GitHub Username: teddoll

Interval Trainer

Description

An interval timer with notifications for use in training. Enter, save, and edit training intervals with intuitive interface. Press run and let your device notify you when your intervals are complete.

Intended User

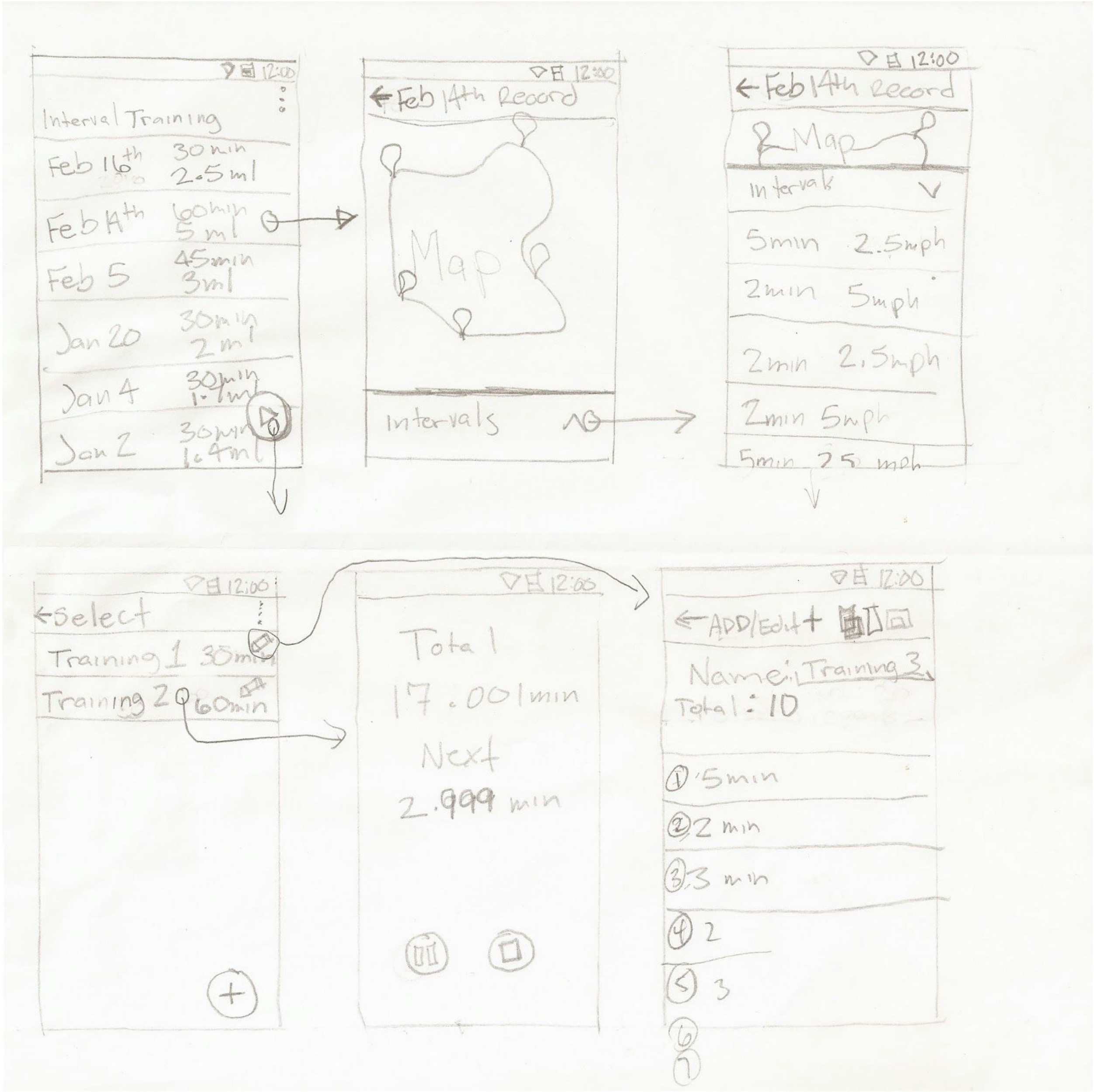
Runners, walkers, and fitness enthusiasts.

Features

- Input, save, and edit intervals.
- Timer with interval notification.
- Map and split times when training is complete.
- VIew previous training sessions.

User Interface Mocks

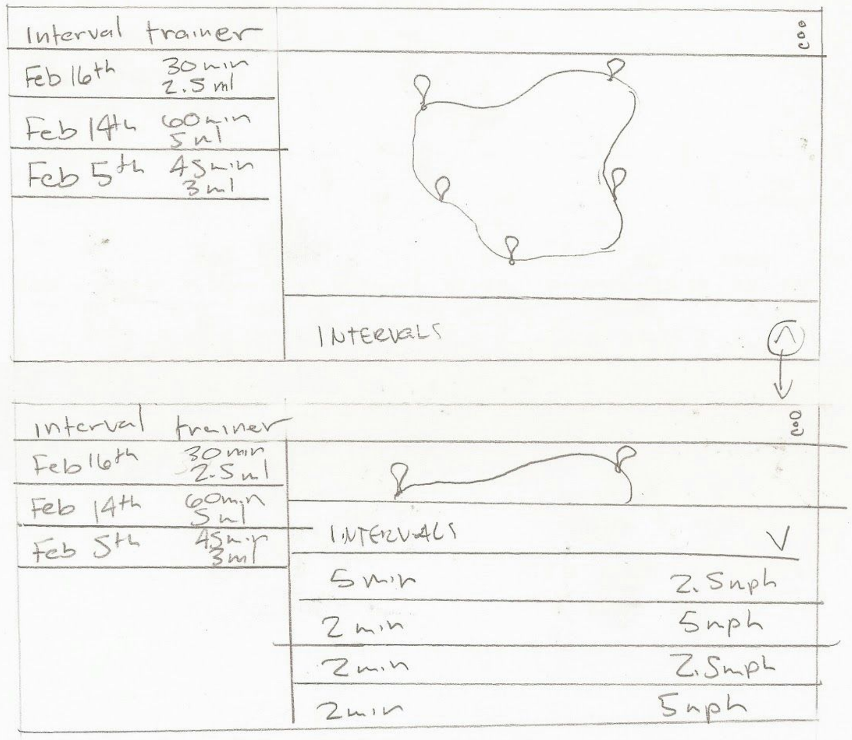
Mobile (phone) mockup and flow



Mobile flow.
Starting top right. Landing page of existing training.
Top middle. Map View of single training.
Top right. List if split times.
Bottom Left. Interval training selection.
Bottom Middle. Interval training session (tracking location and time)

Bottom right. Edit/Add new interval training splits.

Tablet Interval training sessions view.



Top. Map view.
Bottom. List Split view.

Key Considerations

How will your app handle data persistence?

The app will include a content provider for storing data on the device.

Describe any corner cases in the UX.

In the case that location service are not available or disabled for tracking interval training session need to display warning to user to enable or block start until gained a location fix.
User will not be able to back out of tracking interval training session while it is running. App will maintain a persistent notification for when the user leaves the app.

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

In addition to the Android app compat and Google play service libraries, the app will also be using Timber, for debug logging.

Next Steps: Required Tasks

Task 1: Project Setup

Create new project in Android Studio.

- Setup all dependencies in app build.gradle
- Create new signing keystore.
(http://developer.android.com/tools/publishing/app-signing.html#studio)

Task 2: Implement Data model, contract and provider.

- Create data model for interval training set.
- Create data model for the recorded interval training session.
- Create the contract and provider.

Task 3: Create Interval tracking Service

- Create logic for recording location and times based on given Interval training set.
- Create logic to create/update notifications based on the given set.
- Create logic to save session to content provider when session is complete.

Task 4: Create Interval tracking Activity

- Create layouts
- Create logic to wait or notify of location errors.
- Create logic to start Interval Tracking service and bind to it.
- Create logic to pause or quit session.

Task 5: Create Landing Page and Detail UI

- Create layouts.
- Create LandingActivity and SessionDetailActivity.
- Create Fragment for displaying a list of recorded interval training sessions.
- Create fragment for displaying map and list of single session (detail page)
- Set up tablet display logic.

Task 6: Create Interval Activity Selection UI

- Create layouts.
- Create SelectionActivity.
- Create Fragment for displaying a list of interval training sets.

Task 7: Create Add/Edit Interval Set UI

- Create layouts
- Create Activity
- Create fragment for displaying list of intervals.
- Create dialog fragments for copy/paste actions

Task 8: Create daily tip service

- Create App engine Api to serve up daily tips.
- Create IntentService to fetch daily tip and store in SharedPreferences.
- Create BroadcastReceiver for daily alarm to kick-off fetch service every 24hrs.
- Tie LandingActivity to display tip onResume if not previously displayed in last 24hrs.

Task 9: Create Widget to display last session with distance and daily tip.

- Create widget provider
- Create intentService to pull data from provider and update widget.