Abdul Khalil – Project 3

Socceringo!

# Project Two Title: Socceringo

**Team Number:3**

**Team Name:Socceringo**

## Team Members:

|  |  |
| --- | --- |
| **Name** | **Email Contact** |
| Abdul Khalil | Rehmank11@gmail.com |

## Logo

**{Create a logo and place it here}**

## Project Board Website Link and Screenshots

**{Please place your link to your GitHub or Trello Project Board here}**

**{Place a baseline screenshot of your project board with your initial tasks**

## 

## Project Description (High Level):

{Describe here the purpose and intent of your project}

### Motivation

{ Describe why the team feels that the project idea can be an important commodity or service }

## User Story

{Describe below using the “As a [persona], I [want to], so that [outcome] “, to define the user story - see information for example at <https://www.atlassian.com/agile/project-management/user-stories> and detail the acceptance tests for the user story “Given…when…then” as described in <http://bit.ly/given-when-then> }

Example…

**### User Story:**

```

AS a avid football fan,

I WANT to easily and safely be able to browse football tickets for the teams i like

SO THAT I can add items to my cart for checkout.

```

**### Acceptance Test for User Story:**

```

GIVEN an e-commerce website,

WHEN I open the site,

THEN I am able to see the app’s name, logo, and content.

WHEN I click on “Create an Account”,

THEN I am directed to creating a new account which will later be saved.

IF I click the “Sign In’ button and did not previously make an account,

THEN I will not be able to access any account.

IF I do not enter a valid username or password,

THEN I will also be unable to access any account.

WHEN I click on the “Sign In” button and do already have an account,

THEN I can sign into a previously made account and view saved items.

IF I chose to browse the website without creating an account,

THEN I can continue to browse and add items to my cart without creating an

account.

BUT I will not be able to proceed to the checkout unless I make an account.

WHEN I go through the application’s inventory,

THEN I have the ability to view items based on genre or trending selections

and see available records for purchase.

WHEN I click on a club,

THEN I am given the option to view the tickets for the upcomming matches for the club

WHEN I click on a match,

THEN I will be shown the available tickets in different areas for that matches.

WHEN I select on my cart,

THEN I am able to view the items I have saved for checkout and proceed

with the order.

WHEN I go through my cart,

THEN I am able to see my saved items in a list and a total price for all items.

IF I have not placed any items in my cart,

THEN if I click on my cart, it will tell me that it is empty.

WHEN I navigate through the application on a tablet or smartphone,

THEN the application is still responsive, and I can still view and save records

to my cart.

## Description

## APIs to be Used:

{list the Names, descriptions and URLs of the APIs being used for your project below, along with the reasons on why you are using it }

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Resource | URL | HTTP Verb (GET, POST) | Action | Used For |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

## Libraries to be Used:

{List client-side CSS and JavaScript Libraries being used in this project. List the names, descriptions and URLs for these libraries, along with the reason to use it . Examples include JQuery, Materials UI}

## Packages Required:

{Not Applicable for Project One}

## Minimum Viable Product (MVP) Requirements

{Describe here your minimum features that are required to make your project functional and presentable}

## Stretch Goals

{Describe here what features you would like to see if there is time permitting in your project period of development, or post-presentation, outline features that you would like to add for future development}

## Breakdown of Tasks (Ownership by Group Member):

Documentation – Abdul Khalil

Design - Abdul Khalil

Backend – Abdul Khalil

Frontend - Abdul Khalil

Testing - Abdul Khalil

## Schedule for Completion of Tasks:

{Use a calendar to forecast estimated completion of stories and tasks, once they are defined. Have your teammates commit to it.}

|  |  |  |
| --- | --- | --- |
| **Date** | **Task** | **Notes** |
| **08/05** | Idea generation |  |
| **14/05** | Static pages for Main page, avatar, game, login/sign up page | Due on |
| **15/05** | Sample data - Main page, avatar, login/signup | Dynamic pages due on |
| **15/05** | Finalize and test GameSession class functionality | Due on |
| **16/05** | Game session | Dynamic pages due on & passed on for integration |
| **17/05** | Connect game logic to IndexedDB and graphql | Due on |
| **…** | … | **…** |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  | READY MVP |  |
|  |  |  |
|  |  |  |
|  | PROJECT POLISH |  |

## Schema

{ Describe your organization of your data used in the website, either in diagrams or text }

Example:

|  |  |  |  |
| --- | --- | --- | --- |
| Model | Properties / Fields | Type | Details |
|  |  |  |  |
| User | firstName | String |  |
|  | lastName | String |  |
|  | email | String |  |
|  | password | String (scrambled) |  |
|  |  |  |  |
| UserList | User (many) | Array (list) |  |
|  |  |  |  |
| Product | Name | String |  |
|  | Description | String |  |
|  | Price | Number |  |
|  | SKU | Number |  |
|  | ID | String |  |

Etc.

## Associations:

Example:

* A userList has many users
* ProductCategories has a list of individual Product Category information
* A Product Category has one or more products

Etc.

## Validations:

{Describe what forms will require validations, what fields within a form are being validated, and what type of validation is required}

Example:

New User Form

* Required Fields
  + First Name (text)
  + Last Name (text)
  + Email (valid email address)
  + Password (minimum, maximum, character type)

Search on Map Form

* Required Fields
  + City (text)
  + Province / State / District (text)
  + Country (text)

Etc.

## Migrations and Seed Information

{Describe here any initial data that you might need to initially apply (populate) before your website will work. For example , a list of “default users” }

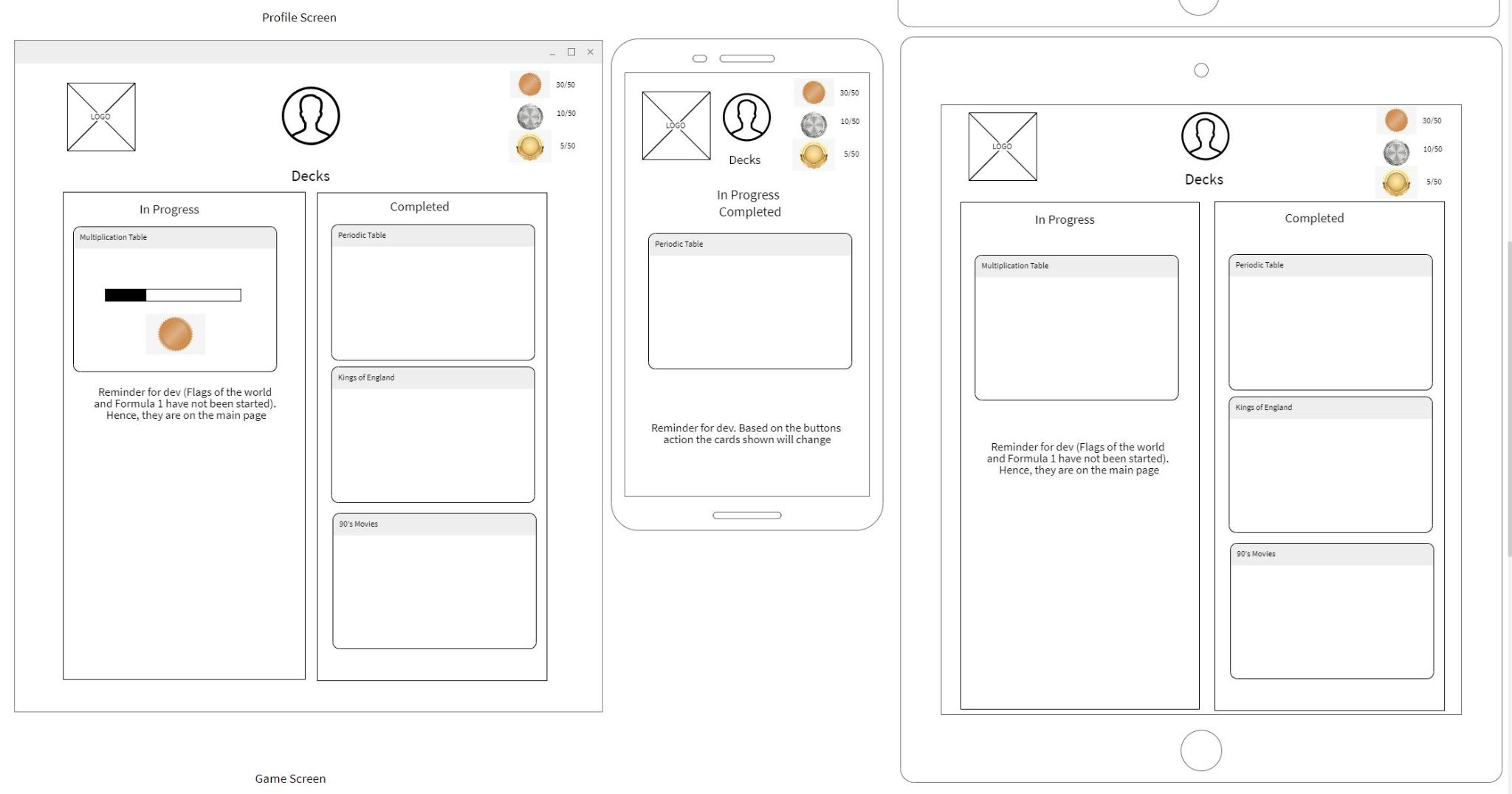
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| User | firstName | lastName | email | password | status |
|  | John | Doe | [john@doe.com](mailto:john@doe.com) | P@$$w0rd! | Active |
|  | Sally | Smith | [sally@smith.com](mailto:sally@smith.com) | T1geR\_1IlY | Inactive |
| … | … | … | … | … | … |

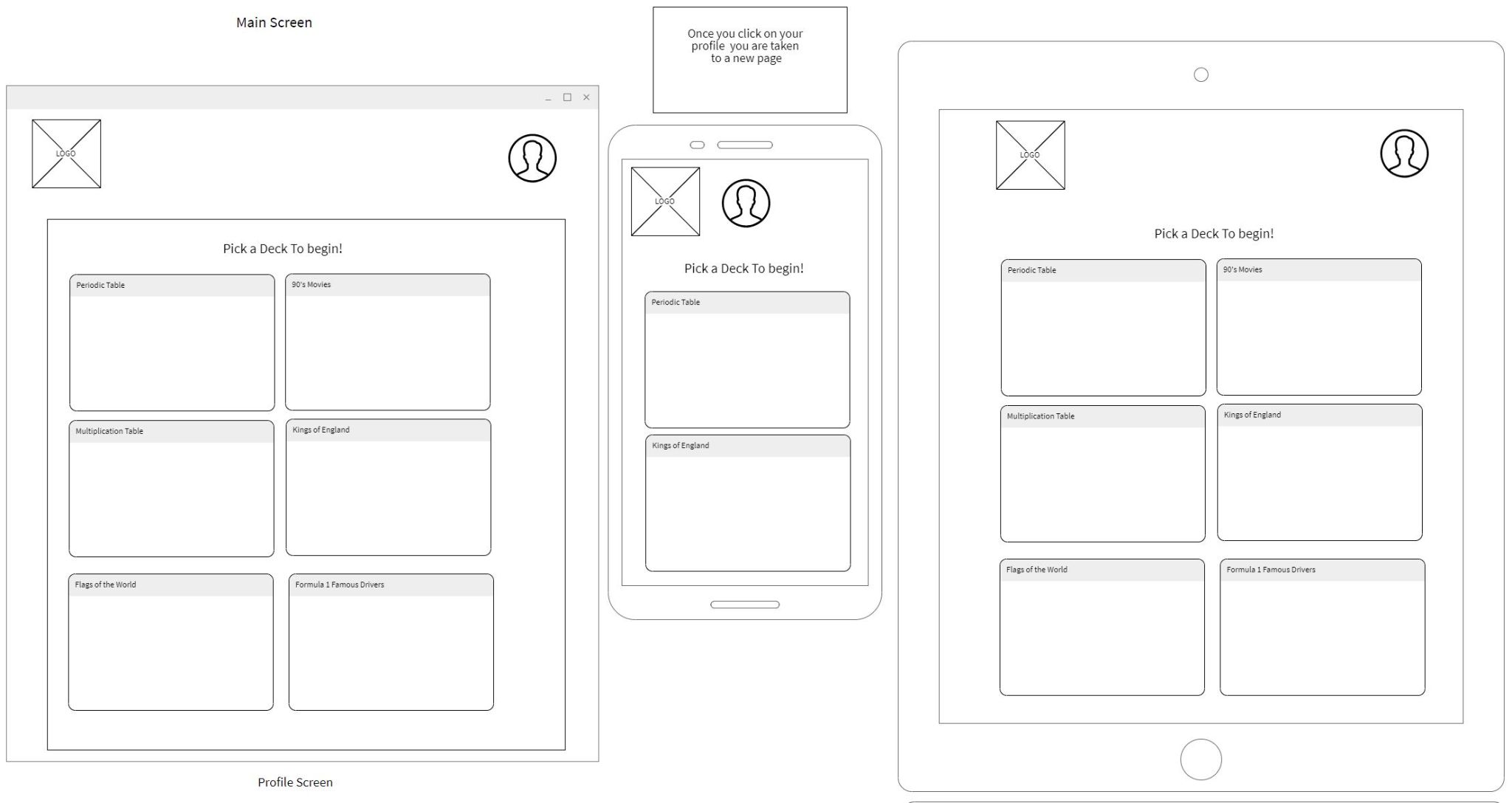
## Wireframes:

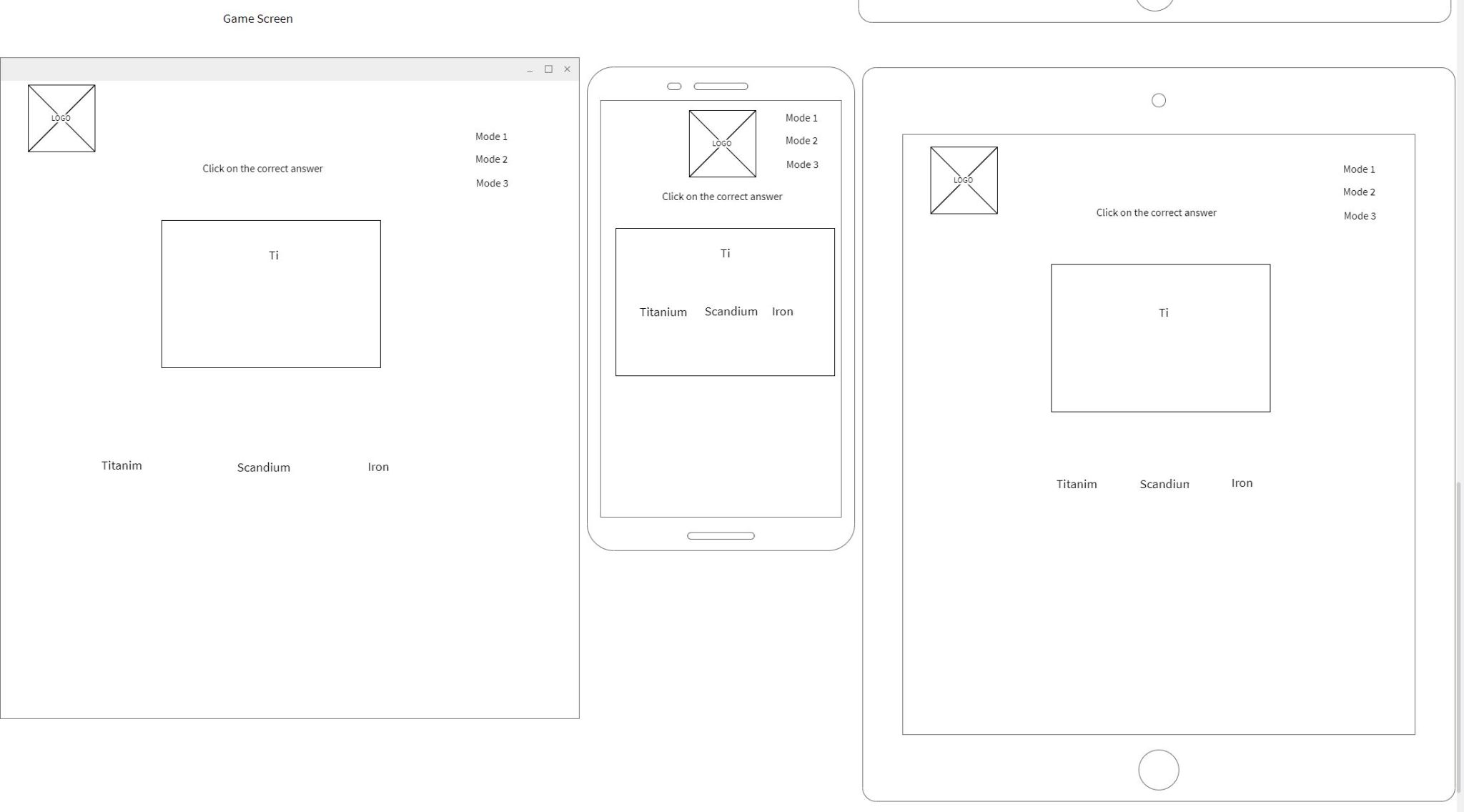
{ A set of DETAILED screen-by-screen design layouts with annotations describing all UI/UX components and all data relevant to the screen}

Acceptable formats:

* Scanned or photographed, neat and clean sketches, or
* UX/UI images created from tools such as balsamiq (<https://balsamiq.com>) , Lucidchart, PowerPoint, word, mockplus, Draftium etc.







## Website “Components” and/or “Sections”

{ Based on your wireframes, describe the common components on the website “navigation”, “forms”, “headers”, “footers” either with images / and or content and describe the intention / usage of the components. In addition, describe the primary “views” or “sections” that your project will have, for example “About Us”, “Products and Services”, etc.

}

## Git Workflow, Website Directory Structure:

{describe your workflow and methodology that all team members will abide by when using Git, AND describe in text or imagery the directories and filenames and their descriptions that will be used for the project}

Example:

### GIT Workflow

Production Branch Name: Main

Development Branch Name: develop

All developers will create branches off develop according to the naming convention:

{

example:

***feature/lastname/ticketnumber/task i.e. feature/Apostol/001/navigation-bar***

***docs/lastname/ticketnumber/task i.e. docs/Wayne/020/update-README***

***bug/lastname/ticketnumber/bug-desc i.e. bug/Kent/030/fix-navigation-bar-color***

}

### Code Review and Merge Requests

Identify Name(s) of people doing code review and approving Merge Requests: (lists names here)

* Team members should “pair up” to perform code reviews and merge requests
* At the halfway point, new pairs should be recreated
* If there is an odd number of participants in the team, then the extra member can float between different pairs to code review,

|  |  |  |
| --- | --- | --- |
| Week One | Team Member | Team Member |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
| Week Two |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

### Directory Naming and Structure; Filenames and descriptions

Only list the important files and folders.

|  |  |  |  |
| --- | --- | --- | --- |
| Directories | name | description | detail |
|  | images | Stores images |  |
|  | images/home | Images in home page |  |
|  | images/products | Product images |  |
|  | images/bio | Images for bio |  |
|  | css | Stores css | Main.css is principal stylesheet |
|  | css/nav | Navigation style |  |
| … | … | … | … |
|  |  |  |  |
| Files |  |  |  |
|  | Index.html | home |  |
|  | Contactus.html | Contact us page |  |
| … | … | … | … |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

## “Stuck Time” Agreement:

The team must identify and agree to a time frame that a team member must use as a limit before reaching out to (a) fellow team members and then (b) the instructional team (in-class) for support. That way, *no one should be stuck for ‘days’ or ‘many hours’ on a task.*

We the team members of (insert project team number or team name here) agree that the following period of time of investigation (with evidence) is allowed before identifying themselves as being “stuck” or “bottlenecked”, in which case, the team member can then seek the assistance of other team members, the instructional team, or AskBCS for assistance.

Period of time permitted before asking to seek assistance (hours): \_\_\_\_\_\_\_\_\_\_ hours

As agreed upon by the following team members (name and signature below)

|  |  |
| --- | --- |
| Name | Signature |
| Jane Doe |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

## Retrospectives – To Be Handed In At the Project Midpoint

Each team is to hold a *retrospective* meeting at the half-way point of the project period.

References:

* [What is a sprint retrospective - EasyRetro former FunRetro](https://easyretro.io/blog/what-is-a-sprint-retrospective/#:~:text=To%20facilitate%20this%2C%20the%20retrospective%20is%20based%20on,commit%20to%20doing%20differently%20in%20the%20next%20sprint%3F)
* [What is a retrospective? – All About Retrospectives (retromat.org)](https://retromat.org/blog/what-is-a-retrospective/)

Each team is to submit to the instructional team a short three paragraph Microsoft Word or PDF document at **the mid-point** of the project that describes the following:

* What worked well in the sprint?
* What didn’t work well and could be improved?
* What will we commit to doing differently in the next sprint?

## GitHub Repository Link, Production Website Link Submission Dates

To be filled in by the project team under advisement of the instructional team (and set out in Canvas)

* Presentation Date
* Project Submission Due Date
* Ideal Submission Date

## Additional Notes: