TED UNIVERSITY

GYM MANAGEMENT SYSTEM

Software Requirements Specification (SRS) Document
27.04.2023

Team Members:

Mehmet Koç –
Ecem Sıla Gök –
Ahmet Efe Ersoy –
Pelin Hamdemir –
İrem Beşiroğlu -

Revision History

Name	Date	Reason For Changes	Version
Online Gym Training SRS	27.04.2023	Creation of a new SRS document	1.0.0

TABLE OF CONTENTS

1. Introduction 1
1.1 Purpose 1
1.2 Document Conventions 1
1.3 Intended Audience and Reading Suggestions 1
1.4 Product Scope 2
1.5 References 2
2. Overall Description 3
2.1 Product Perspective 3
2.2 Product Functions 3
2.3 User Classes and Characteristics 4
2.4 Operating Environment 5
2.5 Design and Implementation Constraints 5
2.6 User Documentation 5
2.7 Assumptions and Dependencies 6
3. External Interface Requirements 6
3.1 User Interfaces 6
3.2 Hardware Interfaces 7
3.3 Software Interfaces 8
3.4 Communications Interfaces 8
4. System Features 9
4.1 Customization Form 9
4.2 Personalized Training Plans 10
4.3 Progress Tracking 11
4.4 Communication System 12
4.5 External Resource Library 12
4.6 Payment Management 13
5. Other Nonfunctional Requirements 14
5.1 Performance Requirements 14
5.2 Safety Requirements 14
5.3 Security Requirements 15
5.4 Software Quality Attributes 15
5.5 Business Rules 15
6. Other Requirements 16
Appendix A: Glossary 16
Appendix B: Analysis Models 18
Appendix C: To Be Determined List 23

1. Introduction

1.1 Purpose

Software Requirements Specification (SRS) Document for a Java-Based Gym System (Version 1.0, Revision 1.0)

The purpose of this document is to define the requirements for a gym system that will be used to manage gym memberships, track member progress, and provide workout plans and guidance. This SRS document describes the specific features and capabilities that the gym system must have in order to meet its intended requirements. These requirements are categorized into functional requirements (which describe what the system should do) and non-functional requirements (which describe how the system should perform).

The SRS document also includes details about how the gym system will be designed and how the user interface will look and function. This may include information about the system architecture, software components, data structures, and algorithms used in the system design, as well as the efficiency and effectiveness of the system. The revision number is included to indicate that this is the initial version of the SRS document, and future revisions may be made to update or modify the requirements as needed. The version number indicates that this is the first major release of the gym system and may be updated in the future to reflect significant changes or updates to the system.

This document is expected to be useful to developers working on the project, the gym owners, trainers, accountants, and future users.

1.2 Document Conventions

Product: The outcome of the project, the end product.

User/Member: Person who has subscribed to the gym membership.

Trainer/Coach: Person who is employed by the gym and works with users to create training plans.

1.3 Intended Audience and Reading Suggestions

The SRS report for the gym system is intended for multiple types of readers, including developers, project managers, testers, and documentation writers. This document is organized to provide a detailed description of the requirements of the product, as well as the system design and user interface specifications.

The SRS report contains the following sections:

General Description: This section provides a high-level description of the gym system, including the key features and functions that the system must perform.

External Interface Requirements: Hardware interfaces specify the physical components that the system must interact with, such as fitness tracking devices, Software interfaces describe the software components that the system must interact with, such as database management systems or payment gateways and communication interfaces specify the protocols and methods used for communication ways. This may include integration with social media platforms or email and messaging systems for sending notifications and reminders to gym members.

System Design: This section describes the overall system architecture of the product, including software components, data structures, and algorithms used in the system design.

Non-functional requirements: Performance, safety, and security requirements for providing well-secured high usage.

Appendices: This section includes any additional information that may be useful, such as a glossary of terms or a list of acronyms.

For developers and testers, it is recommended to start with the General Description section to gain an understanding of the purpose and scope of the gym system. They should then proceed to the Functional Requirements and Non-Functional Requirements sections to gain a detailed understanding of the specific requirements that the system must meet.

For project managers and documentation writers, it is recommended to start with the General Description section to gain an overview of the gym system. They should then proceed to the System Design and User Interface Design sections to understand the technical details of the system and how it will function from a user perspective.

Overall, the SRS report is organized in a logical and easy-to-follow manner, ensuring that all readers can quickly and easily understand the key requirements and specifications of the gym system.

1.4 Product Scope

The main aim of this software is to simplify the process of managing gym operations and provide a more personalized experience for gym members. The software helps gym owners and trainers manage member information, create customized workout plans, track progress, and communicate with members. By using this software, gym owners and trainers can save time, increase efficiency, and improve the overall experience for their members. The objectives and goals of this gym system are to improve gym operations, increase member satisfaction, and ultimately grow the gym's membership base.

1.5 References

Ref 1. IEEE Computer Society, "IEEE Recommended Practice for Software Requirements Specifications," IEEE Std 830-1998, Oct. 20, 1998.

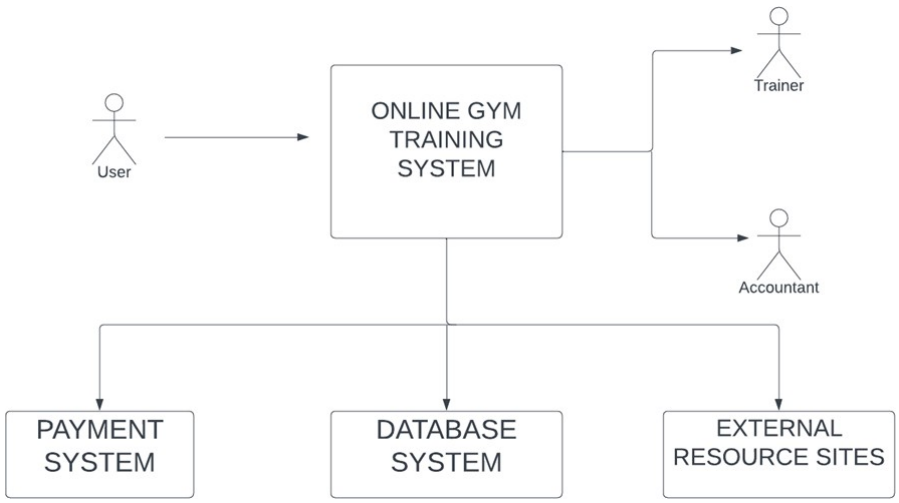
2. Overall Description

2.1 Product Perspective

The gym system specified in this SRS is a new, self-contained product. It is being developed to provide a digital platform for gym members to manage their memberships, track their progress, and receive guidance on workout plans. The system is not intended to replace any existing systems, but rather to enhance and streamline the gym membership experience.

The major components of the product include a user interface for members to interact with the system, a database to store member information and workout data, and a set of algorithms and data structures to provide workout plans and guidance. The system will also need to interface with external systems such as payment processors and email providers to facilitate membership payments and notifications.

A UML Context diagram for the gym system is shown below:



2.2 Product Functions

User Authentication and Registration: Allow users to create a new account and login to their existing accounts.

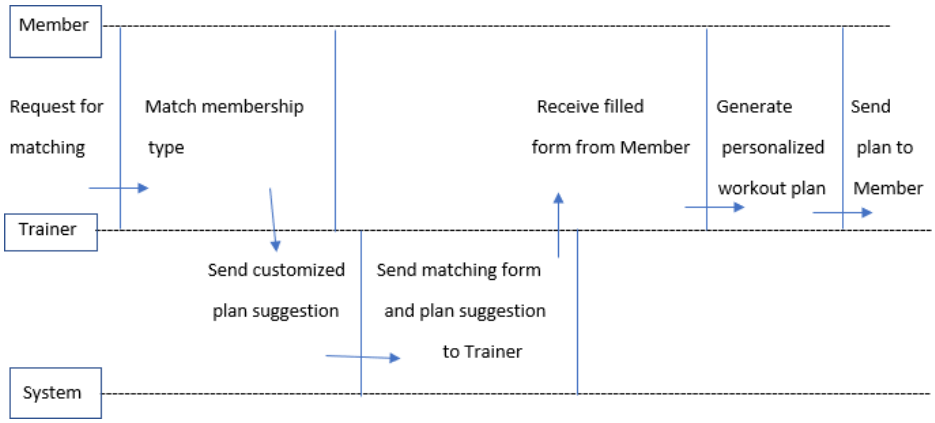
Profile Management: Enable users to manage their personal profiles, including updating their personal information, uploading profile pictures, and setting fitness goals.

Exercise Tracking and Logging: Allow users to track their exercises and log their progress over time.

Workout Plan Generation: Provide users with customized workout plans based on their fitness goals and personal preferences.

Video Instructional Guides: Provide users with video tutorials and instructions for exercises to help them perform them correctly.

Payment Processing: Enable users to purchase gym memberships, pay for personal training sessions, and other services through the system.



2.3 User Classes and Characteristics

Beginners primarily use the system for guided workouts and basic exercise routines, have limited technical ability, and may require a user-friendly interface and detailed instructions. Beginners have a limited educational level or experience in fitness and may require detailed instructions and guidance.

Intermediate Users use personalized training plans, goal tracking, and progress monitoring features. Intermediate Users have more technical expertise and may require more advanced features and customization options. They have some educational level or experience in fitness and may require more advanced training plans and techniques.

Advanced features and qualified technical expertise are adopted by trainers as well as intermediate users. Trainers, whose fields are yoga, cardio, and strength training, use the system to create and customize training plans for their clients. In addition, they require elevated security or privilege levels to access and manage their clients' information and training plans.

Advanced Users require more specialized and advanced workout routines and techniques, which may not be available to other user classes. Advanced Users require even more advanced technical features, such as real-time tracking and advanced workout analytics. Advanced Users have extensive educational level or experience in fitness and may require access to specialized workout routines and techniques.

The most important user classes for this gym system are the Beginners, Intermediate Users, and Trainers. These user classes are likely to be the primary users of the system and have the most significant impact on the system's success. They are likely to use a wide range of product functions and require different levels of technical expertise and security or privilege levels.

2.4 Operating Environment

The software operates in a web-based environment and should be accessible from any device with internet access, including computers, tablets, and smartphones.

As for hardware requirements, the software is designed to work on a wide range of devices, from low-end to high-end devices, to ensure broad compatibility. The operating system requirements depend on the device used to access the software.

The software also works with the bank's API system and database system. This means that the software should be compatible with the programming languages and protocols used by the bank's API system, and it should be able to interact with the database system to store and retrieve data seamlessly. Additionally, the software is designed to operate efficiently and securely in this environment, with appropriate security measures in place to protect sensitive information.

2.5 Design and Implementation Constraints

Databases, Interfaces, and tools: The system interfaces with other systems, such as social media or health tracking systems. In addition, the development team may be required to use specific technologies or tools, such as programming languages, development frameworks, or integrated development environments. Also, the system has a database to store user data or workout plans.

User input: The system relies on users checking off the exercises they do. If they don't do this accurately or at all, the system's data will be inaccurate, which affects the user's experience.

External video links: The system relies on external video sources for exercise demonstrations. If these resources become unavailable or are removed, it limits the functionality of the system, and the system searches for new resources for itself; if it does not find them, the system is turned off for a while to be replaced.

Payment system API: The product uses an API, which is a payment system. If this system does not work properly or stops working, it limits the functionality of the system. In this case, the system tries to find a new payment system or fix the existing one.

Localization: It may need to be localized for different languages and regions, which may affect the design and implementation of the system. In this case, the program will be revised.

Memory requirements: The product may require significant memory resources to store workout plans, videos, and user data. This could limit the system's ability to run on devices with limited storage or memory.

2.6 User Documentation

The user documentation for the gym system includes a user manual in the form of an online file. This manual provides step-by-step instructions on how to use the system, including how to create an account, set up a workout plan, track progress, and access additional features. The manual also includes screenshots and illustrations to help users visualize the different screens and interactions within the system.

In addition to the user manual, the gym system also includes other user documentation components, such as online help and tutorials. These components would provide additional guidance and support for users who need help getting started with the system or who want to learn more about its features and functionality.

The user documentation will be delivered in digital format, accessible through the system. The documentation will be designed to be easy to read and navigate, with clear instructions and helpful tips.

2.7 Assumptions and Dependencies

Third-party or commercial components: API for payment or database for managing certain features and functionality are the requirements stated in the SRS assume that these components will function as expected and meet the necessary specifications. If these assumptions are incorrect, or if the third-party components change or become unavailable, it could impact the development and functionality of the gym system.

Operating environment: The SRS assumes that the gym system will be developed and deployed in a specific operating environment, such as a particular operating system. If the operating environment changes or is incompatible with the product's requirements, it affects the system's performance and functionality.

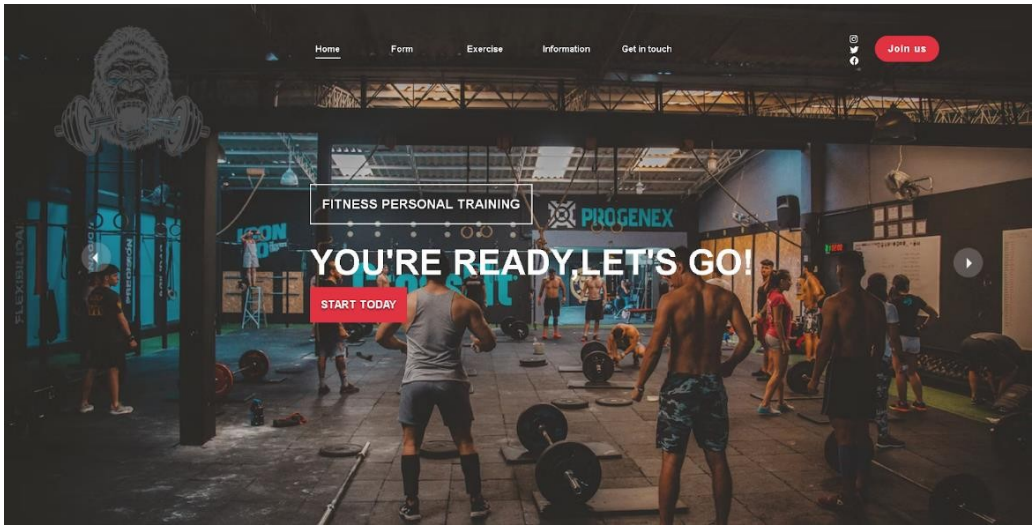
Dependencies on external factors: The gym system may have dependencies on external software components, such as libraries or frameworks, which are being reused from other projects. The SRS assumes that these components will be available and compatible with the product's requirements. If these assumptions are incorrect, or if the external components change or become unavailable, it could impact the development and functionality of the gym system.

Since users can easily access the user manual, it is not thought that they will make mistakes that may affect the functionality of the system.

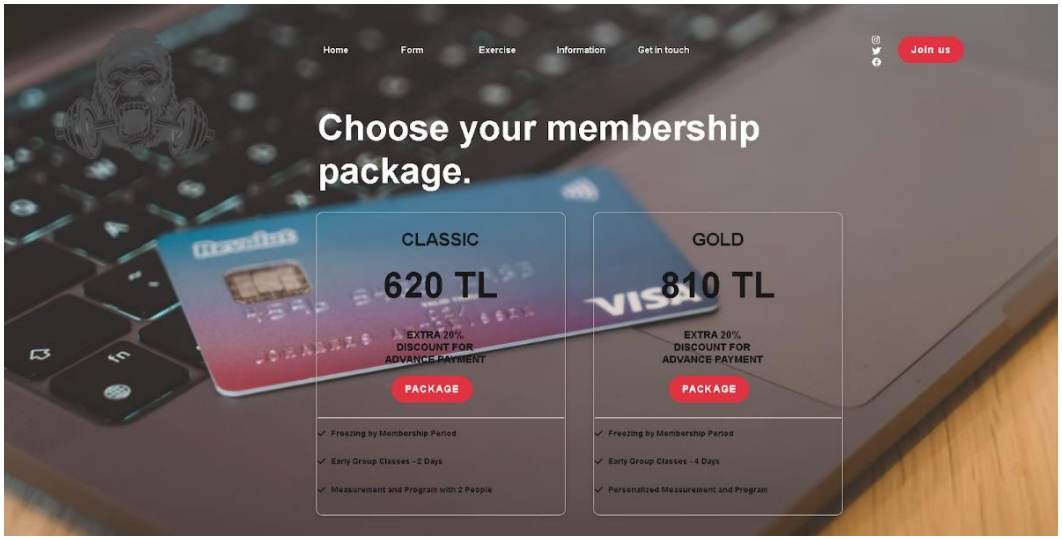
3. External Interface Requirements

3.1 User Interfaces

When the user logs in to the site, the main functions of the site are "Home", "Form", "Exercise", "Information", "Get in Touch". In addition, it is possible to "Start Today" and "Join Us" sections to register and pay with the membership system. There is also a communication section via social media.



When clicked on "Join Us" or “Start Today” sections and entered in the membership system, the membership page is displayed. The user can choose from the available membership packages to their liking and subscribe to it. After subscribing to one of them and registering to the membership system, the user is then considered a member and can log in whenever we want.



After subscription, the user is presented with a form. In this form, data such as the user's name, height, fat percentage is taken and is used to plan their training by their coaches.

On the Exercise page the user can select the workout they want to do from their training plan. Once they check off the exercise as done, the system records it, and this is rewarded by 'achievement points' that the system gives as rewards.

In the "Sources" section, the system provides the user with external source links to the source that could be of use to them. In addition to this, in the "Highlights" section, it is possible to see images of the popular external content.

In the "Get in Touch" section, the system presents the user with a simple text box that they can write their message in, which is then sent to their trainer.

3.2 Hardware Interfaces

In order to create a fluid and effective interaction between software and hardware components, the Online Gym Training website, a JavaScript and CSS-based platform, effortlessly incorporates both logical and physical qualities into its interfaces.

The website, which uses HTML, CSS, and JavaScript coding; is made to work on a variety of devices, including desktop, laptop, tablet, and smartphone models. Users may access the platform's functions, such as filling out forms, working out, getting information, processing payments, and contacting coaches, across a variety of devices thanks to this interoperability. Customers benefit from a user-friendly experience since the data and control interactions between the software and hardware are tuned for real-time responsiveness. Communication protocols are used to enable secure data transmission, safeguarding users' private data and payment information.

The system will be using the Regional Database Management System (mentioned as RDBMS from now on) provided by Amazon Web Services (mentioned as AWS from now on) for all frontend and backend operations it will perform.

3.3 Software Interfaces

The website depends on a reliable database system to maintain user data, workout plans, and other crucial information. User profiles, exercise progress, and pertinent multimedia content, including videos and articles, are all stored in this database. The website can provide further services like tracking user performance or securely processing payments by integrating with APIs like fitness tracking APIs or third-party payment APIs.

User registration information, workout preference form inputs, and user-generated material (like reviews or feedback) are all examples of incoming data pieces. Workout recommendations, educational content, and notifications or alerts are examples of outgoing data elements. These data points are used for a variety of tasks, including developing customized exercise programs, offering resources, and preserving user interest. For the website to function properly, a number of services are needed, including safe authentication, data storage and retrieval, the distribution of multimedia content, and effective communication between various components. These communications can be facilitated by the use of GraphQL or RESTful APIs, guaranteeing a modular and scalable design.

Application programming interfaces (APIs) and the specific protocols governing them should be separately documented, with each API endpoint's structure, required inputs, and anticipated outputs described. This documentation will serve as a reference for programmers and guarantee platform-wide uniformity. User profiles and workout progress are examples of data that should be shared between software components in order to personalize experiences and keep users interested. If necessary, an implementation constraint can be provided that prohibits the use of a particular data-sharing method, such as a global data area in a multitasking operating system.

3.4 Communications Interfaces

A fitness platform that integrates form-filling, activity tracking, information access, payment processing, and communication elements is the Online Gym Training website, which is built using JavaScript and CSS. As a result of the platform's incorporation of numerous software elements, databases, APIs, and tools, devices are compatible, data is transmitted securely, and user and coach interactions with the system are effective. To improve user experience and security, communication features like email compatibility, web browser compatibility, and standardized message formatting are addressed. Overall, the website bridges the gap between technology and individual health by providing a thorough, user-friendly, and secure fitness solution.

E-mail: With the e-mail part, two protocols are needed to ensure the mutual communication between users and coaches. One of them is SMTP, which is a set of rules about how an e-mail is sent and received over the internet in general. The other is POP, which allows users to download email messages to their local device, where they can be accessed even when the device is offline. In addition to these two protocols, we use protocols such as IMAP to communicate E-mail.

Electronic Form: With Electronic Form standard protocols such as HTTP POST and GET are used when collecting data from databases.

Network: Special protocols are used to communicate our network system between the servers of our website and the user's servers. For example, with HTTP/HTTPS and TCP/IP protocols, we ensure security and establish our network system in a way that adheres to the actual rules that enable communication.

Payment: Protocols provided by the internet are used when adding the payment system for membership to our website. With SSL/TLS protocols, we have added secure protocols applied to shopping systems on the

web to our site. In addition, we have developed this security need with protocols such as EMV and 3D Secure.

4. System Features

4.1 Customization Form

4.1.1 Description and Priority

The system has a customization form that lets the gym members express their personal needs such as their own fitness goals, their preferred workout type, whether they have any physical injuries that the trainers should be mindful of etc. This feature is a medium priority feature in our system as it is a tool to provide the users with a high-quality gym experience uniquely created for them and thus increases satisfaction rates.

Rating this feature in terms of benefit, we can say that it is a 6 out of 9 since it is what separates the gym experience ve offer from others. The penalty for not having this feature is a 6 because not having it does not align with our 'personalized gym experience' outlook and will cause lower satisfaction in our customers. Since it will be in a simple form format mostly consisting of multiple-choice questions, the cost of developing it will not be too much; therefore we can rate it a 4 out of nine. The risk factor for this feature is a 6 since the users might not be fully aware of what they want and need, causing them to fill out the form incorrectly.

4.1.2 Stimulus/Response Sequences

1. After subscribing, the system alerts the user to fill out the customization form.
2. User clicks on the alert.
3. System takes the user to the form that displays aforementioned questions.
4. User fills out the form and submits it.
5. The system sends a work-out plan suggestion and the filled-out form to the trainer that the user is matched with.

4.1.3 Functional Requirements

REQ-1: The system shall have a customization form that will detect the user's preferred workout type, their personal information, personal goals, workout frequency and duration, and intensity level of the workout.

REQ-2: The system shall allow the users to select which workout equipment they will have access to.

REQ-3: The system shall check the user's answers to see if they are acceptable. It should provide the user with an error message if they are not acceptable or are incomplete.

REQ-4: The system shall save the information obtained and send a copy to the user's trainer.

REQ-5: The system shall allow the user to change their answers at any time and it shall modify their training plans accordingly.

4.2 Personalized Training Plans

4.2.1 Description and Priority

With the customization form each member fills out, we have a better understanding of what our users want and need to achieve their goals. The system and our trainers will provide the user with a training plan uniquely created for them according to their personal preferences, goals, and limitations. This feature is arguably our most important feature and therefore is a high priority for the system.

In terms of benefit, we can say that this feature is a 9 as the most important part of the product as it is the core feature and is necessary. The penalty for not having this feature is also a 9 because not having it would mean that we do not have a product. The design process for this feature will be connected to other features and will include a lot of details, therefore we can classify the cost as another 9. The risk factors, however, are not as present as others so it is a 2 out of 9.

4.2.2 Stimulus/Response Sequences

1. System sends the trainer the customization form filled out by the user with a list of exercise suggestions based on those answers.
2. Trainer reviews the form and creates a training plan.
3. Trainer sends the training plan to the user.
4. User reviews the plan and notifies the trainer about any requests or questions.
5. After certain points are reached in the progress tracking system, system notifies the trainer.
6. Trainer updates the training program to be a more challenging version.

4.2.3 Functional Requirements

REQ-1: The system shall have access to the use's data obtained by the customization form.

REQ-2: The system shall create a list of suggested exercises based on the user's answers. The exercises should include a duration or repetition number.

REQ-3: The system shall give the trainers access to the user's information and the generated exercise list.

REQ-4: The system shall display the user's plan as a checklist, and it shall let the users check off the exercises they have completed.

REQ-5: The system shall not erase exercises marked as complete, rather those exercise shall remain in the list crossed out, allowing the user to take it back if they accidentally marked it as complete.

REQ-6: The system shall allow the trainer to update the user's training plan according to their progress.

4.3 Progress Tracking

4.3.1 Description and Priority

The system provides a progress tracking system for the user that will consist of keeping track of the amount of exercises the user completes in the form of a point system. The user is expected to check off each exercise they have successfully completed, which will in return increase the user's achievement point kept in the system. After certain milestones set by trainers are reached, the trainer is notified, and the user's training plan is updated. This feature is a medium priority in this project since the project would be able to exist and work without it, but it still provides an extra service.

The benefit of this feature is a 6 since it keeps track of what the user has been doing and can help the trainer determine what changes need to be made. The penalty for not having a progress tracking system, however, is a 1 since the program could function just as well without it, although it wouldn't be as motivating to the user. The cost is a relatively lower 5 as it doesn't require an entire extra system for it, but still requires certain changes for the other components like a training program the user can check off. It does involve the risk of the user not checking off their completed exercises and measuring their progress incorrectly because of that, so the risk factor is another 5.

4.3.2 Stimulus/Response Sequences

1. With each exercise they successfully complete, the user checks the exercise to show they have done it.
2. The system increases the user's achievement point for each exercise they complete.
3. After a point milestone is reached, the system notifies the trainer.
4. Trainer reviews information about the user, given feedback, and the user's goals and updates the user's training plan accordingly.

4.3.3 Functional Requirements

REQ-1: The system shall keep an 'achievement point' for each user with the initial value of 0 upon subscription.

REQ-2: The system shall keep track of the exercises the user checks as done.

REQ-3: The system shall increase the achievement point by one with each exercise completed.

REQ-4: The system shall calculate and display users' 'achievement points' on a regular basis.

REQ-5: The system shall notify the user and their trainer once the ‘achievement point’ reaches the milestone set by the trainer.

4.4 Communication System

4.4.1 Description and Priority

The communication system is mainly an e-mail system involved in our website that allows both the trainers and the members to send each other messages through text. Users can use this system to clarify any questions, concerns or requests they may have regarding their training plan and the trainers can use it to provide feedback, answers, or additional advice. It is critical for the trainers and the trainees to communicate, when necessary, therefore this feature is a high priority.

The benefit of the communication feature is a 9 as it allows the members to interact with their programs and be involved in the decision making in their journey. The penalty for not having it is also high considering the system does not provide any other ways of communicating and not having it could lead to dissatisfied members and a significant loss of customers. The implementation and d=the design is relatively simple, so the cost is 4 out 9. Since e-mailing is not considered to be a ‘fast’ way to communicate, it could cause delays in communication. Data privacy breaches are also a concern, so the risk factor is a 7 for this feature.

4.4.2 Stimulus/Response Sequences

This sequence is written with the assumption that the user needs to communicate with the trainer.

1. User clicks the communication section from the web page.
2. System displays the page that has a simple text form for the user to express themself.
3. System sends the user's message to their trainer via e-mail.
4. The trainer receives and responds to the user’s message via the same system.

4.4.3 Functional Requirements

REQ-1: The system shall allow both users and trainers to send each other text-based messages.

REQ-2: The system shall send the message to the person of interest via email.

4.5 External Resource Library

4.5.1 Description and Priority

The system provides an external resource library that offers a variety of resources such as instructional videos that show how exercises are correctly done, workout guides, nutritional advice etc. It is a low priority

since it is not necessary for the system to function, and users could find the resources themselves if they tried to.

This feature allows the members to have a better understanding about what they are doing' why they are doing it, and how it should be done, so the benefit rating for it is an 8 out of 9. Although it is useful to have this feature it is not necessary, so the penalty of not having it would be a 1. The cost is also a 1 since it only consists of creating a library that displays the sources in an organized manner. The risk factor is a 1 since it does not affect the main functionality of the product.

4.5.2 Stimulus/Response Sequences

1. User clicks the library section from the web page.
2. System displays the library of external sources.
3. The user selects any source in which they are interested.

4.5.3 Functional Requirements

REQ-1: The system shall have a collection of external resources that should be categorized.

REQ-2: The system shall redirect the user to the according website of the resource.

REQ-3: The system shall provide an error message if the resource has been removed.

REQ-4: The system should periodically check whether the resources are valid and available.

4.6 Payment Management

4.6.1 Description and Priority

The payment management system lets the members purchase a subscription and pay their fees. This feature is a high priority since it is crucial for the financial component of the system and can cause a lot of problems if not implemented properly.

The benefit of the payment management system is 9 because it ensures that the members are charged correctly according to their subscriptions. Not having this feature would cause huge financial losses and potentially legal trouble, therefore the penalty rate for it is another 9. The cost depends on the integration with other software, but the expected costs are a 6 out of 9. The risk rate is a low 2 considering that safety measures are taken to prevent fraud and incorrect payment amounts.

4.6.2 Stimulus/Response Sequences

1. User clicks the payment section from the web page.
2. User clicks on their preferred subscription.

3. User enters their bank and card information to be billed.

4. System connects with the bank's software to process payment.

4.6.3 Functional Requirements

REQ-1: The system shall allow members to choose a subscription plan.

REQ-2: The system shall allow the users to securely pay their subscription fees.

REQ-3: The system shall ensure the security of user's personal and financial information.

REQ-4: The system shall be integrated with a third-part payment processing provider and should ensure that the payments are being processed accurately and on time.

5. Other Nonfunctional Requirements

5.1 Performance Requirements

- **Concurrent User Handling:** At least 100 people can utilize the product simultaneously thanks to the way it is set up. This requirement is designed to ensure that the system can function properly and handle a considerable amount of traffic during periods of high usage.
- **Data Processing:** For all non-computational duties, the gym management system is built to respond to user actions within 2 seconds. This criterion was established to guarantee a rapid and efficient user experience, which is crucial for the system's efficient operation.
- **Availability and Downtime:** The product is designed to be available 24/7 with a maximum downtime of 1 hour per month. The purpose of this criterion is that the user should always be able to use the system as comfortably as he or she wants.
- **Scalability:** The product is designed to be expandable in case of an increase in the number of members in the future. The purpose of this criterion is that the system should be designed in such a way that it should not give errors and perform smoothly, even if the number of users shows an exceptionally large increase.

5.2 Safety Requirements

- **Data Security:** All user data, including personal data and payment information, is safeguarded by the product against unauthorized access or compromise. It conforms with all applicable laws governing data privacy.
- **Access Control:** Only people with permission to view the data are allowed to use the product's suitable access control procedures. This method of digital access control uses two-factor authentication and passwords.
- **Certification and Compliance:** The product complies with all relevant safety certifications and standards, such as ISO 9001 and OSHA regulations. Additionally, the system complies with any additional safety requirements established by regional or federal regulatory organizations.

5.3 Security Requirements

- **Authentication:** The product requests authentication from the user by requesting a password before allowing access to any features or data.
- **Access control:** The product implements appropriate access controls to restrict access to and functionality of sensitive data to authorized users only. User roles and permissions are predefined to allow different levels of access to different types of users and are designed with the ability to be changed as competence changes.
- **Encryption:** Encryption is used in the product to protect sensitive data during transit and at rest. Encryption has been implemented using industry standard algorithms and methods.
- **Compliance:** The product complies with all relevant external policies and regulations, such as the General Data Protection Regulation (GDPR) and the Health Insurance Portability and Accountability Act (HIPAA), as well as any internal policies and procedures set by the gym.
- **Certifications:** The product meets all applicable security and privacy certifications, such as the Payment Card Industry Data Security Standard (PCI DSS) and the ISO/IEC 27001:2013 Information Security Management System (ISMS) certification.

5.4 Software Quality Attributes

- **Usability:** The product is user-friendly and easy to use, with clear and intuitive navigation. Users are able to easily perform common tasks with minimal effort and without requiring extensive training. The product also has a consistent and visually appealing interface.
- **Reliability:** The product is dependable and consistently available. It is able to handle large volumes of data and users without experiencing performance issues or downtime. The product is able to recover from errors and failures quickly and without data loss.
- **Maintainability:** The product is convenient to maintain and update over time. The codes are well-structured and modular, with clear documentation and comments. The system also has automated tests to help ensure that changes do not introduce new bugs.
- **Security:** The product protects sensitive user data. Access to the system is limited to authorized users, and authentication is required for all transactions. The system also used encryption to protect both transit and resting data.
- **Scalability:** The product is designed to scale to accommodate increasing numbers of users and data without compromising performance. It is designed to handle peak loads and provide a consistent user experience regardless of usage levels. The system will also be able to integrate with other systems as needed to support business requirements.

5.5 Business Rules

- **Role-based access control:** The product allows access and permissions to be assigned based on predefined roles, such as member, trainer, manager, and administrator. Each role has specific permissions and restrictions, and users are only able to access the functions and data they are authorized to.
- **Data privacy:** The product protects the privacy of sensitive user data, such as personal information and payment details. Access to such data is restricted to authorized personnel only.
- **Transaction logging:** The product logs all user actions and transactions, including login attempts, data modifications, and access attempts. The logs will be stored securely and be available for audit and forensic purposes.
- **Backup and recovery:** The product has a backup and recovery mechanism to prevent data loss in the event of system failure or disaster. The backups are stored securely and tested periodically to ensure they can be restored successfully.

6. Other Requirements

1. **Database Requirements:** The gym management system uses a reliable database management system that ensures data integrity, security, and scalability. The database is designed to be able to process large amounts of data and support system data backup and recovery.
2. **Internationalization Requirements:** The product is designed to support multiple languages, currencies, and time zones to accommodate users from different countries.
3. **Legal Requirements:** The product complies with data protection regulations such as the General Data Protection Regulation (GDPR) and other relevant laws in the jurisdiction in which it operates.
4. **Reuse Requirements:** The product was designed with reusability in mind, which means that the code is modular and easy to maintain. The system is able to integrate with other applications, such as accounting software, payroll software, and attendance management systems.
5. **User Interface Requirements:** Users will be able to access all the content of the site as they can enter the site with the membership system, while non-users will be able to view the home page and the membership section of the site.
6. **Maintenance Requirements:** Although there is not a great obligation in the maintenance part, databases and APIs should be checked regularly and maintenance should be provided in this direction.

Appendix A: Glossary

API: A set of tools, a protocol that communicates with and uses other software resources, hardware resources or data resources.

RESTful or GraphQL: Used to describe a set of software architectural styles, each request from a client to a server contains all the information needed to complete the request. Examples of these requests are HTTP methods such as GET, POST, PUT and DELETE.

ISO 9001: The international standard that specifies requirements for a quality management system (QMS). Organizations utilize standards as a means of showcasing their ability to consistently deliver products and services that align with not only customer expectations, but also comply with relevant regulatory mandates.

OSHA standards: The set of guidelines prescribing the approaches that employers are required to employ for ensuring the safety of their employees from potential dangers.

General Data Protection Regulation (GDPR): Europe’s new data privacy and security law.

HIPAA: The Health Insurance Portability and Accountability Act of 1996 is a federal legislation that necessitated the implementation of standard practices at the national level with an aim to safeguard confidential patient health information from being disclosed without the explicit consent or knowledge of the patient.

PCI DSS: The Payment Card Industry Data Security Standard (PCI DSS) represents a widely acknowledged framework of policies and procedures designed to enhance the security of transactions involving credit, debit, and cash cards. The primary objective of the PCI DSS is to safeguard cardholders against the unauthorized access and misuse of their confidential information.

ISMS: Information Security Management System is a structured framework of policies and procedures that govern the systematic management of sensitive data within an organization.

RDMS: Relational Database Management System is a data storage system in which data is stored in rows and columns in tables and has a high data consistency. A relational database can be Deciphered as a database consisting of data organized between various tables.

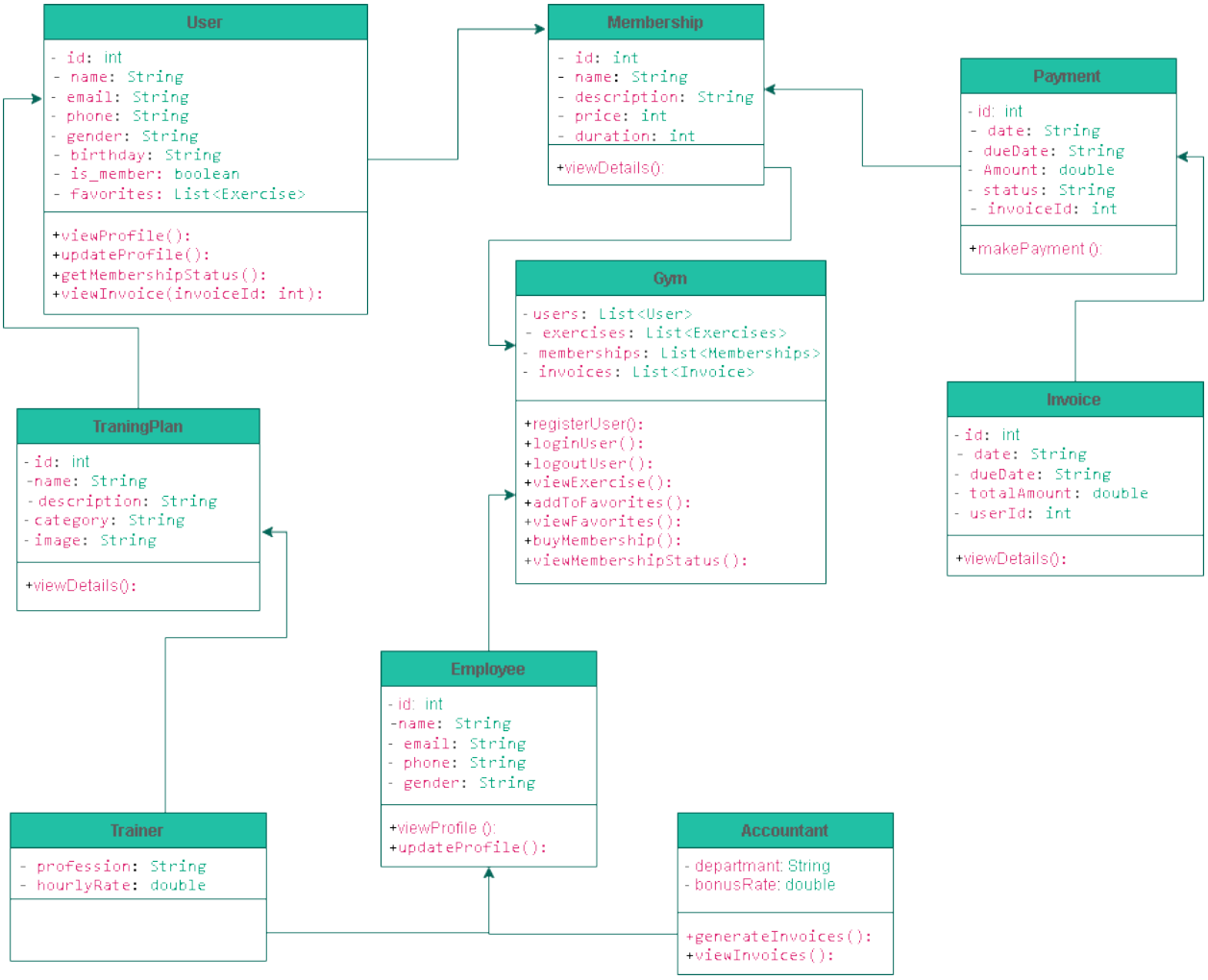
AWS: The Amazon-provided cloud computing platform is a highly inclusive and continuously developing infrastructure.
HTTP/HTTPS: HTTP is an application protocol that defines how data is transmitted between a client (such as a web browser) and a web server. It is used to request and receive web pages and other resources from web servers. HTTPS is a secure version of HTTP that uses encryption to protect the data being transmitted between the client and the server. HTTPS is often used for transmitting sensitive information such as passwords, credit card numbers, and other personal information.

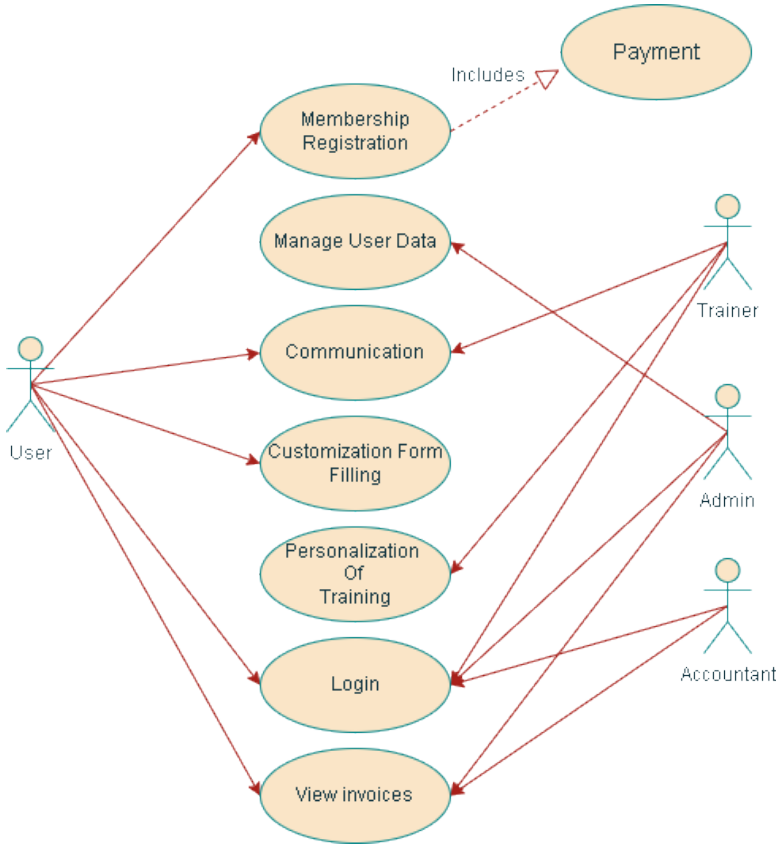
TCP/IP: TCP is responsible for breaking the data into packets and ensuring that each packet is transmitted reliably and in the correct order. It also provides flow control and congestion control mechanisms to prevent the network from becoming overloaded. IP is responsible for routing the packets to their destination across the internet. It does this by assigning a unique IP address to each device on the network and using that address to route the packets to their destination.

SSL/TLS: SSL/TLS protocols work by encrypting the data being transmitted between a client and a server, making it difficult for anyone to intercept and read the data. This is done using a combination of symmetric and asymmetric encryption, and the client and server negotiate which encryption algorithm to use during the initial handshake process.

EMV: EMV (Europay, Mastercard, and Visa) is a global standard for secure payment transactions. The three companies developed it to provide a more secure alternative to traditional magnetic stripe cards. EMV cards contain a small computer chip that stores and processes payment information. When a customer uses an EMV card to make a payment, the chip generates a unique, one-time code that is used to authenticate the transaction. This makes it much more difficult for fraudsters to steal and use cardholder information.

Appendix B: Analysis Models





Use case name	**Login**
Actors	User, Accountant, Admin, Trainer
Preconditions	The actor must have a registered account.
Normal Flow	• User requests to log in. • System presents the login form. • User enters their login information. • System validates the information and grants access. • System notifies the user of successful login.
Alternative Flows or Exceptions	If the verification fails, the system informs the user, and the login process is repeated.
Non-functional requirements	The login process should be easy to understand.

Use case name	**View Invoices**
Actors	User, Accountant, Admin
Preconditions	The actor must be a member of the gym.
Normal Flow	• Actor requests to view invoices. • System generates invoices based on specified membership. • System presents the invoices report to the actor.
Alternative Flows or Exceptions	If the view fails, the system notifies the user, and the process is repeated.
Non-functional requirements	The view invoices process should be easy to understand.

Use case name	**Manage User Data**
Actors	Admin
Preconditions	The actor must be logged in as an Admin.
Normal Flow	• Admin requests to manage user data. • System presents a list of user and management options. • Admin selects a user and chooses to add, edit, or delete user information. • System processes the changes and updates the user data. • System notifies the admin of successful changes.
Alternative Flows or Exceptions	If there's an error processing the changes, the system informs the admin, and the management process is halted.
Non-functional requirements	The management process should be easy to understand.

Use case name	**Payment**
Actors	User
Preconditions	The actor must be logged in. The user must select an invoice.
Normal Flow	• User requests to make a payment. • System presents available payment options. • User selects a payment option and provides the necessary information. • System processes the payment. • System notifies the customer of successful payment.
Alternative Flows or Exceptions	If the payment fails, the system notifies the user, and the payment process is repeated.
Non-functional requirements	The payment process should be easy to understand.

Use case name	**Membership Registration**
Actors	User
Preconditions	The user must not be currently registered, or their current membership must have expired.
Normal Flow	• User requests to register. • System presents a registration form. • User fills out the registration form. • System validates the information and creates a new account. • System notifies the user of successful registration.
Alternative Flows or Exceptions	If the verification fails, the system informs the user, and the registration process is repeated.
Non-functional requirements	The Membership Registration process should be easy to understand.

Use case name	**Communication**
Actors	User, Trainer
Preconditions	The actors must be logged in. The user must have an active membership.
Normal Flow	• User requests to send a message. • System presents a trainer list. • The user selects the trainer and writes a message. • System verifies the user and checks the content of the message. • System notifies the user of successful registration and notifies the trainer.
Alternative Flows or Exceptions	If the communication fails, the system informs the user, and the communication process is repeated.
Non-functional requirements	The communication process should be easy to understand.

Use case name	**Customization Form Filling**
Actors	User
Preconditions	The user must be logged in. The user must have an active membership.
Normal Flow	• User requests to Customization Form. • System presents a form. • The user fills the form. • System verifies the user and checks to see if the questions are blank. • System notifies the user of successful customization.
Alternative Flows or Exceptions	If the Customization Form Filling fails, the system informs the user, and the Customization Form Filling process is repeated.
Non-functional requirements	The Customization Form Filling process should be easy to understand.

Use case name	**Personalization Of Training**
Actors	Trainer
Preconditions	The trainer must be logged in.
Normal Flow	• Trainer requests customization forms filled out by users. • System presents users forms. • The trainer chooses the user. • System verifies the trainer, and the trainer prepares a specific plan and informs the user. • System notifies the user of successful customization.
Alternative Flows or Exceptions	If the personalization of training fails, the system informs the trainer, and the personalization of the training process is repeated.
Non-functional requirements	The personalization of the training process should be easy to understand.

Appendix C: To Be Determined List

1. Adding new membership packages.
2. Creating a chatbot for the users to access trainers more easily.
3. Interpreting the information entered in the form section with machine learning algorithms and transmitting it to the coach rather than interpreting it by the coaches.