Software Requirements

Specification (SRS)

Version 1.3
Last Updated: 6/10/20

Hanane Elkarni – Project Manager
Francys Cunanan – Assistant Project Manager
Ronnie Espinosa – Software Engineer
Daniel Firestone – Software Engineer
Ali Randell Kasim B Hamody – Software Engineer
Christopher Wise – Software Engineer
Kelsey Burgos – Software Engineer

Table of Contents

1. Introduction

1.1. Purpose

“Udrop” is an application (app) requested by the client, Micah Langston of *United Christian Academy*. His app will drop messages via text, images, or short videos to other recipients only to specific locations. This can range from being a fun activity for friends and families or be a helpful guide. As such, we will only be focusing on the user navigation and interface of the app for both iOS and Android.

Software Engineering students from California State University, San Bernardino (CSUSB) will be developing said app. Computer Science Engineering (CSE) 455 students will work to make a more engaging activity of internet communication in mobile devices. Throughout this document, the teams of both iOS and Android will define the app’s functionalities, limitations, requirements, and objectives to meet the client’s requested app.

The basic framework is already developed by the client in iOS. Yet, due to a split of phone preferences within the team, not only are we going to improve our client’s work but we have decided to make an Android version for the app as well. For 7 weeks, our team will be working with the client until the specified tasks are completed. The goal of the team is to provide a navigation system: changing page-to-page as well as update and push notification of the message within a designated area.

1.2. Scope

This app will entertain the youth of current generations, provide efficient step-by-step tasks for co-workers and peers, and keep fresh reminders of people when reaching specified destinations. Moreover, this app keeps people engaged, on the right track, and attentive.

Since we are going to improve an 18-month development of the project, the number one priority of this project is to guarantee that the app’s system works. Specifically, it is the team’s responsibility to be able to push notifications for the user and be able to move from home page to friend’s list without crashing. Likewise, the features (primarily settings) need to be interactable. It is also the team’s responsibility to have the same level of features acquired in both iOS and Android.

On our first phase prototype, we will be able to have the basic UI implemented on both platforms. We will have the layout, the home page, and be able to add friend's by the time the first prototype is due. However, due to our current situation this quarter, it is undetermined whether we will be able to do anything that involves private servers for our client. It may be possible, but as of right now we might not see that happening in our first prototype.

As for our second phase prototype, we plan on transferring messages to each other including images and possibly short videos as well. If the servers are given to us, we will also make accounts for users. Our biggest challenge for this prototype is to send messages that can only be received under certain locations. It is hard for numerous reasons, given the pandemic, needing outside help from Google Maps, and the reliance of servers. Overall, what will be promised is that we can send messages, images, and create accounts.

The fundamental system for this app is to include well-standard pages, such as a home page, add list, my contacts, profile page, address view, My Drops, and View Drops. Currently, the team will work on making the pages functional and easy to navigate while providing the same aesthetic in both iOS and Android platforms. Over the course of the 7 weeks, it is the team's top priority to fix any errors or crashes that may occur for the user.

1.3. Definitions, Acronyms, and Abbreviations

Android - mobile operating system designed primarily for touchscreen mobile devices such as smartphones and tablets.

App - Short for Application.

Client - Micah Langston of *United Christian Academy*.

Crash - When a software system malfunctions and has no choice but to exit.

CSE - Computer Science and Engineering.

CSUSB - California State University of San Bernardino.

Drop(s) - messages.

iOS - mobile operating system created and developed by Apple Inc. exclusively for its hardware.

Home page - The introductory page of the application.

Internet - Global network that provides information, communication, and storage of data

MB - Megabyte.

Memory - Stored information for immediate use of a device.

Mobile App - Mobile Application.

QA - Quality Assurance.

RAM - Random Access Memory.

SRS - Software Requirement Specification.

UI - User Interface.

User-friendly - Easy to use and understand.

Wifi - A facility allowing computers, smartphones, or other devices to connect to the Internet or communicate with one another wirelessly within a particular area.

Xcode - Integrated development environment for macOS containing a suite of software development tools developed by Apple.

1.4. References

Google Maps Platform - https://developers.google.com/maps/documentation/?_ga=2.208144658.1420330511.1587758017-505607879.1587758017&_gac=1.15993412.1587758028.CjwKCAjwnIr1BRAWEiwA6GpwNRM_m0v9pFy7MJuyPoOH0x4s1b2ztc3_6asSUDnk1h-NBdRZrzHBLxoCx-EQAvD_BwE

Xcode – https://developer.apple.com/xcode/

IEEE Std 830-1998

LLU "Health Hearts at Home" App Software Requirements Specification (SRS) -

https://mobileappdev.academic.csusb.edu/wp-content/uploads/2019/04/SRS_CHD-V2.1.pdf

Optimal State of Living (SRS) -

https://mobileappdev.academic.csusb.edu/wp-content/uploads/2019/04/SRS-.pdf

1.5. **Overview**

Section 2 Overview

-Interfaces: UML deployment, viewpoint of user interactions, list of required software and usage of multiple communication protocols

-Regulations & Assumptions: Memory limitations, app development restrictions, and application dependencies

-Functionalities: App operations and maintenance, case diagrams, and user characteristics

Section 3 Overview

This section of the document will detail the requirements of Udrop. Requirements for the various interfaces, functionalities, and other aspects of Udrop will be listed and a description of how we plan to meet these requirements.

2. Overall Description

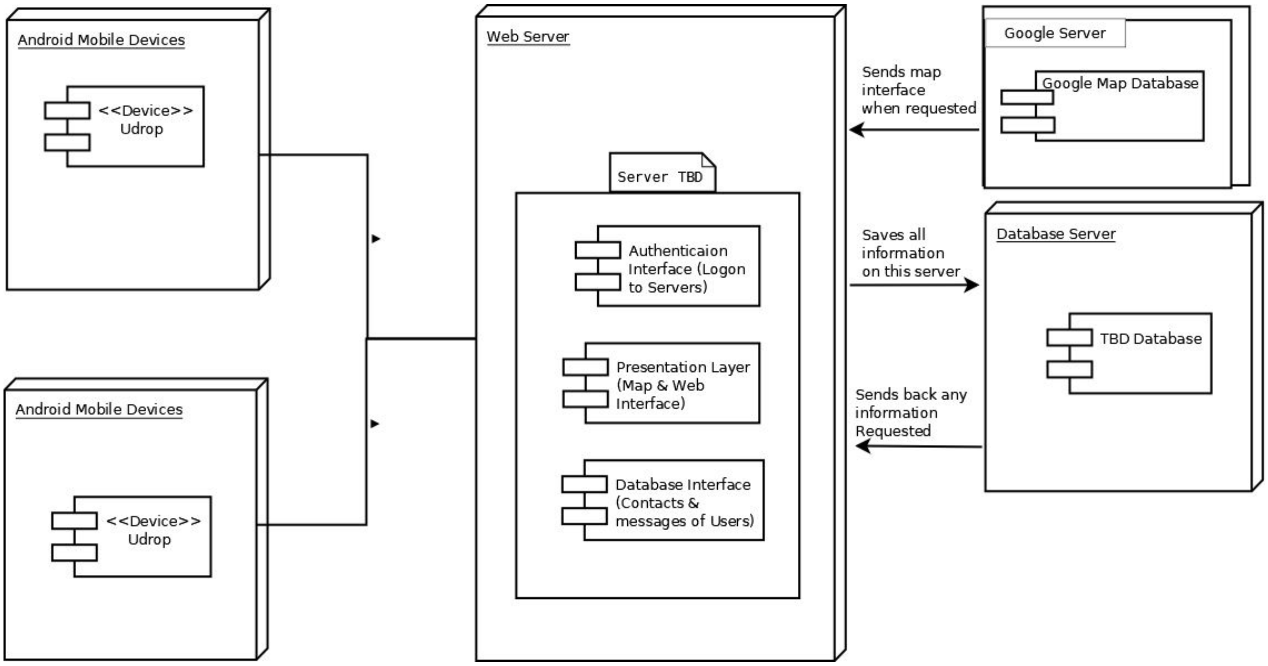
2.1. Product Perspectives

UDrop will allow users to be able to send and receive messages. Messages can be sent privately, to only be viewable between sender and recipient.

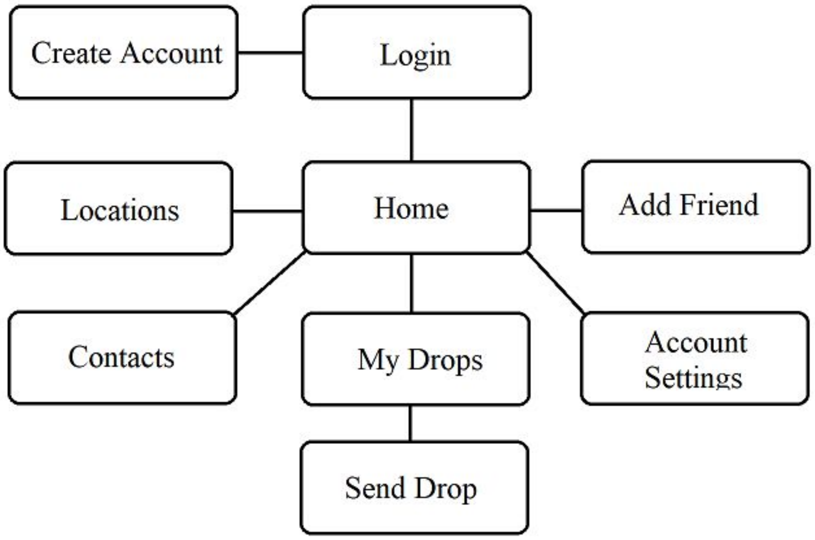
UDrop will be used to connect strangers and be used to enhance social interactions. Our app can be used to create conversation between two different users that have both opened the same public message or be used between friends as a grocery list.

The app will be co-developed for iOS and Android side by side.

2.1.1. System Interfaces (deployment diagram)


Android Mobile Devices
<<Device>>
Udrop
Android Mobile Devices
<<Device>>
Udrop
Web Server
Server TBD
Authenticaion Interface (Logon to Servers)
Presentation Layer (Map & Web Interface)
Database Interface (Contacts & messages of Users)
Google Server
Google Map Database
Sends map interface when requested
Saves all information on this server
Database Server
TBD Database
Sends back any information Requested

2.1.2. User Interfaces



This is a wireframe of how the app will traverse through different screens. Upon first launch Udrop will open the login screen. If a user does not have an account registered, they may access the Create Account screen from the Login Screen. Once the user has logged in, subsequent openings of the app will load from the Home Screen. The Home Screen has Access to all the features of the app and each screen will feature a back button to return to the Home Screen.

2.1.3. Software Interfaces

XCode - for iOS Development
Android Studio - for Android Development
Server setup by CSUSB

2.1.4. Communication Interfaces

Communication occurs between the server and the mobile device. The mobile device will establish an HTTP connection between the server which is then relayed back to the mobile device.

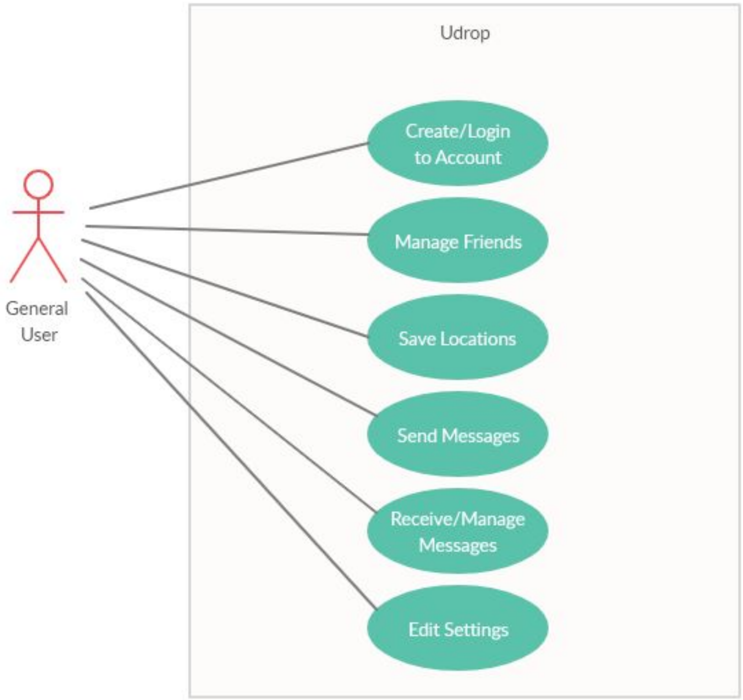
2.1.5. Memory

The UDrop server should limit each user to around 10 to 15 GB of storage. Videos will be at a maximum length of 10 seconds to preserve storage. Most of the users data will be stored server side and as a result the total memory allocated for the app should be around 300 MBs.

2.1.6. Operation

The UDrop server should remain running at all times to ensure a smooth experience for our users. Messages between users will be stored on the server but users will have the option to be able to save messages, such as videos and pictures, onto their device at the cost of their own storage.

2.2. **Product Functions (use case diagram)**



2.3. **User Characteristics**

The general user is anyone who wants to connect with friends or family through social media. They are people who enjoy sending and receiving video, text or image messages, and managing friends lists. They are also familiar with basic GPS functionality to select an address of location.

2.4. **Constraints**

Becoming familiar with GPS integration
Becoming familiar with technologies such as Android Studio and servers
Learning the design and engineering process of mobile app development

2.5. **Assumptions and Dependencies**

The development team will assume that the user has access to the internet in order to be able to send/receive messages, utilize the features in UDrop. In addition, users will be expected to have some basic familiarity with GPS along with familiarity in the device itself. Updates may occur leading the user to have the knowledge of how to update UDrop.

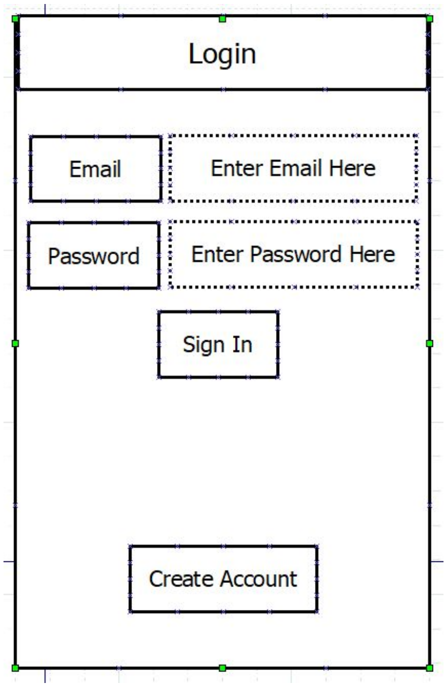
3. Specific Requirements

3.1. External Interface Requirements

3.1.1. User Interfaces (subject to change)

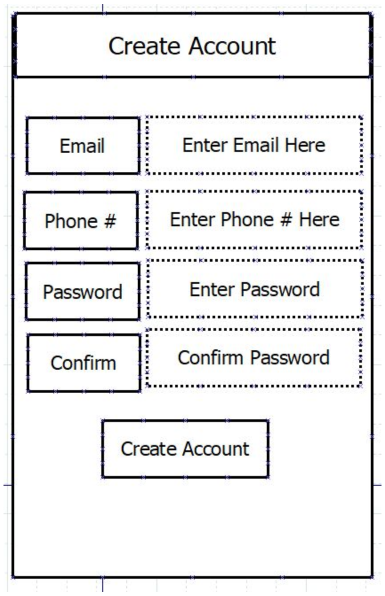
3.1.1.1. Login

The Login page allows users to login to their account, which brings them to their homepage. If they do not already have an account, they can navigate to create an account.



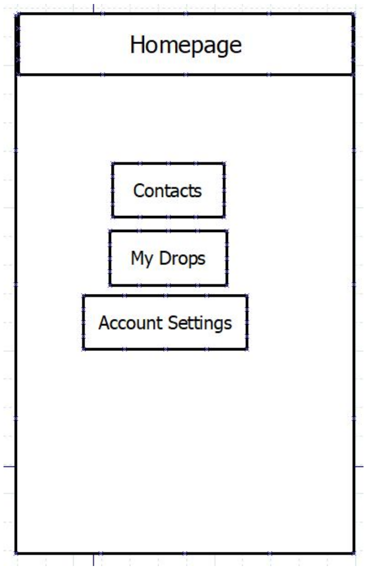
3.1.1.2. Create Account

The Create Account page allows new users to create an account by submitting an email, phone number and password to be associated with the account.



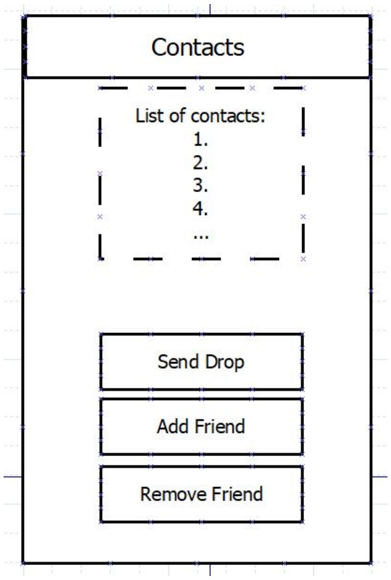
3.1.1.3. Homepage

The Homepage allows the user to navigate to their contacts, drops, or account settings.



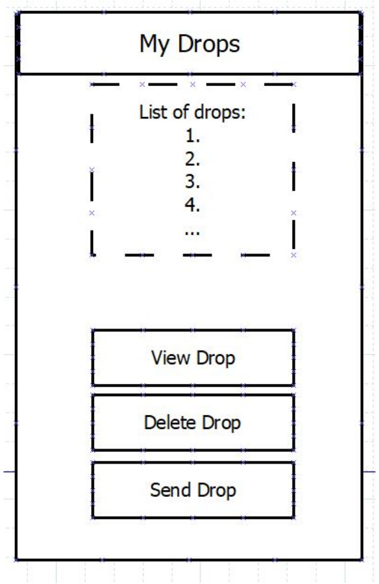
3.1.1.4. Contacts

The Contacts Page allows the user to view their contacts list, send drops, add friends, or remove friends.



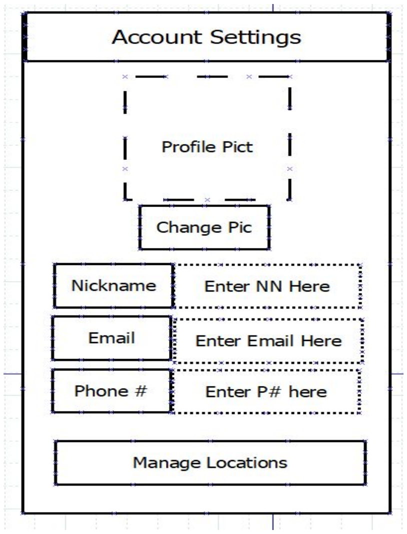
3.1.1.5. My Drops

The My Drops page allows the users to view their list of drops, view individual drops, delete drops, or send drops.



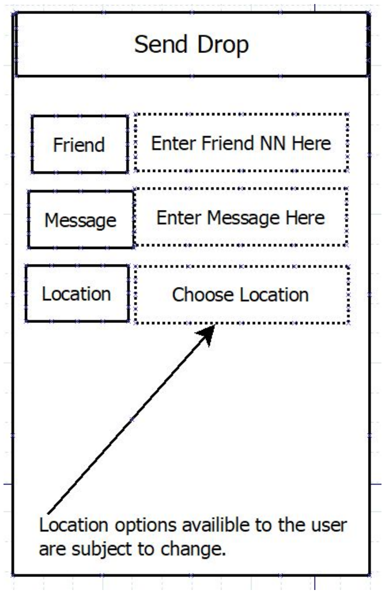
3.1.1.6. Account Settings

The Account Settings page allows the user to change the following: profile picture, nickname, account email, account phone number. The user may also navigate to their saved locations.



3.1.1.7. Send Drop

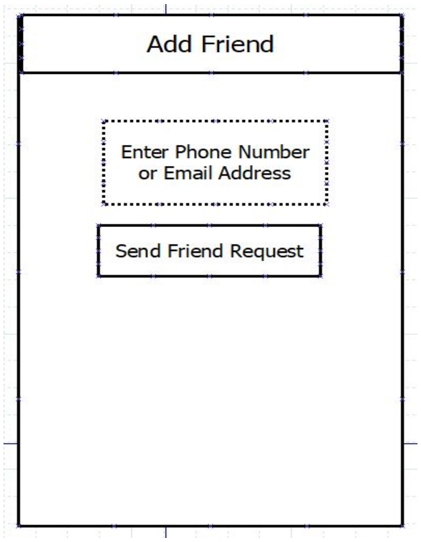
The Send Drop page allows the user to choose a friend, message, and location to send a drop to.



Location options availible to the user are subject to change.

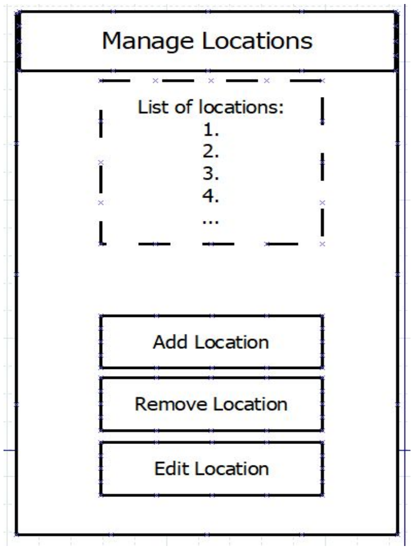
3.1.1.8. Add Friend

The Add Friend page allows the user to send a friend request to another user by searching for the other user's phone number or email address.



3.1.1.9. Manage Locations

The Manage Locations page allows the user to view their saved locations, add new locations, remove saved locations, or edit saved locations.



3.1.2. Hardware Interfaces

Udrop is being developed for both iOS and Android. It is unlikely that the app will be ready for the app stores at the end of our team's development period.

3.1.3. Software Interfaces

Drops, user profiles, and related information will be stored on a database. The database provider will be determined at a later date.

3.1.4. Communication Interfaces

Since Udrop is a social media app, an internet connection is required for most app functions to be performed successfully. However, the user may edit their profile without an internet connection.

3.2. **Functional Requirements**

3.2.1. Upon Launching Udrop

3.2.1.1. The user will be brought to the login / create account page or user homepage if they are already logged in.

3.2.2. Drop Functionality

3.2.2.1. The user chooses a friend from their friends list to send a drop to.

3.2.2.2. The user creates a drop as a video, image, or text message.

3.2.2.3. We intend to enable the user to choose the drop location from one of the following: saved location list, user specified address (using Google Maps), user specified latitude/longitude (using Google Maps).

3.2.3. User Profile Feed

3.2.3.1. The user may choose drops received from friends to share on their feed. Other users may view the feed.

3.2.4. Sending and Receiving Friend Requests

3.2.4.1. The user may send friend requests to their friends phone number or email address associated with Udrop. The receiver will receive a notification from Udrop.

3.2.4.2. The user may view pending friend requests and choose to ‘accept’ (add to friends list) or ‘ignore’ (remove request without accepting).

3.2.5. Saved Locations

3.2.5.1. The user may add or remove locations (latitude/longitude or address) from their Saved Locations list. The user can give each saved location a name like “Home” or “School”.

3.3. **Performance Requirements**

Udrop will be available for download on the App store and Google Play. It will run on most versions of Apple iOS and Android.

3.4. **Design Constraints**

Udrop will be designed on Xcode and Android Studio. As this app’s primary purpose is to connect users with each other through messages sent over the internet, each user’s cellphone should be connected to the internet in order to use this Udrop properly.

3.5. **Software System Attributes**

3.5.1: Reliability

It is ideal that Udrop messages can only be picked up if the specified recipient is in that designated area. People should not be able to see what the user sent privately unless it is instructed to be a public thread. The QA team will be there with independent testing to make sure messages can be sent reliably.

3.5.2: Availability

Currently over this course, the application will only be available to the CSUSB team and private testers. Eventually, as its progress continues, it will be available in both the Apple and Google Play Store.

3.5.3: Security

As expected, the devices and methods will follow security protocols such as the class of CSE 455 would need approval for design decisions and quality assurance. It should be acknowledged that nothing should be used to perverse personal data or info of the user. Therefore, we have an independent team to redirect the prototype into the morally legal and just product.

3.6. Document Approval

Client Signature:________________________________

Date:____