Test4Time

Feasibility Study

David Fletcher
Halston Raddatz
Alan Bunch
Trevor Chappell

Languages

The application will be written in primarily Java 1.8. By using the latest version of Java we will have access to the latest features that the language provides. The latest version of Java also allows the application to being maintained the longest.

Frameworks / Libraries

- Android (4.4 - 7.0.1):
    - The application will be compatible for Android versions from 4.4 up to 7.0.1. This will allow users with without the latest updates to still be able to access the application without difficulty.

- Fitbit API:
    - The application will communicate with the Fitbit API to retrieve the data from the server. The data that will be returned will be in the form of JSON objects. These JSON objects can then be interpreted to communicate with our application.

- Jackson API:
    - The Jackson API is a Java based library that serializes JSON to Java objects. This is how our application will take the retrieved data from the Fitbit API and communicate with our application in the Java language.

Rationale for approach

- Android structure
    - iOS is extremely locked down in comparison to Android. Our app will be able to lock out specific applications - a function that, as far as we can tell, is extremely complicated or even impossible on iOS
- Accessibility of development in Android
    - The process for creating an application in iOS requires development to take place with direct access to Apple products. Additionally, Apple requires a much more rigorous development application than Google.
- Knowledge of language
    - Our team has little to no experience with iOS development. Alternatively, our group has significant experience with Java. This made the decision to go the Android route very simple. By utilizing our knowledge in Java with Android, we are able to deliver a product much quicker with far less obstacles.

- Fitbit's popularity & API
    - Fitbit, at the time of the creation of this document, has the largest market share of any fitness tracker. This means that they are the safest choice for the focus of our application. Additionally, their API is moderately easy to access - only requiring a developer account that can be made and verified relatively quickly.

Feature Requirements

- Admin/Parent menu
    - Parent/guardian has to be able to have the ability to set the step requirement for their child, we will incorporate a menu that will appear when you swipe down on the phone (with the app open). This feature will use a passcode to access a parental menu that can be used to add, alter, remove, or temporarily ignore the step requirement.
- App locking
    - Our initial plan is to launch a background service that has a list of apps to block, where the service will launch our app as soon as it detects an app from that list has been launched. However, if the user has completed the required number of steps, applications will not be blocked for a time period set by the admin.
- Step requirement
    - Using Fitbit's proprietary API, we will sync steps allotted from the user's fitness band. They will be displayed in the app and once the Admin-set step requirement is reached, the app locking function of the app will disengage and allow the user access to the previously locked applications.
- Functional UI
    - Interface will be attractive and functional at all resolutions, on varying devices, incorporating design elements from Google's material guidelines (https://material.io/guidelines/).

Code Repository Organization

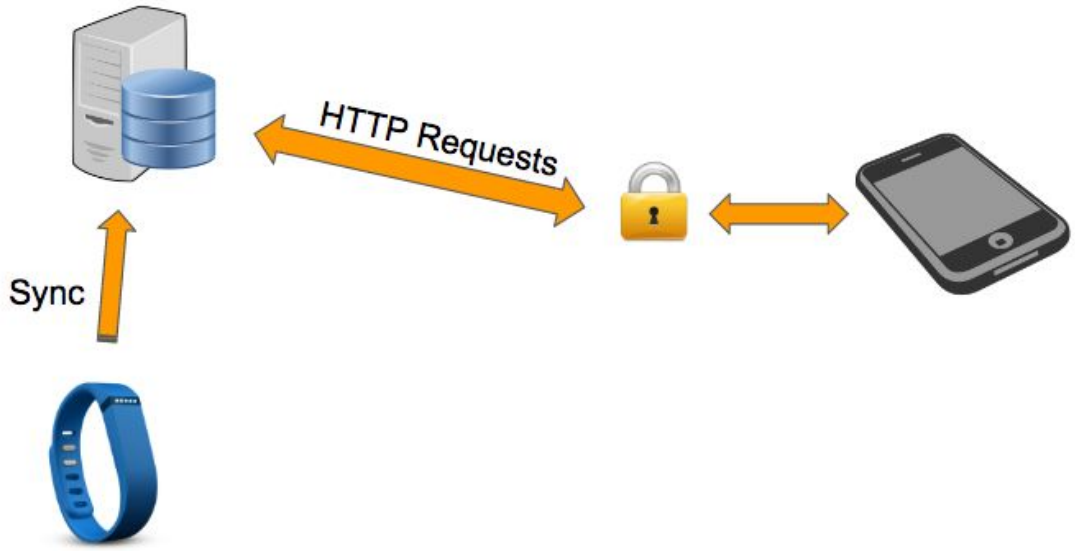
- Issues:
    - create new issues for feature discussion, feature implementation, enhancements, or bugs.
- Branching:
    - Each issue should be made into its own branch if it would require more than 1 significant commit (and commits should be small and focused). This way we can more easily detect and manage merge conflicts.
- Code Reviews:

    - Code reviews by every other member of the repository; comment directly on pull requests, cite lines, etc. when a pull request is submitted
- Merging:
    - Collaborate with all members before merging pull requests.
- Collaboration:
    - Collaborators should comment on issues directly for discussion related to the issue, but also added to the Github issue when a final decision or suggestion has been reached, so that the communication is visibly tied to the code

Database Information

There should be no database required. The data is stored on the Fitbit servers and synced with our application.

System Organization

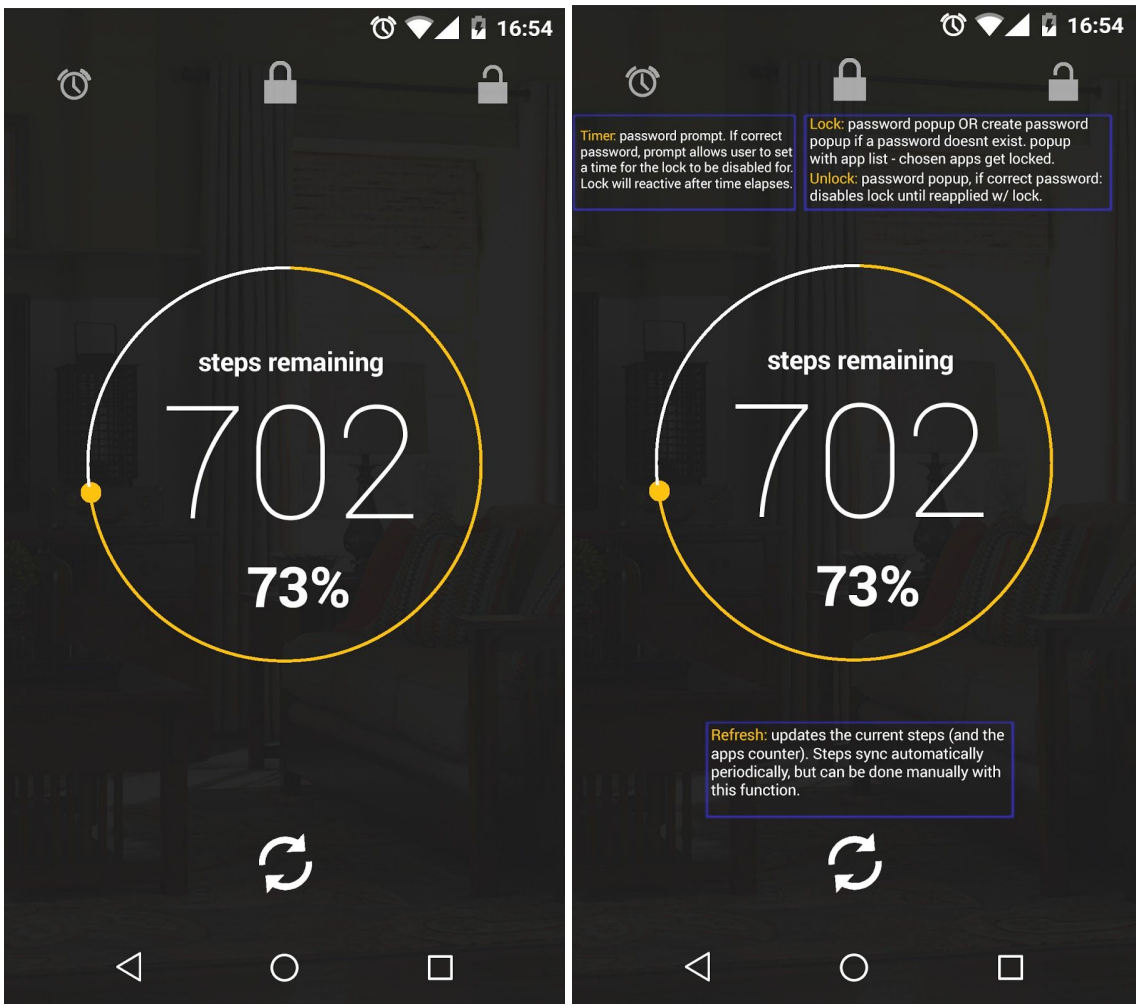


Division of Labor

- David Fletcher:
    - David will primarily be focused on the service that communicates the application and the Fitbit API. To do this, we will have to utilize the Jackson library in our application to successfully interpret the data.

- Halston Raddatz:
    - Halston will primarily be focused on working with design and construction of the user interface. Halston will also be working with the admin panel that would include features such as a password prompt, lock feature, unlock feature, and refresh the data feature.
- Alan Bunch:
    - Alan will be working on implementing the background service to block the applications. Aside from this task, Alan will also collaborate on working on the user interface.
- Trevor Chappell:
    - Trevor will be designing a built in step counter as an add on feature. He will also be contributing to the backend data management that would allow the application to have separate user accounts.

User Interface Mockup



Teamwork Policies

- Bi-weekly Meetings:
    - Our goal is to meet the sponsor weekly. Usually these meetings are held on Thursdays at 12:00 at the downtown campus. Understandably meeting on a weekly basis may not always be feasible so our realistic goal is once every two weeks at the latest.
- Updated Journals on Github:
    - By keeping our journals on Github updated we can keep track of what other team members have been working on. This can assist us in not doing the same development work as someone else.

- Maintain active communication:
    - Active communication among the team members ensures that we are all collaborating effectively toward our end goal for the application. Active communication with our sponsor means that we can receive feedback and adjust our priorities and vision as the project moves along.