

ChocAn System

Software Engineering

Final Project

KATO JR., Alberto Sussumu

OLIVEIRA, Rafael Silvestre de

RIBEIRO, Jorge Luís Melo

Waukesha, December 10, 2015

Contents

Communications Plan	4
Use Case Diagram	5
Use Case Specifications.....	6
1. Validate Member Number	6
2. Register The Service Provided.....	7
3. Validate Service Code.....	8
4. Display Service Name.....	9
5. Request Provider Directory	10
6. Generate Report	11
7. Generate Banking File	13
8. Request Report	14
9. Add Member	15
10. Update Member.....	16
11. Delete Member	17
12. Add Provider	18
13. Update Provider	19
14. Delete Provider	20
15. Log In.....	21
Class Diagram	22
Sequence Diagram	23
Test Plan.....	24
1. Login.....	24
2. Password	25
3. Operator Main Screen.....	27
4. Member Operations Screen.....	28
5. Add a Member	29
6. Update a Member.....	31
7. Delete Member	33
8. Provider Operations Screen	34
9. Add a Provider.....	36
10. Update Provider	37
11. Delete Provider	39
12. Main Screen	41

13.	Provider Main Screen.....	43
14.	Check Member's Status	44
15.	Show Member's Status	45
16.	Request the Date of the Service Provided	47
17.	Shows the Provider's Directory.....	48
18.	Confirm Service Code.....	49
19.	Comments.....	50
20.	Description of Service	51
21.	Verification Form	52
22.	Check Fee's Total	53
23.	Request Provider Directory.....	54

Communications Plan

How will you communicate with each other?

Personal meetings on Friday at 2PM.

Daily emails on demand.

Where are the meetings going to happen?

In the learning commons, Todd Wehr Library.

How will you assign tasks?

The tasks will be divided by the amount of use-cases, that is, each member of the project is responsible for a third part of the project, covering every single artifact of the designated part (requirements, analysis, design, implementation and testing).

In order to keep everyone synchronized, the regular meetings will have as goal to keep everyone in track and integrate the modules that each other is responsible for.

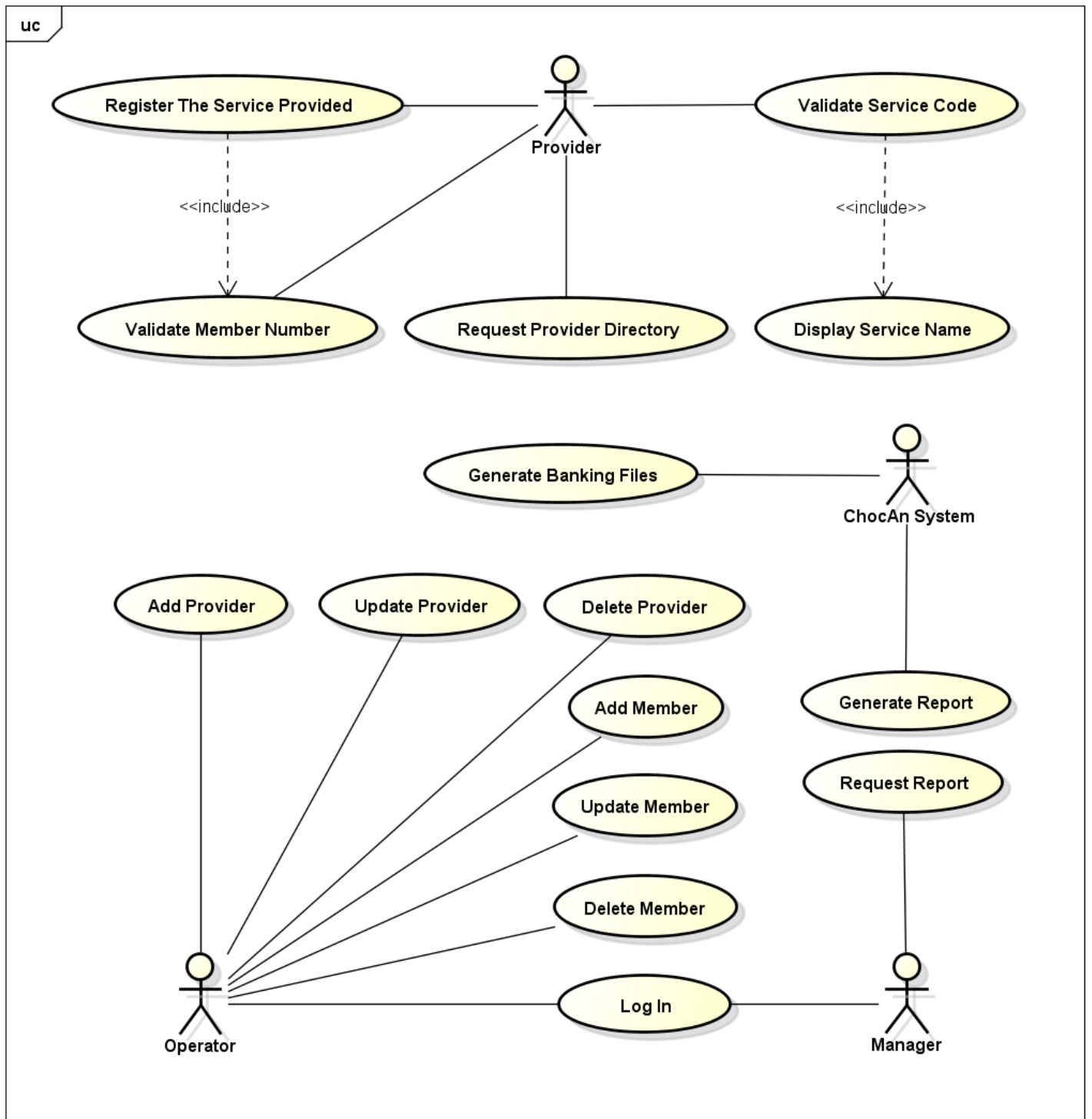
Where will the work be stored?

All the work will be stored in a distributed version control system (GitHub) to allow decentralized management of the software.

How to track the work done?

Utilizing Gantt Chart.

Use Case Diagram



Use Case Specifications

1. Validate Member Number

Brief Description: *the provider swipes the member card through the card reader on the terminal and the process of member identification is started.*

Main Actor: Provider

Pre-condition:

1. None.

Basic flow:

1. Provider swipes member's card;
2. The system verifies the member number;
3. The system displays "Validated" message;
4. The use case ends successfully.

Alternative flows:

1. **Alternative #1 to Step 3.** Invalid number: The number is not valid anymore. A message is displayed "Invalid number";
2. The use case is finished unsuccessfully.

1. **Alternative #2 to Step 3.** Member suspended: The member is suspended due to late fees. A message is displayed "Member suspended";
2. The use case is finished unsuccessfully.

Post-condition:

1. In case of success, the login is validated, otherwise login fails.

2. Register The Service Provided

Brief Description: *the provider enters the information of the service provided in order to track the payment afterwards.*

Main Actor: Provider

Pre-Conditions:

1. Provider must be logged in the system, member must be validated, and the service code must be validated.

Basic flow:

1. The system display a form with text boxes waiting for the information;
2. The process starts in the field date. The provider inserts the date of the service;
3. The provider clicks on “Next” button;
4. The process continues with the fields time, date the service was provided, member name and number, service code, and fee to be paid;
5. The system shows a list of services from the database.
6. The provider select one service code.
7. The provider clicks on “End” button after inserts the fee;
8. The use case ends successfully.

Alternative flows:

1. **Alternative #1 to step 2 and 4.** The provider inserts an invalid data for the field and clicks on “Next” button: the system shows an alert informing what happened;
2. **Alternative #2 to step 6:** The provider insert a wrong service code. The system demand a new service code in a loop.
3. The provider clicks “OK” acknowledging the occurrence;
4. The system redirects to the field insertion screen;
5. The use case is finished unsuccessfully.

Post-condition:

1. The provider is redirected to the system’s menu.

3. Validate Service Code

Brief Description: *the service code is entered by the provider, then a validation process is started by the ChocAn System Server.*

Main Actor: Provider

Pre-Conditions:

1. Provider must be logged in the system.

Basic flow:

1. The system display a text box waiting for the service code;
2. The provider inserts the service code to be validated;
3. The provider clicks on “Validate” button;
4. The system looks for the inserted code in the designated table;
5. The system finds the information regarded to the data informed;
6. The system starts use case “4. Display Service Name”;
7. The use case ends successfully.

Alternative flows:

1. **Alternative #1 to step 5.** The system does not find any information in the designated table: the system shows an alert informing what happened;
2. The provider clicks “OK” acknowledging the occurrence;
3. The system redirects to the code insertion screen.
4. The use case is finished unsuccessfully.

Post-condition:

1. The code is passed to the use case: “4. Display Service Name”.

Business rules:

1. Service code is unique

4. Display Service Name

Brief Description: *the system recognizes the service code provided and shows the service name for confirmation.*

Main Actor: Provider

Pre-condition:

1. Requires success response from the use case “3. Validate Service Code” to be initialized.

Basic flow:

1. The system displays service name;
2. The system displays a confirm box;
3. The provider clicks “yes”;
4. The provider can type comments;
5. The use case ends successfully.

Alternative flows:

1. **Alternative #1 to step 3.** The providers clicks “No”: The system redirects to the type code screen again.
2. The use case is finished unsuccessfully.

Post-condition:

1. The service name is displayed correctly and confirmed.

5. Request Provider Directory

Brief Description: *the provider requests the ChocAn System Server his/her list of services along with service names, codes, and fees.*

Main Actor: Provider

Pre-Conditions:

1. Provider must be logged in the system.

Basic flow:

1. The provider selects the option “Request Provider Directory” on system’s menu;
2. The system displays a confirm box;
3. The provider clicks “Yes”;
4. The system generates the Provider Directory;
5. The system displays all services, codes and fees.
6. The system displays an alert confirming the send;
7. The use case ends successfully.

Alternative flows:

1. **Alternative #1 to step 3.** The provider clicks “No”: The system redirects to the menu;
2. The use case is finished successfully.

Post-condition:

1. The provider is redirected to the system’s menu.

6. Generate Report

Brief Description: *generates the following reports: services provided by each provider, services utilized by each member, and summaries for payable accounts.*

Main Actor: ChocAn System

Pre-Conditions:

1. Use Case is run every midnight on Fridays.

Basic flow:

1. The following reports must be generated:
 - 1.1. Services Utilized by Each Member
The system gathers all information regarded to the services utilized. For each member, these attributes are retrieved or calculated:
 - Member name;
 - Member number;
 - Member street address;
 - Member city;
 - Member state;
 - Member ZIP code.For each service utilized:
 - Date of service;
 - Provider name;
 - Service name.
 - 1.2. Summaries for Payable Accounts
The system gathers all information about all the providers to be paid that week. This report includes the following attributes:
 - Provider name;
 - Provider number;
 - Provider street address;
 - Provider city;
 - Provider state;
 - Provider ZIP code;
 - Number of Consultations;
 - Total Fee.Along with each individual information, the system also prints:
 - Total Number of Active Providers;
 - Total Number of Consultations;
 - Overall Total Fee.
 - 1.3. Services Provided by Each Provider
The system gathers all information regarding the services provided. For each provider, these attributes are retrieved or calculated:
 - Provider name;
 - Provider number;
 - Provider street address;
 - Provider city;

Provider state;
Provider ZIP code.

For each service provided:

Date of service;
Date and time the data were received;
Member name;
Member number;
Service code;
Fee to be paid.

General attributes:

Total number of consultations;
Total fee for week.

2. The use case ends successfully.

Alternative flows:

1. None. Everything is automatic.

Post-condition:

1. All the reports are send individually to each provider/member/manager that it is intended to.

7. Generate Banking File

Brief Description: *an EFT data is written to disk, so that banking computers will ensure the provider's payment.*

Main Actor: ChocAn System

Pre-condition:

1. None.

Basic flow:

1. The system generates EFT;
2. The EFT lists: All providers names ,codes , and the total of money generated from the services provided.
3. The system write everything in the disk;
4. The system permits that bank computer access EFT data;
5. The use case ends successfully.

Alternatives flows:

1. None. Everything is automatic.

Post-condition:

1. The system will provide proper data to the bank computer.

Business Rules:

1. Assumption : Provider get 40% and ChocAn 60%.

8. Request Report

Brief Description: *requests the ChocAn System Server a file containing the services provided by each provider.*

Main Actor: Manager

Pre-Conditions:

1. The manager must be logged in to the system.

Basic flow:

1. The manager selects the option “Request Services Provider Report” on system’s menu;
2. The system displays a confirm box;
3. The manager clicks “Yes”;
4. The system generates the report;
5. The system displays the report with an option to print;
6. The use case ends successfully.

Alternative flows:

1. **Alternative #1 to step 3.** The manager clicks “No”;
2. The system redirects to the menu.

1. **Alternative #2 to step 5.** The manager clicks “Print”: The system prints the report;
2. The use case is finished successfully.

Post-condition:

1. The manager is redirected to the system’s menu.

9. Add Member

Brief Description: *The operator adds a member to the ChocAn Server System.*

Main Actor: Operator

Pre-condition:

1. The operator must be logged in to the system.

Basic flow:

1. The system requires that operator fill a form with the new member info;
2. Operator fills the fields (Name, State, City, Street, and ZIP Code);
3. Operator click on save button;
4. The use case ends successfully.

Alternative flows:

1. **Alternative #1 to Step 2.** Invalid zip code: Error message is displayed on the screen;
2. The system requires the operator to type the zip code until a valid one is inserted;
3. User caser Resumes on the form screen.

Post-condition:

1. The system will save the new member in the database successfully.

Validation rules:

1. All fields are required.
2. Zip Code 5 digits. Only numbers.
3. State only 2 characters.

10. Update Member

Brief Description: *the operator can edit the information of an existing member.*

Main Actor: Operator

Pre-Conditions:

1. The operator must be logged in to the system.

Basic flow:

1. The system shows a list containing the members;
2. The operator selects a member of the list;
3. The operator clicks the "Update" button;
4. A confirmation screen says "Are You Sure?";
5. The operator clicks "Yes";
6. The system loads the form with the info of that member;
7. The operator update the needed fields;
8. The operator clicks on "Update" button;
9. A confirmation screen says "Are You Sure?";
10. The operator clicks "Yes";
11. The system updates the selected member;
12. The use case ends successfully.

Alternative flows:

1. **Alternative #1 to step 4.** The operator clicks "No";
2. The system redirects to the members list.
1. **Alternative #2 to step 7.** The operator inserts an invalid content for the field and clicks on "Next" button;
2. The system shows an alert informing what happened;
3. The provider clicks "OK" acknowledging the occurrence;
4. The system redirects to the field insertion screen.
1. **Alternative #3 to step 9.** The operator clicks "No": the system redirects to the field insertion screen;
2. The use case is finished unsuccessfully.

Post-condition:

1. The system will save the updated member in the database successfully.

Validation rules:

1. All fields are required.
2. Zip Code 5 digits. Only numbers.
3. State only 2 characters.

11. Delete Member

Brief Description: *the operator deletes an existing member from the system.*

Main Actor: Operator

Pre-Conditions:

1. The operator must be logged in to the system.

Basic flow:

1. The system shows a list containing the members;
2. The operator selects a member of the list;
3. The operator clicks the "Delete" button;
4. A confirmation screen says "Are You Sure?";
5. User clicks "Yes";
6. The system deletes the selected member;
7. This Use Case ends successfully.

Alternative flows:

1. **Alternative #1 to step 5.** User clicks "No";
2. The system redirects to the members list.

Post-condition:

1. The system has the selected member deleted.

12. Add Provider

Brief Description: *the operator adds a new provider record to the system by inserting the data correctly.*

Main Actor: Operator

Pre-Conditions:

1. The operator must be logged in to the system.

Basic flow:

1. The system displays the “Insert New Provider” screen;
2. The operator fills in the following fields:
 - Provider Name;
 - Provider Number;
 - Provider street address;
 - Provider city;
 - Provider state;
 - Provider ZIP code.
3. The operator clicks the button “Submit”;
4. The system stores the informed data in the database;
5. The use case ends successfully.

Alternative flows:

1. **Alternative #1 to step 3.** The operator clicks the button “Clear”;
2. All the inserted information will be cleared;
3. The cursor focus is changed to the first field;
4. The use case ends unsuccessfully.

Post-condition:

1. The new provider record is inserted in the database.

Validation rules:

4. All fields are required.
5. Zip Code 5 digits. Only numbers.
6. State only 2 characters.

13. Update Provider

Brief Description: *the operator edits the information of a provider record.*

Main Actor: Operator

Pre-Conditions:

1. The operator must be logged in to the system.

Basic flow:

1. The system shows a list containing the providers;
2. The operator selects a provider of the list;
3. The operator clicks the "Update" button;
4. A confirmation screen asks "Are You Sure?";
5. The operator clicks "Yes";
6. The system loads the form with the info of that provider;
7. The operator update the needed fields;
8. The operator clicks on "Update" button;
9. A confirmation screen says "Are You Sure?";
10. The operator clicks "Yes";
11. The system updates the selected provider;
12. The use case ends successfully.

Alternative flows:

1. **Alternative #1 to step 4.** The operator clicks "No";
2. The system redirects to the providers list.
1. **Alternative #2 to step 7.** The operator inserts an invalid content for the field and clicks on "Next" button;
2. The system shows an alert informing what happened;
3. The provider clicks "OK" acknowledging the occurrence;
4. The system redirects to the field insertion screen.
1. **Alternative #3 to step 9.** The operator clicks "No";
2. The system redirects to the field insertion screen;
3. The use case is finished unsuccessfully.

Post-condition:

1. The system will save the updated provider in the database successfully.

Validation rules:

1. All fields are required.
2. Zip Code 5 digits. Only numbers.
3. State only 2 characters.

14. Delete Provider

Brief Description: *The operator deletes a provider from the ChocAn Server System.*

Main Actor: Operator

Pre-Conditions:

1. The operator must be logged in to the system.

Basic flow:

1. The system displays a list with all providers;
2. The operator select a specific provider;
3. The operator clicks on delete button;
4. The system displays a message ("Are you sure?");
5. The operator clicks "yes";
6. The use case ends successfully.

Alternative flows:

1. **Alternative #1 to Step 5.** Operator clicks "NO";
2. Use case resumes on the list screen until a next action.

Post-condition:

1. The system will delete a specific provider from ChocAn database.

15. Log In

Brief description: *The login provides access to the system.*

Main Actor: Operator

Pre-Conditions:

1. The operator must be logged in to the system.

Basic flow

1. The system displays the login form;
2. The user fills the fields (Login and password);
3. The user clicks in the login button;
4. The system validates user login;
5. The use case ends successfully.

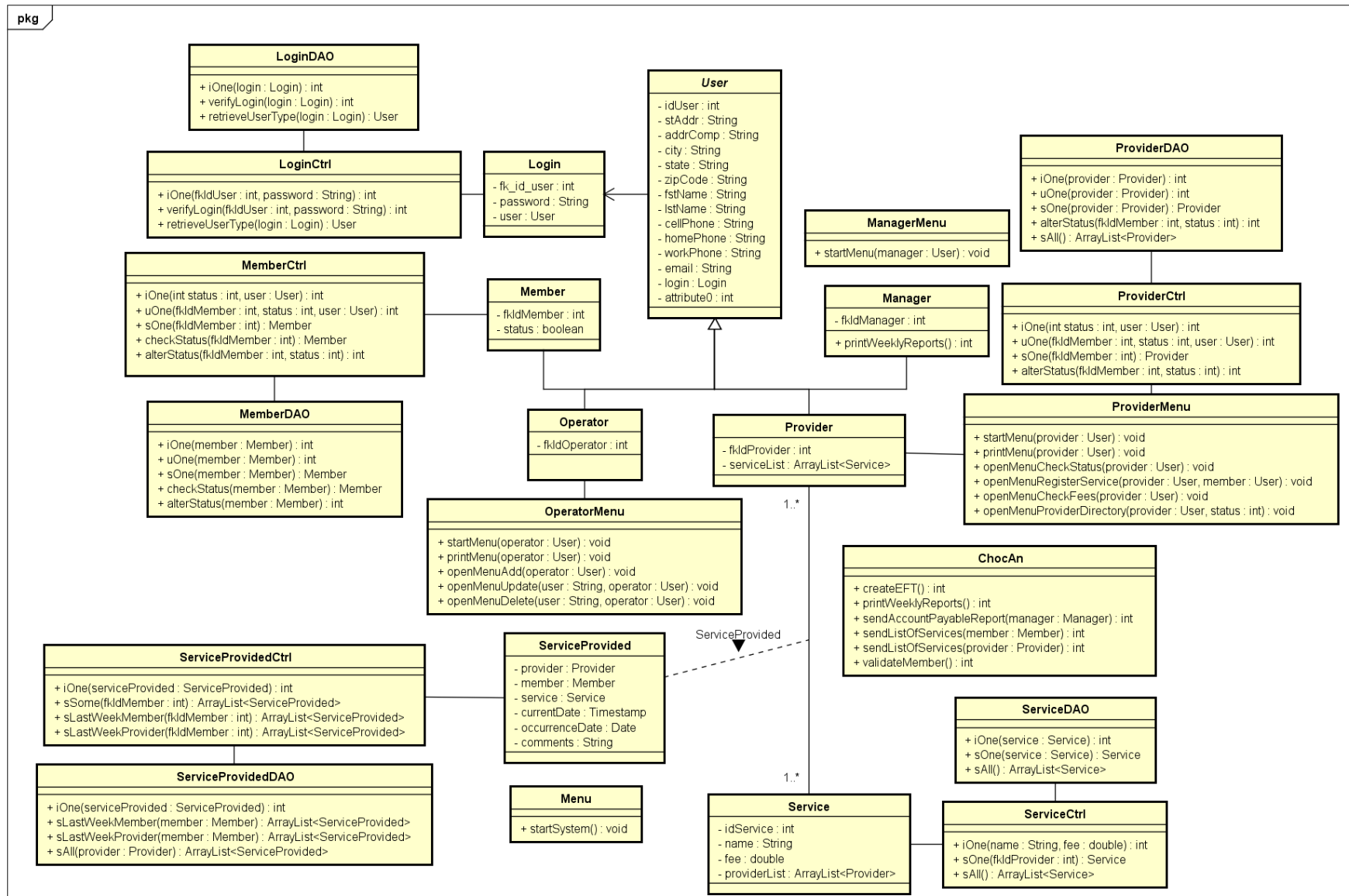
Alternative flows:

1. **Alternative #1 to the step 4.** Username and/or password not recognized;
2. An error message is displayed;
3. The use case ends unsuccessfully.

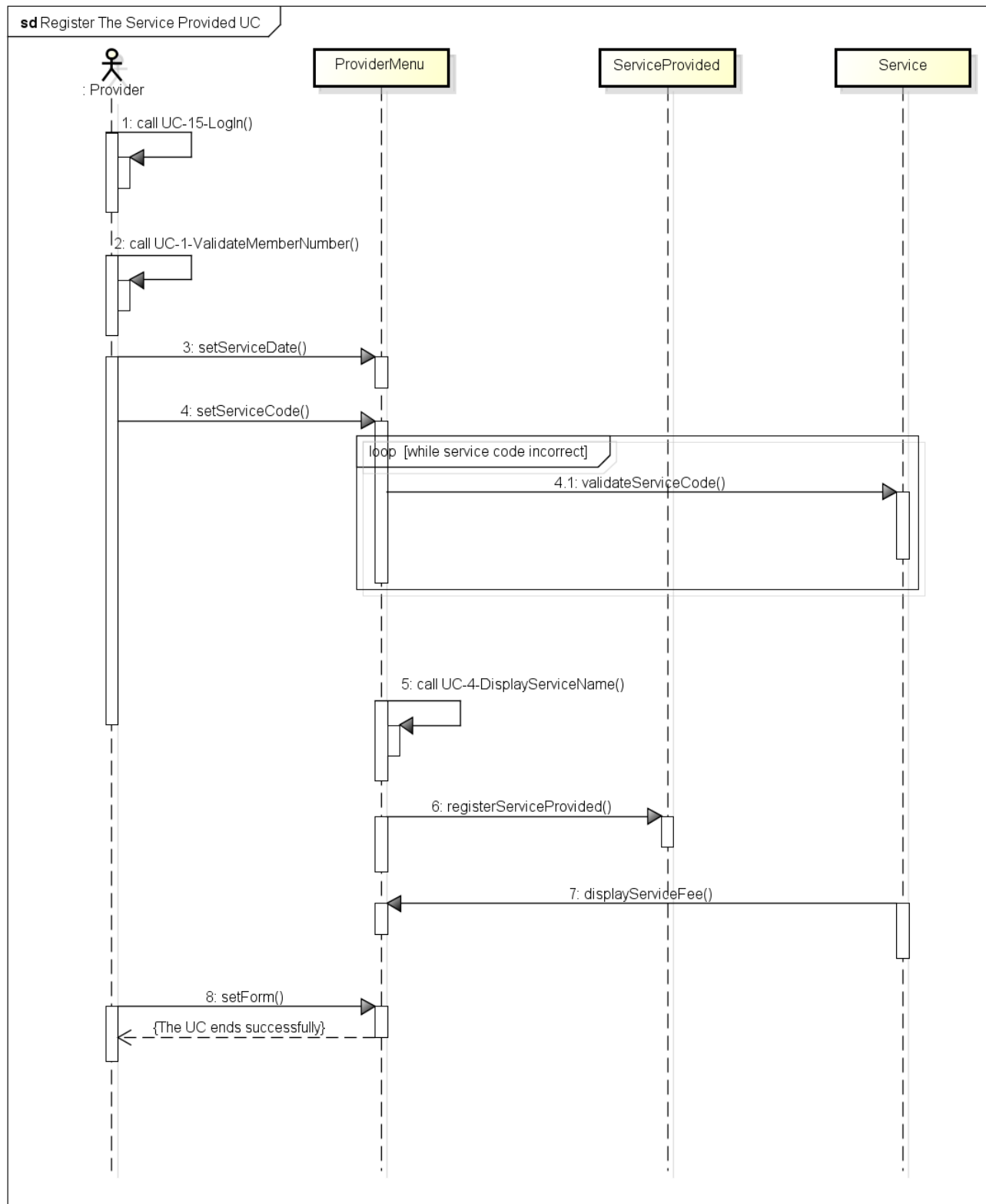
Post-condition:

1. The Operator, or the manager will be redirected to their respective page.

Class Diagram



Sequence Diagram



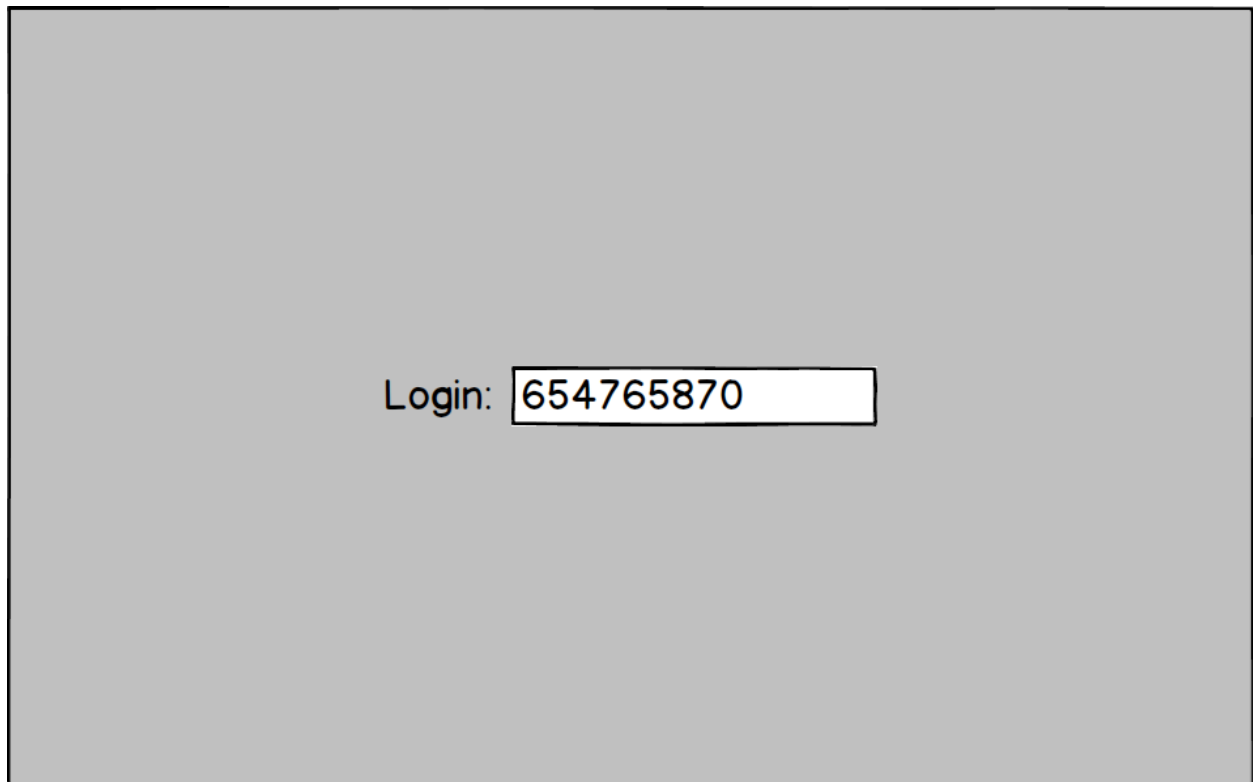
Test Plan

Table of Assigned Tests for each team member:

Assigned Tests	
Tester	Responsibility
Alberto	He will be responsible for the following range of tests: 01 - 08
Jorge	He will be responsible for the following range of tests: 09 - 15
Rafael	He will be responsible for the following range of tests: 16 - 23

1. Login

1.1. User type the login number.



Login: 654765870

1.2. Test Table:

Test description	Input data	Expected results	Passed: Y N?
User types wrong login number.	123456	Shows message: Invalid login. Asks for login number again.	
User types correct login number.	654765870	Redirects password screen	
Provider types blank.	""	Shows message: Type a valid login.	

2. Password

1.3. User types password.

Password:

1.4. Test Table:

Test description	Input data	Expected results	Passed: Y N?
Wrong password.	dragon12345	Shows message: Invalid password. Asks for password again.	
User type: Operator && Valid password.	operator12	Redirects to operator main screen	
User type: Provider && Valid password.	provider12	Redirects to provider main screen.	
User type: Manager && Valid password.	manager12	Redirects to manager main screen.	

3. Operator Main Screen

1.5. Operator types operation number.

Operator: Jorge

Choose the operation:

1. Member
2. Provider
3. Logoff

Option:

1.6. Test Table:

Test description	Input data	Expected results	Passed: Y N?
Operator types inexistent option.	98	Shows message: This operation is not valid.	
Operator types blank.	""	Shows message: Type a valid operation	
Operator selects member operations.	1	Redirects to screen: Member - operations	

Operator selects provider operation.	2	Redirects to screen: Provider - operations	
Operator selects logoff operation.	3	Redirects to login screen	

4. Member Operations Screen

1.7. Operator types one member-operation number.

Operator: Jorge

Choose the type of operation:

1. Add member
2. Update member
3. Delete member
4. Back

Option:

1.8. Test Table:

Test description	Input data	Expected results	Passed: Y N?
------------------	------------	------------------	-----------------

Operator wants to Add a member.	1	Redirects to screen Add a member.	
Operator wants to update a member.	2	Redirects to screen Update a member.	
Operator wants to delete a member.	3	Redirects to screen Delete a member.	
Operator wants to return to previous screen.	4	Redirects to screen Operator main screen.	
Operators types an invalid number.	77	Shows message: Invalid operation.	
Operator inputs a not valid number.	a	Shows message: Input a valid number.	
Operator keys 'enter' (blank value)	""	Shows message: Type something valid.	

5. Add a Member

- 1.9. Operator types First Name (Required)
- 1.10. Operator types Last Name (Required)
- 1.11. Operator types Email (Required)
- 1.12. Operator types cell/Home/Work (at least one Required)
- 1.13. Operator types Address (Required)
- 1.14. Operator types Address Line 2 (Optional)
- 1.15. Operator types City (Required)
- 1.16. Operator types State (Required)
- 1.17. Operator types Zip Code (Required)

Operator: Jorge

Operation: Add member

First Name:

Last Name:

Email:

Cell Phone:

Home Phone:

Work Phone:

Address:

Address Line 2:

City:

State:

Zip Code:

Do you want to add this member? (Y/N):

1.18. Test Table

Test description	Input data	Expected results	Passed: Y N?
Operator left a blank field in any of the required fields.	""	System asks for the operator fill the field again.	
All the required fields are filled properly and operator types Y-y.	All member information and Y	Saves a new member record in the database.	
Operator types an invalid Zip Code.	12	Requests for a new Zip Code.	
Operator types a wrong state abbreviation.	ZZ	Requests for new US state abbreviation.	
Operator does not want to add the member anymore.	N-n	Returns to member operation screen without saving anything in the database.	

Operator types a different letter than (Y-N) in the question.	i	Shows message: Invalid answer. Request for other letter input.	
---------------------------------------------------------------	---	-------------------------------------------------------------------	--

6. Update a Member

- 1.19. Operator types the member ID.
- 1.20. System displays all member information.
- 1.21. Operator leave the field in blank for no changers, otherwise fill with new data.
- 1.22. Operator types Y

Operator: Jorge

Operation: Update member

Enter the ID:

First Name:	Jorge
Last Name:	Ribeiro
Email:	jriberio@pio.carrollu.edu
Cell Phone:	262 309 8725
Home Phone:	
Work Phone:	
Address:	135 S East Ave
Address Line 2:	North Bergstrom Hall Room
City:	Waukesha
State:	WI
Zip Code:	53186

Leave the field in blank if you don't want to change.

First Name:	
Last Name:	
Email:	
Cell Phone:	
Home Phone:	
Work Phone:	
Address:	
Address Line 2:	
City:	
State:	
Zip Code:	

Do you want to update this member? (Y/N):

1.23. Test Table

Test description	Input data	Expected results	Passed: Y N?
------------------	------------	------------------	-----------------

Fields are filled properly and operatory types Y-y.	All member requested information and Y	Updates member record in the database.	
Operator types an invalid Zip Code.	12	Requests for a new Zip Code.	
Operator types a wrong state abbreviation.	ZZ	Requests for new US state abbreviation.	
Operator does not want to add the member anymore.	N-n	Returns to member operation screen without updating anything in the database.	
Operator types a different letter than (Y-N) in the question.	i	Shows message: Invalid answer. Request for other letter input.	

7. Delete Member

- 1.24. Operatory types the member ID
- 1.25. System displays all the member information.
- 1.26. Operators types Y and deletes the member from the system.

Operator: Jorge

Operation: Delete member

Enter the ID:

First Name:	<input type="text" value="Jorge"/>
Last Name:	<input type="text" value="Ribeiro"/>
Email:	<input type="text" value="jriberio@pio.carrollu.edu"/>
Cell Phone:	<input type="text" value="262 309 8725"/>
Home Phone:	<input type="text"/>
Work Phone:	<input type="text"/>
Address:	<input type="text" value="135 S East Ave"/>
Address Line 2:	<input type="text" value="North Bergstrom Hall Room"/>
City:	<input type="text" value="Waukesha"/>
State:	<input type="text" value="WI"/>
Zip Code:	<input type="text" value="53186"/>

Do you want to delete? (Y/N):

1.27. Test Table

Test description	Input data	Expected results	Passed: Y N?
Operator types invalid member ID	83729	Show message: invalid member ID.	
Operator types valid member ID	123456	Display all member information	
Operator types Y-y to the delete question.	Y-y	System deletes the member record from the database	
Operator types N-n to the delete question	N-n	System redirects operator to member operation screen	
Operator types invalid letter for the delete question.	3	System display message: invalid input and asks for the answer again.	

8. Provider Operations Screen

1.28. Operator types one provider-operation number.

Operator: Jorge

Choose the type of operation:

1. Add provider
2. Update provider
3. Delete provider
4. Back

Option:

1.29. Test Table

Test description	Input data	Expected results	Passed: Y N?
Operator wants to Add a provider.	1	Redirects to screen Add a provider.	
Operator wants to update a provider.	2	Redirects to screen Update a provider.	
Operator wants to delete a provider.	3	Redirects to screen Delete a provider.	
Operator wants to return to previous screen.	4	Redirects to screen Provider main screen.	
Operators types an invalid number.	77	Shows message: Invalid operation.	
Operator inputs a not valid	a	Shows message:	

number.		Input a valid number.	
Operator keys 'enter' (blank value)	""	Shows message: Type something valid.	

9. Add a Provider

- 1.30. Operator types First Name (Required)
- 1.31. Operator types Last Name (Required)
- 1.32. Operator types Email (Required)
- 1.33. Operator types cell/Home/Work (at least one Required)
- 1.34. Operator types Address (Required)
- 1.35. Operator types Address Line 2 (Optional)
- 1.36. Operator types City (Required)
- 1.37. Operator types State (Required)
- 1.38. Operator types Zip Code (Required)

Operator: Jorge

Operation: Add provider

First Name:	Jorge
Last Name:	Ribeiro
Email:	jriberio@pio.carrollu.edu
Cell Phone:	262 309 8725
Home Phone:	
Work Phone:	
Address:	135 S East Ave
Address Line 2:	North Bergstrom Hall Room
City:	Waukesha
State:	WI
Zip Code:	53186

Do you want to add this provider? (Y/N):

1.39. Test Table:

Test description	Input data	Expected results	Passed: Y N?
Operator left a blank field in any of the required fields.	""	System asks for the operator fill the field again.	
All the required fields are filled properly and operatory types Y-y.	All member information and Y	Saves a new member record in the database.	
Operator types an invalid Zip Code.	12	Requests for a new Zip Code.	
Operator types a wrong state abbreviation.	ZZ	Requests for new US state abbreviation.	
Operator does not want to add the member anymore.	N-n	Returns to member operation screen without saving anything in the database.	
Operator types a different letter than (Y-N) in the question.	i	Shows message: Invalid answer. Request for other letter input.	

10. Update Provider

- 1.40. Operator types the provider ID.
- 1.41. System displays all provider information.
- 1.42. Operator leave the field in blank for no changes, otherwise fill with new data.
- 1.43. Operator types Y

Operator: Jorge

Operation: Update provider

Enter the ID: 1

First Name: Jorge

Last Name: Ribeiro

Email: jriberio@pio.carrollu.edu

Cell Phone: 262 309 8725

Home Phone:

Work Phone:

Address: 135 S East Ave

Address Line 2: North Bergstrom Hall Room

City: Waukesha

State: WI

Zip Code: 53186

Leave the field in blank if you don't want to change.

First Name:

Last Name:

Email:

Cell Phone:

Home Phone:

Work Phone:

Address:

Address Line 2:

City:

State:

Zip Code:

Do you want to update this provider? (Y/N): ☒ Y

1.44. Test Table:

Test description	Input data	Expected results	Passed: Y N?
Fields are filled properly and operator types Y-y.	All provider requested information and Y	Updates provider record in the database.	
Operator types an invalid Zip Code.	12	Requests for a new Zip Code.	
Operator types a wrong state abbreviation.	ZZ	Requests for new US state abbreviation.	
Operator does not want to	N-n	Returns to provider operation screen without	

add the provider anymore.		updating anything in the database.	
Operator types a different letter than (Y-N) in the question.	i	Shows message: Invalid answer. Request for other letter input.	

11. Delete Provider

- 1.45. Operator types the provider ID
- 1.46. System displays all the provider information.
- 1.47. Operators types Y and deletes the provider from the system.

Operator: Jorge

Operation: Delete provider

Enter the ID: 1

First Name:	Jorge
Last Name:	Ribeiro
Email:	jriberio@pio.carrollu.edu
Cell Phone:	262 309 8725
Home Phone:	
Work Phone:	
Address:	135 S East Ave
Address Line 2:	North Bergstrom Hall Room
City:	Waukesha
State:	WI
Zip Code:	53186

Do you want to delete? (Y/N): Y

1.48. Test Table:

Test description	Input data	Expected results	Passed: Y N?
Operator types invalid provider ID	83729	Show message: invalid provider ID.	
Operator types valid provider ID	123456	Display all provider information	
Operator types Y-y to the delete question.	Y-y	System deletes the provider record from the database	
Operator types N-n to the	N-n	System redirects operator	

delete question		to provider operation screen	
Operator types invalid letter for the delete question.	3	System display message: invalid input and asks for the answer again.	

12. Main Screen

- 1.49. Manager confirms that he or she wants to see the report.
- 1.50. Manager confirms that he or she wants to generate a txt file.

Manager: Alberto

Do you want to visualize the weekly report? (Y/N) : ☒ Y

Current date and time: 11/20/2015 - 02:55pm

Date service was provided: 11/20/2015

Provider Number: 334567346

Member Number: 786654560

Service code: 232457

Comments:

Current date and time: ...

Date service was provided: ...

Provider Number: ...

Member Number: ...

Service code: ...

Comments: ...

Current date and time: ...

Date service was provided: ...

Provider Number: ...

Member Number: ...

Service code: ...

Comments: ...

Do you want to generate a txt file? (Y/N): ☒ Y

1.51. Test Table:

Test description	Input data	Expected results	Passed: Y N?
Manager wants to visualize the report or generate the txt file.	Y-y	Shows the report in the first field and generates the txt file in the second field.	

Manager does not want to visualize the report anymore or generate the txt file.	N-n	Returns to login screen.	
Manager types a different letter than (Y-N) in the question.	i	Shows message: Invalid answer. Request for other letter input.	

13. Provider Main Screen

1.52. Provider types one operation.

Provider: Rafael

Choose the operation:

1. Check member's status
2. Register a service for a member
3. Check fee's total
4. Request Provider Directory

Option:

1.53. Test Table:

Test description	Input data	Expected results	Passed: Y N?
------------------	------------	------------------	-----------------

Provider types inexistent option.	98	Shows message: This operation is not valid.	
Provider types blank.	""	Shows message: Type a valid operation.	
Provider selects Check member's status.	1	Redirects to screen: Check member's status..	
Provider selects Register a service for a member.	2	Redirects to screen: Register a service for a member.	
Provider selects Check fee's total.	3	Redirects to screen: Check fee's total.	
Provider selects Request provider directory	4	Redirects to screen: Request provider directory.	

14. Check Member's Status

1.54. Provider types the member's number.

Provider: Rafael

Check member's status

Insert the ID: 234114565

1.55. Test Table:

Test description	Input data	Expected results	Passed: Y N?
Provider types an inexistent ID.	000000000	Shows message: This ID is inexistent or invalid.	
Provider types blank.	""	Shows message: Type a valid ID.	
Provider types a valid ID.	234114565	Redirects to screen that shows the member's status..	

15. Show Member's Status

1.56. Provider confirms that he or she wants to start a service.

Provider: Rafael

Status of the Member: Name of the Member
Member Validated!

Do you want to start a service? (Y/N):

1.57. Test Table

Test description	Input data	Expected results	Passed: Y N?
Provider wants to start a service with the member validated.	Y-y	Redirects to screen that asks for the date of service	
Provider does not want to start a service with the member validated or suspended	N-n	Returns to Provider Main Screen.	
Provider types a different letter than (Y-N) in the question.	i	Shows message: Invalid input. Request for other letter input.	

16. Request the Date of the Service Provided

1.58. Provider inputs the date of the service.

Provider: Rafael

Enter the date the service was provided:
(Format: MM/DD/YYYY)

1.59. Test Table:

Test description	Input data	Expected results	Passed: Y N?
Provider keys an invalid input.	AA-12-2312	Shows message: Invalid input Requests date again.	
Provider types blank.	""	Shows message: Type a valid date.	
Provider types a valid date.	09/15/2015	Redirects to screen Provider Directory	

17. Shows the Provider's Directory

1.60. Provider inputs the service number.

Provider: Rafael
Provider Directory
883948 - Aerobics exercise session \$39.99
598470 - Session with a dietitian \$29.99
556431 - xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx \$YY.YY
678555 - xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx \$YY.YY
223489 - xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx \$YY.YY
556431 - xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx \$YY.YY
678555 - xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx \$YY.YY
223489 - xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx \$YY.YY
556431 - xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx \$YY.YY
678555 - xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx \$YY.YY
223489 - xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx \$YY.YY

What service do you want to register?

1.61. Test Table:

Test description	Input data	Expected results	Passed: Y N?
Provider keys an invalid input.	BG4324K	Shows message: Invalid input. Requests service code again.	
Provider types blank.	""	Shows message: Type a valid service code.	
Provider types a valid service code.	883948	Redirects to screen Confirm service code	

18. Confirm Service Code

18.1. Provider confirms the service name.

Provider: Rafael

The service keyed was: Aerobics exercise session
Confirm? (Y/N) ☒ Y

18.2. Test Table:

Test description	Input data	Expected results	Passed: Y N?
Provider confirms the service name.	Y-y	Redirects to screen Comments.	
Provider does not confirm the service name	N-n	Returns to Provider Directory.	
Provider types a different letter than (Y-N) in the question.	i	Shows message: Invalid input. Request for other letter input.	

19. Comments

18.3. Provider adds comments to the service.

Provider: Rafael

Comments:

18.4. Test Table:

Test description	Input data	Expected results	Passed: Y N?
Provider types blank.	""	Provider does not want to add comments.	
Provider types some information.	"Service provided in less than 30 minutes. The member requested a second session for improve	Provider adds a comment to the service and is redirected to the screen	

	the results.”	Description of Service.	
--	---------------	-------------------------	--

20. Description of Service

18.5. Provider confirms the registration of the service.

Provider: Rafael

Current date and time 09/15/2015 02:32:45pm

Date service was provided 09/15/2015

Provider number 654765870

Member number 234114565

Service code 883948

Comments _____

Confirm registration of this service? (Y/N) ☒ Y

Registered successfully!

The fee to be paid for this service is \$15.99

18.6. Test Table:

Test description	Input data	Expected results	Passed: Y N?
Provider confirms the service name.	Y-y	Redirects to screen Verification Form.	

Provider does not confirm the service name	N-n	Returns to Provider Directory.	
Provider types a different letter than (Y-N) in the question.	i	Shows message: Invalid input. Request for other letter input.	

21. Verification Form

18.7. Provider fills a verification form of the service provided.

Provider: Rafael

Current date (MM/DD/YYYY): 09/15/2015
Current time (HH:MM am or pm): 2:31pm
Date service was provided: 09/15/2015
Member name: Robert de Niro
Member number: 234114565
Service code: 883948
Fee: \$39.99

18.8. Test Table:

Test description	Input data	Expected results	Passed: Y N?
------------------	------------	------------------	-----------------

Provider left a blank field in any of the required fields.	""	System asks for the provider fill the field again.	
Provider types an invalid date.	12/DD/2018	Requests for a new date.	
Provider types an invalid time.	2:ABpm	Requests for a new time.	
Provider fills all the fields.	Fields filled properly	Service completed. Returns to provider main screen.	

22. Check Fee's Total

18.9. Provider confirms the returning to menu.

Provider: Rafael
Fee's Total of current week
Date: 09/13/2015
Service: Session with a dietitian
Fee: \$11.99

Date: 09/15/2015
Service: Aerobics exercise session
Fee: \$15.99

Total: \$27.98

Return to menu? (Y/N) ☒ Y

18.10. Test Table:

Test description	Input data	Expected results	Passed: Y N?
Provider confirms the service name.	Y-y	Redirects to screen Verification Form.	
Provider does not confirm the service name	N-n	Returns to Provider Directory.	
Provider types a different letter than (Y-N) in the question.	i	Shows message: Invalid input. Request for other letter input.	

23. Request Provider Directory

18.11. Provider confirms the returning to menu.

Provider: Rafael

Provider Directory

883948 - Aerobics exercise session \$39.99
598470 - Session with a dietitian \$29.99
556431 - xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx \$YY.YY
678555 - xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx \$YY.YY
223489 - xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx \$YY.YY
556431 - xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx \$YY.YY
678555 - xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx \$YY.YY
223489 - xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx \$YY.YY
556431 - xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx \$YY.YY
678555 - xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx \$YY.YY
223489 - xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx \$YY.YY

Return to menu? (Y/N)

18.12. Test Table:

Test description	Input data	Expected results	Passed: Y N?
Provider confirms the service name.	Y-y	Redirects to screen Verification Form.	
Provider does not confirm the service name	N-n	Returns to Provider Directory.	
Provider types a different letter than (Y-N) in the question.	i	Shows message: Invalid input. Request for other letter input.	