Cyber Justitia: Testing Document

1. Introduction

Purpose

This is a testing document detailing the steps we took to test Cyber Justitia.

Scope

Our testing process covered unit testing, as well as headless front-end browser testing using Playwright.

2. Overview of Testing Methodology

We employed automated tests to maintain code quality and verify the functionality of various components of Cyber Justitia. Automated testing helps in identifying and resolving issues efficiently and ensures that new changes do not introduce bugs.

Types of Testing

Unit Testing: We focused on unit testing to validate the smallest parts of our application in isolation. Each unit test checks a specific piece of functionality to ensure it works as intended.

Tools

Django Test Framework: We used the Django Test Framework to write and execute our unit tests. This framework integrates with our application, providing tools to test models, views, and forms.

Playwright: We used Playwright for front-end testing all of the HTML templates. Playwright tests run in headless mode, meaning the process is streamlined as there is no need to open browser windows.

3. Test Environment

Local Development Setup: All tests were conducted in a local development environment configured with Django.

4. Test Cases and Results

The following table contains the more important test cases whose functions failed the initial tests and had to be debugged.

Category	ID	Description	Expected Result	Actual Result (Initial)	Actual Result (Final)

TCM	8	Test creating a message with text exceeding the maximum length	ValidationError raised	ValidationError not raised	ValidationError raised
TFV	21	Test downvoting a comment	Downvote created successfully, stored in the database	Downvote not created, AssertionError: False is not true	Downvote created successfully
TFV	18	Test downvoting a post	Downvote created successfully, stored in the database	Downvote not created, AssertionError: False is not true	Downvote created successfully
TUM	31	Test that start_date cannot be in the future (Education Model)	ValidationError raised	ValidationError not raised	ValidationError raised
TUM	40	Test that start_date cannot be in the future (Employment Model)	ValidationError raised	ValidationError not raised	ValidationError raised
TUV	24	Test deactivating the account when the checkbox is checked	Status code 302 (redirect to index)	Status code 200, AssertionError: 200 != 302	Status code 302 (redirect to index)
TFV	9	Test creating a comment with blank data	Proper error handling and response	ValueError: The view didn't return an HttpResponse object	Proper error handling and response
TUM	31	Test that start_date cannot be in the future (Education Model)	ValidationError raised	ValidationError not raised	ValidationError raised
TUM	40	Test that start_date cannot be in the future (Employment Model)	ValidationError raised	ValidationError not raised	ValidationError raised

This table contains an overview of every single test in the project.

Category	ID	Description	Pass/Fail
TCM	2	Test creating a session with a valid user.	Pass
TCM	3	Test creating a session without specifying a user.	Pass
TCM	4	Test the string representation of a session.	Pass
TCM	5	Test creating a message with valid data.	Pass
TCM	6	Test creating a message without specifying a session.	Pass
TCM	7	Test creating a message with empty text.	Pass
TCM	8	Test creating a message with text exceeding the maximum length.	Pass
TCM	9	Test creating a message with an invalid role.	Pass
TCM	10	Test creating messages with different roles.	Pass
TCM	11	Test retrieving messages ordered by creation date.	Pass
TCM	12	Test that deleting a session cascades and deletes its messages.	Pass
TCM	13	Test the constraints and metadata of the Session model.	Pass
TCM	14	Test the constraints and metadata of the Message model.	Pass
TCM	15	Test creating multiple sessions for the same user.	Pass
TCM	16	Test creating multiple messages in a single session.	Pass
TCM	17	Test creating a message that includes emojis.	Pass
TCM	18	Test updating the text of an existing message.	Pass
TCM	19	Test creating a message with text at the maximum length boundary.	Pass
TCM	20	Test creating and retrieving a session by its creation date.	Pass
TCM	21	Test the string representation of a session with a long username.	Pass
TCV	2	Test accessing the chatbot home page.	Pass
TCV	3	Test accessing the chatbot home page as an authenticated user.	Pass
TCV	4	Test accessing a chatbot session.	Pass
TCV	5	Test accessing a non-existent chatbot session.	Pass
TCV	6	Test accessing another user's chatbot session.	Pass
TCV	7	Test processing a chat message.	Pass
TCV	8	Test processing a chat message when not authenticated.	Pass

TCV	9	Test processing a chat message with an invalid request method.	Pass
TCV	10	Test creating a new session.	Pass
TEH	1	Test the 400 error handler.	Pass
TEH	2	Test the 403 error handler.	Pass
TEH	3	Test the 404 error handler.	Pass
TEH	4	Test the 500 error handler.	Pass
TEH	5	Test the 503 error handler.	Pass
TFF	1	Test creating a post with valid data.	Pass
TFF	2	Test creating a post with an empty title.	Pass
TFF	3	Test creating a post with empty text.	Pass
TFF	4	Test creating a post with a title exceeding the maximum length.	Pass
TFF	5	Test creating a post with text exceeding the maximum length.	Pass
TFF	6	Test creating a comment with valid data.	Pass
TFF	7	Test creating a comment with an empty comment field.	Pass
TFF	8	Test creating a comment with text exceeding the maximum length.	Pass
TFM	2	Test creating a post with valid data.	Pass
TFM	3	Test creating a post with an empty title.	Pass
TFM	4	Test creating a post with a title exceeding the maximum length.	Pass
TFM	5	Test creating a post with empty text.	Pass
TFM	6	Test creating a post with text exceeding the maximum length.	Pass
TFM	7	Test that each post has a unique slug.	Pass
TFM	8	Test deleting a post.	Pass
TFM	10	Test creating a comment with valid data.	Pass
TFM	11	Test creating a comment with an empty text.	Pass
TFM	12	Test creating a comment with text exceeding the maximum length.	Pass
TFM	13	Test deleting a comment.	Pass
TFM	15	Test creating a PostVote with valid data.	Pass
TFM	16	Test creating a PostVote without specifying a user.	Pass
TFM	17	Test creating a PostVote without specifying a post.	Pass
TFM	18	Test creating a duplicate PostVote.	Pass

TFM	20	Test creating a CommentVote with valid data.	Pass
TFM	21	Test creating a CommentVote without specifying a user.	Pass
TFM	22	Test creating a CommentVote without specifying a comment.	Pass
TFM	23	Test creating a duplicate CommentVote.	Pass
TFM	24	Test creating a Comment downvote.	Pass
TFM	25	Test creating a Post downvote.	Pass
TFM	26	Test creating CommentVotes on multiple comments.	Pass
TFM	27	Test creating PostVotes on multiple posts.	Pass
TFV	2	Test accessing the forums page.	Pass
TFV	3	Test viewing a post detail page.	Pass
TFV	4	Test viewing a non-existent post.	Pass
TFV	5	Test creating a post with valid data.	Pass
TFV	6	Test creating a post with blank data.	Pass
TFV	7	Test creating a post when not logged in.	Pass
TFV	8	Test creating a comment with valid data.	Pass
TFV	9	Test creating a comment with blank data.	Pass
TFV	10	Test creating a comment when not logged in.	Pass
TFV	11	Test deleting a post as the post owner.	Pass
TFV	12	Test deleting a post as a staff member.	Pass
TFV	13	Test deleting a post as a non-owner and non-staff member.	Pass
TFV	14	Test deleting a comment as the comment owner.	Pass
TFV	15	Test deleting a comment as a staff member.	Pass
TFV	16	Test deleting a comment as a non-owner and non-staff member.	Pass
TFV	17	Test upvoting a post.	Pass
TFV	18	Test downvoting a post.	Pass
TFV	19	Test voting on a post with an invalid form.	Pass
TFV	20	Test upvoting a comment.	Pass
TFV	21	Test downvoting a comment.	Pass
TFV	22	Test voting on a comment with an invalid form.	Pass
TFV	23	Test creating a post with whitespace data.	Pass
TUF	2	Test updating user details with valid data.	Pass
TUF	3	Test updating user details with an empty first name.	Pass

TUF	4	Test updating user details with an empty last name.	Pass
TUF	5	Test updating user details with an empty email.	Pass
TUF	6	Test updating user details with a first name exceeding max length.	Pass
TUF	7	Test updating user details with a last name exceeding max length.	Pass
TUF	8	Test updating user details with a duplicate email.	Pass
TUF	9	Test updating the password with valid data.	Pass
TUF	10	Test updating the password with an incorrect old password.	Pass
TUF	11	Test updating the password with non-matching new passwords.	Pass
TUF	12	Test updating the password with the new password same as the old password.	Pass
TUF	13	Test updating the description with valid data.	Pass
TUF	14	Test deactivating the account with the checkbox checked.	Pass
TUF	15	Test deactivating the account without checking the checkbox.	Pass
TUF	16	Test updating the flair with valid data.	Pass
TUF	17	Test updating the flair with an empty value.	Pass
TUF	18	Test updating the flair with a value exceeding max length.	Pass
TUF	20	Test form validation with valid data (Employment).	Pass
TUF	21	Test form validation with missing company field.	Pass
TUF	22	Test form validation with missing position field.	Pass
TUF	23	Test form validation with missing start date field.	Pass
TUF	24	Test form validation with a start date set in the future (Employment).	Pass
TUF	25	Test form validation with an end date before the start date (Employment).	Pass
TUF	27	Test form validation with valid data (Education).	Pass
TUF	28	Test form validation with missing school name field.	Pass
TUF	29	Test form validation with missing degree field.	Pass
TUF	30	Test form validation with missing start date field.	Pass
TUF	31	Test form validation with a start date set in the future (Education).	Pass

TUF	32	Test form validation with an end date before the start date (Education).	Pass
TUF	34	Test form validation with valid data.	Pass
TUF	35	Test form validation with missing reason for banning.	Pass
TUF	36	Test form validation with missing confirmation for ban.	Pass
TUF	37	Test form validation for already banned user.	Pass
TUF	38	Test form validation for banning an admin user.	Pass
TUF	39	Test saving a banned user.	Pass
TUM	2	Test that the user is created correctly with the provided attributes.	Pass
TUM	3	Test the _str_ method of the user model.	Pass
TUM	4	Test that the email field enforces uniqueness.	Pass
TUM	5	Test that an invalid email format raises a ValidationError.	Pass
TUM	6	Test that first_name and last_name fields cannot be blank.	Pass
TUM	7	Test that the username field cannot be blank.	Pass
TUM	8	Test that the username field enforces uniqueness.	Pass
TUM	9	Test that the is_banned field defaults to False.	Pass
TUM	10	Test that the max length constraints are enforced for various fields.	Pass
TUM	11	Test that a username with special characters raises a ValidationError.	Pass
TUM	12	Test that a username with only whitespace raises a ValidationError.	Pass
TUM	13	Test that first_name and last_name fields with only whitespace raise a ValidationError.	Pass
TUM	14	Test that an email with leading or trailing whitespace raises a ValidationError.	Pass
TUM	15	Test that an invalid password raises a ValidationError.	Pass
TUM	17	Test creating a professional user with a flair.	Pass
TUM	18	Test that a blank flair raises a ValidationError.	Pass
TUM	19	Test that the flair field allows a maximum length of 100 characters.	Pass
TUM	20	Test that a flair longer than 100 characters raises a ValidationError.	Pass
TUM	21	Test updating the flair of a professional user.	Pass

TUM	22	Test creating a professional user with a valid reason_banned.	Pass
TUM	23	Test that a blank reason_banned is allowed.	Pass
TUM	25	Test creating a valid education record.	Pass
TUM	26	Test that school_name field cannot be blank.	Pass
TUM	27	Test that degree field cannot be blank.	Pass
TUM	28	Test that start_date field cannot be blank.	Pass
TUM	29	Test that school_name field cannot exceed max length of 100 characters.	Pass
TUM	30	Test that degree field cannot exceed max length of 100 characters.	Pass
TUM	31	Test that start_date cannot be in the future (Education).	Pass
TUM	32	Test that end_date field can be blank (Education).	Pass
TUM	34	Test creating a valid employment record.	Pass
TUM	35	Test that company field cannot be blank.	Pass
TUM	36	Test that position field cannot be blank.	Pass
TUM	37	Test that start_date field cannot be blank.	Pass
TUM	38	Test that company field cannot exceed max length of 100 characters.	Pass
TUM	39	Test that position field cannot exceed max length of 100 characters.	Pass
TUM	40	Test that start_date cannot be in the future (Employment).	Pass
TUM	41	Test that end_date field can be blank (Employment).	Pass
TUV	2	Test that the index page renders correctly.	Pass
TUV	3	Test that an unauthenticated user can access the register page.	Pass
TUV	4	Test that an authenticated user is redirected from the register page.	Pass
TUV	5	Test that a new standard user can register with valid data.	Pass
TUV	6	Test that a new professional user can register with valid data.	Pass
TUV	7	Test that a weak password prevents registration.	Pass
TUV	8	Test that non-matching passwords prevent registration.	Pass
TUV	9	Test that an existing username prevents registration.	Pass
TUV	10	Test that an existing email prevents registration.	Pass
TUV	11	Test that an unauthenticated user can access the login page.	Pass

TUV	12	Test that an authenticated user is redirected from the login page.	Pass
TUV	13	Test that a user can log in with valid credentials.	Pass
TUV	14	Test that logging in with invalid credentials fails.	Pass
TUV	15	Test that an authenticated user can log out.	Pass
TUV	17	Test that the dashboard page is accessible.	Pass
TUV	18	Test updating user details with valid data.	Pass
TUV	19	Test that updating first name with a blank value fails.	Pass
TUV	20	Test that updating last name with a blank value fails.	Pass
TUV	21	Test updating the password with valid data.	Pass
TUV	22	Test that changing the password with mismatched new passwords fails.	Pass
TUV	23	Test that changing the password with blank fields fails.	Pass
TUV	24	Test deactivating the account when the checkbox is checked.	Pass
TUV	25	Test that deactivating the account without checking the checkbox fails.	Pass
TUV	26	Test updating the user description with valid data.	Pass
TUV	27	Test updating the user flair with valid data.	Pass
TUV	28	Test that updating the flair with a blank value fails.	Pass