



DEVELOPMENT OF TRACKR

by:

Longjie Chao

Sophia Patrice Manalo Mendoza

A Thesis Submitted in Partial Fulfillment of the Requirements for the degree of Web Application Development

Approved by:

Mr. Josep Queralt

Mr. Jordi Bosom

Ciclo de Grado Superior

Institut Nicolau Còpernic

May, 2021

Table of Contents

1 Introduction	1
2 Project Planning.....	2
3 Marketing	3
3.1 Executive Summary.....	3
3.2 Products and services.....	3
3.3 Mission.....	3
3.4 Vision.....	3
3.5 Competition.....	3
3.6 Competitive advantage	4
3.7 SWOT Analysis.....	4
3.7.1 Strengths.....	4
3.7.2 Weaknesses.....	4
3.7.3 Opportunities	4
3.7.4 Threats.....	4
3.8 Buyer persona.....	4
3.9 Marketing objectives	5
3.10 Marketing budget.....	5
3.10.1 Development Schedule.....	5
3.10.2 Marketing Mix	5
3.10.3 Social media.....	6
4 Analysis Phase.....	7
4.1 Non functional requirements.....	7
4.1.1 Technologies.....	7
4.2 Functional requirements	8
4.2.1 Use Case Actor Census	8
4.2.2 Use Case Census	8
4.2.3 Analysis Class Diagram	10
4.2.4 Use Case Diagram.....	10
4.2.5 Use Case Specification.....	11
5 Design	20
5.1 Database design.....	20
5.2 Sketch.....	21
5.3 Mockup.....	29
5.3.1 Login	29
5.3.2 Registry.....	30
5.3.3 Main page	31
5.3.4 Menu.....	32
5.3.5 Mood Tracker	33
5.3.6 Financial Tracker	34
5.3.7 Old Transactions	35
5.3.8 To do folder	36
5.3.9 Task and Challenges.....	37
5.3.10 Task and Challenges' CRUD.....	38
5.3.11 Contacts	39
5.3.12 Contacts' CRUD	40
5.3.13 Calendar.....	41
5.3.14 Scheduler	42
5.3.15 Scheduler's CRUD.....	43

5.4 Style Guide.....	44
6 User's Manual.....	45
6.1 Login.....	45
6.2 Register	45
6.3 Menu	46
6.4 Mood Tracker.....	47
6.5 Financial Tracker.....	47
6.6 Old Transactions.....	48
6.7 To do folder.....	49
6.7.1 To do folder's CRUD	49
6.8 Tasks and Challenges	49
6.8.1 Task and Challenges CRUD.....	50
6.9 Scheduler.....	51
6.9.1 Scheduler's CRUD.....	51
6.9.2 Schedule task and its CRUD.....	52
6.10 Contacts and its CRUD.....	53
6.11 Calendar	55
6.12 Main.....	56
7 Installation Manual	57
8 Bibliography	58

Table of Figures

Figure 1: Gantt Diagram (1)	2
Figure 2: Gantt Diagram (2)	2
Figure 3: Analysis Class Diagram	10
Figure 4: Use Case Diagram.....	10
Figure 5: Database design	20
Figure 6: Sketch (Login and Registry).....	21
Figure 7: Sketch (Main and Menu)	22
Figure 8: Sketch (Mood and Financial Transactions)	23
Figure 9: Sketch (Old Transactions and To do folder)	24
Figure 10: Sketch (Task and Challenge, and its CRUD)	25
Figure 11: Sketch (Contact and its CRUD).....	26
Figure 12: Sketch (Scheduler and its CRUD)	27
Figure 13: Sketch (Calendar).....	28
Figure 14: Mockup (Login)	29
Figure 15: Mockup (Registry)	30
Figure 16: Mockup (Main)	31
Figure 17: Mockup (Mood Tracker)	33
Figure 18: Mockup (Financial Tracker)	34
Figure 19: Mockup (Old Transactions).....	35
Figure 20: Mockup (To do folder)	36
Figure 21: Mockup (Task and Challenge).....	37
Figure 22: Mockup (Tasks and Challenges' CRUD)	38
Figure 23: Mockup (Contacts).....	39
Figure 24: Mockup (Contacts' CRUD)	40
Figure 25: Mockup (Calendar)	41
Figure 26: Mockup (Scheduler).....	42
Figure 27: Mockup (Scheduler's CRUD)	43
Figure 28: Style Guide	44
Figure 29: User's Manual (Login).....	45
Figure 30: User's Manual (Registry).....	45
Figure 31: User's Manual (Menu Button)	46
Figure 32: User's Manual (Menu).....	46
Figure 33: User's Manual (Mood Tracker).....	47
Figure 34: User's Manual (Financial Tracker's CRUD).....	47
Figure 35: User's Manual (Financial Tracker).....	48
Figure 36: User's Manual (Old Transaction's Button)	48
Figure 37: User's Manual (Old Transaction)	48
Figure 38: User's Manual (Old Transaction's Data).....	48
Figure 39: User's Manual (To do folder).....	49
Figure 40: User's Manual (To do folder's CRUD)	49
Figure 41: User's Manual (Tasks and Challenges)	50
Figure 42: User's Manual (Task's CRUD).....	50
Figure 43: User's Manual (Challenges' CRUD)	51
Figure 44: User's Manual (Scheduler).....	51
Figure 45: User's Manual (Scheduler's CRUD (1))	52
Figure 46: User's Manual (Scheduler's CRUD (2))	52
Figure 47: User's Manual (Scheduler's Task).....	53
Figure 48: User's Manual (Scheduler's Task's CRUD)	53

Figure 49: User's Manual (Contacts) 54

Figure 50: User's Manual (Contacts' CRUD)..... 54

Figure 51: User's Manual (Calendar)..... 55

Figure 52: User's Manual (Main)..... 56

Index of Tables

Table 1: Use Case Actor Census.....	8
Table 2: Use Case (Sign Up)	11
Table 3: Use Case (Sign In).....	11
Table 4: Use Case (Contact List).....	12
Table 5: Use Case (To do Folders)	13
Table 6: Use Case (Tasks and Challenges)	15
Table 7: Use Case (Mood Tracker).....	16
Table 8: Use Case (Financial Tracker).....	17
Table 9: Use Case (Scheduler)	18
Table 10: Use Case (Daily Schedule)	18
Table 11: Use Case (Calendar)	19

1 Introduction

In our current era of non-stop technological innovation, fuzzy wishful thinking has yielded to the very core of our mind. It is no longer enough to imagine our way to a better state of mind and body. We must now chart our progress, count our steps log our sleep rhythms, tweak our diets, record our negative thoughts – then analyze the data, calibrate and repeat.

2 Project Planning

At first we had everything planned, thinking it was simple by doing something everyday and it kind of was, until it wasn't. Not everything went as we initially planned. As you can see, the red squares means we used extra time for doing something that we expected to finish earlier. Also, some of them were started earlier because we needed to start some task before we could continue.

Planning is easier said than done.

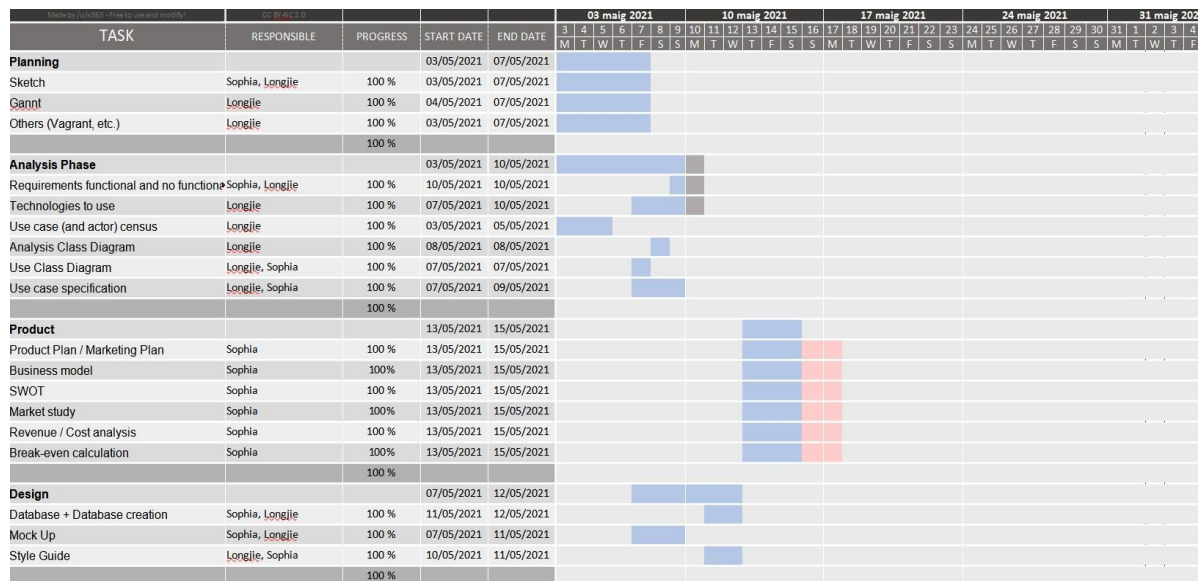


Figure 1: Gantt Diagram (1)

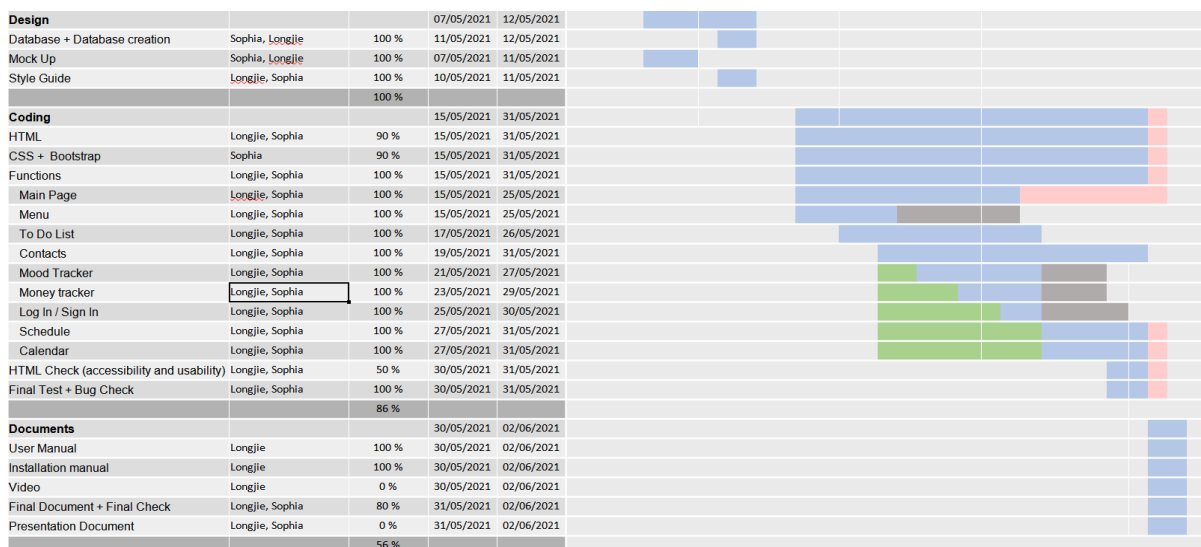


Figure 2: Gantt Diagram (2)

3 Marketing

3.1 Executive Summary

We live in a world of technology that is both a curse and blessing. Sadly enough, people are connected than ever, but feel less connected at the same time.

Trackr is a unique freemium tracker that provides an effective way on how to manage your time wisely.

Trackr is not your typical tracker where you can only use it as a self-tracking device. It is an extension of one's life to effectively manage our time to our own rhythm.

3.2 Products and services

- Mood tracker with integrated diary
- Financial tracker
- To-do list
- Scheduler

3.3 Mission

Trackr's mission is to provide a rather friendly and useful application that people can use to manage their lives more efficiently and effectively, to not lose time and rather spend it wisely.

3.4 Vision

We want the world to be more social. We live in a busy world that loses the personal touch of human connection, so we thrive to create a tool to help people manage their time and spend it wisely with the things and people they love. Life is too short to be too busy.

3.5 Competition

- Our biggest competitors are the paper based mood trackers and agendas.
- Single functional applications

3.6 Competitive advantage

We live in a technological world where a mobile phone is an essential and an extension of our lives. Comparing to our biggest competitors which are mostly paper based, our application has a lot of potential to integrate in our daily lives. Nonetheless, it is easily portable and you can always use it knowing it is on your phone.

3.7 SWOT Analysis

3.7.1 Strengths

Trackr is an application which offers a lot of utilities and functionalities that are needed for our daily necessities.

3.7.2 Weaknesses

There are lots of well-known applications that only focus on one thing.

3.7.3 Opportunities

Having the application means that no more managing too many applications.

3.7.4 Threats

A lot of different competitors in different fields of the application are prevalent. While having a lot of functionalities, the application is a jack of all trades, master of none.

3.8 Buyer persona

Trackr's customers can be broken down into two groups, busy professionals and students and your everyday consumers. The former are smaller than the latter by the fact that the former can and will utilize the application to its maximum potential while the latter can use the application based on their own needs.

The larger group of everyday consumers consists of:

- Age: 16-60. People tend to be the busiest.
- Marital Status: Single to Widowed
- Area of living: Anywhere in the world
- Women tend to use the application more than men due to their motherly nature.

3.9 Marketing objectives

- Develop strong brand awareness: Invite a friend promotion. Local news feature.
- Create brand loyalty: Be a go to application in which people use on their daily lives.
- Increase in sales and decrease in marketing expenses: Holiday promotions, Student / enterprise promotion

3.10 Marketing budget

I calculated that the business will need to make at least 50.000€ a month in revenue to reach the break-even point. The first two months will be used to set up the physical location, hire personnel, rent servers and obtain a patented license. Business during the second month will be understandably slow as a steady customer base takes time to build. This is the stage where the majority of our marketing budget will go.

3.10.1 Development Schedule

What: Marketing campaign and Social Media campaign

Who: Longjie

Objectives: Marketing, Socials

Budget: 20.000€

What: Sales research

Who: Sophia

Objectives: Sales

Budget: 10.000€

3.10.2 Marketing Mix

Product: Mood tracker with integrated diary, Financial tracker, To-do list, Scheduler

Price: Free, single purchase: 9.99€, promotional prices: 4.99€

Distribution: App Store, Play Store and Official Website

Promotion: Social media

TRACKR

3.10.3 Social media

Instagram: Purchase of sponsored ads targeting our specified list of interests. Influencer posts. Heavy use of hashtag promotion.

Facebook ads: Purchase of sponsored ads targeting our specified list of interests. Influencer posts. Heavy use of hashtag promotion.

Google ads: Purchase of sponsored ads targeting our specified list of interests.

When it comes to collaborations with schools and enterprises, there can be cross-promotional discounts! When looking for influencers we wanted to work with we would hyper-focus on ones with at least 20K followings, and topics that are of interest to us.

4 Analysis Phase

4.1 Nonfunctional requirements

- **Compatibility:** As a software it needs to be compatible with different Operating System and platforms.
- **Conformance:** It needs to at least meet the minimum requirements of the product
- **Dependability:** As a software that saves personal data, it needs to be safe and secured.
- **Installation:** A software needs to be easy to install.
- **Maintainability:** A software that is not maintainable is no better than garbage.
- **Performance:** Every software must always put performance at the top of the list if not the utmost top.
- **Portability:** As a modern software, it needs to be at least compatible with mobile phones.
- **Reliability:** The software must work at all times
- **Usability:** A software must be easy to use.
- **Security:** A software must always have some sort of secure and data encryption at all times.

4.1.1 Technologies

- **Web Server Software**
 - **Apache2:** One of the most popular web server software. It allows us to save the files and be able to connect to the server.
- **Database Engine**
 - **MariaDB:** Used to create the entire database and used to save all the data of the users and their tasks.
- **Languages**
 - **PHP:** One of the most popular back-end languages for web. It allow us to keep users session, connect to the database, create interaction, encrypt data, etc.
 - **HTML:** the basic language to create a website.
 - **JavaScript:** front-end programming language used to make the website more interactive and for example, to make the user create more rows.
 - **Bootstrap/CSS:** all the visuals of the web site are made with Bootstrap and CSS.

4.2 Functional requirements

4.2.1 Use Case Actor Census

Table 1: Use Case Actor Census

Actor	Responsibility	Requirement	Guide
User	A person who wants to use the application and wants to have a better life.	Any user that has email.	<p>The users can see their calendar</p> <p>The users can modify their schedules, to do list, mood tracker, financial tracker...</p>

4.2.2 Use Case Census

1. Sign up / Sign In
2. Manage Contact list
 - Add, Modify and Delete Contacts
3. Manage To Do List
 - Add and Modify To Do List
 - Set / Change Tag name
 - Manage Tasks
 - Check / Uncheck Important tasks
 - Set Contact
 - Set / Change Description
 - Set type of the To do list
 - Normal
 - Challenge
 - Delete To Do List
4. Manage Mood Tracker
 - Select daily mood
 - Show Mood Calendar

TRACKR

- Set daily mood
- 5. Manage Financial Tracker
 - Enable and Disable function
 - Set Total and Income
 - Set Daily Expenses
 - Set Expenses name and quantity
 - Show Money Stats and Graphic
- 6. Manage Scheduler
 - Add / Change daily entries
 - Delete entries
- 7. Show current schedule
 - Show current Schedule
 - Show current To do list
- 8. Show Calendar
 - Show Schedule
 - Show To Do List
 - Show Exposures and Income
 - Show mood

4.2.5 Use Case Specification

Table 2: Use Case (Sign Up)

Use Case	Sign up
Pre-condition	<ul style="list-style-type: none"> • The user doesn't have an account to access the application. • The user must have an email to register to the application • The website must be online
Post-condition	<ul style="list-style-type: none"> • Account created for the user to access the application
Basic Path	<ol style="list-style-type: none"> 1. The user puts its email, password and password confirmation to create an account. 2. The system checks if both password coincide. 3. The system saves the data to its database
Alternative paths	<p>2a The passwords don't coincide.</p> <p>2b Email already in use</p> <p>2c Invalid email format</p>
Exception paths	<ul style="list-style-type: none"> • Cannot create the account • Cannot create new account • The application cannot connect to the database

Table 3: Use Case (Sign In)

Use Case	Sign in
Pre-condition	<ul style="list-style-type: none"> • The user must have an account to access the application • The website must be online
Post-condition	<ul style="list-style-type: none"> • The user accessed the application and its functionalities.
Basic Path	<ol style="list-style-type: none"> 1. The user enters his email and password. 2. The system verifies his email and password, comparing to the

TRACKR

	<p>database's.</p> <p>3. The email and password are both correct</p>
Alternative paths	<p>3a. The email or account doesn't exist.</p> <p>3b. The password is incorrect</p>
Exception paths	<ul style="list-style-type: none"> • Cannot access to the application • The application cannot connect to the database

Table 4: Use Case (Contact List)

Use Case	Manage contact list
Pre-condition	<ul style="list-style-type: none"> • The user already signed in • The website must be online
Post-condition	<ul style="list-style-type: none"> • The user added a new contact • The user updated a contact • The user deleted a contact • The database was updated • The user sees the information
Basic Path	<ol style="list-style-type: none"> 1. The user go to the Contact List page 2. The system gets the information of the users contact 3. The user manages the contact list: <ol style="list-style-type: none"> 1. The user wants to add a new contact <ol style="list-style-type: none"> 1. The user clicks the button to add a new contact 2. The system shows the user a screen to add a new contact. 3. The user enters the new contact information (Name, Last name, Phone, Email, etc) and confirms 4. The system adds the new contact with the information given by the user to the database 2. The user wants to update a contact

TRACKR

	<ol style="list-style-type: none"> 1. The user clicks the edit button of the specified contact 2. The system shows the user a screen to edit the specified contact. 3. The user changes the information of the contact and confirms 4. The system updates the information changed by the user to the database <ol style="list-style-type: none"> 3. The user wants to delete a contact <ol style="list-style-type: none"> 1. The user clicks the delete button of the interested contact 2. The system asks if the user wants to delete the contact or not 3. The user chooses “yes” 4. The system deletes the contact from the database 5. The system updates and now shows the contact page without the specified contact
Alternative paths	<p>2a. The application shows no information of the users contact, because there is none</p> <p>3.1.3a The user denies on adding a new contact</p> <p>3.2.3a The user denies on updating the contact</p> <p>3.3.3a The user chooses “no”</p> <p>3.3.4b The system doesn’t delete the contact</p>
Exception paths	<ul style="list-style-type: none"> • The application cannot connect to the database

Table 5: Use Case (To do Folders)

Use Case	Manage To Do list (folders)
Pre-condition	<ul style="list-style-type: none"> • The website must be online • The user already signed in
Post-condition	<ul style="list-style-type: none"> • The user added a new folder • The user updated a folder • The user deleted a folder

TRACKR

	<ul style="list-style-type: none"> • The user enters to a folder • The database was updated • The user sees the information
Basic Path	<ol style="list-style-type: none"> 1. The user go to the To do list page 2. The system loads the “folders” and shows the user 3. The user manages the folders: <ol style="list-style-type: none"> 1. The user enters to a folder <ol style="list-style-type: none"> 1. The user clicks to the interested folder and enters 2. The application shows the tasks within the folder 2. The user adds a new folder <ol style="list-style-type: none"> 1. The user clicks on “add new folder” button 2. The system gives to the user a form to set a name 3. The user puts a names for the new folder and confirms 4. The system creates a new folder and inserts into the database 3. The user renames a folder <ol style="list-style-type: none"> 1. The user clicks edit on the specified folder 2. The system leads the user to another page 3. The system loads the old name of the folder 4. The user changes the old name of the folder and confirms 5. The system updates the folder’s name and updates into the database 4. The user deletes a folder <ol style="list-style-type: none"> 1. The user clicks delete on the interested folder 2. The system asks the user to delete the folder 3. The user accepts 4. The system deletes the folder from the database and reloads the page
Alternative paths	<p>2a There’s no folders</p> <p>3.2.3a The user denies on adding a new folder</p>

TRACKR

	<p>3.2.4a The system doesn't add a new folder</p> <p>3.3.4a The user denies on updating the folder</p> <p>3.3.5a The system doesn't update the folder</p> <p>3.4.3a The user denies on deleting the folder</p> <p>3.4.4a The system doesn't delete the folder</p>
Exception paths	<ul style="list-style-type: none"> The application cannot connect to the database

Table 6: Use Case (Tasks and Challenges)

Use Case	Manage To Do list (tasks and challenges)
Pre-condition	<ul style="list-style-type: none"> The website must be online The user already signed in There must be at least a folder created first The user must select a folder first
Post-condition	<ul style="list-style-type: none"> The user added a new task/challenge The user modified a task/challenge The user deleted a task/challenge The database was updated The user sees the information
Basic Path	<ol style="list-style-type: none"> The user chooses the folder The system shows the user tasks or challenges created The user manages the tasks: <ol style="list-style-type: none"> The user adds a new task <ol style="list-style-type: none"> The user clicks on add new task The system gives the user a form about the task The user enters all the information (name, important check, date, time and contact). The system gets the information and inserts it to the database The user modifies a task

TRACKR

	<ol style="list-style-type: none"> 1. The user clicks on the specific task to update 2. The system loads the task information to update 3. The user changes the information about the task and confirms 4. The System updates the information to the database <ol style="list-style-type: none"> 3. The user deletes a task <ol style="list-style-type: none"> 1. The user clicks on the specific task to delete 2. The system asks the user to delete 3. The user confirms it. 4. The system deletes the task from the database
Alternative paths	<p>2a There's no task or challenge</p> <p>3.1.3 a The user changes the type of the task to "challenge" and enters the information (only name)</p> <p>3.1.4 a The system didn't add a new task</p> <p>3.2.3 a The user changes task to normal or to challenge type, and the system changes the form</p> <p>3.2.4 a The system didn't update the tasks</p> <p>3.3.3a The user denies the confirmation</p> <p>3.3.4a The system didn't delete the task</p>
Exception paths	<ul style="list-style-type: none"> • The application cannot connect to the database

Table 7: Use Case (Mood Tracker)

Use Case	Manage mood tracker
Pre-condition	<ul style="list-style-type: none"> • The website must be online • The user already signed in
Post-condition	<ul style="list-style-type: none"> • The user set their mood of the day • The user set a description of their day • The database updated the data
Basic Path	<ol style="list-style-type: none"> 1. The user goes to the mood tracker page

TRACKR

	<ol style="list-style-type: none"> The system shows the page to the user The user selects his mood and writes a description of the day The user clicks on the confirm button The system adds a new entry to the database
Alternative paths	<p>2a The system loads the mood and the description wrote previously</p> <p>5a The system updates the entry if it already exists</p>
Exception paths	<ul style="list-style-type: none"> The application cannot connect to the database

Table 8: Use Case (Financial Tracker)

Use Case	Manage Financial tracker
Pre-condition	<ul style="list-style-type: none"> The website must be online The user already signed in The Financial tracker is enabled
Post-condition	<ul style="list-style-type: none"> The user sees his information about the expenses, income and total The user sees his information about all the different expenses and incomes
Basic Path	<ol style="list-style-type: none"> The user goes to the financial page The system shows all of his financial transactions The user adds their expenses or income The system inserts the entry from the database
Alternative paths	<ol style="list-style-type: none"> The user click on the old transaction button The system shows the user's monthly transactions
Exception paths	<ul style="list-style-type: none"> The application cannot connect to the database

TRACKR

Table 9: Use Case (Scheduler)

Use Case	Manage Scheduler
Pre-condition	<ul style="list-style-type: none"> • The website must be online • The user already signed in
Post-condition	<ul style="list-style-type: none"> • The user set his schedule • The database was updated
Basic Path	<ol style="list-style-type: none"> 1. The user goes to the scheduler page 2. The system shows the scheduler 3. The user chooses which day of the week will they update 4. The system shows the information about that day 5. The user adds, modifies or deletes entries and confirms 6. The system makes changes on the database
Alternative paths	5a The user doesn't want to make the changes
Exception paths	<ul style="list-style-type: none"> • The application cannot connect to the database

Table 10: Use Case (Daily Schedule)

Use Case	Show Current schedule (Main page)
Pre-condition	<ul style="list-style-type: none"> • The website must be online • The user already signed in
Post-condition	<ul style="list-style-type: none"> • The user sees his data • The user sees their daily Schedule and its information (Scheduler and To do)
Basic Path	<ol style="list-style-type: none"> 1. The user goes to the main page 2. The system loads up the database all information about the scheduler and to do list 3. The system uses those information to make a list and shows to the user

TRACKR

Alternative paths	
Exception paths	<ul style="list-style-type: none"> The application cannot connect to the database

Table 11: Use Case (Calendar)

Use Case	Show Calendar
Pre-condition	<ul style="list-style-type: none"> The website must be online The user already signed in
Post-condition	<ul style="list-style-type: none"> The user sees his data The user sees his monthly calendar and its information (To do, mood, financial, etc).
Basic Path	<ol style="list-style-type: none"> The user goes to the calendar page The system loads from the database the information about the scheduler, to do list, financial tracker and mood tracker The system put those information in a calendar and shows the user The user click on a certain date The system shows to the user a summary of that day
Alternative paths	
Exception paths	<ul style="list-style-type: none"> The application cannot connect to the database

5 Design

5.1 Database design

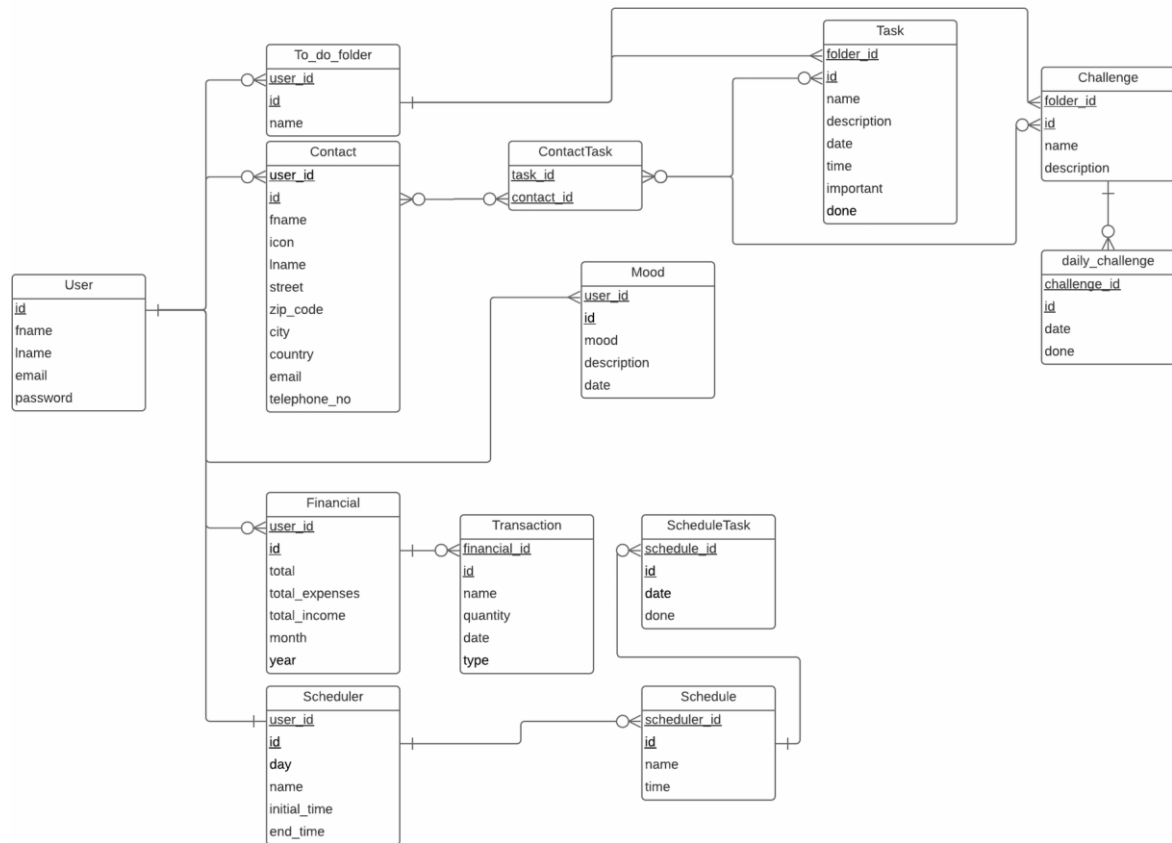


Figure 5: Database design

5.2 Sketch

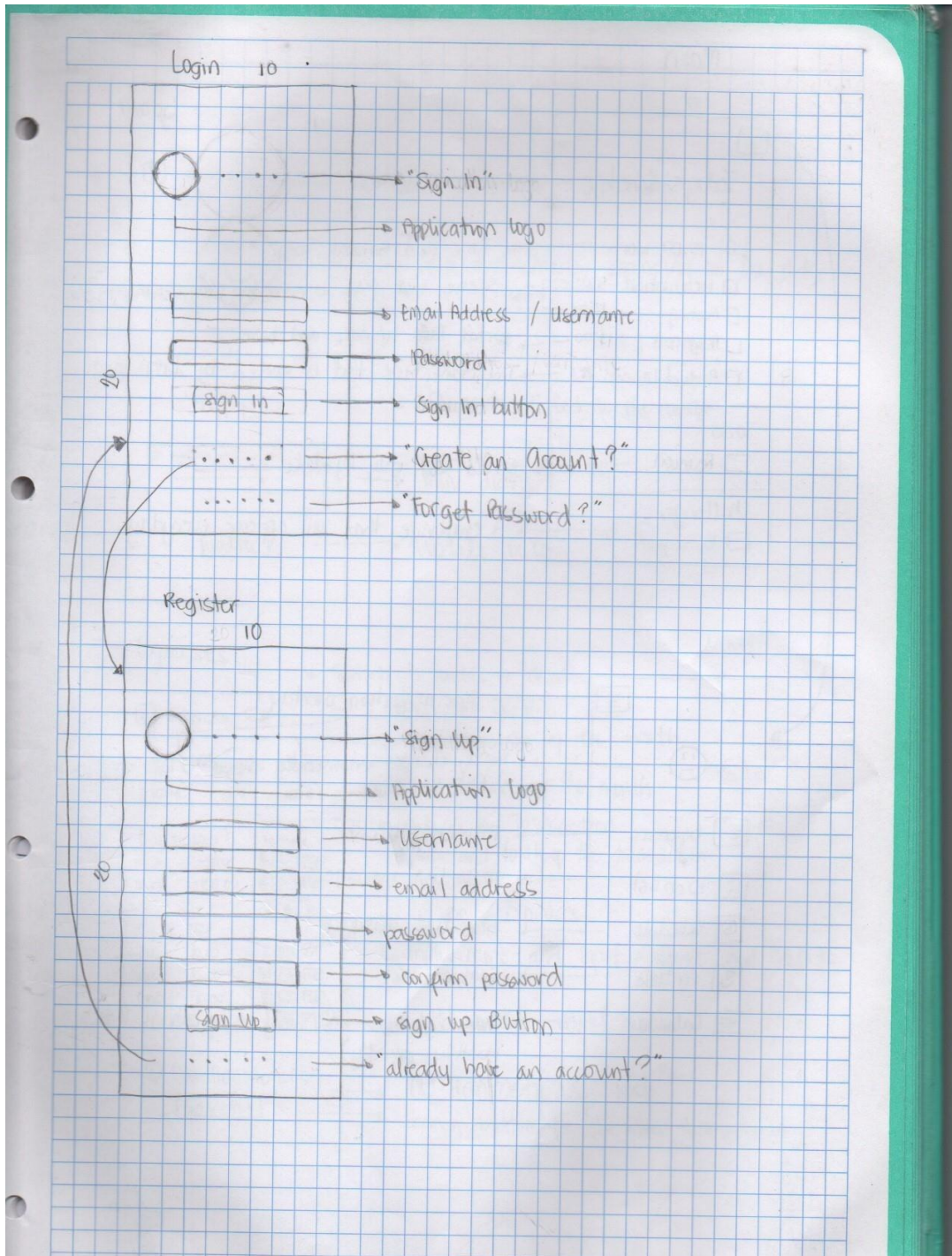


Figure 6: Sketch (Login and Registry)

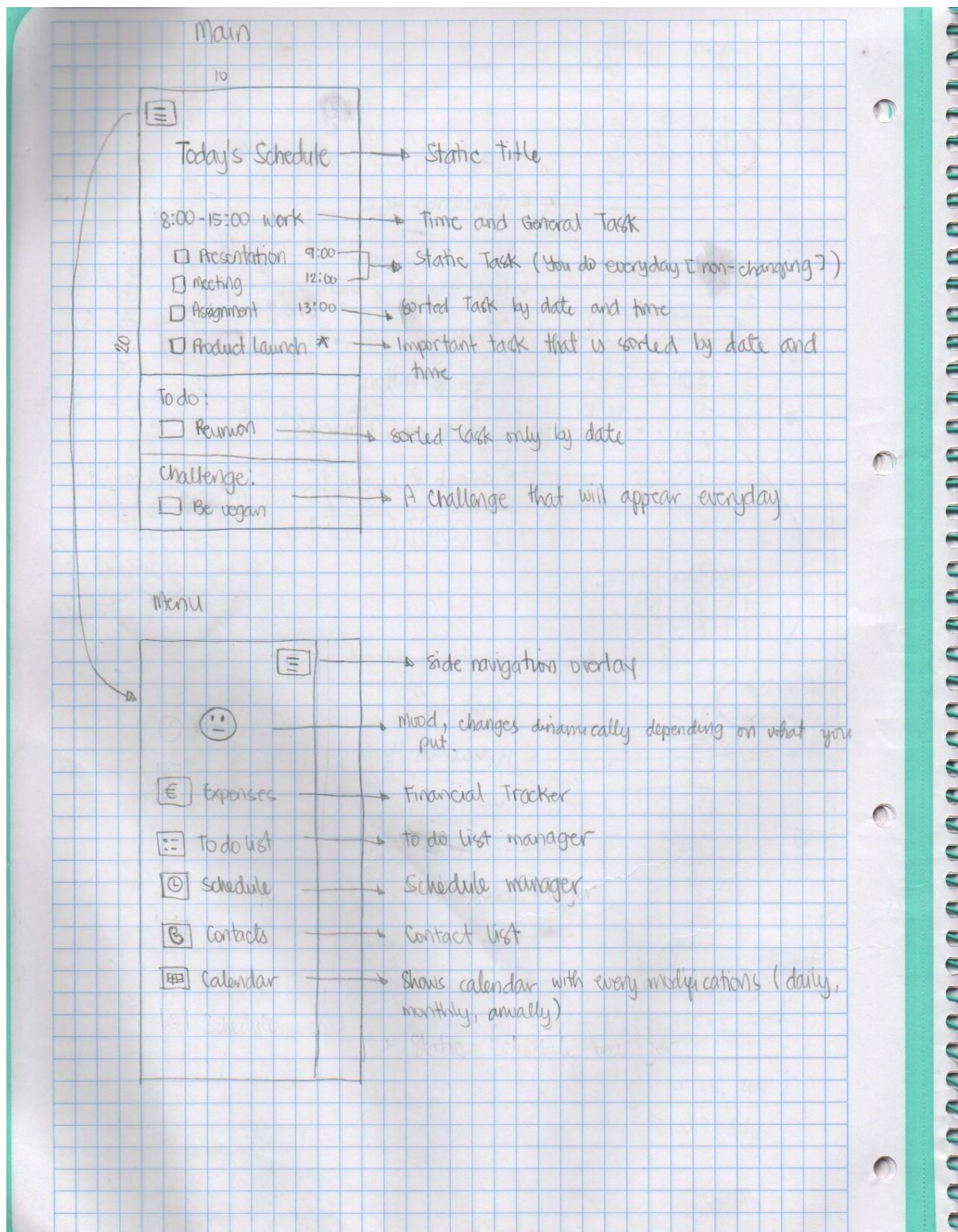


Figure 7: Sketch (Main and Menu)

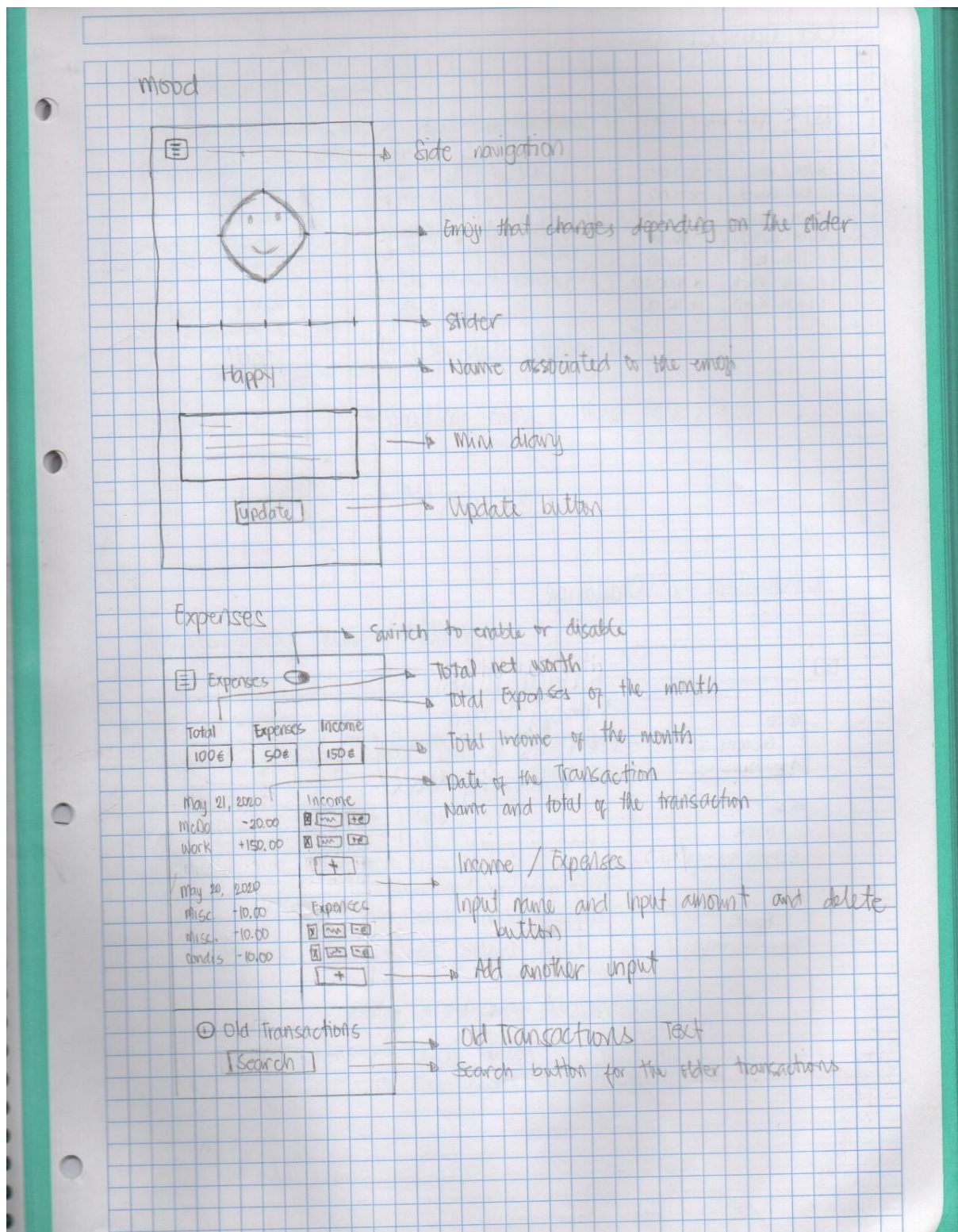


Figure 8: Sketch (Mood and Financial Transactions)

TRACKR

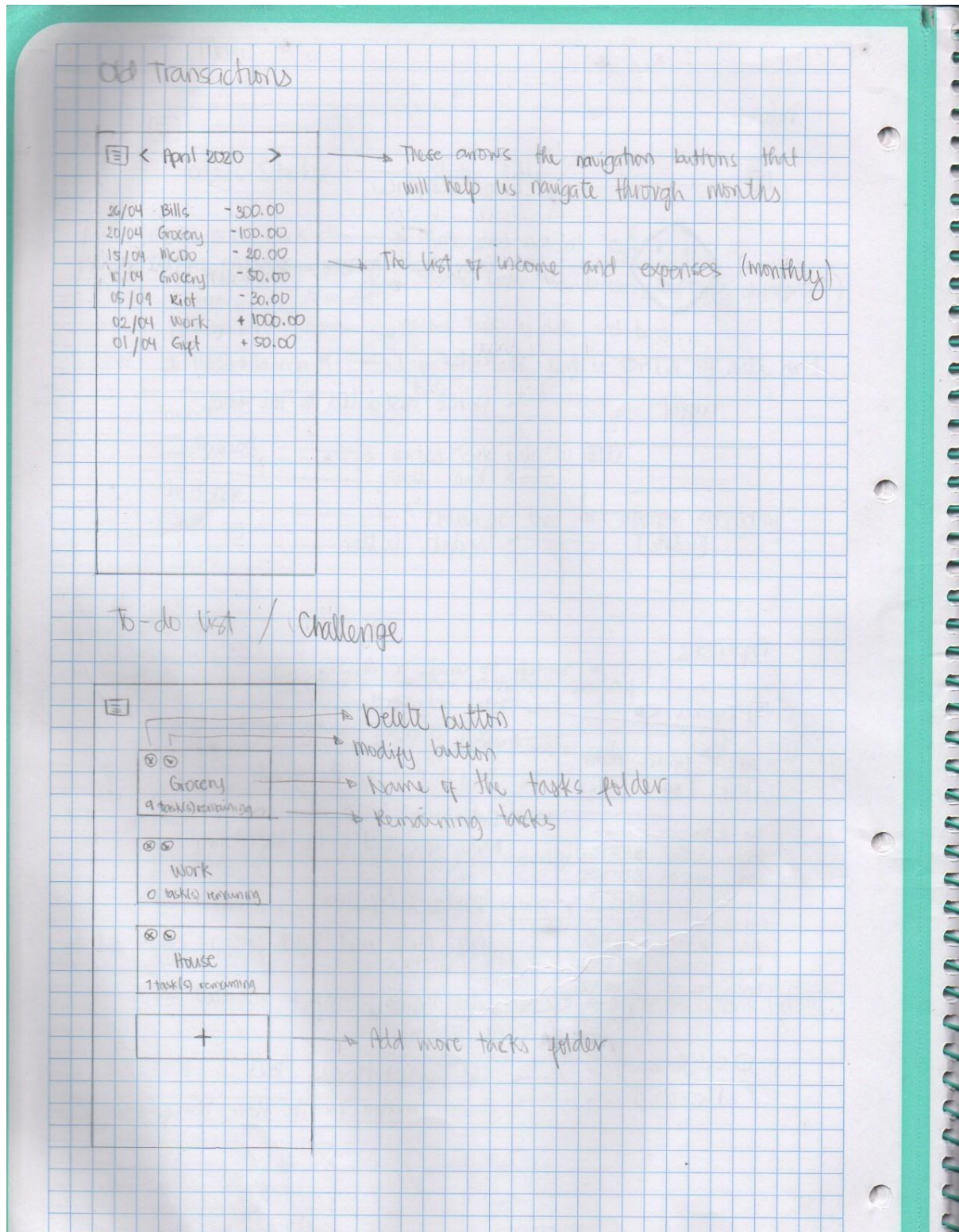


Figure 9: Sketch (Old Transactions and To do folder)

TRACKR

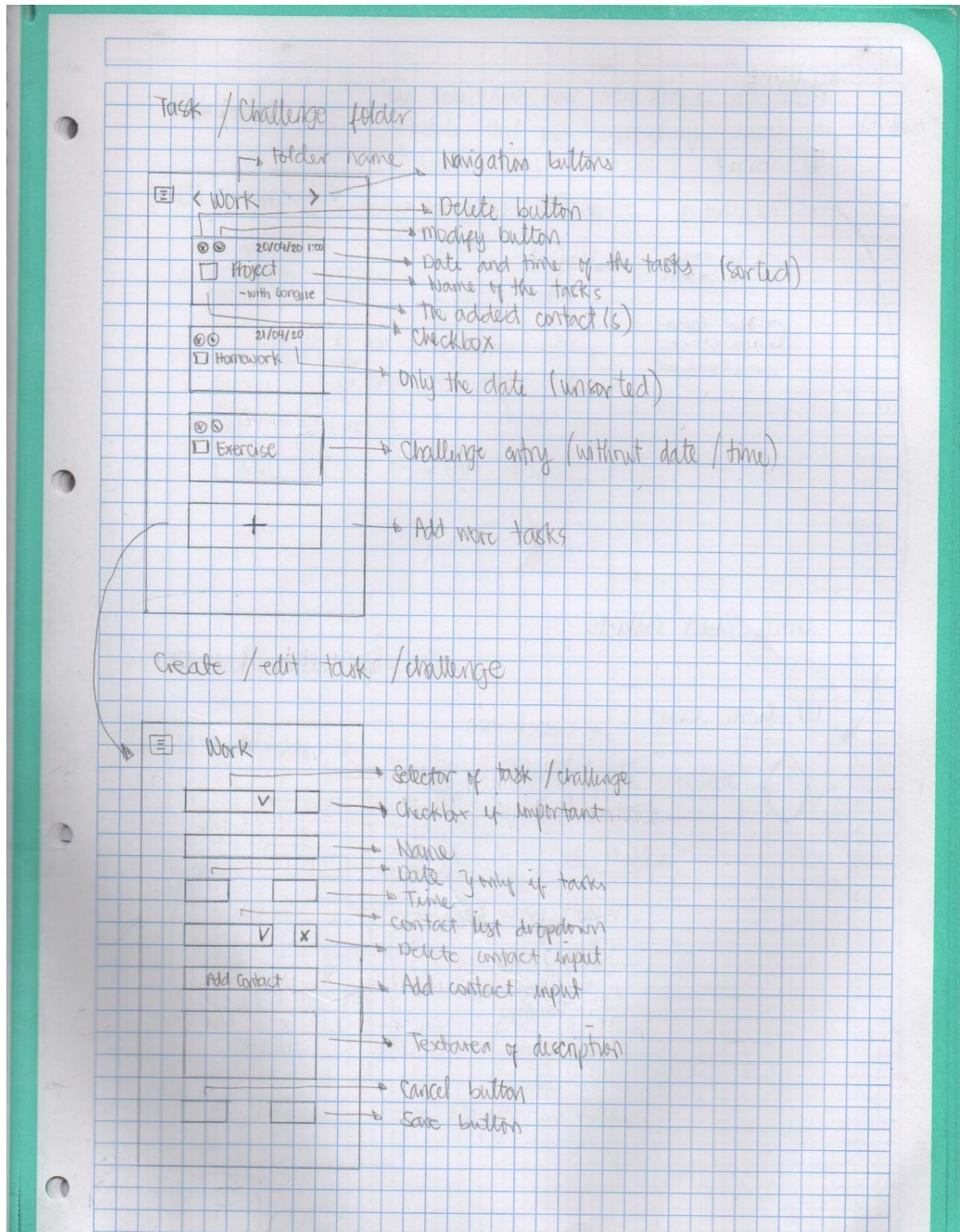


Figure 10: Sketch (Task and Challenge, and its CRUD)

TRACKR

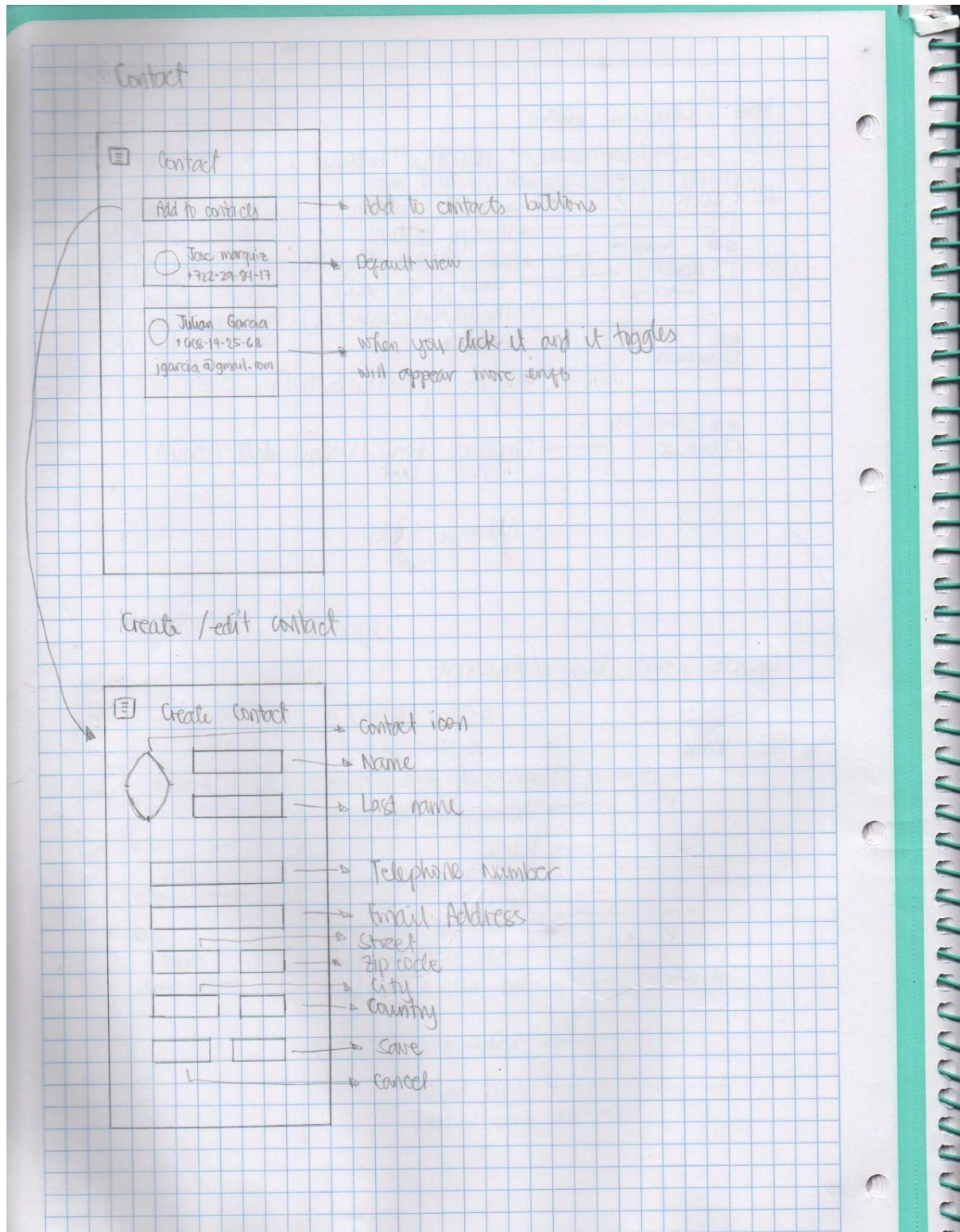


Figure 11: Sketch (Contact and its CRUD)

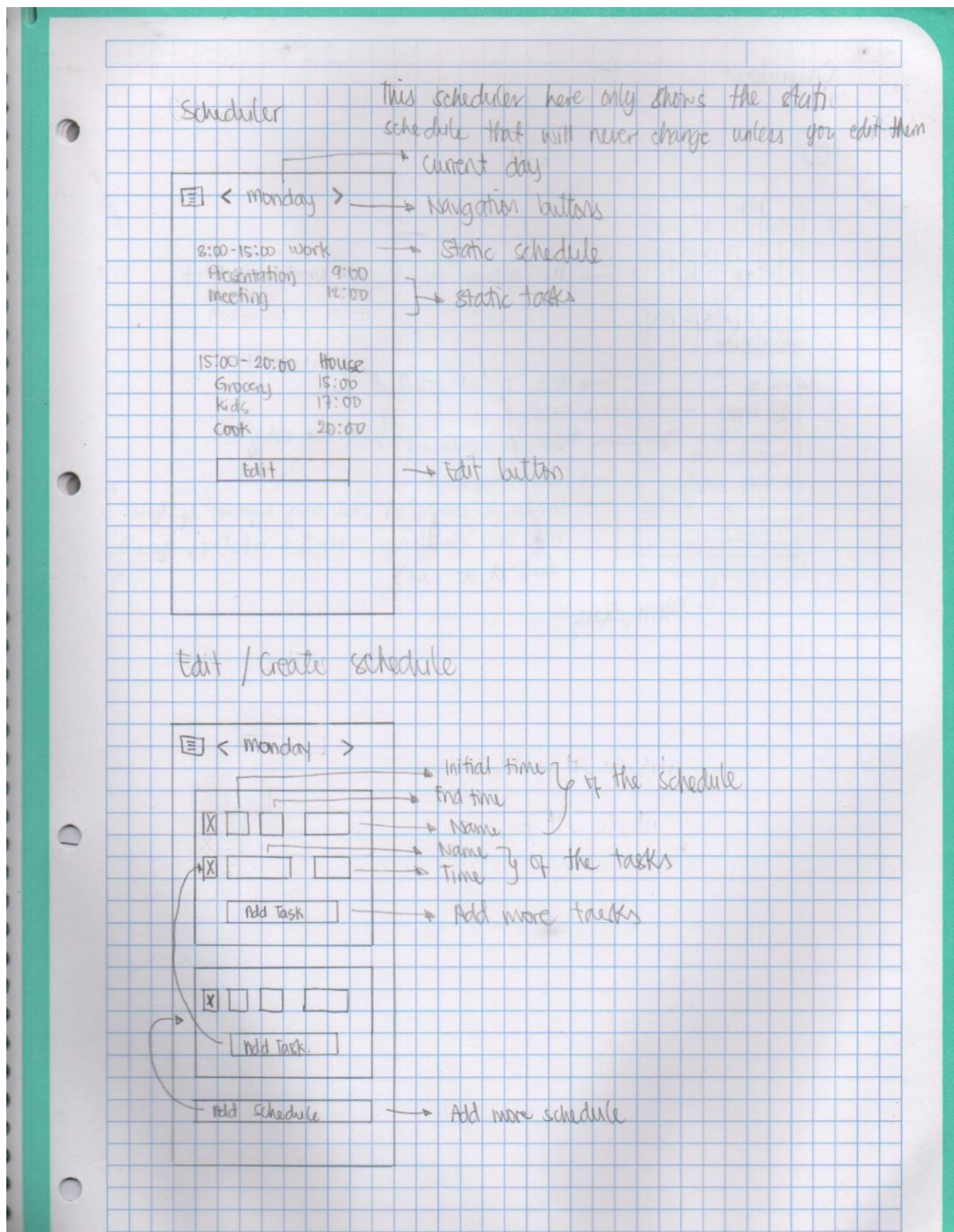


Figure 12: Sketch (Scheduler and its CRUD)

TRACKR

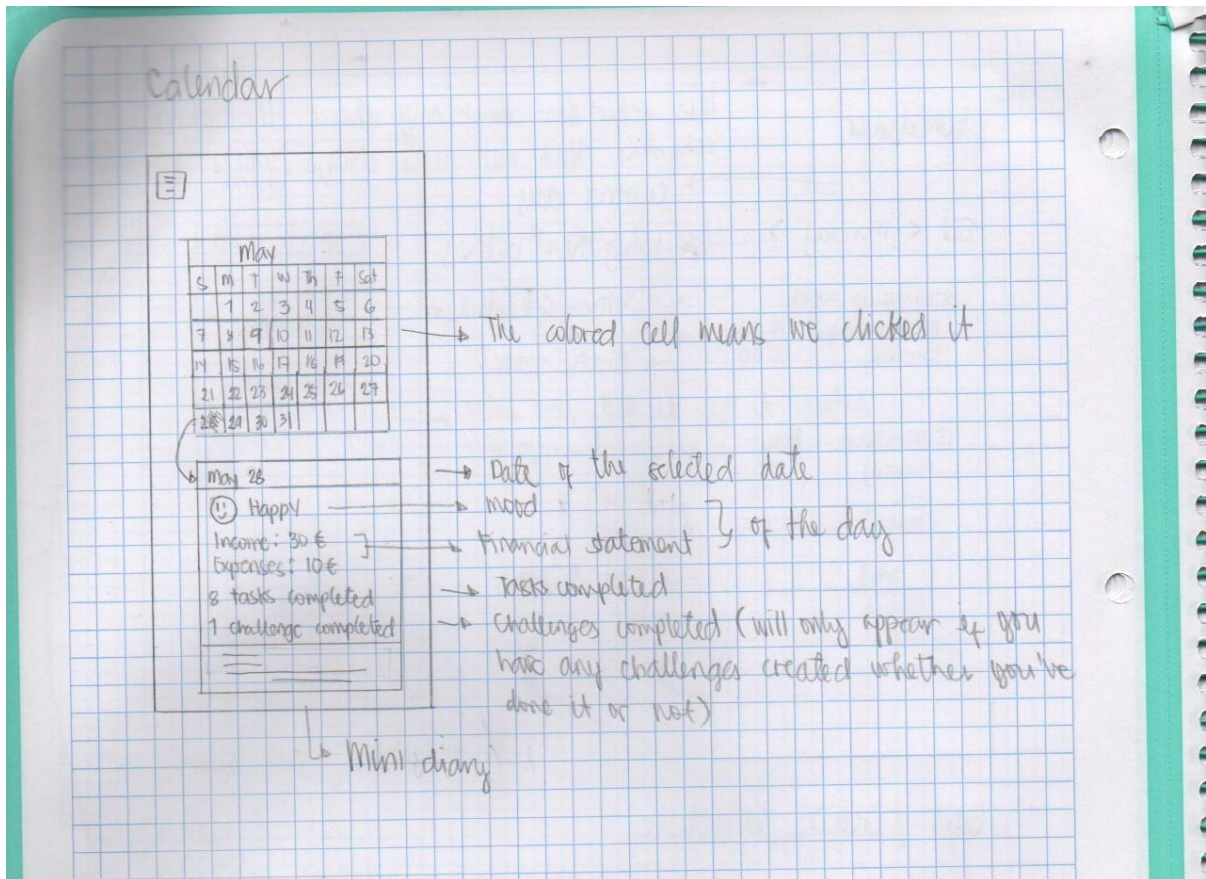


Figure 13: Sketch (Calendar)

TRACKR

5.3 Mockup

5.3.1 Login

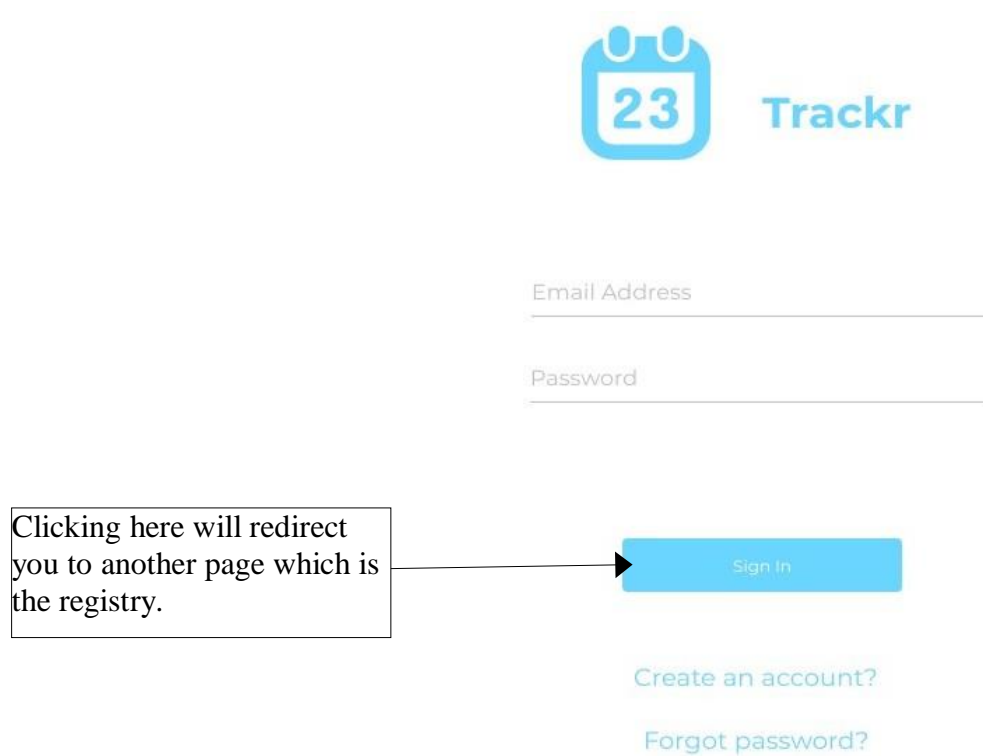



Figure 14: Mockup (Login)

TRACKR

5.3.2 Registry



Username

Email Password

Password

Confirm Password

Clicking here will redirect you back to login screen.

Sign Up

Already have an account?

Figure 15: Mockup (Registry)

5.3.3 Main page

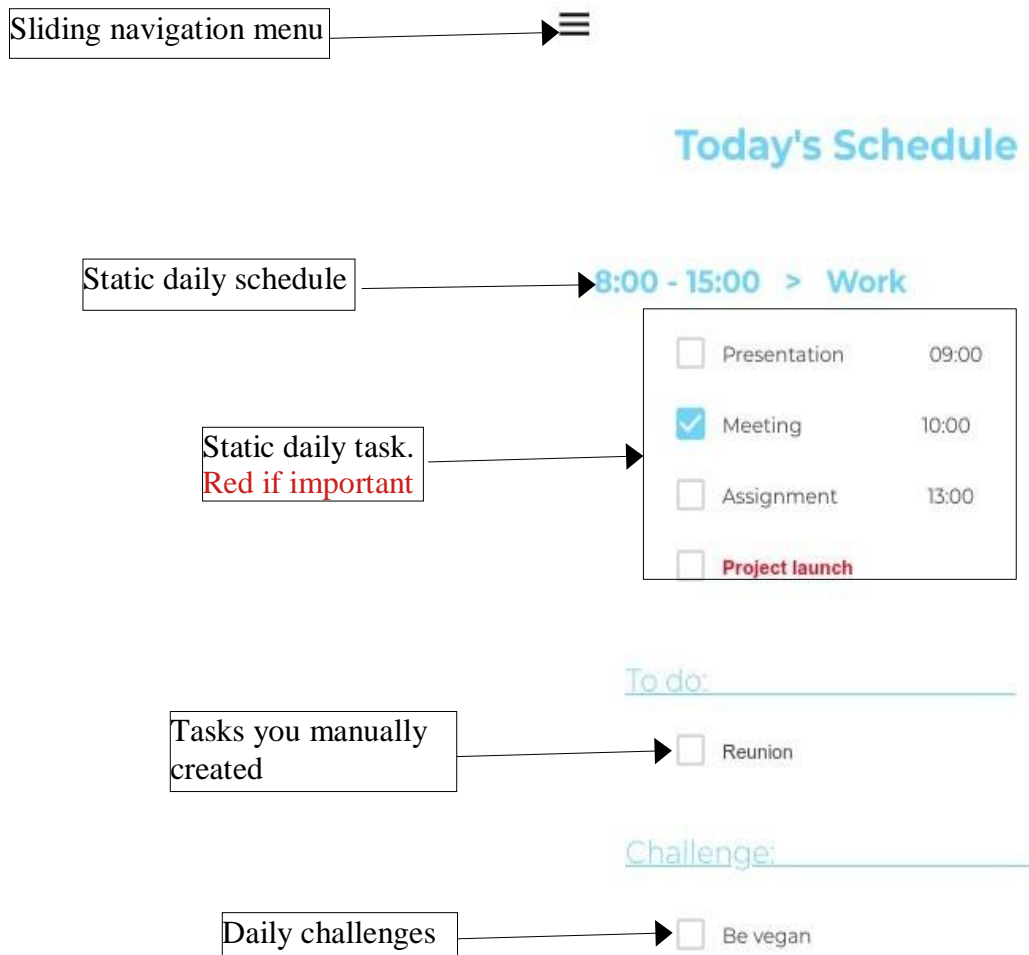
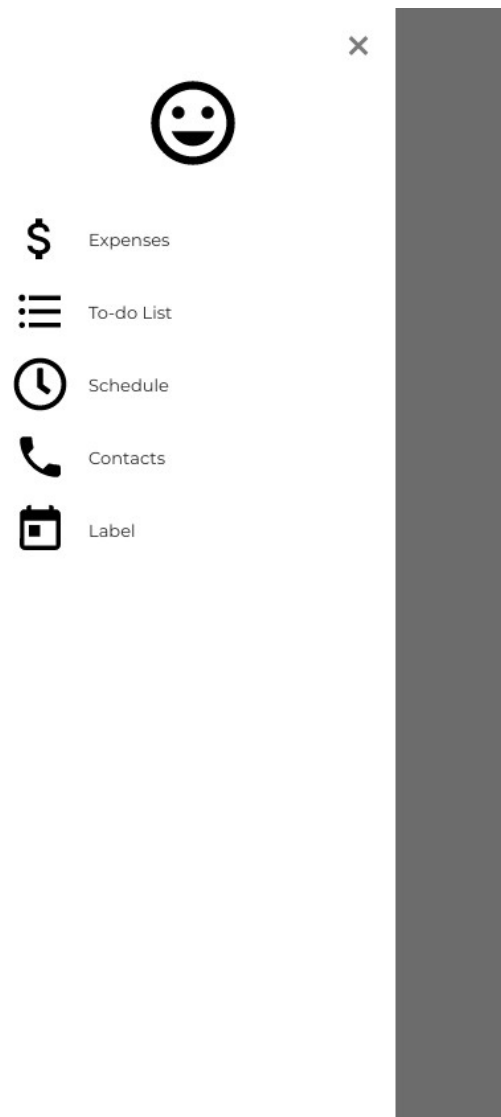


Figure 16: Mockup (Main)

TRACKR

5.3.4 Menu



5.3.5 Mood Tracker

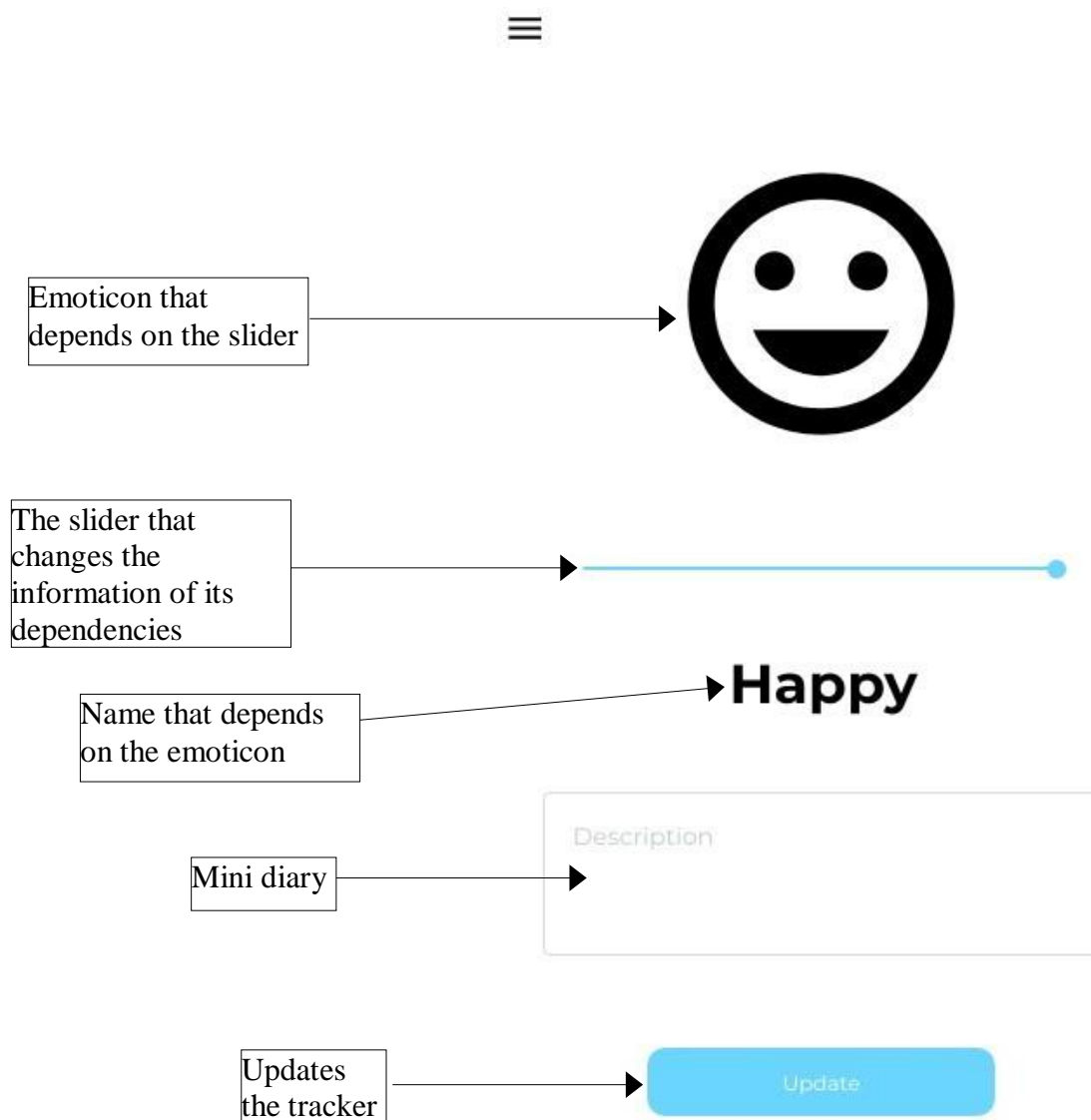


Figure 17: Mockup (Mood Tracker)

5.3.6 Financial Tracker

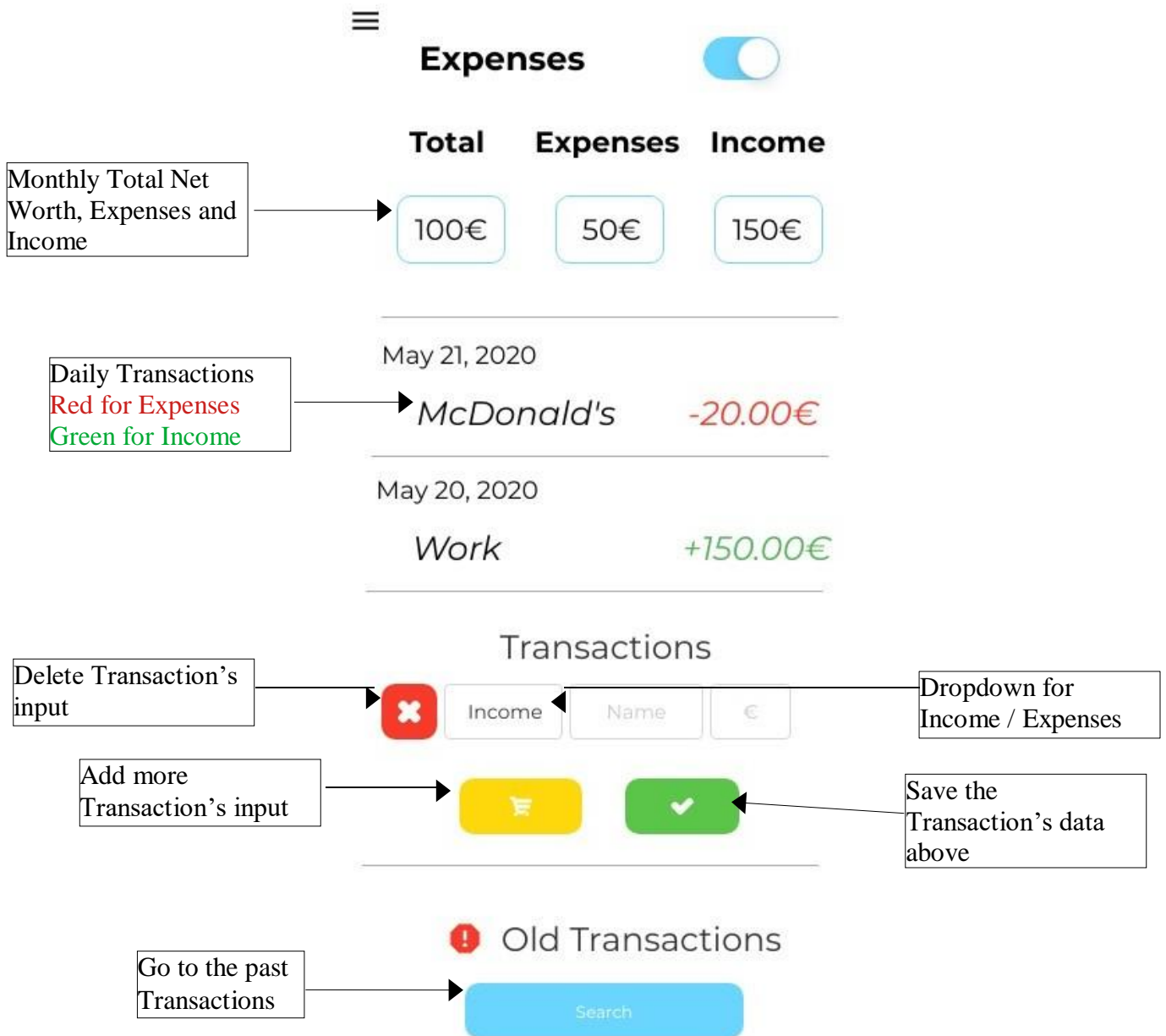
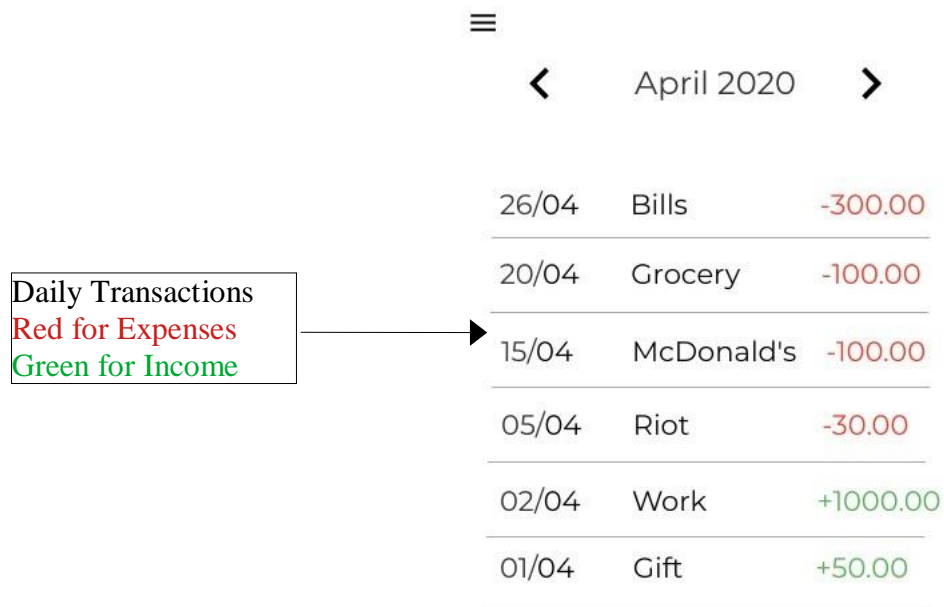


Figure 18: Mockup (Financial Tracker)

5.3.7 Old Transactions



Legend:

- Daily Transactions
- Red for Expenses
- Green for Income

≡		
< April 2020 >		
26/04	Bills	-300.00
20/04	Grocery	-100.00
15/04	McDonald's	-100.00
05/04	Riot	-30.00
02/04	Work	+1000.00
01/04	Gift	+50.00

Figure 19: Mockup (Old Transactions)

5.3.8 To do folder

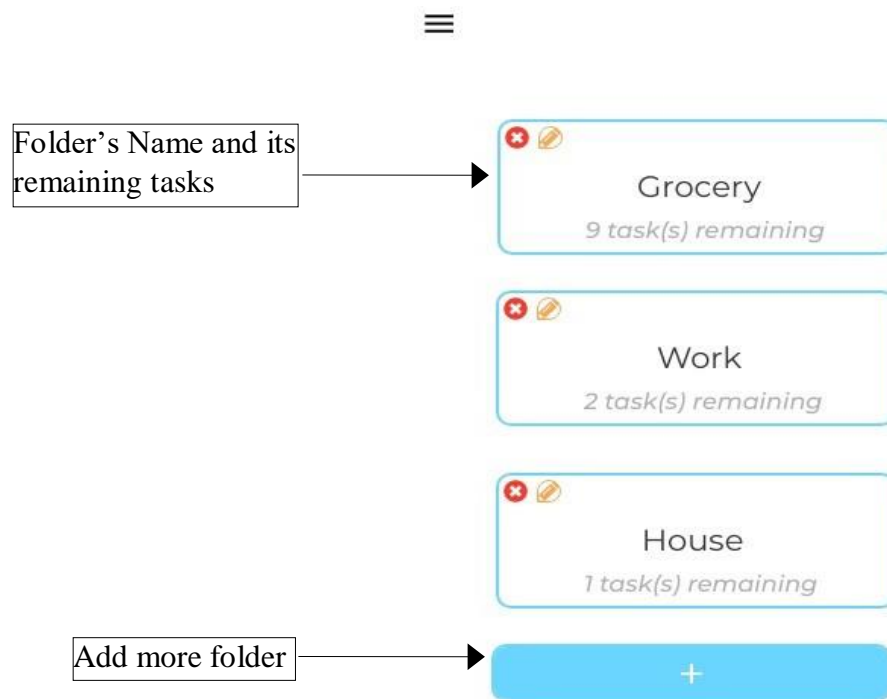


Figure 20: Mockup (To do folder)

5.3.9 Task and Challenges

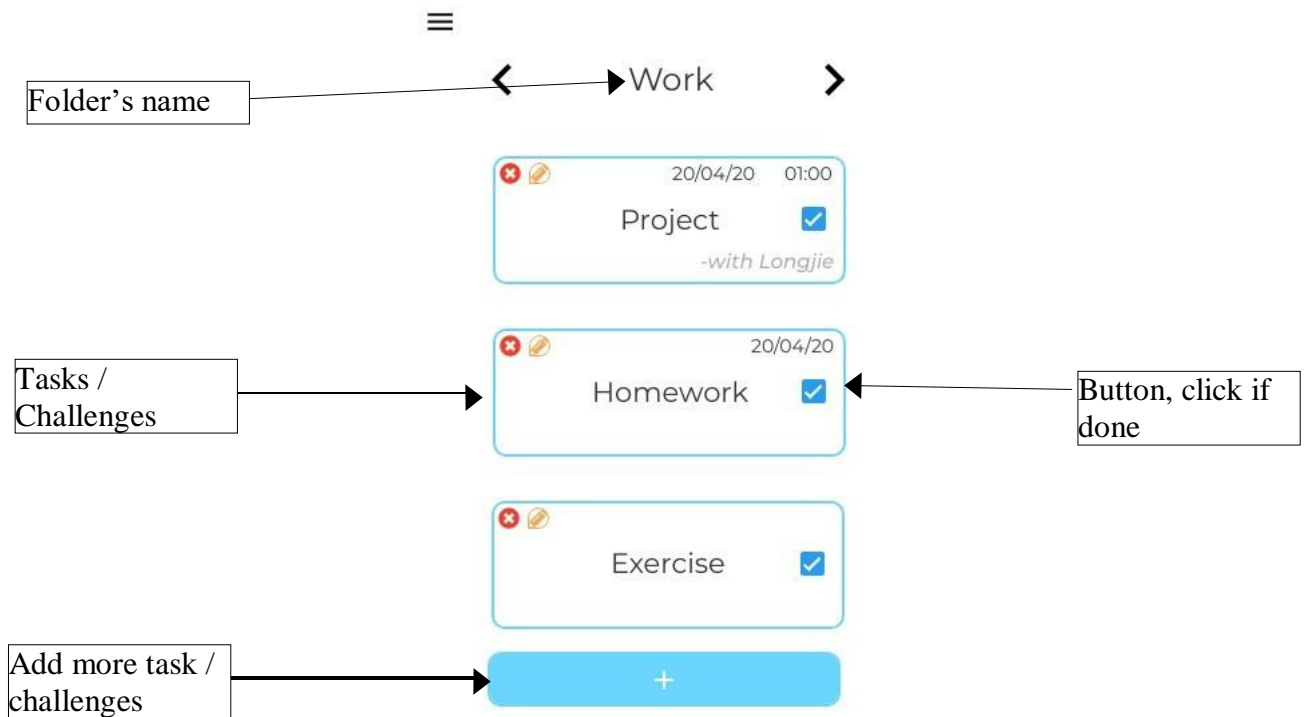


Figure 21: Mockup (Task and Challenge)

5.3.10 Task and Challenges' CRUD

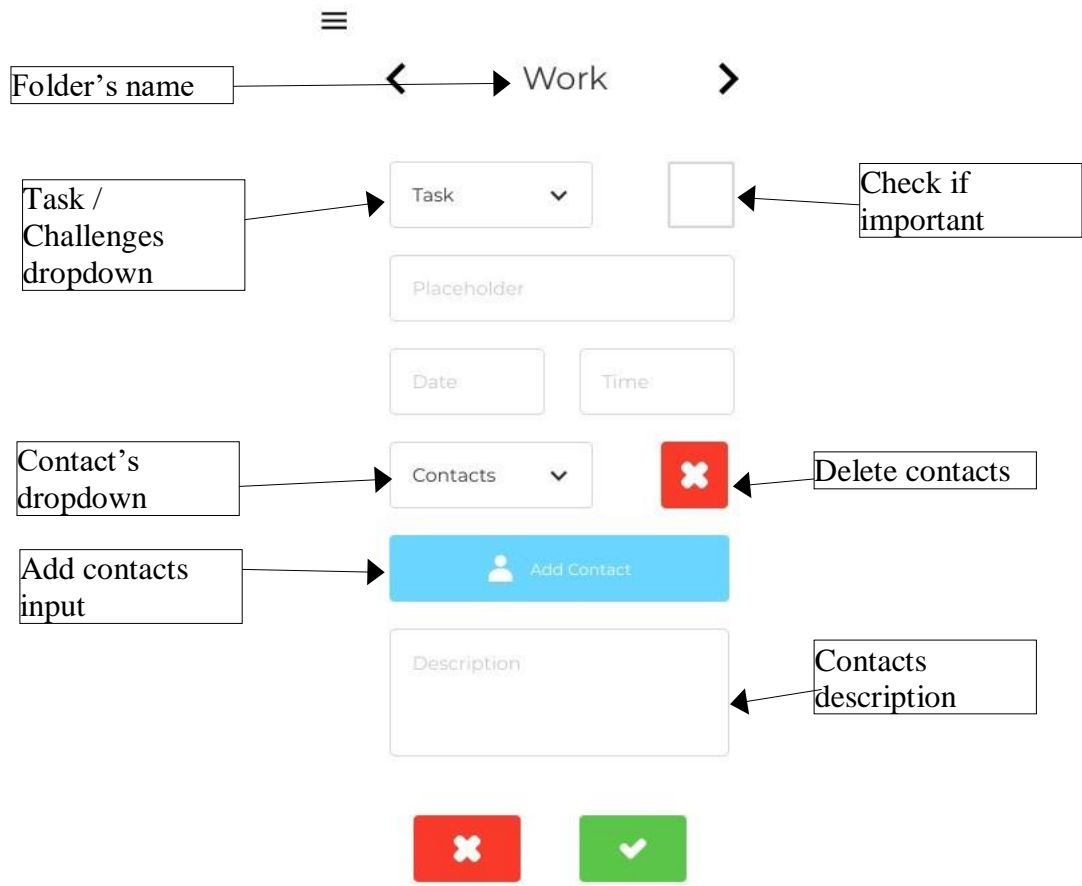


Figure 22: Mockup (Tasks and Challenges' CRUD)

5.3.11 Contacts

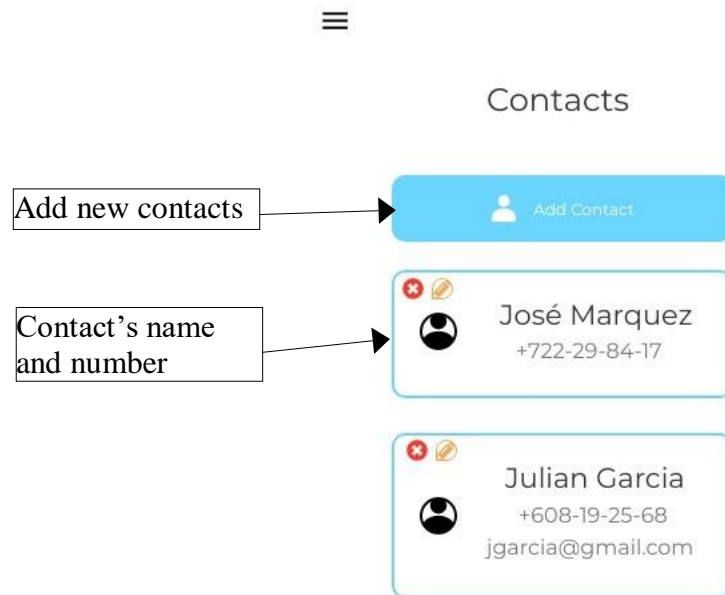



Figure 23: Mockup (Contacts)

5.3.12 Contacts' CRUD


≡


Work



 First Name

Last Name

 Telephone Number

 Email

 Street

Zip City

Country


 

Figure 24: Mockup (Contacts' CRUD)

5.3.13 Calendar



Figure 25: Mockup (Calendar)

5.3.14 Scheduler

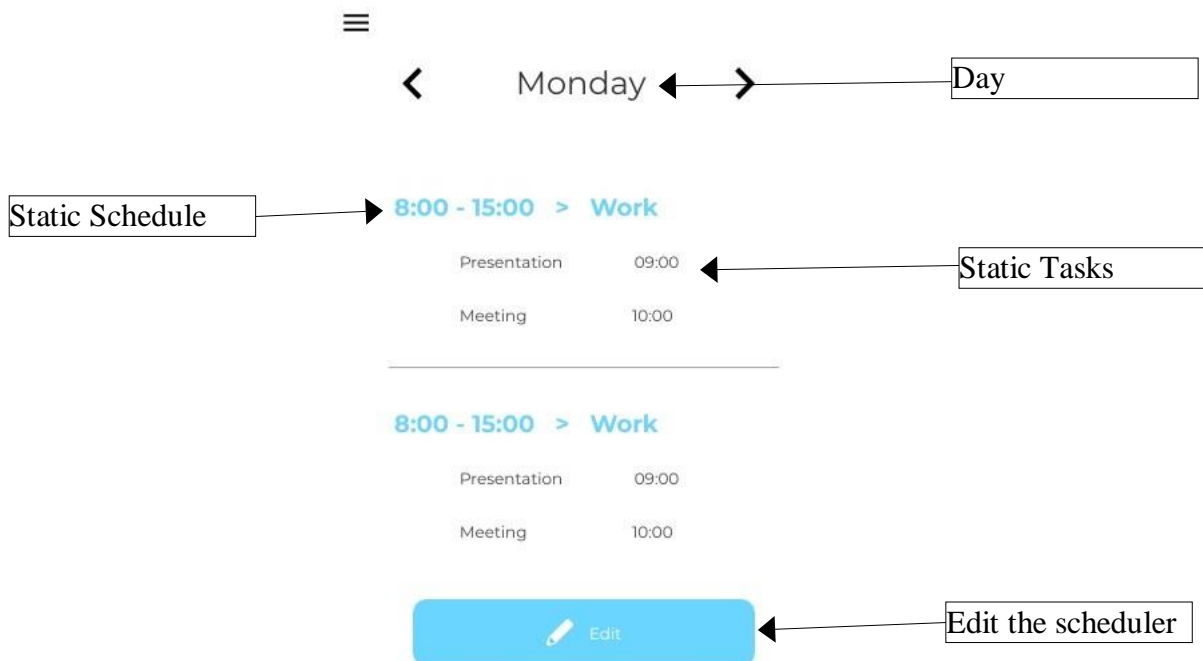


Figure 26: Mockup (Scheduler)

5.3.15 Scheduler's CRUD

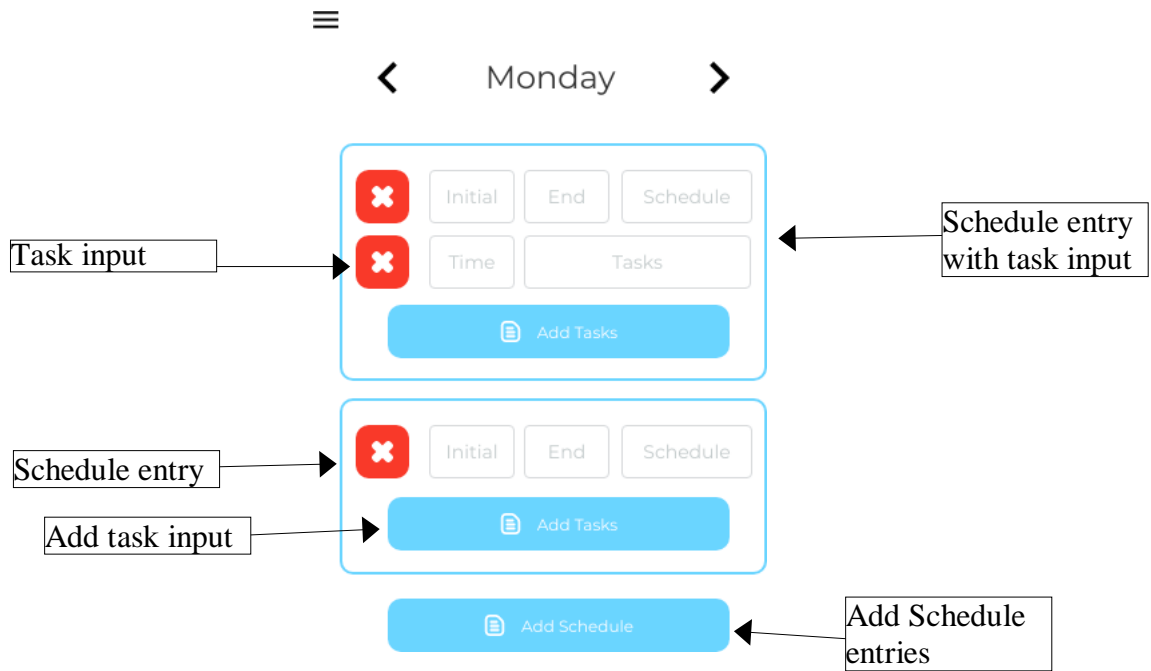
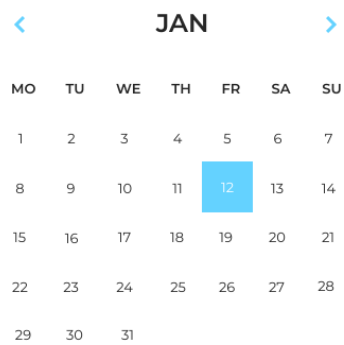
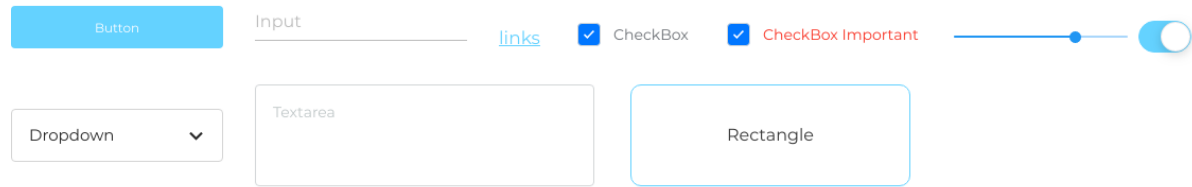


Figure 27: Mockup (Scheduler's CRUD)

5.4 Style Guide

Trackr Style Guide

Buttons, inputs, links...



Icons



Font

Monsterrat
Monsterrat

Color




Figure 28: Style Guide

6 User's Manual

6.1 Login

This page is where the user logs in. Apart from logging in, we are given the option to register as a new user.




[Create an account?](#)

Figure 29: User's Manual (Login)

6.2 Register

In this page, the user can register as a new user by assigning them their first name, last name, email and password.



[Already have an account?](#)

Figure 30: User's Manual (Registry)

6.3 Menu

Every page has a button, which opens a menu where the users can browse the pages.



*Figure 31: User's Manual
(Menu Button)*

On the menu, it shows the user's first and last name, an emoticon which shows their emotion, the pages button and a button to log out.

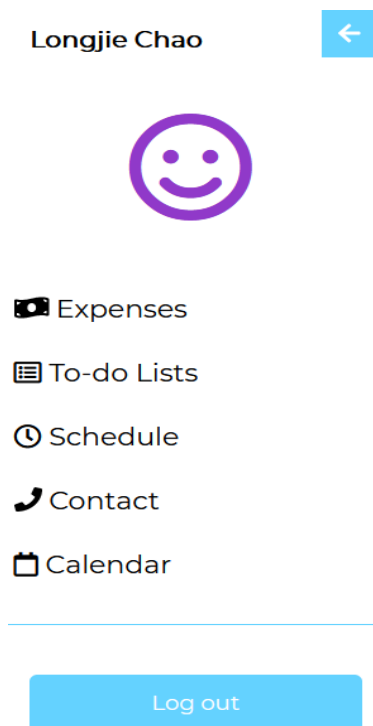
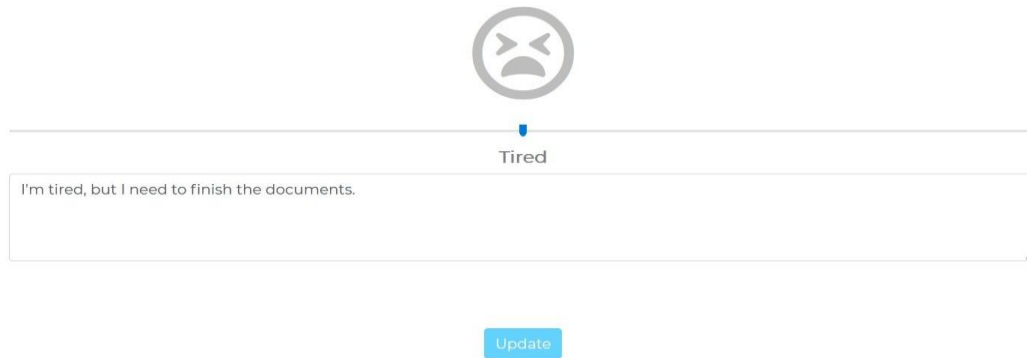


Figure 32: User's Manual (Menu)

6.4 Mood Tracker

When the user clicks on the emoticon, it will lead them to the mood page. This page has a slider to set their mood and a box where they can express how they feel or write how the user's day is.



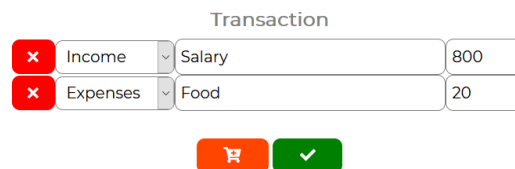
The Mood Tracker interface features a sad face emoticon at the top. Below it is a horizontal slider with a blue dot indicating the current mood level. The word "Tired" is centered below the slider. Underneath is a large text input box containing the text "I'm tired, but I need to finish the documents." At the bottom right of the input box is a small "ai" icon. Below the input box is a blue "Update" button.

Figure 33: User's Manual (Mood Tracker)

6.5 Financial Tracker

In this page, the user can set their income and expenses by assigning a name and quantity.

To add a new transaction, the user needs to click the add button first and when they finish setting the data, they need to click on the submit button.



The Financial Tracker interface shows a table with two rows of transactions. Each row has a red square with a white 'x' icon, a dropdown menu, a text input field, and a quantity input field. Below the table are two buttons: an orange button with a shopping cart icon and a green button with a checkmark icon.

Transaction			
x	Income	Salary	800
x	Expenses	Food	20

Figure 34: User's Manual (Financial Tracker's CRUD)

TRACKR

After submitting, the page will refresh and update the data, and show what the user added. Only the data of the current month will be shown. After the month ends the data of the previous month won't show.

Total	Expenses	Income
1020€	300€	1320€
2021-06-02 Salary	+800€	
2021-06-02 Food	+20€	

Figure 35: User's Manual (Financial Tracker)

If the user wants to see the previous month's transaction, they can click the "Old Transaction" button.

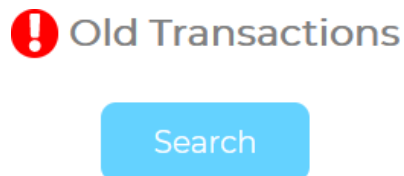


Figure 36: User's Manual (Old Transaction's Button)

6.6 Old Transactions

In this page, the user can see their previous transactions. They can change the month with the arrows.

<

June, 2021

>

02/06	Salary	+800€
02/06	Food	+20€

Figure 37: User's Manual (Old Transaction)

If there's no data, it will only show a text saying "No data to show".

No data to show

Figure 38: User's Manual (Old Transaction's Data)

6.7 To-do folder

To-do list, it's a folder to organize the tasks that the user will add.

In this page, the user can see the to-do lists folder where they can delete, edit or add a new entry. The button "x" is to delete, "pencil" to edit and "+" to add a new to do list.

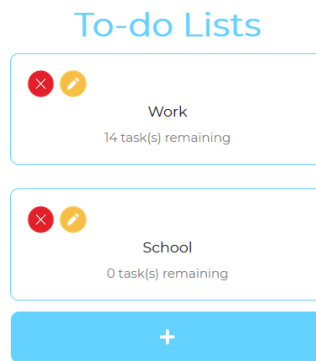


Figure 39: User's Manual (To do folder)

6.7.1 To do folder's CRUD

When the user clicks either the add button or edit button, a modal will appear where the user can add or update the name of the to-do list. And click the check button to confirm.

If the user is updating the to-do list, the folder's name will appear.



Figure 40: User's Manual (To do folder's CRUD)

6.8 Tasks and Challenges

In this page, the user can see the to-do list's task or challenge where they can delete, edit or add a new to do list. The button "x" is to delete, "pencil" to edit and "+" to add a new to do list.

Each task will show the name and a button to check as done or not, and if the user will do a task with someone, that someone's information will appear.

TRACKR

Each challenge will just show its name.

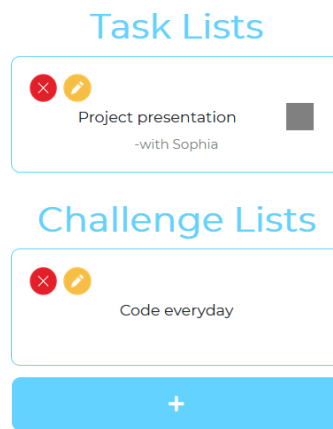


Figure 41: User's Manual (Tasks and Challenges)

6.8.1 Task and Challenges CRUD

When updating or adding, the user can choose if they want a task or a challenge.

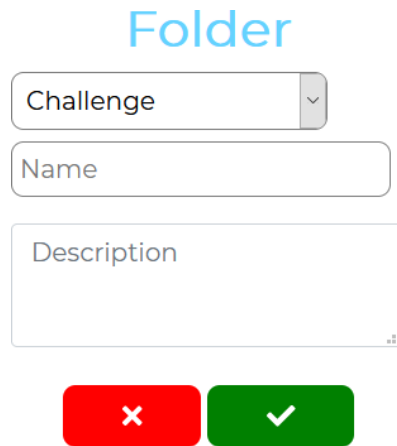
Task: The user needs to put a name, a date, a time and a description. Also they can add contacts, by clicking “Add Contact” if they’re going to do the task with someone. And if the user checks on the check mark, it will be considered as an important task.

The screenshot shows the "Folder" screen of the TRACKR app. It features a dropdown menu set to "Task" with a grey square icon. Below this is a "Name" text input field. A date and time picker is shown with "dd/mm/a" and "--:--" respectively. There is a "Contact" dropdown menu with a red 'X' icon next to it. Below the contact dropdown is a blue button with a person icon and the text "Add Contact". A large "Description" text area is at the bottom. At the very bottom of the screen are two buttons: a red one with a white 'X' and a green one with a white checkmark.

Figure 42: User's Manual (Task's CRUD)

TRACKR

Challenge: The user just needs to add a name and description. Unlike the task, the challenge doesn't need to set a date, because everyday the users will do the challenge. It's a daily challenge.



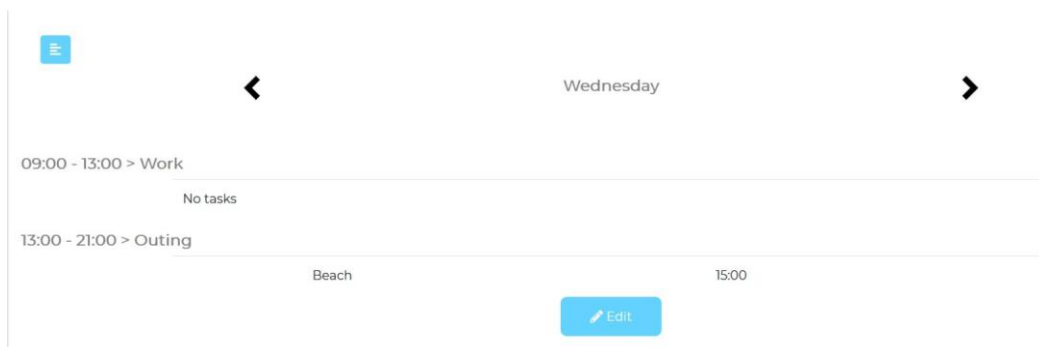
The screenshot shows a form titled "Folder" in blue. It contains three input fields: a dropdown menu labeled "Challenge", a text field labeled "Name", and a larger text area labeled "Description". Below the form are two buttons: a red button with a white "X" icon and a green button with a white checkmark icon.

Figure 43: User's Manual (Challenges' CRUD)

6.9 Scheduler

In this page the user will see what they will be going to do everyday.

The user can change which day of the week they want to see. To edit they just need to click on edit.



The screenshot shows a scheduler interface. At the top, there is a blue square icon, a left arrow, the text "Wednesday", and a right arrow. Below this, there are two time slots: "09:00 - 13:00 > Work" and "13:00 - 21:00 > Outing". Under the "Work" slot, it says "No tasks". Under the "Outing" slot, there is a task labeled "Beach" with a time range from 13:00 to 15:00. At the bottom right, there is a blue button with a pencil icon and the text "Edit".

Figure 44: User's Manual (Scheduler)

6.9.1 Scheduler's CRUD

To edit the schedule, the users will see the tags. The first thing they need to do is go on what day of the week they want to make a schedule and by clicking the button "+", they can add a new schedule by setting a name, initial time and end time.

TRACKR

After that they can edit, by clicking on the pencil button or delete, by clicking on the “X” button.

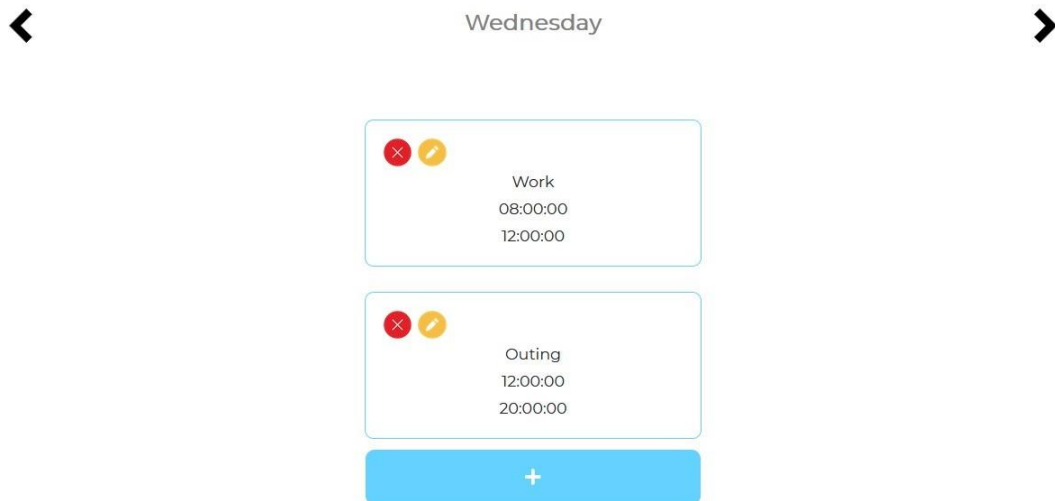


Figure 45: User's Manual (Scheduler's CRUD (1))

The screenshot shows a form titled "Add Scheduler" with a close button (X) in the top right corner. The form contains three input fields: the first is labeled "Folder Name", the second is a time field with a placeholder "--:--", and the third is another time field with a placeholder "--:--". Below the input fields is a green button with a white checkmark, indicating a confirmation or save action.

Figure 46: User's Manual (Scheduler's CRUD (2))

6.9.2 Schedule task and its CRUD

Now that the user has added a schedule, they can now add a specific task to do between both times.

To do that they first need to click on schedule's name and then the application will lead them to this page.

Here they can do the same: delete by clicking on “x”, edit and add new task by clicking on

TRACKR

pencil by clicking the button “+”. Everything they need to set is the name and the time. The application will show them the name and the time.

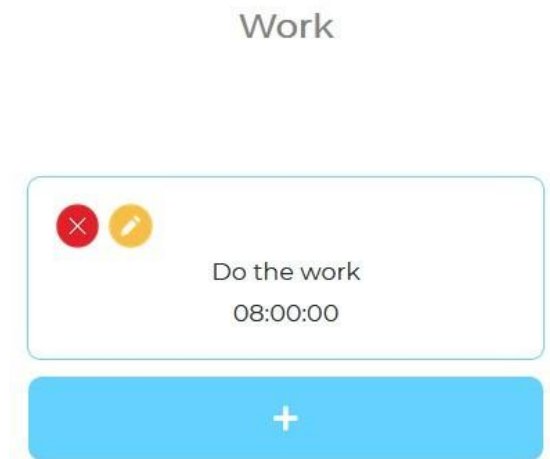


Figure 47: User's Manual (Scheduler's Task)

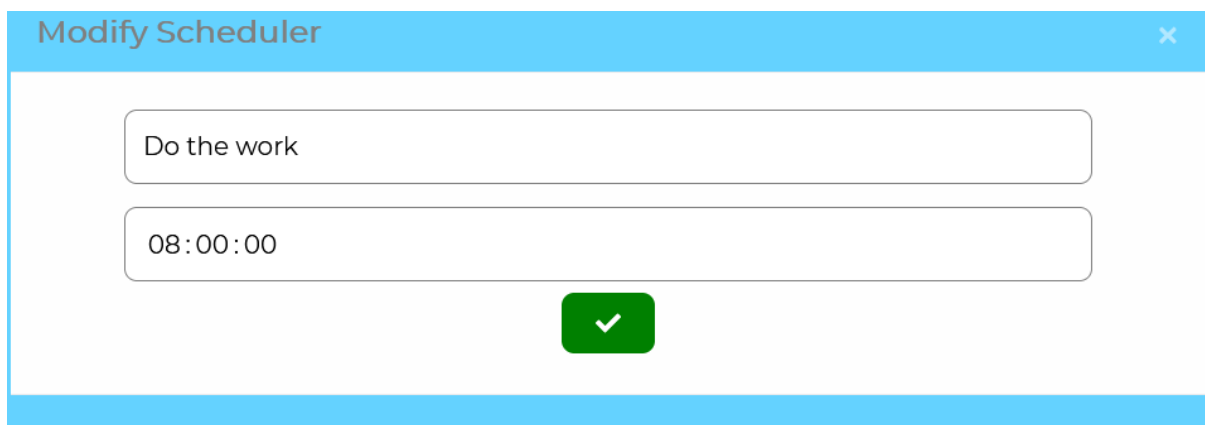


Figure 48: User's Manual (Scheduler's Task's CRUD)

6.10 Contacts and its CRUD

In this page users can delete, edit and add a contact. The contact is used to add into the to-do tasks, to see who they are going to do the tasks with or they can just add a contact to remember them.

To add a contact the user needs to click on the add contact button and it will lead them to the page.

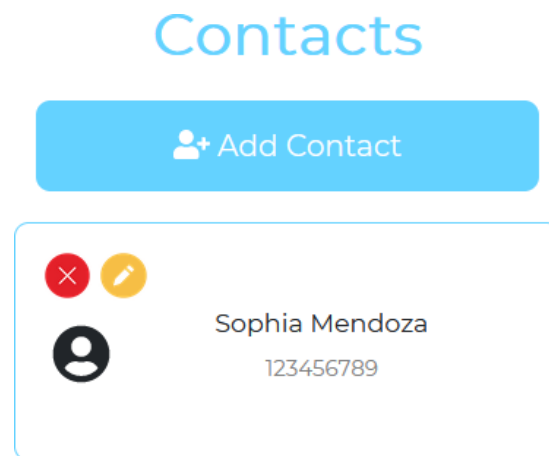


Figure 49: User's Manual (Contacts)

Now they need to set the first name, last name, telephone, mail, street, zip, city and country in order to add a new contact. And to update the contact, by clicking on the pencil button, they just need to change that data.

The screenshot displays a form titled 'Add Contacts'. It features several input fields with corresponding icons on the left: a person icon for 'First Name' and 'Last Name', a telephone icon for 'Telephone Number', an envelope icon for 'Email', and a house icon for 'Street'. Below the 'Street' field are separate fields for 'Zip' and 'City', followed by a 'Country' field. At the bottom, there are two large buttons: a red one with a white 'X' and a green one with a white checkmark.

Figure 50: User's Manual (Contacts' CRUD)

6.11 Calendar

In this page, the user can see what they did everyday for a month. They can choose a specific day and see it, the application will show them their mood and the description of the day, also will show the number of tasks they did.

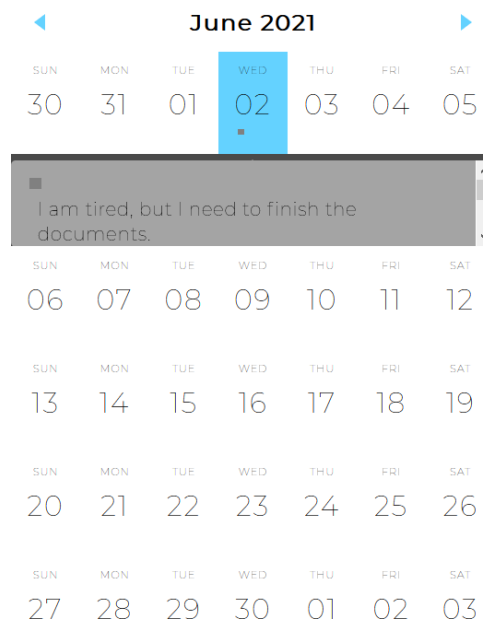


Figure 51: User's Manual (Calendar)

6.12 Main

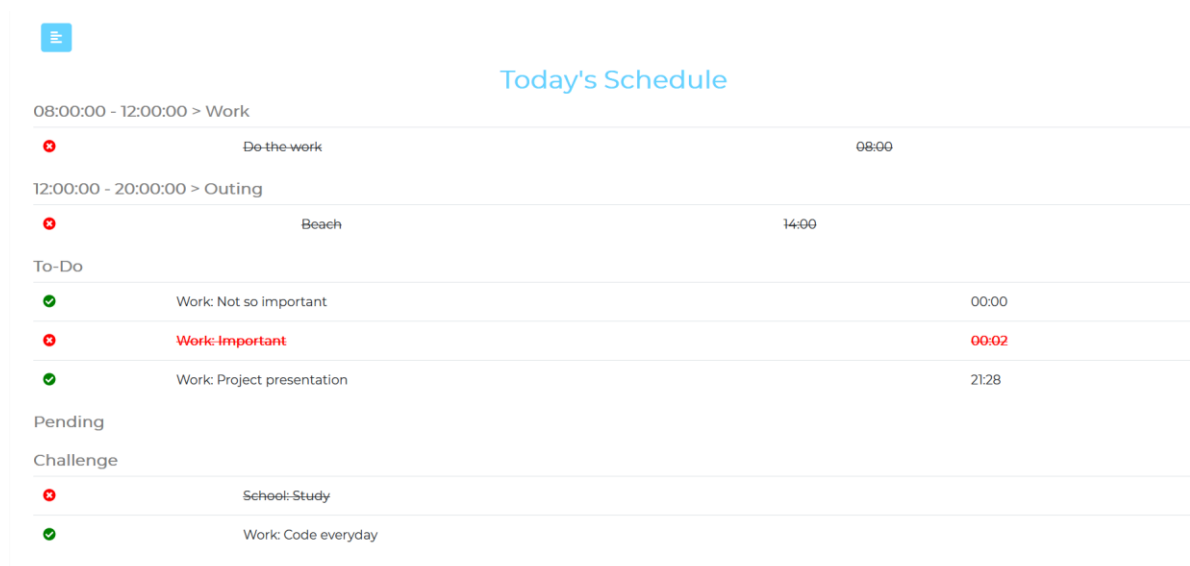
This is the main page, where the application shows what tasks they have on the current day. They can check as done by clicking on the button or uncheck.

The first row shows the schedule that the user set on every day of the week.

The to-do row shows the to-do task and shows the time.

The pending is the same as to-do row, but it shows when the to-do task doesn't have a specific date on it. It means that it will appear everyday until the user marks them as done.

And the last one is the challenge row. The application will give the challenge everyday, until the user deletes them.



Today's Schedule		
08:00:00 - 12:00:00 > Work		
✖	Do-the-work	08:00
12:00:00 - 20:00:00 > Outing		
✖	Beach	14:00
To-Do		
✔	Work: Not so important	00:00
✖	Work: Important	00:02
✔	Work: Project presentation	21:28
Pending		
Challenge		
✖	School: Study	
✔	Work: Code everyday	

Figure 52: User's Manual (Main)

7 Installation Manual

To install the application, first you will need a server with Apache, PHP and MariaDB. Set up the entire server, transfer all the files, set up the database and make the php able to connect to the database.

Or we can register a domain, find a web hosting company, transfer all the files to the web hosting, create and set up the database and connect the files to the database.

After we installed the web application in the hosting or a private server, the users can use a browser and connect to the server and use the application.

8 Bibliography

Visual Paradigm: “What is Use Case Specification?”,
<https://www.visual-paradigm.com/guide/use-case/what-is-use-case-specification/>
r/x365: “Simple Gantt Template”,
https://www.reddit.com/r/excel/comments/axedht/simple_gantt_template/
Stack Exchange Inc.: Stack Overflow, <https://stackoverflow.com/>
HashiCorp: Vagrant, <https://www.vagrantup.com/>
The Apache Software Foundation: Apache NetBeans, <https://netbeans.apache.org/>
Refsnes Data: W3Schools, <https://www.w3schools.com/>
Fonticons, Inc.: Font Awesome; <https://fontawesome.com/>
MIT: Bootstrap, <https://getbootstrap.com/>
The PHP Group: PHP, <https://www.php.net/>
MariaDB Foundation: MariaDB Server, <https://mariadb.org/>
The Apache Software Foundation: Apache, <https://httpd.apache.org/>
Software Freedom Conservancy: Git, <https://git-scm.com/>
CodePen: CodePen, <https://codepen.io/>
Lucid Software Inc.: Lucid, <https://lucid.app/>
Navasard Paul: “Event Calendar Widget”, <https://codepen.io/peanav/pen/CfFjl>