

ANEP

E.T.S Maldonado

NOMBRE DEL PROYECTO

Software de Gestión de un Vivero

ASIGNATURAS

Análisis y Diseño de Aplicaciones

Formación Empresarial

Programación III

Proyecto

Sistemas de Bases de Datos II

Sistemas Operativos III

INTEGRANTES DEL EQUIPO

**Jordan Scalabrini, Martina González, Germán Giménez y
Jun Yamaki**

AÑO

2019

Índice

1	Empresa	4
1.1	Datos	4
1.1.1	Nombre	4
1.1.2	Logo	4
1.1.3	Fundamentación	4
1.1.4	Forma Jurídica	4
1.1.5	Localización	4
1.2	Misión	5
1.3	Visión	5
1.4	Objetivos	5
1.5	Valores	6
2	Anteproyecto	7
2.1	Introducción	7
2.2	Presentación del cliente	7
2.2.1	Descripción proceso Plantar	9
2.2.2	Descripción proceso Venta	9
2.2.3	Descripción proceso Fumigar (Prevención)	9
2.2.4	Descripción del proceso Trasplantación	10
2.2.5	Descripción del proceso Regado	10
2.2.6	Descripción del proceso Poda	10
2.2.7	Diagrama del Proceso Plantar	11
2.2.8	Diagrama del proceso Venta	12
2.2.9	Diagrama del proceso Fumigar (Prevención)	13
2.2.10	Diagrama del proceso Trasplantación	14
2.2.11	Diagrama del proceso Riego	16
2.2.12	Diagrama del proceso Poda	17
2.3	Actores involucrados	18
2.3.1	Diagrama	18
2.3.2	Descripción	18
2.4	Descripción del entorno	19
2.5	Lista de necesidades del cliente	20
2.5.1	Conjunto de necesidades de la empresa del cliente	20
2.6	Objetivos	20
2.6.1	Actores del Sistema	20
2.7	Requerimientos funcionales	21
2.7.1	Alta de cliente	21
2.7.2	Baja de cliente	21
2.7.3	Modificación de cliente	21
2.7.4	Listado de cliente	22
2.7.5	Alta de proveedor	22
2.7.6	Baja de proveedor	22
2.7.7	Modificación de proveedor	22
2.7.8	Listado de proveedor	22
2.7.9	Alta de empleado	23
2.7.10	Baja de empleado	23
2.7.11	Modificación de empleado	23
2.7.12	Listado de empleado	23
2.7.13	Alta de planta	23

2.7.14	Baja de planta.....	24
2.7.15	Modificación de planta.....	24
2.7.16	Listado de planta	24
2.7.17	Listado de enfermedades de la planta y medidas de prevención	24
2.8	Requerimientos no Funcionales	24
2.9	Alcances y limitaciones	24
2.10	Informe sobre características fundamentales del lenguaje VB.NET	25
2.10.1	Características de VB.NET	25
2.10.2	Instrucciones en VB .NET.....	25
2.10.3	Procedimientos en VB .NET	26
2.10.4	Llamada al procedimiento.....	26
2.10.5	Tipos de procedimientos.....	27
2.10.6	Tipo de datos	27
2.10.7	Operadores y expresiones en VB .NET	29
2.10.8	Objetos y clases	29
2.10.9	Interfaces.....	30
2.10.10	Flujo de Control en VB .NET	30
2.11	Informe sobre Sistema Gestor de Base de Datos elegido.....	31
2.12	Investigación de Sensores	37
2.12.1	Investigación de Planta de Trabajo y Equipamiento	39
2.12.2	Justificación técnica de la elección del SO de los Servidores	40
2.12.3	Licenciamiento.....	41
2.12.4	Análisis de los Sistemas Operativos a Seleccionar en Terminales y Servidores	42
2.12.5	Estudio de Alternativas	49
2.12.6	Esquemas de Arquitectura	50
2.12.7	Diseño de Cableado Estructurado	53
2.12.8	Esquema de Ramales	54
2.12.9	Selección de placa de control y alimentación	54
3	Anexos	56
3.1	Equipo	56
3.1.1	Reglamento del equipo	56
3.1.2	Desarrollo de las actividades- Acta de Reunión.....	57
3.1.3	Análisis de riesgos	58
3.1.4	Bitácora Actualizada	61

1 Empresa

1.1 Datos

1.1.1 Nombre

El nombre de la empresa se establece como “SoftCarry”.

1.1.2 Logo



1.1.3 Fundamentación

La empresa se creó a partir de una investigación al mercado donde se descubrió la necesidad de crear un producto que facilite el cuidado de las plantas.

Motivados con ello se tuvo la idea de crear un software el cual permitiese que cualquier dueño de una planta pueda dejarlas en un vivero que se haga cargo de ellas por un tiempo determinado, al igual de que facilitar el registro y gestión de riego del vivero mismo.

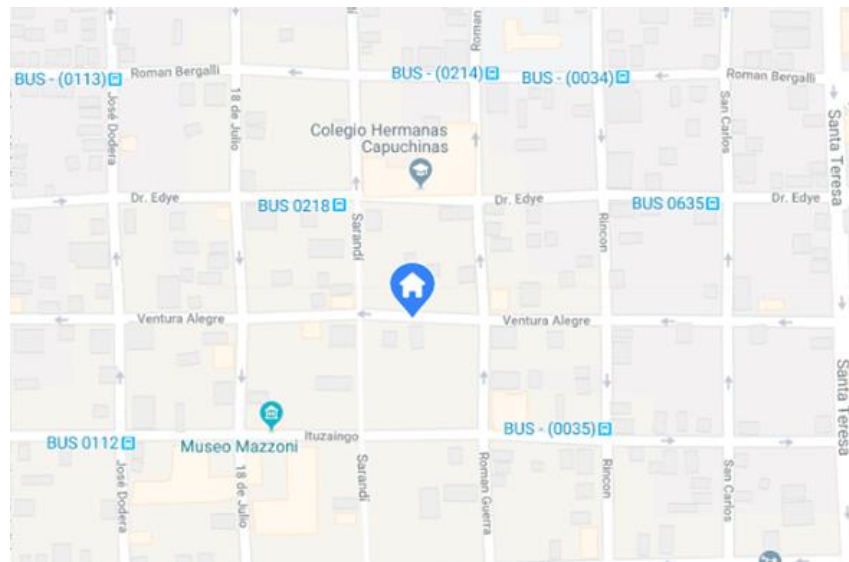
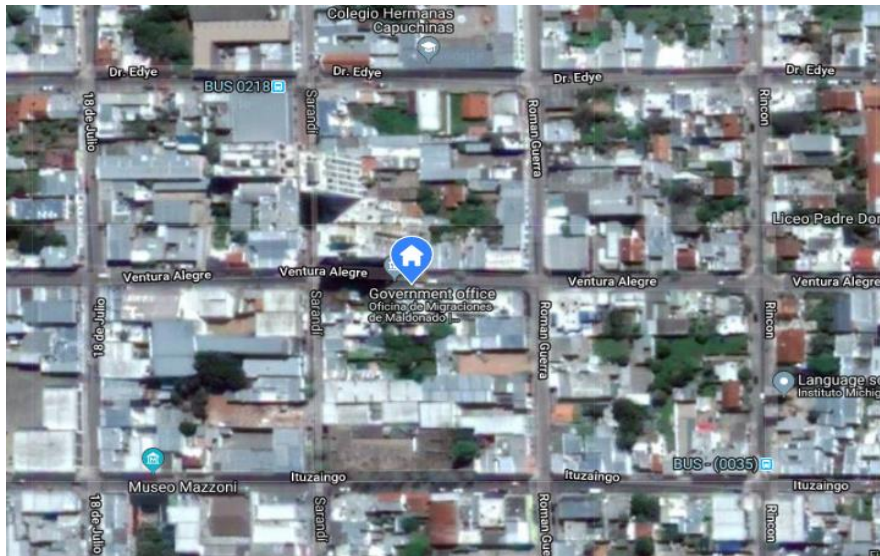
De esta forma, se puede ayudar a la naturaleza con la tecnología, creando así una alianza entre ambas, que se mantuviese como benefactora para ambas partes.

1.1.4 Forma Jurídica

Sociedad Limitada Unipersonal.

1.1.5 Localización

La empresa está ubicada en Sarandí y Ventura Alegre, en el centro de Maldonado y con fácil acceso para cualquier posible cliente.



1.2 Misión

Elaborar software que permita al cliente ofrecer una más amplia variedad de servicios y satisfacer sus necesidades por mantener registros. Todo aquello mientras se compromete a ser: innovadora, competitiva y fuertemente orientada a la satisfacción de los clientes.

1.3 Visión

Lograr la expansión de la empresa con tal de proporcionar software para todo el país.

1.4 Objetivos

Tener una amplia cantidad de clientes para 2024.

Triplicar sus sucursales, en busca de ofrecer mayor servicio.

1.5 Valores

Los valores de la empresa son:

Transparencia. Todas las relaciones, sea entre miembros de la empresa o con los clientes, serán tratadas con honestidad y realidad de lo que ocurre realmente.

Originalidad. Mantenerse único y alejado de lo que se ha visto ya innumerables veces cuando se sea posible.

Calidad. Ser minucioso hasta el último detalle que cerciore que se está entregando un producto de calidad.

Responsabilidad. Responder a lo prometido y cargar con aquello hecho, sea algo bueno o malo.

Autocriticar. Plantearse aquello que se hace bien y en qué se falla, en busca de perfeccionar nuestras acciones y comportamientos para ofrecer el mejor producto.

Competitividad. Siempre mantenerse ideando y creando nuevos proyectos para satisfacer al cliente o aquellos posibles clientes, sin importar el flujo actual de la competencia.

2 Anteproyecto

2.1 Introducción.

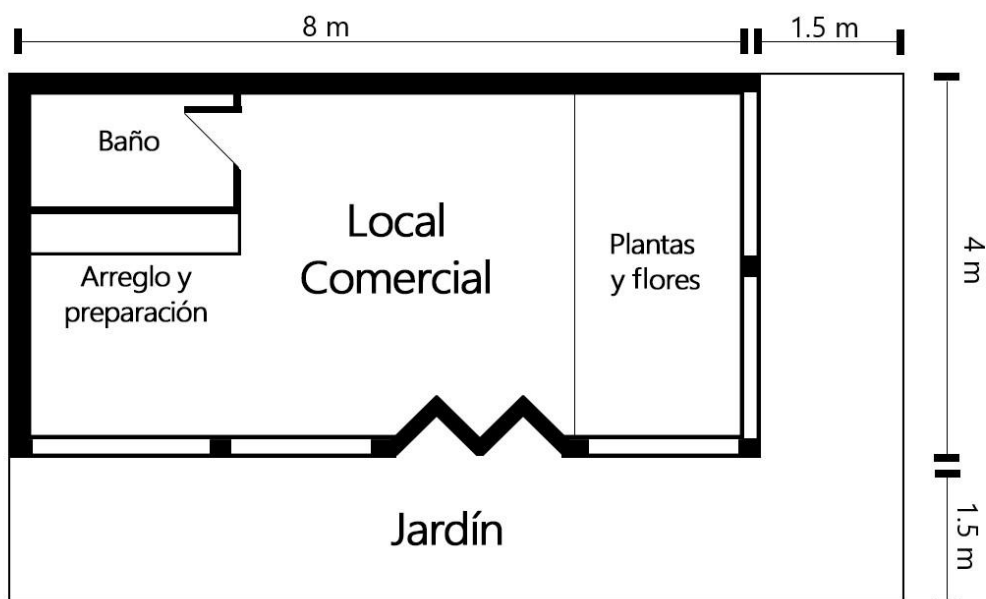
2.2 Presentación del cliente.

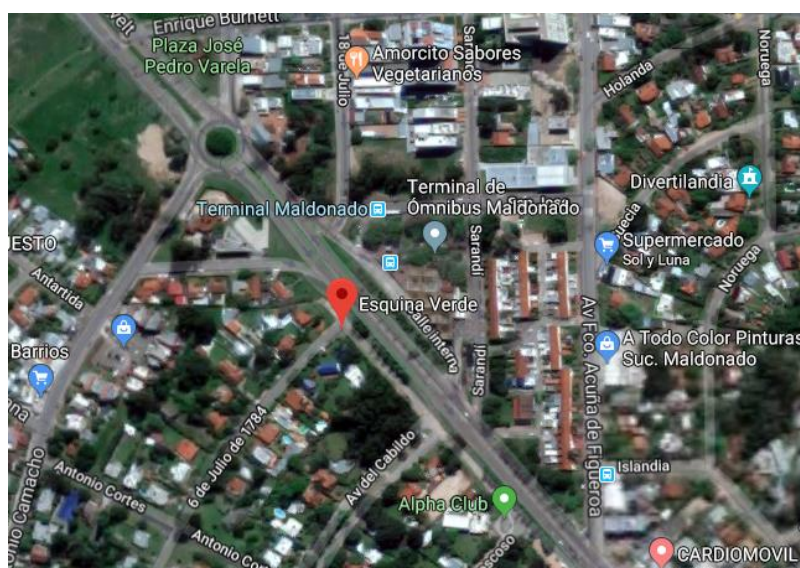
Nuestro cliente es Javier Sivak, dueño de la empresa Esquina Verde, ubicada en la Avenida Franklin Delano Roosevelt y 6 de Julio de 1784, frente a la terminal de Maldonado. Sus horarios son de 09:00 AM a 13:00PM y 15:00PM a 19:00PM de lunes a viernes.

La empresa se fundó en noviembre de 2017 y hasta la actualidad se mantiene como el único empleado, encargándose de los roles de caja, jardinería, exterminación y depósito. No tiene sucursales, y se mantiene en su local sin expansiones ni cambio valorativo en el mismo.

El local mencionado es pequeño, de 8 m x 4 m dentro de alrededor de 52.5m cuadrados totales del terreno, donde mayor parte del espacio está dedicado a las plantas que se mantienen a la intemperie.

De aquí a cinco años, el cliente prevé que





2.2.1 Descripción proceso Plantar

El jardinero busca las semillas.

En caso de que no las tenga, se acerca al dueño y solicita que se haga un pedido para las mismas, quién una vez recibe el mensaje se contacta con su proveedor. Una vez el proveedor recibe dicho mensaje lo enviará en un momento determinado, y el dueño pagará el recibo otorgado por el proveedor, con dinero justo o dinero de más, que hará que le devuelva el cambio. El dueño recibe el pedido y le hace saber al jardinero que llegó.

En caso de tenerlas, o al ya tenerlas, las busca en el cajón donde usualmente se guardan las semillas y las toma, al igual que tomas las herramientas de su lugar usual sobre la mesa del local. Una vez tenga las herramientas y las semillas, sale al jardín y se acerca a la zona donde suele plantar, paralelo a la puerta de entrada al local. Da unos pasos y toma una de las bolsas de tierra que mantiene afuera, estando seguro de que es la que necesita para el tipo de semilla que plantará, recordando cuál y cuánta necesita. Cuando las semillas estén cubiertas por tierra, se levanta y busca la regadera, que llenará en caso de que se encuentre vacía y procederá a volver a donde plantó previamente y regar las semillas, sabiendo cuánta agua necesita.

2.2.2 Descripción proceso Venta

Una vez que el cliente ingresa y saluda, habla con el dueño y hace el pedido de la planta que desea comprar, a lo cual el dueño busca la planta para asegurarse de tenerla. Al encontrarla se la muestra al cliente, el cual pregunta por el costo de la misma a lo que el dueño responde y le dirige al cajero. El cliente puede pagar con el monto justo o no, en el último caso le devuelve el cambio y hace registro de la venta. Tras pagar, el cliente recibe el producto y se despide.

En cambio, si el dueño no encuentra la planta, se lo informa al cliente, el cual vuelve a preguntar por otra planta o se retira.

2.2.3 Descripción proceso Fumigar (Prevención)

El jardinero se fija en la fecha en la que se fumigó por última vez, si esta fue hace más de un mes, entonces procederá a hacerlo. Busca el rociador, el cual está en el local paralelo a la mesa, y si este no está cargado procede a desmontarlo y buscar el pesticida, el específico para esta tarea y lo coloca en el tanque según la medida que marca el envoltorio, entonces lleva el tanque hasta una llave de agua y lo disuelve en el agua necesaria. Una vez tenga el tanque cargado, lo vuelve a colocar en el rociador.

Una vez el rociador esté cargado, lo lleva hasta las plantas, rociando las mismas hasta que estén cubiertas por la sustancia. Cuando todas las plantas hayan sido rociadas, se devuelve el rociador hasta su lugar. El jardinero hace un recordatorio de la fecha en la que fumigó.

2.2.4 Descripción del proceso Trasplantación

El jardinero verifica que las plantas se hayan plantado hace un año o más, y si las plantas parecen necesitar una maceta más grande, procede a trasplantarlas. Si no es fecha de trasplante o las plantas no crecieron demasiado, entonces no se trasplanta.

Cuando va a trasplantar, toma sus herramientas del lugar usual y vuelve al jardín, donde desentierra las plantas que trasplantará, y luego las lleva al nuevo lugar donde las plantará. También toma un balde con preparado de tierra para la planta específica, lo cual sabe de memoria, y las lleva al mismo lugar donde están las plantas. Allí comienza a hacer los agujeros donde después serán colocadas las plantas, cubriendo las raíces de estas con la tierra. Entonces tomará el regador, el cual llenará de agua en caso de que no tenga ninguna, y regará las plantas. Una vez terminado, guarda las herramientas utilizadas y anota la fecha en la que hizo trasplante.

2.2.5 Descripción del proceso Regado

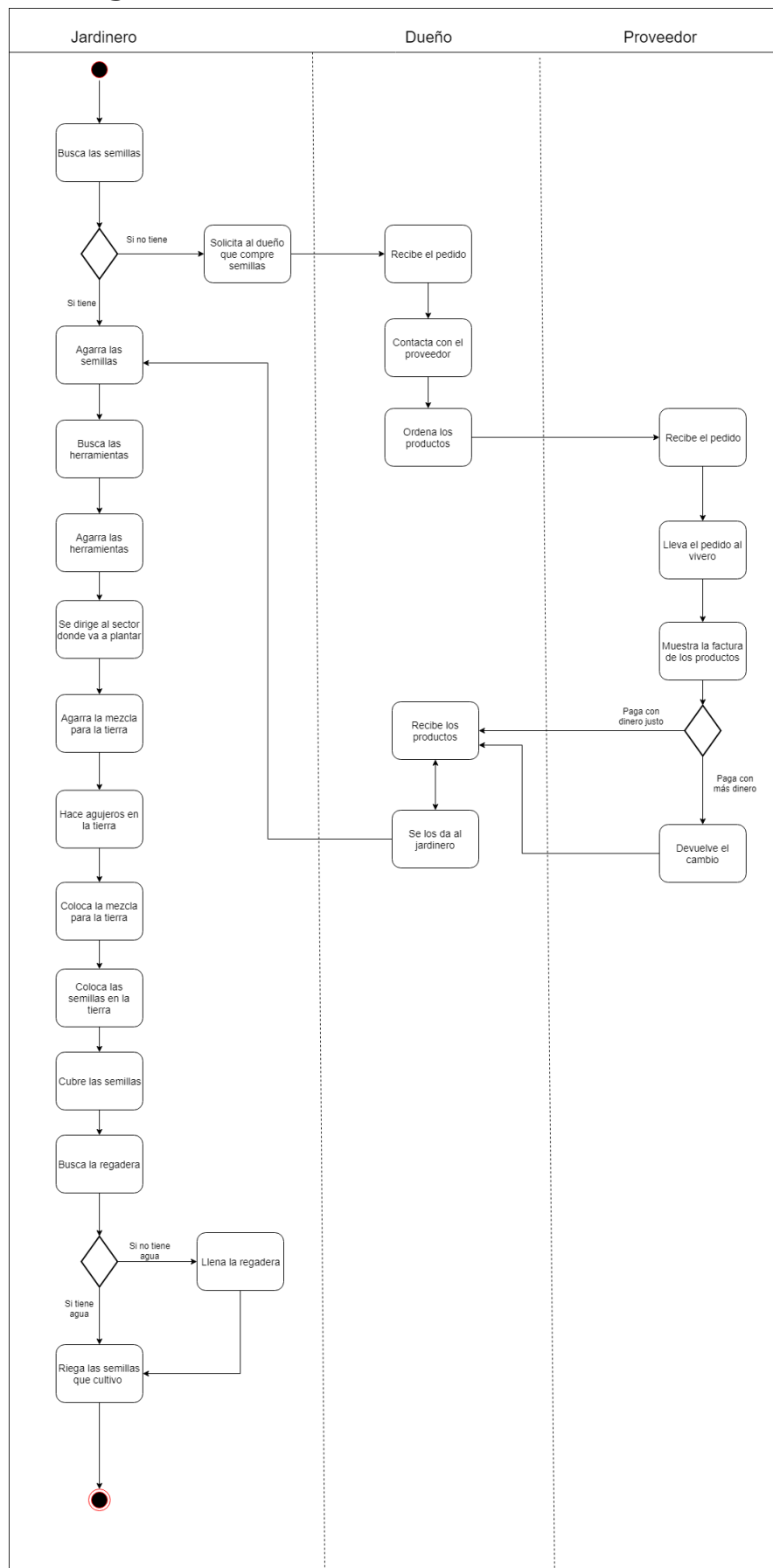
El jardinero se asegura de que sea el día de regado, no es día de regado examina las plantas, si a pesar de que no sea fecha, pero las plantas se encuentran secas, riega de todas formas.

Busca la regadera, la cual llena de agua y riega todas aquellas plantas que necesiten riego ese día. Guarda la regadera y anota la fecha en la que hizo riego.

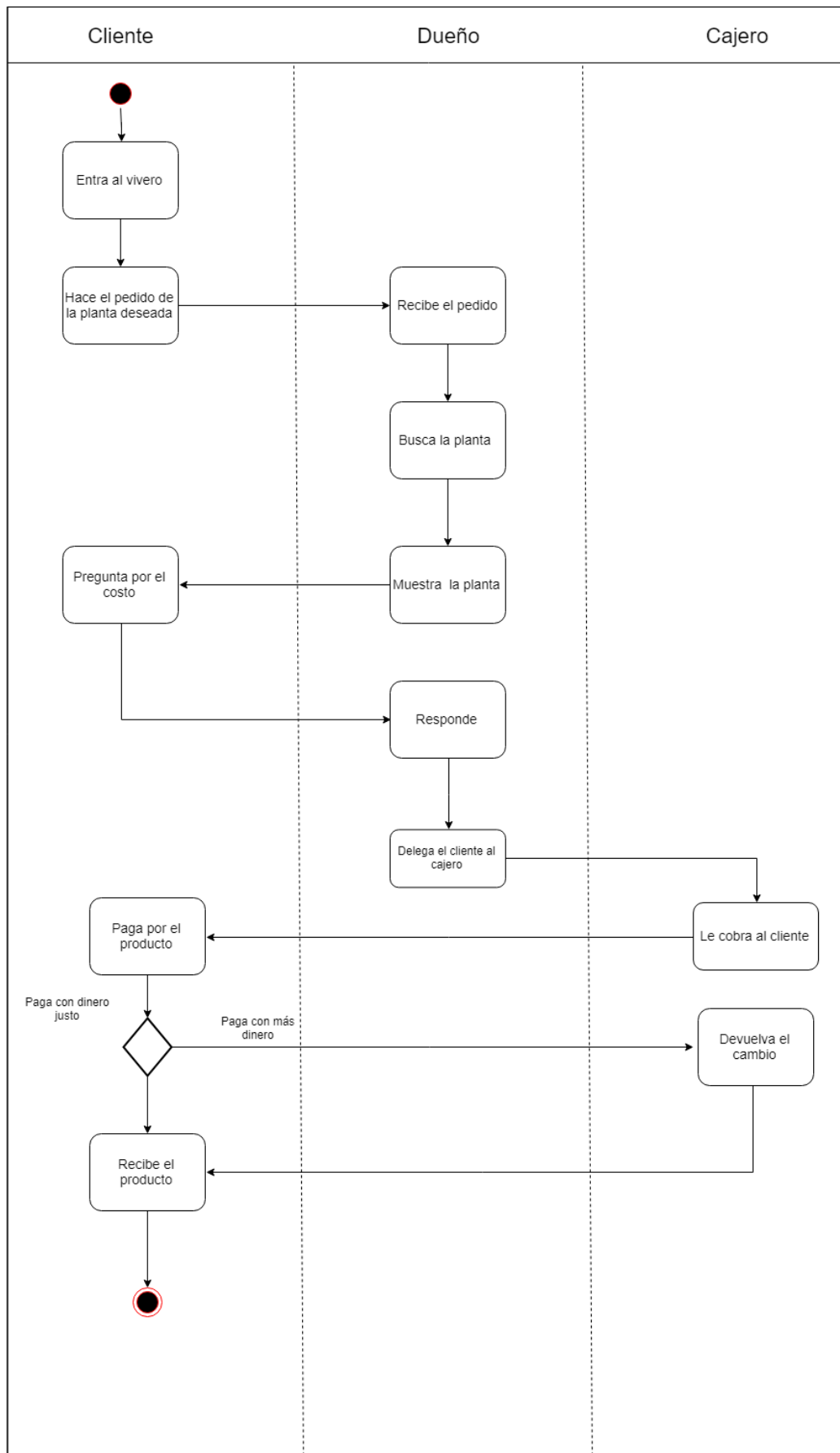
2.2.6 Descripción del proceso Poda

El dueño verifica que sea fecha de poda por memoria, y si lo es le encarga al jardinero la tarea de podar. El jardinero toma las tijeras, y se dirige hasta el patio, acercándose al árbol que debe de podar, cortando las ramas que se muestren sobrecrecidas. Repite la tarea con todos los árboles necesarios. Una vez terminado, se dirige a devolver las tijeras a su lugar, antes de informarle al dueño sobre su tarea. Cuando el dueño verifique que la tarea fue hecha correctamente, registra que se podaron los árboles en aquella fecha.

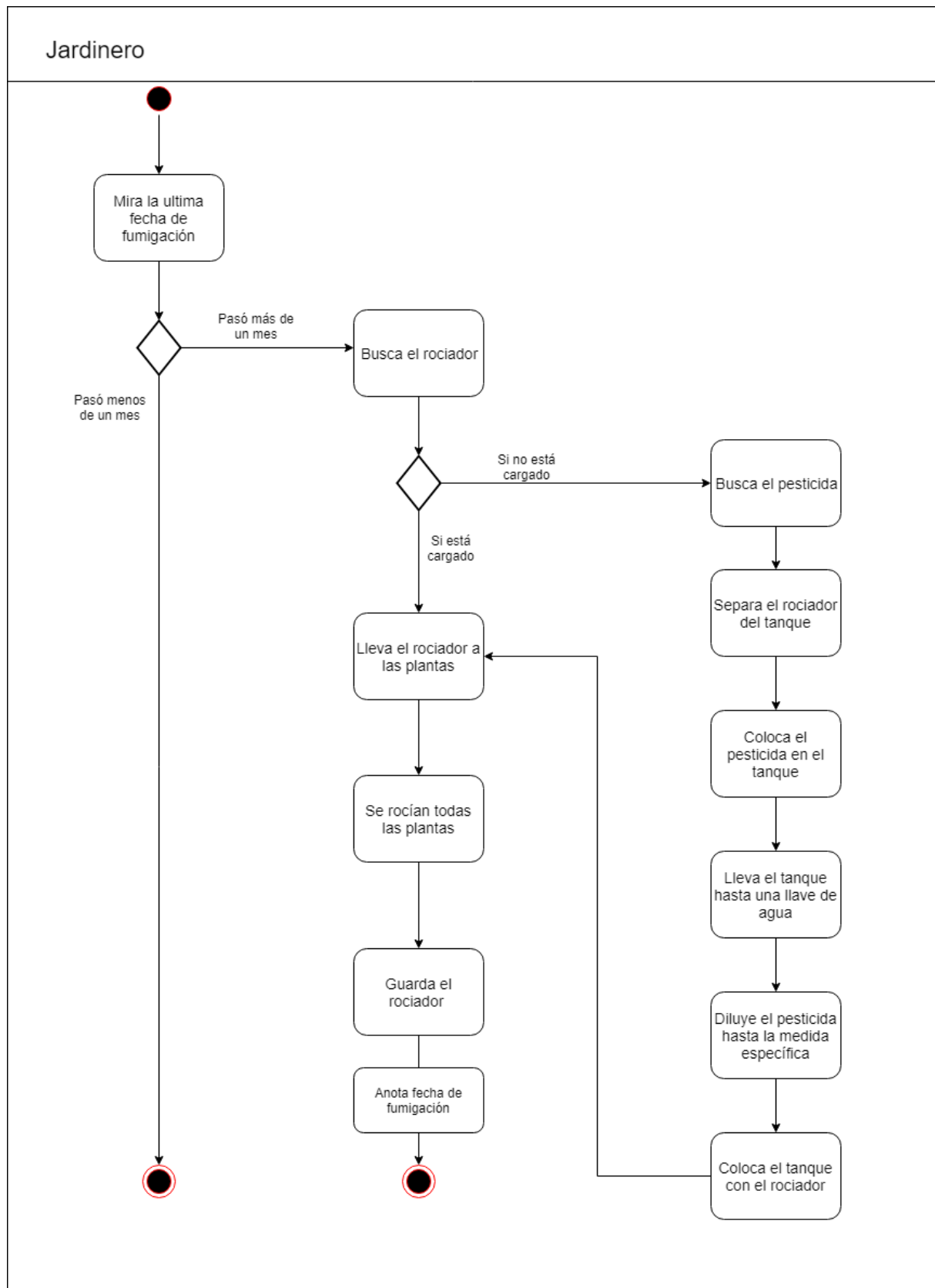
2.2.7 Diagrama del Proceso Plantar



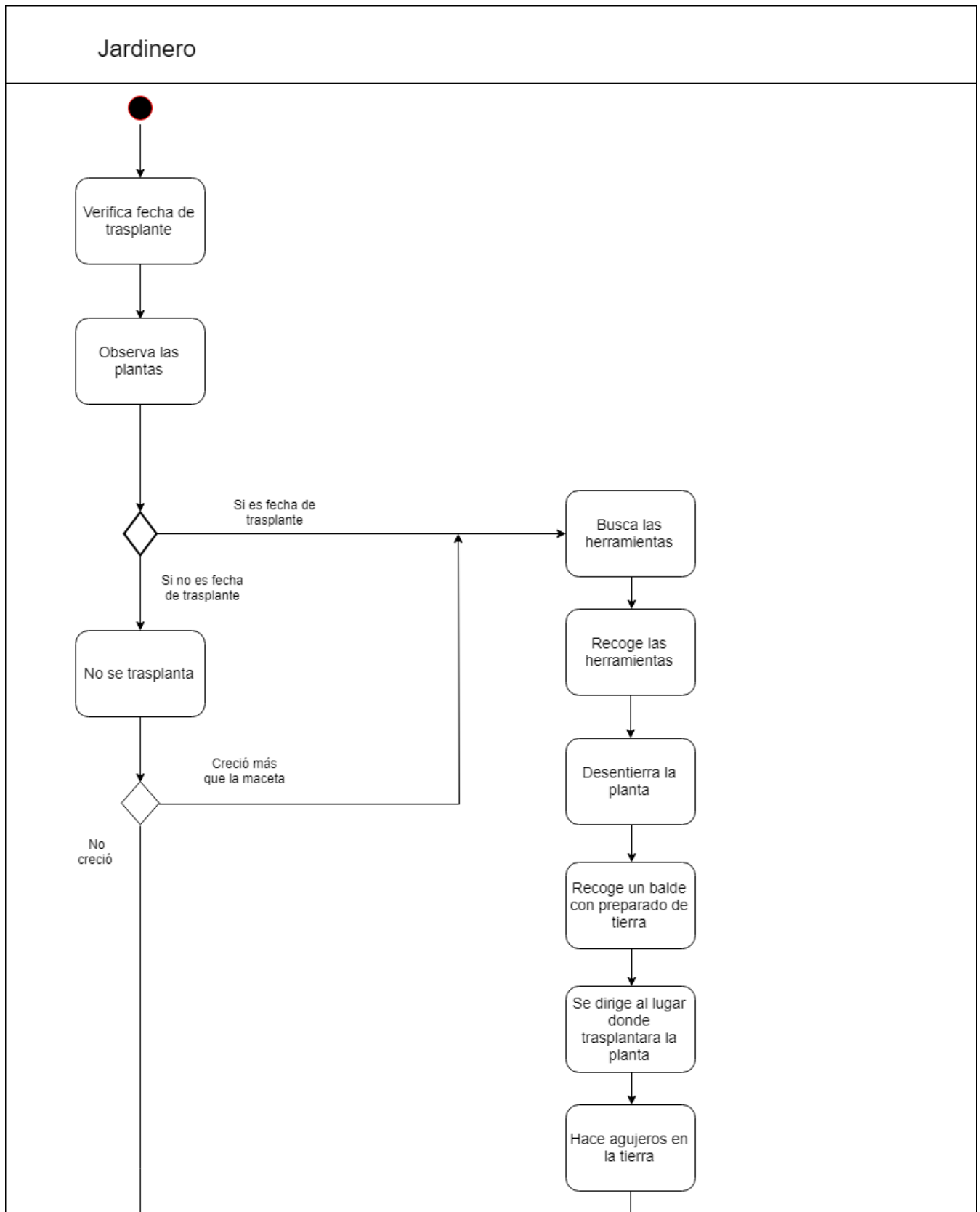
2.2.8 Diagrama del proceso Venta

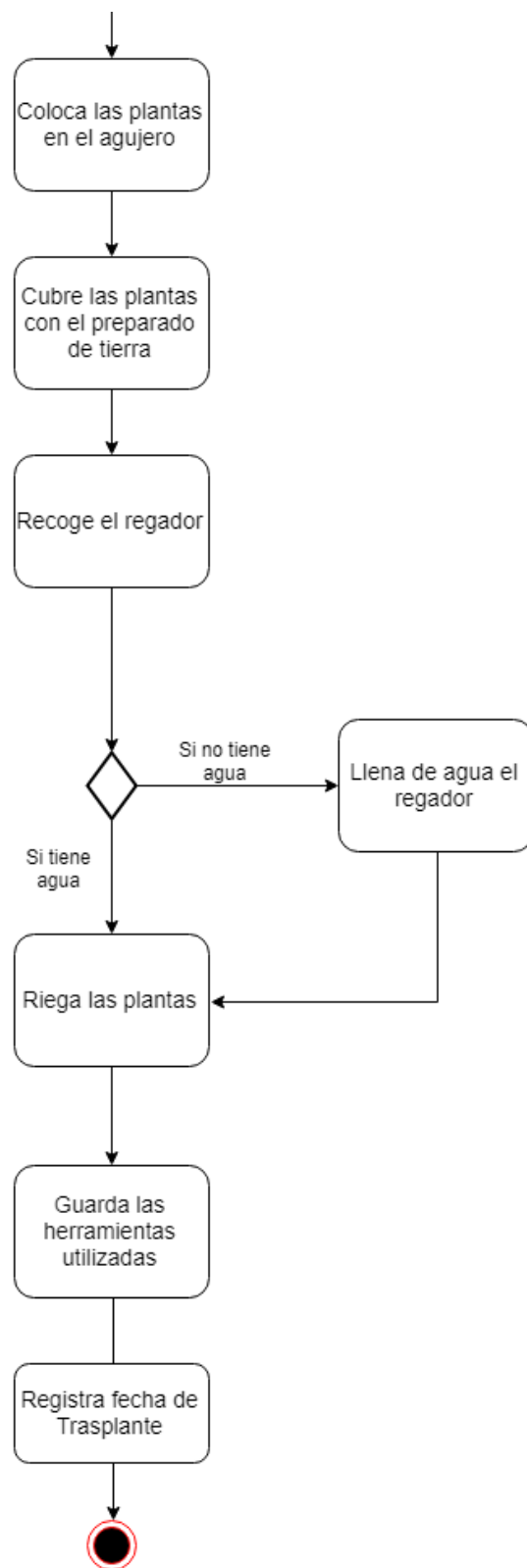


2.2.9 Diagrama del proceso Fumigar (Prevención)

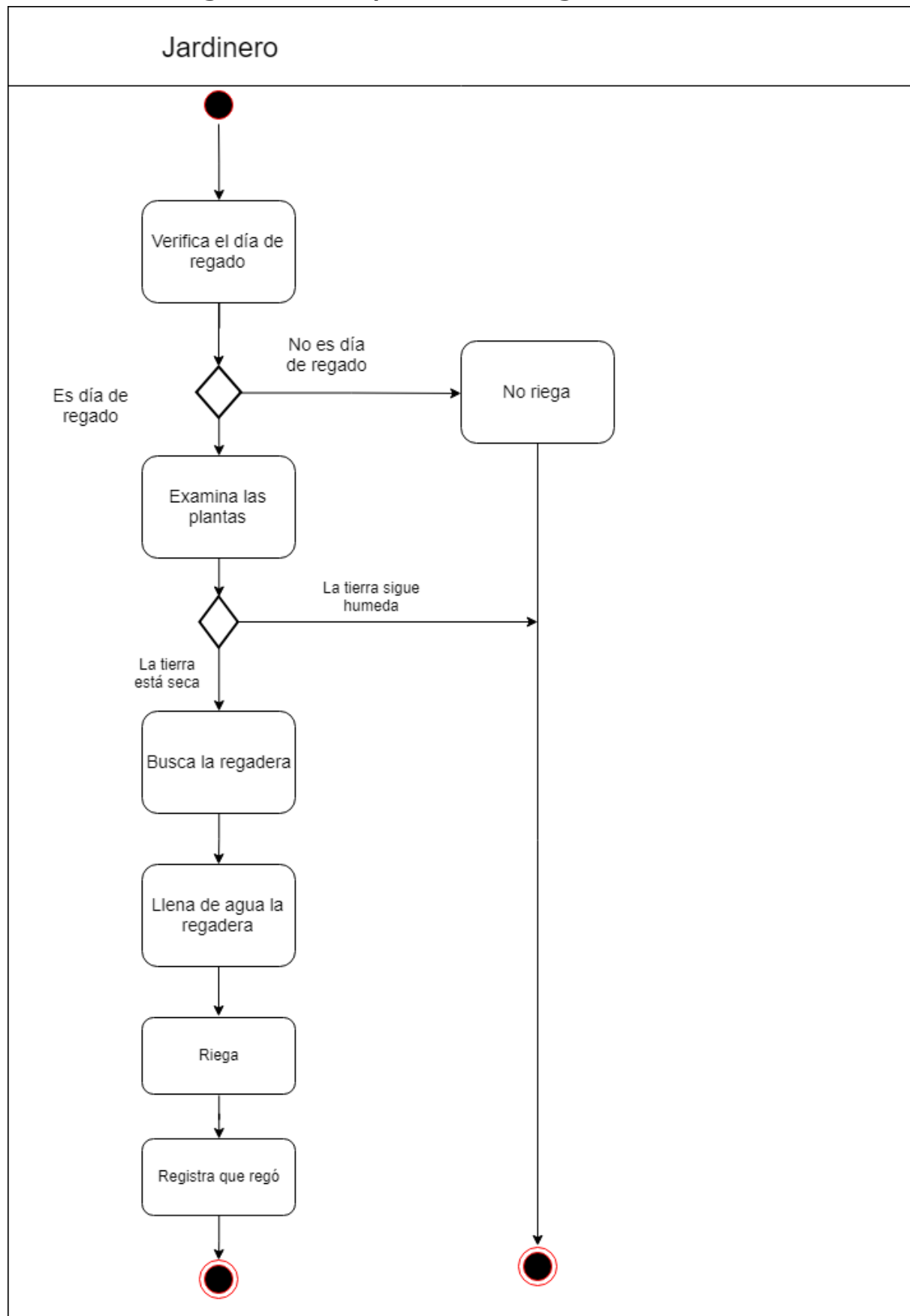


2.2.10 Diagrama del proceso Trasplantación

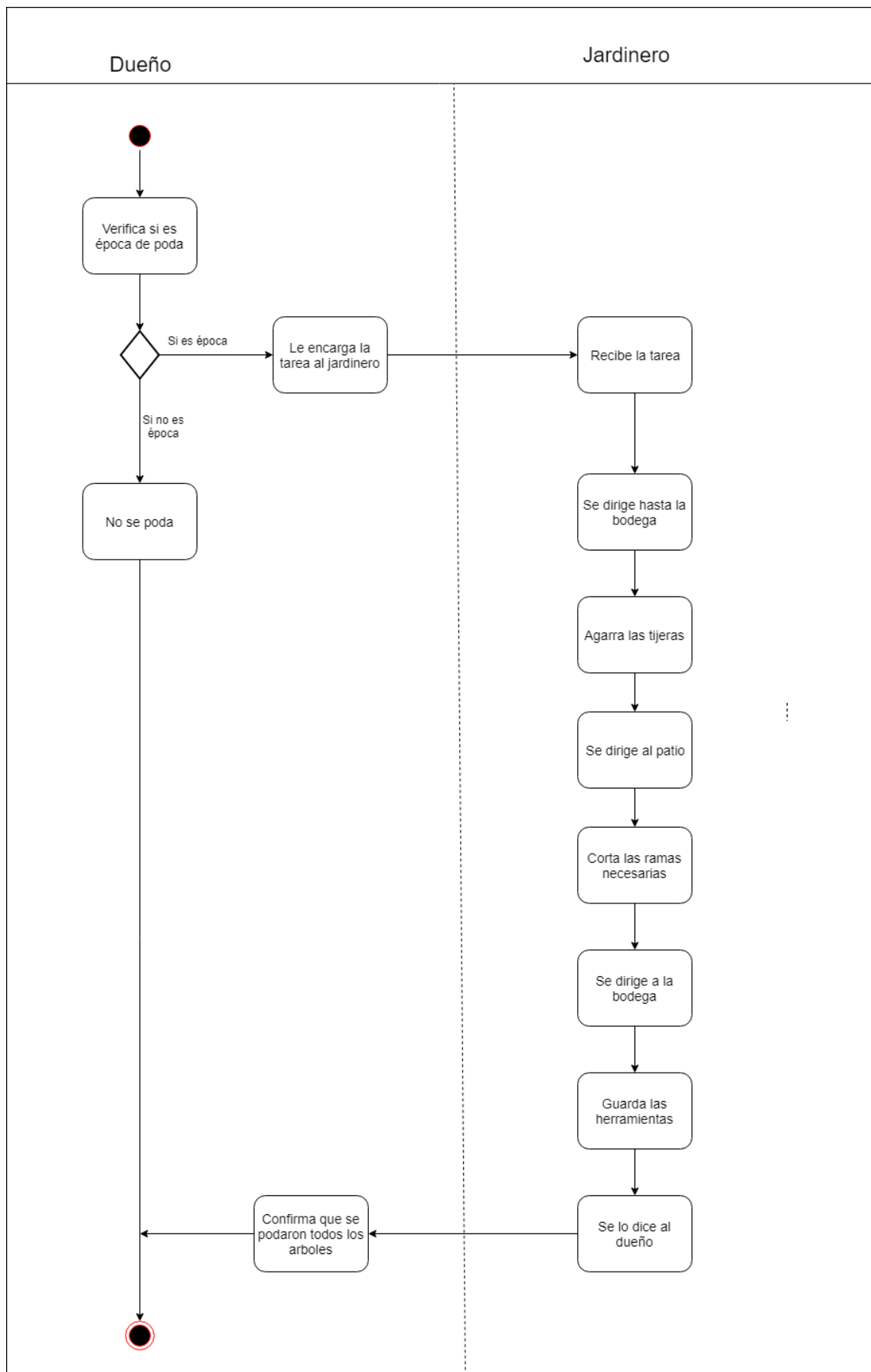




2.2.11 Diagrama del proceso Riego

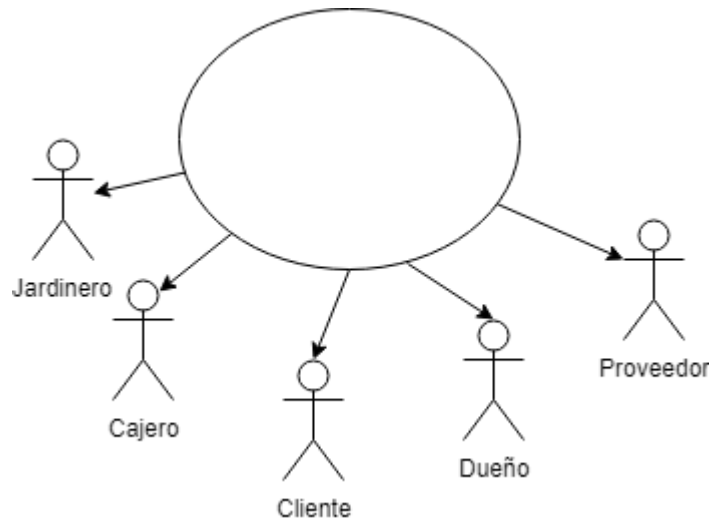


2.2.12 Diagrama del proceso Poda



2.3 Actores involucrados

2.3.1 Diagrama



2.3.2 Descripción

Dueño. Persona que ejerce dominio sobre determinada cosa o persona, o tiene poder sobre ella. Un dueño es un individuo que cuenta con el dominio, la potestad o el mando sobre algo o alguien.

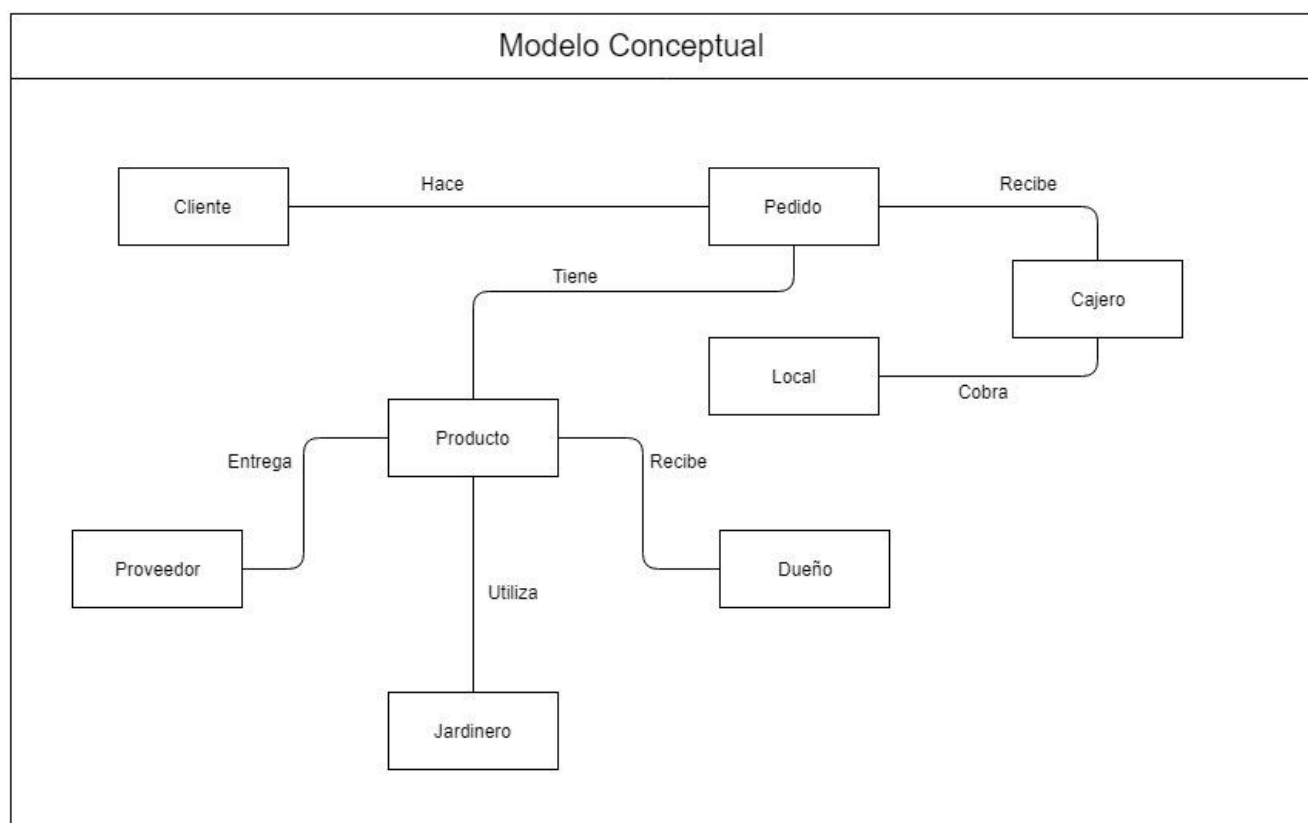
Cajero. Persona que tiene por oficio llevar el control de caja y atender los pagos y cobros en ciertos establecimientos.

Jardinero. Persona que tiene por oficio cuidar o cultivar los jardines.

Cliente. Persona que utiliza los servicios de un profesional o de una empresa, especialmente la que lo hace regularmente.

Proveedor. Persona que provee o abastece a otra persona de lo necesario o conveniente para un fin determinado.

2.4 Descripción del entorno



2.5 Lista de necesidades del cliente

2.5.1 Conjunto de necesidades de la empresa del cliente.

- NEC1. Controlar el acceso a la información.
- NEC2. Gestión de plantas.
- NEC3. Gestión de sensores de riego y climatización.
- NEC4. Gestión de usuarios.
- NEC5. Gestión de clientes y guardería de plantas
- NEC6. Gestión de poda.
- NEC7. Gestión de trasplantación.
- NEC8. Cronograma de tareas.
- NEC9. Registro de ventas.
- NEC10. Control de enfermedades de las plantas.

2.6 Objetivos

- OB1. Crear un Control de usuarios seguro.
- OB2. Informatizar la gestión de plantas.
- OB3. Informatizar la gestión de sensores de riego y climatización.
- OB4. Crear un registro de ventas.
- OB5. Informatizar sobre las enfermedades de las plantas.
- OB6. Tener una interfaz de fácil uso y amigable al usuario.
- OB7. Administrar usuarios
- OB8. Ver cronograma de actividades

2.6.1 Actores del Sistema

Van a existir 3 tipos de usuarios:

- Súper Usuario.
- Administrador.
- Usuario estándar.

El súper usuario será el único que podrá acceder a cualquier información del sistema.

El administrador será el único con privilegios para gestionar usuarios del sistema.

Los usuarios estándar podrán utilizar las funciones restantes del sistema.

2.7 Requerimientos funcionales

RF1.	Alta de cliente
RF2.	Baja de cliente
RF3.	Modificación de cliente
RF4.	Listado de cliente
RF5.	Alta de proveedor
RF6.	Baja de proveedor
RF7.	Modificación de proveedor
RF8.	Listado de proveedor
RF9.	Alta de empleado
RF10.	Baja de empleado
RF11.	Modificación de empleado
RF12.	Listado de empleado
RF13.	Alta de planta
RF14.	Baja de planta
RF15.	Modificación de planta
RF16.	Listado de planta
RF17.	Listado de enfermedades y medidas de prevención

2.7.1 Alta de cliente

El sistema permitirá registrar los datos de los clientes.

De los cuales se almacenarán:

- Cédula
- Nombre
- Apellido
- Teléfono
- Correo electrónico

Actores participantes: Súper usuario y Administrador del sistema.

2.7.2 Baja de cliente

El sistema permitirá eliminar de forma lógica los datos del cliente a partir de su cédula.

Actores participantes: Administrador del sistema.

2.7.3 Modificación de cliente

El sistema permitirá modificar de forma lógica los datos del cliente en caso de ser necesario.

Actores participantes: Administrador del sistema.

2.7.4 Listado de cliente

El sistema mostrará una lista con los datos del cliente

Actores participantes: Súper usuario y Administrador del sistema.

2.7.5 Alta de proveedor

El sistema permitirá registrar los datos de los proveedores.

De los cuales se almacenarán:

- Cédula
- Nombre
- Apellido
- Teléfono
- Correo electrónico

Actores participantes: Súper usuario y Administrador del sistema.

2.7.6 Baja de proveedor

El sistema permitirá eliminar de forma lógica los datos del proveedor a partir de su cédula.

Actores participantes: Administrador del sistema.

2.7.7 Modificación de proveedor

El sistema permitirá modificar de forma lógica los datos del proveedor en caso de ser necesario.

Actores participantes: Administrador del sistema.

2.7.8 Listado de proveedor

El listado mostrará una lista con los datos del proveedor.

Actores participantes: Súper usuario y Administrador del sistema.

2.7.9 Alta de empleado

El sistema permitirá registrar los datos de los empleados.

De los cuales se almacenarán:

- Cédula
- Nombre
- Apellido
- Rol de trabajo
- Teléfono

Actores participantes: Súper usuario y Administrador del sistema.

2.7.10 Baja de empleado

El sistema permitirá eliminar de forma lógica los datos del empleado a partir de su cédula.

Actores participantes: Administrador del sistema.

2.7.11 Modificación de empleado

El sistema permitirá modificar de forma lógica los datos del proveedor en caso de ser necesario.

Actores participantes: Súper usuario y Administrador del sistema.

2.7.12 Listado de empleado

El sistema mostrará una lista con los datos de los empleados

Actores participantes: Súper usuario y Administrador del sistema.

2.7.13 Alta de planta

El sistema permitirá registrar los datos de las plantas.

De los cuales se almacenarán:

- Id
- Nombre
- Tipo de especie
- Fecha de plantación

Actores participantes: Súper usuario y Administrador del sistema.

2.7.14 Baja de planta

El sistema eliminara los datos de la planta a partir de su Id

Actores participantes: Administrador del sistema.

2.7.15 Modificación de planta

El sistema permitirá modificar los datos de las plantas en caso de ser necesario

Actores participantes: Súper usuario y Administrador del sistema.

2.7.16 Listado de planta

El sistema mostrará una lista con los datos de la planta.

Actores participantes: Súper usuario y Usuarios estándares

2.7.17 Listado de enfermedades de la planta y medidas de prevención

El sistema mostrará una lista de enfermedades que puedan tener las plantas y los medicamentos para los mismos

Actores participantes: Usuarios estándares.

2.8 Requerimientos no Funcionales

Se desarrollará mediante:

- Lenguaje de programación Visual Basic .NET 2015.
- Sistema de gestor de base de datos MySQL, instalado en un servidor remoto Linux.
- Servidor web y herramientas necesarias.
- Servidor DHCP para la red local.
- Placa Arduino y sensores necesarios.

2.9 Alcances y limitaciones

Para el desarrollo de este software de gestión se utilizará el lenguaje de programación Visual Basic .NET 2015, con una base de datos ya instalado en un servidor remoto Linux. Tendrá un servidor web y se tiene previsto usar placa Arduino con sensores necesarios para el desarrollo. El sistema contará con una interfaz de fácil uso y amigable para el usuario, pero no contará con mantenimiento a largo plazo después de concluido.

2.10 Informe sobre características fundamentales del lenguaje VB.NET

Visual Basic .NET (VB.NET) es un lenguaje de programación orientado a objetos que se puede considerar una evolución de Visual Basic implementada sobre el framework .NET. Su introducción resultó muy controvertida, ya que debido a cambios significativos en el lenguaje VB.NET no es retro compatible con Visual Basic, pero el manejo de las instrucciones es similar a versiones anteriores de Visual Basic, facilitando así el desarrollo de aplicaciones más avanzadas con herramientas modernas.

2.10.1 Características de VB.NET

2.10.2 Instrucciones en VB .NET

Una instrucción en Visual Basic es una instrucción completa. Puede contener las palabras clave, operadores, variables, constantes y expresiones. Cada instrucción pertenece a una de las siguientes categorías:

- Las instrucciones de declaración, que el nombre de una variable, constante o procedimiento y también puede especificar un tipo de datos.
- Las instrucciones ejecutables, que inician acciones. Estas instrucciones pueden llamar a un método o función, y pueden bucle o a través de bloques de código. Las instrucciones ejecutables incluyen instrucciones de asignación, que asignan un valor o expresión a una variable o constante.

Instrucciones de declaración

Utilice las instrucciones de declaración para un nombre y definen procedimientos, variables, propiedades, matrices y constantes. Cuando se declara un elemento de programación, también puede definir su tipo de datos, el nivel de acceso y el ámbito.

2.10.3 Procedimientos en VB .NET

Un *procedimiento* es un bloque de instrucciones de Visual Basic dentro de una instrucción de declaración (Function, Sub, Operator, Get, Set) y una búsqueda de coincidencias End declaración. Todas las instrucciones ejecutables en Visual Basic deben estar dentro de algún procedimiento.

2.10.4 Llamada al procedimiento

Cuando invoca un procedimiento desde otra parte del código, está realizando una *llamada a procedimiento*. Cuando el procedimiento termina de ejecutarse, devuelve el control al código que lo ha invocado, el cual se conoce como código de llamada. El código de llamada es una instrucción, o una expresión de una instrucción, que especifica el procedimiento por el nombre y le transfiere el control.

Devolución desde un procedimiento

Un procedimiento devuelve el control al código de llamada cuando termina de ejecutarse. Para ello, puede usar una instrucción Return, la instrucción Exit adecuada para el procedimiento o la instrucción End keyword Statement del procedimiento. A continuación, el control pasa al código de llamada seguido del punto de la llamada a procedimiento.

- Con una instrucción Return, el control vuelve inmediatamente al código de llamada. Las instrucciones que siguen la instrucción Return no se ejecutan. Puede tener más de una instrucción Return en el mismo procedimiento.
- Con una instrucción Exit Sub o Exit Function, el control vuelve inmediatamente al código de llamada. Las instrucciones siguientes a la instrucción Exit no se ejecutan. Puede tener más de una instrucción Exit en el mismo procedimiento. Además, puede combinar instrucciones Return y Exit en el mismo procedimiento.
- Si un procedimiento no tiene instrucciones Return o Exit, concluye con una instrucción End Sub o End Function, End Get, o End Set posterior a la última instrucción del cuerpo del procedimiento. La instrucción End devuelve inmediatamente el control al código de llamada. Solo puede tener una instrucción End en un procedimiento.

2.10.5 Tipos de procedimientos

Visual Basic usa varios tipos de procedimientos:

- Los procedimientos Sub realizan acciones, pero no devuelven un valor al código de llamada.
- Los procedimientos de control de eventos son procedimientos Sub que se ejecutan en respuesta a un evento debido a las acciones del usuario o a un incidente en un programa.
- Los procedimientos Function devuelven un valor al código de llamada. Pueden realizar otras acciones antes de devolver el valor.
- Algunas funciones escritas en C# devuelven un *valor devuelto de referencia*. Los autores de llamadas a funciones pueden modificar el valor devuelto. Esta modificación se refleja en el estado del objeto al que se ha llamado. A partir de Visual Basic 2017, el código de Visual Basic puede consumir referencias de valores devueltos, aunque no puede devolver un valor por la referencia. Para obtener más información, consulte Valores devueltos de referencia.
- Los procedimientos Property devuelven y asignan valores de propiedades en objetos o módulos.
- Los procedimientos Operator definen el comportamiento de un operador estándar cuando uno de los operandos (o ambos) es una clase definida recientemente o una estructura.
- Los procedimientos genéricos de Visual Basic definen uno o varios *parámetros de tipo* además de sus parámetros normales, por lo que el código de llamada puede pasar tipos de datos determinados cada vez que se hace una llamada.

2.10.6 Tipo de datos

El tipo de datos de un elemento de programación hace referencia al tipo de datos que puede contener y cómo almacena los datos. Los tipos de datos se aplican a todos los valores que se pueden almacenar en la memoria del equipo o participar en la evaluación de una expresión. Cada variable, literal, constante, enumeración, propiedad, parámetro de procedimiento, argumento de procedimiento y valor devuelto de un procedimiento tiene un tipo de datos.

Tipos de datos declarados

Define un elemento de programación con una instrucción de declaración y especifica su tipo de datos con la cláusula As.

Elemento de programación	Declaración de tipos de datos
Variable	<p>En una instrucción Dim</p> <p>Dim amount As Double</p> <p>Static yourName As String</p> <p>Public billsPaid As Decimal = 0</p>
Literal	<p>Con un carácter de tipo literal; consulte "Caracteres de tipo literal" en Caracteres de tipo</p> <p>Dim searchChar As Char = "." C</p>
Constante	<p>En una instrucción Const</p> <p>Const modulus As Single = 4.17825F</p>
Enumeración	<p>En una instrucción Enum</p> <p>Public Enum colors</p>
Propiedad	<p>En una instrucción Property</p> <p>Property region() As String</p>
Parámetro de procedimiento	<p>En una instrucción Sub, instrucción Function o una instrucción Operator</p> <p>Sub addSale(ByVal amount As Double)</p>
Argumento de procedimiento	<p>En el código de llamada; cada argumento es un elemento de programación que ya se declaró, o bien una expresión que contiene los elementos declarados</p> <p>subString = Left(inputString , 5)</p>
Valor devuelto de un procedimiento	<p>En una instrucción Function o una instrucción Operator</p> <p>Function convert(ByVal b As Byte) As String</p>

2.10.7 Operadores y expresiones en VB .NET

Un operador es un elemento de código que realiza una operación en uno o más elementos de código que contienen valores. Los elementos de valor incluyen variables, constantes, literales, propiedades, devoluciones de procedimientos Function y Operator y expresiones.

Una expresión es una serie de elementos de valor combinados con operadores, lo que produce un nuevo valor. Los operadores actúan en los elementos de valor mediante cálculos, comparaciones y otras operaciones.

Tipos de operadores

Visual Basic proporciona los siguientes tipos de operadores:

- Los operadores aritméticos realizan cálculos familiares en valores numéricos, incluido el desplazamiento de los patrones de bits.
- Los operadores de comparación comparan dos expresiones y devuelven un valor Boolean que representa el resultado de la comparación.
- Los operadores de concatenación unen varias cadenas en una sola.
- Los operadores lógicos y bit a bit de Visual Basic combinan valores Boolean o numéricos y devuelven un resultado del mismo tipo de datos que los valores.

2.10.8 Objetos y clases

Cada objeto en Visual Basic se define por una clase. Una clase describe las variables, las propiedades, los procedimientos y los eventos de un objeto. Los objetos son instancias de clases; puede crear tantos objetos como sean necesarios una vez que haya definido una clase.

Para comprender la relación entre un objeto y su clase, piense en las galletas y en su molde. El molde de la galleta es la clase. Define las características de cada galleta, por ejemplo, tamaño y forma. La clase se usa para crear objetos. Los objetos son las galletas. Debe crear un objeto para poder acceder a sus miembros.

2.10.9 Interfaces

Las interfaces definen las propiedades, los métodos y los eventos que pueden implementar las clases. Las interfaces permiten definir características como grupos pequeños de propiedades, métodos y eventos estrechamente relacionados; esto reduce los problemas de compatibilidad porque se pueden desarrollar implementaciones mejoradas para las interfaces sin comprometer el código existente. En cualquier momento se pueden agregar nuevas características mediante el desarrollo de implementaciones e interfaces adicionales.

Hay otras razones por las que se podrían usar las interfaces en vez de la herencia de clases:

- Las interfaces se adaptan mejor a situaciones en las que las aplicaciones necesitan que muchos tipos de objetos posiblemente no relacionados proporcionen una funcionalidad determinada.
- Las interfaces son más flexibles que las clases base porque se puede definir una sola implementación que implemente varias interfaces.
- Las interfaces funcionan mejor en situaciones en las que no es necesario heredar la implementación de una clase base.
- Las interfaces son útiles cuando no se puede usar la herencia de clases. Por ejemplo, las estructuras no pueden heredar de clases, pero pueden implementar interfaces.

2.10.10 Flujo de Control en VB .NET

Si un programa no se regula, procede a través de las instrucciones que contiene de principio a fin. Algunos programas muy simples pueden escribirse únicamente con este flujo unidireccional. Pero gran parte del rendimiento y la utilidad de un lenguaje de programación se debe a la capacidad de cambiar el orden de ejecución mediante instrucciones de control y bucles.

Las estructuras de control permiten regular el flujo de la ejecución del programa. Uso de estructuras de control, puede escribir código de Visual Basic que tome decisiones o repita acciones. Otras estructuras de control permiten garantizar la eliminación de un recurso o ejecutar una serie de instrucciones en la misma referencia de objeto.

2.11 Informe sobre Sistema Gestor de Base de Datos elegido



MySQL es una base de datos muy rápida en la lectura cuando utiliza el motor no transaccional MyISAM, pero puede provocar problemas de integridad en entornos de alta concurrencia en la modificación. En aplicaciones web hay baja concurrencia en la modificación de datos y en cambio el entorno es intensivo en lectura de datos, lo que hace a MySQL ideal para este tipo de aplicaciones. Sea cual sea el entorno en el que va a utilizar MySQL, es importante monitorizar de antemano el rendimiento para detectar y corregir errores tanto de SQL como de programación.

Utilidad de MySQL

Como comentábamos anteriormente este gestor de base de datos es muy utilizado en desarrollo web, ya que permite a los desarrolladores y diseñadores, realizar cambios en sus sitios de manera simple, con tan sólo cambiar un archivo, evitando tener que modificar todo el código web. Esto se debe a que MySQL, trabaja con un sistema centralizado de gestión de datos, que permite realizar cambios en un solo archivo y que se ejecuta en toda la estructura de datos que se comparte en la red. Además, permite incluir noticias e información rápidamente en un sitio web, utilizando un simple formulario, sin tener que tocar el código del website.

Características

- Software libre
- Licencia GNU/GPL
- Velocidad de accesibilidad a los datos
- Múltiples motores de almacenamiento
- Agrupación de transacciones, reuniendo múltiples transacciones de varias conexiones para incrementar el número de transacciones por segundo, etc.

Ventajas

- MySQL software es Open Source.
- Velocidad al realizar las operaciones, lo que le hace uno de los gestores con mejor rendimiento.
- Bajo costo en requerimientos para la elaboración de bases de datos, ya que debido a su bajo consumo puede ser ejecutado en una máquina con escasos recursos sin ningún problema.
- Facilidad de configuración e instalación.
- Soporta gran variedad de Sistemas Operativos.

- Baja probabilidad de corromper datos, incluso si los errores no se producen en el propio gestor, sino en el sistema en el que está.
- Su conectividad, velocidad y seguridad hacen de MySQL Server altamente apropiado para acceder bases de datos en Internet.
- El software MySQL usa la licencia GPL.

Desventajas

- No es intuitivo, como otros programas (ACCESS).
- Un gran porcentaje de las utilidades de MySQL no están documentadas.



Oracle la Primera Base de Datos Diseñada para Grid Computing, es un sistema de gestión de base de datos relacional fabricado por Oracle Corporation. Oracle es básicamente una herramienta cliente/servidor para la gestión de base de datos la gran potencia que tiene y su elevado precio hace que solo se vea en empresas muy grandes y multinacionales, por norma general.

Características

Desarrollado sobre Oracle Database, Oracle Content Database ha sido diseñada para que las organizaciones puedan controlar y gestionar grandes volúmenes de contenidos no estructurados en un único repositorio con el objetivo de reducir los costes y los riesgos asociados a la pérdida de información.

Ventajas

Ventaja: sistema de gestión y control centralizado

Las sentencias de Oracle SQL permiten que los datos se controlen desde un repositorio central tabular. Un administrador de bases de datos (DBA por sus siglas en inglés) es responsable de crear usuarios, asignar privilegios, añadir registros, eliminar información redundante, modificar datos existentes y procesar preguntas.

Estos datos almacenados centralmente son compartidos y accedidos por varias aplicaciones. Esto elimina la redundancia en la entrada y almacenamiento de datos.

Ventaja: estandarización

Una ventaja principal de Oracle SQL es su estandarización y consistencia entre distintas implementaciones. SQL fue estandarizado por primera vez por el ANSI (Instituto Estadounidense de Estandarización) en 1986, y luego ratificado en 1987 por la Organización Internacional de Estandarización (ISO), el cual sigue siendo el organismo de estandarización.

- Oracle es el motor de base de datos objeto-relacional más usado a nivel mundial.
- Puede ejecutarse en todas las plataformas, desde una Pc hasta un supercomputador.
- Oracle soporta todas las funciones que se esperan de un servidor “serio”: un lenguaje de diseño de bases de datos muy completo (PL/SQL) que permite implementar diseños “activos”, con triggers y procedimientos almacenados, con una integridad referencial declarativa bastante potente.
- Permite el uso de particiones para la mejora de la eficiencia, de replicación e incluso ciertas versiones admiten la administración de bases de datos distribuidas.
- El software del servidor puede ejecutarse en multitud de sistemas operativos.
- Existe incluso una versión personal para Windows 9x, lo cual es un punto a favor para los desarrolladores que se llevan trabajo a casa.
- Oracle es la base de datos con más orientación hacia INTERNET.

Desventajas

- Las versiones más recientes de Oracle son la 11g, 10g, 9g, 8g, desde el lanzamiento original de la 8 se sucedieron varias versiones con correcciones, hasta alcanzar la estabilidad en la 8.0.3. El motivo de tantos fallos fue, al parecer, la remodelación del sistema de almacenamiento por causa de la introducción de extensiones orientadas a objetos.
- El mayor inconveniente de Oracle es quizás su precio. Incluso las licencias de Personal Oracle son excesivamente caras, en mi opinión. Otro problema es la necesidad de ajustes. Un error frecuente consiste en pensar que basta instalar el Oracle en un servidor y enchufar directamente las aplicaciones clientes. Un Oracle mal configurado puede ser desesperantemente lento.
- También es elevado el coste de la información, y sólo últimamente han comenzado a aparecer buenos libros sobre asuntos técnicos distintos de la simple instalación y administración.
- Desventaja: inhabilidad de implementar el procesamiento recursivo

- De acuerdo con “SQL para tontos”, una de las mayores desventajas de SQL es su incapacidad de ejecutar procesamiento recursivos. El procesamiento recursivo es un tipo de función de computadora (o programa) en el cual uno de los pasos o procedimientos vuelve a hacer correr el programa entero (o el procedimiento). SQL carece de construcciones de tipo lazo que son comunes en otros tipos de lenguajes de programación de alto nivel. No se pueden repetir acciones y no hay forma de definir construcciones repetitivas en SQL.
- Desventaja: incompatibilidad y complejidad
- Una de las mayores desventajas de Oracle SQL es la inconsistencia e incompatibilidad de datos en las áreas del tiempo y sintaxis de datos, concatenación de cadenas y sensibilidad de caracteres. El lenguaje es complejo, con un enfoque de palabras clave similar en estructura a COBOL (por las cifras en inglés de lenguaje común orientado a los negocios), con menos reglas de sintaxis y gramática.
- Desventaja: funcionalidad limitada
- SQL es un dominio específico o lenguaje de propósito especial, y su uso está limitado a un dominio de programa específico. Las sentencias de SQL son operadas en tablas y conjuntos de datos, como por ejemplo bases de datos de personal y hojas de cálculo de contabilidad. SQL es un lenguaje declarativo específico de dominio que está limitado a la representación tabular de los datos.



Visual FoxPro es un lenguaje de programación orientado a objetos y procedural, un Sistema Gestor de Bases de datos o Database Management System (DBMS), y desde la versión 7.0, un Sistema administrador de bases de datos relacionales, producido por Microsoft.

Sistema orientado a objetos para la generación de Base de Datos y el desarrollo de aplicaciones. Desde Visual FoxPro se pueden crear Formulario, Base de Datos, Informes, Vistas, Proyectos entre otros.

Características

Visual FoxPro ofrece a los desarrolladores un conjunto de herramientas para crear aplicaciones de bases de datos para el escritorio, entornos cliente/servidor, Tablet PC o para la Web.

Entre sus características se pueden enumerar:

- Capacidades poderosas y muy veloces para el manejo de datos nativos y remotos.
- Flexibilidad para crear todo tipo de soluciones de bases de datos.
- Lenguaje de Programación orientado a objetos.
- Utilización de Sentencias SQL en forma nativa.
- Poderoso manejo de vistas, cursores y control completo de estructuras relacionales.
- Su propio gestor de base de datos incorporado. Sin embargo, también puede conectarse con servidores de base de datos, tales como Oracle, Microsoft SQL Server o MySQL.
- Cuenta con un motor de generación de informes renovado y muy flexible para soluciones más robustas.
- Desde la versión 9.0, amplio soporte de XML, tanto como fuente de datos (por ej., servicios Web basados en XML) como por generar reportes en formato XML.
- Desde la versión 7.0, soporte de la tecnología Intel lisense de Microsoft.

Ventajas

- Menor costo
- Adquisición (Es asequible para pequeñas empresas)
- Capacitación (Por ser más popular)
- Soporte (Por la misma razón de antes)
- Desarrollo (Encontras más programadores y menos costosos)
- Entorno Windows (Es más familiar y predecible)

Desventajas

- Integridad (Mayor sensibilidad frente a variaciones en transmisiones y fluctuaciones eléctricas).
- Seguridad (Menos niveles de acceso).
- Menor capacidad de procesamiento (no funciona adecuadamente con grandes volúmenes de información). Para volúmenes medianos se utiliza el motor de Microsoft SQL, sin mayores traumatismos y excelente performance.



Microsoft Access. Sistema de Administración de Bases de Datos para Microsoft Windows el cual pone a su alcance la capacidad de organizar, buscar y presentar información, aprovechando al máximo la potencia gráfica de Windows ofreciendo métodos visuales de acceso a sus datos y proporcionando maneras simples y directas de presentar y trabajar con su información.

Ventajas

Por medio de Microsoft Access, puede administrar toda la información desde un único archivo de base de datos. Dentro del archivo, puede utilizar:

- Tablas para almacenar los datos.
- Consultas para buscar y recuperar únicamente los datos que necesita.
- Formularios para ver, agregar y actualizar los datos de las tablas.

- Informes para analizar o imprimir los datos con un diseño específico.
- Páginas de acceso a datos para ver, actualizar o analizar los datos de la base de datos desde Internet.
- Almacenar los datos una vez en una tabla y verlos desde varios lugares.

Desventajas

- Tiene limitaciones en el procesamiento de las búsquedas, además que si la quieres usar para ambientes corporativo no te va a servir por su poca estabilidad, si quieres usarla para proyectos de la escuela o bases de pequeños negocios que no requieren alta disponibilidad, la puedes utilizar.
- Si quiere una base de datos más robusta, para ambientes más grandes, aunque un poco más complicada puede usar MYSQL es gratuita si es para sistemas internos o personales y con un pequeño pago para que puedas distribuirla.
- SQL express de Microsoft también es gratuita y funciona bien para integrar sistemas .NET y también puedes pedir una licencia para usarla en sistemas que quieras comercializar, aunque no sé si tiene un costo por esto.
- SQL server es mucho más completa y puede soportar bases de datos empresariales con alta cantidad de peticiones, pero esta es algo cara (si no la consigues en pirata).
- Oracle, es una de las opciones más completas para grandes ambientes transaccionales de alta disponibilidad, es muy completa pero el costo de las licencias es extremadamente alto.

2.12 Investigación de Sensores

Los sensores son aparatos que son capaz de transformar magnitudes físicas o químicas, llamadas variables de instrumentación, en magnitudes eléctricas. Las variables de instrumentación dependen del tipo de sensor y pueden ser, por ejemplo: temperatura, intensidad lumínica, distancia, aceleración, inclinación, desplazamiento, presión, fuerza, torsión, humedad, pH, etc. Una magnitud eléctrica obtenida puede ser una resistencia eléctrica (como en una RTD), una capacidad eléctrica (como en un sensor de humedad), una tensión eléctrica (como en un termopar), una corriente eléctrica (como un fototransistor), etc.

Entre las características principales técnicas de un sensor Se pueden clasificar en dos tipos: Estáticas y Dinámicas.

- Estáticas. Las características de los instrumentos cuando estos están midiendo cantidades estables, o sea, mientras no presentan variaciones bruscas en su magnitud.
- Dinámicas. Puede ocurrir que la cantidad bajo medición sufra una variación en un momento determinado y por lo tanto es necesario que conozcamos el comportamiento dinámico del instrumento cuando sucedan estas variaciones.

Tipos de sensores compatibles con Arduino.

Sensor de Corriente CA y CD 20A

Se trata de un módulo con el integrado ACS712ELCTR-05B-T (sensor de corriente).

El sensor proporciona una medición precisa de corriente para CA y señales de CD.

Conductor de cobre grueso permite la supervivencia del dispositivo hasta 5 veces las condiciones de sobre corriente.

La corriente debe pasar por el signo positivo a negativo.



Características

- Voltaje de operación 5V CD.
- Medición de corriente CA y CD.
- Compatible con Arduino.
- Interfaz de 3 pines.
- Rango máximo: 20^a.

Sensor de Humedad y Temperatura DTH11

Este sensor de temperatura y humedad proporciona una salida digital pre-calibrada.

Cuenta con una excelente fiabilidad y estabilidad a largo plazo. Este sensor no funciona para temperaturas por debajo de 0 grados.



Características

- Voltaje de operación 5V CD.
- Humedad relativa y medición de la temperatura.
- Compatible con Arduino.
- Bajo consumo de energía.

Sensor de inclinación

Este sensor de inclinación es un interruptor equivalente a un botón, y se utiliza como una entrada digital. Cuando está nivelado el interruptor queda abierto, y cuando se inclina, se cierra el interruptor y emitirá la señal a S.



Características

- Voltaje de operación 5V CD.
- Interfaz de 3 pines
- Compatible con Arduino.

Sensor Magnético

Esta es una interfaz electrónica que se puede utilizar como interruptor magnético.

Se basa en el CT10. CT10 es unipolar y tiene contactos normalmente abiertos de rutenio. Este sensor puede ser accionado por un electroimán, un imán o una combinación de ambos. El interruptor magnético es un instrumento que se puede utilizar para un circuito de encendido y apagado basado en la proximidad.

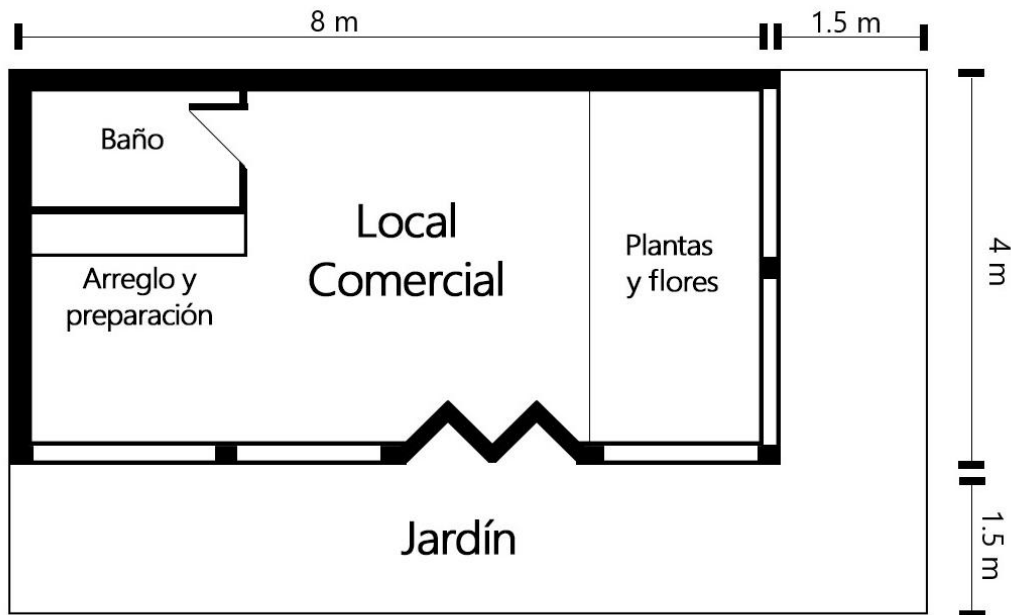


Características.

- Voltaje de operación 5V CD.
- Interfaz de 3 pines
- Compatible con Arduino

2.12.1 Investigación de Planta de Trabajo y Equipamiento

Plano de la planta de trabajo



2.12.2 Justificación técnica de la elección del SO de los Servidores

Hemos decidido utilizar Ubuntu server debido a que es el SO con el que contamos experiencia además de sus ventajas de costes al ser un sistema gratuito y su personalización en caso de ser requerida al ser completamente abierta, sin contar además con el soporte que brinda la misma comunidad en variadas situaciones permitiendo la facilidad a la hora de personalizar las opciones del servidor.

2.12.3 Licenciamiento

Licencias de Ubuntu Server

Basadas en la distribución Debian, las distribuciones Ubuntu se entregan de base con acceso SSH.

Esta distribución no es compatible con algunas de nuestras gamas de servidores, para las cuales OVH ha concebido OVH U-Release: Un sistema operativo instalado basado en el software de Open Source Ubuntu y adaptado por OVH para garantizar la compatibilidad de los componentes tecnológicos del servidor con el entorno del software. Ubuntu es una marca registrada de Canonical Ltd y tiene licencia libre.

Licencias GNU/Linux

- GPL: Aplica copyleft a determinadas versiones siempre y cuando estén bajo licencia GPL y sean libres.
- BDS: Posibilita que otras versiones puedan disponer de otras licencias diferentes tipos BSD o GPL
- MPL: Aplica licencias dobles a los ejecutables y al código fuente, obligando a entregar las modificaciones del código a su creador, licenciado sólo los binarios.

Licencias OEM

OEM – Original Equipment Manufacturer / Fabricante de equipos originales, este tipo de licencia es una de las más comunes que existen ya que son las licencias que viene incluidas en los equipos que compramos de manera legal y son las más usadas a equipos cliente. Esta licencia la encontramos, generalmente, con un sticker de hologramas en la torre del equipo.

Licencias Retail

Este tipo de licencia, también conocidas como RTL o FPP (Full Packaged Product) son las licencias que adquirimos en una tienda oficial o a través de la web y es muy útil para aquellos usuarios que requieren menos de cinco licencias.

Este tipo de licencias pueden venir en un DVD o en su defecto en una imagen ISO para realizar su posterior instalación y una de las ventajas es que podemos transferir la licencia a otro equipo sin ningún inconveniente.

Licencias por volumen

Este tipo de licencia es muy útil para grandes empresas y también se denomina GVLK - Global Volume License Key y está enfocado en las empresas de mediano o gran tamaño debido al volumen de licencia (más de 100 usuarios)

Este tipo de licencia es la forma más práctica y rentable de activar y actualizar el software en la organización por diversas razones como:

- Precios
- Flexibilidad
- Soporte

- Estabilidad

2.12.4 Análisis de los Sistemas Operativos a Seleccionar en Terminales y Servidores

Ubuntu Server

Requisitos

Los requisitos para una versión server de Linux son mínimos debido a que no utiliza el entorno gráfico, pero para que actúe como servidor dependiendo del tráfico que tengamos puede requerir más.

Mínimo (consola)

- 256MB de memoria
- 2GB de espacio en HDD (incluido swap)
- AMD o Intel Procesador de 64-32 bits
- Incluido AMD Optaron e Intel EM64T, para versiones de 64

Mínimo (Gráfico)

- 512 MB de memoria
- 4GB de espacio en HDD (incluido swap)
- AMD o Intel Procesador de 64-32 bits
- Tarjeta gráfica VGA, monitor con resolución de 800x600

Gestión de usuarios para Ubuntu

Los usuarios en Ubuntu se identifican por un número único de usuario, User ID, UID. Y pertenecen a un grupo principal de usuario, identificado también por un número único de grupo, Group ID, GID. El usuario puede pertenecer a más grupos además del principal.

Es posible identificar tres tipos de usuarios en Ubuntu:

Usuario Root

- También llamado súper usuario o administrador.
- Su UID (User ID) es 0 (cero).
- Es la única cuenta de usuario con privilegios sobre todo el sistema.
- Acceso total a todos los archivos y directorios con independencia de propietarios y permisos.
- Controla la administración de cuentas de usuarios.
- Ejecuta tareas de mantenimiento del sistema.
- Puede detener el sistema.
- Instala software en el sistema.
- Puede modificar o reconfigurar el kernel, controladores, etc.

Usuarios Especiales

- Ejemplos: bin, daemon, adm, lp, sync, shutdown, mail, operator, squid, apache, etc.
- Se les llama también cuentas del sistema.
- No tiene todos los privilegios del usuario root, pero dependiendo de la cuenta asumen distintos privilegios de root.

- Lo anterior para proteger al sistema de posibles formas de vulnerar la seguridad.
- No tienen contraseñas pues son cuentas que no están diseñadas para iniciar sesiones con ellas.
- También se les conoce como cuentas de "no inicio de sesión" (nologin).
- Se crean (generalmente) automáticamente al momento de la instalación de Linux o de la aplicación.
- Generalmente se les asigna un UID entre 1 y 100.

Usuarios normales

- Se usan para usuarios individuales.
- Cada usuario dispone de un directorio de trabajo, ubicado generalmente en /home.
- Cada usuario puede personalizar su entorno de trabajo.
- Tienen solo privilegios completos en su directorio de trabajo o HOME.
- Por seguridad, es siempre mejor trabajar como un usuario normal en vez del usuario root, y cuando se requiera hacer uso de comandos solo de root, utilizar el comando su.
- En las distros actuales de Linux se les asigna generalmente un UID superior a 500.

Soporte

Ubuntu no sólo soporta drivers y componentes compatibles con Gnu/Linux, sino que también da soporte a drivers y software privativo por lo que la gama de compatibilidad se amplía. Aun así es bueno consultar esta base de datos porque nos puede ayudar a elegir el componente ideal a la hora de construir nuestro ordenador e incluso ayudarnos en caso de haber algún problema con el hardware o con las actualizaciones.

Las versiones no-LTS se liberan cada 6 meses, y Canonical proporciona soporte técnico y actualizaciones de seguridad durante 9 meses a partir de la versión 13.04 hacia adelante, anteriormente eran 18 meses de soporte para las versiones no-LTS. Las versiones LTS (Long Term Support) ofrecen un soporte técnico de 5 años para la versión de escritorio y servidor, a partir de la fecha del lanzamiento.

Versión	Nombre en clave	Lanzamiento	Fin de soporte
4.10	Warty Warthog	20/octubre/2004	30/abril/2006
5.04	Hoary Hedgehog	8/abril/2005	31/octubre/2006
5.10	Breezy Badger	13/octubre/2005	13/abril/2006
6.06 LTS	Dapper Drake	1/junio/2006	14/julio/2009 (escritorio) 1/junio/2011 (servidor)
6.10	Edgy Eft	26/octubre/2006	25/abril/2008
7.04	Feisty Fawn	19/abril/2007	19/octubre/2008
7.10	Gutsy Gibbon	18/octubre/2007	18/abril/2009
8.04 LTS	Hardy Heron	24/abril/2008	12/mayo/2011 (escritorio) 9/mayo/2013 (servidor)
8.10	Intrepid Ibex	30/octubre/2008	30/abril/2010
9.04	Jaunty Jackalope	23/abril/2009	23/octubre/2010
9.10	Karmic Koala	29/octubre/2009	30/abril/2011
10.04 LTS	Lucid Lynx	29/abril/2010	9/mayo/2013 (escritorio) 15/abril/2015 (servidor)
10.10	Maverick Meerkat	10/octubre/2010	10/abril/2012
11.04	Natty Narwhal	28/abril/2011	28/octubre/2012
11.10	Oneiric Ocelot	13/octubre/2011	9/mayo/2013
12.04 LTS	Precise Pangolin	26/abril/2012	abril/2017
12.10	Quantal Quetzal	18/octubre/2012	abril/2014
13.04	Raring Ringtail	25/abril/2013	enero/2014
13.10	Saucy Salamander	17/octubre/2013	julio/2014
14.04 LTS	Trusty Tahr	17/abril/2014	abril/2019
14.10	Utopic Unicorn	23/octubre/2014	abril/2015
Color	Significado		
Rojo	Versión ya sin soporte		
Amarillo	Solo versión de servidor con soporte		
Verde	Versión con soporte (escritorio y servidor)		
Azul	Versión en desarrollo		

Utilidades de Ubuntu server

Ubuntu Server es un Sistema Operativo sin entorno gráfico (aunque podemos instalarlo) lo que quiere decir que todas las acciones se realizan mediante consola, y normalmente ni siquiera a través del propio servidor, sino desde una conexión remota. El manejo de Ubuntu Server es muy similar al de cualquier otro Sistema Linux, pero con las particularidades de Ubuntu (como el sudo).

Ubuntu Server es una variante de Ubuntu que sale con cada versión y está dedicada especialmente para su uso en servidores. El uso de Ubuntu como servidor se ha extendido mucho en los últimos años, sobre todo para el uso de servidores web, de un modo tanto particular como profesional.

Gestión de Red

Cuando instalamos Ubuntu Server, éste detecta automáticamente la configuración de la red si disponemos de un servidor DHCP. Si no es así, nos pide que escribamos la IP para nuestro equipo, la máscara de red, la puerta de enlace y la dirección de un servidor DNS y la configuración de red se almacena en el archivo

/etc/network/interfaces por lo que tendremos que editar este fichero para hacer cualquier modificación.

Compatibilidad con Aplicaciones

A la hora de valorar la posibilidad de pasarse a Ubuntu, muchos usuarios tienen dudas sobre la incompatibilidad de sus formatos y programas. Tanto Windows como OS X tienen formatos que no pueden abrirse en Ubuntu, como .exe.

Muchos de ellos pueden instalarse desde el Centro de Software de Ubuntu, mientras que otros requieren de algún código introducido en la terminal. También existen tiendas de aplicaciones para Ubuntu en las que podrás encontrar todo tipo de alternativas OpenSource, con funciones variadas. AppGrid es la más conocida. Son programas que imitan a los que ya existen en otros sistemas operativos, que ofrecen nuevas opciones o que directamente permiten emular a un SO rival. Sin embargo, algunas acciones que requieren programas de terceros en otros sistemas, vienen completamente integradas en Ubuntu. Como por ejemplo es el gestor de archivos, alternativa a WinRAR, y Chromium, que es la alternativa de Google

Ubuntu Desktop 18.04 LTS

Requisitos mínimos para instalar Ubuntu 18.04 LTS

- 2 GB de RAM
- Procesador de doble núcleo (2 GHz)
- 25 GB de espacio libre en disco duro
- Medios de instalación (DVD o USB)
- Recomendada la conectividad a Internet para instalar software terceros y actualizaciones durante la instalación

Soporte

Las versiones LTS de Ubuntu ofrecen cinco años de soporte, pero solo en la edición para servidores. En escritorio son tres años, si bien los dos restantes se siguen recibiendo actualizaciones de seguridad relacionadas con el kernel y el software base del sistema.

Asimismo, las versiones LTS incluyen desde el año pasado la posibilidad de adherirse al Extended Security Maintenance (ESM) o mantenimiento de seguridad extendido, un nuevo servicio de pago que proporciona actualizaciones de seguridad durante al menos un año más.

Gestión de Red

La configuración de red en Ubuntu 18.04 ha cambiado. `/etc/network/interfaces` ya no es el archivo que se utiliza para configurar tus interfaces de red. La herramienta que debemos usar es `netplan`.

Esta herramienta utiliza archivos de configuración en `/etc/netplan`, ahí podrás encontrar el archivo en formato `yaml`. Si no hay ningún archivo, o necesitas generar uno nuevo, puedes hacerlo con: `netplan generate`.

Requisitos de Windows Server 2012

- Procesador: 1,4 GHz de procesador x64.
- Memoria RAM: 512 MB.
- Espacio en disco: 32GB para una instalación correcta. ...
- Unidad DVD.
- Pantalla Súper VGA (800 x 600) o de mayor resolución.
- Acceso a Internet para activación de la licencia.
- Teclado y mouse de Microsoft® (u otro dispositivo señalador compatible).

Gestión de usuarios

- *Administrador*: Tiene control total sobre el dominio y no se podrá eliminar ni retirar del grupo Administradores (aunque sí podemos cambiarle el nombre o deshabilitarla).
- *Invitado*: Está deshabilitada de forma predeterminada y, aunque no se recomienda, puede habilitarse, por ejemplo, para permitir el acceso a los usuarios que aún no tienen cuenta en el sistema o que la tienen deshabilitada. De forma predeterminada no requiere contraseña, aunque esta característica, como cualquier otra, puede ser modificada por el administrador.
- *Asistente de ayuda*: se utiliza para iniciar sesiones de Asistencia remota y tiene acceso limitado al equipo. Se crea automáticamente cuando se solicita una sesión de asistencia remota y se elimina cuando dejan de existir solicitudes de asistencia pendientes de satisfacer.

Soporte

Se han ampliado las fechas del soporte estándar y el soporte extendido de Windows Server 2012 para los clientes, con el propósito de alinearlas con la escala de tiempo de transición del ciclo de vida estándar.

La directiva de ciclo de vida de Windows Server 2012 establece que se proporcionará soporte estándar durante cinco años o durante dos años después del lanzamiento del producto sucesor (N+1, donde N = versión del producto), lo que suponga más tiempo de soporte. Microsoft también proporcionará soporte extendido durante los cinco años posteriores al soporte estándar o durante los dos años posteriores al lanzamiento del segundo producto sucesor (N+2), lo que suponga más tiempo de soporte.

Ventajas y Desventajas

Ventajas

- Seguridad avanzada
- Arquitectura confiable
- Rendimiento mejorado
- Fácil de usar
- Ahorro de espacio
- Múltiples servidores
- Mayor grado de acceso

Desventajas

- Costo alto
- No hay puente intermedio
- Sin opción de virtualización
- Los mensajes tienden a fallar
- Requiere altos conocimientos

Versiones

- *Windows server 2012 R2*: Es actualmente la versión de Windows Server más moderna y mejorada. Las ediciones de Windows Server 2012 R2 siguen siendo optimizadas y simplificadas para que los administradores puedan elegir más fácilmente la edición que más les convenga de acuerdo con sus necesidades, a continuación, una descripción general del enfoque de cada versión.
- *Windows server 2012 R2 Datacenter*: Para un entorno altamente virtualizado que requiera características de alta disponibilidad, incluida la agrupación en clústeres.
- *Windows server 2012 R2 Standard*: Para un entorno no virtualizado o poco virtualizado en el que se desee incluir características de alta disponibilidad, incluida la agrupación en clústeres.
- *Windows server 2012 R2 Essentials*: Para pequeñas empresas con hasta 25 usuarios, especialmente aquellas empresas que quieran implementar su primer servidor.
- *Windows server 2012 R2 Foundation*: Para pequeñas empresas con hasta 15 usuarios (solo disponible a través de partners OEM directos).

Las licencias son un mecanismo que es utilizado para activar las funcionalidades específicas de un software servidor, Windows Server ofrece distintas licencias en cada una de sus versiones que se ajustan a las necesidades de cada empresa o usuario como se muestran en la siguiente tabla.

Windows 7

Requisitos del sistema de Windows 7

- Procesador de 1 gigahercio (GHz) o más rápido de 32 bits (x86) o de 64 bits (x64)
- 1 GB de RAM (32 bits) o 2 GB de RAM (64 bits)
- 16 GB de espacio disponible en el disco duro (32 bits) o 20 GB (64 bits)
- Tarjeta gráfica DirectX 9 con controlador WDDM 1.0 o superior.

Gestión de usuarios

El sistema operativo Windows 7 distingue dos tipos de usuarios:

- Estándar: Con limitaciones en el acceso.
- Administrador: Sin ningún tipo de restricción.

Los grupos permiten simplificar la administración:

- Los usuarios Estándar pertenecen al grupo Usuarios.
- El usuario Administrador pertenece al grupo Administradores.

Durante la instalación se crean por defecto tres cuentas de usuario: Usuario, Administrador, Invitado.

Edición	Aplicado para	Comparación de las características	Modelo de licencias
Datacenter	Entornos altamente virtualizados	Todas las funciones de Windows Server con instancias virtuales ilimitadas	Procesador + CAL
Standard	Entornos de baja densidad o no virtualizados	Todas las funciones de Windows Server con dos instancias virtuales	Procesador + CAL
Essentials	Entornos de pequeña empresa para servidores con dos procesadores como máximo	Interfaz más sencilla, conectividad pre-configurada con servicios basados en la nube; una instancia virtual de Essentials	Server (límite de 25 usuarios)
Foundation	Servidor de uso general económico con un único procesador	Funcionalidad de servidor de uso general sin derechos de virtualización	Server (límite de 15 usuarios)

Compatibilidad

Las versiones cliente de Windows 7 se lanzaron en versiones para arquitectura 32 bits y 64 bits en las ediciones Home Basic, Home Premium, Professional y Ultimate.

No obstante, las versiones servidor de este producto fueron lanzadas exclusivamente para arquitectura 64 bits.

Esto significa que las versiones cliente de 32 bits aún soportan programas Windows 16 bits y MS-DOS. Y las versiones 64 bits (incluidas todas las versiones de servidor) soportan programas tanto de 32 como de 64 bits.

2.12.5 Estudio de Alternativas

Aplicación de Escritorio

Una aplicación de escritorio es aquella que se encuentra instalado en el ordenador o sistema de almacenamiento (USB) y podemos ejecutarlo sin internet en nuestro sistema operativo, al contrario que las aplicaciones en la nube que se encuentran en otro ordenador (servidor) al que accedemos a través de la red o internet a su software.

Aplicación Web

En la ingeniería de software se denomina aplicación web a aquellas herramientas que los usuarios pueden utilizar accediendo a un servidor web a través de internet o de una intranet mediante un navegador. En otras palabras, es un programa que se codifica en un lenguaje interpretable por los navegadores web en la que se confía la ejecución al 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. Existen aplicaciones como los correos web, wikis, blogs, tiendas en línea y la propia Wikipedia que son ejemplos bastante conocidos de aplicaciones web.

Aplicación Mixta o Híbrida

Es una combinación de las dos anteriores, se podría decir que recoge lo mejor de cada una de ellas. Las apps híbridas se desarrollan con lenguajes propios de la

web app, es decir, HTML, JAVASCRIPT, PHP y CSS por lo que permite su uso en diferentes plataformas, pero también dan la posibilidad de acceder a gran parte de las características del hardware del dispositivo.

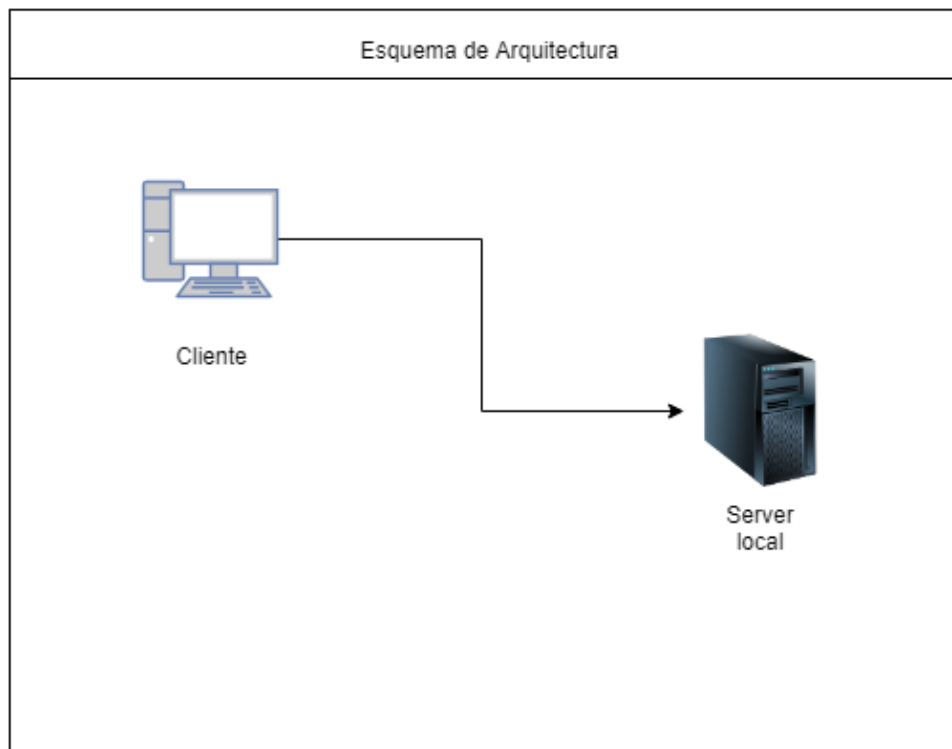
Aplicación Nativa

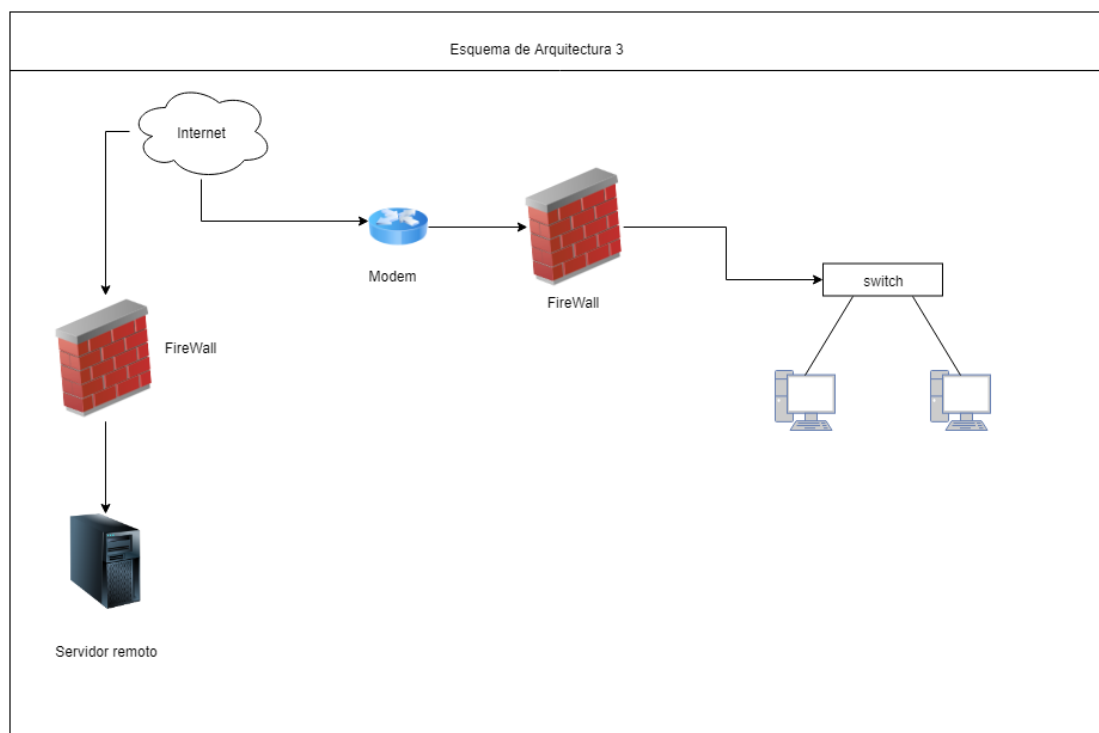
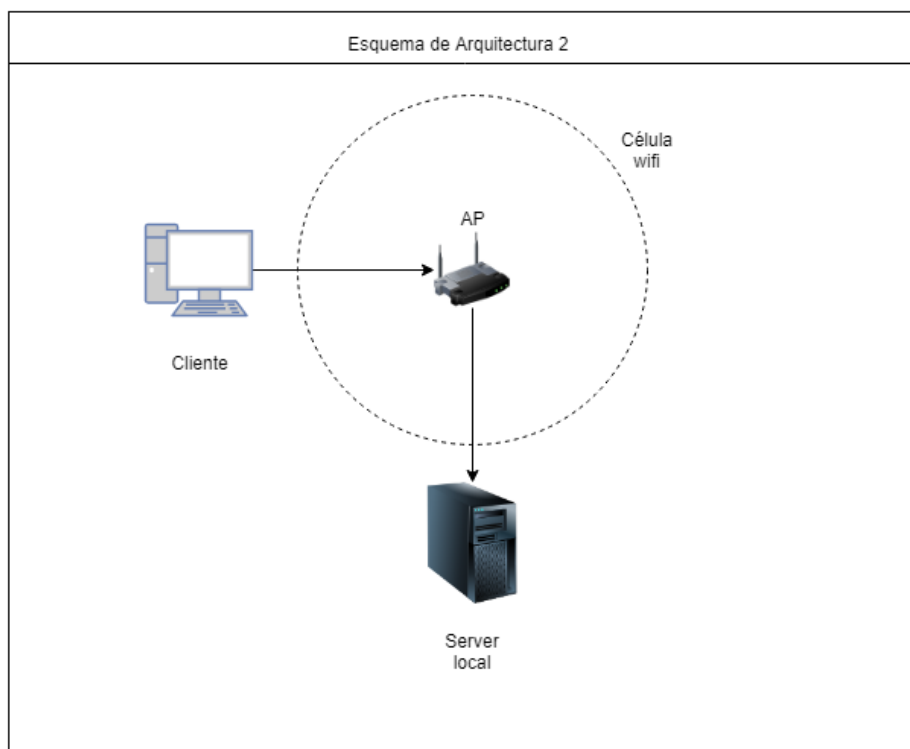
La aplicación nativa está desarrollada y optimizada específicamente para el sistema operativo determinado y la plataforma de desarrollo del fabricante (Android, iOS, etc).

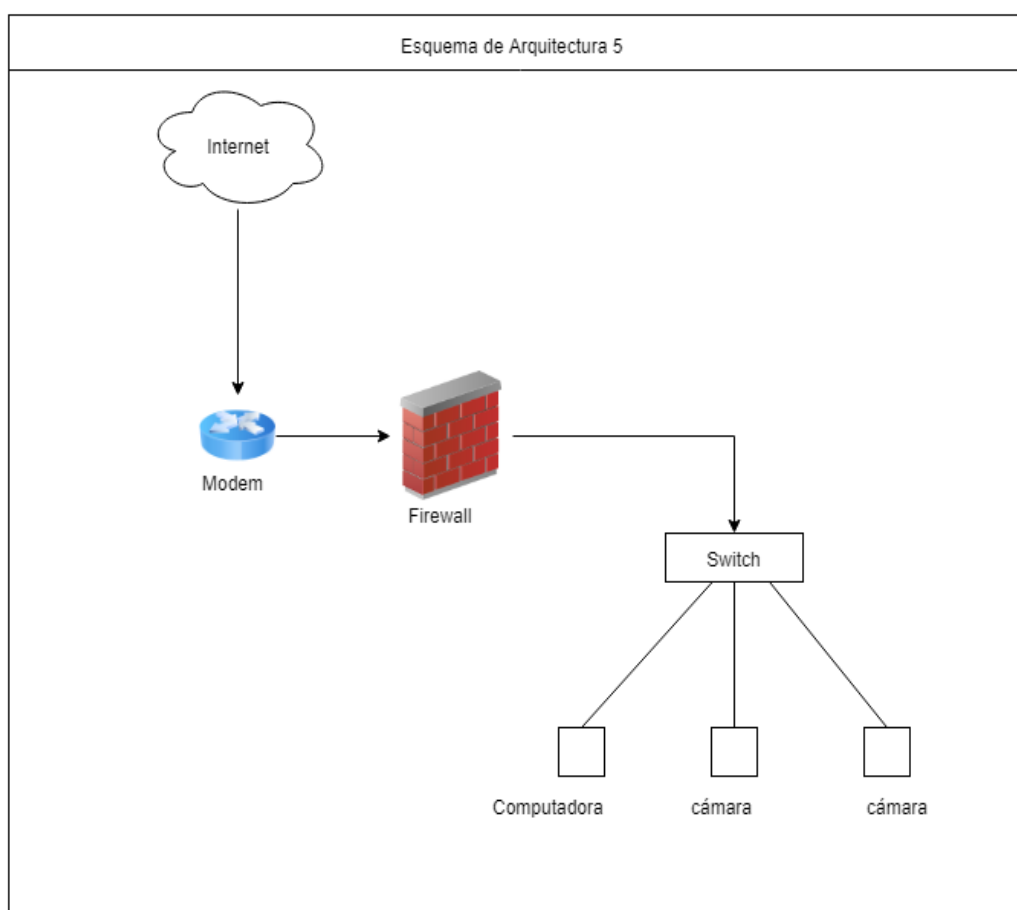
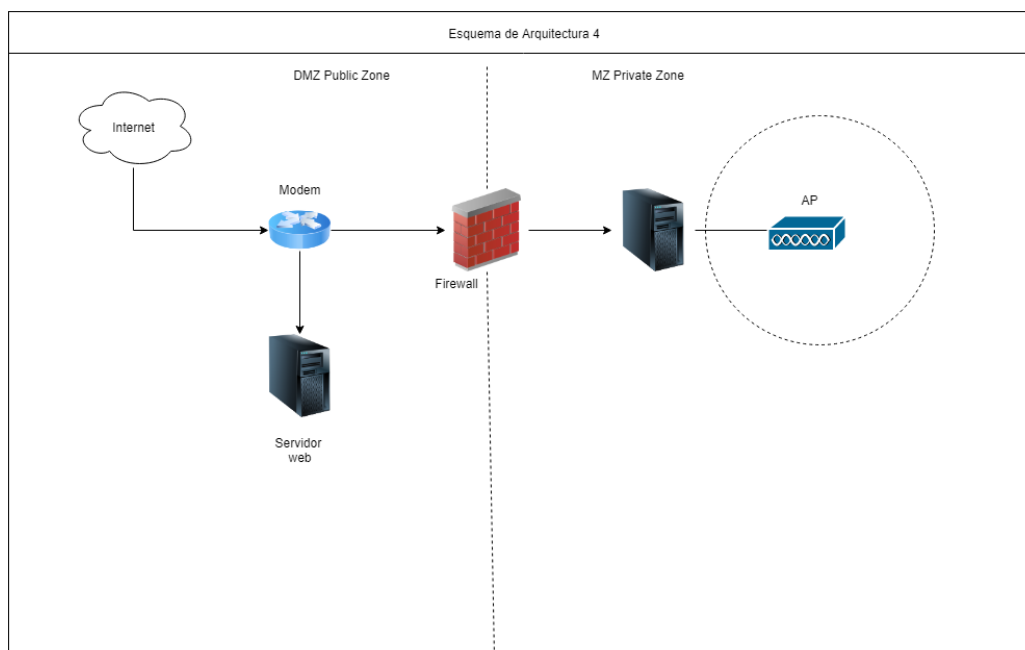
Este tipo de aplicaciones se adapta al 100% con las funcionalidades y características del dispositivo obteniendo así una mejor experiencia de uso. Sin embargo, el desarrollo de una aplicación nativo comporta un mayor coste, puesto que si se desea realizar una aplicación multiplataforma se ha de realizar una nueva versión para cada sistema operativo, multiplicando así los costes de desarrollo.

Algunos ejemplos de aplicación nativa, serían WhatsApp o Facebook.

2.12.6 Esquemas de Arquitectura

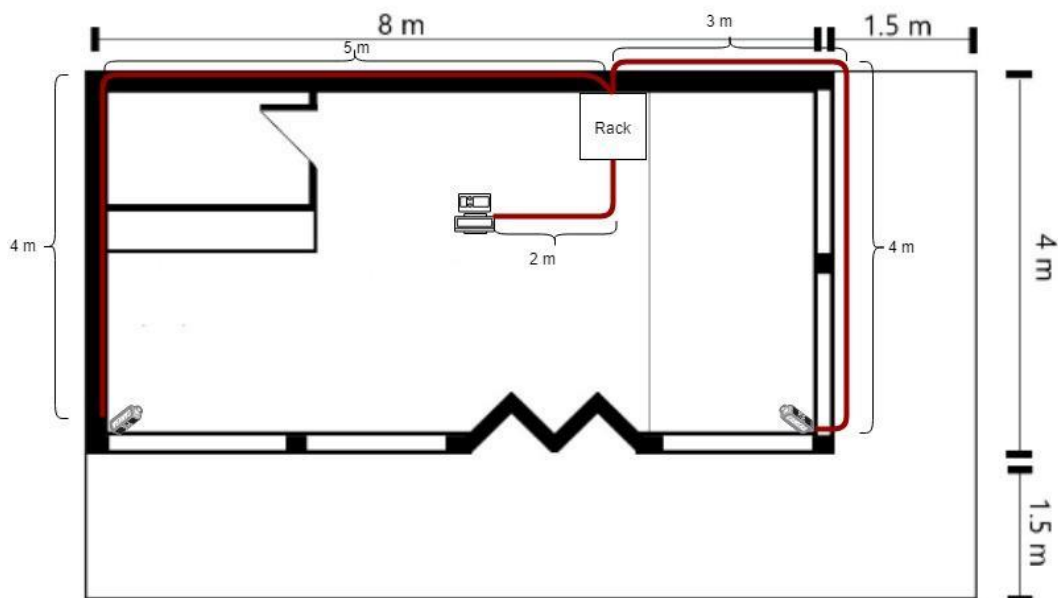




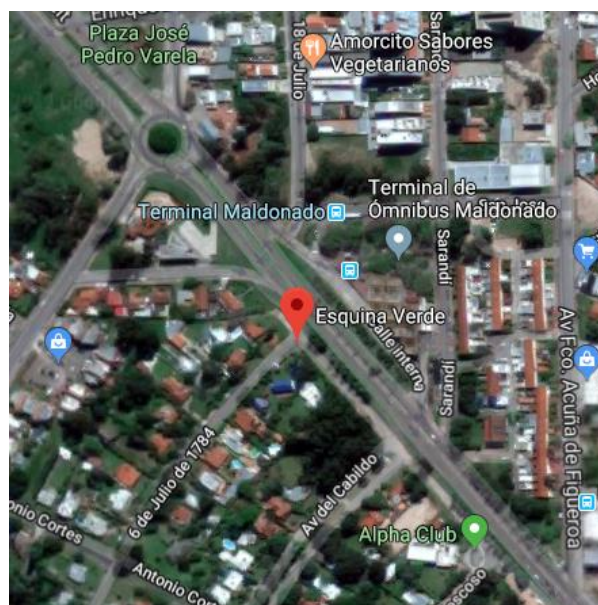


2.12.7 Diseño de Cableado Estructurado

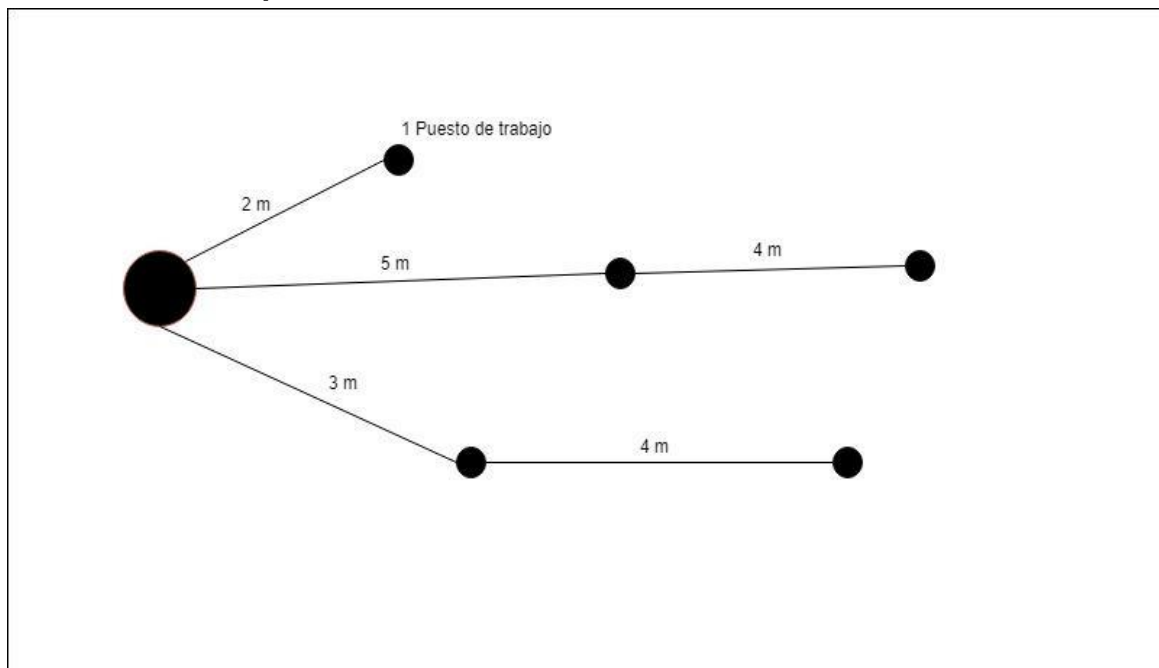
Plano de la planta de trabajo y canalizaciones.



Ubicación del vivero



2.12.8 Esquema de Ramales



2.12.9 Selección de placa de control y alimentación

Placa Arduino Leonardo R3

El Arduino Leonardo es una placa de desarrollo basada en el microcontrolador ATmega32u4. Tiene 20 pines de entrada/salida digital (7 de ellos pueden ser usados como salida PWM), un cristal de 16Mhz, conexión micro USB, conector DC de alimentación, conector ICSP y botón de reseteo.

La placa integra todo lo necesario para que el microcontrolador opere, solo necesita conectar al computador por el cable USB o a un transformador de AC-DC.

Alimentación de la placa Arduino

Para alimentar Arduino podemos:

- Aplicar 6-12V en la clavija jack que disponen Arduino UNO, Mega, entre otros modelos

- Aplicar 6-12V entre el pin GND y el pin RAW (pin Vin en Arduino Mini)
- Alimentar mediante USB
- Aplicar 5V (regulados y estables) en el pin 5V (3,3V en ciertos modelos)



3 Anexos

3.1 Equipo

3.1.1 Reglamento del equipo

- 1) Asistencia obligatoria a clase en asignaturas relacionadas al proyecto (SBD, SO, Taller, Prog, ADA, Proyecto, Formación Empresarial).
 - a. Justificar falta.
- 2) Cumplimiento de tareas designadas en reuniones.
 - a. Cumplimiento en tiempo designado.
 - b. Documentación de las mismas.
- 3) Reuniones Obligatorias.
 - a. Preferentemente presenciales.
 - b. De 1 a 3 reuniones semanales.
 - c. Asistencia obligatoria (faltas solo con justificativo probado y justificación/aviso de llegada tarde).
 - d. La reunión comienza hasta 15 minutos luego de la hora estipulada con o sin todos los participantes.
 - e. Bitácora escrita obligatoria (con fecha, hora, asistentes y detalle de trabajo realizado).
 - f. Mínimo de una hora por reunión.
- 4) Respaldo de datos de documentación y software
 - a. Cada miembro del equipo deberá contar con una copia.
 - b. Además, copia en la nube con la última versión.
- 5) Todos deben tener conocimiento y criterio sobre lo que suceda con el proyecto.
- 6) Costos del proyecto se dividen por igual.
- 7) Se limita la información que se pueda brindar a grupos ajenos. La ayuda que se pueda brindar debe de ser respecto a formas de trabajar o consejos.
- 8) No se permite el generar pleitos o peleas innecesarias.

Sanciones

1. Por inasistencias.
 - a. Cada falta injustificada castigo apropiado puesto por el resto del equipo.
 - b. Cada 4 falta injustificada en una materia debe cumplir tareas extras.
 - c. Falta una semana completa (sin justificación probada) moción a expulsión.
2. Por incumplimiento de tareas.
 - a. Si no se cumplen las tareas propuestas en un tiempo pautado (sin justificación) debe de cumplir con el doble de tareas en el mismo periodo de tiempo. Si es justificado se distribuye entre aquellos miembros hábiles.

- b. Si no se cumplen estas tareas, se distribuyen y se plantea castigo apropiado del momento.
 - c. El acumulo o falta de continuidad en la entrega de tareas permite votación a expulsión.
- 3. Inasistencia a reuniones.
 - a. Una falta injustificada a una reunión será castigada por el equipo de forma apropiada en el momento.
 - b. Falta a tres o más reuniones sin justificación permite al equipo llevar a cabo moción a expulsión.
- 4. Si un miembro pierde la documentación o la borra de forma accidental, deberá de volver a escribirla por completo (o lo que se haya perdido).

Expulsión

Sólo será llevada a cabo la expulsión de un miembro si este no presenta en las reuniones y clase lo suficiente (sin justificación), al igual que no aportar nada a la misma. La votación se llevará a cabo por los miembros del equipo y se desarrollará de forma democrática, una vez se llegue a una decisión se debe consultar/avisar a los profesores.

3.1.2 Desarrollo de las actividades- Acta de Reunión

Martes 07/05/19 (13:41 - 15:00)

La reunión propuesta a las 13.30 comenzó a las 13.41 dentro del intervalo de espera a los miembros ausentes, dos de ellos no asistieron (Germán) o llegaron tarde (Jordan. Ambos con aviso de llegada tarde, no de ausencia).

Se discutieron las sanciones y reglamento del equipo, dejando lugar a que los miembros ausentes (Germán y Jordan quién llegó tarde) aportarán en caso de encontrar nuevas reglas a futuro en la próxima reunión.

También se volvieron a discutir las posiciones de los miembros en el equipo como la posibilidad de quitar miembros no eficientes, modificando el reglamento conforma a eliminar aquellos miembros que fuesen ineficientes o que no diesen beneficio al equipo de forma alguna.

Se tomó tiempo para discutir con el cliente la fecha de la primera entrevista al igual que se arregló y optimizó la documentación incompleta, es decir que agregaron preguntas al documento de las preguntas a realizar en la entrevista.

La próxima reunión pautada para el sábado 11/05 y entrevista pautada para el 10/05.

Viernes 10/05/19 (15:30 - 16:00)

Primera entrevista con el cliente, con tres miembros presentes (Jun, Jordan, Martina) y un miembro ausente (Germán).

Se consiguieron planos, información del cliente y permiso para grabar la entrevista, al igual que se planteó el modo de trabajo que se tendrá a lo largo del año y aclarar los límites.

Con el cliente se recopilaron datos de la forma de trabajo que se lleva en el local, empleados, funcionamiento y tipos de registros que se hacen, concluyendo en que debido a la falta de registros y datos

En clase se estableció la gestión de riesgos basados en aquellos hechos en la clase anterior, con los mismos miembros presentes.

Sábado 11/05/19 (16:41 - 19:00)

Se comenzó la reunión a las 16:41 con dos miembros presentes (Jun, Martina) y dos miembros con falta, una justificada y avisada (Jordan) y otra injustificada y sin aviso (Germán).

Comienzo de la documentación oficial, entre ello: Establecer nombre de la empresa, la fundamentación de la misma y la descripción del cliente además de iniciar con la descripción de procesos y establecer una norma media de cómo describirlo.

Se planteó el expulsar a Germán del equipo, dando una última oportunidad de presentarse a una reunión antes de discutir oficialmente el expulsarlo.

Se dividieron las actividades a realizar para la entrega, la numeración de los procesos, en busca de que todo estuviese completo para la fecha.

Domingo 26/05/19 (16:23 – 18:34)

Se comenzó la reunión a las 16:23 con tres miembros presentes.

Se presentaron las tareas asignadas previamente y se visaron si estaban o no hechas correctamente, explicando temas para los miembros del equipo que no habían entendido las consignas o temas que se encontrasen inseguros respecto al proyecto.

Se dividieron las tareas para la realizar a la próxima entrega, acordando el entregar las partes por forma separada.

Jueves 30/05/19 (22:35 – 00:45)

Se realizó la reunión a las 22:36 con dos de los miembros (Jun, Martina) por medio de la aplicación Discord en la que se terminó de corregir el documento previo a la entrega, agregando partes que faltaban agregar al documento y últimas correcciones.

3.1.3 Análisis de riesgos

Identificación de Riesgos

Nº	Riesgo	Categoría	Ocurrencia	Incidencia
1	Pérdida de Cliente.	Proyecto	Media	Alta
2	Pérdida de Miembros.	Proyecto	Media	Media
3	Enfermedad.	Proyecto	Media	Baja
4	Información Incorrecta.	Proyecto	Baja	Media
5	Pérdida de Información.	Software	Media	Alta
6	Ausencia de Miembros.	Proyecto	Baja	Media

7	Mala relación con el cliente.	Proyecto	Media	Alta
8	No llevar constancia de actualizaciones.	Proyecto	Media	Media
9	No ponerse de acuerdo.	Proyecto	Baja	Media
10	Tardar en conseguir materiales electrónicos.	Taller	Media	Alta
11	Tardar en terminar la base de datos.	Base de Datos, Software	Media	Alta
12	Mal gestión de la base de datos.	Base de Datos	Media	Alta
13	No tener presupuesto para lo planificado.	Taller	Media	Alta

Gestión de Riesgos

Nº	Disminución	Anulación	Plan de Contingencia
1	Mantener una buena relación y estar seguro del nivel de compromiso al proyecto.	Asegurarse de que el cliente realmente no quiere colaborar en busca de encontrar una solución sin que se pierda.	-Contactar todos los viveros posibles en busca de un nuevo cliente. -Solucionar cualquier descontento que haya llevado al cliente a querer alejarse (si existe).
2	Mantener una buena relación y estar seguro del nivel de compromiso al proyecto.	Este riesgo no puede ser anulado.	Redistribuir las tareas y cargos de forma permanente buscando el menor impacto a la fecha de entrega-
3	Este riesgo no se puede evitar.	Asegurarse de si el miembro puede trabajar a pesar de estar enfermo.	-Redistribuir las tareas temporalmente. -Reunirse de forma remota o que el miembro trabaje

			desde su casa.
4	Formular preguntas concretas y apropiadas al cliente.	Este riesgo no puede ser anulado.	Volver a contactar/entrevistar lo más rápido posible en busca de la información correcta.
5	Mantener copias seguras y nunca una única copia.	Este riesgo no puede ser anulado.	Reescribir lo perdido lo antes posible.
6	Este riesgo no se puede evitar.	Avisar de la sanción al miembro y distribuir tareas.	-Si la falta se justifica se retira la sanción y se vuelven a distribuir las tareas.
7	Mantener una buena relación y evitar los conflictos a toda costa.	Este riesgo no puede ser anulado.	Solucionar cualquier descontento que haya llevado al cliente a sentirse incómodo/molesto.
8	Utilizar un programa que mantenga constancia de los cambios.	Este riesgo no puede ser anulado.	Si se edita la misma información, optar por la más eficiente.
9	Votaciones democráticas.	Si no se llega a una decisión, se exponen ambos puntos frente a un tercero neutral que permita llegar a un acuerdo.	Se deberá replantear las decisiones tomadas y elegir la más coherente.
10	Este riesgo no se puede evitar.	Este riesgo no puede ser anulado.	Avanzar todo lo posible en aquello que no requiera el dispositivo necesario.
11	Ser eficiente y asignar correctamente las tareas, siendo aquél más hábil con el cargo de responsabilidad sobre la Base de Datos	Este riesgo no puede ser anulado.	Distribuir las tareas para completar lo necesario lo más rápido posible.
12	Este riesgo no se puede evitar.	Volver a revisar en busca de errores y soluciones.	Sustituir a quien se haya encargado de mayor parte del error o mala gestión de la Base de Datos.
13	Ahorrar y distribuir.	Este riesgo no puede ser anulado.	Este riesgo no tiene plan de contingencia.

3.1.4 Bitácora Actualizada

Fecha	Hora	Presentes	Detalles
11/04/19	16:56 - 17:50	Jordan, Jun, Martina.	Comienzo de documentación, establecer preguntas de la entrevista al cliente.
26/04/19	16:35 -17:50	Jordan, Jun, Martina.	Establecer análisis riesgos del proyecto.
07/05/19	13:41 -15:00	Martina, Jun. (Llegada tarde c/ aviso: Jordan)	Establecer reglamento (sanciones), y fecha de entrevista con el cliente. Distribución de tareas y posiciones.
10/05/19	15:30 - 16:00 / 16:15 - 17:50	Jordan, Jun, Martina.	Entrevista con el cliente, establecer gestión de riesgos.
11/05/19	16:40 - 19:00	Jun, Martina.	Comienzo oficial de la documentación, delegación de tareas.
26/05/19	16:23 – 18:34	Jun, Jordan, Martina.	Delegación de tareas, edición y corrección de las tareas hechas
30/05/19	22:35 – 00:45	Jun, Martina.	Última edición antes de la pre-entrega.