Hasura
161, 19th Main Road
6th Block Koramangala
Bangalore 560095

HPDF: Individual task for Week-1

Deadline: 09th December, 2017 (Extensions shall be provided in case of exams)

The first task as part of HPDF will test your individual understanding of the basics of the frameworks/libraries assigned to you.

You are expected to follow the instructions only for the language/framework assigned to you and run these tasks on your local machine. More instructions on deploying your tasks will be shared around the deadline. Please note that no integration between frontend/backend is required and you are only expected to use any mock data, etc. wherever necessary. [IMP] JOIN SLACK- https://hasura.slack.com/signup (registered email only!)

Frontend

React-Native

Use NativeBase (see http://docs.nativebase.io/docs/GetStarted.html for installation) to design the following screens (*see note below for screenshots***) from the Twitter Android app (***with no backend integration***):**

1) **Twitter timeline**
2) **Twitter profile menu**
3) **Twitter search box**

Note:

- **Screenshots of the above screens can be found here: https://drive.google.com/drive/folders/1tYQOVjJSSduX-0wVkj4nbAt1iyrE9rmf**
- **NativeBase is part of the recommended tutorial here: http://www.reactnativeexpress.com/component_libraries**

ReactJS

Create a twitter account to use as reference and use http://www.material-ui.com to create a new ReactJS app with only the following pages:

1) **Main timeline: https://twitter.com**
2) **Search results: https://twitter.com/search?q=aadhaar**

➔ **Please hard-code any data you need and implement basic scrolling, etc. wherever it is needed. It doesn't have to be a live version of the website. :)**

Backend

Both Python-Flask and NodeJS-Express

The following tasks will have to be demonstrated:

1) A simple hello-world at `http://localhost:8080/` that displays a simple string like "Hello World - Arpit"; replace "Arpit" with your own first name).

2) Add a route, for e.g. `http://localhost:8080/authors`, which:
    a) fetches a list of authors from a request to https://jsonplaceholder.typicode.com/users
    b) fetches a list of posts from a request to https://jsonplaceholder.typicode.com/posts
    c) Respond with only a list of authors and the count of their posts (a newline for each author).

3) Set a simple cookie (*if it has not already been set*) at `http://localhost:8080/setcookie` with the following values: `name=<your-first-name>` and `age=<your-age>`.

4) Fetch the set cookie with `http://localhost:8080/getcookies` and display the stored key-values in it.

5) Deny requests to your `http://localhost:8080/robots.txt` page. (or you can use the response at `http://httpbin.org/deny` if needed)

6) Render an HTML page at `http://localhost:8080/html` or an image at `http://localhost:8080/image`.

7) A text box at `http://localhost:8080/input` which sends the data as `POST` to any endpoint of your choice. This endpoint should log the received the received to `stdout`.

Please note that `http://localhost:8080/` is just an example, you can run the flask, express web-servers at their default ports on your local machine. You will receive more instructions on deploying your app after the deadline.