**Employee Gamification Web Application**

The Employee Gamification App is a full-stack web application that provides a platform for managers and administrators to monitor, motivate, and engage employees through the use of gamification strategies. The app enables team and task management, performance tracking, and dynamic reporting.

**3. Objective**

To build an interactive and modular web platform that helps:

* Improve employee productivity and engagement
* Streamline task and absence management
* Visualize performance and activity using reports and leaderboards
* Integrate gamification through badges and points

|  |  |
| --- | --- |
| Frontend | React.js |
| Backend | Django + Django REST Framework (PYTHON) |
| Database | MySQL (cu MySQL Workbench) |
| Tools | VS Code, GitHub |

|  |  |
| --- | --- |
| Authentication | Login and register system with user roles (Admin, Manager, Employee) |
| Role-based Permissions | Specific access rights depending on the user's role |
| Team Management | Creation and assignment of employees to teams |
| Task Management | Assigning, editing, prioritizing, and tracking tasks |
| Task Priorities & Status | Each task has a priority (Low, Medium, High, Urgent) and a status (To Do, In Progress, Done) |
| Badges and Points | Employees earn badges/points when completing tasks or milestones |
| Projects | Group related tasks under larger projects |
| Work Schedule | Weekly plans and shift assignments per employee |
| Absence Calendar | Employees can request leave and log sick days or vacations |
| Reports | Statistical reports for task completion, activity, and performance |
| Leaderboard | Rank employees by badges, points, or completed tasks |

|  |  |
| --- | --- |
| Admin | Full system access. Manages users, roles, projects, teams, and settings |
| Manager | Manages tasks, teams, work schedules, and tracks performance |
| Employee | Views and completes assigned tasks, tracks progress, requests absence |

ROLES

**Architecture**

**Client (React)** → makes requests to → **API Server (Django/DRF)** → communicates with → **MySQL Database**

**10. Possible Improvements**

* Real-time notifications via WebSocket
* Export reports to PDF or Excel
* Mobile app version using React Native
* Advanced analytics (charts, KPIs)

| **Funcționalitate** | **Admin** | **Manager** | **Employee** |
| --- | --- | --- | --- |
| Vizualizare toate echipele | ✅ | ✅ | ✅ |
| Vizualizare membri echipă proprie | ✅ | ✅ | ✅ |
| Creare echipă | ✅ | ✅ | ❌ |
| Ștergere echipă | ✅ | ❌ | ❌ |
| Adăugare membru | ✅ | ✅ (în echipa proprie) | ❌ |
| Ștergere membru | ✅ | ✅ (cu aprobare admin) | ❌ |
| Mutare membru între echipe | ✅ | ❌ | ❌ |
| Cerere de alăturare la echipă | ❌ | ✅ (pentru aprobare) | ✅ |
| Aprobare/Refuz cerere de alăturare | ✅ | ❌ | ❌ |

Permisiuni Team management

**✅ UI/UX Idei**

Pagina cu carduri pentru fiecare echipă

Badge pentru roluri (Manager, Member)

Butoane dinamice: „Add Member”, „Request to Join”, „Remove” etc. doar dacă ai permisiuni

Modale pentru confirmări (ștergere, mutare etc.)

Altele ++

1. **Notificări sau "Pending Requests" pentru Admin** – când un manager propune ștergerea unui membru sau un angajat cere alăturare.
2. **Istoric modificări echipă** – cine a adăugat/scos membri și când.
3. **Rol de "Team Lead"** (opțional) – un membru din echipă desemnat ca lider fără a fi manager general.
4. **Limite/Reguli la echipe** – max. X membri, obligatoriu minim 1 manager etc.
5. **Tema proiectului vizibilă** pentru Employees la fiecare echipă (inspiră alegerea unei echipe).

PT TASK

**3. Sistem de feedback pentru taskuri**

* După finalizare, managerul poate da rating/feedback pentru calitatea taskului
* Employee vede scorul și sugestii (motivațional)

**5. Calendar Integrat**

* Să vezi vizual în ce zile sunt taskuri, cine e în concediu, ce proiecte sunt în derulare

**6. Upload de fișiere/documente la taskuri**

* Poți atașa fișiere (PDF-uri, rapoarte, imagini) la un task sau proiect

**9. Export rapoarte în PDF sau Excel**

* Adminul/managerul poate genera rapoarte lunare: taskuri finalizate, cine a lipsit, top performeri etc.

**10. Teme și personalizare UI**

* Modul dark/light
* Fiecare echipă poate avea o „temă” vizuală (culoare, logo etc.)

TODO TODAY: 16/04/2025

Unul care sa dea start task, sa se stie ca angjatul respectiv lureaza la acel task, si se va pune cu statusul In progress iar dupa sa poata da finish buotnul pe care il am deja si sa treaca in verification, de asemenea mai trebuie sa implementez asa: DONE

După finalizare, managerul poate da rating/feedback pentru calitatea taskului Employee vede scorul și sugestii (motivațional) : DONE

Upload de fișiere/documente la taskuri Poți atașa fișiere (PDF-uri, rapoarte, imagini) la un task sau proiect DONE

6. Trivia and game quizzes

Cand se adauga un nou utilizator (sign in), pentru fiecare utilizator trebuie sa se genereze un salt criptografic care este o cantitate generate aleator odata cu creearea utilizatorului.

Aceasta cantitate generata aleator (salt-ul) se concateneaza cu parola, iar concatenarea lor va fi intrare pentru o functie hash (sha256).

Nume utilizator (id), salt, hash (salt concatenate cu parola)

Exemplu: Gabi, 123/$^$, 32bytesrandom

**2. Ce face codul:**

* Pentru fiecare request adaugă tokenul de acces
* Dacă primește 401 Unauthorized, încearcă să refacă request-ul după ce reînnoiește token-ul cu refresh token-ul
* Dacă refresh-ul e invalid sau expirat, te redirecționează automat la login
* Gestionează toate cererile care se pot executa paralel când se reînnoiește token-ul, fără să trimită refresh token de mai multe ori simultan

| **Funcționalitate** | **Cine o folosește** | **Descriere** | **UI / Componentă propusă** | **Prioritate** |
| --- | --- | --- | --- | --- |
| **Grid orar** | Manager / Admin / Angajat | Tabel cu zilele săptămânii pe coloane și orele 09:00–18:00 pe rânduri. | <ScheduleGrid> cu celule editabile | 🔥 Urgent |
| **Asignare shift** | Manager / Admin | Click pe celulă pentru a selecta un angajat din dropdown și a-l asigna la acea oră/zi. | Dropdown în celulă, tooltip cu detalii angajat | 🔥 Urgent |
| **Drag & Drop** | Manager | Mutare rapidă a unui shift prin drag&drop între celule/oră/zi. | React DnD sau similar | ⭐ Mediu |
| **Solictare schimb** | Angajat | Buton pe celulă „Request swap” care deschide modal pentru alegerea orelor alternative. | <SwapModal> + notificare manager | ⭐ Mediu |
| **Modificare proprie** | Angajat | Butoane „Move me” sus/jos (schimbă ora cu +1h / -1h) și „Change day” (mută shiftul în altă zi). | Butoane + dropdown de zi | ⭐ Mediu |
| **Aprobări & notificări** | Manager / Admin | Flux de aprobare pentru cereri de schimb (+ notificări email/in-app). | Inbox cereri + badge-uri (🔔) | ⭐ Mediu |
| **Vizualizare personalizată** | Toate rolurile | Filtre: doar propriu, doar propria echipă, toți angajații. | Butoane / dropdown de filtrare | 🔥 Urgent |
| **Recurring shifts** | Manager / Admin | Setare cicluri: zilnic, săptămânal, lunar pentru un anumit angajat. | Modal „Set Recurrence” | ⭐ Scăzut |
| **Export / Print** | Manager / Admin | Export orar în CSV / PDF / imprimare directă. | Buton „Export” / „Print” | ⭐ Scăzut |
| **Sincronizare calendar** | Angajat | Sincronizare one-click în Google Calendar / Outlook. | Buton „Sync Calendar” | ⭐ Scăzut |
| **Color-coding** | Toate rolurile | Culoare diferită pentru tipuri de shift (normal, overtime, liber). | Legendă + color picker în celulă | ⭐ Mediu |
| **Overtime warning** | Manager / Admin | Afișează semnală (roșu) când se alocă ore extra peste 18:00. | Validare pe click + alertă | 🔥 Urgent |
| **Disponibilitate angajați** | Angajat | Angajații își marchează zile/ore preferate sau indisponibilități. | Calendar de availabilities | ⭐ Mediu |
| **Dashboard sumar** | Manager / Admin | Grafice simple: ore lucrate pe zi/săpt, distribuție echipe. | ChartJS / Recharts widget | ⭐ Scăzut |
| **Mobile-friendly view** | Toate rolurile | Aspect adaptat pe mobil: listă pe zile, swipe între zile. | Responsive breakpoints + swipe gestures | ⭐ Mediu |

Deci pagina mea de WorkSchdule va continue urmatoarele functinalitati:

Trebuie sa contina un table cu zilele saptamanii pe coloane si orele de la 09:00 la 18:00 pe randuri,

Asignare shift adica Manager-ul SI ADMINUL sa poata da click pe celula pentru a selecta un angajat din dropndown si a-l asigna la acea ora/zi, sa se salveze angajatul acolo ca acesta sa poata vedea cand lucreaza.

Drag & drop, managerul si adminul sa poata muta rapid un shift prin drag and drop intre cellule/ore/zi

Solicitare schimbare angajat, adica angajatul sa aibe un buton pe celula “REQUEST SWAP” care deschide modal pentru alegerea orelor alternative, insa trebuie aprobarea managerului oentru mutare.

Modificare proprie angajat, adica butoane move me sus/jos adica sa schimbe ora cu +1h/ -1h, oferind posibilitatea sa incepe serviciu la 10 si sa termine mai tarziu de exemplu,

Filtre: Un anagajat poate sa vada shift-ul lui si cel al membrilor cu care acesta este in echipa.

Color-coding: culoare diferita pentru tipurile de shift ( normal,overtime, liber),

Deci backend: 1. title, description, category (daily, weekly, team, special), xp\_reward, points\_reward, badge\_reward -> care vor aparea in pagina de badges & points mai tarziu, start\_date, due\_date, is\_active Model UserQuest : status, progress de la (0-5) model badge va fii pe viitor nume iconita descriere afisare puncte etc .3. Quest Detail Modal Componentă: <QuestDetailModal> Conţinut: Header: icon + titlu + categorie Descriere: ce trebuie să faci (+ checklist pas cu pas) Progres: bara de progres + % Rewards: XP + points + badge preview Acţiune: buton “Start” / “Continue” / “Complete” Timer (pentru daily/weekly quests) 8. Elemente vizuale și animații Progress Bar animat Badge pop-up când atingi un prag de XP (ex: level up) Tooltips pentru explicații scurte (de ex. “XP pentru completarea acestui quest”) Dark mode toggle

ok acum urmatorul lucru, ar fi sa implementam pagina BadgesPoints.js si backend-ul pentru aceasta,

Am asa: 4 badge-uri pentru fiecare:

4 pentru xp acumulat,

4 pentru task-uri acumulate

4 pentru puncte acumulate, dupa cum se vede si in imagine, acestea badge-uri vor fi acordate atunci cand utilizatorul:

Pentru cele de xp:

Level 1 incepe cu badge denumit "xp1" , de la level 10 in sus, acesta trebuie sa primeasca badge-ul denumit "xp2", de la level 20 in sus, acesta sa primeasca badge-ul denumit "xp3" iar de la level 30 in sus, pana la maxim sa ramana mereu badge-ul denumit "xp4".

Pentru task-uri, 0 task-uri nu are badge, de la 1task rezolvat pana la 14 task-uri, badge denumit "task1", de la 15 la 29 task-uri sa primeasca badge-ul denumit "task2", de la 30 pana la 44 badge-ul denumit "task3" iar peste 45 badge-ul denumite "task4".

Pentru puncte la 0 puncte nu are badge, de la 1 punct pana la 4999 puncte are badge-ul denumit "point1", de la 5000 pana la 9999 are badge-ul denumit "point2" , de la 10000 pana la 19999 are badge-ul denumit " point3" iar tot ce este peste 20000 o sa aibe badge-ul denumit "point4"