# Pending - Abstract

/what is the project and why did you do it?

/what did you do and how?

/what are the important findings?

# Introduction

// briefly describe the goals software

Laser Mate! is a self-defined software engineering project that aims to develop a mobile web platform enabling sit-in restaurant customers to order and pay for their meals online.

## Done - Motivation

// what issue would restaurant owner faces without the software?

The continuous business closure due to the current covid-19 pandemic resulted in a sharp financial loss for many restaurant owners. To rebound from this year-long recession, owners must adapt and implement innovative ways to survive and recover from these circumstances. This dissertation work aims to provide a cost-effective software platform that promises an approximate 50% reduction in waitering expenditure, helping all restaurant businesses get through this difficult economic crisis and skyrocket from previous business performance.

The issue with traditional restaurant food ordering is the cost implications, in particular, an average waitering cost of ~£37,000. To operate a restaurant business, owners must recruit waitering staff to take customers to the table and to record and deliver meal orders to the kitchen. Other tasks of a restaurant waiter include to deliver and collect meals and to give and take payment. According to PayScale [1], the current waitering staff cost in the UK is £6.95 per hour and the average number of waiters in a restaurant is 2.6 people [2] in the U.S. A typical restaurant also requires waitering staff for 6 hours each opening day, from 12pm-2pm and 5pm-9pm. Assuming hypothetically that a restaurant opens for ~340 days a year, the subsequent average waitering staff cost is 2.6 people x 6 hours x £6.95 per hour x 340 days = ~£37,000.

// how and why this software solves the problem?

The sit-in mobile ordering platform reduces waitering costs by approximately 50% as it redirects the meal ordering and payment process to the customers. Restaurant owner would no longer require to record and deliver meal orders to the kitchen, as well as that to give and take payment, saving approximately 50% of the aforementioned £37,000 staff cost, at £18,500. This cost saving outcome is particularly crucial in light of the covid-19 pandemic, in which restaurant owners would hope for a significant profit boost to recover from previous losses due to restaurant closure.

// how does your evaluation outcomes support your software idea?

// how is your software progress? Does the software works?

## Aim

// what are the goals of the software (for different user types – user stories)?

// what are the business requirements

## Round 1 - Dissertation Outlines

This paper comprises of:

* The final software product chapter, which showcases the workflow of the overall software work, in the form of digital prototype design.
* The software requirement specifications chapter, that outlines the user stories and the non-functional requirements of the finalised system.
* The UI/UX design chapter, highlighting the HCI design principles and the related software products in the market.
* The implementation chapter, that explores the software tech stack and responsive design, as well as the database organisation.
* The evaluation chapter, which discusses the three phrases of system improvement processes. They are paper prototype, digital prototype, and the final software product.
* The test-driven development chapter, that demonstrates the unit-testing, and user acceptance criteria.
* The future work chapter, which summarises the foreseeable work that can be carried out for this project.

# Final Software Product

// describe each prototype (with diagrams and descriptions)

Find 12. Dissertation document

# Software Requirement Specifications

## Done - User Stories

|  |
| --- |
| Restaurant Customer |
| View, select, and tailor the meals available using a QR code  So that I can order the food and drinks that I want  View and adjust all the meals I ordered  So that I can confirm if these are the meals I want to order  Decide if the meals should come together or separately  So that the meals come in the way I expect  Tip and pay for the meals I ordered  So that the restaurant will receive the payment they require  Get a e-receipt  So that I can claim the money back from my employer |

|  |
| --- |
| Restaurant Owner |
| Register an account  So that I can use the platform  Edit menu details  So that I can customize the details in the restaurant ordering platform  Have help documentations  So that I can resolve any technical difficulties myself and go through the app tutorials with my employees. I can also get advice with restaurant business growth strategies. |

|  |
| --- |
| Company Staff |
| Have a database to record all the restaurant basic details  So that I can confirm a person's identity before helping them with a business enquiry  Edit the menu details for all the restaurant platforms  So that I can set up and update the restaurant ordering platform for the clients  Have a database to record all the restaurant QR codes  So that I can post a restaurant owner additional QR codes when they lost the ones they have obtained |

|  |
| --- |
| Company CEO |
| Have a database to record all the employee details  So that I can contact them when organizing team work and arrange staff payroll  Have a database to record the weekly restaurant business transactions  So that I can automatically organize weekly service charnges for the restaurant owners  Have a database to record the weekly company revenue and the number of active users  So that I can monitor and evaluate company performance and devise customer retention strategies |

## Non-Functional Requirements

### Restaurant Customer Ordering Platform

The restaurant customer ordering platform must be in the form of mobile web. The mobile layout must only work in portrait. The web framework must be responsive for all mobile phone screen sizes.

// what are the system requirements (data volume, scalability, test driven, responsive design, mobile web)?

## MOSCOW