|  |  |  |  |
| --- | --- | --- | --- |
| Use case name | Register | Unique ID | CAREBOT-User-001 |
| Area | Care Bot | | |
| Actor(s) | User (Patient, or Doctor) | | |
| Level | Blue | | |
| Description | User creates account | | |
| Triggering Event | User click “Register” button in the application | | |
| Preconditions | * The user needs to download application then open it * The user needs to have internet access | | |
| Postconditions | * User has successfully create account | | |
| Assumptions | * User have Care bot application * A valid data | | |
| Steps Performed | | Information for Steps | |
| 1. Open application 2. Choose if he is a patient or doctor 3. User enters his data 4. Click on “Create Account” button 5. Validation of entered data by application | | Step 3: Name, UserName, Password, E-mail, SSN, EMSN (Egyptian Medical Syndicate Number if he was a doctor) | |
| Extensions (Alternative Flows) | * If the downloading interrupted for any reason, use should try again and download it * If user entered a non-valid data, a warning message should appear to him | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use case name | Login | Unique ID | CAREBOT-User-002 |
| Area | Care Bot | | |
| Actor(s) | User(Patient, or Doctor) | | |
| Level | Blue | | |
| Description | User login to his account | | |
| Triggering Event | User click “Login” button in the application | | |
| Preconditions | * The user needs to download application then open it * The user needs to have internet access * The user needs to have account | | |
| Postconditions | * User has successfully logged in to his account | | |
| Assumptions | * User have Care bot application * A valid data | | |
| Steps Performed | | Information for Steps | |
| 1. Open application 2. User enters his data 3. Click on “Login” button 4. Validation of entered data by application | | Step 2: E-mail, Password | |
| Extensions (Alternative Flows) | * If user entered a non-valid data, a warning message should appear to him | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use case name | Adding old medical history | Unique ID | CAREBOT-Patient-001 |
| Area | Care Bot | | |
| Actor(s) | Patient | | |
| Level | Blue | | |
| Description | Patient adds old medical history | | |
| Triggering Event | Patient click “Add old medical history” button in the application | | |
| Preconditions | * The Patient needs to login in to his account | | |
| Postconditions | * Patient has successfully added old medical history to his account | | |
| Assumptions | * Patient have Care bot application * A valid data | | |
| Steps Performed | | Information for Steps | |
| 1. Open application 2. Patient log in 3. Patient click “Add old medical history” button 4. Patient Enters his data 5. Patient clicks on “Save” button 6. Validation of entered data by application | | Step 2: E-mail, Password  Step 4: Symptoms, Disease, Treatment | |
| Extensions (Alternative Flows) | * If user entered a non-valid data, a warning message should appear to him | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use case name | Start audio recording | Unique ID | CAREBOT-Doctor-001 |
| Area | Care Bot | | |
| Actor(s) | Doctor | | |
| Level | Blue | | |
| Description | Doctor start new diagnose | | |
| Triggering Event | Doctor click “Start audio recording” button in the application | | |
| Preconditions | * The Doctor needs to login in to his account | | |
| Postconditions | * Audio has successfully recorded, submitted to server and returned result | | |
| Assumptions | * Doctor have Care bot application * Patient should speak in a clear voice * Patient should talk about symptoms in a clear and direct way | | |
| Steps Performed | | Information for Steps | |
| 1. Open application 2. Doctor log in 3. Doctor click “Start audio recording” button 4. Patient start talking and describes how he feels ‘symptoms’ 5. Doctor clicks on “Finish recording” button 6. Validation of entered data by application and send data to server 7. Server respond with result 8. Doctor adds note if he want 9. Doctor click “Save” button | | Step 2: E-mail, Password  Step 4: Audio  Step 8: Note  Step 9: Symptoms, Disease, Treatment, Note | |
| Extensions (Alternative Flows) | * If doctor entered a non-valid data in login, a warning message should appear to him | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use case name | Scan patient QR code | Unique ID | CAREBOT-Doctor-002 |
| Area | Care Bot Application, Server | | |
| Actor(s) | Doctor | | |
| Level | Blue | | |
| Description | Get patient’s old medical history | | |
| Triggering Event | Doctor click “Scan QR” button in the application | | |
| Preconditions | * Doctor needs to login in to his account * Patient needs to login in to his account * Patient must have QR code | | |
| Postconditions | * Doctor has successfully get patient’s old medical history on his device | | |
| Assumptions | * Doctor have Care bot application * Patient should have QR code * Doctor phone have a camera | | |
| Steps Performed | | Information for Steps | |
| 1. Open application 2. Doctor log in 3. Doctor click “Scan QR” button 4. Doctor scan QR code by his camera 5. Validation of entered data by application and send data to server 6. Server respond with result | | Step 2: E-mail, Password  Step 4: QR code  Step 6: Old\_Medical\_History | |
| Extensions (Alternative Flows) | * If doctor entered a non-valid data in login, a warning message should appear to him * If patient have non-valid QR code, then patient must generate new QR code in his account | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use case name | Convert audio to text and extract symptoms | Unique ID | CAREBOT-NLP-001 |
| Area | Server | | |
| Actor(s) | NLP | | |
| Level | Blue | | |
| Description | Convert audio received to text then extract symptoms | | |
| Triggering Event | API send audio | | |
| Preconditions | * There is no preconditions | | |
| Postconditions | * NLP module has successfully converted audio to text and extracted symptoms | | |
| Assumptions | * The audio have to be clear | | |
| Steps Performed | | Information for Steps | |
| 1. Convert audio to text 2. Analyze text 3. Extract symptoms from text 4. Send symptoms | | Step 1: Audio  Step 2: Text  Step 4:Symptoms | |
| Extensions (Alternative Flows) | * If audio wasn’t clear, a warning message should appear to doctor to re-record audio * If language wasn’t supported , a warning message should appear to doctor to re-record audio in supported language * If patient described an untrained disease for program, a warning message should appear to doctor that program couldn’t extract symptoms * If module couldn’t extract symptoms from text, a warning message should appear to doctor that program couldn’t extract symptoms | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use case name | Classification of the disease | Unique ID | CAREBOT-DiseaseClassifier-001 |
| Area | Server | | |
| Actor(s) | Disease Classifier | | |
| Level | Blue | | |
| Description | Getting symptoms and predict disease | | |
| Triggering Event | NLP module sent symptoms to disease classifier | | |
| Preconditions | * There is no preconditions | | |
| Postconditions | * Disease classifier module has successfully predicted disease | | |
| Assumptions | * NLP module must send symptoms | | |
| Steps Performed | | Information for Steps | |
| 1. Disease classifier analyze symptoms and predict disease 2. Return result | | Step 1: Symptoms  Step 2: Disease | |
| Extensions (Alternative Flows) | * If module couldn’t analyze symptoms to predicate disease, a warning message should appear to doctor that program couldn’t predict disease | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use case name | Suggest medicine | Unique ID | CAREBOT- TreatmentPredictor-001 |
| Area | Server | | |
| Actor(s) | Treatment predictor | | |
| Level | Blue | | |
| Description | Suggest medicine | | |
| Triggering Event | Disease classifier sent predicted disease to treatment predictor | | |
| Preconditions | * Disease classifier sent predicted disease | | |
| Postconditions | * Treatment predictor provided appropriate treatment | | |
| Assumptions | * Disease classifier must sent predicted disease | | |
| Steps Performed | | Information for Steps | |
| 1. Treatment predictor analyze disease and search for the best medicine for it 2. Return result | | Step 1: Disease  Step 2: Medicines | |
| Extensions (Alternative Flows) | * If module couldn’t analyze disease to predicate medicine, a warning message should appear to doctor that program couldn’t predict appropriate medicine | | |