Part 1

Supplier

1.0
Update inventory with newly received stock

Order inventory

DS1: Inventory database

3.0
Generate orders

Notification of threshold reached

Order request

Amounts added to database

Amounts subtracted from database

Counts

Stock on Hand

2.0
Update inventory decrementing for items used

Supplier

Payment for supplies

Invoices

4.0
Make payments

Invoices
Part 3.

MEMORANDUM

To: John Smith, City Manager
From: Hugo Menendez, IT Analyst
Date: September 14, 2015
Re: Adoption of Open Source Platform

The use of technology in the workplace is at an all time high. The purpose of the local government is to provide the citizens with the goods and services they require. In order to do so, it is imperative that the city government allocates its resources in the most efficient manner. Moving away from traditional software and adopting an open source platform would bring many benefits to the city government and I believe those benefits will far outweigh the costs.

One of the most popular open source platforms available is LAMP. LAMP is composed by Linux (Operating system), Apache (Web server), MySQL (Database Management) and PHP (Programming). Before we decide to adopt LAMP, it is important to state its main advantages and disadvantages.

The main advantage of using open source software is its customizability. The use of open source software allows the city developers to use already existing code and create software that is unique to our city. Another advantage of open source software is its adaptability. Developers can use existing software and modify it as necessary to achieve a specific goal. The third and final advantage of open source software is operating cost reduction. When buying software from vendors (Microsoft, Oracle), the buyer is required to pay an annual fee in order to keep access to updated versions of the software and its security features. The cost of these yearly memberships can be quite significant.

As mentioned above, open source can help reduce operating costs. However, one of the main disadvantages to take into consideration is the initial investment that open source software requires to get it up and running. Upfront costs can be quite substantial. In addition, open source software will require the city to do its own security, which can be quite costly.

In conclusion, there are many advantages and disadvantages to adopting an open source platform. However, I believe that the benefits of open source outweigh the costs. The ability to customize our software and adapt it to the citizens needs cannot be obtained by buying software licenses from external vendors. Finally, as years pass and the use of technology keeps growing, the creation of our own software will allow for better scalability. This means that the city’s systems will be able to handle an increasing amount of work.

Hugo Menendez
Hugo Menendez, IT Analyst

Each one of the steps mentioned above has the potential to create problems and impede the creation of a new IT application. During the preliminary investigation, it is important to understand what the purpose for the new software is. Should we actually commit our resources to develop this application or is there an existing application that can provide us with the same function?

The second step, requirement determination, deals with the scope of the application. In particular, the inputs, outputs, performance, and security features needed to make a successful application. This step is critical and failing to determine each one of these requirements can lead to a failed process. According to the GAO, lack of security features is one of the biggest reasons why systems fail. The third step consists of conceptually designing the application and then the physical design begins. The main risk in this stage is that the developers may not specify enough on all the deliverables during the conceptual design, which would lead to issues creating the physical system and its subsequent failure.

The fourth step is Development and Construction of the software. In this stage, the software code is written and the software and its data are tested. In addition, the system is tested as a whole to try to avoid unexpected errors during implementation. During this stage, any issues during the system tests can lead to cost overruns and schedule delays. The GAO includes both of those issues in the top 10 reasons why systems fail.

The fifth step, implementation, is composed by three main parts: training the users, site preparation for installing the software, and conversion strategies (whether we want to use a pilot, parallel systems, phase in, or direct cut over). One of the main reasons why systems fail during implementation is that the users do not receive enough training and cannot use the software up to its potential. In addition, lack of site preparation can lead to the software not being able to be installed in a certain machine. During the process of evaluation, the main error that can occur is incorrectly measuring the performance of the system, its usability or coming to the realization that the original purpose for which the system was developed is not being satisfied.

One example of a failed system is the TAURUS initiative of the London stock. TAURUS was created in the 1980’s and was supposed to help the London stock Exchange to transfer from paper communication to an automated system. The main two reasons why the project failed were cost overruns and scope creep. Both of those issues happened because of poor project management, especially in the early stages of the SDLC process. Its creators wanted to keep making TAURUS bigger and better and that led to cost overruns (TAURUS cost £75M). The result of the bad implementation of the SDLC steps led to TAURUS’ failure and cancellation.