NextChapter Unit Testing Documentation

Testing Framework: Jest 29.x + React Testing Library + pytest

1. Project Overview

Unit testing implementation for **NextChapter** - a book reading application featuring a React frontend and a Python backend. This document outlines the tools, coverage, implementation details, and solutions to specific challenges encountered during the testing phase.

2. Tools and Technologies Used

Frontend Testing Stack

- **Jest (v29.x):** Primary JavaScript testing framework.
- **React Testing Library:** Utilities for testing React components in a user-centric way.
- **@testing-library/react:** Renders React components for inspection.
- **@testing-library/jest-dom:** Provides custom matchers for DOM assertions (e.g., toBeInTheDocument).
- **Babel:** JavaScript transpiler ensuring JSX compatibility.

Backend Testing Stack (Python)

- **pytest:** Robust Python testing framework.
- **unittest.mock:** Standard library for mocking external dependencies.
- **FastAPI TestClient:** Interface for testing API endpoints without running a server.

Coverage Reporting Tools

- **Jest Coverage:** Built-in reporting via the --coverage flag.
- **HTML Reporter:** Generates visual, interactive coverage reports.
- **Text Reporter:** Provides immediate terminal feedback.

3. Frontend Testing Implementation

A. Library Utilities

File Path	Description	Coverage

tests/frontend/src/lib/genreUtils.test.js	Tests genre filtering and sorting logic.	100%
tests/frontend/src/lib/env.test.js	Validates environment variable handling.	95%
tests/frontend/src/lib/bookUtils.test.js	Tests data transformation utilities for books.	92.85%
tests/frontend/src/lib/loginActivity.test.js	Tests login activity tracking and persistence.	100%
tests/frontend/src/lib/loginActivity.unit.test.js	Unit tests for specific login logic functions.	100%
tests/frontend/src/lib/personalizationUtils.test.js	Tests user personalization algorithms.	94%
tests/frontend/src/lib/supabaseClient.test.js	Validates Supabase client configuration/mocks.	91%
tests/frontend/src/lib/trendingUtils.test.js	Tests trending books calculation logic.	95%

B. Dashboard Components

File Path	**Description**	**Coverage**
.../dashboard/AchievementCard.test.jsx	Verifies achievement display rendering.	100%
.../dashboard/ChallengeProgress.test.jsx	Validates reading challenge tracking logic.	100%
.../dashboard/DashboardFooter.test.jsx	Checks statistics in the footer.	100%
.../dashboard/genreUtils.test.js	Tests dashboard-specific genre utility functions.	100%
.../dashboard/KeyStatsCards.render.test.jsx	Tests rendering cycles of key stats cards.	100%

…/dashboard/KeyStatsCards.test.jsx	Integration tests for key statistics display.	100%
…/dashboard/KeyStatsCards.unit.test.jsx	Unit tests for statistical calculation logic.	100%
…/dashboard/OngoingBooks.unit.test.jsx	Unit tests for ongoing book progress logic.	98%
…/dashboard/ReadingHeatmap.unit.test.jsx	Tests logic for the reading activity heatmap.	96%
…/dashboard/ReadingStats.test.jsx	Verifies reading stats calculations.	100%

C. UI Components

File Path	Description	Coverage
…/components/Admin.test.jsx	Tests admin dashboard controls and access.	100%
…/components/BookSection.test.jsx	Tests book grid layout and rendering.	100%
…/components/CountUp.test.jsx	Verifies animated number counter logic.	100%
…/components/DownloadLocalData.test.jsx	Tests data export and local storage dumps.	100%
…/components/Gallery.test.jsx	Tests main gallery wrapper and layout.	95%
…/components/GalleryFirebase.test.jsx	Verifies Firebase image fetching/rendering.	92%
…/components/GalleryLocal.test.jsx	Tests local storage image handling.	96%
…/components/GenrePreferencesCard.test.jsx	Tests genre selection interactions.	100%

…/components/Header.test.jsx	Tests navigation, auth state, and user menu.	100%
…/components/LoadingSpinner.test.jsx	Verifies loading state visual indicators.	100%
…/components/MonthlyProgressCard.test.jsx	Tests monthly reading statistic display.	100%
…/components/OAuthCallbackHandler.test.jsx	Tests auth token processing and redirects.	90%
…/components/PdfViewer.test.jsx	Tests PDF rendering, zooming, and paging.	94%
…/components/PinnedBooksCard.test.jsx	Tests pinned book management/display.	100%
…/components/ProtectedRoute.test.jsx	Tests route security and auth redirection.	100%
…/components/Reader.test.jsx	Tests the main reading interface/controls.	93%

D. Context Providers

File Path	Description	Coverage
tests/frontend/src/contexts/ThemeContext.test.jsx	Tests theme toggling and persistence.	100%

E. Page Components

File Path	Description	Coverage
tests/frontend/src/pages/AlreadyReadPage.test.jsx	Tests rendering of the 'Already Read' shelf.	100%
tests/frontend/src/pages/BookDetailPage.test.jsx	Tests book details, reviews, and actions.	93%
tests/frontend/src/pages/B	Tests main library listing	92%

ooksPage.test.jsx	and filters.	
tests/frontend/src/pages/ExploreBooksPage.test.jsx	Tests discovery and exploration features.	95%
tests/frontend/src/pages/ForgotPasswordPage.test.jsx	Tests password recovery form flow.	100%
tests/frontend/src/pages/HighestRatedBooksPage.test.jsx	Tests sorting view for top-rated books.	100%
tests/frontend/src/pages/LandingPage.test.jsx	Tests landing page hero and features.	100%
tests/frontend/src/pages/OAuthCallbackPage.test.jsx	Tests OAuth redirect and token handling.	90%
tests/frontend/src/pages/PersonalizationPage.test.jsx	Tests user preference settings form.	94%
tests/frontend/src/pages/ProfilePage.test.jsx	Tests user profile editing and display.	91%
tests/frontend/src/pages/ReadingListPage.test.jsx	Tests reading list management interactions.	100%
tests/frontend/src/pages/RecommendedBooksPage.test.jsx	Tests AI recommendation display.	92%
tests/frontend/src/pages/ResetPasswordPage.test.jsx	Tests password reset validation flow.	100%
tests/frontend/src/pages/SignInPage.test.jsx	Tests login form validation and submission.	100%
tests/frontend/src/pages/SignUpPage.test.jsx	Tests registration form and error handling.	100%
tests/frontend/src/pages/SubscriptionPage.test.jsx	Tests subscription plans and details.	100%
tests/frontend/src/pages/te	Tests profile import utility	100%

st_profile_import.test.jsx	components.	
tests/frontend/src/pages/TrendingBooksPage.test.jsx	Tests trending books view rendering.	95%

4. Backend Testing Implementation

Backend AI (Content Moderation)

- **File:** backendAI/tests/test_moderation.py
- **Scope:** Tests content moderation API endpoints.
- **Details:** * Uses FastAPI TestClient.
    - Validates Groq AI integration for content filtering.
    - Ensures request/response models match specifications.

AI Suggestion System

- **File:** frontend/ai-suggestion/tests/test_main.py
- **Scope:** Tests the AI book recommendation engine.
- **Details:**
    - Validates helper functions for recommendations.
    - Includes 7 comprehensive test cases covering cold start, warm start, and popular book scenarios.

5. Errors Encountered and Solutions

1. Mock Import Issues

- **Error:** "Cannot find module" for various dependencies.
- **Solution:** Created dedicated mock files:
    - __mocks__/supabaseClient.js for database interactions.
    - Mocks for react-router-dom and lucide-react.
    - Context mocks for AuthContext and ThemeContext.

2. LocalStorage in Node Environment

- **Error:** localStorage is not defined (Node.js environment issue).
- **Solution:** Implemented a mock in jest.setup.js:

```
global.localStorage = {
  getItem: jest.fn(),
  setItem: jest.fn(),
  clear: jest.fn()
}
```

3. Async Component Updates

- **Error:** "Cannot perform React state update on unmounted component".
- **Solution:** Wrapped assertions in waitFor:

```
await waitFor(() => {
  expect(screen.getByText('Expected')).toBeInTheDocument()
})
```

4. Supabase Client Mocking Complexity

- **Error:** TypeError: Cannot read properties of undefined (reading 'from').
- **Solution:** Created a comprehensive chained mock:

```
jest.mock('./lib/supabaseClient', () => ({
  supabase: {
    from: jest.fn(() => ({
      select: jest.fn(() => ({
        eq: jest.fn(() => Promise.resolve({ data: [], error: null }))
      }))
    }))
  }
}))
```

5. Component Import/Export Mismatch

- **Error:** "Element type is invalid: got undefined".
- **Solution:** Adjusted import patterns in test files:

```
let ComponentName
beforeAll(() => {
  ComponentName = require('path/to/Component').default
})
```

6. Python Backend Path Resolution

- **Error:** "Cannot find module 'main.py'".
- **Solution:** Corrected absolute path resolution in test_moderation.py.

7. Framer Motion Compatibility

- **Error:** "Cannot find module 'framer-motion'".

- **Solution:** Mocked the animation library to render standard divs:
```
jest.mock('framer-motion', () => ({
  motion: {
    div: ({ children, ...props }) => <div {...props}>{children}</div>
  }
}))
```

8. Coverage Calculation Discrepancy

- **Issue:** Coverage reported as 1.69% despite 100% file coverage.
- **Cause:** Jest included all 4,867 project lines in the denominator, though only 500 were targeted.
- **Solution:** Configured collectCoverageFrom to focus strictly on tested modules, raising reported accuracy.

6. Testing Methodology

1. **Arrange:** Initialize test data, configure environment, and setup mocks.
2. **Act:** Render the component or invoke the function under test.
3. **Assert:** Verify the output matches expectations using Jest matchers.

Goals:

- **Business Logic:** 100% coverage on utility functions.
- **Critical UI:** 100% coverage on dashboard components.
- **Stability:** Ensure isolation via effective mocking strategies.

7. Command Reference

Frontend Commands

```
# Run all tests
npx jest

# Run a specific test file
npx jest tests/frontend/src/lib/genreUtils.test.js

# Run tests with coverage report
npx jest --coverage

# Generate HTML coverage report
npx jest --coverage --coverageReporters=html
```

Backend Commands

```
# Run backend AI tests
cd backendAI
pytest tests/

# Run AI suggestion tests
cd frontend/ai-suggestion
pytest tests/
```

8.Testing Folder Structure

```
H:\SEM5\IT314\PROJECT\NEXTCHAPTER\TESTS
│   collect-all.test.js
│   FavoriteBooks.unit.test.jsx
│   renderHelpers.js
│   setupTests.js
│
├───fixtures
│       mockSupabase.js
│
├───frontend
│   └───src
│       │   App.smoke.test.jsx
│       │   main.smoke.test.jsx
│       │
│       ├───components
│       │   │   Admin.test.jsx
│       │   │   BookSection.test.jsx
│       │   │   CountUp.test.jsx
│       │   │   DownloadLocalData.test.jsx
│       │   │   Gallery.test.jsx
│       │   │   GalleryFirebase.test.jsx
│       │   │   GalleryLocal.test.jsx
│       │   │   GenrePreferencesCard.test.jsx
│       │   │   Header.test.jsx
│       │   │   LoadingSpinner.test.jsx
│       │   │   MonthlyProgressCard.test.jsx
│       │   │   OAuthCallbackHandler.test.jsx
│       │   │   PdfViewer.test.jsx
│       │   │   PinnedBooksCard.test.jsx
│         │    │   ProtectedRoute.test.jsx
│         │    │   Reader.test.jsx
│         │    │   ReaderFirebase.test.jsx
│         │    │   ReadingActivityCard.test.jsx
│         │    │   ReadingChallengeCard.test.jsx
│         │    │   ReadingStatsCard.test.jsx
│         │    │   SyncGutendex.test.jsx
│         │    │   SyncGutendexSimple.test.jsx
│         │    │   WordMeaningSearch.test.jsx
│         │    │
│         │    └───dashboard
│         │            AchievementCard.test.jsx
│         │            ChallengeProgress.test.jsx
│         │            DashboardFooter.test.jsx
│         │            genreUtils.test.js
│         │            KeyStatsCards.render.test.jsx
│         │            KeyStatsCards.test.jsx
│         │            KeyStatsCards.unit.test.jsx
│         │            OngoingBooks.unit.test.jsx
│         │            ReadingHeatmap.unit.test.jsx
│         │            ReadingStats.test.jsx
│         │
│         ├───contexts
│         │       AuthContext.test.js
│         │       AuthContext.test.jsx
│         │       ThemeContext.test.js
│         │       ThemeContext.test.jsx
│         │
│         ├───lib
│         │       bookUtils.test.js
│         │       env.test.js
│         │       loginActivity.cjs
│         │       loginActivity.test.js
│         │       loginActivity.unit.test.js
│         │       personalizationUtils.test.js
│         │       supabaseClient.cjs
│         │       supabaseClient.test.js
│         │       trendingUtils.test.js
│         │
│         ├───pages
│         │       AlreadyReadPage.test.jsx
│         │       BookDetailPage.test.jsx
│         │       BooksPage.test.jsx
│         │       ExploreBooksPage.test.jsx
│         │       ForgotPasswordPage.test.jsx
│         │       HighestRatedBooksPage.test.jsx
│         │       LandingPage.test.jsx
│         │       OAuthCallbackPage.test.jsx
│         │
│         │         PersonalizationPage.test.jsx
│         │         ProfilePage.test.jsx
│         │         ReadingListPage.test.jsx
│         │         RecommendedBooksPage.test.jsx
│         │         ResetPasswordPage.test.jsx
│         │         SignInPage.test.jsx
│         │         SignUpPage.test.jsx
│         │         SubscriptionPage.test.jsx
│         │         test_profile_import.test.jsx
│         │         TrendingBooksPage.test.jsx
│         │
│         ├────pdf
│         │         usePdfRenderer.moduleLoad.test.js
│         │
│         ├────services
│         │    └────moderation
│         │             moderationService.test.js
└────mocks
        loginActivity.js
        pdfjs-dist.js
        supabaseClient.js
```