**Code Refactoring Report for**

**Java Refactoring Test Project**

**Submitted By:**

Sajjad Ashraf

ssajjadashraf@yahoo.com

**Date:**

09 Oct, 2018

**Table of Contents**

[1. Introduction 3](#_Toc507624900)

[2. Conceptual Design(s): 3](#_Toc507624901)

[2.1 Transforming user requirements into a conceptual model: 3](#_Toc507624902)

[2.1.1 Multiple Interfaces 3](#_Toc507624903)

[2.1.2 Adaptive Menu 5](#_Toc507624904)

[2.1.3 Persistence 7](#_Toc507624905)

[3. Selecting the interface metaphor: 8](#_Toc507624906)

[4. Selecting interaction type: 10](#_Toc507624907)

[5. Prototyping: 10](#_Toc507624908)

[5.1 Prototype - 1: 11](#_Toc507624909)

[5.1.1 Usability of the design: 18](#_Toc507624910)

[5.1.2 Error tolerance of the design: 18](#_Toc507624911)

[5.1.3 Efficiency of the design: 19](#_Toc507624912)

[5.2 Prototype-2: 19](#_Toc507624913)

[5.2.1 Usability of the design: 25](#_Toc507624914)

[5.2.2 Error tolerance of the design: 26](#_Toc507624915)

[5.2.3 Efficiency of the design: 26](#_Toc507624916)

[5.3 Prototype - 3: 26](#_Toc507624917)

[5.3.1 Usability of the design: 34](#_Toc507624918)

[5.3.2 Error tolerance of the design: 35](#_Toc507624919)

[5.3.3 Efficiency of the design: 35](#_Toc507624920)

[6. Team Contribution: 36](#_Toc507624921)

[7. References 37](#_Toc507624922)

# 

1. Introduction

The task is to refactor the ‘java-test’ project in a way that it would break any existing functionality, test cases and improve the code readability, extensibility and maintainability.

2. Steps for Improvements:

* Java docs, Code formatting :

I have added Java docs in every class it will help to understand the code easily. I have also formatted the code (code indentation) in every class of the project so that it looks consistent and easy to read. I also remove unused variable and imports.

* Handle Concurrent Requests

To facilitate concurrent requests, I have introduced ‘**synchronized**’ code block where we are adding, updating and deleting a user in ‘UserDao’ class.

|  |
| --- |
|  |

We can also handle concurrent request by using following code Collections.synchronizedList(new ArrayList<>());

It is preferable to synchronize only part of a method which can cause a concurrency issue.

* Removing element while Iterating through Array List

‘deleteUser’ method of ‘UserDao’ has been updated to avoid java.util.ConcurrentModificationException .

* Update, Delete and Find user on the base on his email
* Design pattern Singleton
* User porpr HTTP method UserResource
* Introduce logging.

7. References

* User Interface Prototyping [http://agilemodeling.com/artifacts/uiPrototype.htm]
* Online Tool for prototype designing. [https://proto.io/]
* Lecture from slides and book.