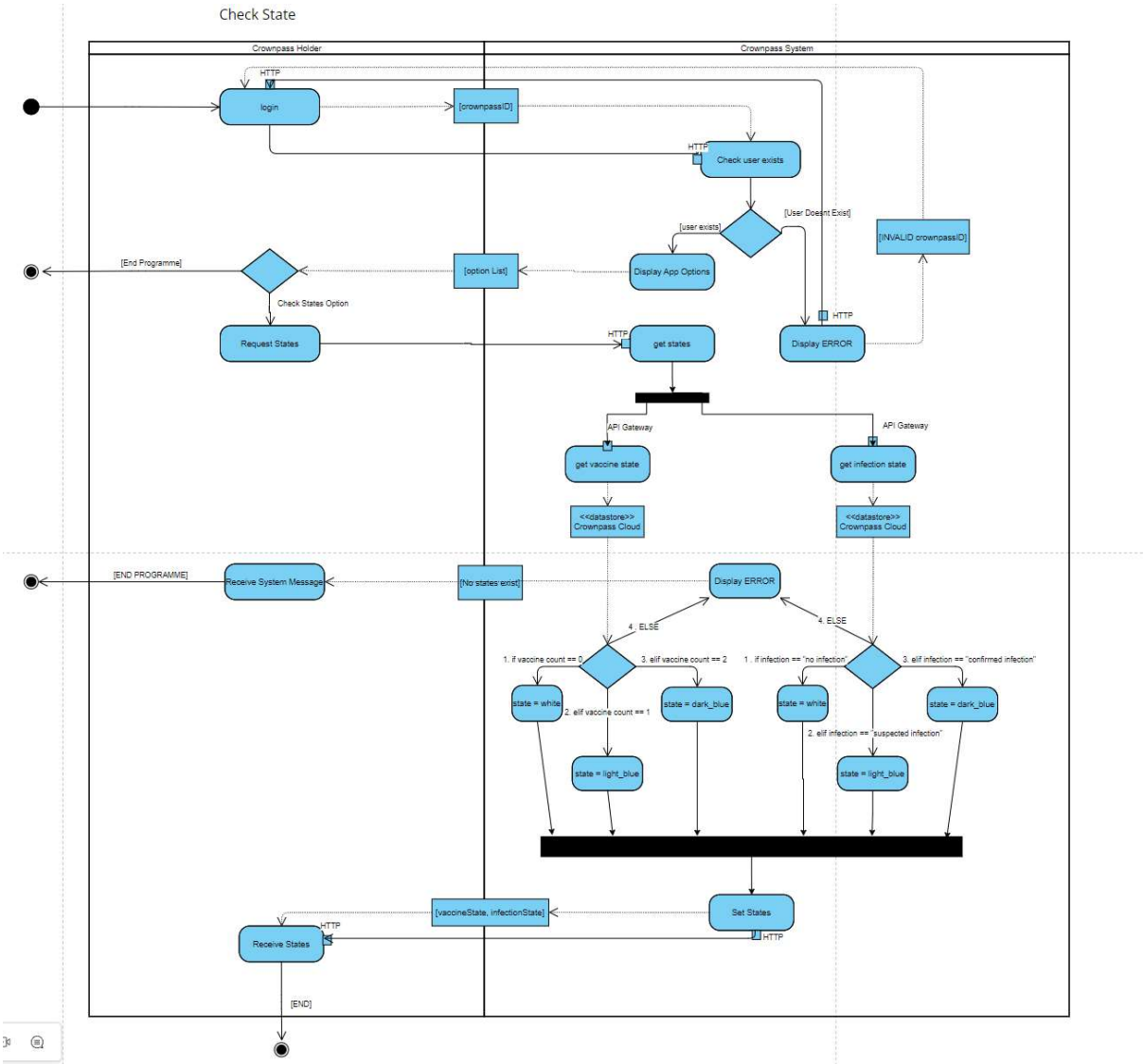
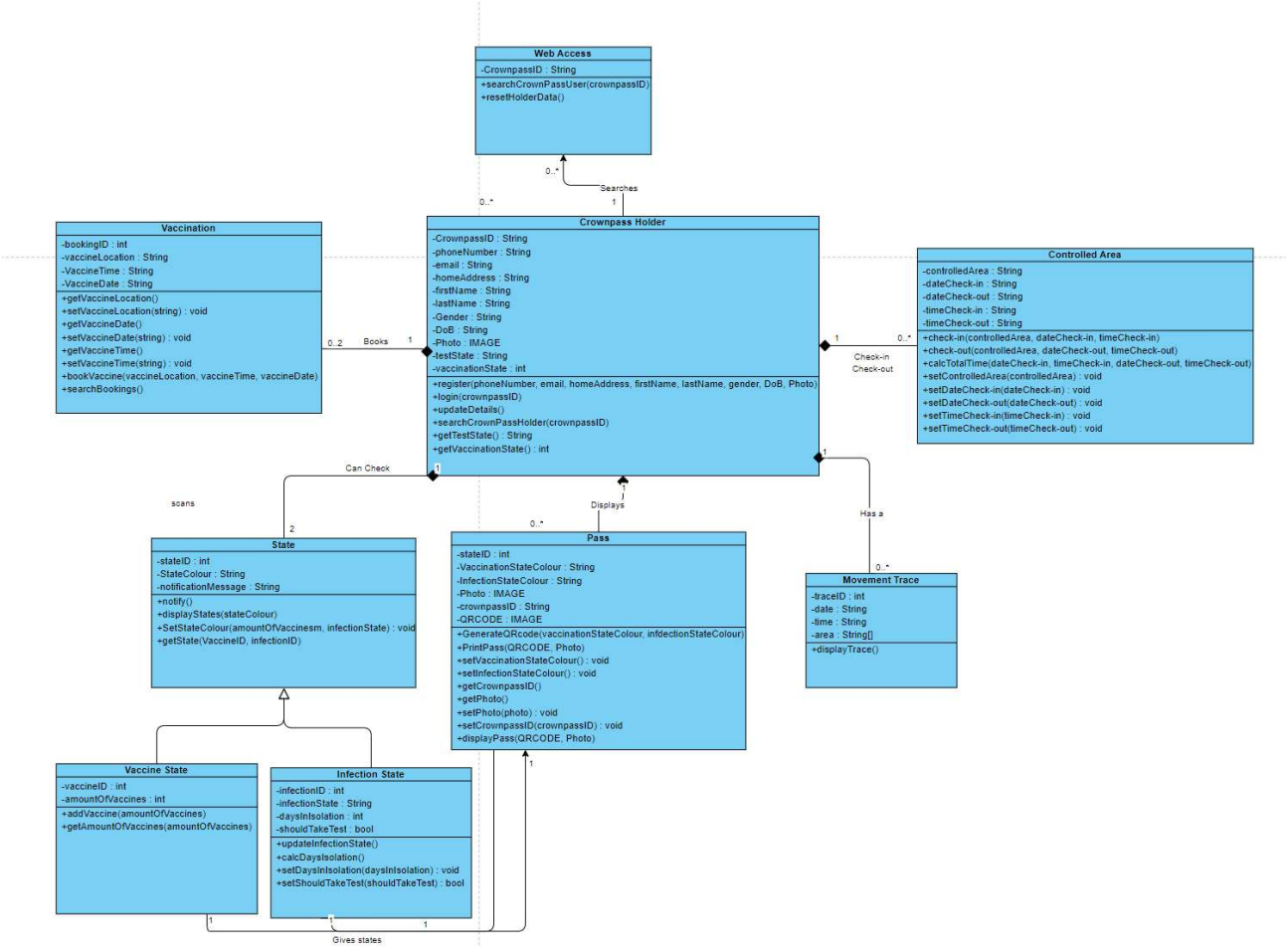
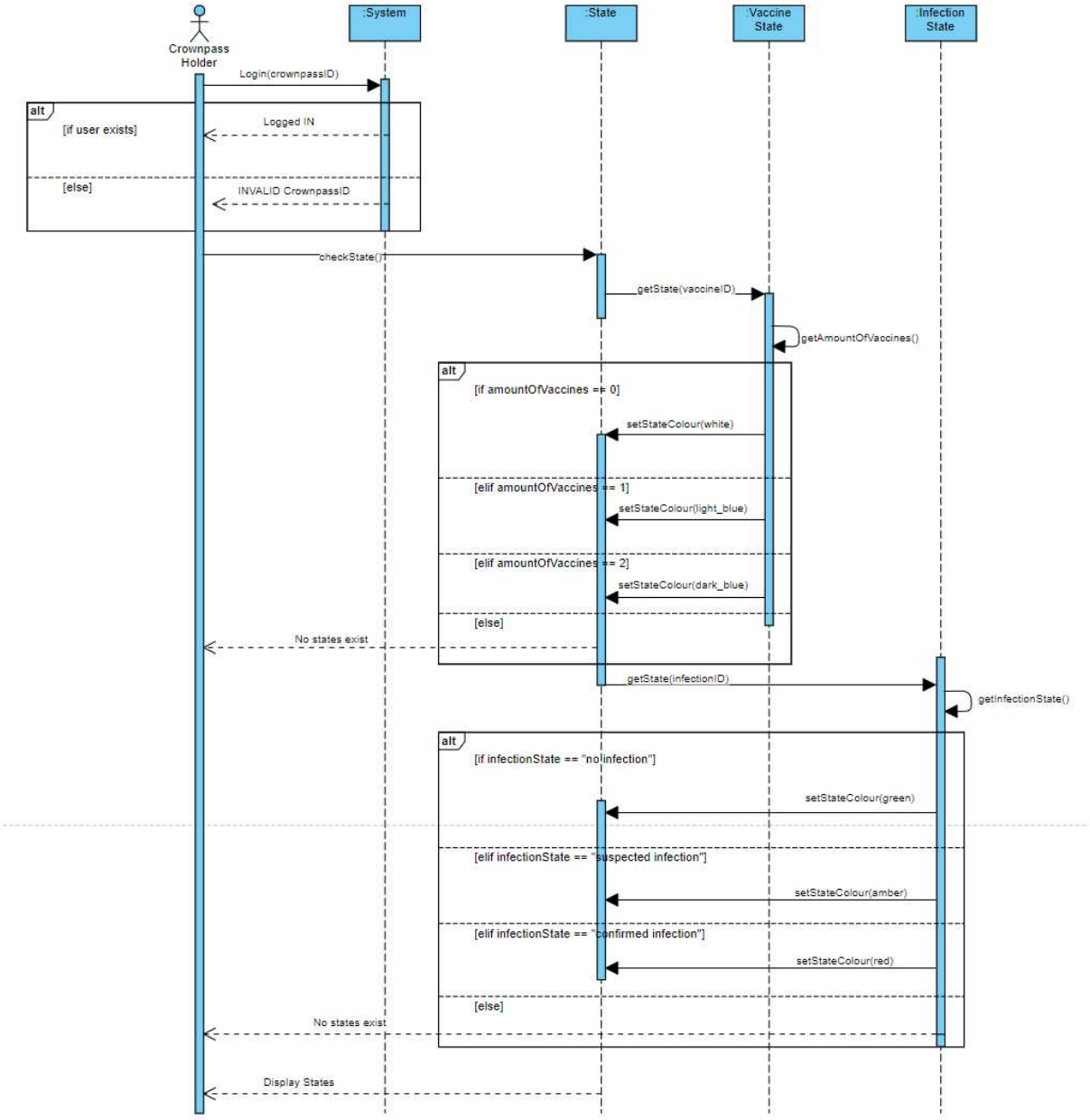
# Holder Activity Diagram – Check State



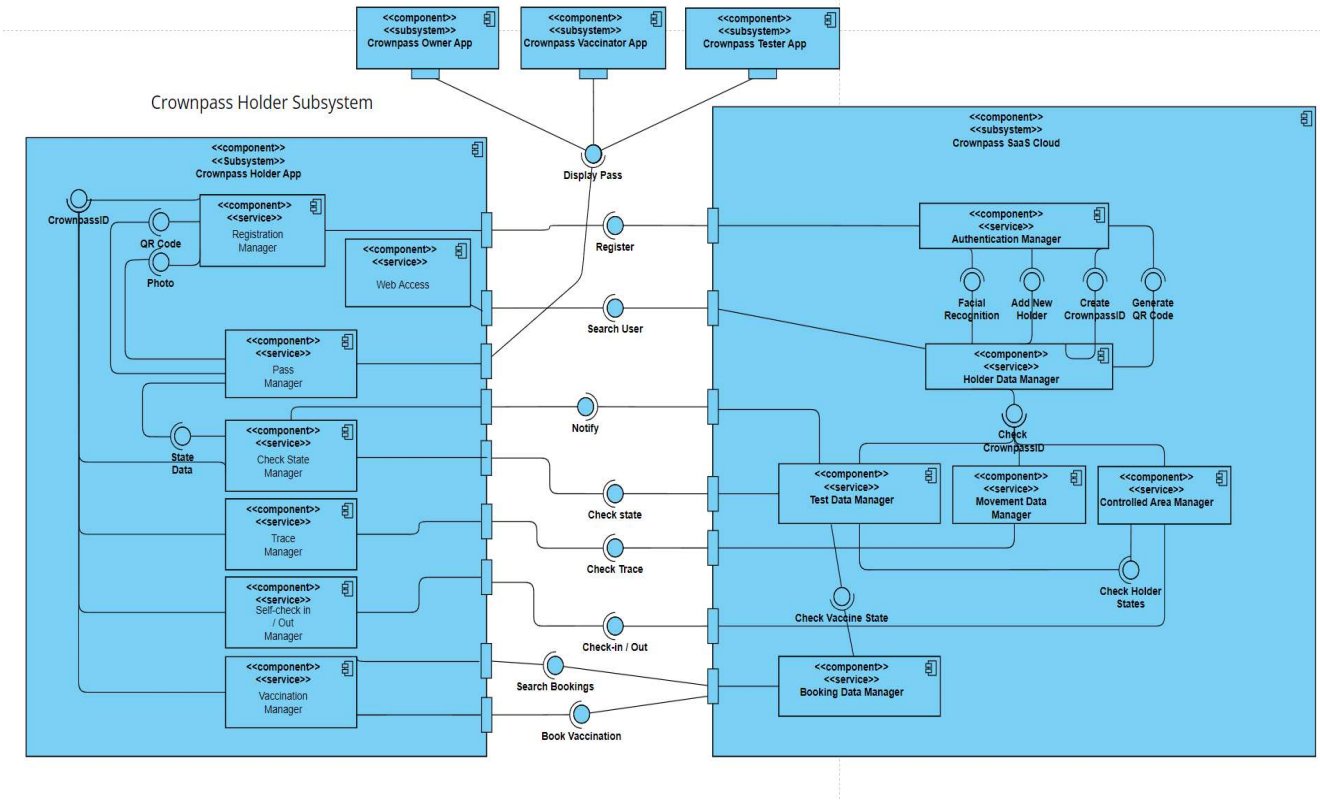
# Holder Class Diagram



# Holder Sequence Diagram – check state



## Holder Component Diagram




## Holder Unit Test Plan (Pass Class)

| Test Case | Attributes/ variables used | Method and Parameters | Expected Output |
|---|---|---|---|
| Generate QR Code successfully | VaccinationStateColour = “White” InfectionStateColour = “Green” | GenerateQRcode(VaccinationStateColour : String, InfectionStateColour : String) | QR code image should be generated. QR Code should include the correct vaccine state and infection state. |

| | | | |
|---|---|---|---|
| Generate QR Code unsuccessful | VaccinationStateColour = error InfectionStateColour = error | GenerateQRcode(VaccinationStateColour : String, InfectionStateColour : String) | QR Code should not be generated and display an ERROR. |
| | | | |
| Print Pass on Paper Success | QRCODE | PrintPass(QRCODE : IMAGE) | QR code should be printed on a piece of paper. |
| Print Pass on Paper unsuccessful. | | PrintPass(null : IMAGE) | ERROR Message. QRCODE should not be printed. |
| | | | |
| Set the vaccination state colour to white | From state class: stateColour = “white” | setVaccinationStateColour(stateColour : String) | Vaccination State set to White. |
| Set the vaccination state colour to white | From state class: stateColour = “light blue” | setVaccinationStateColour(stateColour : String) | Vaccination State set to light blue. |
| Set the vaccination state colour to white | From state class: stateColour = “dark blue” | setVaccinationStateColour(stateColour : String) | Vaccination State set to dark blue. |
| set vaccination colour to null | From state class: stateColour = “” | setVaccinationStateColour(stateColour : String) | Display error. |
| | | | |
| Set the infection state colour to green | From state class: stateColour = “green” | setInfectionStateColour(stateColour : String) | Infection State set to Green. |
| Set the infection state colour to amber | From state class: stateColour = “amber” | setInfectionStateColour(stateColour : String) | Infection State set to Amber. |
| Set the infection state | From state class: stateColour = “red” | setInfectionStateColour(stateColour : String) | Infection State set to Red. |

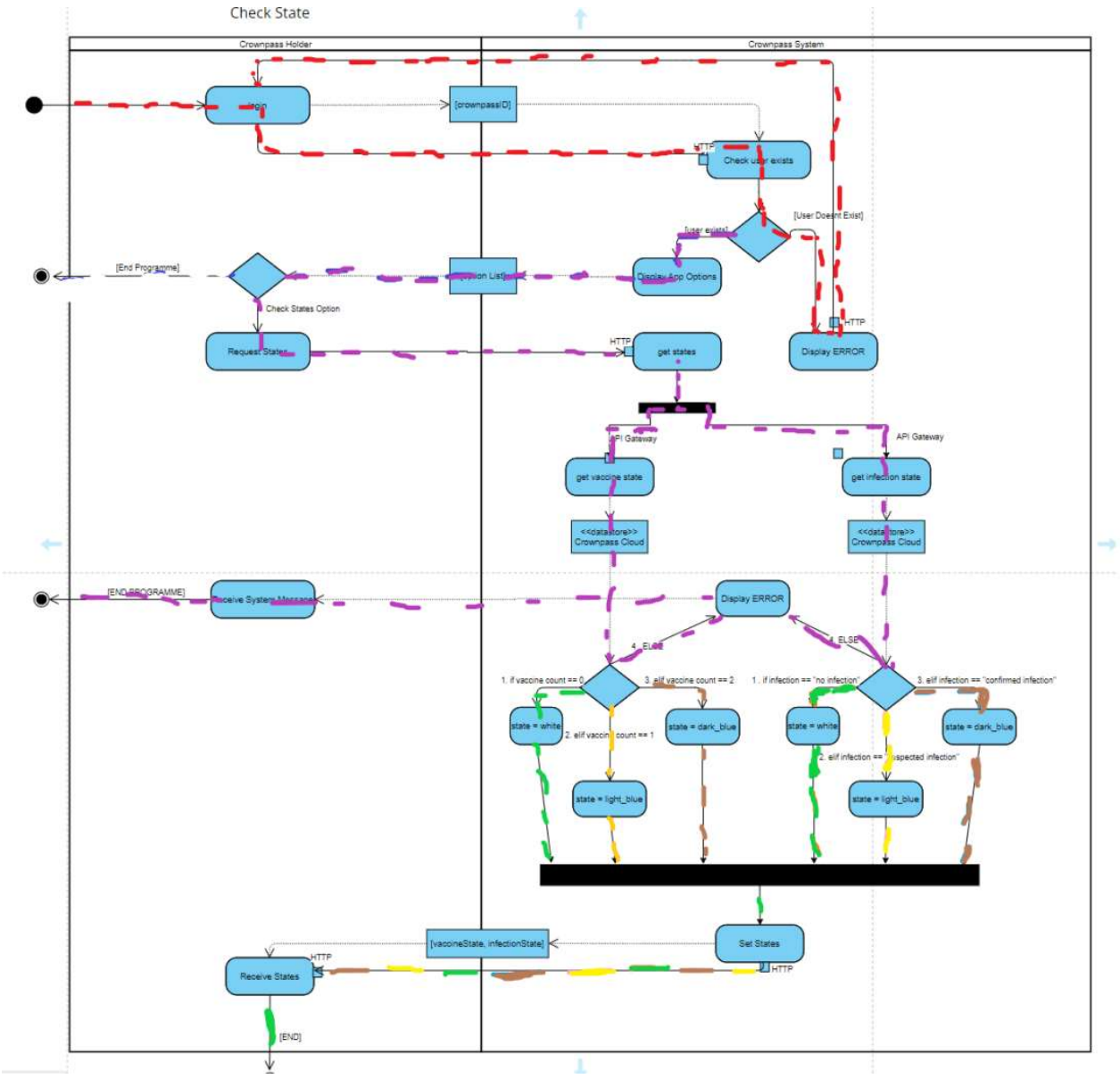
| | | | |
|---|---|---|---|
| colour to red | | | |
| set infection colour to null | From state class: stateColour = “” | setInfectionStateColour(stateColour : String) | Display error. |
| | | | |
| Get the crownpass Holders ID and set the local crowpassID variable CORRECTLY. | crownpassID | getCrownpassID()<br>setCrownpassID(crownpassID :String) | Local crownpassID should match logged in users ID. |
| Get the crownpass Holders ID and set the local crowpassID variable INCORRECTLY. (check error handling) | | getCrownpassID()<br>setCrownpassID(null :String) | Show error. Log user out. |
| | | | |
| Get Holders PHOTO from another class and set it as a local variable CORRECTLY. | Photo | getPhoto()<br>setPhoto(Photo : IMAGE) | Photo now accessible by Pass class. |
| Get Holders PHOTO from another class and set it as a local variable INCORRECTLY. (check | | getPhoto()<br>setPhoto(null : IMAGE) | Show error. Log user out. |

| | | | |
|---|---|---|---|
| error handling) | | | |
| | | | |

## 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.
   a. Crownpass Holder database: Contains Valid CrownpassIDs.
   b. Test user: Does not have valid CrownpassID.
2. Tester: Input invalid CrownpassID.
3. System: Check CrowpassID.
   a. Expected output: “INVALID CrownpassID”.
   b. Check: (a) Error message is correct; (b) user is not logged in and CANNOT check state.
4. 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
   a. Test User: Has valid CrownpassID.
   b.
   c. Crownpass Holder database: Has no states.
2. Tester: chooses “check state” option.
3. System: Checks CrownpassID.
   a. Expected result: User can check state.
4. System gets vaccination & infection states.
   a. Expected output: No states exist.
   b. Check: Error messaged received is correct.
5. 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
   a. Test User: Has valid CrownpassID.
   b. Holder Data Manager: Has a stored CrowpassID that matches test user's ID.
   c. Vaccine Data Manager: amountOfVaccines for test user = 0
   d. Test Data Manager: infectionState for test user = "no infection"
2. Tester chooses "check state" option.
3. System checks CrownpassID.
   a. Expected result: User is accepted and can check state
4. System gets vaccination state.
   a. Expected result: (a) System gets correct number of vaccines; (b) System displays White state.
   b. Check: White is displayed.
5. System gets infection state.
   a. Expected result: (a) System gets infection state; (b) System displays Green state.
   b. Check: Green is displayed.
6. 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**

7. Set up test context
    e. Test User: Has valid CrownpassID.
    f. Holder Data Manager: Has a stored CrowpassID that matches test user's ID.
    g. Vaccine Data Manager: amountOfVaccines for test user = 1
    h. Test Data Manager: infectionState for test user = "suspected infection"
8. Tester chooses "check state" option.
9. System checks CrownpassID.
    b. Expected result: User is accepted and can check state
10. System gets vaccination state.
    c. Expected result: (a) System gets correct number of vaccines; (b) System displays light blue state.
    d. Check: light blue is displayed.
11. System gets infection state.
    c. Expected result: (a) System gets infection state; (b) System displays Amber state.
    d. Check: Amber is displayed.
12. 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**

13. Set up test context
    i. Test User: Has valid CrownpassID.
    j. Holder Data Manager: Has a stored CrowpassID that matches test user's ID.
    k. Vaccine Data Manager: amountOfVaccines for test user = 2
    l. Test Data Manager: infectionState for test user = "Confirmed infection"
14. Tester chooses "check state" option.
15. System checks CrownpassID.
    c. Expected result: User is accepted and can check state
16. System gets vaccination state.
    e. Expected result: (a) System gets correct number of vaccines; (b) System displays dark blue state.
    f. Check: dark blue is displayed.
17. System gets infection state.
    e. Expected result: (a) System gets infection state; (b) System displays Red state.
    f. Check: Red is displayed.
18. END.