PROYECTO FINAL:

GESTOR CALIFICACIONES

JAIDER LAMES

JOAN BEJARANO

YOEL PANTOJA

UNIVERSIDAD DEL VALLE FACULTAD DE INGENIERÍA

TECNOLOGÍA EN SISTEMAS DE LA INFORMACIÓN SANTANDER DE QUILICHAO

2023

PROYECTO FINAL: GESTOR CALIFICACIONES

JAIDER LAMES

JOAN BEJARANO

YOEL SANTIAGO PANTOJA ESTRADA

INFORME

JOHN ROBERT CAUSIMANSE INGENIERO DE SISTEMAS DOCENTE

UNIVERSIDAD DEL VALLE FACULTAD DE INGENIERÍA

TECNOLOGÍA EN SISTEMAS DE LA INFORMACIÓN SANTANDER DE QUILICHAO

2023

Contenido

1. [INTRODUCCIÓN 4](#_bookmark0)
2. [DESCRIPCIÓN DEL PROYECTO 5](#_bookmark1)
3. [PLANIFICACIÓN 7](#_bookmark2)
4. [DISEÑO Y CODIFICACIÓN 8](#_bookmark3)
5. [CONCLUSIONES 14](#_bookmark4)
   1. INTRODUCCIÓN

Se realiza un proyecto de Gestor de Calificaciones utilizando como metodología de desarrollo eXtreme Programming (XP) dado que esta permite hacer entregas de software funcional continuamente, lo que va a dar ese voto de confianza en el cliente. Se puede adaptar a los cambios que se requieran o se soliciten en cualquier instancia, esta metodología nos promueve la comunicación entre el equipo que estuvo haciendo el correcto desarrollo del proyecto y los clientes, teniendo esto se puede llegar a una buena calidad en el software a entregar y que a su vez el cliente quede satisfecho con el producto final.

Como objetivos para este informe, se tiene hacer diferentes entregas de requerimientos funcionales y no funcionales; además se usará modelos de alta fidelidad con el software de Figma, para que de esta manera se puedan solicitar los cambios en momentos oportunos y que el cliente a su vez este al tanto de que cosas se van a realizar, cuando se harán y en que tiempo se van a realizar.

* 1. DESCRIPCIÓN DEL PROYECTO

La idea principal del proyecto que tiene como nombre Gestor de Calificaciones es realizar un aplicativo web, el cual debe tener funcionalidades similares a lo que se conoce como Campus Virtusles, en este caso, como funciones principales se tienen: subir archivos en diferentes formatos (Word, Excel, png, pdf, entre otros), realizar la descarga de los mismos, Inscribir estudiantes a las materias para diferentes trabajos para su calificacion, los usuarios tendrán distintos roles y cada uno de estos tendrá estas funcionalidades base, el usuario administrador podrá dar roles a los usuarios, crear nuevos usuarios y crear nuevos roles de ser necesario.

Para la correcta realización de este proyecto se necesito de tres personas que conocieran el correcto funcionamiento de las bases de datos y junto con Laravel, ayudados de la comunicación, destacamos los puntos de enfoque y conocimos las necesidades del cliente, las cuales se van a referenciar en las siguientes historias de usuario.

| **IDENTIFICADOR CASO DE USO:**  CU001 | | **NOMBRE:**  Compartir archivo público | |
| --- | --- | --- | --- |
| **REQUERIMIENTO FUNCIONAL ASOCIADO:** | | | |
| **ACTORES:**  Usuario | | | |
| **CASOS DE USO ASOCIADOS:**  **No aplica** | | | |
| **PRECONDICIÓN:**  El archivo debe existir y ser público | | | |
| **FLUJO NORMAL DE TRABAJO:**   1. El usuario navega al archivo que desea compartir. 2. El usuario selecciona la opción "Compartir archivo público". 3. El usuario selecciona el archivo y se comparte el archivo con otros usuarios. | | | |
| **POSCONDICION:**  El archivo es compartido con otros usuarios | | | |
| **FLUJO ALTERNATIVO:Si el archivo no es público, el sistema le pide al usuario que lo haga público antes de compartirlo.** | | | |
| **Paso** | **Actor** | | **Sistema** |
| **1** | **Usuario** | |  |
|  |  | |  |

| **IDENTIFICADOR CASO DE USO:**  CU002 | | **NOMBRE:**  Descargar archivo | |
| --- | --- | --- | --- |
| **REQUERIMIENTO FUNCIONAL ASOCIADO:** | | | |
| **ACTORES:**  Usuario | | | |
| **CASOS DE USO ASOCIADOS:**  **No aplica** | | | |
| **PRECONDICIÓN:**  El archivo debe existir y estar disponible para su descarga | | | |
| **FLUJO NORMAL DE TRABAJO:**   1. El usuario navega al archivo que desea descargar. 2. El usuario selecciona la opción "Descargar archivo". 3. El sistema descarga el archivo al dispositivo del usuario. | | | |
| **POSCONDICION:**  El archivo es descargado al dispositivo del usuario. | | | |
| **FLUJO ALTERNATIVO:** | | | |
| **Paso** | **Actor** | | **Sistema** |
| **1** | **Usuario** | |  |
|  |  | |  |

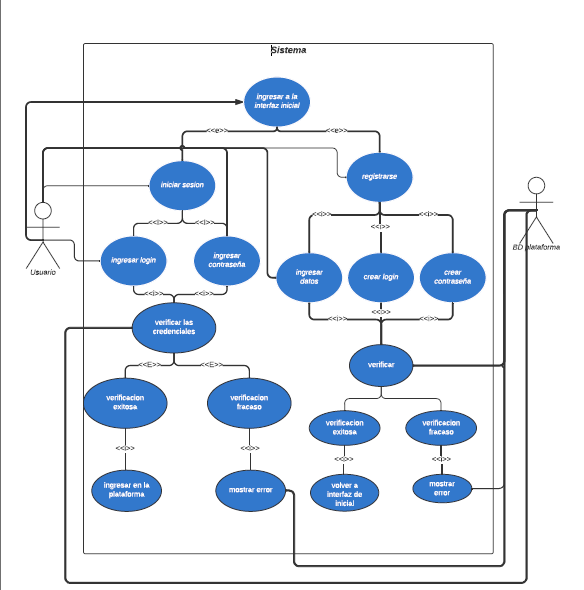
| **IDENTIFICADOR CASO DE USO:**  CU001 | | **NOMBRE:**  Generar reporte en formato PDF | |
| --- | --- | --- | --- |
| **REQUERIMIENTO FUNCIONAL ASOCIADO:** | | | |
| **ACTORES:**  Administrador | | | |
| **CASOS DE USO ASOCIADOS:**  **No aplica** | | | |
| **PRECONDICIÓN:**  El sistema debe tener datos para generar el reporte | | | |
| **FLUJO NORMAL DE TRABAJO:**   1. El administrador navega a la sección de reportes. 2. El administrador selecciona la opción "Generar reporte en formato PDF". 3. El sistema genera el reporte en formato PDF. 4. El sistema muestra el reporte generado al administrador. | | | |
| **POSCONDICION:**  El reporte es generado en formato PDF y mostrado al administrador | | | |
| **FLUJO ALTERNATIVO:** | | | |
| **Paso** | **Actor** | | **Sistema** |
| **1** | **Administrador** | |  |
|  |  | |  |

| **IDENTIFICADOR CASO DE USO:**  CU002 | | **NOMBRE:**  Generar reporte en formato gráfico | |
| --- | --- | --- | --- |
| **REQUERIMIENTO FUNCIONAL ASOCIADO:** | | | |
| **ACTORES:**  Administrador | | | |
| **CASOS DE USO ASOCIADOS:**  **No aplica** | | | |
| **PRECONDICIÓN:**  El sistema debe tener datos para generar el reporte | | | |
| **FLUJO NORMAL DE TRABAJO:**   1. El administrador navega a la sección de reportes. 2. El administrador selecciona la opción "Generar reporte en formato gráfico". 3. El sistema genera el reporte en formato gráfico. 4. El sistema muestra el reporte generado al administrador. | | | |
| **POSCONDICION:**  El reporte es generado en formato gráfico y mostrado al administrador. | | | |
| **FLUJO ALTERNATIVO:** | | | |
| **Paso** | **Actor** | | **Sistema** |
| **1** | **Administrador** | |  |
|  |  | |  |

Y los Requerimientos no funcionales para la correcta manipulación, para la migración en diferentes entornos de desarrollo:

|  |  |
| --- | --- |
| **Caso de Estudio No.** | Proyecto DIU |
| **Cliente** | Cliente x |
| Usuario | Usuario x |
| Requerimien to funcional | RF001: iniciarSecion correoElectronico, contraseña,  registrarUsuario(correoElectronico,crearContraseñ a) RF002: creacionUsuariosYPerfiles  crearUsuarios, crearPerfiles, borrarPerfil, borrarUsuario, funcionesUsurios,  funcionesPerfiles RF003: CRUD  opcionCrear, opcionActualizar, opcionBorrarArchivo y opcionConsultar  RF004: compartirArchivos  subirArchivo, compartirUsuarioRegistrado (correoElectronico),  notificarCompartirArchivo RF005: archivoPublico compartirUrlArchivo,descargarArchivo RF006: ReporteAdministrador  generarReportePDF,generarReporteGrafico |
| Requerimien to no  funcional | RNF001: usabilidad  navegacionIntuitiva, legibilidadInterfaz, coherenciaDiseño, adaptabilidadDiferentesDispositivos.  RNF002: rendimiento  tiempoRespuesta, capacidadCarga, escalabilidadOptimizaciónCódigo.  RNF003: seguridad  autenticación, autorización, cifrado, detecciónIntrusiones, gestiónRiesgos.  RNF004: mantenibilidad  modularidad, documentación, versionado, retrocompatibilidad.  RNF005: fiabilidad toleranciaFallos, redundancia,  recuperaciónAnteErrores, monitorizaciónSistema  Las versiones de el sofware de utilización:  Laravel 5, Bootstrap 5, Javascript, Xampp Desktop 3.0.2,  Windows 8 o Sup. |

CASOS DE USOS



Diagrama

Descripción generada automáticamente

* 1. PLANIFICACIÓN

Se entienden las necesidades del cliente y la manera en como quiere que su proyecto se desarrolle, se cuenta con aproximadamente un mes y dos semanas para la realización de este, en el transcurso de este tiempo vamos a definir las tareas para cada desarrollador y poder hacer entregas de adelantos al cliente. Dichos entregables se plantean uno cada tres semanas desde que se pidió la realización del proyecto.

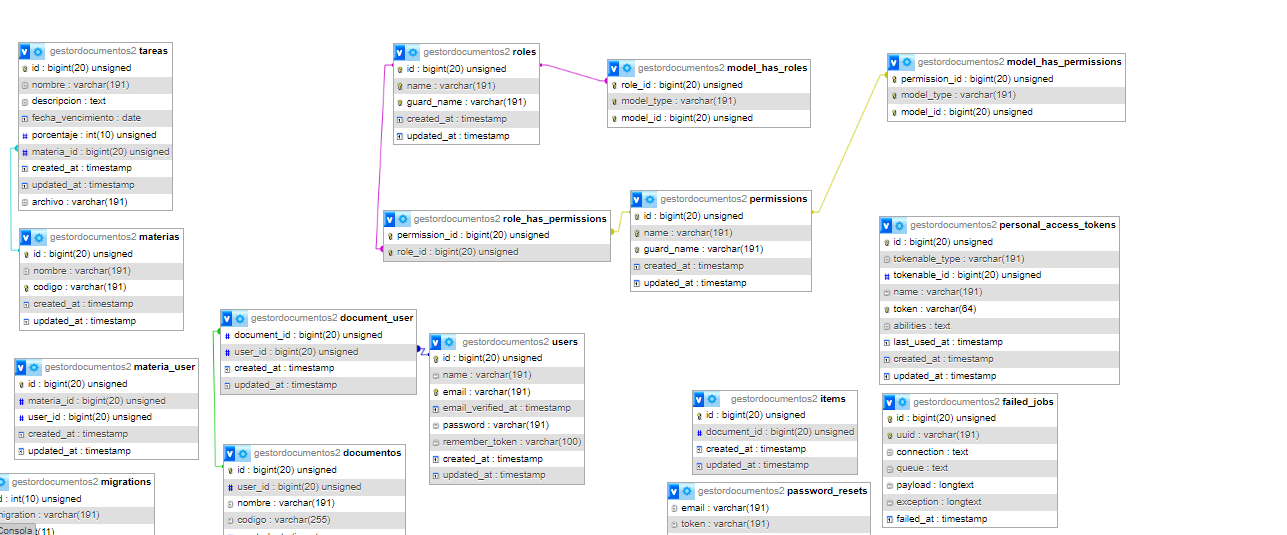
Como desarrolladores entregamos al usuario requerimientos no funcionales a tener en cuenta para que su aplicativo sea eficaz.

Tenemos como requerimientos no funcionales los siguientes:

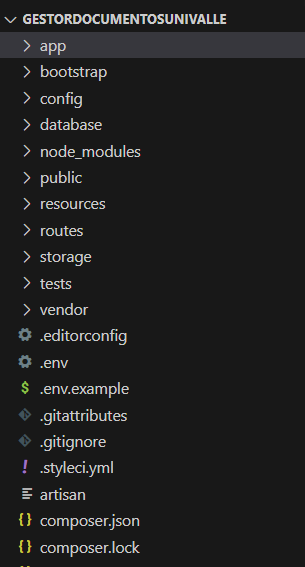
* El aplicativo web debe funcionar en cualquier navegador de internet.
* Requiere como mínimo 5 megas de internet para su correcto funcionamiento.
* El aplicativo fue creado usando php y laravel.
* El tiempo de espera para ingresar al aplicativo web con óptimas condiciones de internet (mencionadas anteriormente) no debe superar los 3 segundos.
* Para el inicio de sesión, el tiempo de verificación en la base de datos no debe demorarse más de 1.5 segundos.
* Enviar al usuario a la interfaz de inicio de sesión debe demorarse 1 segundo luego de llenar y confirmar el formulario para su registro.
* El tiempo para ingresar a la interfaz de las diferentes opciones debe ser de al menos 2 segundos.
* El tamaño máximo para subir un archivo será de 2Gb.
* La demora de carga o descarga de archivos va a depender del estado del internet y de su tamaño.
* Cada usuario tiene 16Gb para subir archivos.
  1. DISEÑO Y CODIFICACIÓN

El proyecto se realiza en Laravel junto con Php, se usa como gesto de bases de datos Xampp, como primer paso realizamos el diseño de la base de datos en donde se podrá visualizar de manera sencilla las relaciones que existen entre las tablas.

A continuación, se anexa el modelo entidad-relación de la base de datos, que contiene todas las tablas con sus respectivos atributos, y de la misma manera las relaciones que habrá entre ellas.

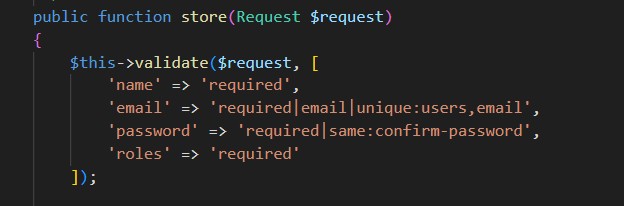


Se va a distribuir la forma de realizar el código en diferentes carpetas, las cuales tendrán en ellas diferentes funcionalidades, para que de esa manera sea mucho mas sencillo ubicarse dentro del código y buscar y solucionar errores en el momento dado.



Visualizando las diferentes carpetas tenemos que, por ejemplo, en la de nombre app se construye la aplicación, database nos permite realizar la conexión a la base de datos y realizar distintas consultas, en routes vamos a encontrar las diferentes rutas que existen para poder navegar dentro de la aplicación y en la carpeta public encontramos los archivos que contienen las dependencias a usar en el proyecto.

En la carpeta app se encuentran diferentes subcarpetas, entre ellas se tiene una llamada https, la cual va a contener los controladores de las diferentes vistas que trae consigo el aplicativo que se encuentra en desarrollo, la cual va a tener diferentes funciones, como ejemplo vamos a tomar la función de guardar el usuario que se está registrando.



Teniendo hecha la correcta conexión a la base de datos, se realiza la función que debe guardar los datos ingresados en los diferentes campos que se encuentren en la tabla de usuarios en la base de datos.

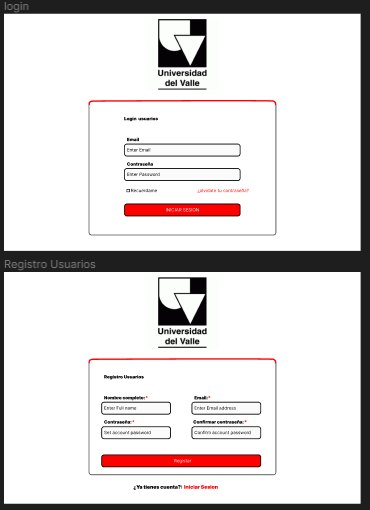
MODELAMIENTO DE ALTA FIDELIDAD

En el modelamiento utilizamos un software de modelamiento, que nos permite diseñar prototipos de alta fidelidad, para la presentación de un entorno visual compuesto

Fuente: https://www.figma.com/file/6pMepS6RNkFu4XAi02OL4I/proyecto\_login?type=design&node-id=0%3A1&mode=design&t=mkdczDuU7djFnw3l-1

Interfaz de usuario gráfica, PowerPoint

Descripción generada automáticamente



Fuente código proyecto:

Repositorio Proyecto (Código)

https://github.com/BlackDecker2/ProyectoCalificaciones.git

CONCLUSIONES:

Las referencias utilizadas, permiten crear un proyecto de calidad alta de software para el uso de componentes importantes para la USABILIDAD, NAVEGABILIDAD, ACCESIBILIDAD. Teniendo en cuenta los tipos de poblaciones que pueden hacer uso de la herramienta y el servicio.