

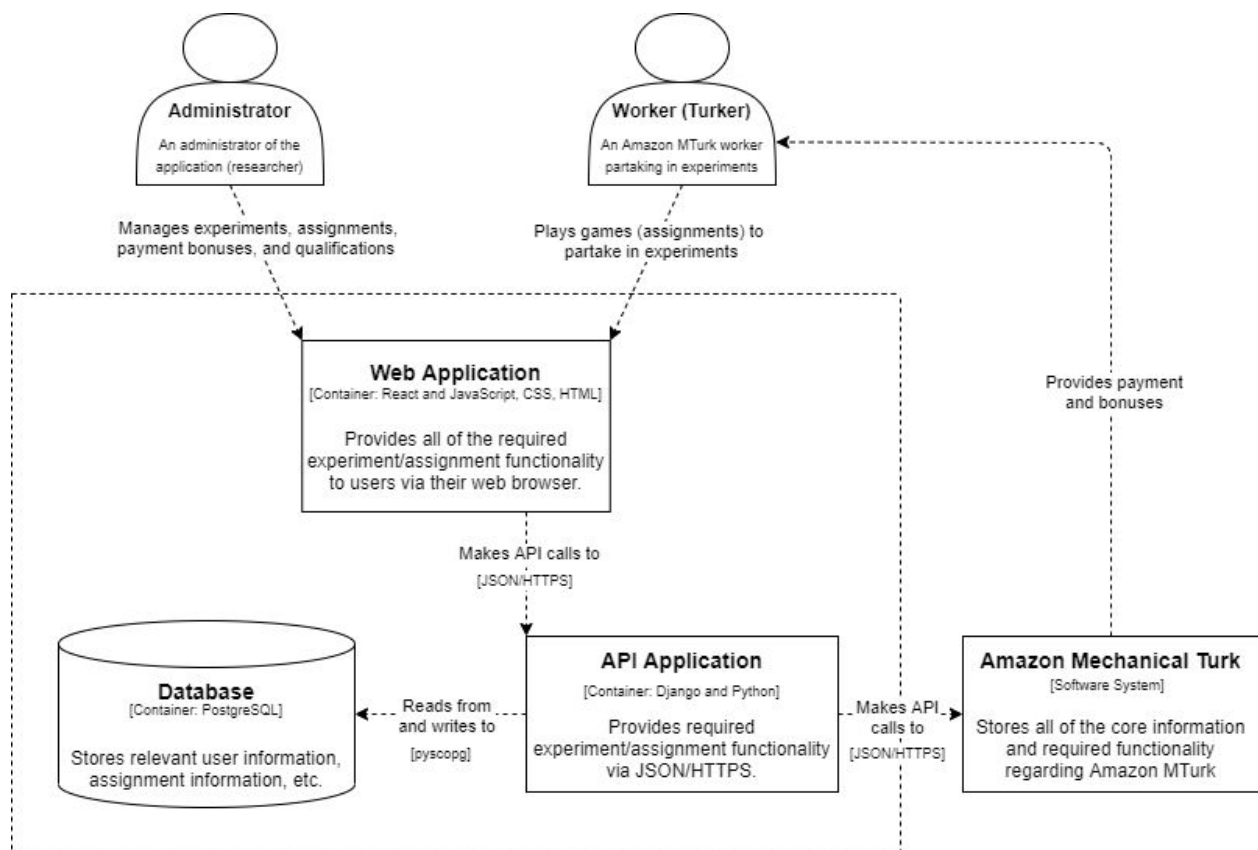
# MTurk Framework Design Document

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

February 6, 2021

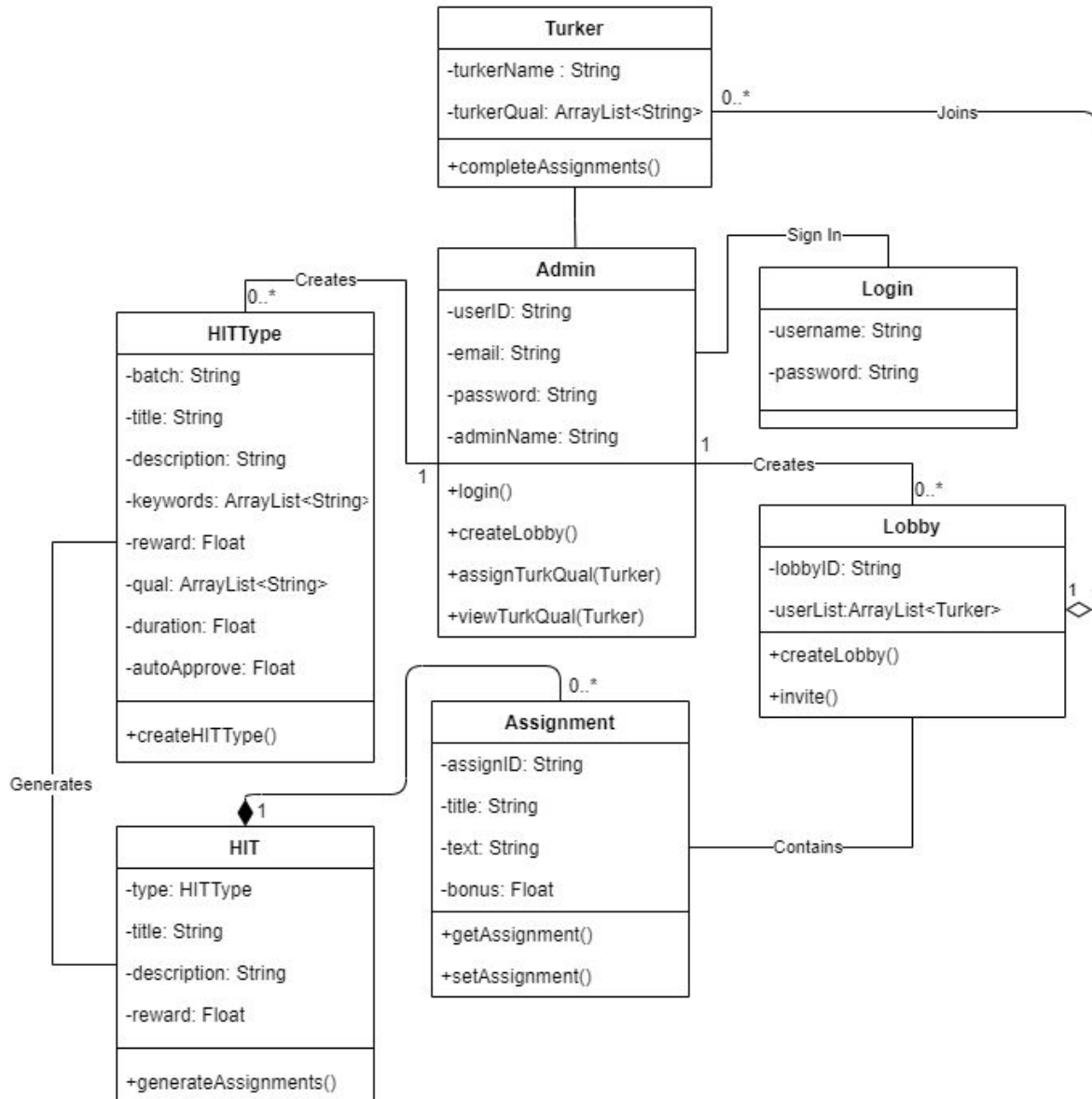
## High-Level Architecture

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



## Major Data Elements

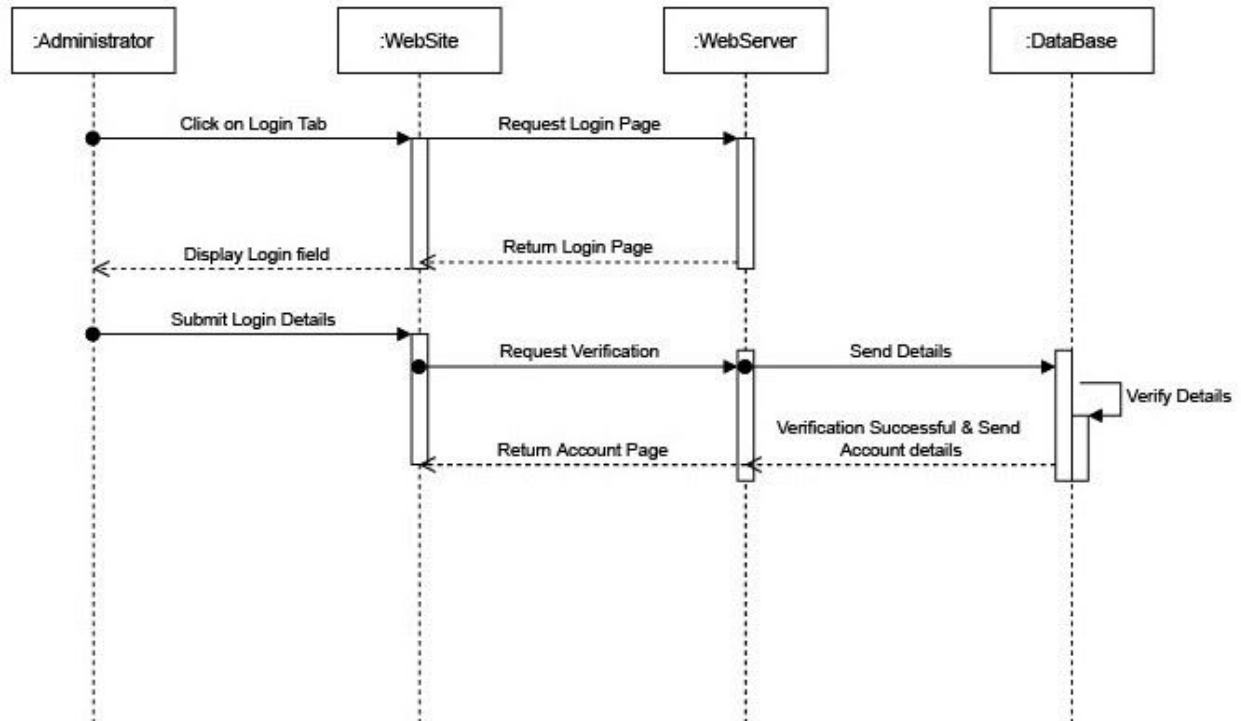
The UML class diagram lays out all the major data elements and their relationships that we expect to include in our framework.



# Interaction Scenarios

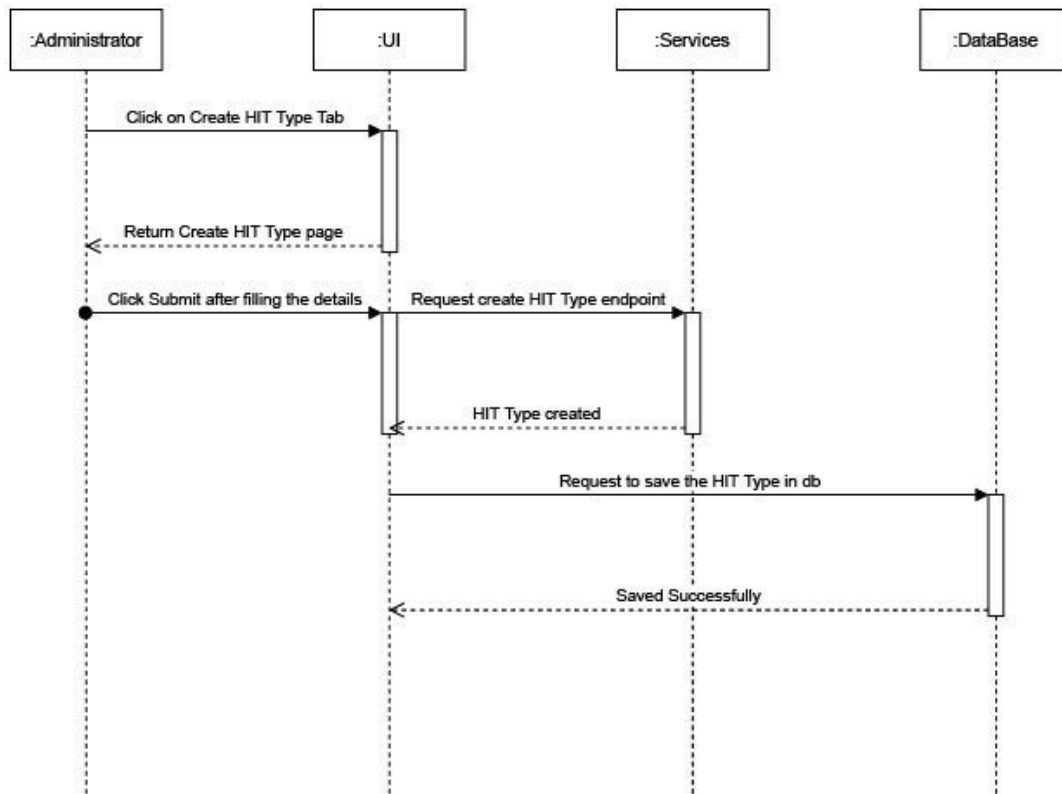
## Login:

- Administrator: User of our application.
- WebSite: The Webpage that the user can view.
- WebServer: the server on which our application is running
- Database: Database that has the login information about all the users of our application.



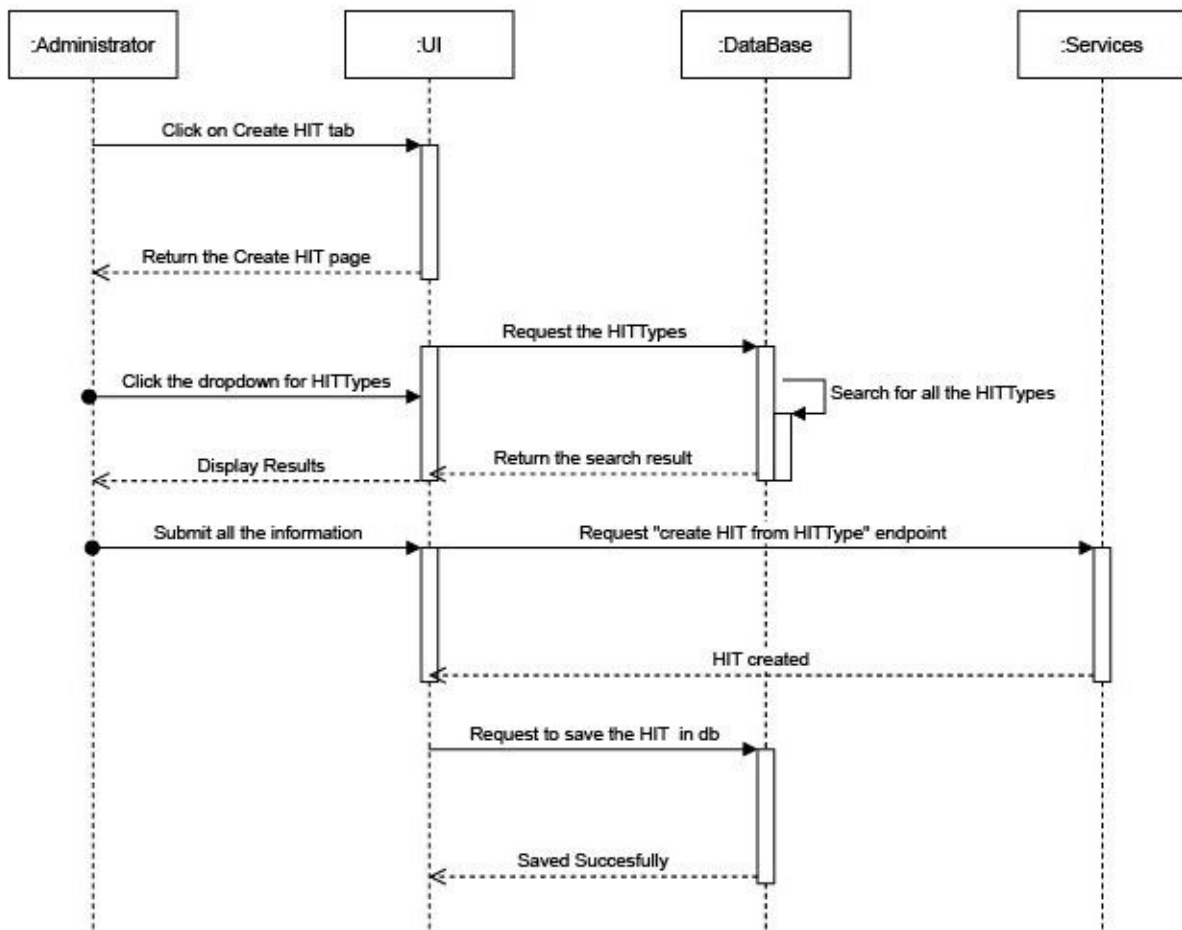
### Create HITType:

- Administrator: User of our application.
- UI: The Webpage & WebServer that the user can view.
- Services: Includes the Mturk API that is used to provide the create HITType endpoint
- Database: Database that stores all the details the administrator enters on the page about the HITType



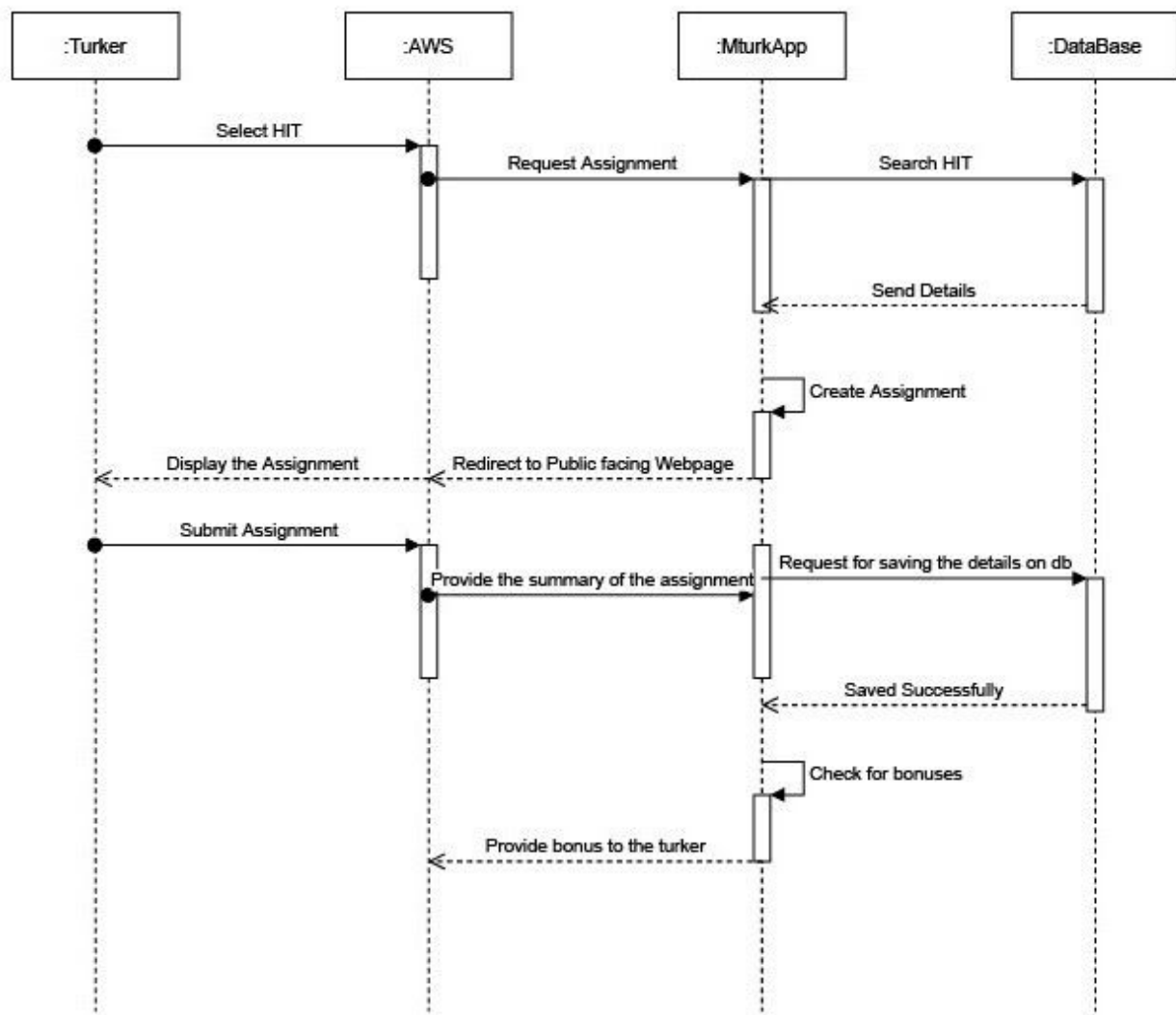
### Create HIT:

- Administrator: User of our application.
- UI: The Webpage & WebServer that the user can view.
- Services: Includes the Mturk API that is used to provide the create HIT from HITType endpoint.
- Database: Database that has the details about the HITType created by the User is used to provide details about it to create the HIT and then the HIT along with its details are saved in the database.



## Do HIT:

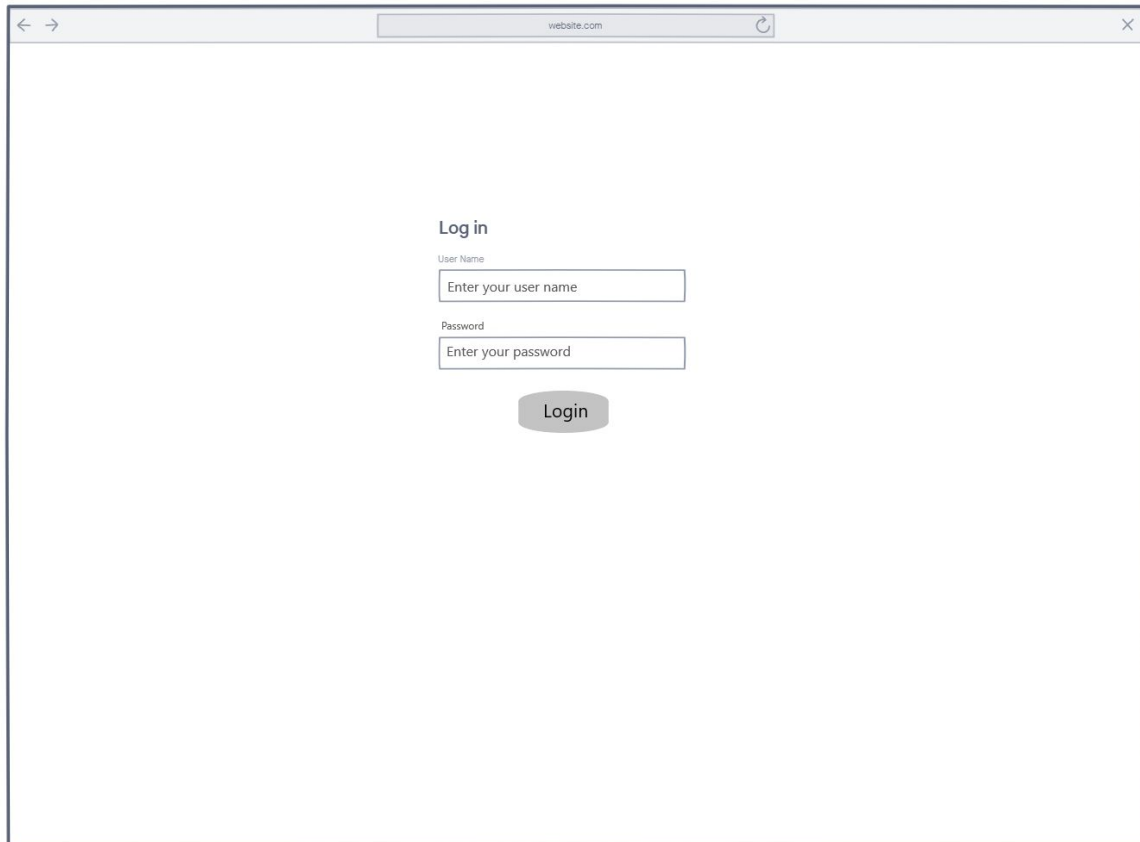
- Turker: User of AWS Services who will view our public facing web page to do the assignment.
- AWS: Amazon Mechanical Turk ( Amazon web services )
- MTurk App: Our application that would provide the redirection details about the public facing web page that will have the assignment related to the HIT. It will also calculate bonuses if there is any and provide the payment to the AWS account of the User
- Database: Database that stores all the details the HIT and assignment that the Turker did on the Public facing web page, so that the administrator can review the progress of the HITs.



# Low-Fidelity User Interface

**This interface is the basic design of the login page.**

This interface encapsulates the functionality described in User Stories: *MH1*



A low-fidelity wireframe of a login page. The page is enclosed in a rectangular frame representing a browser window. At the top, there is a header bar with a back arrow, a forward arrow, a search bar containing the text "website.com", and a close button (X). The main content area is white and contains a "Log in" section. This section includes a "User Name" label, a text input field with the placeholder text "Enter your user name:", a "Password" label, a text input field with the placeholder text "Enter your password:", and a "Login" button. The button is a rounded rectangle with a grey background and the text "Login" in white.

Log in

User Name

Enter your user name:

Password

Enter your password

Login

This interface is the basic design for the assignments page.

This interface encapsulates the functionality described in User Stories: *MH2, MH6, MH7, MH10, SH4, SH5, CH1, CH2, WL1*

The screenshot shows a web application interface for 'Maintenance'. The browser address bar shows 'website.com'. The page has a sidebar with 'Maintenance' selected, and a top navigation bar with 'Mturk' and 'Admin' links. The main content area is titled 'Maintenance' with a 'HELP' icon and a 'View system status' link. Below the title are three buttons: 'Refresh Assignment State', 'Approve All Assignments', and 'Pay All Bonuses'. A search section includes a checked 'Advanced Search' checkbox, a filter bar with 'Date', 'Amount', 'Paid', and '...' buttons, and input fields for 'From' and 'To' dates, 'Showing Amount', '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', and 'worker3'. Each worker entry shows their IP address, a list of instances, and a table of assignment details. 'worker1' and 'worker3' have a checked 'Instances' checkbox, while 'worker2' has an unchecked one. The assignment details for each worker include 'Accepted' and 'Submitted' dates, a 'Status' dropdown menu, and a 'Bonus' amount. A 'PAY SELECTED' button is located at the bottom right of the assignment list.

Maintenance [HELP](#) [View system status](#)

Refresh Assignment State Approve All Assignments Pay All Bonuses

☒ Advanced Search Date X Amount X Paid X ... X L

From  To  Showing Amount ☐ Is Paid ☐ Is Completed

Assignment Exit Data

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   
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   
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   
Bonus 2.33



**This interface is the basic design for the Lobby page.**

This interface encapsulates the functionality described in User Stories: *SH2*, *SH3*

← → website.com

+ Create New Lobby

**Current Lobby**

Behavioral Economics Pilot (Tur... ▾

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

0 users in current lobby

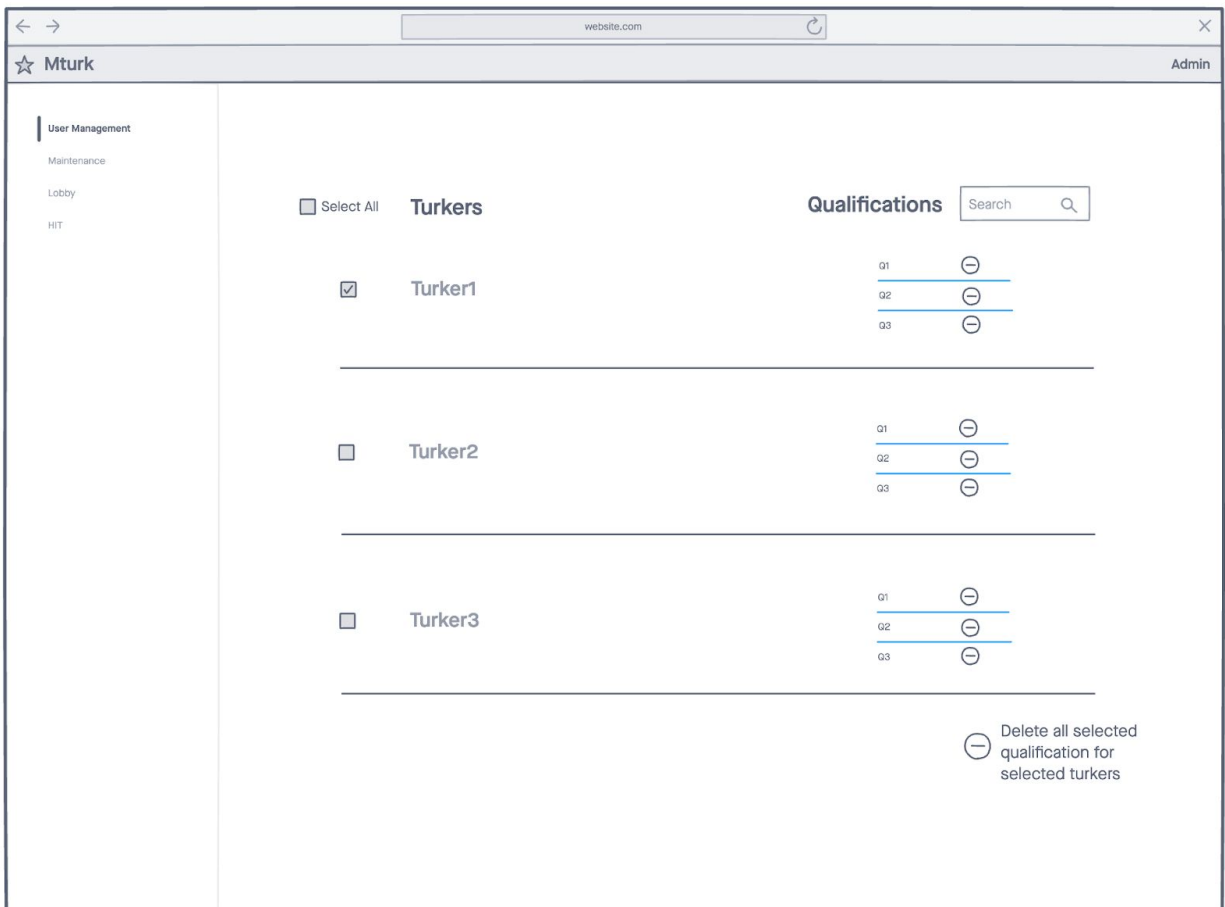
0 ready users are in current lobby

Trigger Lobby Event

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

**This interface is where you can manage qualifications for Turkers.**

This interface encapsulates the functionality described in User Stories: *MH8, MH9*



**This interface is where you can manage HITs/HITTypes.**

This interface encapsulates the functionality described in User Stories: *MH3*, *MH4*, *MH5*

