

Trabajo Práctico N°8

Simulación

“Autogestión de Alumnos”

“End-To-End y Web Testing”

Grupo N°1

UTN FRRO 2020

Integrantes:

- Octavio Berruti
- Giovanni Martin

Legajo: 45332

Legajo: 44912

Autogestión de Alumnos

Abstract:

El objetivo de nuestro proyecto es realizar una pagina web que le sea de gran utilidad a los estudiantes para que se puedan organizar de una manera mas eficiente y no se descuiden en los compromisos que deban realizar. Además, queremos proporcionar un espacio de almacenamiento a los datos que tengan los alumnos acerca de cada asignatura, tarea o parcial.

Introducción:

Testing es un proceso, un método de trabajo, una herramienta, que se utiliza para identificar fallas en el software o producto con el fin de que este alcance un grado aceptable de madurez y estabilidad antes de ser lanzado.

End-to-end Testing:

- Prueba todo el flujo del software desde el punto de vista del usuario final.
- Prueba el software desde la interfaz de usuario y no desde el código interno.
- Está enfocado en detectar posibles problemas que pudieran encontrar nuestros usuarios en su interacción con el flujo general del programa.

Web Testing:

- Prueba de funcionalidad
- Pruebas de usabilidad
- Prueba de interfaz
- Pruebas de compatibilidad
- Pruebas de rendimiento
- Pruebas de seguridad

Narrativa:

Nosotros vamos a realizar un sistema web de autogestión de calificaciones para alumnos. Consiste en un sitio donde cada alumno se deberá registrar, registrar las asignaturas que está cursando con los profesores que dictan en dicha comisión y el sistema generará un archivo pdf con el horario semanal del alumno. Una vez registrado el alumno podrá ingresar tareas o parciales de las distintas asignaturas

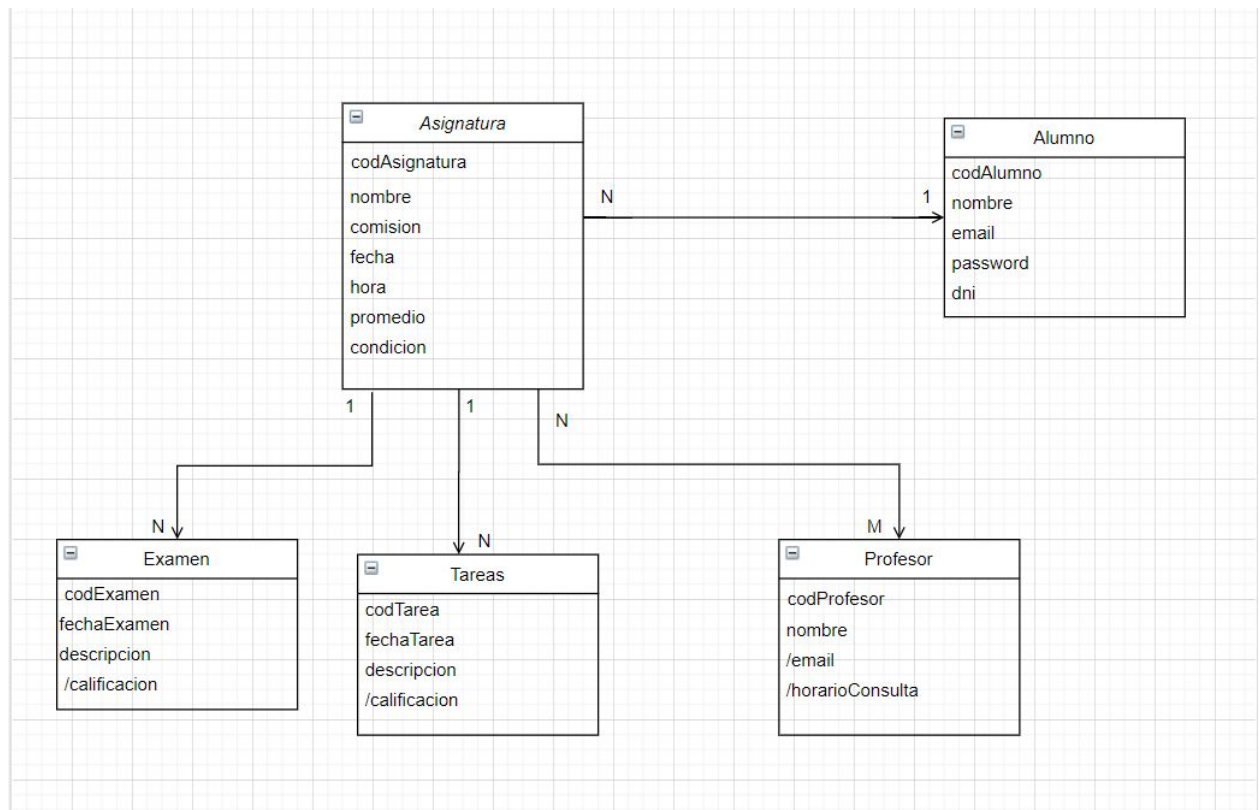
ingresando la fecha de entrega y el sistema los ordenara y le irá mostrando al alumno cuales son sus próximos compromisos, las anotaciones que el alumno realizó del mismo (ya sean temas, notas o lo que desee) y una vez finalizado el parcial o la tarea podrá ingresar una nota que luego el sistema calculará el promedio y la tendrá disponible para que el alumno la consulte.

A esta página web vamos a realizarle End-To-End Testing y Web testing para poder optimizar su rendimiento y para garantizarnos que funcione como la imaginamos.

Condiciones a cumplir:

1. Programación en Python.
2. Utilizar la arquitectura de capas.
3. Bases de datos MySql.
4. Conexión a la Base de datos con pymysql y ORMAlchemy.
5. Interfaz web utilizando HTML, CSS.
6. Conexión del back end y front end con Flask.

Modelo de Domino:



Casos de uso:

CUU01 - Registrarse en el sistema

Descripción: Alumno se registra en el sistema

Actores: Alumno, sistema

Precondiciones: --

Flujo normal:

1. El alumno ingresa a la página web.
2. El alumno ingresa sus datos.
3. Sistema comprueba la validez de los datos y registra.
4. Alumno ingresa las asignaturas y los datos correspondientes.

Flujo alternativo:

3.a.1 El sistema comprueba la validez de los datos, si los datos no son correctos o ya están registrados comunica al alumno.

3.a.2 Alumno verifica los datos y los ingresa nuevamente en caso de error.

Postcondiciones: Alumno se ha registrado exitosamente

CUU02 - Ingresar en el sistema

Descripción: Alumno ingresa en el sistema

Actores: Alumno, sistema

Precondiciones: Alumno ya registrado en el sistema

Flujo normal:

1. El alumno ingresa a la página web.
2. El alumno ingresa sus datos.
3. Sistema comprueba la validez de los datos y accede a la pagina web principal.

Flujo alternativo:

- 3.a.1 El sistema comprueba la validez de los datos, si los datos no son correctos lo comunica al alumno.
- 3.a.2 Alumno verifica los datos y los ingresa nuevamente en caso de error.

Postcondiciones: Alumno ha ingresado exitosamente

CUU03 - Registrar examen

Descripción: Alumno registra nuevo examen

Actores: Alumno, sistema

Precondiciones:

Alumno ingresó con su usuario

Flujo normal:

1. Alumno ingresa nuevo examen, asigna una materia y sus anotaciones.
2. Sistema registra nuevo examen

Postcondiciones: Alumno ha ingresado exitosamente

CUU04 - Registrar tarea

Descripción: Alumno registra nueva tarea

Actores: Alumno, sistema

Precondiciones:

Alumno ingresó con su usuario

Flujo normal:

1. Alumno ingresa nueva tarea, asigna una materia y sus anotaciones.
2. Sistema registra nueva tarea.

Postcondiciones: Tarea registrada correctamente