Software Requirements Specification

for

MAPSTER

**Version 1.0 Approved**

**Prepared by Chris Duhan**

**Team 2**

**6 November 2018**

**Table of Contents**

**Table of Contents ii**

**Revision History ii**

**1. Introduction** 3

1.1 Purpose 3

1.2 Intended Audience and Reading Suggestions 3

1.3 Product Scope 3

**2. Overall Description** 3

2.1 Product Perspective 3

2.2 Product Functions 3

2.3 User Classes and Characteristics 3

2.4 Operating Environment 4

2.5 Design and Implementation Constraints 4

2.6 User Documentation 4

**3. External Interface Requirements** 4

3.1 User Interfaces 4

3.2 Hardware Interfaces 4

3.3 Software Interfaces 4

3.4 Communications Interfaces 4

**4. System Features** 5

4.1 Connect to Friends 5

4.2 Take and Upload Images 5

4.3 View Locally Relevant Images 5

4.4 Message Friends and Groups 5

**5. Other Nonfunctional Requirements** 5

5.1 Security Requirements 5

**Appendix A: Glossary** 6

**Appendix B: Analysis Models** 6

**Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Date** | **Reason For Changes** | **Version** |
| Chris Duhan | 11/6/18 | Initial document | 1.0 |
|  |  |  |  |

# Introduction

## Purpose

The purpose of this document is to give a detailed description of the requirements for the “MAPSTER” software. It will illustrate the purpose and the complete declaration for the development of the system. It will also explain system constraints, interface, and interactions with other external applications. This document is primarily intended to be a reference for developing the first version of the system for the development team.

## Intended Audience and Reading Suggestions

This document is intended for the use of the developers and those who want to gain further knowledge of the application, from the standpoint of a user. It is written to give in a in depth description on the background development of the application such.

## Product Scope

Mapster is a social networking application that allows users to share their location with a community of friends using pictures as the primary source of interaction (Imagine an Instagram and Snapchat offspring). Users will have the ability to search users or photos within a certain radius of their location in order to expand their Mapster community. Users may add to their profile by posting their own images or geolocation, all of which are viewable to those the users allow.

# Overall Description

## Product Perspective

This app is to be a new, self-contained intellectual property. It may make use of existing technologies including but not limited to, Google Firebase, Google Firestore, and Ionic Cordova.

## Product Functions

The app provides a secure login, access to the users personal account, the ability to take and upload pictures and metadata, the ability to connect to other users, the ability to engage in conversation with other users and the ability to see the locations of other users they have connected to.

## User Classes and Characteristics

This app is intended to be used by any individual of a certain age. Its functionality may be greater to highly active people, but those individuals do not necessarily make up the entire user base.

## Operating Environment

The app can utilize any iPhone, Android or Windows phone operating in their most resent respective versions.

## Design and Implementation Constraints

The design and functionality of the app will be limited to the abilities of the Ionic Framework and the hardware that it supports.

## User Documentation

Apart from this document there is not and will be not be any additional documentation provided.

# External Interface Requirements

## User Interfaces

The user must first register with the service via a registration page. If already registered they will instead be greeted by a login page. After logging in they have access to the full range of implemented features, in a style that has yet to be determined.

## Hardware Interfaces

To interact with an of the app’s functionality an internet connection is required as well as the use of an Apple, Android or Windows mobile device.

## Software Interfaces

The MAPSTER app is developed using the Ionic Framework utilizing both Google Firebase Database and Google Firebase Authentication Service to handle user data.

## Communications Interfaces

To utilize this app a user is required to have an Email Address for password recovery as well as an Android, iPhone or Windows smart phone. The use of a tablet running a similar operating system will also suffice.

# System Features

## Connect to Friends

By using the Firebase Database users can choose to connect to other users as “Friends”. This allows each user to see what the other is doing, as well as their location in real-time. These Friends may be added and removed at the will of the user so they may cultivate their desired demographic of reach.

## Take and Upload Images

Each user of the app can take pictures using their personal devices built in camera and upload them to the Firebase Database. These photos may also be tagged with keyword information and geolocation so that other users may see the images if they relate to things that interest them.

## View Locally Relevant Images

Users will be able to view all photos taken within a given radius of their location to gain knowledge of their surroundings and of potential attractions available to them in their area.

## Message Friends and Groups

Users can send and receive personal messages from other users or groups of users that the original user has brought together. This will help people create and cultivate plans they wish to carry out together, or just to catch up with each other.

# Other Nonfunctional Requirements

## Security Requirements

MAPSTER uses the Google Firebase Authentication that adds a user’s email, password and user name. This allows each person to have access to their account thus allowing them to alter only the information that they have created.

**Appendix A: Glossary**

Mapster, or MAPSTER: The app being developed by team 2.

Ionic: all relevant information can be found here: <https://ionicframework.com/>

Google Firebase: all relevant information can be found here: <https://firebase.google.com/>

Google Firestore: all relevant information can be found here: <https://firebase.google.com/docs/firestore/>

**Appendix B: Analysis Models**

The following link is to the most up to date design for the database structure.

<https://camo.githubusercontent.com/410dd28dfc7fff9154559a48421271afd7c9f464/68747470733a2f2f627269636577612e6d652f7265736f75726365732f696d672f70686f746f2f6d6170737465722d64622d6469616772616d2e737667>