





| **USE CASE ID: UCD\_RCP\_001** | **USE CASE NAME: LOGIN** |
| --- | --- |
| **USE CASE DESCRIPTION** | This Use case enables functions of Receptionist in logging into the system |
| **ACTOR** | * Primary Actor: Receptionist * Secondary Actor: NA |
| **PRE- CONDITIONS** | * Active internet connection * System should be up & running * Login credentials |
| **BASIC FLOW** | * 1. Receptionist opens the application   2. For Login:      1. Enter User ID      2. Enter Password      3. Click on ‘Login’ functionality      4. If Credentials are valid then redirects to Home Page.      5. Use case ends successfully   3. For Reset Password: If Receptionist forgets the password      1. Select ‘Reset Password’ Functionality      2. Enters New password      3. Re-enters New Password      4. Enter Mobile Number      5. Click on “Send OTP to Mobile Number” Functionality      6. Enter the OTP (received on the Mobile Number) in the OTP Field      7. Click on “Submit” Functionality      8. System validates the OTP. Once OTP is valid then “Your password has been successfully changed”      9. Use case ends successfully |
| **ALTERNATE FLOW** | * 1. Invalid User ID:      1. In Step 1.2.1, If Invalid User ID entered, display “Invalid User ID. Please re-enter User ID”      2. Resume from Step 1.2.1   2. Invalid Password:      1. In Step 1.2.2, If Invalid Password entered, display “Invalid Password. Please re-enter Password”      2. Resume from Step 1.2.2   3. OTP not received:      1. In Step 1.3.5, if Receptionist did not receive OTP      2. Use case ends with a failure condition   4. Invalid OTP:      1. In Step 1.3.8, if OTP is Invalid      2. Use case ends with a failure condition |
| **EXCEPTIONAL FLOW** | * 1. If Invalid User ID entered more than 2 times,      1. System displays “ID blocked. Please contact Admin to unblock”      2. Use case ends with a failure condition   2. If Invalid Password entered more than 2 times      1. System displays “ID blocked. Please contact Admin to unblock”      2. Use case ends with a failure condition   3. If Reset Password is performed more than 2 times on the same day      1. System displays “ID blocked. Please contact Admin to unblock”      2. Use case ends with a failure condition   4. If server is down and system fails to accept input, then valid error condition should be displayed and use case remains a failure |
| **POST CONDITION** | * **Successful Execution:** Receptionist successfully logs into the system * **Unsuccessful Execution**: Receptionist cannot perform the action, due to system failure |
| **INPUTS** | * User ID * Password * OTP * New Password |
| **OUTPUTS** | * Login to system * Reset Password |
| **BUSINESS RULES** | 1. Receptionist will be able to access the system only during operating business hours. 2. Unique User ID is allotted for each employee 3. User ID and Password of Receptionist are created by the Admin |
| **ASSUMPTIONS** | NA |
| **CONSTRAINTS** | * User ID should be Employee ID * Password should be 8 characters * User ID and Password are alphanumeric |
| **KEY SCENARIOS** | Respective User ID and Password Dashboard should be displayed |
| **DEPENDENCIES** | NA |
| **SPECIAL REQUIREMENTS** | NA |
| **WIREFRAME REFERENCE** | NA |

| **USE CASE ID: UCD\_RCP\_002** | **USE CASE NAME: UPDATE ATTENDANCE** |
| --- | --- |
| **USE CASE DESCRIPTION** | This Use case enables functions of Receptionist in updating attendance for the day |
| **ACTOR** | * Primary Actor: Receptionist * Secondary Actor: Biometric Machine |
| **PRE- CONDITIONS** | * Active internet connection * Biometric Machine should be up & running * Receptionist Fingerprint is saved in the Biometric Machine |
| **BASIC FLOW** | * 1. Receptionist gives an input of fingerprint for scanning to Biometric Machine   2. Fingerprint is validated by Biometric Machine   3. If Fingerprint is valid, Entry/ Exit time of Receptionist is recorded by Biometric Machine   4. Use case ends successfully |
| **ALTERNATE FLOW** | * 1. Invalid fingerprint:      1. If in Step 2.2, the fingerprint input is not recognized by Biometric Machine, display “Fingerprint not valid. Please try again” and resume from Step 2.1 of basic flow.      2. Usecase ends successfully.   2. Manual Time Sheet Entry:      1. If in Step 2.3, the Biometric is unable to recognize the employee fingerprint, due to technical issues, employee updates through system login      2. Enters User ID      3. Enter Password      4. Open “Update Attendance”      5. Feeds the Entry and Exit Times      6. Usecase ends successfully. |
| **EXCEPTIONAL FLOW** | * 1. If Invalid fingerprint is performed more than 2 times      1. System displays “Fingerprint not recognized. Please contact Admin to unblock”      2. Use case ends with a failure condition   2. If server is down and Biometric Machine fails to accept input, then valid error condition is displayed and use case remains a failure. |
| **POST CONDITION** | * **Successful Execution**: Receptionist successfully updates attendance * **Unsuccessful Execution**: Receptionist cannot perform the action, due to system failure |
| **INPUTS** | * Fingerprint * Start Time * End Time * User ID * Password |
| **OUTPUTS** | * Attendance is updated |
| **BUSINESS RULES** | * Receptionist will be able to access the Biometric Machine only during operating business hours. * Entry/Exit Times are recorded by Biometric Machine whenever input is given |
| **ASSUMPTIONS** | * Receptionist must give fingerprint while Entry and Exit Times * Attendance are updated automatically after the input |
| **CONSTRAINTS** | * Only one fingerprint per employee * No ID Cards are accepted for timesheet updating |
| **KEY SCENARIOS** | * Relevant Attendance should be updated when Fingerprint is validated by Biometric |
| **DEPENDENCIES** | NA |
| **SPECIAL REQUIREMENTS** | NA |
| **WIREFRAME REFERENCE** | NA |

| **USE CASE ID: UCD\_RCP\_003** | **USE CASE NAME: SEARCH PATIENT DETAILS** |
| --- | --- |
| **USE CASE DESCRIPTION** | This Use case enables functions of Receptionist in searching for patient details |
| **ACTOR** | * Primary Actor: Receptionist * Secondary Actor: NA |
| **PRE- CONDITIONS** | * Active internet connection * System should be up & running * Registered Patient details are saved in the database * Valid Login Credentials |
| **BASIC FLOW** | * 1. Login to the system   2. Selects ‘Search Patient’ functionality   3. Fills the Patient details Field, based on details given by patient      1. Option-1: Patient ID (or)      2. Option-2: Patient Name & Phone Number   4. Select ‘Get Patient Information’ functionality   5. If Patient Information are found, then displays Patient Information   6. Use case ends successfully |
| **ALTERNATE FLOW** | * 1. Invalid Patient Details Input:      1. Invalid Patient ID:         1. In Step 3.4.1, if the Patient ID input is incorrect, display “Patient ID not found. Please try again” and resume from Step 3.2.1         2. Use case ends successfully      2. Invalid Patient Name & Phone Number:         1. In Step 3.4.2, if Patient Name and Phone Number inputs are incorrect, display “Patient Name or Phone number not found. Please try again” and resume from Step 3.2.2         2. Use case ends successfully   2. Patient Details not found:      1. In Step 3.5, if the Patient Information not found, display “Patient Information not found. Please register” and resume from Step 3.2.1      2. Use case ends successfully |
| **EXCEPTIONAL FLOW** | 3.9. If server is down and system fails to accept input, then valid error condition should be displayed and use case remains a failure. |
| **POST CONDITION** | * **Successful Execution**: Receptionist successfully views Patient Information * **Unsuccessful Execution**: Receptionist cannot perform the action, due to system failure |
| **INPUTS** | * User ID * Password * Patient ID * Patient Name * Phone Number |
| **OUTPUTS** | * Patient ID * Patient Name * Patient Age * Patient Gender * Patient Phone Number * Patient Email * Last Visited Date * Medical History * Prescription History |
| **BUSINESS RULES** | * Receptionist will be able to access the system only during operating business hours * One Patient will have only one Unique ID |
| **ASSUMPTIONS** | * Inputs are given by receptionist as per information received regarding Patient * Patient Information gets displayed when valid input is given |
| **CONSTRAINTS** | * Patient ID is alphanumeric – 3 alphabets and 3 digits * Patient ID is 6 characters |
| **KEY SCENARIOS** | * Relevant Patient Details should be displayed as per the Pat ID. |
| **DEPENDENCIES** | 1. Use Case Login |
| **SPECIAL REQUIREMENTS** | NA |
| **WIREFRAME REFERENCE** | NA |

| **USE CASE ID: UCD\_RCP\_004** | **USE CASE NAME: PATIENT REGISTRATION** |
| --- | --- |
| **USE CASE DESCRIPTION** | This Use case enables functions of Receptionist in Patient Registration |
| **ACTOR** | * Primary Actor: Receptionist * Secondary Actor: NA |
| **PRE- CONDITIONS** | * Active internet connection * System should be up & running * Valid Login Credentials * Only New Patients are allowed |
| **BASIC FLOW** | * 1. Login to the system   2. Select ‘Register a Patient’ functionality   3. Enter following in respective fields      1. Patient Name      2. Phone Number      3. Gender      4. Age      5. email ID      6. Address   4. Select ‘Save Patient Details’ functionality   5. All the Mandatory details are fill then System displays ‘Patient Registration completed. Patient ID is XXXXXX’   6. Use case ends successfully |
| **ALTERNATE FLOW** | * 1. Incomplete Information:      1. In Step 4.3, if the mandatory fields are not filled, display “Incomplete Information. Please fill in mandatory fields that are marked with asterisk (\*)’      2. The mandatory fields are also filled in with patient inputs and resume from Step 4.4      3. Usecase ends successfully |
| **EXCEPTIONAL FLOW** | * 1. If server is down and system fails to accept input, then valid error condition is displayed, and use case remains a failure. |
| **POST CONDITION** | **Successful Execution**: Receptionist successfully completes Patient Registration and Generates Patient ID  **Unsuccessful Execution**: Due to system failure, Receptionist cannot perform the action. |
| **INPUTS** | * User ID * Password * Patient Name * Phone Number * Gender * Age * Address * email ID |
| **OUTPUTS** | * Patient ID is generated |
| **BUSINESS RULES** | * Receptionist will be able to access the system only during operating business hours. * Patient ID is generated and it is unique to every patient |
| **ASSUMPTIONS** | * Patient Registration is successful only if Patient ID is generated * Patient ID differs for different departments but remains same if other doctors of same department * Inputs are given by receptionist as per the information received regarding Patient |
| **CONSTRAINTS** | * Patient ID is alphanumeric * Patient ID 6 characters |
| **KEY SCENARIOS** | NA |
| **DEPENDENCIES** | * Login |
| **SPECIAL REQUIREMENTS** | NA |
| **WIREFRAME REFERENCE** | NA |

| **USE CASE ID: UCD\_RCP\_005** | **USE CASE NAME: CHECK DOCTOR AVAILABILITY** |
| --- | --- |
| **USE CASE DESCRIPTION** | This Use case enables functions of Receptionist in checking Doctor availability |
| **ACTOR** | * Primary Actor: Receptionist * Secondary Actor: NA |
| **PRE- CONDITIONS** | * Active internet connection * System should be up & running * Valid Login Credentials * Doctor Appointment Slots are available with the Receptionist |
| **BASIC FLOW** | * 1. Login to the system   2. Select Doctor Name from dropdown menu   3. Enter Date   4. Enter Time   5. Select ‘Open Doctor Appointment Slots” functionality   6. If Slot is available, system displays “Available Slot”   7. Use case ends successfully |
| **ALTERNATE FLOW** | * 1. Invalid Date Input:      1. If in Step 5.3, if a date of more than 7 days prior to the day of checking doctor availability is entered, system displays “Doctor Time Slots are unavailable”      2. The Usecase ends successfully   2. Time Slot is not Available:      1. If in Step 5.6, if Time Slot is not available, system displays “Slot not unavailable”      2. Resume from Step 5.3      3. The Usecase ends successfully |
| **EXCEPTIONAL FLOW** | * 1. If server is down and system fails to accept input, then valid error condition is displayed and use case remains a failure. |
| **POST CONDITION** | **Successful Execution**: Receptionist successfully checks Doctor’s Availability  **Unsuccessful Execution**: Due to system failure, Receptionist cannot perform the action. |
| **INPUTS** | * User ID * Password * Doctor Name * Date * Time |
| **OUTPUTS** | * Display Doctor Availability |
| **BUSINESS RULES** | * Receptionist will be able to access the system only during operating business hours. * Doctor Appointment Slots should be updated regularly * Doctor Time slots for next 1 week should be available at any point of time. * Receptionist is given access to Check Availability |
| **ASSUMPTIONS** | * Receptionist will be able to check Doctor Appointment slots for 7 days (slots of that day and 6 working days after that day) |
| **CONSTRAINTS** | * Only 10 slots/ Appointments can be displayed per page |
| **KEY SCENARIOS** | * Timesheet of corresponding Doctor should be displayed |
| **DEPENDENCIES** | * Login * Updating Doctor Time Sheet |
| **SPECIAL REQUIREMENTS** | NA |
| **WIREFRAME REFERENCE** | NA |

| **USE CASE ID: UCD\_RCP\_006** | **USE CASE NAME: BOOKING APPOINTMENT** |
| --- | --- |
| **USE CASE DESCRIPTION** | This Use case enables functions of Receptionist in Booking an appointment with doctor |
| **ACTOR** | * Primary Actor: Receptionist * Secondary Actor: Printer |
| **PRE- CONDITIONS** | * Active internet connection * System should be up & running * Valid Login Credentials * Doctor Appointment slots are available with the Receptionist * Doctor Slot should be available |
| **BASIC FLOW** | * 1. Login to the system   2. Click on “Book Appointment” Functionality   3. Enter Patient ID   4. Confirm Booking   5. Generates Booking ID   6. Booking ID is printed on Consultation Form   7. Use case ends successfully |
| **ALTERNATE FLOW** | NA |
| **EXCEPTIONAL FLOW** | * 1. If server is down and system fails to accept input, then valid error condition is displayed, and use case remains a failure. |
| **POST CONDITION** | **Successful Execution**: Receptionist successfully Books an appointment and generates Booking ID  **Unsuccessful Execution**: Receptionist cannot perform the action, due to system failure |
| **INPUTS** | * User ID * Password * Patient ID |
| **OUTPUTS** | * Booking ID |
| **BUSINESS RULES** | * Receptionist will be able to access the system only during operating business hours. * Doctor Appointments list is updated automatically * Receptionist is given access to book appointments |
| **ASSUMPTIONS** | * NA |
| **CONSTRAINTS** | * Receptionist will be able to book appointments for 7 days (slots of that day and 6 working days after that day) * Booing ID should be 8 Digits and combination of Alpha-Numeric |
| **KEY SCENARIOS** | * Doctor Appointment Slots should be updated whenever appointment is booked |
| **DEPENDENCIES** | * Login * Updating Doctor Time Sheet * Check Doctor Availability |
| **SPECIAL REQUIREMENTS** | NA |
| **WIREFRAME REFERENCE** | NA |

| **USE CASE ID: UCD\_RCP\_007** | **USE CASE NAME: COLLECT CONSULTATION FEES** |
| --- | --- |
| **USE CASE DESCRIPTION** | This Use case enables functions of Receptionist in Collecting Consultation Fees |
| **ACTOR** | * Primary Actor: Receptionist * Secondary Actor: FCH, Swipe Machine |
| **PRE- CONDITIONS** | * Active internet connection * System should be up & running * Valid Login Credentials * Patient ID and Booking ID must be available |
| **BASIC FLOW** | * 1. Receptionist login to the system   2. Select “Collect Consultation Fees” Functionality   3. Enter following details:      1. Doctor Name      2. Patient ID      3. Booking ID      4. Privilege Card Number (if any)   4. View Consultation fees:      1. Consultation Fee for Non-Privilege Card Holders      2. If Privilege Card number is added, Discounted Consultation Fees for Privilege Card Holders   5. Select “Payment Mode”      1. Payment through Cash:         1. Select “Payment through Cash”         2. Enter denomination of Collected Cash in multiples of Rs1, Rs.10,Rs.50,Rs.100, Rs.200, Rs.500, Rs.2000;         3. View Change to be Returned         4. Select “OK. Change returned”         5. Continue to Step 7.6      2. Payment through Card:         1. Select “Payment through Card”         2. Swipe the card in the Swipe Machine         3. Card Validation by the Swipe Machine         4. Enter PIN in the Swipe Machine         5. PIN validation by the Swipe Machine         6. Enter Amount         7. Swipe Machine displays “Processing Payment”         8. Swipe Machine displays “Payment Successful”         9. Swipe Machine Prints Acknowledgement         10. Continue to Step 7.6      3. Payment through Wallet:         1. Select “Payment through Wallet”         2. Display QR Code         3. Patient scans QR Code         4. Continue to Step 7.6   6. Receptionist receives Payment notification through SMS   7. Generate Transaction ID   8. Use case ends successfully |
| **ALTERNATE FLOW** | * 1. In Payment through Cash: Step 7.5.1.2:      1. If there is no change to be returned to Patient, view “Return No Change”      2. Select “OK”      3. Resume Use case from Step 7.6   2. In Payment through Card: Step 7.5.2.3:      1. If Card is not valid, Swipe Machine displays “Invalid Card. Try with another card”      2. Resume Use case from Step 7.5.2.2   3. In Payment through Card: Step 7.5.2.5:      1. If PIN is not valid, Swipe Machine displays “Invalid PIN. Re-Enter PIN”      2. Resume Use case from Step 7.5.2.4 |
| **EXCEPTIONAL FLOW** | * 1. In Payment through Card: Step 7.5.2.3:      1. If Card is not valid, and swiped more than 2 times, Swipe Machine displays “Card Blocked for 24 hours. Try with another card”      2. Resume Use case from Step 7.5.2.2   2. If server is down and system fails to accept input, then valid error condition displayed and use case remains a failure. |
| **POST CONDITION** | **Successful Execution**: Receptionist successfully collects consultation fees and generates Transaction ID  **Unsuccessful Execution**: Receptionist cannot perform the action, due to system failure |
| **INPUTS** | * User ID * Password * Privilege Card Number (if any) * Denomination for Cash Collection * Card * PIN |
| **OUTPUTS** | * Consultation fees for Privilege Card Holders * Consultation fees for Non-Privilege Card Holders * Details of Change to be returned, when payment through cash * Acknowledgement when payment through Card * SMS for All the Payments * Transaction ID Generation |
| **BUSINESS RULES** | * Receptionist will be able to access the system only during operating business hours. * Denomination to be entered in multiples of Rs1, Rs.10,Rs.50,Rs.100, Rs.200, Rs.500, Rs.2000 * Consultation fees of each doctor is different, based on their experience |
| **ASSUMPTIONS** | * Discounts should be fed in the system for Privilege Card Holder |
| **CONSTRAINTS** | * Receptionist cannot apply discounts on consultation fees, apart from Privilege Card Discounts |
| **KEY SCENARIOS** | * Payment should be mapped to corresponding Patient ID and Booking ID |
| **DEPENDENCIES** | * Login * Check Doctor Availability * Booking Appointment |
| **SPECIAL REQUIREMENTS** | * Privilege Card Discounts |
| **WIREFRAME REFERENCE** | NA |

| **USE CASE ID: UCD\_RCP\_008** | **USE CASE NAME: CANCELLATION OF APPOINTMENT** |
| --- | --- |
| **USE CASE DESCRIPTION** | This Use case enables functions of Receptionist in Cancellation of a Booked appointment with doctor |
| **ACTOR** | * Primary Actor: Receptionist * Secondary Actor: NA |
| **PRE- CONDITIONS** | * Active internet connection * System should be up & running * Valid Login Credentials |
| **BASIC FLOW** | * 1. Receptionist login to the system   2. Select based on Patient request:      1. Reschedule Appointment: Go to “Reschedule Appointment” Usecase      2. Cancel Appointment: Select “Cancel Appointment Request” Functionality   3. Fills up “Reason for Cancellation” (not mandatory)   4. Enter following      1. Booking ID (or)      2. Patient ID and Doctor Name (or)      3. Patient Name, Patient Phone Number and Doctor Name   5. View “Booked Appointment Details”   6. Select “Cancel Appointment and Generate Cancellation ID” Functionality   7. Appointment is successfully cancelled   8. Cancellation ID is generated   9. Booking slot is will be added to the Available Time Slots   10. Enters Feedback Form, not mandatory   11. Submit Feedback to Admin   12. Doctor Appointment Slots are available with the Receptionist   13. Use case ends successfully |
| **ALTERNATE FLOW** | * 1. In Step 8.4, Booking ID or other Input Details are Invalid:      1. System displays “Invalid Booking ID. Please try again”      2. Resume from Step 8.4      3. The Usecase ends successfully   2. In Step 8.10, Patient does not give feedback      1. Use case ends without capturing feedback |
| **EXCEPTIONAL FLOW** | * 1. If server is down and system fails to accept input, then valid error condition is displayed and use case remains a failure. |
| **POST CONDITION** | **Successful Execution**: Receptionist successfully Cancels an appointment and generates Cancellation ID  **Unsuccessful Execution**: Receptionist cannot perform the action, due to system failure |
| **INPUTS** | * User ID * Password   + - Booking ID     - Patient ID     - Doctor Name     - Patient Name     - Patient Phone Number |
| **OUTPUTS** | * Cancellation ID * Reason for Cancellation * Feedback from Patient |
| **BUSINESS RULES** | * Receptionist will be able to access the system only during operating business hours. * Doctor Appointment Slots are updated automatically * Receptionist is given access to cancel appointments |
| **ASSUMPTIONS** | * NA |
| **CONSTRAINTS** | * Appointment Cancellation is successful only if Cancellation ID is generated * Cancellation ID is alphanumeric – 6 characters |
| **KEY SCENARIOS** | * Doctor Appointment Slots should be updated whenever appointment is booked |
| **DEPENDENCIES** | Login |
| **SPECIAL REQUIREMENTS** | NA |
| **WIREFRAME REFERENCE** | NA |

| **USE CASE ID: UCD\_RCP\_009** | **USE CASE NAME: RESCHEDULE APPOINTMENT** |
| --- | --- |
| **USE CASE DESCRIPTION** | This Use case enables functions of Receptionist in Rescheduling an appointment |
| **ACTOR** | * Primary Actor: Receptionist * Secondary Actor: NA |
| **PRE- CONDITIONS** | * Active internet connection * System should be up & running * Valid Login Credentials * Doctor Appointment Slots are available with the Receptionist |
| **BASIC FLOW** | * 1. Receptionist login to the system   2. Select “Reschedule Appointment Request” Functionality   3. Enter following details:      1. Booking ID or      2. Patient ID or      3. Patient Name and Patient Phone Number   4. Fill up “Reason for Reschedule” (not mandatory)   5. Select “Reschedule Appointment” Functionality   6. Redirect to “Booking Appointment” Usecase   7. Rescheduled Booking ID is generated   8. Use case ends successfully |
| **ALTERNATE FLOW** | * 1. In Step 9.3.1, if Booking ID is not available:      1. “Booking ID is not available. Please search with alternate Booking ID and resume from Step 9.3      2. Use case ends successfully |
| **EXCEPTIONAL FLOW** | * 1. If server is down and system fails to accept input, then valid error condition is displayed and use case remains a failure. |
| **POST CONDITION** | **Successful Execution**: Receptionist successfully reschedules appointment thereby generating Reschuled Booking ID  **Unsuccessful Execution**: Receptionist cannot perform the action, due to system failure |
| **INPUTS** | * User ID * Password   + - Booking ID     - Patient ID     - Patient Name     - Patient Phone Number * Reason for Reschedule |
| **OUTPUTS** | * Cancellation ID * Rescheduled Booking ID |
| **BUSINESS RULES** | * Receptionist will be able to access the system only during operating business hours. * Doctor Appointment Slots are updated automatically * Receptionist is given access to reschedule appointments |
| **ASSUMPTIONS** | * NA |
| **CONSTRAINTS** | * Receptionist will be able to book appointments for 7 days (slots of that day and 6 working days after that day) |
| **KEY SCENARIOS** | * Doctor Appointment Slots should be updated whenever appointment is booked |
| **DEPENDENCIES** | Login  Updating Doctor Time Sheet  Check Doctor Availability  Booking Appointment |
| **SPECIAL REQUIREMENTS** | NA |
| **WIREFRAME REFERENCE** | NA |

| **USE CASE ID: UCD\_RCP\_010** | **USE CASE NAME: ISSUE REFUND** |
| --- | --- |
| **USE CASE DESCRIPTION** | This Use case enables functions of Receptionist in Issuing Refund |
| **ACTOR** | * Primary Actor: Receptionist * Secondary Actor: NA |
| **PRE- CONDITIONS** | * Active internet connection * System should be up & running * Valid Login Credentials * Cancellation ID must be available |
| **BASIC FLOW** | * 1. Receptionist login to the system   2. Select “Issue Refund” Functionality   3. Enter following details:      1. Cancellation ID (or)      2. Booking ID (or)      3. Patient ID   4. If Cancellation ID is valid, View recent Booking and Cancellation transactions   5. Select “Issue Refund Request” Functionality   6. Intimate Refund Policy   7. If patient agrees for refund terms, Select “Initiate Refund Process”   8. If Patient’s “Original Payment Mode” is      1. Through Cash:         1. Return Rs.XXX         2. Generate Transaction ID         3. Use case ends successfully      2. Through Card/Wallet:         1. View “Credit Rs.XXX into Account XXXX towards refund for Cancellation ID.CCC through Transaction ID.TTT”         2. Generate Transaction ID         3. Use case ends successfully |
| **ALTERNATE FLOW** | * 1. In Step 10.4, if Cancellation ID is Invalid:      1. Display “If Cancellation ID not found. Please try with another ID” and resume from Step 10.3      2. Use case ends successfully   2. In Step 10.6, if Not Eligible Refund Amount”      1. Display “Patient is not eligible for Refund “      2. Use case ends successfully |
| **EXCEPTIONAL FLOW** | * 1. If server is down and system fails to accept input, then valid error condition is displayed and use case remains a failure. |
| **POST CONDITION** | **Successful Execution**: Receptionist successfully refunds amount towards appointment cancellation and generates Receipt  **Unsuccessful Execution**: Receptionist cannot perform the action, due to system failure |
| **INPUTS** | * User ID * Password * Transaction ID |
| **OUTPUTS** | * Refund Amount towards appointment cancellation * Cancellation ID |
| **BUSINESS RULES** | * Receptionist will be able to access the system only during operating business hours. * Refund amount will be given to the Patient through Original Payment Mode only * According to the Refund Policy of the hospital:   + 75% of fees is refunded, if appointment is cancelled before 12 hours   + 50% of fees is refunded, if appointment is cancelled before 3 hours   + 25% of fees is refunded, if appointment is cancelled any time later |
| **ASSUMPTIONS** | * Refund is calculated by the system |
| **CONSTRAINTS** | NA |
| **KEY SCENARIOS** | * Payment should be mapped to corresponding Cancellation ID |
| **DEPENDENCIES** | * Login * Canceling Appointment |
| **SPECIAL REQUIREMENTS** | NA |
| **WIREFRAME REFERENCE** | NA |

| **USE CASE ID: UCD\_RCP\_011** | **USE CASE NAME: ISSUE PRIVILEGE CARD** |
| --- | --- |
| **USE CASE DESCRIPTION** | This Use case enables functions of Receptionist in Issuing Privilege Card |
| **ACTOR** | * Primary Actor: Receptionist * Secondary Actor: NA |
| **PRE- CONDITIONS** | * Active internet connection * Valid System should be up & running * Valid Login Credentials |
| **BASIC FLOW** | * 1. Receptionist login to the system   2. Select “Issue Privilege Card” Functionality   3. Enter Patient ID   4. If Patient ID is valid, Generate Privilege Card Number   5. Use case ends successfully |
| **ALTERNATE FLOW** | * 1. In Step 11.4, if Patient ID is invalid:      1. System displays “Invalid Patient ID. Please try again” R      2. esume from Step 11.3      3. The Usecase ends successfully |
| **EXCEPTIONAL FLOW** | * 1. If server is down and system fails to accept input, then valid error condition is displayed and use case remains a failure. |
| **POST CONDITION** | **Successful Execution**: Receptionist successfully Issues Privilege Card to the Patient  **Unsuccessful Execution**: Receptionist cannot perform the action, due to system failure |
| **INPUTS** | * User ID * Password   + - Patient ID |
| **OUTPUTS** | * Privilege Card Number |
| **BUSINESS RULES** | * Receptionist will be able to access the system only during operating business hours * Patients is eligible for Privilege Card only if more than 20 visits in 6 months |
| **ASSUMPTIONS** | * Privilege Card gets activated immediately * Card Validity should be defined * Card Number can be traced through Patient Name and Phone Number |
| **CONSTRAINTS** | * Privilege Card Number is alphanumeric – 6 characters |
| **KEY SCENARIOS** | * Privilege Card is linked to Patient Name and Patient Phone Number |
| **DEPENDENCIES** | * Login |
| **SPECIAL REQUIREMENTS** | NA |
| **WIREFRAME REFERENCE** | NA |

| **USE CASE ID: UCD\_RCP\_012** | **USE CASE NAME: MANAGE ISSUES** |
| --- | --- |
| **USE CASE DESCRIPTION** | This Use case enables functions of Receptionist in Managing Issues |
| **ACTOR** | * Primary Actor: Receptionist * Secondary Actor: NA |
| **PRE- CONDITIONS** | * Active internet connection * System should be up & running * Valid Login Credentials |
| **BASIC FLOW** | * 1. Receptionist login to the system   2. Selects “Manage Issues” Functionality      1. Raise Issue: Selects “Raise Issue” from dropdown menu         1. Fill Issue Type         2. Fill Description         3. select Priority Type         4. Select “Submit to Admin” Functionality         5. Generate Issue ID         6. Use case ends successfully      2. Track Issue: Selects “Track Issue” from dropdown menu         1. System displays list of issues         2. Select issue ID or Provide Start Date and end Date         3. View Issue Status         4. If Status is updated, Provide Feedback         5. Use case ends successfully |
| **ALTERNATE FLOW** | NA |
| **EXCEPTIONAL FLOW** | * 1. If server is down and system fails to accept input, then valid error condition is displayed and use case remains a failure. |
| **POST CONDITION** | **Successful Execution**: Receptionist successfully Raises an Issue or Tracks an Issue  **Unsuccessful Execution**: Receptionist cannot perform the action, due to system failure |
| **INPUTS** | * + - * User ID       * Password       * Issue Type       * Description       * Priority Type       * Start Date       * End Date       * Feedback |
| **OUTPUTS** | * + - * Issue ID       * List of Issues |
| **BUSINESS RULES** | * Receptionist will be able to access the system only during operating business hours |
| **ASSUMPTIONS** | * Date and Time of Issue raised are automatically saved |
| **CONSTRAINTS** | * Issue ID is alphanumeric – 6 characters * “To Date” should be greater than “From Date” |
| **KEY SCENARIOS** | NA |
| **DEPENDENCIES** | NA |
| **SPECIAL REQUIREMENTS** | NA |
| **WIREFRAME REFERENCE** | NA |

| **USE CASE ID: UCD\_RCP\_013** | **USE CASE NAME: HANDLING CHAT** |
| --- | --- |
| **USE CASE DESCRIPTION** | This Use case enables functions of Receptionist in Handling Chat |
| **ACTOR** | * Primary Actor: Receptionist * Secondary Actor: Chat Bot |
| **PRE- CONDITIONS** | * Active internet connection * System should be up & running * Chat Bot sends Notification to Receptionist through email * Valid Login Credentials |
| **BASIC FLOW** | * 1. Receptionist login to the system   2. Open notification from Chat Bot   3. Select Query   4. Reply to Query   5. Click Submit   6. Chat Bot submits reply   7. Usecase ends successfully |
| **ALTERNATE FLOW** | NA |
| **EXCEPTIONAL FLOW** | * 1. If server is down and system fails to accept input, then valid error condition is displayed and use case remains a failure. |
| **POST CONDITION** | **Successful Execution**: Receptionist successfully handles chat  **Unsuccessful Execution**: Receptionist cannot perform the action, due to system failure |
| **INPUTS** | * + - * User ID       * Password       * Entry of Reply |
| **OUTPUTS** | * + - * Submit Reply |
| **BUSINESS RULES** | * Receptionist will be able to access the system only during operating business hours. * After working hours, queries should be stored * If reply is not submitted within 10 minutes from the opening of message, send notification again |
| **ASSUMPTIONS** | * Chat bot sends notification to receptionist through message on receiving query |
| **CONSTRAINTS** | NA |
| **KEY SCENARIOS** | NA |
| **DEPENDENCIES** | * Login |
| **SPECIAL REQUIREMENTS** | NA |
| **WIREFRAME REFERENCE** | NA |

| **USE CASE ID: UCD\_RCP\_014** | **USE CASE NAME: MANAGE PLANNED LEAVES** |
| --- | --- |
| **USE CASE DESCRIPTION** | This Use case enables functions of Receptionist for Leave Request |
| **ACTOR** | * Primary Actor: Receptionist * Secondary Actor: NA |
| **PRE- CONDITIONS** | * Active internet connection * System should be up & running * Valid Login Credentials |
| **BASIC FLOW** | * 1. Receptionist login to the system   2. Clicks on “Raise Leave Request” functionality      1. Select “Start Date”      2. Select “End Date”      3. Select “Type of leave”      4. Select “Submit to Admin”      5. Usecase ends successfully   3. Click on “Check Leave Approval Status” functionality      1. Status – “Leave approved” or “Leave Rejected” is displayed      2. Views the status      3. Usecase ends successfully |
| **ALTERNATE FLOW** | NA |
| **EXCEPTIONAL FLOW** | * 1. If server is down and system fails to accept input, then valid error condition is displayed and use case remains a failure. |
| **POST CONDITION** | **Successful Execution**: Receptionist successfully applies for leave and gets a response from Admin  **Unsuccessful Execution**: Receptionist cannot perform the action, due to system failure |
| **INPUTS** | * + - * User ID       * Password       * Start Date       * End Date       * Type of Leave |
| **OUTPUTS** | * + - * Leave request ID       * Leave Request submitted       * Leave Request Tracked |
| **BUSINESS RULES** | * Receptionist will be able to access the system only during operating business hours |
| **ASSUMPTIONS** | * + - * Leave Application includes Employee ID |
| **CONSTRAINTS** | NA |
| **KEY SCENARIOS** | NA |
| **DEPENDENCIES** | * Login |
| **SPECIAL REQUIREMENTS** | NA |
| **WIREFRAME REFERENCE** | NA |

| **USE CASE ID: UCD\_RCP\_015** | **USE CASE NAME: GENERATE RECEIPTS** |
| --- | --- |
| **USE CASE DESCRIPTION** | This Use case enables functions of Receptionist in Generating Receipts |
| **ACTOR** | * Primary Actor: Receptionist * Secondary Actor: Printer |
| **PRE- CONDITIONS** | * Active internet connection * System should be up & running * Valid Login Credentials * Transaction ID must be available |
| **BASIC FLOW** | * 1. Receptionist login to the system   2. Select: “Generate Receipt” Functionality   3. Enter Transaction ID   4. Select option from dropdown menu: Consultation fees or Refund      1. Consultation Fees:         1. Select option from dropdown menu: Cash or Card or Wallet         2. Continue to Step 15.3      2. If Refund         1. Enter Transaction ID         2. Continue to Step 15.3   5. Generate Receipt ID   6. Print Receipt with Receipt ID, Patient ID and Transaction ID   7. Use case ends successfully |
| **ALTERNATE FLOW** | NA |
| **EXCEPTIONAL FLOW** | * 1. If server is down and system fails to accept input, then valid error condition should be displayed and use case remains a failure |
| **POST CONDITION** | **Successful Execution**: Receptionist successfully Generates Receipt ID and Print Receipt  **Unsuccessful Execution**: Receptionist cannot perform the action, due to system failure |
| **INPUTS** | * User ID * Password * Transaction ID |
| **OUTPUTS** | * Receipt ID * Print Receipt |
| **BUSINESS RULES** | * Receptionist will be able to access the system only during operating business hours. * Denomination to be entered in multiples of Rs1, Rs.10,Rs.50,Rs.100, Rs.200, Rs.500, Rs.2000 * Consultation fees of each doctor is different * System automatically applies Discount on Consultation Fees for Privilege Card Holder |
| **ASSUMPTIONS** | * Discounts should be fed in the system for Privilege Card Holder |
| **CONSTRAINTS** | * Receptionist cannot apply discounts on consultation fees |
| **KEY SCENARIOS** | * Payment should be mapped to corresponding Patient ID and Booking ID |
| **DEPENDENCIES** | * Login * Check Doctor Availability * Booking Appointment |
| **SPECIAL REQUIREMENTS** | * Privilege Card Number |
| **WIREFRAME REFERENCE** | NA |

| **USE CASE ID: UCD\_RCP\_016** | **USE CASE NAME: UPDATE PATIENT DETAILS** |
| --- | --- |
| **USE CASE DESCRIPTION** | This Use case enables functions of Receptionist in updating Patient Information |
| **ACTOR** | * Primary Actor: Receptionist * Secondary Actor: NA |
| **PRE- CONDITIONS** | * Active internet connection * System should be up & running * Valid Login Credentials * Only Registered Patients are allowed |
| **BASIC FLOW** | * 1. Login to the system   2. Select ‘Update Patient Details’ functionality   3. To track patient details, Enter      1. Patient ID or      2. Patient Name and Phone Number   4. View Patient Details   5. Edit information like Patient Name, Phone Number, Gender, Age, email ID or Address, as requested by Patient   6. Select ‘Save Patient Details’ functionality   7. Use case ends successfully |
| **ALTERNATE FLOW** | * 1. Patient ID is not tracked’      1. In Step 16.4, if patient ID is not tracked, display “Patient ID not found. Please try again”      2. Resume from Step 16.3      3. Usecase ends successfully |
| **EXCEPTIONAL FLOW** | * 1. If server is down and system fails to accept input, then valid error condition is displayed, and use case remains a failure. |
| **POST CONDITION** | **Successful Execution**: Receptionist successfully completes Patient Registration and Generates Patient ID  **Unsuccessful Execution**: Due to system failure, Receptionist cannot perform the action. |
| **INPUTS** | * User ID * Password * Patient ID * Patient Name * Phone Number * Information to be edited/ incorporated |
| **OUTPUTS** | * Patient details updated |
| **BUSINESS RULES** | * Receptionist will be able to access the system only during operating business hours. * Patient ID is unique to every patient |
| **ASSUMPTIONS** | * Inputs given by Patient are submitted by receptionist |
| **CONSTRAINTS** | * Patient ID is alphanumeric * Patient ID 6 characters |
| **KEY SCENARIOS** | NA |
| **DEPENDENCIES** | * Login |
| **SPECIAL REQUIREMENTS** | NA |
| **WIREFRAME REFERENCE** | NA |