Software Requirement Specification

Group members		Ryan Brady
		Jacob Alexander
		Nick Augustus
		Aaron Matthews
		Elijah Borum
		Victoria Castagnola

1. Project Description

The main goals for this program is to have a streamlined system for adding and discharging patients as well as booking E.R., in-patient, and O.R. rooms. The user will also be able to utilize an advanced search for patients (both present and past) and provide useful information about them such as a patient's room number, chart, unique identifier number, and schedule. There will also be an easy way for doctors/nurses to book rooms and what times they need those rooms as well as a visual interface to show which rooms are available and if not the rooms will be color coded to let you know the availability of that room and or any other rooms that may fit your need and if they're available. There will be security and failsafe procedures put into place to make sure the program continues to run smoothly without any interruptions.

2. Functional Requirements

FR01	The software must enable the hospital staff to add patients
FR02	The software must enable the hospital staff to discharge patients
FR03	The software must enable the hospital staff to view a patient's room number
FR04	The software must enable the hospital staff to view a patient's chart and notes
FR05	The software must enable the hospital staff to view a patient's unique identification number
FR06	The software must enable the hospital staff to search for patients
FR07	The software must enable the hospital staff to view a GUI of rooms
FR08	The software must enable the hospital staff to update room availability
FR09	The software must enable the hospital staff to book rooms for several kinds of events
FR10	The software must enable the hospital staff to update a patient's schedule
FR11	The software must enable the hospital staff to view a list of all current and or past patients
FR12	The software must enable the hospital staff to sort a list of current and or past patients
FR13	The software must enable the hospital staff to send thank you emails to former patients
FR14	The software must enable the hospital staff to add event information after booking

3. Non-Functional Requirements

NFR01	The patient search must resolve in 5000 milliseconds
NFR02	Updates to schedules should be visible on all systems within 10 seconds
NFR03	Floor map GUI must update upon event being added within 10 seconds
NFR04	Floor map GUI must display 10 rooms at any time
NFR05	The database must be able to support upwards of 10,000 patients

NFR06	The GUI must display the status of rooms (occupied, booked, or open) within 1 second of opening or refreshing the interface, using distinct icons or color codes with real-time updates every 5 seconds for accuracy
NFR07	The software must be able to handle up to 100 simultaneous users
NFR08	The patient database must ensure data security by preventing unauthorized access, with a password policy that requires at least 8-character complexity for staff, and the system must withstand a minimum of 10,000 failed login attempts per minute without performance degradation or data compromise
NFR09	The system must be able to recover from unexpected errors within 5 seconds, 95% of the time, and log all errors with detailed information, including a timestamp, and error type
NFR10	Emails must be sent to patients within 30 minutes of discharge

4. Use Case Specification

<< Select **three** functional requirements and describe them in detail using use cases.>>

UC01 Name:	**User Search For Patients**
Description:	The software must enable the hospital staff to search for patients
Actor:	Hospital Staff
Entry condition:	The actor selects the search option on the home screen
Basic path:	1. The system presents the actor with a search screen **[PRO01]** containing: - Text input for first name - Text input for last name - Advanced search option - Button to initiate search 2. The actor fills in first and last name before initiating search **[A01][E01]** 3. The system presents the actor with a list of patients matching the first name and last name parameters followed by more patients whose names are similar to search parameters 4. The actor selects the patient whose information they want to view, and the system presents the following **[PRO03]**: a. First name, Last name (read only) b. Patient number (read only) c. Room number (read only) d. PCP (read only)

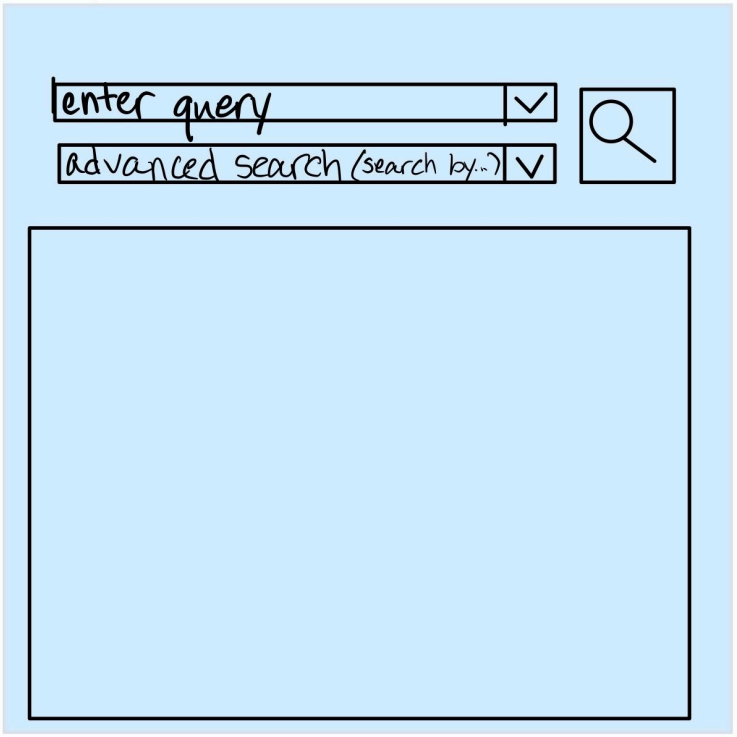
	e. A list of events which pertain to the patient (patient chart) (read only) f. A back button to return to home g. An edit button to update patient info 5. The actor selects the back button **[A02]** 6. The use case is concluded 7. The system returns to the home page
Alternative paths:	**[A01] The actor selects the advanced search option** 1. The system presents the advanced search screen that allows search input for patient **[PRO02]**: - Text input for room number - Text input for MRN (**M**edical **R**ecord **N**umber) - Text input for Admission date - Text input for PCP (**P**rimary **C**are **P**rovider) 2. The user inputs new search inputs as necessary before initiating search 3. Return to step 3 of the basic path **[A02] The actor selects the Edit option** 1. The system displays a form for editing the selected patient: - Name: First and Last (editable) - Patient number (editable) - Room number (editable) - PCP (containing the list of Hospital physicians) (read only) - Patient event / schedule(editable) - The options: - Confirm - Back 2. The actor enters patient data and selects the Confirm option. **[A04]** 3. The system verifies if the info is valid 4. The use case is concluded 5. The system returns to the home screen **[A03] The actor selects the discharge option**

<table>
<tr><td></td><td>1. The system confirms that the actor wants to discharge the patient
2. The system excludes the patient
3. The system returns to the basic path
[A04] The actor selects the option Back
1. The use case returns to step 1 of the alternative path [A01]</td></tr>
<tr><td>Exception paths:</td><td>[E01] The actor attempts an empty search
1. The system displays to the user that the search parameters were invalid
2. The system returns to step 2 of the basic path</td></tr>
<tr><td>Business Rules:</td><td>[BR01] . No patients can be scheduled for more than one event at a given time
[BR02] . All patients must have insurance information</td></tr>
<tr><td>Data description</td><td>
<table>
<tr><th>Name</th><th>Type</th><th>Length</th><th>Mask</th><th></th></tr>
<tr><td>Patient Name</td><td>String</td><td>50</td><td></td><td></td></tr>
<tr><td>SSN</td><td>String</td><td>11</td><td>XXX-XX-XXXX</td><td></td></tr>
<tr><td>Room Number</td><td>Integer</td><td>3</td><td></td><td>first digit: 0-9 second digit: 0-9 third digit: 0-9</td></tr>
<tr><td>MRN</td><td>String</td><td>12</td><td></td><td>medical record number</td></tr>
<tr><td>Admission date</td><td>int(?)</td><td>6</td><td>DD/MM/YY</td><td>DDMMYY</td></tr>
<tr><td>PCP</td><td>PCP object</td><td>n/a</td><td></td><td>String for name Attach times</td></tr>
</table>
</td></tr>
</table>

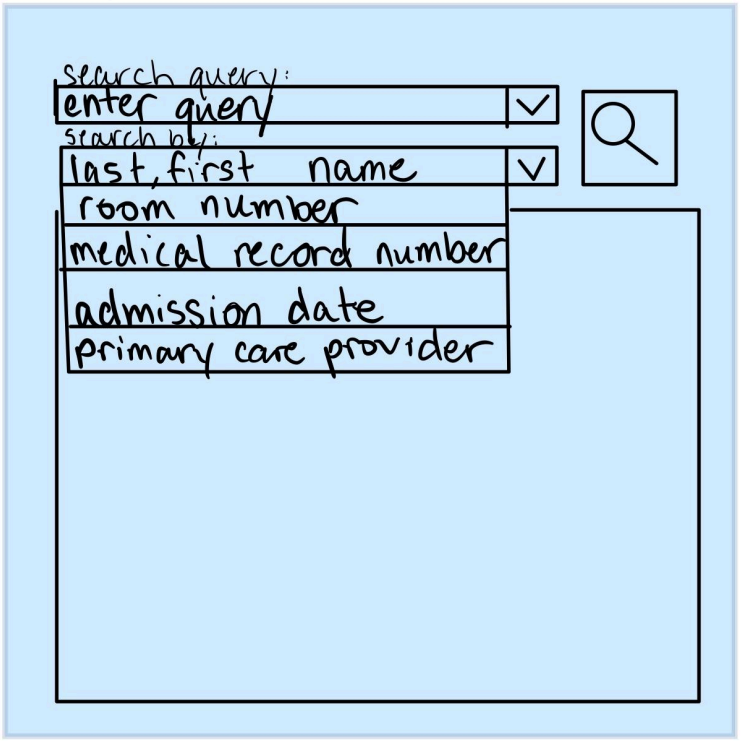
					doctor is occupied
	Insured?	boolean	1 (T or F/0 or 1)	Yes or No	
	Admission status	Boolean	1	Yes or No	
	Medical Chart	String	3000		For storing relevant medical information and events
	Event Schedule	Event object	n/a		For storing upcoming events

Prototype:

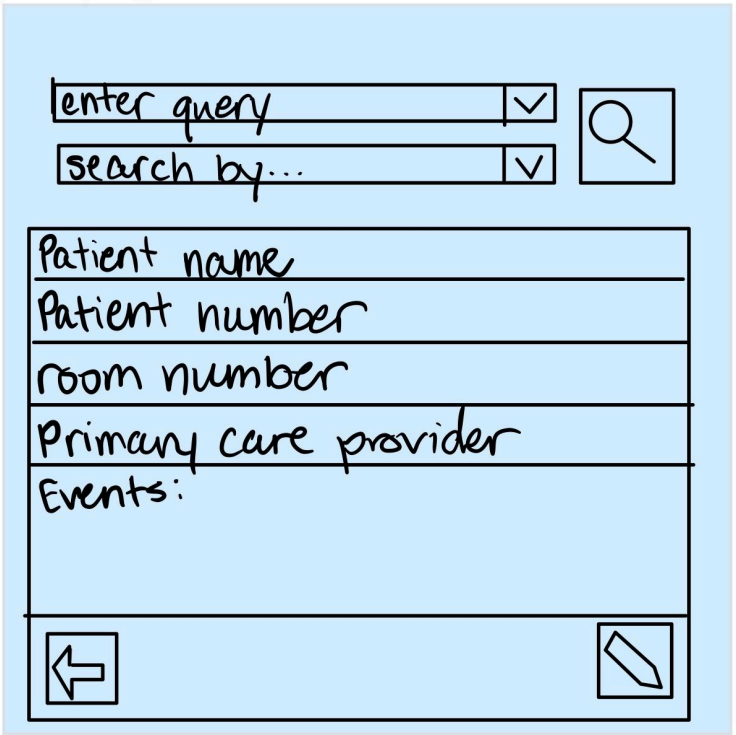
[PRO01]



[PRO02]


search query:
enter query
search by:
last, first name
room number
medical record number
admission date
primary care provider

[PRO03]


enter query
search by...
Patient name
Patient number
room number
Primary care provider
Events:

UC02 Name:	**View a Hospital Room**
Description:	The software allows hospital staff to view rooms by floor and see what events are currently happening, if the room is open or booked and see upcoming bookings
Actor:	Hospital staff
Entry condition:	The actor selects the room view option on the home page
Basic path:	1. The system displays the a visualization of the hospital floor plan including all rooms and their booking status **[PRO01]** a. Rooms will be green if they are available b. Rooms will be red if they are currently booked c. This page will have datetime selection to change viewing time of rooms d. This page will have event drop down selection to choose event e. This page will have a back button to return to home page 2. The actor selects the back option **[A01] [E01][E02]** 3. This use case is concluded 4. The system returns to the home page
Alternative paths:	**[A01] The actor changes the event dropdown** 1. The system will update to show rooms which would be available for a booking for the selected event a. The actor may change the starting date time for the booking at any time and the system will update by showing room availability starting at selected date time **[E01][E02]** 2. The actor clicks on a room to book **[A02][E03][E04]** 3. The system prompts the actor to confirm booking and the actor confirms **[A03]** 4. The system updates event database

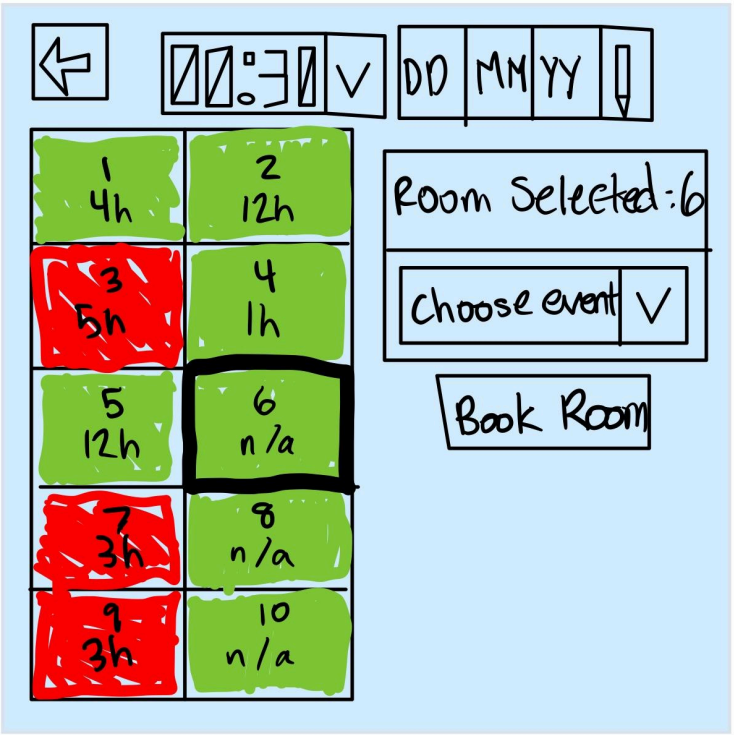
<table>
<tr><td></td><td>5. The system returns to A01 step 1 with updated visuals

[A02] The user selects the back option
1. The use case is concluded
2. The system returns to the home page

[A03] The user declines confirmation of booking

1. The system returns to A01 step 1</td></tr>
<tr><td>Exception paths:</td><td>[E01] Invalid date selection
1. The system prompts the actor to select a valid time
2. The system returns to the path and step it was on before the error path was taken

[E02] No valid reason for booking selected when trying to book
1. The system prompts the actor to select an event before booking
2. The system returns to step 1 of the basic path
[E03] No rooms available
1. The system prompts the actor that no rooms are available to be booked within the selected time period
[E04] No double booking
1. If the actor attempts to select a room that is already booked, the system will not allow the actor to select it.</td></tr>
<tr><td>Business Rules:</td><td>[BR01] . All event bookings must incorporate an extra 30 minute time frame at the beginning and ending of the bookings before another event booking can be made
[BR02] . All bookings must have a valid patient id assigned to it</td></tr>
<tr><td>Data description</td><td>
<table>
<tr><th>Name</th><th>Type</th><th>Length (characters)</th><th>Mask</th><th>Description</th></tr>
<tr><td>Room Booked value</td><td>Boolean</td><td>1 (T or F)
(1 or 0)</td><td>Booked (T) or Open (F)</td><td>Red or green room</td></tr>
</table>
</td></tr>
</table>

	Time Start	int	4	XX:XX	first digit: 0,1,2 second, third, fourth: 0-9
	Time End	int	4	XX:XX	1: 0,1,2 2,3,4: 0-9
	Room number	int	3		
	Room Type	string	8		OR/ER/ICU
	Patient MRN	string	12		
Prototype:	**[PRO01]** 				

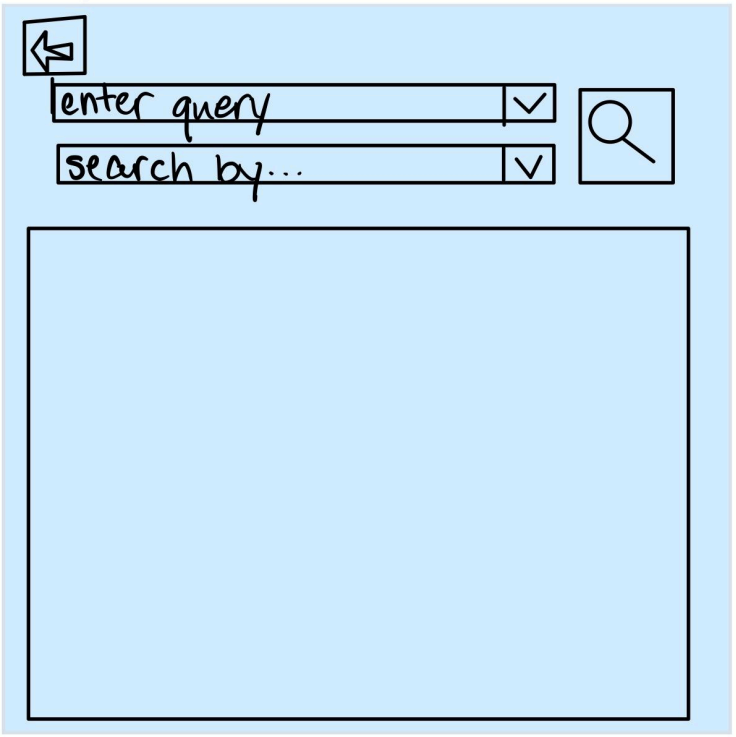
UC03 Name:	**Manage Events**
Description:	The software must allow staff to view, manage, and search events
Actor:	Hospital Staff
Entry condition:	The actor selects the manage events button on the home screen.
Basic path:	1. The system presents the hospital staff with all upcoming events sorted by upcoming time. Hospital staff are presented with field at top of screen a. Input for start time b. Input for end time c. Drop down for search parameter d. Search button e. Back button 2. The actor inputs the room number they are searching for, then hits search **[A01]** 3. The system presents a table containing a list of events corresponding with the given room number, sorted by earliest to latest **[E01][E02]** 4. The actor selects which event they were interested in 5. The system presents the following: a. Event information i. Event room number (read only) ii. Event type (read only) iii. Start time (read only) iv. End time (read only) v. Assigned Personnel (read only) vi. Patient ID number (read only) vii. Additional information/comments (read only) b. Back button c. Edit button d. Delete button 6. If changes are required, the actor selects the Edit option **[A01][A02][A03]** 7. The system presents the following: a. Event information

	i. Event room number (editable) ii. Event type (editable) iii. Start time (editable) iv. End time (editable) v. Assigned Personnel (editable) vi. Patient ID number (editable) vii. Additional information/comments (editable) b. Back button c. Save Changes button 8. The actor selects the save changes option **[A05]** 9. The system returns to step 5
Alternative paths:	**[A01] The actor selects the back option** 1. The use case is concluded 2. The system returns to the home page **[A02] The actor selects the delete event option** 1. The system prompts the actor to confirm deletion of event and the actor confirms **[A04]** 2. The system deletes the events and schedules are updated 3. The system returns to basic path step 1 **[A03] The actor edits the information** 1. The actor has selected the edit option and is presented with these editable fields a. Event information (editable) i. Event room number (editable) ii. Event type (editable) iii. Start time (editable) iv. End time (editable) v. Assigned Personnel (editable) vi. Patient ID number (editable) vii. Additional information/comments (editable) **[A04] The actor declines confirmation** 1. The system returns to basic path step 5 **[A05] The actor selects back while editing information**

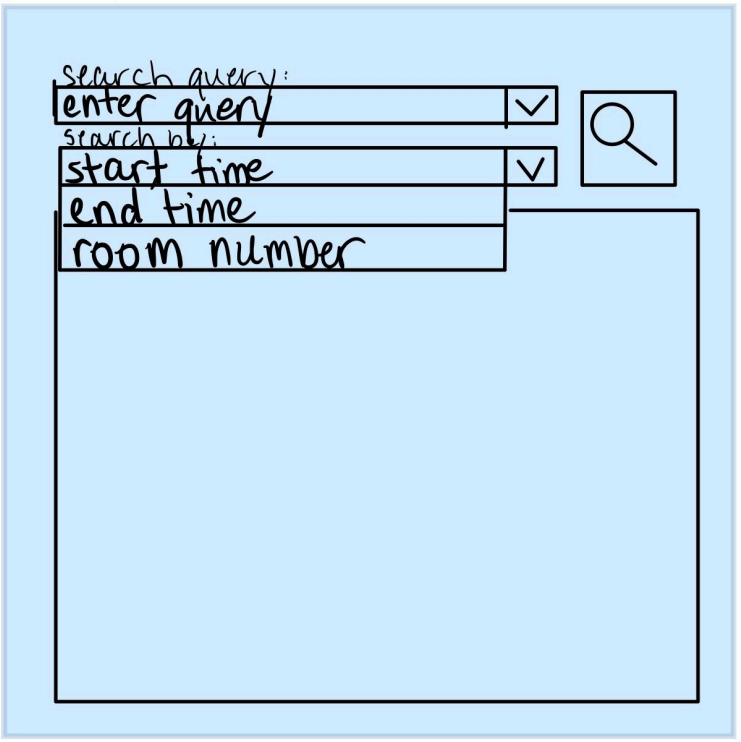
<table>
<tr><td></td><td>
1. The system asks the user is they would like to discard changes made
2. The user selects no
3. The system returns to basic path step 7

[A06] The actor selects yes and discards changes

1. The system discards changes made and returns to basic path step 5
</td></tr>
<tr><td>Exception paths:</td><td>
[E01] Incorrect data type as search input
1. System displays an error message saying that input does not match expected parameters.
2. System prompts user with confirmation button
3. System returns to step 1 of the basic path
[E02] No event found matching criteria
1. System displays error message saying that input does not return any search results
2. System prompts user with confirmation button
3. System returns to step 1 of the basic path
</td></tr>
<tr><td>Business Rules:</td><td>[BR01] . Only searchable patient information is MRN
[BR02] . All event must have at least one hospital staff assigned to it</td></tr>
<tr><td>Data description</td><td>
<table>
<tr><th>Name</th><th>Type</th><th>Length</th><th>Mask</th><th></th></tr>
<tr><td>Start time</td><td>int</td><td>4</td><td>XX:XX</td><td></td></tr>
<tr><td>End time</td><td>int</td><td>4</td><td>XX:XX</td><td></td></tr>
<tr><td>PCP</td><td>object</td><td>n/a</td><td></td><td></td></tr>
<tr><td>Event description</td><td>string</td><td>2000</td><td></td><td></td></tr>
<tr><td>Patient MRN</td><td>string</td><td>12</td><td></td><td></td></tr>
<tr><td>Room number</td><td>int</td><td>3</td><td></td><td></td></tr>
<tr><td>Event Type</td><td>string</td><td>10</td><td></td><td></td></tr>
</table>
</td></tr>
<tr><td>Prototype:</td><td>[PRO01]</td></tr>
</table>


enter query
search by...

[PRO02]


search query:
enter query
search by:
start time
end time
room number

[PRO03]


enter query
search by...
Room number
Event type
start time - end time
assigned personnel
patient ID number
additional info