**Royal Melbourne Institute of Technology**

COSC 2299 - Software Engineering Process & Tools

**Online Booking System**

**Acceptance Tests**

Version 1.0

**Members:**

SangYeon Lee - s3742751

Mitchell Gust - s3782095

David Brown - s3785994

**Template**

|  |  |
| --- | --- |
| **ID** | 123 |
| **Purpose** | Test the Register User with existing username |
| **Setup** | A user called Icavedon already exists in the system |
| **Steps** | 1. Navigate to Register new User page  2. Select Register New User  3. Enter Icavedon as username, other values for other fields |
| **Expected Result** | System reports that username already exists, New User not added |

**Anh is writing below:**

|  |  |
| --- | --- |
| **ID** | 1 |
| **Purpose** | Test ‘Get Booking History’ with an existing user with prior booking history |
| **Setup** | A user called ‘user’ already exists in the system |
| **Steps** | 1. Login to the system with the username ‘user’  2. Get redirected to the Customer Dashboard  3. Select ‘View History’ |
| **Expected Result** | The system displays a list of prior bookings with their respective statuses. |

|  |  |
| --- | --- |
| **ID** | 2 |
| **Purpose** | Test ‘Get Booking History’ with an existing user with no prior booking history |
| **Setup** | A user called ‘user’ already exists in the system |
| **Steps** | 1. Login to the system with the username ‘user’  2. Get redirected to the Customer Dashboard  3. Select ‘View History’ |
| **Expected Result** | System reports that the user has no prior booking |

|  |  |
| --- | --- |
| **ID** | 3 |
| **Purpose** | Test ‘View A Worker Calendar’ with an existing worker who has upcoming appointments |
| **Setup** | A user called ‘worker’ already exists in the system |
| **Steps** | 1. Login to the system with the username ‘worker’  2. Get redirected to the Worker Dashboard  3. Select ‘View Calendar’ |
| **Expected Result** | System displays a calendar filled with upcoming appointment for the worker |

|  |  |
| --- | --- |
| **ID** | 4 |
| **Purpose** | Test ‘View A Worker Calendar’ with an existing worker who has no upcoming appointments |
| **Setup** | A user called ‘worker’ already exists in the system |
| **Steps** | 1. Login to the system with the username ‘worker’  2. Get redirected to the Worker Dashboard  3. Select ‘View Calendar’ |
| **Expected Result** | System displays an empty calendar |

|  |  |
| --- | --- |
| **ID** | 5 |
| **Purpose** | Test ‘Edit Worker Profile’ with an existing worker |
| **Setup** | A user called ‘worker’ already exists in the system |
| **Steps** | 1. Login to the system with the username ‘worker’  2. Get redirected to the Worker Dashboard  3. Select ‘View Profile’  4. Click ‘Edit’  5. Enter fields with appropriate information |
| **Expected Result** | System displays a ‘success’ message and save the new information to the database |

|  |  |
| --- | --- |
| **ID** | 6 |
| **Purpose** | Test ‘Edit Worker Profile’ with an existing worker |
| **Setup** | A user called ‘worker’ already exists in the system |
| **Steps** | 1. Login to the system with the username ‘worker’  2. Get redirected to the Worker Dashboard  3. Select ‘View Profile’  4. Click ‘Edit’  5. Enter fields with inappropriate information |
| **Expected Result** | System displays a ‘failed’ message and the reasons |

|  |  |
| --- | --- |
| **ID** | 7 |
| **Purpose** | Test ‘View Worker Profile’ with an existing worker |
| **Setup** | A user called ‘worker’ already exists in the system |
| **Steps** | 1. Login to the system with the username ‘worker’  2. Get redirected to the Worker Dashboard  3. Select ‘View Profile’ |
| **Expected Result** | System displays the worker’s profile |

**David (Requirements 3 & 4)**

|  |  |
| --- | --- |
| **ID** | 8 |
| **Purpose** | Test blank input for employee creation |
| **Set Up** | Existing admin user ‘admin’ in system. |
| **Steps** | 1. Log into admin user account ‘admin’  2. Navigate to staff page  3. Press on ‘add’ button  4. Press ‘create employee’ |
| **Expected Results** | System displays error message to user. |

|  |  |
| --- | --- |
| **ID** | 9 |
| **Purpose** | Test existing username for employee creation |
| **Set Up** | Existing admin user ‘admin’ in system. |
| **Steps** | 1. Log into admin user account ‘admin’  2. Navigate to staff page  3. Press on ‘add’ button  4. Input existing username  5. Input random (valid) inputs for rest  6. Press ‘create employee’ |
| **Expected Results** | System displays error message to user. |

|  |  |
| --- | --- |
| **ID** | 10 |
| **Purpose** | Test all valid inputs for employee creation |
| **Set Up** | Existing admin user ‘admin’ in system. |
| **Steps** | 1. Log into admin user account ‘admin’  2. Navigate to staff page  3. Press on ‘add’ button  4. Input valid (non existing) username  5. Input random (valid) inputs for rest  6. Press ‘create employee’ |
| **Expected Results** | System displays confirmation message to user. System creates user. |

|  |  |
| --- | --- |
| **ID** | 11 |
| **Purpose** | Test invalid username for employee edit |
| **Set Up** | Existing admin user ‘admin’ in system. |
| **Steps** | 1. Log into admin user account ‘admin’  2. Navigate to staff page  3. Press on ‘edit button next to employee  4. Input new invalid (existing) username  5. Input random (valid) inputs for rest  6. Press ‘update employee’ |
| **Expected Results** | System displays error message to user. |

|  |  |
| --- | --- |
| **ID** | 12 |
| **Purpose** | Test all valid inputs for employee edit |
| **Set Up** | Existing admin user ‘admin’ in system. |
| **Steps** | 1. Log into admin user account ‘admin’  2. Navigate to staff page  3. Press on ‘edit’ button next to employee  4. Input new valid inputs for all fields  5. Press ‘update employee’ |
| **Expected Results** | System displays confirmation message to user. System updates user. |

|  |  |
| --- | --- |
| **ID** | 13 |
| **Purpose** | Test invalid hours input for add shift |
| **Set Up** | Existing admin user ‘admin’ in system. |
| **Steps** | 1. Log into admin user account ‘admin’  2. Navigate to staff page  3. Press on ‘allocate’ button  4. Input invalid shift data into fields  5. Press ‘update hours’ |
| **Expected Results** | System displays error message to user. |

|  |  |
| --- | --- |
| **ID** | 14 |
| **Purpose** | Test valid input for add shift |
| **Set Up** | Existing admin user ‘admin’ in system. |
| **Steps** | 1. Log into admin user account ‘admin’  2. Navigate to staff page  3. Press on ‘allocate’ button  4. Input valid shift data into fields  5. Press ‘update hours’ |
| **Expected Results** | System displays confirmation message to user. System updates hours. |

|  |  |
| --- | --- |
| **ID** | 15 |
| **Purpose** | Test invalid hours input for edit shift |
| **Set Up** | Existing admin user ‘admin’ in system. |
| **Steps** | 1. Log into admin user account ‘admin’  2. Navigate to staff page  3. Press on ‘edit hours’ button  4. Input invalid edited shift data into fields  5. Press ‘update hours’ |
| **Expected Results** | System displays error message to user. |

|  |  |
| --- | --- |
| **ID** | 16 |
| **Purpose** | Test valid hours input for edit shift |
| **Set Up** | Existing admin user ‘admin’ in system. |
| **Steps** | 1. Log into admin user account ‘admin’  2. Navigate to staff page  3. Press on ‘edit hours’ button  4. Input valid edited shift data into fields  5. Press ‘update hours’ |
| **Expected Results** | System displays confirmation message to user. System updates hours. |

|  |  |
| --- | --- |
| **ID** | 17 |
| **Purpose** | Test viewing past bookings with no past bookings |
| **Set Up** | Existing admin user ‘admin’ in system. |
| **Steps** | 1. Log into admin user account ‘admin’  2. Navigate to booking page  3. Press on ‘past’ button |
| **Expected Results** | System displays no past bookings. |

|  |  |
| --- | --- |
| **ID** | 18 |
| **Purpose** | Test viewing past bookings with existing past bookings |
| **Set Up** | Existing admin user ‘admin’ in system. |
| **Steps** | 1. Log into admin user account ‘admin’  2. Navigate to booking page  3. Press on ‘past’ button |
| **Expected Results** | System displays all past bookings. |

|  |  |
| --- | --- |
| **ID** | 19 |
| **Purpose** | Test viewing upcoming bookings with no upcoming bookings |
| **Set Up** | Existing admin user ‘admin’ in system. |
| **Steps** | 1. Log into admin user account ‘admin’  2. Navigate to booking page  3. Press on ‘upcoming’’ button |
| **Expected Results** | System displays no upcoming bookings. |

|  |  |
| --- | --- |
| **ID** | 20 |
| **Purpose** | Test viewing upcoming bookings with no upcoming bookings |
| **Set Up** | Existing admin user ‘admin’ in system. |
| **Steps** | 1. Log into admin user account ‘admin’  2. Navigate to booking page  3. Press on ‘upcoming’’ button |
| **Expected Results** | System displays all upcoming bookings. |

|  |  |
| --- | --- |
| **ID** | 21 |
| **Purpose** | Test viewing worker availability with no available workers |
| **Set Up** | Existing admin user ‘admin’ in system. |
| **Steps** | 1. Log into admin user account ‘admin’  2. Navigate to worker page  3. Press on ‘availability’ button |
| **Expected Results** | System displays error message. |

|  |  |
| --- | --- |
| **ID** | 22 |
| **Purpose** | Test viewing worker availability with available workers |
| **Set Up** | Existing admin user ‘admin’ in system. |
| **Steps** | 1. Log into admin user account ‘admin’  2. Navigate to worker page  3. Press on ‘availability’ button |
| **Expected Results** | System displays availabilities for individual workers. |

|  |  |
| --- | --- |
| **ID** | 23 |
| **Purpose** | Test viewing service availability with no available services |
| **Set Up** | Existing admin user ‘admin’ in system. |
| **Steps** | 1. Log into admin user account ‘admin’  2. Navigate to service page  3. Press on ‘availability’ button |
| **Expected Results** | System displays error message. |

|  |  |
| --- | --- |
| **ID** | 24 |
| **Purpose** | Test viewing service availability with available services |
| **Set Up** | Existing admin user ‘admin’ in system. |
| **Steps** | 1. Log into admin user account ‘admin’  2. Navigate to service page  3. Press on ‘availability’ button |
| **Expected Results** | System displays availabilities for individual services. |

|  |  |
| --- | --- |
| **ID** | 25 |
| **Purpose** | Test attempting booking with no available times |
| **Set Up** | Existing customer user ‘user’ in system. |
| **Steps** | 1. Log into customer user account ‘user’  2. Navigate to booking page  3. Press on “make booking” button  4. Input invalid time data into fields  5. Press ‘booking’ |
| **Expected Results** | System displays no time error message to user. |

|  |  |
| --- | --- |
| **ID** | 26 |
| **Purpose** | Test attempting booking with unavailable services |
| **Set Up** | Existing customer user ‘user’ in system. |
| **Steps** | 1. Log into customer user account ‘user’  2. Navigate to booking page  3. Press on “make booking” button  4. Input invalid service data into fields  5. Press ‘booking’ |
| **Expected Results** | System displays no service error message to user. |

|  |  |
| --- | --- |
| **ID** | 27 |
| **Purpose** | Test attempting booking with unavailable services |
| **Set Up** | Existing customer user ‘user’ in system. |
| **Steps** | 1. Log into customer user account ‘user’  2. Navigate to booking page  3. Press on “make booking” button  4. Input valid data into fields  5. Press ‘booking’ |
| **Expected Results** | System displays confirmation message, creates booking. |

**Mitchell Gust (Acceptance Tests for Requirements 1 & 2)**

|  |  |
| --- | --- |
| **ID** | 28 |
| **Purpose** | Test the Registered User can access the home page |
| **Setup** | A user called registeredUser already exists in the system and is logged in |
| **Steps** | 1. Enter home page URL |
| **Expected Result** | Home page is shown |

|  |  |
| --- | --- |
| **ID** | 29 |
| **Purpose** | Test the Unregistered User can access the home page |
| **Setup** | No user is currently logged in. |
| **Steps** | 1. Enter home page URL |
| **Expected Result** | Home page is shown |

**Test Access to Admin Dashboard from all User Types**

|  |  |
| --- | --- |
| **ID** | 30 |
| **Purpose** | Test an Admin with Admin permissions can access the Admin Dashboard |
| **Setup** | A User called registeredAdmin exists in the system and is registered as an Admin |
| **Steps** | 1. Admin logs in  2. Navigates to the URL of the Admin Dashboard |
| **Expected Result** | The Admin Dashboard is shown |

|  |  |
| --- | --- |
| **ID** | 31 |
| **Purpose** | Test the Admin is redirected to the Admin Dashboard after login |
| **Setup** | A User called registeredAdmin exists in the system and is registered as an Admin |
| **Steps** | 1. Enter website URL  2. Navigate to login screen  3. Registered Admin logs in |
| **Expected Result** | The Admin Dashboard is shown |

|  |  |
| --- | --- |
| **ID** | 32 |
| **Purpose** | Test a Customer without Admin permissions can not access the Admin Dashboard |
| **Setup** | A User called registeredCustomer exists in the system and is not registered as an Admin |
| **Steps** | 1. Customer logs in  2. Navigates to the URL of the Admin Dashboard |
| **Expected Result** | The system reports that the Customer is unauthorised to visit the Admin Dashboard |

|  |  |
| --- | --- |
| **ID** | 33 |
| **Purpose** | Test a Worker without Admin permissions can not access the Admin Dashboard |
| **Setup** | A User called registeredWorker exists in the system and is not registered as an Admin |
| **Steps** | 1. Worker logs in  2. Navigates to the URL of the Admin Dashboard |
| **Expected Result** | The system reports that the Worker is unauthorised to visit the Admin Dashboard |

**Test Access to Worker Dashboard from all User Types**

|  |  |
| --- | --- |
| **ID** | 34 |
| **Purpose** | Test a Worker with Worker permissions can access the Worker Dashboard |
| **Setup** | A User called registeredWorker exists in the system and is registered as a Worker. |
| **Steps** | 1. Customer logs in  2. Navigates to the URL of the Worker Dashboard |
| **Expected Result** | The Worker Dashboard is shown |

|  |  |
| --- | --- |
| **ID** | 35 |
| **Purpose** | Test the Worker is redirected to the Worker Dashboard after login |
| **Setup** | A User called registeredWorker exists in the system and is registered as a Worker. |
| **Steps** | 1. Enter website URL  2. Navigate to login screen  3. Registered Worker logs in |
| **Expected Result** | The Worker Dashboard is shown |

|  |  |
| --- | --- |
| **ID** | 36 |
| **Purpose** | Test a Customer without Worker permissions can not access the Worker Dashboard |
| **Setup** | A User called registeredCustomer exists in the system and is not registered as an Worker |
| **Steps** | 1. Customer logs in  2. Navigates to the URL of the Worker Dashboard |
| **Expected Result** | The system reports that the Customer is unauthorised to visit the Worker Dashboard |

|  |  |
| --- | --- |
| **ID** | 37 |
| **Purpose** | Test an Admin without Worker permissions can not access the Worker Dashboard |
| **Setup** | A User called registeredAdmin exists in the system and is not registered as an Worker |
| **Steps** | 1. Admin logs in  2. Navigates to the URL of the Worker Dashboard |
| **Expected Result** | The system reports that the Admin is unauthorised to visit the Worker Dashboard |

**Test Access to Customer Dashboard from all User Types**

|  |  |
| --- | --- |
| **ID** | 38 |
| **Purpose** | Test the Customer can access the Customer Dashboard |
| **Setup** | A User called registeredCustomer exists in the system and is registered as a Customer. |
| **Steps** | 1. Worker logs in  2. Navigates to the URL of the Customer Dashboard |
| **Expected Result** | The Customer Dashboard is shown |

|  |  |
| --- | --- |
| **ID** | 39 |
| **Purpose** | Test the Customer is redirected to the Customer Dashboard after login |
| **Setup** | A User called registeredCustomer exists in the system and is registered as a Customer. |
| **Steps** | 1. Enter website URL  2. Navigate to login screen  3. Registered Customer logs in |
| **Expected Result** | The Customer Dashboard is shown |

|  |  |
| --- | --- |
| **ID** | 40 |
| **Purpose** | Test a Worker without Customer permissions can not access the Customer Dashboard |
| **Setup** | A User called registeredWorker exists in the system and is not registered as a Customer |
| **Steps** | 1. Worker logs in  2. Navigates to the URL of the Customer Dashboard |
| **Expected Result** | The system reports that the Worker is unauthorised to visit the Customer Dashboard |

|  |  |
| --- | --- |
| **ID** | 41 |
| **Purpose** | Test an Admin without Customer permissions can not access the Customer Dashboard |
| **Setup** | A User called registeredAdmin exists in the system and is not registered as an Customer |
| **Steps** | 1. Admin logs in  2. Navigates to the URL of the Customer Dashboard |
| **Expected Result** | The system reports that the Admin is unauthorised to visit the Customer Dashboard |

**Log in / Register**

|  |  |
| --- | --- |
| **ID** | 42 |
| **Purpose** | Test User with invalid credentials can not log in |
| **Setup** |  |
| **Steps** | 1. Enter website URL  2. Navigate to login screen  3. User enters invalid credentials and clicks log in |
| **Expected Result** | The system reports that the user credentials are invalid and prompts the user to try again. |

|  |  |
| --- | --- |
| **ID** | 43 |
| **Purpose** | Test User can not Register with sign up fields left blank |
| **Setup** |  |
| **Steps** | 1. Enter website URL  2. Navigate to signup screen  3. Enters signup details, leaving one or more form inputs blank |
| **Expected Result** | The system reports that one or more details are left blank, and must be amended to sign up. |

|  |  |
| --- | --- |
| **ID** | 44 |
| **Purpose** | Test that a user can not register with an invalid email |
| **Setup** |  |
| **Steps** | 1. Enter website URL  2. Navigate to signup screen  3. Enters signup details, with an invalid email |
| **Expected Result** | The system reports that the email inputted is invalid, and must be amended to sign up. |

|  |  |
| --- | --- |
| **ID** | 45 |
| **Purpose** | Test that a user can not register with an invalid phone number |
| **Setup** |  |
| **Steps** | 1. Enter website URL  2. Navigate to signup screen  3. Enters signup details, with an invalid phone number |
| **Expected Result** | The system reports that the phone number inputted is invalid, and must be amended to sign up. |

|  |  |
| --- | --- |
| **ID** | 46 |
| **Purpose** | Test that a user can not register with an invalid name |
| **Setup** |  |
| **Steps** | 1. Enter website URL  2. Navigate to signup screen  3. Enters signup details, with an invalid name (includes non-alphabetical characters) |
| **Expected Result** | The system reports that the name inputted is invalid, and must be amended to sign up. |

|  |  |
| --- | --- |
| **ID** | 47 |
| **Purpose** | Test that a user can not register with a duplicate username |
| **Setup** | A User with the name “paulSmith” already exists in the system |
| **Steps** | 1. Enter website URL  2. Navigate to signup screen  3. Enters signup details, with the username “paulSmith” |
| **Expected Result** | The system reports that the username is already in use, and must be amended to sign up. |

|  |  |
| --- | --- |
| **ID** | 48 |
| **Purpose** | Test that a user can register with valid details |
| **Setup** |  |
| **Steps** | 1. Enter website URL  2. Navigate to signup screen  3. Enters signup details |
| **Expected Result** | The user is registered as a Customer, the Customer Dashboard is shown. |

**-LEE**

|  |  |
| --- | --- |
| **ID** | 49 |
| **Purpose** | User can not change user name to the name that has been already taken. |
| **Setup** | User want to change the username to “besamly2018”.  Username “besamly2018” already exists in the database. |
| **Steps** | 1. Find the user name field.  2. Click the edit button next to it.  3. Type username “besamly2018”. |
| **Expected Result** | The system should show error message “username already exists” |

|  |  |
| --- | --- |
| **ID** | 50 |
| **Purpose** | Users can not edit passwords without meeting password policy. |
| **Setup** | User wants to change the password to something that is the same as the current password or something that is not a combination of capital and small letter with number and characters. |
| **Steps** | 1. Go to profile page  2. Find password field  3. Click edit  4. Type invalid password “asd123” |
| **Expected Result** | The system should show error message “Password should be a combination of capital and small letter with number and characters.” |

|  |  |
| --- | --- |
| **ID** | 51 |
| **Purpose** | Customer can not cancel booking less than 24 before the event |
| **Setup** | Customers have an upcoming booking within 24hours. |
| **Steps** | 1. Go to the booking history page.  2. Find the event you wish to cancel.  3. Click cancel button |
| **Expected Result** | Customer should see error message “Booking can not be canceled less than 24 hours before the event” |