BUZZ

BUZZ CAMPUS POLLS & MEMES PLATFORM
Final Project Submission Document

Project Name: BUZZ - Campus Polls & Memes Platform
Submission Date: 19/10/2025
Project Repository: https://github.com/WAD621S-2025/CAMPUS-POLLS.git

NAME	STUDENT NUMBER	ROLE
CASEY MULUTI	223022861	Project Manager Full Stack Developer
DYRALL BEUKES	223058467	Backend Developer Database Administrator
ADEN BEUKES	221138072	Database Administrator QA Engineer

SUMMARY

BUZZ is a comprehensive web-based platform designed to enhance campus community engagement through interactive features including polls, memes, events, and real-time updates. The platform addresses the lack of a centralized, engaging space for student interaction on campus by providing an interface where students can express opinions, share humor, and stay connected with campus activities.

1. PROJECT OVERVIEW

1.1 Background
Many campuses lack a single, easy-to-use platform where students can share their thoughts, opinions, and get updates quickly. Traditional noticeboards and group chats are either too formal, messy, or not engaging enough. This makes it harder for students to participate, feel part of the community, and have their voices heard.

1.2 Solution
BUZZ provides a centralized, interactive platform that combines social features with practical campus utilities, creating a vibrant digital community space where students can:
- Participate in polls and express opinions
- Share and enjoy campus-related memes
- Stay updated on campus events
- Engage with peers through comments and reactions
- Access real-time campus news and announcements

1.3 Target Audience
- University/College students
- Campus organizations and clubs
- Student affairs departments
- Campus event organizers

2. PROBLEM STATEMENT & OBJECTIVES

2.1 Problem Statement
Students currently face:
- Fragmented communication across multiple platforms

- Lack of engagement with campus activities
- Difficulty staying updated on events and news
- Limited peer interaction and expression
- Inefficient event promotion and discovery

2.2 General Objective

To create a web-based platform where students can take part in polls, share memes, and interact in real time, helping them connect, express themselves, and engage with campus life.

3. TECHNICAL ARCHITECTURE

3.1 Technology Stack

Frontend
- **HTML5** - Semantic markup and structure
- **CSS3 & Tailwind CSS** - Modern, responsive styling
- **JavaScript (ES6+)** - Dynamic interactions and AJAX calls
- **Font Awesome** - Icon library
- **Google Fonts (Inter)** - Typography

Backend
- **PHP 8.0.30** - Server-side logic and API endpoints
- **Apache 2.4.58** - Web server

Database
- **MySQL (MariaDB)** - Relational database
- **Port 3307** - Custom database port configuration

Development Tools
- **XAMPP** - Local development environment
- **Visual Studio Code** - Code editor
- **GitHub** - Version control and collaboration
- **phpMyAdmin** - Database management

3.2 Project Structure

```
└── CAMPUS-POLLS/
    ├── about.html
    ├── chart.js
    ├── edit_profile.php
    ├── events.html
    ├── forgot_password.html
    ├── index.html
    ├── login.html
    ├── memes.html
    ├── package-lock.json
    ├── package.json
    ├── polls.html
    ├── README.md
    ├── register.html
    ├── reset_password.html
    ├── script.js
    ├── server.js
    ├── style.css
    ├── test_db.php.php
    ├── test_session.php
    ├── user_profile.php
    ├── includes/
    │   ├── config.php
    │   ├── database.php
    │   ├── functions.php
    │   └── session_check.php
    ├── images/
    │   ├── meme1.jpg
    │   ├── meme2.jpg
    │   ├── meme3.jpg
    │   ├── meme4.jpg
    │   ├── meme5.jpg
    │   └── meme6.jpg
    ├── api/
    │   ├── add_event.php
    │   ├── events.php
    │   ├── forgot_password.php
    │   ├── get_events.php
    │   ├── get_memes.php
    │   ├── get_polls.php
    │   ├── index.php
    │   ├── login.php
    │   ├── logout.php
    │   ├── memes.php
    │   ├── meme_comment.php
    │   ├── meme_like.php
    │   ├── polls.php
    │   ├── poll_results.php
    │   ├── profile.php
    │   ├── register.php
    │   ├── reset_password.php
    │   ├── test.php
    │   ├── upload_meme.php
    │   └── vote.php
    └── uploads/
        ├── profile_pics/
        │   └── 1_68f391e7bc3c2.jpg
        └── memes/
            ├── meme_68f38b61aef545.84479777.jpg
            ├── meme_68f3b46e1e4c14.11887738.jpg
            └── meme_68f401e306dfe1.74969362.jpg
```

4. FEATURES IMPLEMENTED

4.1 User Authentication & Secur
User Registration & Login
- Secure password hashing using password_hash()
- Session-based authentication
- Persistent login state management

Session Management
- Automatic logout after 30 minutes of inactivity
- Session timeout tracking in database
- Activity monitoring and updates
- Force logout after password reset
- Session cleanup and garbage collection

4.2 Polls System
Features:
- Create custom polls with multiple options
- Real-time voting with instant results
- Visual results display
- Poll categories and filtering

- Anonymous voting option
- Poll expiration dates
- Share poll results

Technical Implementation:
- AJAX-based voting for seamless UX
- Database normalization for poll data
- Vote validation and integrity checks
- Real-time result calculation

4.3 Events Calendar
Features:
- Dynamic event posting with form validation
- Date and time picker integration
- Event categories (Academic, Social, Sports, Workshop, Other)
- Event filtering by category
- Upcoming events feed
- Real-time event loading via API

4.4 Memes Gallery
Features:
- Image upload and display
- Meme categorization
- Like/reaction system
- Comment functionality
- Trending memes section
- User-generated content moderation
- Responsive image gallery

4.5 User Interface Features
Design Elements:
- **Dark Mode Toggle**
- **Honeycomb Background**
- **Card-based Layout** - Modern, clean interface
- **Smooth Animations** - Slide-in alerts, hover effects
- **Loading Indicators** - User feedback during operations
- **Success/Error Messages** - Clear user notifications
- **Icon Integration**

5. DEVELOPMENT PROCESS
5.1 Project Management

Methodology
We adopted an Agile-inspired approach with:
- **Sprint-based development** - Weekly iteration cycles
- **Daily standups** - Team sync meetings
- **Code reviews** - Peer review before merging
- **Continuous integration** - Regular testing and deployment
- **Collaboration Tools Used:** Trello, Github

6. DATABASE DESIGN
6.1 Database Schema
Entity Relationship Diagram (ERD)

6.2 Table Specifications

CREATE TABLE `active_polls_view` (
`poll_id` int(11)

```
,`user_id` int(11)
,`question` varchar(255)
,`description` text
,`poll_type` enum('single','multiple')
,`category` varchar(50)
,`is_anonymous` tinyint(1)
,`total_votes` int(11)
,`is_active` tinyint(1)
,`start_date` datetime
,`end_date` datetime
,`created_at` timestamp
,`username` varchar(50)
,`option_count` bigint(21)
);

CREATE TABLE `activity_log` (
  `log_id` int(11) NOT NULL,
  `user_id` int(11) NOT NULL,
  `activity_type` varchar(50) NOT NULL,
  `reference_id` int(11) DEFAULT NULL,
  `description` text DEFAULT NULL,
  `created_at` timestamp NOT NULL DEFAULT current_timestamp()
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;


CREATE TABLE `comments` (
  `comment_id` int(11) NOT NULL,
  `user_id` int(11) NOT NULL,
  `comment_text` text NOT NULL,
  `created_at` timestamp NOT NULL DEFAULT current_timestamp()
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;

CREATE TABLE `events` (
  `event_id` int(11) NOT NULL,
  `title` varchar(255) NOT NULL,
  `category` varchar(50) NOT NULL,
  `description` text NOT NULL,
  `event_date` datetime NOT NULL,
  `is_public` tinyint(1) NOT NULL DEFAULT 1,
  `created_at` timestamp NOT NULL DEFAULT current_timestamp()
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;

CREATE TABLE `memes` (
  `meme_id` int(11) NOT NULL,
  `user_id` int(11) NOT NULL,
  `image_url` varchar(255) NOT NULL,
  `caption` text DEFAULT NULL,
  `likes_count` int(11) DEFAULT 0,
  `comments_count` int(11) DEFAULT 0,
  `is_active` tinyint(1) DEFAULT 1,
  `created_at` timestamp NOT NULL DEFAULT current_timestamp()
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;

CREATE TABLE `meme_comments` (
  `comment_id` int(11) NOT NULL,
  `meme_id` int(11) NOT NULL,
  `user_id` int(11) NOT NULL,
  `comment_text` text NOT NULL,
```

```sql
  `created_at` timestamp NOT NULL DEFAULT current_timestamp()
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;

CREATE TABLE `meme_likes` (
  `like_id` int(11) NOT NULL,
  `meme_id` int(11) NOT NULL,
  `user_id` int(11) NOT NULL,
  `liked_at` timestamp NOT NULL DEFAULT current_timestamp()
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;

CREATE TABLE `notifications` (
  `notification_id` int(11) NOT NULL,
  `user_id` int(11) NOT NULL,
  `content` varchar(255) NOT NULL,
  `is_read` tinyint(1) DEFAULT 0,
  `link` varchar(255) DEFAULT NULL,
  `created_at` timestamp NOT NULL DEFAULT current_timestamp()
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;


CREATE TABLE `polls` (
  `poll_id` int(11) NOT NULL,
  `user_id` int(11) NOT NULL,
  `question` varchar(255) NOT NULL,
  `description` text DEFAULT NULL,
  `poll_type` enum('single','multiple') DEFAULT 'single',
  `category` varchar(50) DEFAULT NULL,
  `is_anonymous` tinyint(1) DEFAULT 0,
  `total_votes` int(11) DEFAULT 0,
  `is_active` tinyint(1) DEFAULT 1,
  `start_date` datetime DEFAULT NULL,
  `end_date` datetime DEFAULT NULL,
  `created_at` timestamp NOT NULL DEFAULT current_timestamp()
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;

CREATE TABLE `poll_options` (
  `option_id` int(11) NOT NULL,
  `poll_id` int(11) NOT NULL,
  `option_text` varchar(255) NOT NULL,
  `vote_count` int(11) DEFAULT 0,
  `option_order` int(11) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;

CREATE TABLE `poll_votes` (
  `vote_id` int(11) NOT NULL,
  `poll_id` int(11) NOT NULL,
  `option_id` int(11) NOT NULL,
  `user_id` int(11) NOT NULL,
  `voted_at` timestamp NOT NULL DEFAULT current_timestamp()
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;

CREATE TABLE `sessions` (
  `session_id` varchar(128) NOT NULL,
  `user_id` int(11) NOT NULL,
  `user_agent` varchar(255) DEFAULT NULL,
  `ip_address` varchar(45) DEFAULT NULL,
  `last_activity` timestamp NOT NULL DEFAULT current_timestamp(),
  `created_at` timestamp NOT NULL DEFAULT current_timestamp()
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;
```

```sql
CREATE TABLE users (
    user_id INT AUTO_INCREMENT PRIMARY KEY,
    username VARCHAR(50) UNIQUE NOT NULL,
    email VARCHAR(100) UNIQUE NOT NULL,
    password_hash VARCHAR(255) NOT NULL,
    full_name VARCHAR(100),
    profile_picture VARCHAR(255),
    bio TEXT,
    created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    updated_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
    INDEX idx_email (email),
    INDEX idx_username (username)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;


CREATE TABLE events (
    event_id INT AUTO_INCREMENT PRIMARY KEY,
    title VARCHAR(255) NOT NULL,
    category VARCHAR(50) NOT NULL,
    description TEXT NOT NULL,
    event_date DATETIME NOT NULL,
    is_public TINYINT(1) DEFAULT 1,
    created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    INDEX idx_event_date (event_date),
    INDEX idx_category (category)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;


CREATE TABLE password_resets (
    id INT AUTO_INCREMENT PRIMARY KEY,
    user_id INT NOT NULL,
    token_hash VARCHAR(64) NOT NULL,
    expires_at DATETIME NOT NULL,
    created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    UNIQUE KEY unique_user (user_id),
    INDEX idx_token_hash (token_hash),
    INDEX idx_expires_at (expires_at),
    FOREIGN KEY (user_id) REFERENCES users(user_id)
        ON DELETE CASCADE
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```

7. USER INTERFACE & EXPERIENCE

7.1 Visual Identity
Color Palette:

Primary: #F59E0B (Amber) - Brand color
Secondary: #FBBF24 (Yellow) - Accent
Dark: #1F2937 (Gray) - Text/backgrounds
Light: #FEF3C7 (Amber) - Backgrounds
Success: #10B981 (Green) - Confirmations
Error: #EF4444 (Red-500) - Alerts

7.3 Key UI Components

Navigation Bar
- Sticky header for easy access
- Active page highlighting
- Responsive hamburger menu (mobile)
- User profile dropdown
- Dark mode toggle

Cards
- Consistent padding and spacing
- Hover effects for interactivity
- Shadow depth for visual hierarchy
- Border accents by category

Forms
- Clear labels and placeholders
- Real-time validation feedback
- Error message display
- Success confirmations
- Loading states

Modals & Alerts
- Slide-in animations
- Auto-dismiss after 5 seconds
- Manual close option
- Color-coded by type (success/error/warning)

12. CONCLUSION

12.1 Project Summary

BUZZ Campus Platform successfully addresses the identified problem of fragmented campus communication by providing a centralized, engaging, and interactive platform for students. Through the implementation of polls, events, memes, and real-time updates, we have created a digital space that fosters community engagement and enhances campus life.

12.3 Key Learnings

Technical Skills:
- Full-stack web development
- RESTful API design
- Database design and optimization
- Security best practices
- Version control and collaboration

Project Management:
- Agile methodology application
- Task prioritization and tracking
- Team coordination using Trello
- Git workflow and branching strategies
- Documentation

Problem-Solving:
- Debugging complex issues
- Finding creative solutions
- Adapting to constraints
- Learning from failures
- Iterative improvement

SYSTEM SCREENSHOTS

LOGIN PAGE

FORGOT PASSWORD PAGE

REGISTER PAGE

🏛️ 🐝 **BUZZ**

Join the Campus Community

Create Your Account

👤 Full Name

> John Doe

@ Username

> johndoe

✉️ Email

> your.email@university.edu

🔒 Password

> At least 8 characters

Must be at least 8 characters

🔒 Confirm Password

> Re-enter password

👤+ **Create Account**

Already have an account? **Login**

← Back to Home

POLLS PAGE

🏛️ 🐝 BUZZ   Home   Polls   Events   Memes   About Us   🌙 Hi, CASEYDABESH!   [→ Logout   ©

🗒️ **Campus Buzz Polls**

➕ **Create a New Poll**

> Poll question (e.g., Which department is the best?)

> Option 1

> Option 2

+ Add Another Option

☐ Anonymous Poll

Category: Academic ▾

➤ **Submit Poll**

≡ **Active Polls**   Newest First ▾

Fav food spot on campus?   academic

Posted by: User • **0 votes**

○ kasi burger

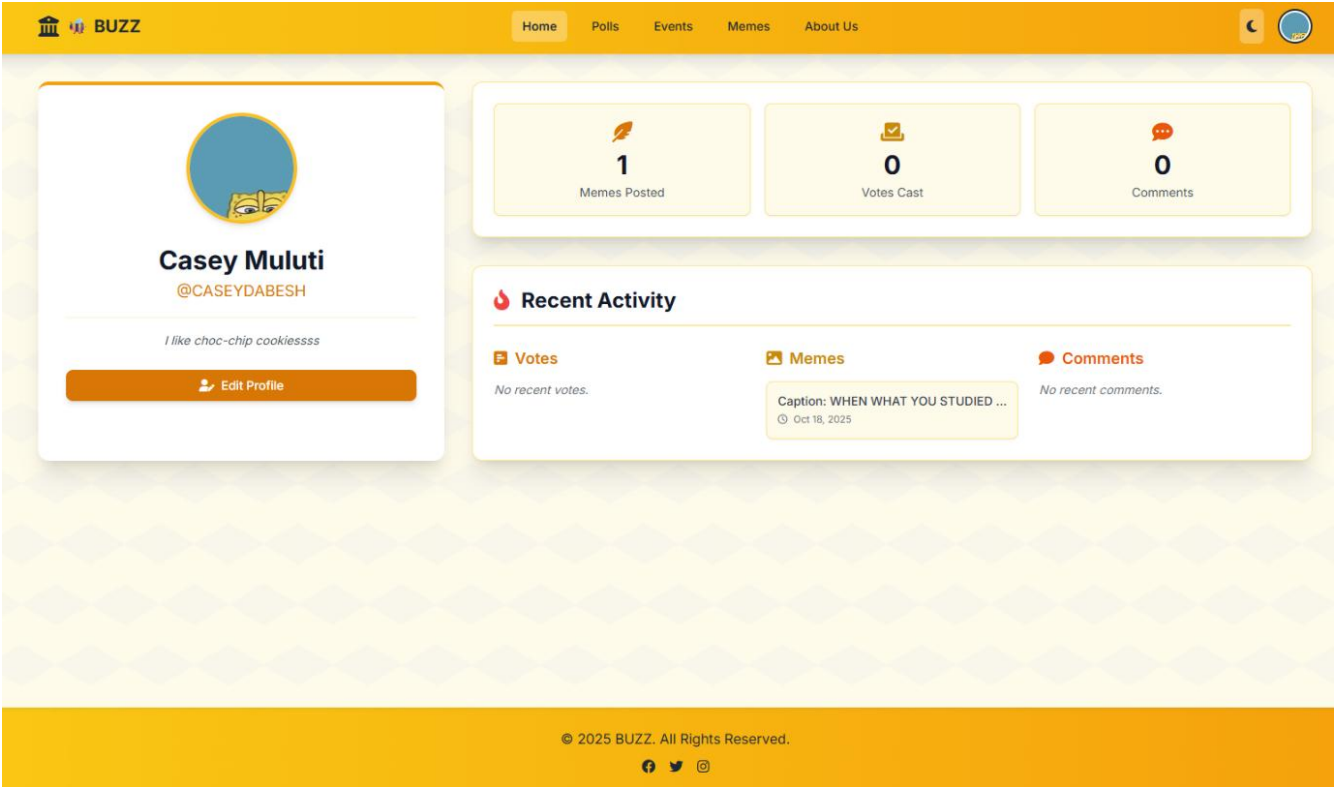○ hotspot burger

Hi there!
🐝

EVENTS PAGE

MEMES PAGE

ABOUT US PAGE
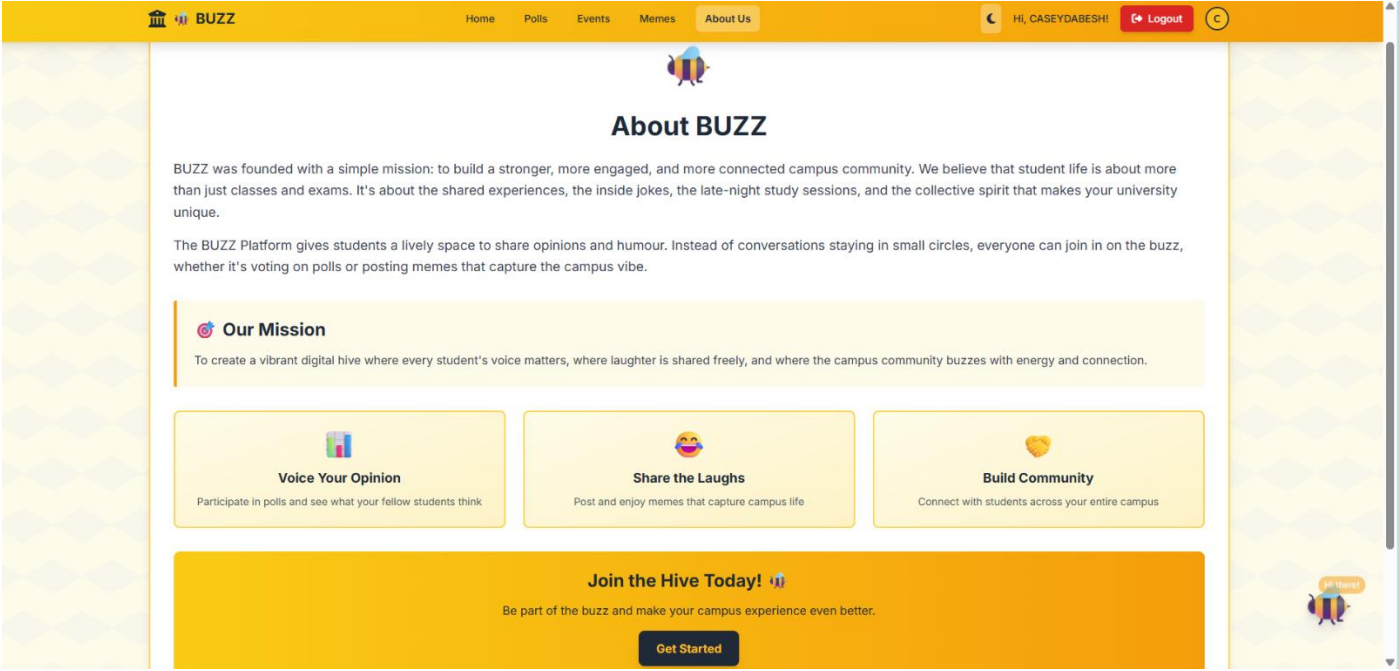
PROFILE PAGE

TRELLO BOARD