Eujim Graduate Engagement Platform

System Documentation

This documentation provides a comprehensive overview of the Eujim Graduate Engagement Platform through a series of UML diagrams that describe the core logic, user interactions, system architecture, and class structure of the platform. The goal is to provide technical and non-technical stakeholders with a clear understanding of the platform's behavior, flow, and responsibilities.

1. Use Case Diagram

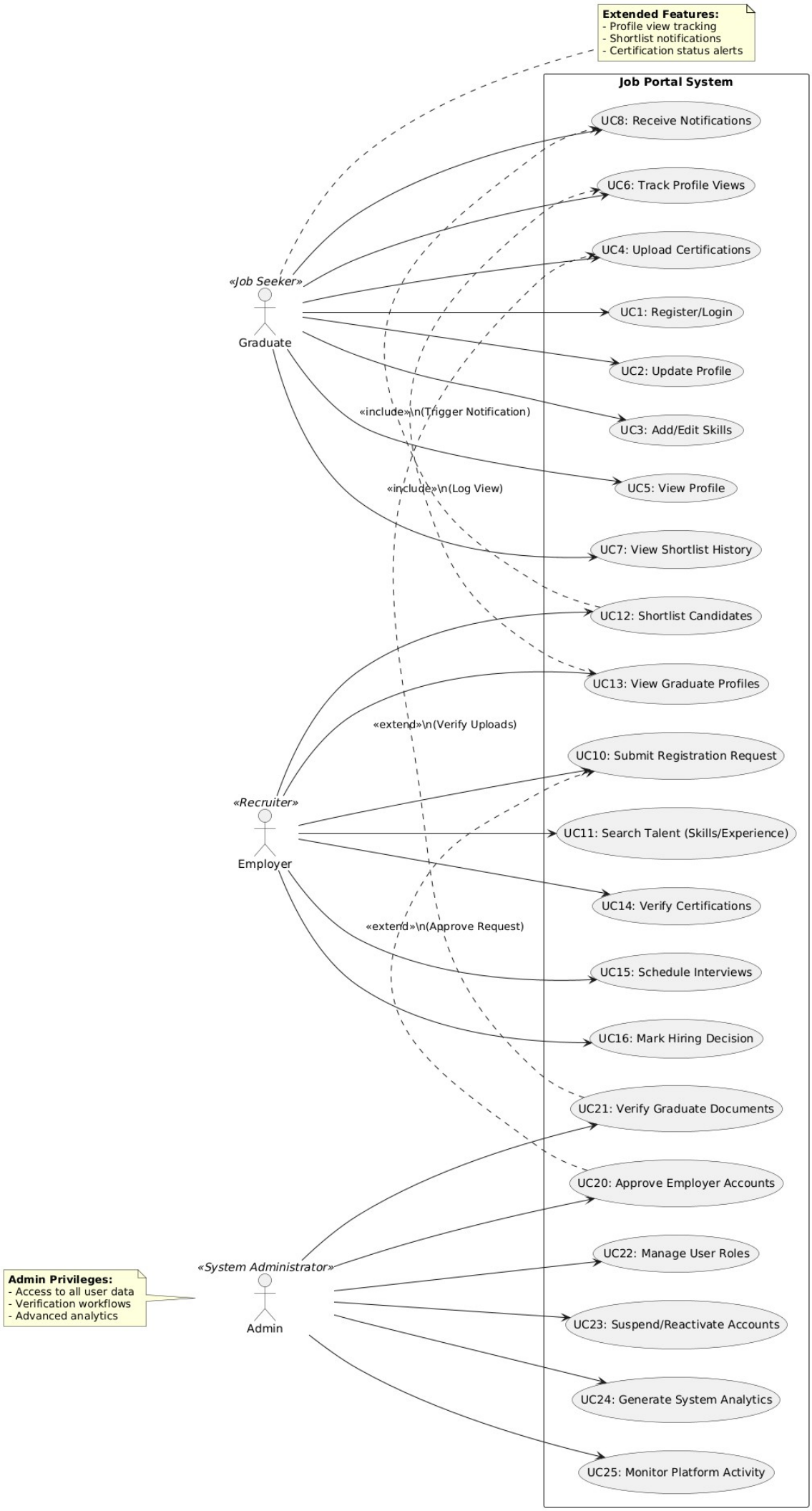
Actors:

- **Job Seeker (Graduate)**
- **Recruiter (Employer)**
- **Admin**

Key Use Cases:

- Graduates can register, update their profile, add skills, and upload certifications.
- Employers submit admission requests, search for graduates by skills, and shortlist candidates.
- Admins verify employer credentials, manage users, and generate insights.

This diagram visualizes the core functionalities accessible to each actor.


Extended Features:
- Profile view tracking
- Shortlist notifications
- Certification status alerts
Job Portal System
UC8: Receive Notifications
UC6: Track Profile Views
UC4: Upload Certifications
«Job Seeker»
Graduate
UC1: Register/Login
UC2: Update Profile
«include»\n(Trigger Notification)
UC3: Add/Edit Skills
UC5: View Profile
«include»\n(Log View)
UC7: View Shortlist History
UC12: Shortlist Candidates
UC13: View Graduate Profiles
«extend»\n(Verify Uploads)
UC10: Submit Registration Request
«Recruiter»
Employer
UC11: Search Talent (Skills/Experience)
UC14: Verify Certifications
«extend»\n(Approve Request)
UC15: Schedule Interviews
UC16: Mark Hiring Decision
UC21: Verify Graduate Documents
UC20: Approve Employer Accounts
«System Administrator»
Admin
UC22: Manage User Roles
UC23: Suspend/Reactivate Accounts
UC24: Generate System Analytics
UC25: Monitor Platform Activity
Admin Privileges:
- Access to all user data
- Verification workflows
- Advanced analytics

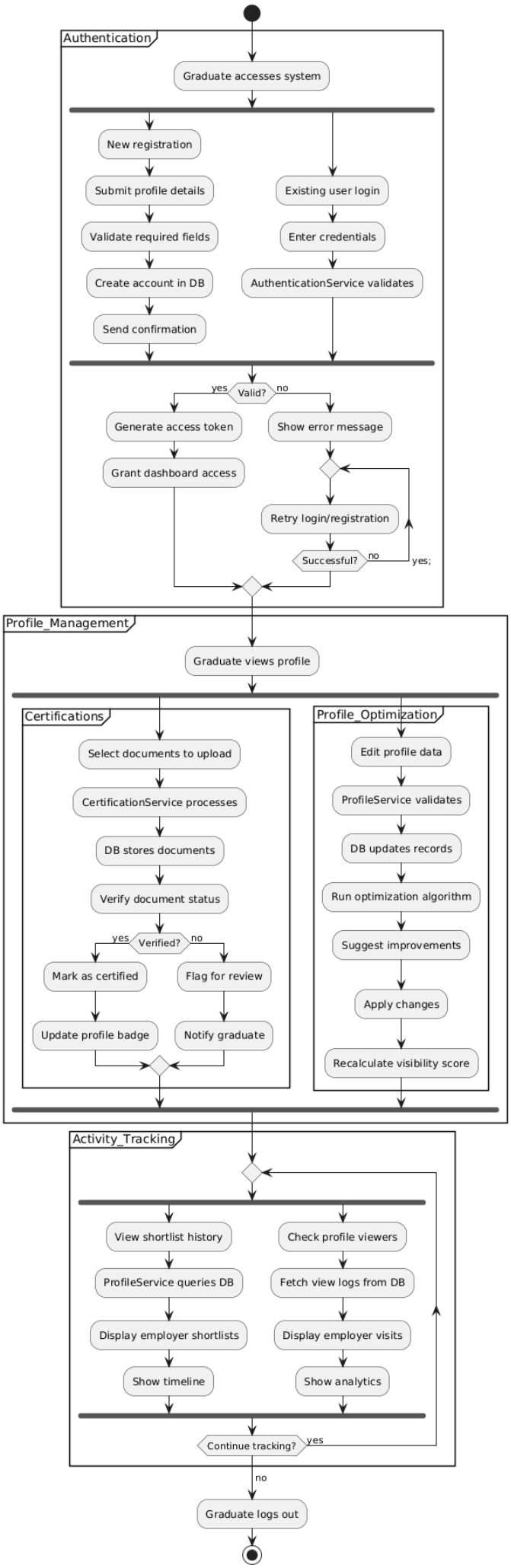
2. Activity Diagrams

a) Graduate Activity Flow

Description: Illustrates the steps a graduate follows in the system:

1. Registers and logs in
2. Fills out profile with bio and social links
3. Adds skills and uploads certifications
4. Becomes searchable to employers
5. Gets notified if shortlisted by a recruiter

Purpose: This diagram helps understand the self-service nature of the graduate experience and the trigger points for engagement.


Graduate User Workflow
Authentication
Graduate accesses system
New registration
Submit profile details
Validate required fields
Create account in DB
Send confirmation
Existing user login
Enter credentials
AuthenticationService validates
Valid?
yes
no
Generate access token
Grant dashboard access
Show error message
Retry login/registration
Successful?
no
yes;
Profile_Management
Graduate views profile
Certifications
Select documents to upload
CertificationService processes
DB stores documents
Verify document status
Verified?
yes
no
Mark as certified
Update profile badge
Flag for review
Notify graduate
Profile_Optimization
Edit profile data
ProfileService validates
DB updates records
Run optimization algorithm
Suggest improvements
Apply changes
Recalculate visibility score
Activity_Tracking
View shortlist history
ProfileService queries DB
Display employer shortlists
Show timeline
Check profile viewers
Fetch view logs from DB
Display employer visits
Show analytics
Continue tracking?
yes
no
Graduate logs out

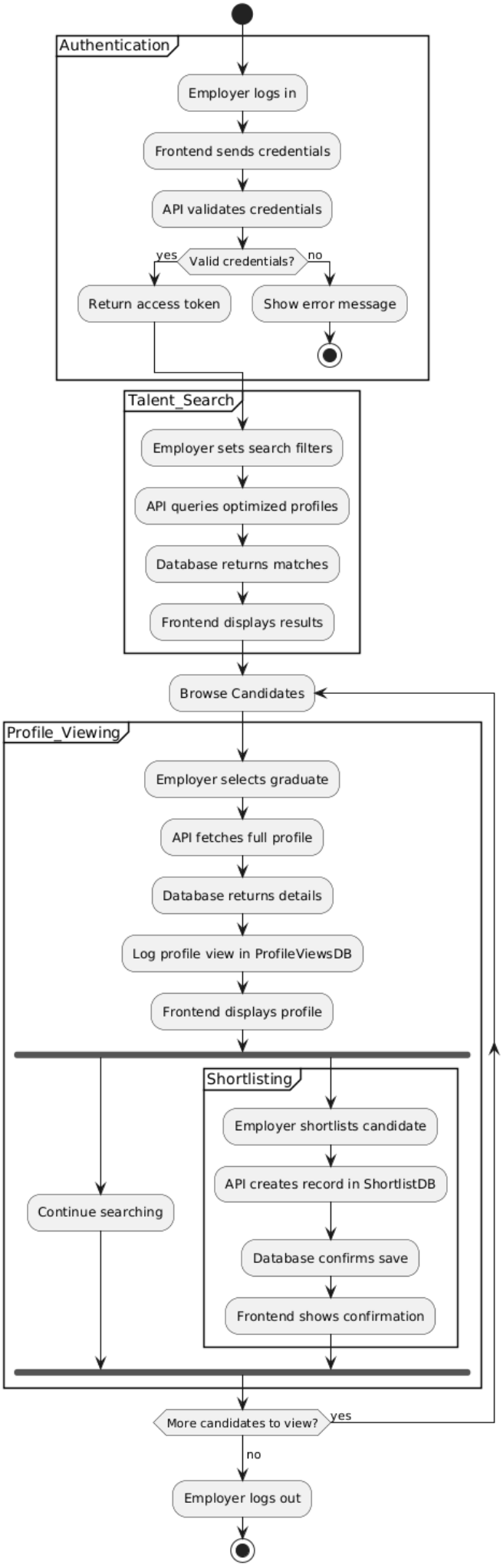
b) Recruiter Activity Flow

Description: Visualizes the workflow for employers:

1. Submit admission request
2. Await approval by admin
3. Login and search for graduates
4. View profiles and shortlist candidates
5. System sends notifications to graduates
6. Recruiter reviews documents and marks final outcomes (hired, rejected, interviewed)

Purpose: Highlights the semi-automated engagement process between employers and graduates.

Employer Workflow - Talent Search and Shortlisting

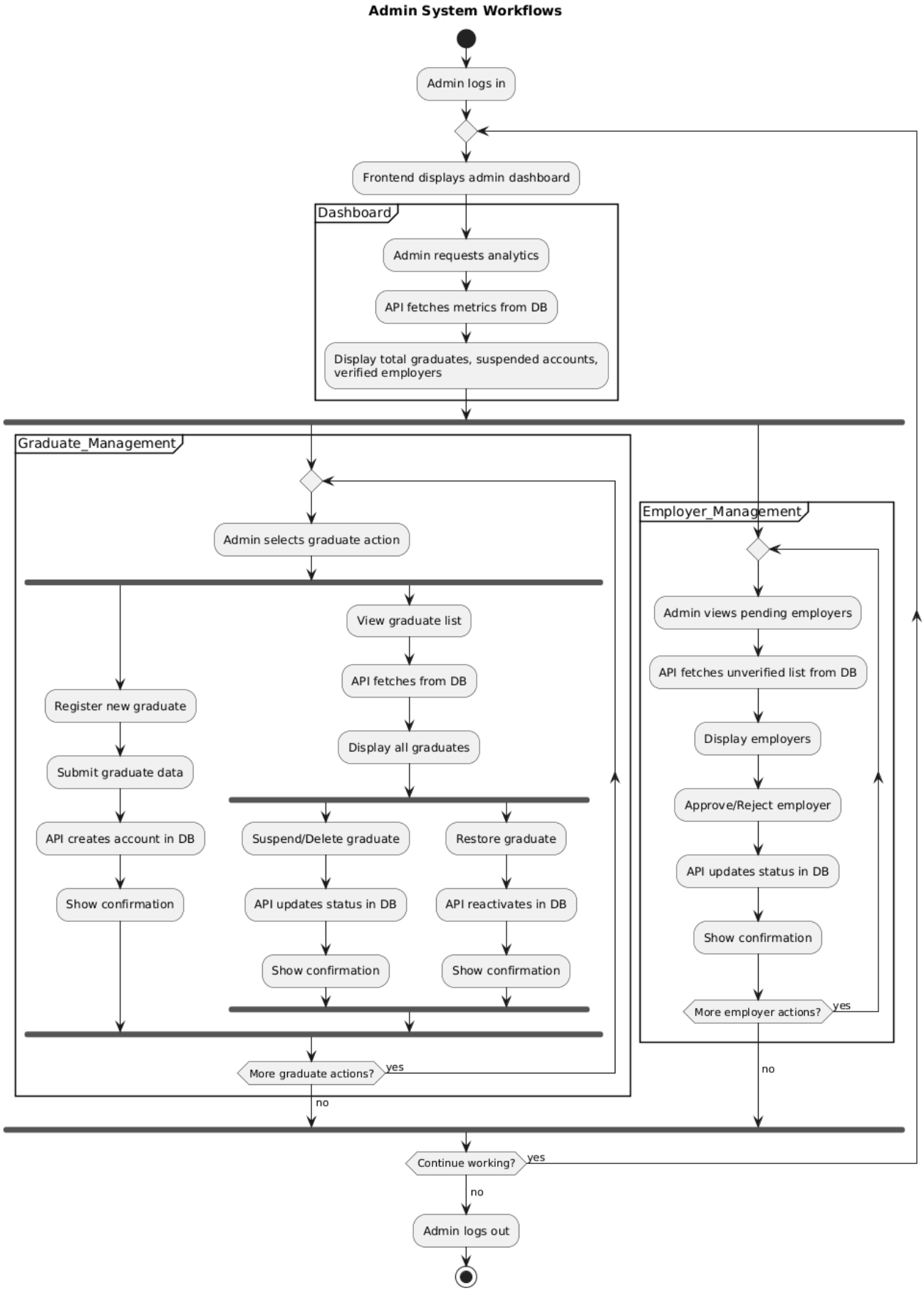


c) Admin Activity Flow

Description: Demonstrates the administrative responsibilities and system moderation flow:

1. Review employer admission requests
2. Approve or reject employer registration
3. Verify recruiter-uploaded documents
4. Monitor graduate certifications and data quality
5. Manage user roles and access levels
6. Generate analytics and reports

Purpose: Provides a clear view of how the admin ensures data integrity, approves participants, and supports operational and strategic functions.

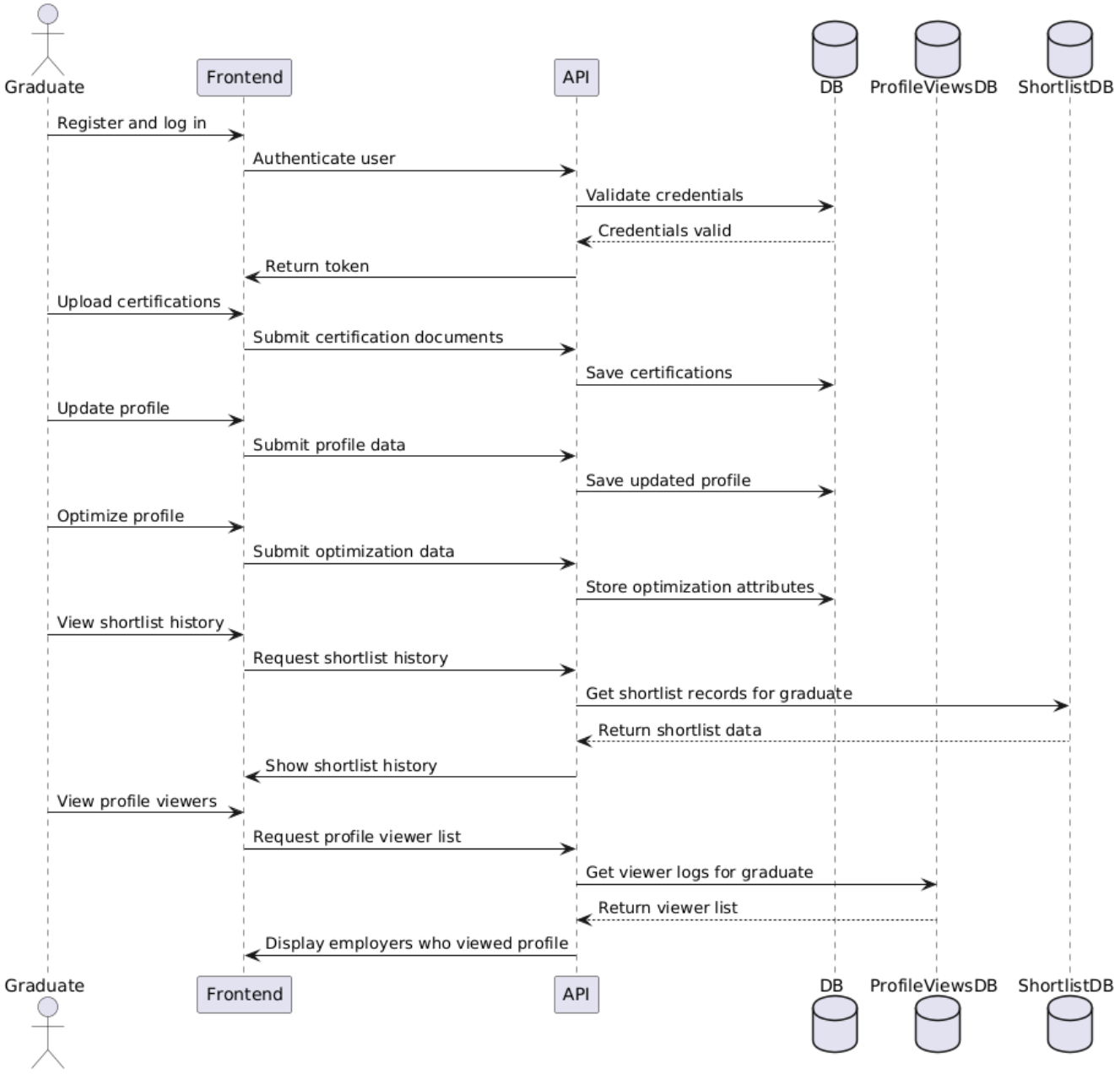

Admin System Workflows
Admin logs in
Frontend displays admin dashboard
Dashboard
Admin requests analytics
API fetches metrics from DB
Display total graduates, suspended accounts, verified employers
Graduate_Management
Admin selects graduate action
Register new graduate
Submit graduate data
API creates account in DB
Show confirmation
View graduate list
API fetches from DB
Display all graduates
Suspend/Delete graduate
API updates status in DB
Show confirmation
Restore graduate
API reactivates in DB
Show confirmation
More graduate actions?
yes
no
Employer_Management
Admin views pending employers
API fetches unverified list from DB
Display employers
Approve/Reject employer
API updates status in DB
Show confirmation
More employer actions?
yes
no
Continue working?
yes
no
Admin logs out

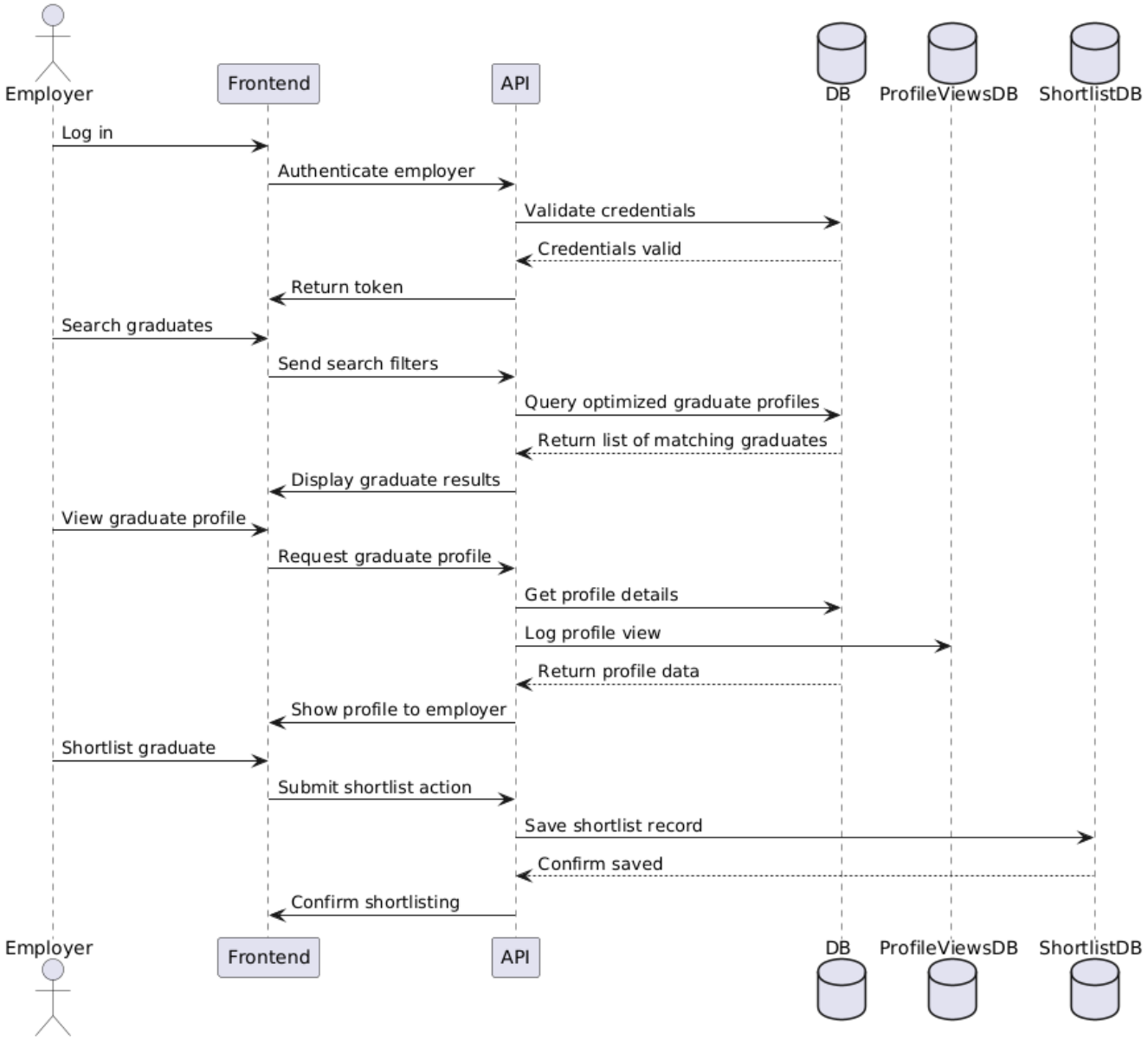
Sequence Diagram (Summary)

Flow:

- Employer submits request → Admin approves → Employer logs in → Searches graduates → Shortlists candidate → System notifies graduate → Employer marks outcome

Purpose: Demonstrates time-based interaction between system components and actors.


Graduate
Frontend
API
DB
ProfileViewsDB
ShortlistDB
Register and log in
Authenticate user
Validate credentials
Credentials valid
Return token
Upload certifications
Submit certification documents
Save certifications
Update profile
Submit profile data
Save updated profile
Optimize profile
Submit optimization data
Store optimization attributes
View shortlist history
Request shortlist history
Get shortlist records for graduate
Return shortlist data
Show shortlist history
View profile viewers
Request profile viewer list
Get viewer logs for graduate
Return viewer list
Display employers who viewed profile
Graduate
Frontend
API
DB
ProfileViewsDB
ShortlistDB


Employer
Frontend
API
DB
ProfileViewsDB
ShortlistDB
Log in
Authenticate employer
Validate credentials
Credentials valid
Return token
Search graduates
Send search filters
Query optimized graduate profiles
Return list of matching graduates
Display graduate results
View graduate profile
Request graduate profile
Get profile details
Log profile view
Return profile data
Show profile to employer
Shortlist graduate
Submit shortlist action
Save shortlist record
Confirm saved
Confirm shortlisting
Employer
Frontend
API
DB
ProfileViewsDB
ShortlistDB


Admin
Frontend
API
DB
Log in
Authenticate admin
Validate credentials
Return admin data
Return token
Dashboard Analytics
Access dashboard
Request analytics data
Fetch total graduates, suspended accounts, verified employers, etc.
Return analytics data
Display dashboard metrics
Graduate Management
Register new graduate
Submit graduate data
Save graduate account
Confirmation
Show success message
View all graduates
Request graduate list
Fetch all graduates
Return graduate data
Display graduate list
Suspend/Delete graduate
Submit suspension/deletion action
Update graduate account status
Confirmation
Show action status
Restore graduate account
Submit restore action
Reactivate graduate status
Confirmation
Show success message
Employer Management
View pending employers
Request pending employer list
Fetch unverified employers
Return list
Display employers
Approve or Reject employer
Submit decision
Update employer status
Confirmation
Show success message
Admin
Frontend
API
DB

4. Class Diagram

Purpose: Demonstrates system architecture and object responsibilities using three layers:

a) Presentation Layer

- `AdminUI`, `EmployerUI`, `GraduateUI`: Interfaces used to render dashboards, profile editors, and verification tools.

b) Business Layer

- Controllers (`AdminController`, `EmployerController`, `GraduateController`) encapsulate core logic.
- Services handle search logic, analytics, and notifications.

c) Persistence Layer

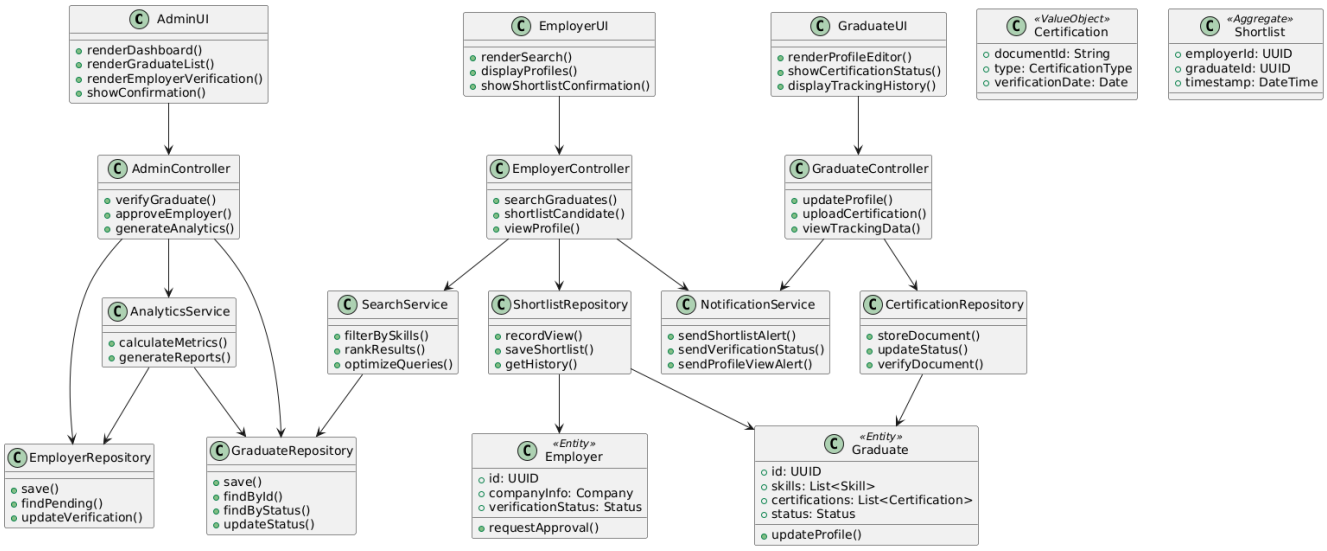
- Repositories manage saving, querying, and updating entities like graduates, employers, certifications, and shortlist actions.

d) Domain Models

- `Graduate`, `Employer`, `Shortlist`, and `Certification`: Represent core entities in the system.
- Encapsulate profile details, verification status, and shortlisting history.

Relationships:

- UI classes invoke Controllers
- Controllers use Repositories and Services
- Services access lower-layer repositories
- Entities are persisted via their associated repositories



Summary

The Eujim Graduate Engagement Platform is structured around strong domain-driven principles and provides seamless interaction between job seekers and employers. Key features like skill-based discovery, document uploads, shortlisting, and recruiter-driven hiring decisions are supported by a layered backend architecture.

Each UML diagram included serves a critical purpose:

- **Use Case Diagram:** Outlines what each user can do
- **Activity Diagrams:** Illustrate workflows and business rules
- **Class Diagram:** Maps the logical and technical architecture

By following these diagrams and flows, both developers and stakeholders can confidently understand and contribute to the system.