CS321-P4-Abreu-Calciu-Hayes-Tang

Change Log:

These required changes were identified recently, but expediently dispatched. The changes were easy to accommodate as they required minimal alterations that didn't impact interfaces between components, just changes in data returned to requests, and minimal changes to processing returned data.

- JsonKeyEnum.java:
  - This enumeration represents keys in the JSON data model returned by BarcodeLookup that we want to extract and use. It implements Clifton Labs's JsonKey interface.
- JsonProcessor.java:
  - This class processes the JSON returned by BarcodeLookup and produces a list of Product.java instances.
- ShowCamera.java:
  - Called in CameraActivity to display the camera when the CameraActivity class is created.
- ProductListAdapter.java:
  - Defines the behavior of the ListView used to display product listings in ProductActivity. Various layout files were created to support this.
- RecyclerViewAdapter.java:
  - Defines the behavior of the RecyclerView used to display keyword checkboxes in ProductActivity. A layout file was created as part of ProductActivity's layout to support this.

Technology Used

Android Studio: This is the main environment we will be using to develop our app.

AWS:

- Cloud9: Coding environment and implementation for the EC2
- CloudWatch: Log for requests to the server. Used for testing and diagnostics.
- EC2: Serverless host using lambda functions
- Rekognition API is a vital component to our project as it will handle generating keywords based on the picture a user will take.

BarcodeLookup API: Used to retrieve web storefront listings for items matching a set of keywords

Git (Bitbucket): We will be using git for our VCS with Bitbucket as our frontend

Android Device (Camera/Permissions): The app is being developed solely for android devices. As such, accessing the camera and checking permissions will need to be handled in the context of our users being on an Android device

Contributions

- Home Page: Morgan (100%)
- Camera: Mike (80%), Morgan (20%)
- JSON parser: Adrian (50%), Mike (50%)
- Web Search: Adrian (40%), Mike (40%), Morgan (20%)
- Product Screen: Morgan (100%
- Git Management: Adrian (100%)
- Integration: Adrian (100%)
- AWS: George (100%)
- Adrian managed P4 deliverable

Software Reuse

Camera:

This video playlist was referenced to write the CameraActivity and Show Camera classes.

- https://www.youtube.com/watch?v=_wZvds9CfuE&list=PLF0BIlN2vd8upfo6mcCSvkeluvzH0wUp

The following video was used to help ask for user permissions on camera startup.

- https://www.youtube.com/watch?v=SMrB97JuIoM

AWS:

The following AWS documentations were used to finish coding the AWS

- https://docs.aws.amazon.com/lambda/latest/dg/nodejs-prog-model-handler.html
- https://aws.amazon.com/lambda/faqs/
- https://docs.aws.amazon.com/AWSJavaScriptSDK/latest/AWS/Rekognition.html

Barcode Lookup:

The following documentations were used to further understand and implement the API

- https://www.barcodelookup.com/api-documentation
- https://account.barcodelookup.com/

Product Screen:

These three videos were used as a guide to make the product listing page. The code was adapted to work with our web search and rekognition results.

- https://www.youtube.com/watch?v=E6vE8fqQPTE
- https://www.youtube.com/watch?v=cKUxiqNB5y0
- https://www.youtube.com/watch?v=SLFrwl1hFcw

The following Stack Overflow thread was used as a guide to make the product listings clickable and send users to the product's website.

- https://stackoverflow.com/questions/35324023/android-making-a-listview-item-open-a-url-in-the-browser

The code found in this GitHub repo was adapted for our scrolling list of keyword checkboxes.

- https://github.com/mitchtabian/Recyclerview/

Web Search:

For parsing JSON returned by BarcodeLookup, we used Clifton Labs's json-simple project.

- https://cliftonlabs.github.io/json-simple/

Reuse was extremely useful to our project. If we had not reused software, the entire semester would have been spent implementing a JSON parser, that probably wouldn’t have worked. The entire project would have had to have been built from scratch, as software reuse was active at every level. Even the Android toolkit is reuse. This project would not have been useful without reuse.