CHAPTER 2

THEORETICAL FOUNDATION

2.1 Chapter Overview

The application will be divided into two parts: (i) the web based application and (ii) the database management. The web based application will use PHP scripting [22], with an internet browser as the browsing tool. For the database management, the application will use MySQL[15]. Several concepts will be used for the information system development will be described below.

2.2 Process Model

The process model that will be used for the development of tracking web-based application is the Waterfall model, or also known as classic life cycle. The reason for using such model is the simplicity of the working scope and time. According to Pressman [11] the Waterfall model, or sometimes called the classic life cycle, suggests a systematic, sequential approach to software development that begins with customer specification of requirements and progresses through planning, modeling, construction, and deployment, culminating in on-going support of completed software.

Furthermore, Whitten[16] also mentions that this model has some drawback such as

- Real projects rarely follow the sequential flow that the model proposes. Although the linear model can accommodate iteration, it does indirectly. As a result, changes can cause confusion as the project team proceeds

- It is often difficult for the customer to state all requirements explicitly. The Waterfall model requires this and has difficulty accommodating the natural uncertainty that exists at the beginning of many projects.

- The customer must have patience. A working version of the program(s) will not be available until late in the project time span. A major blunder, if under detected until the working program is reviewed, can be disastrous.

Some advantages of the Waterfall model according to Pressman [1] are very useful process model in situations where requirements are fixed and work is to proceed to completion.
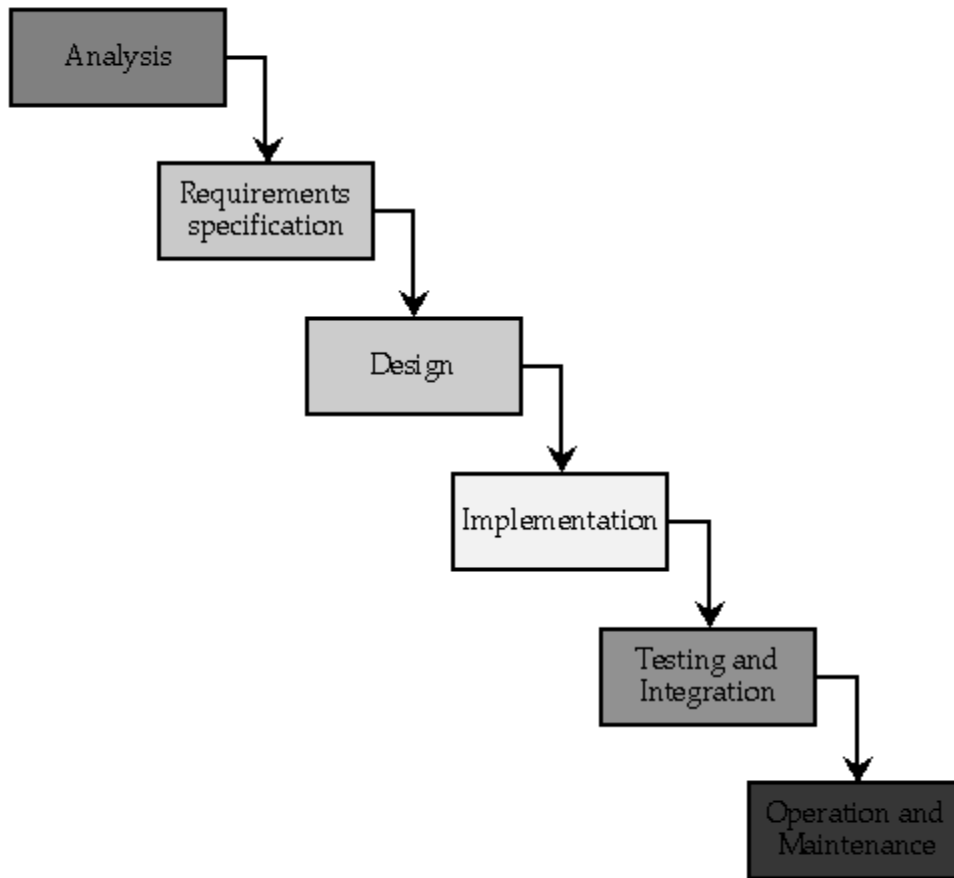
Figure 2.1 The Linear Sequential Model[9]

To understand more about the linear model, we need to discuss about the activities that are highlighted in it. The Waterfall model [17] encompasses the following activities:

- System/Information engineering and modeling

  This system view is essential when software must interact with other elements such as hardware, people, and database. System engineering and analysis encompass requirements gathering at the system level with small amount of top

level design and analyses. Information engineering encompasses requirements gathering at strategic business level.

- Software requirement analysis

    The requirement gathering process is intensified and focused specifically on software. Requirements for both the system and the software are documented and reviewed with the customer.

- Design

    Software design is actually a multi step process that focuses on four distinct attributes of a program: data structure, software architecture, interface representations, and procedural (algorithmic) detail. The design process translates requirements into a representation of the software that can be assessed for quality before the coding begins. Like requirement, the design the design is documented and become part of the software configuration.

- Code Generation

    The design must be translated into the machine-readable form. The code generation step performs this task. If design is performed in a detailed manner, code generation can be accomplished mechanistically.

- Testing

  Once code has been generated, program testing begins. The testing process focuses on the logical internals of the software, ensuring that all statements have been tested on the functional externals; that is, concluding tests to uncover errors and ensure that defines input will produce actual results that agree with required results.

- Support

  Software will undoubtedly undergo change after it is delivered to the customer (a possible exception is embedded software). Change will occur because errors have been encountered, because the software must be adapted to accommodate changes in its external environment, or because the software requires functional or performance enhancements. Software support/maintenance reapplies each of the preceding phases to an existing program rather than a new one.

By analyzing the problems of the previous system and the solutions that will be implemented, Waterfall model is the best match criteria on finishing the web-based tracking application.

2.3 Theoretical Framework

2.3.1 Web Based System

As for the web-based system, the author would use PHP scripting [8], as PHP is basically the running script(s) on certain web page. The script(s) will be embedded to the HTML page and work according to what the script(s) is intended for.

According to Sidik[13], PHP is script, suitable for server-side web script programming, script that enable the creation of HTML document on the fly, HTML document that is created from an application, not HTML document that is created using text editor or HTML editor. By using PHP, then the maintenance of a website becomes easier. Data update process can be done by using the application that is created by PHP scripts.

Sklar[14] mentions that PHP has a lot of advantages to be used as scripting language for the web-based side of the application. One of its advantage is the programming does not have to write all HTML documents as the part of the output of PHP script, just need to write which part is part of HTML tag and which part that has to be written or produced from the PHP script's program. PHP script will then be embedded using the specific beginning tag and end of tag, which enable the programming to enter and leave from PHP script's mode.

Moreover, it[18] also described that PHP can basically finish all tasks that can be done by CGI program, such as: get data from form, produces dynamic web page content, and accepts cookies. Yet, the most reliable, remarkable, and significant feature is the support to a number databases. However, in this thesis, the author uses MySQL as the database system and management of the web-based tracking system.

2.3.2 Three Tier Application

.   For the application itself, it is a three tier application. The advantages of using three tier application, according to Ratschiller[12], are thin clients make easy upgrades, centralized application server management is easier to manage, and model allows for horizontal scalability.

To understand more about the three tier application, we need to know what are the three tier. The three tier is presentation logic, which is the customer(s); business logic, which is the application server; and, database logic, which is the server that keeps all the database inside.

A multi-tier (often referred to as three-tier) architecture provides greater application scalability, lower maintenance, and increased reuse of components

Ratsciller[4] also mentions that three-tier architecture offers a technology-neutral method of building client/server applications with vendors who employ standard interfaces, providing services for each logical layer.

In the first tier is a thin client—translated to the world of Web applications, this would be the browser. This thin client wouldn't be necessarily a customer, but anyone who uses the system by accessing it through the web browser, for example: customer to search for their goods, or management to search for reports.

Second of all, the middle tier (application server) is obviously PHP scripts embedded application (and the Web server as host application). It's the business logic that will operate and automate all the input, which has been done throughout the web-browser by the clients (on the first tier).

Third of all, while the third tier consists of the Database Management System or shortly named as Data Server, that acts as data storage of all input and output that are done and produced throughout the thin client and application server.

In the web-based tracking system's application, the application server will contain the web-based system, while the web-based system itself will be supported by the database server.

According to Frankel[6], The thin client that thus became *de rigueur* for Web-based Internet and intranet applications spawned a variation of three-tier architecture in which the presentation facilities of the client machine were limited to a web browser. Some of the user interaction and presentation facilities ran on remote web servers, which dynamically generated HTML and sent it to the client; in that sense, web servers spanned the client tier and mid tier. Three-tier architecture's concept of the database tier also evolved to include various enterprise information servers that are not mainframe-based, which mean, the entire tier do not have to be collected at one place (in this case, mainframe), but also separated on several computers.
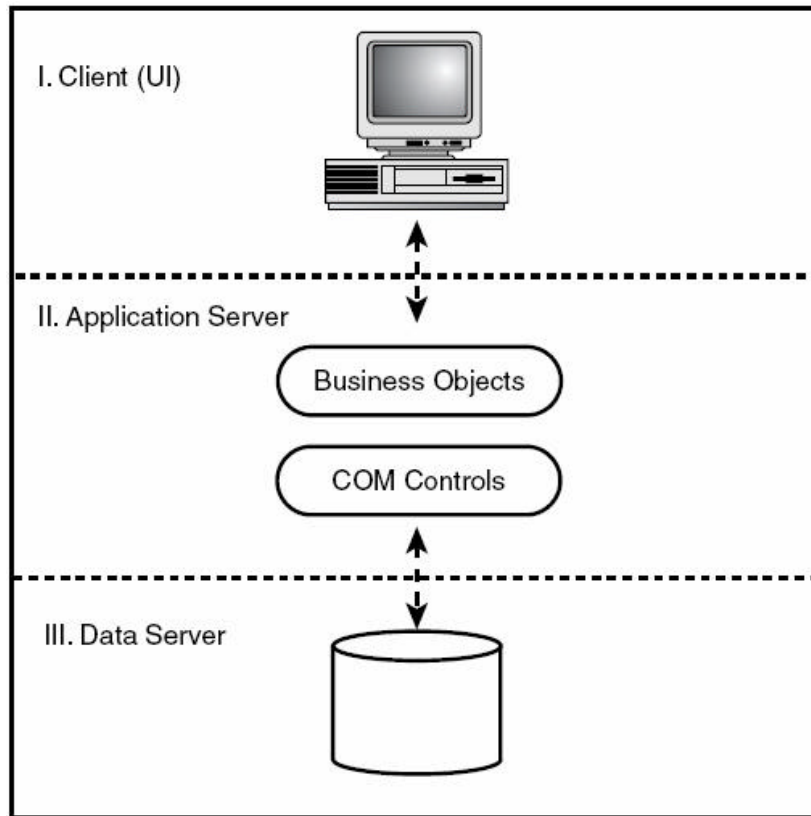
Figure 2.2 Multi-Tier Application Layers (Three Tier Application)[6]

Therefore, in the application and system, the application server can be accessed with any internet browser, and the business logic is handled using PHP scripts and database logic is handled using MySQL.