External API Test Cases - Automation Documentation

	Assertions				
	Test Cases/Scenarios		Expected Result	Actual Result	Status
UserController					
POST /TeeOff/v1/User/AuthenticateUser	Verify the following username and password combinations.				
	Valid Parameters	valid username, valid password	1. Should be able to successfully signin using username. 2. Response data should have the correct user details (e.g ContactId, Email, UserName & DateCreated)	1. Able to signin using username successfully. 2. Response data has the correct user details (e.g ContactId, Email, UserName & DateCreated)	PASS
		valid email , valid password	1. Should be able to successfully signin using email address. 2. Response data should have the correct user details (e.g ContactId, Email, UserName & DateCreated)	1. Able to signin using email address successfully. 2. Response data has the correct user details (e.g ContactId, Email, UserName & DateCreated)	PASS
	Invalid Parameters	empty username/email, empty password	Invalid/missing request object	1. Message displays "Invalid/missing request object"	PASS
		empty username/email, invalid pasword	Invalid/missing request object	1. Message displays "Invalid/missing request object"	PASS
		empty username/email, valid password	Invalid/missing request object	1. Message displays "Invalid/missing request object"	PASS
		invalid username/email, empty password	Invalid/missing request object	1. Message displays "Invalid/missing request object"	PASS
		valid username/email, empty password	Invalid/missing request object	1. Message displays "Invalid/missing request object"	PASS
		invalid username/email, invalid password	1. Sign in should fail. 2. Error Message should display as "Invalid user account or password".	1. Sign in fails. 2. Error Message displays "Invalid user account or password".	PASS
		invalid username/email, valid password	1. Sign in should fail. 2. Error Message should display as "Invalid user account or password".	1. Sign in fails. 2. Error Message displays "Invalid user account or password".	PASS
		valid username/email, invalid password	1. Sign in should fail. 2. Error Message should display as "Invalid user account or password".	1. Sign in fails. 2. Error Message displays "Invalid user account or password".	PASS
POST /TeeOff/v1/User/SessionStart					
	Valid Parameters	successful starting session	1. Should successfully start a session without error..	1. Able to start session without problems	PASS
GET /TeeOff/v1/User/GetUser					
	Valid Parameters	correct contactId passed	1. Should be able to successfully get user details. 2. Response data should have the correct user details (e.g Username, DateCreated, FirstName, LastName, ContactID, Email)	1. Able to get user details successfully. 2. Response data has correct user details (e.g Username, DateCreated, FirstName, LastName, ContactID, Email)	PASS
	Invalid Parameters	using wrong ContactID	1. Message should display "User has not been authenticated for the requested session."	1. Message displays "User has not been authenticated for the requested session."	PASS
		no ContactID provided	1. Message should display "The request is invalid"	1. Message displays "The request is invalid"	PASS
POST /TeeOff/v1/User/AddUser					
	Valid Parameters	new registration with: a. correct FirstName b. correct LastName c. valid & unregistered email d. valid postal code	1. Should be able to successfully register user. 2. Should save correct details after registration (FirstName, Lastname, Email, Postal Code)	1. Able to register user successfully. 2. Saves correct details after registration (FirstName, Lastname, Email, Postal Code)"	PASS
		new registration with: a. correct FirstName b. correct LastName c. valid & unregistered email d. valid postal code e) Correct phone number after registration f) correct country after registration g) correct city after registration	1. Should be able to successfully register user. 2. Should save correct details after registration (FirstName, Lastname, Email, Postal Code) 3) Correct phone number after registration 4) correct country id after registration 5) correct city after registration	1. Able to register user successfully. 2. Saves correct details after registration (FirstName, Lastname, Email, Postal Code) 3) Returns correct phone number after registration 4) Returns correct country id after registration 5)Returns correct city after registration	PASS
	Invalid Parameters	Using existing email (example: limesourcetester@gmail.com)	1. Registration should fail. 2. Message should display "User Name [limesourcetester@gmail.com] error. This user already exists."	1. Registration fails. 2. Message displays "User Name [limesourcetester@gmail.com] error. This user already exists."	PASS
		setting password to less than 6 characters	1. Registration should fail. 2. Message should display "Your password must be at least 6 characters!"	1. Registration fails. 2. Message displays "Your password must be at least 6 characters!"	PASS
		passing invalid email address	1. Registration should fail. 2. Message should display " Email address (invalidemail) appears to be invalid, please enter unique username or contact support at (855) 383-3633."	1. Registration fails. 2. Message displays " Email address (invalid email) appears to be invalid, please enter unique username or contact support at (855) 383-3633."	PASS
		Invalid Postal Code. (example: $$$$)	1. Registration should not be successful.	1. Able to register with invalid postal code	FAIL
		If user is able to signin after successful registration using the correct email and password	1. Should be able to successfully signin after successful registration.	1. Able to successfully signin after successful registration using the correct email and password	PASS
		using existing username but unique email address	1. Registration should fail. 2. Message should display "User Name [{{existing username}}] error. This user already exists."	1. Registration fails. 2. Message displays "User Name [{{existing username}}] error. This user already exists."	PASS
POST /TeeOff/v1/User/UpdateUser					
	Valid Parameters	Update User - Successful Note: Please run "Revert Original User Password & Name" Test case every after successful update.	1. Should be able to successfully update user details. 2. Response data should show the updated details correctly (eg. firstName, lastName, ContactID, Username, CountryId, Username, CountryId, PostalCode, Address1, Address2, City, Email, HomePhone,)	1. Able to successfully update details. 2. User details are updated correctly.	PASS
		If user is able to sigin using updated password.	1. Should be able to sigin using the updated password.	1. User is able to signin using the updated password.	PASS
		Update User Gender	1. Should return correct Gender after update.	1. Gender is not updated correctly.	FAIL
	Invalid Parameters	Update User - Empty Email	1. Should not update successfully. 2. Message should display " Your email address () is invalid. Please contact the TeeOff.com helpdesk for assistance if necessary."	1. Update user detail fails. 2. Message displays "Your email address () is invalid. Please contact the TeeOff.com helpdesk for assistance if necessary."	PASS
		Update User - Invalid Email	1. Should not update successfully. Message should display "Your email address ({invalidemail}) is invalid. Please contact the TeeOff.com helpdesk for assistance if necessary."	1. Update user detail fails. 2. Message displays "Your email address ({invalidemail}) is invalid. Please contact the TeeOff.com helpdesk for assistance if necessary."	PASS
		Update User - Updating email address that is already used by other account.	1. Should not update successfully. 2. Message should display "Your email address (limesourcetester@gmail.com) is already in use on the TeeOff.com site. Please contact the TeeOff.com helpdesk for assistance if necessary."	1. Update user detail fails. 2. Message displays "Your email address (limesourcetester@gmail.com) is already in use on the TeeOff.com site. Please contact the TeeOff.com helpdesk for assistance if necessary."	PASS
		Update User - Updating password to empty	1. Should not update successfully. 2.Message should display "Your password must be at least 6 characters!"	1. Update user detail fails. 2. Message displays "Your password must be at least 6 characters."	PASS

External API Test Cases - Automation Documentation						
				Assertions		
		Test Cases/Scenarios		Expected Result	Actual Result	Status
UserController						
	POST /TeeOff/v1/User/AuthenticateUser	Verify the following username and password combinations.				
			Update User - Updating password to less 6 characters	1. Should not update successfully. 2. Message should display "Your password must be at least 6 characters!"	1. Update user detail fails. 2. Message displays "Your password must be at least 6 characters."	PASS
			Update User - Empty FirstName	1. Should not update successfully.	1. Able to update firstname to empty	FAIL
			Update User - Empty LastName	1. Should not update successfully.	1. Able to update lastname to empty	FAIL
			Update User - Updating postal code to empty	1. Should not update successfully.	1. Able to update Postal code to empty.	FAIL
			using existing username but unique email address	1. Udate should fail. 2. Message should display "User Name [{{existing username}}] error. This user already exists."	1. Updae fails. 2. Message displays "User Name [{{existing username}}] error. This user already exists."	PASS
	POST /TeeOff/v1/User/UpdateUserPreference					
		Valid Parameters	UpdateUserPreference - Set All to True	1. All User Preference (AllowNewsLetter, AllowAlerts, AllowDealsDiscounts) should be updated to True	1. Update User Preference is successful. 2. All User Preference (AllowNewsLetter, AllowAlerts, AllowDealsDiscounts) are updated to True	PASS
			UpdateUserPreference - Set All to False	1. All User Preference (AllowNewsLetter, AllowAlerts, AllowDealsDiscounts) should be updated to False	1. Update User Preference is successful. 2. All User Preference (AllowNewsLetter, AllowAlerts, AllowDealsDiscounts) are updated to False	PASS
			UpdateUserPreference - Only set AllowNewsLetter to true	1. AllowNewsLetter is True 2. AllowAlerts is False 3. AllowDealsDiscounts is False 4. Success is True	1. AllowNewsLetter is True 2. AllowAlerts is False 3. AllowDealsDiscounts is False 4. Success is True	PASS
			UpdateUserPreference - Only set AllowAlerts to true	1. AllowNewsLetter is False 2. AllowAlerts is True 3. AllowDealsDiscounts is False 4. Success is True	1. AllowNewsLetter is False 2. AllowAlerts is True 3. AllowDealsDiscounts is False 4. Success is True	PASS
			UpdateUserPreference - Only set AllowDealsDiscounts to true	1. AllowNewsLetter is False 2. AllowAlerts is False 3. AllowDealsDiscounts is True 4. Success is True	1. AllowNewsLetter is False 2. AllowAlerts is False 3. AllowDealsDiscounts is True 4. Success is True	PASS
		Invalid Parameters	UpdateUserPreference - Wrong/Empty Session	1. UpdateUserPreference should fail. 2. Message should display " Invalid/missing request object."	1. Update User Preference fails. 2. Message displays "Invalid/missing request object."	PASS
			UpdateUserPreference - Wrong ContactID	1. UpdateUserPreference should fail. 2. Message should display " Invalid/missing request object."	1. Update User Preference fails. 2. Message displays "Invalid/missing request object."	PASS
	POST /TeeOff/v1/User/RegisterGuest					
		Valid Parameters	Register Guest (Valid Guest Details using Unregistered Account)	1. Should successfully register guest. 2 Email should be the same as the GuestEmail passed. 3. Firstname should be saved correctly. 4. Lastname should be saved correctly. 5. Username should be generated (unregistered account)	1. Able to register guest successfully. 2. Email is the same as the GuestEmail passed. 3. Firstname is correct. 4. Lastname is correct. 5. Username is generated(unregistered account)	PASS
			Register Guest (Valid Guest Details using Registered Account)	1. Should successfully register guest. 2 Email should be the same as the GuestEmail passed. 3. Firstname should be saved correctly. 4. Lastname should be saved correctly. 5. Username should be correct.	1. Able to register guest successfully. 2. Email is the same as the GuestEmail passed. 3. Firstname is correct. 4. Lastname is correct. 5. Username is correct.	PASS
		Invalid Parameters	Register Guest (Register Guest with Empty Email)	1. Should not register successfully. 2. Should display "Invalid/missing request object."	1. Registration fails. 2. Displays "Invalid/missing request object."	PASS
			Register Guest (Valid Guest Details using Registered Account)	1. Should not register successfully. 2. Should display "Invalid/missing request object."	1. Registration fails. 2. Displays "Invalid/missing request object."	PASS
			Register Guest (No Firstname provided)	1. Should not register successfully. 2. Should display "Invalid/missing request object."	1. Registration fails. 2. Displays "Invalid/missing request object."	PASS
			Register Guest (No LastName provided)	1. Should not register successfully. 2. Should display "Invalid/missing request object."	1. Registration fails. 2. Displays "Invalid/missing request object."	PASS
			Register Guest (Invalid provided)	1. Should not register successfully. 2. Should display "Invalid email address."	1. Registration fails. 2. Displays "Invalid email address."	PASS
	POST /TeeOff/v1/User/ResetPassword					
	to test: - check reset code expiration (24 hours) (in db)? - reset using valid but unregistered account *DONE Low Priority, but **NOT IMPOSSIBLE** when we can data-drive test (https://www.soapui.org/tutorials/tutorial-data-driven-testing.html) -- checking of ResetCodeExpiration under User table (if DEV has the ability to set this) -- User table has ResetCode != Null && ResetCodeExpiration != Null -- we need to atleast test that somehow reset code is available or possible in API, if no other way then OK.	Valid Parameters	Reset Password (Using Valid Email Address)	1. Should return Success as True. 2. Response Data is the same as the Email Address passed in the request.	1. Success returns True. 2. Response Data is the same as the Email Address passed.	PASS
			Reset Password (Using Valid UserName)	1.Should return Success as True. 2. Response Data is the same as the userName passed in the request.	1. Success returns True. 2. Response Data is the same as the userName passed in the request.	PASS
		Invalid Parameters	Reset Password (Using Invalid Email Address/UserName)	1. Should return Success as False. 2. Message should return "The requested user does not exist in the TeeOff.com system. Please register a new account."	1. Reset Password fails. 2. Message displays "The requested user does not exist in the TeeOff.com system. Please register a new account."	PASS
			Reset Password (No Email Address/Username provided)	1. Should return "Invalid/missing request object" error.	1. Reset Password fails. 2. Displays "Invalid/missing request object."	PASS
			Reset Password (Using valid but unregistered account)	1. Should return Success as False. 2. Message should return "The requested user does not exist in the TeeOff.com system. Please register a new account."	1. Reset Password fails. 2. Message displays "The requested user does not exist in the TeeOff.com system. Please register a new account."	PASS
	GET /TeeOff/v1/User/GetFutureReservations					
	Viewing Future reservations of Guest using registered account: --- a. non-registered guest account --- b. registered guest account & same registered signed in account --- c. registered guest account but different registered signed in account	Valid Parameters	Get Future Reservations - Successful Note: User must have future reservations.	1. Should return Success as True. 2. Should return future reservations of user. 3. Should return correct reserved tee time confirmation number 4. Should return correct reserved tee time Paid amount 5. Should return correct reserved tee time Due at Course	1. Success returns True. 2. Able to retrieve user's future reservations. 3. Returns correct reserved tee time confirmation number 4. Returns correct reserved tee time Paid amount 5. Returns correct reserved tee time Due at Course	PASS
		Invalid Parameters	Get Future Reservations - Wrong Session	1. Should return "Invalid/missing request object" error.	1. Returns "Invalid/missing request object"	PASS
			Get Future Reservations - Wrong ContactID	1. Should return "The request is invalid" error.	1. Returns "Invalid/missing request object"	PASS
			Get Future Reservations - Empty ContactID provided	1. Should return "The request is invalid" error.	1. Returns "Invalid/missing request object"	PASS
			Get Future Reservations - No Reservations	1. Should return Success to True. Data count must be 0.	1. Success returns True 2. Data count is 0.	PASS
	GET /TeeOff/v1/User/GetPastReservations					
		Valid Parameters	Get Past Reservations - Successful Note: User must have past reservations.	1. Should return Success as True. 2. Should return past reservations of user. 3. Should return correct reserved tee time confirmation number 4. Should return correct reserved tee time Paid amount 5. Should return correct reserved tee time Due at Course	1. Returns Success as True. 2. Returns past reservations of user. 3. Returns correct reserved tee time confirmation number 4. Returns correct reserved tee time Paid amount 5. Returns correct reserved tee time Due at Course	PASS

External API Test Cases - Automation Documentation					
	Assertions				
	Test Cases/Scenarios		Expected Result	Actual Result	Status
UserController					
POST /TeeOff/v1/User/AuthenticateUser	Verify the following username and password combinations.				
	Invalid Parameters	Get Past Reservations - Wrong Session	1. Should return "Invalid/missing request object" error.	1. Returns "Invalid/missing request object" error.	PASS
		Get Past Reservations - Wrong ContactID	1. Should return "The request is invalid" error.	1. Returns "The request is invalid" error.	PASS
		Get Past Reservations - Empty ContactID provided	1. Should return "The request is invalid" error.	1. Returns "The request is invalid" error.	PASS
		Get Past Reservations - No Reservations	1. Should return Success to True. Data is 0.	1. Returns Success to True. Data is 0.	PASS
GET /TeeOff/v1/User/GetCreditCardInfo					
	Valid Parameters	Get Credit Card - Successful	1. Should return Success as True. 2. Should return saved credit cards of user.	1. Returns Success as True. 2. Returns saved credit cards of user.	PASS
	Invalid Parameters	Get Credit Card - Wrong Session	1. Should return "Invalid/missing request object" error.	1. Returns "Invalid/missing request object" error.	PASS
		Get Credit Card - Wrong ContactId	1. Should return "The request is invalid" error.	1. Returns "The request is invalid" error.	PASS
POST /TeeOff/v1/User/DeleteCreditCard					
	Valid Parameters	Delete Credit Card - Successful	1. Should return Success as True. 2. Should delete the correct Card.	1. Returns Success as True. 2. Deletes the correct Card.	PASS
	Invalid Parameters	Delete Credit Card - Wrong Session	1. Should return "Invalid/missing request object" error.	1. Returns"Invalid/missing request object" error.	PASS
		Delete Credit Card - Empty ContactId	1. Should return "Invalid/missing request object" error.	1. Returns"Invalid/missing request object" error.	PASS
		Delete Credit Card - Wrong ContactId	1. Should return "Invalid/missing request object" error.	1. Returns"Invalid/missing request object" error.	PASS
		Delete Credit Card - Empty CardId	1. Should return "Invalid/missing request object" error.	1. Returns"Invalid/missing request object" error.	PASS
		Delete Credit Card - Wrong CardId	1. Should return "Invalid/missing request object" error.	1. Returns"Invalid/missing request object" error.	PASS
		Delete Credit Card - Previously Deleted Card	1. Should not return successful deletion.	1. Returns Success as True. 2. Able to delete again the card that was previously deleted.	FAIL
POST /TeeOff/v1/User/AddCreditCardInfo					
	Valid Parameters	Successful - using correct parameters: a. valid credit card b. valid expiration month and year c. acceptable card type	1. Should return Success as True. 2. Should save the correct Last Four digits of credit card. 3. Should return the correct CardType of Credit card. 4. Should save the correct Expire Date.	1. Success is True. 2. Saves the correct Last Four digits of credit card. 3. Returns the correct CardType of Credit card. 4. Saves the correct Expire Date.	PASS
	Invalid Parameters	adding on wrong ContactId	1. Should return Success to False. 2. Should return "<b>Sorry, We Need a Mulligan.</b> Let's try again." 3. Should not save card	1. Success is False. 2. Returns "<b>Sorry, We Need a Mulligan.</b> Let's try again." 3. Card is not saved.	PASS
		adding but no Card Number provided	1. Should return Success as False. 2. Should return "User Error: All fields are required." message. 3. Should not save card	1. Success is False. 2. Returns "User Error: All fields are required." 3. Card is not saved.	PASS
		invalid Card Number provided.	1. Should return Success as False. 2. Should return "Credit Card Error: The card entered was not valid. Please reenter payment information. Need help? Call us: 1-855-383-3633. (C03)" message. 3. Should not save card	1. Success is False. 2.Returns "Credit Card Error: The card entered was not valid. Please reenter payment information. Need help? Call us: 1-855-383-3633. (C03)" message. 3. Card is not saved.	PASS
		no Postal Code Provided	1. Should return Success as False. 2. Should return "User Error: All fields are required." message. 3. Should not save card	1. Success is False. 2. Returns "User Error: All fields are required." message. 3. Card is not saved.	PASS
		invalid Postal Coded provided (Example: $$$$)	1. Should return Success as False. 2. Should not save the card successfully.	1. Returns Success as True. 2. Able to save card successfully with invalid postal code.	FAIL
		Add Credit Card - No card Expiration provided	1. Should return Success as False. 2. Should return "<b>Sorry, We Need a Mulligan.</b> Let's try again."	1. Success is False. 2. Returns"<b>Sorry, We Need a Mulligan.</b> Let's try again."	PASS
		Add Credit Card - Invalid Expire Month provided (example: 13 or more than 12; 00 or less than 1)	1. Should return Success as False. 2. Should return "<b>Sorry, We Need a Mulligan.</b> Let's try again."	1. Success is False. 2. Returns "<b>Sorry, We Need a Mulligan.</b> Let's try again."	PASS
		Add Credit Card - Using Past year as Expire year	1. Should return Success as False. 2. Should return "Credit Card Expired: Please update the expiration date entered or try a new card. Need help? Call us: 1-855-383-3633. (C01)" message.	1. Returns Success as False. 2. Returns "Credit Card Expired: Please update the expiration date entered or try a new card. Need help? Call us: 1-855-383-3633. (C01)" message.	PASS
		Add Credit Card - Using Expired Card . - valid expiration month and year but still a past date. example: 01 / 16 which is invalid because today is 11/16)	1. Should return Success as False. 2. Should return "Credit Card Expired: Please update the expiration date entered or try a new card. Need help? Call us: 1-855-383-3633. (C01)" message.	1. Returns Success as False. 2. Returns "Credit Card Expired: Please update the expiration date entered or try a new card. Need help? Call us: 1-855-383-3633. (C01)" message.	PASS
		Add Credit Card - No NameOncard provided	1. Should return Success as False. 2. Should return "User Error: All fields are required." message.	1. Returns Success as False. 2. Returns "User Error: All fields are required." message.	PASS
		Add Credit Card - Wrong/Empty Session	1. Should return "Invalid/missing request object" error.	1. Returns "Invalid/missing request object" error.	PASS
POST /TeeOff/v1/User/SetDefaultCreditCard					
	Valid Parameters	- Set Default Credit Card - Successful	1. Should return Success as True.	1. Returns Success as True.	PASS
	Invalid Parameters	- Set Default Credit Card - Wrong Session	1. Should return "Invalid/missing request object" error.	1. Returns "Invalid/missing request object" error.	PASS
ReservationController					
POST /TeeOff/v1/Reservation/ProcessTeeTimeCheckout					
	Valid Parameters	ProcessTeeTimeCheckout - Successful	1. Should return Success as True. 2. Tee time should be added to the cart successfully. 3. correct player number should be added 4. correct tee time should be added 5. tee time added should be from the correct course	1. Returns Success as True. 2. Tee time is added to the cart successfully. 3. correct player number is added 4. correct tee time is added 5. tee time added is from the correct course	PASS
	Invalid Parameters	- Tee time is no longer available	1. Should return Success as False. 2. Message should display "We're sorry, this tee time is no longer available."	1. Returns Success as False. 2. Message displays "We're sorry, this tee time is no longer available."	PASS
		- wrong session	1. Should return "Invalid/missing request object" error.	1. Returns "Invalid/missing request object" error.	PASS
		- no tee time provided	1. Should return Success as False. 2. Message should display "Sorry, We Need a Mulligan."	1. Returns Success as False. 2. Message displays "Sorry, We Need a Mulligan."	PASS
POST /TeeOff/v1/Reservation/ApplyPromoCode					
TODO: - check course AllowPromos == true		- using promo code that is valid only in Mobile app	1. Should return Success as True 2. ResultMessage contains something like Coupon is valid for 2 golfer (s). Your calculated discount is... 3. Coupon description should be "Valid only on Mobile app" 4. Coupon discount is applied. 5. Result message should not contain "Coupon code is invalid".	1. Returns Success as True 2. ResultMessage contains something like Coupon is valid for 2 golfer(s). Your calculated discount is... 3. Coupon description is "Valid only on Mobile app" 4. Coupon discount is applied. 5. Result message contains "Coupon code is invalid".	PASS
		- using promo code that is valid only in TeeOff.com website	1. Should return Success as True 2. ResultMessage contains something like "Coupon [Valid only on Teeoff.com] can only be redeemed from the TeeOff.com website." 3. Coupon discount should be 0.0	1. Returns Success as True 2. ResultMessage contains something like "Coupon [Valid only on Teeoff.com] can only be redeemed from the TeeOff.com website." 3. Coupon discount is 0.0	PASS
		- using invalid promo code	1. Should return Success as True 2. ResultMessage should be "Coupon code is invalid. Coupon does not exist or is inactive." 3. Coupon discount should be 0.0	1. Returns Success as True 2. ResultMessage is "Coupon code is invalid. Coupon does not exist or is inactive." 3. Coupon discount is 0.0	PASS
		- using empty promo code	1. Should return "Invalid/missing request object" error.	1. Returns "Invalid/missing request object" error.	PASS
		- applying promo code to a Non-Deal time	1. Should return Success as False. 2. Message should display "Sorry, We Need a Mulligan."	1. Returns Success as False. 2. Message displays "Sorry, We Need a Mulligan."	PASS

External API Test Cases - Automation Documentation						
		Assertions				
		Test Cases/Scenarios		Expected Result	Actual Result	Status
UserController						
	POST /TeeOff/v1/User/AuthenticateUser	Verify the following username and password combinations.				
			- using promo code that is valid only for a specific player (example: valid only for 4 players)	Unable to test. No available promo code that is valid for only 4 players		PASS
	POST /TeeOff/v1/Reservation/ModifyReservation					
			- if tee time is valid for modification	TODO		NO RUN
			- wrong session	TODO		NO RUN
			- correct price details after player number is changed -- Total Due Now -- Total Due at course --- Taxes? --- Promo Code should apply or not?	TODO		NO RUN
	POST /TeeOff/v1/Reservation/BookReservation					
		Valid Parameters	- Booking Tee time as Registered User	1. Should return Success as True. 2. Status description should be "Tee time is reserved and paid." 3. ContactId should be the same as the OriginatingContactId 4. Booked tee time amount paid should be correct. 5. Booked tee time Due at course should be correct. 6. Booked tee time should be on the correct course.	1. Returns Success as True. 2. Status description is "Tee time is reserved and paid." 3. ContactId is the same as the OriginatingContactId 4. Booked tee time amount paid is correct. 5. Booked tee time Due at courseis correct. 6. Booked tee time is on the correct course.	PASS
			- Booking Tee time as Guest User (registered account)	1. Should return Success as True. 2. Status description should be "Tee time is reserved and paid." 3. ContactId should be the same as the OriginatingContactId 4. Booked tee time amount paid should be correct. 5. Booked tee time Due at course should be correct. 6. Booked tee time should be on the correct course.	1. Returns Success as True. 2. Status description is "Tee time is reserved and paid." 3. ContactId is the same as the OriginatingContactId 4. Booked tee time amount paid is correct. 5. Booked tee time Due at course is correct. 6. Booked tee time is on the correct course.	PASS
			- Booking Tee time as Guest User (Unregistered account)	1. Should return Success as True. 2. Status description should be "Tee time is reserved and paid." 3. ContactId should be the same as the OriginatingContactId 4. Booked tee time amount paid should be correct. 5. Booked tee time Due at course should be correct. 6. Booked tee time should be on the correct course.		NO RUN
		Invalid Parameters	- Booking Tee time but **No** Tee Time associated with current session.	1. Should return Success as False. 2. Message should be "No tee time(s) associated with current session"	1. Returns Success as False. 2. Message is "No tee time(s) associated with current session"	PASS
			- Booking tee time with Invalid Session	1. Should return Success as False. 2. Message should be "Invalid session requested"	1. Returns Success as False. 2. Message is "Invalid session requested"	PASS
	POST /TeeOff/v1/Reservation/CancelReservation					
	- canceling tee time outside cancellation window is successful (example: cancelation window is 36hours, user is able to cancel 48 hours from tee time and not 24 hours from reserved tee time) -*DONE - should not allow to cancel tee time within cancellation window (example: cancelation window is 36hours, user is able to cancel 48 hours from tee time and not 24 hours from reserved tee time) - when tee time canceled is processing cancelation again *DONE - cancelling guest reservation --- a. non-registered guest account --- b. registered guest account & same registered signed in account --- c. registered guest account but different registered signed in account -- check Reservation Status after cancellation Note: optimize test case to book at most 1 tee time for that day to test invalid cancellation 1) If currently no tee time booked for the day, book playdate == currentDate, then get recent reservation to test invalid cancellation 2) if user has reservation playDate == currentDate already, get reservation to test invalid cancellation	Valid Parameters	Cancel Reservation - Successful	1. Should return Success as True 2. Should cancel the correct tee time. 3. Cancelled tee time Should return Cancellation Number	1. Returns Success as True 2. Cancels the correct tee time. 3. Cancelled tee time returns Cancellation Number	PASS
		Invalid Parameters	Cancel Resevation - Reservation was already cancelled.	1. Should return Success as False. 2. Message should be "We're Sorry. The operation could not be completed. Please try again. Or, we're here to help: 1-855-383-3633. (G03)"	1. Returns Success as False. 2. Message is "We're Sorry. The operation could not be completed. Please try again. Or, we're here to help: 1-855-383-3633. (G03)"	PASS
			Cancel Reservation - Wrong Session	1. Should return Success as False. 2. Message should be "We're Sorry. The operation could not be completed. Please try again. Or, we're here to help: 1-855-383-3633. (G03)"	1. Returns Success as False. 2. Message is "We're Sorry. The operation could not be completed. Please try again. Or, we're here to help: 1-855-383-3633. (G03)"	PASS
SearchController						
	GET /TeeOff/v1/Search/SearchAutocomplete					
		Valid Parameters	SearchAutocomplete - City	1. Should return Success as True. 2. Criteria should be the same with criteria passed in request. 3. Class must be a City. 4. Should have only 1 result when search key is a complete city (example: Arizona City, AZ).	1. Returns Success as True. 2. Criteria is the same with criteria passed in request. 3. Class is a City. 4. Returns 1 result: Arizona City, AZ.	PASS
			SearchAutocomplete - Course	1. Should return Success as True. 2. Criteria should be the same with criteria passed in request. 3. Class must be a Course. 4. Should have only 1 result.	1. Return Success as True. 2. Criteria is the same with criteria passed in request. 3. Class is a Course. 4. Returns 1 result	PASS
			SearchAutocomplete - Region	1. Should return Success as True. 2. Criteria should be the same with criteria passed in request. 3. First result should be Arizona as an example. 4. First result Class should be Region.	1. Returns Success as True. 2. First result is Arizona. 3. First result Class is a Region.	PASS
			SearchAutocomplete - Zipcode	1. Should return Success as True. 2. Criteria should be the same with criteria passed in request. 3. Class must be a Zip. 4. Result count should only be 1. (example: Sedona, Arizona) 5. Address must be Sedona, Arizona when searched zipcode is 86351 as an example	1. Returns Success as True. 2. Criteria is the same with criteria passed in request. 3. Class is a Zip. 4. Result count should only be 1. (example: Sedona, Arizona) 5. Address is Sedona, Arizona when search zipcode is 86351	PASS
			SearchAutocomplete - Searching with & (ampersand)	1. Should return Success as True. 2. Should return course names with '&'	1. Returns success as True. 2. Returns criteria/courses with '&' on its name.	PASS
			SearchAutocomplete - Searching with # (number sign)	1. Should return Success as True. 2. Should return course names with '#'	1. Returns success as True. 2. Returns criteria/courses with '#' on its name.	PASS
			SearchAutocomplete - Searching with % (percent)	1. Should return Success as True. 2. Should match content of [criteria] using RegEx. Result is not null.	1. Returns success as True.	PASS
			SearchAutocomplete - Searching with ' (apostrophe)	1. Should return success as True. 2. Should return criteria/courses with apostrophe (') on its name.	1. Returns success as True. 2. Returns criteria/courses with apostrophe (') on its name.	PASS
			SearchAutocomplete - Searching with " (quotation mark)	1. Should return success as True. 2. Should return criteria/courses with quotaion mark (") on its name.	1. Returns success as True. 2. Returns criteria/courses with quotaion mark (") on its name.	PASS
			SearchAutocomplete - Searching with _ (underscore)	1. Should return success as True. 2. Should match content of [criteria] using RegEx. Result is not null.	1. Returns success as True. 2. Should match content of [criteria] using RegEx. Result is not null.	PASS
			SearchAutocomplete - Searching with - (dash)	1. Should return success as True. 2. Should return criteria/courses with dash - on its name.	1. Returns success as True. 2. Returns criteria/courses with dash - on its name.	PASS
			SearchAutocomplete - No Results	1. Should return Success as True. 2. Count must be zero (No Result). 3. Data must be empty.	1. Returns Success as True. 2. Count is zero (No Result). 3. Data is empty.	PASS

External API Test Cases - Automation Documentation

		Assertions				
		Test Cases/Scenarios		**Expected Result**	**Actual Result**	**Status**
UserController						
	POST /TeeOff/v1/User/AuthenticateUser	Verify the following username and password combinations.				
		Invalid Parameters	SearchAutocomplete - Empty Query	1. Should return Success as False. 2. Error should be handled properly.	1. Error is not handled properly. 2. Returns "Object reference not set to an instance of an object." error. Need to add some verbose message like in SearchCourses - Empty Query.	FAIL
	GET /TeeOff/v1/Search/SearchCourses					
	TODO: - update assertion that maps searchCriteria by getting CourseInfo and comparing requested name w/ Course Nar - check returned courses if in the same search region area (RegionId in dbo.Course) -	Valid Parameters	SearchCourses - City	1. Should return Success as True. 2. SearchCriteria should match with value passed to request. 3. SearchTypeId should be 5.	1. SearchCriteria is empty 2. SearchTypeId is empty	FAIL
			SearchCourses - Course	1. Should return Success as True. 2. SearchCriteria should match with value passed to request. 3. SearchTypeId should be 6. 4. Result should only be 1. 5. CourseId should be correct. 6. Latitude should be correct. 7. Longtiude should be correct.	1. SearchCriteria is empty 2. SearchTypeId is empty	FAIL
			SearchCourses - Region	1. Should return Success as True. 2. SearchCriteria should match with value passed to request. 3. SearchTypeId should be 3.	1. SearchCriteria is empty 2. SearchTypeId is empty	FAIL
			SearchCourses - Zipcode	1. Should return Success as True. 2. SearchCriteria should match with value passed to request. 3. SearchTypeId should be 7.	1. SearchCriteria is empty	FAIL
			searching course with # on its name	1. SearchCriteria should match with value passed to request. 2. SearchResult Count is only 1. 3. IsDirectoryOnly should be True.	1. SearchCriteria is empty 2. SearchResult Count is only 1. 3. IsDirectoryOnly is True.	FAIL
			searching course with ampersand & on its name	1. SearchCriteria should match with value passed to request. 2. SearchResult Count is only 1. 3. IsDirectoryOnly should be True.	1. SearchCriteria is empty 2. SearchResult Count is only 1. 3. IsDirectoryOnly is True.	FAIL
			searching course with apostrophe ' on its name	1. SearchCriteria should match with value passed to request. 2. SearchResult Count is only 1. 3. IsDirectoryOnly should be True.	1. SearchCriteria is empty 2. SearchResult Count is only 1. 3. IsDirectoryOnly is True.	FAIL
			searching course with dash - on its name	1. SearchCriteria should match with value passed to request. 2. SearchResult Count is only 1. 3. IsDirectoryOnly should be True.	1. SearchCriteria is empty 2. SearchResult Count is only 1. 3. IsDirectoryOnly is True.	FAIL
			searching course with quotation mark " on its name	1. SearchCriteria should match with value passed to request. 2. SearchResult Count is only 1. 3. IsDirectoryOnly should be True.	1. SearchCriteria is empty 2. SearchResult Count is only 1. 3. IsDirectoryOnly is False.	FAIL
		Invalid Parameters	SearchCourses - No Results	1. Should return Success as False. 2. Count must be zero (No result) 3. Data must be empty. 4. Message should display "No matching results found"	1. Returns Success as False. 2. Count is zero (No result) 3. Data is empty. 4. Message displays "No matching results found"	PASS
			SearchCourses - Empty Query	1. Should return Success as False. 2. Count must be zero (No result) 3. Data must be empty. 4. Message should display "We're Sorry. The operation could not be completed. Please try again. Or, we're here to help: 1-855-383-3633. (G03)"	1. Returns Success as False. 2. Count is zero (No result) 3. Data is empty. 4. Message displays "We're Sorry. The operation could not be completed. Please try again. Or, we're here to help: 1-855-383-3633. (G03)"	PASS
			SearchCourses - Searching with &	1. Should return Success as False. 2. Message should display "No matching results found"	1. Returns Success as False. 2. Message displays "No matching results found"	PASS
			SearchCourses - Searching with #	1. Should return Success as False. 2. Message should display "No matching results found"	1. Returns Success as False. 2. Message displays "No matching results found"	PASS
			SearchCourses - Searching with %	1. Should return Success as False. 2. Message should display "No matching results found"	1.Returns Success as False. 2. Message displays "No matching results found"	PASS
	GET /TeeOff/v1/Search/GetTeeTimesByCourse					
	Known issue: https://jira.ezlinks.com/browse/TMA-167 TODO: Check tee times returned are correct when: - set request 9 or 18 holes - set min / max Distance - set beginTime/endTime - check correct PlayDate -- update assertions to check tee times are within the filter range	Valid Parameters	GetTeeTimesByCourse - Invalid PlayDate	1. Should return Success as True. 2. CourseID should match the requested courseId 3. Course Tee times count is not 0.	1. Returns Success as True. 2. CourseID matches the requested courseId 3. Course Tee times count is not 0.	PASS
			GetTeeTimesByCourse - Filtered	1. Any tee time allows 2 players 2. No tee times priced more than $20.00 (filtered maxPrice @ $20.00) 3. CancellationPeriod should be 24 hours 4. Set Filters to False (response should be false) -- HasDrivingRange -- HasPuttingGreen -- HasChippingArea -- HasGPSAvailable -- HasRentalClubs -- HasLockerRoom -- HasFood -- HasBeverages -- HasRentalShoes -- HasPracticeBunker -- HasGolfShop -- HasNineHolesOnly -- HasLessons -- HasGolfCartIncluded	1. Any tee time allows 2 players 2. No tee times priced more than $20.00 (filtered maxPrice @ $20.00) 3. CancellationPeriod should be 24 hours 4. Filter response as follows: -- No Driving Range -- No Putting Green -- No ChippingArea -- No GPSAvailable -- No RentalClubs -- No LockerRoom -- No Food -- No Beverages -- No RentalShoes -- No PracticeBunker -- No GolfShop -- No NineHolesOnly -- No Lessons -- No GolfCartIncluded	PASS
		Invalid Parameters		1. Should return Success as False. 2. Should return mesage "We're Sorry. The operation could not be completed. Please try again. Or, we're here to help: 1-855-383-3633. (G03)	1. Returns Success as False. 2. Returns mesage "We're Sorry. The operation could not be completed. Please try again. Or, we're here to help: 1-855-383-3633. (G03)	PASS
			GetTeeTimesByCourse - Empty CourseId	1. Should return "Invalid/missing request object" error.	1. Returns "Invalid/missing request object" error.	PASS
			GetTeeTimesByCourse - No Results using past date.	1. Should return Success as True. 2. Data count must be 0.	1. Returns Success as True. 2. Data count is 0.	PASS
	GET /TeeOff/v1/Search/GetTeeTimesDeals					

External API Test Cases - Automation Documentation						
		Assertions				
		Test Cases/Scenarios		Expected Result	Actual Result	Status
UserController						
	POST /TeeOff/v1/User/AuthenticateUser	Verify the following username and password combinations.				
	Check tee times returned are correct when: - set request 9 or 18 holes - set min / max Distance - set beginTime/endTime - check correct PlayDate -- update assertions to check tee times are within the filter ranges	Valid Parameters	GetTeeTimesDeals - Successful	1. Should return Success as True. 2. Deals count must not be 0. 3. Any Teetime Result is a DEAL Time	1. Returns Success as True. 2. Deals count is greater than 0. 3. Any Teetime Result is a DEAL Time (Sample: First Tee time result)	PASS
			GetTeeTimesDeals - Filtered	1. Should return Success as True. 2. Any tee time returned has Cart included. (Filtered to include cart) 3. Any tee time returned is a DEAL Time. 4. No tee times priced lower than $30.00 (Filtered to minPrice @ $30.00) 5. Any tee time allows 2 players (filtered to numberOfPlayers == 2) 6. CancellationPeriod should be 24 hours 7. set Filters to True (response should be True) -- HasDrivingRange -- HasPuttingGreen -- HasChippingArea -- HasRentalClubs -- HasFood -- HasBeverages -- HasPracticeBunker -- HasGolfCartIncluded -- HasGolfShop -- HasLessons 8. Set Filters to False (response should be false) -- No GPSAvailable -- No LockerRoom -- No NineHolesOnly -- No RentalShoes	1. Returns Success as True. 2. Any tee time returned has Cart included. (Sample: First Tee time result) 3. Any tee time returned is a DEAL Time. (Sample: First Tee time result) 4. No tee times returned. 5. Any tee time allows 2 players (Sample: First Tee time result) 6. CancellationPeriod is 24 hours 7. Filter response as follows: -- Has DrivingRange -- Has PuttingGreen -- Has ChippingArea -- Has RentalClubs -- Has Food -- Has Beverages -- Has PracticeBunker -- Has GolfCartIncluded -- Has GolfShop -- Has Lessons 8. Filter response as follows: -- No GPSAvailable -- No LockerRoom -- No NineHolesOnly -- No RentalShoes	PASS
		Invalid Parameters	GetTeeTimesDeals - Invalid PlayDate	1. Should return Success as False. 2. Should return mesage "We're Sorry. The operation could not be completed. Please try again. Or, we're here to help: 1-855-383-3633. (G03)"		PASS
			GetTeeTimesDeals - Empty CourseId	1. Should return "Invalid/missing request object" error.	1. Returns "Invalid/missing request object" error.	PASS
			GetTeeTimesDeals - No Results	1. Should return Success as True. 2. Data count must be 0.	1. Returns Success as True. 2. Data count is 0.	PASS
	GET /TeeOff/v1/Search/GetTeeTimesSummaries					
	Known issue: https://jira.ezlinks.com/browse/TMA-167 TODO: Check tee times returned are correct when: - set request 9 or 18 holes - set min / max Distance - set beginTime/endTime - check correct PlayDate - set other Filters: -- HasDrivingRange -- HasPuttingGreen -- HasChippingArea -- HasGPSAvailable -- HasRentalClubs -- HasLockerRoom -- HasFood -- HasBeverages -- HasRentalShoes -- HasPracticeBunker -- HasGolfShop -- HasNineHolesOnly -- HasLessons -- HasGolfCartIncluded -- update assertions to check tee times are within the filter range	Valid Parameters	GetTeeTimesSummaries - Successful	1. Should return Success as True. 2. TeeTimeSummaryResult count should not be 0.	1. Returns Success as True. 2. TeeTimeSummaryResult count is greater than 0.	PASS
		Invalid Parameters	GetTeeTimesSummaries - Invalid PlayDate	1. Should return Success as False. 2. Should return mesage "We're Sorry. The operation could not be completed. Please try again. Or, we're here to help: 1-855-383-3633. (G03)"	1. Returns Success as False. 2. Returns mesage "We're Sorry. The operation could not be completed. Please try again. Or, we're here to help: 1-855-383-3633. (G03)"	PASS
			GetTeeTimesSummaries - Empty CourseId	1. Should return "Invalid/missing request object" error.	1. Returns "Invalid/missing request object" error.	PASS
	GET /TeeOff/v1/Search/GetCourseTeeTimesSummaries					
	TODO: Check tee times returned are correct when: - set request 9 or 18 holes - set min / max Distance - set beginTime/endTime - check correct PlayDate - set other Filters: -- HasDrivingRange -- HasPuttingGreen -- HasChippingArea -- HasGPSAvailable -- HasRentalClubs -- HasLockerRoom -- HasFood -- HasBeverages -- HasRentalShoes -- HasPracticeBunker -- HasGolfShop -- HasNineHolesOnly -- HasLessons -- HasGolfCartIncluded -- update assertions to check tee times are within the filter range	Valid Parameters	GetCourseTeeTimesSummaries - Successful	1. Should return Success as True. 2. CourseID should match the requested courseId. 3. Data Count must not be 0. 4. CourseName must be correct for CourseId.	1. Returns Success as True. 2. CourseID match the requested courseId. 3. Data Count is not 0. 4. CourseName is correct for CourseId.	PASS
		Invalid Parameters	GetCourseTeeTimesSummaries - Invalid PlayDate	1. Should return Success as False. 2. Should return mesage "We're Sorry. The operation could not be completed. Please try again. Or, we're here to help: 1-855-383-3633. (G03)"	1. Returns Success as False. 2. Returns mesage "We're Sorry. The operation could not be completed. Please try again. Or, we're here to help: 1-855-383-3633. (G03)"	PASS
			GetCourseTeeTimesSummaries - Empty CourseId	1. Should return "Invalid/missing request object" error.	1. Returns "Invalid/missing request object" error.	PASS
			GetCourseTeeTimesSummaries - No Results	1. Should return Success as True. 2. Data count must be 0.	1. Returns Success as True. 2. Data count is 0.	PASS
	GET /TeeOff/v1/Search/GetAllTeeTimesByCourse					
	TODO: Check tee times returned: - correct courseIds - correct playDate	Valid Parameters	GetAllTeeTimesByCourse - Successful	1. Should return Success as True. 2. Data Count must not be 0.	1. Returns Success as True. 2. Data Count is not 0.	PASS
			GetAllTeeTimesByCourse - Invalid PlayDate	1. Should return "The request is invalid" message.	1. Returns "The request is invalid" message.	PASS

External API Test Cases - Automation Documentation						
		Assertions				
		Test Cases/Scenarios		Expected Result	Actual Result	Status
UserController						
	POST /TeeOff/v1/User/AuthenticateUser	Verify the following username and password combinations.				
	- correct playDate	Invalid Parameters	GetAllTeeTimesByCourse - Empty CourseId	1. Should return "Invalid/missing request object" error.	1. Returns "Invalid/missing request object" error.	PASS
			GetAllTeeTimesByCourse - No Results	1. Should return Success as True. 2. Data count must be 0.	1. Returns Success as True. 2. Data count is 0.	PASS
	GET /TeeOff/v1/Search/GetCourseInformation					
		Valid Parameters	GetCourseInformation - Successful	1. Should return Success as True. 2. CourseID should match the requested courseId. 3. Should return correct latitude of course. 4. Should return correct longitude of course. 5. Should return correct CourseName of course.	1. Returns Success as True. 2. CourseID match the requested courseId. 3. Returns correct latitude of course. 4. Returns correct longitude of course. 5. Returns correct CourseName of course.	PASS
		Invalid Parameters	GetCourseInformation - Empty CourseId	1. Should return "The request is invalid" message.	1. Returns "The request is invalid" message.	PASS
	GET /TeeOff/v1/Search/GetCourseIdsByLocation					
	TODO: set distanceInMiles and check that result count is < than no distance set	Valid Parameters	GetCourseIdsByLocation - Successful	1. Should return Success as True. 2. Data Count must not be 0.	1. Returns Success as True. 2. Data Count is greater than 0.	PASS
		Invalid Parameters	GetCourseIdsByLocation - No Results	1. Should return Success as True. 2. Data count must be 0.	1. Returns Success as True. 2. Data count is 0.	PASS
			GetCourseIdsByLocation - Empty Query	1. Should return "The request is invalid" message.	1. Returns "The request is invalid" message.	PASS