GitHub Username: muhammadabdulsalam

Law Assistant

Description

Law Assistant helps Lawyers/Solicitors to build case files. Lawyers need to do allot of research in order to build a case they need to find old cases and their rulings to pursue the case further. CaseLaws helps Lawyers to see the previous cases of same genre and then build the case accordingly. For this search Lawyers need to go through a long list of books and digest to find out previous cases regarding the issue.
Law assistant will help lawyers to search CaseLaws by “KEYWORDS” and can get them in range of particular dates. In real life scenario if a lawyer needs to find out old case laws from year 2000 to 2018 he will have to go through all the digests and find out the perfect match for the case.
Law assistant takes 3 parameters , that are a KEYWORD, DATE FROM and DATE TO. It searches through internet to find out all the relative case Laws and presents them in date order. It shows items by their titles, case Lawyers and jurisdiction. Upon selecting any case item user will be able to see the details of the case along with the case judgments.
Good thing about Law assistant is , it's easy to use so even a person with no Law background can search for caseLaws to get information.
Lawyers will have to add profiles along with their locations that can then be used to see the nearby lawyers. In case they would want to get in touch or would like to know about the lawyers near them. This feature is also beneficial for the non-professionals as they will be able to see the nearby lawyers to connect with.

Intended User

Intended users are Lawyers/Solicitors/Judges or anyone with the LAW background. It also focuses on everyday users who would want to search caseLaws for their personal or professional use.

Features

List the main features of your app.

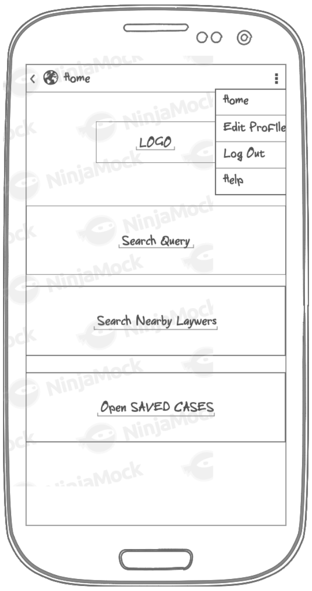
- App is written Solely in Java Programming Language
- Login Authentications using Phone Number
- Creating profile with profession and details of work
- Searches and gets current locations
- Searches case Laws based on Keywords
- Users can save the relevant case Laws in order to keep a record
- Can find nearby professionals in case they would need help
- App Uses a Home screen widget that will display currently selected caseLaw

User Interface Mocks

Screen 1

Main Screen that Allows Users to choose from one fo three options that are:

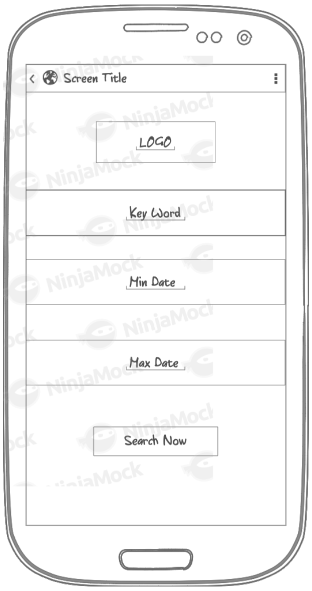
1. Search for a caseLaw
2. Search Nearby Lawyers
3. Check already saved caselawys/bookmarks



Screen 2

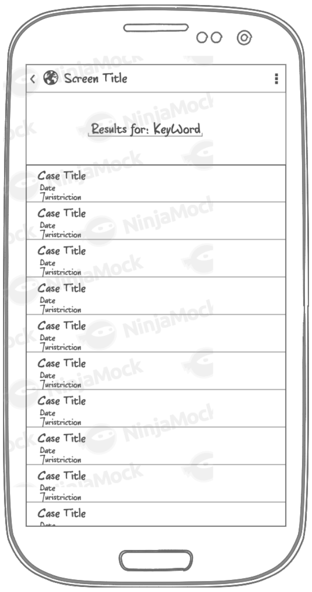
Upon Selecting the First option that is to search for a caselaw. Second screen will appear that will have following three fields to dill in.

1. Search KEYWORD that they would like to search.
2. Minimum date of case or FROM date
3. Maximum date of case or TO date.



Screen 3

When User hits the **Search Now** button app will find all the related caseLaws and will present it in a recycler view as shown in screen. Details of items will include:
1.Case title
2.Case Jurisdiction
3.Case Date



Screen 4

Selected item will then be opened in a new fragment to show the details of the case in a detailed manner. Where user will be able to read the text of the case judgment and all the details such as lawyer names, cour names and decision.
Figure 1 shows the view from a mobile phone and Figure 2 shows the view from a tablet.

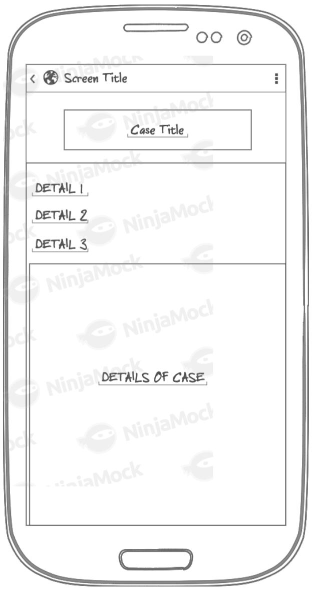


Figure 1. Phone View

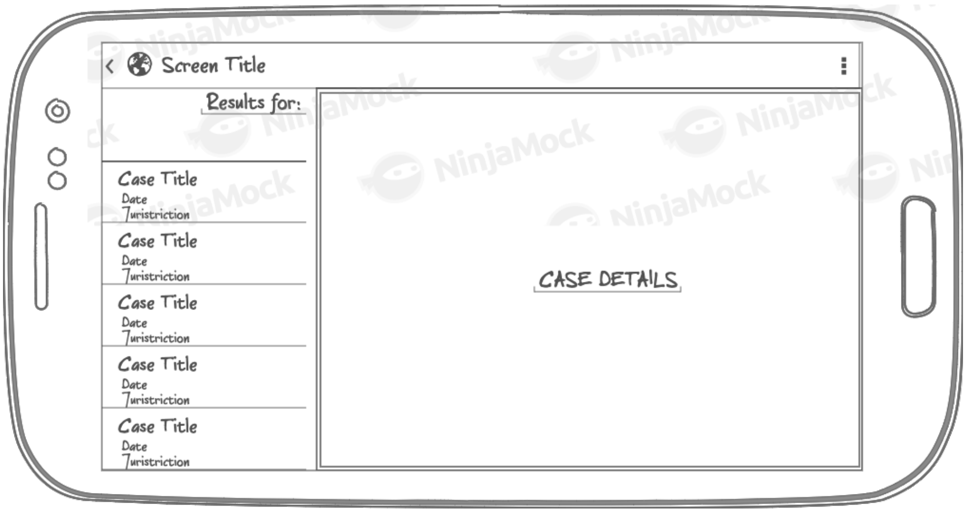
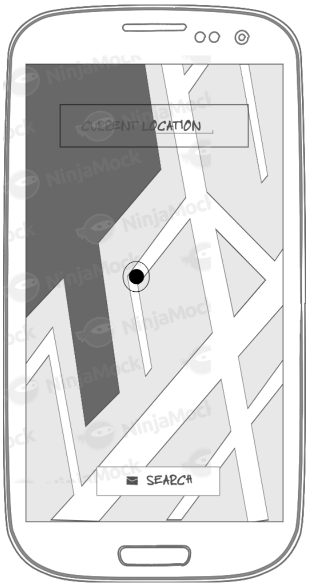


Figure 2. Tablet View

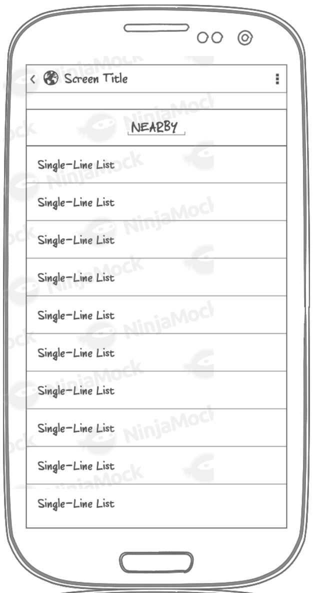
Screen 5

When user Selects To search for the nearby Lawyers following Screen will appear to get the current location of the user to search accordingly.



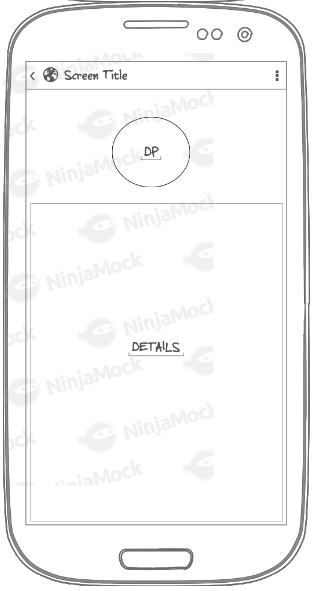
Screen 6

Once the search is complete all the nearby users details will be available in recycler view,



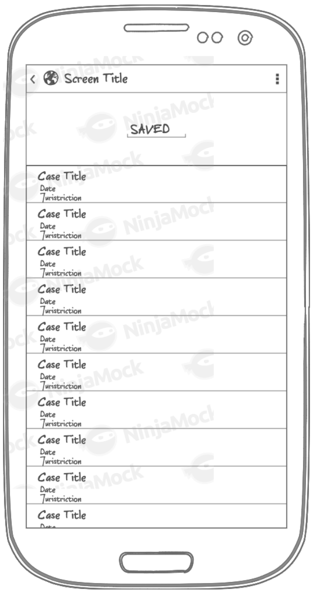
Screen 7

Once user clicks on an item from the recycler view it will then be displayed in a detailed view in next fragment.



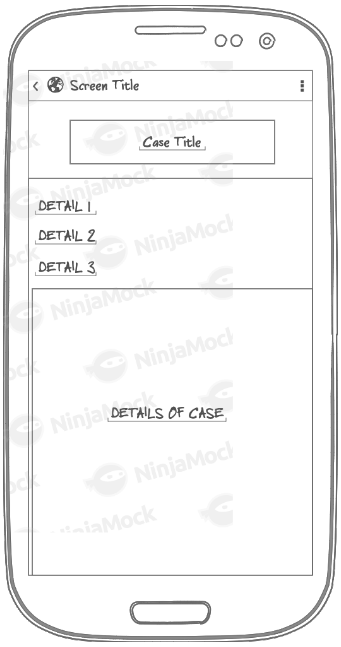
Screen 8

If user wants to see the previous stored data or the bookmarks from main activity this page will be listed to show the previous saved data on the phone.



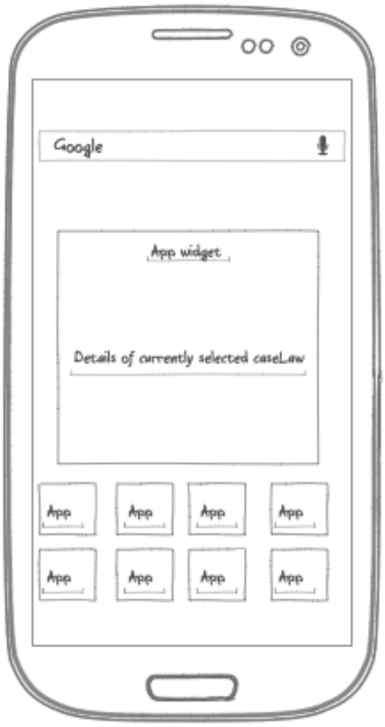
Screen 9

Clicking an item from Bookmarks will open the page to check for the details of the item.



Screen 10

App will have a Home screen widget that can be added manually from widget list if the user wishes to. Widget will hold details of currently selected caselaw so its details will be shown on the main screen.



Key Considerations

How will your app handle data persistence?

- To store users profile information, profile images and location Data Firebase realTime database will be used to make sure it will be possible to show and share the profiles with others users.
- Login details and authentications will also be done using Firebase authentications with phone number verifications.
- Items saved as bookmarks will be stored in Room database so it will be saved in the phone memory and will be available efficiently.
- The CaseLaw Access Project API by Harvard Law School to retrieve caseLaw data from internet. It's free to use API.

Describe any edge or corner cases in the UX.

- User can return back to screens by pressing back button any time. In order to make navigations easy app uses navigation drawer with options to return the the home page or any desired page at any time.
- Navigation Drawer will help user to navigate through the app.
- To make back and forth into the master Detail fragment user can hit a back button from detail fragment to go back the Master List.
-

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

Libraries required for the project are as follows.
- Firebase Libraries for using firebase Database and authentication
- Picasso to load images to profile images
- Google play services Location service to get the location fo User
- Google play services Maps API to show map and current location of the user.
- Volley to make network requests efficient.
- Room database Library to use Room to store data Locally.

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

Firebase Authentication:
Firebase Will be used to create login for the users with phone number verifications

Firebase Database:
Firebase database will be used to store users data that will include the profile info , profile DP, Users professional location, Phone number so it can then be used to share between the users.

Google Play Location Service:
Location Service will be used to get users current location that will then be used to compare with firebase database location data.

Google Play Maps:
Maps will be used to show locations of the user upon creating account and every time user makes a request to search nearby professionals. It will used to show current location of the user on the map

Required Tasks

Task 1: Project Setup

First step will be to setup the project by adding all the libraries needed and make sure everything is up to date and ready to be used.
Add following versions of Libraries

Android Studio: 3.4
Android Gradle Version : 5.1.1
Android Gradle Plugin Version: 3.4.0

Dependencies

com.google.firebase:firebase-core:17.0.1

com.google.firebase:firebase-storage:18.1.1

com.google.firebase:firebase-auth:18.1.0

com.android.volley:volley:1.1.1

com.google.android.gms:play-services-location:17.0.0

com.squareup.picasso:picasso:2.71828

Task 2: Implement UI for Each Activity and Fragment

- Creating First UI for All activities
- Adding Menus and maps
- Set up navigation drawer menu
- Setup Menu items
- Select icons needed for activities

Task 3: Setup Firebase and Maps

Setup Firebase and maps to get the project started. So it can be tested by adding data into the Firebase and Maps will be visible.

- Start coding/ Development
- Create Firebase authentications class and get Registration Ready.
- Setup Firebase Database and Firebase console to get it ready to save data and authentications
- Setup Google Maps to show on the screen

Task 4: Complete Firebase and Maps Locations

Setup google Maps and Location Services as this data is nessacerry to make a new account. User will not be able to start the app without saving locations data and firebase authentications
Describe the next task. List the subtasks. For example:

- Setup google Location Service
- Save locations to Firebase database
- Save users information to Firebase Database

Task 5: Setup API data, recycler views and AsyncTasks

Once Logins and Maps are completed next step will be to setup recycler views data and implementing CaseLaw API to retrieve data from internet

- Create Recycler Adapters
- Create Model Classes
- Setup Network request query
- Setup AsyncTasks for queries to run in background

Task 6: Setup Room to store Users Selected Data

Setup Room database to store , read and delete data from the database Using LiveModel pattern. So app does not query API requests unnecessary .

- Create Room Database
- Create DAO
- Implement LiveModel

Task 7: Create Widget

- Add widget UI for home screen
- Code widget to receive the data from main activity

Task 8 Add accessibility to app

- Add content description where needed
- Recycler-view Navigation using D-pad

Task 9: Polish the UI and Finish

- Finalise design
- Recycler cards final designs
- Add animations and background images
- Add all the hard-coded strings in strings.xml file