

1. Basic functionality

1.1 Register

User needs to **register** before starting the game. Username, email address and password are needed in registration. After registration, user will have the initial property(see 2.2 Model *MyProperty*), which represents the accumulated attributes. Additionally, user can **choose a play role** (several different play roles are available) as a default play role(see 2.3 Model *PlayRole*). After completing registration, user will be able to start the game anytime.

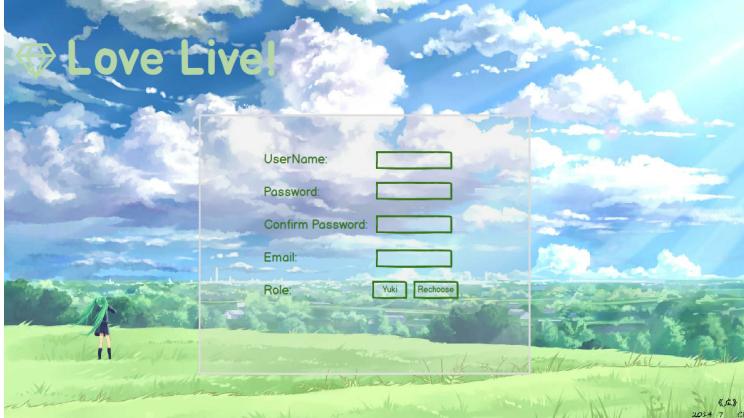
Homepage:



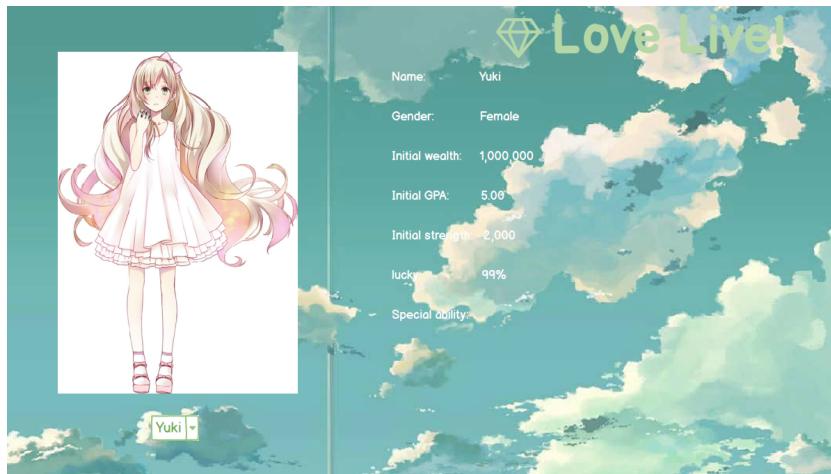
Login Page:



Registration Page:



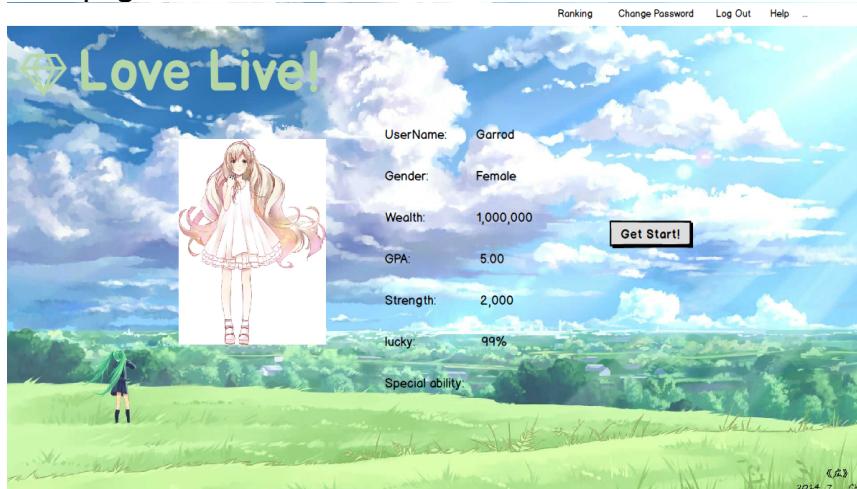
Choose Role Page:



1.1.1 Main page

After registration, user will be redirected to this page. In this page, user's properties will be displayed, such as accumulated wealth, accumulated GPA, accumulated strength, experience, level, ranking...etc. "Ranking" is also linked to a **score board** which demonstrates the first 50 users of highest level.

Main page:



Score Board:



1.2 Before game

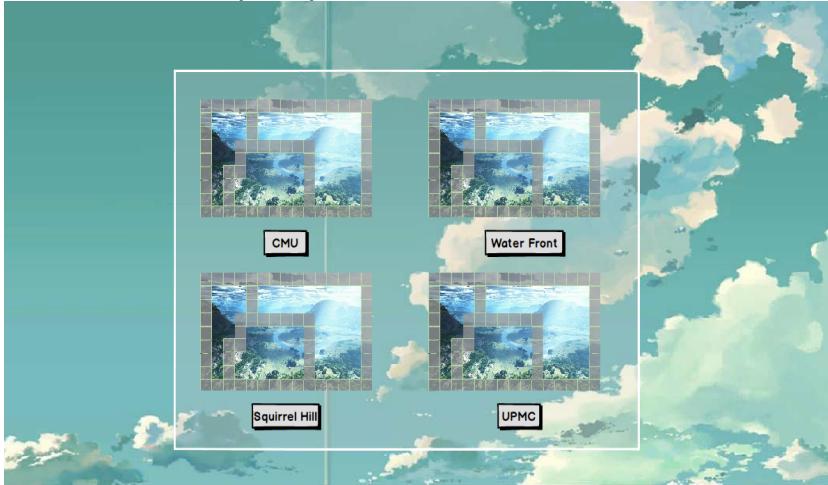
By clicking "**Get Start**", a new round is ready to begin.

1.2.1 Choose play role

User will be able to change his/her play role in this [Choose play role page](#). Note that each play role has different initial attributes (*see 2.3 Model PlayRole*). Some roles may be wealthier, some roles may have higher GPA or some roles are just luckier.

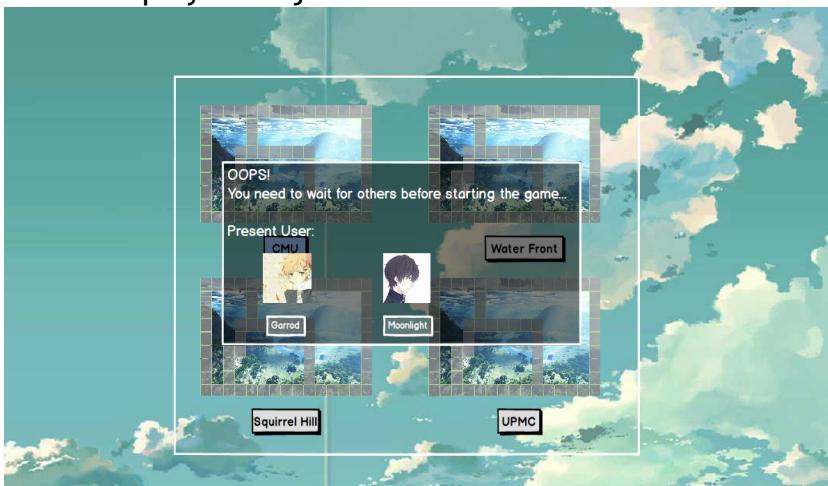
1.2.1 Choose map

After deciding the play role, user can [choose a map](#) he/she desires to play at. Different maps represent different themes (*see 2.4 Model PlayMap*).



1.2.2 Wait for other players

Each game requires exactly 4 players. Thus, if user chose a map that was chosen by less than 4 players, user needs to stay in the [waiting page](#) waiting for other players to join in.



1.3 Game started

The [game page](#) is shown below.



Initially, four players stand on four corners of the map square separately. Then players take turns to roll the dice. The number they rolled decides the number of steps they will be able to walk on the map. Each time they arrive at the new position, corresponded event (if exists) will be triggered (*see 3. Event*). Some events could possibly affect players' attributes, such as by deducting money, increasing GPA, gaining strength etc. If a play's money is less than zero, GPA is less than 3 or strength is less than zero, either of these situations will result in failure in this round of game. The game will continue until only one player left, who will be the winner. Additionally, players will be able to chat with each other during the game in the [chat panel](#) in right-hand side of page.

1.4 After game

After the game, a user's properties will be all updated, which will be show in the [result page](#).



The relative value of wealth in the beginning of the time and after game will be added to the user's wealth property. For example, the initial wealth of the role that player chose was 100\$. During the game, the player lost all the money. Thus, the relative wealth will be -100\$, which will be added to a user's accumulated wealth (wealth property).

The new GPA will be updated by calculating the average value of the GPA before and after the game. For example, the initial GPA of the role the user chose is 4.0, during the game, that user successfully increased his/her GPA to 4.5. If user's accumulated GPA before the game is 3.5, now his/her GPA will be updated to 4.0, which is the average value of GPA before (3.5) and after (4.5) the game.

The strength will also be updated and the updating process is similar to that of GPA, in which the average strength is calculated.

The experience value will be updated according to the ranking in the game. The winner will rank first, the last loser will rank second...etc. The ranking decides the points will be added to the accumulated experience.

2. Models

2.1 Model User (Django authentication model)

Username

Email

Password

2.2 Model MyProperty

The model represents the accumulated properties of a user. Fields are listed below:

Owner

Current role

Accumulated Wealth (default=200\$)

Accumulated GPA (default=3.5)

Accumulated strength (default=20)

Accumulated Experience (default=0)

Level

2.3 Model PlayRole

The model represents the initial properties of a play role. Fields are listed below:

Role name

Initial wealth

Initial GPA

Initial strength

lucky

Special ability

2.4 Model PlayMap

The model represents the properties of a map. Fields are listed below:

Map name

Map picture

2.5 Model Event

The model represents the properties of an event. The map field links a certain map to corresponded events. The Message represents the message displayed on the webpage. The Delta fields represent the change made to a player's properties. Fields are listed below.

Map

Event name

Message
Delta wealth
Delta GPA
Delta strength

2.6 Model ChatMessage

The model represents the properties of a chat message. Fields are listed below.

Chat message text
User
pub_date

3. Event

3.1 Events

Event was triggered when a play arrived in a certain position. Events may vary according to different themes. For example, in CMU campus theme, there is an event called “preparing for the midterm exam”. If a player triggered this event, his/her strength will be decreased by 2. Each event has different delta value that will be added to the player’s attribute. Besides, some events will also affect the user’s behavior, such as making user stay in the same position for two rounds.

3.2 Fortune cards and Chance cards

Some positions have fortune cards or chance cards. Now the lucky value of each role could play a significant role when choosing a card. The role of higher lucky value will probably be able to choose cards that triggered better events.