Design document

Dimitar Prisadnikov
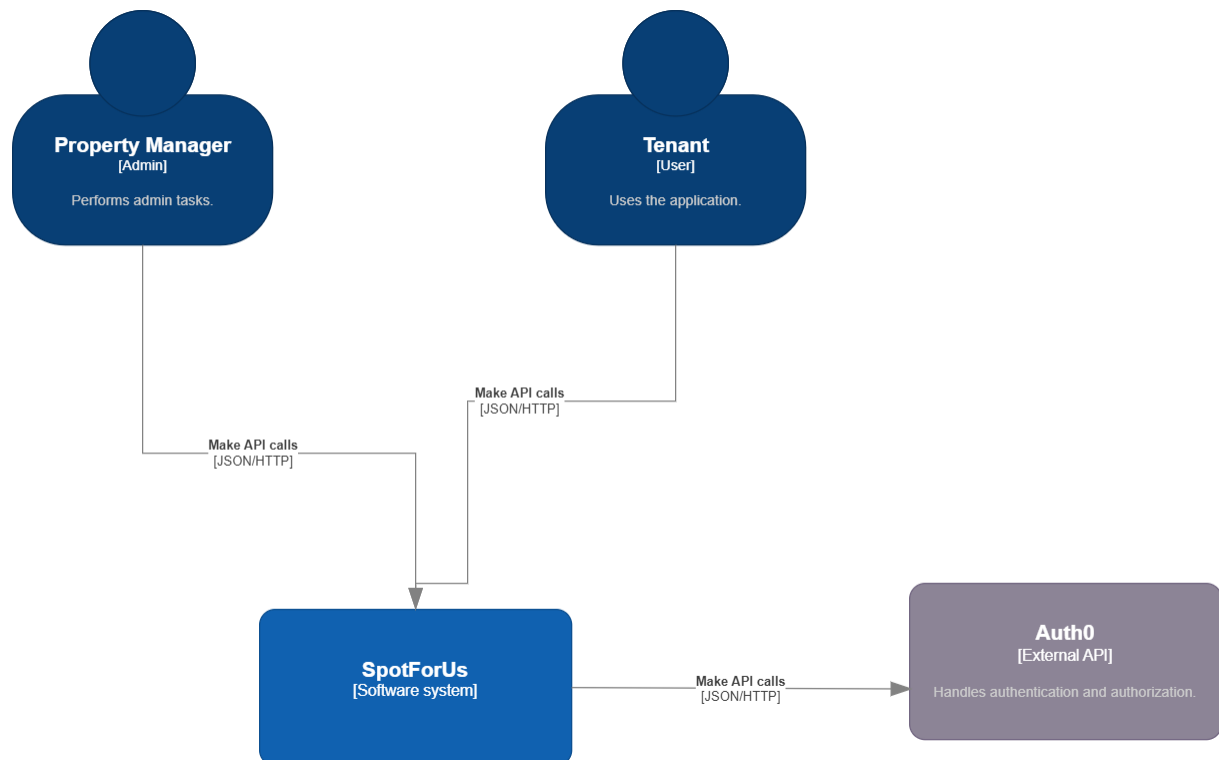
Contents

C4 architecture

C1

The system is a website with back-end in Java Spring boot and front-end in React.js. The two personas that will be using it are Tenant (user) and Property Manager (admin). The first one is regular user who can get on the website and use its features while the admin is someone who manages the content (including other users) but cannot make use of the same features as the user since their purpose is just to manage content and not consume it.

C2

The front-end communicates with the back-end by making API calls to the controllers of the back-end. The sent and retrieved data is formatted in JSON. The back-end access the data stored in the MySQL database and make changes according to the requests made by the React client.
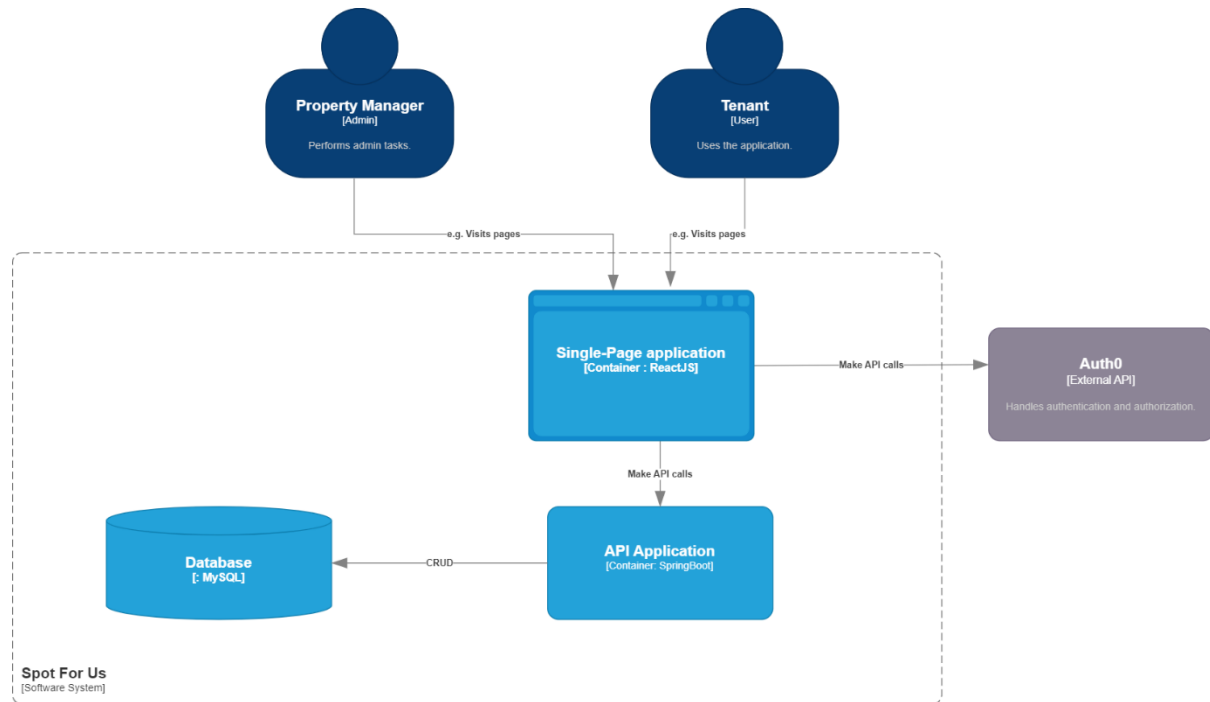
C3

On the backend the data is separated in a few key categories – controllers, models, services, repositories, and the relevant interfaces for all of them to insure proper dependency injection.

**Controllers** – handles incoming API request from the client.

Services – logic layer where all the calculations and changes happen.

Repository – used for CRUD-ing the MySQL database. For most of the cases it extends JpaRepsitory.

Entities – domain objects used for creating and managing the database tables. Used by all layers but the controllers.

Dtos – data transfer objects that are used by the controllers to send/retrieve only the essential data without exposing the entities.
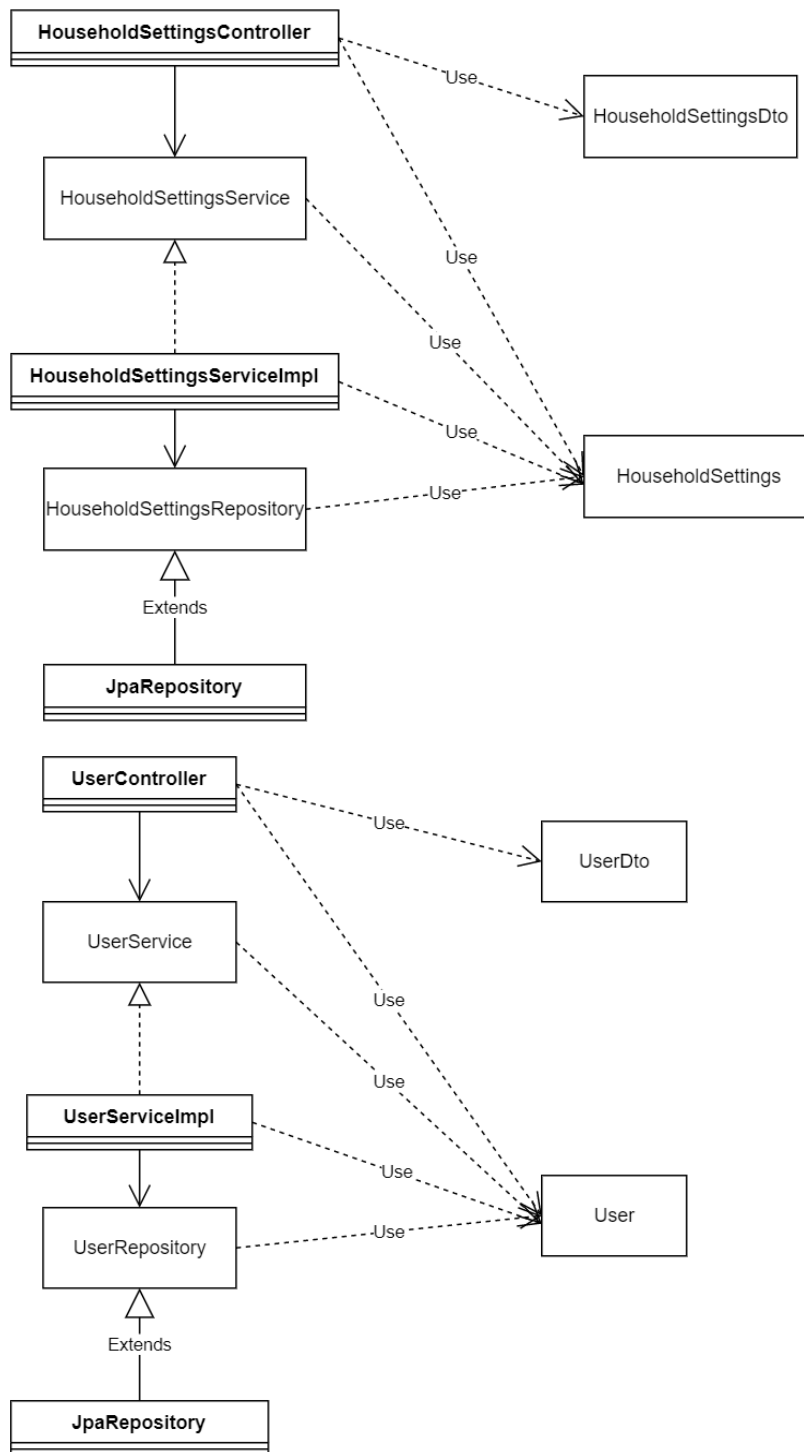
ModelMapper – used by controllers to perform DTO-Entity conversion and vice versa.
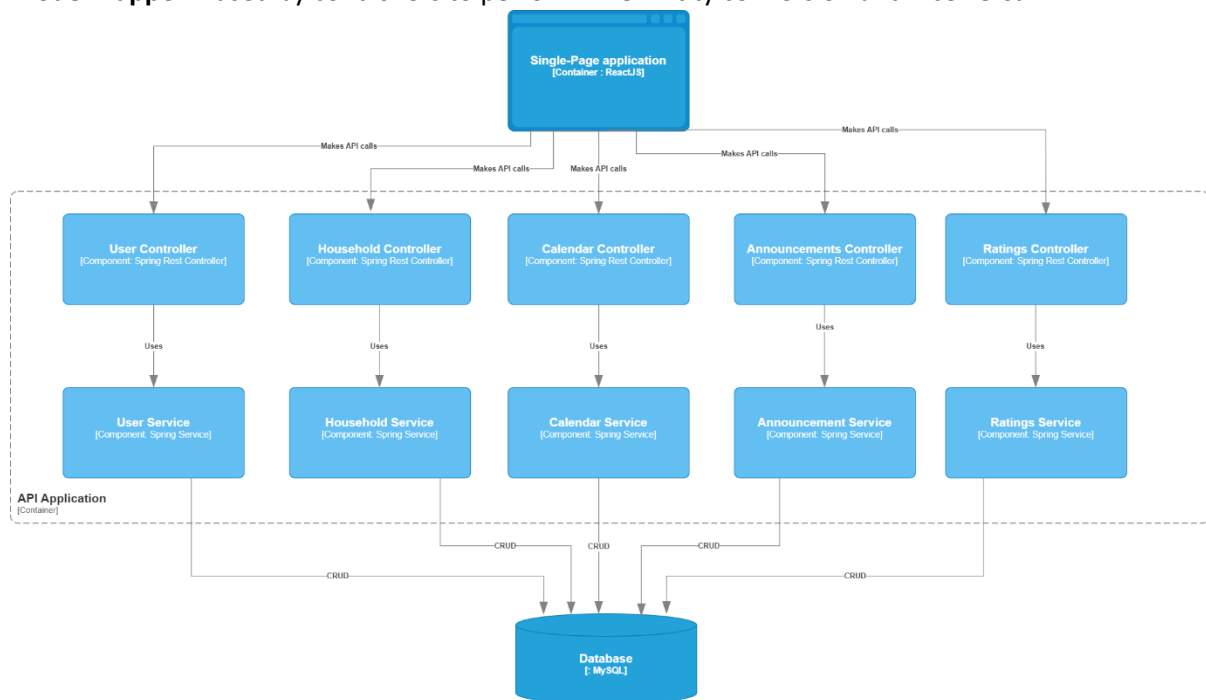
C4

This level is used to show in detail how the code works and prove the presence of Dependency Inversion. UML class diagrams are present for the major parts of the back-end application.
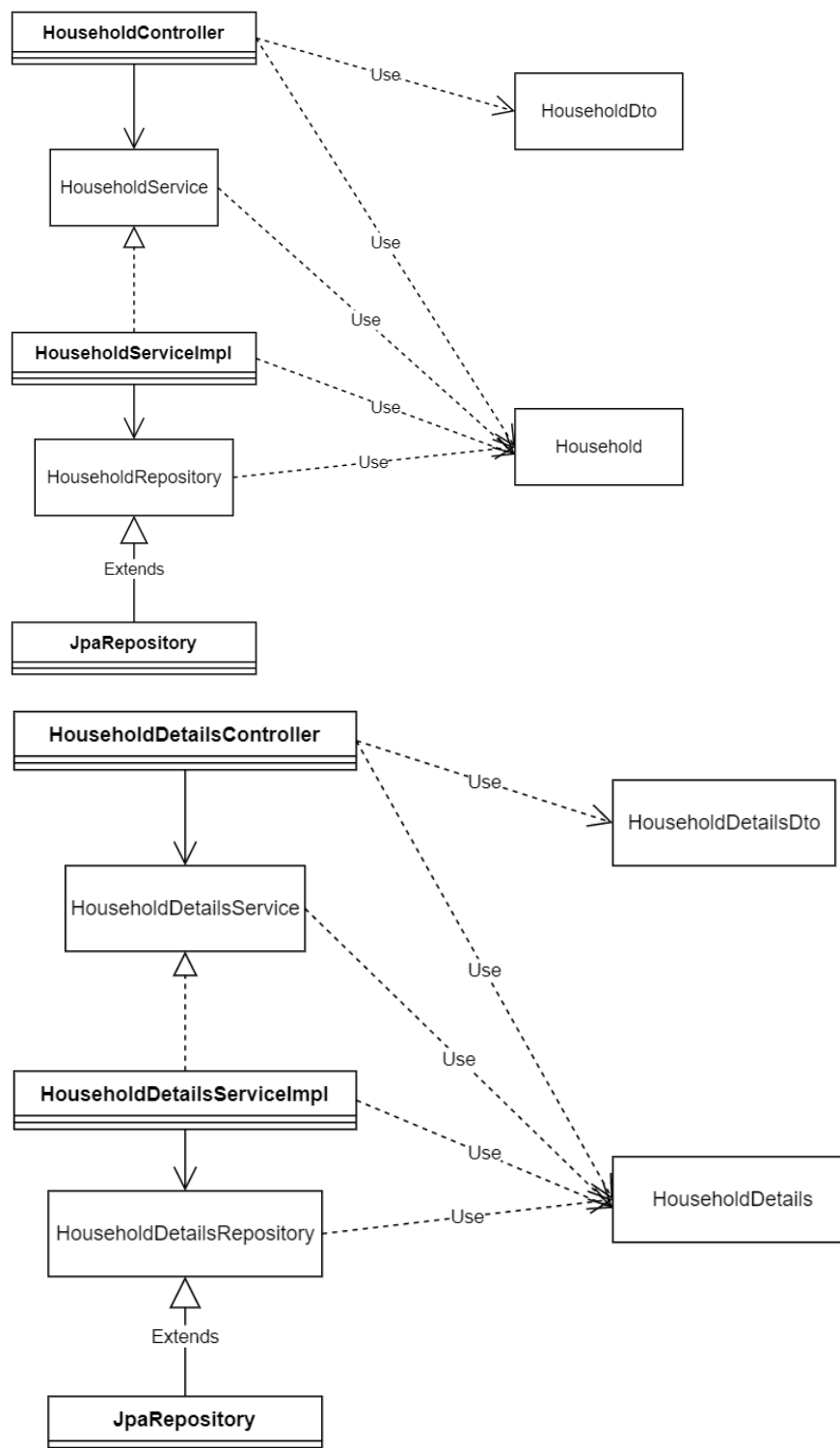
Database

The current database for the project is MySQL running in a locally stored Docker container. MySQL is a relational database management system (RDBMS) developed by Oracle that is based on structured query language (SQL). As for accessing the database as a developer, MySQL Workbench is used. Both Spring Data JPA and Spring JDBC can be used when it comes to accessing data stored in the MySQL database from the Spring Boot application.