



IT314 — Software Engineering

Mutation Testing

Project Title: Next Chapter

Online Book Reading Platform

Group — 10

Extension Used:

StrykerJS (Stryker Mutation Testing Framework)

Mutation Testing Report

0.1 0.1 Content Moderation Testing

For the content moderation module of our project (implemented in Python), **MutPy** — a mutation testing tool for Python — was used.

MutPy was applied to validate and strengthen:

- Detection of inappropriate or harmful words
- Filtering user comments on books

The mutation testing using MutPy helped expose weaknesses such as:

- Filters failing on mixed-case offensive patterns
- Missing edge cases involving punctuation
- Logic that passed even when parts of filters were mutated

Target

- `main`

Tests [11]




























- `test_main` [0.331 s]

Result summary

-  Score - 100.0%
-  Time - 19.0 s

Mutants [9]

-  - 9
-  - 0
-  - 0
-  - 0

100.0%						
#	Module	Operator	Tests run	Duration	Result	
1		COD [14]	2	2.038 s		
2		COI [14]	2	1.234 s		
3		COI [101]	2	1.484 s		
4		COI [109]	2	2.521 s		
5		COI [167]	2	1.54 s		
6		COI [167]	2	1.75 s		
7		ROR [101]	2	1.528 s		
8		ROR [136]	2	1.816 s		
9		ROR [136]	2	3.386 s		

0.2 0.2 Website Testing (Mutation Analysis)

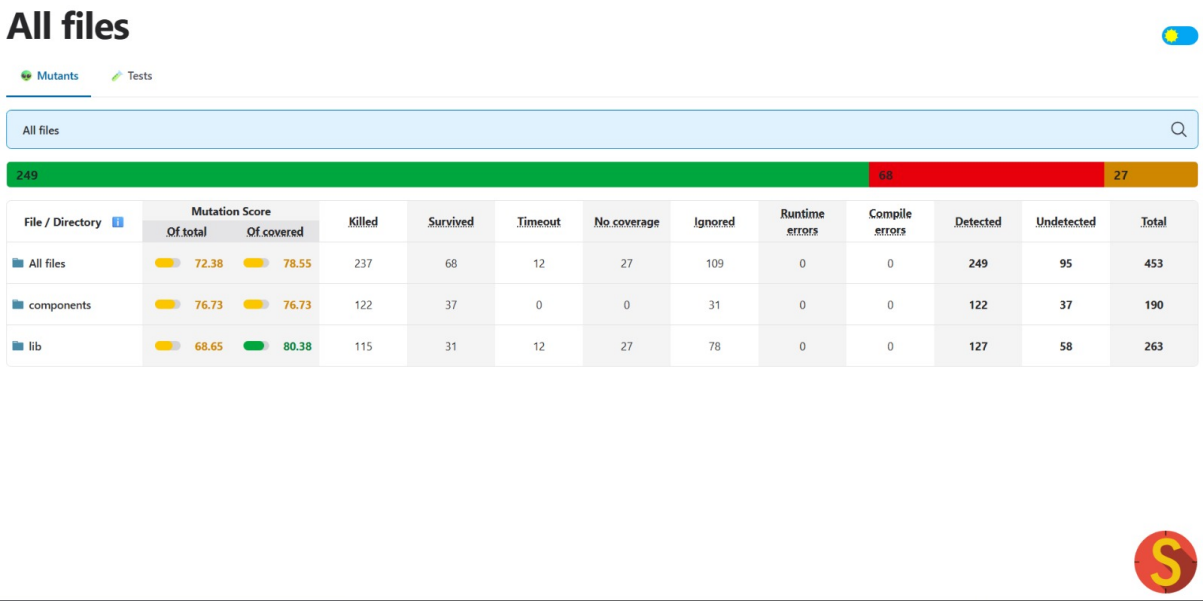
The main Next Chapter web application is built using React (Vite) and Supabase as the backend.

Mutation testing for the website was performed using **StrykerJS**, which evaluates:

- Component rendering logic
- Hook-based state management
- API and database interactions
- Error handling and fallback UI

StrykerJS helped identify untested flows such as unexpected null states, API failures, and incorrect assumptions in UI rendering.

• Mutants Screenshot



components

Mutants

Tests

All files / components

249

68

27

File / Directory	Mutation Score		Killed	Survived	Timeout	No coverage	Ignored	Runtime errors	Compile errors	Detected	Undetected	Total
	Of total	Of covered										
components	76.73	76.73	122	37	0	0	31	0	0	122	37	190
CountUp.jsx	75.41	75.41	46	15	0	0	20	0	0	46	15	81
ProtectedRoute.jsx	65.52	65.52	38	20	0	0	1	0	0	38	20	59
TrendingBooks.jsx	95.00	95.00	38	2	0	0	10	0	0	38	2	50

lib

Mutants

Tests

All files / lib

249



68

27

File / Directory	Mutation Score		Killed	Survived	Timeout	No coverage	Ignored	Runtime errors	Compile errors	Detected	Undetected	Total
	Of total	Of covered										
lib	68.65	80.38	115	31	12	27	78	0	0	127	58	263
bookUtils.js	94.12	94.12	13	1	3	0	5	0	0	16	1	22
personalizationUtils.js	96.23	96.23	51	2	0	0	18	0	0	51	2	71
trendingUtils.js	52.17	68.18	51	28	9	27	55	0	0	60	55	170

● Tests Screenshot

All tests

 Mutants  Tests

All tests					
249					
File / Directory		Killing	Covering	Not Covering	Total tests
All tests		60	162	63	285
components		33	84	40	157
lib		27	78	0	105
App.test.jsx		0	0	23	23





components

 Mutants  Tests

All tests / components					
249					
File / Directory		Killing	Covering	Not Covering	Total tests
components		33	84	40	157
Admin.test.jsx		0	1	0	1
CountUp.test.jsx		13	29	0	42
Header.test.jsx		0	0	37	37
ProtectedRoute.test.jsx		9	15	3	27
TrendingBooks.test.jsx		11	39	0	50



lib

 Mutants  Tests

All tests / lib					
249					
File / Directory		Killing	Covering	Not Covering	Total tests
lib		27	78	0	105
bookUtils.test.js		4	27	0	31
personalizationUtils.test.js		15	23	0	38
trendingUtils.test.js		8	28	0	36



Overall Mutation Score

Mutation score is calculated using the standard formula:

$$MutationScore = \frac{KilledMutants}{Killed + Survived + Timeout} \times 100$$

From the mutation testing results (StrykerJS):

- **Killed Mutants:** 237
- **Survived Mutants:** 68
- **Timeout Mutants:** 12

Thus,

$$MutationScore = \frac{237}{237 + 68 + 12} \times 100 = \frac{237}{317} \times 100 = 74.76\%$$

Final Mutation Score: 74.76%