Group Members: Kohiin Desravines, Mohamed Hassan, Levy Andrew, Franklin Ordonez

Manager: Kohiin
Presenter: Mohamed
Documentation: Franklin / Levy

What is the application?

Our application is **SeeFood!** With SeeFood ALWAYS KNOW what you are eating. Eat with your Eyes! This application allows you to take a picture of any food item or dish and instantly identify it. The app will also provide nutrition facts about the item/dish as well as a list of potential recipes. Come across a cold noodle dish? Confused!? No worries with SeeFood!

Process Documentation:

1. Deciding Monolithic vs Microservice
    a. We threw around some app ideas and decided that we should definitely use a microservice architecture. We hope to grow our app and a microservice architecture will allow for suitable scaling in the future.

2. Deciding on the App Idea
    a. We decided on a general food app that tells you what you are about to eat.

3. Figuring out Github Setting
    a. We then paused on ideas because we wanted to figure out how we would all connect to the same repo securely. We discussed how we could make 4 ssh key pairs and deploy 4 public keys in github.com. This way, we would each keep our private keys private.

4. Our manager created a new Repo in github, called GroupFoodApp.
    a. Next we needed to make a branch of our main, we called it ideas.
    b. git@github.com:SrKoDes/GroupFoodApp
    c. git@github.com:SrKoDes/GroupFoodApp/tree/ideas

5. Setting up Local Project Folder with Github Remote Repo
    a. We each created a local project folder, called FoodApp
    b. We all ran git init in the project folder

c. We all need to connect the correct origin url with the command git remote add origin git@github.com:srkodes/groupfoodapp
d. Check config with cat .git/config

6. Setting up SSH with Remote Repo
    a. We each had to copy our previously made ssh public keys
    b. Went into ssh folder that is in our home directory and ran the command cat id_rsa.pub (so we can copy it)
    c. Prerequisite: in order to use a previously made key, you will need to disconnect it from other repos (unless the public key is assigned to your entire github profile)
    d. We all shared our public keys with the Manager so he can deploy them to the remote repo

7. Make Idea Files in Project folder and Check Github Connection
    a. Touch name_idea.txt
    b. WE MUST GIT PULL FIRST EVERY TIME WE WANT TO MAKE A LOCAL CHANGE AND SEND IT TO THE REMOTE REPO → git pull
    c. Make sure local remote is synced/switched to correct branch of remote repo → git checkout ideas
    d. Add to staging → git add .
    e. Commit to local repo → git commit -m “message here”
    f. Push to remote repo → git push

8. Adding feature to you text file in the repo
    a. Edit your idea text with a text editor → Nano name_idea.txt
    b. Type in your feature for the app
    c. Save → clt + o
    d. To exit the text editor → clt + x
    e. Sync with github remote repo → git pull
    f. Make sure local remote is synced/switched to correct branch of remote repo → git checkout ideas
    g. Add to staging → git add .
    h. Commit to local repo → git commit -m “message here”
    i. Push to remote repo → git push

Why did your group choose the architecture for this application?

We chose a Microservice Architecture because then our services could be dynamically scaled. When a photo is taken of a food, not every service needs to run. For example if you don't have gps enabled, you don't need to run an instance of that service. This will help us save money and computing power!

Features Section

Kohiin_Ideas.txt:

SeeFood, not to be confused with Seafood. An app that recognizes food based on its appearance.

Recipe premium microservice.

GPS for increased recognition capabilities as well as geotagging/recipe discovery.

Data mining service to steal our user's data.

Levy_Ideas.txt:

A calculator to calculate how many calories you should eat to lose or keep your weight.

Mohamed_Ideas.txt:

Nutrition Facts/ a display showing the nutrition facts of the specific food that was scanned.

Franklin_Ideas.txt:

Payment/Billing Feature. This app will offer a premium subscription option that leverages its advanced technologies and tailors them to the individual. Each user would need a profile and health details. The inclusion of a subscription option requires us to develop a billing/payment service and a user profile service.