WEBREPORT

A blazingly fast analyzer for web content.

Project Overview

WebReport is a web content analysis tool built for Software Engineering education at HTWG.

Key Features

- **Web Content Download -** Fetch and analyze websites
- **Text Analysis -** Character/word count, top words, complexity scoring
- **Content Filtering -** Filter content by keywords
- **Undo/Redo -** Full command history support
- **Session Persistence -** Save/load sessions in XML or JSON
- **Dual Interface -** TUI (Terminal) + GUI (ScalaFX)
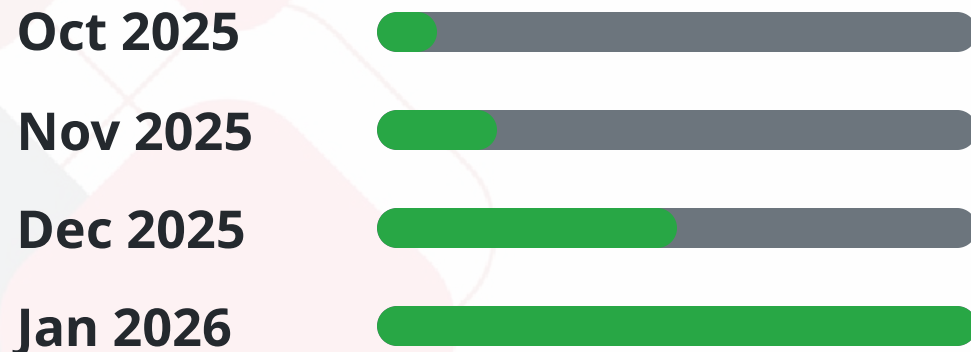
Development Timeline

Oct 2025 Project Initialization — Repository created

Nov 2025 Core Development — MVC Architecture setup

Dec 2025 Feature Implementation — TUI & GUI, Tests, Scoverage
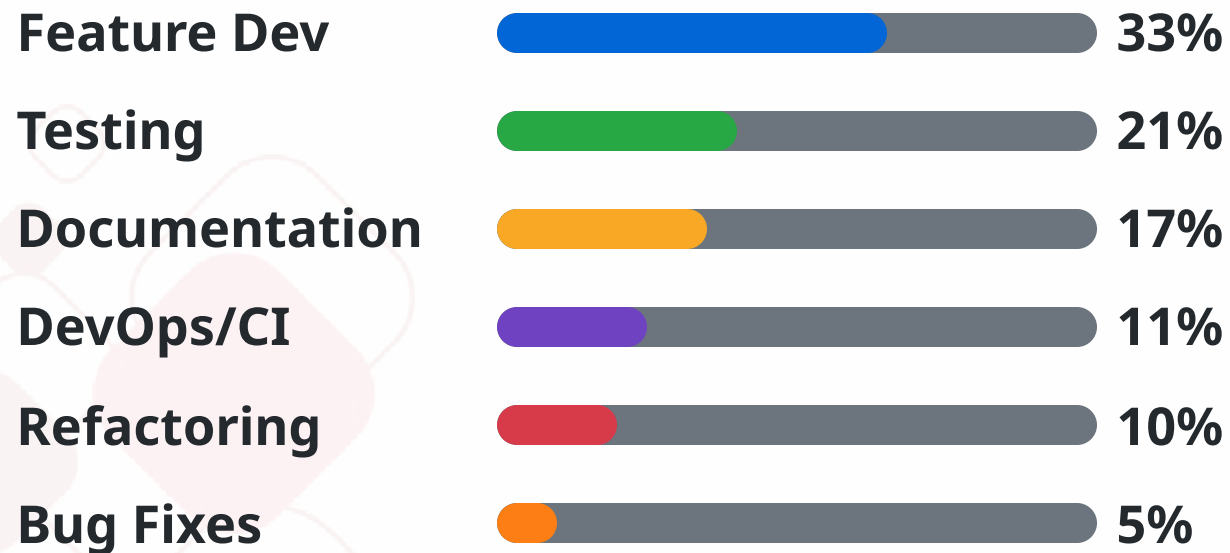
Jan 2026 Final Sprint — Docker, Coverage optimization, GUI improvements

Activity Intensity

Oct 2025

Nov 2025

Dec 2025

Jan 2026

Commit Activity Distribution

Based on 105 commits (Oct 2025 - Jan 2026)

Category		Percentage
Feature Dev		**33%**
Testing		**21%**
Documentation		**17%**
DevOps/CI		**11%**
Refactoring		**10%**
Bug Fixes		**5%**

■ Feature Development (35)   ■ Testing & Coverage (22)   ■ Documentation (18)

■ DevOps & CI/CD (12)   ■ Refactoring (10)   ■ Bug Fixes (5)

Technology Stack

Component	Technology	Version
Language	Scala 3	3.4.2
Build Tool	SBT	latest
GUI Framework	ScalaFX	21.0.0-R32
DI Framework	Google Guice	7.0.0
Testing	ScalaTest	3.2.19
Coverage	Scoverage + Coveralls	2.1.0

Codebase Structure

src/main/scala/de/htwg/webreport/

```
├── aview/        ━━━━━━━━  ~350 LOC
│   ├── Gui.scala
│   ├── Tui. scala
│   ├── TuiState.scala
│   └── Renderer.scala
├── controller/   ━━━━━━  ~150 LOC
│   └── sessionManager/
├── model/        ━━━━━━  ~250 LOC
│   ├── data/
│   ├── analyzer/
│   ├── webClient/
│   └── fileio/
└── util/         ━━━━  ~60 LOC
├── Command.scala
└── Observer.scala
```

Test Suite Overview

Test File	Purpose
WebReportSpec. scala	Integration tests
IntegrationSpec.scala	Full system tests
TuiSpec.scala	Terminal UI tests
TuiStateSpec.scala	State pattern tests
GuiSpec.scala	GUI component tests
RendererSpec.scala	Output rendering tests
DataSpec.scala	Data model tests
ObserverSpec.scala	Observer pattern tests

Exclusions: Main 87%, Tui 94%, TuiState 97%, Renderer 98%, Gui 64%

Architecture Overview

VIEW LAYER

TUI (Terminal)   GUI (ScalaFX)

← Observer P. | Template P. | Decorater P. | State P.

CONTROLLER LAYER

SessionManager (Observable + Originator)

← Command Pattern | Memento Pattern

MODEL LAYER

Analyzer (Trait)   WebClient (Trait)   FileIO (XML/JSON)

← Strategy Pattern

Pattern	Purpose	Location
Observer	UI Synchronization	util/Observer.scala
Command	Operation Encapsulation	util/Command.scala
Memento	Undo/Redo Support	util/Command.scala
State	TUI Mode Switching	aview/TuiState.scala
Template Method	Report Structure	aview/Renderer.scala
Decorator	Feature Extensions	aview/Renderer.scala
Strategy	Interchangeable Serialization	model/fileio/
DI (Guice)	Loose Coupling	WebReportModule.scala

Component Complexity

SessionManager ●●●●●

Renderer ●●●●○

Tui ●●●○○

Gui ●●●○○

Data ●●○○○

FileIO ●●○○○

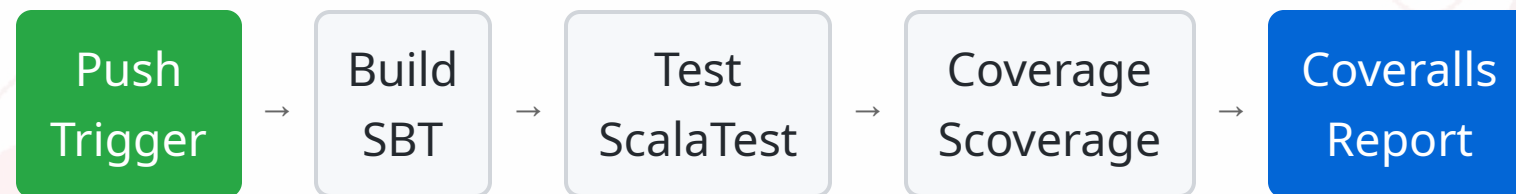Low ◄─────────────► High

Dependencies

WebReport 1.0.0

├── org.scala-lang » scala3-library `3.4.2`

├── org.scalafx » scalafx `21.0.0-R32`

│   └── org.openjfx » javafx-* `21.0.2`

├── com.google.inject » guice `7.0.0`

├── org.scala-lang.modules » scala-xml `2.2.0`

├── com.typesafe.play » play-json `2.10.4`

└── **[test]**

├── org.scalatest » scalatest `3.2.19`
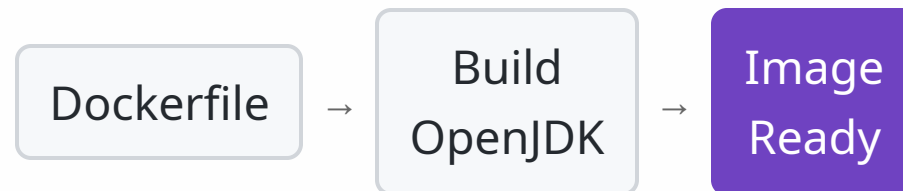
└── org.scalactic » scalactic `3.2.14`

Plugins

├── org.scoverage » sbt-scoverage `2.1.0`

└── org.scoverage » sbt-coveralls `1.3.15`

CI/CD Pipeline

GitHub Actions Workflow

Push Trigger → Build SBT → Test ScalaTest → Coverage Scoverage → Coveralls Report

Docker Build

Dockerfile → Build OpenJDK → Image Ready

Deployment Options

Method	Command	Description
Local Development	`sbt run`	Launches TUI + GUI simultaneously
Docker Container	`./docker-run.sh`	X11 forwarding for GUI support
Test Mode	`sbt coverage test`	Run tests with coverage report

Key Takeaways

- Clean MVC Architecture with clear layer separation

- 8 Design Patterns applied in practice

- Modern Scala 3 with trait-based interfaces

- Dual Interface (TUI + GUI) via Observer pattern

- Full Undo/Redo via Command + Memento patterns

- Extensible serialization with Strategy pattern

- Comprehensive test suite with Scoverage

- Docker support for containerized deployment

- CI/CD pipeline with GitHub Actions

Project Resources

- **GitHub Repository**
  github.com/Herbert-Haase/WebReport

- **License**
  MIT License — Free for educational and commercial use.

- **Documentation**
  Check the `README.md` for build instructions and Scoverage reports.

Thank you for your attention! Questions?