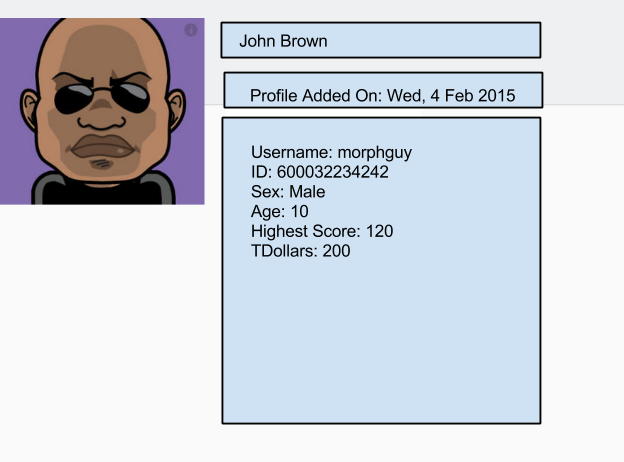
Project 1 and 2 of 4 - Dynamic profile thing and Profile API

At the end of this project you will have a Flask based application that can accept and display profile information. The profile information will be stored in a POSTGRESQL database.

Each profile will look like the screenshot below:



Specifications

**Part 1**

Create a Flask App that  accepts input for a user profile. The form will contain the following:

* file field called "image" which accepts the profile image
* text fields "firstname", "lastname", "age"
* select (option) fields "sex",

Note: High score and TDollars would not need to be in the submission form

Upon submission the form will  validate entries, to prevent bad data and **generate a userid**.

All of this input will be stored in the Postgresql database.

**Routes:**

Name the route for adding a profile  "**/profile**"

Name the route for viewing a list of profiles  "**/profiles**"

Name the route for viewing a profile "**/profile/<userid>**"

**Part 2**

* The API will use the following routes:  
  "**/profiles" GET  
  "/profile/<userid>" GET   
  "/profile/<userid>" POST**
* Each of these routes will return JSON ONLY when a header of *'Content-Type: application/json'* sent with the request
* It should be possible to retrieve a list of all the users in the system by sending a POST request to **/profiles**.  
  The result will be in json format:

|  |
| --- |
| { "users":[          {"username":"morphyguy","userid":"63453363"},          {"username":"morphyguy","userid":"63453363"},...        ]  } |

* It should be possible to retrieve a user's profile by sending a POST request to **/profile/userid** (DO NOT RETURN the first name and last name in the output).  
  The result will be in json format

|  |
| --- |
| {   "userid":"63453363",       "username":"",       "image":"...",       "sex":"",       "age":"",       "profile\_add\_on":"",       "high\_score":"",       "tdollars":""  } |