Sprint 3 – Manual Testing

The purpose of this document is to detail the manual testing of the new flask server and UI of the Polybius app. This includes tests for multiple stories located in our current sprints planning document. In addition, any external tests will be mention at the end of this document.

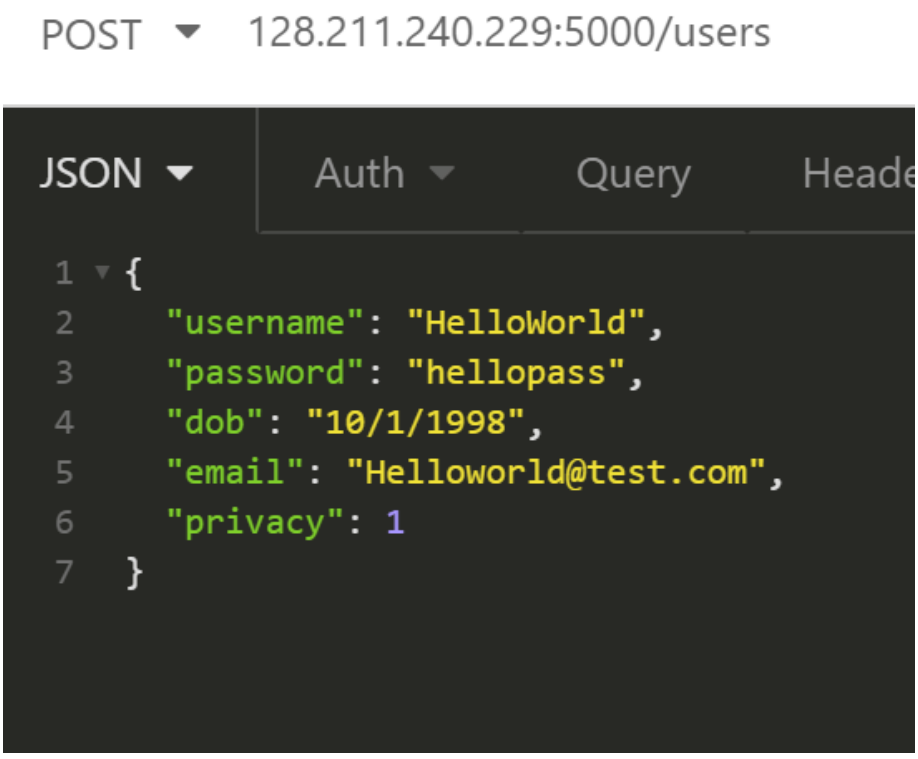
Test One: The Flask server should be able to receive requests for legacy functions

Procedure:

Steps followed to verify legacy functionality if working

- Run tests through insomnia for Users
- Run tests through insomnia for Friends
- Run tests through insomnia for Lobbies
- Run tests through insomnia for Messaging

1. Run tests through insomnia for Users is successful. The below images show that once the top has been sent to the flask server the database can correctly use the data and return a success. This works for the other user commands and has been removed for redundancy.



2. Run tests through insomnia for Friends is successful. When someone adds a new user as a friend the server can correctly parse the information and send back that the are now friends. Pictures below to show this.

3. Run tests through insomnia for Lobbies is successful. When a new lobby is created it is successfully added to the server. When accessing the current lobbies the server correctly displays any that have been added. Pictures below for reference.

GET 128.211.240.229:5000/lobbies?gameType=pong


```
[
  {
    "gameType": "pong",
    "latCoord": 40.426971435546875,
    "lobbyID": 1,
    "longCoord": -86.91646575927734,
    "name": "GGG"
  },
  {
    "gameType": "pong",
    "latCoord": 10.0,
    "lobbyID": 2,
    "longCoord": 10.0,
    "name": "HelloGame!"
  }
]
```


4. Run tests through insomnia for Messaging is successful. The messages are sent and stored in the database and when searched for can be seen by the recipient. Pictures below for reference.

POST 128.211.240.229:5000/messages

JSON Auth Query Header 1


```
{
  "senderID": 3,
  "receiverID":2,
  "message": "Test Hello"
}
```



```
{
  "msg": [
    {
      "message": "Test Hello",
      "messageID": 2,
      "receiverID": 2,
      "senderID": 3,
      "time": "2018-11-28 18:16:26.424471"
    }
  ]
}
```

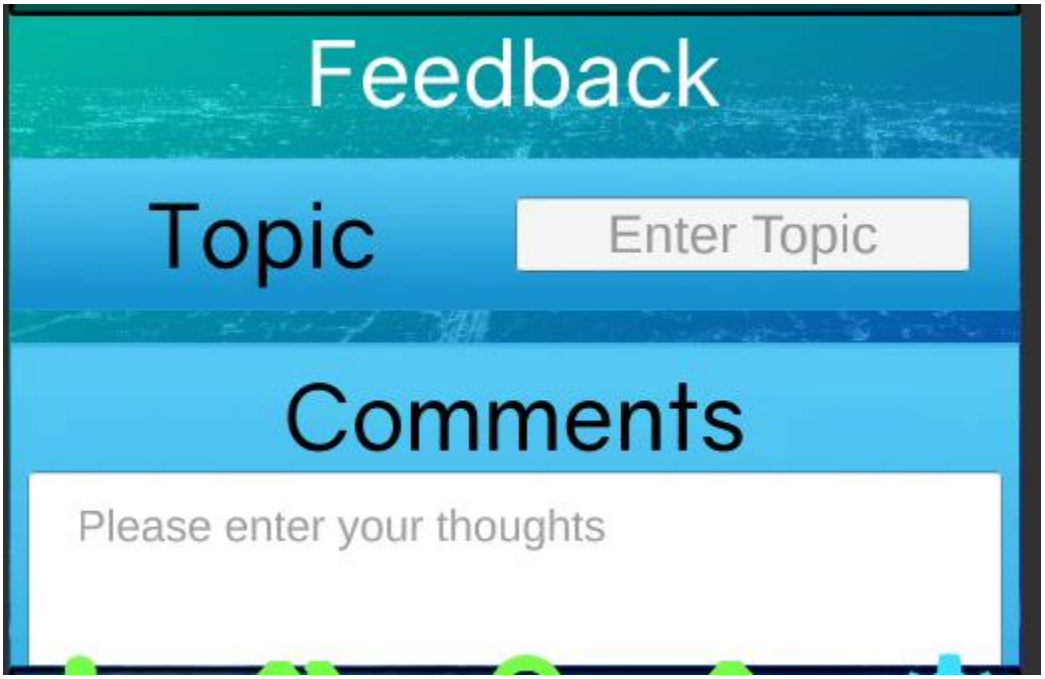

Test Two: A user can successfully submit feedback about the app

Procedure:

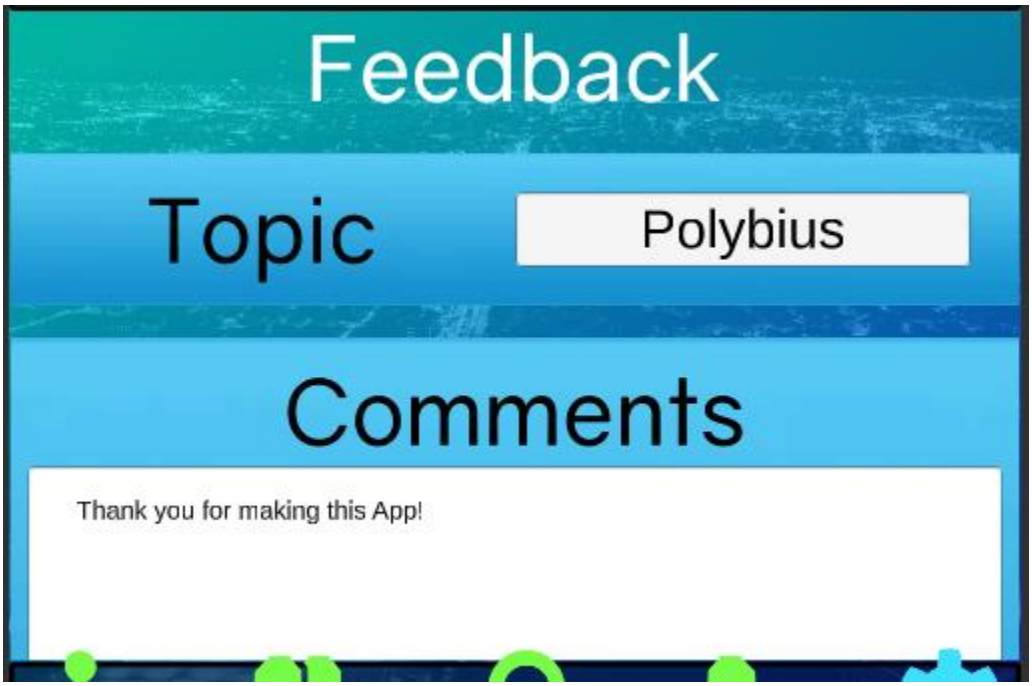
Steps followed to verify feedback was sent

- Navigate to feedback
- Send feedback through app
- Verify it was sent through insomnia

1. Navigate to feedback is successful. The user can click on the send feedback button and is brought to a feedback page. Picture for reference.



2. Send feedback through the app is successful. A user can enter their information and submit their feedback using the send button. Pictures for reference.



3. Verify Feedback is sent through insomnia is successful. Insomnia displays the feedback that the user has sent and we can view it easily through insomnia. Picture for reference.


```
{
  "feedback": [
    {
      "feedback": "Topic: Polybius",
      "id": 1,
      "subject": "Thank you for making this App!"
    }
  ]
}
```

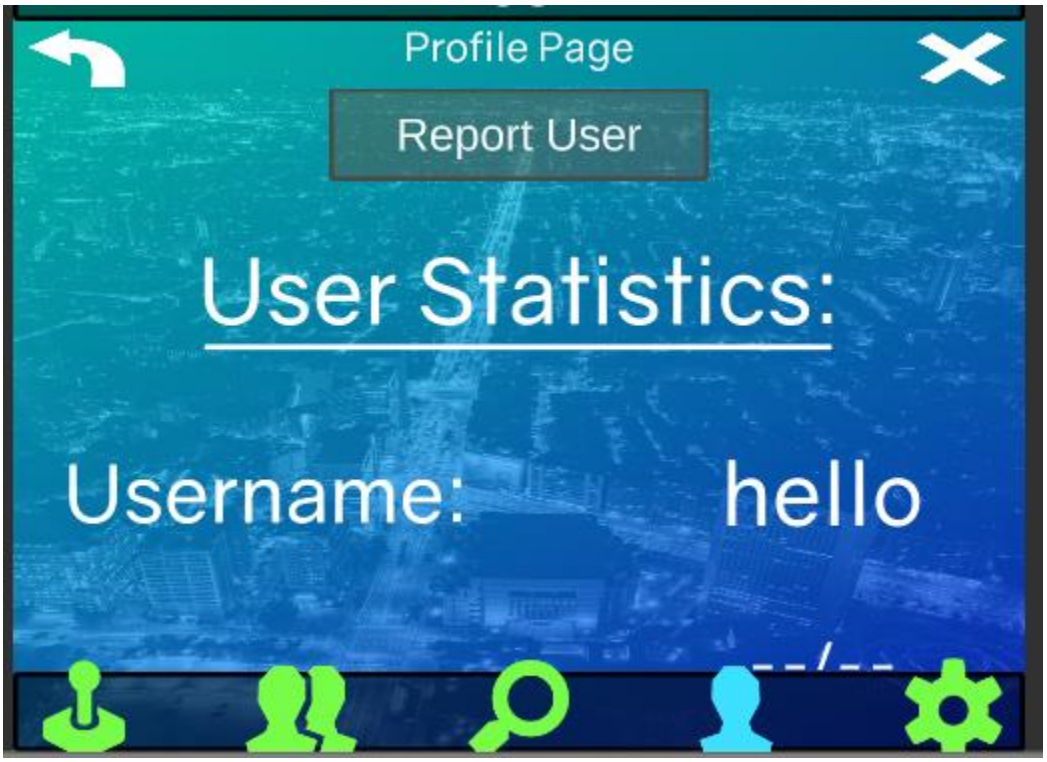

Test Three: A user can successfully block a user

Procedure:

Steps followed to verify blocking works

- The user can press block on a user
- User cannot see the user in the player list

1. The user can press block on a separate user. The user can view and press the block button on another profile. Picture for reference.



2. User cannot see the user in the player list is succesful. The blocked user does not appear in the list of users. Picture for reference.

Test Three: A user can successfully report a user

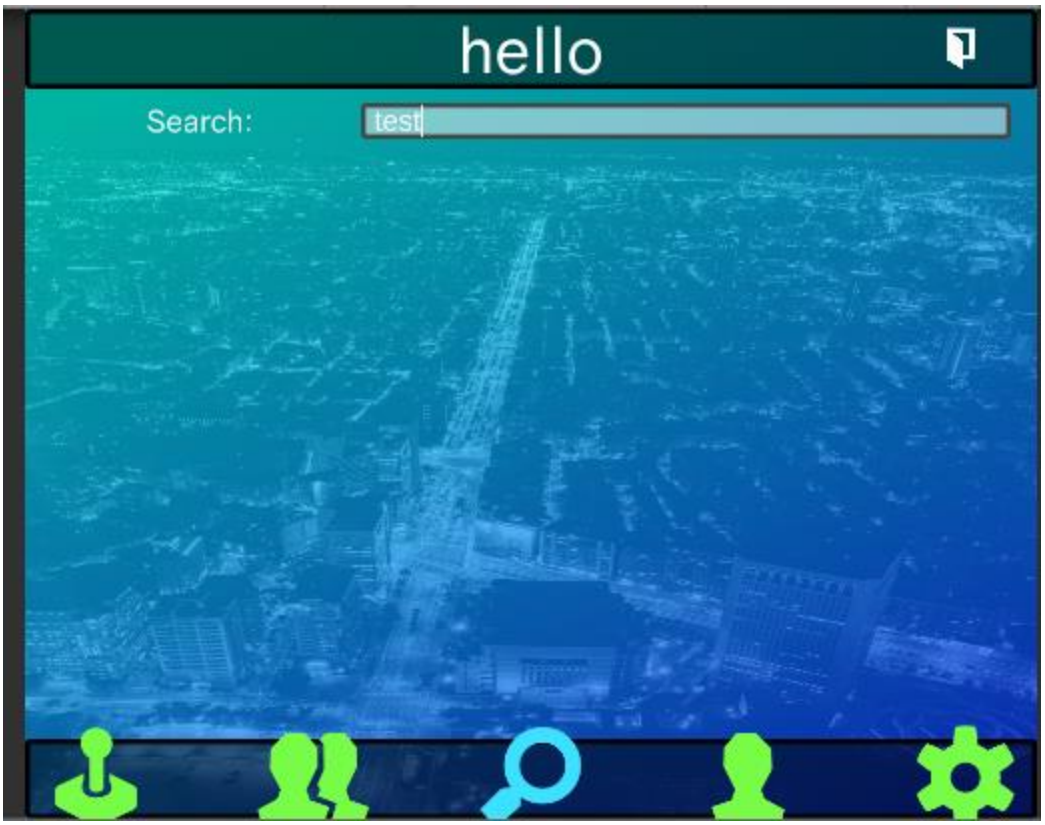
Procedure:

Steps followed to verify reporting works

- The user can press the report button
- The user can report another user
- A user will be blocked after a certain amount of reports

1. The user can navigate to the report page is successful. Once a user has been selected, they can navigate to their report button. Pictures for reference.

2. The user can report another user is successful. The selected user can be reported. Picture removed for redundancy.
3. A user will be blocked after a certain amount of reports is successful. Once a user has been reported enough, they are removed from the user search. Picture for reference.



Test Four: