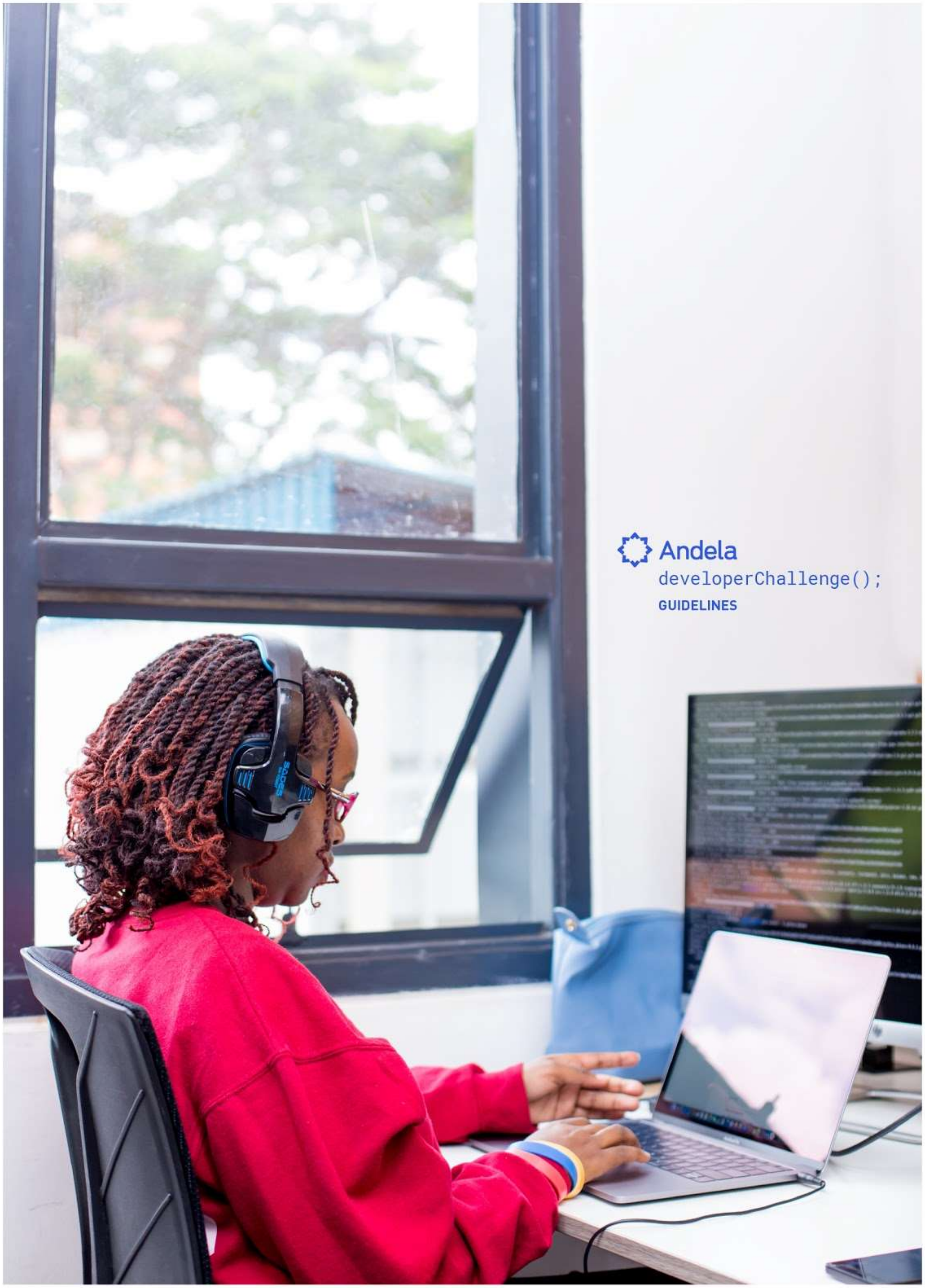

Andela

developerChallenge();

GUIDELINES

Andela Developer Challenge

Build A Product: **SendIT**

BUILD A PRODUCT: SendIT

Project Overview

SendIT is a courier service that helps users deliver parcels to different destinations. SendIT provides courier quotes based on weight categories.

Required Features

1. Users can create an account and log in.
2. Users can create a parcel delivery order.
3. Users can change the destination of a parcel delivery order.
4. Users can cancel a parcel delivery order.
5. Users can see the details of a delivery order.
6. Admin can change the **status** and **present location** of a parcel delivery order.

Optional Features

1. The application should display a Google Map with Markers showing the **pickup location** and the **destination**.
2. The application should display a Google Map with a line connecting both Markers (**pickup location** and the **destination**).
3. The application should display a Google Map with computed travel distance and journey duration between the **pickup location** and the **destination**.
4. The user gets real-time email notification when Admin changes the **status** of their parcel.
5. The user gets real-time email notification when Admin changes the **present location** their parcel.

NB:

1. The user can only cancel or change the **destination** of a parcel delivery when the parcel's **status** is yet to be marked as **delivered**.
2. Only the user who created the parcel delivery order can cancel the order.

Preparation Guidelines

These are the steps you ought to take to get ready to start building the project

Steps

1. Create a **Pivotal Tracker Board**

2. Create a **Github Repository, add a README, and clone it to your computer**

   Tip*: find how to create a Github Repository here.*

Challenge 1 - Create UI Templates

Timelines

- **Expected Length to Complete: 1 week**
- **Due Date: 18/11/2018 12:00AM**

Challenge Summary

You are required to create UI templates with **HTML**, **CSS**, and **Javascript**.

NB:

- *You are not implementing the core functionality yet, you are only building the User Interface elements, pages, and views!*
- *You are to create a pull request to elicit review and feedback for the UI templates when you are done working on them*
- *Do not use any CSS frameworks e.g Bootstrap, Materialize, sass/scss.*
- *Do not download or use an already built website template.*

Guidelines

1. **On Pivotal Tracker, create user stories to set up the User Interface elements:**
    a. User sign-up and sign-in pages.
    b. A page/pages where a user can do the following:
        i. Create a parcel delivery order.
        ii. Change the destination of a parcel delivery order.
        iii. See the details of a parcel delivery order such as the pickup location, destination, and price.
        iv. Cancel a parcel delivery order.
        v. View all parcel delivery order the individual user has created
    c. A page/pages for a user's profile which, at minimum displays:
        i. The number of parcel delivery order that has been delivered.
        ii. The number of parcel delivery orders that are yet to be delivered (in transit).
        iii. List of all parcel delivery orders.
    d. A page/pages where an Admin can do the following:
        i. Change the **status** of a parcel delivery order.
        ii. Change the **present location** of a parcel delivery order.

2. **On Pivotal Tracker, create stories to capture any other tasks not captured above. A task can be feature, bug or chore for this challenge.**
3. **On a feature branch, create a directory called UI in your local Git repo and build out all the necessary pages specified above and UI elements that will allow the application function into the UI directory**
4. **Host your UI templates on GitHub Pages.**

   Tip: *It is recommended that you create a* ***gh-pages*** *branch off the branch containing your UI template. When following the GitHub Pages guide, select* ***"Project site" >> "Start from scratch"****. Remember to choose the* ***gh-pages*** *branch as the* ***source*** *when configuring Repository Settings.*

Target skills

After completing this challenge, you should have learned and be able to demonstrate the following skills.

Skill	Description	Helpful Links
Project management	Using a project management tool (Pivotal Tracker) to manage your progress while working on tasks.	● To get started with Pivotal Tracker, use Pivotal Tracker quick start. ● Here is a sample template for creating Pivotal Tracker user stories.
Version control with GIT	Using GIT to manage and track changes in your project.	● Use the recommended Git Workflow, Commit Message and Pull Request (PR) standards.
Front-End Development	Using HTML and CSS to create user interfaces.	● See this tutorial ● See this tutorial also

Self / Peer Assessment Guidelines

Use this as general guidelines to assess the quality of your work. Peers, mentors, and facilitators should use this to give **feedback** on areas that should be improved on.

Criterion	Does not Meet Expectation	Meets Expectations	Exceed Expectations
Project management	Fails to break down modules into smaller, manageable	Breaks down each module into smaller	Accurately, assigns points to the tasks.

	tasks. Cannot tell the difference between chores, bugs, and features	tasks and classifies them. Constantly updates the tool with progress or lack of it	Informs stakeholders of project progress/blockers in a timely manner
Version Control with Git	Does not utilize branching but commits to master branch directly instead.	Utilizes branching, pull-requests, and merges to the develop branch. Use of recommended commit messages.	Adheres recommended GIT workflow and uses badges.
Front-End Development	Fails to develop specified HTML/CSS web pages or uses an already built out website template, or output fails to observe valid HTML document structure	Successfully develops HTML/CSS web pages while observing standards such as doctype declaration, proper document structure, no inline CSS in HTML elements, and HTML document has consistent markup	Writes modular CSS that can be reused through markup selectors such as class, id. Understands the concepts and can confidently rearrange divs on request.