Table of Contents

[Introduction (brief of the task) 1](#_Toc96508171)

[Research 2](#_Toc96508172)

[Software 2](#_Toc96508173)

[Hardware 3](#_Toc96508174)

[Emerging technology 3](#_Toc96508175)

[Digital solutions 4](#_Toc96508176)

[Guidelines and regulations. 4](#_Toc96508177)

[proposed solution (what im going to do for all user requiments) 4](#_Toc96508178)

[Functional (Main features that are needed/users storys) 4](#_Toc96508179)

[Story 1 4](#_Toc96508180)

[Story 2 4](#_Toc96508181)

[Non-functional (common features e.g. fast, responsive and easy to use) 5](#_Toc96508182)

[Security 5](#_Toc96508183)

[Scalability 5](#_Toc96508184)

[Performance 5](#_Toc96508185)

[Quality 5](#_Toc96508186)

[Accessibility 5](#_Toc96508187)

[Technical requirements 5](#_Toc96508188)

[Software needed for development(VSCODE, IDLE) 5](#_Toc96508189)

[Hardware needed for DEVELOPMENT (PC, specs) 5](#_Toc96508190)

[Hardware and software needed for deployment/PRODUCTION (what standered specs needed) 5](#_Toc96508191)

[Business context (is it worth making this product, do they get value out of it) 5](#_Toc96508192)

[Project managment/methodology 5](#_Toc96508193)

[Agile 5](#_Toc96508194)

[Risk assessment 5](#_Toc96508195)

[ASSUMPTIONS (what you think is right and backs you up) 5](#_Toc96508196)

[Decomposition of PROBLEMS (has been replaced with functional and non-functional) 5](#_Toc96508197)

[KPIs(simple milestones e.g.can run, can enter details, is very resposensive) 5](#_Toc96508198)

[user acceptance CRITERIA (assuiming that the user likes the solution in storys) 5](#_Toc96508199)

[Potential SOLUTIONS (what the user could use instead of making their own product) 5](#_Toc96508200)

[For the client (does it have the features they asked for) 6](#_Toc96508201)

[For the existing an potential customers 6](#_Toc96508202)

[Justification 6](#_Toc96508203)

[Needs of the client and users(write this in propsoed soultion and saves writing it again) 6](#_Toc96508204)

[Risks mitigated 6](#_Toc96508205)

[Regulatory guidelines and legal requirements 6](#_Toc96508206)

[Bibliography. 6](#_Toc96508207)

# Introduction (brief of the task)

I created this document to help provide information about a product that our client wants. We talk about: pre-existing products that our client might want to use instead, Risk and regulations that are involved, software and hardware we need. To see full description of the document, check table of contents. Our client currently provides a product that can:

* Access to a sale system.
* Access to customer details
* Advise customers about different products.

From the information I have been given it seems that our client works in the retail sector. They want a new product that will have these features integrated into the software:

* Provide employees remote access from anywhere.
* Provide tracking of customer interactions.
* Provide management with information for decision making processes.

From the information provided, the client wants some sort of webpage and server that can send and requests information to update the webpage. You can do this with many programming languages and there is probably products that can do this to. They also did some research of their own with pre-existing customers and gave us some extra features that could be added:

* 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.

Overall, these seems like simple product that probably already exist. However, we will investigate into this and find products that are like ours and see if they are better or if we can adapt their features into our own product.

## Research

### Software

#### Hubspot

#### Power Bi

### Hardware

#### Examples 1

### Emerging technology

### Digital solutions

### Guidelines and regulations.

#### GDPR

#### Computer regulations

#### User computer rights

## proposed solution (what im going to do for all user requiments)

## Functional (Main features that are needed/users storys)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Functional requirement number. | Functional requirement | Task | tasks | User acceptance criteria |
| 1 | As the client, I want to create an application that collects and sends data. | 1 | Create the base application. | Operational application. |
|  |  |  | Import the libraries. |  |
|  |  |  | Create a simple loop. |  |
| 2 | As the client, I want to send emails and receive emails about user details and data. | 2 | Create Webpage. | Integrated Outlook and Gmail. |
|  |  |  | Create email integration. |  |
|  |  |  | Loads email page. |  |
|  |  |  | Send email to client and store user data as well. |  |
|  |  |  | Store inside of excel or SQL. |  |
| 3 | As the client, I would like a form that the user can input their details into. | 3 | Create form page. | Interactive user form. |
|  |  |  | Take user inputs. |  |
|  |  |  | Send form to server. |  |
|  |  |  | Save form details inside of excel or SQL. |  |
| 4 | As the client, I want to input data through the CRM | 4 | Create input button. | Manual input of data. |
|  |  |  | Load CRM form page. |  |
|  |  |  | Send data to server. |  |
|  |  |  | Load data in CRM. |  |
| 5 | As the client, I want to store the user data. | 5 | Create server that stores data in SQL or spreadsheet. | Save user data. |
| 6 | As the client, I want to load my user data into a dashboard. | 6 | Create simple dashboard in webpage. | Dashboard of data. |
|  |  |  | Request data from server and load in webpage. |  |

## Non-functional (common features e.g. fast, responsive and easy to use)

### Security

### Scalability

### Performance

### Quality

### Accessibility

## Technical requirements

### Software needed for development(VSCODE, IDLE)

### Hardware needed for DEVELOPMENT (PC, specs)

### Hardware and software needed for deployment/PRODUCTION (what standered specs needed)

## Business context (is it worth making this product, do they get value out of it)

## Project managment/methodology

### Agile

### Risk assessment

#### Outline risks

#### How to avoid risks

## ASSUMPTIONS (what you think is right and backs you up)

## Decomposition of PROBLEMS (has been replaced with functional and non-functional)

## KPIs(simple milestones e.g.can run, can enter details, is very resposensive)

## user acceptance CRITERIA (assuiming that the user likes the solution in storys)

## Potential SOLUTIONS (what the user could use instead of making their own product)

### For the client (does it have the features they asked for)

### For the existing an potential customers

## Justification

### Needs of the client and users(write this in propsoed soultion and saves writing it again)

### Risks mitigated

### Regulatory guidelines and legal requirements

## Test strategy

I have implanted a test strategy to help with guiding this project and help with the agile method we will be using. The main test types will be using are Whitebox and Blackbox. Whitebox is the user knowing knowledge of the program and how it functions e.g., knowing how the code works and what each part does. Blackbox would be having none or very little knowledge about the program. Blackbox would be like pressing a button to see if it responds. I have added the sub testing types as well to give more depth and better understanding for future developers who work on this product as well.

|  |  |  |  |
| --- | --- | --- | --- |
| Test number | Component to be test | Test type on component | Prerequisites and dependencies. |
| 1 | Server status is active | Whitebox test/unit testing | Is the server running and requesting/sending data from the dashboard. |
| 2 | Webpage loading | Blackbox test test/functional testing | Is the webpage loading and showing all the features. |
| 3 | Form page loading | Blackbox test /functional testing | Is the form page loading and showing all text boxes and buttons. |
| 4 | Features and design. | Blackbox test/ acceptance testing/beta testing | Is the client happy with the product and meets their requirements of the product. |
| 5 | Data being stored | Whitebox test/ unit testing | Is the server storing the data to the correct file |
| 6 | Webpage data being sent | Whitebox test/ unit testing | Is the data being sent to the server and being recognised. |
| 7 | Login page protection | Whitebox test/ security test | Login page checks user details and only opens to correct credentials. |
| 8 | Webpage speed | Blackbox test/non functional | Is the webpage loading quickly and not slowing down performance. |
| 9 | New data | Blackbox test/ regression testing | New data does not corrupt old data and cause website to not function |
| 10 | functions are working | Whitebox test/ module testing | Run each function by itself to see it works and runs correctly. |

## Bibliography.

Data requirements

|  |  |  |  |
| --- | --- | --- | --- |
| Variable name | Function | Data type | Reason |
| User\_Information | Checks if the user data they inputted in the form is correct. | Boolean | If the info entered:  Is valid (True) form is sent  Is invalid (False) form is not sent and display error message. |
| User\_Email | Checks that the email they have entered is correct. | String | Email is saved a string:  Is valid (True)  Is invalid (False) throws error |
| Login\_Validation | Compares the details of the input and login credentials to see if it matches. | String | Login input:  Correct, website will load.  Incorrect, error message will be displayed |
| Error\_Message | Sends error when something does not match. | String | When information is incorrect sent error message. |
| Data\_Stored | Data being sent and pulled from server | String/int | Server:  Request data  Send data |
| Forename | Checks that Forename is not integer | String | If forename  Is a string it will be accepted  Otherwise error message will be displayed. |
| Surname | Checks that Surname is not integer | String | If surname  Is a string it will be accepted  Otherwise error message will be displayed. |