# Console System Test Cases

## Menu Test

*Purpose*

1. Verify that the items in the menu function properly.

*Requirement Traceability*

Log in NetBeans ANT.log

*Setup*

Open NetBeans. Pull the CIT360 app remotely from GitHub. Press F6 to run the app and interact with the console.

*Test Data*

|  |  |  |
| --- | --- | --- |
| **Action** | **Input** | **Expected Output** |
| List | Type “list” at the prompt | Collections script runs |
| Hibernate | Type “del” at the prompt | Hibernate script runs |
| HTTPConnection | Type “url” at the prompt | HTTPConnection/JSON script runs |
| Threading | Type “thread” at the prompt | Threading script runs |
| Quit | Type “exit” at the prompt” | Program exits |

## Post-Build Test

*Purpose*

1. Verify that the build of the CIT360 application worked properly.

*Requirement Traceability*

Log in NetBeans ANT.log

*Setup*

Open NetBeans. Pull the CIT360 app remotely from GitHub. Open command prompt.

*Test Data*

|  |  |  |
| --- | --- | --- |
| **Action** | **Input** | **Expected Output** |
| Build | Right-click cit360 app in NetBeans, click Build | Build Successful |
| Verify JAR | Open Explorer, look for byui.cit360.jar file. | File Found (note folder location) |
| Run | Navigate to JAR file location, then run ‘java -jar byui.cit360.jar’. | Initial menu appears |

## Internet Connectivity Test

*Purpose*

1. Verify that the application can successfully access the Internet, and specifically the Rackspace Cloud API.

*Requirement Traceability*

Log in NetBeans ANT.log

*Setup*

Open NetBeans. Pull the CIT360 app remotely from GitHub. Press F6 to run the app and interact with the console. Alternately, build and run the app using the instructions in the Build Test.

*Test Data*

|  |  |  |
| --- | --- | --- |
| **Action** | **Input** | **Expected Output** |
| Run URL app | Type “url” at the prompt | “Enter User Name:” prompt appears |
| Enter Credentials | Enter provided credentials at the Enter User Name and Enter API Key prompts. | Response from server: 200 OK |

## Database Connectivity Test

*Purpose*

1. Verify that the application can successfully access the local MySQL database.

*Requirement Traceability*

Log in NetBeans ANT.log

*Setup*

Open NetBeans. Pull the CIT360 app remotely from GitHub. Press F6 to run the app and interact with the console. Alternately, build and run the app using the instructions in the Build Test.

*Test Data*

|  |  |  |
| --- | --- | --- |
| **Action** | **Input** | **Expected Output** |
| Run Hibernate app | Type “del” at the prompt | “0 records deleted.” |

## Exception Test

*Purpose*

1. Verify that the Collection application correctly handles incorrectly typed names or numbers.

*Requirement Traceability*

Log in NetBeans ANT.log

*Setup*

Open NetBeans. Pull the CIT360 app remotely from GitHub. Press F6 to run the app and interact with the console. Alternately, build and run the app using the instructions in the Build Test.

*Test Data*

|  |  |  |
| --- | --- | --- |
| **Action** | **Input** | **Expected Output** |
| Start Collections app | Enter “list” at the prompt | “Enter a Last Name” prompt appears |
| Test input | Enter a name not in the list the app provides. | Name Not Found, program continues. |

## Database Insert Test

*Purpose*

1. Verify that the application can successfully insert records into the MySQL database instance, or fails correctly if the record to add already exists.

*Requirement Traceability*

Log in NetBeans ANT.log

*Setup*

Open NetBeans. Pull the CIT360 app remotely from GitHub. Press F6 to run the app and interact with the console. Alternately, build and run the app using the instructions in the Build Test.

*Test Data*

|  |  |  |
| --- | --- | --- |
| **Action** | **Input** | **Expected Output** |
| Run Collections app | Type “list” at the prompt | “Enter a Last Name:” prompt appears |
| Enter last name | Enter a name in the provided list.. | “Enter an Employee Number:” prompt appears |
| Enter employee number | Enter an employee number from the list provided. | JSON of employee data appears.  Records Stored successfully.  Menu appears. |
| Re-run Collections app | Type “list” at the prompt | “Enter a Last Name:” prompt appears. |
| Follow steps 2 and 3 | Same inputs as steps 2 and 3 above | No Hibernate exceptions |

## Database Load Test

*Purpose*

1. Verify that the local MySQL database can support multiple concurrent connections.

*Requirement Traceability*

MySQL.log, Performance Monitor log

*Setup*

Open MySQL Studio. Run Performance Monitor locally, with counters in place to capture CPU, RAM and disk counters. Write SQL script in c:\test\script.sql to connect to the MySQL instance and perform multiple queries. Open command prompt.

*Test Data*

|  |  |  |
| --- | --- | --- |
| **Action** | **Input** | **Expected Output** |
| Test script | At command prompt, run ‘mysql -e c:\test\script.sql’ | Query results display on screen. |
| Run script x times | At command prompt, run ‘for /l %i in (1,1,x) do start “Test-%i” mysql -e c:\test\script.sql’ | X command windows appear with the sql script |
| Check performance | None | After Each test, verify max CPU, RAM and disk load |
|  |  |  |
|  |  |  |