Project Blue Light

MVP Specification

Minimum Viable Product Definition

1 Mid-Level Developer · 4 Weeks (22 Working Days)

Target: VER 1 — Without LEW Participant

• LEW/Drafter marketplace, bidding system not required

• Company's own LEW handles EMA submission offline

• Applicant self-service journey validated first

• Immediately monetizable via kVA-based pricing

Document Version: 1.0

Date: 2026-02-06

Classification: Internal Planning Document

Tech Stack: React 19 + Spring Boot 4.0.2 + MySQL 8.0

Table of Contents

A. MVP Scope (IN vs OUT)

IN — 12 Features

ID	Feature	MVP	Scope
A-1	Sign Up / Login	IN	Backend complete. **Build UI pages only**
A-2	New Licence Application	IN	Core flow: address → kVA selection → auto pricing → SLD upload → status tracking. Actual EMA submission handled offline by company LEW
A-4	Service Type Selection	IN	Single service type in MVP ("New Licence"). Dropdown structure preserved for future types (Renewal, Supply)
A-5	kVA-Based Auto Pricing	IN	MasterPrice entity ready. Key differentiator — instant transparent pricing
A-10	My Profile	IN	Edit name, phone. Change password
A-12	My Applications	IN	Application list + status tracking with colour-coded badges
A-14	Dashboard	IN	Summary cards (Total / Pending / In-Progress / Completed) + recent list
A-16	Document Upload/ Download	IN	SLD upload, completed docs download. Local disk storage (S3-ready interface)
A-17	Payment (simplified)	IN	Offline payment (PayNow/bank transfer) + admin manual confirmation. Real PG deferred
A-18	Payment History	IN	Per-application payment record list
C-1	User Management	IN	Admin: user list with search
C-3	Application Monitoring	IN	Admin: all applications list / status change / payment confirm / licence issuance

OUT — 23 Features (Deferred)

ID	Feature	MVP	Reason for Deferral
A-3	Licence Renewal	OUT	Nearly identical to A-2. Add `applicationType` enum later (2-3 days)
A-6	SLD Service Request	OUT	Requires Drafter marketplace → VER 2
A-7	Find & Engage LEW	OUT	Requires LEW marketplace → VER 2
A-8	Find SLD/Layout Drafter	OUT	Requires Drafter marketplace → VER 2
A-9	SLD Verification	OUT	Advanced feature → VER 2+
A-11	My Licenses	OUT	Manual licence registration is nice-to-have
A-13	My Engagement	OUT	No LEW/Drafter engagement in VER 1
A-15	Chat	OUT	WebSocket real-time feature → VER 2
B-1~B-8	LEW/Drafter Portal (all)	OUT	Entire portal deferred to VER 2
C-2	Professional Verification	OUT	No LEW/Drafter in VER 1
C-4	Support Handling	OUT	Use email for MVP support
D-1~D-5	System/Notifications (all)	OUT	AI chatbot, email alerts, reminders → post-MVP

B. User Stories

P1 **Priority 1 — Critical Path (Must Ship)**

US-1: Applicant Registration & Login A-1

- Sign up with email / password / name / phone → JWT-based login
- Protected routes redirect unauthenticated users to login
- Role-based route guards (APPLICANT vs ADMIN)

US-2: Create New Licence Application A-2, A-4, A-5

- Multi-step form: Step 1 (Address + Building Type) → Step 2 (kVA Selection + Auto Price) → Step 3 (Review + Confirm)
- kVA dropdown fetches active MasterPrice entries; price auto-calculated on selection
- On submit: Application created with `PENDING_PAYMENT` status; payment instructions displayed

US-3: Upload SLD Document A-16

- Accept PDF / JPG / PNG (max 10MB); store on local disk
- Create FileEntity with `DRAWING_SLD` type; allow file replacement

US-4: View My Applications List A-12

- Display: address, kVA, amount, status, created date
- Colour-coded status badges: PENDING_PAYMENT=yellow, PAID=blue, IN_PROGRESS=orange, COMPLETED=green

US-5: View Application Detail A-12, A-16, A-18

- Full fields + uploaded files (download links) + payment history
- Step tracker progress indicator; licence info when COMPLETED

US-6: Applicant Dashboard A-14

- Summary cards: Total / Pending Payment / In Progress / Completed counts
- Recent 5 applications list + "New Application" CTA button

P2 ## Priority 2 — Admin Operations (Must Ship)

US-7: Admin Application Monitoring C-3

- Paginated list of all applications + filter by status / search
- Status transitions: PENDING_PAYMENT → PAID → IN_PROGRESS → COMPLETED
- On COMPLETED: enter licence number + expiry date

US-8: Admin Payment Confirmation A-17 (admin)

- "Confirm Payment" on PENDING_PAYMENT application
- Creates Payment record (SUCCESS) + Application status → PAID

US-9: Admin User Management C-1

- User list: email, name, role, registration date + search

P3 ## Priority 3 — Enhancements (Should Ship)

US-10: Profile Management A-10

- Edit name / phone; change password (requires current password)

US-11: Download Completed Documents A-16

- Applicant downloads licence PDF uploaded by admin for COMPLETED applications

US-12: Admin File Upload A-16 (admin)

- Admin uploads LICENSE_PDF / REPORT_PDF for completed applications

C. Architecture

C-1. Frontend Routes

Route	Role	Page
`/login`	Public	LoginPage
`/signup`	Public	SignupPage
`/dashboard`	APPLICANT	Applicant Dashboard
`/applications`	APPLICANT	Application List
`/applications/new`	APPLICANT	New Application (multi-step form)
`/applications/:id`	APPLICANT	Application Detail
`/profile`	APPLICANT	Profile Management
`/admin/dashboard`	ADMIN	Admin Dashboard
`/admin/applications`	ADMIN	Admin Application List
`/admin/applications/:id`	ADMIN	Admin Application Detail + Status Mgmt
`/admin/users`	ADMIN	Admin User List

C-2. Frontend Directory Structure

```
src/
  api/
    axiosClient.ts          (existing)
    authApi.ts              (existing)
    applicationApi.ts       (new)
    priceApi.ts             (new)
    fileApi.ts              (new)
    paymentApi.ts           (new)
    userApi.ts              (new)
  stores/
    authStore.ts            (existing)
  types/
    index.ts                (existing — all interfaces defined)
  components/
    common/   Layout, ProtectedRoute, StatusBadge,
              FileUpload, StepTracker, DataTable,
              LoadingSpinner, EmptyState
    applicant/ DashboardCards, ApplicationCard,
              ApplicationForm/ (Step1, Step2, Step3)
    admin/    ApplicationStatusManager,
              PaymentConfirmation, UserTable
  pages/
    auth/     LoginPage, SignupPage
    applicant/ DashboardPage, ApplicationListPage,
              NewApplicationPage, ApplicationDetailPage,
              ProfilePage
    admin/    AdminDashboardPage, AdminApplicationListPage,
              AdminApplicationDetailPage, AdminUserListPage
  router/     index.tsx, routes.tsx
```

C-3. Backend API Endpoints

Method	Endpoint	Auth	Description
Auth (existing)			
POST	`/api/auth/signup`	Public	User registration
POST	`/api/auth/login`	Public	User login
Prices (new)			
GET	`/api/prices`	Public	List active kVA price tiers
GET	`/api/prices/calculate?kva=`	Public	Calculate price for given kVA
Applications - Applicant (new)			
POST	`/api/applications`	APPLICANT	Create new application
GET	`/api/applications`	APPLICANT	List my applications
GET	`/api/applications/:id`	APPLICANT	Application detail (+ files, payments)
GET	`/api/applications/summary`	APPLICANT	Dashboard summary counts
Applications - Admin (new)			
GET	`/api/admin/applications`	ADMIN	List all applications (paginated)
GET	`/api/admin/applications/:id`	ADMIN	Any application detail
PATCH	`/api/admin/applications/:id/status`	ADMIN	Update application status
POST	`/api/admin/applications/:id/complete`	ADMIN	Mark complete with licence info
Files (new)			
POST	`/api/applications/:id/files`	Authenticated	Upload file
GET	`/api/applications/:id/files`	Authenticated	List files
GET	`/api/files/:fileId/download`	Authenticated	Download file
DELETE	`/api/files/:fileId`	Authenticated	Delete file (soft)
Payments (new)			
GET	`/api/applications/:id/payments`	Authenticated	List payments for application
POST		ADMIN	Confirm offline payment

`/api/admin/applications/:id/payments/confirm`

Users (new)			
GET	`/api/users/me`	Authenticated	Current user profile
PUT	`/api/users/me`	Authenticated	Update profile
PUT	`/api/users/me/password`	Authenticated	Change password
GET	`/api/admin/users`	ADMIN	List all users

C-4. Database

> **No schema changes required.** All 6 existing entities (User, Application, Payment, Inspection, FileEntity, MasterPrice) are sufficient for MVP. Inspection entity is dormant until VER 2.

Required setup:

- `schema.sql` — CREATE TABLE scripts matching all JPA entities (JPA validate mode)
- `data.sql` — MasterPrice seed data (kVA pricing tiers) + Admin user account

D. Sprint Plan (4 Weeks, 22 Working Days)

Week 1: Foundation + Auth UI + Backend APIs (Day 1-5)

Day	Task	Deliverable
1	DB initialisation (schema.sql + data.sql), frontend router/layout shell setup	MySQL tables created, seed data loaded, React Router + Layout working
2	Auth UI: LoginPage, SignupPage, ProtectedRoute component	Full auth flow working E2E
3	Backend: PriceController + ApplicationService/Controller (create, list, detail)	Application CRUD APIs tested via Postman
4	Backend: Admin APIs (all applications, status update, payment confirmation)	Admin workflow APIs complete
5	Backend: FileStorageService + FileController (upload/download)	File upload/download working

Week 2: Applicant Frontend (Day 6-10)

Day	Task	Deliverable
6	Frontend API layer (applicationApi, priceApi, fileApi, paymentApi) + common components	API modules + StatusBadge, DataTable, etc.
7	Applicant Dashboard (summary cards + recent applications)	Dashboard page functional
8	New Application multi-step form (Address → kVA/Price → Review)	Application creation E2E
9	Application List + Detail pages	List with filters + Detail with step tracker
10	File upload/download integration + Applicant E2E testing	Complete applicant journey tested

Week 3: Admin Frontend + Profile + Integration (Day 11-15)

Day	Task	Deliverable
11	Admin Dashboard	Admin overview page
12	Admin Application Management (list / filter / detail)	Admin can browse all applications
13	Admin: Payment confirmation + licence completion + file upload	Full admin workflow functional
14	Admin User Management + Applicant Profile page	User list + Profile edit working
15	E2E integration testing (create → pay → status → complete → download)	All flows validated end-to-end

Week 4: Polish + Stabilisation (Day 16-22)

Day	Task	Deliverable
16-17	Responsive design (mobile optimisation, sidebar toggle)	Mobile-friendly UI
18	Error handling + empty states + session expiry	Robust error UX
19	Loading states + toast notifications + UX improvements	Polished interactions
20	Security review (role verification, data access scope, CORS)	Security-hardened
21	Documentation (README, API list, env variable guide)	Developer documentation
22	Final testing + buffer	Production-ready MVP

Critical Path: Backend Application API (Day 3) → Frontend Application Form (Day 8) → Admin Application Management (Day 12) → E2E Testing (Day 15)

E. Expansion Strategy (Deferred Features)

Feature	How to Expand	Estimated Effort
Licence Renewal (A-3)	Add `applicationType` enum (NEW / RENEWAL) to Application entity	2-3 days
LEW/Drafter Portal (B-all)	Add LEW / DRAFTER to UserRole enum. New Engagement entity. /lew/* and /drafter/* routes	3-4 weeks
Real Payment Gateway (A-17)	Stripe / PayNow webhook handler replaces admin manual confirmation	1 week
Inspection / PDF Report (B-3, B-4)	Existing Inspection entity + iText/Jasper PDF generation	1-2 weeks
Chat (A-15)	Message entity + WebSocket (STOMP)	2-3 weeks
Email Notifications (D-3, D-4)	Spring Mail + event listeners + email templates	3-5 days
S3 File Storage	FileStorageService interface → swap LocalStorage for S3 implementation	1 day
AI Chatbot (D-1)	/api/chat endpoint + AI API integration	1-2 weeks

> **Key Design Principles for Extensibility:**
>
> - **FileStorageService** designed as interface (Local → S3 swappable)
> - **UserRole** as enum (LEW / DRAFTER easily addable)
> - **Application** entity's flexible status machine is the foundation for all expansions

F. Risk Assessment

Risk	Likelihood	Impact	Mitigation
Spring Boot 4.0.2 compatibility	Medium	High	4.x is very new (Jakarta EE migration). If blockers arise, downgrade to 3.4.x
JPA validate + tables missing	High	Medium	**Create schema.sql immediately on Day 1**
Frontend workload underestimated	Medium	High	10+ pages in 2 weeks. Leverage Tailwind component library aggressively
Scope creep	High	High	This document is the agreed scope. All additions go to deferred backlog
kVA pricing data unavailable	Low	Medium	Use placeholder values; make configurable via admin
No automated tests	High	Medium	Write integration tests for 3 critical endpoints: create application, confirm payment, update status