Analysis and Design Document

Student: Ioana Bozdog

**Group: 30431**

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Description** | **Author** |
| <18/3/20> | <1.0> | <the general view of the app> | <Ioana Bozdog> |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Table of Contents

I. Project Specification 4

II. Elaboration – Iteration 1.1 4

1. Domain Model 4

2. Architectural Design 4

2.1 Conceptual Architecture 4

2.2 Package Design 4

2.3 Component and Deployment Diagrams 4

III. Elaboration – Iteration 1.2 5

1. Design Model 5

1.1 Dynamic Behavior 5

1.2 Class Design 5

2. Data Model 5

3. Unit Testing 5

IV. Elaboration – Iteration 2 6

1. Architectural Design Refinement 6

2. Design Model Refinement 6

V. Construction and Transition 6

1. System Testing 6

2. Future improvements 6

VI. Bibliography 6

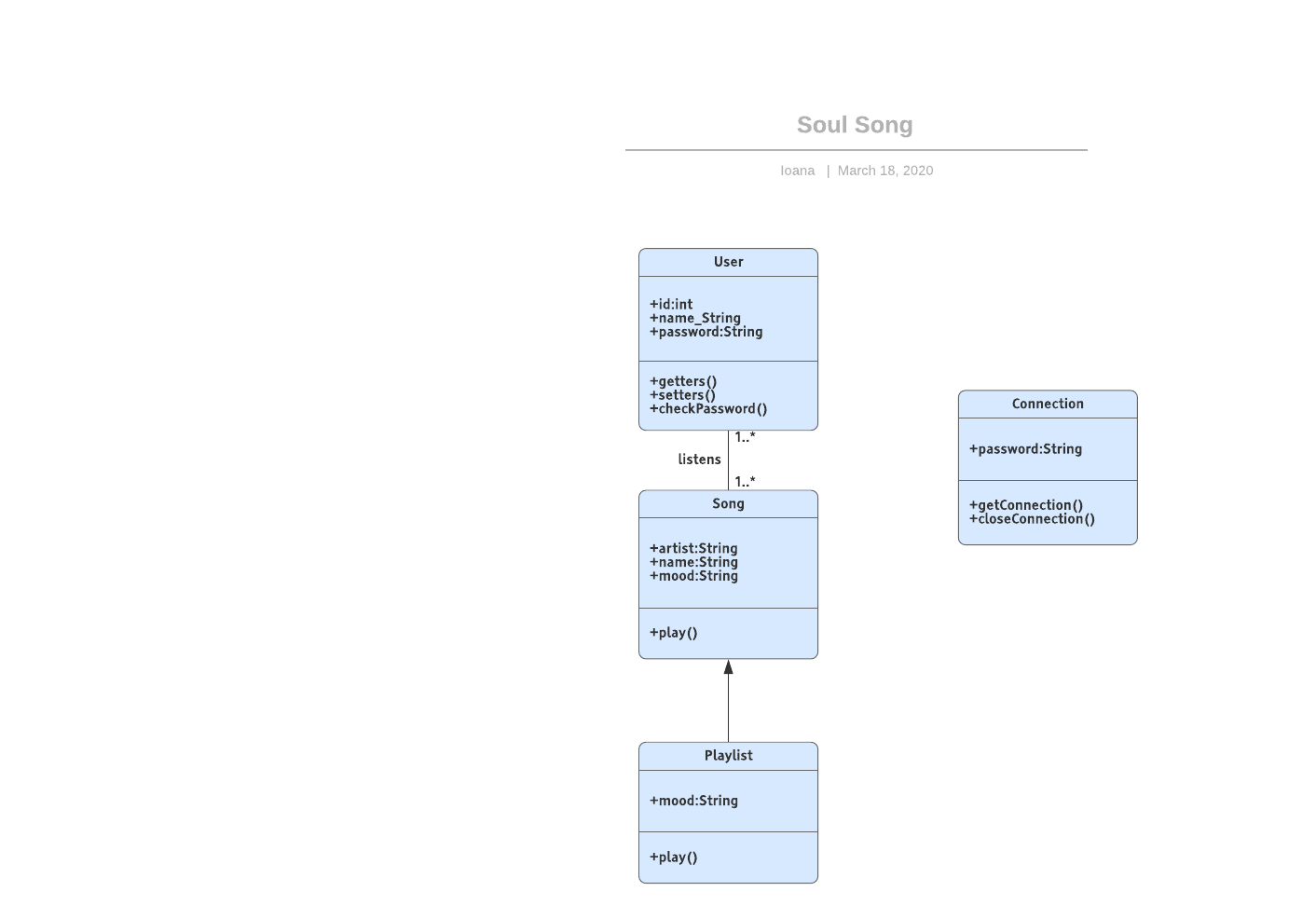
# Project Specification

*This project consists of a music application that can read people’s emotions in real time and suggest a playlist according to the mood they are in. The project will be implemented either in Android Studio using Java or in Visual Studio using C# and an API from Microsoft Azure called emotion API. The user could also search for a playlist or even only a song to play. The app will also use a database such as Firebase to store and retrieve songs. The user will also be capable to create an account and login.*

# Elaboration – Iteration 1.1

# Domain Model

*The domain model of the app can be seen below in the form of conceptual classes. Please keep in mind that this is the first version of the app, so the model below is a very rough interpretation of what the app will look like.*

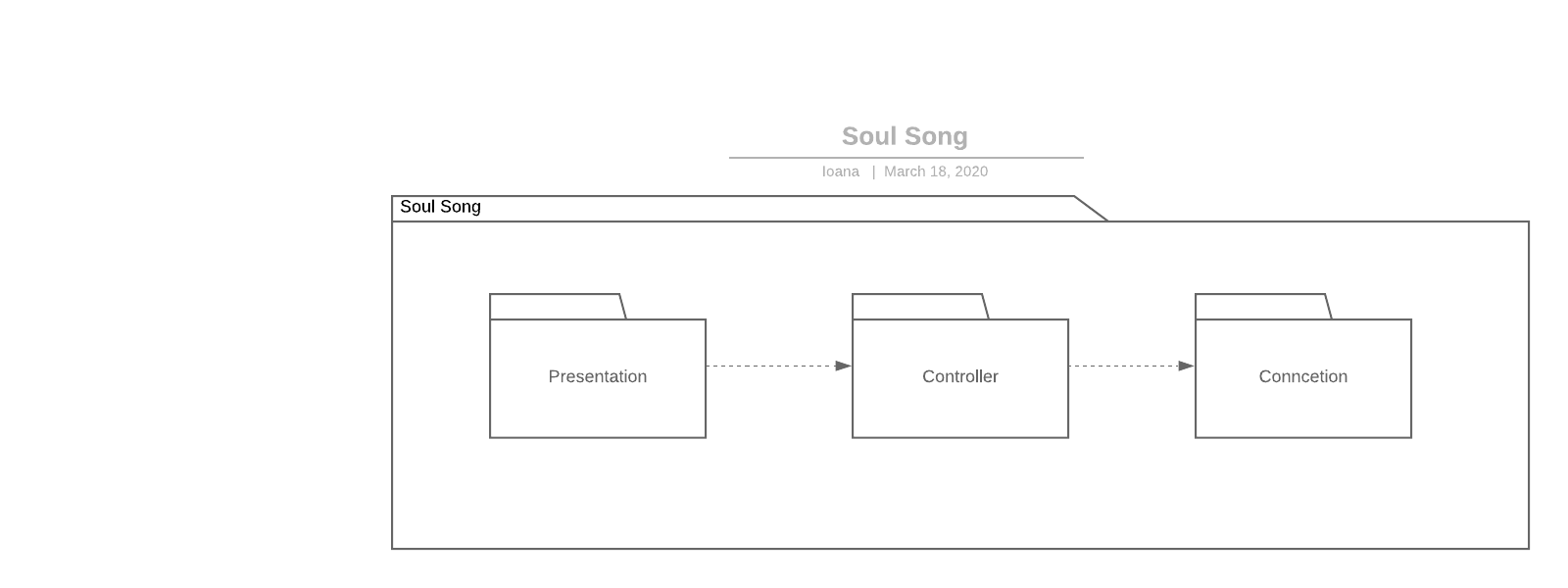
**

# Architectural Design

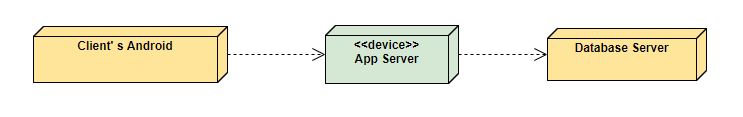
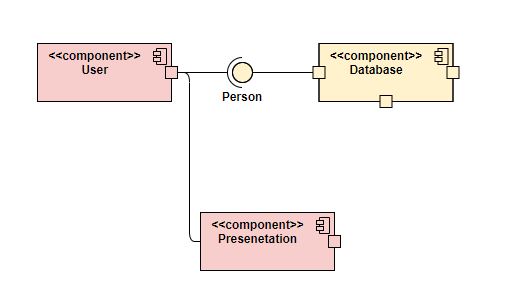
## Conceptual Architecture

*Since the application uses Firebase, I believe the best architecture would be Client-Server as the Client will have to request data.*

## Package Design

**

## Component and Deployment Diagrams



# Elaboration – Iteration 1.2

# Design Model

## Dynamic Behavior

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

## Class Design

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

# Data Model

*[Create the data model for the system.]*

# Unit Testing

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

# Elaboration – Iteration 2

# 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.]*

# Design Model Refinement

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

# Construction and Transition

# System Testing

*The tests will provide an evaluation of the basic operations present in the application, such as login, create account, open camera, detect emotion, connect to database.*

# Future improvements

*I would really like to implement this to have multiple playlists for a single emotion, not just one.*

# Bibliography