### **Comparison Report: Manual vs. Randoop Generated Test Cases for Library Management Project**

#### Introduction

This report compares the test cases written manually, specifically in "LibraryTest.java", with those generated by Randoop for the Library Management Project. The key metrics under comparison are code coverage, readability, and usefulness.

#### 1. Code Coverage

* Manual Test Cases:
  + The manual test cases, as detailed in "LibraryTest.java", focus primarily on critical functional areas of the software, such as registering the user and signing in with those credentials. The test cases are designed to cover significant scenarios that are essential for the application's stability and performance.
* Randoop Generated Test Cases:
  + According to the "RandoopTestResults.html", Randoop achieved 65% instruction coverage and 50% branch coverage. This suggests that while Randoop provides a good spread across the codebase, it may not always focus deeply on the most critical paths that manual tests target.

#### 2. Readability

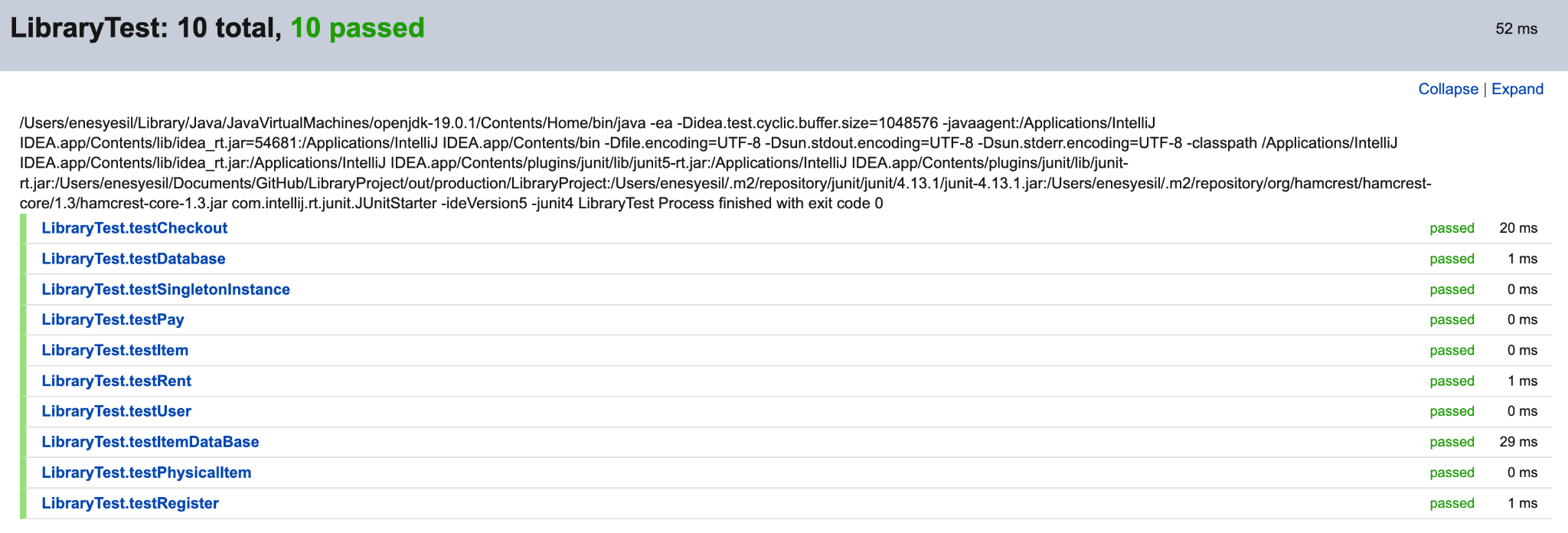
* Manual Test Cases:
  + The "LibraryTest.java" contains tests that are well-documented with clear and meaningful test method names and comments that explain the purpose of each test. This practice significantly aids in maintaining the test suite and facilitates easier updates and reviews by developers.
* Randoop Generated Test Cases:
  + Randoop’s test cases often lack descriptive names and detailed comments, which can make them harder to interpret and maintain. Because they're automated, they might not match how developers think about the code, so they might get overlooked during updates.

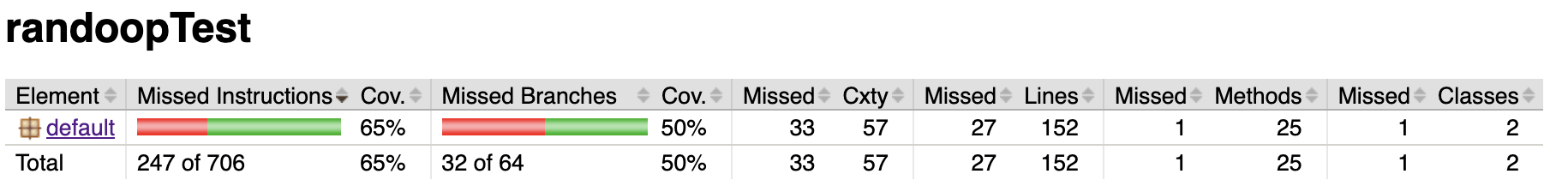
#### 

#### 3. Usefulness

* Manual Test Cases:
  + The manual tests are highly tailored to the project's requirements, focusing on areas that are most likely to affect the application's functionality if broken. This makes them extremely valuable for regression testing and ensures that key features perform as expected after updates or refactoring.
* Randoop Generated Test Cases:
  + While Randoop tests are useful for exploring a wide range of inputs and detecting unexpected edge cases, they may generate some tests that are less relevant to the project’s immediate and critical testing needs. However, their broad coverage is beneficial for discovering hidden issues.

#### Conclusion

In the Library Management Project, the manual testing approach provides depth and targeted testing that is crucial for ensuring the functionality and stability of key features. On the other hand, Randoop complements this by offering broad coverage that helps uncover less obvious faults. For the most effective testing strategy, a hybrid approach should be employed, utilizing the strengths of both manual precision and Randoop's extensive coverage capabilities.



**Participation**

| **Name of the team member** | **Tasks completed by the team member** | **Participation of the team member** |
| --- | --- | --- |
| Mohamed Ahmed | Randoop test | 25% |
| Stefan Hightower | Gui classes implementation | 25% |
| Hulya Yasar | Reports | 25% |
| Enes Yesil | Manuel test | 25% |