Cake Master

Turn any of your creations into an amazing dessert

Software Design Document

Authors:

Ofer Elfassi 038110086

Dekel Ben-David 311157051

17/06/2021

Cake Master, Dekel Ben David & Ofer Elfassi

Table of Contents

1. Introduction

This section will briefly describe the main goal of Cake master application and the problems it aiming to solve.

1.1. System Overview

These days, people who want to bake pastries are reaching for the web to get recipes, and therefore limited to the same premade recipes.

Cake Master will provide a way to develop new and genuine pastries over 3D cake dashboard painter tool, while also taking care of dietary preferences, allergies, favorite food, and other user predefined and calculated preferences.

At the end, Cake master will provide full guided recipe with grocery list and list of available pastry kitchens that can made the pasty for them.

1.2. Problem Description

Nowadays:

- Searching for new pastry recipes that meet one's creative ideas, needs and limitations can be hard task, and usually the end result is unknown.
- While reaching for new recipe, sometimes there are missing ingredients or proper means to execute the recipe directions.

1.3. Goals

Our goal is to:

- Provide users with the ability to plan their pastries as 3D module, with detailed information about their composition, as well as limitations for a specific diet plan, allergies, and other preference.
- Generate full detailed recipes, grocery list, potential pastry kitchen list that can make the pastry and full guided step by step directions for making at home.

1.4. Scope

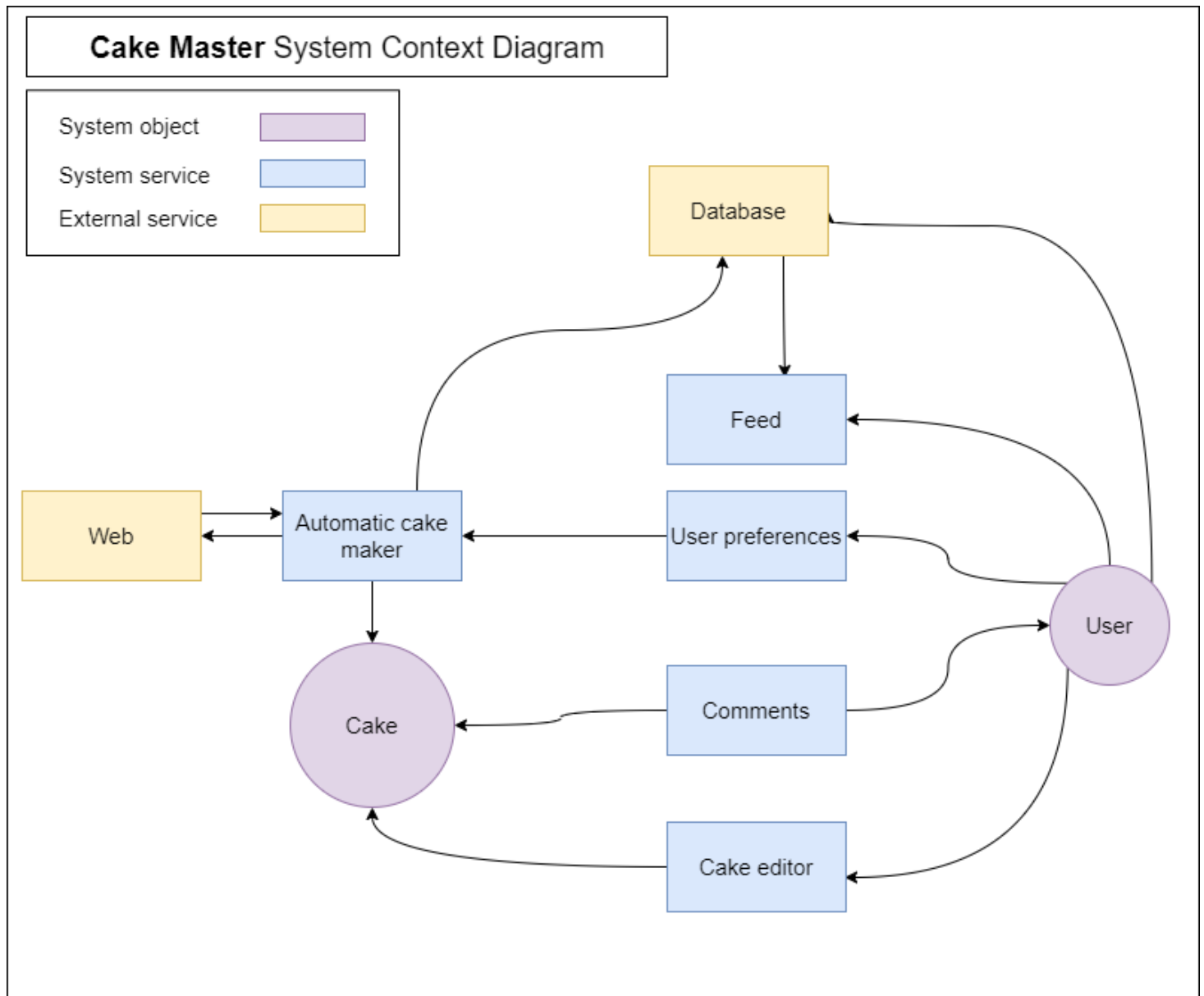Culinary, Pastry recipes.

1.5. Glossary

Tags - define the user preferences and restrictions

Client interface – related to the client interaction interface with the server and database

Server interface – related to the server section where all business logic is implemented

2. System Architecture – System Context Diagram

Cake Master System Context Diagram

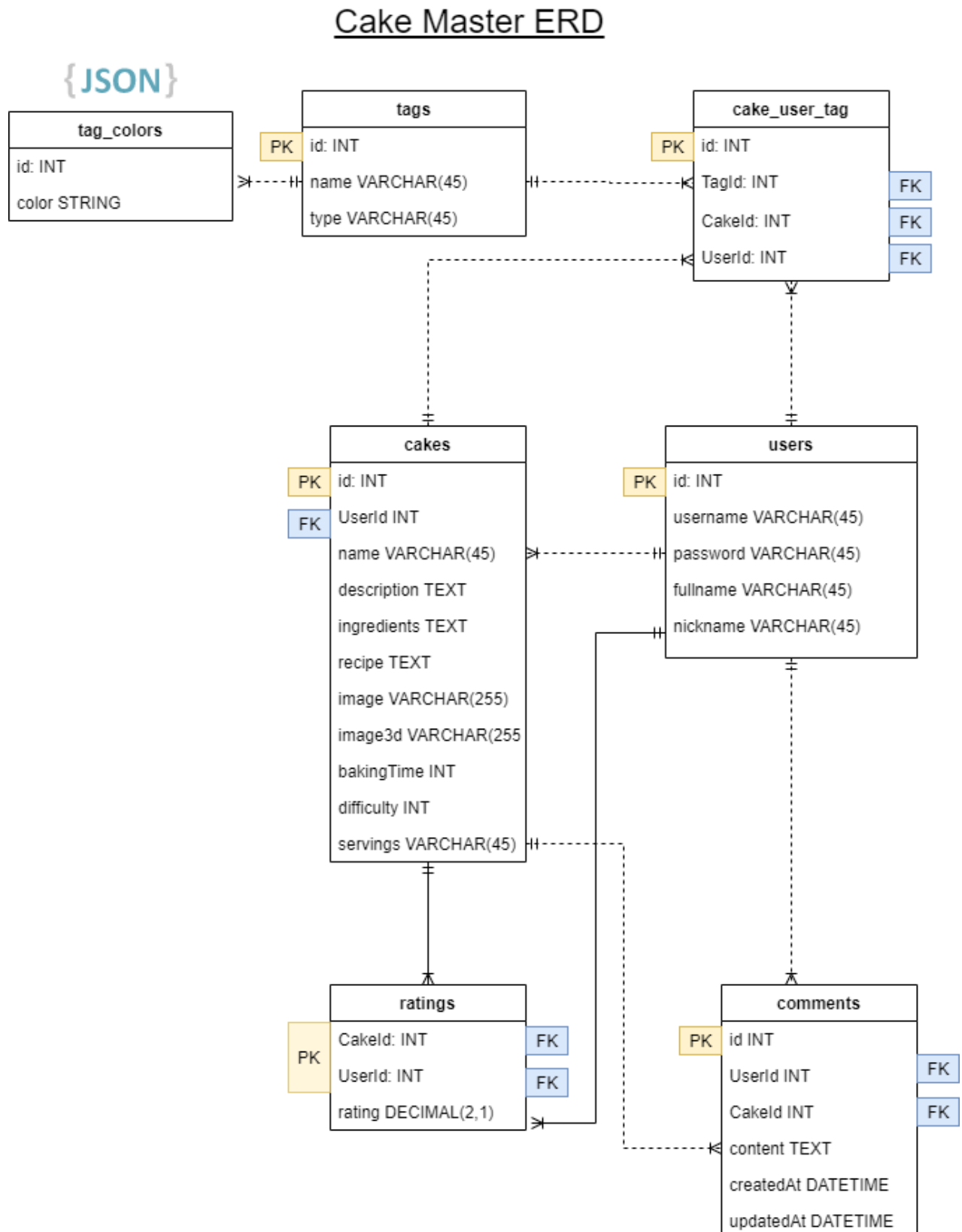| Legend	
System object	
System service	
External service	

Cake Master, Dekel Ben David & Ofer Elfassi

3. Design
The following section will describe the system analysis and design diagrams

3.1. Data Design
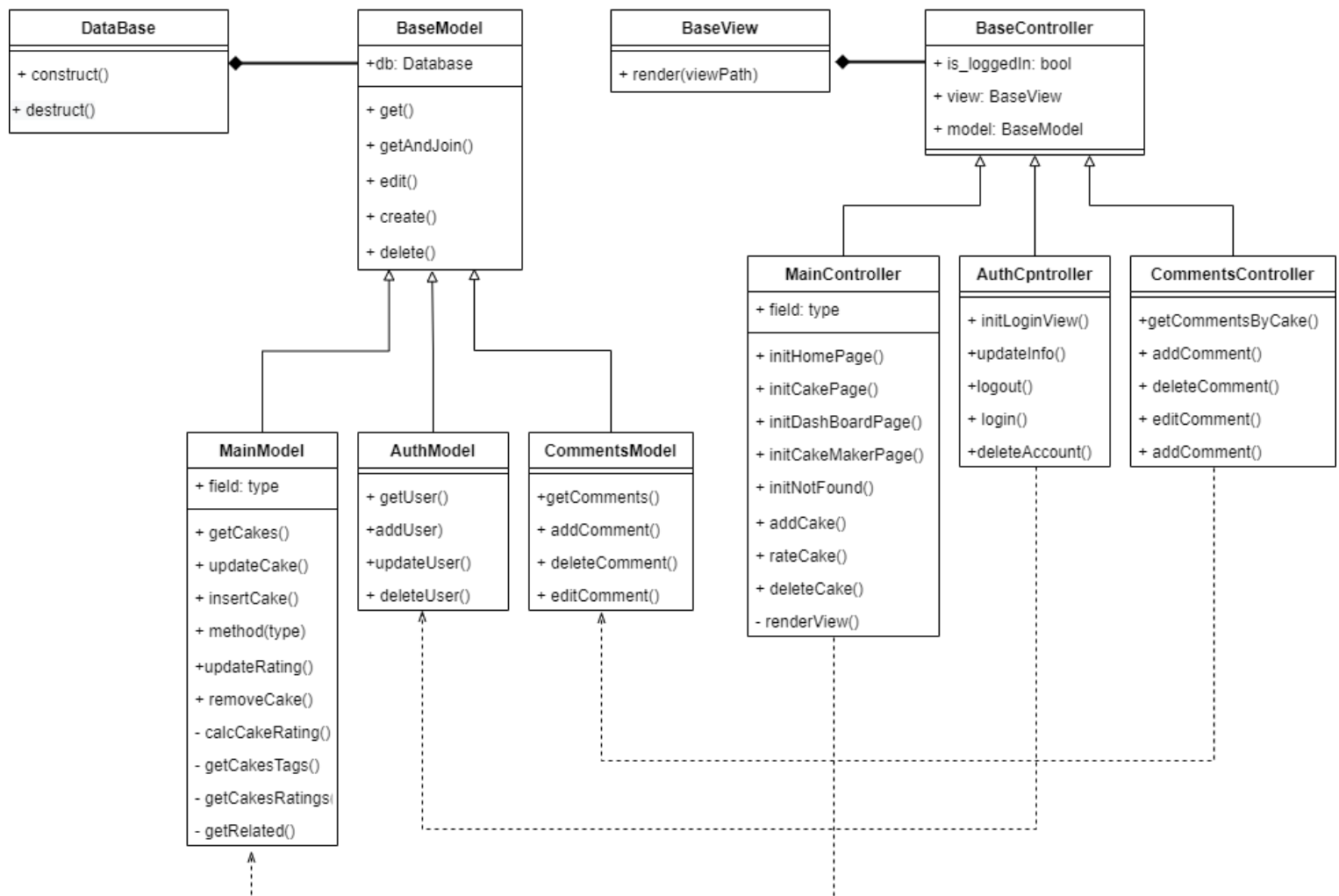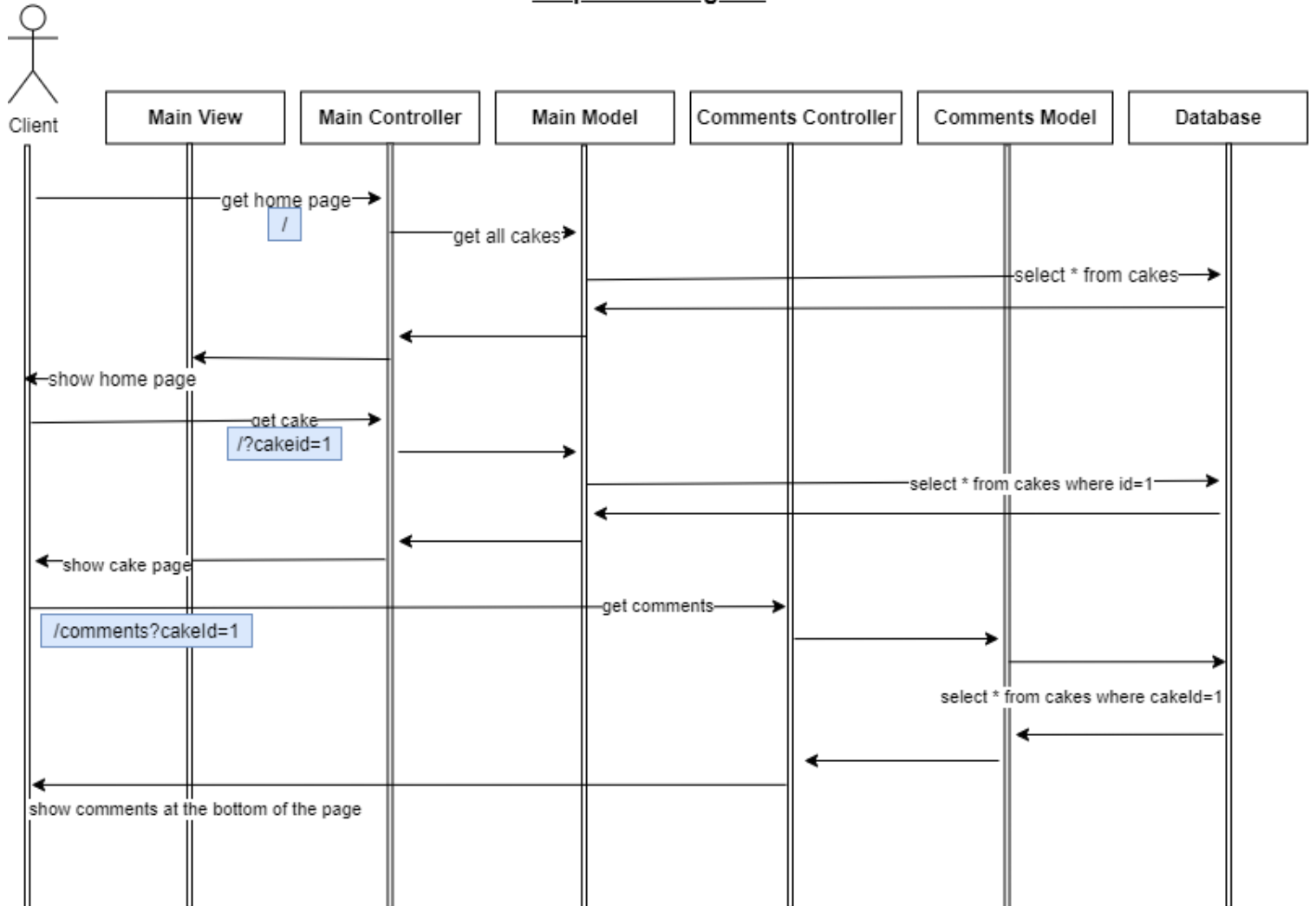Database Description- include 3 tables and JSON structure

Cake Master ERD

3.2. Structural Design

Class Diagram

CLIENT INTERFACE

SERVER INTERFACE

DataBase
- + construct()
- + destruct()

BaseModel
- +db: Database
- + get()
- + getAndJoin()
- + edit()
- + create()
- + delete()

BaseView
- + render(viewPath)

BaseController
- + is_loggedIn: bool
- + view: BaseView
- + model: BaseModel

MainModel
- + field: type
- + getCakes()
- + updateCake()
- + insertCake()
- + method(type)
- +updateRating()
- + removeCake()
- - calcCakeRating()
- - getCakesTags()
- - getCakesRatings()
- - getRelated()

AuthModel
- + getUser()
- +addUser)
- +updateUser()
- + deleteUser()

CommentsModel
- +getComments()
- + addComment()
- + deleteComment()
- + editComment()

MainController
- + field: type
- + initHomePage()
- + initCakePage()
- + initDashBoardPage()
- + initCakeMakerPage()
- + initNotFound()
- + addCake()
- + rateCake()
- + deleteCake()
- - renderView()

AuthCpntroller
- + initLoginView()
- +updateInfo()
- +logout()
- + login()
- +deleteAccount()

CommentsController
- +getCommentsByCake()
- + addComment()
- + deleteComment()
- + editComment()
- + addComment()

3.3. Interaction Design

Sequence Diagram

Cake Master, Dekel Ben David & Ofer Elfassi

4. Software Architecture
The following section will demonstrate the system data flow architecture.
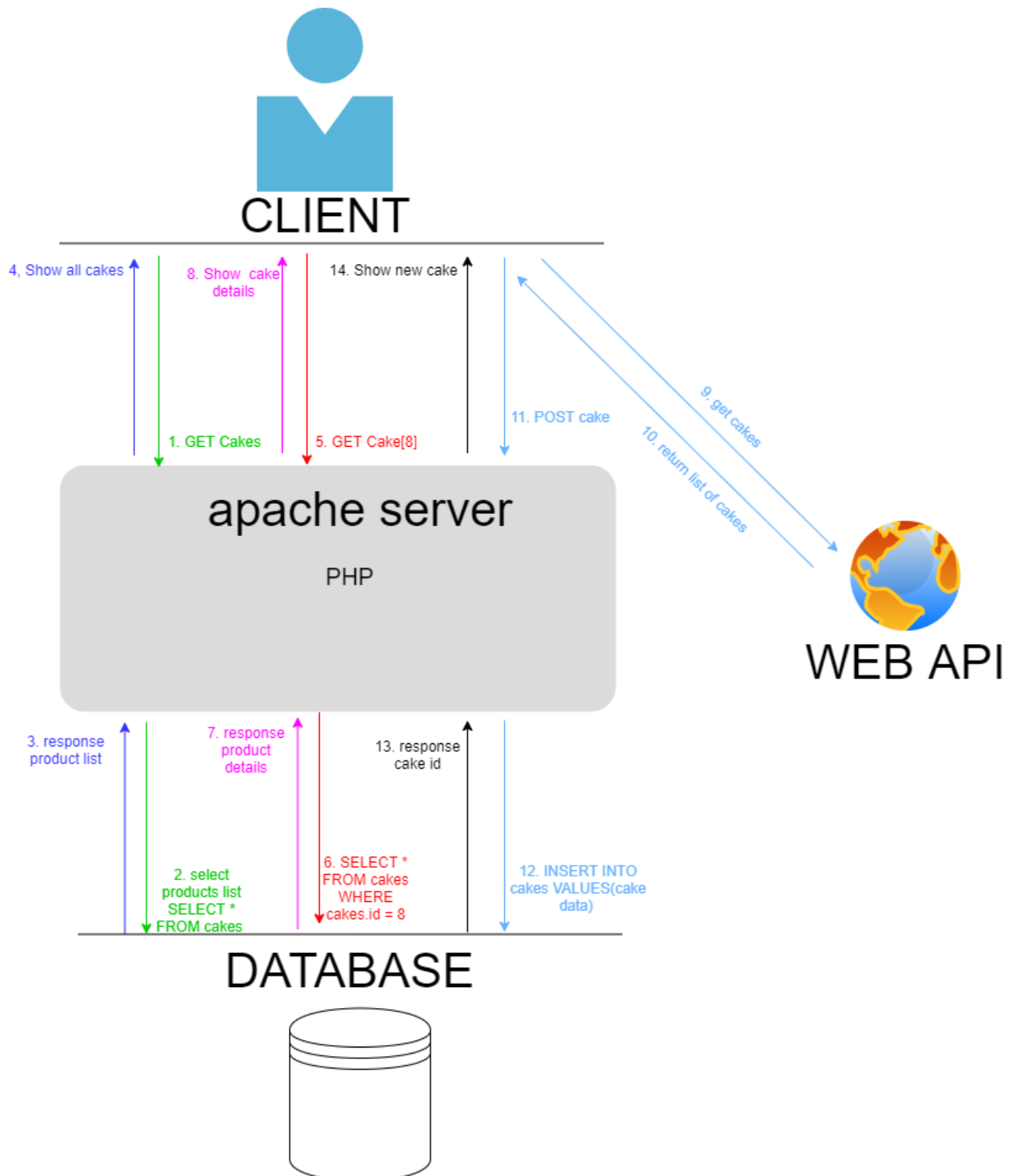
5. Verification and Validation

This section will discuss tests requirement before and after deployment.

Tests before deployment

- Visual inspection of the application on different screen sizes
- Make sure there is proper error handling for different scenarios
  Unauthorized user cannot comment
  Unauthorized user cannot rate cakes
  Unauthorized user cannot access cake maker page
  Authorized user cannot edit other users cakes
  Authorized user cannot edit other users comments
- Attempt to call URL with invalid parameters will lead to error page

Tests after deployment

- Redo all tests from previous section
- Test and adjust URL paths to the hosting server path
- Make sure that all database operations work properly