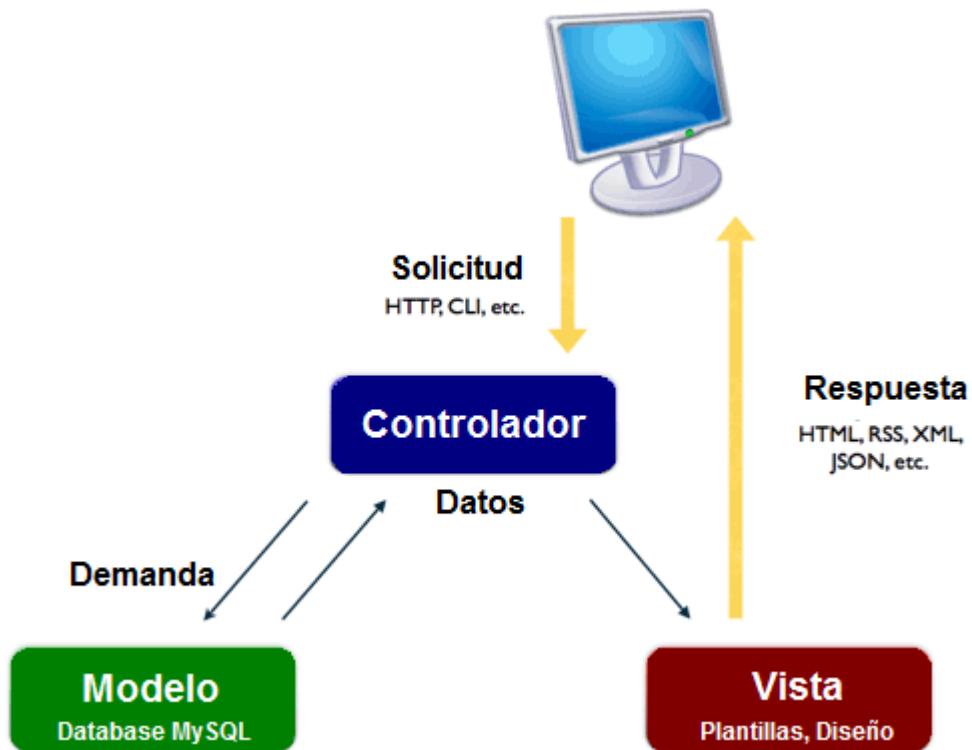


Diagrama 3.14 Arquitectura MVC

Fuente: Elaboración propia

CAPÍTULO IV

ANÁLISIS DE COSTOS

4 ANÁLISIS DE COSTOS

Para el análisis de costos se consideraron los siguientes recursos:

4.1 Recursos Humanos

El desarrollo del sistema fue realizado por los estudiantes: Albarado Sánchez Mirna y Encalada Camargo José Cristian.

Tabla 4.31 Costo Recursos Humanos

Item	Cantidad Horas	Valor Unitario [Bs]	Valor Total [Bs]
Análisis y Diseño	50	40	2000
Programación	100	40	4000
Pruebas e Implementación	30	40	1200
Total			7200

Fuente: Elaboración Propia, 2017

4.2 Recursos Tecnológicos

Para el desarrollo del sistema, se utilizaron recursos de Hardware como de Software detallados a continuación:

4.2.1 Recursos de Hardware

Para el desarrollo del módulo de control y almacenamiento de datos.

- Placa Raspberry pi como servidor para la plataforma web del proyecto.
- Placa Arduino Uno como dispositivo de adquisición de datos.
- Placa NodeMCU como dispositivo de envío de datos en tiempo real.
- Sensores y Actuadores para el control y monitoreo.

En la tabla 4.2 se mencionan los costos de los recursos de hardware utilizados para el desarrollo de este proyecto.

Tabla 4.32 Costo Recursos de Hardware

Item	Cantidad	Valor Unitario [Bs]	Valor Total [Bs]
Raspberry Pi 2 Modelo B	1	450	450
Arduino Uno Rev3	1	60	60
NodeMCU	1	70	70
Sensor DHT11	3	20	60
Sensor LDR	3	5	15
Calefactor	1	100	100
Ventilador	1	150	150
Bomba Hidráulica	1	350	350
Total			1255

Fuente: Elaboración Propia, 2017

4.2.2 Recursos de Software

Para el desarrollo de software se opta por:

- Apache como servidor web HTTP de código abierto.
- MySQL como motor de base de datos.
- PHP/HTML 5 como lenguaje de programación para la plataforma web.
- Phyton para la programación de sensores para la obtención de datos.
- Sublime Text 3 como editor de texto.
- ArgoUml para el Modelamiento del sistema.
- Edraw Max 8.4 para la elaboración de Diagramas.

En la tabla 4.3 se mencionan los costos de los recursos de software utilizados para el desarrollo de este proyecto.

Tabla 4.33 Recursos de Software

Item	Cantidad	Valor Unitario [Bs]	Valor Total [Bs]
Modelamiento (ArgoUml)	1	Software Libre	0
Elaboración de Diagramas	1	EDraw Max 8.4	0
EDraw Max 8.4			
Editor de Texto	1	Software Libre	0
Sublime Text 3			
Programación:	1	Software Libre	0
Lenguaje Html 5			
Programación:	1	Software Libre	0
Lenguaje PHP			
Programación:	1	Software Libre	0
Lenguaje Python			
Motor de Base de Datos:	1	Software Libre	0
MySql			
IDE Arduino	1	Software Libre	0
Total			0

Fuente: Elaboración Propia, 2017

4.3 Material de Apoyo

Para el desarrollo del proyecto se investigaron diferentes áreas como automatización, metodologías de cultivo hidropónico y dispositivos electrónicos para el monitoreo y control.

Para el desarrollo de la parte de control se necesitaron conocimientos básicos en electrónica y automatización.

5 ANÁLISIS DE COSTOS PROYECTADOS A CINCO AÑOS

Para el estudio de costos se consideró un horizonte de evaluación de 5 años, con 4 ciclos de producción cada una.

Durante cada ciclo de producción se cultivara 10.000 lechugas de tipo manteca y 10.025 lechugas tipo Francesa, un total de 20.025uds/ciclo, haciendo un total de 80.100 lechugas anuales, (En un espacio de 20m²). Utilizando el software.

5.1 Inversión Activo Fijo

CONCEPTO	MONTO[Bs]	VIDA UTIL[años]	DEPRECIACIÓN[Bs/año]
Invernadero	5000	5	1000,00
Raspberry Pi	450	5	90,00
Arduino Uno Rev 3	60	5	12,00
Mode MCU	70	5	14,00
Calefactores	300	5	60,00
Ventiladores	450	5	90,00
Bomba de agua 1ph	1100	8	137,50
Bombillas LED(10uds)	180	17	10,59
Tanque Colector	600	5	120,00
Tubos PVC 3"(833ud de 4m)	41650	8	5206,25
Canastillos(20000uds)	8000	5	1600,00
Bandejas Almacigueras(133uds)	1300	5	260,00
TOTAL	59160		8600,34

5.2 Inversión Activo Diferido

CONCEPTO	MONTO[Bs]
Software Hidropónico	7200

5.2.1 Amortización Activo Diferido

Amortización Activo Diferido[Bs/años]
AAD=Inv.AD/5=1440

5.3 Inversión Total

INVERSIÓN TOTAL=INVERSIÓN ACTIVO FIJO +INVERSIÓN ACTIVO DIFERIDO + CAPITAL DE TRABAJO	
INVERSIÓN TOTAL[Bs]	
	663
	60

5.4 Costos Operativos

5.4.1 Costo de Administración

CONCEPTO	MONTO[Bs/año]
Insumos de limpieza	80
Licencia Sanitario	300
Servicio telefónico	320
Imprevistos	6000
TOTAL	6700

5.4.2 Costo de Producción

CONCEPTO	MONTO[Bs/ciclo de producción]
Semillas de lechuga	900
Kit hidropónico	130
Agua	50
Mano de Obra	2000
Energía Eléctrica	88

TOTAL cantidad de ciclos de producción al año	3168 4
TOTAL COSTO DE PRODUCION [Bs/Año]	12672
Depreciación	8600,34
COSTO PRODUCCION	21272,34

5.4.3 Costo de Comercialización

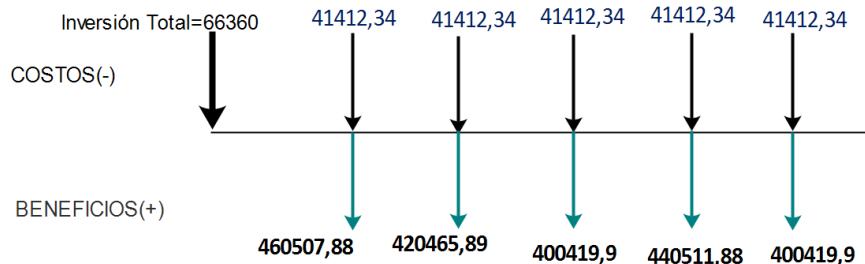
CONCEPTO	MONTO[Bs/año]
Envase	8000
Trasporte	4000
TOTAL	12000

5.5 Estado de Resultados Proyectados para cinco años

CONCEPTO	AÑO 1	AÑO 2	AÑO 3	AÑO 4	AÑO 5
Cantidad de lechuga Manteca	40000	40000	40000	40000	40000
Perdidas [0,02%]	8	8	8	8	8
Cantidad de lechuga Francesa	40100	40100	40100	40100	40100
Perdidas [0,02%]	8,02	8,02	8,02	8,02	8,02
CANTIDAD TOTAL LECHUGAS Manteca	39992	39992	39992	39992	39992
CANTIDAD TOTAL LECHUGAS Francesa	40091,98	40091,98	40091,98	40091,98	40091,98
Precio de Venta lechuga Manteca	5,5	5	5	5	5
Precio de Venta lechuga Francesa	6	5,5	5	6	5
Ingreso Bruto x lechuga Manteca	219956	199960	199960	199960	199960
Ingreso Bruto x lechuga Francesa	240551,88	220505,89	200459,9	240551,88	200459,9

INGRESO NETO	46050	42046	40041	44051	400419
TOTAL	7,88	5,89	9,9	1,88	,9
COSTOS OPERATIVOS	41412,3 4	41412,3 4	41412,3 4	41412,3 4	41412,3 4
Costo de Producción	21272,3 4	21272,3 4	21272,3 4	21272,3 4	21272,3 4
Costo de Administración	8140	8140	8140	8140	8140
Costo de Comercialización	12000	12000	12000	12000	12000
Utilidad Bruta	419095, 54	379053, 55	359007, 56	399099, 54	359007, 56
Utilidad Neta	419095, 54	379053, 55	359007, 56	399099, 54	359007, 56

5.5.1 Esquema para cálculo de Rentabilidad



VAN: Valor Actual de los beneficios netos que genera el proyecto.

Calculo del VAN: $VAN = I_0 - \sum F_t / (1+r)^t$

$$VAN = -66360 + 419095.54/(1+0.10)^1 + 379053.55/(1+0.10)^2 + 359007.56/(1+0.10)^3 + 399099.54/(1+0.10)^4 + 359007.56/(1+0.10)^5$$

VAN=1.459.430 Bs

VAN>0 El proyecto es rentable.

Conclusión: Realizar el proyecto es mucho más rentable que colocar el dinero en una actividad financiera.

Tasa interna de retorno

TIR=622%>Tasa de interés 10%

Conclusión: El proyecto es rentable ya que Tasa Interna de Retorno es superior al costo de oportunidad del capital, es decir superior al mínimo aceptable.

Beneficio costo

$$R_{B/C} = 1915263,75 / 207061,7 = 9,25$$

B/C>1 Aceptable el proyecto

Conclusión: Como la relación Costo -Beneficio es mayor que 1, podemos afirmar que el proyecto será rentable en los próximos 5 años.

CAPÍTULO V

CONCLUSIONES Y RECOMENDACIONES

6 CONCLUSIONES Y RECOMENDACIONES

6.1 CONCLUSIONES

- ✓ Para la selección de componentes electrónicos (hardware), se eligió trabajar con las placas Raspberry Pi 2, Arduino UNO y NodeMCU debido a que los costes de PLCs son muy elevados para trabajar con esta tecnología.
- ✓ Se construyó el prototipo del Invernadero hidropónico, con una capacidad para un cultivo de 41 lechugas, ocupando un espacio de 1.3 [m] de alto por 0.93 [m] de largo y 0.6 [m] de profundidad.
- ✓ Se desarrolló el módulo de gestión de Usuarios, para el registro de nuevos usuarios, así como la validación, autenticación y manejo de dos roles diferentes de usuario: administrador y usuario normal.
- ✓ Se desarrolló el módulo Web de Monitoreo, que permite visualizar las variaciones de temperatura, humedad relativa y luz en tiempo real.
- ✓ Se desarrolló el módulo Web de Control, que permite el control individual de los actuadores (bomba de agua, ventilador, luces Led y calefactor) y activación rápida de los mismos, a través de la aplicación web desde cualquier lugar con acceso a internet.
- ✓ Se desarrolló el módulo Web de Reportes, desde el cual el usuario obtiene reportes de variación de temperatura, humedad e iluminación del ambiente, en formatos .jpeg, .png o .pdf listos para la impresión.

6.2 RECOMENDACIONES

Las recomendaciones para el uso correcto y adecuado del sistema son:

6.2.1 Recomendaciones con el Hardware

- ✓ Debido al entorno de trabajo dentro de un invernadero, se recomienda instalar la Raspberry en un lugar fuera del alcance del agua y mantener la Raspberry con su carcasa protectora.
- ✓ Mantener buena ventilación del sitio donde se instalará la Raspberry tarjeta para evitar el sobrecalentamiento y daño del equipo.
- ✓ Mantener una conexión activa de internet ya sea cableada o inalámbrica en las instalaciones del invernadero o donde instalará la Raspberry.
- ✓ El sistema de calefacción y enfriamiento, así como el motor y la iluminación LED deben estar conectado a una fuente eléctrica que proporcione un voltaje de 220 voltios.

6.2.2 Recomendaciones a Futuro

Las posibles evoluciones y mejoras que se pueden dar a futuro son las siguientes:

- ✓ En futuros proyectos se podrían realizar mejoras al sistema como la implementación de una cámara para verificar el funcionamiento de los dispositivos.
- ✓ Se podrían utilizar dispositivos más robustos como PLCs para mayor confiabilidad del hardware del sistema, resaltando que el costo se elevaría en gran medida.

7 BIBLIOGRAFIA

Arduino, C. (10 de 08 de 2017). *Arduino*. Obtenido de Arduino: <https://www.arduino.cc>

Carrasco, G. (1996). LA TECNICA DE LA SOLUCION NUTRITIVA RECIRCULANTE. En Gilda Carrasco, *LA EMPRESA HIDROPONICA DE MEDICA ESCALA*. Chile.

Favela, E. (2006). MANUAL PARA LA PREPARACIÓN DE SOLUCIONES NUTRITIVAS. En E. Favela, *MANUAL PARA LA PREPARACIÓN DE SOLUCIONES NUTRITIVAS*. Mexico.

Istruments, N. (10 de 08 de 2017). *Natinal Instruments: Sistemas de Pruebas, Medidas y Embebidos*. Obtenido de Natinal Instruments: Sistemas de Pruebas, Medidas y Embebidos: <http://www.ni.com/data-acquisition/what-is/esa/>

J. C. Gilsanz. (2007). Uruguay: Unidad de Comunicacion y transferencia en tecnología. En J. C. Gilsanz, *HIDROPONIA*. Montevideo.

Jacobson, S. (2002). The Rational Objectory Process - A UML-based Software Engineering Process. En S. Jacobson, *The Rational Objectory Process - A UML-based Software Engineering Process*.

Kimmel, P. (2013). Manual de UML. En P. Kimmel, *Manual de UML* (pág. 8).

- Laudon, L. &. (2012). Sistemas de informacion gerencial. En L. &. Laudon, *Sistemas de informacion gerencial*. Mexico.
- Luján, S. (2001). Programación en Internet: Clientes Web. En S. Luján, *Programación en Internet: Clientes Web*. Editorial Club Universitario.
- Molina, U. A. (2012). CURSO PRACTICO INTERNACIONAL DE HIDROPONIA. En U. A. Molina, *CURSO PRACTICO INTERNACIONAL DE HIDROPONIA*.
- Rodriguez, A. (2001). Hidroponía: Prespectivas y Futuro. En A. Rodríguez, *MANUAL PRACTICO DE HIDROPONIA*.
- Silberschatz, K. y. (2002). Database System Concepts. En K. y. Silberschatz, *Database System Concepts* (pág. 25).
- Agricolas.com. (05 de 12 de 2016). Obtenido de <http://www.novedades-agricolas.com/es/venta-invernaderos-novedades/tipos-de-invernaderos>
- Agropinos.com. (05 de 02 de 2017). Obtenido de <http://www.agropinos.com/invernaderos-hidroponicos>
- Highcharts. (22 de 03 de 2017). Obtenido de <https://en.wikipedia.org/wiki/Highcharts>
- Pressman, R. (2001). Ingenieria de software: Un enfoque práctico. New York: Mc Graw Hill.
- Scrum en la Metodología Ágil. (20 de 03 de 2017). Obtenido de <http://www.i2btech.com/blog-i2b/tech-deployment/para-que-sirve-el-scrum-en-la-metogologia-agil/>

ANEXOS

ANEXO A

INTERFACES DEL SISTEMA

Interfaz inicio sesión

The screenshot shows a login form titled "Iniciar sesión". It has two input fields: "Nombre de Usuario" containing "mirna" and "Contraseña" containing "*****". Below the inputs is a blue button labeled "Iniciar Sesión". At the bottom right of the form is a link "Registrar una cuenta".

Fuente: Elaboración Propia, 2017

Interfaz de Registro de Usuarios

The screenshot shows a registration form titled "Registrar nueva cuenta". It includes fields for "Nombre" (with placeholder "Introduzca su nombre"), "Apellido Paterno" (with placeholder "Introduzca su apellido paterno"), "Apellido Materno" (with placeholder "Introduzca su apellido materno"), "Dirección de correo electrónico" (with placeholder "Introduzca su correo electrónico"), "Usuario" (with placeholder "nombreUsuario") and "Contraseña" (with placeholder "Contraseña"), and "Confirmar contraseña" (with placeholder "Confirmar Contraseña"). At the bottom left is a blue button labeled "Registrarse" and at the bottom right is a link "Página de inicio de sesión".

Fuente: Elaboración Propia, 2017

Interfaz Principal Administrador

The screenshot shows the administrator's main interface. On the left is a sidebar with the following menu items:

- Bienvenido
- Principal
- Nuevo Usuario/Administrador
- Monitoreo
- Datos históricos
- Control >
- Reportes >
- Acerca de...

On the right, the main content area has a header bar with "Página Principal / Gestión de Usuarios" and a "Salir" button. Below this is a sub-header "Lista de Usuarios". There are filters for "Mostrar 10 registros" and a search bar with placeholder "Introduzca los datos". A table lists six user entries:

ID Usuario	Nombre	1er Apellido	2do Apellido	Email	Usuario	Contraseña	Rol
1					admin	admin	1
2					cris	cris	2
3	Cristian	Encalada	Camargo	cris@gmail.com	criss	criss	2
4	Cristian	Encalada	Camargo	cris@gmail.com	cristian	12345	2
5	pavel	caceres	torrez	pavel@gmail.com	pavel	pavel	2
6	Crístian	Encalada	Camargo	cris@gmail.com	cris	cris	2

Fuente: Elaboración Propia, 2017

Interfaz Principal Usuario

BIENVENIDO AL SISTEMA DE MONITOREO Y CONTROL DE CULTIVOS ✖ Salir

Página Principal / Bienvenido

Seleccione una opción del Menú:

- Principal
- Monitoreo
 - Datos históricos
 - Control
 - Reportes
 - Contactos

Monitoreo
Realiza el monitoreo en tiempo real de las variables ambientales



Datos Históricos
Muestra en una tabla las variables ambientales almacenadas



Control
Realiza el control de dispositivos del cultivo
Control Individual: Permite el control individual de cada dispositivo
Control Rápido: Permite el control de varios dispositivos a la vez



Reportes
Permite la descarga de reportes ambientales como ser:
Reporte de Temperatura
Reporte de Humedad
Reporte de Luz



Sistemas Informáticos © Incos Nocturno 2017

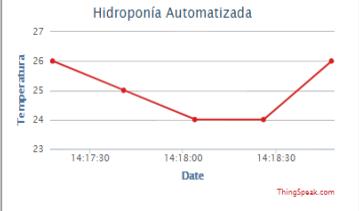
Fuente: Elaboración Propia, 2017

Interfaz de Monitoreo de variables Ambientales

BIENVENIDO AL SISTEMA DE MONITOREO Y CONTROL DE CULTIVOS ✖ Salir

Monitoreo

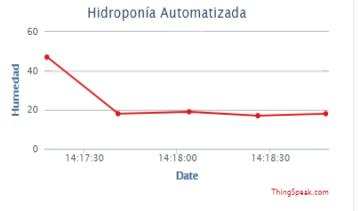
Temperatura
Hidroponía Automatizada



Date	Temperatura
14:17:30	26
14:18:00	25
14:18:30	24
14:19:00	24
14:19:30	26

Temperatura actual del ambiente

Humedad Relativa
Hidroponía Automatizada



Date	Humedad
14:17:30	55
14:18:00	20
14:18:30	20
14:19:00	18
14:19:30	18

Humedad relativa actual del ambiente

Iluminación
Hidroponía Automatizada



Date	Luminosidad
14:17:30	750
14:18:00	400
14:18:30	750
14:19:00	750
14:19:30	750

Fuente: Elaboración Propia, 2017

Interfaz de Control Rápido de Dispositivos

BIENVENIDO AL SISTEMA DE MONITOREO Y CONTROL DE CULTIVOS

Página Principal / Control Rápido

Control Rápido

Control Rápido Activado

Estado Actual Modo Automático

Sistemas Informáticos © Incos Nocturno 2017

Estado de los Dispositivos

Dispositivo	Estado
Calefactor	Apagado
Ventilador	Encendido
Motor	Encendido
Iluminación	Encendido

Fuente: Elaboración Propia, 2017

Interfaz de Control manual de Dispositivos

BIENVENIDO A CONTROL MANUAL DE LOS ACTUADORES

Página Principal / Control Manual

Control Manual

Motor

Iluminación

Ventilación

Calefacción

Estado de los Dispositivos

Dispositivo	Estado
Calefactor	Apagado
Ventilador	Apagado
Motor	Apagado

Fuente: Elaboración Propia, 2017

Interfaz de Reporte de Temperatura



Fuente: Elaboración Propia, 2017

Interfaz de Reporte de Iluminación



Fuente: Elaboración Propia, 2017

Interfaz de Reporte de Humedad Relativa



Fuente: Elaboración Propia, 2017

ANEXO B

ACTAS DE REUNIÓN DE SPRINTS

Acta Sprint 1



Acta Sprint 1

Cochabamba, 05 de junio de 2017

Ref: Acta de reunión

Asistentes:

Sebastián Alcalá – *Product Owner*

Mirna Albarado / Cristian Encalada – *Equipo de Desarrollo*

Ing. Pavel Cáceres Torrez – *Scrum Master*

En la reunión se llega al acuerdo que los siguientes elementos deberán ser terminados entre el 5 de junio y 20 de junio 2017, cada día se hará una pequeña reunión de 10 minutos para revisar los problemas en el avance.

- ✓ Definición de actores del sistema
- ✓ Elaboración de Diagramas de Casos de Uso
- ✓ Refinamiento de Casos de Uso
- ✓ Realizar el Diccionario de Casos de Uso
- ✓ Elaboración de Diagramas de Clases para el Diseño
- ✓ Definición de los requerimientos funcionales
- ✓ Definición de los requerimientos no funcionales
- ✓ Elaboración del Diagrama de Capas
- ✓ Elaboración del Diagrama de Despliegue

Teléfonos: 72205308-69495967
Dirección: Av. Ayacucho No. 0415 esquina Jordán
Cba - Bolivia

Fuente: Elaboración Propia, 2017

Acta Sprint 2



Acta Sprint 2

Cochabamba, 21 de junio de 2017

Ref: Acta de reunión

Asistentes:

Sebastián Alcalá – *Product Owner*

Mirna Albarado / Cristian Encalada – *Equipo de Desarrollo*

Ing. Pavel Cáceres Torrez – *Scrum Master*

En la reunión se llega al acuerdo que los siguientes elementos deberán ser terminados entre el 21 de junio y 06 de julio 2017, cada día se hará una pequeña reunión de 10 minutos para revisar los problemas en el avance.

- ✓ Selección de Herramientas para el Desarrollo de Software
- ✓ Selección de Dispositivo de Control y Almacenamiento de Datos
- ✓ Selección de Dispositivo de Adquisición de Datos
- ✓ Selección de Sensores
- ✓ Selección de Actuadores
- ✓ Construcción el prototipo hidropónico

Teléfonos: 72205308-69495967
Dirección: Av. Ayacucho No. 0415 esquina Jordán
Cba – Bolivia

Fuente: Elaboración Propia, 2017

Acta Sprint 3



Acta Sprint 3

Cochabamba, 07 de julio de 2017

Ref: Acta de reunión

Asistentes:

Sebastián Alcalá – *Product Owner*

Mirna Albarado / Cristian Encalada – *Equipo de Desarrollo*

Ing. Pavel Cáceres Torrez – *Scrum Master*

En la reunión se llega al acuerdo que los siguientes elementos deberán ser terminados entre el 07 de julio y 24 de julio 2017, cada día se hará una pequeña reunión de 10 minutos para revisar los problemas en el avance.

- ✓ Definir los roles de usuario del sistema
- ✓ Creación de la Interfaz de Usuario
- ✓ Creación de la Interfaz de Administrador
- ✓ Elaborar la Autenticación de Usuarios
- ✓ Realizar la Validación de Usuarios
- ✓ Realizar el Registro de Usuarios
- ✓ Creación de la interfaz de monitoreo en la aplicación Web

Teléfonos: 72205308-69495967
Dirección: Av. Ayacucho No. 0415 esquina Jordán
Cbba – Bolivia

Fuente: Elaboración Propia, 2017

Acta Sprint 4



Acta Sprint 4

Cochabamba, 25 de julio de 2017

Ref: Acta de reunión

Asistentes:

Sebastián Alcalá – *Product Owner*

Mirna Albarado / Cristian Encalada – *Equipo de Desarrollo*

Ing. Pavel Cáceres Torrez – *Scrum Master*

En la reunión se llega al acuerdo que los siguientes elementos deberán ser terminados entre el 25 de julio y 08 de agosto 2017, cada día se hará una pequeña reunión de 10 minutos para revisar los problemas en el avance.

- ✓ Instalación del SO Raspbian en la Raspberry Pi 2
- ✓ Creación de la Interfaz de Control
- ✓ Programación del Arduino Uno.
- ✓ Creación de la interfaz de Reportes.
- ✓ Realizar el análisis de Costos.
- ✓ Realizar la Documentación del proyecto.

Teléfonos: 72205308-69495967
Dirección: Av. Ayacucho No. 0415 esquina Jordán
Cbba – Bolivia

Fuente: Elaboración Propia, 2017