GitHub Username: briancap

Reflection

Description

Reflection prompts users daily to fill out a quick response to a questions designed for introspection and reflection. Reflection will store responses and occasionally revisit old entries and track how responses change over time.

Intended User

Reflection is for anyone who enjoys journaling and needs a portable app for recording their swirling thoughts.

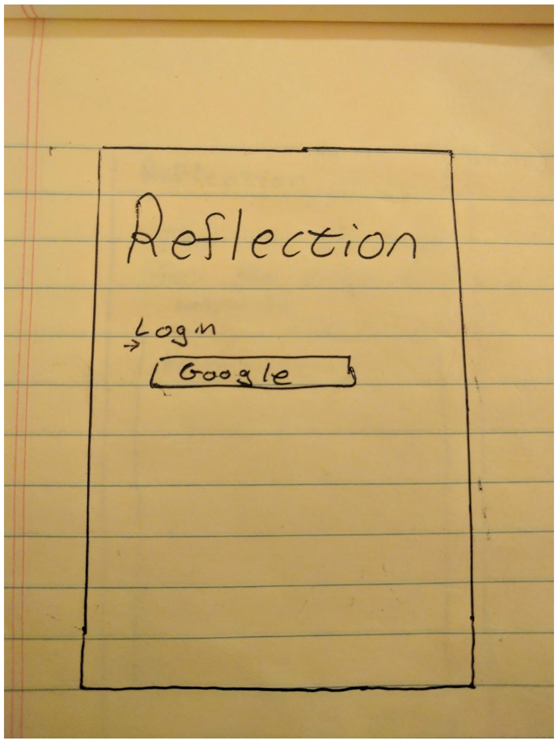
Features

- Saves and displays journal entries
- Guides users with thought questions. Users can also provide their own thought questions

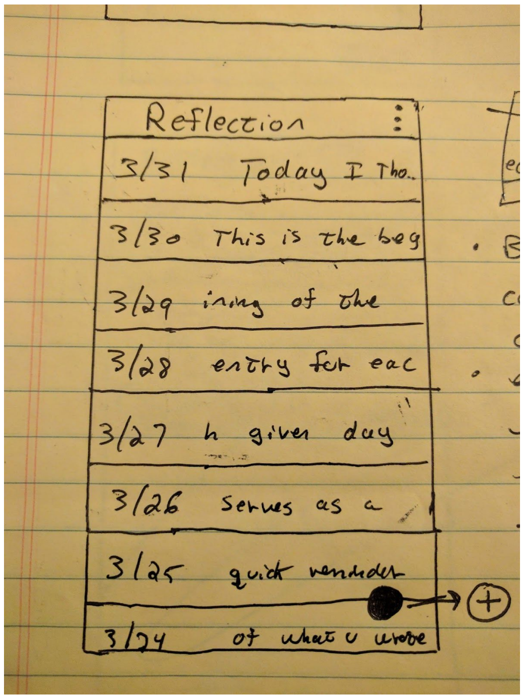
User Interface Mocks

These can be created by hand (take a photo of your drawings and insert them in this flow), or using a program like Photoshop or Balsamiq.

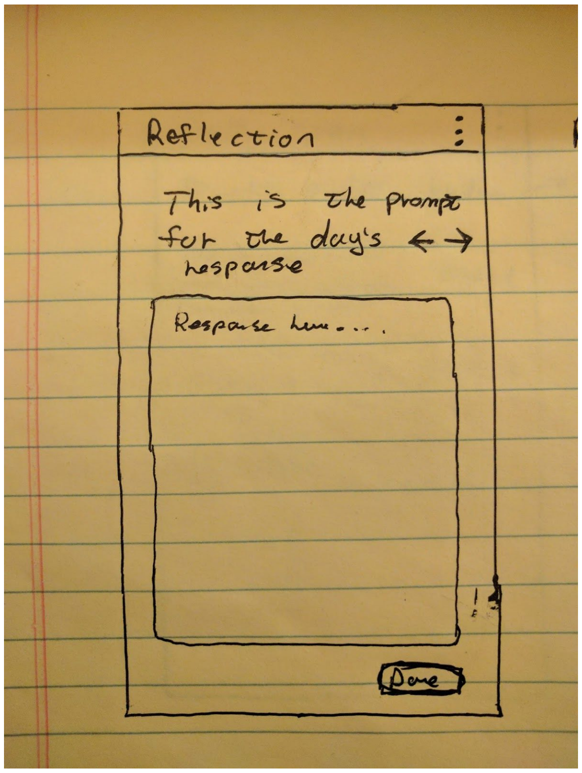
Sign In
Simple sign in screen that integrates Google Auth. Google Auth is the only way to sign in so may need a link to create a Google account if a user doesn't have one.
In version 1.0 this won't do much but will provide the ability to see the same content on multiple devices in subsequent versions



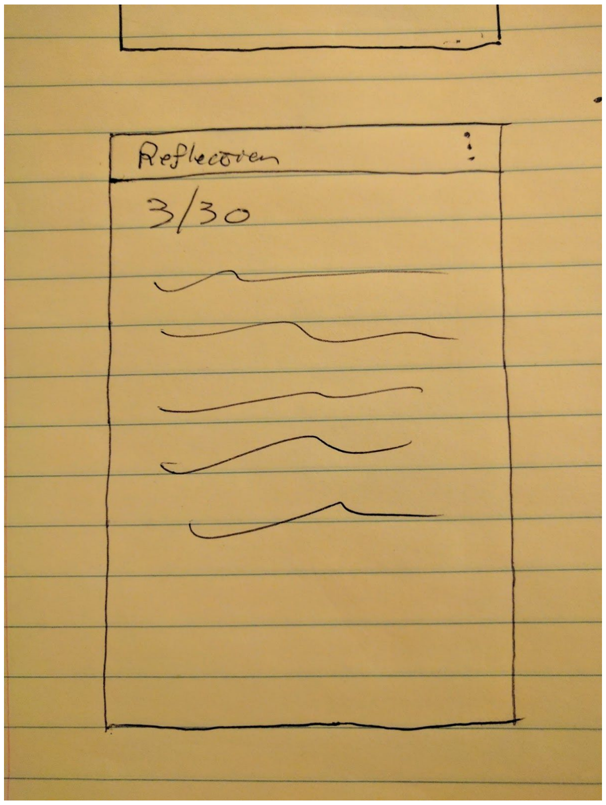
List of Reflections
List of previous reflects organized by date. Entries will be color coded according to the category of reflection (free thought, response to a prompt, user defined prompt). The background of the list item will provide the color coding. List items will contain the date of the reflection as well as the first few words of the reflection. FAB will launch new reflection activity



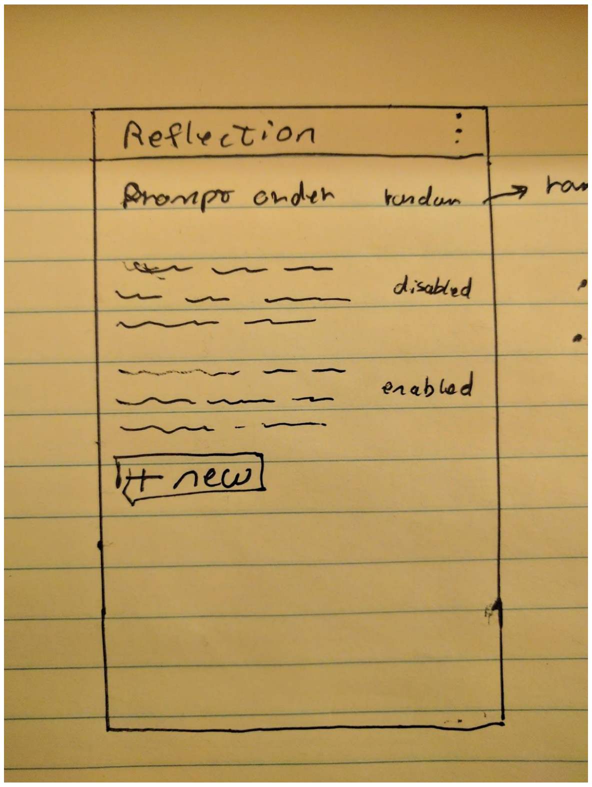
New Reflection
Title of the entry will be the category/prompt for the user. User can switch between category with left and right arrows. The body will be text entry and submit button below text.



View Reflection
Screen shows a reflection from a previous day. Shows the date, prompt, and body of the reflection.



Prompt/Settings
Lets the user decide if they want the app to randomly give them a new prompt each day or if the prompts should be displayed in order. List of prompts that the user can activate/deactivate based on preferences. User can also add custom prompts through the plus at the bottom of the list. User navigates to this screen through the settings button from any of the above activities. This will not be a settings activity because it will interact with the database instead of shared preferences. Letting the user add prompts means prompts cannot be stored in static xml arrays and instead will be stored in a database table



Key Considerations

How will your app handle data persistence?

Reflection will have a custom sqlite database and content provider for storing and retrieving reflection entries. User prompts and thought questions will be declared in a separate database table.

Describe any corner cases in the UX.

UI is pretty simple and should have no edge cases. Reflection has a reflection entry screen, reflection list, and settings screen to add/remove/disable prompts

Describe any libraries you'll be using and share your reasoning for including them.

Reflection will use Butter Knife for view binding and Timber for logging.

Describe how you will implement Google Play Services.

Reflection will use com.google.gms.ads for advertising and com.google.android.gms.auth for google account sign-in

Next Steps: Required Tasks

This is the section where you can take the main features of your app (declared above) and decompose them into tangible technical tasks that you can complete incrementally until you have a finished app.

Task 1: Project Setup

1. Create project and initialize git repo
2. Add libraries and Google Play Services to build.gradle
3. Create all Classes in below folder structure
    a. java/com/example/brian/reflection (Activity, Fragment, Adapter)
        i. MainActivity, ReflectionListFragment, ReflectionListAdapter
        ii. NewRelectionActivity, NewReflectionFragment
        iii. SignInActivity
        iv. ViewReflection
        v. PromptsAndSettings, PromptAdapter
        vi. DataStructure - Folder
            1. Contract
            2. Database
            3. Provider
    b. res/layout
        i. activity_main, fragment_reflection_list, list_item_reflections
        ii. activity_new_reflection, fragment_new_reflection
        iii. activity_sign_in
        iv. activity_view_reflection
        v. activity_prompts_and_settings, list_item_prompt

Task 2: Build Database, Contract, and Content Provider

Starting with the database because the data structure of the app will drive what data elements the activities contain. There will be a Reflections table and a prompts table.

1. Build Contract.java

2. Build Databse.java
3. Build Provider.java

Task 3: Outline UI for Each Activity/Fragment and Build Adapters

1. Outline the layouts
    a. MainActivity, ReflectionListFragment
    b. NewRelectionActivity, NewReflectionFragment
    c. SignInActivity
    d. ViewReflection
    e. PromptsAndSettings
2. Build Necessary Adapters
    a. ReflectionListAdapter
    b. PromptAdapter

Task 4: Add Prompts to the Database Test the Provider

1. Determine prompts and add them to the prompts table
    a. Quotes
    b. Thought questions
    c. Open ended response
    d. Specific goals for the day/week/month

Task 5: Implement GPS and Test

1. Implement Ads and Signin from Google Play Services.
2. Test everything and submit project

Add as many tasks as you need to complete your app.

Submission Instructions

1. After you’ve completed all the sections, download this document as a PDF [File → Download as PDF]
2. Create a new GitHub repo for the capstone. Name it “**Capstone Project**”
3. Add this document to your repo. Make sure it’s named “**Capstone_Stage1.pdf**”