

# T1-UP Individual Contributions

By: Team1

# NavBar & AgreeCount/Upvote System

I agree with Side A because if you look at it this way and not imagine it like this

I agree with Side A because if you look at it this way and not imagine it like this then the other side is wrong!

AGREE? 0

AGREE? 1



**Whole Lotta Nothing** • [Questions](#) • [Your Responses](#) • [Other's Responses](#) • [Account](#)

<https://www.cs.umb.edu/~hdeblois/cs410/longproj01/t1-UP/Artifacts/walterArtifacts/upvoteSystem.html>

# Research Essay Topic -Walter

- My research essay topic will be on why software projects fail. Software projects don't usually fail because of hardware and technical limitation but because of poor management and practices. Whether it is from a lack of communication or from unrealistic expectations my paper will explore why many software projects never see the end goal.

# User Posting Platform – Valentina Zheng

**Whole Lotta Nothing** [Home](#) [Your Responses](#) [Other's Responses](#) [Account](#)

**Moderator Panel** [User UI](#) [Moderator Home](#) [Content Mod](#)

[Load Latest Question](#)

**Current Argument Question**  
**Does pineapple belong on pizza?**

**Pick Your Stance**

No side selected yet!

**Your Reason**

0 / 500 characters

**Create/Edit Argument Question**

**Custom Sides (Add/Edit/Delete)**

[Open User UI in New Tab](#)

- My research essay topic will be on automated testing vs. manual testing in software engineering.

# Moderator UI: Publish Custom Questions/Stances

**Moderator Panel** User UI Moderator Home Content Mod

**Create/Edit Argument Question**

**Custom Sides (Add/Edit/Delete)**

Delete

Delete

[+ Add Another Side](#)

[PUBLISH TO USER UI](#)

[Refresh User UI \(Load Latest Question\)](#)

[Open User UI in New Tab](#)

# Research Essay Topic - Valen

- My research essay topic will be on automated testing vs. manual testing in software engineering. Software teams don't just pick one type randomly, each has its own strengths and best uses. Whether it's using tools like Jest for quick automated checks during development or relying on manual testing for usability and edge cases, my paper will explore when to use each to get the best software quality.

# Account Creation/Login System - James



Whole Lotta Nothing • [Questions](#) • [Your Responses](#) • [Other's Responses](#) • [Account](#)

## A Whole Lotta Nothing

### Create Account

Username:

Password:

Create Account

# Research Essay Topic - James

- How a web server receives, processes, and responds to an HTTP request.

# Report System - Samantha

## Rules:

- 1.) Do not ask questions that target specific groups. Any questions that discriminate others by race, gender, sexuality, or religion will not be tolerated.
- 2.) Do not ask questions/provide arguments in relation to politics or serious discussion. Questions and answers must be fun and not cause serious arguments.
- 3.) Do not use explicit language that targets specific groups. Language that discriminates others by race, gender, sexuality, or religion will not be tolerated.
- 4.) Users may debate, but any harrassment or harmful threats against other users is strictly prohibited.
- 5.) Explicit content is prohibited and will result in an account ban.

See a problem?

Report

# Report System - Samantha

## What's the problem?

- Rule 1: This content contains discriminatory language, topic and/or imagery
- Rule 2: This content is too serious
- Rule 3: This language is offensive
- Rule 4: This user will not leave me alone/making threats towards me.
- Rule 5: This content is explicit.

Other:

Submit

# Research Essay Topic - Samantha

- For my research project, I'm interested in analyzing errors/bugs that occur during the production of the project. Even when the program compiles, the output may only be ideal for some inputs, but not all. I want to explore how these errors/bugs occur and how they're resolved. I also want to connect these ideas with a team's structure, so team members can work together, learn, and improve on their skills to prevent similar errors in the future.

# User permissions - Domenic

- Owner – Has all permissions of admin and access to backend code.
- Admin – Has separate account registration and handles moderation etc.
- User – Post answers, upvote other user posts, etc
- Banned Users – Can no longer access account/log in is restricted.

```
// permissions for each account type
const permissions = {
  owner: [
    "host_website",
    "backend_access",
    "write_permissions",
    "content_moderation",
    "select_questions",
    "remove_questions",
    "remove_responses",
    "remove_threads",
    "schedule_rotating_question",
    "ban_users",
    "give_warnings",
    "give_timeouts",
    "submit_questions",
    "edit_own_account",
    "post_answers",
    "vote_on_responses",
    "switch_sides",
    "invite_users"
  ],
  admin: [
    "content_moderation",
    "select_questions",
    "remove_questions",
    "remove_responses",
    "remove_threads",
    "schedule_rotating_question",
    "ban_users",
    "give_warnings",
    "give_timeouts"
  ],
}
```

Whole lotta Nothin

## Create Admin Account

Username:

Password:

Confirm Password:

Admin Key Phrase:

[Create Admin Account](#)

Research topic: I'll be researching ethical and social impacts of AI implementation in the software engineering workplace.