SOFTWARE REQUIREMENTS ANALYSIS AND SPECIFICATION DOCUMENT

Software Engineering

Supervisor:
Prof. Matteo Giovanni Rossi

Authors:	ID:
Ahmed Ibrahim Abdelrazzak Hamed	10682755
Khaled Said Ahmed Maamoun	10696857
Mahmoud Mohamed Aboelwafa Medany	10715340

Table of Contents

List Of Tables *1*

Table Of Figures *2*

1. Introduction *3*

1.1 Objective **3**

1.2 Overview **3**

1.3 Acronyms **3**

2. Specific Requirements *4*

2.1 Functional Requirements **4**

2.1.1 Use Case Diagram 4

2.1.2 Requirements 5

2.2 Non-Functional Requirements **7**

3. UML- CLASS MODEL *8*

4. SEQUENCE MODELS *9*

List Of Tables

TABLE 1 - USE CASE FOR USRS SIGNING UP 9
TABLE 2 - USE CASE FOR USRS SIGNING IN 10
TABLE 3 - USE CASE FOR USRS SELECTING A SERVICE 11
TABLE 4 - USE CASE FOR USRS GETTING A SERVICE HISTORY 12
TABLE 5 USE CASE FOR USRS EVALUATING A PROPOSAL 13
TABLE 6 - USE CASE FOR USRS PAYING FOR A SERVICE 14
TABLE 7 - USE CASE FOR USRS REQUESTING A CLAIM 15
TABLE 8 USE CASE AYLIS SIGNUP 16
TABLE 9 - USE CASE FOR AYLIS SIGNING IN 17
TABLE 10 - USE CASE FOR AYLIS EVALUATING A "SOR" 18
TABLE 11 - USE CASE FOR AYLIS SENDS A PROPOSAL 19
TABLE 12 - USE CASE FOR A USRS ADDS A BROADCAST 20

Table Of Figures

FIGURE 1 USE CASE DIAGRAM....4
FIGURE 2 UML- CLASS MODEL....8
FIGURE 3 – SEQUENCE MODEL OF A USRS SIGNING UP....9
FIGURE 4 – SEQUENCE MODEL OF A USRS SIGNS IN....10
FIGURE 5 - SEQUENCE MODEL OF A USRS SELECTS A SERVICE....11
FIGURE 6 - SEQUENCE MODEL OF A USRS GETS A SERVICE HISTORY....12
FIGURE 7 - SEQUENCE MODEL OF A USRS EVALUATING A PROPOSAL....13
FIGURE 8 - SEQUENCE MODEL OF A USRS PAYING FOR A SERVICE....14
FIGURE 9 - SEQUENCE MODEL OF A USRS REQUESTING A CLAIM....15
FIGURE 10 SEQUENCE MODEL OF AYLIS SIGNUP....16
FIGURE 11 - SEQUENCE MODEL OF AYLIS SIGNING IN....17
FIGURE 12 - SEQUENCE MODEL OF AYLIS EVALUATING "SOR"....18
FIGURE 13 - SEQUENCE MODEL OF AYLIS SENDS A PROPOSAL....19
FIGURE 14 - SEQUENCE MODEL OF A USRS SENDING A BROADCAST....20

1. Introduction

1.1 Objective

The objective of this report is to present a Start Up company named “AYLI” as an application on the web browser. The main target for AYLI is to improve the relation between customers and the entertainment and accommodation hubs, including hotels, hostels, restaurant. AYLI acts as an agency model. It is an intermediary agent between people who want to book an entertainment service and the different companies who offer these kinds of services.

The upcoming documentations are description of the system technicality and is done through Use Case diagrams, Class diagrams and Sequence diagrams.

1.2 Overview

This document gives the detailed specifications for the system. It is organized as follows:

- Specific Requirements:

Each functional requirement gives a desired behavior for the system, a business justification, and a measure to determine if the final system has successfully met the objective. These objectives are organized in a sequential order.
The non-functional requirements section is organized by category. Each objective specifies a technical requirement or constraint on the overall characteristics of the system.

- Class Model:

A class is a collection of objects in the system that have the same data and behavior. All analysis classes and their relationships are shown on the class diagram from a high-level point of view.

- Sequence Model:

A sequence depicts interaction between objects in a sequential order i.e., the order in which these interactions take place. The most relevant interactions of the objects, according to the criteria of the authors, are shown in the sequence diagram.

1.3 Acronyms

- AYLI: As You Like It
- AYLIs: AYLI Providers
- USrs: AYLI’s customers who wants to receive a service
- Administrator: Administrator Interface
- SOR: Service order request
- SON: Service order number
- AR: Admission Request

2. Specific Requirements

2.1 Functional Requirements

2.1.1 Use Case Diagram

Functionalities are necessary for the interaction of USrs, AYLIs and/or Administrator with the system. After signing in into the system, actors, either they are USrs, AYLIs or Administrators, will have the ability to access all the operations enabled for their profiles. Figure 1 shows the actors and their functionalities. In addition, non-functionalities, such as web reliability and web security, are very important for the satisfaction of AYLI's customers.

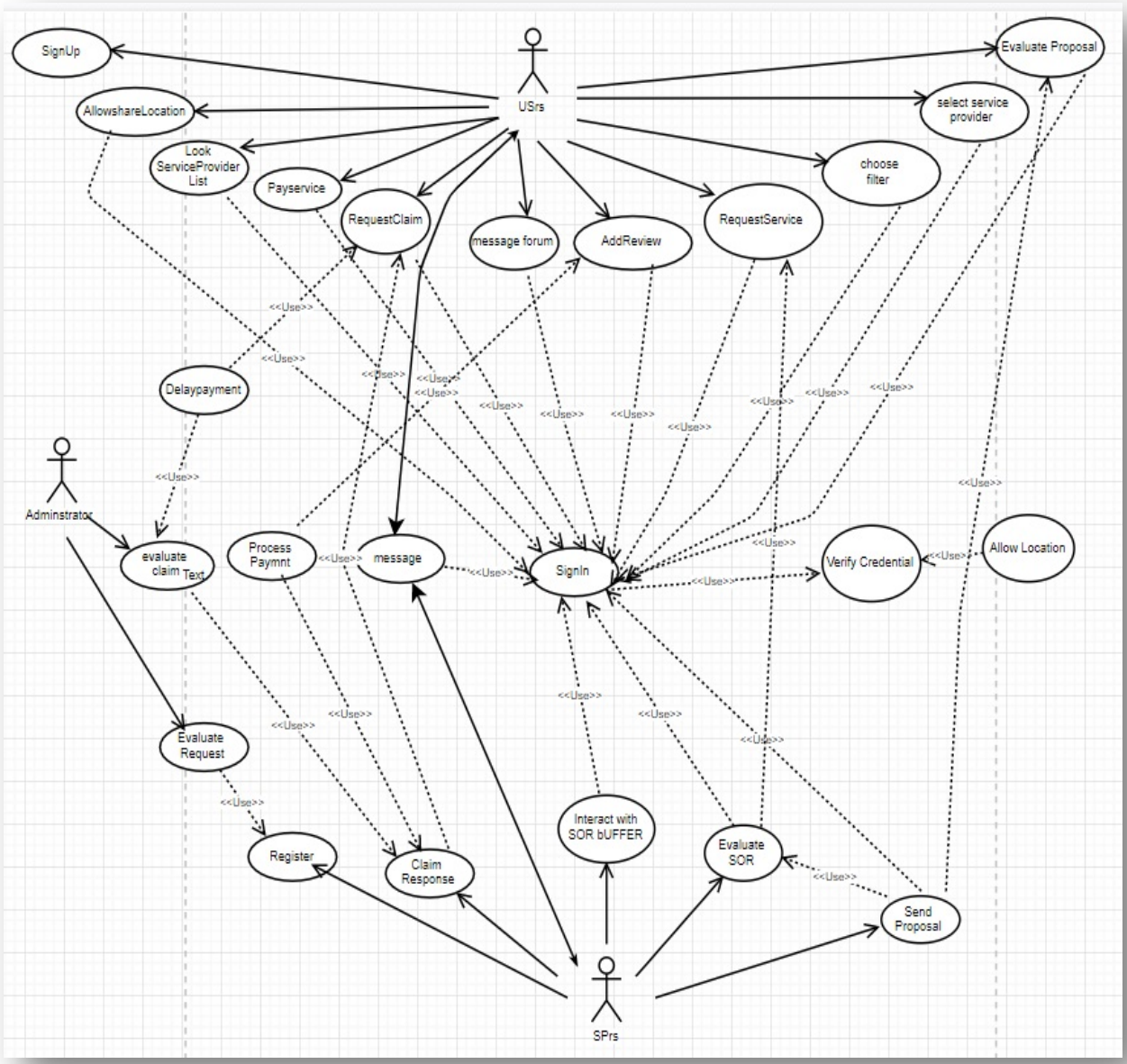

Figure 1 USE CASE DIAGRAM

2.1.2 Requirements

This section relates the requirements to the different actors:

ACTOR: Usrs

- **Sign Up**

  By inserting the first name, the surname, email address, a password of minimum 6 characters and the phone number, the system should allow the actor to create an account. Once this process is finished, the USrs gets a notification.

- **Sign In**

  The customer fills in the email and password. The system would either verify the actor's credentials if they are correct or deny them if not correct.
  If correct, the actor will get a notification which states if he would agree to provide his own location or not.

- **AYLI Map**

  AYLI map would be shown to the customer in which he could easily find the address of an AYLIs. The customer would be able to make filters to his selected service such as the price, distance in km and the services rating. And as a result, he/she would be allowed by AYLI to know all what he needed to know about a service, he/she wishes to spend his money at, such as the AYLIs's name, telephone number, etc.

- **Request A Service**

  Once a USrs select a specific service, from an AYLIs from existing on the map. A Service Order Request (SOR) which includes the order number, the personal information of the actor would be generated. The system should send a notification to the selected AYLIs with the generated SOR.

- **Service History**

  The customer could check by himself all his selected services through AYLI.

- **Proposal Evaluation**

  Customer would be able to read and accept a proposal per a service order request. And upon accepting the proposal, a confirmation message is sent to AYLIs, and an invoice is generated. And also, vice versa. As upon rejecting the proposal, a confirmation message is sent. The system should allow the customer to reject a proposal. When the proposal is rejected, the system should return a rejection message. The system should eliminate all the proposals that have been rejected by AYLIs.

- **To Pay**

  Payment is made online through providing the credit or debit card of the actor along with providing the name of the holder of the card, the expiry date and the CVV code.

- **Service Rating**

  The Customer would be able to rate the service after a day of using it from 1 to 5 out of 5. And once, the actor gives a rate of at least 2 out of 5, the payment is processed. And if the actor gives a rating of 1, automatically, AYLI's customer service will reach him by email to solve the issue by providing same service in another place or issue him a Refund.

ACTOR: AYLIs

- **Declaration Request & Signing in**

AYLIs would be able to declare that it's a business company in order to join AYLI's website by providing its name, the services which offer, tax code, the company's email and a password in which after it would be able to access the system by entering its email and password. And a confirmation message would be sent upon that.

- **Service Order Request Evaluation**

A confirming email would be sent to the customer once the AYLIs approves the order request. Or else, a rejecting email would be sent when the system would eliminate the order request when it is rejected by all service centers related to this kind of order.
Send Proposal
AYLIs would be able to send a proposal upon an order request and a confirmation message would be sent to Administrator as a result, who will notify the customer with the proposal or eliminate it if the customer does not accept it.

- **Response to USrs Complaints**

AYLIs would be able to reply to the complaints of the customers once the administrator confirms it after which a confirmation email is sent
Communication with AYLIs.
AYLIs would be allowed by the system to put an order request in the "BUFFER" section if the service is rejected. And vice versa. As if the service would be used,
The system should allow AYLIs to take a SOR given in the "buffer" of the platform in case the service could be fulfilled.

- **Post a broadcast**

Any AYLIs will have the possibility to post an advertisement using broadcast function, where pictures and description can be added.

ACTOR: Administrator

- **Sign Up**

The Administrator would be allowed by the system to read the declaration requests done by the AYLIs and whether to accept or reject the requests and upon it, a confirmation notification would be sent to AYLIs.

- **Complaints**

The Administrator would be able to read the complaints from the USrs and the replies from AYLIs and take an action regards the situation.

2.2 Non-Functional Requirements

- **Reliability & Availability**
  The system functions the whole year, 24 hours in 365 days, guaranteeing 100 % of Availability.

- **Web Accessibility**
  Web accessibility means that websites, tools, and technologies are designed and developed so that people with disabilities can use them. Anyone can: perceive, understand, navigate, and interact with the Web, contribute to the Web. The web site is designed and developed so that all people can use very easily. Anyone who uses a web browser would easily access AYLI's Website.

- **Web Performance**
  The web site assets would be loaded in a sensible order, so that the user can start to use it very quickly.

- **Web Security**
  The system would prevent any sorts of attacks. It would protect AYLI's website from unauthorized access, use, modification, destruction, or disruption.

- **Web Compatibility**
  AYLI is available to function on all different web browsers in the market.

3. UML- CLASS MODEL

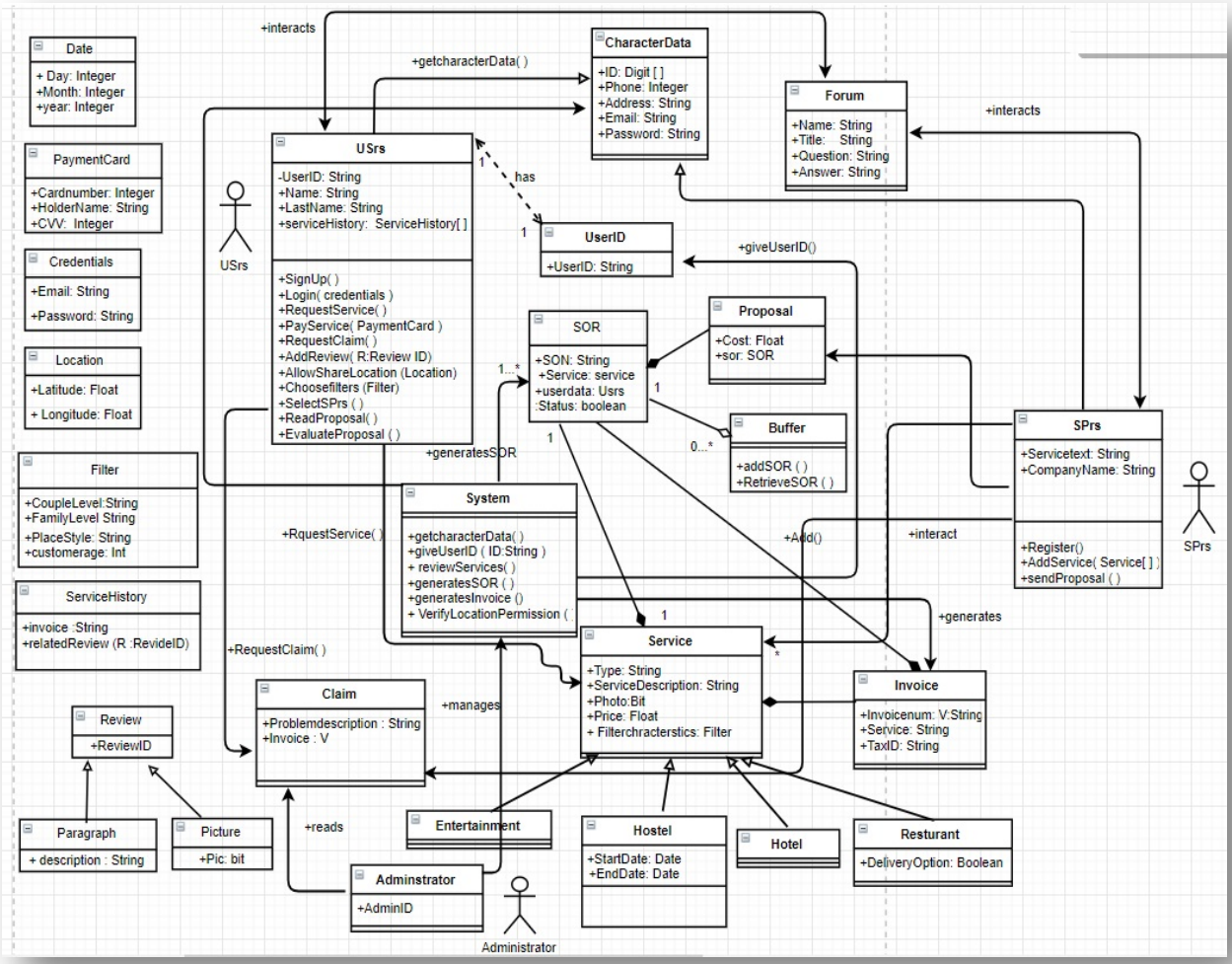

Figure 2 UML- CLASS MODEL

4. SEQUENCE MODELS

4.1 USrs SIGNS UP

Table 1 - USE CASE for USrs Signing Up

Name:	USrs Signs Up
Use Case:	Case 1
Actors:	USrs
Pre-Conditions	USrs must be on the “sign up” interface for USrs.
Flow of Events:	1. USrs fills out all mandatory fields. 2. USrs clicks “Sign Up”. 3. The system shows “A Sign Up process is confirmed”.
Post-Conditions:	USrs is registered and is able to sign in using his credentials.
Exception:	1. USrs is already registered. 2. USrs doesn’t fill all mandatory fields. 3. USrs enters invalid data.

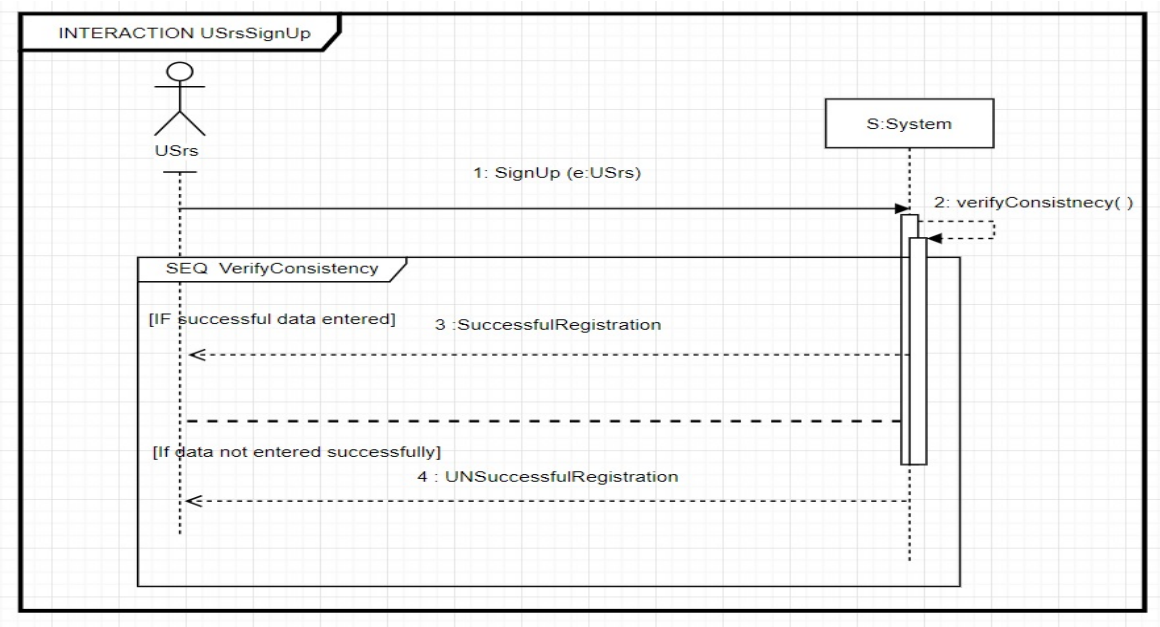

Figure 3 – SEQUENCE MODEL of A USrs Signing Up

4.2 USrs SIGNS IN

Table 2 - USE CASE for USrs Signing In

Name:	USrs Signs In
Use Case:	Case 2
Actors:	USrs
Pre-Conditions	USrs must be on main page of the website.
Flow of Events:	1. USrs fills out credential's fields. 2. USrs clicks "Login". 3. The system verifies USrs credentials. 4. The system shows "You are allowed to sign in" message. 5. The system asks for USrs location permission. 6. The system clicks on accept.
Post-Conditions:	USrs is in the home page of the website.
Exception:	1. USrs enters invalid data. 2. USrs doesn't fill all mandatory fields.

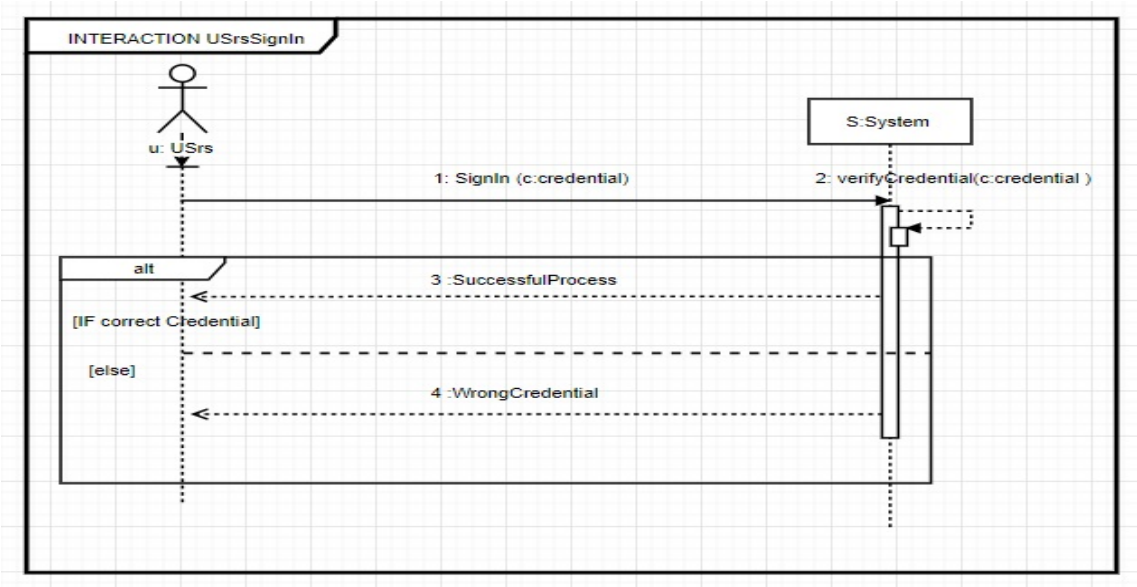

Figure 4 – SEQUENCE MODEL Of A USrs Signs In

4.3 USrs SELECTS A SERVICE

Table 3 - USE CASE for USrs Selecting A Service

Name:	USrs selects a service.
Use Case:	Case 3
Actors:	USrs
Pre-Conditions	USrs must be on the home page of the website. USrs must have at least one service selected.
Flow of Events:	1. USrs Select an AYLIs. 2. USrs fills all mandatory fields. 3. The system shows "Successful Registration" message.
Post-Conditions:	USrs is in the home page of the website.
Exception:	1. USrs enters invalid data. 2. USrs doesn't fill all mandatory fields.

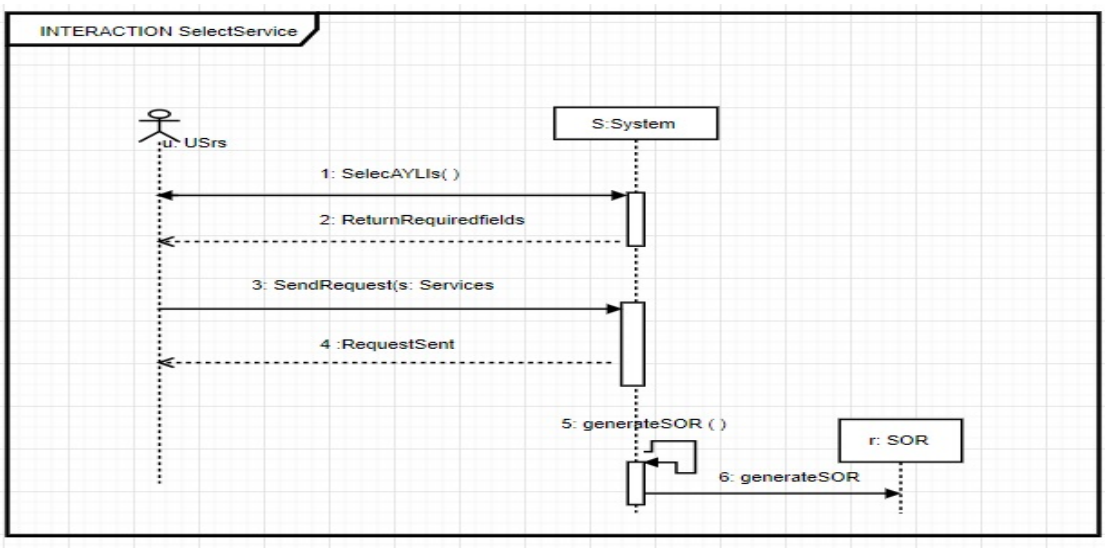

Figure 5 - SEQUENCE MODEL Of A USrs Selects A Service

4.4 USrs GETS A SERVICE HISTORY

Table 4 - USE CASE for USrs getting A Service History

Name:	USrs gets a service history.
Use Case:	Case 4
Actors:	USrs
Pre-Conditions	USrs must be on the home page of the website. USrs must have at least one service selected.
Flow of Events:	1. USrs clicks “History”. 2. The system returns a list of services
Post-Conditions:	An interface with all the services requested by the USrs.
Exception:	-

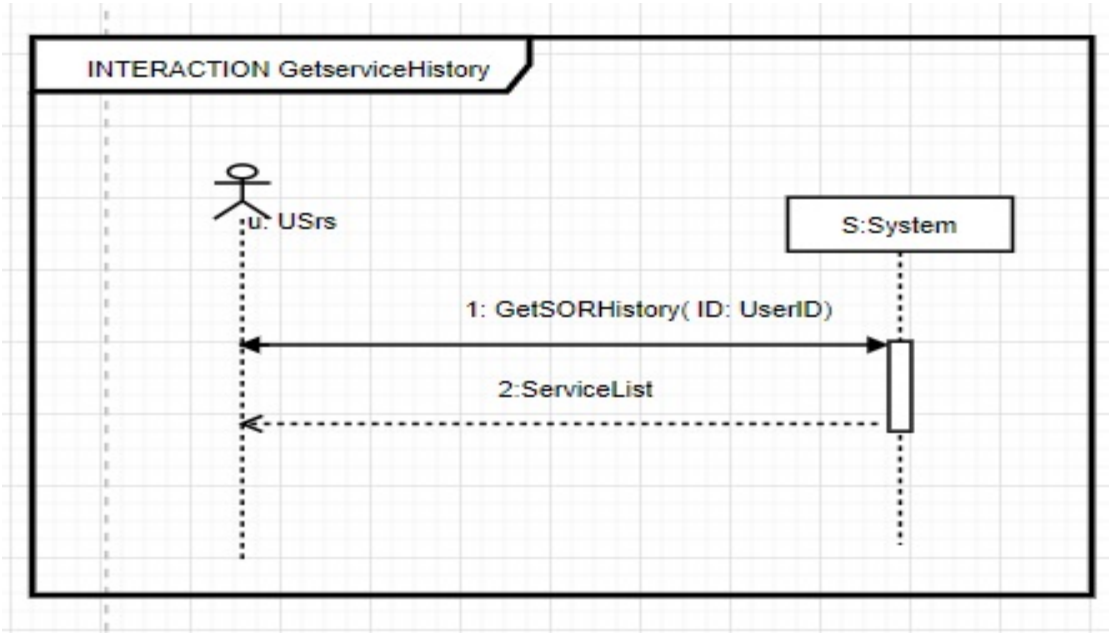

Figure 6 - SEQUENCE MODEL Of A USrs Gets A Service History

4.5 USrs EVALUATES A PROPOSAL

Table 5 USE CASE for USrs Evaluating A Proposal

Name:	USrs evaluate a proposal.
Use Case:	Case 5
Actors:	USrs
Pre-Conditions	USrs must be on the home page of the website. USrs have already sent a request. USrs must have a proposal to evaluate.
Flow of Events:	1. USrs clicks on the proposal that wants to evaluate. 2. USrs read the proposal. 3. USrs clicks on "Confirm" to accept the proposal. 4. The system shows "proposal accepted" message.
Post-Conditions:	The proposal now can be paid and later on rated.
Exception:	-

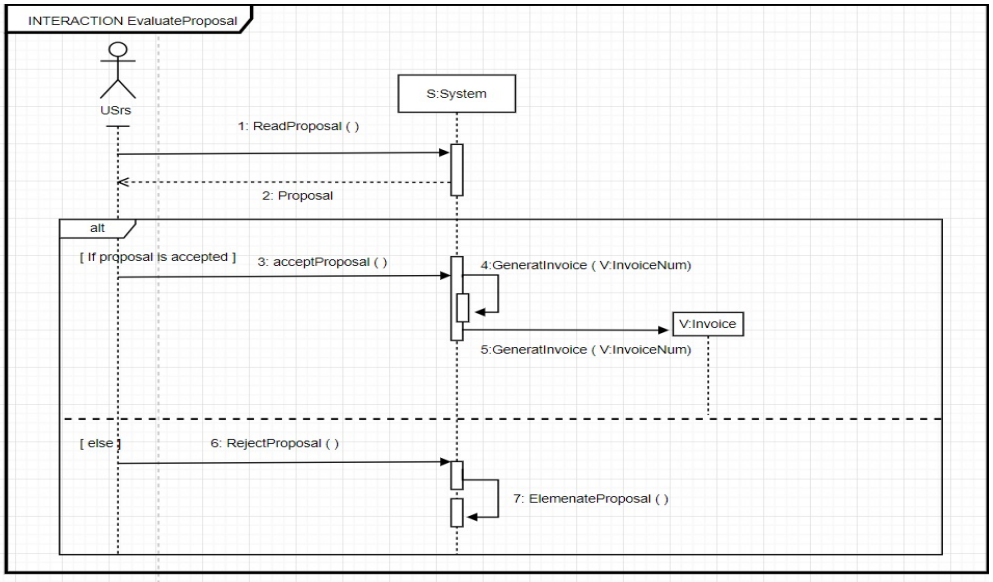

Figure 7 - SEQUENCE MODEL Of A USrs Evaluating A Proposal

4.6 USrs PAYS FOR A SERVICE

Table 6 - USE CASE for USrs Paying for a service

Name:	**USrs Pays for a service.**
Use Case:	Case 6
Actors:	USrs
Pre-Conditions	USrs must have an approved invoice. USrs must be on the approved invoice interface.
Flow of Events:	1. USrs fills out all the information of the credit card 2. USrs clicks on "Pay" 3. The system saves the information of the credit card 4. The system shows "Payment done" message 5. The system sends a message according to the delivery conditions
Post-Conditions:	USrs is in the home page of the home page.
Exception:	1. USrs enters invalid data. 2. USrs doesn't fill all mandatory fields.

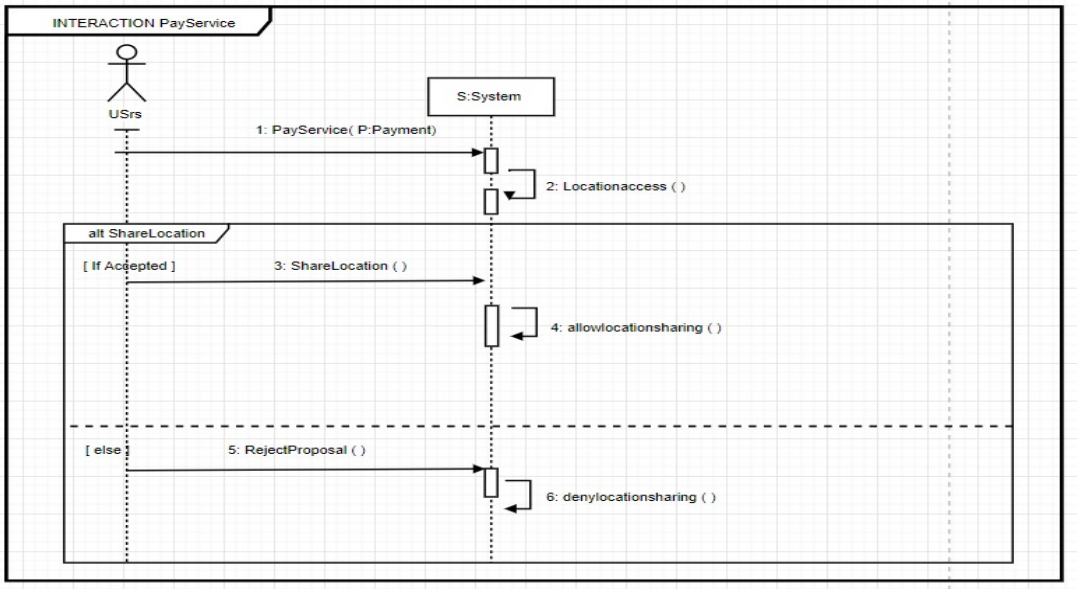

Figure 8 - SEQUENCE MODEL Of A USrs Paying for a service

4.7 USrs REQUESTS A CLAIM

Table 7 - USE CASE for USrs Requesting A Claim

Name:	USrs requests a claim.
Use Case:	Case 7
Actors:	USrs
Pre-Conditions	USrs must be on the page for contact the administrator. USrs must a service done; this means an invoice generated in the system.
Flow of Events:	1. USrs fills all mandatory fields. 2. USrs clicks "Send". 3. The system shows "Successful Registration" message. 4. The system delays the payment of the service.
Post-Conditions:	The system save the information of the service and it is related to the USrs account.
Exception:	1. USrs enters invalid data. 2. USrs doesn't fill all mandatory fields. 3. USrs does not have any invoice generated in the system.

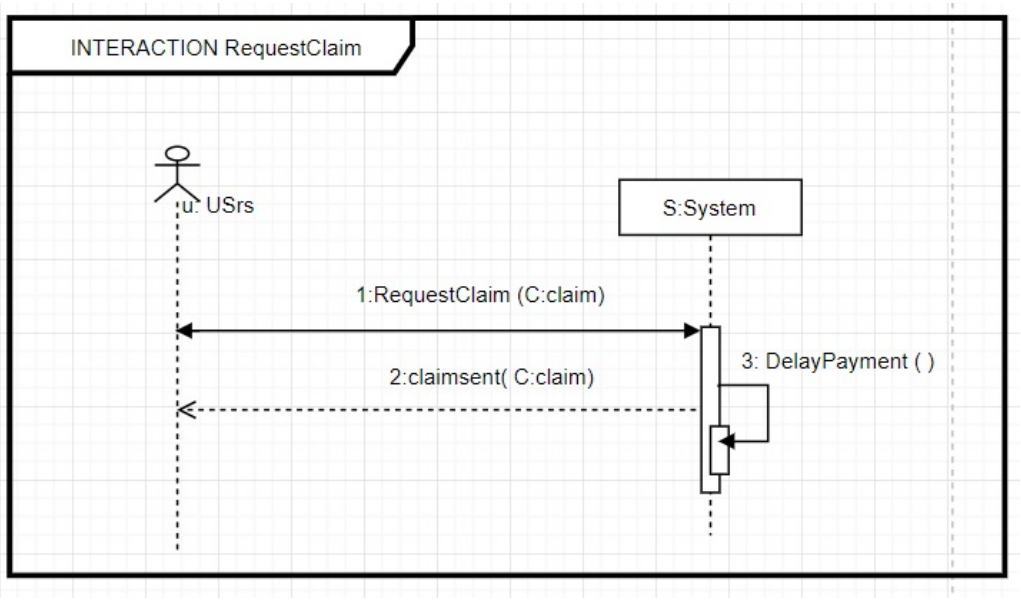

Figure 9 - SEQUENCE MODEL Of A USrs Requesting A Claim

4.8 AYLIs SIGNS UP

Table 8 USE CASE AYLIs Sign Up

Name:	AYLIs Signs Up
Use Case:	Case 8
Actors:	AYLIs
Pre-Conditions	AYLIs must be on the "sign up" interface for AYLIs.
Flow of Events:	1. AYLIs fills out all mandatory fields. 2. AYLIs clicks "Sign Up". 3. The system shows "A Sign Up process is confirmed".
Post-Conditions:	AYLIs is registered and is able to sign in using his credentials.
Exception:	1. AYLIs is already registered. 2. AYLIs doesn't fill all mandatory fields. 3. AYLIs enters invalid data.

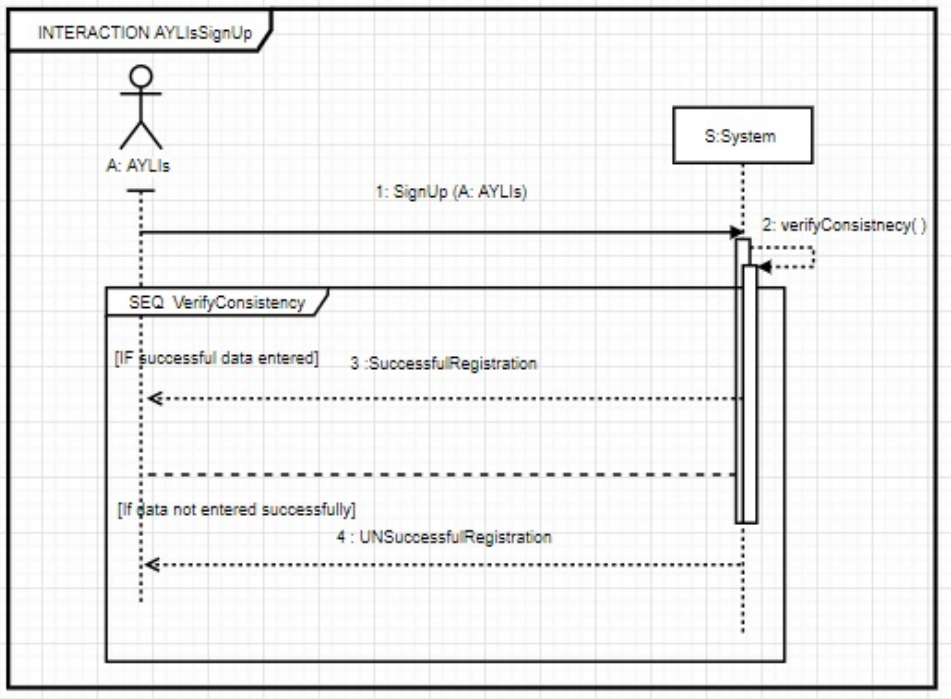

Figure 10 SEQUENCE MODEL OF AYLIs SignUp

4.9 AYLIs SIGNS IN

Table 9 - USE CASE for AYLIs SIGNING IN

Name:	AYLIs signs in.
Use Case:	Case 9
Actors:	AYLIs
Pre-Conditions	AYLIs must be on the "sign in" interface for AYLIs.
Flow of Events:	1. AYLIs fills out credential's fields. 2. AYLIs clicks "Login". 3. The system verifies AYLIs credentials. 4. The system shows "Successful Sign In" message. 5. The system asks for AYLIs location permission. 6. The system clicks on accept.
Post-Conditions:	AYLIs is in the home page for AYLIs.
Exception:	1. AYLIs enters invalid data. 2. AYLIs doesn't fill all mandatory fields.

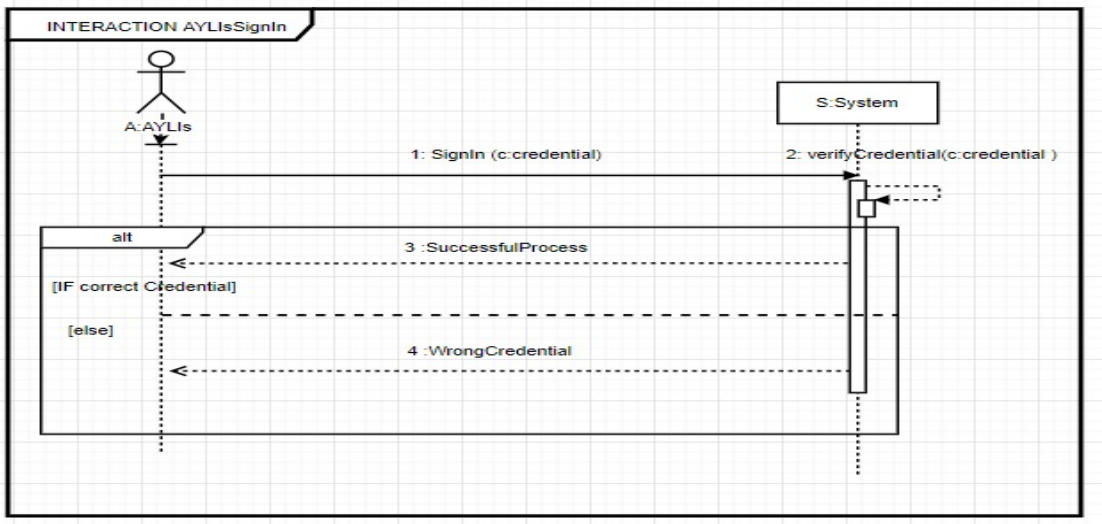

Figure 11 - SEQUENCE MODEL Of AYLIs SIGNING IN

4.10 AYLIs EVALUATES A “SOR”

Table 10 - USE CASE for AYLIs Evaluating A "SOR"

Name:	AYLIs evaluates a “SOR”.
Use Case:	Case 10
Actors:	AYLIs
Pre-Conditions	AYLIs must be in the home page for AYLIs. AYLIs must have at least one SOR sent by USrs.
Flow of Events:	1. AYLIs clicks “SOR” 2. AYLIs reads the received SOR. 3. AYLIs clicks “Confirm”. 4. The system shows “request accepted” message.
Post-Conditions:	AYLIs is in the home page for AYLIs.
Exception:	1. AYLIs enters invalid data. 2. AYLIs doesn’t fill all mandatory fields.

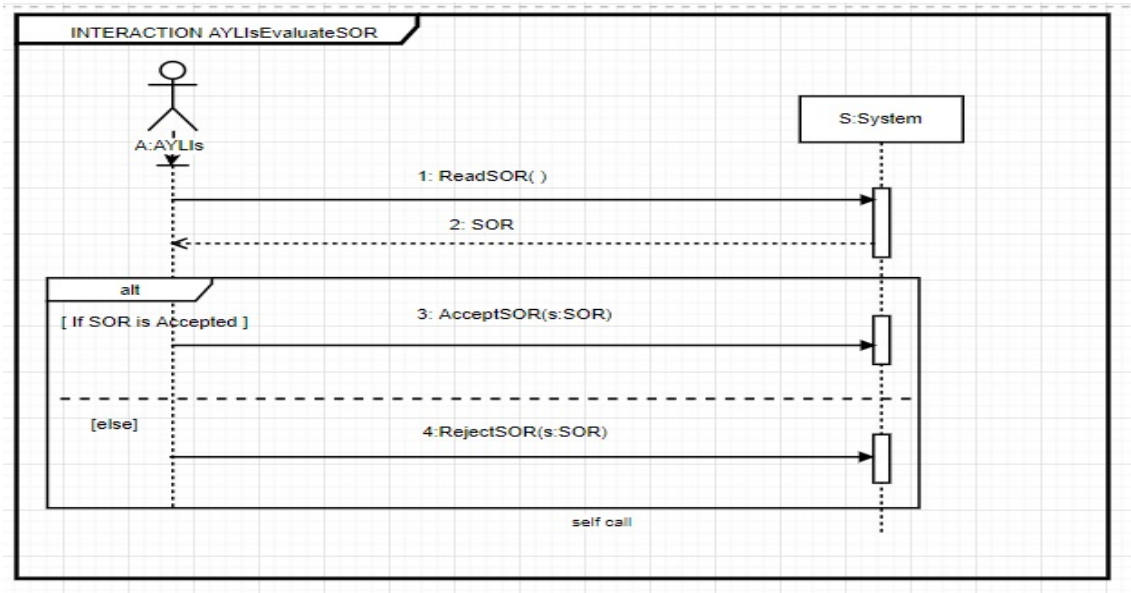

Figure 12 - SEQUENCE MODEL Of AYLIs EVALUATING "SOR"

4.11 AYLIs SENDS A PROPOSAL

Table 11 - USE CASE for AYLIs Sends A Proposal

Name:	AYLIs sends a proposal.
Use Case:	Case 11
Actors:	AYLIs
Pre-Conditions	AYLIs must be in proposal page. AYLIs must have at least one SOR accepted.
Flow of Events:	1. AYLIs fills out all the mandatory fields. 2. AYLIs clicks "Create". 3. The system shows "Successful Proposal" message. 4. The system generates a proposal.
Post-Conditions:	AYLIs is on its home page.
Exception:	1. AYLIs enters invalid data. 2. AYLIs doesn't fill all mandatory fields.

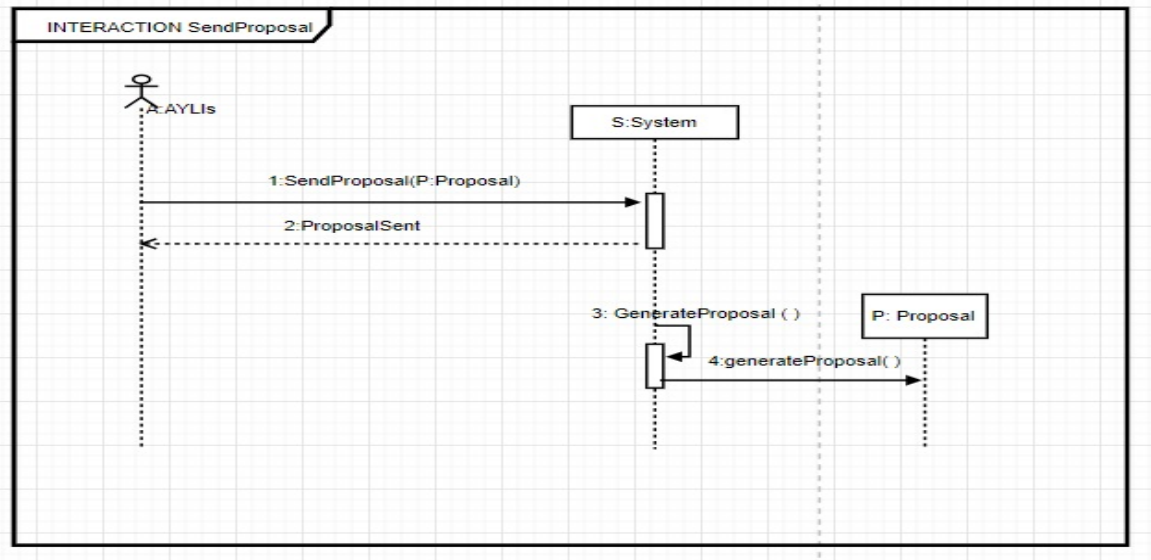

Figure 13 - SEQUENCE MODEL Of AYLIs Sends A Proposal

4.12 AYLIs ADDS A BROADCAST

Table 1212 - USE CASE for A USrs Adds A Broadcast

Name:	USrs adds a broadcast.
Use Case:	Case 12
Actors:	AYLIs
Pre-Conditions	AYLIs must be signed in.
Flow of Events:	1. AYLIs clicks “Broadcast”. 2. AYLIs fills all mandatory fields. 3. The system Register the broadcast message.
Post-Conditions:	AYLIs on the broadcast page
Exception:	1. AYLIs doesn’t fill all mandatory fields.

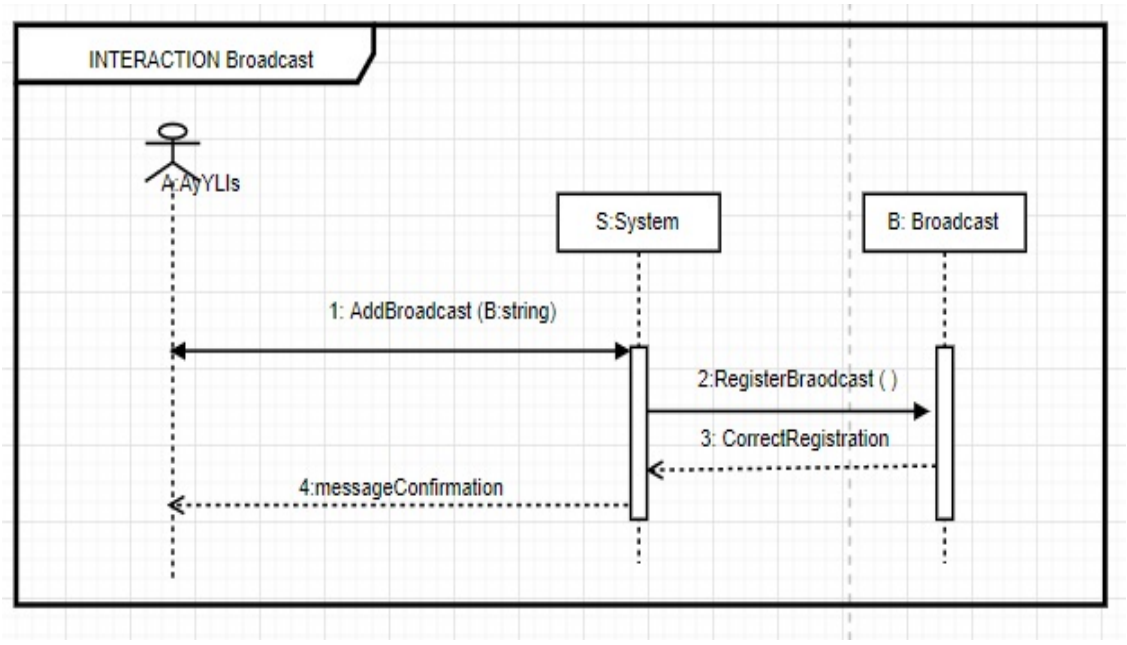

Figure 1414 - SEQUENCE MODEL Of A USrs Adds A Broadcast