

Tarea 3

Prototipo de Implementación

Integrantes: Pedro Belmonte
Ricardo Cordova
Israel Peña
Joaquín Pérez
Profesora: Jocelyn Simmonds
Auxiliar: Constanza Escobar
Ayudantes: Sergio Leiva
Pablo Miranda
Santiago, Chile

1. Introducción

En la tercera iteración del trabajo realizado, se deben desarrollar las interfaces diseñadas para el sistema, por el grupo seleccionado en la tarea anterior.

Las interfaces solicitadas corresponden al landing page para personas naturales, el landing page para administradores, el perfil del usuario (visto por el dueño del perfil) y la ficha de un artículo. Para cada uno de ellos, se utiliza el diseño realizado por Soft-of-War (según lo votado por el curso). En este informe, se presenta las metodologías utilizadas por el equipo de Soft-Where para realizar estas implementaciones, el modelo de datos utilizado y su relación con las interfaces gráficas implementadas; finalmente, se presenta como las interfaces implementadas satisfacen los requisitos funcionales establecidos para esta entrega. Se concluye este informe con reflexiones sobre el trabajo y la metodología utilizada.

2. Metodologías de Equipo

El equipo de trabajo utilizó una metodología de trabajo tipo Scrum: esto es, con trabajo diario de avance en "features", divididos para los distintos miembros del equipo, necesarios para la implementación de los requisitos, con revisión constante de los logros conseguidos.

Sin embargo, hay ciertas limitantes que no permiten realizar un trabajo Scrum a cabalidad: no es posible hacer reuniones diarias dada la distancia entre los miembros, así como los horarios de la universidad, además de que no se dispone de los espacios necesarios como para poder llevar una tabla de avances como lo requiere esta metodología de trabajo. Herramientas como Git y Git Flow ayudan a suplir algunas de estas falencias, ya que permiten ir revisando los avances que realizan los otros miembros del equipo y los "features" sobre los que trabajan.

El desarrollo de las implementaciones se realiza utilizando el Framework de Django, que permite levantar el funcionamiento de un servicio web rápidamente con backend basado en Python y frontend basado en html y sintaxis de DTL que separa el funcionamiento del diseño, permitiendo que cada parte pueda trabajar de forma independiente sin tener que repetir información de uno en otro. Otros elementos que se añaden para mejorar el trabajo desarrollado son Bootstrap, para tener un diseño de web más responsivo, y Pillow, para facilitar el manejo de imágenes en la base de datos. Dada esta facilidad de separar las partes, y las restricciones a las que nos enfrentamos, se decide que los miembros del equipo deberían trabajar en las áreas que consideren son las más fuertes para ellos mismos. De esta forma, se aprovechan las habilidades y los intereses de cada uno de los miembros del equipo.

Por ejemplo, Ricardo trabajó principalmente en Backend, en lo referido al modelo de datos y el sistema de login y creación de usuarios; mientras que Pedro trabajó principalmente en el Frontend, pues conocía el funcionamiento de Bootstrap y tiene buenas nociones de diseño. De esta forma, cada uno de los miembros del equipo pudo aportar lo mejor de cada uno.

Miembro Equipo	Participación
Pedro Belmonte	• • • •
Ricardo Cordova	• • • •
Israel Peña	• • • •
Joaquín Pérez	• • •

Gracias a esta metodología de trabajo, se pudieron implementar los requerimientos indicados por lo solicitado en las instrucciones de la tarea n° 3: los requisitos 1-2, 4-5, 7-8, 12, 14-18, 23, 25-26, 28, 37-40, 51 y 54-56. En la sección 4 de este informe se especifica como se satisfacen estos requisitos mediante las interfaces implementadas. En la sección 3 se observara el modelo de datos que subyace a las interfaces implementadas, y como estas se relacionan con los modelos para poder implementar sin tener que re-escribir código.

3. Modelo de Datos

El modelo de datos es el siguiente:

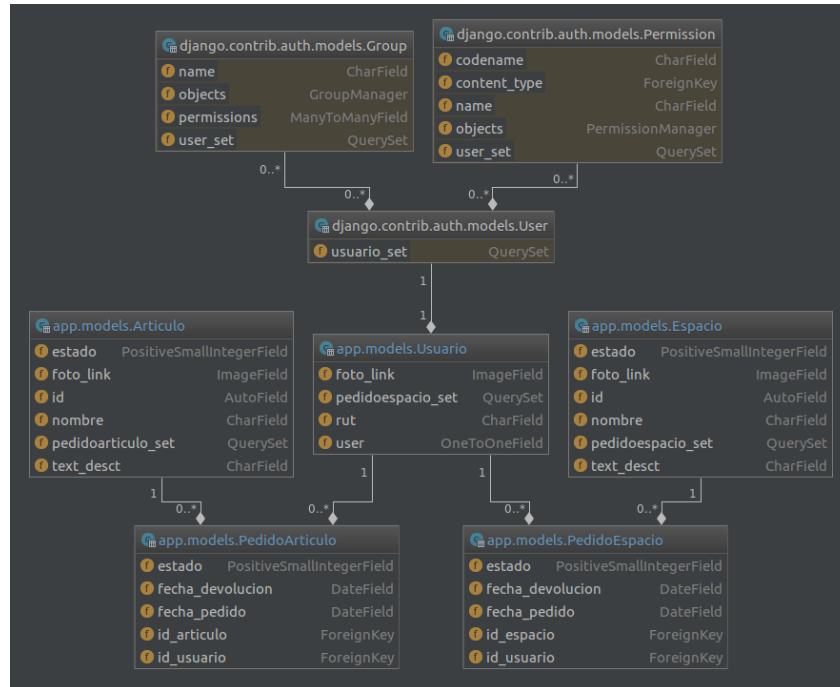


Figura 1: Diagrama del modelo de datos implementado

En este modelo se pueden distinguir tres niveles: un primer nivel generado automáticamente por Django en cuanto se utilizan sus modelos de Groups, Permission y User; un segundo nivel creado por el equipo donde se encuentran los elementos directamente relacionados con los requisitos que son los Artículos, Usuarios y Espacios; el último nivel es creado también por el equipo, pero son tablas relacionales entre los usuarios y artículos y espacios que representan los pedidos realizados por los usuarios y los resultados históricos de estas solicitudes. Los modelos del primer nivel permiten generar un sistema de creación de usuario y login con facilidad. Estos modelos disponen de nombre, apellido, correo, clave, permisos y grupos, todo ello por defecto dentro de Django. Así, podemos diferenciar una Persona Natural de un Administrador utilizando grupos, dando permisos por separado a cada grupo según lo indicado en los requisitos, así como poder guardar la información requerida por los usuarios.

Sin embargo, se hace necesario complementar esta información con datos adicionales que no corresponden al modelo de Django. Para ello, en el segundo nivel del modelo se extiende el usuario por defecto de Django con un usuario personalizado con nueva información adicional, como el rut y la foto de perfil; esta información servirá para poblar de contenido el perfil del usuario. Además, consideramos tablas adicionales para los artículos y los espacios que dispondrá el sistema para prestar a sus usuarios. La información que ahí se almacena servirá para poder poblar las fichas de los artículos y espacios.

El último nivel, que posee las tablas relacionales entre usuario y artículo, y usuario y espacio; estas tablas almacenan la información de los pedidos hechos por un usuario, la fecha de pedido y

devolución. Esta información se puede usar por el perfil del usuario como un registro histórico de sus pedidos, además de poder indicar al sistema que elementos se encuentran disponibles para que otros usuarios puedan pedir o no un artículo (según las fechas de los pedidos), así como informar a los administradores sobre la vigencia o caducidad de una solicitud (entre otros posibles).

4. Cumplimiento Requisitos

5. Conclusiones

con