**K1J LTD Project**

Client Requirements and how I will meet them.

Firstly, I will access the task brief and create some ideas on how the program could be a success. The program will need to be built around an employee and a customer user interface that is easy to navigate around. In order to complete the client’s requirements, I will need to implement the following:

• Provide employees secure access from any location

• Provide tracking of customer interactions

• Provide management with information for decision making processes

Some additional features that I could add are:

• Dashboard-based analytics

• Email tracking and integration with Outlook and Gmail

• Instant messaging between employees and customers to update orders and tracking

• Files and content sharing.

I will need to make sure that the program is easy to understand and can be accessible to anyone even if they struggle to read. I will be able to do this by making sure that the font size is bigger than usual or better yet have a setting in the program that can determine the size of the fonts. The program will also include a button to change the appearance of the app. This will have 2 options, light mode and dark mode. This will mean the everyone will have an equal opportunity to use the program. The software must be easy to navigate around so that no matter what age a user is, they are able to know how to use the app effectively. This means that the program will need each page of the program easy to reach with clear buttons that will navigate users to different pages. The buttons will be labelled correctly and clearly so that the user is left with no confusion. The client has asked that the program is accessible by employees from anywhere. This will be done by making a database that will store all the employee’s data. This will mean that the employee can work remotely if they have local access to the database. This program will include the ability to be able to track customers interactions. This will include the tracking of users outgoing messages, item purchases and complaints.

**Functional Requirements / User Acceptance Criteria**

1. Allows the user to create an account.

There will be 2 options to create an account, one is to make a customer account the other is to make an employee account. Employee accounts can only be made if they have an employee pin meaning customers can’t make employee accounts. There will also be validation on the email and password so that their login is secure.

1. Allows the user to log into an account.
2. Allows the Employee to contact the customer.

There will be a shared database where the 2 account will read and write messages to each other on it

1. Allows the Customer to contact the employee.

There will be a shared database where the 2 account will read and write messages to each other on it

1. Gives the Employee user an insight to customer actions.

Get a list of customers on the page that is available for the employee to look over

1. Allows the employee to send links to customers.

When the employee is sending a message to a user, there is a option to add a link into the message.

**Non-Functional Requirements**

1. Easy to read interface
2. Secure system
3. Easy to understand error messages
4. Easy to navigate
5. Easy to scale up
6. All data must be safe and secure
7. Light and dark mode
8. Customers can write feedback

**Risks**

The app will need to have personal user data stored, such as: email, forename and surname all in which are protected by GTPR. Our system is at risk of accidently leaking data so as a software we need to make the app as secure as possible for this data not to be leaked.

The program will need to avoid copy right issues. This will mean images that will get used in the program will require permission from the original creator. Credit must be given to these images.

**KPI’s**

Customers can see their messages within a minute.

A customer purchases an item more than 4 times a month.

0 complaints have been made in 1 month.

Customers have logged into their account more than 10 times a month.