TicketWave test suites

Contents

[Test #1 – Event management 1](#_Toc156327432)

[Test #2 – Event Details function 8](#_Toc156327433)

[Test #3 – User interaction 10](#_Toc156327434)

[Test #4 – Scan QR code function 13](#_Toc156327435)

[Test #5 – Create account function 16](#_Toc156327436)

[Test #5.1 – Create account function 18](#_Toc156327437)

[Test #6 – Add and remove from favourites 19](#_Toc156327438)

[Test #7 – Search bar test 22](#_Toc156327439)

[Test #7 – Deletion of expired events 30](#_Toc156327440)

## Test #1 – Event management

Test description

Verification that a user can create an event successfully including attempting to break all validation rules.

Test prerequisites

User account already created.

Test preconditions

|  |  |
| --- | --- |
| **Field Name** | **Test Data** |
| **Username** | TestAccount5 |
| **Password** | !TestAccount5 |
| **Email address** | [Testaccount5@gmail.com](mailto:Testaccount5@gmail.com) |
| **Date of Birth** | 10/04/1999 |

|  |  |
| --- | --- |
| **Field Name** | **Test Data** |
| **Name** | The Lumineers |
| **Venue** | The Horse and Head |
| **Ticket price** | 100 |
| **Ticket quantity** | 50 |
| **Genre** | Music |
| **Month** | 01 |
| **Day** | 15 |
| **Address** | 1, the horse and head, oxford |
| **Promotional discount** | 17 |
| **Event description** | Null |
| **Upload image (Y/N)** | Y |

Test steps

* Open the application and log in.
* Navigate to the ‘Add Event’ button.
* Complete event details using test preconditions data, enter details one box at a team then click ‘add event’ to test validation rules until all data entered.
* Click ‘Add Event’ button
* Verify that the event has been added to the database.

Test results

A screenshot of a computer program

Description automatically generated

#### Figure 1: Validation check of event name.

A screenshot of a computer program

Description automatically generated

#### Figure 2: Validation check of event venue.

A screenshot of a computer program

Description automatically generated

#### Figure 3: Validation check of event ticket price.

A screenshot of a computer program

Description automatically generated

#### Figure 4: Validation check of event ticket quantity.

A screenshot of a computer program

Description automatically generated

#### Figure 5: Validation check of event genre.

A screenshot of a computer program

Description automatically generated

#### Figure 6: Validation check of event month.

A screenshot of a computer

Description automatically generated

#### Figure 7: Validation check of event day.

A screenshot of a computer program

Description automatically generated

#### Figure 8: Validation check of event address.

A screenshot of a computer

Description automatically generated

#### Figure 9: Validation check of event description.

A screenshot of a computer

Description automatically generated

#### Figure 10: Validation check of event image.

A screenshot of a computer

Description automatically generated

#### Figure 11: Screenshot of event tile on main page.

A screenshot of a computer

Description automatically generated

#### Figure12: Screenshot of output in VS Code terminal.

## Test #2 – Event Details function

Test description

Ensure that the events details are displayed correctly in the event details page.

Test prerequisites

Existing events in the database and a user account.

Test preconditions

|  |  |
| --- | --- |
| **Field Name** | **Test Data** |
| **Username** | TestAccount5 |
| **Password** | !TestAccount5 |
| **Email address** | [Testaccount5@gmail.com](mailto:Testaccount5@gmail.com) |
| **Date of Birth** | 10/04/1999 |

|  |  |
| --- | --- |
| **Field Name** | **Test Data** |
| **Name** | Muse |
| **Venue** | Wembley Arena |
| **Ticket price** | 50 |
| **Ticket quantity** | 1000 |
| **Genre** | Music |
| **Month** | 08 |
| **Day** | 08 |
| **Address** | Wembley Arena, London |
| **Promotional discount** | - |
| **Event description** | This is a live performance of the band Muse! |
| **Upload image (Y/N)** | N |

Test steps

* Open the application and log in.
* Click on an event tile to display the events details.
* Verify that the events details are being displayed accurately.

Test results

A group of men posing for a picture

Description automatically generated

#### Figure 13: Screenshot of event tile used for test.

A screenshot of a black and purple event

Description automatically generated

#### Figure 14: Screenshot of event details.

A screen shot of a computer

Description automatically generated

#### Figure 15: Screenshot of VS Code output containing events details and event ID.

## Test #3 – User interaction

Test description

Confirm that a user can add event tickets to their account and remove tickets.

Test prerequisites

Existing events in the database and a user account.

Test preconditions

|  |  |
| --- | --- |
| **Field Name** | **Test Data** |
| **Username** | TestAccount5 |
| **Password** | !TestAccount5 |
| **Email address** | [Testaccount5@gmail.com](mailto:Testaccount5@gmail.com) |
| **Date of Birth** | 10/04/1999 |

|  |  |
| --- | --- |
| **Field Name** | **Test Data** |
| **Name** | Jimmy Carr Standup |
| **Venue** | O2 arena |
| **Ticket price** | 500 |
| **Ticket quantity** | 75 |
| **Genre** | Comedy |
| **Month** | 05 |
| **Day** | 17 |
| **Address** | The O2 Arena, London, England |
| **Promotional discount** | 4 |
| **Event description** | This is a stand-up comedy gig, for Jimmy Carr! |
| **Upload image (Y/N)** | Y |

Test steps

* Open the application and log in.
* Click on an event tile to display the events details.
* Click on ‘add to basket’ button.
* Click on ‘Basket’ button.
* Verify that event has been added to the users basket.

Test results

A screenshot of a computer

Description automatically generated

#### Figure 16: Screenshot of event tile used.

A screen shot of a black and white event

Description automatically generated

#### Figure 17: Screenshot of event details.

A screenshot of a computer

Description automatically generated

#### Figure 18: Screenshot of successful adding of ticket to basket.

A screenshot of a computer

Description automatically generated

#### Figure 19: Screenshot of basket.

A screenshot of a computer

Description automatically generated

#### Figure 20: Screenshot of basket.

## Test #4 – Scan QR code function

Test description

Ensure that the system can correctly scan a QR code and display the events details.

Test prerequisites

Existing events in the database and a user account.

Test preconditions

|  |  |
| --- | --- |
| **Field Name** | **Test Data** |
| **Username** | TestAccount5 |
| **Password** | !TestAccount5 |
| **Email address** | [Testaccount5@gmail.com](mailto:Testaccount5@gmail.com) |
| **Date of Birth** | 10/04/1999 |

|  |  |
| --- | --- |
| **Field Name** | **Test Data** |
| **Name** | Jimmy Carr Live |
| **Venue** | Wembley Arena |
| **Ticket price** | 150 |
| **Ticket quantity** | 100 |
| **Genre** | Comedy |
| **Month** | 01 |
| **Day** | 15 |
| **Address** | 1, Wembley Arena, London |
| **Promotional discount** | - |
| **Event description** | This is a live performance of Jimmy Carr! |
| **Upload image (Y/N)** | Y |

Test steps

* Open the application and log in.
* Navigate to the ‘Scan QR code’ button.
* Select a QR code image from the file directory
* Verify that the QR code associated with the event is displaying the correct event information.

Test results

A screenshot of a computer

Description automatically generated

#### Figure 21: Screenshot of QR codes directory.

A screen shot of a black and white screen

Description automatically generated

#### Figure 22: Screenshot of events details.

A black screen with white text

Description automatically generated

#### Figure 23: Screenshot of event ID 1 from QR Code.

A screenshot of a computer

Description automatically generated

#### Figure 24: Screenshot of QR Code directory displaying deleted event QR Code.

A screen shot of a computer error

Description automatically generated

#### Figure 25: Screenshot of error message for no event found.

## Test #5 – Create account function

Test description

Ensure that a new user can create an account successfully.

Test prerequisites

None

Test preconditions

|  |  |
| --- | --- |
| **Field Name** | **Test Data** |
| **Username** | TestAccount5 |
| **Password** | !TestAccount5 |
| **Email address** | [Testaccount5@gmail.com](mailto:Testaccount5@gmail.com) |
| **Date of Birth** | 10/04/1999 |

Test steps

* Open the application and navigate to the ‘Create account’ page.
* Enter user details and select ‘Create account’ button.
* Verify that the users information is stored in the database by attempting to log in.

Test results

A screenshot of a login form

Description automatically generated

#### Figure 26: Screenshot of data entered into create account page.

The test resulted in an error message displayed relating to the password entry field.

A screenshot of a computer screen

Description automatically generated

#### Figure 27: Screenshot of error message relating to password entry.

This is likely due to an oversight in relation to the regular expression placed upon the password entry to not include the ‘!’ character. The regular expression has been adjusted to allow special characters.

A computer code with numbers and symbols

Description automatically generated

#### Figure 28: Screenshot of original regular expression pretesting.

A computer code on a black background

Description automatically generated

#### Figure 29: Screenshot of fixed reg expression post testing.

## Test #5.1 – Create account function

Test description

Ensure that a new user can now be created successfully using special characters.

Test prerequisites

None

Test preconditions

|  |  |
| --- | --- |
| **Field Name** | **Test Data** |
| **Username** | TestAccount5 |
| **Password** | !TestAccount5 |
| **Email address** | [Testaccount5@gmail.com](mailto:Testaccount5@gmail.com) |
| **Date of Birth** | 10/04/1999 |

Test steps

* Open the application and navigate to the ‘Create account’ page.
* Enter user details and select ‘Create account’ button.
* Verify that the users information is stored in the database by attempting to log in.

Test results

A screenshot of a computer

Description automatically generated

#### Figure 30: Screenshot of data entered into create account function.

A screen shot of a computer

Description automatically generated

#### Figure 31: Screenshot of VS Code output.

## Test #6 – Add and remove from favourites

Test description

Ensure that a user can successfully add and remove an event from their favourites

Test prerequisites

None

Test preconditions

|  |  |
| --- | --- |
| **Field Name** | **Test Data** |
| **Username** | TestAccount5 |
| **Password** | !TestAccount5 |
| **Email address** | [Testaccount5@gmail.com](mailto:Testaccount5@gmail.com) |
| **Date of Birth** | 10/04/1999 |

|  |  |
| --- | --- |
| **Field Name** | **Test Data** |
| **Name** | The Lumineers |
| **Venue** | The Horse and Head |
| **Ticket price** | 100 |
| **Ticket quantity** | 50 |
| **Genre** | Music |
| **Month** | 01 |
| **Day** | 10 |
| **Address** | 1, the horse and head, oxford |
| **Promotional discount** | 17 |
| **Event description** | Null |
| **Upload image (Y/N)** | Y |

Test steps

* Open the application and navigate to the ‘Log in’ page.
* Enter user details and select ‘Log In’ button.
* Verify that a user can successfully use the favourites buttons and validation rules are utilised.

Test results

A screenshot of a computer

Description automatically generated

#### Figure 32: Screenshot of event details and add to favourites button.

A screenshot of a computer

Description automatically generated

#### Figure 33: Screenshot of successful adding to favourites.

A screen shot of a black and white event

Description automatically generated

#### Figure 34: Screenshot of event details and remove event from favourites button.

A screenshot of a computer

Description automatically generated

#### Figure 35: Screenshot of successful removal of event from favuorites.

## Test #7 – Search bar test

Test description

Ensure a user can use full and partial match to search for an event.

Test prerequisites

None

Test preconditions

|  |  |
| --- | --- |
| **Field Name** | **Test Data** |
| **Username** | TestAccount5 |
| **Password** | !TestAccount5 |
| **Email address** | [Testaccount5@gmail.com](mailto:Testaccount5@gmail.com) |
| **Date of Birth** | 10/04/1999 |

|  |  |
| --- | --- |
| **Field Name** | **Test Data** |
| **Name** | Jimmy Carr Standup |
| **Venue** | O2 arena |
| **Ticket price** | 500 |
| **Ticket quantity** | 75 |
| **Genre** | Comedy |
| **Month** | 01 |
| **Day** | 11 |
| **Address** | The O2 Arena, London, England |
| **Promotional discount** | 4 |
| **Event description** | This is a stand-up comedy gig, for Jimmy Carr! |
| **Upload image (Y/N)** | Y |

|  |  |
| --- | --- |
| **Field Name** | **Test Data** |
| **Name** | The Lumineers |
| **Venue** | The Horse and Head |
| **Ticket price** | 100 |
| **Ticket quantity** | 50 |
| **Genre** | Music |
| **Month** | 01 |
| **Day** | 10 |
| **Address** | 1, the horse and head, oxford |
| **Promotional discount** | 17 |
| **Event description** | Null |
| **Upload image (Y/N)** | Y |

Test steps

* Open the application and navigate to the ‘Log in’ page.
* Enter user details and select ‘Log In’ button.
* Verify that a user can successfully use the search bar for full and partial match searches.

Test results

A screen shot of a computer

Description automatically generated

#### Figure 36: Screenshot of exact match search.

A screenshot of a computer

Description automatically generated

#### Figure 37: Screenshot of partial match search.

A screenshot of a computer

Description automatically generated

#### Figure 80: Screenshot of no match search.

#### Test #7 – Login validation test

Test description

Ensure a user cannot enter without a username and password match.

Test prerequisites

None

Test preconditions

|  |  |
| --- | --- |
| **Field Name** | **Test Data** |
| **Username** | TestAccount5 |
| **Password** | !TestAccount5 |
| **Email address** | [Testaccount5@gmail.com](mailto:Testaccount5@gmail.com) |
| **Date of Birth** | 10/04/1999 |

Test steps

* Open the application and navigate to the ‘Log in’ page.
* Enter user details incorrectly and select ‘Log In’ button.
* Verify that a user can unsuccessfully log in with incorrect credentials and empty credentials.

Test results

A screenshot of a computer

Description automatically generated

#### Figure 38: Screenshot of error from no password input.

A screenshot of a computer

Description automatically generated

#### Figure 39: Screenshot of error from no username input.

A screenshot of a computer

Description automatically generated

#### Figure 40: Screenshot of error from invalid username.

A screenshot of a login screen

Description automatically generated

#### Figure 41: Screenshot of error from invalid password.

## Test #7 – Deletion of expired events

Test description

Ensure an event is deleted if the user selects ‘Delete Event’

Test prerequisites

None

Test preconditions

|  |  |
| --- | --- |
| **Field Name** | **Test Data** |
| **Username** | TestAccount5 |
| **Password** | !TestAccount5 |
| **Email address** | [Testaccount5@gmail.com](mailto:Testaccount5@gmail.com) |
| **Date of Birth** | 10/04/1999 |

|  |  |
| --- | --- |
| **Field Name** | **Test Data** |
| **Name** | Tom Misch |
| **Venue** | Wembley Arena |
| **Ticket price** | 150 |
| **Ticket quantity** | 1000 |
| **Genre** | Music |
| **Month** | 09 |
| **Day** | 12 |
| **Address** | 1, The Wembley Arena |
| **Promotional discount** | 0 |
| **Event description** | This is a live performance of Tom Misch! |
| **Upload image (Y/N)** | Y |

Test steps

* Open the application and navigate to the ‘Log in’ page.
* Select an event tile the user has created
* Verify that an event is deleted.
* Reload application to ensure it has been deleted.

Test results

A screenshot of a video

Description automatically generated

#### Figure 42: Screenshot of event on main page.

A screen shot of a computer

Description automatically generated

#### Figure 43: Screenshot of event that has been deleted.

A screenshot of a computer

Description automatically generated

#### Figure 44: Event no longer visible on main page due to deletion