OCTOBER 26, 2022

SENG 696 GROUP 10

ASSIGNMENT 2

I Document Control

I.I Authors

SN#	UCID	Name
1	30146191	Aadharsh Hariharan
2	30200639	Ali Dehaghi
3	30088441	Tariq Al Shoura

Table of Contents

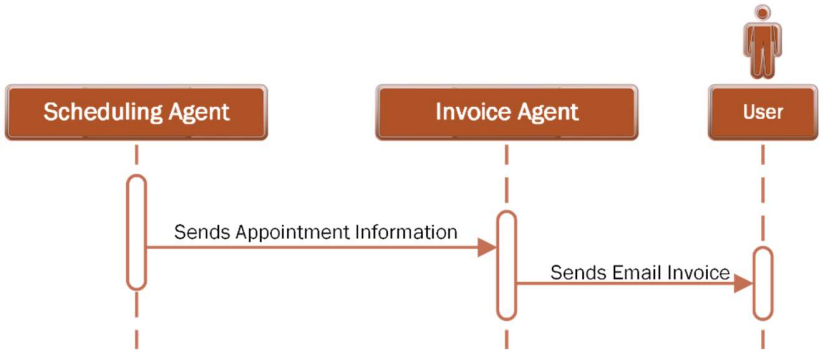
1 Detailed Design

1.1 Use Cases

1.1.1 Notification Use Cases Definition

Brief description	The system uses user information to send a notification (email) with the appointment details (invoice)	
Precondition(s)	The user is already registered and has an appointment	
Postcondition(s)	If preconditions are met, an automated email with the invoice details will be sent to the user	
Process Steps		
1	The invoice agent receives the user information and appointment information once the appointment is booked	
2	The invoice agent packs all the information into a structured invoice format and sends the invoice to the user in the form of an email.	
Exceptions:		
1a	The system is not reachable at this moment	Ignore exception
Relationships:		
Initiating	Scheduling Agent	
Collaborating	Invoice Agent	
Other Diagrams:		
Data Requirements:		
Data Required	User's email Appointment details	

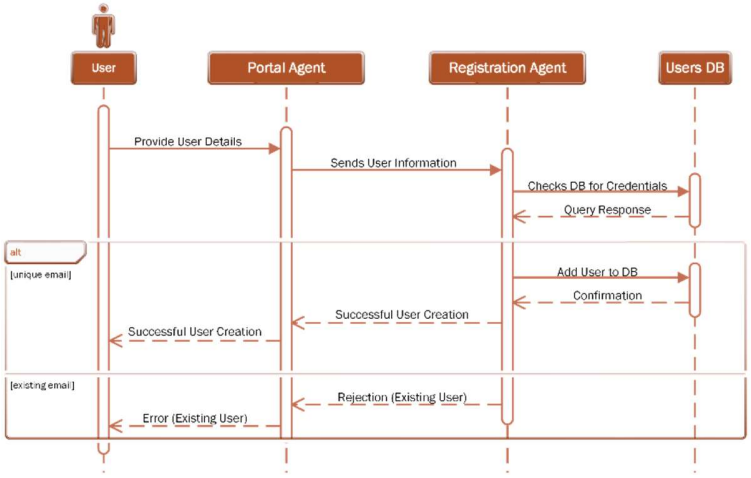
1.1.2 Notification Use Cases Sequence Diagram



1.1.3 Registration Use Cases Definition

Brief description	The system requires a user to be registered for the user to be able to use the system and services	
Precondition(s)	User's credentials should be unique and not present on the database	
Postcondition(s)	If preconditions are met the database should create a new record for the user	
Process Steps		
1	The user puts the required information like "name", "email", "password", …etc. on the portal	
2	By submitting the information, the Portal agent will package the information and sends it to the Registration agent	
3	The Registration agent will communicate with the Users database to check whether the user information is unique or not	
4(a)	If the email is unique, the Registration agent will create a record for the user in the database	
5(a)	The Registration agent will send a “success” feedback Portal agent	
6(a)	The Portal agent will display a “success” message to the user	
4(b)	If the email exists, in the database the Registration agent will send a “failure” feedback Portal agent	
5(b)	The Portal agent will display a “failure” message to the user	
Exceptions:		
1a	The system is not reachable at this moment	The system shows an error message to the user
4(a)a	The system cannot allocate a record due to already existing user	The system shows "There is a user with this email, please log in"
Relationships:		
Initiating	User	
Collaborating	Portal Agent, Login Agent	
Other Diagrams:		
Data Requirements:		
Data Required	First Name Last Name Email Address Phone Password	

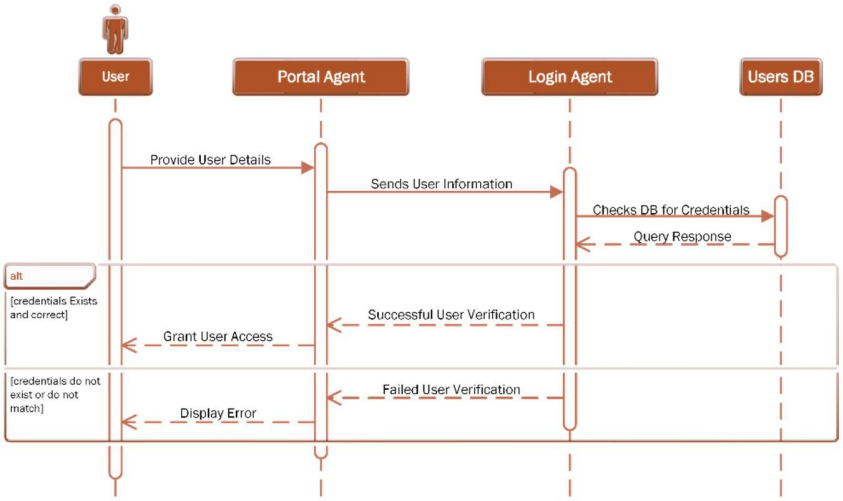
1.1.4 Registration Use Cases Sequence Diagram



1.1.5 Login Use Cases Definition

Brief description	The system requires the user's information to the user to be able to use the system and services	
Precondition(s)	The user must be registered before using the system	
Postcondition(s)	If preconditions are met access to the services should be given	
Process Steps		
1	The user provides the required information like "email", and "password" on the portal to be authenticated	
2	By submitting the information, the Portal agent will package the information and sends it to the Login agent	
3	The Login agent will communicate with the Users database to check whether the provided information matches any existing record in the database	
4(a)	If the credentials exist, the Registration agent will send "success" feedback to the Portal agent	
5(b)	The Portal agent will give the user access to the system	
4(b)	If the credentials do not exist, the Registration agent will send a "failure" to the feedback Portal agent	
5(b)	The Portal agent will display a "failure" message to the user	
Exceptions:		
1a	The system is not reachable at this moment	The system shows an error message to users
4(a)a	The system cannot allow logging in because the password is wrong	The system shows "Entered password is incorrect, please try again"
4(a)b	The system cannot find the user with the given credentials	The system shows "User cannot be found, please sign up"
Relationships:		
Initiating	User	
Collaborating	Portal Agent, Registration Agent	
Other Diagrams:		
Data Requirements:		
Data Required	Email (Username) Password	

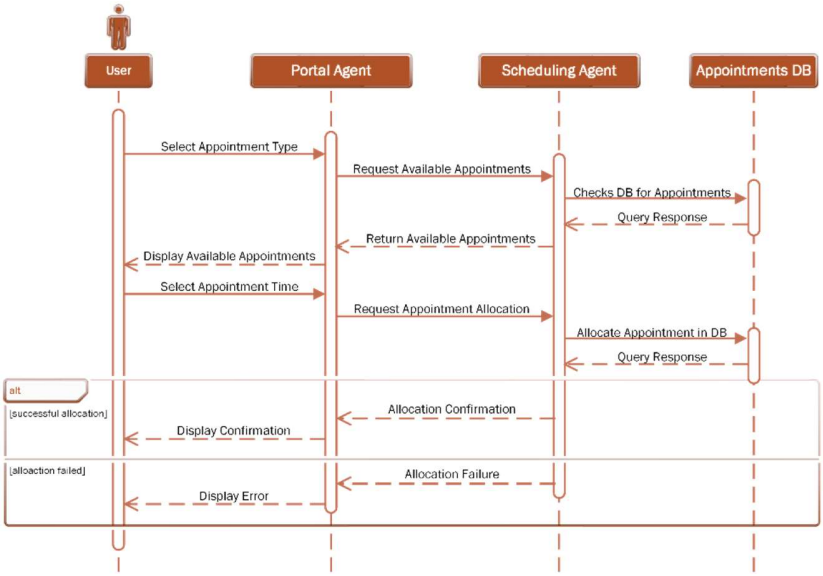
1.1.6 Login Use Cases Sequence Diagram



1.1.7 Appointment Use Cases Definition

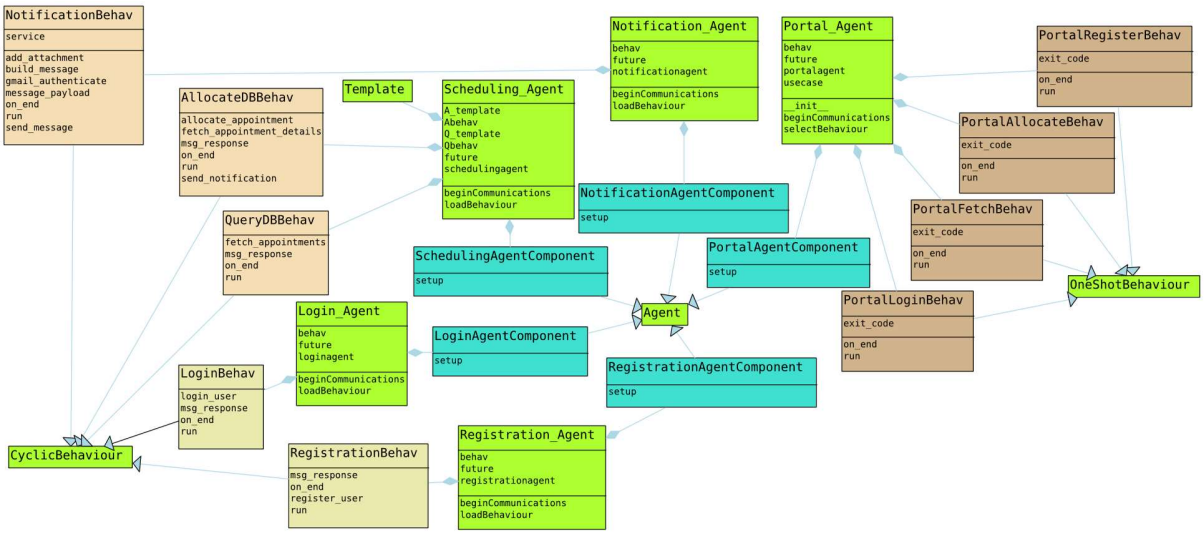
Brief description	users can book an appointment based on their preferences (time, doctor, specialization)	
Precondition(s)	the user is already registered	
Postcondition(s)	if preconditions are met the preferred appointment can be booked.	
Process Steps		
1	The user selects the required appointment type	
2	By submitting the information, the Portal agent will package the information and sends it to the Scheduling agent	
3	The Scheduling agent communicates with the Appointments database and fetches all available appointments matching the required type	
4	The Scheduling agent packages the information and returns it to the Portal agent	
5	The Portal agent will display the available information to the user	
6	The user selects the desired appointment time	
7	By submitting the information, the Portal agent will package the information and sends it to the Scheduling agent	
8	The Scheduling agent communicates with the Appointments database and allocates/update the selected appointment	
9(a)	If the appointment is successfully allocated, the Registration agent will send “success” feedback to the Portal agent	
10(a)	The Portal agent will give the user access to the system	
9(b)	If the appointment could not be allocated, the Registration agent will send “failure” feedback to the Portal agent	
10(b)	The Portal agent will display a “failure” message to the user	
Exceptions:		
1a	System is not reachable at this moment	System shows a error message to users and let them know
9(a)a	System tries to book an already filled appointment	System shows "please select another appointment. It's been filled"
Relationships:		
Initiating	User	
Collaborating	Portal Agent, Scheduling Agent	
Other Diagrams:		
Data Requirements:		
Data Required	Appointment_ID USER_ID EMAIL	

1.1.8 Appointment Use Cases Sequence Diagram



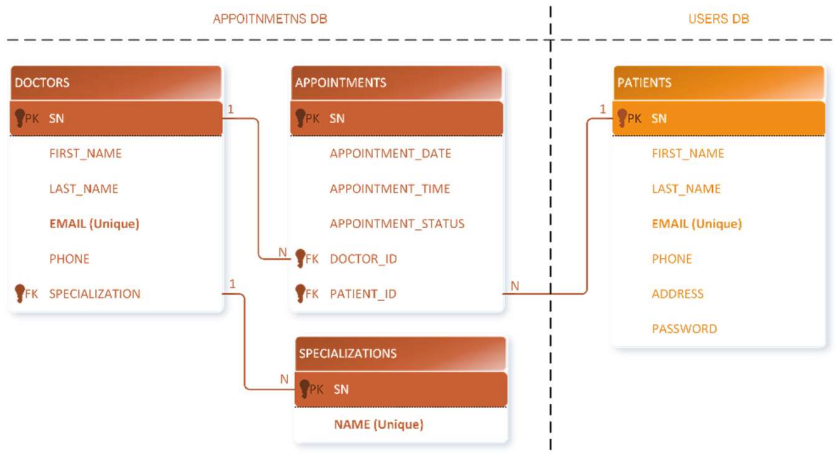
1.2 Class Diagram

An additional Class Diagram is Available in the Appendix



2 Data Specification

2.1 E-R Diagram



2.2 Data Definition

2.2.1 Users_DB

Table Name	Attributes	Type	Notes
PATIENTS	SN	INT	Primary, Auto_Increment
	FIRST_NAME	VARCHAR(45)	
	LAST_NAME	VARCHAR(45)	
	EMAIL	VARCHAR(45)	Unique
	PHONE	CHAR(10)	
	ADDRESS	VARCHAR(100)	
	PASSWORD	CHAR(32)	

2.2.2 Appointments_DB

Table Name	Attributes	Type	Notes
SPECIALIZATIONS	SN	INT	Primary, Auto_ Increment
	NAME	VARCHAR(45)	Unique
DOCTORS	SN	INT	Primary, Auto_ Increment
	FIRST_NAME	VARCHAR(45)	
	LAST_NAME	VARCHAR(45)	
	EMAIL	VARCHAR(45)	Unique
	PHONE	CHAR(10)	
	SPECIALIZATION	INT	Foreign Key
APPOINTMENTS	SN	INT	Primary, Auto_ Increment
	APPOINTMENT_DATE	DATE	
	APPOINTMENT_TIME	TIME	
	APPOINTMENT_STATUS	CHAR(1)	
	DOCTOR_ID	INT	Foreign Key
	PATIENT_ID	INT	Foreign Key

3 Inter-Agent Communication

This section documents the inter-agent communications structures'

3.1 Login Request

3.1.1 Input Parameters

Parmeters	Description

<message
to=LoginAgent@domain.postfix
from=PortalAgent@domain.postfix
{
"Email": String (email),
"Password": String (hexadecimal)
}
</message>
	user's credentials for authentication

3.1.2 Output Parameters

Parmeters	Description

<message
to= PortalAgent @domain.postfix
from= LoginAgent @domain.postfix
{
"SN": Integer,
}
</message>
	user is authenticated anduser's ID is sent back

Failure to authenticate output is reflected in the *performative* header with failure.

3.2 Register User

3.2.1 Input Parameters

Parmeters	Description

<message
to=RegisterationAgent@domain.postfix
from=PortalAgent@domain.postfix
{
"FirstName": String,
"LastName": String,
"Email": String (email),
"Phone": String (Numbers),
"Address": String,
"Password": String (hexadecimal)
}
</message>
	user's information for registration

3.2.2 Outputs Parameters

Output is a reply that is reflected in the *performative* header with either confirm or failure.

3.3 Fetch Appointments

3.3.1 Input Parameters

Parmeters	Description

<message
to=SchdeulingAgent@domain.postfix
from=PortalAgent@domain.postfix
{
"Doctor_ID": Interger
}
</message>
	doctor's ID to fetch the available appointment

3.3.2 Output Parameters

Parmeters	Description
<message to=SchdeulingAgent@domain.postfix from=PortalAgent@domain.postfix [{ "SN": Interger, “APPOINTMENT_DATE”: Date, “APPOINTMENT_TIME”: Time, “APPOINTMENT_STATUS”: CHAR, }, { "SN": Interger, “APPOINTMENT_DATE”: Date, “APPOINTMENT_TIME”: Time, “APPOINTMENT_STATUS”: CHAR, }, . . .] </message>	A list of available appointments’ information like date, time and status are fetched and sent back to user

3.4 Book Appointment

3.4.1 Input Parameters

Parmeters	Description

<message
to=SchdeulingAgent@domain.postfix
from=PortalAgent@domain.postfix
{
"Appointment_ID": Interger,
"USER_ID": Interger,
"USER_EMAIL": String (email),
}
</message>
	ID of the selected appointment and the selecting user

3.4.2 Outputs Parameters

Output is a reply that is reflected in the *performative* header with either confirm or failure.

3.5 Send Notification

3.5.1 Input Parameters

Parmeters	Description

<message
to=SchdeulingAgent@domain.postfix
from=PortalAgent@domain.postfix
{
"USER_EMAIL": String (email),
“APPOINTMENT_DATE”: Date,
“APPOINTMENT_TIME”: Time,
“DOCTOR_FIRST_NAME”: String,
“DOCTOR_LAST_NAME”: String,
}
</message>
	ID of the selected appointment and the selecting user

3.5.2 Outputs Parameters

Output is a reply that is reflected in the *performative* header with either confirm or failure.

Appendix A

Variant Class Diagram