‹ Return to "Optimize Your GitHub Profile" in the classroom

Optimize Your GitHub Profile

REVIEW | HISTORY

REVIEWED BY
Nyah

Dear student,

This is a great profile so far. It shows a lot of time and effort was put into this work to achieve this. There are some very minor changes required to make your profile perfect. I believe resolving these issues will make your profile captivating for recruiters. You surely won't be able to resubmit your profile for review. However, consider updating your profile as recommended.

General

- **Account has at least three projects**
- **Account shows knowledge about how to make incremental commits**
- **The commit graph shows many green squares for the last two weeks (indicating that commits have been pushed regularly)**

Required

Your account has at least three projects. However, the commit graph doesn't show many green squares for the last two weeks meaning commits have not been pushed regularly. In order to validate this rubric please endeavor to commit almost everyday for the next 2 weeks or more and then resubmit.

Suggestions

Contributing to Open source projects is a great way to enhance your profile and build up your commit streak. Here is a link with a lot of resources to help you get started.

Leaning Notes

- https://guides.github.com/activities/hello-world/
- https://gist.github.com/hofmannsven/6814451
- https://opensource.guide/how-to-contribute/
- https://help.github.com/articles/merging-a-pull-request/

Personal Profile

- **GitHub username is professional**
- **Profile picture is a professional image of student**
- **Profile includes at least one up-to-date links for: 'URL' and/or 'Company' fields and/or ‘Contact Email’**
- **Profile includes current location**

Required

- Your GitHub name should be professional enough. You could consider using at least 2 of your names.
- Your profile should display a professional picture of you. On the current profile picture, it's nearly impossible to see your face. Consider using an image cropped around your head and shoulders.

Projects

- **Last commit made matches the Udacity Commit Message Style Guide or the student has indicated that they are following another style guide**

Required

- Consider adopting the Udacity commit message style in your commit message and ensure that at least your last commit matches the Udacity Commit Message Style Guide or you should clearly indicate the type of style guide you are following.
- Consider using an imperative tone to describe what a commit does, rather than what it did. For example, use `Change`; not `Changed` or `Changes`.

- **Projects have meaningful names**
- **Projects have meaningful descriptions**
- **Most recent three projects have a completed README**

Awesome job, your Readme files are very elaborate. 👍

Suggestions and comments

For every project that necessitates execution, you should clearly state the instructions and commands needed to set up and launch the project.

Details about the projects aim/problem statement, your approach and implementation towards a solution, instructions to install and use, use-cases/screenshots, critical analysis of your results, instructions on how to contribute, listing of the various skills/new libraries/concepts learned and demonstrated over the projects are some of the areas you could consider looking into.

General Notes About READMEs

A README is very important to explain more details about the project which could not be covered in the project's name or description.

For example, the project motivation, the designing, a link to a working prototype, installation instructions, contributor's details are all sections that could be added to the README.

A completed README gives the viewer a holistic view of the project without having to go into the code.

Not all forked repos/changes require modifications to the README, especially when contributing upstream to another's project (would need to be a substantial change). All changes should be documented with commit messages to explain the changes.

Know that technical recruiters may or may not be programmers. Many times, they may not be able to understand code or run code. Even if they do, they may not want to invest time in seeing what your project looks like. READMEs are a way to immediately tell the recruiter how you can produce a great product or project. A poor README is a poor reflection on the effort you'd put into a work project.

Learning Notes

- Markdown cheatsheet
- How to write a great readme
- Write a README
- Udacity's course on writing READMEs

RETURN TO PATH