**Project Workbook**

**Fall 2024**

**Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Issue** | **Description** | **Author** |
| September 3, 2024 | Project1-Phase1 | Adding Glossary (Question 4) | Group 12 |
| September 3, 2024 | Project1-Phase1 | Adding system actors (Question 6.1) | Group 12 |
| September 3, 2024 | Project1-Phase1 | Adding system use cases (Question 7.1) | Group 12 |
| September 3, 2024 | Project1-Phase1 | Adding use case diagram (Question 8.1) | Group 12 |
| September 13, 2024 | Project1-Phase2 | Produce a list of candidate classes | Group 12 |
| September 13, 2024 | Project1-Phase2 | Filtering the list of candidate classes | Group 12 |
| September 13, 2024 | Project1-Phase2 | Produce a potential class diagrams | Group 12 |
| September 13, 2024 | Project1-Phase2 | Identified potential class attributes | Group 12 |
| September 13, 2024 | Project1-Phase2 | Revised the diagrams | Group 12 |
| September 13, 2024 | Project1-Phase2 | Update the glossary | Group 12 |
| October 7, 2024 | Project1-Phase3 | Adding Potential iCARE Sequence Diagrams (Question 3.2) | Group 12 |
| October 7, 2024 | Project1-Phase3 | Adding iCARE Statechart Diagram (Question 5) | Group 12 |
| October 11, 2024 | Project1-Phase3 | Update the glossary | Group 12 |

**Table of Contents**

[iCARE System Glossary 3](#_Toc848736163)

[Introduction 4](#_Toc1152786002)

[Glossary 4](#_Toc692086708)

[iCARE System Actors 4](#_Toc412108806)

[iCARE System Use cases 5](#_Toc1327419480)

[iCARE System Use case Diagram 7](#_Toc961253290)

[List of Candidate Classes 8](#_Toc282457264)

[Potential iCARE Classes 9](#_Toc531848608)

[Potential iCARE class diagrams (entity classes only) 13](#_Toc349716962)

[Revised iCARE UML class diagram 17](#_Toc2062806036)

[Potential iCARE Sequence Diagrams 23](#_Toc302619395)

[iCARE Statechart Diagram 28](#_Toc1739559495)

**iCARE System Glossary**

**Introduction**

This document is used to define terminology specific to the problem domain, explaining terms, which may be unfamiliar to the reader of the use-case descriptions or other project documents. Often, this document can be used as an informal *data dictionary*, capturing data definitions so that use-case descriptions and other project documents can focus on what the system must do with the information.

**Glossary**

The glossary contains the working definitions for the key concepts in the iCARE System.

|  |  |
| --- | --- |
| Term | definition |
| Authentication | Authentication is the process of determining whether someone or something is, in fact, who or what it is declared to be. |
| User account | A user is a person who uses iCARE system. Each user should have an account in order to be identified by iCARE. To login to an account, a user is typically required to authenticate himself/herself with a password or other credentials for the purposes of accounting, security, logging, and resource management. |
| User friendly interface | It is a computer application screen that makes it easier for novices to use this application. Menu-driven programs, for example, are considered more user-friendly than command-driven systems. Graphical user interfaces (GUIs) are also considered user-friendly. |
| Physicians | A physician is a professional who practices medicine, which is concerned with promoting, maintaining or restoring human health through the study, diagnosis, and treatment of disease, injury, and other physical and mental impairments. |
| Digital ink | Refers to technology that digitally represents handwriting in its natural form. In a typical digital ink system, a digitizer is laid under or over an LCD screen to create an electromagnetic field that can capture the movement of a special-purpose pen, or stylus, and record the movement on the LCD screen. The effect is like writing on paper with liquid ink. The recorded handwriting can then be saved as handwriting or converted to typewritten text using handwriting recognition technology. |
| UI component | UI stands for User Interface. It is a junction between a user and a computer program. An interface is a set of commands or menus through which a user communicates with a program. |
| PDF Document | Portable Document Format (PDF) is a file format used to present and exchange documents reliably, independent of software, hardware, or operating system. |
| Pen-based document | Any document that is created using tablet and stylus as pointing devices in addition to handwriting recognition capability. |
| Windows-based desktop computers | A desktop computer is a personal computer powered by Microsoft Windows operating system in a form intended for regular use at a single location desk/table due to its size and power requirements. |
| Tablet PCs | A tablet PC, commonly shortened to tablet, is a mobile computer with a touchscreen display, circuitry, and battery in a single device. |
|  |  |

**iCARE System Actors**

The first useful step to analyze the system functionality is to look in the problem statement at the things that interact with the system. In UML use case analysis, these external things are called **actors**. Actors are identified based on the following:

* Actors are always external to the system – they are therefore outside our control.
* Actors interact directly with the system.
* Actors represent roles that people and things play in relation to the system, not specific people or specific things.
* Each actor has a unique name and description.

|  |  |
| --- | --- |
| Actor | Description |
| iCARE User | A general user interacting with iCARE system. This general user can be a doctor, a nurse, or the system administrator. We will use the name “worker” to refer to a doctor or a nurse. In order to this general user to use iCARE, he/she needs to successfully login to the system. |
| iCARE Admin | A special type of iCARE User who is responsible for ensuring around-the-clock technical maintenance and support. This normally includes maintaining workers accounts and the connectivity issues with the drugs management system. |
| iCARE Worker | A special type of iCARE User who wish to use iCARE system to automate the paper-based processes and to speed up its data entry related tasks, i.e., doctor or nurse. |
| Drugs Management System | A drug management system is an external software solution used by or integrated with the iCARE system to provide accurate drug names and descriptions. |

**iCARE System Use cases**

|  |  |
| --- | --- |
| Use Case | Description |
| Authenticate User | iCARE shall provide a functionality to authenticate its user by using a pen only. The user taps his name and can "ink" his password in the box. The user can of course also type (typically using the on-screen keyboard) into the box if desired. Once the user has logged in, the set of controls dynamically changes to match the permissions of this account. Note that, each user should be successfully authenticated before using any of iCARE functionalities. |
| Manage Worker Account | iCARE shall provide the functionality for the system administrator to create user accounts for doctors and nurses to use iCARE according to a set of access controls predefined for each user type. The administrator account itself is shipped with the system. |
| Display iCARE Board | iCARE shall provide the functionality for its workers to show a list of the existing patients corresponding to the geographic unit view. |
| Assign Patient | iCARE shall provide the ability for its workers to browse through the provided list of patients exist in the iCARE Board, select one or more patients and assign himself to them. |
| Display Palette | Display Palette UC shall provide a graphical user interface GUI to help simplify the choosing of documents among large number of potential documents. It's one tablet screen in size so the user can see all choices in one view which consists of a series of buttons for the user to tap on. The Palette shall be able to manage larger number of document choices than can fit on one screen. |
| Display My Board | iCARE shall provide the functionality for its workers to show only the active patients list of the logged on worker, and hence My Board view can be different for doctors and nurses. |
| Manage Document | iCARE shall allow the workers to create a new document. It is a text based document and the works shall be able to modify its contents.  This document will be saved in a PDF format and will be tagged with the patient’s a metadata. This metadata includes Patient ID, Date of creation/imported, the user ID of the creator/importer, Modification date, the user ID of the modifier, Brief description.  This UC includes two UCs Manage Text Insertion and Recognize Drug names. |
| Manage Text Insertion | This is a supplier UC for the Manage Document UC that shall allow the workers to insert different types of text into the new and/or the old documents created by the base UC. This text can be any paragraph that describes treatments, drug orders, or prescriptions. |
| Recognize Drug Names | This is a supplier UC for the Mange Document UC that shall allow iCARE to recognize generic and brand drug names during the text insertion process to help auto completion capability. This information will be extracted from a common repository managed by the Drugs Management System. |
| Import Images | This UC provides a tool to import images using a scanner or by browsing the internal stored image files into the application. These imported files will then be converted to PDFs, to become a part of the iCARE documents repository. |
| Manage Patient Record | This UC shall allow the workers at iCARE to maintain (add, modify) patient records. Patient’s record includes (but not limited to) the following information: ID, name, address, date of birth, height, weight, bloodGroup, BedID, Treatment area. Each patient record will be associated with zero or more treatment records and a set of digital documents. |

**iCARE System Use case Diagram**

Diagram

Description automatically generated

**List of Candidate Classes**

pen

user

his name

his password

box

set of controls

account

iCARE Board

My Board

patient

the geographic unit

Worker

Palette

documents

tablet

screen

text

treatments

drug orders

prescriptions

images

scanner

application

PDFs

documents

metadata

Patient ID

Date of creation/imported

user ID

the creator/importer

Modification date

Brief description

administrator

doctors

nurses

records

Treatment area

treatment records

digital documents

**Potential iCARE Classes**

|  |  |  |
| --- | --- | --- |
| Class name | Type | Brief Description |
| UserAuthenticationForm | Boundary | The UserAuthenticationForm will be displayed by the iCARE system asking the user to ink or enter its username and password to be authenticated. |
| iCAREUser | Entity | The common information of the iCARE works and the iCARE admin such as User ID and User name will be maintained by this entity class. |
| iCAREWorker | Entity | iCAREWorker is a specification class inherits the iCAREUser class and add specific information about the worker such as the profession name and type. |
| iCAREAdmin | Entity | iCAREAdmin is a specification class inherits the iCAREUser class and add specific information about the system administrator such as the starting and ending date for the admin responsibility. |
| UserPassword | Entity | The UserPassword class store the iCARE user account information that include a password related information like an encrypted password of a user, the day the password expires, whether or not the password has to be changed from time to time, the minimum and maximum time between password changes, etc. |
| UserAuthenticationController | Control | A control class to accept and validate the iCARE username and its corresponding password. It takes the decision whether the iCARE user is allowed to access the iCARE services or not. |
| ManageAccountsForm | Boundary | iCARE displays the ManageAccountsForm when the system admin initiate the process for adding/modifying iCARE user account. This form will help the admin to inter the user general information like name and addresses and the secret information like password and the associated access roles. |
| UserRole | Entity | UserRole class maintains a list of all available iCARE access roles and the related permissions. For example, System administrator who has a privilege to create/modify user accounts, Physician who has a privilege for all iCARE functionality for doctors, or Nurse who has the privilege to access all nurses’ duties at iCARE. |
| ManageAccountsController | Control | A control class to create and store the new iCARE users’ information. It also determines whether the entered username has been used by other user or not, it encrypts the password before storing, and finally assign a specific system access role to the user. |
| iCAREBoardForm | Boundary | A boundary class to define a graphical user interface GUI elements that display a list of all patients at iCARE admitted to the current unit location. |
| PatientRecord | Entity | The PatientRecord class defines the fundamental information of the patient at iCARE. For example, ID, name, address, etc. |
| GeoCodes | Entity | GeoCodes is a class holding the attributes that will be used as a look-up dictionary for the iCARE units’ locations. |
| iCAREBoardController | Control | This class defines the attributes and the methods needed to retrieve the information of all patients at iCARE in the current location maintained by the PatientRecord class and display it through the boundary class iCAREBoardForm. |
| AssignPatientForm | Boundary | AssignPatientForm is a boundary class that show all admitted patients at iCARE and give the worker the ability to assign one or a group of patients to him/herself. |
| TreatmentRecord | Entity | This entity class acts as an association class in order to hold the common data between the patients and the workers. The patients and the workers are linked together by a many-to-many relationship because one worker may have many patients and one patient may be treated by many workers. This common data is simply the treatments information. |
| AssignPatientController | Control | AssignPatientController class displays a list of all patients and response to the workers interactions to assign one or more patients to this work. |
| DisplayPalette | Boundary | This class defines a GUI window that show the available iCARE documents and help the interested worker to choose a document he or she wish to work on. This window fits in one tablet screen size so the worker can see all choices in one view which consists of a series of buttons to facilitate such purpose. |
| DocumentMetadata | Entity | This entity class will be used to store the information required to manage iCARE PDF files. This information includes: Patient ID, Date of creation/imported, the user ID of the creator/importer, Modification date, the user ID of the modifier, Brief description about the PDF file. Note that, the PDF file contents will be saved on the local storage and we just need to link these files with the patient IDs and the worker IDs so that the works can access the content of these PDFs easily.  As iCARE provides the functionality to insert texts into these PDFs, these texts will be achieved in a separate entity class called “ModificationHistory” as described below. |
| DisplayPaletteController | Control | This class read all available iCARE records achieved in the DocumentMetadata entity class and group them in a way that help organize and display them in a usable screen size lists. |
| DisplayMyBoard | Boundary | DisplayMyBoard is an application window that shows only the active patients list of the logged on doctor or nurse. |
| DisplayMyBoard Controller | Control | This control class defines the functionality that is needed to let the DisplayMyBoard boundary class to show the active patients list related to the logged on worker. |
| ManageDocumentForm | Boundary | ManageDocumentForm is a class that defines the attributes and methods needed to create or modify the iCARE documents. |
| DocumentMetadata | Entity | This entity class will be used to store the information required to manage iCARE PDF files. This information includes: Patient ID, Date of creation/imported, the user ID of the creator/importer, Modification date, the user ID of the modifier, Brief description about the PDF file. Note that, the PDF file contents will be saved on the local storage and we just need to link these files with the patient IDs and the worker IDs so that the works can access the content of these PDFs easily.  As iCARE provides the functionality to insert texts into these PDFs, these texts will be achieved in a separate entity class called “ModificationHistory” as described below. |
| ModificationHistory | Entity | Any new text addition made on any PDF document by any workers need to be saved using this entity class. This information include the modification date and the modification description. This modification description can be any paragraph that describes treatments, drug orders, or prescriptions to be added to the original PDF.  Note that, each PDF document may have one or more instance of this class as the information of the PDF document being created for the first time need also be achieved.  Each instance of this class “ModificationHistory” should hold information about its creator (the worker). |
| DrugsDictionary | Entity | An entity class that manage and control the access to the external Drugs Management System. It provides read only access to the database of this external system. |
| ManageDocumentController | Control | This control class helps the workers to control the creation process of a new document. The original contents of the document will be save in a PDF format and will be linked to the Document Metadata. This UC includes two UCs Manage Text Insertion and Recognize Drug names. |
| ImportImageWindow | Boundary | This boundary class defines the user interface controls that help the worker to scan or browse the patient image-based documents. |
| ImportImageController | Control | This class controls the process of importing images using a scanner or by browsing the internal stored image files into the application. This control class will then convert these imported images into PDFs, to become a part of the iCARE documents repository. |
| ManagePatientRecordForm | Boundary | iCARE provides their workers with a form represented by the ManagePatientRecordForm boundary class by which the worker can fill and/or modify his/her patients information. |
| ManagePatientRecordController | Control | This control class defines the method that allow the workers at iCARE to maintain (add, modify) patient records. Patient’s record includes (but not limited to) the following information: ID, name, address, date of birth, height, weight, bloodGroup, BedID, Treatment area. Each patient record will be associated with zero or more treatment records and a set of digital documents. |

**Potential iCARE class diagrams (entity classes only)**

Graphical user interface

Description automatically generated

Diagram

Description automatically generated

Graphical user interface, text

Description automatically generated

Diagram

Description automatically generated

Graphical user interface, diagram

Description automatically generated

Diagram

Description automatically generated

Diagram

Description automatically generated

Graphical user interface, text

Description automatically generated

Chart, waterfall chart

Description automatically generated

**Revised iCARE UML class diagram**

1. **iCAREUser Class**

|  |  |  |
| --- | --- | --- |
| Attribute name | Type | Brief Description |
| ID | String | This is the primary key of the iCARE user info class. Every registered iCARE user is assigned a unique user ID. |
| name | String | The name attribute stores the full name of the iCARE user. |

1. **iCAREWorker Class**

|  |  |  |
| --- | --- | --- |
| Attribute name | Type | Brief Description |
| profession | String | The profession attribute store the worker profession name. |

1. **iCAREAdmin Class**

|  |  |  |
| --- | --- | --- |
| Attribute name | Type | Brief Description |
| adminEmail | String | The adminEmail attribute stores the administrator email address. |
| dateHired | Date | The date when the system administrator is hired by iCARE. |
| dateFinished | Date | The date when the system administrator left the iCARE. |

1. **UserPassword Class**

|  |  |  |
| --- | --- | --- |
| Attribute name | Type | Brief Description |
| ID | String | This is the primary key of the UserPassword class. The value of this attribute need to be the same as the value corresponding customer. |
| userName | String | The name attribute stores the use account name of the customers. |
| encryptedPassword | String | The userEncryptedPassword attribute stores the encrypted version of the customer password. A salted hash will be used in order to encrypt the password. |
| passwordExpiryTime | Integer | From time to time the system requires the customer to change the password. The passwordExpiryTime attribute stores this period of time. |
| userAccountExpiryDate | Date | The userAccountExpiryDate attribute stores the expiry date of the customer account if any. |

1. **UserRole Class**

|  |  |  |
| --- | --- | --- |
| Attribute name | Type | Brief Description |
| ID | String | This is the primary key of the UserRole class. The value of this attribute distinguishes one role from the other. |
| roleName | String | The roleName attribute stores the iCARE access roles and the related permissions. Example, System administrator who has a privilege to create/modify user accounts, Physician who has a privilege for all iCARE functionality for doctors, or Nurse who has the privilege to access all nurses’ duties at iCARE. |

1. **GeoCodes Class**

|  |  |  |
| --- | --- | --- |
| Attribute name | Type | Brief Description |
| ID | String | This is the primary key of the GeoCodes class. The value of this attribute distinguishes one geographical unit from the other. |
| description | String | The description attribute stores the location and characteristics of the iCARE units. |

1. **PatientRecord Class**

|  |  |  |
| --- | --- | --- |
| Attribute name | Type | Brief Description |
| ID | String | This is the primary key of the patient info class. Every admitted patient is assigned a unique ID. |
| name | String | The name attribute stores the full name of the patient. |
| address | String | The address attribute stores the address of the iCARE patient. |
| dateOfBirth | Date | The DOB attribute stores the date of birth of the patient. |
| height | double | The height attribute stores the patient height in inches. |
| weight | double | The weight attribute stores the patient weight in pounds. |
| bloodGroup | String | The bloodGroup attributes store the classification of patient’s blood. Example: A, B, AB and O, with +, − |
| bedID | String | The bedID attribute stores the patient bed ID |
| treatmentArea | String | The treatmentArea attribute stores the patient treatment unit or department. |

1. **TreatmentRecord Class**

|  |  |  |
| --- | --- | --- |
| Attribute name | Type | Brief Description |
| treatmentID | String | This is the primary key of the treatment record class. The value of this attribute distinguishes one treatment from another. |
| description | String | The description attribute stores the detail worker’s recommendation and treatment. |
| treatmentDate | Date | The date and time of the treatment. |

1. **DocumentMetadata Class**

|  |  |  |
| --- | --- | --- |
| Attribute name | Type | Brief Description |
| docID | String | This is the primary key of the document metadata class. |
| docName | String | The docName attribute stores the name and he title of the PDF document. |
| dateOfCreation | Date | The date and time of the document. |

1. **ModificationHistory Class**

|  |  |  |
| --- | --- | --- |
| Attribute name | Type | Brief Description |
| dateOfModification | Date | The date and time of the document modification. |
| description | String | The description attribute stores the detail worker’s modification made on the document. |

1. **DrugsDictionary Class**

|  |  |  |
| --- | --- | --- |
| Attribute name | Type | Brief Description |
| ID | String | This is the primary key of the drugs dictionary class. |
| name | String | The name of the drugs to be used in the auto completion facility. |

Graphical user interface, text, application

Description automatically generated Diagram

Description automatically generated

Graphical user interface, text

Description automatically generated

Diagram

Description automatically generated

A picture containing diagram

Description automatically generated

Diagram

Description automatically generated

Diagram

Description automatically generated

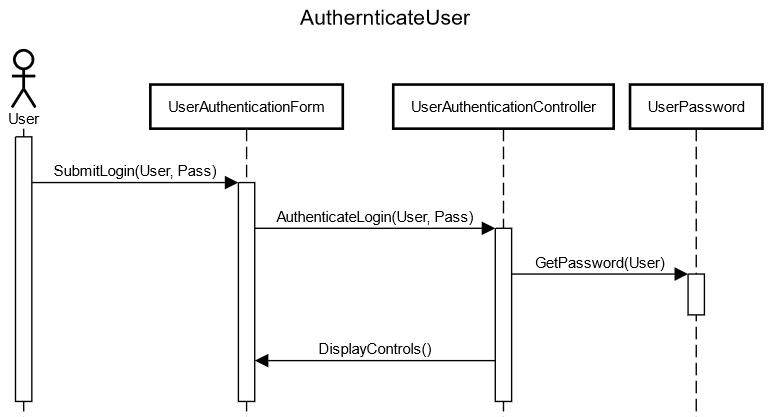
Graphical user interface

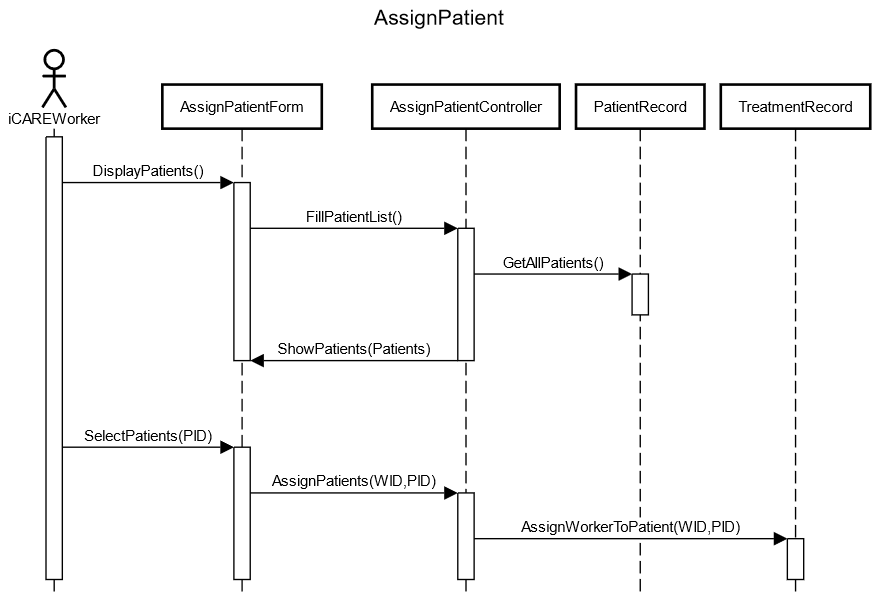
Description automatically generated

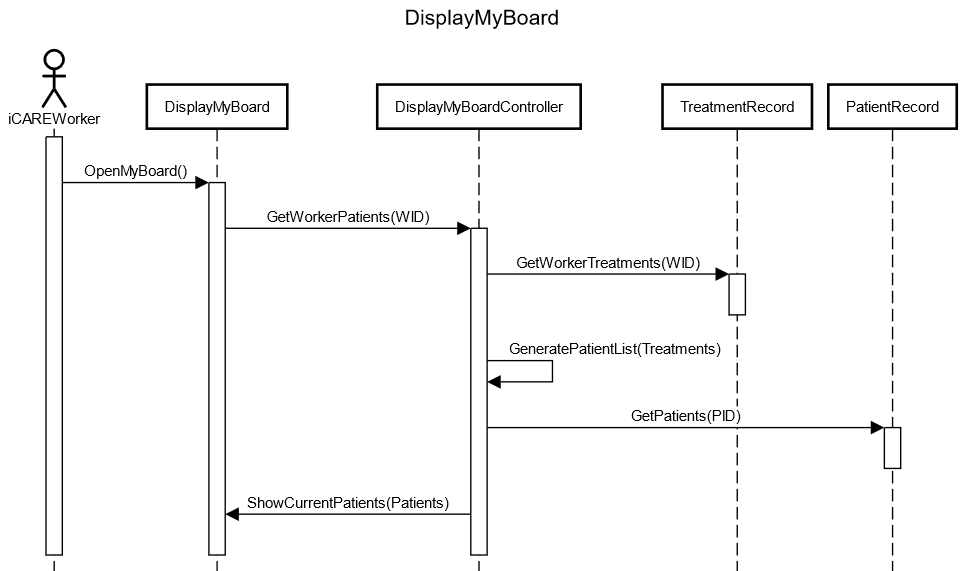
Diagram

Description automatically generated

**Potential iCARE Sequence Diagrams**





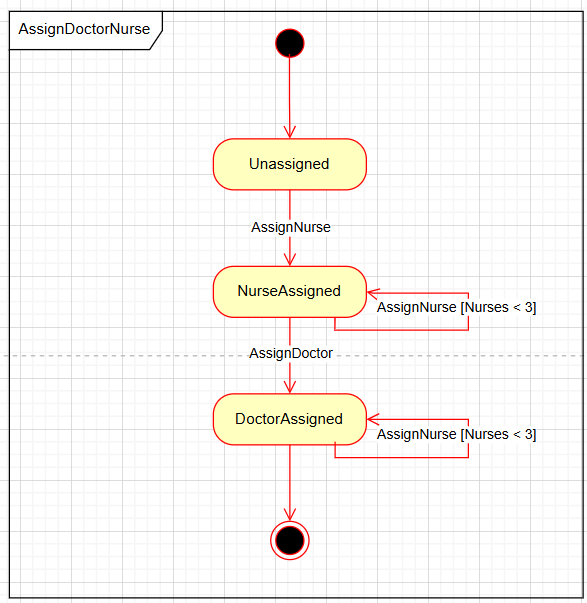


Detailing Operations

|  |  |  |
| --- | --- | --- |
| **Class name** | **Receiving message** **(operation)** | **Brief Description** |
| UserAuthenticationForm | SubmitLogin() | **Description:** This operation will submit the login credentials of the User to the UserAuthentificationForm  **Parameters:**  -ID: String (The ID of the user)  -password: String (This is for authenticating the user)  **Returns:** N/A |
| DisplayControls() | **Description:** This operation will display the controls to match the permissions of the user that is signed in. Controls can vary across users  **Parameters:**  -userID (for displaying the appropriate controls a user)  **Returns:** N/A (Display operation) |
| UserAuthenticationContoller | AuthenticateLogin() | **Description:** This operation will approve or reject the login credentials that were submitted by the user.  **Parameters:**  -ID: string (user ID that is being authenticated)  -password: string (password that is being authenticated)  **Returns:** NA |
| UserPassword | GetPassword() | **Description:** This operation gets the password from the UserPassword that is associated with the User’s ID.    **Parameters:**  -ID: string (the user ID for the password that it is associated to)  **Returns:**  -password: String (The password of the user, used to compare for authentication) |
| AssignPatientForm | DisplayPatients() | **Description:** This operation will initiate the process of showing all the current iCARE patients to be chosen from.  **Parameters:** N/A  **Returns:** N/A (Display operation) |
| SelectPatients() | **Description:** This operation will allow a worker to choose a patient from the patient list.  **Parameters:**  -PatientID: string (the ID of the patient that is selected)  **Returns:** N/A |
| ShowPatients() | **Description:** This operation will take the list of current patients assigned to the worker to be displayed on the Assign patient form.  **Parameters:**  -PatientIDs: PatientRecord[] (A list of all patients in iCARE)  **Returns:** N/A (Display operation) |
| AssignPatientController | FillPatientList() | **Description:** This operation initiates filling the list of available patients by interacting with the patient records.  **Parameters:** N/A  **Returns:**  -patientList: String[] (List of patients information to display) |
| AssignPatients() | **Description:** This operation will map allow a single patient to the current worker.  **Parameters:**  -WorkerID (the id of the worker that is being assigned to the patient)  -PatientID (the id of the patient that is being assigned)  **Returns:** N/A |
| TreatmentRecord | GetWorkerTreatments() | **Description:** This operation returns all treatment records that a designated Worker is assigned to.  **Parameters:** WorkerID (the id of the worker that is being assigned to the patient)  **Returns:**  -TreatmentIDs: String[] (A list of all treatment records that have the designated Worker on the assignment.) |
| AssignWorkerToPatient() | **Description:** This operation will designate what worker is assigned to a patient within the patients treatment records.  **Parameters:**  -WorkerID (the id of the worker that is being assigned to the patient)  -PatientID (the id of the patient that is being assigned)  **Returns:** N/A |
| PatientRecord | GetPatients() | **Description:** This operation will return the patient information for the provided patient ids  **Parameters:**  -PatientIDs: String[] (a list of ids of patients to be returned)  **Returns:**  -PatientList: PatientRecord[] (Patient information specified by the patient ID list provided.) |
| GetAllPatients() | **Description:** This operation will return the list of all current patients in iCARE.  **Parameters:** N/A  **Returns:**  -patients: PatientRecord (A list of all currently available patients.) |
| DisplayMyBoard | OpenMyBoard() | **Description:** This operation is initiated by the active user and displays their unique myboard with their assigned patient list.  **Parameters:**  -UserID: String (The ID of the active user)  **Returns:** N/A (Display operation) |
| ShowCurrentPatients() | **Description:** This operation will show all active assigned patients for the given worker.  **Parameters:**  -patients: PatientRecord[] (a list of all patients with corresponding treatment records that include the worker.)  **Returns:** N/A (Display operation) |
| DisplayMyBoardController | GetWorkerPatients() | **Description:** This operation will access all the patient treatment records associated with a unique worker ID.  **Parameters:**  -workerID: String (the worker id of the assigned patients)  **Returns:**  -patientIDs: String[] (the patients that the worker is associated to) |
| GeneratePatientList() | **Description:** This operation will generate a patient list in the worker’s MyBoard.  **Parameters:**  -treatmentIDs: String[] (treatment IDs associated with the worker and patients, for retrieving the patients the worker is assigned)  **Returns:** N/A |

**iCARE Statechart Diagram**

**Diagram**:



Brief Description:

The Patient starts in the Unassigned state. The patient then transitions to the NurseAssigned state where more nurses can be assigned if there are 2 or less assigned. Only after a nurse has been assigned can the patient transition to the DoctorAssigned state, where more nurses can also be assigned (at most 3).