# **HANOI UNIVERSITY Faculty of Information Technology (FIT)**

**SOFTWARE QUALITY ASSURANCE**

**CODE INSPECTION REORT**

**Faculty**: **Information Technology**

**Module Name**: **Software Engineering 2**

**Instructor: Dang Dinh Quan**

**Semester**: **Spring 2021**

**Topic**: **Amazing e-commerce Website**

**Group: Tutorial 2 Group 5**

**Group Members**: **1801040115 – Ly Trung Kien**

**1801040130 – Le Duc Long**

**1701040013 – Ton Nu Tu Anh**

**1701040053 – Dang Tran Trung Hieu**

**1801040218 – Nghiem Thi Xuan Thuy**

**Table of Content**

1. **Introduction**
   1. **Purpose of this document**

This document represents process to find defects of maintainability , reliability and functionally at the final phases and orientate any advancement for the process.

* 1. **References:**

All references are mentioned in the end of the report.

* 1. **Coding Convention**

To make the code cleaner, more readable and easier to maintain, there are some common coding convention that we have to applied in the source code:

* + - Identifiers name have to follow **camelCase** concept and have to begin with a character, not a number. Global variables and constants have been written in UPPERCASE
    - Always put spaces around operators and after commas.
    - Always use 2 spaces for indentation of code block and no use tab for indentation.
    - Always end a simple statement with a semicolon.
    - Keep line length under 80 characters.
    - Place the closing bracket on a new line, without a leading space.

Moreover, the source code avoid as much as possible using comment to explain meaning of code, which will make code become more complex

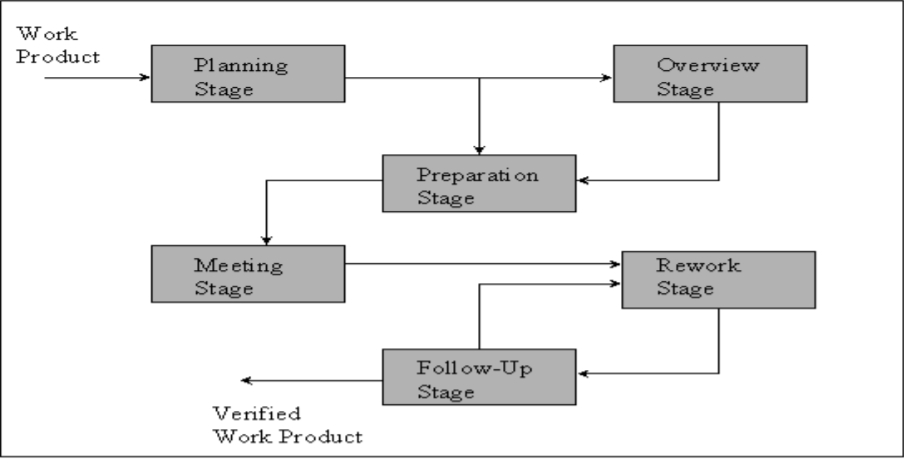
* 1. **Defect checklist**

During the inspection process, we focus on create categories for the defects, especially Coding Convention and Logic Errors. Methods that put into defects will be programming-language-specific. These method will be stick into at Appendix at the end of the report.

* 1. **Code Inspector roles:** 
     + Moderator, Author: Ly Trung Kien
     + Inspector: Le Duc Long, Ton Nu Tu Anh
     + Recorder: Le Duc Long, Ton Nu Tu Anh, Nghiem Thi Xuan Thuy, Dang Tran Trung Hieu

1. **Code Inspection Process**
   1. **Description**

Our code inspection process contains 6 steps:



* + - **Planning:** The moderator plan the code inspection.
    - **Overview meeting:** The author describes the background of work product
    - **Preparation:** Each inspector examines the work product to identify process defects.
    - **Inspection meeting:** During this meeting, the readers read through the work product, part by part and the inspectors point out the defects for every part.
    - **Rework:** The author makes changes to the work product according to the action plans from the inspection meeting.
    - **Follow up:** The changed are checked to make sure that everyting is correct.
  1. **Inspection Meetings**

The code inspection meetings will be held 1 time a week because of Covid-19 pandemic. Google Meet is the chosen meeting platform to hold the meeting. In case of Google Meet is broken, we will use Discord instead. The first meeting is held in March 10th, 2021. In general, all member must be online in this meeting. We use Individual Reviewer Log form during Preparation stage, Log Form during Meeting Stage, Inspection Issue Log, Inspection Checklist form and Inspection Summary in the end of Code Inspection Process. Those form will be stick into Appendix in the end of the report.

1. **Module Inspected**

The top-level inspected module list:

* Log in/Log out
* Account Management( Include Edit Password)
* Cart Management
* Product Management
* Order Management

Evaluation:

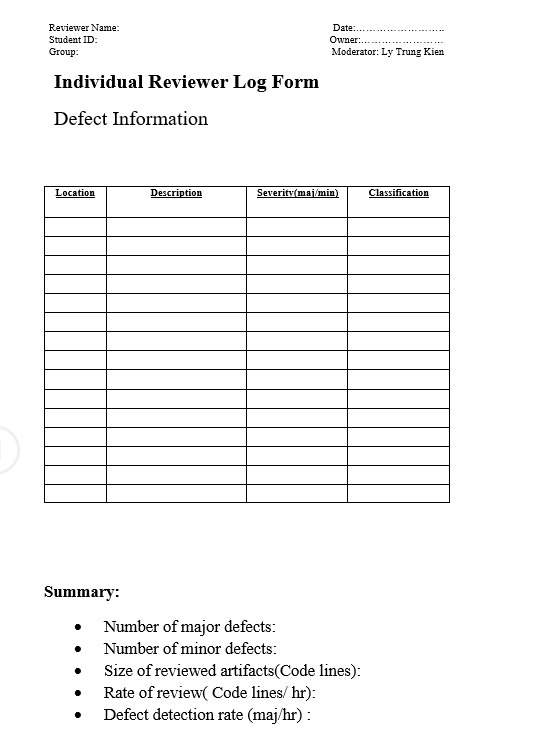
* Almost functions fit with the design and software architecture. Sometimes transactions of Cart management is interupt because of functional changes, and register function does not validate the inserted information.
* The source code is mostly compliant with coding conventions.
* Opportunity for code reuse: 3 of 5 scale.
* Compare planned algorithms to actual algorithms used in functions: 4 of 5 scale.
* Compare planned IO file to actual IO file: 5 of 5 scale.
* Compare planned data structure to actuals data structure: 5 of 5 scale
* Copare planned UX/UI to actuals UX/UI: 4 of 5 scale.

1. **Defects**

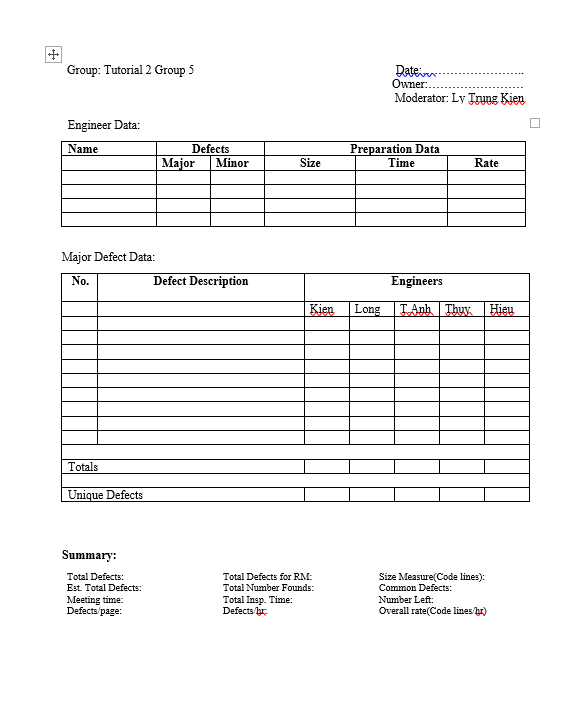
The report include very few effect than expected. The defects found contains coding convention non-compliant and a little bit misunderstandable elements. Those defects do not bring any serious effect to our product. But with each defects found, we still need to consider some important requirements in our process:

* With each defects, clearly describe it in most deficient and understandable way.
* Instruct whether defects has been fixed or not:
  + If it was fixed, briefly described the fixing process outside the code inspection.
  + If it was not fixed yet, indicate that it would be fixed later.
  + If it was not planning to be dealed with, discover its difficulty or insufficiently beneficial.

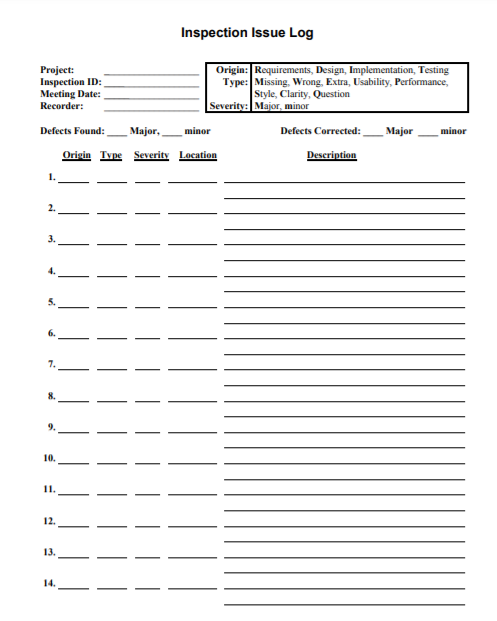
**Appendix. Individual Reviewer Log Form**

****

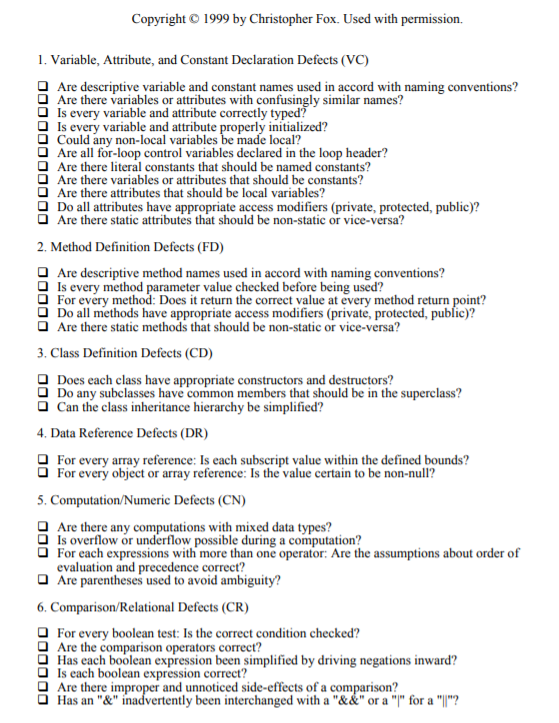
**Appendix. Log form used during Meeting Stage**

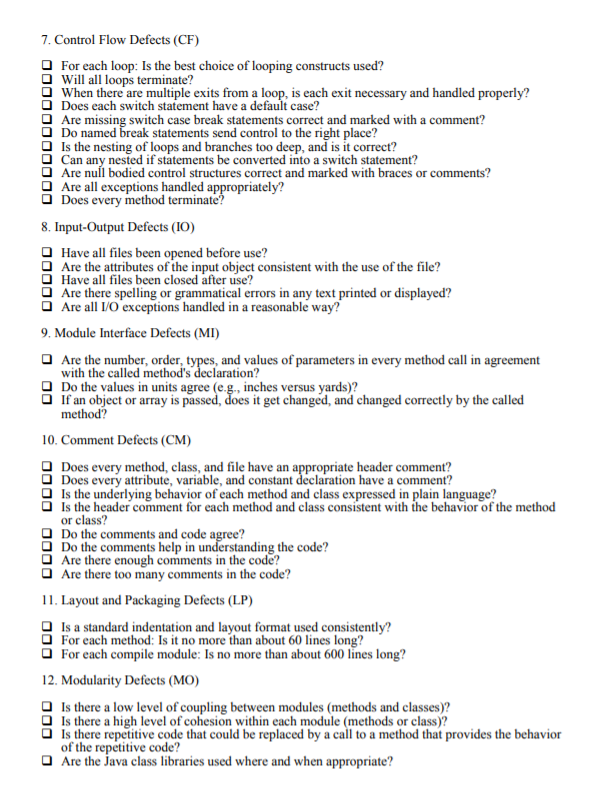
****

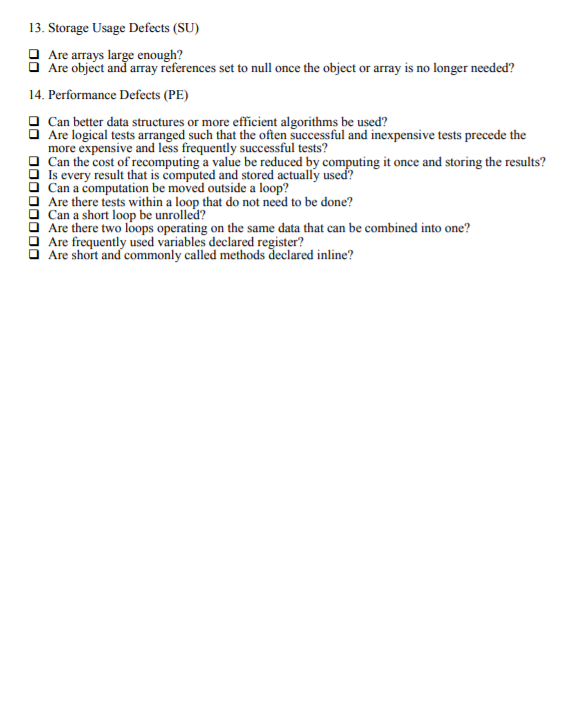
**Appendix. Inspection Issue Log**

****

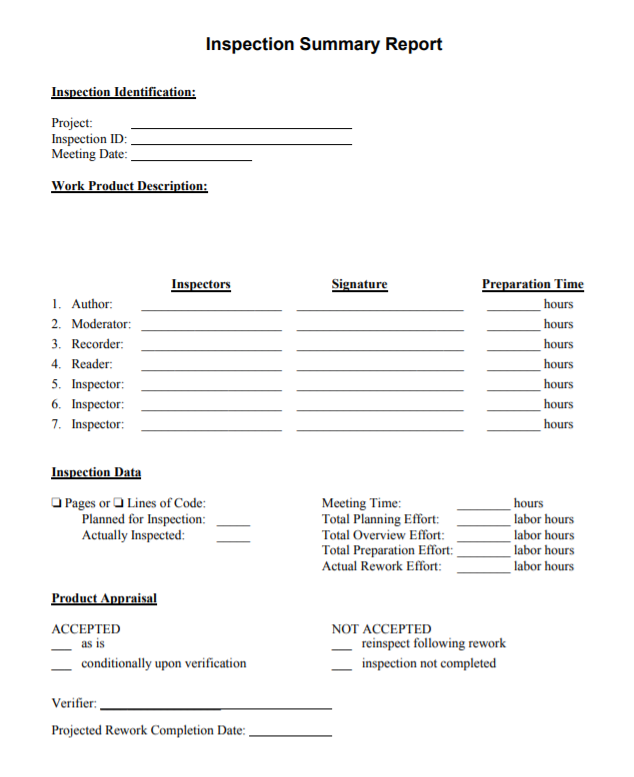
**Appendix. Inspection Checklist**

****

****

****

**Appendix. Inspection Summary Report**

****

**References**

Copyright © 1999 by Christopher Fox from Toronto University, restricted from: <http://www.cs.toronto.edu/~sme/CSC444F/handouts/java_checklist.pdf>

University of Victoria, Code Inspection Slide, restricted from: https://www.ece.uvic.ca/~shsaad/seng426/resources/Lab%20Slides/Lab2-SENG09.pdf