Description

Intended User

Features

User Interface Mocks
- Screen 1
- Screen 2

Key Considerations
- How will your app handle data persistence?
- Describe any edge or corner cases in the UX.
- Describe any libraries you'll be using and share your reasoning for including them.
- Describe how you will implement Google Play Services or other external services.

Next Steps: Required Tasks
- Task 1: Project Setup
- Task 2: Implement UI for Each Activity and Fragment
- Task 3: Add database and Load data
- Task 4: Retrieve and Parse Recommended products Json data of Amazon
- Task 5: Setting menu and set recommended products list

GitHub Username: Julie Kim (juliekim8312@gmail.com)

Purchase Notification App

Description

Purchase Notification App helps users purchase products that require regular purchases by notification.
Also this app recommends good products at low prices to users. Users can search for items and compare the prices to purchase on each regional amazon site.

Intended User

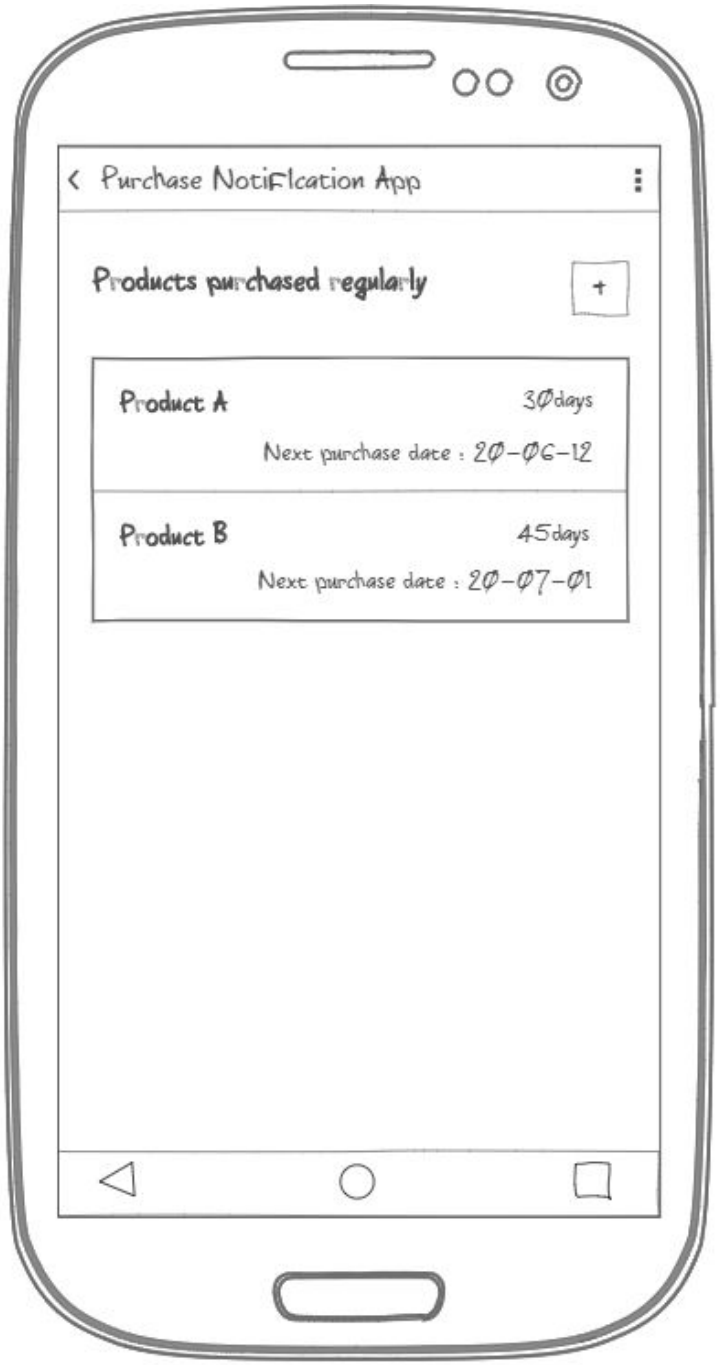
Anyone who wants to buy a product regularly.
It's even better for users who buy on Amazon.

Features

- Manage products purchased regularly with products and purchase terms
- Notify the user when the purchase time comes
- Recommend products from Amazon using product information such as name, price, image, rating.
- Link to the product page of Amazon

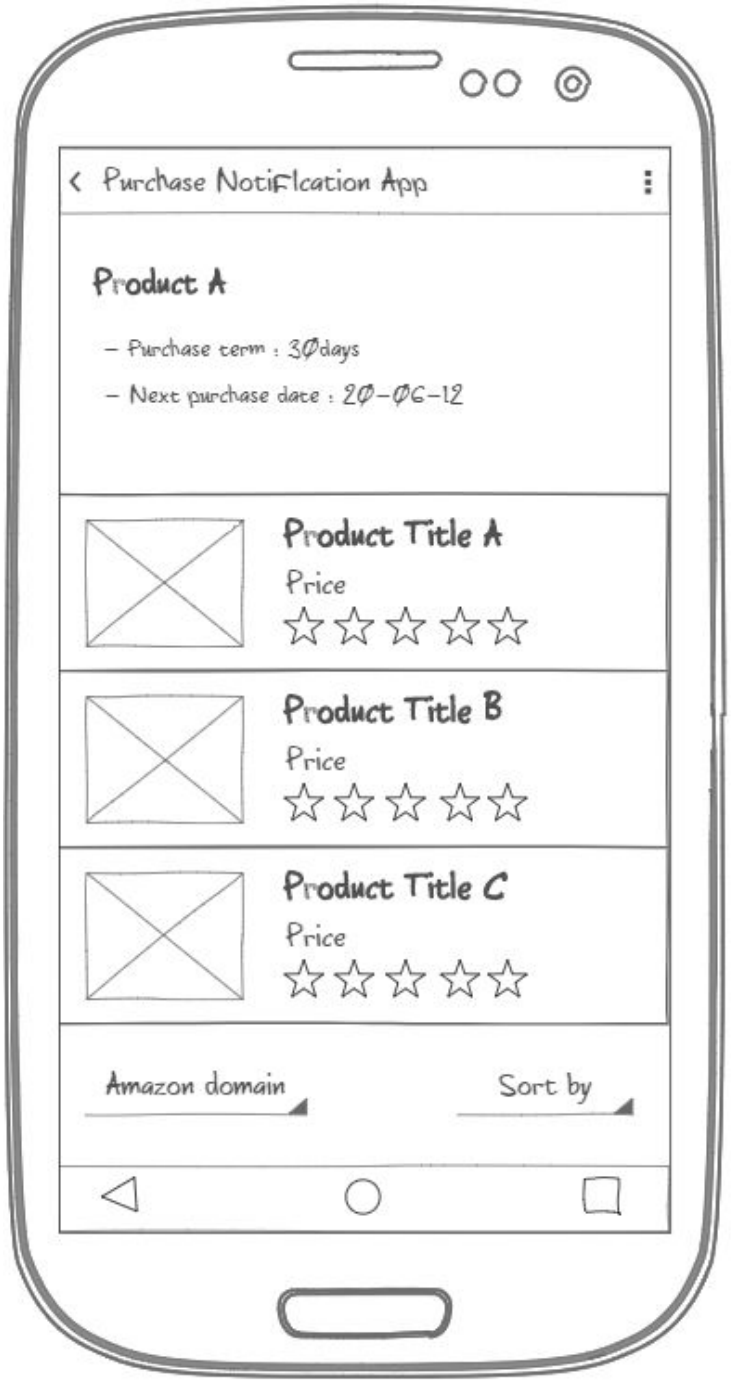
User Interface Mocks

Screen 1



Users can add product items to purchase regularly by [+] button.
There is a list of products purchased regularly with its purchase term and next purchase date.

Screen 2



When users select the product to purchase, the recommended item list is showing with each item's price, image and rating.

You can change the Amazon domain by domain menu, and change sorting by sort by menu. When you select each product item, link to the Amazon product page to purchase it.

Key Considerations

How will your app handle data persistence?

Handle products data using a Content Provider.

Describe any edge or corner cases in the UX.

If the loading of the product list from Amazon doesn't finish, users can change the Amazon domain or sort by menu. To prepare for such a case, disable menu changes while loading.

Describe any libraries you'll be using and share your reasoning for including them.

Picasso library to handle image loading and caching.
: 'com.squareup.picasso:picasso:2.71828'

Retrofit library for Http connection and Json parsing.
: 'com.squareup.retrofit2:retrofit:2.5.0', ‘
: 'com.squareup.retrofit2:converter-gson:2.5.0'

Support library for this app
: 'androidx.appcompat:appcompat:1.1.0'
: 'androidx.constraintlayout:constraintlayout:1.1.3'
: 'androidx.recyclerview:recyclerview:1.1.0'

Describe how you will implement Google Play Services or other external services.

For notification of purchase products : AlarmManager - start ALARM_SERVICE through pending intent, user RTC_WAKEUP for alarm type

* Programming language : Java Programming Language (JavaVersion.VERSION_1_8)
* Android Studio version : 3.6.3
* gradle version : 3.6.3
* min sdk version : 16
* target sdk version : 29

Next Steps: Required Tasks

This is the section where you can take the main features of your app (declared above) and break them down into tangible technical tasks that you can complete one at a time until you have a finished app.

Task 1: Project Setup

- Configure libraries - Picasso, Retrofit
- Enable Data binding for layout binding
- Add Room dependencies

Task 2: Implement UI for Each Activity and Fragment

- Build UI for MainActivity
    - Label text
    - Add button
    - Products list using RecyclerView : each item includes product name, purchase term, and next purchase date

- Build UI for RecommendActicity
    - Product information text : product name, purchase term, and next purchase date
    - Recommending Products list using RecyclerView : each item includes product name, image, price, and rating

Task 3: Add database and Load data

- Add products database using ROOM
    - product name, purchase term, and next purchase date

- Create a DAO
    - Insert, Update, Delete, Query(All, one)

- Load Products list from database

- Open RecommendActivity when selecting a product item.

Task 4: Retrieve and Parse Recommended products Json data of Amazon

- Compose request url using Rainforest Api
    - API document site : https://rainforestapi.com/docs/product-data-api/parameters/search
    - Get API Key for request url
    - Request url example : https://api.rainforestapi.com/request?api_key=1D13FD0ACB744E669406718FAA260F5C&type=search&customer_location=kr&amazon_domain=amazon.com&search_term=memory+cards&sort_by=price_low_to_high
      * search_term=[product name]

- Connect Http and retrieve Json using Retrofit

Task 5: Setting menu and set recommended products list

- Change request url with amazon domain setting menu and sort by setting menu
    - amazon_domain=[xxx]
    - sort_by=[yyy]

- Set recommended products data by result data

- Link to Amazon product page when selecting a product item.