



DEPI – JAVA Final Project

Gym Management System

Name	Email
Amr Saber Fathy Hassan	Amr.saber.fathy@gmail.com
Felopatier Sayed Himaia	felosayed0@gmail.com
Ali Khamis Mahmoud Gebril	aligbril65@gmail.com
Mohamed Abdullah Yassin Ahmed	mohamedabdullahaz@gmail.com

Instructor Name: Dr/ Mina Younan

Software Development: Software Tester

Technical Provider: AST

Group Code: ONL1_SW D7_M1d

Main Scenario:

Develop an application to help a gym manager manage a GYM hall, with various equipment, trainers/customers, equipment's, and exercise plan, and simple scheduling of the customers subscriptions. Consider the following information gathered during the analysis process.

Project Analysis:

- **Basic Classes:** Gym halls, equipment, trainers, exercise plans and customers.
 - **The Gym-hall aggregates several equipment.**
 - **Each trainer works 8 hours a day in one hall.**
 - **A trainer submits a daily exercise plan with steps assigned to each equipment for a specified duration in minutes.**
 - **Each customer subscribes to a particular hall, at a particular time and date, with a particular trainer, with a specified exercise plan for one month.**
 - **System administrator can print reports, can save/load data on and from files.**
-

Extra Features:

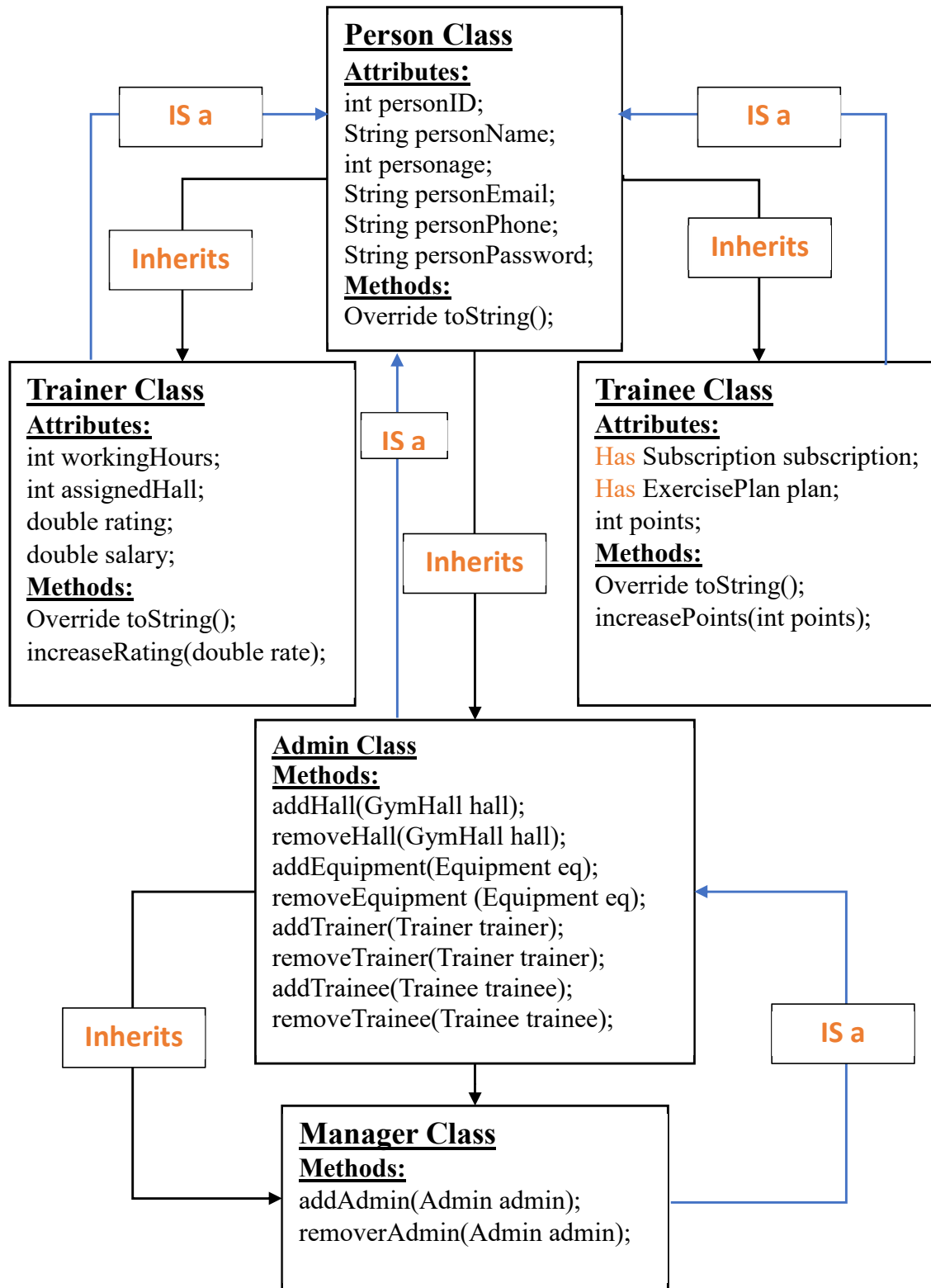
- **Connection to SQL Database to save and load data to and from it.**
- **Connection with Excel file to check the attendance of Trainers and Trainees with the date and for daily attendance registration.**
- **Connection with Text file to save the System Logs [Add/ Delete/ Update data from Database] with the email of the admin in charge of that action.**
- **Designing the project as a Maven project to be run easily on any OS without downloading all the related libraries manually.**
- **Display a main menu to the system user to handle all the actions without dealing with SQL commands to the code.**
- **A reminder to the system user about the expiration of the trainee's subscription while registering his/ her attendance on the system.**
- **The System is connected with Rest API (Using Spring Java).**
- **SMS notifications System to send notifications for trainees.**

Packages Distribution:

- **people Package contains:**
 - **Person as Parent Class.**
 - **Trainer as Sub Class inherits from Person. (Trainer is a Person)**
 - **Trainee as Sub Class inherits from Person. (Trainee is a Person)**
 - **Admin as Sub Class inherits from Person. (Admin is a Person)**
 - **Manager as Sub Class inherits from Admin. (Manager is an Admin)**
- **sqlConnection Package contains:**
 - **SQLConnection Class for Database connection, Table creation, Data insertion and Data deletion.**
 - **UpdateValues Class for updating Database inserted Data.**

- **SearchData Class** for searching for Data on the Database.
- **papers Package** contains:
 - **Subscription Class** to handle Trainees subscription type, Start Date, End Date and Renewal Date.
 - **ExercisePlan Class** to handle the duration of the plan and the steps included.
 - **ExcelSheet Class** to handle the connection with the Excel sheets for attendance for both Trainers and Trainees.
- **facility Package** contains:
 - **GymHall Class** to handle all the actions related to the gym halls like add a new gym hall, remove one from the system if this hall is under maintenance, add new equipment to a gym hall and assign a trainer for a gym hall.
 - **Equipment Class** to handle all the actions related to the equipment like adding a new one, removing or moving equipment to another gym hall.
- **main Package** contains:
 - **Logs Class** to handle the connection with a text file to record all the actions that are done to the system [Adding / Removing / Updating] Data.
 - **MainRun Class** to handle the simple command line user interface to help the final user [who may be can't code] to deal with the system without dealing with any SQL commands or any code.
- **notifications Package** contains:
 - **Class Initialization: Twilio.init:** This is a static block that initializes the Twilio client when the SMSNotification class is loaded. It uses the Account SID and Auth Token provided by the user to authenticate with the Twilio API.
 - **Class: SMSNotification:** This method is used to send an SMS message to a specified phone number using the Twilio API.

Classes Attributes and Relations:



Subscription Class

Attributes:

```
int subscriptionID;  
String subscriptionType;  
LocalDate startDate;  
LocalDate endDate;  
double price;  
double discount;
```

ExercisePlan Class

Attributes:

```
int exercisePlanID;  
List<String> exerciseSteps;  
int exerciseDuration;
```

Methods:

```
addStep(String step);  
removeStep(String step);
```

GymHall Class

Attributes:

```
int gymHallID;  
String gymHallName;  
Has List< Equipment > equipment;
```

Methods:

```
Override toString();
```

Equipment Class

Attributes:

```
int equipmentID;  
String equipmentName;  
String equipmentType;
```

Methods:

```
Override toString();
```

ExcelSheets Class

Static Methods:

```
createSheet();  
takeAttendance(Person person);  
createCard(Person person);
```

Logs Class

Static Methods:

```
createLogsFile();  
addLogs(Admin admin, Object  
appended, String operation);
```

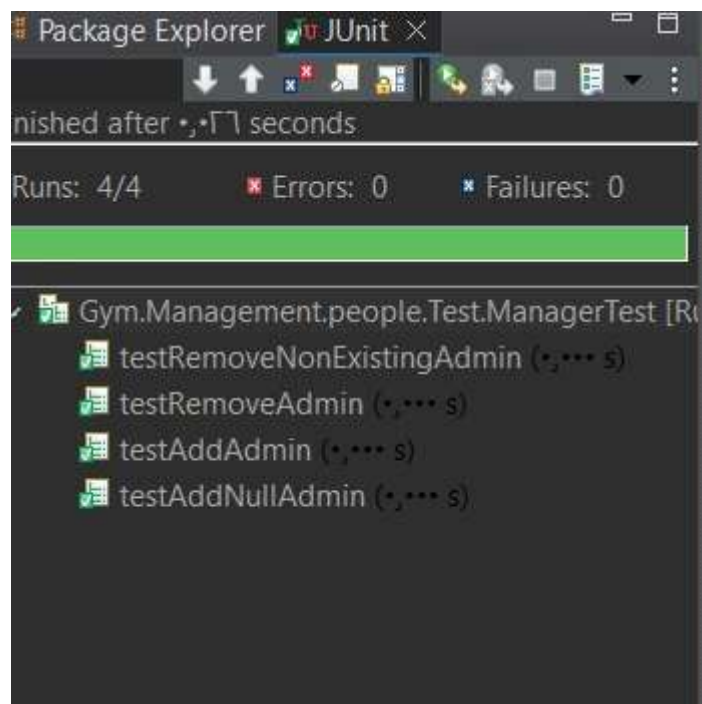
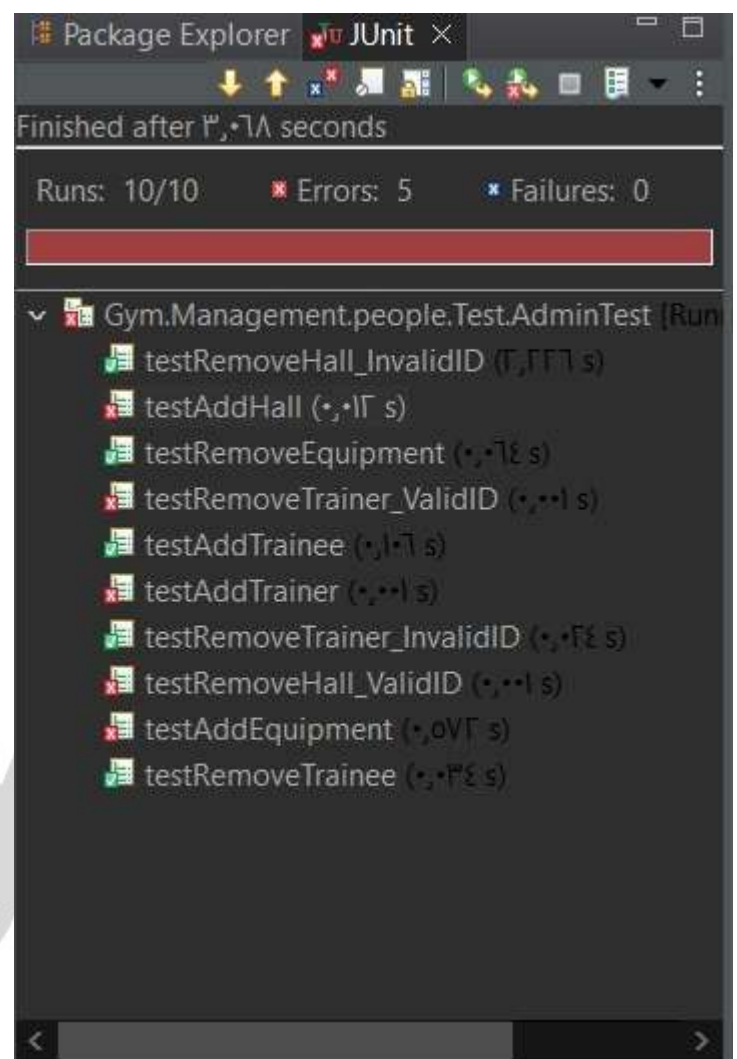
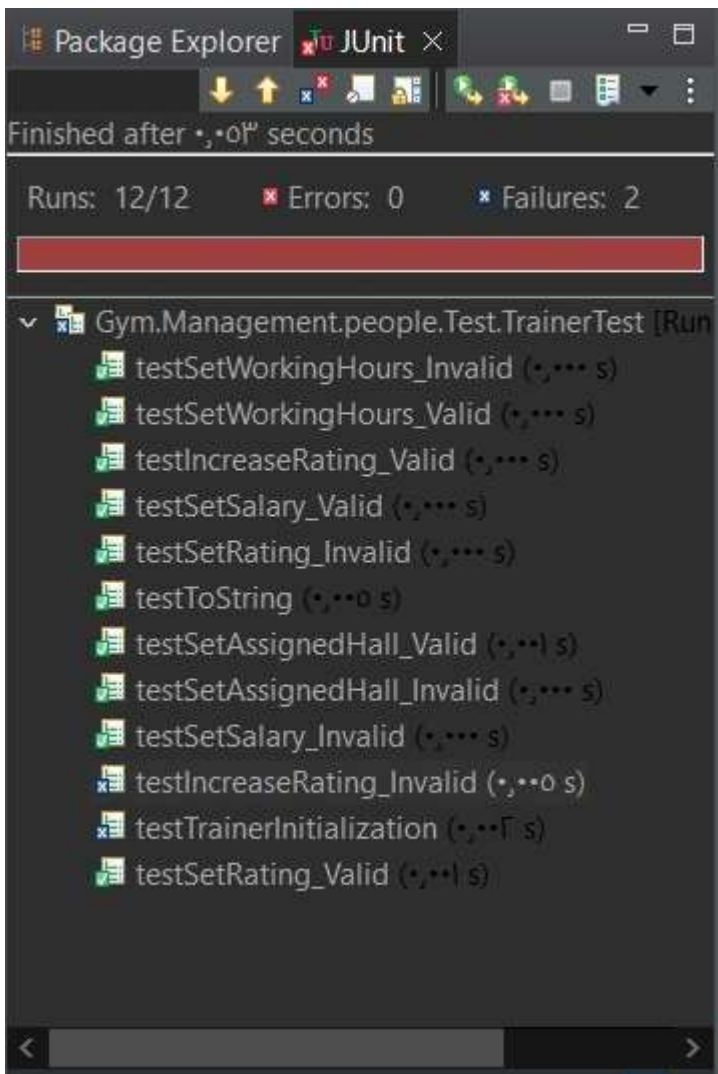
Conclusion:

In conclusion, we aimed to make a simple Gym Management System to handle all the possible actions that could be in a gym and save the data on a Database management system (My SQL) for future Data Analysis to help the final user (Gym owner) develop his work easily.

Project link on GitHub:

https://github.com/Amr-Saber-Fathy/DEPI_Final_Project

Samples of Unit Testing using JUnit:



Samples of API Testing using Postman:

Gym Management System / Trainees / Add Trainee

SaveShare

POST{{BaseURL}}/traineesSend

ParamsAuthorizationHeaders (8)BodyScriptsSettingsCookies

noneform-datax-www-form-urlencoderauwbinaryGraphQLJSONBeautify

```
1 {
2   "person": {
3     "ssr": "30507142700354",
4     "name": "Noor",
5     "email": "noor216@gmail.com",
6     "phoneNumber": "01143787674",
7     "accountPass": "noooooorj123456",
8     "age": 20
9   },
10  "points": 50,
11  "subscription": {
12    "subscriptionType": "silver",
13    "discount": 0.0
14  },
15  "explan": {
16    "exerciseSteps": "Push ups 20*3 + Bench Press 15*3",
17    "exerciseDuration": 15
18  }
19 }
```

Gym Management System / Trainees / Edit Trainee

SaveShare

PUT{{BaseURL}}/trainees/252Send

ParamsAuthorizationHeaders (8)BodyScriptsSettingsCookies

noneform-datax-www-form-urlencoderauwbinaryGraphQLJSONBeautify

```
1 {
2   "person": {
3     "ssr": "30507142700354",
4     "name": "Noor",
5     "email": "noor226@gmail.com",
6     "phoneNumber": "01543787548",
7     "accountPass": "noooooorj123456",
8     "age": 20
9   },
10  "points": 150,
11  "subscription": {
12    "id": 1,
13    "subscriptionType": "gold",
14    "startDate": "2024-09-30",
15    "endDate": "2024-11-29",
16    "price": 80.0,
17    "discount": 0.0
18  },
19  "explan": {
20    "id": 1,
21    "exerciseSteps": "Push ups 20*3 + Bench Press 20*4",
22    "exerciseDuration": 20
23  }
24 }
```

Gym Management System / Trainees / Delete Trainee

SaveShare

DELETE{{BaseURL}}/trainees?ssr=30507142700355Send

ParamsAuthorizationHeaders (6)BodyScriptsSettingsCookies

Query Params

<input type="checkbox"/> Key	Value	Description	⋮ Bulk Edit
<input type="checkbox"/> id	204		
<input checked="" type="checkbox"/> ssr	30507142700355		
<input type="checkbox"/> Key	Value	Description	

Samples of managing the project using Jira:

The image displays two screenshots of the Jira project management interface for a project named 'Gym' (Software project). The interface is in a dark theme and shows a Kanban board for 'classes Sprint 1'.

Top Screenshot:

- Header:** 'Gym' logo, 'Software project', 'Projects / Gym', 'classes Sprint 1', 'finish all basic classes in the estimated time', '10 days', 'Start stand-up', 'Complete sprint'.
- Left Sidebar:** 'PLANNING' (Timeline, Backlog, Board, Goals, Add view), 'DEVELOPMENT' (Code, Project pages, Project settings), 'Invite people'.
- Board:** Three columns: 'TO DO 7', 'IN PROGRESS', 'DONE ✓'. The 'TO DO' column contains four items: 'SCRUM-1', 'SCRUM-2', 'SCRUM-3', and 'SCRUM-4'. Each item has a description and a status icon (AG).
- Right Sidebar:** 'GROUP BY: None', 'Insights', 'View settings', 'Quickstart'.

Bottom Screenshot:

- Header:** 'Jira' logo, 'Your work', 'Projects', 'Filters', 'Dashboards', 'Teams', 'Plans', 'Apps', 'Create', '27 days left', 'Search', 'AG'.
- Left Sidebar:** 'Gym' logo, 'Software project', 'Projects / Gym', 'classes Sprint 1', 'finish all basic classes in the estimated time', '10 days', 'Start stand-up', 'Complete sprint'.
- Board:** Three columns: 'TO DO 6', 'IN PROGRESS 1', 'DONE ✓'. The 'TO DO' column contains four items: 'SCRUM-2', 'SCRUM-3', 'SCRUM-4', and 'SCRUM-1'. Each item has a description and a status icon (AG).
- Right Sidebar:** 'GROUP BY: None', 'Insights', 'View settings', 'Quickstart'.

Jira

Your work

Projects

Filters

Dashboards

Teams

Plans

Apps

Create

27 days left

Search

AG

Gym

Software project

Projects / Gym

classes Sprint 1

finish all basic classes in the estimated time

10 days

Start stand-up

Complete sprint

GROUP BY

Assignee

Insights

View settings

Assignee

PLANNING

Timeline

Backlog

Board

Goals

Add view

DEVELOPMENT

Code

Project pages

Project settings

TO DO 6

IN PROGRESS

DONE 1

ali gbril (3 issues)

as a backed i want to add details
The Gym-hall aggregates several
equipment.
SCRUM-2

as a backed i want to add A
trainer submits a daily exercise
plan with steps assigned to each
equipment for a specified
duration in minutes.
SCRUM-3

Create issue

Unassigned (4 issues)

as i backed i want to Each
customer subscribes to a
particular hall, at a particular
time and date, with a particular

Jira

Your work

Projects

Filters

Dashboards

Teams

Plans

Apps

Create

27 days left

Search

AG

Gym

Software project

Projects / Gym

classes Sprint 1

finish all basic classes in the estimated time

10 days

Start stand-up

Complete sprint

GROUP BY

None

Insights

View settings

PLANNING

Timeline

Backlog

Board

Goals

Add view

DEVELOPMENT

Code

Project pages

Project settings

TO DO 6

IN PROGRESS

DONE 1

as a backed i want to add details
The Gym-hall aggregates several
equipment.
SCRUM-2

as a backed i want to add A
trainer submits a daily exercise
plan with steps assigned to each
equipment for a specified
duration in minutes.
SCRUM-3

as i backed i want to Each
customer subscribes to a
particular hall, at a particular
time and date, with a particular
trainer, with a specified exercise
plan for one month
SCRUM-4

as a System administrator print
reports, save/load data on and
from files.

Invite people

Quickstart

Start Sprint

7 issues will be included in this sprint.

Required fields are marked with an asterisk *

Sprint name *

classes Sprint 1

Duration *

custom

Start date *

10/6/2024

8:34 PM



Planned start date: 10/3/2024, 12:00 AM

A sprint's start date impacts velocity and scope in reports. [Learn more.](#)

End date *

10/20/2024

12:00 AM



Sprint goal

finish all basic classes
in the estimated time

Cancel

Start