

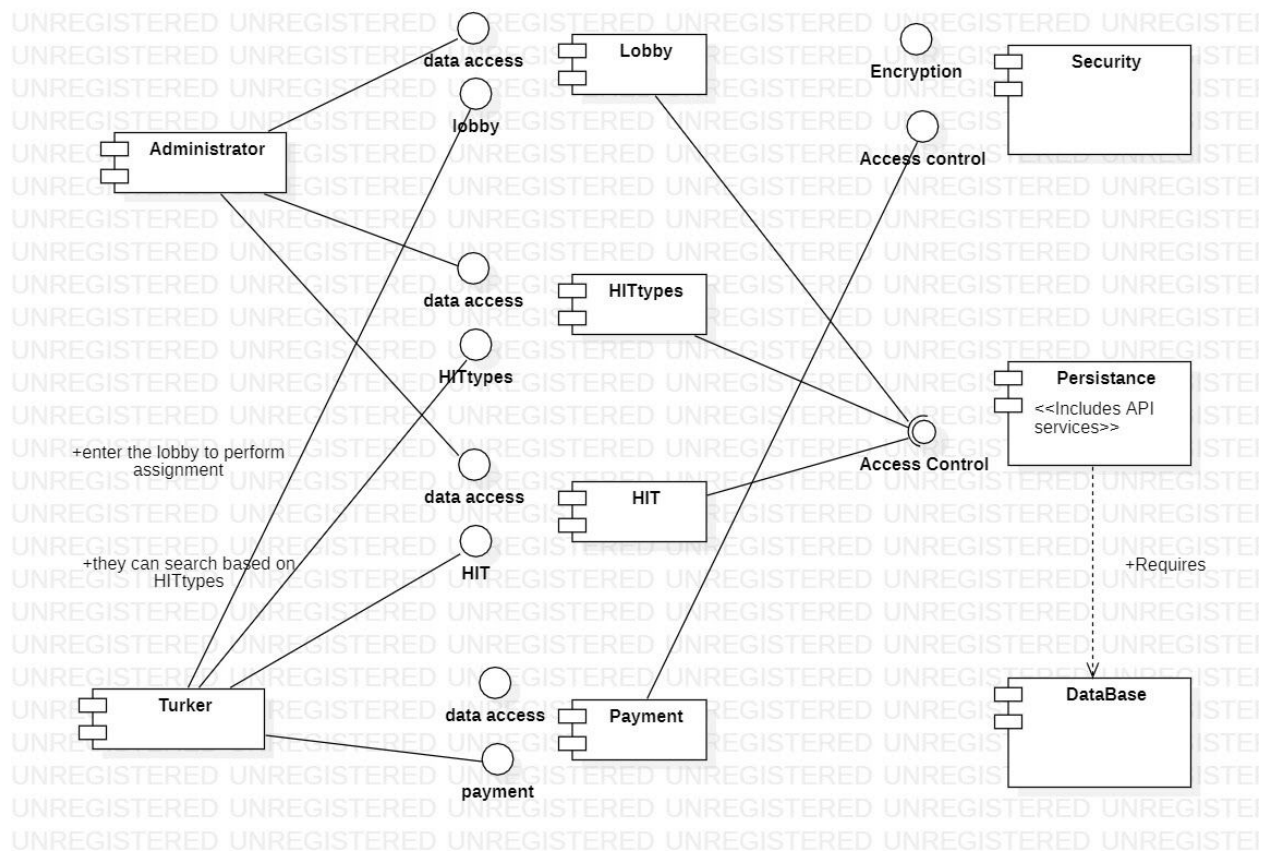
# MTurk Framework Design Document

By Akshat, Jay, Jonathan, Muhammad, Walker, Xinjian

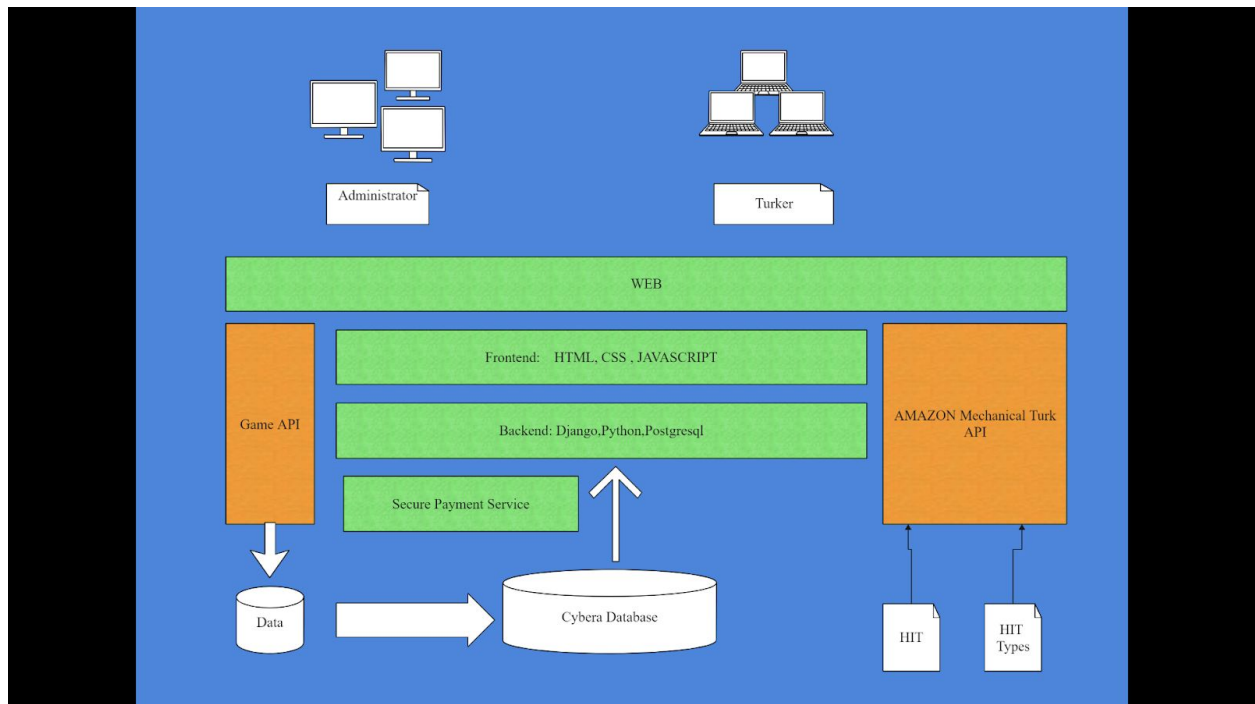
February 6, 2021

## High-Level Architecture

This is a component diagram to include the working of the 4 important components in our application. This shows who has the data access to the 4 components and who is just a user.

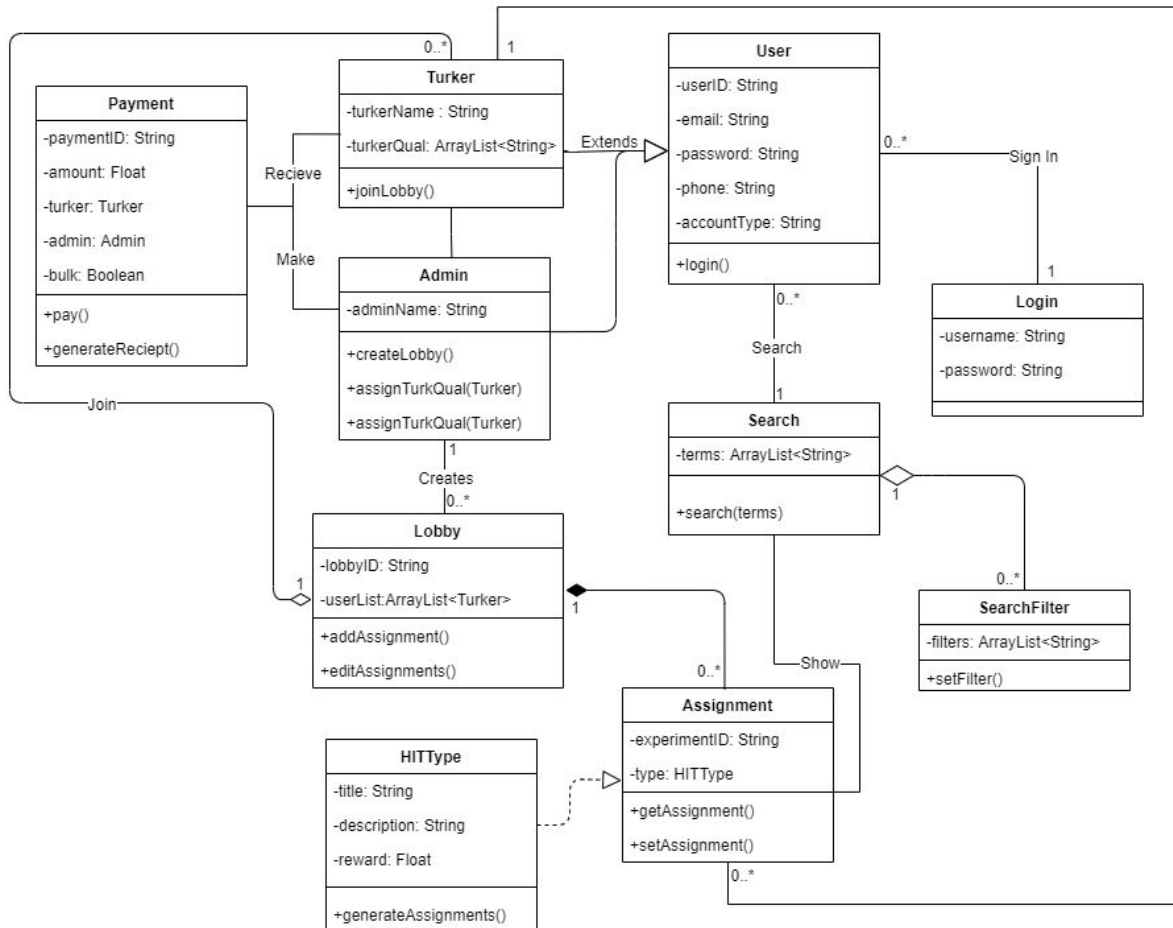


This is an architecture diagram that defines all the programme components that we will use to make our application.



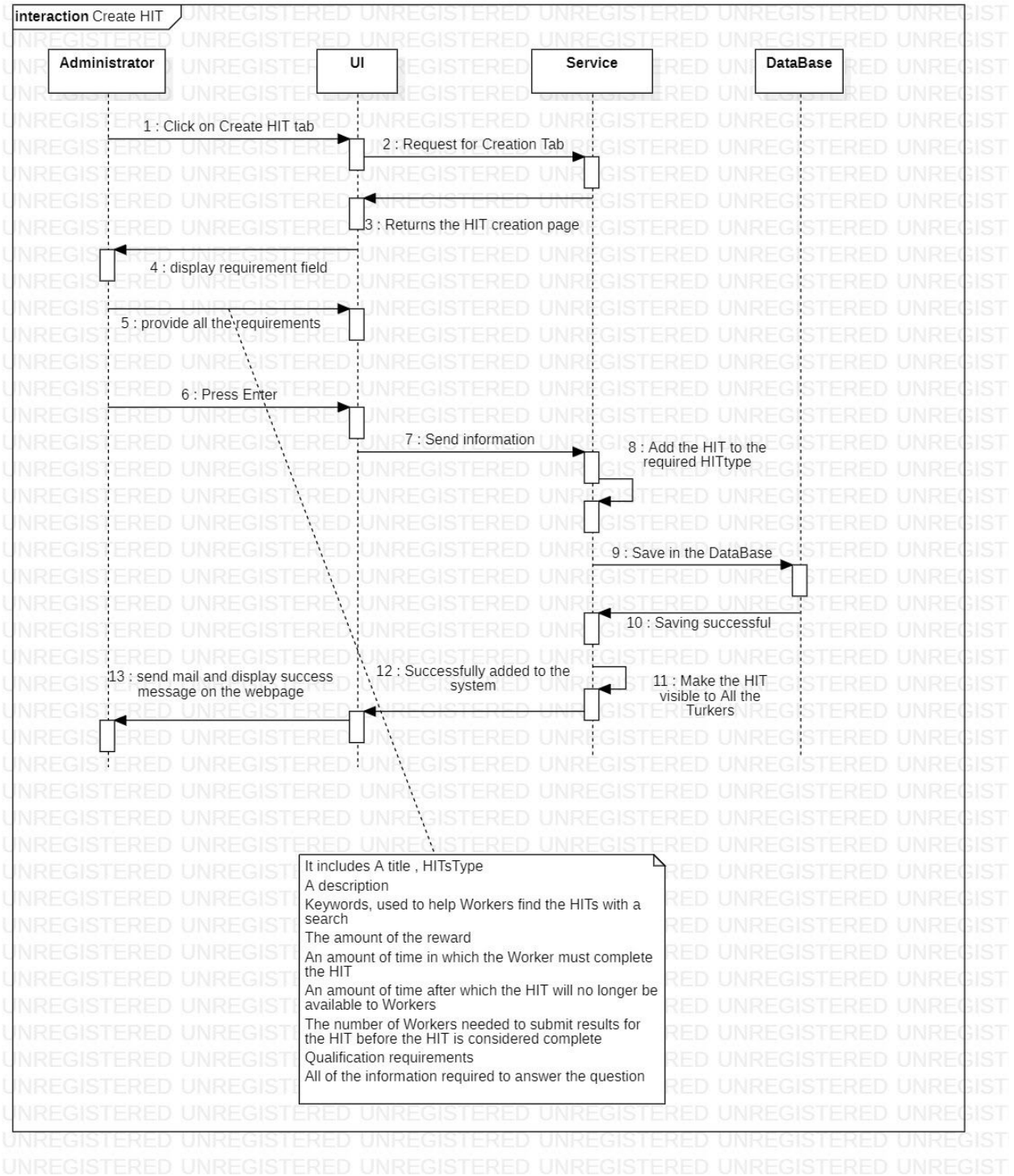
## Major Data Elements

The UML class diagram lays out all the major data elements and their relationships that we expect to include in our framework.

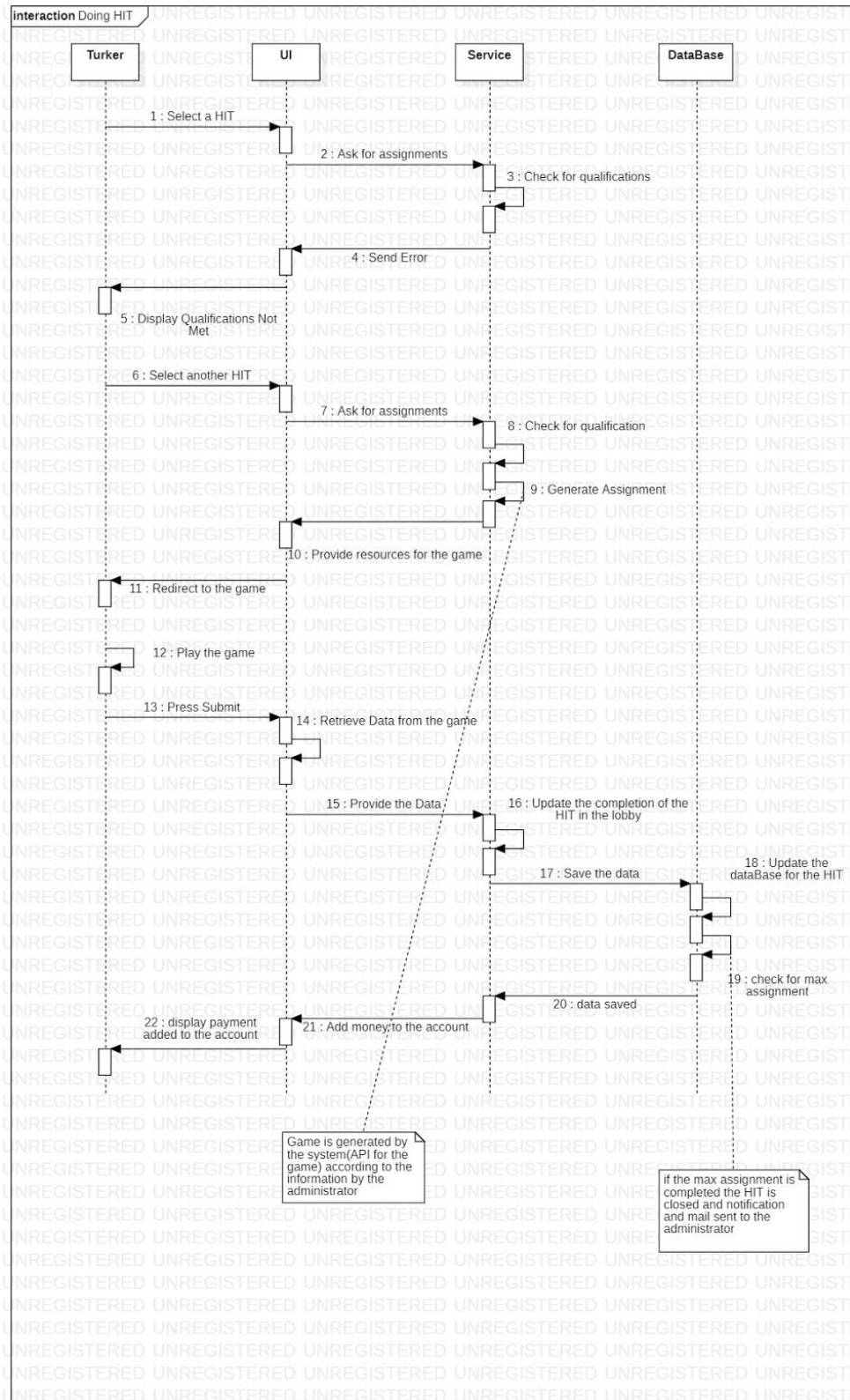


# Interaction Scenarios

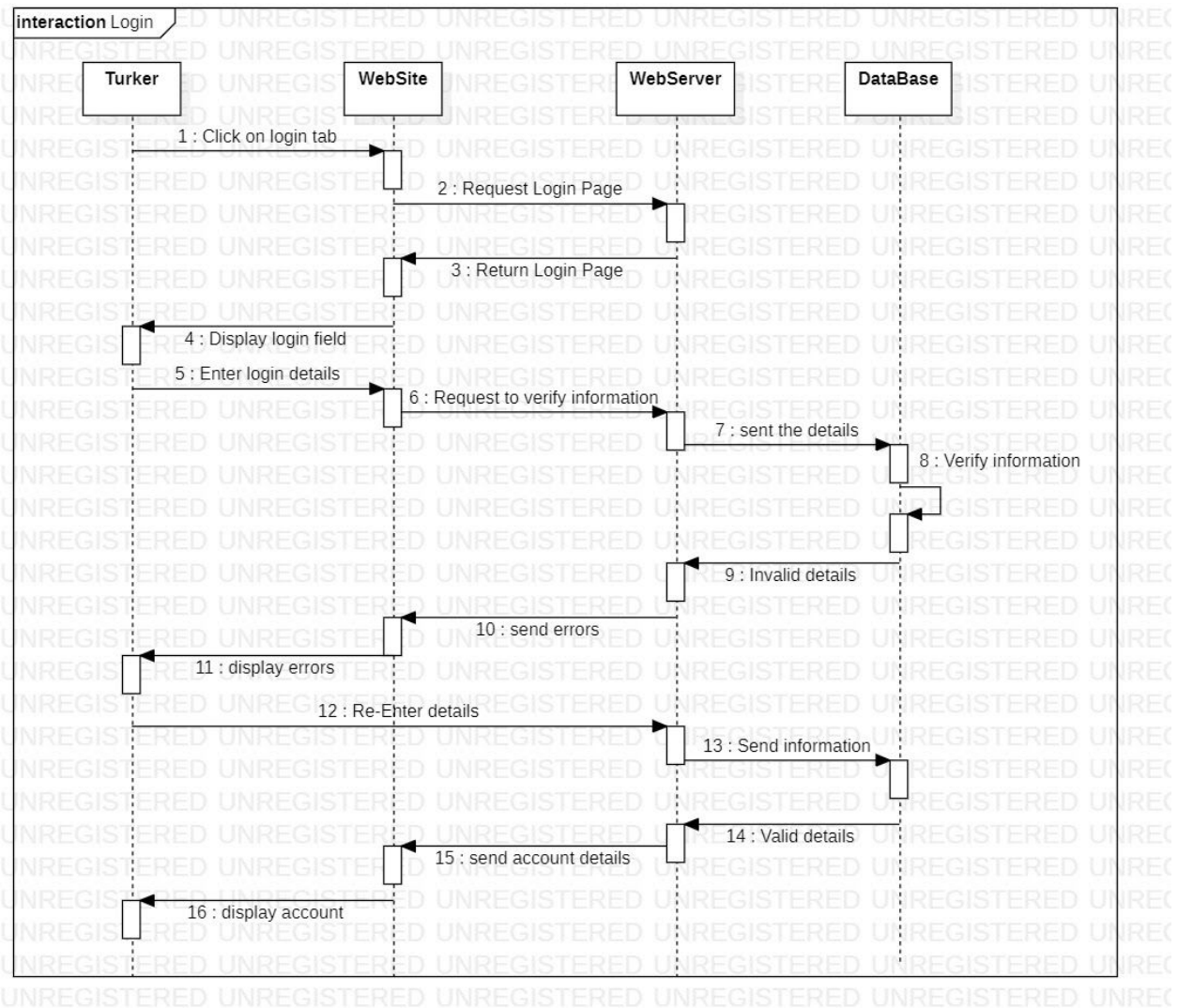
**Create HIT:** It is to show the interaction between administrator, UI, Services(includes the API) server which takes place when the administrator creates HIT.



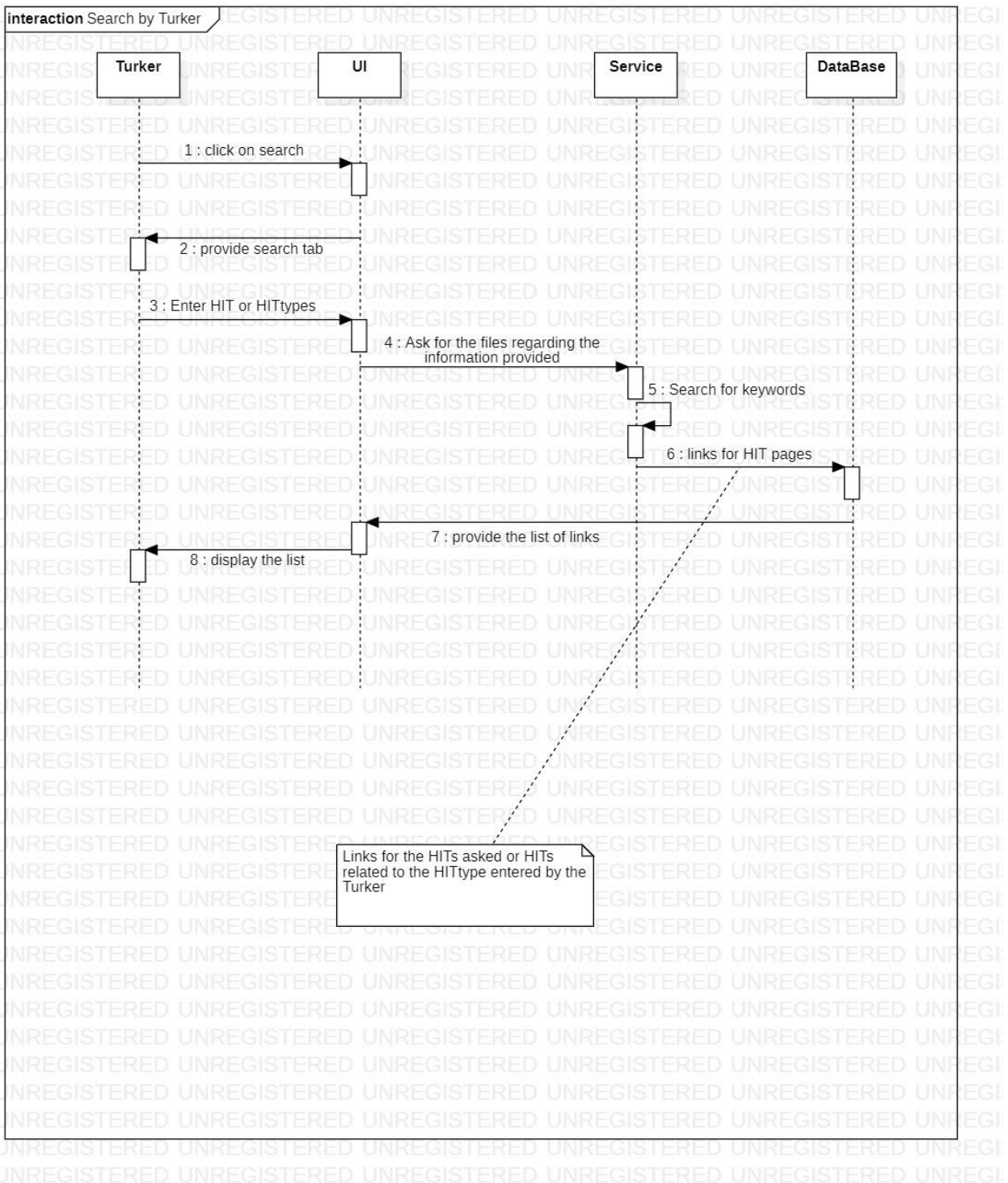
**Doing HIT:** It is to show the interaction between Turker and services ( including the API ) with the UI as an intermediate when the turker decides to do a HIT.



**Search HIT:** It is to show the interaction between Turker and Services (including the API) with UI as an intermediate when a Turker searches for HIT or search based on HITtypes



**Login:** It is to the interaction that takes place when administrator/Turker logs into the application.



# Low-Fidelity User Interface

This interface is the basic design of the login page.

A low-fidelity wireframe of a login page. It features a browser window with a back arrow, forward arrow, and address bar containing 'website.com'. The main content area is titled 'Log in'. Below the title are three input fields: 'User Name' with the placeholder 'Enter your user name', 'Email Address' with the placeholder 'Enter your email Address', and a 'Role' dropdown menu with the placeholder 'Select your Role' and two options: 'Administrator' and 'Turner'. Below these fields is a 'SUBMIT' button.

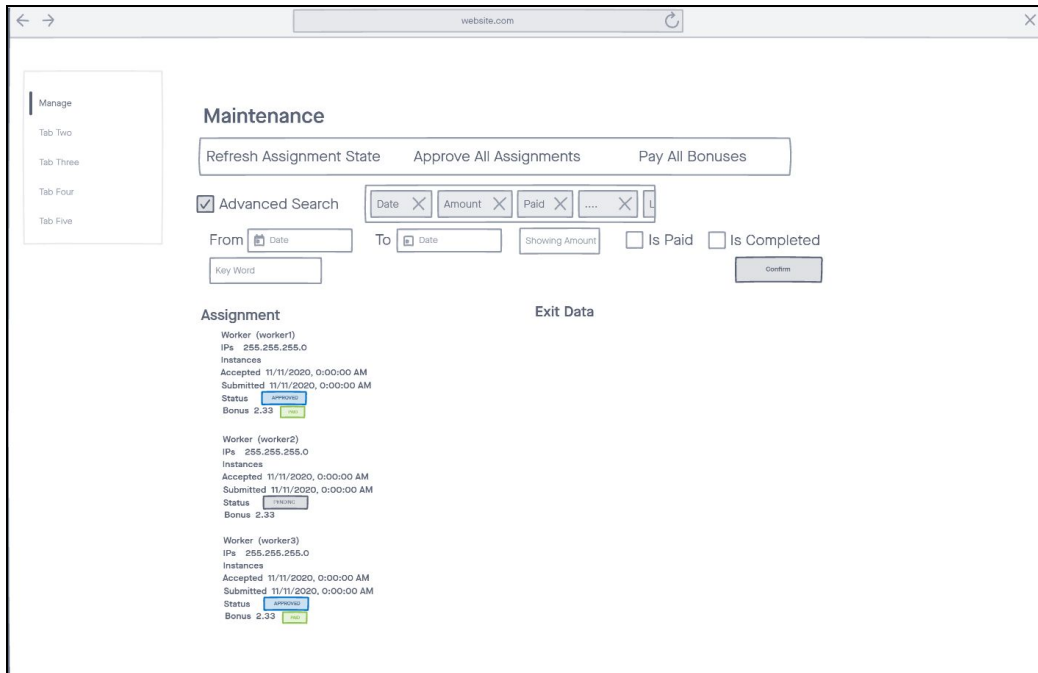
This interface is the basic design of the Dashboard page.

A low-fidelity wireframe of a dashboard page. It features a browser window with a back arrow, forward arrow, and address bar containing 'website.com'. On the left is a sidebar with a 'Dashboard' link and four tabs: 'Tab Two', 'Tab Three', 'Tab Four', and 'Tab Five'. The main content area is titled 'HITS' and contains a search bar with the placeholder 'Search For HITS'. Below the search bar is a table with the following columns: 'Requester', 'Title', 'HITS', 'Reward', 'Time', and 'Actions'. The table contains six rows of data. Above the table are two filter panels. The 'Filter' panel on the left has a 'Sort by' dropdown, a 'is Qualified' checkbox, and a 'Payment from' dropdown. The 'Sort By' panel on the right has a 'Sort By' dropdown with three options: 'Creation Date', 'HITS', and 'Reward'. A green arrow points from the 'Sort by' dropdown in the 'Filter' panel to the 'Sort By' dropdown in the 'Sort By' panel.

Requester	Title	HITS	Reward	Time	Actions
Requester1	Some Request Title1	10,000,001	\$ 0,08	5h ago	Accept Work
Requester2	Some Request Title2	900,001	\$ 0,03	2d ago	Accept Work
Requester3	Some Request Title3	888,888	\$ 0,12	1y ago	Not Qualified
Requester4	Some Request Title4	3,001	\$ 0,02	8h ago	Accept Work
Requester5	Some Request Title5	1,999	\$ 0,01	4d ago	Accept Work
Requester6	Some Request Title6	800	\$ 0,90	2y ago	Not Qualified



This interface is the basic design for the maintenance page.



The Maintenance page interface includes a sidebar with a 'Manage' section and tabs for 'Tab Two', 'Tab Three', 'Tab Four', and 'Tab Five'. The main content area is titled 'Maintenance' and features three action buttons: 'Refresh Assignment State', 'Approve All Assignments', and 'Pay All Bonuses'. Below these is an 'Advanced Search' section with a checked checkbox and several filters: 'Date', 'Amount', 'Paid', and a dropdown menu. There are also input fields for 'From' and 'To' dates, a 'Showing Amount' field, and checkboxes for 'Is Paid' and 'Is Completed'. A 'Key Word' input field and a 'Confirm' button are also present. The 'Assignment' section lists three workers (worker1, worker2, worker3) with their respective IP addresses, instance counts, acceptance and submission timestamps, status (pending or approved), and bonus amounts. An 'Exit Data' button is located to the right of the assignment list.

Manage

Tab Two

Tab Three

Tab Four

Tab Five

### Maintenance

Refresh Assignment State Approve All Assignments Pay All Bonuses

☒ Advanced Search

Date X Amount X Paid X ... X

From  To  Showing Amount ☐ Is Paid ☐ Is Completed

Key Word  Confirm

#### Assignment

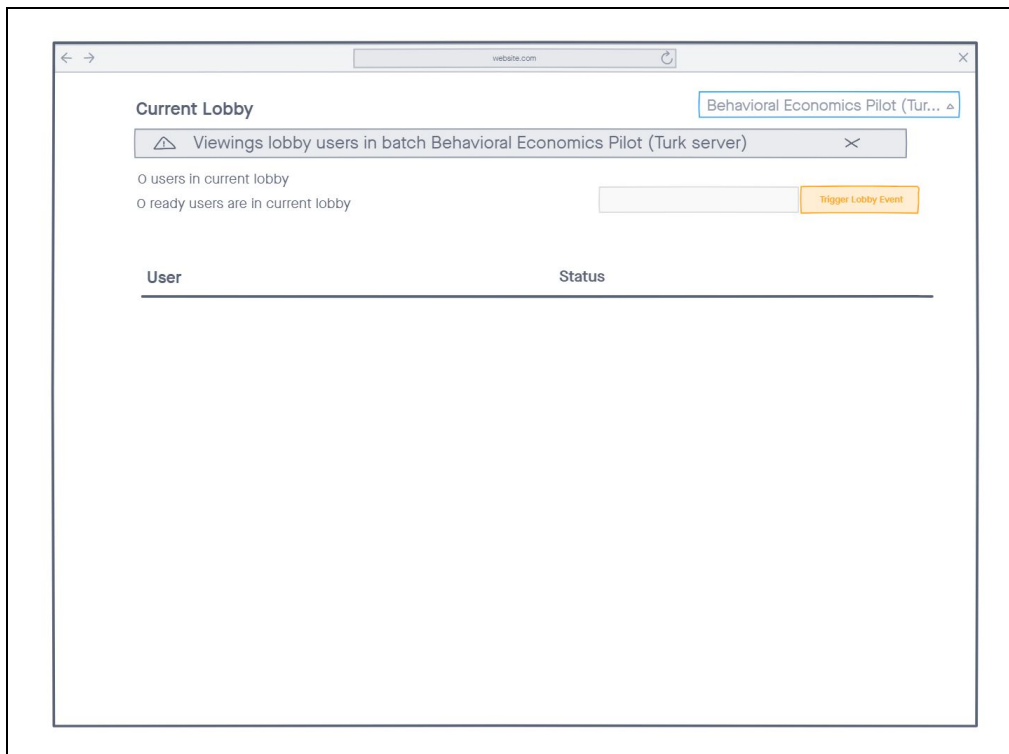
Worker (worker1)  
IPs 255.255.255.0  
Instances  
Accepted 11/11/2020, 0:00:00 AM  
Submitted 11/11/2020, 0:00:00 AM  
Status pending  
Bonus 2.33

Worker (worker2)  
IPs 255.255.255.0  
Instances  
Accepted 11/11/2020, 0:00:00 AM  
Submitted 11/11/2020, 0:00:00 AM  
Status approved  
Bonus 2.33

Worker (worker3)  
IPs 255.255.255.0  
Instances  
Accepted 11/11/2020, 0:00:00 AM  
Submitted 11/11/2020, 0:00:00 AM  
Status approved  
Bonus 2.33

Exit Data

This interface is the basic design for the Lobby page.



The Lobby page interface shows a 'Current Lobby' section with a dropdown menu displaying 'Behavioral Economics Pilot (Turk server)'. Below this is a button labeled 'Viewings lobby users in batch Behavioral Economics Pilot (Turk server)'. The page indicates '0 users in current lobby' and '0 ready users are in current lobby'. A 'Trigger Lobby Event' button is located next to the ready users count. At the bottom, there is a table with two columns: 'User' and 'Status'.

### Current Lobby

Behavioral Economics Pilot (Turk server)

Viewings lobby users in batch Behavioral Economics Pilot (Turk server)

0 users in current lobby

0 ready users are in current lobby

Trigger Lobby Event

User	Status
------	--------