Opriş Bogdan-Alexandru
30432_2

FoodTrack
Supplementary Specification

Version 1.0

Revision History

Date	Version	Description	Author
20/MAR/2019	1.0	Initial Requirements Statement	Opriş Bogdan-Alexandru

Table of Contents

Supplementary Specification

1. Introduction

The Supplementary Specification captures the system requirements that are not readily captured in the use cases of the use-case model. Such requirements include:

- Legal and regulatory requirements, including application standards.
- Quality attributes of the system to be built, including usability, reliability, performance, and supportability requirements.
- Other requirements such as operating systems and environments, compatibility requirements, and design constraints.

2. Non-functional Requirements

2.1 Availability

The system is not expected to be used in urgent scenarios so we can afford a SLA[1] of 99.5%. This translates into a yearly downtime of roughly 1 day and 19 hours, or a monthly downtime of 3 hours and 39 minutes. This time can be used to perform software updates, data compression and garbage collection.

2.2 Performance

Performance is a thoroughly key factor for this system. In this case we can agree for an almost instantaneous response time for the updates made by the administrator in the database or by the users in their diary.

2.3 Security

The system will be secured using https encrypted connections. Also we will demand user authentication and will not keep passwords in plain text. Other user data will not be encrypted as we do not find it as being sensible information.

2.4 Testability

The system will be precisely tested with different stages of the design process respecting the flow shown below (Figure 1 . V-Model Testing): Unit Testing, Integration Testing, System Testing, Acceptance Testing.

1 SLA = Service Level Agreement = Availability

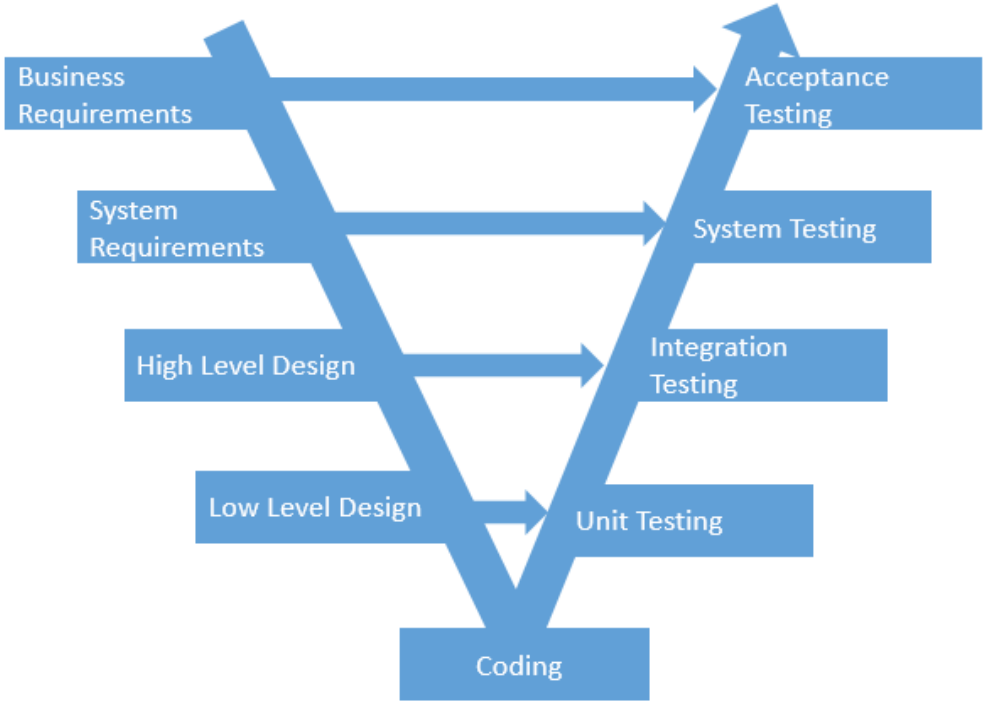

Figure 1 V-Model Testing

2.5 Usability

The user should be able to reach their desired goal very quickly. The interface provided by the application should be user-friendly both for the users and administrators.

3. Design Constraints

The system's implementation language will be Java 8, which is suited to provide all the requirements in order to provide this system. The software development process will be the Rational Unified Process (RUP), tailored to fit the team and the project. The conceptual architecture of the system will be a client server as illustrated in Figure 2 Conceptual Architecture. The required development tools are either Eclipse IDE or IntelliJ IDEA. In terms of libraries we will use: JavaFX, Hibernate, JDBC and GSON.

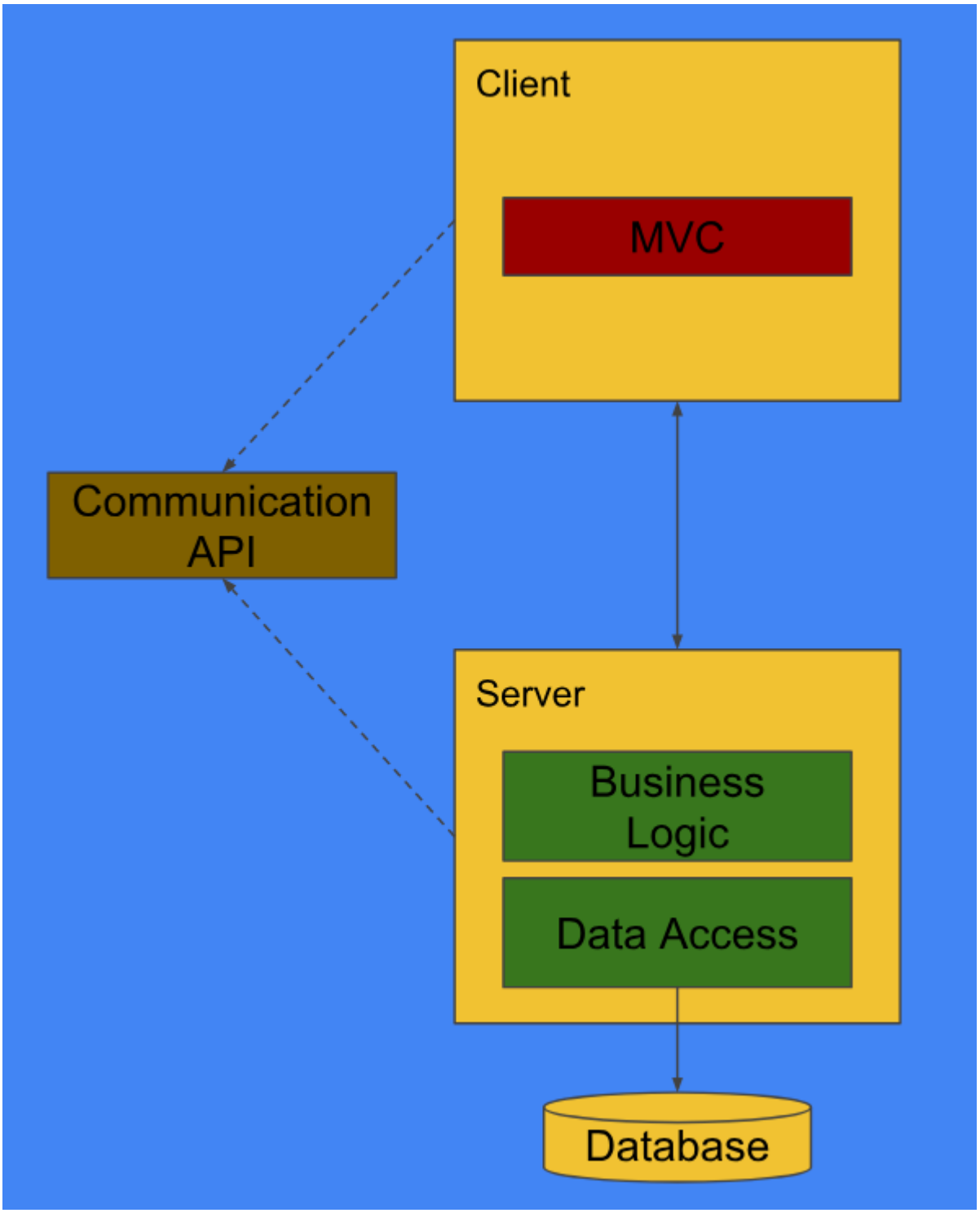

Figure 2 Conceptual Architecture