

# Brain Jam: Code Clash Arena

## Executive Summary

**Brain Jam** is an advanced, university-level competitive programming platform designed to nurture algorithmic thinking and promote collaborative coding through real-time matches. Inspired by industry leaders such as Codeforces, LeetCode, and CodeChef, Brain Jam introduces a unique military-rank-based leveling system, engaging multiplayer duels, and intelligent progress tracking to enhance the coding experience.

---

## Core Features

### 1. Structured Problem Sets

**Rank-Based Difficulty Levels:** -  **Private Recruit** (Beginner) -  **Cadet Coder** (Novice) -  **Code Corporal** (Intermediate) -  **Tech Lieutenant** (Advanced) -  **Algorithm Captain** (Expert) -  **Legendary General** (Elite)

Each level includes a curated collection of challenges, categorized and managed by platform administrators. Problems span various algorithmic domains such as dynamic programming, graph theory, string manipulation, and more.

### 2. Intelligent Matchmaking System

**Match Modes:** - **1v1 Challenge:** Direct invitation to a coding duel. - **Open Arena:** Public lobbies where multiple users can participate.

**Workflow:** 1. Host configures match settings (difficulty, duration, number of problems). 2. Participants join via unique code or invitation link. 3. Problems are randomly selected based on difficulty. 4. Real-time countdown initiates match. 5. Submissions are evaluated instantly by the auto-judging engine. 6. Results and winner rankings are displayed upon completion.

### 3. Automated Judging System

- Utilizes secure sandbox environments (Docker or Sphere Engine API).
- Evaluates code against predefined test cases.
- Provides immediate feedback (pass/fail status, error messages).
- Supports multiple programming languages (Python, C++, Java).

### 4. Comprehensive User System

- Secure authentication and profile management.
- Add and invite friends for matches.
- Maintain history of matches and problem-solving activity.

- Earn rating points and unlock higher ranks.

## 5. Dynamic Leaderboard

- Displays global and rank-specific performance.
- Rankings based on user rating (Codeforces-style), problem-solving record, and match outcomes.
- Weekly and monthly top performer highlights.

## 6. Rating & Rank Progression System

**Adopts an Elo-style rating model similar to Codeforces.**

Action	Rating Impact
Solve Easy Problem	+1 to +3 rating
Solve Medium Problem	+5 to +8 rating
Solve Hard Problem	+10 to +15 rating
Win a Match vs Equal Rank	+20 rating
Win a Match vs Higher Rank	+30 to +40 rating
Lose a Match vs Lower Rank	-15 to -25 rating
Draw a Match	±5 rating (based on rank difference)

**Initial Rating:** 800

**Rank Distribution:** - **Private Recruit:** 800–999 - **Cadet Coder:** 1000–1199 - **Code Corporal:** 1200–1399 - **Tech Lieutenant:** 1400–1599 - **Algorithm Captain:** 1600–1899 - **Legendary General:** 1900+

## 7. Interactive Dashboard & Analytics

**Graphical Elements:** - **Rating Progress Graph:** Line graph showing rating changes over time. - **Radar Chart:** Visual breakdown of user strengths across problem categories. - **Progress Ladder:** Horizontal timeline of rank achievements. - **Promotion Animations:** Engaging visual effects triggered upon level-up.

## 8. Administrative Control Panel

- Upload, edit, and manage problems and their respective difficulty levels.
- Define and validate test cases.
- Monitor user activity and handle reported content.

## Technical Architecture

Layer	Technology Stack
Frontend	React.js, Tailwind CSS
Backend	Node.js with Express.js / Django
Database	PostgreSQL / MongoDB
Authentication	Firebase, Auth0, or JWT-based system
Realtime Communication	Socket.io / Django Channels
Judging Engine	Docker-based sandbox / Sphere Engine API
Data Visualization	Recharts, Chart.js, or D3.js

## Roadmap & Future Enhancements

- Integration of team-based matches.
- AI-driven hint system using LLMs (e.g., OpenAI GPT).
- Advanced filtering and tagging for problems.
- Unlockable achievement badges and coding streaks.
- Native mobile application development.

## Deployment Infrastructure

- **Frontend Hosting:** Vercel, Netlify
- **Backend Deployment:** Heroku, Railway, AWS EC2
- **Database Hosting:** Supabase, MongoDB Atlas
- **Domain & SSL:** Namecheap, Cloudflare

## Acknowledgements

- Developed by: Final Year Computer Science Project Team, 2025
- Supervised by: [Faculty Advisor Name]
- Special Thanks: [University Name] Computer Science Department

## License

**MIT License** – Open source for educational and personal use. Commercial use requires explicit permission.