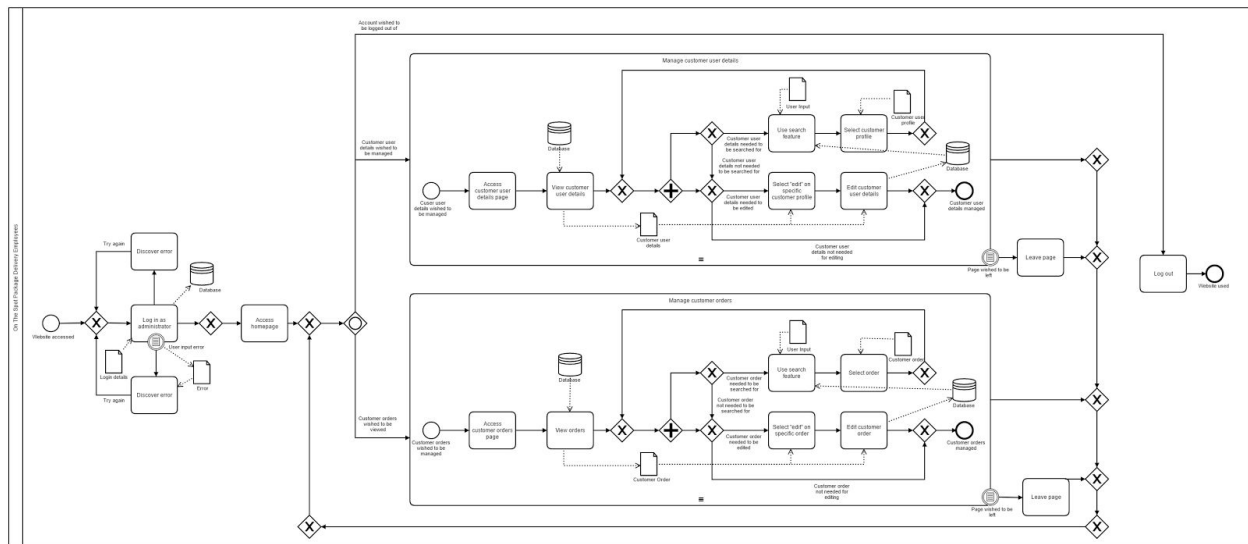


HQ link to the BPMN model of the customer perspective: <http://i.imgur.com/cKpErsO.png>

BPMN model of the employee perspective in the Release 2 perspective:



HQ link to the BPMN model of the employee perspective: <http://i.imgur.com/ZFT22R0.png>

Sprint 3 and 4 Reflection

A reflection of the team's performance in Sprint 3 and 4 has been performed. This reflection was performed to gain an understanding of how the team can perform better in future work. The reflection was performed by investigating the team's commitment to these Sprints. A reflection of Sprint 3 and 4 is especially necessary as these Sprints are content heavy and crucial as they lead to the completion of Release 2. The team's adherence to the SCRUM process will also be reflected on. The information generated from this analysis will be used to guide future work on the project.

Release 2 of the Project Delivery project was expected to be fully functional and encompass all user stories listed for the release. It expands on Release 1 by integrating more technical features on the fundamental items that were developed in Release 1. It also integrates the features that were intended to be included in Release 1 although not included. These features are very orientated around design, implementation, functionality and delivery and were very dependant on feedback generated from customer involvement and UAT.. The integration of these features were aligned with a working schedule and separated into two Sprints (3 and 4). The team referred to many artefacts -- such as project survey analysis, business models and client feedback -- through the project to assist in all areas of the project.

Sprint 3 was approached with caution for two main reasons. The team realized that the velocity for Release 1 was not realistic, as made evident by a significant user story being constantly delayed and an inconsistent working schedule. It was also realized that the working schedule would need major shifts after a team member effectively dropped out of the project. This was concerning as it left 3 members in a group against the expected minimum of 4 members per group. These issues were tackled by accelerating the velocity rate for Release 2 -- so that user stories could be completed within their delegated Sprint -- and adjusting a better working schedule -- in spite of a team member abruptly leaving the project. Although this plan effectively solved the issues at hand -- it created another. This issue was that the intensified velocity rate meant that more work would be placed on the shoulders of each team member. This made the team unsure if this workload was realistic, creating concerns that an unrealistic velocity rate would result in unfinished user stories. There was not much the team could do to control the issue -- besides adapting to the work schedule as closely as possible, and discussing with their tutor -- as it was crucial for all user stories to be implemented. In future, the team should develop backup plans for issues, like this, that can heavily shake the project. For example: in the event of a team member refusing to communicate for a considerable amount of time, the development team should take action as soon as felt appropriate, instead of extensively waiting for the unlikely chance of them ever responding. This manifests from the fact that extensively waiting for this team member caused many delays in the project development -- such as the database implementation phase -- as their designated work had to be spread across the existing team members at a very inconvenient time. As part of a backup plan, the client team should have been made aware of the issue so arrangements can be made to extend due dates and adjust expectations.

Sprint 3 and 4 introduced many new design challenges for the team. Unlike Sprint 1 and 2, Sprint 3 focused on the implementation of technical and complex features. Many concepts -- such as enabling administration privileges -- were unfamiliar so consequently took unexpected amounts of time to complete. Because apprehension was approached to these features, extensive testing time was dedicated to them to ensure they were actually fully functional after implementation. At this point, it became very obvious to the team that time-management was something that was neglected as they felt pressure and stress while trying to meet deadlines. Sprint 4 was met with greater difficulty than Sprint 3 for many reasons. A key reason is that the user stories expected for implementation were even more technical and complicated than the ones from Sprint 3 as they relied and expanded on them. An example of this includes implementing functions that allow enabling editing, searching for and viewing customer orders and user details.' Nonetheless, all user stories expected for implementation for Sprint 4 managed to be implemented. From this, it can be realised that these new, complex and unexpected design challenges could be tackled in a more efficient way. During the user stories creation phase, the design aspects of every user story should be researched and be given an implementation plan. For example, if the team comes creates user stories that seem problematic, then detailed research should be made on to implement these user stories, instead of conducting this research at an inconvenient time later on on the project. This would help prevent time from being wasted and frustration later on in the project. Additionally, perhaps stronger discussion and research on the choice of programming language -- and the challenges it would create -- for the project should have been made. This would absolutely reassure the team if the programming language C# was really the most appropriate and efficient option for all team members.

The overarching lesson learned from Sprints 3 and 4 is that: things are likely to not go to plan, despite how planned they may be, while somethings need a plan, despite how easy they seem to be. In future, the team will ensure that they develop backup plans over potential major issues -- despite how unlikely they appear --, conduct more research and discussion over the design phase of the project (to improve clarity and time management), and inform the client team of any internal issues in the group (so they can appropriately adjust their expectations). Release 3 will implement the final user stories set for the project: accessing order overviews, manually adding users to the database and order notifications for customers and employees.

The development team has adhered to the SCRUM process very effectively throughout the entire project. All changes to the design of the project have consistently been uploaded to Github and merged with the master branch. This has allowed each team member to easily access updated versions of the project, and upload any changes by simply merging their changes to a branch. All documentation related to the project has consistently been uploaded to Github, too. This documentation was supported by Google Drive and Google docs to allow real-time online editing and viewing of these documents. Github was the perfect choice for a project hosting website and will gladly be used for future work.