

# Change Report

Group 26

devCharles

Ross Holmes  
Sabid Hossain  
Thomas Pauline  
Joel Paxman  
Andrey Samoilov  
Louis Warren

# Introduction

First of all we had to decide which team we wanted to choose to complete assessment 2. We immediately narrowed it down to three teams based on their presentation and game architecture. These were:

- Team 28 - (OuseWorks)
  - used Entity Component Systems (as we did) making it much easier for us
- Team 30 - (Triple Ten)
  - Seemed to showcase a good game

However after looking at both teams we thought their code was poorly maintained and documented and therefore we didn't choose them.

- Team 32 - (Team 32)
  - Team 32 was chosen as their documentation was okay but more importantly their code was well commented and fairly well structured.

After picking Team 32 we collected all the documents from team 32's website and stored them on our google docs. We also created a working relationship with them so that we could clarify any queries that we may have. We then had a group meeting in which we looked over the documents and created comments on what needed to be changed. This meant when the documents were revised the comments could be used as notes for what should and should not be changed.

We also created a second copy of these documents, which we then edited until we believed each deliverable was now relevant to assessment 2. These documents were all stored on google docs as:

- Each copy can be safely stored on the cloud (e.g. multiple back ups)
- Any member of the team has easy access to each document
- All edited documents contain a complete version history, allowing the team to have detailed information on what was changed.

The code base's evolution was tracked entirely through Github using its version control system, so that we could:

- See where new features were added
- What old code was refactored/removed

Alongside this method of tracking development we had a change management process in which issues would be created on github and then solved later by team members. This means that if a team member desires to create an issue it could be discussed within the team, enabling less issues to be created with clear priorities and trackable issues.

# Requirements

[Req1](#), [Req2](#)

The requirements documentation was drastically changed in both the introduction, the quantity, specifics of requirements and the formatting of the document for reasons explained below:

First, we decided to look at the requirements already created by Team 32 and we quickly realised that several of their requirements had been interpreted differently or their group asked different things about their game; therefore they had a few discrepancies that needed to be changed.

Due to this, we changed their requirements slightly so that they became in line with our own answers from the client. Such as:

UR_BRANDING	The level should be consistent with the Piazza building and contain UoY branding	Should
-------------	--	--------

Which was completely removed since our customer did not require or want this to be in the game.

Several requirements were reworded too such as:

NFR_USABILITY	The system shall contain no technical jargon	UR_INSTRUCTION S	Anyone from ages 5 and above should be able to understand and play this game
---------------	--	------------------	--

Which was changed from:  
“Not use any complicated terminology”

This specifically was changed due to the fact that this requirement: “Not use any complicated terminology” is effectively a repeat of “system shall contain no technical jargon” and we believe our new fit criteria more successfully describes the goal of NFR\_USABILITY.

Several other fit criteria were changed too including:

Changing NFR\_AVAILABILITY from:

NFR_AVAILABILITY	The system shall be highly available	UR_CONTROL_CH EFS	Uptime: 100% during the open days
------------------	--------------------------------------	-------------------	-----------------------------------

To:

NFR_AVAILABILITY	The system shall be highly stable	UR_CONTROL_COOKS	Uptime: 99.5% during the open days
------------------	-----------------------------------	------------------	------------------------------------

This was since the fit criteria was too unrealistic, so our user requirement was changed and the description was also changed to say stable rather than available.

During the process of this, we determined that many user requirements defined by Team 32 were closer to being functional requirements, and so we merged some of them together as new user requirements to better fit. This made them less specific and therefore more like user requirements, but therefore also required us to change what functional requirements mapped to which user requirements.

For example UR\_COOK\_FOOD was a newly created User requirement using both UR\_COOK\_FOOD and UR\_SERVE\_FOOD from the previous team and also added the additional requirements such as the additional foods

Shown here:

UR_COOK_FOOD	The user shall be able to serve: pizza, jacket potatoes, burger, and salad, as well as collect ingredients for those recipes.	Shall
UR_COOK_FOOD	The player shall be able to make salad and burgers	Shall
UR_SERVE_FOOD	The player shall be able to serve salad and burgers to customers	Shall

Once we had changed much of the requirements we then reformatted their tables, changing their appearance and re-wording them according to the [Easy Approach to Requirements Syntax](#). This was all so that the document followed the standards identified by the industry and also improved feedback from assessment 1.

By this point our requirements documentation have become rather distinct from the original documents, both ours and Team 32. We chose to build upon Team 32's documents, whilst bringing in our own best points and information relevant to our team, using the feedback from assessment 1.

Once this had been completed we then looked at their introduction to their requirements but, since Team 32 had taken a large period of time describing how they elicited their requirement and how they formatted them, we needed to change them. This was because we had reformatted the document slightly to improve upon the document based on feedback given by the module leaders to Team 32 and we also had to change the introduction since we had started with a lot of requirements already, so our process of creating the document differed greatly by now.

The changes to the introduction of requirements was as such:

- Team 32's first paragraph was mainly about the client meeting and questions posed to stakeholders of the game, however we didn't speak to - or have a meeting with - anyone other than our client (the main stakeholder) so we removed their paragraph since it had no relation to what we have done and the client responses we got.
- The second paragraph was describing how they formatted their document, since we also changed this we then had to remove that paragraph too.
- Once this had been removed since it was all now irrelevant, we then described how we formatted our document and why and how we elicited new requirements from assessment 2.

To elicit the new requirements for assessment 2 we had a team meeting as described in the document. Once all of these new requirements were created we then had to append them into the correct tables within the

document, which is shown in the requirements document as “appended”. This was so that the document contained all the requirements relevant to assessment 2 such that it fulfilled all criteria for our assessment.

FR_DIFFICULTY	When starting the game the user shall be able to choose easy, normal or hard difficulty. The higher the difficulty, the higher the frequency of grouped customers.	UR_SETTINGS, UR_MODES
---------------	---	-----------------------

This is an example of a functional requirement we added, as in the first assessment there was no need for variable difficulties because there was only one fixed scenario. In assessment 2 however, the product brief specified a requirement to give the option for: easy, medium, or hard difficulties (as well as the option for endless mode). This meant that we had to amend the requirements document to display this change.

# Architecture

## [Arch1](#), [Arch2](#)

We chose Team 32 specifically for the code they had made, as we believe that implementing the new requirements would be the hardest part of the project. Therefore, we chose their code as it was the most well designed, documented and organised of the groups in our cohort. Additionally their architecture document progresses in a timeline-like manner, which means that any changes to the architecture session will merely be additions. The document clearly displays how the group progressed and developed their code overtime in response to the requirements, and thus our changes to this section are additional sections to detail any changes we were forced to make to their existing architecture in response to the new requirements of assessment 2. This way, the architecture report is a simple flow of Team 32's original plans of an architecture, through as they began to code and restructure in order to meet requirements, before concluding with our own changes for assessment 2.

This meant we needed to create additional graphs since the game's design had evolved since Team 32 had made their graph and these graphs were added to the relevant section of the document. We mainly chose to include graphs that represent the changes we made to meet the new requirements, those being our introduction of saving and an overview of the new structure of Customers. Additionally, we chose to include new sequence diagrams to represent any complex sections such as saving. These diagrams are figures 10-13.

Section B (describing how it evolved over time) was kept the same as mentioned, then similarly to section A it was expanded on, explaining our thought processes behind what we changed and how the game continued to evolve until completion. This forms a continuous timeline from initial idea creation and their methodology in selecting the file structure that they did all through until our additions and restructuring for the assessment 2 requirements.

## Method Selection and Planning: [Plan1](#), [Plan2](#)

Because our team and Team 32 used very similar tools, programs, resources and communication mediums during the project, whilst we reformatted a lot of it and re-explained parts of section A, the underlying part of it is Team 32's.

The structure of the document was changed completely, from a paragraph type document to a more bullet point type snappier document; This was changed due to the fact that this is asked for by the assessment criteria and is significantly more readable.

Whilst the team also had mentioned Github, we used substantially more features and tools from GitHub such as the projects tool and the Continuous Integration, meaning we needed to add more details about these into the section.

We added information about Visual Studio Code since this was one of the main IDE's we used alongside IntelliJ, as we made a large amount of use of the Live Share extension available on Visual Studio Code.

Google Drive was our source of file sharing and we heavily relied on it, particularly with report creation, so we spent a lot of time considering competitors such as OneDrive. This meant that we chose to expand on what was said by Team 32, to properly display our consideration of the benefits and drawbacks of each service and why we ultimately settled on Google Drive.

The last section of part A was kept and minorly adjusted based on how the team felt, to make it more succinct and improve clarity.

In section B a lot was changed content-wise and structurally, this is because we used a different system to organise the team and therefore also had different reasons for the way the team was organised. Along with this, we had to make the section have shorter paragraphs and add bullet points to make it clearer to read and understand, as was asked of us in the assessment criteria.

The second paragraph was split into bullet points explaining key points of significant note and removing useless wording. Along with this, we added additional detail to points and explained why some team members had two roles.

The table was updated to include our names, update the roles of each team member and included developer roles which is all necessary since it previously had Team 32's members in the table.

From then on, the report completely changed since we organised the team differently. We added that all members of the group will be involved in implementation or testing to some degree, explained the reasoning behind developer roles in the table and explained how and why the team is organised the way it is.

The last section of Method Selection and Planning is completely changed due to the fact that we are a different team, working on a new set of deliverables, starting from a different point and it's now a completely different time frame so it's simply impossible to use any of what was written by the previous team.

## Risk Assessment and Mitigation

## [Risk1](#), [Risk2](#)

The introduction of risk assessment was completely changed due to the fact that it spoke mainly about how the previous team collected risks and how to mitigate them, with a very small focus on how it was formatted; Whereas we already had a significant quantity of risks and therefore our approach in creating the document was vastly different.

Firstly, within the introduction we discussed the process of how we added risks to the document and what procedures we followed to achieve this as this is how our group functioned differently and therefore needed to be added. We made sure it was only a small focus on the introduction as we had a significant amount of things to mention. With this, we also discussed how we would have a mitigation plan but this was mainly carried over from Team 32 since our strategies were similar.

Within the introduction we spent a great deal of time describing how we formatted the risks, which we decided to add (and changed from Team 32) since we believed the way we tabulated the risks was clearer and needed to be explained than the prior method of display. This is why we put a great deal of effort into explaining why it is tabulated the way it is.

We then explained what the owner was and added a few details as to why there may be multiple owners for one task. There was no aspect of this present in the previous team's original documents, and thus it was a new addition to the document.

### Risks:

Within the table, status was removed and instead a small paragraph at the end was added to explain any risks that did occur and what was done in those cases, this was since our team thought it would be simpler and easier to understand.

We colour-coded severity and likelihood to show their importance immediately to any viewer, which much improved their readability.

Each risk was reassigned a risk ID in chronological order, therefore simplifying the process of having a risk ID and working out which one it is within the table.

Our team also spent a significant amount of time identifying new risks for assessment 2, which were all appended to the document, and alongside this a new type of risk was identified (business risks) and so a new table was created for business risks.

Alongside this several risks did occur (R8 and R1), resulting in two small paragraphs being added to the document explaining what had happened and how we planned to prevent them from reoccurring.

A third significantly worse risk occurred (R9). However, thanks to some pre-planning and safeguards we managed to avoid it. This was explained and added to the document, with the mitigation strategy being changed and improved to make it significantly less impactful if it were to happen again.

Finally we had a second look into what Team 32's risks were and one was changed (R17) giving it a different risk description and improving its mitigation plan.