FAKULTA
INFORMAČNÍCH
TECHNOLOGIÍ
ČVUT V PRAZE

Library Information System

Architecture and Database

Documentation of a project for the purpose of the course BIE-SI1.

Authors:

Contents

1. System architecture

In the deployment diagram , the database is located in the same place as the server for the web application having a connection to a relational database in PostgreSQL. For the deployment was used IIS(Internet Information Services). The application is created in .NET framework using for the main language the latest version of C# 6.0.
Users will be able to access the app via browser having a stable connection to internet.

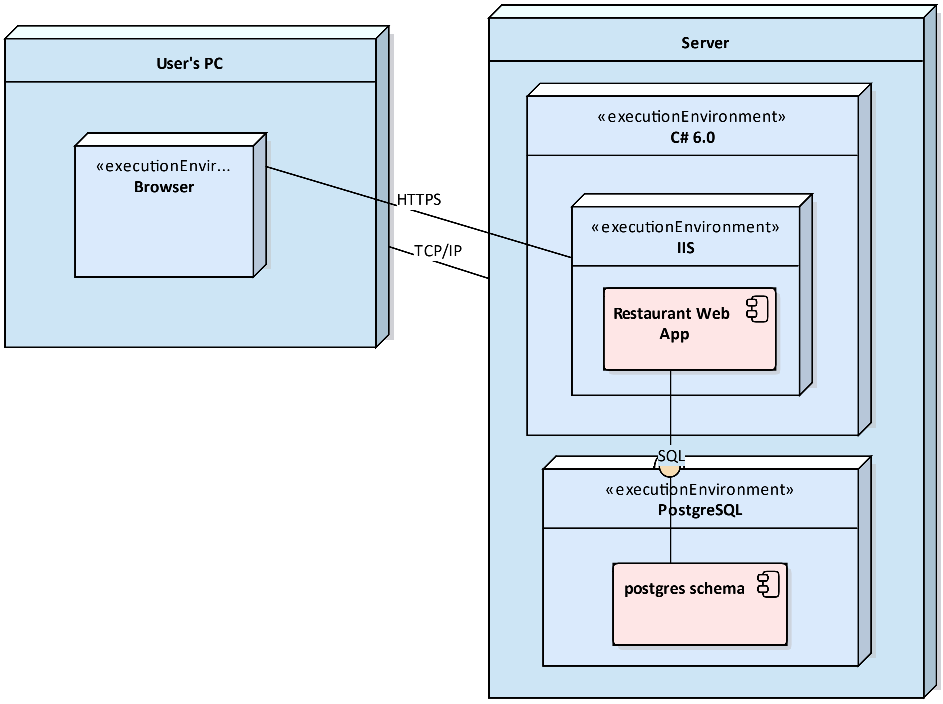

Figure 1 - System architecture

2. Architecture

The logical architecture describes the principles of implementation of the application. It identifies the basic components of the application and the way how these components are connected together.
This project contains a common used three-layered architecture very popular for separating the presentation logic, the business logic and the data logic.
This chapter describes the architecture of the web application of the Restaurant Information System.

The web application for public information about books and their authors will be implemented as a web application based on the following technologies:

- Angular (HTML, CSS, JavaScript)
- ASP.NET framework
- C# 6.0
- csHTML

The architecture is divided into three independent layers:

- Presentation Layer - layer responsible for presentation of application data. For this particular project it is only a visual representation for the packages containing the home controller and the views because in reality it doesn't exist
- Business Layer- layer responsible for all business logic of the application
- Data Layer - layer responsible for data persistence

The layers are mainly packages containing related classes in terms of functionalities.

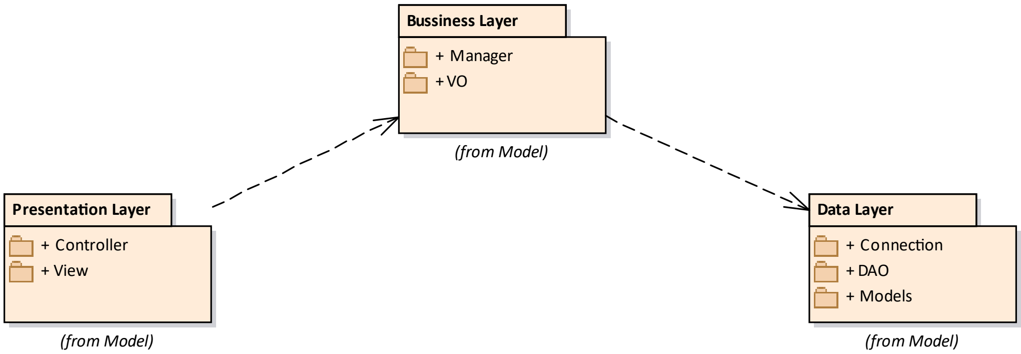

Figure 2 - PackageDiagrams

3. Bussiness Layer

The business layer contains implementation of the business logic and other business related operations of the system. It consists of the managers implementing the system behavior and value objects for sharing information with the presentation layer.

3.1 Manager

Contains the classes handling the business processes and errors.

3.2 VO

Contains composite models used for parsing data from viewer or for sending data to the viewer.

4. Data Layer

The data layer is responsible for data persistence. It consists of DAO classes implementing the persistence operations and entities representing the persistent data.
The implementation of the data layer is based on the SpringBoot and its native support for database.

4.1 Connection

Class used for making the connection to the database using Singleton design pattern.

4.2 DAO

Dao represents the classes accessing the database using different queries. It is divided by the business layer using an interface and an abstract generic class.

4.3 Models

The models are classes according to the database.

5. Presentation Layer

The presentation layer is responsible for sharing the information with a user or other systems. It consists of a web user interface.

5.1 Controller

Contains the main home controller which starts the web application for client and server.

5.2 View

View contains cshtml pages for displaying information to the user.

6. Database Model

In this paragraph it is presented the relationship between entities. The main focus lies on reservation and meal in this iteration.

6.1 Database Model

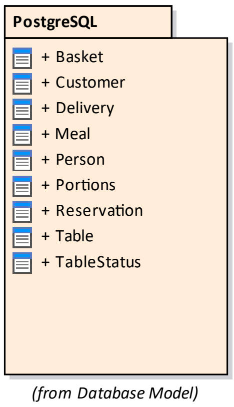

Figure 3 - Database Model

6.2 PostgreSQL

This package describes a general view of the database as how it should look right now. The implementation works currently with this tables.

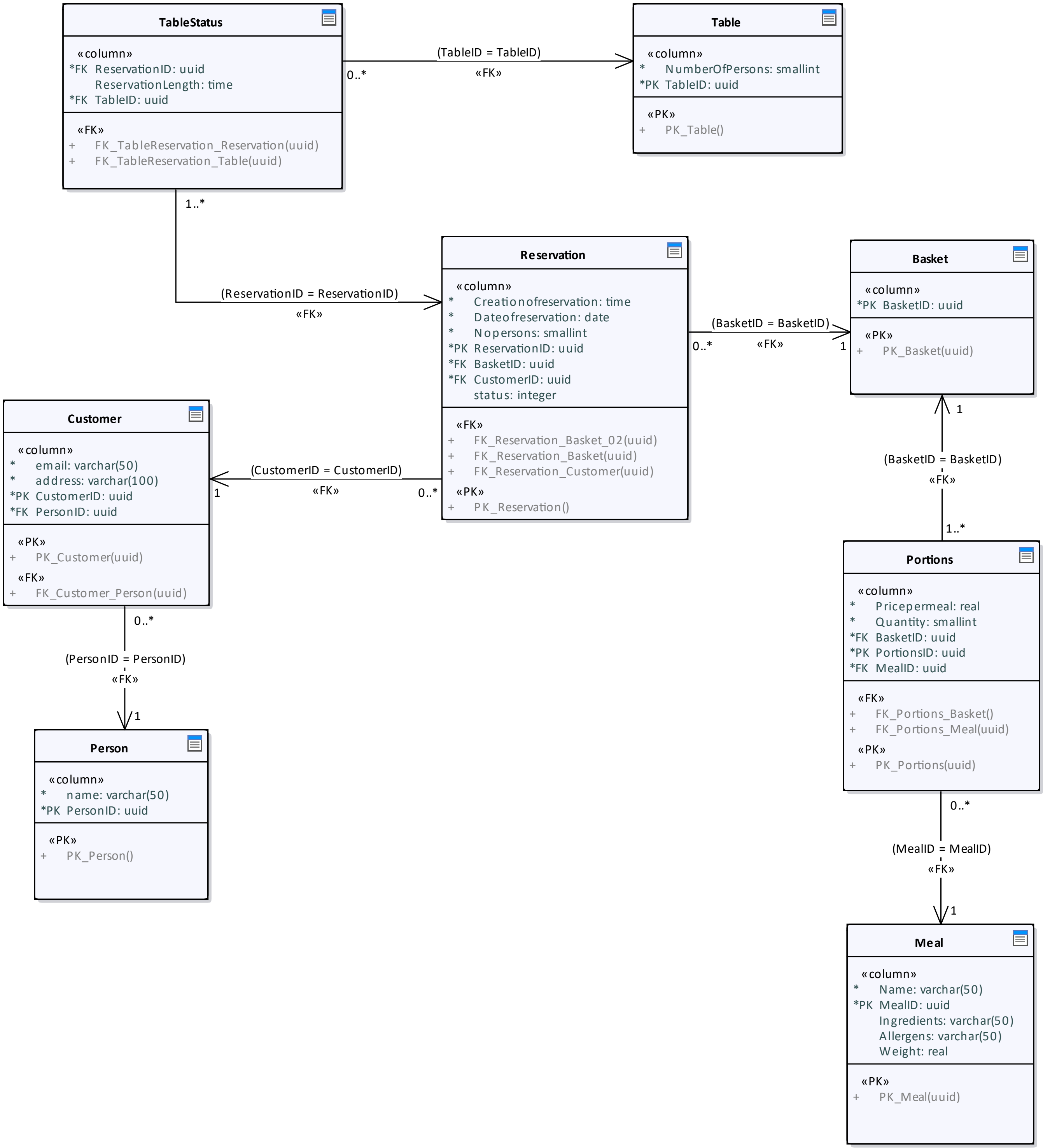

Figure 4 - PostgreSQL

6.2.1 «table» Basket

Basket table represents a container for multiple portions.

Column name	Data type	Not null	Description
BasketID	uuid	True	Primary key for basket.

6.2.2 «table» Customer

Customer is a special type of person who can create reservations and orders.

Column name	Data type	Not null	Description
email	varchar(50)	True	The e-mail address of the customer needed for receiving information.
address	varchar(100)	True	Postal address used for restaurant deliveries.
CustomerID	uuid	True	Primary key for customer table.
PersonID	uuid	True	Foreign key referencing a person.

6.2.3 «table» Delivery

Delivery is an extension of Order , having more fields needed for successfully delivering the list of meals.

Column name	Data type	Not null	Description
DeliveryAddress	varchar(50)	True	The address for delivering the meals.
TimeOfDelivery	time	True	The supposed time of arrival for the meals to arrive at the destination.
DeliveryID	uuid	True	
OrderID	uuid	False	

6.2.4 «table» Meal

Meal entity represents the main product to be ordered and manipulated.

Column name	Data type	Not null	Description
Name	varchar(50)	True	Name of the dish.
MealID	uuid	True	Primary key for meal table.
Ingredients	varchar(50)	False	List of ingredients in meal.
Allergens	varchar(50)	False	List of allergens in meal.
Weight	real	False	The weight in grams of a meal.

6.2.5 «table» Person

A generic entity for a person interacting with the restaurant.

Column name	Data type	Not null	Description
name	varchar(50)	True	First name of the person.
PersonID	uuid	True	

6.2.6 «table» Portions

Portion represents the entity for meal duplicates.

Column name	Data type	Not null	Description

Column name	Data type	Not null	Description
Pricepermeal	real	True	Price for one meal.
Quantity	smallint	True	The number of meals ordered.
BasketID	uuid	True	Foreign key to the basket.
PortionsID	uuid	True	Primary key for portion table.
MealID	uuid	True	Foreign key referencing a meal.

6.2.7 «table» Reservation

Reservation represents the entity of booking a table for a specific time via the web site.

Column name	Data type	Not null	Description
Creationofreservation	time	True	The time chosen for reservation.
Dateofreservation	date	True	The date chosen for reservation.
Nopersons	smallint	True	Represents a quantifying attribute for the number of people present in the reservation.
ReservationID	uuid	True	Primary key of reservation table.
BasketID	uuid	True	Foreign key to a basket of meals.
CustomerID	uuid	True	Foreign key pointing at the customer.
status	integer	False	

6.2.8 «table» Table

Entity for describing a physical table.

Column name	Data type	Not null	Description
NumberOfPersons	smallint	True	The number of people a table can have.
TableID	uuid	True	Primary key for table.

6.2.9 «table» TableStatus

Table Status creates a mapping between tables and reservation used for searching available tables in a particular day and time.

Column name	Data type	Not null	Description
ReservationID	uuid	True	Primary and foreign key to reservation table.
ReservationLength	time	False	Maximum possible length in minutes between two reservations so that they won't overlap.
TableID	uuid	True	Primary and foreign key for table.