

# BragBoard: Internal Employee Recognition Wall

**Project Statement:** BragBoard is an internal recognition tool for workplaces that lets employees appreciate their colleagues by posting shout-outs. It promotes a culture of appreciation by allowing tagging, commenting, and company-wide visibility. Admins can monitor engagement and moderate flagged content.

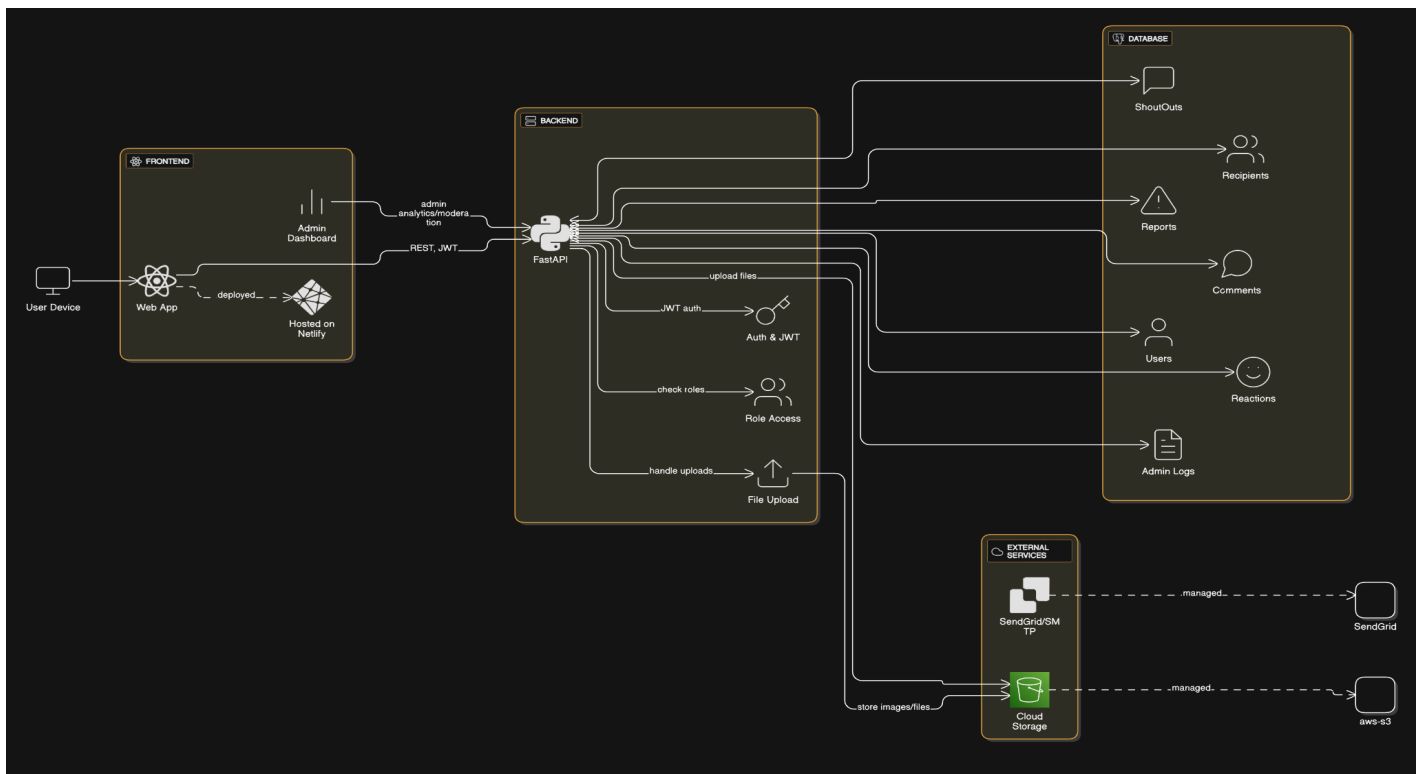
## Key Features:

1. Employees can send shout-outs to teammates
2. Recipients can be tagged (multi-select)
3. Reactions: like, clap, star
4. Commenting system for shout-outs
5. Admin dashboard for insights and moderation

## Tech Stack:

- Frontend: React.js + Tailwind CSS
- Backend: FastAPI
- Database: PostgreSQL
- Authentication: JWT (access + refresh)

## Architecture Diagram:



## Modules:

- Module A: User Management & Authentication
  - Module B: Shout-Out Creation & Feed
  - Module C: Tagging, Reactions & Comments
  - Module D: Admin Tools & Analytics
- 

## 8-Week Milestone Plan

### Milestone 1: Weeks 1–2 – User System & Basic UI

#### Week 1:

- Set up project structure (React + FastAPI)
- Create database models for users
- Implement user registration/login with JWT

#### Week 2:

- Build a basic layout with Tailwind CSS.
- Display the user dashboard after login.
- Set up department-wise scoping in the backend.

#### Expected Output:

- Secure auth with login/register
  - Basic React layout and routing
  - User role & department info captured
- 

### Milestone 2: Weeks 3–4 – Shout-Out Posting & Feed

#### Week 3:

- Create a shout-out form with recipient tagging
- Store shout-out in DB with tagged users

#### Week 4:

- Display all shout-outs on the feed.
- Filter by department, sender, date
- Attachments or image uploads (optional)

#### Expected Output:

- Shout-out post from functional

- Feed showing all shout-outs by tag/time
- 

### **Milestone 3: Weeks 5–6 – Reactions & Comments**

#### **Week 5:**

- Add reactions (like, clap, star) to posts
- Reaction counters + user-specific tracking

#### **Week 6:**

- Add a commenting system under shout-outs
- Show avatars, timestamps, and nesting (optional)

#### **Expected Output:**

- Reaction buttons with live updates
  - Comments with user info + moderation logic
- 

### **Milestone 4: Weeks 7–8 – Admin Tools & Analytics**

#### **Week 7:**

- Admin dashboard: top contributors, most tagged
- Ability to delete posts/comments
- View and resolve reported shout-outs

#### **Week 8:**

- Export reports (PDF/CSV)
- Leaderboard for gamified appreciation
- Deployment (if needed) + testing & polish

#### **Expected Output:**

- Admin panel with moderation tools
  - Downloadable reports
  - Final working system with styling & QA
- 

**Expected Project Outcome:** By the end of Week 8, BragBoard will support:

- JWT-authenticated employee logins
- Posting, tagging, and reacting to shout-outs
- Comment threads and real-time feed updates
- Admin oversight and analytics reports

## Database Schema:

### Users:

- id (INT, PK)
- name (VARCHAR)
- email (VARCHAR, UNIQUE)
- password (VARCHAR)
- department (VARCHAR)
- role (ENUM: 'employee', 'admin')
- joined\_at (TIMESTAMP)

### ShoutOuts:

- id (INT, PK)
- sender\_id (FK to [Users.id](#))
- message (TEXT)
- created\_at (TIMESTAMP)

### ShoutOutRecipients:

- id (INT, PK)
- shoutout\_id (FK to [ShoutOuts.id](#))
- recipient\_id (FK to [Users.id](#))

### Comments:

- id (INT, PK)
- shoutout\_id (FK to [ShoutOuts.id](#))
- user\_id (FK to [Users.id](#))
- content (TEXT)
- created\_at (TIMESTAMP)

### Reactions:

- id (INT, PK)
- shoutout\_id (FK to [ShoutOuts.id](#))
- user\_id (FK to [Users.id](#))
- type (ENUM: 'like', 'clap', 'star')

### Reports:

- id (INT, PK)
- shoutout\_id (FK to [ShoutOuts.id](#))
- reported\_by (FK to [Users.id](#))
- reason (TEXT)

- created\_at (TIMESTAMP)

## AdminLogs:

- id (INT, PK)
- admin\_id (FK to Users.id)
- action (TEXT)
- target\_id (INT)
- target\_type (VARCHAR)
- timestamp (TIMESTAMP)

