Software Requirements Specifications

Version 1.0,

prepared by TheFlippedCompany

Table of Content

1. Introduction

1.1 *Overview*

The OnGo is an application that provides you access to all the files stored on your personal computer from *any device* you want.

This document provides informations on the requirements for the OnGo application.

Project goals, scope and definitions are given in the *introduction*.

Design constraints and *application environment* are described in the following section.

Nonfunctional requirements are outlined for later verification.

Functional requirements are given to show the system features and expected user interaction.

The *Software Project Management Plan* will give specifics on project budget and schedule.

1.2 *Goals and Objectives*

The main objective of this project is to allow users to have an easy and quick access to any file stored on other devices. The OnGo application is expected to:

- function in an simple and intuitive manner
- provide an easy access to the user's files without uploading them first, as cloud services require to.
- work on any operative system and device

1.3 *Definitions*

Origin - It's the computer on which files are *actually* stored, your "*home pc*" if you want to.

User - It's the person that's trying to get to the *origin* via OnGo

2. General Design Constrains

2.1 *Application Environment*

The OnGo product includes:

- A desktop application of our design that has to be installed on the *origin* system
- A web application of our design that is used by the *user* to interact with his own registered *origin* system

Both applications interact with our Web Api that will handle the communications between the user and the corresponding origin.

2.2 *User Characteristics*

Users will need to have a valid account in order to download the OnGo origin program and they will also need to use their accounts to log into the user web application in order to use it. There is *no limitation* on the device that the user can use.

2.3 *Mandated Constrains*

The whole system has to work via HTTP connection (https). This is mandatory because the *origin* machine will probably be inside a private network and it might not be allowed to host a server accessible from the public internet due to several security issues or privacy policies.

3. Nonfunctional Requirements

3.1 *Operational Requirements*

The user web application has to provide an easy to use interface and should be simple enough to let every user connect to it's own origin without any serious problems.

The origin main application has to generate and give the code to the user in the most clear way possible. The user will still need to remember that code by himself.

3.2 *Performance Requirements*

When a file is *downloaded* / *uploaded*, the system has to deliver the whole document without losing any part regardeless of the file dimension.

3.3 *Security Requirements*

The first versions of the OnGo application will only require:

- User login and registration (to download the origin application and to use the user web app)

3.4 *Documentation and Training*

The OnGo application *itself* will not include any documentation or training for the final user but video tutorials will be available on the team website and on the official YouTube channel.

3.5 *External Interface*

3.5.1 *User Interface*

The interface will be *eye-catching* in order to let the user interact with a *familiar environment*.

The interface will be easy to use on any device since it will resemble a *Windows-like operating system UI*.

3.5.2 *Software Interface*

The OnGo Api works as an interface between the origin application and the user application. It handles and direct all the communications between the *users* and their corresponding *origin* system.

4. Functional Requirements

4.1 *Required Features*

4.1.1 *Use Case 1*

Description: Create a new folder and navigate inside it

Actors: OnGo *User*

Basic path *origin*:

1. The user opens the *OnGo Desktop Application*
2. The user copies the generated code
3. *The user can now decide when to start working on his other devices*

Basic path *user*:

1. The user goes to our website
2. The user clicks on the *OnGo Application Icon*
3. The user *logs into* his account
4. The user prompts the *previously given* code
5. The system provides an *accurate view* of the user's origin system
6. The user navigates to the desired folders by *double-clicking* on them
7. To create a new folder, the user *right-clicks* on the page and clicks on "*New folder*"
8. The user decides a proper name and then clicks *create*
9. A new folder has just been created on the origin system

4.1.2 *Use Case 2*

Description: Transfer a file from the User to the Origin

Actors: OnGo *User*

Basic path *origin*:

1. The user opens the *OnGo Desktop Application*
2. The user copies the generated code
3. *The user can now decide when to start working on his other devices*

Basic path *user*:

1. The user goes to our website
2. The user clicks on the *OnGo Application Icon*
3. The user *logs into* his account
4. The user prompts the *previously given* code
5. The system provides an *accurate view* of the user's origin system
6. The user navigates to the desired folders by *double-clicking* on them
7. To transfer a document, the user *drags and drops* the desired file from the local system to the web application
8. A new file has just been transfered to the origin system

4.1.2 *Use Case 3*

Description: Require a file located in the Origin as a User

Actors: OnGo *User*

Basic path *origin*:

1. The user opens the *OnGo Desktop Application*
2. The user copies the generated code
3. *The user can now decide when to start working on his other devices*

Basic path *user*:

1. The user goes to our website
2. The user clicks on the *OnGo Application Icon*
3. The user *logs into* his account
4. The user prompts the *previously given* code
5. The system provides an *accurate view* of the user's origin system
6. The user navigates to the desired folders by *double-clicking* on them
7. The user, to require a file, *right-clicks* on the desired document and clicks on "*Download*"
8. After a few seconds, the required file is downloaded on the user's system

4.2 *Optional Features*

4.2.1 *Use Case 1*

Description: Send a full proper command through a DOS emulator. Computer shutdown from CLI

Actors: OnGo *User*

Basic path *origin*:

1. The user opens the *OnGo Desktop Application*
2. The user copies the generated code
3. *The user can now decide when to start working on his other devices*

Basic path *user*:

1. The user goes to our website
2. The user clicks on the *OnGo Application Icon*
3. The user *logs into* his account
4. The user prompts the *previously given* code
5. The system provides an *accurate view* of the user's origin system
6. The user clicks on the *CMD* icon on the top-left corner
7. A CLI will appear in the view and the user will be able to send proper commands
8. To shut the system down in 60 seconds, the user types *sudo shutdown -s -t 60*
9. The system will be shut down and the connection will be closed

4.2.2 *Use Case 2*

Description: Add a new file with content in it

Actors: OnGo *User*

Basic path *origin*:

1. The user opens the *OnGo Desktop Application*
2. The user copies the generated code
3. *The user can now decide when to start working on his other devices*

Basic path *user*:

1. The user goes to our website
2. The user clicks on the *OnGo Application Icon*
3. The user *logs into* his account
4. The user prompts the *previously given* code
5. The system provides an *accurate view* of the user's origin system
6. The user navigates to the desired folders by *double-clicking* on them
7. To create a new file, the user clicks on the *Notepad* icon on the top-left corner
8. When the web editor loads up, the user will be able to type and save the document in the desired format by clicking the "*save*" button
9. The user will then type the desired name and extension in the modal
10. A new file with content *has been added* to the origin system in the desired path