# Software Development Tools and Practices PUSL2020

# Coursework 2022–2023

Term: Term 2

Submission Deadline: TBA

Coursework Type: Group Assignment

Element of Assessment: C1

Module Leader: Dr. Rasika Ranaweera | Ms. Pavithra Subhashini

E-mail: ranaweera.r@nsbm.lk | pavithras@nsbm.ac.lk

# 1 Coursework

This coursework for PUSL2020 is worth 50% of the module marks. It is made up of two parts (i) a software product and (ii) a report containing your testing strategies and implementation. Students must create their own groups, up to a maximum of six group members. Each group must then decide and agree upon which technology they will use as well as specifying and allocating individual roles, and responsibilities.

# 2 Project Scenario

Colombo Municipal Council (CMC) is planning to launch a web site to optimize garbage collection, where volunteers within a district can report thrown away garbage and garbage trucks can collect them.

- The registered members of the Green Task Force (GTF) understood here as the volunteers, can take picture(s) of thrown away garbage, explain the impact such as attracting wildlife, terrible smell, etc., set the location, and report this incident.
- CMC has a dedicated Green Captain, a manager who can view all the reported incidents (as shown in Figure 1), validate by looking at the uploaded photo(s), and approve or reject stating how soon the garbage should be cleared.
- Garbage collection staff can see the approved reports with a summary on a Map along with the importance.
- General public can see the garbage collection spots on a map and news articles posted by the CMS.

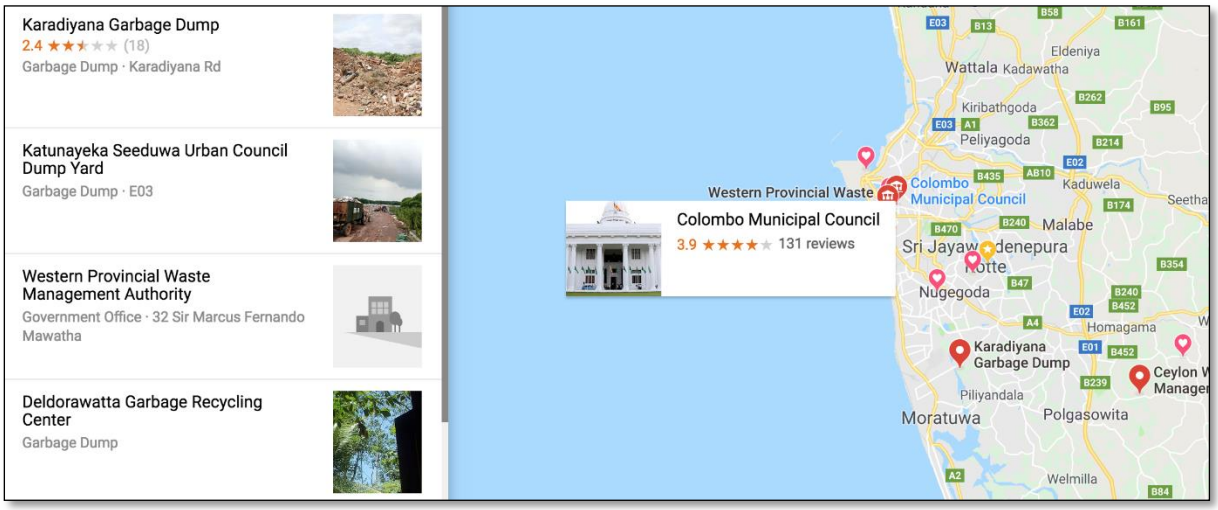


Figure 1. Sample GUI where the captain can see all the reports as a list and a map.

# 3 Functional Requirements

Three types of users are to be supported: GTF Members, Green Captain and a Web Master (administrator).

- **GTF Members**
  - Should be able to register and sign up for free.
  - Once registered, GTF members can input data (including images), delete and update his/her own incidents.
- **Green Captain**
  - Can see the reported incidents as a list and on a map
  - Should be able to select an individual incident to view all the accident details (similar interface like Figure 1).
  - Should be able to approve or reject a selected incident.
  - Should be able to set a flag based on importance such as red flag for immediate cleanup.
- **Web Master (Admin)**
  - Should be able to create accounts for the captains and collecting staff.
  - Can also post articles to improve public awareness about garbage collection and add garbage spots.

# 4 Deliverables

There are two deliverables, and both should be submitted via DLE.

## 4.1 Deliverable 1 [Development]

Create a web application using PHP or ASP.NET C# with MySQL/Oracle/SQL Server database or similar web technologies to demonstrate the above scenario while adhering to the given functional requirements.

## 4.2 Deliverable 2 [Test and Report]

Write a comprehensive report comprising 2000 words explaining your testing strategies. The report should include:

- 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 tests and the integration tests
  - Your functional test plans
  - A critical analysis of your test strategy, explaining why you have tested the software in the way that you have.

## 5 Detailed Assessment Criteria

| Category and Weighting | Criteria | Marks | LOs |
|---|---|---|---|
| Software Development **50%** | • Application should satisfy all functional requirements. | 40 | LO1 |
| | • Use of validations, pagination, and proper security measures. | 20 | |
| | • Application integrated appropriately with the database. | 10 | |
| | • Source code runs with test data. Data can be viewed, inserted, edited, and deleted from the database. | 30 | |
| Testing and Reporting **50%** | • Style of report is clear, professional and has a logically developed thread of argument throughout. | 20 | LO2 |
| | • Unit tests and integration tests. | 30 | |
| | • Functional testing. | 20 | LO3 |
| | • Test and validation metrics (e.g., code coverage). | 10 | |
| | • Use of tools and techniques. | 10 | LO4 |
| | • One-page (per member) report explaining the individual contribution to the development and testing | 10 | |

## 6 Academic Offences

*See below for excerpt of Section AST10.2. Full details on the academic offences framework and procedures, consult Section AST10 is available here.*

Academic offences occur when activity is undertaken which could confer an unfair advantage to any candidate(s) in assessment. The University recognises the following (including any attempt to carry out the actions described) as academic offences, regardless of intent:

a) Copying or paraphrasing of other people's work or ideas into a submitted assessment without full acknowledgement (plagiarism).
b) Unauthorised collaboration of students (or others) in a piece of work (collusion).
c) Making false declarations in an attempt to obtain either modified assessment provisions or special consideration (e.g. of extenuating circumstances).
d) Persuading another member of the University or partner institution (student, staff, or other) to participate in any way in actions which would be in breach of these regulations.
e) Misrepresenting research outcomes and results.
f) Being party to any arrangement which would constitute a breach of these regulations.
g) The inclusion in a piece of assessed work (other than an examination or test) of material which is identical or substantially similar to material which has already been submitted for any other assessment within the University.
h) Any other activity which could confer an unfair advantage to any candidate(s).

## 7 Submission Type

The report should be a PDF (preferred) or Word document. A zip file containing project work and other assets should be uploaded to the Plymouth Digital Learning Environment (DLE).