**SPE: Sistema de Pasantías Empresariales**

**Documento de la Arquitectura del Software**

**Versión 1.0**

**Historia de Revisión**

|  |  |  |  |
| --- | --- | --- | --- |
| **Fecha** | **Versión** | **Descripción** | **Autor** |
| 05/03/2016 | 1.0 | DAS unificado | InnovaTech  S&Q Solutions  Sis+  Hexacode |
| 25/04/2016 | 1.1 | Revision | E-Corp |
| 13/05/2016 | 1.2 | Se agregó diagrama ERE y diccionario de datos | E-corp |
| 19/05/2016 | 1.3 | Diagramas wae | E-Corp |

**Tabla de Contenidos**

1. Introducción

1.1 Propósito.

1.2 Alcance.

1.3 Definiciones, Siglas, y Abreviaciones.

1.4 Referencias.

1.5 Vista Global.

2. Representación Arquitectónica.

3. Metas y Restricciones Arquitectónicas.

4. Vista de Casos de Uso.

5. Vista Lógica.

5.1 Visión general.

5.2 Paquetes de Diseño Significativos Arquitectónicamente.

5.3 Realizaciones de los casos de uso.

6. Vista de Procesos.

7. Vista de Implantación.

8. Vista de Implementación.

8.1 Vista General.

8.2 Capas.

9. Vista de Datos.

10. Tamaño y Desempeño.

11. Calidad.

**Documento de la Arquitectura del Software**

1. **Introducción**

Según la metodología RUP, la fase de Análisis y Diseño es importante para las demás, dado que, ésta define la estructura, componentes, propiedades e interconexiones del sistema. El presente documento muestra la arquitectura de software utilizada para el desarrollo del Sistema de Pasantías Empresariales(SPE) de la Universidad Simón Bolívar. También, se detallan el propósito, alcance, definiciones y referencias importantes para el usuario.

* 1. **Propósito**

El propósito de este documento es describir de forma concisa los elementos que componen al Sistema de Pasantías Empresariales(SPE) de la Universidad Simón Bolívar, a través de distintas vistas que desarrollan un aspecto particular del sistema . Además, se muestra el tamaño, desempeño y criterios de calidad del sistema.

* 1. **Alcance**

Este documento se realizó siguiendo el modelo de vistas de arquitectura 4+1 diseñado por Philippe Kruchten que describen la arquitectura de un sistema de software en cuatro vistas: Lógica, Desarrollo, Proceso y Física junto a una vista de *escenarios*.

Se desarrolló la descripción del sistema centrándonos en la vista lógica.Además, se muestran casos de uso junto a un diagrama de actividad que forman parte de la vista de *escenarios.*

Se utiliza una vista de datos para mostrar la estructura de la base de datos usada en el SPE.

* 1. **Definiciones, Siglas, y Abreviaciones**

Definiciones:

*Stakeholder*: personas involucradas con el sistema, principalmente interesadas en que sus necesidades sean satisfechas.

*Backup*: Respaldo de la información contenida en la base de datos del sistema.

*Framework*: Entorno de trabajo para el desarrollo del sistema, en nuestro caso, *Web2Py*.

Abreviaciones:

* USB: Universidad Simón Bolívar.
* DEX: Decanato de Extensión.
* CCT: Coordinación de Cooperación Técnica.
* CAS: Servicio Centralizado de Autenticación.
* SPE: Sistemas de Pasantías Empresariales.
* DST: Dirección de Servicios Telemáticos.
* 1. **Referencias**

*Documento Visión*. 01/02/2016. Equipo Sist+.

*Documento Visión*. 01/02/2016. Equipo Hexacode.

*Documento Visión*. 01/02/2016. Equ*ipo InnovaTech.*

*Documento Visión. 01/02/2016. Equipo: S&Q Solutions*.

* 1. **Vista Global**

El siguiente documento muestra, de manera general, las decisiones arquitectónicas del desarrollo del Sistema mediante notaciones, modelos y estándares utilizados para la representación del mismo. Se detalla cada una de las vistas a desarrollar, incluyendo diagramas y modelos necesarios para una mayor comprensión de la arquitectura. Se incluye la Vista de Casos de Uso y se muestra la Vista de Datos, que comprende el Diagrama de ERe.

El resto del documento contiene la representación arquitectónica, metas y restricciones arquitectónicas, las diferentes vistas: casos de uso, lógica, procesos, implantación, implementación, datos, el tamaño y desempeño y calidad.

1. **Representación Arquitectónica**

|  |  |  |
| --- | --- | --- |
| **Vista** | **Elemento de modelado** | **Descripción** |
| Vista de Casos de Uso | Modelo de casos de uso de SPE | Representa los procesos globales de la aplicación y los tipos de usuarios que interactúan en dichos procesos. |
| Vista Lógica | Representa la estructura conceptual del software | Representación de las herramientas y procesos involucrados en la solución. Describen las partes del modelo de diseño que son significativas arquitectónicamente. |
| Vista de Datos | Diagrama ER | Muestra una visión de cómo se estructuran las tablas dentro del Sistema. |
| Vista de Implementación | Diagrama de Despliegue | Se describen las configuraciones de redes físicas (hardware), sobre las cuales el software será desarrollado. |

1. **Metas y Restricciones Arquitectónicas**

El objetivo principal buscado por el grupo es proporcionar a los usuarios un Sistema que presente las siguientes características:

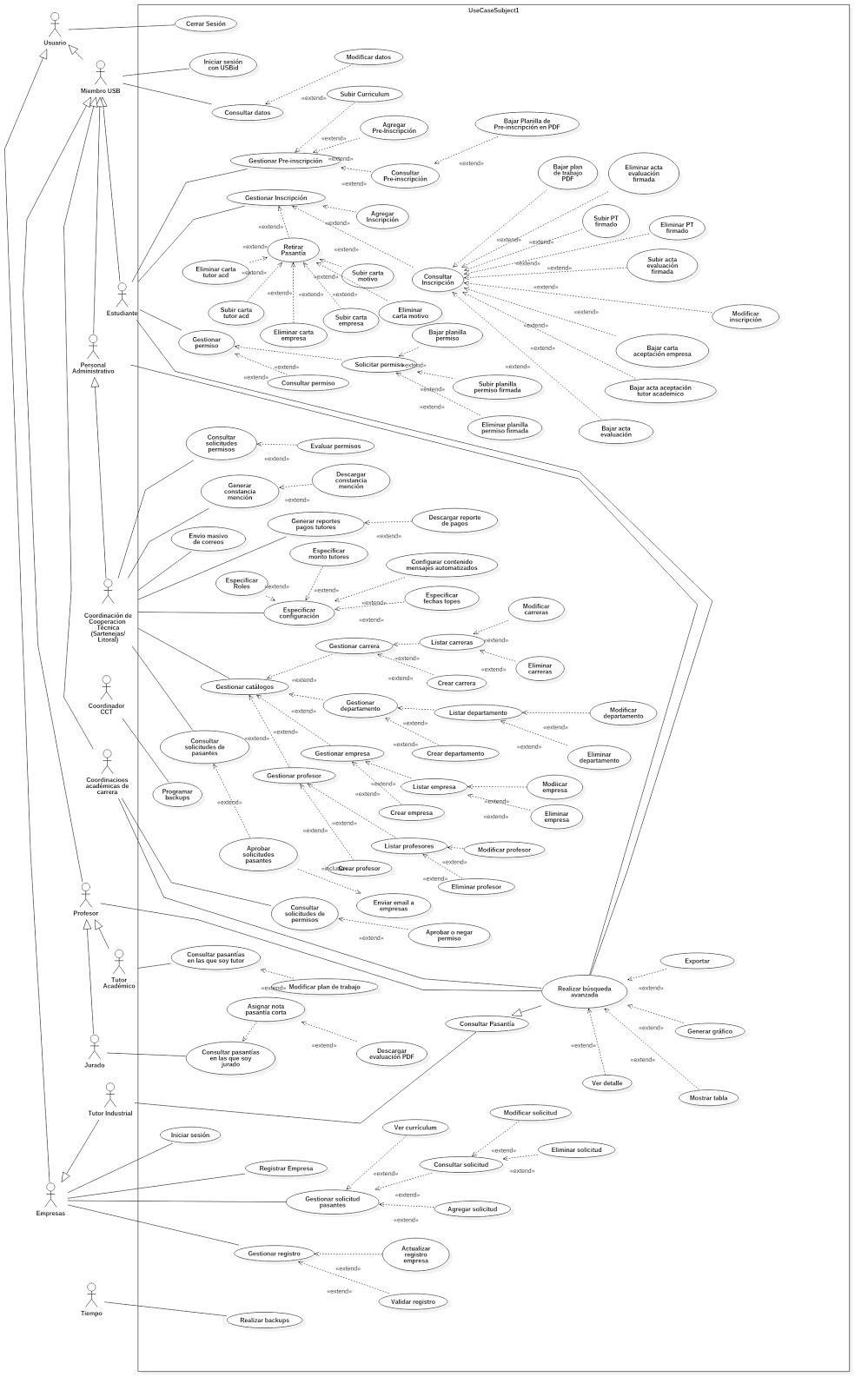
* Mantenibilidad: Los programadores deben requerir la menor cantidad de esfuerzo para localizar y arreglar errores como también actualizarlo. Igualmente que el sistema este diseñado de manera modular, es decir que todas las funcionalidades estén encapsuladas, facilitando la agregación de nuevas funciones.
* Practicidad: Se busca que el sistema cumpla el principio de eficiencia abarcando los conceptos de usabilidad y portabilidad.
* Usabilidad: Los usuarios deben requerir el menor esfuerzo posible para aprender, operar y preparar los datos de entrada e interpretar las salidas del sistema. Todo esto con una interfaz sencilla, amigable e intuitiva de manera que sea mas fácil y atractivo para los usuarios.

Restricciones halladas, durante el desarrollo del proyecto son:

* El diseño debe seguir los lineamientos dados por los clientes, de manera que cubra todas las necesidades
* Restricciones de seguridad y privacidad: Es necesario que ciertas gestiones y algunos procesos sean permitidos exclusivamente en fechas específicas. Al igual que la permisología de acceso, manejo de roles y funciones establecidas.
* Compatibilidad: el sistema debe ser compatible con la mayoría de los navegadores, garantizándole a los usuarios el acceso al mismo.

1. **Vista de Casos de Uso**

|  |  |
| --- | --- |
| **Actores** | **Descripción** |
| Usuario |  |
| Miembro USB | Persona perteneciente a la USB, que posea USBID. |
| Estudiante | Persona perteneciente a la USB, que se encuentra cursando una carrera de pregrado. |
| Coordinaciones Académicas de Carrera | Coordinaciones correspondientes a las diversas carreras que ofrece la USB. |
| Profesores | Profesores que participen en el Sistema de Pasantías Empresariale como tutores o como jurado. |
| Empresas | Empresas en las que estudiantes de la USB han realizado o realizaran pasantías. |
| Tutor Industrial | Persona que representa a una empresa, en la cual algún estudiante de la USB está realizando una pasantía |
| SysAdmin | Persona que tiene permisos especiales para modificar todos los catalogos. |
| Tiempo | Tiempo necesario para realizar un backup. |



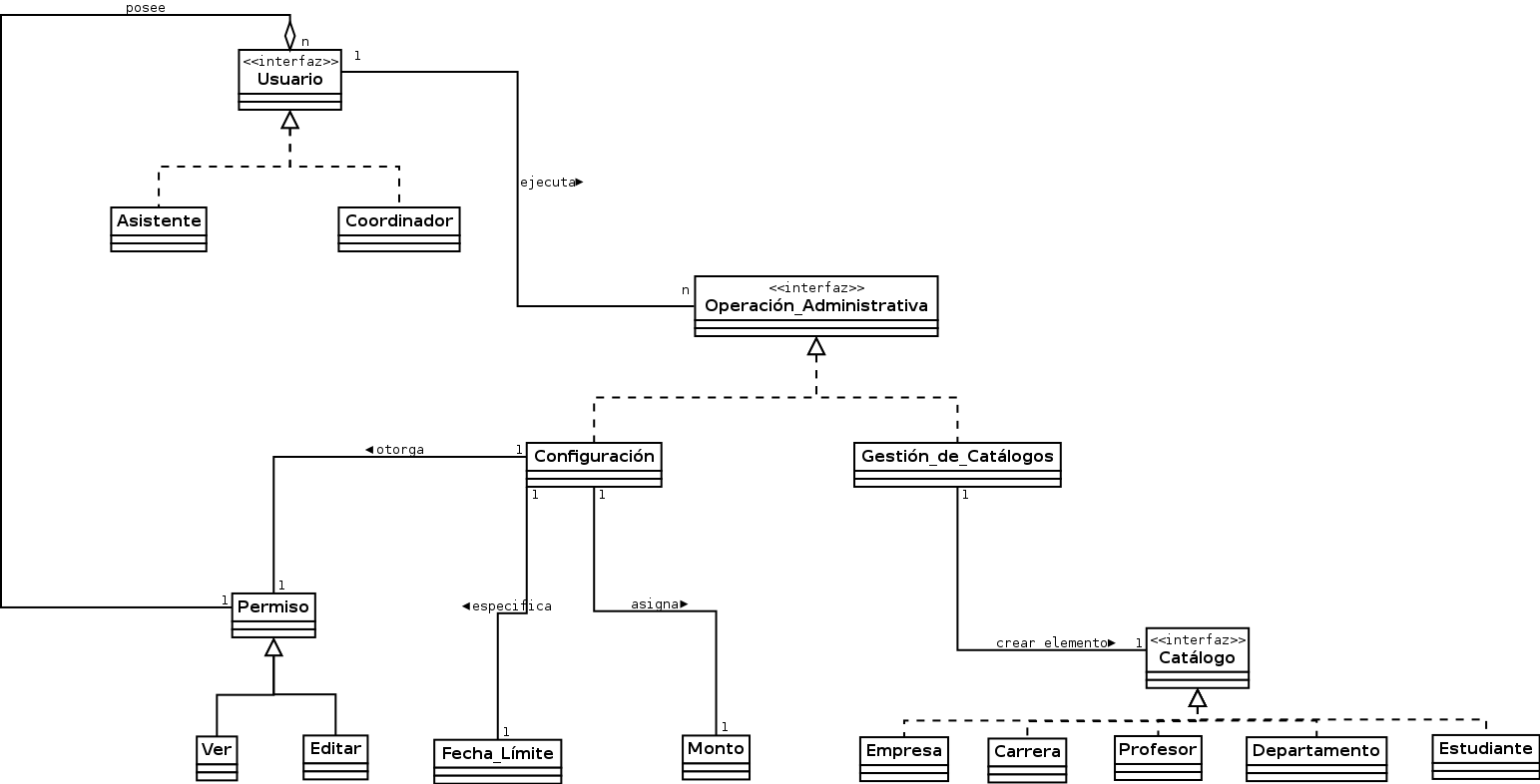
1. **Vista Lógica**

Se utiliza MVC, Modelo-Vista-Controlador, que posee tres componentes:

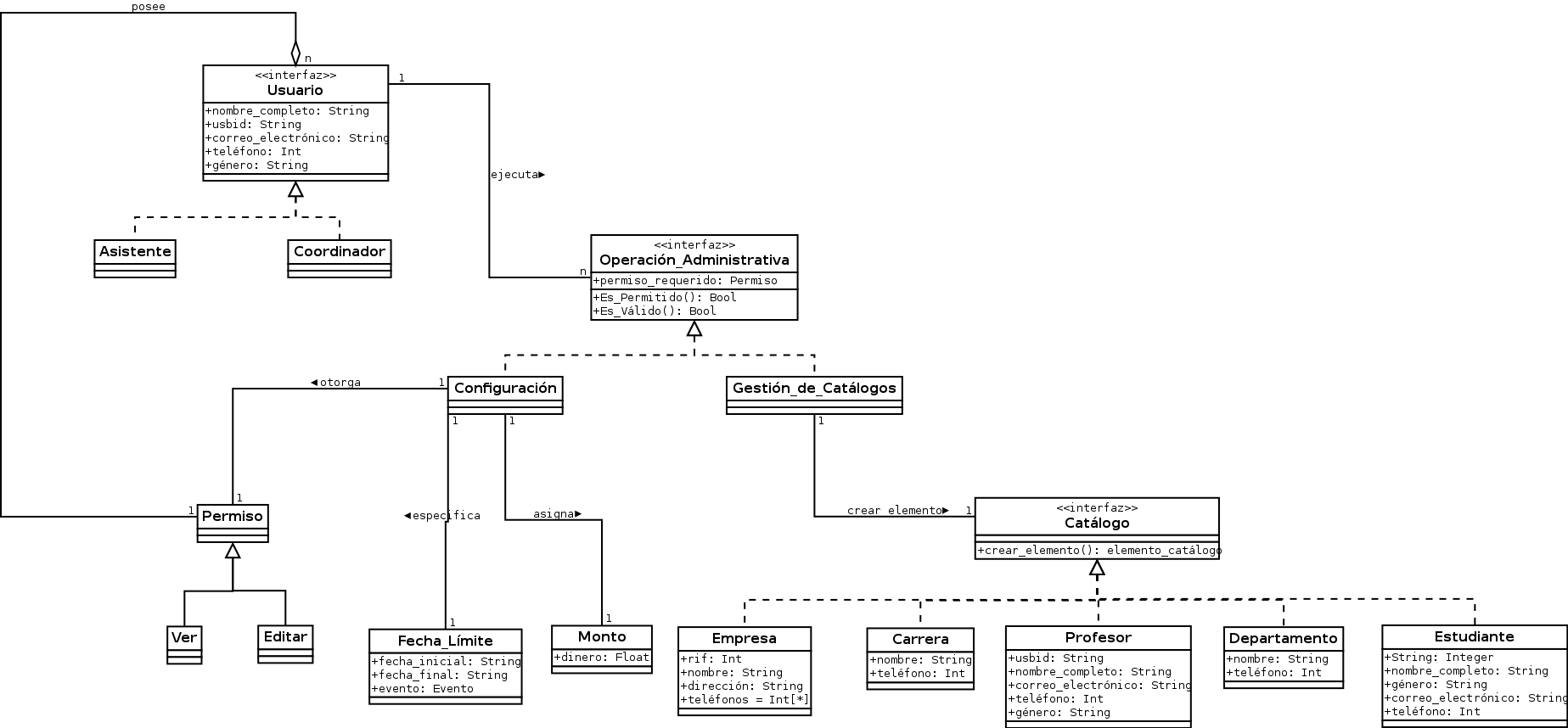
* Modelo
* Vista
* Controlador

Los archivos comienzan con la inicial del nombre del componente al que corresponden.

* 1. **Visión general**

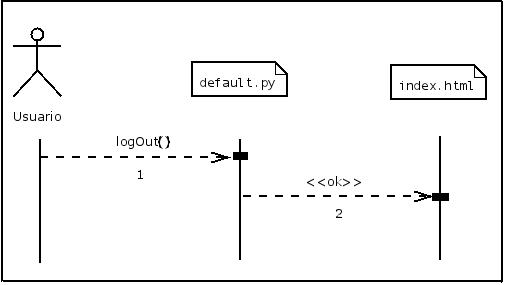


* 1. **Paquetes de Diseño Significativos Arquitectónicamente**

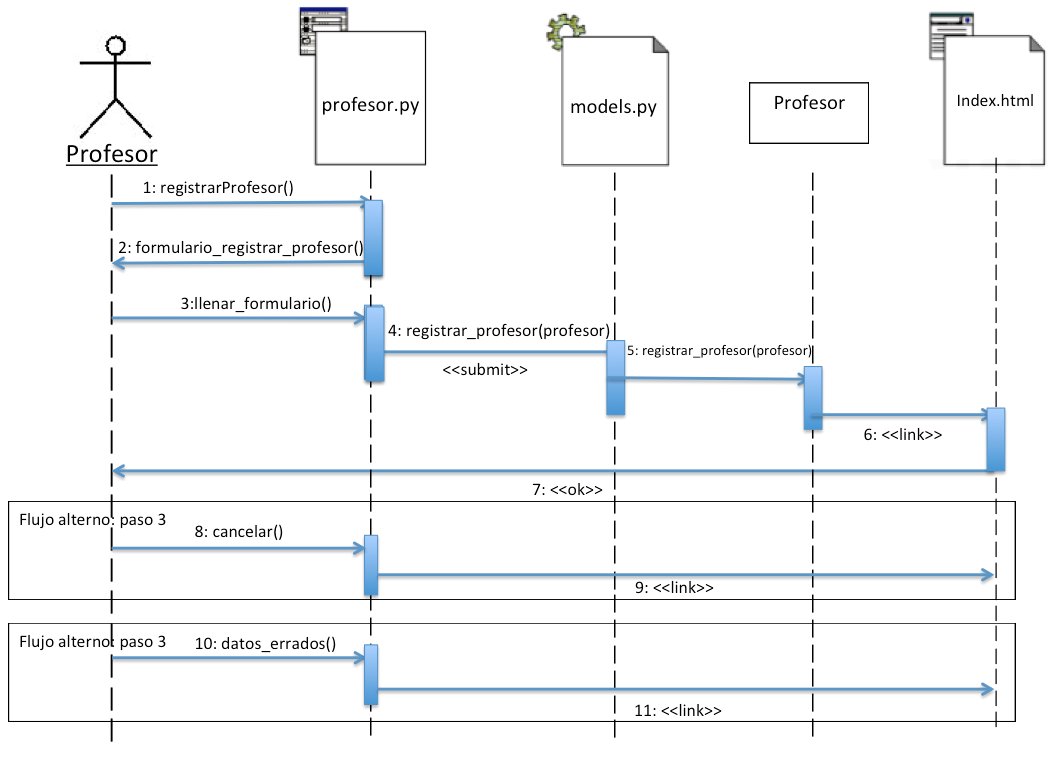


* 1. **Realizaciones de los casos de uso**

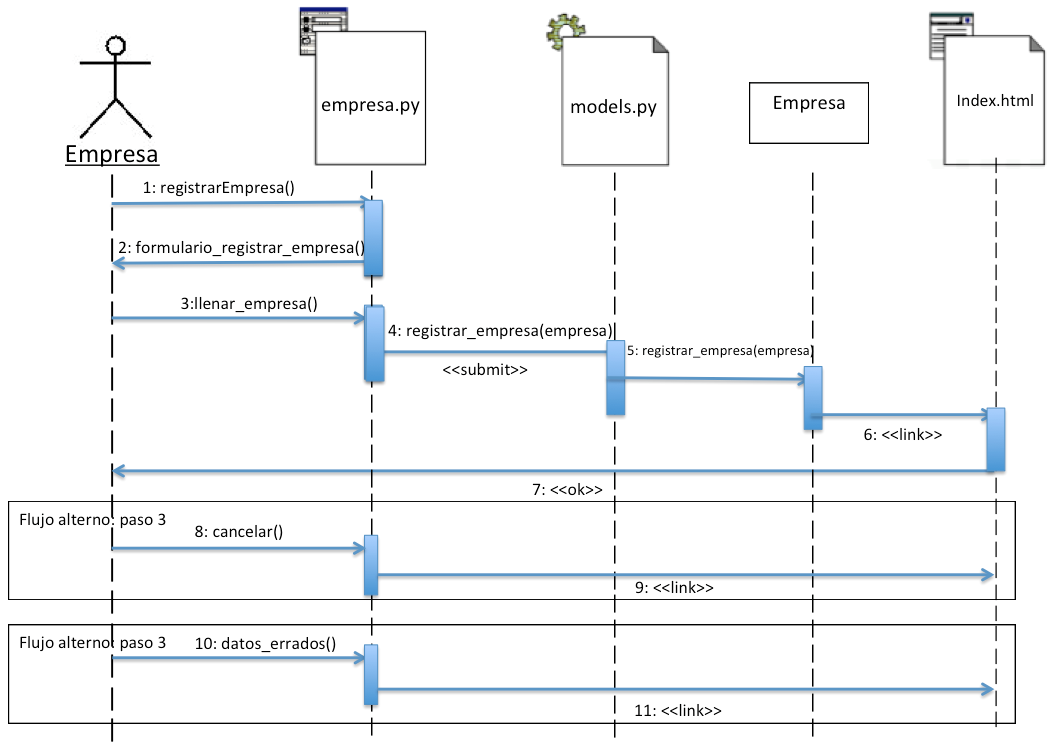
**5.3.1 Cerrar Sesión**



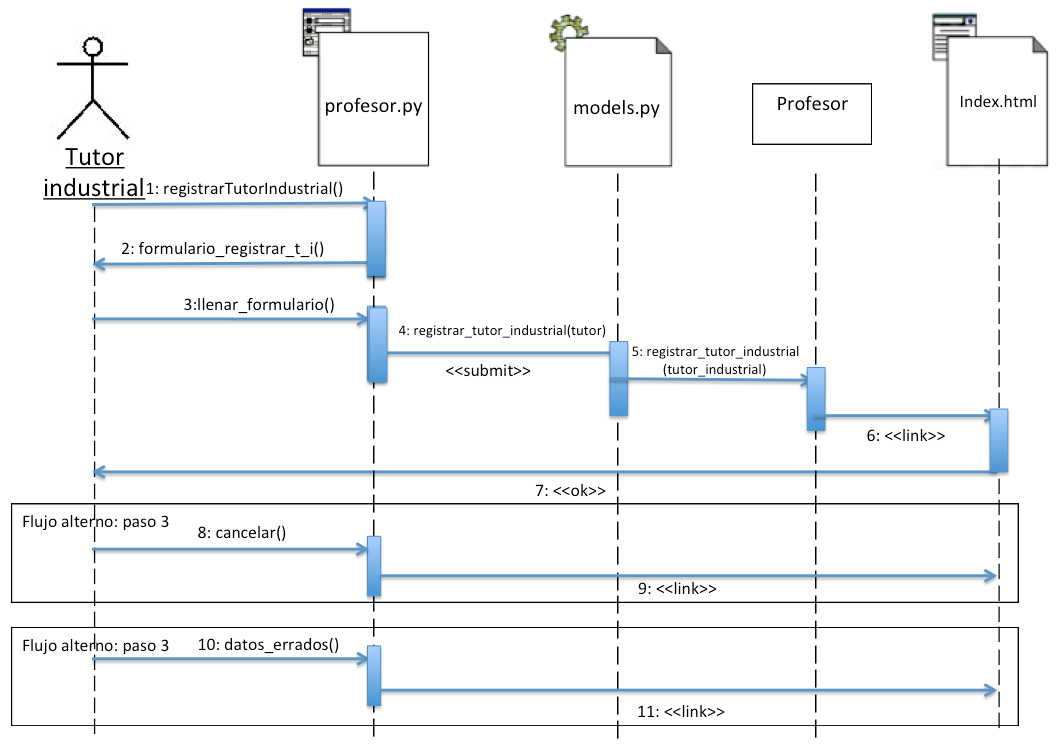
5.3.2 **Registrar Profesor**



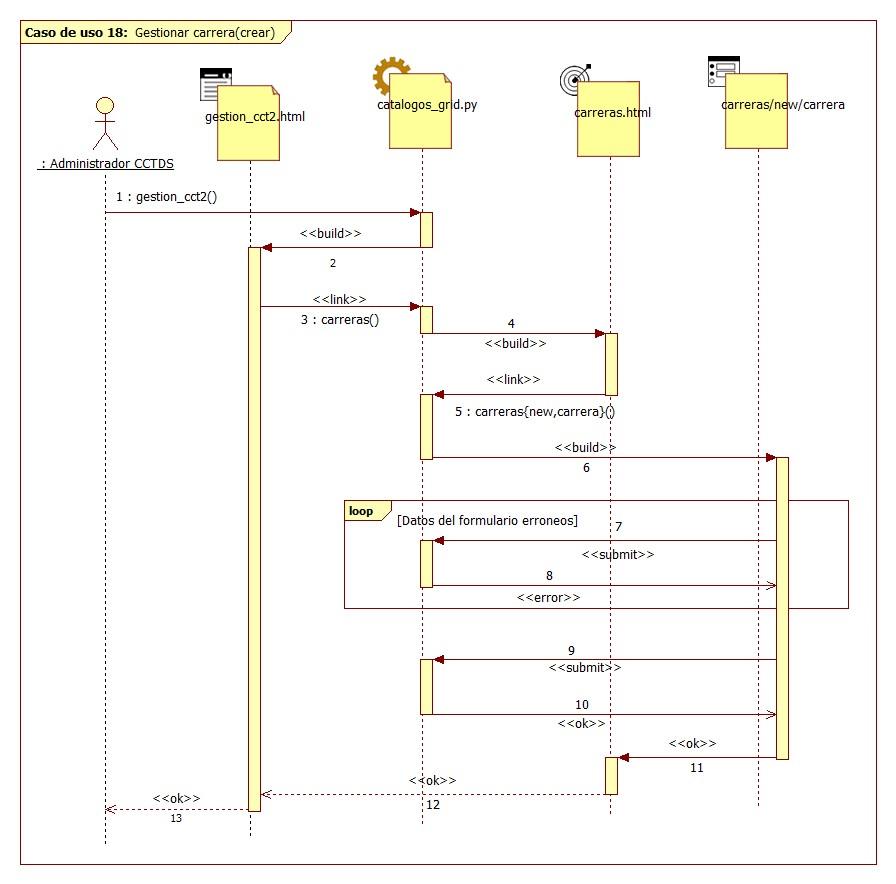
**5.3.3 Registrar Empresa**



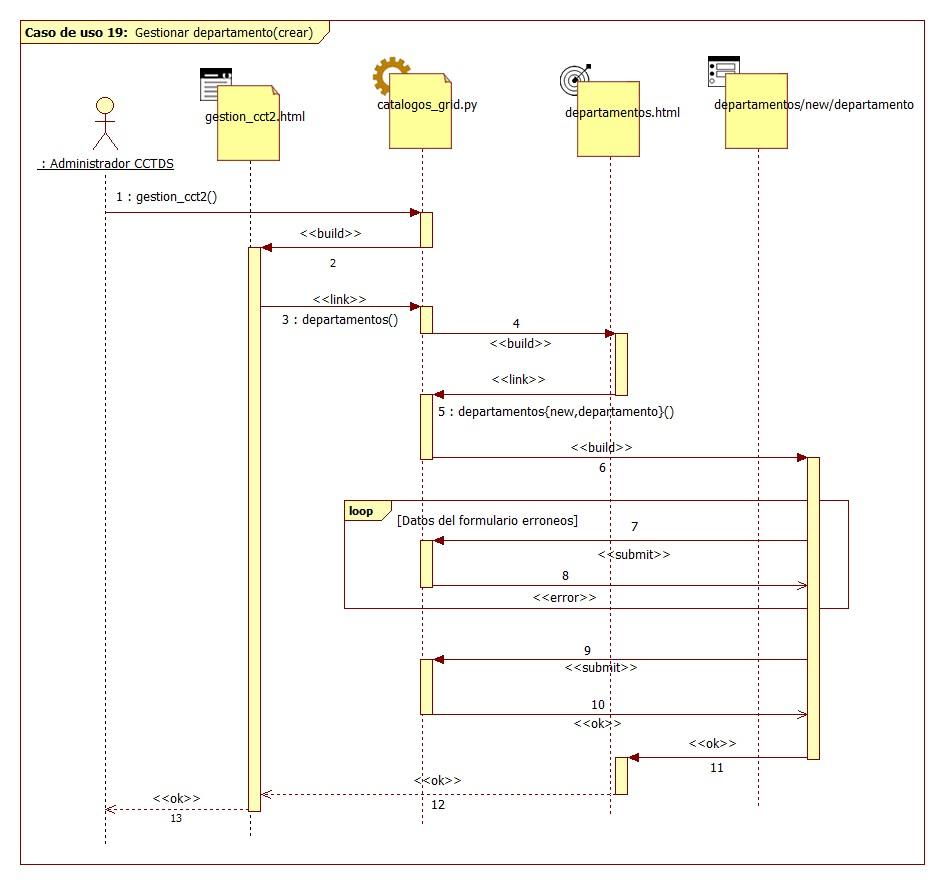
**5.3.4 Registrar Tutor Industrial**



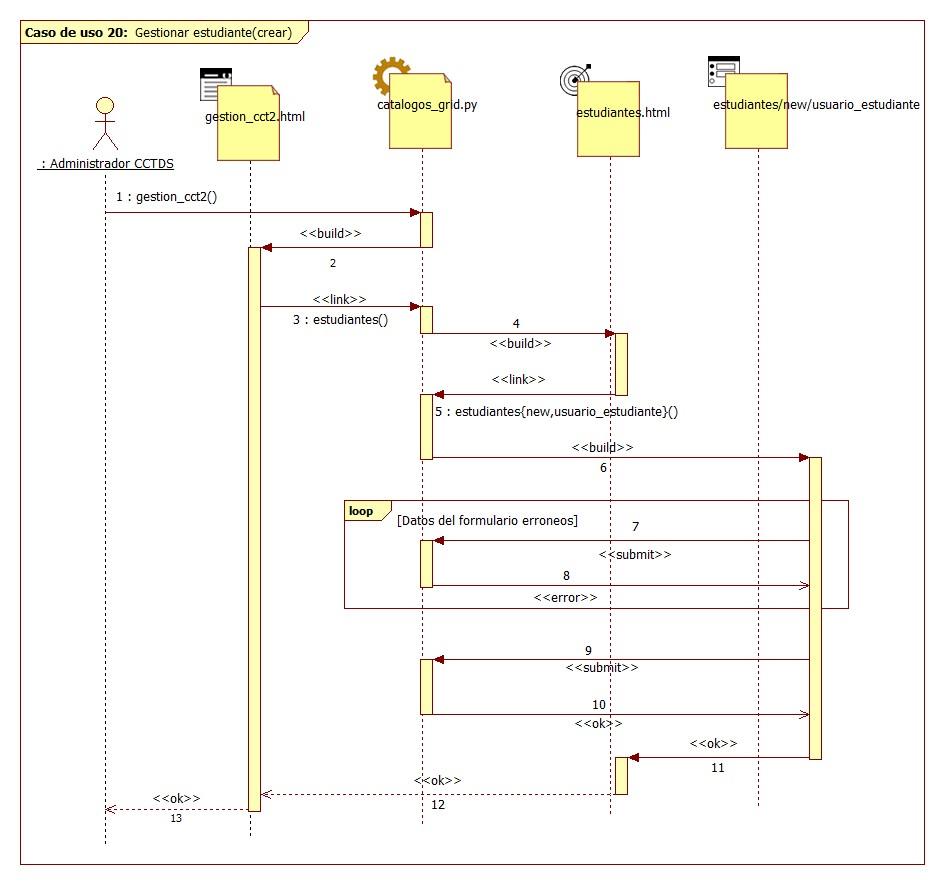
**5.3.5 Gestionar carrera(Agregar)**



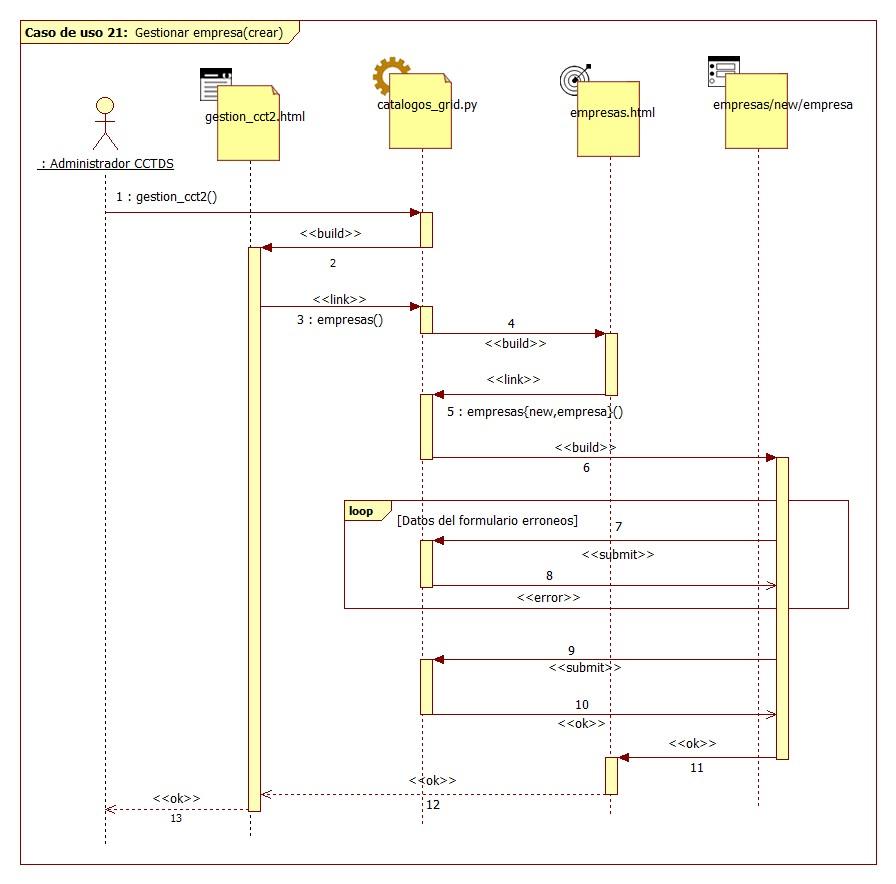
**5.3.6 Gestionar departamento(Agregar)**



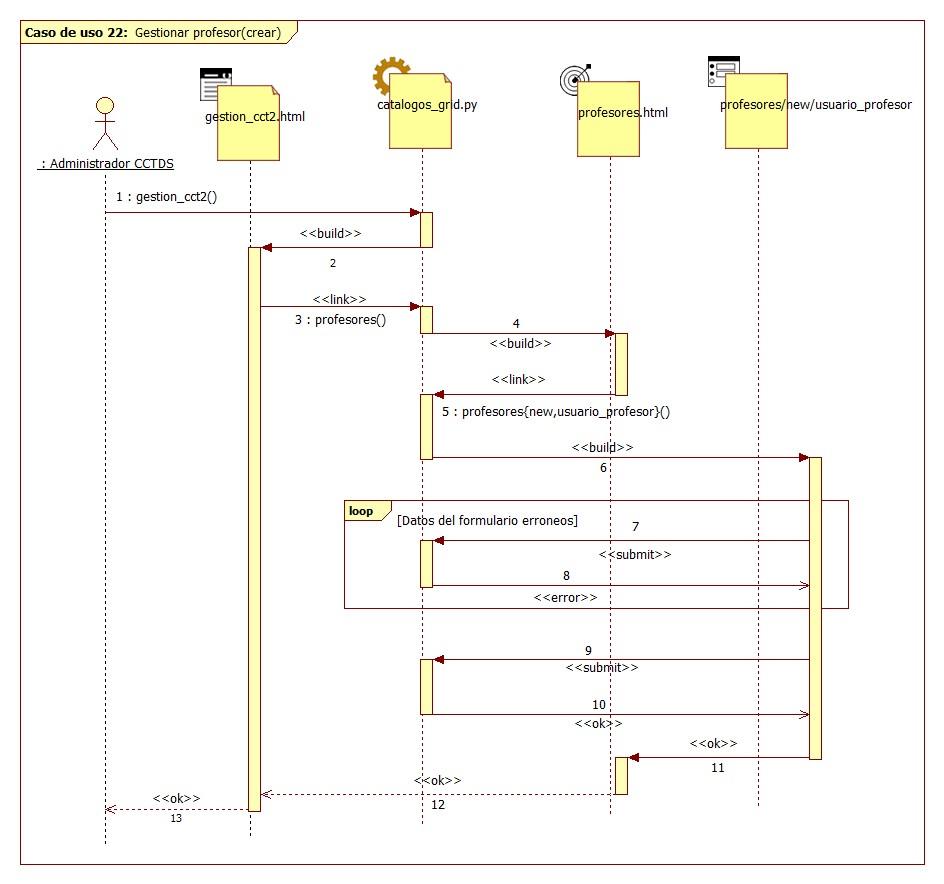
**5.3.7 Gestionar estudiante(Agregar)**



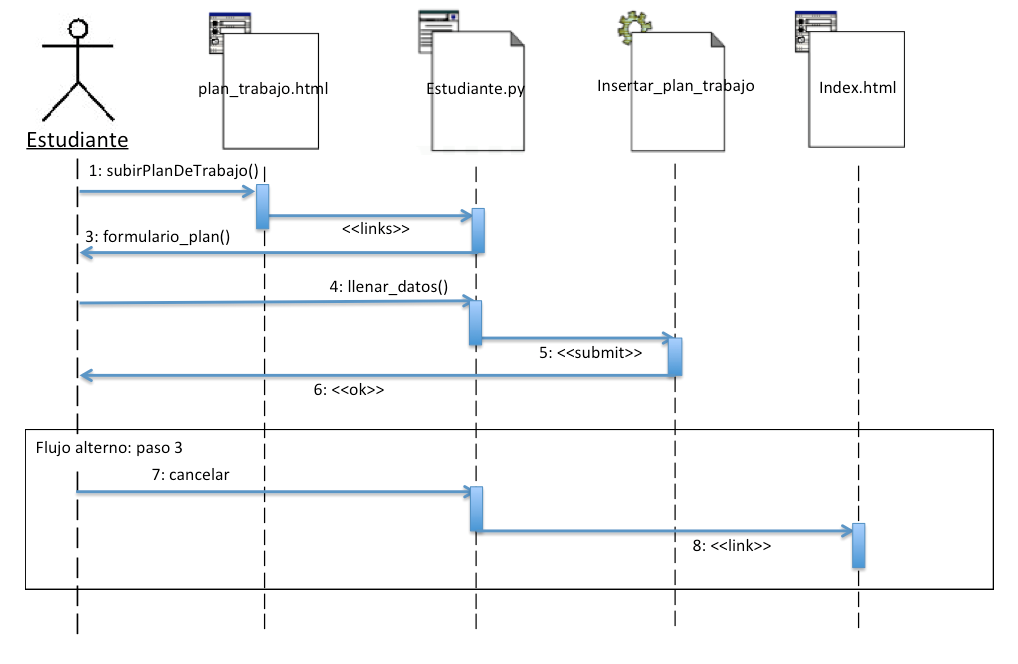
**5.3.8 Gestionar empresa(Agregar)**



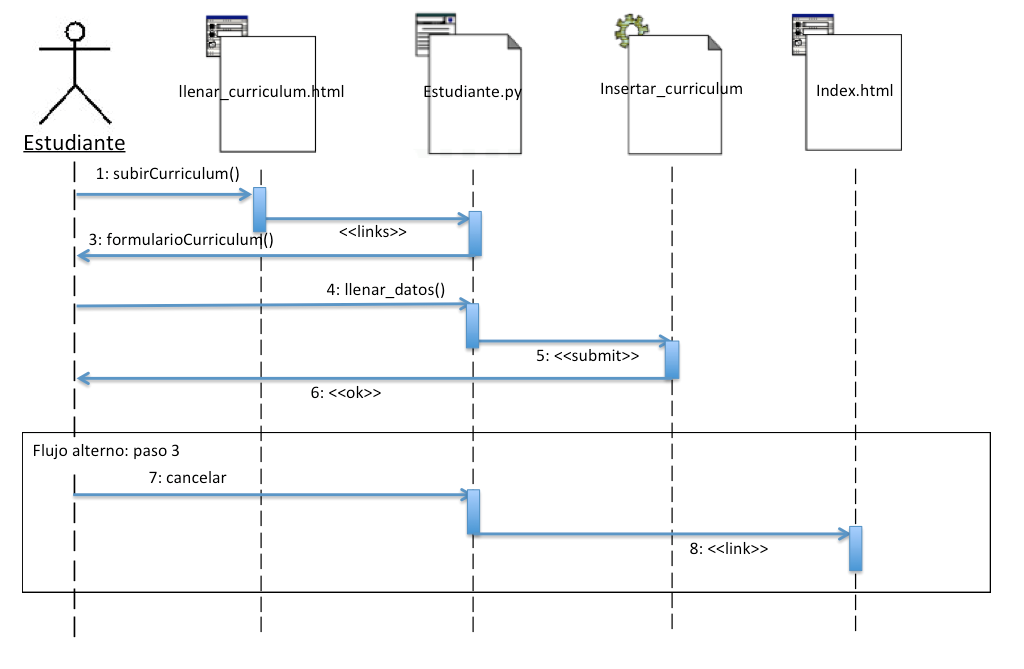
**5.3.9 Gestionar profesor(Agregar)**



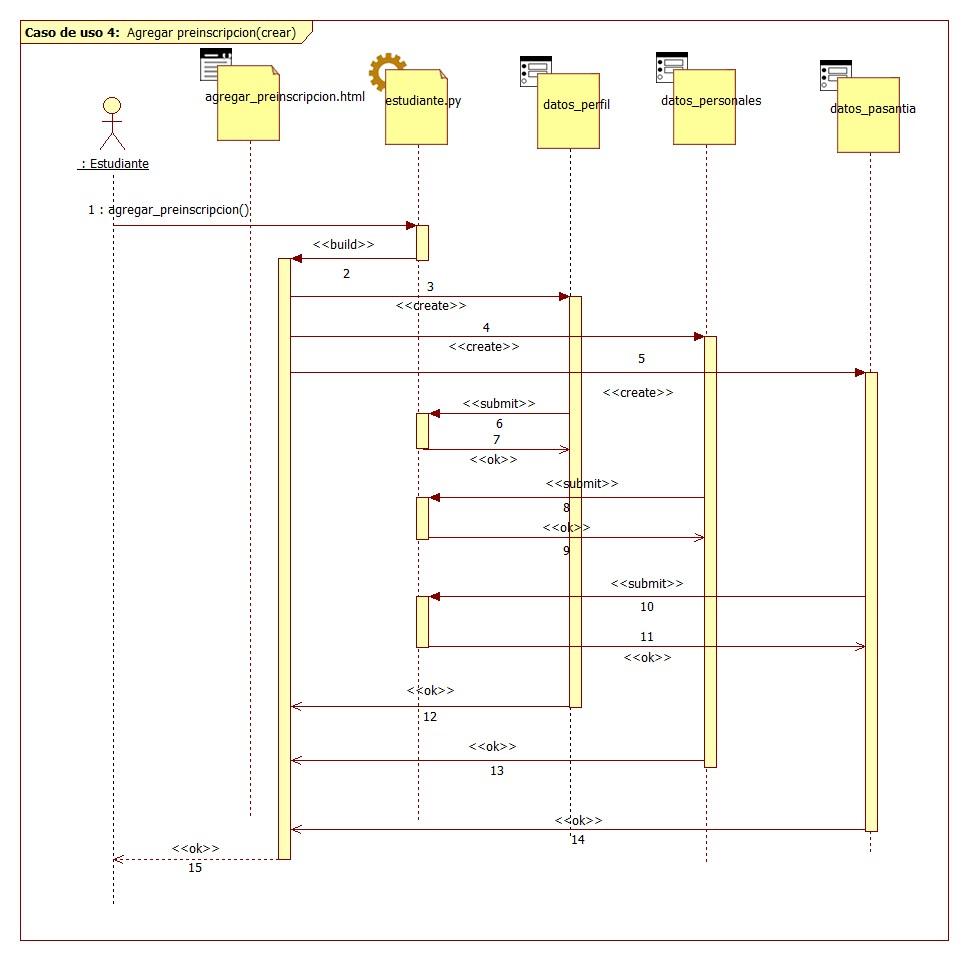
**5.3.10 Subir Plan de Trabajo**



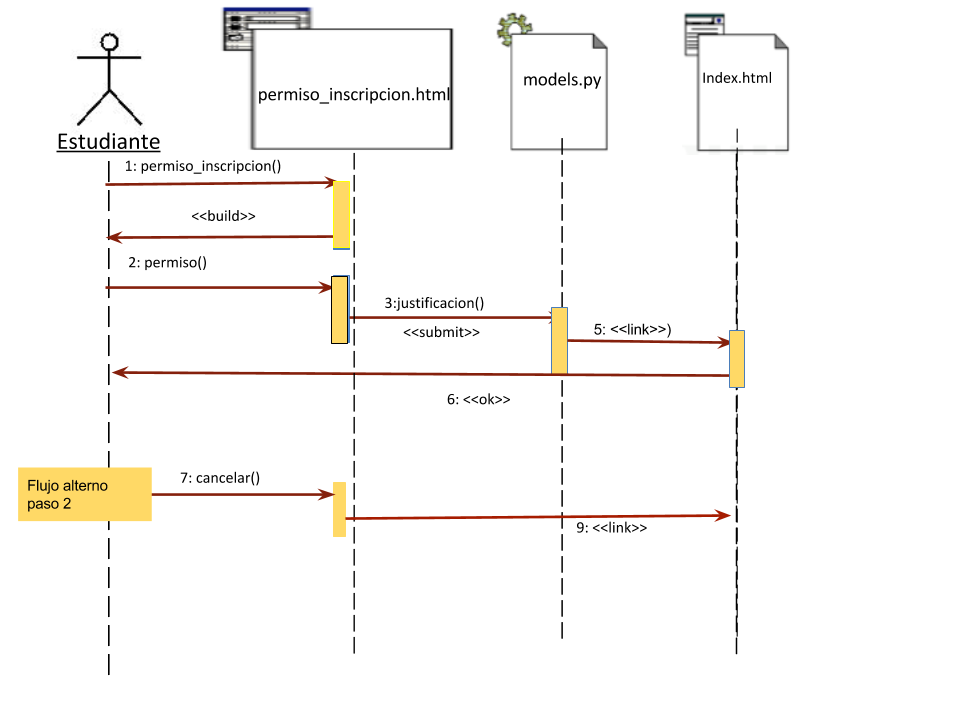
**5.3.11 Subir Curriculum**



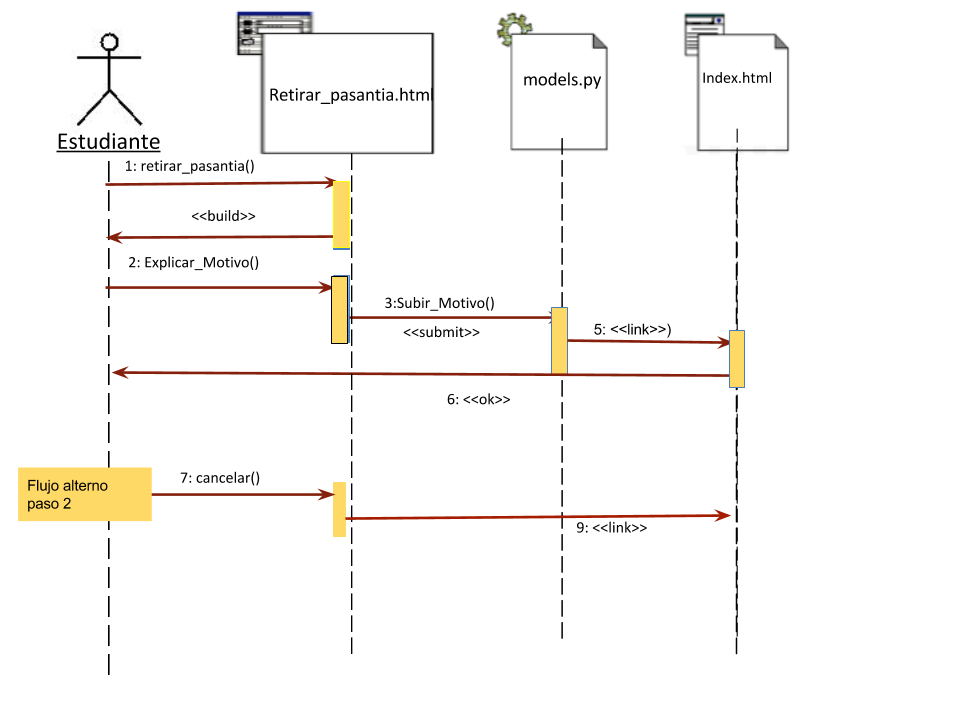
**5.3.12 Agregar preinscripción**



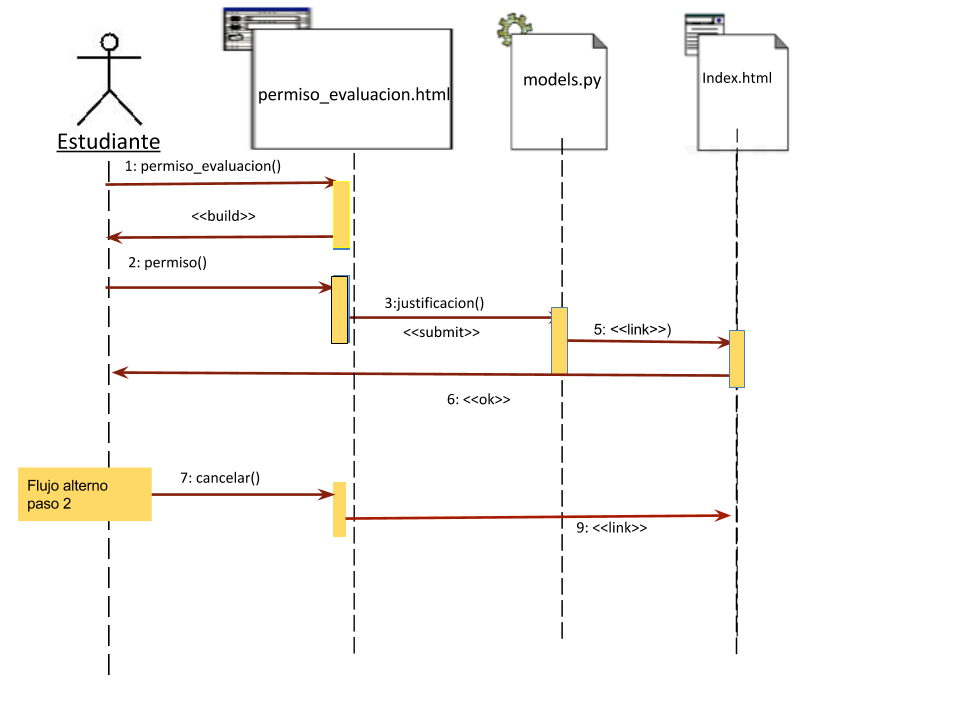
**5.3.13 Solicitar permiso inscripción extemporánea**



**5.3.14 Retirar pasantia**



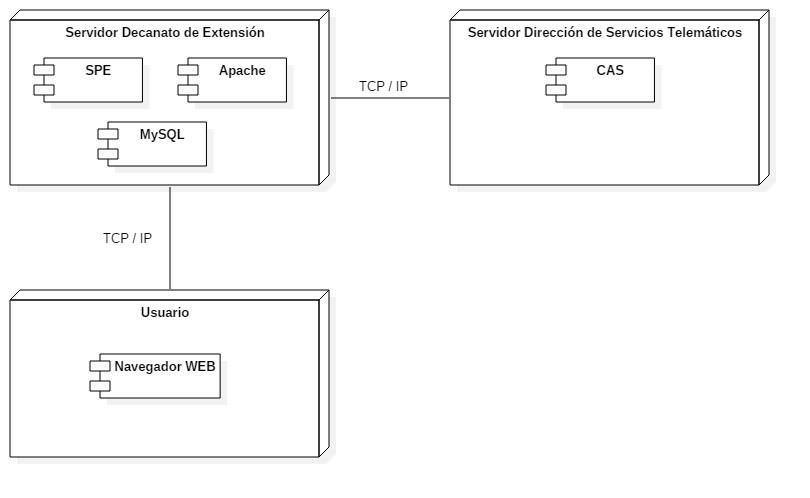
**5.3.15 Solicitar permiso de evaluación extemporánea**



1. **Vista de Procesos**

La vista de procesos es manejada por el servidor web automáticamente. Por esta razón, no se incluye en este documento.

1. **Vista de Implantación**



1. **Vista de Implementación** 
   1. **Vista General**

Se utiliza el patrón MVC para el diseño de arquitectura de software. Este es un patrón muy conocido y utilizado en paginas web.

Este patrón consta de tres capas. La primera capa es el Modelo

En la segunda capa tenemos la Vista. En esta están todos los objetos con los que el usuario podrá interactuar, además de definir la estructura y la interfaz que verá el usuario. En pocas palabras contiene todo lo referente a la parte gráfica del sistema.

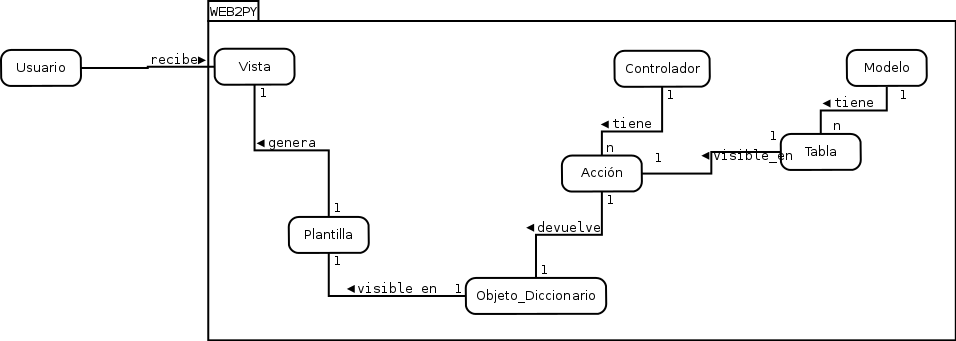
Por último está el Controlador el cual contiene todas las funciones que el sistema requiere para cumplir con sus funciones. En el Controlador se encuentra la lógica del programa.

* 1. **Capas**

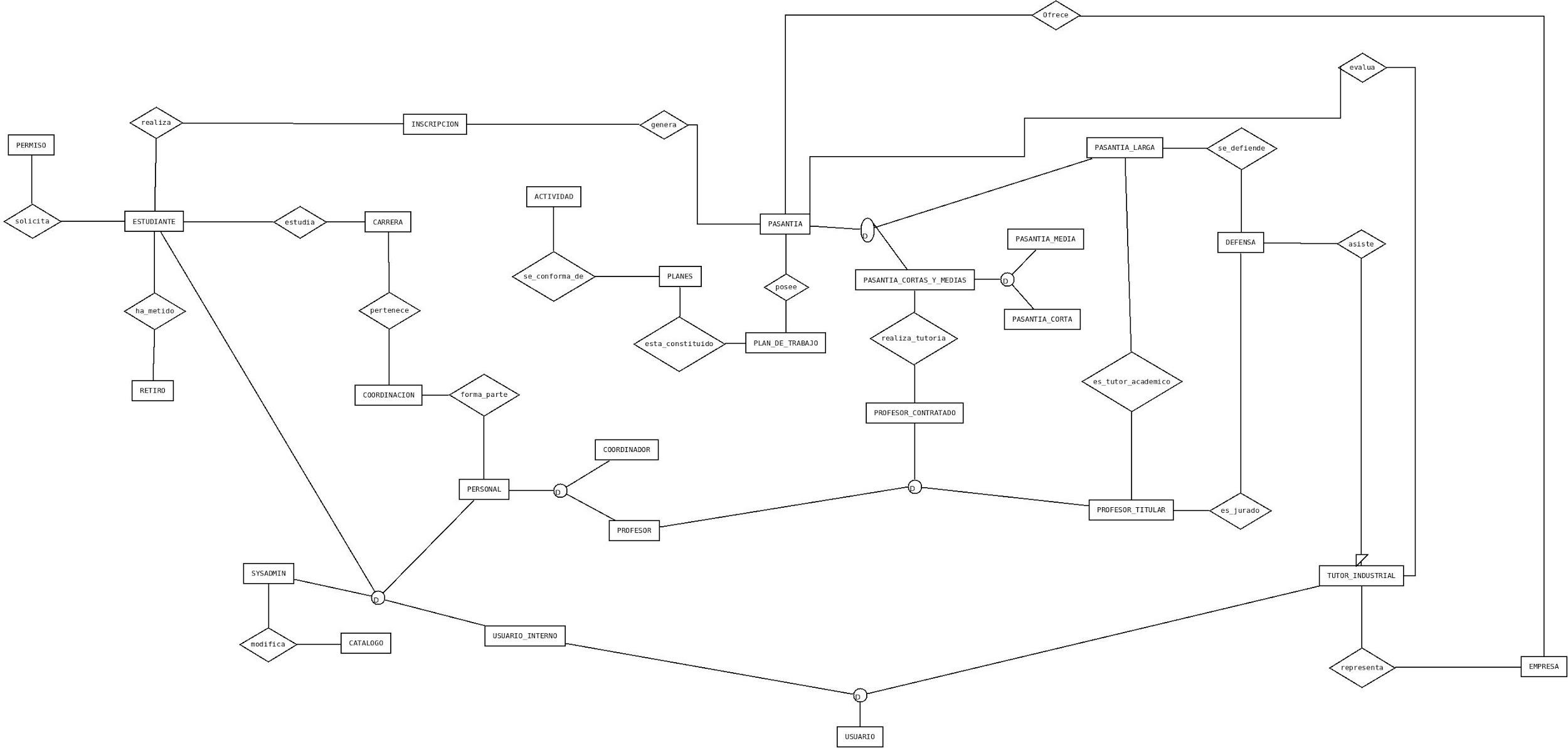
*Modelo***:** Esta es la sección del MVC que se refiere a los datos y a las operaciones sobre los mismos. Está constituido por un conjunto de funciones escritas en código python, éstas son procesadas por Web2Py que se encarga de comunicarse con el manejador de bases de datos MySQL y así crear tablas, ingresar datos, buscar en tablas, entre otras.

*Vista*: Esta es la sección del MVC que se refiere a la parte visual del sistema. Está constituido por templates html que son procesadas por Web2py, para luego ser interpretadas por el navegador y así terminar en la interfaz gráfica que ve el usuario.

*Controlador*: Es la última sección del MVC. Este es el contiene todos los procedimientos que son necesarios para realizar los cálculos y acciones que realiza el sistema, los cuales serán mostrados, de ser necesarios, al usuario por medio de la interfaz gráfica.



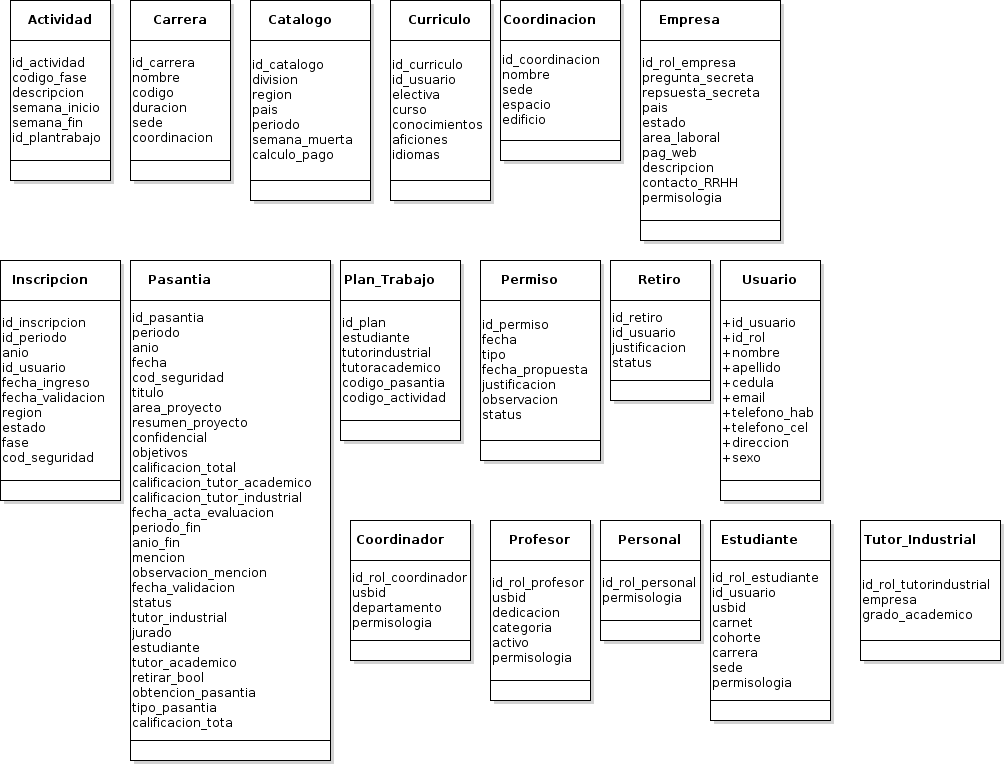
9**. Vista de Datos**

En la siguiente página se puede apreciar el modelo ERE.

**Restricciones explícitas**

* Solo puede haber una solicitud de plan de trabajo por estudiante bajo estatus 'Pendiente', 'Inscrito', 'Enviado'. Solo puede ocurrir que haya más de una en estado 'Retirado' y 'culminado' para los casos de pasantía corta y larga.
* Debe haber como máximo dos permisos de solicitud de evaluación extemporánea para un periodo, año, tipo de pasantía (que se corresponden con una pasantía) y usuario dado.
* Si la persona tiene mención tiene sentido en la base de datos el atributo observacion\_mencion. De lo contrario es nulo.
* Opciones status en tabla pasantía: ‘Pendiente’, ‘Enviado’, ‘Inscrito’, ‘Retirado’, ‘Culminado’, ‘Rechazado’, ‘Rezagado’.
* Opciones status Retiro tabla pasantía: ‘Pendiente’, ‘Solicitud’, ‘Aprobado’, Rechazado
* Opciones status Permiso: Pendiente, Aprobado, Rechazado
* Un tutor académico califica una pasantía, si y solo si esta es pasantía corta.
* El status de una pasantía larga pasa a ser *culminado* cuando el estudiante sube el acta de evaluación firmado.
* El status de una pasantía corta pasa a ser *culminado* cuando el tutor académico y el tutor industrial evalúan la pasantía.
* La carta de mención se puede generar cuando el status de la pasantía está *culminado* y el atributo *tiene\_mencion* está en 0.

**Diccionario de datos**



1. **Tamaño y Desempeño**

El sistema está diseñado para ser usado, principalmente, por personas pertenecientes a la Universidad Simón Bolívar y empresas en las cuales sus estudiantes hayan realizado pasantías. Por esto el tamaño del sistema es limitado. Así como la curva de crecimiento de usuarios es restringida, pues es imposible que en un trimestre, se inscriban más pasantías que el número de estudiantes pertenecientes a la USB.

1. **Calidad**

* *Portabilidad*: el sistema puede usarse en la mayoría de los navegadores web que pueden estar ejecutándose en distintos sistemas de operación.
* *Usabilidad*: el sistema puede ser usado por cualquier persona que tenga un manejo básico de Internet.
* *Mantenibilidad*: no se necesitan grandes esfuerzos para mantener el software.
* *Seguridad:* el sistema provee soporte para que se falsifiquen documentos y tiene autenticación robusta.