

# Informe académico

Aylín Álvarez Santos C-312  
Grettel Hernandez Garbey C-311  
Carlos Toledo Silva C-311

## Índice

<b>1. Introducción</b>	<b>2</b>
<b>2. Análisis de requerimientos</b>	<b>2</b>
2.1. Requerimientos funcionales . . . . .	2
2.2. Requerimientos informacionales . . . . .	3
<b>3. Solución propuesta</b>	<b>4</b>
3.1. Arquitectura . . . . .	4
3.2. Descripción de los módulos . . . . .	4
<b>4. Resultados obtenidos</b>	<b>6</b>
<b>5. Conclusiones y recomendaciones</b>	<b>6</b>
<b>6. Bibliografía</b>	<b>6</b>

# 1. Introducción

La alquimia es un juego muy sencillo que se hizo muy popular en los últimos años. El juego consiste en generar elementos nuevos a partir de elementos ya conocidos. Cada vez que se genera un elemento nuevo, mezclando dos elementos conocidos de forma correcta, este pasa a ser un elemento conocido. El objetivo de un usuario es el de formar todos los elementos admisibles o al menos tratar de acercarse a dicha cantidad. Para que el juego fuera interesante, no toda pareja de elementos genera un elemento nuevo. Dado esto, el usuario deberá razonar que combinación de elementos puede generar un elemento nuevo.

El objetivo que persigue este proyecto es utilizar la lógica de la alquimia de una manera didáctica para medir la capacidad de aprendizaje y razonamiento de los estudiantes en una determinada asignatura. De esta forma, los elementos serán conceptos y como muchas veces un concepto depende de otros conceptos, la unión de dos conceptos generaría un nuevo concepto. El enfoque de esta idea es que los estudiantes partan de un conjunto de elementos básicos y puedan mezclarlos y otros que vayan creando para tratar de crear la mayor cantidad de elementos posibles en una determinada asignatura.

## 2. Análisis de requerimientos

### 2.1. Requerimientos funcionales

La aplicación permite el registro de 3 tipos de usuarios: estudiante, profesor y administrador, además del rol de superusuario.

El superusuario es un usuario que tiene acceso a todas las funcionalidades de la aplicación, además que puede añadir, agregar o eliminar a otros usuarios.

El administrador es un usuario que puede ser creado por un superusuario u otro administrador previamente creado. Este además es el encargado de crear, editar y eliminar asignaturas de la aplicación, además de que es el que puede agregar y eliminar profesores tanto del sistema en general como de una asignatura en específico. Además puede eliminar estudiantes tanto del sistema en general como de una asignatura en específico.

El profesor es un usuario que puede ser creado por un superusuario o un administrador. Además a los profesores también se les asignan las asignaturas que imparten, por parte de los dos usuarios anteriores. Este solo puede acceder a las asignaturas que imparte. Además el profesor es capaz de aceptar o rechazar las solicitudes de mezclas enviadas por los estudiantes matriculados en una asignatura que imparte. Además este tiene la opción de crear elementos básicos de una asignatura, así como poder editar y eliminar elementos básicos de una asignatura siempre que no estén marcados como visibles. Cuando un elemento se marca como visible este ya no puede ser modificado.

Un estudiante es un usuario que entra al sistema y se registra rellenando un formulario con su información personal. Los estudiantes pueden matricularse en las diferentes asignaturas disponibles. En cada asignatura, un estudiante puede hacer solicitudes de mezcla utilizando los elementos que ya posee, es decir los básicos de la asignatura y los que haya creado él mismo a partir de mezclas aceptadas por algún profesor de la asignatura; además debe proporcionar una justificación de porqué él considera posible hacer esa mezcla. Los elementos mezclados obtienen valor igual a la suma de los valores de los dos elementos que lo conformaron. Por ejemplo si  $e_1 + e_2 = e_3$  y  $v(e_1) = 10000$  y  $v(e_2) = 10000$ . entonces  $v(e_3) = 20000$ . La puntuación de un estudiante es la suma de todos los elementos que ha formado y la puntuación de un estudiante en una asignatura es la suma de todos los elementos formados en la asignatura (obviamente que hayan sido aceptados).

## 2.2. Requerimientos informacionales

Al acceder a la aplicación los usuarios pueden observar, en la página principal, una breve descripción de como funciona la aplicación.

Al loguearse un administrador, este es capaz de visualizar todos los usuarios inscritos en el sistema, divididos según su rol. Al acceder a la página de las asignaturas este puede ver todas las asignaturas en el sistema y al acceder a una en específico puede visualizar los profesores que imparten la asignatura, así como los estudiantes matriculados en la misma.

Al loguearse un profesor este tiene la opción de ver a los administradores registrados en el sistema. Esto por si necesitara contactar con algún administrador para reportar algún problema. Al acceder a la sección de las asignaturas este puede ver solo las asignaturas que imparte. Accediendo a alguna asignatura en específico puede ver las solicitudes de mezclas hechas por los estudiantes matriculados en la asignatura. Además en la página de los elementos básicos es capaz de ver todos los elementos básicos creados por cualquier profesor de la asignatura.

Al loguearse un estudiante este puede acceder a la página de las asignaturas y dentro de esta página puede ver todas asignaturas y matricularse en las que desee. Al acceder a la página de una asignatura puede ver todos los puntos(créditos) alcanzados en todas las asignaturas y de forma independiente en la asignatura de la página en la que se encuentra. Además puede ver a los profesores que imparten dicha asignatura. Además tiene tres opciones que le permiten ver los elementos básicos de la asignatura, los elementos que le han sido aceptados por algún profesor de la asignatura y los elementos que están pendientes de aceptación o rechazo.

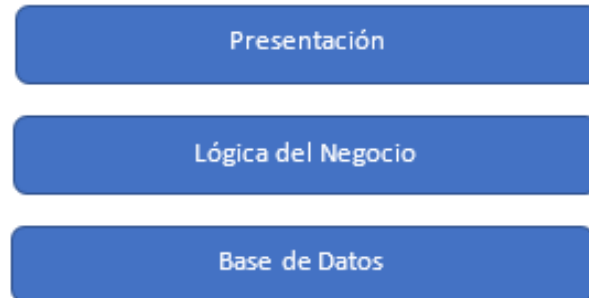
Además cada uno de los usuarios(estén loguados o no) pueden acceder a la página de consultas, donde pueden realizar las siguientes:

- Los  $n$  estudiantes con más creditos (ordenados de mayor a menor).
- Los  $n$  estudiantes con más elementos creados (ordenados de mayor a menor).
- Los  $n$  elementos más utilizados para realizar mezclas por asignaturas (ordenados de mayor a menor).
- Los  $n$  elementos más valiosos de los elementos no básicos por asignaturas (ordenados de mayor a menor).
- Los  $n$  elementos más valiosos de los elementos básicos por asignaturas (ordenados de mayor a menor).
- Los elementos creados en un año.
- Los elementos creados en un intervalo de fechas.
- Ranking de estudiantes por asignatura.
- Los  $n$  estudiantes del ranking de estudiantes por asignatura.
- El mejor estudiante por asignatura.
- Ranking de estudiantes en un intervalo de fechas.
- Ranking de estudiantes en un intervalo de fechas por asignatura.

### 3. Solución propuesta

#### 3.1. Arquitectura

La arquitectura a emplear en el software será la arquitectura cliente-servidor junto a una arquitectura por capas, o sea, la arquitectura propuesta es dividir todo el software en capas, en particular tres:



Donde la capa de presentación será con la que se relacionarán los clientes del software mientras que las capas restantes serán las relacionadas con el servidor. Cada una de estas capas será cerrada, por tanto, cuando un cliente haga una consulta de los datos la consulta debe descender por las capas donde pueden ser procesadas o sencillamente delegar la consulta a la capa inferior.

#### Descripción de las capas. Funciones.

**Capa de Presentación:** como se mencionó anteriormente, esta capa desarrolla sus procesos en los dispositivos del cliente, en la arquitectura cliente-servidor. En ella ha de mostrarse todas las estadísticas requeridas en el sistema, así como el resto de requerimientos funcionales e informacionales. Además, a través de esta capa el cliente hará las peticiones al servidor (hará consultas y modificaciones a la base de datos).

**Capa de Lógica del Negocio:** esta capa desarrolla todo su procesamiento en el servidor. En ella se ha de validar usuarios utilizando informaciones provenientes de la base de datos, y otros archivos. Además, de convertir las consultas provenientes de la capa de presentación en consultas de MySQL. En la capa, además se recibirán las respuestas provenientes de la base de datos, sean tablas o errores, y reportará los mismos a la capa de Presentación. En esta capa de lógica se ejecuta todo el procesamiento de datos y consultas entre la capa superior e inferior.

**Capa de Base de Datos:** esta capa desarrolla todo su procesamiento en el servidor. En ella se implementa la base de datos conceptualizada, modelada y normalizada hasta lograr su correctitud en el informe anterior.

#### 3.2. Descripción de los módulos

El proyecto está conformado por las siguientes aplicaciones:

- admins:** Esta aplicación contempla el registro de los administradores, así como los permisos que se le otorgan a este en su creación, además del listado y eliminación de los mismos. También esta aplicación contempla la funcionalidad de ver los profesores y estudiantes de una asignatura.
- basic\_elements:** Esta aplicación contempla el listado, creación, edición y eliminación de los elementos básicos, así como la funcionalidad de hacerlos visibles a los estudiantes.
- elements:** Esta aplicación simplemente contiene el modelo `Element`, del cual heredan los modelos `BasicElement` y `NonBasicElement`. Este contiene los atributos comunes a ambos tipos de elementos.
- imparts:** Esta aplicación contempla la relación de impartir una asignatura por parte de un profesor. Esta contiene las funcionalidades de hacer que un profesor  $i$  imparta la asignatura  $j$ , eliminar esta relación y obtener las solicitudes de mezclas hechas por los estudiantes.
- main:** Aplicación principal. Muestra la página principal del sitio con algunas informaciones correspondientes al sitio
- non\_basic\_element:** Esta aplicación contempla lo relacionado a los elementos no básicos, o sea los que crean los estudiantes. Más concretamente contempla las funcionalidades de listar los elementos aceptados y pendientes, la creación y eliminación de un elemento por parte de los estudiantes y la aceptación o rechazo de un elemento por parte de un profesor.
- professors:** Esta aplicación contempla el listado, registro y eliminación de los profesores por parte de los administradores.
- queries:** Esta aplicación contempla un conjunto de clases que gestionan las consultas: desde la selección de la consulta, la entrada de los parámetros, el procesamiento y la posterior muestra de los resultados de la consulta.
- students** Esta aplicación contempla el listado, registro y eliminación de los estudiantes.
- study:** Esta aplicación contempla la relación de estudiar/cursar una asignatura por parte de un estudiante. Esta contempla las funcionalidades de matricularse y mostrar la página de la asignatura para un estudiante; y eliminar la relación de estudiar por parte de algún administrador.
- subjects:** Esta aplicación contempla el listado de las asignaturas, según sea el usuario que accede a la página de las asignaturas (pues como hemos dicho no todos los usuarios visualizan las mismas asignaturas); la creación, edición y eliminación de las asignaturas por parte de los administradores; y que cuando se trate de acceder a la página de una asignatura por parte de algún usuario, según sea el tipo del usuario, se acceda a la página correcta.
- users:** Esta aplicación contempla la edición del usuario logueado en el sistema. Se sitúa en una aplicación aparte pues se hace de la misma forma para todos los usuarios.

Todos estos módulos fueron implementados por Carlos, excepto el módulo queries que fue implementado por Grettel. Para el frontend de la aplicación se utilizó el propio framework Django. Se agregaron estilos a las vistas para un manejo agradable de la aplicación, de lo cual se encargó Aylín

## 4. Resultados obtenidos

El software obtenido es una aplicación web, la cual cumple con los requerimientos funcionales e informacionales abordados en secciones anteriores.

Todas las páginas de la aplicación muestran una barra superior, en la cual aparece el nombre del sitio y las diferentes páginas que se pueden visitar.

En la página de inicio se muestra la explicación del funcionamiento de la aplicación.

Un usuario que no esté registrado puede acceder a la página de Consultas, donde puede ver estadísticas relacionadas con los estudiantes. También podrá registrarse en el sistema e iniciar su sesión.

Un usuario logueado, además de las opciones mencionadas anteriormente, tiene un perfil, ya sea de estudiante, profesor o administrador, y además cuenta con la opción de poder editar su perfil.

Un usuario estudiante logueado en la aplicación, también tiene acceso a la página Asignaturas. En esta se muestran las asignaturas disponibles, en las cuales los estudiantes se pueden matricular, y así acceder a las opciones que se brindan en cada una de ellas: crear un nuevo elemento, ver los elementos básicos disponibles de la asignatura, ver los elementos que le han aceptado hasta ese instante y los elementos pendientes a revisar por los profesores de la asignatura.

Un usuario profesor logueado en la aplicación, también tiene acceso a la página Asignaturas, pero con opciones distintas a las opciones del usuario estudiante: aceptar o denegar las solicitudes de los estudiantes, ver elementos básicos de la asignatura, así como añadir, modificar, eliminar o hacer visible un elemento básico. Además tiene acceso a la página Administradores, en la cual se muestran todos los administradores de la aplicación.

Los administradores de la aplicación, como se mencionó anteriormente son usuarios que pueden crear, modificar y eliminar cualquier información de la base de datos. Este tipo de usuario puede ser creado por un superusuario u otro administrador previamente creado. Una vez logueado, tendrá opciones que le permitirán crear, editar y eliminar usuarios y asignaturas.

## 5. Conclusiones y recomendaciones

Por lo anteriormente expuesto, consideramos que esta aplicación podría resultar de gran utilidad para evaluar el desempeño de los estudiantes en las asignaturas.

La aplicación presenta una interfaz bastante intuitiva y agradable a la vista para los usuarios.

En un futuro, se le podría agregar la opción de crear un nuevo elemento, realizando la mezcla con más de dos elementos, por lo que ayudaría a los estudiantes a obtener una mayor cantidad de créditos, y por ende mejorar su rendimiento en la asignatura.

## 6. Bibliografía

1. Documentación de Django 3.2: <https://docs.djangoproject.com/en/3.2/>. Consultado el 25 de mayo de 2021.
2. Canal de youtube: Developer.pe <https://www.youtube.com/channel/UCOzf2U3BTng85gsFJHvIInQ>. Consultado el 10 de mayo de 2021.
3. Canal de youtube: Neunapp <https://www.youtube.com/channel/UCOzf2U3BTng85gsFJHvIInQ>. Consultado el 15 de mayo de 2021.