

Release 1.0 Acceptance

Product Name: Virtual Time Capsule

Team Name: VTC

Release Date: 3/16/2018

### **High Level Goals:**

Allow users to create and personalize an account on the VTC application

Allow users to upload media through the application

Allow users to send uploaded media to themselves and/or friends at established time

Allow users to be able to connect with other users by adding a friend request system

### **Implemented User Stories and Features:**

#### **Sprint 1**

- As a product owner, I want users to be able to create accounts efficiently on VTC so that users aren't pushed away from a tedious sign up and therefore there are more downloads.
- As a user, I want to be able to create an account in a short amount of time
- As a developer, I want to learn the new technologies involved in this project

#### **Sprint 2**

- As a user, I want to be able to choose photos from my camera storage to upload.
- As a developer, I want to ensure that our cloud server allows account signup/verification, user sign in authentication, and upload/retrieval of media. (All in Firebase)
- As a developer, I want to ensure there is a space within the app where users can find their friends and create groups.

#### **Sprint 3**

- As developers, once everyone's parts are finished, we want to connect Firebase to the user pages; thus, allowing the user pages to update automatically.

#### **Sprint 4**

- As developers, once everyone's parts are finished, we want to connect them
- As a user, I want to receive my time capsule at the time that I set.
- As a user, I want to be able to receive time capsules and look at its contents.

#### **1. Backend and database (Firebase)**

##### **a. Sprint 1**

- i. Firebase project set up

##### **b. Sprint 2**

- i. Sign in and email/sign up verification research and set up
- ii. Photo upload research

##### **c. Sprint 3**

- i. Sign in and email/sign up verification in app

- ii. Photo upload to cloud storage
  - d. Sprint 4
    - i. Friend system in database
    - ii. Photo Retrieval from cloud storage
- 2. User Accounts
  - a. Sprint 1
  - b. Sprint 2
    - i. Friend pages
  - c. Sprint 3
    - i. User signup page created and connected to database
    - ii. User signin page created and connected to database
  - d. Sprint 4
    - i. Friend system(searching and adding) connected to database
- 3. Front end
  - a. Sprint 1
    - i. Start screen
    - ii. Design
  - b. Sprint 2
  - c. Sprint 3
  - d. Sprint 4
    - i. Connected camera roll, camera
    - ii. Connected friend pages
- 4. Camera and photo selection from camera roll
  - a. Sprint 1
    - i. Use camera to take pictures from the app
    - ii. Listing of photos in the phone storage
  - b. Sprint 2
    - i. Navigation between camera and camera roll
    - ii. Select (1) photo from camera roll
  - c. Sprint 3
    - i. Select multiple photos from camera roll
    - ii. Photos can be uploaded to database
  - d. Sprint 4
    - i. Camera and camera roll in main app
    - ii. Photos can be downloaded from database
- 5. Photo retrieval from database on user-set time
  - a. Sprint 1
  - b. Sprint 2
  - c. Sprint 3

- d. Sprint 4
  - i. Images are retrieved from the database if the set date has passed

**Known Bugs:**

1. Pressing the login button repeatedly before the screen changes makes you login multiple times (but does not crash or prevent functionality in the app)
2. When a new user does not have friends and tries to look at the friends list, the app throws an error “null is not an object” .
3. Image is not rendering when being retrieved from database

**Product Backlog:**

Allow users to add keywords to media before sending it  
Send notification to the user when a time capsule is received  
Facebook login  
Importing friends from Facebook  
Video Uploading