Case Study 2 – User Requirement Specifications

Media storage service



Project Members & Student Numbers:

* Francisco Marcó (4467752)
* Sava Vasilev (4663438)
* Kaloyan Andreev (4408020)

Elaboration Date: 16/05/2022

Group 10

Table of Content

[1. Introduction 2](#_Toc105927618)

[2. Functional Requirements 3](#_Toc105927619)

[3. Non-functional Requirements 4](#_Toc105927620)

[4. Website Wireframe: 5](#_Toc105927621)

[4.1. Login page 5](#_Toc105927622)

[4.2. Welcome Page 5](#_Toc105927623)

[4.3. Right-click file 5](#_Toc105927624)

[4.4. Clicked on “Get Info”: 6](#_Toc105927625)

[4.5. Clicked on “Delete” 6](#_Toc105927626)

[4.6. Clicked on “Rename” 6](#_Toc105927627)

[4.7. Clicked on “Send to” 7](#_Toc105927628)

[4.8. Clicked on “Upload Button”: 7](#_Toc105927629)

[4.9. Selected file after clicking on “Upload” button: 7](#_Toc105927630)

[5. MoSCoW table 8](#_Toc105927631)

# Introduction

This document aims to explain the User Requirement Specifications for our Case Study Project. It will go into detail about the functional requirements, non-functional requirements, the website wireframe, and MoSCoW table. It will also showcase a C4 diagram of our project.

Our product is aimed at large scale media companies, who seek to manage their media files through our software solution. Through our webpage interface, we provide the ability to upload, delete, rename, get information, and send files using a secure and encrypted protocol. All the files are stored in our own Server, inside of a Database. All the file synchronization between machines is done using SyncThing.

# Functional Requirements

Our solution offers the following to the user:

* **Upload, Delete and send files to other users in the network.**
* **Manage and view the files using the web application.**
* **Reduce storage costs by storing all the files remotely.**
* **Synchronize files between the server and a local machine.**

All of these will be available only when the User connects to the network through VPN/RADIUS.

# Non-functional Requirements

Our solution has:

* **Security: By use of SyncThing and other security protocols we ensure that the data is always kept safe and is compliant to various guidelines (GDPR, CIA, and more)**
* **Scalability: Quickly scale operations by easily adding more users, managing files divided by folders called [Workspaces]**
* **Usability: By click of a button, and an easy-to-use GUI, the User can manage their files and synchronize it real-time with the server**
* **Portability: Our solution is platform-agnostic, meaning it will run on any platform the user desires (as long as it’s connected to our network)**

# Website Wireframe:

## Interfaz de usuario gráfica, Aplicación Descripción generada automáticamenteLogin page

## Interfaz de usuario gráfica, Aplicación Descripción generada automáticamenteWelcome Page

## Interfaz de usuario gráfica Descripción generada automáticamente con confianza mediaRight-click file

## Interfaz de usuario gráfica Descripción generada automáticamenteClicked on “Get Info”:

## Interfaz de usuario gráfica, Texto, Aplicación Descripción generada automáticamenteClicked on “Delete”

## Interfaz de usuario gráfica, Aplicación Descripción generada automáticamenteClicked on “Rename”

## Interfaz de usuario gráfica Descripción generada automáticamenteClicked on “Send to”

## Interfaz de usuario gráfica, Texto, Aplicación Descripción generada automáticamenteClicked on “Upload Button”:

## Interfaz de usuario gráfica, Aplicación, Teams Descripción generada automáticamenteSelected file after clicking on “Upload” button:

# Texto Descripción generada automáticamenteMoSCoW table