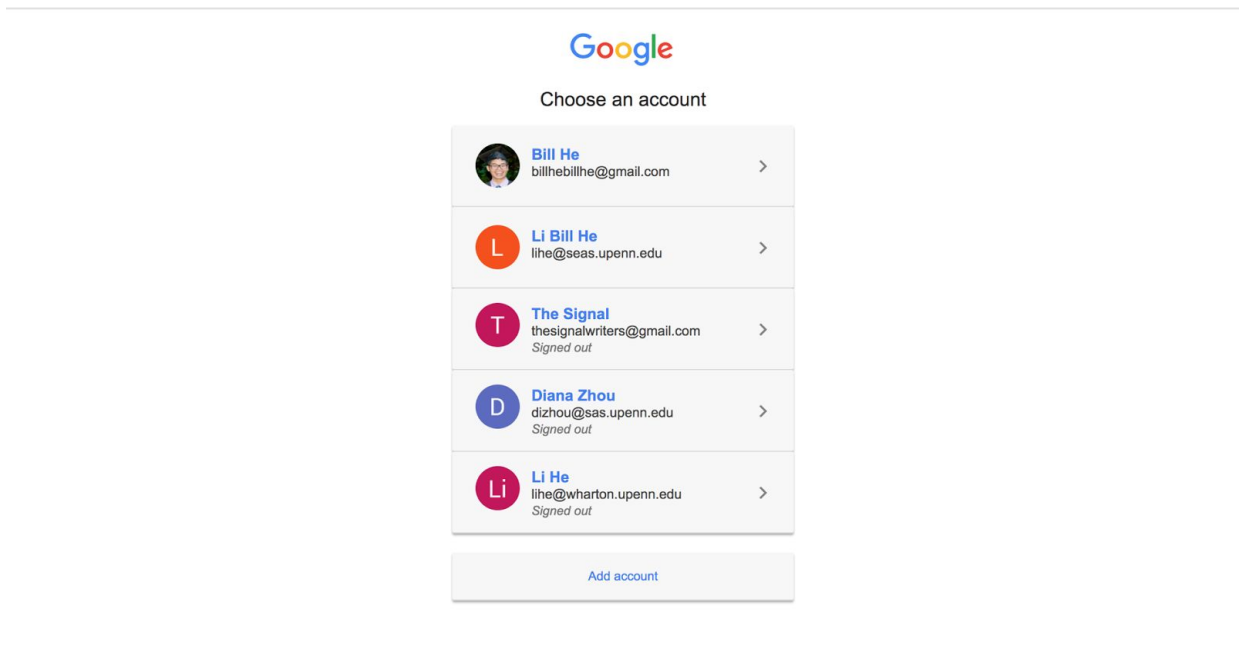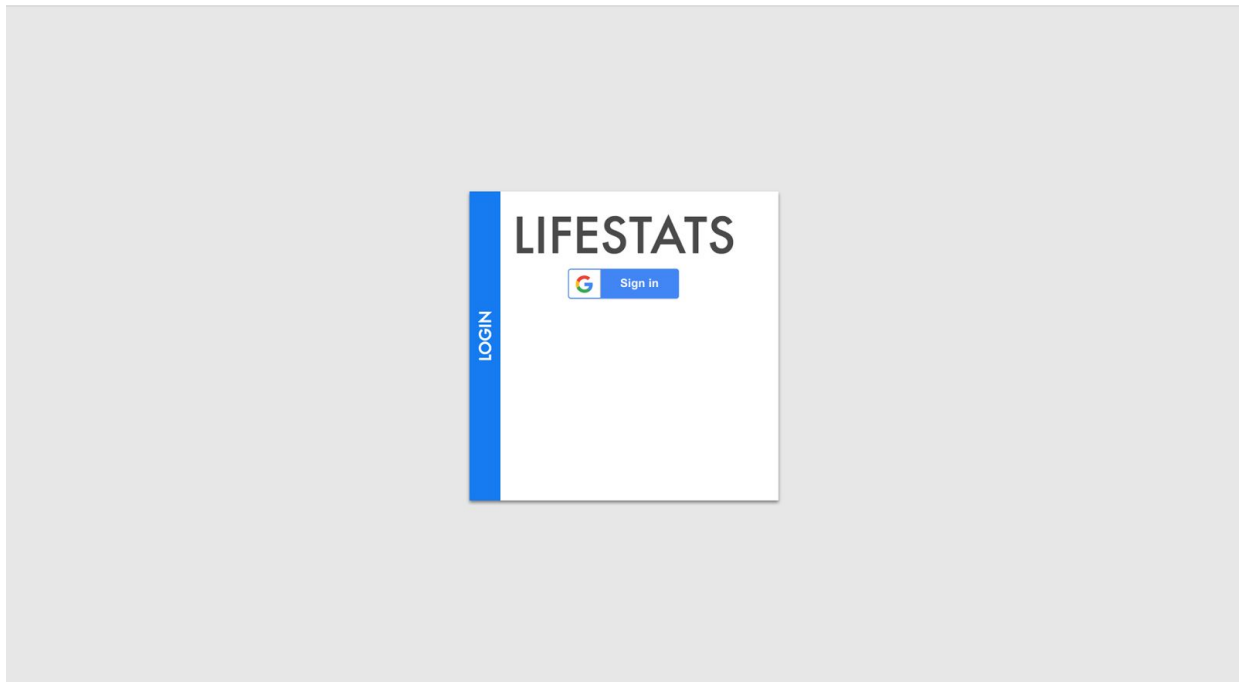# Life Stats

Final Deliverable

Developers: Lawrence Chen, Bill He, Ila Kumar, Joy Wang,
Yuting Summer Yue

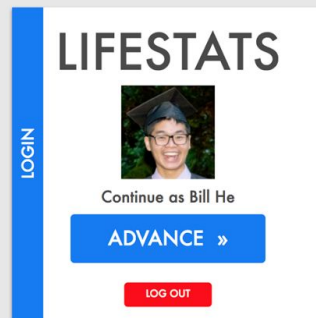CIS 350

User Manual

Login:

When you first visit the page, you will be greeted with the option to login using your Google credentials:
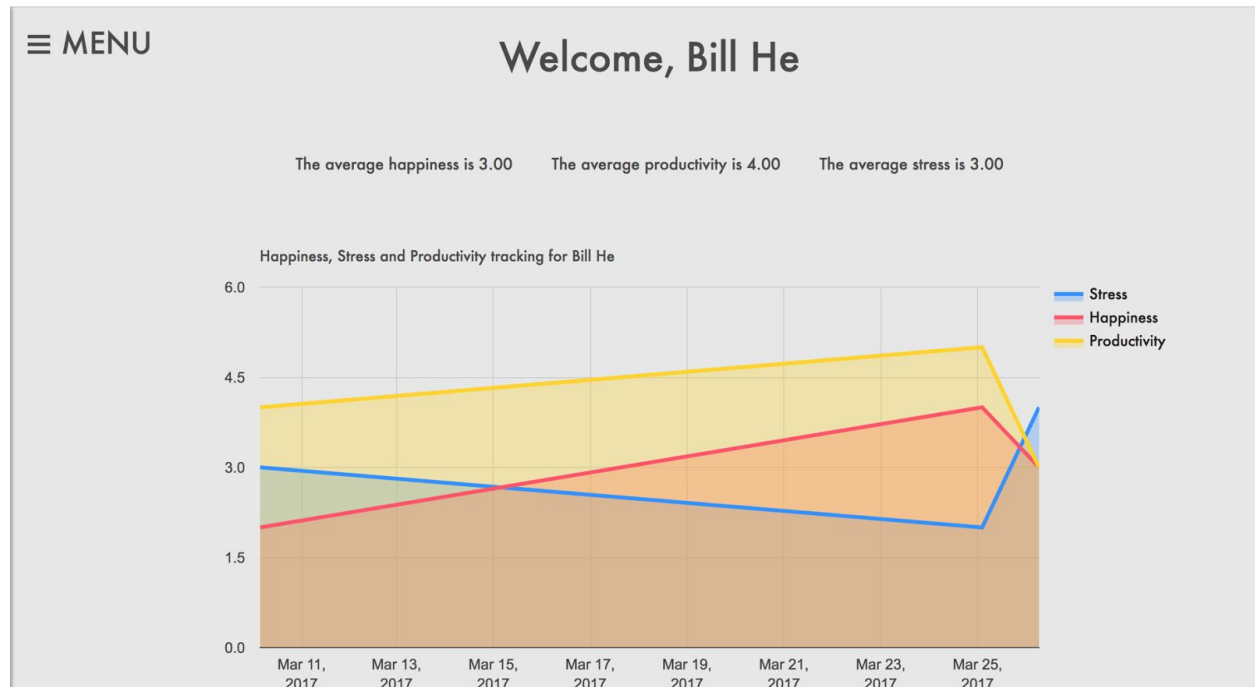
After you select the appropriate profile, you will be given the option to advance onto our application:
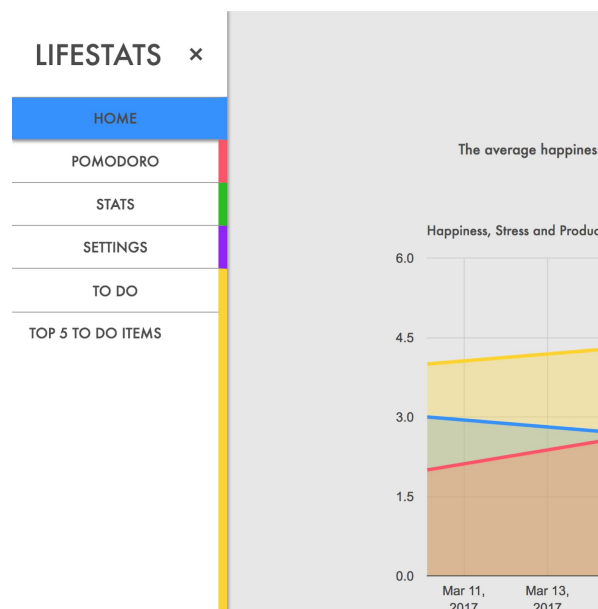

Once you press "ADVANCE", you will be directed to the homepage.

Home Page Tab:

This gives an overview of your performance of all your tasks since when you first started using the application. The app will be tracking 3 metrics that you enter in: Happiness, Stress, and Productivity.

There will also be a "MENU" button on the side. Clicking it will make the sidebar visible.

Here, you will be able to navigate to the other functions of our app: 1) your POMODORO timer, 2) your STATs, 3) your SETTINGS page, and 4) your list of TO-DO items.

After you have added a few items on your to-do list, they will also show up under "TOP 5 TO DO ITEMS":
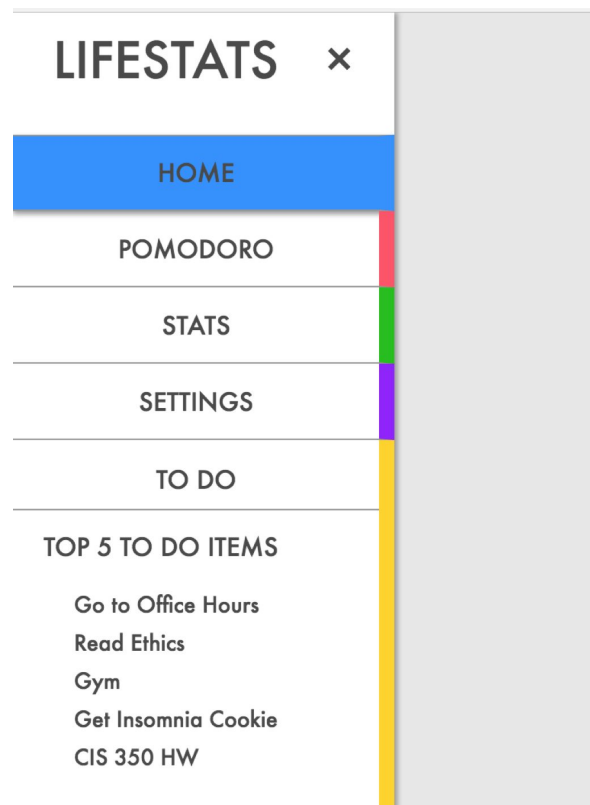
You will be able to access this navigation bar from anywhere inside our application.

Todo Tab:

When you click on the Todo tab, you will be greeted by something that looks like the following:

To Do's for Lawrence Chan 2

☐ Work on Lifestats

☐ Enjoy Fling

Work

Social

Next, I am going to…

Add

The todo tab will allow you to create, tag, and complete todo items. A counter next to the title displays the number of tasks remaining to be completed. Initially, no todo items are displayed. You can add a todo by typing the name of your todo item in the "Next, I am going to… box", then clicking the Add button.

To Do's for Lawrence Chan 0

Work on Lifestats

Add

After adding a todo item, you can click the "add tags button" to select one or more tags to add to the todo item. Currently, only three tags are enabled: social, work, and other. Tags are used in the statistics tab to help track what you do over the course of a day.

To Do's for Lawrence Chan 1

Add Tags to your Todo:

Select Tags:

Work
Social
Other

Save Changes

Add tags!

You can click on the text of the todo item to edit the text of the todo item or the tags associated with the Todo item:

To Do's for Lawrence Chan 1

Edit your todo item: ✕

Text: Work on Lifestats

Select Tags:

Work
Social
Other

Save Changes

Work   Social

Finally, you can complete a todo item by clicking the checkbox next to the todo item. This will prompt a popup that will ask you for your productivity, happiness, and stress levels associated with the todo item:

To Do's for Lawrence Chan 1

How productive were you at Work on Lifestats?

How many hours did it take: _____

Please rank your productivity from 1 to 5:
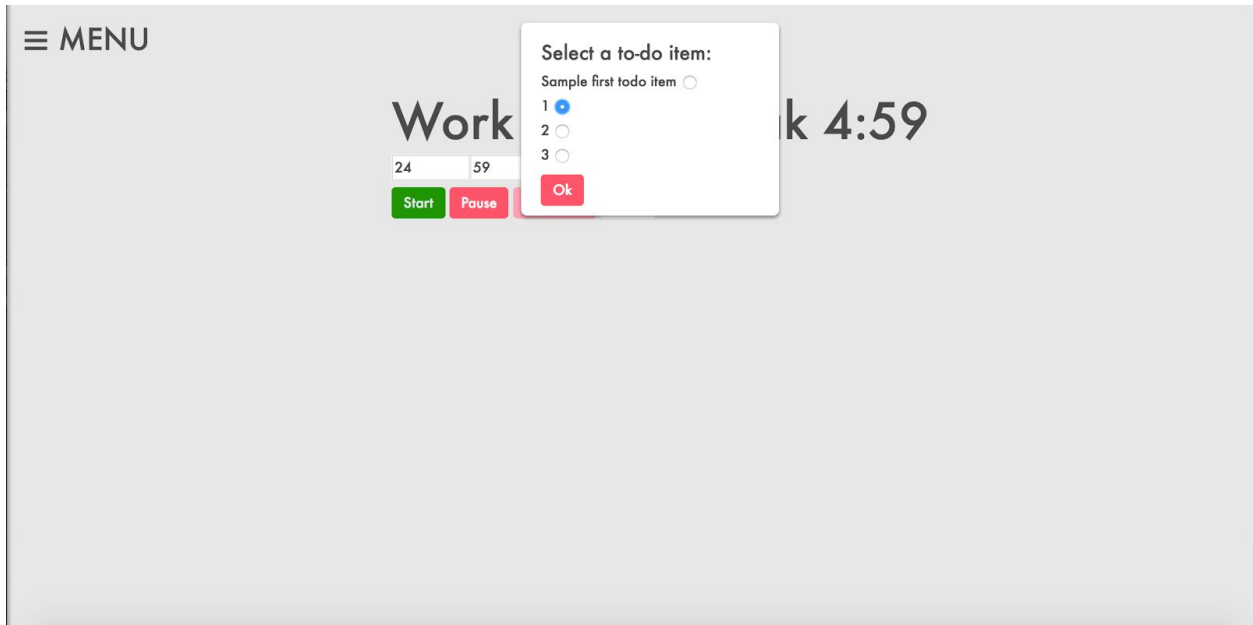
★ ★ ★ ☆ ☆

How happy did you feel:

★ ★ ★ ☆ ☆

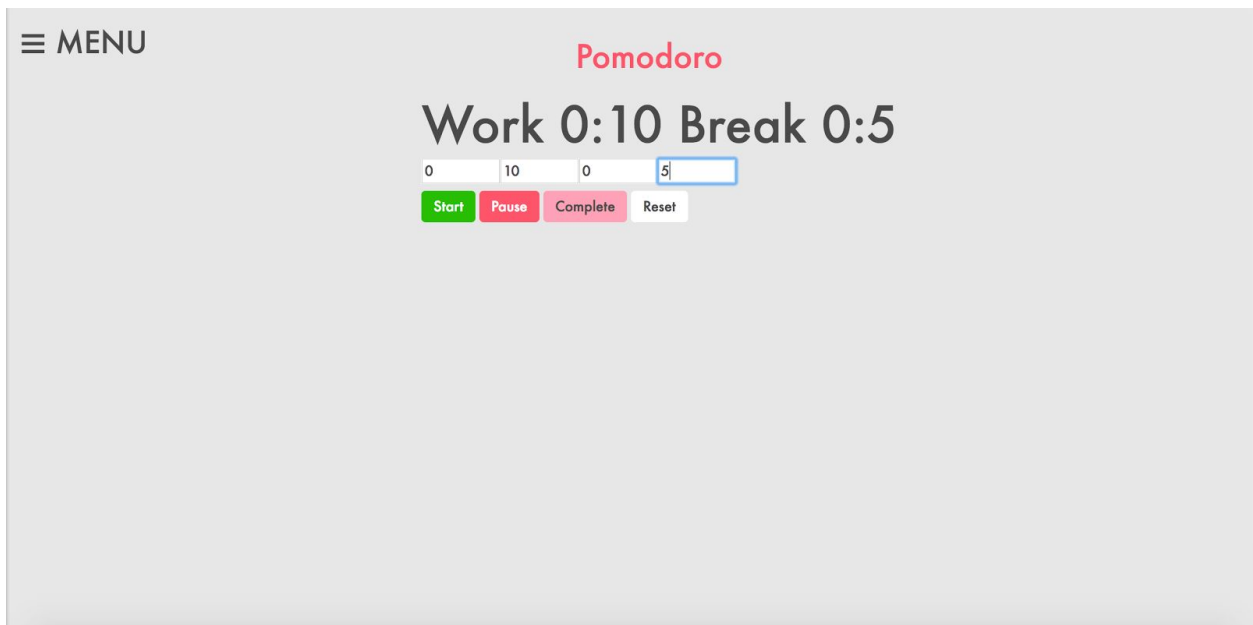How stressed did you feel:

★ ★ ★ ☆ ☆

Submit

Pomodoro:

When you go to the Pomodoro tab, you can see a list of to-do items you added in the to-do tab and you can select the task you want to start a Pomodoro session for:

Once you select the task and click on OK, the popup module will disappear and you will see a timer.
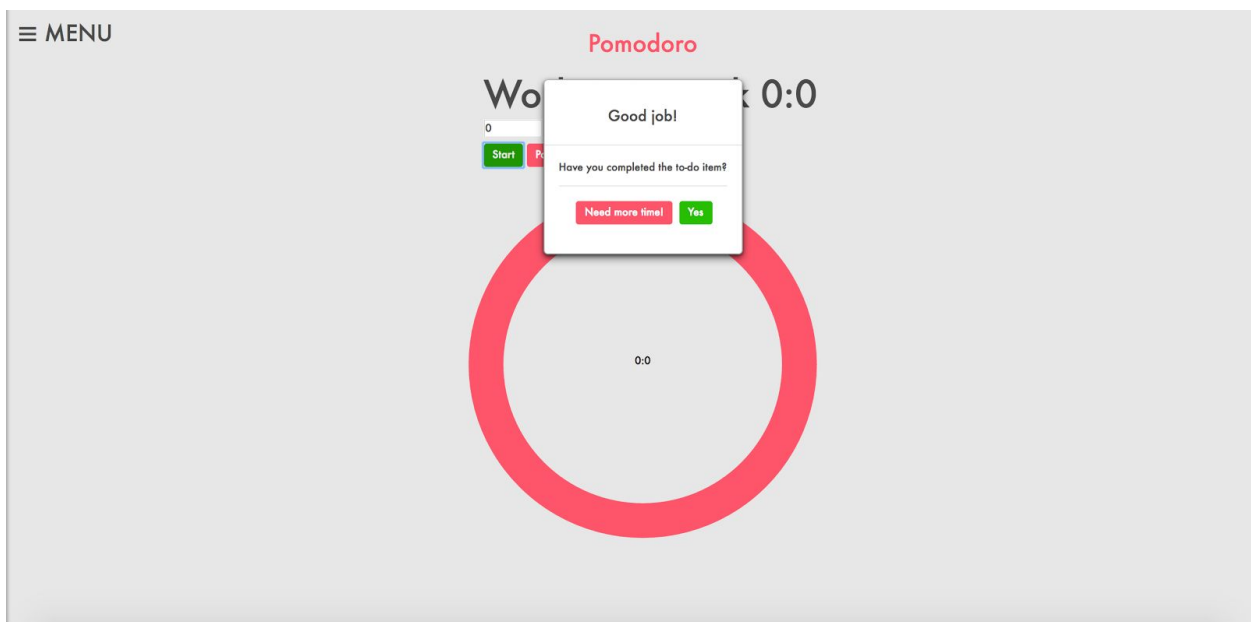
You can enter in the textbox the length of time you want to work and the length you take a break. The default value is 25 minutes work and 5 minutes break. Once you set the value, you can press the start button and the work timer will start countdown. Once the work timer counts down to 0, the break timer will start counts down.

≡ MENU

Pomodoro

Work 0:11 Break 0:5

| 0 | 11 | 0 | 5 |

Start Pause Complete Reset

0:10
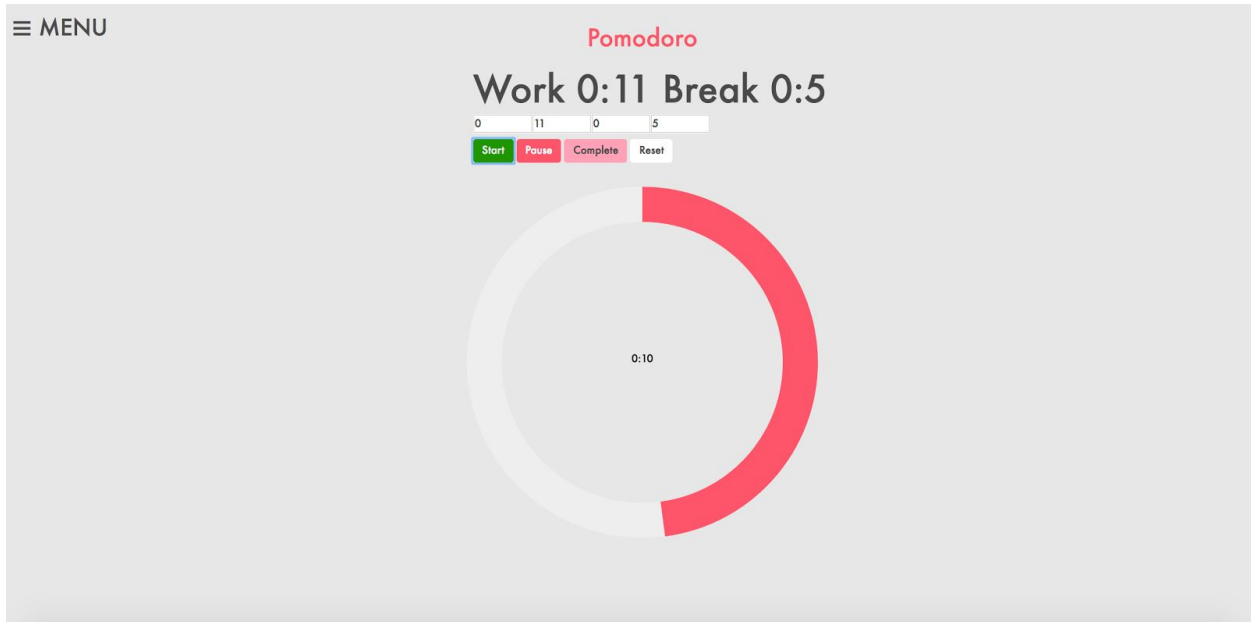
Once you start the timer, you can see a circular visualization indicating your progress.
- You can hit pause button and the timer will pause.
- You can hit complete button to complete the task. Then you will see the popup window described in next next screenshot.
- You can hit reset and the timer will reset to 25 minutes work and 5 minutes break.

≡ MENU

Pomodoro

Wo   0:0

0

Start P

Good job!

Have you completed the to-do item?

Need more time!   Yes

0:0

If both the work timer and the break timer run out to zero, you will get the popup window. You can hit the button "need more time", meaning you need more Pomodoro session to complete the task in your to-do. The session time will be incremented to the task in the database. So next time you open your Pomodoro / To-do tab, the task will remain in your to-do list.
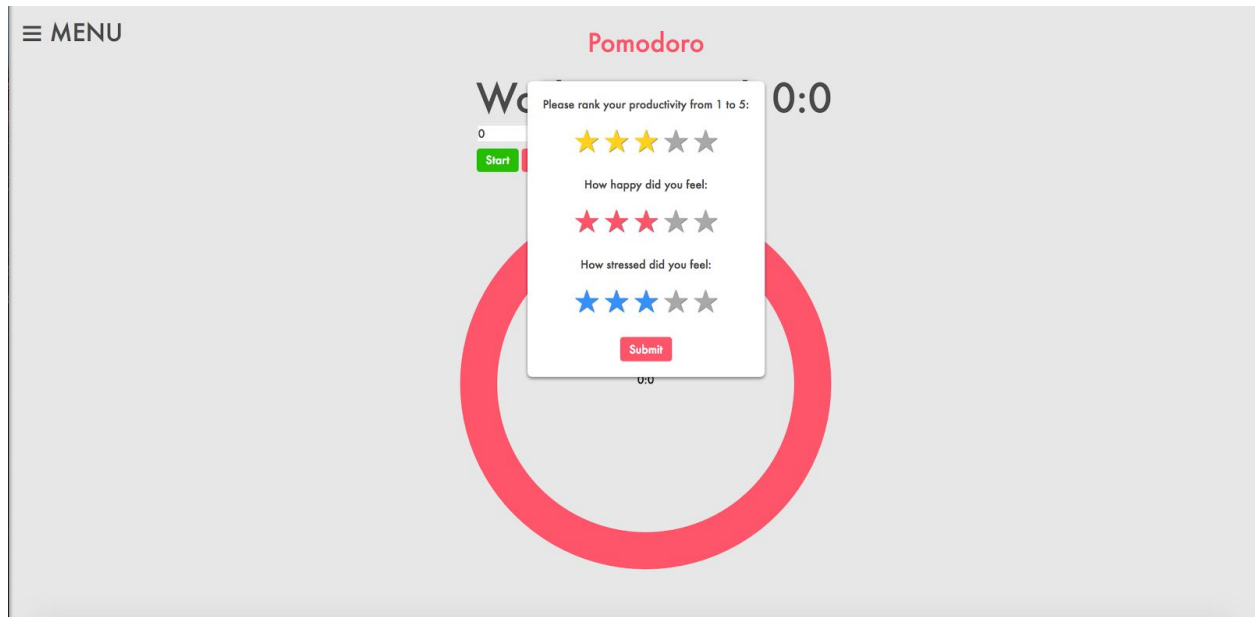
You will be prompted for this popup for two scenarios: (1) if you click on the Yes button asking you if you completed your task (2) you hit the complete button before the timer runs out. You can click on the star bars to indicate your productivity, your happiness and your stress level from 1 to 5.
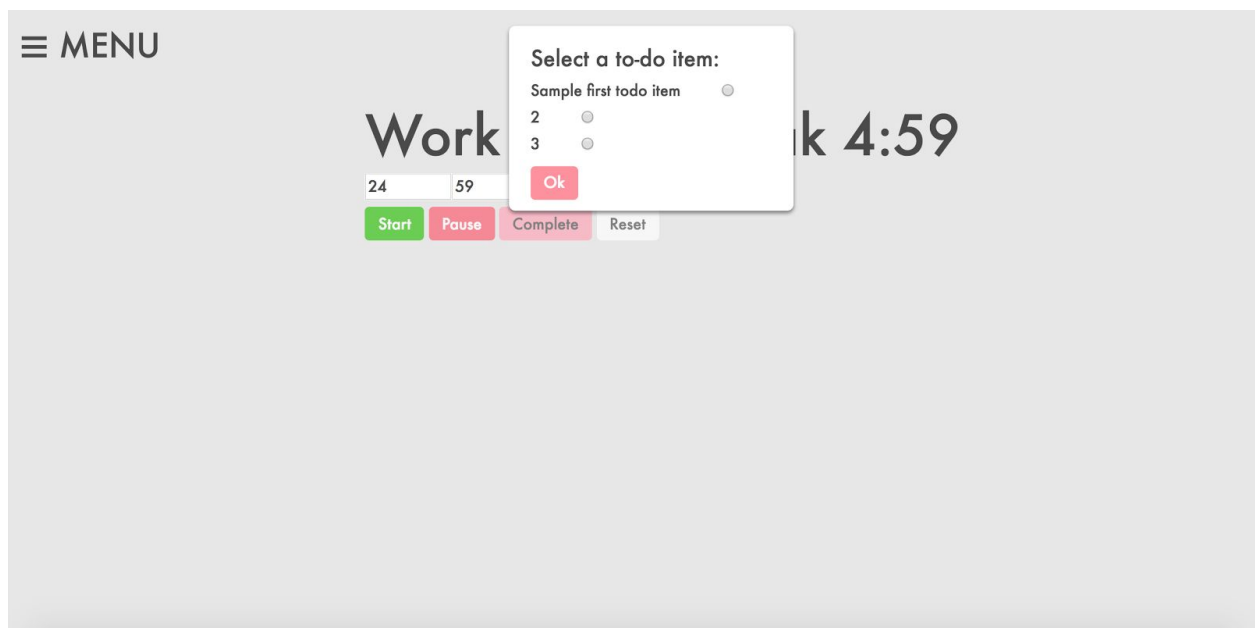
After that, you will be prompted to select the next task to work on the Pomodoro session. The task you just completed will be gone from the to-do list.
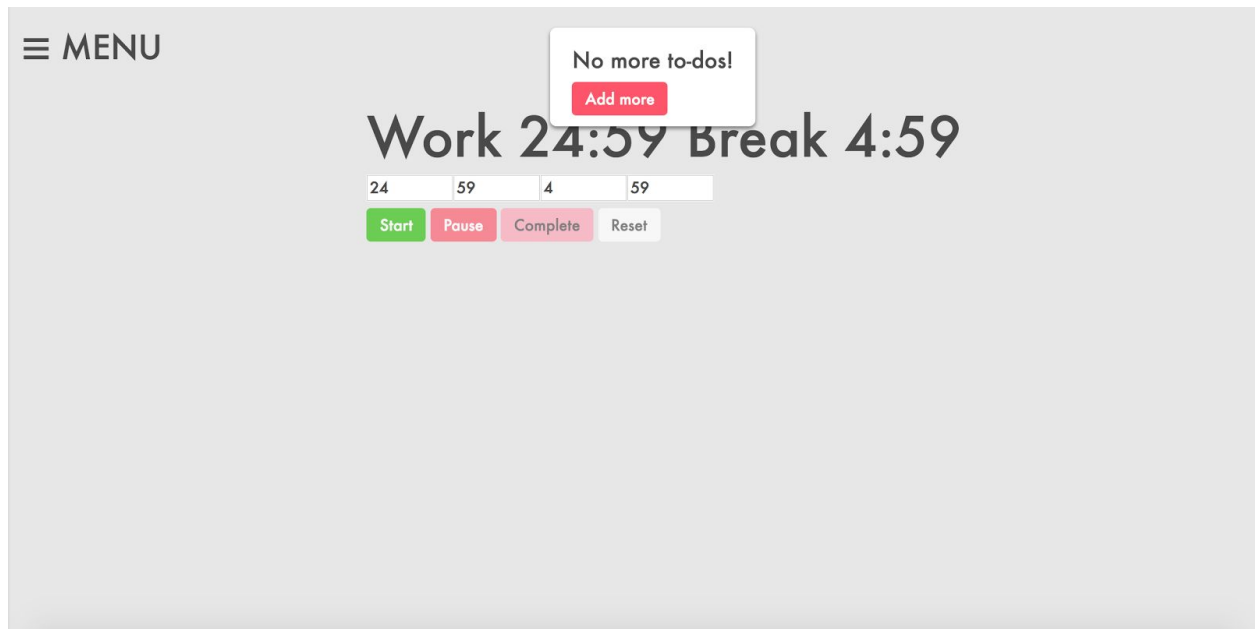
When there is no more to-do items, the popup will show a button to add more to-dos, which will take you to the to-do page to add more to-dos.

Settings Tab:

When you first enter the settings page, you can see your name on the top of page and your current settings:

Settings

Yuting Yue

☑ Happiness

☑ Stress

How often would you like us to track your stats?

Every 2 hours ⇕

Save Changes

You can select/unselect the features you want to track by clicking the checkboxes, including happiness and stress. The items checked would become part of the stats visualization.

You can select the frequencies on how often the notification popup shows up. The three options are every hour, every 2 hours and every 4 hours.

Settings

Yuting Yue

☑ Happiness

☑ Stress

How often would you like us to track your stats?

Every 2 hours ▲▼

Save Changes

You've updated your settings at Sun Apr 23 2017 13:07:04 GMT-0400 (EDT)

The settings are not saved until you click save changes, the settings changes will be saved then, and a message would show up on the bottom of the "save changes" button indicating the time when you updated settings last time.

Please rank your happiness from 1 to 5:

★ ★ ★ ☆ ☆

Please rank your stress from 1 to 5:

★ ★ ★ ☆ ☆

Submit

Setting

Yuting Yue

☑ Happine

☑ Stress

How often

Every 2 hours ▲▼

Save Changes

You've updated your settings at Sun Apr 23 2017 13:07:04 GMT-0400 (EDT)

A notification popup would show up in your selected frequency. It will ask you to rank the features you selected to track. In the example, the popup will track user's happiness and stress level every 2 hours.

You can rank your happiness and stress levels by clicking on the stars. 5 stars mean very high happiness/stress level and 1 star means very low. After you click the submit button, the data is saved and will be visualized in the stats tab. The popup disappears.


If you unselect "Happiness" in settings, the popup will only prompt you to rank stress level. Same if you unselect stress. If you neither happiness nor stress is selected in settings, the notification popup will simply not show up.

Statistics Tab:

When you go to the Statistics tab, you see a graph of your happiness, stress and productivity over time (the Overall Statistics page).
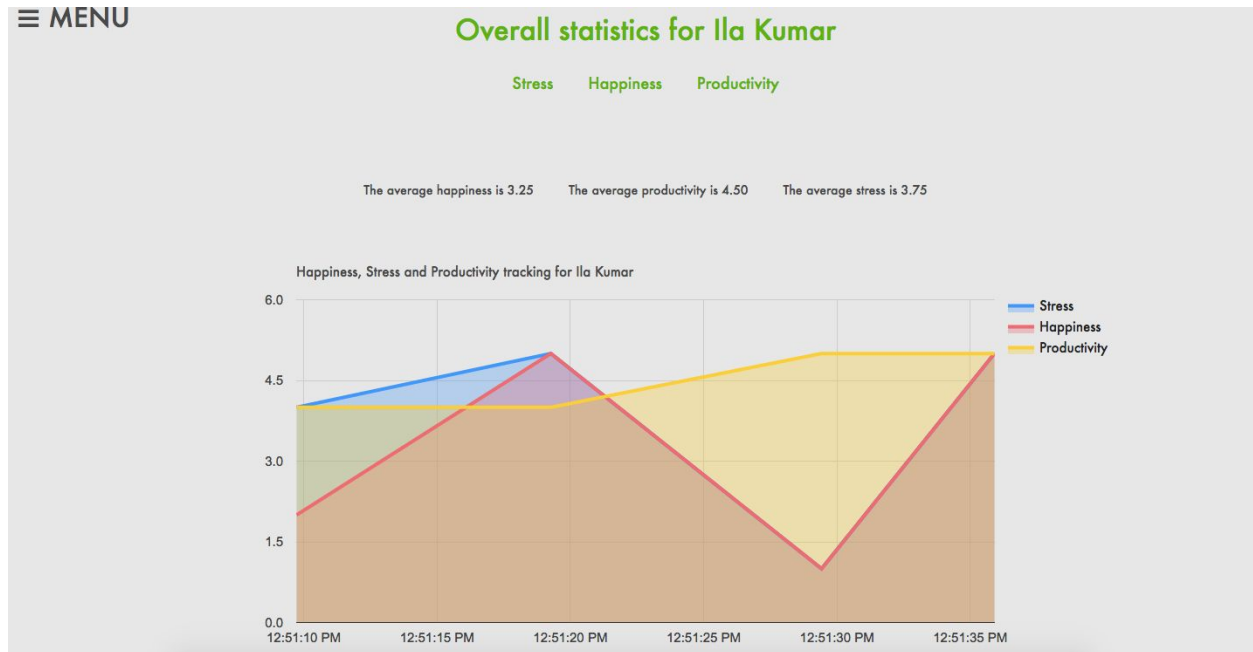
If you hover over the data points on the graph, you can see the date, time and stress/happiness/productivity rating of the point.
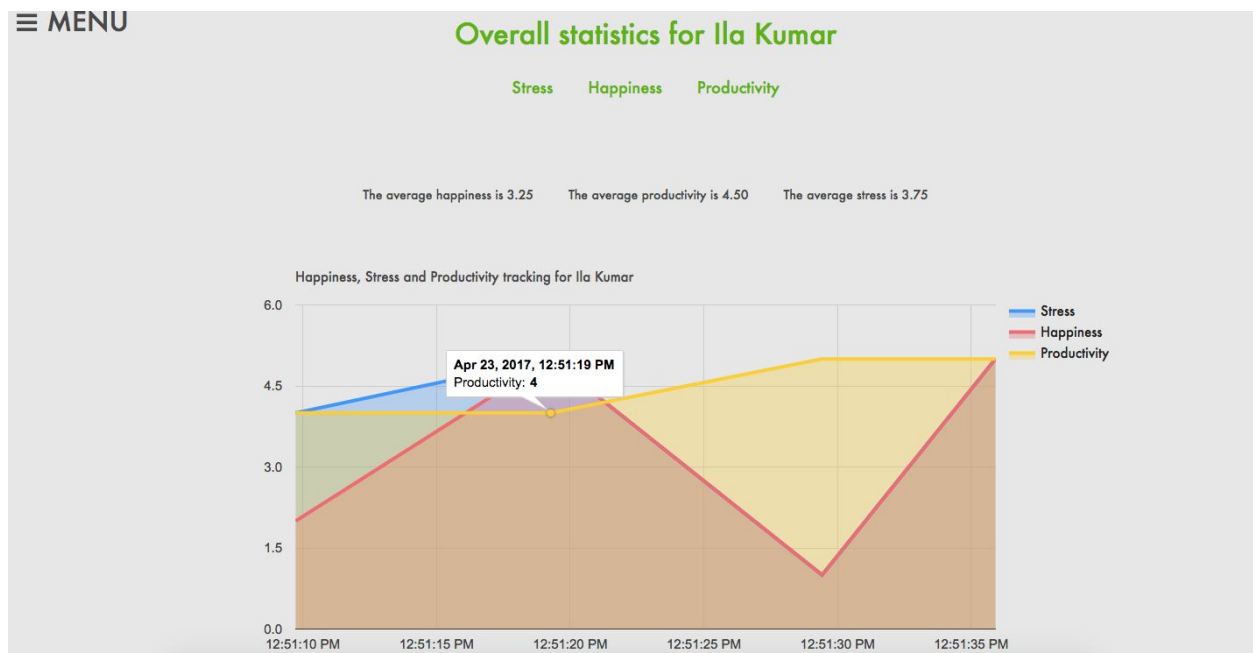
If you click on the Stress link of the Overall Statistics page, you see an option to select the time frame of the graph you wish to display.

Once you select a time frame and click the Display button, you see a graph of just Stress ratings.
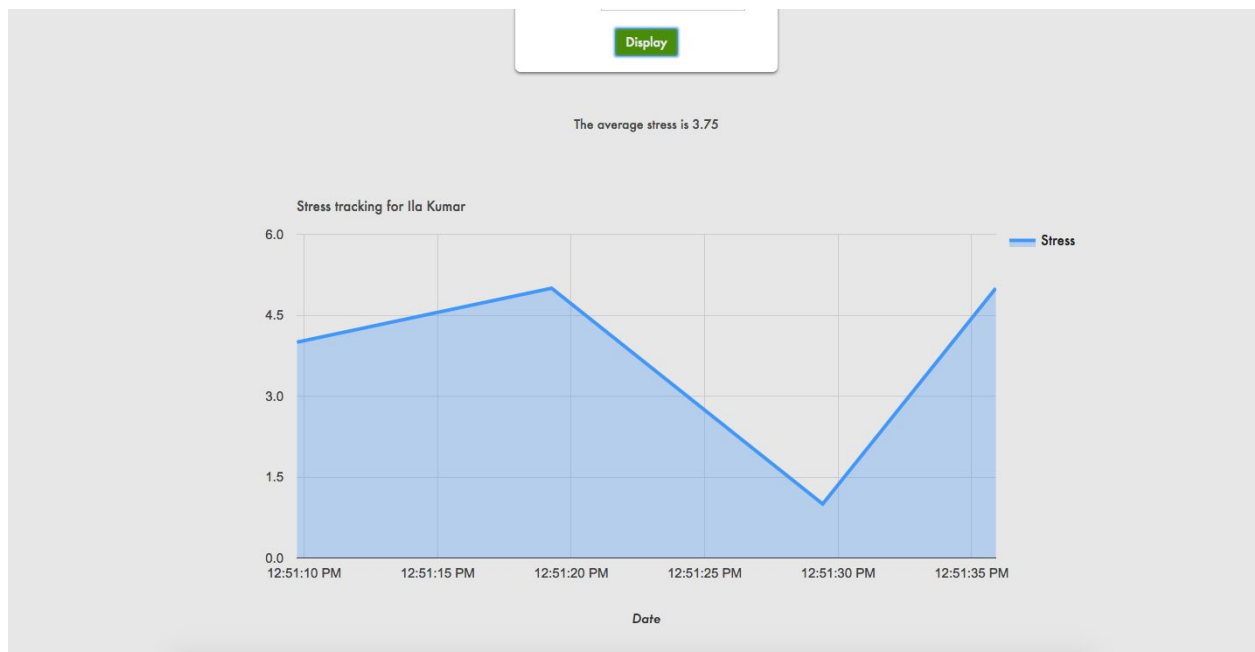
If you click on the Happiness link of the Overall Statistics page, you see an option to select the time frame of the graph you wish to display.

Once you select a time frame and click the Display button, you see a graph of just Happiness ratings.
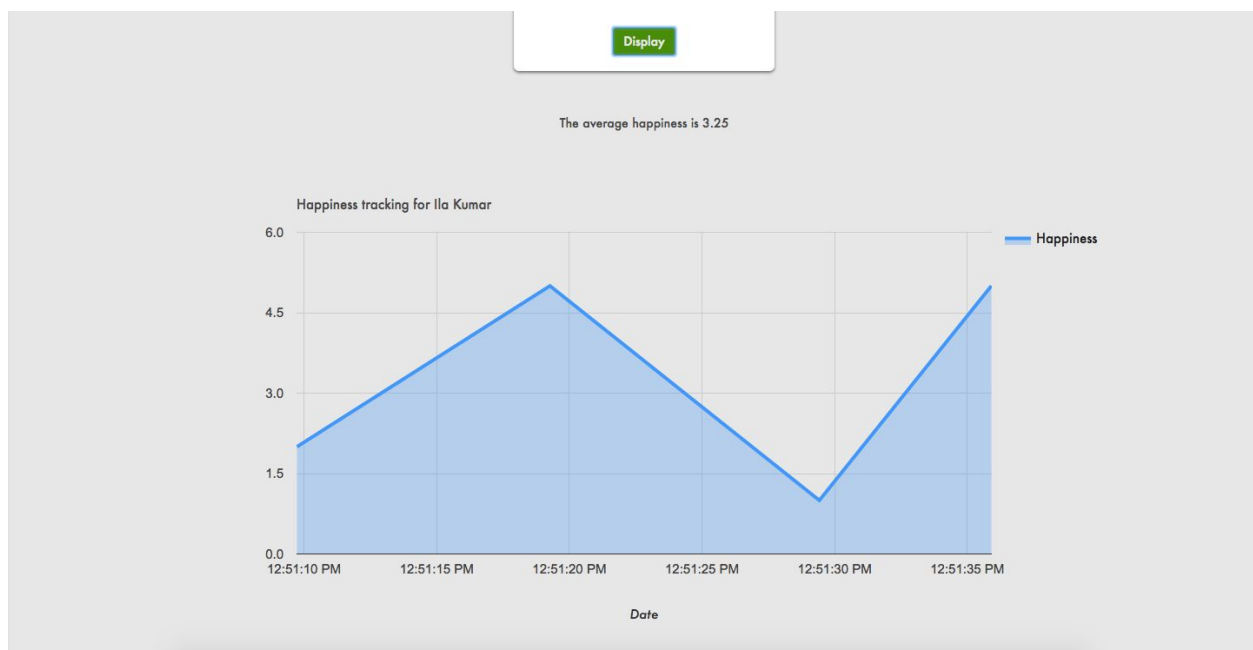
If you click on the Productivity link of the Overall Statistics page, you will see a happy dog if you have completed more than 80% of your tasks today.
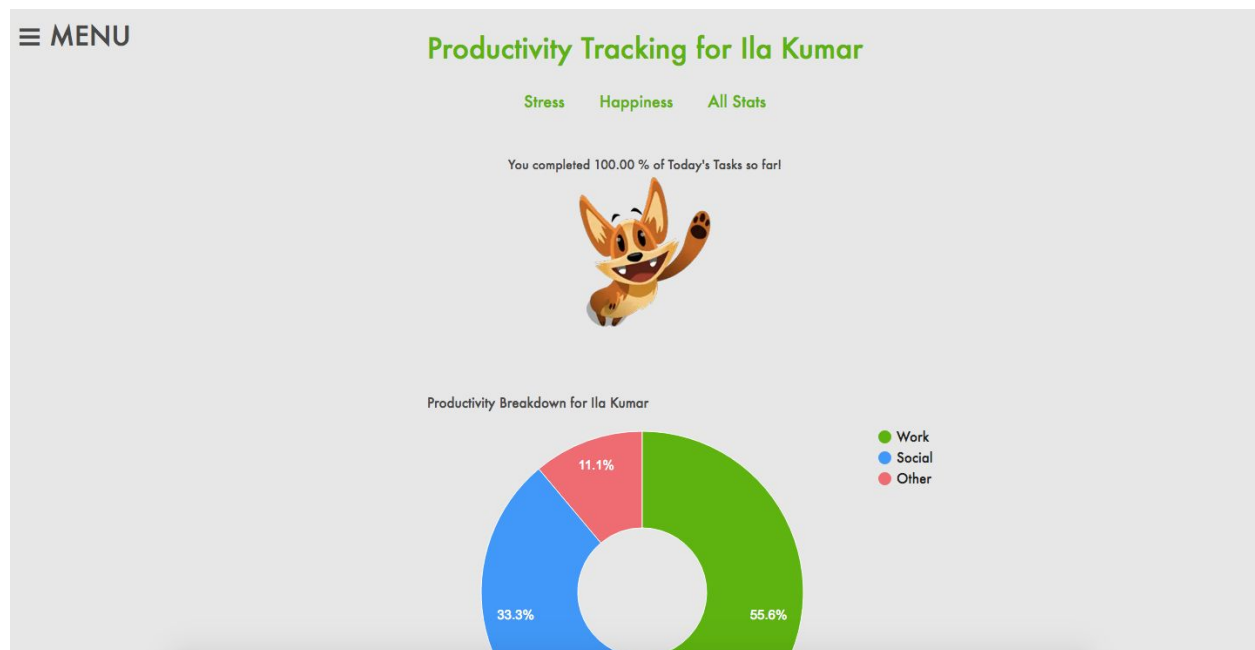
You will see a sad dog if you have completed less than 20% of your tasks today.
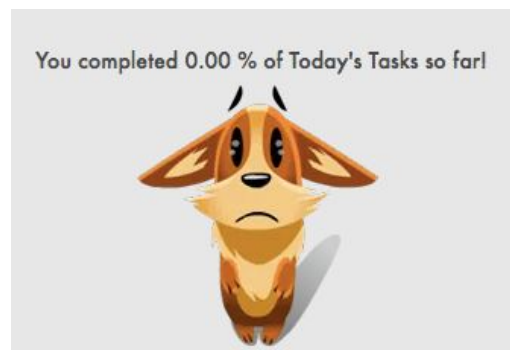
You will see a neutral dog if you have completed between 20% and 80% of your tasks today.
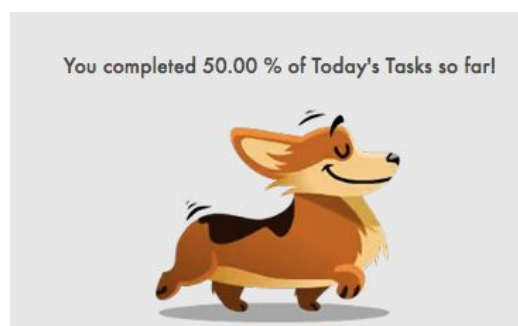
Below the dog, you will see a pie chart of the percentage time you spent on tasks of each type of tag.
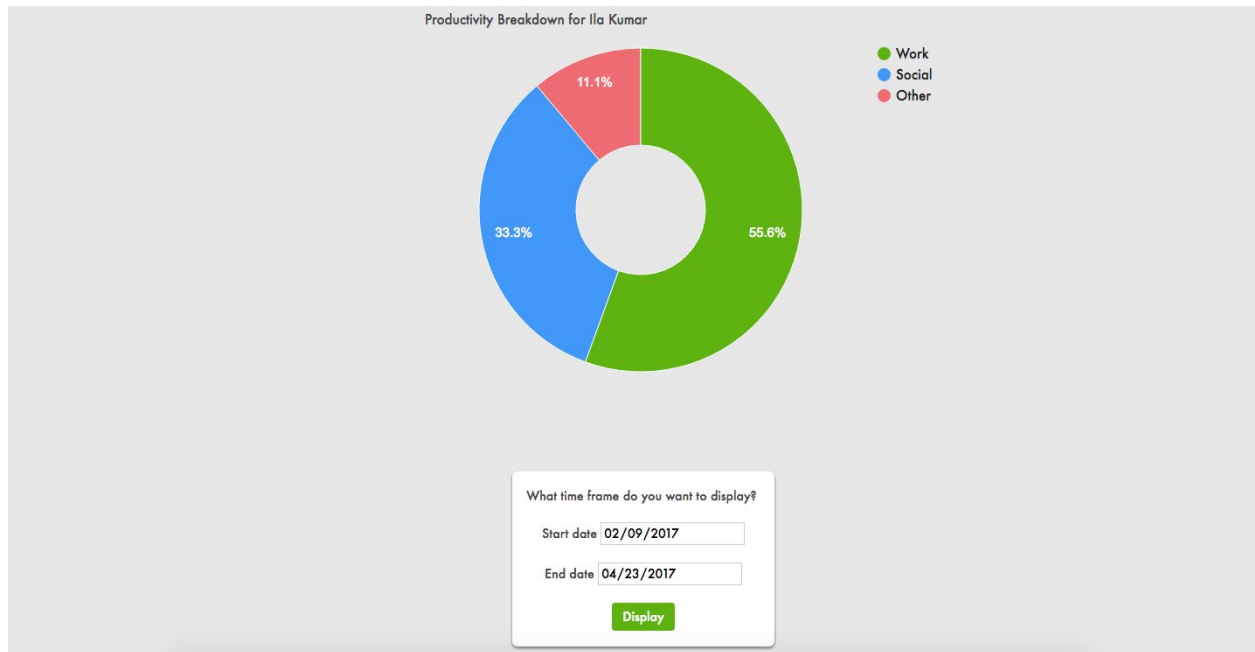
If you hover over the sections of the graph, you can see the tag name, number of tasks there are of that tag, and the percentage of overall time you spent on that type of task. Below the pie chart, you will see an option to select the time frame of the graph you wish to display.
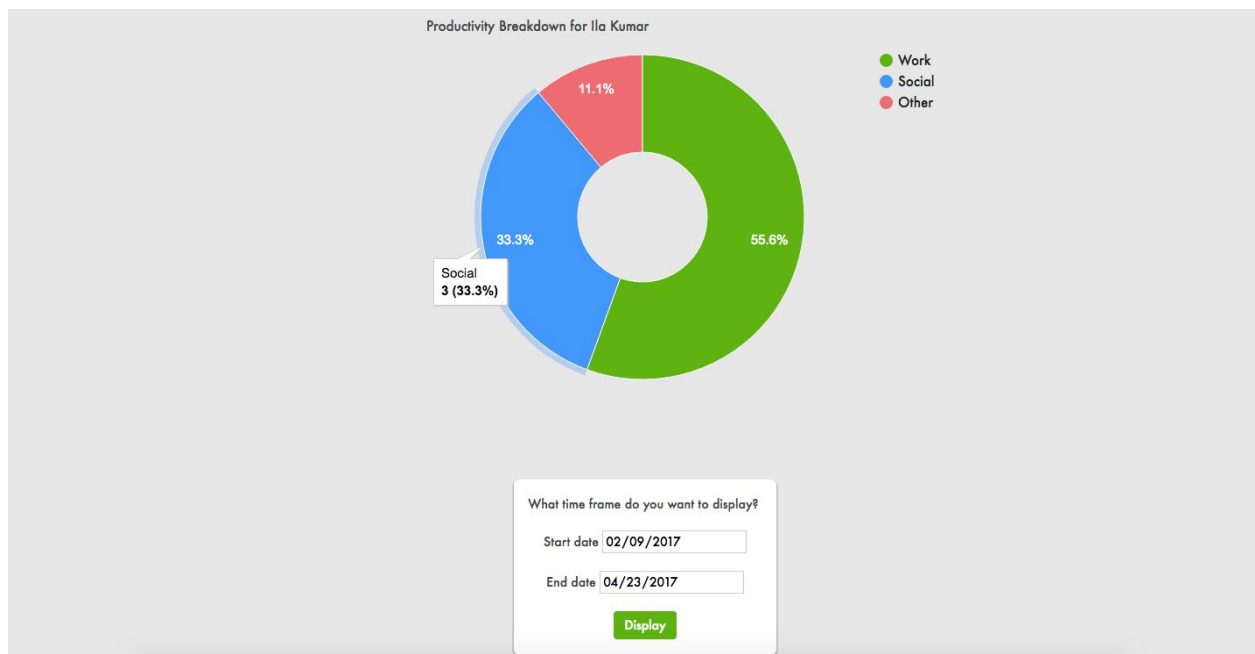
Once you select a time frame and click the Display button, you see a graph of just Productivity ratings.
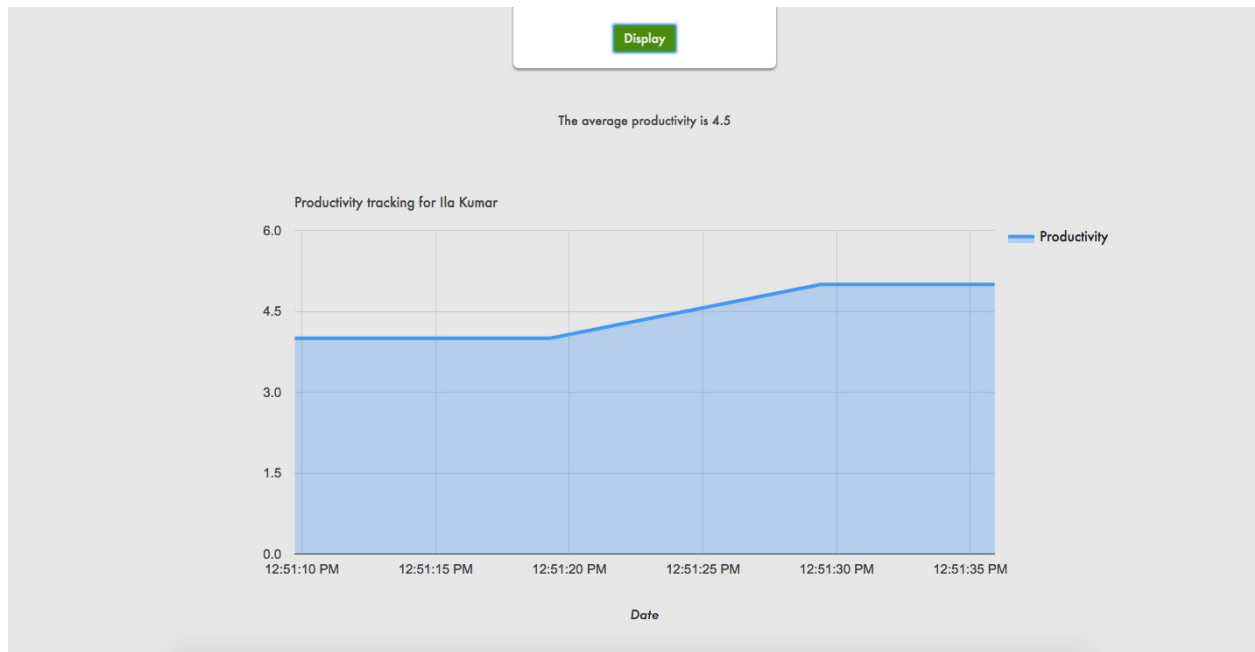
Known Bugs:

Todo Tab

Currently, if you refresh the page after completing a todo item, there will be no hours and no productivity/happiness/stress associated with the todo item.

Saving the changes to a todo item triggers a refresh of the whole page instead of editing the todo item directly.

The CSS doesn't extend far enough when there are more than 11 todo items.

Technical Manual

Relevant Technologies

1. MongoDB db version v3.2.9, Mongoose@4.8.2
2. Node@7.8.0, Express@4.14.1
3. AngularJS v1.6.5-build.5353+sha.136a42a
4. Morgan@1.8.1, bodyParser@1.16.1, methodOverride@2.3.7

Summary of MVC

Category	Model	View	Controller
Login	models/user.js	public/index.html public/css/login.css	public/js/userController.js
	Responsible for collecting user information after he or she logs in. Login screen is the view that collects this information.		
Home	models/user.js	public/homepage.html public/css/homepage.css	public/js/userController.js
	Responsible for displaying user information. The homepage is the view that gets information about the user.		
Todo	models/todo.js	public/todo.html public/css/todo.css	public/js/todoController.js
	Responsible for the Todo tab where user can add, edit and complete their todo		

	tasks.		
Pomodoro	models/todo.js	public/pomodoro.html public/css/pomodoro.css	public/js/pomodoroController.js
	Responsible for the Pomodoro tab where user can use the timer to track task completion.		
Stats	models/user.js	public/stats-html public/css/stats.css	public/js/statsController.js
	The controller provides access to google visualization libraries. Upon calling apis, task data is displayed to the user.		
Settings	models/user.js	public/settings.html public/css/settings.css	public/js/settingsController.js
	Responsible for the settings tab where users can modify their user settings on what features to use, including happiness and stress tracking, as well as how often they would like the tracking notification box to pop up.		
Navigation		public/navigation.html public/css/navigation.css	
Notif	models/user.js	public/notif.html public/css/notif.css	
App			public/app.js (contains TimerService, notif directive and starRating directive) routes.js server.js
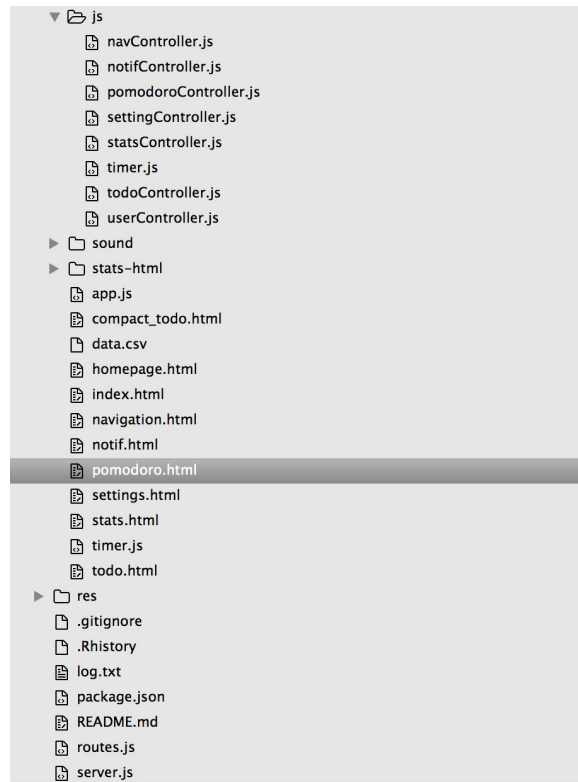
Back End Files

1. Server: server.js is the main file for the backend. One server.js is running, the databases are initialized. The app starts. The http requests associated with the app including various APIs can be viewed in this file.
2. routes.js specifies how the API calls in server.js are implemented. The APIs are exported as a module.

Mongo Database Schema

1. todoSchema

Field Name	Type	Explanation
user	String	email of the user associated with the Todo item
text	String	Text of the todo item
productivity	Number	A number from 1-5 indicating how productive the user is at this todo, filled after the todo is completed
hours	Number	The number of hours taken to complete this todo, filled after the todo is completed
tags	List	List of tags associated with the Todo

createdTime	Date	When the todo is created
finishedTime	Date	When the todo is finished
done	Boolean	Boolean value indicating if todo is finished
happiness	Number	The happiness number from 1 to 5 associated with the todo
stress	Number	The stress number from 1 to 5 associated with the todo

2. userSchema

Field Name	Type	Explanation
email	String	Email associated with user login
username	String	Name of the user
joinedTime	Date	When the user first logged into the app
trackHappiness	Boolean	Boolean indicating if happiness level is tracked in stats
trackStress	Boolean	Boolean indicating if stress level is tracked in stats
trackPomodoro	Boolean	Boolean indicating if pomodorois tracked in stats
trackTodo	Boolean	Boolean indicating if todo is tracked in stats
popupFrequency	String	Frequency on how often notification popup shows up, in the format "1h", "2h" and "4h"
happinessData	[{"time": Date, "happinessLevel": Number}]	All happiness data recorded for the user, including time and happiness level for each point.
stressData	[{"time": Date, "stressLevel": Number}]	All stress data recorded for the user, including time and stress level for each point.
age	Number	[NOT USED] Age of the user.
lang	String	[NOT USED] User's language.
id_token	String	ID token that Google Authentication provides. We use this to authenticate the users.
id	String	The Google ID of the user.
img	String	URL link to an image of the user (provided by the Google API).

Google Authentication

Our authentication is done through OAuth 2.0 client credentials that we got from the Google API. Here, you can find the instructions that we followed.

https://developers.google.com/identity/protocols/OAuth2

The following block of code exposes the method that allows us to login using the GoogleAPI. This is found in **public/js/userController.js**. This returns a list of information associated with the user that we store in our database, including the access_token.

```
    $scope.renderSignInButton = function() {
        console.log("renderSignInButton called");
        gapi.signin.render('signInButton',
            {
                'callback': $scope.signInCallback, // Function
handling the callback.
                'clientid':
'659758037057-afohj28tbl3ofjvkem042t7nt43rhcd5', // CLIENT_ID
from developer console which has been explained earlier.
                'requestvisibleactions':
'http://schemas.google.com/AddActivity', // Visible actions,
scope and cookie policy wont be described now,

// as their explanation is available in Google+ API
Documentation.
                'scope':
'https://www.googleapis.com/auth/plus.login
https://www.googleapis.com/auth/userinfo.email',
                'cookiepolicy': 'single_host_origin'
            }
        );
    }
```

Contact Information

Developers
Bill He <lihe@seas.upenn.edu>
Lawrence Chan <chanlaw@sas.upenn.edu>
Ila Kumar <ilakumar@sas.upenn.edu>
Yuting Yue <yyue@seas.upenn.edu>
Qiaoyi Wang <qiaoyi@sas.upenn.edu>

Project Manager
Julio Reynaga Galeas <jreyn@seas.upenn.edu>