1. Problem statement

We need a software to manage a hospital’s information through user interface connected to a database and controllers that manage different system requests base on the selected request.

1. Use cases

* Add patient
* Checkout patient
* Search patient
* Modify patient
* Schedule appointment
* Reserve bed for patient
* Add doctor
* Remove doctor
* Add department
* Delete department

1. Use case diagram
2. Use case description

|  |  |
| --- | --- |
| Use Case Name | Add Patient |
| Goal in Context | Adding a new patient to the hospital |
| Pre-Conditions | Administrator is logged in |
| Successful End Condition | Patient added |
| Failure End Condition | Patient wasn’t added |
| Primary Actor | Administrator |
| Secondary Actor | None |
| Trigger | “Add patient” button clicked |
| Input | Patient information |
| Output | None |
| Main Flow | 1. Administrator selects “Add patient” button  2. System shows “Add patient” form  3.Administrator adds patient information into the form  4.Administrator clicks “Add” button  5 Systems confirms if patient information is valid  6. Patient added to hospital system  7.Success message appears |
| Extensions | 5.1 Systems rejects invalid patient information  5.2 Show error message “Invalid patient information |

|  |  |
| --- | --- |
| Use Case Name | Checkout patient |
| Goal in Context | Checking out patient from hospital |
| Pre-Conditions | Administrator is logged in and patients exists and is currently an active patient |
| Successful End Condition | Patient checked out |
| Failure End Condition | Patient wasn’t able to checkout |
| Primary Actor | Administrator |
| Secondary Actor | None |
| Trigger | “Checkout patient” button clicked |
| Input | Patient information, services offered to patient and medical costs. |
| Output | Medical bill |
| Main Flow | 1. Administrator selects “Checkout patient” button  2. System shows “Checkout patient” form  3. Administrator adds patient information and medical costs into the form  4.Administrator clicks “Checkout” button  5. Systems confirms if patient exists  6. Systems confirms that patient is currently an active patient  7. Patient is marked as an inactive patient in the hospital system  9.Medical bill printed  8.Success message appears |
| Extensions | 5.1. No patient found in system  5.2 Failed message appears showing “No patient found”  6.1 Patient is found to be an inactive patient.  6.2 Failed message appears showing “Patient is currently inactive and has no bills”  8.1 Printer not working  8.2 Show error message “Problem with printer” |

|  |  |
| --- | --- |
| Use Case Name | Schedule appointment |
| Goal in Context | Schedule an appointment between a patient and a doctor |
| Pre-Conditions | Administrator is logged in, patients exists, doctor exists |
| Successful End Condition | Appointment reserved |
| Failure End Condition | Appointment not reserved |
| Primary Actor | Administrator |
| Secondary Actor | None |
| Trigger | “Schedule appointment” button clicked |
| Input | Patient information and doctor’s information and an appointment note |
| Output | None |
| Main Flow | 1. Administrator selects “Schedule appointment” button  2. System shows “Schedule appointment” form  3. Administrator adds patient information and doctor’s information into the form  4.Administrator clicks “Schedule” button  5. Systems confirms if patient exists  6. Systems confirms if doctor exists  7. System reserves appointment  8. Success message appears showing appointment date |
| Extensions | 5.1. No patient found in system  5.2 Failed message appears showing “No patient found”  6.1. No doctor found in system  6.2 Failed message appears showing “No doctor found” |

|  |  |
| --- | --- |
| Use Case Name | Add doctor |
| Goal in Context | Adding a new doctor to the hospital |
| Pre-Conditions | Administrator is logged in and there is a department for the doctor |
| Successful End Condition | Doctor added |
| Failure End Condition | Doctor wasn’t added |
| Primary Actor | Administrator |
| Secondary Actor | None |
| Trigger | “Add doctor” button clicked |
| Input | Doctor’s information and department’s name |
| Output | None |
| Main Flow | 1. Administrator selects “Add doctor” button  2. System shows “Add doctor” form  3.Administrator adds doctor’s information and department’s information into the form  4.Administrator clicks “Add” button  5. Systems confirms if department exists  6. Doctor added to hospital system  7. Success message appears |
| Extensions | 5.1. No department found for doctor  5.2 Failed message appears showing “Department doesn’t exist” |

|  |  |
| --- | --- |
| Use Case Name | Add Department |
| Goal in Context | Adding new Department to Hospital |
| Pre-Conditions | The admin must be logged in to the System |
| Successful End Condition | Department added successfully |
| Failure End Condition | Department failed to be added |
| Primary Actor | Admin |
| Secondary Actor | None |
| Trigger | The Admin clicks Add Department Button |
| Input | All Department Information |
| Output | None |
| Main Flow | 1- The Admin selects Add Department Button  2- The System shows Add Department Form  3- The Admin enters Department Information  4- The Admin clicks the Add Button  5- The System checks that there is no existing department with the same name  6- The System verifies that that there is no existing department with the same name  7- The Department is added Successfully  8- The system shows a form indicating that the Department is Added |
| Extensions | 6.1- The System found an existing Department with the same name  6.2- The System Rejects Adding this Department  6.3- The System shows a form indicating that the Department wasn’t Added |

|  |  |
| --- | --- |
| Use Case Name | Delete Department |
| Goal in Context | Deleting an Existing Department in Hospital |
| Pre-Conditions | 1- The admin must be logged in to the System  2- The Department already exists in the Hospital System |
| Successful End Condition | Department deleted successfully |
| Failure End Condition | Department failed to be deleted |
| Primary Actor | Admin |
| Secondary Actor | None |
| Trigger | The Admin clicks Delete Department Button |
| Input | Department Name |
| Output | None |
| Main Flow | 1- The Admin selects Delete Department Button  2- The System shows Delete Department Form  3- The Admin enters Department Name  4- The Admin clicks the Delete Button  5- The System checks that the Department Name exists in the System  6- The System verifies that that the Department exists  7- The Department is deleted Successfully  8- The system shows a form indicating that the Department is Deleted |
| Extensions | 6.1- The System didn’t find the Department Name in the System  6.2- The System Rejects Deleting this Department  6.3- The System shows a form indicating that the Department doesn’t exist in System and The Deletion process is Rejected |

|  |  |
| --- | --- |
| Use Case Name | Search Patient |
| Goal in Context | Finding certain patient data |
| Pre-Conditions | 1- The admin must be logged in to the System  2- The Patient must exist in the System |
| Successful End Condition | Patient’s Data is shown to the Admin |
| Failure End Condition | Failed to show patient’s Data to the Admin |
| Primary Actor | Admin |
| Secondary Actor | None |
| Trigger | The Admin clicks Search Patient Button |
| Input | Patient’s Name |
| Output | None |
| Main Flow | 1- The Admin selects Search patient Button  2- The System shows Search patient Form  3- The Admin enters Patient Name  4- The Admin clicks the Search Button  5- The System checks that there is an existing patient satisfying the entered Name  6- The System verifies that that there is an existing patient satisfying this Name  7- The patient’s Data is shown to the Admin |
| Extensions | 6.1- The System didn’t find a patient satisfying the entered Name  6.2- The System shows a form indicating that there is no patient with such Name |

|  |  |
| --- | --- |
| Use Case Name | Reserve Bed for Patient |
| Goal in Context | Reserving a bed to the Patient |
| Pre-Conditions | 1- The admin must be logged in to the System  2- The Patient must exist in the System  3- There is a free Bed in the Hospital |
| Successful End Condition | A Bed is allocated to the Patient |
| Failure End Condition | Failed to allocate a Bed to the Patient |
| Primary Actor | Admin |
| Secondary Actor | None |
| Trigger | The Admin clicks Reserve Bed Button |
| Input | Patient’s Name |
| Output | None |
| Main Flow | 1- The Admin selects Reserve Bed Button  2- The System shows Reserve Bed Form  3- The Admin enters Patient Name or ID  4- The Admin enters Bed Number  5- The Admin clicks the Reserve Button  6- The System checks that there is an existing patient satisfying the entered Data  7- The System checks that there is a Bed with such a Number  8- The System checks that this Bed is Available  9- The System verifies that that there is an existing patient satisfying this Data  10- The System verifies that there is an available Bed  11- A form is shown to the Admin showing the reserved Bed Number |
| Extensions | 6.1- The System didn’t find a patient satisfying the entered Data  6.2- The System shows a Form indicating that Allocating Bed in Rejected  7.1- The System didn’t find a Bed with such Number  7.2- The System shows a Form indicating that Allocating Bed in Rejected  8.2- The System found that this Bed is not Available  8.3- The System shows a Form indicating that Allocating Bed in Rejected |