GitHub Username: GHSam

Habit Tracker

Description

Habit Tracker helps you track your progress creating/breaking habits and reaching goals in a simple and clear interface.

You can set reminders for certain times and/or locations and easily mark your progress from the reminder without ever having to open the app.

The app graphs your progress as well as letting you set goals you're trying to reach.

Intended User

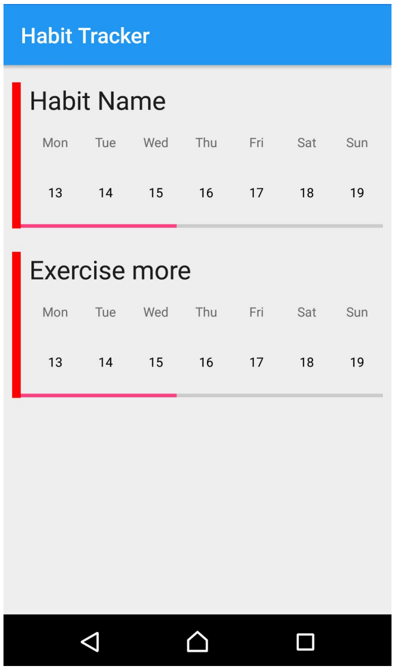
People that are trying to create new habits (e.g. people who want to make a habit of exercising regularly or practicing guitar, etc) or break bad habits and log their progress while doing it.

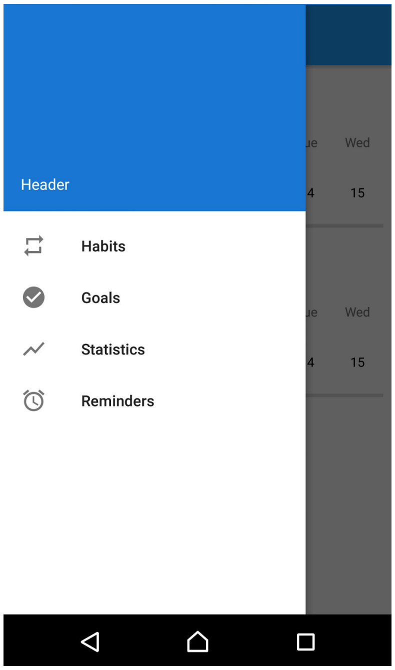
Features

- Reminder notifications based on time and/or location
- Add reminders/habits calendar
- Ability to only show habits on certain days/locations
- Progress graphs and statistics
- Widget to show progress and allow easy checking off of habits
- Ability to set goals and see progress towards them
- Support yes/no/skip habits
- Support time based habits (entering how long you've done X)
- A timer for time based habits so can press start/stop and it logs all the time in between to the habits log
- Ability to share progress on habits and goals
- Quote of the day from the They Said So API

User Interface Mocks

Main View

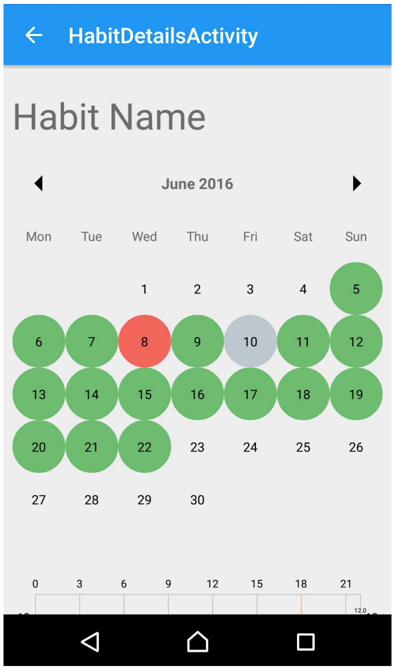




The main screen shows a list of habits along with a week calendar view to allow easy logging. Clicking on a habit will go to the habit details page.

The main view also has a navigation drawer to navigate to the global statistics page, goals page and reminders page.

Habit Details View

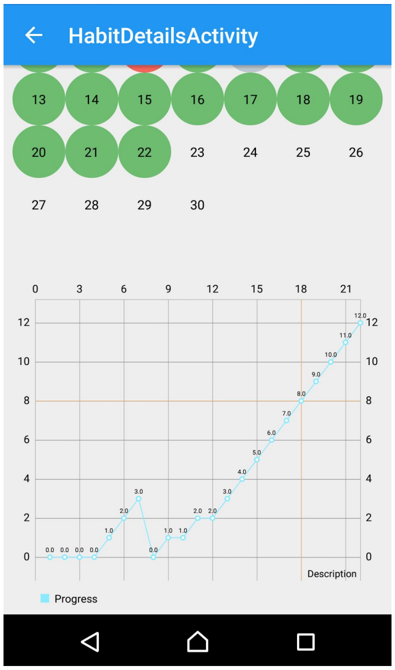


HabitDetailsActivity

Habit Name

June 2016

Mon	Tue	Wed	Thu	Fri	Sat	Sun
		1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30			

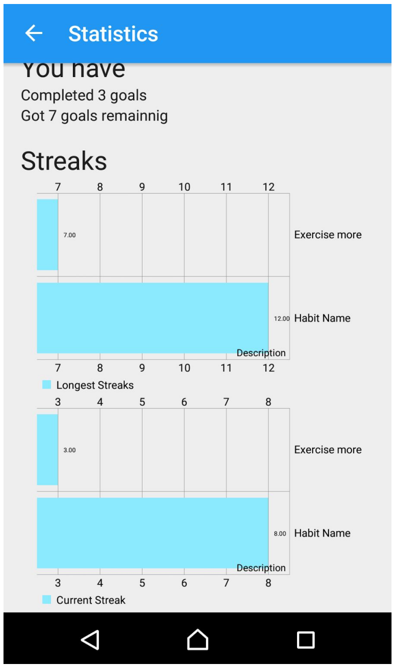


HabitDetailsActivity

13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30			

The detailed view shows a full calendar starting from when the habit was created until now showing current progress. Below are statistics including a graph of progress, longest streak and current streak and progress on any set goals.

Global Statistics View



Statistics

You have

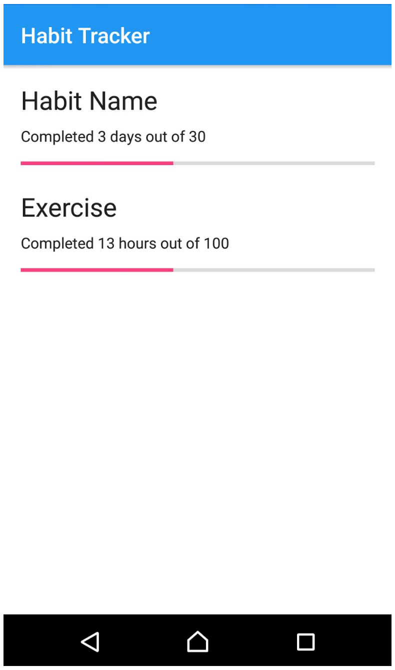
Completed 3 goals

Got 7 goals remainnig

Streaks

Shows various statistics like remaining goals as well as progress graphs for all habits.

Goals View



Habit Tracker

Habit Name

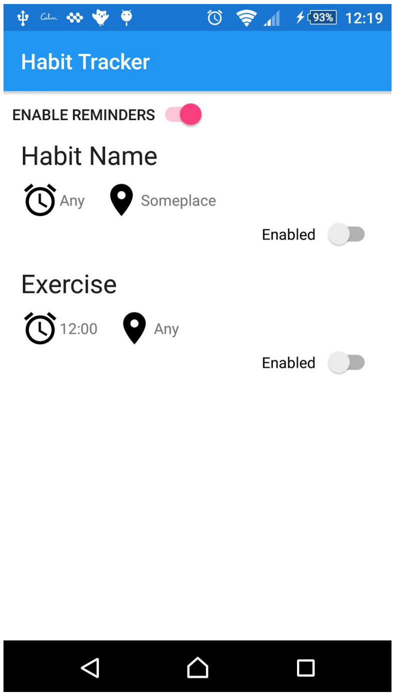
Completed 3 days out of 30

Exercise

Completed 13 hours out of 100

Shows a list of current goals for all habits and any progress made on them.

Reminders View



Habit Tracker

ENABLE REMINDERS

Habit Name

Any Someplace

Enabled

Exercise

12:00 Any

Enabled

Shows list of current reminders with their time and location to allow easy removal/disabling/editing.

Also has a toggle to easily turn all notifications on/off.

Key Considerations

How will your app handle data persistence?

The data will be stored in a local SQLite database and exposed via a content provider.

Describe any corner cases in the UX.

- Habit names need to be unique to avoid confusion.
- Should prevent scrolling the calendar view after the current date or before the habit was created in the habit details view.

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

- Picasso for image loading/caching as it's efficient and it's the library used in the Udacity course so I'm already familiar with the API
- MPAndroidChart for graphs because it supports both pie and bar charts and looks to be quite popular
- Butterknife for view binding, again because already used it in the Udacity course so familiar with the API
- Material Calendar View for the calendar views instead of the built in CalendarView widget as according to the docs you can easily draw backgrounds for the days which would allow adding marks to days

Next Steps: Required Tasks

Task 1: Project Setup

- Create the project and do the initial setup (naming, colours, etc.)
- Add all the required libraries
- Create a git repository

Task 2: Implement UI for Each Activity and Fragment

- Build UI for main view
- Build UI for habit details view
- Build UI for the goals view
- Build UI for statistics view
- Build UI for reminders view

Task 3: Implement the Content Provider & DB

- Create the SQLite DB contract and open helper
- Create the content provider
- Add cursor loaders to the views to load the data
- Implement persisting data from the views via the content provider

Task 4: Implement reminder notifications

- Build geofence selection view / input
- Implement geofencing notifications
- Implement timer based notifications

Task 5: Implement widget

- Create widget layout
- Create the backend code for the widget
- Add widget screenshots

Task 6: Implement quote of the day

- Create sync adapter to fetch the quote of the day every 24 hrs
- Show quote and add an attribution link to match quote API requirements

Task 7: Final polish

- Check all strings are in strings.xml
- Check for any accessibility issues: make sure contentDescriptions are set where needed and d-pad navigation
- Verify layout works in RTL mode