Fall 2020 LaunchCode LiftOff: Project Planning, User Stories and Wireframe
Project Group: Team-Chris-Massie
Team Members: Christopher Massie(mentor), Scott Sullentrop, Zachary Ainsworth, Richard Foster, Simon Duffy and Kavitha Krishnarao

As a [type of user] I want to [perform some task] so that I can [achieve some goal]

As a user I need to be able to create an account with a password and username.
As a user I need to be able to login to access my account
As a user I need to be able to logout of my account
As a user I need to be able to add a game on to my account
As a user I need to be able to search for my saved/stored game
As a user, with my game, I need to be able to add achievements to the game
As a user, with my game, I need to be able to import existing achievements
As a user, with my game, I need to be able to delete completed achievements to the game.
As a user, with my game, I need to be able to add a review for the game
As a user, with my game, I need to be able to access the review for the game
As a user, with my game, I need to be able to edit the review for the game
As a user, with my game, I need to be able to delete a review for the game

As a developer, I have to design a repository holding account information for the users
As a developer, I have to design a repository to store a list of games
As a developer, I have to design a repository to store the reviews for the games
As a developer, I have to design a repository to store the achievements for each game
As a developer, I have to design:
- Index or anchor page
- List page for the complete game list
- Page/Form output for each game including: a link to achievements and reviews
- List page for the complete achievement list
- Page/Form output for each achievement including notes
- Page/Form output for each achievement with the ability to delete the achievement
- Search page for games
- Search page for achievements
- Search page for reviews

As a developer, I have to design controllers for:
- Index or anchor pages
- Retrieving game data
- Retrieving achievement data
- Deleting achievement data
- Adding achievement data
- Retrieving review data
- Adding review data
- Search for games

- Search for achievements
- Search for reviews
- List games
- List achievements
- List reviews

Wireframes

As user I want to add a gamePreview attachment 20201128_111427.jpg
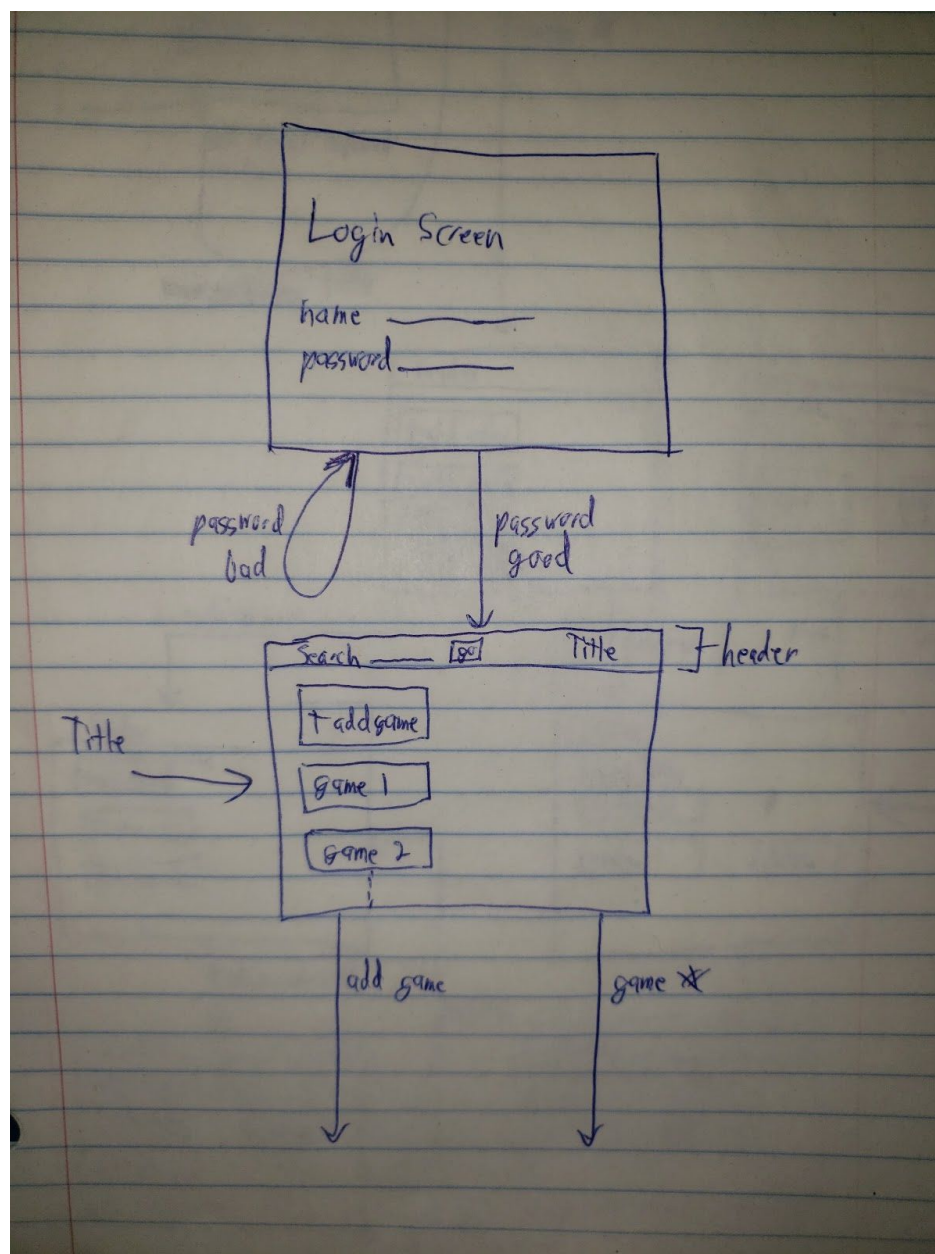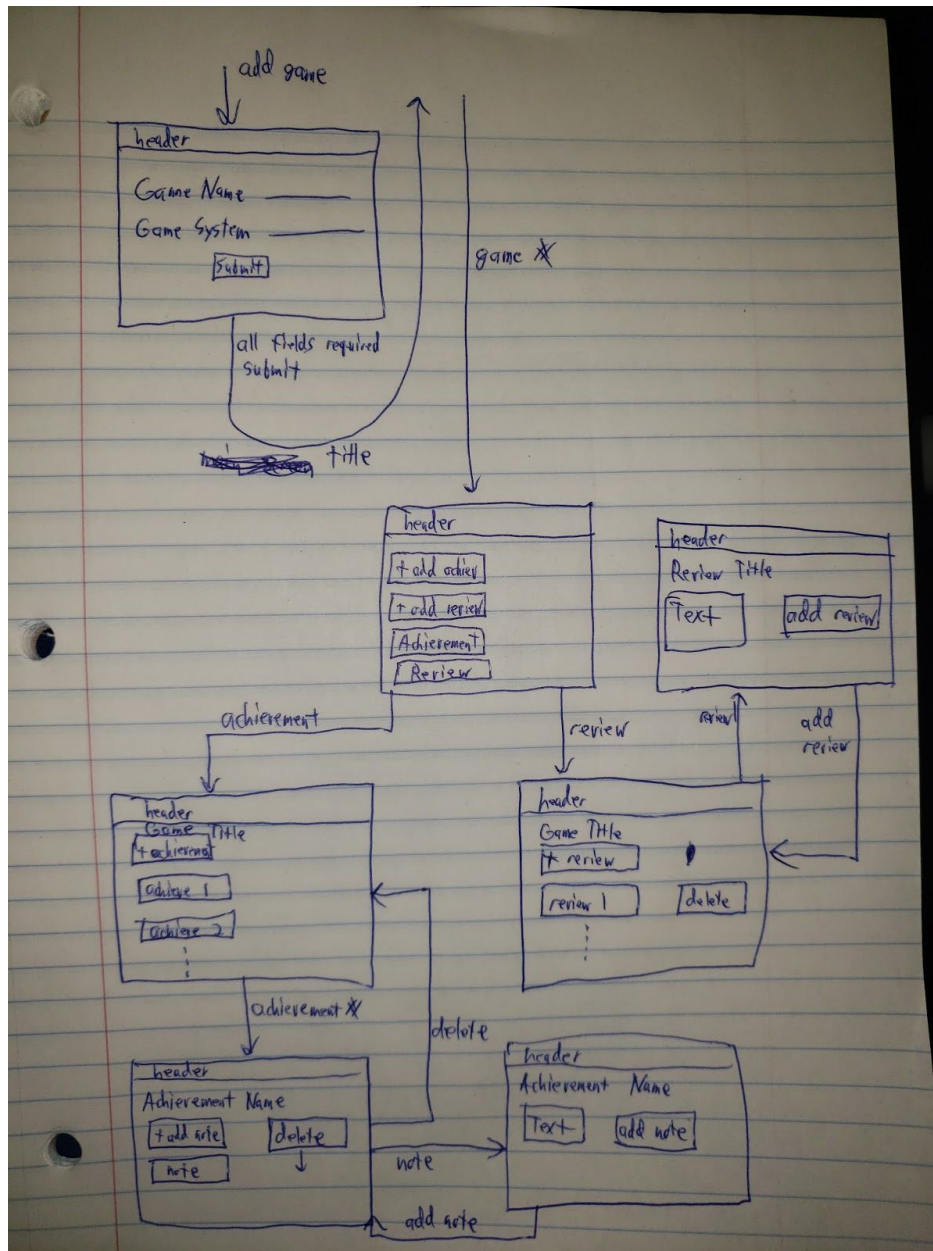

20201128_111427.jpg
3.7 MB

I like the structure. I think it'll need a little more space for the achievement/goal because they rarely will fit in a small text box. I attached a picture of the ground level class definitions at the end of the document, along with a translation, but I figured this was a good way to get things started. I also like the idea of typing in the system. I thought about having a drop down menu or a button to make a choice, but this is a lot more open and adaptable for a variety of systems, which was the idea. :)

Login Screen

name ———

password ———

password
bad

password
good

Search —— [go]   Title   ] header

Title

┌──────────┐
│ + add game │
└──────────┘

┌──────────┐
│ game 1 │
└──────────┘

┌──────────┐
│ game 2 │
└──────────┘

add game   game ✗

add game

header

Game Name _____
Game System _____
[Submit]

all fields required
submit

game X

~~title~~ title

header
[+ add achiev]
[+ add review]
[Achievement]
[Review]

header
Review Title
[Text]   [add review]

achievement

review   review   add review

header
Game Title
[+ achievement]
[achieve 1]
[achieve 2]
⋮

header
Game Title
[+ review]
[review 1]   [delete]
⋮

achievement X

delete

note

add note

header
Achievement Name
[+ add note]   [delete]
[note]

header
Achievement Name
[Text]   [add note]

Game *✓

String title
Int game_id
List achievements achievements
List reviews reviews

User ✓

Int user_id
List game games

Achievement **✓

String title
Int ach_id
String description
[get + set must include delete]

Review **✓

Int review_id
String description
[get + set must include delete]

User has Games has Achievements
has Reviews

** Has its own repository
* Foreign Keys managed by SQL

Game:

String title
Int game_id
List <Achievement> achievements
List <Review> reviews

Achievement:

String title
Int achievement_id
String description

Review:
       Int review_id
       String description

User will have the same basic idea, but I didn't want to own in on that since Kavitha said she was working on that. Each of these classes will also have repositories and each of these must have the "Delete" function coded in as well, though I think that can happen on the Controller side of things as opposed to the actual class side.

For the relationship table I saw in lecture, I didn't draw it out, but I kind of wrote it out. Thankfully, our project is very direct so:

User   ->   Game -> Achievements
                              -> Reviews