User Case Descriptions

**Use case:** Login

**Actors:**  Receptionist

**Goal:**  To log into the system

**Overview:**

When the user launches the application the system, the system should go to the login screen, requesting the user to enter their username and password. The system recognises the user’s details and allows them to login to their account.

**Typical course of events**

|  |  |
| --- | --- |
| **Actor action:** | **System response:** |
| 1. User open the login screen |  |
|  | 1. Display the login screen |
| 1. User enter their user and password and click login |  |
|  | 1. Check the user’s Username and password |
|  | 1. Logs the user on to their user account. |

**Alternative courses**

Step 2: The system encounters and error and doesn’t display the login screen correctly.

Step 4: The login details are not recognised by the system, the system doesn’t allow the user to login.

**Use case:** Check individual nurse/GP availability

**Actors:**  Receptionist

**Goal:**  To check individual nurse and GP availability on specific days

**Overview:**

When the user clicks the button ‘GP/nurse availability’, the system should then bring up the correct screen asking the user to select the GP or nurse name from a drop down list. Then ask the user what day they want to search if the GP or nurse is available on. The system should search for the GP or nurse straight away and display the correct information on the screen.

**Typical course of events**

|  |  |
| --- | --- |
| **Actor action:** | **System response:** |
| 1. User clicks GP/nurse availability |  |
|  | 1. Displays the input page for nurse/GP availability |
| 1. User selects nurse or GP’s last name from a drop down box |  |
| 1. Users selects the date they want to see the availability of GP or nurse |  |
|  | 1. Checks if all input fields have been filled in |
|  | 1. Checks if the GP or nurse is in the selected day |
|  | 1. Checks if the Dr surgery is open that day |
|  | 1. Displays the Nurse or GP availability details on screen |

**Alternative course**

Step 1: The system doesn’t respond when the GP/nurse availability button is clicked, this may terminate the application.

Step 3: The Dr or nurse might not be on the system, resulting in the user not been able to check their availability.

Step 8: The system encounters an error when displaying the availability of a GP or nurse, thus not allowing the user to see the GP or nurse availability.

**Use case:** GP/Nurse timetable

**Actors:**  Receptionist

**Goal:**  To check nurse and GP availability in the surgery

**Overview:**

When the user clicks the button ‘GP/nurse timetable’, the correct screen will be displayed showing the user the timetable for all GP and nurses who are in the surgery that week. The user should also be able to view the timetable for the following week.

**Typical course of events**

|  |  |
| --- | --- |
| **Actor action:** | **System response:** |
| 1. User clicks GP/nurse timetable |  |
|  | 1. Search all staff who are in the surgery that week |
|  | 1. Displays the timetable for all nurse and GPs in the surgery that week |
| 1. User clicks to view the next following week for the availability of GP and nurses |  |
|  | 1. Search all staff who are in the surgery the following week |
|  | 1. Displays the timetable for all nurse and GPs in the surgery for the following week |

**Alternative course**

Step 2, 5: The system doesn’t find any members of staff in for the current week or next week, thus not allowing the user to view the staff timetable.

Step 3, 6: The system encounters an error and doesn’t allow the user to view the timetable due to it not been displayed correctly.

**Use case:** Register patient

**Actors:**  Receptionist

**Goal:**  To add a new patient to the system

**Overview:**

When the user clicks the ‘register a new patient’ button, the system will display input fields and drop down menus for the patients details. The system will give the new patient a unique ID so the data can be saved. The system will then check if the user has entered all the details in every mandatory field before saving the data.

**Typical course of events**

|  |  |
| --- | --- |
| **Actor action:** | **System response:** |
| 1. The user clicks ‘register a new patient’ |  |
|  | 1. Displays the register patient screen and input boxes and drop down menu’s |
|  | 1. Give the patient a unique ID |
|  | 1. Asks user to enter the patient details in the correct boxes |
| 1. The user enters all the relevant patient details into the fields. |  |
| 1. The user clicks the save button to save the patients details |  |
|  | 1. Checks if any of the mandatory fields are empty, displays error message if they are |
|  | 1. Saves the details of the patient in the database |

**Alternative course**

Step 2: The system may not display all the input fields and drop down menus in the correct format resulting in the user not been able to enter the new patient details.

Step 3: The system may not give the user a unique ID if it encounters an error, which could lead to the patient details not been saved correctly.

Step 7: The system may not display an error message if any fields are empty which may result in important patient details not been saved correctly.

Step 8: The system may not save the patient details in the database, which will result in the patient not been on the system.

**Use case:** Search for patient

**Actors:**  Receptionist

**Goal:**  To search for a patient on the system

**Overview:**

When the user clicks ‘search for a patient’, the system will display the correct screen asking the user to enter the patients: last name, DOB and post code. The system will then search for the user based on these three details, and then display all the users’ details on the screen.

**Typical course of events**

|  |  |
| --- | --- |
| **Actor action:** | **System response:** |
| 1. User clicks ‘search for a patient’ |  |
|  | 1. Displays the search for a patient screen and the input fields for patient details |
| 1. The user enter the patients: last name, DOB, post code |  |
|  | 1. Checks if all details have been entered into the fields |
|  | 1. Searches for the patient in the database |
|  | 1. Displays the patient details on screen |

**Alternative course**

Step 2, 6: The system may not display the pages correctly resulting in the user not been able to enter the patient’s details or been able to view the patient’s details.

Step 3: The user may forget to enter the details or may not enter the details correctly which will lead to the system not been able to find the patient.

Step 5: The system may not be able to find the patient resulting in the user having to recheck the details that they have entered.

**Use case:** Extend the patient medication

**Actors:**  Receptionist <<extends>>

**Goal:**  Allow the receptionist to extend the patients medication

**Overview:**

The user will click ‘extend patients medication’, the system will display all the patients’ medication and when they finish their medication. The user can then extend a particular medication to a specific day, then save the new medication details into the system.

**Typical course of events**

|  |  |
| --- | --- |
| **Actor action:** | **System response:** |
| 1. User clicks ‘extend patient medication’ |  |
|  | 1. Displays the patients medication and length of time till the finish the medication |
| 1. User selects the medication they wish to extend |  |
| 1. User selects the amount of time they wish to extend the medication |  |
| 1. User saves the data by clicking the ‘save’ button |  |
|  | 1. Checks the appropriate fields have been filled in |
|  | 1. The data is saved to the database |

**Alternative course**

Step 2: The page may not be displayed correctly due to an error, which may not allow the user to see the information on the page.

Step 6: The system may encounter an error when checking all the fields have filled in, which could result in the medication not been extended for the patient or the patient been on the medication for too long.

Step 7: The data may not save on the system due to an error, resulting in the relevant medication extensions not been changed.

**Use case:** Print the results of patients test

**Actors:**  Receptionist <<extends>>

**Goal:**  To allow the receptionist to print the patients test results

**Overview:**

The user clicks the ‘print patient test results’, the test results for the patient are then displayed on the screen. The user can then select the test result they want to print, and then print the test result.

**Typical course of events**

|  |  |
| --- | --- |
| **Actor action:** | **System response:** |
| 1. User clicks ‘print patients test results’ |  |
|  | 1. Displays the patients test results on screen |
| 1. User selects the test results they wish to be printed |  |
| 1. User clicks ‘print’ |  |
|  | 1. The results are put into a standard printing format and then it prints the results |

**Alternative course**

Step 2: The page may not be displayed correctly due to an error, which may not allow the user to see the information on the page.

Step 5: The system may not print the page according to the page template which may the illegibility of the information.

**Use case:** Delete a patient

**Actors:**  Receptionist <<extends>>

**Goal:**  To allow the receptionist to delete a patient from the system

**Overview:**

The user selects ‘delete a patient’; all the patients’ details are then displayed on the screen, the user the clicks ‘delete a patient’, the system asks if the user is sure they want to delete the patient. The patient’s deleted details are then sorted in a separate database and will be deleted from the system after a certain amount of time.

**Typical course of events**

|  |  |
| --- | --- |
| **Actor action:** | **System response:** |
| 1. User clicks ‘delete a patient’ |  |
|  | 1. Displays all the patients details on screen |
| 1. The user clicks ‘delete the patient’ |  |
|  | 1. The system asks if the user is sure they want to delete a patient |
| 1. The user can click ‘yes’ or ‘cancel’ |  |
|  | 1. The patient’s details are then deleted from the system and stored in a separate part of the database for a certain amount of time. |

**Alternative course**

Step 2: The page may not be displayed correctly due to an error, which may not allow the user to see the information on the page.

Step 4: The system may not display this message due to an error, which may result in the patients been deleted from the system.

Step 6: The system may encounter an error when storing the deleted patient’s details in the system, and as a result the details may be permanently deleted from the system.

**Use case:** Change patient details

**Actors:**  Receptionist <<extends>>

**Goal:**  To allow the receptionist to change patients details in the system

**Overview:**

The user clicks ‘change patient details’, the system brings up the change details page. The user then can change any details of the patients, and then save the changed details. The system should overwrite the old patient’s details in the system with the new ones.

**Typical course of events**

|  |  |
| --- | --- |
| **Actor action:** | **System response:** |
| 1. User clicks ‘change patient details’ |  |
|  | 1. Displays all the patients details on screen |
| 1. The user edits the patients details |  |
| 1. The user clicks ‘save patient details’ |  |
|  | 1. Checks all the fields have been filled in |
|  | 1. The patients details are saved in the database, overwriting the old patient details in the system |

**Alternative course**

Step 2: The page may not be displayed correctly due to an error, which may not allow the user to see the information on the page.

Step 6: The system may encounter an error when storing the patient’s new details into the system, or may not overwrite the existing patient’s details resulting in the database not been correct and the patient having incorrect details.

**Use case:** Cancel patient appointment

**Actors:**  Receptionist <<extends>>

**Goal:**  To allow the receptionist to cancel patient appointments

**Overview:**

The user clicks ‘cancel patient appointment’, the system then displays the appointments that patient for that week. The user can the select the patients appoint they wish to cancel, then cancel it. The system should then update itself and display the cancelled appointment as now available.

**Typical course of events**

|  |  |
| --- | --- |
| **Actor action:** | **System response:** |
| 1. User clicks ‘cancel patient appointments’ |  |
|  | 1. Displays the patients appointments for that Week |
| 1. The user selects the appointment they wish to cancel |  |
| 1. The user then clicks ‘cancel appointment’ |  |
|  | 1. Asks if the user is sure they want to cancel the appointment |
| 1. The user can click ‘yes’ or ‘cancel’ |  |
|  | 1. Cancels the appointment from the system |
|  | 1. Displays the time as available on the ‘Booking page’ |

**Alternative course**

Step 2: The page may not display correctly resulting in the user not been able to view the page.

Step 3: The appointment may not be there, which would not allow the user to cancel the patient’s appointment.

Step 5: The system may not display this message due to an error, which may result in the patients appointment been cancelled.

Step 8: The system may not reallocate the deleted time back into the system, which in turn would not allow patients to book that appointment again.

**Use case:** Change appointment

**Actors:**  Receptionist <<extends>>

**Goal:**  To allow the receptionist to change patient appointment

**Overview:**

The user clicks ‘change patient appointment’, the system then displays the available appointments for the week. The user then selects the appointment they wish to change and a new appointment the patient wants. The user then saves the database. The system should delete the old time and replace it with the new time, then reallocate that old time back into the system.

**Typical course of events**

|  |  |
| --- | --- |
| **Actor action:** | **System response:** |
| 1. User clicks ‘change patients appointment’ |  |
|  | 1. Displays the patients available appointments for that week |
| 1. The user select what appointment they wish to change |  |
| 1. The user selects the new appointment the patient wishes to have |  |
| 1. The user selects the Dr/nurse the patient wants |  |
| 1. User clicks ‘save appointment’ |  |
|  | 1. Checks all the fields have been filled in |
|  | 1. Deletes the old appointment and save the new one |
|  | 1. Puts the old appointment time back on the ‘Booking page’ |

**Alternative course**

Step 2: The page may not display correctly resulting in the user not been able to view the page.

Step 8: The system may encounter an error when changing the patient’s appointment, this could result in the old patient’s time not been overwritten by the system.

Step 9: The system may not reallocate the changed time back into the system, which in turn would not allow patients to book that appointment again.

**Use case:** Book a patient

**Actors:**  Receptionist <<extends>>

**Goal:**  To allow the receptionist to book a patient in for an appointment

**Overview:**

The user clicks ‘book patient appointment’, the system then displays the available appointments for the week. The user then selects the date, time and Dr/ nurse the patient requires and clicks save appointment. The system should then save the appointment into the database and not allow that time, date and Dr/nurse to be booked again.

**Typical course of events**

|  |  |
| --- | --- |
| **Actor action:** | **System response:** |
| 1. User clicks ‘Book patient appointment’ |  |
|  | 1. Displays the patients available appointments for that week |
| 1. The user selects the Date and time the user desires |  |
| 1. The user selects the Dr/nurse the patient wants |  |
| 1. The user then clicks ‘save appointment’ |  |
|  | 1. Checks all the fields have been filled in |
|  | 1. Saves the appointment in the database |

**Alternative course**

Step 2: The page may not display correctly resulting in the user not been able to view the page.

Step 3, 4: The date and time or Dr or nurse may not be available when the patient wants them, which could mean the patient isn’t seen that day.

Step 7: The system may not save the patients appointment, resulting in the patient not been in the database and not been seen by the Dr or nurse. This will also allow that time to be rebooked by another patient.