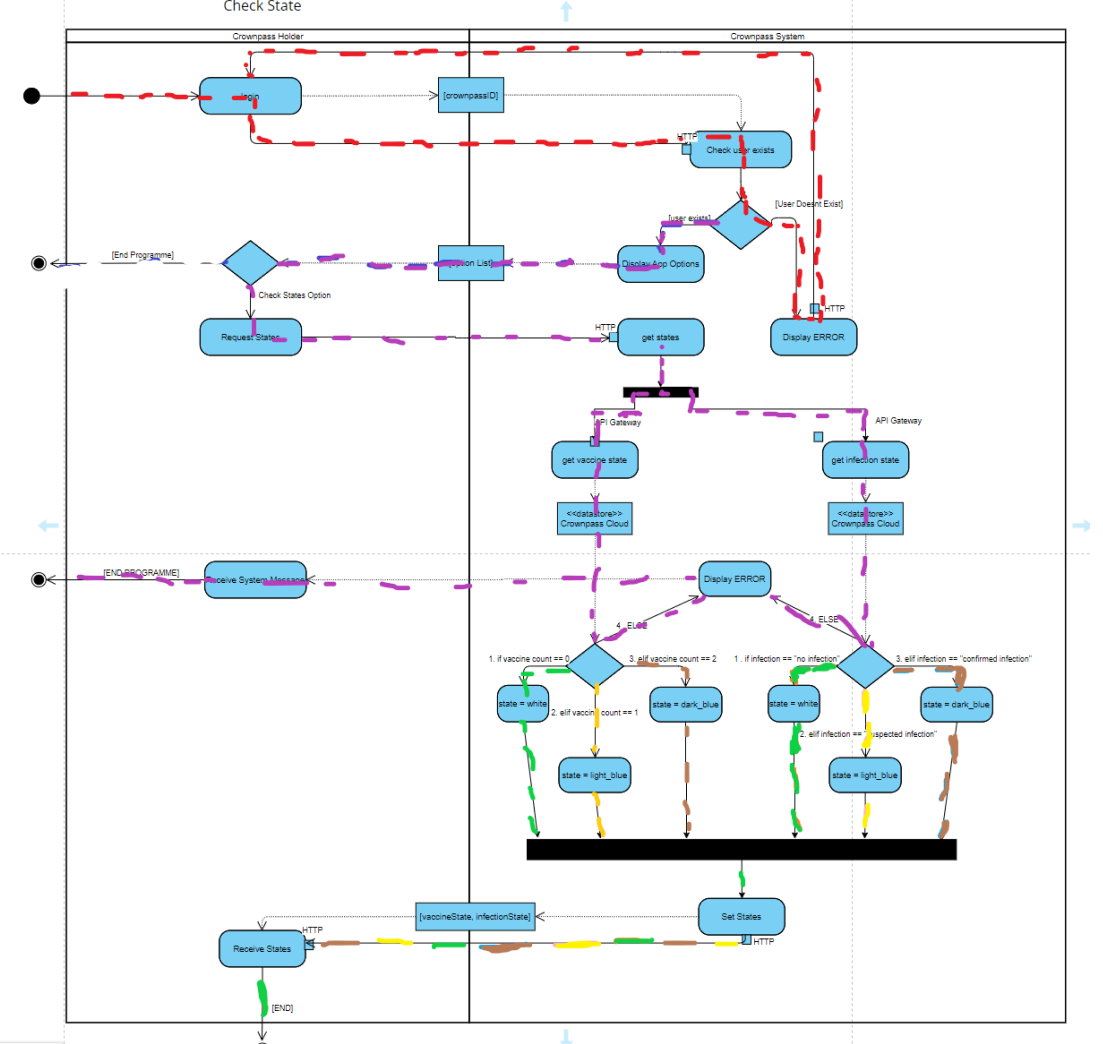
System test plan

“Check State” use case

# Deriving Scenarios from Activity Diagram

Each colour represents a different path. Each path creates a unique scenario. There are in total 5 scenarios generated from this activity diagram:



# Test Case 1 (red line from activity diagram)

The “Check State” use case specifies that a holder MUST be registered to perform “Check State”.

**Use case:** Check State

**Scenario:** Unable to Check State if not registered

|  |  |
| --- | --- |
| Crownpass Holder | System |
| 1) Holder tries to check state without valid crownpassID | 2) Confirm Holder is not registered. Displays error. Redirect user back to login. |
| 3) Receive error message. Redirected to login. |  |

**Test Data**

|  |  |  |
| --- | --- | --- |
|  | Variable | Test Data |
| Input | CrowpassID | 123456 |
| Stored info (Holder data Manager) | Valid\_CrownpassIDs | 111111, 222222 |
| Expected Output | Server Message | INVALID CrownpassID |

**Test Process**

1. Set up test context.
2. Crownpass Holder database: Contains Valid CrownpassIDs.
3. Test user: Does not have valid CrownpassID.
4. Tester: Input invalid CrownpassID.
5. System: Check CrowpassID.
6. Expected output: “INVALID CrownpassID”.
7. Check: (a) Error message is correct; (b) user is not logged in and CANNOT check state.
8. END.

# Test Case 2 (Purple line from activity diagram)

**Use Case:** Check State

**Scenario:** Holder is registered and chooses check state option, but not able to check state due to having no vaccine or infection status.

|  |  |
| --- | --- |
| Crownpass Holder | System |
| 1) Holder chooses check state menu option | 2) Confirm Holder is registered |
|  | 3) Calculate states for holder |
|  | 4) Display error as no states exist. |
| 5) Receive error message. |  |

**Test Data**

|  |  |  |
| --- | --- | --- |
|  | Variable | Test Data |
| Input | CrowpassID | 123456 |
| Menu Option | Check State |
| Stored info (Holder Data Manager) | Valid\_CrownpassIDs | 111111, 222222, 123456 |
| Stored info (Vaccine Data Manager) | amountOfVaccines | null |
| Stored info (Test Data Manager) | InfecionState | null |
| Expected Output | Server Message | No states exist |

**Test Process**

1. Set up test context
2. Test User: Has valid CrownpassID.
3. Crownpass Holder database: Has no states.
4. Tester: chooses “check state” option.
5. System: Checks CrownpassID.
6. Expected result: User can check state.
7. System gets vaccination & infection states.
8. Expected output: No states exist.
9. Check: Error messaged received is correct.
10. END.

# Test Case 3 (Green lines from activity diagram)

**Use case:** Check State

**Scenario:** Holder is registered, Holder has no vaccines & is not infected. Holder chooses “check state” option and is displayed the colour of their vaccination and infection states.

|  |  |
| --- | --- |
| Crownpass Holder | System |
| 1) Holder chooses check state menu option | 2) Confirm Holder is registered |
|  | 3) Get amount of vaccine for holder (0) and calculate the state colour (white) |
|  | 4) Get infection state (not infected) and calculate the state colour (Green) |
| 5) Receive correct state colours (white & green). |  |

**Test Data**

|  |  |  |
| --- | --- | --- |
|  | Variable | Test Data |
| Input | CrowpassID | 123456 |
| Menu Option | Check State |
| Stored info (Holder Data Manager) | Valid CrownpassIDs | 111111, 222222, 123456 |
| Stored info (Vaccine Data Manager) | amountOfVaccines | 0 |
| Stored info (Test Data Manager) | InfectionState | “no infection” |
| Expected Output | Vaccination state | White |
| Infection state | Green |

**Test Process**

1. Set up test context
2. Test User: Has valid CrownpassID.
3. Holder Data Manager: Has a stored CrowpassID that matches test user’s ID.
4. Vaccine Data Manager: amountOfVaccines for test user = 0
5. Test Data Manager: infectionState for test user = “no infection”
6. Tester chooses “check state” option.
7. System checks CrownpassID.
8. Expected result: User is accepted and can check state
9. System gets vaccination state.
10. Expected result: (a) System gets correct number of vaccines; (b) System displays White state.
11. Check: White is displayed.
12. System gets infection state.
13. Expected result: (a) System gets infection state; (b) System displays Green state.
14. Check: Green is displayed.
15. END.

# Test Case 4 (Yellow lines from activity diagram)

**Use case:** Check State

**Scenario:** Holder is registered, Holder has 1 vaccine & is suspected to be infected. Holder chooses “check state” option and is displayed the colour of their vaccination and infection states.

|  |  |
| --- | --- |
| Crownpass Holder | System |
| 1) Holder chooses check state menu option | 2) Confirm Holder is registered |
|  | 3) Get amount of vaccine for holder (1) and calculate the state colour (light blue) |
|  | 4) Get infection state (suspected infection) and calculate the state colour (amber) |
| 5) Receive correct state colours (light blue & amber). |  |

**Test Data**

|  |  |  |
| --- | --- | --- |
|  | Variable | Test Data |
| Input | CrowpassID | 123456 |
| Menu Option | Check State |
| Stored info (Holder Data Manager) | Valid CrownpassIDs | 111111, 222222, 123456 |
| Stored info (Vaccine Data Manager) | amountOfVaccines | 1 |
| Stored info (Test Data Manager) | InfectionState | “suspected infection” |
| Expected Output | Vaccination state | Light blue |
| Infection state | Amber |

**Test Process**

1. Set up test context
2. Test User: Has valid CrownpassID.
3. Holder Data Manager: Has a stored CrowpassID that matches test user’s ID.
4. Vaccine Data Manager: amountOfVaccines for test user = 1
5. Test Data Manager: infectionState for test user = “suspected infection”
6. Tester chooses “check state” option.
7. System checks CrownpassID.
8. Expected result: User is accepted and can check state
9. System gets vaccination state.
10. Expected result: (a) System gets correct number of vaccines; (b) System displays light blue state.
11. Check: light blue is displayed.
12. System gets infection state.
13. Expected result: (a) System gets infection state; (b) System displays Amber state.
14. Check: Amber is displayed.
15. END.

# Test Case 5 (Brown lines from activity diagram)

**Use case:** Check State

**Scenario:** Holder is registered, Holder has 2 vaccine & has a confirmed infecttion. Holder chooses “check state” option and is displayed the colour of their vaccination and infection states.

|  |  |
| --- | --- |
| Crownpass Holder | System |
| 1) Holder chooses check state menu option | 2) Confirm Holder is registered |
|  | 3) Get amount of vaccine for holder (2) and calculate the state colour (dark blue) |
|  | 4) Get infection state (suspected infection) and calculate the state colour (red) |
| 5) Receive correct state colours (light blue & amber). |  |

**Test Data**

|  |  |  |
| --- | --- | --- |
|  | Variable | Test Data |
| Input | CrowpassID | 123456 |
| Menu Option | Check State |
| Stored info (Holder Data Manager) | Valid CrownpassIDs | 111111, 222222, 123456 |
| Stored info (Vaccine Data Manager) | amountOfVaccines | 2 |
| Stored info (Test Data Manager) | InfectionState | “confirmed infection” |
| Expected Output | Vaccination state | Dark blue |
| Infection state | Red |

**Test Process**

1. Set up test context
2. Test User: Has valid CrownpassID.
3. Holder Data Manager: Has a stored CrowpassID that matches test user’s ID.
4. Vaccine Data Manager: amountOfVaccines for test user = 2
5. Test Data Manager: infectionState for test user = “Confirmed infection”
6. Tester chooses “check state” option.
7. System checks CrownpassID.
8. Expected result: User is accepted and can check state
9. System gets vaccination state.
10. Expected result: (a) System gets correct number of vaccines; (b) System displays dark blue state.
11. Check: dark blue is displayed.
12. System gets infection state.
13. Expected result: (a) System gets infection state; (b) System displays Red state.
14. Check: Red is displayed.
15. END.