## 

[Disclaimer](#_uz2bs4660cyv)

[Note on Decision points](#_13ucey2f9ri6)

[US001](#_627xyhu2208t)

[Use Case Testing](#_6s39bjeglkc)

[Variables for TC-1 and TC-2](#_yxtskvve1m9)

[Variables options for each step](#_ql61jy5yjxjh)

[Test Case Matrix](#_qq3wzwd93dme)

[US002](#_flldgcouevhv)

[Use Case Testing](#_ssbwvpatnj0r)

[Variables for TC-3, TC-4, and TC-5](#_rsbhus1zzk2)

[Variables options for each step](#_wvdtj8edfive)

[Test Case Matrix](#_3e55vc3088i3)

[US003](#_iqklj4rk5avf)

[Use Case Testing](#_o8544sx0nlja)

[Variables for TC-6, TC-7, and TC-8](#_akdy7eqe7xiy)

[Variables options for each step](#_g51esqyblek8)

[Test Case Matrix](#_7at9iep323vq)

[Problem Description](#_fg1pcdul7vio)

[US005](#_yy2hzls9qjtp)

[Use Case Testing](#_w9c8q3qq0k0j)

[Variables for TC-9 and TC-10](#_huwpwhbxm03x)

[Variables options for each step](#_4d01p46z7tu7)

[Test Case Matrix](#_5bewolyboqca)

## 

## 

|  |
| --- |
| Disclaimer |
| *Telegram is blocked in Russia:*  [*A letter from the FSB about an administrative offense (in Russian)*](https://agora.legal/fs/a_delo2doc/55_file_Telegram_FSB_140917.pdf)  *It was the customer’s requirement for the product to be developed as a Telegram bot (see* [*Requirements document*](https://github.com/maloescher/BBQReserverOfTeam3/blob/master/%5BRE%5D%20Use%20Cases.pdf) *page 2).emo\_user1* |

## Note on Decision points

*For every test case, decision points are covered in the name of the test case step. So that:*

* *Await customer input (“A”) will correspond to TC-#-A#,*
* *System action (“S”) will correspond to TC-#-S#,*
* *and Exception condition (“E”) will correspond to TC-#-E#.*

# US001

|  |  |
| --- | --- |
| **Use Case Code US001** | Update reservation record |
| **Actors** | User, Server |
| **Pre-conditions** | User has a reservation record and request update |
| **Assumptions** | User wants to update something in his record |
| **Flow of events** | 1. User request update for his record 2. User choose data to modify 3. System validates the data   3.1 If new data is valid - modify record  3.2 If something went wrong - return error   1. User receives confirmation message |
| **Post-conditions** | Server informs the user is update successful or not |
| **Alternate flow** | There could be some system or network error, send user notification to try again |

## Use Case Testing

|  |  |  |  |
| --- | --- | --- | --- |
| **Test case name:** | Update reservation record without overlap | | |
| **Test ID:** | **TC-1** | | |
| **Test suite:** | Update reservation record | | |
| **Priority:** |  | | |
| **Setup:** | 1. User is logged in. 2. User has started a Bot. 3. User has a reservation. | | |
| **Teardown:** | Change the modified reservation to the previous version (before change). Save changes. | | |
| **Step:** | **Description** | **Result** | **Problem ID** |
| TC-1-A1 | The user presses “Update or Remove an existing reservation” button. | Passed |  |
| TC-1-S2 | The system responds with all available reservations. | Passed |  |
| TC-1-A3 | User selects a reservation from the list to update. | Passed |  |
| TC-1-A4 | The User presses the “Update” button. | Passed |  |
| TC-1-A5 | The User selects the new starting time. | Passed |  |
| TC-1-A6 | The User selects the new ending time. | Passed |  |
| TC-1-S7 | The appointment gets updated. | Passed |  |
| **Status:** | **Passed** | | |
| **Tester:** | **All step is passed without any bug.** | | |
| **Date complete:** | **23 November 2019** | | |
| **Test run** | [**https://youtu.be/7Er2gzrP-tk**](https://youtu.be/7Er2gzrP-tk) | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Test case name:** | **Update reservation with overlap** | | |
| **Test ID:** | **TC-2** | | |
| **Test suite:** | Update reservation record | | |
| **Priority:** |  | | |
| **Setup:** | 1. User is logged in. 2. User has started a Bot. 3. User has a reservation. | | |
| **Teardown:** | Change the modified reservation to the previous version (before change). Save changes. | | |
| **Step:** | **Description** | **Result** | **Problem ID** |
| TC-2-A1 | The user presses “Update or Remove an existing reservation” button. | Passed |  |
| TC-2-S2 | The system responds with all available reservations. | Passed |  |
| TC-2-A3 | The user selects a reservation from the list to update. | Passed |  |
| TC-2-A4 | The User presses the “Update” button. | Passed |  |
| TC-2-A5 | The User selects the new starting time (which overlaps with time for existing reservations). | Passed |  |
| TC-2-A6 | The User selects the new ending time. | Passed |  |
| TC-2-S7 | The appointment has not been updated. | Warning | W-1 |
| **Status:** | **Passed** | | |
| **Tester:** | **All step is passed with a warning.** | | |
| **Date complete:** | **23 November 2019** | | |
| **Test run** | [**https://youtu.be/yNgjgq1f9fM**](https://youtu.be/yNgjgq1f9fM) | | |

## Variables for TC-1 and TC-2

|  |  |
| --- | --- |
| **Step Number** | **Variable** |
| A1 | No variable or selection |
| S2 | No variable or selection |
| A3 | Reservation record (string) |
| A4 | No variable or selection |
| A5 | Start time (string) |
| A6 | End time (string) |
| A7 | No variable or selection |

## Variables options for each step

|  |  |  |
| --- | --- | --- |
| Type | Name | Value |
| String | regular\_record | “23 November, 11:00-13:00” |
| invalid | “1235465trfhgrty” |
| time | “11:00” |
| overlap\_time | “14:00” |
| min\_time | “8:00” |
| max\_time | “22:00” |

## Test Case Matrix

|  |  |  |  |
| --- | --- | --- | --- |
| Step | Variable or selection | TC-1 | TC-2 |
| A3 | Reservation record | regular\_record | regular\_record |
| A5 | Start time | time | overlap\_time |
| A6 | End time | max\_time | time |

## Problem Description

**W-1:** The test is passed. The reservation was not created but there is no “warning message” to inform the user about the creation status whether is created or not.

# US002

|  |  |
| --- | --- |
| **Use Case Code US002** | Create a new reservation |
| **Actors** | User, Server |
| **Pre-conditions** | The user wants to reserve the time at the BBQ place |
| **Flow of events** | 1. User clicks “Create new reservation” button 2. User enters a time interval 3. System checks if the time interval is available 4. Server approves the time interval 5. User approves his reservation request 6. Server saves reservation to the schedule |
| **Post-conditions** | The schedule is updated by adding a new entry. |
| **Alternate flows and exceptions** | The inputted time interval is not available |
| **Issues** | Duplicates the existing Concierge service feature |

## Use Case Testing

|  |  |  |  |
| --- | --- | --- | --- |
| **Test case name:** | **Approve a reservation creation** | | |
| **Test ID:** | **TC-3** | | |
| **Test suite:** | Create a new reservation | | |
| **Priority:** |  | | |
| **Setup:** | 1. User is logged in. 2. User has started a Bot. | | |
| **Teardown:** | Delete created reservation. | | |
| **Step:** | **Description** | **Result** | **Problem ID** |
| TC-3-A1 | The user presses “Create a new reservation” button. | Passed |  |
| TC-3-S2 | The system responds with the list of months. | Passed |  |
| TC-3-A3 | User selects a month from the list to create a reservation. | Passed |  |
| TC-3-S4 | The system responds with the request to the user to select the day. | Passed |  |
| TC-3-A5 | The User enters a date (day number). | Passed |  |
| TC-3-S6 | The system responds with the list of available time slots for the start of reservation. | Passed |  |
| TC-3-A7 | The User selects time slots for the start of reservation from the list. | Passed |  |
| TC-3-S8 | The system responds with the list of available time slots for the end of reservation. | Passed |  |
| TC-3-A9 | The User selects a time slot for the end of reservation from the list. | Passed |  |
| TC-3-S10 | The system responds with the request to the user to approve the creation of the reservation. | Passed |  |
| TC-3-A11 | The user presses the “Approve” button. | Passed |  |
| TC-3-S12 | The reservation is created. | Passed |  |
| **Status:** | **Passed** | | |
| **Tester:** | **All step is passed without any bug.** | | |
| **Date complete:** | **23 November 2019** | | |
| **Test run** | [**https://youtu.be/FeF\_gz-Qnh0**](https://youtu.be/FeF_gz-Qnh0) | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Test case name:** | **Decline a reservation creation** | | |
| **Test ID:** | **TC-4** | | |
| **Test suite:** | Create a new reservation | | |
| **Priority:** |  | | |
| **Setup:** | 1. User is logged in. 2. User has started a Bot. | | |
| **Teardown:** | Delete created reservation. | | |
| **Step:** | **Description** | **Result** | **Problem ID** |
| TC-4-A1 | The user presses “Create a new reservation” button. | Passed |  |
| TC-4-S2 | The system responds with the list of months. | Passed |  |
| TC-4-A3 | User selects a month from the list to create a reservation. | Passed |  |
| TC-4-S4 | The system responds with the request to the user to select the day. | Passed |  |
| TC-4-A5 | The User enters a date (day number). | Passed |  |
| TC-4-S6 | The system responds with the list of available time slots for the start of the reservation. | Passed |  |
| TC-4-A7 | The User selects time slots for the start of reservation from the list. | Passed |  |
| TC-4-S8 | The system responds with the list of available time slots for the end of the reservation. | Passed |  |
| TC-4-A9 | The User selects time slots for the end of the reservation from the list. | Passed |  |
| TC-4-S10 | The system responds with the request to the user to approve the creation of the reservation. | Passed |  |
| TC-4-A11 | The user presses “I’ve changed my mind” button. | Passed |  |
| TC-4-S12 | The reservation is not created. | Passed |  |
| **Status:** | **Passed** | | |
| **Tester:** | **All step is passed without any bug.** | | |
| **Date complete:** | **23 November 2019** | | |
| **Test run** | [**https://youtu.be/FvQMHxPoBeg**](https://youtu.be/FvQMHxPoBeg) | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Test case name:** | **Create reservation with the start time equal with the end time of existing reservation** | | |
| **Test ID:** | **TC-5** | | |
| **Test suite:** | Create a new reservation | | |
| **Priority:** |  | | |
| **Setup:** | 1. User is logged in. 2. User has started a Bot. 3. User has a reservation. | | |
| **Teardown:** | Delete created reservation. | | |
| **Step:** | **Description** | **Result** | **Problem ID** |
| TC-5-A1 | The user presses “Create a new reservation” button. | Passed |  |
| TC-5-S2 | The system responds with the list of months. | Passed |  |
| TC-5-A3 | User selects the same month as for existing reservation from the list to create a reservation. | Passed |  |
| TC-5-S4 | The system responds with the request to the user to select the day. | Passed |  |
| TC-5-A5 | The User enters the same date (day number) as for existing reservation. | Passed |  |
| TC-5-S6 | The system responds with the list of available time slots for the start of reservation. | Passed |  |
| TC-5-A7 | The User selects a time slot for the start of reservation from the list corresponding to the end time of existing reservation. | Passed |  |
| TC-5-S8 | The system responds with a suggestion to merge them. | Passed |  |
| TC-5-S9 | The reservation was created without merging each other. | Passed |  |
| **Status:** | **Passed** | | |
| **Tester:** | **All step is passed without any bug.** | | |
| **Date complete:** | **23 November 2019** | | |
| **Test run** | [**https://youtu.be/auKgDbwJTQ4**](https://youtu.be/auKgDbwJTQ4) | | |

## Variables for TC-3, TC-4, and TC-5

|  |  |
| --- | --- |
| **Step Number** | **Variable** |
| A1 | No variable or selection |
| S2 | No variable or selection |
| A3 | Month (string) |
| S4 | No variable or selection |
| A5 | Day (integer) |
| S6 | No variable or selection |
| A7 | Start time (string) |
| S8 | No variable or selection |
| A9 | End time (string) |
| S10 | No variable or selection |
| A11 | Approve option |
| S12 | No variable or selection |

## Variables options for each step

|  |  |  |
| --- | --- | --- |
| Type | Name | Value |
| String | regular\_month | “November” |
| invalid | “1235465trfhgrty” |
| time | “11:00” |
| adjacent\_time | “14:00” |
| min\_time | “8:00” |
| max\_time | “22:00” |
| Integer | regular\_day | 2 |
| max\_day | 31 |
| min\_day | 1 |
| invalid\_day | 0 |
| Approve options | approve | “Approve” |
| decline | “I’ve changed my mind” |

## Test Case Matrix

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Step | Variable or selection | TC-3 | TC-4 | TC-5 |
| A3 | Month | regular\_month | regular\_month | regular\_month |
| A5 | Day | regular\_day | regular\_day | regular\_day |
| A7 | Start time | time | time | time |
| A9 | End time | max\_time | max\_time | adjacent\_time |
| A11 | Approve option | approve | decline | - |

# US003

|  |  |
| --- | --- |
| **Use Case Code US003** | Delete a booking record |
| **Actors** | User, Server |
| **Pre-conditions** | User has a booking record |
| **Assumptions** | - |
| **Flow of events** | 1. User press modify button 2. User press delete button 3. System delete the record |
| **Post-conditions** | The reservation is deleted. |
| **Alternate flow** | 1. User cancels the operation 2. System return to the Modify screen |

## Use Case Testing

|  |  |  |  |
| --- | --- | --- | --- |
| **Test case name:** | **Delete a booking record** | | |
| **Test ID:** | **TC-6** | | |
| **Test suite:** | **Delete a booking record** | | |
| **Priority:** |  | | |
| **Setup:** | 1. The User is logged in. 2. The User has started the Bot. 3. The User has a reservation. | | |
| **Teardown:** | Recreate the deleted booking record. | | |
| **Step:** | **Description** | **Result** | **Problem ID** |
| TC-6-A1 | The User presses the “Update or Remove an existing reservation” button | Passed |  |
| TC-6-S2 | The system responds with buttons for every boked record of the user. | Passed |  |
| TC-6-A3 | The User selects the booking by pressing the corresponding button. | Passed |  |
| TC-6-A4 | The User selects the “Delete” button. | Passed |  |
| TC-6-S5 | The booking record is deleted. | Passed |  |
| **Status:** | **Passed** | | |
| **Tester:** | **All step is passed without any bug.** | | |
| **Date complete:** | **23 November 2019** | | |
| **Test run** | [**https://youtu.be/KEv--21D3ZY**](https://youtu.be/KEv--21D3ZY) | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Test case name:** | **Delete a booking record of another user** | | |
| **Test ID:** | **TC-7** | | |
| **Test suite:** | **Delete a booking record** | | |
| **Priority:** |  | | |
| **Setup:** | 1. The User is logged in. 2. The User has started the Bot. 3. Another User has a reservation. | | |
| **Teardown:** | If the system behaves as expected, nothing has to be done. | | |
| **Step:** | **Description** | **Result** | **Problem ID** |
| TC-7-A1 | The User presses the “Update or Remove an existing reservation” button | Passed |  |
| TC-7-S2 | The system responds with buttons for every boked record of the user. | Passed |  |
| TC-7-A3 | The User enters another existing entry by entering the corresponding String. | Passed |  |
| TC-7-A4 | The User selects the “Delete” button. | Passed |  |
| TC-7-S5 | The booking record is not deleted. | Failed | P-1 |
| **Status:** | **Failed** | | |
| **Tester:** | **There is one bug that system can delete reservation from another user.** | | |
| **Date complete:** | **23 November 2019** | | |
| **Test run** | [**https://youtu.be/ZXpuge0pxEk**](https://youtu.be/ZXpuge0pxEk) | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Test case name:** | **Delete a booking record without having a reservation** | | |
| **Test ID:** | **TC-8** | | |
| **Test suite:** | **Delete a booking record** | | |
| **Priority:** |  | | |
| **Setup:** | 1. The User is logged in. 2. The User has started the Bot. | | |
| **Teardown:** | If the system behaves as expected, nothing has to be done. | | |
| **Step:** | **Description** | **Result** | **Problem ID** |
| TC-8-A1 | The User presses the “Update or Remove an existing reservation” button | Passed |  |
| TC-8-S2 | The system responds with the standard buttons since there are no appointments. | Warning | W-2 |
| **Status:** | **Passed** | | |
| **Tester:** |  | | |
| **Date complete:** | **23 November 2019** | | |
| **Test run** | [**https://youtu.be/KYzQSjL1l0Y**](https://youtu.be/KYzQSjL1l0Y) | | |

## Variables for TC-6, TC-7, and TC-8

|  |  |
| --- | --- |
| **Step Number** | **Variable** |
| A1 | No variable or selection |
| S2 | No variable or selection |
| A3 | Reservation record (string) |
| A4 | No variable or selection |
| S5 | No variable or selection |

## Variables options for each step

|  |  |  |
| --- | --- | --- |
| Type | Name | Value |
| String | regular\_record | “23 November, 11:00-13:00” |
| invalid | “1235465trfhgrty” |
| other\_user\_record | “22 November, 09:00-10:00” |

## Test Case Matrix

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Step | Variable or selection | TC-6 | TC-7 | TC-8 |
| A3 | Reservation record | regular\_record | other\_user\_record | - |

## Problem Description

**P-1:** A user can delete a reservation created by another user by entering a date and time of his reservation.

**W-2:** The system shows the standard button and start to infinite loop.

# US005

|  |  |
| --- | --- |
| **Use Case Code US005** | Checking a schedule |
| **Actors** | User, Server |
| **Pre-conditions** | The user is logged in in the telegram, The used have the chat with Bot opened. The system shows to the user main menu. |
| **Assumption** | The user wants to view the schedule for some time period (day, week, month) |
| **Flow of events** | 1. The user presses the “My schedule” button in the menu. 2. The system shows all reservations at once. |
| **Post-conditions** | The system shows the user all of his/her reservations. |
| **Alternative flows** | None |

## Use Case Testing

|  |  |  |  |
| --- | --- | --- | --- |
| **Test case name:** | **Checking the schedule with a user reservation** | | |
| **Test ID:** | **TC-9** | | |
| **Test suite:** | Checking a schedule | | |
| **Priority:** |  | | |
| **Setup:** | 1. User is logged in. 2. User has started a Bot. 3. User has a reservation. | | |
| **Teardown:** |  | | |
| **Step:** | **Description** | **Result** | **Problem ID** |
| TC-9-A1 | The user presses the “My Schedule” button. | Passed |  |
| TC-9-S2 | The system shows all reservations at once which belong to the user. | Passed |  |
| **Status:** | **Passed** | | |
| **Tester:** | **All step is passed without any bug.** | | |
| **Date complete:** | **23 November 2019** | | |
| **Test run** | [**https://youtu.be/7xWQi-9cm7M**](https://youtu.be/7xWQi-9cm7M) | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Test case name:** | **Checking schedule without any reservation** | | |
| **Test ID:** | **TC-10** | | |
| **Test suite:** | Checking a schedule | | |
| **Priority:** |  | | |
| **Setup:** | 1. User is logged in. 2. User has started a Bot. 3. User has no reservation. | | |
| **Teardown:** |  | | |
| **Step:** | **Description** | **Result** | **Problem ID** |
| TC-10-A1 | The user presses the “My Schedule” button. | Passed |  |
| TC-10-S2 | The system responds with the standard buttons since there are no appointments. | Passed |  |
| **Status:** | **Passed** | | |
| **Tester:** | **All step is passed without any bug.** | | |
| **Date complete:** | **23 November 2019** | | |
| **Test run** | [**https://youtu.be/dBchNPwI9Uc**](https://youtu.be/dBchNPwI9Uc) | | |

## Variables for TC-9 and TC-10

|  |  |
| --- | --- |
| **Step Number** | **Variable** |
| A1 | No variable or selection |
| S2 | No variable or selection |

Since, for the operation of Schedule checking only an invocation of action (pressing the button) is required, we made a decision that there is no variable on any step for this test case.

## Variables options for each step

There is no variable.

## Test Case Matrix

|  |  |  |  |
| --- | --- | --- | --- |
| Step | Variable or selection | TC-9 | TC-10 |
| A1 | - | - | - |
| S2 | - | - | - |

# Use case interaction matrix

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Update reservation record** | **Create a new reservation** | **Delete a booking record** | **Checking a schedule** |
| Update reservation record |  |  |  | RC-reservation record |
| Create a new reservation | RC-reservation record |  | RC-reservation record | RC-reservation record |
| Delete a booking record | RD-reservation record |  |  | RD-reservation record |
| Checking a schedule |  |  |  |  |

# Interaction Test Cases

## Use Case Testing

|  |  |  |  |
| --- | --- | --- | --- |
| **Test case name:** | Update reservation record which does not exist | | |
| **Test ID:** | **ITC-1** | | |
| **Test suite:** | Update reservation record | | |
| **Priority:** |  | | |
| **Setup:** | 1. User is logged in. 2. User has started a Bot. 3. User has no reservations. | | |
| **Teardown:** | If the system behaves as expected, nothing has to be done. | | |
| **Step:** | **Description** | **Result** | **Problem ID** |
| ITC-1-A1 | The user presses “Update or Remove an existing reservation” button. | Passed |  |
| ITC-1-S2 | The system responds with the standard buttons since there are no appointments. | Passed |  |
| **Status:** | **Passed** | | |
| **Tester:** | **All step is passed without any bug.** | | |
| **Date complete:** | **25 November 2019** | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Test case name:** | Update reservation record which was deleted before | | |
| **Test ID:** | **ITC-2** | | |
| **Test suite:** | Update reservation record | | |
| **Priority:** |  | | |
| **Setup:** | 1. User is logged in. 2. User has started a Bot. 3. User deleted all reservations. | | |
| **Teardown:** | If the system behaves as expected, nothing has to be done. | | |
| **Step:** | **Description** | **Result** | **Problem ID** |
|  |  |  |  |
| ITC-2-A1 | The user presses “Update or Remove an existing reservation” button. | Passed |  |
| ITC-2-S2 | The system responds with the standard buttons since there are no appointments. | Passed |  |
| **Status:** | **Passed** | | |
| **Tester:** | **All step is passed without any bug.** | | |
| **Date complete:** | **25 November 2019** | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Test case name:** | **Delete a booking record** which does not exist | | |
| **Test ID:** | **ITC-3** | | |
| **Test suite:** | **Delete a booking record** | | |
| **Priority:** |  | | |
| **Setup:** | 1. The User is logged in. 2. The User has started the Bot. 3. User has no reservations. | | |
| **Teardown:** | If the system behaves as expected, nothing has to be done. | | |
| **Step:** | **Description** | **Result** | **Problem ID** |
| ITC-3-A1 | The User presses the “Update or Remove an existing reservation” button | Passed |  |
| ITC-3-S2 | The system responds with the standard buttons since there are no appointments. | Warning | W-2 |
| **Status:** | **Passed** | | |
| **Tester:** |  | | |
| **Date complete:** | **25 November 2019** | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Test case name:** | **Checking schedule without creation of reservations prior** | | |
| **Test ID:** | **ITC-4** | | |
| **Test suite:** | Checking a schedule | | |
| **Priority:** |  | | |
| **Setup:** | 1. User is logged in. 2. User has started a Bot. 3. User has not created any reservation. | | |
| **Teardown:** |  | | |
| **Step:** | **Description** | **Result** | **Problem ID** |
| ITC-4-A1 | The user presses the “My Schedule” button. | Passed |  |
| ITC-4-S2 | The system responds with the standard buttons since there are no appointments. | Passed |  |
| **Status:** | **Passed** | | |
| **Tester:** | **All step is passed without any bug.** | | |
| **Date complete:** | **25 November 2019** | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Test case name:** | **Checking schedule with deletion of all reservations prior** | | |
| **Test ID:** | **ITC-5** | | |
| **Test suite:** | Checking a schedule | | |
| **Priority:** |  | | |
| **Setup:** | 1. User is logged in. 2. User has started a Bot. 3. User deleted all reservations. | | |
| **Teardown:** |  | | |
| **Step:** | **Description** |  |  |
| ITC-5-A1 | The user presses the “My Schedule” button. |  |  |
| ITC-5-S2 | The system responds with the standard buttons since there are no appointments. |  |  |
| **Status:** | **Passed** | | |
| **Tester:** | **All step is passed without any bug.** | | |
| **Date complete:** | **25 November 2019** | | |

## 

|  |  |  |  |
| --- | --- | --- | --- |
| **Test case name:** | **Checking the schedule after updation of reservations** | | |
| **Test ID:** | **ITC-6** | | |
| **Test suite:** | Checking a schedule | | |
| **Priority:** |  | | |
| **Setup:** | 1. User is logged in. 2. User has started a Bot. 3. User updated reservation. | | |
| **Teardown:** |  | | |
| **Step:** | **Description** | **Result** | **Problem ID** |
| ITC-6-A1 | The user presses the “My Schedule” button. | Passed |  |
| ITC-6-S2 | The system shows all reservations at once which belong to the user. | Passed |  |
| **Status:** | **Passed** | | |
| **Tester:** | **All step is passed without any bug.** | | |
| **Date complete:** | **25 November 2019** | | |