|  |  |  |  |
| --- | --- | --- | --- |
| 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 | Classification of the disease | Unique ID | CAREBOT-Doctor-001 |
| Area | server | | |
| Actor(s) | Disease classification | | |
| Level | Blue | | |
| Description | Getting symptoms and prediction the name of the disease | | |
| Triggering Event | Disease classification get symptoms and analysis it then prediction disease | | |
| Preconditions | * The NLP must extraction symptoms | | |
| Postconditions | * Disease classification prediction disease | | |
| Assumptions | * The NLP must send symptoms | | |
| Steps Performed | | Information for Steps | |
| 1. Disease classification get symptoms 2. Disease classification analyzes symptoms 3. Disease classification prediction disease | | Step 2: Symptoms | |
| Extensions (**Alternative Flows**) | If Disease classification can’t get symptoms   * , a warning message should appear to him | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use case name | Suggect medicine | Unique ID | CAREBOT-Doctor-001 |
| Area | server | | |
| Actor(s) | Treatment predictor | | |
| Level | Blue | | |
| Description | Getting disease and Suggect medicine | | |
| Triggering Event | trearments get disease and search or prediction drugs | | |
| Preconditions | * Disease classification must prediction disease | | |
| Postconditions | * trearments get or search the best disease | | |
| Assumptions | * the Disease classification must send disease | | |
| Steps Performed | | Information for Steps | |
| 1. trearments classification get disease 2. trearments search the best drugs for this disease 3. trearments Suggect medicine | | Step 2: disease | |
| Extensions (**Alternative Flows**) | If trearments can’t get disease   * , a warning message should appear to him | | |