# CHAPTER FOUR

# DESIGN

## 4.1: Design in Software Engineering

A software design is a meaningful engineering representation of some software product that is to be built. It can be traced to the customer’s requirements and can be accessed for quality against predefined criteria. In the software engineering context, design focuses on four major areas of concern: data, architecture, interfaces, and components. [[1](#Tag01_4)]

The design process is very important, because the emphasis in design is on quality; therefore it provides the representation of software that can be accessed for quality.

Furthermore, this is the only phase in which the customer’s requirements can be accurately translated into a finished software product or system. Thus, software design serves as the foundation for all software engineering steps that follow regardless of which process model is being employed. [[1](#Tag01_4)]

During the design process the software specifications are transformed into design models that describe the details of the data structures, system architecture, interface and components. Each design product is reviewed for quality before moving to the next phase of software development. At the end of the design process a design specification document is produced, which is composed of the design models that describe the data, architecture, interfaces and components. [[1](#Tag01_4)]

Todo

## 4.2: Flow of the data

Todo

## 4.3: Overall Sequence diagram

Todo

## 4.4: Website Architecture

Website architecture is an approach to the design and planning of websites which, like architecture itself, involves technical, aesthetic and functional criteria. As in traditional architecture, the focus is properly on the user and on user requirements. This requires particular attention to web content, a business plan, usability, interaction design, information architecture and web design.

"Website architecture" has the potential to be a term used for the intellectual discipline of organizing website content. "Web design", by way of contrast, describes the practical tasks, part-graphic and part-technical, of designing and publishing a website.

Before I got started with anything, I had to keep in mind few design guidelines to lead the project design. The design of the website TODO

### 4.4.1: Architecture Model

There are quite a few website architecture models that can be adopted, depends on the type of website you are developing. Following are some of the architecture models; I will briefly explain three models and then the model that has been used to develop the Web Based Payroll System.

Website Architecture Models:

#### 4.4.1.1: All-in-one model



**Figure x: All-in-one Architecture Model**

*http://www.webdesignfromscratch.com/snippets/ia\_diagram\_allinone.gif*

This is the simplest possible model. Everything goes on a single Home page.

#### 4.4.1.2: Flat model



**Figure x: Flat Architecture Model**

*http://www.webdesignfromscratch.com/snippets/ia\_diagram\_monocline2.gif*

A flat pattern is where all pages are arranged as peers, and everyone is accessible from every other one. This is very common for simple sites, where there are a few standard topics, such as: Home, About Us, Contact Us, Products.

*I will be using flat model for the Web Based Payroll System so the accessing and navigation can be simpler and easier.*

#### 4.4.1.3: Hub-and-spoke / Daisy model



**Figure x: Hub-and-spoke (or Daisy) pattern**

*http://www.webdesignfromscratch.com/snippets/ia\_diagram\_hub\_and\_spoke.gif*

This model is useful for multiple, distinct linear workflows. A good example may be an email application, where you will return to your inbox at several points, e.g. after reading a message, after sending a message, or after adding a new contact.

## 4.5: Navigation Design

Todo