

# Software Engineer - FullStack Intern

<https://go.golinks.com/fullstack-intern-hackday>

**Task:** Given a GitHub username as input, build an API endpoint that returns the aggregated statistics across all of the user's repositories. Then create a user interface that displays the aggregated statistics.



## Basic requirements

- Backend - Build an API endpoint that takes a username and returns their aggregated stats in JSON format.
  - Use GitHub's API to pull necessary data about a user.
  - The API request should specify a username as an input.
  - The API response should return these aggregated stats across all the user's repositories:
    - Total count of repositories
    - Total fork count for all repositories
    - A list of used languages with their counts, sorted by the most used to least used
  - The API response should be in JSON format.
- Frontend - Build a user interface that interacts with your API.
  - We highly recommend using React
  - Create an input that allows you to enter a username
  - Display the user's information from your endpoint in any format you like.

## Hint

- The default API will give you only 30 repository results, but there could be more.

## References

- <https://api.github.com/users/seantomburke> - We will be testing with seantomburke's user account.

**Bonus Ideas - the goal is to wow the panel, here are some suggestions, but feel free to be creative**

- Add more Github stats
  - Ex: total stargazers for all repositories or the average size of a repository in the appropriate KB, MB or GB
- Add sorting, filtering or other options to modify the API response
  - Ex: When a checkbox or query parameter "forked" is set to false, it will only show statistics for repositories that are not forked.
- Add in functionality to search for users

- Add in functionality to view all the user's repositories
- UI/UX improvements
- Add appropriate HTTP response codes and error messages
- Handling different frontend states
  - Ex: loading, empty, error
- Host the application on AWS, Heroku, Vercel, Github, or any web host of your choice.

## Instructions

The requirements are left vague on purpose to see your creative ability to architect a solution here. Use any language you are comfortable with; for the frontend, we highly recommend React. After you finish your project, sign up to present your project to our Engineering team. You will have 5 minutes for the presentation - 2 minutes to present, and 3 minutes for Q&A. You can format your presentation any way you want, but here is a template you can use: ●

**30 seconds:** Overview of the tech stack

- **30 seconds:** Key challenges faced
- **1 minute:** Live demo
- **3 minutes:** Questions

We have a hard stop at 5 minutes, so please keep the presentation brief to allow time for questions.

Please submit your project to a public repo on Github so the engineering team can review your code. Good luck and happy hacking!

## Our thoughts on AI:

As a company, we encourage the use of new tools. This includes Generative AI, however, for this project, we want to see your skills in producing a solution, not what an AI can produce. We want to understand your thought process behind your solution, and understand how you approach problems. **Do not use AI-generated code in your solution.** You can use it for non-code questions, however we would much prefer that you reach out to a mentor to discuss your approach or ask questions. The use of web resources such as Stack Overflow and Google can be used.