Software Requirements Specifications

for

LibriSphere

Version 1.0 approved.

Prepared by Deepro Chakravorty and Srijit Sen,

Computer Science and Engineering, Section A.

October 28, 2024.

Table of Contents

Changelog

Name	Date	Reason	Version
Creation	July 28, 2024	Created the document	1.0

1. Introduction

1.1. Purpose

This Software Requirements Specification (SRS) will provide a description for the *LibriSphere* App. The SRS will provide a clear understanding of what is to be expected from the newly constructed *LibriSphere* App. The clear understanding and functionalities provided will allow for the correct software to be developed for the end user and also for future development of the app.

1.2. Document Conventions

LibriSphere will be referred to as 'the system' or 'the app' from now on.

1.3. Intended Audience and Reading Suggestions

The intended audience of this document are:

a. Software Developer
b. Software Tester
c. Technical Administrators
d. End users (Students)

The SRS will provide a clear idea about the system that is being built. Brief outline of the project is:

a. Overall Description
b. System Features
c. Functional Requirements
d. Non-Functional Requirements (if any)

1.4. Product Scope

LibriSphere aims to bring static textbook images to life by allowing users to visualize and explore content in a highly interactive 3D format. By displaying AR models that users can inspect from different angles, *LibriSphere* helps deepen understanding beyond traditional 2D representations.

1.5. References

https://developers.google.com/ar

https://docs.unity3d.com/Packages/com.unity.xr.arfoundation@5.1/manual/index.html

https://docs.unity3d.com/Manual/XR.html

2. Overall Description

2.1. Product Perspective

LibriSphere is an innovative AR application that enhances traditional learning by converting textbook images into detailed 3D models in augmented reality. The app is designed to be easy to use, accessible across different devices, and adaptable for various educational settings.

2.2. Product Functions

The features of this product include:

a. Image scanning for recognition of education content.
b. Rendering of AR 3D model aligned with scanned images.
c. Get information and knowledge on the model being spawned.

2.3. User Classes and Characteristics

The User Classes present in this SRS are:

a. Technical Administrator
b. Professor
c. Student

Characteristics of the User Classes:

a. Technical Administrator: Responsible for maintenance, updates, and ensuring smooth functioning.
b. Professor: Uses the app to assist students in visualizing complex concepts.
c. Student: Uses the app for interactive learning.

2.4. Operating Environment

The hardware requirements are:

a. Phone: 4GB RAM, dual core processor.
b. Storage: 250MB

The software requirements are:

a. Android 8 or iOS 13 or above.

2.5. Design and Implementation Constraints

LibriSphere requires AR compatibility in devices. The app is designed to deliver high-quality visualizations and may require sufficient processing power for seamless interaction with 3D models.

2.6. User Documentation

A user-friendly manual will be provided, detailing how to use *LibriSphere* effectively for educational purposes.

2.7. Assumptions and Dependencies

It is assumed that users have access to AR-compatible devices. The app depends on stable hardware specifications for optimal performance.

3. External Interface Requirements

3.1 User Interfaces

The following are the user interface requirements:

a. A start button to start the app.
b. A refresh button to delete the already spawned model and start scanning again.
c. A button to show and hide information about the model.
d. Text saying the name of the app.

3.2. Hardware Interfaces

The app requires a mobile device camera to capture images for AR model generation.

3.3. Software Interfaces

The app will operate within AR frameworks (ARCore for Android, ARKit for iOS). It also needs Android 8 or iOS 13 or above.

4. System Features (Functional Requirements)

4.1. Scan and Recognize Image Content

Input: Tap the screen to start the app, or the refresh button to bring it back to scanning mode.

Process: Scans the image being shown.

Output: Removes the scanning effect around the edge of the screen, signifying that the app has recognized the image.

4.2. Generate and Display AR Model

Input: Click refresh.

Process: Spawns the 3D model.

Output: Displays the 3D model in AR, hovering over the image just scanned.

4.3. Get More Information

Input: Click on the ‘eye’ icon in the bottom left corner.

Process: Fills in text according to the model spawned.

Output: A dialogue box opens, giving more information about the image.

5. Non-functional Requirements

5.1. Performance Requirements

The system must be interactive, and the delays involved must be fewer. In every action-response of the system, there should be no immediate delays. In the case of checking details, popping up error messages and saving the settings or sessions the delay should be below 2 seconds.

5.2. Safety Requirements

Images and personal data must be processed securely to protect user privacy.

5.3. Security Requirements

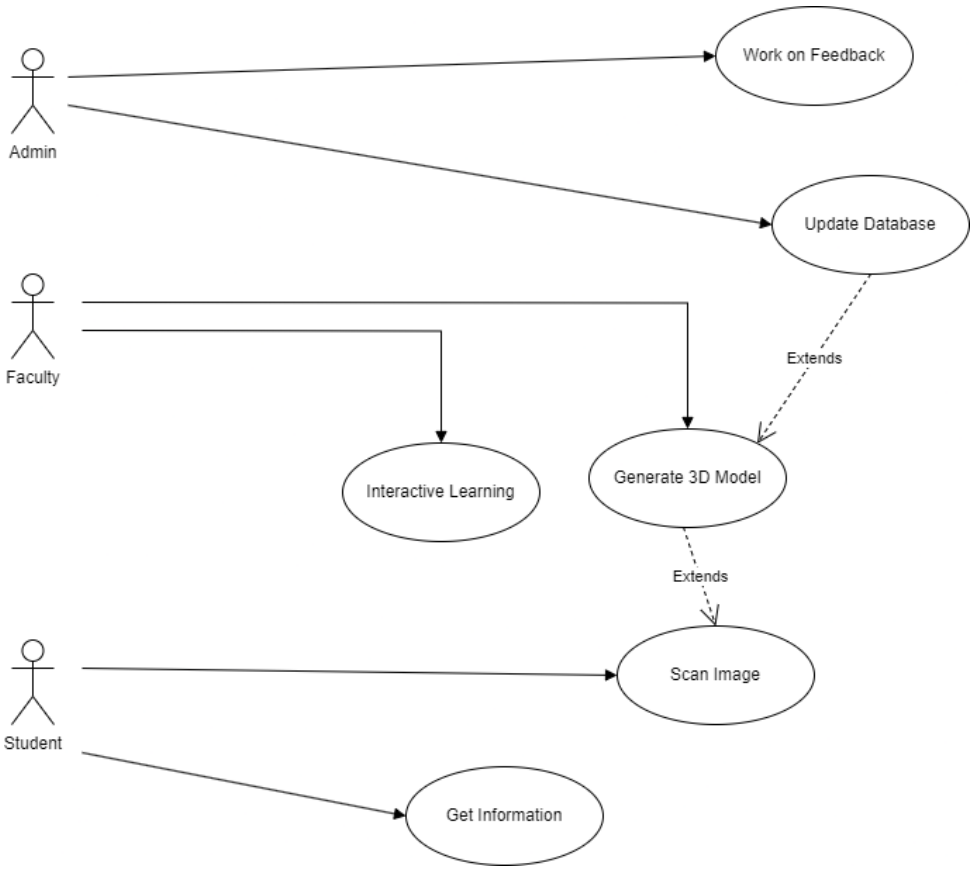
The app will use secure storage and transmission for any user data and model files.

5.4. Availability Requirements

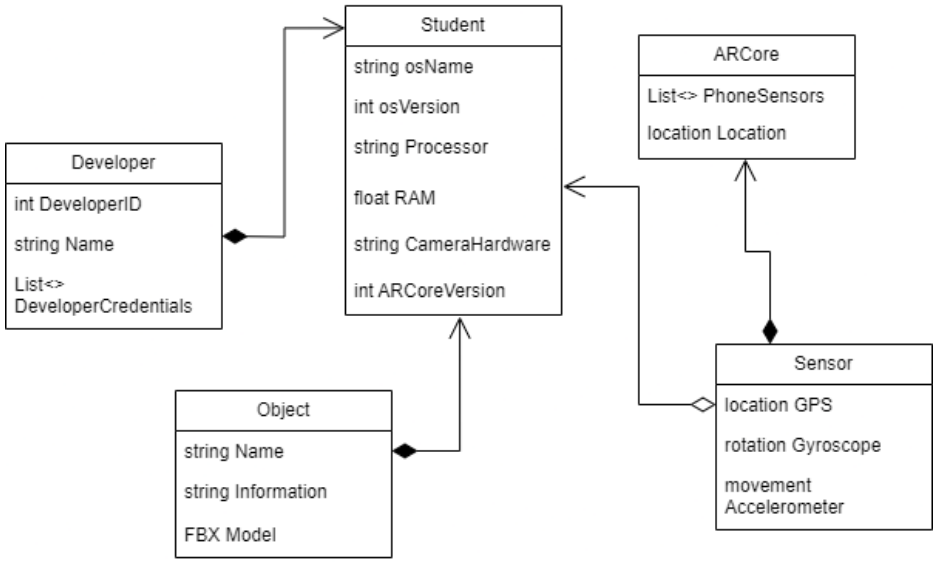
This system will be open 24/7. The users must be able to use the system anytime.

END OF SRS DOCUMENT. PTO FOR UML DIAGRAMS.

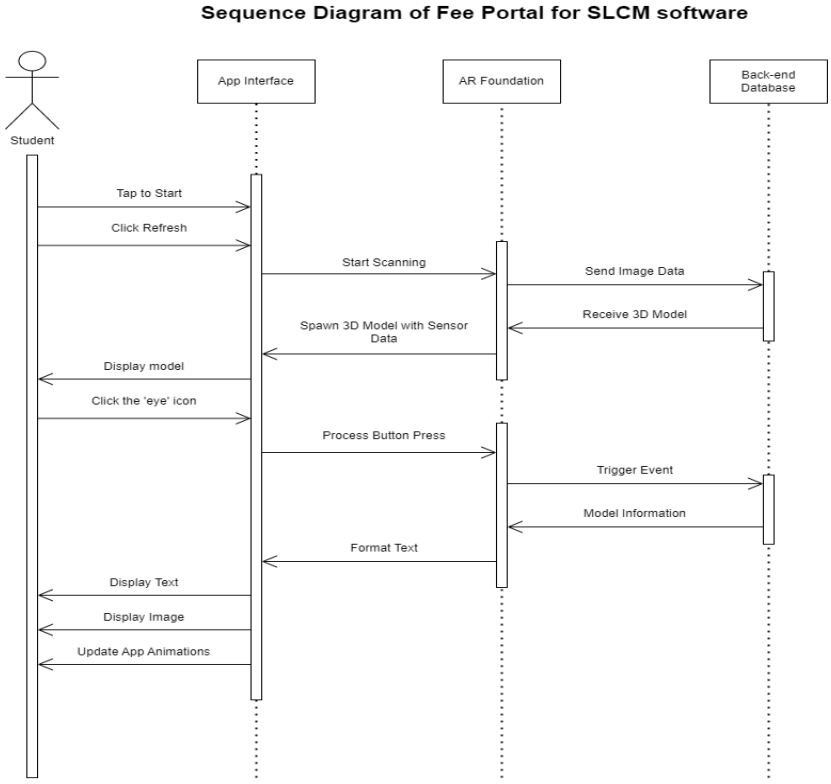
Use-Case Diagram



Class Diagram



Sequence Diagram



Activity Diagram