T02 Group 5 Deliverable #1

SE 3A04: Software Design II – Large System Design

February 13, 2017

1 Introduction

The following is a description of the product to be developed, as well as an overview of the SRS.

1.1 Purpose

SpaceBase Ephemeris (SBE) is a mobile simulation game that models the overall behavior of a space settlement on a celestial body, i.e. a real-life system. The game represents how different sub-systems interact with each other and affect the overall system. It replicates how each of the sub-systems of a space base react to different stimuli from outside of the system. Therefore, the player must ensure that the sub-systems are working in the desired fashion to keep the base operating. The purpose of this document is to provide a general description of this project and to specify the requirements for this game. It is meant to be a form of communication between the developers of this game and its clients, i.e. Dr. Ridha Khedri and the teaching assistants for SFWR ENG 3A04. The SRS is also meant to be used as a reference by the developers to ensure that the specified requirements have been fulfilled.

1.2 Scope

The focus of this SRS is the development of the software product, SBE. The game mimics the operations of a space settlement, and places the user as the head of the base. Hence, the key duty of the player is to ensure that the base remains operating. The player does this by keeping an eye on all the sub-systems and maintaining them whenever there is a need. This will be done by assigning tasks to all the members of the base, which could be anything from fixing a breach in the walls of the base to interaction with alien flora and fauna. If there are multiple stimuli happening concurrently, then the player must prioritize the tasks, based on how critical they are and on what level they affect the safety of the community. The main objective of this game is to provide entertainment to its users. However, it will also develop time management skills of the users, as well as improve their multitasking abilities.

1.3 Definitions, Acronyms, and Abbreviations

1. SBE: SpaceBase Ephemeris, the title of the game

1.4 References

No references were used for this document.

1.5 Overview

This document describes the product software that is to be developed as well as the requirements specifications.

- The second section of this document provides an overall description of the game. It provides product perspective when compared to another related product. It describes major functions of the product as well as the user characteristics. It also explains the constraints of the system as well as the assumptions for this project.

- The third section of this document specifies the functional software requirements of this game. It provides sufficient details to design a system with the specific requirements and to test that the design has fulfilled those requirements.
- The final section of the document is the non-functional requirements, which vary from different perceptions and are used to ensure successful integration of the game into society.
- A Division of Labour section is also placed at the end of the document which details the contributions of each team member.

2 Overall Description

The following is a general description of the product and its requirements. For more specific requirements, see the requirements sections.

2.1 Product Perspective

SBE is an independent and totally self-contained system. It does not require network communication to function normally. The game will contain elements of various existing simulation games such as [[dwarf fortress]], but is not intended to completely emulate any of them. The simulation is intended for entertainment purposes only, so bears only a thematic relationship to some scientific simulation software. As the game is an Android app, it may be distributed on the Google Play Store, but that does not constitute part of the system.

2.2 Product Functions

1. The user will be able to view compositions of various sub-views of the system. Each sub-view corresponds to a different subsystem.
2. The user will be able to stimulate the system. Each stimuli will be able to one or many subsystems, with reactions cascading appropriately. Each subsystem will have at least 1 stimuli.
3. The application will simulate the various subsystems and their interactions.
4. Major ways the user will be able to interact with the system include: expanding their station, issuing orders to the population, and managing power and atmospheric controls of their station.
5. **Creative/Innovative:** The user will be able to save the state of their game, and share the corresponding file with other users, who will in turn be able to load it.

2.3 User Characteristics

Users are expected to have at least a high school level diploma and reading level. Users are expected to be generally familiar with the Android operating system and Android apps. Users are expected to have only a cursory understanding of the subject matter, as SBE is intended for entertainment rather than scientific simulation.

2.4 Constraints

The following are constraints on the development of the system.

1. The system must be produced as an Android app.
2. The system must consist of several separate subsystems. (at least 3)

2.5 Assumptions and Dependencies

The following are assumptions that affect the requirements for the system.

1. It is assumed that the device running the application will have the Android operating system available.
2. It is assumed that the application will be run with sufficient privileges to read and write necessary files on the device.
3. It is assumed that the device will have access to the Google Play Store (or an alternate distribution method if one is chosen).

2.6 Apportioning of Requirements

The following requirements may be delayed until future versions of the system.

1. Functionality allowing the user to showcase their system (for example on social media).
2. Functionality allowing the user to swap between multiple saved states within the application.
3. Non-token graphical features.
4. More than the minimum number of subsystems.
5. Ability to enable or disable subsystems at runtime.

3 Functional Requirements

1. The user launches the application
   (a) Android Device OS
      i. The system launcher starts the application
   (b) User
      i. The system provides a view of all the sub–systems in their present state to the user
      ii. The system should allow the user to select a sub-system to interact with
2. The user wants to interact with a sub-system
   (a) User
      i. The system should show all stimuli that can stimulate the sub–system
      ii. The system should allow the user to select a stimuli
      iii. The system must allow the user to control the stimuli
      iv. The system must respond to the stimuli
3. The application is updated
   (a) Android Device OS
      i. The system must notify the operating system that an update is required for the system
   (b) User
      i. The system must prompt the user to update the system when the application is launched
4. Time passes within the system
   (a) User
      i. The system should update the attributes that have changes since the last time period
      ii. The system should show the user the result of the stimuli from the previous time period

5. The user wants to change settings of the system
    (a) User
        i. The system should provide an interface for editing values of how the system works based on customizable aspects of the system
        ii. The system should store these new settings and incorporate them into the system's functionality
6. The user wants to share snapshot of the system
    (a) Android Device OS
        i. The system should be able to use the OS to interact with social media applications to send data from system to external applications
    (b) Social Media Application
        i. The system should be able to provide data that is compatible with the specified social media framework
    (c) User
        i. The system should provide a means of sharing user's system data via social media
7. The user wants to shut down the system
    (a) Android Device OS
        i. The operating system should close the application
    (b) User
        i. The system should provide a way to close the current running of the system's processes
        ii. The system should provide a way to save progress within the system and sub–systems

4 Non-Functional Requirements

4.1 Look and Feel Requirements

4.1.1 Appearance Requirements

1. LF1. The application must have graphical interface (not just textual) for the users to use.
2. LF2. The graphical interface must be spannable, zoomable, and rotatable.
3. LF3. The interface must be isometric or top-down view.
4. LF4. The product shall be attractive to all audiences.

4.1.2 Style Requirements

1. S1. N.A.

4.2 Usability and Humanity Requirements

4.2.1 Ease of Use Requirements

1. UH1. The application shall be easy for anybody of age 13 and up to use.
2. UH2. The application shall include a brief tutorial or manual to guide new users.
3. UH3. The application shall introduce game mechanics step-by-step so the users are not overwhelmed.
4. UH4. The application shall be used by anyone, by substituting symbols with language.

4.2.2 Personalization and Internalization Requirements

1. UH1. The application shall create memory space in the device to store data.
2. UH2. There will only be English option, however everything shall be understood through symbolic icons.
3. UH3. Personal settings or game files shall be able to be saved, and loaded at users will.

4.2.3 Learning Requirements

1. LER1. This application shall be easy for anyone over 13 to learn.
2. LER2. The application will provide a short learning tutorial to ease of any confusion.

4.2.4 Understandability and Politeness Requirements

1. UPR1. The application must use symbols and words that are naturally understood.
2. UPR2. Any abbreviations or acronyms must be easily looked up or understood.

4.2.5 Accessibility Requirements

1. ACCR1. This application shall conform to the Canadians with Disabilities Act.

4.3 Performance Requirements

4.3.1 Speed and Latency Requirements

1. SOLR1. Any interaction between the user and the system must have a maximum response time of 1 second.
2. SOLR2. Any interaction relating to changing the view of the system must have an instant response time.

4.3.2 Safety-Critical Requirements

1. SCR1. Any colour scheme must not induce dizziness, tiredness, or seizures.
2. SCR2. The graphics must be low enough to not overheat the GPU.

4.3.3 Precision or Accuracy Requirements

1. PAR1. All monetary values must be to two decimal places.
2. PAR2. Any other critical values the system depends on must be to six significant digits.

4.3.4 Reliability and Availability Requirements

1. PR1. The application must not crash before saving user progress and data.
2. PR2. No data must be lost even in case of system failure.
3. PR3. This application must be available 24 hours per day, 365 days per year.

4.3.5 Robust or Fault-Tolerance Requirements

1. RFTR1. N.A.

4.3.6 Capacity Requirements

1. CAPR1. The application shall be at maximum 300MB.

4.3.7 Scalability or Extensibility Requirements

1. SER1. N.A.

4.3.8 Longevity Requirements

1. LR1. The application is expected to operate as long as the hardware supports it.

4.4 Operational and Environmental Requirements

4.4.1 Expected Physical Environment

1. EP1. N.A.

4.4.2 Requirements for Interfacing with Adjacent Systems

1. RIAS1. This application must work on at least the last four generations of smartphones.

4.4.3 Productization Requirements

1. PRR1. This application shall be distributed as APK file.
2. PRR2. The application must be available on Android Play Store.

4.4.4 Release Requirements

1. RR1. The application must fix any bugs that are brought forward by its customers.
2. RR2. There must be a new release every other month, at minimum.

4.5 Maintainability and Support Requirements

4.5.1 Maintenance Requirements

1. MSM1. A new Module Interface Specification must be made within a week of the date when the requirements are agreed upon.
2. MSM2. Any dependent system must be able to be identified through all of its independent systems in order to solve any bugs.

4.5.2 Supportability Requirements

1. MSS1. An email system where users can interact with the developing team must be set up.

4.5.3 Adaptability Requirements

1. ADR1. This application is to run under Android.
2. ADR2. The application may eventually be ported to Windows, Mac, Linux, or iOS.

4.6 Security Requirements

4.6.1 Access Requirements

1. SAR1. N.A.

4.6.2 Integrity Requirements

1. IR1. The application shall ensure that any attempts to breach any personal data be deterred.
2. IR2. The application will make sure that any stored data is correct to desired system.

4.6.3 Privacy Requirements

1. PR1. The application shall not collect any personal information.
2. PR2. The application will notify the user if any agreement policy changes.

4.6.4 Audit Requirements

1. AR1. N.A.

4.6.5 Immunity Requirements

1. IMR1. N.A.

4.7 Cultural and Political Requirements

4.7.1 Cultural Requirements

1. CR1. This application shall not contain any symbols or messages that may offend any ethnic or religious groups.
2. CR2. The application must properly distinguish characters from different origins.

4.7.2 Political Requirements

1. PP3. N.A.

4.8 Legal Requirements

4.8.1 Compliance Requirements

1. LCR1. N.A.

4.8.2 Standards Requirements

1. LSR1. N.A.

A Division of Labour

1. Arfa: Introduction
2. Ian: Overall Description
3. Areeb: Functional Requirements
4. Nishanth: Functional Requirements
5. Steven: Nonfunctional Requirements