

PRCO204

GROUP X

REPORT

Joseph Stephens (10619861), Marc Rasell, Jack Edwards, Amoata Eyorekon

**Contents**

**1.0 Introduction**

1.1 Project Links: Website, GitHub, YouTube video

1.2 Our scenario

1.3 Assumptions made

**2.0 Understanding of Agile Philosophy and Approach Taken**

2.1 Agile approach rather than Waterfall

2.2 Resources consulted

2.3 Implementation of principles learned

**3.0 Project Management**

3.1 Roles taken by the team

3.2 Meetings and decisions, remote working

3.3 Communication

**4.0 Development of the Project**

4.11 Sprint 1 plan, meeting 1

4.12 Sprint 1 review

4.21 Sprint 2 plan, meeting 2

4.22 Sprint 2 review

4.23 Story map

4.3 Sprint 3 plan, meeting 3

4.31 Sprint 3 review

4.4 Sprint 4 plan, meeting 4

4.41 Sprint 4 review

4.5 Sprint 5 plan, meeting 5

4.51 Sprint 5 review

4.61 HCI Assessment, Ethical approval

4.62 Accessibility

4.63 GDPR

4.64 UML diagrams and Normalization

4.65 Risk Assessment

4.66 Security Risk Assessment and Mitigation

4.67 Product Backlog, what was implemented and who implemented it

4.68 Product Release Plan

4.7 Screenshots of the solution and product factsheet

4.8 Problems encountered

**5.0 Lessons Learned**

5.1 Importance of regular meetings

5.2 Communication

5.2 Remote working

5.3 Coding as a Team

5.4. Identifying individual strengths in the team

**6.0 Conclusion**

6.1 Reflection of project successes/challenges

6.2 Personal development: teamworking, agile philosophy

6.3 Looking forward: what could have been done differently

**1.0 Introduction**

1.1 Project Links

Website: <http://web.socem.plymouth.ac.uk/intproj/prco204_x/index.php>

GitHub: <https://github.com/Plymouth-University/prco204-flight-crew>

YouTube video: To be completed by Joseph Stephens

1.2 Our scenario

To be completed by Joseph Stephens

1.3 Assumptions made

To be completed by Joseph Stephens

**2.0 Understanding of Agile Philosophy and Approach Taken**

2.1 Agile approach rather than Waterfall

Our approach to the project differed from the usual Waterfall approach of completing the project step by step i.e. design, database, web, testing. Rather than complete each step completely we tried to focus on implementing pieces of functionality each sprint, building up the application step by step.

We defined 4 stages to the project in our story map and aimed to complete one stage per sprint over a two-week period. This approach worked well as it kept the project on track and allowed us to focus on small pieces of functionality at a time as a group. However, even small pieces of functionality could be complex, so breaking it down was the right approach.

This approach meant the design and testing would be developing as the project grew, which meant even the design was agile and could change.

2.2 Resources consulted

There were several online resources in the reading list for this module that helped guide the approach taken together with the recommended textbook, “Agile Software Development, Principles, Patterns and Practices”. These resources helped to provide the knowledge of the agile approach and how we were to define a project backlog and create two-week sprints to implement functionality into the application. They textbook suggested the use of xml for acceptance tests, and that OOP should only be implemented where necessary. These resources were useful given that we had no prior experience of using this methodology.

2.3 Implementation of principles learned

We planned the project in two-week sprints, implementing functionality from the stages in the story map. Each person in the team had their assigned role – project owner, scrum leader and technical lead. This helped to give different perspectives in the team meetings.

We gradually built up the functionality and we also had an acceptance test file in xml. Implementing unit testing was challenging because of the close integration of functions with the database. Automating the xml tests also seemed difficult so they were left as manual tests which could be run to check the site was working. This was not ideal but provided a way for testing.

**3.0 Project Management**

3.1 Roles taken by the team

To be completed by Jack Edwards

3.2 Meetings and decisions, remote working

To be completed by Jack Edwards

3.3 Communication

To be completed by Jack Edwards

**4.0 Development of the Project**

4.11 Sprint 1 plan, meeting 1

To be completed by Joseph Stephens

4.12 Sprint 1 review

To be completed by Joseph Stephens

4.21 Sprint 2 plan, meeting 2

To be completed by Joseph Stephens

4.22 Sprint 2 review

To be completed by Joseph Stephens

4.23 Story map

To be completed by Joseph Stephens

4.3 Sprint 3 plan, meeting 3

To be completed by Joseph Stephens

4.31 Sprint 3 review

To be completed by Joseph Stephens

4.4 Sprint 4 plan, meeting 4

To be completed by Joseph Stephens

4.41 Sprint 4 review

To be completed by Joseph Stephens

4.5 Sprint 5 plan, meeting 5

To be completed by Joseph Stephens

4.51 Sprint 5 review

To be completed by Joseph Stephens

4.61 HCI Assessment, Ethical approval

To be completed by Joseph Stephens

4.62 Accessibility

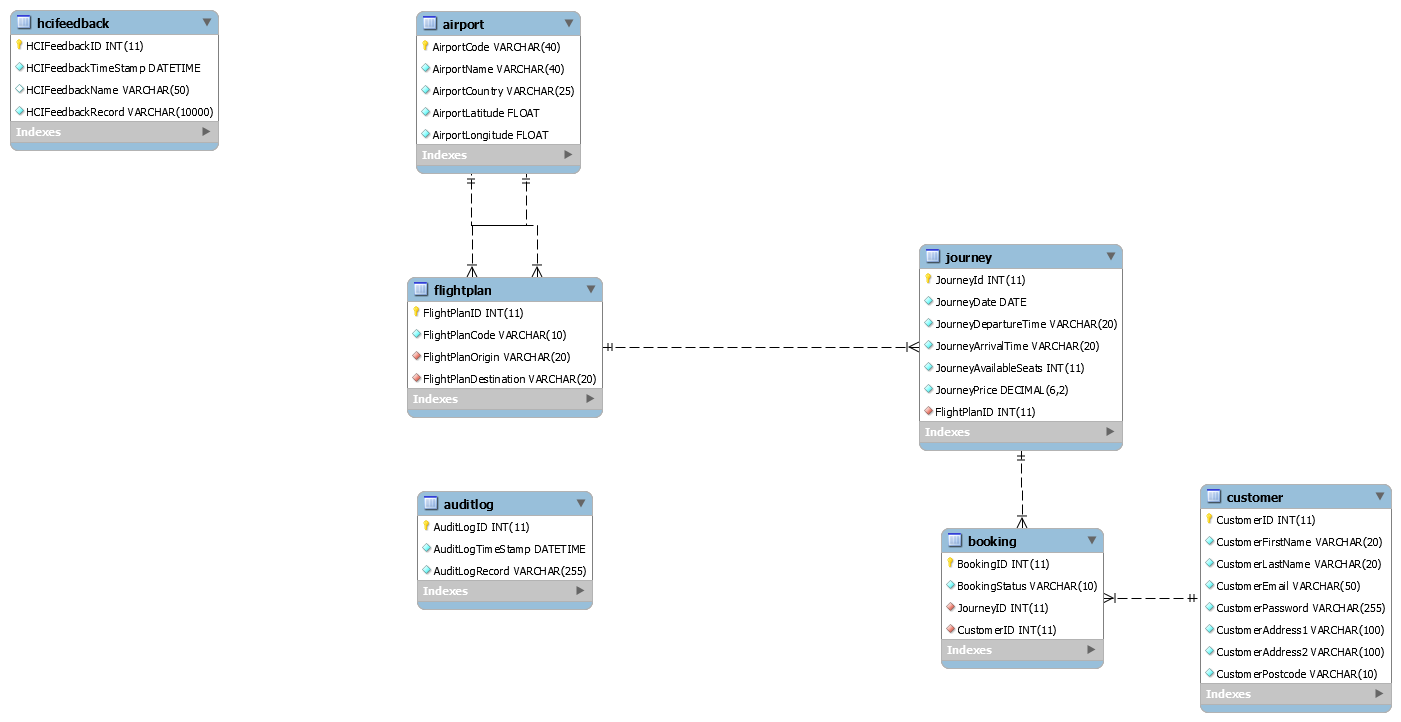
To be completed by Joseph Stephens

4.63 GDPR

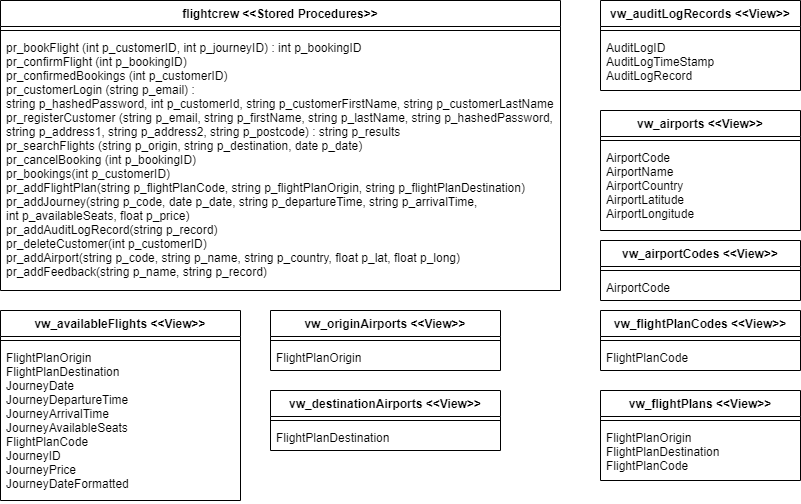
We had a terms and conditions page set up early in the development of the application. However, when the requirements of GDPR were investigated it became clear that more needed to be added to this. We did not have any cookies or mailing lists, but we needed to make it clear that user data would not be sold. As we were planning to implement admin stats and this functionality might draw on user data, we needed consent for this in the terms and conditions. We also needed to tell customers that their data would only be used for providing a service and for admin stats/security log and that it would be deleted after a set time, alternatively they could delete it from their account. We also needed to inform them they could request a copy of their data. So as a result of this the terms and conditions were updated to reflect current practices.

4.64 UML diagrams and Normalization

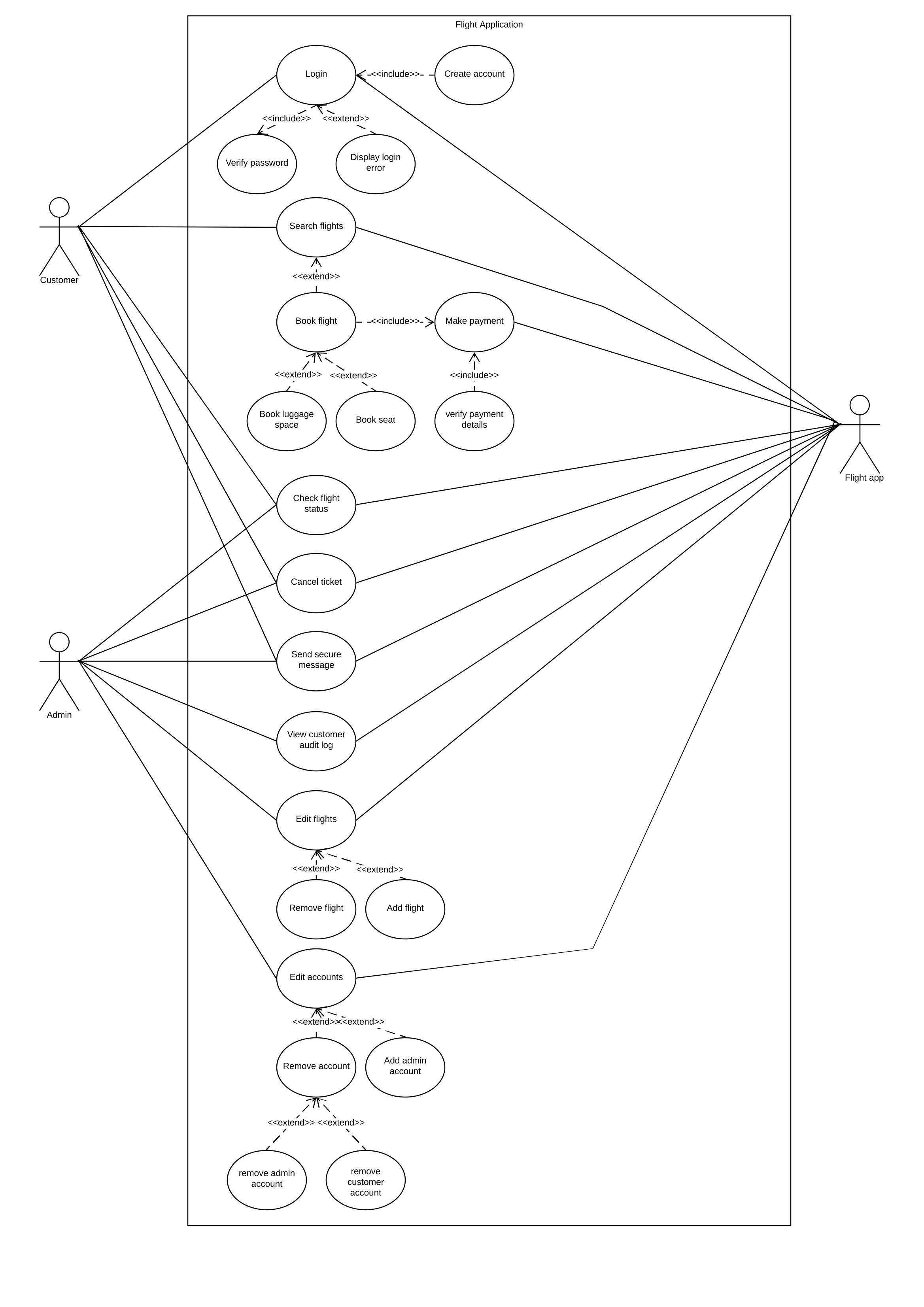
Entity Relationship Diagram



Procedures, Views and Triggers



Use case diagram



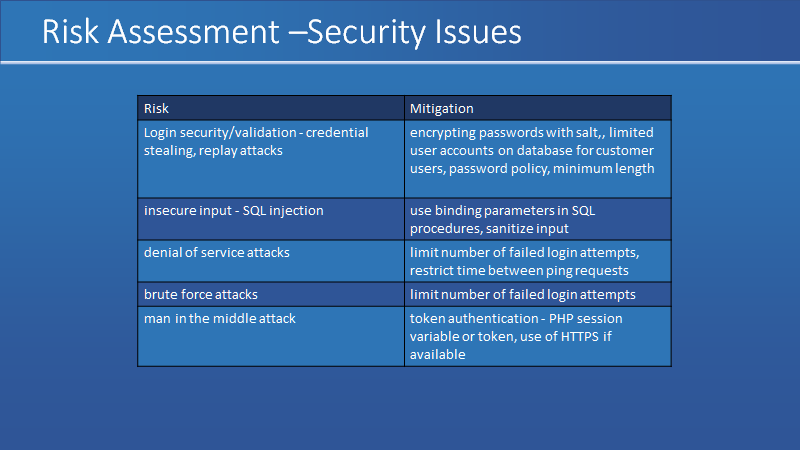
Normalization

|  |  |  |  |
| --- | --- | --- | --- |
| **UNF** | **1NF** | **2NF** | **3NF** |
|  |  |  |  |
| **JourneyID** | **JourneyID** | **JourneyID** | **JourneyID** |
| JourneyStartTime | JourneyStartTime | JourneyStartTime | \*FlightPlanId |
| JourneyEndTime | JourneyEndTime | JourneyEndTime | JourneyStartTime |
| JourneyAvailableSeats | JourneyAvailableSeats | JourneyAvailableSeats | JourneyEndTime |
| FlightPlanId | FlightPlanId | FlightPlanId | JourneyAvailableSeats |
| FlightPlanOrigin | FlightPlanOrigin | FlightPlanOrigin |  |
| FlightPlanDestination | FlightPlanDestination | FlightPlanDestination | **FlightPlanId** |
| (BookingID |  |  | FlightPlanOrigin |
| BookingPaid | **JourneyID** | **BookingID** | FlightPlanDestination |
| CustomerID | **BookingID** | BookingPaid |  |
| CustomerFirstName | BookingPaid |  | **CustomerID** |
| CustomerLastName | CustomerID | **JourneyID** | CustomerFirstName |
| CustomerAddress | CustomerFirstName | **BookingID** | CustomerLastName |
| CustomerPostCode | CustomerLastName | CustomerID | CustomerAddress |
| CustomerPhoneNumber | CustomerAddress | CustomerFirstName | CustomerPostCode |
| CustomerEmail | CustomerPostCode | CustomerLastName | CustomerPhoneNumber |
| CustomerPassword) | CustomerPhoneNumber | CustomerAddress | CustomerEmail |
|  | CustomerEmail | CustomerPostCode | CustomerPassword |
|  | CustomerPassword | CustomerPhoneNumber |  |
|  |  | CustomerEmail | **BookingID** |
|  |  | CustomerPassword | \*JourneyID |
|  |  |  | \*CustomerID |
|  |  |  | BookingPaid |

4.65 Risk Assessment

To be completed by Jack Edwards

4.66 Security Risk Assessment and Mitigation



Passwords were hashed in the database for security, if we had been able to implement https this could have enhanced security to stop the plain text passwords being sent to the server unencrypted.

Binding parameters were used in the “database.php” file to prevent SQL injection attacks. And all user input was put through the “secure\_input.php” function to remove any dangerous characters.

If we had more time, we could have added the failed login attempts functionality by adding a failed number of login attempts to the customer table.

4.67 Product Backlog, what was implemented and who implemented it

To be completed by Jack Edwards

4.68 Product Release Plan

To be completed by Jack Edwards

4.7 Screenshots of the solution and product factsheet

To be completed by Jack Edwards

4.8 Problems encountered

To be completed by Jack Edwards

**5.0 Lessons Learned**

5.1 Importance of regular meetings

To be completed by Amoata Eyorekon

5.2 Communication

To be completed by Amoata Eyorekon

5.2 Remote working

To be completed by Amoata Eyorekon

5.3 Coding as a Team

To be completed by Amoata Eyorekon

5.4. Identifying individual strengths in the team

To be completed by Amoata Eyorekon

**6.0 Conclusion**

6.1 Reflection of project successes/challenges

To be completed by Amoata Eyorekon

6.2 Personal development: teamworking, agile philosophy

To be completed by Amoata Eyorekon

6.3 Looking forward: what could have been done differently

To be completed by Amoata Eyorekon