

FLASHCODE

CAPSTONE PROJECT

TEAM DELTA (BLACK SQUIRRELS)

TEAM MEMBERS

Raghava Jaya Sai Rakesh Kota - 811300200

Nithin Kumar Jammichedu Kuruva – 811314769

Shiva Krishna Reddy Nannuru - 811283939

Nithin Kalavagunta - 811357782

Srinivas Kadalal - 811313880

Tarun Gopi Reddy Vajrala - 811323522



2. Executive Summary:

FlashCode is an easy – to – use application, which is designed mainly for the students to make a collaborative question – and – answer platform especially who are learning and developing their skills. This is more welcoming for beginners, mainly those are in the academic settings. The main aim is to support students but giving them a helpful place to post their doubts, and to get the peer responses, also grow through the community learning.

2.1 Goals and Objectives:

- This will allow the students to ask any kind of questions which are related to their majors and the other users can send the answers.
- This will also enable users to tag questions based on their topics like ReactJS, Python, etc.
- This application is making the coding process more accessible through a clean and simple interface.
- Encouraging interaction through the voting, bookmarking and user profiles.
- Providing the application with light and dark theme.
- Developing an admin dashboard to monitor the performance of users, and admin can perform either delete or modify the details of user like blocking them.
- Also, in admin dashboard we are aiming to provide a feature to the admins like adding the details of upcoming event, published magazines and schedule of university.

- Providing the code editor for the user to practice and develop their skills.
- Providing tags route to filter the question based on the particular domain or section.

2.2 Major Features Delivered:

- Designed a fully responsive and accessible web design.
- Used OAuth – based login for the system with the session management.
- Developed some Community page (KentX) to display the content which discussions are trending.
- Built user profile pages which are showing the questions, answers and their achievements.
- Provided features to submit the question with tag which indicating the topic and addition to that provided answer voting.
- Easy transition between the light and dark mode.
- Daily challenges for coding practice to develop the students coding skills with random question along with the filtering option of the levels.
- Developed admin dashboard for the analysis of performance and provided the feature to modify the contents and giving access to add information of other things like events, magazines and schedule or university calendar.
- Used clerk authentication in admin dashboard for the secure validation.

2.3 Achievements:

- We have successfully developed a working application which can be used by the authorized students.
- Used ShadCN and Tailwind for the better and clean modern UI.
- This is also ensuring the form input is valid and secure using Zod.

- We also organized the system with the well – structured backend and MongoDB database.
- Developed secure login and user role handling.

2.4 Challenges:

- We faced challenge mainly at the time of keeping the design consistent across the different pages while using the different component libraries.
- Also, managing the user state across the different parts of the site.
- This web application had multiple components which makes integration difficult with maintaining the consistency.
- Loading of profile card in the main home page under the question card.
- Faced difficulties while fetching the data from main application to the admin dashboard.
- We faced issues with the state management, which is meant getting the data and updating the data across various components when they were redirect to other pages.
- While integration “MDX Editor” it became difficult to match with THE layout and loading it properly.
- Another issue is showing the icons which should match with the tags.
- The other main issue we faced while getting the data accurately for the activity tracking in the admin dashboard.

3. Problem Statement:

Mostly nowadays the students are relying on the AI tools for studying purposes, because those will generate answers in short time with more accuracy which is leading to reduction of thinking capabilities for the students. There are some existing platforms which are often too advance or unfriendly for beginners. This application is built to address

this issue by giving an opportunity to students through a moderated space where they can learn and they can help each other. Through this we can even develop the new connections in the educational field with the other domain people which will allow the students to explore more topics in other domain.

3.1 Stakeholders:

- Our stakeholder is Brent Hanni currently he is working in culinary services.
- Universities and college students could be the stakeholders.
- Teaching assistants and the instructors.
- Some educational institutions which promote the peer learning.

3.2 Who Needs This:

- Mainly for the students who are interested in learning.
- Teacher assistants mainly who interested in helping the learners.
- The schools or colleges which are mainly aiming to promote the peer – to – peer learning.

3.3 Why It Matters:

- This will encourage the teamwork and learning with the help of discussion.
- Which will help the students to save the time when they were encountered with the small issues.
- This will help in building a stronger sense of coding community mainly in the academic environment.

4. System Overview / Solution Description:

FlashCode is a full stack web – application which is build using a modern technology which includes both user – friendly interface and the strong backend system.

4.1 High – Level System Architecture:

- **Frontend (Client)**

- **Framework:** Next.js (React-based)
- **Languages:** TypeScript, Tailwind CSS
- **Key Libraries:**
- **Clerk:** User authentication (login, sign-up, role handling)
- **TanStack Query:** Data fetching and caching
- **Recharts:** Analytics and chart visualizations

- **Pages & Features:**

- Landing page, question feed, profile, blog, chat, and admin dashboard
- Dark/light mode toggle
- Responsive UI for mobile and desktop

- **Backend APIs**

- **Framework:** Flask (Python)
- **API Design:** RESTful endpoints under /api

- **Responsibilities:**

- Handle all business logic for questions, answers, votes, user actions

- Role-based access control (especially for admin routes)
- Integration with MongoDB for storing and retrieving data

- **Database**

- **Type:** MongoDB (NoSQL)
- **ORMs/Drivers:** Mongoose (for frontend admin dashboard), PyMongo (for backend)

- **Collections:**

- Users, Questions, Answers, Tags, Votes, Blogs, Chat Requests, Messages

- **Design:**

- Optimized with indexes for search
- Relationships between collections handled through reference IDs

- **Authentication & Authorization**

- **Library:** Clerk

- **Features:**

- Secure authentication (OAuth/email/password)
- User roles: student, admin

- Role-based route protection (especially for Admin Dashboard)

- **Deployment**

- **Frontend:** Deployed on Vercel / Cassini (Kent Server)
- **Backend:** Hosted via Render or Cassini Flask app
- **Environment Config:** .env files for keys and URIs

- **Additional Services**

- **Blog Integration:** DEV.to API for tech articles
- **Chat Feature:** Internal chat system with request-approve messaging
- **Dashboard Analytics:** Real-time stats and visual insights for admins

4.2 Technologies Used:

- **Frontend:** Next.js (React), TypeScript, Tailwind CSS, ShadCN UI components.
- **Backend:** Flask (Python), NextAPI routes used for building REST APIs.
- **Database:** MongoDB is used for storing user's data, questions and answers.

4.3 Core Modules:

- Providing functionality of asking a question by the student and answering that question is the main feature of our application.
- Providing Login & Signup with the secure authentication.
- Asking or answering the functionality with the tag filtering.

- KentX page is responsible in providing the trending magazines, upcoming events and schedule of university.
- Provided the information of main events that are placing in the university, hackathons and daily coding challenges.
- User Profile Page which discloses the personal information with their recent activities and provided basic profile edit options.
- Form handling and validation with the Zod, and React Hook Form
- Provided special feature for admins to make the basic functionalities like adding event, reviewing user, fetching the details of number of questions posted in the admin dashboard.

4.4 Design Decisions:

- Utilized MongoDB for the storage of the diverse user and question data.
- Using ShadCN UI which is reusable with the components for fast development.
- Implemented the responsive layout for the accessibility over the different devices with vary screen sizes.
- Used clerk for the authentication which will provide the easy integration, secure and oauth support.
- Used Flask for backend which is lightweight and easy for the integration of python.

5. Development Process:

5.1 Timeline Overview:

- Week 1: Focused on requirement gathering, roles of team members and tech setup
- Week 2: Developed and worked on UI/UX design, wireframes and made plans on the database schema.
- Week 3: Implemented Flask backend setup which is initial for the API development.

- Week 4: Worked on the setup of frontend using Next.js and Tailwind, along with the routing and layout.
- Week 5: Worked on the Auth system with the Clerk, and protected routes and user roles.
- Week 6: Deployed full Q & A functionality and the chat request.
- Week 7: Final deployment of Admin dashboard, blog integration and made the final testing of all features.
- Week 8: Deployment of total project and giving all the polish to the components and fixed all bugs and then produced final product.

5.2 Team Workflow:

- Our team followed Agile workflow with the stand ups and clearly defined roles. Each team member of our group mainly focused on their assigned work such frontend, backend, authentication or UI design. For the process of tracking the work we have used Trello and at the same time we worked as sprints to build and test features. We are usually intent to give feedbacks and helped to the other team members in solving the issues they have faced and made changes to UI based on some inputs and feedback. We have used GitHub for the control of version and made pull request were peer – reviewed before doing the merging step. Instructors are also given some feedback helped us to improve security and UI. This collaborative and iterative process helped us to stay organized and efficient and mainly focused on delivering the polished project within the deadline.

5.3 Team Roles:

- **Nithin:** Frontend Lead (Next.js, TypeScript, Tailwind CSS)
- **Rakesh:** UI/UX Developer (ShadCN, Tailwind Design System)
- **Srinivas:** Backend Lead (Flask, MongoDB, API Integration)
- **Shiva:** Authentication & Security (NextAuth.js, OAuth, JWT)
- **Tarun:** Form Handling & Validation (React Hook Form, Zod)

- **Nithin K:** Database & Data Modelling (MongoDB, Schema Design)

6. Implementation Details:

6.1 Important Files and Logic:

- **Voting Logic:** Through this it is possible to prevent the duplication of votes from the same user. By using the MongoDB queries make the real – time updates of vote counts and answer rankings.
- **Role – Based Access:** For the admin dashboard the admin user can only get access with the help of clerk authentication. With the help of backend, it verifies the user role before giving access to the sensitive roles.
- **Blog Filtering:** The blogs which are fetched from the DEV.to API were got filtered using the tags with the help of local search.

6.2 Project Structure:

📁 app	Update page.tsx	last month
📁 components	Update CodingChallenges.tsx	last month
📁 constants	Create challenges.ts	last month
📁 context	deployment	last month
📁 database	migrate flask to next server for faster fetching	last month
📁 hooks	deployment	last month
📁 lib	migrate flask to next server for faster fetching	last month
📁 public	add community love block	last month
📁 types	deployment	last month
📄 .gitignore	initial commit	last month
📄 README.md	initial commit	last month
📄 auth.ts	fix deployment issue	last month
📄 components.json	deployment	last month
📄 eslint.config.mjs	deployment	last month
📄 middlewate.ts	fis auth_issues	last month
📄 next.config.ts	add blogs and insights	last month
📄 package-lock.json	fix next version	last month
📄 package.json	Update package.json	last month
📄 postcss.config.mjs	initial commit	last month
📄 tailwind.config.ts	deployment	last month
📄 tsconfig.json	initial commit	last month

📁 app	enhancen auth layout	last month
📁 components	fix route name	last month
📁 hooks	first commit	2 months ago
📁 lib	add routes for all featres	last month
📁 public	enhancen auth layout	last month
📁 server	add routes for all featres	last month
📄 .gitignore	add routes for all featres	last month
📄 README.md	update readme	last month
📄 components.json	first commit	2 months ago
📄 eslint.config.mjs	Initial commit from Create Next App	2 months ago
📄 middleware.ts	add routes for all featres	last month
📄 next.config.ts	add routes for all featres	last month
📄 package-lock.json	enhancen auth layout	last month
📄 package.json	add routes for all featres	last month
📄 postcss.config.mjs	Initial commit from Create Next App	2 months ago
📄 tailwind.config.ts	add files to deploy in vercel	2 months ago
📄 tsconfig.json	Initial commit from Create Next App	2 months ago
📄 README		

7. Testing and Validation:

7.1 Testing Types Used:

- Done unit testing for components with the usage Jest.
- Made API testing with the help of Pytest to deal with backend.
- Made Manual validation by using the Postman and browser testing.

7.2 Test Scenarios:

- Try to login with valid/invalid credentials.
- Submission of forms with or without required data.

- Trying to get the proper navigation between the profile, KentX and home.
- Posting of questions.
- Filtering of questions and the voting functionality for the answers.
- Proper generation of daily challenges and submissions.
- Fetching the data from the main application to the admin dashboard and disclosing them with the accurate values.

7.3 Result:

- Almost all test cases were get passed.
- Verified with the proper redirection, error handling and response structure.
- Receiving the good user feedback for the simplicity and clarity.

8. Results and Evaluation:

8.1 Success Factors:

- Successfully implemented all the core features.
- Application is got deployed and working across all kinds of display devices.
- Users can ask questions and receive answers easily.
- Users can do the upvote and downvote of a question successfully.
- Successfully implemented the API to generate random questions for the daily challenges categorized with easy, medium and hard.
- Successful differentiation of filtering of questions using tags.
- Successfully disclosed the information of upcoming events, providing magazines and schedule of university.

8.2 Metrics:

- Fast loading of pages which takes about less than 1.5 seconds.

- Session management worked reliably.
- The user interface compatibility which is tested on the desktop and the mobile.

8.3 Team Learnings:

- Through this project we have learned to work across with both frontend and back at same time by integrating some technologies like Next.js, Flask, and MongoDB
- We also got experience in implementing the secure login system, user roles also protecting the admin routes.
- We experienced improvements in planning of project workflow along with the division of task for each team member and got adapted to the weekly sprints and stand – ups.
- Got an command on development and building of REST APIs with the real – world data and the services.
- Understood the importance of minor UI changes and the effect on the entire user experience and usability of the application.
- Because of this project we had developed our skill in using the GitHub and explored some new things in GitHub such as pull request, code reviews, and the branch management.
- Gained in confidence to deal with the problem by identifying bugs and fixing them with the team participation by the help of console logs, network monitoring and the testing tools.
- Understand how to deploy the both frontend and the backend apps with the help of usage of some platforms such as Cassini server(used for main application), Vercel server(used for admin dashboard).

9. Future Work and Recommendations:

9.1 Future Enhancements:

- Proving the chat feature.
- AI Integration to solve the basic questions to avoid the waiting.
- Provide and option to make connections.
- Providing a feature for user to comment on an answer.

- Instead of working for one university we want to make it accessible for other universities by replicating their university theme.

9.2 Current Gaps:

- Not provided any real-time collaboration/chat yet.
- Providing only limited analytics for user engagement.
- There are some moderation tools still basic.

10. User Manual

To Run Locally:

Frontend:

```
cd frontend  
npm install  
npm run dev
```

Backend:

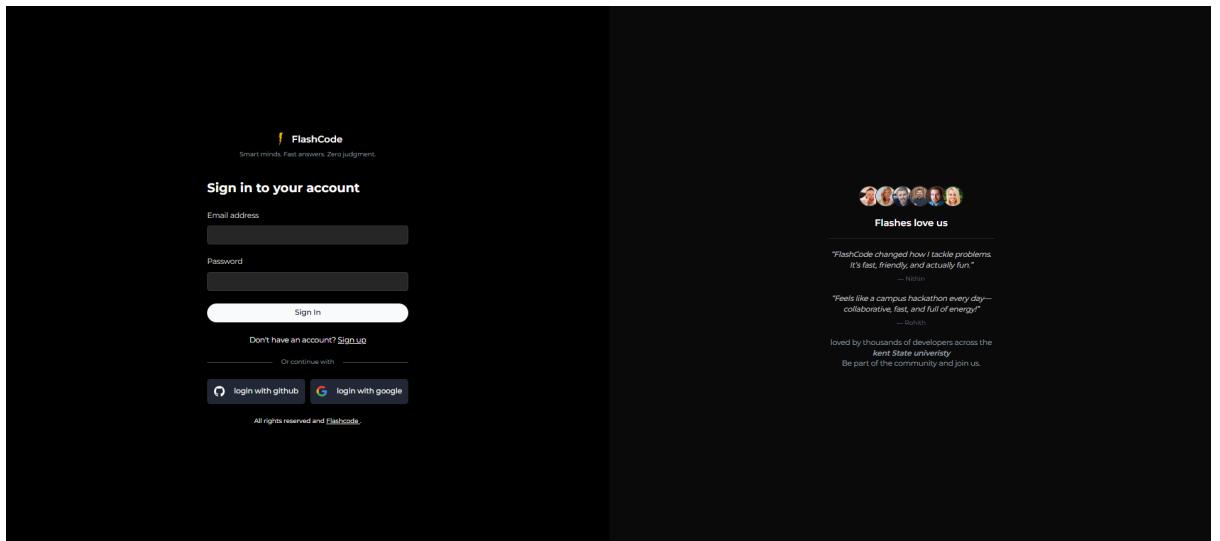
```
cd backend  
pip install -r requirements.txt  
python app.py
```

10.1 Deployment URL: <http://cassini.cs.kent.edu:8010/>

10.1.1 Deployment URL(Admin Dashboard): <https://admin-flashcode.vercel.app/>

10.2 Screenshots Include:

- Login/Signup Page:



• Homepage:

All Questions

- what is antimatter

srinivas - asked 1 week ago
- what is cassini

3K. Nithin kumar - asked 2 weeks ago
- Grounded Theory

Brent Hanni - asked 3 weeks ago

Buzzing Questions

- What is the difference between a framework and a library, and how... >
- What is duck typing in Python? >
- Grounded Theory >
- Why we should choose nextjs instead of reactjs? >
- How does React handle state management? What are some... >

Events & Hackathons

Stay updated with the latest coding events and hackathons at Kent State University.

Coding Challenge of the Day

Flash up your logic, Golden Flash!

Today's challenge is just one click away !

Try Now +

The screenshot shows the FlashCode homepage in light mode. The left sidebar contains links for Home, Community, Collections, KentX, Tags, Profile, Ask a question, and Logout. The main content area displays three questions: "react jgdhfkdtjkhilyjkgflhvivdjk.hf.cnoaXEFa" (React), "How do I get the index of an item in a list in Python?" (Python, List, Indexing), and "hello" (Hello). A navigation bar at the bottom includes About, Events, Student Opportunities, Insights & Stories, X (Twitter), Contact (LinkedIn), and a copyright notice. On the right, there's a sidebar for Events & Hackathons, a Coding Challenge of the Day, and a Popular Tags section.

With Dark Theme

The screenshot shows the FlashCode homepage in dark mode. The layout is identical to the light mode version, featuring the same sidebar, question cards, and sidebar modules. The dark theme changes the background color and the appearance of the UI elements to a darker palette.

• Community Page:

The screenshot shows the FlashCode community page. On the left, a sidebar menu includes Home, Community (selected), Collections, KentX, Tags, Profile, Ask a question, and Logout. The main content area displays a grid of user profiles. Each profile card contains a small profile picture, the user's name, and their GitHub handle. To the right of the grid, there is a sidebar with sections for "Events & Hackathons" (with a "More +" button) and "Coding Challenge of the Day" (with a "Try Now +" button). Below these are "Popular Tags" with icons and counts: REACT (13), KENT (7), PYTHON (6), JAVA (4), and JAVASCRIPT (4).

• Collection:

The screenshot shows the FlashCode Saved Questions collection page. The sidebar menu is identical to the Community page. The main content area displays three saved questions in a grid format. Each card shows the question title, a small thumbnail, the asker's name and profile, and some basic statistics (Votes, Answers, Views). To the right of the cards, there is a sidebar with sections for "Events & Hackathons" (with a "More +" button) and "Coding Challenge of the Day" (with a "Try Now +" button). Below these are "Popular Tags" with icons and counts: REACT (13), KENT (7), PYTHON (6), JAVA (4), and JAVASCRIPT (4).

• Ask a Question:

• KentX:

Event Name	Event Date
New Year's Day (See Note 1)	Tue, Dec 31, 2024
Spring classes begin for first-year Podiatric Medicine students	Sun, Jan 5, 2025
Spring classes begin for second-year Podiatric Medicine students	Sun, Jan 5, 2025
Spring Classes Begin	Sun, Jan 12, 2025
Martin Luther King Jr. Day (See Note 2)	Sun, Jan 19, 2025
Spring classes end for fourth-year Podiatric Medicine students	Thu, Jan 30, 2025
Summer classes begin for fourth-year Podiatric Medicine students	Sun, Feb 2, 2025

• Tags:

The screenshot shows the 'Tags' section of the FlashCode website. It features a search bar at the top right and a sidebar on the left with links for Home, Community, Collections, KentX, Tags (which is the active tab), Profile, Ask a question, and Logout. The main area displays cards for different tags: react, kent, python, java, javascript, angular, ksu, c, py, frontend, and kent. Each card includes a small icon, a brief description, and the number of questions associated with that tag. To the right, there are sections for 'Buzzing Questions', 'Events & Hackathons', and 'Coding Challenge of the Day'.

• Profile Page:

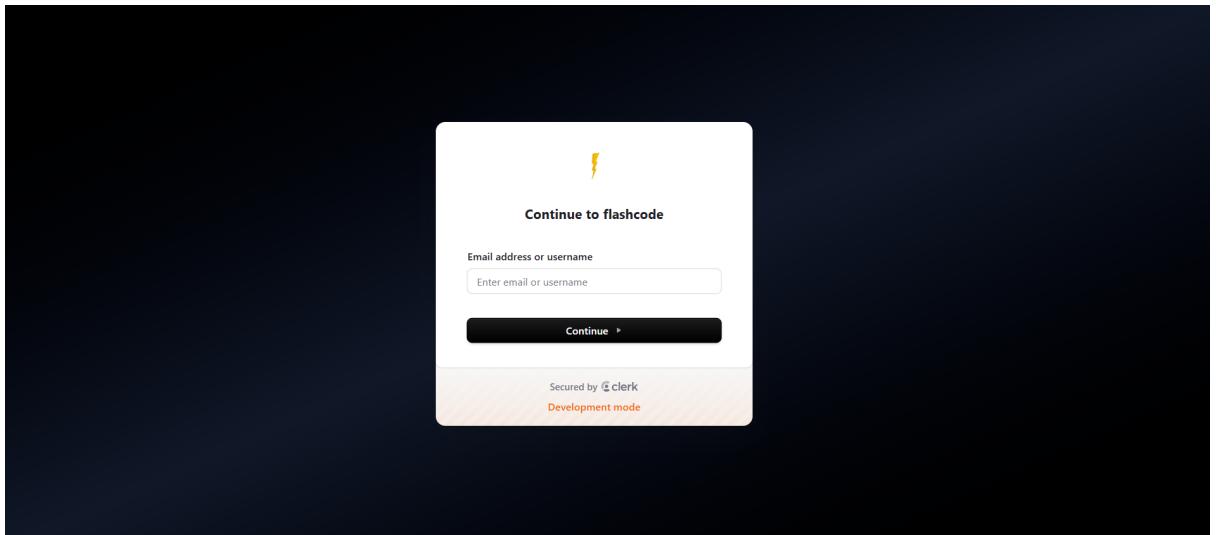
The screenshot shows the profile page for user @rakeshjr. It includes a profile picture, the username, a creation date (April 2025), and edit profile and stats buttons. Below the stats, there are three badge boxes: 'Blue Flash Badge' (0), 'Golden Eagle Badge' (0), and 'Founders Torch Badge' (0). The 'Top Posts' and 'Answers' tabs are visible. A central message says 'Ahh, No Questions Yet!' with a squirrel icon. To the right, there are sections for 'Top Tags For You', 'Buzzing Questions', 'Events & Hackathons', and 'Coding Challenge of the Day'.

• Daily Challenges:

The screenshot shows the 'Flashes - Code Challenge' section. It features a sidebar with links for Home, Community, Collections, KentX, Tags, Profile, Ask a question, and Logout. The main area has a search bar and a 'Change Question' button. A specific challenge titled 'Array Length' is shown, with the task description: 'Get length of [1,2,3]. Expected output: 3'. Below it is a large input field for code, with a 'Run Code' button. At the bottom, there's an 'Output' field showing 'No output yet.' To the right, there are sections for 'Popular Tags', 'What is duck typing in Python?', 'Grounded Theory', 'Why we should choose nextjs instead of reactjs?', 'How does React handle state management? What are some...', 'Events & Hackathons', and 'Coding Challenge of the Day'.

ADMIN DASHBOARD:

- Login/Signup page:

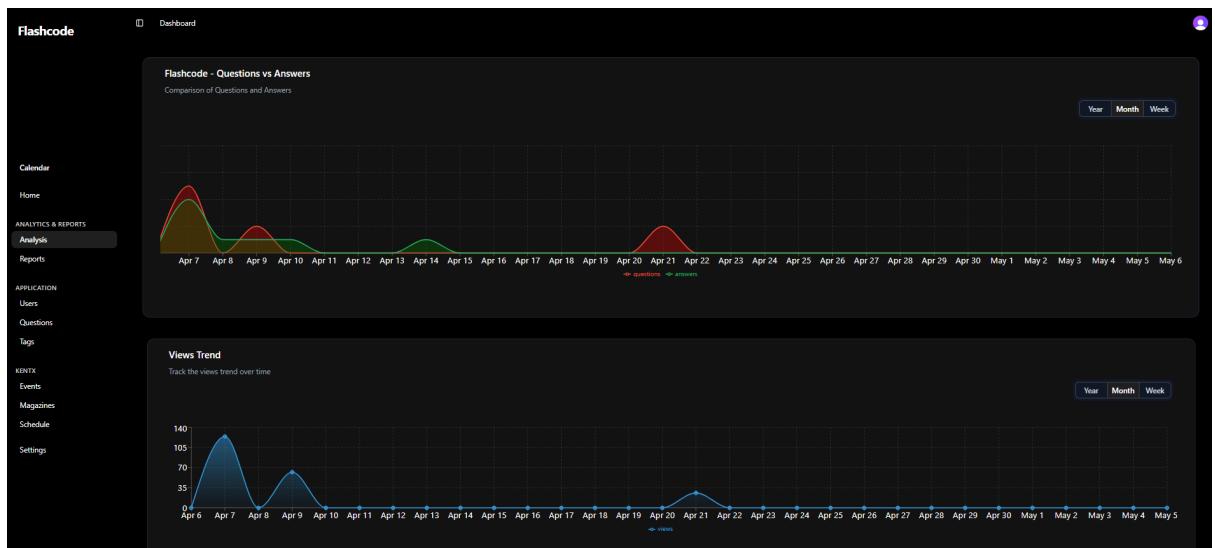


- Home Page:

A screenshot of the Flashcode dashboard. The top navigation bar shows "Flashcode" and "Dashboard". The main header "Flashcode" has a subtitle "Built for better". On the left, there's a sidebar with sections for "Calendar", "Analytics & Reports" (with "Home" selected), and "APPLICATION" (with "Users", "Questions", and "Tags"). The main content area has four cards: "Total Users" (30, -92.9%), "Total Questions" (32, -100.0%, "Declining question activity"), "Total Answers" (23, -100.0%, "Declining answer activity"), and "Answer Growth" (53.1%, "53.1% of questions are answered, decent engagement!"). Below these is a section titled "Flashcode Leaderboard: Top Contributors & Activity Metrics" with a table showing top contributors. The table has columns: RANK, NAME, EMAIL, ANSWERS, ACCOUNT AGE, and LAST ACTIVE. The data is as follows:

RANK	NAME	EMAIL	ANSWERS	ACCOUNT AGE	LAST ACTIVE
1	demo	demo1@gmail.com	0	1 month ago	N/A
2	user	user1@gmail.com	0	1 month ago	N/A
3	Nani	jknithin012@gmail.com	0	1 month ago	N/A
4	SRINIVAS	srinivas10@gmail.com	0	1 month ago	N/A
5	Nithin	njamich@kent.edu	0	1 month ago	N/A
6	Chintu	beaanme@gmail.com	0	1 month ago	N/A
7	JK. Nithin kumar	jknithin0332@gmail.com	0	1 month ago	N/A
8	JK. Nithin Kumar	jknithin36@gmail.com	0	1 month ago	N/A

- Analysis:



• Application:

User Management							
Manage registered Flashcode users, monitor engagement, and perform actions.							
<input type="text" value="Search users..."/> Export to CSV							
S.No	Name	Email	Username	Total Questions	Created Date	# of Answers	Modify
1	JK.Nithin Kumar	jnithin36@gmail.com	jnithin36@	5	2025-03-21	4	...
2	JK.Nithin kumar	jnithin0332@gmail.com	nani1234	11	2025-03-21	2	...
3	Shiva	shivaprasad@gmail.com	shiva1	0	2025-03-24	0	...
4	Chintu	bessaar@gmail.com	shyan90	0	2025-03-24	0	...
5	Nani	jnithin017@gmail.com	nani9332	1	2025-03-24	1	...
6	Nithan	njanmich@kent.edu	nani098	1	2025-03-24	5	...
7	Srinivaskadalal	kadaloriniva2000@gmail.com	srinivas	5	2025-03-25	2	...
8	SRINIVAS	srinivas1@gmail.com	srinivas1	0	2025-03-25	0	...
9	demo	demo1@gmail.com	demo1	0	2025-03-25	0	...
10	user	user1@gmail.com	user1	0	2025-03-25	0	...

Page 1 of 3 Previous Next

Question Insights						
Browse, analyze, and manage all questions posted on Flashcode. Monitor engagement with upvotes, answers, and actions.						
<input type="text" value="Search questions..."/> Export to CSV						
S.No	Question	# of Answers	Created By	Upvotes	Downvotes	Modify
Q001	How to optimize performance when rendering a large list in React?	1	JK.Nithin Kumar	0	0	...
Q002	Why does React log everything twice in development mode?	2	JK.Nithin Kumar	1	0	...
Q003	What transportation options are available for students at Kent State University?	0	JK.Nithin kumar	0	0	...
Q004	What academic programs is Kent State University best known for?	0	JK.Nithin kumar	0	0	...
Q005	Why do we use StringBuilder instead of String for concatenation in Java?	0	JK.Nithin kumar	0	0	...
Q006	What is the difference between == and .equals() in Java?	1	JK.Nithin kumar	0	0	...
Q007	What's the difference between a list and a tuple in Python?	0	JK.Nithin kumar	0	0	...
Q008	How do I create a REST API in Spring Boot?	1	JK.Nithin kumar	0	0	...
Q009	How do I connect a Flask app to a SQLite database?	1	JK.Nithin kumar	0	0	...
Q010	What are the key differences between React and Angular in terms of architecture and learning curve?	0	JK.Nithin kumar	0	0	...

https://admin-flashcode.vercel.app/questions 4 Previous Next

Flashcode

Tags Overview
Analyze and manage all coding language tags used in Flashcode. View usage distribution, search tags, and export tag data.
Total Tags in System: 36

Calendar
Home
ANALYTICS & REPORTS
Analysis
Reports
APPLICATION
Users
Questions
Tags
KENTX
Events
Magazines
Schedule
Settings

All Tags List
Search tags... Export to CSV

S.No	Tag	Questions
1	quantumphysics	1
2	science	1
3	cassini	1
4	mba	1
5	Culinary study	1
6	interview	1
7	general	1
8	indexing	1
9	list	1
10	hello	1

Previous Page 1 of 4 Next

Tag Distribution
Breakdown of top-used tags across all questions.

Tags Missing Icons
The following tags don't have associated icons.

S.No	Tag	Questions
1	astro	3
2	bubble	5
3	qsharp	2

● KentX:

Flashcode

Manage Events
Add and manage upcoming Flashcode events

Calendar
Home
ANALYTICS & REPORTS
Analysis
Reports
APPLICATION
Users
Questions
Tags
KENTX
Events
Magazines
Schedule
Settings

Add New Event

Title: Hackathon 2025
Date: dd-mm-yyyy 07:00 PM
Location: Student Union Hall
Description: Tech fest full of innovation
Link: https://eventsite.com

Add Event

All Events
Loading...

Flashcode

Magazine Management
Easily upload, view, and manage your organization magazines. Provide links, descriptions, and cover images for students and readers to explore.

Calendar
Home
ANALYTICS & REPORTS
Analysis
Reports
APPLICATION
Users
Questions
Tags
KENTX
Events
Magazines
Schedule
Settings

Add a New Magazine
Fill out the form below to upload a new magazine to Flashcode.

Magazine Name
Magazine Description
Magazine Link (https://...)
Choose File No file chosen
Add Magazine

Did You Know?

- The first magazine was published in 1731.
- National Geographic launched in 1888.
- Magazines still influence over 18 global readers.

All Magazines

A Magazine
Focuses on fashion, beauty, and culture, providing the latest industry news and trends. Distributed at Kent State and the Fashion School's New York studio.
Read More Delete

The Burr Magazine
Kent State's first student-produced magazine, offering compelling stories on current events, trends, and campus life.
Read More Delete

Luna Negra
The university's literary arts magazine, featuring poetry, prose, photography, and illustrations.
Read More Delete

Uhuru Magazine
An award-winning magazine promoting innovation, creativity, and culture through the voices of minority and underrepresented groups.
Read More Delete

<https://admin-flashcode.vercelapp/magazines>

- **Settings:**

11. Appendix:

GitHub repo –

<https://github.com/Srinivaskadalal/flashcode>

clone the repo

For Main Application –

cd client

npm intsall

npm run dev

for server

pip install -r requirements.txt

flask run

for Dashbaord

cd admin

cd client

npm I

npm run dev

Live Links –

<http://cassini.cs.kent.edu:8010/>

admin dashboard –

<https://admin-flashcode.vercel.app/>

