## **Test Case Design**

**Black Box Testing:**

Control Class Chosen: RouteFactory

Class Method: getDefaultRoute()

Inputs:

1. All Valid Inputs
2. Empty set of intermediate points
3. Invalid endPt (outside Singapore)
4. Invalid startPt (outside Singapore)

Equivalence classes with invalid endPt and startPt inputs were tested.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | startPt | intermediatePts | endPt | Expected Output | Actual Output |
| 1 | “1.3406787810831653,103.80102557666015” | [“1.3355303441379147,103.81029529101562”, “1.3329561216181542,103.81716174609375”] | “1.3370748763558524,103.82643146044921” | API returns 200 OK Response | API returns 200 OK Response |
| 2 | “1.3406787810831653,103.80102557666015” | [] | “1.3370748763558524,103.82643146044921” | API returns 200 OK Response | API returns 200 OK Response |
| 3 | “1.3406787810831653,103.80102557666015” | [] | “1.4697847651217744,103.55368419713508” | API returns 400 Bad Request | API returns 400 Bad Request |
| 4 | “1.4697847651217744,103.55368419713508” | [] | “1.3406787810831653,103.80102557666015” | API returns 400 Bad Request | API returns 400 Bad Request |

Class Method: getLuckyRoute()

Inputs:

1. All Valid Inputs, with just-above routeDistance input of 1, and just-below startPt of  “1.408,103.925”
2. Invalid on-the-boundary routeDistance input of 0
3. Invalid just-below routeDistance input of -1
4. Valid on-the-boundary startPt of “1.4089433,103.9255495” (Singapore boundary)
5. Invalid just-above startPt of “1.41,103.93” (outside Singapore boundaries)

Equivalence classes with invalid startPt and routeDistance inputs were tested, and boundary value testing was applied for routeDistance and startPt inputs

|  |  |  |  |
| --- | --- | --- | --- |
| startPt | routeDistance | Expected Output | Actual Output |
| “1.408,103.925” | 1 | API returns 200 OK Response | API returns 200 OK Response |
| “1.408,103.925” | 0 | API returns 400 Bad Request | API returns 400 Bad Request |
| “1.408,103.925” | -1 | API returns 400 Bad Request | API returns 400 Bad Request |
| “1.4089433,103.9255495” | 1 | API returns 200 OK Response | API returns 200 OK Response |
| “1.41,103.93” | 1 | API returns 400 Bad Request | API returns 400 Bad Request |

Class Method: geocode()

Inputs:

1. All Valid Inputs
2. Invalid point (outside Singapore)

Equivalence classes with invalid point was tested.

|  |  |  |
| --- | --- | --- |
| point | Expected Output | Actual Output |
| “1.361902944854688,  103.70723262475029” | API returns 200 OK Response | API returns 200 OK Response |
| “1.4697847651217744,103.55368419713508” | API returns 400 Bad Request | API returns 400 Bad Request |

**White Box Testing:**

Method: submitForm() for sign up

Decision points = 8

CC = 8+1 =9

Basis Paths:

#1: 2-5, 6, 8, 10, 12, 14, 16, 18, 20, 22

#2: 2-5, 6, 7

#3: 2-5, 6, 8, 9

#4: 2-5, 6, 8, 10, 11

#5: 2-5, 6, 8, 10, 12, 13

#6: 2-5, 6, 8, 10, 12, 14, 15

#7: 2-5, 6, 8, 10, 12, 14, 16, 17

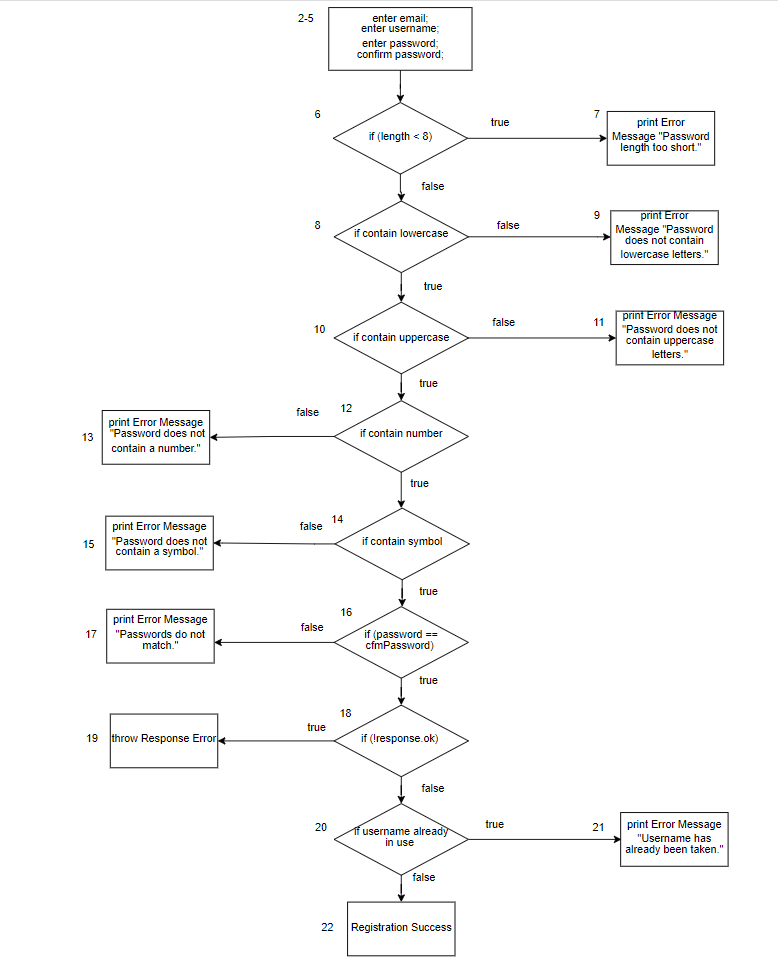
#8: 2-5, 6, 8, 10, 12, 14, 16, 18, 19

#9: 2-5, 6, 8, 10, 12, 14, 16, 18, 20, 21

Inputs:

1. Valid input
2. Invalid password length
3. Password without lowercase letters
4. Password without uppercase letters
5. Password without numbers
6. Password without symbols
7. Non-matching passwords
8. Backend no responses
9. Duplicated username

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | email address | username | password | Confirm password | Expected Output | Actual Output |
| 1 | User@gmail.com | User | 123456aA\* | 123456aA\* | Successfully registered | Successfully registered |
| 2 | User1@gmail.com | User1 | 123456 | 123456 | Password length too short. | Password length too short. |
| 3 | User1@gmail.com | User1 | 12345678A\* | 12345678A\* | Password does not contain lowercase letters. | Password does not contain lowercase letters. |
| 4 | User1@gmail.com | User1 | 12345678a\* | 12345678a\* | Password does not contain uppercase letters. | Password does not contain uppercase letters. |
| 5 | User1@gmail.com | User1 | abcdABCD\* | abcdABCD\* | Password does not contain a number. | Password does not contain a number. |
| 6 | User1@gmail.com | User1 | 123456aA | 123456aA | Password does not contain a symbol. | Password does not contain a symbol. |
| 7 | User1@gmail.com | User1 | 123456aA\* | aA123456\* | Passwords do not match. | Passwords do not match. |
| 8 | User1@gmail.com | User1 | 123456aA\* | 123456aA\* | Backend response error. | Backend response error. |
| 9 | User2@gmail.com | User | 123456aA\* | 123456aA\* | Username has already been taken. | Username has already been taken. |



**White Box Testing:**

Method: favouriteRoute()

Decision points = 2

CC = 2+1 =3

Basis Paths:

#1: 2-3, 4, 6, 7, 9, 10, 11

#2: 2-3, 4, 5

#3: 2-3, 4, 6, 7, 8

Inputs:

1. Valid User ID and Route ID
2. Invalid User ID or invalid Route ID
   1. Valid User ID and invalid Route ID
   2. Invalid User ID and valid Route ID
3. Valid User ID and Route ID which is already in “Favourites” array

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | User ID | Route ID | Expected Output | Actual Output |
| 1 | test | 3TbKc61tF5rFMoY1Wq1g | Successful: route added to favourites | Successful: route added to favourites |
| 2a | test | 123 | Unsuccessful: invalid user or routes | Unsuccessful: invalid user or routes |
| 2b | user | 8EOn7GNDSXORNQXBvDfQ | Unsuccessful: invalid user or routes | Unsuccessful: invalid user or routes |
| 3 | test | 3TbKc61tF5rFMoY1Wq1g | Unsuccessful: route already saved by user | Unsuccessful: route already saved by user |

