AMARANTH

REDEFINING STREAMS, PRESERVING DREAMS

HASSAN ISMAIL | REHAL SAEED | SAMRA MASHAAM

Amaranth

INTRODUCTION

Video streaming platform

No midroll ads

Creators earn through donations and subscriptions

USER STORIES

As a viewer, I want to search for a topic and see matching results so that I can pick a video to watch.

Acceptance Criteria:

Given the viewer is on the platform's search page,
When they enter a search query and submit it,
Then the system should return a list of videos relevant to the query.
And the results should be sorted by relevance by default.

USER STORIES

As a viewer, I want to leave a like or dislike on a video so that I can quickly show my thoughts on it.

Acceptance Criteria:

Given the viewer is watching a video,
When they click the like or dislike button,
Then the system should register the reaction.
And the system should display that change.

USER STORIES

As a viewer, I want to leave a comment on a video so that I can express my opinion on it.

Acceptance Criteria:

Given the viewer is watching a video and is logged in,
When they enter a comment and submit it,
Then the system should display the comment under the video.
And the comment should be visible to other users.

USER STORIES

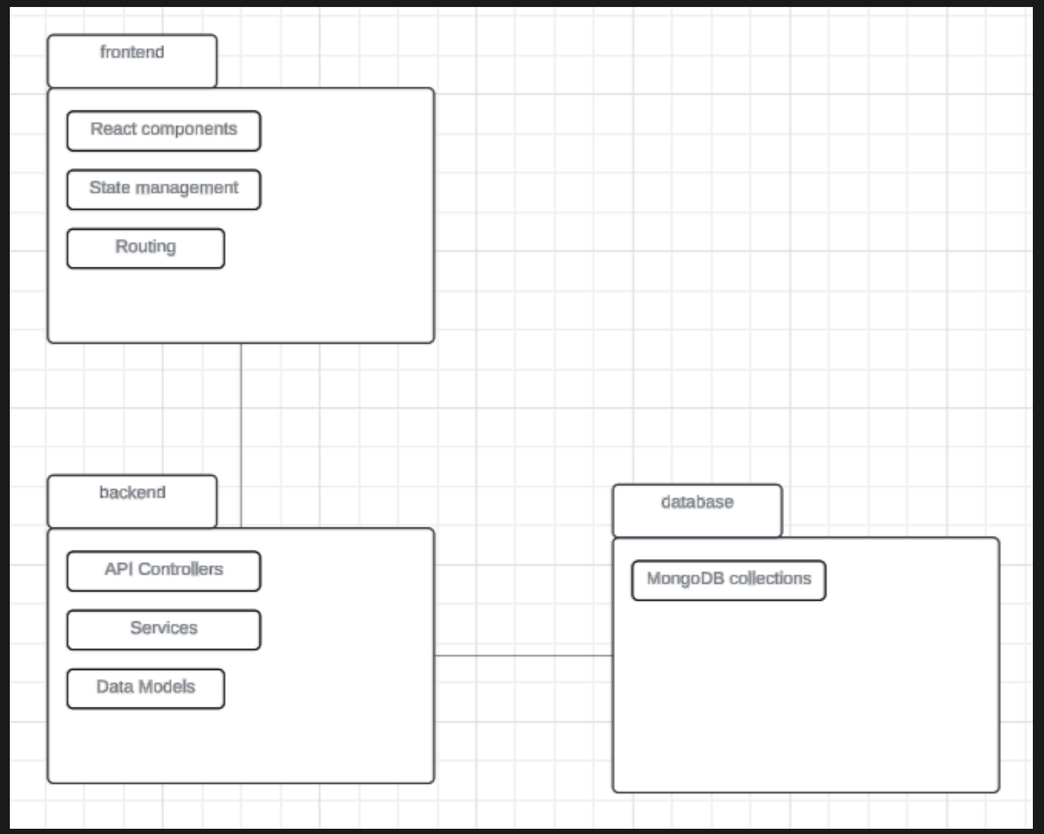
As a content creator, I want to upload a video so that the viewers can watch it.

Acceptance Criteria:

Given a content creator is logged in,
When they upload a video with necessary details,
Then the system should process and publish the video.
And it should be available to watch for others

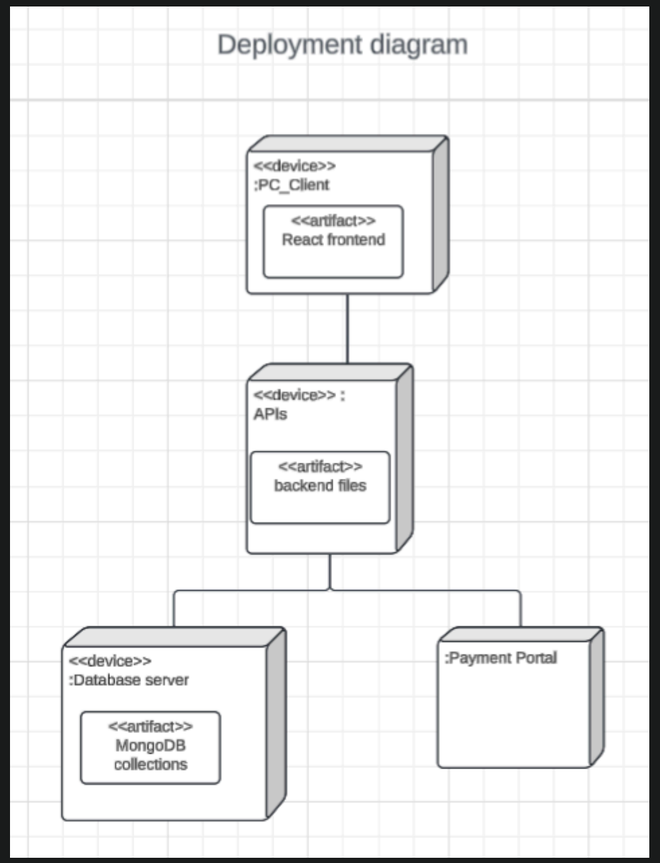
Amaranth

PACKAGE DIAGRAM



Amaranth

DEPLOYMENT DIAGRAM



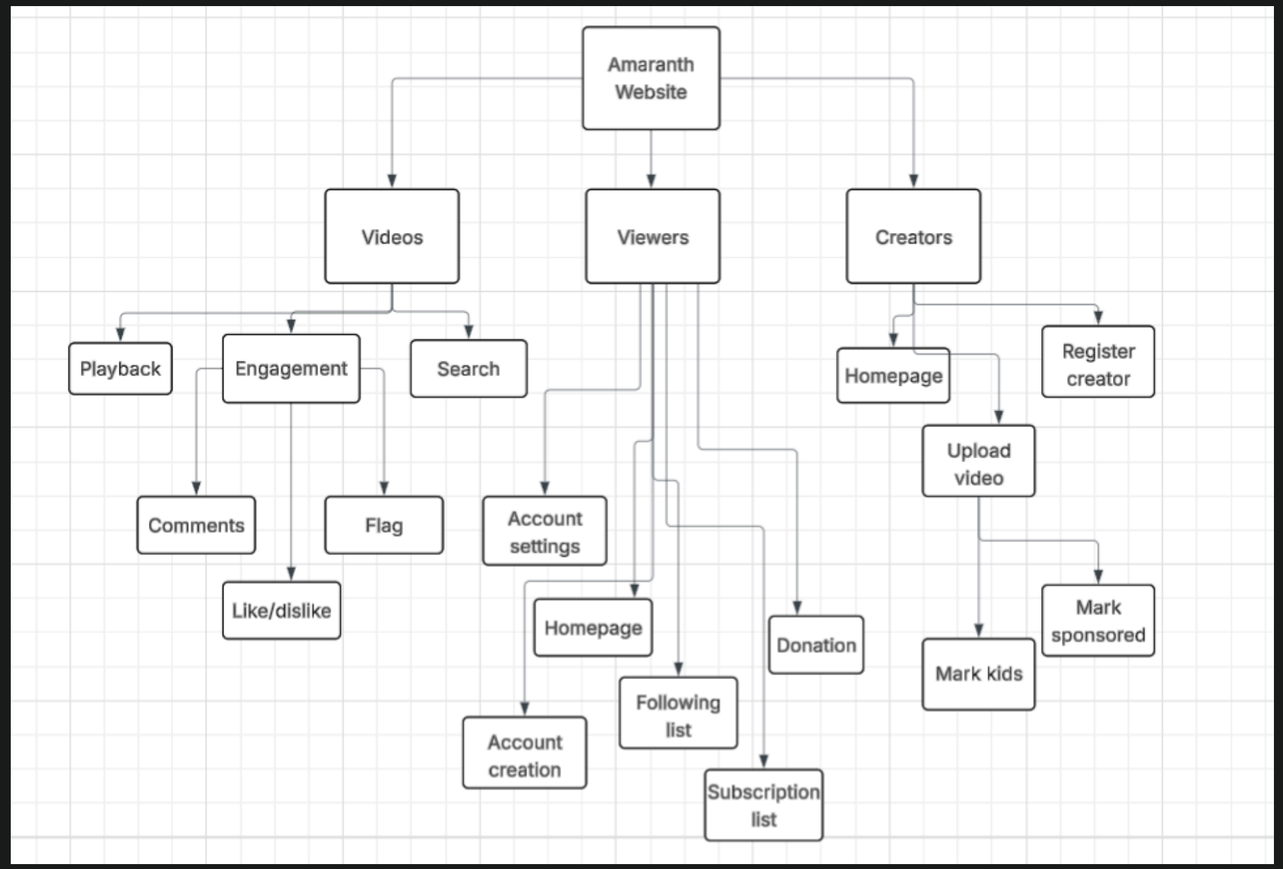
ARCHITECTURE STYLES

Layered Architecture: The frontend, backend, and database layers are all separated and depend on each other in a layered manner. The frontend cannot connect directly to the database. This allows for better modularity and gives a clear separation of concerns.

Client-server architecture: The frontend (React) acts as a client and makes requests to the backend (Express) which acts as the server. This works well because the data can be stored centrally and different types of clients can be handled.

Amaranth

WORK BREAKDOWN STRUCTURE



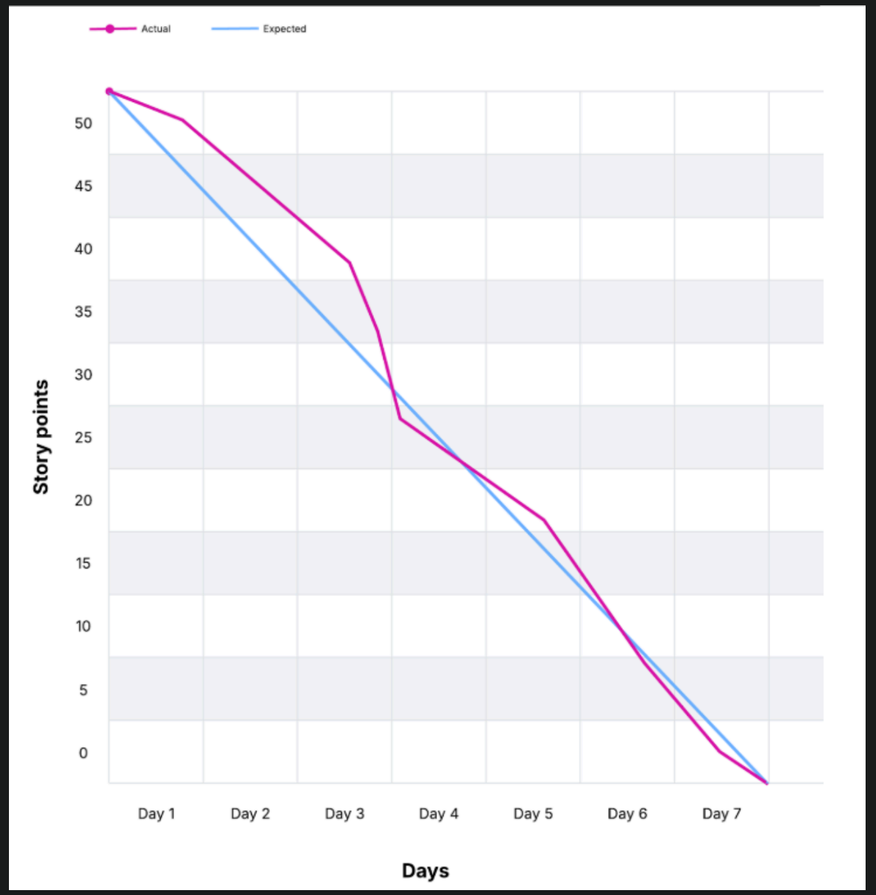
Amaranth

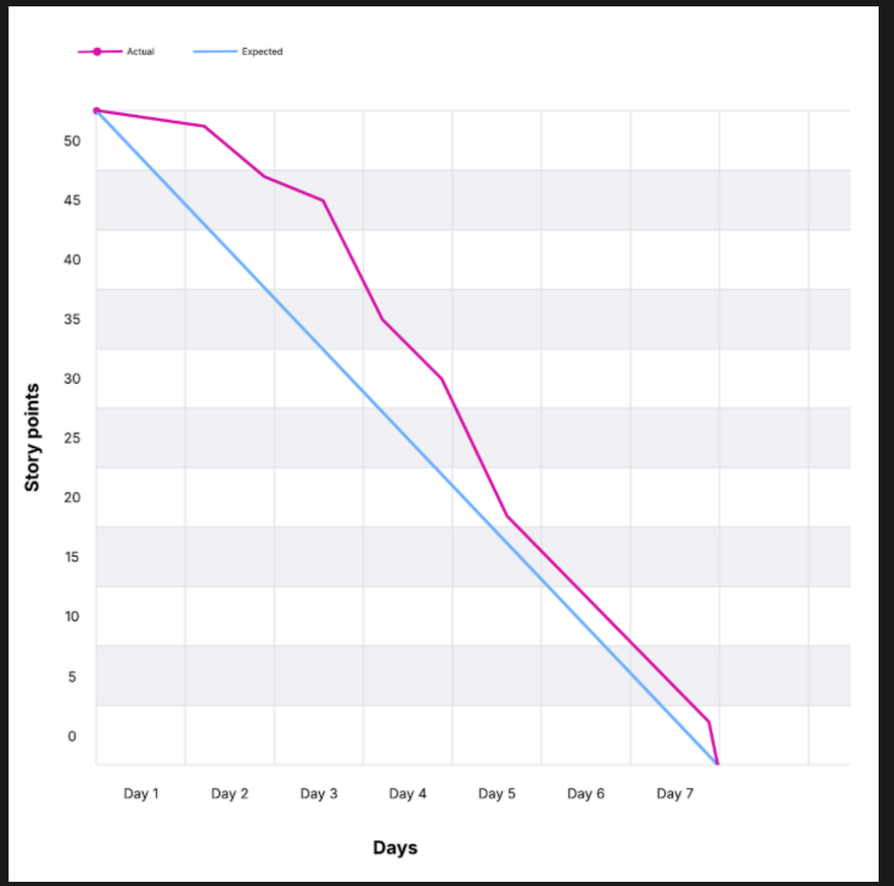
GANTT CHART

Amaranth Gantt	start	end	0h	100%
Rehal's Tasks	**20/03/25**	**27/04/25**	**0h**	**100%**
2. Like/dislike video (frontend)	20/03	21/03	0	100%
3. Leave comment (frontend)	20/03	21/03	0	100%
7. Register as creator (frontend)	21/03	22/03	0	100%
12. Flag content (frontend)	21/03	22/03	0	100%
21. Control video playback	20/03	21/03	0	100%
18. Like/dislike video (backend)	26/04	26/04	0	100%
19. Leave comment (backend)	27/04	27/04	0	100%
1. Search Video	26/04	27/04	0	100%
28. Flag content (backend)	27/04	27/04	0	100%
Hassan's Tasks	**21/03/25**	**27/04/25**	**0h**	**100%**
5. Edit following (frontend)	21/03	23/03	0	100%
10. Subscribe (frontend)	21/03	22/03	0	100%
11. Review subscriptions (frontend)	22/03	23/03	0	100%
16. Mark as sponsored	25/04	26/04	0	100%
15. Upload video	26/04	27/04	0	100%
17. Mark rating	26/04	26/04	0	100%
21. Edit following (backend)	25/04	26/04	0	100%
27. Review subscriptions (backend)	25/04	27/04	0	100%
26. Subscribe (backend)	26/04	27/04	0	100%
20. Follow creator (backend)	27/04	27/04	0	100%
Samra's Tasks	**21/03/25**	**27/04/25**	**0h**	**100%**
6. Register as viewer (frontend)	21/03	21/03	0	100%
8. Donate to creator (frontend)	22/03	22/03	0	100%
9. Account settings (frontend)	21/03	21/03	0	100%
20. Register as kid (frontend)	21/03	21/03	0	100%
4. Follow creator (frontend)	23/03	23/03	0	100%
29. Register as kid (backend)	27/04	27/04	0	100%
23. Register as creator (backend)	26/04	27/04	0	100%
22. Register as viewer (backend)	26/04	26/04	0	100%
24. Donate to creator (backend)	26/04	27/04	0	100%
25. Account settings (backend)	27/04	27/04	0	100%

Amaranth

BURNDOWN CHARTS





BLACK BOX TESTS

Test case 1: TC_VIDEO_01
Feature: Video player
Priority: High
Test type: Function
Equivalence class: All videos from any creator.
Steps:

1. Navigate to a video page
2. Verify the video starts within a second
3. Click the skip buttons and verify it skips forward and backward by 10 seconds
4. Click the pause/play button and verify the video pauses/plays accordingly

Expected Results:
Video loads with pause, play, and skip buttons.
Skip buttons skip 10 seconds each.
Video pauses and plays when required.

Actual results: Pass
Edge cases: Video of 1 second, video of 1 hour.

BLACK BOX TESTS

Test case 2: TC_LK_02
Feature: Like/dislike system
Test type: Functional
Priority: High
Equivalence class: All registered users like/dislike feedback
Steps:

1. Navigate to a video
2. Click the like (thumbs up) button
3. Refresh page or navigate back to the page
4. Repeat for dislike

Expected results:
Count goes up when clicked
Count goes back down clicked again, or when other button is clicked
Count stays after refresh

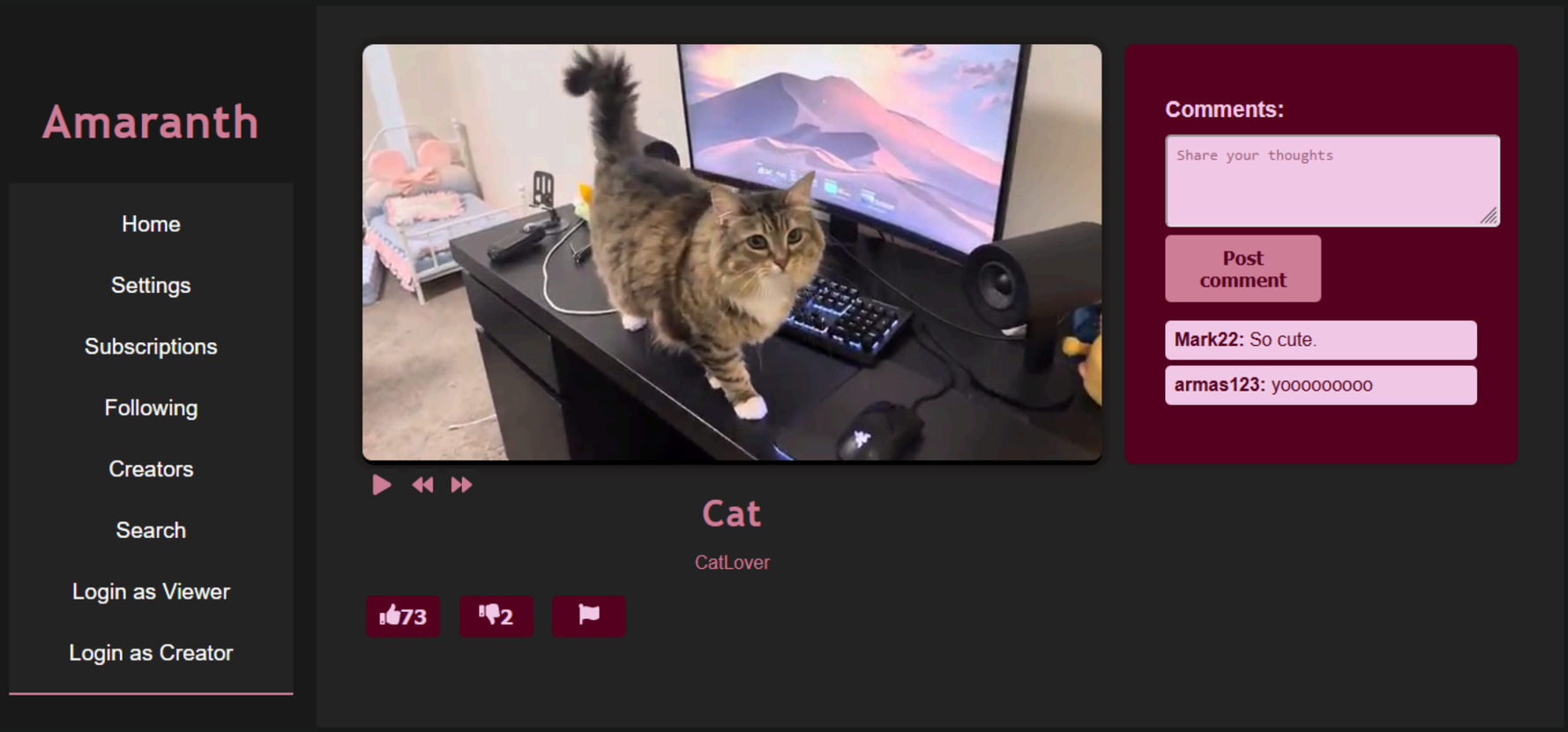
Actual results: Pass
Edge cases: Spamming the button is handled without issues, 0 likes are incremented to one, max integer likes

WHITE BOX TESTS

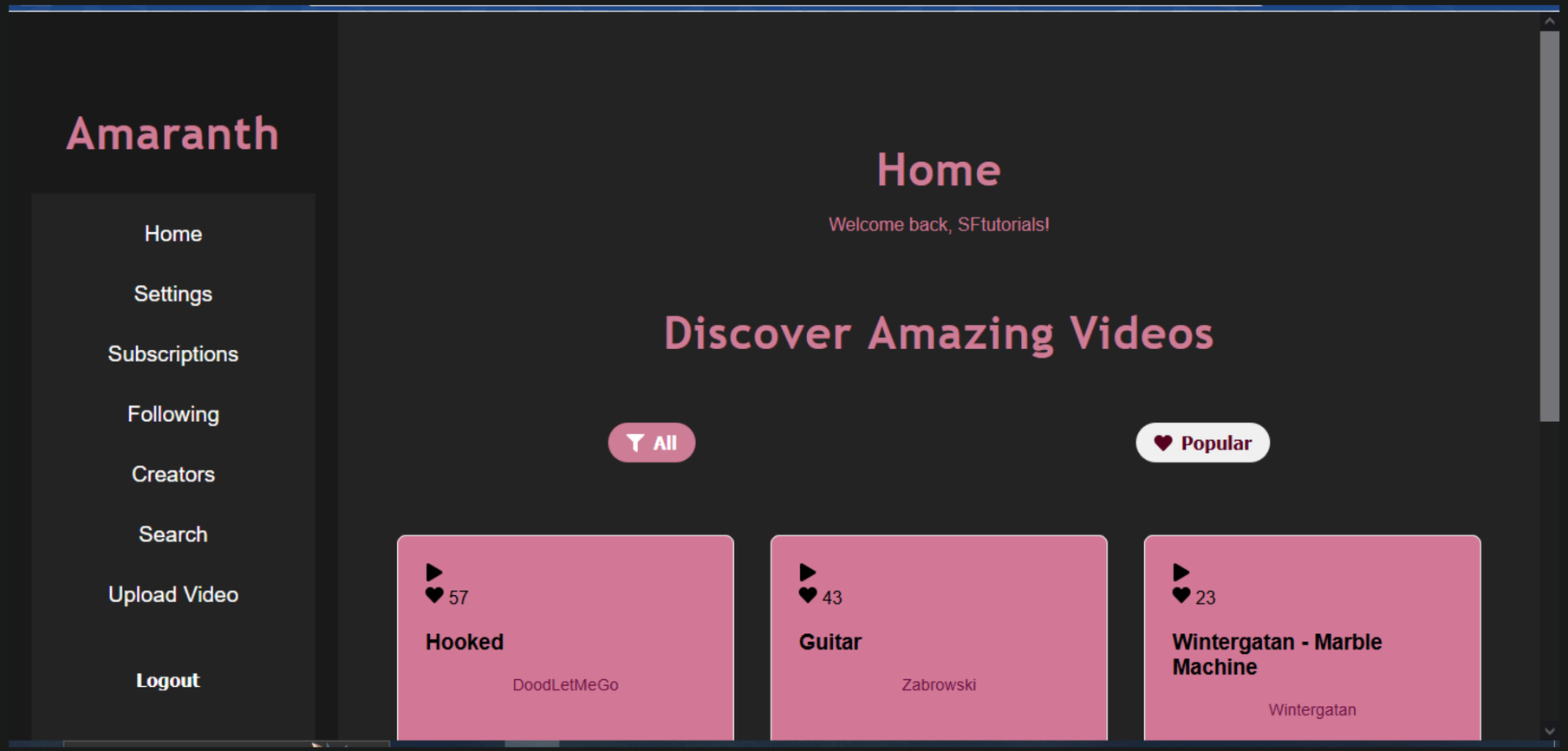
Test Case ID: WB-VIDEO-001
Component: GET /search route
Objective: Ensure the API returns a 400 error if no search query (q) is provided.
Input Data: No query parameter, just /api/videos/search
Output: HTTP 400, { message: "Search query is required" }
Status: Pass

Test Case ID: WB-VIDEO-002
Component: GET/:id route
Objective: Verify that a valid video ID returns the correct video data.
Input Data: /api/videos/653c1ab7f2d2a9a89f10b2c3
Output: HTTP 200, JSON object with video details.
Status: Pass

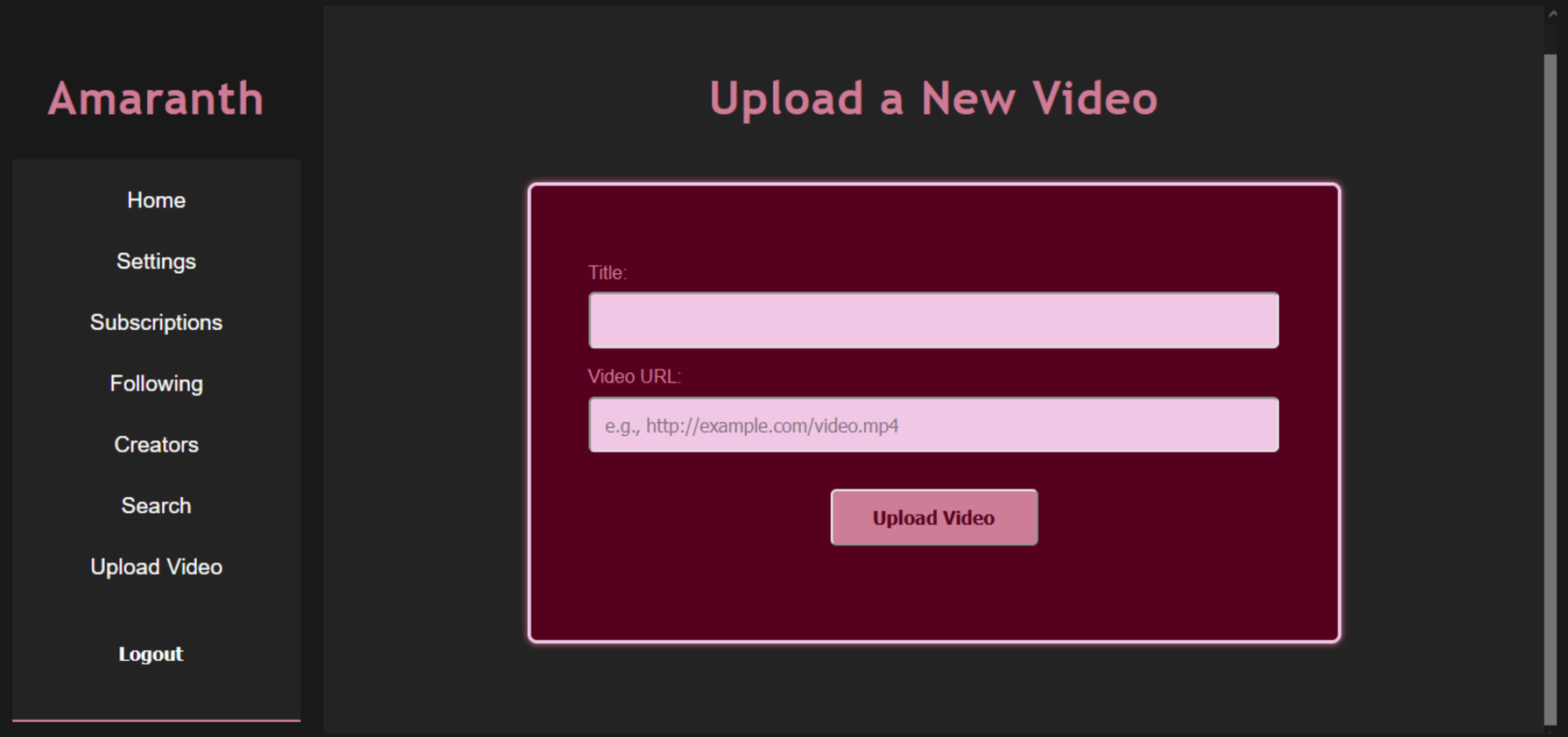
IMPLEMENTATION SCREENSHOTS



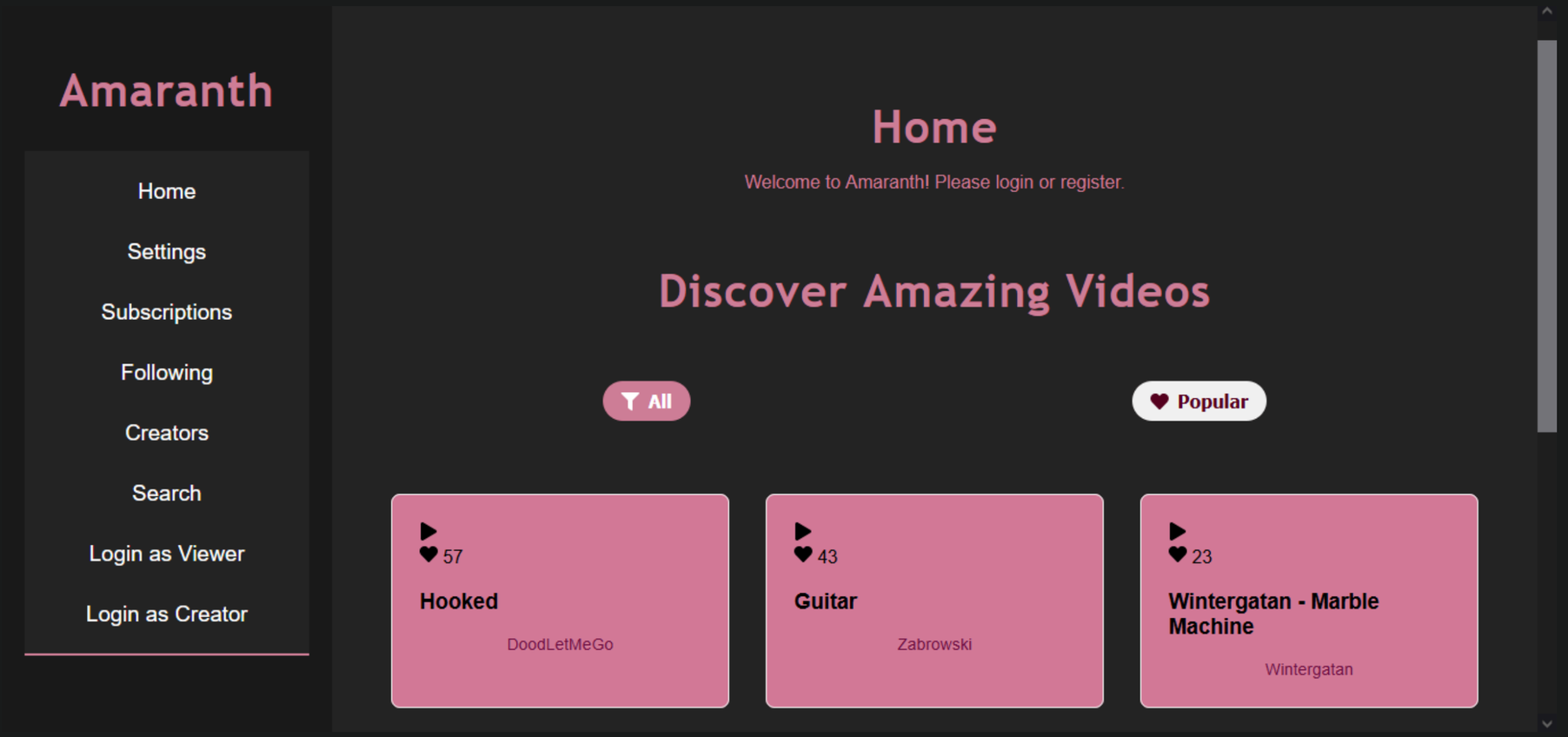
IMPLEMENTATION SCREENSHOTS



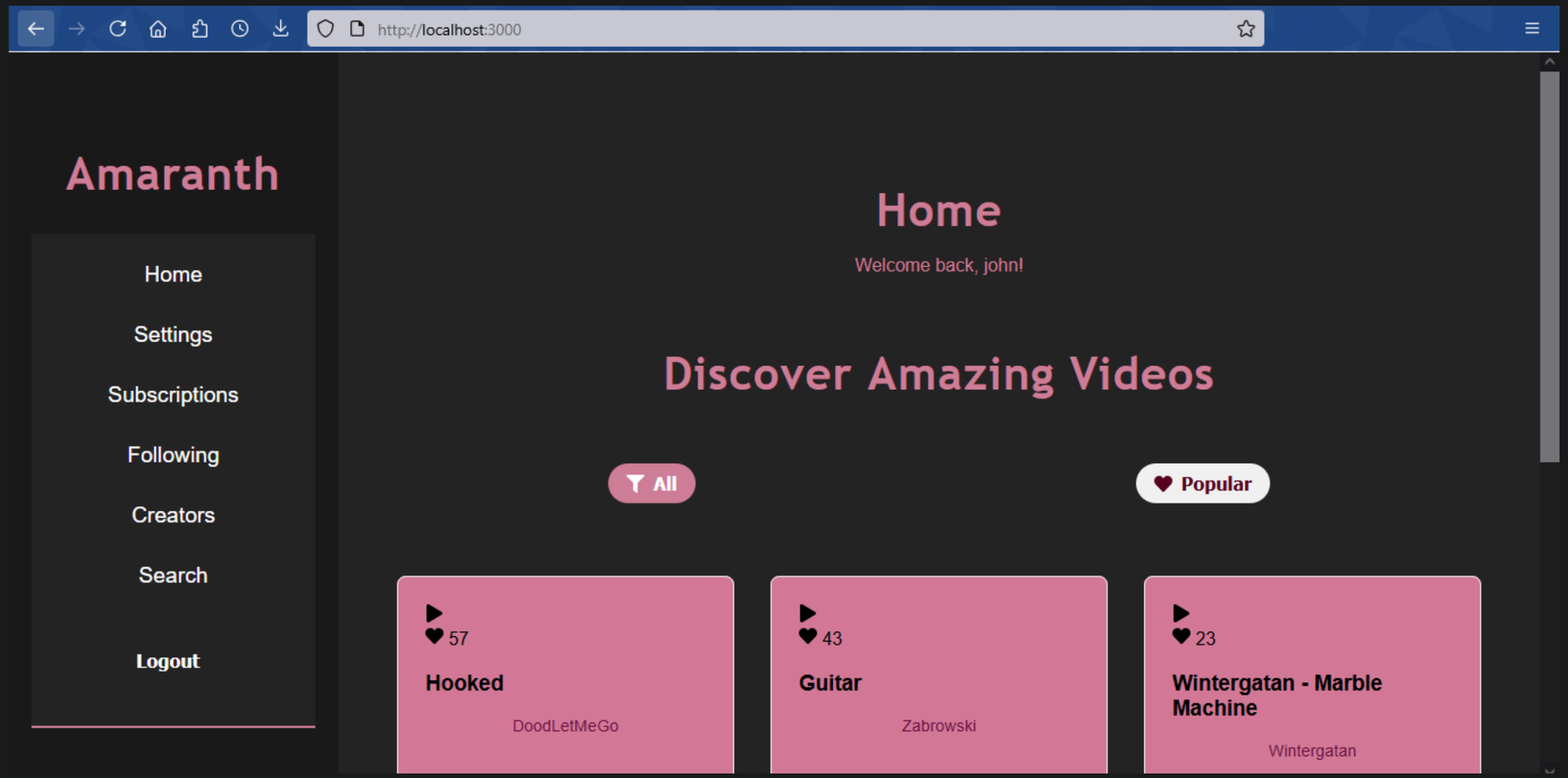
IMPLEMENTATION SCREENSHOTS



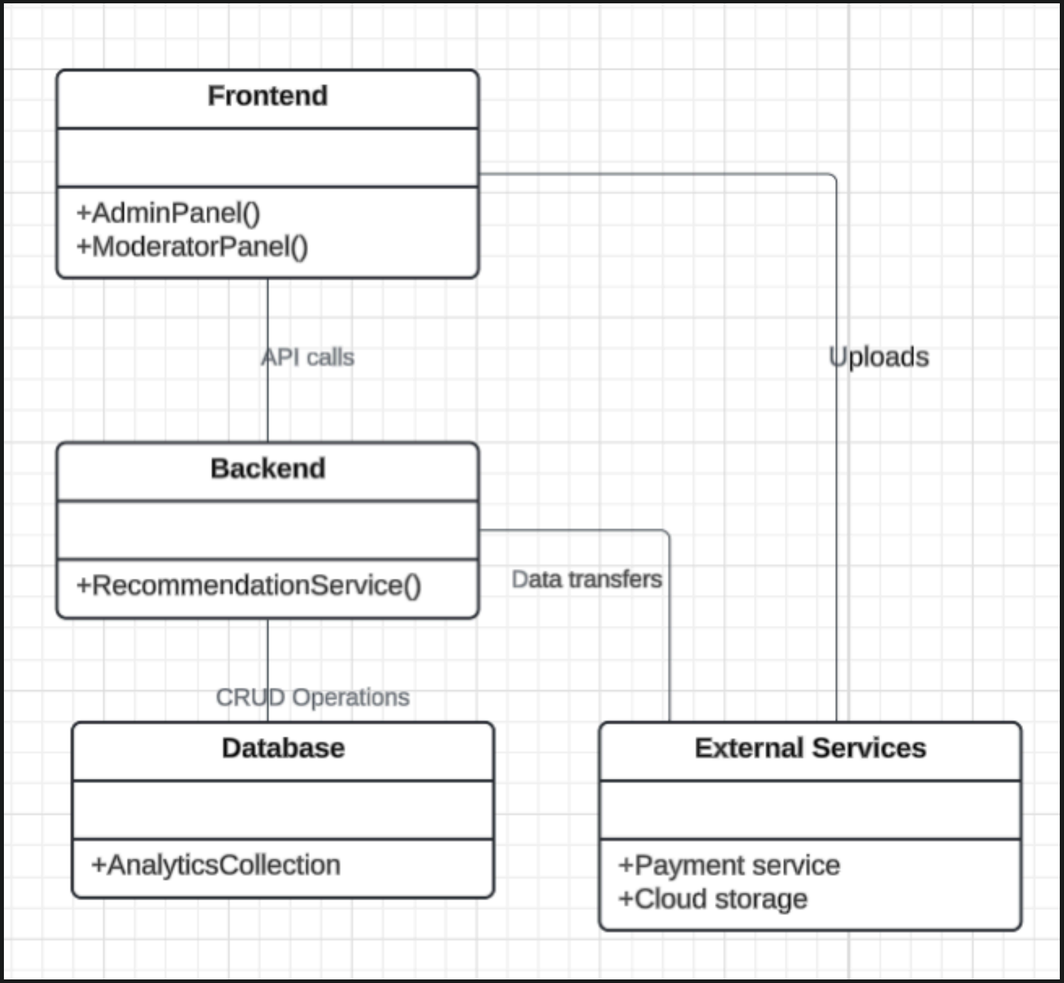
IMPLEMENTATION SCREENSHOTS



IMPLEMENTATION SCREENSHOTS



FUTURE IMPROVEMENTS



WORK DIVISION

Hassan Ismail: Viewing and handling subscriptions and following, and uploading video.

Rehal Saeed: Search video, video playback and controls, video feedback, viewer home page.

Samra Mashaam: Registration pages, login authentication, donation and account settings.

LESSONS LEARNT:

- Understanding feasibility
- Understanding scope
- Understanding our framework
- Tracking progress
- The importance of communication
- Properly structured testing

THANK YOU