Republic of Yemen

Ministry of High Education

and Research

International University of Technology Twintech

FITFAT Gym Management System

Supervisors:

- **Dr.Hamzah Jamel**
- **Ms.Inas**

Major: Computer science

Semester: 5

Subject: Software Engineering

Done by:

Heba Al-Dobai

Firas Al-Azizy

2019-2020

INDEX

Contents

Chapter One

(Analysis)

Introduction

Purpose:

Making smart Management System for Gym business that achieved all required business.

Intended audience:

Describe which part of the SRS document is intended for which reader. Include a list of all stakeholders of the project, developers, project managers, and testers for better clarity.

Objectives:

- Eliminate the manual systems that exists in every gym in Sana'a.
- Eliminate Time consuming.
- No data loss or data redundancy.
- Accurate in keeping and retrieving information.
- UI is not complicated.
- Ease of maintenance.

Problem Statement:

The purpose of this project is to eliminate the manual system that exists in every gym in Sana'a. According to that, all the problems that generated because of the manual system will be solved in FITFAT Gym Management System project for example, time consuming as data entry which include calculations took lot of time, searching was very complex, lots of data loss and data redundancy. The project will be implemented in order to maintain effectiveness, efficiency and to be faster than the existing.

Glossary

Abbreviation	Definition
FFGMS	FitFat Gym Management System
UI	User Interface

User Requirements Definition

Scenario:

These are the actors in the scenario:

Ali (as the user of the system)

Ahmed (as the External client)

Mohammed (as the trainer)

Amjad (as the Admin)

Ahmed enters the gym to register into the system.

Ali as a preregistered Gym staff logs in to the system to and opens the trainee page to register Ali by entering his personal information like (name, age, gender, email address, phone number, weight, height, training program, name of the trainer, membership time, training time). He takes the Whole money amount from Ahmed or divide it into weekly or monthly instalments.

Ali sends Ahmed's information to Mohammed (the trainer), Mohammed will direct Ahmed through the training programs to see what is the most suitable program for Ahmed.

Ahmed's Training program will contain the specific exercises and its used equipment's in the specific training time (days, hours) ex: (Sunday, Tuesday, Thursday from 10:00am to 12:00pm), after that , Mohammed will assign Ahmed to a training group , then with this additional information to his page as a trainee, Ali will print for Ahmed the receipt of his information.

Training Groups pages will be filled by the system user, it can only contain 4 trainees and 1 trainer as a supervisor to direct them through their training program and check their improvement over time.

When a trainer Like Mohmed wants to join the Gym, Ali will open the trainers page to register Mohammed by entering his personal information like (name, age, gender, email address, phone number, Experience years, payroll), then add the trainer to the system.

If Ahmed's wanted to leave the gym, Ali will remove his membership from the gym and keeps his information if he returns

System requirements specification

Functional requirements:

Functional requirements may be expressed as functions, services or tasks or which system is required to perform. The following subsections illustrate functional requirements to be fulfilled by the proposed system.

Receptionist	Create Account Log In Add client Update current client View client status View client history Print the bill Generate reports Search the details
Admin	Add client View client status View client history Update current client Remove client Print the bill Add new trainer View trainer status View trainer history Update current trainer Delete new trainer Add new receptionist View receptionist status View receptionist history Update current receptionist Delete new receptionist Divide Time and Groups Create training program Generate reports Search for info Add machinery Update machinery Delete machinery Manage Transactions
Trainer	Choosing training program

Manage Machinery	Have a complete record of the purchased machinery in the gym along with prices as well
Manage Transactions	Have a complete record of the transactions
Bill	Display the client name Display the client training group Display the date and time of starting and ending the training Display how much he will pay and how much will remain

Non-Functional requirements:

Usability (Capability of the product to be understood & Friendly user interface)

Performance (Robustness & Quality of the Software)

Maintainability & Reliability (Flexibility in Error-handling)

Security & Safety (Data access & encapsulation)

Constraints:

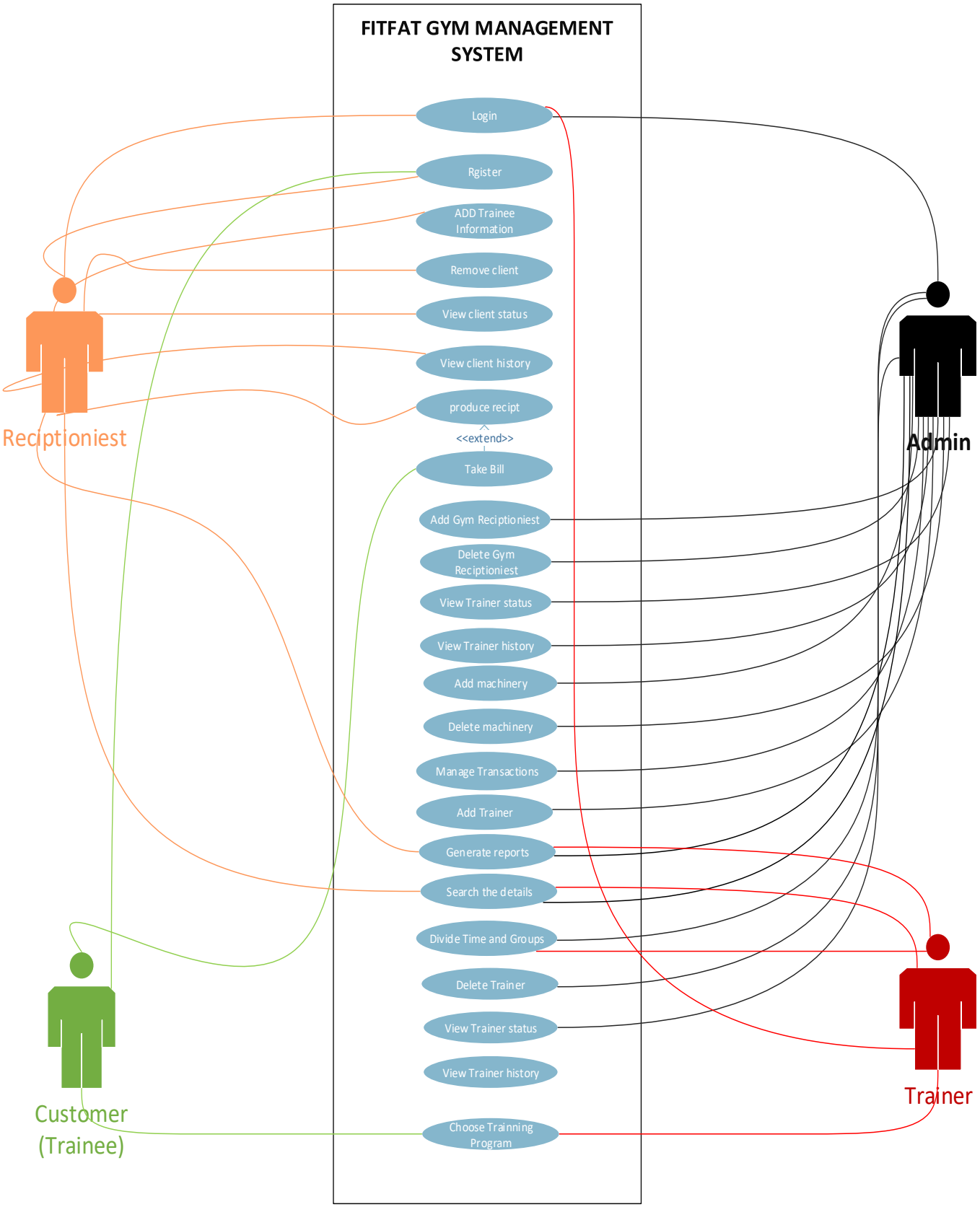
This Gym Management system will not contain other activities in any gym that are not mentioned in the scenario Like (Keeping information of Buying or selling Fitness Products), (interfering in more financial records or activities, will only record income, outcome, price of services and price of equipment's)

Chapter Two

(Design)

System models

1. Use Case Diagram:



- **Use Case Description:**
  - **a. Training Program:**

Use Cas Name	Create training program.	
Brief Description	Trainer create excise schedule for certain muscles which the trainee will follow each day.	
Trigger event	New trainee wants to enroll into training program.	
Actors	Trainer and Trainee	
Precondition	Trainee must be pre-register into the system and payed registration fees.	
Postcondition	New trainee decides which group he will join.	
Main Flow	Actor	System
	1. Create indicates desire to create training program. 2. Entering the name of trainee. 3. Choose the target muscles and the day of the training. 4. Trainer press save.	1.1 System create empty field. 2.1 Display the trainee required information. 3.1 prompt the new training program. 4.1 save the changes.
Exceptions	If Trainee name was written in wrong way pop up an error message. Not selecting muscles or machinery or leaving any fields empty the operation will not complete.	

  - **b. Dividing Time and groups:**

Use Cas Name	Dividing Time and Groups.	
Brief Description	Admin should divide groups that contain of four trainees and decide which trainer is responsible of training them. The duration for each group is 2 hours. There are at least two groups working at the same time slot.	
Trigger event	New trainee registers into the system and decides which time he wants to join at.	
Actors	Admin and Trainee.	
Precondition	Trainee must register into the system and decides which time he wants to join at. It must be four available trainees to open up the group.	
Postcondition	Trainer does not take more than one group at the same time.	
Main Flow	Actor	System
	1. Create indicates desire to create divide the groups of trainees. 2. Admin choose the name of trainer and choose which time of his group and the members. 3. Admin add another group. 4. Admin press save.	1.1 System create empty field to receive information. 2.1 System fill this information in the field 3.1 System add another field with new information. 4.1 System save information in the database.

c. Print Status:

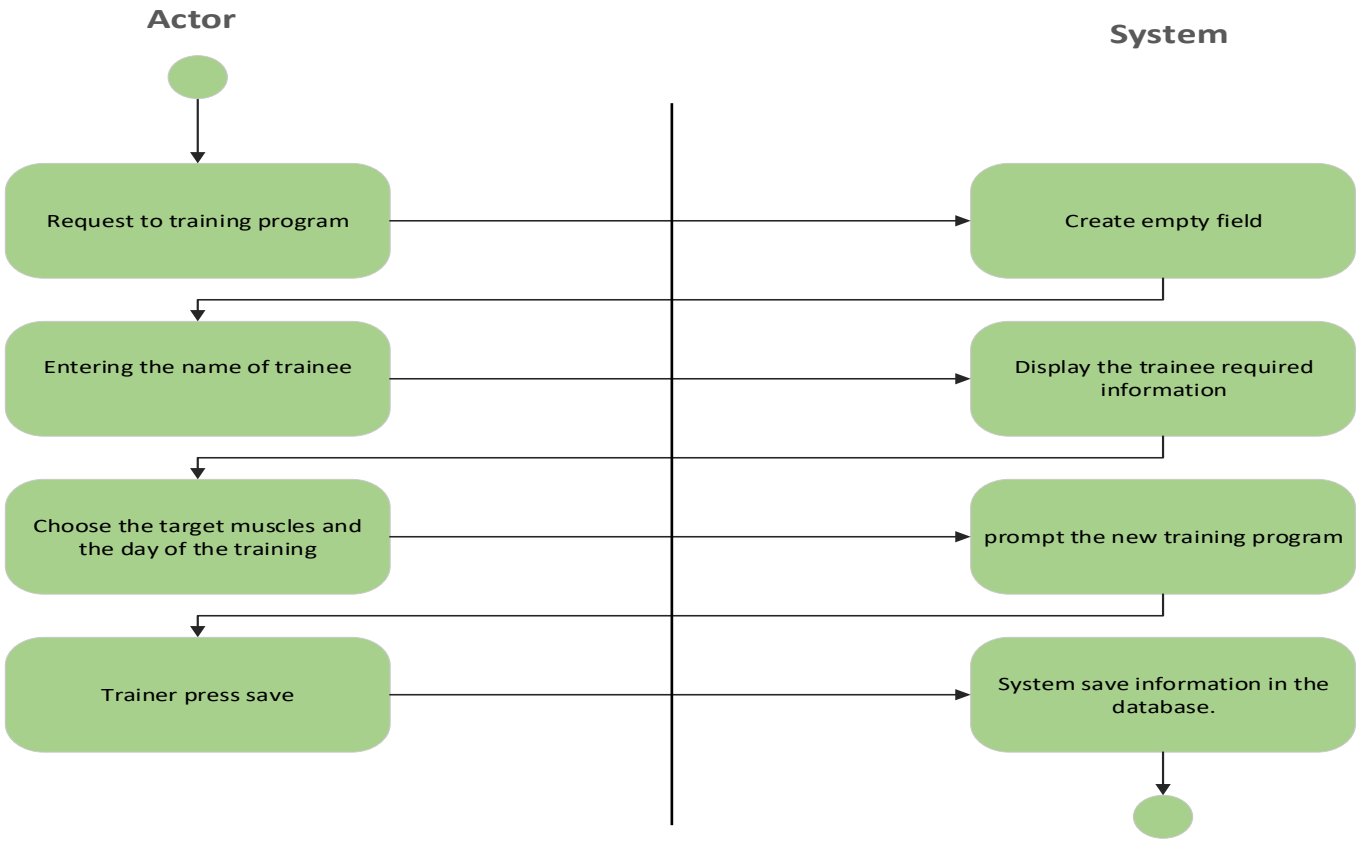
Use Cas Name	Print status.	
Brief Description	When the trainee wants to know his status like his current weight or his report.	
Trigger event	Receptionist wants to print the status of the trainee	
Actors	Trainer and Trainee	
Precondition	Each User has to have account in the system.	
Postcondition	Trainee can ask only for printing his status by his name and id and cannot ask for his mates' information.	
Main Flow	Actor	System
	1. Receptionist select a trainee 2. Receptionist choose which information the trainee wants to print	1.1 System display the trainee information. 2.1 System print information.

d. Search for Info:

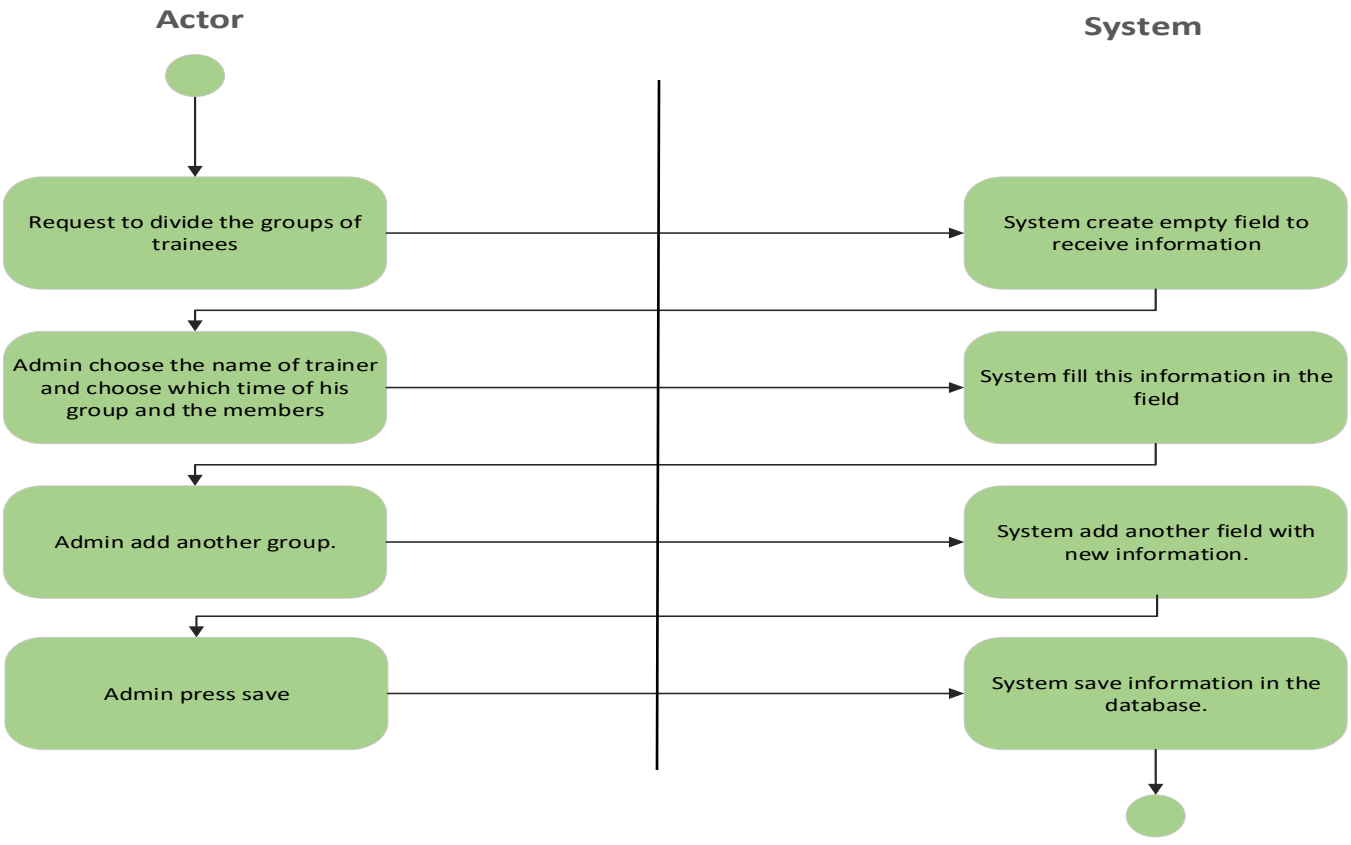
Use Cas Name	Search information.	
Brief Description	Search any data. It is not case sensitive to make the search easier.	
Trigger event	When user wants to search about any data	
Actors	Admin, Trainer, Receptionist.	
Precondition	Each User has to have account in the system. Each User has a certain domain that can search in.	
Main Flow	Actor	System
	1. User type the data that he looking for in the search box 2. User select a specific data from the list.	1.1 System displays a list of related result. 2.1 System display the selected data.
Exceptions	If the data that was search about it is not existing the system will notify the user by showing him a message.	

2. Activity Diagram:

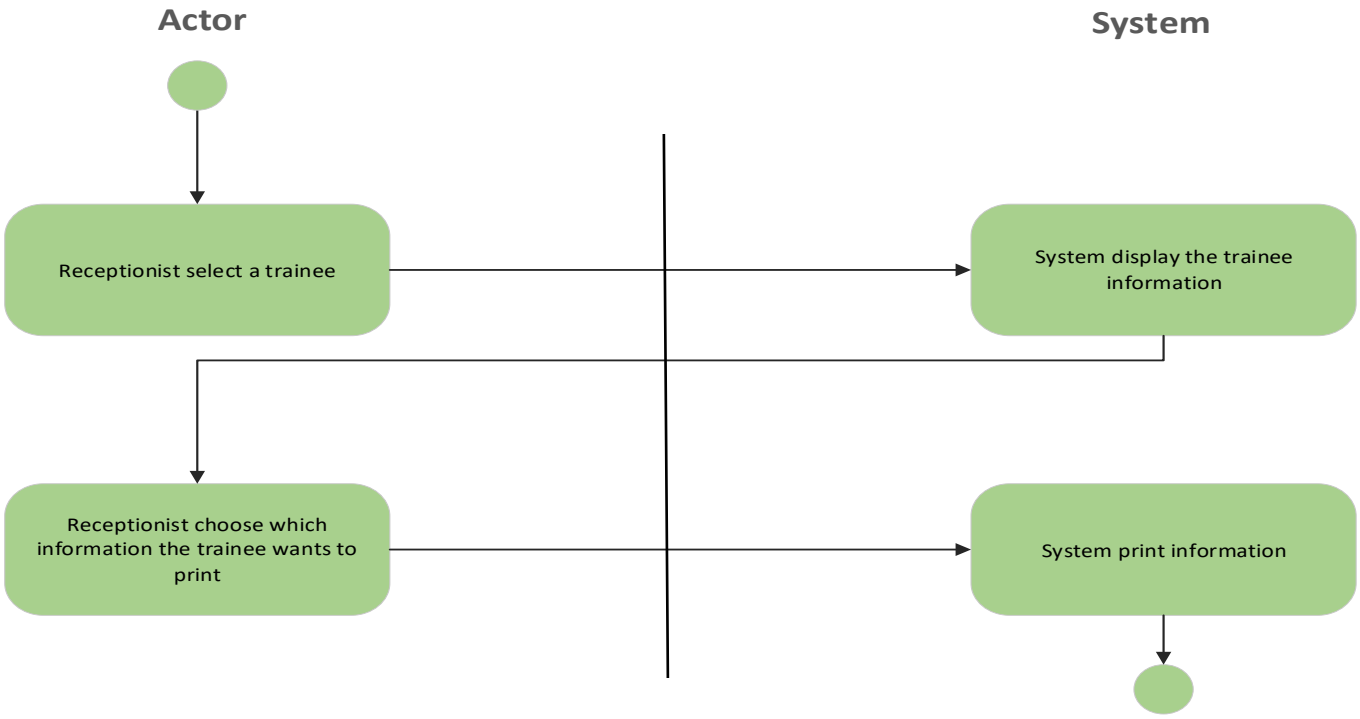
a. Training Program:



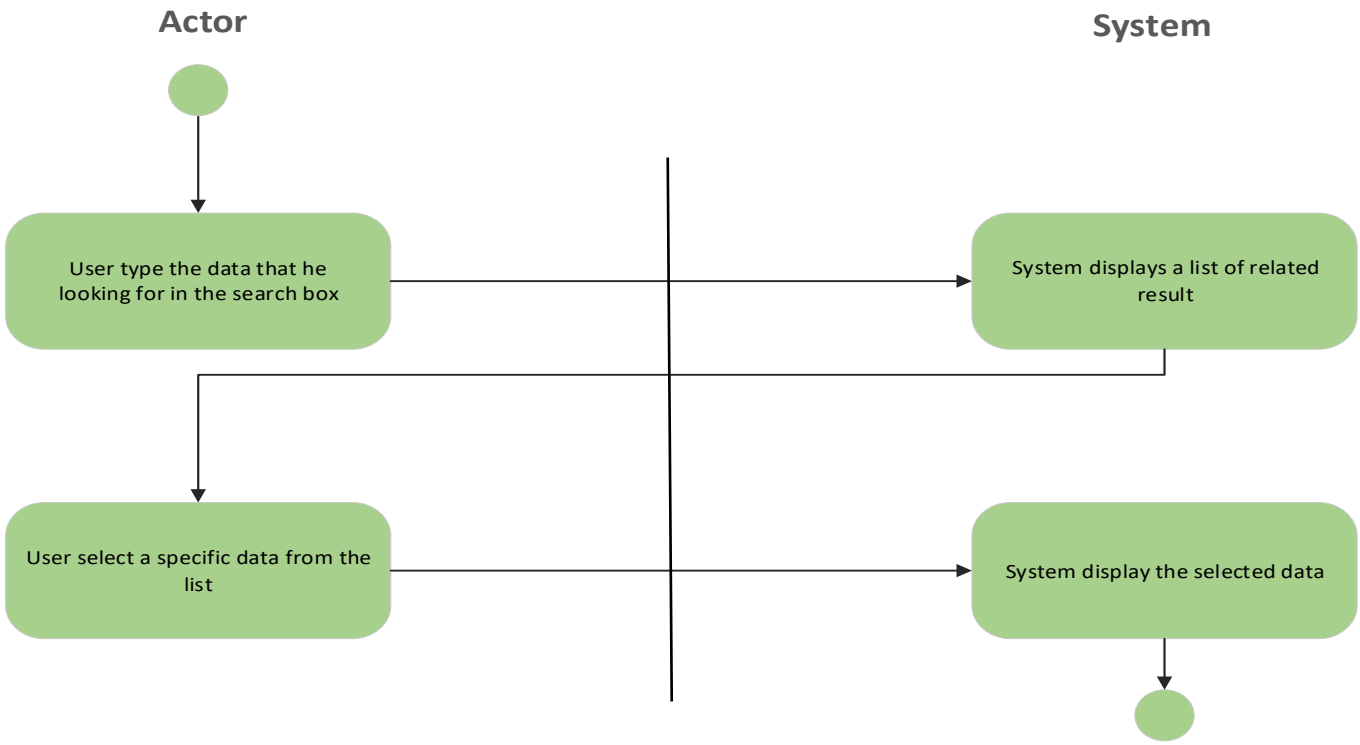
b. Dividing Time:



c. Print Status:

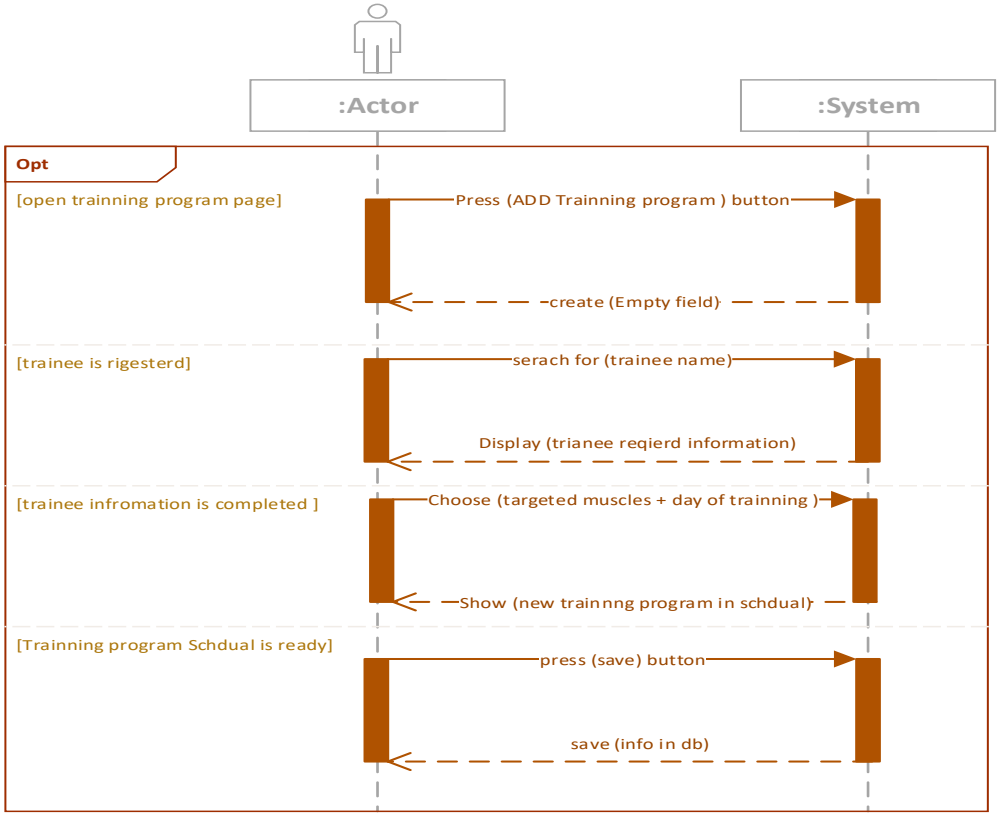


d. Search for Info:

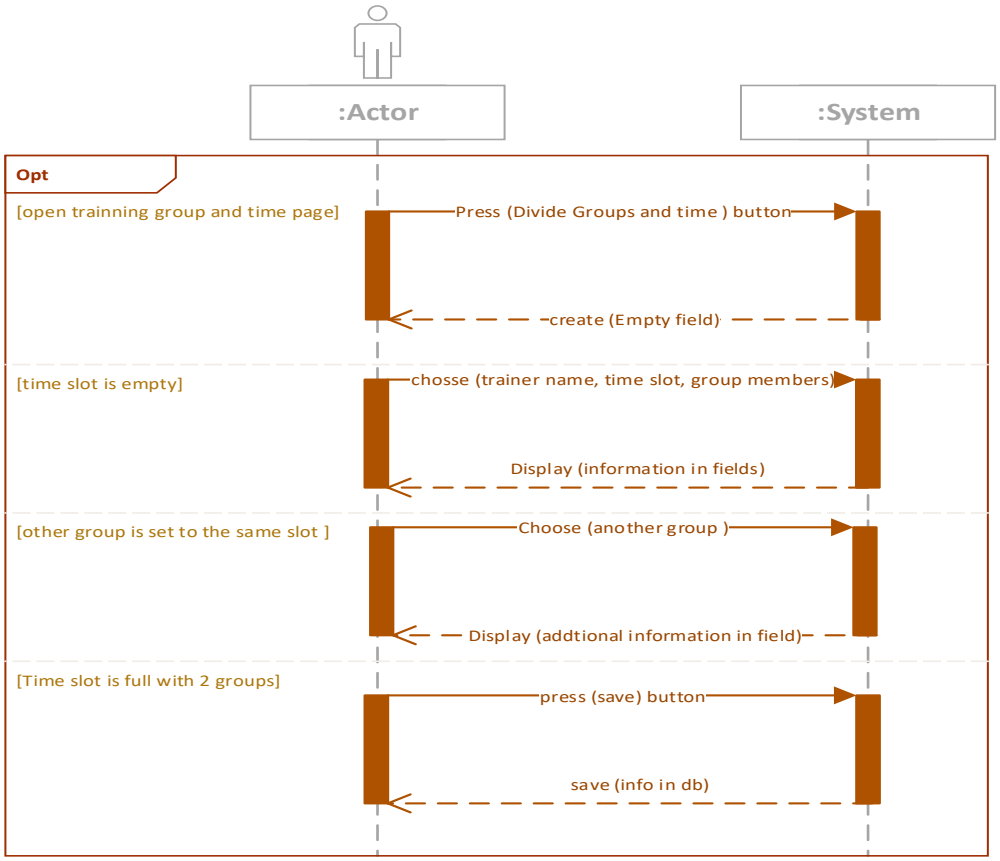


3. Sequence Diagram:

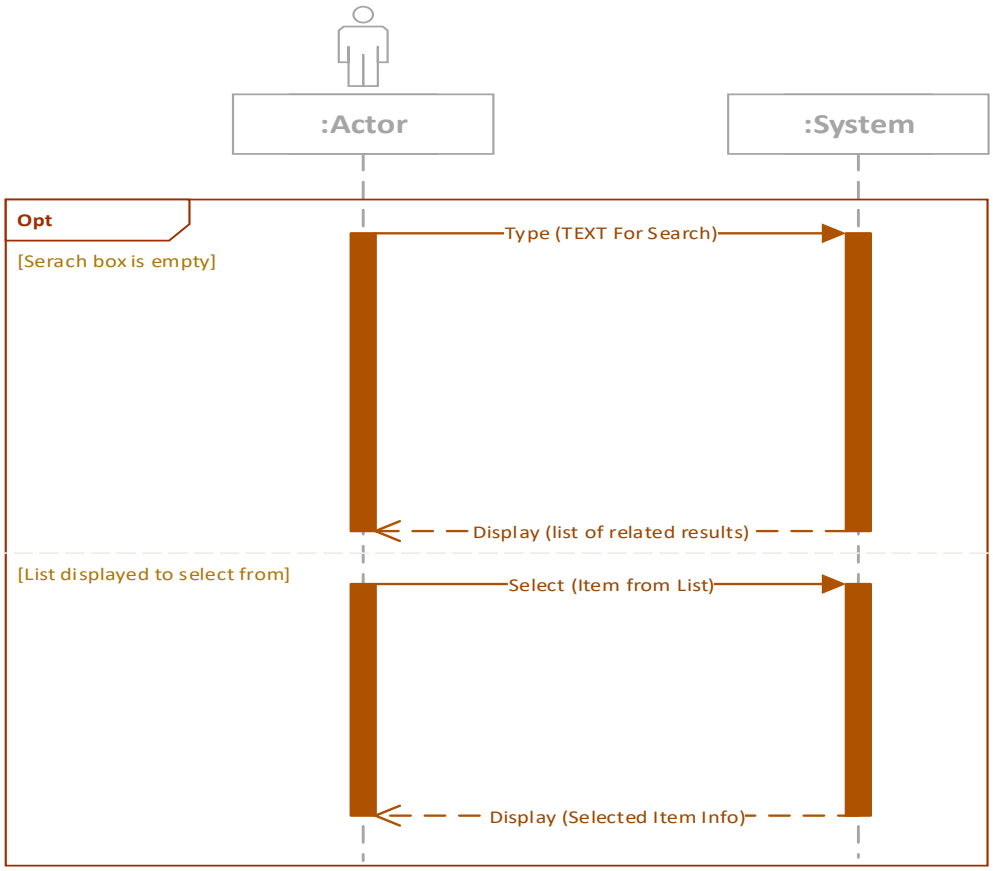
- **Training Program:**



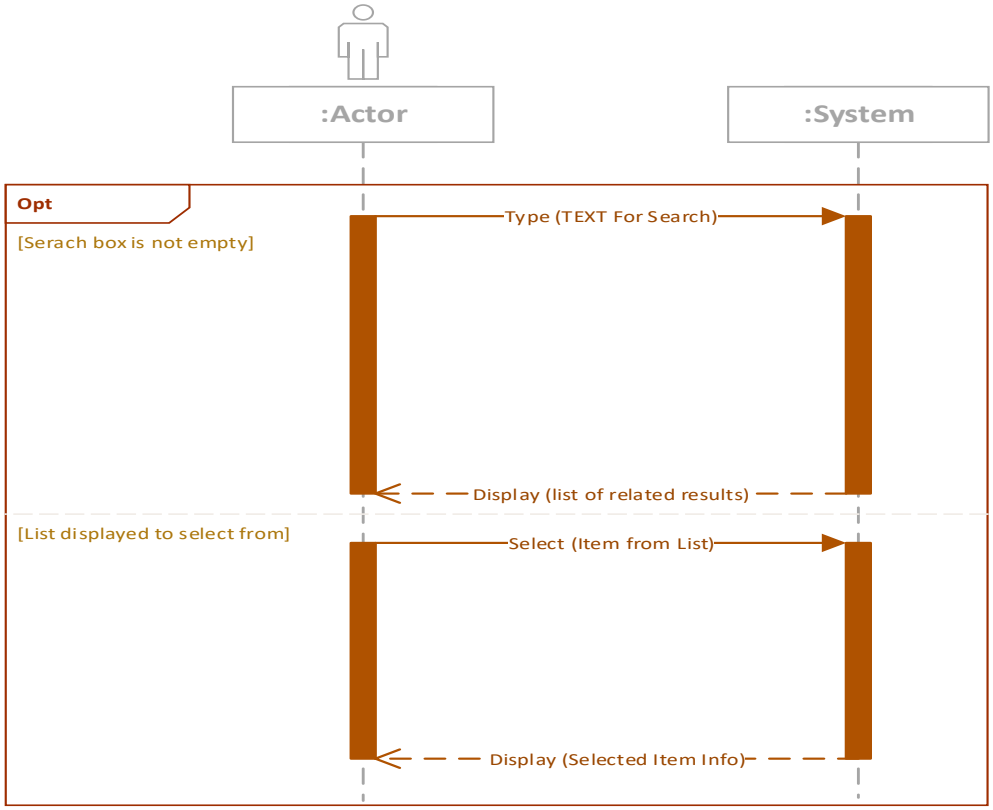
- **Dividing Time:**



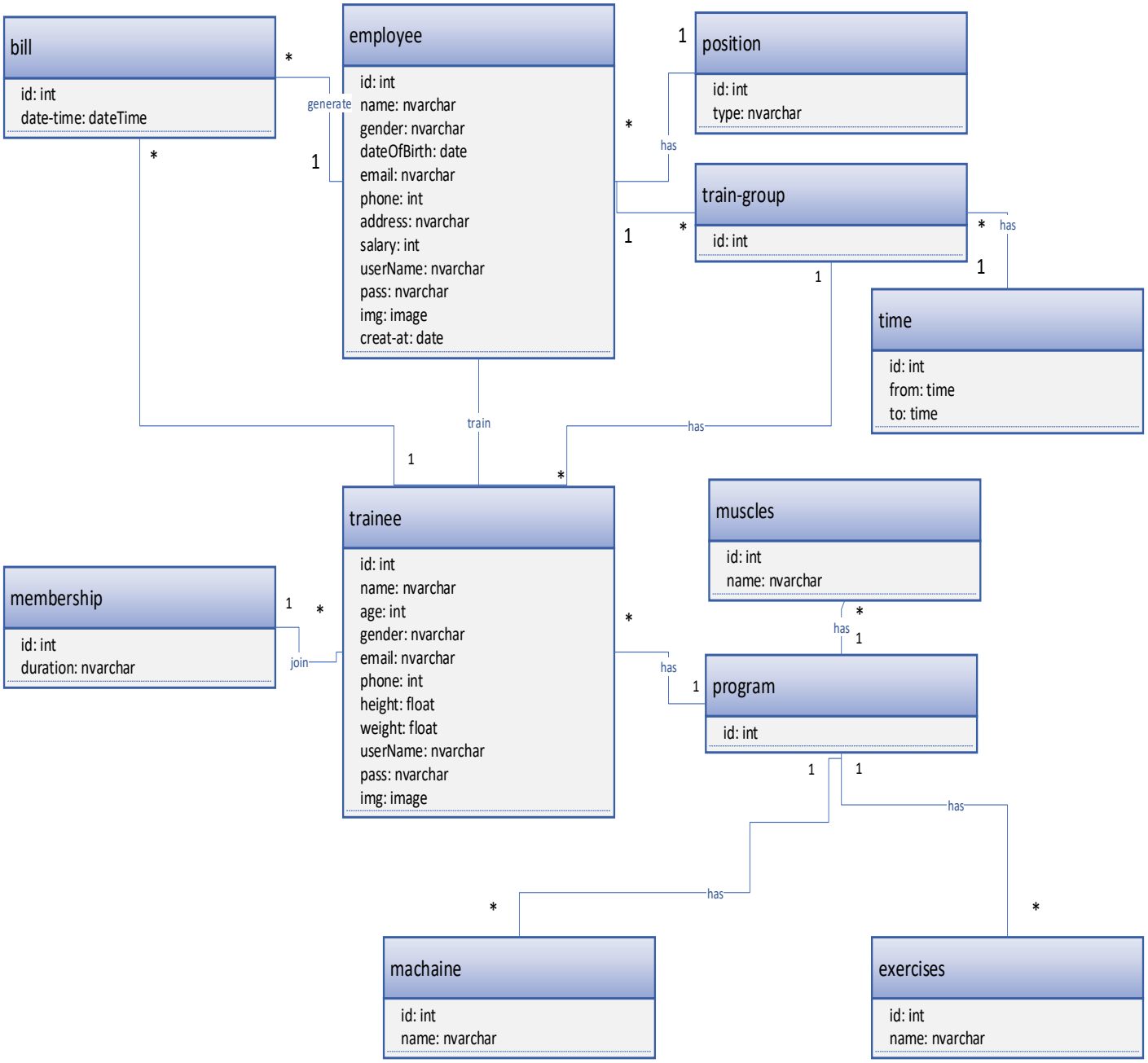
- **Print Status:**



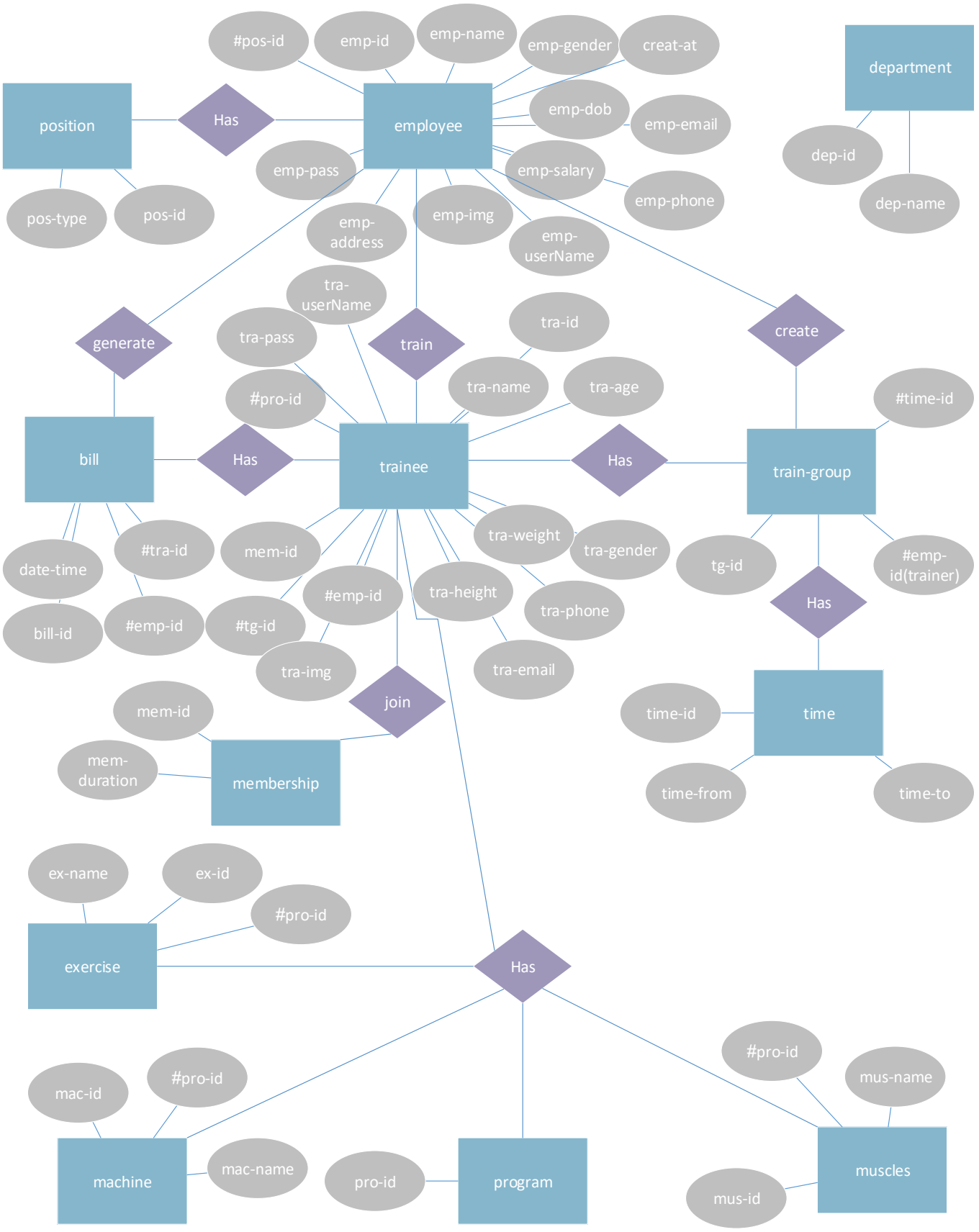
- **Search for Info:**



4. Class Diagram:



5. ERD:


#pos-id
emp-id
emp-name
emp-gender
creat-at
department
position
Has
employee
emp-dob
emp-email
dep-id
emp-pass
emp-salary
emp-phone
dep-name
pos-type
pos-id
emp-address
emp-img
emp-userName
tra-userName
tra-id
create
generate
tra-pass
train
tra-name
tra-age
#time-id
#pro-id
bill
Has
trainee
Has
train-group
tra-weight
tra-gender
#tra-id
mem-id
tg-id
#emp-id(trainer)
date-time
#emp-id
tra-height
Has
bill-id
#tg-id
#emp-id
tra-phone
tra-img
tra-email
join
time-id
time
mem-id
mem-duration
membership
time-from
time-to
ex-name
ex-id
#pro-id
exercise
Has
#pro-id
mus-name
mac-id
#pro-id
machine
mac-name
pro-id
program
muscles
mus-id

System Architecture

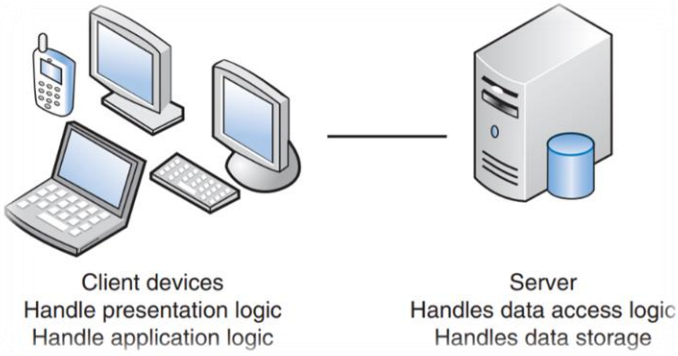
Most organizations today are utilizing or moving to client–server architectures, which attempt to balance the processing between client devices and one or more server devices. In these architectures, the client is responsible for the presentation logic, whereas the server is responsible for the data access logic and data storage. The application logic may reside on the client, reside on the server, or be split between both.

FFGMS under consideration will be based upon client-server architecture. It is possible that the architecture will have secure managed interfaces to isolate systems from illegal access. The Architecture may be a simple client-server system in which C# Applications are used to provide data access from a simple local or remote server that can be accessed by the system user (client) .

The (client) may have to cater for a range of events that may eventually need to access information. Access to information may be required to allow the Client to certain pieces of information stored within the overall data systems operated by the Gym Data local or remote Server.

We chose the client-server architecture because we will use the visual studio to create the logical view of the system containing the main functions by the system in the shape of events, and the Sql server as a data storage to handle data access logic from different clients, client-server architecture has also some benefits as:

- It is easy to increase or decrease the storage and processing capabilities of the servers
- Can support many different types of clients and servers
- If a server fails in a client–server architecture, only the applications requiring that server will fail. The failed server can be swapped out and replaced and the applications can then be restored.



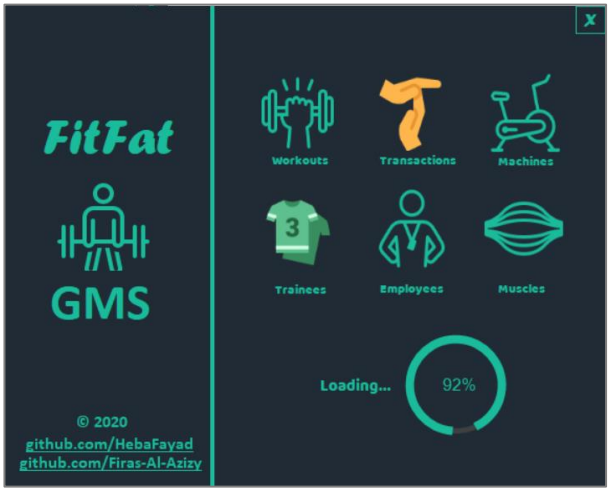
Chapter Three

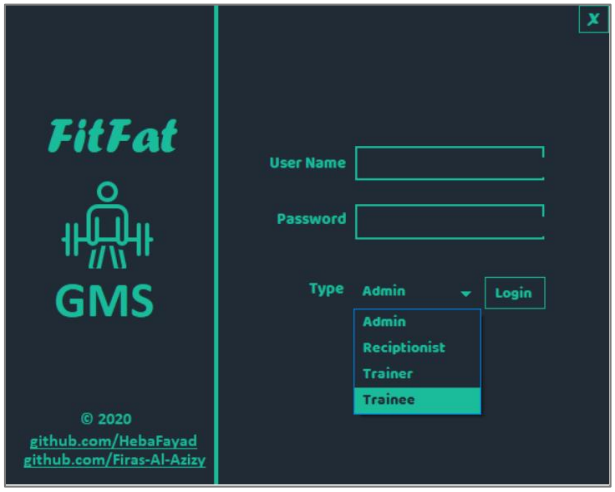
(Implementation)

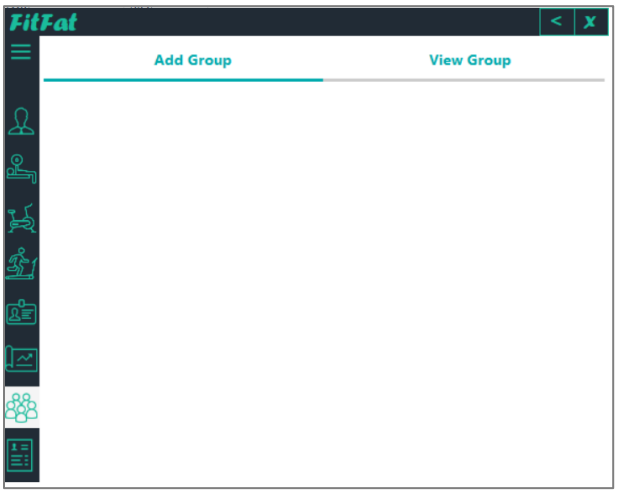
System Algorithms

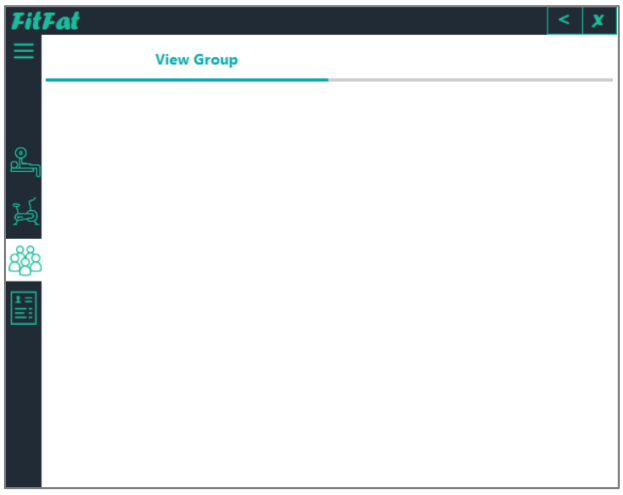
(soon)

System Interfaces:









System evolution

In some time of the future we may consider these changes to be added to the system.

User Interfaces:

More Interactive , Well-designed Interfaces

Hardware Usage Optimization:

Involve threading & memory management to make the system lighter.

Communications system:

Add a sub system to communicate with other users of the system in the network.

Software Enhancements :

Correcting a problem with the current system or improving the performance of operational software.

Chapter Four

(Testing)

Test Cases:

1. Training Program:

An operation of adding a training program to the main gym schedule is associated with:

1. Searching for the required trainee information to be displayed name therefor, the name input should be checked if it's true or existed or not empty.
2. Choosing the trainee to add to the program should be based on his chosen days for the training and the targeted muscles so those fields should be selected, if not the operation will not be completed.
3. Saving the new created program after creating it, if not, the operation will not be completed or saved to the database of the system.

2. Dividing Time:

An operation of dividing Time and groups in the main gym schedule is associated with:

1. Searching for the required trainer name + empty time slot + selected trainees to join the group, the trainer name + time slot + group members inputs should be checked if it's true or existed or not empty.
2. Dividing the groups should be based on time slots, if the selected one is empty and not used before then continue, if not the operation will not be completed.
3. Saving the new created Group after creating it, if not, the operation will not be completed or saved to the database of the system.

Appendices

1. Hardware requirements:

- 1gb RAM
- CORE 2 DUE PROSSESOR
- 1GB HARDDRIVE

2. Software requirements:

- WINDOWS OS
- NET FRAMWORK 4.6
- SQL SERVER BUNDLE