Harlem Sky Milestone 4: Implementation

Amanda Chen (aec255), Carrie Huang (ch646), Boonakij Palipatana (bap93), Martín Eizayaga (me393), Anna Tedijanto (ajt232)

7.1 Table of Contents

In your documents folder, you are required to include a table of contents for your submission. This should be a Markdown file named m4-toc.md. In your table of contents you should organize all of your submission materials and provide proper links to these resources. This is the file that we will use to start grading your submission. Take care to organize it well and include all of your materials, including evidence of your design process.

7.2 Sharing Effort & Accountability

Amanda Chen

- Forms to create new record and new tracker

Carrie Huang

- Displaying records on the newsfeed
- Displaying existing trackers on “My Trackers” page
- Styling
- Bottom Nav Bar

Boonakij Palipatana

- Adding widgets
- Create new record and edit record
- Styling

Martín Eizayaga

- Edit and delete a record
- Goal Page Visualization
- Editing Goals
- Created Testing Json

Anna Tedijanto

- Search by keywords
- Filter by tags
- Integrating goals into the newsfeed and new goal page
- Routing

7.3 Scenario-based Design

We chose to maintain most of the vocabulary that we had come up with in Milestone 3, but we did also reduce any extraneous vocabulary that we did not end up needing.

- ***Tracker*** is like a “notebook” for each specific thing a user wants to track
    - A user can have multiple *trackers*
    - For instance one for traveling, one for spending, etc.
- ***Records*** are like ***“entries”*** in a specific tracker
    - One tracker can have a LOT of records/entries
    - What each record/entry records is dependant on the user - *customizable*
    - Have a default entry format, but allow user to add more fields

What we chose to remove is **Template**, which we found to be unnecessary given our scenarios. Originally, we designed our template to afford users the ability to add widgets such as weather, video, etc. However, for our three target audiences (Tiffany for travel, Harry for habits, and Sasha for spending), adding additional notes and images were prioritized.

In our previous milestone, we also came up with the idea to toggle between a newsfeed and calendar view. For this milestone, we decided not to implement the calendar view given that users would only use the calendar view to quickly see past progress, however the Newsfeed allows them to scroll down to the given date and see that information. A calendar may also be used for tracking goals, but given that we decided to create a specific “Goals” page, we found that a calendar was just an alternate way of displaying the exact same information that the newsfeed affords.

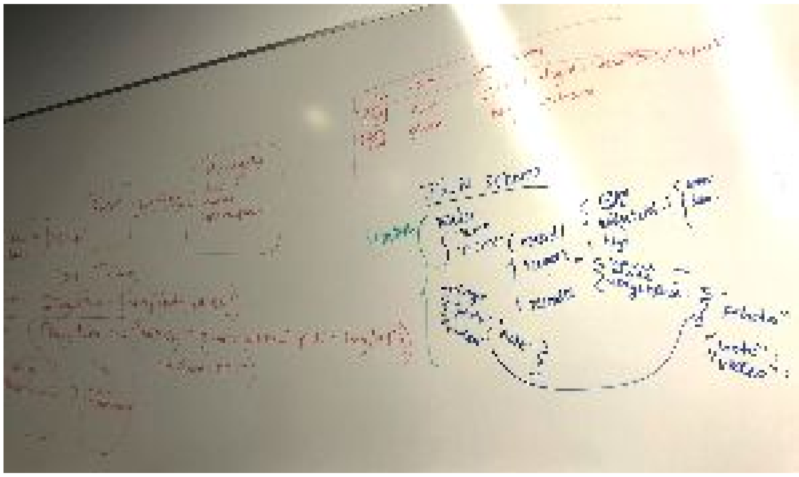
Going back to the scenarios written in Milestone 3 before moving on to implementation, we decided to assess whether or not the tasks were able to be completed given our designs. In one scenario with Traveling Tiffany, she wanted to look up a specific event from the past that she had tagged. Thus, we decided to prioritize searching by event name and also filtering by tag so that users do not have to scroll down potentially hundreds of entries.

The main ability of our “Goals” page is to afford users with the ability to create goals that can easily be tracked and answers. We decided to do a question-based implementation that allows users to create goal questions that they want to answer at a certain frequency. From user testing, we discovered that our users, especially those in the Habit and Spending audiences, wanted a way to keep track of some threshold or goal consistently, but they wanted a simple system since they did not have previous experience making goals. Our current implementation allows users to come up with a question that will be asked to them based on a frequency, while allowing them to see their progress and previous answers. They can also edit easily and see their success rate. Having the goals on a separate page also separates out logic in case the user does not necessarily want to keep track of any goals.

Tiffany’s needs are being met. The first scenario is met since Tiffany can highlight the best parts of her day in her vacation in Thailand by making a record with a note. The second scenario is met since Tiffany can look back on previous records of her trip. The third scenario is met since Tiffany can search records for the restaurant name to figure out where Casey’s birthday dinner in NYC was.

The habits scenarios are being met, including Harry's. The first scenario is met since Harry can create a goal of whether he is gaining one pound a week. He can also see the visualization page in the goals page to see what his success rate has been. The second habits scenario is being met since the person can keep track of hours of sleep by creating a record with a note describing the amount. The third habits scenario is met since they can records how much sleep they get a night, while still making a goal to keep track of a success rate (which can be seen in the goals page). The last scenario is met since they can use the search functionality to find the record that described the dream in question.

7.4 Planning & Pseudocode



Whiteboard diagrams of our component dependencies and JSON schema.

We first worked on creating a JSON schema for storing data within localStorage. After we figured out how each object would be structured and which fields were available, it was much easier to start implementation because we knew exactly how data would be structured. This is our final JSON schema:

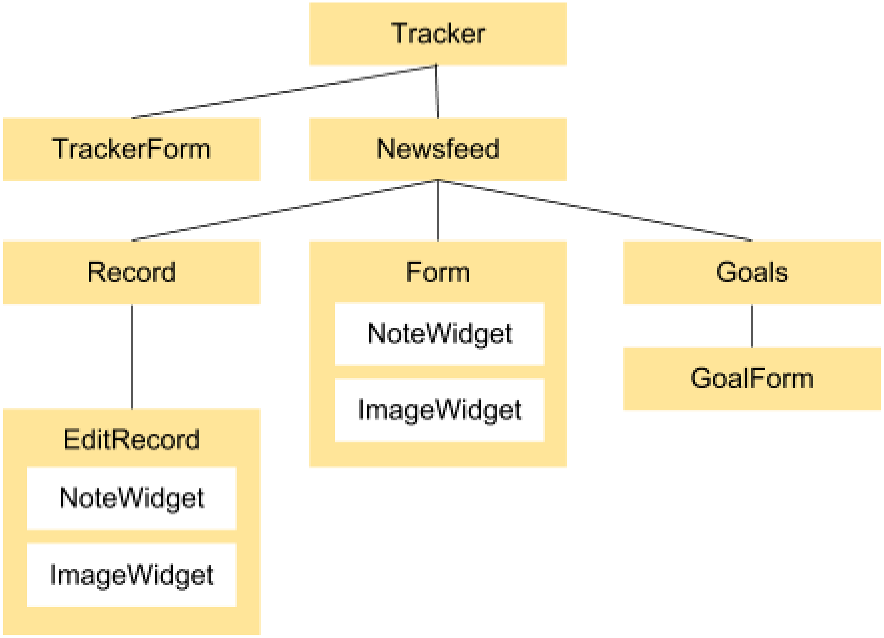
```
   {"trackers": [
   {
       "id": 0,
       "name": "Workout",
       "image":"Exercise.png",
       "records": [
           {
               "id": 0,
               "date": "2018-10-16",
               "tags": ["rough"],
               "widgetList": [{"type":"Title", "value": "Rough Day"},
{"type":"NoteWidget", "value": "3 squats 60lbs"}]
```

```
           }],
       "defaultWidgets": {"Title": "Title", "NoteWidget":
"NoteWidget", "ImageWidget": "ImageWidget"},
       "tags": ["rough", "progress", "motivation"],
       "goals": [{
           "id": 0,
           "question": "Did I workout today?",
           "frequency": "daily",
           "time": "16:25",
           "date": ["2018-10-14", "2018-10-15"],
           "answers": [null, true]
       }]
   }]}
```

Components:

- firstTracker.vue
  - Create first tracker
- Trackers.vue
  - Display all trackers
- TrackersForm.vue
  - Create a new tracker on the Trackers page
- Newsfeed.vue
  - See all your records for a specific tracker in one place
  - Filter and sort them on the top right of the nav bar
- Form.vue
  - Create a new record
- Record.vue
  - Edit and delete a new record
  - Add new widgets (more notes, images) to the specific record
- Goals.vue
  - See your goals and progress on those goals
- GoalForm.vue
  - Add a goal and frequency
- BottomNav.vue
  - Uses Vuetify component to navigate within tracker
- Individual components for each widgets
  - NoteWidget
  - ImageWidget

Component Hierarchy



Global widgets: [“title”, “note”, “photo”, “slider”, “mood”, “motivational”, “location”, “weight”]
## global widgets that are **essential** [“title”, “note”, “photo”] all the other ones are to be done later
## a record can only have a max of one “title”
## you can insert multiple photos in “photo” widget
## “mood” will be a slider to input how good you feel from 1 to 5 with faces from frowny to happy on top of the slider
## “motivational” will display a random motivational quote
## “location” will give a map with a marker when inputted an address

Newsfeed contains multiple instances of Record

- For number of records : data.json
  - Create a record component
  - Fill it with date, tags, widgetList[title]

Record:

- Date header
- Title (optional, from JSON) - text
- Tags component
- Perhaps an image if it’s attached to the record

Pseudocode

All Trackers Page

1. Fetch tracker data
2. Display tracker data on tracker news feed
3. On click of tracker, send data to Specific Tracker component, open Specific Tracker Newsfeed component

Specific Tracker Newsfeed

1. Look through JSON for data with id of the tracker clicked from All Trackers Page
2. Display Title of Specific Tracker
3. Loop through all Records
    a. Pass id to Records Newsfeed Component
    b. Display Records Newsfeed Component
4. On click of record, pass data to Specific Record component, open Read
5. If goals are answered on Newsfeed, change JSON and push to localStorage to reflect change

Create New Record

1. On click of new record button, route to new record form with params such as tracker id
2. On click add new widget button
    a. Append widget component to array of widget components
3. On click create button
    a. Get form data (date, title, notes, etc.) by looping through widget components and storing name, value pairs
    b. Loop through JSON to get the current records list for the current tracker
    c. Update the JSON by adding a new element to the records list with the current form data and an ID that is one more than the last

Specific Record (Read)

1. Look through JSON for data with id of the specific record clicked from Specific Tracker Newsfeed
2. Loop through all data and display
3. On click of edit, pass data to Edit Record Component

Specific Record (Edit)

1. Route to Edit Record page with params including record info
2. Loop through widgets from record info and populate input elements accordingly
3. On save, loop through widgets and store widget type, value pairs
4. Edit local record
5. Save record in tracker
6. Update the json to store in local storage

Goal Page (Read)

1. Choose goal question from tracker
2. Get goal responses from goal question
3. Loop through goal response and display the correct symbol and color

Goal Page (Edit)

1. Get current success status of clicked button
2. Get next success status
3. Edit goal question
4. Save to local storage
5. Refresh webpage view