**UNIVERSIDAD TÉCNICA NACIONAL**

**SEDE SAN CARLOS**

**CARRERA**

**Ingeniería del Software**

**Documentación del Proyecto Final del curso Ingeniería en Software**

**ECOMASTER**

**ELABORADO POR**

Kevin Arias Arce

Roque Chacón Corrales

2018.

**Contenido**

[Índice de figuras 3](#_Toc511839416)

[Índice de Tablas 4](#_Toc511839417)

[Descripción del problema 5](#_Toc511839418)

[Justificación del problema. 6](#_Toc511839419)

[Objetivo general. 7](#_Toc511839420)

[Objetivos específicos. 7](#_Toc511839421)

[Herramientas y técnicas que se utilizan 8](#_Toc511839422)

[GIT: 9](#_Toc511839423)

[Requerimientos funcionales, no funcionales y del sistema. 12](#_Toc511839424)

[Solución del problema. 26](#_Toc511839425)

[Diagramas 28](#_Toc511839426)

[Criterios de aceptación. 32](#_Toc511839427)

[Resultados de las pruebas 42](#_Toc511839428)

[Conclusiones. 43](#_Toc511839429)

[Recomendaciones. 43](#_Toc511839430)

[Información del cliente 44](#_Toc511839431)

[Anexos 45](#_Toc511839432)

[Plan de contingencia 46](#_Toc511839433)

# **Índice de figuras**

Figura 1: Logo de PHP…………………………………………………………………….8

Figura 2: Logo de Laravel…………………………………………………………………9

Figura 3: Logo Git…………………………………..……………………………………..10

Figura 4: Logo MySQL…………………………………………..………………………..10

Figura 5: Estructura del patrón de diseño MVC…………………………………...………11

Figura 6: Diagrama de casos de uso………………………………………………28

Figura 7: Diagrama de clases……………………………………………………...29

Figura 8: Diagrama de base de datos……………………………………………..30

Figura 9: Diagrama de Arquitectura………………………………………………31

Índice de Tablas

[Tabla 1: Requisito Funcional-001 – Agregar Usuario 12](#_Toc490500185)

[Tabla 2: Requisito Funcional-002 – Editar Usuario](#_Toc490500185) 13

[Tabla 3: Requisito Funcional-003 – Eliminar Usuario](#_Toc490500186) 14

[Tabla 4: Requisito Funcional-004 – Ver Usuarios](#_Toc490500187) 15

[Tabla 5: Requisito Funcional-005 – Agregar Supermercado](#_Toc490500188) 16

[Tabla 6: Requisito Funcional-006 – Editar Supermercado](#_Toc490500189) 17

[Tabla 7: Requisito Funcional-007 – Eliminar Supermercados](#_Toc490500190) 18

[Tabla 8: Requisito Funcional-008 – Ver Supermercados](#_Toc490500191) 19

[Tabla 9: Requisito Funcional-009 – Crear de Horario](#_Toc490500192) 20

[Tabla 10: Requisito Funcional-010 – Editar Horario](#_Toc490500193) 21

[Tabla 11: Requisito Funcional-011 – Eliminar Horarios](#_Toc490500194) 22

[Tabla 12: Requisito Funcional-012 – Ver Horarios](#_Toc490500195) 23

[Tabla 13: Requisito Funcional-013 – Login](#_Toc490500196) 24

Tabla 14: Requisito Funcional-014 – Reportes…………………………………………….25 Tabla 15: Caso de prueba Inicio de Sesión…..…………………………………………….31 Tabla 16: Caso de prueba Agregar…………...…………………………………………….32 Tabla 17: Caso de prueba Editar...…………...…………………………………………….33 Tabla 18: Caso de prueba Eliminar...………...…………………………………………….34 Tabla 19: Caso de prueba Detalles…………...…………………………………………….35 Tabla 20: Caso de prueba Crear Horarios.…...…………………………………………….36 Tabla 21: Caso de prueba Editar Horarios.….……………………….…………………….37 Tabla 22: Caso de prueba Detalles Horarios.….…….…….……………………………….38 Tabla 23: Caso de prueba Eliminar Horarios....….…………………………..…………….39 Tabla 24: Caso de prueba Reportes.………….…………………………………………….40

**Descripción del problema.**

La cadena de supermercados EconoMás tiene varias sedes ubicadas en la provincia de Alajuela, distribuidas entre los cantones de Pital, Viento Fresco y dos más ubicadas en Aguas Zarcas, en una de estas dos últimas sedes la compañía cuenta con la oficina de administración, desde ese punto, la empresa hace la toma de decisiones.

Entre el personal administrativo, está la Supervisora y la encargada de Gestión de talento humano, esos dos puestos, están a cargo de la creación y administración de horarios, para todo el personal de la cadena, esto se refiere a que cada empleado sea asignado en un tiempo específico a un rol dentro de una de las sedes de la empresa.

La creación de los horarios lo realizan una vez por semana y es uno de los problemas que tienen en la parte administrativa, ya que dichos horarios se realizan en Excel y están propensos a muchos fallos, por ejemplo, más de 48 horas semanales por empleado establecidas por el código de trabajo en el artículo 136, aparecer en dos horarios de diferentes puntos de venta a la misma, dos días libres por empleado en una semana, y muchos fallos que se pueden cometer.

# **Justificación del problema.**

El déficit de poder crear horarios más rápido y de una manera más sencilla y cómoda ha perjudicado en tiempo de toma de decisiones a la empresa Ecónomas, tanto que en momentos se ha tenido problemas en choques de horario de un determinado empleado, lo que genera una ineficiencia de los trabajadores.

El problema es la lentitud por la complejidad que lleva asignar un horario a un empleado, problema se agrava con el tiempo de espera que requiere la distribución de estos horarios a todo el personal ya que se hace uno a uno vía mensaje de texto, muchas veces pueden tardar hasta el sábado para ser enviado el horario a los empleados de la siguiente semana, esto perjudica a empleados y también a la parte administrativa.

Por eso se tomó la decisión de realizar una aplicación que pueda satisfacer todas las necesidades de esta empresa a la hora de la creación de horarios, para que así pueden mejorar la calidad de atención al colaborador y así mismo a los clientes.

# **Objetivo general.**

Desarrollar una aplicación web para el mejoramiento de la gestión de horarios, para la empresa ECONOMAS.

# **Objetivos específicos.**

1. Determinar cuáles son los factores, que se toman en cuenta para la creación de un horario.
2. Diseñar una interfaz amigable, para brindarle al usuario una usabilidad sencilla.
3. Construir reportes, para la administración de los horarios, para toma de decisiones de la empresa EconoMás.

**Herramientas y técnicas que se utilizan.**

Se usará el patrón MVC, trabajando con este patrón se facilitará el entendimiento del código, ya que separa el mismo en tres capas: MODELOS, VISTAS y CONTROLADORES, esto favorece la calidad de la aplicación y permitiría la escalabilidad a futuro.

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.

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



Figura 1: Logo de PHP

**PHP es un lenguaje totalmente libre y abierto**. Tiene una curva de aprendizaje muy baja, su sintaxis es simple y cumple estándares básicos de la programación orientada a objetos. No son necesarios complejos entornos de desarrollo, que incluso necesitan su propio periodo de aprendizaje. Puedes programar en PHP sin más ayuda que el bloc de notas, todos los IDEs disponibles son gratuitos y los entornos de desarrollo son de rápida y fácil configuración.

LARAVEL

Laravel es uno de los frameworks de código abierto más fáciles de asimilar para PHP. Es simple, muy potente y tiene una interfaz elegante y divertida de usar. Fue creado en 2011 y tiene una gran influencia de frameworks como Ruby on Rails, Sinatra y ASP.NET MVC.

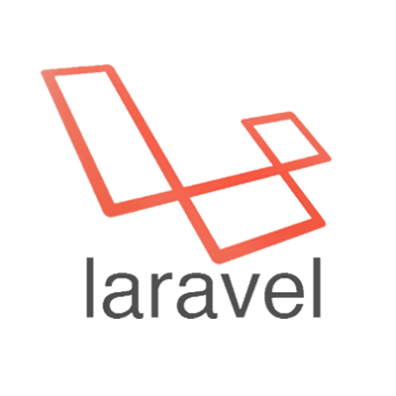


Figura 2: Logo de Laravel

Se utiliza un lenguaje de programación web muy conocido como los es PHP, y en este caso será con el framework LARAVEL ya que se considera que facilitara de una gran manera la creación de esta aplicación web, también permite la reducción de costos y tiempos en el desarrollo y mantenimiento otra razón por cual usar LARAVEL es que la curva de aprendizaje es relativamente baja.

### 

### GIT:

Git es un software de control de versiones diseñado por Linus Torvalds, pensando en la eficiencia y la confiabilidad del mantenimiento de versiones de aplicaciones cuando éstas tienen un gran número de archivos de código fuente. Su propósito es llevar registro de los cambios en archivos de computadora y coordinar el trabajo que varias personas realizan sobre archivos compartidos.



Figura 3: Logo Git

MySQL

MySQL es un sistema de gestión de bases de datos relacional desarrollado bajo licencia dual: Licencia pública general/Licencia comercial por Oracle Corporation y está considerada como la base datos de código abierto más popular del mundo, y una de las más populares en general junto a Oracle y Microsoft SQL Server, sobre todo para entornos de desarrollo web.

MySQL fue inicialmente desarrollado por MySQL AB (empresa fundada por David Axmark, Allan Larsson y Michael Widenius). MySQL A.B. fue adquirida por Sun Microsystems en 2008, y ésta a su vez fue comprada por Oracle Corporation en 2010, la cual ya era dueña desde 2005 de Innobase Oy, empresa finlandesa desarrolladora del motor InnoDB para MySQL.



Figura 4: Logo MySQL

MVC

Modelo-vista-controlador es un patrón de arquitectura de software, que separa los datos y la lógica de negocio de una aplicación de su representación y el módulo encargado de gestionar los eventos y las comunicaciones. Para ello MVC propone la construcción de tres componentes distintos que son el modelo, la vista y el controlador, es decir, por un lado, define componentes para la representación de la información, y por otro lado para la interacción del usuario. ​ Este patrón de arquitectura de software se basa en las ideas de reutilización de código y la separación de conceptos, características que buscan facilitar la tarea de desarrollo de aplicaciones y su posterior mantenimiento.



Figura 5: Estructura del patrón de diseño MVC

# **Requerimientos funcionales, no funcionales y del sistema.**

**Requerimientos funcionales**

|  |  |
| --- | --- |
| Requisito Funcional-001 | Agregar Usuario |
| Versión | 1.0 |
| Autor | Roque Chacón – Kevin Arce |
| Descripción | El sistema deberá contar con un módulo que permita agregar Usuarios a la Base de datos. |
| Precondición | Login |
| Secuencia normal | |  | | --- | | El usuario selecciona la opción de ingreso y se muestra la pantalla correspondiente. | | El administrador del sistema ingresa los datos respectivos al nuevo usuario. Se debe asignar el rol del empleado. | | | El sistema verifica que los datos ingresados sean correctos. Si los datos son correctos, el sistema realiza la inserción y muestra un mensaje informativo. Por el contrario, si los datos son incorrectos el sistema le indica al usuario que corrija los datos y realice nuevamente el ingreso de datos. | | | |
| Importancia | Vital |
| Comentarios | Se crean diferentes tipos de roles. |

*Tabla 1: Requisito Funcional-001 – Agregar Usuario*

|  |  |
| --- | --- |
| Requisito Funcional-002 | Editar Usuario |
| Versión | 1.0 |
| Autor | Roque Chacón – Kevin Arce |
| Descripción | El sistema deberá contar con un módulo que permita editar Usuarios y actualizarlos en la Base de datos. |
| Precondición | Login |
| Secuencia normal | |  | | --- | | El usuario(administrador) selecciona la opción de editar y se muestra la pantalla correspondiente. | | El administrador del sistema ingresa los datos respectivos al usuario. Se debe asignar el rol del empleado. | | | El sistema verifica que los datos ingresados sean correctos. Si los datos son correctos, el sistema realiza la inserción y muestra un mensaje informativo. Por el contrario, si los datos son incorrectos el sistema le indica al usuario que corrija los datos y realice nuevamente el ingreso de datos. | | | |
| Importancia | Vital |
| Comentarios | Solo los administradores pueden editar el perfil. |

*Tabla 2: Requisito Funcional-002 – Editar Usuario*

|  |  |
| --- | --- |
| Requisito Funcional-003 | Eliminar Usuario |
| Versión | 1.0 |
| Autor | Roque Chacón – Kevin Arce |
| Descripción | El sistema deberá contar con un módulo que permita eliminar Usuarios y actualizarlos en la Base de datos. |
| Precondición | Login |
| Secuencia normal | |  | | --- | | El usuario(administrador) selecciona la opción de eliminar y se muestra la pantalla correspondiente.  Se le muestra una alerta y si y solo si el administrador la acepta se elimina de la base de datos. | |
| Importancia | Vital |
| Comentarios | Solo los administradores pueden eliminar usuarios. |

*Tabla 3: Requisito Funcional-003 – Eliminar Usuario*

|  |  |
| --- | --- |
| Requisito Funcional-004 | Ver lista de Usuarios(empleados) |
| Versión | 1.0 |
| Autor | Roque Chacón – Kevin Arce |
| Descripción | El sistema deberá contar con un módulo que permita ver Usuarios almacenados en la Base de datos. |
| Precondición | Login |
| Secuencia normal | |  | | --- | | El usuario(administrador) selecciona la opción de ver y se muestra la pantalla correspondiente. | |
| Importancia | Vital |
| Comentarios | Solo los administradores pueden ver los usuarios (empleados). |

*Tabla 4: Requisito Funcional-004 – Ver Usuarios*

|  |  |
| --- | --- |
| Requisito Funcional-005 | Agregar Supermercado. |
| Versión | 1.0 |
| Autor | Roque Chacón – Kevin Arce |
| Descripción | El sistema deberá contar con un módulo que permita agregar Supermercados a la Base de datos. |
| Precondición | Login |
| Secuencia normal | |  | | --- | | El usuario selecciona la opción de ingreso y se muestra la pantalla correspondiente. | | El administrador del sistema ingresa los datos respectivos al nuevo Supermercado. | | | El sistema verifica que los datos ingresados sean correctos. Si los datos son correctos, el sistema realiza la inserción y muestra un mensaje informativo. Por el contrario, si los datos son incorrectos el sistema le indica al usuario que corrija los datos y realice nuevamente el ingreso de datos. | | | |
| Importancia | Vital |
| Comentarios | Hay diferentes sedes. |

*Tabla 5: Requisito Funcional-005 – Agregar Supermercado*

|  |  |
| --- | --- |
| Requisito Funcional-006 | Editar Supermercado |
| Versión | 1.0 |
| Autor | Roque Chacón – Kevin Arce |
| Descripción | El sistema deberá contar con un módulo que permita editar Supermercados y actualizarlos en la Base de datos. |
| Precondición | Login |
| Secuencia normal | |  | | --- | | El usuario(administrador) selecciona la opción de editar y se muestra la pantalla correspondiente. | | El administrador del sistema ingresa los datos respectivos al Supermercado. | | | El sistema verifica que los datos ingresados sean correctos. Si los datos son correctos, el sistema realiza la inserción y muestra un mensaje informativo. Por el contrario, si los datos son incorrectos el sistema le indica al usuario que corrija los datos y realice nuevamente el ingreso de datos. | | | |
| Importancia | Vital |
| Comentarios | Solo los administradores pueden editar el punto de venta. |

*Tabla 6: Requisito Funcional-006 – Editar Supermercado*

|  |  |
| --- | --- |
| Requisito Funcional-007 | Eliminar Supermercado |
| Versión | 1.0 |
| Autor | Roque Chacón – Kevin Arce |
| Descripción | El sistema deberá contar con un módulo que permita eliminar Supermercados y actualizarlos en la Base de datos. |
| Precondición | Login |
| Secuencia normal | |  | | --- | | El usuario(administrador) selecciona la opción de eliminar y se muestra la pantalla correspondiente.  Se le muestra una alerta y si y solo si el administrador la acepta se elimina de la base de datos. | |
| Importancia | Vital |
| Comentarios | Solo los administradores pueden eliminar Supermercados. |

*Tabla 7: Requisito Funcional-007 – Eliminar Supermercados*

|  |  |
| --- | --- |
| Requisito Funcional-008 | Ver lista de Supermercados. |
| Versión | 1.0 |
| Autor | Roque Chacón – Kevin Arce |
| Descripción | El sistema deberá contar con un módulo que permita ver Supermercados almacenados en la Base de datos. |
| Precondición | Login |
| Secuencia normal | |  | | --- | | El usuario(administrador) selecciona la opción de ver y se muestra la pantalla correspondiente. | |
| Importancia | Vital |
| Comentarios | Solo los administradores pueden ver los Supermercado. |

*Tabla 8: Requisito Funcional-008 – Ver Supermercados*

|  |  |
| --- | --- |
| Requisito Funcional-009 | Crear Horario. |
| Versión | 1.0 |
| Autor | Roque Chacón – Kevin Arce |
| Descripción | El sistema deberá contar con un módulo que permita la creación de horarios y almacenarlos a la Base de datos. |
| Precondición | Login |
| Secuencia normal | |  | | --- | | El usuario selecciona la opción de ingreso a un supermercado y se muestra la pantalla correspondiente. | | El administrador del sistema ingresa los datos respectivos al nuevo horario. Se debe asignar el empleado para ese horario. | | | El sistema verifica que los datos ingresados sean correctos. Si los datos son correctos, el sistema realiza la inserción y muestra un mensaje informativo. Por el contrario, si los datos son incorrectos el sistema le indica al usuario que corrija los datos y realice nuevamente el ingreso de datos. | | | |
| Importancia | Vital |
| Comentarios | El empleado puede tener diferentes horarios en diferentes supermercados, siempre y cuando los horarios no choquen. |

*Tabla 9: Requisito Funcional-009 – Crear de Horario*

|  |  |
| --- | --- |
| Requisito Funcional-010 | Editar Horario |
| Versión | 1.0 |
| Autor | Roque Chacón – Kevin Arce |
| Descripción | El sistema deberá contar con un módulo que permita editar Horarios y actualizarlos en la Base de datos. |
| Precondición | Login |
| Secuencia normal | |  | | --- | | El usuario(administrador) selecciona la opción de editar y se muestra la pantalla correspondiente. | | El administrador del sistema ingresa los datos respectivos al Horario. Se debe asignar el empleado. | | | El sistema verifica que los datos ingresados sean correctos. Si los datos son correctos, el sistema realiza la inserción y muestra un mensaje informativo. Por el contrario, si los datos son incorrectos el sistema le indica al usuario que corrija los datos y realice nuevamente el ingreso de datos. | | | |
| Importancia | Vital |
| Comentarios | Solo los administradores pueden editar los horarios. |

*Tabla 10: Requisito Funcional-010 – Editar Horario*

|  |  |
| --- | --- |
| Requisito Funcional-011 | Eliminar Horario |
| Versión | 1.0 |
| Autor | Roque Chacón – Kevin Arce |
| Descripción | El sistema deberá contar con un módulo que permita eliminar Horarios y actualizarlos en la Base de datos. |
| Precondición | Login |
| Secuencia normal | |  | | --- | | El usuario(administrador) selecciona la opción de eliminar y se muestra la pantalla correspondiente.  Se le muestra una alerta y si y solo si el administrador la acepta se elimina de la base de datos. | |
| Importancia | Vital |
| Comentarios | Solo los administradores pueden eliminar Horarios. |

*Tabla 11: Requisito Funcional-011 – Eliminar Horarios.*

|  |  |
| --- | --- |
| Requisito Funcional-012 | Ver lista de Horarios |
| Versión | 1.0 |
| Autor | Roque Chacón – Kevin Arce |
| Descripción | El sistema deberá contar con un módulo que permita ver Horarios almacenados en la Base de datos. |
| Precondición | Login |
| Secuencia normal | |  | | --- | | El usuario(administrador) selecciona la opción de ver y se muestra la pantalla correspondiente, lo podrá ver tanto por supermercados como por empleado. | |
| Importancia | Vital |
| Comentarios | Solo los administradores pueden ver los horarios. |

*Tabla 12: Requisito Funcional-012 – Ver Horarios*

|  |  |
| --- | --- |
| Requisito Funcional-013 | Login |
| Versión | 1.0 |
| Autor | Roque Chacón – Kevin Arce |
| Requisitos asociados | Administradores en la base de datos. |
| Descripción | El sistema deberá autentificar las credenciales de los usuarios que apliquen para ingresar a la plataforma. |
| Secuencia normal | |  | | --- | | Se ingresa Usuario y contraseña, el sistema verifica credenciales. | |
| Importancia | Vital |
| Comentarios | Hay diferentes roles. Verificar roles también. |

*Tabla 13: Requisito Funcional-013 – Login*

|  |  |
| --- | --- |
| Requisito Funcional-014 | Generar Reportes |
| Versión | 1.0 |
| Autor | Roque Chacón – Kevin Arce |
| Requisitos asociados | CRUD de horarios. |
| Descripción | El sistema luego de la confirmación de los horarios creados genera dos reportes, uno el horario de los empleados y seguidamente se podrá visualizar también cantidad de horas trabajadas por empleado por semana. |
| Secuencia normal | |  | | --- | | Al pulsar el botón de reportes el mismo le generara los reportes, sino los hay genera una alerta. | |
| Importancia | Vital |
| Comentarios | Horarios-productividad de semana. |

*Tabla 14: Requisito Funcional-014 – Reportes*

# **Solución del problema.**

Se diseña una aplicación web que pueda satisfacer todas las necesidades que tiene la empresa Supermercados Ecónomas con respecto a la creación y distribución de horarios para sus colaboradores. La aplicación cuenta con varias pantallas entre las cuales se encontrarán la creación de roles por usuario ya que no todos los usuarios creados podrán crear y modificar dichos horarios, la siguiente pantalla será la creación y modificación de puestos de trabajo, como por ejemplo puede ser cajero, gondolero, carnicero, entre otros, esta pantalla se realizará para un mejor manejo de pagos por puesto. La siguiente pantalla será para los supermercados o puntos de venta que ellos cuentan con varios puntos de venta alrededor de la zona de Pital, Venecia y Aguas Zarcas, y esto aportará mayor administración de todos los puntos de venta.

Por otro lado, se realiza la pantalla de horarios que será la más importante de la aplicación, ya que la idea principal de esta va a ser la administración de horarios; para poder solucionar el gran problema que tienen con la manipulación de horarios, se buscará una manera fácil y sencilla que todos los usuarios que utilicen la aplicación puedan aprovechar todas y cada una de las ventajas, se procederá diseñando una interfaz simple con pocos botones e información para poder crear, editar, o modificar los horarios, se agregará un seleccionador de fechas y hora, así el usuario podrá elegir una fecha y hora mucho más fácil.

Una vez listo los horarios el usuario podrá guardar lo y luego generar el reporte del horario para poder distribuirlo entre todos los colaboradores, y también podrá realizar un reporte para conocer cuantas horas trabajará un colaborador a la semana para así llevar el control de que cada colaborador trabaje las 48 horas semanales estipuladas por la ley del trabajador.

Para dar solución a la problemática de dicha cadena de super mercados, se utiliza para la manipulación de datos, una base de datos llamada SQLLITE la facilita la interacción con la aplicación construida, dicha base de datos es la que provee y almacena la información de la aplicación, de las horas por persona que trabajaran. Con esta herramienta podemos dar solución a nuestra lógica de negocios, ya que, la aplicación se crea con un framework llamado LARAVEL, el cual con su ORM nos genera consultas muy eficientes y eficaces, de las cuales se puede confiar para una sensata toma de decisiones. La estructura de la base de datos será relacional, se usa así pensando en la escabilidad de la aplicación.

A la hora de un desarrollador crea una aplicación tiene que analizar y pensar en la escalabilidad, esto se refiere a añadir cosas en futuro, por ejemplo, API’s, esto mejorara la lógica de negocios, pero no la afectara, ya que una aplicación tiene que ir orientada a esa metodología, para cumplir estas características se pensó en un lenguaje sencillo como lo es PHP, y a su vez usamos el framework LARAVEL para hacer el scaffoling más rápido, así ahorrándonos tiempo en programación. Luego se usa plantillas de Bootstrap las cuales nos guían y orientan en el uso interactivo y sencillo de la aplicación. Bootstrap cuenta con una galería muy extensa de diseño lo cual genera una herramienta útil para el diseño ya que también tiene por defecto el método de grilletas, lo cual nos genera una aplicación “responsiva” o adaptable a cualquier navegador.

# **Diagramas**

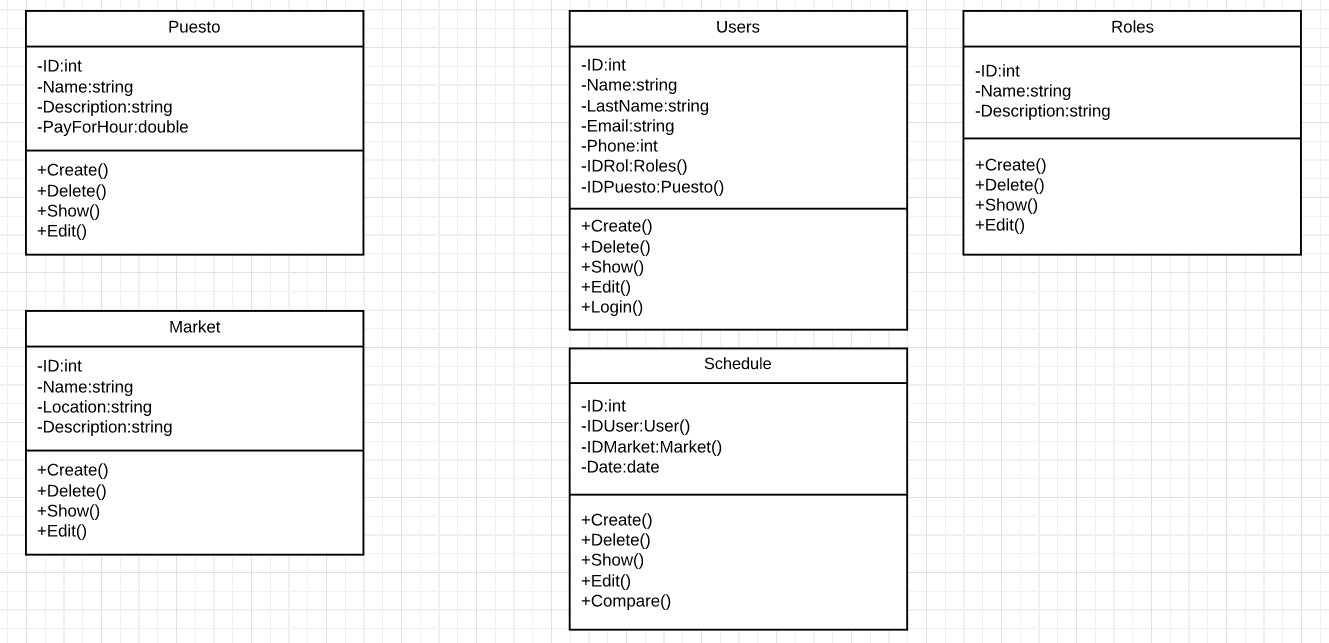
**Diagrama de casos de uso.**

Imagen que contiene texto

Descripción generada con confianza alta

*Figura 6: Diagrama de casos de uso.*

**Diagrama de clases.**



*Figura 7: Diagrama de clases.*

**Diagrama de base de datos.**

Imagen que contiene captura de pantalla

Descripción generada con confianza muy alta

*Figura 8: Diagrama de base de datos.*

**Diagrama de Arquitectura**

Imagen que contiene texto

Descripción generada con confianza muy alta

*Figura 9: Diagrama de Arquitectura.*

# **Criterios de aceptación.**

Tabla 15: Caso de prueba Inicio de Sesión

|  |  |
| --- | --- |
| **Caso de prueba – Inicio de Sesión** | |
| Código | CP001 |
| Hardware y software | Computadora, Laptop |
| Sistema operativo | Windows, Mac o Linux |
| Navegador | Google Chrome |
| Resolución | N/A |
| Comentarios | N/A |
| Nombre | Pruebas para el Inicio de Sesión |
| Descripción | Verificar que los usuarios registrados anteriormente en la aplicación puedan ingresar correctamente. |
| Encargado | Roque Chacon – Kevin Arias |
| Precondiciones | * Tener una computadora de escritorio o laptop con acceso a internet. * Tener un usuario en el sistema de Horarios Ecónomas. |
| Pasos | 1. Ingresar a la aplicación web Horarios Ecónomas por medio del navegador Google Chrome. 2. Ingresar los datos solicitados para poder ingresar, como lo son nombre de usuario y contraseña. 3. Dar click en el botón ingresar. |
| Resultados esperados | **Pass:** Se debe mostrar la pantalla inicio de la aplicación.  **Fail:** Debe mostrar un mensaje indicándole cual fue el error. |
| Prioridad | Alta |

|  |  |
| --- | --- |
| **Caso de prueba – Agregar** | |
| Código | CP002 |
| Hardware y software | Computadora, Laptop |
| Sistema operativo | Windows 10 |
| Navegador | Windows, Mac o Linux |
| Resolución | N/A |
| Comentarios | N/A |
| Nombre | Pruebas para Agregar Usuario, Agregar Rol, Agregar Puesto, Agregar Supermercado |
| Descripción | Verificar que el usuario, rol, puesto o supermercado se crea de la manera adecuada. |
| Encargado | Roque Chacon – Kevin Arias |
| Precondiciones | * Tener una computadora de escritorio o laptop con acceso a internet. * Tener un usuario en el sistema de Horarios Ecónomas. * Tener el rol permitido para poder hacer las modificaciones. |
| Pasos | 1. Ingresar a la pantalla principal de usuarios, roles, puestos o supermercados. 2. Hacer clic en el Botón crear nuevo. 3. Llenar los datos solicitados. 4. Hacer clic en el Botón de crear. |
| Resultados esperados | **Pass:** Se debe mostrar la pantalla de todos los registros.  **Fail:** Debe mostrar un mensaje indicándole cual fue el error. |
| Prioridad | Alta |

Tabla 16: Caso de prueba Agregar

Tabla 17: Caso de prueba Editar

|  |  |
| --- | --- |
| **Caso de prueba - Editar** | |
| Código | CP003 |
| Hardware y software | Computadora, Laptop |
| Sistema operativo | Windows, Mac o Linux |
| Navegador | Google Chrome |
| Resolución | N/A |
| Comentarios | N/A |
| Nombre | Pruebas para Editar Usuario, Editar Rol, Editar Puesto, Editar Supermercado |
| Descripción | Verificar que el usuario, rol, puesto o supermercado se pueda editar satisfactoriamente. |
| Encargado | Roque Chacon – Kevin Arias |
| Precondiciones | * Tener una computadora de escritorio o laptop con acceso a internet. * Tener un usuario en el sistema de Horarios Ecónomas. * Tener el rol permitido para poder hacer las modificaciones. |
| Pasos | 1. Ingresar a la pantalla principal de usuarios, roles, puestos o supermercados. 2. Hacer clic en el Botón editar en cualquiera de los registros que quiera modificar. 3. Modificar los datos que desee. 4. Hacer clic en el Botón de Guardar. |
| Resultados esperados | **Pass:** Se debe mostrar la pantalla de todos los registros.  **Fail:** Debe mostrar un mensaje indicándole cual fue el error. |

Tabla 18: Caso de prueba Eliminar

|  |  |
| --- | --- |
| **Caso de prueba -Eliminar** | |
| Código | CP004 |
| Hardware y software | Computadora, Laptop |
| Sistema operativo | Windows, Mac o Linux |
| Navegador | Google Chrome |
| Resolución | N/A |
| Nombre | Pruebas para Eliminar Usuario, Eliminar Rol, Eliminar Puesto, Eliminar Supermercado |
| Descripción | Verificar que el usuario, rol, puesto o supermercado se pueda eliminar correctamente. |
| Encargado | Roque Chacon – Kevin Arias |
| Precondiciones | * Tener una computadora de escritorio o laptop con acceso a internet. * Tener un usuario en el sistema de Horarios Ecónomas. * Tener el rol permitido para poder eliminar el registro. |
| Pasos | 1. Ingresar a la pantalla principal de usuarios, roles, puestos o supermercados. 2. Hacer clic en el Botón eliminar en cualquiera de los registros que quiera modificar. 3. Verificar que eligió el registro que quiere eliminar. 4. Hacer clic en el Botón de eliminar. |
| Resultados esperados | **Pass:** Se debe mostrar la pantalla de todos los registros.  **Fail:** Debe mostrar un mensaje indicándole cual fue el error. |
| Prioridad | Alta |

|  |  |
| --- | --- |
| **Caso de prueba - Detalles** | |
| Código | CP005 |
| Hardware y software | Computadora, Laptop |
| Sistema operativo | Windows, Mac o Linux |
| Navegador | Google Chrome |
| Resolución | N/A |
| Comentarios | N/A |
| Nombre | Pruebas para Ver Usuario, Ver Rol, Ver Puesto, Ver Supermercado |
| Descripción | Verificar que los usuarios, roles, puestos o supermercados se visualicen de manera satisfactoria. |
| Encargado | Roque Chacon – Kevin Arias |
| Precondiciones | * Tener una computadora de escritorio o laptop con acceso a internet. * Tener un usuario en el sistema de Horarios Ecónomas. * Tener el rol permitido para poder hacer la visualización. |
| Pasos | 1. Ingresar a la pantalla principal de usuarios, roles, puestos o supermercados. 2. Hacer clic en el Botón detalles en cualquiera de los registros que quiera visualizar. |
| Resultados esperados | **Pass:** Se debe mostrar la pantalla de todos los registros.  **Fail:** Debe mostrar un mensaje indicándole cual fue el error. |
| Prioridad | Alta |

Tabla 19: Caso de prueba Detalles

Tabla 20: Caso de prueba Crear Horario

|  |  |
| --- | --- |
| **Caso de prueba – Crear Horario** | |
| Código | CP006 |
| Hardware y software | Computadora, Laptop |
| Sistema operativo | Windows, Mac o Linux |
| Navegador | Google Chrome |
| Resolución | N/A |
| Comentarios | N/A |
| Nombre | Pruebas para Crear un Horario. |
| Descripción | Comprobar que los horarios se crean correctamente |
| Encargado | Roque Chacon – Kevin Arias |
| Precondiciones | * Tener una computadora de escritorio o laptop con acceso a internet. * Tener un usuario en el sistema de Horarios Ecónomas. * Tener el rol permitido para poder hacer la creación. |
| Pasos | 1. Ingresar a la pantalla principal de horarios 2. Hacer clic en el Botón crear. 3. Elegir el supermercado al cual se va a crear el horario. 4. Llenar los datos solicitados. 5. Dar click en el botón crear |
| Resultados esperados | **Pass:** Se debe mostrar la pantalla de todos los registros.  **Fail:** Debe mostrar un mensaje indicándole cual fue el error. |
| Prioridad | Alta |
| Notas | N/A |

Tabla 21: Caso de prueba Editar Horario

|  |  |
| --- | --- |
| **Caso de prueba – Editar Horario** | |
| Código | CP007 |
| Hardware y software | Computadora, Laptop |
| Sistema operativo | Windows, Mac o Linux |
| Navegador | Google Chrome |
| Resolución | N/A |
| Comentarios | N/A |
| Nombre | Pruebas para editar un Horario. |
| Descripción | Comprobar que los horarios se puedan Editar correctamente |
| Encargado | Roque Chacon – Kevin Arias |
| Precondiciones | * Tener una computadora de escritorio o laptop con acceso a internet. * Tener un usuario en el sistema de Horarios Ecónomas. * Tener el rol permitido para poder hacer la creación. |
| Pasos | 1. Ingresar a la pantalla principal de horarios 2. Hacer clic en el Botón editar en cualquier registro que desee editar. 3. Modificar los datos que desee. 4. Hacer clic en el botón guardar. |
| Resultados esperados | **Pass:** Se debe mostrar la pantalla de todos los registros.  **Fail:** Debe mostrar un mensaje indicándole cual fue el error. |
| Prioridad | Alta |
| Notas | N/A |

Tabla 22: Caso de prueba Detalles Horario

|  |  |
| --- | --- |
| **Caso de prueba - Detalles Horario** | |
| Código | CP008 |
| Hardware y software | Computadora, Laptop |
| Sistema operativo | Windows, Mac o Linux |
| Navegador | Google Chrome |
| Resolución | N/A |
| Comentarios | N/A |
| Nombre | Pruebas para Visualizar un Horario. |
| Descripción | Verificar que los horarios se puedan visualizar correctamente |
| Encargado | Roque Chacon – Kevin Arias |
| Precondiciones | * Tener una computadora de escritorio o laptop con acceso a internet. * Tener un usuario en el sistema de Horarios Ecónomas. * Tener el rol permitido para poder hacer la visualización. |
| Pasos | 1. Ingresar a la pantalla principal de horarios 2. Hacer clic en el Botón detalles en cualquier registro que desee. 3. Verificar que eligió el registro correctamente. |
| Resultados esperados | **Pass:** Se debe mostrar la pantalla del registro seleccionado.  **Fail:** Debe mostrar un mensaje indicándole cual fue el error. |
| Prioridad | Alta |
| Notas | N/A |

Tabla 23: Caso de prueba Eliminar Horario

|  |  |
| --- | --- |
| **Caso de prueba - Eliminar Horario** | |
| Código | CP009 |
| Hardware y software | Computadora, Laptop |
| Sistema operativo | Windows, Mac o Linux |
| Navegador | Google Chrome |
| Resolución | N/A |
| Comentarios | N/A |
| Nombre | Pruebas para Eliminar un Horario. |
| Descripción | Comprobar que los horarios se puedan eliminar correctamente |
| Encargado | Roque Chacon – Kevin Arias |
| Precondiciones | * Tener una computadora de escritorio o laptop con acceso a internet. * Tener un usuario en el sistema de Horarios Ecónomas. * Tener el rol permitido para poder hacer la creación. |
| Pasos | 1. Ingresar a la pantalla principal de horarios 2. Hacer clic en el Botón editar en cualquier registro que desee eliminar. 3. Verificar que eligió el registro correctamente. 4. Si no está seguro puede regresar a todos los datos haciendo clic en el botón volver a la lista. 5. Si está seguro puede hacer clic en el botón eliminar. |
| Resultados esperados | **Pass:** Se debe mostrar la pantalla de todos los registros.  **Fail:** Debe mostrar un mensaje indicándole cual fue el error. |
| Prioridad | Alta |

Tabla 24: Caso de prueba Reportes

|  |  |
| --- | --- |
| **Caso de prueba - Reportes** | |
| Código | CP010 |
| Hardware y software | Computadora, Laptop |
| Sistema operativo | Windows, Mac o Linux |
| Navegador | Google Chrome |
| Resolución | N/A |
| Comentarios | N/A |
| Nombre | Pruebas para generar reportes. |
| Descripción | Comprobar que se generan bien los reportes a partir de un horario ya creado. |
| Encargado | Roque Chacon – Kevin Arias |
| Precondiciones | * Tener una computadora de escritorio o laptop con acceso a internet. * Tener un usuario en el sistema de Horarios Ecónomas. * Tener el rol permitido para poder hacer la creación. |
| Pasos | 1. Dar clic en reportes que se encuentra ubicado en el menú lateral de la izquierda. 2. Elegir el horario para generar el reporte 3. Dar clic generar. |
| Resultados esperados | **Pass:** Se descargará un PDF con el reporte  **Fail:** Debe mostrar un mensaje indicándole cual fue el error. |
| Prioridad | Alta |
| Notas | N/A |

# **Resultados de las pruebas**

Laravel incluye en el directorio principal de tu proyecto un directorio llamado /tests. En este directorio se escribe el código que probara el correcto funcionamiento de la aplicación.

* El directorio Feature donde escribimos pruebas que emulan peticiones HTTP al servidor.
* El directorio Unit donde escribimos pruebas que se encargan de probar partes individuales de la aplicación (como clases y métodos).

La primera prueba generada es la prueba de rutas, ya que esta nos permitirá la interacción de la aplicación.

Ingreso al sistema sin usuarios registrados. Se pretende medir las validaciones de Login o ingreso al sistema.

Ingreso con usuario default y creación de usuarios(empleados). Se logra ingresar y hacer un CRUD completo de los usuarios al sistema.

CRUD de roles, CRUD de puntos de venta, se toman las medidas necesarias, se hace esto para no dejarle nada al azar y ser precavidos en cualquier acontecimiento.

En la parte más mediática es la creación de horarios ya que este maneja la lógica y funcionabilidad de la aplicación, entre las pruebas que se aplican son, creación de horarios en diferentes puntos de venta, creación exitosa, creación de horarios con asignación de empleados en diferentes sedes, resultado exitoso, asignar días libres a la jornada laboral, la manipulación fue exitosa del mismo.

Creación de reportes, este proceso es mostrarle de manera pdf el horario creado para una semana x. En donde se asignan diferentes empleados. Este pdf es generado por cada punto de venta. O bien por empleado.

# **Conclusiones.**

El proyecto ha logrado cumplir todos las propuestas y objetivos básicos que se habían definido y respetando las especificaciones planteadas.

Se ha desarrollado la aplicación para la parte de administrativa de los supermercados EconoMás, la cual fue revisa por dos de esas personas. Las cuales se mostraron muy satisfechas del trabajo presentado por este grupo de estudiantes.

Ellos muestran su satisfacción ya que se construyó una agradable y simple interfaz con un menú lateral que da la sensación de rapidez al ir navegando atreves de todas las vistas de la aplicación. Esto permite que la experiencia del usuario sea satisfactoria, además de que se le permite mantener un orden la hora de administrar los horarios. Al ser una aplicación web orientada a negocios, se pretende que esta sea lo mas optima posible para ello se han tomado las medidas del caso. Realizando pruebas en diferentes entornos.

La creación de los horarios como la generación de reportes han sido probados en distintos navegadores modernos con satisfactorio resultado, gracias a que ha sido desarrollado empleando estándares y tecnologías abiertas. Además, independientemente del navegador utilizado, los resultados se ajustan al espacio dejado libre por la ventaja del navegador y junto fue probado en dispositivos móviles.

# **Recomendaciones.**

Siempre desarrollar pensando en la escalabilidad, usar tecnología con respaldo en documentación, Laravel es una excelente plataforma que cumple con todos los estándares necesarios, para futuro se recomienda migrar la lógica de negocio al lado del servidor. Ya que esto daría mayor seguridad a los datos.

Omitir los eliminar, no borrar datos, ya que podrían ser tomados en cuenta en un futuro inmediato, ya que estos son indispensables para la toma de decisiones. Y al final de esto se basa la continuidad de un negocio, una aplicación que almacene la mayor cantidad de datos, ayuda a facilitar una eficiente toma de decisiones. Se recomienda crear un atributo tipo booleano para así deshabilitar los datos y no borrarlos.

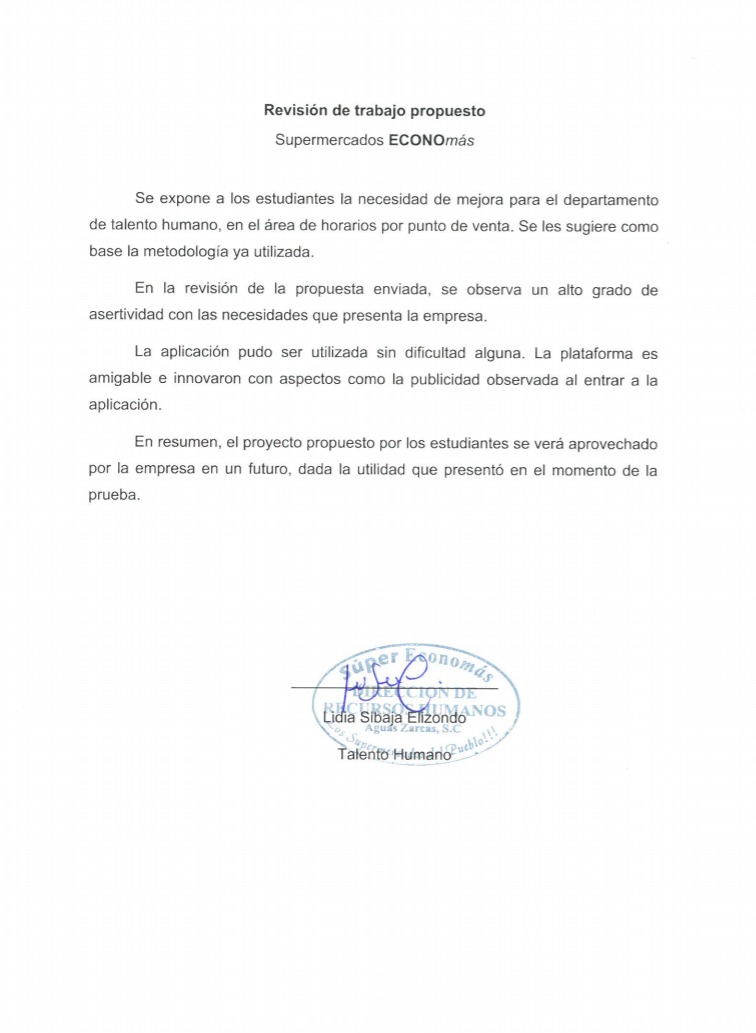
Se le recomienda al cliente que, a la hora de mostrar los datos, omitir en el reporte el salario, ya que podría ocasionar rencillas entre los colaboradores. Otra buena practica seria, enviar solo el horario a cada persona y evitar enseñar los horarios a los otros. Ya que en este momento el horario generado no se divide según el empleado sino va en un diagrama todos los horarios.

# **Información del cliente**

* **Maureen María Porras Méndez**
* [porrasmau08@gmail.com](mailto:porrasmau08@gmail.com)
* **Supermercados ECONO *más***| (506) 2474-4399 Ext. 115

# **Anexos**

Carta entregada por el cliente, mostrando el agradecimiento por la aplicación.



# **Plan de contingencia**

Falla en el IDE, en este caso podemos usar cualquier editor para así seguir con la edición del programa. Entre otras cosas manejamos nuestro proyecto en GIT en el cual podemos controlar las versiones de nuestro proyecto por alguna eventualidad en el mismo.

Creación de validaciones para cada uno de los métodos para así asegurar un funcionamiento eficaz y eficiente de la aplicación.

Si se presenta alguna eventualidad con la base de datos, nosotros manejamos una característica de Laravel y de muchos frameworks, el cual nos crea migraciones. Y las podemos ejecutar cuando se presente alguna eventualidad. Y así no perder tiempo, el cual es estos modelos de negocio es de mucha importancia.