# Use Cases

Add additional rows or copies on the Alternative Flows table as needed.

1. User wants to register

2. User wants to login

3. User wants to view the status of the system

4. User wants to add a new device to the system

5. User wants to record motions

6. User wants to upload/discard the data

7. User wants to individualize the model

8. User wants to reset the model

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case | User wants to register | | |
| Version | 1.0 | Created (date): | 3.31 |
| Author | Jiazheng Pan, Haotian Tang, Weibin Cheng | | |
| Source | / | | |
| Goals | The user would like to register an account to access the software. | | |
| Summary | The user input all the necessary information as well as captcha and then get a new account. | | |
| Actors | End user | | |
| Trigger | The user wants to get an account so he/she can use the software. | | |
| Precondition | The Internet is available and the registration page is displayed. | | |
| Frequency | Usually just once, before starting to use the software. | | |
| Postconditions | Automatically log in with the lastly registered account. | | |
| Diagram |  | | |

|  |  |  |
| --- | --- | --- |
| **Basic Flow** | *Actor* | *System* |
|  | Click the registration button |  |
|  | Input the necessary information including e-mail |  |
|  |  | Send the captcha to the e-mail |
|  | Input the captcha |  |
|  |  | Generate a new account |
|  | Receive a notification and enter the software |  |

|  |  |  |
| --- | --- | --- |
| **Alternative Flow** | *Actor* | *System* |
|  | From basic flow 2 |  |
|  | Input illegal registration information such as wrong e-mail address or too short password |  |
|  |  | Prompts the user, gives the correct format and locks the ‘next step’ button |
|  | User corrects the information entered |  |
|  |  | Go to basic flow Step 3 |

|  |  |  |
| --- | --- | --- |
| **Alternative Flow** | *Actor* | *System* |
| 1. | From basic flow step 4 |  |
| 2. |  | Send the captcha to the e-mail |
| 3. | Input wrong captcha and click ‘sign up’ button |  |
| 4. |  | Prompts the users of wrong captcha |
| 5. | User corrects the captcha |  |
| 6. |  | Go to basic flow step 5 |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case | User wants to login | | |
| Version | 1.0 | Created (date): | 3.31 |
| Author | Jiazheng Pan, Haotian Tang, Weibin Cheng | | |
| Source | / | | |
| Goals | The user would like to log in the account. | | |
| Summary | The user input the username and password, and if it’s correct, then jump to the main page. | | |
| Actors | End user | | |
| Trigger | The user wants to enter the software and use it. | | |
| Precondition | The user already has an account and the login page is displayed. | | |
| Frequency | Every time before the user enter the software. | | |
| Postconditions | Jump tp the main page of the software. | | |
| Diagram | / | | |

|  |  |  |
| --- | --- | --- |
| **Basic Flow** | *Actor* | *System* |
| 1 | Input the username and password |  |
| 2 |  | Determine whether they match |
| 3 |  | If match, jump to the main page |
| 4 |  | If not, prompt password error |
| 5 | Enter the main page |  |

|  |  |  |
| --- | --- | --- |
| **Alternative Flow** | *Actor* | *System* |
| 1 | From basic flow step 1 |  |
| 2 | Inputs wrong user name or password |  |
| 3 |  | Prompts the user of wrong or unmatched user name and password |
| 4 | Corrects the user name and/or password |  |
| 5 |  | Go to basic flow step 5 |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case | User wants to view the status of the system | | |
| Version | 1.0 | Created (date): | 3.31 |
| Author | Jiazheng Pan, Haotian Tang, Weibin Cheng | | |
| Source | / | | |
| Goals | The user wants to view the current status of the system, such as connection status, work status, etc. | | |
| Summary | The user selects “the status of the model” page and can see the related status in the page. So the user can decide what to do next. | | |
| Actors | End user | | |
| Trigger | The user wants to know what to do according to the system status. | | |
| Precondition | The user has logged in and the system is running normally. | | |
| Frequency | Uncertain, according to the user. | | |
| Postconditions | The current status of the system is displayed in the page. | | |
| Diagram |  | | |

|  |  |  |
| --- | --- | --- |
| **Basic Flow** | *Actor* | *System* |
| 1 | Select “the status of the model” page |  |
| 2 |  | Show the current system status |
| 3 | See the system status |  |

|  |  |  |
| --- | --- | --- |
| **Alternative Flow** | *Actor* | *System* |
| 1 | From basic flow step 1 |  |
| 2 |  | Failed to connect to the server |
| 3 |  | Prompt the user of network failure |
| 4 | Go to basic flow step 1 |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case | User wants to add a new device to the system | | |
| Version | 1.0 | Created (date): | 3.31 |
| Author | Jiazheng Pan, Haotian Tang, Weibin Cheng | | |
| Source | / | | |
| Goals | The user wants to add a new sensor to the system. | | |
| Summary | The user select the “add devices” option and chooses the certain device in the detection list. Then the device is connected to the system. | | |
| Actors | End user | | |
| Trigger | The user wants to use a device to record motions. | | |
| Precondition | The user has logged in. The motion device is available and can link the Internet. | | |
| Frequency | Every time a new device is used. | | |
| Postconditions | The new added device can be seen in the device list. | | |
| Diagram | / | | |

|  |  |  |
| --- | --- | --- |
| **Basic Flow** | *Actor* | *System* |
| 1 | Get the new device and activate it |  |
| 2 | Select the “add device” option |  |
| 3 |  | Scan all devices within range |
| 4 |  | Show the detected devices |
| 5 | Choose the corresponding device |  |
| 6 |  | Add the chosen device to the system |

|  |  |  |
| --- | --- | --- |
| **Alternative Flow** | *Actor* | *System* |
| 1 |  | From basic flow step 4 |
| 2 | Cannot find the expected device |  |
| 3 | Check the availability of the device |  |
| 4 | Go to basic flow step 1 |  |
| **Alternative Flow** | *Actor* | *System* |
| 1 |  | From basic flow step 6 |
| 2 |  | Fails to add the chosen device |
| 3 |  | Prompts the user of failure and its reason |
| 4 | Check the availability of the device |  |
| 5 | Go to basic flow step 1 |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case | User wants to start record motions | | |
| Version | 1.0 | Created (date): | 3.31 |
| Author | Jiazheng Pan, Haotian Tang, Weibin Cheng | | |
| Source | / | | |
| Goals | The user wants to use the device to record motions. | | |
| Summary | The user click the “start/stop” button to start or stop recording motions and does some movements next. | | |
| Actors | End user | | |
| Trigger | The user wants to collect new data. | | |
| Precondition | The user has logged in. The device is intact and is connected to the system. | | |
| Frequency | Every time the user wants to collect some data. | | |
| Postconditions | The motions are recorded and temporally stored. | | |
| Diagram | / | | |

|  |  |  |
| --- | --- | --- |
| **Basic Flow** | *Actor* | *System* |
| 1 | Click the “start” button |  |
| 2 | Do some movements |  |
| 3 |  | Start to record with the device |
| 4 | Pause/continue the recording |  |
| 5 | Click the “stop” button |  |
| 6 |  | Finish recording and store the data temporally |

|  |  |  |
| --- | --- | --- |
| **Alternative Flow** | *Actor* | *System* |
| 1 | From basic flow step 2 |  |
| 2 |  | Fails to track user’s movement due to broken connection, e.t.c. |
| 3 |  | Prompts the user of failure and its reason |
| 4 | Go to basic flow step 1 |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case | User wants to upload/discard the data | | |
| Version | 1.0 | Created (date): | 3.31 |
| Author | Jiazheng Pan, Haotian Tang, Weibin Cheng | | |
| Source | / | | |
| Goals | The user wants to upload/ discard the collected data. | | |
| Summary | The user chooses the “upload” or “discard” button to decide whether upload the data to the database or discard it. | | |
| Actors | End user | | |
| Trigger | The user wants to deal with the collect data in order for the next recording | | |
| Precondition | The user has logged in. The recording is stopped and the motions are successfully collected. | | |
| Frequency | Every time the recording is successfully completed. | | |
| Postconditions | The collected data is uploaded to the database or discarded. | | |
| Diagram | / | | |

|  |  |  |
| --- | --- | --- |
| **Basic Flow** | *Actor* | *System* |
| 1 | Click “upload” or “discard” button |  |
| 2 |  | Upload the data to the database or discard it |
| 3 |  | Show processing results |

|  |  |  |
| --- | --- | --- |
| **Alternative Flow** | *Actor* | *System* |
| 1 |  | From basic flow step 2 |
| 2 |  | Fails to upload the data to the server due to network congestion or broken connection |
| 3 |  | Prompts the user of failure and its reason |
| 4 | Go to basic flow 1 |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case | User wants to individualize the model | | |
| Version | 1.0 | Created (date): | 3.31 |
| Author | Jiazheng Pan, Haotian Tang, Weibin Cheng | | |
| Source | / | | |
| Goals | The user wants to individualize the model with the collected data to adaptively make predictions. | | |
| Summary | The user selects the “individualize” option and let the system train the model with the new data. Then use the new model to predict. | | |
| Actors | End user | | |
| Trigger | The user wants to finetune the existing model. | | |
| Precondition | The user has logged in. Some new data is collected and uploaded. | | |
| Frequency | Every time the user wants to finetune the model. | | |
| Postconditions | The existing model is trained and the finetuned model is ready for prediction. | | |
| Diagram |  | | |

|  |  |  |
| --- | --- | --- |
| **Basic Flow** | *Actor* | *System* |
| 1 | Select a certain basic model |  |
| 2 | Select “individualize” option |  |
| 3 |  | Train the chosen model with the new data |
| 4 |  | Dispaly the loss function and related index |
| 5 | Synchronize the newly trained model when finishing |  |

|  |  |  |
| --- | --- | --- |
| **Alternative Flow** | *Actor* | *System* |
| 1 | From basic flow step 5 |  |
| 2 |  | Fails to synchronize the newly trained model due to certain reason such as network disconnection |
| 3 |  | Prompts the user of failure and its reason |
| 4 | Go to basic flow step 5 |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case | User wants to reset the model | | |
| Version | 1.0 | Created (date): | 3.31 |
| Author | Jiazheng Pan, Haotian Tang, Weibin Cheng | | |
| Source | / | | |
| Goals | The user wants to reset the finetuned model to the initial state. | | |
| Summary | The user selects the “reset” option and clear the newly finetuned model, resetting it to the initial state. | | |
| Actors | End user | | |
| Trigger | The user doesn’t want to use the new model. | | |
| Precondition | The user has logged in. There is a model that has been trained with the new data. | | |
| Frequency | Uncertain, according to the user. | | |
| Postconditions | The newly trained model is restored. | | |
| Diagram | / | | |

|  |  |  |
| --- | --- | --- |
| **Basic Flow** | *Actor* | *System* |
| 1 | Select the “reset” option |  |
| 2 |  | Download the original model from server |
| 3 |  | Discard the old model |
| 4 |  | Reset the model to the initial state |
| 5 | Receive successful tips |  |

|  |  |  |
| --- | --- | --- |
| **Alternative Flow** | *Actor* | *System* |
| 1 |  | From basic flow step 2 |
| 2 |  | Downloads fails due to insufficient memory, network anomaly or other unknown reason |
| 3 |  | Prompts the user of failure and its reason |
| 4 | Go to basic flow step 1 |  |