How to Use this Template

1. Make a copy [File → Make a copy...]
2. Rename this file: "**Capstone_Stage1**"
3. Replace the text in green

Submission Instructions

1. After you've completed all the sections, download this document as a PDF [File → Download as PDF]
2. Create a new GitHub repo for the capstone. Name it "**Capstone Project**"
3. Add this document to your repo. Make sure it's named "**Capstone_Stage1.pdf**"

GitHub Username: itsnotafake

Atakr

Description

Atakr is a hub for sharing and browsing videos of your favorite games. At Atakr we value getting our user's involved, so if you have a video of your own that you'd like to share or a video that you enjoy, don't hesitate to share it to Atakr.

Intended User

Intended to be used by people that enjoy playing and watching video games, and that want a centralized location for browsing video game (video) content.

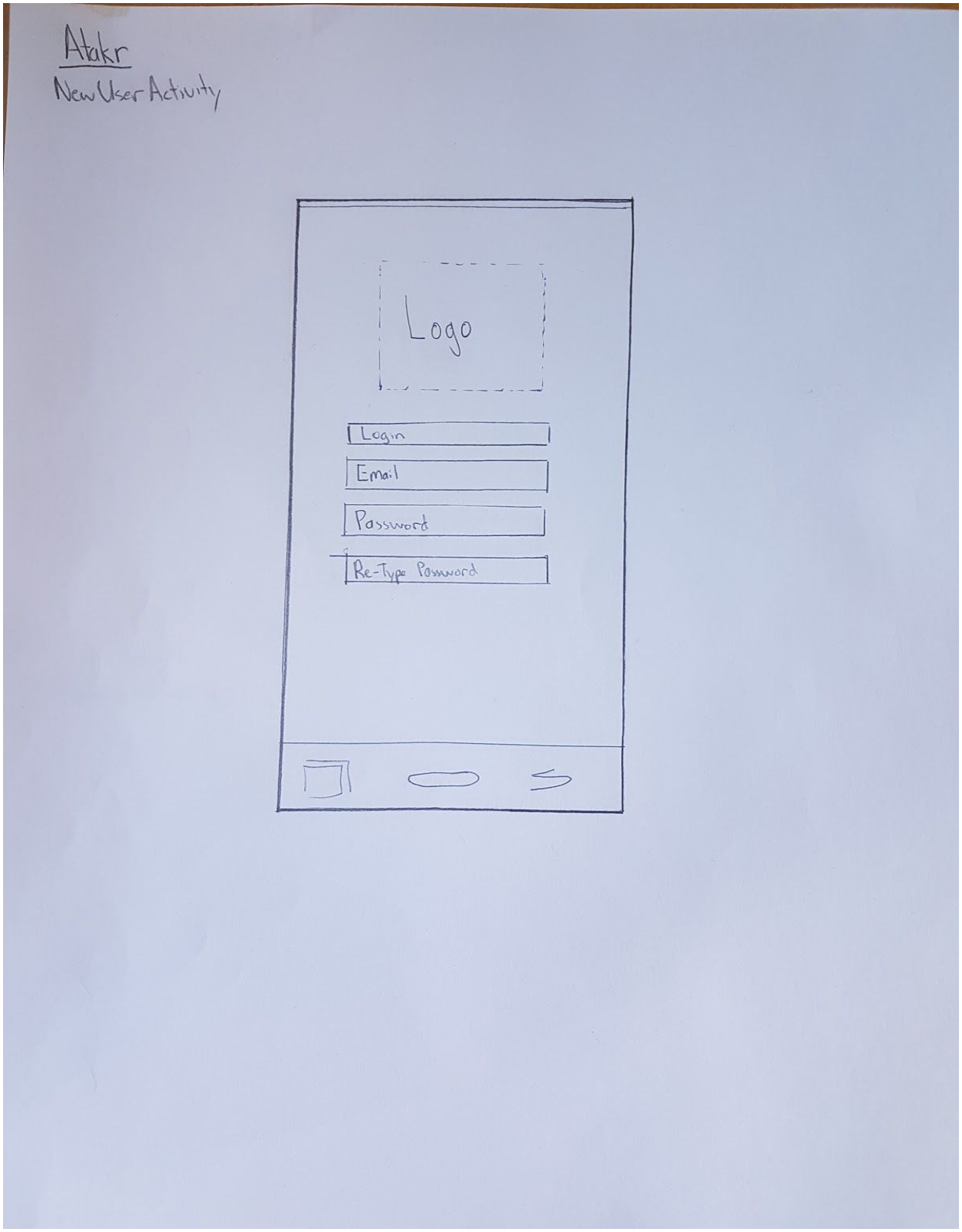
Features

- Users can share videos from other sites or apps (Youtube, gyfcat) which will then be displayed in Atakr's browsing section.
- User's browse Atakr's categories (game, game genre, video genre) to enjoy numerous videos at their leisure.
- User's can follow users and games to get a more customized experience (sends notifications when a followed user releases a new video, or when a followed game has a trending video.)

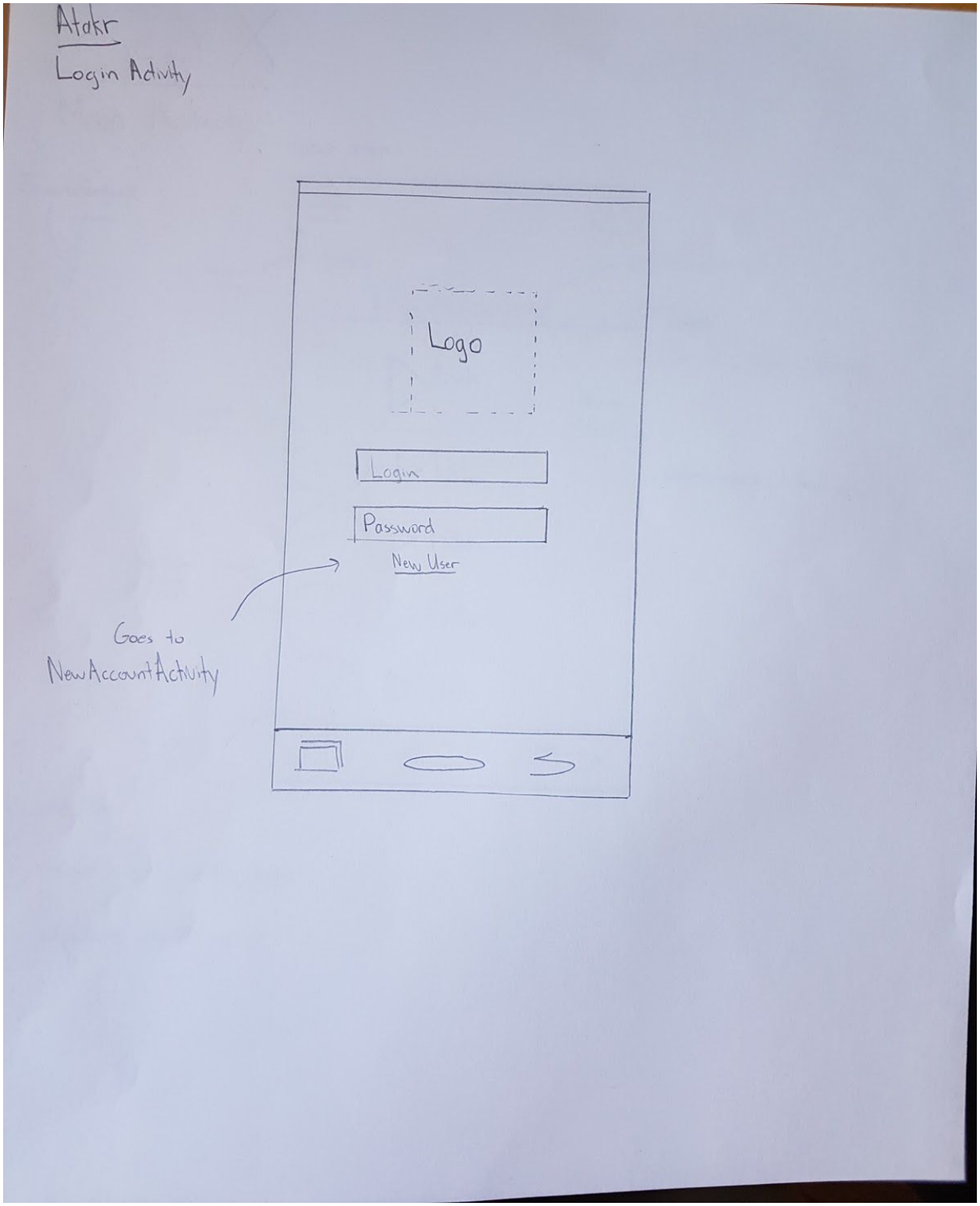
User Interface Mocks

These can be created by hand (take a photo of your drawings and insert them in this flow), or using a program like Photoshop or Balsamiq.

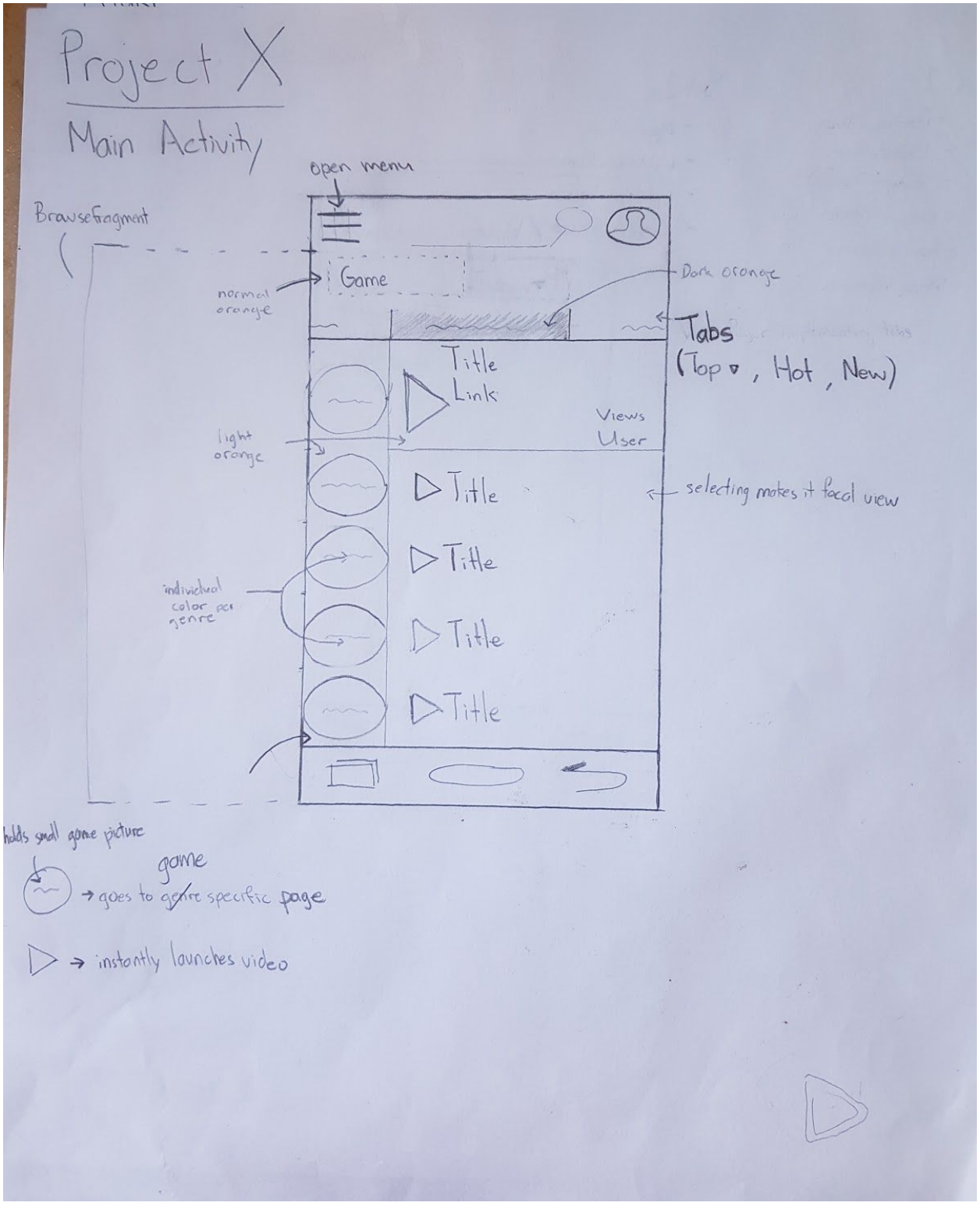
NewUserActivity



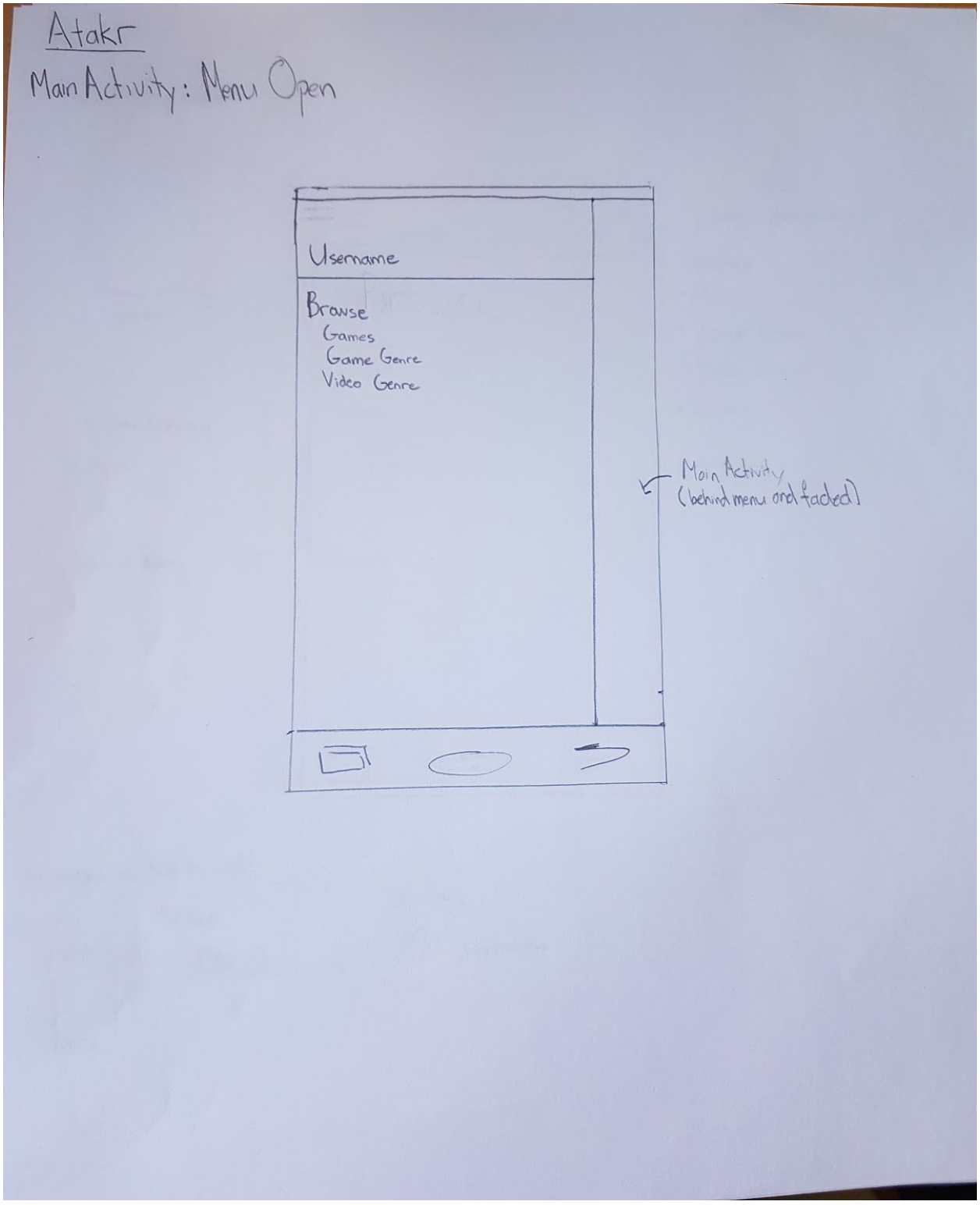
LoginActivity



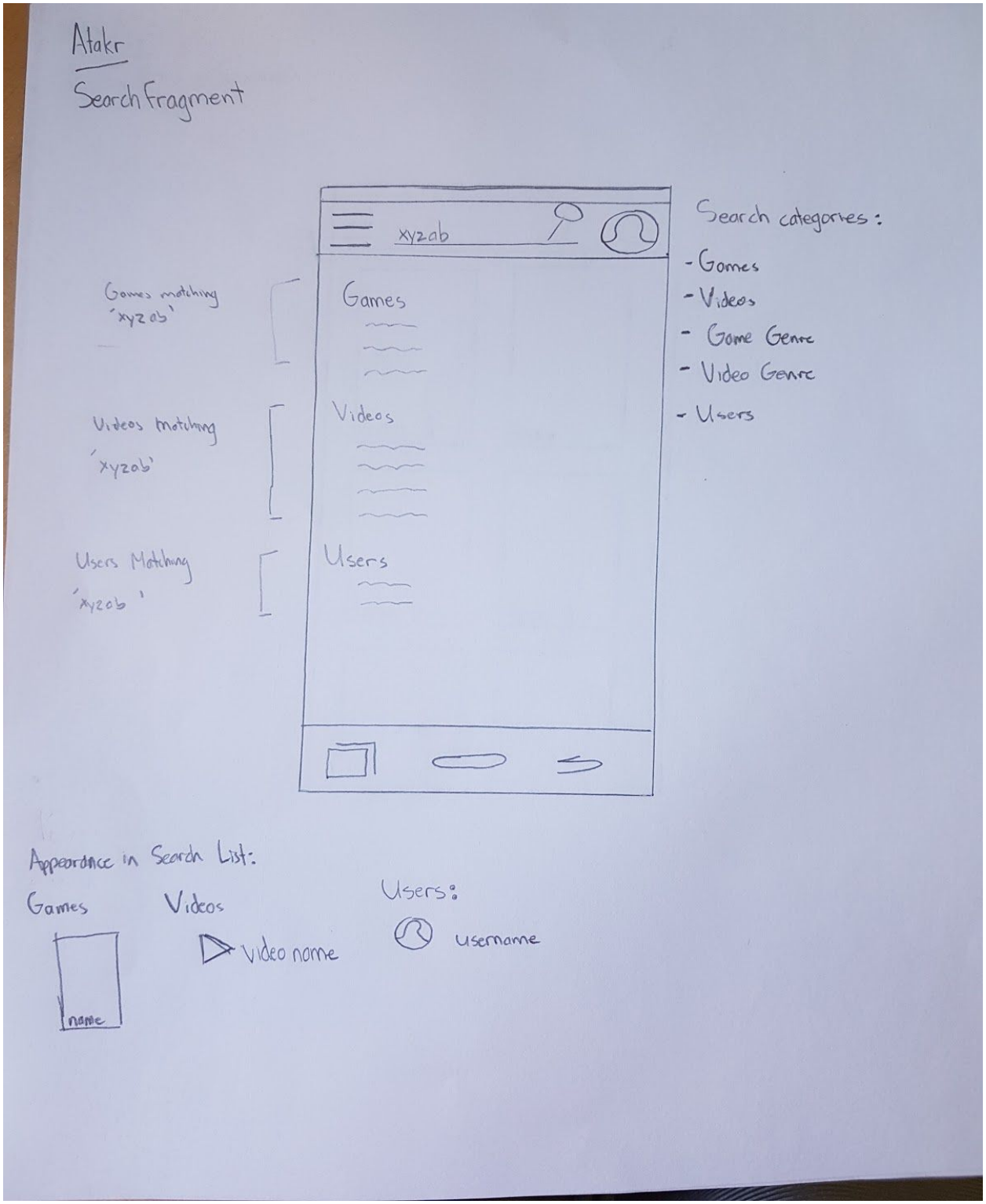
MainActivity + BrowseFragment



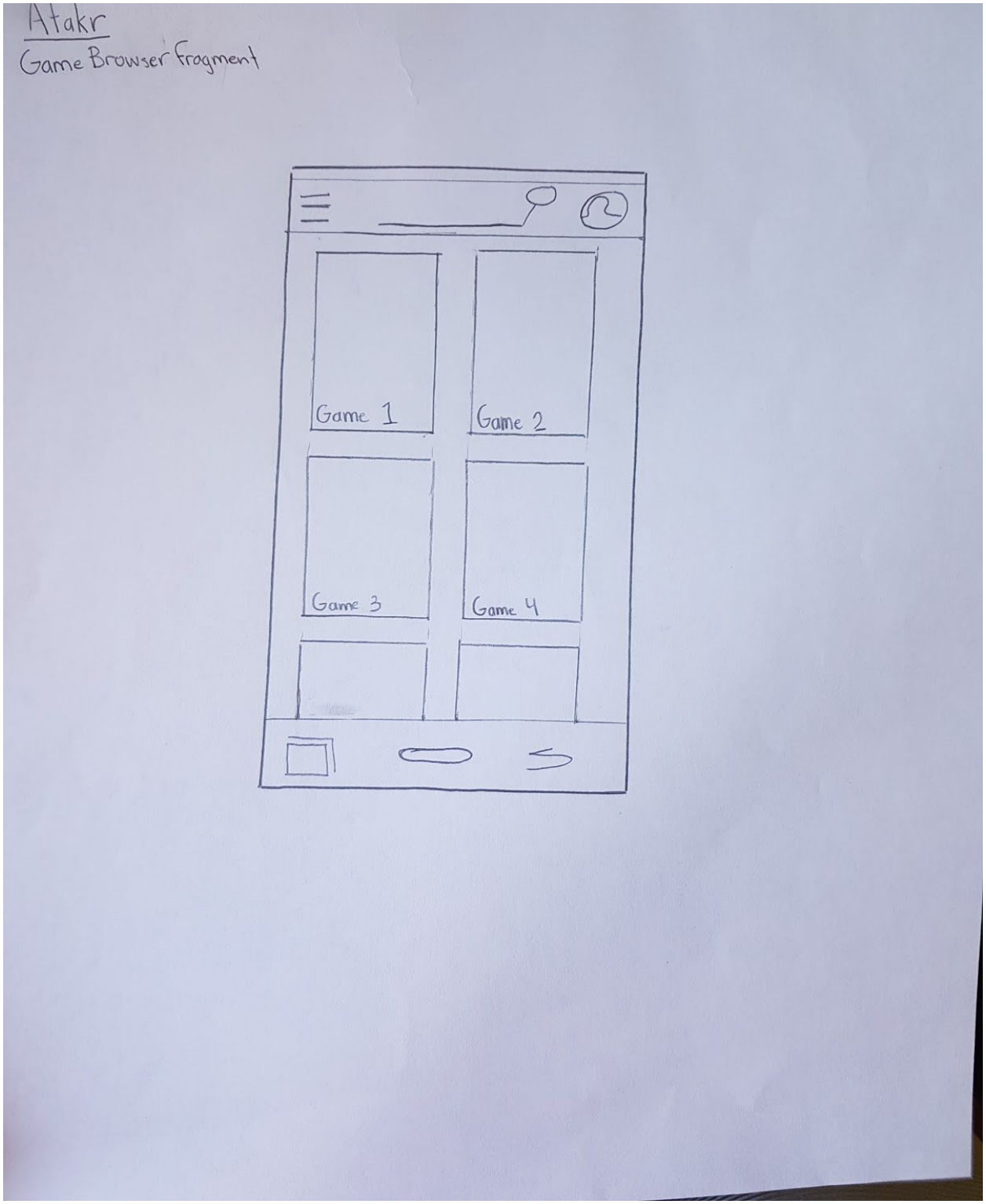
MainActivity - Menu Open



SearchFragment



GameBrowseFragment(Possibly TODO)



Key Considerations

How will your app handle data persistence?

To share video content with user's I intend to maintain a server-side SQL database on Google Firebase that contains video names and links to where the video is hosted. The idea is not to host videos myself but to serve as a gateway to where the videos are held.

I will also have to maintain a server-side database of users that syncs to a client content provider. Personal user data will be stored locally, but of course the server will have to keep track of everything.

Lastly, there will be a client side content provider that holds links to relevant videos. For example, if the user is currently browsing Hot All Games, then this database will hold, say, 50 videos that fit this description. When the user scrolls past these 50 videos are switches to a different browsing category, we send a call to the server side to get the next 50 videos.

Describe any corner cases in the UX.

Pressing the back button when the video player is open closes the video player and returns to the browsing fragment.

For the NewUserActivity and LoginActivity, pressing 'go' on the phone's keyboard when the user is not on the last input box moves the user to the next input box. Pressing 'go' when on the last input box submits the data.

Pressing back when the menu is open closes the menu. The only other way to close the menu is to press the menu button again.

When browsing games, 'All' will always be the first result so that the user has a way to navigate back to 'all'.

Describe any libraries you'll be using and share your reasoning for including them.

Volley: Handling client-server communication (not yet sure if this is necessary since I haven't done course on Firebase)

Describe how you will implement Google Play Services.

Google mobile ads (on every x videos I will display an interstitial ad).

Next Steps: Required Tasks

This is the section where you can take the main features of your app (declared above) and decompose them into tangible technical tasks that you can complete incrementally until you have a finished app.

Task 1: Project Setup

You may want to list the subtasks. For example:

- Setup Github repo.
- Find and setup relevant libraries.
- Setup Google Firebase

Task 2: Implement User Authentication

List the subtasks. For example:

- Create Server-side database of users on firebase (firebase has user authentication built in)
- Create LoginActivity and NewUserActivity and their relevant xml layout files and hook them up to firebase server.
- Create client-side content provider that stores user specific information synced with server database.

Task 3: Implement UI for Each Activity and Fragment

Design the UI for Main activity and implement the sliding menu.

Describe the next task. List the subtasks. For example:

- Create Main Activity UI: simple toolbar with menu button, searchbar, and profile pic.
- Configure menu button so that onClick a sliding menu appears that allows for refined browsing and navigation to various fragments.
- Configure search bar such that queries to it lead to the (currently non-existent) searchfragment.

Task 4: Implement MainActivity's Fragments

Design and implement the various fragments (browse, gamebrowser, search)

- Browse fragment utilizes a recyclerview, and LoaderManager. At this stage we worry about designing the layout only. Filling it with content and implementing actions on that content comes later
- Design gamebrowser by getting images of popular games and labeling them with their names (might make this TODO, since browsing by game isn't a necessity to the app, especially with a low user count)
- Design and implement search fragment.

Task 5: Implement Sharing

Provide intent filter so that videos can be shared to this application. For example, if on youtube and you click share, this application will appear as an option.

- Add intent filter and necessary actions
- Application handles sharing by sending URL to server-side where it is added to database with relevant information.

Task 6: Load Videos from Firebase Server and Create Video Player

- Give browse fragment the ability to query the server and get back the necessary and relevant URL's to place in its recycler view.
- Interacting with items in the recycler view launches the apps video player that loads the videos based on the stored URL. We may scrap the video player and instead just launch the video in whatever app or website it is hosted on depending on the difficulty of this request. Ideally we play youtube videos, twitch videos, and gifs locally (for example, a youtube player contained in some sort of frame on this app), but if this is not possible or if this is too difficult, I will simply launch the Youtube application and play the video there. Pressing the back button will return to my app and further browsing.

Task 6: Implement GooglePlayServices: Ads

Add as many tasks as you need to complete your app.

Submission Instructions

1. After you've completed all the sections, download this document as a PDF [File → Download as PDF]
2. Create a new GitHub repo for the capstone. Name it "**Capstone Project**"
3. Add this document to your repo. Make sure it's named "**Capstone_Stage1.pdf**"