
Vanier College
Faculty of Science and Technology
System Development
420-436-VA

Deliverable 03

Brown Team

Team Leader: Peter Isaac Fishman

Sadaf Zakria

Nabil Ramadan

Hibba Qaraman

*In collaboration with **Justin Eberwein***

Next Team Leader: Nabil Ramadan

We, the Brown Team, certify that this assignment is our own work

I, Sadaf Zakria, [2151361], certify that I have contributed to this deliverable, S. Z.

I, Nabil Ramadan, [2195270], certify that I have contributed to this deliverable, N.R.

I, Hibba Qaraman, [1760010], certify that I have contributed to this deliverable, H.Q.

I, Peter Isaac Fishman, [1980427], certify that I have contributed to this deliverable, P.I.F.

Table of Contents

Executive Overview	4
Summary of the Client	5
Business Problem	6
How will we solve this:	6
Narrative Description of System	7
Appendix 1 - FlowChart	8
Appendix 2 - Use Cases UML Diagram	9
Appendix 3 - Fill out a detailed document of use cases found in Appendix 2	10
Appendix 4 - UML Class Diagram	18
Appendix 5 – Copies of Forms	19
Project Plan	20

Executive Overview

The following overview highlights the significant progress made in Deliverable 3, focusing on the development of UML diagrams, flowcharts, and use cases for the "Just B Fitness" booking web application project.

Justin Eberwein, founder of "Just B Fitness," established the business in 2016. As a small organization, Justin has successfully managed a clientele of over 100 clients without a dedicated database management system. This has led to challenges in scheduling appointments efficiently.

The primary goal of this project is to upgrade the management of "Just B Fitness" by creating a booking web application. This application will ease the booking process for Justin and his clientele, offering a more efficient and convenient scheduling experience. Additionally, the platform will serve as a foundation for the expansion of services beyond fitness training sessions, and also potentially allow future trainers to book, and add their own clients.



Summary of the Client

Justin's journey in the fitness industry began in 2016 when he founded Just B Fitness. Back then, his online presence was limited to Facebook and Instagram, until 2020 when he launched his own website. Although the website was created by Silo, who is no longer part of the team, Justin can still rely on him for any necessary website modifications. Today, Justin operates independently but maintains valuable connections with former colleagues Francesca and Ian. He conducts his client sessions at Monster Gym, renting their space for his fitness endeavors. Notably, Justin has collaborated with fitness models throughout his career, as evidenced by the glowing testimonials on his webpage. Despite his lack of computer and programming skills, Justin manages a substantial clientele of over 100 individuals, without the help of a database. His work days span from 6:30 AM to 11:30 PM, Monday through Saturday, with Sundays dedicated to scheduling sessions with clients individually. To streamline his clients' fitness journeys, he relies on the FitLog mobile app, using it to monitor their health and fitness goals. Additionally, Justin collaborates with nutritionists to offer tailored diet plans through the FitLog app, ensuring a comprehensive approach to his client's well-being.

Business Problem

The new booking web application is poised to address critical business challenges that have been plaguing Justin Eberwein's "Just B Fitness." Firstly, the absence of a dedicated database and the reliance on a manual client management process has led to inefficiencies in scheduling and information management. Justin currently manages a substantial clientele of over 100 clients, and this manual approach has proven to be both time-consuming and prone to errors. **Without a centralized data repository, client information is scattered and challenging to access, update, and utilize effectively. This reduces the overall operational efficiency of the fitness business, increases the risk of scheduling conflicts, and can result in missed appointments and frustrated clients.**

Secondly, Justin's exhaustive working hours, which extend from 6:30 AM to 11:30 PM, Monday to Saturday, present a considerable business challenge. The current booking process consumes his entire Sundays as he schedules sessions individually with clients, and currently uses a third-party app "FitLog" ¹ to provide dietary recommendations to his clients. This not only takes a toll on his personal well-being but also limits his capacity to focus on other vital aspects of growing and managing his fitness business. It is unsustainable in the long term, and Justin's dedication to his clients should be channeled into more strategic and growth-oriented activities.

How will we solve this:

The new booking web application is expected to alleviate the strain on Justin's schedule, allowing him to optimize his time and energy for broader business development while ensuring a more efficient and client-friendly booking process.

The following application will be used by both the admin (Justin) and his clientele. The app will allow clients to create an account, create/cancel a booking based on the admin's (Justin) availability, and view their booking history. The admin Justin will be able to modify his availability, view all his previous and current bookings, cancel bookings, and view all client information.

¹ FitLog is a mobile app for managing coaching that lets businesses make personalized workout plans, meal schedules, supplement recommendations and other things.

Narrative Description of System

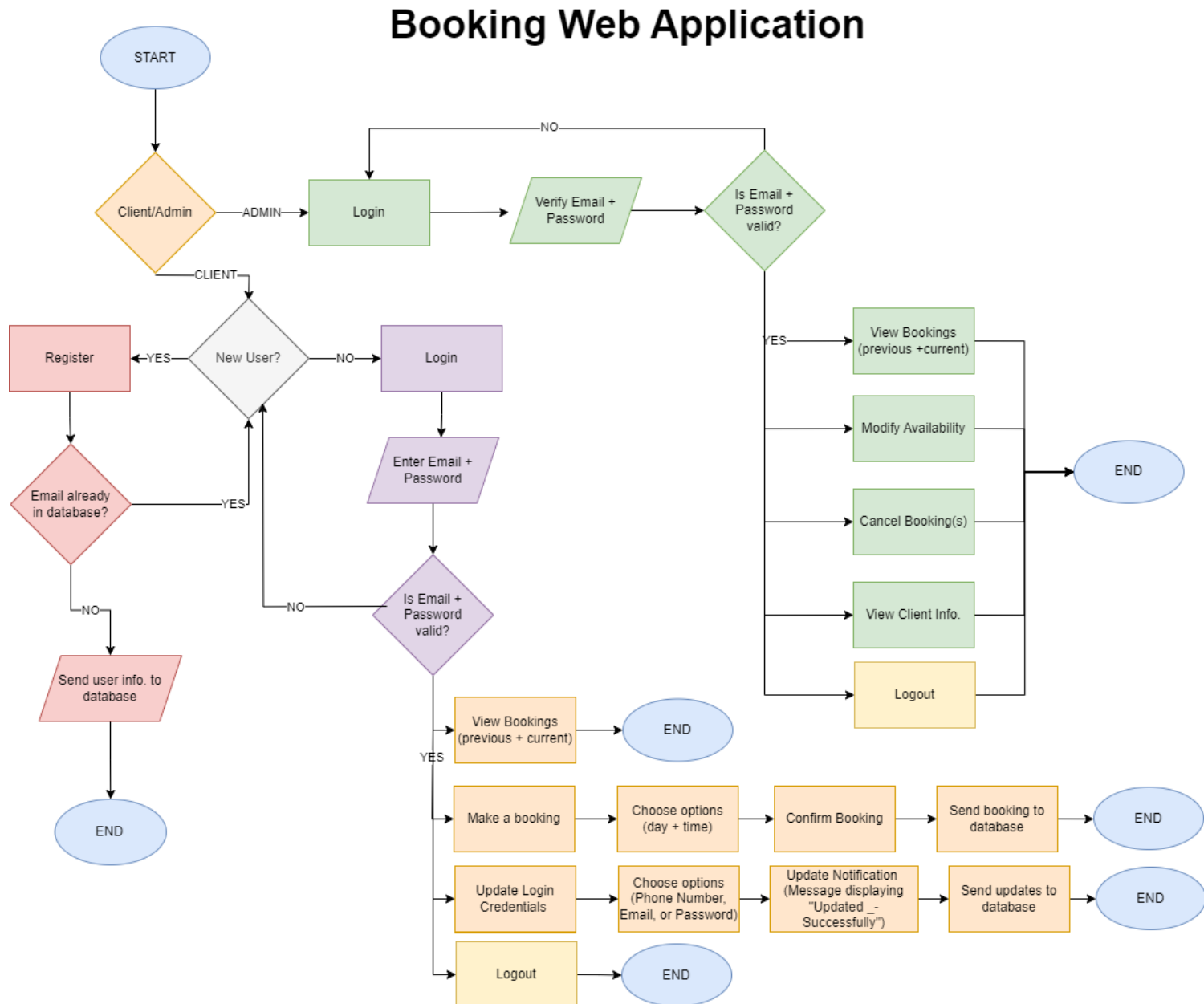
The fitness and health personal coach and trainer booking web application is designed to simplify the booking process for both clients and admins who are trainers.

When a client wants to book an appointment, they must first create an account by registering with an email and password and log in. Once logged in, they can view their bookings, schedule new appointments according to the trainer's availabilities, and even cancel existing appointments if necessary. The process for changing login credentials is also straightforward, so clients can update their information as needed.

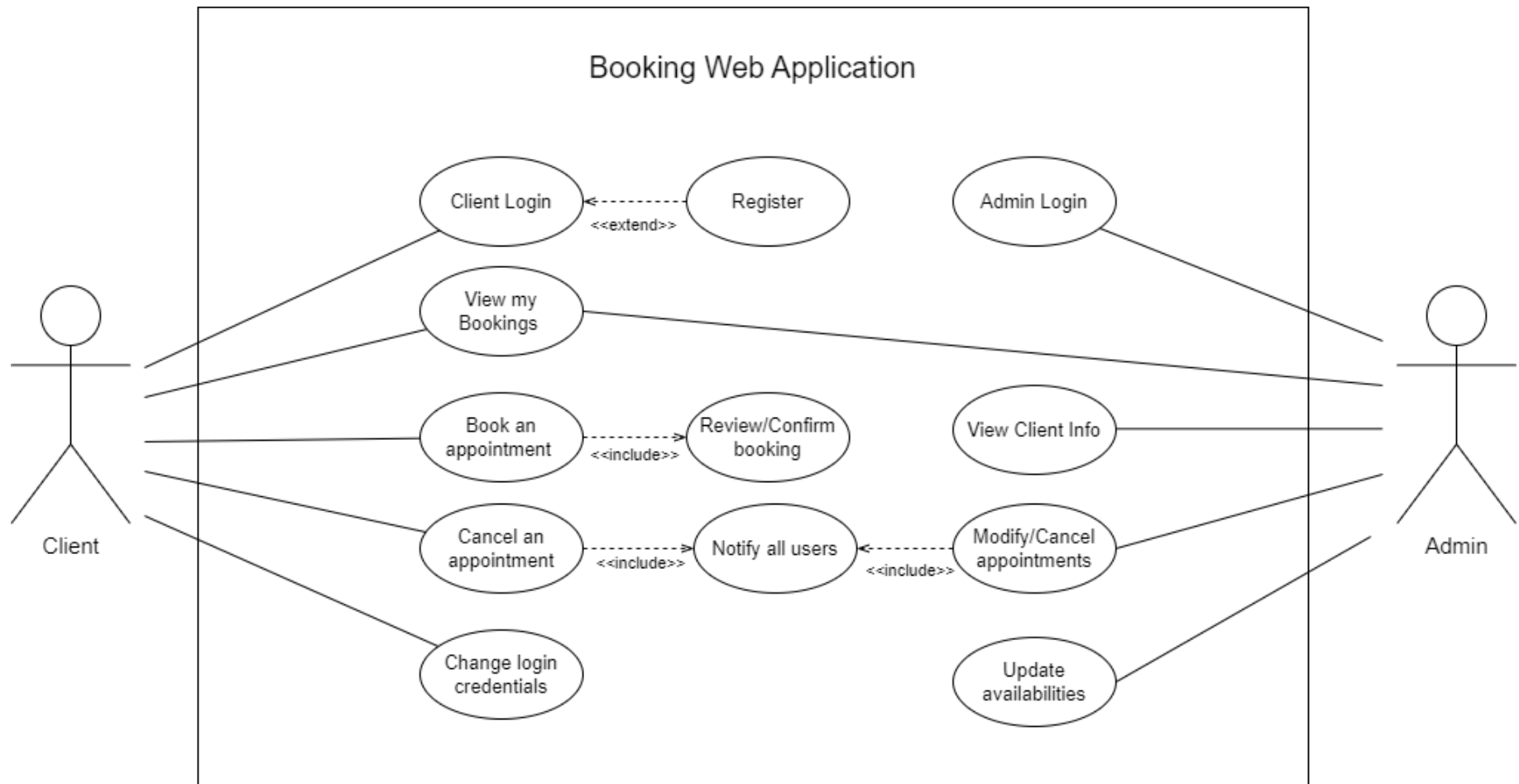
On the other hand, from the admin's side, an overview of all the bookings made by clients is displayed on the dashboard. Admins can even modify or cancel appointments on behalf of clients if necessary. They also can view detailed client information, namely contact details such as email and phone number. Trainers can also easily update their availability in the booking web application.

Overall, the system has been designed with simplicity and accessibility in mind. Whether clients or admins, both parties can navigate the booking web application with ease, reducing the risk of confusion. Additionally, this allows the owner to store client data securely while client information remains private and protected.

Appendix 1 - FlowChart



Appendix 2 - Use Cases UML Diagram



Appendix 3 - Fill out a detailed document of use cases found in Appendix 2

Use Case ID:	UC-Register		
Use Case Name:	Register		
Created By:	Team Brown	Last Updated By:	Nabil Ramadan
Date Created:	2023/09/25	Last Revision Day:	2023/09/27
Actors:	Client (Primary)		
Description:	Client registers with their information.		
Trigger:	Client wants to create an account.		
Preconditions:	Client has an email address that is not registered in the database.		
Postconditions:	Client's account information is recorded in the database.		
Normal Flow:	1- client opens the website 2- client chooses the register page 3- client enters their information 4- client is directed to the dashboard		
Alternative Flow:	In step 3. Check if the email address was not registered in the database.		
Exceptions:	In step 3. If the email address was registered in the database, an error message will be displayed that the email address is already associated with an existing account		
Includes:	None		
Frequency of Use:	On-demand		
Special Requirements:	Not applicable.		
Assumptions:	Client has not registered with their email before.		
Notes and Issues:	No		

Use Case ID:	UC-Login		
Use Case Name:	Login		
Created By:	Team Brown	Last Updated By:	Nabil Ramadan
Date Created:	2023/09/25	Last Revision Day:	2023/09/27
Actors:	Client (Primary)		
Description:	Client login with their information.		
Trigger:	Client wants to enter their account.		
Preconditions:	Client has an email address registered in the database.		
Postconditions:	Successful login.		

Normal Flow:	1- client opens the website 2- client chooses the login page 3- client enters their email address and password 4- client is directed to the dashboard
Alternative Flow:	In step 3. If the email address was not registered in the database, the user must create an account on the New User page.
Exceptions:	In step 3. If the email address was registered in the database but the password was wrong an error message will be displayed.
Includes:	UC-Register
Frequency of Use:	On-demand
Special Requirements:	Not applicable.
Assumptions:	Client did not forget their password.
Notes and Issues:	No

Use Case ID:	UC-Booking		
Use Case Name:	Booking		
Created By:	Team Brown	Last Updated By:	Nabil Ramadan
Date Created:	2023/09/25	Last Revision Day:	2023/09/27
Actors:	Client (Primary)		
Description:	Client book an appointment.		
Trigger:	Client visits the Booking page.		
Preconditions:	Client has logged in.		
Postconditions:	Booking is referred to the review/confirm page then saved in the database.		
Normal Flow:	1- client logs in to the website 2- client chooses to book an appointment and is redirected to the Booking page 3- client selects the day and the time. 4- client reviews and confirms their booking.		
Alternative Flow:	In step 1. Clients cannot log in.		
Exceptions:	In step 3. if the client chooses an unavailable day or time.		
Includes:	UC-login		
Frequency of Use:	On-demand		
Special Requirements:	Not applicable.		
Assumptions:	Client has logged in.		
Notes and Issues:	No		

Use Case ID:	UC-Review		
Use Case Name:	Review/Confirm booking		
Created By:	Team Brown	Last Updated By:	Nabil Ramadan
Date Created:	2023/09/27	Last Revision Day:	2023/09/27

Actors:	Client (Primary)
Description:	Client reviews the booking information.
Trigger:	Client tries to book an appointment.
Preconditions:	Client has logged in and selected the information for booking.
Postconditions:	Booking is confirmed and saved in the database.
Normal Flow:	1- client logins in to the website 2- client chooses to book an appointment and is redirected to the Booking page 3- client selects the day and the time. 4- client reviews and confirms their booking.
Alternative Flow:	In step 1. Clients cannot log in.
Exceptions:	In step 3. if the client chooses an unavailable day or time.
Includes:	UC-Booking
Frequency of Use:	On-demand
Special Requirements:	Not applicable.
Assumptions:	Client has logged in.
Notes and Issues:	No

Use Case ID:	UC-ViewBooking		
Use Case Name:	View Booking		
Created By:	Team Brown	Last Updated By:	Nabil Ramadan
Date Created:	2023/09/25	Last Revision Day:	2023/09/27
Actors:	Client (Primary)		
Description:	Client view their booking.		
Trigger:	Client logins and have access to the dashboard.		
Preconditions:	Client has booked before.		
Postconditions:	Client can see the date and the time of their booking.		
Normal Flow:	1- client logins to the website 2- client accesses their dashboard 3- client views their future and past bookings.		
Alternative Flow:	In step 1. Clients cannot log in.		
Exceptions:	In step 2. If the client has not booked yet, there would be nothing.		
Includes:	UC-Booking		
Frequency of Use:	On-demand		
Special Requirements:	Not applicable.		
Assumptions:	Client has booked before.		
Notes and Issues:	No		

Use Case ID:	UC-Canceling		
Use Case Name:	Cancel an appointment		
Created By:	Team Brown	Last Updated By:	Nabil Ramadan
Date Created:	2023/09/26	Last Revision Day:	2023/09/27
Actors:	Client (Primary)		
Description:	Client cancel their booking.		
Trigger:	Client logins and have access to the dashboard.		
Preconditions:	Client has booked before.		
Postconditions:	Client can cancel their booking.		
Normal Flow:	1- client logins in the website 2- client views their bookings in the dashboard 3- client selects a future booking 4- client cancels the appointment.		
Alternative Flow:	In step 1. Client cannot log in.		
Exceptions:	In step 4. If the client has no future bookings, they cannot cancel appointments.		
Includes:	UC-Booking		
Frequency of Use:	On-demand		
Special Requirements:	Not applicable.		
Assumptions:	Client has booked before.		
Notes and Issues:	No		

Use Case ID:	UC-NotifyAdmin		
Use Case Name:	Notify all users		
Created By:	Team Brown	Last Updated By:	Nabil Ramadan
Date Created:	2023/09/27	Last Revision Day:	2023/09/27
Actors:	Client (Primary)		
Description:	After client cancel their booking a message will notify admin		
Trigger:	Client cancel their booking.		
Preconditions:	Client has booked before.		
Postconditions:	Admin knows that the client has canceled their booking.		
Normal Flow:	1- client logins in the website 2- client views their bookings in the dashboard 3- client selects a future booking 4- client cancels the appointment selected. 5- A message will be sent to the Admin about the cancellation.		
Alternative Flow:	none.		
Exceptions:	In step 4. If the client has no future bookings, they cannot cancel appointments.		
Includes:	UC-Canceling		
Frequency of Use:	On-demand		
Special Requirements:	Not applicable.		

Assumptions:	Client has booked before.
Notes and Issues:	No

Use Case ID:	UC-ChangeInfo		
Use Case Name:	Change login credentials		
Created By:	Team Brown	Last Updated By:	Nabil Ramadan
Date Created:	2023/09/26	Last Revision Day:	2023/09/27
Actors:	Client (Primary)		
Description:	Client can change their login credentials.		
Trigger:	Client updates their email, phone number, or password.		
Preconditions:	Client is registered to the database and is currently logged in.		
Postconditions:	Client can log in with their new credentials.		
Normal Flow:	1- client logs in to the website 2- client chooses to change login credentials 3- client changes their email, phone number, or password 4- client successfully saves the changes		
Alternative Flow:	None		
Exceptions:	In step 1. Clients cannot log in.		
Includes:	UC-register		
Frequency of Use:	On-demand		
Special Requirements:	Not applicable.		
Assumptions:	Client knows their old credentials.		
Notes and Issues:	No		

Use Case ID:	UA-VCI		
Use Case Name:	View Client Info		
Created By:	Team Brown	Last Updated By:	Peter Isaac Fishman
Date Created:	2023/09/27	Last Revision Day:	2023/09/27
Actors:	Admin		
Description:	Admin can view any client info.		
Trigger:	Client has an account in the database.		
Preconditions:	Client has made an account.		
Postconditions:	Client info is relayed to the admin.		
Normal Flow:	1- admin enters login 2- admin selects View Client Info page 3- admin selects a client 4- admin views info about client		
Alternative Flow:	User does not have admin privileges		
Exceptions:	Client is not in database		
Includes:	UC-Register		

Frequency of Use:	On-demand
Special Requirements:	User is listed as an admin.
Assumptions:	User has an account as an admin and wants to view a client.
Notes and Issues:	No

Use Case ID:	UA-Login		
Use Case Name:	Admin Login		
Created By:	Team Brown	Last Updated By:	Peter Isaac Fishman
Date Created:	2023/09/27	Last Revision Day:	2023/09/27
Actors:	Admin/Client		
Description:	Admin can log in to admin account.		
Trigger:	Admin wants to log in to the main admin account.		
Preconditions:	Admin has the admin login and password.		
Postconditions:	Admin login and password for the account are in the database and listed as an admin.		
Normal Flow:	1- admin opens the website 2- admin goes to login page 3- admin enters the information 4- admin is sent to the main page		
Alternative Flow:	Check if the email is listed as an admin.		
Exceptions:	If the login info is wrong they will be sent back to login.		
Includes:	Admin privileges		
Frequency of Use:	On-demand		
Special Requirements:	Account is listed as an admin.		
Assumptions:	User has an account and is an admin		
Notes and Issues:	No		

Use Case ID:	UA-MCA		
Use Case Name:	Modify/Cancel Appointments		
Created By:	Team Brown	Last Updated By:	Peter Isaac Fishman
Date Created:	2023/09/27	Last Revision Day:	2023/09/27
Actors:	Admin/Client		
Description:	Admin can view all appointments and chose to cancel.		
Trigger:	An appointment was created		
Preconditions:	Client has booked an appointment.		
Postconditions:	Admin can view or cancel the appointment.		
Normal Flow:	1-Admin logs in 2-Admin select Modify/Cancel Appointments 3-Admin selects an appointment		
Alternative Flow:	User does not have admin privileges.		

Exceptions:	Client has yet to book an appointment
Includes:	UA-Notify
Frequency of Use:	On Demand
Special Requirements:	User is an admin.
Assumptions:	Client has booked an appointment.
Notes and Issues:	No

Use Case ID:	UA-Availabilities		
Use Case Name:	Admin Availabilities		
Created By:	Team Brown	Last Updated By:	Peter Isaac Fishman
Date Created:	2023/09/27	Last Revision Day:	2023/09/27
Actors:	Admin		
Description:	Admin can enter availabilities for appointments.		
Trigger:	Admin enters availabilities page.		
Preconditions:	The selected day is open to set availabilities.		
Postconditions:	Admin can View availabilities		
Normal Flow:	1- admin logs in 2- admin selects Availabilities page 3- admin updates time slots		
Alternative Flow:	User does not have admin privileges.		
Exceptions:	The selected time is during closed hours.		
Includes:	None		
Frequency of Use:	On-demand		
Special Requirements:	User is an admin.		
Assumptions:	User has an account and is an admin.		
Notes and Issues:	None		

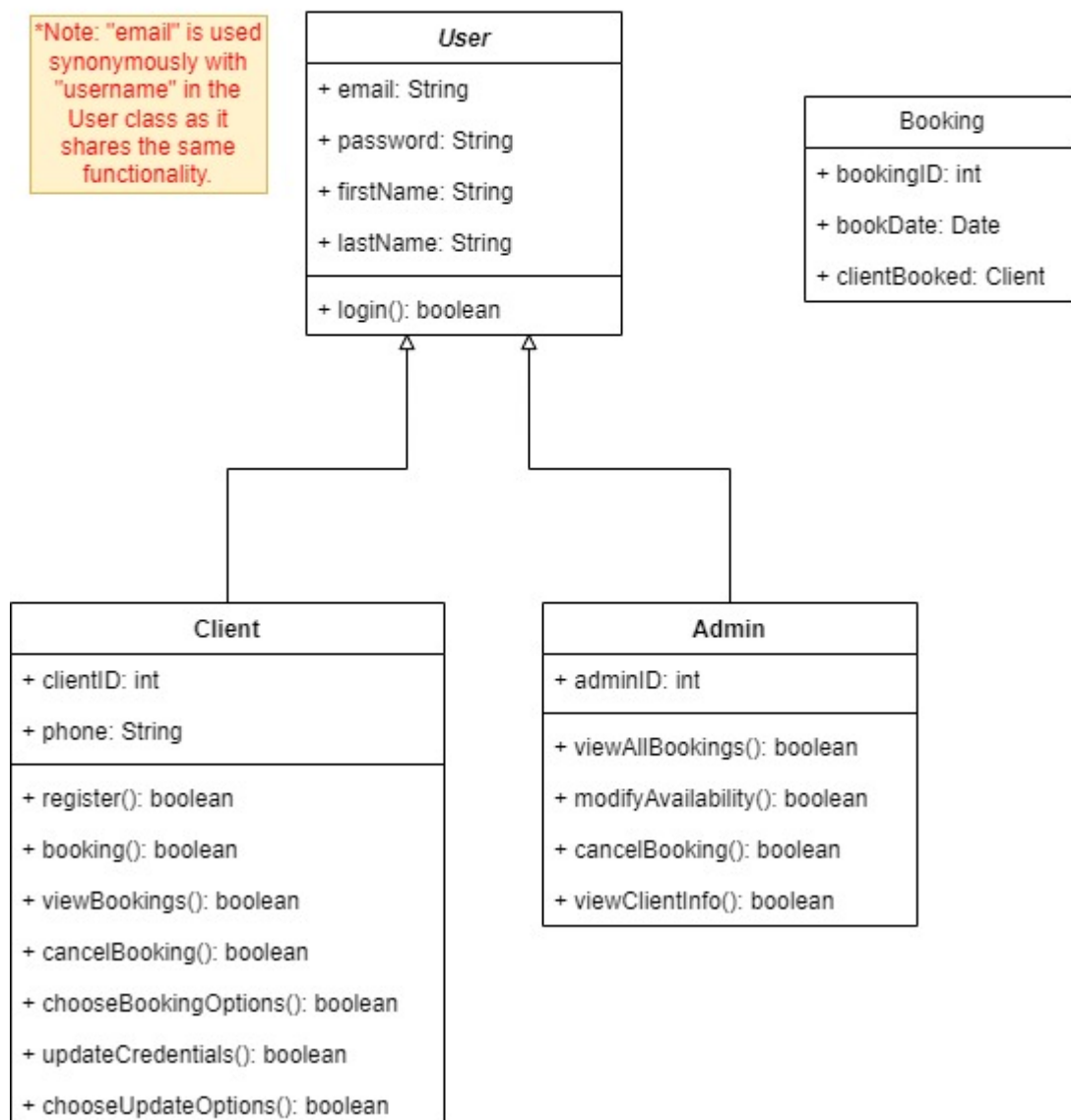
Use Case ID:	UA-ViewBooking		
Use Case Name:	View Booking		
Created By:	Team Brown	Last Updated By:	Peter Isaac Fishman
Date Created:	2023/09/27	Last Revision Day:	2023/09/27
Actors:	Admin/Client		
Description:	Admin can view clients bookings.		
Trigger:	Admin selects booking page.		
Preconditions:	Client has booked before.		
Postconditions:	Admin can see the date and time of the booking.		
Normal Flow:	1- admin logs in 2- admin selects clients bookings page in admin dashboard 3- admin can view bookings of all clients		

Alternative Flow:	User does not have admin privileges.
Exceptions:	The client has not booked anything.
Includes:	None
Frequency of Use:	On-demand
Special Requirements:	User is an admin.
Assumptions:	User has an admin account, and client has booked.
Notes and Issues:	None

Use Case ID:	Ua-NotifyClient		
Use Case Name:	Notify all users		
Created By:	Team Brown	Last Updated By:	Peter Isaac Fishman
Date Created:	2023/09/27	Last Revision Day:	2023/09/27
Actors:	Admin / Client (Primary)		
Description:	After admin cancel their booking a message will notify the client		
Trigger:	Admin can cancel the client booking.		
Preconditions:	Client has booked before.		
Postconditions:	Client knows that the admin has canceled their booking.		
Normal Flow:	1- admin logins in the website 2- admin views the client bookings in the dashboard 3- admin selects a future booking 4- admin cancels the appointment selected. 5- A message will be sent to the client about the cancellation.		
Alternative Flow:	User does not have admin privileges.		
Exceptions:	In step 4. If the client has no future bookings, the admin cannot cancel appointments.		
Includes:	UA-ViewBooking		
Frequency of Use:	On-demand		
Special Requirements:	User is an admin.		
Assumptions:	User has an admin account, and client has booked.		
Notes and Issues:	None		

Appendix 4 - UML Class Diagram

UML Class Diagram



Appendix 5 – Copies of Forms

Justin's Fitness & Health Coaching Services

Pricing:

- \$60 per 1-hour session

Schedule Time Slots (Monday to Saturday):

- 6:30 am - 7:30 am
- 7:30 am - 8:30 am
- 8:30 am - 9:30 am
- 9:30 am - 10:30 am
- 10:30 am - 11:30 am
- 11:30 am - 12:30 pm
- 12:30 pm - 1:30 pm
- 1:30 pm - 2:30 pm
- 2:30 pm - 3:30 pm
- 3:30 pm - 4:30 pm
- 4:30 pm - 5:30 pm
- 5:30 pm - 6:30 pm
- 6:30 pm - 7:30 pm
- 7:30 pm - 8:30 pm
- 8:30 pm - 9:30 pm
- 9:30 pm - 10:30 pm
- 10:30 pm - 11:30 pm

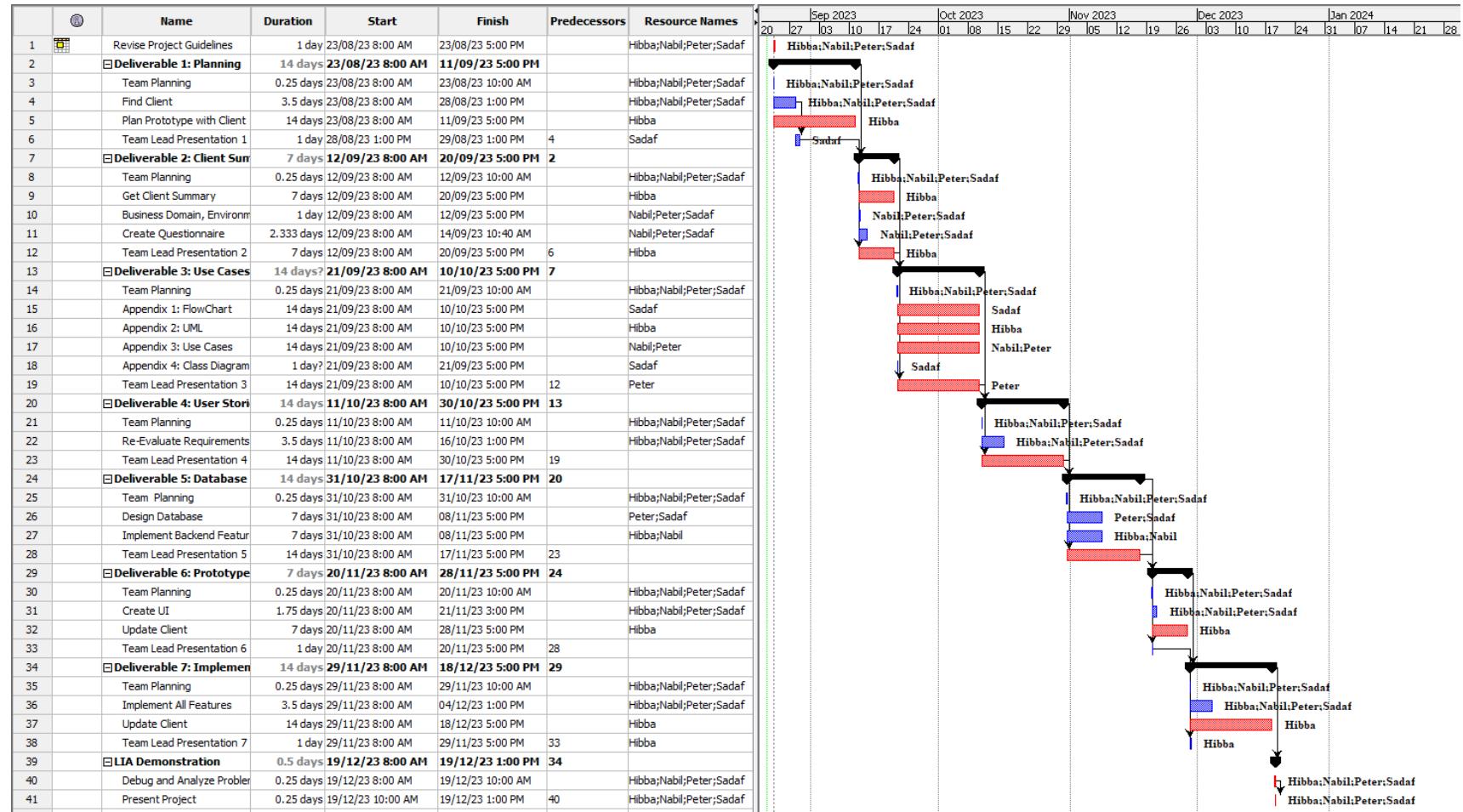
Location:

- 2101 Rte Transcanadienne, Dorval, QC, Canada, Quebec

NOTE:

While Justin's coaching sessions are usually set for one hour, they may finish earlier, allowing him to take short breaks in between appointments or in the event of cancellations.

Project Plan



[Link to WBS and Gantt Chart & PDF Version](#)