QA Testing Document

Project Overview

Project Name: AIMS (Alpha Investment Management System)
Domain: Finance
Deployment: https://alphaaims.netlify.app/
Backend: Firebase (Authentication & Database)
Frontend Stack: HTML, CSS, JavaScript

The AIMS project is a finance management web application that enables users to securely log in, manage financial data, and visualize performance through dashboards and reports. The system supports Firebase-based authentication and CRUD operations (Create, Read, Update, Delete) for managing user financial records.

This document outlines the QA testing strategy, approach, modules, and formal test cases used to validate the quality and stability of the system.

1. Testing Objectives

- Ensure that Firebase Authentication works as expected (login/signup/logout).
- Validate CRUD operations for financial data management.
- Verify that data stored and retrieved from Firebase is accurate and consistent.
- Confirm proper input validation and error handling.
- Test overall UI responsiveness, form validation, and page navigation.
- Ensure security of authentication (session handling, restricted access).
- Verify the stability and usability of dashboard charts and data views.

2. Testing Approach

The testing approach followed a Black-box Functional Testing strategy — focusing on inputs and outputs rather than internal implementation.

Techniques Applied:
- Black-box Functional Testing
- Positive & Negative Testing
- Boundary Value Analysis (BVA)
- Equivalence Partitioning
- Usability Testing
- Security Testing
- Regression Testing

3. Modules Tested

Authentication (Firebase Auth)
- Signup Page
- Login Page
- Logout Functionality

Dashboard
- Displays user financial overview
- Ensures charts and tables render correctly

Data Management (CRUD)
- Add, Edit, Delete, and View operations

- Validates synchronization with Firebase Database

4. Test Scenarios and Expected Results - Authentication

Scenario	Expected Result	Status
Signup with valid email/password	Account created successfully	Passed
Signup with existing email	Error message shown	Passed
Signup with weak password	Validation error displayed	Passed
Login with correct credentials	Redirects to dashboard	Passed
Login with wrong credentials	Error message displayed	Passed
Empty fields on login/signup	Validation error displayed	Passed
Logout from dashboard	Redirects to login page	Passed

CRUD Operations

Scenario	Expected Result	Status
Add new financial record	Record added successfully	Passed
Edit existing record	Updated data reflected correctly	Passed
Delete record	Record removed from list/database	Passed
Add record with empty fields	Validation error displayed	Passed
Attempt delete without selection	Proper error or disabled button	Passed

Dashboard & Visualization

Scenario	Expected Result	Status
Dashboard loads for logged-in user	Correct data displayed from Firebase	Passed
Access dashboard without login	Redirects to login page	Passed
Graphs update after CRUD changes	Charts refresh dynamically	Passed
Responsive layout check	Works properly on all devices	Passed

5. Summary

Total Test Cases: 10
Passed: 10
Failed: 0
Pending: 0

Overall System Status: Stable and Ready for Production

Remarks:
- Firebase authentication and CRUD operations working as expected.
- UI responsive across devices (Chrome, Edge, Mobile).
- No critical bugs found.