**Deployment Plan**

**Project**

Project Name

**Prepared for**

Client Name

**Prepared by**

[Your Name]

Day/Month/Year

**Purpose**

Describe the goal of this deployment.

**Deployment Scope**

|  |  |
| --- | --- |
| **Component** | **Description** |
| Backend API | FastAPI application with MySQL database |
| Frontend App | React-based admin dashboard |
| Database | MySQL 8.0, includes schema and seed data |
| Infrastructure | Deployed on Ubuntu 22.04 server (or AWS, etc.) |
| Services | Nginx, Gunicorn, Docker, etc. |

**Roles and Responsibilities**

|  |  |  |
| --- | --- | --- |
| **Name** | **Role** | **Responsibility** |
| Ahmed Al-Mohyedat | Project Manager | Oversee deployment, stakeholder communication |
| DevOps Engineer | Deployment Lead | Execute deployment steps |
| QA Engineer | Test Lead | Validate deployment success |
| IT Admin | Infrastructure Support | Server setup, firewall, access control |

**Deployment Schedule**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Task** | **Owner** | **Start Date** | **End Date** | **Status** |
| Final code freeze | Development Team | YYYY-MM-DD | YYYY-MM-DD |  |
| QA sign-off | QA Team | YYYY-MM-DD | YYYY-MM-DD |  |
| Backup existing environment | IT Admin | YYYY-MM-DD | YYYY-MM-DD |  |
| Database migration | DevOps | YYYY-MM-DD | YYYY-MM-DD |  |
| App & service deployment | DevOps | YYYY-MM-DD | YYYY-MM-DD |  |
| Smoke testing | QA Team | YYYY-MM-DD | YYYY-MM-DD |  |
| Go-live | All | YYYY-MM-DD | YYYY-MM-DD |  |

**Deployment Steps**

1. **Backup**
   * Snapshot existing production database and codebase
   * Store backups in secure offsite location
2. **Environment Preparation**
   * Confirm OS packages, Python/Node version
   * Pull latest image/tag from DockerHub or repository
   * Configure environment variables
3. **Database**
   * Apply schema changes using migration scripts
   * Seed required initial data (roles, default admin, etc.)
4. **Application Deployment**
   * Deploy backend services (via Docker, Gunicorn, etc.)
   * Deploy frontend (React/Vue build) to Nginx or S3
   * Set up reverse proxy via Nginx
5. **Testing**
   * Run smoke tests: authentication, API endpoints, UI load
   * Check error logs and server health
6. **Monitoring & Verification**
   * Enable uptime monitoring and logging (e.g., Prometheus, Grafana)
   * Review logs for any warnings/errors

**Rollback Plan**

|  |  |
| --- | --- |
| **Scenario** | **Rollback Action** |
| Deployment failure | Restore previous Docker image & DB backup |
| Database migration corrupts data | Restore DB from pre-deployment snapshot |
| Critical feature regression detected | Revert to last stable tag/version |

**Post-Deployment Verification**

|  |  |  |
| --- | --- | --- |
| **Area** | **Check** | **Status** |
| User Login | Confirm login functionality | ☐ Pass |
| API Connectivity | All key endpoints respond with expected data | ☐ Pass |
| Dashboard Rendering | Frontend loads correctly in all browsers | ☐ Pass |
| Email/SMS Alerts | Notifications are sent successfully | ☐ Pass |
| System Logs | No new errors/warnings logged | ☐ Pass |

**Dependencies and Risks**

|  |  |
| --- | --- |
| **Dependency** | **Mitigation Strategy** |
| Internet access for updates | Download offline packages if needed |
| DB server downtime | Schedule during off-peak hours |
| SSL certificate issues | Validate renewal before deployment |

**Approval**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Role** | **Signature** | **Date** |
| Project Manager |  |  |  |
| DevOps Lead |  |  |  |
| QA Lead |  |  |  |