Design Philosophy

Last Updated: 2023-Feb-22

Introduction

The Vehicle Maintenance Application (VMA) is deliberately designed with end users as the number one priority. This prioritizing is not solely about how the end user interfaces with the application, but also about real ownership, platform independence, affordability and custom modification as required.

VMA is designed as a tool that adds value to a client's relevant garage or service operations. The hope is that:

1. it realizes increased efficiency in operations and planning
2. other developers will extend it, adding extra needed functionality
3. end users would derive long term value from its use

To meet the above objectives, the design philosophy inevitably greatly influences the functional design objectives of the software. This will be seen in the decisions for application type (web application), software stack, (PHP, MySQL and JavaScript), licensing (free and open source) and pricing model, (low one time payment).

Platform Independence

Some core non-functional requirements[1] of this system are:

- minimum dependency on proprietary technologies, frameworks, plugins, workflows and associated external 3rd party libraries, apis, services etc. Specifically, there will be no inclusion of two-factor authentication schemes as part of the application's operation.
- The application must also not require specialized hardware.
- The application can be owned by persons who choose to pay for it., or even if it is distributed freely. That is, it must not be leased as a service requiring recurring fees similar to subscription based software products/services.

In line with these non-functional requirements, platform and technology independence is a key design requirement of this application. The philosophy here is that end users can fully own the application, and that it's functionality should never be constrained by external agents that are not part of the application's core architecture[2]. The hope is that the functionality provided can exceed the lifetime of the businesses that it supports, for a fair, one time price. Changes in hardware, computer setup and even the loss of internet service must not impact the usability of this application.

1 These are constraints on the services or functions offered by the system. They include timing constraints, constraints on the development process, standards, etc. (Sommerville, 2001).

2 E.g. social media plugins, authentication plugins