**CS2S567 Professional Practice and Employability:**

**Team Based Software Development Workshop**

**Portfolio 2**

**A Quality Assurance (QA) Portfolio**

Detailing own role in the team effort

**By**

*Lois Grimstead, 16095103*

|  |  |
| --- | --- |
| Team Number | **5** |
| Team Task | Recruitment Team |
| Own role(s) and tasks in Team | Role:  Tasks: |

**Table of Content**

[QA Activity Overview Log 2](#_3znysh7)

[Weeks 1 & 2 3](#_30j0zll)

[Weeks 2 & 3 4](#_2et92p0)

[Appendix: Additional Evidence 5](#_1fob9te)

**Frequently Asked Question: What is Quality Assurance (QA)?**

'Quality Assurance' comprises all activities that are necessary in order to produce a product of the highest possible standard or as Wikipedia[[1]](#footnote-0) defines it:

"*...is a way of preventing mistakes or defects in manufactured products and avoiding problems when delivering solutions or services to customers; which* [*ISO 9000*](https://en.wikipedia.org/wiki/ISO_9000) *defines as "part of* [*quality management*](https://en.wikipedia.org/wiki/Quality_management) *focused on providing confidence that quality requirements will be fulfilled*"

For the production of software this means:

Preparation work

* making sure that the program requirements are clear and well defined (PRD)
* making sure that project staff have all the required knowledge in terms of standards, processes, policies, procedures, programming know-how,
* Agreeing on backup procedures

So, in this portfolio you need to show that you have considered this, researched the 'how-to' and have taken action to fill any gaps in your knowledge, etc.

Programming work

* There are programming techniques such as 'Extreme programming' which address activities such as good communications, pair programming and continuous testing.
* Making backups (and verifying they are indeed ok)
* Verification and validation activities[[2]](#footnote-1) such as unit testing , static testing, dynamic testing, regression testing, integration testing, etc. And all of those in black box and/or white box mode. Not all of these apply of course, it depends.

So in this portfolio you need to show that you have considered this (e.g. do background research on which tests to do and why) and taken action (e.g. done testing and reacted to the outcome). See lecture slides of 'Secure Software Development' weeks 17, 18 and 19 for this.

Rolling it out work

* How do you securely deploy your software
* How do you plan the software maintenance and updating
* What information does the user need (manuals, training, preparation for change-over to the software,...)

So in this portfolio you need to show that you have considered this (e.g. read-up on it) and taken action (e.g. contact users, produce a training manual). See 'Secure Deployment' lecture slides of 'Secure Software Development'.

**How to report**

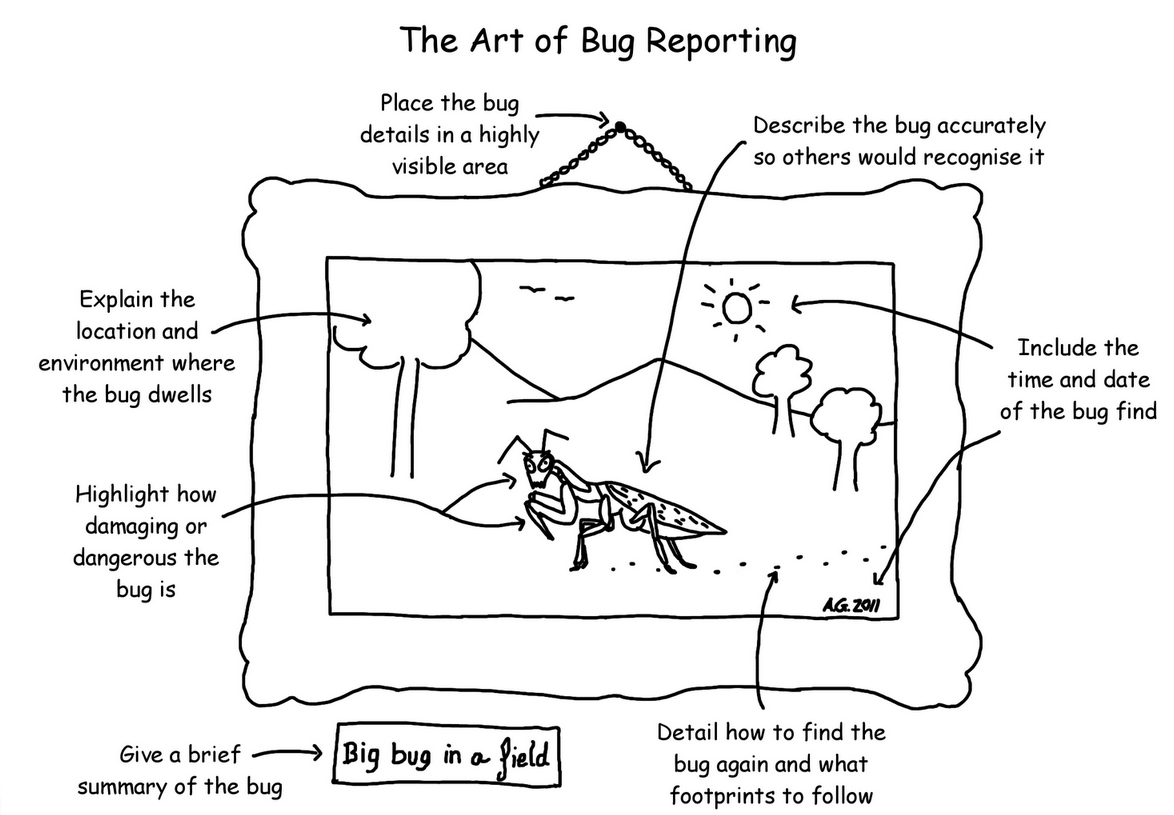
Use the table structure below. Fortnightly reports, i.e. one table every 2 weeks are sufficient. The blue 'Team' items can and should be completed jointly, the green 'Own' column you obviously do on your own.

Each team member should report on 6 consecutive weeks only, i.e. 3 of the tables below for each individual. Since a typical team is usually made up of 4 members the team could divide the reporting up like this: Team member 1: weeks 1 to 6, Team member 2: weeks 7 to 12, Team member 3: weeks 13 to 18, Team member 4: weeks 19 to 24. (Team with fewer members just have a few unreported weeks.)

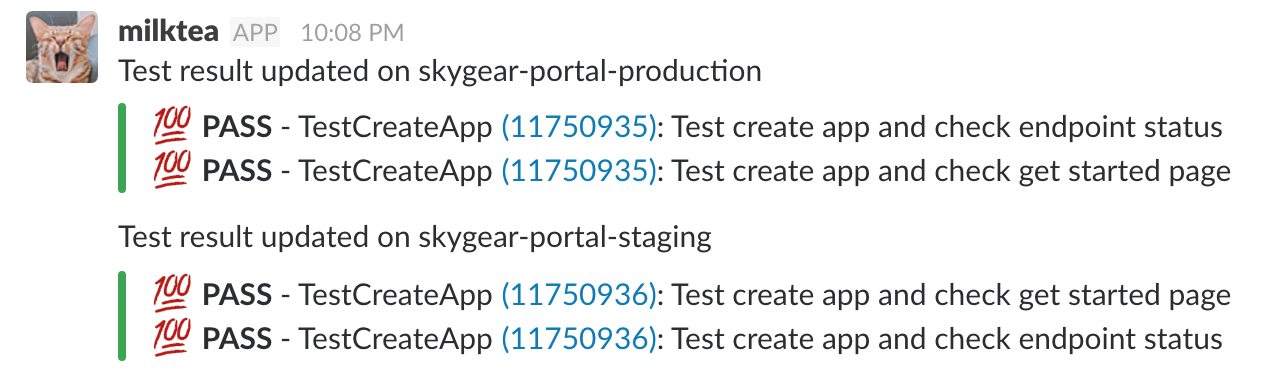
|  |  |  |
| --- | --- | --- |
| Weeks 1 & 2 | **Project Setup** | |
|  | **Team** | **Own** |
| **Activities undertaken** |  |  |
| **Outcome of QA activities** |  |  |
| **Activities resulting from QA process** |  |  |
| **Personal QA development** |  | |
| **Evidence of own activities** |  | |

# Any larger pieces of information that don't conveniently fit into the tables is placed in the Appendix.

# Appendix: Additional Evidence

****

**Item 1:** Methodology applied for bug reporting to team, Andy Glover ©, Blogspot.com

**Item 2:** Example of GitLab CI testing report for the 'start and stop' modules

1. https://en.wikipedia.org/wiki/Quality\_assurance [↑](#footnote-ref-0)
2. Verification = Are we building the product right? Validation=Are we building the right product? [↑](#footnote-ref-1)