

SOA Project

Purpose of the project

The purpose of the project is to allow the users to search any book any ISBN and to save the desired books in their own list. They can also delete book from their book list.

Structure

This system contains two components: Book API and Book portal. Book API provides CRUD operations for books and users and authenticates users. Book portal is a web application – a user interface for Book API. See below the component diagram.

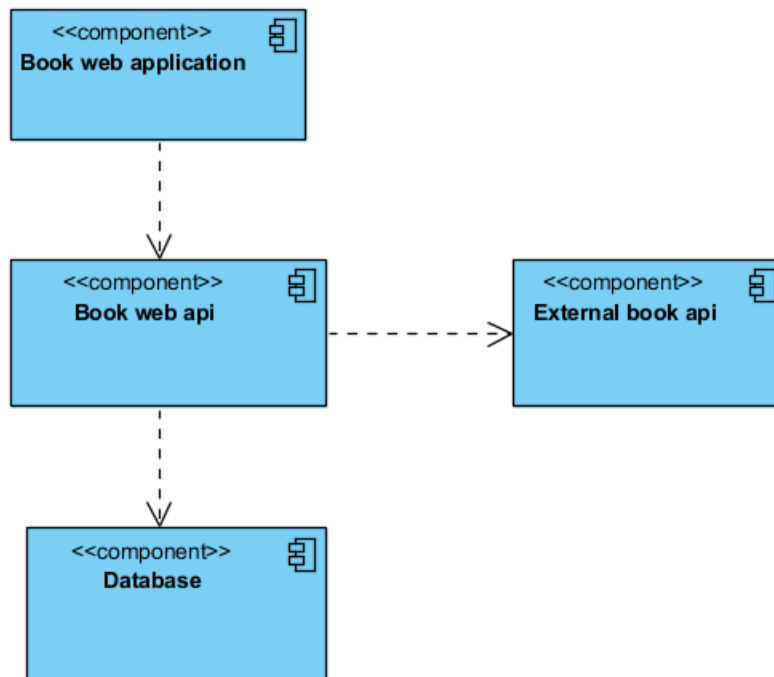


Figure 1: Component diagram

SOA Patterns Used

Authentication Broker

The authentication broker can authenticate clients, such as users or services, and grant them a certificate or a token. They will use further this token to access other services.

Service Façade

A service façade sits between a service and a contract. Its logic allows the service contract to remain decoupled from the underlying logic and further shields it from changes to the core business logic.

Message Translator

Different applications typically use different data types. Therefore, for two applications to successfully communicate, we should translate the messages (that pass from one application) to the data type compatible with the receiving application. In this case, a translator changes the context of a message from one interface to another. This ensures that the message adheres to the context rules of the back-end service.

Diagrams

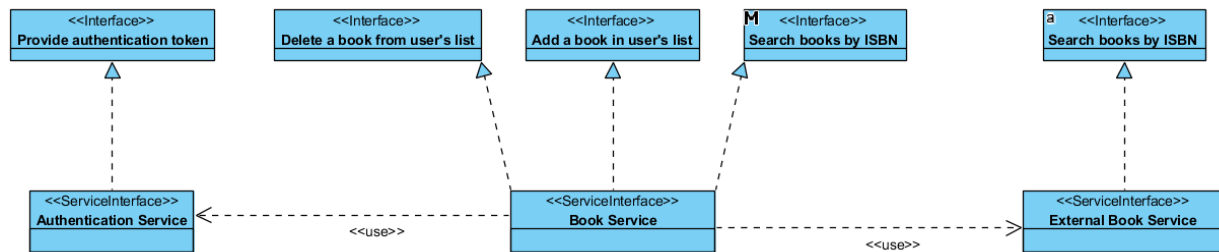


Figure 2: Service interface diagram

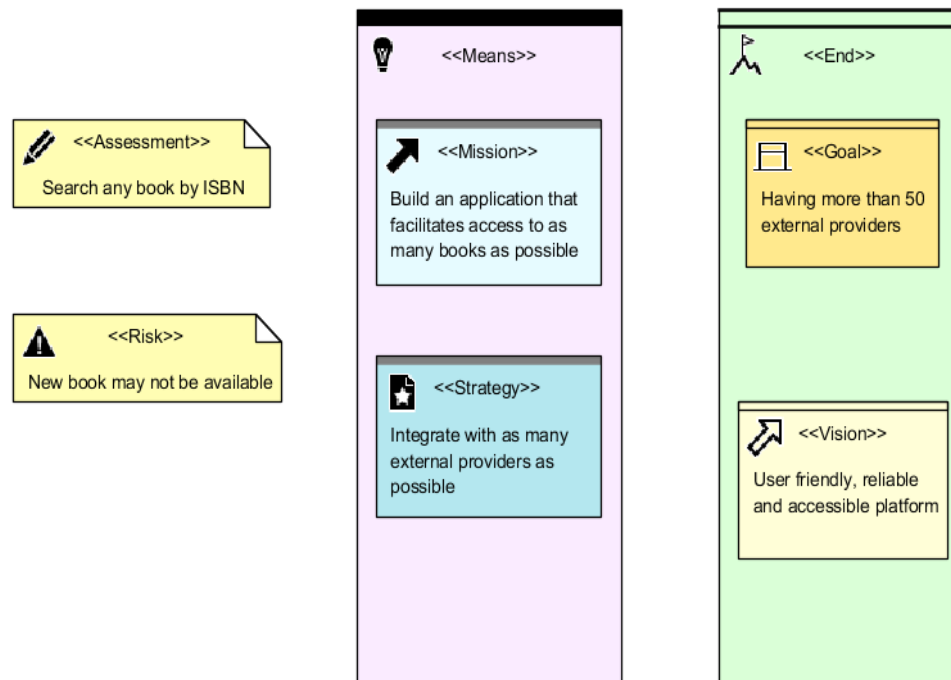


Figure 3: Business motivation model diagram

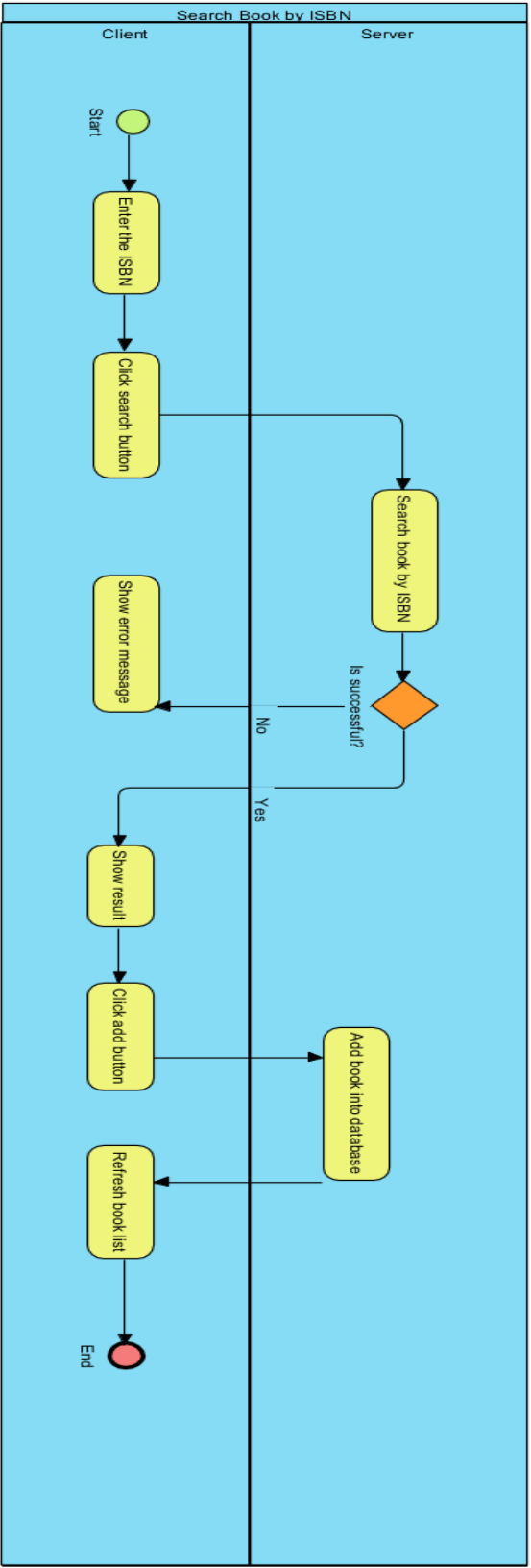


Figure 4: Business process model and notation diagram

How to run

Book Web API

```
$ docker-compose build
```

```
$ docker-compose up
```

Web API will be available on <http://localhost:80/swagger>.

Book Web Portal

```
$ npm install
```

```
$ npm start
```

Web portal will be available on <http://localhost:3000>.