1. Test Cases for the File-Sharing System

<u>Authentication & Authorization:</u>

- Ops User Login with Valid Credentials: Verify login succeeds for an Ops user with correct username and password
- Ops User Login with Invalid Credentials: Ensure login fails with an error message for incorrect credentials
- Client User Signup: Verify a client user can sign up successfully and receives an encrypted URL as a response
- Client User Login with Valid Credentials: Confirm login works correctly for a client user with correct credentials
- Role-Based Access: Ensure only Ops users can upload files and only client users can download files

<u>File Handling:</u>

- Valid File Upload by Ops User: Verify that an Ops user can upload files with extensions .pptx, .docx, or .xlsx
- Invalid File Type Upload Attempt by Ops User: Ensure attempts to upload unsupported file types result in an error message
- File Download by Client User: Confirm that a client user receives a valid, encrypted download URL when accessing a file
- Unauthorized File Download Attempt: Verify the system denies file download requests from non-client users or if the URL has expired

<u>Email Verification & URL Handling:</u>

- Email Verification: Verify that a client user receives a verification email with a confirmation link after signup
- Encrypted URL Expiry: Test that an encrypted URL expires after a specified time or usage limit to ensure file security
- Invalid URL Access: Check that attempts to access expired or malformed URLs are denied with an error message

<u>File Listing:</u>

- List Files for Client User: Confirm that a client user can list all files uploaded by the Ops user
- File Listing Restriction for Ops User: Verify that the Ops user does not have access to the file listing API meant for client users

2. Deployment Plan for Production

For deployment, use a reliable cloud provider like AWS for hosting, MySQL on a cloud-managed database, and secure storage like Amazon S3 for file uploads. Secure the environment with HTTPS and SSL, use environment variables for sensitive data, and monitor with tools like AWS CloudWatch. Automate testing and deployment with CI/CD tools like GitHub Actions or Jenkins for streamlined production management.