

Pocket Caddy

Aaron Chen, Ben Ko, Gio Ong



Agenda

01.

Introduction

02.

App Demo

03.

Code Walkthrough

04.

Limitations

05.

Future extensions

06.

References





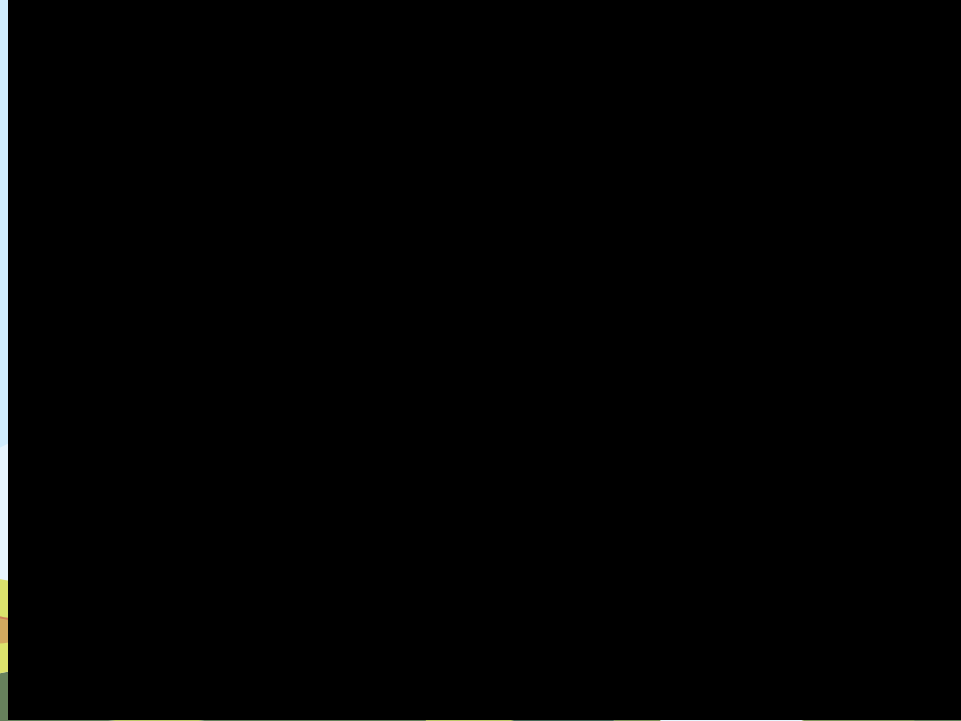
What's a “Caddy”?

The caddy is a personal assistant and teammate to the golfer.

A caddy's responsibility composed of three main elements:

- **Equipment management**
- **Club choices** 💡
- **Play strategy** 💡

App Demo



UI <-> class

```
1  /**
2   * This class represents the fourth fragment, which based on the user input,
3   * shows the recommended club and the
4   * picture of the club recommended.
5   */
6  public class FourthFragment extends Fragment {
7      private FragmentFourthBinding binding;
8      private GolferViewModel viewModel;
9
10     /**
11      * Initializes the binding and inflates the layout for the fragment
12      *
13      * @param inflater The LayoutInflater object that can be used to inflate
14      * any views in the fragment,
15      * @param container If non-null, this is the parent view that the
16      * fragment's
17      * UI should be attached to. The fragment should not add the view itself,
18      * but this can be used to generate the LayoutParams of the view.
19      * @param savedInstanceState If non-null, this fragment is being re-
20      * constructed
21      * from a previous saved state as given here.
22      */
```

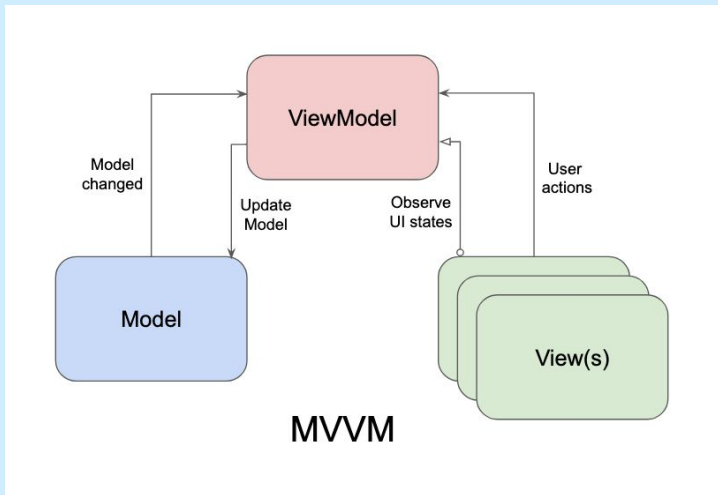
Allows the code to work with the user input and interactions through UI (xml).

For example:

```
binding.buttonNewSwing.setOnClickListener(v ->
    NavHostFragment.findNavController(FourthFragment.this)
        .navigate(R.id.action_FourthFragment_to_ThirdFragment)
        );
```

Inheritance

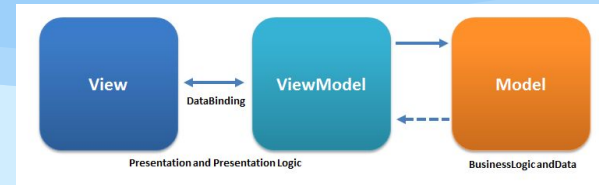
Why MVVM?



Data Binding 
Takes input from UI directly

Lifecycle-awareness 
Stores data through lifecycle

ViewModel (GolferViewModel)



```
1 package com.example.phonecaddy;
2
3 import androidx.lifecycle.ViewModel;
4
5 /**
6  * Represents GolferViewModel class which is a ViewModel class, which stores
7  * and manages data
8  * gathered from the user interface throughout the entire fragment that the
9  * user interacts with.
10 */
11 public class GolferViewModel extends ViewModel {
12     private String firstName;
13     private String lastName;
14     private int age;
15     private String gender;
16     private String experienceLevel;
17     private String dominantHand;
18     private int distance;
19     private String weatherCondition;
20
21     /**
22      * Gets the first name of the user
23      * @return the first name of the user
24      */
25     public String getFirstName() {
26         return firstName;
27     }
28 }
```



```
1 /**
2  * Based on the user inputs in the second and the third fragment, generates
3  * a club recommendation,
4  * and sets the text to show the club recommendation and shows the image of
5  * the club corresponding
6  * to the resource ID of the club that is recommended.
7  */
8 private void generateRecommendation() {
9     String firstName = viewModel.getFirstName();
10    String lastName = viewModel.getLastName();
11    int age = viewModel.getAge();
12    Gender gender = Gender.valueOf(viewModel.getGender().toUpperCase());
13    Person person = new Person(firstName, lastName, age, gender);
14
15    TypeOfGolfer skillLevel = TypeOfGolfer.valueOf(viewModel
16        .getExperienceLevel().toUpperCase());
17    DominantHand dominantHand = DominantHand.valueOf(viewModel
18        .getDominantHand().toUpperCase());
19    Golfer newGolfer = new Golfer(person, skillLevel, dominantHand);
20
21    int distance = viewModel.getDistance();
22    TypeOfCondition condition = TypeOfCondition.valueOf(viewModel
23        .getWeatherCondition().toUpperCase());
24    String clubRecommendationGenerated = newGolfer.recommendClub(distance,
25        condition);
26
27    int clubImageOnFile = setClubImages(clubRecommendationGenerated);
28 }
```

Limitations

01

Beginner Centric

02

Ignores Terrain

03

Only on Android



What's next?



Birdies!

Do you have any questions?



Resources

- <https://www.c-sharpcorner.com/article/how-to-implement-chatgpt-in-android-application/>
- <https://www.geeksforgeeks.org/how-to-build-a-chatgpt-like-app-in-android-using-openai-api/>
- <https://blog.gopenai.com/i-build-an-android-app-that-uses-chatgpt-api-here-is-how-it-works-9ecbf636f9ba>
- https://cookbook.openai.com/examples/how_to_format_inputs_to_chatgpt_models
- <https://google.github.io/volley/>
- <https://github.com/skydoves/chatgpt-android/tree/main>
- <https://www.bunkered.co.uk/golf-news/who-is-the-caddie-for-scottie-scheffler/>
- <https://www.sportskeeda.com/golf/news-who-rose-zhang-s-caddie-uncovering-details-breakout-golfer-s-bag-man>
- <https://developer.android.com/reference/android/app/Fragment#Lifecycle>
- <https://proandroiddev.com/a-holistic-approach-to-handling-user-input-in-android-38f51ca5592>
- <https://developer.android.com/guide/fragments/saving-state>
- <https://medium.com/@dheerubhadoria/android-mvvm-how-to-use-mvvm-in-android-example-7dec84a1fb73>



Sources for Images Utilized in the App

<https://www.globalgolf.com/golf-clubs/1046336-xxio-prime-9-iron-individual/>

<https://wedgewoodgolf.com/products/8-iron-38>

<https://www.sevengolf.com/>

<https://wedgewoodgolf.com/products/6-iron-30>

<https://www.uskidsgolf.com/ul54s-5-iron>

<https://www.sportspar.com/jelex-x-heiner-brand-golf-club-iron-4-right-handed>

<https://the3iron.golf/shop/golf-clubs/fairway-woods/legno-4-xxio-modello-x-loft-165/>

<https://www.pinemeadowgolf.com/golf-clubs/hybrids/excel-egi-hybrids/>

<https://wedgewoodgolf.com/products/gold-series-hybrid-driver>

<https://www.golfwrx.com/668653/this-pga-tour-pro-is-using-a-ridiculously-expensive-1-of-1-sand-wedge-heres-why/>

<https://www.callawaygolf.com/golf-clubs/wedges/wedges-2022-jaws-raw-chrome.html>