Laundry App - Developer Implementation Plan
Prepared: 2025-09-28

Timeline Constraint: Complete delivery in 4 consecutive working days (high-compression plan).

Team Assignments:
Developer A - Backend Lead (Spring Boot, PostgreSQL)
Developer B - Frontend Lead (React, TypeScript)
Developer C - Full-Stack & QA (Tooling, Automation, Tests)

Objective: finish all outstanding features referenced in the repositoryâ placeholder pages, TODOs, empty modulesâ and production-harden the platform within four working days. Teams execute in parallel with twice-daily integration checkpoints to sustain speed and quality.

Scope: complete laundry slot overrides, rooftop booking history, dashboard metrics, presidium content, supporting backend services, mock data generation, automated pipelines, and observability updates.

Execution Timeline (4-Day Target):

- Day 1 â Kickoff & Foundation Build: Morning: joint architecture session, finalize API contracts, confirm database migrations. Afternoon: Developer A begins BE-01 overrides and seeds BE-02 structures; Developer B scaffolds dashboard and admin UIs against mocked data; Developer C prepares mock data schema (FS-01) and dev tooling (FS-04).
- Day 2 â Backend Completion & Frontend Wiring: Developer A wraps BE-01, delivers BE-02 rooftop history, and exposes initial BE-03 metrics contract. Developer B integrates FE-01 dashboard and FE-03 buffer fix using live endpoints, begins FE-02 override UI. Developer C implements mock data script (FS-01) and initiates CI scaffolding (FS-03).
- Day 3 â Feature Finalization & QA Prep: Developer A ships BE-03 dashboard metrics, BE-04 presidium backend, and starts BE-05 tests. Developer B completes FE-02 overrides UI, FE-04 rooftop history view, and FE-05 presidium page; begins FE-06 polish. Developer C builds e2e suite (FS-02) leveraging mock data and advances FS-03 pipeline & FS-04 tooling.
- Day 4 â Hardening, Observability & Sign-off: Morning regression using e2e suite, fix defects. Developer A finalizes BE-05 coverage; Developer B finishes FE-06 polish. Developer C completes CI/CD pipeline (FS-03) and observability enhancements (FS-05). Afternoon stakeholder demo, release checklist, documentation updates.

Parallelization Notes:

- Twice-daily syncs (09:00 kickoff, 16:00 integration review) to realign tasks.
- Feature branches merged before 18:00 to allow nightly automated runs.
- Pairing sessions planned for overrides (A+B) and QA automation (B+C) to remove blockers quickly.

Cross-Team Dependencies:

- BE-01 must precede FE-02 for admin slot management UI.
- BE-02 enables FE-04 rooftop history experience.
- BE-03 feeds FE-01 dashboard metrics.
- BE-04 supplies data to FE-05 presidium page.
- FS-01 mock data required before FS-02 e2e and Day-4 regression.

Task BE-01 (Developer A): Laundry Slot Override & Schedule Management API

Summary: Allow laundry/master admins to block, extend, or shrink slot availability without editing machine definitions directly.

Technical Notes:

- Create entity (LaundrySlotOverride) keyed by machine, date range, and slot indices with status (BLOCKED, EXTENDED).
- Expose CRUD endpoints under /api/laundry/overrides restricted to LAUNDRY_ADMIN+ with bulk operations.
- Update booking validation to honor overrides and treat blocked slots as unavailable.
- Persist audit metadata (createdBy, createdAt) for traceability.

Dependencies: None (foundational story kick-started Day 1).

Acceptance Criteria:

- Overrides mutate availability without service restart.
- Booking attempts against blocked slots return HTTP 400 with descriptive message.
- Integration tests cover bookings near overrides and weekly limits.

Deliverables:

- Updated JPA entities, repositories, controllers.
- Database migration or schema update notes.
- REST contract examples for frontend.

Task BE-02 (Developer A): Rooftop Booking History & Audit Endpoints

Summary: Expose user and admin APIs to list historical rooftop bookings/requests with status transitions.

Technical Notes:

- Add pageable GET /api/rooftop/bookings/me returning future and past entries filtered by status.
- Provide admin GET /api/rooftop/bookings with filters (date range, approver, status).
- Extend RooftopBookingRequest with status enum (REQUESTED, APPROVED, REJECTED, CANCELLED) plus decision metadata.
- Allow admins to reject requests with reason and residents to cancel pending requests within allowed window.

Dependencies: Runs after BE-01 migration skeleton confirmed (Day 1 afternoon).

Acceptance Criteria:

- API responses include pagination metadata consumed by FE-04.
- Security annotations enforce approval/rejection permissions.
- Tests cover approval, rejection, cancellation flows.

Deliverables:

- Controller updates with method-level security.
- Entity/DTO adjustments and migration notes.
- API contract update (Markdown/OpenAPI).

Task BE-03 (Developer A): Dashboard Metrics Aggregator

Summary: Provide aggregated metrics for the home dashboard (counts, upcoming events, quota usage).

Technical Notes:

- Implement /api/dashboard/summary returning next laundry slots, quota usage, pending requests, upcoming rooftop events.

- Introduce short-lived caching (e.g., Caffeine) to keep response time under 200ms.
- Support admin flag to return global stats when authorized.

Dependencies: Feeds FE-01 dashboard (target completion Day 2-3).

Acceptance Criteria:

- Endpoint returns deterministic schema with numeric counts and ISO timestamps.
- Performance benchmark <200ms with 1k bookings dataset.
- Role-based access verified via integration test.

Deliverables:

- New service aggregator with unit tests.
- API contract documented for frontend.

Task BE-04 (Developer A): Presidium Directory Backend

Summary: Manage presidium members (name, role, contact, portrait URL) with public read and admin CRUD.

Technical Notes:

- Create PresidiumMember entity with ordering and visibility flags.
- Expose GET /api/presidium (public) and admin CRUD endpoints for MASTER_ADMIN.
- Include optional bio/description field surfaced to frontend.

Dependencies: Coordinate payload shape with FE-05 prior to implementation.

Acceptance Criteria:

- CRUD operations enforce optimistic locking or updated timestamp checks.
- Validation rejects invalid URLs/missing required fields.
- Provide default seed entry for demo environments.

Deliverables:

- Entity + repository + controller code.
- Admin initializer update avoiding duplicates.

Task BE-05 (Developer A): Backend Test Coverage & Hardening

Summary: Add automated coverage for booking flows and new endpoints.

Technical Notes:

- Write Spring Boot slice tests for laundry and rooftop controllers.
- Add integration tests covering overrides, history filters, dashboard metrics.
- Introduce testcontainers-based Postgres profile for CI parity.

Dependencies: Requires BE-01..BE-04 complete (Day 3-4).

Acceptance Criteria:

- â¥80% line coverage on laundry and rooftop packages.
- Gradle tasks configured for unit + integration tests.
- Tests executed successfully in CI pipeline.

Deliverables:

- Test sources with data builders.
- Gradle configuration updates.

Task FE-01 (Developer B): Home Dashboard Experience

Summary: Replace placeholder HomePage with data-driven dashboard using BE-03 summary.

Technical Notes:

- Design responsive cards showing next laundry slot, quota progress, pending rooftop requests, quick actions.
- Implement loading/error states consistent with design system.

- Include announcements panel leveraging static copy (extendable later).

Dependencies: Depends on BE-03 contract (mock Day 1, live Day 2).

Acceptance Criteria:

- Authenticated users see live data; guests view marketing copy.
- Mobile layout stacks cards with accessible order.
- React Testing Library coverage for formatting and skeleton states.

Deliverables:

- Updated HomePage.tsx and supporting components.
- Screenshots/Storybook references for QA.

Task FE-02 (Developer B): Laundry Slot Management UI

Summary: Provide admin interface to manage overrides from BE-01.

Technical Notes:

- Add "Slot Overrides" window under AdminPage â Laundry.
- Implement table/calendar visualising overrides per machine/date.
- Wire create/update/delete flows with optimistic updates and toast feedback.

Dependencies: Blocked on BE-01 (target Day 2-3).

Acceptance Criteria:

- Admins can block slots and see immediate effect on schedule.
- Form validation prevents overlapping/invalid ranges.
- Jest tests cover API hook happy path and error handling.

Deliverables:

- React Query hooks under features/laundry/overrides.
- CSS modules consistent with design.

Task FE-03 (Developer B): Next Free Slots Buffer Fix

Summary: Resolve TODO in NextFreeSlots component for previous-day buffers.

Technical Notes:

- Adjust free-slot calculation to merge bookings returned with includeBuffer flag.
- Add utilities to normalise booking dates/slots across day boundaries.
- Display tooltip when slot partially occupied by previous-day usage.

Dependencies: Uses existing endpoint (complete by Day 2).

Acceptance Criteria:

- Adjacent buffered slots hidden or labelled unavailable.
- Unit tests cover midnight boundary conditions.
- Manual QA checklist updated for DST edge cases.

Deliverables:

- Updated NextFreeSlots.tsx and utilities.
- Tests under features/laundry/__tests__.

Task FE-04 (Developer B): Rooftop History & Request Timeline UI

Summary: Surface user rooftop booking/request history with filters.

Technical Notes:

- Replace placeholder tab in HistoryPage with timeline grouped by month/status.
- Enable cancellation of pending requests per BE-02 rules.
- Implement pagination/infinite scroll for older records.

Dependencies: Requires BE-02 (target Day 3).

Acceptance Criteria:

- History view paginates smoothly and supports lazy loading.
- Actions update UI state without full reload.
- Accessibility review ensures screen-reader friendly announcements.

Deliverables:

- New components/hooks under features/rooftop/history.
- Router/localization updates if needed.

Task FE-05 (Developer B): Presidium Page Content

Summary: Populate PresidiumPage with data from BE-04.

Technical Notes:

- Fetch presidium members and render cards (name, role, contact, office hours).
- Provide fallback imagery when portrait missing.
- Add admin-only edit link placeholder for future management UI.

Dependencies: Depends on BE-04 (Day 3).

Acceptance Criteria:

- Page passes Lighthouse accessibility score â¥90.
- Graceful empty state messaging when no members configured.

Deliverables:

- Updated PresidiumPage.tsx and styling.
- Unit tests for render and error states.

Task FE-06 (Developer B): UI Polish & Theme Consistency

Summary: Ensure new components honor theme toggle and responsive behaviour.

Technical Notes:

- Audit dark mode styles and adjust CSS variables.
- Add focus states and keyboard navigation to admin tables/calendars.
- Refine shared Button/Form components for new use cases.

Dependencies: Runs alongside finish work (Day 3-4).

Acceptance Criteria:

- No WCAG AA colour contrast violations.
- Keyboard-only interaction works across modals/forms.

Deliverables:

- CSS module updates and documentation screenshots.

Task FS-01 (Developer C): Mock Data Generator

Summary: Implement scripts/generate_mock_data.py to seed realistic demo/test data.

Technical Notes:

- Generate users, machines, overrides, bookings, rooftop requests across edge cases.
- Provide CLI flags for environment (dev/staging) and data volume.
- Ensure passwords hashed when inserting directly.

Dependencies: Needs schema confirmation from BE-01..BE-04 (start Day 1 with stubs).

Acceptance Criteria:

- Script populates empty database via JDBC/REST in under 2 minutes.
- Idempotent: supports reset flag to clean previous mock data.
- README updated with usage instructions.

Deliverables:

- Executable script plus dependency notes.
- Sample .env.dev with credentials.

Task FS-02 (Developer C): End-to-End Test Suite

Summary: Automate critical flows using Playwright or Cypress.

Technical Notes:

- Cover login/logout, booking creation/cancellation, override management, rooftop approvals.
- Launch backend/frontend via docker-compose for deterministic runs.
- Leverage FS-01 mock data for fixtures.

Dependencies: Depends on key features being ready (Day 3).

Acceptance Criteria:

- Headless CI execution under 10 minutes.
- Screenshots/videos captured for failures.
- Documented npm script (npm run e2e).

Deliverables:

- tests/e2e directory with config.
- CI job invoking suite (ties to FS-03).

Task FS-03 (Developer C): CI/CD Pipeline Hardening

Summary: Create workflows for linting, tests, builds, and artifact publication.

Technical Notes:

- Stages: backend unit+integration, frontend lint+build, e2e tests.
- Cache Gradle and npm dependencies to reduce runtime.
- Publish build artifacts (frontend dist, backend jar) as workflow outputs.

Dependencies: Requires BE-05 tests + FS-02 e2e (Day 4).

Acceptance Criteria:

- Pipelines pass on main/feature branches with status badges.
- Failing tests block merges via required checks.
- Documentation for CI secrets/environment variables.

Deliverables:

- .github/workflows YAML files.
- README section describing pipeline/troubleshooting.

Task FS-04 (Developer C): Unified Local Development Tooling

Summary: Simplify onboarding and environment parity.

Technical Notes:

- Extend compose.yml to include frontend dev container or scripts for simultaneous startup.
- Create makefile or npm script orchestrating docker compose, backend bootRun, frontend dev server.
- Document SSL/CORS troubleshooting for cross-host access.

Dependencies: Parallel track across days 1-3.

Acceptance Criteria:

- One-command startup for full stack.
- Works on macOS/Linux/Windows WSL with prerequisites documented.

Deliverables:

- Updated compose files or new root-level compose.
- DEV_SETUP.md instructions.

Task FS-05 (Developer C): Observability & Logging Enhancements

Summary: Improve insight into booking operations.

Technical Notes:
  - Configure Spring Boot actuator endpoints with appropriate security.
  - Add structured logging (JSON) for booking lifecycle events.
  - Provide placeholder Prometheus/Grafana configuration for future deployment.

Dependencies: Finalize Day 4 after features stable.

Acceptance Criteria:
  - Logs include booking ID, user, machine, outcome, latency.
  - README documents metrics enablement.
  - Health endpoint accessible only with authentication.

Deliverables:
  - application.properties updates and logging config.
  - docs/observability.md.

Quality & Release Checklist (Developer C ownership, all sign-off):
- Security review for new endpoints (authentication, authorization, CSRF implications).
- Database migration tested against existing data snapshot.
- Performance smoke test: booking creation under 50 concurrent users.
- Rollback plan documented for overrides feature.
- User documentation updated (README plus in-app help copy if applicable).

Communication Cadence:
- Daily 15-minute stand-up at 09:00 and integration review at 16:00 focused on task IDs.
- Dedicated Slack/Teams channel for sub-30-minute blocker resolution.
- Day-4 stakeholder demo using mock data set to validate acceptance criteria.
- Use shared tracking board (e.g., Jira) keyed by task IDs above.

Risks & Mitigations:
- Schema changes overlapping tasks - Mitigation: feature-flag overrides and supply fallback queries for legacy clients.
- Limited test data diversity - Mitigation: extend FS-01 script with edge-case scenarios (overlapping slots, quota near limits).
- Frontend performance regression - Mitigation: enforce bundle analyzer check during FE-06 and defer heavy modules to lazy loading.

Sign-off Requirements:
- All tasks BE-01..FS-05 merged with approvals and QA evidence attached.
- Automated pipelines green on main branch for final 24 hours.
- Product owner sign-off on dashboard, slot management, rooftop history demos.