**Software Engineering**

**Software Requirements Specification**

**(SRS) Document**

**Encrypted Messaging Application**

**Due Date: 12/05/2019**

**Version 1.0**

**Marc Minnick, Galen Shirey, Joshua Moran**

|  |
| --- |
| **Revisions** |

| Version | Primary Author(s) | Description of Version | Date Completed |
| --- | --- | --- | --- |
|  |  |  |  |

|  |
| --- |
| **Review & Approval** |

Requirements Document Approval History

| Approving Party | Version Approved | Signature | Date |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |

Requirements Document Review History

| Reviewer | Version Reviewed | Signature | Date |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**Table of Contents**

[1. Introduction 3](#_Toc26435908)

[1.1. Purpose 3](#_Toc26435909)

[1.2. Scope 3](#_Toc26435910)

[1.3. Definitions. Acronyms, and Abbreviations 3](#_Toc26435911)

[1.4. Overview 3](#_Toc26435912)

[2. General Description 3](#_Toc26435913)

[2.1 Product perspective 3](#_Toc26435914)

[2.2 Product features 3](#_Toc26435915)

[2.3 User class and characteristics 4](#_Toc26435916)

[2.4 Operating environment 4](#_Toc26435917)

[2.5 Constraints 4](#_Toc26435918)

[3. Functional Requirements 4](#_Toc26435919)

[3.1 Design Requirements 4](#_Toc26435920)

[3.2 Graphics Requirements 4](#_Toc26435921)

[3.3 Operation System Requirements 4](#_Toc26435922)

[3.4 User Requirements 4](#_Toc26435923)

[3.5 Constraints 4](#_Toc26435924)

[4.External Interface Requirements 5](#_Toc26435925)

[4.1 User Interfaces 5](#_Toc26435926)

[4.2 Hardware Interfaces 5](#_Toc26435927)

[5. Non-Functional Requirements 6](#_Toc26435928)

## 1. Introduction

1.1. Purpose

1.1.1. The purpose of this document is to build a system on the iOS platform that encompasses encrypted private messaging, group messaging, and audio/video calling functionalities.

1.1.2. The intended audience for this document is the back-end architects and end-users

### 1.2. Scope

1.2.1. The purpose of this system is provide a secure and private messaging platform for end-users to maintain contact with whomever they would like.This system will be based on a typical client-server architecture using an application accessing the users information from hashed information on the server.

### 1.3. Definitions. Acronyms, and Abbreviations

1.3.1. AES – Advanced Encryption Method

1.3.2. AWS – Amazon Web Services

### 1.4. Overview

1.5.1. The rest of the SRS will be an in depth description into the implementation and usage of the system itself.

1.5.2. Customers and end-users will be interested in section 2 and Developers will care about section 3.

## 2. General Description

### **2.1** **Product perspective**

The messaging application will be an independent system running on iOS. A server will be used for authentication purposes and will be part of the system as well.

1. System interfaces – AWS interaction with application
2. User Interfaces – Screen size: 1125x2436 with a notch (iPhone 11 Pro, X, XS)
3. Hardware Interfaces – iPhone 11 Pro, X, XS and iOS
4. Memory – less than 1 GB of storage

### 2.2 Product features

The software will allow a user the ability to add and remove contacts. The user can then send and receive messages, privately, to and from a contact. The user will also be able to join a group chat with those in their list of contacts. This will allow all users in the group chat to send a message to all other users in the chat simultaneously as well as receive messages from other users in the chat. The messages can be text, a picture, an audio file, or a video file. User will also be able to make audio and videos calls to a contact.

The software will also have a short tutorial that demonstrates what the application can do and how to do it.

### 2.3 User class and characteristics

The primary user is anyone that is interested in a private and secure way of communication.

### 2.4 Operating environment

The application will run on iOS and will be made using C#. The server will be an AWS server.

### 2.5 Constraints

* + The development team has no prior experience working with Amazon Web Services.
  + The project must be completed by the end of April 2019.
  + The application must be developed for the iOS operating system.

## 3. Functional Requirements

### 3.1 Design Requirements

* Encrypt peer-to-peer messages

### 3.2 Graphics Requirements

* Must be able to fit within iPhone screen size

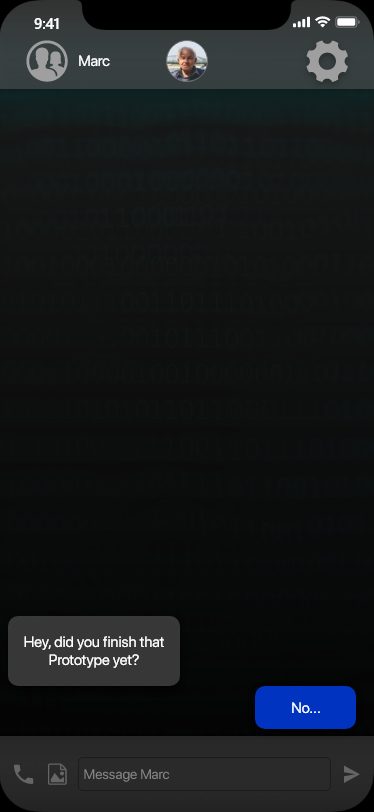
### 3.3 Operation System Requirements

### 3.4 User Requirements

* Add a new contact
* Delete a contact
* Block a user
* Send a picture to a contact
* Video call a contact
* Delete a message
* Logout
* Set availability status

### 3.5 Constraints

## 4.External Interface Requirements

4.1 User Interfaces

Every user interaction can be found in our prototype.

* Current types of interactions:
  + Open App
  + Login/Sign Up
  + Open Contacts
  + Type Message
  + Send Message
  + Send Picture
  + Call
  + Settings
  + Logout
  + Add New Contact

To the right is an example screen of a user to user message:

4.2 Hardware Interfaces

A list of supported devices goes as follows:

* + iPhone X
  + iPhone XS
  + iPhone 11 Pro
    - All of the above must be running the latest iOS (iOS 13 at the time of writing)

A connection between a server and device will be used to store user information such as:

* + Profile Picture
  + Public Key
  + User Status
  + User Bio

## 5. Non-Functional Requirements

**5.1 Safety requirements**

Some safeguards that may will be incorporated as a measure against any possible harm via the software include

* + RSA Public and Private key generation
  + AES Block Cypher to encrypt user data

**5.2 Security requirements**

We will be using modern day encryption standards including RSA and AES to keep all our user’s information safe. This includes database information of users as well as messages sent to and from users.

**5.3 Software quality attributes**

User interface is going to be able to adaptable to the customer’s needs.

(User interface is always subject to change and updates start to roll out)