**DESARROLLAR UNA APLICACIÓN WEB PARA APOYO AL PROCESO DE CONVOCATORIA DE DOCENTES CATEDRATICOS DEL INSTITUTO DE EDUCACION A DISTANCIA DE LA UNIVERSIDAD DE TOLIMA**

**EDILBERTO HERRERA NIÑO**

**ANDERSON MENDOZA AGUIAR**

**Trabajo de grado como requisito parcial para obtener el título de Tecnólogo En Gestión de Bases de Datos**

**Asesor**

**IVAN ANDRES BLANCO POLANIA**

**UNIVERSIDAD DEL TOLIMA**

**INSTITUTO DE EDUCACIÓN A DISTANCIA**

**TECNOLOGIA EN GESTION DE BASES DE DATOS**

**2019**

Este trabajo es dedicado quienes se esfuerzan por alcanzar las metas que se han propuesto

**Agradecimientos**

Agradecemos en primer lugar a Dios quien nos ha permitido alcanzar nuestras metas.

A nuestra familia que nos han brindado apoyo incondicional durante este proceso, al Ingeniero Iván Blanco que con su experiencia nos orientó para llevar a cabo este trabajo.

CONTENIDO

[1. RESUMEN 5](#_Toc11005289)

[2. INTRODUCCIÓN 6](#_Toc11005290)

[3. OBJETIVOS 7](#_Toc11005291)

[3.1 OBJETIVO GENERAL 7](#_Toc11005292)

[3.2 OBJETIVOS ESPECÍFICOS 7](#_Toc11005293)

[4. DISEÑO METODOLÓGICO 8](#_Toc11005294)

[4.1 METODOLOGÍA DE DESARROLLO 8](#_Toc11005295)

[4.2 FASES DE EJECUCION DEL PROYECTO. 9](#_Toc11005296)

[4.2.1 Preparación: 9](#_Toc11005297)

[4.2.2. Elaboración: 9](#_Toc11005298)

[4.2.4. Pruebas: 10](#_Toc11005299)

[5. MARCO TEÓRICO Y ANTECEDENTES 11](#_Toc11005301)

[5.1 TEORÍAS GENÉRICAS BASADAS EN LA INGENIERÍA 13](#_Toc11005302)

[5.1.1 Laravel 12](#_Toc11005303)

[5.1.1.1 Las migraciones de Laravel 13](#_Toc11005304)

[5.1.1.2 Los modelos 14](#_Toc11005305)

[5.1.1.3 Eloquent 15](#_Toc11005306)

[5.1.1.4 Middleware 15](#_Toc11005307)

[5.1.1.5 Laravel Collective 16](#_Toc11005307)

[5.1.1.6 Paquete Shinobi (Roles y permisos en laravel) 16](#_Toc11005307)

[5.1.1.7 SoftDelete o Eliminación lógica 16](#_Toc11005307)

[5.1.1.8 Laravel Auditing 16](#_Toc11005307)

[5.1.2 Bootstrap4 16](#_Toc11005307)

[5.1.3 JavaScript 17](#_Toc11005307)

[5.1.4 MySQL 17](#_Toc11005307)

[5.1.5 Git 18](#_Toc11005307)

[5.1.5.1 GitKraken 18](#_Toc11005307)

[6. DESARROLLO E IMPLEMENTACIÓN 19](#_Toc11005313)

[6.1 ANALISIS DE LA INFORMACION 19](#_Toc11005313)

[6.2 DOCUMENTACION Y DIAGRAMAS 19](#_Toc11005313)

[6.2.1 REQUERIMIENTOS 19](#_Toc11005313)

[6.2.2 CASOS DE USO 22](#_Toc11005313)

[6.2.3 SUB CASOS DE USO 23](#_Toc11005313)

[6.2.4 DESCRIPCIÓN DE CASOS DE USO 23](#_Toc11005313)

[6.2.5 DIAGRAMSAS DE CLASE Y SECUENCIA 35](#_Toc11005313)

[6.3 CODIFICACIÓN Y BANCO DE PRUEBAS 40](#_Toc11005313)

[6.3.1Codificación 4](#_Toc11005313)

[6.3.2 Verificación 40](#_Toc11005313)

[7. RESULTADOS ESPERADOS 43](#_Toc11005313)

[8. CONCLUSIONES 44](#_Toc11005314)

[9. ANEXOS 45](#_Toc11005315)

[10. BIBLIOGRAFÍA 46](#_Toc11005317)

# RESUMEN

En la actualidad, la recepción de documentos, gestión de la información, notificación a aspirantes del proceso de convocatoria docentes catedráticos del Instituto de Educación a Distancia de la Universidad del Tolima se realiza de forma manual, siendo esta una labor compleja y sensible para los funcionarios encargados de llevarla a cabo.

La falta de un sistema de información ocasiona demoras en el proceso y costos económicos para la universidad del Tolima debido a que la documentación de los docentes postulados para dicha convocatoria debe ser enviada de forma física a la sede principal desde los diferentes centros regionales, y la documentación de quienes no sean elegidos de nuevo debe ser enviada al centro regional en el cual fueron entregados, generándose costos de envió que asume la universidad y un desgaste operativo, tanto para los funcionarios como para la Universidad del Tolima.

En el proceso de convocatoria de catedráticos se evalúa mediante entrevista a los aspirantes cuya hoja de vida ha sido aprobada. Los resultados de dicha entrevista también deben ser ingresados y consolidados en un documento impreso, disponible en formato Excel en la página de la Universidad del Tolima.

Los docentes que actúan como jurados para realizar su labor solo cuentan con un formato físico que aunque diligencian por separado, se debe promediar la evaluación antes de ingresar la calificación final de cada participante en la convocatoria, ponderado consolidado que publica en la página web de la institución el área responsable del proceso.

Evidenciada esta problemática, se propone desarrollar una aplicación web que permita realizar esta actividad con el apoyo de un sistema de información, disminuyendo de esta manera la carga de trabajo para los funcionarios encargados de esta labor, optimizar el proceso, ahorrar recursos tanto en tiempo de trabajo como económicos ya que se sistematizarán algunas de las funciones del personal administrativo, permitiendo también implementar la estrategia cero papel promovida por el Ministerio de Tecnologías de la Información y las Comunicaciones (Min TIC) (Resolución 00277, 2013).

# INTRODUCCIÓN

Las aplicaciones web a lo largo de su consolidación y desarrollo se han empleado por diferentes sectores públicos y privados para ofrecerproductos y servicios, también permiten recopilar información a través de formularios la cual puede ser consultada y consolidada en cualquier momento, de esta manera se optimizan los recursos laborales dentro de las entidades que las emplean, ya que se disminuye la carga de trabajo de los funcionarios en un área en particular. (Prieto, 1996).

La aplicación web para apoyo al proceso de convocatoria docentes del Instituto de Educación a Distancia de la Universidad del Tolima, facilitará en gran medida este proceso, pues ya no será necesario que un funcionario se desgaste ingresando los datos de los postulados en cada convocatoria de forma manual sino que en el momento que se abra la convocatoria cada docente aspirante ingresara sus datos personales así como sus soportes a través de la plataforma. De esta manera no será necesario enviar documentación en forma física desde cada centro regional además facilitara el proceso de selección ya que los docentes que participen como jurados podrán subir su calificación para el docente postulado y la aplicación se encargara de promediar la calificación de igual manera permitirá que los docentes postulados puedan verificar mediante la plataforma si fueron seleccionados, adicional a eso la aplicación alimentara la base de datos de docentes de acuerdo con sus títulos y experiencia laboral.

# OBJETIVOS

3.1 OBJETIVO GENERAL

Desarrollar una aplicación web para apoyo al proceso de convocatoria docentes catedráticos del Instituto de Educación a Distancia de la Universidad del Tolima.

3.2 OBJETIVOS ESPECÍFICOS

* Determinar requerimientos funcionales y no funcionales para establecer los alcances y limitaciones de la aplicación
* Analizar las reglas de negocio requeridas para llevar a cabo el proceso de convocatoria de docentes catedráticos para el instituto de educación a distancia de la universidad del Tolima
* Diseñar una base de datos para almacenar información de catedráticos que prestan sus servicio al instituto de educación a distancia de la universidad del Tolima
* Desarrollar una aplicación web para la gestión de la información de los docentes que se postulen a las diferentes convocatorias realzadas por el instituto de educación a distancia de la universidad del Tolima
* Realizar la fase de pruebas con su respectiva documentación hasta el nivel beta para validar y verificar el correcto funcionamiento de la aplicación web.

# DISEÑO METODOLÓGICO

4.1 METODOLOGÍA DE DESARROLLO

El desarrollo de este proyecto se basó en la creación de una aplicación web que soporte acceso a bases de datos, para lo cual nos apoyamos en el motor de bases de datos MariaDB, además se hizo uso de los lenguajes de programación PHP con el uso del Framework Laravel, Javascript, Jquery así de esta manera se permitirá una mejor gestión del proceso de convocatoria docentes catedráticos para el Instituto de Educación a Distancia de la Universidad del Tolima.

En cuanto la metodología utilizada se optó por aquella que se adaptara más a nuestro medio, pues se trataba de elaborar una la aplicación de manera colaborativa por eso escogimos la metodología SCRUM.

SCRUM es una metodología de desarrollo que reduce a gran escala el proceso de desarrollo para satisfacer las necesidades de los clientes, debido a que se puede trabajar colaborativamente en equipo así de esta manera se obtiene el mejor resultado posible de un proyecto. (Menzinsky, 2016).

SCRUM permite la retroalimentación a medida que se va avanzando en el desarrollo del proyecto, es decir SCRUM está basado en un modelo de proceso empírico con respecto a las personas y basado en la auto organización de los equipos para lidiar con lo imprevisible y resolver problemas complejos inspeccionando y adaptando continuamente, por eso SCRUM permite redefinir requerimientos en caso de ser necesario.

Durante el desarrollo del proyecto es necesario hacer uso de Sprint que como es bien sabido es el corazón de SCRUM Sprint es considerado como un mini proyecto en donde el equipo de desarrollo focaliza todos sus esfuerzos en conseguir el objetivo definido en el Sprint planning(es donde se planifican las tareas a realizar en Sprint en curso), es importante tener en cuenta que cada Sprint se debe finalizar en el tiempo establecido pues por tratarse de un proyecto de 14 semanas si se retrasa la finalización de un Sprint se pone en peligro la culminación a tiempo del proyecto.

Antes de iniciar cada Sprint se hará una planificación rigurosa de que actividades se van a llevar a cabo y como se hará el trabajo que se ha elegido realizar, en este punto es importante asegurar que se hayan comprendido los propósitos del Sprint que está por iniciar para que este se lleve a cabo con absoluta normalidad.

De igual manera cada que se culmine un Sprint se debe evaluar para determinar los resultados obtenidos y en caso de surgir algún tipo de mejora o recomendación se debe tener en cuenta al momento de iniciar la siguiente etapa de trabajo.

También se lleva a cabo ya que es una buena práctica de la metodología seleccionada el Scrum Diario (reunión corta al inicio de cada jornada) donde se expondrán los avances y dificultades que se hayan presentado en la jornada anterior, así mismo se hará una retroalimentación o se redefinirán requisitos en caso de ser necesario.

4.2 FASES DE EJECUCION DEL PROYECTO

4.2.1 Preparación: Durante esta fase se hizo un análisis de las dificultades que se presentaban durante el proceso de convocatoria de docentes del Instituto de Educación a Distancia de la Universidad del Tolima (IDEAD) y se planteó una solución que facilitara tal proceso y disminuyera notablemente la carga de trabajo para el personal encargado.

Durante esta fase se realizaron las siguientes actividades:

* Recopilación de información bibliográfica referente al contexto del problema a abordar.
* Entrevista con la persona encargada del proceso de convocatoria de catedráticos del Instituto de Educación a Distancia de la Universidad del Tolima (IDEAD), en la cual pudimos apreciar el formato en el que se llevaba a cabo el proceso mediante un libro de Excel (ver anexo 1).
* Toma de requerimientos

4.2.2 Planificación: En esta fase se definieron las fases restantes para lograr la culminación del proyecto. Se realizaron las siguientes actividades

* Análisis de los requerimientos recopilados
* Definición del cronograma de trabajo
* Definición de los recursos económicos técnicos y tecnológicos para la elaboración de proyecto

4.2.3 Elaboración: El objetivo de esta fase fue la implementación de la aplicación, para ello se trabajó de forma colaborativa e incremental haciendo uso de las tecnologías seleccionadas para alcanzar los objetivos propuestos en cuanto a funcionalidad y diseño, para llevar a cabo esta fase se realizaron las siguientes actividades:

* Planificación de objetivos a cumplir
* Planificar las tareas para cumplir con cada requisito del sistema

4.2.4 Pruebas: El objetivo de esta fase es evaluar periódicamente la aplicación para detectar posibles errores de codificación y aplicar el correctivo necesario, en esta fase se llevaron a cabo las siguientes actividades:

* Determinar aspectos a evaluar
* Identificar las fallas
* Corrección de las fallas presentadas

# MARCO TEÓRICO Y ANTECEDENTES

Los sistemas de información responden a la satisfacción de necesidades de una organización o de un individuo o grupo, por tanto se diseñan para responder a objetivos concretos, y en su operación permanentemente se intenta comprobar el grado de eficiencia del mismo. Por tanto el elemento vital de un sistema de información es su uso.

Según el artículo cuarto del acuerdo 0023 de 2004 la evaluación de los aspirantes a profesores de catedra se hará sobre los siguientes parámetros a) hoja de vida que tendrá una puntuación máxima de treinta puntos b) una prueba de conocimiento que constara de la elaboración de un ensayo de dos cuartillas sobre el tema escogido al azar y de una exposición oral sobre el tema del ensayo la cual tendrá la siguiente puntuación: hasta treinta puntos por el ensayo y hasta cuarenta puntos para la exposición oral, se considerara ganador del concurso al aspirante que obtenga la mayor puntuación por encima de sesenta puntos(Universidad del Tolima, 2004 ).

Sin embargo este proceso de convocatoria de docentes catedráticos del instituto de educación a distancia de la universidad del Tolima se lleva a cabo de manera manual, convirtiendo dicho proceso en toda una odisea para los funcionarios encargados de llevar a cabo esta labor ya que es demasiado engorrosa, por esta razón surge la necesidad de automatizar este proceso haciendo uso de las tecnologías disponibles.

El manejo de datos e informaciones constituye uno de los aspectos más importantes para cualquier organización contemporánea. El manejo de información abarca diferentes actividades como la recolección, almacenamiento, recuperación, difusión hacia lugares y personas indicadas, así como el uso que de ellos se hace para varias actividades dentro de una organización(Villaderfrancos, 2004).

Habitualmente un sistema de información es grande y complejo pero esta complejidad no necesariamente obedece a aspectos de ingeniería sino a la diversidad de componentes que lo conforman y de las relaciones que están presentes en ellos.

Con la implementación de sistemas de información suelen lograrse ahorros significativos de mano de obra debido a que automatizan tareas operativas de la organización, dichos sistemas Tienen la propiedad de ser recolectores de información, es decir, a través de estos sistemas se cargan las grandes bases de información para su explotación posterior.

La implementación de un sistema de información dentro de una organización ofrece principalmente las siguientes ventajas:

* Mejora la eficacia, es decir se logra una mayor efectividad ya que aumenta la productividad.
* Mejora la eficiencia es decir se puede hacer más trabajo con menos recursos ya que reduce labores rutinarias y repetitivas.
* Aumenta la ventaja competitiva de la organización en general se caracteriza por permitir o habilitar un crecimiento del negocio los sistemas de información proporcionan una ventaja sobre nuestro rival.

Los beneficios de eficacia, eficiencia y aumento de las ventajas competitivas no se dan aislados, no son únicos. Un SI (Sistema de información) produce una combinación de estos beneficios. En la decisión de invertir en un SI es raro observar uno solo de los beneficios mencionados. En general los sistemas ofrecen algo de las tres ventajas aunque uno de ellos puede ser el foco, el motor o generador de la opción de compra (CIAMPAGNA, 2014).

Por otra parte es de gran importancia que un sistema de información lleve un control de los usuarios que realizan cambios o modificaciones, por eso es indispensable que se establezcan un sistema de auditoría, La auditoría de los sistemas de información se define como cualquier auditoria que abarca la revisión y evaluación de todos los aspectos (o de cualquier porción de ellos). Con frecuencia la palabra auditoría se ha empleado incorrectamente y se ha considerado como una evaluación cuyo único fin es detectar errores y señalar fallas; por eso se ha llegado a acuñar la frase "tiene auditoría" como sinónimo de que, desde antes de realizarse, ya se encontraron fallas y por lo tanto se está haciendo la auditoría. El concepto de auditoría es más amplio: no sólo detecta errores, sino que es un examen crítico que se realiza con objeto de evaluar la eficiencia y eficacia de una sección o de un organismo. (Huesca, A. G. (2000).Auditoria Informática 1).

5.1 TEORÍAS GENÉRICAS BASADAS EN LA INGENIERÍA

5.1.1 Laravel

**Laravel** es un framework de código abierto para desarrollar aplicaciones y servicios web con PHP 5. Su filosofía es desarrollar código PHP de forma elegante y simple, evitando el "código espagueti". Fue creado en 2011 y tiene una gran influencia de frameworks como Ruby on Rails, Sinatra y ASP.NET MVC. (Gallego, 2013).

laravel es muy fácil de utilizar debido a que su curva de aprendizaje es bastante rápida y está basado en el patrón de diseño modelo vista controlador, este framework cuenta con un sistema de plantillas denominado Blade el cual incluye directivas que se pueden usar desde archivos con extensión .blade .php. Para que las vistas puedan hacer uso de blade es necesario que los archivos tengan esta terminación con ello se podrán interpretar las directivas de blade.

La directiva @extends es propia de blade e indica una herencia de tal forma que determinada vista se construye sobre la vista padre de ahí que las vistas padre se conozcan como layouts las vista padre tiene por lo general código HTML pero adicional a eso van a tener algunas variables impresas sobre el código y algunas otras directivas.

La directiva @yield(‘content’) permite definir una sección que finalmente será definida por una vista que extienda de esta vista padre es decir se define secciones que posteriormente van a recibir un contenido

5.1.1.1 Las migraciones de Laravel

Las migraciones permiten hacer cambios sobre la base de datos las migraciones son archivos php que se interpretan luego y dan origen a cambios sobre la base de datos, es decir no se escriben consultas sql directamente para alterar la estructura de la base de datos acciones como “modificar columnas campos, tipos de datos, nombre de tablas, crear nuevas tablas etc.” este proceso no se requiere hacer manualmente sino que cada vez que vaya a haber un cambio sobre la base de datos se va crear una migración o varias, lo cual permite en cierta forma tener un control de versiones sobre nuestra base de datos e ir sabiendo cómo ésta ha ido cambiando con el paso del tiempo.

Dentro de la base de datos que nos genera laravel va a existir una tabla llamada migrations la cual lleva un control de las migraciones, gracias a esta tabla laravel sabe cuáles migraciones ya se han ejecutado y cuales no, entonces si creamos migraciones nuevas y hacemos uso del comando php artisan migrate solo se migraran las migraciones nuevas ya que laravel reconoce cuales migraciones son las que están pendientes, lo más común en laravel es tener una migración por cada tabla de tal manera que si se necesitan más entidades se crearan más tablas y cada una de ellas va a corresponder con una migración, pero lo recomendable es tener una migración por cada cambio

Las migraciones internamente ejecutan las consultas sql requeridas para el cambio dependiendo del motor de bases de datos que estemos usando.

Puesto que las migraciones de laravel son compatibles además de mysql también con postgres y sql server entre otros, por esta razón si eventualmente se quisiera cambiar de motor de base de datos solo se tendría que indicar en el archivo de configuraciones y las migraciones van a migrar los cambios sobre un motor de base de datos distinto.

5.1.1.2 Los modelos

Un modelo representa a una entidad de nuestro sistema e decir entre un modelo y una tabla va a haber una asociación por una convención que laravel define, solo se debe tener en cuenta que el modelo se escribe con la primera letra en mayúscula y en singular mientras que la migración se escribe todo en minúsculas pero laravel hace que esta quede en plural y funciona perfectamente para palabras que estén en ingles pero si se escribe la palabra en español y laravel lo convierte en plural de manera inadecuada el modelo no va a funcionar de manera adecuada ya que al aplicar la convención no va a lograr encontrar la tabla sin embargo hay formas de asociar un modelo con una tabla a pesar que su nombre no coincida

Los comandos para crear migraciones y modelos son los siguientes

php artisan make:migration

php artisan make:model

Sin embargo el comando Php artisan make:model + nombre –m crea el modelo y de paso la migración

Estos comando lo que hacen es generar una clase en la dirección donde corresponde cada usuario podría crear el archivo manualmente copiando el contenido de otro archivo y hacerlo funcionar pero lo adecuado es hacerlo mediante los comando ya que esto facilita el desarrollo

Se debe tener en cuenta que algunos modelos se van a relacionar, de tal forma que algunos modelos van a depender de la existencia previa de otros

Para ello se debe crear primero el modelo al cual se va a hacer referencia y luego al modelo que va a referenciar ya en la definición se la tabla se hace necesario la creación de una llave foránea

Primero se debe indicar a laravel que campo dentro de esa tabla será considerada llave foránea

Se hace con el uso del comando $table->integer('category\_id')->unsigned();

Y luego se le indica cual será la tabla de referencia con el siguiente comando

$table->foreign('category\_id')->references('id')->on('categories')

Aunque ya se hayan creado las llaves foráneas y en nuestra base de datos se refleje dicha relación aun laravel no reconoce la relación entre los modelos pues esto se debe indicar de forma explícita y esto se hace precisamente dentro de la clase modelo.

5.1.1.3 Eloquent

Las relaciones entre modelos nos va a permitir hacer uso de eloquent que es el ORM (Object Relational Mapping ”mapeo objeto relacional”)de laravel. (Laguna, 2013), lo que significa que nosotros vamos a poder interactuar con nuestra base de datos a través de objetos cada tabla va a estar mapeada como un modelo determinado de tal manera que si queremos añadir un nuevo registro a una tabla bastara con crear un nuevo objeto en nuestra aplicación con el nombre de la tabla y darle la orden de que se guarde, entonces cada vez que se realice alguna operación este efecto se va a ver reflejado sobre la base de datos si es que así se le indica al ORM de laravel, es decir si por ejemplo se tiene un modelo usuario se va a poder usar la clase usuario para acceder a la información de todos los usuarios así como poder editar dichos usuarios ya que un modelo es una clase a partir de la cual nosotros vamos a poder hacer operaciones, consultas a la base de datos.

5.1.1.4 Middleware

Un middleware se puede definir como un a capa intermedia, (Gallego, 2013), como un mecanismo que filtra las solicitudes HTTP que un usuario realiza, ya que cuando un usuario hace una petición a una ruta determinada esta se resuelve en un controlador de distintas maneras por ejemplo puede devolver datos, haciendo operaciones, devolviendo una vista, haciendo una redirección es decir se pueden ejecutar múltiples acciones en un controlador, pero antes que el flujo de control llegue al controlador se encuentran los middleware que van a evaluar ciertas condiciones, si estas se cumplen y son adecuadas el flujo de control pasa al controlador y la petición se resuelve normalmente, pero si cierta condición no se cumple entonces el middleware podrá hacer una redirección a medio camino y ejecutar acciones totalmente distintas a las esperadas por el usuario.

5.1.1.5 Laravel Collective

Laravel cuenta con muchas herramientas de desarrollo ágil que facilita a la vida a los desarrolladores con este Framework, uno de estos es Laravel Collective, que se hizo muy popular por su capacidad de ayudar mucho a manejar elementos HTML de manera limpia y sin reescribir mucho código.

Se compone de un conjunto de elementos para Formularios HTML tales como Labels, Botones, Listas de Formulario, Radio Buttons, Checkboxs, etc. Por otro lado cuenta con Anotaciones, Remote SSH, Iron Queue y Command Bus para otro tipo de tareas.

A pesar que Laravel Collective cuenta con otros servicios, esta herramienta se hizo popular más por el uso de los elementos para formularios mencionados.

5.1.1.6 Paquete Shinobi (Roles y permisos en Laravel)

Shinobi es un paquete que al igual que spatie/laravel-permission se dedica a la implementación de Roles y Permisos en Laravel,permite asociar usuarios a roles y permisos.

La función de roles y permisos en un sistema nos proporciona poder y podríamos decidir otorgar o quitar acceso a un módulo a cualquier usuario.

5.1.1.7 SoftDeletes o Eliminación Lógica

SoftDelete es una función que nos proporciona el ORM de laravel Eloquent la cual nos permite marcar los registros de la base de datos como eliminados sin eliminarlas realmente, Esto nos permite inspeccionar estos registros después para ser usados en auditorias.

5.1.1.8 Laravel Auditing

Este paquete ayuda a comprender los cambios en los modelos Eloquent, al proporcionar información sobre posibles discrepancias y anomalías que podrían indicar inquietudes comerciales o actividades sospechosas.

Laravel Auditing permite mantener un historial de cambios en el los modelos simplemente usando un rasgo. Recuperar los datos auditados es sencillo, lo que permite visualizarlos de varias maneras.

5.1.2 Bootstrap 4

Bootstrap es una biblioteca multiplataforma o conjunto de herramientas de código abierto para diseño de sitios y aplicaciones web. Contiene plantillas de diseño con tipografía, formularios, botones, cuadros, menús de navegación y otros elementos de diseño basado en HTML y CSS, así como extensiones de JavaScript adicionales. A diferencia de muchos frameworks web, solo se ocupa del desarrollo front-end.

Bootstrap es el segundo proyecto más destacado en GitHub y es usado por la NASA y la MSNBC entre otras organizaciones.

5.1.3 JavaScript

JavaScript  es un lenguaje de programación interpretado que permite crear efectos atractivos y dinámicos en las páginas web, una de las ventajas que ofrece JavaScript es que se pueden realizar solicitudes al servidor sin tener que recargar la página.

JavaScript nació con la necesidad de permitir a los autores de sitio web crear páginas que permitan intercambiar con los usuarios, ya que se necesitaba crear webs de mayor complejidad. El HTML solo permitía crear páginas estáticas donde se podía mostrar textos con estilos, pero se necesitaba interactuar con los usuarios.

Este lenguaje posee varias características, entre ellas podemos mencionar que es un lenguaje basado en acciones que posee menos restricciones. Además, es un lenguaje que utiliza Windows y sistemas X-Windows, gran parte de la programación en este lenguaje está centrada en describir objetos, escribir funciones que respondan a movimientos del mouse, aperturas, utilización de teclas, cargas de páginas entre otros.

Es necesario resaltar que hay dos tipos de JavaScript: por un lado está el que se ejecuta en el cliente, este es el JavaScript propiamente dicho, aunque técnicamente se denomina Navigator JavaScript. Pero también existe un JavaScript que se ejecuta en el servidor, es más reciente y se denomina LiveWire JavaScript

5.1.4 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,[1](https://es.wikipedia.org/wiki/MySQL" \l "cite_note-1)​[2](https://es.wikipedia.org/wiki/MySQL" \l "cite_note-2)​ y una de las más populares en general junto a Oracle y Microsoft SQL Server, sobre todo para entornos de desarrollo web

MySQL es muy utilizado en aplicaciones web, como Joomla, Wordpress, Drupal o phpBB, en plataformas (Linux/Windows-Apache-MySQL-PHP/Perl/Python), y por herramientas de seguimiento de errores como Bugzilla. Su popularidad como aplicación web está muy ligada a PHP, que a menudo aparece en combinación con MySQL.

MySQL es una base de datos muy rápida en la lectura cuando utiliza el motor no transaccional MyIASM, 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.

5.1.5 Git

Git es un sistema de control de versiones. Un sistema de control de versiones nos va a servir para trabajar en equipo de una manera mucho más simple y optima cuando estamos desarrollando software.

Con Git vamos a poder controlar todos los cambios que se hacen en nuestra aplicación y en nuestro código y vamos a tener control absoluto de todo lo que pasa en el código, pudiendo volver atrás en el tiempo, pudiendo abrir diferentes ramas de desarrollo, etc.

Vamos a poder trabajar en equipo de una manera muy sencilla y optimizada, de forma que si tenemos dos o tres personas trabajando en ciertas funcionalidades del proyecto y nosotros podemos estar trabajando en nuestra parte del código. Cuando acabamos de desarrollar nuestro código, utilizamos Git para mezclar los cambios con los otros compañeros. De forma que el código se mezcla de manera perfecta sin generar ningún tipo de fallo y de forma rápida.

También nos va a proporcionar un listado de los cambios (commits) y podemos volver atrás en el tiempo a cualquiera de esos cambios o commits.

Además tendremos la posibilidad de trabajar con ramas de desarrollo, que nos van a permitir desarrollar cosas que divergen mucho del programa principal.

5.1.5.1 GitKraken

Es una potente y elegante interfaz gráfica multiplataforma para git desarrollada con Electron. De forma muy sencilla podemos llevar el completo seguimiento de nuestros repositorios, ver ramas, tags, crear nuevos, todo el historial de nuestro trabajo, commits etcétera.

# DESARROLLO E IMPLEMENTACION

6.1 ANÁLISIS DE LA INFORMACIÓN

Como herramienta principal para el desarrollo de la aplicación se utilizó el framework Laravel, en razón a que es uno de los más avanzados para programar en php y tiene la mejor curva de aprendizaje, además permite desarrollar páginas con aspecto profesional usando código html e implementado el uso de bootstrap el cual simplifica el proceso de maquetación, sirviéndonos de guía para aplicar las buenas prácticas y los diferentes estándares para tener unas páginas web bien organizadas.

Una de las características por la cual se eligió Laravel es la facilidad para generar las bases de datos, pues solo basta con crear los modelos necesarios (clases) y hacer las migraciones para que laravel inserte el código SQL para que nuestra base de datos sea creada y mediante mysql se pueden apreciar los cambios en las diferentes tablas en tiempo real.

6.2 DOCUMENTACION Y DIAGRAMAS

6.2.1 REQUERIMIENTOS

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | R1 | **Estado** | Solicitado |
| **Nombre** | Registrar y modificar cat | **Prioridad** | Alta |
| **Descripción** | Permitir que se puedan registrar y modificar CATs | | |
| **Entrada** | Datos del cat que se va a registrar | | |
| **Salida** | Almacenamiento del cat en la base de datos | | |
| **Restricción** | Que el cat ya se encuentre registrado en el sistema | | |
| **Encargado** | Equipo desarrollador | **Solicitante** | Administrador del sistema |

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | R2 | **Estado** | Solicitado |
| **Nombre** | Registrar curso y programas | **Prioridad** | Alta |
| **Descripción** | Permitir que se puedan registrar y modificar cursos y programas | | |
| **Entrada** | Datos del curso que se va a registrar | | |
| **Salida** | Almacenar el curso en la base de datos | | |
| **Restricción** | Que el cuso ya se encuentre registrado en el sistema | | |
| **Encargado** | Equipo desarrollador | **Solicitante** | Administrador del sistema |

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | R3 | **Estado** | Solicitado |
| **Nombre** | Crear convocatorias | **Prioridad** | Alta |
| **Descripción** | Permitir que se puedan crear y modificar convocatorias con los respectivos requisitos para poder postularse | | |
| **Entrada** | Información correspondiente a cada convocatoria | | |
| **Salida** | Creación de una convocatoria | | |
| **Restricción** | No se pueden crear más de una convocatoria del mismo tipo | | |
| **Encargado** | Equipo desarrollador | **Solicitante** | Administrador del sistema |

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | R4 | **Estado** | Solicitado |
| **Nombre** | revisar aspirantes | **Prioridad** | Alta |
| **Descripción** | Permitir que se pueda revisar las postulaciones y preseleccionar los aspirantes que cumplan con los requisitos necesarios | | |
| **Entrada** | Datos de los aspirantes | | |
| **Salida** | Listado de aspirantes | | |
| **Restricción** | Ninguna | | |
| **Encargado** | Equipo desarrollador | **Solicitante** | Administrador del sistema |
| **Encargado** | Equipo desarrollador | **Solicitante** | Administrador del sistema |

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | R5 | **Estado** | Solicitado |
| **Nombre** | Registrar evaluación de aspirantes | **Prioridad** | Alta |
| **Descripción** | Registra la evaluación obtenida por cada aspirante | | |
| **Entrada** | Evaluación de los aspirantes | | |
| **Salida** | Registrar calificación de los aspirantes | | |
| **Restricción** | ninguna | | |
| **Encargado** | Equipo desarrollador | **Solicitante** | Administrador del sistema |

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | R6 | **Estado** | Solicitado |
| **Nombre** | Usuarios y roles | **Prioridad** | Alta |
| **Descripción** | Permitir que se puedan crear y modificar usuarios y roles | | |
| **Entrada** | Datos de usuarios | | |
| **Salida** | Registro de usuarios y roles | | |
| **Restricción** |  | | |
| **Encargado** | Equipo desarrollador | **Solicitante** | Administrador del sistema |

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | R7 | **Estado** | Solicitado |
| **Nombre** | Generar informes | **Prioridad** | Alta |
| **Descripción** | Genera los informes de perfiles, de preseleccionados y el informe final de quienes serán elegidos para ocupar los cargos disponibles | | |
| **Entrada** | datos del proceso de convocatoria | | |
| **Salida** | Informes de proceso de convocatoria | | |
| **Restricción** | ninguna | | |
| **Encargado** | Equipo desarrollador | **Solicitante** | Administrador del sistema |

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | R8 | **Estado** | Solicitado |
| **Nombre** | Postulación convocatoria | **Prioridad** | Alta |
| **Descripción** | Permitir que los aspirantes se postulen y puedan seguir el proceso de la convocatoria | | |
| **Entrada** | Registro de soportes del postulante | | |
| **Salida** | Registrar postulación | | |
| **Restricción** | Solo se pude postular una vez a cada convocatoria | | |
| **Encargado** | Equipo desarrollador | **Solicitante** | Administrador del sistema |

6.2.2 CASOS DE USO

|  |
| --- |
|  |

6.2.3. SUB CASOS DE USO

|  |
| --- |
|  |

6.2.4 DESCRIPCION DE CASOS DE USO

|  |  |
| --- | --- |
| **Nombre** | Registrar CAT |
| **Descripción** | Adiciona a la base de datos un nuevo CAT |
| **Actores** | Responsable proceso convocatoria |
| **Precondiciones** | El usuario debe autenticarse por medio de un usuario y contraseña validos que valide sus privilegios |
| **referencia** | De requerimiento R1 |
| **Flujo Normal** | 1. El sistema muestra un menú en el cual se visualizan las actividades que puede ejecutar el responsable del proceso de convocatoria  2. El responsable del proceso de convocatoria elige la opción Cat luego elige la opción crear  3. El sistema despliega un formulario para capturar la información del Cat  4. El responsable del proceso de convocatoria llena el formulario con la información del CAT  5. el sistema valida la información diligenciada  El sistema almacena la información del Cat en la base de datos |
| **Flujo Alternativo** | 1. La información del Cat no se diligencio en su totalidad por lo tanto se muestra un mensaje indicando que se diligencien todos los campos |
| **Poscondiciones** | La información del Cat se almaceno correctamente en la base de datos |

|  |  |
| --- | --- |
| **Nombre** | Modificar CAT |
| **Descripción** | Modifica la información de un CAT almacenado en la base de datos |
| **Actores** | Responsable proceso convocatoria |
| **Precondiciones** | El usuario debe autenticarse por medio de un usuario y contraseña validos que valide sus privilegios |
| **Referencias** | De requerimientos R1 |
| **Flujo Normal** | 1. El sistema muestra un menú en el cual se visualizan las actividades que puede ejecutar el responsable del proceso de convocatoria  2. El responsable del proceso de convocatoria elige la opción Cat luego elige la opción editar  3. El sistema despliega un formulario con la información del cat seleccionado  4. El responsable del proceso de convocatoria modifica los datos que estime convenientes  5. El sistema valida que la información del formulario haya sido diligenciada correctamente  6. El sistema actualiza la información del CAT |
| **Flujo Alternativo** | 1. La información del Cat no se diligencio en su totalidad por lo tanto el sistema muestra un mensaje que indica que se deben diligenciar todos los datos |
| **Poscondiciones** | El CAT se actualiza correctamente en la base de datos |

|  |  |
| --- | --- |
| **Nombre** | Registrar cursos |
| **Descripción** | Agrega diferentes cursos para cada programa |
| **Actores** | Responsable proceso convocatoria |
| **Precondiciones** | El usuario sistema debe autenticarse por medio de un usuario y contraseña validos que valide sus privilegios |
| **Referencias** | De requerimientos R2 |
| **Flujo Normal** | 1. El sistema muestra un menú en el cual se visualizan las actividades que puede ejecutar el responsable del proceso de convocatoria  2. El responsable del proceso de convocatoria elige la opción cursos luego elige la opción crear  3. El sistema despliega un formulario para capturar la información correspondiente al curso  4. El responsable del proceso de convocatoria llena el formulario con la información correspondiente al curso  5. El sistema valida la información ingresada  6. El sistema almacena en la base de datos el nuevo curso |
| **Flujo Alternativo** | 1. La información del programa no se diligencio en su totalidad por lo tanto se muestra un mensaje que indica que se deben completar los campos  2. El nombre del curso ya se encuentra registrado por lo tanto se muestra un mensaje que indica que el curso ya se encuentra registrado |
| **Poscondiciones** | El curso se almacena correctamente en la base de datos |

|  |  |
| --- | --- |
| **Nombre** | Modificar cursos |
| **Descripción** | Modifica la información de un curso almacenado en la base de datos |
| **Actores** | Responsable proceso convocatoria |
| **Precondiciones** | El usuario debe autenticarse por medio de un usuario y contraseña validos que valide sus privilegios |
| **Referencias** | De requerimientos R2 |
| **Flujo Normal** | 1. El sistema muestra un menú en el cual se visualizan las actividades que puede ejecutar el responsable del proceso de convocatoria  2. El responsable del proceso de convocatoria elige la opción Cursos luego elige la opción editar  3. El sistema despliega un formulario con la información del curso seleccionado  4. El responsable del proceso de convocatoria modifica los datos que estime convenientes  5. El sistema valida que la información del formulario haya sido diligenciada correctamente  6. El sistema actualiza la información del curso |
| **Flujo Alternativo** | 1. La información del curso no se diligencio en su totalidad por lo tanto el sistema muestra un mensaje que indica que se deben diligenciar todos los datos |
| **Poscondiciones** | El curso se actualiza correctamente en la base de datos |

|  |  |
| --- | --- |
| **Nombre** | Registrar programa |
| **Descripción** | Agrega programas disponibles en los CATs |
| **Actores** | Responsable proceso convocatoria |
| **Precondiciones** | El usuario debe autenticarse por medio de un usuario y contraseña validos que valide sus privilegios |
| **Referencia** | De requerimientos R2 |
| **Flujo Normal** | 1. El sistema muestra un menú en el cual se visualizan las actividades que puede ejecutar el responsable del proceso de convocatoria  2. El responsable del proceso de convocatoria elige la opción Programas luego elige la opción crear  3. El sistema despliega un formulario para capturar el nombre del programa  4. El responsable del proceso de convocatoria digita el nombre del programa y selecciona si está activo o no  5. El sistema valida los datos ingresados  6. El sistema almacena el nombre del programa |
| **Flujo Alternativo** | 1. No se digitar el nombre del programa por lo tanto se muestra un mensaje que indica que es necesario llenar este campo  2. se digita el nombre de un programa que ya existe entonces el sistema muestra un mensaje donde indica que el programa ya existe |
| **Poscondiciones** | El nombre del programa se almacena con éxito en la base de datos |

|  |  |
| --- | --- |
| **Nombre** | Modificar programa |
| **Descripción** | Modifica la información de un programa almacenado en la base de datos |
| **Actores** | Responsable proceso convocatoria |
| **Precondiciones** | El usuario debe autenticarse por medio de un usuario y contraseña validos que valide sus privilegios |
| **Referencias** | De requerimientos R2 |
| **Flujo Normal** | 1. El sistema muestra un menú en el cual se visualizan las actividades que puede ejecutar el responsable del proceso de convocatoria  2. El responsable del proceso de convocatoriaelige la opción Programa luego elige la opción editar  3. El sistema despliega un formulario con la información del programa seleccionado  4. El responsable del proceso de convocatoriamodifica los datos que estime convenientes  5. El sistema valida que la información del formulario haya sido diligenciada correctamente  6. El sistema actualiza la información del programa |
| **Flujo Alternativo** | 1. La información del programa no se diligencio en su totalidad por lo tanto el sistema muestra un mensaje que indica que se deben diligenciar todos los datos |
| **Poscondiciones** | El programa se actualiza correctamente en la base de datos |

|  |  |
| --- | --- |
| **Nombre** | Crear convocatoria |
| **Descripción** | Crear una convocatoria para ser consultada por los aspirantes |
| **Actores** | Responsable proceso convocatoria |
| **Precondiciones** | El usuario debe autenticarse por medio de un usuario y contraseña validos que valide sus privilegios |
| **Referencias** | De requerimientos R3 |
| **Flujo Normal** | 1. El sistema muestra un menú en el cual se visualizan las actividades que puede ejecutar el responsable del proceso de convocatoria  2. El responsable del proceso de convocatoria elige la opción Convocatorias luego elige la opción crear  3. El sistema despliega un formulario para registrar la información de la convocatoria  4. El responsable del proceso de convocatoria ingresa los datos concernientes a la convocatoria y deja la convocatoria en estado activo  5. El sistema valida los datos ingresados  6. El sistema almacena la convocatoria creada |
| **Flujo Alternativo** | 1. La información de la convocatoria no se diligencio en su totalidad por lo tanto el sistema muestra un mensaje que indica que se deben diligenciar todos los datos |
| **Poscondiciones** | La convocatoria se almacena correctamente en la base de datos |

|  |  |
| --- | --- |
| **Nombre** | Modificar convocatoria |
| **Descripción** | Modifica la información de una convocatoria almacenada en la base de datos |
| **Actores** | Responsable proceso convocatoria |
| **Precondiciones** | El usuario debe autenticarse por medio de un usuario y contraseña validos que valide sus privilegios |
| **Referencias** | De requerimientos R3 |
| **Flujo Normal** | 1. El sistema muestra un menú en el cual se visualizan las actividades que puede ejecutar el responsable del proceso de convocatoria  2. El responsable del proceso de convocatoria elige la opción Convocatorias luego elige la opción editar  3. El sistema despliega un formulario con la información de la convocatoria seleccionada  4. El responsable del proceso de convocatoria modifica los datos que estime convenientes  5. El sistema valida que la información del formulario haya sido diligenciada correctamente  6. El sistema actualiza la información de la convocatoria |
| **Flujo Alternativo** | 1. La información de la convocatoria no se diligencio en su totalidad por lo tanto el sistema muestra un mensaje que indica que se deben diligenciar todos los datos |
| **Poscondiciones** | La convocatoria se actualiza correctamente en la base de datos |

|  |  |
| --- | --- |
| **Nombre** | Crear detalle convocatoria |
| **Descripción** | Permite crear los requisitos y el perfil para postularse en la convocatoria |
| **Actores** | Responsable proceso convocatoria |
| **Precondiciones** | El usuario debe autenticarse por medio de un usuario y contraseña validos que valide sus privilegios |
| **Referencias** | De requerimientos R3 |
| **Flujo Normal** | 1. El sistema muestra un menú en el cual se visualizan las actividades que puede ejecutar el responsable del proceso de convocatoria  2. El responsable del proceso de convocatoria elige la opción Convocatorias luego la opción detalle  3. En el siguiente menú se selecciona la opción crear  4. se diligencian los datos requeridos y se guarda el datalle |
| **Flujo Alternativo** | 1. No se seleccionó ninguna convocatoria  2. no se selecciona ningún detalle |
| **Poscondiciones** | Se crea el detalle para una convocatoria |

|  |  |
| --- | --- |
| **Nombre** | Modificar detalle convocatoria |
| **Descripción** | Permite crear los requisitos y el perfil para postularse en la convocatoria |
| **Actores** | Responsable proceso convocatoria |
| **Precondiciones** | El usuario debe autenticarse por medio de un usuario y contraseña validos que valide sus privilegios |
| **Referencias** | De requerimientos R3 |
| **Flujo Normal** | 1. El sistema muestra un menú en el cual se visualizan las actividades que puede ejecutar el responsable del proceso de convocatoria  2. El responsable del proceso de convocatoria elige la opción Convocatorias luego la opción editar  3. se despliega un formulario con los detalles de la convocatoria  4. se modifican los datos necesarios y se guarda |
| **Flujo Alternativo** | 1. No se seleccionó ninguna convocatoria |
| **Poscondiciones** | Se modifica el perfil y requisitos para postularse en dicha convocatoria |

|  |  |
| --- | --- |
| **Nombre** | Revisar postulaciones |
| **Descripción** | Permite revisar la cantidad de aspirantes que se han postulado a una convocatoria |
| **Actores** | Responsable proceso convocatoria |
| **Precondiciones** | El usuario debe autenticarse por medio de un usuario y contraseña validos que valide sus privilegios |
| **Referencias** | De requerimientos R4 |
| **Flujo Normal** | 1. responsable del proceso de convocatoria  2. El administrador elige la opción Convocatorias luego la convocatoria  3. El sistema despliega un formulario con la información de los aspirantes para esa convocatoria |
| **Flujo Alternativo** | 1. No se seleccionó ninguna convocatoria |
| **Poscondiciones** | Se puede apreciar el listado de aspirantes para la convocatoria seleccionada |

|  |  |
| --- | --- |
| **Nombre** | Preseleccionar aspirantes |
| **Descripción** | Permite seleccionar los aspirantes que cumplan con el perfil requerido |
| **Actores** | Responsable proceso convocatoria |
| **Precondiciones** | El usuario debe autenticarse por medio de un usuario y contraseña validos que valide sus privilegios |
| **Referencias** | De requerimientos R4, R7 |
| **Flujo Normal** | 1. El sistema muestra un menú en el cual se visualizan las actividades que puede ejecutar el responsable del proceso de convocatoria  2. El responsable del proceso de convocatoria elige la opción Convocatorias luego la convocatoria  3. El sistema despliega un formulario con la información de los aspirantes para esa convocatoria  4. se elige la opción ver hoja de vida  5.se selecciona los aspirantes que cumplan con los requisitos solicitados para esa convocatoria  6. el aspirante es notificado mediante correo electrónico que ha sido preseleccionado |
| **Flujo Alternativo** | 1. No se seleccionó ninguna convocatoria  2. no se selecciona ningún aspirante |
| **Poscondiciones** | Se seleccionan los aspirantes que cumplan con el perfil para las convocatoria en mención |

|  |  |
| --- | --- |
| **Nombre** | Generar informes |
| **Descripción** | Genera los informes tanto en la etapa de preselección de los aspirantes como al finalizar la convocatoria |
| **Actores** | Responsable proceso convocatoria |
| **Precondiciones** | El usuario debe autenticarse por medio de un usuario y contraseña validos que valide sus privilegios |
| **Referencias** | De requerimientos R7 |
| **Flujo Normal** | 1. El sistema muestra un menú en el cual se visualizan las actividades que puede ejecutar el responsable del proceso de convocatoria  2. durante la fase de preselección cada aspirante es notificado vía correo electrónico si ha sido preseleccionado  3. en la fase final los aspirantes preseleccionados serán informados de sus resultados en el proceso y si ha sido seleccionados para ocupar el cargo |
| **Flujo Alternativo** |  |
| **Poscondiciones** | Los aspirantes son notificados de sus resultados en el proceso |

|  |  |
| --- | --- |
| **Nombre** | Registrar evaluación de los aspirantes |
| **Descripción** | Permite al evaluador registrar en el sistema la calificación obtenida por cada aspirante |
| **Actores** | Responsable proceso convocatoria |
| **Precondiciones** | El usuario debe autenticarse por medio de un usuario y contraseña validos que valide sus privilegios |
| **Referencias** | De requerimientos R5, R7 |
| **Flujo Normal** | 1. El sistema muestra un menú en el cual se visualizan las actividades que puede ejecutar el responsable del proceso de convocatoria  2. El responsable del proceso de convocatoria elige la opción Convocatorias luego la convocatoria  3. El sistema despliega un formulario con la información de los aspirantes preseleccionados  4. se elige la opción ver hoja de vida  5.se elige la opción calificar  6. se registra la calificación obtenida por el aspirante  7. el aspirante es notificado vía correo electrónico del puesto que ha ocupado en el proceso y si ha sido seleccionado para ocupar el cargo |
| **Flujo Alternativo** | 1. No se seleccionó ninguna convocatoria  2. no se selecciona ningún aspirante |
| **Poscondiciones** | Se seleccionan los aspirantes que cumplan con el perfil para las convocatoria en mención |

|  |  |
| --- | --- |
| **Nombre** | registrar usuarios |
| **Descripción** | Agrega usuarios al sistema tanto administradores como aspirantes |
| **Actores** | Administrador, Aspirante |
| **Precondiciones** | 1. Usuario administrador  1.1. El usuario debe estar autenticado en el sistema como administrador  2. Usuario aspirante  2.1. ingresar a la dirección web de la aplicación |
| **Referencias** | De requerimientos R6 |
| **Flujo Normal** | 1. Usuario administrador  1.1. El sistema muestra un menú en el cual se visualizan las actividades que puede ejecutar el administrador  1.2. El administrador elige la opción Usuarios luego elige la opción crear  2. Usuario aspirante  2.1. El sistema muestra un menú con las actividades que puede realizar un aspirante  2.2. El aspirante elige la opción registro  3. El sistema despliega un formulario para registrar la  información del usuario  4. Se deben ingresar los datos correspondientes al usuario  5. El sistema valida los datos ingresados  6. El sistema almacena el registro del nuevo usuario |
| **Flujo Alternativo** | 1. No se diligenciaron la totalidad de los campos por los tanto el sistema muestra un mensaje indicando que se deben diligenciar todos los campos  2. se intenta registrar un usuario que ya esta registrado por lo tanto el sistema muestra un mensaje de error donde indica que el usuario ya se encuentra registrado |
| **Poscondiciones** | El registro del usuario se culmina con éxito |

|  |  |
| --- | --- |
| **Nombre** | Modificar usuarios |
| **Descripción** | Modificar la información de un usuario almacenado previamente en la base de datos |
| **Actores** | Administrador del sistema |
| **Precondiciones** | El usuario debe estar autenticado en el sistema como administrador |
| **Referencias** | De requerimientos R6 |
| **Flujo Normal** | 1. El sistema muestra un menú en el cual se visualizan las actividades que puede ejecutar el administrador  2. El administrador elige la opción Usuarios luego elige la opción editar  3. El sistema despliega un formulario con la información del usuario seleccionado  4. El administrador modifica la información en los campos que estime convenientes  5. El sistema valida que la información del formulario haya sido diligenciada correctamente  6. El sistema actualiza la información del usuario |
| **Flujo Alternativo** | 1. la información del usuario no fue diligenciada en su totalidad por lo tanto el sistema muestra un mensaje informando que se deben diligenciar toda la información |
| **Poscondiciones** | La información del usuario se actualiza correctamente en la base de datos |

|  |  |
| --- | --- |
| **Nombre** | Crear roles y permisos |
| **Descripción** | Crear los roles para los usuarios para que rengan acceso al sistema según permiso |
| **Actores** | Administrador del sistema |
| **Precondiciones** | El usuario debe estar autenticado en el sistema como administrador |
| **Referencias** | De requerimientos R6 |
| **Flujo Normal** | 1. El sistema muestra un menú en el cual se visualizan las actividades que puede ejecutar el administrador  2. El administrador elige la opción Roles luego selecciona la opción crear  3. El sistema despliega un formulario con la lista de permisos para el sistema  4. El administrador elige el usuario y le da los permisos necesarios  5. El sistema valida que la información del formulario |
| **Flujo Alternativo** | 1. No se diligencio el formulario en su totalidad |
| **Poscondiciones** | Se crean permisos para los usuarios |

|  |  |
| --- | --- |
| **Nombre** | modificar roles y permisos |
| **Descripción** | Crear los roles para los usuarios para que rengan acceso al sistema según permiso |
| **Actores** | Administrador del sistema |
| **Precondiciones** | El usuario debe estar autenticado en el sistema como administrador |
| **Referencias** | De requerimientos R6 |
| **Flujo Normal** | 1. El sistema muestra un menú en el cual se visualizan las actividades que puede ejecutar el administrador  2. El administrador elige la opción Roles luego selecciona el usuario y elige editar  3. El sistema despliega un formulario con la lista de permisos para el sistema  4. El administrador elije el permiso que quiere darle a cada usuario  5. El sistema valida que la información del formulario |
| **Flujo Alternativo** |  |
| **Poscondiciones** | Se otorgan diferentes permisos a los usuarios |

|  |  |
| --- | --- |
| **Nombre** | Registrar postulación |
| **Descripción** | Permite al aspirante postularse en una convocatoria |
| **Actores** | Aspirante |
| **Precondiciones** | El usuario debe estar registrado en el sistema como aspirante |
| **Referencias** | De requerimientos R5, R7 |
| **Flujo Normal** | 1. El sistema muestra el listado con las convocatorias disponibles.  2. El aspirante elige la convocatoria a la cual desea postularse y hace click en aplicar.  3. El sistema despliega un en el cual se puede anexar la hoja de vida.  4. se elige la opción seleccionar archivo  5.se selecciona el archivo correspondiente en formato pdf y se hace click en la opción abrir  6. El aspirante hace click sobre el botón enviar hoja de vida |
| **Flujo Alternativo** | 1. El formato de archivo no es pdf  2. El aspirante ya se encuentra postulado para esa convocatoria |
| **Poscondiciones** | El aspirante registra su hoja de vida para ser tenido en cuenta en el proceso de preselección |

6.2.5 DIAGRAMAS DE CLASE Y SECUENCIA

|  |  |
| --- | --- |
| DIAGRAMA DE CLASES INGRESAR AL SISTEMA (ADMIN) | |
| DIAGRAMA DE SECUENCIA INGRESAR AL SISTEMA (ADMIN) |

|  |
| --- |
| DIAGRAMA DE CLASES REGISTRAR USUARIO EN EL SISTEMA |

|  |
| --- |
| DIAGRAMA DE SECUENCIA REGISTRAR USUARIO EN EL SISTEMA |

|  |
| --- |
| DIAGRAMA DE CLASES CREAR CONVOCATÓRIA |

|  |
| --- |
| Diagrama de secuencia crear convocatória |

6.3 CODIFICAIÓN Y BANCO DE PRUEBAS

6.3.1 Codificación: Esta etapa consiste en implementar o escribir cada uno de los requerimientos como un programa de computadora en un lenguaje de programación, convirtiendo cada tarea en instrucciones en el lenguaje de programación.

En esta etapa se hizo necesario el uso de la herramienta GIT ya que nos permite llevar un control de versiones y trabajar la codificación en distintas ramas o módulos y luego unirlas o fusionarlas mediante el merge (unión de dos o más ramas)

6.3.2 Verificación: La verificación es el proceso por el cual se comprueba que un diseño o producto funciona tal como se esperaba. Al programa que está siendo verificado se le llama producto y al conjunto de pruebas que se le aplican en la verificación se le llama banco de pruebas

En la siguiente imagen se muestra como los usuarios de la aplicación se pueden autenticar y acceder a la aplicación de acuerdo con su perfil, bien sea como administrador del sistema o como aspirante, ya que de acuerdo a su perfil llevara a cabo una tarea especifica

|  |
| --- |
|  |

Figura 1 inicio de sesión

El perfil del administrador muestra las siguientes opciones

|  |
| --- |
|  |

Figura 2 menú de opciones para el administrador

Al seleccionar la opción Cat se muestra el listado con la información de los Cats registrados en la base de datos

|  |
| --- |
|  |

Figura 3 listado de CATs registrados en el sistema

En esta vista es posible realizar la actualización de toda la información del CAT seleccionado también es posible consultar o eliminar un CAT de la base de datos.

Si se requiere agregar un nuevo CAT se selecciona la opción crear y se despliega un formulario para capturar de la información correspondiente

|  |
| --- |
|  |

Figura 4 formulario de captura de datos de CATs

La información del nuevo CAT se mostrara en el listado de CATs registrados en el sistema

# RESULTADOS ESPERADOS

Una vez culminado el proyecto se ofrece a la Universidad del Tolima y específicamente al Instituto de Educación a Distancia IDEAD, una aplicación web para apoyo al proceso de convocatoria de docentes catedráticos para el IDEAD, la cual facilita la realización de esta labor mediante el uso de tecnologías modernas, esta aplicación ayudara a disminuir los costos que implica el envió de documentación desde los diferentes centros regionales hacia la sede principal de la universidad del Tolima, también se optimizara el uso de papel ya que todo el proceso se realizara y se consolidara en archivos digitales.

# CONCLUSIONES

En el proceso de elaboración del proyecto el equipo centro sus esfuerzos en construir un producto de calidad que supliera las necesidades tecnológicas para mejorar el proceso de convocatoria de docentes catedráticos para el Instituto de Educación a distancia de la Universidad del Tolima, que permitiera consolidar la información digitalmente para reducir el consumo de papel.

Antes de desplegar la aplicación hubo todo un proceso análisis de requerimientos, estudio de factibilidad, implementación de un prototipo, diseño de un plan de pruebas que aunque el usuario final no vera, nos permitió establecer que el proceso de construcción de software es un proceso complejo y más aún cuando lo que nos interesaba era poder lograr que se cumplieran las necesidades para las cuales fue concebido, aquí jugo un papel muy importante la elección de una metodología de desarrollo clara la cual nos permitió alcanzar cada objetivo en el tiempo prestablecido, de esta manera se culminó con el desarrollo de un producto que brindara una experiencia agradable al personal encargado del proceso de convocatoria de docentes catedráticos del Instituto de Educación a Distancia de la Universidad del Tolima.

# ANEXOS

**Anexo 1 base convocatorias.xlsx**

**Anexo 2 base de datos tutores.xlsx**

**Anexo 3 Pruebas\_elixir\_2019\_10\_14.docx**

**Anexo 4 Manual\_Elixir\_v1.pdf**

**9. BIBLIOGRAFÍA**

Universidad, T. (2004). *Acerdo 023.* ibague.

Villardefrancos, M.(2004). Sistemas de informacio, Sistemas de informacion principios y aplicaciones.

Ciampagna, J.(2014). Ventajas de los sistemas de información en las organizaciones

Prieto, F. Zornoza, A. Peiro, J.(1997). Nuevas tecnologías de la Información en la empresa.

Menzinsky, A. (2016). Scrum Manager

Laguna, A. (2013). Desarrollo de aplicaciones web con la versión 4 del

framework Laravel para principiantes.

Gallego, A. (2013). Laravel 5.

Andreu,R. Ricart,J.( 1996). Estrategia y sistemas de información.

Bytheway,E.( 1998). Fundamentos de Sistemas de Información

Huesca, A. G. (2000).Auditoria Informática 1

9.1 Referencias web

<https://www.mininter.gob.pe/content/que-es-laravel>

Consultada el 15 de Junio de 2019 a las 15:30 horas.

<http://www.maestrosdelweb.com/que-es-javascript/>

Consultada el 20 de Junio de 2019 a las 20:30 horas.

<https://es.wikipedia.org/wiki/MySQL>

Consultada el 21 de Junio de 2019 a las 07:30 horas.

<https://es.wikipedia.org/wiki/Git>

Consultada el 15 de Agosto de 2019 a las 11:45 horas.

<https://victorroblesweb.es/2018/04/28/que-es-git-y-para-que-sirve/>

Consultada el 12 de Septiembre de 2019 a las 10:20 horas.