

# Software Testing Assignment - E-Commerce Utility Library

coverage 98%  Test and Coverage passing

## Project Overview

This repository contains comprehensive unit tests for a utility library used in an e-commerce application selling food products from small vendors.

**Course:** COMP.SE.200-2025-2026-1 - Software Testing

**Institution:** Tampere University

**Team Members:**

- Md Ozayer Islam (151960099)
- Sayem Rahman (152132363)

## Test Coverage

- **302 unit tests** implemented
- **98.55% line coverage** (exceeds 80% requirement)
- **94.95% branch coverage** (exceeds 75% requirement)
- **10 critical functions** tested comprehensively

## Functions Tested

1. **isEmpty** (Critical) - Form validation
2. **get** (Critical) - Safe object access
3. **filter** (Critical) - Product search
4. **add** (Critical) - Price calculations
5. **toNumber** (High) - Type conversion
6. **reduce** (High) - Cart totals
7. **map** (High) - Data transformation
8. **defaultTo** (High) - Optional fields
9. **clamp** (Medium) - Range validation
10. **capitalize** (Medium) - Text formatting

## Quick Start

## Prerequisites

- Node.js 18+
- npm

## Installation

```
npm install
```

## Run Tests

```
npm test
```

## Run Tests with Coverage

```
npm run test:coverage
```

## View Coverage Report

```
open coverage/lcov-report/index.html
```

## Documentation

- [Testing Instructions](#) - Complete guide for running tests locally
- [Bug Report](#) - Detailed documentation of 4 bugs found during testing

## Bugs Found

During comprehensive testing, we discovered **4 significant bugs**:

1. 🛑 **clamp()** - CRITICAL: Always returns lower bound (completely broken)
2. 🟡 **add()** - HIGH: String concatenation instead of numeric addition
3. 🟡 **filter()** - HIGH: Returns `[]` instead of `[]` for empty results
4. 🟡 **defaultTo()** - MEDIUM: Doesn't handle NaN values

See [BUG\\_REPORT.md](#) for detailed analysis and business impact.

# CI/CD Pipeline

- **GitHub Actions:** Automated testing on every push
- **Coveralls:** Coverage tracking and reporting
- **Quality Gates:** 80% line coverage, 75% branch coverage

## Project Structure

```
.
├── src/                                # Source code (DO NOT MODIFY)
│   ├── .internal/                     # Excluded from testing
│   └── *.js                           # Utility functions
├── __tests__/                          # Test suites
│   ├── isEmpty.test.js                # Pre-planned tests
│   ├── isEmpty.ai.test.js             # AI-generated tests
│   ├── get.test.js
│   ├── get.ai.test.js
│   ├── add.test.js
│   ├── filter.test.js
│   ├── toNumber.test.js
│   ├── reduce.test.js
│   ├── map.test.js
│   ├── defaultTo.test.js
│   ├── clamp.test.js
│   └── capitalize.test.js
├── coverage/                          # Coverage reports (generated)
├── BUG_REPORT.md                      # Bug documentation
├── TESTING_INSTRUCTIONS.md            # Testing guide
└── README.md                         # This file
```

## Testing Approach

### Self-Designed Tests

- 18 tests for isEmpty and get (Phase 1 pre-planned)
- 224 tests for remaining 8 functions
- Focus on e-commerce scenarios and business logic

### AI-Generated Tests

- 105 tests for isEmpty and get
- Comprehensive edge case coverage
- Generated using Microsoft Copilot

**Total: 302 Tests**

- Normal cases
- Edge cases
- Boundary conditions
- E-commerce specific scenarios

## Quality Assessment

### ✅ Achievements

- All 302 tests passing
- Coverage exceeds requirements
- 4 bugs discovered and documented
- CI/CD pipeline functional

### ⚠️ Findings

- Library is **NOT production-ready** due to critical bugs
- Requires bug fixes before deployment
- Further integration and E2E testing recommended

## License

Source code folder contains a separate license file that must **NOT** be removed under any circumstances! Removing this license file directly violates terms and conditions of the software under testing.

## Links

- **GitHub Repository:** <https://github.com/Ozayer/software-testing-class-project>
- **Coveralls:** <https://coveralls.io/github/Ozayer/software-testing-class-project>
- **GitHub Actions:** <https://github.com/Ozayer/software-testing-class-project/actions>

**Last Updated:** December 2025