# REQUIREMENT MODEL FOR LOCAL GOVERNMENT ENTERPRISE INFORMATION PORTAL (EIP)

**NORHAYATI ABDULLAH**

**UNIVERSITI UTARA MALAYSIA**

**2004**

# REQUIREMENT MODEL FOR LOCAL GOVERNMENT ENTERPRISE INFORMATION PORTAL (EIP)

A dissertation submitted to the Faculty of Information Technology in partial fulfillment of the requirements for the degree
Master of Science (Information Technology),
Universiti Utara Malaysia

By
Norhayati Abdullah

**JABATAN HAL EHWAL AKADEMIK**
**(*Department of Academic Affairs*)**
**Universiti Utara Malaysia**

**PERAKUAN KERJA KERTAS PROJEK**
***(Certificate of Project Paper)***

Saya, yang bertandatangan, memperakukan bahawa
*(I, the undersigned, certify that)*

**NORHAYATI ABDULLAH**

calon untuk Ijazah
*(candidate for the degree of )* **MSc. (INFORMATION TECHNOLOGY)**

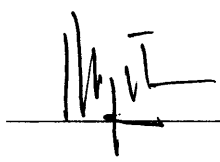
telah mengemukakan kertas projek yang bertajuk
*(has presented his/her project paper of the following title)*

**REQUIREMENTS MODEL FOR LOCAL GOVERNMENT ENTERPRISE INFORMATION PORTAL (EIP)**

seperti yang tercatat di muka surat tajuk dan kulit kertas projek
*(as it appears on the title page and front cover of project paper)*

bahawa kertas projek tersebut boleh diterima dari segi bentuk serta kandungan dan meliputi bidang ilmu dengan memuaskan.
*(that the project paper acceptable in form and content, and that a satisfactory knowledge of the filed is covered by the project paper).*

Nama Penyelia Utama
*(Name of Main Supervisor)*: **PROF. MADYA NAZIB NORDIN**

Tandatangan
*(Signature)* : 

Tarikh
*(Date)* : 31 Mac 2003

## PERMISSION TO USE

In presenting this thesis in partial fulfilment of the requirement for a postgraduate degree from Universiti Utara Malaysia, I agree that the University Library may make it freely available for inspection. I further agree that permission for copying of this thesis in any manner, in whole or in part, for scholarly purposes may be granted by my supervisor, in their absence, by the Dean of the Faculty of Information Technology. It is understood that any copying or publication or use of this thesis or parts thereof for financial gain shall not be allowed without my written permission. It is also understood that due recognition shall be given to me and to Universiti Utara Malaysia for any scholarly use which may be made of any material from my thesis.

Requests for permission to copy or to make other use of material in this thesis, in whole or in part should be addressed to:

**Dean of Faculty of Information Technology**
**Universiti Utara Malaysia**
**06010 UUM Sintok**
**Kedah Darul Aman.**

## ABSTRAK

Kajian ini dijalankan untuk mengenalpasti ciri-ciri atau elemen-elemen utama yang terkandung di dalam sesebuah portal industri (Enterprise Information Portal) bagi kerajaan tempatan di Malaysia, mengulas kaedah-kaedah yang digunakan untuk mendapatkan maklumat keperluan pengguna terhadap portal tersebut, dan mentakrifkan satu model keperluan atau "requirement model" untuk portal kerajaan tempatan. Model keperluan (requirement model) adalah penting kerana ia memberi kemudahan kepada pembangun sistem untuk memahami keperluan pengguna, menjimatkan masa, dan kos pembangunan sistem, meningkatkan kecekapan dan keberkesanan sistem, dan mengurangkan risiko kegagalan sesuatu sistem.

Kajian ini menumpukan kepada dua portal kerajaan tempatan bagi tujuan mendapatkan maklumat keperluan. Portal-portal tersebut adalah portal bagi Majlis Perbandaran Subang Jaya (MPSJ), dan portal bagi Majlis Perbandaran Ampang Jaya (MPAJ). Selain itu Majlis Perbandaran Kota Star (MPKS) turut menjadi lapangan kajian untuk mendapatkan maklumat keperluan yang lain atau untuk. mengukuhkan lagi maklumat keperluan yang telah didapati sebelumnya. Maklumat-maklumat keperluan tersebut dianalisa dan digabungkan untuk menghasilkan keperluan yang standard untuk portal kerajaan tempatan. Kajian ini walaubagaimanapun hanya menumpukan kepada keperluan pengguna awam kerana kekangan masa.

Keperluan yang dikenalpasti dari kajian dimodelkan dengan menggunakan notasi UML. Model yang dihasilkan adalah merangkumi "use-case package", "use-case diagram", dan "use-case specification". Selain itu senibina sistem atau "system architecture" turut disertakan. Teknik prototaip telah digunakan bagi mengesahkan keperluan. Prototaip yang dibangunkan pula telah disahkan dengan menggunakan teknik "system requirement testing" dan "heuristic evaluation method". Kesimpulan, dari pengujian tersebut adalah, keperluan pengguna telah ditepati dan prototaip dapat beroperasi dengan baik. Walaupun begitu. pada masa hadapan fungsi prototaip dan rekabentuk antaramuka haruslah dilengkapkan bagi meningkatkan keupayaan sistem portal yang dibangunkan. Kajian ini diakhiri dengan kesimpulan, yang menyatakan masalah dan limitasi yang dihadapi dalam melaksanakan kajian disamping mengutarakan beberapa cadangan untuk kajian akan datang.

## ABSTRACT

The purposes of this project are to identify the characteristics of local government Enterprise Information Portal (EIP), review the various methods use in capturing the requirement, and define a requirement model for local government Enterprise Information Portal. A requirement model is important as it serves as a good starting point for system developers to understand users' requirement, and it saves time, resources, and cost, reduces risk, improves effectiveness and efficiency, and reduces ambiguity.

The study focuses on two local governments portals for the purpose of requirement capture. The local governments' portals chose are Subang Jaya Municipal Council (MPSJ) portal, and Ampang Jaya Municipal Council (MPAJ) portal. Besides that Kota Star Municipal Council (MPKS) also becomes the fields of study to capture the user' requirements. The requirements captured from the MPSJ and MPAJ portals and requirements from MPKS are analysed and combined together to produce the requirements for local government in general. However due to time constraint this study only focuses on public user requirements.

The requirements captured from the study are modelled out using Unified Modelling Language's (UML) notation. The models are use case packages, use case diagrams, and use case specification. Besides that system's architecture is also included. The prototyping technique is used to validate the requirements. In this technique, a simple prototype is developed by using PHP scripting language, MySQL Database server, Apache Web Server, and Macromedia Dreamweaver MX for interface. The portal system was validated using system requirements testing and Nielsen's Heuristic Evaluation method. It was concluded that, users requirements were met and the prototype system was operating well, however the functionality and interface of the prototypes were recommended for further improvement. This project concludes by discussing problems and limitations that were encountered in completing this project, and offers a few recommendations for future development in this subject.

## ACKNOWLEDGEMENTS

My heartiest acknowledgements to several individuals who have helped to shape my perspective on this study. First and foremost, my supervisor, Assoc. Prof. Nazib Nordin, who has had a profound influence on me ever since I got to know him. He is not only a deep thinker, but also an excellent lecturer. His unique style of encouragement has provided great motivation. Prof. Nazib, ensured that I was realistic in my goals, provided excellent expert advice and clear path for the completion of this thesis.

Heartfelt thanks also go to my beloved family, for their prayers and constant support, morally and financially. Thanks dad (Abdullah), mom (Rokiah), uncle (Mansor), sisters (Nor 'Aini and Siti Hajar), and brother (Yusni).

Not forgetting my friends, Mazlina, Suhana, Mawarny, Shakirah, Fatrah, Suhaizawati, Zulaini, and Azie for their support and help. Their assistance has been a great compromise for the accomplishment of my study.

Thank to you all, great and wonderful persons.

Last but not least, my utmost thanks and gratitude to God for giving me the courage to persevere with high dedication till the accomplishment of my Masters degree study.

## TABLE OF CONTENTS

## LIST OF TABLES

## LIST OF FIGURES

CHAPTER 1

# INTRODUCTION

Internet and web technologies have changed the way we view and interact with information. In the past year, portals have received more attention from organization than any other internet technology (Finkelstein, 2000). The Enterprise Information Portal (EIP) is a concept for web-site that serves as a single gateway to a company's information and knowledge base for employees, customers, business partners, decision makers, and the general public as well. According to Shilakes and Tylman (1998), Enterprise Information Portal is application that enables companies to unlock internally and externally stored information, and provide users a single gateway to personalized information needed to make informed business decision. EIP combine software applications that consolidate, manage, analyze, and distribute information across and outside an enterprise.

In Malaysia, the use of EIP in government sector is still in the early stage. This is because based on review on related literatures; there are no specific guidelines such as requirement model for EIP and the lack of expertise. Requirement model is important to make sure that the portal will meet the expectations of customers, users, or stakeholders and are delivered on time and within the budget. Besides that, the requirement model is important to avoid project failure. According to McIntosh (2003), 78% of the IT software projects in United Kingdom failed because the requirements of the project were not defined well.

At this stand, the motivations behind this study are to identify the characteristics of Enterprise Information Portal, review various methods use in capturing the

requirement and define a requirement model for Enterprise Information Portal specifically for local government.

This chapter starts with the discussion on the problems, the research objectives, and scope. It is then followed by the significance of the study and a summary of the structure of this report.

## 1.1 Problem Statements

Nowadays Enterprise Information Portal (EIP) is widely used by many organizations all over the world. It is because EIP serves a lot of benefit to the organization. EIP provides ready access to information from the Data Warehouse or Data Marts via the Internet and Intranet. It also provides a way to integrate the many disparate systems and processes that are typically used within an enterprise.

An Enterprise Information Portal gives business users a common interfaces and access point to all data inside and outside the corporation, making location, navigation, and retrieval of information easier (Scheucher, 2000). Users can access any information appropriate to their needs and privileges, including structured and unstructured data without having to know its location or format. With just one search engine and indexing scheme, a single question can be asked to find the information rather than traveling to several destinations to find an item. Result are aggregated and presented in a context tailored to the representative's need.

With a lot of benefits as stated above, every organization should grasp this opportunity to implement portal application in their company. But in Malaysia especially in Government sector the amount of portal usage is very low. According to Local Government Department Director (The Star, July 8, 2003), a total of 41 local councils are either without a website or have an incomplete one which is no use to public while another 92 councils have five or fewer online application systems. Only 11 of the 144 local councils throughout the country are found to be established with more than five application systems in

place such as e-complaints, e-assessments, e-licensing, e-compounds, e-business, and e-community.

The report shows that most local governments are still left behind the internet technology such as portal application. Local government needs web-based system such as portal to assist their daily transaction. The 11 councils that have applied the online application were way ahead of others in term of leadership and revenue. They earn gross income up to RM 100 million a year compared to RM 2 million for others. Furthermore, the online application will make easier for public to interact with the councils.

The reasons behind the very small numbers of the local government that applied the online application such as portal are because they are not familiar with the portal concept, lack of expertise in developing the portal and also they have no reference about the portal construction to be followed.

Presently based on review on related literatures there is no specific requirement model for Enterprise Information Portal to support local government needs. As such, the portal is build according to the personal expertise. The result is different kind of portal characteristic is produced and may not meet the users or stakeholders needs. The portal then has to be redesign and this of course takes time and cost.

This problem is one of the examples of ongoing problem in software development. According to the Standish Group's CHAOS Reports (1994, 1997), the most significant contributors to project failure are related to requirements. A study conducted by Sequent Computer Systems, Inc. to 500 IT managers in the United States and United Kingdom also found that the most frequently cause of project failure was changing requirement (Computer Industry Daily, 1997).Thus, in order to minimize chances of failure, requirement management should be regarded as one of the most important phases of system development. Users' requirements should be put into attention and documented well to make sure that the proposed system will meet the users' expectation.

## 1.2 Objectives

The objectives of this project are:

1. To identify the characteristics of Enterprise Information Portal
2. To review the various methods use in capturing the requirement
3. To define a requirement model for Enterprise Information Portal

## 1.3 Scope of Project

The study focuses on 2 local governments portals for the purpose of requirement capture. The local government portal chose are Subang Jaya Municipal Council (MPSJ) and Ampang Jaya Municipal Council (MPAJ). MPSJ and MPAJ portals are chose because the councils are among the 11 local governments in Malaysia that has successfully applied online application (The Star, July 8, 2003). Besides, a field study is performed on the Kota Star Municipal Council (MPKS) to capture the users' requirement. The requirements captured from the MPSJ and MPAJ portals and requirements from MPKS users are integrated to produce the requirements for local government.

The stakeholders of the EIP for local government are the local government staff, administrators, politicians, and the public. This study focuses only on the public user requirements.

## 1.4 Significance of the Study

The requirement model will provide specific guidelines to build Enterprise Information Portal for local governments based on user needs. The Local governments also do not have to hire consultant before executing the portal project because the guidelines provided by the requirement model are beneficial in producing the portals they expected. So it is hope that in the future the portal is produced within user requirements, save time and within the budget.

## 1.5 Report Structure

This chapter addresses a few essential issues of the study. First and foremost it discusses the definitions of Enterprise Information Portal and the portal usage especially in Malaysia's government sector. It also highlights the needs to model requirements for Enterprise Information Portal for local governments as a guideline to build standard portal for every local councils. Besides that the project objectives and its scope are also stated.

As the requirement model is a subset for reference model, chapter 2 reviews the literature dealing with the reference model. The next chapter discussed about the methodology used to perform the study. In this project the methodology used to capture requirement were heuristic evaluation on portals, portals analysis, background reading, interview, and derived rich picture. The Unified Modelling Language (UML) graphical notations were used to model out the requirements for better notations. Besides that the UML's specification were also used to show the flow of events for each use cases.

The details about the model being produced are reported on chapter 4. Chapter 5 describes the prototype application while chapter 6 reports the system's validation. Finally, chapter 7 concludes the whole report, highlights the problems encountered during the study, and states some recommendations to improve this study.

CHAPTER 2

# LITERATURE REVIEW

## 2.1 Introduction to Enterprise Information Portal

Internet Portals become very popular during 1998 (Finkelstein, 2000). These provide consumers with personalized points of entry or gateways to a wide variety of information on the Internet. MyYahoo is the first personalize consumer portal which was launched in December 1996. This was the starting point for portals (Scheucher, 2000). Already at that time Yahoo helped to navigate the WWW before most people even knew about the internet. Enterprises are interested in portals because portals offer a means of pulling together all the various computer technology into a single system that enables the employees to find information regardless of its physical location (eMarketer, 2000).

According to Donegan (2000), there are two different types of portal: horizontal portals and vertical portals. Horizontal portals are also called Consumer portals, web portals, or public portals and it provides a bit of information for everyone. Example of horizontal portal is MyYahoo, Excite, Alta Vista, Lycos, and Netcenter. Vertical portals (vortals) are also called Enterprise Information Portals (EIPs) or Corporate Portals. It provides information for a particular group of specific interest (Spitzer, 2000). Examples of vertical portals or EIPs are Hummingbird portal, Oracle, Sybase, Plumtree, Engenia, and Brio Portal.

There is no unique definition of the term "portal" in the field of Information Technology (Scheucher, 2000). The word "portal" describes an entry point or homepage for accessing Internet content and services.

Shilakes and Tylman (1998) who coined the term Enterprise Information Portals in November 1998 noted that Enterprise Information Portals (EIPs) are an amalgamation of software applications that consolidate, manage, analyze, and distribute information across and outside of an enterprise. Portals should be a single gateway to information that provides interactivity and the ability to use technologies to transmit information to users through a standardized web based interface.

Murray (1999) points out that it is not enough to focus on the content. It is very important that people are not only connected with everything they need, but with everyone we need. Murray (1999) found that portal should include groupware as well as workflow and desktop application.

White (1999) divides EIPs into Decision Processing EIPs and Collaborative Processing EIPs. Decision Processing EIPs helps users organize and find corporate information while Collaborative Portals are supporting users organize and share workgroup information, such as e-mail, discussion group material, reports, memos, and meeting minutes.

Viador (1999), another early portal vendor, defines EIPs in a manner that is on the surface similar to White (1999). EIPs, according to Viador (1999) are applications that enable companies to provide access to internally and externally stored information, and offer users within and external to the enterprise a single window to personalized information needed to make informed business decisions. An Enterprise Information Portal is a browser-based system that provides ubiquitous access to vital business information in the same manner that internet content portals like Yahoo are the gateway to the wealth of content on the web.

Reynolds and Koulopoulos (1999) emphasize the user-centric focus, and work flow and task integrative functions of corporate portals. They saw corporate portals as centralizing enterprise information access in a graphically rich, application-independent interface that mirrors "knowledge centric" work flow and providing a single point of integration through the enterprise.

As a conclusion it could be said that portals should help people to work efficiently and to have an organized access to information. Users should also have the option to personalize information in a way that this information is meaningful to them.

### 2.1.1 Enterprise Information Portals Characteristics

In conventional intranets, users often have trouble in searching and finding the required information because the data is usually stored according to organizations but not according to the task that have to be carried out. The search results are sometimes not up to date and complete, and information from different sources does not match.

Enterprise Information Portal on the other hand supports users in their daily work by efficiently integrating contents, process and applications. In one model, an EIP is made up of these elements which are: access or search, single logon and security, categorization, collaboration, personalization, presentation, expertise and profiling, and application integration.

#### 2.1.1.1 Access or search

Access or search allows a user to get all the information needed in the desired context. The Enterprise Information Portal is uniquely suited to host unified search and taxonomy services that span multiple data stores, web sites, collaborative applications and other unstructured data sources (Matte, 2000). Portal-based directories can supplement keyword and conceptual searches by offering a

hierarchy of subjects that can be casually browsed, or used to narrow a search domain.

Search result can be made more relevant by matching application-specific Meta data to user preferences and by employing search agents that crawl Enterprise file systems and data stores, automatically generating taxonomic classifications for the documents they discover. An effective search will be supported by comprehensive indexing, metadata access, full-text access, and concept-based search.

#### 2.1.1.2 Single logon and Security

Security is a necessary component of the Enterprise Information Portal (Cadmus, 2000). It is important to safeguard access to sensitive and strategic information assets. Equally important is the ability to share user credentials between multiple applications even if the applications themselves aren't integrated.

Single logon eliminates the need for users to remember multiple usernames and passwords. The goal is to have a single point of access to all information sources and applications that the user normally has access to. These applications include all network folders, e-mail systems, solutions, and other password-protected accounts. EIP solutions provide a single point of access for all corporate data. The EIP is designed to collect information from a wide variety of heterogeneous data sources into a single repository (Watkins, 2002).

#### 2.1.1.3 Categorization

Valuable information, ideas, and concepts may be undiscovered through traditional access and search methods. Categorization provides a means to unearth and filter these concepts and ideas and

organize them in a meaningful taxonomy that can be navigated by the user (Hummmingbird, 2000). An EIP also categorizes all information so that it is delivered to the user within the context needed. The real benefit that categorization brings to EIP is information context (Hummmingbird, 2000). Within each organization, element such as current business practices, management initiatives, corporate history, structure and culture, available professional resources, and learning requirements build up a context for working with information.

#### 2.1.1.4 Collaboration

The collaboration element in EIP allows individuals to collaborate regardless of geographical location. Collaboration extends the role of the EIP from passive information kiosk to a new forum for organizational interactions (Hummmingbird, 2000).

In other words a portal should become a new forum for organizational interactions, for example between employees, customers, partners, and other stakeholders. As corporations expand and as the speed and demand of business increase, interaction becomes simultaneously more difficult and more important. Users are frequently geographically and departmentally separated so companies need a common, central place where distributed teams can work together. This collaborative environment should be designed to overcome the shortcomings of working in that distributed environment and focus on users not the application used to complete task.

By enabling this level of interactivity, EIP solutions can dramatically reduce the time required for such things as customer service and improve stakeholder relations. Within the organization, EIP solutions address several levels of collaboration. Similar to the functionality of Microsoft Exchange or Lotus Notes Platforms, the

EIP can provide enterprise-wide collaboration. In addition to this organizational level, EIP can enable collaboration at the project or group level, an interest level, or a functional group level.

#### 2.1.1.5 Personalization

Personalization extends beyond the content provided by roles. Personalization can determine the page layout, the look, and feel of the portal, and even which information users receive and how they receive it. The personalization feature enhances productivity and individual information management (Matte, 2000).

Personalization can be defined in three ways: at the administrator level, the user level, and automatically through predictive technology. Administrators can define personalization for each user by changing the design of the portal structure for different users. Administrators can define the roles and work sets for different users, and can determine how to configure the layout of the portal to be most effective for users. For example, a customer may have portal access only through a WAP phone or Pocket PC. So, the administrator may ensure that portal pages available to those customers are only in those formats.

#### 2.1.1.6 Expertise and Profiling

Expertise and profiling is essential for the collaboration element of an EIP. Individuals within an enterprise are profiled according to their experience and competencies. If an individual needs to collaborate with others, he can choose those that are qualified for the project.

#### 2.1.1.7 Application Integration

Application integration allows individuals to deliver, access, and share information regardless of applications used. A portal must be able to integrate people and there are three types of information: structured, unstructured, and process. Structured information includes enterprise applications, legacy systems, data warehouse, and business intelligence. Unstructured information on the other hand includes document collections, groupware applications, message archives, web pages and links, and online information feeds. Last but not least process information includes alerts, real time collaboration, process cues or actions and object interaction.

Business Intelligent applications such as Data Marts, Data Mining, OLAP, Query and Reporting products aid organizations in discovering actionable business information in their structured data system. On the whole these are large, complex, and often expensive products, increasingly fronted by web browsers which serve to increase the potential for information distribution. Together, this products segment provide access to enterprise data, presentation services, business analytic services, multi-dimensional views and complex pattern analysis and trending capabilities, all will increased user efficiency and less impact on production system.

### 2.1.2 The Enterprise Information Portal Advantages

The Enterprise Information Portal will serve various user groups like staff members, customers, suppliers, and other entities. Each user will have its own profile and authority to see information from the portal depending on authority (Finkelstein, 2000). The users will have choice to operate on English version or local language version. The EIP combine, manage, analyze, and distribute information among the entities.

With EIP, organizations can perform more efficiently and accurately, save time and money, and gain significant competitive advantage (Roberts-Witt, 1999). EIP bridges the information gap between an enterprise and its business channels, and linking it with partners, alliances, and customers.

Enterprise Information Portal is an advanced security model to ensure the integrity of information, applications, and resources (Matte, 2000). Enterprise Information Portal is reaching all the way out to individual customers, providing them with a personalized view of the organization or enterprise.

## 2.2 Requirement Analysis

According to Institute of Electrical and Electronics Engineers (IEEE) standard 610 (Macaulay, 1996), the term requirement refers to a condition or capacity needed by a user to solve a problem or achieve an objective, a condition or capability that must be met or possessed by a system or system component to satisfy a contract, standard, specification, or other formally imposed documents, and a documented representation of a condition or capability as an above statements.

Gunter et al. (2000), stated that requirement indicate what the customers need from the system, described in terms of its effect in the environment. Requirement stage is important because it helps to understand the new needs and to identify how they can be satisfied (Macaulay, 1996). For example a system could not be design before the future situation about the system needs and usage is known.

Lethbridge and Laganiere (2001) define requirement as a statement about what the proposed system will do that all stakeholders agree must be made true in order for the customer's problem to be adequately solved. "A requirement is a statement" means that each requirement is relatively short and concise piece of information, expressed in fact. It can be written as a sentence or can be

expressed using some kind of diagram. Lethbridge and Laganiere (2001) added that, a requirement says something about the task the system is supposed to accomplish but not describe the domain or how the system will be implemented. In a simple word requirement could be said as what the users need from the new or proposed system in order to solve their problem of the current system. So with requirements, the system developed will satisfy the user needs and will not have problem or have to redesign.

Requirement can be divided into two major types: functional requirement and non-functional requirement (Sommerville, 1995). Functional requirement describe what the system should do, i.e. the services provided for the users and for other systems. The functional requirements should include everything that a user of the system would need to know regarding what the system does. Functional requirement should also include everything that would concern any other system that has to interface to this system.

Non-functional requirement is a requirement that relates to system features such as performance, maintainability, and portability. (Bennett et. al, 2002). Examples of non-functional requirements are desired response time for updating data in the system or retrieving data from the system, anticipated volumes of data, either in terms of throughput or what must be stored and last but not least security considerations. According to Lethbridge and Laganiere (2001), non-functional requirement can be divided into three groups. The first group reflect the five qualities attributes which is usability, efficiency, reliability, maintainability, and reusability. The second group of non-functional requirement constrain the environment and technology while the last group constrain the project plan and development methods.

Several techniques can be use to gather and analyze requirement from user. Some of the techniques are background reading, interviews, heuristic evaluation, and prototyping. These techniques will be discussed further in Chapter 3.

## 2.3 Requirement Model and Reference Model

In the previous part the term requirement have been defined as what the user needs from the proposed system in order to solve their problem of current situation or works. The users' requirements need to be documented properly so that everyone especially the designer, developer, or user itself will understand the requirement to execute the next process.

Requirement model is one of the techniques used to model out the requirement. Requirement model is a model that gives a complete view of the requirement for a particular system (Bennett et. al, 2002). In projects that use UML, a requirement model will consist of a Use Case Model, which comprises use case diagrams, use case descriptions, and prototypes.

Requirement model is a subset of reference model. A reference model is a framework for understanding significant relationships among the entities of some environment, and for the development of consistent standards or specifications supporting that environment. A reference model is based on a small number of unifying concepts and may be used as a basis for education and explaining standards to a non-specialist (Gunter et al., 2000). A reference model is important because it serves as a good starting point for system developers to understand users' requirement (Arasan, 2003). Many reference models for a variety of subjects, industry and discipline have been developed over the years. As the requirement model is a part of reference model, these previous literatures are beneficial to do this project. Below are related literatures on the reference models that have been produced and used in various categories:

### 2.3.1 CRM System Reference Model for Small and Medium Enterprises

CRM System Reference Model is a model for an integrative Customer Relationship Management System of IISME Consortium. This model is specializing on the requirement of small and medium enterprises (SME).The CRM approach was translated into a model that allowed interaction to be organized with the customer across all communication

channels, as well as to integrate the sales and marketing departments effectively in the overall structure of an SME (IISME Consortium, 2000).

The IISME Consortium CRM System reference model provides users and developers of a CRM system with an overview of the important aspects to be considered before a CRM type system is introduced in an organization. A detailed catalogue of the system's functionalities serves as a useful checklist for requirement analysis of a typical CRM project in an SME. Relevant areas of the model can be identified, transferred, and customized to meet specific needs of similar projects (IISME Consortium, 2000). This literature provides a clear reference on how a reference model can be developed and important factors to be considered before embarking on a reference model development project.

### 2.3.2 Business Reference Model for Federal Enterprise Architecture (FEA)

The Office of Management and Budget of US Federal Government have come out with Business Reference Model (BRM) which is a business-based framework for Government-wide improvement and provides the Federal agencies with a new way of describing, analyzing, and improving the federal government and its ability to serve the citizen (FEAPMO, 2002).

This model was important as it provides examples of how interrelated reference model can be used to achieve a strategic goal that is wider in scope. The BRM has been validated and is being store in an automated tool to ensure its accessibility, usability, and currency. Through this effort, the different agencies are encouraged to take maximum advantage of the work in their strategic performance improvements effort.

### 2.3.3 Strategic E-government Concepts Reference Model

Strategic E-government Concepts Reference Model involved the E-government Competence Center at Berne University of Applied Science,

Switzerland (Brucher, 2002). The center developed a six-stage reference model to assist the town of Luzern and the canton of Berne in their definition of an overall e-government strategy. The objective of the reference model was to support the planning process from a strategic to an operational level, so that the definition of an overall concept and the final derivation of specific implementation measures can be done under the same guidelines.

The major outcome of these projects was a six-stage reference model to assist in the planning process of e-government. The stages cover vision, strategy, concept, project-portfolio, projects, and measures. The reference model is deemed important as in the long run; an overall blueprint saves time and money. It allows planners to consider alternatives, select the best option, work out the details, and achieve agreement before anyone starts building the application.

Brucher (2002) believes that it is much less costly to use a model than it is to modify an e-government application after it has been assembled. More importantly, a good model documents the application's structure and simplifies modifying it later.

### 2.3.4 Reference Model for Electronic Business Messages based on XML

Another reference model project was the one developed by X12C's EDI Architecture Task Group. The reference model proposes features of XML that can be used in e-business messages, and offers design principles for organizing business data into modularity flexible messages using XML.

This reference model offers companies a way to reuse their business data across different electronic documents in their own industries and provides interoperability across industries (Cover, 2002) and also gives guidelines for companies and industries on writing electronic business messages.

### 2.3.5 Public Transport Data Model (TRANSMODEL) reference model

TRANSMODEL is a pre-standard reference data model for Public Transport operations (European Prestandard, 1996). It has been developed by a European Team. It was developed for the public transport companies and other providers of services related to the process of passenger transportation and information.

TRANSMODEL increases the efficiency of transport operations by supporting them with a more secure and reliable Information System. The reference model, which was initially designed to document the information needs of a public transport company, can be extended to serve as a starting point and reference for the definition of similar database schema needed for the physical implementation of data storage systems, to be used by applications directly, or for exchange of data between different applications.

## 2.4 Conclusion

This chapter discusses some reviews related to Enterprise Information Portal, requirement, requirement model, and reference model. Although this project is about requirement model but the reviews on reference model is highlighted because requirement model is a part of the reference model. So, this literature will provide a good guidance in understanding and designing the model for this project.

CHAPTER 3

# RESEARCH METHODOLOGY

## 3.1 Introduction

The purpose of this chapter is to describe the research methodology, the methodology used to capture and model users' requirement, and validation techniques used to validate the prototype system against users' requirements. There are seven main activities which have been followed to accomplish the study. The activities are review on related literatures, requirement capture, derived rich picture, UML's graphical notation, UML's specification, requirement validation, and system validation.

## 3.2 Review on related literatures

As a starting point of this study, related topics and information regarding Enterprise Information Portal (EIP), requirement, requirement model, and reference model were gathered for literature review.

## 3.3 Requirements Capture

Requirements capture is a process to find out from users what they require in a new information system (Bennet et. al, 2002). Various methods are used to capture requirements for local government EIP. The methods are as follows:

### 3.3.1 Heuristic Evaluation on Portals

Heuristic evaluation is a discount usability engineering method for quick, cheap, and easy evaluation of a user interface design (Nielsen, 1994). Heuristic evaluation is the most popular of the usability inspection methods. Heuristic evaluation is done as a systematic inspection of a user interface design for usability. The goal of heuristic evaluation is to find the usability problems in the design so that they can be attended to as part of an iterative design process. Heuristic evaluation involves having a small set of evaluators examine the interface and judge its compliance with recognized usability principles.

In this study, the heuristic evaluation had been conducted on the portal of Majlis Perbandaran Subang Jaya (MPSJ) and Majlis Perbandaran Ampang Jaya (MPAJ) in order to identify usability problems in the portals. Before the evaluation, lists of web site heuristics documents were prepared. This heuristics were adapted from Jacob Nielsen's heuristics and evaluation checklist from Pierotti (1994).

Next, the suitable evaluators were identified for the evaluation session. The evaluators must be expert in usability. The evaluators should preferably be usability specialist, graphic or web designer or system developer with minimum of four years experience in developing or designing web sites. Three evaluators were used in the study. This is based on research done by Nielsen and Landauer (1993) where they determined that the maximum benefit-cost ratio can be achieved when using three and five subjects. Nielsen (1995) found that "many projects were able to use heuristic evaluation with a smaller number of evaluators". There is no need for test to be recorded in a lab as note takers able to capture all the essential findings by hand.

#### 3.3.1.1 Data Collection Procedures

Each evaluator evaluated two portals which means that they have to evaluate in two sessions, one session for each portal. Every session for individual evaluator lasted about one hour. The evaluator browsed the portal to find the usability problems based on Nielsen's ten heuristic and review checklist of web usability adapted from Pierotti (1994). Each evaluator was observed in separated evaluation session when each of them browsed the portal. The evaluators were not provided with scenarios or specific tasks to perform for the evaluation. This is to allow evaluators to have the freedom to browse any pages in the portal and perform any tasks that they had in mind. Each evaluator had browsed through the portal at least twice: once to familiarize with the interface and second time to inspect the elements against the list of heuristics (refer table 3.1). A review checklist of web usability was given for each principle (refer Appendix B) The review checklist will help the evaluator to understand the scope of heuristic. The evaluator also can add comment to the usability problems identified.

| Nielsen's Heuristics | Web Adaptation |
|---|---|
| 1. Visibility as system status | Are the URL line and the status line used to provide effective feedback? |
| 2. Match between system and the real world | Does the web site reflect users' language, tasks, and intentions? |
| 3. User control and freedom | Can users select and sequence tasks? Can they easily return to where they were if they choose an inappropriate path? |

| | |
|---|---|
| 4. Consistency and standards | Do the web pages work with different browsers? Can people resize windows or adjust browser options without compromising information or the task? |
| 5. Error prevention | Does the Web site give enough directions and information so that users can find desired pathways and complete desired operations? If an error does occur, can users recover easily? |
| 6. Recognition rather than recall | For the web, this heuristic is closely related to system status. If users can tell where they are by looking at the current page, they are less likely to get loss. Good labels and descriptive links are crucial for recognition. |
| 7. Flexibility and efficiency of use. | Do links take users where they expect to go? Are images and data loaded as effectively as possible? |
| 8. Aesthetic and minimalist design | Are the best media and screen images used to convey a message? |
| 9. Help users recognize, diagnose, and recover from errors. | Every error message should offer a solution or a link to a solution. For example, if a search yields no hits, do not just tell the user to broaden the search: provide a link that will broaden the search. |
| 10. Help and documentation. | Is the web site self-documenting? |

Table 3.1: Nielsen's Ten Heuristics Adapted for the Web

After getting the answers and comments from the evaluators, the author analyzes the answers and rated the heuristic principles

according to its severity. The comments are also important because it helps to identify and explain the usability problems identified. The answers given were calculated and rated based on the answers of "Yes" and "No" only. The author assigned severity ratings based on five-point scale. Each heuristics is rated from scale 0 to 4 as follows:

- Scale 0: Totally disagree that this is a usability problem
- Scale 1: Cosmetic problem only (Not important to fix it)
- Scale 2: Minor usability problem (only be given low priority)
- Scale 3: Major usability problem (important to fix it, therefore should be given high priority)
- Scale 4: Usability catastrophe (very serious problem and should be fixed immediately)

The rating is allocated to the counted number of answer "Yes". The following criteria are set for severity ratings for each heuristics:

- 100% from the answers in the checklist is "Yes": rating scale 0
- 75%-99% from the answers in the checklist is "Yes": rating scale 1
- 50%-74% from the answers in the checklist is "Yes": rating scale 2
- 25%-49% from the answers in the checklist is "Yes": rating scale 3
- 0%-24% from the answers in the checklist is "Yes": rating scale 4

Discussions with the evaluators were also done to identify clearly about the usability problems and their suggestion to improve the usability of the portal.

### 3.3.2 Portals Analysis

The portal of Majlis Perbandaran Subang Jaya (MPSJ) and Majlis Perbandaran Ampang Jaya (MPAJ) was chosen to perform the portal analysis. This method requires the author to browse each portal and list out the characteristics or components found in each portal. The results found on each portal were compared and analyzed to find the most important component or characteristics of local government portal.

### 3.3.3 Background reading

Background reading technique was used to gain general understanding about the field of study which is Majlis Perbandaran Kota Star or MPKS. Background reading was done by reading MPKS annual reports, organization charts, and job description. After this method had done, the author can visualize in general the activities of the local government staff and community by referred to the business objective of every department, and job description of the staff inside the department. It also gives an idea of the overall situation so that the author could prepare related questions in interview session.

### 3.3.4 Interview

Interview is an important method to gain better understanding about the current situation. In this project the interview was conducted with the MPKS Public Relation Officer (PRO) because the main objective is to understand the function of MPKS that related to the public user. To achieve the objective a set of related questions were prepared before the interviews take place. During the interview session the questions were asked according to the prepared plan. Besides that other explanations were also put into consideration. Information obtained was written up in notes properly. Interview session was done at the Public Relation Officer room in MPKS building and it took about 2 hours.

## 3.4 Derived Rich Picture

The results of portal analysis or the requirements captured from portal analysis and the requirements captured from the field of study (MPKS) were combined together to make sure that the requirement meets each other. New requirements that are important were added to meets the actual situation. The rich picture was constructed to show graphically the requirement obtained. The rich picture is cartoon-like representation that identifies all the stakeholders, their concerns, and some of the structure underlying the work context (Monk and Howard, 1998). Rich picture originated in the Soft System Methodology (SSM). The advantage of having a rich picture that is comprehensible to the people who have given the information. The informer or user can review back the rich picture produced and correct mistakes of interpretation and add new information.

### 3.4.1 Rich Picture Components

Drawing a rich picture is an iterative process of understanding and refining that understand (Monk and Howard, 1998). The three most important components of a rich picture are structure, process, and concerns. Structure refers to aspects of the work context that are slow to change. These might be things such as the organizational hierarchy of a firm, physical equipment, and so on. Most important it includes all the people who will use the proposed system.

In this study the structures refers to the main stakeholders of the local government EIP. The stakeholders include public users. Process refers to the transformations that occur in the process of the work. This transformation might be part of a flow of goods, documents, or data. In this study process refers to the action of each actor in achieving the data or other benefits from the EIP. In other ways process also shows what benefits does EIP provides to the users. Lastly, concerns are the most useful component because it captures more clearly the idea of a particular individual's motivation for using the system. For example the concerns of

the community are about the complaints, compounds and so on. The concerns are represented in thought bubbles.

### 3.4.2 Steps on Drawing Rich Picture

Monk and Howard (1998) stated that there is no correct way of drawing a rich picture. There are as many styles as analysts and the same analyst will find different styles useful in different situations. When drawing a rich picture for this project the first step was sketching in the middle of large sheet of paper some figure that represents the primary operator. The MPKS symbol was used in the study to represent the local government in a whole. Next, the stakeholders that directly influence the operators work were pictured along with the elements of structure needed to explain the process of work. The stakeholders for this study were public users. When the major structures and processes have been added, then the concerns were addressed in a thought bubble format.

## 3.5 UML's Graphical Notation

The Unified Modelling Language's (UML) graphical notations were employed in the study to model the users' requirements. UML is a language used to specify, visualize, and document the artefacts of a system under development. It represents the unification of the Booch, OMT, and Objectory notations, as well as the best ideas from a number of other methodologies (Quatrani, 2000). The models used in the study were including use case package diagrams and several use cases.

Packages are a way of organizing model elements and grouping them together (Bennet et. al, 2002). Packages help to organize the elements in the models so that the models are easily to understand. Use cases are descriptions of the functionality of the system from the users' perspective (Bennet et. al, 2002). Use case diagrams are used to show the functionality that the system provide and to show which users are communicate with the system in some way to use that functionality.

The use case model is part of requirements model (Jacobson et al., 1992). Use case describes the interaction as the user sees it, and is not a definition of the internal process within the system, or some kind of program specification. Use case diagrams shows three aspects of the system: actors, use cases and the system or sub-system boundary. Use case diagrams in UML could be used to visualize the behaviour of a system, subsystem, or class so that users can comprehend how to use that element, and the developers can implement that element.

### 3.6 UML's Specification

The Unified Modelling Language's (UML) specifications were also used in the study to show the flow of events for each use cases. Specification provides a textual statement of the syntax and semantics of a specific building block (Booch et. al, 1999). Gunter et. al (2000), stated that specifications provide enough information for a programmer to build a system to satisfy the requirements. The UML's graphical notations were used in the study to visualize a system while the UML's specifications were used to state the system's details. The UML's specifications provide a semantic backplane that contains all the parts of all the models of a system, each part related to one another in a consistent fashion (Booch et. al, 1999).

### 3.7 Requirement Validation

A Prototyping technique was used to validate the requirement model. A Simple prototype was developed by using Macromedia Dreamweaver MX, PHP Scripting Language, MySQL for database, and Apache server. The sets of requirement captured in the model were applied in the prototypes. For this prototype the functionality is limited and the prototype is concentrate on some specific aspects only.

## 3.8 System Validation

In this project, the system (portal) was tested and validated by using system requirement testing and heuristic evaluation method.

### 3.8.1 System Requirement Testing

Employing this method, each user's requirements are tested. Testing a system (portal) not only uncovers errors in the system, but also demonstrates that the system appears to be working according to user's specification and that performance requirements appear to be satisfied.

### 3.8.2 Heuristic Evaluation on MPKS Portal

The Heuristic evaluation techniques were also applied to evaluate the usability of MPKS portal from public users' perspective. Three evaluators were choosed to evaluate the MPKS portal. Please refer to section 3.3.1 for details description on how this technique was conducted.

## 3.9 Conclusion

Methodology provides general principles that guide a practitioner to the choice of a particular method suited to a specific task or project. The requirement model in this project was produced after the users' requirements have been captured by using five techniques. The techniques are heuristic evaluation on portals, portals analysis, background reading, interview and constructing rich picture. The requirements are model out in UML graphical notations and UML specifications. Prototyping techniques were used to validate the requirements model while for system validation, system requirement testing and heuristic evaluation technique were implemented.

CHAPTER 4

# FINDINGS AND DISCUSSION

## 4.1 Introduction

This chapter reports the findings of the three research objectives: the characteristics of Enterprise Information Portal for local government, the requirements captured by various methods, and the requirement model for local government EIP. The users' requirement will be described in detail, and documented using UML notations. Recommended architecture and functionality of the system will also be illustrated.

## 4.2 Characteristics of Local Government Enterprise Information Portal (EIP)

Enterprise Information Portal supports users in their daily work by efficiently integrating contents, process and applications. In a public view, the characteristics of local government Enterprise Information Portal (EIP) identified from the study are:

### 4.2.1 Local Government General Information

The portal contains general information about the local government. The information is including local government background, profile, history, departments, functions, and corporate information.

### 4.2.2 Application Services

The portal provides several local government online services to the user. The services are:

- Compound service: user can check their compound status online and pay the compound online
- Tax service: User can check their tax and pay online
- Renting service: User can rent public facilities such as hall, fields, and court via online application. Specifications about the service are also provided.
- License service: User can apply and check their application status via online system. Types of license and license requirement are also included.
- Complaint Service: User can make complaint directly via online application.
- Tender service: Offer tender and provide specification to the user.
- Downloaded items: User can download forms to apply the service manually.

### 4.2.3 Shared Contents

The portal provide link to other local governments web site or other related organization website, thus help users to find information easily.

### 4.2.4 News and advertisement

The portal keep users inform about the latest news or activity related to the local government. Besides that advertisement such as job vacancy and promotion also permitted here.

### 4.2.5 Event Calendar

The local government activities plan in a year is stated in the event calendar.

### 4.2.6 Collaboration

The collaboration elements in the portal allows individual to collaborate regardless of geographical location. Collaboration extends the role of the EIP from passive information kiosk to a new forum for organizational interactions. The collaboration elements in the portal are forum and chat.

### 4.2.7 Directory

The directory helps users to find the contact number of the local government staff.

### 4.2.8 Unified Search

Access or search allows a user to get all the information needed in the desired context. The EIP is uniquely suited to host unified search that span multiple data stores, web sites, and other unstructured data sources.

### 4.2.9 Single logon and security

Single logon eliminates the need for users to remember multiple usernames and passwords. The goal is to have a single point of access to all information sources and applications that the user normally has access to. Security is an important component to safeguard access to sensitive and strategic information assets.

### 4.2.10 Familiar User Interface

The portal uses familiar Explorer-like interface to easily navigate through data, files and application

## 4.3 Results: Various Methods Used to Capture Requirements

Various methods have been used to capture the requirement. Each method contributes significant result to build the requirement model for local government EIP.

### 4.3.1 Portal Evaluation Result

The results (Table 4.1 and Table 4.2) obtained from the heuristic evaluation on MPSJ and MPAJ is a list of all usability problems that have been identified. Please refer Appendix A for details result of each evaluator.

| Heuristics | Results of First Evaluator | Results of Second Evaluator | Results of Third Evaluator | Mean of Result | Severity Rating |
|---|---|---|---|---|---|
| Heuristics 1 | 75.86 % | 60.0 % | 54.55 % | 63.47 % | 2 |
| Heuristics 2 | 68.18 % | 60.0 % | 75.0 % | 67.73 % | 2 |
| Heuristics 3 | 73.33 % | 56.52 % | 73.68 % | 67.84 % | 2 |
| Heuristics 4 | 76.09 % | 74.47 % | 70.59 % | 73.72 % | 2 |
| Heuristics 5 | 84.21 % | 52.63 % | 52.63 % | 63.16 % | 2 |
| Heuristics 6 | 76.92 % | 66.67 % | 73.33 % | 72.31 % | 2 |
| Heuristics 7 | 75.68 % | 68.42 % | 75.0 % | 73.03 % | 2 |
| Heuristics 8 | 40.0 % | 40.0 % | 81.25 % | 53.75 % | 2 |
| Heuristics 9 | 81.82 % | 60.0 % | 83.33 % | 75.05 % | 1 |
| Heuristics 10 | 75.0 % | 63.16 % | 45.0 % | 61.05 % | 2 |

Table 4.1: Analysis of Results and Severity Rating on MPSJ portal

The MPSJ data in Table 4.1 has shown that Heuristic No. 1 until No. 8 and No. 10 has severity rating of scale 2 while Heuristic No. 9 has severity rating of scale 1.

| Heuristics | Results of First Evaluator | Results of Second Evaluator | Results of Third Evaluator | Mean of Result | Severity Rating |
|---|---|---|---|---|---|
| Heuristics 1 | 82.76 % | 65.52 % | 75.0 % | 74.43 % | 2 |
| Heuristics 2 | 70.83 % | 85.0 % | 87.5 % | 81.11 % | 1 |
| Heuristics 3 | 89.47 % | 57.14 % | 72.22 % | 72.94 % | 2 |
| Heuristics 4 | 82.98 % | 93.02 % | 95.74 % | 90.58 % | 1 |
| Heuristics 5 | 77.78 % | 85.0 % | 84.21 % | 82.33 % | 1 |
| Heuristics 6 | 83.33 % | 69.23 % | 83.33 % | 78.62 % | 1 |
| Heuristics 7 | 77.14 % | 90.0 % | 84.21 % | 83.78 % | 1 |
| Heuristics 8 | 87.50 % | 78.57 % | 50.0 % | 72.02 % | 2 |
| Heuristics 9 | 83.33 % | 91.67 % | 83.33 % | 86.11 % | 1 |
| Heuristics 10 | 82.61 % | 66.67 % | 66.67 % | 71.98 % | 2 |

Table 4.2: Analysis of Results and Severity Rating on MPAJ portal

The MPAJ data in Table 4.2 also shown that Heuristic No. 1, No. 3, No. 8, and No. 10 has severity rating of scale 2 while Heuristic No. 2, No.4, No. 5, No. 6, No. 7, and No. 9 has severity rating of scale 1. Those problems rated with scale 1 and 2 are considered as minor usability problem and not important to be solved.

However, the experts had identified 5 usability problems in the Majlis Perbandaran Subang Jaya (MPSJ) and Majlis Perbandaran Ampang Jaya (MPAJ) portal as follows:

**Problem 1: Inconsistent use of Malay and English Language in MPSJ portal**

There is a mixture of English and Malay Language used in the portal. For example, at the portal main page, some of the icon is labeled in Malays and some are labeled in English or the mixture of Malays and English.

**Problem 2: Too much icons and option in the MPSJ portal**

There are too much icons and option especially at the main page of the portal. It makes the page look crowded and complex.

**Problem 3: Too much animation in MPSJ and MPAJ portal**

There is too much visual animation in MPSJ and MPAJ portal. It will slow down the download time of the web page. Too much animation will also irritate certain user.

**Problem 4: Invisible Help function in MPSJ and MPAJ portal**

There is no special menu or key labeled HELP in both portals. Users will face difficulty in getting help.

**Problem 5: Cannot cancel out of operations in MPSJ and MPAJ portal**

Users cannot cancel out of operation in progress when they choose system function by mistake

The evaluation result presented above shown that it is not important to fix the problem and it should be given low priority. However the problem identified by the expert should put into consideration to increase the portal usability. Therefore the problems identified became a part of requirement for local government EIP.

### 4.3.2 Background Reading Result

To conclude this technique the author understood that there are 18 departments in MPKS and every department have its own business

objective. The staffs in each department perform specific functions that comply with the departments objectives.

### 4.3.3 Interview Result

The results of interview session are the author gain better understanding about the functions and real situation inside the local government. As the scope of this project is to capture the requirement for public users of local government, the question asked is mainly about the service provided for this group of users and problem faced by the public user with the existing system. The overall results of the interviews are concluded as the following:

- Services provided for public users are application for business license, rent stalls, hall, fields, and other public facilities, tax and compound service, tender service, and also health service.
- The services are performing manually according to department. For example business license is managed by Business Department of MPKS.
- Some services are performing by specific system but the system is not centralized.
- The public user always reported that they find difficulties in finding the appropriate departments to complete their task since there are 18 departments in present and the departments are separated.
- The interviewee emphasis that the local government needs a new approach to overcome the difficulties of the public user.

### 4.3.4 Rich picture

The rich picture is derived by combining the requirement captured from portal analysis and evaluation result, background reading and interviewing result. Figure 4.1 shows the rich picture obtained.

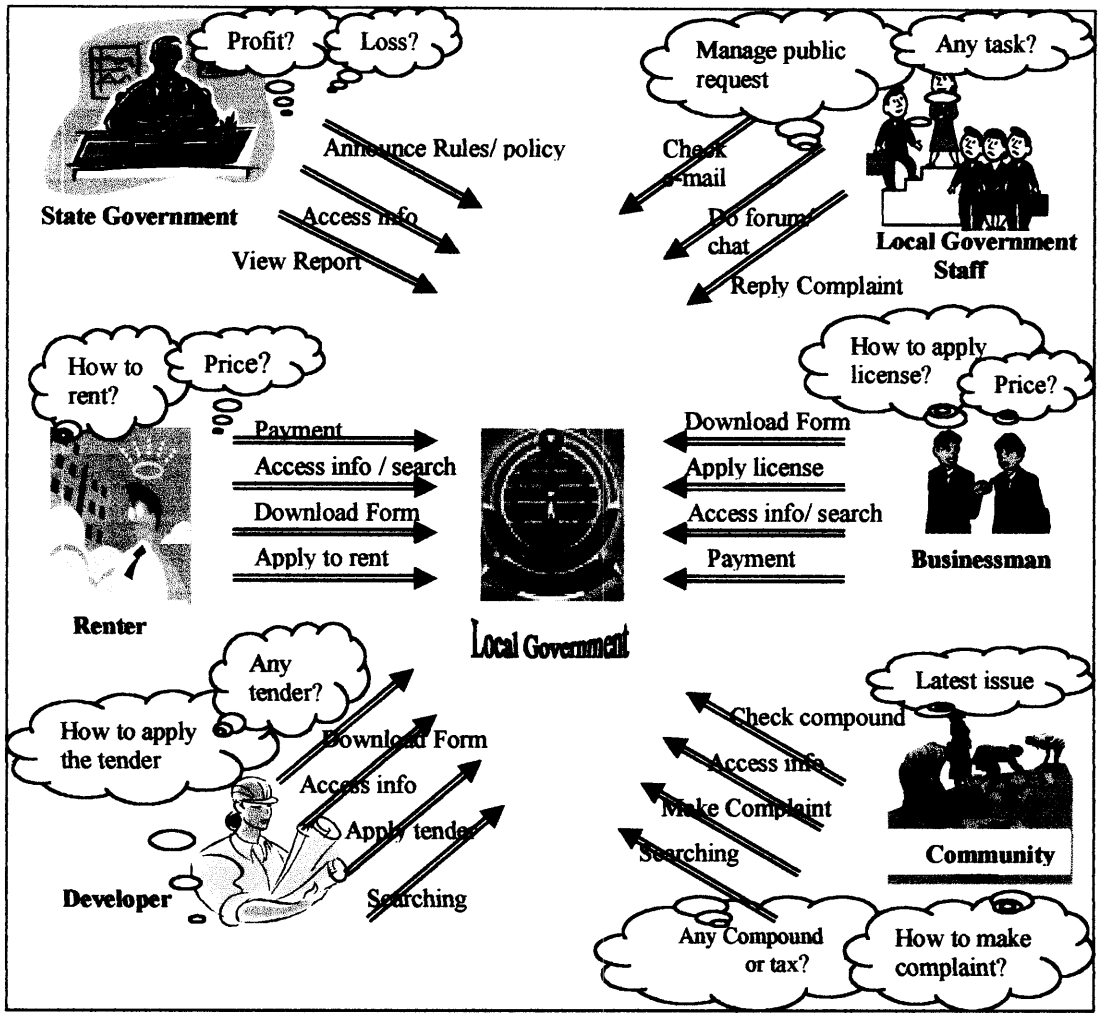



Figure 4.1 : Rich picture of local government EIP

The rich picture shows the requirement captured for several users. From the rich picture, the public users which are businessman, community, developer and renter have their own concerns. For example the community is concern on how to make complaint and check compound. The arrows represent the functions required by the user from the portal. However in the UML model the different public users are group together and identified as public user.

## 4.4 Requirement model for local government EIP

Following are the elements of requirement model for local government EIP. The models are consisting of use case packages, use cases, use case specification, and system architecture. The public user functional requirements are model out in use case packages and use case diagram. The flow of events for each use cases are describe in specification section. Besides that the description of each requirement are also included in requirement list section.

### 4.4.1 Use case Packages

Figure 4.2 depicts an overview of the requirement model's packages diagram, based on the UML syntax for local government portal system. It describes the system border, the modules of the system, and their interaction with external actors. Actors represent external entities that interact with the system.

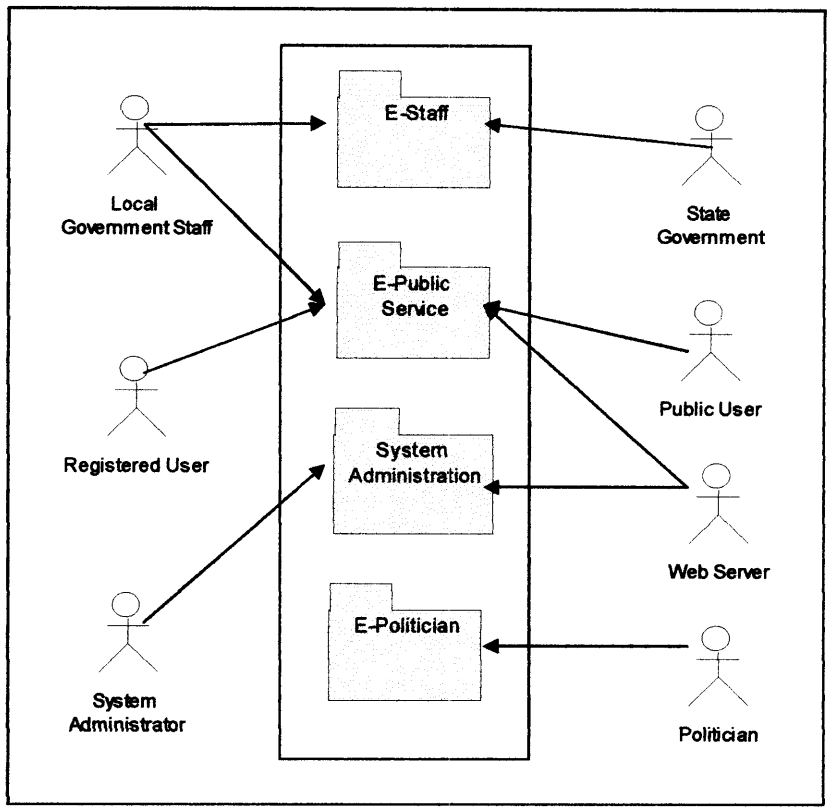



Figure 4.2: Use case Packages: Local Government EIP

The study found that there are four main modules in local government EIP. The modules are staff modules, public modules, politician modules and system administration modules. The staff module generally covers functions performs by the local government staff and state government staff, the public module covers functions performs by public users (examples of public users are renter, businessman, community, and developer), the politician module covers functions performs by politician and system administration modules covers functions perform by system administrator. However, this study focuses on public module only.

### 4.4.2 Use Case Diagram

Use cases model a dialogue between an actor and the system. They represent the functionality provided by the system; that is what capabilities will be provided to an actor by the system (Quatrani, 2000).

Figure 4.3 depicts the uses cases in public modules. There are six actors and 22 use cases in the diagram. Some of the use cases are packaged into similar types or groups of application. There are five packages named as e-checking package, e-application, e-payment, e-complaint, and online management package. The main actor of this module is public user. There are two types of public user; registered and unregistered public user.

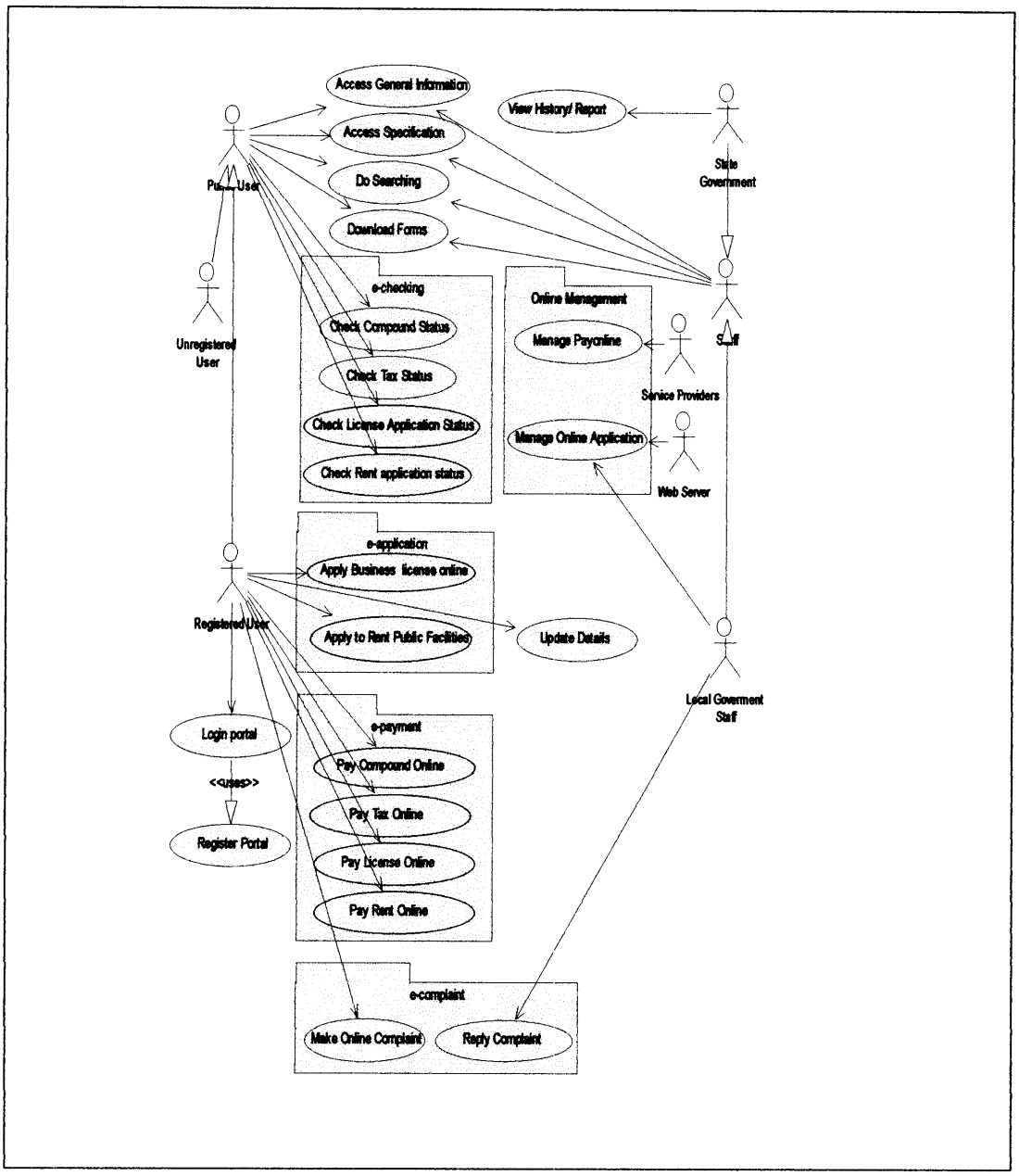



Figure 4.3: Use Case Diagram for E-Public Service Package

Figure 4.3 shows that all public users can access general information about the local government, access service specification, do searching, download services forms, check compound status, check tax status, check license application status, and check rent application status. The check compound function need the user to key in the vehicle plat Number. The systems then will find the input data in database. If the plat number is not matching in the database system it means the user do not have any compound record.

The same method is use to check license application status, and rent application status but the input data to check license application status and rent application status is the application forms serial number.

Registered user is public user that has register to the portal system. Registered user inherits all the public users' functions. The different function performing by the registered users are apply business license online, apply to rent public facilities online, pay compound online, pay tax online, pay license online, pay rent online, and make online complaint. All these functions need the user to login first because only authorized users are allowed to use the applications. Besides that the users are also allowed to update their details if there are any changes.

### 4.4.3 Requirement list

Table 4.3 depicts the list of requirements, use case, and actors involved.

| No. | Requirements | Use Case (s) | Actor (s) |
|---|---|---|---|
| 1 | To record user name, password, full name, address, and contact details for each registered portal user | Register portal | Public User |
| 2. | To authenticate the portal user in order to prevent unauthorized access to information and functions. | Log in portal | Registered user |
| 3. | To browse the portal content (View local government general information) | Access General Information | Public User |
| 4. | To view the service specification such as rent specification, tender specification, and license specification | Access Specification | Public User |
| 5. | Update user's record such as change the address | Update Details | Registered user |
| 6. | To provide complete history of business activities that have been done related documents, task, and | View History/ Report | State Government |

| | | | |
|---|---|---|---|
| | appointment) | | |
| 7. | To search the portal meta-content | Do Searching | Public user and Staff |
| 8. | To download local governments application forms (eg: license form) | Download Forms | Public user and Staff |
| 9. | To send complaint to local government. The complaint details such as person name, e-mail address, contact number, and complaint message will be recorded into local government database for further action. | Make Online Complaint | Registered user |
| 10. | To give feedback for the users' complaint based on the record in the database. | Reply Complaint | Local Government Staff |
| 11. | To check whether the user have any compound record | Check Compound Status | Public User |
| 12. | To pay the compound (if any) via pay online system | Pay Compound Online | Registered User |
| 13. | To check whether the user have to pay any tax. | Check Tax Status | Public User |
| 14. | To pay the tax (if any) via pay online system | Pay Tax Online | Registered User |
| 15. | To apply business license by fill in the applicant details such as name, address, contact number and so on in the online form. The details will be recorded in the local government database | Apply Business License Online | Registered User |
| 16. | To check the license application status e.g.: approved, disapproved or in | Check License Application Status | Public User |

| | | | |
|---|---|---|---|
| | process | | |
| 17. | To pay the approved license via pay online system | Pay License Online | Registered User |
| 18. | To apply to rent public facilities by fill in the applicant details such as name, address, contact number and so on in the online form. The details will be recorded in the local government database | Apply to Rent Public Facilities | Registered User |
| 19. | To check the rent application status e.g.: approved, disapproved or in process | Check Rent Application Status | Public User |
| 20. | To pay for the rent via pay online system | Pay Rent Online | Registered User |
| 21. | Online applications via the web server are routed to the staff in charge. The users data is recorded and note related to the users are created. | Manage Online Application | Local Government Staff, Web Server |
| 22. | Registered user pay the service providers by charging to their Visa or MasterCard | Manage Pay online | Service Providers |

Table 4.3: Requirements list for local government EIP

### 4.4.4 Use Case Specification

The flow of events for a use case is a description of the events needed to accomplish the required behavior of the use case. Activity diagram is not illustrated or discussed, because the flow of events in this project provide a detail description of use cases' activity, and was regarded by the author as more suitable for the audience of this project.

#### 4.4.4.1 E-checking Package

This section will describe the flow of events for each use case in e-checking package

##### 4.4.4.1.1 Flow of events for Check Compound Status

**a) Brief Description**

This use case is activated by public user. It enables the user to check whether they have any compound record or not.

**b) Pre-Conditions**

None

**c) Characteristic of Activation**

On user demand

**d) Flow of Events (Description)**

**1) Basic Flow**

- This use-case is initiated when the public user want to check whether they have any compound record or not
- User click on **Service** Menu and select **compound** service
- User select check compound status
- Next, user key-in the car plat Number and click **SUBMIT** button
- The system retrieve the information from database and display in the same window

**2) Alternative Flows**

Not applicable

**3) Exception Flows**

**E1**: The link cannot be open

Report dead link to the admin using e-mail

**E2**: User does not enter the plat Number

The system will display error message. The user asked to enter the car plat Number

**e) Post-Conditions**

Information about user compound status will be displayed

**f) Rule (s)**

The plat number must contains maximum 7 characters

**g) Constraints (s)**

None

**4.4.4.1.2 Flow of events for Check Tax Status**

**a) Brief Description**

This use case is activated by public user. It enables the user to check whether they have any assessment tax or not

**b) Pre-Conditions**

None

**c) Characteristic of Activation**

On user demand

**d) Flow of Events (Description)**

**1) Basic Flow**

- This use-case is initiated when the public user wants to check whether they have any assessment tax record or not
- User click on **Service** Menu and select tax option
- User click check tax option
- User key-in their IC number

- The system retrieve the information from database and display in the same window

**2) Alternative Flows**

Not applicable

**3) Exception Flows**

**E1**: The link cannot be open

Report dead link to the admin using e-mail

**E2**: User does not enter the IC number

The system will display error message. The user asked to enter the IC number again

**E3**: The system cannot retrieve the data in the database

Error message will be display to inform the user to try again later. The system will inform the user that there was a problem with the database.

**e) Post-Conditions**

Information about user tax status will be displayed

**f) Rule (s)**

Not applicable

**g) Constraints (S)**

None

**4.4.4.1.3 Flow of events for Check License Application status**

**a) Brief Description**

This use case is activated by public user. It enables the user to check whether their license application is approved or not.

**b) Pre-Conditions**

None

**c) Characteristic of Activation**

On user demand

**d) Flow of Events (Description)**

**1) Basic Flow**

- This use-case is initiated when the public user wants to check whether their license application is approved or not
- User click on Service Menu and select license
- Next, user select check license application status
- User key-in application form serial Number
- The system retrieve the information from database and display in the same window

**2) Alternative Flows**

Not applicable

**3) Exception Flows**

**E1**: The link cannot be open

Report dead link to the admin using e-mail

**E2**: User entering wrong serial No

The system will display error message. The user asked to enter the serial number again

**e) Post-Conditions**

Information about user application status will be displayed

**f) Rule (s)**

Not applicable

**g) Constraints (s)**

None

#### 4.4.4.2 E-application Package

This section will describe the flow of events for each use case in e-application package

##### 4.4.4.2.1 Flow of events for Apply Business License Online

**a) Brief Description**

This use case is activated by registered user. It enables the user to apply the business license online.

**b) Pre-Conditions**

Registered user must log-in first before accomplish this function

**c) Characteristic of Activation**

On user demand

**d) Flow of Events (Description)**

**1) Basic Flow**

- This use-case is initiated when the registered user want to apply for business license.
- User click Service Menu and select e-application
- User click on icon Online Business License
- User fill the form and click on SUBMIT button
- The system will receive the data and insert in the database
- The system will display message showing that the user has successfully submit the form.

**2) Alternative Flows**

Not applicable

**3) Exception Flows**

**E1**: The link cannot be open

Report dead link to the admin using e-mail

**E2**: User do not entering the MUST field in the form

The user will receive error message from the system.

User been asked to fill again the form.

**E3**: The system cannot insert the data in the database

Error message will be display to inform the user to try again later. The system will inform the user that there was a problem with the database.

**e) Post-Conditions**

User's application will be processed

**f) Rule(S)**

User must log in first to use this application

**g) Constraint(s)**

User only allowed to apply one business license in one time

**4.4.4.2.2 Flow of events for Apply to Rent Public Facilities**

**a) Brief Description**

This use case is activated registered user. It enables the user to apply for renting a building or other public facilities online.

**b) Pre-Conditions**

Registered user must log-in first before accomplish this function

**c) Characteristic of Activation**

On user demand

**d) Flow of Events (Description)**

**1) Basic Flow**

- This use-case is initiated when the public user want to rent building or other public facilities via online application
- User will select the Service Menu and choose online rent
- User fill the rent form and click on SUBMIT button
- The system will receive the data and insert in the database
- The system will display message showing that the user has successfully submit the form.

**2) Alternative Flows**

Not applicable

**3) Exception Flows**

**E1**: The link cannot be open

Report dead link to the admin using e-mail

**E2**: User do not entering the MUST field in the form

The user will receive error message from the system.

User been asked to fill again the form.

**E3**: The system cannot insert the data in the database

Error message will be display to inform the user to try again later. The system will inform the user that there was a problem with the database

**e) Post-Conditions**

User's application will be processed

**f) Rule(s)**

User must log in first to use this application

**g) Constraint(s)**

User only allowed applying one facility in one time

#### 4.4.4.3 E-payment Package

This section will describe the flow of events for each use case in e-payment package

##### 4.4.4.3.1 Flow of events for Pay Compound Online

**a) Brief Description**

This use case is activated by registered user. It enables the user to pay compound online.

**b) Pre-Conditions**

Registered user must log-in first before accomplish this function

**c) Characteristic of Activation**

On user demand

**d) Flow of Events (Description)**

**1) Basic Flow**

- This use-case is initiated when the registered user wants to pay compound online.
- User will click on Service Menu and choose Online Payment
- User choose compound payment
- User will fill the form and choose method of payment
- User submit the form
- The system will receive the data and insert in the database
- The system will display message showing that the user has successfully submit the form

**2) Alternative Flows**

Not applicable

**3) Exception Flows**

**E1**: The link cannot be open

Report dead link to the admin using e-mail

**E2**: User do not entering the MUST field in the form

The user will receive error message from the system.

User been asked to fill again the form.

**E3**: The system cannot insert the data in the database

Error message will be display to inform the user to try again later. The system will inform the user that there was a problem with the database

**e) Post-Conditions**

- System will process the payment.
- User account will be credited

**f) Rule (S)**

User must log in first to use this application

**g) Constraints (s)**

Only Master Card or Visa Card are accepted

**4.4.4.3.2 Flow of events for Pay Tax Online**

**a) Brief Description**

This use case is activated by registered user. It enables the user to pay tax online.

**b) Pre-Conditions**

Registered user must log-in first before accomplish this function

**c) Characteristic of Activation**

On user demand

**d) Flow of Events (Description)**

**1) Basic Flow**

- This use-case is initiated when the registered user wants to pay for tax or assessment online.
- User will click on Service Menu and choose Online Payment
- User choose types of service payment
- User will fill the form and choose method of payment
- User submit the form
- The system will receive the data and insert in the database
- The system will display message showing that the user has successfully submit the form

**2) Alternative Flows**

Not applicable

**3) Exception Flows**

**E1**: The link cannot be open

Report dead link to the admin using e-mail

**E2**: User do not entering the MUST field in the form

The user will receive error message from the system. User been asked to fill again the form.

**E3**: The system cannot insert the data in the database

Error message will be display to inform the user to try again later. The system will inform the user that there is a problem with the database

**e) Post-Conditions**

- System will process the payment.
- User account will be credited

**f) Rule (s)**

User must log in first to use this application

**g) Constraints (s)**

Only Master Card or Visa Card are accepted

#### 4.4.4.3.3 Flow of events for Pay License Online

**a) Brief Description**

This use case is activated by registered user. It enables the user to pay license online.

**b) Pre-Conditions**

Registered user must log-in first before accomplish this function

**c) Characteristic of Activation**

On user demand

**d) Flow of Events (Description)**

**1) Basic Flow**

- This use-case is initiated when the registered user want to pay for license online.
- User will click on Service Menu and choose Online Payment
- User choose license payment
- User will fill the form and choose method of payment
- User submit the form
- The system will receive the data and insert in the database
- The system will display message showing that the user has successfully submit the form

**2) Alternative Flows**

Not applicable

**3) Exception Flows**

**E1**: The link cannot be open

Report dead link to the admin using e-mail

**E2**: User do not entering the MUST field in the form

The user will receive error message from the system. User been asked to fill again the form.

**E3**: The system cannot insert the data in the database

Error message will be display to inform the user to try again later. The system will inform the user that there is a problem with the database

**e) Post-Conditions**

- System will process the payment.
- User account will be credited

**f) Rule (S)**

- User must log in first to use this application

**g) Constraints (s)**

Only Master Card or Visa Card are accepted

**4.4.4.3.4 Flow of events for Pay Rent Online**

**a) Brief Description**

This use case is activated by registered user. It enables the user to pay rent online.

**b) Pre-Conditions**

Registered user must log-in first before accomplish this function

**c) Characteristic of Activation**

On user demand

**d) Flow of Events (Description)**

**1) Basic Flow**

- This use-case is initiated when the registered user want to pay for rent online.
- User click on Service Menu and choose Online Payment
- User choose rent payment
- User will fill the form and choose method of payment
- User submit the form
- The system will receive the data and insert in the database
- The system will display message showing that the user has successfully submit the form

**2) Alternative Flows**

Not applicable

**3) Exception Flows**

**E1**: The link cannot be open

Report dead link to the admin using e-mail

**E2**: User do not entering the MUST field in the form

The user will receive error message from the system. User been asked to fill again the form.

**E3**: The system cannot insert the data in the database

Error message will be display to inform the user to try again later. The system will inform the user that there is a problem with the database

**e) Post-Conditions**

- System will process the payment.
- User account will be credited

**f) Rule (s)**

User must log in first to use this application

**g) Constraints (s)**

Only Master Card or Visa Card are accepted

#### 4.4.4.4 E-complaint Package

This section will describe the flow of events for each use case in e-complaint package

##### 4.4.4.4.1 Flow of events for Make Online Complaint

**a) Brief Description**

This use case is activated by registered user. It enables the community to make complaint online.

**b) Pre-Conditions**

Registered user must log-in first before accomplish this function

**c) Characteristic of Activation**

On user demand

**d) Flow of Events (Description)**

**1) Basic Flow**

- This use-case is initiated when the registered user wants to make complaint to local government
- User will click Service Menu on and choose icon Online Complaint Form
- User fill the form and click on SUBMIT button
- The system will receive the data and insert in the database
- The system will display message showing that the user has successfully submit the form.

**2) Alternative Flows**

Not applicable

**3) Exception Flows**

**E1**: The link cannot be open

Report dead link to the admin using e-mail

**E2**: User do not entering the MUST field in the form

The user will receive error message from the system. User been asked to fill again the form.

**E3**: The system cannot insert the data in the database

Error message will be display to inform the user to try again later. The system will inform the user that there is a problem with the database

**e) Post-Conditions**

- User's complaint will be processed.
- The complaint will be replied within 3 days

**f) Rule(s)**

- User must log in first to use this application
- MPKS will only process complaint within the Kota Star area.

**g) Constraint(s)**

Complaint must less than 50 words

#### 4.4.4.5 Non Package use cases

This section will describe the flow of events for non package use case

##### 4.4.4.5.1 Flow of events for Access General Information

**a) Brief Description**

This use case is activated by all public user and registered user of MPKS. It enables the user to access information related to MPKS.

**b) Pre-Conditions**

User must open the MPKS Portal

**c) Characteristic of Activation**

On user demand

**d) Flow of Events (description)**

**1) Basic Flow**

- This use-case is initiated when the public user or registered user want to access information about MPKS
- User click on Profile icon to preview MPKS information
- The browser will open the link on the same window

**2) Alternative Flows**

Not applicable

**3) Exception Flows**

**E1**: The link cannot be open

Report dead link to the administrator using e-mail

**4) Post-conditions**

User gain information on the same window

**5) Rule (s)**

Not applicable

**6) Constraints (s)**

Public user are only allowed to access general information

#### 4.4.4.5.2 Flow of Events for Access Specification

**a) Brief Description**

This use case is activated by all public user and registered user of the portal. It enables the user to access the local council service specification such as building specification, tender specification and etc.

**b) Pre-Conditions**

User must open the MPKS Portal

**c) Characteristic of Activation**

On user demand

**d) Flow of Events (Description)**

**1) Basic Flow**

- This use-case is initiated when the public user or registered user want to access local council service specifications
- User click on Service icon to preview the service provided by the council
- Next, user click on type of service for example rent
- The browser will open the link on the same window

**2) Alternative Flows**

Not applicable

**3) Exception Flows**

**E1**: The link cannot be open

Report dead link to the administrator using e-mail

**4) Post-conditions**

User gain information on the same window

**5) Rule (s)**

Public user are only allowed to access general information

**6) Constraints (s)**

None

#### 4.4.4.5.3 Flow of events for Do Searching

**a) Brief Description**

This use case is activated by public or registered user. It enables the user to do searching based on keywords given.

**b) Pre-Conditions**

Must Open MPKS portal first

**c) Characteristic of Activation**

On user demand (based on keywords)

**d) Flow of Events (Description)**

**1) Basic Flow**

- This use-case is initiated when the user want to do searching
- User type keyword (or keywords) to the search box
- User Click GO button
- The search system will search the meta content that were matching with the keyword input by the user
- The system will display the result in the same window
- User will choose the result
- The browser will open the link

**2) Alternative Flow**

Not applicable

**3) Exception Flow**

**E1**: The link cannot be open

Report dead link to the admin using e-mail

**E2**: The keyword do not matching with the Meta content

Display error message. Ask the user to input again

**E3**: User input too many keywords

Display error message. Ask the user to input again.

**e) Post-Conditions**

- Search result will be displayed in the same window
- Browser will open the link chosen by the user

**f) Rule (S)**

User must type appropriate keyword to get the best result

**g) Constraints (s)**

The keywords must less than 8 words

#### 4.4.4.5.4 Flow of events for Download Forms

**a) Brief Description**

This use case is activated by all public user and registered user of the portal. It enables the user to download the council forms such as license application form and rent application form

**b) Pre-Conditions**

None

**c) Characteristic of Activation**

On user demand

**d) Flow of Events (description)**

**1) Basic Flow**

- This use-case is initiated when the public user or registered user want to download the local council's forms.
- User click on Service icon to preview the service provided by the council
- Next, user click on download form icon and choose the form they need

**2) Alternative Flows**

Not applicable

**3) Exception Flows**

**E1: The link cannot be open**

Report dead link to the administrator using e-mail

**4) Post-conditions**

The message box will be displayed to ask whether to save or open the form

**5) Rule (s)**

User must have the application software such Microsoft Word to download the form

**6) Constraints (s)**

Only rich text format is provided at the moment. The PDF format is not applied yet.

#### 4.4.4.5.5 Flow of events for Login Portal

**a) Brief Description**

This use case is activated by registered user. It enables the user to enter the portal and use several functions provided for registered user.

**b) Pre-Conditions**

User has to register the portal first

**c) Characteristic of Activation**

On user demand

**d) Flow of Events (Description)**

**1) Basic Flow**

- This use-case is initiated when the registered user want to enter a portal as an authorized person and use the functions provided for registered user
- User type their username and password and click the **SUBMIT** button.
- The system compare the id and password the user input with data in the database
- The system verify and user gain access into the system

**2) Alternative Flows**

Not applicable

**3) Exception Flows**

**E1**: The link cannot be open

Report dead link to the admin using e-mail

**E2**: User entering wrong id or password

The system will display error message. The user asked to enter id or password again

**e) Post-Conditions**

The registered user web page will be displayed

**f) Rule (s)**

Not applicable

**g) Constraints (s)**

- User name and password are case-sensitive.
- User name and password must less than 8 letters each.

#### 4.4.4.5.6 Flow of events for Register Portal

**a) Brief Description**

This use case is activated by public user. It enables the user to register the portal and become registered user

**b) Pre-Conditions**

None

**c) Characteristic of Activation**

On user demand

**d) Flow of Events (Description)**

**1) Basic Flow**

- This use-case is initiated when the public user want to register the portal
- User click the **REGISTER PORTAL** button
- User fill in the registration form and click **SUBMIT** button
- The user data will be recorded in database system

**2) Alternative Flows**

Not applicable

**3) Exception Flows**

**E1**: The link cannot be open

Report dead link to the admin using e-mail

**E2**: User do not entering the MUST field in the form

The user will receive error message from the system. User been asked to fill again the form.

**E3**: The system cannot insert the data in the database

Error message will be display to inform the user to try again later. The system will inform the user that there is a problem with the database

**e) Post-Conditions**

The confirmation message with user name and password will be displayed to inform user that they have successfully registered the portal.

**f) Rule (s)**

- User must fill in all the **MUST** field
- One user are allowed to register once only

**g) Constraints (s)**

- User name and password are case-sensitive.
- User name and password must less than 8 letters each.

**. 4.4.4.1 Flow of events for Update Details**

**a) Brief Description**

This use case is activated by registered user. It enables the user to update their record

**b) Pre-Conditions**

User must login first

**c) Characteristic of Activation**

On user demand

**d) Flow of Events (Description)**

**1) Basic Flow**

- This use-case is initiated when the registered user want to update their details
- User click the **UPDATE DETAILS** button
- User fill in new information in the registration form and click **SUBMIT** button
- The user data will be updated in database system

**2) Alternative Flows**

Not applicable

**3) Exception Flows**

**E1**: The link cannot be open

Report dead link to the admin using e-mail

**E2**: The system cannot insert the data in the database

Error message will be display to inform the user to try again later. The system will inform the user that there was a problem with the database

**e) Post-Conditions**

The confirmation message will be displayed to inform user that they have successfully updated their details.

**f) Rule (s)**

None

**g) Constraints (s)**

- User name and password are case-sensitive.
- User name and password must less than 8 letters each.

#### 4.4.4.5.8 Flow of events for View History/Report

**a) Brief Description**

This use case is activated by the state government staff. It enables the staff to access the history of business activities that have been done and also to know whether the local government have make profit or not.

**b) Pre-Conditions**

The user has to log in as a state government staff

**c) Characteristic of Activation**

On user demand

**d) Flow of Events (Description)**

**1) Basic Flow**

- This use-case is initiated when the state government staff wants to view a complete history or business activities of the local council.
- The staffs have to click the view history icon on the main panel of state government site.
- The staffs have to select the activities and time period
- The system will check the activities record in the database

**2) Alternative Flows**

Not applicable

**3) Exception Flows**

**E1**: The link cannot be open

Report dead link to the admin using e-mail

**E2**: User entering wrong id or password

The system will display error message. The user is asked to enter id or password again

**e) Post-Conditions**

Reports will be displayed

**f) Rule (s)**

Only state governments' staffs (higher level) are allowed to use this function

**g) Constraints (s)**

- Only monthly and annually business report are provided
- Report for the present month are only can be view on the month after

#### 4.4.4.6 Online Management Package

This section will describe the flow of events for each use case in online management package

##### 4.4.4.6.1 Flow of events for Manage Online Application

**a) Brief Description**

This use case is involved by the local government staff and the web server. The online application via the web server are routed to the person in charge

**b) Pre-Conditions**

Users 'online application such as apply for license

**c) Characteristic of Activation**

Compulsory if the user perform the online application

**d) Flow of Events (Description)**

**1) Basic Flow**

- This use-case is initiated when the council receive e-mail or online application forms from the user
- The system analyses the mail or application type
- System record data in database
- System route e-mail or the application according to person in charge

**2) Alternative Flows**

Not applicable

**3) Exception Flows**

**E1: System receives irrelevant e-mail or empty mail.**

Delete the mail.

**E2: Unreliable information or data**

Ignore the application or request

**e) Post-Conditions**

The person in charge will receive the mail and give response or feedback to the user

**f) Rule (s)**

Not applicable

**g) Constraints (s)**

Person in charge are not standardized, sometimes e-mail are routed to the wrong person

### 4.4.5 System Architecture

Figure 4.4 depicts the recommended architecture for the local government portal system.

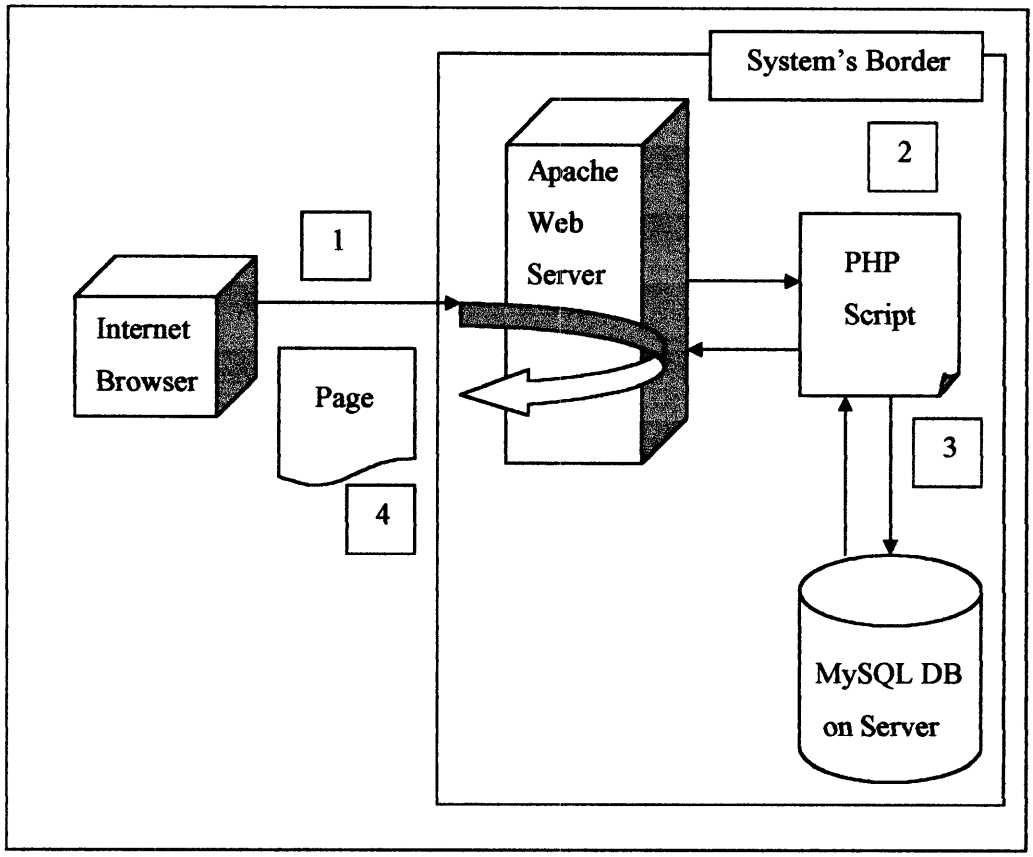



Figure 4.4: Overview of System Architecture

The architecture is explained in public user perspective or external use. When a user accesses to the system on-line (via the web interface). It is referred as external use. Following are the description for the system architecture.

1) User visits the local government portal with a web browser and tries to access information from portal web pages.

2) The browser connects to the Apache Web Server. PHP Scripts that have the information on how to connect to MySQL Database Server

and how to retrieve the required information from the database will run.

3) MySQL database server will send the required information based on the PHP Scripts.

4) Apache Web Server will dynamically create HTML pages for user viewing.

## 4.5 Conclusion

This chapter addressed the findings of the study. The requirements model for local government Enterprise Information Portal (EIP) is described in detail and modeled using use case packages, use cases, and use case specification. Additionally the system architecture is also discussed.

CHAPTER 5

# PROTOTYPE APPLICATION INTERFACE DESIGN

## 5.1 Introduction

In this chapter, user interfaces of the prototype application will be illustrated and explained. Only the main user interfaces will be explained and illustrated in this chapter.

## 5.2 Prototype Development

In this project PHP MySQL were used to create the database-enabled Web site. PHP is an open source server-side scripting language (designed specifically for the web). PHP was chosen due to its strength (Welling and Thomson, 2001), which includes, high performance, interfaces to many different database system, built-in libararies available for many common Web task, low cost, ease of learning and use, and portability. Besides that Macromedia Dreamweaver MX was also use to create the interface design for this project.

MYSQL was used as the database because it is low cost, very fast, robust, provides high performance, multi-threaded, multi-user, provides high security, and easy to use. Apache Web Server was used due to its high performance, versatility, low cost (open source), and its compatibility with PHP and MYSQL.

## 5.3 Interface Design

The following are interface design and transaction description for some of the prototype system's main features.

### 5.3.1 Portal Main Page


Figure 5.1: Portal Main Page

Figure 5.1 displays the main page of MPKS portal. There are seven main menus in the interface. The menus are:

- Home
- MPKS Profile
- Services
- Tourism
- Sport and Recreation
- General Info
- Other links

This page is considered as a main page which consist login for registered users and the MPKS staff to enter the system. Here public user can also choose to register the portal.

The Home page contains the latest information around Alor Star and MPKS Bulletin.

### 5.3.2 MPKS Profile Interface

Figure 5.2 depicts the MPKS Profile user interface.

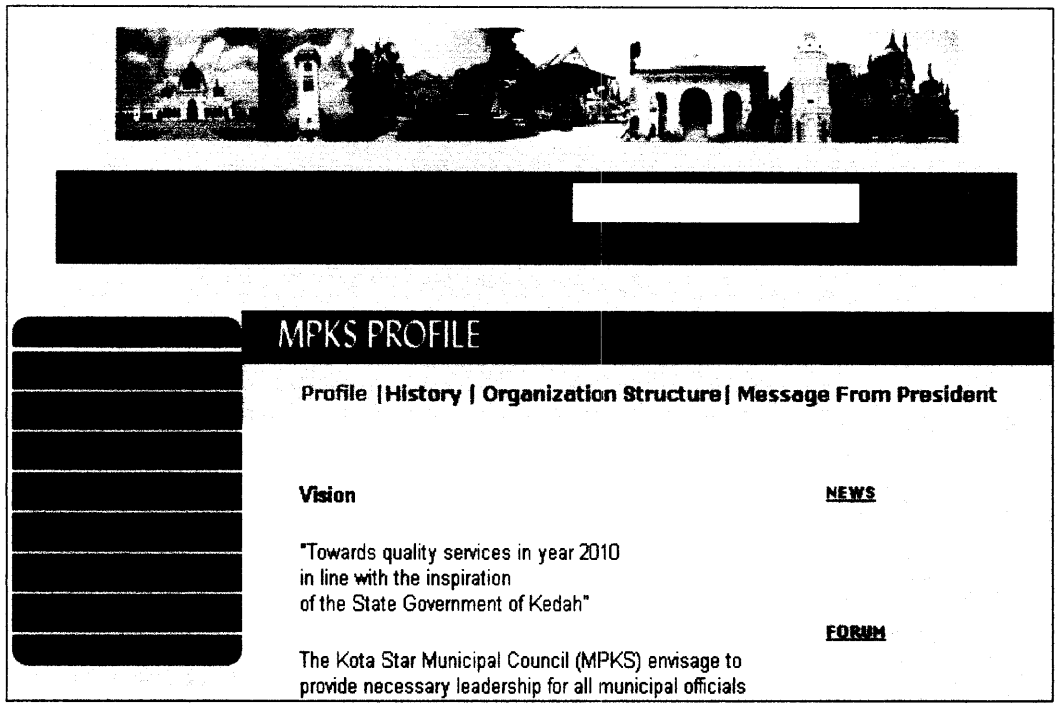



Figure 5.2: MPKS Profile Interface

MPKS Profile provides detail information about MPKS. There are 4 sub functions. The functions are :

1. Profile- history, vision, mission, and its function to the community
2. Organization Chart- Shows the structure of the MPKS, showing the ranking from the top management to the lowest level of staff.
3. Message from President- the president honors to the MPKS
4. Government act or policies and act used in MPKS

### 5.3.3 Service Interface

Figure 5.3 depicts the public user service interface.

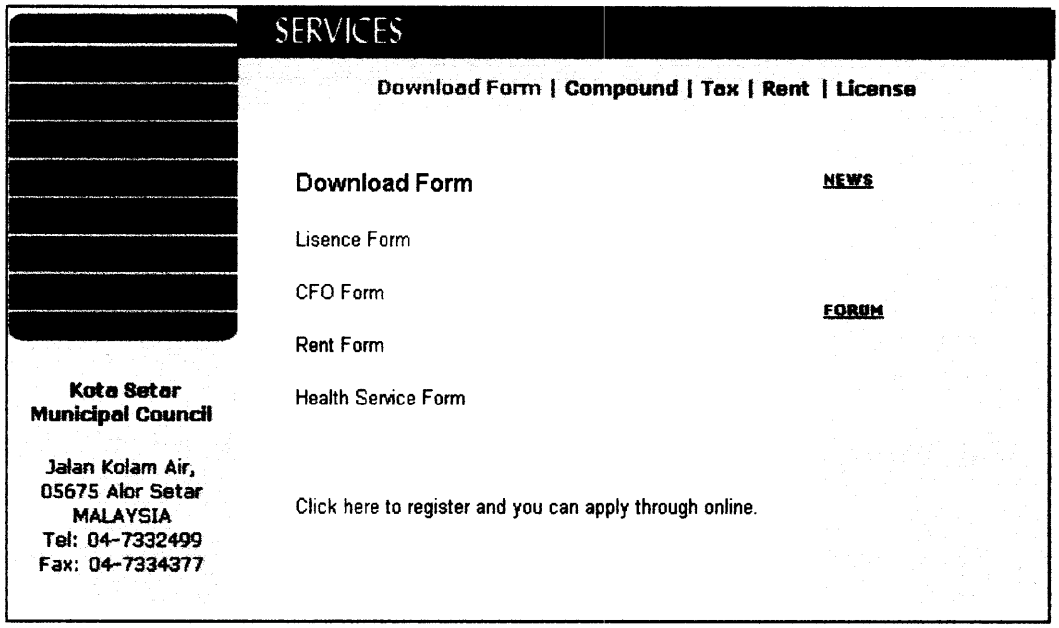



**Figure 5.3: Public User Service Interface**

In this menu, public user can access all the services that being offered by MPKS and download the form from the portal. The services provided for public user are access general information about compound, tax, public facilities and business license. Besides that users can also check the compound status, tax status and license application status.

For the registered user several functions are added in the interface. Besides perform all the functions of public users they can also make complaint through online application and apply the service through online forms provided in the portal. Figure 5.4 below shows the registered user service interface.

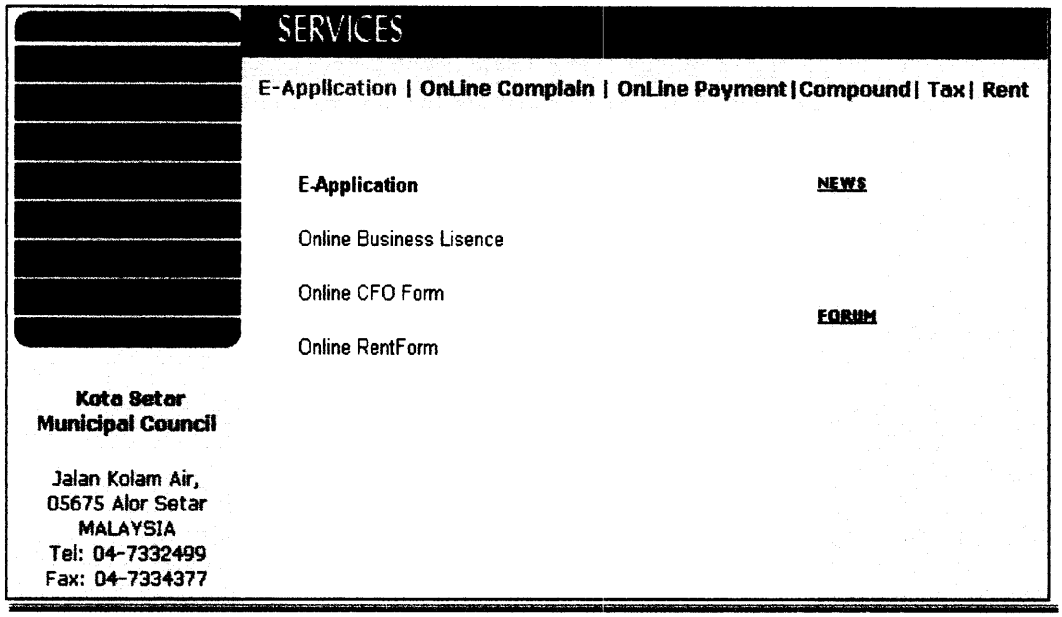



Figure 5.4: Registered User Service Interface

### 5.3.4 Check Compound Status Interface

Figure 5.5 below is the interface for checking the traffic compound status. Users have to key in the car plat number to check their compound status.

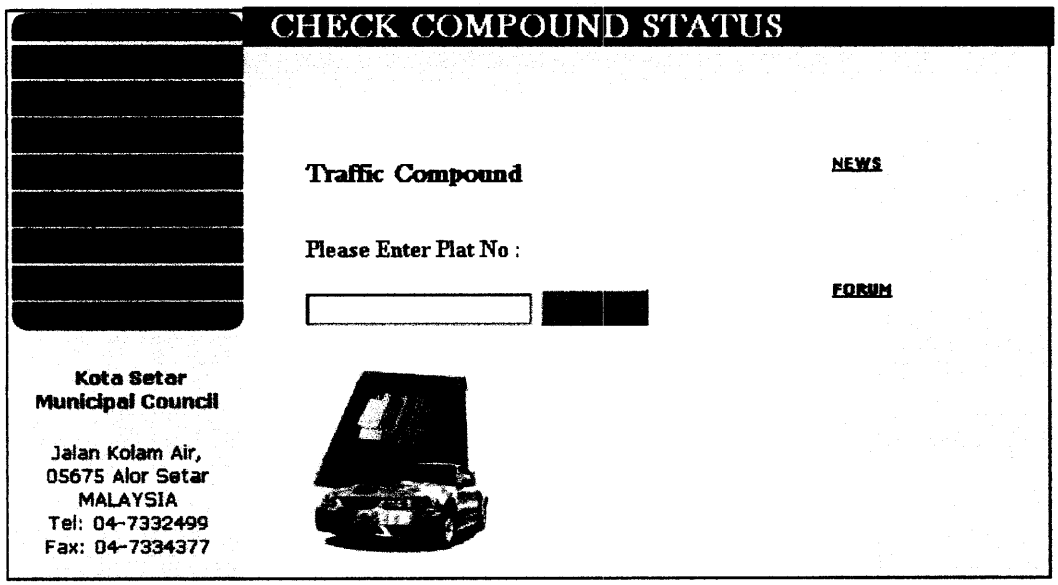



Figure 5.5: Check Compound Status Interface

If the users have compound record, the compound details will be displayed as figure 5.6 below, otherwise the message will be displayed to inform the user that they have no compound record.

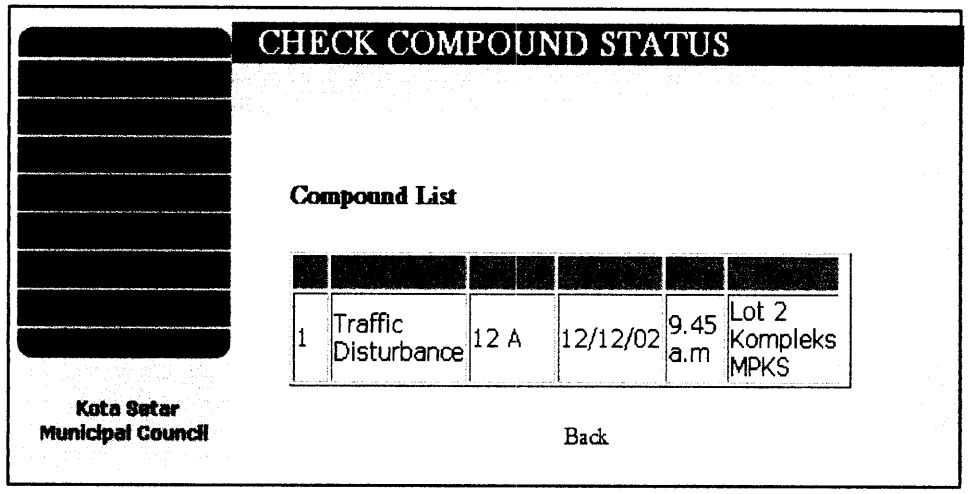



Figure 5.6: List of compound record found or compound details

### 5.3.5 Make Complaint Interface

Figure 5.7 shows the interface for make complaint function. To activate the function user has to fill in the form and click the SUBMIT button.

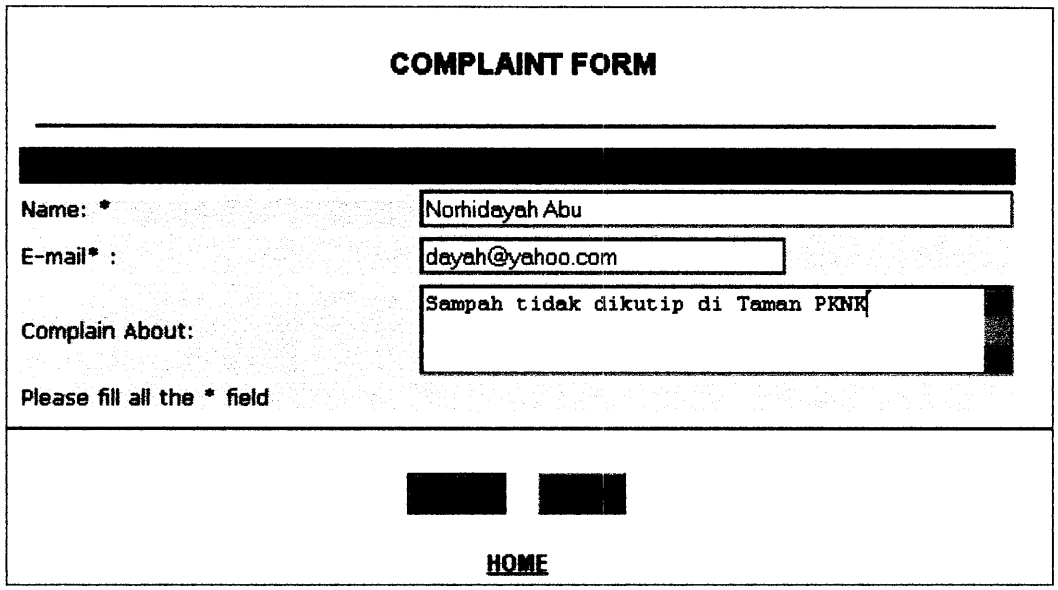



Figure 5.7: The interface to make online complaint

Only registered users are allowed to make online complaint. If the complaint is received by the MPKS the message in figure 5.8 will be displayed otherwise user have to fill in the form again.

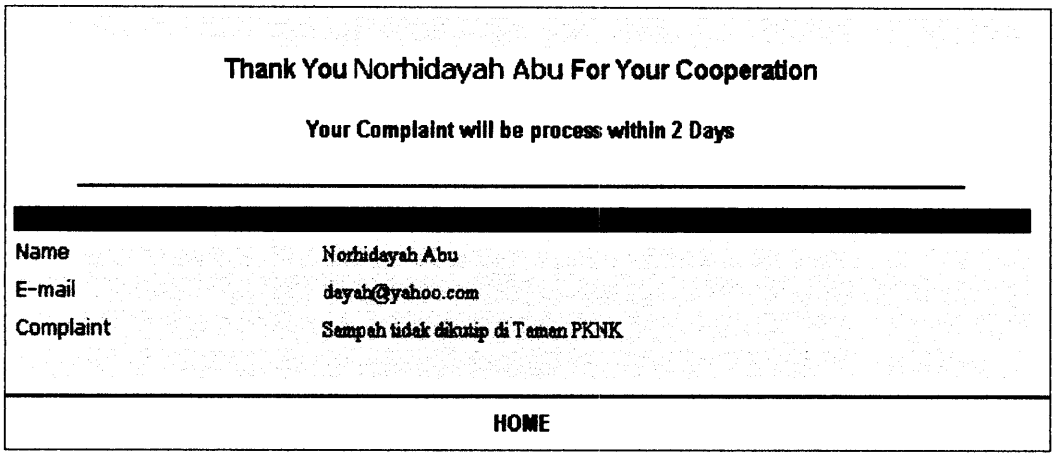



Figure 5.8: Message box for successful complaint

### 5.3.5 Tourism interface

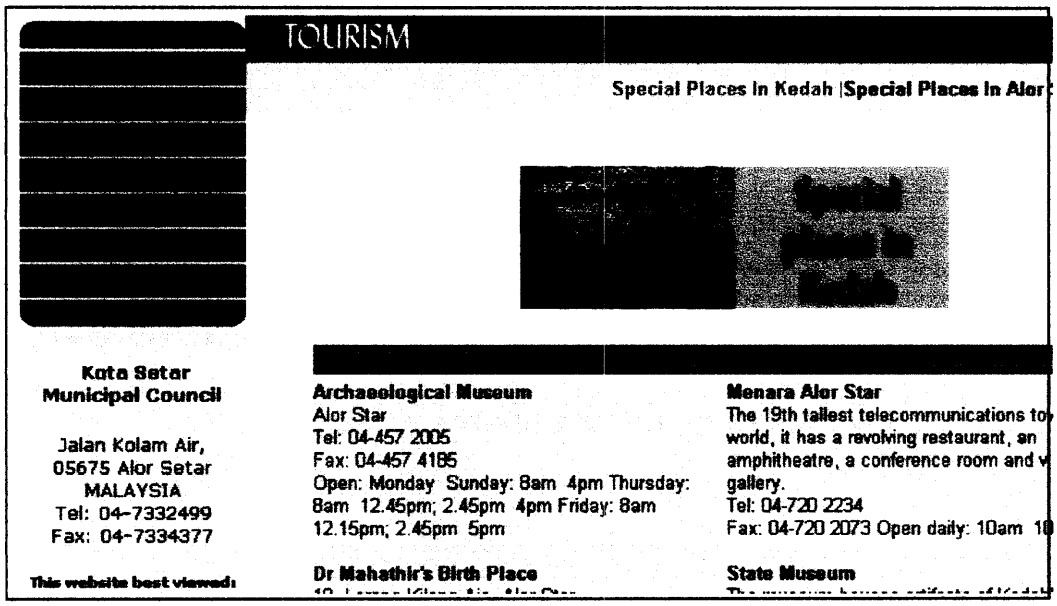



Figure 5.9: Tourism Interface

All types of user can access this function. It contains:

- Maps – Malaysia, Kedah, Alor Star
- Hotel- In Kedah and Alor Star
- Interesting Places- In Kedah and Alor Star
- Historical Places- In Kedah and Alor Star. For example the place are Dr Mahathir's birth place, Archeological museum, state museum, Lembah Bujang and Pulau Langkawi

### 5.3.7 Additional features

The portal also contains several additional features. The features are:

- Search engine
- Waktu solat
- Calendar
- E-mail
- Poll

## 5.4 Conclusion

Application interface and transaction design provide users and developers a way to view and validate the users' requirement. Only the main application interfaces and transaction design for the MPKS portal was illustrated in this chapter via the prototype system that was developed using various programs.

CHAPTER 6

# SYSTEM'S VALIDATION

## 6.1 Introduction

In this chapter, the MPKS portal will be crossed check against users' requirements. The goal of this chapter is to ensure users' requirements are met and to gauge the level of functionality and operability of the system.

## 6.2 Validation

In this project, the system was tested and validated in the following ways:

### 6.2.1 System Requirements Testing

In this method, each user's requirements are tested to ensure that the system appears to be working according to users' specification and requirements. Additionally, outcome of testing also provides a measure of software reliability and quality. The portal system was tested using Microsoft Internet Explorer 6.0, Apache server but via offline application. All tests were carried out by stand-alone personal computer, which operates under Windows 98 environment with a Pentium four processor.

The tests conducted and its results can be viewed in table 6.1 below.

| Test | Description | Actual System Performance |
|---|---|---|
| Requirement 1 | To record user name, password, full name, address, and contact details for each registered portal user | MPKS EIP meets this requirement as the users' details can be recorded in the database |
| Requirement 2 | To authenticate the portal user in order to prevent unauthorized access to information and functions. | MPKS EIP meets this requirement as the system can check the username and password correctly before the user enter the system |
| Requirement 3 | To browse the portal content (View general information related to local government) | MPKS EIP meets this requirement as the user can browse for information |
| Requirement 4 | To view the service specification | MPKS EIP meets this requirement |
| Requirement 5 | Update user's record such as change the address or change user name and password. | MPKS EIP meets this requirement as user's record can be updated in database |
| Requirement 6 | To search the portal meta-content | MPKS EIP does not support this function yet |
| Requirement 7 | To download local governments application forms (eg:license form, rent form) | MPKS EIP meets this requirement as users can download the forms |
| Requirement 8 | To send complaint to local government. | MPKS EIP meets this requirement as user's complaint can be recorded |
| Requirement 9 | To check whether the user have to pay any compound | MPKS EIP meets this requirement as the user's |

| | | |
|---|---|---|
| | | compound record can be displayed |
| Requirement 10 | To pay the compound (if any) via pay online system | MPKS EIP does not support this function yet |
| Requirement 11 | To check whether the user have to pay any tax. | MPKS EIP does not support this function yet |
| Requirement 12 | To pay the tax (if any) via pay online system | MPKS EIP does not support this function yet |
| Requirement 13 | To apply business license by fill in the applicant details such as name, address, contact number and so on in the online form. The details will be recorded in the local government database | MPKS EIP does not support this function yet |
| Requirement 14 | To check the license application status e.g.: approved, disapproved or in process (for application via manual system) | MPKS EIP meets this requirement as the user's license application status can be displayed |
| Requirement 15 | To pay the approved license via pay online system | MPKS EIP does not support this function yet |
| Requirement 16 | To apply to rent public facilities by fill in the applicant details such as name, address, contact number and so on in the online form. The details will be recorded in the local government database | MPKS EIP does not support this function yet |
| Requirement 17 | To check the rent application status e.g.: approved, disapproved or in process | MPKS EIP meets this requirement as the user's application status can be displayed |
| Requirement | To pay for the rent via pay | MPKS EIP does not |

| | | |
|---|---|---|
| 18 | online system | support this function yet |
| Requirement 19 | Online applications via the web server are routed to the staff in charge. The users data is recorded and note related to the users are created. | MPKS EIP does not support this function yet |
| Requirement 20 | Registered user pay the service providers by charging to their Visa or MasterCard | MPKS EIP does not support this function yet |

Table 6.1: System Requirement Testing Table

Table 6.1 records the outcome of user requirements testing done on MPKS EIP. From the outcome in table 6.1, only 50% of the requirements were met. This is because the feature was not implemented in MPKS EIP due to time constraint to complete this project. Besides that the prototype developed in this project is only a simple prototype to test out the design ideas. As the objective of this project is only to define a requirement model, so the author is focused more on the model rather than prototype development.

The system's performance was generally good, as no performance problems were encountered during testing. Consequently, the test proves that users' requirements have been met. This is because although only 50% of the functional requirements were successful but in terms of design ideas the prototype has successfully implements the entire requirements been captured in the study.

### 6.2.2 Heuristic Evaluation on MPKS Portal

This method on the other hand, evaluates the usability of MPKS EIP from user's perspective. The same evaluators were choosed to perform the evaluation on MPKS EIP portal. This is to ensure the consistency of result between all portals. The methodology to perform the evaluation is discussed

in chapter 3. The results of the Heuristic evaluation on MPKS EIP are as follows:

| Heuristics | Results of First Evaluator | Results of Second Evaluator | Results of Third Evaluator | Mean of Result | Severity Rating |
|---|---|---|---|---|---|
| Heuristics 1 | 75.0 % | 70.37 % | 75.0 % | 73.46 % | 2 |
| Heuristics 2 | 76.47 % | 82.35 % | 73.91 % | 77.58 % | 1 |
| Heuristics 3 | 65.22 % | 47.62 % | 50.0 % | 54.28 % | 2 |
| Heuristics 4 | 78.57 % | 90.70 % | 80.95 % | 83.41 % | 1 |
| Heuristics 5 | 80.0 % | 65.0 % | 55.56 % | 66.85 % | 2 |
| Heuristics 6 | 50.0 % | 60.0 % | 41.67 % | 50.56 % | 2 |
| Heuristics 7 | 59.46 % | 80.0 % | 66.67 % | 68.71 % | 2 |
| Heuristics 8 | 35.71 % | 40.0 % | 64.29 % | 46.67 % | 3 |
| Heuristics 9 | 87.5 % | 85.71 % | 88.89 % | 87.37 % | 1 |
| Heuristics 10 | 63.16 % | 66.67 % | 60.0 % | 63.28 % | 2 |

Table 6.2 Result of Heuristic Evaluation on MPKS portal

The MPKS data in Table 6.2 shown that Heuristic No. 8 has severity rating of scale 3, Heuristic No. 1, No. 3, No. 5, No.6, No.7, and No. 10 have severity rating of scale 2 while Heuristic No. 2, No.4, and No. 9 have severity rating of scale 1. Those problems rated with scale 1 and 2 are considered as minor usability problem and not important to be solved. For Heuristic No. 8, with severity rating of scale 3 it is considered as major usability problem that needed to be solved in this project. The major usability violations faced in MPKS portal are:

- The design of the portal site is not flexible for efficient use.
- The portal also doesn't provide alternative means of access and operation for users

The experts had also identified 5 usability problems of MPKS portal as follows:

**Problem 1: No shortcut icon provided in each site.**

The users have to travel from page to page to use certain function, such as to check compound. This is takes time especially for user who operate computer with a slow processor

**Problem 2: No keyboard shortcut provided**

There is no keyboard shortcut for each menu options. User will find difficulties if menu list is many or when the mouse is not function

**Problem 3: No link for help**

Users will find difficulty in getting help as the help function is not available.

**Problem 4: The width of search box is too small for user to view the keyed-in text**

The width of the search box is too small for user who has long key words.

**Problem 5: No navigational feedback**

The user cannot determine their current position in the document structure and user also has difficulty to return to an initial state.

From the above results, as the MPKS EIP only has one major problem with severity rating of scale 3, it is conclude that the MPKS EIP is usable, and meet users' requirements. Additionally, it can also be concluded that the MPKS EIP interface should be improved, as the author believes that, ease of using a system depends greatly on a particular system's interface and interaction with its users.

## 6.3 Conclusion

A system is validated to ensure users' requirements are met and to gauge the level of functionality and operability of a particular system. In this study, the system requirement testing and heuristic evaluation method were used to validate the MPKS Enterprise Information Portal (EIP). It was concluded that, user's requirement were met and the system is operating well. However, the system's functionality and interface could be improved to make it more intuitive and easy to use.

CHAPTER 7

# CONCLUSION

## 7.1 Introduction

This chapter concludes the development and discussion on local government Enterprise Information Portal (EIP) Requirement Model. It elaborates on problems and limitations that the author encountered in completing the project.

## 7.2 Problems and Limitations

This project's main objectives are to captures users' requirement and define a comprehensive requirement model for local government Enterprise Information Portal (EIP) aim specifically at the public users' requirement. However, the development of this paper was not without its problems and limitations. Some of the problem and limitations in developing this requirement model includes:

1) A requirement model specifically for a local government Enterprise Information Portal (EIP), or requirement model for any system was not available or found by the author. Thus, the learning curve to create the local government Enterprise Information Portal requirement model was steep, and time consuming.

2) Implementation of a more comprehensive requirement model that includes the staff module, politician module and administration module or functions was precluded due to time constraints.

3) The prototype system was not fully functional due to time constraints in developing the prototype.

4) The prototype system was not equipped with a complete error-handling function, due to the perceived unimportance of this function for the prototype system, and the additional time required to properly incorporating this function.

5) The MPKS Enterprise Information Portal has yet to be uploaded on the internet for final testing due to time constrain.

### 7.3 Recommendations for Future Project

The following recommendations are provided in order to preclude some of the problems and limitations mentioned above in future project:

1) Future projects should try to incorporate the staff, politician, and administrator requirements in the requirement model to ensure completeness.

2) If time permits, the prototype system should be equipped with all the functions required by users, and features that would make the system more interactive and intuitive.

3) Heuristic evaluation should be conducted to more than 2 local governments' portal to make sure that the requirements captured are more reliable and standardized.

### 7.3 Conclusion

As with any project papers, all the objectives are achieved. The requirement model for local government Enterprise Information Portal (EIP) is defined in this projects, suffers from some problems and limitations that limited its scope and capability. The main limitation identified was time constraint to complete this paper. Nevertheless, several recommendations were put forward for the benefits

of future project. It is hope that in the future a complete requirement model for local government Enterprise Information Portal (EIP) can be defined by adding the requirement from local government staff, state government, politician, and requirement from administrator.

## REFERENCES


Arasan J. (2003). Reference Model for Office of Defense Cooperation (ODC) Training Support System (OTSS). *Thesis UUM.*

Bennett, S., McRobb, S. and Farmer, R. (2002). *Object Oriented System Analysis and Design Using UML (2nd Ed.).* Berkshire: McGraw-Hill Education.

Blaine, T. *Enterprise Information Portal Overview.* Available Online at: http://knowledgemanagement.ittoolbox.com/documents/document.asp? [10th October 2003].

Booch, G., Rumbaugh, J., and Jacobson, I. (1999). *The Unified Modeling Language User Guide.* Boston: Addison-Wesley

Brucher, H. (2002). *A Reference Model to develop Strategic E-Government Concepts.* Available online at: http://www.ercim.org/publication/Ercim_News/enw48/ bruecher.html [15th November 2003]

Cadmus, D. (2000). *Enterprise Information portal Revolution.* Available online at: www.tdan.com/i017hy02.htm. [22nd December 2003]

CHAOS. (1994, 1997). The Standish Group International, Inc., Dennis, MA

Computer Industry Daily. (1997, December 12)

Cover, R. (2002). *ASC X12 Reference Model for XML Design.* Available online at: http://xml.coverpages.org/ni2002-07-29-a.html [15thNovember 2003]

Donegan, M. (2000) *Contemplating portal strategies*; Telecommunications (Intenational Edition); February, 2000, Available at: http://www.telecomsmag.com/issues/200002/tci/portal.html

eMarketer, Inc. (2000): *Top 10 Online Portals*. Available Online at: http://www.emarketer.com/elist/t10portals/top10portals.html

European Prestandard (1996). *TRANSMODEL*. Available online at: http://www.transmodel.org/en/transmodel/online.htm [15th November 2003]

FEAPMO. (2002). *The Business Reference Model Version 1.0: A Foundation for Government-wide Improvement*. Available online at: http://coverpages.org/ni2002-08-13-c.html[15th November 2003]

Firestone, J.M.(2000) Defining *the Enterprise Information Portal*. Available at: http://www.tgc.com/dsstar/00/0822/102054.html [10th October 2003]

Finkelstein, C. (2000). *The Emergence and Potential of Enterprise Information portals* (EIPs). Available Online at: http://www.infoworld.com/cgi-bin/displayStory.pl?/features/990125eip.htm

Gunter, A., Gunter, L., Jackson, M., and Zave, P. (2000). A Reference Model for Requirements and Specifications. *ACM Transactions on Software Engineering and Methodology*, 6(1): 1-30, January 2000

Hummingbird Ltd. (2000): *Hummingbird EIP. A Hummingbird White Paper*; Available Online at: http://www.hummingbird.com/whites/pdf/EIP_Whitepaper.pdf [15th January 2004]

IISME Consortium. (2000). *CRM System Reference Model for Small and Medium Enterprises*. Available Online at: http://www.vis.iao.fhg.de/vis-de/projects/iisme/index.htm . [15th November 2003].

Jacobson, I., Christerson, M., Jonsson, P., and Overgaard, G. (1992). *Object Oriented Software Engineering: A Use Case Driven Approach*. Wokingham: Addison-Wesley.

Lethbridge, T and Laganiere, R. (2001). *Object Oriented Software Engineering: Practical Software Development Using UML and Java.* England: Mc-Graw Hill Education.

Loucopoulos, P. and Karakostas, V. (1995). *System Requirement Engineering.* Berkshire: McGraw-Hill Book Company Europe.

Macaulay, A.L. (1996). *Requirement Engineering.* London: Springer-Verlag London

Matte, F. (2000). Enterprise Information Portals: A Brief Summary and White Paper Critical Reviews. *Technical Module Assignment:* Available at: http://www.magma.ca/~fmatte/Interests/Papers/EIP-paper.htm[ [10th October 2003]

McIntosh, S. (2003). *Qualitative Modelling for Requirement Engineering.* INCOSE UK Autumn Assembly.

Monk, A., and Howard, S. (1998). The Rich Picture: A Tool for Reasoning About Work Context. Interactions: Mac and April 1998. Available Online at: http://www.ics.uci.edu/~wscacchi/Software-Process/Readings/RichPicture.pdf

Murray, G. (1999). *The Portal is the Desktop.* Intraspect, Inc., Los Altos, CA

Nielsen, J.(1994). *Heuristic Evaluation.* Available at : http://www.useit.com/papers/heuristic/heuristic_evaluation..html [22nd December 2003]

Nielsen, J.(1995, June 27). T*echnology Transfer of Heuristic Evaluation and Usability Inspection.* Available Online at: http://www.useit.com/papers/heuristic/learning_inspection.html [15th December 2003]

Nielsen, J. and Landeur, T.K. (1993). A Mathematical Model of the finding of usability problems. *Proceedings* ACM/IFIP INTERCHI 1993 Conference (Amsterdam, The Netherlands, April 24-29), 206-213.

Pierotti, D.(1994). Usability Techniques: Heuristic Evaluation activities. Available at: http://www.stcsig.org/usability/topics/articles/he-activities.html. [14th February 2004]

Quatrani, T. (2000). *Visual Modeling with Rational Rose 2000 and UML*. Addison-Wesley

Reynolds, H., and Koulopoulos, T. (1999): *Enterprise knowledge has a face; Intelligent Enterprise*, Available Online at: http://www.intelligententerprise.com/993003/feat1.shtml [22th November 2003]

Roberts-Witt (1999). *Making Sense of Portal Pandemonium*: *What's behind the portal boom and what it means for the future of the enterprise.* Available Online at: http://www.destinationkm.com/articles/default.asp?ArticleID=84 [2nd October 2003]

Shilakes and Tylman (1998). *Sagemarker*. Available at: http://www.sagemarker.com [10th October 2003]

Scheucher, R.(2000). *A study on Information Portals*. Available at http://www2.iicm.edu/cguetl/education/projects/rscheuch/seminar/node21.html[21th December 2003]

Sommerville, I. (1995). Software Engineering (5th ed.). England: Addison-Wesley

Spitzer, T., (2000) *Vertical horizon. Surveying the landscape of online industry*; WEB Techniques; February, 2000, Available Online at: http://www.webtechniques.com/archives/2000/02/spitzer/index.html

*The Star* (July 8th, 2003). Most Council doesn't have Proper Websites.

Viador, Inc. (1999). *Enterprise Information Portals: Realizing the Vision of Information at your Fingertips.* Available Online at: http://knowledgemanagement.ittoolbox.com/documents/document.asp?i=1086 [6th October 2003]

Watkins, J. (2002). *Portal Evaluation Criteria.* Available Online at: https://evals.webmethods.com [12th January 2004]

Welling, L. and Thomson, L. (2001). *PHP and MySQL Web Development.* Indianapolis: Sams Publishing.

White. C. (January, 1999). *The Enterprise Information Portal Marketplace, Decision Processing Brief*:DP-99-01, Database Associates International, Inc., Morgan Hill, CA

Wieringa, R.J. (1996). *Requirement Engineering: Framework for Understanding.* England: John Wiley & Sons Ltd.

## APPENDICES

## APPENDIX A

1) Data Collection Procedures
Majlis Perbandaran Subang Jaya

**Evaluator 1**

| Heuristics | Number of "Yes" | Number of "No" | Number of "N/A" | Percentage = Number of "Yes" divided by number of "Yes" and "No" |
|---|---|---|---|---|
| **Heuristics 1** | 22 | 7 | 0 | 75.86 % |
| **Heuristics 2** | 15 | 7 | 2 | 68.18 % |
| **Heuristics 3** | 11 | 4 | 8 | 73.33 % |
| **Heuristics 4** | 35 | 11 | 5 | 76.09 % |
| **Heuristics 5** | 16 | 3 | 2 | 84.21 % |
| **Heuristics 6** | 10 | 3 | 2 | 76.92 % |
| **Heuristics 7** | 28 | 9 | 3 | 75.68 % |
| **Heuristics 8** | 6 | 9 | 1 | 40.0 % |
| **Heuristics 9** | 9 | 2 | 1 | 81.82 % |
| **Heuristics 10** | 15 | 5 | 3 | 75.0 % |

**Evaluator 2**

| Heuristics | Number of "Yes" | Number of "No" | Number of "N/A" | Percentage = Number of "Yes" divided by number of "Yes" and "No" |
|---|---|---|---|---|
| **Heuristics 1** | 15 | 10 | 4 | 60.0 % |
| **Heuristics 2** | 12 | 8 | 4 | 60.0 % |
| **Heuristics 3** | 13 | 10 | 0 | 56.52 % |
| **Heuristics 4** | 35 | 12 | 4 | 74.47 % |
| **Heuristics 5** | 10 | 9 | 2 | 52.63 % |

| | | | | |
|---|---|---|---|---|
| **Heuristics 6** | 10 | 5 | 0 | 66.67 % |
| **Heuristics 7** | 26 | 12 | 2 | 68.42 % |
| **Heuristics 8** | 8 | 6 | 2 | 57.14 % |
| **Heuristics 9** | 6 | 4 | 2 | 60.0 % |
| **Heuristics 10** | 12 | 7 | 4 | 63.16 % |

**Evaluator 3**

| Heuristics | Number of "Yes" | Number of "No" | Number of "N/A" | Percentage = Number of "Yes" divided by number of "Yes" and "No" |
|---|---|---|---|---|
| **Heuristics 1** | 12 | 10 | 7 | 54.55 % |
| **Heuristics 2** | 15 | 5 | 4 | 75.0 % |
| **Heuristics 3** | 14 | 5 | 4 | 73.68 % |
| **Heuristics 4** | 36 | 15 | 0 | 70.59 % |
| **Heuristics 5** | 10 | 9 | 2 | 52.63 % |
| **Heuristics 6** | 11 | 4 | 0 | 73.33 % |
| **Heuristics 7** | 30 | 10 | 0 | 75.0 % |
| **Heuristics 8** | 13 | 3 | 0 | 81.25 % |
| **Heuristics 9** | 10 | 2 | 0 | 83.33 % |
| **Heuristics 10** | 9 | 11 | 3 | 45.0 % |

**Mean Result**

| Heuristics | Results of First Evaluator | Results of Second Evaluator | Results of Third Evaluator | Mean of Result | Severity Rating |
|---|---|---|---|---|---|
| Heuristics 1 | 75.86 % | 60.0 % | 54.55 % | 63.47 % | 2 |
| Heuristics 2 | 68.18 % | 60.0 % | 75.0 % | 67.73 % | 2 |
| Heuristics 3 | 73.33 % | 56.52 % | 73.68 % | 67.84 % | 2 |
| Heuristics 4 | 76.09 % | 74.47 % | 70.59 % | 73.72 % | 2 |
| Heuristics 5 | 84.21 % | 52.63 % | 52.63 % | 63.16 % | 2 |
| Heuristics 6 | 76.92 % | 66.67 % | 73.33 % | 72.31 % | 2 |
| Heuristics 7 | 75.68 % | 68.42 % | 75.0 % | 73.03 % | 2 |
| Heuristics 8 | 40.0 % | 40.0 % | 81.25 % | 53.75 % | 2 |
| Heuristics 9 | 81.82 % | 60.0 % | 83.33 % | 75.05 % | 1 |
| Heuristics 10 | 75.0 % | 63.16 % | 45.0 % | 61.05 % | 2 |

2) Data Collection Procedures
Majlis Perbandaran Ampang Jaya

**Evaluator 1**

| Heuristics | Number of "Yes" | Number of "No" | Number of "N/A" | Percentage = Number of "Yes" divided by number of "Yes" and "No" |
|---|---|---|---|---|
| **Heuristics 1** | 24 | 5 | 0 | 82.76 % |
| **Heuristics 2** | 17 | 7 | 0 | 70.83 % |
| **Heuristics 3** | 17 | 2 | 4 | 89.47 % |
| **Heuristics 4** | 39 | 8 | 4 | 82.98 % |
| **Heuristics 5** | 14 | 4 | 3 | 77.78 % |
| **Heuristics 6** | 10 | 2 | 3 | 83.33 % |
| **Heuristics 7** | 27 | 8 | 5 | 77.14 % |

| | | | | |
|---|---|---|---|---|
| **Heuristics 8** | 14 | 2 | 0 | 87.50 % |
| **Heuristics 9** | 10 | 2 | 0 | 83.33 % |
| **Heuristics 10** | 19 | 4 | 0 | 82.61 % |

**Evaluator 2**

| Heuristics | Number of "Yes" | Number of "No" | Number of "N/A" | Percentage = Number of "Yes" divided by number of "Yes" and "No" |
|---|---|---|---|---|
| **Heuristics 1** | 19 | 10 | 0 | 65.52 % |
| **Heuristics 2** | 17 | 3 | 4 | 85.0 % |
| **Heuristics 3** | 12 | 9 | 2 | 57.14 % |
| **Heuristics 4** | 40 | 3 | 8 | 93.02 % |
| **Heuristics 5** | 17 | 3 | 1 | 85.0 % |
| **Heuristics 6** | 9 | 4 | 2 | 69.23 % |
| **Heuristics 7** | 36 | 4 | 0 | 90.0 % |
| **Heuristics 8** | 11 | 3 | 2 | 78.57 % |
| **Heuristics 9** | 11 | 1 | 0 | 91.67 % |
| **Heuristics 10** | 14 | 7 | 2 | 66.67 % |

**Evaluator 3**

| Heuristics | Number of "Yes" | Number of "No" | Number of "N/A" | Percentage = Number of "Yes" divided by number of "Yes" and "No" |
|---|---|---|---|---|
| **Heuristics 1** | 18 | 6 | 5 | 75.0 % |
| **Heuristics 2** | 14 | 2 | 8 | 87.5 % |
| **Heuristics 3** | 13 | 5 | 5 | 72.22 % |
| **Heuristics 4** | 45 | 2 | 4 | 95.74 % |

| | | | | |
|---|---|---|---|---|
| **Heuristics 5** | 16 | 3 | 2 | 84.21 % |
| **Heuristics 6** | 10 | 2 | 3 | 83.33 % |
| **Heuristics 7** | 32 | 6 | 2 | 84.21 % |
| **Heuristics 8** | 6 | 6 | 4 | 50.0 % |
| **Heuristics 9** | 10 | 2 | 0 | 83.33 % |
| **Heuristics 10** | 14 | 7 | 2 | 66.67 % |

**Mean Result**

| Heuristics | Results of First Evaluator | Results of Second Evaluator | Results of Third Evaluator | Mean of Result | Severity Rating |
|---|---|---|---|---|---|
| Heuristics 1 | 82.76 % | 65.52 % | 75.0 % | 74.43 % | 2 |
| Heuristics 2 | 70.83 % | 85.0 % | 87.5 % | 81.11 % | 1 |
| Heuristics 3 | 89.47 % | 57.14 % | 72.22 % | 72.94 % | 2 |
| Heuristics 4 | 82.98 % | 93.02 % | 95.74 % | 90.58 % | 1 |
| Heuristics 5 | 77.78 % | 85.0 % | 84.21 % | 82.33 % | 1 |
| Heuristics 6 | 83.33 % | 69.23 % | 83.33 % | 78.62 % | 1 |
| Heuristics 7 | 77.14 % | 90.0 % | 84.21 % | 83.78 % | 1 |
| Heuristics 8 | 87.50 % | 78.57 % | 50.0 % | 72.02 % | 2 |
| Heuristics 9 | 83.33 % | 91.67 % | 83.33 % | 86.11 % | 1 |
| Heuristics 10 | 82.61 % | 66.67 % | 66.67 % | 71.98 % | 2 |

3) Data Collection Procedures

Majlis Perbandaran Kota Star (MPKS)

**Evaluator 1**

| Heuristics | Number of "Yes" | Number of "No" | Number of "N/A" | Percentage = Number of "Yes" divided by number of "Yes" and "No" |
|---|---|---|---|---|
| **Heuristics 1** | 18 | 6 | 5 | 75.0 % |
| **Heuristics 2** | 13 | 4 | 7 | 76.47 % |
| **Heuristics 3** | 15 | 8 | 0 | 65.22 % |
| **Heuristics 4** | 33 | 9 | 9 | 78.57 % |
| **Heuristics 5** | 16 | 4 | 1 | 80.0 % |
| **Heuristics 6** | 6 | 6 | 3 | 50.0 % |
| **Heuristics 7** | 22 | 15 | 3 | 59.46 % |
| **Heuristics 8** | 5 | 9 | 2 | 35.71 % |
| **Heuristics 9** | 7 | 1 | 4 | 87.5 % |
| **Heuristics 10** | 12 | 7 | 4 | 63.16 % |

**Evaluator 2**

| Heuristics | Number of "Yes" | Number of "No" | Number of "N/A" | Percentage = Number of "Yes" divided by number of "Yes" and "No" |
|---|---|---|---|---|
| **Heuristics 1** | 19 | 8 | 2 | 70.37 % |
| **Heuristics 2** | 14 | 3 | 7 | 82.35 % |
| **Heuristics 3** | 10 | 11 | 2 | 47.62 % |
| **Heuristics 4** | 39 | 4 | 8 | 90.70 % |
| **Heuristics 5** | 13 | 7 | 1 | 65.0 % |
| **Heuristics 6** | 9 | 6 | 0 | 60.0 % |

| | | | | |
|---|---|---|---|---|
| **Heuristics 7** | 28 | 7 | 5 | 80.0 % |
| **Heuristics 8** | 6 | 9 | 1 | 40.0 % |
| **Heuristics 9** | 6 | 1 | 5 | 85.71 % |
| **Heuristics 10** | 14 | 7 | 2 | 66.67 % |

**Evaluator 3**

| Heuristics | Number of "Yes" | Number of "No" | Number of "N/A" | Percentage = Number of "Yes" divided by number of "Yes" and "No" |
|---|---|---|---|---|
| **Heuristics 1** | 18 | 6 | 5 | 75.0 % |
| **Heuristics 2** | 17 | 6 | 1 | 73.91 % |
| **Heuristics 3** | 9 | 9 | 5 | 50.0 % |
| **Heuristics 4** | 34 | 8 | 9 | 80.95 % |
| **Heuristics 5** | 10 | 8 | 3 | 55.56 % |
| **Heuristics 6** | 5 | 7 | 3 | 41.67 % |
| **Heuristics 7** | 22 | 11 | 7 | 66.67 % |
| **Heuristics 8** | 9 | 5 | 2 | 64.29 % |
| **Heuristics 9** | 8 | 1 | 3 | 88.89 % |
| **Heuristics 10** | 12 | 8 | 3 | 60.0 % |

**Mean Result**

| Heuristics | Results of First Evaluator | Results of Second Evaluator | Results of Third Evaluator | Mean of Result | Severity Rating |
|---|---|---|---|---|---|
| Heuristics 1 | 75.0 % | 70.37 % | 75.0 % | 73.46 % | 2 |
| Heuristics 2 | 76.47 % | 82.35 % | 73.91 % | 77.58 % | 1 |
| Heuristics 3 | 65.22 % | 47.62 % | 50.0 % | 54.28 % | 2 |
| Heuristics 4 | 78.57 % | 90.70 % | 80.95 % | 83.41 % | 1 |
| Heuristics 5 | 80.0 % | 65.0 % | 55.56 % | 66.85 % | 2 |
| Heuristics 6 | 50.0 % | 60.0 % | 41.67 % | 50.56 % | 2 |
| Heuristics 7 | 59.46 % | 80.0 % | 66.67 % | 68.71 % | 2 |
| Heuristics 8 | 35.71 % | 40.0 % | 64.29 % | 46.67 % | 3 |
| Heuristics 9 | 87.5 % | 85.71 % | 88.89 % | 87.37 % | 1 |
| Heuristics 10 | 63.16 % | 66.67 % | 60.0 % | 63.28 % | 2 |

**APPENDIX B**

**Heuristic Evaluation - A System Checklist**

By Deniese Pierotti, Xerox Corporation

**1. Visibility of System Status**

The system should always keep user informed about what is going on, through appropriate feedback within reasonable time.

| # | Review Checklist | Yes No N/A | Comments |
|---|---|---|---|
| 1.1 | Does every display begin with a title or header that describes screen contents? | O O O | |
| 1.2 | Is there a consistent icon design scheme and stylistic treatment across the system? | O O O | |
| 1.3 | Is a single, selected icon clearly visible when surrounded by unselected icons? | O O O | |
| 1.4 | Do menu instructions, prompts, and error messages appear in the same place(s) on each menu? | O O O | |
| 1.5 | In multipage data entry screens, is each page labeled to show its relation to others? | O O O | |
| 1.6 | If overtype and insert mode are both available, is there a visible indication of which one the user is in? | O O O | |
| 1.7 | If pop-up windows are used to display error messages, do they allow the user to see the field in error? | O O O | |
| 1.8 | Is there some form of system feedback for every operator action? | O O O | |
| 1.9 | After the user completes an action (or group of actions), does the feedback indicate that the next group of actions can be started? | O O O | |
| 1.10 | Is there visual feedback in menus or dialog boxes about which choices are selectable? | O O O | |
| 1.11 | Is there visual feedback in menus or dialog boxes about which choice the cursor is on now? | O O O | |

| | | | |
|---|---|---|---|
| 1.12 | If multiple options can be selected in a menu or dialog box, is there visual feedback about which options are already selected? | O O O | |
| 1.13 | Is there visual feedback when objects are selected or moved? | O O O | |
| 1.14 | Is the current status of an icon clearly indicated? | O O O | |
| **#** | **Review Checklist** | **Yes No N/A** | **Comments** |
| 1.15 | Is there feedback when function keys are pressed? | O O O | |
| 1.16 | If there are observable delays (greater than fifteen seconds) in the system's response time, is the user kept informed of the system's progress? | O O O | |
| 1.17 | Are response times appropriate to the task? | O O O | |
| 1.18 | Typing, cursor motion, mouse selection: 50-1 50 milliseconds | O O O | |
| 1.19 | Simple, frequent tasks: less than 1 second | O O O | |
| 1.20 | Common tasks: 2-4 seconds | O O O | |
| 1.21 | Complex tasks: 8-12 seconds | O O O | |
| 1.22 | Are response times appropriate to the user's cognitive processing? | O O O | |
| 1.23 | Continuity of thinking is required and information must be remembered throughout several responses: less than two seconds. | O O O | |
| 1.24 | High levels of concentration aren't necessary and remembering information is not required: two to fifteen seconds. | O O O | |
| 1.25 | Is the menu-naming terminology consistent with the user's task domain? | O O O | |
| 1.26 | Does the system provide *visibility:* that is, by looking, can the user tell the state of the system and the alternatives for action? | O O O | |
| 1.27 | Do GUI menus make obvious which item has been selected? | O O O | |
| 1.28 | Do GUI menus make obvious whether deselection | O O O | |

| | | | |
|---|---|---|---|
| | is possible? | | |
| 1.29 | If users must navigate between multiple screens, does the system use context labels, menu maps, and place markers as navigational aids? | O O O | |

**2. Match Between System and the Real World**

The system should speak the user's language, with words, phrases and concepts familiar to the user, rather than system-oriented terms. Follow real-world conventions, making information appear in a natural and logical order.

| # | Review Checklist | Yes No N/A | Comments |
|---|---|---|---|
| 2.1 | Are icons concrete and familiar? | O O O | |
| 2.2 | Are menu choices ordered in the most logical way, given the user, the item names, and the task variables? | O O O | |
| 2.3 | If there is a natural sequence to menu choices, has it been used? | O O O | |
| 2.4 | Do related and interdependent fields appear on the same screen? | O O O | |
| 2.5 | If shape is used as a visual cue, does it match cultural conventions? | O O O | |
| 2.6 | Do the selected colors correspond to common expectations about color codes? | O O O | |
| 2.7 | When prompts imply a necessary action, are the words in the message consistent with that action? | O O O | |
| 2.8 | Do keystroke references in prompts match actual key names? | O O O | |
| 2.9 | On data entry screens, are tasks described in terminology familiar to users? | O O O | |
| 2.10 | Are field-level prompts provided for data entry screens? | | |
| 2.11 | For question and answer interfaces, are questions stated in clear, simple language? | O O O | |

| | | | |
|---|---|---|---|
| 2.12 | Do menu choices fit logically into categories that have readily understood meanings? | O O O | |
| 2.13 | Are menu titles parallel grammatically? | O O O | |
| 2.14 | Does the command language employ user jargon and avoid computer jargon? | O O O | |
| 2.15 | Are command names specific rather than general? | O O O | |
| 2.16 | Does the command language allow both full names and abbreviations? | O O O | |
| 2.17 | Are input data codes meaningful? | O O O | |
| 2.18 | Have uncommon letter sequences been avoided whenever possible? | O O O | |
| 2.19 | Does the system automatically enter leading or trailing spaces to align decimal points? | O O O | |
| 2.20 | Does the system automatically enter a dollar sign and decimal for monetary entries? | O O O | |

| # | Review Checklist | Yes No N/A | Comments |
|---|---|---|---|
| 2.21 | Does the system automatically enter commas in numeric values greater than 9999? | O O O | |
| 2.22 | Do GUI menus offer activation: that is, make obvious how to say *"now do it"?* | O O O | |
| 2.23 | Has the system been designed so that keys with similar names do not perform opposite (and potentially dangerous) actions? | O O O | |
| 2.24 | Are function keys labeled clearly and distinctively, even if this means breaking consistency rules? | O O O | |

**3. User Control and Freedom**

Users should be free to select and sequence tasks (when appropriate), rather than having the system do this for them. Users often choose system functions by mistake and will need a clearly marked "emergency exit" to leave the unwanted state without having to go through an extended dialogue. Users should make their own decisions (with clear information) regarding the costs of exiting current work. The system should support undo and redo.

| # | Review Checklist | Yes No N/A | Comments |
|---|---|---|---|
| 3.1 | If setting up windows is a low-frequency task, is it particularly easy to remember? | O O O | |
| 3.2 | In systems that use overlapping windows, is it easy for users to rearrange windows on the screen? | O O O | |
| 3.3 | In systems that use overlapping windows, is it easy for users to switch between windows? | O O O | |
| 3.4 | When a user's task is complete, does the system wait for a signal from the user before processing? | O O O | |
| 3.5 | Can users type-ahead in a system with many nested menus? | O O O | |
| 3.6 | Are users prompted to confirm commands that have drastic, destructive consequences? | O O O | |
| 3.7 | Is there an "undo" function at the level of a single action, a data entry, and a complete group of actions? | O O O | |
| 3.8 | Can users cancel out of operations in progress? | O O O | |
| 3.9 | Are character edits allowed in commands? | O O O | |
| 3.10 | Can users reduce data entry time by copying and modifying existing data? | O O O | |
| 3.11 | Are character edits allowed in data entry fields? | O O O | |
| 3.12 | If menu lists are long (more than seven items), can users select an item either by moving the cursor or by typing a mnemonic code? | O O O | |
| 3.13 | If the system uses a pointing device, do users have the option of either clicking on menu items or using a keyboard shortcut? | O O O | |
| 3.14 | Are menus broad (many items on a menu) rather | O O O | |

| | | | |
|---|---|---|---|
| | than deep (many menu levels)? | | |
| 3.15 | If the system has multiple menu levels, is there a mechanism that allows users to go back to previous menus? | O O O | |

| # | Review Checklist | Yes No N/A | Comments |
|---|---|---|---|
| 3.16 | If users can go back to a previous menu, can they change their earlier menu choice? | O O O | |
| 3.17 | Can users move forward and backward between fields or dialog box options? | O O O | |
| 3.18 | If the system has multipage data entry screens, can users move backward and forward among all the pages in the set? | O O O | |
| 3.19 | If the system uses a question and answer interface, can users go back to previous questions or skip forward to later questions? | O O O | |
| 3.20 | Do function keys that can cause serious consequences have an undo feature? | O O O | |
| 3.21 | Can users easily reverse their actions? | O O O | |
| 3.22 | If the system allows users to reverse their actions, is there a retracing mechanism to allow for multiple undos? | O O O | |
| 3.23 | Can users set their own system, session, file, and screen defaults? | O O O | |

**4. Consistency and Standards**

Users should not have to wonder whether different words, situations, or actions mean the same thing. Follow platform conventions.

| # | Review Checklist | Yes No N/A | Comments |
|---|---|---|---|
| 4.1 | Have industry or company formatting standards been followed consistently in all screens within a system? | O O O | |
| 4.2 | Has a heavy use of all uppercase letters on a | O O O | |

| | | | |
|---|---|---|---|
| | screen been avoided? | | |
| 4.3 | Do abbreviations not include punctuation? | O O O | |
| 4.4 | Are integers right-justified and real numbers decimal-aligned? | O O O | |
| 4.5 | Are icons labeled? | O O O | |
| 4.6 | Are there no more than twelve to twenty icon types? | O O O | |
| 4.7 | Are there salient visual cues to identify the active window? | O O O | |
| 4.8 | Does each window have a title? | O O O | |
| 4.9 | Are vertical and horizontal scrolling possible in each window? | O O O | |
| 4.10 | Does the menu structure match the task structure? | O O O | |
| 4.11 | Have industry or company standards been established for menu design, and are they applied consistently on all menu screens in the system? | O O O | |
| 4.12 | Are menu choice lists presented vertically? | O O O | |
| 4.13 | If "exit" is a menu choice, does it always appear at the bottom of the list? | O O O | |
| 4.14 | Are menu titles either centered or left-justified? | O O O | |
| 4.15 | Are menu items left-justified, with the item number or mnemonic preceding the name? | O O O | |
| 4.16 | Do embedded field-level prompts appear to the right of the field label? | O O O | |
| 4.17 | Do on-line instructions appear in a consistent location across screens? | O O O | |
| 4.18 | Are field labels and fields distinguished typographically? | O O O | |
| 4.19 | Are field labels consistent from one data entry screen to another? | O O O | |
| 4.20 | Are fields and labels left-justified for alpha lists and right-justified for numeric lists? | O O O | |

| # | Review Checklist | Yes No N/A | Comments |
|---|---|---|---|
| 4.21 | Do field labels appear to the left of single fields and above list fields? | O O O | |
| 4.22 | Are attention-getting techniques used with care? | O O O | |
| 4.23 | Intensity: two levels only | O O O | |
| 4.24 | Size: up to four sizes | O O O | |
| 4.25 | Font: up to three | O O O | |
| 4.26 | Blink: two to four hertz | O O O | |
| 4.27 | Color: up to four (additional colors for occasional use only) | O O O | |
| 4.28 | Sound: soft tones for regular positive feedback, harsh for rare critical conditions | O O O | |
| 4.29 | Are attention-getting techniques used only for exceptional conditions or for time-dependent information? | O O O | |
| 4.30 | Are there no more than four to seven colors, and are they far apart along the visible spectrum? | O O O | |
| 4.31 | Is a legend provided if color codes are numerous or not obvious in meaning? | O O O | |
| 4.32 | Have pairings of high-chroma, spectrally extreme colors been avoided? | O O O | |
| 4.33 | Are saturated blues avoided for text or other small, thin line symbols? | O O O | |
| 4.34 | Is the most important information placed at the beginning of the prompt? | O O O | |
| 4.35 | Are user actions named consistently across all prompts in the system? | O O O | |
| 4.36 | Are system objects named consistently across all prompts in the system? | O O O | |
| 4.37 | Do field-level prompts provide more information than a restatement of the field name? | O O O | |
| 4.38 | For question and answer interfaces, are the valid | O O O | |

| | | | |
|---|---|---|---|
| | inputs for a question listed? | | |
| 4.39 | Are menu choice names consistent, both within each menu and across the system, in grammatical style and terminology? | O O O | |
| 4.40 | Does the structure of menu choice names match their corresponding menu titles? | O O O | |
| 4.41 | Are commands used the same way, and do they mean the same thing, in all parts of the system? | O O O | |
| 4.42 | Does the command language have a consistent, natural, and mnemonic syntax? | O O O | |
| 4.43 | Do abbreviations follow a simple primary rule and, if necessary, a simple secondary rule for abbreviations that otherwise would be duplicates? | O O O | |
| **#** | **Review Checklist** | **Yes No N/A** | **Comments** |
| 4.44 | Is the secondary rule used only when necessary? | O O O | |
| 4.45 | Are abbreviated words all the same length? | O O O | |
| 4.46 | Is the structure of a data entry value consistent from screen to screen? | O O O | |
| 4.47 | Is the method for moving the cursor to the next or previous field consistent throughout the system? | O O O | |
| 4.48 | If the system has multipage data entry screens, do all pages have the same title? | O O O | |
| 4.49 | If the system has multipage data entry screens, does each page have a sequential page number? | O O O | |
| 4.50 | Does the system follow industry or company standards for function key assignments? | O O O | |
| 4.51 | Are high-value, high-chroma colors used to attract attention? | O O O | |

**5. Help Users Recognize, Diagnose, and Recover From Errors**

Error messages should be expressed in plain language (NO CODES).

| # | Review Checklist | Yes No N/A | Comments |
|---|---|---|---|
| 5.1 | Is sound used to signal an error? | O O O | |
| 5.2 | Are prompts stated constructively, without overt or implied criticism of the user? | O O O | |
| 5.3 | Do prompts imply that the user is in control? | O O O | |
| 5.4 | Are prompts brief and unambiguous. | O O O | |
| 5.5 | Are error messages worded so that the system, not the user, takes the blame? | O O O | |
| 5.6 | If humorous error messages are used, are they appropriate and inoffensive to the user population? | O O O | |
| 5.7 | Are error messages grammatically correct? | O O O | |
| 5.8 | Do error messages avoid the use of exclamation points? | O O O | |
| 5.9 | Do error messages avoid the use of violent or hostile words? | O O O | |
| 5.10 | Do error messages avoid an anthropomorphic tone? | O O O | |
| 5.11 | Do all error messages in the system use consistent grammatical style, form, terminology, and abbreviations? | O O O | |
| 5.12 | Do messages place users in control of the system? | O O O | |
| 5.13 | Does the command language use normal action-object syntax? | O O O | |
| 5.14 | Does the command language avoid arbitrary, non-English use of punctuation, except for symbols that users already know? | O O O | |
| 5.15 | If an error is detected in a data entry field, does the system place the cursor in that field or highlight the error? | O O O | |
| 5.16 | Do error messages inform the user of the error's severity? | O O O | |
| 5.17 | Do error messages suggest the cause of the | O O O | |

| | | | |
|---|---|---|---|
| | problem? | | |
| 5.18 | Do error messages provide appropriate semantic information? | O O O | |
| 5.19 | Do error messages provide appropriate syntactic information? | O O O | |
| 5.20 | Do error messages indicate what action the user needs to take to correct the error? | O O O | |
| 5.21 | If the system supports both novice and expert users, are multiple levels of error-message detail available? | O O O | |

**6. Error Prevention**

Even better than good error messages is a careful design which prevents a problem from occurring in the first place.

| # | Review Checklist | Yes No N/A | Comments |
|---|---|---|---|
| 6.1 | If the database includes groups of data, can users enter more than one group on a single screen? | O O O | |
| 6.2 | Have dots or underscores been used to indicate field length? | O O O | |
| 6.3 | Is the menu choice name on a higher-level menu used as the menu title of the lower-level menu? | O O O | |
| 6.4 | Are menu choices logical, distinctive, and mutually exclusive? | O O O | |
| 6.5 | Are data inputs case-blind whenever possible? | O O O | |
| 6.6 | If the system displays multiple windows, is navigation between windows simple and visible? | O O O | |
| 6.7 | Are the function keys that can cause the most serious consequences in hard-to-reach positions? | O O O | |
| 6.8 | Are the function keys that can cause the most serious consequences located far away from low-consequence and high-use keys? | O O O | |
| 6.9 | Has the use of qualifier keys been minimized? | O O O | |

| | | | |
|---|---|---|---|
| 6.10 | If the system uses qualifier keys, are they used consistently throughout the system? | O O O | |
| 6.11 | Does the system prevent users from making errors whenever possible? | O O O | |
| 6.12 | Does the system warn users if they are about to make a potentially serious error? | O O O | |
| 6.13 | Does the system intelligently interpret variations in user commands? | O O O | |
| 6.14 | Do data entry screens and dialog boxes indicate the number of character spaces available in a field? | O O O | |
| 6.15 | Do fields in data entry screens and dialog boxes contain default values when appropriate? | O O O | |

**7. Recognition Rather Than Recall**

Make objects, actions, and options visible. The user should not have to remember information from one part of the dialogue to another. Instructions for use of the system should be visible or easily retrievable whenever appropriate.

| # | Review Checklist | Yes No N/A | Comments |
|---|---|---|---|
| 7.1 | For question and answer interfaces, are visual cues and white space used to distinguish questions, prompts, instructions, and user input? | O O O | |
| 7.2 | Does the data display start in the upper-left corner of the screen? | O O O | |
| 7.3 | Are multiword field labels placed horizontally (not stacked vertically)? | O O O | |
| 7.4 | Are all data a user needs on display at each step in a transaction sequence? | O O O | |
| 7.5 | Are prompts, cues, and messages placed where the eye is likely to be looking on the screen? | O O O | |
| 7.6 | Have prompts been formatted using white space, justification, and visual cues for easy scanning? | O O O | |
| 7.7 | Do text areas have "breathing space" around | O O O | |

| | | | |
|---|---|---|---|
| | them? | | |
| 7.8 | Is there an obvious visual distinction made between "choose one" menu and "choose many" menus? | O O O | |
| 7.9 | Have spatial relationships between soft function keys (on-screen cues) and keyboard function keys been preserved? | O O O | |
| 7.10 | Does the system gray out or delete labels of currently inactive soft function keys? | O O O | |
| 7.11 | Is white space used to create symmetry and lead the eye in the appropriate direction? | O O O | |
| 7.12 | Have items been grouped into logical zones, and have headings been used to distinguish between zones? | O O O | |
| 7.13 | Are zones no more than twelve to fourteen characters wide and six to seven lines high? | O O O | |
| 7.14 | Have zones been separated by spaces, lines, color, letters, bold titles, rules lines, or shaded areas? | O O O | |
| 7.15 | Are field labels close to fields, but separated by at least one space? | O O O | |
| 7.16 | Are long columnar fields broken up into groups of five, separated by a blank line? | O O O | |
| 7.17 | Are optional data entry fields clearly marked? | O O O | |
| 7.18 | Are symbols used to break long input strings into "chunks"? | O O O | |
| 7.19 | Is reverse video or color highlighting used to get the user's attention? | O O O | |
| 7.20 | Is reverse video used to indicate that an item has been selected? | O O O | |
| 7.21 | Are size, boldface, underlining, color, shading, or typography used to show relative quantity or importance of different screen items? | O O O | |
| 7.22 | Are borders used to identify meaningful groups? | O O O | |
| 7.23 | Has the same color been used to group related | O O O | |

| | | | |
|---|---|---|---|
| | elements? | | |
| 7.24 | Is color coding consistent throughout the system? | O O O | |
| 7.25 | Is color used in conjunction with some other redundant cue? | O O O | |
| 7.26 | Is there good color and brightness contrast between image and background colors? | O O O | |
| 7.27 | Have light, bright, saturated colors been used to emphasize data and have darker, duller, and desaturated colors been used to de-emphasize data? | O O O | |
| 7.28 | Is the first word of each menu choice the most important? | O O O | |
| 7.29 | Does the system provide *mapping:* that is, are the relationships between controls and actions apparent to the user? | O O O | |
| 7.30 | Are input data codes distinctive? | O O O | |
| 7.31 | Have frequently confused data pairs been eliminated whenever possible? | O O O | |
| 7.32 | Have large strings of numbers or letters been broken into chunks? | O O O | |
| 7.33 | Are inactive menu items grayed out or omitted? | O O O | |
| 7.34 | Are there menu selection defaults? | O O O | |
| 7.35 | If the system has many menu levels or complex menu levels, do users have access to an on-line spatial menu map? | O O O | |
| 7.36 | Do GUI menus offer affordance: that is, make obvious where selection is possible? | O O O | |
| 7.37 | Are there salient visual cues to identify the active window? | O O O | |
| 7.38 | Are function keys arranged in logical groups? | O O O | |
| 7.39 | Do data entry screens and dialog boxes indicate when fields are optional? | O O O | |
| 7.40 | On data entry screens and dialog boxes, are dependent fields displayed only when necessary? | O O O | |

**8. Flexibility and Minimalist Design**

Accelerators-unseen by the novice user-may often speed up the interaction for the expert user such that the system can cater to both inexperienced and experienced users. Allow users to tailor frequent actions. Provide alternative means of access and operation for users who differ from the "average" user (e.g., physical or cognitive ability, culture, language, etc.)

| # | Review Checklist | Yes No N/A | Comments |
|---|---|---|---|
| 8.1 | If the system supports both novice and expert users, are multiple levels of error message detail available? | O O O | |
| 8.2 | Does the system allow novices to use a keyword grammar and experts to use a positional grammar? | O O O | |
| 8.3 | Can users define their own synonyms for commands? | O O O | |
| 8.4 | Does the system allow novice users to enter the simplest, most common form of each command, and allow expert users to add parameters? | O O O | |
| 8.5 | Do expert users have the option of entering multiple commands in a single string? | O O O | |
| 8.6 | Does the system provide function keys for high-frequency commands? | O O O | |
| 8.7 | For data entry screens with many fields or in which source documents may be incomplete, can users save a partially filled screen? | O O O | |
| 8.8 | Does the system automatically enter leading zeros? | O O O | |
| 8.9 | If menu lists are short (seven items or fewer), can users select an item by moving the cursor? | O O O | |
| 8.10 | If the system uses a type-ahead strategy, do the menu items have mnemonic codes? | O O O | |
| 8.11 | If the system uses a pointing device, do users | O O O | |

| | | | |
|---|---|---|---|
| | have the option of either clicking on fields or using a keyboard shortcut? | | |
| 8.12 | Does the system offer "find next" and "find previous" shortcuts for database searches? | O O O | |
| 8.13 | On data entry screens, do users have the option of either clicking directly on a field or using a keyboard shortcut? | O O O | |
| 8.14 | On menus, do users have the option of either clicking directly on a menu item or using a keyboard shortcut? | O O O | |
| 8.15 | In dialog boxes, do users have the option of either clicking directly on a dialog box option or using a keyboard shortcut? | O O O | |
| 8.16 | Can expert users bypass nested dialog boxes with either type-ahead, user-defined macros, or keyboard shortcuts? | O O O | |

**9. Aesthetic and Minimalist Design**

Dialogues should not contain information which is irrelevant or rarely needed. Every extra unit of information in a dialogue competes with the relevant units of information and diminishes their relative visibility.

| # | Review Checklist | Yes No N/A | Comments |
|---|---|---|---|
| 9.1 | Is only (and all) information essential to decision making displayed on the screen? | O O O | |
| 9.2 | Are all icons in a set visually and conceptually distinct? | O O O | |
| 9.3 | Have large objects, bold lines, and simple areas been used to distinguish icons? | O O O | |
| 9.4 | Does each icon stand out from its background? | O O O | |
| 9.5 | If the system uses a standard GUI interface where menu sequence has already been specified, do menus adhere to the specification whenever possible? | O O O | |

| | | | |
|---|---|---|---|
| 9.6 | Are meaningful groups of items separated by white space? | O O O | |
| 9.7 | Does each data entry screen have a short, simple, clear, distinctive title? | O O O | |
| 9.8 | Are field labels brief, familiar, and descriptive? | O O O | |
| 9.9 | Are prompts expressed in the affirmative, and do they use the active voice? | O O O | |
| 9.10 | Is each lower-level menu choice associated with only one higher level menu? | O O O | |
| 9.11 | Are menu titles brief, yet long enough to communicate? | O O O | |
| 9.12 | Are there pop-up or pull-down menus within data entry fields that have many, but well-defined, entry options? | O O O | |

**10. Help and Documentation**

Even though it is better if the system can be used without documentation, it may be necessary to provide help and documentation. Any such information should be easy to search, focused on the user's task, list concrete steps to be carried out, and not be too large.

| # | Review Checklist | Yes No N/A | Comments |
|---|---|---|---|
| 10.1 | If users are working from hard copy, are the parts of the hard copy that go on-line marked? | O O O | |
| 10.2 | Are on-line instructions visually distinct? | O O O | |
| 10.3 | Do the instructions follow the sequence of user actions? | O O O | |
| 10.4 | If menu choices are ambiguous, does the system provide additional explanatory information when an item is selected? | O O O | |
| 10.5 | Are data entry screens and dialog boxes supported by navigation and completion instructions? | O O O | |
| 10.6 | If menu items are ambiguous, does the system | O O O | |

| | | | |
|---|---|---|---|
| | provide additional explanatory information when an item is selected? | | |
| 10.7 | Are there memory aids for commands, either through on-line quick reference or prompting? | O O O | |
| 10.8 | Is the help function visible; for example, a key labeled HELP or a special menu? | O O O | |
| 10.9 | Is the help system interface (navigation, presentation, and conversation) consistent with the navigation, presentation, and conversation interfaces of the application it supports? | O O O | |
| 10.10 | Navigation: Is information easy to find? | O O O | |
| 10.11 | Presentation: Is the visual layout well designed? | O O O | |
| 10.12 | Conversation: Is the information accurate, complete, and understandable? | O O O | |
| **#** | **Review Checklist** | **Yes No N/A** | **Comments** |
| 10.13 | Is the information relevant? | O O O | |
| 10.14 | Goal-oriented (What can I do with this program?) | O O O | |
| 10.15 | Descriptive (What is this thing for?) | O O O | |
| 10.16 | Procedural (How do I do this task?) | O O O | |
| 10.17 | Interpretive (Why did that happen?) | O O O | |
| 10.18 | Navigational (Where am I?) | O O O | |
| 10.19 | Is there context-sensitive help? | O O O | |
| 10.20 | Can the user change the level of detail available? | O O O | |
| 10.21 | Can users easily switch between help and their work? | O O O | |
| 10.22 | Is it easy to access and return from the help system? | O O O | |
| 10.23 | Can users resume work where they left off after accessing help? | O O O | |

System Title:_____

Evaluator: _____

Date: _____