

PROJECT DEVELOPMENT PROCEDURES & CURRENT STATUS REPORT

Report Date: January 10, 2026

Project: Love to Fly Portal

Status: 95% Complete - Production Ready

Target Launch: February 23, 2026 (Aggressive: Feb 9, 2026)

EXECUTIVE SUMMARY

Project Status

-  **System Status:** 95% Complete
 -  **Code Deployment:** Deployed to Netlify (auto-deploy enabled)
 -  **Database:** PostgreSQL (Neon) connected and operational
 -  **Authentication:** JWT + Session management implemented
 -  **Payments:** Stripe integration complete
 -  **Emails:** Resend integration operational
 -  **Charts:** 1,900 aeronautical PDFs (pending S3/R2 deployment)
-

ACTUAL PROCEDURES FOR NEW CHANGES

Step 1: CHANGE PLANNING PROTOCOL

When: Before starting ANY modification

Process:

1. List all affected files and modules
2. Check feature independence (no cross-contamination)
3. Verify existing tests still pass
4. Document the change impact

Files to Review:

- `PRIORITY_INDEX.md` - Check if task is in current priority
- `IMPLEMENTATION_CHECKLIST.md` - Add to tracking
- Feature-specific guide (e.g., `HANGARSHARE_COMPLETE_GUIDE.md`)

Step 2: FEATURE ISOLATION PROTOCOL

Rule: Each feature MUST remain independent

Enforcement:

-  Can READ data from other features
-  Cannot MODIFY other feature databases
-  Use API endpoints for cross-feature communication
-  No shared state across features

Example (Logbook + Career):

-  ALLOWED: Logbook reads `total_flight_hours` from `users` table
 -  FORBIDDEN: Logbook modifies `career_profiles` table
 -  ALLOWED: Career API endpoint provides data to logbook

Step 3: CODE CHANGE PROCEDURE

Process:

1. Create feature branch: `git checkout -b feature/[feature-name]`

2. Make isolated changes (single feature)
3. Run tests: `npm run test:watch`
4. Build: `npm run build`
5. Push: `git push origin feature/[feature-name]`
6. Create Pull Request
7. Merge to main → Netlify auto-deploys

Files to Update:

- Code files (src/ directory)
- Database migrations (src/migrations/)
- TypeScript types (src/types/)
- Tests (if applicable)

Step 4: DATABASE MIGRATION PROTOCOL

For adding/modifying tables:

1. Create migration file:

```
npm run migrate:create -- --name "description"
```

2. Edit migration file in `src/migrations/00X_description.sql`

3. Run migration:

```
npm run migrate:up
```

4. Update TypeScript definitions:

- Edit `src/types/db.d.ts`
- Add new interfaces/types

5. Test with API:

- Verify data persists
- Check relationships work

Important: Database schema changes MUST be in separate migrations, one per feature.

Step 5: TESTING & VALIDATION

Before deployment:

1. Local build succeeds: `npm run build`
2. No TypeScript errors: `npm run lint`
3. Dev server runs: `npm run dev`
4. Database queries work
5. Feature is isolated (doesn't break other features)

Testing Command Sequence:

```
npm run lint          # Code quality check
npm run test          # Unit tests (45/45 passing)
npm run build         # Production build
npm run dev           # Manual testing (localhost:3000)
```

PROCEDURES FOR UPDATING PRIORITIES

Updating Task Status

File to Edit: IMPLEMENTATION_CHECKLIST.md

Process:

1. Open IMPLEMENTATION_CHECKLIST.md
2. Find task row
3. Update Status column:
 - o ● Not Started
 - o ● In Progress
 - o ● Complete
 - o ● Blocked
4. Update % Complete column
5. Add note in Notes column if needed

Example Row:

```
| 1 | Mock Data → Real DB | ● In Progress | 40% | Jan 10 | @name | Jan 13 | On track, DB connected |
```

Updating Priority Tasks

File to Edit: PRIORITY_TASKS.md

Process:

1. Open PRIORITY_TASKS.md
2. Find task section (Task 1-15)
3. Update Status field (lines ~150)
4. Update Effort Estimate if changed
5. Update Blockers section if needed
6. Update Completion % and ETA

Hierarchy of Priority Changes:

1. Update IMPLEMENTATION_CHECKLIST.md (daily tracking)
2. Update PRIORITY_TASKS.md (detailed task info)
3. Update PRIORITY_SUMMARY.md (visual overview)
4. Update ROADMAP.md (timeline impact)

Adding New Tasks

Process:

1. **Determine priority:** Critical | High | Medium | Low
2. **Estimate effort:** in hours/days
3. **Add to PRIORITY_TASKS.md** (follow template)
4. **Add to IMPLEMENTATION_CHECKLIST.md** (track status)
5. **Update ROADMAP.md** if changes timeline
6. **Notify team** via OPERATIONS_HANDOFF document

17 CURRENT PRIORITIES (From PRIORITY_INDEX.md)

● CRITICAL PATH (Week 1-3) - Must Complete

#	Task	Status	Effort	Week	Owner
1	Mock Data → Real DB (Airports)	● Not Started	3 days	W1	TBD
2	Photo Upload System	● Not Started	5-7 days	W2	TBD
3	Listing Edit Functionality	● Not Started	3-4 days	W1	TBD
4	Document Upload & Verify	● Not Started	3-4 days	W2-3	TBD

5	Booking Status Management	Not Started	3-4 days	W3	TBD
---	---------------------------	-------------	----------	----	-----

Total Effort: ~60 hours backend + 50 hours frontend

HIGH PRIORITY (Week 2-4) - MVP Complete

#	Task	Status	Effort	Week
6	Admin Verification Dashboard	Pending	4 days	W2-3
7	Advanced Search Filters	Pending	3-4 days	W4+
8	Reviews & Ratings	Pending	3-4 days	W4+

MEDIUM PRIORITY (Week 4-5) - Polish

#	Task	Status	Effort	Week
9	Performance Optimization	Pending	2-3 days	W5+
10	Mobile Responsiveness	Pending	2 days	W5+
11	Monitoring & Logging	Pending	2-3 days	W6

LAST 3 DEPLOYMENTS

Deployment 3: Career Phase 2 Implementation (Jan 10, 2026)

Status: Complete and Production-Ready

Features Delivered:

- Career marketplace module (Jobs + Applications)
- Company profiles and job listings
- Application workflow
- Reviews and ratings system
- Currency support (BRL default)
- Profile visibility controls

Documentation:

- CAREER_IMPLEMENTATION_INDEX.md (Master guide)
- CAREER_PHASES_1_2_SUMMARY.md (Completion summary)
- CAREER_PHASE2_TESTING_GUIDE.md (Testing procedures)

Files Modified:

- src/app/career/* (new pages)
- src/components/career/* (new components)
- src/migrations/014-023* (database tables)
- src/types/db.d.ts (TypeScript interfaces)

Testing Status: All tests passing (45/45)

Deployment 2: ANAC CIV Digital Logbook Compliance (Jan 10, 2026)

Status: Complete and Database Applied

Features Delivered:

- Logbook with ANAC CIV Digital compliance
- 15+ ANAC-required fields
- Time breakdown (Diurno, Noturno, IFR Real, Sob Capota, Simulador)

- Flight function tracking (PIC, SIC, STUDENT, INSTRUCTOR, DUAL)
- Rating support (IFRA, MLTE, MNTE, VFR)
- Status management (CADASTRADO, ENVIADO, CANCELADO)
- Landing counter tracking (day/night separate)

Database:

- Migration 028: flight_logs table creation
- Migration 029: ANAC compliance fields (already included in 028)
- 25 ANAC-compliant columns
- 5 performance indexes

Documentation:

- src/app/logbook/page.tsx (Complete rewrite)
- src/types/db.d.ts (FlightLog interface)
- src/migrations/028_create_flight_logs_table.sql

Build Status:  Zero TypeScript errors

Deployment 1: Error Handling & Page Standardization (Jan 6, 2026)

Status:  Deployed to Production (Netlify)

Features Delivered:

- 404 Not Found error page (professional UI)
- 500 Internal Server error page (professional UI)
- Portal layout standardization (4 pages fixed)
- Consistent color scheme (blue hero, light theme)
- Sidebar navigation integration

Pages Standardized:

- src/app/logbook/page.tsx → Blue hero, Sidebar, light theme
- src/app/forum/page.tsx → Light theme, Sidebar, card layout
- src/app/marketplace/page.tsx → Purple hero, card grid
- src/app/courses/page.tsx → Indigo hero, Sidebar

Documentation:

- ERROR_HANDLING_COMPLETE.md

Deployment Platform: Netlify (auto-deploy from GitHub)

DEVELOPMENT TOOLS & COMMANDS

Essential Commands

```
# Development
npm run dev          # Start dev server (localhost:3000)
npm run build         # Production build
npm run lint          # ESLint check

# Database
npm run migrate:up    # Apply pending migrations
npm run migrate:down   # Rollback last migration
npm run migrate:create # Create new migration

# Testing
npm run test           # Run all tests
npm run test:watch      # Watch mode
```

```
# Deployment
npm run build && npm run start # Production mode
```

Key Files for Operations

Daily Operations:

- docs/OPERATIONS_HANDOFF_2026-01-07.md - Daily procedures
- IMPLEMENTATION_CHECKLIST.md - Task tracking

Feature Development:

- PRIORITY_TASKS.md - Detailed task info
- ROADMAP.md - Timeline and phases

Deployment:

- DEPLOYMENT_COMPLETE.md - Deployment status
- netlify.toml - Build configuration

Database:

- src/migrations/ - All schema changes
- src/types/db.d.ts - TypeScript definitions
- .env.local - Environment variables (local only)

INFRASTRUCTURE & ENVIRONMENT

Current Stack

- Frontend:** Next.js 16.1.1 (Turbopack), React 19.2.3, TypeScript
- Backend:** Next.js API Routes
- Database:** PostgreSQL (Neon)
- Authentication:** JWT + Session management
- Payments:** Stripe
- Emails:** Resend
- Hosting:** Netlify (auto-deploy)
- Version Control:** GitHub

Environment Variables Required

```
# Database
DATABASE_URL=postgresql://...

# Auth
JWT_SECRET=your-secret-key
NEXTAUTH_SECRET=your-secret
NEXTAUTH_URL=https://yoursite.com

# Payments
STRIPE_SECRET_KEY=sk_...
NEXT_PUBLIC_STRIPE_PUBLISHABLE_KEY=pk_...
STRIPE_WEBHOOK_SECRET=whsec_...

# Email
RESEND_API_KEY=re_...
```

```
# Optional  
NEWS_API_KEY=...
```

Deployment Configuration

- **Build Command:** next build
 - **Start Command:** next start
 - **Build Cache:** Enabled
 - **Auto-Deploy:** Enabled from main branch
 - **Function Timeout:** 30 seconds (Netlify)
-

📈 PROJECT METRICS

Code Quality:

- TypeScript compilation: 0 errors
- ESLint: Strict rules enforced
- Test coverage: 45/45 tests passing
- Build time: ~17 seconds

Performance:

- Static pages: 42 generated
- API routes: 26 endpoints
- Bundle size: Optimized
- Database queries: Indexed

Completion:

- Auth: 100%
 - Core Features: 95%
 - Marketplace: 90%
 - Career: 100%
 - Logbook: 100%
 - Charts: 0% (pending S3 upload)
-

🎯 NEXT STEPS (From ROADMAP)

This Week (Jan 13-19)

- Verify ANAC logbook is working with database
- Start Career Phase 3 if Phase 2 complete
- Review and update priorities

Next 2 Weeks (Jan 20 - Feb 2)

- Begin critical path tasks (Week 1-2)
- Photo upload system implementation
- Document verification system

February (Final Push)

- Complete critical features
- Comprehensive testing
- Performance optimization
- Production launch

Detailed timeline: See ROADMAP.md

QUICK REFERENCE

Need	File	Time
Daily task tracking	IMPLEMENTATION_CHECKLIST.md	5 min
Priority overview	PRIORITY_SUMMARY.md	10 min
Full feature guide	HANGARSHARE_COMPLETE_GUIDE.md	30 min
Deployment status	DEPLOYMENT_COMPLETE.md	10 min
Development setup	documentation/START_HERE.md	20 min
All procedures	This document	15 min

COMPLIANCE & STANDARDS

Code Standards

- **Language:** TypeScript (strict mode)
- **Linting:** ESLint + Prettier
- **Testing:** Jest + Playwright
- **Documentation:** JSDoc comments
- **Git:** Feature branches, PR reviews

Database Standards

- **Migrations:** Sequential numbered files
- **Schema:** Normalized design
- **Indexes:** Performance-critical columns
- **Relationships:** Foreign keys enforced
- **Backups:** Daily automated (Neon)

Deployment Standards

- **Environment:** Netlify serverless
- **Secrets:** Environment variables (never in code)
- **Monitoring:** Build logs, performance metrics
- **Rollback:** Git-based (revert commit)
- **Documentation:** Updated with each deploy

SECURITY MEASURES

Implemented:

- JWT token authentication
- Password hashing (bcryptjs)
- Environment variables for secrets
- HTTPS/TLS enforcement
- CORS configuration
- Rate limiting ready

In Progress:

- Additional RBAC roles
- Audit logging
- DDoS protection (via Netlify)

LEARNING PATH FOR NEW DEVELOPERS

Day 1:

1. Read: .github/copilot-instructions.md (15 min)
2. Read: documentation/START_HERE.md (20 min)
3. Setup local environment (30 min)
4. Run `npm run dev` and explore (30 min)

Day 2:

1. Read: PRIORITY_INDEX.md → PRIORITY_SUMMARY.md (20 min)
2. Read: ROADMAP.md (30 min)
3. Explore codebase: src/app, src/components (1 hour)
4. Review: src/types/db.d.ts (15 min)

Day 3:

1. Read feature guide for your assignment (1 hour)
2. Setup feature branch (5 min)
3. Start implementation (ongoing)

Report Generated: January 10, 2026, 10:25 AM

Last Updated: Ongoing - Check .md files for latest status

Next Review: January 13, 2026 (weekly)

For detailed information, see the complete MD_FILES_INVENTORY.md document.