

**UNIVERSIDAD MAYOR DE SAN SIMÓN**

**FACULTAD DE CIENCIAS Y TECNOLOGÍA**

**CARRERA DE INGENIERIA DE SISTEMAS**

**SISTEMA DE MANEJO Y DIFUSIÓN DE INFORMACÍON DE LA FACULTAD POLITÉCNICA DEL VALLE ALTO**

***PROYECTO DE GRADO – ADSCRIPCIÓN***

***DESARROLLADO POR:***

*DORIS MIREYA TERCEROS OVANDO*

***TUTOR:***

*LIC. CORINA JUSTINA FLORES VILLARROEL*

***REVISOR:***

*ARQ. HUASCAR ARIEL RIVERA CASTELLÓN*

***COCHABAMBA – BOLIVIA***

***2017***

**DEDICATORIA:**

*Agradezco a la vida y a Dios, por darme la oportunidad de vivir y por estar en cada paso que doy, por fortalecer mi espíritu y corazón e iluminar mi mente para culminar este proyecto. Así también, agradezco a todas aquellas personas que han sido mi soporte y compañía durante todo el periodo de estudio.*

**FICHA RESUMEN**

La Facultad Politécnica del Valle Alto, es una unidad desconcentrada dependiente de la Universidad Mayor de San Simón, esta se encuentra a 50 Km. de Cercado en la provincia Punata.

Actualmente, cuenta con cinco carreras a nivel Técnico Superior, uno a nivel Técnico Medio y dos programas complementarios a nivel licenciatura. Interna a ella, se encuentra a Unidad de Tecnologías de Información Facultativa (UTI), la cual se encarga del manejo de información referente al estatuto docente, estatuto administrativo y estatuto estudiantil. El sistema administrativo fue estudiado y analizado, del cual se diagnosticaron ciertos inconvenientes que conllevan a una falta de acceso y difusión de información entre los diferentes estatutos orgánicos de la Facultad Politécnica del Valle Alto.

En este sentido, el trabajo nace de la necesidad que enfrentan los diferentes estatutos orgánicos, teniendo como propósito el desarrollo de un sistema de información web para el manejo y difusión de información, que permita mejorar el desarrollo de las actividades. Para el desarrollo e implementación del sistema se utilizó tecnologías Web. El método de recolección de datos fue la observación directa, la entrevista no estructurada, encuesta y revisión documental. Los resultados obtenidos con la implementación del sistema fueron la accesibilidad a información y acortar la brecha digital en cuanto al manejo de información, apoyando así, el proceso de difusión de información.

*CAPITULO 1*

1. ANTECEDENTES Y JUSTIFICACIÓN DEL PROYECTO.
   1. **INTRODUCCIÓN**

Laravel en un framework de código abierto para desarrollar aplicaciones 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.

Gran parte de Laravel está formado por dependencias, especialmente de Symfony, esto implica el desarrollo de Laravel dependa también del desarrollo de sus dependencias.

Laravel propone un desarrollo a base de “Routes con Closures”, en lugar de un MVC tradicional con el objetivo de hacer el código más claro.

* 1. **PRESENTACIÓN DEL PROYECTO**

Sabemos por experiencia lo importante que es la normalización de datos en cualquier aplicación. Debido a la falta de conocimientos sobre estándares y buenas prácticas de desarrollo de software se producen sistemas que muchas veces no pueden ser mantenibles en el tiempo, con código poco entendible, fuertemente acoplados, difíciles de incrementar funcionalidad, entre otras desventajas.

Un framework, es un conjunto estandarizado de conceptos, prácticas y criterios para enfocar un tipo de problemática particular que sirve como referencia, para enfrentar y resolver nuevos problemas de índole similar.

Actualmente, las instituciones exigen que se cumpla un mínimo de estándares en desarrollo de software, esto para lograr que las instituciones que utilicen frameworks reduzcan el tiempo de desarrollo, manejen estándares y buenas prácticas de desarrollo. Es así, como la Facultad Politécnica del Valle Alto (FPVA) unidad dependiente de la Universidad Mayor de San Simón (UMSS), patrocino un estudio de un sistema de información necesario para darse a conocer en la región. Con la participación de las distintas autoridades, se buscó un modelo para medir la actuación de las mismas. Con la experiencia y conocimientos aportados por los participantes se llegó a un producto final: El Sistema de Manejo y Difusión de Información de la Facultad Politécnica del Valle Alto (FPVA), este sistema nos permite difundir información referente a la FPVA, la cual debe ser accesible, actual y fácilmente modificable. El sistema cuenta con administración de usuarios, gestión de funciones y roles, manejo de publicaciones por facultad, carrera, materia y unidad de información, control de accesos, reportes de visitas al portal web y envío de mensajes informativos entre usuarios del sistema.

El Sistema de Información de la FPVA, se lo desarrollo en Laravel, este framework nos permite obtener un sistema estandarizado de fácil entendimiento y con una interfaz amigable al visitante.

Para el desarrollo del sistema se aplicó las fases de trabajo del método de las 6’D que es interactivo e incremental y se puede utilizar la Ingeniería Reversa para actualizar los modelos a partir del código creado; con el desarrollo del sistema se identificó las características Laravel implementando un modelo vista controlador (MVC); integrando Eloquent ORM con servicios RESTFUL.

La documentación se encuentra en base al perfil del proyecto que contiene los siguientes objetivos:

**OBJETIVO GENERAL**

* Desarrollar un sistema web que difunda información e interactúe con los distintos tipos de usuario, aplicando tecnología móvil, para asegurar el fácil acceso a los usuarios del sistema y de esta forma adecuarse a nuevas tendencias tecnológicas.

**OBJETIVOS ESPECIFICOS**

El sistema web cumplirá con los siguientes objetivos específicos:

1. Mantener informada a la comunidad docente – estudiantil por medio de la implementación de una página web.
2. Informar al plantel docente - estudiantil por carrera con el desarrollo de un portal propio.
3. Generar publicaciones científicas para la comunidad docente – estudiantil por carrera con la implementación de foros de discusión.
4. Publicar información para la comunidad docente – estudiantil mediante el desarrollo de un portal web de difusión de contenido.
5. Fusionar al Politécnico por medio de enlaces con otros portales web como el WEBSISS.

Para llegar a nuestro acometido con un adecuado desarrollo del proyecto se desglosa en los siguientes capítulos:

* **METODOLOGIA**

Es este capítulo recompilaremos elementos que nos permitan definir un marco de trabajo para el desarrollo de software, para lo cual se consideró el uso de la metodología de las 6’D.

* **TECNOLOGIAS**

Veremos las todas las tecnologías utilizadas para el desarrollo del sistema, como el framework Laravel, biblioteca de JavaScript como JQuery, lenguajes de programación AJAX, el framework de JS Bootstrap, entre otros.

* **UML 2.0**

Para este capítulo se planificara, distribuirá los objetivos y alcances del proyecto, mediante el modelamiento UML; de manera que los requisitos estén priorizados dependiendo al beneficio que aportan.

* **PLANIFICACIÓN Y ORGANIZACIÓN**

En el capítulo se pondrá en práctica el método de las 6’D como marco de trabajo para el desarrollo de software.

* **EVALUACION Y RESULTADOS**

Opinaremos sobre el desarrollo del tema del proyecto.

* 1. **SÍNTESIS**

El sistema de manejo y difusión de información de la Facultad Politécnica del Valle Alto está desarrollado en Laravel, que es un framework que propone desarrollo a base de “Routes con Closures”.

El sistema permite a la FPVA darse a conocer en la región. Difundiendo información de tipo informativa y/o científica para el estatuto docente, estatuto administrativo y estatuto estudiantil. Llegando a cumplir los objetivos planteados satisfactoriamente.

*CAPITULO 2*

1. METODOLOGIA
   1. **INTRODUCCIÓN**

En el desarrollo de software es importante considerar el estudio de las metodologías, técnicas de programación que nos ayuden en el desarrollo de software.

Por lo cual se considera que un método de ingeniería de software interactivo e incremental, donde los requerimientos y etapas evolucionan mediante el avance progresivo y la ingeniería reversa, para encontrar una solución objetiva al problema planteado.

* 1. **METODO DE LAS 6’D Y SUS CARACTERISTICAS**

Para llevar a cabo el desarrollo del sistema de Manejo y Difusión de Información se consideró el estudio de metodologías y técnicas de programación que nos colaboraron en el desarrollo de software.

En este sentido el Método de la 6’D es un método de tipo estructurado fácil de comprender y guía paso a paso hasta la solución del problema. Está compuesto de seis etapas, cada una de las cuales consta de una serie de pasos que se van modificando o ajustando, dependiendo del grado de complejidad del problema y las herramientas que se utilicen para su solución.

* + 1. **ETAPAS** 
       1. **ETAPA 01: DESCRIPCIÓN DEL PROBLEMA**

En esta primera etapa identificamos los problemas que deseamos resolver, dividiéndolos en módulos y tareas.

Una vez identificado los problemas, es necesario comprenderlos en su totalidad, es decir, entender que es exactamente lo que se deseamos resolver. Finalmente, se debe escribir un enunciado claro, concreto y conciso del problema a resolver.

* + - 1. **ETAPA 02: DEFINICIÓN DE LA SOLUCIÓN**

En la etapa 02, es necesario estudiar a fondo los problemas para poder solucionarlos, saber exactamente en que consiste y descomponerlo en cada una de sus partes, facilitando su comprensión y posterior solución.

* + - 1. **ETAPA 03: DISEÑO DE LA SOLUCIÓN**

Una vez definida una solución se procederá a diseñar la lógica, modelando y desarrollando algoritmos.

Para el modelado de la solución del problema se utilizara el Lenguaje Unificado de Modelado (UML 2.0), que es una herramienta utilizada para describir clases, objetos y sus relaciones.

* + - 1. **ETAPA 04: DESARROLLO DE LA SOLUCIÓN**

Para la etapa 04, se procederá a la codificación del problema.

La codificación involucra traducir los diagramas, las especificaciones de las clases y los pasos del algoritmo de cada método, en sentencias de un lenguaje de programación determinado.

* + - 1. **ETAPA 05: DEPURACIÓN Y PRUEBAS**

Luego de codificar el modulo y/o clases, fueron probados mediante su ejecución. Al realizar este paso puede surgieron diferentes tipos de errores, siendo lógicos o de sintaxis.

Para lo cual corregimos, anulamos, modificamos o creamos nuevas sentencias, volviendo a probar el sistema y continuamos con la corrección y pruebas hasta obtener la respuesta deseada.

* + - 1. **ETAPA 06: DOCUMENTACIÓN**

Finalmente, para esta etapa se recompiló toda la documentación generada en las etapas anteriores, la cual fue de utilidad para la elaboración del manual técnico. El tiempo dedicado a esta etapa fue de mucha ayuda para desarrollar buenos hábitos de desarrollo de software a nivel profesional.

La **Ingeniería Reversa** nos permitió crear o actualizar el modelo preliminar de Diseño de la solución (Etapa 03), para lo cual fue posible programar algunas partes que no estaban especificadas en la Etapa 03. A partir de esta solución, se actualizó los diagramas hasta llegar a la solución deseada. (Ilustración 2.1 Método de las 6’D)



* 1. **PROCESO**

Para el método de las 6’D, el proyecto se ejecutó por módulos. Cada módulo proporciona un resultado completo, es decir, un incremento al producto final que sea susceptible a ser entregado con un mínimo de esfuerzo al cliente cuando lo solicitó.

En el método de las 6’D, el proyecto desarrollado se ejecutó por la recolección de datos en bloques cortos y fijos, este se inició en la etapa 01 Descripción del problema tomando un lapso aproximado de un mes, cada iteración tuvo que proporcionar un resultado completo por modulo, la culminación de cada módulo fue un incremento al producto final.

El proceso partió de la lista de requerimientos priorizada del producto, que actuó como plan de proyecto. Por lo cual el cliente priorizo los módulos como objetivo balanceando el valor que le aportan y quedan repartidos en etapas. De manera regular el cliente pueda maximizar la utilidad de lo desarrollado y el retorno de inversión mediante la **ingeniería reversa** de los objetivos que realiza la etapa 05: Depuración de pruebas, es decir, que lo desarrollado si no llego a mostrar el resultado esperado, se tuvo que replantear el problema con la etapa 03: Diseño Solución, para que de esta manera llegar a los objetivos planteados y a la satisfacción del cliente.

* 1. **SÍNTESIS**

Para el desarrollo del sistema se empleó el método de las 6’D, este es de tipo estructurado fácil de comprender y guía paso a paso hasta la solución del problema.

El método de las 6’D cuenta de seis etapas: descripción del problema, definición de la solución, diseño de la solución, desarrollo de la solución, depuración - pruebas y documentación.

Los cambios propiciados durante el desarrollo del sistema se la realizo mediante la ingeniería reversa, llegando a cumplir los objetivos planteados.

*CAPITULO 3*

1. APLICACIÓN DE LA METODOLOGIA
   1. **INTRODUCCIÓN**

En este capítulo se muestra el análisis y diseño los diferentes módulos del sistema, de tal manera que este sea entendible.

Por lo cual se llevaron a cabo diferentes entrevistas al cliente para una mejor comprensión del producto que desea. El resultado del análisis de cada módulo fue el desarrollo de cada una de las etapas del modelo de las 6’D.

* 1. **ANALISIS**

De las diferentes entrevistas con el estatuto docente, estatuto administrativo y el estatuto estudiantil, se pudo recabar la siguiente información para el desarrollo del sistema: la Facultad Politécnica del Valle Alto tiene la necesidad de manejar y difundir información; si es para la difusión de información, el visitante del portal web puede ver información referente a la facultad, carrera y/o materia de tipo académica y/o científica. Por otra parte, el manejo de información, solo lo puede realizar usuarios del sistema.

Así también, los usuarios del sistema reciben información referente a las publicaciones, envían correos electrónicos entre usuarios del sistema. Pero no todos los usuarios tienen el mismo rol o tarea, ya que según la función o trabajo que realizan tienen un rol y tareas.

Por otro lado, la Facultad tiene una Oferta Académica (Carreras) y cada una de estas cuenta con Materias, el conjunto de materias forma una malla curricular por Oferta Académica a cargo de un coordinador de carrera. Cada carrera se maneja de forma independiente al igual que cada una de sus materias.

Las publicaciones pertenecen a un tipo de información; cursos, talleres, seminarios, eventos, entre otros. Un requisito es poder dar de baja alguna publicación inadecuada o que no cumpla con el propósito del portal web.

Como ya se mencionó el sistema se divide en módulos:

* + Módulo de administración de usuarios.
  + Módulo de administración de funciones.
  + Módulo de administración de la facultad.
  + Módulo de administración de publicaciones.
  + Módulo de administración web.
    1. **Módulo Administración Usuarios.**
* ***Etapa 01 – Descripción del Problema***

**Enunciado:**Desarrollar un módulo que nos permita administrar usuarios, donde el usuario tenga un rol dentro el sistema. Asignando al usuario funcionalidad y tareas.

* ***Etapa 02 – Definición de la Solución***

**Resultado Deseado:**Almacenar y manipular los datos de los usuarios de la facultad. Asignar un rol, función y tarea.

**Datos Necesarios:**id, name, email, password, tipo, educación, biografia, estado, log, id\_funcion, función, id\_tarea, tarea, vista, id\_rol, rol, estado, created\_at, update\_at, id\_usuario\_tarea, id\_tareas\_rol.

**Procesamiento:** El usuario se registra en el sistema y se almacenara en una tabla denominada Users, donde también interactúan las tabla Rol, Tarea y Función. Se desarrollaran los métodos para poder manipular los datos almacenados.

* ***Etapa 03 – Diseño de la Lógica***

**Nombre del Proyecto:**ProyAdminUsers

**Diagrama de Paquetes:**



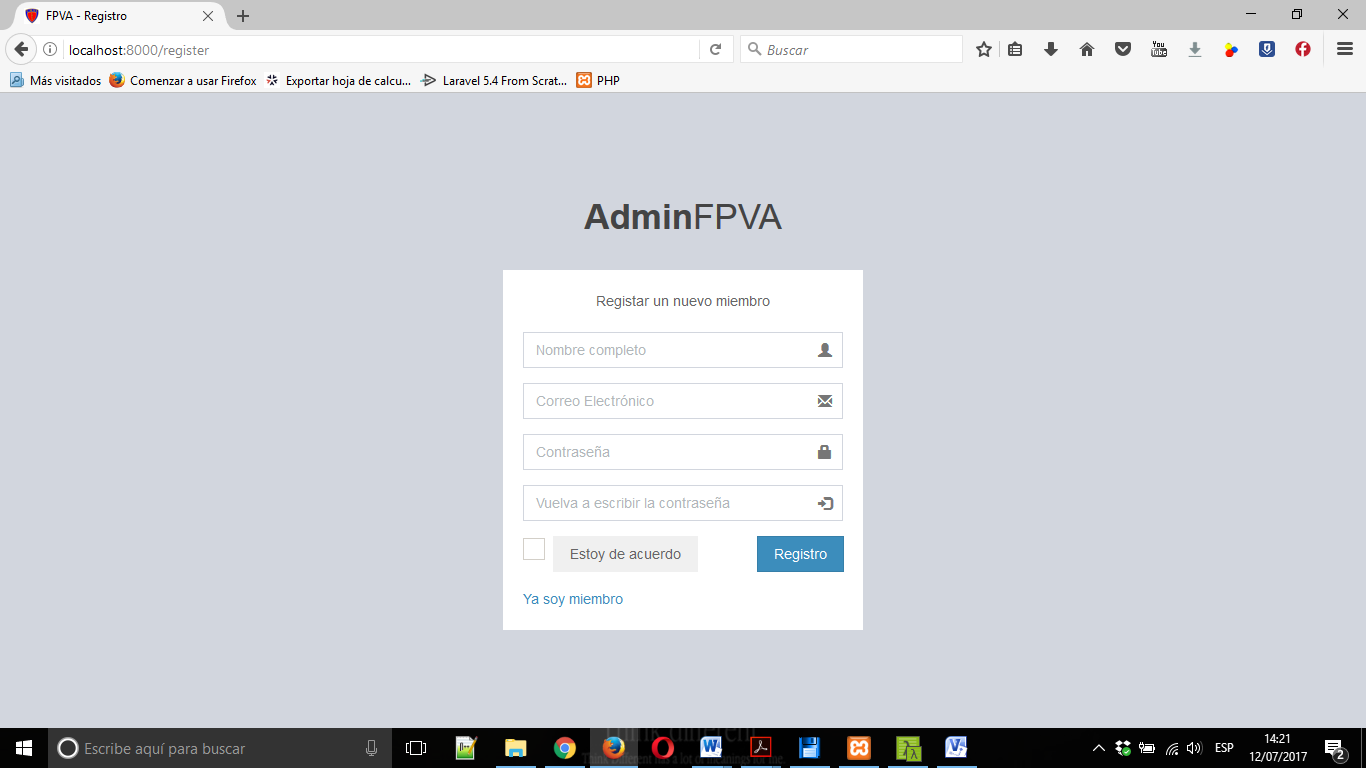
Figura 3.1 Modulo Administración Usuarios

**Diagrama de Clases:**

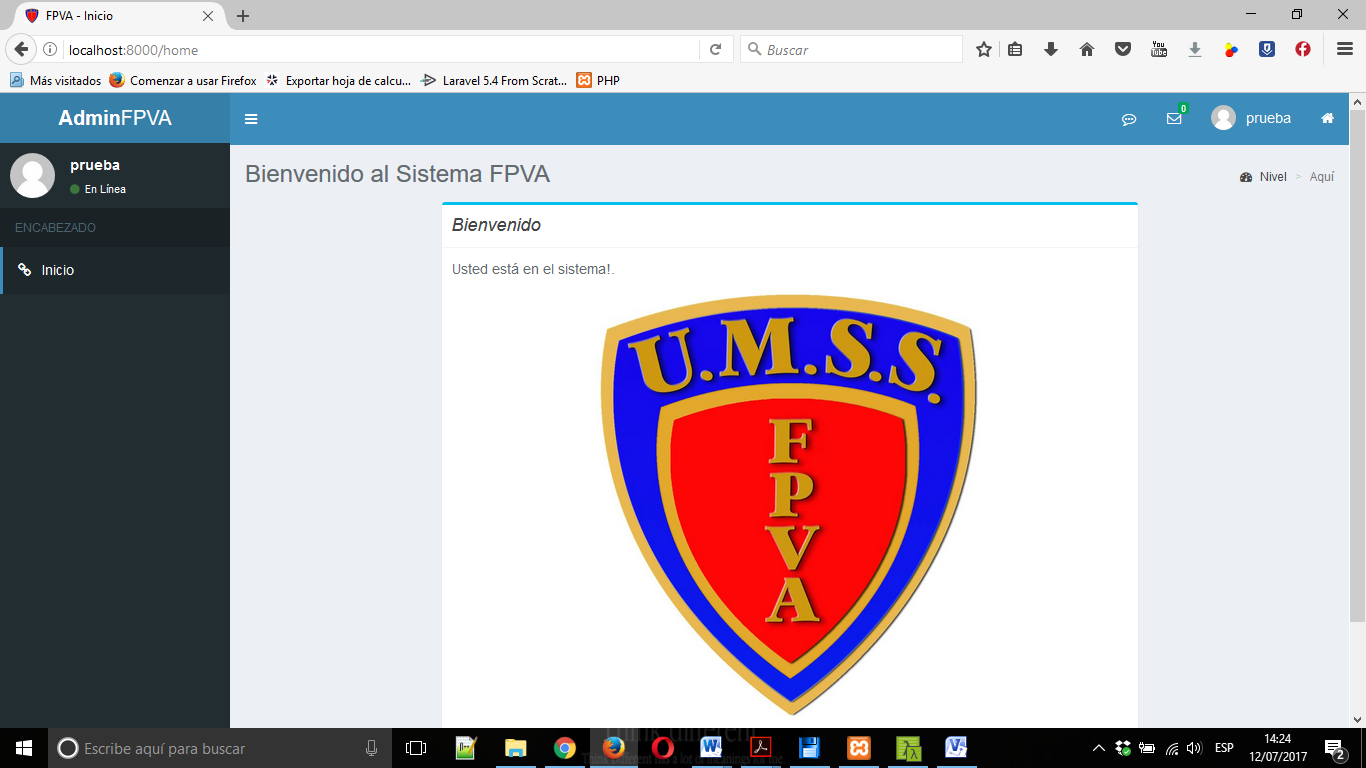


Figura 3.2 Diagrama de Clases Modulo Administración Usuarios

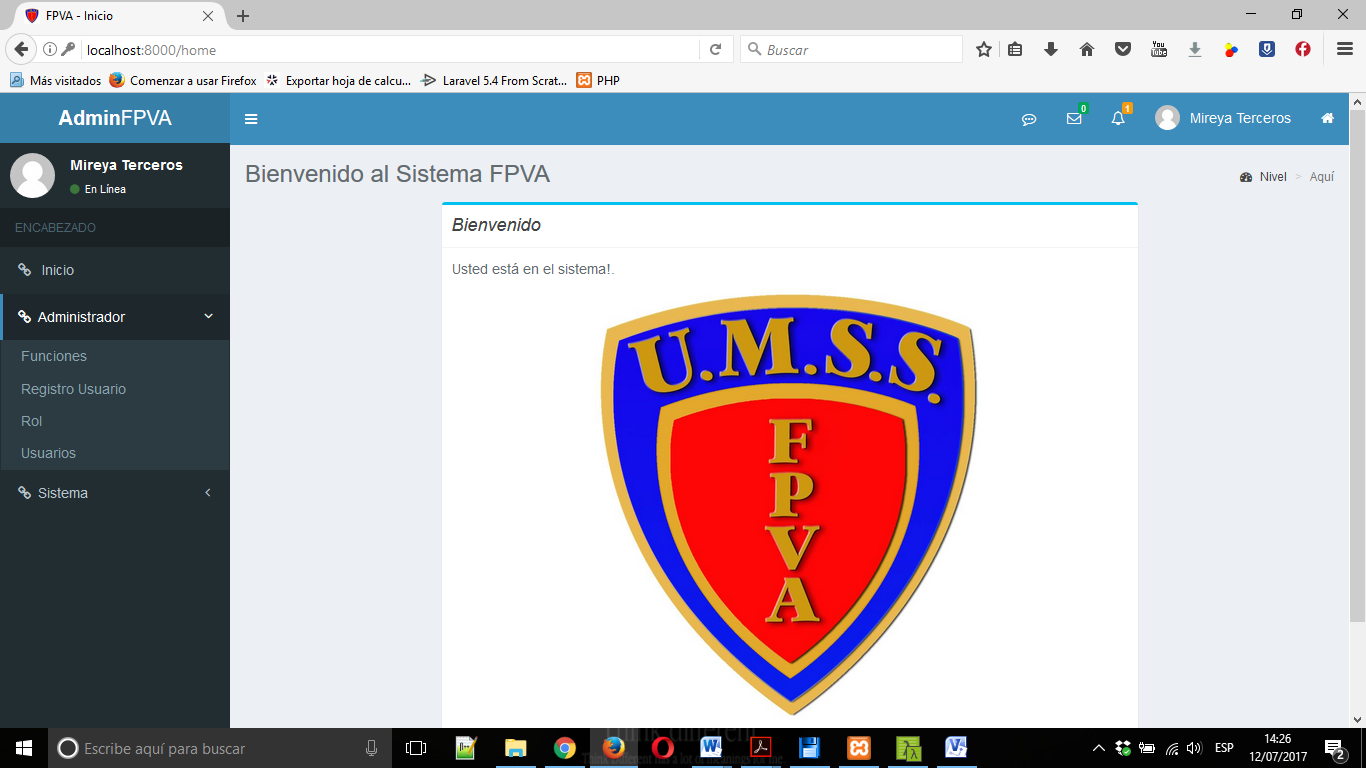
*Diseño de Interfaz Gráfica de Usuario (GUI):*



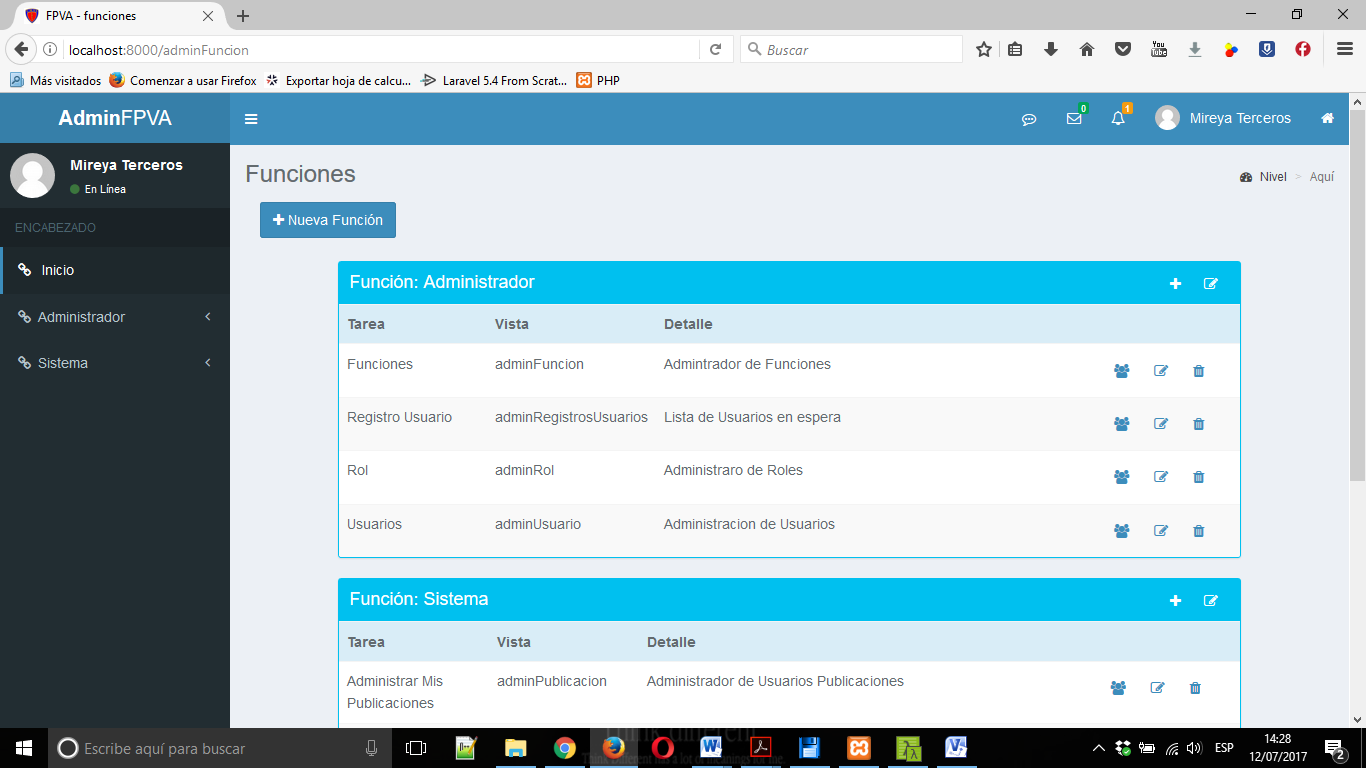
Interfaz Gráfica 3.1 Registro Usuarios



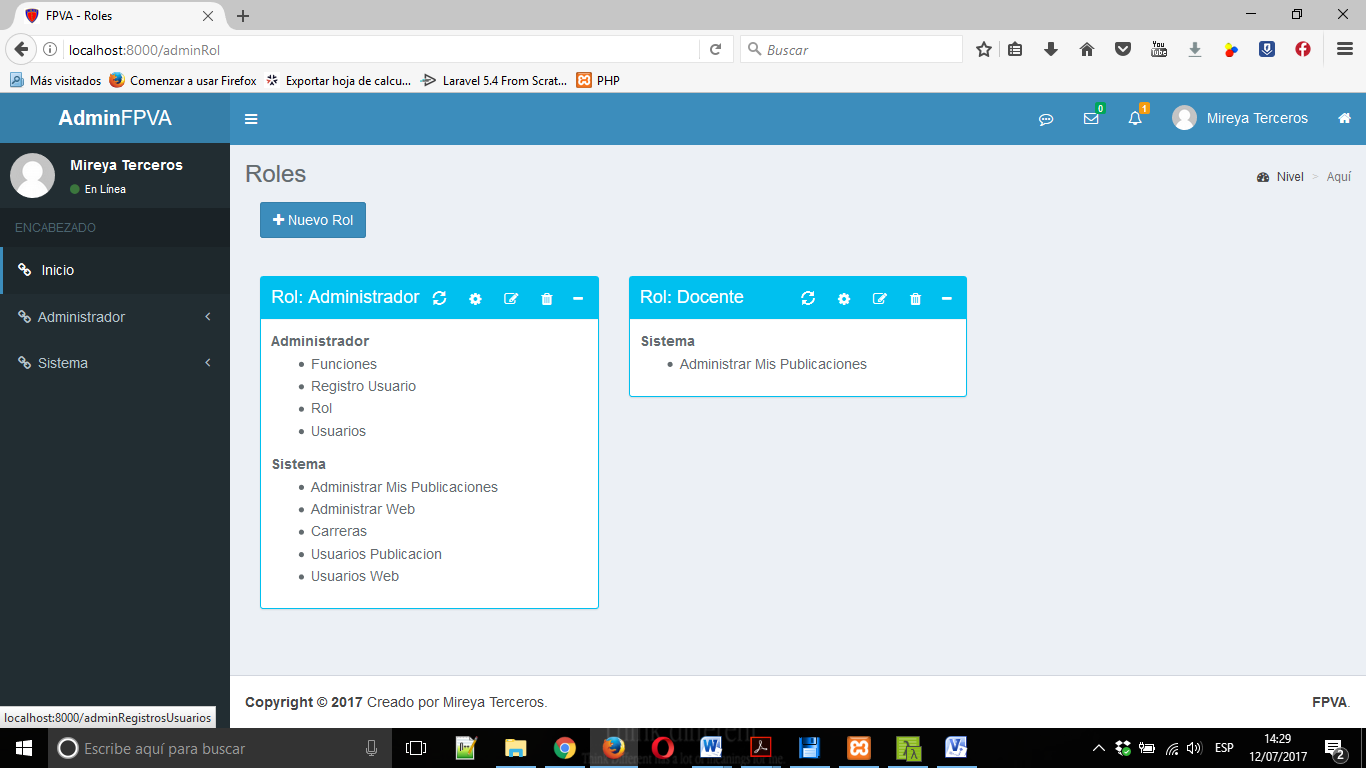
Interfaz Gráfica 3.2 Login Usuario Registrado Sin Permisos



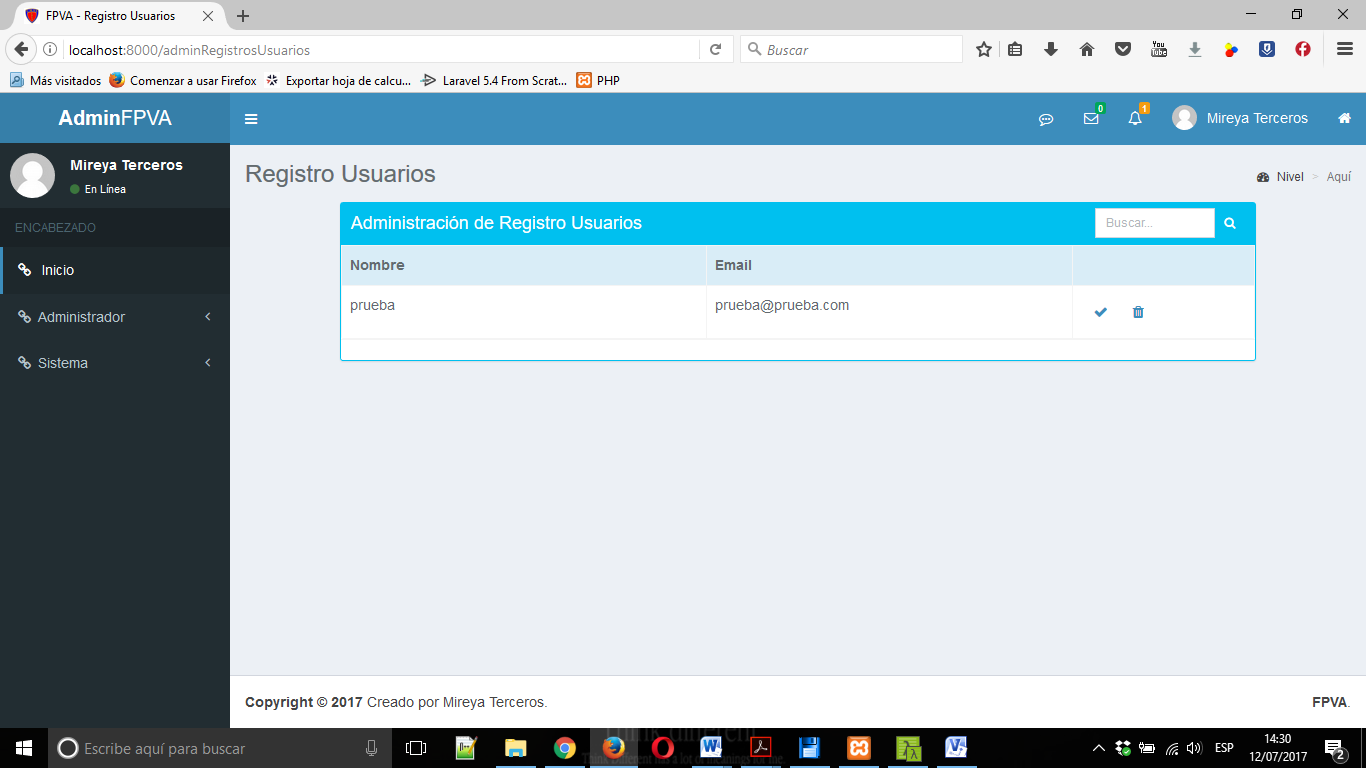
Interfaz Gráfica 3.3 Login Usuario Registrado Administrador



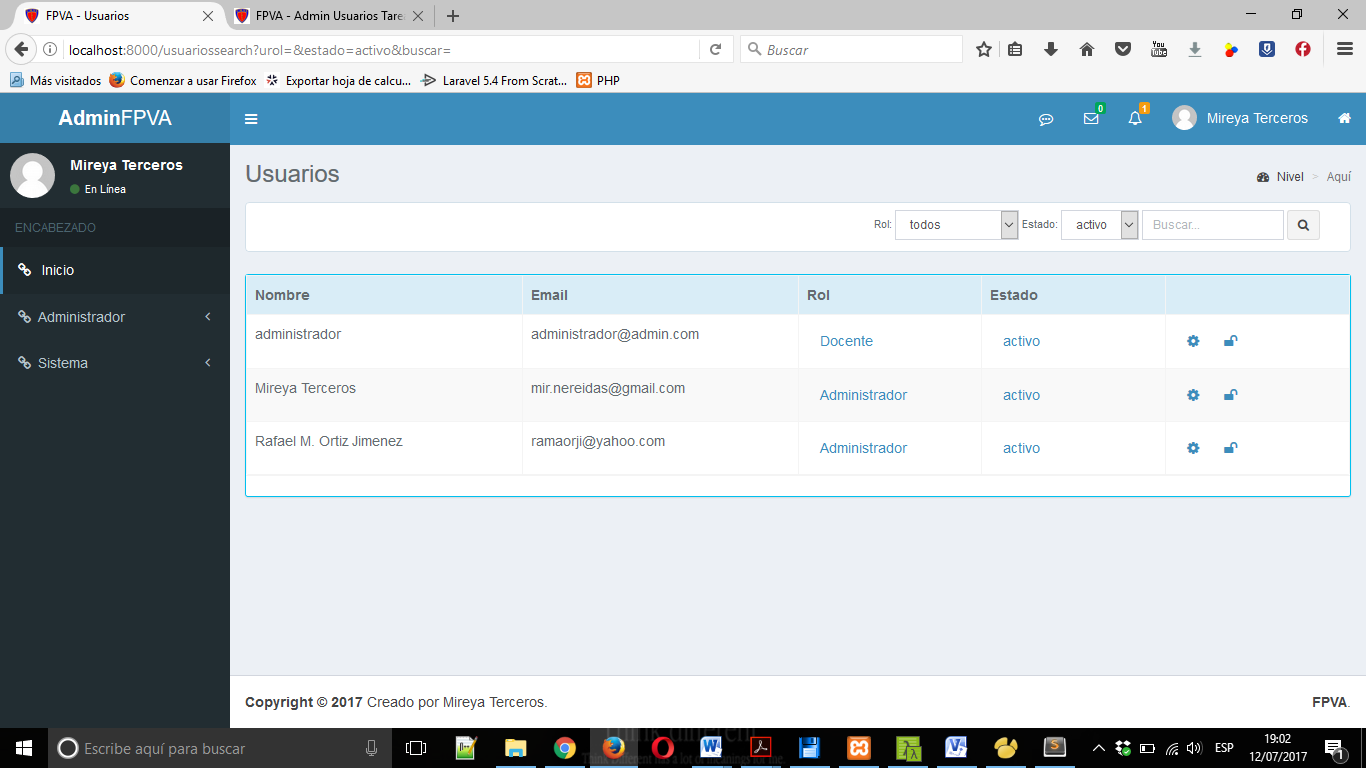
Interfaz Gráfica 3.4 CRUD Funciones



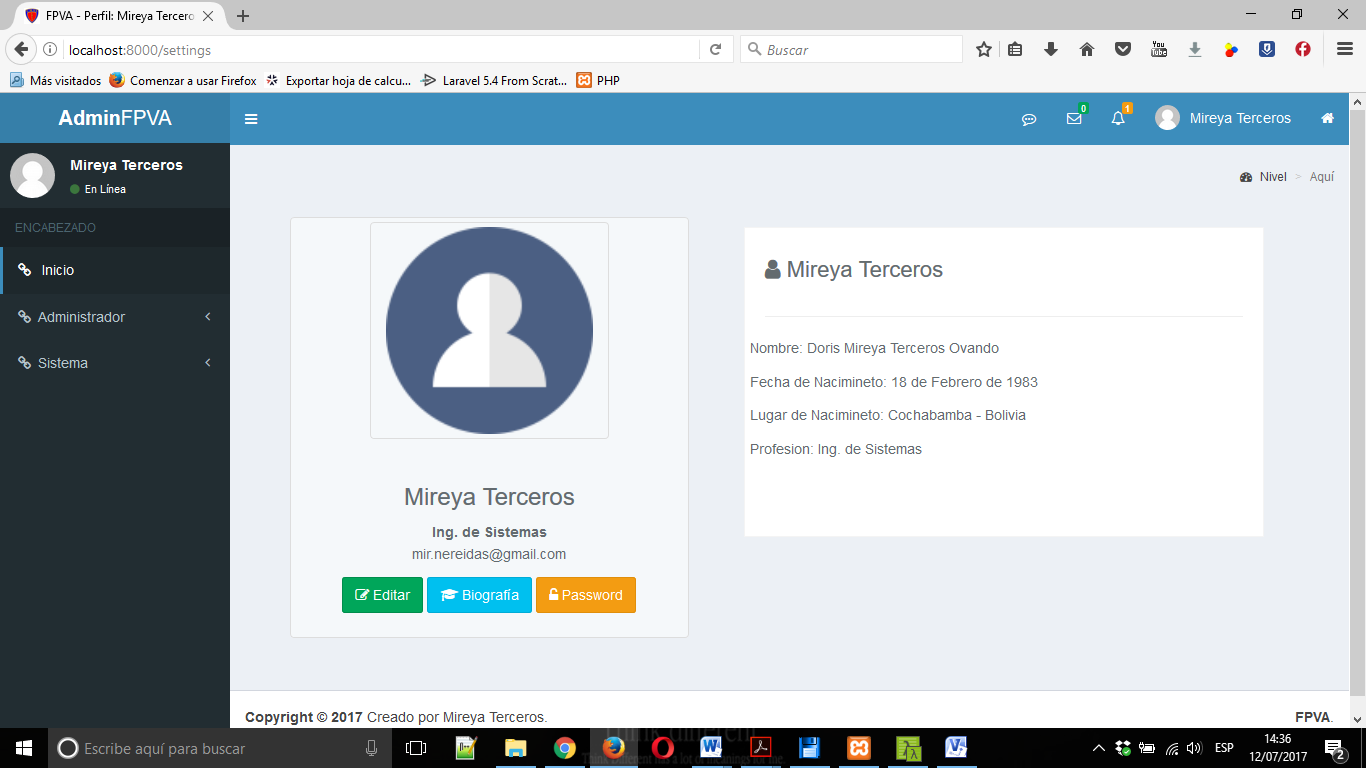
Interfaz Gráfica 3.5 CRUD Roles



Interfaz Gráfica 3.6 Administración Registro Usuarios



Interfaz Gráfica 3.7 Administración Usuarios



Interfaz Gráfica 3.8 Perfil de Usuario

* + 1. **Módulo Administración Funciones.**
* ***Etapa 01 – Descripción del Problema***

**Enunciado:** Desarrollar una solución que permita administrar toda la funcionalidad del sistema. Tomando como bases al usuario, la publicación y la facultad.

* ***Etapa 02 – Definición de la Solución***

**Resultado Deseado:**Generar y administrar los permisos del usuario a partir del rol. El rol asignado mostrara la funcionalidad y las tareas del usuario del sistema. Controlando los permisos de los módulos facultad y publicaciones con que cuenta el rol.

**Datos Necesarios:**id, id\_ftp, id\_ufw, id\_ufp, id\_umw, id\_ump, id\_ucw, id\_ucp, id\_ufw, id\_ufp, id\_mensaje, mensaje, id\_origen, id\_destino.

**Procesamiento:**La administración de la funcionalidad se la llevara a cabo por medio de tablas intermedias que nos permitan relacionar y filtrar datos de las tablas de la base de datos. Llegando a desarrollar métodos que nos permitan generar niveles de seguridad óptimos para el sistema.

Crear mensajería instantánea, recepción de Buzón de mensajes, registros de usuarios, editar el perfil personal.

* ***Etapa 03 – Diseño de la Lógica***

**Nombre del Proyecto:**ProyAdminFunction

**Diagrama de Paquetes:**



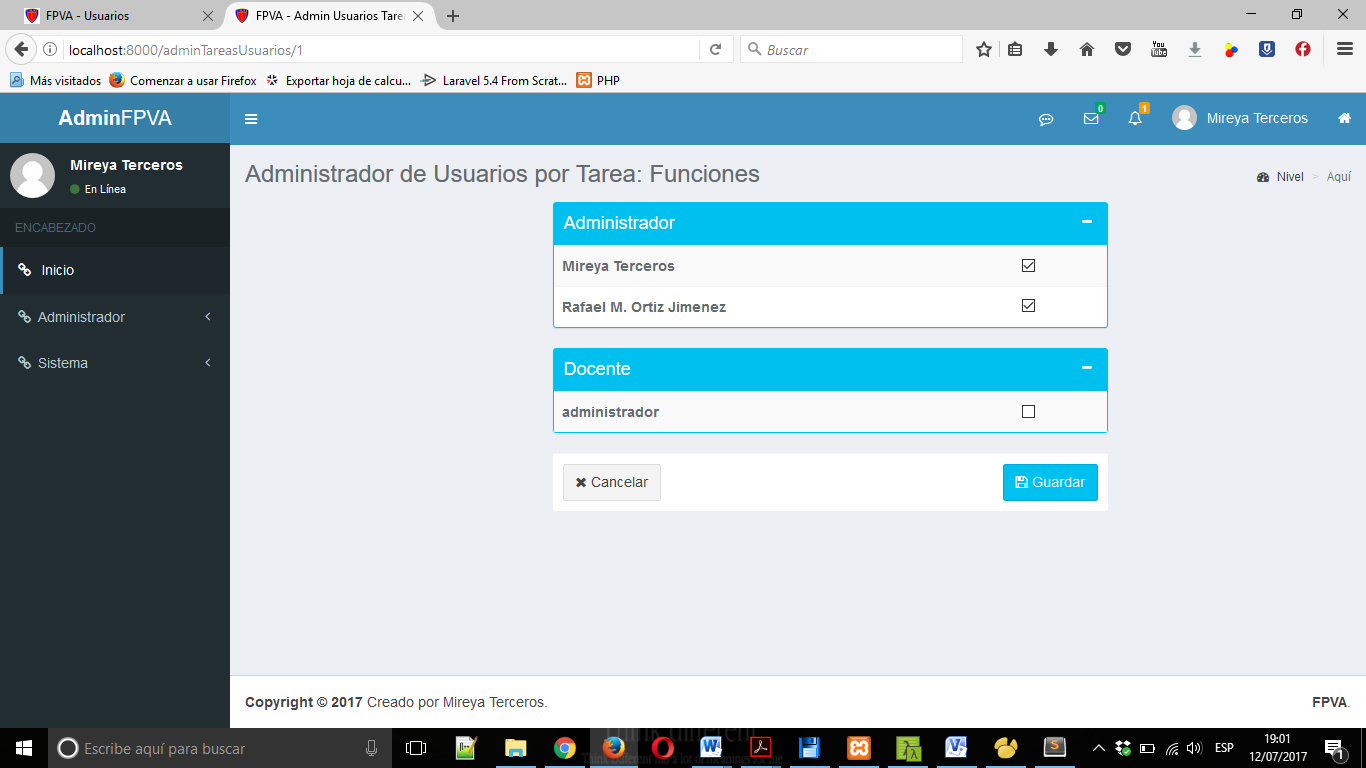
Figura 3.3 Modulo Administración Funciones

**Diagrama de Clases:**

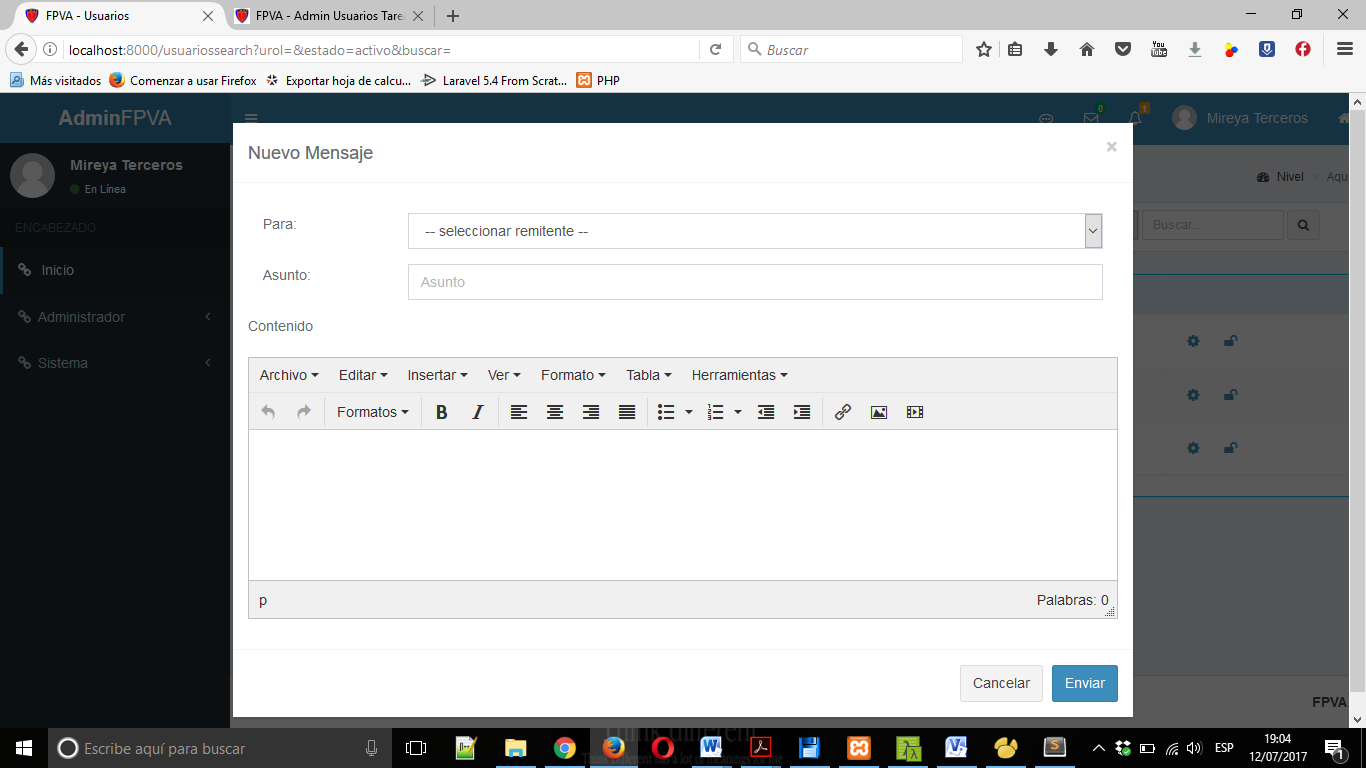


Figura 3.4 Diagrama de Clases Modulo Administración Funciones

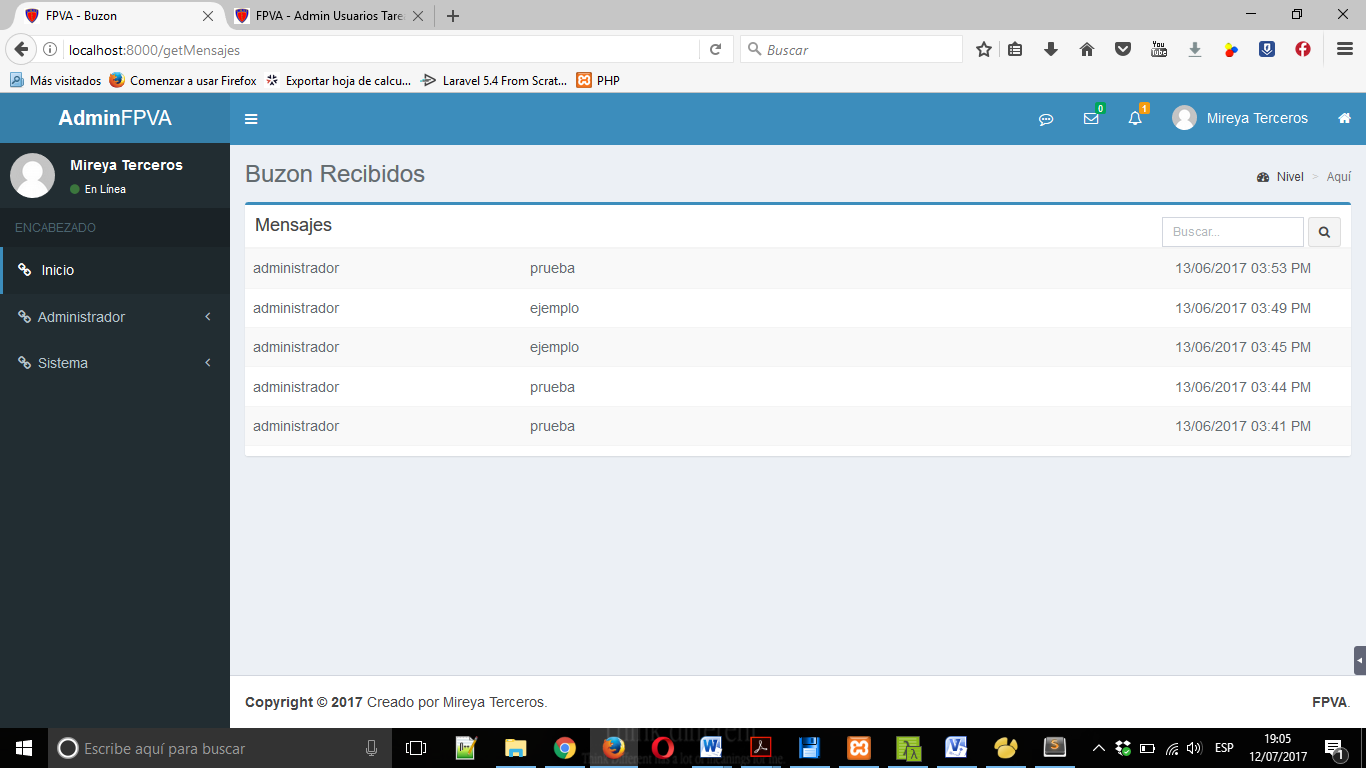
*Diseño de Interfaz Gráfica de Usuario (GUI):*



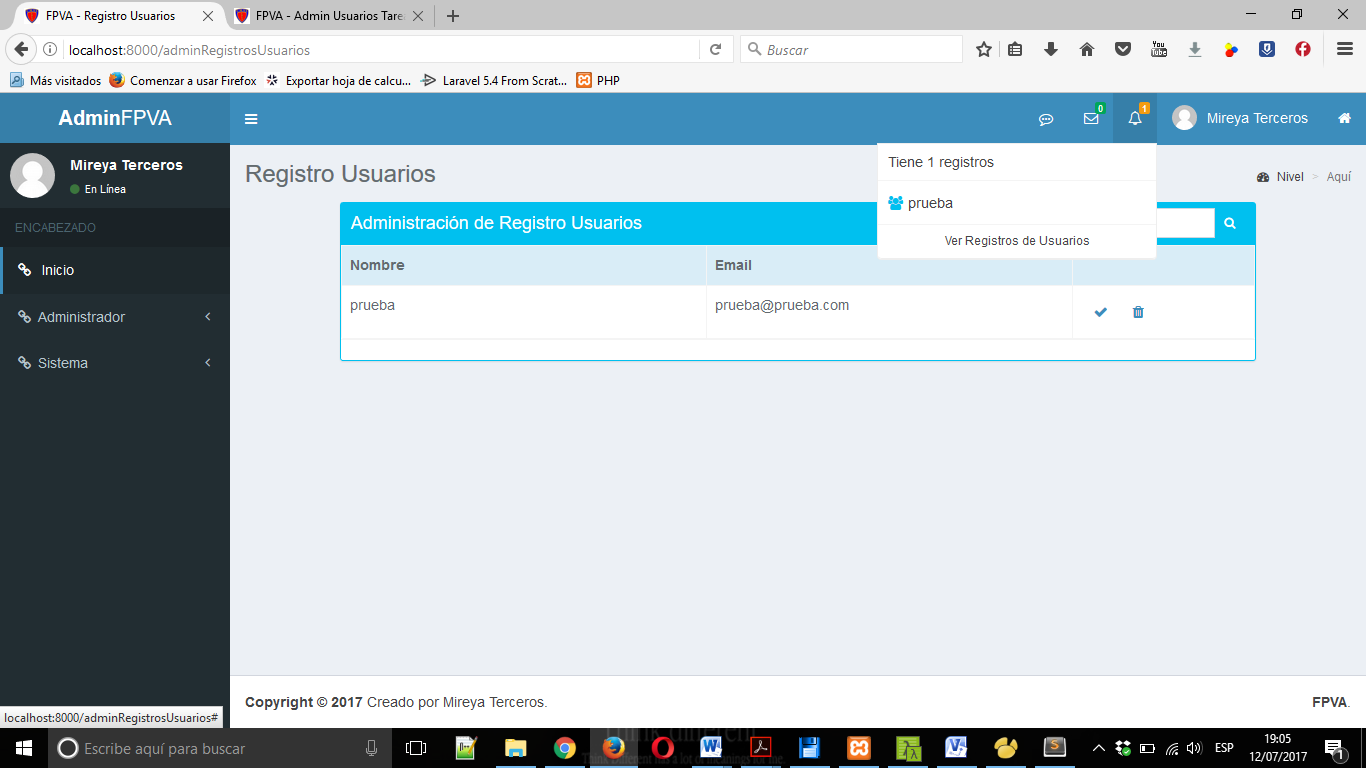
Interfaz Gráfica 3.9 Función Tarea Rol



Interfaz Gráfica 3.10 Nuevo Mensaje



Interfaz Gráfica 3.11 Buzón de Mensajes



Interfaz Gráfica 3.12 Usuarios Registrados

* + 1. **Módulo Administración Facultad.**
* ***Etapa 01 – Descripción del Problema***

**Enunciado:** Implementar un módulo que permita administrar facultad, carrera y materia. Tomando en cuenta que una es dependiente de la otra. Desarrollando un generador de mallas curriculares por carrera, mostrando la dependencia entre materias de la carrera.

* ***Etapa 02 – Definición de la Solución***

**Resultado Deseado:** Administrar y almacenar datos de facultad, carrera y/o materia. Generar dependencias entre materias de una carrera. Implementar el diseño del Portal web.

**Datos Necesarios:**id\_facultad, facultad, estado, id\_carrera, carrera, codigo, estado, id\_materia, materia, nivel, codigo, sigla, estado, id\_dependencia, id\_previa.

**Procesamiento:** Lo primero que se debe crear es una Facultad, la misma cuenta con carreras y cada carrera cuenta con materias, estas serán almacenadas en su respectiva tabla Facultades, Carreras y Materias. También almacenaremos las dependencias de cada materia en la tabla Dependencias. Llegando a desarrollar métodos modulables que permitan una eficiente administración de la facultad.

* ***Etapa 03 – Diseño de la Lógica***

**Nombre del Proyecto:**ProyAdminFacultad

**Diagrama de Paquetes:**



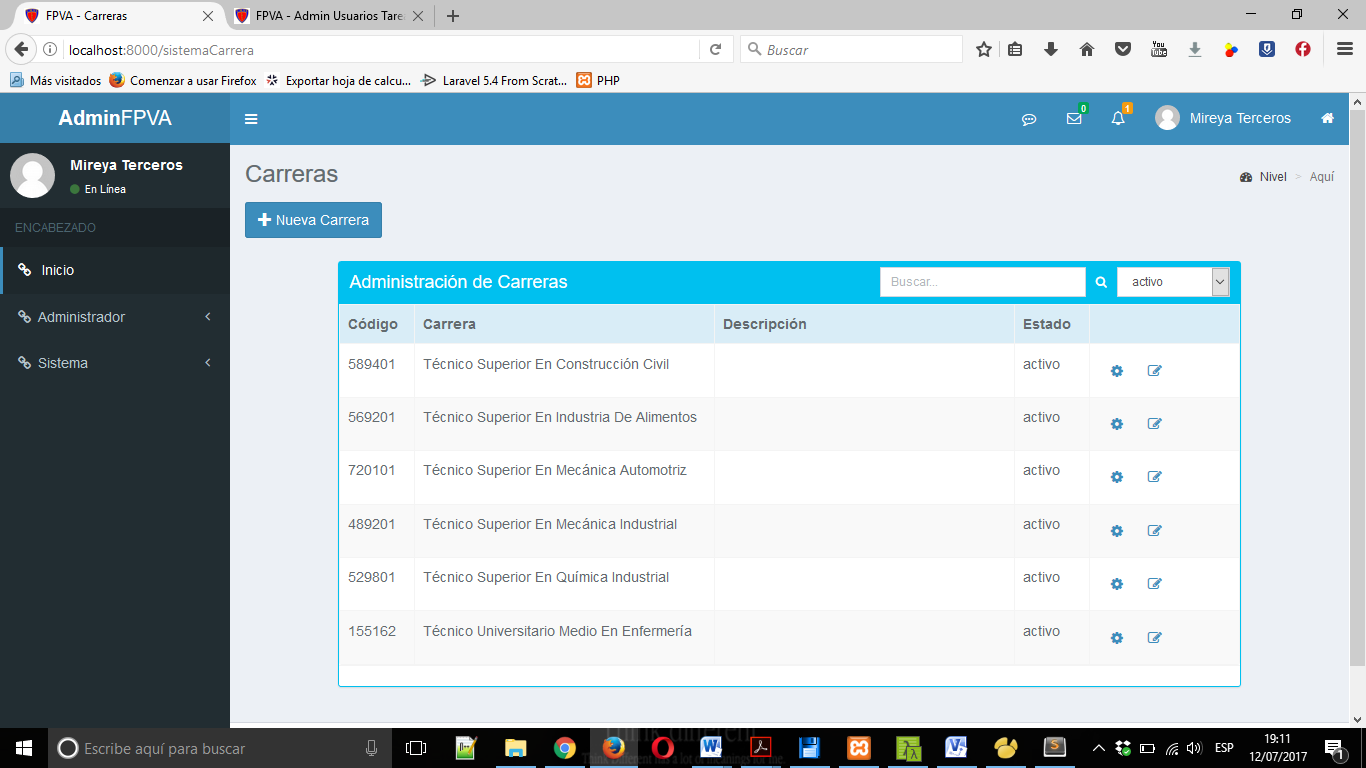
Figura 3.5 Modulo Administración Facultad

**Diagrama de Clases:**

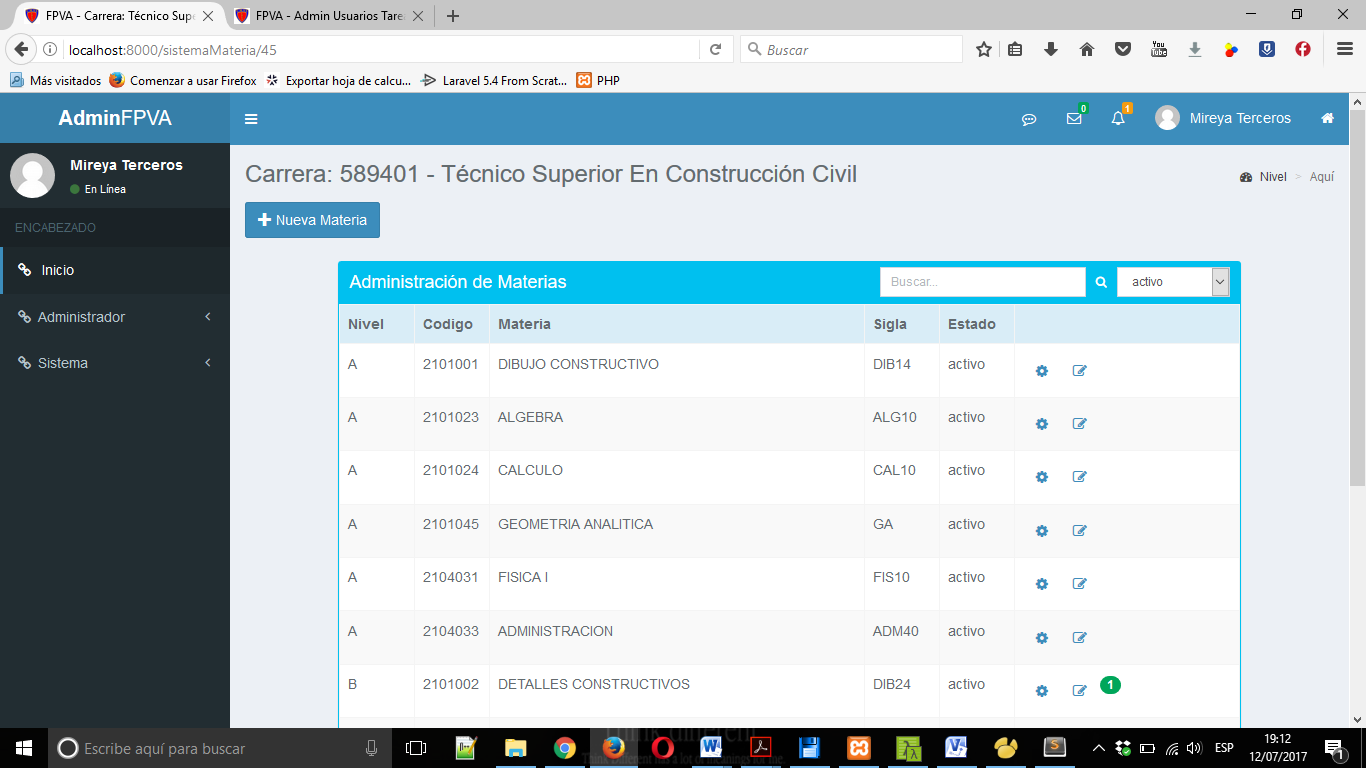


Figura 3.6 Diagrama de Clases Modulo Administración Facultad

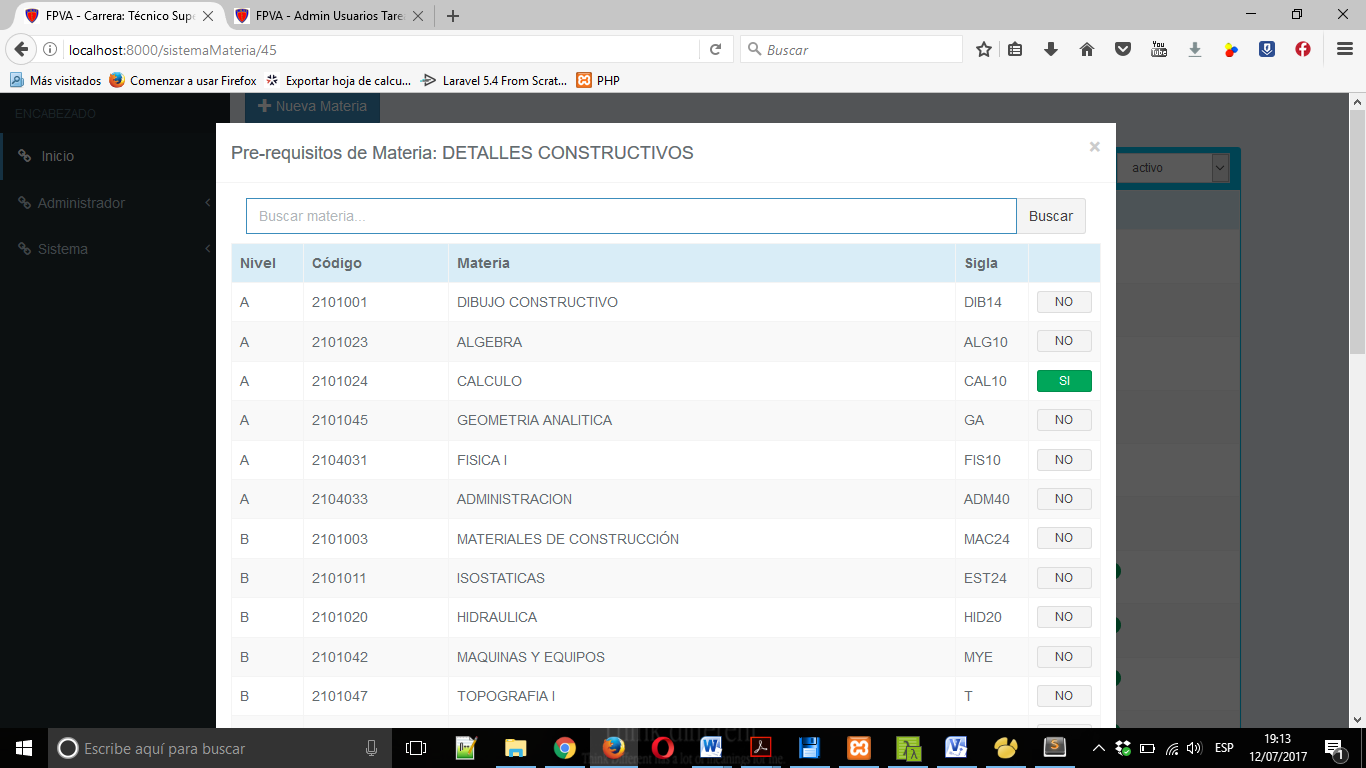
*Diseño de Interfaz Gráfica de Usuario (GUI):*



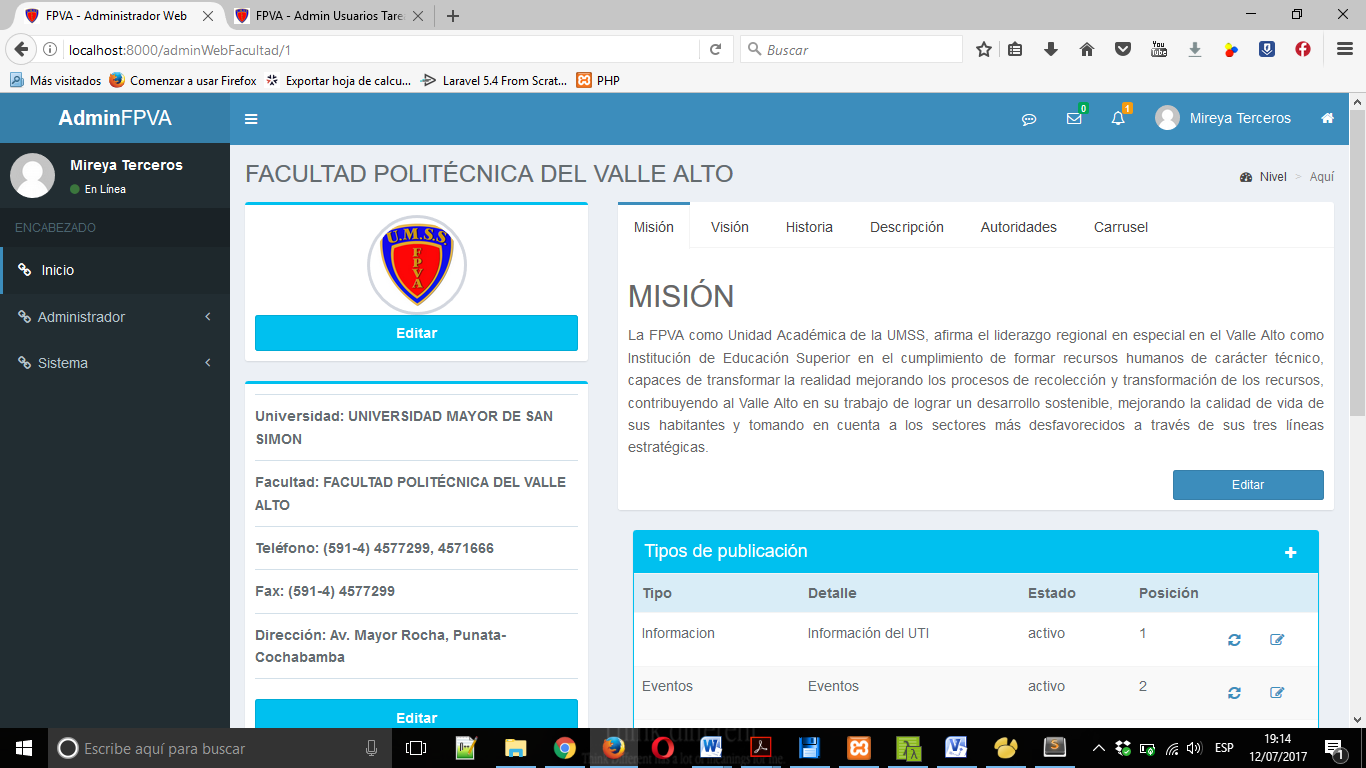
Interfaz Gráfica 3.13 CRUD Carrera



Interfaz Gráfica 3.14 CRUD Materia



Interfaz Gráfica 3.15 Generación de Dependencias



Interfaz Gráfica 3.16 Editar Facultad

* + 1. **Módulo Administración Publicaciones.**
* ***Etapa 01 – Descripción del Problema***

**Enunciado:**Diseñar un módulo que administre publicaciones por facultad, carrera y/o materia. Para este módulo se debe tomar en cuenta que debemos manejar tipos de usuario y que si una publicación esta fuera de las normas de la FPVA, esta deberá ser dada de baja por el usuario administrador. El usuario podrá realizar publicaciones únicamente en la materia o carrera que tenga permisos. También se implementara el diseño del Portal Web.

* ***Etapa 02 – Definición de la Solución***

**Resultado Deseado:**Almacenar y gestionar las publicaciones de facultad, carrera y/o materia por tipo de usuario, publicación e importancia. Mostar las publicaciones en el portal web para visitantes.

**Datos Necesarios:** id\_publicacion, titulo, contenido, etiqueta, publicador, periodo\_publicacion, id\_tabla, tabla, área, id\_ftp, tipo, id importancia, importancia, id\_etiqueta, id\_fpe.

**Procesamiento:** Las publicaciones estarán almacenadas en la tabla fpublicaciones y la misma está relacionada con las tablas ftipo\_publicaciones, importancias, etiquetas, visitas. Donde por medio de métodos controlaremos y administraremos los datos de publicación.

* ***Etapa 03 – Diseño de la Lógica***

**Nombre del Proyecto:**ProyAdminPublicaciones

**Diagrama de Paquetes:**



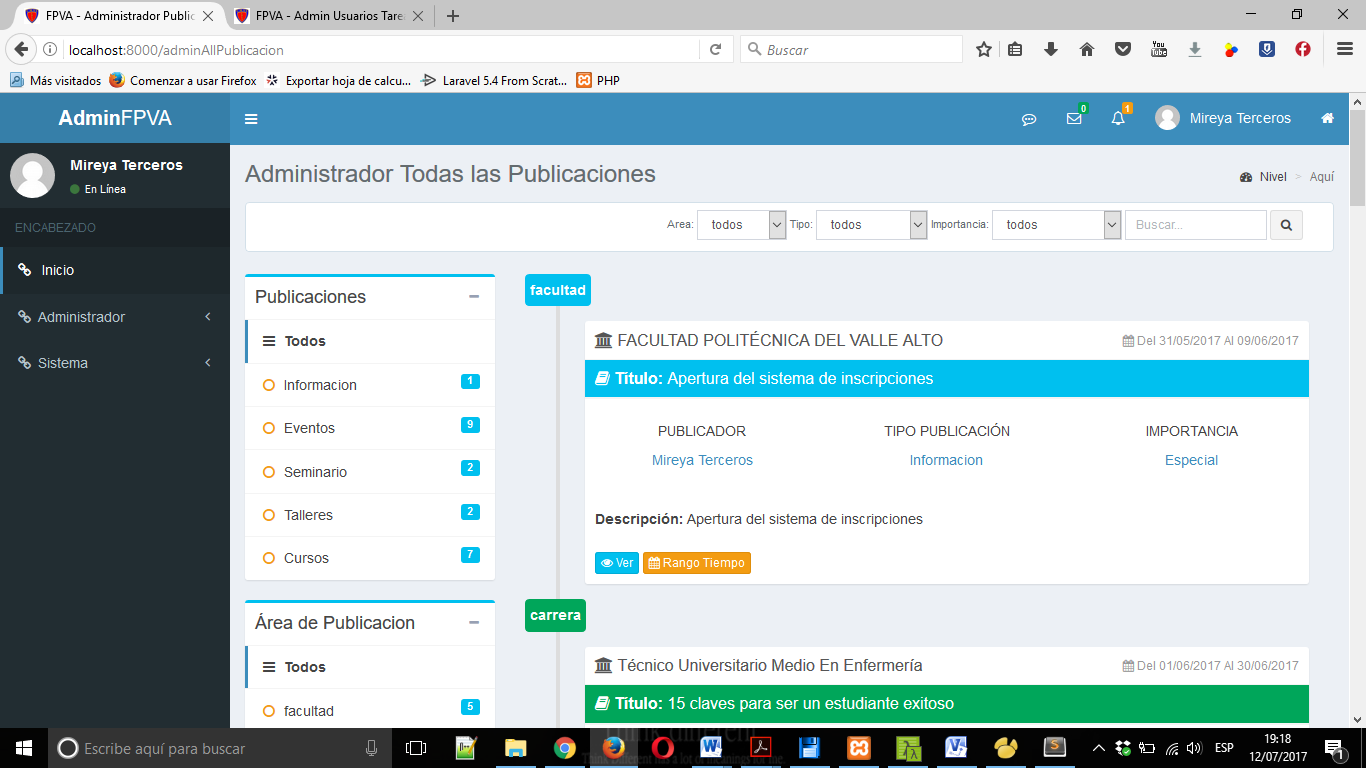
Figura 3.7 Modulo Administración Publicación

**Diagrama de Clases:**

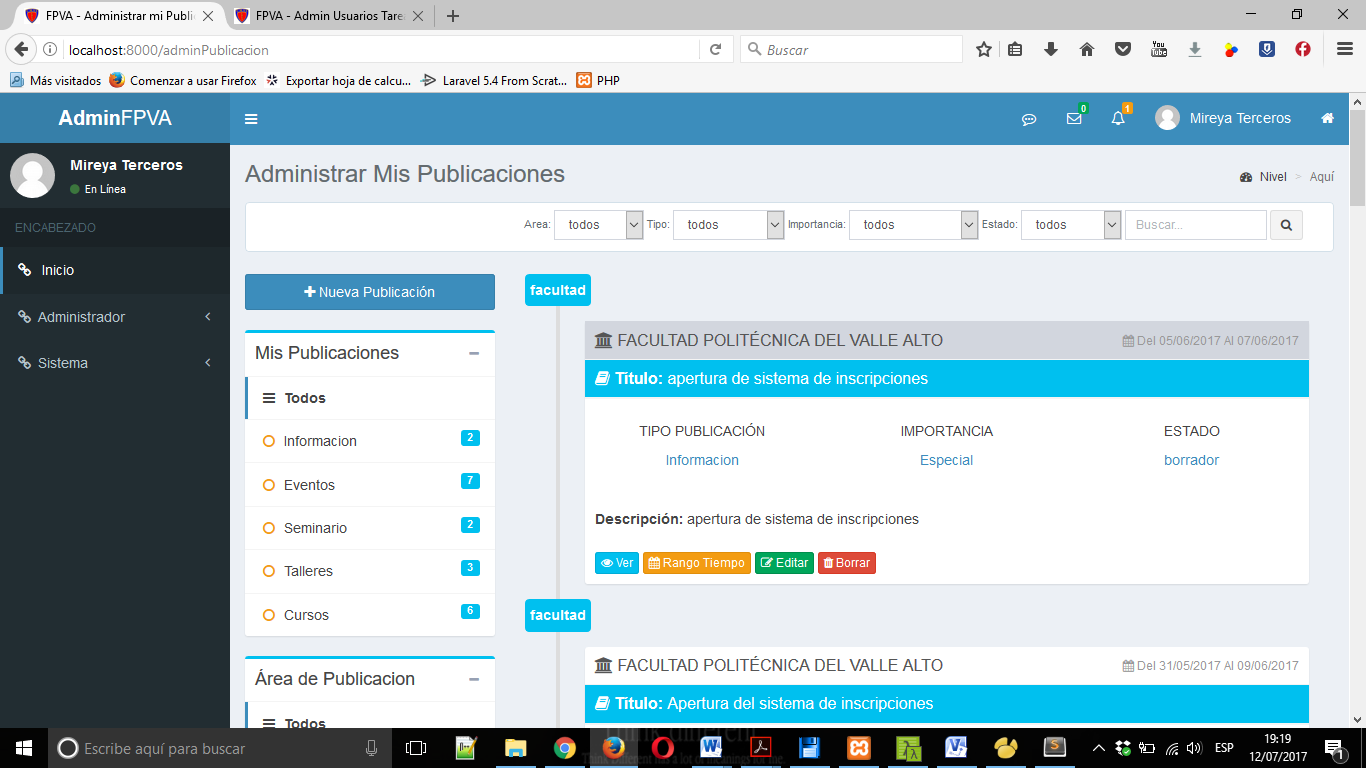


Figura 3.8 Diagrama de Clases Modulo Administración Publicación

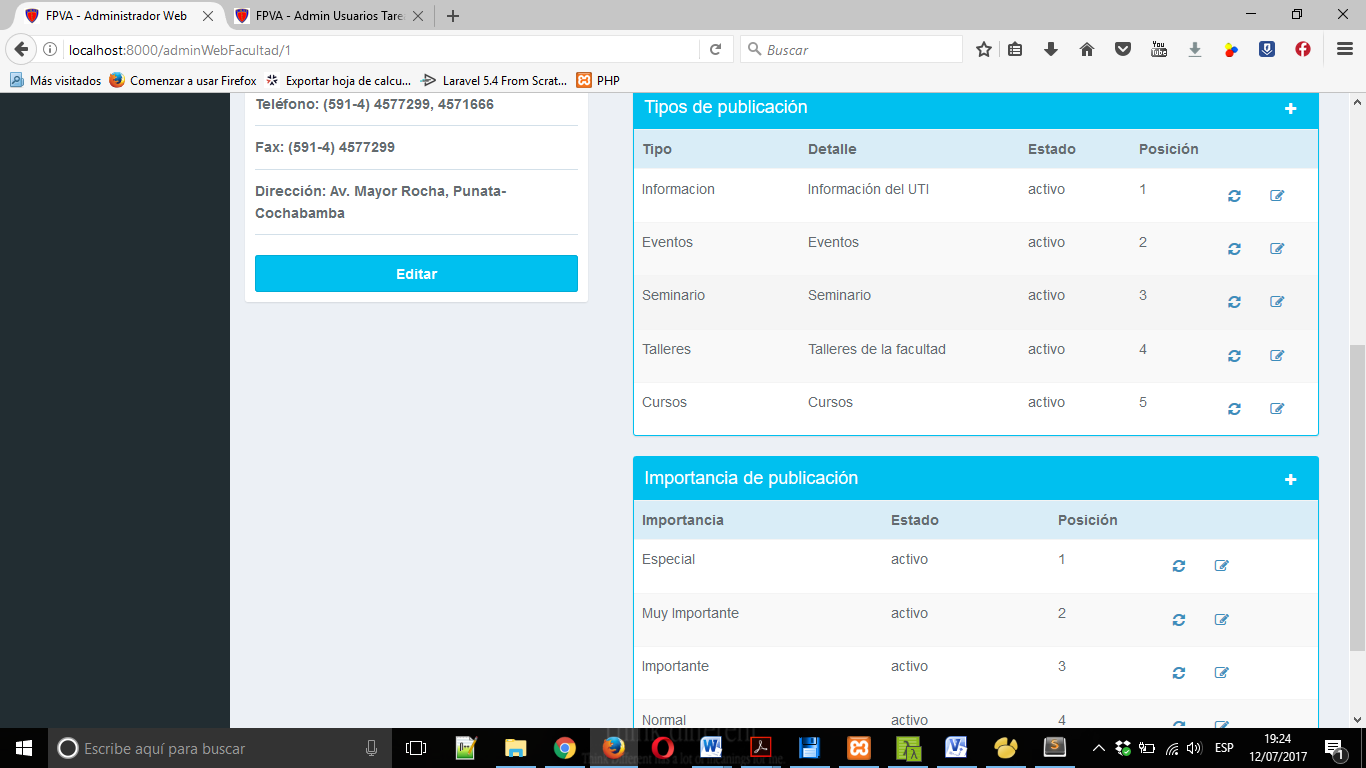
*Diseño de Interfaz Gráfica de Usuario (GUI):*



Interfaz Gráfica 3.17 Administrar Publicaciones del Sistema



Interfaz Gráfica 3.18 Administrar Mis Publicaciones



Interfaz Gráfica 3.19 CRUD Tipo Publicación e Importancia

* + 1. **Módulo Administración Web.**

***Etapa 01 – Descripción del Problema***

**Enunciado:**Se debe desarrollar un módulo que nos permita fusionar el sistema de administración con el portal web, para que el mismo pueda mostrar información filtrada por los usuarios del sistema a los usuarios visitantes.

***Etapa 02 – Definición de la Solución***

**Resultado Deseado:**Administrar de manera conjunta al sistema de administración como el portal web. Generar reportes y estadísticas de visitas. Desarrollar un gestor de mensajería y de búsqueda de información.

**Datos Necesarios:** La base de datos en general.

**Procesamiento:** El usuario del sistema podrá ver la fusión del sistema administración como el portal web, es decir, acceder a la información del sistema desde el portal web.

***Etapa 03 – Diseño de la Lógica***

**Nombre del Proyecto:** ProyAdminWeb

**Diagrama de Paquetes:**



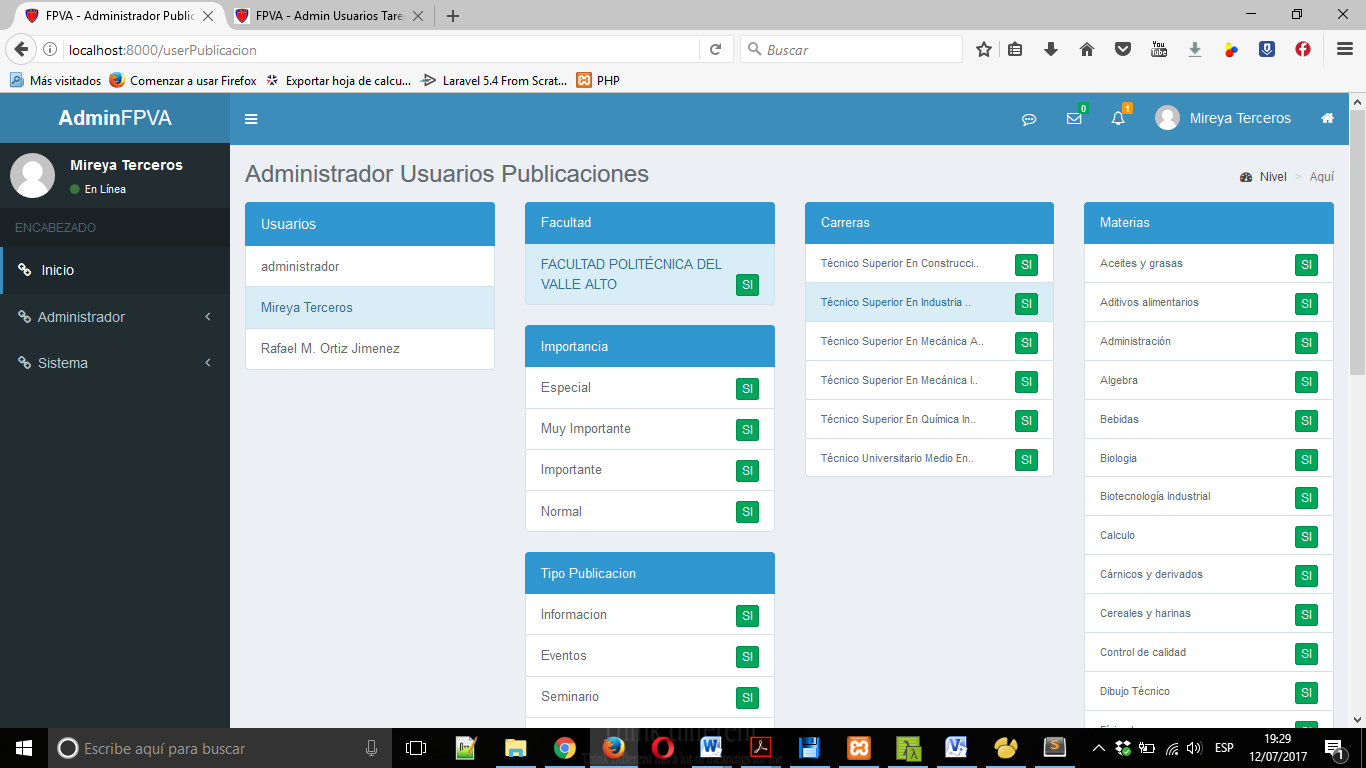
Figura 3.9 Modulo Administración Web

**Diagrama de Clases:**

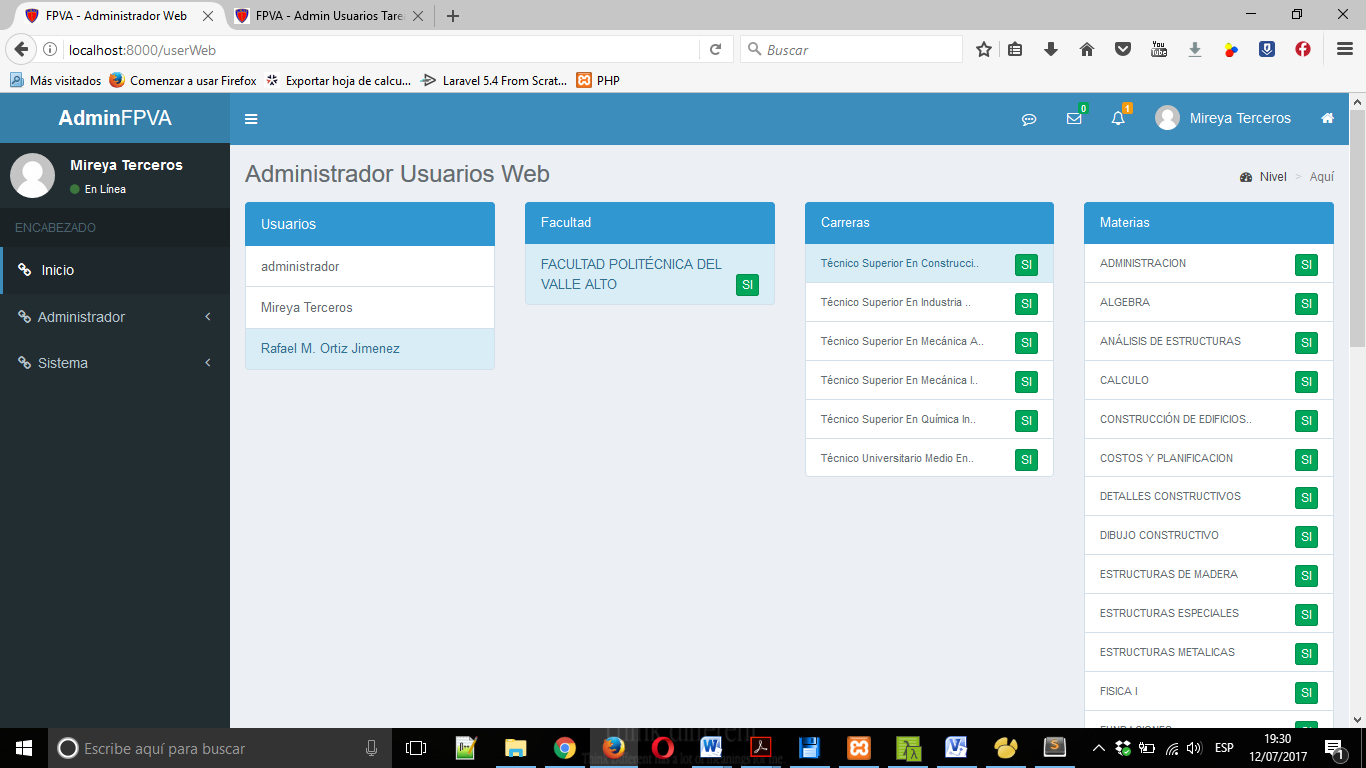


Figura 3.10 Diagrama de Clases Modulo Administración Web

*Diseño de Interfaz Gráfica de Usuario (GUI):*



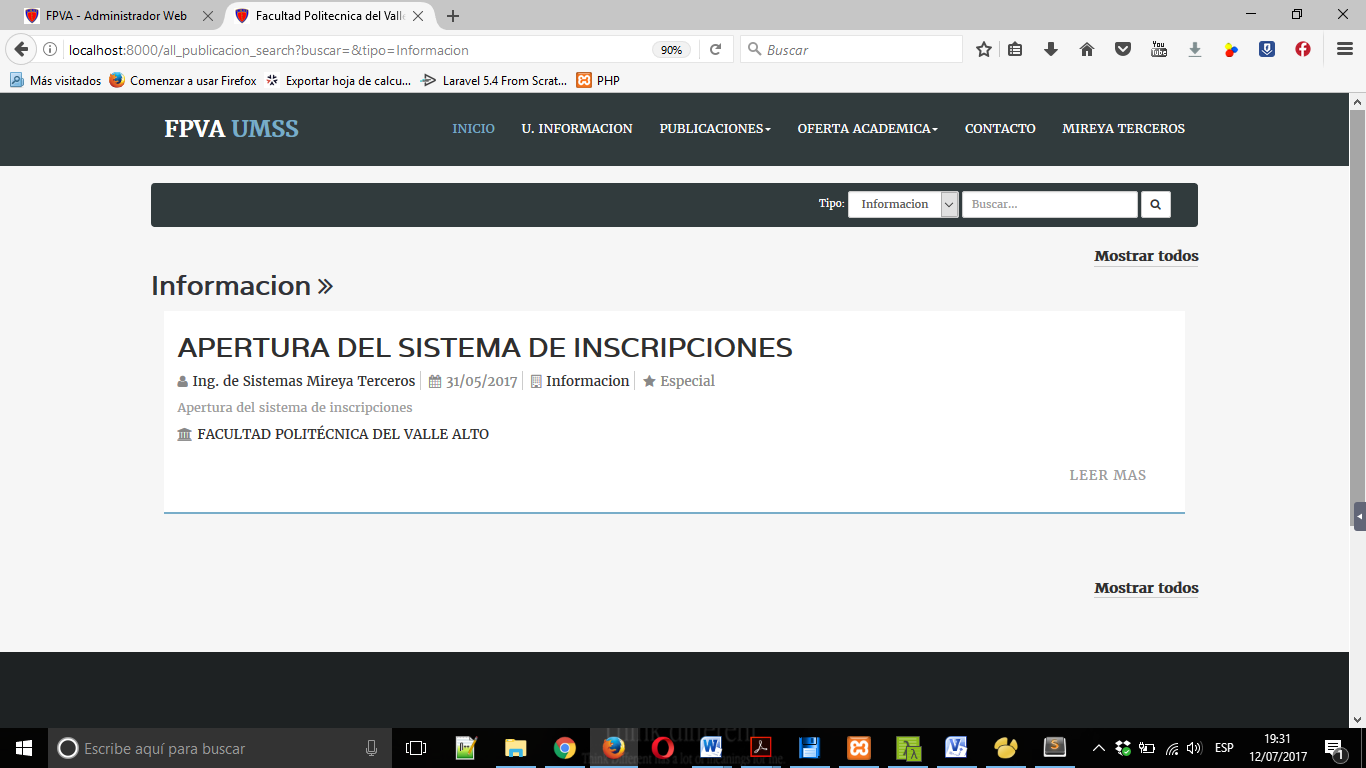
Interfaz Gráfica 3.20 Administración de Usuarios para Publicaciones



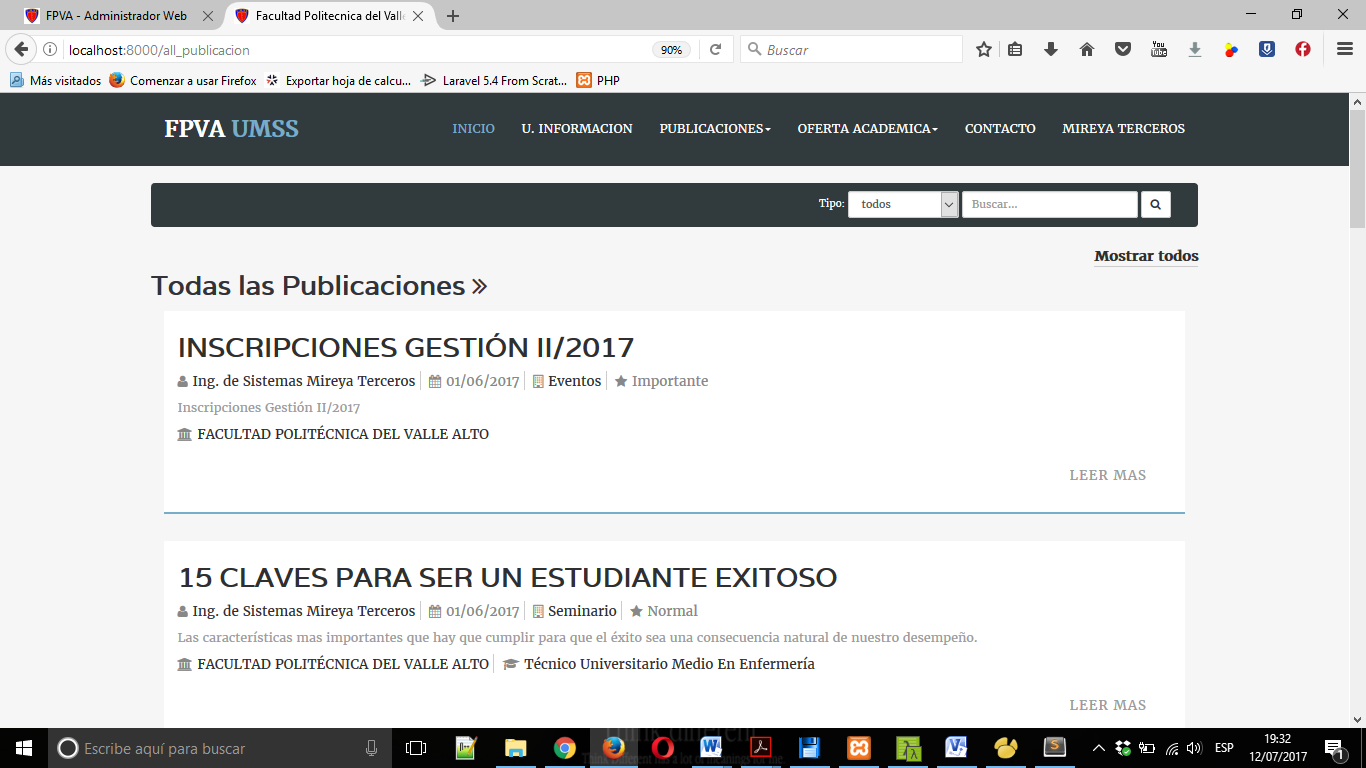
Interfaz Gráfica 3.21 Administración de Usuarios en la Web



Interfaz Gráfica 3.22 Página Principal Portal Web



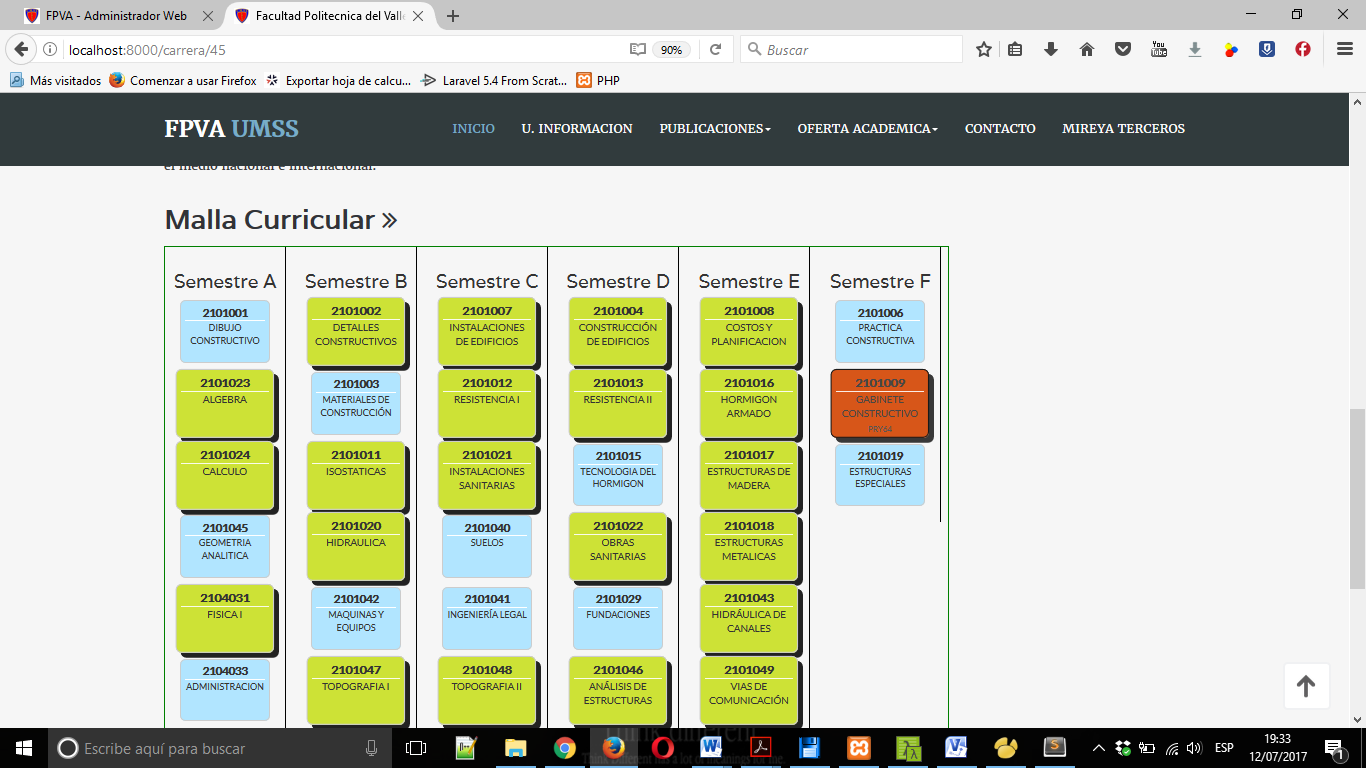
Interfaz Gráfica 3.23 Página Unidad de Información (UTI) Portal Web



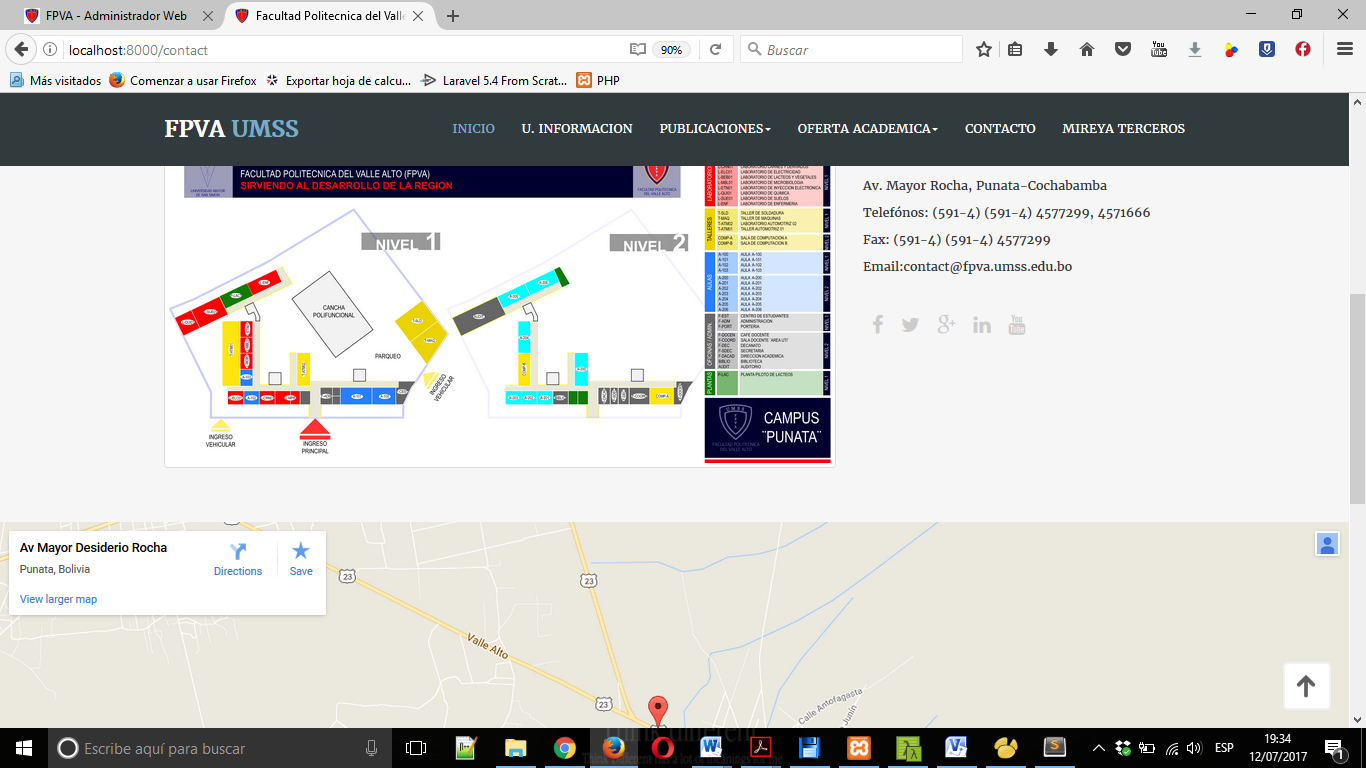
Interfaz Gráfica 3.24 Página Publicaciones Portal Web



Interfaz Gráfica 3.25 Página Carrera Portal Web



Interfaz Gráfica 3.26 Página Carrera Malla Curricular Portal Web



Interfaz Gráfica 3.27 Página Contacto Portal Web

* 1. **SÍNTESIS**

Para desarrollar el sistema se realizó un análisis, lo cual nos llevó a determinar que el sistema debía estar diseñado por módulos para una mejor comprensión del mismo.

Los módulos diseñados para el sistema son cinco: módulo de administración de usuarios, módulo de administración de funciones, módulo de administración de la facultad, módulo de administración de publicaciones y el módulo de administración web.

*CAPITULO 4*

1. TECNOLOGIAS
   1. **INTRODUCCIÓN**

Las tecnologías implementadas para el desarrollo del Sistema Web de Manejo y Difusión de Información de la Facultad Politécnica del Valle Alto, son RESPONSIVE, ya que el sistema es accesible desde cualquier dispositivo móvil que cuente con conexión a Internet. Por lo cual se hizo uso de Bootstrap como una de las tecnologías principales. Así también Laravel en su versión 5.4.

El sistema también está desarrollado en HTML5 como lenguaje de programación, complementario a esto se empleó JQuery.JS que es un framework de Javascript el cual contiene funcionalidades dando a nuestra web funcionalidad y efectos complejos, BOOTSTRAP es un framework que nos permitió crear interfaces Web con CSS y JavaScript posibilito que nuestra web sea Responsive Desing (Diseño Adaptativo), Composer es un manejador de dependencias que facilita el manejo de librerías. También se usó el servidor HTTP Apache de código abierto. Finalmente, para la base de datos se usó MySQL.

* 1. **LARAVEL** 
     1. **HISTORIA**

Taylor Otwell creó Laravel como un intento de proporcionar una alternativa más avanzada al framework CodeIgniter, que no proporcionaba ciertas características tales como soporte incorporado para la autenticación y autorización del usuario. La primera versión beta de Laravel se puso a disposición el 9 de junio de 2011, seguida por la versión de Laravel 1 lanzada el mismo mes. Laravel 1 tiene un soporte integrado para autenticación, localización, modelos, vistas, sesiones, enrutamiento y otros mecanismos, pero carecía de soporte para los controladores que le impidieron ser un verdadero framework MVC.

Laravel 2 fue lanzado en septiembre de 2011, trayendo varias mejoras del autor y la comunidad. Las nuevas características principales incluyeron el soporte para los controladores, lo que hizo que Laravel 2 fuera un framework totalmente compatible con MVC, soporte incorporado para soportar el principio de control de inversión (loC) y un sistema de plantillas llamado Blade. Como un inconveniente, el soporte para paquetes se eliminó.

Laravel 3 fue lanzado en febrero del 2012 con un conjunto de nuevas características incluyendo la Interfaz de Línea de Comandos (CLI) llamada Artisan, soporte integrado para Sistemas de Gestión de Bases de Datos, la migración y bases de datos como una forma de control de versiones para bases de datos, eventos, y un sistema de empaquetado llamado Bundles. El aumento de usuarios y popularidad de Laravel promovió el lanzamiento de Laravel 3.

Laravel 4, con Illuminate, fue lanzado en mayo de 2013. Se reescribió por completo el framework Laravel, la migración de su diseño en un conjunto de paquetes distribuidos a través de Composer, que sirve como un gestor de paquetes a nivel de aplicación. Esto mejoro la extensibilidad de Laravel 4, que fue lanzada a la par de su lanzamiento oficial. Otras nuevas funciones de la versión4 de Laravel incluyen la preselección de la base de datos, el soporte de mensajes queues, soporte de envío de diferentes tipos de correos electrónicos y soporte en la eliminación de registros de la base de datos llamada eliminación suave.

Laravel 5 fue lanzado en febrero de 2015 como resultado de cambios internos. Las nuevas características del lanzamiento de Laravel 5 incluyen soporte para la programación de tareas ejecutadas periódicamente a través del paquete llamado Scheuduler, una capa de abstracción llamada Flysystem que permite el almacenamiento remoto para ser usado de la misma manera que los sistemas de archivos locales, mejorando la gestión de paquetes activos a través de Elixir, simplificado el manejo de autentificación externa por medio del paquete Socialite. Laravel 5 tambien introdujo una nueva estructura de directorios para aplicaciones desarrolladas.

Laravel 5.1, lanzado en junio de 2015, en esta ocasión es el primer lanzamiento de Laravel para recibir soporte a largo plazo (LTS), con disponibilidad prevista para correcciones de errores por tres años y parches de seguridad.

Laravel 5.3, lanzado el 23 de agosto de 2016, entre sus nuevas características se centran en mejorar la velocidad del desarrollador añadiendo mejoras adicionales en las tareas comunes.

Laravel 5.4, lanzado el 24 de enero de 2017. Esta versión tiene nuevas características como Laravel Dusk, Laravel Mix, Blade Components y Slots, Markdown Emails, Automatic Facades, Route Improvements, Higher Order Messaging for Collections, entre otros.

* + 1. **CARACTERISTICAS DE LARAVEL 5.4**

Las siguientes características sirven como puntos clave de diseño de Laravel:

* Los paquetes proporcionan un sistema de empaquetado modular, con las características incluidas para la adición fácil a las aplicaciones. Ademas, Laravel utiliza Composer como un gestor de dependencias para agregar paquetes PHP específicos de Laravel y disponibles para el repositorio Packagist.
* ORM Eloquente (Modelo Objeto-Relación) es una implementación avanzada de PHP del patrón de registro activo, proporcionando al mismo tiempo métodos internos para imponer restricciones en las relaciones entre objetos de la base de datos. ORM Eloquente presenta tablas de bases de datos como clases, con las instancias de objeto unidas a las filas de la tabla como única.
* Query Builder, proporcióna una alternativa de acceso a bases de datos más directa al ORM Eloquente. En lugar de requerir que las consultas SQL se escriban directamente, el constructor de consultas de Laravel proporciona un conjunto de clases y métodos capaces de generar consultas de forma programática. También permite el almacenamiento en cache seleccionable de los resultados de la consultas ejecutadas.
* La lógica de la aplicación es una parte integral de las aplicaciones desarrolladas, implementadas mediante controladores o como parte de las declaraciones de ruta. La sintaxis utilizada para definir la lógica de la aplicación es similar a la utilizada por Sinatra framework.
* El enrutamiento inverso define una relación entre los enlaces y las rutas, lo que nos hace posible que los cambios posteriores en las rutas se propaguen automáticamente en enlaces relevantes. Cuando los vínculos se crean mediante el uso de nombres de rutas existentes, los apropiados identificadores de recursos uniformes (URIs) son creados por Laravel.
* Restful Controllers proporcionan una forma opcional de separar la lógica de servicios requeridos HTTP GET and POST.
* Class Auto Loading Providers carga de forma automática las clases PHP sin necesidad de mantenimiento manual de inclusión de rutas. On-demand loading prevents impide la inclusión de componentes innecesarios, por lo que solo se cargan los componentes realmente necesarios.
* View composers serve son unidades de código lógico que pueden ser ejecutadas cuando se carga una vista.
* Blade es una plantilla ingeniosa que combina uno o más plantillas con un modelo de datos para obtener como resultado vistas, haciendo que transmita las plantillas en código PHP para mejorar el rendimiento. Blade también proporciona un conjunto de sus propias estructuras de control tales como declaración de condicionales y bucles, que se asignan internamente a PHP. Además, los servicios de Laravel pueden ser llamados a partir de plantillas Blade, y su motor de plantillas se puede ampliar con directivas personalizadas.
* IoC containers permiten generar objetos nuevos siguiendo el principio de control de inversión (IoC), en el que el framework llama al código de aplicación o tarea específica, con la instanciación y referencia opcional de objetos nuevos.
* Migrations provide proporciona un sistema de control de versiones para el esquema de la base de datos, lo que permite asociar cambios en la base del código de la aplicación y los cambios necesarios en el diseño de la bases de datos. Como resultado, la característica simplifica el despliegue y la actualización de las aplicaciones basadas en Laravel.
* Database seeding provides proporciona una forma de rellenar las tablas de bases de datos con los datos predeterminados que se pueden utilizar para las pruebas de aplicación que se realizaran como parte de la configuración inicial de la aplicación.
* Unit testting proporciona como parte integral de Laravel, que contiene pruebas unitarias que detectan y previenen regresiones en el framework. Las pruebas de unidad se pueden ejecutar a través de la línea de comando de artisan provider.
* Automatic pagination simplifica la tarea de implementar la paginación, reemplazando métodos habituales de implementación manual con métodos automatizados integrados a Laravel.
* Form Request es una característica de Laravel, que sirve como base para la validación de la entrada de formularios mediante los eventos de listeners, dando como resultado la invocación automática de métodos de validación de formularios y la generación de un formulario actual.
* Homestead – Vagrant Virtual Machine que proporciona a los desarrolladores de Laravel todas las herramientas necesarias para desarrollar Laravel.
  + 1. **ARTISAN CLI**

Laravel’s command-line interface (CLI), llamado Artisan, fue introducida en la Laravel 3 con un conjunto limitado de capacidades. La posterior migración de Laravel en una arquitectura basada en Composer permitió a Artisan incorporar diferentes componentes del framework Symfony, resultado es la disponibilidad de características adicionales de Artisan en Laravel 4.

Las características de Artisan se asignan a diferentes subcomandos de la utilidad artisan command-line, proporcionando funcionalidad que ayuda en la gestión y la construcción de aplicaciones basadas en Laravel. Los usos más comunes de Artisan incluyen la gestión de migraciones de bases de datos, generando código estandarizado para nuevos controllers y migrations; Este último libera al desarrollador de crear estructura de código adecuado. La funcionalidad y las capacidades de Artisan también se pueden aplicar mediante la implantación de nuevos comandos personalizados, que puedes utilizar para automatizar las tareas recurrentes específicas de la aplicación.

* 1. **BOOTSTRAP**
     1. **HISTORIA**

Bootstrap fue desarrollado por Mark Otto y Jacbod Thornton de Twitter, con un marco de trabajo para fomentar la consistencia entre las herramientas internas. Antes de Bootstrap, se usaba varias librerías para el desarrollo de interfaces de usuario, las cuales llevan a inconsistencias y a una gran carga de trabajo en su mantenimiento.

El primer desarrollo en condiciones reales ocurrió durante la primera “semana de Hackeo” de Twitter. Marck Otto mostro a algunos colegas como acelerar el desarrollo de sus proyectos con la ayuda de la herramienta de trabajo. Como resultado, decenas de temas se han introducido en el marco de trabajo.

* + 1. **CARACTERISTICAS**

Bootstrap tiene un soporte relativamente incompleto para [HTML5](https://es.wikipedia.org/wiki/HTML5) y [CSS 3](https://es.wikipedia.org/wiki/CSS_3), pero es compatible con la mayoría de los navegadores web. La información básica de compatibilidad de sitios web o aplicaciones está disponible para todos los dispositivos y navegadores. Existe un concepto de compatibilidad parcial que hace disponible la información básica de un sitio web para todos los dispositivos y navegadores. Por ejemplo, las propiedades introducidas en CSS3 para las esquinas redondeadas, gradientes y sombras son usadas por Bootstrap a pesar de la falta de soporte de navegadores antiguos. Esto extiende la funcionalidad de la herramienta, pero no es requerida para su uso.

Desde la versión 2.0 también soporta [diseños sensibles](https://es.wikipedia.org/w/index.php?title=Responsive_Web_Design&action=edit&redlink=1). Esto significa que el diseño gráfico de la página se ajusta dinámicamente, tomando en cuenta las características del dispositivo usado (Computadoras, tabletas, teléfonos significa que el diseño gráfico de la página se ajusta dinámicamente móviles). Bootstrap es de código abierto y está disponible en GitHub.

* + 1. **ESTRUCTURA Y FUNCION**

Bootstrap es modular y consiste esencialmente en una serie de [hojas de estilo LESS](https://es.wikipedia.org/wiki/LESS_(lenguaje_de_hojas_de_estilo)) que implementan la variedad de componentes de la herramienta. Una hoja de estilo llamada bootstrap.less incluye los componentes de las hojas de estilo. Los desarrolladores pueden adaptar el mismo archivo de Bootstrap, seleccionando los componentes que deseen usar en su proyecto.

Los ajustes son posibles en una medida limitada a través de una hoja de estilo de configuración central. Los cambios más profundos son posibles mediante las declaraciones LESS.

* + - 1. **SISTEMA DE CUADRILLA Y DISEÑO SENCIBLE**

Bootstrap viene con una disposición de cuadrilla estándar de 940 píxeles de ancho. Alternativamente, el desarrollador puede usar un diseño de ancho-variable. Para ambos casos, la herramienta tiene cuatro variaciones para hacer uso de distintas resoluciones y tipos de dispositivos: teléfonos móviles, formato de retrato y paisaje, tabletas y computadoras con baja y alta resolución (pantalla ancha). Esto ajusta el ancho de las columnas automáticamente.

* + - 1. **ENTENDIENDO LA HOJA DE ESTILOS CSS**

Bootstrap proporciona un conjunto de hojas de estilo que proveen definiciones básicas de estilo para todos los componentes de HTML. Esto otorga una uniformidad al navegador y al sistema de anchura, da una apariencia moderna para el formateo de los elementos de texto, tablas y formularios.

* + - 1. **COMPONENTES RE-USABLES**

En adición a los elementos regulares de HTML, Bootstrap contiene otra interfaz de elementos comúnmente usados. Ésta incluye botones con características avanzadas, etiquetas, capacidades avanzadas de miniaturas tipográficas, formatos para mensajes de alerta y barras de progreso.

* + - 1. **PLUG-INS DE JAVASCRIPT**

Los componentes de JavaScript para Bootstrap están basados en la librería [jQuery](https://es.wikipedia.org/wiki/JQuery) de JavaScript. Los plug-ins se encuentran en la herramienta de plug-in de jQuery. Proveen elementos adicionales de interfaz de usuario como diálogos, tooltips y carruseles.

* 1. **PHP**
     1. **HISTORIA**

Fue originalmente diseñado en [Perl](https://es.wikipedia.org/wiki/Perl), con base en la escritura de un grupo de [CGI](https://es.wikipedia.org/wiki/Interfaz_de_entrada_com%C3%BAn) binarios escritos en el [lenguaje C](https://es.wikipedia.org/wiki/C_(lenguaje_de_programaci%C3%B3n)) por el [programador](https://es.wikipedia.org/wiki/Programador) danés-canadiense [Rasmus Lerdorf](https://es.wikipedia.org/wiki/Rasmus_Lerdorf) en el año 1994 para mostrar su [currículum vítae](https://es.wikipedia.org/wiki/Curr%C3%ADculum_v%C3%ADtae) y guardar ciertos datos, como la cantidad de tráfico que su página web recibía. El 8 de junio de 1995 fue publicado "Personal Home Page Tools" después de que Lerdorf lo combinara con su propio *Form Interpreter* para crear PHP/FI.

Dos programadores [israelíes](https://es.wikipedia.org/wiki/Israel) del [Technion](https://es.wikipedia.org/wiki/Technion), [Zeev Suraski](https://es.wikipedia.org/wiki/Zeev_Suraski) y [Andi Gutmans](https://es.wikipedia.org/wiki/Andi_Gutmans), reescribieron el [analizador sintáctico](https://es.wikipedia.org/wiki/Analizador_sint%C3%A1ctico) (*parser*, en [inglés](https://es.wikipedia.org/wiki/Idioma_ingl%C3%A9s)) en 1997 y crearon la base del PHP3, y cambiaron el nombre del lenguaje por *PHP: Hypertext Preprocessor*.[1](https://es.wikipedia.org/wiki/PHP#cite_note-historia-1)​ Inmediatamente comenzaron experimentaciones públicas de PHP3, y se publicó oficialmente en junio de 1998. Para 1999, Suraski y Gutmans reescribieron el código de PHP, y produjeron lo que hoy se conoce como [motor Zend](https://es.wikipedia.org/wiki/Motor_Zend). También fundaron [Zend Technologies](https://es.wikipedia.org/wiki/Zend_Technologies) en [Ramat Gan](https://es.wikipedia.org/wiki/Ramat_Gan), [Israel](https://es.wikipedia.org/wiki/Israel).[1](https://es.wikipedia.org/wiki/PHP#cite_note-historia-1)​

En mayo del 2000, PHP 4 se lanzó bajo el poder del [motor Zend](https://es.wikipedia.org/wiki/Motor_Zend) 1.0. El 13 de julio del 2007 se anunció la suspensión del soporte y desarrollo de la versión 4 de PHP,[10](https://es.wikipedia.org/wiki/PHP#cite_note-10)​ y, a pesar de lo anunciado, se ha liberado una nueva versión con mejoras de seguridad, la 4.4.8, publicada el 13 de enero del 2008, y posteriormente la versión 4.4.9, publicada el 7 de agosto del 2008.[11](https://es.wikipedia.org/wiki/PHP#cite_note-11)​ Según esta noticia,[12](https://es.wikipedia.org/wiki/PHP#cite_note-12)​ se le dio soporte a fallos críticos hasta el 9 de agosto del 2008.

El 13 de julio del 2004, se lanzó PHP 5, utilizando el motor Zend Engine 2.0 (o Zend Engine 2).[1](https://es.wikipedia.org/wiki/PHP#cite_note-historia-1)​ Incluye todas las ventajas que provee el nuevo Zend Engine 2, como:

* Mejor soporte para la [programación orientada a objetos](https://es.wikipedia.org/wiki/Programaci%C3%B3n_orientada_a_objetos), que en versiones anteriores era extremadamente rudimentario.
* Mejoras de rendimiento
* Mejor soporte para [MySQL](https://es.wikipedia.org/wiki/MySQL) con extensión completamente reescrita
* Mejor soporte a [XML](https://es.wikipedia.org/wiki/Extensible_Markup_Language) (XPath, DOM, etc.)
* Soporte nativo para [SQLite](https://es.wikipedia.org/wiki/SQLite)
* Soporte integrado para [SOAP](https://es.wikipedia.org/wiki/Simple_Object_Access_Protocol)
* [Iteradores](https://es.wikipedia.org/wiki/Iterador_(patr%C3%B3n_de_dise%C3%B1o)) de datos
* [Manejo de excepciones](https://es.wikipedia.org/wiki/Manejo_de_excepciones)
* Mejoras con la implementación con [Oracle](https://es.wikipedia.org/wiki/Oracle_Database)
  + 1. **CARACTERISTICAS**
* Orientado al desarrollo de [aplicaciones web](https://es.wikipedia.org/wiki/Aplicaci%C3%B3n_web) dinámicas con acceso a información almacenada en una [base de datos](https://es.wikipedia.org/wiki/Base_de_datos).
* Es considerado un lenguaje fácil de aprender, ya que en su desarrollo se simplificaron distintas especificaciones, como es el caso de la definición de las variables primitivas, ejemplo que se hace evidente en el uso de [php arrays](https://es.wikipedia.org/wiki/Php_arrays).
* El código fuente escrito en PHP es invisible al [navegador web](https://es.wikipedia.org/wiki/Navegador_web) y al cliente, ya que es el servidor el que se encarga de ejecutar el código y enviar su resultado HTML al navegador.
* Capacidad de conexión con la mayoría de los motores de base de datos que se utilizan en la actualidad, destaca su conectividad con [MySQL](https://es.wikipedia.org/wiki/MySQL) y [PostgreSQL](https://es.wikipedia.org/wiki/PostgreSQL).
* Capacidad de expandir su potencial utilizando módulos (llamados *ext's* o extensiones).
* Posee una amplia documentación en su sitio web oficial, entre la cual se destaca que todas las funciones del sistema están explicadas y ejemplificadas en un único archivo de ayuda.
* Es [libre](https://es.wikipedia.org/wiki/Software_libre), por lo que se presenta como una alternativa de fácil acceso para todos.
* Permite aplicar técnicas de [programación orientada a objetos](https://es.wikipedia.org/wiki/Programaci%C3%B3n_orientada_a_objetos).
* No requiere definición de tipos de variables aunque sus variables se pueden evaluar también por el tipo que estén manejando en tiempo de ejecución.
* Tiene manejo de excepciones (desde PHP5).
* Si bien PHP no obliga a quien lo usa a seguir una determinada metodología a la hora de programar, aun haciéndolo, el programador[34](https://es.wikipedia.org/wiki/PHP#cite_note-34)​ puede aplicar en su trabajo cualquier técnica de programación o de desarrollo que le permita escribir código ordenado, estructurado y manejable. Un ejemplo de esto son los desarrollos que en PHP se han hecho del [patrón de diseño](https://es.wikipedia.org/wiki/Patr%C3%B3n_de_dise%C3%B1o) [Modelo Vista Controlador](https://es.wikipedia.org/wiki/Modelo_Vista_Controlador) (MVC), que permiten separar el tratamiento y acceso a los datos, la [lógica de control](https://es.wikipedia.org/wiki/L%C3%B3gica_de_control) y la [interfaz de usuario](https://es.wikipedia.org/wiki/Interfaz_de_usuario) en tres componentes independientes.
* Debido a su flexibilidad ha tenido una gran acogida como lenguaje base para las aplicaciones WEB de manejo de contenido, y es su uso principal.
  1. **JQUERY**
     1. **HISTORIA**

**jQuery** es una [biblioteca](https://es.wikipedia.org/wiki/Biblioteca_(inform%C3%A1tica)) [multiplataforma](https://es.wikipedia.org/wiki/Multiplataforma) de [JavaScript](https://es.wikipedia.org/wiki/JavaScript), creada inicialmente por [John Resig](https://es.wikipedia.org/wiki/John_Resig), que permite simplificar la manera de interactuar con los documentos [HTML](https://es.wikipedia.org/wiki/HTML), manipular el árbol [DOM](https://es.wikipedia.org/wiki/Document_Object_Model), manejar eventos, desarrollar animaciones y agregar interacción con la técnica [AJAX](https://es.wikipedia.org/wiki/AJAX) a páginas web. Fue presentada el 14 de enero de 2006 en el [BarCamp](https://es.wikipedia.org/wiki/BarCamp) NYC. jQuery es la biblioteca de JavaScript más utilizada.[1](https://es.wikipedia.org/wiki/JQuery#cite_note-1)​ ​

**jQuery** es [software libre y de código abierto](https://es.wikipedia.org/wiki/Software_libre_y_de_c%C3%B3digo_abierto), posee un doble licenciamiento bajo la [Licencia MIT](https://es.wikipedia.org/wiki/MIT_License) y la [Licencia Pública General de GNU](https://es.wikipedia.org/wiki/GNU_General_Public_License) v2, permitiendo su uso en proyectos [libres](https://es.wikipedia.org/wiki/Software_libre) y [privados](https://es.wikipedia.org/wiki/Software_propietario).[2](https://es.wikipedia.org/wiki/JQuery#cite_note-2)​ ​ jQuery, al igual que otras bibliotecas, ofrece una serie de funcionalidades basadas en JavaScript que de otra manera requerirían de mucho más código, es decir, con las funciones propias de esta biblioteca se logran grandes resultados en menos tiempo y espacio.

* + 1. **CARACTETISTICAS**

**jQuery** consiste en un único fichero JavaScript que contiene las funcionalidades comunes de DOM, eventos, efectos y AJAX.

* Selección de elementos [DOM](https://es.wikipedia.org/wiki/Document_Object_Model).
* Interactividad y modificaciones del árbol DOM, incluyendo soporte para [CSS 1-3](https://es.wikipedia.org/wiki/Hojas_de_estilo_en_cascada) y un [plugin](https://es.wikipedia.org/wiki/Complemento_(inform%C3%A1tica)) básico de [XPath](https://es.wikipedia.org/wiki/XPath).
* Eventos.
* Manipulación de la hoja de estilos CSS.
* Efectos y animaciones.
* Animaciones personalizadas.
* [AJAX](https://es.wikipedia.org/wiki/AJAX).
* Soporta [extensiones](https://es.wikipedia.org/wiki/Complemento_(inform%C3%A1tica)).
* Utilidades varias como obtener información del [navegador](https://es.wikipedia.org/wiki/Navegador_web), operar con [objetos](https://es.wikipedia.org/wiki/Objeto_(programaci%C3%B3n)) y [vectores](https://es.wikipedia.org/wiki/Vector_(inform%C3%A1tica)), funciones para rutinas comunes, entre otros.
* Compatible con los navegadores [Mozilla Firefox](https://es.wikipedia.org/wiki/Mozilla_Firefox) 2.0+, [Internet Explorer](https://es.wikipedia.org/wiki/Internet_Explorer) 6+, [Safari](https://es.wikipedia.org/wiki/Safari_(navegador)) 3+, [Opera](https://es.wikipedia.org/wiki/Opera_(navegador)) 10.6+ y [Google Chrome](https://es.wikipedia.org/wiki/Google_Chrome) 8+.[5](https://es.wikipedia.org/wiki/JQuery#cite_note-5)​ ​
  1. **APACHE** 
     1. **HISTORIA**

El **servidor HTTP Apache** es un [servidor web](https://es.wikipedia.org/wiki/Servidor_web) [HTTP](https://es.wikipedia.org/wiki/Hypertext_Transfer_Protocol) de [código abierto](https://es.wikipedia.org/wiki/Software_de_c%C3%B3digo_abierto), para plataformas [Unix](https://es.wikipedia.org/wiki/Unix) ([BSD](https://es.wikipedia.org/wiki/BSD), [GNU/Linux](https://es.wikipedia.org/wiki/GNU/Linux), etc.), [Microsoft Windows](https://es.wikipedia.org/wiki/Microsoft_Windows), [Macintosh](https://es.wikipedia.org/wiki/Macintosh) y otras, que implementa el protocolo HTTP/1.1[2](https://es.wikipedia.org/wiki/Servidor_HTTP_Apache#cite_note-2)​ y la noción de sitio virtual. Cuando comenzó su desarrollo en 1995 se basó inicialmente en código del popular [NCSA HTTPd](https://es.wikipedia.org/wiki/NCSA_HTTPd) 1.3, pero más tarde fue reescrito por completo. Su nombre se debe a que alguien quería que tuviese la connotación de algo que es firme y enérgico pero no agresivo, y la tribu Apache fue la última en rendirse al que pronto se convertiría en gobierno de EEUU, y en esos momentos la preocupación de su grupo era que llegasen las empresas y "civilizasen" el paisaje que habían creado los primeros ingenieros de internet. Además Apache consistía solamente en un conjunto de parches a aplicar al servidor de NCSA. En inglés, *a patchy server* (un servidor "parcheado") suena igual que *Apache Server*.

El servidor Apache es desarrollado y mantenido por una comunidad de usuarios bajo la supervisión de la [Apache Software Foundation](https://es.wikipedia.org/wiki/Apache_Software_Foundation) dentro del proyecto HTTP Server (httpd).

Apache presenta entre otras características altamente configurables, bases de datos de autenticación y negociado de contenido, pero fue criticado por la falta de una interfaz gráfica que ayude en su configuración.

Apache tiene amplia aceptación en la red: desde 1996, Apache, es el servidor HTTP más usado. Jugó un papel fundamental en el desarrollo fundamental de la World Wide Web y alcanzó su máxima cuota de mercado en [2005](https://es.wikipedia.org/wiki/2005) siendo el servidor empleado en el 70% de los sitios web en el mundo, sin embargo ha sufrido un descenso en su cuota de mercado en los últimos años. En 2009 se convirtió en el primer servidor web que alojó más de 100 millones de sitios web.[4](https://es.wikipedia.org/wiki/Servidor_HTTP_Apache#cite_note-100millionsites-4)​

La mayoría de las vulnerabilidades de la seguridad descubiertas y resueltas tan sólo pueden ser aprovechadas por usuarios locales y no remotamente. Sin embargo, algunas se pueden accionar remotamente en ciertas situaciones, o explotar por los usuarios locales malévolos en las disposiciones de recibimiento compartidas que utilizan PHP como módulo de Apache.

* + 1. **CARACTERISTICAS**

Apache es usado principalmente para enviar páginas web estáticas y dinámicas en la World Wide Web. Muchas aplicaciones web están diseñadas asumiendo como ambiente de implantación a Apache, o que utilizarán características propias de este servidor web.

Apache es el componente de servidor web en la popular plataforma de aplicaciones [XAMP](https://es.wikipedia.org/wiki/LAMP), junto a [MySQL](https://es.wikipedia.org/wiki/MySQL) y los lenguajes de programación [PHP](https://es.wikipedia.org/wiki/PHP)/[Perl](https://es.wikipedia.org/wiki/Perl)/[Python](https://es.wikipedia.org/wiki/Python) (y ahora también [Ruby](https://es.wikipedia.org/wiki/Ruby)).

Este servidor web es redistribuido como parte de varios paquetes propietarios de software, incluyendo la base de datos [Oracle](https://es.wikipedia.org/wiki/Oracle) y el IBM [WebSphere](https://es.wikipedia.org/wiki/WebSphere) application server. Mac OS X integra apache como parte de su propio servidor web y como soporte de su servidor de aplicaciones [WebObjects](https://es.wikipedia.org/w/index.php?title=WebObjects&action=edit&redlink=1). Es soportado de alguna manera por [Borland](https://es.wikipedia.org/wiki/Borland) en las herramientas de desarrollo Kylix y Delphi. Apache es incluido con Novell NetWare 6.5, donde es el servidor web por defecto, y en muchas distribuciones Linux.

Apache es usado para muchas otras tareas donde el contenido necesita ser puesto a disposición en una forma segura y confiable. Un ejemplo es al momento de compartir archivos desde una computadora personal hacia Internet. Un usuario que tiene Apache instalado en su escritorio puede colocar arbitrariamente archivos en la raíz de documentos de Apache, desde donde pueden ser compartidos.

Los programadores de aplicaciones web a veces utilizan una versión local de Apache con el fin de pre visualizar y probar código mientras éste es desarrollado.

Microsoft [Internet Information Services](https://es.wikipedia.org/wiki/Internet_Information_Services) (IIS) es el principal competidor de Apache, así como Sun Java System Web Server de [Sun Microsystems](https://es.wikipedia.org/wiki/Sun_Microsystems) y un anfitrión de otras aplicaciones como Zeus Web Server.

* 1. **MYSQL**
     1. **HISTORIA**

**MySQL** es un [sistema de gestión de bases de datos](https://es.wikipedia.org/wiki/Sistema_de_gesti%C3%B3n_de_bases_de_datos) [relacional](https://es.wikipedia.org/wiki/Modelo_relacional) desarrollado bajo licencia dual [GPL](https://es.wikipedia.org/wiki/Licencia_p%C3%BAblica_general_de_GNU)/[Licencia comercial](https://es.wikipedia.org/wiki/Software_propietario) por [Oracle Corporation](https://es.wikipedia.org/wiki/Oracle_Corporation) y está considerada como la base datos [open source](https://es.wikipedia.org/wiki/Open_source) más popular del mundo,[1](https://es.wikipedia.org/wiki/MySQL#cite_note-1)​ ​[2](https://es.wikipedia.org/wiki/MySQL#cite_note-2)​ ​ y una de las más populares en general junto a [Oracle](https://es.wikipedia.org/wiki/Oracle_Database) y [Microsoft SQL Server](https://es.wikipedia.org/wiki/Microsoft_SQL_Server), sobre todo para entornos de [desarrollo web](https://es.wikipedia.org/wiki/Desarrollo_web).

MySQL fue inicialmente desarrollado por [MySQL AB](https://es.wikipedia.org/wiki/MySQL_AB) (empresa fundada por [David Axmark](https://es.wikipedia.org/w/index.php?title=David_Axmark&action=edit&redlink=1), [Allan Larsson](https://es.wikipedia.org/w/index.php?title=Allan_Larsson&action=edit&redlink=1) y [Michael Widenius](https://es.wikipedia.org/wiki/Michael_Widenius)). MySQL A.B. fue adquirida por [Sun Microsystems](https://es.wikipedia.org/wiki/Sun_Microsystems) en 2008, y ésta a su vez fue comprada por [Oracle Corporation](https://es.wikipedia.org/wiki/Oracle_Corporation) en 2010, la cual ya era dueña desde 2005 de [Innobase Oy](https://es.wikipedia.org/w/index.php?title=Innobase_Oy&action=edit&redlink=1), empresa [finlandesa](https://es.wikipedia.org/wiki/Finlandia) desarrolladora del motor [InnoDB](https://es.wikipedia.org/wiki/InnoDB) para MySQL.

Al contrario de proyectos como [Apache](https://es.wikipedia.org/wiki/Servidor_HTTP_Apache), donde el software es desarrollado por una comunidad pública y los [derechos de autor](https://es.wikipedia.org/wiki/Derechos_de_autor) del código están en poder del autor individual, MySQL es patrocinado por una [empresa privada](https://es.wikipedia.org/wiki/Empresa_privada), que posee el copyright de la mayor parte del código. Esto es lo que posibilita el esquema de doble licenciamiento anteriormente mencionado. La base de datos se distribuye en varias versiones, una *Community*, distribuida bajo la Licencia pública general de GNU, versión 2, y varias versiones *Enterprise*, para aquellas empresas que quieran incorporarlo en productos privativos. Las versiones *Enterprise* incluyen productos o servicios adicionales tales como herramientas de [monitorización](https://es.wikipedia.org/w/index.php?title=Monitorizaci%C3%B3n_de_sistemas&action=edit&redlink=1) y [soporte](https://es.wikipedia.org/wiki/Soporte_t%C3%A9cnico) oficial. En [2009](https://es.wikipedia.org/wiki/2009) se creó un [*fork*](https://es.wikipedia.org/wiki/Bifurcaci%C3%B3n_(desarrollo_de_software)) denominado [MariaDB](https://es.wikipedia.org/wiki/MariaDB) por algunos desarrolladores (incluido algunos desarrolladores originales de MySQL) descontentos con el modelo de desarrollo y el hecho de que una misma empresa controle a la vez los productos MySQL y [Oracle Database](https://es.wikipedia.org/wiki/Oracle_Database).[3](https://es.wikipedia.org/wiki/MySQL#cite_note-3)​ ​

Está desarrollado en su mayor parte en [ANSI C](https://es.wikipedia.org/wiki/ANSI_C) y [C++](https://es.wikipedia.org/wiki/C%2B%2B).[4](https://es.wikipedia.org/wiki/MySQL#cite_note-4)​ ​ Tradicionalmente se considera uno de los cuatro componentes de la pila de desarrollo [LAMP](https://es.wikipedia.org/wiki/LAMP) y [WAMP](https://es.wikipedia.org/wiki/WAMP).

* + 1. **CARACTERISTICAS**

Inicialmente, MySQL carecía de elementos considerados esenciales en las bases de datos relacionales, tales como [integridad referencial](https://es.wikipedia.org/wiki/Integridad_referencial) y [transacciones](https://es.wikipedia.org/wiki/Transacci%C3%B3n_(base_de_datos)). A pesar de ello, atrajo a los desarrolladores de páginas web con contenido dinámico, justamente por su simplicidad.

Poco a poco los elementos de los que carecía MySQL están siendo incorporados tanto por desarrollos internos, como por desarrolladores de [software libre](https://es.wikipedia.org/wiki/Software_libre). Entre las características disponibles en las últimas versiones se puede destacar:

* Amplio subconjunto del lenguaje [SQL](https://es.wikipedia.org/wiki/SQL). Algunas extensiones son incluidas igualmente.
* Disponibilidad en gran cantidad de plataformas y sistemas.
* Posibilidad de selección de [mecanismos de almacenamiento](https://es.wikipedia.org/wiki/Mecanismos_de_almacenamiento_(MySQL)) que ofrecen diferentes velocidades de operación, soporte físico, capacidad, distribución geográfica, transacciones.
* Transacciones y [claves foráneas](https://es.wikipedia.org/wiki/Clave_for%C3%A1nea).
* Conectividad segura.
* [Replicación](https://es.wikipedia.org/wiki/Replicaci%C3%B3n_(Inform%C3%A1tica)).
* Búsqueda e [indexación](https://es.wikipedia.org/w/index.php?title=Indexar&action=edit&redlink=1) de campos de texto.

MySQL es un sistema de administración de bases de datos. Una base de datos es una colección estructurada de tablas que contienen datos. Para agregar, acceder a y procesar datos guardados en un computador. Dado que los computadores son muy buenos manejando grandes cantidades de información, los administradores de bases de datos juegan un papel central en computación, como aplicaciones independientes o como parte de otras aplicaciones.

MySQL es un sistema de administración relacional de bases de datos. Una base de datos relacional archiva datos en tablas separadas en vez de colocar todos los datos en un gran archivo. Esto permite velocidad y flexibilidad. Las tablas están conectadas por relaciones definidas que hacen posible combinar datos de diferentes tablas sobre pedido.

* 1. **SÍNTESIS**

Para el desarrollo web del sistema se aplicó múltiples tecnologías, entre las más importantes tenemos a: Laravel como framework PHP, Bootstrap para diseño Responsive donde el diseño gráfico de la página se ajusta dinámicamente y JQuery.JS para manejar eventos, desarrollar animaciones y agregar interacción con la técnica [AJAX](https://es.wikipedia.org/wiki/AJAX).

Finalmente, Apache como componente del servidor web con plataforma LAMP y con sistema de gestión de bases de datos MySQL.

*CAPITULO 5*

1. PLANIFICACION Y ORGANIZACION
   1. **INTRODUCCIÓN**

Tomando en cuenta los resultados de la metodología de las 6’D se organizó, planifico y estimo el Desarrollo Solución del producto con el cliente, para luego empezar a ejecutarlo como desarrollador en los diferentes módulos del sistema.

* 1. **REPORTE DE LA ETAPA 03: DISEÑO SOLUCIÓN**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Nº | Módulos Sistema | Origen | Pri. | Estimación Inicial (días) | Factor de Ajuste | Estimación Ajustada (días) |
| 1 | Autentificar Usuario | Mireya |  | 7 |  | 7 |
| 2 | Crear Template Para El Sistema Administración | Mireya |  | 15 |  | 15 |
| 3 | Implementación De Funciones Del Sistema | Mireya |  | 5 |  | 5 |
| 4 | Implementación De Tareas Para Las Funciones | Mireya |  | 5 |  | 5 |
| 5 | Implementación De Rol De Usuario | Mireya |  | 5 |  | 5 |
|  | **Modulo Administración Usuarios** |  |  | **37** |  | **37** |
| 1 | Crear El Menú Principal Por Rol De Usuario | Mireya |  | 10 |  | 10 |
| 2 | Crear Nuevo Mensaje | Mireya |  | 3 |  | 3 |
| 3 | Implementación De Buzón De Mensajes | Mireya |  | 7 |  | 7 |
| 4 | Modificar Perfil De Usuario | Mireya |  | 3 |  | 3 |
| 5 | Implementar Usuarios Registrados | Mireya |  | 5 |  | 5 |
|  | **Modulo Administración Funciones** |  |  | **28** |  | **28** |
| 1 | CRUD De Facultad | Mireya |  | 7 |  | 7 |
| 2 | CRUD De Carrera | Mireya |  | 3 |  | 3 |
| 3 | CRUD De Materia | Mireya |  | 3 |  | 3 |
| 4 | Generación De Dependencias | Mireya |  | 4 |  | 4 |
|  | **Modulo Administración Facultad** |  |  | **17** |  | **17** |
| 1 | CRUD Publicación | Mireya |  | 7 |  | 7 |
| 2 | Administración De Todas Las Publicaciones | Mireya |  | 2 |  | 2 |
| 3 | Habilitar De Tareas De Usuarios De Facultad | Mireya |  | 3 |  | 3 |
| 4 | Habilitar De Tareas De Usuarios De Carrera | Mireya |  | 1 |  | 1 |
| 5 | Habilitar De Tareas De Usuarios De Materia | Mireya |  | 1 |  | 1 |
| 6 | Habilitar De Tareas De Usuarios Por Importancia | Mireya |  | 2 |  | 2 |
| 7 | Habilitación De Tareas De Usuarios Por Tipo Publicación | Mireya |  | 1 |  | 1 |
|  | **Modulo Administración Publicación** |  |  | **17** |  | **17** |
| 1 | Ver estadísticas y reporte de visitas | Mireya |  | 7 |  | 7 |
| 2 | Crear template para el portal | Mireya |  | 3 |  | 3 |
| 3 | Implementar facultad para el portal web | Mireya |  | 1 |  | 1 |
| 4 | Implementar carrera para el portal web | Mireya |  | 2 |  | 2 |
| 5 | Implementar materia para el portal web | Mireya |  | 1 |  | 1 |
| 6 | Implementar unidad de información para el portal web | Mireya |  | 1 |  | 1 |
| 7 | Implementar publicación completa para el portal web | Mireya |  | 1 |  | 1 |
| 8 | Implementar contacto para el portal web | Mireya |  | 1 |  | 1 |
|  | **Modulo Administración Web** |  |  | **17** |  | **17** |
|  | **ENTREGA FINAL** |  |  | **116** |  | **116** |

**REPORTE DE LA ETAPA 04: DESARROLLO SOLUCIÓN**

Para este reporte presentamos las estimaciones y asignaciones de las diferentes tareas.

**Etapa 04: Desarrollo Solución**

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Módulo Administración Usuarios | | | | | | | | | | |
| IMPLENTACION POR MODULO | **TAREA** | **QUIEN** | **ESTADO** | **DIA** | | | | | | |
| **1** | **2** | **3** | **4** | **5** | **6** | **7** |
| Crear Template Para El Sistema Administración | Aprender Laravel | Mireya | Completo | 4 | 4 | 4 | 4 | 4 | 4 | 4 |
| Crear Template Para El Sistema Administración | Aprender AdminLTE | Mireya | Completo | 4 | 4 | 4 | 4 | 4 | 4 | 4 |
| Autentificar Usuario | Aprender Migrate de Laravel | Mireya | Completo | 4 | 4 | 4 | 4 | 8 | 8 | 4 |
| Implementación De Funciones, tareas y Roles Del Sistema | Modelar la base de datos | Mireya | Completo | 8 | 4 | 4 | 4 | 4 |  |  |
| Implementación De Funciones Del Sistema | Crear un menú principal | Mireya | Completo | 4 | 8 | 4 |  |  |  |  |
| Implementación De Funciones Del Sistema | Crear vista | Mireya | Completo | 4 | 4 | 4 |  |  |  |  |
| Implementación De Tareas Para Las Funciones | Crear menús anidados | Mireya | Completo | 4 | 4 |  |  |  |  |  |
| Implementación De Tareas Para Las Funciones | Crear vista | Mireya | Completo | 4 | 4 | 4 |  |  |  |  |
| Implementación De Rol De Usuario | Crear vista | Mireya | Completo | 4 | 4 |  |  |  |  |  |
| Módulo Administración Función | | | | | | | | | | |
| Crear El Menú Principal Por Rol De Usuario | Aprende Jquery.JS | Mireya | Completo | 4 | 4 | 4 | 4 | 4 |  |  |
| Crear El Menú Principal Por Rol De Usuario | Crea Modal | Mireya | Completo | 4 | 4 | 4 | 4 | 4 |  |  |
| Crear Nuevo Mensaje | Modelar la base de datos | Mireya | Completo | 4 | 4 | 4 |  |  |  |  |
| Crear Nuevo Mensaje | Crear Modal | Mireya | Completo | 4 | 4 | 4 |  |  |  |  |
| Implementación De Buzón De Mensajes | Modelar la base de datos | Mireya | Completo | 4 | 4 | 4 |  |  |  |  |
| Implementación De Buzón De Mensajes | Crear Modal | Mireya | Completo | 8 | 8 | 8 |  |  |  |  |
| Modificar Perfil De Usuario | Modelar la base de datos | Mireya | Completo | 8 | 8 |  |  |  |  |  |
| Modificar Perfil De Usuario | Crear Vista | Mireya | Completo | 8 |  |  |  |  |  |  |
| Implementar Usuarios Registrados | Crear Vista | Mireya | Completo | 8 | 4 | 8 | 8 | 4 |  |  |
| Módulo Administración Facultad | | | | | | | | | | |
| CRUD De Facultad | Modelar la base de datos | Mireya | Completo | 8 | 8 | 8 | 8 |  |  |  |
| CRUD De Facultad | Crear Vista | Mireya | Completo | 8 | 8 | 8 | 8 | 8 | 8 |  |
| CRUD De Carrera | Modelar la base de datos | Mireya | Completo | 8 |  |  |  |  |  |  |
| CRUD De Carrera | Crear Vista | Mireya | Completo | 8 | 8 |  |  |  |  |  |
| CRUD De Materia | Modelar la base de datos | Mireya | Completo | 8 |  |  |  |  |  |  |
| CRUD De Materia | Crear Vista | Mireya | Completo | 8 | 8 |  |  |  |  |  |
| Generación De Dependencias | Modelar la base de datos | Mireya | Completo | 8 |  |  |  |  |  |  |
| Generación De Dependencias | Crear Vista | Mireya | Completo | 8 | 8 | 8 |  |  |  |  |
| Módulo Administración Publicación | | | | | | | | | | |
| CRUD Publicación | Modelar la base de datos | Mireya | Completo | 8 | 8 | 8 |  |  |  |  |
| CRUD Publicación | Crear Vista | Mireya | Completo | 8 | 8 | 8 | 8 |  |  |  |
| Administración De Todas Las Publicaciones | Crear Vista | Mireya | Completo | 8 | 8 |  |  |  |  |  |
| Habilitar De Tareas De Usuarios De Facultad | Crear Vista | Mireya | Completo | 8 | 8 | 8 |  |  |  |  |
| Habilitar De Tareas De Usuarios De Carrera | Crear Vista | Mireya | Completo | 8 |  |  |  |  |  |  |
| Habilitar De Tareas De Usuarios De Materia | Crear Vista | Mireya | Completo | 8 |  |  |  |  |  |  |
| Habilitar De Tareas De Usuarios Por Importancia | Crear Vista | Mireya | Completo | 8 | 8 |  |  |  |  |  |
| Habilitación De Tareas De Usuarios Por Tipo Publicación | Crear Vista | Mireya | Completo | 8 |  |  |  |  |  |  |
| Módulo Administración Web | | | | | | | | | | |
| Ver estadísticas y reporte de visitas | Modelar la base de datos | Mireya | Completo | 8 | 8 | 8 |  |  |  |  |
| Ver estadísticas y reporte de visitas | Crear Vista | Mireya | Completo | 8 | 8 | 8 | 8 |  |  |  |
| Crear template para el portal | Aprender Degree | Mireya | Completo | 8 | 8 | 8 |  |  |  |  |
| Implementar facultad para el portal web | Crear Vista | Mireya | Completo | 8 |  |  |  |  |  |  |
| Implementar carrera para el portal web | Crear Vista | Mireya | Completo | 8 | 8 |  |  |  |  |  |
| Implementar materia para el portal web | Crear Vista | Mireya | Completo | 8 |  |  |  |  |  |  |
| Implementar unidad de información para el portal web | Crear Vista | Mireya | Completo | 8 |  |  |  |  |  |  |
| Implementar publicación completa para el portal web | Crear Vista | Mireya | Completo | 8 |  |  |  |  |  |  |
| Implementar contacto para el portal web | Crear Vista | Mireya | Completo | 8 |  |  |  |  |  |  |

* 1. **SÍNTESIS**

El sistema se desarrolló en un periodo de ciento dieciséis días calendario.

Pero, para colocarlo en modo producción nos tomó quince días calendario, ya que los servidores de Universidad Mayor de San Simón no contaban con las características técnicas especificadas para la implementación del sistema. Por lo que se optó con la instalación y configuración de un servidor propio de servicios web para la Facultad Politécnica del Valle Alto.

*CAPITULO 6*

1. MODELOS FINALES DEL SISTEMA
   1. **INTRODUCCIÓN**

En el presente capitulo abordaremos los principales aspectos del modelado de la base de datos del sistema. Ya que Laravel hace uso de Modelos y un Mapeo Objeto – Relacional (ORM) llamado Eloquent.

Eloquent hace uso de Modelos para recibir y enviar la información a la base de datos. Los Modelos usan convenciones para que a Laravel se le facilite el trabajo y nos ahorre tanto en líneas de código como en tiempo para relacionar modelos.

Por lo cual, los Modelos se escribieron en singular, en contraste a las tablas de la Base de Datos que se escribieron en plural.

* 1. **MODELO DE BASE DE DATOS**

En la ilustración 6.1 mostramos diseño de la base de datos con un modelo Entidad Relación (ER), la misma muestra toda la estructura de tablas del sistema.



Ilustración 6.1 Modelo de la Base de Datos del Sistema de Manejo y Difusión de Información de la Facultad Politécnica del Valle Alto.

* 1. **DICCIONARIO DE DATOS DEL SISTEMA**

En cuanto al diccionario de datos a continuación mostramos el listado de todos los datos que pertenecen al sistema, dándonos precisión de datos de entrada, salida, componentes de almacén, flujos, detalles de las relaciones entre almacenes, entre otros.

C:\Users\Mire\Desktop\DIC_htm_images\T6301_Z100_0002.gif

**carreras**

   Field         Type               Extra

P  id\_carrera    int(10) UNSIGNED   Auto Increment

   id\_facultad   int(10) UNSIGNED

   carrera       varchar(191)

   codigo        varchar(191)       Allow Null

   mision        text               Allow Null

   vision        text               Allow Null

   proyeccion    text               Allow Null

   autoridad     text               Allow Null

   logo          varchar(191)       Allow Null

   detalle       text               Allow Null

   log           int(11)

   estado        varchar(191)

   created\_at    timestamp          Allow Null

   updated\_at    timestamp          Allow Null

C:\Users\Mire\Desktop\DIC_htm_images\T6301_Z100_0002.gif

   Index                          Fields        Extra

   carreras\_carrera\_unique        carrera       Unique

   carreras\_id\_facultad\_foreign   id\_facultad

C:\Users\Mire\Desktop\DIC_htm_images\T6301_Z100_0002.gif

**cpublicaciones**

   Field             Type               Extra

P  id\_cpublicacion   int(10) UNSIGNED   Auto Increment

   titulo            varchar(191)

   detalle           varchar(191)

   contenido         text

   etiqueta          varchar(191)

   publicador        int(11)

   fecha\_inicio      date

   fecha\_fin         date

   id\_importancia    int(10) UNSIGNED

   id\_carrera        int(10) UNSIGNED

   id\_ftp            int(10) UNSIGNED

   estado            varchar(191)

   log               int(11)

   created\_at        timestamp          Allow Null

   updated\_at        timestamp          Allow Null

C:\Users\Mire\Desktop\DIC_htm_images\T6301_Z100_0002.gif

   Index                                   Fields           Extra

   cpublicaciones\_id\_carrera\_foreign       id\_carrera

   cpublicaciones\_id\_ftp\_foreign           id\_ftp

   cpublicaciones\_id\_importancia\_foreign   id\_importancia

C:\Users\Mire\Desktop\DIC_htm_images\T6301_Z100_0002.gif

**dependencias**

   Field            Type               Extra

P  id\_dependencia   int(10) UNSIGNED   Auto Increment

   id\_materia       int(10) UNSIGNED

   id\_previa        int(11)

   log              int(11)

   created\_at       timestamp          Allow Null

   updated\_at       timestamp          Allow Null

C:\Users\Mire\Desktop\DIC_htm_images\T6301_Z100_0002.gif

   Index                             Fields       Extra

   dependencias\_id\_materia\_foreign   id\_materia

C:\Users\Mire\Desktop\DIC_htm_images\T6301_Z100_0002.gif

**docentes**

   Field        Type               Extra

P  id\_docente   int(10) UNSIGNED   Auto Increment

   id           int(10) UNSIGNED

   tipo         varchar(191)

   telefono     varchar(191)       Allow Null

   educacion    varchar(191)       Allow Null

   biografia    text               Allow Null

   foto         varchar(191)       Allow Null

   detalle      varchar(191)       Allow Null

   log          int(11)

   estado       varchar(191)

   created\_at   timestamp          Allow Null

   updated\_at   timestamp          Allow Null

C:\Users\Mire\Desktop\DIC_htm_images\T6301_Z100_0002.gif

   Index                  Fields   Extra

   docentes\_tipo\_unique   tipo     Unique

   docentes\_id\_foreign    id

C:\Users\Mire\Desktop\DIC_htm_images\T6301_Z100_0002.gif

**docentes\_materias**

   Field                  Type               Extra

P  id\_docentes\_materias   int(10) UNSIGNED   Auto Increment

   id\_docente             int(10) UNSIGNED

   id\_materia             int(10) UNSIGNED

   log                    int(11)

   created\_at             timestamp          Allow Null

   updated\_at             timestamp          Allow Null

C:\Users\Mire\Desktop\DIC_htm_images\T6301_Z100_0002.gif

   Index                                  Fields       Extra

   docentes\_materias\_id\_docente\_foreign   id\_docente

   docentes\_materias\_id\_materia\_foreign   id\_materia

C:\Users\Mire\Desktop\DIC_htm_images\T6301_Z100_0002.gif

**etiquetas**

   Field         Type               Extra

P  id\_etiqueta   int(10) UNSIGNED   Auto Increment

   etiqueta      varchar(191)

   log           int(11)

   created\_at    timestamp          Allow Null

   updated\_at    timestamp          Allow Null

C:\Users\Mire\Desktop\DIC_htm_images\T6301_Z100_0002.gif

**facultades**

   Field         Type               Extra

P  id\_facultad   int(10) UNSIGNED   Auto Increment

   facultad      varchar(191)

   universidad   varchar(191)

   telefono      varchar(191)       Allow Null

   fax           varchar(191)       Allow Null

   web           varchar(191)       Allow Null

   direccion     varchar(191)       Allow Null

   autoridad     text               Allow Null

   mision        text               Allow Null

   vision        text               Allow Null

   historia      text               Allow Null

   logo          varchar(191)       Allow Null

   detalle       text               Allow Null

   log           int(11)

   estado        varchar(191)

   created\_at    timestamp          Allow Null

   updated\_at    timestamp          Allow Null

C:\Users\Mire\Desktop\DIC_htm_images\T6301_Z100_0002.gif

   Index                        Fields     Extra

   facultades\_facultad\_unique   facultad   Unique

C:\Users\Mire\Desktop\DIC_htm_images\T6301_Z100_0002.gif

**files**

   Field        Type               Extra

P  id\_file      int(10) UNSIGNED   Auto Increment

   titulo       varchar(191)

   detalle      varchar(191)

   direccion    varchar(191)

   tabla        varchar(191)

   id\_tabla     int(11)

   estado       varchar(191)

   log          int(11)

   created\_at   timestamp          Allow Null

   updated\_at   timestamp          Allow Null

C:\Users\Mire\Desktop\DIC_htm_images\T6301_Z100_0002.gif

**fpublicaciones**

   Field             Type               Extra

P  id\_fpublicacion   int(10) UNSIGNED   Auto Increment

   titulo            varchar(191)

   detalle           varchar(191)

   contenido         text

   etiqueta          varchar(191)

   publicador        int(11)

   fecha\_inicio      date

   fecha\_fin         date

   tabla             varchar(191)

   id\_tabla          int(11)

   area              varchar(191)

   id\_importancia    int(10) UNSIGNED

   id\_ftp            int(10) UNSIGNED

   estado            varchar(191)

   log               int(11)

   created\_at        timestamp          Allow Null

   updated\_at        timestamp          Allow Null

C:\Users\Mire\Desktop\DIC_htm_images\T6301_Z100_0002.gif

   Index                                   Fields           Extra

   fpublicaciones\_id\_ftp\_foreign           id\_ftp

   fpublicaciones\_id\_importancia\_foreign   id\_importancia

C:\Users\Mire\Desktop\DIC_htm_images\T6301_Z100_0002.gif

**fpublicaciones\_etiquetas**

   Field             Type               Extra

P  id\_fpe            int(10) UNSIGNED   Auto Increment

   id\_fpublicacion   int(10) UNSIGNED

   id\_etiqueta       int(10) UNSIGNED

   log               int(11)

   created\_at        timestamp          Allow Null

   updated\_at        timestamp          Allow Null

C:\Users\Mire\Desktop\DIC_htm_images\T6301_Z100_0002.gif

   Index                                              Fields            Extra

   fpublicaciones\_etiquetas\_id\_fpublicacion\_foreign   id\_fpublicacion

   fpublicaciones\_etiquetas\_id\_etiqueta\_foreign       id\_etiqueta

C:\Users\Mire\Desktop\DIC_htm_images\T6301_Z100_0002.gif

**ftipo\_publicaciones**

   Field        Type               Extra

P  id\_ftp       int(10) UNSIGNED   Auto Increment

   tipo         varchar(191)

   detalle      varchar(191)

   estado       varchar(191)

   posicion     int(11)

   log          int(11)

   created\_at   timestamp          Allow Null

   updated\_at   timestamp          Allow Null

C:\Users\Mire\Desktop\DIC_htm_images\T6301_Z100_0002.gif

   Index                             Fields   Extra

   ftipo\_publicaciones\_tipo\_unique   tipo     Unique

C:\Users\Mire\Desktop\DIC_htm_images\T6301_Z100_0002.gif

**funciones**

   Field        Type               Extra

P  id\_funcion   int(10) UNSIGNED   Auto Increment

   funcion      varchar(191)

   icono        varchar(191)       Allow Null

   detalle      varchar(191)       Allow Null

   estado       varchar(191)

   created\_at   timestamp          Allow Null

   updated\_at   timestamp          Allow Null

C:\Users\Mire\Desktop\DIC_htm_images\T6301_Z100_0002.gif

   Index                      Fields    Extra

   funciones\_funcion\_unique   funcion   Unique

C:\Users\Mire\Desktop\DIC_htm_images\T6301_Z100_0002.gif

**importancias**

   Field            Type               Extra

P  id\_importancia   int(10) UNSIGNED   Auto Increment

   importancia      varchar(191)

   posicion         int(11)

   estado           varchar(191)

   log              int(11)

   created\_at       timestamp          Allow Null

   updated\_at       timestamp          Allow Null

C:\Users\Mire\Desktop\DIC_htm_images\T6301_Z100_0002.gif

   Index                             Fields        Extra

   importancias\_importancia\_unique   importancia   Unique

C:\Users\Mire\Desktop\DIC_htm_images\T6301_Z100_0002.gif

**materias**

   Field        Type               Extra

P  id\_materia   int(10) UNSIGNED   Auto Increment

   id\_carrera   int(10) UNSIGNED

   materia      varchar(191)

   nivel        varchar(191)       Allow Null

   codigo       varchar(191)       Allow Null

   sigla        varchar(191)       Allow Null

   horario      text               Allow Null

   detalle      text               Allow Null

   log          int(11)

   estado       varchar(191)

   created\_at   timestamp          Allow Null

   updated\_at   timestamp          Allow Null

C:\Users\Mire\Desktop\DIC_htm_images\T6301_Z100_0002.gif

   Index                         Fields       Extra

   materias\_id\_carrera\_foreign   id\_carrera

C:\Users\Mire\Desktop\DIC_htm_images\T6301_Z100_0002.gif

**mensajes**

   Field        Type               Extra

P  id\_mensaje   int(10) UNSIGNED   Auto Increment

   id\_origen    int(10) UNSIGNED

   asunto       varchar(191)

   mensaje      text               Allow Null

   fecha        datetime

   estado       varchar(191)

   log          int(11)

   created\_at   timestamp          Allow Null

   updated\_at   timestamp          Allow Null

C:\Users\Mire\Desktop\DIC_htm_images\T6301_Z100_0002.gif

**mensajes\_destinos**

   Field        Type               Extra

P  id\_md        int(10) UNSIGNED   Auto Increment

   id\_mensaje   int(10) UNSIGNED

   id\_destino   int(10) UNSIGNED

   visto        varchar(191)

   created\_at   timestamp          Allow Null

   updated\_at   timestamp          Allow Null

C:\Users\Mire\Desktop\DIC_htm_images\T6301_Z100_0002.gif

   Index                                  Fields       Extra

   mensajes\_destinos\_id\_mensaje\_foreign   id\_mensaje

C:\Users\Mire\Desktop\DIC_htm_images\T6301_Z100_0002.gif

**migrations**

   Field       Type               Extra

P  id          int(10) UNSIGNED   Auto Increment

   migration   varchar(191)

   batch       int(11)

C:\Users\Mire\Desktop\DIC_htm_images\T6301_Z100_0002.gif

**mpublicaciones**

   Field             Type               Extra

P  id\_mpublicacion   int(10) UNSIGNED   Auto Increment

   titulo            varchar(191)

   detalle           varchar(191)

   contenido         text

   etiqueta          varchar(191)

   publicador        int(11)

   fecha\_inicio      date

   fecha\_fin         date

   id\_importancia    int(10) UNSIGNED

   id\_materia        int(10) UNSIGNED

   id\_ftp            int(10) UNSIGNED

   estado            varchar(191)

   log               int(11)

   created\_at        timestamp          Allow Null

   updated\_at        timestamp          Allow Null

C:\Users\Mire\Desktop\DIC_htm_images\T6301_Z100_0002.gif

   Index                                   Fields           Extra

   mpublicaciones\_id\_materia\_foreign       id\_materia

   mpublicaciones\_id\_ftp\_foreign           id\_ftp

   mpublicaciones\_id\_importancia\_foreign   id\_importancia

C:\Users\Mire\Desktop\DIC_htm_images\T6301_Z100_0002.gif

**password\_resets**

   Field        Type           Extra

   email        varchar(191)

   token        varchar(191)

   created\_at   timestamp      Allow Null

C:\Users\Mire\Desktop\DIC_htm_images\T6301_Z100_0002.gif

   Index                         Fields   Extra

   password\_resets\_email\_index   email

C:\Users\Mire\Desktop\DIC_htm_images\T6301_Z100_0002.gif

**roles**

   Field        Type               Extra

P  id\_rol       int(10) UNSIGNED   Auto Increment

   rol          varchar(191)

   detalle      varchar(191)       Allow Null

   estado       varchar(191)

   created\_at   timestamp          Allow Null

   updated\_at   timestamp          Allow Null

C:\Users\Mire\Desktop\DIC_htm_images\T6301_Z100_0002.gif

   Index              Fields   Extra

   roles\_rol\_unique   rol      Unique

C:\Users\Mire\Desktop\DIC_htm_images\T6301_Z100_0002.gif

**tareas**

   Field        Type               Extra

P  id\_tarea     int(10) UNSIGNED   Auto Increment

   id\_funcion   int(10) UNSIGNED

   tarea        varchar(191)

   vista        varchar(191)

   icono        varchar(191)       Allow Null

   detalle      varchar(191)       Allow Null

   estado       varchar(191)

   created\_at   timestamp          Allow Null

   updated\_at   timestamp          Allow Null

C:\Users\Mire\Desktop\DIC_htm_images\T6301_Z100_0002.gif

   Index                       Fields       Extra

   tareas\_tarea\_unique         tarea        Unique

   tareas\_id\_funcion\_foreign   id\_funcion

C:\Users\Mire\Desktop\DIC_htm_images\T6301_Z100_0002.gif

**tareas\_roles**

   Field          Type               Extra

P  id\_tarea\_rol   int(10) UNSIGNED   Auto Increment

   id\_tarea       int(10) UNSIGNED

   id\_rol         int(10) UNSIGNED

   created\_at     timestamp          Allow Null

   updated\_at     timestamp          Allow Null

C:\Users\Mire\Desktop\DIC_htm_images\T6301_Z100_0002.gif

   Index                           Fields     Extra

   tareas\_roles\_id\_tarea\_foreign   id\_tarea

   tareas\_roles\_id\_rol\_foreign     id\_rol

C:\Users\Mire\Desktop\DIC_htm_images\T6301_Z100_0002.gif

**ucps**

   Field        Type               Extra

P  id\_ucp       int(10) UNSIGNED   Auto Increment

   id           int(10) UNSIGNED

   id\_carrera   int(10) UNSIGNED

   log          int(11)

   created\_at   timestamp          Allow Null

   updated\_at   timestamp          Allow Null

C:\Users\Mire\Desktop\DIC_htm_images\T6301_Z100_0002.gif

   Index                     Fields       Extra

   ucps\_id\_foreign           id

   ucps\_id\_carrera\_foreign   id\_carrera

C:\Users\Mire\Desktop\DIC_htm_images\T6301_Z100_0002.gif

**ucws**

   Field        Type               Extra

P  id\_ucw       int(10) UNSIGNED   Auto Increment

   id           int(10) UNSIGNED

   id\_carrera   int(10) UNSIGNED

   log          int(11)

   created\_at   timestamp          Allow Null

   updated\_at   timestamp          Allow Null

C:\Users\Mire\Desktop\DIC_htm_images\T6301_Z100_0002.gif

   Index                     Fields       Extra

   ucws\_id\_foreign           id

   ucws\_id\_carrera\_foreign   id\_carrera

C:\Users\Mire\Desktop\DIC_htm_images\T6301_Z100_0002.gif

**ufps**

   Field         Type               Extra

P  id\_ufp        int(10) UNSIGNED   Auto Increment

   id            int(10) UNSIGNED

   id\_facultad   int(10) UNSIGNED

   log           int(11)

   created\_at    timestamp          Allow Null

   updated\_at    timestamp          Allow Null

C:\Users\Mire\Desktop\DIC_htm_images\T6301_Z100_0002.gif

   Index                      Fields        Extra

   ufps\_id\_foreign            id

   ufps\_id\_facultad\_foreign   id\_facultad

C:\Users\Mire\Desktop\DIC_htm_images\T6301_Z100_0002.gif

**ufws**

   Field         Type               Extra

P  id\_ufw        int(10) UNSIGNED   Auto Increment

   id            int(10) UNSIGNED

   id\_facultad   int(10) UNSIGNED

   log           int(11)

   created\_at    timestamp          Allow Null

   updated\_at    timestamp          Allow Null

C:\Users\Mire\Desktop\DIC_htm_images\T6301_Z100_0002.gif

   Index                      Fields        Extra

   ufws\_id\_foreign            id

   ufws\_id\_facultad\_foreign   id\_facultad

C:\Users\Mire\Desktop\DIC_htm_images\T6301_Z100_0002.gif

**umps**

   Field        Type               Extra

P  id\_ump       int(10) UNSIGNED   Auto Increment

   id           int(10) UNSIGNED

   id\_materia   int(10) UNSIGNED

   log          int(11)

   created\_at   timestamp          Allow Null

   updated\_at   timestamp          Allow Null

C:\Users\Mire\Desktop\DIC_htm_images\T6301_Z100_0002.gif

   Index                     Fields       Extra

   umps\_id\_foreign           id

   umps\_id\_materia\_foreign   id\_materia

C:\Users\Mire\Desktop\DIC_htm_images\T6301_Z100_0002.gif

**umws**

   Field        Type               Extra

P  id\_umw       int(10) UNSIGNED   Auto Increment

   id           int(10) UNSIGNED

   id\_materia   int(10) UNSIGNED

   log          int(11)

   created\_at   timestamp          Allow Null

   updated\_at   timestamp          Allow Null

C:\Users\Mire\Desktop\DIC_htm_images\T6301_Z100_0002.gif

   Index                     Fields       Extra

   umws\_id\_foreign           id

   umws\_id\_materia\_foreign   id\_materia

C:\Users\Mire\Desktop\DIC_htm_images\T6301_Z100_0002.gif

**users**

   Field            Type               Extra

P  id               int(10) UNSIGNED   Auto Increment

   name             varchar(191)

   email            varchar(191)

   password         varchar(191)

   tipo             varchar(191)       Allow Null

   educacion        varchar(191)       Allow Null

   biografia        text               Allow Null

   foto             varchar(191)       Allow Null

   estado           varchar(191)       Allow Null

   log              int(11)

   remember\_token   varchar(100)       Allow Null

   created\_at       timestamp          Allow Null

   updated\_at       timestamp          Allow Null

C:\Users\Mire\Desktop\DIC_htm_images\T6301_Z100_0002.gif

   Index                Fields   Extra

   users\_email\_unique   email    Unique

C:\Users\Mire\Desktop\DIC_htm_images\T6301_Z100_0002.gif

**usuarios\_ftps**

   Field        Type               Extra

P  id\_uftp      int(10) UNSIGNED   Auto Increment

   id           int(10) UNSIGNED

   id\_ftp       int(10) UNSIGNED

   log          int(11)

   created\_at   timestamp          Allow Null

   updated\_at   timestamp          Allow Null

C:\Users\Mire\Desktop\DIC_htm_images\T6301_Z100_0002.gif

   Index                          Fields   Extra

   usuarios\_ftps\_id\_foreign       id

   usuarios\_ftps\_id\_ftp\_foreign   id\_ftp

C:\Users\Mire\Desktop\DIC_htm_images\T6301_Z100_0002.gif

**usuarios\_importancias**

   Field            Type               Extra

P  id\_ui            int(10) UNSIGNED   Auto Increment

   id               int(10) UNSIGNED

   id\_importancia   int(10) UNSIGNED

   log              int(11)

   created\_at       timestamp          Allow Null

   updated\_at       timestamp          Allow Null

C:\Users\Mire\Desktop\DIC_htm_images\T6301_Z100_0002.gif

   Index                                          Fields           Extra

   usuarios\_importancias\_id\_foreign               id

   usuarios\_importancias\_id\_importancia\_foreign   id\_importancia

C:\Users\Mire\Desktop\DIC_htm_images\T6301_Z100_0002.gif

**usuarios\_tareas**

   Field              Type               Extra

P  id\_usuario\_tarea   int(10) UNSIGNED   Auto Increment

   id\_usuario         int(10) UNSIGNED

   id\_tarea           int(10) UNSIGNED

   created\_at         timestamp          Allow Null

   updated\_at         timestamp          Allow Null

C:\Users\Mire\Desktop\DIC_htm_images\T6301_Z100_0002.gif

   Index                                Fields       Extra

   usuarios\_tareas\_id\_usuario\_foreign   id\_usuario

   usuarios\_tareas\_id\_tarea\_foreign     id\_tarea

C:\Users\Mire\Desktop\DIC_htm_images\T6301_Z100_0002.gif

**visitas**

   Field             Type               Extra

P  id\_visita         int(10) UNSIGNED   Auto Increment

   id\_fpublicacion   int(10) UNSIGNED

   visita            varchar(191)

   ciudad            text               Allow Null

   ip                varchar(45)

   fecha             datetime

   created\_at        timestamp          Allow Null

   updated\_at        timestamp          Allow Null

C:\Users\Mire\Desktop\DIC_htm_images\T6301_Z100_0002.gif

   Index                             Fields            Extra

   visitas\_id\_fpublicacion\_foreign   id\_fpublicacion

* 1. **SÍNTESIS**

En síntesis, tanto el modelado de la base de datos como el diccionario de la base de datos, nos permiten visualizar la arquitectura de la base de datos en la que está diseñado el sistema. Ya que se siguió las convenciones que Laravel sugiere para el desarrollo del sistema.

*CAPITULO 7*

1. PRUEBAS

*CAPITULO 8*

1. EVALUACION Y RESULTADOS
   1. **INTRODUCCIÓN**

Para terminar, el objetivo del proyecto era desarrollar un sistema web que difunda información e interactúe con los distintos tipos de usuario, aplicando tecnología móvil, para asegurar el fácil acceso a los usuarios y de esta forma adecuarse a nuevas tendencias tecnológicas, que se consiguió gracias a la creación de un entorno web plenamente funcional.

* 1. **EVALUACION**

En términos generales se ha conseguido desarrollar un sistema robusto que define claramente las funcionalidades solicitadas por la FPVA. La cual parte de cumplir los objetivos planteados.

* + 1. **OBJETIVOS**
* Mantener informado a la comunidad docente – estudiantil por medio de la implementación de una página web.
* Informar al plantel docente – estudiantil por carrera con el desarrollo de un portal propio.
* Generar publicaciones científicas para la comunidad docente – estudiantil por carrera.
* Fusionar a la FPVA por medio de enlaces con otros portales web.
  + 1. **ALCANCE**
* El sistema web se encuentra alojada en [www.fpva.umss.edu.bo](http://www.fpva.umss.edu.bo).
* El portal web está optimizado para poder acceder desde cualquier dispositivo móvil que cuente con servicio de Internet.
  + 1. **PRODUCTO DE EVALUACION**

Como producto de evaluación llegamos a la siguiente conclusión.

* + - 1. **CONCLUSION**

Una vez realizado el estudio, análisis, diseño y desarrollo del sistema de manejo y difusión de información se concluye lo siguiente:

1. La FPVA acarreaba una problemática a la hora de publicar información referente a las actividades de la Facultad, ya que esta no contaba con un portal oficial que difunda información fehaciente de la Facultad. Así también, otro factor que impulso el desarrollo de un sistema web fue los constantes problemas sociales que la región sufre, lo que ocasiona que el estatuto docente y el estatuto administrativo no pueda llegar a su fuente de trabajo y por lo tanto el estatuto estudiantil quedaba en total desinformación.
2. Durante la formulación y el estudio de la situación de la FPVA, en combinación con el modelado de negocio, inicialmente se pudo conocer la gestión de la organización en estudio, a través de los procesos ejecutados, los objetivos, las reglas por las que se rigen, las actividades, los proyectos e indicadores de gestión y reportes manejados, lo que permitió describir el funcionamiento de la Facultad Politécnica del Valle Alto.

Seguidamente con el diagnóstico del nivel de eficiencia con que se las publicaciones impresas llegaban a los diferentes actores de la facultad, esto permitió conocer en detalle las variables que inciden de manera negativa en el desempeño de las actividades en estudio.

1. En la fase de planificación se determinó el ámbito de desarrollo del sistema de web, además de un plan elaborado por módulos para llevar a cabo el proyecto con mayor seguridad y precisión.
2. En la etapa de análisis, se pudo establecer los requerimientos en general tanto los de usuarios, como los del sistema a través de la iteración constante establecida entre el cliente y el desarrollador, además de la estructura del contenido, las funciones de aplicación y la iteración usuario-sistema, determinado así, cuál será la función y la forma de operar del sistema.
3. El sistema al haber sido desarrollado en Laravel y Bootstrap, llegamos a comprobar que efectivamente con estos frameworks el tiempo de desarrollo de los sistemas orientados a la web toma un tiempo aproximado a tres meses.
4. Con el desarrollo de la fase de diseño, se produjo contenido a ser incorporado y la forma que es presentado en el sistema, se definió la estructura global de la aplicación, así como el diseño de la base de datos y la estructura de datos, el de navegación y la interfaz del usuario, proporcionando sencillez para el fácil manejo e interacción con los mismos.
5. En la fase de generación de páginas y pruebas, se construyeron los componentes ejecutables del sistema (Módulos y Bases de Datos) bajo el uso de herramientas basadas en software libre y posteriormente se integraron para dar funcionalidad y culminación al sistema. Las pruebas y revisiones al sistema web permitieron asegurar su funcionamiento en varios aspectos, reflejando con ello, su buen desenvolvimiento.
6. Durante la fase de evaluación del cliente, los usuarios finaes realizaron las revisiones correspondientes a los modulos de cada fase, dando a conocer sus inquietudes acerca el proyecto, además mostraron disposición a involucrarse en el mismo, obteniendo como resultado la aceptación positiva del sistema para su posterior incremento y uso.
7. Finalmente, el desarrollo del sistema aporto una serie de beneficios no solo para el visitante sino para todos los estatutos de la Facultad, prevaleciendo en la facilidad para la gestión de sus procesos en tiempo real, aumento de nivel de eficiencia y productividad, disminución de costos y esfuerzos, permitiendo con ello, el cumplimiento cabal de los objetivos y la obtención de resultados satisfactoriosreal, aumento del nivel de eficiencia y productividad, y disminución de
   1. **RESULTADOS**

* Hablar de los obejtivos
* Desarrollo de un sistema de administración y un portal web para la Facultad Politécnica del Valle Alto.
* Implementación de dos templates, uno para el sistema administración AdminLTE y Degree para el portal web.
  1. **SÍNTESIS**

*CAPITULO 8*

1. BIBLIOGRAFIA

[1]FLORES CUETO Juan José y BERTILOTTI ZUÑIGA Carmen Rosa (2014) Método de las 6’D Modelamiento – Algoritmo – Programación Tomo I. Ed. Macro.

[2]FLORES CUETO Juan José y RODRIGUEZ VILA Juan José (2014) Método de las 6’D Modelamiento – Algoritmo – Programación Tomo II. Ed. Macro.

[3]FLORES CUETO Juan José y ACUÑA FLORES Carlos Christian (2014) Método de las 6’D Modelamiento – Algoritmo – Programación Tomo III. Ed. Macro.

[4]HERNÁNDEZ SAMPIERI Roberto, FERNÁNDEZ COLLADO Carlos y BAPTISTA LUCIO Pilar (2014) Metodología de la Investigación 6° Edición. Ed. Mc Graw Hill Education.

[5]BERMÚDEZ C. Lilia Teresa y RODRÍGUEZ C. Luis Felipe (2015) Investigación de la Gestión Empresarial. Ed. Macro ECOE.

[6]CID Alma, MÉNDEZ Rosemary y SANDOVAL Franco (2015) Investigación Fundamentos y Metodología. Ed. Pearson.

[7]SOMMERVILE Ian (2005) Ingeniería del Software 7° Edición. Ed. Pearson.

[8]<https://es.wikipedia.org/wiki/Laravel> Consultado en: 12/07/2017

[9]<https://es.wikipedia.org/wiki/Framework> Consultado en: 12/07/2017

[10]<https://jquery.com/> Consultado en: 12/07/2017

[11]<https://es.wikipedia.org/wiki/Bootstrap_(framework)> Consultado en: 13/07/2017

[12]<https://es.wikipedia.org/wiki/PHP> Consultado en: 13/07/2017

[13]<https://es.wikipedia.org/wiki/JQuery> Consultado en: 13/07/2017

[14]<https://es.wikipedia.org/wiki/Servidor_HTTP_Apache> Consultado en: 13/07/2017

[15]<https://es.wikipedia.org/wiki/MySQL> Consultado en: 13/07/2017