# COMP2005 Assessment 2: Report

## Assessment Materials

|  |  |
| --- | --- |
| D2 GitHub Repository: | <https://github.com/Plymouth-University/comp2005-assessment2-corey-richardson> |
| D3 YouTube Link: |  |

## Introduction

## Implementation

## Unit and Integration Testing for Part A

// A description of the test cases you have designed.  
// A description of the structure and role of any mock objects you have used.  
// An explanation of how to run the unit and integration tests, and continuous integration (in code environment).

Each layer of the application was tests according to its role in the application architecture. Classes and service methods were tested using Unit Tests to verify their behaviour in isolation from the API returns. Services were tested using mocked dependencies and returns to ensure that the implementations matched expected business logic. Controllers were tested using Integration Tests to verify the API’s behaviour.

|  |  |  |
| --- | --- | --- |
| **Test File** | **Layer Tested** | **Test Types** |
| AdmissionControllerTest.java | Controller | Integration Tests |
| AdmissionServiceTest.java | Service | Unit Tests |
| AdmissionTest.java | Class | Unit Tests |
| AllocationTest.java | Class | Unit Tests |
| ApiHelperTest.java | Service | Unit Tests |
| Comp2005ApiApplicationTests.java | Application Context | Smoke Test |
| EmployeeTest.java | Class | Unit Tests |
| PatientControllerTest.java | Controller | Integration Tests |
| PatientServiceTest.java | Service | Unit Tests |
| PatientTest.java | Class | Unit Tests |

The following tables list each test case and explain their purpose in testing the application.

|  |  |  |
| --- | --- | --- |
| **AdmissionControllerTest.java** | | |
| **Layer Tested** | Controller | |
| **Test Types** | Integration Tests | |
| returnsExpectedMonth | | Checks that the controller returns the expected month string when admissions exist and are returned by the Service layer. |
| returnsEmptyWhenNoAdmissions | | Ensures that the controller returns a fallback message that is passed by the Service layer if no admissions exist. |
| handlesErrorsGracefully | | Checks that the controller returns a fallback message that is passed by the Service layer if it encounters an error or exception. |

|  |  |  |
| --- | --- | --- |
| **AdmissionServiceTest.java** | | |
| **Layer Tested** | Service | |
| **Test Types** | Unit Tests | |
| **MonthWithMostAdmissionsTests** | | |
| returnsExpectedMonth | |  |
| returnsEmptyWhenNoAdmissions | |  |
| handlesErrorsGracefully | | Checks that the controller returns a fallback message that is passed by the Service layer if it encounters an error or exception. |

|  |  |  |
| --- | --- | --- |
| **AdmissionTest.java** | | |
| **Layer Tested** | Class | |
| **Test Types** | Unit Tests | |
| testParameterlessNotNull | |  |
| testConstructedNotNull | |  |
| testSetAndGetId | |  |
| testSetAndGetPatientId | |
| testSetAndGetAdmissionDate | |
| testSetAndGetDischargeDate | |
| constructedAdmissionTest | |  |

|  |  |  |
| --- | --- | --- |
| **AllocationTest.java** | | |
| **Layer Tested** | Class | |
| **Test Types** | Unit Tests | |
| testParameterlessNotNull | |  |
| testConstructedNotNull | |  |
| testSetAndGetId | |  |
| testSetAndGetAdmissionId | |
| testSetAndGetEmployeeId | |
| testSetAndGetStartTime | |
| testSetAndGetEndTime | |
| constructedAllocationTest | |  |

|  |  |  |
| --- | --- | --- |
| **ApiHelperTest.java** | | |
| **Layer Tested** | Service | |
| **Test Types** | Unit Tests | |
| **AdmissionTests** | | |
| getAllAdmissions\_returnsAdmissions | |  |
| getAdmissionById\_returnsAdmission | |  |
| getAdmissionById\_handles404 | |  |
| **AllocationTests** | | |
| getAllAllocations\_returnsAllocations | |  |
| getAllocationById\_returnsAllocation | |  |
| getAllocationById\_handles404 | |  |
| **EmployeeTests** | | |
| getAllEmployees\_returnsEmployees | |  |
| getEmployeeById\_returnsEmployee | |  |
| getAllocationById\_handles404 | |  |
| **PatientTests** | | |
| getAllPatients\_returnsPatients | |  |
| getPatientById\_returnsPatient | |  |
| getPatientById\_handles404 | |  |

|  |  |  |
| --- | --- | --- |
| **Comp2005ApiApplication.java** | | |
| **Layer Tested** | Application Context | |
| **Test Types** | Smoke Test | |
| contextLoads | | Ensures that the Spring Boot application context starts up correctly. |

|  |  |  |
| --- | --- | --- |
| **EmployeeTest.java** | | |
| **Layer Tested** | Class | |
| **Test Types** | Unit Tests | |
| testParameterlessNotNull | |  |
| testConstructedNotNull | |  |
| testSetAndGetId | |  |
| testSetAndGetFirstName | |
| testSetAndGetLastName | |
| constructedEmployeeTest | |  |

|  |  |  |
| --- | --- | --- |
| **PatientControllerTest.java** | | |
| **Layer Tested** | Controller | |
| **Test Types** | Integration Tests | |
|  | |  |
|  | |  |

|  |  |  |
| --- | --- | --- |
| **PatientServiceTest.java** | | |
| **Layer Tested** | Service | |
| **Test Types** | Unit Tests | |
|  | |  |
|  | |  |

|  |  |  |
| --- | --- | --- |
| **PatientTest.java** | | |
| **Layer Tested** | Class | |
| **Test Types** | Unit Tests | |
|  | |  |
|  | |  |

### Use of the Test Driven Development Approach

### Tools Used

// Junit, Mockito

## Metrics and Code Coverage

## Usability Testing for Part B

## Functional Test Plan

## System Testing

## Tools and Practices

### Version Control

### GitHub Actions, CI/CD

// On push  
// GitHub Local Actions  
// YAML snippet

|  |  |
| --- | --- |
| GitHub Local Actions for VS Code (Sanjula Ganepola) | <https://marketplace.visualstudio.com/items?itemName=SanjulaGanepola.github-local-actions> |

## Evaluation