

## Top Right Corner

### Updates to assessment 1 deliverables

#### Risk assessment changes

(Note the risk numbers given below refer to the new risk numbers after changes were made).

As we frequently forgot to push the latest versions of our code to git we decided to raise the likelihood of that risk 5 from improbable to probable. We kept the severity marginal as it never caused too many problems. All we had to do was message another group member and ask them to commit their changes, which only took a few minutes.

We raised the severity of risk 4 (Given that we have to take on another team's project, then there is a concern that (possibly) we may pick a poor project) from critical to catastrophic as now that we have thought more about assessment 3 and the short period of time we have to implement the required changes we've realised that if the system we inherit is badly written, buggy and poorly tested it will take us far longer and require a lot more work on our part to correct it. We still believe that is an improbable event as we are going to make sure we thoroughly review all products before making our decision.

This risk was struck of our list, as everyone that is planning on doing a year in industry has their placement sorted so it is no longer an issue.

11	Given that many of our team are planning to do a year in industry, then there is concern that (possibly) interviews may clash with group meetings or SEPR lectures and practical sessions	Probable	Marginal	We have agreed to catch up on any missed lectures ASAP and always upload a copy of the notes from any meetings we have to our shared folder on Google Drive so that team members can catch up on work that they would have completed during that meeting and can see what we discussed.
----	---	----------	----------	---

One of the comments made in the feedback we received on assessment 1 was: "Ideally, each risk should be assigned to / owned by specific team members (e.g. project manager, customer interface)". We decided against doing this in the first place as we are following a Scrum-like agile methodology and we don't have specific roles other than a scrum master, so it doesn't make much sense for any one individual to take ownership of an individual risk.

After reviewing the feedback we decided that we agreed with the marker that the categories we chose were perhaps not sensible and decided to change the categories used for likelihood.

from: frequent (over a 74% chance of occurring), probable (40%-74% chance of occurrence) and improbable (up to a 39% chance of occurrence).

To: frequent (over a 74% chance of occurring), probable (30%-74% chance of occurrence) and improbable (up to a 29% chance of occurrence). Afterwards we reviewed each risk and decide that no changes to the likelihood were needed.

We also renamed the categories chosen for severity to: critical (over 6 hours worth of work), considerable (between 1.5 and 6 hours of work), marginal (up to 1.5 hours of work). As the marker was right in stating that over 6 hours of work should not be considered catastrophic.

As per the feedback we coloured each of our risks according to the type of risk: Project Risks are coloured red, Product Risks are coloured blue while Business Risks are coloured green.

Updated version: <https://sepr-topright.github.io/SEPR/documentation/assessment2/updatedriskassessment.pdf>

### Requirements updates

Our requirements have not changed from the initial requirements we drew out. There were many aspects to consider when deciding whether the requirements need to be changed. Our requirements were accepted by the client and this led us to think it best not to change any of the requirements. Our client didn't request for us to make any changes to the requirements and to make changes after this point would have led to many complications. This is because it would cause a great deal of confusion and waste a lot of time if we were to alter requirements after already having agreed them with our client. We also found that there were no reasons to change the requirements when it came to implementing them. This was due to there being no complications with the original requirements being implemented.

### Updates to the methods and planning document

In the “Software engineering methods, development and collaboration tools” section of the document we added links to all tools and libraries used as requested. We also mentioned that we used JMockit for mocking out dependencies when unit testing and gradle to handle build automation.

No major changes were made to the plans as we feel that the plan for assessments 3 and four are still sensible and contain enough detail and for the most part we stuck to our assessment 2 plan and didn’t feel the need to update it. The only changes that were made were that we added an extra week to the assessment 2 plan during the christmas period as when creating the plan we thought that the christmas break was four weeks, not five. Although this had no effect on any of our self-imposed deadlines. We also provided links to larger versions of the plans on our website as requested.

Updated version:

<file:///home/josh/website/SEPR/documentation/assessment2/updatedMethod.pdf>