Kainotomia

Aphrodite

Daniel Moore (n01354875)
Alyssa Gomez (n01042777)
Jose Antonio Teodoro (n01384776)
Ryan Black (n01305403)

Table of Contents

Kainotomia 1

Brief Description of Project 3

Effort Table 3

GitHub Repo Link 3

Sprint Goals 3

Gantt Chart (AirTable) 4

Daily Stand-Ups 5

Sprint Retrospective(Lucidspark) 6

C4 Model 7

Design Principles Used 8

Design Patterns Used 9

Coding Progress since Deliverable 2 10

Brief Description of Project

The Aphrodite App is an interface between the user and the Aphrodite Smart Mirror. The user will be able to create an account in the app that will store the created layouts. Voice commands are also available to the user to control the mirror, as well as an LED colour switcher for the LEDs that are (will be) built into the smart mirror.

Effort Table

Name	ID	Signature	Effort
Daniel Moore	n01354875		100%
Jose Antonio Teodoro	n01384776		100%
Alyssa Gomez	n02042777		100%
Ryan Black	n01305403		100%

GitHub Repo Link

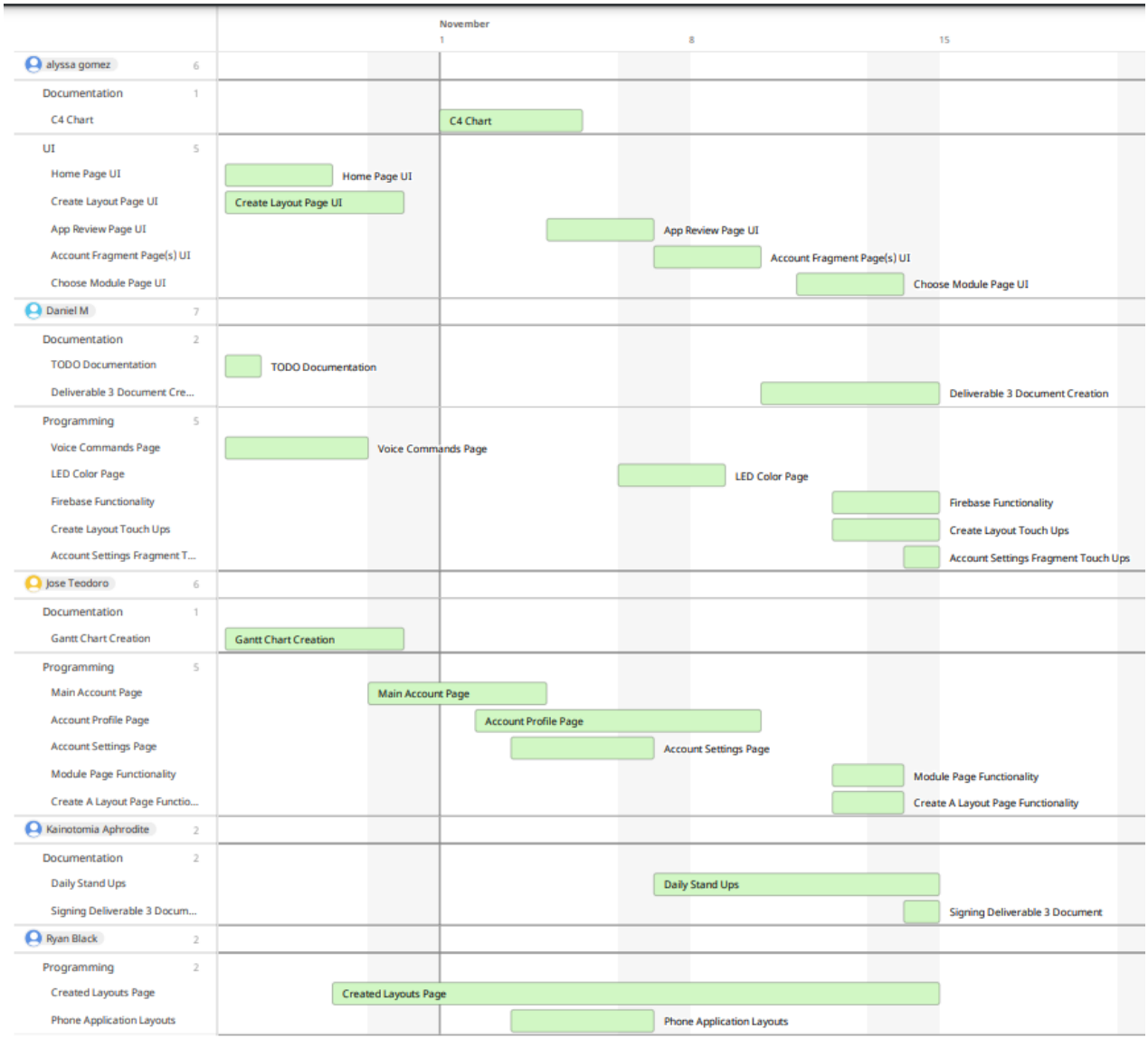
https://github.com/DanielMoore4875/Aphrodite

Sprint Goals

- Save layouts to database
- Save user information to database
- Refactor classes and remove unwanted code
- Landscape layouts

Full goals on:
https://trello.com/b/Bx0Fipkw

Gantt Chart (AirTable)



Daily Stand-Ups

Daily Stand-Ups

Nov 7, 2021

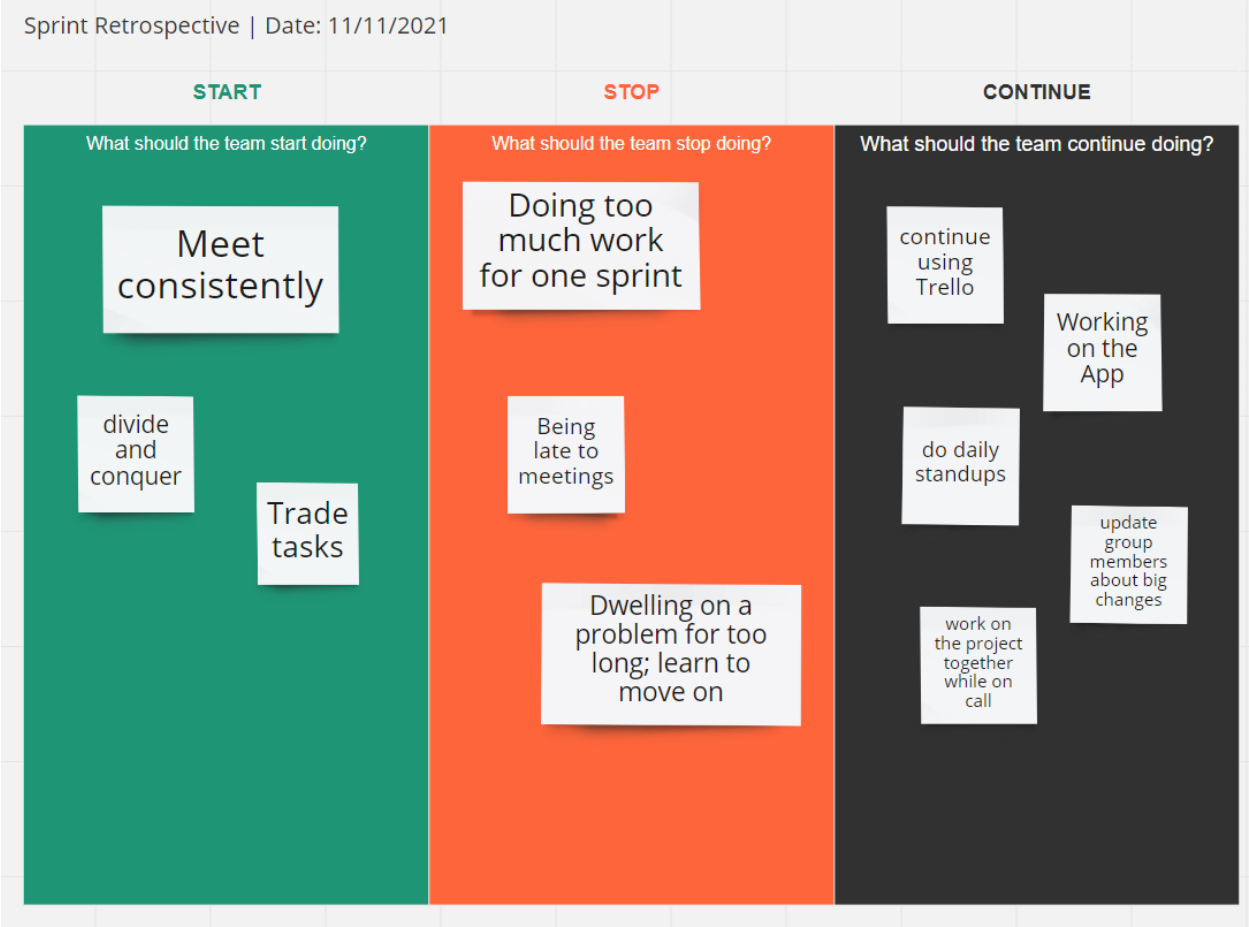
Alyssa Gomez	Daniel Moore	Jose Antonio Teodoro	Ryan Black
What did you do yesterday? • Made the Create Layout Page • Set up the page	**What did you do yesterday?** • Research about Realtime Database	**What did you do yesterday?** • Troubleshot the account fragments	**What did you do yesterday?** • Rearranged layout of buttons on page
What will you do today? • Add components • Connect Create Layout page to Home Page	**What will you do today?** • Fixed login and testing fragment transactions	**What will you do today?** • Make sure the main account fragment buttons work	**What will you do today?** • Begin to plan the redesign of the created layouts page
What(if any)is blocking your progress?	**What(if any)is blocking your progress?**	**What(if any)is blocking your progress?** • Trying to figure out why the program is not compiling properly.	**What(if any)is blocking your progress?**

Daily Stand-Ups

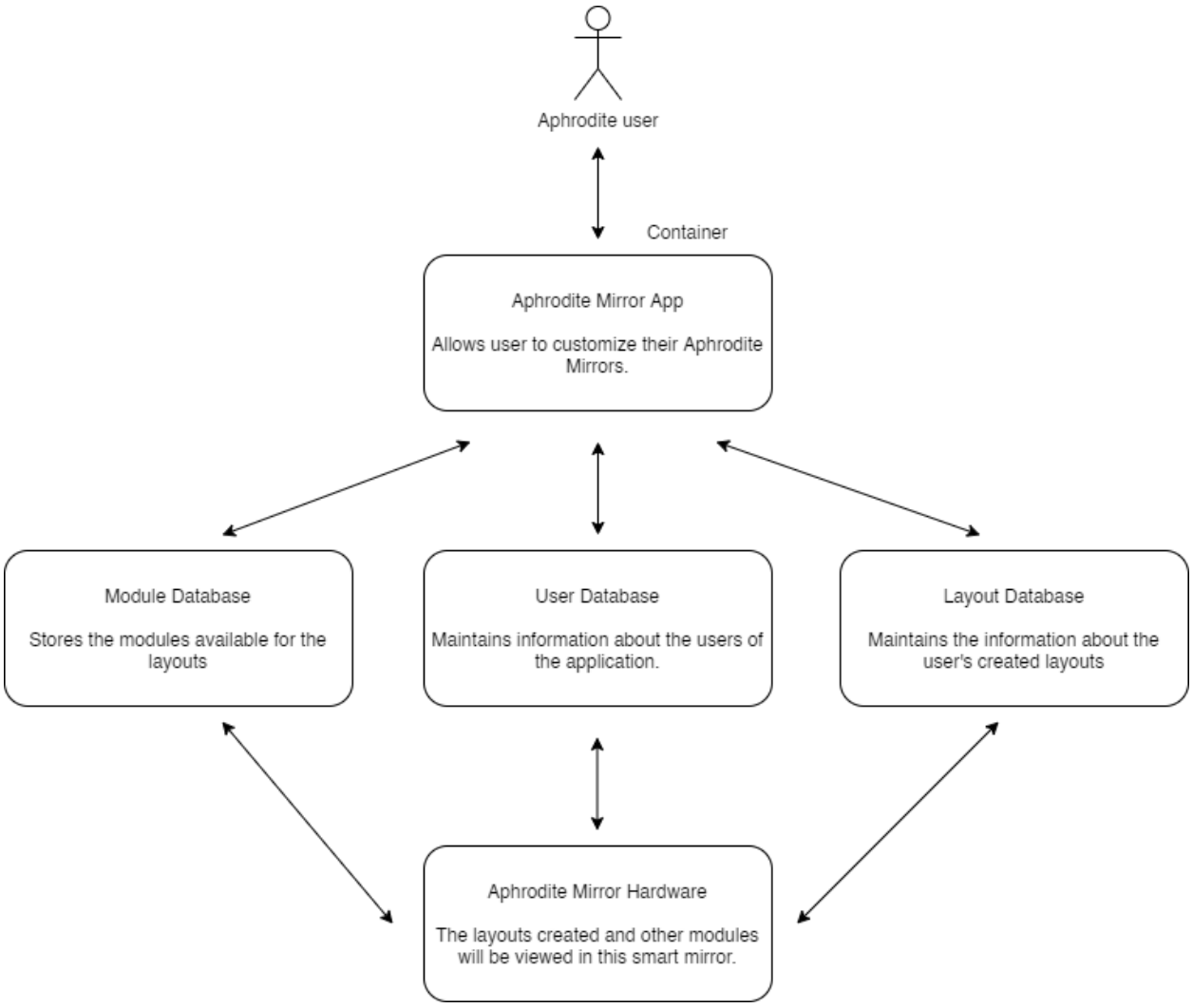
Nov 8, 2021

Alyssa Gomez	Daniel Moore	Jose Antonio Teodoro	Ryan Black
What did you do yesterday? • Add components • Connect Create Layout page to Home Page	**What did you do yesterday?** • Added remember login feature	**What did you do yesterday?** • Make sure the main account fragment buttons work	**What did you do yesterday?** • Planned the redesign of the created layouts page
What will you do today? • Make Create Layout page scrollview • Update text information	**What will you do today?** • Learn realtime database	**What will you do today?** • Try to implement toast and snack bars to specific buttons for the account fragments.	**What will you do today?** • Start researching more about firebase implementation with recyclerview and cardview
What(if any)is blocking your progress?	**What(if any)is blocking your progress?** • Database rules • Planning of page	**What(if any)is blocking your progress?**	**What(if any)is blocking your progress?**

Sprint Retrospective(Lucidspark)



C4 Model


Aphrodite user
Container
Aphrodite Mirror App
Allows user to customize their Aphrodite Mirrors.
Module Database
Stores the modules available for the layouts
User Database
Maintains information about the users of the application.
Layout Database
Maintains the information about the user's created layouts
Aphrodite Mirror Hardware
The layouts created and other modules will be viewed in this smart mirror.

Design Principles Used

UpdateDBNode.java

```
public class UpdateDBNode {

    /*
        Principle: Don't Repeat Yourself
            This class is the interface between the app and the database. Each method corresponds to
            some functionality that exists in the app that needs to store or retrieve data from the
            database.
     */
```

LEDColourFragment.java

```
/*
    Principle: Keep It Simple and Stupid
        Each slider updates the colour value in real time and has a separate listener to
        clearly see where the data is being saved. The button then calls a method that
        handles the saving of the data in the database.
 */

red.addOnChangeListener(((slider, value, fromUser) -> {
    int redI = (int) value;
    redTxt.setText("R: " + redI);
}));

green.addOnChangeListener((slider, value, fromUser) -> {
    int greenI = (int) value;
    greenTxt.setText("G: " + greenI);

});

blue.addOnChangeListener((slider, value, fromUser) -> {
    int blueI = (int) value;
    blueTxt.setText("B: " + blueI);

});
```

Design Patterns Used

MainActivity.java

```
/*
Pattern: Builder
    Using the Alert Dialog Builder to for the back button press
 */

private AlertDialog AskExit() {
    return new AlertDialog.Builder( context: this)
            .setTitle("Exit App")
            .setIcon(R.drawable.cancel_button)
            .setMessage("Are you sure you want to exit?")
            .setPositiveButton("Yes, exit app", (dialog, which) -> {
/                FirebaseAuth.getInstance().signOut();
                finish();

            })
            .setNegativeButton("No, stay in app", (dialog, which) -> dialog.dismiss())
            .create();
}
```

CreateLayoutFragment.java

```
/*
    Pattern: Adapter
        Using this array adapter to store strings to be used in the dropdown without changing the code
 */

ArrayAdapter<CharSequence> adapter = ArrayAdapter.createFromResource(getActivity().getBaseContext(), R.array.Layout_Values, android.R.layout.simple_spinner_item);
adapter.setDropDownViewResource(android.R.layout.simple_spinner_dropdown_item);
```

Coding Progress since Deliverable 2

Home Page

- Created a page called “Create Layout” page/fragment
- Connected Home page to "Create Layout" Page using the "Add" button in Home.
- Switches toggle whether the module is active and the spinner selects the location for that module
- Save button in Create Layout page saves the layout to the user’s id under the layouts node in the database

Layouts Page

- Created RecyclerView and CardView for the created layouts page
- Title and description boxes with hints for the user to create a layout from created layouts page
- Create layout button

Account Page

- Made the buttons on the page functional (About Us, Review and Settings)
- The About Us page consists of a short text about the project and text that displays the company email.
- The Review Page lets users give star reviews and also text reviews. The data will then be sent to the database once the Submit button is pressed. Made the page functional.
- The Settings Page contains a terminate account button and a logout button. This page will let users log out of the app and also have the choice to permanently terminate their Aphrodite account. After pressing on the terminate button, the user will then be brought to the login page. Made the page functional.

Voice Commands

- Will show the voice commands that are available on the mirror and an option to record a sound for them
- LED Colour button functional, it brings the user to the Customize LED Page.

LED Page

- Added LED Page to customize LED Colour
- Submit button will save the colour to the database under the current users id