Sdlc for food ordering system assignment

Introduction Generally, as it was mentioned in our first assignment, we supposed to choose any organization and try to develop an accurate system for it. We chose " Taktaz Restaurant" which serves Iranian traditional foods as our target and decided to develop an " online food ordering system" for it. We use Structured Analysis Method, which is time-tested and easy to understand method, as our system developing technique. Structured Analysis uses a series of phases, called the system development life cycle (SDLC), to plan, analyze, design, implement, and support an information system.

In the first assignment we asked to concentrate on the first two phases of SDLC, planning and analyzing. In this assignment we are going to do last three phases, which are design, implement and support. The most important parts of these three phases that we are supposed to do in this assignment are as following: a. Output and user interface design b. Data design – ER Diagram c. System architecture d. System implementation (system prototype) – specify the hardware, software and platform used. e. System support maintenance Output Design Types of output

In the system design phase, we will create the actual forms, reports, documents and other types of output. During this process we must consider the format and how it will be delivered, stored and retrieved. 1. Internet-Base Information Delivery Output In our information system we used internet-base information delivery as the output type. To support our online food ordering system, we have provided user-friendly screen interfaces that display output and accept input from customers. Customers visiting the site

can check available food items and their prices in our menu, then can register for online ordering. 2. Screen Report Output

Our information system also provides daily reports for Taktaz manager. To be useful, a report must include the information that a user need. Our system provides daily detailed reports in two categories: i. Total amount of daily orders ii. Total number of daily orders grouped by different customers' name The examples of our system detailed report are shown in following. Total amount of daily orders Total number of daily orders grouped by different customers' name As you can see, our system detailed report provides total for numeric fields. It also adds tax percentage to total selling amount for giving better estimation.

The report also have auto incremental control field to controls output. User Interface Design Although output design involves a separate set of physical design issues, it is a part of the user interface (UI). User interface describes how users interact with a computer system and consists of all the hardware, software, screens, menus, functions, output and features that affect two-way communications between the user and the computer. Our customers through the website will work with a varied mix of input, output and, indirectly, date queries as they perform ordering process.

Because all those tasks require interaction with the computer system, the user interface is a vital element in the systems design phase. It is important to design a user interface that is easy to use, attractive and efficient. To achieve these goals we have used following rules: 1. Focusing on basic objectives According to this rule, we tried to create a simple design that is

easy to learn and understand. We aimed to design a logical and attractive layout as well. 2. Building an interface that is easy to learn and use. Based on this rule, we only used images that users can understand easily, to dentify icons. We also made it easy to return to any level in our menu structure system. 3. Minimizing input data problems We provided data validation checks and event-driven message and reminders as you can see in below sample from our system screen shot. 4. Providing feedback to users According to this rule, we made all of our feedback messages specific, understandable and professional. We also displayed messages on top of the screen and follow consistency every time. 5. Designing an attractive layout To satisfy this rule we tried to fulfill following guidelines in layouts of our system. Use appropriate colors to highlight different areas of the screen. •Use hyperlink that allows user to jump to related topics. •Group related objects and information by using similarity and proximity principles of Gestalt Theory. •When the user enters data that completely files the field, don't move automatically to the next field. We asked users to confirm the entries by pressing ok icon and tab key at the end of every fill-in field. User Interface Controls We include many control features, such as: •menu bar or •command button •text box •drop-down list box

Date design Basic understanding of data design concepts includes data structure and the characteristics of file processing and database system which including web-based database design, will help the system analyst to come out with good and fulfilled information system. A data structure is a framework for organizing and storing data in an information system. Typically it contains files or tables that are linked in various ways. These files

or table is data all about people, places, things or events that interact with the information system.

A file processing system or more known as a file-oriented system, stores and manages data in one or more separate files. While, a database system consists of linked tables that form one overall data structure which offers great flexibility and efficiency in processing data. Database Management System (DBMS) is a collection of tools, features and interfaces that enable users to add, update, manage access and analyze the contents of a set of data. Data manipulation language is one of DBMS component which controls database operations, including storing, retrieving, updating and deleting data.

Moreover, the complete definition of a database together with descriptions of all fields, tables and relationship, is called a schema. A subschema is a view of the database used by one or more systems or users and typically classifies only those portions of the database that a particular system or user needs or is allowed to access. Data dictionary •'empprofile' Table This table supposed to store information about the staff of Taktaz. If the value for " active" column of this table become ' a', then the proposed staff can log in as an administrator. •'foodmnu' Table This table stores data of available foods in the food menu. ' order' Table This table contains data of the ordered items. • ' userprofile' Table This table stores information of registered customers. Entity Relationship Diagram (ERD) An entity-relationship (ER) diagram is a specialized graphic that illustrates the interrelationships between entities in a database. To design a system it is necessary to uses these models during the requirements analysis to describe information needs or the type of

information that is to be stored in a database The entities for our system would be customers, order, bill, inventory and kitchen.

The relations & the attributes are shown in the following ERD diagram. Use Case Diagram Use case diagram is a type of behavioral diagram defined by and created from a Use-case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases. The users of our system are customers, manager, kitchen staffs & IT supporters. According to our systems point of view we can implement a use case diagram as given below. System Architecture

System architecture is the conceptual model that defines the structure, behavior, and more views of a system. The system architecture includes seven issues which are as follow: 1. Enterprise resource planning (ERP) 2. Initial and total cost of ownership (TCO) 3. Scalability 4. Web Integration 5. Legacy system interface requirements 6. Processing options 7. Security issues The Taktaz Restaurant had no online reservation or ordering system for customers, so to have such a system, we are required to design completely a new online system (or a website).

The first step was to analyze the system which the restaurant was working for. Everything in Taktaz, buying the ingredients , preparation , ordering and delivery process, was done manually. We needed to have the owner's idea about the old system and the expectation of the new system. By knowing a clear idea of the future system, we started to design an online food ordering system that fulfilled the restaurant requirement and expectations.

Developing a new system has its own cost and we, as designer should have deal with it.

The desire system and design must be cost-effective to the owner and also to us to design, operate and perform the system. Also the maintenance cost should be noticed in the designing stage. The new system should be flexible enough so that the users (restaurant's owner and staff) can easily work with it and if needed, make some changes in it, expand, change or even downsize to meet the changing needs of the business. We tried to consider all of the above mentioned issues in the designing the system. However, some of them needed and took much more time than the others.

For instance, to perform scalability or the last mentioned issue, we must interview with the owner and staff of the restaurant to get their idea and expectation of the oncoming system and then, we could interpolate those ideas to the system. As a designer, we should consider many more issues in the developing of the system. First of all, we should notice how user friendly the website will be. In a project, especially in online ordering, it is extremely important that how user deal with the website, whether the user can use and access the different section of the website in the first attempts or not.

It was one of our main concerns. The administrator section also was one of our concerns. This section, which is the management section, must be designed in a way that the manager (or website controller) can easily work and modify the system. The website should be as simple as possible so that the first time users can easily understand how to work with it. Also it must be

as easy to use as possible for administrator to manage the ordering website. We should design a system with a minimum number of errors.

To achieve it, we should first accurately analyze the requirements and expectations of the restaurant's owner and after gathering all different information, and then we start to design a website. Designing of the website needed to be looked from two perspectives: as administrator and as customer. Obviously a website with many errors is not a comfortable website for neither of both users. From the password protection to billing the order, security threats and defenses were a major concern to us. As analysts and designers, we should consider and determine how the business (restaurant) will deal with the security threats.

Users (usually customers), they put their personal information and details in the system. So they must do it trustfully and have no doubt about the security of the system. Our System Architecture Diagram is coming in below. System Architecture Diagram System Implementation Prototyping produces an early, rapidly constructed working version of the proposed information system. Prototyping, which involves a repetitive sequence of analysis, design, modeling and testing is a common technique that we used to design our system. Our system is being run on windows as its Platform.

In addition, since it is based on the internet connections, the Network Card must be installed and used. A KEY must be defined and generated for security purposes and boot strap to prevent entering any unauthorized parties and any viruses and worms. We used PHP and our IDE is Dreamviwer. For our database we use MySQL. System Support Maintenance System

support maintenance is the modification of the system. We have to correct faults, improve system's performance and adapt the system to environment and requirements changes.

The objective of our maintenance is to keep our system always running at an acceptable level. Corrective maintenance diagnoses and corrects errors operational levels. In this type of maintenance we will try to correct Design errors, Coding errors and Logic errors. Applying the Corrective maintenance can be helpful to avoid probable bugs and to fix logic errors. . In our project, which is Online Food Ordering System, the first step of support maintenance is to ensure that the issues related to programming and coding are correctly run.

Debugging and troubleshooting can be the second step. The coding is the most important part to check especially when you are dealing with financial issues related to ordering. Therefore, we must be sure about the correctness of all functions and modules in our coding and proper connection with database to store and retrieve data. In addition to the software hard ware must also be checked regularly. Adaptive maintenance adds enhancements to an operational system and makes the system easier to use. Adaptive maintenance entails changes related to our application's operating environment.

These modifications cam consist of Incorporating new operating systems, Hardware configuration changes, Regulations & Localizations alterations and Changes in data format. As our system goes on we need to enhance our system with new features, capability and functions in response to new

technology. Adaptive maintenance in our system is related to update new features. For example, adding new meals or recipes to the current menu, can ensure the capability of our system. Also when we need to upgrade or add new functions to the system we need to apply the adaptive maintenance.

Perfective maintenance involves changing an operational system to make it more efficient, reliable or maintainable. Perfective maintenance can improve maintainability and performance. It involves computer programmers who work to improve the way software program functions or how quickly it processes requests. For example to change our menu layouts, command interfaces for some functions or changing the number of daily orders. The other maintenance type which is useful for our system is preventative maintenance.

In this kind of maintenance we try to predict and avoid probable problems before occurring . For example, to ensure that our system is scalable to have more customers in the future, we must check if our system is able to handle high data loading and other stressful operations without problem or we need to limit the number of daily orders to avoid problems. Installing antivirus software to protect our system from attacks, having a regular backup schedule to use it when something wrong goes with the system, are also examples of preventative maintenance that we can apply simply for our system.

Appendix Perform research on the Internet to learn more about Total Cost of Ownership (TCO) and develop a TCO checklist that includes the five most important elements of TCO. Relate your answers to your Assignment 2.

(Please put this analysis as appendices in Assignment 2 report). Total cost of ownership (TCO) generally includes ongoing support and maintenance costs, as well as acquisition costs. TCO totally formed from two different types of cost, which are Direct and indirect costs. Direct costs can be measured in dollars.

These costs result from a decrease in expenses, an increase in revenues or both. If we want to list the most important elements of direct costs for an information system, the list would be as following: 1. Hardware and software (capital and lease expenses) In our project, because we do it in-house and we didn't pay for hardware or software, this part didn't add any cost to our TCO. 2. Support (maintenance, training, travel etc) In this project, our group members have done all SDLC phases based on their knowledge gain from SAD course and have used their background knowledge for programming parts.

Therefore, we didn't pay for training. But, there are some costs for transportation and traveling to and from Taktaz restaurant which is placed in Serdang. 3. Development Because we did the whole project in-house with our group members, we didn't pay any extra fees for labors or out-sourcing. 4. Communication fees The biggest part of our project costs, was communication fees which we paid to contact our group members in each phase of SDLC. Indirect costs are difficult to measure in dollars but are important and need to consider as a part of TCO.

The most important indirect costs in our project are indicating in below. 1. User time User time cost is the cost of users supporting themselves and each

other instead of out-sourcing costs. Costs for user formal training, casual learning, and self-studies are included in user time cost. We needed to learn some issues mostly related to programming part by ourselves during the project. 2. Opportunity cost The highest valued alternative that we need to forego when we are doing our project is part of opportunity cost or downtime cost . lost wages can be a measurement for this kind of costs.