

Cuisine 2.0

An AI-powered cuisine for smart consumption with enhanced nutrition



Team: AYB Creations

*Diploma in Information Technology*

*in*

*Big Data Analytics*

**First Review**

**Design**

**Supervisor:** Mr Zoubeir Aungnoo

**AYB Team Members:**

* BABOOLULL Yougesh
* BHAGOOLI Divesh Singh
* SUBRON Jhivesh Kumar

**Date: 07 April 2021**

# Abstract

Designing is an interesting section in the project management cycle, where developers anticipate upcoming challenges and difficulties. It is where user requirements are transformed into a functional prototype.

Our Cuisine 2.0 project has 3 main parts:

* Cuisine Vision: A screen mount using a raspberry pi.
* Cuisine Android app
* Cuisine 2.0 Website

Each part has been linked to a main server, each accessing the same Database using our custom written PHP APIs.

//More intro to be review @Divesh @Yougesh

**Table of Contents**

[Abstract 2](#_Toc72761573)

[Our Potential Users 4](#_Toc72761574)

[Project Models 5](#_Toc72761575)

[Database 5](#_Toc72761576)

[Database-Relationship Diagram 5](#_Toc72761577)

[Description of Tables Used 5](#_Toc72761578)

[API 6](#_Toc72761579)

[Object Detection 7](#_Toc72761580)

[Model 7](#_Toc72761581)

[Algorithm 7](#_Toc72761582)

[Android App 7](#_Toc72761583)

[Class Diagrams of Activities 7](#_Toc72761584)

[Class Diagrams of Fragments 9](#_Toc72761585)

[Website 13](#_Toc72761586)

[Raspberry Pi 15](#_Toc72761587)

[Server 15](#_Toc72761588)

[User Interface 16](#_Toc72761589)

[Android App 16](#_Toc72761590)

[Raspberry Pi 23](#_Toc72761591)

[Website 23](#_Toc72761592)

**Table of figures**

No table of figures entries found.

# Our Potential Users

//To be done by @Yougesh

# System Architecture

## Database

Our project uses a MySQL database (u511941246\_cuisineUserAcc) to store user credentials. The table has 2 tables: UserAuth, User\_Secret\_Keys.

### Database-Relationship Diagram

Via PHP APIs

u511941246\_cuisineUserAcc

Cuisine 2.0Android App

UserAuth

User\_Secret\_Keys

Cuisine Vision

Cuisine 2.0 Website

## Description of Tables Used

The fields in the UserAuth table are: UserId, First Name, Last Name, Email Address, Verified.

|  |  |  |
| --- | --- | --- |
| **Field Name** | **Datatype** | **Description** |
| UserId | Integer | It is a unique id assigned to each user account created to be able to identify each user correctly. |
| First Name | String | It is the First Name of the User. |
| Last Name | String | It is the Last Name of the User. |
| Email Address | String | It is the email of the User |
| Verified | Boolean | It acts as a flag to check if a user has verified his email address or not. |

The fields in the User\_Secret\_Keys table are: UserId, Password.

|  |  |  |
| --- | --- | --- |
| **Field Name** | **Datatype** | **Description** |
| UserId | Integer | It is a unique id assigned to each user account created to be able to identify each user correctly. |
| Password | String (256) | It stores the password of the user which is encrypted using PHP SHA-256 cryptographic hashing method. There is no possible way of reversing a cryptographic hashed password as it uses a one-way function which makes the user account one of the most secured. |

//inventory and grocery list tables not yet added.

## API

//PHP APIs to be done by @Yougesh

## Object Detection

//Documentation already done and reviewed… @Divesh need to be add the part we reviewed already

### Model

//To be reviewed @Divesh

### Algorithm

//to be reviewed @Divesh

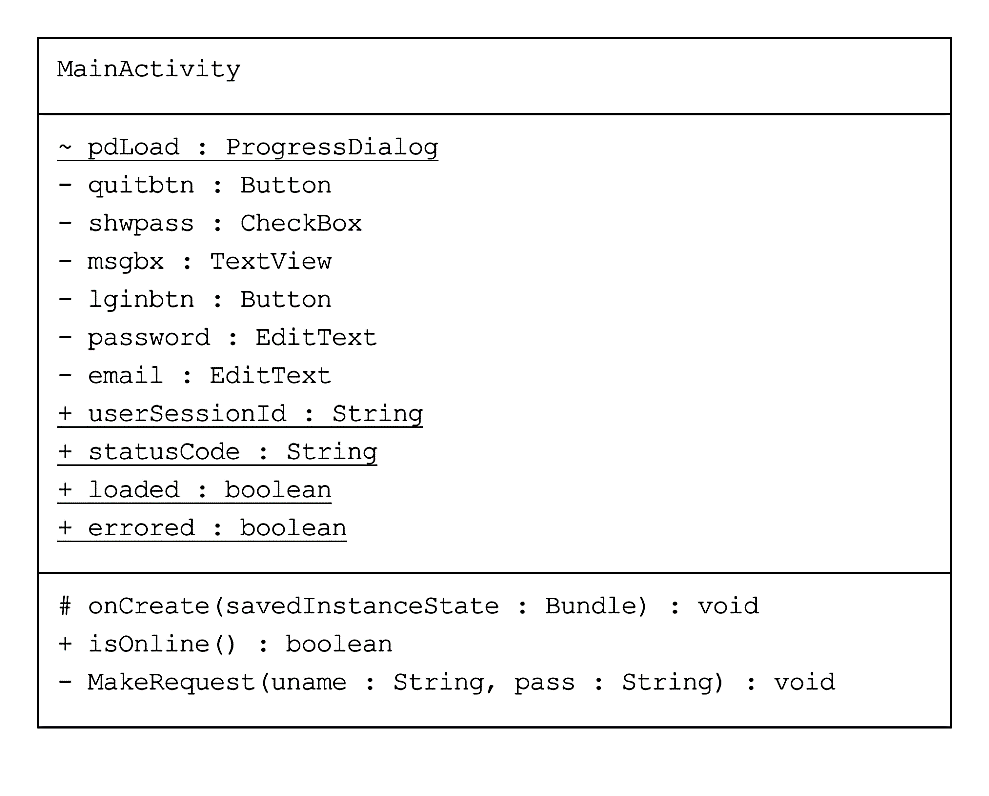
## Android App

The Android App makes use of a combination of activities and fragments to make user navigation easier.

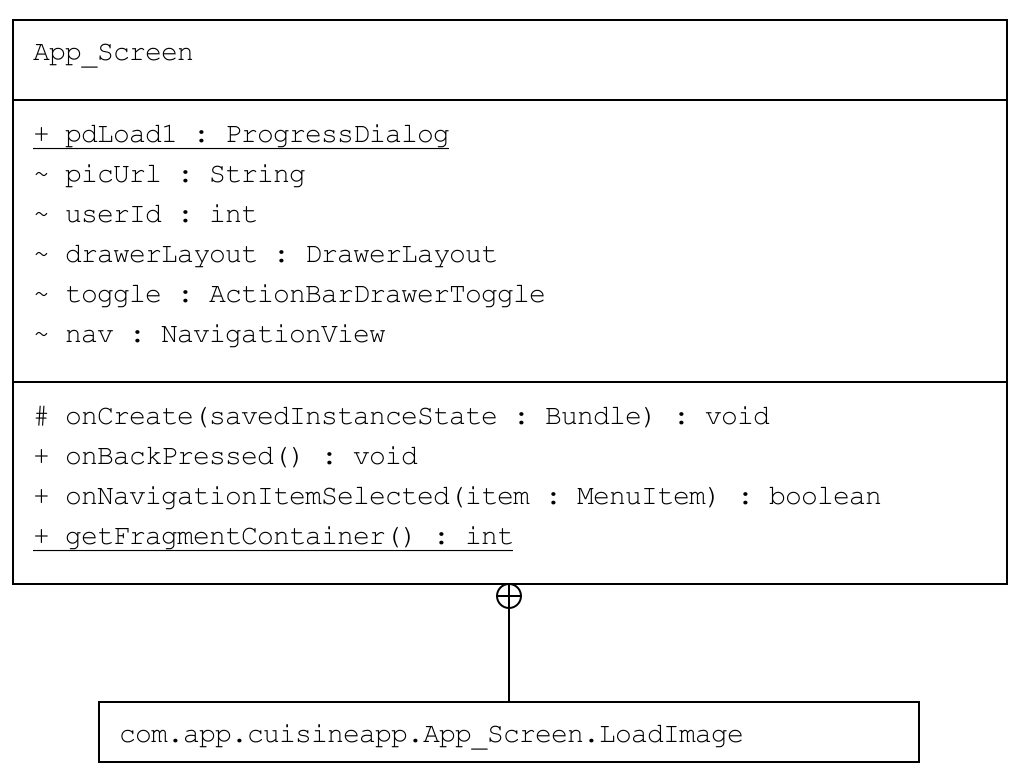
### Class Diagrams of Activities

* **MainActivity Activity**

This activity handles the login page of the app. It makes php requests to the Server using Google’s Volley library. Volley is a standard HTTP library that makes networking for android apps easier and faster.

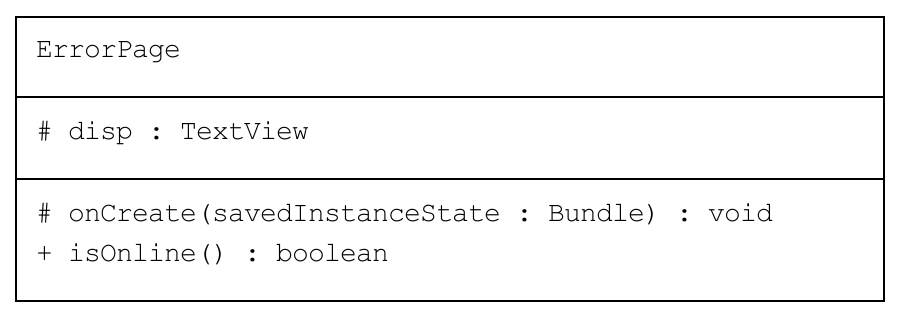


* **App\_Screen Activity**

This Activity extends the navigation view which is a left slider menu. It also handles the fragment views by swapping the fragment container with the selected item layout.

* **Error Page Activity**

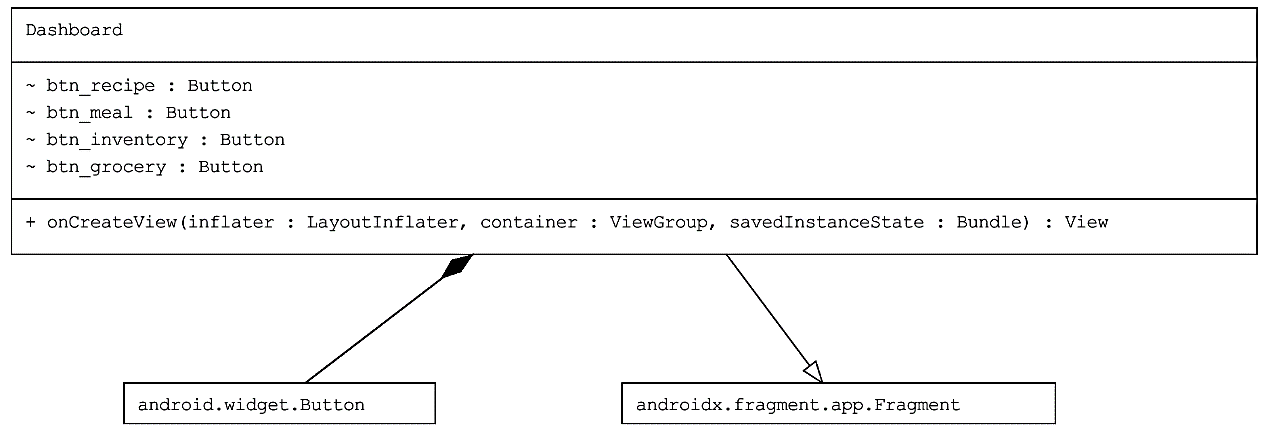
This activity is shown when the app faces a network issue or loss of internet connection. It has a timer that checks for internet connection every 5 seconds. If stable internet connection is found, it redirects the user to the page he was on



### Class Diagrams of Fragments

* Dashboard

This is the primary page after a user login. It contains useful tips and information for the user and well as quick links for the user to navigate through the app.

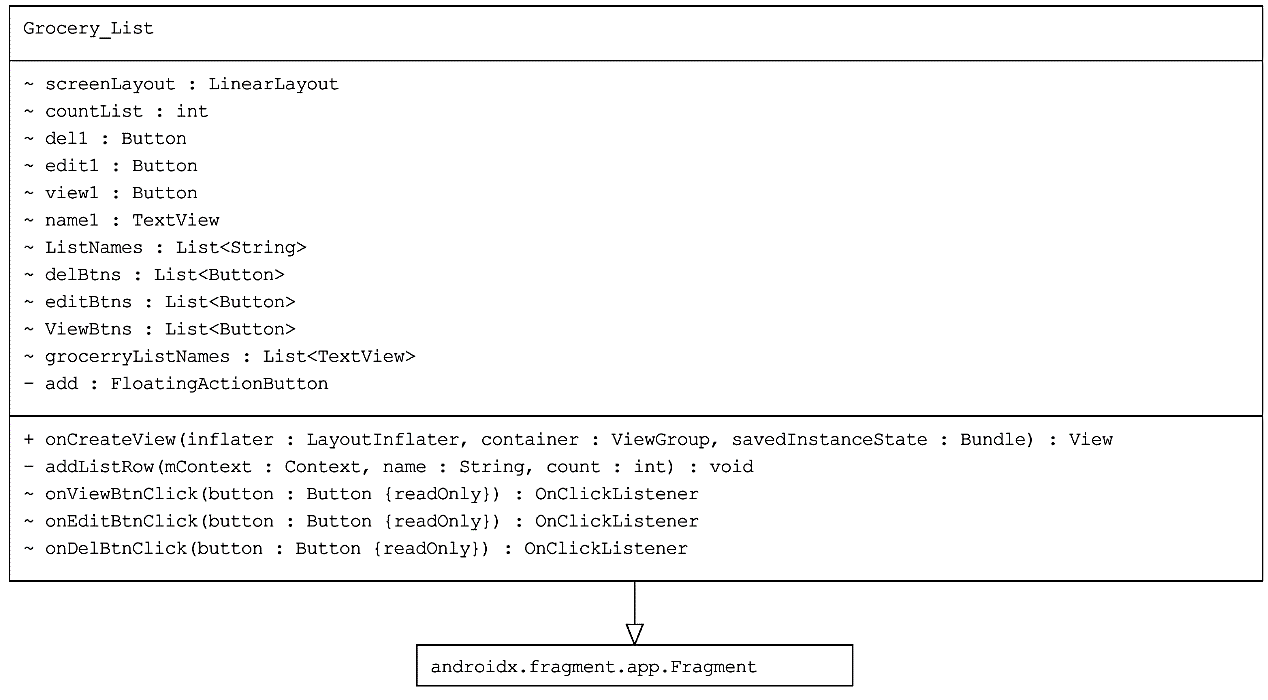


* Inventory

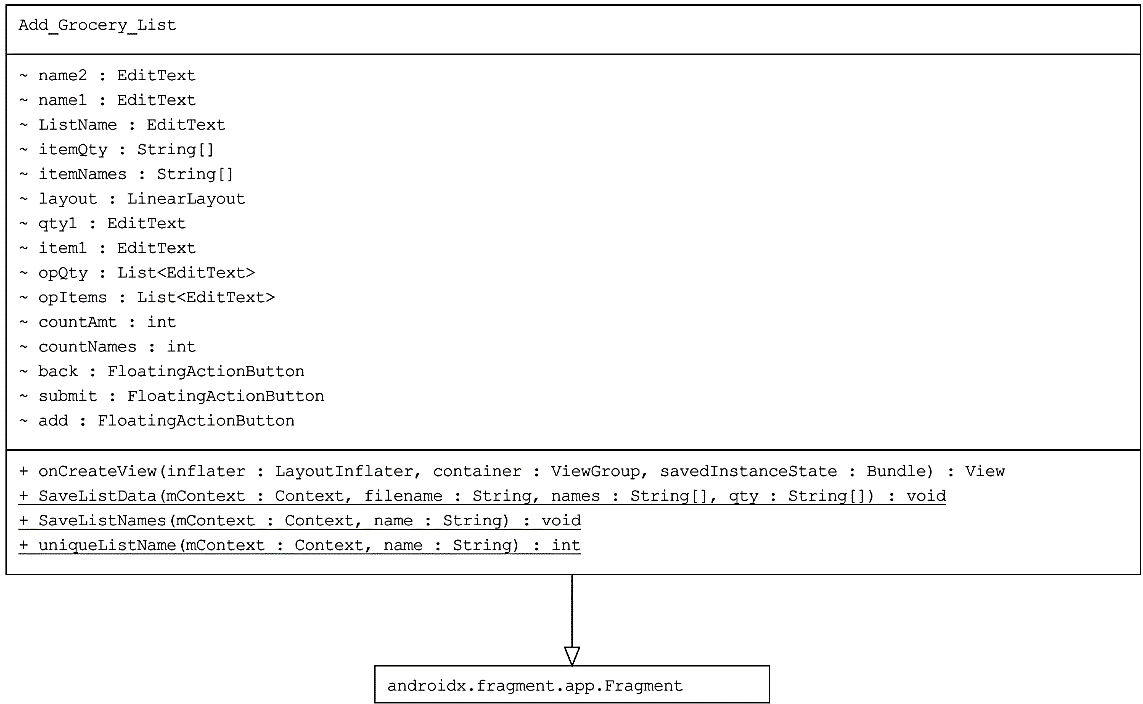
This fragment handles the inventory part of the user. It allows the user to manage his food inventory, that is, to add, delete or edit food items.

//not complete

* Grocery\_List

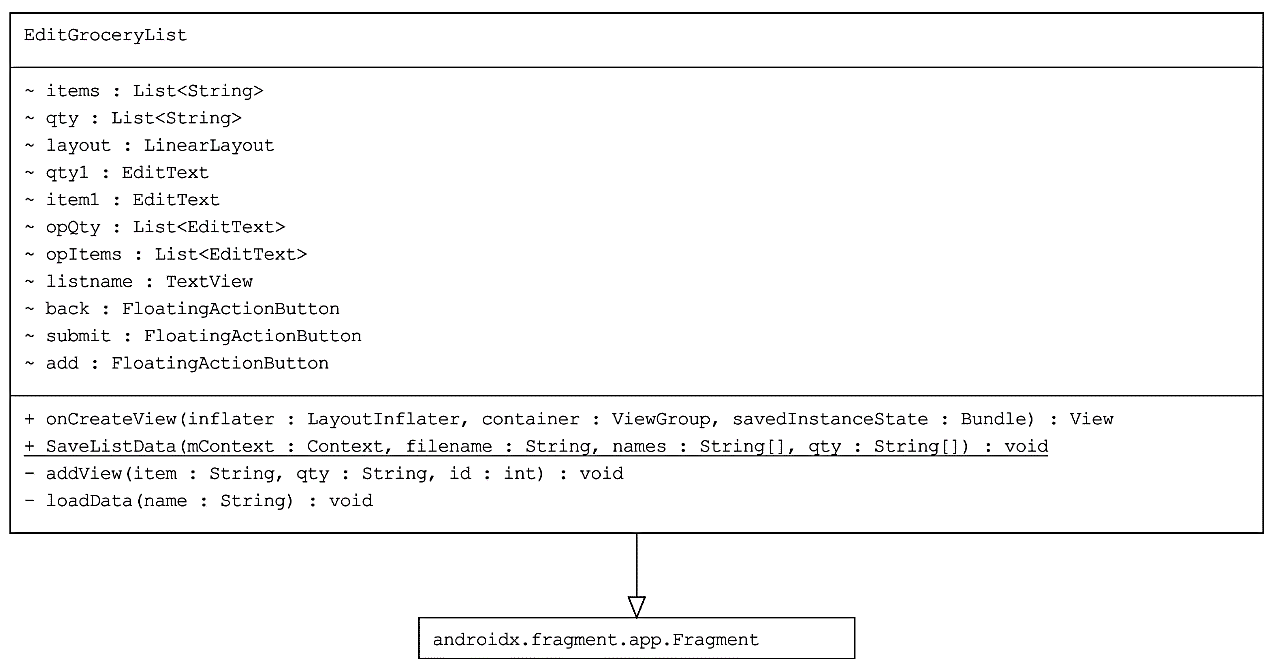
This fragment handles the grocery list for the user. It displays existing grocery lists and also allows the user to create a new grocery list.

* Add\_Grocery\_List

This is a sub fragment which provides an interface for the user to create a new grocery list.

* EditGroceryList

This sub-fragment provides an interface for the user to edit a grocery list.



* View\_Grocery\_List

//not complete

* Meal\_Planning

//not complete

* Recipe\_Suggestion

//not complete

## Website

The link to our website is: <https://cuisine.ayb-creations.tech>. The website of Cuisine 2.0 has been designed and implemented using WordPress. WordPress is a free and open-source content management system written in PHP and MySQL or MariaDB Database

The theme used to construct the website is Astra. Astra is one of the most lightweight and reliable website themes providing full customization. We chose Elementor as our website builder tool as it allows many plugins to integrate with itself and has all our required tools.

Plugins used for the website:

|  |  |
| --- | --- |
| **Plugin Name** | **Description** |
| Elementor - Header, Footer & Blocks | Used to create custom header and footer for website |
| Essential Addons for Elementor | Provides additional widgets for use |
| Fluent Forms | Provides Contact for interface |
| FluentSMTP | SMTP service for the fluent form which sends the administrator an email after someone has contacted them. |
| HT Mega - Absolute Addons for Elementor Page Builder | Elementor Addons Package providing additional widgets. |
| LiteSpeed Cache | Cache Webserver which makes loading of the resources faster |
| Premium Addons for Elementor | Elementor Addons Package with provides additional widgets for different functions |
| Simple Custom CSS and JS | Allows us to post our custom CSS and JavaScript codes |
| Wordfence Security | It is an anti-virus, firewall and malware scanner. It also protects the website from login bots. |
| WP Super Cache | A caching service to load high-definition images really fast. |

***Note:*** *All plugins are open-source plugins and are free.*

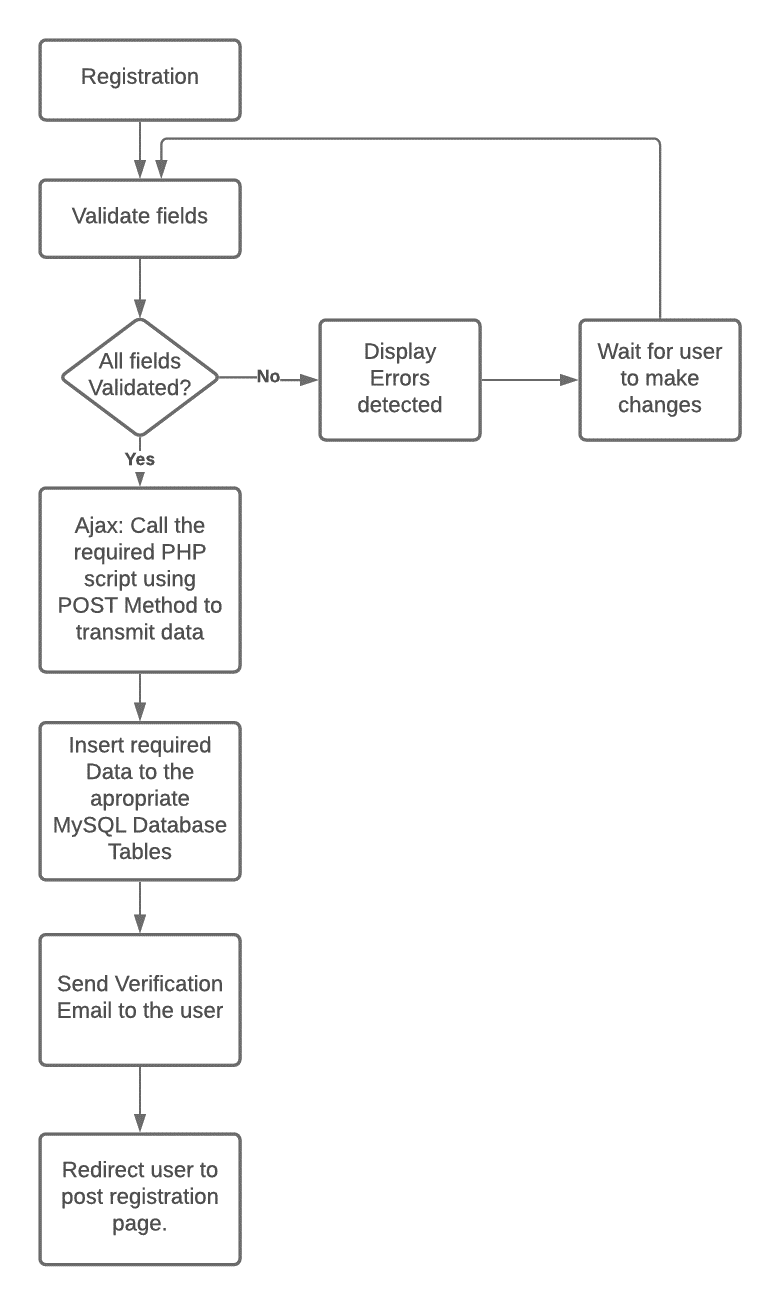
Our Website follows a different approach using a vertical slider. The website has a single page which slides multiple templates on scroll.

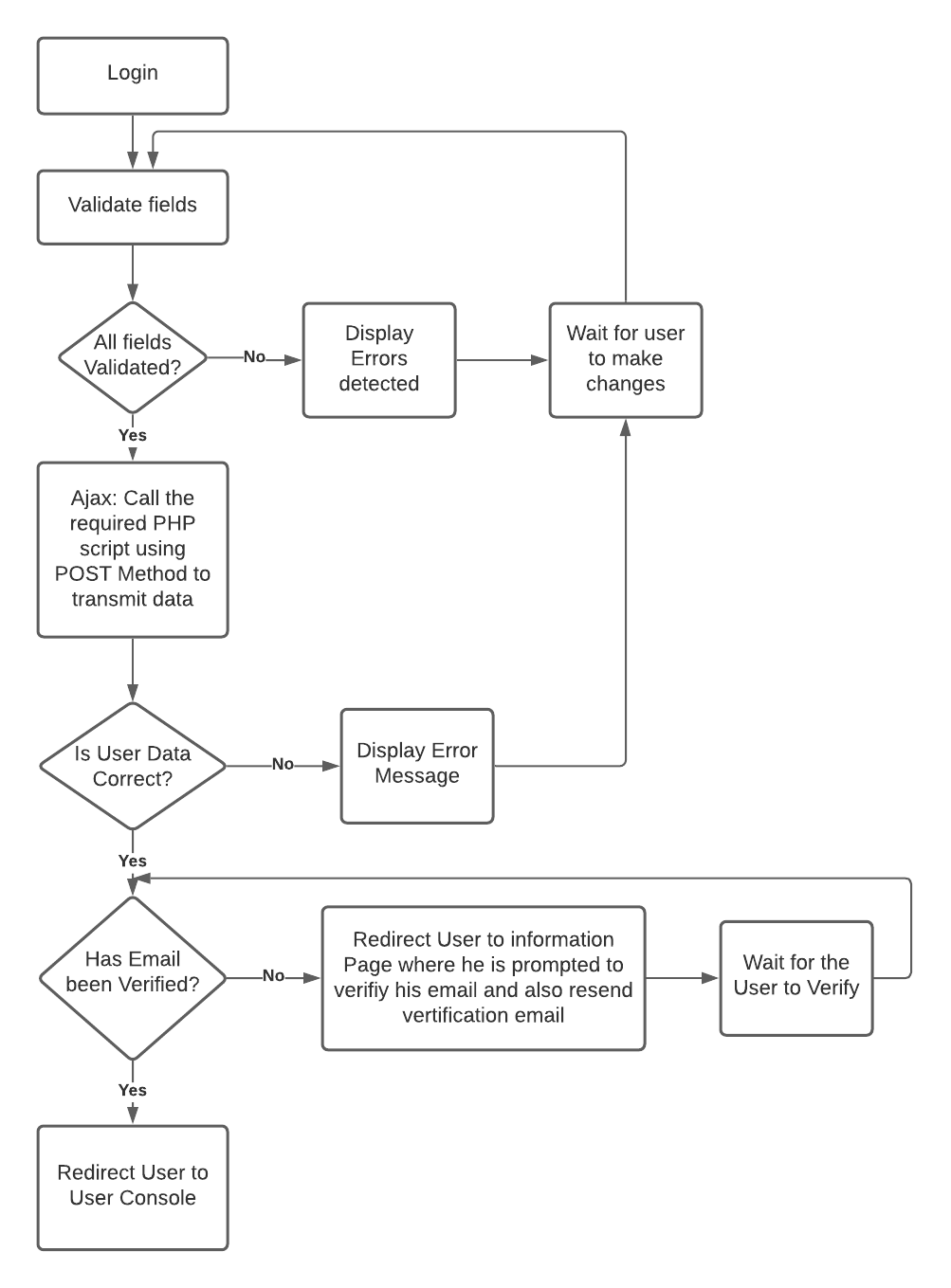
The Homepage has options for the user to login or create a new user account. All the user data are stored and queried from a MySQL database via our custom written PHP APIs.

The Login and Registration are displayed as a JavaScript overlay window which is written in HTML, CSS and JavaScript. When the user submits the form, the form connects to the PHP script using JavaScript Ajax.

We used Ajax to make calls to the PHP codes because:

1. WordPress doesn’t allow custom PHP to be included in the codes because WordPress is itself made of PHP which will create conflicts between internal codes and plugins.
2. Ajax makes asynchronous calls to be webserver which improves performance and speed.

**Flowchart demonstrating the user registration process:**

**Flowchart demonstrating user login:**

## Raspberry Pi

// photos and description by @Divesh

// Circuitry by @Yougesh

# Server

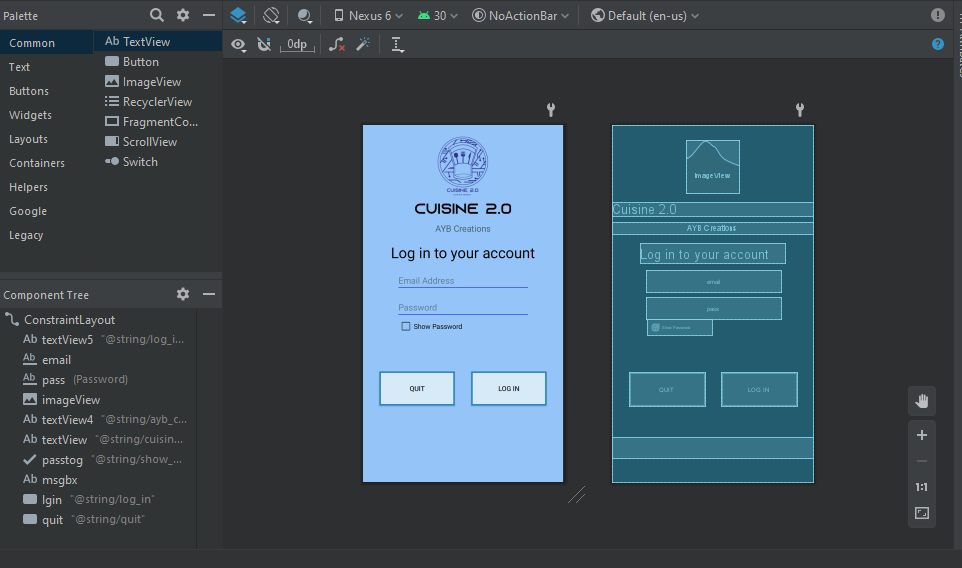
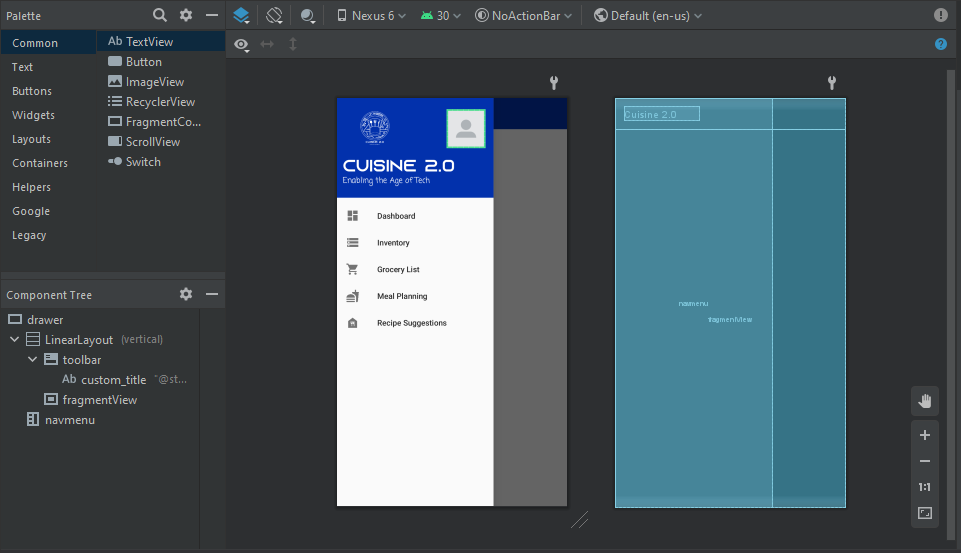
//@Subron complete this part

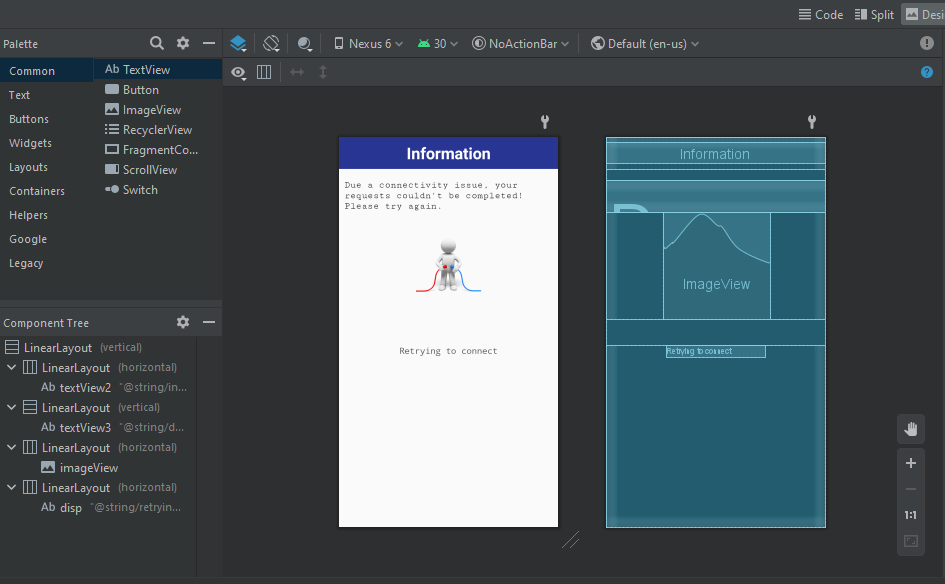
//Review areas: Hostinger, ibm cloud

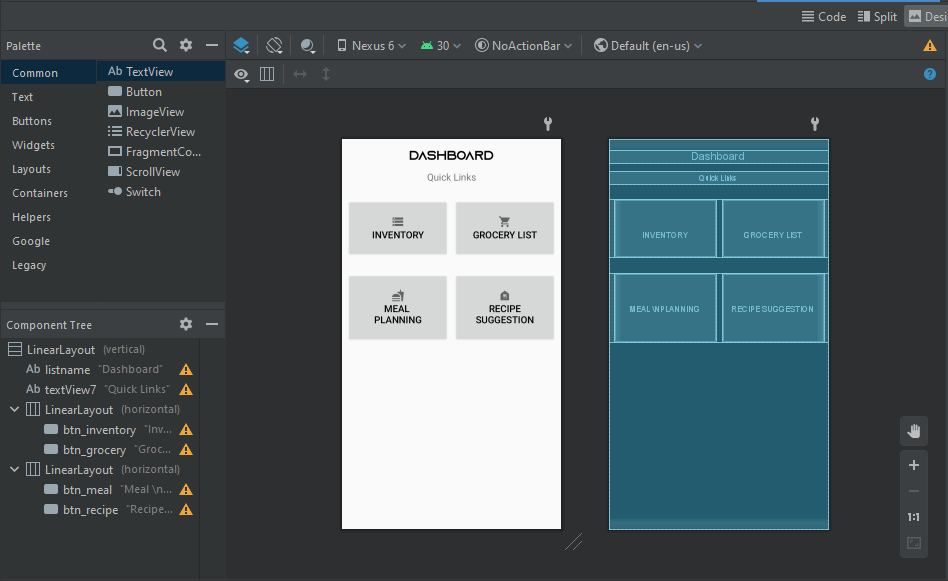
//Hostinger you already have access. IBM cloud: Communicate with @Divesh for further explanation.

# User Interface

## Android App

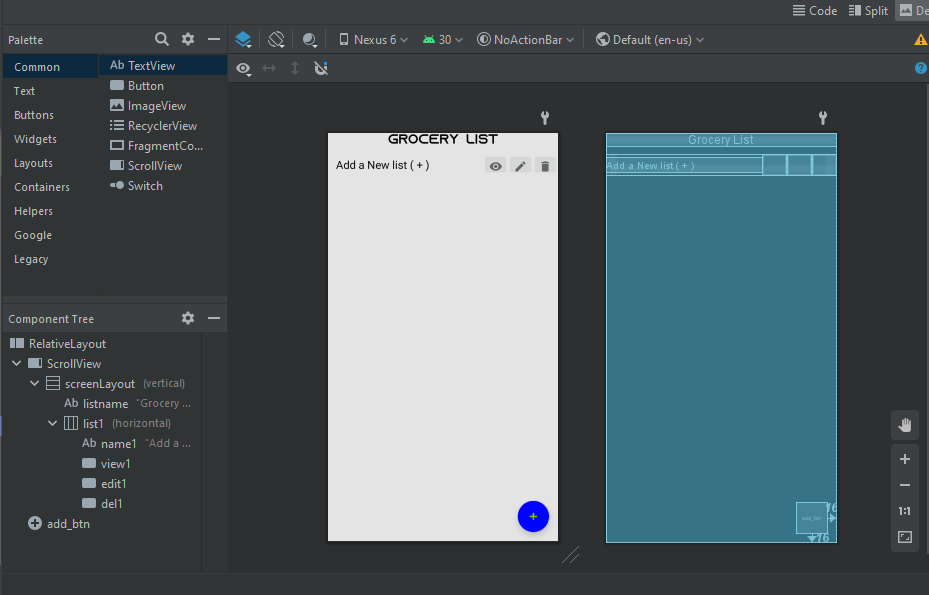
* MainActivity
* App\_Screen – Navigation Left Slider and Action Bar
* Error\_Page

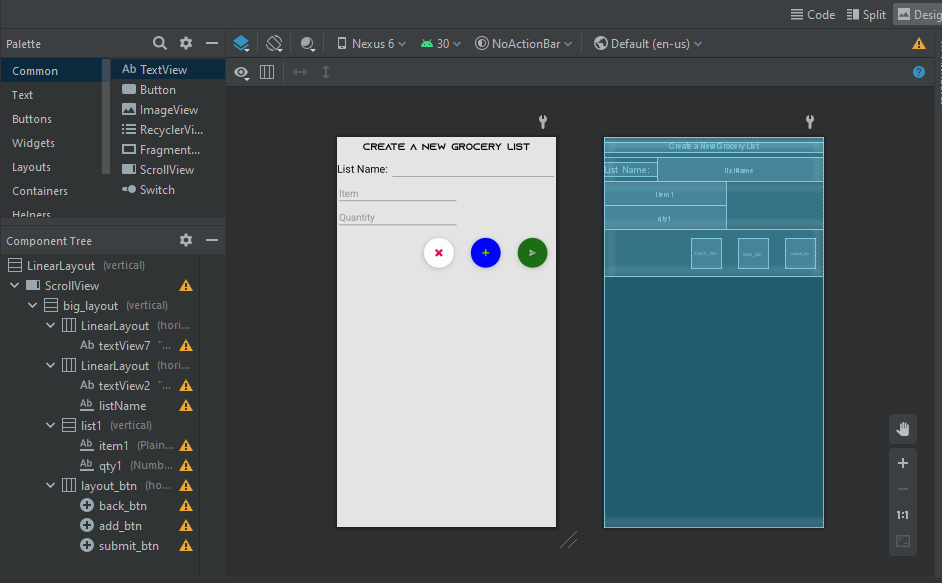
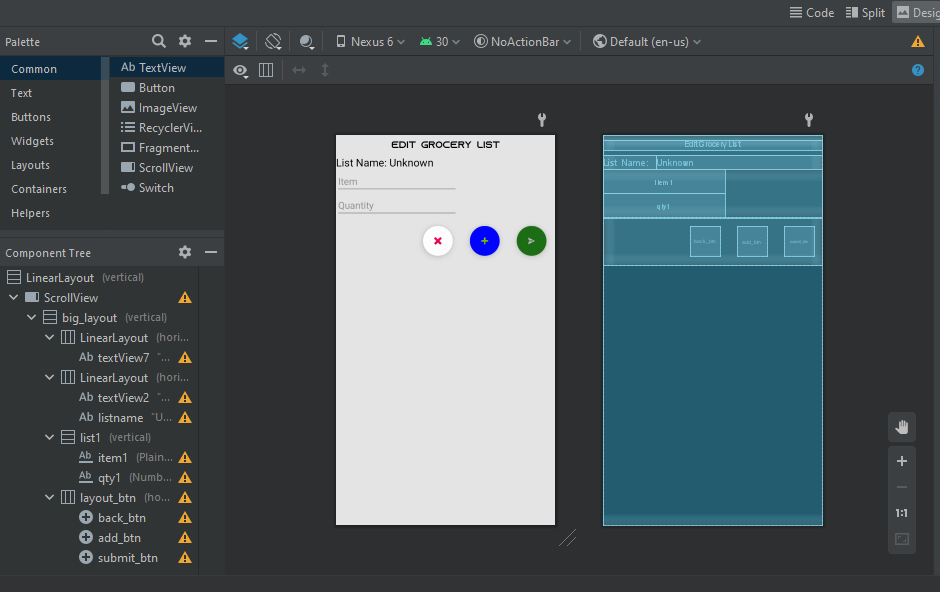
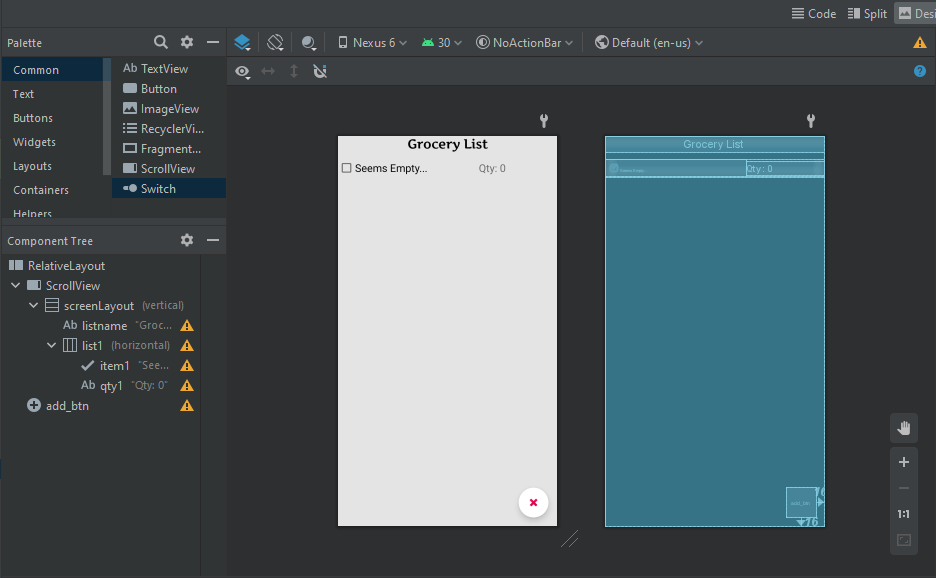


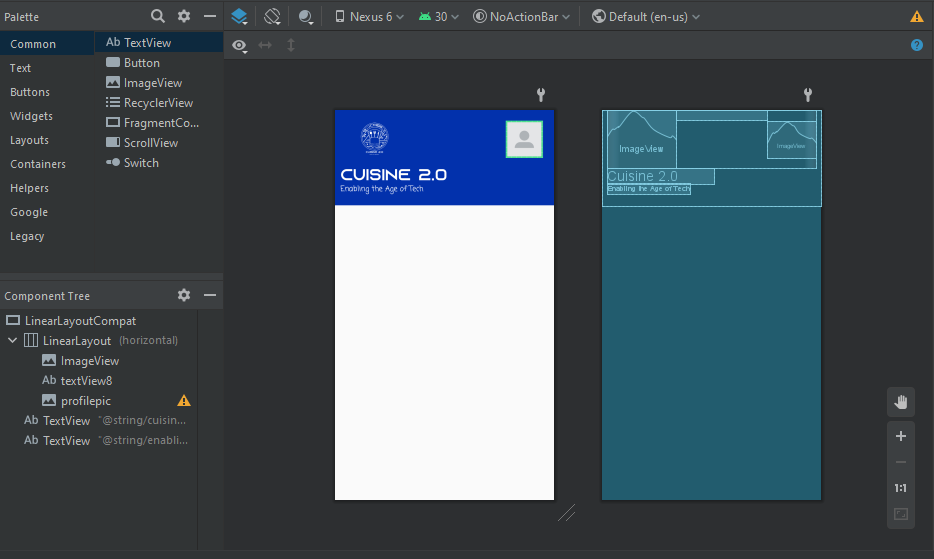
* Dashboard Fragment
* Inventory Fragment

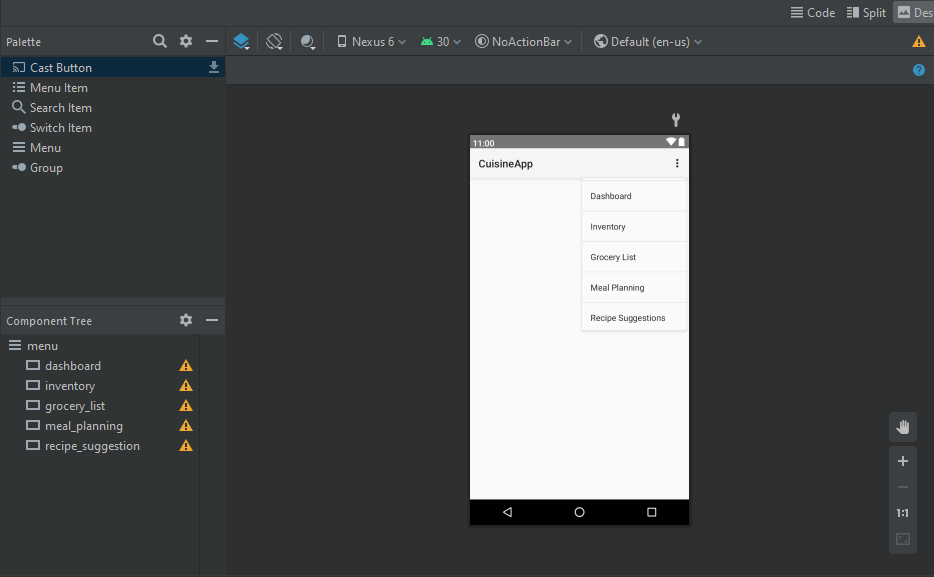
//not complete

* Grocery\_List Fragment



* Add\_Grocery\_List Sub-Fragment
* Edit\_Grocery\_List Sub-Fragment
* View\_Grocery\_List Sub-Fragment
* Header Part of the Navigation Slider



* Menu of the Navigation Slider

//More Fragments and some more layouts to be reviewed – by @Yougesh

## Raspberry Pi

//Still under development

## Website

// @Subron Check this one too.

// Screenshot by viewing each page the dashboard.