Working Prototype Known Problems Report

HSV to RGB has some minor rounding. This is most noticeable while editing a color and switching between RGB and HSV modes.
- Location: ColorEditAcitvity
- Possible action for removal: Instead of getting color values from the seekbars pass actual values and only round for the seekbars.

Color names on the generated harmony palettes don't work. Shows "Color i" instead of a name.
- Location: PaletteInfoActivity
- Possible action for removal: Add a method specifically for the color harmonies to generate the color names when loading the color data into the recycler for the palette info activity.

Transparency is ignored. This can easily be seen when tapping on a transparent background on your image, the color will not be transparent.
- Location: ColorPickerFragment
- Possible action for removal: Either take in the pixel color including its transparency, get the combination of that transparent color and pure white, or at least inform the user that transparency is ignored.

When you click on the colors in the palettes they don't pass the right color to color info, instead using data from saved colors. Thus, if the user does not have as many saved colors as there are in a dummy palette, if the user clicks a color on the palette color list greater than the number of currently saved colors, the app will crash/restart.
- MySavedColorsRecyclerViewAdapter
- Possible action for removal: The fix is to revert the code of getColorClickListener to the code for accessing the color object data from the palette list instead of the code for fetching from the color database which was mistakenly added.

Palette names do not look pretty when they go to a second line.
- Location: PaletteInfoActivity
- Possible action for removal: Requires a simple format fix with readjusting the weights of all of the views on screen to accommodate for longer palette names. Could also put a character limit on palette name input in the save dialog.

Edit buttons on palette info pages have no functionality.
- Location: PaletteInfoActivity
- Possible action for removal: The fix is to add the method for palette name change (showSetNameDialog) from CustomDialog to the onClick listener for that button, passing in the palette data.

Search bar on "my palettes" has no functionality.
- Location: PaletteFragment
- Possible action for removal: Add a listener to the search bar and an ability to search the palette database. (Requires implementation of palettes)

If you try to see the info/edit/save a color before its name has been retrieved it believes its name is "...". (Not a common problem, only if the app is being slow)
- Location: ColorPickerFragment, ColorInfoActivity, ColorEditActivity
- Possible action for removal: Don't allow saving/switching activities until the name has been fully fetched.

While editing a color: Saving the "new" color when the color hasn't been edited saves whitespace for the name.
- Location: ColorEditActivity
- Possible action for removal: In the on click listener for saving, check if color name is " ".

On the save dialog you cannot save to a palette - all options send the color to saved colors.
- Location: ColorPickerFragment
- Possible action for removal: Requires implementation of palettes. Then when the user selects a saving to palette option, save the color to the palette instead of saved colors.

My palettes only contains dummy data.
- Location: PaletteFragment
- Possible action for removal: Requires implementation of a palette database.

When there is an "empty" image the add image button can permanently disappear when clicking/dragging. (It comes back if you switch fragments)
- Location: ColorPickerFragment
- Possible action for removal: Check for empty image state, if empty show the button.

The color name will load as "Your Color" when you do not have access to the internet.
- Location: ColorPickerFragment, ColorInfoActivity, ColorEditActivity
- Possible action for removal: Indicate to the user that they do not have internet and therefore cannot get the color names. Possibly fetch the color names of previously saved colors once internet connection resumes.

Text may squish on different devices and screen sizes.
- Location: All fragments/activities
- Possible action for removal: Adjust layouts/weights. Possibly decrease text sizes.

When camera permission is denied, trying to take a picture crashes the app.
- Location: ColorPickerFragment
- Possible action for removal: The permission function just needs to be called again when it is needed instead of only when the app starts, this was overlooked in testing.

When gallery permission is denied, you can still upload a picture from the gallery.
- Location: ColorPickerFragment
- Possible action for removal: Unsure

When clicking on the save button in ColorPickerFragment, canceling the save dialog will prevent saving the color.
- Location: ColorPickerFragment
- Possible action for removal: When canceling the save dialog, reset the save button.

List of Suggested User Stories/Acceptance Criteria
for Acceptance Test

Sprint 1 User Story 2:

As a traditional artist, I want to be able to use my camera and gallery to observe colors of what I am drawing, and have easier reference of what colors I need.

Sprint 2 User Story 1:

As a general user, I want to be able to view the photo I've taken on the main color picking page to select colors from

Sprint 2 User Story 2:

As a general user, I want to be able to tap on the color I want to pick anywhere the image is on the screen, whether using the live camera or a static image.

Note: Live camera interaction was moved to the backlog. The user can either take a picture or browse pre-existing pictures in their gallery.

Sprint 2 User Story 3:

As a digital artist, I want to see the RGB/HEX/HSV values of a color that I pick.

Sprint 3 User Story 1:

As a long-term user, I want to be able to save colors for later usage in a library directly on the app.

Sprint 3 User Story 3:

As a frontend designer, I want the ability to export RGB, Hex, and HSV values from the app to more easily use these colors in my personal projects.