Prepared By James Hayes, Mei’lani Eyre, Juanita Aguilar, and Sharief Youssef
BIT 381 | Spring 2019

GYBITG Mobile App (iOS) Project Backlog

Updated On: 2019-06-11

User Story 1: As an athlete, I want to be able to capture and store video(s) of me playing or practicing basketball, so that Coaches viewing my profile can see my skills.

~~Feature 1: Video Capture and Upload (Gallery)~~

~~All Athletes shall have a *Gallery* for their highlight *Video* clips, so they can be viewed by Coaches. An Athlete shall be able to access their phone’s camera via their *Gallery*, then upload that *Video* to their *Gallery*.~~

~~Feature 1.1: Video Model~~

~~A *Video* model shall be coded, that represents a video’s properties, such as a title, description, length, upload date, and file path.~~

~~Feature 1.2: Media Upload (Gallery) Storyboard~~

~~In order for the Athlete to showcase their basketball skills, a media upload (*Gallery*) view shall be designed so that the user can access their phone’s camera and upload a Video.~~

~~Feature 1.3: Video Data Repository~~

~~As a way to separate the controller from the data and to add a single entry point to work with the Video model, there shall be a data repository layer and repository protocol to access the model and data.~~

~~Feature 1.4: TableView with Custom Cells for the Gallery Screen~~

~~Create the table view that will hold each video as well as its title in custom cells. It should have a thumbnail of the video, the date it was uploaded, and the title of the video.~~

~~Feature 1.5: Camera Access~~

~~In order for the user to upload videos to their gallery, there shall be the necessary code to access the devices' camera via the GYBITG app.~~

Feature 1.6: Create a video editing feature

The user shall select a thumbnail of the video and edit the Description or delete the video from the tableView.

Feature 1.7: Be able to upload a video from an outside

The user shall be able to upload a video from an outside site like Youtube to their gallery.

User Story 2: As a user, I want to be able to register an account, so that I can have a profile to store my basketball videos and game stats, so that coaches can view my basketball skills.

~~Feature 2: Account Registration~~

~~A user shall be able to sign up for an Athlete Account, The registration process shall require minimal input (Email, Password), so the process is quick and streamlined.~~

~~Feature 2.1: Account Model~~

~~A model that represents information a user would need to register an *Account* such as a unique email address, and password shall be coded.~~

~~Feature 2.2: Account Registration & Login View~~

~~A form to be filled out by the user so they can register an *Account* shall be designed. The form will require an email address and desired password textboxes.~~

~~Feature 2.3: ViewControllers for the *Account* model~~

~~The view controller code that shall allow a user to create a new account and login.~~

~~Features 2.4: ViewControllers use the Account model~~

~~The view controllers shall start using the accounts model during the account creation and login.~~

User Story 3: As an athlete, I want to be able to input my stats after one of my games, so that I can track my career stats, and I know which aspects of my game I should focus on improving.

~~Feature 3: Athlete Stats Entry~~

~~A way for users to track their stats over a season; after each of their games the user shall manually input their game stats, such as but not limited to: points, rebounds, assists, steals, blocks, turnovers, minutes played, the opposing team, and home/away game location. This way the user shall track their progression/ regression, so they can emphasize their gym time on those skills that are lacking.~~

~~Feature 3.1: *GameStat* Model~~

~~To be able to track an athlete's career game stats, a *Stat* model shall be created for local and/or remote storage later on. The model shall have properties that a basketball player would normally track during a game, i.e.: points, rebounds, assists, steals, blocks, and minutes played.~~

~~Feature 3.2: Athlete *GameStat* Entry Form Storyboard~~

~~A way for users to track their stats over a season; after each basketball game the athlete shall manually input their game stats (points, rebounds, assists, etc.) through a form view.~~

~~Feature 3.3: *GameStat* Data Repo and Protocol~~

~~As a way to separate the controller from the data and to add a single entry point to work with the *GameStat* model, there shall be a data repository layer and repository protocol to access the model and data.~~

~~Feature 3.4: Save *GameStat* through the form~~

~~When the user is able to save a *GameStat* object to the data array in the GameStat repo. When the stats table-view displays all the user specific *GameStat* objects stored in the repo data array.~~

~~Feature 3.5: View and edit *GameStat* details~~

~~When the user is able to select a *GameStat* entry in the table view and is segued to the form view, with that entry's details displayed. The user shall be able to edit the *GameStat* properties and save it.~~

Feature 3.6: Save partially filled out form as a draft

If a user is unable to fill out the entire game stat form, they shall be able to save it as a draft, so they are able to go back and fill it out at a later time. This game stat is not saved to the main repo data-array. The user shall be reminded that they have an unfinished stat form through some sort of in-app notification.

Feature 4: Points System

To incentivize users to login, upload game stats and highlight-videos, a points/"likes" system can be used. When a user reaches a certain number of points GYBITG merchandise (t-shirts, wristbands, etc.) or mentions/”shoutouts” on social media can be given. There can be different tiers to the points system, for more incentive.

Feature 5: User Profile Editing

Once a User registers an account, depending on their membership type, they will want to fill out (edit) the various profile details that are associated with their membership (Athlete or Coach). At anytime the User should be able to navigate to their profile and edit these details.

Feature 6: Coach to Athlete Connection

If a Coach sees a player that they are interested in recruiting, there should be a way for them to contact that User, through an instant message or email type feature on the app. Only Coaches can initiate the conversation and it is not possible for Athletes to contact other Athletes. Athletes should have the option to set their desired contact method and to turn on/off their ability to be contacted.

Feature 7: Skill Development (Basketball Drills)

As a way of getting Athletes in to the gym the app should feature a set of basketball drills such as, but not limited to: shooting drills, passing drills, and dribbling drills. After completing a set of drills the User will earn a Badge. The drills should have several levels of difficulty (easy/ med/ hard) and some timed.

Feature 8: Badges (Verification of Completed Drills)

To showcase an Athletes completed Skill Drills or progression in their stats tracking their will be a visual display on the Athletes profile informing viewers of what they have accomplished.]

Feature 9: Setup Persistent Data Storage (on the Cloud)

Create a datastore, potentially on the cloud, that will persist the user's data, including their profile, videos, and stats. The datastore shall be in a secure server with secure access setup.