**Project Proposal:** GotoGro-MRM

**Solution Direction**

In a realistic setting, the project calls for a web hosted database management system (DBMS) with desktop clients on all relevant point of sale (POS) machines. Together these constitute and enterprise framework. However, given the timeframe, we can simplify the implementation to either be entirely web hosted – accessible through a browser – or entirely desktop hosted, where the database exists directly on the host machine. Table 1 shows a comparison between the two alternatives:

**Table 1. Comparison of Solution Directions**

|  |  |  |
| --- | --- | --- |
| **No.** | **Desktop Application** | **Web Server** |
| 1 | Database is stored locally on machine | Database is stored on the web on a server |
| 2 | Program is self-contained | Program must be stored a separate web server |
| 3 | Program is not easily accessible from mobile devices | Program can run on any device with internet access |
| 4 | No additional hardware required | Server required, either through cloud-hosting or physical storage |
| 5 | Generally cheaper | Generally more overheads |
| 6 | Requires coding skills only | Requires coding skills, also skill with markup languages HTML, CSS etc. |

To make a distinction between the two, we must consider the problem domain, the solution domain as well as personal familiarity with relevant business knowledge, skills, and technology:

**Table 2. Knowledge of the Problem Domain**

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Item** | **Description** | **Competency** |
| 1 | Customer needs | Knowledge of what a customer needs from the business, specifically, why are they using a member-based retail chain | High |
| 2 | Business needs | Knowledge of business needs, why they chose to use a member system, what benefits this provides | High |
| 3 | Technology | What kind of physical systems do they currently have in place, how do these function, how can they be used/repurposed | High |
| 4 | Network Architecture | How does the business currently manage its network, security concerns, how can this be leveraged to solve problems | Medium |

**Table 3. Knowledge of the Solution Domain**

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Item** | **Description** | **Competency** |
| 1 | Technology | What technologies exist already to deliver digital reporting, how are these used in industry | High |
| 2 | Server management | How to set up and manage a physical/cloud-hosted server to store company information | Low |
| 3 | Cybersecurity | What are the threats to the system and how can these be managed | Medium |
| 4 | Coding | Competency in a coding language enough to deliver the desired functionality | High |
| 5 | UI | Competency in linking raw code to a user interface which can be used by the intended users without issue | Medium |

**Table 4. Personal Knowledge**

|  |  |  |
| --- | --- | --- |
| **No.** | **Item** | **Description** |
| 1 | Worked in retail industry | Being both a customer and a server in a retail environment, I am well versed in the needs of both customer and business |
| 2 | Worked on a project with local database management | Having worked on a project involving storing a database on a raspberry pi edge computer accessed through a local network, I have a firm grasp of both database management and UI design, especially with web technologies |
| 3 | Website design | I have designed 3 fully operational websites. I have a strong understanding of HTML, CSS and how to implement them to deliver a quality front end |

**Table 5. Personal Skills**

|  |  |  |
| --- | --- | --- |
| **No.** | **Item** | **Description** |
| 1 | HTML, CSS, PHP, Javascript | Strong markup language skills as well as Javascript to enable processing in web applications |
| 2 | PgAdmin and PostgreSQL | Strong skills with PgAdmin to administer a PostgreSQL database to store information coming in in Realtime. |
| 3 | MySQL | Basic university level of understanding with MySQL workbench and MySQL databases |
| 4 | C#, C++, Ruby, Python, Javascript | Strong programming skills, specifically geared towards desktop applications |

Given my knowledge and skillset, I feel I am well equipped to handle either solution direction. Even though we will likely pick one or the other, I will present a high-level design for an enterprise architecture involving both solution directions.

**High Level Design**

The high-level design is broken down into three layers:

The client layer (application layer), which provides a UI and a visual method of interacting with the system.

The logic layer, where business logic takes place and drives things like calculation and reporting.

The database layer, which houses the raw business data.

Together they create an architecture as follows:

**A picture containing diagram

Description automatically generated**