Backend Tests

Test No:	API:	Class	Test Name:	Function:	Expected	Result:	Test Date:	Ran by:
1	Roster	Controller	givenRoster_whenGetRosterById_thenReturnRoster	Testing that the roster controller correctly returns a collection of roster objects with the corresponding IDs when a get request is sent	We see a collection of roster objects with the correct ID	Passed	16/9/20	Julian
2		Controller	givenRoster_whenGetAllRosters_thenReturnRosterList	Testing that the roster controller when all of the roster objects are returned when a get request is sent	We see a collection of all of the roster objects in the system	Passed	16/9/20	Julian
3		Controller	givenRoster_whenDeleteRosterById_thenSuccessfulDeleteRequest	Testing that the roster controller when a delete request correctly removes the corresponding Roster object	We see the roster object removed from our roster table	Passed	16/9/20	Julian
4		Roster	whenInsertRosterWithNoTime_thenReturnFalse	Testing that a roster object will be rejected if the time field is empty	We see a rejection message when we attempt to initialise a roster with a null time field	Passed	16/9/20	Julian
5		Roster	whenInsertRosterWithNoDate_thenReturnFalse	Testing that a roster object will be rejected if the date field is empty	We see a rejection message when we attempt to initialise a roster with a null date field	Passed	16/9/20	Julian
6		Roster	whenInsertRosterWithNonExistingEmployee_thenReturnFalse	Testing that a roster object will be rejected if the employee doesn't exist	We see a rejection message when we attempt to initialise a roster with a non-persistant employee	Passed	16/9/20	Julian
7		Roster	whenInsertRosterWithExistingEmployee_thenReturnTrue	Testing that a roster object will be accepted if the employee exists in the database	We see no rejection message	Passed	16/9/20	Julian
8		Repository	findAllRostersForTheEmployee_returnListRoster	Testing that all roster objects are returned that correspond to the given employee	We see a list of rosters that have the given employee as their employee field	Passed	16/9/20	Julian
9		Repository	findAllRostersForNonExistingEmployee_returnNull	Testing that an empty list is returned when the given employee has no roster objects	We see an empty list returned	Passed	16/9/20	Julian
10		Repository	getAllRosters_returnListOfRosters	Testing that all roster objects in the system are returned when we send a get request	We see a list of all rosters in the system	Passed	16/9/20	Julian
11		Repository	findAllRostersOnTable_thenAssertTrue	Testing that all roster objects in the system are returned when we send a get request	We see a list of all rosters in the system	Passed	16/9/20	Julian
12		Service	whenGettingRosterWithValidIdentifier_thenRosterShouldBeFound	Testing that a list of roster objects is returned when a roster object exists with the given employee	We see a list of roster objects with the given employee	Passed	16/9/20	Julian
13		Service	whenInvalidIdentifier_ThenThrowRosterException	Testing that an empty list is returned when the given employee has no roster objects	We see an empty list returned	Passed	16/9/20	Julian
14		Service	getAllRoster_thenReturnListOfRosters	Testing that a list of roster objects is returned when a get request is sent	We see a list containing all roster objects in the system	Passed	16/9/20	Julian
15		Service	whenDeletingRosterWithValidIdentifier_thenSuccessfulDeleteRequest	Testing that a roster object is deleted when a delete request is sent	We see the roster object removed from our roster table	Passed	16/9/20	Julian
16	Customer	Controller	givenCustomers_whenGetCustomers_thenReturnJsonArray	Testing that sending a get request returns an array containing all customers in the system	We see a list of customer objects	Passed	16/9/20	Jack
17		Controller	givenNoCustomers_whenGetCustomers_thenReturnNullArray	Testing that an empty customer table returns an empty array	We see an empty list returned	Passed	16/9/20	Jack
18		Controller	givenNewCustomer_whenPostNewCustomer_thenNewCustomerIsAdded	Testing that sending a post request adds a new customer to the database	We see a new customer added to the database	Passed	16/9/20	Jack
19		Controller	givenCustomerExists_whenDeleteCustomer_thenCustomerIsDeleted	Testing that sending a delete request removes the relevant customer	We see the customer removed from our customer table	Passed	16/9/20	Jack
20		Customer	whenInsertCustomerWithBlankUsername_thenReturnFalse	Testing that attempting to input a customer with a blank username is rejected	We see a rejection message returned	Passed	16/9/20	Jack
21		Customer	whenInsertCustomerWithValidFirstname_thenReturnTrue	Testing that inputting a customer with a non-blank username is accepted	We see a customer created	Passed	16/9/20	Jack
22		Customer	whenInsertCustomerWithBlankLastname_thenReturnFalse	Testing that attempting to input a customer with a blank lastname is rejected	We see a rejection message	Passed	16/9/20	Jack
23		Customer	whenInsertCustomerWithValidLastname_thenReturnTrue	Testing that inputting a customer with a non-blank last name is accepted	We see a customer successfully created	Passed	16/9/20	Jack
24		Customer	whenInsertUsernameWithLessThan3Characters_thenReturnFalse	Testing that attempting to input a customer with a <3 character username is rejected	We see a rejection message	Passed	16/9/20	Jack
25		Customer	whenInsertUsernameWithGreaterThan20Characters_thenReturnFalse	Testing that attempting to input a customer with a >20 character username is rejected	We see a rejection message	Passed	16/9/20	Jack
26		Customer	whenInsertUsernameBetween3and20Characters_thenReturnTrue	Testing that attempting to input a customer with a 3 < usernameLength <20 character username is accepted	We see a customer successfully created	Passed	16/9/20	Jack
27		Customer	whenInsertBlankEmail_thenReturnFalse	Testing that attempting to input a customer with a blank email is rejected	We see a rejection message	Passed	16/9/20	Jack
28		Customer	whenInsertBlankAddress_thenReturnFalse	Testing that attempting to input a customer with a blank address is rejected	We see a rejection message	Passed	16/9/20	Jack
29		Customer	whenInsertBlankPassword_thenReturnFalse	Testing that attempting to input a customer with a blank password is rejected	We see a rejection message	Passed	16/9/20	Jack
30		Repository	whenCustomerExistsWithUsername_thenReturnCustomer	Testing that when we send a get request containing the username corresponding to a given customer, that the customer object is returned	We see a customer object returned with the given username	Passed	16/9/20	Jack
31		Repository	whenCustomerDoesNotExistWithUsername_thenReturnNull	Testing that when we send a get request containing the username not corresponding to a given customer, that no object is returned	We see a blank object returned	Passed	16/9/20	Jack
32		Repository	whenCustomerExistsWithFirstname_thenReturnCustomer	Testing that when we send a get request containing the first not corresponding to a given customer, that no object is returned	We see a customer object returned with the given first name	Passed	16/9/20	Jack
33		Service	whenValidUsernameIsUsed_CustomerShouldBeReturned	Testing that when we send a post request with all valid inputs, a new customer is created	We see a new customer added to the database	Passed	16/9/20	Jack
34		Service	whenValidEmailIsUsed_CustomerShouldBeReturned	Testing that when we send a post request with all valid inputs, a new customer is created	We see a new customer added to the database	Passed	16/9/20	Jack
35		Service	whenInvalidEmailIsUsed_NullIsReturned	Testing that when we send a post request with an invalid email input, no customer is created	We see no customer added to the database	Passed	16/9/20	Jack
36		Service	whenGetAllIsUsed_ReturnListOfCustomers	Testing that when we send a customer get request, all customers returned	We see a list containing all customers returned	Passed	16/9/20	Jack
37	Employee	Controller	givenEmployees_whenGetEmployees_thenReturnJsonArray	Testing that we send a get request to the employee database, list of all employees is correctly returned	We see a list containing all employee objects	Passed	16/9/20	Theo
38		Controller	givenEmployeesId_whenGetEmployeesById_thenReturnEmployee	Testing that we send a get request containing the employeeID, then we return the relevant employee	We see an employee object returned	Passed	16/9/20	Theo
39		Controller	givenEmployeeId_whenDeleteEmployeeById_thenReturnDelete	Testing that when a delete request is sent, the employee is deleted from the database and a valid message is returned	We see the emploee deleted from the database	Passed	16/9/20	Theo
40		Controller	givenEmployee_whenAddingNewEmployee_thenSuccessfulPostRequest	Testing that when a post request is sent, a new employee is created in the database	We see a new employee added to the database	Passed	16/9/20	Theo
41		Controller	getAllEmployeeWithUserNameAndPassword_thenSuccessfulGetRequest	Testing that when we send a get request containing username and password, then employee is returned	We see the relevant employee returned	Passed	16/9/20	Theo
42		Repository	findEmployeeWithEmail_ReturnEmployee	Testing that when we find an employee by their email address, the relevant employee is returned	We are returned the relevant employee	Passed	16/9/20	Theo
43		Repository	findEmployeeWithIncorrectEmail_thenReturnNull	Testing that when we find employees by a non matching email, a null character is returned	We are returned a null object	Passed	16/9/20	Theo
44		Repository	findEmployeeWithPhoneNumber	Testing that by their phone number, the relevant employee is returned	We are returned the relevant employee	Passed	16/9/20	Theo
45		Repository	findEmployeeWithIncorrectPhoneNumber_thenReturnNull	Testing that we find an employee by an invalid phone number, a null object is returned	We are returned a null object	Passed	16/9/20	Theo
46		Repository	findEmployeeWithUserName_thenReturnEmployee	Testing that we find an employee by their username, the relevant employee is returned	We are returned the relevant employee	Passed	16/9/20	Theo
47		Repository	findEmployeeWithIncorrectUserName_thenReturnNull	Testing that we find an employee by an invalid username, a null object is returned	We are returned a null object	Passed	16/9/20	Theo
48		Repository	getAllEmployees_thenReturnListOfEmployees	Testing that when we find all employees, then a list containing all employees in the database are returned	We are returned a list of all employee objects	Passed	16/9/20	Theo
49		Repository	getEmployeeByValidIdentifier_thenReturnEmployee	Testing that when we find an employee by their identifier, the relevant employee is returned	We are returned the relevant employee	Passed	16/9/20	Theo
50		Repository	getEmployeeWithInCorrectIdentifier_theReturnNull	Testing that when we find an employee by an invalid identifier, a null object is returned	We are returned a null object	Passed	16/9/20	Theo
51		Repository	getEmployeeByUsernameAndPassword_returnEmployee	Testing that when we find an employee by their username and password, the relevant employee is returned	We are returned the relevant employee	Passed	16/9/20	Theo
52		Repository	getEmployeeByInvalidUsernameAndInvalidPassword_returnNull	Testing that when we find an employee by an invalid username and invalid password, a null object is returned	We are returned a null object	Passed	16/9/20	Theo
53		Repository	getEmployeeByValidUsernameAndInvalidPassword_returnNull	Testing that when we find an employee by their username and invalid password, a null object is returned	We are returned a null object	Passed	16/9/20	Theo
54		Repository	getEmployeeByInvalidUsernameAndValidPassword_returnNull	Testing that when we find an employee by an invalid username and valid password, a null object is returned	We are returned a null object	Passed	16/9/20	Theo
55		Service	whenValidIdentifier_ThenEmployeeShouldBeFound	Testing that when we find an employee by a corresponding identifier, the relevant employee is returned	We are returned the corresponding employee	Passed	16/9/20	Theo
56		Service	whenInValidIdentifier_ThrowEmployeeException	Testing that when we find an invalid identifier, we are returned a null object	We are returned a null object	Passed	16/9/20	Theo
57		Service	whenValidUserName_ThenEmployeeShouldBeFound	Testing that when we find an employee by a valid username, we are returned the valid employee	We are returned the valid employee	Passed	16/9/20	Theo
58		Service	whenInValidUserName_ThrowEmployeeException	Testing that when we find an employee by a invalid username, an exception is thrown	An exception is thrown	Passed	16/9/20	Theo
59		Service	whenValidEmail_ThenEmployeeShouldBeFound	Testing that when we find an employee by a valid email, we are returned the valid employee	We are returned the valid employee	Passed	16/9/20	Theo
60		Service	whenInValidEmail_ThenThrowEmployeeException	Testing that when we find an employee by a invalid email, an exception is thrown	An exception is thrown	Passed	16/9/20	Theo
61		Service	whenValidPhoneNumber_ThenEmployeeShouldBeFound	Testing that when we find an employee by a valid phone number, we are returned the valid employee	We are returned the valid employee	Passed	16/9/20	Theo
62		Service	whenInValidPhoneNumber_ThenThrowEmployeeException	Testing that when we find an employee by a invalid phone number, an exception is thrown	An exception is thrown	Passed	16/9/20	Theo
63		Service	getAllEmployees_returnListOfEmployees	Testing that when we get all employees, a list of all employees in the database are returned	A list of employees in the database are returned	Passed	16/9/20	Theo
64		Service	getEmployeeUsingValidUsernameAndPassword_returnEmployee	Testing that when we find an employee by their username and password, the relevant employee is returned	We are returned the relevant employee	Passed	16/9/20	Theo
65		Service	getEmployeeWithInvalidUsernameAndPassword_returnEmployeeException	Testing that when we find an employee by an invalid username and password, a null object is returned	An exception is thrown	Passed	16/9/20	Theo
66		Service	getEmployeeWithValidUsernameAndInvalidPassword_returnEmployeeException	Testing that when we find an employee by an invalid username and password, a null object is returned	An exception is thrown	Passed	16/9/20	Theo
67		Service	getEmployeeWithInvalidUsernameAndValidPassword_returnEmployeeException	Testing that when we find an employee by an invalid username and valid password, a null object is returned	An exception is thrown	Passed	16/9/20	Theo
68		Employee	whenInsertEmployeeFirstNameWithWrongSize_thenReturnFalse	Testing that when we try to insert an employee with an incorrect length, the employee is rejected	The employee is rejected	Passed	16/9/20	Theo
69		Employee	whenInsertFirstNameGreaterThanThreeCharacter_thenReturnTrue	Testing that we try to insert an employee with first name greater than 3 characters, the employee is accepted	The employee is accepted	Passed	16/9/20	Theo
70		Employee	whenInsertLastnameLessThanThreeCharcters_thenReturnFalse	Testing that we try to insert an employee with a last name less than 3 characters, the employee is rejected	The employee is rejected	Passed	16/9/20	Theo
71		Employee	whenInsertLastnameGreaterThanThreeCharcters_thenReturnTrue	Testing that we try to insert an employee with a last name greater than 3 characters, the employee is accepted	The employee is accepted	Passed	16/9/20	Theo
72		Employee	insertFirstNameThatIsBlank_thenReturnFalse	Testing that when we insert an employee with a blank first name, the employee is rejected	The employee is rejected	Passed	16/9/20	Theo
73		Employee	insertLastNameThatIsBlank_theReturnTrue	Testing that when we insert an employee with a blank last name, the employee is accepted	The employee is accepted into the database	Passed	16/9/20	Theo
74		Employee	insertEmailThatIsBlank_ThenReturnFalse	Testing that when we insert an employee with a blank email, the employee is rejected	The employee is rejected	Passed	16/9/20	Theo
75		Employee	insertEmailThatIsNotBlank_thenReturnTrue	Testing that when we insert an employee with a non-blank email, the employee is accepted	The employee is accepted	Passed	16/9/20	Theo
76		Employee	insertAddressThatIsBlank_thenReturnFalse	Testing that when we insert an employee with a blank address, the employee is rejected	The employee is rejected	Passed	16/9/20	Theo
77		Employee	insertAddressThatIsNotBlank_thenReturnTrue	Testing that when we insert an employee with a non-blank address, the employee is accepted	The employee is accepted	Passed	16/9/20	Theo
78		Employee	insertUserNameThatIsBlank_thenReturnFalse	Testing that when we insert an employee with a blank username, the employee is rejected	The employee is rejected	Passed	16/9/20	Theo
79		Employee	insertUserNameThatIsNotBlank_thenReturnTrue	Testing that when we insert an employee with a non-blank address, the employee is accepted	The employee is accepted	Passed	16/9/20	Theo
80		Employee	insertPasswordThatIsBlank_thenReturnFalse	Testing that when we insert an employee with a blank password, the employee is rejected	The employee is rejected	Passed	16/9/20	Theo
81		Employee	insertPasswordThatIsNotBlank_thenReturnTrue	Testing that when we insert an employee with a non-blank password, the employee is accepted	The employee is accepted	Passed	16/9/20	Theo
82		Employee	insertIdentifierThatIsBlank_thenIdentifier_thenCanNotBeBlank	Testing that when we insert an employee with a non-blank identifier, the employee is accepted	The employee is accepted	Passed	16/9/20	Theo
83		Employee	insertCorrectIdentifier_returnTrue	Testing that when we insert an employee with a non-blank identifier, the employee is accepted	The employee is accepted	Passed	16/9/20	Theo
84	ServiceType	Controller	givenServiceType_whenGetServiceType_thenReturnJsonArray	Testing that when we send a get request, all services in the database are returned in a list	A JSON list is returned with all of the services	Passed	16/9/20	Ghaida
85		Controller	givenNoServiceType_whenGetServiceType_thenReturnNullArray	Testing that when no services exist in the database, a blank list is returned when we send a get request	An empty JSON list is returned	Passed	16/9/20	Ghaida
86		Controller	givenNewServiceType_whenPostNewServiceType_thenNewServiceTypeIsAdded	Testing that when we send a post request containing valid service information, that a new service is created in the database	The new service is accepted and added to the database	Passed	16/9/20	Ghaida
87		ServiceType	whenInsertServiceTypeWithBlankServiceNo_thenReturnFalse	Testing that when we attempt to insert into the database a service with no service number, the object is rejected	The service is rejected	Passed	16/9/20	Ghaida
88		ServiceType	whenInsertServiceTypeWithValidServiceNo_thenReturnTrue	Testing that when we attempt to insert into the database a service with all valid fields, the object is accepted	The service is accepted	Passed	16/9/20	Ghaida
89		ServiceType	whenInsertServiceTypeWithBlankServiceName_thenReturnFalse	Testing that when we attempt to insert into the database a service with a blank service name, the object is rejected	The service is rejected	Passed	16/9/20	Ghaida
90		ServiceType	whenInsertServiceTypeWithValidServiceName_thenReturnTrue	Testing that when we attempt to insert into the database a service with all valid fields, the object is accepted	The service is accepted	Passed	16/9/20	Ghaida
91		Repository	whenFindServiceTypeByNo_thenReturnServiceName	Testing that when we find all services by service number, we are returned the relevant service objects	The relevant service objects are returned in a list	Passed	16/9/20	Ghaida
92		Repository	whenFindServiceTypeByServiceName_thenReturnServiceName	Testing that when we find all services by service name, we are returned the relevant service objects	The relevant service objects are returned in a list	Passed	16/9/20	Ghaida
93		Repository	whenServiceTypeDoesntExistWithServiceName_thenReturnNull	Testing that when we find all services by service name, and no services with that name exist, a null object is returned	A null object is returned	Passed	16/9/20	Ghaida
94		Service	whenValidName_ThenServiceTypeShouldBeFound	Testing that when we find a service by its name, we return the relevant service	The relevant service is returned	Passed	16/9/20	Ghaida
95		Service	getAllServiceType_returnListOfServiceType	Testing that when we get all service objects, we return all the services in the database in a list	A list of service is returned containing all services in the database	Passed	16/9/20	Ghaida
96	Booking	Controller	givenBooking_whenGetBookingById_thenReturnBooking	Testing that when we get bookings by their id, we are returned the relevant booking in the database	The relevant booking is returned	Passed	16/9/20	Dulshan
97		Controller	givenBooking_whenGetAllBookings_thenReturnBookingList	Testing that we get all bookings, a list of all of the bookings in the database are returned	A list of all bookings in the database is returned	Passed	16/9/20	Dulshan
98		Controller	givenBooking_whenDeleteBookingById_thenSuccessfulDeleteRequest	Testing that when we send a delete request, the relevant booking is removed from the database	The relevant booking is removed from the database	Passed	16/9/20	Dulshan
99		Booking	whenInsertBookingWithNoTime_thenReturnFalse	Testing that when we create a booking with no time attribute, the booking is rejected	The booking is rejected	Passed	16/9/20	Dulshan
100		Booking	whenInsertBookingWithNoDate_thenReturnFalse	Testing that when we create a booking with no date attribute, the booking is rejected	The booking is rejected	Passed	16/9/20	Dulshan
101		Booking	whenInsertBookingWithNoCustomer_thenReturnFalse	Testing that when we create a booking with no Customer attribute, the booking is rejected	The booking is rejected	Passed	16/9/20	Dulshan
102		Booking	whenInsertBookingWithNoEmployee_thenReturnFalse	Testing that when we create a booking with no Employee attribute, the booking is rejected	The booking is rejected	Passed	16/9/20	Dulshan
103		Booking	whenInsertBookingWithValidValues_thenReturnTrue	Testing that when we create a booking with all attributes valid, then the booking object is created	The new booking is created	Passed	16/9/20	Dulshan
104		Repository	whenFindBookingByValidEmployee_thenAssertTrue	Testing that when we find bookings by an employee, we find all of the relevant bookings in the database	A list of booking objects are returned associated with the relevant employee	Passed	16/9/20	Dulshan
105		Repository	whenFindBookingByNonExistentEmployee_thenAssertFalse	Testing that when we find bookings by an employee that doesn't exist in the database, we are returned an empty list	An empty list is returned	Passed	16/9/20	Dulshan
106		Repository	whenFindBookingByValidCustomer_thenAssertTrue	Testing that when we find bookings by a customer, we find all of the relevant bookings in the database	A list of booking objects are returned associated with the relevant customer	Passed	16/9/20	Dulshan

Test No:	Component	Description	Test Name:	Function:	Expected:	Result:	Test Date:	Ran by:
107	Repository		whenFindBookingByNonExistentCustomer_thenAssertFalse	Testing that when we find bookings by a customer that doesn't exist in the database, we are returned an empty list	An empty list is returned	Passed	16/9/20	Dulshan
108	Repository		whenFindAllThreeBookingsEntered_thenAssertTrue	Testing that when we find all bookings in the database, all of the booking objects are returned	A list with all booking objects are returned	Passed	16/9/20	Dulshan
109	Service		whenGettingBookingWithValidIdentifier_ThenBookingShouldBeFound	Testing that when we get bookings by the booking identifier, the relevant booking is returned	The relevant booking object is returned	Passed	16/9/20	Dulshan
110	Service		whenInvalidIdentifier_ThenThrowBookingException	Testing that when we get bookings by an invalid identifier, a booking exception is thrown	A booking exception is thrown	Passed	16/9/20	Dulshan
111	Service		getAllBookings_thenReturnListOfBookings	Testing that when we get all bookings, a list of all of the bookings in the database are returned	A list of all of the bookings in the database are returned	Passed	16/9/20	Dulshan
112	Service		whenDeletingBookingWithValidIdentifier_ThenSuccessfulDeleteRequest	Testing that when we delete bookings by a valid identifier, the relevant booking is removed from the database	The relevant booking is removed from the database	Passed	16/9/20	Dulshan
113	Service		whenDeletingBookingWithInvalidIdentifier_ThenThrowBookingException	Testing that when we delete bookings by an invalid identifier, a booking exception is thrown	A booking exception is thrown	Passed	16/9/20	Dulshan

Frontend Tests

Test No:	Component	Description	Test Name:	Function:	Expected:	Result:	Test Date:	Ran by:
1	App	testing custAuth method	should loggedInStatus should be set to LOGGED_IN	Testing that when a customerAith method is called the logged in status is set correctly	Login Status set to LOGGED_IN	Passed	16/9/20	Theo
2			customer should be set to true if method is called	Testing that when customerAuth is called that the customer state object should be set to true	The Customer state object contains all of the customer info	Passed	16/9/20	Theo
3			user should be reated as an object in app state	Testing that the user object will be effectively created in the app state when customerAuth is called	The User state object contains all of the user info	Passed	16/9/20	Theo
4		Testing employeeAuth	if employeeAuth is called then Logged in status should be set to LOGGED_IN	Testing that when employeeAuth is called the logged in status is set correctly	Logged In status is set to LOGGED_IN	Passed	16/9/20	Theo
5			If employee auth is called the employee should be set to true	Testing that when employeeAuth is called the logged in status is set correctly	Logged in status state is set to LOGGED_IN	Passed	16/9/20	Theo
6			if employeeAuth method is called a user should be in the App state	Testing that when employeeAuth is called the user state object is correctly updated	User state object contains all of the relevant employee info	Passed	16/9/20	Theo
7		Testing the toggleLogout method	Customer should be false	Testing that when toggleLogout is called, the customer state object is reset correctly	Customer object is set to false	Passed	16/9/20	Theo
8			Employee should be false	Testing that when toggleLogout is called, the employee state object is reset correctly	Employee object is set to false	Passed	16/9/20	Theo
9			loggedInStatus should be set to NOT_LOGGED_IN	Testing that when toggleLogout is called, the logged in status state object is change accordingly	Logged in status is set to NOT_LOGGED_IN	Passed	16/9/20	Theo
10			user should be blank	Testing that when toggleLogout is called, the user state object is updated accordingly	User state is updated to blank	Passed	16/9/20	Theo
11	Booking	<Booking /> component tests	Testing User redirected to home if not logged in	Testing that when a user navigates to the booking page without the logged in status set correctly, they are handled accordingly	User is navigated to the home page	Passed	16/9/20	Jack
12			Testing Services are Set when Booking is rendered	Testing that when the <Booking> component has been rendered, the service state object is updated correctly	The services state object is set correctly	Passed	16/9/20	Jack
13			Testing Services state when selected	Testing that when a service is selected, the selected service state object is set correctly	The selected service state object is set to the relevant service number	Passed	16/9/20	Jack
14			Testing Service updates to null when null state when selected	Testing that when the service is selected to non-null selection, then update to the null selection, the selected service state object updates correctly	The selected service state object is set to null	Passed	16/9/20	Jack
15			Testing Employees Field is diabled when no service is selected	Testing that when no service is selected, the employee selection field is disabled	The employee selection field is set to disabled	Passed	16/9/20	Jack
16			Testing Date Field is diabled when no service is selected	Testing that when no service is selected, the date selection field is disabled	The date selection field is set to disabled	Passed	16/9/20	Jack
17			Testing Time Field is diabled when no service is selected	Testing that when no service is selected, the time selection field is disabled	The time selection field is set to disabled	Passed	16/9/20	Jack
18			Testing Submit Button is diabled when no service is selected	Testing that when no service is selected, the submit button is disabled	The submit button is set to disabled	Passed	16/9/20	Jack
19			Testing Employees update when service selected	Testing that when a service is selected, the employees state object is updated to contain the relevant employees	The employees state object contains the correct employees	Passed	16/9/20	Jack
20			Testing Timeslots Are set When date is set	Testing that when a date is selected, the times state object is set and contains the relevant times	The times state object is set to be the correct length	Passed	16/9/20	Jack
21			Testing already booked times are not displayed	Testing that when a date is selected containing pre-booked times, no already booked times are displayed	Only the non-booked times are displayed	Passed	16/9/20	Jack
22			Testing submit button is enabled when all fields are selected	Testing that when all of the fields are selected, the submit button is enabled	The submit button is set to enabled	Passed	16/9/20	Jack
23			Testing submit button is disabled when fields are deselcted	Testing that when the fields are selected then deselected, the submit button is disabled	The submit button is set to disabled	Passed	16/9/20	Jack
24	Header	Header elements	should contain nav elements Home, about us, staff Login, login and register	Testing that when user is not logged in, the navbar contains all of the relevant links	The navbar contains a link to home, about us, login, staffLogin and createLogin	Passed	16/9/20	Theo
25			handle logout when employee is logged in	Testing that when an employee is logged in, the navbar contains the logout component	The navbar contains a logout component	Passed	16/9/20	Theo
26			handle logout when customer is logged in	Testing that when a customer is logged in, the navbar contains the logout component	The navbar contains a logout component	Passed	16/9/20	Theo
27	Login	<Login /> test for login component	Username check	Testing that when a user types their name into the login, the username state object is correctly set	The username state object is updated to the correct value	Passed	16/9/20	Theo
28			Password Check	Testing that when a user types their password into the login, the password state object is correctly set	The password state object is updated to the correct value	Passed	16/9/20	Theo
29			toggleGet has been implemented	Testing that the toggleGet method correctly sends a get request and updateds the state correctly	The loggedIn state object is set correctly	Passed	16/9/20	Theo
30			axios get request is successful should set logged in state to true	Testing that when the toggleGet method is called and the get request suceeds, the logged in status is set correctly	The loggedIn sate is set to true	Passed	16/9/20	Theo
31			axios get request is unsuceful and loggedIn state is false	Testing that when the toggleGet method is called and the get request is called unsuccessfully, the logged in status is set to false	The loggedIn state is set to fasle	Passed	16/9/20	Theo
32			Page redirects suscesfully after succesful axios request	Testing that when a user successfully logs in, they are redirected to the home page	The component has now been rendered to the front end	Passed	16/9/20	Theo
33	Signup	Signup page check fields	First Name check	Testing that when the first name field is updated the state is updated correctly	The first name state object is set correctly	Passed	16/9/20	Theo
34			Last Name check	Testing that when the last name field is updated, the state is updated correctly	The last name state object is set correctly	Passed	16/9/20	Theo
35			Email check	Testing that when the email field is updated, the state is updated correctly	The email state object is set correctly	Passed	16/9/20	Theo
36			Address check	Testing that when the address field is updated, the state us updated correctly	The address state object is set correctly	Passed	16/9/20	Theo
37			Phone Number check	Testing that when the phone number field is updated, the state us updated correctly	The phone number state object is set correctly	Passed	16/9/20	Theo
38			Username check	Testing that when the username field is updated, the state us updated correctly	The usernmae state object is set correctly	Passed	16/9/20	Theo
39			password check	Testing that when the password field is updated, the state us updated correctly	The password state object is set correctly	Passed	16/9/20	Theo
40			Confirm password check	Testing that when the confirm password field is updated, the state us updated correctly	The confirm password state object is set correctly	Passed	16/9/20	Theo
41			validate password to be true when passwords are the same	Testing that when both of the input password fields are set to the same value, the validate method returns true	The validate method returns true	Passed	16/9/20	Theo
42			Validate password to be false when passwords are not the same	Testing that when both of the input password fields are set to differnet values, the validate method returns false	The validate method returns false	Passed	16/9/20	Theo
43		Axios post request Sign up page	should succesfully perform a post request will set isSignedUp to true	Testing that a succesful post request sets the isSignedUp state object to be true	The isSignedUp state object is set to true	Passed	16/9/20	Theo
44			should unSuccessfully perform a post request and isSignedUp Should Be false	Testing that an unsuccessful post request will set isSignedUp to be false	The isSignedUp state object is set to false	Passed	16/9/20	Theo
45			should redirect page upon succesfull post request to homepage	Testing that a successful sign up will redirect the user to the home page	The home screen has been rendered	Passed	16/9/20	Theo
46	StaffLogin	Tests for Staff login Component	Username check in state	Testing that when the username field is updated, the state us updated correctly	The first name state object is set correctly	Passed	16/9/20	Theo
47			Password check in state	Testing that when the password field is updated, the state us updated correctly	The password state object is updated to the correct value	Passed	16/9/20	Theo
48			toggleGet has been implemented	Testing that the toggleGet method correctly sends a get request and updateds the state correctly	The loggedIn state object is set correctly	Passed	16/9/20	Theo
49			axios get request is successful should set logged in state to true	Testing that when the toggleGet method is called and the get request suceeds, the logged in status is set correctly	The loggedIn state is set to true	Passed	16/9/20	Theo
50			axios get request is unsuceful and loggedIn state is false	Testing that when the toggleGet method is called and the get request is called unsuccessfully, the logged in status is set to false	The loggedIn state is set to fasle	Passed	16/9/20	Theo
51			Page redirects suscesfully after succesful axios request	Testing that when a user succesfully logs in, they are redirected to the home page	The component has now been rendered to the front end	Passed	16/9/20	Theo
52	Customer	Customer Page information display	First Name check	Testing that when the user logged in changes the first name displayed is correct	The first name state object is set correctly	Passed	20/9/20	Dulshan
53			Last Name check	Testing that when the user logged in changes the last name displayed is correct	The last name state object is set correctly	Passed	21/9/20	Dulshan
54			Email check	Testing that when the user logged in changes the email displayed is correct	The email state object is set correctly	Passed	22/9/20	Dulshan
55			Address check	Testing that when the user logged in changes the address displayed is correct	The address state object is set correctly	Passed	23/9/20	Dulshan
56			Phone Number check	Testing that when the user logged in changes the phone number displayed is correct	The phone number state object is set correctly	Passed	24/9/20	Dulshan
57			username check	Testing that when the user logged in changes the user name displayed is correct	The user name state object is set correctly	Passed	25/9/20	Dulshan
58	Employee	Employee Page Information display	First Name check	Testing that when the user logged in changes the first name displayed is correct	The first name state object is set correctly	Passed	20/9/20	Dulshan
59			Last Name check	Testing that when the user logged in changes the last name displayed is correct	The last name state object is set correctly	Passed	21/9/20	Dulshan
60			Email check	Testing that when the user logged in changes the email displayed is correct	The email state object is set correctly	Passed	22/9/20	Dulshan
61			Address check	Testing that when the user logged in changes the address displayed is correct	The address state object is set correctly	Passed	23/9/20	Dulshan
62			Phone Number check	Testing that when the user logged in changes the phone number displayed is correct	The phone number state object is set correctly	Passed	24/9/20	Dulshan
63	AccountEdit	Account edit page details display correctly	First Name check	Testing that when the user edits their details, their existing first name is already entered, and changes affect the state value	The first name state object is set correctly	Passed	22/9/20	Julian
64			Last Name check	Testing that when the user edits their details, their existing last name is already entered, and changes affect the state value	The last name state object is set correctly	Passed	23/9/20	Julian
65			Email check	Testing that when the user edits their details, their existing email is already entered, and changes affect the state value	The email state object is set correctly	Passed	24/9/20	Julian
66			Address check	Testing that when the user edits their details, their existing address is already entered, and changes affect the state value	The address state object is set correctly	Passed	25/9/20	Julian
67			Phone Number check	Testing that when the user edits their details, their existing phone number is already entered, and changes affect the state value	The phone number state object is set correctly	Passed	26/9/20	Julian
68		Check for user logged in status	Logged out user	Test the change details form will not render	Form was not rendered, message appears instead	Passed	27/9/20	Julian
69			Logged in user	Test the change details form will render	Form was rendered	Passed	28/9/20	Julian
70	PasswordChange	Check for user logged in status	Logged out user	Test the change password form will not render	Form was not rendered, message appears instead	Passed	29/9/20	Julian
71			Logged in user	Test the change password form will render	Form was rendered	Passed	30/9/20	Julian
72	BusinessWorkingHours	Testing redirecting of page when not logged in as admin	Testing User redirected to staffLogin if not logged in as admin	Testing User redirected to staffLogin if not logged in as admin, he will be redriected to the staff login page	Gets redirected to the StaffLogin Page	Passed	12/10/20	Dulshan
73		Testing current start Time display	Testing current start Time	Testing current start Time	properly displays the preset start time	Passed	13/10/20	Dulshan
74		Testing current start Time display	Testing current end Time	Testing current end Time	properly displays the preset end time	Passed	14/10/20	Dulshan
75			Testing that the set new values get set in the database correctly	Testing current end Time current start time being updated	Testing current end Time and Time current start time being updated	Passed	15/10/20	Dulshan
76	Roster	<Roster/>test for Roster.js	Testing that the state updates correctly when page renders	Testing that when the Roster component is loaded that state is updated correctly from the given props	The user roster was set correctly	Passed	15/10/20	Jack
77			Testing that the state updates when change is selected	Testing that when a rostering change is selected that the state is updated correctly	The requestedSunday state was updated correctly	Passed	15/10/20	Jack
78			Testing that the state updates when request is sent	Testing that when the submit button is clicked the put request is sent and the state is updated accordingly	The request was sent correctly and the isApproved value is updated	Passed	15/10/20	Jack
79								
80								
81								
82								
83								
84								
85								
86								
87								
88								
89								