Spring 2018, CS5551: Advanced Software Engineering

Department of Computer Science Electrical Engineering

University of Missouri-Kansas City

Smart Shopping

Cameron L'Ecuyer (Team Leader)
Ruthvic Punyamurtula
Navya Pillala
Sneha Mishra

App Demo: https://www.youtube.com/watch?v=wyObDwxgtMU

GitHub: https://github.com/camlecuyer/CS5551_Team_11_Project

ZenHub: https://github.com/camlecuyer/CS5551_Team_11_Project#boards?repos=118971460

Acknowledgement Statement:

This work has been completed under the guidance of Dr. Yugi Lee and the TAs (Rohith Nagulapati, Sidrah Junaid, Nageswara Nandigam) in CS5551 Advanced Software Engineering, University of Missouri - Kansas City), Spring 2018.

Project Deployment

Smart Shopping User Manual

By:
Cameron L'Ecuyer
Sneha Mishra
Ruthvic Punyamurtula
Navya Pillala

Table of contents:

1. Introduction
2. How to use the system with Screen dumps
    2.1. Installation
    2.2. App Usage
        2.2.1. Home page
        2.2.2. Shopping page
        2.2.3. Trends page
        2.2.4. Suggestions page - BMI and social media
        2.2.5. Augmented Reality page
    2.3. Un-Installation
3. Error recognition and handling
4. Bugs and Deficiencies

1. Introduction - Android App to help user enhance their shopping for furniture. This App allows user to do the following:

   1.1 Login via registered email or Twitter credentials.

   1.2 Shop using text or image analysis.

   1.3 View the trends in the field of furniture based on the collected tweets.

   1.4 Give recommendations to the user based on their calculated BMI.

   1.5 Allows User to share the recommendations and suggestions on social media such as Facebook, Instagram, WhatsApp etc.

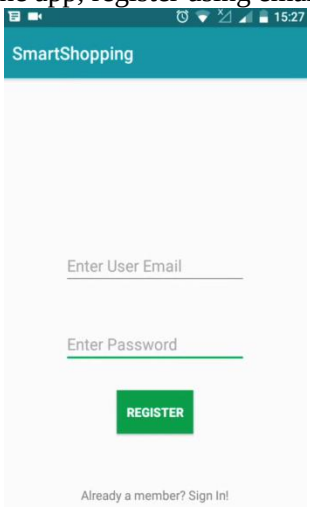
   1.6 View the product in AR (augmented reality), this helps the User to check-out the product in any desired location wherever they place the marker.

2. How to use the system with Screen dumps -

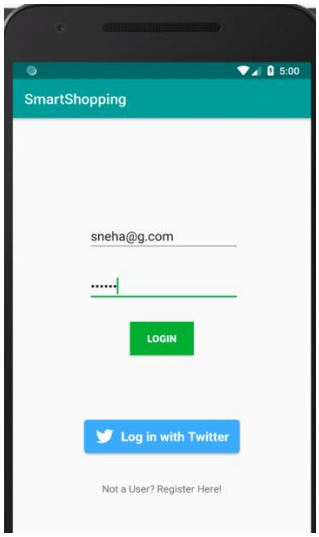
   2.1 Installation :

   2.1.1 Download the ‘Smart Shopping’ app.

   2.1.2 If new to the app, register using email ID or use an existing twitter credential.



   2.1.3 Log-in to the shopping app.



2.1.4 Download the Unity .apk file

2.2 App Usage :

2.2.1 Home page

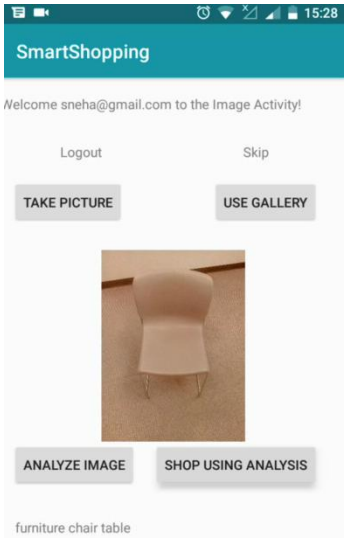
2.2.1.1 The default home page is the image analysis page, which the user can skip as per requirement.

2.2.1.2 If the User skips this page, they are redirected to the Shopping activity.

2.2.1.3 If the User is interested, they can click and image at run-time for image analysis.

2.2.1.4 Alternately, the user can select an existing image from the gallery for analysis.

2.2.1.5 The image analysis is done on the selected image and the description of the analysis is displayed to the User at the bottom of the screen.



2.2.2. Shopping page

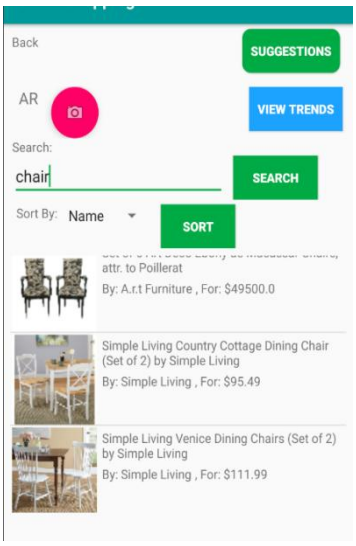
2.2.2.1. In shopping page, User can search for the product

2.2.2.2. User can also sort the products using the name or price

2.2.2.3. User can view the furniture trends by clicking on trends

2.2.2.4. The app also provides suggestions by giving user details as the input.

2.2.2.5. User can view the product by clicking on AR – Camera button in the shopping page



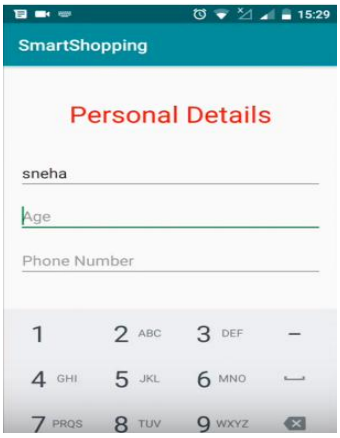
2.2.3. Trends page

2.2.3.1. User can view the shopping trends based on the furniture categories in this page

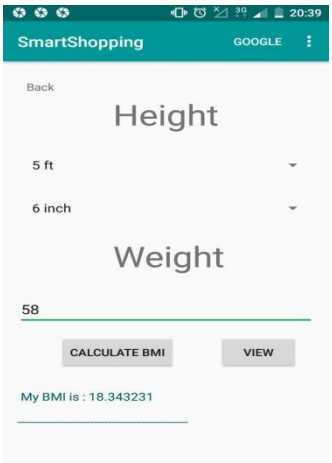
2.2.3.2. User can navigate back to the shopping page by clicking on the back button

2.2.4. Suggestions page - BMI and social media

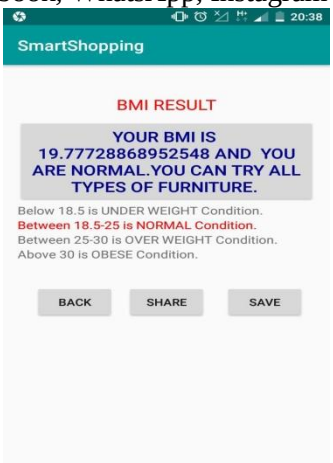
2.2.4.1. In the personal details, User has to enter name, age and phone number to get to the suggestions



2.2.4.2. Then user has the option to set user height in “Feet” and weight in “KG”, to find the BMI



2.2.4.3. User can save the result or share the recommendations to the social media like Facebook, WhatsApp, Instagram etc.



2.2.5. Augmented Reality page

User can view the product in Augmented Reality – 3D view anywhere by using the marker.

2.3. Un-Installation

The installed .apk file needs to be un-installed to remove the app from the device.

3. Error recognition and handling -

3.1 Once the User enters the AR mode to view any particular product, they cannot re-direct back to the shopping app. At this time the User can quit the app and re-open it to use the shopping app again.

3.2 The Image analysis returns the analysis report related to a fixed data set from the furniture field, if the image selected for analysis does not contain any furniture item, then the analysis report will be blank. In this case the User can modify or edit the search text as per their needs.

4. Bugs and Deficiencies -

4.1 App is not platform dependent, works only on android devices.

4.2 The password reset option is not provided to the user.

4.3 While registering user, we don’t confirm the password, so if the user needs to type carefully at the registration.

4.4 The trend analysis is based on the already collected twitter data and not done at run-time.

4.5 The trends when collected in real time slows down the app performance because of processing huge data

4.6 The shopping API returns only one image per product.

4.7 BMI part again requires user registration.

4.8 User cannot re-direct to the shopping app from the AR view.

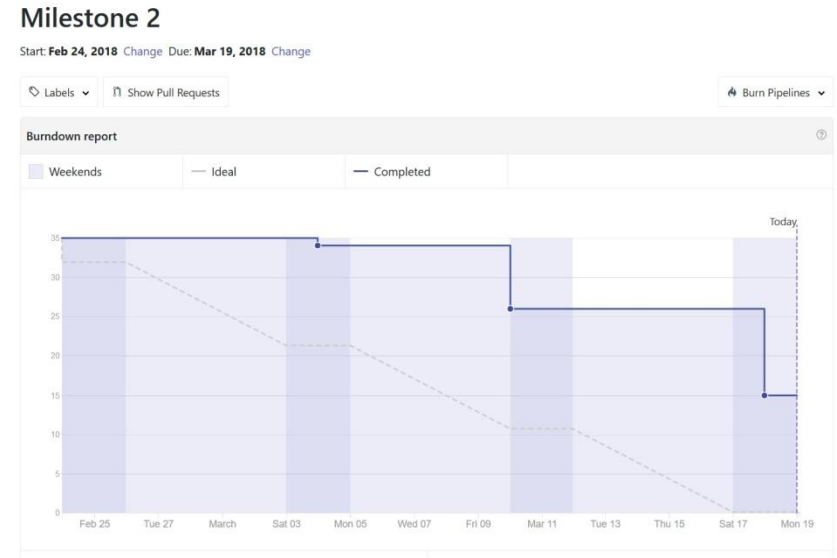
4.8 Log-out option is only provided at the home page of the app.

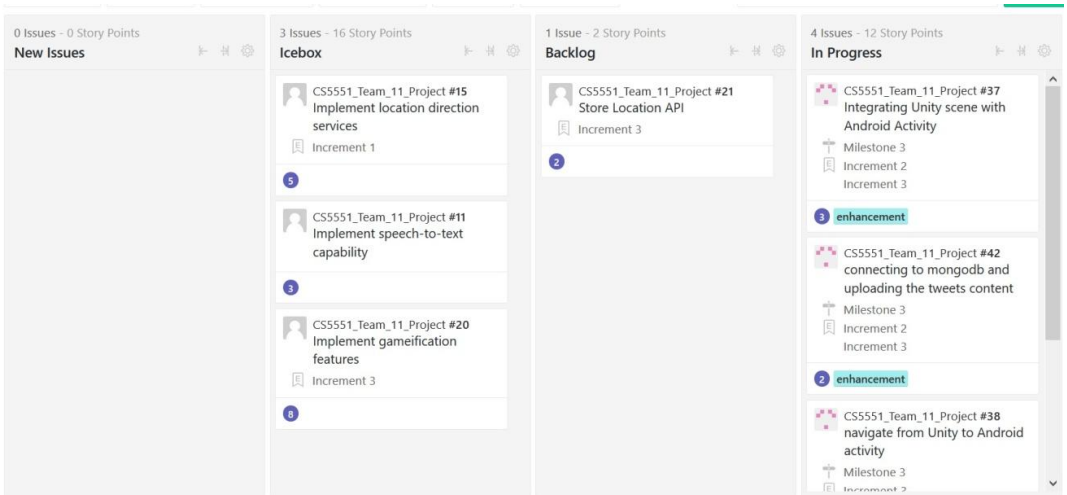
Project Management

Project management report:

Overall, the project management went alright, however, this was the first time I was acting as a project manager, and things did not go as smoothly as I had hoped. Part of the problem became that I was busy on other work and assigning tasks slipped my mind, at least at the start of the project. The next major issue was miscommunication that slowed the development process, and the team's overall inexperience working with GitHub as a team.

This is the burndown chart from the end of Increment 2, and the images below that are the boards from Increment 3, more information about the ZenHub files can be found in the increment reports





Remaining Issues	Story points
Integrate social media into search results CS5551_Team_11_Project #19 In Progress Milestone 3	5
Integrate Project CS5551_Team_11_Project #34 In Progress Milestone 3	5
Integrating Unity scene with Android Activity enhancement CS5551_Team_11_Project #37 In Progress Milestone 3	3
navigate from Unity to Android activity help wanted CS5551_Team_11_Project #38 In Progress Milestone 3	2
connecting to mongodb and uploading the tweets content enhancement CS5551_Team_11_Project #42 In Progress Milestone 3	2

Completed Issues	Story points
Increment 3 Epic CS5551_Team_11_Project #4 In Progress Milestone 3	Not estimated
Implement search by barcode CS5551_Team_11_Project #10 New Issues Milestone 3	2
Implement social media integration CS5551_Team_11_Project #18 In Progress Milestone 3	5
navigate from android activity to unity scene help wanted CS5551_Team_11_Project #39 New Issues Milestone 3	3

Team member's contributions:

Overall, everyone was in charge of designing, implementing, and testing their part of the app. During integration, Ruthvic, Cameron, and Sneha worked to ensure the app worked together without crashing.

- Ruthvic:
  - Ruthvic was in charge of the augmented reality part of the app, and some of the social media integration. He also assisted in developing diagrams, UML models, and mockups for the application during development. Near the end of the project he assisted Cameron in updating the UI of the project.
  - Overall Ruthvic worked hard on the project and did good work
  - **Points: 30**
- Sneha:
  - Sneha was tasked with creating the login and register pages, integrating Firebase, and dealing with image recognition.
  - Overall, Sneha did good work, although there was some confusion on how to properly use GitHub, as we once ended up with multiple versions of code for the same part all in one place.
  - **Points: 30**
- Navya:
  - Navya was charge with working on the BMI part, and the suggestions part for recommending products for the user.
  - Navya's part was the most unstable, as when we changed the target of the app in Increment 2, she had to shift her focus the most. She was also the one who responded to the group chat the least, and she got her Increment 3 report part in barely before the deadline.
  - **Points: 20**
- Cameron:
  - Cameron was in charge of getting the product search API working, writing various reports, bug fixing of the app, and leading the team. Near the end he worked with Ruthvic to update the UI of the project, and helped squash bugs from integration.
  - **Points: 20**

Final project evaluation:

- *How well your project satisfies your original requirement specifications?*
    - The project has the ability to search for information related to the topic of the project, however, our original idea was to do clothes with AR. However, due to the lack of resources available for clothing, we switched to furniture.
    - We can search for items, and have the ability to recognize images to find object, but they do not talk with each other perfectly. There is also no way for the main app to communicate with the AR part of the app, and the suggestions, and social media part do not talk to the shopping part.
    - Overall, I would say the project satisfies about 80% of the original specifications.
- *Were you satisfied with your design process?*
    - The design process went okay, but I would have rather had more time to plan and test the parts of the application, but with the time constraints of the semester, we did fairly well for the time we had.
- *How helpful was the agile process? How would you do the agile process next time?*
    - It was somewhat helpful, if we had to do it next time, we would probably have more regular meetings and deadlines for parts of the project.
- *Did you stick to your project plan schedule?*
    - We stuck to it some, the first increment was shortened due to several of our members traveling, and a delayed start to the project.
    - We also had trouble with miscommunication that slowed the integration of the project.
    - In increment 2 we also decided to switch the focus of the app, which caused more delays.
- *What was the real management structure within your group?*
    - Cameron mostly led the team for increment 1 and 2, increment 3 was mostly integration, but with the lack of regular meetings, the team members had more self-guidance than normal.
    - Most things were still went to Cameron for approval, but some things were done without really talking to the group.
- *Does it bear any resemblance to structure that you had planned?*
    - Some of the major pieces were still in their original timetables, but integration was moved around the most.
- *Did you have any problems getting each member to do his/her share of the work?*
    - There was not really a problem with any member, but we had one member who would wait until close to the last minute to get things done. That same team member also pushed broken code to the GitHub that other members had to fix.
- *Do you have any suggestion on how this could have been handled better?*
    - More meetings, and setting more deadlines for things to be done.
- *Discuss what you might have done differently if this were a real world project?*
    - If this was a real world project, we would have more time to focus on tasks related to the project, and be better able to maintain a schedule around this project.
- *Any recommendations for next year?*
    - Better time management

Presentation Slides

Slide 1

SMART SHOPPING

Team 11:
Cameron L'Ecuyer – Class ID: 17 (Team Leader)
Sneha Mishra – Class ID: 21
Navya Pillala – Class ID: 26
Ruthvic Punyamurtula – Class ID: 30

The work has been completed under the guidance of Dr. Yugi Lee and TAs (Rohith Nagulapati, Sidrah Junaid, Nageswara Nandigam) in CS5551 Advanced Software Engineering,
University of Missouri - Kansas City, Spring 2018

Slide 2

PROJECT'S PURPOSE

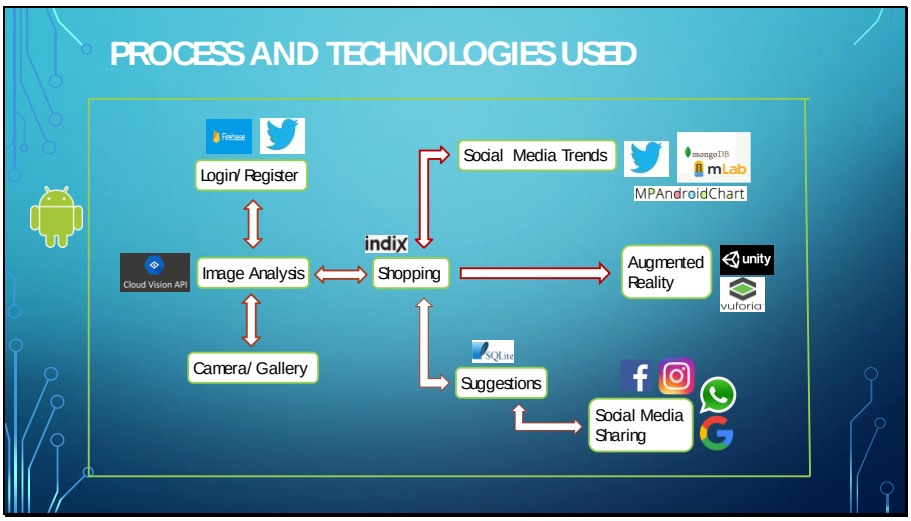
- Develop an application that can assist a user when they are shopping for furniture online
- Our app can assist our users by allowing them to use pictures to find similar items using image recognition software
- Allow the users to search products using online APIs and view furniture in Augemented reality

Slide 3

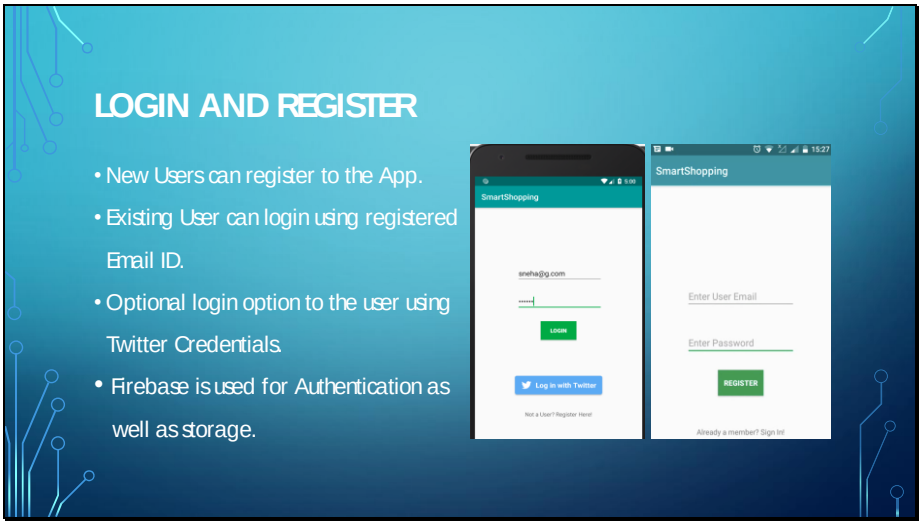
TECHNOLOGIES USED

- Android Studio for developing the Android application
- Unity 3D for developing the Augmented Reality part of the application
- Google Cloud Vision for image recognition
- Indix Product Search API for searching for products online
- BMI calculator for user suggestions
- MongoDB for Tweet data storage
- Twitter API for retrieving Tweet data
- Vuforia for image database modelling

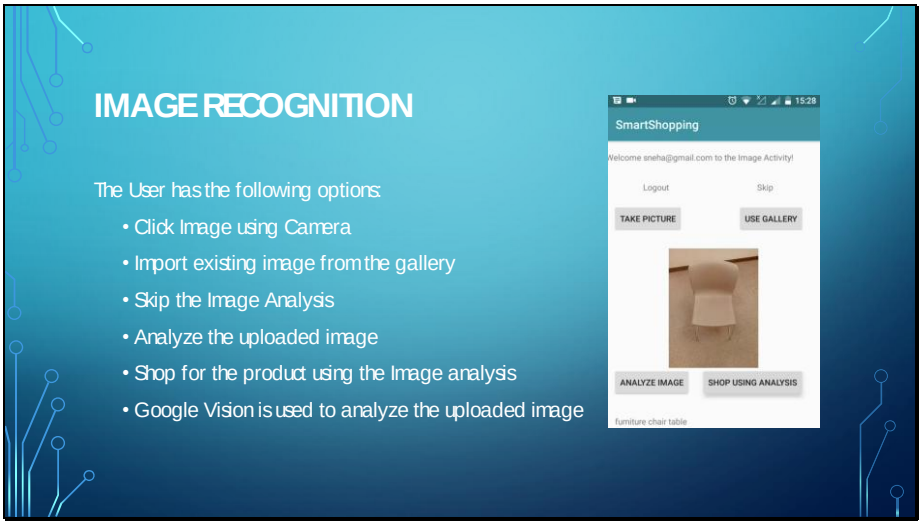
Slide 4



Slide 5



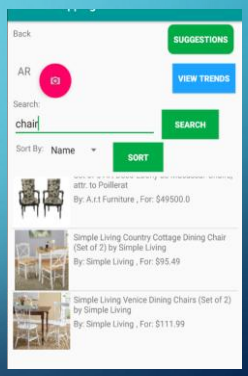
Slide 6



Slide 7

SHOPPING

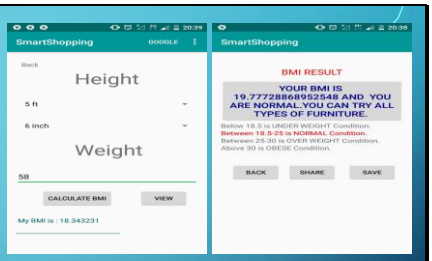
- Powered by Indix Product Search API
- Indix API allows for searching of a variety of different products
- Draw back: limited data that is available, such as limited number of images
- Data is retrieved from Indix API using Google Network Framework – Volley
- Flexible searching
- Sorting is done in code

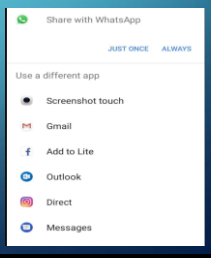


Slide 8

SUGGESTIONS & SHARING

- On clicking the suggestions button the user will get suggestions which are suitable for them.
- The BMI of the user will be calculating by taking height and weight of the user as a input.
- The user can also save the records for further usage by clicking the save button.
- The user can also share the results to social media.
- And also He can search the google by navigating to google.





Slide 9

TRENDS

- Collected Tweets using twitter4j-Stream, twitter4j-async
- Stored the furniture trends into mongo DB
- Used mongo-java-driver to support the integration in android
- Retrieved the data from the mongo DB to visualize using Do-nut charts

https://www.letsintern.com/blog/intern-worthy-skills-how-to-develop-research-and-analysis-skills/

Slide 10

AUGMENTED REALITY

- Created a Unity scene using Vuforia image modelling
- Created own markers to display varied categories of furniture
- 3D – models of furniture

Slide 11

FUTURE WORK

- Real time 3D image modelling
- Sharing the results/ content between unity and Android
- Speech to text search
- Including other categories of products – One Shopping App

The Demo video is available here -

https://www.youtube.com/watch?v=wyObDwxgtMU&t=1s

THANK YOU

- TEAM 11

Project Proposal and Plan

1. **Team members**:
    - Cameron L'Ecuyer – Class ID: 17 (Team Leader)
    - Sneha Mishra – Class ID: 21
    - Navya Pillala – Class ID: 26
    - Ruthvic Punyamurtula – Class ID: 30

2. **Project Goal**:

    Motivation:

    To develop an augmented reality application to assist users when shopping

    Significance:

    Our application will be different from others in that in will include the ability to enter augmented reality mode or allow the user to use it as a standard application. It will also be able to assist the user with food and beverage purchases, and with clothing and furniture purchasing.

    Objectives:

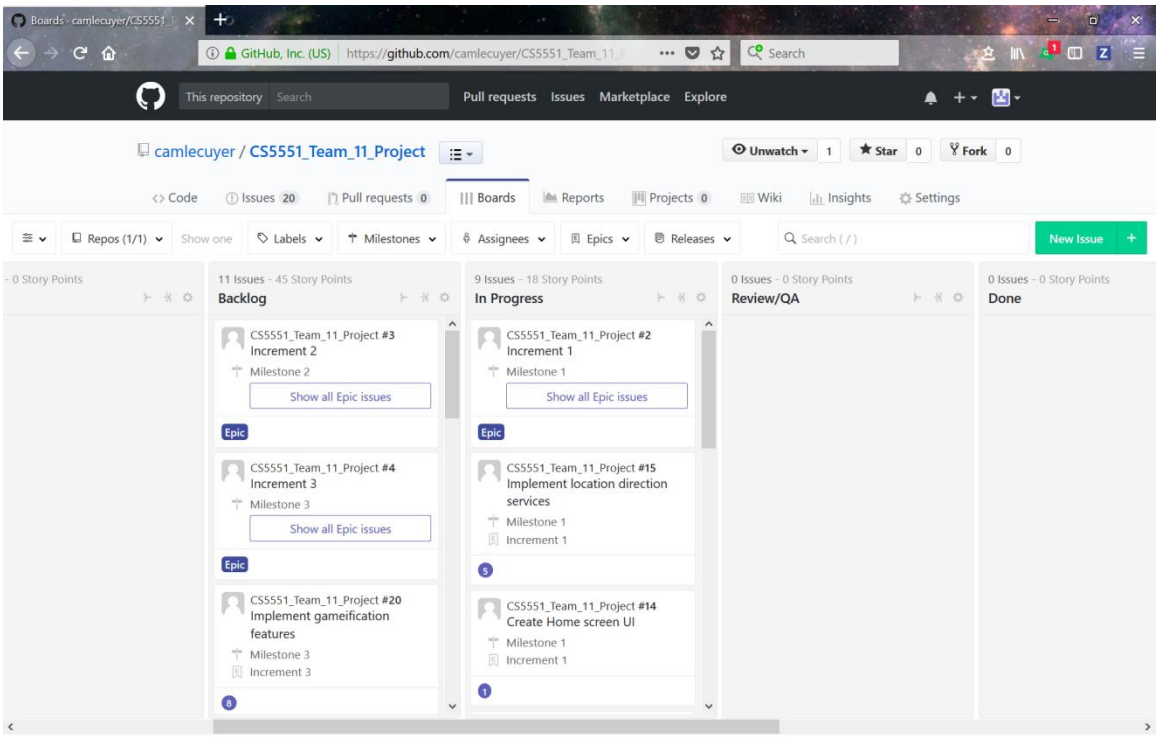
    - Develop an application that can assist a user when they are shopping
    - Allow the user to use the application in a standard assist mode, or in an augmented reality assist mode
    - Integrate social media into the application for suggestions from the user's friends and followers
    - Integrate gamification features into the application to increase user involvement and retention
    - Integrate machine learning into the software for image recognition, speech recognition, and intelligent searching

    System Features:

    - Developed for the Android operating system, and if time permits, the Apple iOS operating system
    - Clean and friendly UI
    - Integrated unobtrusive social media features
    - Integrated with virtual/augmented reality features to give the user virtual access to items, with minimal effort
        1. These features may allow the user to change the color of objects and/or replace objects with objects found through searching
        2. Options for this include: Vuforia, ARToolKit, MaxST, or other depending on our needs and budget
    - Integrated searching consumer level stores using available APIs
        1. Searching through retail stores is the main backbone of the application, and will allow users to find similar objects to what they select and learn the location and price of the options they want to buy
        2. Available APIs: Amazon ItemSearch API, Target Developer API, Walmart open API, and others to expand our area of search

- Integrated image recognition to allow for more options when searching, these features may be provided by IBM's Watson or another service if the need arises
- Integrate barcode scanning to find items quickly and easily
- Access to Speech-to-Text features for user's who have trouble with a smart phone's keyboard, poor eyesight, or who do not feel like typing
    1. Options for this include: Google's speech recognition software, IBM's Watson, or other APIs if needed
- Gamification features and options to keep the user returning to the application, and provide the user with rewards
- Integrate deals/coupons into the search mechanism if time allows

3. **Related Work**:
    - TheMine: A mobile furniture placement app
    - IKEA place: iOS AR app for furniture
    - Amazon Shopping: Furniture placement app for iOS, also uses product recognition using camera/barcode, and has product search



camlecuyer / CS5551_Team_11_Project

Unwatch 1 | Star 0 | Fork 0

Code | Issues 20 | Pull requests 0 | Boards | Reports | Projects 0 | Wiki | Insights | Settings

Implement object search by text #6

Edit | New issue

Open camlecuyer opened this issue 20 hours ago · 0 comments

camlecuyer commented 20 hours ago — Owner

As a user, I want to search from an item by typing in the name, so I can get more information about it.

Implement object search by text has no dependencies — + add dependency

See older events

camlecuyer changed the pipeline from **New Issues** to **In Progress**

camlecuyer added this to the **Milestone 1** milestone 20 hours ago

camlecuyer set the estimate to **3**

Pipeline: In Progress
Assignees: No one—assign yourself
Labels: None yet
Projects: None yet
Milestone: Milestone 1
Estimate: 3

Increment 1 #2

Edit | New issue

Open | Epic | camlecuyer opened this issue 21 hours ago · 0 comments

camlecuyer commented 21 hours ago — Owner

No description provided.

Increment 1 has no dependencies — + add dependency

Increment 1 is an Epic

0 of 8 Issues completed

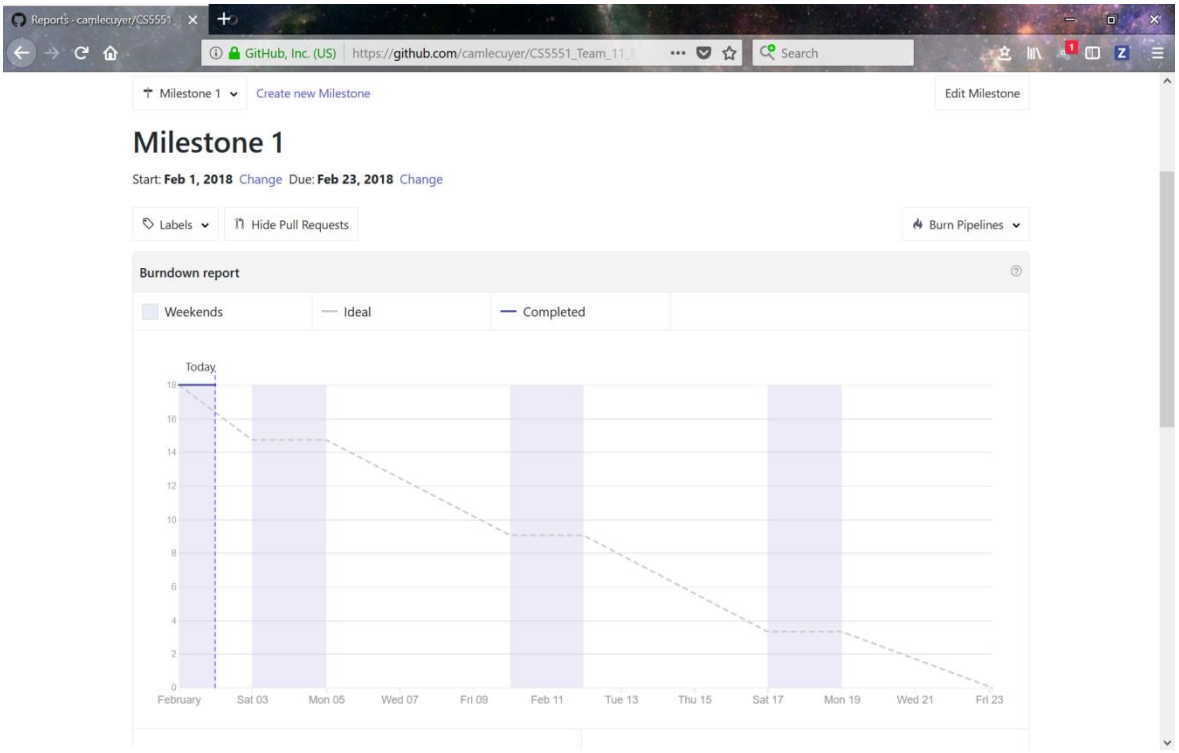
0 of 18 Epic Points completed

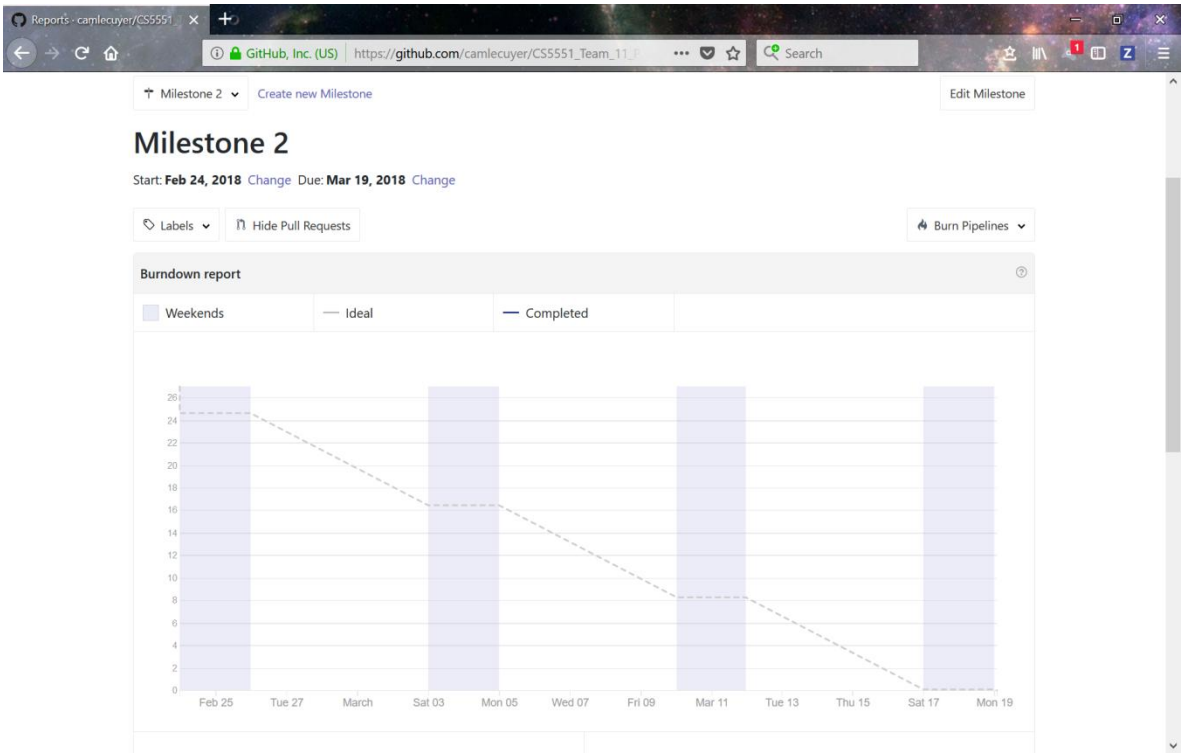
In Progress

- CS5551_Team_11_Project#1 Investigate Augmented Reality (1)
- CS5551_Team_11_Project#5 Investigate item search APIs (1)
- CS5551_Team_11_Project#6 Implement object search by text (3)
- CS5551_Team_11_Project#7 Display object info returned by search (2)
- CS5551_Team_11_Project#9 Implement search by image (3)
- CS5551_Team_11_Project#10 Implement search by barcode (2)
- CS5551_Team_11_Project#14 Create Home screen UI (1)
- CS5551_Team_11_Project#15 Implement location direction services (5)

Hide Epic Issues | Board view | Convert to Issue | Modify this Epic

Pipeline: In Progress
Assignees: No one—assign yourself
Labels: Epic
Projects: None yet
Milestone: Milestone 1
Estimate: 18 epic points
Releases: Not inside a Release
Epics


Reports · camlecuyer/CS5551
GitHub, Inc. (US)
https://github.com/camlecuyer/CS5551_Team_11
Search
Milestone 1
Create new Milestone
Edit Milestone
Milestone 1
Start: Feb 1, 2018 Change Due: Feb 23, 2018 Change
Labels
Hide Pull Requests
Burn Pipelines
Burndown report
Weekends
Ideal
Completed
Today
18
16
14
12
10
8
6
4
2
0
February
Sat 03
Mon 05
Wed 07
Fri 09
Feb 11
Tue 13
Thu 15
Sat 17
Mon 19
Wed 21
Fri 23


Reports · camlecuyer/CS5551
GitHub, Inc. (US)
https://github.com/camlecuyer/CS5551_Team_11
Search
Milestone 2
Create new Milestone
Edit Milestone
Milestone 2
Start: Feb 24, 2018 Change Due: Mar 19, 2018 Change
Labels
Hide Pull Requests
Burn Pipelines
Burndown report
Weekends
Ideal
Completed
26
24
22
20
18
16
14
12
10
8
6
4
2
0
Feb 25
Tue 27
March
Sat 03
Mon 05
Wed 07
Fri 09
Mar 11
Tue 13
Thu 15
Sat 17
Mon 19

Increment 1 Report

4. **Team members**:
    - Cameron L'Ecuyer – Class ID: 17 (Team Leader)
    - Sneha Mishra – Class ID: 21
    - Navya Pillala – Class ID: 26
    - Ruthvic Punyamurtula – Class ID: 30
    - https://github.com/camlecuyer/CS5551_Team_11_Project

5. **Project Goal**:

Motivation:

To develop an augmented reality application to assist users when shopping for clothing.

Significance:

Our application will be different from others in that in will an augmented reality mode to allow a person to "try on clothing" before buying.

Objectives:

- Develop an application that assists users who want to shop for clothing
- Integrate social media to allow suggestions from the user's friends
- Integrate gamification features into the application to increase user involvement and retention
- Integrate machine learning into the software for image recognition, speech recognition, and intelligent searching

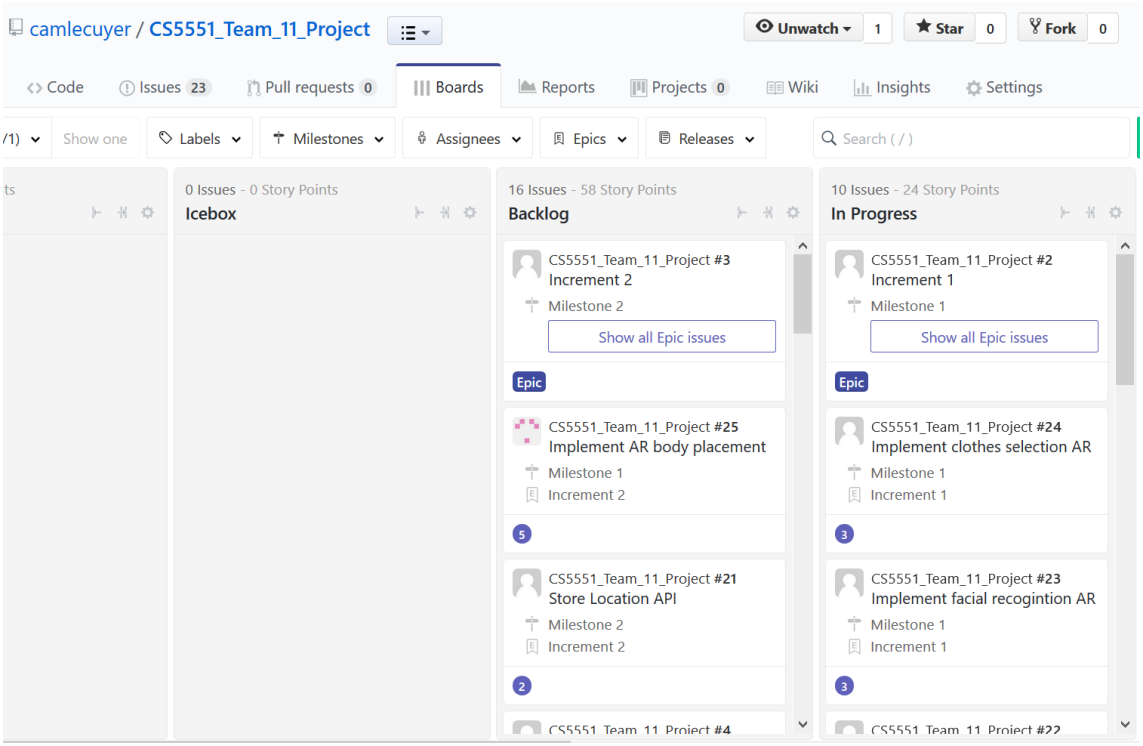
System Features:

- Developed for the Android operating system
- Clean and friendly UI
- Integrated unobtrusive social media features
- Integrated augmented reality features to give the user virtual access to clothing
    1. One feature may allow the user to change the color of clothing
    2. The user will also be able to specify their preferences
    3. The user will be shown results fitting to their body shape
- Integrated searching of retail stores using available APIs
    1. Searching through retail stores will allow users to find similar objects to what they select.
    2. They will be able to find the location and price of the what they want to buy
- Integrated image recognition will be used to determine the clothing parameters
- Access to Speech-to-Text features for user's who have trouble with a smart phone's keyboard, poor eyesight, or who do not like typing
- Gamification features to keep the user returning to the application
- If time allows, integrated deals/coupons into the search mechanism to allow for better pricing options

Related Work:

- Amazon Shopping: Product recognition that uses the camera on a smart phone, and it can do product search

6. Project Plan:



camlecuyer/CS5551_Team_11_Project#2 **Increment 1**

powered by | ZenHub

Increment 1 has no dependencies

+ add dependency

Increment 1 is an Epic

4 of 13 Issues completed

4 of 28 Epic Points completed

Backlog

CS5551_Team_11_Project#10 Implement search by barcode 2

CS5551_Team_11_Project#15 Implement location direction services 5

In Progress

CS5551_Team_11_Project#6 Implement object search by text 3

CS5551_Team_11_Project#7 Display object info returned by search 2

CS5551_Team_11_Project#9 Implement search by image 3

CS5551_Team_11_Project#14 Create Home screen UI 1

CS5551_Team_11_Project#22 Implement BMI detection/data entry 2

CS5551_Team_11_Project#23 Implement facial recogintion AR 3

CS5551_Team_11_Project#24 Implement clothes selection AR 3

Closed

CS5551_Team_11_Project#1 Investigate Augmented Reality 1

Assignees

No one—assign yourself

Labels

Epic

Projects

None yet

Milestone

Milestone 1

Estimate

28 epic points

Releases

Not inside a Release

Epics

Not inside an Epic

Notifications

camlecuyer/CS5551_Team_11_Project#6 **Implement object search by text**

powered by | ZenHub

Implement object search by text #6

Edit

Open camlecuyer opened this issue Feb 1, 2018 · 0 comments

camlecuyer commented Feb 1, 2018 Owner

As a user, I want to search from an item by typing in the name, so I can get more information about it.

Implement object search by text has no dependencies

+ add dependency

camlecuyer added this to camlecuyer/CS5551_Team_11_Project#2 Increment 1

camlecuyer changed the pipeline from New Issues to In Progress

camlecuyer added this to the Milestone 1 milestone Feb 1, 2018

camlecuyer self-assigned this Feb 22, 2018

Pipeline

In Progress

Assignees

camlecuyer

Labels

None yet

Projects

None yet

Milestone

Milestone 1

camlecuyer/CS5551_Team_11_Project#9 Implement search by image

powered by | ZenHub

Implement search by image #9

Edit

Open camlecuyer opened this issue Feb 1, 2018 · 0 comments

camlecuyer commented Feb 1, 2018 • edited

Owner

As a user, I want to search for an object by taking its picture, so that I can get information on it without typing it.

Implement search by image has no dependencies

+ add dependency

camlecuyer added this to camlecuyer/CS5551_Team_11_Project#2 Increment 1

camlecuyer changed the pipeline from New Issues to In Progress

camlecuyer added this to the Milestone 1 milestone Feb 1, 2018

camlecuyer assigned SnehaMishra28 Feb 22, 2018

Pipeline

In Progress

Assignees

SnehaMishra28

Labels

None yet

Projects

None yet

Milestone

Milestone 1

Labels

Hide Pull Requests

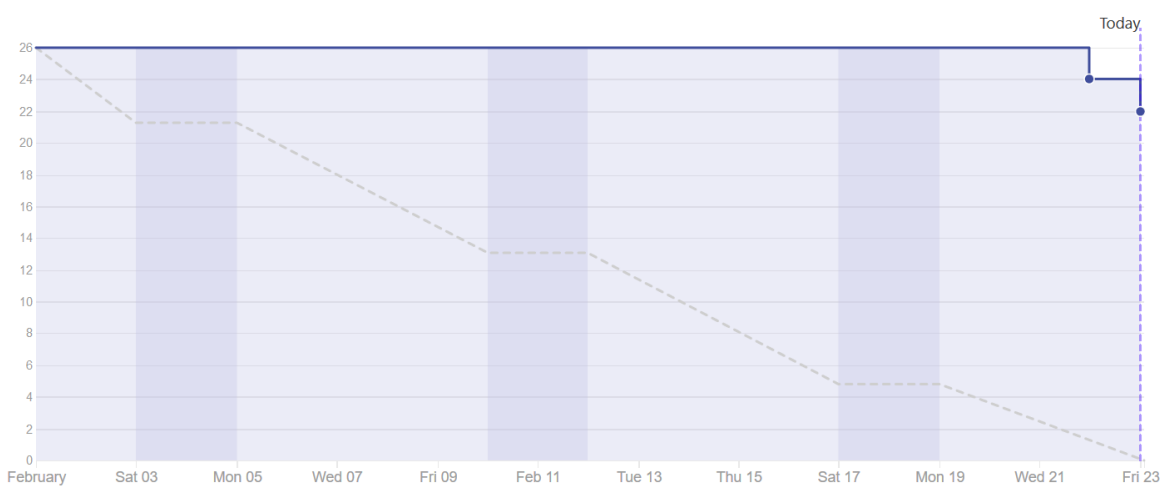
Burn Pipelines

Burndown report

Weekends

Ideal

Completed

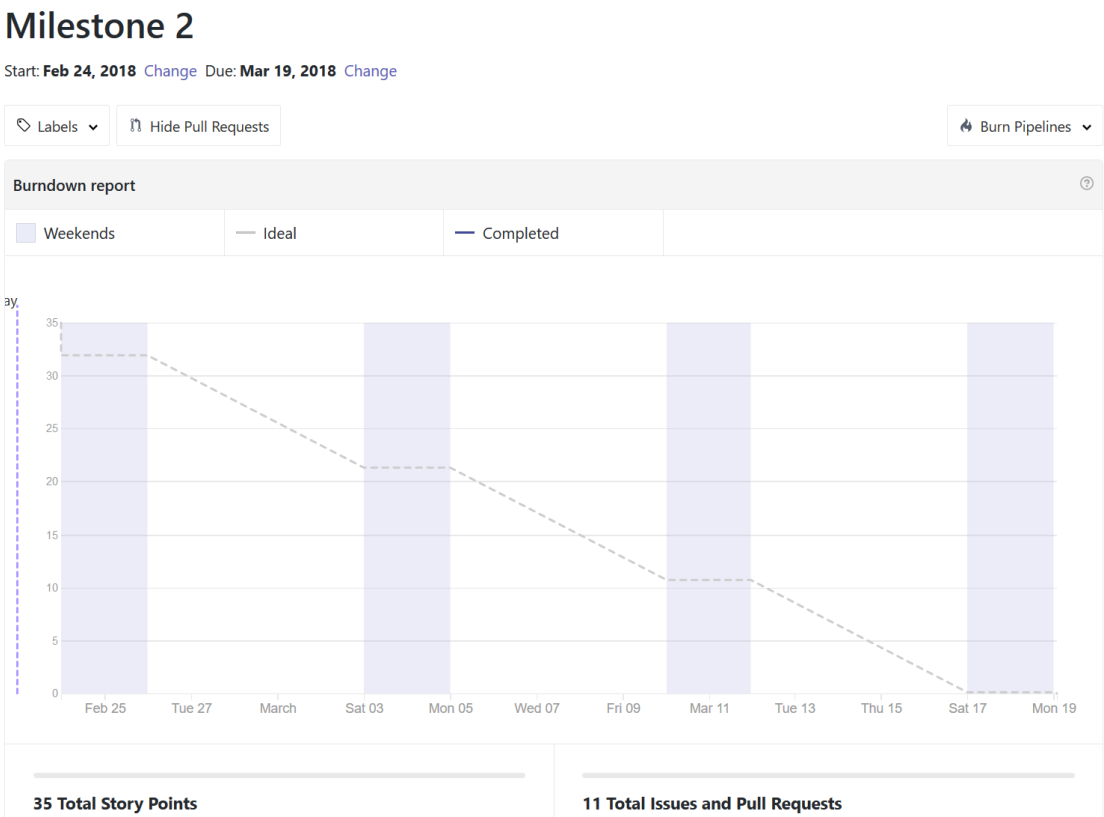


26 Total Story Points

4 Completed / 22 Remaining

13 Total Issues and Pull Requests

4 Completed / 9 Remaining

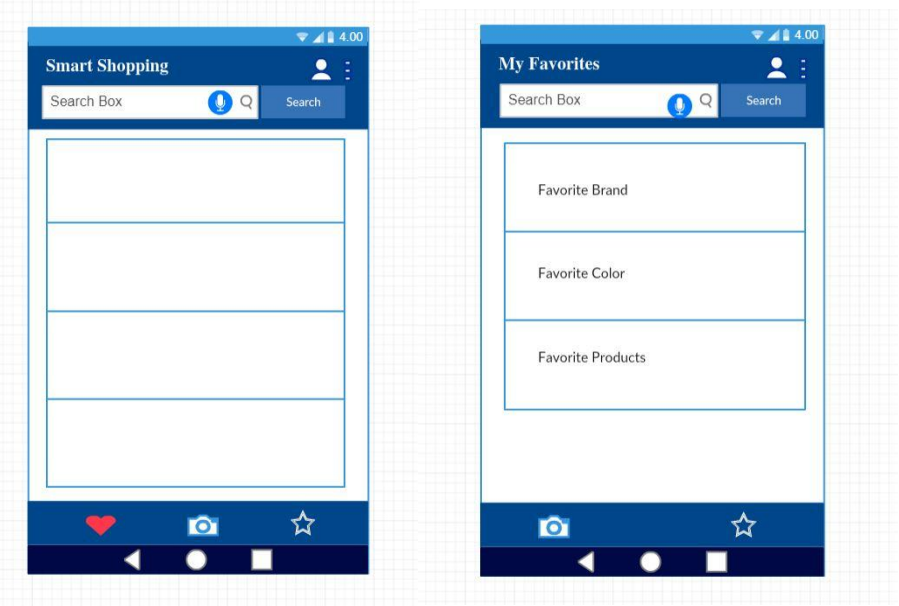


7. **First Increment Report**
    - Each team member submitted what they had completed and the APIs/services they used
    - Rothvic created the UML diagrams and Mock ups

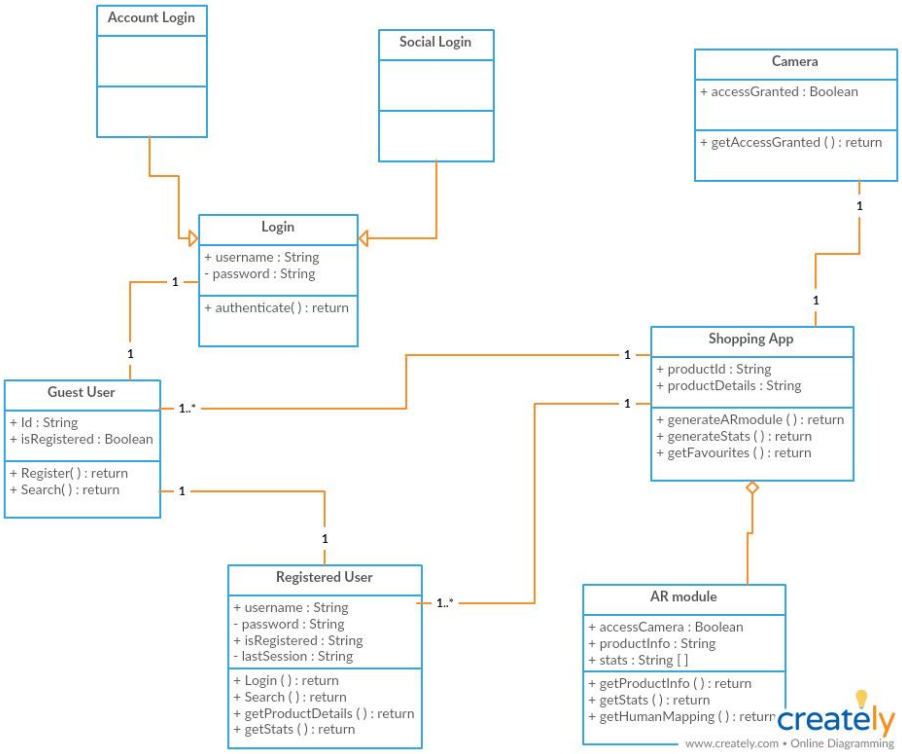
1. **Existing Services/REST API**
    - https://www.faceplusplus.com/body-outlining/#demo
    - https://www.betafaceapi.com/wpa/
    - https://stackoverflow.com/questions/21698044/basic-bmi-calculator-html-javascript
    - Google Reverse Image Search
    - Amazon Product Advertising API
    - Target API: https://developer.target.com/
    - Walmart API: https://developer.walmartlabs.com/
    - Macy's API: http://developer.macys.com/
    - Google Network Framework - Volley for Android

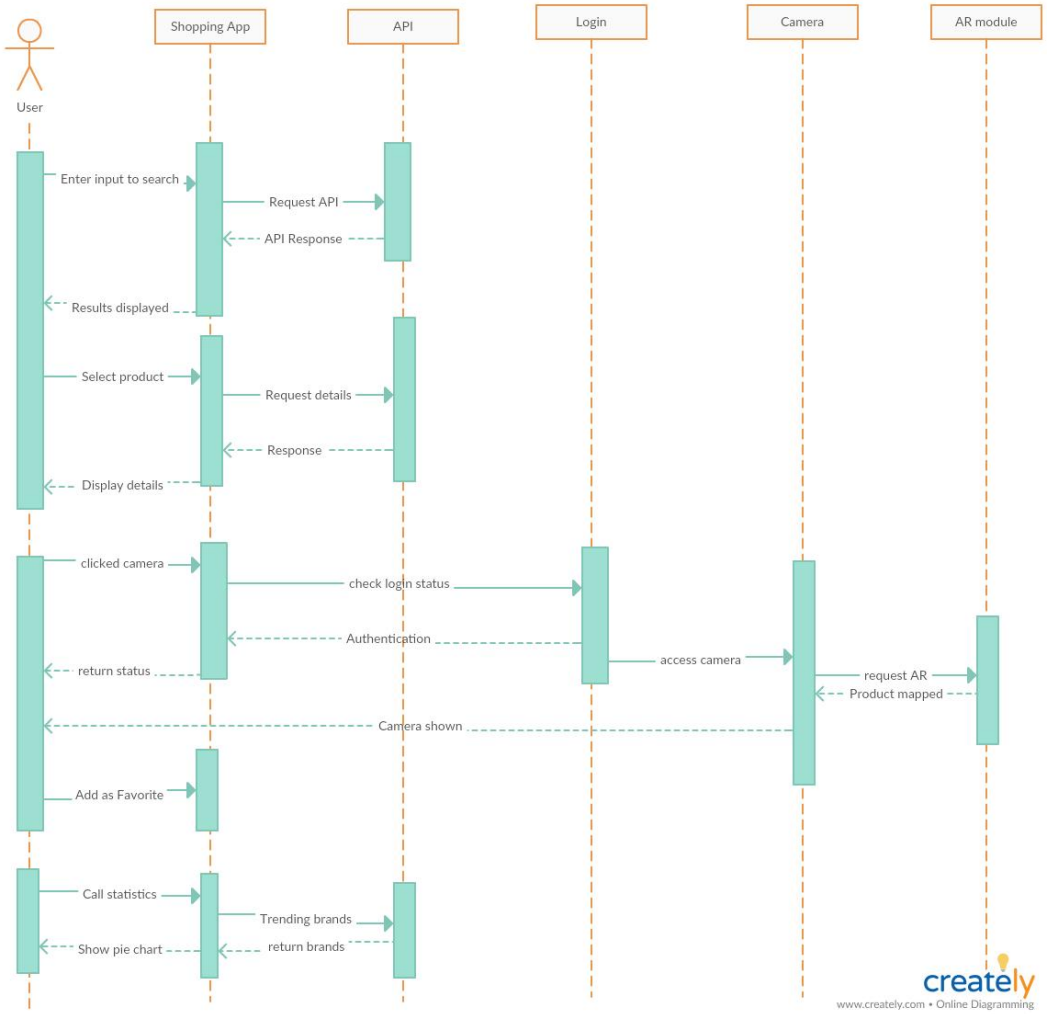
2. **Detail Design of Features:**

    UI Mockups



UML Diagrams


Account Login
Social Login
Camera
+ accessGranted : Boolean
+ getAccessGranted ( ) : return
Login
+ username : String
- password : String
+ authenticate( ) : return
Shopping App
+ productId : String
+ productDetails : String
+ generateARmodule ( ) : return
+ generateStats ( ) : return
+ getFavourites ( ) : return
Guest User
+ Id : String
+ isRegistered : Boolean
+ Register( ) : return
+ Search( ) : return
Registered User
+ username : String
- password : String
+ isRegistered : String
- lastSession : String
+ Login ( ) : return
+ Search ( ) : return
+ getProductDetails ( ) : return
+ getStats ( ) : return
AR module
+ accessCamera : Boolean
+ productInfo : String
+ stats : String [ ]
+ getProductInfo ( ) : return
+ getStats ( ) : return
+ getHumanMapping ( ) : return
1
1..*
1
1
1
1..*
1
1
1
1
1
1..*
1
1
1
1
1
1

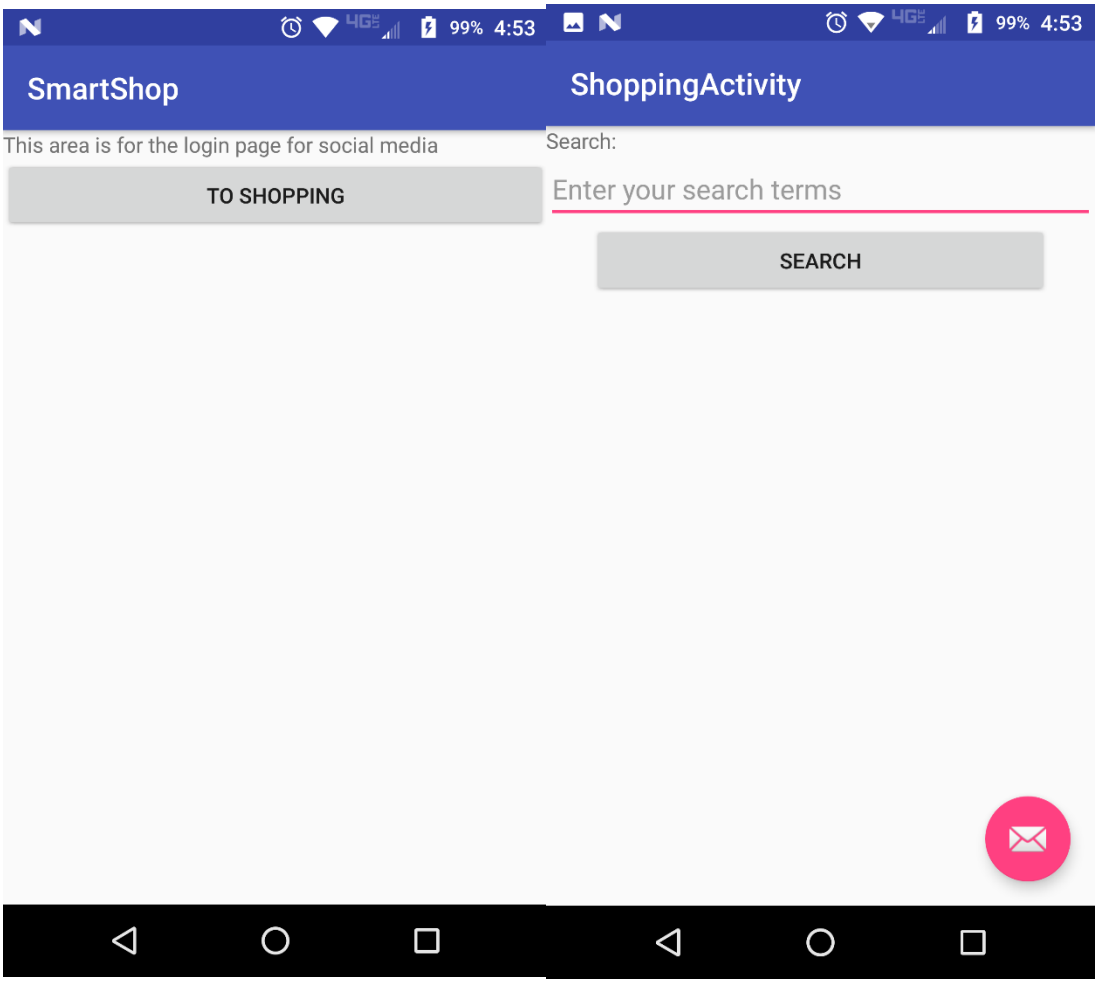


- **Testing**

SNO	TEST CASE	DESCRIPTION	EXPECTED OUTCOME	RESULT
1	To detect skin color	The API should detect skin color of given image	API must work	
2	To define BMI of a person	BMI of a person should be calculated based on height and weight of a person	Exact BMI of a person is calculated	success

- **Implementation**
  - Android Studio
  - Exploration of features done in HTML and JavaScript
- **Deployment**

- Currently we only have the home page and the start of the page for the clothing search



Sample Code snippet of Betaface API

```
public interface Betaface {

    @POST("GetImageInfo")
    Observable<Call<GetImageInfoResponse>> getImageInfo(@Body GetImageInfoRequest request);

    @POST("GetRecognizeResult")
    Observable<Call<GetRecognizeResultResponse>> getRecognizeResult(@Body GetRecognizeResultRequest request);

    @POST("RecognizeFaces")
    Observable<Call<RecognizeFacesResponse>> recognizeFaces(@Body GetRecognizeResultRequest request);

    @POST("UploadImage")
    Observable<Call<UploadImageResponse>> uploadImage(@Body UploadImageRequest request);

    @POST("UploadNewImage_Url")
    Observable<Call<UploadNewImage_UrlResponse>> uploadNewImage_Url(@Body UploadNewImage_UrlRequest request);
}
```

Sample code snippet of BMI**:**

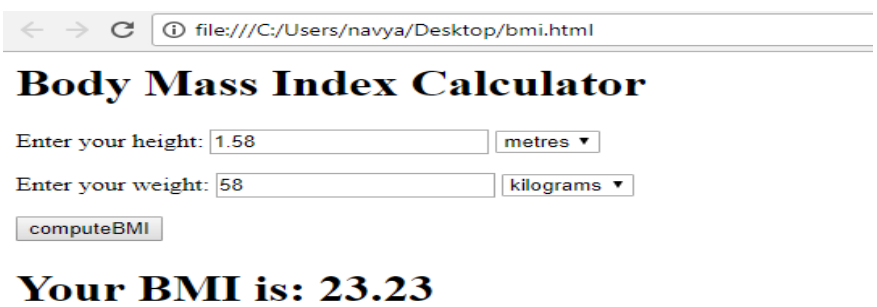
```
            //Convert all units to metric
            if (heightunits=="inches") height/=39.3700787;
            if (weightunits=="lb") weight/=2.20462;

            //Perform calculation
            var BMI=weight/Math.pow(height,2);

            //Display result of calculation
            document.getElementById("output").innerText=Math.round(BMI*100)/100;

            if (output<18.5)
            document.getElementById("comment").value = "Underweight";
            if (output>=18.5 && output<=25)
            document.getElementById("comment").value = "Normal";
            if (output>=25 && output<=30)
            document.getElementById("comment").value = "Obese";
            if (output>30)
            document.getElementById("comment").value = "Overweight";
            document.getElementById("answer").value = output;
        }
    </script>
/head>
body>
    <h1>Body Mass Index Calculator</h1>
    <p>Enter your height: <input type="text" id="height"/>
        <select type="multiple" id="heightunits">
            <option value="metres" selected="selected">metres</option>
            <option value="inches">inches</option>
        </select>
    </p>
    <p>Enter your weight: <input type="text" id="weight"/>
        <select type="multiple" id="weightunits">
            <option value="kg" selected="selected">kilograms</option>
            <option value="lb">pounds</option>
        </select>
    </p>
    <input type="submit" value="computeBMI" onclick="computeBMI();">
    <h1>Your BMI is: <span id="output">?</span></h1>
```

Output of BMI code**:**



3. **Project Management**
    - **Work Completed**
        - Ruthvic
            - Created android UI mock up using wire frames for the respective pages which helps in building the app layout
            - Overall design of the application is represented using class diagram and sequence diagram
            - Researched on body detection API
            - I will be working on modeling the clothing product to the human in camera vision
        - Navya
            - Implemented BMI calculator
            - Implemented facial recognition
                - Betaface API is used for facial recognition. It can scan uploaded image files or image URL to find and analyze faces
                - This API helps us to estimate gender, age, ethnicity, emotion (smile/neutral), eyes, skin color, clothes and background colors etc
        - Sneha
            - Worked on image search
            - Worked on Walmart API
        - Cameron
            - Worked on Walmart API
            - Worked on Amazon API
            - Worked on Target API
            - Worked on Macy's API
            - Built UI for Android app
    - **Work to Be Completed**
        - Ruthvic
            - Finish implementing Body AR
        - Sneha
            - Image recognition
        - Cameron
            - Due to problems getting a product API, I may have to implement a static database for product items before the searching will work
            - Search for objects using text
            - Display search results

4. **Issue**
    - Will we be able to map the product on the image?
    - Google Reverse Image Search APIs are deprecated
    - The retail APIs are causing problems:
        - Target API is only available to Target affiliates
        - Walmart API will not send confirmation email to access site
        - Macy's API gave a key, but will not activate it
        - Amazon API requires Amazon's Associates Program, which requires a fully developed and deployed website/app for them to review before you are approved

Increment 2 Report

1. Team members:

- Cameron L'Ecuyer – Class ID: 17 (Team Leader)
- Sneha Mishra – Class ID: 21
- Navya Pillala – Class ID: 26
- Ruthvic Punyamurtula – Class ID: 30
- https://github.com/camlecuyer/CS5551_Team_11_Project

2. Project Goal:

Motivation:

Develop an application that makes use of augmented reality to assist user with shopping for furniture.

Significance:

Our application will be different from others, in that, it will allow users to select items, not just use pre-built models.

Objectives:

- Develop a mobile application that can assist users shopping for furniture
- Integrate social media, like Twitter, to gather suggestions from trending brands, or friend recommendations
- Introduce gamification features into the app to improve user retention and interaction

System Features:

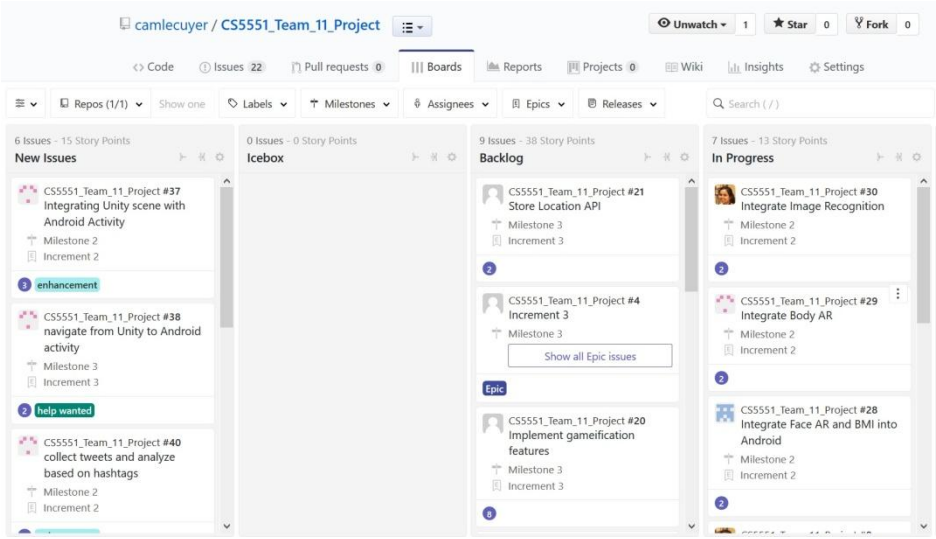
- Developed for mobile operating systems, i.e. Android
- A clean and user friendly UI
- Unobtrusive integration of social media interactions
- Augmented reality integration to allow the user to virtually see furniture
    1. Ability to change furniture color, time permitting
- Searching online retail services using APIs
    1. By searching retail store services, the user can select what they like, and the system can return suggestions related to that item
    2. The user will be able get the location, price, and details of the what they want to buy, if the APIs give the information
- Image recognition will allow the system to analyze an image to determine the shape and color
- Speech-to-Text features for more options for searching, time permitting
- Gamification features to interact more with users
- Access to deals/coupons for better pricing, if time allows

Related Work:

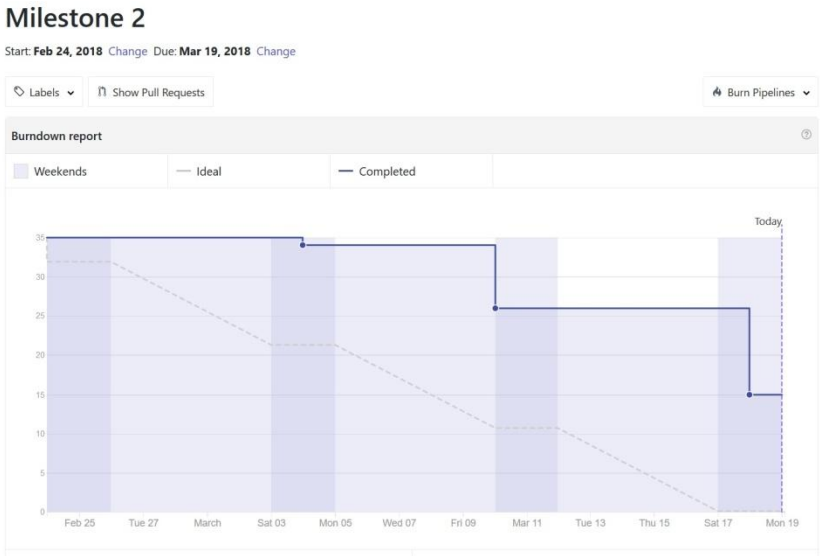
- Amazon Shopping: Using a mobile device's camera, the app can identify what is being viewed
- IKEA Place: Allows viewing of IKEA furniture using augmented reality

3. Project Plan

These are images of the boards taken close to the end of increment 2. New issues were added recently that have yet to be moved, and the increment 3 issues have not been moved to active yet



This is the burndown chart at about 3:30 PM on March 19



This is a screenshot of the closed issues at about 3:30 PM on March 19

Completed Issues	Story points
Add augmented reality features for displaying search results CS5551_Team_11_Project #12 New Issues Milestone 2	5
Implement augmented reality object placement CS5551_Team_11_Project #13 In Progress Milestone 2	5
Add search result sorting and searching CS5551_Team_11_Project #17 In Progress Milestone 2	3
Generate data for use with searching CS5551_Team_11_Project #31 New Issues Milestone 2	1
Implement augmented reality color selection CS5551_Team_11_Project #32 New Issues Milestone 2	3
Database creation in Vuforia and integrating it to unity enhancement CS5551_Team_11_Project #35 In Progress Milestone 2	1
Creating multiple markers in unity, and detecting them using AR camera enhancement CS5551_Team_11_Project #36 In Progress Milestone 2	2

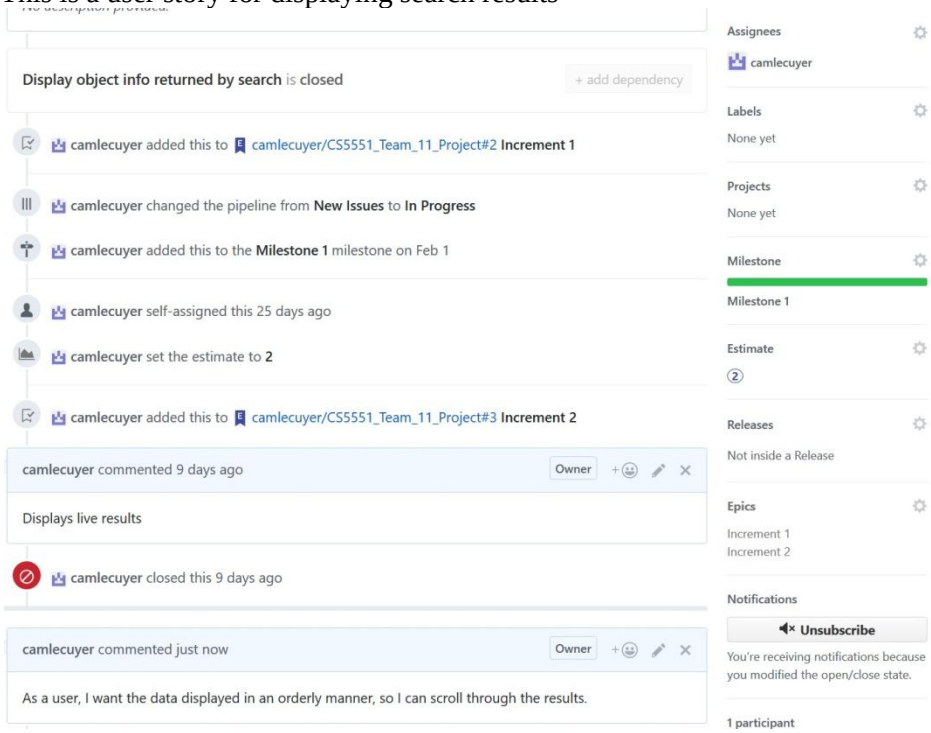
This is a screenshot of the open issues at about 3:30 PM on March 19

35 Total Story Points
20 Completed / 15 Remaining

15 Total Issues
7 Completed / 8 Remaining

Remaining Issues	Story points
Increment 2 Epic CS5551_Team_11_Project #3 In Progress Milestone 2	Not estimated
Implement user preferences CS5551_Team_11_Project #8 In Progress Milestone 2	3
Integrate Face AR and BMI into Android CS5551_Team_11_Project #28 In Progress Milestone 2	2
Integrate Body AR CS5551_Team_11_Project #29 In Progress Milestone 2	2
Integrate Image Recognition CS5551_Team_11_Project #30 In Progress Milestone 2	2
Integrate Firebase CS5551_Team_11_Project #33 In Progress Milestone 2	1
Integrating Unity scene with Android Activity enhancement CS5551_Team_11_Project #37 New Issues Milestone 2	3
collect tweets and analyze based on hashtags enhancement CS5551_Team_11_Project #40 New Issues Milestone 2	2

This is a user story for displaying search results



This is the burndown chart for Milestone 3

4. Increment 2 Report

- Each team member submitted what they had completed and the APIs/services they used
- Ruthvic created some of the UML diagrams and mock ups, Cameron created other mock ups

1. Existing Services/REST API

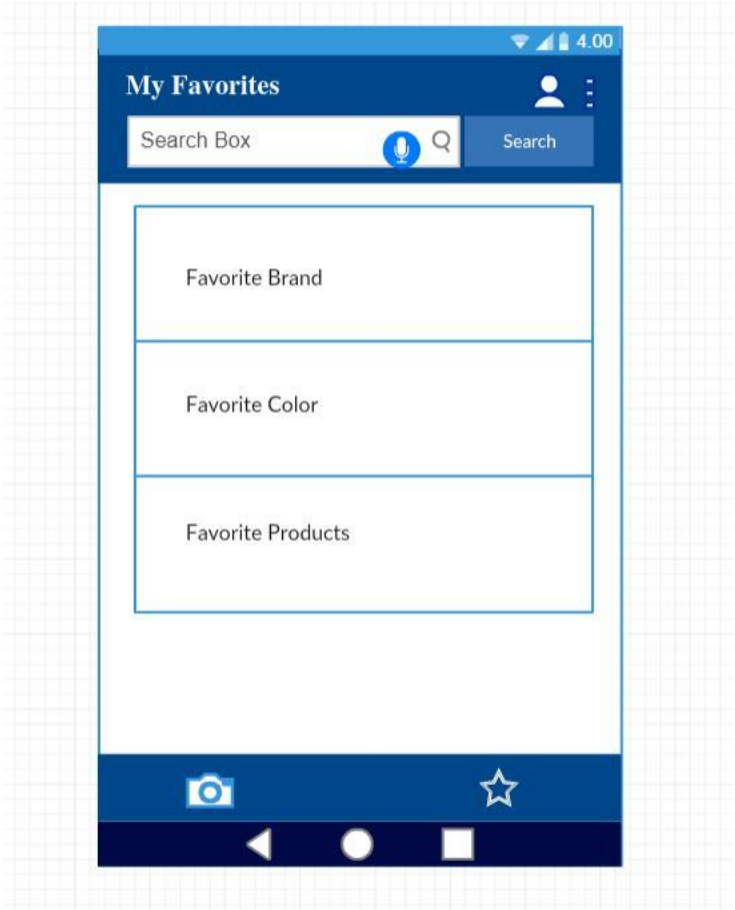
- Face++
- BetaFace API
- BMI Calculator with JavaScript
- Azure Computer Vision
- Google Network Framework - Volley for Android
- Indix Product Search API

2. Detail Design of Features:

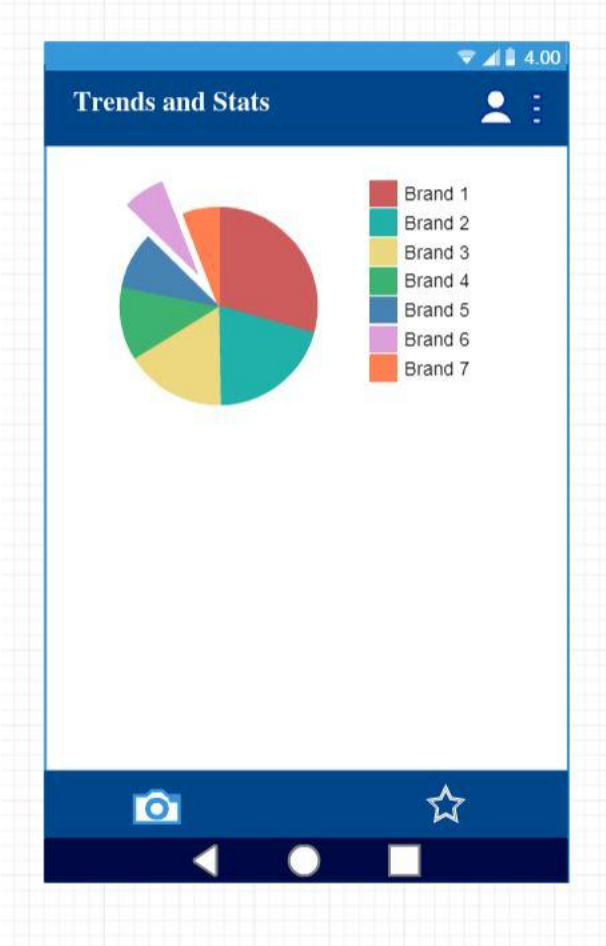
UI Mockups

These Mock Ups were created by Cameron early in Increment 2, the pages in order are: login, face and BMI activity, search activity, body activity, and results activity

These Mock Ups were created by Ruthvic during Increment 1, this image shows what the favorites page may have looked like

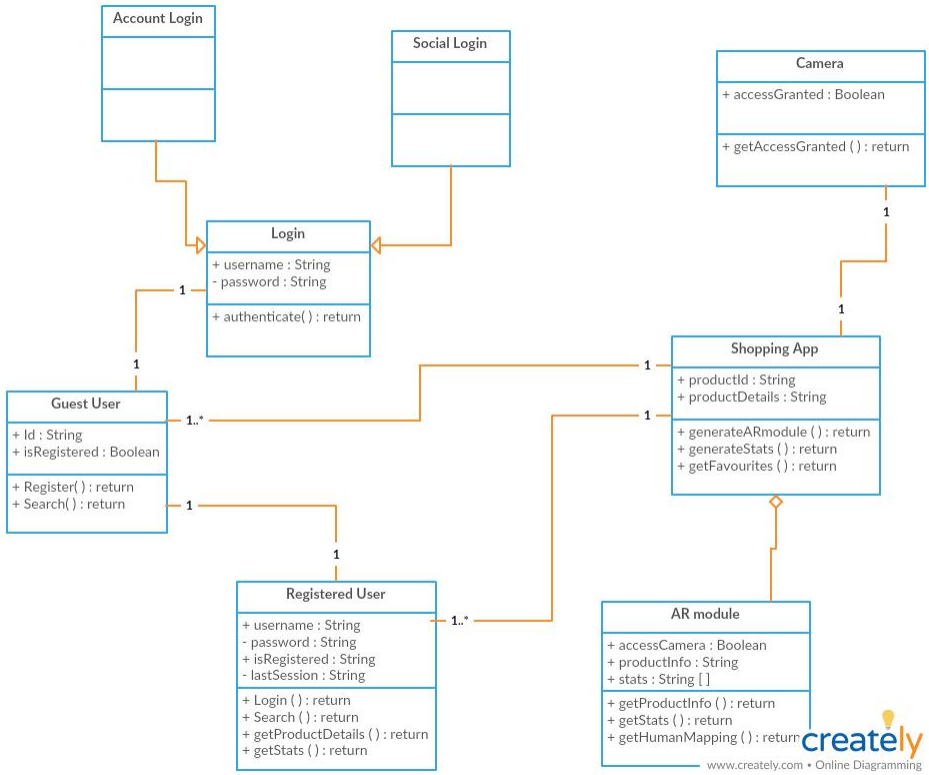

4.00
My Favorites
Search Box
Search
Favorite Brand
Favorite Color
Favorite Products

This image shows what the graphs created using social media data may look like

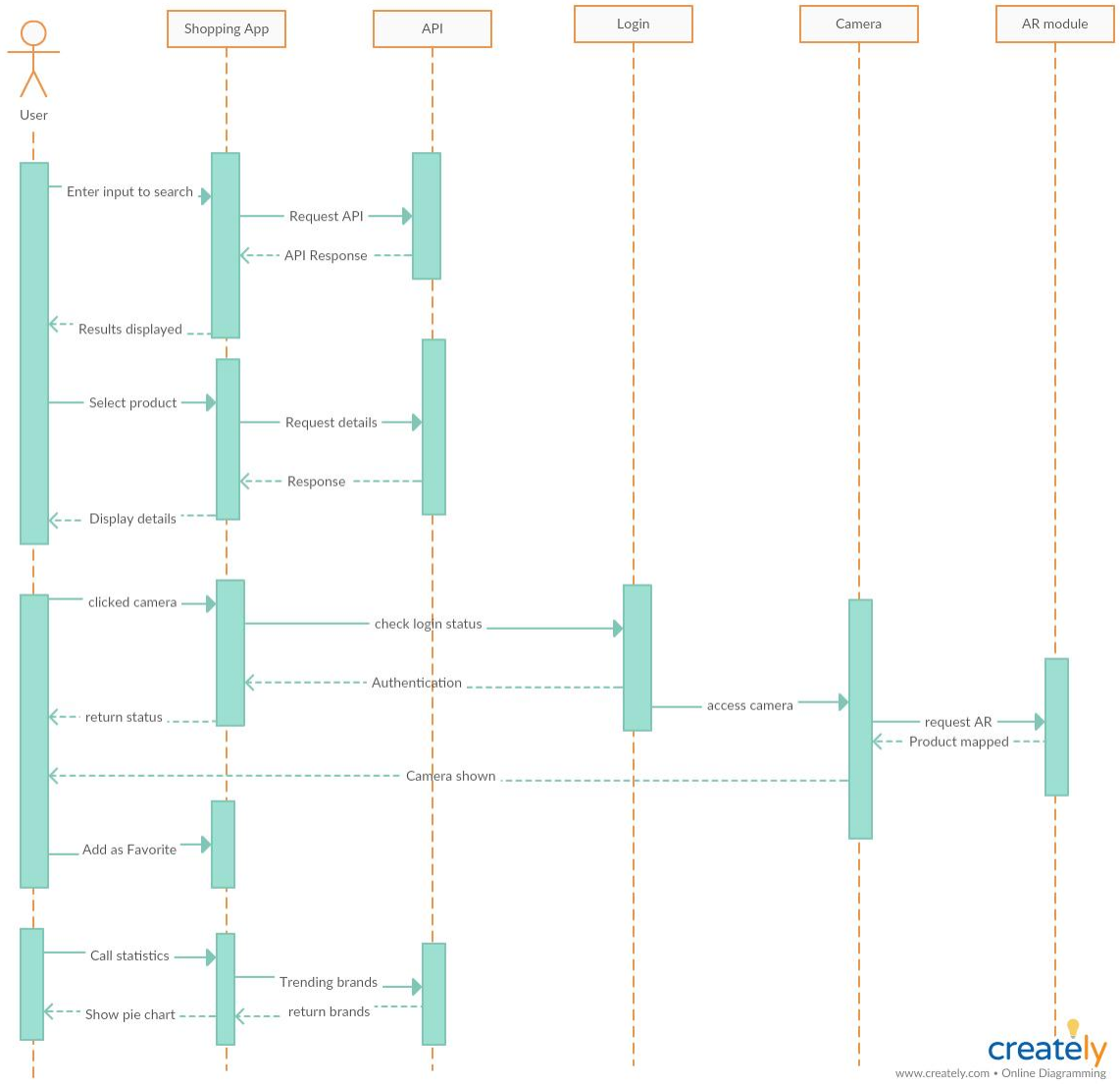


UML

These were both created by Ruthvic This is the class diagram created during Increment 1



This is the sequence diagram for using our app, created during Increment 1

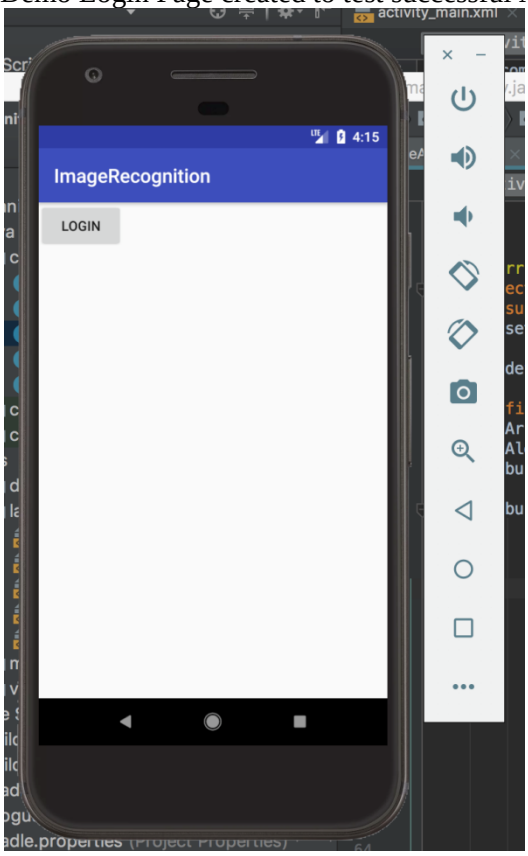


3. Testing

Sneha:

Unit testing done to test the basic functionality of the image recognition Activity. Screenshots below indicate the activity transition.

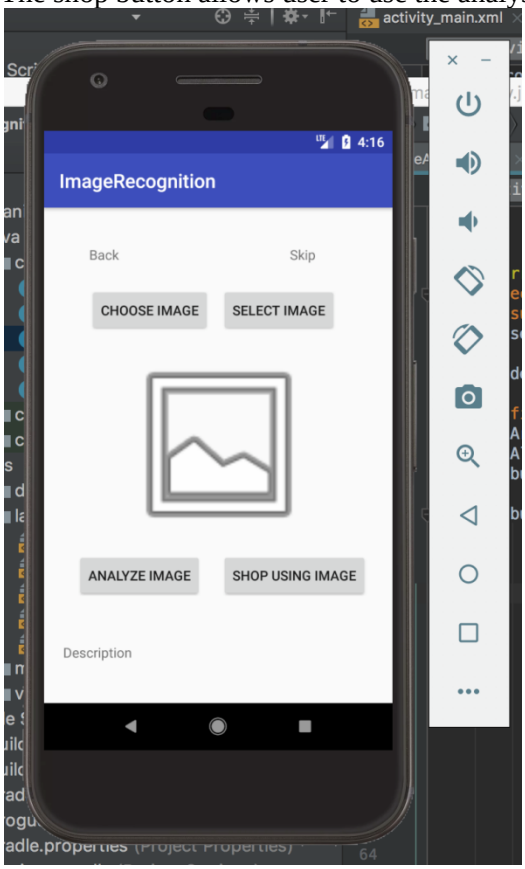
Demo Login Page created to test successful Activity transition.



Home Page for the Image activity. Here the different buttons are used for different feature implementation.

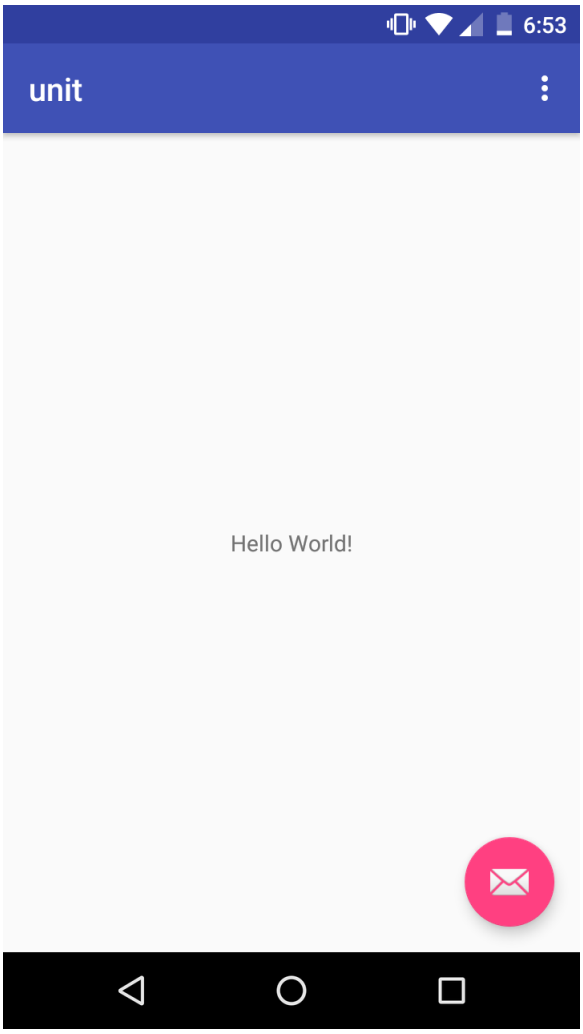
- The back and skip is to allow user to move back to the login page or to skip the current image recognition activity and move directly to the shopping page.
- The capture button allows user to click a picture for image analysis.
- The select button allows user to select a pre-clicked image from the gallery.
- The analyze button will analyze the selected image using the Microsoft cognitive API.

- The shop button allows user to use the analysis in shopping.



Ruthvic:

Screen shot of android home page from which unity scene is called. The floating button is used to call the AR camera



This navigates from android activity to unity scene, and initializes AR camera

In unity, for the product to be displayed, I have used a currency note as a marker. Also used human body as a marker to project the 3d image of a shirt on to the body.

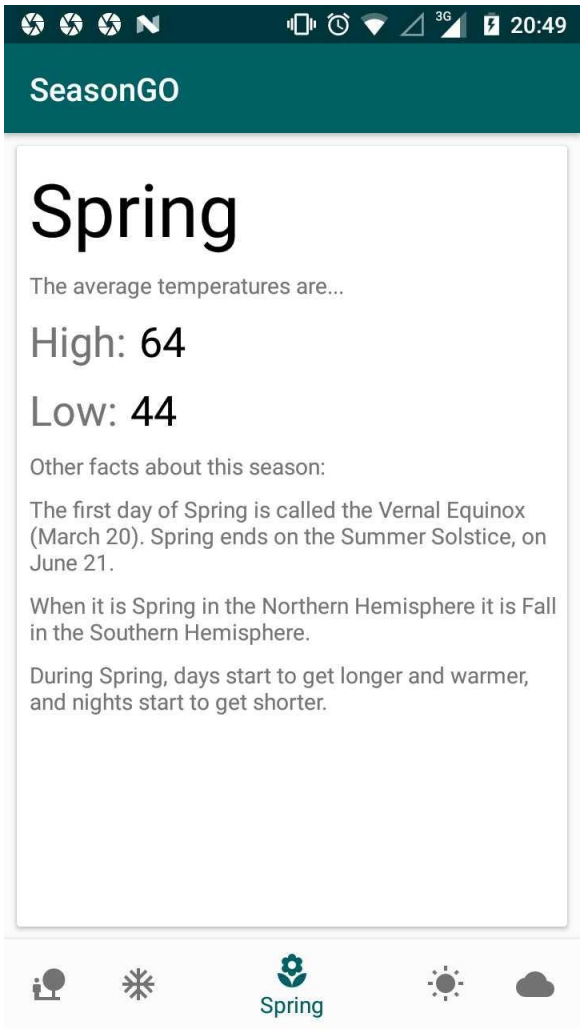
Navya:

WEATHER APP THAT DISPLAYS CURRENT WEATHER

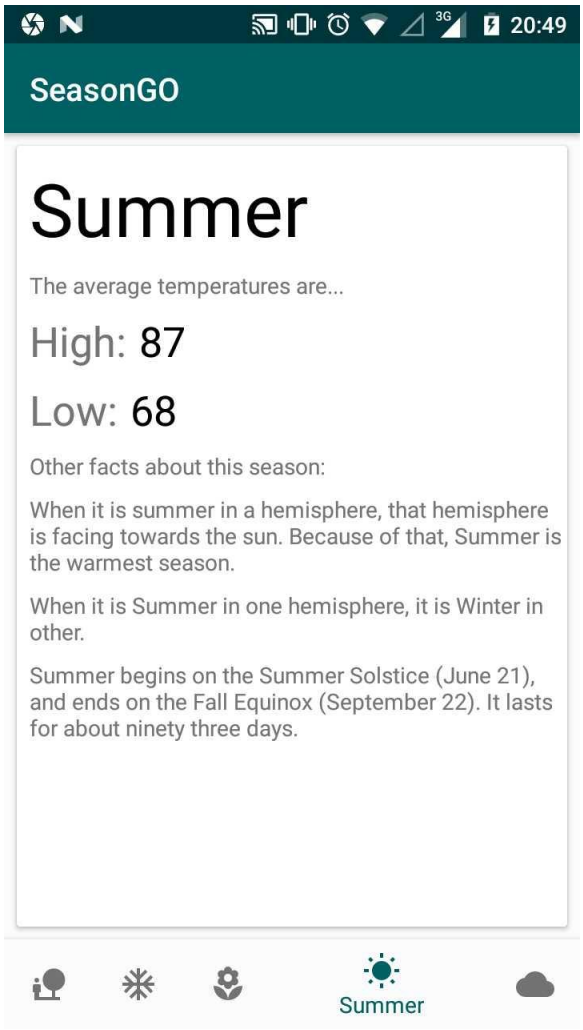
WINTER

20:49
SeasonGO
Winter
The average temperatures are...
High: 42
Low: 27
Other facts about this season:
Winter begins on Winter Solstice (December 21), and ends on Vernal Equinox (March 20). Winter is about eighty eight days.
During winter, the hemisphere that it's in faces away from
the sun. This causes that hemisphere to have
long nights, low temperatures, and short days.
Winter gets the most indirect sunlight out of all the seasons.
Winter

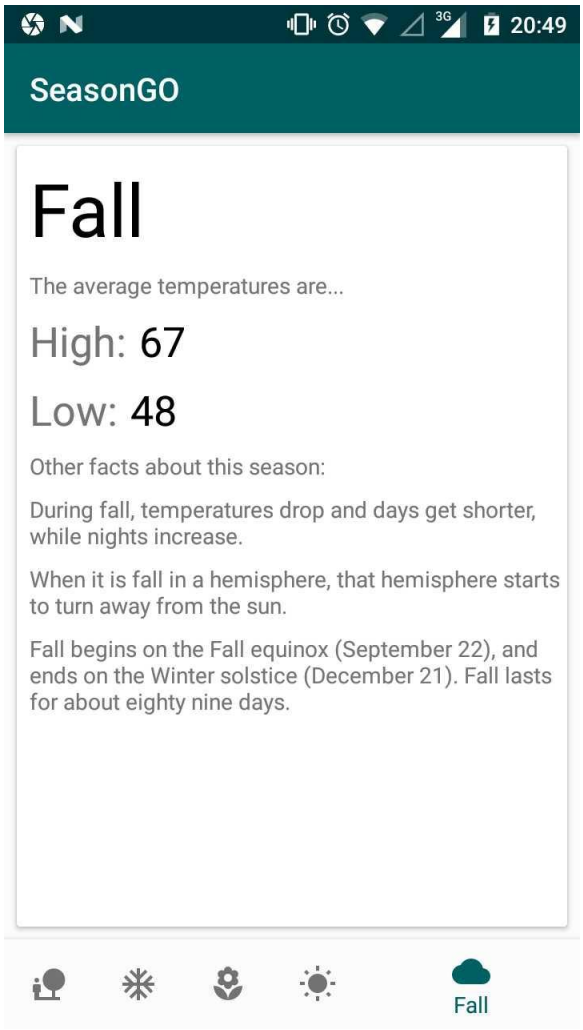
SPRING


20:49
SeasonGO
Spring
The average temperatures are...
High: 64
Low: 44
Other facts about this season:
The first day of Spring is called the Vernal Equinox (March 20). Spring ends on the Summer Solstice, on June 21.
When it is Spring in the Northern Hemisphere it is Fall in the Southern Hemisphere.
During Spring, days start to get longer and warmer, and nights start to get shorter.
Spring

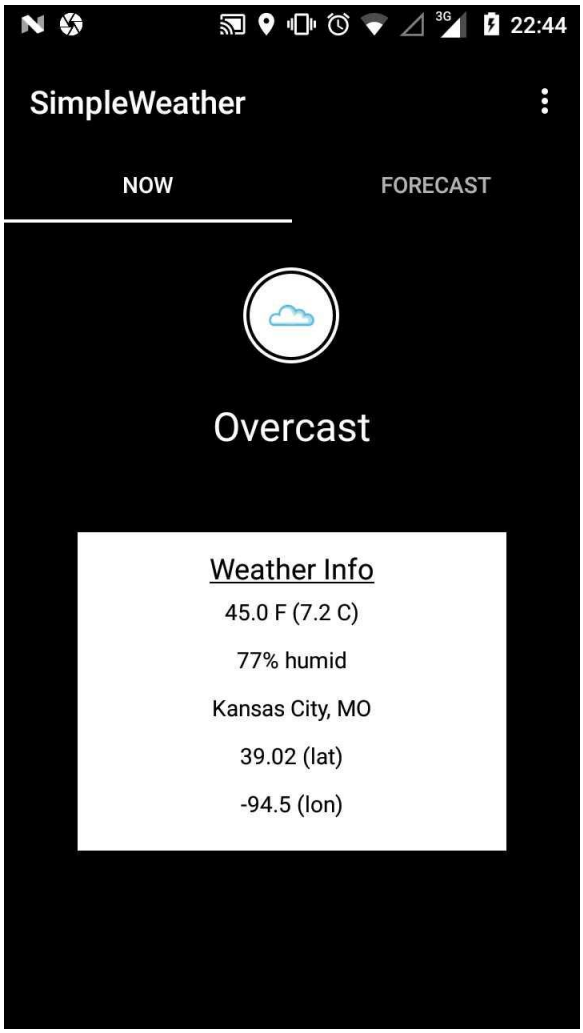
SUMMER


20:49
SeasonGO
Summer
The average temperatures are...
High: 87
Low: 68
Other facts about this season:
When it is summer in a hemisphere, that hemisphere is facing towards the sun. Because of that, Summer is the warmest season.
When it is Summer in one hemisphere, it is Winter in other.
Summer begins on the Summer Solstice (June 21), and ends on the Fall Equinox (September 22). It lasts for about ninety three days.
Summer

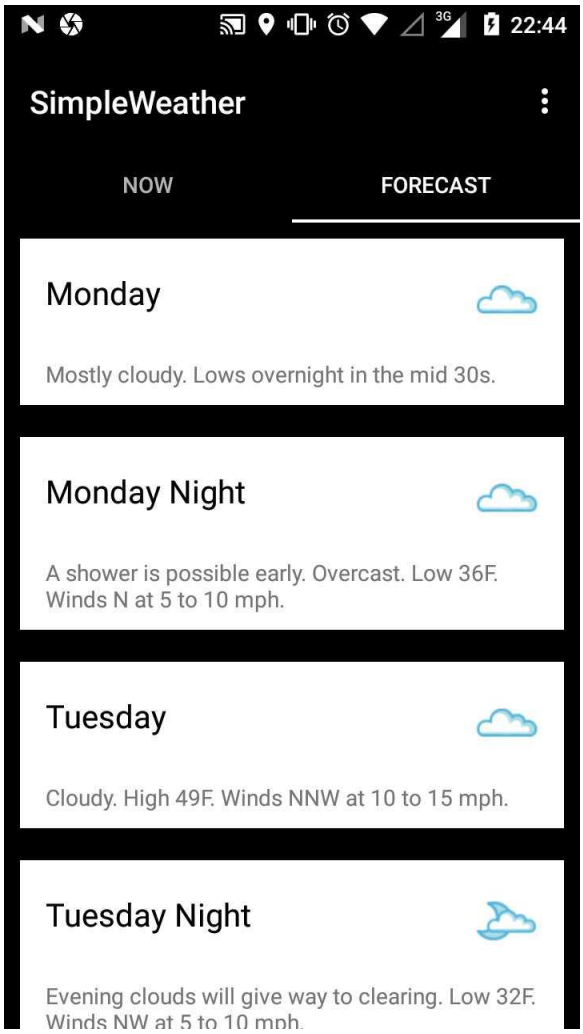
FALL



WEATHER APP THAT DISPLAYS CURRENT DAY WEATHER

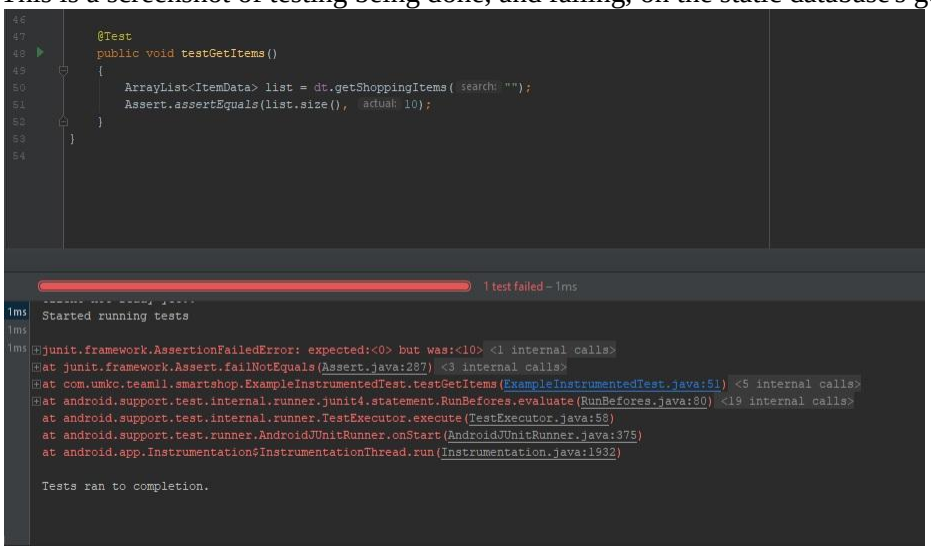


FORECAST OF WEATHER APP SHOWING THE WEATHER FOR NEXT THREE DAYS

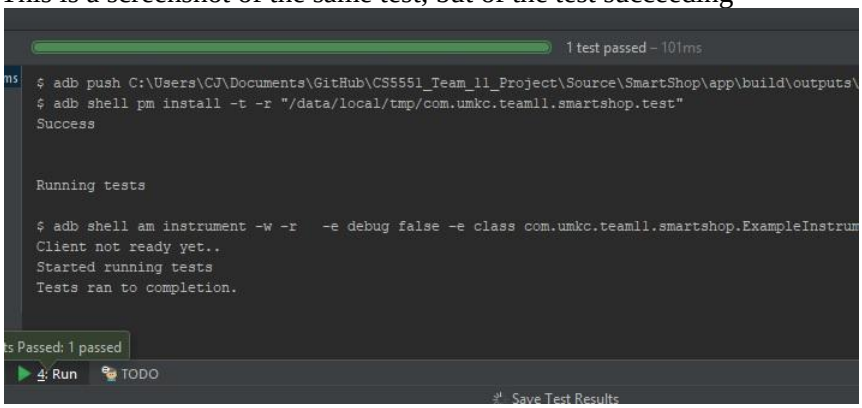


Cameron:

This is a screenshot of testing being done, and failing, on the static database's getItem function



This is a screenshot of the same test, but of the test succeeding

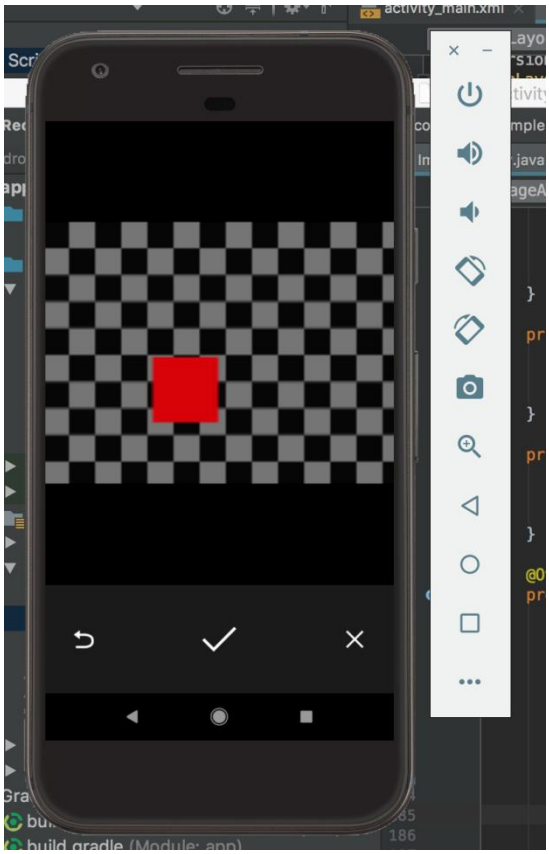


4. Deployment

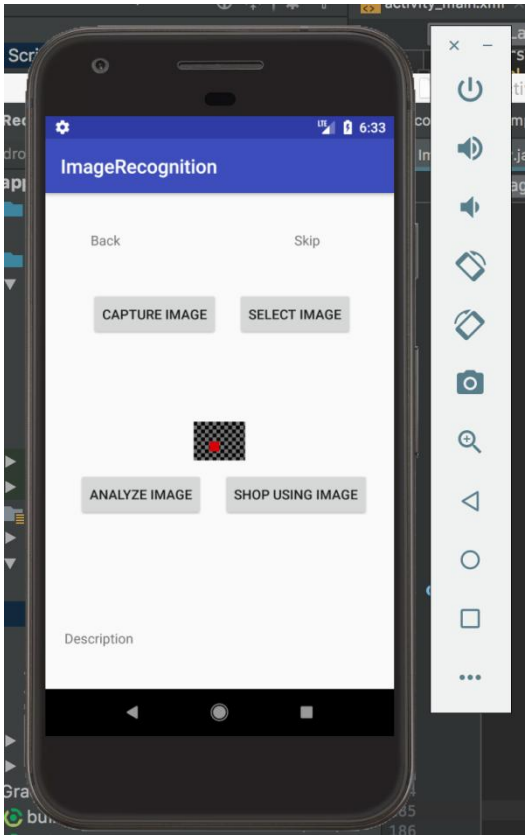
Sneha:

Hardware not available. Deployed on the emulator provided by android studio. Results are as shown in the screenshot below.

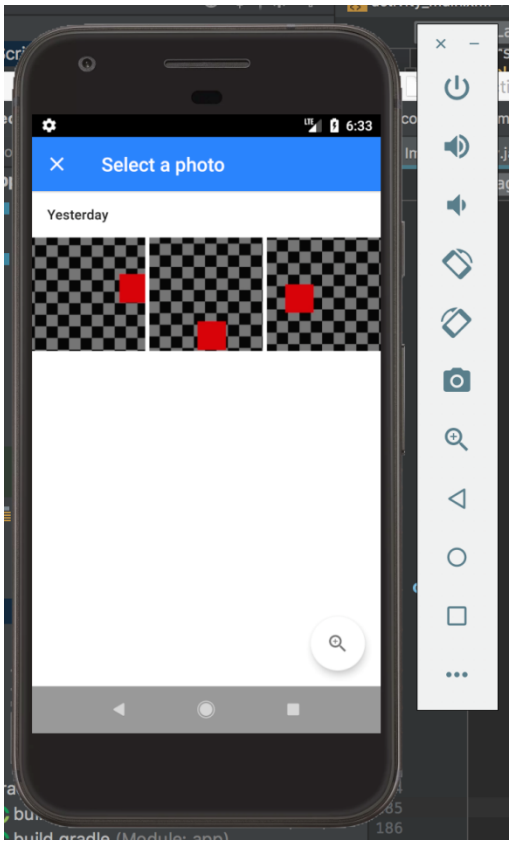
Image showing taking a picture.



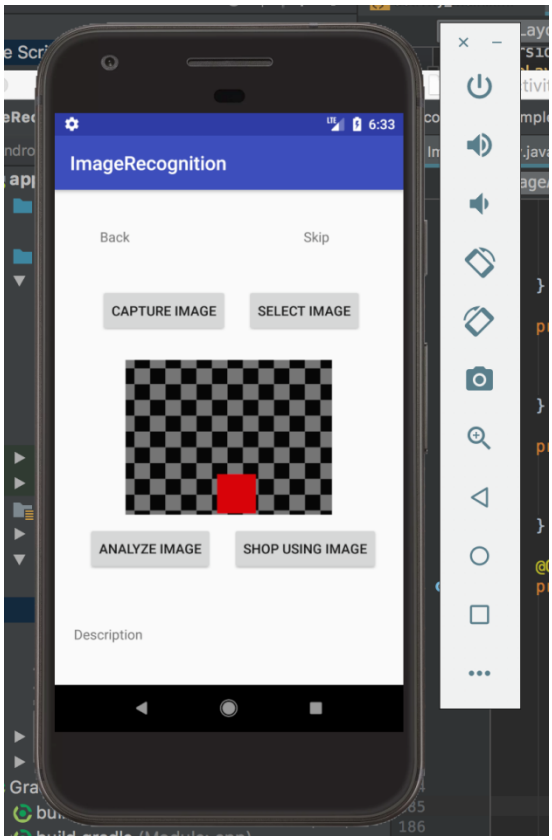
Displaying the clicked image



Selecting pre-clicked image from the gallery



Displaying the selected image



Ruthvic:

The same screen shots are taken from the android device. So please refer to the above section

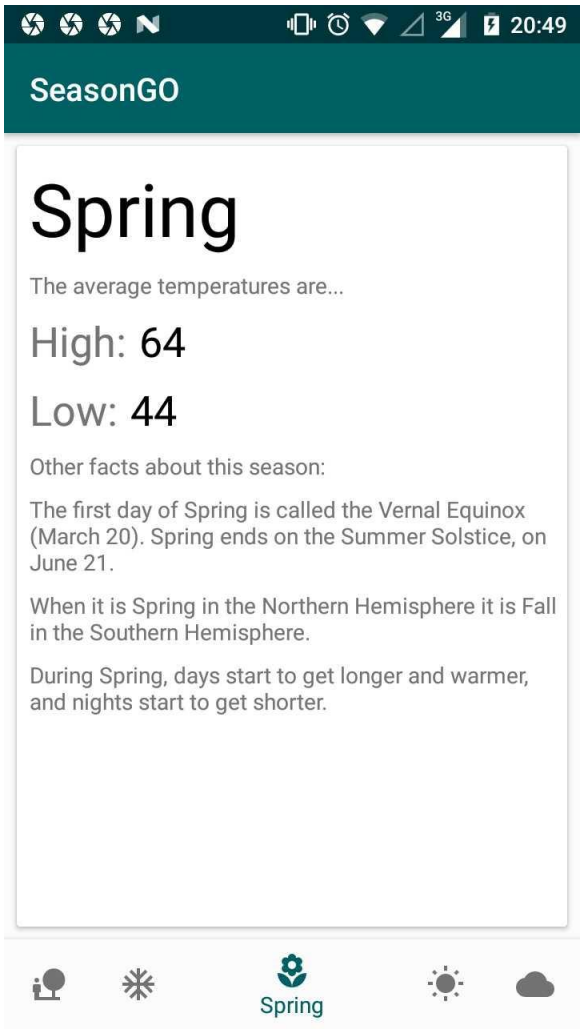
Navya:

displays different seasons

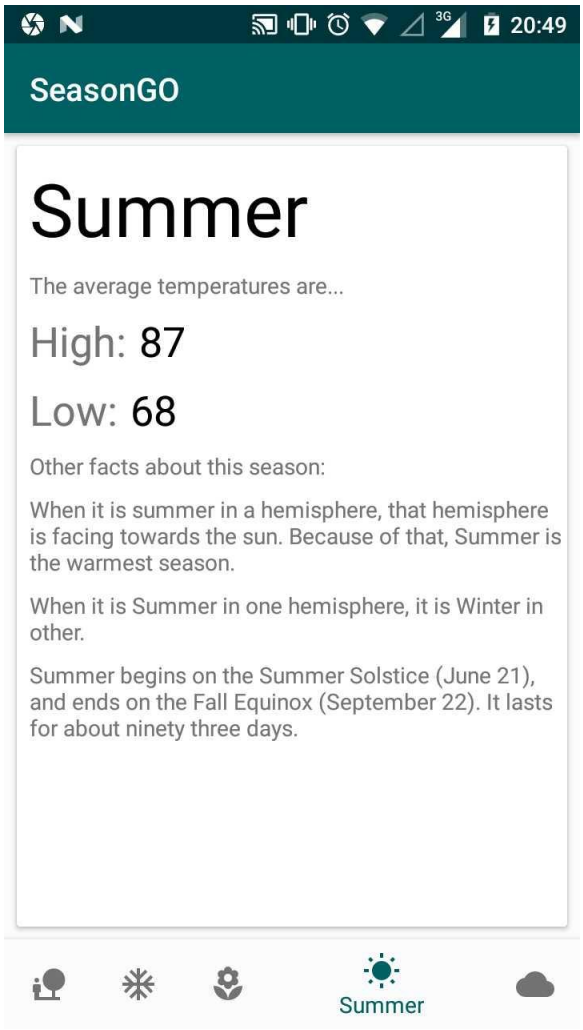
WINTER

20:49
SeasonGO
Winter
The average temperatures are...
High: 42
Low: 27
Other facts about this season:
Winter begins on Winter Solstice (December 21), and ends on Vernal Equinox (March 20). Winter is about eighty eight days.
During winter, the hemisphere that it's in faces away from
the sun. This causes that hemisphere to have
long nights, low temperatures, and short days.
Winter gets the most indirect sunlight out of all the seasons.
Winter

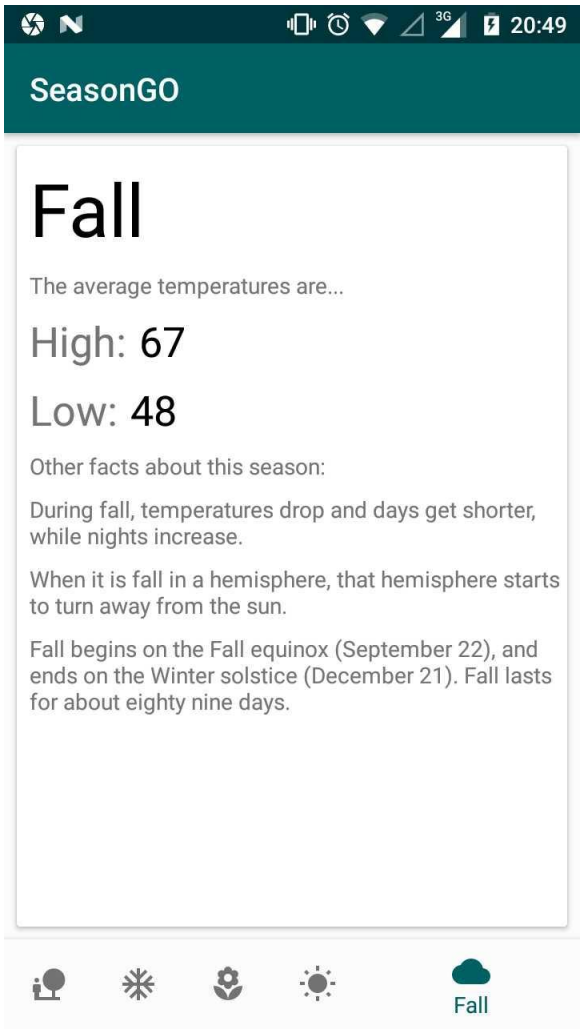
SPRING


20:49
SeasonGO
Spring
The average temperatures are...
High: 64
Low: 44
Other facts about this season:
The first day of Spring is called the Vernal Equinox (March 20). Spring ends on the Summer Solstice, on June 21.
When it is Spring in the Northern Hemisphere it is Fall in the Southern Hemisphere.
During Spring, days start to get longer and warmer, and nights start to get shorter.
Spring

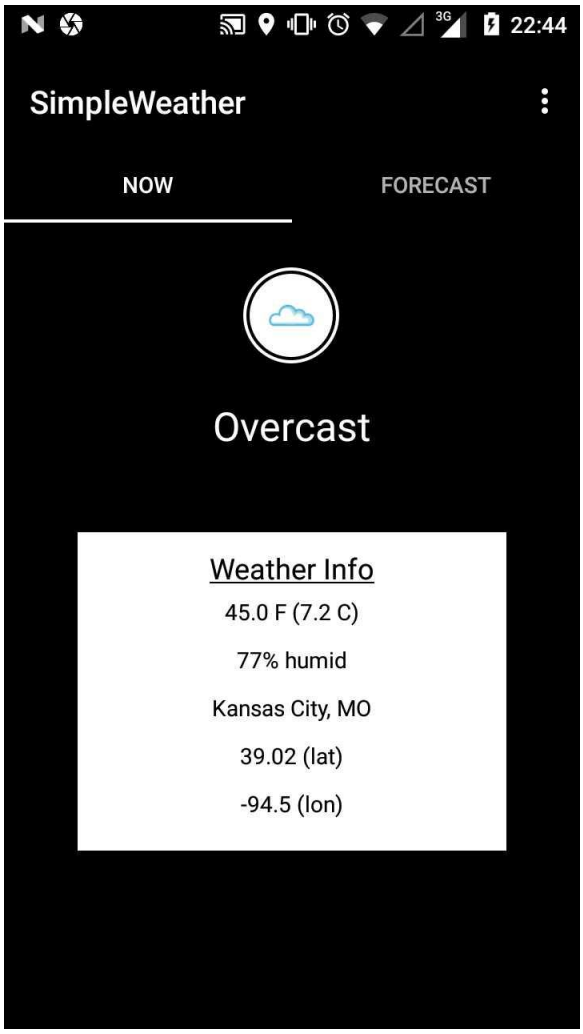
SUMMER


20:49
SeasonGO
Summer
The average temperatures are...
High: 87
Low: 68
Other facts about this season:
When it is summer in a hemisphere, that hemisphere is facing towards the sun. Because of that, Summer is the warmest season.
When it is Summer in one hemisphere, it is Winter in other.
Summer begins on the Summer Solstice (June 21), and ends on the Fall Equinox (September 22). It lasts for about ninety three days.
Summer

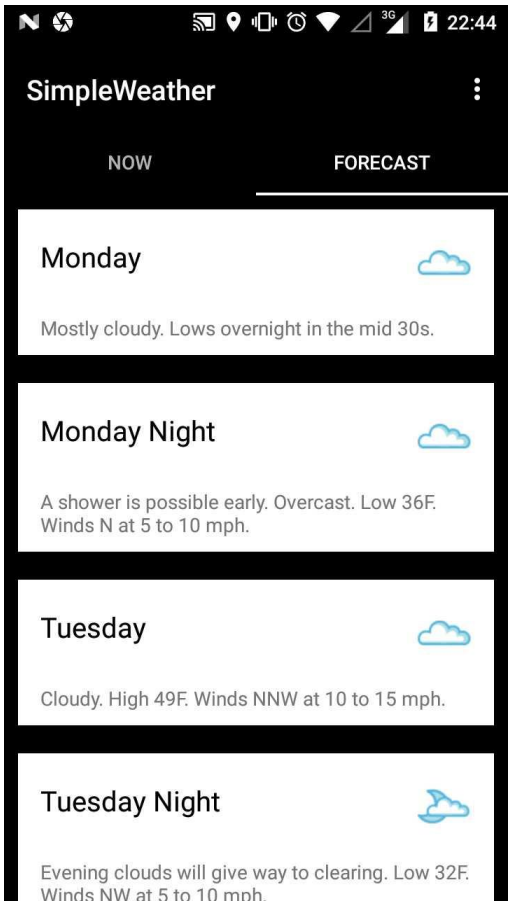
FALL

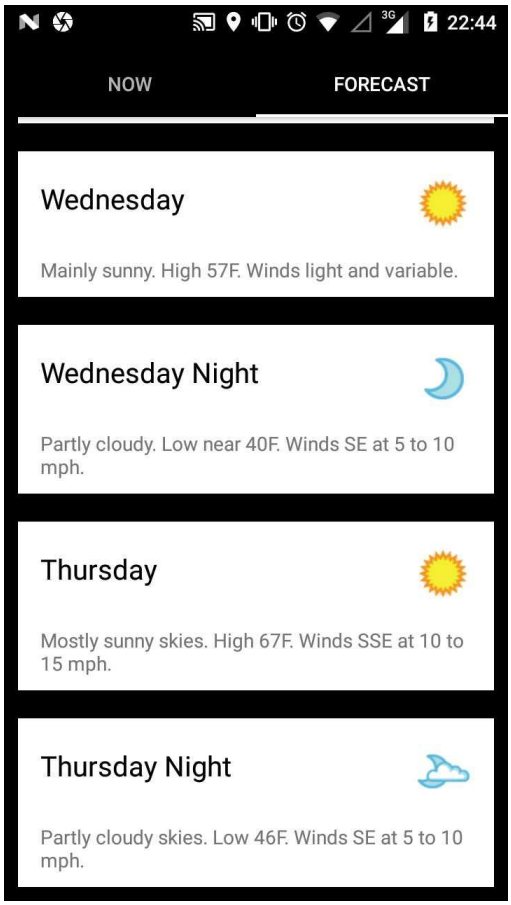


WEATHER APP THAT DISPLAYS CURRENT DAY WEATHER



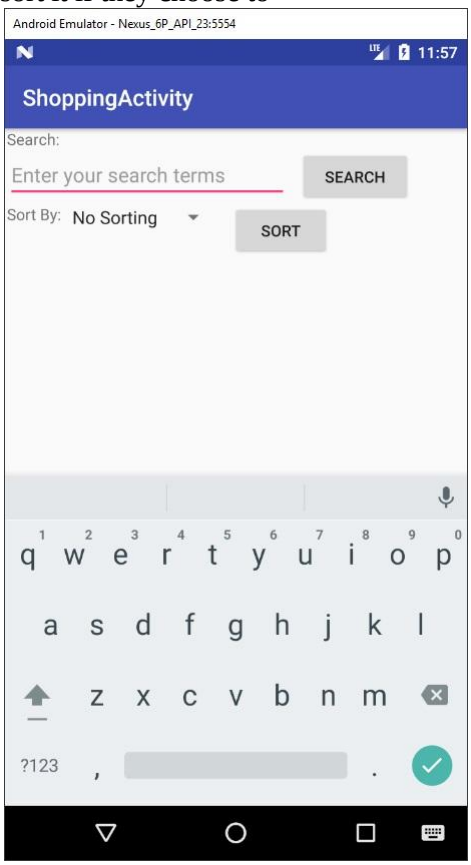
FORECAST OF WEATHER APP SHOWING THE WEATHER FOR NEXT THREE DAYS


22:44
SimpleWeather
NOW
FORECAST
Monday
Mostly cloudy. Lows overnight in the mid 30s.
Monday Night
A shower is possible early. Overcast. Low 36F. Winds N at 5 to 10 mph.
Tuesday
Cloudy. High 49F. Winds NNW at 10 to 15 mph.
Tuesday Night
Evening clouds will give way to clearing. Low 32F. Winds NW at 5 to 10 mph.

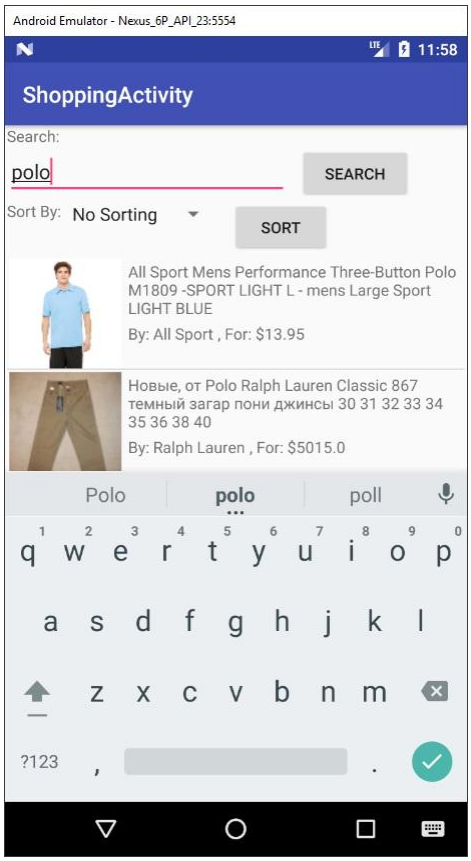

22:44
NOW
FORECAST
Wednesday
Mainly sunny. High 57F. Winds light and variable.
Wednesday Night
Partly cloudy. Low near 40F. Winds SE at 5 to 10 mph.
Thursday
Mostly sunny skies. High 67F. Winds SSE at 10 to 15 mph.
Thursday Night
Partly cloudy skies. Low 46F. Winds SE at 5 to 10 mph.

Cameron:

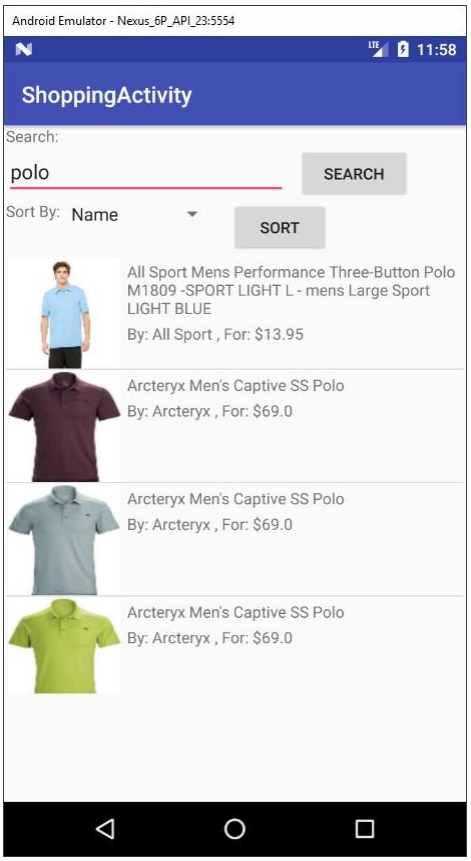
This is screenshot of the shopping activity, which will allow the user to enter in a search term, and then sort it if they choose to



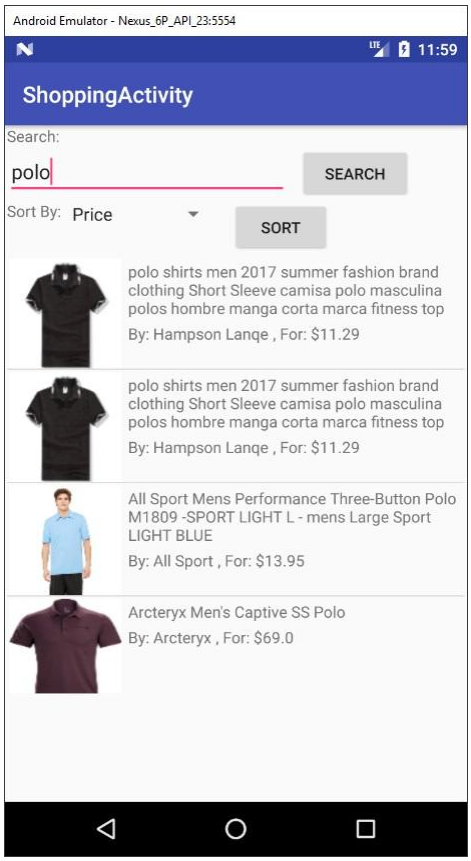
This is a screenshot taken after searching for a piece of clothing



This screenshot shows the results after sorting by name



This shows the results after sorting by price



5. Project Management

Work Completed

Sneha:

Currently the User can:

1. Move to the previous Activity
2. Skip the Image recognition activity to directly go to shopping page.
3. Take a picture for the image analysis
4. Upload the Image from the gallery for the analysis
5. Use the analysis to shop or move to shopping without the analysis. (In progress)
6. Get the analysis of the selected image.

Ruthvic:

- Create a database in Vuforia with the 3d objects
- Link the database to unity assets, and build the scene
- Integrate AR camera instead of the default camera and set the image properties
- Create a marker and add it to scene and change the camera preferences and image transform
- Build and export it to android device
- Create a simple android home page from which the user has to navigate from activity to scene

Navya:

- created weather app.
- According to the weather app it generates current weather as output.
- Basing on the weather the user will get suggestion of products which are preferable according to that weather.
- working on database that links the weather app to the shopping app.
- working on integrating weather app with the other modules of the project.
- also created simple weather app that displays the current weather conditions in that particular location, so that basing on that output user will get more appropriate suggestions.
- working on integrating season and weather app.

Cameron:

- Gathered data and created a static database
- Implemented searching for an item by text
- Implemented selecting an item
- Implemented displaying results
- Found an API for searching for clothing, that can also search for furniture

Work to be Completed

Sneha:

1. Share the image analysis data to the Shopping Activity.
2. Add User preference (if time permits).

Ruthvic:

- Navigate from unity to android
- Unity button to take a screenshot of the camera view
- store the tweets count in mongo db

Navya:

- build season app which gives the current season as output which allows user to choose products based on season and weather conditions.
- build weather app that displays current weather of that particular location.

Cameron:

- None for Increment 2

Increment 3 Report

1. Team members:

- Cameron L'Ecuyer – Class ID: 17 (Team Leader)
- Ruthvic Punyamurtula – Class ID: 30
- Navya Pillala – Class ID: 26
- Sneha Mishra – Class ID: 21
- https://github.com/camlecuyer/CS5551_Team_11_Project

2. Project Goal:

Motivation:

Develop a mobile application that has access to augmented reality in order to assist a user when they are shopping for furniture, so they can see it before they buy it.

Significance:

Our application differs from others in that the user would be able to have models generated for any piece of furniture they take a picture of, or picture they find online. Once the model is generated, the user could view how it would look in their home.

Objectives:

- Develop an augmented reality application for smart phones, in this instance Android, that can assist users shopping online for furniture
- Integrate social media platforms, like Twitter or Facebook, into the app so that the application can gather suggestions from the user's friend's recommendations, popular searches, and/or trending brands

System Features:

- Developed for smart phone operating systems, i.e. Android
- A clean and friendly user interface
- Clean and non-distracting integration of social media platforms, like Twitter
- Integration of augmented reality that allows visualization of furniture that a user may want to purchase
- Online retail searching using API services to view retailers catalogs
- Image recognition allows the system to analyze a photo to determine the furniture's shape and color

Related Work:

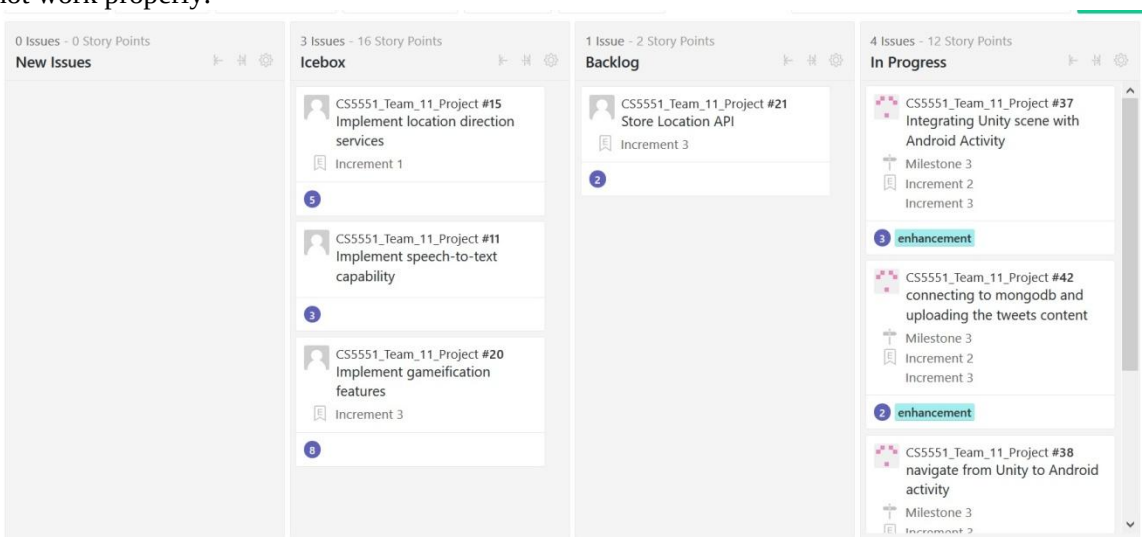
- Amazon Shopping: Using a mobile phone's camera, the app can identify what product is being viewed without needing a static picture
- IKEA Place: Allows users to view IKEA furniture with a users smart phone by leveraging augmented reality

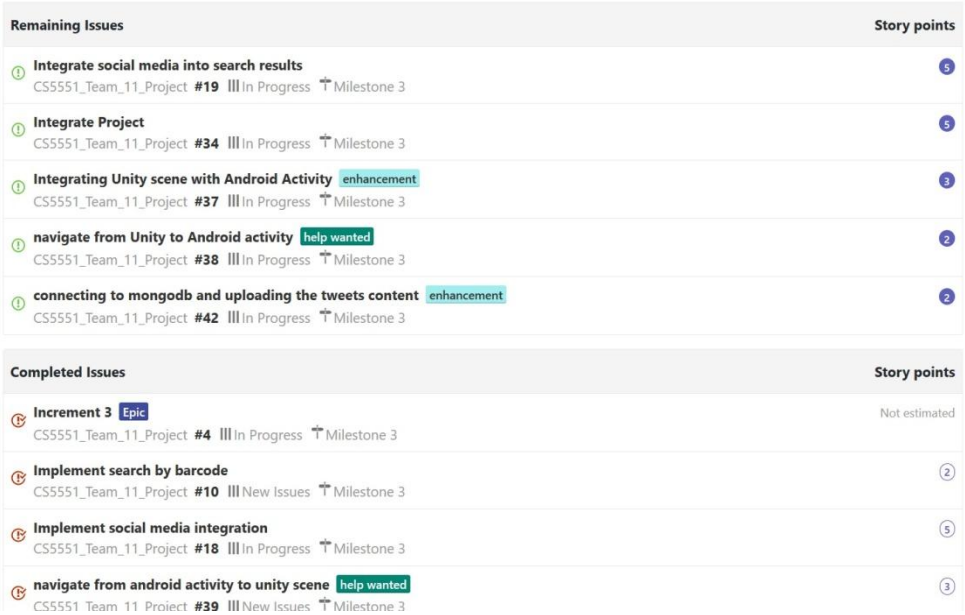
3. Project Plan

The primary objective for Increment 3 was originally to add more features to make the app more fleshed out, but due to time constraints, and issues in the previous Increment, the main task was integration and social media.

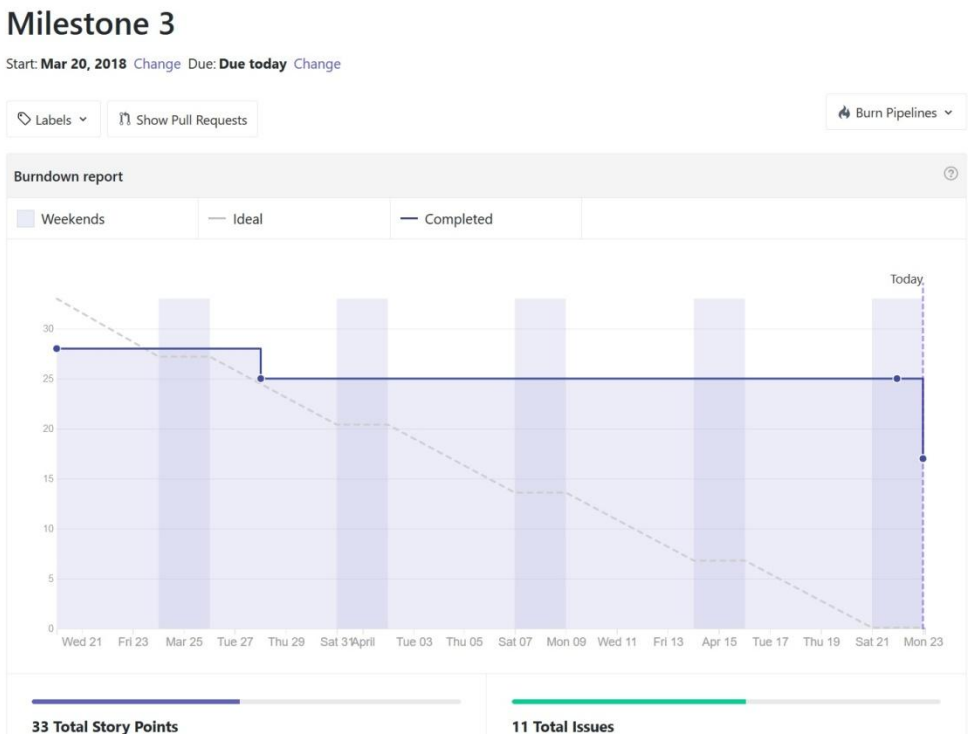
ZenHub Boards and Issues:

Several issues were removed from the boards due to them not fitting well with the current implementation of the project, others were moved to the icebox as features that we would like to implement, but do not have time for. Issues like the store location service would be useful, but due to limits of the API, would not work properly.





Burndown Chart:



4. Increment 3 Report

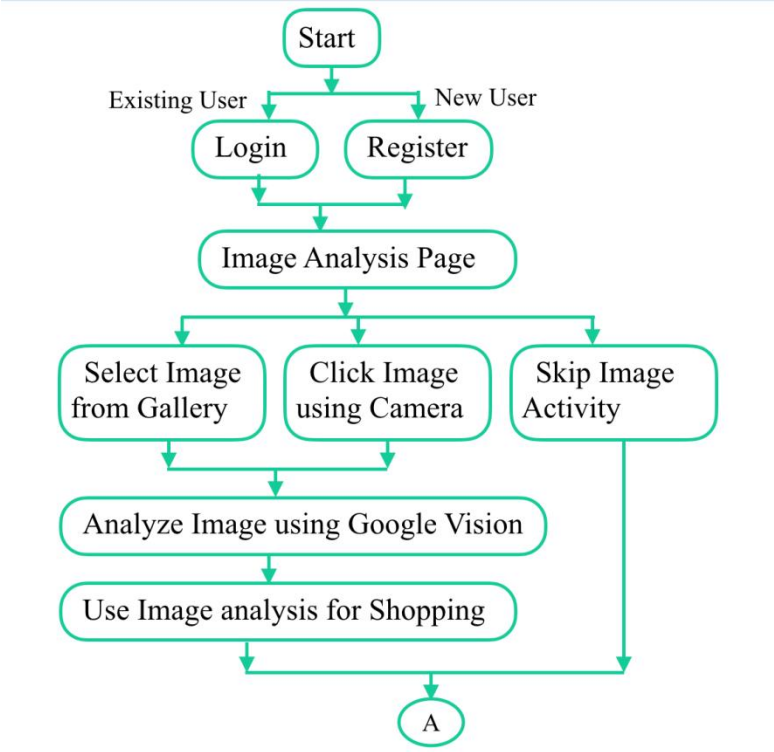
- Each team member compiled and submitted the list of what they had completed and what APIs and services they each used
- Ruthvic and Cameron created the UML diagrams and Mock ups

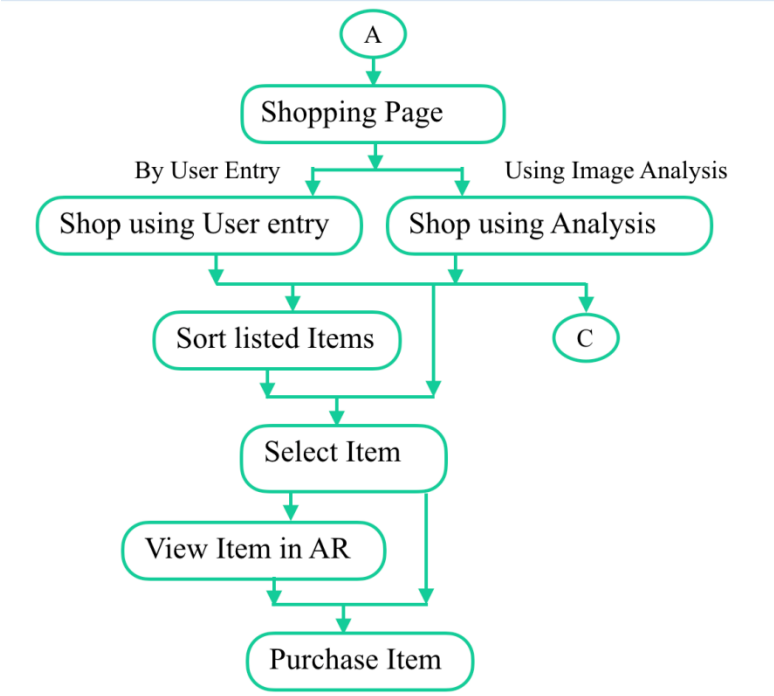
1. Existing Services/REST API

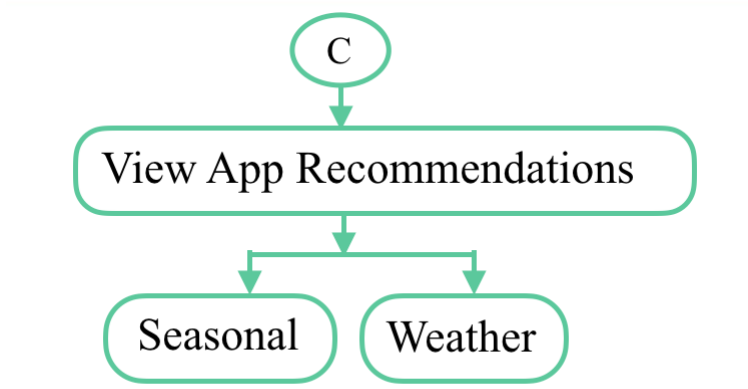
- Google Cloud Vision
- Twitter API for login
- Google Network Framework - Android's Volley Framework
- Indix API for Product Search
- Charts for Android

2. Detail Design of Features:

Flowchart of the App:


Start
Existing User
New User
Login
Register
Image Analysis Page
Select Image from Gallery
Click Image using Camera
Skip Image Activity
Analyze Image using Google Vision
Use Image analysis for Shopping
A

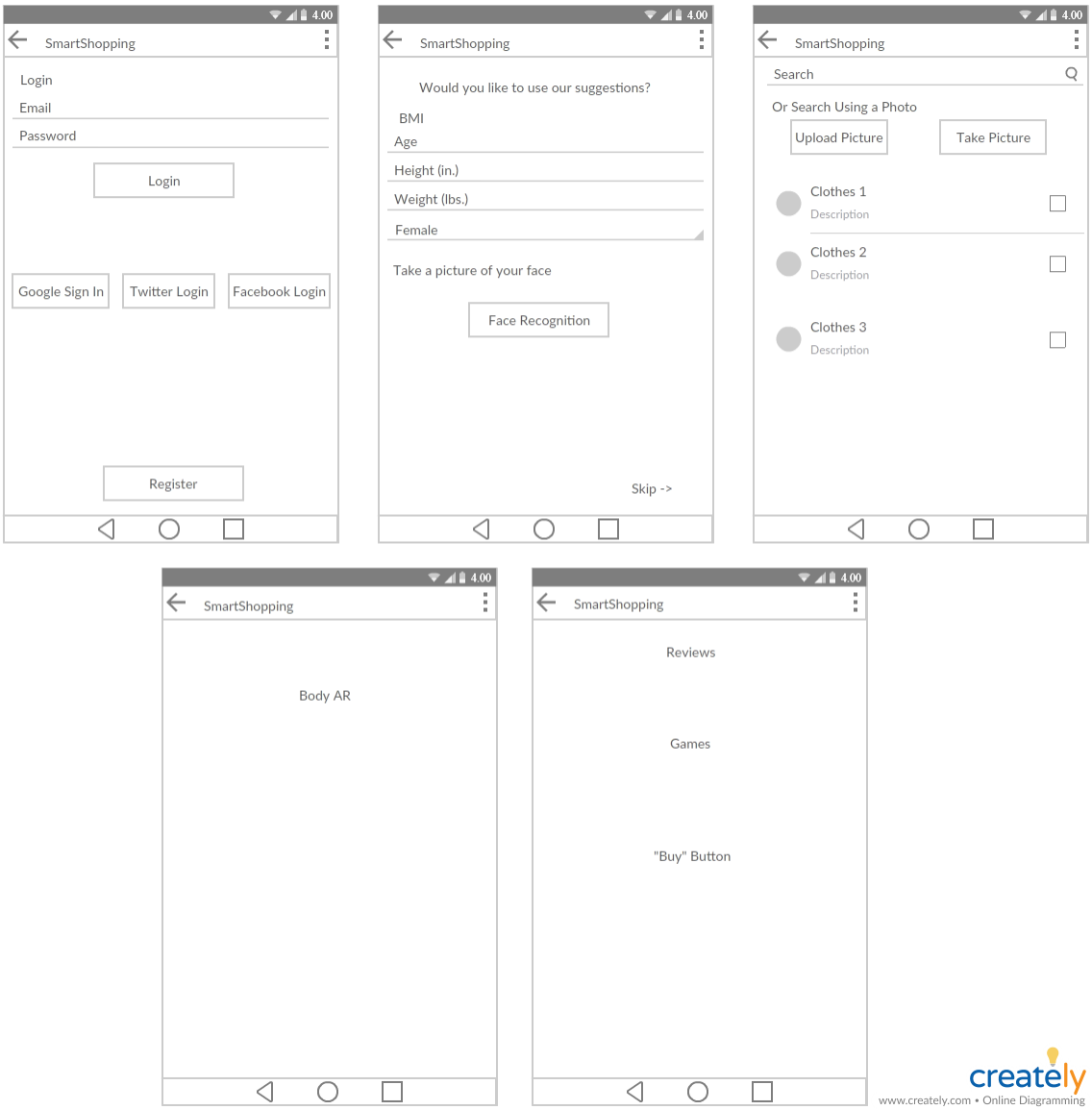

A
Shopping Page
By User Entry
Using Image Analysis
Shop using User entry
Shop using Analysis
Sort listed Items
C
Select Item
View Item in AR
Purchase Item



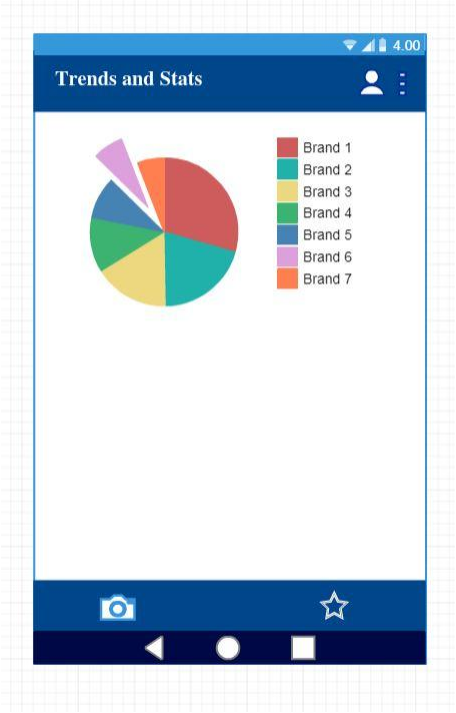
UI Mockups

These are mock ups of login, face and BMI activity (which are no longer used, and we now have a page for image recognition), search activity, body activity (which was replaced by charts), and results activity

(which is replaced with suggestions)

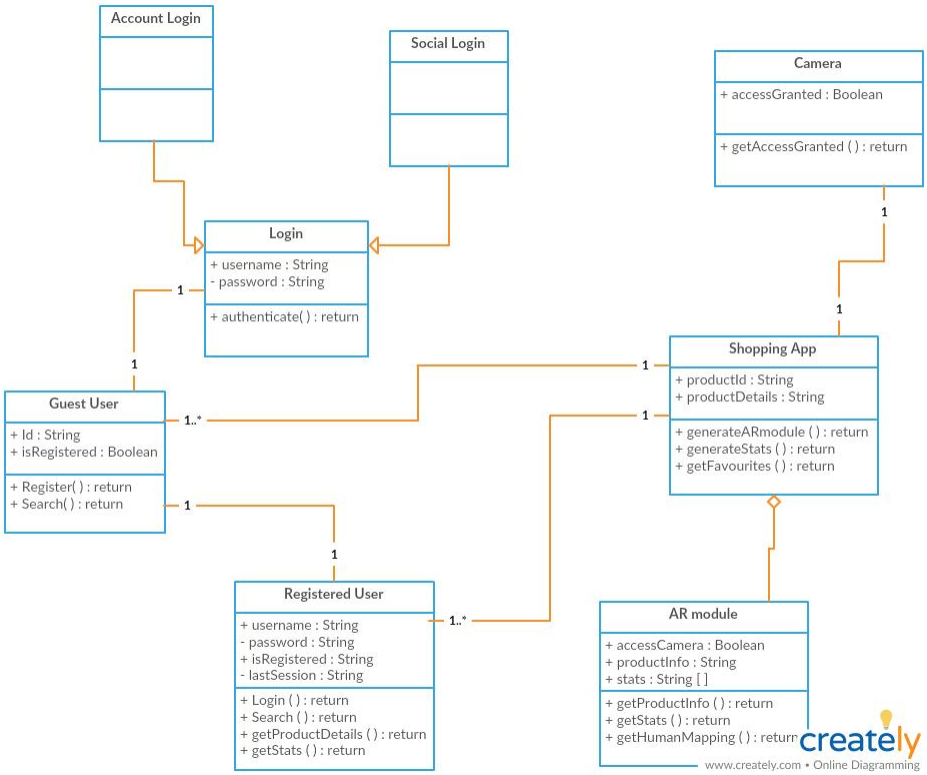


This image was created by Ruthvic during Increment 1 to display what the graphs may look like

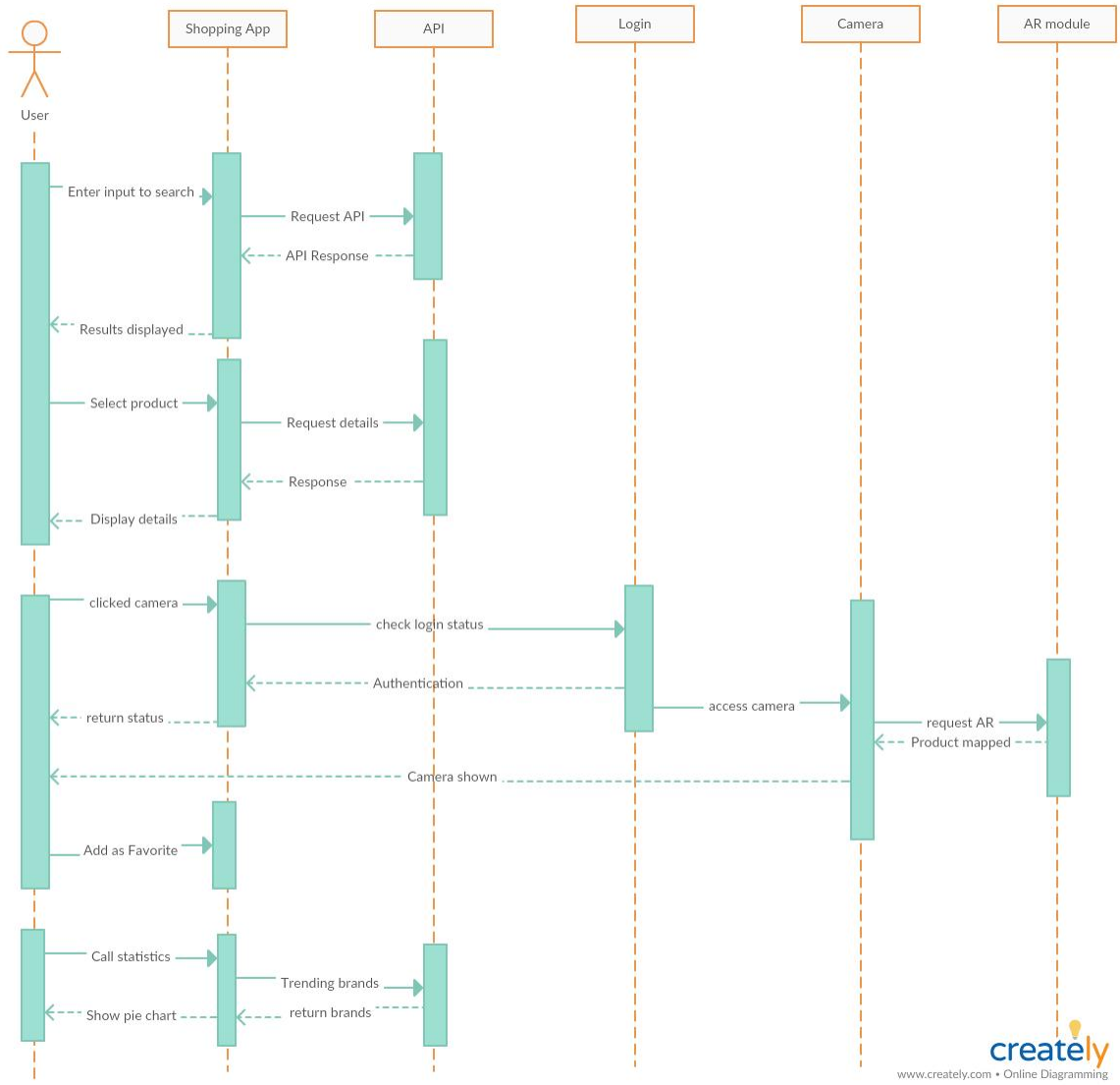


UML

Ruthvic created the UML diagrams
This image shows the class diagram, this image was created early in the project



This image shows the sequence diagram, also created early in the project



3. Testing

Sneha:

1. Tested the login/ Registration using Firebase.
2. Tested the Twitter Login.
3. Tested the Google vision after integration, everything works as expected.

Ruthvic:

- Tested the trends activity after integration and also the navigation from trends to home page
- Tested AR activity launch from Shopping Activity after integration
- Tested the unity scene with new markers and furniture images

Navya:

- Tested the Suggestions Activity and BMI activity.
- Created and tested local storage for storing the results of BMI for further use of user.
- Tested and navigated from BMI app to Google so that user can search for products or suggestions.

SOCIAL MEDIA:

- user can share his BMI results to different social media websites by using share button.

Cameron:

- Tested that the search functionality still worked when integrated into the whole app
- No other testing was done other than what was done in Increment 2

4. Implementation

- Mobile
  Mobile development and deploying is done with Android Studio, the Android application includes Google Cloud Vision for image recognition, and Google Volley for REST calls to the product search API - Indix
- Augmented Reality

1. Augmented reality is done using Unity 3D engine for game development
2. Vuforia database for the image storage and mapping on to markers

- Social Media

1. Twitter Login is implemented using twitter Oauth
2. collected tweets and stored the hashtags count in MongoDB
3. Retrieved the values from MongoDB to display the trends using a Donut chart

5. Deployment

Sneha:

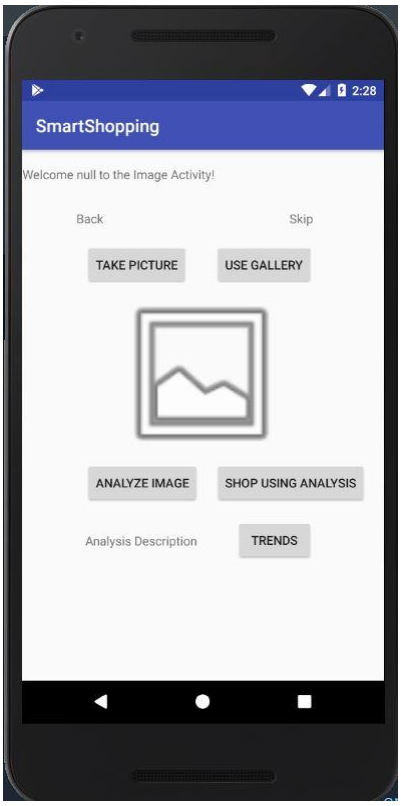
1. The Login/ Registration works after integration along with the Google vision.
2. The images are included below.

Ruthvic:

- Developed and integrated the "trends" activity
- Added MongoDB dependency jars and charts dependency to the integrated app
- Developed new unity scenes with latest markers and updated image database with furniture
- Created navigation from shopping activity to Unity Scene

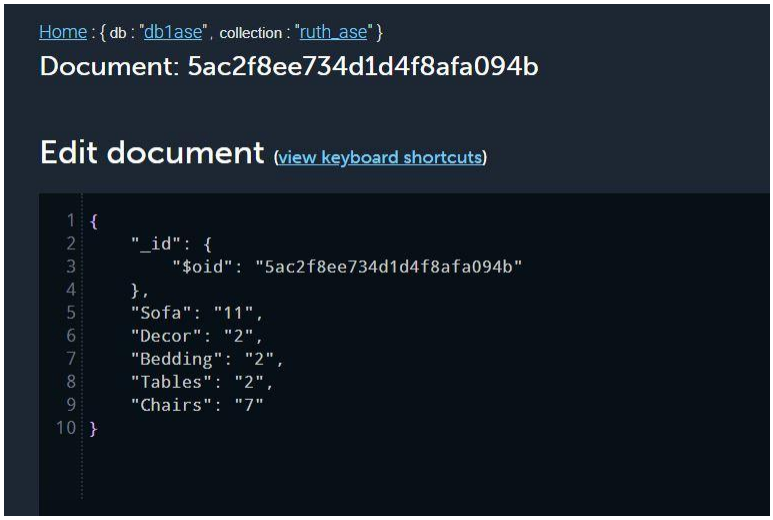
Below are the screen shots of deployment

Home page showing the button to navigate to Trends page

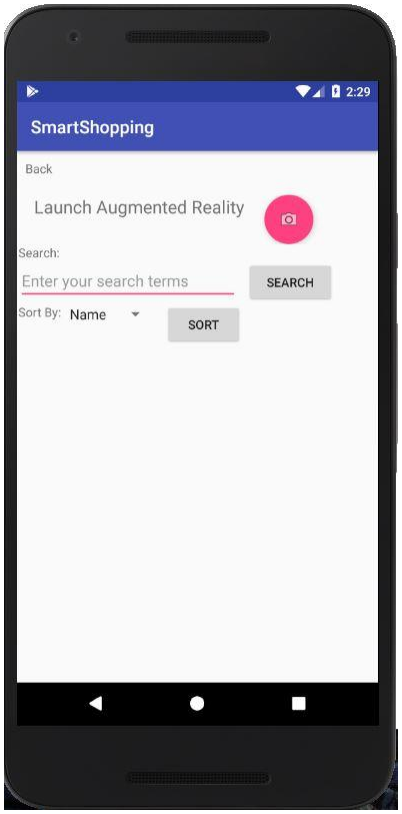


Trends activity page

Document in mmlab

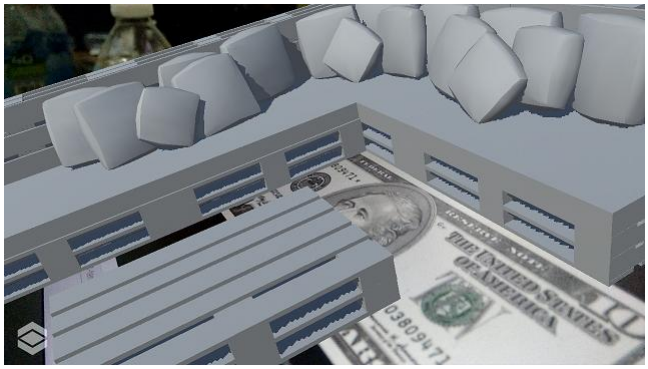


Shopping page with camera button to navigate to Unity scene



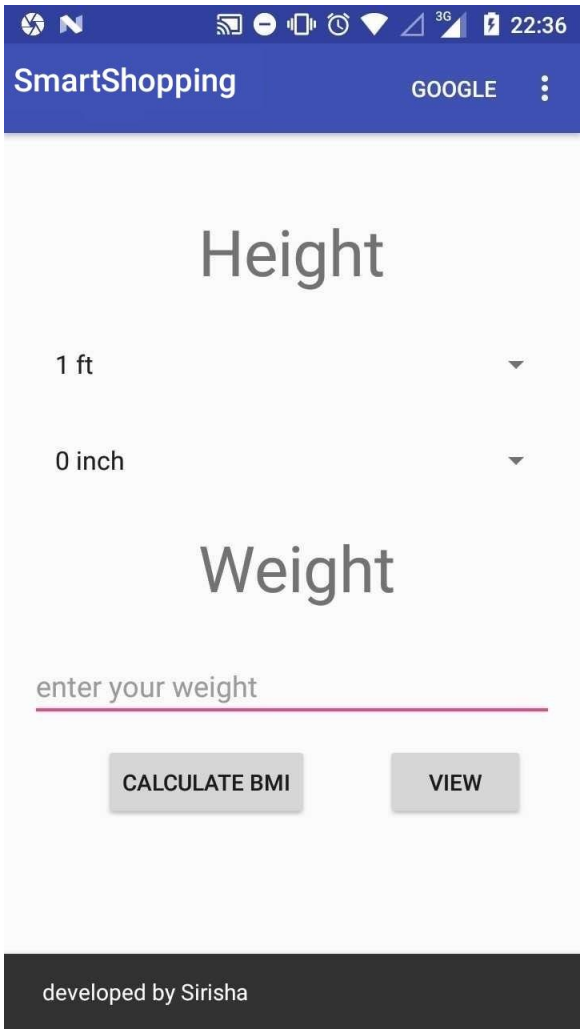
Once unity scene is launched, we test the app with different markers whose output is shown below

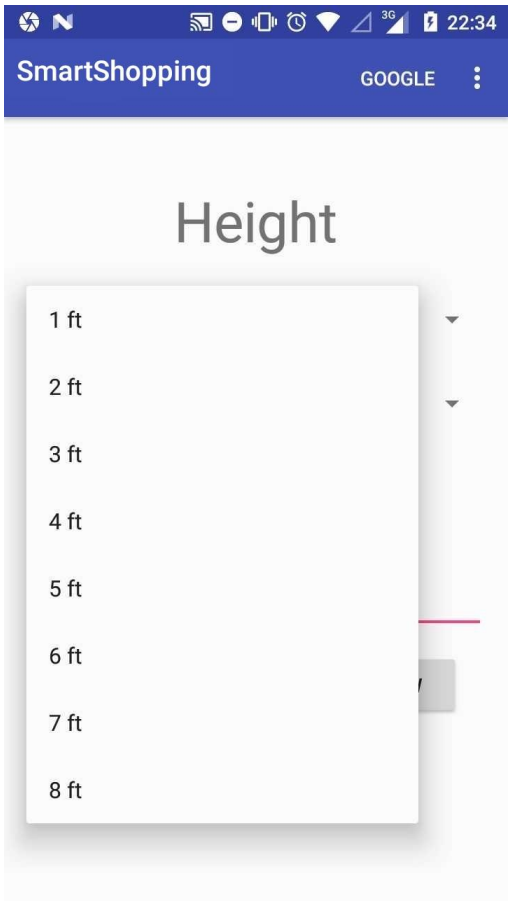
Another marker showing sofa 2 in 3D

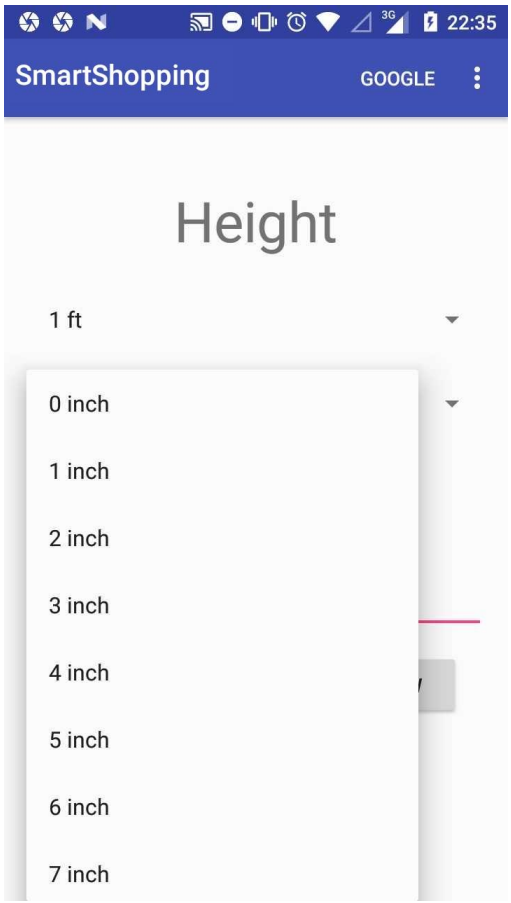


Navya:

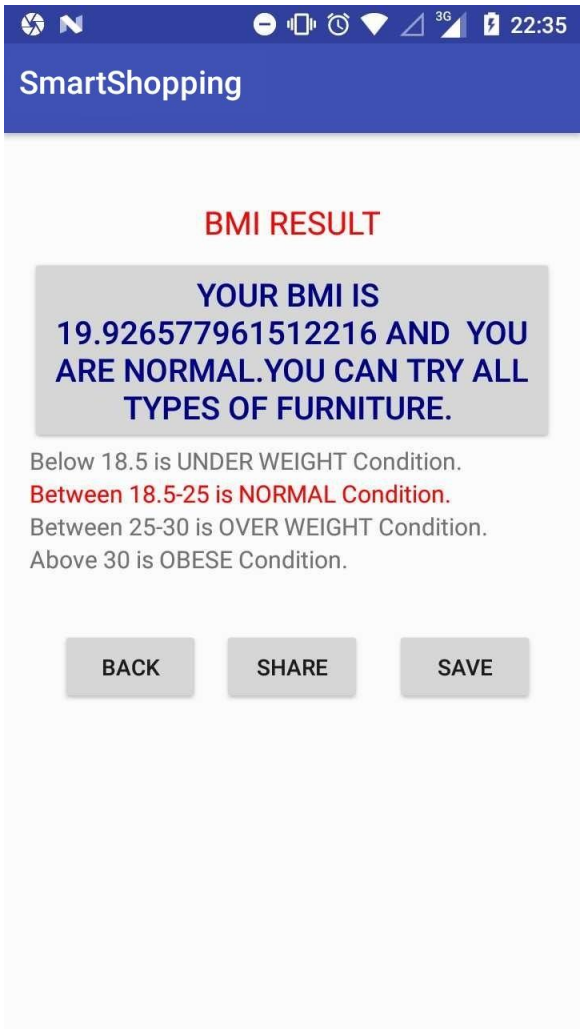
suggestions button will take the user to recommendation section.


22:36
SmartShopping
GOOGLE
Height
1 ft
0 inch
Weight
enter your weight
CALCULATE BMI
VIEW
developed by Sirisha

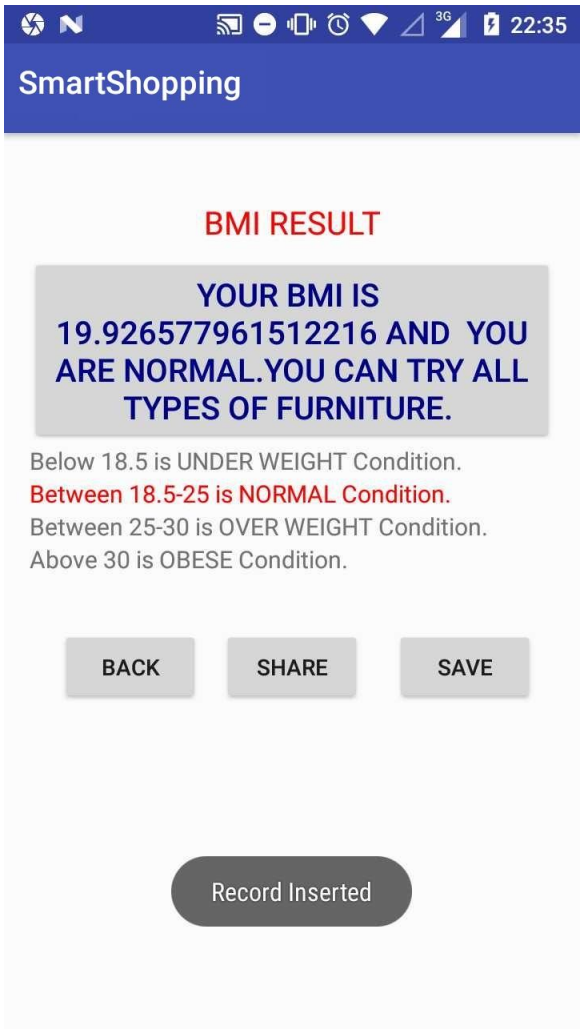




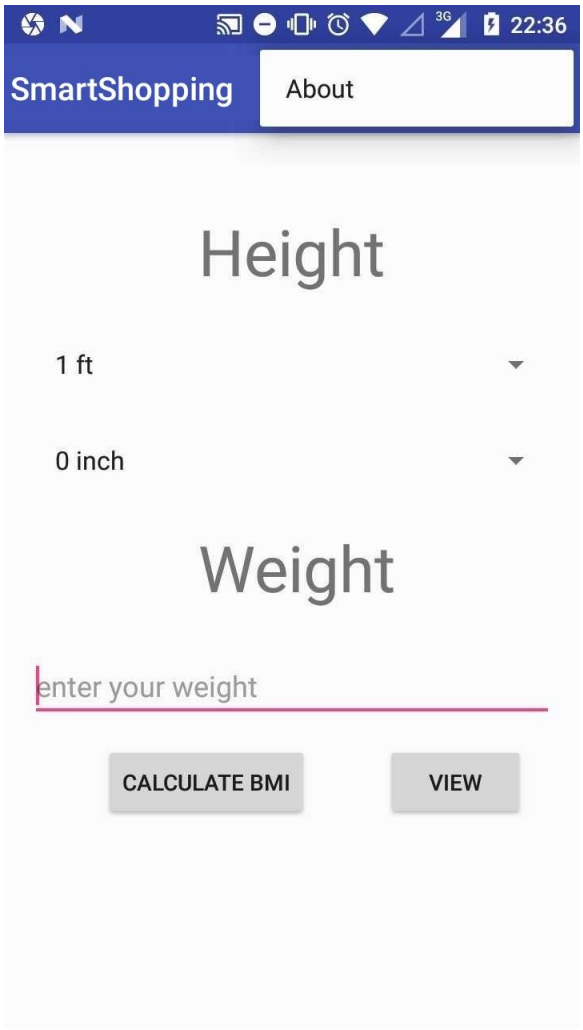
on clicking the calculate BMI button the BMI of user and suggestions will be displayed.



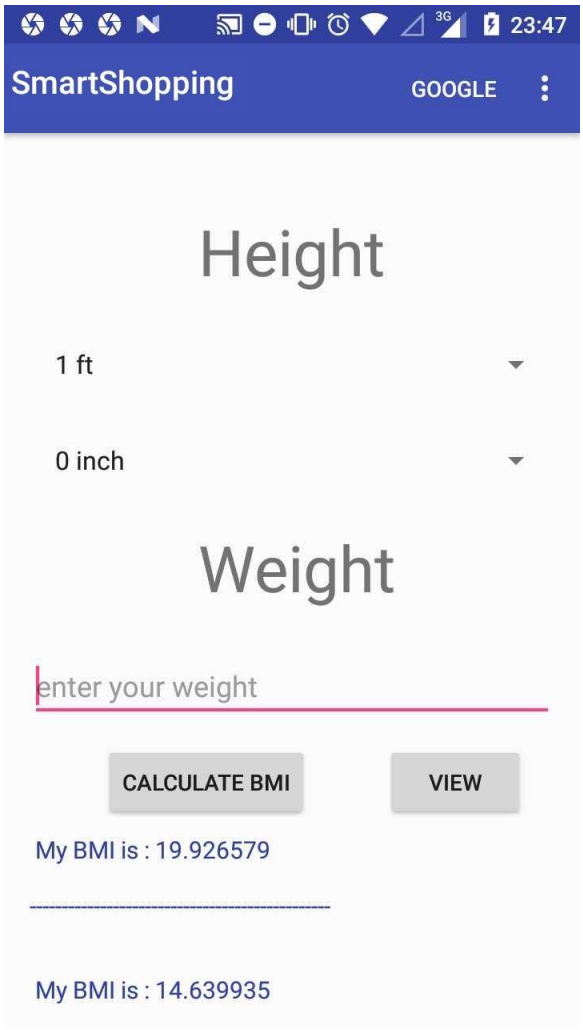
on clicking the save button it saves the result.



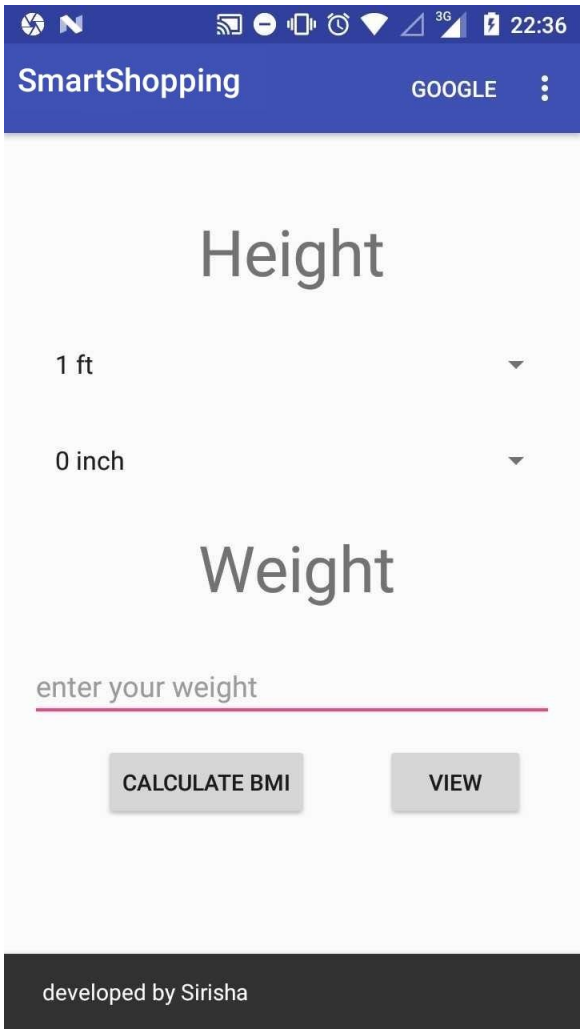
on clicking back button it navigates to home screen



the View button displays all the saved results of user



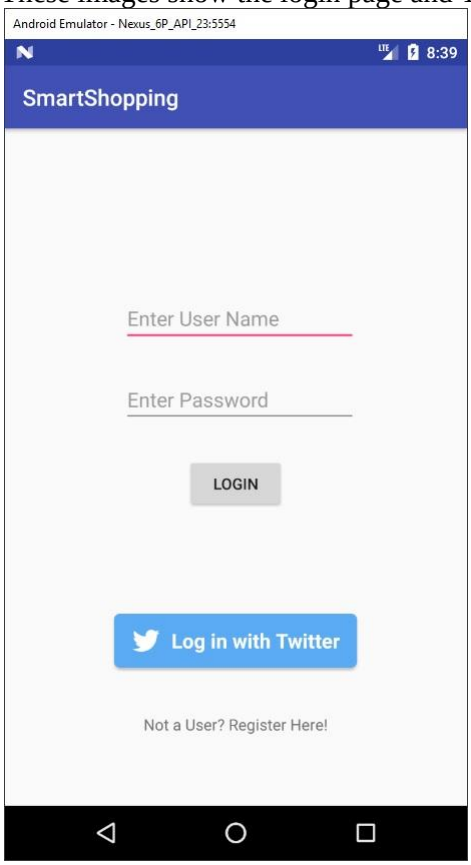
Google button navigates the user to Google so that he can search on web for more products.

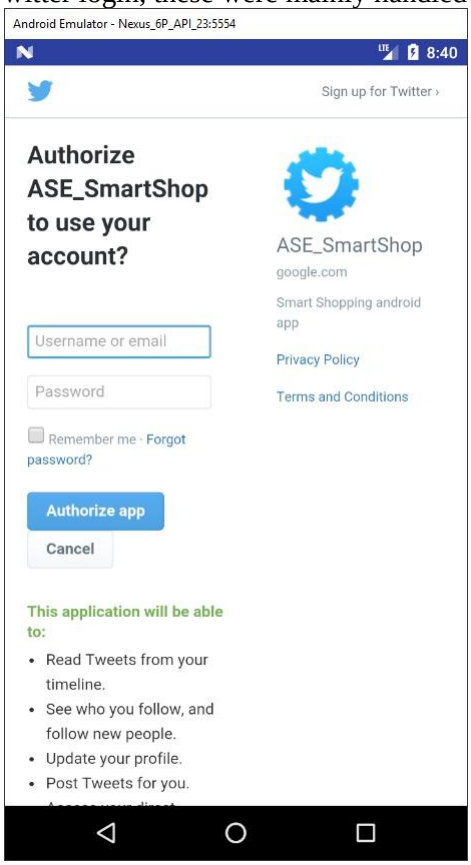


Cameron:

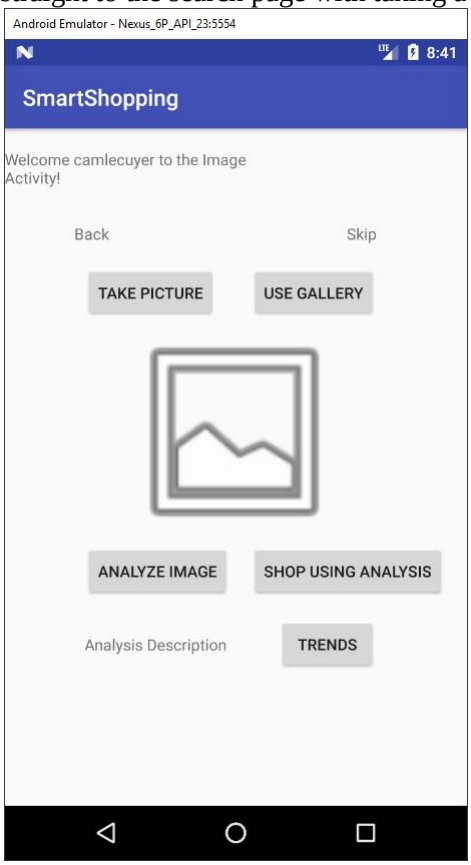
The search portion of the app with integration into the whole app

These images show the login page and Twitter login, these were mainly handled by Sneha

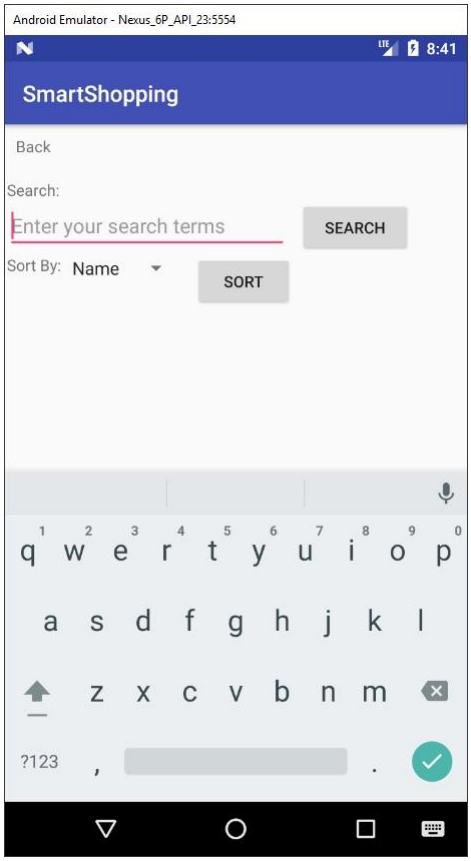


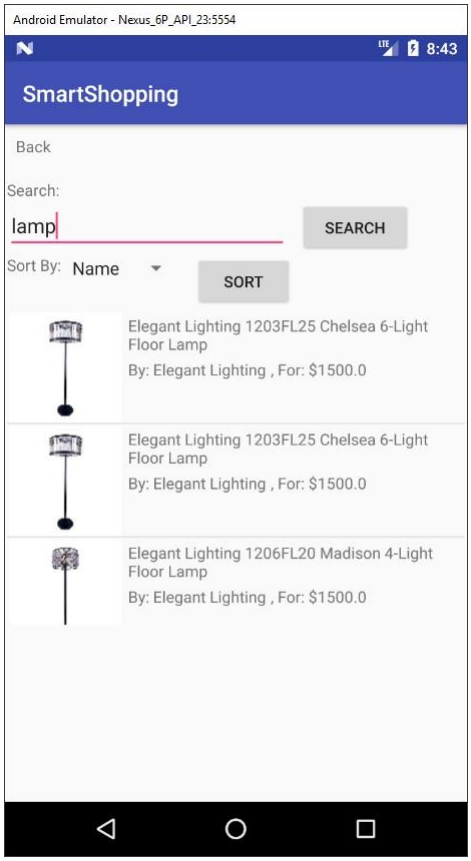


Next is the image recognition page also handled by Sneha, the Skip button allows the user to move straight to the search page with taking a picture

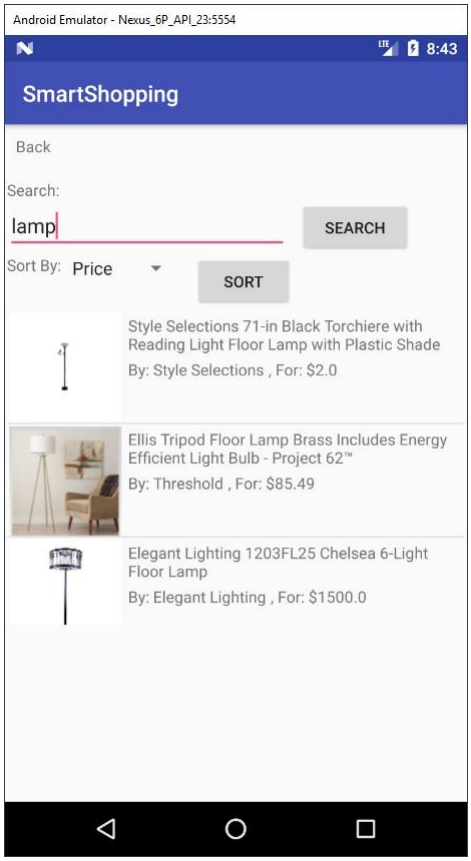


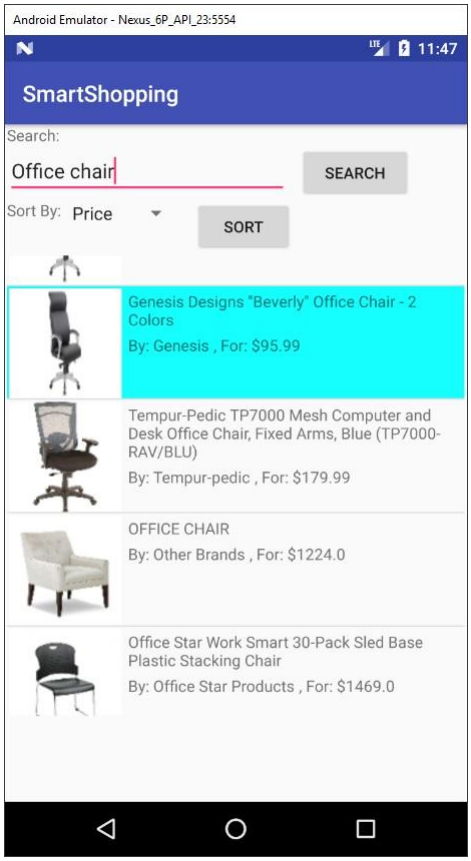
The search page:
First is the page when first navigated to, the next image shows what the page looks like after searching for an item


Android Emulator - Nexus_6P_API_23:5554
8:41
SmartShopping
Back
Search:
Enter your search terms
SEARCH
Sort By:
Name
SORT


Android Emulator - Nexus_6P_API_23:5554
8:43
SmartShopping
Back
Search:
lamp
SEARCH
Sort By:
Name
SORT
Elegant Lighting 1203FL25 Chelsea 6-Light Floor Lamp
By: Elegant Lighting , For: $1500.0
Elegant Lighting 1203FL25 Chelsea 6-Light Floor Lamp
By: Elegant Lighting , For: $1500.0
Elegant Lighting 1206FL20 Madison 4-Light Floor Lamp
By: Elegant Lighting , For: $1500.0

This image shows the sorting the results sorted by price instead of by name, and the one below shows what it looks like when the user selects one of the items





6. Project Management

Work Completed

Sneha:

1. Successful integration with other functionalities.
2. Minor bug fixes, which occurred due to integration.

Ruthvic:

- Integrated trends activity to main app
- Added navigation from home to trends and vice-versa
- Added navigation from shopping to AR activity
- Resolved a bug with mongo DB data extraction and added mmlab latest dependency
- Added multiple markers to single unity scene

Navya:

Cameron:

- Successfully integrated searching into the main app

- Fixed a bug with searching, where the app would crash if an empty image path was passed into the image loader

Work to be Completed

Sneha:

1. Making the UI more appealing.
2. Image analysis description handling.

Ruthvic:

- Improving UI appearance and navigation handling
- Creating a button in unity that could navigate back to android

Navya:

Cameron:

- Adding store location with Google maps if, time allows

Bibliography:

Project Plan and Proposal Bibliography:

- TheMine app: https://itunes.apple.com/us/app/envisioned/id1293488677?ls=1&mt=8
- IKEA place app: https://itunes.apple.com/us/app/ikea-place/id1279244498?mt=8
- Amazon Shopping app: https://play.google.com/store/apps/details?id=com.amazon.mShop.android.shopping&hl=en
- IBM Watson: https://www.ibm.com/watson/products-services/
- Vuforia: https://library.vuforia.com/api
- ARToolKit: https://www.artoolkit.org/documentation/
- MaxST: https://developer.maxst.com/
- Amazon API: https://docs.aws.amazon.com/AWSECommerceService/latest/DG/Welcome.html
- Target API: https://developer.target.com/
- Walmart API: https://developer.walmartlabs.com/
- Google Speech-to-Text: https://weston.ruter.net/2009/12/12/google-tts/

Increment 1 Bibliography:

- Amazon Shopping app: https://play.google.com/store/apps/details?id=com.amazon.mShop.android.shopping&hl=en
- IBM Watson: https://www.ibm.com/watson/products-services/
- Vuforia: https://library.vuforia.com/api
- ARToolKit: https://www.artoolkit.org/documentation/
- MaxST: https://developer.maxst.com/
- Amazon API: https://docs.aws.amazon.com/AWSECommerceService/latest/DG/Welcome.html
- Target API: https://developer.target.com/
- Walmart API: https://developer.walmartlabs.com/

- Google Speech-to-Text: https://weston.ruter.net/2009/12/12/google-tts/

Increment 2 Bibliography

- Amazon Shopping app: https://play.google.com/store/apps/details?id=com.amazon.mShop.android.shopping&hl=en
- IKEA Place: https://itunes.apple.com/us/app/ikea-place/id1279244498?mt=8
- IBM Watson: https://www.ibm.com/watson/products-services/
- Vuforia: https://library.vuforia.com/api
- ARToolKit: https://www.artoolkit.org/documentation/
- MaxST: https://developer.maxst.com/
- Amazon API: https://docs.aws.amazon.com/AWSECommerceService/latest/DG/Welcome.html
- Target API: https://developer.target.com/
- Walmart API: https://developer.walmartlabs.com/
- Google Speech-to-Text: https://weston.ruter.net/2009/12/12/google-tts/
- Indix API: https://developer.indix.com/docs
- Google Volley: https://developer.android.com/training/volley/index.html
- Microsoft Azure: https://azure.microsoft.com/en-us/services/cognitive-services/computer-vision/
- Unity tutorials : https://docs.unity3d.com
- Face++ https://www.faceplusplus.com/body-outlining/#demo
- BetaFace API https://www.betafaceapi.com/wpa/
- BMI Calculator with JavaScript https://stackoverflow.com/questions/21698044/basic-bmi-calculator-html-javascript

Increment 3 Bibliography

- Amazon Shopping app: https://play.google.com/store/apps/details?id=com.amazon.mShop.android.shopping&hl=en
- IKEA Place: https://itunes.apple.com/us/app/ikea-place/id1279244498?mt=8
- IBM Watson: https://www.ibm.com/watson/products-services/
- Vuforia: https://library.vuforia.com/api
- ARToolKit: https://www.artoolkit.org/documentation/
- MaxST: https://developer.maxst.com/
- Google Speech-to-Text: https://weston.ruter.net/2009/12/12/google-tts/
- Indix API: https://developer.indix.com/docs
- Google Volley: https://developer.android.com/training/volley/index.html
- Microsoft Azure: https://azure.microsoft.com/en-us/services/cognitive-services/computer-vision/
- Unity tutorials: https://docs.unity3d.com
- Charts for Android: https://github.com/PhilJay/MPAndroidChart
- https://stackoverflow.com/questions/45500934/error-fix-the-version-conflict-google-services-plugin
- http://androidkt.com/vision-api/
- https://apps.twitter.com/
- https://console.cloud.google.com/apis/library/vision.googleapis.com/?id=957e5f12-b03d-4788-9f56
- https://developers.google.com/android/guides/client-auth?authuser=0
- https://cloud.google.com/vision/docs/