**📦 EPICS – Phase 6: Analytics, Admin & Scaling**

| **Epic ID** | **Epic Title** | **Description** | **Key Features** |
| --- | --- | --- | --- |
| EPIC-31 | Admin Dashboard Setup | Build tools for internal admin users to manage platform data, users, and flagged content. | User management, job post moderation, flag review system, admin-level analytics |
| EPIC-32 | Moderation & Reporting System | Enable users to report inappropriate content, and route it to admins for action. | Report buttons on posts/jobs, abuse categories, review workflows |
| EPIC-33 | Analytics & Usage Insights | Collect and display insights on user activity, popular features, and engagement metrics. | Pageviews, click-through rates, feature usage, dashboard graphs |
| EPIC-34 | Performance Optimization | Improve backend performance, reduce load times, and optimize queries and APIs. | Query optimization, CDN usage, lazy loading, database indexing |
| EPIC-35 | Infrastructure Scaling & DevOps | Prepare infrastructure for scaling globally and handling higher loads. | Auto-scaling, CI/CD pipelines, containerization (Docker), monitoring systems |

**✅ EPIC-31: Admin Dashboard Setup**

| **User Story ID** | **As a...** | **I want to...** | **So that...** | **Acceptance Criteria** |
| --- | --- | --- | --- | --- |
| US-31.1 | Admin | View a dashboard with key metrics (user count, job posts, reports) | I can monitor platform activity | Dashboard shows real-time stats with filter options |
| US-31.2 | Admin | View and edit user profiles | I can manage user accounts | Ability to search, suspend, and update user roles |
| US-31.3 | Admin | Review and manage job posts | I can control platform quality | Admins can approve, reject, or feature job posts |

**✅ EPIC-32: Moderation & Reporting System**

| **User Story ID** | **As a...** | **I want to...** | **So that...** | **Acceptance Criteria** |
| --- | --- | --- | --- | --- |
| US-32.1 | User | Report a job post or comment | I can flag inappropriate content | Report button is present with category selection |
| US-32.2 | Admin | View all reported content | I can review user-submitted flags | Filtered view by severity, type, and time |
| US-32.3 | Admin | Take action on reported content | I can remove or warn users | Approve/reject/warn options available per case |

**✅ EPIC-33: Analytics & Usage Insights**

| **User Story ID** | **As a...** | **I want to...** | **So that...** | **Acceptance Criteria** |
| --- | --- | --- | --- | --- |
| US-33.1 | Admin | View engagement stats on posts and jobs | I can understand user interests | Data shown as graphs, sortable by time range |
| US-33.2 | Admin | Track signup and retention trends | I can monitor growth metrics | Retention chart, daily active users, monthly active users |
| US-33.3 | Admin | Export analytics data | I can use it for reports and strategy | CSV/Excel export enabled for reports |

**✅ EPIC-34: Performance Optimization**

| **User Story ID** | **As a...** | **I want to...** | **So that...** | **Acceptance Criteria** |
| --- | --- | --- | --- | --- |
| US-34.1 | Developer | Optimize database queries | Page load times are faster | Query time < 300ms for key API calls |
| US-34.2 | User | Experience faster page loads | I can use the platform smoothly | Pages load in < 2 seconds |
| US-34.3 | Developer | Implement image compression and CDN | Assets load efficiently | Assets served via CDN, image size reduced by 60% |

**✅ EPIC-35: Infrastructure Scaling & DevOps**

| **User Story ID** | **As a...** | **I want to...** | **So that...** | **Acceptance Criteria** |
| --- | --- | --- | --- | --- |
| US-35.1 | DevOps Engineer | Set up CI/CD pipelines | I can deploy safely and quickly | Automatic build/test/deploy with rollback support |
| US-35.2 | DevOps Engineer | Containerize the app | It’s easier to scale and manage | App runs inside Docker containers |
| US-35.3 | DevOps Engineer | Implement monitoring & alerts | I can proactively handle outages | Alerts set for CPU, memory, and failed requests |

**Product Backlog – Phase 6**

| **PB ID** | **User Story** | **Priority** | **Story Points** | **Sprint** |
| --- | --- | --- | --- | --- |
| PB-31.1 | View a dashboard with key metrics (user count, job posts, reports) | High | 5 | Sprint 1 |
| PB-31.2 | View and edit user profiles | High | 3 | Sprint 1 |
| PB-31.3 | Review and manage job posts | Medium | 3 | Sprint 1 |

| PB-32.1 | Report a job post or comment | High | 3 | Sprint 2 | | PB-32.2 | View all reported content | Medium | 3 | Sprint 2 | | PB-32.3 | Take action on reported content | High | 5 | Sprint 2 |

| PB-33.1 | View engagement stats on posts and jobs | Medium | 3 | Sprint 3 | | PB-33.2 | Track signup and retention trends | High | 3 | Sprint 3 | | PB-33.3 | Export analytics data | Medium | 2 | Sprint 3 |

| PB-34.1 | Optimize database queries | High | 5 | Sprint 4 | | PB-34.2 | Experience faster page loads | High | 3 | Sprint 4 | | PB-34.3 | Implement image compression and CDN | Medium | 2 | Sprint 4 |

| PB-35.1 | Set up CI/CD pipelines | High | 5 | Sprint 5 | | PB-35.2 | Containerize the app | High | 5 | Sprint 5 | | PB-35.3 | Implement monitoring & alerts | High | 3 | Sprint 5 |

**🗓 Sprint Planning – Phase 6**

**🚀 Sprint 1: Admin Dashboard – Users & Jobs**

**Duration:** 2 weeks  
**Sprint Goal:** Build a foundational admin dashboard for managing users and jobs.

| **PB ID** | **Task** | **Owner** | **Story Points** | **Notes** |
| --- | --- | --- | --- | --- |
| PB-31.1 | Design and develop dashboard layout for key metrics | Frontend Dev | 5 | Reusable admin UI layout |
| PB-31.2 | Build user profile management UI | Full-stack | 3 | Includes edit, deactivate, ban |
| PB-31.3 | Create interface for job post management | Backend + Frontend | 3 | Requires pagination |

**🔒 Sprint 2: Moderation Tools**

**Duration:** 2 weeks  
**Sprint Goal:** Enable basic content moderation and reporting workflows.

| **PB ID** | **Task** | **Owner** | **Story Points** | **Notes** |
| --- | --- | --- | --- | --- |
| PB-32.1 | Implement "Report" functionality on jobs/comments | Frontend | 3 | Button + feedback form |
| PB-32.2 | Build moderation queue for reported content | Backend | 3 | Sort by severity/date |
| PB-32.3 | Add admin controls to resolve/report actions | Full-stack | 5 | Approve, delete, escalate |

**📊 Sprint 3: Analytics & Insights**

**Duration:** 2 weeks  
**Sprint Goal:** Provide visibility into user engagement and platform usage.

| **PB ID** | **Task** | **Owner** | **Story Points** | **Notes** |
| --- | --- | --- | --- | --- |
| PB-33.1 | Track job post and feed engagement metrics | Backend | 3 | View count, CTR |
| PB-33.2 | Add graphs for signups and active users | Frontend | 3 | Line/bar charts |
| PB-33.3 | Implement CSV export for analytics | Backend | 2 | Admin-only access |

**⚙️ Sprint 4: Performance Optimization**

**Duration:** 2 weeks  
**Sprint Goal:** Improve performance and scalability.

| **PB ID** | **Task** | **Owner** | **Story Points** | **Notes** |
| --- | --- | --- | --- | --- |
| PB-34.1 | Audit and optimize slow queries | Backend | 5 | Use indexing |
| PB-34.2 | Enable lazy loading and caching | Frontend | 3 | LCP < 2.5s |
| PB-34.3 | Integrate image optimization + CDN | DevOps | 2 | Compress and serve efficiently |

**🛠 Sprint 5: Infrastructure Scaling**

**Duration:** 2 weeks  
**Sprint Goal:** Set up deployment pipelines and scaling systems.

| **PB ID** | **Task** | **Owner** | **Story Points** | **Notes** |
| --- | --- | --- | --- | --- |
| PB-35.1 | Implement CI/CD with GitHub Actions | DevOps | 5 | Auto-deploy on merge |
| PB-35.2 | Containerize backend and frontend | DevOps | 5 | Docker & Docker Compose |
| PB-35.3 | Set up monitoring and alerts | DevOps | 3 | Use Grafana/Prometheus |