

Module	SEPR
Year	2019/20
Assessment	2
Team	Dalai Java
Members	Jack Kershaw, Max Lloyd, James Hau, Yuqing Gong, William Marr, Peter Clark.
Deliverable	Updates on Assessment 1 Deliverables

Updates to requirements

For Assessment 2, the section on Requirements from Assessment 1 needed to be updated and can be found below here alongside a changelog.

https://baffledwhiskey.github.io/Updated_Assessment_1_Deliverables.html#item-1

To determine which areas needed to be updated or changed, we had a group meeting where we reviewed the documentation for Assessment 1 and directly compared it to the specifications for the program. We then reviewed the briefing to determine what features were present in Assessment 2 that were not present in the current state of requirements. To determine specifically what requirements needed to be added we looked at the specification for the program and used abstraction to create a list of features which were then adapted into functional requirements.

We added two references to specific subsections in the IEEE Requirements Specification, being 4.1.1 and 4.1.3. This is to allow for a direct comparison to the set of guidelines that were followed while creating the tables for User Requirements, Functional Requirements and Non-Functional Requirements. It also explains the rationale behind the formatting of the provided document, for example in the formatting of each unique identifier based on what table of requirements it lies in.

We also removed any ambiguous terms from the tables of requirements as there would be a struggle to implement a quantitative scale for each term. This can be shown in the changes to FR_CHANGE_GAME_MODES, NFR_GAME_TIME and the removal of ambiguous terms from their descriptions. This was to stop there being subjective terms present in the exact specification of requirements and leaves less room for doubt.

We needed to add some more functional requirements into the table as we had noticed that there were core features of the game that had not been defined in the requirements document in Assessment 1. The updated document sees the addition of UR_DEVICE, FR_ET_RESPAWN_TIME, FR_NUM_OF_ET, FR_NUM_OF_FORTRESS, FR_ET_SPEC_SPEED, FR_FORTRESS_IN_RANGE and FR_ENGINE_IN_RANGE.

The functional requirements FR_NUM_OF_ET, FR_ET_SPEC_SPEED, FR_ET_RESPAWN_TIME, FR_FORTRESS_IN_RANGE and FR_ENGINE_IN_RANGE have been added to the updated table of requirements as they were missing core features that would stop the system providing a logical and cohesive experience.

The User Requirement UR_DEVICE is a feature that we decided should be implemented in the future to allow a greater audience access to KROY, thus including an increased amount of perspective students.

Updates to methods and planning

To find the updated Methods document and its corresponding changelog please follow this link:

https://baffledwhiskey.github.io/Updated_Assessment_1_Deliverables.html#item-2

In our meeting we discussed all Assessment 1 deliverables and the changes we needed to make.

Most importantly we decided that our team structure needed to be changed considerably as the flat structure that we started with led to issues with team organisation. Moreover, we thought that the original document should be edited based upon the constructive feedback we were given after Assessment 1.

We assigned abstract roles to each team member as we thought this would allow us to be more flexible with individual task assignments. Specifically the roles were:- Jack:PM, Peter:QA, Will:Art, James:Art/Dev, Max:Dev/WebDev, Yuqing:Dev. Selection of roles was based on aptitude; for example Jack was made PM as he's showed solid leadership throughout, and Peter is in charge of QA as he had experience in JUnit tests etc.

The Assessment 1 document was edited according to feedback. The two redundant images were removed as they added nothing to the report. Contradictory material was also removed; the portion about RUP was attempting to explain the structure of the project as it has been given to us, i.e. how the assessments themselves seemed to follow RUP. However, this was explained poorly and, in hindsight, needs not mentioning.

When we initially chose which programs to use with our project we discussed extensively which organisational/planning software to use. However, we omitted this from the first report in Assessment 1. A small justification summing up why we chose Trello over alternatives has been added. The decision came down to Trello's simplicity, we felt like we didn't need any complicated features offered by other tools as this would distract from the important part of the tool, the tasks which we needed to complete.

Furthermore, we added a section about proportional task delegation in place of the justification we submitted in Assessment 1. Having reviewed what was written for that assessment it was obvious that the wording wasn't clear. The intended message was to justify adding more team members to a difficult task early on rather than late into the project. Advantages to this are clear; the cost of adding more developers to a task and bringing them up to speed will likely outweigh its benefits.

The plans for Assessment 2 were redundant thus were removed from the document. The gantt chart for this section was enlarged and the page rotated so it is readable, this wasn't done in the first document as there was very limited space. In addition, plans were added for Assessment 3. They highlight our approach to 'buying' another team's code, for example choosing well maintained code is as important as choosing code with major progress towards the final project. On top of this, we added that we could work towards features not stipulated by the clients as long as all core requirements had been met, and as long as the additional features did not stray from the intended design. Any features that fall under this category would be posed to the clients first for their approval as we wish to add to the product and not add anything classified as unnecessary by the customer.

Updates to Risk assessment and mitigation

In our group meeting to discuss the appropriate changes and alterations to the deliverables from Assessment 1, we decided that for our Risk Assessment, we did not want to change the structure of how the risks were laid out. We believed it to be a very logical approach to categorising the possible risks that we would encounter in our development lifecycle. We did update the document for the changing of needs which can be found here:

https://baffledwhiskey.github.io/Updated_Assessment_1_Deliverables.html#item-3. Each team member reviewed the risks they were assigned, determining whether the significance, impact and mitigation strategies were still appropriate.

However we did want to change some of the risks themselves as we determined them to be inappropriate in the current scenario and too far-fetched. This namely includes the removal of the Business Risk R3.1 - "Java may become obsolete" and R3.2 - "Government policy may lead to our game not being legal". While they were concerns, we decided that in a real-life, everyday scenario they are not things to be actively concerned about as the likelihood of it occurring is so low, it would be inefficient to put in place mitigation.

Furthermore we changed R3.1 to be "Ideal game specifications were not researched properly, leading to poor reception of the game". We deemed this an appropriate and realistic risk as the features the players want in the game will be an ever changing ideal and we need to do our best to accommodate for it. Our proposed mitigation would be to observe how the needs of the players changes using customer surveys so we can better develop the game.

We changed R3.2 to "May need to upscale services based on number of players". After discussion and review of our requirements table, we included in the Requirements document, the User Requirement UR_COMPARE_SCORE. This will mean that various player data needs to be stored with all devices having limited access to this data. If enough players were active on the game, there is a possibility of us needing to upgrade our data storage, which could lead to corruption of data, potential loss of user data and game downtime, all leading to a loss in players. To mitigate this we decided to try to have appropriate technology in place to keep downtime to a minimum and keep regular backups of data.

Having user data being stored lead to the addition of R3.4, "Servers may be hacked leading to loss of customer data". Having a breach of customer data would not only put us to be liable for legal action against us, but also a loss of players from the game. Therefore we want to put the appropriate security measures in place to make sure we are aware of any potential attacks but can also stop it from happening using encryption and firewalls.

We also added a number of new technical risks which we have encountered whilst developing the game. We have added R1.7 "Unit tests may not correctly test the functionality of classes" and R1.8 "Unit tests may not produce complete code coverage" in order to acknowledge that a working unit test does not necessarily equate to a working program. We have also added R1.9 "There may not be code available to implement the desired features". This relates to the fact that if we do not have a back-up plan, our game may be missing vital features. In terms of project risks, we changed the frequency of R2.5 "Set deadlines may not be met" to high, as this has been a recurring issue whilst creating our project, and it is important that we are constantly taking action to mitigate this risk.