

Feature #6: Implementation Plan Summary

Overview

Complete planning and architecture for Feature #6: Smart Document Processing & Compliance

Files Created (19 total)

Documentation (4 files)

1. `docs/features/FEATURE_6_DOCUMENT_PROCESSING.md` - Complete feature specification
2. `docs/features/DOCUMENT_API_SPEC.md` - API endpoint documentation
3. `docs/features/FEATURE_6_CHECKLIST.md` - Week-by-week implementation checklist
4. `docs/features/FEATURE_6_READY.md` - Implementation readiness summary

Database (2 files)

1. `prisma/schema.prisma` - Updated with Document models and enums
2. `prisma/migrations/draft_add_document_processing/migration.sql` - Draft SQL migration

Code Structure (7 files)

1. `src/types/documents/index.ts` - TypeScript types and interfaces
2. `src/lib/documents/cloudinary.ts` - File upload/delete utilities
3. `src/lib/documents/ocr.ts` - OCR text extraction
4. `src/lib/documents/extraction.ts` - AI-powered field extraction
5. `src/lib/documents/classification.ts` - Document classification
6. `src/lib/documents/compliance.ts` - Compliance checking
7. `src/lib/documents/generation.ts` - PDF generation

Permissions (1 file)

1. `src/lib/permissions.ts` - Updated with document permissions

Summary Files (5 files)

1. `FEATURE_6_PLANNING_COMPLETE.md` - Executive summary
2. `IMPLEMENTATION_SUMMARY.md` - This file

Key Changes

Database Schema

- Added `Document` model with OCR and compliance tracking
- Added `DocumentTemplate` model for PDF generation
- Added 3 new enums: `DocumentType`, `ExtractionStatus`, `ComplianceStatus`
- Added relations to `User`, `Resident`, and `Inquiry` models

Permissions

Added 9 new document permissions:

- `DOCUMENTS_VIEW`

- DOCUMENTS_CREATE
- DOCUMENTS_UPDATE
- DOCUMENTS_DELETE
- DOCUMENTS_VIEW_ALL
- DOCUMENTS_EXTRACT
- DOCUMENTS_CLASSIFY
- DOCUMENTS_MANAGE_TEMPLATES
- DOCUMENTS_GENERATE

Role mappings:

- ADMIN: Full access
- OPERATOR: Full management capabilities
- CAREGIVER: View and create
- FAMILY: View and upload for their resident

Directory Structure Created

```
src/  
├── app/api/documents/  
│   ├── upload/  
│   ├── search/  
│   ├── compliance/  
│   ├── templates/  
│   └── generate/  
├── components/documents/  
├── lib/documents/  
└── types/documents/
```

Next Steps

1. Install dependencies:

```
bash
```

```
npm install tesseract.js pdf-lib react-pdf @types/pdf-lib cloudinary
```

2. Run database migration:

```
bash
```

```
npx prisma migrate dev --name add_document_processing
```

```
npx prisma generate
```

3. Start Phase 1A implementation (see FEATURE_6_CHECKLIST.md)

Timeline

- Week 1: Foundation (Database & Upload UI)
- Week 2: Extraction (OCR & AI)
- Week 3: Classification & Compliance
- Week 4: Generation & Polish

Total Duration: 4 weeks (Dec 19, 2025 - Jan 16, 2026)

Success Metrics

- Upload success rate > 99%

- OCR accuracy > 95%
 - Field extraction accuracy > 90%
 - 5-10 hours/week saved per operator
 - 80% reduction in data entry errors
-

Status:  Planning Complete

Date: December 19, 2025