Software Requirements Specification

for

Collaborative Visualization for Solar System Treks

Version 1.0 approved

**Prepared by Christopher Smallwood, Abdullah Alshebly
Stanley Do, Montague La France, Zipeng Guo, Johnny Lee, Jose Garcia,
Miguel Sanchez, Odasys Soberanes, David Tang**

Sponsor: NASA JPL

May 14, 2021

Table of Contents

Revision History

Name	Date	Reason for Changes	Version
Stanley Do	11/5/20	Added introduction Scope	0.1
Miguel Sanchez	11/5/20	Added Introduction, 1.1, 1.2	0.1
Stanley Do	11/6/20	Added Section 2.6 and 3.4	0.1
Jose Garcia	11/9/20	3.1 User Interfaces	0.1
Abdullah Alshebly	11/10/20	2.8 Adding Assumptions and Dependencies	0.1
David Tang	11/10/20	2.5 Operating Environment	0.1
Montague La France	11/13/20	2.1 System Analysis, 2.2 Product Perspective, 2.3 Product Functions	0.1
Odasys Soberanes	11/13/20	2.7 User Documentation	0.1
Johnny Lee	11/13/20	3.2 Hardware Interfaces	0.1
Christopher Smallwood	11/13/20	User Classes and Characteristics	0.1
Jose Garcia	11/15/20	5.4 Software Quality Attributes	0.1
Stanley Do	11/16/20	4.1 Functional Requirements	0.1
Stanley Do	11/19/20	4.3 Design Constraints	0.1
Miguel S	11/19/20	5.2 Safety	0.1
Miguel S	11/19/20	5.3 Security	0.1
Montague La France	11/19/20	4.3 Logical Database Requirements	0.1

Odasys Soberanes	11/19/20	Appendix A: Glossary	0.1
Johnny Lee	11/20/20	5.5 Business Rules	0.1
David Tang	11/20/20	Edited and revised Section 1.4, 2, 3, 4.3	0.1
Stanley Do	11/27/20	Revised Functional Requirements, added Appendix C	0.1
David Tang	12/6/20	Edit Section 3.2	0.1
Jose Garcia	12/6/20	Revised User Interface Section	0.1
Stanley Do	12/8/20	General Revisions (Whole doc)	0.1
David Tang	12/9/20	General revisions, updated table of contents, revised Section 4 and 5	0.1
Montague La France	12/10/20	General Revisions (Functional Requirements)	0.1
David Tang	12/10/20	Revised Data Constraints	0.1
Christopher Smallwood	12/11/20	Legal and Ethical Consideration	0.1
David Tang	5/8/21	Updated Sections 1 to 5. Marked requirements not met.	1.0
David Tang	5/9/21	Updated Table of Contents and its page numbers.	1.0
David Tang	5/14/21	Updated Section 3.3 and minor revisions.	1.0

1. Introduction

This document shall cover the requirements, implementation, and features of the Collaborative Visualization for Solar System Treks (CVSST), as well as a summary of the system. The requirements such as technologies, hardware, and software, shall also be discussed to give a comprehensive understanding of the collaborative functionalities designed on top of the system. CVSST shall add a collaborative visual aspect, where users shall be able to navigate through 3D terrain models and view satellite photography from various planets, asteroids, and moons together over the Internet.

1.1 Purpose

The purpose of CVSST is to integrate multi-user collaboration, visualization of markups and text, real-time communication, and state systems within Jet Propulsion Laboratory's (JPL) existing Solar System Treks (SST) software.

1.2 Intended Audience and Reading Suggestions

This document is intended for project managers, developers, students, the university faculty, and any other additional parties involved with the software's requirements process. This includes all members of the NASA JPL team, the primary development team, students, professors, faculty, and staff. The recommended reading sequence for this document is to start with the Overall Description followed by the specific section of the reader's topic of interest.

1.3 Product Scope

This software is identified as the Collaborative Visualization to the Solar System Treks application. CVSST shall provide collaborative markup of 3D solar system terrain such as the creation of waypoints, rapid navigation toward waypoints, text annotations, and freely drawn "ink" annotations on the existing Solar System Treks web portal. CVSST shall also provide networked "rooms" that let the user markup and share different states of 3D solar system terrain simultaneously, communicate via a text communications system, and create waypoints for rapid navigation. Upon release, CVSST shall be completely open to the public and be used for scientific research, mission planning, educational purposes, and general exploration.

1.4 Definitions, Acronyms, and Abbreviations

ACRONYM	**DEFINITION**
NASA	National Aeronautics and Space Administration
JPL	Jet Propulsion Laboratory
CVSST	Collaborative Visualization for Solar System Treks
SST	Solar System Treks
API	Application Programming Interface
GPU	Graphics Processing Unit
HTTP	Hypertext Transfer Protocol
HTML	Hypertext Markup Language
JSON	JavaScript Object Notation
VR	Virtual Reality
AR	Augmented Reality
OS	Operating System
WMTS	Web Map Tile Service
WSS	WebSocket Secure
SDD	Software Design Document
SSL	Secure Sockets Layer
UI	User Interface
GUI	Graphical User Interface

Collaborative Session
An online environment in which users can communicate, exchange data, and work with one another.

Session
Same as Collaborative Session.

Visuals
All visuals from the Solar System Trek software combined with the collaborative software’s visuals (such as markups, waypoints, and other UI elements).

Polyline
A connected sequence of line segments forming a single object.

State or SST State The current state of all planetary data such as the planet/moon, layers data, collaborative markups, and waypoints.

1.5 References

CSULA CS Department & Senior Design Faculty, SRSTemplateSeniorDesign-v2, Fall 2020.
SRS template provided by the CS department for this project.

Telescope Moon Trek Senior Design Team 2019, SRS_Fall_19, Fall 2019
SRS document written by another team from Senior Design Fall 2019.

2. Overall Description

2.1 System Analysis

1. Project Goals

1.1. The current SST software does not have a networked means of collaboration. Our goal is to provide networked collaboration in the form of markups while preserving web functionalities and access to as many people as possible. The collaborative markup shall include features such as creation of waypoints, rapid navigation to waypoints, text annotations, and freely drawn "ink" annotations. The terrain would be selected from the Solar System Treks web portal and imported into the collaborative visualization application.

2. Major Technical Hurdles

2.1. WebXR and VR

2.1.1. While VR has seen some client interest, WebXR's limited documentation and usability prevents us from accessing our desired range of users. The development team is also unable to meet in person to use and test VR equipment.

2.2. Performance

2.2.1. Due to the software being developed as a web application, there may be some performance issues when it comes to speed and latency. Real-time communication requires a fast internet connection, which may be an issue with users transferring a high amount of data.

3. Possible Solutions

3.1. WebXR and VR

3.1.1. We would have to create our own web-based VR API or use other existing libraries and integrate them into the SST software. We can also leverage faculty familiar with VR and online resources for testing VR with the HTC-Vive, Oculus, and Google Cardboard.

3.2. Performance

3.2.1. Limit the amount of data that is being transferred per user and use proper data files and formats to reduce slowdown.

2.2 Product Perspective

1. Relations to Other Products

1.1. The software system is a web application designed to seamlessly integrate into the current Solar Systems Trek system.

2. Existing Solar System Treks Interfaces

2.1. Virtual Reality

2.2. Data Searches

2.3. Layers

2.4. Downloads

2.5. GUI

2.5.1. Reset

2.5.2. Zoom out

2.5.3. Zoom in

2.5.4. Projections

2.5.5. Fly To

2.5.6. Basemaps

2.5.7. Latitude/Longitude

2.6. Explore Treks

2.6.1. Planets

2.6.1.1. Mars

2.6.1.2. Mercury

2.6.2. Moons

2.6.2.1. Icy Moons

2.6.2.2. Titan

2.6.3. Asteroids

2.6.3.1. Bennu

2.6.3.2. Ceres

2.6.3.3. Ryugu

2.6.3.4. Vesta

3. **Collaborative Visualization with Existing Solar System Treks Interfaces**
    - 3.1. Text communications medium.
    - 3.2. Extensive markup feature onto the 3D terrain.
    - 3.3. Extended "Fly To" features with rapid navigation waypoints.
    - 3.4. Control modules for session management.

2.3 Product Functions

1. **Explore Treks (Explore NASA's Solar System Trek data)**
    - 1.1. Planets
        - 1.1.1. Mars
        - 1.1.2. Mercury
    - 1.2. Moons
        - 1.2.1. Icy Moons
        - 1.2.2. Titan
    - 1.3. Asteroids
        - 1.3.1. Bennu
        - 1.3.2. Ceres
        - 1.3.3. Ryugu
        - 1.3.4. Vesta
2. **Virtual Reality (Navigate Solar System Trek data using VR)**
    - 2.1. HTC Vive
    - 2.2. Oculus
    - 2.3. Google Cardboard
3. **Data Searches (Search through a database and select the desired data)**
4. **Layers (View the selected layer in the database)**
5. **Downloads (Be able to download the selected data)**
6. **GUI**
    - 6.1. Reset (Reset back to default view)
    - 6.2. Zoom out (Zoom out of the data viewed)
    - 6.3. Zoom in (Zoom into the viewed data)
    - 6.4. Projections (Different viewing perspectives of the data)
        - 6.4.1. 3D Globe
        - 6.4.2. North Pole Map
        - 6.4.3. Global Map
        - 6.4.4. South Pole Map

6.5. Fly To (Go directly to selected data)
6.6. Basemaps (Display the current Basemap Layer)
6.7. Latitude/Longitude (Display Latitude and Longitude values)

7. **Sessions**
   7.1. Collaboration in session rooms
   7.2. Communication using a textbox
   7.3. List of people in the current session
   7.4. Diagram to be provided after further details are sorted

2.4 User Classes and Characteristics

The intended users for the Collaborative Visualization of Solar System Treks (CVSST) are: Researchers, Teachers, and Students.

2.4.1 Researchers

- Researchers have an anticipated use for scientific purposes.
- Researchers have a high-level of frequency of use.
- Researchers have a high-level of education.
- Researchers have a high-level of experience with the CVSST.
- It is necessary for Researchers to access and use the CVSST with the highest level of functionality that can be achieved to assist them with their research.
- This user class may include mission planners, managers, and scientists.

2.4.2 Teachers

- Teachers have an anticipated use for scholastic purposes.
- Teachers have a mid-level of frequency of use.
- Teachers have a high-level of education.
- Teachers have a mid-level of experience with the CVSST.
- It is necessary for Teachers to have basic functionality of the CVSST, as all the tools present in the CVSST may not be needed by a teacher.

2.4.3 Students

- Students have an anticipated use for scholastic purposes under the guidance of a Teacher.
- Students have a low-level of frequency of use.
- Students have a low-level of education.
- Students have a low-level of experience with the CVSST.

- It is necessary for Students to at least have a low-level of functionality with the CVSST, as Students shall be receiving directions from Teachers and therefore, shall not be needing full functionality.

2.5 Operating Environment

The operating environment shall require an Internet connection and a web browser. Since this software is a web application, it shall be served on a web server on-premise or the cloud, meaning that public-facing server infrastructure is required. The project's features shall become part of JPL's Solar System Trek.

2.6 Design and Implementation Constraints

- **Hardware Limitations –** The user's browser may require a significant amount of graphics rendering, which requires a more powerful GPU. This may lead to performance issues and hurt the user experience.
- **Software Limitations –** The software's markup functionality shall be implemented through the CesiumJS framework, which may create some issues for developers when implementing custom markups if it is not supported. While VR does provide a more immersive user experience, web-based VR may prove to be an extremely challenging requirement due to its poor browser support and lack of stable/usable technologies.

2.7 User Documentation

Documentation that shall be included with the software is:

- Project Plan (.xlsx)
- Personas Document (.pdf, .docx)
- Final Project Report (.pdf, .docx)
- Software Requirements Specification (.pdf, .docx)
- Software Design Document (.pdf, .docx)
- Fall 2020 Presentation Slides (.pdf, .pptx)
- Spring 2021 Presentation Slides (.pdf, .pptx)

2.8 Assumptions and Dependencies

2.8.1. Users are expected to have consistent and stable Internet connections to use this web application.
2.8.2. Users are expected to be familiar with an Internet browser and be familiar with handling the keyboard and mouse.

2.8.3. Users may need to install the WebXR Emulator extension which enables users to run WebXR content in desktop browsers without using a real XR device if VR is integrated. The extension shall emulate the WebXR API on browsers that do not support it.

2.9 Apportioning of Requirements

Requirements that might be delayed until future versions:

- AR and VR capabilities.

3. External Interface Requirements

3.1 User Interfaces

- The system shall provide intuitive input systems for users.
    - Some input systems may include simplified icons for buttons and sliders.
- The system shall provide a functionality where users can use text communication.
- The system shall provide a functionality where users can share SST together.
- The system shall provide a functionality where users can invite a people to a room session.
- The system shall provide a functionality where users can share changes made on the planet with another user.
- The system shall follow the color scheme, design, and theme of the underlying SST software.
- The system shall provide a functionality where users can save a session state of the planet.
- The system shall provide a functionality where users can annotate on the planet.
- The system shall provide a functionality where a user obtains administrative rules and decides what the participants do and not do.
- The system shall provide a functionality where users can search an area of the planet and navigate to that area.

3.2 Hardware Interfaces

There are no hardware interfaces that this software interacts with.

3.3 Software Interfaces

This software shall utilize various APIs included in the current Solar System Treks application such as the Search API and functions, Fly To function, CesiumJS Framework projections, layers and globe, WMTS API, current user interface, geospatial functions for distance, elevation profile, sun angle and distance, and 3D print functions. Due to this software being built upon existing software, communications between interfaces shall occur through JavaScript code and various implemented modules such as JSON parsers, database queries, or API responses.

- Some software interfaces include:
    - **Dojo Toolkit, https://dojotoolkit.org/**
        - Used for the frontend code and user interface.
        - Dojo Toolkit does not support ECMAScript 6 and higher, so functional notation cannot be used.
    - **Esri ArcGIS API for Javascript Version 3.X, https://developers.arcgis.com/javascript/3/jssamples/**
        - Used for 2D visualization.
    - **CesiumJS Framework projections, https://cesium.com/cesiumjs/**
        - Used for creating interactive web applications for sharing dynamic geospatial data.
        - Used for 3D visualization.
    - **WebGL, latest version found at, https://www.khronos.org/webgl/**
        - Used for rendering advanced inter-converting 3D and 2D graphics in any compatible web browser without using plug-ins.
    - **WebSockets, latest version found at https://socket.io/**
        - Used for transferring data from client to server in real-time and with low latency.
    - **Existing APIs in use in the existing Trek application.**
        - Used for transferring data from client to server in real-time and with low latency.

3.4 Communications Interfaces

The software shall require a web browser and use WSS over HTTPS, ensuring fully encrypted communication via SSL. The software shall also query data from JPL's APIs in the form of JSON and/or text/file format. While the WSS connection is active, synchronization between the client and web server shall be achieved with serialization and processing. Although WebSockets allow for a low-latency and full-duplex communication system, data transfer issues may arise when used in large-scale systems. The software system's backend web server shall also require a fast database, which shall assist in the prevention of network bottlenecks. This software shall also include a chatroom service, allowing users to communicate via text in real time.

4. Requirements Specification

4.1 Functional Requirements

4.1.1. The system shall allow users to host a "collaborative session" (See Section 1.4).

- **i.** The system shall reflect all visuals (See Section 1.4) to other users in a session as they are placed.
- **ii.** The system shall synchronize all visuals between users in a session unless configured otherwise.
- **iii.** The system shall share the same 2D/3D solar system terrain between users in a session.
- **iv.** The system shall display the same 2D/3D solar system terrain between users in a session.
- **v.** The system shall provide a text communications system for users in a session.
- **vi.** The system shall provide a voice-based communications system for users in a session.
- **vii.** The system shall allow users to invite other users to their session.

4.1.2. The system shall provide tools for markup onto 2D/3D Solar System terrain in varying sizes and colors.

- **i.** The system shall provide an "ink-based" annotation tool for free-hand drawing.
- **ii.** The system shall allow users to draw simple 2D shapes (squares, circles, triangles, etc.) on the 2D/3D terrain.
- **iii.** The system shall allow users to project 2D text onto the 2D/3D terrain.
- **iv.** The system shall allow users to create polyline (See Section 1.4) annotations.

4.1.3. The system shall allow users to create interactive navigation waypoints on 2D/3D Solar System terrain.

- **i.** The system shall allow users to move the navigation waypoints.
- **ii.** The system shall allow users to modify the navigation waypoints.

4.1.4. The system shall provide administrative powers to a session's host.

   - **i.** The system shall allow the host user to have control of all other users' displays in a session.
   - **ii.** The system shall allow the host user to transfer host privileges to other users' displays in a session.
   - **iii.** The system shall allow the host user to prevent collaborative markups from specific users in a session.
   - **iv.** The system shall allow the host user to grant other users permissions to all collaborative tools.
   - **v.** The host shall have the ability to disable use of specific tools by others.
   - **vi.** The host shall have the ability to give permission to the user to use specific tools.

4.1.5. The system shall allow users to create multiple "SST states" (See Section 1.4) within a session.

   - **i.** The system shall allow users to access different states within a session.
   - **ii.** The system shall reflect all changes made within each SST state.

4.1.6. The system shall allow users to save their current "SST state".

   - **i.** The system shall allow the user to extract all markup to be saved for later use.
   - **ii.** The system shall allow the user to extract layer data to be saved for later use.
   - **iii.** The system shall allow the user to import saved markup to be visualized onto SST.
   - **iv.** The system shall allow the user to import saved layer data to be visualized onto SST.

4.2 External Interface Requirements

- The system shall allow users to input information through supported hardware mediums. Some hardware mediums include:
  - Mobile device touch screens.
  - PC browser mouse input:
    - Click events
    - Mouse movement
    - Scrolling

    - PC browser keyboard events.
        - Key press events

4.3 Logical Database Requirements

- Types of information used by various functions
    - HTTP/ JSON Requests
    - WMTS map data
    - Model Layer Data
    - Mesh data
    - Coordinate data
    - Text data
    - Session Data
- Frequency of use
    - Data pulled based on every host request of the session
- Accessing capabilities
    - Database access based on the current session host and members
- Data entities and their relationships
    - Solr
    - ArcGIS
        - Accumulo
    - Oracle
        - Postgres
- Integrity constraints
    - Database must support NASA’s Trek Data
    - Current Session Memory size
    - Security
- Data retention requirements
    - Data shall be saved temporarily during each session
    - The system shall allow users to revisit previously used data from the current session

4.4 Design Constraints

With the use of certain frameworks in the current Solar System Trek software, issues may arise when tackling networked sessions or free-hand “ink” annotations. It is also unknown how much network traffic shall be utilized concurrently between the server and client and how much server resources may be used in processing requests and other data. This brings up the possibility of the software not being very scalable. While a small number of entities and annotations on the 3D solar system terrain might not be a problem, larger amounts of entities may cause issues for devices with low graphics processing power.

5. Other Nonfunctional Requirements

5.1 Performance Requirements

Some performance requirements that the application shall have:

- The system shall transfer data from client to server in real-time.
- The system shall transfer data with low latency.
- The system shall run smoothly in real time such that there is no noticeable performance or rendering issues.

5.2 Safety Requirements

There are no safety requirements that need to be taken into consideration when using the CVSST.

5.3 Security Requirements

Every user's identity and private data shall always be protected. Hosts shall be authenticated upon request, in order to grant larger freedoms within the application.

5.4 Software Quality Attributes

- Adaptability: Currently has user collaboration, no virtuality reality.
- Availability: It is accessible through a website.
- Correctness: Shows the planet from NASA Trek API.
- Flexibility: It has features for Researchers, Teachers, and Students.
- Maintainability: Grabs data from the NASA Trek API.
- Portability: Can support Desktop, Laptop, and Mobile devices.
- Usability: Simple User Interface for all users

5.5 Business Rules

The product would not be restricted to being used by a specific person.

6. Legal and Ethical Considerations

The primary concern for ethics regarding the Collaborative Visualization for Solar System Treks (CVSST) originates from the Association for Computing Machinery (ACM) Code of Ethics.

As the CVSST shall be a modified version of the existing Solar System Treks (SST), care must be taken to preserve functionality from the original SST. The CVSST shall be the SST with a collaborative environment. To achieve this, it is essential that the CVSST maintains full functionality from the SST. Failure to achieve full functionality of the SST within the CVSST would not ensure employing proper care for modifying systems as functionality would be lost when the objective is to only gain functionality.

As the CVSST shall be online, care must be taken to ensure that users can only access information that is necessary within the environment. With the CVSST system being online, it is essential that users are only able to access information that is deemed necessary by the computing professional. Failure to achieve this would be both irresponsible and unadvised as it may result in potential information leaks.

As the CVSST shall be online, care must be taken to ensure that only essential information is collected from users to ensure proper usability of the CVSST, while maintaining the standards of data collection and privacy set by the Web Privacy Policy for NASA . With the CVSST system being online, it is essential that the system only collects information from the users that is necessary for the functionality of the CVSST. Failure to achieve this may disrespect user privacy and would not benefit the user, NASA/JPL, or the CVSST.

As the CVSST shall be within a collaborative environment, care must be taken to ensure bad faith actors cannot be granted access to collaborative environments the host did not intend for them to attend. With the CVSST system being online, it is essential that there are barriers of entry to deny bad faith actors entry into a collaborative session they were not intended to access. Failure to achieve this may potentially result in information leaks and the bad faith actor may actively sabotage the collaborative environment of the users.

As the CVSST shall be online, care must be taken to ensure the safety of user and host information. With the CVSST being online, it is essential to take the necessary precautions to protect information stored by the users and NASA/JPL. As the CVSST is online it would be irresponsible to enable unintended users to potentially access restricted information.

Appendix A: Glossary

- SDD:
  Software Design Document

- SRS:
  Software Requirements Specifications

- UI:
  Abbreviation for User Interface. The UI is also known as the frontend of the application and is the part of the system that the user interacts with to complete tasks.

- HTTP:
  Hypertext Transfer Protocol is an application protocol for distributed, collaborative, hypermedia information systems.

- HTML:
  Hypertext Markup Language is the standard markup language for creating web pages.

- Application program interface (API):
  Functions or methods for accessing software services or libraries.

- Augmented Reality (AR):
  A technique in computer graphics that superimposes (places) a computer generated object into a device's camera view to alter the perception of the real world.

- Virtual Reality (VR):
  Virtual reality is a simulated experience that can be similar to or completely different from the real world.

- Operating System (OS):
  The software that allows any computer to communicate, modify, and terminate any hardware and software communications based on end-user decisions.

- JSON:
  Abbreviation for JavaScript Object Notation. Objects that are often used to return data

from APIs back to the front.

- JavaScript:
  A programming language that is heavily used for web applications.

- WMTS:
  A Web Map Tile Service is a standard protocol for serving pre-rendered or run-time computed georeferenced map tiles over the Internet.

- CesiumJS:
  CesiumJS is an open source JavaScript library for world-class 3D mapping.

Appendix B: Analysis Models

No analysis models.

Appendix C: To Be Determined List

No pending references to be determined.