Banking Fraud Detection System - Complete Folder Structure

```
fraud-detection-system/
│
├── README.md
├── docker-compose.yml
├── .env.example
├── .gitignore
├── Makefile
│
├── backend/                    # Go Backend Services
│   ├── Dockerfile
│   ├── go.mod
│   ├── go.sum
│   ├── cmd/
│   │   ├── api-gateway/
│   │   │   └── main.go
│   │   ├── auth-service/
│   │   │   └── main.go
│   │   ├── verification-service/
│   │   │   └── main.go
│   │   └── worker/
│   │       └── main.go
│   │
│   ├── internal/
│   │   ├── api/
│   │   │   ├── handlers/
│   │   │   │   ├── auth.go
│   │   │   │   ├── verification.go
│   │   │   │   ├── report.go
│   │   │   │   └── health.go
│   │   │   ├── middleware/
│   │   │   │   ├── auth.go
│   │   │   │   ├── ratelimit.go
│   │   │   │   ├── cors.go
│   │   │   │   └── logging.go
│   │   │   └── routes/
│   │   │       └── router.go
│   │   │
│   │   ├── models/
│   │   ├──── models/
│   │   │   ├──── user.go
│   │   │   ├──── message.go
│   │   │   ├──── verification.go
│   │   │   ├──── report.go
│   │   │   └──── rbi_circular.go
│   │   │
│   │   ├──── repository/
│   │   │   ├──── user_repo.go
│   │   │   ├──── message_repo.go
│   │   │   ├──── verification_repo.go
│   │   │   ├──── report_repo.go
│   │   │   └──── rbi_repo.go
│   │   │
│   │   ├──── service/
│   │   │   ├──── auth_service.go
│   │   │   ├──── verification_service.go
│   │   │   ├──── ml_client.go
│   │   │   ├──── rbi_compliance.go
│   │   │   ├──── header_verification.go
│   │   │   └──── alert_service.go
│   │   │
│   │   ├──── queue/
│   │   │   ├──── producer.go
│   │   │   ├──── consumer.go
│   │   │   └──── message.go
│   │   │
│   │   ├──── cache/
│   │   │   └──── redis.go
│   │   │
│   │   ├──── database/
│   │   │   ├──── postgres.go
│   │   │   └──── migrations/
│   │   │       ├──── 001_create_users.sql
│   │   │       ├──── 002_create_messages.sql
│   │   │       ├──── 003_create_verifications.sql
│   │   │       ├──── 004_create_reports.sql
│   │   │       ├──── 005_create_rbi_circulars.sql
│   │   │       └──── 006_create_sender_registry.sql
│   │   │
│   │   ├──── config/
│   │   │   └──── config.go
│   │   │
│   │   └──── utils/
│   │     ├──── jwt.go
│   │     ├──── validation.go
│   │     ├──── logger.go
│   │       ├── logger.go
│   │       └── errors.go
│   │
│   └── tests/
│       ├── integration/
│       ├── unit/
│       └── mocks/
│
├── ml-service/                 # Python ML Services
│   ├── Dockerfile
│   ├── requirements.txt
│   ├── setup.py
│   ├── pyproject.toml
│   │
│   ├── app/
│   │   ├── __init__.py
│   │   ├── main.py             # FastAPI application
│   │   ├── config.py
│   │   │
│   │   ├── api/
│   │   │   ├── __init__.py
│   │   │   ├── routes/
│   │   │   │   ├── __init__.py
│   │   │   │   ├── inference.py
│   │   │   │   ├── health.py
│   │   │   │   └── feedback.py
│   │   │   └── dependencies.py
│   │   │
│   │   ├── models/
│   │   │   ├── __init__.py
│   │   │   ├── ensemble.py
│   │   │   ├── distilbert_model.py
│   │   │   ├── roberta_model.py
│   │   │   ├── lstm_attention.py
│   │   │   ├── xgboost_model.py
│   │   │   └── model_loader.py
│   │   │
│   │   ├── preprocessing/
│   │   │   ├── __init__.py
│   │   │   ├── text_preprocessor.py
│   │   │   ├── feature_extractor.py
│   │   │   └── tokenizer.py
│   │   │
│   │   ├── inference/
│   │   │   ├── __init__.py
│   │   │   ├── predictor.py
│   │   │   ├── ensemble_predictor.py
│   │   │   ├── ensemble_predictor.py
│   │   │   └── explanation.py
│   │   │
│   │   ├── utils/
│   │   │   ├── __init__.py
│   │   │   ├── logger.py
│   │   │   ├── metrics.py
│   │   │   └── cache.py
│   │   │
│   │   └── schemas/
│   │       ├── __init__.py
│   │       ├── inference.py
│   │       └── response.py
│   │
│   ├── training/                # Training Pipeline
│   │   ├── __init__.py
│   │   ├── train_distilbert.py
│   │   ├── train_roberta.py
│   │   ├── train_lstm.py
│   │   ├── train_xgboost.py
│   │   ├── train_ensemble.py
│   │   │
│   │   ├── data/
│   │   │   ├── __init__.py
│   │   │   ├── dataset.py
│   │   │   ├── data_loader.py
│   │   │   ├── augmentation.py
│   │   │   └── preprocessing.py
│   │   │
│   │   ├── trainers/
│   │   │   ├── __init__.py
│   │   │   ├── base_trainer.py
│   │   │   ├── transformer_trainer.py
│   │   │   └── sklearn_trainer.py
│   │   │
│   │   ├── evaluation/
│   │   │   ├── __init__.py
│   │   │   ├── metrics.py
│   │   │   ├── evaluator.py
│   │   │   └── reports.py
│   │   │
│   │   └── config/
│   │       ├── distilbert_config.yaml
│   │       ├── roberta_config.yaml
│   │       ├── lstm_config.yaml
│   │       ├── xgboost_config.yaml
│   │       └── ensemble_config.yaml
│   │       └── ensemble_config.yaml
│   │
│   ├── notebooks/              # Jupyter Notebooks
│   │   ├── 01_data_exploration.ipynb
│   │   ├── 02_model_development.ipynb
│   │   ├── 03_model_evaluation.ipynb
│   │   └── 04_error_analysis.ipynb
│   │
│   ├── scripts/
│   │   ├── prepare_data.py
│   │   ├── train_all_models.py
│   │   ├── evaluate_models.py
│   │   ├── export_models.py
│   │   └── update_huggingface.py
│   │
│   ├── tests/
│   │   ├── test_models.py
│   │   ├── test_preprocessing.py
│   │   ├── test_inference.py
│   │   └── test_api.py
│   │
│   └── data/                 # Training Data
│      ├── raw/
│      │   ├── fraud_messages.csv
│      │   ├── legitimate_messages.csv
│      │   └── rbi_circulars.json
│      ├── processed/
│      │   ├── train.csv
│      │   ├── val.csv
│      │   └── test.csv
│      └── external/
│         └── sender_registry.json
│
├── models/                   # Saved Models
│   ├── distilbert/
│   │   ├── config.json
│   │   ├── pytorch_model.bin
│   │   ├── tokenizer_config.json
│   │   ├── vocab.txt
│   │   └── training_args.bin
│   │
│   ├── roberta/
│   │   ├── config.json
│   │   ├── pytorch_model.bin
│   │   └── tokenizer_config.json
│   │
│   ├── lstm/
│   ├── lstm/
│   │   ├── model.h5
│   │   ├── config.json
│   │   └── tokenizer.pkl
│   │
│   ├── xgboost/
│   │   ├── model.pkl
│   │   └── feature_names.json
│   │
│   └── ensemble/
│       ├── meta_model.pkl
│       ├── weights.json
│       └── config.json
│
├── frontend/                 # TypeScript/React Frontend
│   ├── Dockerfile
│   ├── package.json
│   ├── tsconfig.json
│   ├── vite.config.ts
│   ├── .env.example
│   │
│   ├── public/
│   │   ├── index.html
│   │   └── assets/
│   │
│   ├── src/
│   │   ├── main.tsx
│   │   ├── App.tsx
│   │   ├── vite-env.d.ts
│   │   │
│   │   ├── components/
│   │   │   ├── Layout/
│   │   │   │   ├── Header.tsx
│   │   │   │   ├── Sidebar.tsx
│   │   │   │   └── Footer.tsx
│   │   │   │
│   │   │   ├── Verification/
│   │   │   │   ├── MessageInput.tsx
│   │   │   │   ├── VerificationResult.tsx
│   │   │   │   ├── FraudScore.tsx
│   │   │   │   └── Explanation.tsx
│   │   │   │
│   │   │   ├── Dashboard/
│   │   │   │   ├── StatsCard.tsx
│   │   │   │   ├── RecentVerifications.tsx
│   │   │   │   ├── TrendChart.tsx
│   │   │   │   └── AlertList.tsx
│   │   │   │   └──── AlertList.tsx
│   │   │   │
│   │   │   ├──── Reports/
│   │   │   │   ├──── ReportForm.tsx
│   │   │   │   ├──── ReportList.tsx
│   │   │   │   └──── ReportDetail.tsx
│   │   │   │
│   │   │   ├──── Education/
│   │   │   │   ├──── FraudPatterns.tsx
│   │   │   │   ├──── TipsCard.tsx
│   │   │   │   └──── ResourceList.tsx
│   │   │   │
│   │   │   └──── Common/
│   │   │     ├──── Button.tsx
│   │   │     ├──── Card.tsx
│   │   │     ├──── Input.tsx
│   │   │     ├──── Modal.tsx
│   │   │     ├──── Loader.tsx
│   │   │     └──── Toast.tsx
│   │   │
│   │   ├──── pages/
│   │   │   ├──── Home.tsx
│   │   │   ├──── Verify.tsx
│   │   │   ├──── Dashboard.tsx
│   │   │   ├──── Reports.tsx
│   │   │   ├──── Education.tsx
│   │   │   ├──── Settings.tsx
│   │   │   ├──── Login.tsx
│   │   │   └──── NotFound.tsx
│   │   │
│   │   ├──── services/
│   │   │   ├──── api.ts
│   │   │   ├──── auth.ts
│   │   │   ├──── verification.ts
│   │   │   └──── reports.ts
│   │   │
│   │   ├──── hooks/
│   │   │   ├──── useAuth.ts
│   │   │   ├──── useVerification.ts
│   │   │   ├──── useReports.ts
│   │   │   └──── useToast.ts
│   │   │
│   │   ├──── store/
│   │   │   ├──── index.ts
│   │   │   ├──── authSlice.ts
│   │   │   ├──── verificationSlice.ts
│   │   │   ├──── uiSlice.ts
│   │   │   └── uiSlice.ts
│   │   │
│   │   ├── types/
│   │   │   ├── auth.ts
│   │   │   ├── verification.ts
│   │   │   ├── report.ts
│   │   │   └── api.ts
│   │   │
│   │   ├── utils/
│   │   │   ├── formatters.ts
│   │   │   ├── validators.ts
│   │   │   └── constants.ts
│   │   │
│   │   └── styles/
│   │       ├── globals.css
│   │       └── tailwind.css
│   │
│   └── tests/
│       ├── unit/
│       ├── integration/
│       └── e2e/
│
├── database/                # Database Scripts
│   ├── init/
│   │   └── 01-init.sql
│   ├── seeds/
│   │   ├── rbi_circulars.sql
│   │   └── sender_registry.sql
│   └── backups/
│
├── nginx/                 # Nginx Configuration
│   ├── Dockerfile
│   ├── nginx.conf
│   └── conf.d/
│       └── default.conf
│
├── scripts/               # DevOps Scripts
│   ├── setup.sh
│   ├── deploy.sh
│   ├── backup.sh
│   ├── migrate.sh
│   └── seed-data.sh
│
├── docs/                  # Documentation
│   ├── API.md
│   ├── DEPLOYMENT.md
│   ├── ARCHITECTURE.md
│   ├── ARCHITECTURE.md
│   ├── ML_MODELS.md
│   ├── CONTRIBUTING.md
│   └── images/
│
├── monitoring/                  # Monitoring & Logging
│   ├── prometheus/
│   │   └── prometheus.yml
│   ├── grafana/
│   │   ├── dashboards/
│   │   └── provisioning/
│   └── logs/
│
└── tests/                       # Integration Tests
    ├── e2e/
    ├── load/
    └── security/
```

Key Directory Descriptions

Backend (/backend)

- **cmd**/: Entry points for different services (microservices pattern)
- **internal**/: Private application code (not importable by external packages)
- **api**/: HTTP handlers, middleware, routing
- **models**/: Data models/entities
- **repository**/: Database access layer
- **service**/: Business logic layer
- **queue**/: Message queue integration
- **tests**/: Unit and integration tests

ML Service (/ml-service)

- **app**/: FastAPI application with inference endpoints
- **training**/: Complete training pipeline for all models
- **notebooks**/: Jupyter notebooks for experimentation
- **scripts**/: Standalone scripts for data prep and model management

- **data/**: Training datasets (excluded from git via .gitignore)

Models (/models)

- Stores trained model artifacts
- Each model type has its own directory
- Version controlled using DVC or Git LFS
- Includes configuration files for reproducibility

Frontend (/frontend)

- **components/**: Reusable React components organized by feature
- **pages/**: Page-level components (routing)
- **services/**: API integration layer
- **hooks/**: Custom React hooks
- **store/**: Redux/state management
- **types/**: TypeScript type definitions

Database (/database)

- **init/**: Initial schema creation scripts
- **seeds/**: Sample data for development/testing
- **backups/**: Database backup scripts

Monitoring (/monitoring)

- Prometheus for metrics collection
- Grafana for visualization
- Log aggregation configuration

Configuration Files

- **docker-compose.yml**: Orchestrates all services
- **.env.example**: Template for environment variables
- **Makefile**: Common development commands

- **README.md**: Project overview and setup instructions