Functional Requirements:
- Tutorial that teaches the user how to take full advantage of the app's features
- Modules
    - Can optionally be rated for helpfulness after completing a module
        - These ratings can be optionally submitted to us by email
    - Guided Breathing
        - Visual cues
        - Haptic cues
        - Audio cues
        - Settings for breath time, haptic strength, and audio noise
        - Can enable and disable haptics and audio
    - Self-Reflection
        - List of questions with audio recordings of each
            - Research cognitive restructuring for evidence-based helpful questions to ask
        - Can create new questions without audio support
        - Can enable randomization of questions (order and/or number)
        - Vision enabled: Prompted with text fields to answer each question
        - Audio enabled: Waits for the user to finish speaking/thinking and indicate they're ready for the next question
            - Dictation of answers (probably handled by the keyboard/mic API)
        - Bounce key setting to prevent accidentally skipping a question by tapping twice too quickly
        - Answers stored locally on device for the user to refer back to later
        - Minimal chatbot functionality
    - Mental Exercises
        - List of tasks (text descriptions) that can be chosen from
            - Find everything [color] in the room
            - Think of [number] of [adjective] things
            - Engaging the senses
                - A "Grounding Technique": 5 things you can see, 4 things you can hear, 3 things you can touch, 2 things you can smell, 1 thing you can taste
            - Try to think of activities that are more inward than outward so that it could be used in public, where we wouldn't want to draw attention to being surrounded by people who might be watching
    - App Exercises
        - List of activities that can be chosen from
            - Keep your finger inside the moving circle
            - Tap the circle when it appears
            - Will seek further ideas from survey responses
    - Haptic Heartbeat Simulator
        - Undecided if it will attempt to measure and match the user's, or just simulate a resting heart rate for them to try to synchronize with

- Accessibility features
    - Works with screen-readers
    - Text can be magnified up to 200%
    - High-contrast mode
    - Rescale UI based on the device
        - Possibly allow the user to rescale it themselves as an option
- Settings
    - App-wide volume
        - Should also feature a volume slider to complement the hardware's volume buttons (important for accessibility as well)
    - App-wide haptic strength
    - Dark-mode
    - Setting to start with menu or automatically launch module sequence
        - Can select which module sequence to automatically launch
- Custom notifications for a certain time with a reminder to complete a module sequence
- SQLite database to store and retrieve data and settings
- Can be rotated into landscape mode

Flow Control (every functional requirement should be reachable):
- Launch Screen
    - On the first use, launches the tutorial and prompts accessibility settings choices
        - Take a look into storing the SQLite file both locally and as a backup on Google Drive so that the user never has to repeat the first use screen or recreate module sequences, past self-reflections, etc. Losing data is very, very bad for this app.
        - Ensure that when the local cache is cleared, the backup could then be grabbed from the Drive automatically(?), just in case the local file is either lost or deleted.
    - On subsequent launches, the screen will be split into two giant buttons (most likely left and right, although maybe it changes depending on the device's orientation to maximize the surface area). One is for an emergency mode, and the other is for general use.
        - Emergency mode launches into pre-selected module sequence
            - Once the sequence is finished, the user has the choice of starting any of their other module sequences (including the one just completed) or entering general use mode.
        - General use mode is more of a traditional app interface, with options for settings, launching modules, creating module sequences, etc.
- General use mode screen is the gateway to the rest of the app
    - Settings
        - Both app and accessibility settings
        - Can submit module ratings

- Modules
  - Tradeoffs to consider. Could have several of the most important features available on this screen, which makes them simpler to get to but the screen is busier. Or, it's just a button to take you to a dedicated module screen so that each page is simpler but the app navigation as a whole has more layers / is more complex.
  - Leaning towards a simple button with its own module screen.
    - Individual modules
      - Can activate it
      - History of when the module has been taken, can view past responses
      - Settings (might be another layer, or might not, just needs to be consistent across all modules)
    - Module sequences
      - Create
      - Edit
      - Delete
    - When modules and sequences are completed, leads to a screen to optionally rate how helpful it was
- Notifications
  - List of currently scheduled notifications
  - Option to create a new scheduled notification
  - When a system notification is clicked, it launches immediately into its corresponding module sequence
- Help
  - Can relaunch tutorial
  - FAQs for anyone who doesn't want to go through an interactive tutorial