

# JPL - Moon Augmented Reality Design Specification

Group 2

December 2025

# Contents

<b>1</b>	<b>System Parts and Modules</b>	<b>2</b>
1.1	User Interface Module (UIM) . . . . .	2
1.2	Registration Module (RM) . . . . .	2
1.3	3D Model Module (3DM) . . . . .	3
<b>2</b>	<b>Directory Structure</b>	<b>3</b>
2.1	Client Directory . . . . .	3
2.2	Server Directory . . . . .	3
<b>3</b>	<b>Tools and Technologies</b>	<b>3</b>
3.1	Vue.js . . . . .	3
3.2	Express.js . . . . .	3
3.3	Python (process.py) . . . . .	4
3.4	MySQL . . . . .	4
3.5	JavaScript Cropping Utility . . . . .	4
3.6	Three.js . . . . .	4
3.7	NASA MoonTrek API . . . . .	4
<b>4</b>	<b>User Interface Summary</b>	<b>4</b>

This Design Specification provides a concise breakdown of all major parts, modules, tools, and components used in the MoonTrek: Telescope Augmented Reality project. It expands the Software Design Document (SDD) by describing each system part in slightly more detail.

# 1 System Parts and Modules

## 1.1 User Interface Module (UIM)

### Purpose

The UIM displays the web application's visual interface, handles user interactions, and forwards user inputs (such as uploaded images and metadata) to the backend.

### Components / Tools Used

- HomePage.vue
- UploadPage.vue
- ModelPage.vue
- ConnectPage.vue
- Supporting components: AnimatedStars.vue, ImageCanvas.vue, ImageUploadForm.vue, NavBar.vue

### Inputs and Outputs

- **Inputs:** user-uploaded images, metadata, navigation actions
- **Outputs:** overlays, 3D model display, telescope connection interface

## 1.2 Registration Module (RM)

### Purpose

The RM processes and registers the user's Moon image by comparing it with a reference image to generate a transformation matrix.

### Components / Tools Used

- SIFT feature detection
- Circle detection algorithm
- Transformation matrix computation

### Inputs and Outputs

- **Inputs:** cropped user image, reference Moon image
- **Outputs:** transformation matrix for overlay placement

## 1.3 3D Model Module (3DM)

### Purpose

The 3DM generates a photorealistic 3D model of the Moon, Earth, and Sun based on the user's location and image timestamp.

### Components / Tools Used

- Three.js rendering engine
- NASA lunar texture maps and reference datasets

### Inputs and Outputs

- **Inputs:** latitude, longitude, date, time metadata
- **Outputs:** rendered Moon reference image for registration

## 2 Directory Structure

As described in the SDD, the project consists of a `client` and `server` directory.

### 2.1 Client Directory

Contains all UI-related Vue components, styling, and route definitions.

### 2.2 Server Directory

Contains:

- `process.py` – Runs RM algorithms.
- Reference images used during image registration.
- Express.js endpoint handlers for uploads and metadata.

## 3 Tools and Technologies

This section provides a concise breakdown of every tool used, aligning with the SDD.

### 3.1 Vue.js

Used for UI structure, routing, and page components.

### 3.2 Express.js

Provides backend routing and communication logic between modules.

### **3.3 Python (process.py)**

Executes SIFT and related image-processing procedures.

### **3.4 MySQL**

Stores user-uploaded images and associated metadata.

### **3.5 JavaScript Cropping Utility**

Allows users to crop Moon images before processing.

### **3.6 Three.js**

Renders the 3D model of the Moon, Earth, and Sun.

### **3.7 NASA MoonTrek API**

Supplies scientific lunar data used for annotations and overlays.

## **4 User Interface Summary**

The application implicitly consists of:

- Home Screen
- Upload Screen
- 3D Model Screen
- Telescope Connection Screen

Each corresponds directly to a Vue component already listed under the UIM section. No additional UI details are necessary for this Design Spec.