CONTENT

1. Introduction

A SoftEng health club that is in the process of clarifying the requirements for a new management system aimed at ensuring a seamless experience for both members and staff. The system will encompass various key components, including membership registration, member check-in, class registration, class cancellation, membership renewal, membership cancellation/application, manager's inactivity report and usage report. Membership registration will involve members providing their personal information, signing a waiver, and receiving a unique **membership ID**. The health club offers three membership options (Green, Purple, and Blue), each differing in terms of costs and duration. Upon entering the club, members will scan their ID cards, allowing staff and club managers to access their information and generate a membership **transcript**. The system will also verify the status of a member's credit card, subscription, and ID. In the event of expiration, the system can automatically charge the credit card on file, or members can renew their subscription on the spot.

Once granted access, a screen will display the available gym classes for the member's selection. If a member decides to cancel their subscription or class, they can do so either in person or through the app. Access to the gym facilities will be permitted until the subscription's expiration date. The system will update the member's profile to confirm the cancellation and trigger a survey to collect feedback regarding the cancellation process.

2. Requirements

1. **Membership Enrollment:**
   Importance: Essential
   On a member's initial visit, they will be asked to complete a form, providing all the necessary personal information as specified in the system's current **membership profile**. This information includes their **first name, last name, contact number, email address, and membership Levels/Plan**. See Requirement 2: Membership Levels for more information.

   1.1. Upon the member's arrival and completion of the initial information form in the system, it will assign a unique **membership ID** to the member. This ID will not duplicate any existing **membership IDs** within the system.
   1.2. he system.
      1.2.1. The member's information will only be entered if they sign the gym's liability waiver.
      1.2.2. If a member is between 16-17 (inclusive), then they must be accompanied by a legal guardian who will sign their waiver.
   1.3. To access the club, the club staff provides new members with a **membership card** that features a barcode for quick and convenient entry scanning.
   1.4. The system's database will then be updated with the addition of new member details and their accompanying **transcript**.

2. **Membership Levels:**
   Importance: Essential
   There are 3 different levels of membership that customers will be able to choose upon registration. These levels are distinguished by the length of membership purchased. All members will be granted full access to the gym as many times as they choose throughout the duration of their membership. As a new member, the **Membership expiration** depends on the levels of membership you choose at the time of registration (as per section 2.2).
   2.1. Memberships will automatically renew unless the credit card information expires or the member cancels their membership.
   2.2. Levels of membership:

      2.2.1. **Green Member**: 6-month membership. Members will be charged $400 every six-month period after starting from the date and time of registration.
      2.2.2. **Purple Member**: 1-year membership (most common). Members will be charged $400 every one-year period after starting from the date and time of registration.
      2.2.3. **Blue Member**: 3-year membership. Members will charge $1200 every three-year period after starting from date and time of registration.

3. Scan-In by Member:

Importance: Essential

The system shall allow users to scan into the gym by scanning their unique membership number through **membership card**/mobile. If the customer's membership has expired, it will allow them to renew on the spot.

3.1. The system shall scan the barcode and validate if the member should be allowed access to the gym.

3.2. Upon scan, the following things will occur:

    3.2.1. The gate will unlock if the customer's membership has not expired. If the membership expires, the employee will be prompted to give the member the option for renewal (see Requirement 4).

    3.2.2. The screen will open the member's **membership profile**.

3.3. The member's **visitation tracker** will then be updated.

4. Membership Renewal:

Importance: Essential

The system is designed to offer users the capability to renew their membership in the event that it has expired. This feature ensures that members can easily extend their membership and continue to enjoy the services and benefits offered by the club.

4.1. A renewal **notification** is sent to club members whose memberships are set to expire within the next month, providing them with a 30-day priority notice.

    4.1.1. The **notification** is sent via **email** and text messages.

4.2. The manager maintains a record of members whose memberships are nearing expiration, and in accordance with the priority notice, the manager reaches out to these members for membership renewal.

5. Membership Cancellation:

Importance: Essential

Members will be able to cancel their subscriptions in person or via the member mobile app. Members will still be allowed access to the gym's facilities until the duration of their current pay period has expired.

5.1. In-person, the member may inform the employee on-staff that they wish to terminate their membership.

    5.1.1. The employee will then open the system and click the "Cancel Membership" button; a confirmation pop-up will then appear to ensure the button was not clicked by mistake.

    5.1.2. Once the cancellation has been confirmed, the member's **membership profile** will then be marked inactive within the system's database, and they will no longer be charged.

    5.1.3. The system will also send an **email** to the member containing a survey that provides insight as to why they decided to cancel their membership. This survey will be created by the company's marketing team.

5.2. The members may also cancel via the mobile app. They may open their profile within the app and will click the "Cancel Membership" button.

6. Class Registration and Cancellation:

Importance: Secondary

There are two different ways that a user will be able to sign up for **classes** provided by the gym. All members have full access to register for classes.

6.1. Members will also be able to register for **classes** on their phone through the SysEng app.

6.1.1. Members will be given access to view all upcoming **classes** and if they wish to sign up for one they will click the register button and will be enrolled.

6.1.2. Members will be notified by the system at the start of the gym's office hours if they are enrolled in a workout class on that day.

6.1.3. The **opening screen** on the member app will display all **classes** that the member is currently enrolled in.

6.1.4. If the member would like to cancel a class, they may click on it where it is displayed on their opening screen. The class's page will be opened, and they may click the button that says "cancel", a pop-up will then appear to ensure that the member wishes to cancel their class.

6.1.4.1. The member may click the "yes" button to cancel the class. They will then be taken off the class's roster and the class will be taken off their opening screen.

6.1.4.2. The member may click the "no" button if they mistakenly clicked the cancel button. They will remain on the class roster.

6.2. Class pages will display the time, instructor and current enrollment number.

6.2.1. If a class is full, it will not allow access for members to enroll. A pop-up will display on the screen that says, "We are sorry, this class time slot is currently full. Would you like to be added to the waitlist?"

6.2.1.1. If the member clicks the button that says "Yes," they will be added to the waitlist. Waitlists act as a queue; as soon as one member cancels, the first to join the waitlist will be enrolled and notified on the member app.

6.2.1.2. If the member clicks "No," they will not be added to the waitlist.

7. Member Activity and Reports:

Importance: Essential

Whenever a club member visits the club, their visit is recorded with the respective time, day, and date in the records. The manager is authorized to request a comprehensive list of all members who have not accessed the club facilities in the preceding 30 days. This report will include the contact information of individual members.

7.1. The manager will reach out to inactive members and make an effort to rekindle their interest in fitness.

7.2. Club staff members have full access to the reports, enabling them to search for members on the list and perform various tasks as needed.

3. Use Case Model and all Use Case Narratives
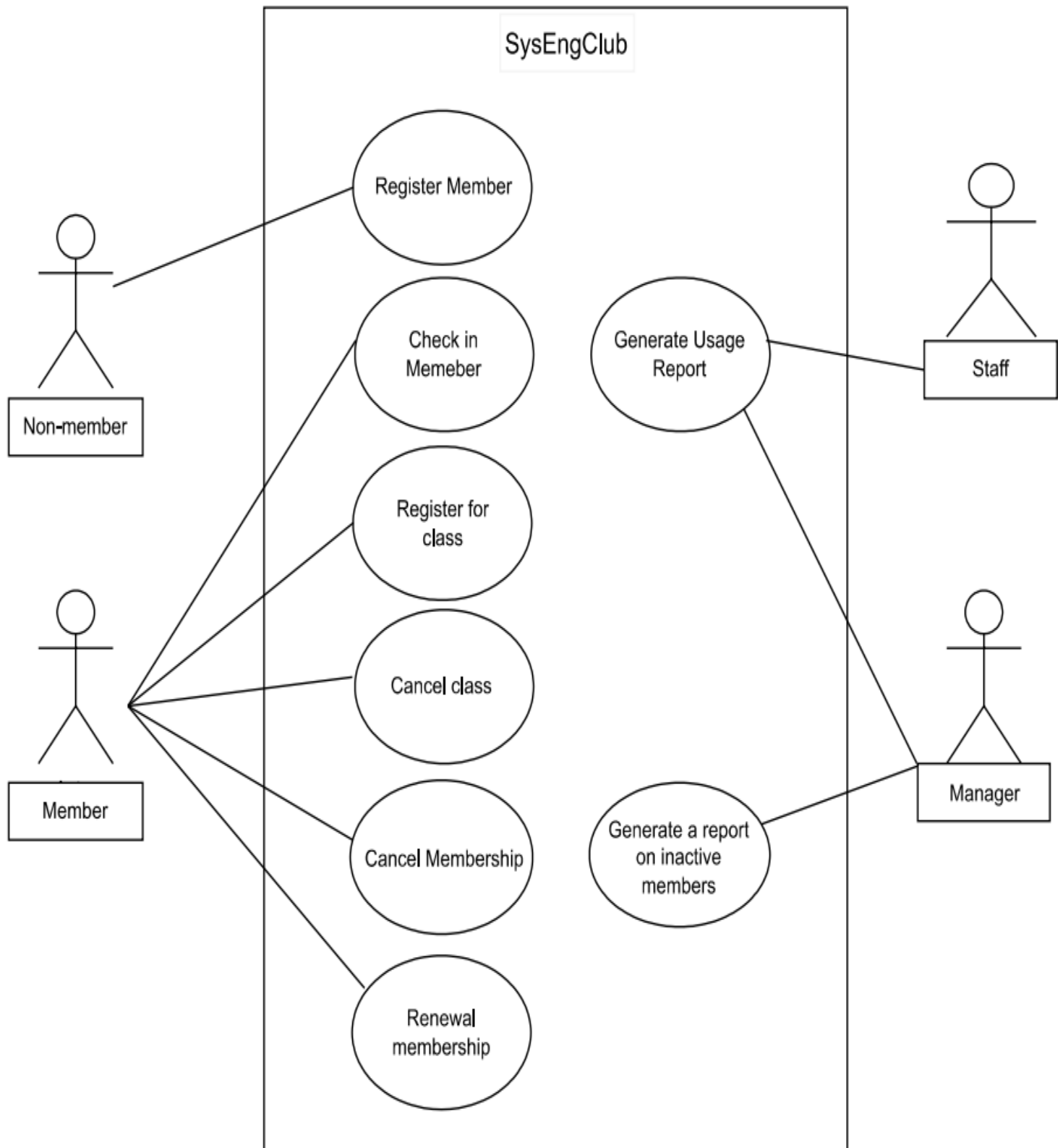
1. Use case: Enroll Member

Actors: Non-Member

Purpose: To facilitate the seamless enrollment of non-members, the system captures their personal information.

Requirements Implemented: Requirement 1: Membership Enrollment

Overview: The new member arrives at the club and expresses their intention to join. The staff provides the new member with a membership application form digitally. The new member completes the form, specifying the desired membership length (ranging from 6 months to 3 years) and makes an upfront payment. The system generates a unique **membership ID** for the new member, records their information, and stores the payment transaction. Once completed, the new member is granted access to the club.

Type: Essential

Preconditions: For successful registration in the system, it is imperative that all the personal details provided by non-member are accurate, and the membership plan and payment information are accurately reflected.

Postconditions: New Member has a valid membership with a unique membership ID.

Special Requirements: System should process payment securely and get a response under 5 seconds.

Flow of Events

Actor Action	System Response
1. The new members express their desire to join the club and seek assistance from the club staff.	6. Upon successful payment, the system generates a unique 8-digit **membership ID** for the new member.
2. The club staff provides the new member with a membership application form, explaining the process.	7. The system records all the provided member information, including their name, contact details, and selected membership duration.
3. The new member completes the membership application form, providing their personal information and selecting the desired membership duration.	8. The system securely stores payment information such as method, time stamp, cost.
4. The new member pays the upfront membership fee using their preferred payment method (e.g., credit card, cash, mobile payment).	9. The system notifies members of successful enrollment and generates a physical card with their unique 8-digit ID number and disables lock on turnstile.
5. New Member submits application with payment.	11. System enrolls member into auto-payment and notifies member regarding auto-payments unless otherwise denied by user.
10. Turnstile unlocked granting member entrance into the facility	12. System also generates member profile on mobile application with scannable QR code that can be used along with physical card.

Alternative Flow of Events

Line (3): If a member fills an application incorrectly, the system will prompt the member to correctly fill the respective sections and resubmit.

Line (4): If the payment fails, the system notifies members of payment failure and prompts for alternative payment methods.

Line (11): If a member opts not to enroll in auto-payment, they will be required to manually pay their dues.

2. **Use case:** Check-in Member
   Actors: Member
   Purpose: Allow club members to check-in when they visit the club's facilities.
   Requirements Implemented: Requirement 3: Scan-In by Member
   Overview: Club members who wish to use the club's facilities are required to check in by showing their membership ID. The system checks the membership's validity and grants access if the membership is active. The system records members' visits, storing information regarding the date, time, and day of each visit.
   Type: Essential
   Preconditions: The club member is physically present at the club and has already been registered.
   Postconditions: The member has either gained access to the club or has been notified that their membership is no longer valid.
   Special Requirements: The system should process check-ins efficiently (under 5 seconds).

Flow of Events

Actor Action	System Response
1. The club member arrives at the club and approaches the check-in area.	3. The system reads the *membership ID*.
2. The club member presents their *membership ID*, such as a physical card or digital QR scan, to the system for verification.	4. If the membership is valid and within its membership duration, the system grants the member access to the club's facilities via disabling lock-on turnstile.
	5. If the membership is not valid (expired or otherwise inactive), the system denies access and notifies the club member.
	6. The system will maintain a record of users who scan-in and those who do not, updating the database with each visit, including the date, day, and time.

Alternative Flow of Events

Line (2): If a Member forgets their ID, they are able to verify identity and membership by a valid government issued ID.

Line (3): If the system is unable to read the member ID, members will be prompted to present ID again.

Line (5): If the membership is not valid due to an expired payment or other issues, the system prompts the club member to renew their membership on the spot. If they choose to renew, they make the necessary payment, and the system updates their membership.

3. **Use case:** Renew Membership
    Actors: Member
    Purpose: To Enable members to seamlessly renew their membership.
    Requirements Implemented: Requirement 4: Membership Renewal
    Overview: Existing club members who wish to continue their membership visit the club to renew.
    The system allows them to renew for the same duration and at the same rate as their prior membership,
    ensuring uninterrupted access to club facilities.
    Type: Essential
    Preconditions: The club member is an existing member with a membership that is about to expire,
    and they have the intention to renew and have not enrolled in auto-payments.
    Postconditions: The member's membership is successfully renewed for the same duration, and their
    access to the club remains active.
    Special Requirements: The system should handle membership renewals efficiently to minimize wait
    times for members.

Flow of Events

Actor Action	System Response
1. The club members arrive at the club with the intention to renew their membership.	4. The system retrieves the member's current membership details, including the expiration date using ID.
2.The club members present their existing ***membership ID*** or provide their identifying information to the club staff.	5. The system calculates the renewal fee based on the same rate and duration as the member's prior membership.
3. The club member makes the payment for the membership renewal.	6. The system updates the membership duration, extending it for the same duration as the prior membership, and records the renewal.
	7. The system issues a confirmation of the renewed membership to the club members and the turnstile will be unlocked allowing members to access facilities.

Alternative Flow of Events

   Line (4): If the system cannot retrieve the members current membership details, the member will be
   prompted to provide other forms of identification such as government issued ID.

   Line (5): If the member chooses to upgrade their membership, the system will recalculate the dues
   accordingly and record the change in membership details.

4. **Use case:** Cancel Membership
   Actors: Member
   Purpose: To allow existing members to seamlessly cancel their membership
   Requirements Implemented: Requirement 5: Membership Cancellation
   Overview: Existing club members who opt to cancel their memberships either visit the club in person or contact the club's staff for cancellation requests. The system guarantees a straightforward and prompt cancellation process. Additionally, members have the option to initiate the membership cancellation through the mobile application.
   Type: Essential
   Preconditions: The club member is an existing member.
   Postconditions: The member's membership is successfully canceled, and their access to the club is terminated.
   Special Requirements: The system should handle membership cancellations efficiently and provide any necessary confirmations or refunds promptly.

Flow of Events

Actor Action	System Response
1. The club member visits the club or contacts club staff to request membership cancellation.	3. The system retrieves the member's current membership details and confirms the request for cancellation.
2. The club members provide their *membership ID* or other identifying information.	4. The system ensures that any outstanding dues or fees are settled and processed.
	5. The system updates the membership status to "canceled," access to the gym's facilities will be officially terminated on the member's membership expiration date.
	6. The system will prompt the user to complete a survey to determine why they chose to terminate.
	7. The system will generate a confirmation email, stating that the membership has been terminated as per the member's request.

Alternative Flow of Events

Line (2): If the club members ID is not available, they will be prompted by the staff to present other IDs such as government issued ID.

Line (4): If there are any outstanding charges, the system prompts members to pay charges before successful cancellation.
Line (5): If there are materials to be returned or refunds to be processed, the club staff ensures that these are handled promptly and to the member's satisfaction.
Line (6): The member will have the option of denying the survey.

5. **Use case:** Generate Report of In-active Members
   Actors: Club Manager
   Purpose: To enable club managers to generate reports showing the gym members who have not used the club in the last 30 days, facilitating efforts to re-engage inactive members.
   Requirements Implemented: Requirement 7: Member Activity and Reports
   Overview: Club managers can access the system to generate a report that lists club members who have not utilized the club's facilities in the last 30 days. This report aids the managers in reaching out to these inactive members and attempting to re-interest them in the club's offerings.
   Type: **Essential** <<always during analysis>>
   Preconditions: The club manager has appropriate access to the system and has chosen to generate the usage report.
   Postconditions: The manager has obtained a report listing inactive members who have not visited the club in the last 30 days.
   Special Requirements: The system should generate usage reports efficiently, allowing for timely outreach to inactive members.

Flow of Events

Actor Action	System Response
1. The club manager logs into the system and navigates to the "Generate Usage Report" function.	3. The system processes the request and generates the report.
2. The manager selects the criteria for generating the report, specifically choosing to list members who have not visited the club in the last 30 days.	4. The system presents the club manager with the report, displaying the names and contact information of members who meet the criteria.

Alternative Flow of Events

Line (2): If the manager wishes to refine the criteria further, such as a longer time frame greater than 30 days, they can make the adjustments before generating the reports.

Line (4): If the system encounters issues with data discrepancies while generating the report, the manager will be alerted and will provide guidance on how to proceed to resolve the issue.

6. **Use case:** Generate Report of Member Visitation
   Actors: Club Manager and Staff
   Purpose: To generate a report of searchable member list that discloses personal information, emergency contact, waiver, form of payment and most frequent club users.
   Requirements Implemented: Requirement 7 Member Activity and Reports
   Overview: Club managers can access the system to generate a report of most frequent gym goers and in case of emergency or issue, they are able to retrieve the information provided.
   Type: Essential
   Preconditions: The manager and staff choose to generate a usage report and have access to the system.
   Postconditions: The manager and staff are granted to obtain a report listing active members and their information provided.
   Special Requirements: The system should generate usage reports effectively, allowing club staff and managers to view member's information.

<center>**Flow of Events**</center>

Actor Action	System Response
1. The club manager logs into the system and navigates to the "Generate Usage Report" function.	3. The system processes the request and generates the report once approved for the manager or staff member.
2. The manager and staff provide the reasoning for generating the report.	4. The system represents the report of the member's name, contact information and a count of the amount of gym visitation.

Alternative Flow of Events

Line(3): If the system recognizes that there was a report requested without an explanation, the report will not be generated.

7. **Use case:** Register for Classes
   Actors: Member
   Purpose: Register a member for classes
   Requirements Implemented: Requirement 5.1.1 - 5.1.3: Class Registration and Cancellation
   Overview: Club members have the option to register for specific classes, such as Zumba, cycling, aerobics, if they wish to attend. This process ensures that members can secure a spot in their chosen classes and enables the club to manage class capacity. Members can participate in these classes at the club, and they also have the flexibility to select and enroll in them through the mobile app. The app will keep track of member's activity.
   Type: Essential
   Preconditions: The club member has an active membership and has selected the class they wish to register for. There must also be space in the class.
   Postconditions: The member's registration for the chosen class is successfully confirmed, and their spot in the class is reserved and for confirmation they will get email/text notification.
   Special Requirements: Members must get a system response within 10 seconds.

Flow of Events

Actor Action	System Response
1. The club member can either log into the mobile application or seek assistance from club staff at the club's physical system.	4. The system presents a list of available classes and their details, including date, time, and instructor.
2. The member navigates to the "Class Registration" feature.	5. The system checks the member's eligibility based on their active membership.
3. The member selects the class they wish to register for.	6. If eligible, the system confirms the member's registration for the selected class and reserves a spot.
	7. The system provides a confirmation message to the member, which may include details about the class and any specific instructions.

Alternative Flow of Events

Line (5): If the system determines that the member's membership is not active. The system will notify the member that they need to renew their membership to register for classes.

Line(6): If a member is not eligible, the system will prompt the member to renew their membership on the spot for which they can renew to not renew.

8. **Use case:** Cancel Classes
Actor: Member
Purpose: Members have the option to request the cancellation of a class for which they have registered
Requirements Implemented: Requirement 6.1.4: Class Registration and Cancellation
Overview: Members can request the cancellation of a class they have previously registered for. It ensures that a Member can easily manage their class schedule from the application.
Type: Essential
Preconditions: Membership is valid and has registered for at least one class
Postconditions: Member cancels a class successfully and is removed from roster.
Special Requirements: System performs cancellation and validation in 6 seconds.

Flow of Events

Actor Action	System Response
1. Member successfully signs in to applications	3. System will update the members class schedule
2. Member locates respective class and chooses to cancel	4. System will send confirmation message to member

Alternative Flow of Events

Line(3): If the cancellation request is not valid and the member is then notified for invalid cancellation.

Line(4): If the system fails to send the confirmation to the member, this failure is logged.

4. UML Class Diagram

SysEngClub

+ memberCheckIn()
+ NewMemberRegistration()

Membership

Member

- membershipID: int
+ firstName: String
+ lastName: String
+ dateOfBirth: Date
+ gender: String
+ mobileNumber: int
+ address: String
+ email: String
+ plan: String
+ membershipPlan: String
+ membershipRegistration: Date
+ membershipExpiration: Date
+ specialClass: String
+ visits: DateAndTime
+ renewalNoticeDate: Date

+ membershipStatus(): String
+ membershipRenewal(): String
+ getNumberOfVisits: int
+ membershipCancellation(): boolean
+ specailClassRegistration(): String
+ specialClassCancellation(): boolean
+ upComingClassShedule(): String
- makePlanPayment(): boolean

Uses

Email/Phone
Notification

Email/Phone
Notification

Report

+ reportTitle: String
+ reportDate: Date
+ reportAuthor: String

+ mostFrequentClubUsers(): Report
+ memberList(): Report
+ inactiveMembers(): Report
+ generateMemberRenewal(): Report

Access

ClubStaff

Access

ClubManager

+ inActiveLastThirtyDays(): Report

RenewalNotices

+ MonthlyMembershipRenewalAlerts()

planPayment

Payments

- transactionID: int
- paymentDate: date
- paymentAmount: long
- paymentMethod: String
- invoiceNumber: int
- typeOfPayment: String
- paymentDescription: String
- billingAddress: String

- processPayment(): int
- generateInvoice(): String
- viewPaymentHistory(): String
- refundPayment(): bool
- generatePaymentReport(): String
- paymentStatus(): boolean

5. Software Quality Report

Group Performance Metrics

Metrics	Documentation	Notes
Attendance	**Our meetings up until the final product were the following:** Sept 8th Report, 11: 30 - 12:18PM Sept 15 Report, 11: 34 -12: 16PM Sept 22 Report, 11:35 - 12:15PM Sept 29 Report, 11:33 - 12:20PM October 6 Report, 10:30 - 12:40PM October 13 Report, 10:30 - 12:12 PM October 20 Report, 10:30- 12:15PM October 27 Report, 10:32 - 12:19 PM Nov 3 Report, 10:30 - 12:20PM Nov 10 Report, 10:30 - 12:14PM Nov 17 Report, 10:30 - 12: 17PM Dec 1 Report, 10:30 - 12:08PM Dec 6 Report 6:30pm - 7:00PM Dec 8 Report 10:30 - 12:30PM	**Meeting Notes:** - Meetings occurred every Friday at 11:30 am. However, as a team we later agreed to 10:30 am as it was effective to get a majority of deliverables done in a timely manner. - Approximately 80-minute meetings - Bugz, Shahrukh, Muhammad, and Areej were punctual for all meetings. - Prior work was done beforehand to address questions and concerns in person.
Deliverable Reviews	**The amount of reviews we requested for deliverables up until final product:** Professor: 5 (Online + In-Person)	- As a team, we reached out to our professor to ask how to go about certain components of the deliverable to ensure that we were on the right track. It is essential that developers request for help with the tasks they are working on to create a successful product that can only be achieved with communication.
Progress	**The # of pages and documentation our group concluded for each deliverable:**	- The pages increased with each deliverable.

	T1: 1 Page + Group Notes T3: 17 Pages + Group Notes T4: 18 Pages + Group Notes	- Our team added as much support and documentation for each deliverable to cover all content to the best of our ability. - The group notes at the end of the meeting helped us reflect on the progress we made and the remainder of the work that needed to be done.

Code Metrics

Metric	Results	Notes

Project Team	Size: 4	Development Team:
		  1. Muhammed Khan
		  2. Shahrukh Syed
		
		Code Inspections and Tracking Team:
		  1. Bugz Bhadare-Valente
		  2. Areej Imran

Development Time	Total Time: 30hrs	Individual Development time:
		  1. Muhammed Khan : 16hrs
		  2. Shahrukh Syed : 14hrs

Project Developed On	Development and Inspection	Development:
		  1. Java 17
		  2. GUI: JavaFX
		  3. Data Store: CSV
		  4. Building tool: Maven
		  5. IDE: IntellIj
		Testing:
		  1. JUnit-5
		  2. Mockito
		Inspections and Review
		  1. Fagan Inspections.

Inspections and reviews (Fagan Inspection)	Quality	Defect Density:
		  - Total defects: 23
		  - size: 787

		- Defect Density: $23/787 = 0.029$ per line of code. Defect Removal Efficiency: - Total defects before: 23 - Total defect after: 0 - Defect removal Efficiency(%): (Defect before/(Defect before + after)) * 100 = 100% Inspection Effort: - Preparation Effort: 5 hrs - Meeting Effort: 2hrs - Rework Effort: 7 hrs Total Inspection Effort = Preparation + meeting + rework = 14 hrs. Defect Types: 1. Data update Issue 2. Conditional Logic 3. Exception Handling 4. Concurrency Issues 5. Confirmation Process 6. File Path or Location 7. File Not Closed Properly 8. Unique Constraint Violation 9. Duplicate Detection Logic 10. Date of Birth Storage 11. Age Calculation Logic 12. Clean and Rebuild issue 13. Check Validation Logic
Number of bugs found	Total: 23	Some of the bugs that were found: 1) The membership type was not being printed. No explanation of membership packages 2) Visitor counter did not increment 3) Status did not change to canceled when member canceled 4) CSV was not being written 5) Duplicate phone numbers upon registration cause crashes in the system 6) new Member Id is not generated by the system.

|		7) Path of CSV is not reading
		8) Date of birth Format is not considered by the system and from that getting age calculation is not happening.
		9) Maven unable to read the FXML file.
		10) Member Cancellation is not getting updated.
		11) Already existing members are not validating at the time of registration.
Lines of code	Total: 787	Per Class Breakdown: 1.) Main: 47 2.) Constants: 28 3.) Controller: 342 4.) MemberService: 370
The number of use cases	Total: 4	Use Case Implemented: 1.) Member Check In 2.) Member Cancellation 3.) Member Enrollment 4.) Usage report
Number of Java classes:	Total: 4	The classes that are in the program: 1) Controller.java 2) MemberService.java 3) Constants.java 4) ApplicationMain.java
Number of GUI files	Total: 4	JavaFX-FXML that are in the program 1) SysEng.fxml 2) SysEngMemberRegistration.fxml 3) SysEngMemberCheckIn.fxml 4) SysEngMemberCancellation .fxml
Report (CSV)	Total: 1	CSV file for data storage, update, and validation 1) memberData.csv
Parameter in CSV	Total: 10	Parameter considered in SysEng report. 1) Member ID 2) First Name 3) Last Name 4) DOB(YYY-MM-dd)

		5) Age 6) Email 7) Mobile Number 8) Membership Level 9) Membership Status 10) Visits
Code Improvement	Total: 5	Improvements: 1.) Verification of existing members, with mobile number. 2.) Full name and last name should have at-least 3 characters. 3.) all fields are mandatory at the time of registration. 4.) At the time of registration, the system will validate the input data against the specified requirements. If the data does not meet the requirements, a specific warning message for that field will be displayed on the screen itself. 5.) Ensuring an individual is 18 at the time of registration.
Test Classes	Total: 2	Test classes in the program 1) MemberServiceTest 2) ControllerTest
Unit Testing in MemberServiceTest Class	Total: 12	JUnit Testing methods: 1) registerMemberTest() 2) cancelMembershipTest() 3) cancelMembershipAlreadyCancelledTest() 4) checkInMemberIDTest() 5) checkInMemberIDAlreadyCancelledTest() 6) isFormValidTest() 7) isFormValidFristNameTest() 8) isFormValidLastNameTest() 9) isFormValidDobTest() 10) isFormValidDobAgeTest() 11) isFormValidEmailIdTest() 12) isFormValidMobileNumTest()
Unit Testing in Controller	Total: 2	JUnit Testing methods: 1) submitCheckInTest() 2) processMessageTest()

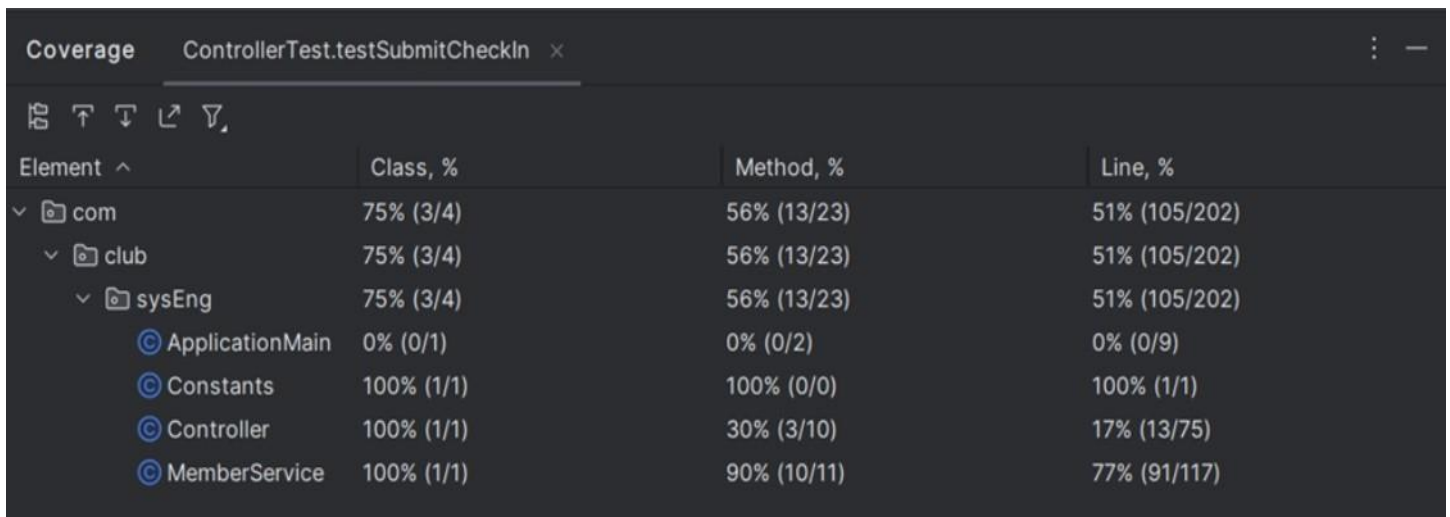| Code Quality percentage	Total: 2	Class test cases covered line by line 1) MemberService.java : 93 % 2) Controller.java : 17%

Note : Code quality percentage never be 100%.

Code Coverage from IntellIj

Unit Testing: Successfully tested.
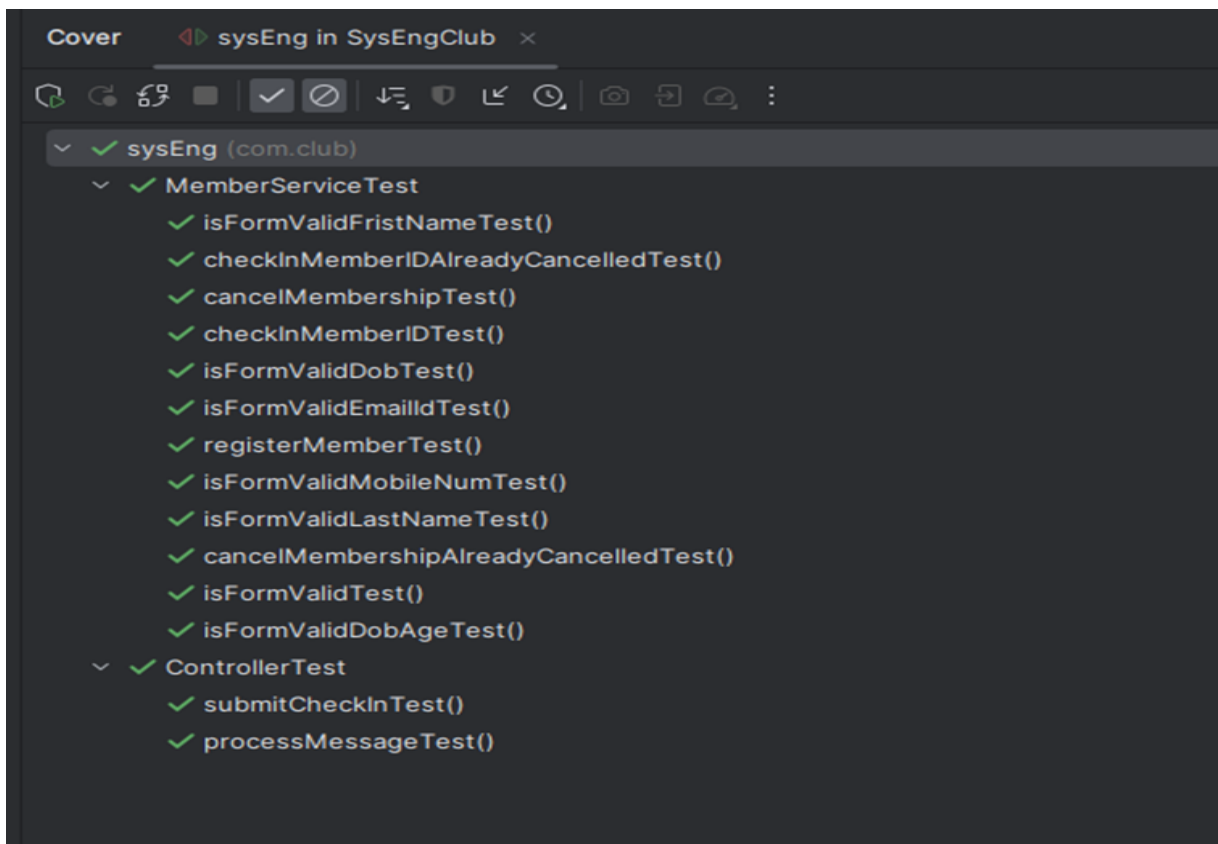
Code coverage Reports.
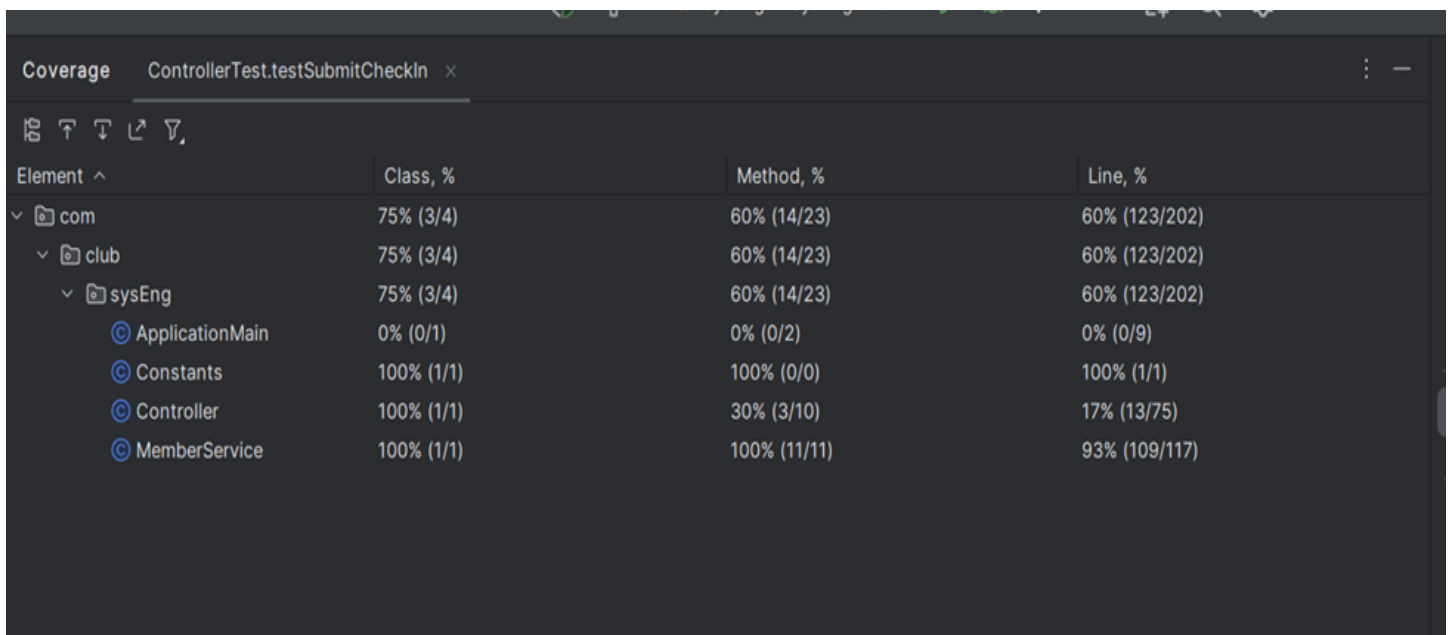
Before: code coverage for MemberService 77%

Element ^	Class, %	Method, %	Line, %
∨ ▢ com	75% (3/4)	56% (13/23)	51% (105/202)
∨ ▢ club	75% (3/4)	56% (13/23)	51% (105/202)
∨ ▢ sysEng	75% (3/4)	56% (13/23)	51% (105/202)
©ApplicationMain	0% (0/1)	0% (0/2)	0% (0/9)
©Constants	100% (1/1)	100% (0/0)	100% (1/1)
©Controller	100% (1/1)	30% (3/10)	17% (13/75)
©MemberService	100% (1/1)	90% (10/11)	77% (91/117)

After: Full test cases for MemberService, code coverage increased to 93%

Element ^	Class, %	Method, %	Line, %
∨ ▢ com	75% (3/4)	60% (14/23)	60% (123/202)
∨ ▢ club	75% (3/4)	60% (14/23)	60% (123/202)
∨ ▢ sysEng	75% (3/4)	60% (14/23)	60% (123/202)
©ApplicationMain	0% (0/1)	0% (0/2)	0% (0/9)
©Constants	100% (1/1)	100% (0/0)	100% (1/1)
©Controller	100% (1/1)	30% (3/10)	17% (13/75)
©MemberService	100% (1/1)	100% (11/11)	93% (109/117)

Requirements Traceability Metrics

* * N/A given as a metric signifies that is not a code-based requirement **

Requirement	Traceability (the percentage that the requirements are covered in final product and implement testing + use case narratives)	Notes
1. **Membership Enrollment**	100%	Method: registerMember 1.1, 1.3 : 100% completion 1.1.1, 1.1.2, 1.2 : N/A
2. **Membership Levels**	100%	Method: membershipPlan Information regarding the details of each membership level are displayed to the user upon registration
3. **Scan-In by Member**	25%	Method: checkInMemberId 3.3 : 100% completion 3.1, 3.2.1, 3.2.2 : 0% completion 3.2 : N/A
4. **Membership Renewal**	0%	4.1 - 4.2 : 0% completion
5. **Membership Cancellation**	75%	Method: cancelMembership 5.1.1, 5.1.2, 5.2 : 100% completion 5.1.3 : 0% completion 5.1 : N/A
6. **Class Registration and Cancellation**	0%	6.1 - 6.2.1.2 : 0% completion
7. **Member Activity and Reports**	100%	Method: MEMBER_DATA_FILE 7.2 : 100% completion 7.1 : N/A

6. Link to Code Repository

[syedShahrukh96/SysEng-Health-Club-Project: SysEng Health Club Project on JavaFx (github.com)](https://github.com)

7. Glossary

a. **Registration:** When a non-member or customer joins the club, they are required to fill out a digital form. Some of the mandatory fields in this form include:

      i. Full Name*
      ii. Date of birth*, (System automatically calculate current age)
      iii. E-mail address*
      iv. Address (address line 1*, Address line 2, City*, State*, Postal code*, Country*)
      v. Contact number*. (10-digit number)
      vi. Emergency contact details*
      vii. Height, weight, any diet restrictions

First Name: should have at least 3 character.
Last Name: should have at least 3 character.
Date of Birth format is YYY-MM-DD

Note: - ' * ' : is a mandatory field that members have to fill.

- **If they miss** filling any of the mandatory requirements, a message will appear on screen, instructing them to complete all mandatory fields before submitting.

- When they sign up, a pop-up message appears on the screen, which reads, "Welcome to the SysEng Club."

b. **E-Mail:** It should have @, letters, "." at the time of registration.

c. **Membership Levels (Green, Purple, Blue):** 3 different levels of membership that customers will be able to choose upon registration. Basic plan is Green, then Purple and last one is Blue

d. **Scan-In Membership Card:** This is an automatic turnstile that unlocks the gate once the member's membership card has successfully scanned into the system.

e. **Membership card**: A card that holds the member's photo, membership ID number, date of initial registration, and a unique barcode.

f. **Notification** : Members with expiring memberships receive 30-day renewal notifications via text and email.

g. **Membership Expiration:** The date and time in which the member's membership expires. It is dependent on the membership level, date and time of registration.

h. **Membership ID**: A unique series of 8 numbers that represents each individual. Ex. 97624187.

i.   **Membership Profile:** Collection of member's personal information, including their photo, membership ID, registration date, account charges, membership expiration, visit history, credit card information, billing address, email, and phone number, stored on the system's database.

j.   **SysEng Club application**: When members sign up with a membership ID, a pop-up message appears on the screen, which reads "Welcome to the SoftEng Club. Please sign in to explore our best membership plan options", Simultaneously, they also receive an email containing a link for account verification.

   i.   After completing the account verification process via email, they will receive another email containing information about the membership plans, which will allow them to sign in easily.

k.   **Opening Screen**: This is the screen the member is met with when they open their SysEng application. It displays their upcoming class schedule.

l.   **Transcript**: This searchable member list from the system's database that holds personal information of the member. The report generated includes the title, date of report and the employee name.

m.   **Visitation Tracker**: This is an int variable within the member's transcript that tracks how many times they have accessed the gym's facilities.

n.   **Classes**: In the SysEng Club, there are a variety of classes available, with the option to register and cancel, the classes are the following:

Zumba   Cycling   Aerobics

High-Intensity Interval Training (HIIT)   CrossFit   Kickboxing

Boxing   Basketball   Swimming