SW Engineering CSC648/848

Mesekai

real-time full-body 3D avatar and virtual room web application

Section 04 Team 04

Caelen Wang (Team Lead)

Vasudevan Venugopal (Frontend Lead)

Matthew Madore (Backend Lead)

Jose Miguel Atienza (Scrum Master)

Eugene San Juan (Git Master)

Mohammad Abdelrahman

"Milestone 2"

October 22, 2021

Revision History Table

Data Definitions

Name	**Definition**	**Usage**
User	Person using the application	Interact with the application through both manual controls (keyboard/mouse) and motion controls
Account	Registered account associated with User	Stores information specific to User such as login credentials
Profile	Each User has 1 Account with multiple Profiles	Stores information such as Avatar/World selection, Objects placed
Avatar	User controlled 3D character (from list of preset models)	Mimics User’s facial expression, body, and finger pose controlled through physical motion captured by webcam (visual only, use 3rd party app for voice features)
World	3D environment in which Avatar resides (from list of preset scenes)	Sandbox that User can customize, move around with Avatar, place and interact with Objects
Object	3D asset in World	Decoration to be placed in World, interactive with Avatar
Interface	In-Game user interface	Menu for User to select Avatar/World presets, Objects to spawn in and translate, rotate, and scale

Primary Data	**Sub Data**
Account	● Id ● Username ● Password ● Email ● Date of Birth
Profile	● Avatar Preset (single choice) ○ Remy, Douglas, Kate, YBot, ... ● World Preset (single choice) ○ Room, Forest, Castle, … ● Objects (multiple choice and placement) ○ [ball: [position, rotation, scale], chair: [position, rotation, scale]]
User	● Pose ○ Pose Landmark ● Hand Pose ○ Hand Landmark ● Face Mesh ○ Face Landmark
Avatar	● Skeleton ○ Bone ● Skinned Mesh ○ Morph Target

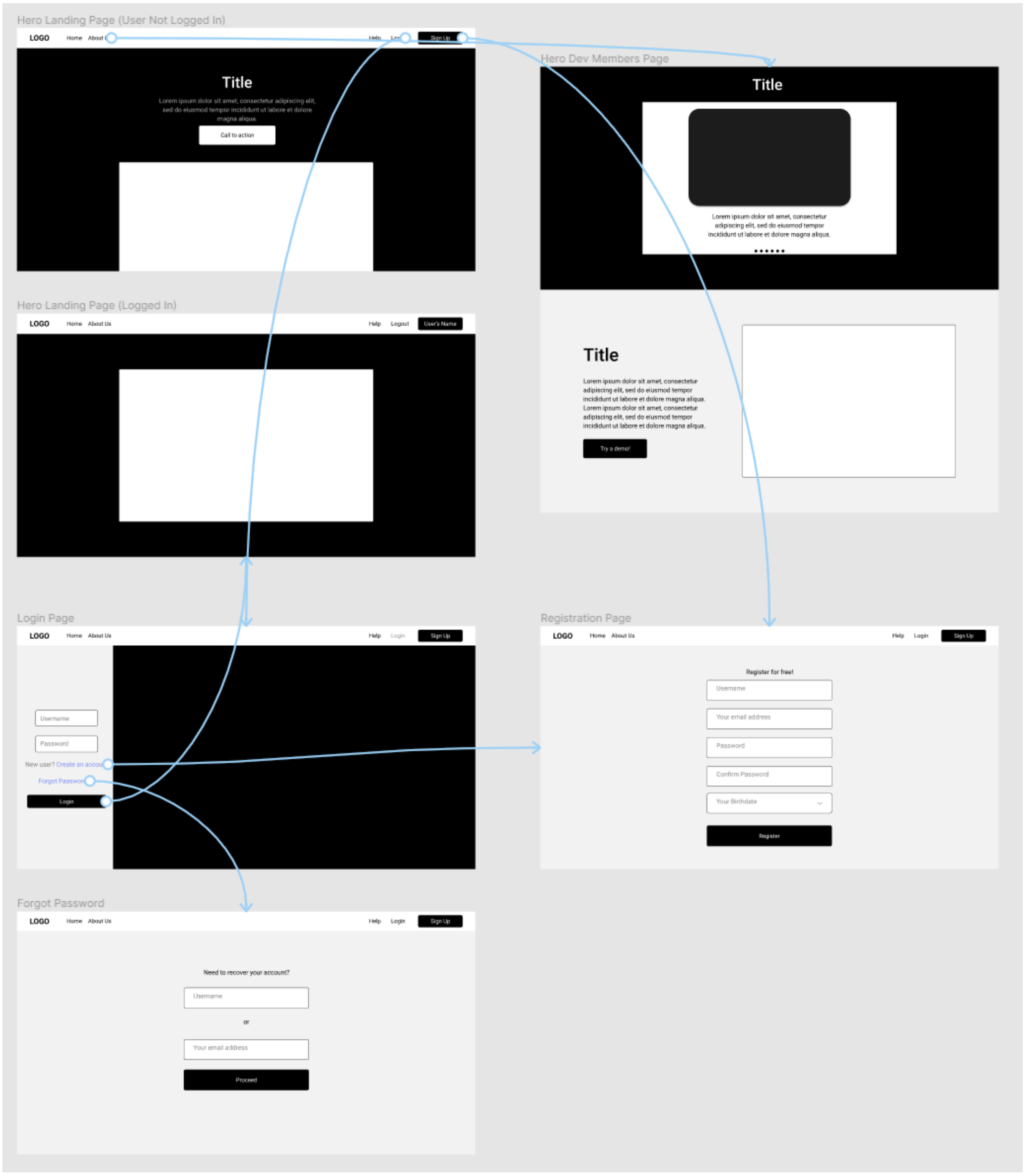
Functional Requirements

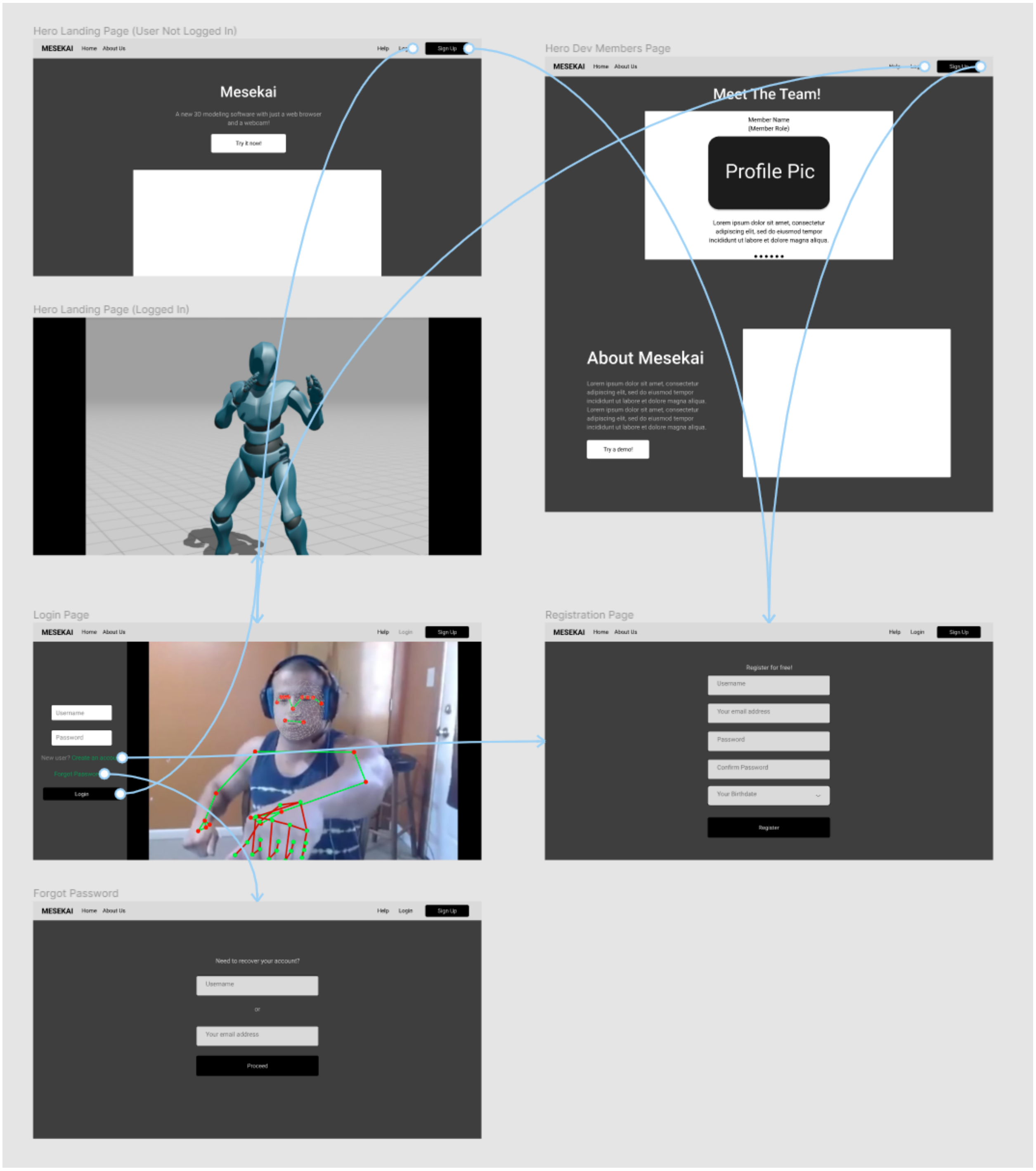
Must Have	Desired	Opportunistic

Priority	ID	Functional Requirement	Details
🟦	1	Account creation	Store User provided information such as username, email, and password in database
🟦	2	Log in and log out of account	Log in to account using credentials stored in database provided during account creation or password reset
🟦	3	Reset password to account	Link on login page that lets User enter their email address, to which a password reset link will be sent
🟦	4	Select Avatar from presets	User chooses from list of available Avatars from Interface: ● Remy ● Douglas ● Kate ● YBot ● … choice stored in Account
🟦	5	Select World from presets	User chooses from list of available Worlds from Interface: ● Room ● Forest ● Castle ● Space ● … choice stored in Account
🟦	6	Select Objects from list	Based on User choice from Interface, different Objects will be spawned in ● Chair ● Bed ● Desk ● Sword ● … choices stored in Account

🟦	7	Avatar / Object Interaction	● Avatar pushes Object through collision ● Avatar picks up Object using fingers ● Object physics
🟦	8	Transform Objects around World	Click, drag, drop Objects to be placed in desired location. Translation, rotation, scale stored in Account
🟩	9	Custom Avatars	Users can upload their own 3D character models. Tutorial to show how to create model to meet system requirements
🟩	10	Custom Objects	Users can upload their own 3D assets. Tutorial to show how to create assets to meet system requirements
🟥	11	Adjust hardware settings	Choose among ● Level 0: CPU only ● Level 1: mid-tier GPU ● Level 2: high-end GPU based on device specifications. Lower levels reduce quality of tracking and graphics but improve performance
🟥	12, 13	Adjust tracking settings	Toggle on or off for ● facial expressions ● full/upper body tracking ● finger tracking based on desired effect
🟩	14	Adjust movement settings	Toggle between ● real-time tracking ● keyboard movement
🟩	15	Adjust perspective	Choose from ● 3rd person mirrored (default) ● 3rd person aligned ● 1st person
🟩	16	Multiplayer	Invite other Users to your World using room code. Synced objects and physics

UI Mockups and Storyboards


Hero Landing Page (User Not Logged In)
LOGO
Home
About Us
Help
Login
Sign Up
Title
Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.
Call to action
Hero Dev Members Page
Title
Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.
Title
Try a demo!
Hero Landing Page (Logged In)
Logout
User's Name
Login Page
Username
Password
New user? Create an account
Forgot Password
Login
Registration Page
Register for free!
Your email address
Confirm Password
Your Birthdate
Register
Forgot Password
Need to recover your account?
or
Proceed



A new 3D modeling software with just a web browser and a webcam!

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.

Date	10/18/21 (Monday)
Duration	5:30 pm - 7:00 pm PST
Note Taker	Jose Miguel Atienza
Discussed Items	● **Discussed Milestone 2** ● **Started wireframe design on Figma**
Tasks	● Finish wireframe ● Start on GUI design/prototype

Date	10/21/21 - 10/22/21 (Thursday/Friday)
Duration	10:00 pm - 1:30 am PST
Note Taker	Jose Miguel Atienza
Discussed Items	● **Discussed Milestone 2** ● **Finished and demoed wireframe**
Tasks	● Complete GUI design/prototype ● Demo GUI design

High level Architecture, Database Organization

User:

Account:
Id (String)
Username (String)
Password (String)
Email (String)
Date of Birth (String)

Profile 1:	**Profile 2:**	**Profile 3:**
Avatar (String)	Avatar (String)	Avatar (String)
World (String)	World (String)	World (String)
Objects (List)	Objects (List)	Objects (List)

Each User can create one Account with a unique username and email. This Account can create multiple Profiles storing different Avatar/World/Object configurations. For example, an User can create a Profile that has a human Avatar in a forest World with trees and rock Objects. They can also create another Profile with an alien Avatar in a space World with spaceship and helmet objects, and switch between the Profiles

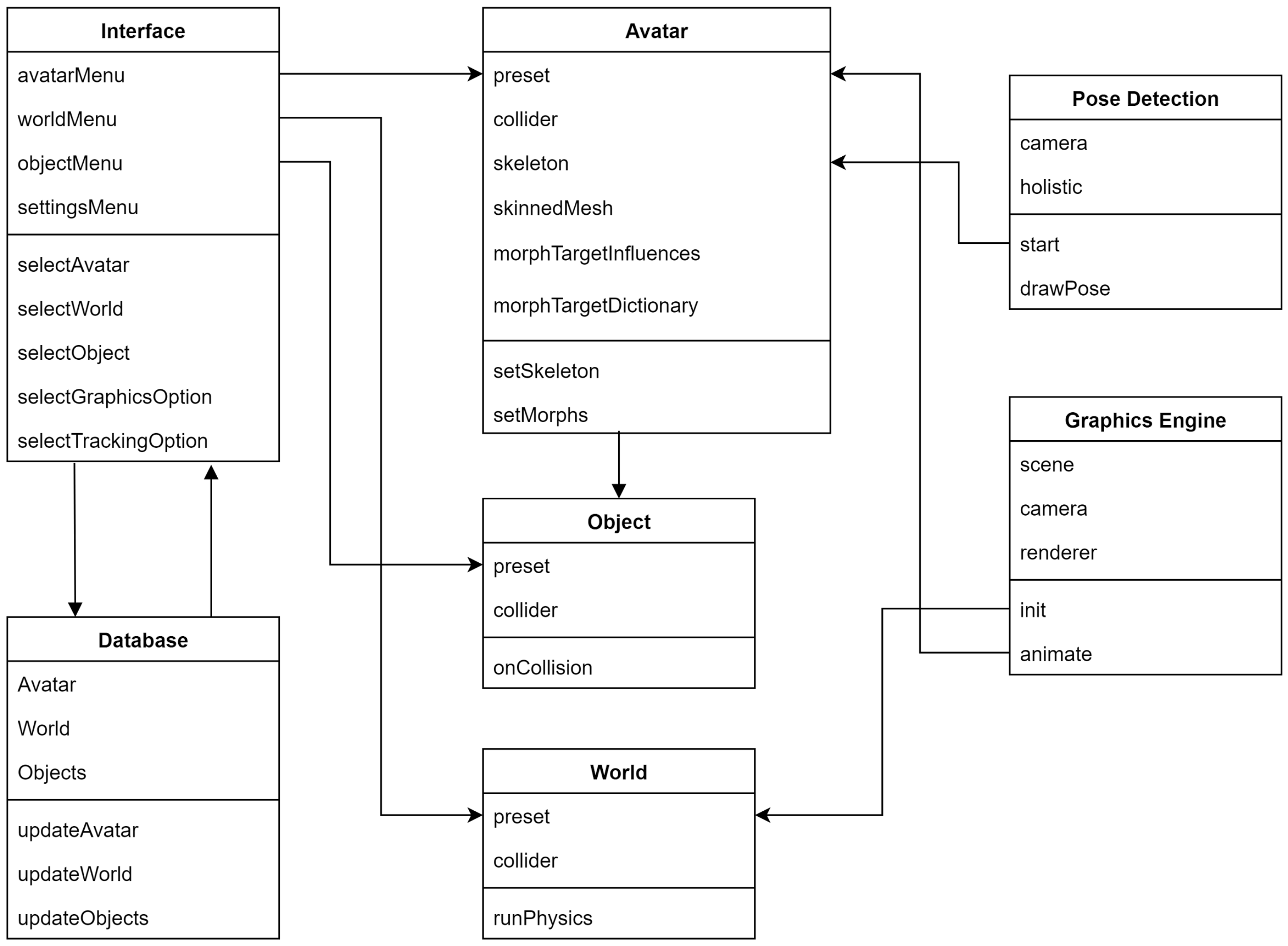
Add/Delete/Search Architecture	**Functional Requirement**
Add/Delete/Search for Users	When users register
Search/Display Avatar	Users can browse through different Avatars and load them to their World
Search/Display World	Users can browse through different Worlds
Search/Display Objects	Users can add Objects to their World which will include model path, position, rotation, and scale
Change and update password	Users will be able to update their password
Change and update username	Users will be able to update their username
Change and update email	Users will be able to update their email

Technical feasibility of those DB operations:

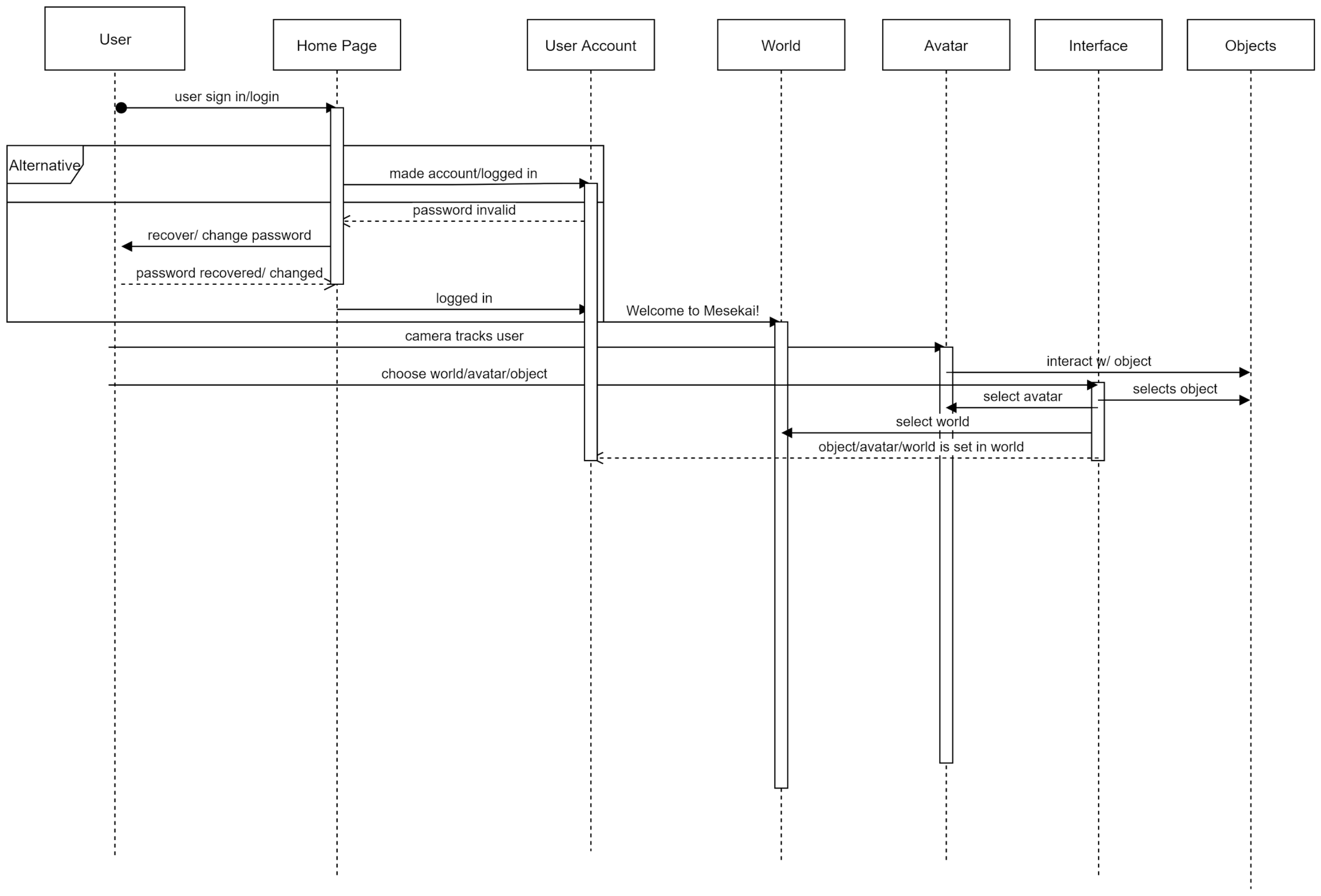
For our database we have two tables to represent both the User's Account information and the Profiles they have created. The tables will be linked with an id key to access the user profile. Our database is on Google Cloud (Firestore) which is NoSQL. Using Express as our Software Tool, we created our APIs on the backend, adding functionality which includes: Add/Delete/Search for Users, Search/Display Avatar, World, Objects and Update Password/Email/Username. The frontend establishes a connection and makes calls to the backend using Axios and Next.

High Level UML Diagrams

Class Diagram:



Sequence Diagram:



Identify actual key risks for your project at this time

Some members are less familiar with certain technologies. Therefore we implemented a subteam structure in which 3 experienced members lead 3 less experienced members in their assigned area of expertise. We use a shared Google Doc as our project management tool. That document is updated biweekly and let's us know what we must get done that week. So far we have been meeting those deadlines on time. For our schedule we meet up on Thursday nights when everyone is free. If other teams must work together to make something work we let each other know if we could meet on another day and figure it out. Everyone is able to meet regularly and consistently and we all keep a good pace on our tasks. In the case that a group member or members are not present or can not make it to a meeting, then we would message them about what they had missed and what is due by this week. If someone is not coming to the meetings or not keeping communications with the group then the Scrum Master will reach out to them and if he cannot reach out to them then the Scrum Master will notify the professor.

Project Management

For Milestone 2 we split the tasks among the 3 subteams: frontend, backend, and core. Each subteam is headed by the Frontend Lead, Backend Lead, and Team Lead, respectively. Each subteam is responsible for meeting their weekly goals and communicating it to the entire team during scrum meetings. When there are confusions about progress, the Team Lead and Scrum Master are responsible for resolving any lingering issues. We use a shared Google Doc to document weekly goals as well as sprint stories.