

TEST PLAN FOR THREADIT

Note that you can refine your testing plan as the project development goes. Keep the change log as follow:

ChangeLog

Version	Change Date	By	Description
version number	Date of Change	Name of person who made changes	Description of the changes made
1	09/02/2023	Nathan Poppe	Create initial test plan

1 Introduction

1.1 Scope

Scope defines the features, functional or non-functional requirements of the software that **will be** tested.

Functional Requirements:

1. User Profiles
2. Create and Interact with Threads
3. Organize Threads
4. Home Feed
5. Spool Suggestions (Optional)

Non-Functional Requirements:

1. Load Testing

1.2 Roles and Responsibilities

Name	Net ID	GitHub username	Role
Anthony Phung	phunga1	awphung	Full-Stack
Cody Wallbridge	wallbric	CodyWallbridge	Backend, Testing
Nathan Poppe	poppen1	nathanpoppe	Full-Stack, Testing
Noah Curoe	curoen	noahc3	Full-Stack, CI/CD

2 Test Methodology

2.1 Test Levels

Core Feature 1: User Profiles

Unit/Integration Test Summary:

- User with unique username and email and valid password can register
- User with registered account and correct credentials can login
- User with non-unique username or email can not register
- User with an invalid length password can not register
- Logged-in user can view user settings
- Logged-in user can customize user settings
- Non-logged-in user can not view or customize user settings

Acceptance Test Summary:

Register/Login:

1. Click Register button on sidebar
2. Enter valid email, username, and password
3. Click Register
4. Click Logout button on sidebar
5. Click Login button on sidebar
6. Enter the same valid credentials
7. Click Login

Customize User Settings:

1. Click Login button on sidebar
2. Enter valid login credentials
3. Click User Settings button on sidebar
4. Change User Settings
5. Click Home button on sidebar
6. Ensure changes have taken effect

Core Feature 2: Create And Interact With Threads

Unit/Integration Test Summary:

- Logged-in user can create a thread with a valid title, content, and topic
- Logged-in user can not create a thread with an invalid title, content, or topic
- Non-logged-in user can not create a thread
- Logged-in user can edit and delete a thread they have created
- User can not edit or delete a thread they have not created
- Logged-in user can comment on a thread with a non-empty comment
- Non-logged-in user can not comment on a thread
- Logged-in user can stitch (upvote) or rip (downvote) a thread
- Non-logged-in user can not stitch or rip a thread
- User can share a link to a thread

Acceptance Test Summary:

Create/Edit A Thread:

1. Click Login button on sidebar
2. Enter valid login credentials
3. Search for a spool in the search bar
4. Click on a spool from the dropdown to enter that spool
5. Click the Post Thread button at the top of the spool
6. Enter a valid title, content, and topic
7. Click Post Thread
8. Ensure that the spool page includes the new thread
9. Click the thread to view it
10. Click the Edit Thread button
11. Make changes to the thread title and content
12. Click the Save Changes button

13. Ensure that the changes are saved
14. Click the Delete Thread button
15. Ensure that the thread is deleted

Interact With A Thread:

1. Click Login button on sidebar
2. Enter valid login credentials
3. Click a thread on the home page
4. Stitch the thread
5. Rip the thread
6. Click the Comment button
7. Enter comment content
8. Click Post Comment
9. Ensure that the comment was created

Core Feature 3: Organize Threads

Unit/Integration Test Summary:

- Logged-in user can create a spool with a valid name
- Logged-in user can not create a spool with an invalid name
- Non-logged in user can not create a spool
- Logged-in user can spin (subscribe) to a spool
- Non-logged-in user can not spin a spool
- Spool creator can edit a spool
- Spool creator can allow other users to edit that spool

Acceptance Test Summary:

Create/Modify a Spool:

1. Click Login button on sidebar
2. Enter valid login credentials
3. Click Create Spool button on sidebar
4. Enter a valid name
5. Click Create Spool
6. Ensure that you are redirected to the new spool page
7. Click the Edit Spool button
8. Enter a new spool name
9. Enter a username to give that user special privileges over the spool
10. Click the Finish Edit button

Subscribe to a Spool:

1. Click Login button on sidebar
2. Enter valid login credentials
3. Search for a spool in the search bar
4. Click on a spool from the dropdown to enter that spool
5. Click the Spin button to subscribe to the spool
6. Ensure that a new button appears in the sidebar for that spool

Core feature 4: Home Feed

Unit/Integration Test Summary:

- Logged-in user can get list of feeds from their spools
- Non-logged-in user can get default list of feeds
- User can sort and filter their feed

Acceptance Test Summary:

Home Feed:

1. Click Home button on sidebar
2. Click Sort Feed dropdown and select Most Popular
3. Ensure home feed is sorted by most stitches
4. Click Login button on sidebar
5. Enter valid login credentials
6. Ensure home feed contains threads from subscribed spools

2.2 Test Completeness

Criteria for Test Completeness:

- 100% code coverage (all the source code should be covered by test cases)
- All manual and automated test cases should be executed
- No bugs or failed tests should exist at the end of a sprint

3 Resource & Environment Needs

3.1 Testing Tools

The following tools are being used in this project for testing:

- Automation: GitHub Actions, Docker
- Bug Tracking: Jira
- Backend Testing: xUnit
- Frontend Testing: TBD
- Load Testing: TBD

3.2 Test Environment

The following software is required:

- WSL
- Docker
- Postgres
- Node
- .NET SDK

4 Terms/Acronyms

Make a mention of any terms or acronyms used in the project

TERM/ACRONYM	DEFINITION
API	Application Program Interface
AUT	Application Under Test
SDK	Software Development Kit
WSL	Windows Subsystem for Linux