Requirements Specification
Gym Management System

Prepared by
Alpha Software Group

Team Member:

Feng (Frank) Sun 208150484

Kim Fung Lai   206773543

Chao Jiang 206634053

Rahul Kamtam   211838422

Tables of Contents

1 Problem Statement

1.1 The current situation

A large percentage of students at York University spend leisure times in working out at Tait Mckenzie Gym Facility. Though the heavy use by students over the facilities in the gym, the current system to tackle user management, membership management, facilities usage management, equipment rental management, in-store sales management, report generation, and order/payment processing is obsolete and unsophisticated. Most of these tasks, if not all, are handled manually either by clerks, administrators, or managers. This induces an unacceptable amount of mistakes when the tasks are processed manually. For instance, if a user wants to reserve a squash court, he or she needs to call in and make a reservation through the phone. However, most of the time users cannot reach a representative because of unavailability or large volume of calls. Even if a user can reach a representative, he or she may not be able to address the time slot where the use want to be allocated in because there is currently no real-time update about reservations done by the other representatives. This has been embarrassing a lot of users who want to make reservations for gym facilities. On the other hand, the current system does not provide the ease for users to register, login, and manage profile through the use of online system. Users do not have an online platform to achieve these tasks. Users do not have access to upgrade or downgrade their membership and make reservation or rent equipments such as squash racquets online in real time. In-store management and payment processing are not properly done through the use of an online system. And moreover, reports are not generated automatically to keep managers or decision makers up-to-date about statuses of membership and equipment sales and facilities reservations.

1.2 The functionality the new system should support

In order to address the issues discussed above in the current system, a 7/24 online system will be implemented to achieve the desired tasks in a timely and reliable manner.

The objectives of The Gym Management System project are to:

- Provide an online platform for users to register, login, and mange profile.
- Provide a framework for users to upgrade membership.
- Provide a calendar for users to make reservations on facilities.
- provide a system to manage equipments rental and its inventory
- provide a system to automatically generate financial and sales reports in a timely manner
- Provide a system to process payments.

1.3 The environment in which the system will be deployed:

The system will be deployed in Unix Red Hat environment using Web logic Server Administration Console, EJBs, and Oracle databases as the backend. The system should run on all browsers that support cookies, JavaScript, and Java Applets. It should also run on both Windows operating system as well as all Unix-based operating system. (MacOS X, Linux, Solaris)

1.4 Deliverables expected by the client:

The expected deliverables are:

- System analysis documentation
- System design documentation
- System implementation documentation

1.5 Delivery dates

Expected delivery dates are as follows:

- System analysis documentation is due on October 19, 2011.
- System design documentation is due on November 2, 2011.
- System implementation documentation is due on November 30, 2011.

1.6 A set of acceptance criteria

The system to-be must provide all functionalities described above.

2 Requirements Specification

2.1 Functional Requirements

2.1.1 Register

1. Users can enter unique login user name (required field).
2. Users can choose a password (alpha characters and numeric characters, required field).
3. Users can enter an email address (required field).
4. Users can enter personal information comprised of first name, last name, address, age, interests, activities, weight, and height.
5. System verifies all required text fields have been filled out
   a. System can deny registration process if not all required fields have been filled out and ask for completed required fields before moving forward.
6. System sends out account confirmation email.
7. System activates user account.

2.1.2 Login

1. Users can enter login user name. (required field)
2. User can enter password. (required field)
   a. System denies incorrect login password and return error messages.
   b. Users are allowed to retry input password. (attempts are no more than 10 times)
3. System verifies correct login user name and password.
4. System grants user specific access according to their profiles.

2.1.3 Update User Profile

1. Users can update user profile comprised of first name, last name, address, age, interests, activities, weight, and height after successfully login.
2. Users can save any updates against their profiles.
3. System verifies and saves update.

2.1.4 Purchase membership

1. Users should be allowed to purchase membership through the use of the system to-be.
   a. Users will be asked to make payment.

2.1.5 Reserve facilities & reserve equipments

1. Users are able to book facilities on the system.
    a. An error message will be returned to User if the facility has been already booked.
2. Users are able to reserve equipments on the system.
    a. An error message will be returned to User if the equipment has been already booked.
3. System verifies reservation.

2.1.6 Manage booking schedules & rental equipments

1. Clients should be able to manage booking schedules and rental equipments if needed.

2.1.7 Manage Inventory

1. Clients should be able to manage inventory.
2. System should send out notifications if inventory is too low.
3. System verifies it.

2.1.8 Manage Equipment Catalog

1. Clients are able to manage and update catalogs.
2. System verifies it.

2.1.9 Generate report

1. System should generate facilities usage report and customers' satisfaction report in a regular basis.
2. System should send these reports to relevant stakeholders.

2.1.10 Process order and payment

1. System should be able to process payment made through debit card, credit card, and PayPal account.
2. System should deny payment if payment info (e.g. credit card number, expiry date) is invalid.
3. System should verify payment.
4. System should print out receipts or send out confirmation to customer personal email account.

2.2 Non-Functional Requirements

2.2.1 Usability Requirements

1. Anonymous users should be able to access facilities booking schedules without prior registration. However, they are not allowed to book a facility before they register.
2. When booking a facility, a calendar should pop up and let users choose from the calendar.
3. System should work on cross-browser. (e.g. Chrome, IE)
4. Highlighted error messages should be returned when payment or login is denied.
5. Highlighted error messages should be returned when reservations not available.
6. Required fields should be highlighted.
7. System should adopt fonts that are cross-browser. (e.g. Serif, Sans-Serif)

2.2.2 Performance Requirements

1. The calendar should display in less than 3 seconds when users click to reserve facilities.
2. The product catalog should display in less than 3 seconds when users purchase.
3. System should be able to process 25 reservations in a second.
4. System should be able to process 25 orders / payments in a second.
5. System should be able to generate at least 5 reports in a second.
6. System should be able to support up to 500 concurrent users simultaneously.
7. CPU utilization should not exceed 75%.
8. System must be scalable to the growth of up to 1000 users.

2.2.3 Reliability Requirements

1. Data must be SSL encrypted. .
2. Payments must be processed using S-HTTP.
3. Data must be backed up in a regular basis. (e.g. weekly, biweekly, monthly)
4. System should prompt user to restart browser or re-login in case of timeout.
5. Reservation and payment confirmations must be delivery guaranteed.
6. Backward compatibility must be enabled in case deployment goes wrong.
7. Error messages should be returned if login credentials or payment info are invalid.
8. System administrator should restart servers in the cluster when system is down.
9. Maximum downtime is 1%, and therefore, 7 hours and 18 minutes per month.

10. System should run 7/24.
11. Dual Password and Secret Questions should be prompted when login to prevent malicious attacks.

2.2.4 Packaging (how is packaged and installed?)

1. Deployment team should install this application in Unix Red Hat environment using Web logic Server Administration Console.
2. Depending on the possibility of new requirements and defects discovered, 3 - 10 installations are expected.
3. Deployments should take place at midnight from 3am - 5am where the minimal affect on customers is guaranteed.

2.2.5 Legal (if any)

1. Customer information, credit card information, payment method and history, and buying habits should be protected and never exposed to any third parties according to Ontario laws.
2. According to The Payment Card Industry Data Security Standard, SSL and S-HTTP should be used together to prevent from frauds and thefts of critical data.
3. Return policy should be clearly stated and instructed when processing order.
4. The Personal Information Protection and Electronic Documents Act - Customer Information must be protected.

2.2.6 Maintainability and Portability

1. System administrators should be able to make change on schedules and add new items on the catalog. This is usually done at midnights from 3am - 5am where minimal impacts on customers are guaranteed.
2. The foreseen extensions to the system are very high. As we will be expanding the boundary of our system, new Java Classes are expected.
3. System must be cross-browser.
4. Sever should be portable from Red Hat to other environments such as HP UX or any other Unix-based machines.
5. System should be able to run on any operating system. (e.g. Mac OS, Windows, Linux)

2.2.7 Implementation

1. System should run on any browsers (e.g. IE, Firefox, and Chrome) which support JavaScript, Java Applet, and cookies.
2. System should run on Windows and any Linux-based Operating systems. (e.g. Mac OS, Ubuntu)
3. Oracle 11G will be used.
4. Presentation logic will be written in JSP.

2.2.8 Cultural and Political Requirements

1. System should be bilingual (English and French)
2. System will not discriminate on grounds of sex, race, age, level of disability, and type of disability.
3. System should also market on persons with disability. (e.g. blind)

3 Functional Models

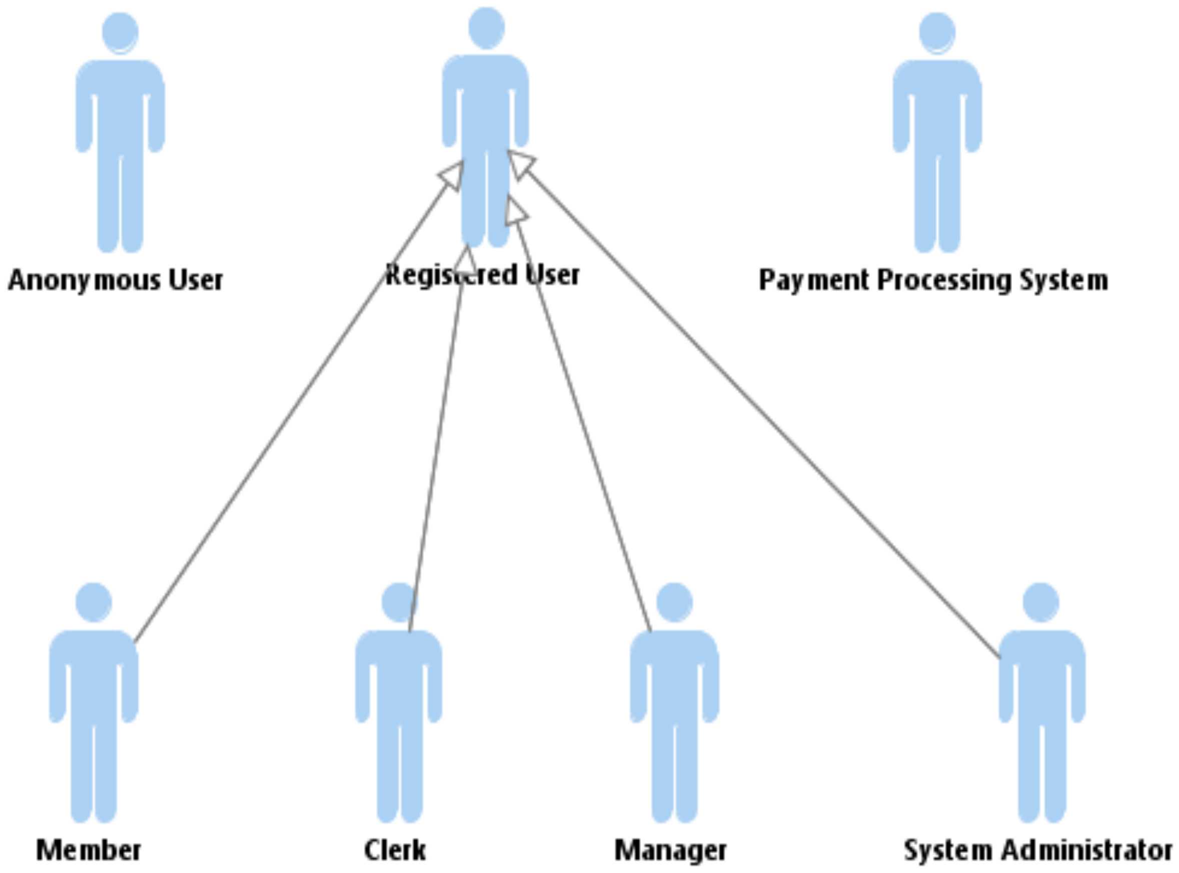
3.1 Actors

3.1.1 Overview

The actors of the system are:

- Anonymous User
- Registered User
- Member
- Clerk
- Manager
- System Administrator
- Payment Processing System

3.1.2 Actor Diagram



3.1.3 Actor Definitions

Anonymous User

Description	A non-registered user who only has the privilege to browse the public area such as schedules.
Aliases	Visitor, Guest
Inherits	*None*
Actor Type	Active – Person
Contact Person	Frank
Contact Details	416-123-4567

Registered User

Description	A registered user that has basic privileges such as book the facility and reserve the equipment. Registered users will make up the majority of the system.
Aliases	User
Inherits	*None*
Actor Type	Active - Person
Contact Person	Frank
Contact Details	416-123-4567

Member

Description	A registered user that has upgraded privileges such as higher priority in booking the facility.
Aliases	Paid User
Inherits	Registered User
Actor Type	Active - Person
Contact Person	Frank
Contact Details	416-123-4567

Clerk

Description	A registered user with the privileges to manage the facility, equipment, users and process the payment.
Aliases	Operator
Inherits	Registered User
Actor Type	Active - Person
Contact Person	Frank
Contact Details	416-123-4567

Manager

Description	A registered user with the privileges and access to the facility management, equipment catalogs management, user management.
Aliases	Operator
Inherits	Registered User
Actor Type	Active - Person
Contact Person	Frank
Contact Details	416-123-4567

Administrator

Description	Top-level user with the full privileges to access the whole system.
Aliases	Admin
Inherits	Registered User
Actor Type	Active - Person
Contact Person	Frank
Contact Details	416-123-4567

Payment Processing System

Description	An external system processes all the online transactions through Credit Card, PayPal, and Debit.
Aliases	None
Inherits	None
Actor Type	Passive - External
Contact Person	Frank
Contact Details	416-123-4567

3.2 Use Case Descriptions

This section documents the complete business use cases within the scope of this project.

3.2.1 Upgrade to member

Description:

The registered user can upgrade to be a member and gain privileges for members.

Actors:

Registered user

Preconditions:

1. User must have a valid account

2. The user name must provide valid user name.

3. User must provide a valid password

Use case Text:

Actor actions	System responses	Payment System
1. select upgrade to member command 3. select membership package 5a. select payment method option 5b. fill in the payment information 5c. submit the payment information	2. display a membership form 4a. display the invoice 4b. display the payment form 6. check the payment 8a. if process fails, prompt the user to re-input the payment information 8b. if process completes, upgrade the user group. 8c. send the receipt to the user.	7.process the transaction

Alternative Courses: None

Extends: None

User Interfaces: upgrade Membership Form

Constraints: None

Questions: None.

Notes: None.

Authors: Frank Sun

3.2.2 Reserve facility

Description: This use case enables the registered users to book the facility.

Actors: Registered user

Preconditions:

1. User must be logged in the system

2. User must not have outstanding balance.

Use case Text:

Actor actions	System responses	Payment System
1. select book the facility command	2. display the facility booking form	
3a. select the facility type		
3b. select the facility item		
3c. select the period		
	4a. check the facility availability	
	4b. if the check fails, prompt the use to reselect.	
	4c. if the check completes, display the invoice	
	4d. display the payment form	
5a. select payment method		
5b. fill in the payment information		
5c. submit the payment information		
	6. check the payment	
		7. process the transaction
	8a. if process fails, prompt the user to re-input the payment information	
	8b. if process completes, upgrade the user group.	
	8c. send the receipt to the user.	

Alternative Courses: None

Extends: None

User Interfaces: Reserve Facility Form.

Constraints:

1. The facility can only be reserved between 8:00am and 11:00pm.
2. Only member can reserve the reserved facility in busy hours between 6:00pm and 9:00pm.

Questions: None

Notes: None.

Authors: Frank Sun

3.2.3 Update the inventory

Description: The clerk upgrades the inventory

Actors: Clerk

Preconditions:

1. User must be logged in.
2. User must be in employee group

Use case Text:

Actor actions	System responses
1. select update inventory command	
	2. display the catalog
3. select the equipment type	
	4. display all items in the selected type
5. select the item	
	6. display the detailed item form
7a. fill in the quantity	
7b. submit the form	
	8. display the updated item

Alternative Courses: None

Extends: None

User Interfaces: Update inventory form

Constraints:

1. The quantity field must be integer.

2. The quantity field cannot be negative.

Questions: None.

Notes: None.

Authors: Frank Sun

3.2.4 Generate the report

Description:

Manager can read the report.

Actors:

Manager

Preconditions:

1. User must be logged in.
2. User must be in manager group or administrator group.

Use case Text:

Actor actions	**System responses**
1. select view report command	2. display the report type
3a. select the report type	
3b. select the period	
	4a. retrieve the selected report data

Alternative Courses: None

Extends: None

User Interfaces: Report Form.

Constraints: None.

Questions: None.

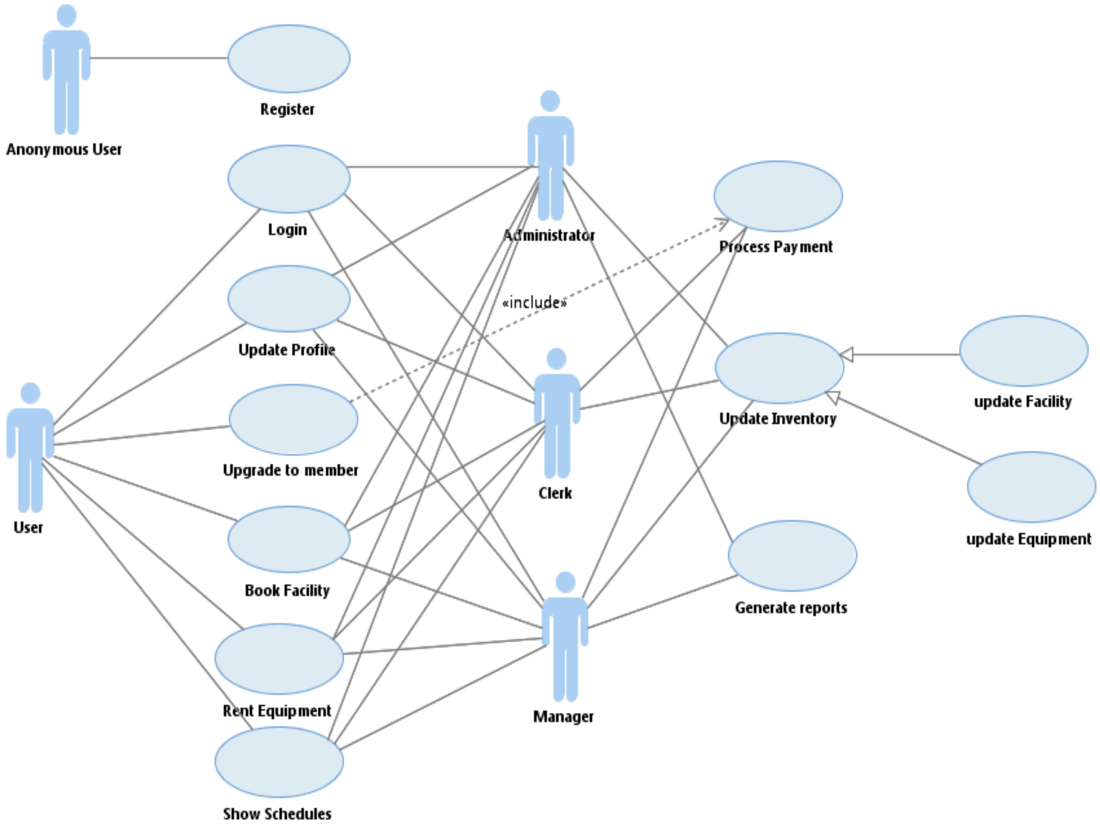
Notes: None.

Authors: Frank Sun

3.3 Use Case Diagrams

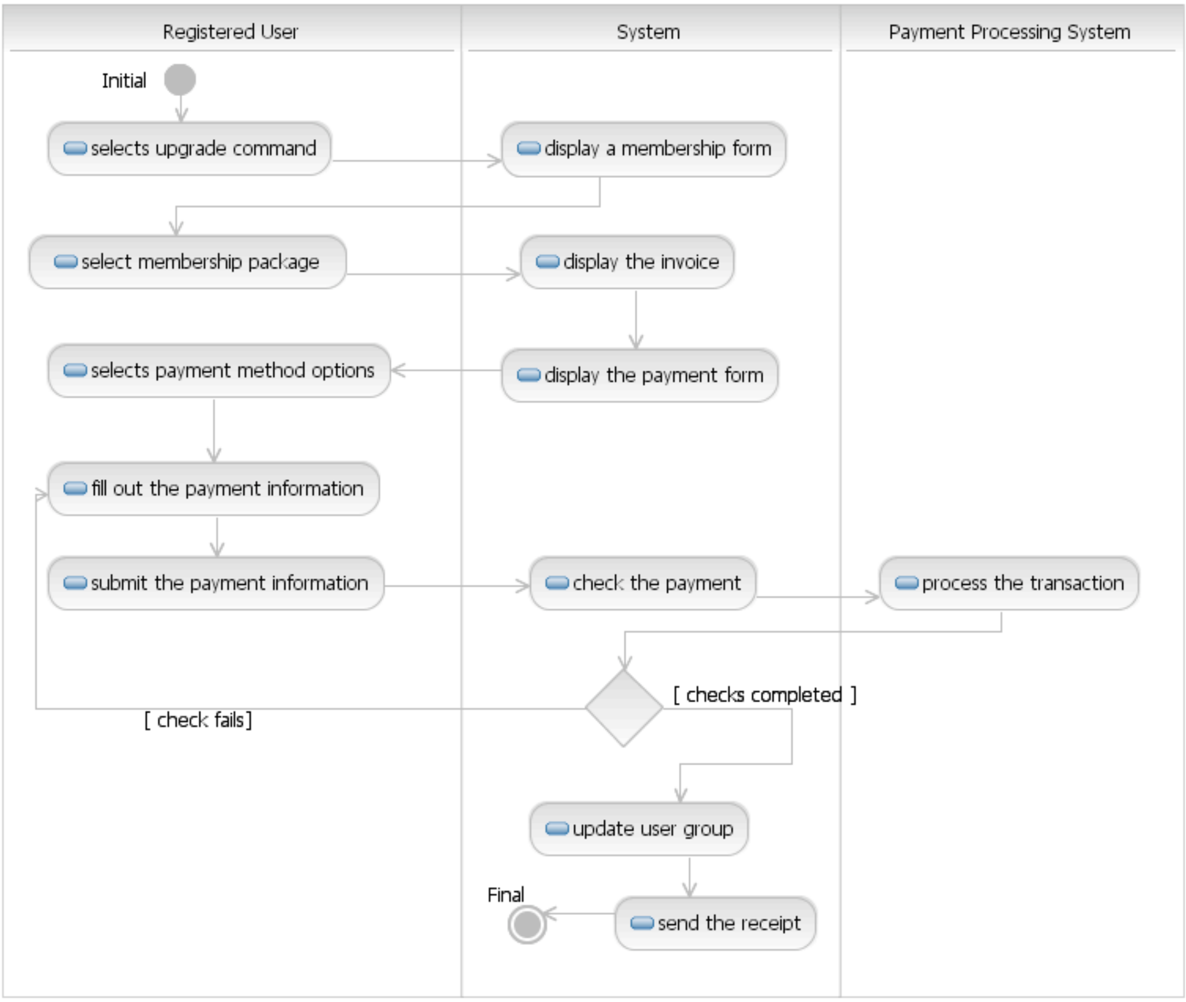
This section presents the business use cases of the subject area in a graphical form.

GMS use cases diagram

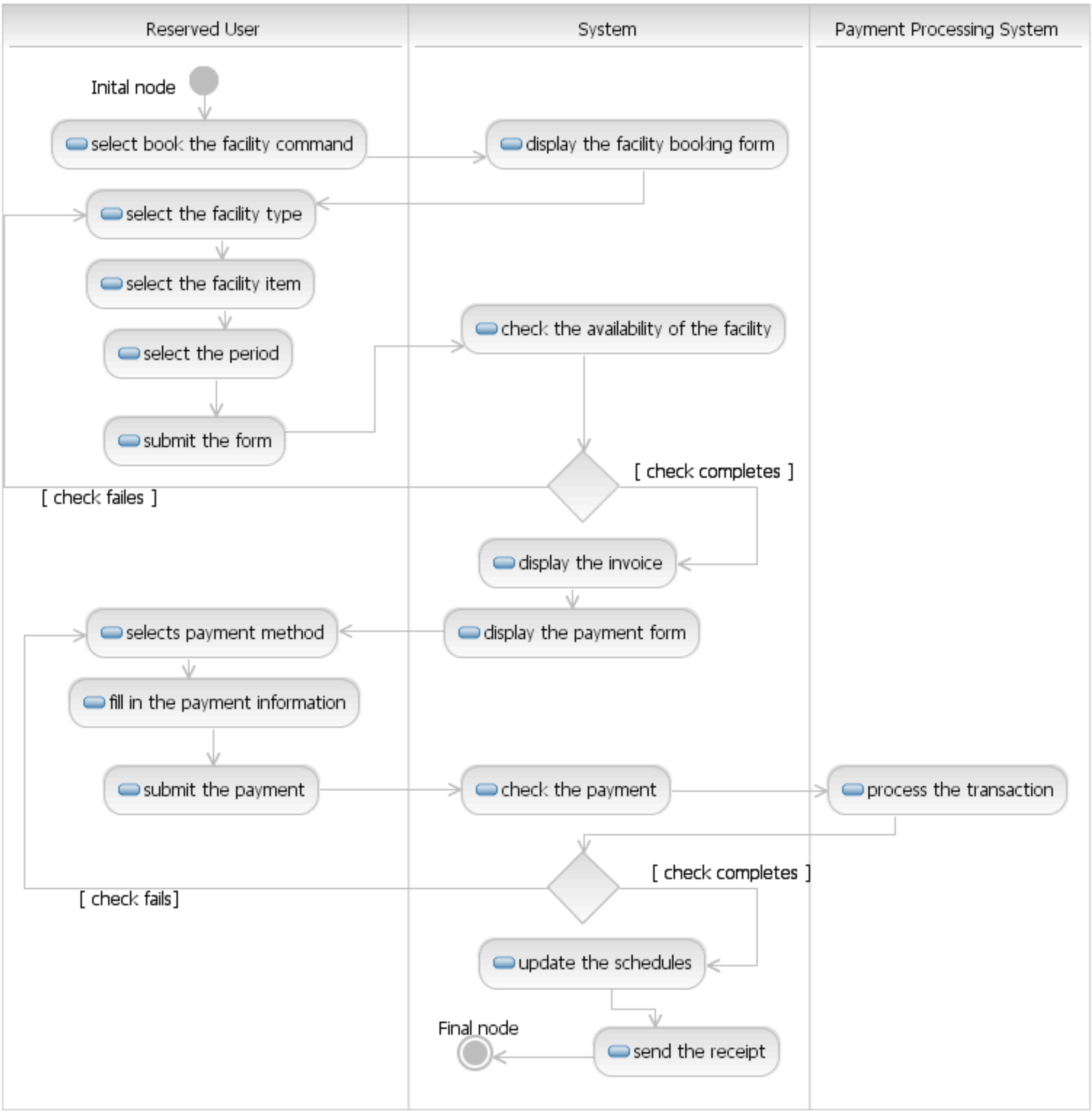


3.4 Activity Diagrams

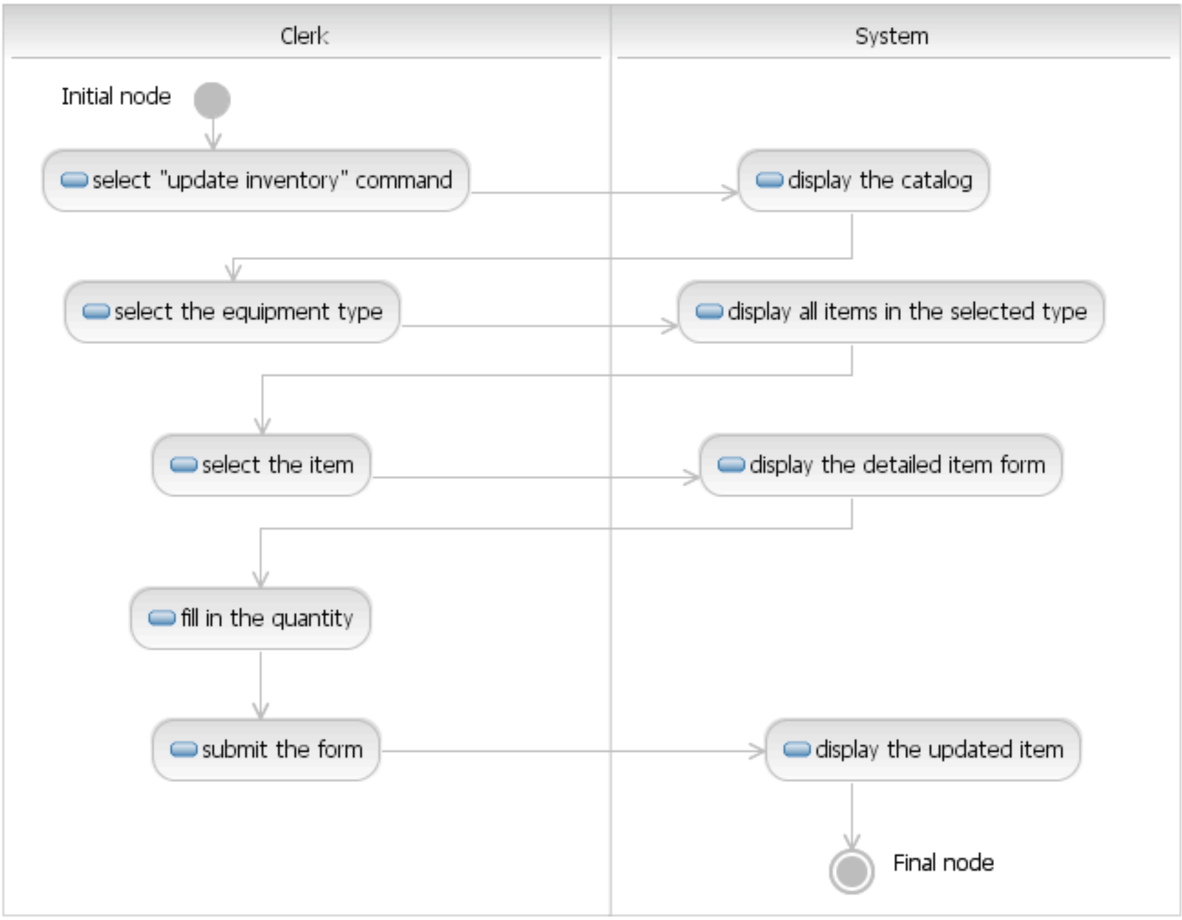
3.4.1 Upgrade to member activity diagram



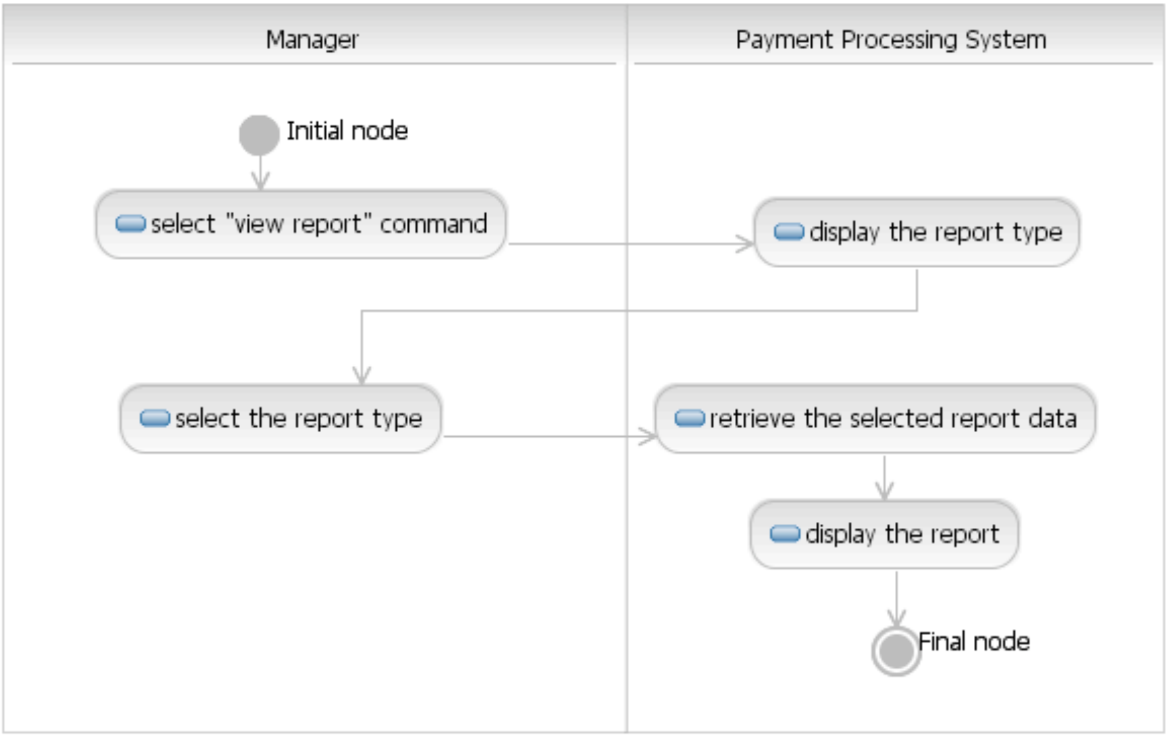
3.4.2 Reserve facility activity diagram



3.4.3 Update inventory activity diagram


Clerk
System
Initial node
select "update inventory" command
display the catalog
select the equipment type
display all items in the selected type
select the item
display the detailed item form
fill in the quantity
submit the form
display the updated item
Final node

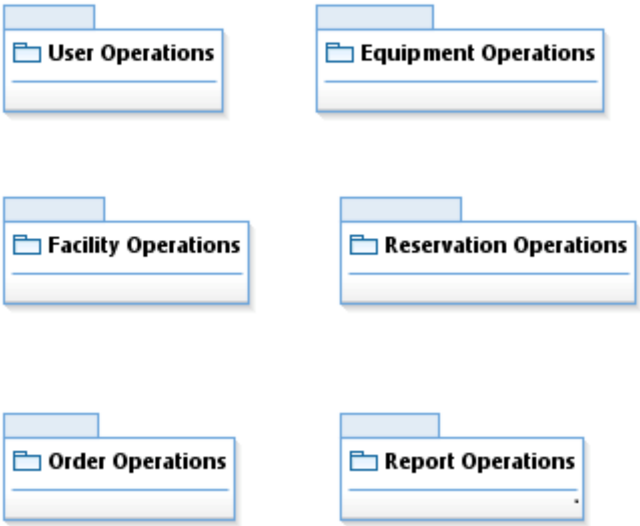
3.4.4 Generate report activity diagram



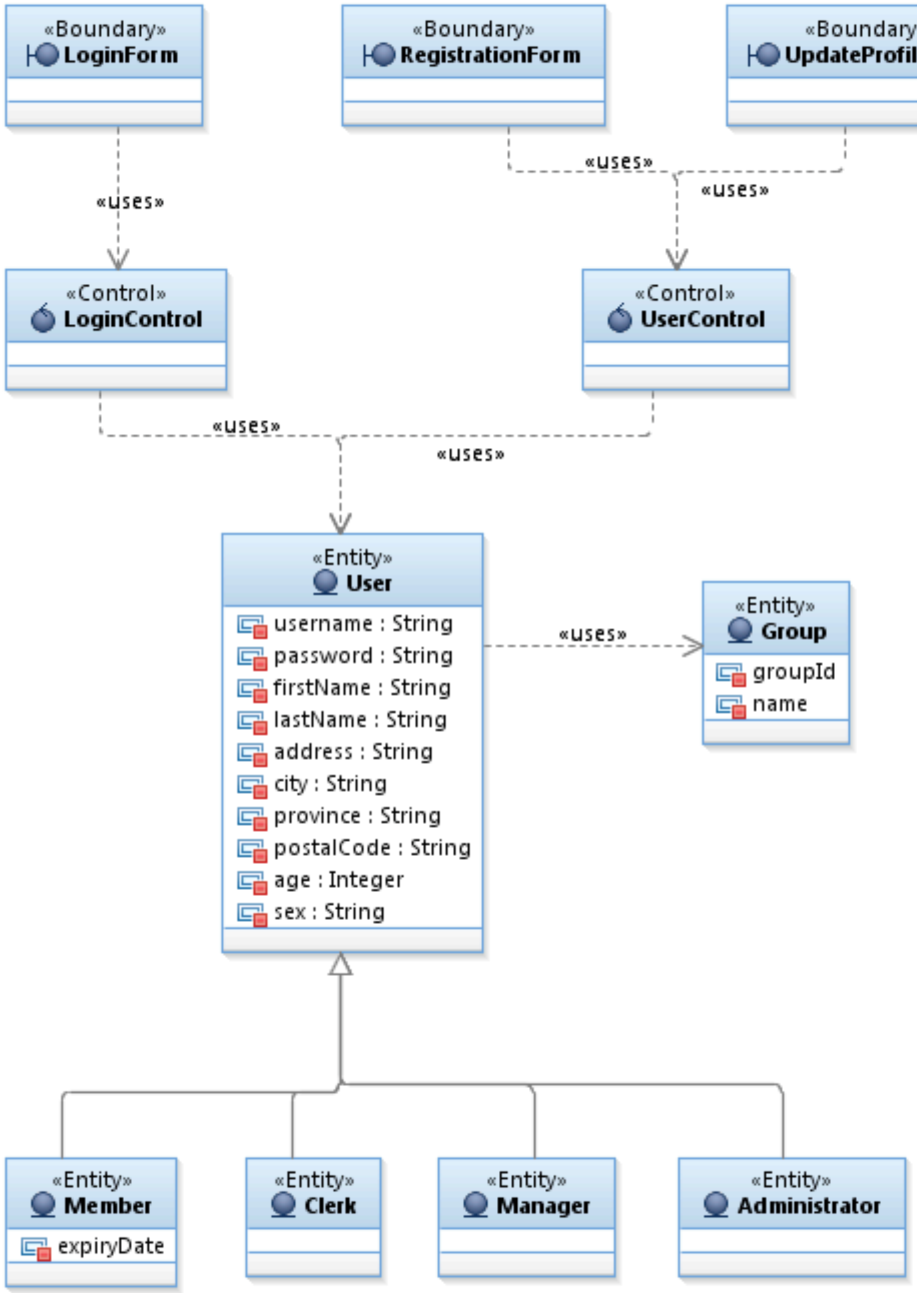
4 Structural Models

4.1 Domain Model Class Diagram

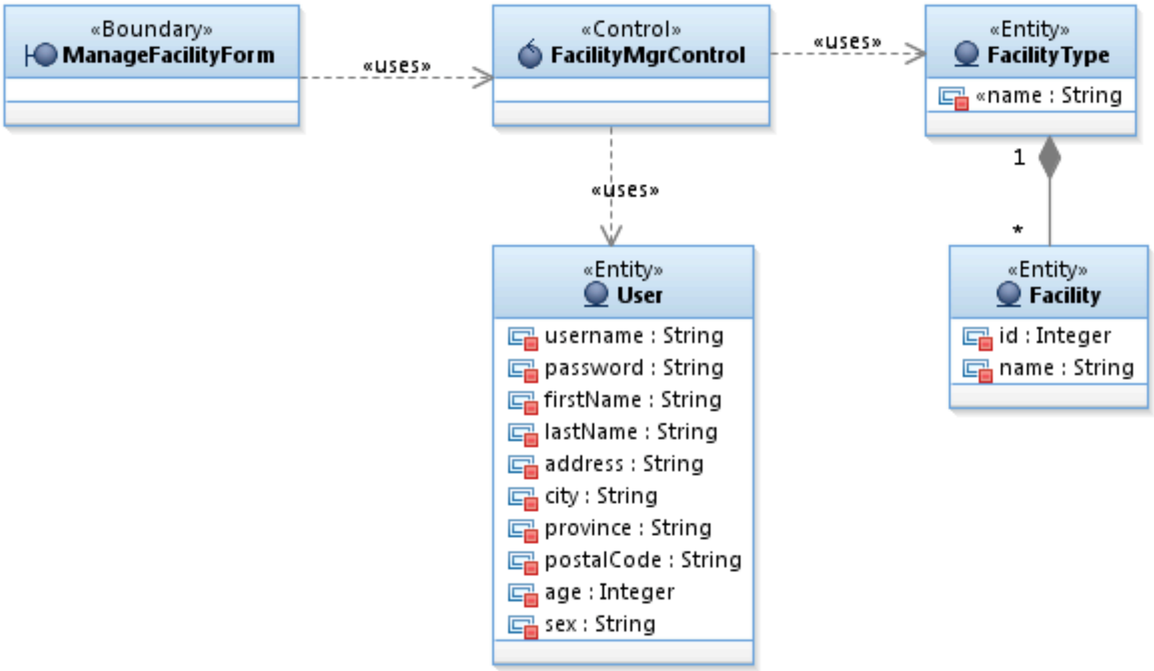
4.1.1 Class Diagram Overview



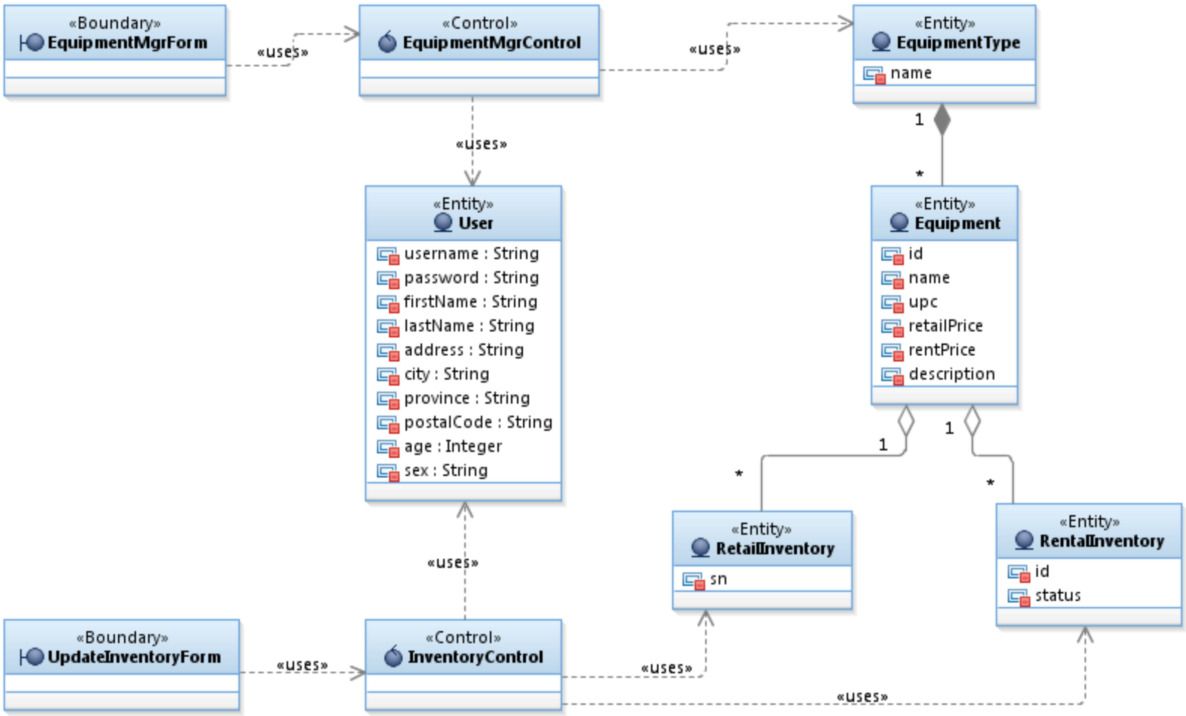
4.1.2 User Operations Class Diagram



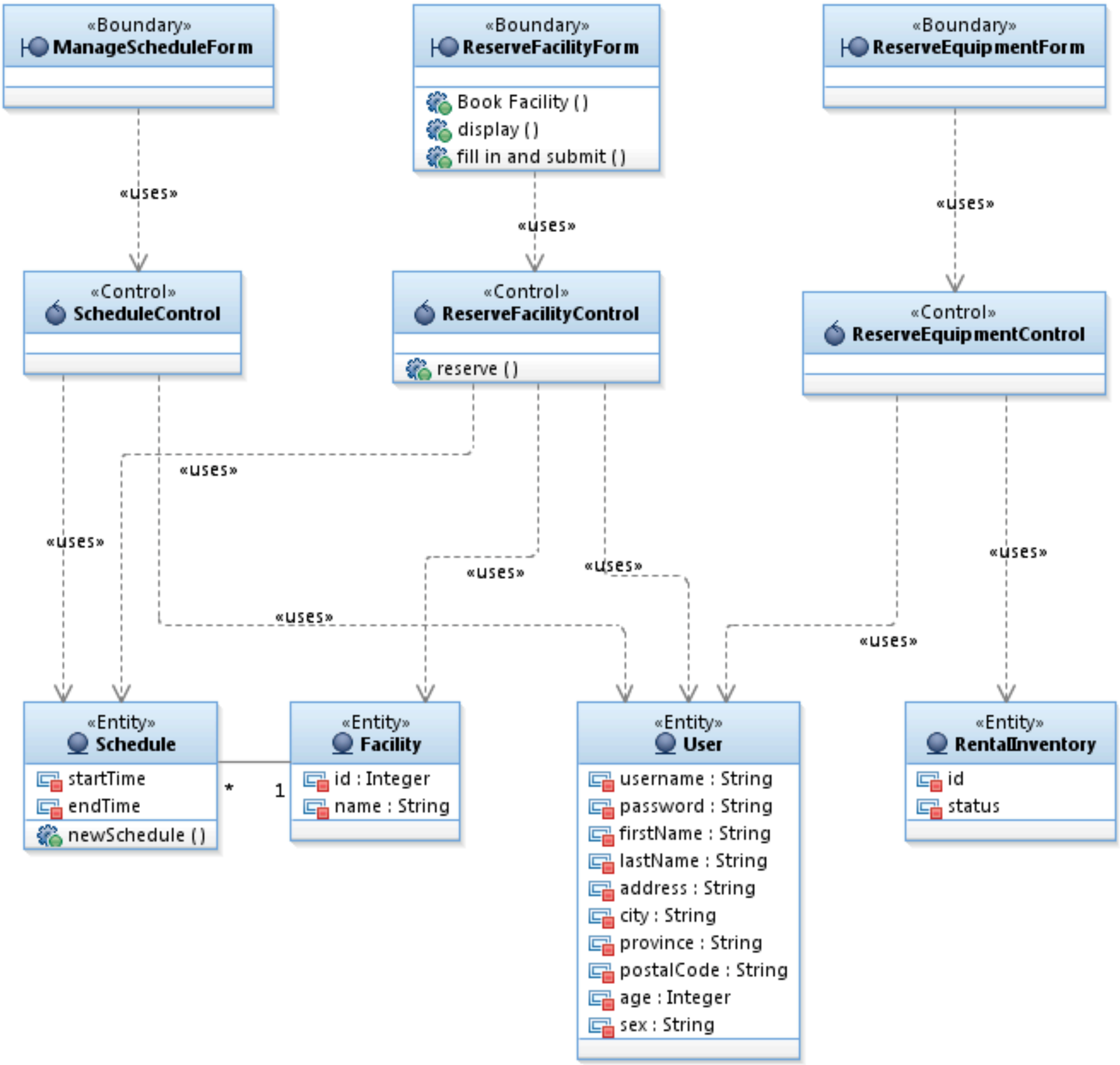
4.1.3 Facility Operations Class Diagram



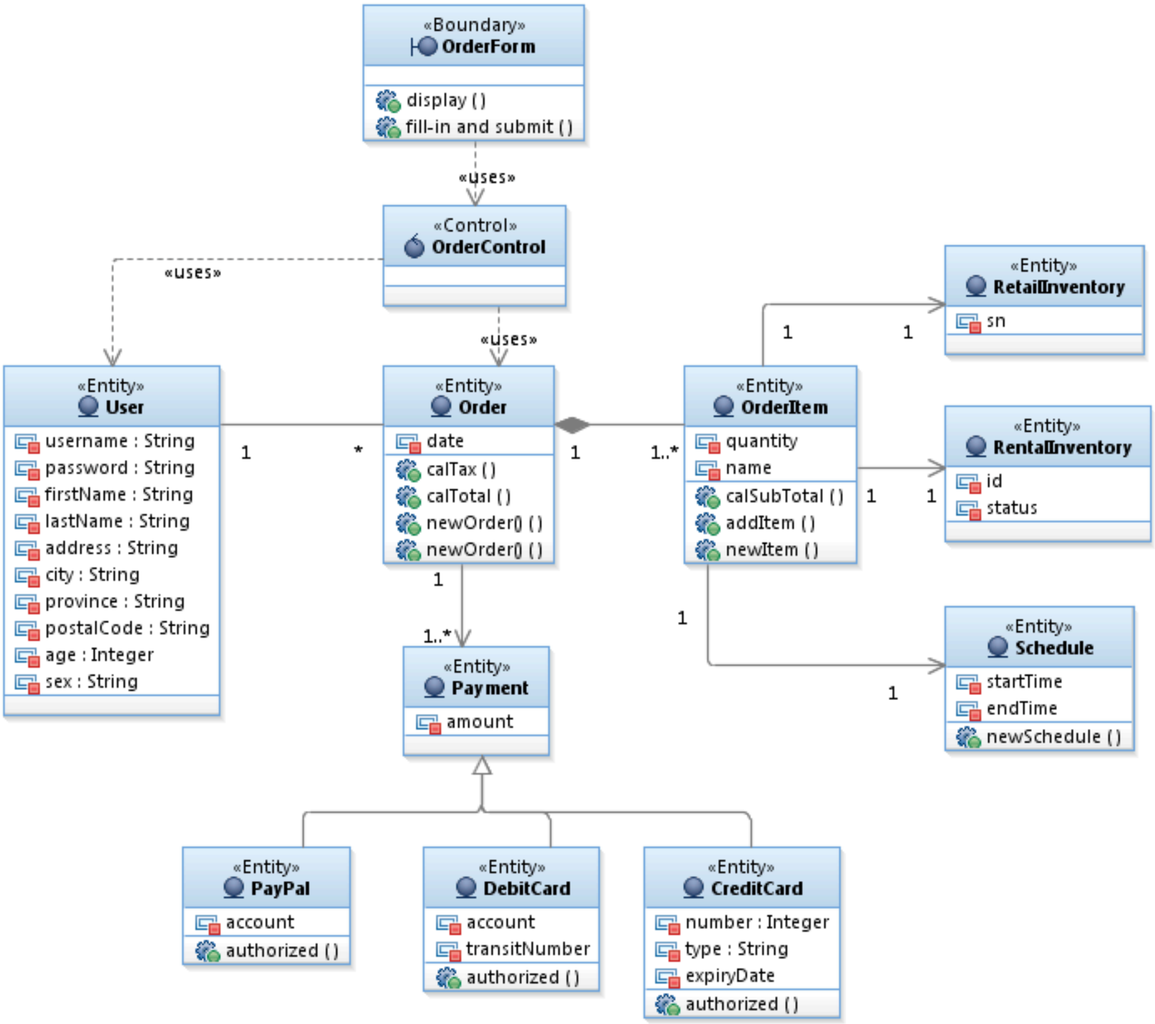
4.1.4 Equipment Operations Class Diagram



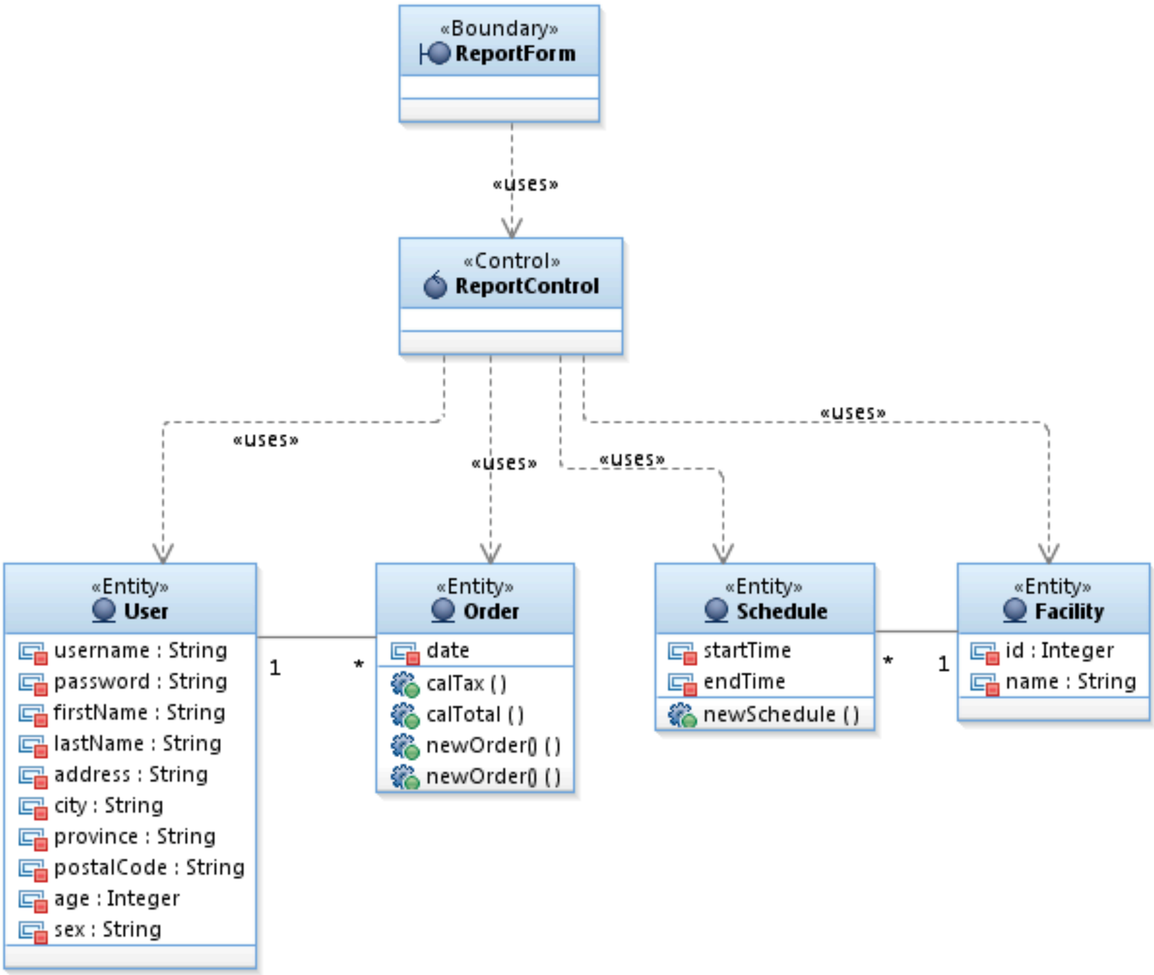
4.1.5 Reservation Operations Class Diagram



4.1.6 Order Operations Class Diagram



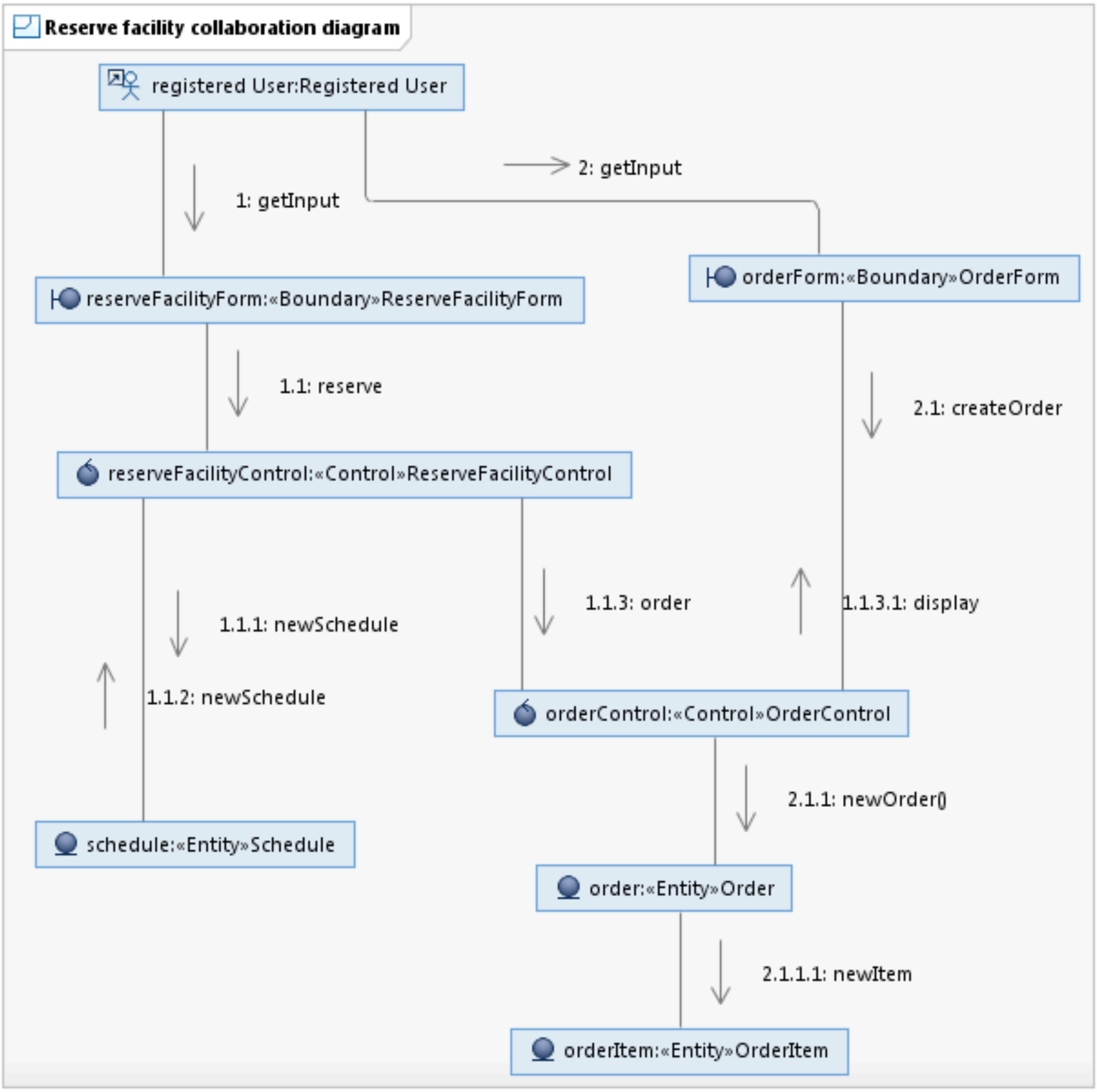
4.1.7 Report Operations Class Diagram



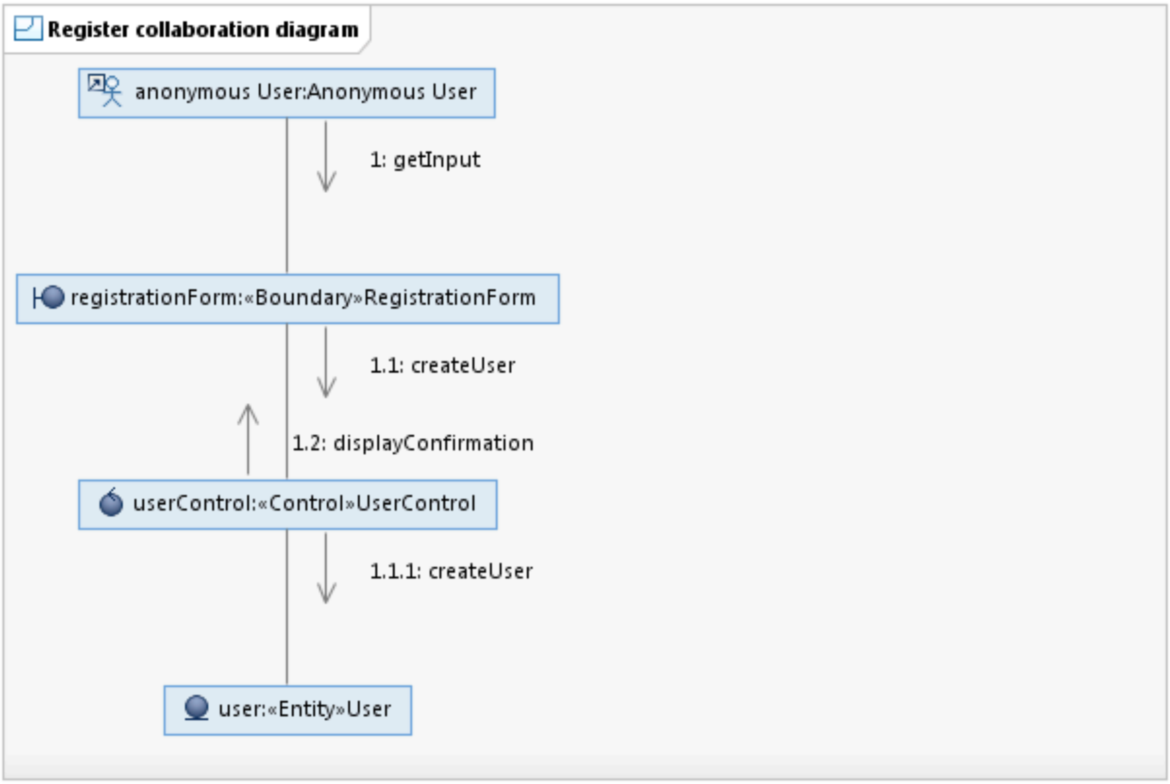
5 Behavior Models

5.1 Collaboration Diagrams

5.1.1 Reserve facility collaboration diagram


Reserve facility collaboration diagram
registered User:Registered User
1: getInput
2: getInput
reserveFacilityForm:«Boundary»ReserveFacilityForm
orderForm:«Boundary»OrderForm
1.1: reserve
2.1: createOrder
reserveFacilityControl:«Control»ReserveFacilityControl
1.1.1: newSchedule
1.1.2: newSchedule
1.1.3: order
1.1.3.1: display
orderControl:«Control»OrderControl
schedule:«Entity»Schedule
2.1.1: newOrder()
order:«Entity»Order
2.1.1.1: newItem
orderItem:«Entity»OrderItem

5.1.2 Register an account collaboration diagram


Register collaboration diagram
anonymous User:Anonymous User
1: getInput
registrationForm:«Boundary»RegistrationForm
1.1: createUser
1.2: displayConfirmation
userControl:«Control»UserControl
1.1.1: createUser
user:«Entity»User

6 Non-Functional Requirements Analysis

6.1 Overview

The Non-Functional Requirements Analysis documents and tracks the necessary information required to effectively define business and non-functional requirements. The document is created during the planning phase of the project. Its intended audience is the project manager, project team, project sponsor, client/user, and any stakeholder whose input or approval into the requirements definitions is needed. Our project "The Alpha Gym Facility Management System" must conform to a few non-functional requirements that we consider critical for the success of our project. These include fast response time, stable CPU usage, use of data encryption to protect sensitive information about customers, appropriate downtime to perform deployment, and approach to recovering data and restarting system.

6.2 Capacity Planning

6.2.1 Permanent Storage

1 TB to be expected. Types of data include image, text, and customer information.

6.3 Network

The best approach to application architecture is MVC model. Since MVC model is nonlinear and since database is separated from the model, controller does not have to wait for the responses from database and therefore, this model is more time efficient than the others. Moreover, backend databases in this model are better suited for a downtime data backup. Database downtime backup does not impact as much as in the other models.

Speaking of LAN design, the best approach will be 10/100 base T Cat 5 Ethernet cables. Cat 5 is a measure of quality and it supports traffic up to 100 MB per second. The advantage of this approach is that client-side can make requests at 10 MB per second to the server-side. This reduces the probability of bottlenecks. Bottlenecks occur when the server receives too many requests from clients and it increases the response time. Cables should be full duplex so that data can travel both directions.

6.4 Workstations

Explore the requirements for a workstation by covering the following subjects:

- Disk space: from 500 MB to 1 GB.
- Performance: Fast processing, enough memory and disk space to process new software which require higher performance.
- Memory: 500 MB to 2 GB
- Screen attributes: support 1280 X 1024 or higher

- Processor requirements: 1.8 GHZ or higher. Intel products
- Interfaces: Major browsers such as IE, Firefox, Chrome, and Safari. Monitors should be at least 19 inches. Windows will be used as operating systems. Video cards based on DDR2 will be used and it generates 533 - 1000 MHz memory clock rate.

6.5 Operational Parameters

6.5.1 Usability

The system to-be must be interface user-friendly. Anonymous users can access and see availability of facilities. However, anonymous users cannot reserve / book facilities before they register as a user. When booking a facility, a calendar should pop up and let users choose from the calendar. System should work on cross-browser. (E.g. Chrome, IE) Highlighted error messages should be returned when payment or login is denied. Highlighted error messages should be returned when reservations not available. Required fields should be highlighted. System should adopt fonts that are cross-browser. (E.g. Serif, Sans-Serif)

6.5.2 Reliability

Data must be SSL encrypted. Payments must be processed using S-HTTP. Data must be backed up in a regular basis. (E.g. weekly, biweekly, monthly) System should prompt user to restart browser or re-login in case of timeout. Reservation and payment confirmations must be delivery guaranteed. Backward compatibility must be enabled in case deployment goes wrong. Error messages should be returned if login credentials or payment info are invalid. System administrator should restart servers in the cluster when system is down. Maximum downtime is 1%, and therefore, 7 hours and 18 minutes per month. System should run 7/24. Dual Password and Secret Questions should be prompted when login to prevent malicious attacks.

Recoverability & Backup

Backward compatibility is guaranteed in our system to-be. In case deployment goes wrong, system will be rolled back to the previous build release version. Databases will be backed up in a regular basis to ensure customer data will not be lost in any occasions. Major server is located will be located in Toronto and Back-up server will be located in Waterloo.

Restart

System administrator must restart servers in clusters no later than 1 hour after the system is down. Maximum downtime per month is 1. That means 7 hours and 18 minutes is the maximum acceptable downtime.

6.5.3 Maintainability

System administrators should be able to make change on schedules and add new items on the catalog (maintenance). This is usually done at midnights from 3am - 5am where minimal impacts on customers are guaranteed. Deployment must also take place during midnights. When performing maintenance, clusters should be turned off and servers should be restarted afterward.

The foreseen extensions to the system are very high. As we will be expanding the boundary of our system, new Java Classes are expected. For instance, some new features such as suggesting fitness plan according to customers' weight and height will be implemented soon. In that case, some new Java Classes will be needed.

6.5.4 Portability

System must be cross-browser. Sever should be portable from Red Hat to other environments such as HP UX or any other Unix-based machines.

- System will be written in Java/J2EE. Java is a portable language. When .java file is complied, a .class file is created. .class files need JVM to run it. JVM can run on any systems (e.g. Linux, Windows) to read the byte code in .class and interpret it into native machine language in runtime. Therefore, the system to-be is portable.
- JDK 6 from Sun Microsystems will be adopted as the compiler. JDK is run able on Linux, UNIX, Windows, and Mac OS. Therefore, it is portable.
- System can be run on Windows, Linux, UNIX, and Mac OS.

7 Glossary (describe the objects of the problem domain)

GMS – Gym Management System

Anonymous User - A non-registered user who only has the privilege to browse the public area such as schedules.

Registered User - A registered user that has basic privileges such as book the facility and reserve the equipment. Registered users will make up the majority of the system.

Member - A registered user that has upgraded privileges such as higher priority in booking the facility.

Clerk - A registered user with the privileges to manage the facility, equipment, users and process the payment.

Manager - A registered user with all the privileges of clerk and access to the accounting information.

Administrator - Top-level user with the full privileges to access the whole system.

Accounting system - An internal system that processes all accounting information.

Payment system - An external system processes all the online transactions through Credit Card, PayPal, Debit Card and Cash.

User Class – actors who uses the system

Group Class – groups that user belongs

LoginControl Class – performs the login action

UserControl Class - performs the actions to the users, e.g. add, modify, delete

FacilityType Class – Entity class that defines the facility type

Facitity Class – Entity class defines facility

FacilityControl Class – Control class that performs actions to facility

EquipmentType Class – Entity Class that defines equipment type

Equipment Class – Entity Class that defines equipment

EquipmentMgrControl Class –Control Class that performs actions to equipment

RetailInventory Class – Entity class that defines the retail equipment inventory

RentalInventory Class – Entity class that defines the rental equipment inventory

InventoryControl class – Control class that performs actions to Inventory

Schedules Class – Entity class that defines the schedule

ScheduleControl Class – Entity class that performs actions to schedules

Order Class – Entity class that defines the order

OrderItem Class – Entity class that defines items in the order

Payment Class – Entity class that defines the amount of the order

Paypal Class – Entity class that inherits from Payment class

CreditCard Class – Entity class that inherits from Payment class

DebitCard Class – Entity class that inherits from Payment class

OrderControl Class – Control class that performs actions to order

ReportControl Class – Control class that performs actions to report

8 Appendix 1. Interview reports

Title of the paper: Gym Management System (GMS system)

Author(s) name: Alpha Software Group

Corresponding author's e-mail ID: sand2@googlegroup.com

Introduction: Gym Management System is a web-application that facilities the usage of stakeholders and end users and helps the gym providing the better service to the customers. We would like to gather as much detailed information as possible to analyze your requirements. Please complete the questions to help us better understand your needs.

Materials & Methods: Personal interview

References: YMCA, Peter rob; Maria Wan, Good life; Nora, Extreme fitness

Results, discussion, and conclusion:

After one to one discussion, below are the highlights:

- Most of the other gym has two level of membership. They are classified as regular and premium. The system should have scalability to accommodate future needs of the customers
- Customers are using phone line to book the facility. Due to innovative smart phone and cheap data plan, more people are expected to use Internet in near future. The system should accommodate this feature
- A real time calendar should reflect up to date status
- The system should provide option to sell and rent the equipment. The profitability will increase by 10% gross and the customer satisfaction will increase by 22%
- The system should provide various payment methods. It is cost efficient to outsource the payment processing function to a third party
- To avoid abuse of the facility, the business rules should be integrated within the system. The system should be restricting excess usage of the facility. Every member should be treated equally.
- Reports are key for any business. Reports can be manipulated id there is any human intervention. The system should focus on automated reports such as facility utilization, member activities. Crystal reporting tools should be integrated for accurate reports

9 Appendix 2. Questionnaire report

The main focus of the questionnaire was to elicit functional requirements for our system. The questionnaire consists of 20 questions and we got 11 responses. Most cases participants needed to single out one most important requirement whereas they were asked to pick all requirements they desired in some questions.

From the analysis of the questionnaire, several patterns became apparent. The popular function requirements are to check the availability of the facility online and to book the facility online. 90% participants desired to check the availability online. Without this system, 45% participant called in and 45% participants went to the gym to check the availability. From the questionnaire, 73% participants are interested in booking the facility online.

We noticed that renting the equipment is very important. Only 45% participants owned their own equipment, 45% participants rent the equipment from the gym. Of which 9% participants did this every time they went to the gym.

One fact that we found was that most participants were not interested in purchasing in equipment online. 82% participants never purchased any equipment from online. Therefore, we decided not implement online store into the system.

The survey also made us aware that some participants are interested in some programs offered in the gym. 55% participants are interested in fitness plan, 27% participants wanted trainer plan. Restricted by the time and expense of this project, we will consider these requirements as future expansion.

Overall, the questionnaire was a helpful tool and allowed us to create a pool of functional requirements for the system.