



**UNIVERSIDAD POLITÉCNICA DE PUEBLA**

## **Ingeniería en informática**

**PROYECTO:**

Desarrollo web

**“Sistema de control escolar: módulo de inscripción de alumnos”**

**PRESENTA:**

Tania Zacariaz Martínez

**ASEESOR TÉCNICO:**

M.C Rebeca Rodríguez Huesca

**ASEESOR ACADÉMICO:**

M.C Rebeca Rodríguez Huesca

Juan C. Bonilla, Puebla

## ÍNDICE

INTRODUCCIÓN.....	4
1. Capítulo I Antecedentes de la empresa .....	6
1.1 Misión .....	6
1.2 Visión.....	6
1.3 Política.....	6
2. Capítulo II Metodología y herramientas de desarrollo.....	8
2.1 Metodología MPIU+A.....	8
2.1.1 Características del modelo MPlu+a.....	9
2.1.2 Fases del modelo.....	11
2.2 Herramientas utilizadas .....	14
2.2.1 Sistema operativo Debian 8 Jessie .....	14
2.2.2 Ruby .....	15
2.2.3 Sinatra .....	15
2.2.4 PostgreSQL .....	16
2.2.5 Sublime Text .....	16
2.2.6 SQL Power Architect.....	17
3. CAPÍTULO III. DESARROLLO .....	18
3.1 Análisis .....	18
3.1.1 Caso de uso.....	18
3.2 Diseño .....	21
3.2.1 Diagrama entidad relación .....	22

---

3.2.2 Diccionario de datos.....	23
3.3 Lanzamiento .....	44
3.3.1 Instalación de herramientas .....	44
3.4 Prototipado .....	44
3.5 Evaluación .....	57
3.5.1 Pruebas de seguridad .....	57
Conclusión.....	67
Bibliografía .....	68

## INTRODUCCIÓN

En todo desarrollo de sistemas de software es de suma importancia el seguir alguna especificación que permita a los desarrolladores el tener una disciplina que haga que todas las etapas del desarrollo del sistema, desde la pesquisa inicial de requerimientos hasta las pruebas finales del sistema, sean no sólo más coherentes si no también más formales. Antes de comenzar el desarrollo de cualquier proyecto, se realiza un estudio de sistemas para detectar todos los detalles de la situación actual en la empresa.

La información reunida con este estudio sirve como base para crear varias estrategias de diseño.

El presente trabajo se deriva de la realización de mi estadía en la Universidad Politécnica de Puebla.

La Universidad Politécnica de Puebla, es una Institución Pública que mediante un modelo de educación basado en competencias, prepara profesionales, con una sólida formación científica, técnica y social, conscientes del contexto económico, político y sociocultural del Estado y del país; además, impulsa la investigación aplicada y la innovación tecnológica, asociadas a las necesidades de las empresas y de la sociedad.

El proyecto a realizar se llama “Sistema de control escolar: módulo de inscripción de alumnos”, en el cual se realizará el modelo entidad relación de la base de datos partiendo del análisis obtenido por el asesor académico, el caso de uso se llama “Inscribir alumno”, con base en este caso de uso se diseñará una base de datos, diseño de interfaces, modelos de base de datos, programación del módulo, implementación y pruebas.

El presente documento se conforma de tres capítulos, en el primero se mostrará información básica de la institución pública: antecedentes, misión, visión, política y organigrama.

En el capítulo II se presentará la fundamentación teórica en que se basa el desarrollo del proyecto, la cual incluirá la descripción de la metodología empleada, las características de las herramientas tecnológicas utilizadas.

Y en el capítulo III contendrá la descripción de las actividades realizadas durante la estadía práctica las cuales consisten en análisis, caso de uso, diseño, modelo entidad relación, diccionario de datos, lanzamiento, instalación de las herramientas, prototipado, evaluación, prueba de compatibilidad, prueba de seguridad, también tendrá un apartado de conclusiones y bibliografía.

En el apartado de conclusión se detallará un resumen breve de los elementos obtenidos como resultado de la elaboración del proyecto, principalmente los hallazgos y experiencias obtenidas.

Finalmente en el apartado de bibliografía se pondrán las referencias de todas las fuentes de consulta por el medio.

## **1. Capítulo I Antecedentes de la empresa**

La Universidad Politécnica de Puebla se encuentra ubicada en el Tercer Carril del Ejido "Serrano" S/N. San Mateo Cuanalá. Juan C. Bonilla, Puebla, Pue. Ofrecen siete programas de ingeniería y siete de posgrado que responden de forma pertinente a las necesidades de la región, con estándares nacionales e internacionales de calidad, preparando profesionistas aptos para competir en México y cualquier parte del mundo, con una visión socialmente responsable como en la actualidad se exige, ya que nuestro entorno cambia constantemente y se requiere siempre de una mejora continua.

### **1.1 Misión**

“Formar íntegramente profesionales competentes que atiendan necesidades de los sectores productivo y social, mediante el desarrollo tecnológico, la innovación y la investigación aplicada, promoviendo una cultura ambiental y de equidad de género.”

### **1.2 Visión**

Al año 2030: “Ser una Universidad acreditada nacional e internacionalmente por su modelo educativo, sus servicios y la calidad de sus egresados(as), en un marco de sustentabilidad, de respeto al medio ambiente y fomento a la equidad de género, que consolida permanentemente alianzas estratégicas con los sectores productivo y social.”

### **1.3 Política**

La universidad tiene como política: “Cumplir y mejorar continuamente la eficacia de su Sistema de Gestión de la Calidad, a través de establecer, revisar y evaluar periódicamente los objetivos de calidad, que están orientados a fortalecer el servicio educativo que brinda la Universidad, para formar integralmente profesionales competentes que atiendan necesidades de los sectores productivo y social.”

Dicha institución está dividida en distintos departamentos, cada uno con una función  
Dicha institución está dividida en distintos departamentos, cada uno con una función

## 1.4 Organigrama

En la Figura 1 se muestra el organigrama de la Universidad Politécnica, el cual establece los diferentes departamentos que la conforman, el proyecto se centra en el de Servicios Escolares, ya que es la Reingeniería del Sistema de Control Escolar, dicho departamento está encargado de coordinar y controlar la ejecución de las actividades referentes a los servicios escolares que se proporcionan a la comunidad. A su cargo se encuentra la Licenciada Liliana Escamilla Moctezuma, quien es la encargada de monitorear el estatus de los alumnos desde que ingresan hasta que egresan de la universidad, entre otros servicios que se le ofrecen a los alumnos académicamente a lo largo de su estancia por la universidad como altas y bajas de materias, préstamo de documentos, expedición de certificados escolares y constancias académicas, promoción de becas internas y externas.

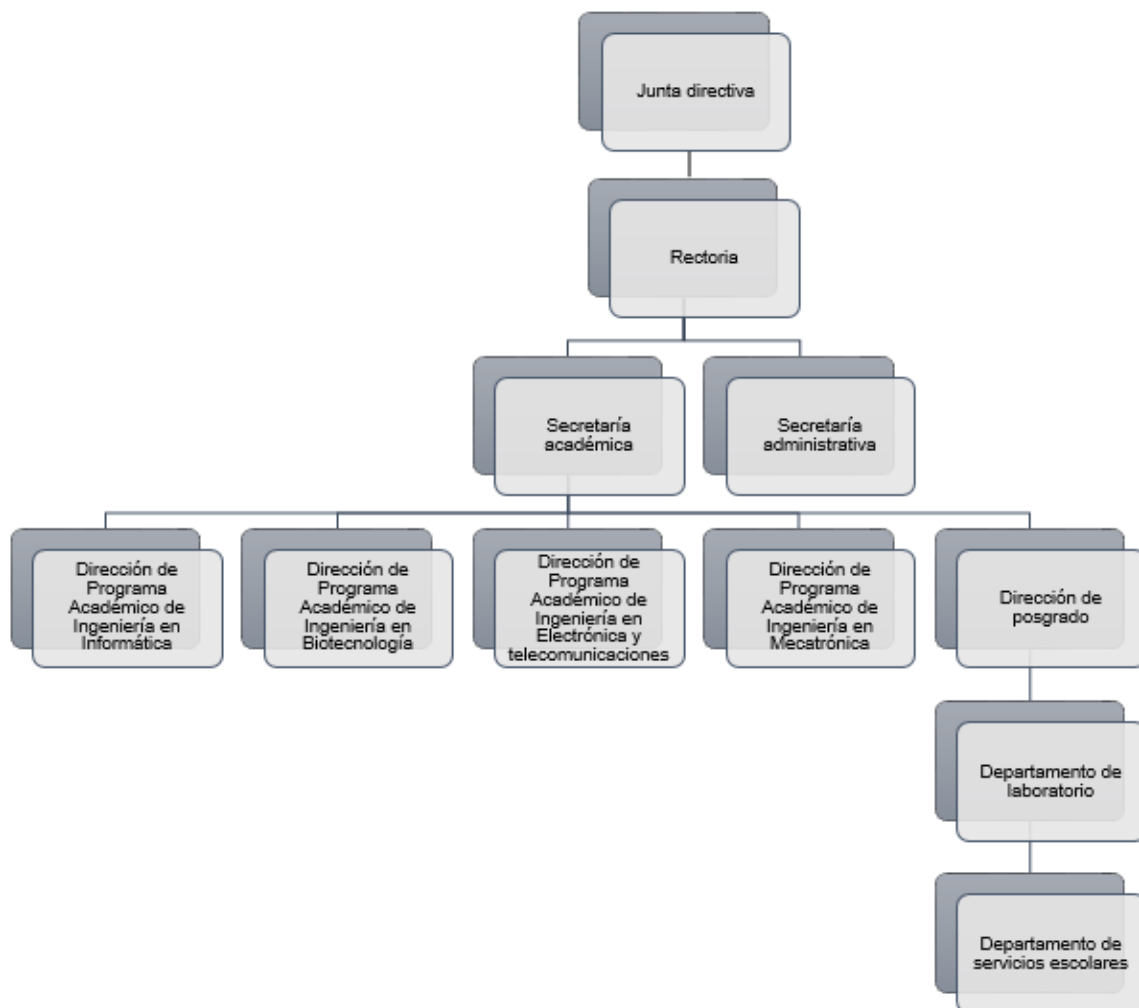


Figura 1 Organigrama de la Universidad Politecnica de Puebla

## 2. Capítulo II Metodología y herramientas de desarrollo

Este capítulo presenta la fundamentación teórica en que se basa el desarrollo del proyecto, la cual incluirá la descripción de la metodología empleada y de las herramientas tecnológicas utilizadas.

Para el desarrollo de todo el sistema se está utilizando la metodología MPIU+A, a continuación se detalla dicha metodología, cabe mencionar que este sistema se ha estado desarrollando desde años atrás por personal de la universidad.

### 2.1 Metodología MPIU+A

Es una metodología que integra la ingeniería del software, la interacción persona-ordenador y la accesibilidad en el contexto de equipos de desarrollo multidisciplinarios. En la Figura 2 se muestra el modelo de Proceso de la Ingeniería de la Usabilidad y la Accesibilidad (MPIU+a) [1].

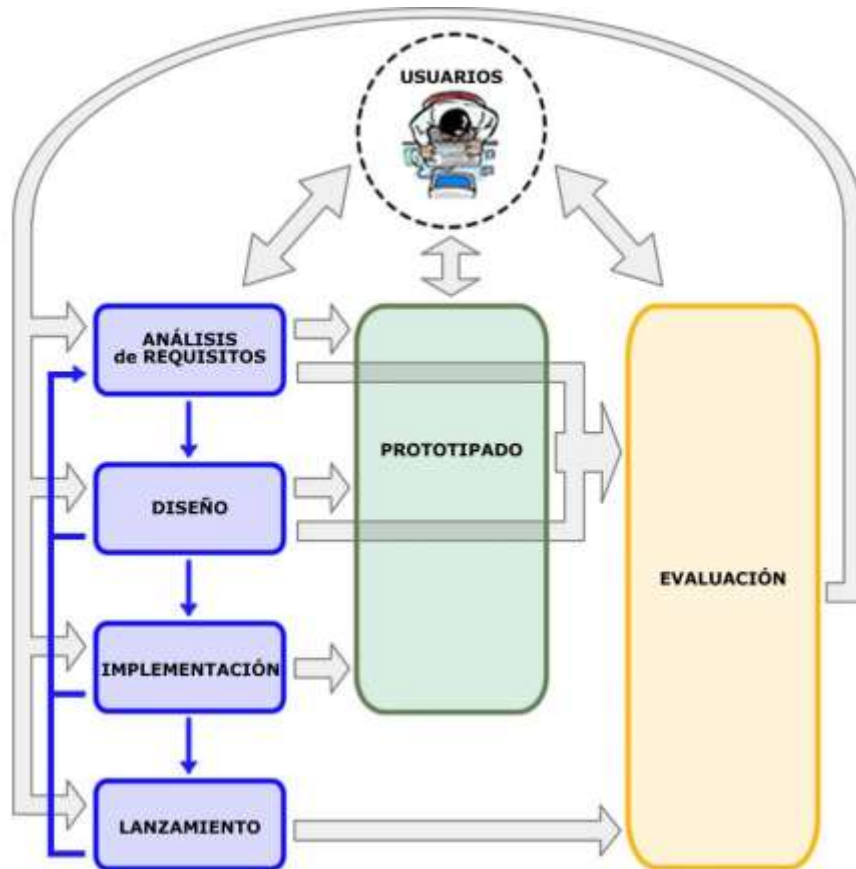


Figura 2 Esquema MPIU+A



### 2.1.1 Características del modelo MPlu+a

MPlu+a es un modelo (o metodología) de desarrollo de sistemas interactivos que sigue los principios del Diseño Centrado en el Usuario cuyas principales características son:

#### 2.1.1.1 Organización conceptual

El esquema se organiza con base a una serie de módulos que determinan la fase de desarrollo en la que nos encontramos y ubica en un nodo concreto la actividad existente en IPO.

#### 2.1.1.2 Tres pilares básicos (ingeniería del software, prototipado, evaluación)

Uno de los objetivos de este modelo era conseguir integrar el modelo de desarrollo de sistemas interactivos de la Ingeniería del Software con los principios básicos de la Ingeniería de la Usabilidad.

En la Ingeniería de la Usabilidad y en la IPO, en general hay dos conceptos muy importantes que deben realizarse de manera sistemática desde el inicio del desarrollo y no pueden cesar hasta la finalización del sistema: El prototipado y la evaluación.

En la Figura 3 se muestran los tres pilares del MPlu+A:

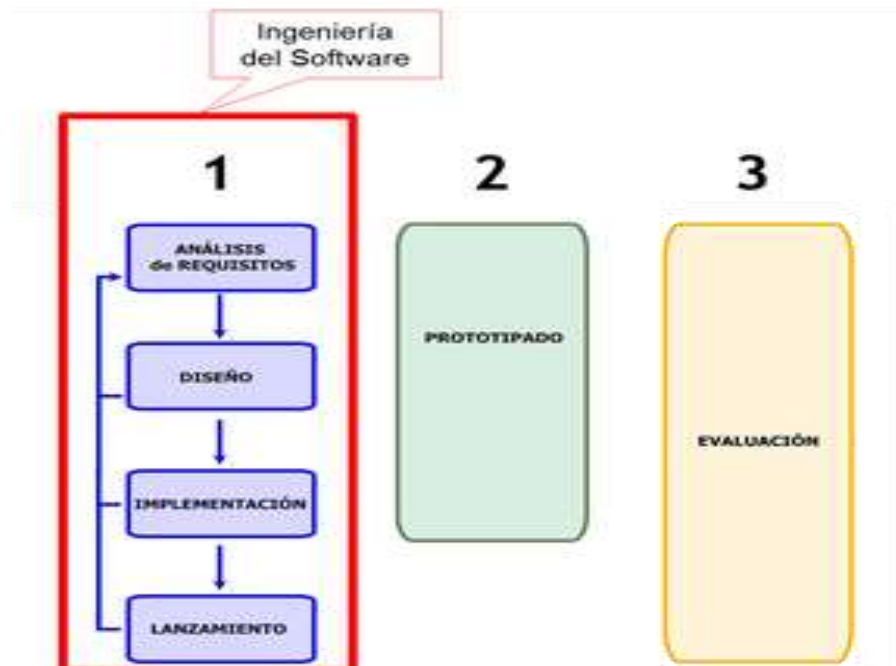


Figura 3 Tres pilares del MPlu+A

### 2.1.1 .3 El usuario

No pueden ser los desarrolladores los que califiquen un producto como usable, amigable para el usuario o de fácil aprendizaje y uso. Los únicos que pueden calificar como tal un producto son los propios usuarios. Por ese motivo, resulta de vital importancia que éstos estén implicados en todo momento y durante todo el proceso. En la Figura 4 se muestra como es la interacción de la metodología con el usuario.

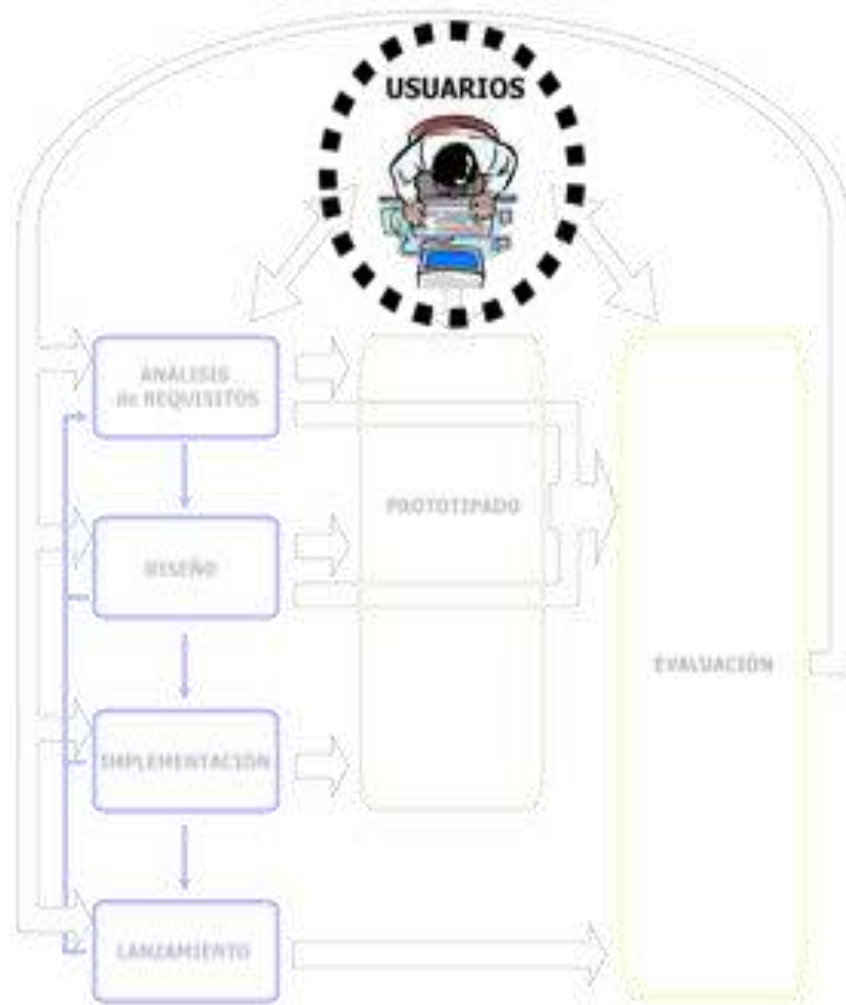


Figura 4 Interacción de usuario en MPlu+A

## 2.1.2 Fases del modelo

### 2.1.2.1 Análisis de requisitos

El punto de vista que de la fase del Análisis de Requisitos realiza la Ingeniería del Software (IS) “clásica” establece los servicios que el sistema debe proporcionar y las restricciones bajo las cuales debe operar.

Se especifican las condiciones que determinan qué debe hacer el sistema y cómo debe hacerlo. Aun así, las aproximaciones al desarrollo de software “centradas en el usuario” reconocen que es imposible especificar todos los requisitos por adelantado. Los clientes no pueden apreciar sus necesidades reales hasta que no pueden ver e interactuar con las opciones de que disponen o incluso más, no aprecian las nuevas tecnologías hasta que no las prueban, simplemente porque las desconocen.

No se sabe lo que se ha desarrollado hasta que no se ha realizado el primer “wrong system”. El modelo de proceso, MPlu+a, utiliza parte de la GC de la IS a la que le aplica un cambio decisivo para conseguir el deseado diseño centrado en el Usuario: En la IS el Diseño de las Interfaces se aborda después de haber especificado el diseño de los datos, el arquitectónico y el de los módulos, mientras que en el MP el Diseño de las Interfaces pasa a primer término y el resto viene condicionado, si procede, a la Interfaz [3].

### 2.1.2.2 Diseño

Repetidamente se llega a esta fase tras realizar actividades relacionadas con el Análisis de Requisitos que proporcionan información necesaria para que el equipo de desarrollo sea capaz de modelar el sistema para, posteriormente, proceder a su codificación.

Durante el desarrollo de un sistema, nuevo o no, una vez resueltas las funcionalidades que debe cubrir junto con el resto de características derivadas del contexto de la interacción se pasa, en esta fase, a realizar el diseño de la actividad y el diseño de la información como principales actividades que conforman el proceso global de diseño de la interacción [3].

### **2.1.2.3 Implementación**

La fase de implementación es conocida también como fase de codificación, pues supone todo el proceso de escribir el código software necesario que hará posible que el sistema finalmente implementado cumpla con las especificaciones establecidas en la fase de análisis de requisitos y responda al diseño del sistema descrito en la fase anterior.

Habitualmente esta fase es la que requiere de mayor dedicación en cuanto a recursos personales, no obstante, este factor se ve minimizado si se sigue el proceso aquí descrito, pues el impacto del cambio se ve minimizado por el buen trabajo previamente realizado.

Cuando se llega a esta fase del modelo de proceso ya se han determinado el o los lenguajes de programación a utilizar para la implementación del proyecto, las bases de datos correspondientes que se precisen, los sistemas de intercomunicación de procesos, y en general toda la tecnología subyacente [3].

### **2.1.2.4 Lanzamiento**

La fase de lanzamiento de todo proyecto, sea o no interactivo, suele ser una de las más críticas de cualquier proceso o desarrollo. Es el momento en que se ven concretadas en mayor o menor grado las expectativas puestas en el producto.

El éxito del producto dependerá de dos factores muy importantes, que el usuario se sienta cómodo con el sistema.

Entendiendo como sentirse cómodo que no le dé errores, que no le resulte complicado usarlo, que recuerde fácilmente dónde están las diferentes opciones y sus funcionalidades, etc. Por otro lado que los responsables obtengan los resultados esperados.

La aplicación del MPlu+a asegura que ambos aspectos se vean satisfechos, pues:

- El diseño se ha hecho en base y para los usuarios.

Haciéndoles partícipes, además, se consigue un efecto doble, por un lado, como en parte se sienten responsables del diseño del sistema no encontrarán motivos para criticarlo duramente, y por otro, como todo ha sido evaluado por ellos mismos su utilización y aprendizaje no les comportará carga cognitiva excesiva.

- Como todo producto software, desarrollado por los métodos clásicos, la evaluación funcional es lo primero que se prima y no se da por bueno si no se cumplen sus especificaciones [3].

#### **2.1.2.5 Prototipado**

Es impensable llegar al final del desarrollo sin haber realizado comprobaciones a lo largo del camino. Los prototipos son precisamente el mecanismo que permite realizar estas comprobaciones.

El MPlu+a no marca ninguna pauta para indicar a los diseñadores en qué situaciones deberán recurrir al uso de una determinada o determinadas técnicas para simular el funcionamiento. Tampoco los limita a poder realizar un primer prototipo en una fase muy inicial del proyecto.

El modelo intenta garantizar que se cumplan los pasos necesarios para disponer de un producto altamente usable y accesible a la vez que concede un alto grado de libertad para que el equipo de desarrollo libremente decida cuándo y cómo deberá aplicar las diferentes técnicas [3].

#### **2.1.2.6 Evaluación**

Evaluar consiste en probar algo. Tanto para saber si funciona correctamente como no, si cumple con las expectativas o no, o simplemente para conocer cómo funciona una determinada herramienta o utensilio.

En nuestro modelo MPlu+a la fase de evaluación del constituye un punto clave para la obtención de sistemas interactivos usables y accesibles. Será en esta fase donde se aplicarán las técnicas necesarias para recibir la realimentación necesaria por parte de los usuarios y/o evaluadores expertos que se verá reflejado en el diseño de las interfaces de los usuarios mejorando sus procesos interactivos.

La evaluación no debe ser pensada solamente como una simple etapa del proceso general del diseño, y mucho menos de la implementación, del sistema, sino que ésta debe realizarse durante todo el ciclo de vida del proceso de desarrollo, los resultados de la cual deben aportar mejoras respecto a las soluciones evaluadas y correcciones respecto a errores reportados [3].

## 2.2 Herramientas utilizadas

El desarrollo de este proyecto se realizó en el sistema operativo Debian 8 Jessie utilizando Ruby como lenguaje principal de programación con el framework Sinatra, PostgreSQL como gestor de base de datos y DataMapper como adaptador para establecer la comunicación con la base de datos, también se utilizó el editor de código sublime text y SQL Power Architect para el modelado de datos.

A continuación se describe cada una de las herramientas utilizadas para la realización del proyecto:

### 2.2.1 Sistema operativo Debian 8 Jessie

Un sistema operativo es un conjunto de programas y utilidades básicas que hacen que la computadora funcione. El centro de un sistema operativo es el núcleo (N. del T.: kernel). El núcleo es el programa más importante en la computadora, realiza todo el trabajo básico y le permite ejecutar otros programas.

Los sistemas Debian actualmente usan el núcleo de Linux o de FreeBSD. Linux es una pieza de software creada en un principio por Linus Torvalds y desarrollada por miles de programadores a lo largo del mundo. FreeBSD es un sistema operativo que incluye un núcleo y otro software.

Las novedades que incorpora Jessie, tras casi 3 años de desarrollo no son pocas, entre las que destacan el cambio del sistema de inicio de sysvinit a SystemD, un instalador renovado acorde para la ocasión, ahora en modo gráfico por defecto y mejoras en el arranque UEFI.

#### **Ventajas;**

Este sistema operativo es adecuado ya que ofrece estabilidad para servidores que trabajan durante más de un año seguido sin reiniciarse, rápido y ligero en memoria y tiene buena seguridad del sistema.

#### **Desventajas:**

Se necesita un previo conocimiento en Linux para poder usar con comodidad el Sistema Operativo ,

### 2.2.2 Ruby

Ruby es un lenguaje de programación dinámico y de código abierto enfocado en la simplicidad y productividad. Su elegante sintaxis se siente natural al leerla y fácil al escribirla [2].

Ruby es un lenguaje de programación interpretado, reflexivo y orientado a objetos, creado por el programador japonés Yukihiro "Matz" Matsumoto, quien comenzó a trabajar en Ruby en 1993, y lo presentó públicamente en 1995. Combina una sintaxis inspirada en Python y Perl con características de programación orientada a objetos similares a Smalltalk. Comparte también funcionalidad con otros lenguajes de programación como Lisp, Lua, Dylan y CLU. Ruby es un lenguaje de programación interpretado en una sola pasada y su implementación oficial es distribuida bajo una licencia de software libre [3].

#### **Ventajas:**

- Es un lenguaje sencillo y fácil de leer.
- Soportado por la mayoría de las plataformas web.
- Se trata de un software libre.
- Integra comandos de manejo de bases de datos.

### 2.2.3 Sinatra

Sinatra es un lenguaje específico de dominio (DSL) para crear aplicaciones web en Ruby de manera sencilla. Sinatra es en sí, es una librería.

Con Sinatra se puede estructurar la aplicación según el problema al que se le quiere dar una solución. Básicamente, se posee bastante control al trabajar con Sinatra.

Sinatra es un framework para aplicaciones web de software libre y código abierto, y lenguaje específico del dominio escrito en Ruby. Es una alternativa a otros frameworks para aplicaciones web como Ruby on Rails, Merb, Nitro, Camping, y Rango. Sinatra depende de Rack interfaz de servidor web [3].

Diseñado y desarrollado por Blake Mizerany, Sinatra es pequeño y flexible. Sinatra no sigue el típico patrón modelo vista controlador que se ve en otros frameworks, como Ruby on Rails. En su lugar, Sinatra se enfoca en la "rápida creación de aplicaciones web en Ruby con el mínimo esfuerzo."

### 2.2.4 PostgreSQL

El lenguaje estructurado de consultas (SQL) es un lenguaje de base de datos normalizado, utilizado por la gran mayoría de los servidores de bases de datos que manejan bases de datos relacionales u objeto-relacionales.

Es un lenguaje declarativo en el que las órdenes especifican cual debe ser el resultado y no la manera de conseguirlo (como ocurre en los lenguajes procedimentales).

Al ser declarativo es muy sistemático, sencillo y con una curva de aprendizaje muy agradable ya que sus palabras clave permiten escribir las órdenes como si fueran frases en las que se especifica (en inglés) que es lo que queremos obtener [4].

#### **Ventajas:**

- Fácil de Administrar.
- Su sintaxis SQL es estándar y fácil de aprender.
- Footprint bajo de memoria, bastante poderoso con una configuración adecuada.
- Multiplataforma.

#### **Desventajas:**

- Es fácil de vulnerar sin protección adecuada.
- InnoDB genera mucho footprint en memoria al indizar.
- Realizar revisiones llegar a ser una labor manual y tediosa para el DBA.
- Reducida cantidad de tipos de datos.

### 2.2.5 Sublime Text

Sublime Text es un editor de código multiplataforma, es una herramienta concebida para programar sin distracciones. Su interfaz de color oscuro y la riqueza de coloreado de la sintaxis, centra nuestra atención completamente.

Sublime Text permite tener varios documentos abiertos mediante pestañas, e incluso emplear varios paneles para aquellos que utilicen más de un monitor. Dispone de modo de pantalla completa, para aprovechar al máximo el espacio visual disponible de la pantalla.



Es un editor de texto y editor de código fuente está escrito en C++ y Python para los plugins. Se puede descargar y evaluar de forma gratuita. Sin embargo, no es software libre o de código abierto, ya que se debe obtener una licencia para su uso continuado, aunque la versión de evaluación es plenamente funcional y no tiene fecha de caducidad.

### 2.2.6 SQL Power Architect

SQL Power Architect es una herramienta de modelado de datos creados por los diseñadores de almacenamiento de datos y tiene muchas características únicas, dirigidas específicamente para el arquitecto de almacenamiento de datos.

SQL Power Architect es una herramienta ideal para grupos de desarrollo donde se puede realizar el modelado de datos y poder así tener documentado el modelo de datos de todas las aplicaciones que se desarrollan.

#### Ventajas:

- Se conecta a múltiples fuentes de datos al mismo tiempo.
- Compara datos de modelos y estructuras de bases de datos e identifica las discrepancias.
- Permite arrastrar y soltar las tablas de origen y las columnas del playpen.
- Guarda el histórico de origen de cada columna.
- Genera fuente en blanco para informes de mapeo visual.
- Ingeniería directa e inversa para PostgreSQL, Oracle, MS SQL Server y mucho más.
- Guarda la estructura de fuente de datos instantáneamente en el diseño, para que pueda trabajar de forma remota.
- Todos los datos del proyecto se almacenan en un formato XML de fácil análisis.
- Modelado OLAP de esquema: cubos, medidas, dimensiones, jerarquías y niveles.

### 3. CAPÍTULO III. DESARROLLO

Este capítulo se mostraran las actividades realizadas durante la estadía: creación de caso de uso a partir del detallado de este, diseño, desarrollo del módulo de inscripción de alumnos.

#### 3.1 Análisis

El proceso de reunión de requisitos se intensifica y se centra especialmente en el software. Dentro del proceso de análisis es fundamental que a través de una colección de requerimientos funcionales y no funcionales, el desarrollador o desarrolladores del software comprendan completamente la naturaleza del programa que deben construir para el desarrollo de la aplicación, la función requerida, comportamiento, rendimiento e interconexión [5].

En esta etapa se realizó el caso de uso inscribir alumno de acuerdo al detallado de este que ya había sido realizado por el asesor académico.

##### 3.1.1 Caso de uso

Un Modelo de Casos de Uso es un modelo de las funciones deseadas de la aplicación y sus ambientes, y sirve como un contrato entre el cliente y los desarrolladores. Los casos de uso sirven como un hilo unificador a lo largo de desarrollo de la aplicación. El Modelo de Casos de Uso es el resultado del flujo de Requisitos y es usado como parte importante del Análisis, Diseño y de la Prueba.

Existen muchas maneras de modelar una aplicación, cada una puede servir para un propósito diferente. Sin embargo, el propósito más importante de un modelo de casos de uso es comunicar el comportamiento de la aplicación al cliente o usuario final. Por consiguiente, el modelo debe ser fácil de entender.

En la Figura 5 se muestra el caso de uso “Inscribir Alumno” y posteriormente el detallado del mismo:

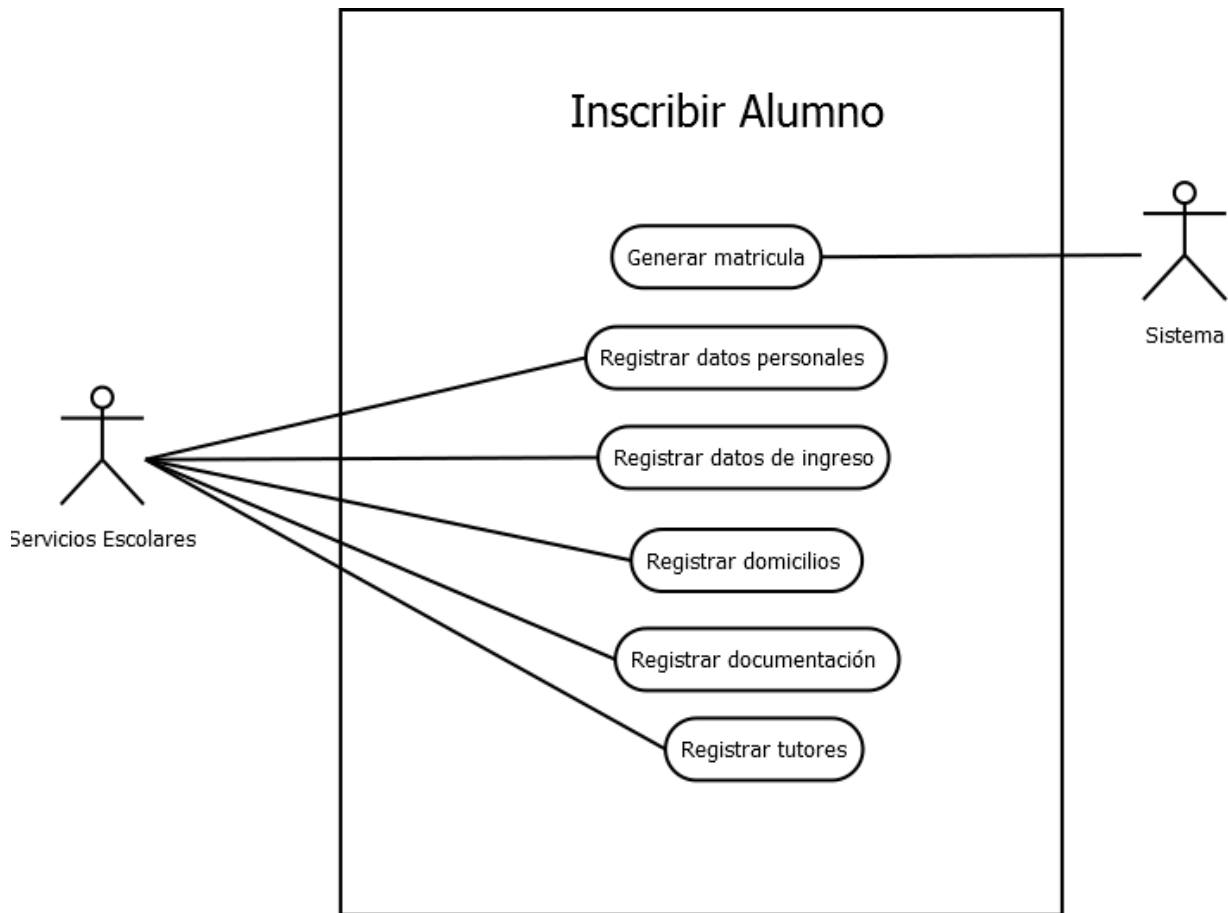


Figura 5 Caso de uso Inscribir Alumno

<i>Inscribir alumno</i>	
Breve descripción:	El caso de uso <i>inscribir Alumno</i> permite al actor inscribir al alumno de nuevo ingreso.
Actores	Servicios Escolares
Descripción paso por paso:	<ol style="list-style-type: none"> <li>1. El actor selecciona el programa educativo y el periodo de ingreso al que se inscribe el alumno.</li> <li>2. El sistema asigna la matrícula del alumno que se inscribe, lo cual se realiza automáticamente, de acuerdo con el programa educativo y el periodo de inscripción de las siguiente manera:  <div style="text-align: center;">AA CC PP 999 donde</div> A= Corresponde al año en que ingresa a la universidad</li> </ol>

	<p>C= Número que identifica a la carrera en la que se inscribe</p> <p>PP = tomará el valor 01, 02 o 03 de acuerdo con el periodo en que se inscriba el estudiante, de manera que 01 = Enero, 02 = Mayo y 03 = Septiembre</p> <p>999 = será un número consecutivo de acuerdo al orden de inscripción.</p> <p>3. El sistema solicitará la información que se detalla a continuación dependiendo del programa educativo:</p> <p>3.1 Si el programa educativo es ingeniería :</p> <p>El actor teclea los siguientes datos del alumno que se indicarán como obligatorios:</p> <p>3.1.1 Datos personales</p> <p>3.1.1.1 El sistema verifica que el actor haya tecleado el puntaje ceneval, en caso contrario no se podrá inscribir al alumno.</p> <p>3.1.2 Datos de ingreso</p> <p>3.1.3 Documentación general, y dependiendo de la nacionalidad se mostrarán los documentos adicionales</p> <p>3.1.4 Datos de domicilio de procedencia y de residencia.</p> <p>3.1.5 Datos del tutor o de los tutores.</p> <p>3.2 Si el programa educativo es Maestría:</p> <p>El actor teclea los siguientes datos del alumno que se indicarán como obligatorios:</p> <p>3.2.1 Datos personales</p> <p>3.2.1.1 El sistema verifica que el actor haya acreditado el examen/propedéutico, en caso contrario no se podrá inscribir al alumno.</p>
--	--

	<p>3.2.2 Documentación general, y dependiendo de la nacionalidad se mostrarán los documentos adicionales</p> <p>3.2.3 Datos de domicilio de procedencia y de residencia.</p> <p>3.2.4 Datos del tutor o de los tutores.</p> <p>3.3 Si e programa educativo es Especialidad:</p> <p>3.3.1 Datos personales</p> <p>3.3.1.1 El sistema verifica que el actor haya acreditado el examen/propedéutico, en caso contrario no se podrá inscribir al alumno</p> <p>3.3.2 Documentación general, y dependiendo de la nacionalidad se mostrarán los documentos adicionales</p> <p>3.3.3 Datos de domicilio de procedencia y de residencia.</p> <p>3.3.4 Datos del tutor o de los tutores.</p> <p>4. Además se verificará que el usuario haya tecleado los datos marcados como obligatorios para poder registrar al alumno.</p>
--	--

### 3.2 Diseño

De acuerdo al análisis se agregaron 15 tablas al modelo ya existente, éstas son necesarias para poder inscribir al alumno, a la tabla existente estudiantes se le tuvo que añadir más campos ya que sólo contaba con los básicos, esto para cumplir con el análisis de inscripción de alumnos.

En la Figura 6 se muestra el diagrama entidad relación para inscribir alumno:

### 3.2.1 Diagrama entidad relación

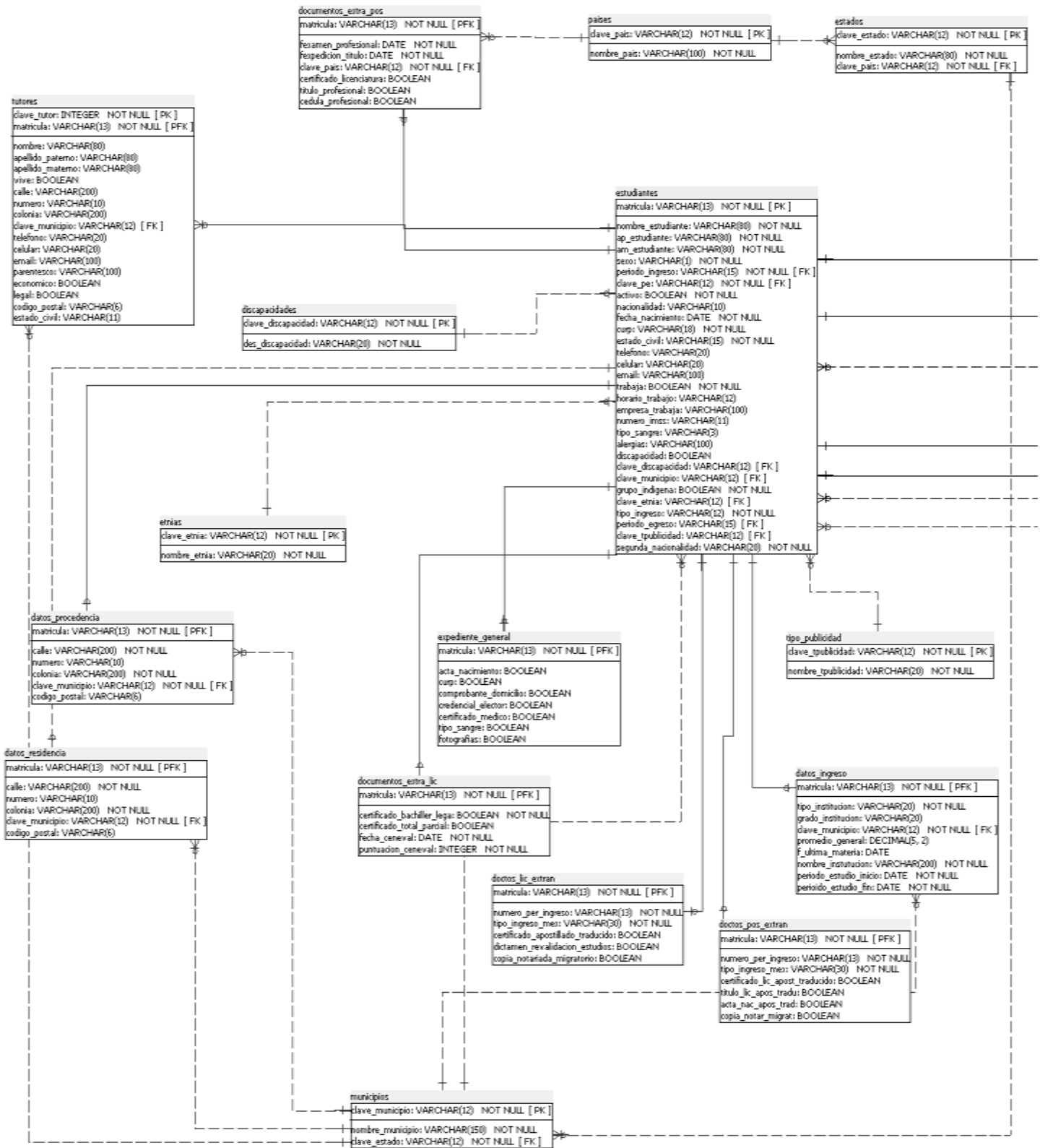


Figura 6 Diagrama entidad relación

### 3.2.2 Diccionario de datos

El diccionario de datos es un listado organizado de todos los datos que pertenecen a un sistema.

El objetivo de un diccionario de datos es dar precisión sobre los datos que se manejan en un sistema, evitando así malas interpretaciones o ambigüedades.

Define con precisión los datos de entrada, salida, componentes de almacenes, flujos, detalles de las relaciones, etc.

A continuación en la Figura 7 se muestra el diccionario de datos del diagrama entidad relación de la Figura 6.

#### Lista de tablas

- |                         |                        |
|-------------------------|------------------------|
| 1. datos_ingreso        | 9. estados             |
| 2. datos_procedencia    | 10. etnias             |
| 3. datos_residencia     | 11. estudiantes        |
| 4. discapacidades       | 12. expediente_general |
| 5. doctos_lic_extran    | 13. municipios         |
| 6. doctos_pos_extran    | 14. paises             |
| 7. documentos_extra_lic | 15. tipo_publicidad    |
| 8. documentos_extra_pos | 16. tutores            |

## datos\_ingreso

(Nombre físico: datos\_ingreso)

Datos de ingreso.

Nombre lógico de columna	Nombre físico de columna	Tipo	PK	Nullable
matricula (PK) ( <a href="#">FK</a> )	matricula	VARCHAR(13)	PK	NOT NULL
Indica la clave con la que se identifica a los estudiantes				
tipo_institucion	tipo_institucion	VARCHAR(20)		NOT NULL
Tipo de institución de procedencia, puede ser bachillerato o preparatoria.				
grado_institucion	grado_institucion	VARCHAR(20)		
Indicará los tipos de grado: Si elige bachillerato los valores son: Abierto, General, Tecnológico y si elige preparatoria los valores serán Incorporado, Universitario.				
clave_municipio ( <a href="#">FK</a> )	clave_municipio	VARCHAR(12)		NOT NULL
Indica el municipio de la institución de procedencia.				
promedio_general	promedio_general	DECIMAL(5,2)		
Indica el promedio obtenido en la institución de procedencia.				
f_ultima_materia	f_ultima_materia	DATE		
Indica la fecha en que curso su última materia en la institución de procedencia.				
nombre_instutucion	nombre_instutucion	VARCHAR(200)		NOT NULL
Indica el nombre de la institución de procedencia del estudiante.				
periodo_estudio_inicio	periodo_estudio_inicio	DATE		NOT NULL
Indica la fecha en que el alumno empezó a estudiar en dicha institución.				
perioido_estudio_fin	perioido_estudio_fin	DATE		NOT NULL
Indica la fecha en que el alumno finalizó de estudiar en dicha institución.				

### Referencias

- [estudiantes](#) through (matricula)
- [municipios](#) through (clave\_municipio)
-



### datos\_procedencia

(Nombre físico: datos\_procedencia)

Datos del domicilio de procedencia.

Nombre lógico de columna	Nombre físico de columna	Tipo	PK	Nullable
matricula (PK) ( <a href="#">FK</a> )	matricula	VARCHAR(13)	PK	NOT NULL
Indica la clave con la que se identifica a los estudiantes				
calle	calle	VARCHAR(200)		NOT NULL
Indica el nombre de la calle de la casa de procedencia del estudiante.				
numero	numero	VARCHAR(10)		
Indica el número de casa de procedencia del estudiante.				
colonia	colonia	VARCHAR(200)		NOT NULL
Indica el nombre de la colonia de procedencia del estudiante.				
clave_municipio ( <a href="#">FK</a> )	clave_municipio	VARCHAR(12)		NOT NULL
Indica el nombre del municipio de procedencia del estudiante.				
codigo_postal	codigo_postal	VARCHAR(6)		
Indica el C.P. de procedencia del estudiante.				

#### Referencias

- [estudiantes](#) through (matricula)
- [municipios](#) through (clave\_municipio)

### datos\_residencia

(Nombre físico: datos\_residencia)

Datos del domicilio de residencia.

Nombre lógico de columna	Nombre físico de columna	Tipo	PK	Nullable
matricula (PK) ( <a href="#">FK</a> )	matricula	VARCHAR(13)	PK	NOT NULL
Indica la clave con la que se identifica a los estudiantes				
calle	calle	VARCHAR(200)		NOT NULL

Indica el nombre de la calle de la casa donde reside del estudiante.				
numero	numero	VARCHAR(10)		
Indica el número de casa donde reside el estudiante.				
colonia	colonia	VARCHAR(200)		NOT NULL
Indica el nombre de la colonia donde reside el estudiante.				
clave_municipio (FK)	clave_municipio	VARCHAR(12)		NOT NULL
Indica el nombre del municipio donde reside el estudiante.				
codigo_postal	codigo_postal	VARCHAR(6)		
Indica el C.P. donde reside el estudiante.				

#### Referencias

- [estudiantes](#) through (matricula)
- [municipios](#) through (clave\_municipio)

#### discapacidades

(Nombre físico: discapacidades)

Catálogo de discapacidades.

Nombre lógico de columna	Nombre físico de columna	Tipo	PK	Nullable
clave_discapacidad (PK)	clave_discapacidad	INTEGER	PK	NOT NULL
Es la clave asignada a una discapacidad.				
des_discapacidad	des_discapacidad	VARCHAR(20)		NOT NULL
Indica el nombre asignado a cada una de las discapacidades.				

Referenciada por

- [estudiantes](#) referencing (clave\_discapacidad)

## doctos\_lic\_extran

(Nombre físico: doctos\_lic\_extran)

Documentos adicionales para alumnos de licenciatura extranjeros o mexicanos con estudios en el extranjero.

Nombre lógico de columna	Nombre físico de columna	Tipo	PK	Nullable
matricula (PK) ( <a href="#">FK</a> )	matricula	VARCHAR(13)	PK	NOT NULL
Indica la clave con la que se identifica a los estudiantes				
numero_per_ingreso	numero_per_ingreso	VARCHAR(13)		
Número otorgado por el Instituto Nacional de Migración.				
tipo_ingreso_mex	tipo_ingreso_mex	VARCHAR(30)		
Tipo de ingreso a México.				
certificado_apostillado_traducido	certificado_apostillado_traducido	BOOLEAN		
Original del certificado total de estudios apostillado y traducido a la UPPue. Sólo para extranjeros. • True = Si • False = No				
dictamen_revalidacion_estudios	dictamen_revalidacion_estudios	BOOLEAN		
Original del dictamen nacional de revalidación de estudios a la UPPue. Sólo para extranjeros • True = Si • False = No				
copia_notariada_migratorio	copia_notariada_migratorio	BOOLEAN		
Copia notariada del documento migratorio a la UPPue. Sólo para extranjeros • True = Si • False = No				

Referencias

[estudiantes](#) through (matricula)

## doctos\_pos\_extran

(Nombre físico: doctos\_pos\_extran)

Documentos adicionales para alumnos de posgrado extranjeros o mexicanos con estudios en el extranjero.

### Referencias

- [estudiantes](#) through (matricula)

## documentos\_extra\_pos

(Nombre físico: documentos\_extra\_pos)

Docuementos extra para posgrado.

Nombre lógico de columna	Nombre físico de columna	Tipo	PK	Nullable
matricula (PK) ( <a href="#">FK</a> )	matricula	VARCHAR(13)	PK	NOT NULL
Indica la clave con la que se identifica a los estudiantes				
fexamen_profesional	fexamen_profesional	DATE		NOT NULL
Fecha en que el estudiante realizó su examen profesional de licenciatura. (aaaa-mm-dd).				
fexpedicion_titulo	fexpedicion_titulo	DATE		NOT NULL
Fecha en que se le expidió el título profesional al estudiante de licenciatura. (aaaa-mm-dd).				
clave_pais ( <a href="#">FK</a> )	clave_pais	VARCHAR(12)		NOT NULL
País de origen del título según catálogo de la base de datos.				
certificado_licenciatura	certificado_licenciatura	BOOLEAN		
Original y copia del certificado de licenciatura a la UPPue • True = Si • False = No				
titulo_profesional	titulo_profesional	BOOLEAN		
Copia del título profesional a la UPPue • True = Si • False = No				

cedula_profesional	cedula_profesional	BOOLEAN		
Copia de la cédula profesional a la UPPue				
• True = Si • False = No				

#### Referencias

- [estudiantes](#) through (matricula)
- [paises](#) through (clave\_pais)

#### estados

(Nombre físico: estados)

Catálogo de estados.

Nombre lógico de columna	Nombre físico de columna	Tipo	PK	Nullable
clave_estado (PK)	clave_estado	VARCHAR(12)	PK	NOT NULL
Es la clave asignada a un estado.				
nombre_estado	nombre_estado	VARCHAR(80)		NOT NULL
Indica el nombre del estado				
clave_pais (FK)	clave_pais	VARCHAR(12)		NOT NULL
Es la clave asignada a un país.				

#### Referencias

- [paises](#) through (clave\_pais)

#### Referenciada por

- [municipios](#) referencing (clave\_estado)

## estudiantes

(Nombre físico: estudiantes)

Nombre lógico de columna	Nombre físico de columna	Tipo	PK	Nullable
matricula (PK)	matricula	VARCHAR(13)	PK	NOT NULL
Indica la clave con la que se identifica a los estudiantes				
nombre_estudiante	nombre_estudiante	VARCHAR(80)		NOT NULL
Indica el o los nombre del estudiante.				
ap_estudiante	ap_estudiante	VARCHAR(80)		NOT NULL
Indica el apellido paterno del estudiante.				
am_estudiante	am_estudiante	VARCHAR(80)		NOT NULL
Indica el apellido materno del estudiante.				
sexo	sexo	VARCHAR(1)		NOT NULL
Indica el sexo del estudiante, sólo puede ser F o M				
periodo_ingreso (FK)	periodo_ingreso	VARCHAR(15)		NOT NULL
Indica la clave del periodo en que ingresó a la universidad.				
clave_pe (FK)	clave_pe	VARCHAR(12)		NOT NULL
Clave del programa educativo al que se encuentra adscrito.				
activo	activo	BOOLEAN		NOT NULL
Indica si el estudiante se encuentra activo o no				
nacionalidad	nacionalidad	VARCHAR(10)		NOT NULL
Indica la nacionalidad del estudiante: Mexicana, extranjera o doble				
fecha_nacimiento	fecha_nacimiento	DATE		NOT NULL
Fecha de nacimiento del estudiante. Formato aaaa-mm-dd				
curp	curp	VARCHAR(18)		NOT NULL
Indica el CURP del estudiante.				
estado_civil	estado_civil	VARCHAR(15)		NOT NULL

Indica el estado civil del estudiante, sólo puede ser: Soltero,casado,divorciado,viudo o unión libre			
telefono	telefono	VARCHAR(20)	
Indica el número teléfono de casa del estudiante.			
celular	celular	VARCHAR(20)	
Indica el número celular del estudiante.			
email	email	VARCHAR(100)	
Indica el correo electrónico del estudiante.			
trabaja	trabaja	BOOLEAN	NOT NULL
Indica si trabaja actualmente el estudiante • True = Si • False = No			
horario_trabajo	horario_trabajo	VARCHAR(12)	
Indica el horario de trabajo del estudiante.			
empresa_trabaja	empresa_trabaja	VARCHAR(100)	
Indica el nombre de la institución donde trabaja.			
numero_imss	numero_imss	VARCHAR(11)	
Indica el número de afiliación al IMSS del estudiante.			
tipo_sangre	tipo_sangre	VARCHAR(3)	
Indica el grupo sanguíneo al que pertenece el estudiante			
alergias	alergias	VARCHAR(100)	
Indica las alergias(s) que padece el estudiante.			
discapacidad	discapacidad	BOOLEAN	
Padece algún tipo de discapacidad el estudiante. • True = Si • False = No			
clave_discapacidad (FK)	clave_discapacidad	INTEGER	NOT NULL
Es la clave asignada a una discapacidad.			
clave_municipio (FK)	clave_municipio	VARCHAR(12)	NOT NULL
Es la clave asignada a un municipio.			
grupo_indigena	grupo_indigena	BOOLEAN	NOT NULL

Indica si pertenece el estudiante a algún grupo indígena. • True = Si • False = No				
clave_etnia (FK)	clave_etnia	INTEGER		NOT NULL
Es la clave asignada a cada tipo de etnia.				
tipo_ingreso	tipo_ingreso	VARCHAR(12)		NOT NULL
Indica el tipo de ingreso: Equivalencia o regular				
periodo_egreso (FK)	periodo_egreso	VARCHAR(15)		
Se guardará en el momento que el estudiante cubra el total de créditos de la carrera que esté cursando.				
clave_tpublicidad (FK)	clave_tpublicidad	INTEGER		NOT NULL
Es la clave asignada a un tipo de publicidad.				
segunda_nacionalidad	segunda_nacionalidad	VARCHAR(20)		
Indica la segunda nacionalidad del estudiante en caso de tenerla.				
otra_publicidad	otra_publicidad	VARCHAR(20)		
Cuando se elige otra publicidad en este campo se guardará.				

#### Referencias

- [programas\\_educativos](#) through (clave\_pe)
- [periodos](#) through (periodo\_ingreso)
- [periodos](#) through (periodo\_egreso)
- [municipios](#) through (clave\_municipio)
- [tipo\\_publicidad](#) through (clave\_tpublicidad)
- [etnias](#) through (clave\_etnia)
- [discapacidades](#) through (clave\_discapacidad)

#### Referenciada por

- [participantes\\_est](#) referencing (matricula)
- [movilidad\\_e](#) referencing (matricula)
- [becas\\_asignadas](#) referencing (matricula)



- [colaboradoresca\\_est](#) referencing (matricula)
- [datos\\_residencia](#) referencing (matricula)
- [tutores](#) referencing (matricula)
- [documentos\\_extra\\_pos](#) referencing (matricula)
- [datos\\_procedencia](#) referencing (matricula)
- [expediente\\_general](#) referencing (matricula)
- [documentos\\_extra\\_lic](#) referencing (matricula)
- [doctos\\_lic\\_extran](#) referencing (matricula)
- [doctos\\_pos\\_extran](#) referencing (matricula)
- [datos\\_ingreso](#) referencing (matricula)

## etnias

### (Nombre físico: etnias)

Catálogo de etnias.

Datos étnicos.

Nombre lógico de columna	Nombre físico de columna	Tipo	PK	Nullable
clave_etnia (PK)	clave_etnia	INTEGER	PK	NOT NULL
Es la clave asignada a cada tipo de etnia.				
nombre_etnia	nombre_etnia	VARCHAR(20)		NOT NULL
Indica el nombre de la etnia.				

Referenciada por

- [estudiantes](#) referencing (clave\_etnia)

## expediente\_general

(Nombre físico: expediente\_general)

Documentos generales entregados por el estudiante.

Nombre lógico de columna	Nombre físico de columna	Tipo	PK	Nullable
matricula (PK) (FK)	matricula	VARCHAR(13)	PK	NOT NULL
Indica la clave con la que se identifica a los estudiantes				
acta_nacimiento	acta_nacimiento	BOOLEAN		
Indica si el estudiante entregó original y copia del acta de nacimiento a la UPPue: • True = Si • False = No				
curp	curp	BOOLEAN		
Indica si el estudiante entregó copia del CURP a la UPPue • True = Si • False = No				
comprobante_domicilio	comprobante_domicilio	BOOLEAN		
Indica si el estudiante entregó copia del comprobante de domicilio a la UPPue : • True = Si • False = No				
credencial_elector	credencial_elector	BOOLEAN		
Indica si el estudiante entregó copia de la identificación oficial a la UPPue : • True = Si • False = No				
certificado_medico	certificado_medico	BOOLEAN		
Indica si el estudiante entregó original y copia del certificado médico a la UPPue : • True = Si • False = No				
tipo_sangre	tipo_sangre	BOOLEAN		
Indica si el estudiante entregó original y copia del documento que indique el tipo de sangre: : • True = Si • False = No				
fotografias	fotografias	BOOLEAN		
Indica si el estudiante entregó 2 fotografías tamaño infantil, blanco y negro : • True = Si • False = No				

## Referencias

- [estudiantes](#) through (matricula)

## municipios

(Nombre físico: municipios)

Catálogo de municipios.

Nombre lógico de columna	Nombre físico de columna	Tipo	PK	Nullable
clave_municipio (PK)	clave_municipio	VARCHAR(12)	PK	NOT NULL
Es la clave asignada a un municipio.				
nombre_municipio	nombre_municipio	VARCHAR(150)		NOT NULL
Indica el nombre de un municipio.				
clave_estado (FK)	clave_estado	VARCHAR(12)		NOT NULL
Es la clave asignada a un estado.				

## Referencias

- [estados](#) through (clave\_estado)

Referenciada por

- [datos\\_ingreso](#) referencing (clave\_municipio)
- [datos\\_procedencia](#) referencing (clave\_municipio)
- [datos\\_residencia](#) referencing (clave\_municipio)
- [tutores](#) referencing (clave\_municipio)
- [estudiantes](#) referencing (clave\_municipio)

## países

(Nombre físico: países)

Catálogo de países.

Nombre lógico de columna	Nombre físico de columna	Tipo	PK	Nullable
clave_pais (PK)	clave_pais	VARCHAR(12)	PK	NOT NULL
Es la clave asignada a un país.				
nombre_pais	nombre_pais	VARCHAR(100)		NOT NULL
Indica el nombre de país.				

Referenciada por

- [documentos extra pos](#) referencin (clave\_pais)
- [estados](#) referencin (clave\_pais)

## tutores

(Nombre físico: tutores)

Datos de los tutores del alumno

Nombre lógico de columna	Nombre físico de columna	Tipo	PK	Nullable
matricula (PK) ( <a href="#">FK</a> )	matricula	VARCHAR(13)	PK	NOT NULL
Indica la clave con la que se identifica a los estudiantes				
nombre	nombre	VARCHAR(80)		
Indica el o los nombres del tutor				
apellido_paterno	apellido_paterno	VARCHAR(80)		
Indica el apellido paterno del tutor.				
apellido_materno	apellido_materno	VARCHAR(80)		
Indica el apellido materno del tutor.				
vive	vive	BOOLEAN		

Indica si la persona vive: : • True = Si • False = No			
calle	calle	VARCHAR(200)	
Calle de la casa donde reside el tutor.			
numero	numero	VARCHAR(10)	
Número de casa donde reside el tutor.			
colonia	colonia	VARCHAR(200)	
Colonia donde reside el tutor.			
clave_municipio (FK)	clave_municipio	VARCHAR(12)	
Municipio donde reside el tutor.			
telefono	telefono	VARCHAR(20)	
Teléfono de casa o trabajo.			
celular	celular	VARCHAR(20)	
Celular del tutor.			
email	email	VARCHAR(100)	
Correo electrónico del tutor.			
parentesco	parentesco	VARCHAR(100)	
Indica si los datos pertenecen a: Madre, padre u otro			
economico	economico	BOOLEAN	
Indica si esta persona es el tutor económico: : • True = Si • False = No			
legal	legal	BOOLEAN	
Indica si esta persona es el tutor legal: : • True = Si • False = No			
codigo_postal	codigo_postal	VARCHAR(6)	
C.P. donde reside el tutor.			
estado_civil	estado_civil	VARCHAR(11)	
Indica el estado civil del tutor.			
clave_tutor	clave_tutor	INTEGER	NOT NULL

Es la clave asignada a un tutor	
---------------------------------	--

## Referencias

- [estudiantes](#) through (matricula)
- [municipios](#) through (clave\_municipio)

Una vez que se creó la base de datos en postgres, el cual se llamó dbsie lo siguiente es crear los modelos de datos en el controlador a este se le llamo sie.rb en el cual se especifican el tipo de dato, tamaño, datos propios y sus relaciones.

A continuación se lista el orden en el que se crearon los modelos de datos ver Figura 7:

- |                         |                        |
|-------------------------|------------------------|
| 1. Discapacidad         | 10. Datos_procedencia  |
| 2. Etnia                | 11. Datos_residencia   |
| 3. Tipo_publicidad      | 12. Expediente_general |
| 4. Pais                 | 13. Doctos_lic_extran  |
| 5. Estado               | 14. Doctos_pos_extran  |
| 6. Municipio            | 15. Estudiante         |
| 7. Dato_ingreso         | 16. Tutor              |
| 8. Documentos_extra_pos |                        |
| 9. Documento_extra_lic  |                        |

```

class Discapacidad
  include DataMapper::Resource
  storage_names[:default] = 'discapacidades'
  property :clave_discapacidad,String,:length=>12,:required=>true,:unique=>true,:key=>true
  property :des_discapacidad,String,:length=>20,:required=>true
  has 1, :estudiante,'Estudiante',:child_key=>[:clave_discapacidad]
end
class Etnia
  include DataMapper::Resource
  storage_names[:default] = 'etnias'
  property :clave_etnia,String,:length=>12,:required=>true,:unique=>true,:key=>true
  property :nombre_etnia,String,:length=>20,:required=>true
  has 1, :estudiante,'Estudiante',:child_key=>[:clave_etnia]
end
class Tipo_publicidad
  include DataMapper::Resource
  storage_names[:default] = 'tipo_publicidad'
  property :clave_tpublicidad,Integer,:required=>true,:unique=>true,:key=>true
  property :nombre_tpublicidad,String,:length=>20,:required=>true
  has n, :estudiante,'Estudiante',:child_key=>[:clave_tpublicidad]
end

class Pais
  include DataMapper::Resource
  storage_names[:default] = 'paises'
  property :clave_pais,String,:length=>12,:required=>true,:unique=>true,:key=>true
  property :nombre_pais, String,:length=>100,:required=>true
  has n, :documentos_extra_pos,'Documentos_extra_pos',:child_key=> [:clave_pais]
  has n, :estados,'Estado', :child_key=>[:clave_pais]
end
end

```

```

class Estado
  include DataMapper::Resource
  storage_names[:default]='estados'
  property :clave_estado,String, :length=>12, :required=>true,:unique=>true, :key=>true
  property :nombre_estado, String, :length=>80, :required=>true
  property :clave_pais, String, :length=>12, :required=>true
  has n, :municipio,'Municipio', :child_key=>[:clave_estado]
  belongs_to :pais,'Pais', :child_key=>[:clave_pais]
end
class Municipio
  include DataMapper::Resource
  storage_names[:default]='municipios'
  property :clave_municipio,String, :length=>12, :required=>true, :unique=>true, :key=>true
  property :nombre_municipio,String, :length=>150, :required=>true
  property :clave_estado,String,:length=>12, :required=>true
  has n,:estudiante,'Estudiante', :child_key=>[:clave_municipio]
  has n, :datos_procedencia,'Dato_procedencia', :child_key=>[:clave_municipio]
  has n, :datos_residencia, 'Dato_residencia', :child_key=>[:clave_municipio]
  has n, :tutores,'Tutor', :child_key=>[:clave_municipio]
  has n, :datos_ingreso,'Dato_ingreso', :child_key=>[:clave_municipio]
  belongs_to :estado,'Estado', :child_key=>[:clave_estado]
end
class Dato_ingreso
  include DataMapper::Resource
  storage_names[:default]='datos_ingreso'
  property :matricula,String, :length=>13, :required=>true, :key=>true
  property :tipo_institucion, String, :length=>20, :required=>true
  property :grado_institucion, String, :length=>20
  property :clave_municipio,String, :length=>12, :required=>true
  property :promedio_general, Decimal, :precision => 5, :scale => 2
  property :f_ultima_materia, Date
  property :nombre_institucion,String, :length=>200,:required=>true
  property :periodo_estudio_inicio, Date, :required=>true
  property :periodo_estudio_fin, Date, :required=>true
  belongs_to :estudiante,'Estudiante', :child_key=>[:matricula], :key=>true
  belongs_to :municipio,'Municipio', :child_key=>[:clave_municipio]
end
end

```



```

class Documentos_extra_pos
  include DataMapper::Resource
  storage_names[:default]='documentos_extra_pos'
  property :matricula,String, :length=>13, :required=>true, :key=>true
  property :fexamen_profesional, Date, :required=>true
  property :fexpedicion_titulo, Date, :required=>true
  property :clave_pais,String, :length=>12, :required=>true
  property :certificado_licenciatura, Boolean
  property :titulo_profesional, Boolean
  property :cedula_profesional, Boolean
  belongs_to :estudiante,'Estudiante', :child_key=>[:matricula], :key=>true
  belongs_to :pais,'Pais', :child_key=>[:clave_pais]
end

class Datos_procedencia
  include DataMapper::Resource
  storage_names[:default]='datos_procedencia'
  property :matricula, String, :length=>13, :required=>true, :key=>true
  property :calle, String, :length=>200, :required=>true
  property :numero, String, :length=>10
  property :colonia, String, :length=>200, :required=>true
  property :clave_municipio,String, :length=>12, :required=>true
  property :codigo_postal, String, :length=>6
  belongs_to :estudiante,'Estudiante', :child_key=>[:matricula], :key=>true
  belongs_to :municipio,'Municipio', :child_key=>[:clave_municipio]
end

class Datos_residencia
  include DataMapper::Resource
  storage_names[:default]='datos_residencia'
  property :matricula, String, :length=>13, :required=>true, :key=>true
  property :calle, String, :length=>200, :required=>true
  property :numero, String, :length=>10
  property :colonia, String, :length=>200, :required=>true
  property :clave_municipio,String, :length=>12, :required=>true
  property :codigo_postal, String, :length=>6
  belongs_to :estudiante,'Estudiante', :child_key=>[:matricula], :key=>true
  belongs_to :municipio,'Municipio', :child_key=>[:clave_municipio]
end

```

```

class Expediente_general
  include DataMapper::Resource
  storage_names[:default]='expediente_general'
  property :matricula, String, :length=>13, :required=>true, :key=>true
  property :acta_nacimiento, Boolean
  property :curp, Boolean
  property :comprobante_domicilio, Boolean
  property :credencial_elector, Boolean
  property :certificado_medico, Boolean
  property :tipo_sangre, Boolean
  property :fotografias, Boolean
  belongs_to :estudiante,'Estudiante', :child_key=>[:matricula], :key=>true
end

class Documento_extra_lic
  include DataMapper::Resource
  storage_names[:default]='documentos_extra_lic'
  property :matricula, String, :length=>13, :required=>true, :key=>true
  property :certificado_bachiller_lega, Boolean, :required=>true
  property :certificado_total_parcial, Boolean
  property :fecha_ceneval, Date, :required=>true
  property :puntuacion_ceneval, Integer, :min => 0, :max => 99999, :required=>true
  belongs_to :estudiante,'Estudiante', :child_key=>[:matricula], :key=>true
end

class Docto_lic_extran
  include DataMapper::Resource
  storage_names[:default]='doctos_lic_extran'
  property :matricula, String, :length=>13, :required=>true, :key=>true
  property :numero_per_ingreso, String, :length=>13, :required=>false
  property :tipo_ingreso_mex, String, :length=>30, :required=>false
  property :certificado_apostillado_traducido, Boolean
  property :dictamen_revalidacion_estudios, Boolean
  property :copia_notariada_migratorio, Boolean
  belongs_to :estudiante,'Estudiante', :child_key=>[:matricula], :key=>true
end

```



```

class Estudiante
  include DataMapper::Resource
  property :matricula, String, :length => 13, :required => true, :unique => true, :key => true
  property :nombre_estudiante, String, :length => 80, :required => true
  property :ap_estudiante, String, :length => 80, :required => true
  property :am_estudiante, String, :length => 80, :required => true
  property :sexo, String, :length => 1, :required => true
  property :periodo_ingreso, String, :length => 15, :required => true
  property :clave_pe, String, :length => 12, :required => true
  property :activo, Boolean, :required => true
  property :nacionalidad, String, :length => 10, :required => false
  property :fecha_nacimiento, Date, :required => true
  property :curp, String, :length => 18, :required => true
  property :estado_civil, String, :length => 15, :required => true
  property :telefono, String, :length => 20, :required => false
  property :celular, String, :length => 20, :required => false
  property :email, String, :length => 100, :required => false
  property :trabaja, Boolean, :required => true
  property :horario_trabajo, String, :length => 12, :required => false
  property :empresa_trabaja, String, :length => 100, :required => false
  property :numero_imss, String, :length => 11, :required => false
  property :tipo_sangre, String, :length => 3, :required => false
  property :alergias, String, :length => 100, :required => false
  property :discapacidad, Boolean, :required => false
  property :clave_discapacidad, String, :length => 12, :required => false
  property :clave_municipio, String, :length => 12, :required => true
  property :grupo_indigena, Boolean, :required => true
  property :clave_etnia, String, :length => 12
  property :tipo_ingreso, String, :length => 15, :required => true
  property :periodo_egreso, String, :length => 15, :required => false
  property :clave_publicidad, String, :length => 12, :required => true
  property :segunda_nacionalidad, String, :length => 20, :required => false
  property :otra_publicidad, String, :length => 20, :required => false
  has 1, :documentos_extra_pos, 'Documentos_extra_pos', :child_key => [:matricula], :constraint => :destroy
  has n, :tutores, 'Tutor', :child_key => [:matricula], :constraint => :destroy
  has 1, :datos_residencia, 'Dato_residencia', :child_key => [:matricula], :constraint => :destroy
  has 1, :datos_procedencia, 'Dato_procedencia', :child_key => [:matricula], :constraint => :destroy
  has 1, :documentos_extra_lic, 'Documento_extra_lic', :child_key => [:matricula], :constraint => :destroy
  has 1, :expediente_general, 'Expediente_general', :child_key => [:matricula], :constraint => :destroy
  has 1, :doctos_lic_extran, 'Docto_lic_extran', :child_key => [:matricula], :constraint => :destroy
  has 1, :doctos_pos_extran, 'Docto_pos_extran', :child_key => [:matricula], :constraint => :destroy
  has 1, :datos_ingreso, 'Dato_ingreso', :child_key => [:matricula], :constraint => :destroy

```

### 3.3 Implementación

La implementación se representa al inicio del desarrollo del proyecto, o de la codificación de los procesos e interfaces. Se fue proporcionado una parte del proyecto ya existente para usarlo como base y de esta forma crear interfaces y procesos similares de acuerdo al módulo a desarrollar.

La Figura 8 muestra parte de la codificación para la inscripción del alumno:

```

get '/regAlumnoDP/:clave_pe/:matricula/:clave_periodo' do
  |opcion_menu = 1
  @acceso = false
  @activa = false
  if session[:user_id].strip == 'SYSDBA'
    @acceso = true
    @activa = true
  else
    if pantalla_activa?(366)
      @activa = true
      if permiso?(366, "inserta")
        @acceso = true
      end
    end
  end
end
end
if @activa
  if @acceso
    @matricula = params[:matricula]
    @peri = params[:clave_periodo]
    @pe = Programa_Educativo.get(params[:clave_pe])
    @periodo = Periodo.get(params[:clave_periodo])
    @periodos = Periodo.all :order => :fecha_inicio, :activo => true
    @países = Pais.all :order => :nombre_pais
    @discapacidades = Discapacidad.all :order => :clave_discapacidad
    @etnias = Etnia.all :order => :clave_etnia

    @estados = Estado.all :order => :nombre_estado
    @municipios = Municipio.all :order => :nombre_municipio
    @tipos_publicidad = Tipo_publicidad.all :order => :clave_tpseudonimo

    @nuevo = Estudiante.count :clave_pe => params[:clave_pe]
    # @nuevo = Estudiante.count > 0 ? Estudiante.max
    @nue = @nuevo + 1
    @cp = params[:clave_pe]

    erb :reg_alumnoDP
  else
    flash[:error] = "Ud. No tiene acceso a esa pantalla"
    redirect '/'
  end
end
end

```

```

if !campo_vacio(params[:matricula]) and !campo_vacio(params[:nombre]) and !campo_vacio(params[:ap]) and !campo_vacio(params[:an]) and
#SE DECLARAN LOS CAMPOS QUE PUEDEN SER NULOS PARA ESTUDIANTE @tel= (params[:telefono]=='' ? nil : params[:telefono].strip.upcase)
@cel= (params[:celular]=='' ? nil : params[:celular].strip.upcase)
@email= (params[:email]=='' ? nil : params[:email].strip.upcase)
@traba = params[:conocido] == "SI" ? true : false
@grupIn=params[:indigena] == "SI" ? true : false
@cla_in=(params[:grupo_indigena] == '' ? nil : params[:grupo_indigena].strip.upcase)
@hrTra=(params[:horario]=='' ? nil : params[:horario].strip.upcase)
@empresa=(params[:empresa]=='' ? nil : params[:empresa].strip.upcase)
@num_inss=(params[:inss]=='' ? nil : params[:inss].strip.upcase)
@tipo_san=(params[:sangre]=='' ? nil : params[:sangre].strip.upcase)
@alergia=(params[:alergias]=='' ? nil : params[:alergias].strip.upcase)
@disca=params[:discapacidad] == "SI" ? true : false
@t_disca=(params[:tipo_disca]=='' ? nil : params[:tipo_disca].strip.upcase)
| @fnac = Date.strptime(params[:f_nacimiento], '%d/%m/%Y')
@fcene = Date.strptime(params[:examen_ceneval], '%d/%m/%Y')
@seg_nacionalidad=(params[:seg_nacionalidad]=='' ? nil : params[:seg_nacionalidad].strip.upcase)

@alumno = Estudiante.new(:matricula => params[:matricula], :nombre_estudiante => params[:nombre].strip.upcase, :ap_estudiante =>
if @alumno.save
  @doc_extra_lic=Documento_extra_lic.new(:matricula=>params[:matricula], :certificado_bachiller_lega => false, :fecha_ceneval:
  if @doc_extra_lic.save
    @guardado = false
  end
end
if params[:btnSig]=="Siguiente"
  flash[:notice] = "Registro guardado correctamente"
  redirect "/regAlumnoDI/#{@alumno.matricula}+ '/' + params[:clave_pe]
else
  flash[:notice] = "Registro guardado correctamente"
  redirect "/insAlumnos"
end
end
else
  flash[:error] = @alumno.errors.to_a
  redirect "/regAlumnoDP/" + params[:clave_pe] + '/' + params[:matricula] + '/' + params[:clave_periodo]
end
end
else
  flash[:error] = "Dejó vacío un campo requerido, por favor verifique!"
end
end

```

Figura 8 Código de inscripción de alumno

Se crearon varias vistas para inscribir al alumno, a continuación se muestra parte del código. En la Figura 9 se muestra parte de la codificación de las vistas creadas y la función combos dependientes de país, estado y municipio:

```

<script type="text/javascript">
// OBTENER ESTADO Y MUNICIPIO DEPENDIENDO DE PAIS
$(document).ready(function(){
//Funcion para estado
$("#clave_pais").change(function(){
var va= document.getElementById("clave_pais").value;

$("#estados_select").load("/gt_estado/"+$("#clave_pais option:selected").val(),function(responseTxt,statusTxt,xhr){
if(statusTxt=="error")
alert("Error al cargar datos: "+xhr.status+": "+xhr.statusText);
});
});

//Funcion para municipio
$("#estados_select").change(function(){
var va= document.getElementById("estados_select").value;
var va= document.getElementById("estados_select").value;
$("#municipio_select").load("/gt_municipio/"+$("#estados_select option:selected").val(),function(responseTxt,statusTxt,xhr){
if(statusTxt=="success") alert("External content loaded successfully!");
if(statusTxt=="error")
alert("Error municipio: "+xhr.status+": "+xhr.statusText);
});
});
});

```

```

<thead>
<legend>Datos generales</legend>
<table>
<tr>
<td><label>Nombre(s)</label>
<input type="text" name="nombre" id="nombre" class="campo_entrada requerido" required title="Se necesita el nombre del alumno" onkeypress="return permite(event, 'car');" />
</td>
<td><label>Apellido paterno</label>
<input type="text" name="ap" id="ap" class="campo_entrada requerido" maxlength=80 size=20 onkeypress="return permite(event, 'car');" required />
</td>
<td><label>Apellido materno</label>
<input type="text" name="am" id="am" class="campo_entrada requerido" maxlength=80 size=20 onkeypress="return permite(event, 'car');" required />
</td>
<td><label>Fecha de nacimiento:</label>
<input type="text" name="f_nacimiento" id="f_nacimiento" class="datepicker requerido" required />
</td>
</tr>
<tr>
<td><label>CURP:</label><input type="text" name="curp" id="curp" class="campo_entrada requerido" maxlength=18 onkeypress="return permite(event, 'num_car');" autocomplete="on" required />
</td>
<td><label>Sexo:</label><input type="radio" name="sexo" id="sexo" value="M" class="campo_entrada" style="box-shadow: 1px 2px 20px yellow;" required />M
<input type="radio" name="sexo" id="sexo" value="F" class="campo_entrada" style="box-shadow: 1px 2px 20px yellow;" required />F
</td>
<td><label>Nacionalidad:</label><select name="naciona" id="naciona" style="width:200px;" class="campo_entrada requerido" maxlength=10 autocomplete="on" onchange="habilitaSegNaci();" required >
<option value="MEXI CANA">MEXI CANA</option>
<option value="EXTRANJERA">EXTRANJERA</option>
<option value="DOBLE">DOBLE</option>
</select>
</td>
</tr>

```

Figura 9 Código de vista de alumno y función

### 3.3 Lanzamiento

La etapa de lanzamiento abarca el proceso de instalación de las herramientas usadas para poder desarrollar el sistema así como para que pueda ser visualizado en las máquinas del cliente para las revisiones del prototipo y su evaluación.

#### 3.3.1 Instalación de herramientas

Para llevar a cabo el desarrollo fue necesario la instalación de diferente software. A continuación se listan los pasos que se siguieron para la instalación:

- 1 Instalación del sistema operativo, en este caso Debian 8.7.1
- 2 Se instaló la versión 2.1.4 de Ruby.
- 3 Se instaló RubyGems 2.6.10. De la página <http://rubygems.org/pages/download> se descargó el archivo .tgz.
- 4 Instalación de Sinatra 1.4.7, el DSL para desarrollo web utilizando el lenguaje de programación Ruby.
- 5 Se instaló el servidor web Thin 1.5.1, pero primero se debe instalar el paquete ruby2.1.4-dev.
- 6 Finalmente se instaló PostgreSQL, se asignó una contraseña al usuario postgres. En este caso se asignó el usuario es SYSDBA y la contraseña db\_erp17.
- 7 Se creó la base de datos "dbsie" y se instaló Datamapper, para el modelado de clases, esto para interactuar con la base de datos creada.

### 3.4 Prototipado

En esta etapa de la metodología se mostraran las interfaces creadas para inscribir a un alumno:

La Figura 10 muestra el acceso al sistema, una vez que haya obtenido el acceso al sistema, se mostrará el menú de inscripción de alumno ver Figura 11.



Figura 10 Pantalla de acceso al sistema



Los campos que se muestran en amarillo indican que son obligatorios para realizar el registro en la base de datos, además están validados es decir cuando el usuario no llene el campo y quiera guardar se mostrará en rojo el contorno e indicara que debe llenar el campo solicitado ver Figura 12:



FigFigura 12 Campos requeridos /

Todas las pantallas llevarán el recuadro de inicio el cual contiene el programa educativo, matrícula y nombre completo ver Figura 13 ya que en este se tiene el nombre del programa educativo que se eligió, el periodo, la matrícula y el nombre del alumno, esto con la finalidad de saber estos datos en todas las pantallas.



Figura 13 Inicio



Cuando se seleccione en el menú la opción de inscripción por examen se mostrará la pantalla ver Figura 14, en la cual se debe elegir el programa educativo y el periodo para que se genere la matricula, la Figura 14 aplica para las tres areas de conocimiento con las que cuenta la Universidad Politécnica de Puebla (ingeniería, maestría y especialidad).



Programa educativo:	Periodo ingreso:	Matricula:
INGENIERIA EN INFORMATICA	ENE-ABR 2017	171101007

*Figura 14 Selección*

Cabe mencionar que los datos solicitados para la inscripción del alumnos son diferentes en cada área de conocimiento, por lo cual se tienen diferentes pantallas. Por lo anteriormente mencionado en las siguientes figuras se mostrará todas las figuras que son diferentes en cada área de conocimiento y se pondrán una sola vez las figuras que son iguales para las tres áreas de conocimiento.

En la Figura 15 se solicitan los datos generales, tipo de ingreso( por puntaje de CENEVAL),datos médicos, datos étnicos y datos laborales del alumno de ingeniería.

**Inscripción por examen de INGENIERIA Y TECNOLOGIA**

**Nació:**  
 Programa Educativo: INGENIERIA EN SISTEMAS AUTOMOTRICES      Período: ENE-ABR 2017      Matrícula: 171501003

**Datos generales:**

Nombre(s):       Apellido paterno:       Apellido materno:       Fecha de nacimiento:

CURP:       Sexo:  M  F      Nacionalidad:       Seg. Nacionalidad:

País:       Edo. de nacimiento:       Mpo. de nacimiento:       Estado civil:

Teléfono:       Celular:       Email:

**Tipo de ingreso:**

Cómo se enteró de la universidad:       Otra publicidad:       Permiso CENEVAL:       Fecha CENEVAL:

Tipo de ingreso:

**Datos médicos:**

Número de IMSS:       Tipo de sangre:       Alergias:       Discapacidad:  SI  NO

**Datos étnicos:**

Pertenece a un grupo indígena:  SI  NO      Grupo indígena:

**Datos laborales:**

Trabaja:  SI  NO      **Datos del trabajo:**

Horario de trabajo:

¿Dónde trabaja?:

Figura 15 Inscripción por examen de ingeniería

En la Figura 16 se solicitan los datos generales, datos de ingreso, datos médicos, datos étnicos y datos laborales del alumno de maestría o especialidad. (Cuando sea un alumno de especialidad en esta Figura sólo cambiará la leyenda a Inscripción por examen de ESPECIALIDAD).

Para los alumnos de maestría y especialidad los datos de ingreso son: por examen de admisión o propedeútico, en ambos casos debe ser acreditado, en caso contrario el sistema no dejará inscribir al alumno ver Figura 17.

**Inscripción por examen de MAESTRÍA**

**Datos Generales**

Programa Educativo: MAESTRÍA EN ENSEÑANZAS DE LAS CIENCIAS      Periodo: ENE-ABR 2017      Matrícula: 174001001

**Datos generales**

Nombre(s): MIGUEL ANGEL      Apellido paterno: VAZQUEZ      Apellido materno: PEREZ      Fecha de nacimiento: \*

CURP:      Sexo:  M  F      Nacionalidad:      Seg. Nacionalidad: \*

País:      Edo. de nacimiento:      Mpio. de nacimiento:      Estado civil: \*

Teléfono:      Celular:      E-mail: \*

**Tipo de ingreso**

Cómo se ingresó de la universidad:      Otra publicidad:      Tipo ingreso:      Acreditó:  SI  NO

**Datos médicos**

Número de IMSS:      Tipo de sangre:      Alergias:      Discapacidad:  SI  NO

**Tipo de ingreso**

Cómo se ingresó de la universidad:      Otra publicidad:      Tipo ingreso:      Acreditó:  SI  NO

**Datos médicos**

Número de IMSS:      Tipo de sangre:      Alergias:      Discapacidad:  SI  NO

Tipo discapacidad: \*

**Datos sociales**

Pertenece a un grupo indígena:  SI  NO      Grupo indígena: \*

**Datos laborales**

Trabaja:  SI  NO

**Datos del trabajo**

Horario de trabajo: \*

¿Dónde trabaja?: \*

Figura 16 Inscripción por examen de maestría/especialidad

**Error: No se puede inscribir al alumno porque no aprobó el EXAMEN**

**Inscripción por examen**

**Datos**

Programa educativo: MAESTRÍA EN INGENIERÍA EN SISTEMAS Y CÓMPUTO INTELIGENTE      Periodo ingreso: ENE-ABR 2017      Matrícula: \*



Error: NO se puede inscribir al alumno(a) porque no aprobó el PROPEDEÚTICO

**Inscripción por examen**

---

**Inscripción**

Programa educativo: MAESTRÍA EN INGENIERÍA EN SISTEMAS Y CÓMPUTO INTELIGENTE      Período ingreso: ENE-ABR 2017      Matrícula:

*Figura 17 No aprobó tipo ingreso*

En la Figura 15 y 16 se tiene la opción de siguiente, guardar y cancelar , si elige guardar se inserta el registro y se re direcciona a la pantalla de inicio Figura 14, si el usuario selecciona la opción siguiente se debe guardar el registro y re direccionar a la Figura 18 si es de ingeniería, Figura 21 si es maestría y Figura 23 si es especialidad.

La Figura 18 solicita los datos de ingreso para ingeniería, aquí se tiene dos opciones siguiente y cancelar si elige la opción cancelar no se registran los datos de ingreso y se re direcciona a la Figura 14 pero si elige la opción siguiente se guardan los datos y se re direcciona a la Figura 19.

Registro guardado correctamente

**Inscripción por examen de INGENIERIA Y TECNOLOGIA**

---

**Inscripción**

Programa Educativo: INGENIERIA EN INFORMATICA      Matrícula: 171181007      Alumno: JUAN PEREZ GALINDO

**Datos de ingreso**

Nombre institución:       Tipo de institución:       Clasificación:       Promedio general:

País:       Estado:       Municipio:       F. Última materia cur:

Período estudios ini:       Período estudios fin:

*Figura 18 Datos de ingreso para ingeniería*

La Figura 19 solicita los documentos generales y adicionales para alumnos extranjeros o mexicanos con estudios en el extranjero, la Figura 20 sólo se mostrará cuando el alumno sea extranjero.

En esta Figura igual se tiene dos opciones la de siguiente y de cancelar, si se elige la opción siguiente se guarda los datos de contra recibo y se re direcciona a la Figura 24 en caso contrario no se guarda el contra recibo y se re direcciona a la Figura 14.

### Contrarecibo de documentos de Licenciatura

Original y copia del acta de nacimiento:

SI  NO

Copia del curp:

SI  NO

Copia del comprobante de domicilio:

SI  NO

Copia de la identificación oficial:

SI  NO

Original y copia del certificado médico:

SI  NO

Original y copia del documento que indique el tipo de sangre:

SI  NO

2 fotografías tamaño infantil, blanco y negro :

SI  NO

Certificado de bachillerato legalizado :

SI  NO

Certificado total o parcial de estudios original, y copia simple :

SI  NO

### Documentos adicionales para alumnos extranjeros o mexicanos con estudios en el extranjero

Certificado total de estudios original apostillado y traducido:

SI  NO

Dictamen nacional de revalidación de estudios:

SI  NO

Copia notariada del documento migratorio(sólo alumnos extranjeros):

SI  NO

Figura 19 Documentación de ingeniería

**Documentos adicionales para alumnos extranjeros**

Número de permiso de ingreso

Tipo de ingreso a México

**Siguiente** **Cancelar**

*Figura 20 Documentación adicional para ingeniería*

La Figura 21 se mostrará cuando el alumno sea de maestría, esta Figura solicita los documentos generales y adicionales para alumnos extranjeros o mexicanos con estudios en el extranjero, la Figura 22 sólo se mostrará cuando el alumno sea extranjero.

En esta Figura igual se tiene dos opciones la de siguiente y de cancelar, si se elige la opción siguiente se guarda los datos de contra recibo de maestría y se re direcciona a la Figura 24 en caso contrario no se guarda el contra recibo y se re direcciona a la Figura 14.

**Contrarecibo de documentos de Posgrado**

Original y copia del acta de nacimiento:  
 SI  NO

Copia del curp:  
 SI  NO

Copia del comprobante de domicilio:  
 SI  NO

Copia de la identificación oficial:  
 SI  NO

Original y copia del certificado médico:  
 SI  NO

Original y copia del documento que indique el tipo de sangre:  
 SI  NO

2 fotografías tamaño infantil, blanco y negro :  
 SI  NO

Fecha del examen profesional  
 x

Fecha de expedición del título  
 x

Pais:  
 ▾

Certificado de licenciatura:  
 SI  NO

Título profesional:  
 SI  NO

Cédula profesional:  
 SI  NO

---

**Documentos adicionales para alumnos extranjeros o mexicanos con estudios en el extranjero**

Certificado total de estudios de licenciatura apostillado y traducido:  
 SI  NO

Título de licenciatura apostillado y traducido:  
 SI  NO

Figura 21 Documentación de maestría

**Documentos adicionales para alumnos extranjeros**

Número de permiso de ingreso

Tipo de ingreso a México

Acta de nacimiento apostillada y traducida(sólo alumnos extranjeros):

SI  NO

Copia notariada del documento migratorio(sólo alumnos extranjeros):

SI  NO

**Siguiente** **Cancelar**

*Figura 22 Documentación adicional para maestría y especialidad*

La Figura 23 se mostrará cuando el alumno sea de especialidad, esta Figura igual solicita los documentos generales y adicionales para alumnos extranjeros o mexicanos con estudios en el extranjero, la Figura 22 sólo se mostrará cuando el alumno sea extranjero.

En esta Figura igual se tiene dos opciones la de siguiente y de cancelar, si se elige la opción siguiente se guarda los datos de contra recibo de especialidad y se re direcciona a la Figura 24 en caso contrario no se guarda el contra recibo y se re direcciona a la Figura 14.

**Contrarecibo de documentos de Especialidad**

Original y copia del acta de nacimiento:  
 SI  NO

Copia del curp:  
 SI  NO

Copia del comprobante de domicilio:  
 SI  NO

Copia de la identificación oficial:  
 SI  NO

Original y copia del certificado médico:  
 SI  NO

Original y copia del documento que indique el tipo de sangre:  
 SI  NO

Fecha del examen profesional  
  
✖

Fecha de expedición del título  
  
✖

Pais:

Certificado de licenciatura:  
 SI  NO

Título profesional:  
 SI  NO

Cédula profesional:  
 SI  NO

**Documentos adicionales para alumnos extranjeros o mexicanos con estudios en el extranjero**

Certificado total de estudios de licenciatura apostillado y traducido:  
 SI  NO

Título de licenciatura apostillado y traducido:  
 SI  NO

Figura 23 Documentación de especialidad

Las siguientes figuras que a continuación se muestran son para las tres áreas de conocimientos que anteriormente se mencionaron.

En la Figura 24 se solicitan los datos de domicilio de procedencia y residencia, Cuando el usuario seleccione la opción siguiente se deben guardar todos los registros solicitados en esta pantalla y re direccionar a la siguiente pantalla ver Figura 25 en caso contrario se re direcciona a la Figura 14.

Registro guardado correctamente

### Inscripción por examen de INGENIERIA Y TECNOLOGIA

**Inicio**

Programa Educativo: INGENIERIA EN INFORMATICA      Matricula: 171101007      Alumno: JUAN PEREZ GALINDO

**Datos de domicilio de procedencia**

Calle:       Número:       Colonia:       Código postal:

País:       Estado:       Municipio:

**Datos de domicilio de residencia**

Calle:       Número:       Colonia:       Código postal:

País:       Estado:       Municipio:

*Figura 24 Domicilios de procedencia y residencia*

En la Figura 25 se solicitan los datos del padre y madre posteriormente se debe indicar quien es el tutor legal y tutor económico, puede ser el padre o la madre u otro, si se elige otro se debe teclear la información de la persona. Cuando se seleccione guardar se deben guardar todos los registros de los tutores del alumno y si se selecciona cancelar se re direcciona a la Figura 14.

Registro guardado correctamente

## Inscripción por examen de INGENIERIA Y TECNOLOGIA

### Datos

Programa Educativo: INGENIERIA EN INFORMATICA

Matricula: 171101007

Alumno: JUAN PEREZ GALFIDO

Nacionalidad: EXTRANJERA

### Datos del padre

Nombre(s)	Apellido paterno	Apellido materno	Vive: <input type="radio"/> SI <input type="radio"/> NO
Calle	Número	Colonia	Código postal
País	Estado	Municipio	Parentesco: PADRE
Estado civil	Teléfono	Celular	E-mail

### Datos de la madre

Nombre(s)	Apellido paterno	Apellido materno	Vive: <input type="radio"/> SI <input type="radio"/> NO
Calle	Número	Colonia	Código postal
País	Estado	Municipio	Parentesco: MADRE
Estado civil	Teléfono	Celular	E-mail

### Tutor

Tutor legal:  PADRE  MADRE  OTRO

Tutor económico:  PADRE  MADRE  OTRO

### Tutor

Tutor legal:  PADRE  MADRE  OTRO

### Datos de otro tutor legal

Nombre(s)	Apellido paterno	Apellido materno	Vive: <input type="radio"/> SI <input type="radio"/> NO
Calle	Número	Colonia	Código postal
País	Estado	Municipio	Parentesco: OTRO
Estado civil	Teléfono	Celular	E-mail

Tutor económico:  PADRE  MADRE  OTRO

### Datos de otro tutor económico

Nombre(s)	Apellido paterno	Apellido materno	Vive: <input type="radio"/> SI <input type="radio"/> NO
Calle	Número	Colonia	Código postal
País	Estado	Municipio	Parentesco: OTRO
Estado civil	Teléfono	Celular	E-mail

Cancelar

Guardar

Figura 25 Datos del tutor



En la Tabla 1 se muestran los permisos establecidos a cada vista

Número de pantalla	Ruta	Descripción	Permisos			
			Agregar	Modificar	Eliminar	Consultar
365	/insAlumnos	Inscripción de alumnos	✓	X	x	X
366	/regAlumnoDP	Agregar datos personales del alumno				
367	/regAlumnoDI	Agregar datos de ingreso del alumno	✓	X	x	x
368	/regAlumnoDoctosI	Agregar documentos de alumno de ingeniería	✓	X	x	x
369	/regAlumnoDoctosP	Agregar documentos de alumno de posgrado	✓	X	X	x
370	/regAlumnoDoctosE	Agregar documentos de alumno de especialidad	✓	X	x	X
371	/regAlumnoDomicilio	Agregar domicilios del alumno	✓	X	x	x
372	/regAlumnoTutor	Agregar tutores del alumno	✓	X	X	x

*Tabla 1 Permisos de vistas*

### 3.5 Evaluación

Las pruebas son el primer filtro de calidad que debe superar el software, a cargo de un equipo profesional distinto al de desarrollo. En esta fase del ciclo de vida se refleja en gran medida el grado de medios y esfuerzo que se han puesto para obtener una buena aplicación[6].

Con esta etapa se concluye el ciclo de la metodología empleada, en este apartado se mostraran las pruebas que se llevaron a cabo para comprobar el funcionamiento del sistema.

#### 3.5.1 Pruebas de seguridad

Para llevar a cabo las pruebas se ejecutó el proyecto en el servidor thin en el puerto 3000, fue necesario utilizar otra computadora que estuviera conectada a la misma red, porque en ella se instaló el software necesario para ejecutar las pruebas.

Se utilizaron tres escáner de vulnerabilidades de aplicaciones web a continuación se describe cada uno de ellos:

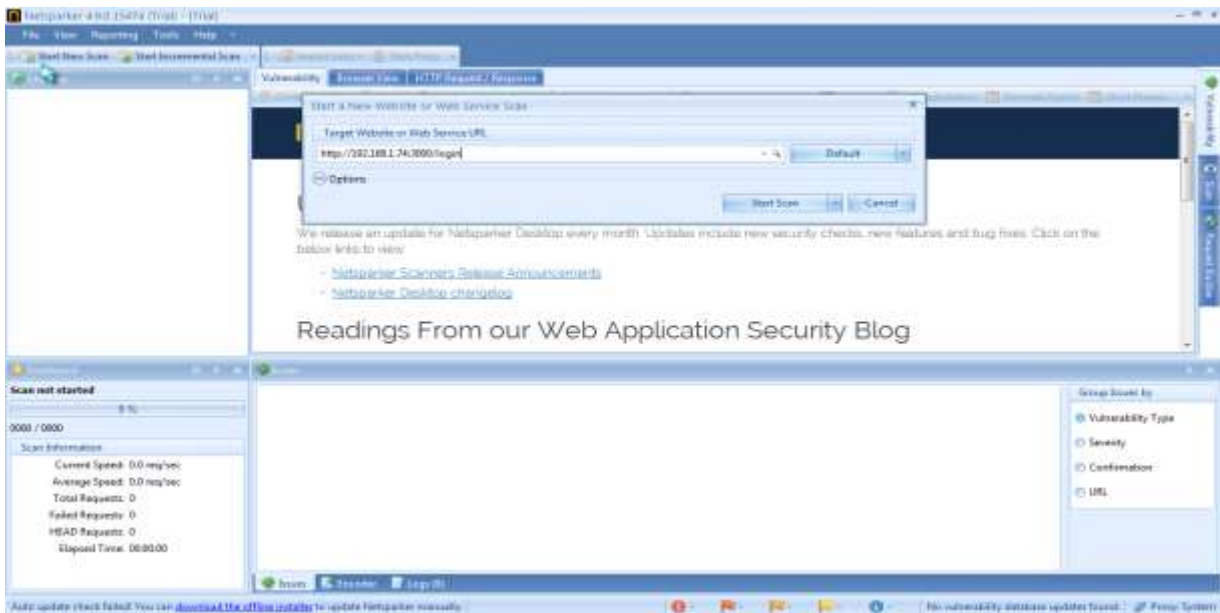
## Netsparker

Netsparker puede rastrear, atacar e identificar vulnerabilidades en todas las aplicaciones Web personalizadas, independientemente de la plataforma y la tecnología en que se basan, al igual que un atacante real [7].

Es capaz de identificar las vulnerabilidades de aplicaciones web, como la inyección de SQL, Cross-Site Scripting (XSS), ejecución remota de código y muchos más.

Se utilizó la versión gratuita de este software.

Lo primero que se hace es elegir la opción de start new scan ver Figura 26 y está pide la dirección del sitio web, en este caso se usó: `http://192.168.1.74:3000/Login`



*Figura 27 Inicio de escaneo con Netsparker*

En la Figura 27 se muestra el proceso de escaneo

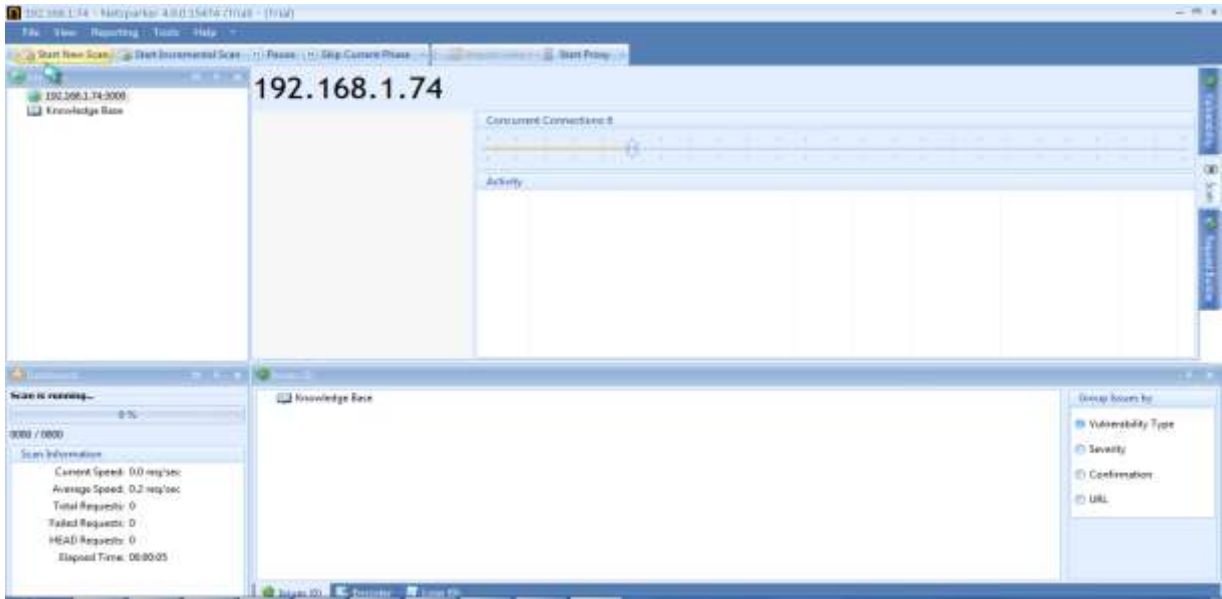


Figura 27 Proceso de escaneo

En la Figura 28 se muestra los resultados obtenidos del escaneo, indica que encontró 6 vulnerabilidades de tipo información, 3 en nivel bajo, 0 en nivel medio, 1 en nivel importante y 0 en nivel crítico.

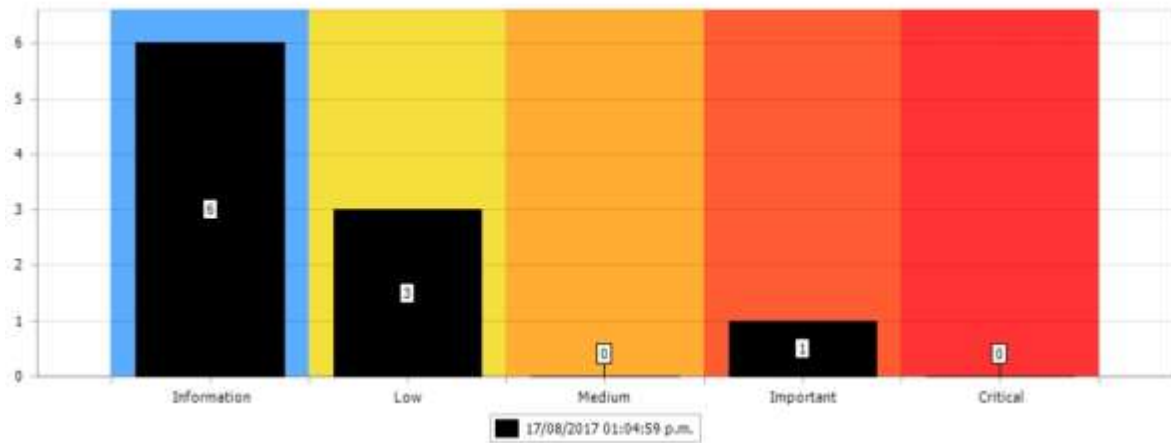


Figura 28 Resultados de escaneo

En la Tabla 2 se muestra el detalle de las vulnerabilidades encontradas

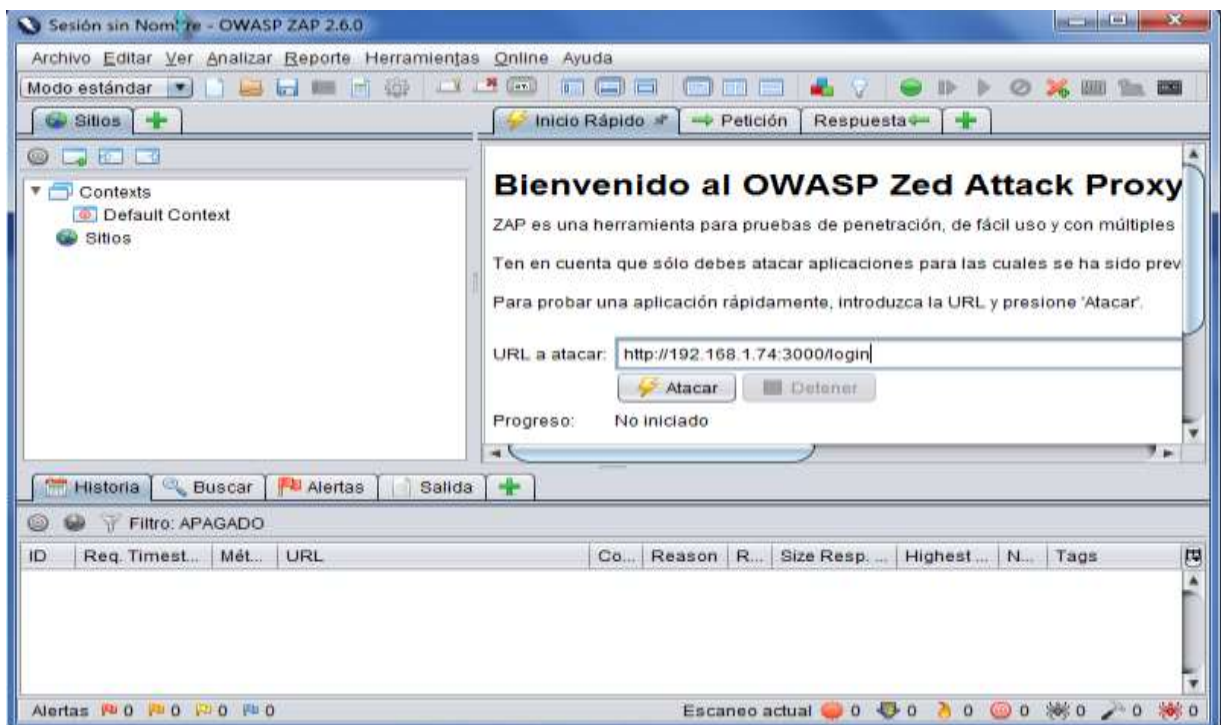
	<b>Vulnerabilidad</b>
	Contraseña transmitida a través de http
	Error de servidor interno
	Autocompletar activado
	Posible falsificación de solicitudes entre sitios en el formulario de inicio de sesión
	Divulgación de la dirección de correo electrónico
	Autocompletado activado (campo de contraseña)
	Falta el encabezado X-XSS-Protection (Evita inyección de código malicioso)
	Cookie de SameSite no implementada
	No se implementó la Directiva de seguridad de contenido (CSP)
	Referrer-Policy no implementada (Evita que la URL se filtre)

*Tabla 2 Lista de vulnerabilidades*

## OWAS Zep

Esta herramienta es de código abierto y es desarrollado por AWASP. Está disponible para las plataformas de Windows, Unix / Linux y Macintosh [7].

Para iniciar el escaneo con esta aplicación se debe seleccionar archivo y luego nuevo escaneo, posteriormente solicita la URL a atacar ver Figura 29:



*Figura 29 Inicio de escaneo OWASP Ze*

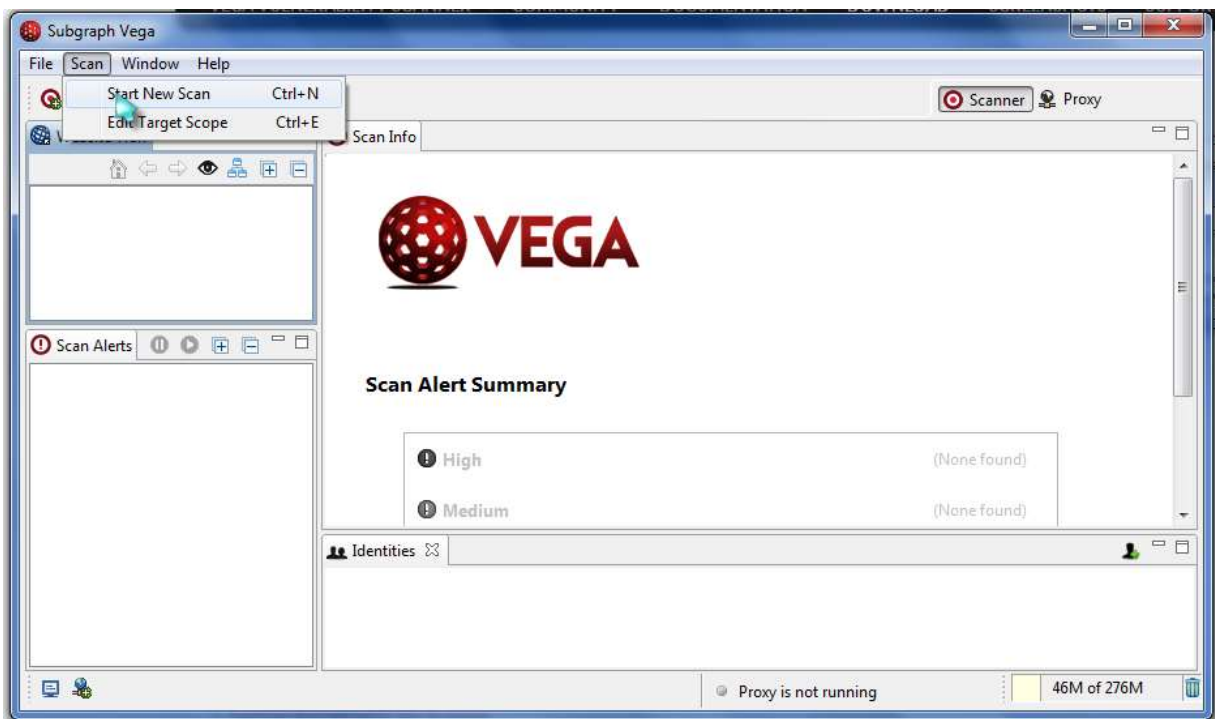


## Vega Scanner

Vega es otro escáner de vulnerabilidades web de código abierto. Con esta herramienta se puede realizar pruebas de seguridad de una aplicación web. Esta herramienta está desarrollada en Java y ofrece un entorno basado en GUI. Está disponible para OS X, Linux y Windows [7].

Se puede utilizar para encontrar inyección SQL, inyección de cabecera, listado de directorios, inyección cáscara, cross site scripting, la inclusión de archivos y otras vulnerabilidades de las aplicaciones web. Esta herramienta también se puede ampliar mediante una potente API desarrollada en JavaScript.

Para iniciar el escaneo se selecciona scan y se elige la opción start new scan ver Figura 32



*Figura 32 Inicio de escaneo en vega*

Posteriormente se debe introducir la URL del objetivo ver Figura 33, se elige la opción siguiente y muestra los dos módulos que se tiene ver Figura 34, en el cual se eligen los tipos de escaneo ver Figura 35



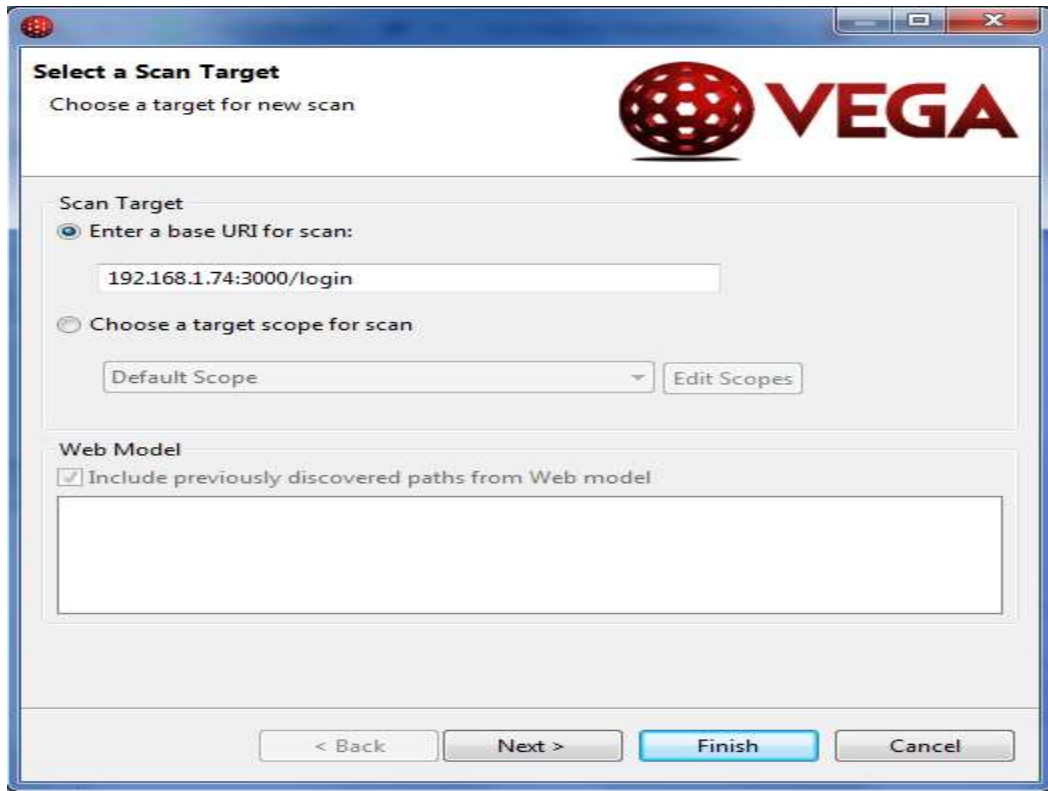


Figura 33 Selección del objetivo

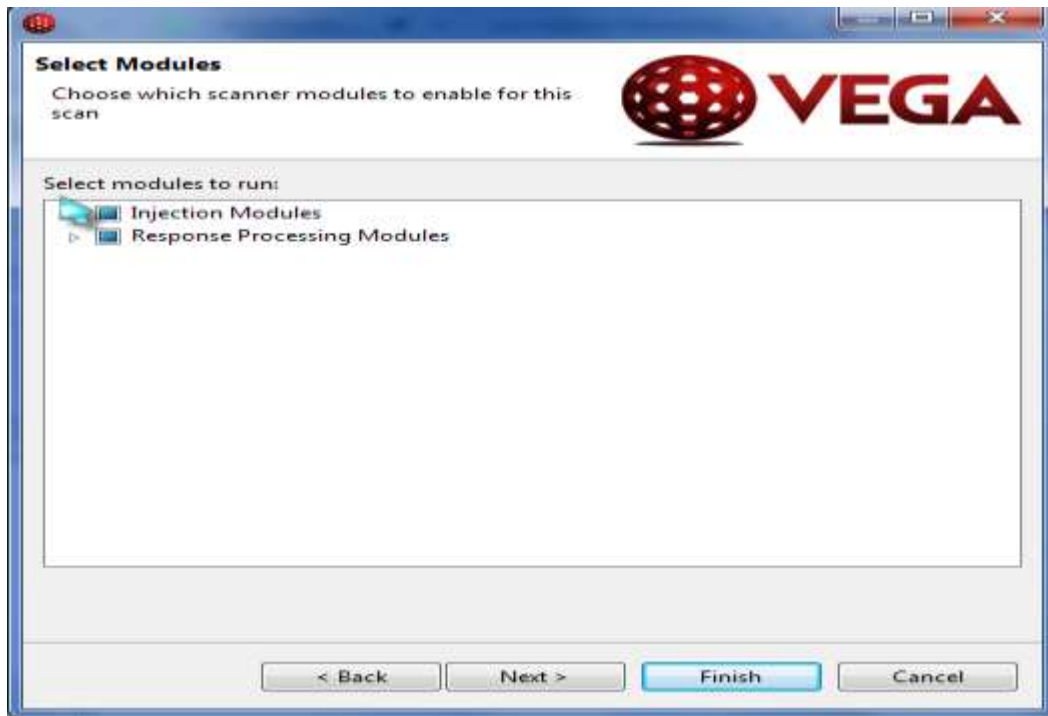
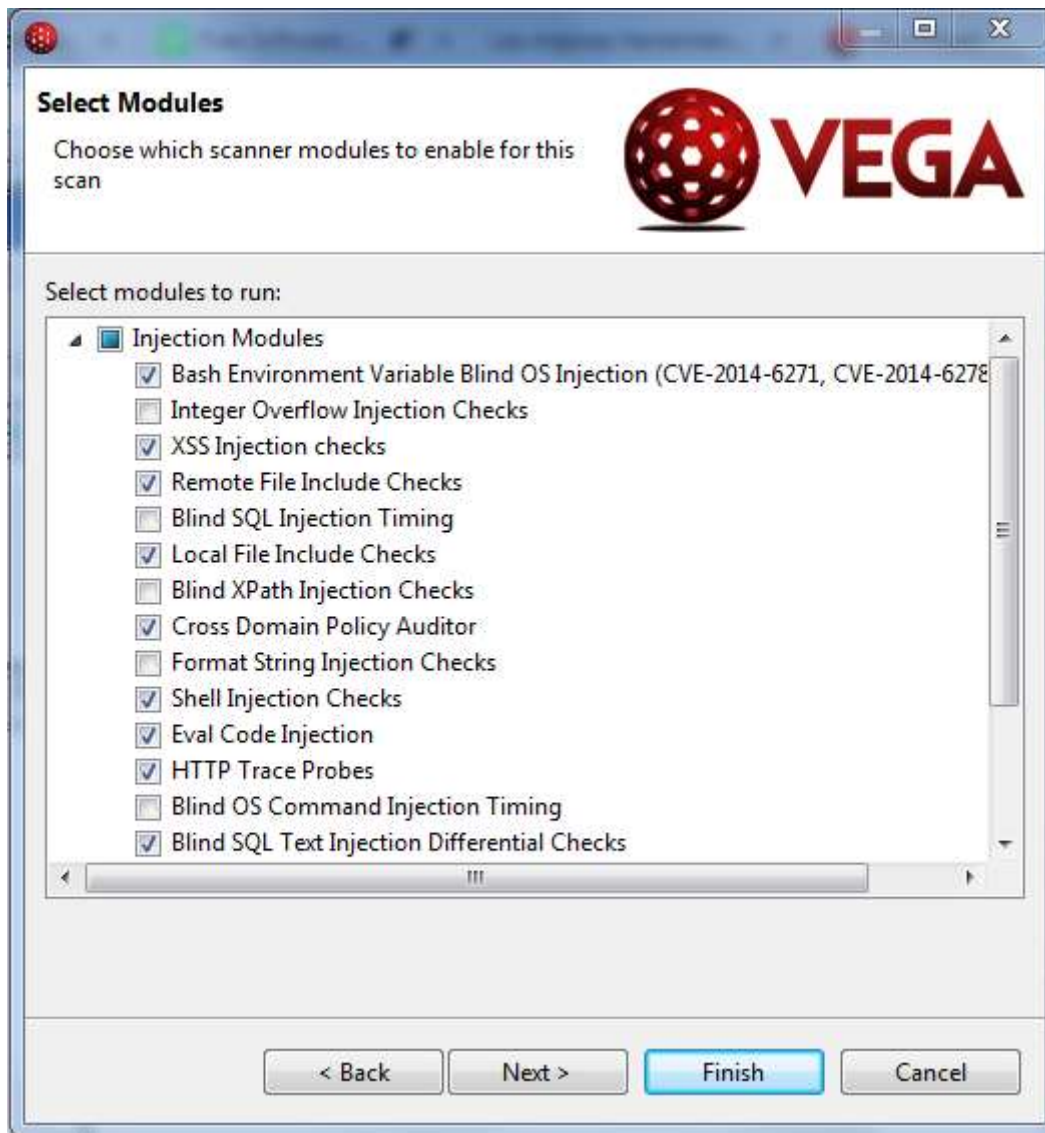


Figura 34 Módulos de escaneo



*Figura 35 Selección de escaneo*

En la Figura 36 se muestra el proceso de escaneo y en la Figura 37 el total de las alertas que obtuvo el escaneo, en este caso muestra que se tiene dos alertas una de nivel alta y una de nivel bajo.



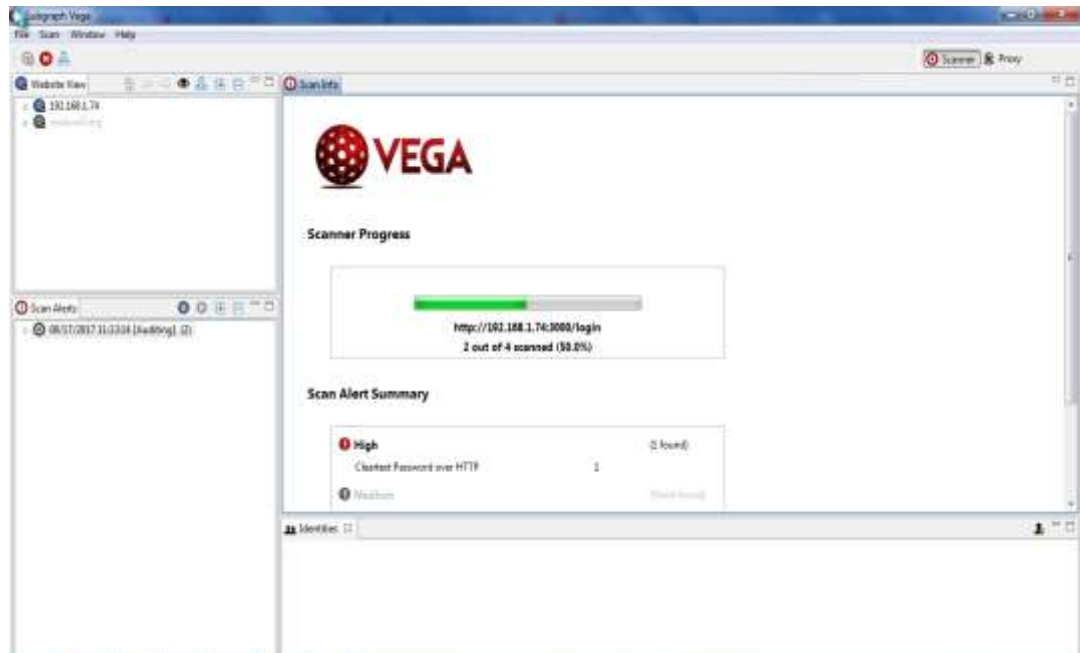


Figura 35 Proceso de escaneo

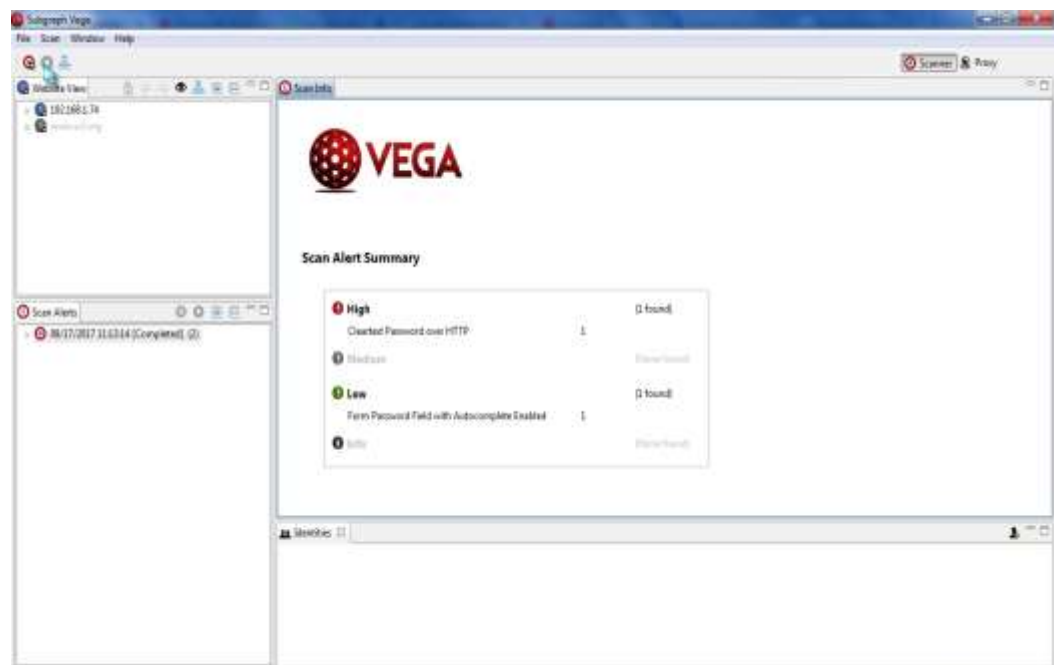


Figura 36 Total de alertas del escaneo

Esta herramienta indica el nivel de la alerta, donde obtuvo la alerta (petición), la discusión, el impacto de esta y una posible solución (recomendación).

En la Figura 37 indica todo lo anteriormente mencionado de la alerta alta porque el input de la contraseña se está enviando por http para esta alerta recomienda nunca enviar la contraseña sobre texto, que podría enviarse por https.

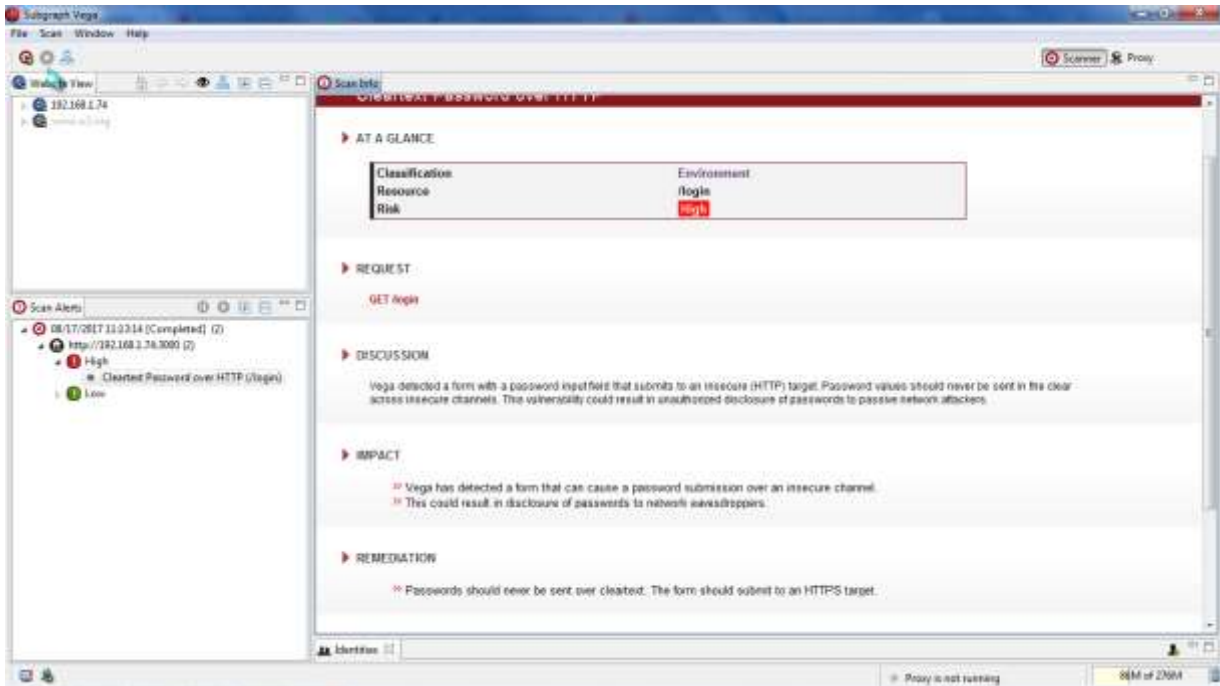


Figura 36 Recomendación alerta de riesgo alto

En la Figura 37 indica que existe una alerta de riesgo bajo, la alerta es que se tiene el autocompletado de contraseña en el navegador, la recomendación para esta es que se declare el atributo de autocompletado con el valor off.



Figura 37 Recomendación alerta de riesgo bajo

## Conclusión

En toda tarea que realizamos en nuestra vida cotidiana tenemos que tener un orden y establecer distintas prioridades para que la actividad que nos proponemos tenga su respectivo éxito y podamos alcanzar un objetivo que nos hemos planteado desde un principio, por ello considero que la importancia de las metodologías en el desarrollo de software son muy importantes.

En este proyecto trabajé con diferentes herramientas de las cuales en la mayoría era por primera vez, pero adquirí nuevos conocimientos, aprendí a instalar el adaptador que permite DataMapper, a instalar gemas, especificar la conexión de la base de datos con el adaptador, definir modelos (asociaciones) con DataMapper y cómo trabaja Sinatra con ruby mediante el modelo MVC.

Por otra parte, del lado del cliente se crearon diferentes vistas que contiene la información solicitada por servicios escolares, mediante estas se obtiene la información necesaria para inscribir a un alumno.

Al investigar sobre las pruebas me di cuenta que hay muchas formas de proteger las aplicaciones web, sólo que a veces desconocemos por completo.

Los resultados obtenidos en las tres pruebas que se realizaron indican que es muy importante la elección del método de envío de información y que no es muy confiable dejar las contraseñas en el navegador.

El análisis de requerimientos del sistema no fue levantado adecuadamente ya que aún hay cosas que no están muy claras o cosas que son incongruentes.

## Bibliografía

[1] WordPress (2017). MPlu +A. 16-08-2017 Sitio web:

[http://www.grihotools.udl.cat/mpiua/fases-mpiua/#caracteristicas\\_mpiua](http://www.grihotools.udl.cat/mpiua/fases-mpiua/#caracteristicas_mpiua)

[2] Matz, hablando en la lista de correo Ruby-Talk, 12 de mayo del 2000.

Kehoe,D.. (2013). Ruby on Rails. 17-08-2016, de Ruby APPS Sitio web:

<http://railsapps.github.io/what-is-ruby-rails.html>

[3] Dave Thomas. "Programming Ruby - The Pragmatic Programmer's Guide" Addison Wesley Longman, Inc. (2011)

[4] Thomas Lockhart, "Tutorial de PostgreSQL", Equipo de desarrollo de PostgreSQL. (2000).

[5] Ingeniería del software, un enfoque práctico 6a edición, Roger S. Pressman, McGraw Hill

[6] IBM Security APP Scan. 17-08-2017 Sitio web:

[https://www.ibm.com/support/knowledgecenter/es/SSW2NF\\_9.0.1/com.ibm.ase.help.doc/topics/c\\_best\\_practices\\_performance.html](https://www.ibm.com/support/knowledgecenter/es/SSW2NF_9.0.1/com.ibm.ase.help.doc/topics/c_best_practices_performance.html)

[7] Desarrollo-geek Los mejores 8 scanners de vulnerabilidades. 17-08-2017 Sitio

web:[https://www.ibm.com/support/knowledgecenter/es/SSW2NF\\_9.0.1/com.ibm.ase.help.doc/topics/c\\_best\\_practices\\_performance.html](https://www.ibm.com/support/knowledgecenter/es/SSW2NF_9.0.1/com.ibm.ase.help.doc/topics/c_best_practices_performance.html)