

Change Report

2A

Upon inheriting the previous team's work, the first thing we did was place the inherited artefacts into appropriate version control repositories. We used Google Drive for the documentation, and created a new GitHub repository for the code and CI infrastructure. We chose these since we were all familiar with them from the previous assessment, and both perform their respective tasks sufficiently well, with built in tools fit for our purpose particularly with reference to change management.

The previous assessment was immensely useful to us in learning which collaborative tools and methods worked best for us as a team, and so moving on from this we have used this insight to best keep track of the changes we have made. One tool we have decided to use again is Kanbanflow, while we still use it to keep track of tasks, we created an additional board to keep track of the changes that we believe must be made, and to mark them as made once completed. The board is set up similarly to our tasks board, with unmade potential changes on the left, currently ongoing changes in the middle and completed changes on the right. This has made it a lot easier to update this document, by referencing the completed changes periodically, we have been able to continually add them to this document so that no major change is left undocumented.

With regards to the product itself, keeping track of the changes has been simple and intuitive to do through the good practices we have adopted when making and committing changes to the code by making small, periodic pushes to github. This way, it is easy for any team member to see at a glance what the latest activity has been, before pulling from the repository to ensure consistency and a lack of bugs or contrasting versions. Referencing both commits to github and our Kanbanflow board means that we can be confident with the progress of our work and be sure that it is completed to the correct standard and on track with our scheduled plan.

It was important to review the changes we have made, and as such we followed an iterative approach to review changes to both the code and documentation. It was important to strike a balance, ensuring that while we regularly reviewed our changes, we did not spend too much time on them or carry out reviews too frequently, as this is costly in a project where time is a limiting factor. We decided to review the work of other team members rather than our own work, in order to reduce bias and ultimately get better results. It was important to keep on top of these reviews however, or else it could have had an adverse effect on the timings of the project, as previously mentioned.

Finally, it is important to briefly note that perhaps one of the best ways we have been able to manage changes is through our frequent meetings, both in person and over Discord, where we have discussed our progress. This way, all changes have been fed back to those responsible for maintaining this change report, meaning that the document has been continually and accurately updated without detracting from progress in the moment. This has also had the indirect effect of aiding with planning and time management.

2B

Requirements

Original URL: <https://western-criminals.github.io/Req1.pdf>

Updated URL:

One of the first major changes we made to the entire project was to update the requirements document, so that we could begin referencing this document in development, looking at the new requirements that have been added to the project. We paid attention to ensure that we followed the style that the previous team had set out as closely as possible, to produce a more coherent and readable final document. Where possible, we tried to change existing requirements, rather than creating new ones, for example with the 3rd chef (UR_CONTROL_CHEFS), new ingredients (UR_INGREDIENTS) and recipes. The new requirements we added were based on the new features the game should have, such as those relating to powerups. We added each new requirement, totalling 5 new user requirements and 10 new functional requirements, to the requirement table as a new entry following the same format. We gave these requirements appropriate names in the same style as the others. For ease of identification, we have italicised the new requirements to make our changes more visible, and previous requirements which have been altered have been marked with an asterisk, with the change in the description italicised.

We added these requirements based on the new features outlined in the brief. It was important to be descriptive of what must be achieved without being overly verbose, and as such we aimed to describe the requirement in around 1 or 2 sentences. This is in keeping with the tradition that the previous team set.

We didn't need to add any new non-functional requirements, as the scope and usage of the project has changed little between the assessments, so all requirements still hold.

It was important to us that the requirements document was updated as soon as possible, and to the highest standard, as it has been referenced throughout the second stage of development extensively. When coding, designing tests and discussing the architecture, the requirements have been invaluable to us in order to keep us on track and as such their completeness has been critical.

We have given more detail on our process and reasoning about updating this document in Part A of the requirements document, where we have added another paragraph.

Architecture

Original URL: <https://western-criminals.github.io/Arch1.pdf>

Updated URL:

We made lots of changes to this document, reflecting the large amount of changes we made to the code itself. One of the first changes you will see is changes to the architectural diagrams. Given that we inherited a very robust code base, many foundational elements of the architecture were already in place at the beginning of assessment 2, so changes to the diagrams are quite minimal, with the new classes created represented in the UML diagrams. We were fortunate that the previous team chose to use PlantUML to create their diagrams, which is what we also used previously. This previous experience meant that we were able to keep the architecture diagrams as consistent as possible. The main changes you can see relate to the completely new features, such as the new game modes, shopping system and power ups. Requirements that affect the behaviour of the chefs and recipes had little bearing on the overall architecture, and as such had little impact on changes to the architectural diagrams. Additionally, we added some new diagrams to give a more detailed look at the new features we added, which are the behavioural diagrams. These diagrams show how the player and the game interact with each other - what actions the user must do in order to solicit a particular result from the system. The new architectural diagrams which we have added can be found under the "Phase 2" subheading in part A, so that there is no ambiguity as to which diagrams were created by our team for the second phase of assessment. We have also added a brief description of our reasoning behind the diagrams and the methods and tools we used to create them.

The majority of the changes we made to this document were additions to Part B: the detailed descriptions of the architecture of the code. It was important to thoroughly update this document to maintain the standard that the previous team had set and to explain the decisions we made when implementing the new features. Some classes such as the chef, remained mostly the same, most changes took the form of changing parameters or adding additional methods - changes that we felt were not significant enough to note in the Architecture. For more significant changes: such as the creation of new classes like LongBoiBank which fulfil the new requirements, or classes like JacketPotato and BurningUI which build on existing requirements, we felt it important to note these changes in the architecture documentation. This is because the traceability of requirements and what we have done to meet them is very important to us. In order to make the changes we made to this document more visible, we continued on from the previous teams descriptions under a new heading "Phase 2", so there is no ambiguity as to which features we have added. Over the course of the assessment, we have systematically updated this document as new features have been added. Therefore, you will notice a paragraph for each new feature and change, describing the implementation of the feature and which requirements it relates to. By doing it this way, it was easy to keep on top of our changes. It was also much more efficient to update the document continually, rather than at the end of development. This also meant that the descriptions we wrote were concise yet accurate due the lack of time between finishing the code and writing the description. Another side effect of this practice is that the additions are listed in chronological order, making our development process easier to follow.

Method selection and planning

Original URL: <https://western-criminals.github.io/Plan1.pdf>

Updated URL:

Only minimal changes needed to be made to this document, as we largely used the same tools and followed the same methods as the previous team, for example using IntelliJ and GitHub for version control. Therefore, Part A remains virtually unchanged aside from a few additions regarding tools we used, such as Kanbanflow, which the other team did not.

One key change we made was updating the roles and strengths table that the previous team had created, to include our own names and respective roles, as well as retaining the previous information. We didn't want to alter the document regarding their plans and actions from each team member too much, thus erasing their credit, so we continued to add to the document rather than edit out what the previous team had already done. As mentioned in the document, we continued with the previous teams practise of creating gantt charts to represent our plan, which can be found on our website under the Gantt Charts section, just below the previous teams.

In each section, we have added additional information regarding our processes, since they are not the exact same as the previous teams. These additions can all be found under a "Phase 2" heading within each section. We didn't want to erase any of the previous teams information, so we felt appending ours would be more appropriate.

Risk assessment and mitigation

Original URL: <https://western-criminals.github.io/Risk1.pdf>

Updated URL:

We only made some brief changes to this document, such as altering the "Owner" column values to include members of our team, since the ownership of all code and documentation, and thus the risks which come with it, has been transferred to us. Additionally, we added a few other new risks to the register that have been introduced in the new assessment. Some are related to the new requirements and the constraints that are linked to them, and others are related to the nature of the assessment itself. Such as the risk about not understanding the other team's code, or having issues accessing the documentation. We have also added risks relating to the new testing and continuous integration portion of the assessment.

We didn't want to add too many additional risks to reflect that this is a continuation of the previous assessment, and thus most key risk factors and mitigation measures remain largely the same. Being a small scale project, dealing with non critical software and infrastructure, it is also unrealistic to produce an excessive risk register.

There are also some slight changes to the format of the document which are due to some formatting issues that arose when we converted the previous teams PDFs into Google Docs, though they are minimal and do not detract from any of the content.

In order to make the changes visible, we have italicised all entries in the risk register which we have added. They run from R_PROJECT_08 - R_PROJECT_10 and R_PRODUCT_04 - R_PRODUCT_08. We added 7 new risks in total. Additionally, we have altered some risks. Again, these changes have been italicised for ease of identification.