

Application Name: TravelWise
Testing Part
Author: Yan Peng

QA TESTING	1
Unit Test	1
Part of Unit Test Report	1
Description of integrated CI work	2
INTEGRATION TEST	2

QA Testing

Unit Test

In this project, I have separated unit tests for controllers and services in order to follow the MVC architecture. Controller unit tests involves testing how controllers handle incoming requests, interact with services, and return the correct responses. Services unit tests encapsulate the business logic of my application and are often responsible for interacting with databases, and external APIs.

Part of Unit Test Report

PASS test/services/auth.test.js					
File	% Stmts	% Branch	% Funcs	% Lines	Uncovered Line #s
All files	96.45	90	96.55	96.92	
api	66.66	50	100	66.66	
db.js	66.66	50	100	66.66	36
api/controllers	97.22	93.75	100	97.22	
auth.js	100	100	100	100	
favorites.js	100	75	100	100	78-79
like.js	100	100	100	100	
qa.js	100	100	100	100	
relationship.js	85.71	100	100	85.71	47-48,86-87
search.js	100	100	100	100	
subscription.js	100	100	100	100	
user.js	96.96	90	100	96.96	29
api/middleware	100	100	100	100	
authenticateToken.js	100	100	100	100	
api/models	84.61	100	66.66	91.66	
post.js	75	100	50	85.71	24
user.js	100	100	100	100	
api/services	96.61	88.98	97.47	97.89	
auth.js	100	100	100	100	
emailServiceManager.js	94.73	100	100	94.73	22
favorites.js	100	100	100	100	
like.js	100	100	100	100	
post.js	93.1	78	93.47	94.4	79,132,241,294-295,302-303,321
qa.js	84.61	0	100	84.61	33-34
relationship.js	100	100	100	100	
search.js	100	100	100	100	
subscribe.js	100	100	100	100	
tagConvertService.js	100	100	100	100	
user.js	100	100	100	100	
Test Suites: 21 passed, 21 total Tests: 139 passed, 139 total Snapshots: 0 total Time: 1.404 s, estimated 2 s					

Description of integrated CI work

I have integrated Jest, the unit testing framework, with GitHub Actions for seamless continuous integration (CI). With this integration, the tests are automatically run whenever code changes are pushed to main branch. This ensures that codebase remains reliable, and any issues are caught early in the development process.

Integration Test

Test ID	Description	Test Case	Test Result
T001	Login verifying that users can successfully log in with valid credentials and that appropriate error messages are displayed for invalid credentials.	1. valid credentials 2. Invalid credentials	Pass
T002	Register: verifying that users can successfully register with valid information and that duplicate registrations or any invalid field if entered are prevented and display relative error messages.	1. valid registration 2. duplicate registration 3. password strength requirements 4. valid/invalid email 5. conform password match or not	Pass
T003	Search: It includes verifying that the search functionality works correctly and that results are filtered based on the selected criteria.	1. search by any keywords 2. search results can be filtered by different criteria. 3. verify search results are accurate and relevant to the user's query.	Pass
T004	Likes: It includes verifying that users can like and unlike content, and that the like count is updated accordingly.	1. registered user can use like. 2. prevent unregistered user use like. 3. users can like or unlike 4. the like count is correctly displayed	Pass
T005	Favorites: It includes verifying that users can add and remove favorites, and that the favorite status is saved and displayed correctly.	1. verify a user can add a post to their favorites list. 2. verify a user can remove a post from their favorites list. 3 verify that user can't add yourself post to favorites list.	Pass
T006	Follower: It includes verifying that users can follow and unfollow other users.	1. verify that a user can follow another user. 2. verify that a user can unfollow another user.	Pass

		3 verify that user can't follow yourself.	
T007	Update account allows users to update their account information, such as their profile picture or bio. It includes verifying that users can update their information successfully and that the changes are reflected in the application.	1. update profile information. 2. update Profile picture. 3. Update email address. 4. verify the upload file type must be image.	Pass
T008	Add Posts: It includes verifying that users can create posts with text and images, and that the posts are displayed correctly.	1. add new post (valid post/invalid post). 2. add post with image: upload image and format. 3. add posts with tags	Pass
T009	Update Posts: It includes verifying that users can edit the content of their posts and that the changes are saved and displayed correctly.	1. verify the user can edit the post first. 2. change post image: update image and check image format. 3. edit post themes and category.	Pass
T010	Comments: Verify that users can view, add and reply to comments to posts or content.	1. add comment. 2. other users can view the comment. 3. user can reply to the comment.	Pass
T011	Logout: provide a logout option to allow users to securely log out of their accounts.	1. verify a user can successfully log out of account. 2. verify after logging out, the user session is ended, and they redirected to home page. 3. verify clicking logout button triggers the logout process. 4. verify that after logging out, users can't access protected pages or perform actions that require authentication.	Pass