

Proyecto de Ingeniería de Software

Juego de Damas

Académico: Sebastián Salazar Molina. Ayudante: Patricio Pérez Valverde

06 de Abril de 2015

1 Motivación

La asignación de recursos es una actividad de gran importancia para las instituciones, nuestra casa de estudio no es ajena en este ámbito: material para laboratorios teórico-prácticos, uso de redes inalámbricas, espacio en servidores, salas de clases, etc.

El último punto es de principal interés, basta ver las secretarías de estudio los primeros días de clases, la información está centralizada en un lugar, donde el acceso a la información se dificulta debido a la alta demanda por parte del estudiantado. No hay forma de saber información extra sobre la asignación de las salas sin tener que visitar la secretaría de estudios, una pérdida innecesaria de tiempo. Si tan sólo alguien creara una forma de obtener esta información en nuestros gadgets ...

Objetivos

1.1 General

Desarrollar un sistema web que permita manejar integralmente la información sobre asignación de salas para la secretaría de estudios de cada campus.

1.2 Específicos

- Modelar el dominio del problema utilizando un paradigma orientado a objetos.
- Aplicar el patrón arquitectónico MVC sobre una aplicación del mundo real.
- Utilizar el framework Laravel para desarrollar un sistema web.
- Aplicar metodologías de desarrollo con un equipo de trabajo definido.

2 Requerimientos funcionales

La plataforma a desarrollar es de tipo web, los requerimientos de lado servidor son los siguientes:

- **Motor DB:** PostgreSQL 9.X

- **Framework PHP:** Laravel 5.0.x.

El ambiente de pruebas será:

- **SO:** Ubuntu Server 14.04 x64.
- **Motor DB:** PostgreSQL 9.3.
- **Framework PHP:** Laravel 5.0.x.
- **Lenguaje:** PHP 5.5.

Los requerimientos funcionales están enfocados en las funcionalidades que cada perfil de usuario puede realizar, detallandose en el siguiente punto.

3 Perfiles

3.1 Administrador

Este perfil esta encargado de la administración general del sistema, puede crear/modificar/archivar campus, asignar encargados a estos, asignar perfiles a usuarios existentes en la base de datos de la universidad (e.g: Promover a un estudiante a ser administrador o encargado de campus).

3.2 Encargado campus

Este perfil esta encargado de la realidad de las salas en su/sus campus asignados:

- Modificar aspectos como la capacidad y nombre de salas.
- Asignación de salas a un respectivo curso/evento en un periodo específico del calendario académico.
- Ingreso de datos académicos (e.g Cursos, asignaturas, estudiantes).

3.3 Estudiante

Este perfil es el principal beneficiado de la información contenida en el sistema, permitiendo consultar el horario de clases asignado a su persona, y sabiendo en que sala le corresponden tales clases. Tambien puede realizar consultas específicas, tales como ver las asignaciones de salas de un día específico en un campus X en un periodo Y.

3.4 Docente

Este perfil puede consultar en que periodos tenga que dictar cátedras/laboratorios, en función a la información ingresada por el encargado de campus. Tambien puede realizar las consultas extras mencionadas para el estudiante (Consultar por campus/periodos/salas/etc).

4 Requerimientos no funcionales

Los requerimientos no funcionales vertidos en este documento se enfocan sobre los ejes de la seguridad, usabilidad y mantenibilidad.

- Las operaciones realizadas por cada perfil descrito deben estar regidas por un control de acceso que evite acciones indeseadas para cada perfil (ej: Un estudiante no puede modificar la capacidad de una sala).
- La operación de carga de datos es en función al volumen de datos, por lo que el sistema es considerado inusable si es necesario ingresar 1000 asignaturas una a una.
- El uso de credenciales debe ser el más sencillo para los usuarios finales, como todos los usuarios pertenecen de alguna forma a la universidad ya podrán imaginarse como autenticarlos.
- El sistema puede ser usado por un largo periodo de tiempo, sin recibir atención del equipo de desarrollo a cargo, por lo que debe considerar los cambios básicos del calendario académico universitario.

5 Parámetros a evaluar

La evaluación considera los siguientes puntos:

- Uso de metodologías.
- Uso de herramientas (Framework/Control de versiones).
- Completitud de los requerimientos funcionales.
- Generación de la documentación pertinente.

6 Proyecto

- El lenguaje de programación debe ser C o C++. El compilador en que se revisará será gcc (aconsejo usar MinGW en Windows).
- El trabajo debe ser realizado por los grupos de 3 personas definidos.
- La fecha de entrega es hasta el 03 de Julio de 2015 a las 23:55 (hora continental de Chile), deben enviarme un correo electrónico con la dirección de su repositorio GitHub.
- Cualquier duda, la pueden expresar en el grupo de Facebook o a mi correo.