Johvonne Keane
100784274
Wednesday October 2nd, 2024
Mobile Application Development,
SOFE 4640U
github link

EMI Calculator

Introduction

In this assignment the task was to create an EMI (equated monthly installment) calculator app in Kotlin or Java. The language that was selected for this implementation was Kotlin. The application consists of two activities: the **Calculator Activity**, and the **DetailedViewActivity**. The purpose of the **Calculator Activity** in this application is to get user input and calculate and display the EMI, but it also calculates the total amount paid and total insurance paid. The **Calculator Activity** only calculates the total amount paid and total insurance paid, but it does not display it to the user as the view would get crowded. This is where the **DetailedView** view comes in. The **DetailedView** view will display everything to the user including the EMI. Both views consist of navigation buttons to navigate between the two.

CalculatorActivity

The **calculator_activity.xml** consists of a couple different elements, such as **ConstraintLayouts**, **TextViews**, **EditTexts**, **Spinners** and **Buttons**. All these elements were necessary to make a user interface to ensure good user experience. **EditTexts** and **Spinners** were used to get user input, while the **TextViews** were used to display information to the user and guide them through what input is needed at what input field. **Buttons** were used so the user could interact with the application and calculate their EMI once they had entered the necessary information. Finally, to hold it all together in a neat layout a **ConstraintLayout** was used to align and position everything within the application, so everything that needs to be visible is visible.

Now the **CalculatorActivity.kt** file consists of the logic needed to bring functionality to the **main_activity.xml**. Some key functions from **CalculatorActivity.kt** are **calculateEMI**, **calculateTotalInterest**, **integerValidation**, **spinnerValidation**, and **errorMessageFormatter**. The **calculateEMI** and **calculateTotalInterest** functions do just as their titles imply, calculating EMI, total interest paid, and total amount paid. The **validation** functions validate the user inputs making sure that they are valid before any type of calculation is performed. This is done to handle errors and avoid having the app from crashing. Without these functions the application would seem effectively broken as no feedback would be sent to the user after a button press. These functions ensure that if the user does end up entering invalid data and or data types, an error message will appear on the screen directing the user to ensure that they enter valid data the next time they try to run the calculations. **SpinnerHeaderAdapter.kt** is a class used to create a **custom adapter** for the Spinner element. Its purpose is to style **Header** and **Item** elements differently in the spinner view. So, the user can differentiate between what is just meant to guide/ describe the elements of that field and what is an actual selection option.

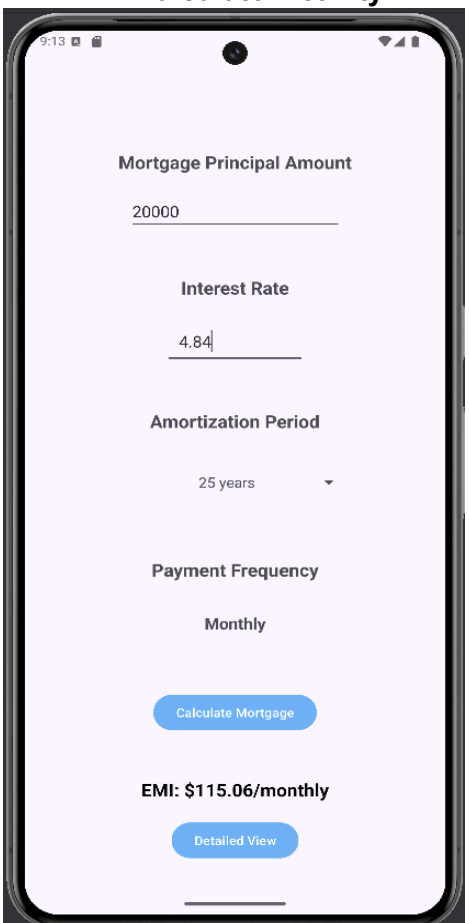
There are two buttons present on the screen are a **Calculate Button** and **Detailed View Button**. The **Calculate Button** is once again pretty self explanatory, it just calculates the EMI and will have it displayed in the same activity. However, under the hood it will also calculate the total amount paid and total insurance paid, but is not displayed to the user. The **Detailed View Button** will open a new activity in which the user can see a more detailed view of the application.

Detailed View

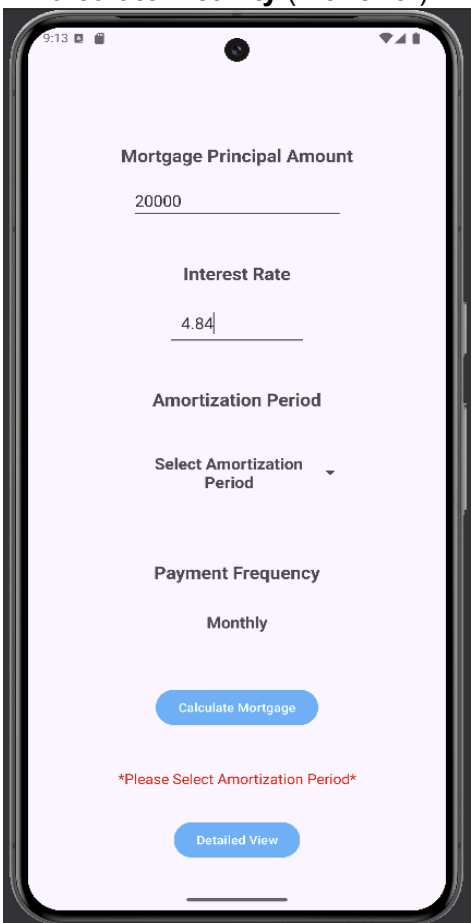
The **detailed_view.xml** is used to create the layout of the **DetailedViewActivity**. This layout does not have any input fields, as its main goal is to display information; it only uses **TextViews**, **ConstraintLayouts**, and a singular **Button**. The **TextViews** feature not only the EMI but also the total interest paid as well as the total amount paid. There is a header on the layout to differentiate which number is what The **Back Button** on this page is to go back to the **CalculatorActivty** when the user is done viewing the more detailed information and ready to make another calculation.

DetailedView.kt has all the logic needed to make the present elements work. It does not feature any named function; however, it does feature the use of intent. As this activity has no functions, the intent has some information that was passed to it. On the creation of the intent in **CalculatorActivity.kt** it stored some information in the intent the EMI, total interest paid and total amount paid, that is then passed to **DetailedView.kt**. Values can be passed via **intent.putExtra()**. The final feature of this file is the **Back Button** which will update the intent and start another activity bringing the user back to **CalculatorActivity.kt**.

CalculatorActivity



CalculatorActivity (with error)



DetailedViewActivity