This is how the app GUI will appear when complete:

There are 7 tableRows. Rows 1 through 3 all contain a TextView/label object and an EditText/textbox object. Row 4 contains a TextView object with a single space character. This blank row is used to separate the three input rows from the results row (Row 5). Row 5 also contains a TextView object and an EditText object. You may also decide to use two TextView objects since the second object will be used to display the calculated monthly payment. Row 6 is another blank row. Row 7 contains two Button objects.

1. Launch the Android Studio. If you are launching for the first time below screen appears. Click on “Start a new Android Studio project”.

![Welcome to Android Studio](image-url)
2. Enter “MortgageCalculator” as Application name. We can also see where the project is getting stored in the Project Location field. Click Next.

3. Select Phone and Tablet. Click Next.
4. Select Empty Activity, Click Next.

5. Click Finish.
6. After you click Finish in above step. Below screen pop’s up. It might take couple of minute to completely setup the project.

7. Now it’s time to verify, if we have setup everything correct. Click on “Run ‘app’”. Below Screen pops up.
Select the Android virtual device. During the installation process Android virtual device will be created. Select the Android virtual device. If Virtual device is not created click on the button next to the Android virtual device drop down list. Follow the instruction in the wizard to set up appropriate virtual device. Click on OK.
8. Below screen pops up. To unlock Hold the left mouse button and scroll mouse upwards from the lock symbol shown in the figure below
9. After unlock the screen. Below screen shows up.
10. Now it’s time to implement Mortgage Calculator application. Click on the Text tab and change the “RelativeLayout” tag to “TableLayout”.

To switch between Design view and Code.
11. We will now add the TableRow, TextView and EditText components and Buttons. The components are found in the Palette panel on the left side of the IDE.

Drag and drop seven TableRow components onto the TableLayout component in the Component Tree window (See figure below). A red rectangle will outline the TableLayout tag when you have properly dragged the TableRow component over the TableLayout component. Now drag a TextView and an EditText component on Rows 1, 2, 3 and 5. Rows 4 and 6 should receive a single TextView component. Finally, Row 7 should receive two Button components.
Using the image of the final GUI at the top of the document, Change the text inside of the TextView components on Rows 1, 2, 3 and 5 by double clicking “Medium Text” in the phone image and typing in the appropriate label value. For example, the text property for the TextView on Row 1 should be changed to “Principal Amount:”.

Double click the EditText component (shows up as an underscore on the phone image) on the corresponding rows and change the id property to something meaningful. For example, the EditText for Row 1 should be changed to “etxtPrincipal”. Repeat this for all the EditText components.

Drag and drop two Button controls on TableRow 7. Double click the button images and change the text properties to “Reset” and “Calculate”. Change the corresponding id properties to btnReset and btnCalculate. Click on the Reset button image and scroll down to the onclick property in the Properties window. Type in the name of the Java method that should be called when the button is click. You choose any name since we will be creating the method in the project Java file (e.g. resetForm for the Reset button and calculatePayment for the Calculate button).

12. After Adding and arranging all the components, your screen should look like snap shot shown below.
13. Now it’s time to write some code to implement functionality to mortgage calculator. In the MainActivity.java we need to declare variables for buttons and Edit text component that we created in Layout file.

```java
package com.example.varsha.mortgagecalculator;

import android.support.v7.app.ActionBarActivity;
import android.os.Bundle;
import android.view.Menu;
import android.view.MenuItem;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.RelativeLayout;

public class MainActivity extends ActionBarActivity {

    RelativeLayout background;
    Button btnCalculate;
    Button btnReset;
    EditText txtPrincipal, txtInterest, txtPeriod, txtPrice;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
    }
}
```
14. Now we need to initialize the variables that were created in the step 13.

15. Now we need to create listeners for the buttons. The formula for calculating the monthly payment: \( \text{Payment} = \frac{(\text{interest} \times \text{principal})}{(1 - \text{Math.pow}(1 + \text{interest}, -\text{period}))} \);

   **Note:** The user will enter in a percentage for the interest rate so you must take the value entered and divide it by 1200 to convert it to a decimal amount and a monthly interest rate.

   ```java
   protected void onCreate(Bundle savedInstanceState) {
       super.onCreate(savedInstanceState);
       setContentView(R.layout.activity_main);

       EditText etxtPrincipal = (EditText) findViewById(R.id.etxtPrincipal);
       EditText etxtInterest = (EditText) findViewById(R.id.etxtInterest);
       EditText etxtPeriods = (EditText) findViewById(R.id.etxtPeriods);
       EditText etxtPayment = (EditText) findViewById(R.id.etxtPayment);
   }
   
   protected void resetForm(View v) {
       etxtPrincipal.setText(""),
   }
   
   protected void calculatePayment(View v) {
       double principal = Double.parseDouble(etxtPrincipal.getText().toString());
       double interest = Double.parseDouble(etxtInterest.getText().toString()) / 1200;
   }
   
   16. Now we have completed coding. It’s time to run the application.