ingoing the contraction of the c	Test Scenario(frontend)	Test Steps	Test Data	Expected Results	Actual Results	Pass
with or different management with counter for management with counter for management or per classed with or per for management or per classed with the part of the part of the management or per classed with the part of the management or per classed with the part of the p	creating an account	fill out all the input feilds (email, username, password and confirm password => select your rank => select profile picture => submit	"email: i@gmail.com, username:manny, password:manny457, confirm password:manny457"	creates user and redirect to login page	creates user and redirects to login page	Pass
settle and five questions after out the management of the equation to be careful in the jump (and early all products) and all products and al	login	fill out all the input feilds (email, password) => submit	"email: i@gmail.com, password:manny457,	logs in and redirects to the channels page	logs in and redirects to the channels page	Pass
contain and view registed cids on the specific quantities of claim or high per specific quantities of claim in the readquarter group of claim or the readquarter group of claim in the readquarter group of claim	create and view channels	write out the name of the channel to be created in the input feild => click the create button	" python channel"	creates and displays python channel	creates and displays python channel	pass
His message-inciples disc and se specific question has buthon for the message or regly you want to like an exagge-inciples disc and exagge-inciples disched message-inciples disched court of the message or regly you want to disched message-inciples disched disched court of the message-inciples disched the like or disched court of the message-inciples disched the like or disched court of the message-inciples disched disc	create and view questions	write out the name of the question to be created in the input feild => add a picture optionally => click the create button	"message: how can i learn python" image "image.jpg"	creates and displays inputted question and the imgage	creates and displays inputtedquestion and the imgage	pass
sake message reliques disc sent ne sensoge religion disc sent ne sensoge reliques disc sent ne sensoge religion disc sent ne sensoge reliques disc sent ne sensoge religion disc sent ne sent ne sensoge religion disc sent ne sent ne sent ne sent sent ne sent sent	create and view replies	click on the specific question you want to relyp to => write out the name of the reply to be created in the input feild => add a picture optionally => click the create button	"message: how can i learn python" image "image.jpg"	creates and displays inputted replies and the imgage	creates and displays inputted replies and the imgage	pass
elector desides alter areasy) king or delating a massage — oftic on the specific question like bothom for the message or ready you want to recommend the ordinate or delays the updated the like or disks cannot not be made to go the beauthor of pages to search or message — pages became with the profession pages or conservation and search "port or season and search" "port can it is any plays were not to handle change and update user or to handle change and update user or do handle change and update user or to handle change and update user or do handle change and upda	like messages/replies	click on the specific question like button for the message or reply you want to like	null	diplays the updated like count of the message	diplays the updated like count of the message	pass
select durinet an select durinet and select durinet and select measures and select durinet and select durinet and select measures and select durinet and select measures are all returned select select to the learning page user gets logged out and redirected back to the learning pa	dislike messages/replies	click on the specific question like button for the message or reply you want to dislike	null	diplays the updated the dislke count of the message	diplays the updated dislike count of the message	pass
we and ed profile cited the username on the markest to good the profile page to view at ex- cited canning and updates user updates cited for pages or the handest change and updates user updates cited control to the selector updates cited for pages or the handest change and updates user updates cited for pages or the profile control to the selector updates cited for pages or the profile control to the selector updates cited for pages or the profile control to the selector updates cited for pages or the profile control to the selector updates cited for pages or the profile control to the selector updates cited for pages or the profile control to the selector updates cited for pages or the profile control to the selector updates cited for pages or the profile control to the selector or the	remove like or dislike	after already liking or dialking a message => click on the specific question like button for the message or reply you want to re,ove like or dislike	null	diplays the updated the like or dislke count of the message	diplays the updated the like or dislke count of the message	pass
count of the six bear who lend logged in can access pages and perfect logged or and redirected back to the landing page user gets locked back out and redirected back to the landing page user gets locked back out and redirected back to the landing page page. Test Somativipation Engloyers Test Data Test	search	click search on the navbar to go to the search page => choose whether to search through channels or messages , => input search query	select channels and search "pyhthon" or select messages and search " how can i learn pyhton"	displays "pyhton channel" or displays "how can i learn python "message	displays "pyhton channel" or displays "how can i learn python "message	pass
Test Sommitticitadered) Engogens be unto mhe site user gets kicked back out and redirected back to the landing page of the unit on the site user gets kicked back out and redirected back to the landing page of the unit on the site user gets kicked back out and redirected back to the landing page of the unit of the landing page of the landing page of the unit of the landing page of the landing	view and edit profile	click the username on the navbar to go to the profile page to view it => click onn edit , => fill it out => click submit	username "newusername" rank ="intermediate"	app logs user out to handle change and update user	app logs user out to handle change and update user	pas
Test Semanticitacism (Englorine Test Steps send a POST request to http://locathost.8080/register with the required parameters (usemame, password, email, rank) in the request body. (usemame, password, email, rank) in the request bo	logout	click on logout on the sidebar	null	user gets logged out and redirected back to the landing page	user gets logged out and redirected back to the landing page	pas
send a PCST request to http://localhost.8080/register with the required parameters (username, password, email, rank) in the request body. (username, password, email, rank) in the request body. (both response to ensure that the user is registered successfully. PCST /register PCST /register Send a PCST request to http://localhost.8080/screate with the required parameters (channel/Name, userld, username) (channel/Name, userld, username) Send a PCST request to http://localhost.8080/create with the required parameters (channel/Name, userld, username) (channel/Name, userld, username) Resage is posted successfully. Resage is po	testing if a user who isnt logged in can acess pages	change the url on the site	url : localhost300:/profile	user gets kicked back out and redirected back to the landing page	user gets kicked back out and redirected back to the landing page	pas
Check the response to ensure that the user is registered successfully. (gername, password, email, rank) in the request body. the user is registered successfully. the user is registered successfully. pOST request to http://ccathost:8080/orsate with the email and password you used during registration. successful login channel screated successfully. channel screated successfu	Test Scenario(backend) Endpoints		Test Data	Expected Results	Actual Results	Pa
POST foreate POST request to http://locathosts8080/login with the email and password you used during registration. semal and password you used during registration. channel is created successfully. channel is created successfully. message is posted successfully. set of a POST request to http://locathost.8080/messages to get a list of messages. you receive the list of messages. you receive the list of ressages. you receive the list of regiss. send a POST request to http://locathost.8080/update user of regiss. you receive the list of regiss. the regis	POST /register		(username password email rank) in the request body	the user is registered successfully	the user is registered successfully	
POST /create Send a POST request to http://localhost.8090/create with the required parameters (channel/Name, userid, username) Channel is created successfully. Message is posted successfully.		send a POST request to http://localhost:8080/login with the email and password you used during registration.				
GET request to http://localhouts/8080/messages to get alst of messages. One receive the list of messages. One receive the list of messages. One of POST request to http://localhouts/8080/messages to get alst of messages. One of POST request to http://localhouts/8080/messages to the required parameters (reply, message id, userid, username). (reply, message, id, userid, username). (reduction in the query parameters. (restaged id, userid, username). (reduction in the query parameters. (restaged id, userid, username). (reduction in the query parameters. (restaged id, userid, username). (reduction in the query parameters. (restaged id, userid, username). (reduction in the query parameters. (restaged id, userid, username). (reduction in the query parameters. (restaged id, userid, username). (reduction in the query parameters. (restaged id, userid, username). (reduction in the query parameters. (restaged id	POST/create	Send a POST request to http://localhost:8080/create with the required parameters (channelName, userid, username)	(channelName, userId, username)	channel is created successfully.	channel is created successfully.	
POST foreate replies Send a POST request to http://locathost.8000/create_replies with the required parameters (reply, message_id, userid, username). (reply is posted successfully. (reply is posted	POST /create_message	Send a POST request to http://localhost:8080/create_message with the required parameters (channel_id, content, userid, username).	(channel_id, content, userid, username).	message is posted successfully.	message is posted successfully.	
Send a GET request to http://localhosts.8080/replies to get a list of replies. Send a POST request to http://localhost.8080/replies to get a list of replies. You receive the	GET /messages	a GET request to http://localhost:8080/messages to get a list of messages.	channel_id	you receive the list of messages.	you receive the list of messages.	
POST iupdateuser Send a POST request to http://localhost.8080/updateuser with the required parameters (editusername, edited/userRank) and user_id in the query parameters. The user information is updated successfully. The user is updated successfully. The user is updated successfully. The downwork is processed successfully.			(seeb, manager id consid consens)	the renty is nosted successfully	the reals is posted as exceptible.	
POST fowmore Send a POST request to http://locathost.8080/upvote with the required parameters (messageid, userid). POST fowmore Send a POST request http://locathost.8080/upvote with the required parameters (messageid, userid). POST fowmore Send a POST request http://locathost.8080/upvote with the required parameters (messageid, userid). POST fowmore Send a GET request in http://locathost.8080/upvote to beta users (messageid, userid). Post of Send a GET request in http://locathost.8080/upvote to beta users (messageid, userid). Post of Send a GET request in http://locathost.8080/upvote to beta users (messageid, userid). Post of Send a GET request in http://locathost.8080/upvote to beta users (messageid, userid). Post of Send a GET request in http://locathost.8080/upvote to beta users (messageid, userid). Post of Send a GET request in http://locathost.8080/upvote to beta users (messageid, userid). Post of Send a GET request in http://locathost.8080/upvote to beta users (messageid, userid). Post of Send a GET request in http://locathost.8080/upvote to beta users (messageid, userid). Post of Send a GET request in http://locathost.8080/upvote to beta users (messageid, userid). Post of Send a GET request in http://locathost.8080/upvote to beta users (messageid, userid). Post of Send a GET request in http://locathost.8080/upvote to beta users (messageid, userid). Post of Send a GET request in http://locathost.8080/upvote to beta users (messageid, userid). Post of Send a GET request in http://locathost.8080/upvote to beta users (messageid, userid). Post of Send a GET request in http://locathost.8080/upvote to beta users (messageid, userid). Post of Send a GET request in http://locathost.8080/upvote to beta users (messageid, userid). Post of Send a GET request in http://locathost.8080/upvote to beta users (messageid, userid). Post of Send a GET request in http://locathost.8080/upvote to beta users (messageid, userid). Post of Send a GET request in http://locathost.8080/upvote to beta users (messag	POST/create_replies	Send a POST request to http://localhost:8080/create_replies with the required parameters (reply, message_id, usernd, username).	(reply, message_id, dsend, dsentame).		the reply is posted successibility.	
POST following the Send a POST request http://localhosts8000fdownvole with the required parameters (messageid, userid). Bert Agoput Send a GET request to http://localhosts8000fdownvole with the required parameters (messageid, userid). Bert Agoput Send a GET request to http://localhosts8000fdownvole with the user is logged out. Bert Agoput Send a GET request to http://localhosts8000fdownvole with the user is logged out. Bert Agoput Send a GET request to http://localhosts8000fdownvole with the user is authenticated of the user is logged out. Bert Agoput Send a GET request to http://localhosts8000fdownvole with the user is logged out. Bert Agoput Send a GET request to http://localhosts8000fdownvole with the user is logged out. Bert Agoput Send a GET request to http://localhosts8000fdownvole with the user is logged out. Bert Agoput Send a GET request to http://localhosts8000fdownvole with the user is logged out. Bert Agoput Send a GET request to http://localhosts8000fdownvole with the downvole is processed successfully. Bert Agoput Send a GET request to http://localhosts8000fdownvole with the downvole is processed successfully. Bert Agoput Send a GET request to http://localhosts8000fdownvole with the downvole is processed successfully. Bert Agoput Send a GET request to http://localhosts8000fdownvole with the user is logged out. Bert Agoput Send a GET request to http://localhosts800fdownvole with the user is logged out. Bert Agoput Send a GET request to http://localhosts800fdownvole with the user is logged out. Bert Agoput Send a GET request to http://localhosts800fdownvole with the user is logged out. Bert Agoput Send a GET request to http://localhosts800fdownvole with the user is logged out. Bert Agoput Send a GET request to http://localhosts800fdownvole with the user is logged out. Bert Agoput Send a GET request to http://localhosts800fdownvole with the user is logged out. Bert Agoput Send a GET request to http://localhosts800fdownvole with the user is logged out. Bert Agoput Send a GET requ			1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1			
GET / logout Send a GET request to http://localhosts8080/logout to log out the user is logged out. The user is untertication status is returned. The correct authentication status is returned. The correct authen	GET /replies	Send a GET request to http://localhost:8080/replies to get a list of replies.	message_id	you receive the list of replies.	you receive the list of replies.	
GET rauthUser Send a GET request to http://localhosts.080/authUser to check if the user is authentication status is returned. Send a GET request to http://localhosts.080/authusers retrieves information about all users, messages, and channels. Tetrieves information about all users, messages, and channels.	GET /replies POST /updateuser	Send a GET request to http://localhost:8080/repiles to get a list of replies. Send a POST request to http://localhost:8080/updateuser with the required parameters (editusername, editedUserRank) and user_id in the query parameters.	message_id (editusername, editedUserRank) and user_id in the query parameters.	you receive the list of replies. the user information is updated successfully.	you receive the list of replies. the user information is updated successfully.	
GET /adminusers Send a GET request to http://localhosts.8080/ladminusers retrieve information about all users, messages, and channels. retrieve information about all users, messages, and channels. GET /adminusers Send a GET request to http://localhosts.8080/ladminusers retrieve information about all users, messages, and channels.	GET /replies POST /updateuser POST /upvote	Send a GET request to http://locathosts/8080/replies to get a list of regiles. Send a POST request to http://locathosts/8080/updateuser with the required parameters (editusername, edited/userRank) and user_id in the query parameters. Send a POST request to http://locathosts/8080/upo	message_id (editusername, editedUserRank) and user_id in the query parameters. parameters (messageid, userid).	you receive the list of replies. the user information is updated successfully. Ithe upvote is processed successfully.	you receive the list of replies. the user information is updated successfully. the upvote is processed successfully	
GET / adminusers Send a GET request to https://localhoss.18880/adminusers n/a retrieve information about all users, messages, and channels. GET / adminusers Send a GET request to https://localhoss.18880/adminusers retrieve information about all users, messages, and channels. retrieve information about all users, messages, and channels. retrieve information about all users, messages, and channels.	POST /create_replies GET /replies POST /updateuser POST /upvote POST /downvote GET /logout	Send a GET request to http://localhost.8080/replies to get a list of replies. Send a POST request to http://localhost.8080/updateuser with the required parameters (editusername, edited/UserRank) and user_id in the query parameters. Send a POST request to http://localhost.8080/update	message_id (editusermame, edited/JserfRank) and user_id in the query parameters. parameters (messageid, userid). parameters (messageid, userid).	you receive the list of replies. the user information is updated successfully. the upvote is processed successfully. the downvote is processed successfully	you receive the list of replies. the user information is updated successfully. the upvote is processed successfully the downvote is processed successfully.	
GET /adminmessages Send a GET request to http://locathost:8080/adminmessages set a GET request to http://locathost:8080/adminmessages retrieve information about all users, messages, and channels.	GET /replies POST /updateuser POST /upvote POST /downvote GET /logout	Send a GET request to http://locathoste/808/replies to get a list of replies. Send a POST request to http://locathosts/808/replies to get a list of replies. Send a POST request to http://locathosts/808/lupudateuser with the required parameters (messageid, userid). Send a POST request to http://locathosts/808/lupudateuser with the required parameters (messageid, userid). Send a POST request) http://locathosts/808/lupudateuser/808/lupudateuse	message_id (editusername, edited/user/Rank) and user_id in the query parameters. parameters (messageid, userid). parameters (messageid, userid). ri/a	you receive the list of replies. the user information is updated successfully. the upvote is processed successfully. the downvote is processed successfully the user is logged out.	you receive the list of replies. the user information is updated successfully. the upvote is processed successfully the downvote is processed successfully. the user is logged out.	
	GET /replies POST /updateuser POST /upvote POST /downvote GET /logout GET /authUser	Send a GET request to http://localhost/8080/replies to get a list of replies. Send a POST request to http://localhost/8080/updateuser with the required parameters (editusername, edited/userRank) and user_id in the query parameters. Send a POST request to http://localhost/8080/updateuser with the required parameters (messageid, userid). Send a POST request http://localhost/8080/updat/localhost/8080/up	message_id (editusemame, edited/userfank) and user_id in the query parameters. parameters (messageid, userid). parameters (messageid, userid). n/a n/a	you receive the list of replies. the user information is updated successfully. the uprote is processed successfully. the downvote is processed successfully the downvote is processed successfully the user is logged out. the correct authentication status is returned.	you receive the list of replies. the user information is updated successfully. the uprote is processed successfully the downvote is processed successfully. the user is logged out. the correct authentication status is returned. retrieves information about all users, messages, and channels.	
	GET /replies POST /updateuser POST /upvote POST /downvote	Send a GET request to http://locathost/808/replies to get a list of replies. Send a POST request to http://locathost/808/replies to get a list of replies. Send a POST request to http://locathost/808/lupdateuser with the required parameters (messageid, userid). Send a POST request to http://locathost/808/lupdawnote with the required parameters (messageid, userid). Send a POST request to http://locathost/808/lupdawnote with the required parameters (messageid, userid). Send a GET request to http://locathost/808/lupdawnote with the required parameters (messageid, userid). Send a GET request to http://locathost/808/lupdawnote with the required parameters (messageid, userid). Send a GET request to http://locathost/808/lupdawnote/send/send/send/send/send/send/send/sen	message_id (editusername, edited/user/tank) and user_id in the query parameters. parameters (messageid, userid). n/a n/a n/a	you receive the list of replies. the user information is updated successfully. the uprote is processed successfully. the downvote is processed successfully the downvote is processed successfully the user is logged out. the correct authentication status is returned. retrieve information about all users, messages, and channels.	you receive the list of replies. the user information is updated successfully. the uprote is processed successfully the downvote is processed successfully. the user is logged out. the correct authentication status is returned. retrieves information about all users, messages, and channels.	