FaceBass – Facial Recognition-based Bus Pass

Analysis and Design Document

Student: Radu Petrisel

Group: 30432

Revision History

Date	Version	Description	Author
03/04/2018	1.0	Domain model, Architectural design, component and deployment diagram	Radu Petrisel
19/04/2018	2.0	Fixed domain model; Added data model	Radu Petrisel

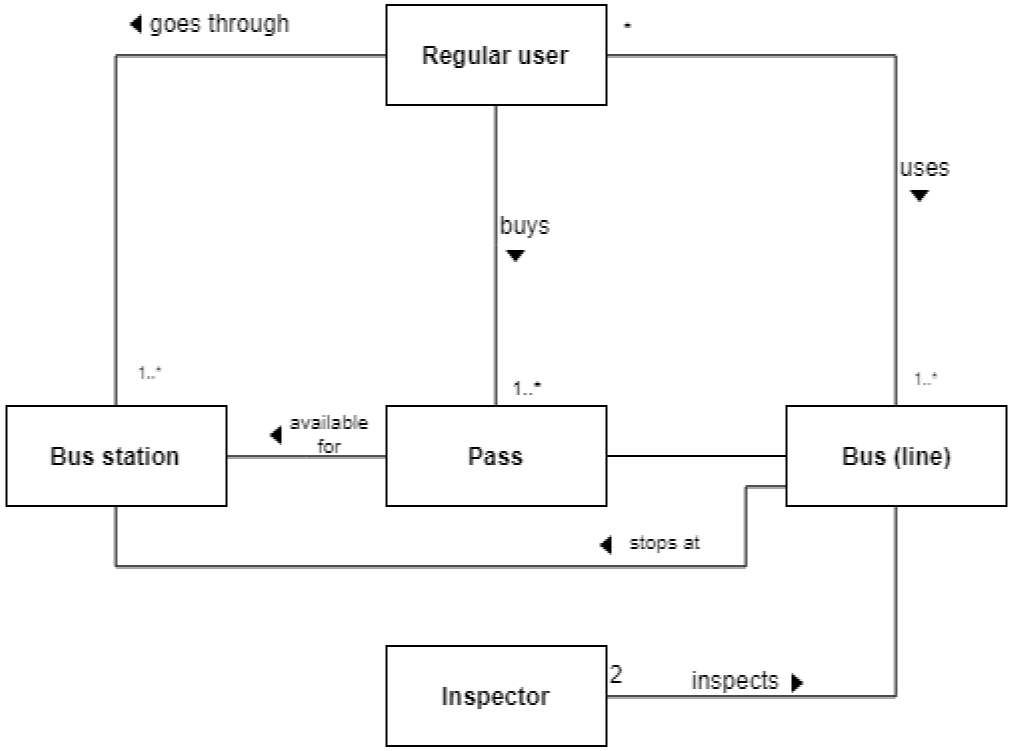
Table of Contents

I. Project Specification

The Facial-recognition based bus pass system presented in this vision aims to solve the problems with most of today's bus passes – people need to carry IDs with them in order to identify themselves. By the new system, all they will need is their face. The system will be titled FaceBass (from bus and pass).

II. Elaboration – Iteration 1.1

1. Domain Model

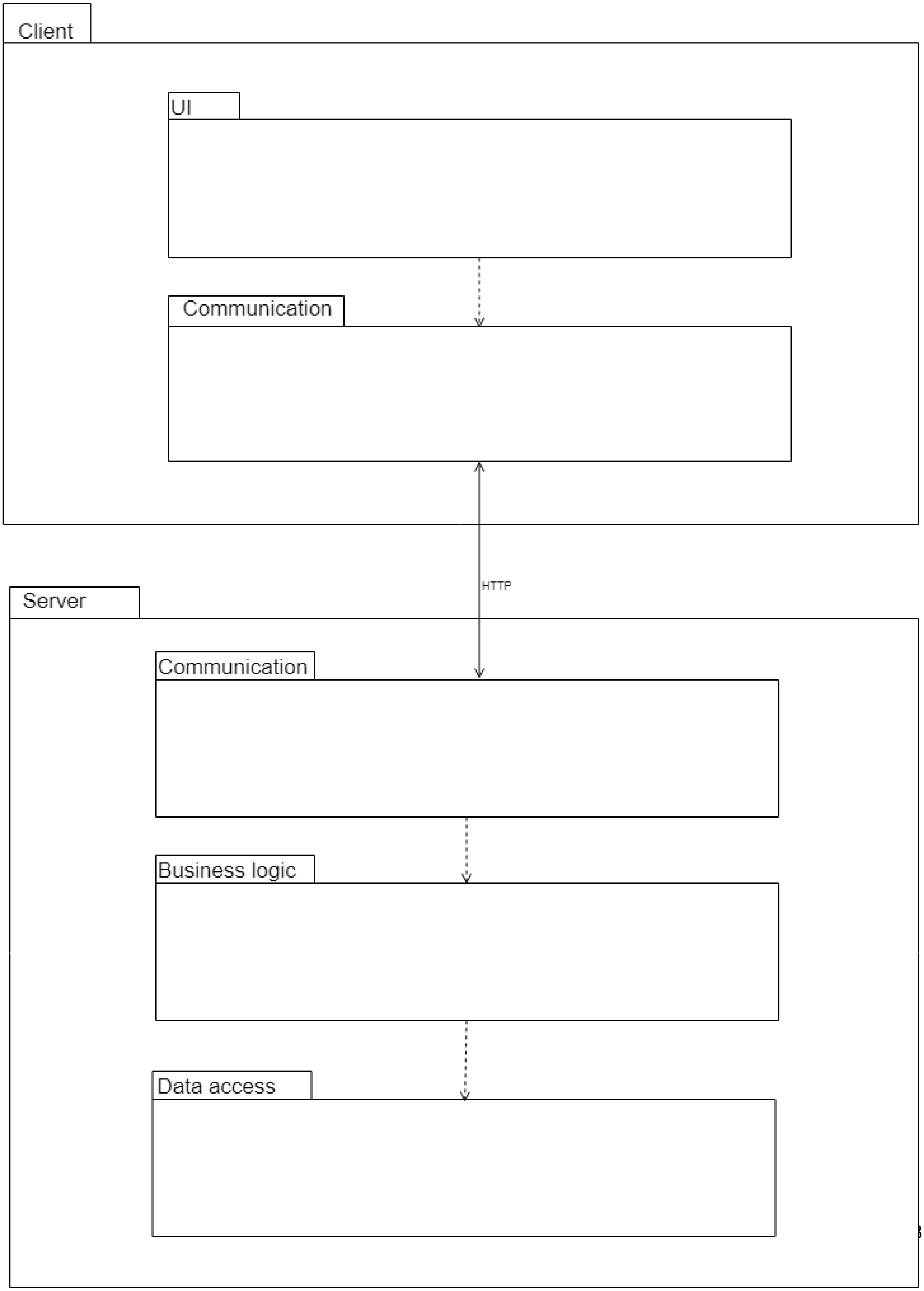


2. Architectural Design

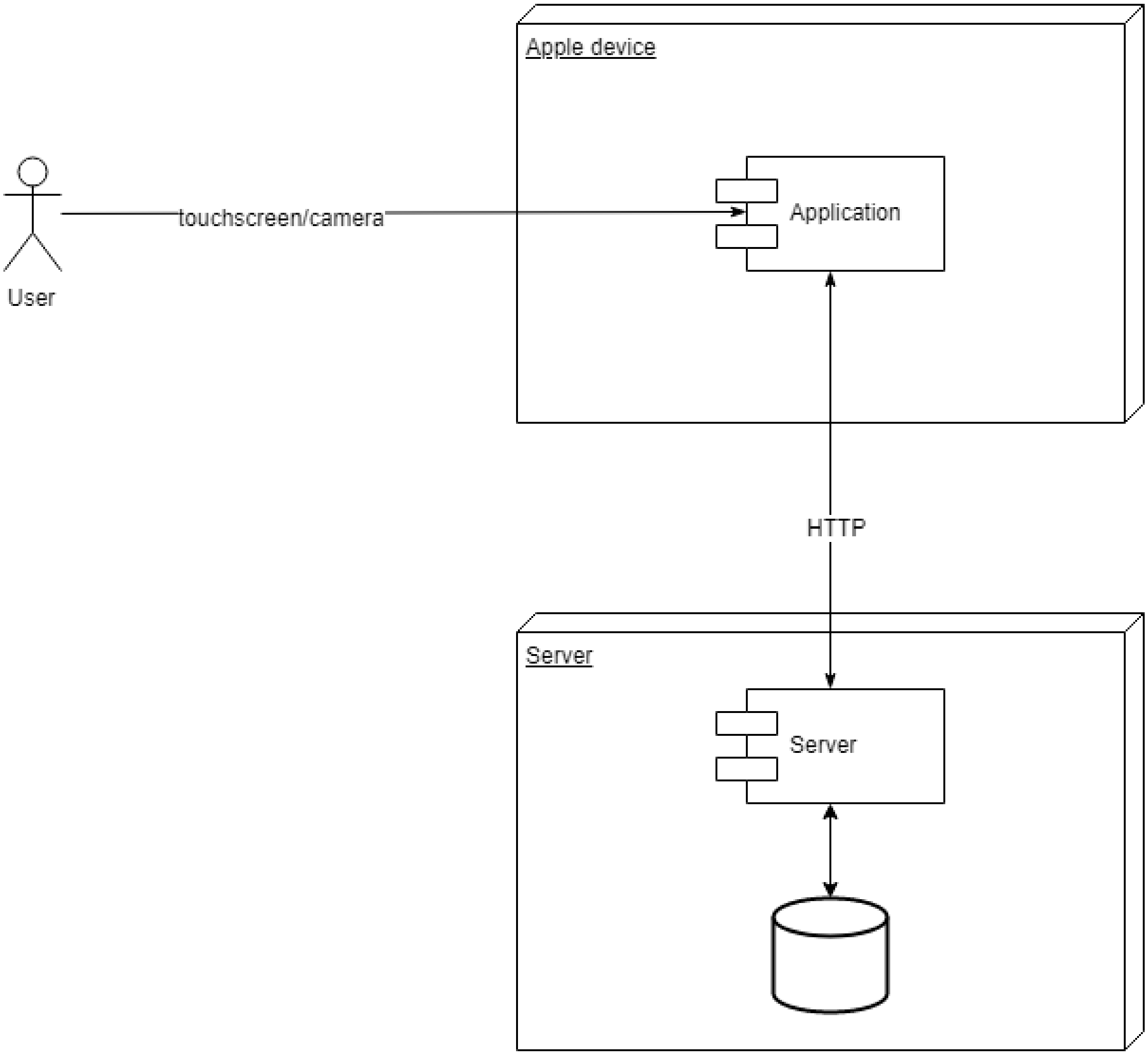
2.1 Conceptual Architecture

The system will use a Client-Server, layered architecture. The iOS application is the client, and the server is a Python application. The client has the presentation layer and a part of the business logic, and the server has the data access and the rest of the business logic layer.

2.2 Package Design



2.3 Component and Deployment Diagrams



III. Elaboration – Iteration 1.2

1. Design Model

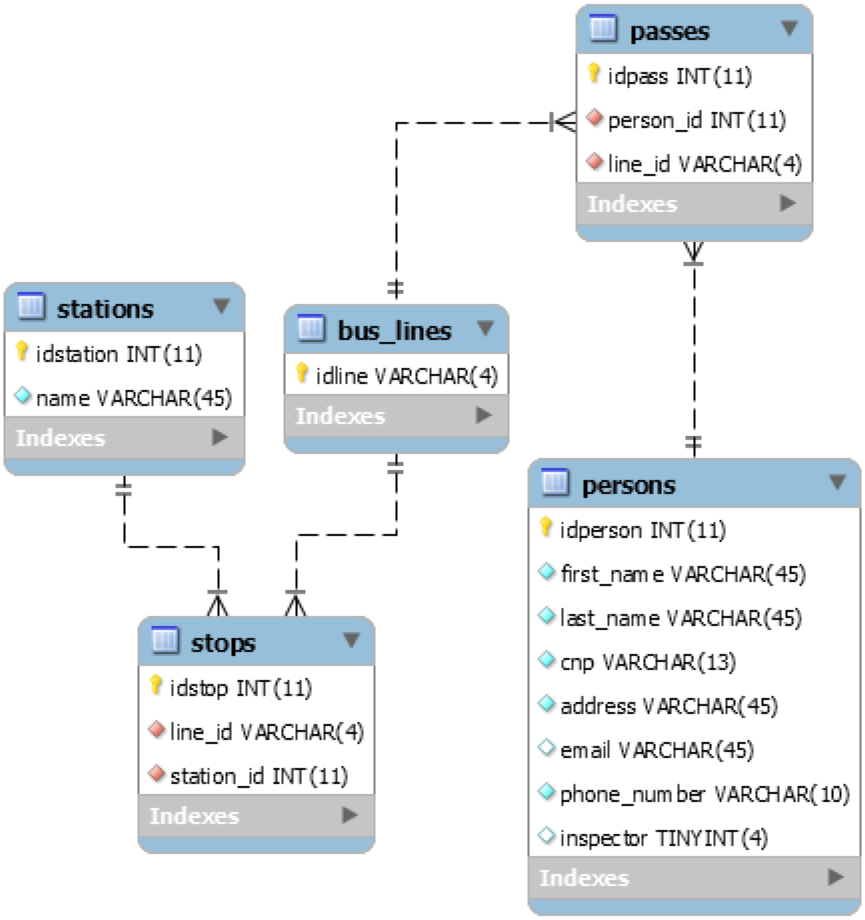
1.1 Dynamic Behavior

[Create the interaction diagrams (1 sequence, 1 communication diagrams) for 2 relevant scenarios]

1.2 Class Design

[Create the UML class diagram; apply GoF patterns and motivate your choice]

2. Data Model



3. Unit Testing

[Present the used testing methods and the associated test case scenarios.]

IV. Elaboration – Iteration 2

1. Architectural Design Refinement

[Refine the architectural design: conceptual architecture, package design (consider package design principles), component and deployment diagrams. Motivate the changes that have been made.]

2. Design Model Refinement

[Refine the UML class diagram by applying class design principles and GRASP; motivate your choices. Deliver the updated class diagrams.]

V. Construction and Transition

1. System Testing

[Describe how you applied integration testing and present the associated test case scenarios.]

2. Future improvements

[Present future improvements for the system]

VI. Bibliography