sgi

Sistema gestión de incidencias

Contenido

Idea y tecnologías utilizadas

Sistema de gestión de incidencias

La idea surge de observar mi entorno más cercano y observar las carencias existentes.

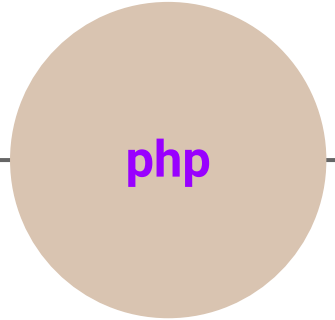
Tecnologías usadas

Front-end

Html
css
javascript
Ajax
jquery

Tecnologías usadas

Backend



Descripción
(Funcionalidades y organización)

Descripción

- El sistema de ticketing o de seguimiento de incidentes consiste en crear, gestionar y hacer un seguimiento de las incidencias y peticiones de los clientes. Este tipo de sistemas es muy utilizado por los departamentos de SAT(Servicio Asistencia Técnica).
- La gestión de incidencias bajo la metodología ITIL, tiene como objetivo prevenir o restaurar en el menor tiempo posible cualquier interrupción o retraso que afecte a la calidad del servicio (no planificada) y minimizar el impacto de las operaciones.

¿Que es un ticket?

Un ticket es donde se encuentra la información de la incidencia que el cliente comunica.

En él está la categoría a la que pertenece la incidencia, se realiza para tratar de comprender e identificar las causas por las que se dio una incidencia. También tenemos la severidad, que se clasifica en alta, media o baja, el título y descripción, para que se describa el problema surgido.

Funcionamiento

1. El sistema de tickets convierte las solicitudes de los clientes en documentos digitales que son conocidos como tickets de soporte.
2. Cada uno de los tickets de soporte aparece en la pantalla principal de la herramienta y son catalogados automáticamente como incidente **“**pendiente**”**.
3. Los Estados de una incidencia son:

Estados	Descripción
Pendiente	Cuando se crea un ticket se categoriza en este estado. Así permanece hasta que algún agente realice una acción.
Asignado	Significa que el ticket es atendido por el personal de soporte y está en proceso de resolución. En este estado es el personal de soporte quien tiene que realizar la próxima acción.
Resuelto	Indica que el personal de soporte ha resuelto el ticket y brindó una solución al cliente.

Funcionamiento

4. El personal técnico podrá atender la incidencia, si es así, el estado de la incidencia pasará a estado “asignado”. Podrá editar la incidencia si fuera necesario, si el problema al que se enfrenta no puede resolverlo podrá derivarlo al siguiente soporte, para que pueda ser atendida. Si el problema puede ser resuelto podrá marcarla como resuelta y su estado pasará a “resuelto”.

5. Si el cliente que creó el ticket marca como resuelta la incidencia está no volverá a abrirse.

6. El sistema de gestión de incidencias cuenta con un administrador, que es el encargado de crear los proyectos, asignar los proyectos al personal de soporte técnico y de crear incidencias si fuera necesario.

7. El equipo de soporte técnico, tendrá proyectos asignados, podrá atender incidencias que no tenga asignadas y podrá crear incidencias.

Organización

Roles	Descripción
Administrador	Gestiona el sistema ● Crea los proyectos ● Asigna proyectos al personal de soporte. ● Puede crear incidencias
Técnico de soporte	● Atiende incidencias ● Puede crear incidencias ● Puede derivar incidencias a otro soporte
Cliente	● Informa incidencias

Organización

Incidencias
Categorías
Programa malicioso
Pérdida de información personal
intrusión
Problema con hardware

Organización

Proyectos

Proyectos	Tipo soporte
proyecto a	● soporte telefónico ● envío técnico
proyecto b	● soporte telefónico ● envío técnico

Instalación

Instalación

Para que nuestro sistema funcione correctamente, necesitamos de un entorno para desplegarlo, para ello, tenemos docker.

- Necesitaremos el archivo docker-compose.yml, donde estarán todas las estructuras necesarias para el sistema.
- En el framework de laravel, tendremos que configurar algunos parámetros en la base de datos, el nombre que le hemos otorgado, el usuario, contraseña y el servidor.

- Nos iremos a archivo .env de laravel y modificaremos lo anteriormente mencionado.
- iremos a nuestra terminal y ejecutaremos la orden: **docker-compose up** ,para que se lleven a cabo las instrucciones indicadas en las estructuras de nuestro sistema, y así comenzar a utilizar la aplicación.

Guia de estilos y prototipado

Botones

botón indica una posible acción del usuario

Primary

Énfasis

A button can be formatted to show different levels of emphasis

Save

Colores

All these buttons are instances of the button.default component

Blue Red Green

Paleta colores

Colores que tiene la interfaz de usuarios

Blue	Purple	Orange	Grey
#6cb2eb	#f8fafc	#FFA500	#767676

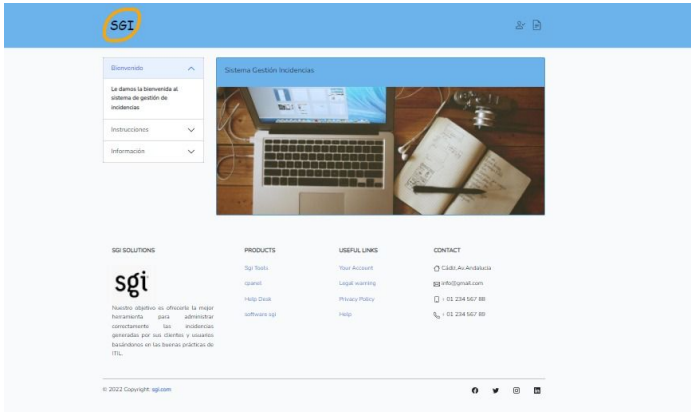
Tipografía

Las fuentes utilizadas son:
HTML: sans-serif
body: Nunito

Aa

Nunito
sans-serif

H1 The spectacle before us was indeed sublime.
H2 The spectacle before us was indeed
H3 The spectacle before us was indeed
H4 The spectacle before us was indeed
H5 The spectacle before us was indeed



Guia de estilos

Diseño

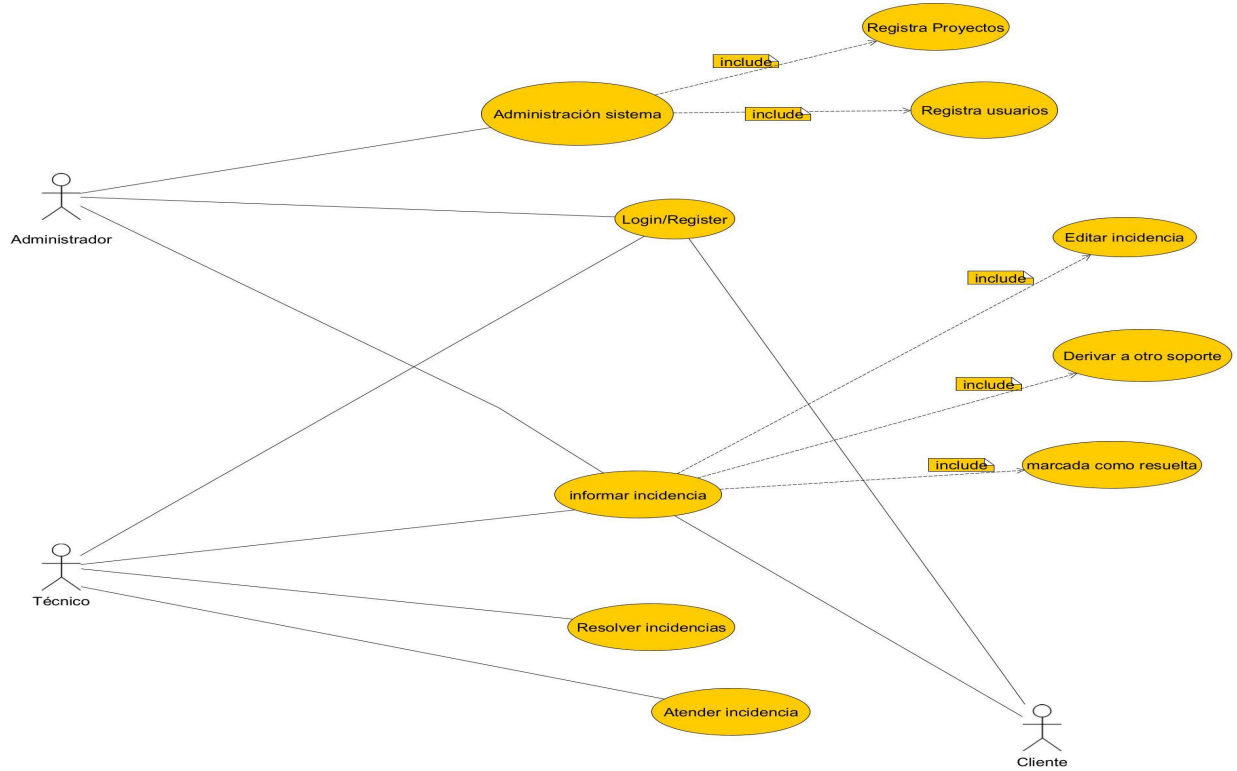
Requisitos funcionales y no funcionales

Los requisitos funcionales son aquellos que describen el comportamiento o función particular de un sistema cuando se cumple unas determinadas condiciones.

Los requisitos no funcionales son aquellos que definen las características de un sistema.

Requisitos Funcionales	Requisitos no Funcionales
La aplicación guardará los datos de los usuarios:administrador,técnico soporte y cliente. Como su nombre, email, imagen…	Los datos de la aplicación serán modificados por las personas autorizadas como por ejemplo el administrador
La aplicación creará, guardará y editará los distintos proyectos las categorías a la que pertenecen	Los datos registrados serán guardados en el gestor de base de datos, donde en un futuro se podrán realizar consultas.
La aplicación creará,guardará y editará los datos de los usuarios(técnicos)	La aplicación resuelve las incidencias.
La aplicación asignará a cada usuario(técnicos) con un proyecto	La aplicación deriva las incidencias a otro técnico de soporte, si el primero no puede resolverlo.
La aplicación creará, guardará, editará a través de los formularios de incidencias la información indicada en ellos.	
La aplicación mostrará una lista con las incidencias	

Diagrama Caso de uso



Desarrollo

Desarrollo

Dificultades encontradas

Detallamos alguno de los inconvenientes en la parte del backend:

- **Con el framework laravel, aunque ya tenía algunas nociones de cómo funciona, me he topado con algunas dificultades como el funcionamiento de los controladores, estructura y organización del framework.**
- **Con el lenguaje de php, que en ocasiones he tenido que buscar información sobre alguna sintaxis.**

En el front-end también ha existido sus dificultades:

- **Con vue.js, ya que no lo había utilizado antes, he tenido que documentarme un poco para familiarizarme con la sintaxis, y su uso.**

Aunque la mayor dificultad a la que me he enfrentado es la combinación del servidor y el cliente.

Secuencia de desarrollo aplicación web:

1. Tormenta de ideas, para la elección de nuestro proyecto.
2. Una vez que ya tenemos la idea, el siguiente paso será obtener los requisitos, que necesidades puede satisfacer nuestro sistema y qué problemas puede solucionar.
3. Definición de las funcionalidades de nuestro sistema.
4. El siguiente paso, es elegir que tipo de tecnología que vamos a utilizar

➔ ***Control de versiones***

Para el control de versiones y revisión de código, he utilizado github, donde he ido subiendo a mi repositorio las distintas versiones del proyecto, así me aseguraba todo esta a buen recaudo, o por si en algún momento realizaba modificaciones en el proyecto y luego estas no funcionaban volvía a una versión anterior.

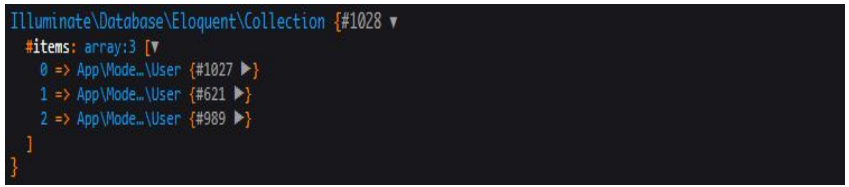
Pruebas

Pruebas

Laravel proporciona una función para depurar se llama Dump and Die(**dd()).** Es una función auxiliar para volcar el contenido de una variable al navegador. También detiene la ejecución del script.

Ej:Para imprimir los datos del usuario usando dd():

```
$users = User::all(); dd($users);
```



La función incorporada de php var_dump() función que muestra información estructurada sobre variables / expresiones, incluido su tipo de datos y el valor de la variable. La matriz y el objeto se exploran de forma recursiva con valores para mostrar su estructura.

Por ejemplo:

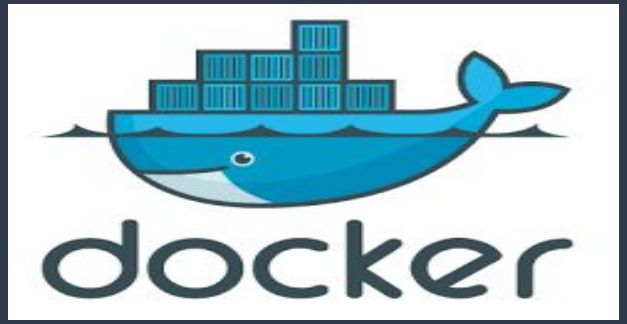
```
$users = User::find(1);

var_dump($users);
```

Nos muestra el objeto con todos los datos.

Despliegue

Despliegue

Plataforma de software que permite crear, probar e implementar aplicaciones rápidamente



- Definimos nuestro archivo, docker-compose.yml para laravel, donde tenemos definidos la versión, todos nuestros servicios y la base de datos.
- Iniciaremos nuestro archivo, ejecutando el comando: ***docker-compose up,*** para iniciar la pila de la aplicación.
- Podemos comprobar si está en ejecución con el comando: ***docker ps.***
- Si tenemos que realizar alguna operación sobre el sistema ejecutaremos el comando: ***docker exec -it (nombre:proyecto_servidor_1) /bin/bash***

Conclusiones

Conclusiones

El sistema cumple con los requisitos y realiza las funciones para las que fue programado.

Aunque admite las siguientes mejoras:

- Interfaz del usuario, la estética de la aplicación puede enriquecerse como por ejemplo: añadiendo imágenes dinámicas, gráficos…
- Otro aspecto mejorable de la aplicación es la accesibilidad web, para que puedan acceder a ellas un número mayor de usuarios. Se podría implantar el uso de letras de gran tamaño, diseños adaptativos, textos predictivos, asistentes a la navegación, etc

Conclusiones

- Con respecto a la interacción de clientes y personal de soporte, enriquecería a la aplicación un sistema de comunicación más dinámico, como un chat, donde las pequeñas consultas o dudas se podrían resolver rápidamente.
- Para que al equipo de soporte le resulte más fácil reconocer alguna incidencia, implementar subida de archivos e imágenes.

Conclusiones

- Para concluir, añadir la funcionalidad de emisión de informes en formato pdf o excel, por parte del equipo de soporte, para que así el cliente pueda descargarlo y ver entre otras aportaciones soluciones a su problema, por si vuelve a surgir un problema similar pueda ser capaz de resolverlo por sí mismo.

Bibliografía

- Libro Laravel, aprende a crear aplicaciones web desde cero.
  - Autor: José Lopez Quijado
  - Editorial : RC libros
- Libro Laravel, Curso práctico avanzado.
  - Autor: José Lopez Quijado
  - Editorial : RC libros
- Libro APRENDER PHP, MYSQL Y JAVASCRIPT
  - Autor: Robin Nixon
  - Editorial: Macondo

➔ https://laravel.com/docs/5.8

➔ https://vuejs.org/

➔ https://www.php.net/manual/es/index.php

Bibliografía

Autor: Teresa Melero Ligero