CIS 320 Assignment 3
Analysis Modeling Skills Development

Background: One of the most important course goals is to develop your critical thinking and systems modeling skills. This assignment will allow you to practice the modeling techniques most often associated with systems analysis.

Objectives: Create a series of models that move from requirements to use cases, and then to behavioral and structural models. Demonstrate your mastery of the analysis and modeling techniques listed below.

Task: Create the series of models and deliverables listed below for the use cases assigned to you. I suggest you create the models in the order in which they are listed below; they build on each other in a logical manner.

For the use cases assigned to you, create: (1) use case descriptions; (2) a use case diagram; (3) a prototype of each use case, (4) a class diagram and (5) a sequence diagram for each use case. Expectations for each model are defined below.

Note: this is an individual assignment. The team should agree on the assignment of use cases to team members. It may also share the set of system requirements. Other forms of collaboration are not permitted.

Item	Description
Use Cases	For the assigned use case, construct a detailed essential use case. Follow the actor/system response format shown in Arlow and Neustadt. Adhere to use case naming conventions. Include the most recent set of system requirements. Provide a trace matrix that associates your use cases with system requirements.
Use Case Diagram	Create a use **case diagram** for your use cases. Use the guidelines in **Arlow and Neustadt** in formatting the diagram**.** Provide a **narrative** explaining how to interpret the use case diagram; i.e., what you would want the client to conclude about the system model when viewing the diagram.

| Item	Description
Use Case Prototypes	Create high-level prototypes (HTML or similar screen mockups) that represents the data needs and process flows of your use cases. The prototype may not connect to databases or use anything other than the minimum of programming.
Class Diagram	Create a class diagram showing the data attributes and operations of all classes identified during this assignment. Show the relationships among classes. Provide a narrative explaining how to interpret the class diagram; i.e., what you would want the client to conclude about the system model when viewing the diagram.
Sequence Diagram	Create sequence diagrams for your use cases. List the use case main flow along the left side of the sequence diagram.
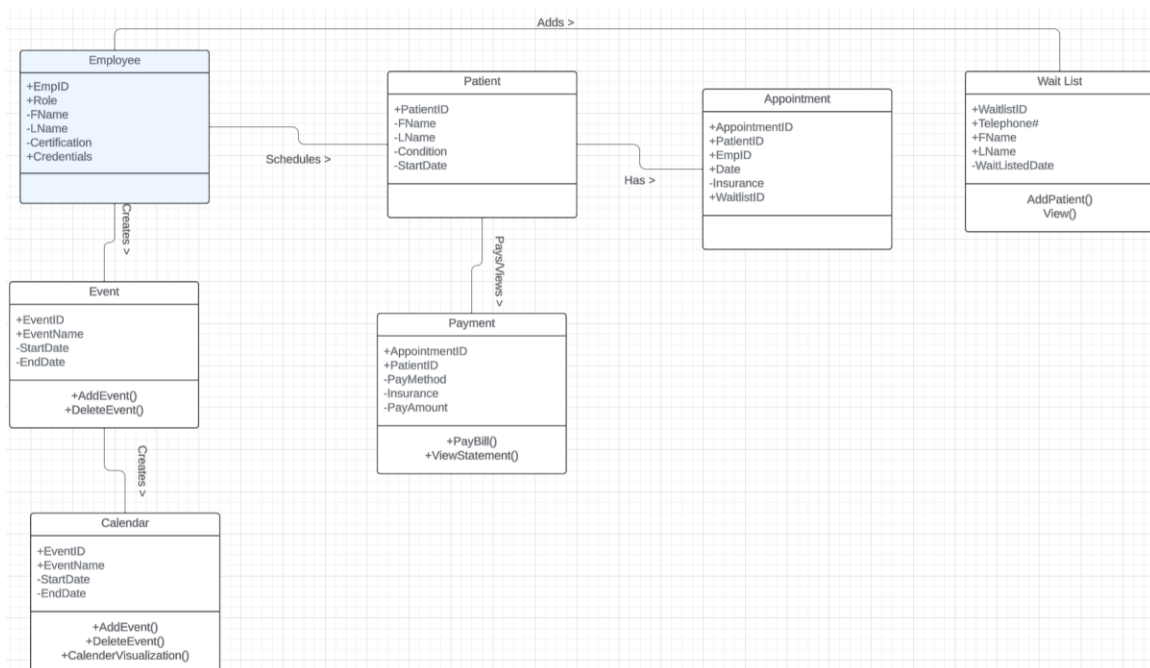
Class Diagram Narrative

The class diagram starts with the employee and from there branches out. The reason for doing this is that the employee has the administrative rights and the patient is paired to a practitioner. Furthermore the employee is connected to the event, calendar classes, and waitlist.

1.A. Use Case 25: Employee adds Calendar Events.

Use Case Name: Employee adds Calendar events	ID: 25	Importance Level: Low
Primary Actor: Employee		Risk Level: Low
Stakeholders and Interests: • *Website User:* To have access to upcoming events led by the organization. • *Organization:* To bring awareness to organization's goals and fundraising events.		
Brief Description: Employee can add new events to the website calendar.		
Trigger: Employee clicks on 'Add New' button. **Type**: External		
Normal Flow of Events: 1. Employee with admin credentials logs in to WordPress. 2. Employee goes to events tab. 3. Employee clicks "Add New" tab.		
Alternate/Exceptional Flows: 1a. User is unable to login due to lack of credentials. 2a. User's authentication is rejected and is unable to access website. 3a. User is unable to access 'Event Calendar' due to limited access.		

Prototypes
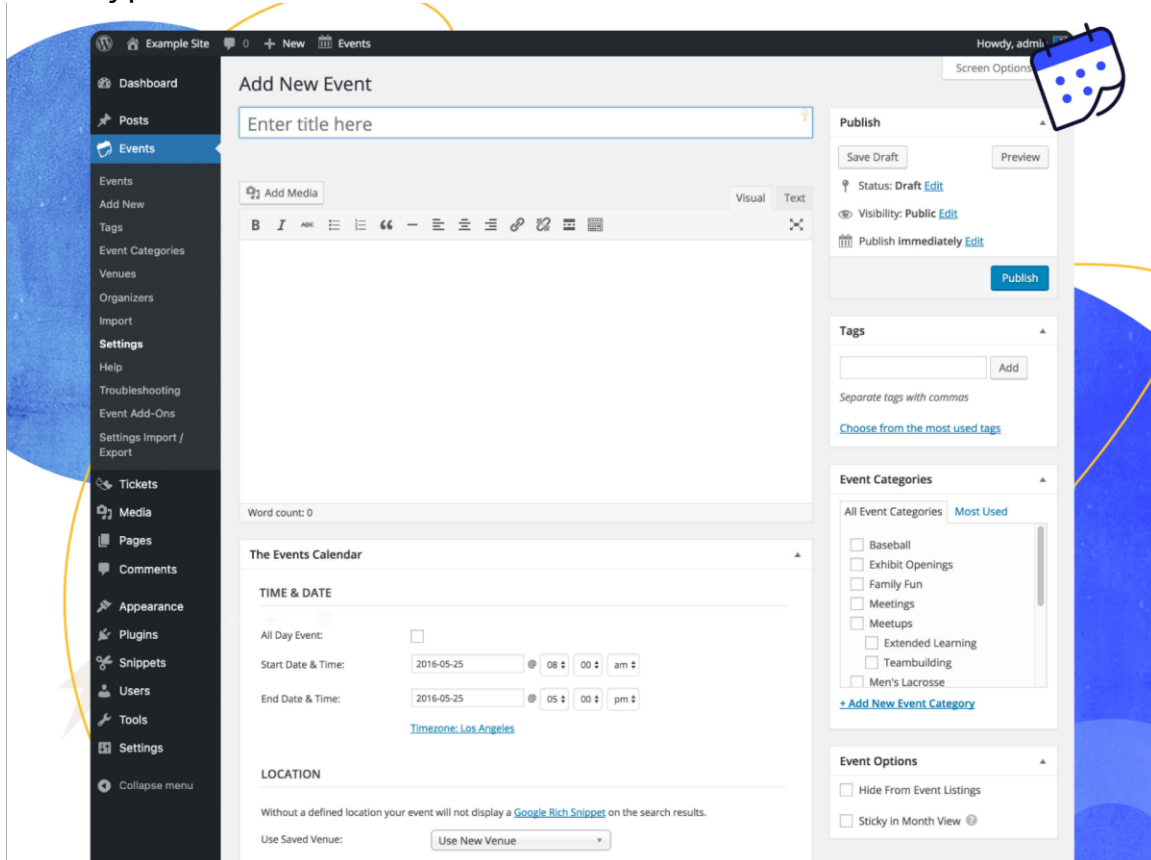
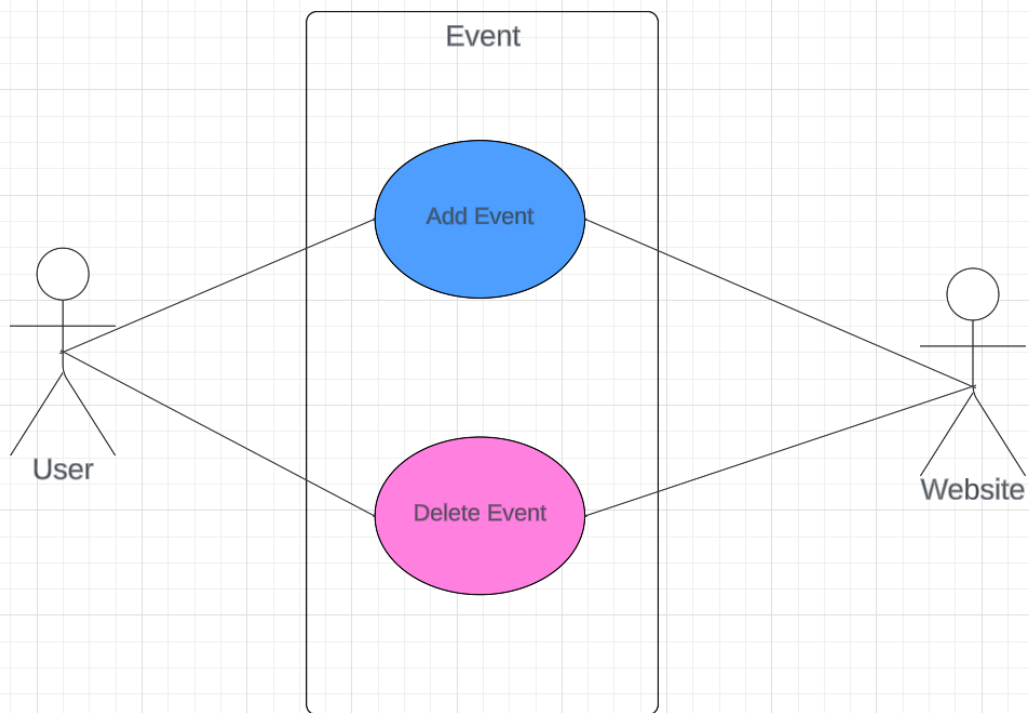## 2.A. Use Case 26: Employee removes Calendar events.

Use Case Name: Employee removes Calendar events	ID: 26	Importance Level: Low
Primary Actor: Employee		Risk Level: Low
Stakeholders and Interests:		

- ***Website User:*** To have access to upcoming events led by the organization.
- ***Organization:*** To bring awareness to organization's goals and fundraising events.

Brief Description: The employee can delete events from the website calendar.

Trigger: Employee selects event and clicks trash button.
Type: External

Normal Flow of Events:
1. Employee with admin credentials logs in to WordPress.
2. Employee goes to events tab.
3. Employee selects an event.
4. Trash icon appears at the top.
5. Employee clicks trash icon.

Alternate/Exceptional Flows:

       1a. User is unable to login due to lack of credentials.
       2a. User's authentication is rejected and is unable to access website.
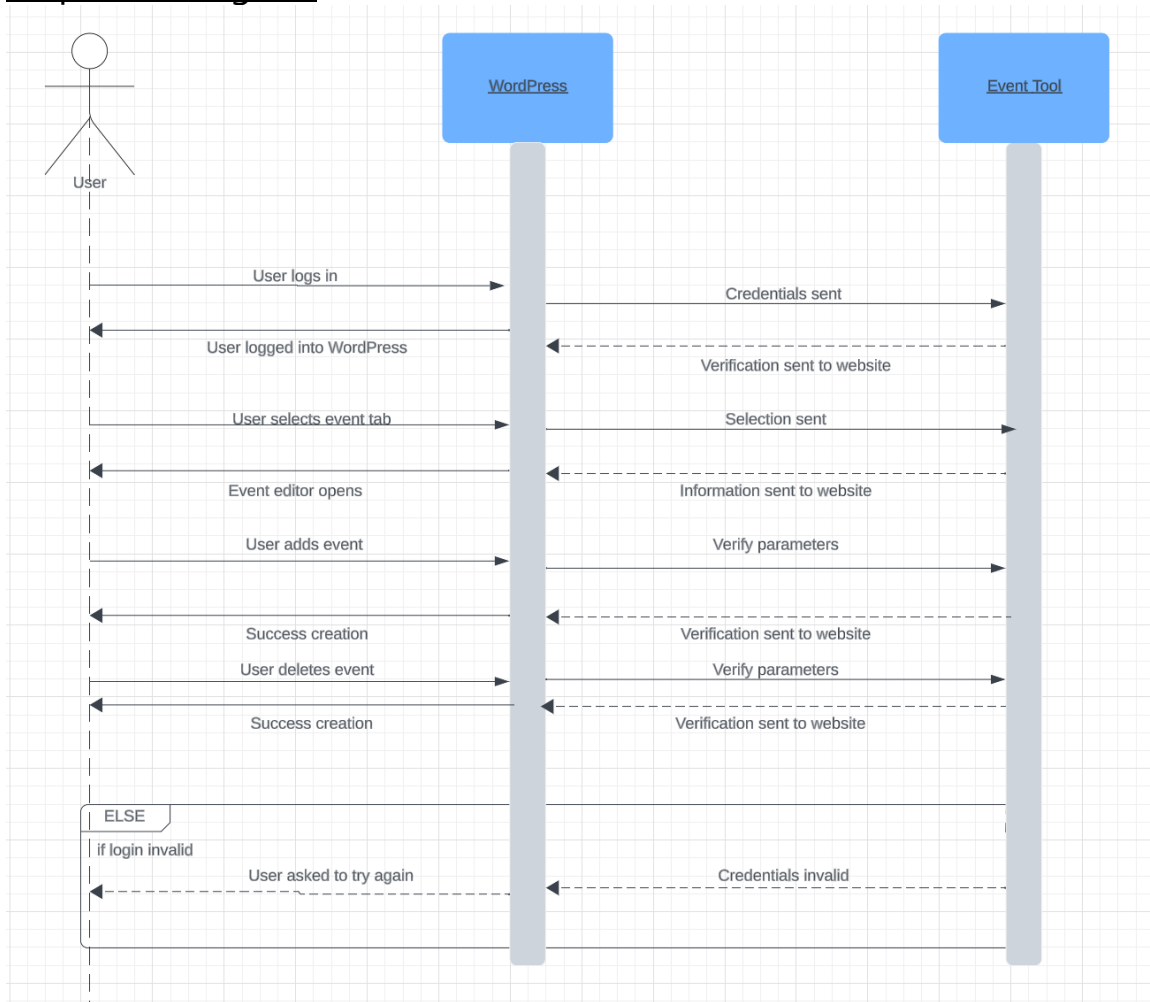   3a. User is unable to access 'Event Calendar' due to limited access.

Trace Matrix

		Employee adds Calendar events	Employee removes Calendar events
	System Requirements		
ID: SR010	The calendar will allow new events to be created	**X**	
ID: SR011	The calendar will allow old events to be deleted		**X**

Use Case Diagram

Sequence Diagram

User	WordPress	Event Tool

User logs in → WordPress

Credentials sent → Event Tool

User logged into WordPress ←

Verification sent to website ←

User selects event tab →

Selection sent →

Event editor opens ←

Information sent to website ←

User adds event →

Verify parameters →

Success creation ←

Verification sent to website ←

User deletes event →

Verify parameters →

Success creation ←

Verification sent to website ←

ELSE

if login invalid

User asked to try again ←

Credentials invalid ←

Prototype

3.A. Use Case 27: Patients pays bill through Patient Portal.

Use Case Name: Patient pays bill through Patient Portal	**ID**: 27	**Importance Level**: Medium
Primary Actor: Patient		**Risk Level**: Low
Stakeholders and Interests: • *Website User:* Patient pays a bill. • *Organization:* To allows collection of revenue from services provided.		
Brief Description: Allows the patient to pay a bill that has been sent from received therapy.		

Trigger: User clicks on 'Pay Now' button in the Billing & Payments tab. **Type**: External

Normal Flow of Events:
    1. Patient logs into the patient portal.
    2. Patient navigates to the Billing & Payments tab.
    3. Patient clicks the 'Pay Now' button.
    4. Patient enters billing information.
    5. Patient clicks 'Pay' button.

Alternate/Exceptional Flows:
 	
    1a. User is unable to login due to lack of credentials.
    2a. User's authentication is rejected and is unable to access website.

Prototype

4.A. Use Case 28: Patients views billing history through Patient Portal.

Use Case Name: Patients views billing history through Patient Portal	**ID**: 28	**Importance Level**: Low
Primary Actor: Patient		**Risk Level**: Low

Stakeholders and Interests: • *Website User:* Patient verifies statements. • *Organization:* To allows collection of revenue from services provided.	

Brief Description: Allows patient to view previous charges and payments made.

Trigger: Patient clicks on 'Billing & Payments' in the patient portal
Type: External

Normal Flow of Events:
1. Patient logs into the patient portal.
2. Patient navigates to the Billing & Payments tab.

Alternate/Exceptional Flows:

1a. User is unable to login due to lack of credentials.
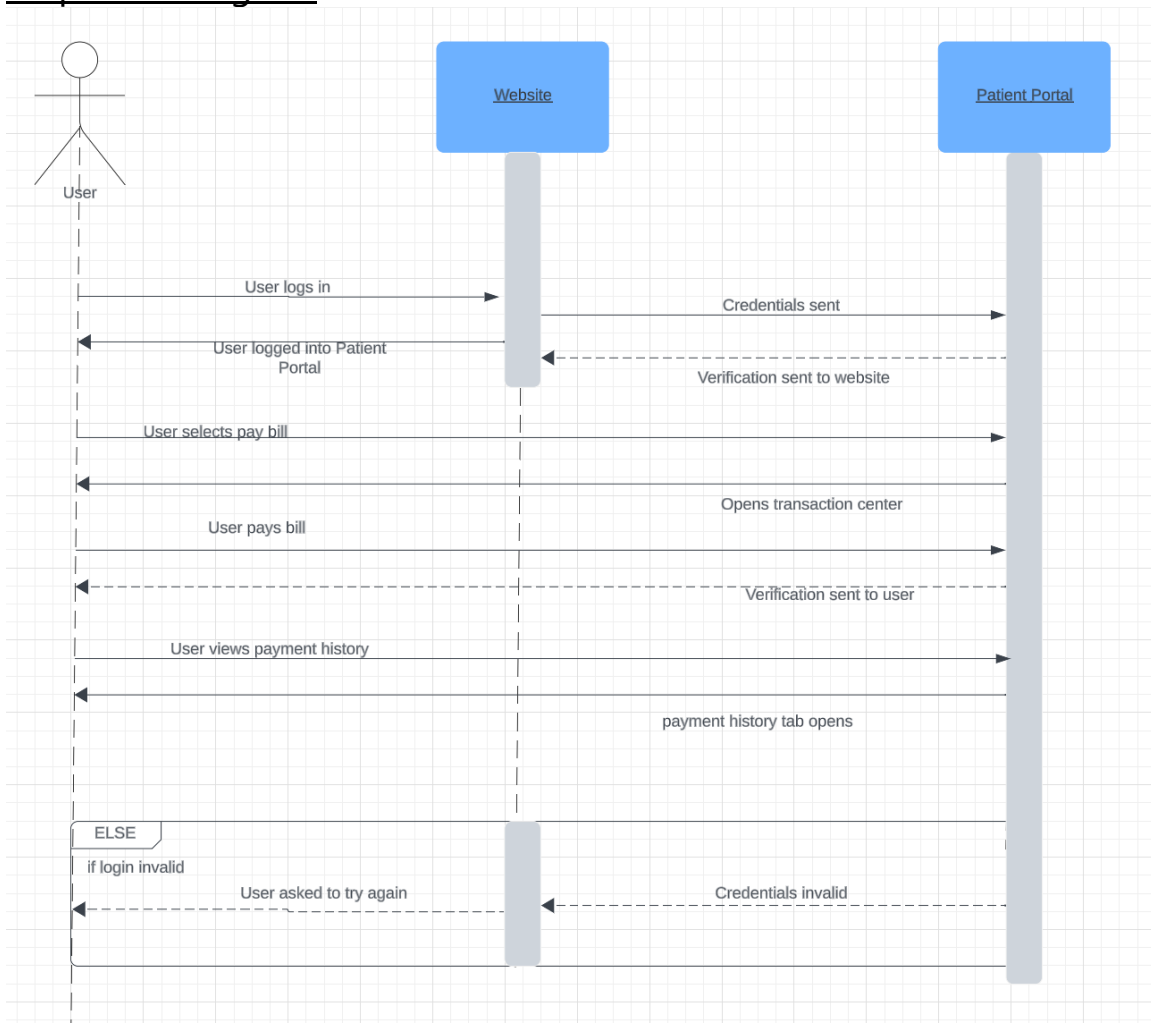2a. User's authentication is rejected and is unable to access website.

Trace Matrix

		Patients pays bill through Patient Portal	Patients views billing history through Patient Porta
	System Requirements		
ID: SR012	The patient portal will allow payment to be made	X	
ID: SR013	The patient portal will allow statements to be viewed		X

Use Case Diagram

Sequence Diagram

User

Website

Patient Portal

User logs in

Credentials sent

User logged into Patient Portal

Verification sent to website

User selects pay bill

Opens transaction center

User pays bill

Verification sent to user

User views payment history

payment history tab opens

ELSE

if login invalid

User asked to try again

Credentials invalid

Prototype

5.A. Use Case 29: Employee adds Patient to waitlist through the Patient Portal.

Use Case Name: Employee adds Patient to waitlist through the Patient Portal	**ID**: 29	**Importance Level**: High
Primary Actor: Employee		**Risk Level**: Medium

Stakeholders and Interests:

- *Patient:* Patient wants appointment
- *Organization:* Needs to view waitlist to better fill opened slots

Brief Description: Allows patients to be waitlisted in case no appointment slots are available

Trigger: Employee clicks on the 'Add Client' button
Type: External

Normal Flow of Events:
1. Employee logs into patient portal.
2. Employee navigates to the calendar tab.
3. Employee clicks the waitlist button.
4. Employee inputs client information.
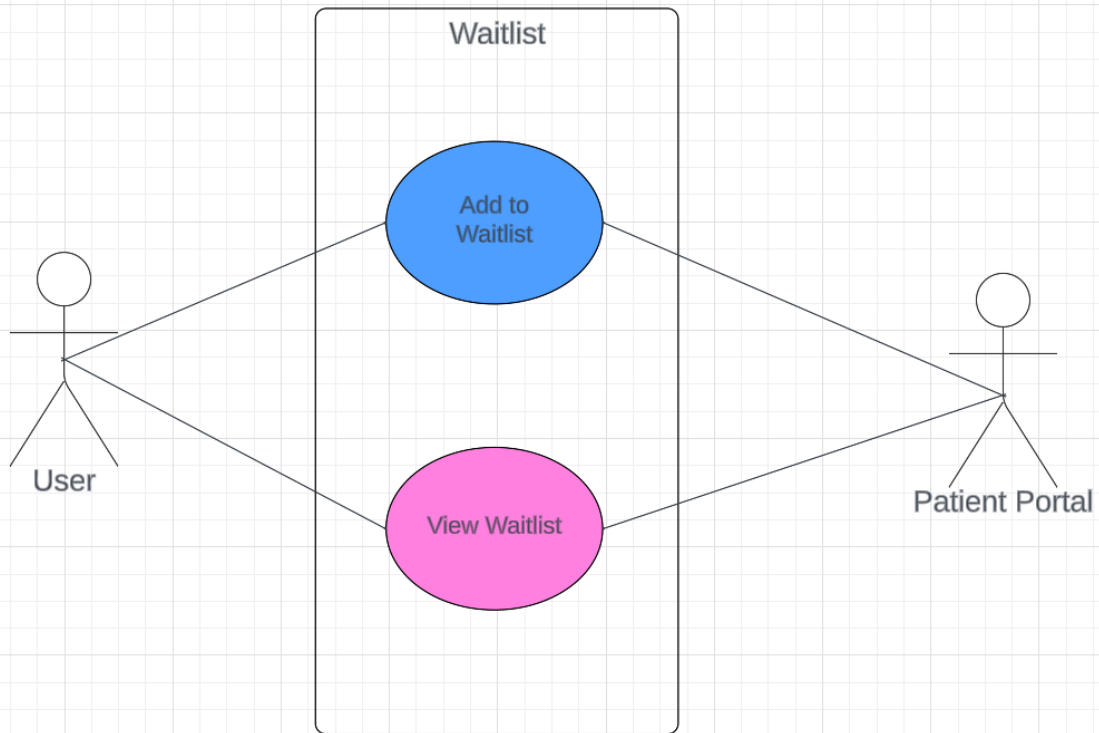5. Employee clicks the 'Add Client' button.

Alternate/Exceptional Flows:

      1a. User is unable to login due to lack of credentials.
      2a. User's authentication is rejected and is unable to access website.

3a. User is unable to access 'Waitlist' due to limited access.

Prototype
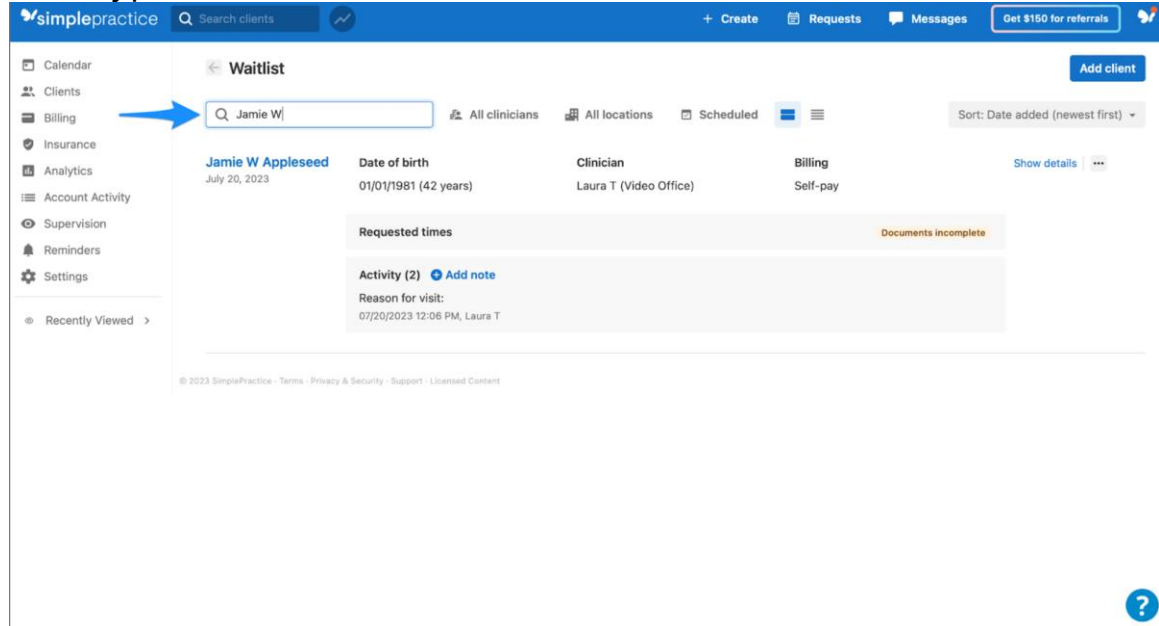
6.A. Use Case 30: Employee views waitlist through the Patient Portal.

Use Case Name: Employee views waitlist through the Patient Portal	**ID**: 30	**Importance Level**: High
Primary Actor: Employee		**Risk Level**: Low
Stakeholders and Interests: • *Patient:* Patient wants appointment • *Organization:* Needs to view waitlist to better fill opened slots		
Brief Description: Allows employees to view the waitlist to see availability		
Trigger: User clicks on the 'Calendar' tab **Type**: External		

Normal Flow of Events:
1. Employee logs into patient portal.
2. Employee navigates to the calendar tab.

Alternate/Exceptional Flows:

1a. User is unable to login due to lack of credentials.
2a. User's authentication is rejected and is unable to access website.
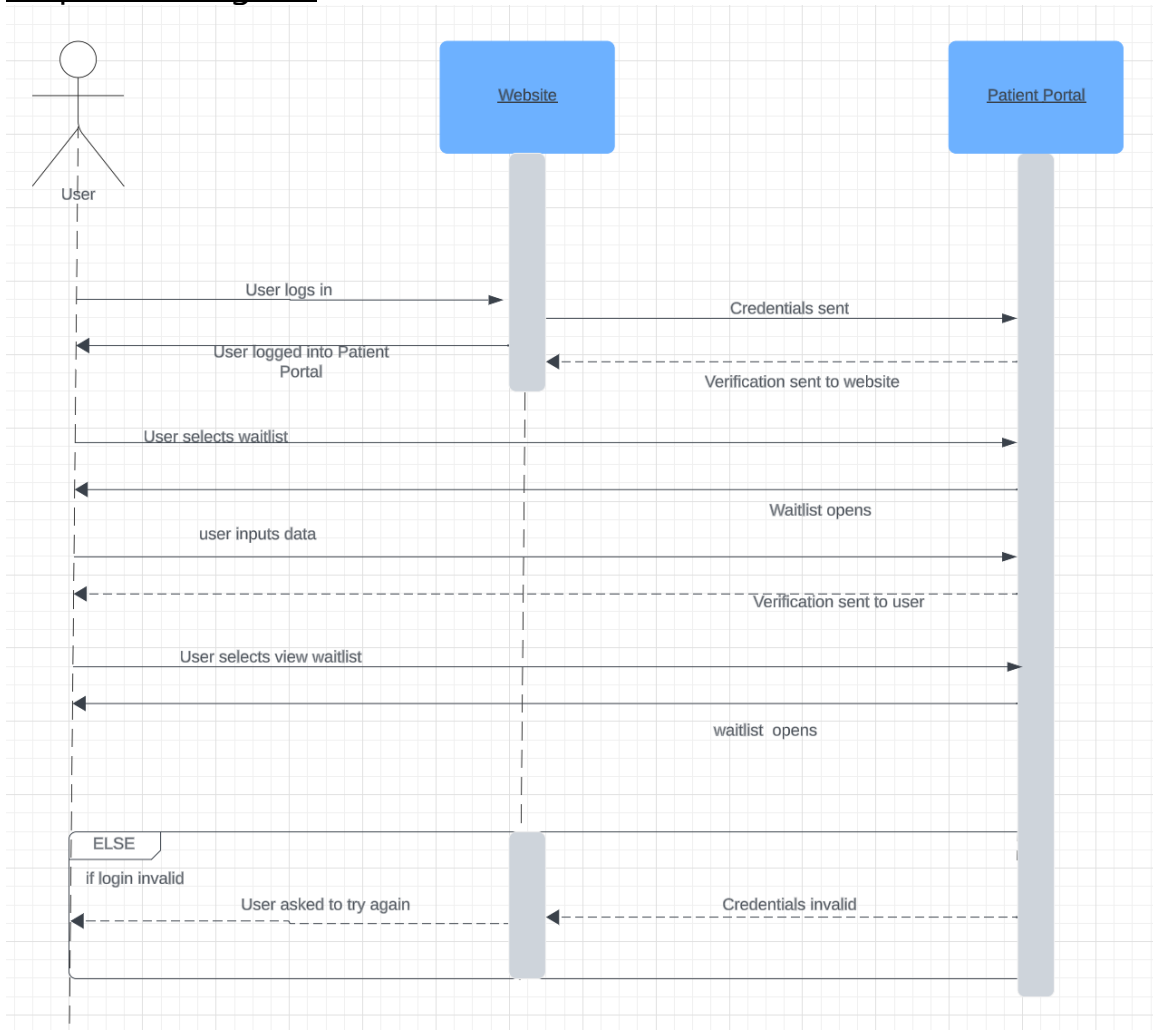3a. User is unable to access 'Waitlist' due to limited access.

Trace Matrix

		Employee adds Patient to waitlist through the Patient Portal	Employee views waitlist through the Patient Portal
	System Requirements		
ID: SR012	The waitlist allows new patients to be added	**X**	
ID: SR013	The waitlist allows the user viewing		**X**

Use Case Diagram

Sequence Diagram

User	Website	Patient Portal

User logs in →

Credentials sent →

← User logged into Patient Portal

← Verification sent to website

Verification sent to website

User selects waitlist →

←

Waitlist opens

user inputs data →

← Verification sent to user

User selects view waitlist →

←

waitlist opens

ELSE

if login invalid

← User asked to try again

← Credentials invalid

Prototype