

<b>PepeFit</b>	<b>Version 1.0</b>
Supporting Requirements Specification	Date: 23/03/2018

# PepeFit

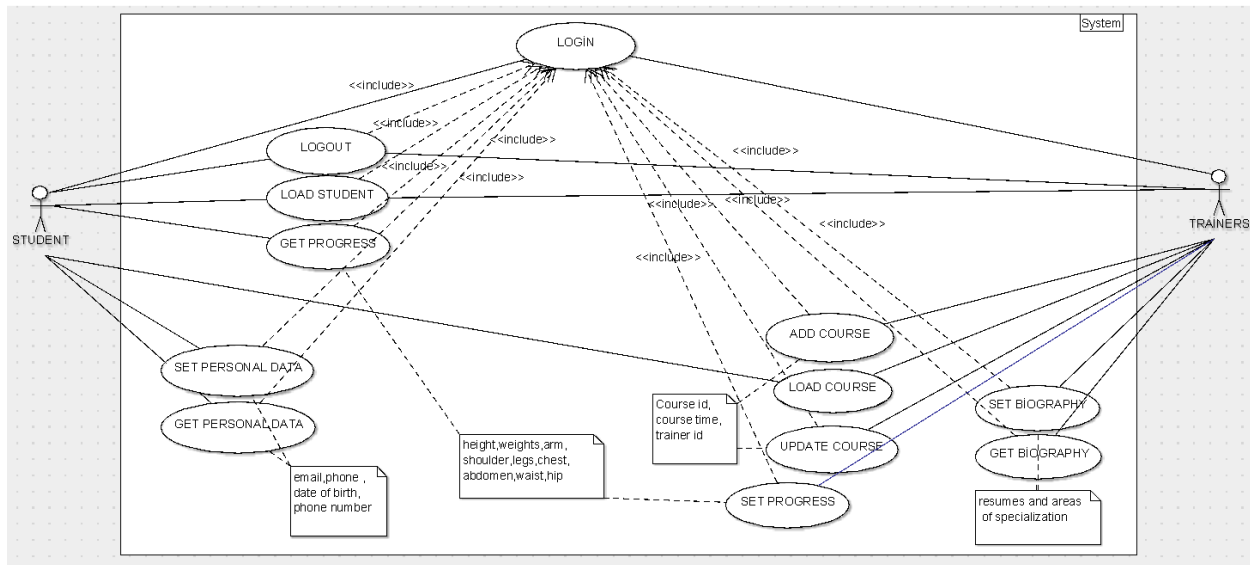
## System-Wide Requirements Specification

### 1. Introduction

This document will give you detailed information about PepeFit project. A detailed description about system's functionality and requirements will be mentioned in this Document. PepeFit aims for a better automation solution for Gyms and requirements shaped around that efficiency.

### 2. System-Wide Functional Requirement

The system's use case diagram is as in below:



To get more information about Use Cases you can check Tabular Descriptions in Appendix A. E/R Diagram of the system (Data Model) is in Appendix C.

<b>PepeFit</b>	Version 1.0
Supporting Requirements Specification	Date: 23/03/2018

USE CASE	USE CASE NUMBER	USE CASE CODE
LOGIN	1	SCWA-US-001
LOGOUT	2	SCWA-US-002
LOAD STUDENT	3	SCWA-US-003
GET PROGRESS	4	SCWA-US-004
SET PROGRESS	5	SCWA-US-005
GET PERSONAL DATA	6	SCWA-US-006
SET PERSONAL DATA	7	SCWA-US-007
ADD COURSE	8	SCWA-US-001
LOAD COURSE	9	SCWA-US-001
UPDATE COURSE	10	SCWA-US-001
SET BIOGRAPHY	11	SCWA-US-001
GET BIOGRAPHY	12	SCWA-US-001

- An actor who does not log in to the system can not process.
- Actor can login the system.
- The logged in actor can logout the system
- The logged in trainer can load students.
- The logged in trainer can set progress.
- The logged in actor can get progress.
- The logged in actor can get personal data.
- The logged in actor can set personal data.
- The logged in trainer can add course.
- The logged in actor can load course.
- The logged in trainer can update course.
- The logged in trainer can set biography
- The logged in trainer can get biography.

<b>PepeFit&lt;Project Name&gt;</b>	<b>Version: 1.0</b>
<b>Supporting Requirements Specification</b>	<b>Date: 23/03/2018</b>

### **3. System Qualities**

For creating a good system, we have to provide some qualities like usability, reliability, performance and supportability.

#### **3.1 Usability**

- The user shall be access easily his/her account.
- The user shall be easily login and logout from the system.
- The user shall be access what he/she wants at most 3 clicks.
- The user shall be freely navigate through the system.
- The system shall be at most 4 deep tabs.
- The user shall be do changes easily into his/her system.

#### **3.2 Reliability**

- The system must secure the information's of the users on any condition.
- The system must do the transactions well with database.
- The system shall have a good connection with DB.
- The system shall stay under control on a bad situation.
- When a user change something his/her account, system provides these changes at any time (durability).

#### **3.3 Performance**

- The system should be handle many users at the same time.
- Login time shall be less than 4 seconds.
- Logout time shall be less than 4 seconds.
- The system's response time shall be less than 5 seconds.
- The database connection time shall be less than 2 seconds.

#### **3.4 Supportability**

- We will perform maintenance of the system monthly.
- Any bug will be resolve immediately.
- Users can be reach our app via any mobile device.

<b>PepeFit&lt;Project Name&gt;</b>	<b>Version: 1.0</b>
<b>Supporting Requirements Specification</b>	<b>Date: 23/03/2018</b>

## **4. System Interfaces**

### **4.1 User Interfaces**

User Interface should be user-friendly and easy to navigate through the system's functions. Detailed User Interface is added in Appendix B Document.

#### **4.1.1 Look & Feel**

User interfaces is very important in a system. Because they are the only part that interact with the user. And user must have a good experience on our system. So user interface has a key role in the system. For to providing this, it should be simple and light. It should not tire the user when he/she navigate through the system. Color shall be designed for this navigating. UI should not be complex. It should not have more than 3-4 deep links. It should not have colors that can be tire user's eyes.

#### **4.1.2 Layout and Navigation Requirements**

Layout can not be complex. So when a user login to the system, he/she will encounter a simple layout. All the functionalities can be reach from the main window and can navigate between these functionalities. Each functionalities have its own window. User will be able to easily do his/her work via these tabs(windows). We have three user types. They have different layout, but have same simplicity for navigating via functionalities. And there should not be an ambiguity while navigating through windows. It can be confusing for the user. Navigation bars' names and windows' names should not be confusing or very similar. Because user can be confused while navigating

#### **4.1.3 Consistency**

One of the requirement for a good user interface experience is consistency. The windows should match its content, navigation bar and the other functions should be consistent with its content. And it should be give the same experience to the user on every time or should provide a better experience while user using that window.

#### **4.1.4 User Personalization & Customization Requirements**

For creating many user types, we have to create different layouts, doing customization for that user type. In our system, admin, user and trainer have different layout. Because admin should be able to control over the system so we have providing this functionality with a different layout customization. Trainer should be able to control his class or a user should be able to organize his/her schedules. So basically we have to customize for every user and theirs functionalities.

### **4.2 Interfaces to External Systems or Devices**

#### **4.2.1 Software Interfaces**

We are doing a Gym Automation. So we have to store the data of the users, trainers etc. For this purpose we have to use a Database. For this we will use MySQL database and our developing language is Java. So we have to connect with this Database and Java Application. We can accomplish this with Hibernate. It's a tool that provides a connection between Database and our application.

#### **4.2.2 Hardware Interfaces**

We don't have any hardware interfaces.

#### **4.2.3 Communications Interfaces**

We don't have any communication interfaces.

<b>PepeFit&lt;Project Name&gt;</b>	<b>Version: 1.0</b>
<b>Supporting Requirements Specification</b>	<b>Date: 23/03/2018</b>

## **5. Business Rules**

### **5.1 Course Related Rules**

#### *5.1.1 Number of Courses That User Can Take Daily Rule*

If user tries to take one more course in that day, system won't allow to do that.

#### *5.1.2 Course Quota Rule*

If user tries to take a course that its quota full, system won't allow to do that.

#### *5.1.3 Course Time Rule*

If user tried to change his/her course time at the same day, if course's quota is full at that time, system won't allow to change the his/her course time.

#### *5.1.4 Swapping Course Rule*

If two users want two swap their courses, system won't allow to do that. They have to contact with the trainer.

#### *5.1.5 Penalty Rule*

If user did not attend the courses three in a row, system will give him/her a penalty that he/she won't take course for a week

### **5.2 User Related Rules**

#### *5.2.1 User Creation Rule*

Only admin can create an user.

#### *5.2.2 User Information Update Rule*

Only trainer or admin can update user's information, his/her body specs that changing in a time.

## **6. System Constraints**

Our system will be written in Java. We will use MySQL as a Database and connect with them Hibernate. On the server side, we will use Tomcat server and for the storing these we will use Google Cloud. In the server side, we will user JSF for connecting to UI and Server.

Our system can be used as a Browser Web Application. You can access anywhere if you have a browser. And we will use Eclipse for develop this java project.

<b>PepeFit&lt;Project Name&gt;</b>	<b>Version: 1.0</b>
<b>Supporting Requirements Specification</b>	<b>Date: 23/03/2018</b>

## **7. System Compliance**

### **7.1 Licensing Requirements**

We are developing this system for the gyms. So if a gym wants to use this application, it has to contact us and buy a license from us according to use this application. Other than that no one can use it.

### **7.2 Legal, Copyright, and Other Notices**

Any and all rights in the PepeFit Software Service are and shall remain the exclusive property of SadePepe Inc. . For purposes of clarity, PepeFit Software Service includes any and all content on the software, such as logos, graphic images, texts, trademarks data and other materials(together,"content") as well as any part of the PepeFit Software Service.

### **7.3 Applicable Standards**

We will use Java 8.0 during the project. Oracle provides Java 8.0 coding standards. We will use Tomcat 8 server. Apache provides server standards for Tomcat 8.

PepeFit runs only via Browser with HTML5 standard.

Our application's language is English. We will use Europe's Metric System for providing the body specs.

## **8. System Documentation**

When the product comes out, we will provide detailed user guidance manuals. For admins, for trainers and for the users. These manuals will be different with each other corresponding their roles in the system. They will include "How the system works ?", "How can you access the PepeFit ?" etc.

The document will be prepared by the whole SadPepe members.

## **9. Appendices**

- SRS - Appendix A - Tabular Descriptions of Use Cases
- SRS - Appendix B - GUI Prototypes
- SRS - Appendix C - E/R Diagram