GL-One Sprint Report

Sprint Period: GL-One

Overview

This sprint focused on establishing core infrastructure and features for the application, with particular emphasis on portfolio management, database integration, and Progressive Web App (PWA) capabilities. The changes encompass both frontend and backend improvements, creating a solid foundation for future development.

Key Changes

Infrastructure & DevOps

- Configured Vite and PWA plugin integration
- Set up local PostgreSQL database for development
- Implemented configurable Express server port through environment variables
- Added proper environment file management
- Created Replit configuration for deployment compatibility

Portfolio Management

- Implemented comprehensive portfolio value tracking system
- Added account history tracking with detailed change records
- Developed portfolio performance visualization with accurate metrics
- Created dedicated account management pages
- Implemented shared types and utilities for consistency
- Added proper schema validation for data integrity

Backend Development

- Migrated from in-memory storage to PostgreSQL database
- Restructured routes and services for better organization
- Implemented new endpoints:
    - Portfolio value tracking
    - Account history management
    - Change record creation

Progressive Web App Features

- Integrated PWA capabilities with proper configuration
- Implemented icon generation for PWA
- Optimized service worker refresh strategy
- Enhanced offline capabilities

User Experience

- Added FIRE calculator with settings persistence

- Improved portfolio visualization with accurate change tracking
- Enhanced account management workflows
- Streamlined value modification process

Technical Improvements

1. Database Integration

   - Implemented proper database schemas
   - Added transaction support for data integrity
   - Created migration path from memory storage

2. API Enhancements

   - Standardized endpoint responses
   - Implemented proper error handling
   - Added validation layers

3. Frontend Architecture

   - Established shared type system
   - Implemented proper state management
   - Enhanced data visualization components

Impact Analysis

The changes in this sprint have:

1. Established a robust data persistence layer
2. Improved application reliability and offline capabilities
3. Enhanced user experience in portfolio management
4. Created a maintainable and scalable architecture

Next Steps

Future sprints should focus on:

- Enhanced data visualization features
- Additional portfolio analysis tools
- Improved performance monitoring
- Extended offline capabilities

Commit History

e5f5b68 vite config should comply **with** replit
ab9737d theres should be **a** replit nix **file**
e21eaf7 Portfolio graph should display correct change detail **for each time** point
1e6775**a** A user should be able **to** save fire settings **and** use **the** page
e1bfcb6 milestones should use correct schemas **and** endpoints
fbf276f record page is now obsolete **and** should **not** exist
25eb5be **an** account should have its own page **for** modifications
2d2a11f adding **a** change record **for** account should use correct endpoint
667292 api should have portfolio/**value** endpoint
fd5290 portfolio history **from** backend should be correct **and include** change detail
1debbfc adding **a** record **to** change **an** accounts **value to add an** account history **item**
fdf695 portfolio should show correct account totals **and** performance percentages
145983e portfolio should use shared types **and** utilities
e2d9de5 portfolio chart should **not** use dummy data **and** use shared schemas
c5c034 entering **a new** account should **add an** initial history **item**
189300 vite-pwa plugin should be used
ad25fb4 should be able **to** generate icons **for the** pwa application
74d5d19 portfolio context should use **the** correct endpoint **for** accounts
e70d006 server should use db **not** memory **with** mods **for** routes **and** services
a7ee1f1 service worker should **not** aggressively refresh
356fb74 express app listener port should be configurable **from** env
0e877dc git should ignore env **files**
4a7df16 **local** dev should use **local** postgres db