

Software de administración y generación de bitácoras para servicios generales

Fecha de elaboración: 20 de marzo de 2021

Última actualización: 31 de octubre de 2022

Nombre del proyecto: Software para administración de servicios generales

Integrantes:

Juan Carlos González Medina

Presentación	2
Introducción	2
Propósito del proyecto	2
Descripción	2
Perspectiva del producto a desarrollar	2
Estudio de viabilidad del proyecto	3
Paradigma de desarrollo	3
Roles del equipo involucrado	3
Planificación	4
Requerimientos	5
Entrevista (instrumento(s) para recabar requerimientos)	5
Requerimientos funcionales	5
Requerimientos no funcionales	8
Diseño del sistema	11
Modelado del sistema	11
Flujo de datos	12
Diccionario de datos	14
Estructura de base de datos	15
Diseño arquitectónico	19
Diseño de interfaces	20
Codificación	42
Pruebas	43
Conclusiones	50
Referencias	51

1. Presentación

Misión

Integrantes de equipo de software dedicados a desarrollar plataformas que ayuden a personas a realizar de manera eficaz sus tareas por medio de tecnologías, desarrollo, y aplicación de software enfocadas principalmente al entorno web.

Visión

Como equipo, demostrar que se es confiable en la entrega de proyectos que aporten un beneficio a las tareas diarias o bien, en desarrollo personal o de equipo para enfocarse principalmente en la mejora de alguna actividad o tarea por medio de estas entregas de manera consistente y mayormente estables en su uso continuo.

Objetivo.

Entrega continua, eficaz, mejorable, adaptable y consistente de software para el crecimiento entre ambos sujetos, como lo es el equipo de desarrolladores y el cliente que recibirá los beneficios de esta misma entrega.

2. Introducción

2.1 Propósito del proyecto

El objetivo principal de este proyecto se centra primordialmente en las carencias, amenazas y debilidades que forman parte en la producción de labores dentro del equipo de trabajo de servicios generales en el Centro Universitario de los Altos. Actualmente el equipo encargado dentro del centro universitario tiene como gestión de tareas el uso de materiales físicos como es el papeleo representado con reportes de labores, bitácoras, análisis de datos y por lo tanto la creación, seguimiento y cierre de reportes de manera poco eficaz y ágil. Por este motivo se crea este proyecto para evitar esas complicaciones y optar por entregar software que entregue un avance temprano y eficiente ante el equipo de trabajo así como reflejar el producto físico o visual dentro del centro universitario.

3. Descripción

3.1 Perspectiva del producto a desarrollar

Las soluciones que se tienen propuestas para resolver los problemas enlistados presentados en el área de servicios generales son:

- 1) Crear una plataforma que apoye en las tareas administrativas, de gestión y usuarios con diferentes roles y permisos.
- 2) Implementación de un servidor web que tenga como propósito el arranque y funcionamiento de la plataforma, donde pueda ser guardada la información que sea solicitada y administrar según convenga.
- 3) Creación digital PDF para posteriormente si es necesario tener material físico de evidencia de los reportes generados.

3.2 Estudio de viabilidad del proyecto

- Viabilidad técnica:

El centro universitario de los Altos cumple con recursos físicos como es el servidor web para dar mantenimiento y seguimiento en la plataforma, cubriendo los requerimientos que necesita la plataforma, al igual que el equipo, por medio de la portabilidad, cumplen con dispositivos y equipo físico que sea funcional para el uso de la misma plataforma.

- Viabilidad económica:

No es requerible equipo o servicios, ya que esto es un servicio social por medio de estudiantes en cuestión de implementación de software al igual que no se requiere equipo físico extra al que ya cumple el centro universitario.

- Viabilidad operacional:

Cumplen los requerimientos, ya que este servicio será sustentado y en revisión constante por alumnos del centro universitario u otro tercero que sea requerido por la universidad, por lo tanto este proyecto es viable en cuestión de funcionamiento y durabilidad.

3.3 Paradigma de desarrollo

Se integran estrategias de desarrollo ágil tal fue como el uso de SCRUM para la creación, continuación y finalización de secciones de código o trabajo necesario por parte del equipo que integra a los desarrolladores del proyecto. Al igual que se tomaron herramientas de software gratuitas y públicas para su inserción en el programa al igual que la constante comunicación e historial entre el equipo de desarrolladores, así como herramientas de software proporcionadas por Google y de comunicación como lo fue Discord.

3.4 Roles del equipo involucrado

Juan Carlos González Medina

Rol: Desarrollador de software

Funciones: Desarrollar código enriquecido en distintos lenguajes de programación como lo fueron

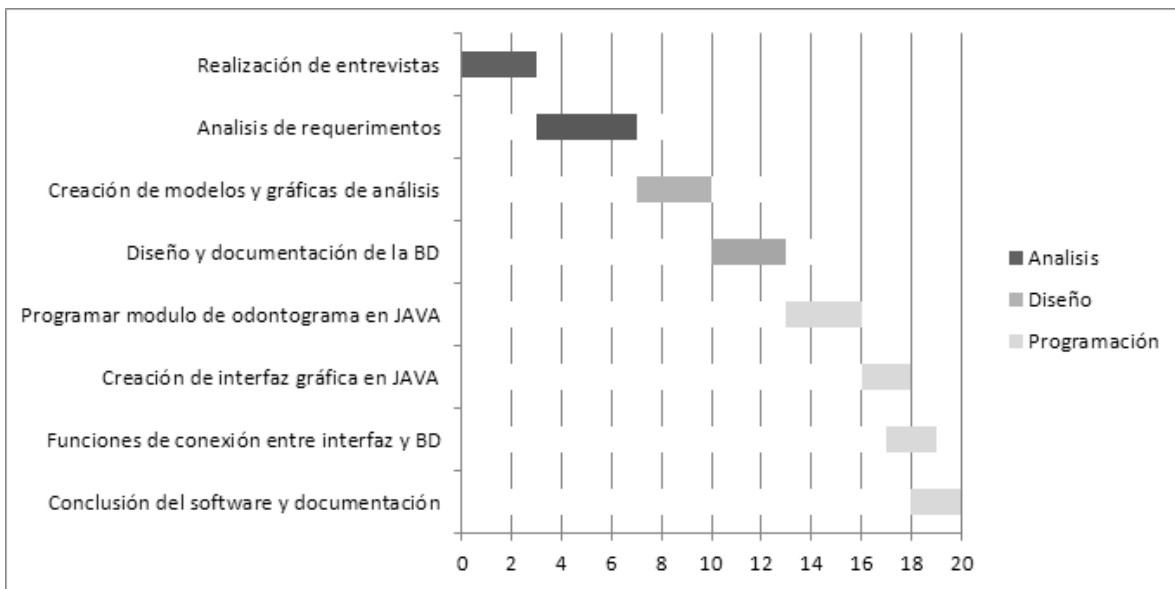
javascript y PHP, así como lo fue en frontend y backend, del mismo modo se utilizaron herramientas de servidores y programación de lenguajes como MySQL por medio de Laragon.

Responsabilidades: Observar, analizar y ejecutar tareas que fueran necesarias para el desarrollo del programa al igual que estar al pendiente de cualquier cambio o sugerencia dada por medio del cliente e implementarla en el programa de forma funcional e intuitiva.

Derechos: Complementar secciones del programa que fueran requeridos para su uso futuro o actual en las que serían analizadas como viables o no, por sí mismo el desarrollo de software y análisis de problemas que pudiesen surgir durante las labores comunes dentro de este proyecto.

3.5 Planificación

Diagrama de Gantt para planificación



4. Requerimientos

4.0 Entrevista (instrumento(s) para recabar requerimientos)

Entrevista realizada:

1. ¿Qué requiere implementar o mejorar en el programa?

R= Secciones de Tickets para su creación, modificación, eliminación al igual que un panel estadístico visual que represente los tickets que hayan sido abiertos o cerrados en determinada fecha dentro de un rango periodico.

2. ¿Deberán incluirse nuevas implementaciones respecto a la elaboración de reportes?

R= Requerimos el mismo control pero si se requieren implementaciones extras que sean adecuadas e intuitivas puede proseguir.

3. ¿En qué consisten los reportes?

R= Puede ser cualquier tarea, con tal que tenga una etiqueta o forma de representación que de a entender que sea un reporte nuevo como abierto, que se esté ejecutando o que ya haya sido o sea un reporte cerrado.

4. ¿Existe alguna tarea en específico que requiera una implementación permanentemente en el programa?

R= Actualmente no

4.1 Requerimientos funcionales [3]

Lista de requerimientos funcionales:

- Registro de reportes abiertos, en ejecución y cerrados.
- Visualización de reportes y detalles, como evidencias fotográficas, especificación del reporte, entre otros.
- Gráfica que representa el total de reportes abiertos y cerrados en un rango de fechas.
- Generación de pdf con los detalles incrustados en los reportes.

Identificador: SG01	
Prioridad: Alta	Fuente: Entrevista
Necesidad: Esencial	
Claridad: Alta	Verificabilidad: Alta
Estabilidad:	Alta
Descripción:	Generación de reportes de tickets que representen los problemas o labores que se encuentren dentro del centro universitario por medio de un CRUD.

Identificador: SG02	
Prioridad: Alta	Fuente: Entrevista
Necesidad: Esencial	

Claridad: Alta	Verificabilidad: Alta
Estabilidad:	Alta
Descripción:	Visualización de los reportes generados con su respectiva evidencia y especificaciones.

Identificador: SG04

Prioridad: Alta	Fuente: Entrevista
Necesidad: Esencial	
Claridad: Alta	Verificabilidad: Alta
Estabilidad:	Alta
Descripción:	Generación de gráfica con los datos de los tickets abiertos y cerrados.

Identificador: SG05

Prioridad: Alta	Fuente: Entrevista
Necesidad: Esencial	
Claridad: Alta	Verificabilidad: Alta
Estabilidad:	Alta
Descripción:	Creación de reportes visualizados por medio de PDF.

4.2 Requerimientos no funcionales

Lista de requerimientos no funcionales:

- El programa requerido será sustentado por medio del framework php Laravel [2].
- El programa actualmente deberá ser funcional en un servidor físico.
- Los reportes serán generados y representados por medio de un PDF.
- La representación gráfica será optimizada por medio de herramientas como lo es Google Charts [6].
- Los datos generados y almacenados serán administrados por medio de SQL.
- Se utilizará el uso de roles [5] por medio de php en conjunto de la base de datos.

Identificador: SGS01

Prioridad: Alta	Fuente: Entrevista
Necesidad: Esencial	
Claridad: Alta	Verificabilidad: Alta
Estabilidad:	Alta
Descripción:	El programa utilizará Laravel PHP 7.4 para el uso medido de la base de datos como lo es PhpMyAdmin en MySQL.

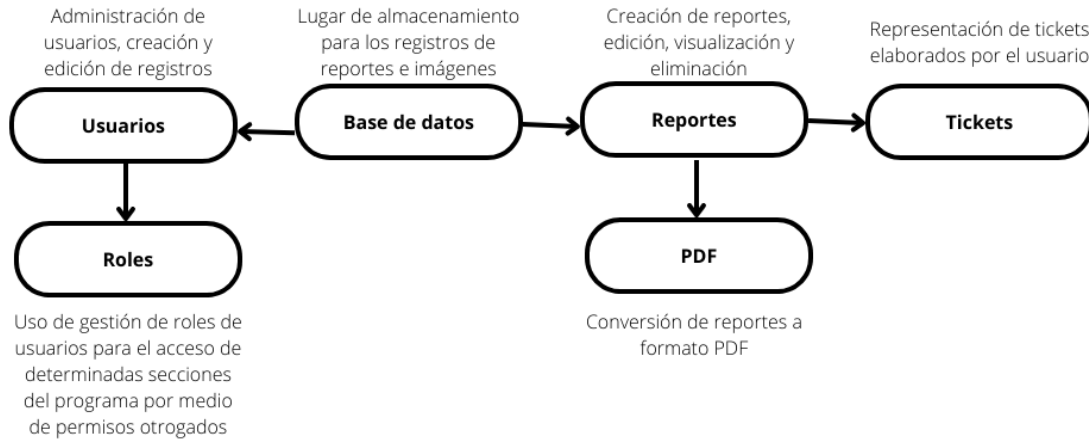
Identificador: SGS02

Prioridad: Alta	Fuente: Entrevista
Necesidad: Esencial	
Claridad: Alta	Verificabilidad: Alta
Estabilidad:	Alta
Descripción:	Será requerido un servidor web físico patrocinado por medio del centro universitario de los altos.

Identificador: SGS05	
Prioridad: Alta	Fuente: Entrevista
Necesidad: Esencial	
Claridad: Alta	Verificabilidad: Alta
Estabilidad:	Alta
Descripción:	Se requiere una base de datos administrada y manejada posteriormente por encargados supervisores y equipo correspondiente para el uso medido de los datos que hayan sido insertados.

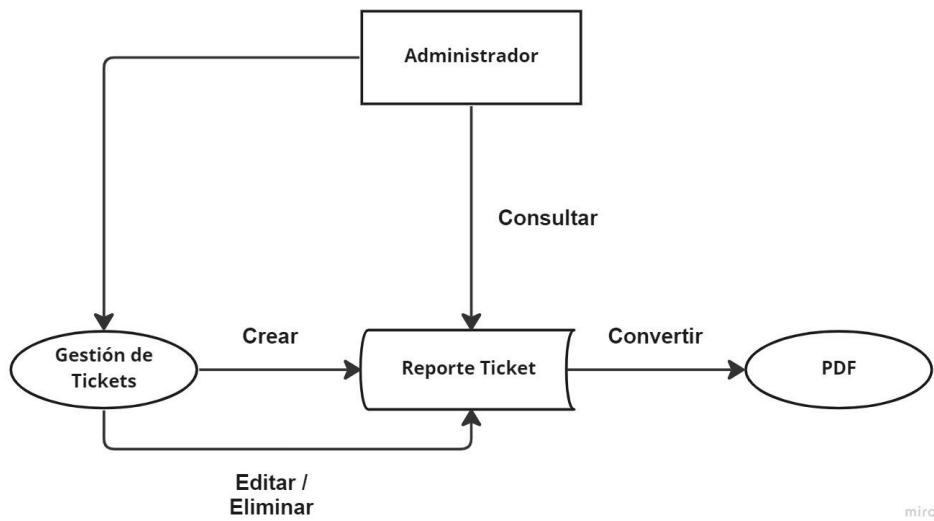
5.0 Diseño del sistema

5.1 Modelado del sistema



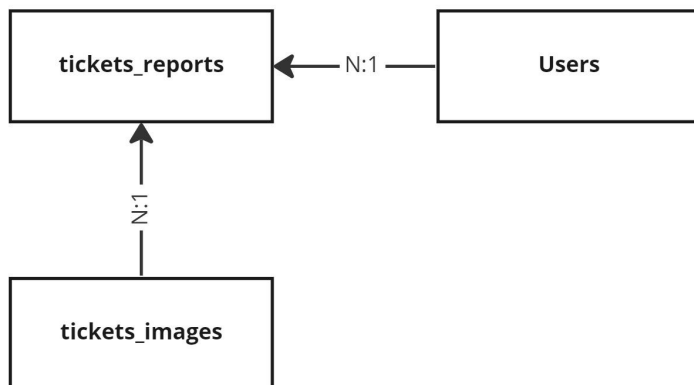
La sección principal de la plataforma son los tickets en forma de reporte, donde es directamente relacionado a la edición y almacenados en la base de datos, estos pueden ser generados por medio de un PDF y ser descargados o impresos en el momento que desee el usuario.

5.2 Flujo de datos



5.3 Diccionario de datos

Entidades	Atributos
Reporte de Tickets	Selección de abierto, cerrado o en ejecución. Selección de fecha de creación, horario de terminación, asignación de personal encargado, especificación de reporte, evidencias fotográficas.



5.4 Estructura de base de datos

A partir de la estructuración de la base de datos fluye principalmente a partir de la entidad de usuarios, estos usuarios cuentan con roles que se desglosan en permisos, las acciones que algún usuario realice adjudican a las demás partes de los reportes o secciones del sistema. Estas se parten de identidades que son:

En términos de la base de datos y su estructuración va directamente relacionado a estos reportes que se representan de la siguiente manera, determinado por permisos anteriormente mencionados con la inserción en las tablas de los reportes y dentro de los reportes se generan o bien se enlazan las evidencias de estos mismos reportes independientemente de cada uno:

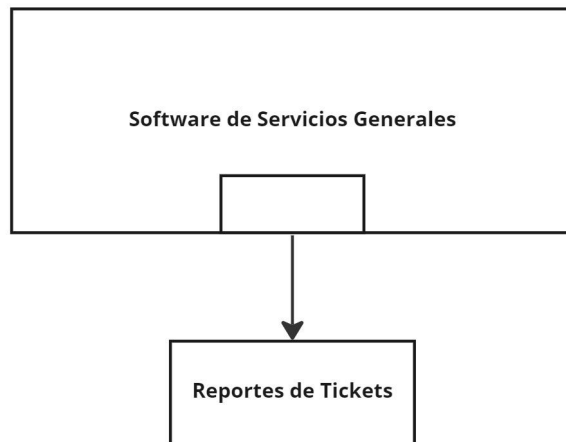
- ticket_reports
- ticket_images

Tablas de la base de datos con sus campos:

ticket_reports
id
user_report
type
date
ticket_report
employer
closet_at
date_finish
hour_finish
created_at
updated_at

ticket_images
id
ticket_id
created_at
updated_at
file

5.5 Diseño arquitectónico



miro

El diseño arquitectónico se encuentra dividido en módulos que se manejan de manera independiente, de esta manera las acciones del usuario serán relacionadas a cada módulo sin afectar a algún otro, por sí mismo se describen de la siguiente manera:

1. Tickets: Registros de Tickets, permite manejar los reportes que sean necesarios representados de forma de ticket, en los que especifica las acciones que deberán realizarse durante el día, cada reporte deberá ser independiente y ser tratado entre abierto, cerrado o en ejecución, a su vez, serán representados de manera estadística visualmente por medio de una gráfica, cada reporte permite la generación de pdfs.

5.6 Diseño de interfaces [4]

Interfaz web:

5.6.1 Sección de reportes

5.6.1.1 Reportes de Tickets

Lista de Tickets

Filtrar por fecha

Inicio

01/01/2021

Fin


06/11/2022

Filtrar

Crear nuevo

Fecha de Creación	Fecha de Expiración	Hora de Expiración	Asignado a	Descripción del Reporte	Estado de Ticket	ID	Opciones
2022-10-26	2022-08-30	20:20:32	Roberto	Limpiar área B2	Abierto	1	<div><div></div><div></div><div></div><div></div></div>
2022-04-18	2022-08-23	18:44:00	Ernesto	Reparación área 7B	Abierto	2	<div><div></div><div></div><div></div><div></div></div>
2022-09-28	2022-08-11	14:49:06	Armantina	Reparación proyector salón 8	Abierto	3	<div><div></div><div></div><div></div><div></div></div>
2022-08-11	2022-08-17	21:13:22	Carlos	Elaborar ticket físico	Abierto	4	<div><div></div><div></div><div></div><div></div></div>

Visualización móvil:



Lista de Tickets

Filtrar por fecha

Inicio01/01/2021

Fin06/11/2022

Filtrar

Crear nuevo

Fecha de Creación: 2022-10-26

Fecha de Expiración: 2022-08-30





Hora de Expiración: 20:20:32

Asignado a: Roberto

Descripción del Reporte: Limpiar área B2

Estado de Ticket: Abierto

ID: 1



Fecha de Creación: 2022-04-18

Fecha de Expiración: 2022-08-23





Hora de Expiración: 18:44:00

Asignado a: Ernesto

Descripción del Reporte: Reparación área 7B

Estado de Ticket: Abierto

ID: 2



5.6.1.1.1 Creación de Tickets

Creación Tickets

Fecha de generación

07/11/2022

Fecha de expiración

dd/mm/aaaa

Hora de expiración

--:-- ----

Estatus del Ticket

Elija el estatus del Ticket

Asignar a:

Nombre completo del empleado

Reporte

Describe su reporte

Evidencias

Elegir archivos

Sin archivos seleccionados

Crear Reporte

5.6.1.1.2 Visualización de reporte

Reporte de Ticket

Estado de Ticket

Abierto

Asignado a

Ernesto

Descripción del Reporte

Descongelar refrigeradores área médica

Fecha límite de Finalización


2022-11-08

Hora límite de Finalización

21:03:00

Evidencias

Imágenes de Evidencia



5.6.1.1.3 Editar reporte seleccionado

Editar Ticket

Fecha de expiración

08/11/2022

Hora de expiración

09:03 p. m.

Cambio de Estatus a:

Abierto


Asignar a


Ernesto

Reporte

Descongelar refrigeradores área médica

Evidencias





Elegir archivos

Sin archivos seleccionados

Confirmar Edición

5.6.1.1.4 PDF de reporte



Universidad de Guadalajara
Centro Universitario de los Altos
Secretaría Administrativa
Coordinación de Servicios Generales

Reporte de Ticket

Fecha de creación:	Estado Ticket:	Asignado a:	Descripción:	Fecha límite:	Hora límite:
2022-10-26	Abierto	Roberto	Limpiar área B2	2022-08-30	20:20:32

6.Codificación

Durante el desarrollo de software existió una comunicación constante en cuestión de programación de módulos de software. Esto, tomando en cuenta que son diferentes módulos dentro del programa hace una tarea mayormente sencilla de seguir y manejar de manera independiente en cualquier módulo en particular e individual, lo cual significa que el módulo que tenga en tarea de desarrollar algún integrante del equipo no se verá directamente afectado a las demás secciones de este.

Al realizar el programa de software, se acordó en realizarlo en un programa de lenguaje PHP por medio de Laravel y Javascript para la inserción de tablas, módulos independientes de codificación y realización de secciones ordenadas en los diferentes problemas e implementaciones que se desearan realizar. Esto en tareas de mantenimiento y estabilidad de programa puede resumirse en falta de pérdida de información y que se maneje de manera consistente adecuando el programa a futuro y en constante actualización por el lenguaje Javascript, acompañado del lenguaje PHP para la consulta de bases de datos y tablas.

Por medio de Laravel que fue el framework de código abierto principal de la aplicación, baja la complejidad del programa en cuestión de simplicidad como lo es un CRUD dentro del programa, para consulta, creación, edición y eliminación de datos a partir de los reportes almacenados en la base de datos donde puede manejarse un enrutamiento y motores de plantillas.

La aplicación final cuenta con ficheros de enrutamiento para secciones del programa mismo de forma independiente, estas secciones cuentan controladores base para el CRUD de archivos y otros para el manejo de datos. Al mismo tiempo, los controladores acceden a ese fichero para la consulta de la base de datos, utilizando middlewares para cumplir la estructura que sea necesaria y estipulada por el programa desarrollado.

Las vistas de datos son creadas a partir de PHP por medio de frameworks de Google para una visualización adecuada en la información, ahora, si se realizan consultas estas son apoyadas por medio de las secciones de helpers y plugins instalados por medio de Javascript.

La gestión de la tabla de datos son estructuradas por medio de modelos de datos, estos se administran de manera ordenada, estructurada y especificada para cada función de la tabla e inserciones de filas con los datos correspondientes a cada columna, a su vez se realiza el CRUD de manera correcta si estos son diseñados específicamente para estas funciones.

7. Pruebas

Se realizaron distintas pruebas, las cuales resultaron ser exitosas en la creación y manejo del software, para su correcta utilización actual para cumplir con los requerimientos deseados por el cliente. De ser así, se deduce que existe un análisis de datos y una realización de estrategias para el almacenado y pruebas que sean adecuadas a la estructura construida, a partir de ahí se desplegarán los siguientes puntos para su correcta ejecución.

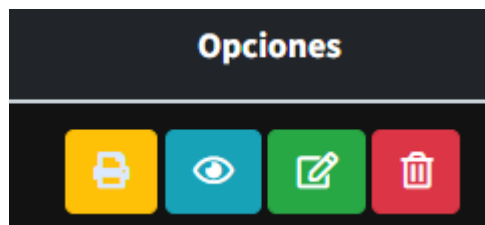
Elementos que serán puestos a prueba:

1. Módulo de reportes agua
 - a. Creación de reporte
 - b. Modificación de reporte
 - c. Visualización de reporte
 - d. Eliminación de reporte
 - e. Creación de pdf

7.1 Integridad de datos

Objetivo: Realizar inserciones por medio de pruebas, análisis de datos que sean acordes a la base de datos pertinente, en las que se respeten las llamadas de cada uno de ellos sin involucrar o afectar a otros módulos o tablas, es decir, que sea siempre protegida la identidad y privacidad de los datos que sean insertados o no.

Las pruebas fueron realizadas a partir de la base de datos, a partir de comandos SQL e inserciones por medio de la aplicación de PhpMyAdmin, para el análisis completo y correcto que fueron implementadas a partir de la especificación de las bases de datos, de esta manera se crearon ahora las tablas de almacenamiento con inserciones de datos. Retomando el concepto de CRUD [Imagen 7.1.1] se analiza y adquiere que la implementación de datos fue adecuada por medio de la gestión de datos e integridad satisfactoria y necesaria de ellos.



7.1.1 CRUD

Resultados: Los resultados fueron satisfactorios y cumplidos.

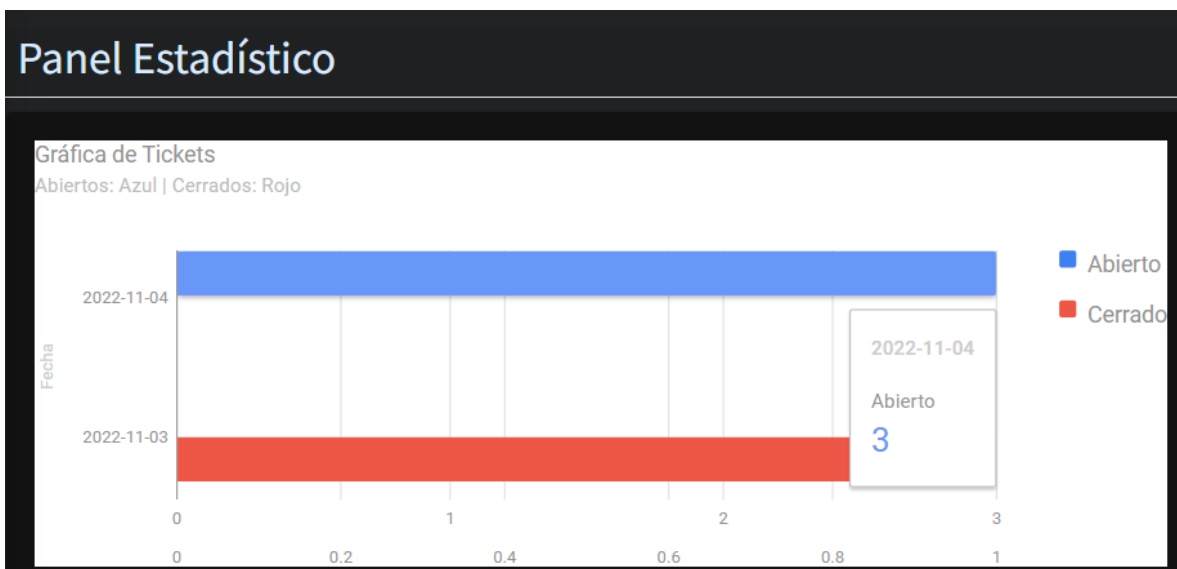
7.2 Funcionalidad

Objetivo: Con las pruebas de funcionalidad, se asegura que el trabajo de los requisitos funcionales, incluyendo la navegación del sistema, entrada de datos, restricciones por usuarios y permisos, procesamiento de los datos, y la implementación del procedimiento por parte de servicios generales.

Esta prueba se realiza basados en las técnicas de caja negra, en la que los casos de pruebas demuestran que las funciones son operativas, y que los datos entran de manera correcta y producen salida de datos de manera correcta.

Procedimiento: Se realizaron casos unitarios de uso y flujo, con datos válidos e inválidos para verificar que cuando se usan datos correctos se obtienen los resultados esperados, y en cuestión de datos inválidos obtener los mensajes correspondientes a los datos relacionados [Imagen 7.2.1], además de las excepciones del sistema para que este siga en funcionamiento, se revisó la navegación de cada una de las páginas del sistema para su correspondiente acceso.

Se revisaron los datos generados para las gráficas [6] generadas a través de los datos ingresados de los reportes [Imagen 7.2.2].



7.2.2 Gráficas

Resultados: En estas pruebas se encontraron que el módulo de compresor hacía un pequeño cambio en los datos generados, la corrección se realizó en la versión final del sistema.

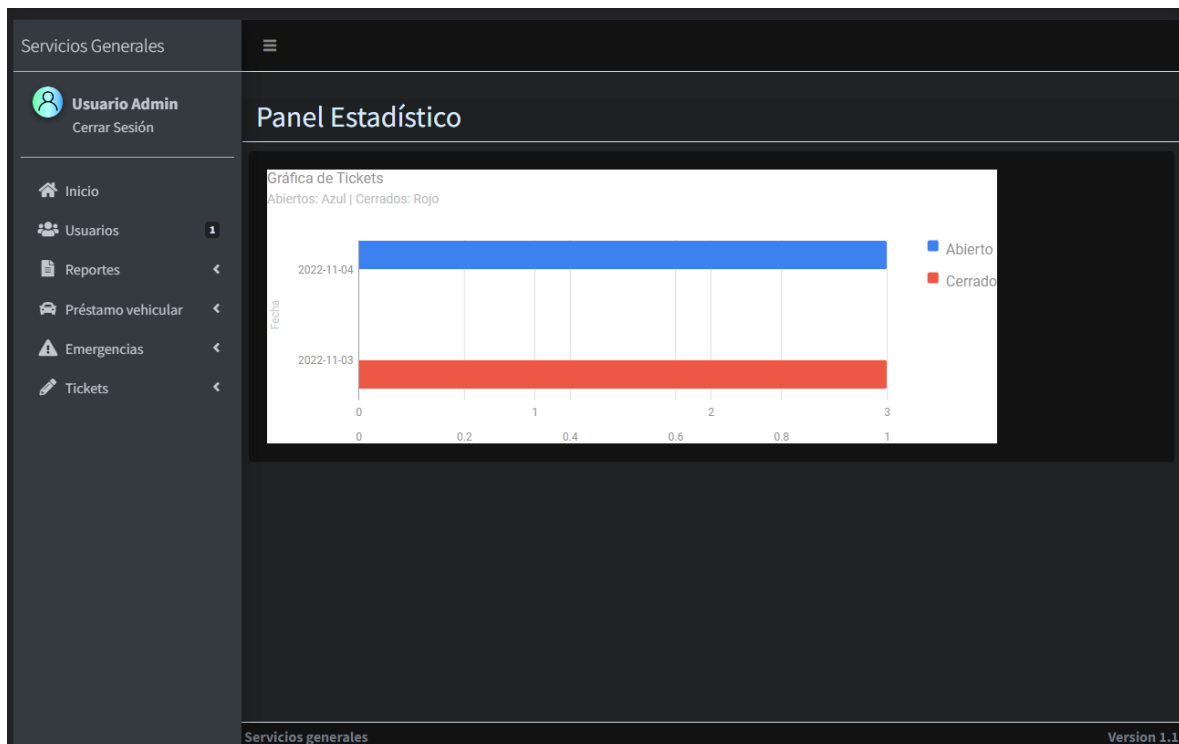
Fuera de lo anteriormente mencionado todas las demás pruebas realizadas resultaron correctamente en las salidas de datos correspondientes.

7.3 Interfaz de usuario [4]

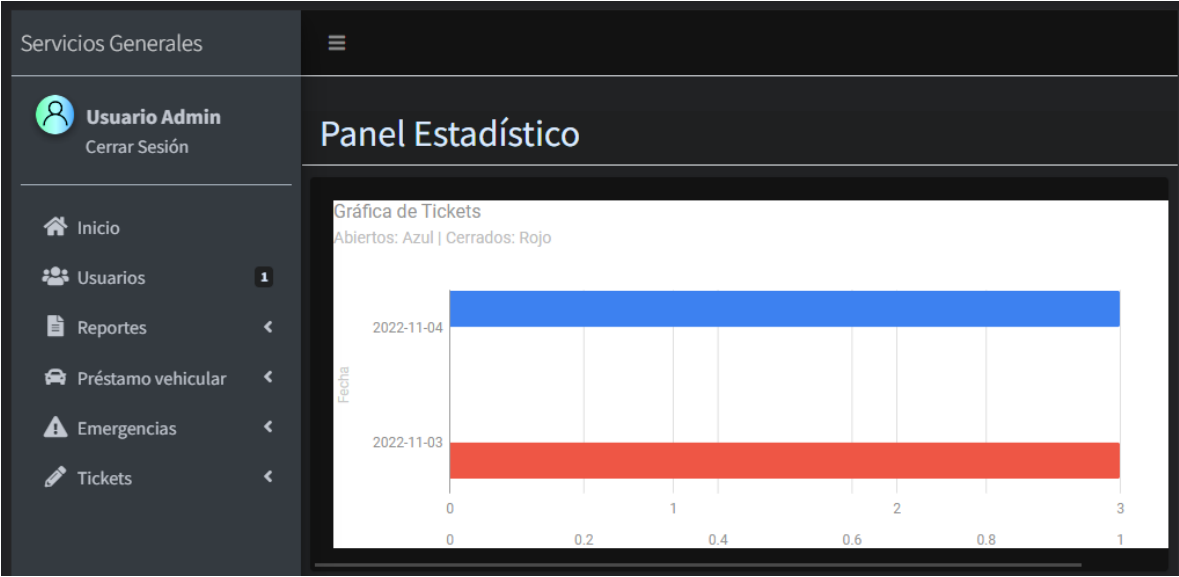
Objetivo: Dar accesibilidad y entregar al usuario una interfaz intuitiva e interactiva en la que pueda crear o manejar cambios dentro de la plataforma, con un diseño constante en la aplicación como son los botones, tablas y ventanas que ofrecen un correcto entendimiento del programa, al igual que tenga inmediato acceso en dispositivos móviles u otros dispositivos de cómputo.

Procedimiento: Estas pruebas fueron realizadas en diferentes dispositivos: Computadora de 14" [Imagen 7.3.2], celular de 6.5" [Imagen 7.3.3] y pantalla de pc de 27" [Imagen 7.3.1], además de que se trabajó estas pruebas en conjunto con los usuarios de Servicios Generales para tener los diseños adecuados en relación a los requerimientos visuales.

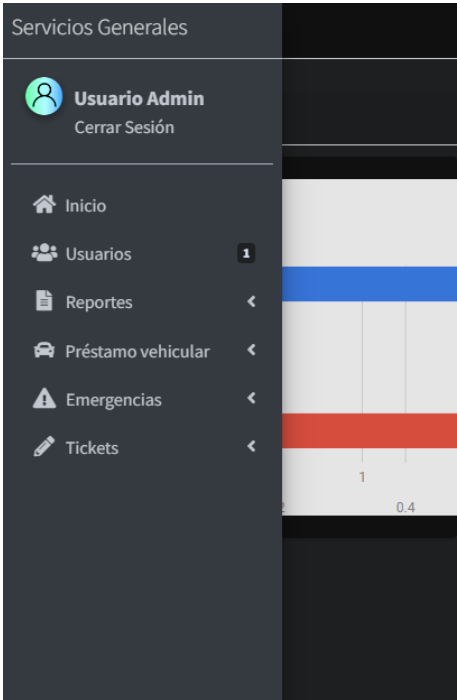
Resultados: La visualización es adecuada, se realizaron pruebas y entregas con éxito.



7.3.1 Resolución 27"



7.3.2 Resolución de 14"



7.3.3 Resolución de 6.5"

2. Conclusiones

En la realización de este proyecto y previa a esta se realizó un análisis acerca de las tareas que debían elaborarse, así como fue el análisis de producción de las labores que se realizan a diario dentro del centro universitario por parte del equipo de servicios generales, de esta manera al desempeñar este proyecto orientado al equipo de trabajo para un mejor control e implementación en la jornada laboral, enfocando las tareas e identificar cuáles de ellas necesitan mayor enfoque de gestión, implementación de recursos o equipo en métricas de tiempo al realizar una labor para su agilización.

Ahora, en propuesta del proyecto, la implementación de las metodologías ágiles que implican constante comunicación de equipo y técnicas de desarrollo de software en mejoras periódicas hace que la entrega de secciones resulte mayormente eficaz a comparación de otras estrategias tradicionales como sería el desarrollo cascada en la cual implica una única entrega de proyecto sin implementar cambios requeridos por el cliente, al igual que muestra mejoras ante el equipo buscando día con día nuevas técnicas de desarrollo de software y entrega temprana. Así mismo, el enfoque que se tiene a las tecnologías dirigidas a plataformas web y servidores hacen que se tenga acceso en cualquier dispositivo dando una respuesta inmediata a las tareas que se verán reflejadas en la plataforma.

Esto significa que anteriormente debería darse el reporte con materiales y presencia física o por medio de otros medios de comunicación, y al hacer esta plataforma digital y accesible para todos

pueden darse a conocer mejores resultados e impulso de progreso en el equipo de trabajo ofreciendo un mejor servicio ante la comunidad universitaria y al equipo mismo.

9. Referencias

[1]Oklobdzija, V.G. (Ed.). (2011). The Computer Engineering Handbook (1st ed.). CRC Press.
<https://doi.org/10.1201/9781420041545>

[2]"Configuration - Laravel - The PHP Framework For Web Artisans", *Laravel.com*, 2022. [Online]. Available: <https://laravel.com/docs/7.x/configuration>. [Accessed: 07- Dec- 2021].

[3]Mustelier Sanchidrian, D. Tesis: Método para determinar la complejidad de los requisitos funcionales de software. Ciudad de La Habana, Diciembre de 2014.

[4] (2020). AdminLTE Control Panel Template.URI:
<https://github.com/jeroennoten/Laravel-AdminLTE>.

[5] M. Ruiz Server, *Gestión de usuarios y control de acceso basado en roles. Un caso real*. Valencia: UNIVERSIDAD POLITÉCNICA DE VALENCIA, 2007.

[6] Quick Start | Charts | Google Developers", *Google Developers*, 2022. [Online]. Available: https://developers.google.com/chart/interactive/docs/quick_start. [Accessed: 07- Dec- 2021].