JetLAG

Student Organization Software Supplementary Specifications

Version 2.0

Revision History

Date	Version	Description	Author
9/29/2022	1.0	First Revision	JetLAG Team

Table of Contents

Table of Contents

Supplementary Specifications

1. Introduction

1.1 Purpose

Supplementary Specifications cover the miscellaneous requirements which are not covered by the Use-Case Specifications or the Software Specifications. These Supplementary Specifications also act as a preface to the document, covering the terms, abbreviations, and assumptions used.

1.2 Scope

These Supplementary Specifications cover the software product and application Student Organization Service made by JetLAG

1.3 Definitions, Acronyms, and Abbreviations

Android
 A Mobile Operating System
Browser
 Computer Software allowing access to the Internet
Device
 The User's Computer or Mobile Device
JetLAG
 Company Name. The organization that is developing the product.
Mobile Device
 A Tablet or Phone running Android and with access to the Internet
SOS
 Student Organization Service. The product being developed

1.4 References

EECS 488 Course Website: https://people.eecs.ku.edu/~saiedian/Teaching/448/

1.5 Overview

The rest of the Supplementary Specifications contain information on the assumptions made while making SOS

2. Assumptions and Dependencies

2.1 Assumptions

2.1.1 User Hardware Assumptions

Users have access to a modern computer or mobile device

2.1.2 User Software Assumptions

If the user is accessing SOS on a computer, the computer has access to a Web Browser. If the user is using a mobile device, it is running the Android Operating System and has access to the Google Play Store.

3. Usability

3.1 Save and Sync times

When a user saves or syncs to their local instance of the UI, it should take no longer than 1 minute to perform this operation

3.2 Login Time

When a user logs in, it should take no longer than 1 minute from when the user starts the login procedure to when they are able to perform operation within the user interface with their classes.

4. Reliability

4.1 Availability

Access to the UI will be limited to specific times when the development team deems it necessary or when demos need to be done for the class, this is due to the computing hour cap imposed by the Azure for Students subscription.

5. Performance

5.1 Capacity

This system will at most support 8 users at a time, due to being limited to 1 cloud virtual machine to handle data transactions.

5.2 Save and Sync times

When a user saves or syncs to their local instance of the UI, it should take no longer than 1 minute to perform this operation

5.3 Login Time

When a user logs in, it should take no longer than 1 minute from when the user starts the login procedure to when they are able to perform operation within the user interface with their classes.

5.4 Storage use

The database cannot use more than 10 GB of storage, due to data caps of free services for Azure.

6. Supportability

6.1 Flutter app

The flutter app will allow developers to use a portable version of the code for UI development

6.2 Data Broker

The data broker will be a java project that is built using spring boot and Gradle for easy library upgrades and automatic and consistent infrastructure builds.

7. Design Constraints

7.1 User interface

The UI will be programmed as a Flutter app. The Backend will be stored on Azure servers, and the Database will be a Cosmos DB.

8. Security

8.1 Log-in

Users will be able to create Unique Log-In information, and their information will be private to that user.

9. Online User Documentation and Help System Requirements

All Documentation and instructions will be hosted on an azure Dev-Ops repository that will periodically be updated by the development team.

10. Interfaces

Data broker to cosmos DB – uses TLS and https connections to communicate between VM and the database.
data broker to Flutter – uses https communication to transfer data between UI and VM

10.1 User Interfaces

Users will interact with SOS either through their Web Browser or on Android App via Flutter. The User Interface should be simple enough that any targeted user can use it without training.

10.2 Software Interfaces

SOS communicates with other components of the SOS system, this includes the UI on the user devices, the Data broker hosted in an Azure VM, and Cosmos DB also hosted in Azure.

10.3 Communications Interfaces

SOS users do not interact with each other. A user's device will need access to the Internet in order to reach Data Broker.

11. Applicable Standards

11.1 Usability Standards

SOS's targeted users are students. The design of SOS should be simple enough for targeted users to be able to be proficient without training. Instruction Dialogue may pop-up to guide new users.