**Software Design Specification (SDS)**

Revision History: (The server and client documents should be combined into one for a single project)

|  |  |  |
| --- | --- | --- |
| Date | Author | Description |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

Contents

[**Software Design Specification (SDS)** 1](#_Toc51163298)

[1. Introduction 2](#_Toc51163299)

[1.1. Intended Audience and Purpose 2](#_Toc51163300)

[1.2. How to use the document 2](#_Toc51163301)

[2. System Design 2](#_Toc51163302)

[2.1. Context 2](#_Toc51163303)

[2.2. Design Pattern 2](#_Toc51163304)

[2.3. Architecture 2](#_Toc51163305)

[2.3.1. <Component Diagram> 2](#_Toc51163306)

[2.3.2. <Deploy Diagram> 2](#_Toc51163307)

[3. Module Interface Design 2](#_Toc51163308)

[4. Detailed Design 2](#_Toc51163309)

[4.1. Server Detailed Design 3](#_Toc51163310)

[4.2. Client (Android) Detailed Design 3](#_Toc51163311)

[4.3. Client (Web/Desktop) Detailed Design 3](#_Toc51163312)

[A.    Appendices 3](#_Toc51163313)

[A.1    Definitions and acronyms 3](#_Toc51163314)

[A.1.1    Definitions 3](#_Toc51163315)

[A.1.2    Acronyms and abbreviations 3](#_Toc51163316)

[A.2    References 4](#_Toc51163317)

## Introduction

## Intended Audience and Purpose

<Every technical document should clearly specify who the document is written for and what purpose the document should serve for each intended audience. This section describes the purpose and audience for the Concept of Operations and the Software Requirements.>

## How to use the document

<Describes the document organization. This section should answer for the reader: “Where do I find particular information about X?”>

## System Design

<Use this section to give a detailed description of the system contexts from an architect's point of view. It should make clear the expected context of the software, such as the platform, design pattern, etc.>

## Context

<Specifies the system's operational context: i.e., the programming languages to develop the software with, the operating system your software runs on, the database management system your data will be stored, the internet protocol for the component communication, etc.>

## Design Pattern

<Specifies the technical details of the software system: i.e., model-view-control division, restful service pattern, etc.>

## Architecture

## <Component Diagram>

<Component Diagram (CD) specifies how the system is parted according to the use cases analyzed from RS. >

## <Deploy Diagram>

## Module Interface Design

< It specifies the contracts with which the modules communicate.>

  (Mogic for System Interface Specifications, extra template available; all groups should contribute via interface design of her own module.)

## Detailed Design

< It specifies the design information inside the modules.>

(Each group should contribute, via diagrams either for the whole system or for her module. Optional diagrams are ER diagram, Sequence diagram, Class diagram)

## Server Detailed Design

## Client (Android) Detailed Design

## Client (Web/Desktop) Detailed Design

## A.    Appendices

## A.1    Definitions and acronyms

## 

## A.1.1    Definitions

|  |  |
| --- | --- |
| **Keyword** | **Definitions** |
|  |  |
|  |  |
|  |  |
|  |  |

## A.1.2    Acronyms and abbreviations

|  |  |
| --- | --- |
| **Acronym or**  **Abbreviation** | **Definitions** |
|  |  |
|  |  |
|  |  |

## A.2    References