**Requirements Analysis – Digital *Maître d’Hôtel***

**Introduction**

This document will explore the requirements for the Digital *Maître d’Hôtel* project.

**Methods of Gathering Requirements**

Four methods will be used to gather requirements for the project.

Firstly, and primarily, a questionnaire has been created which will be issued to front of house workers in the hospitality sector. The questionnaire will centre on the tasks performed by the workers and their desire to see those tasks automated. The questionnaire will also ask the workers about existing technologies – namely, which features of said technologies work well, which features could be further developed or improved, and any features which should be added.

Secondly, personas will be developed, based on people from different demographics who work in the hospitality sector or frequent hospitality establishments.

Thirdly, user interviews will be conducted to gauge the opinions in detail of hospitality workers. The workers will be given the outline of the project and asked for their feedback. User interviews will also be given to potential hospitality customers.

Fourthly, and finally, I will draw upon my own experience of working within the hospitality sector. This method is of least significance but may help to supplement information from the other methods.

**Functional Requirements**

The functional requirements for the project are those requirements which consider the exact function of the system and are specific/bespoke to the project. Functional requirements are thus objective in character. The functional requirements include:

|  |  |
| --- | --- |
| **Requirement** | **Source(s)** |
| Ability for user (customer) to check table availability |  |
| Ability for user to browse menu |  |
| Menu to show items on a user interface, greying out meals which are not available |  |
| Ability for user to book a table |  |
| Ability for client (restaurant) to check table bookings |  |
| Ability for client to check food inventory |  |
| Ability for user to order meals |  |
| Track data on meal orders and table bookings for client |  |
| Ability for user to pay for meals |  |
| Intuitive and minimalist user interface |  |
| Ability to output orders to a client printer |  |
| Ability for client to alter menu when necessary |  |

**Non-functional Requirements**

The non-functional requirements of the project are more generic and are not necessarily bespoke to the project. These requirements consider additional factors such as performance, security etc. which are important but irrespective of the project’s function. Non-functional requirements include:

|  |  |  |
| --- | --- | --- |
| **Requirement** | **Category** | **Source(s)** |
| No storage of user information | Security |  |
| No storage of any payment information |  |
| Prevention measures for SQL injections |  |
| Operation for the entire duration of a service (~6 hours) without crashing (with 95% confidence) | Continuity |  |
| Modular code design | Scalability |  |
| Beta version of the project by February 2023 | Capacity |  |
| Complete version of the project by March 2023 |  |
| No more than 200 hours labour to be expended (~£3000 cost) |  |
| Ability to catch exceptions and prevent them from aborting the program | Reliability |  |
| Integration of pre-existing payment software |  |
| Low memory usage (need to quantify) | Performance |  |
| Low storage usage (need to quantify) |  |
| Complementary, warm colour scheme | Aesthetics |  |
| High-resolution graphics with complementary or no background colour |  |

**Conclusion**