UNIVERSITI PUTRA MALAYSIA

REGISTRATION SYSTEM FOR THE FACULTY OF SCIENCE - SANA'A UNIVERSITY

ALI MOHAMMED AL-SHARAFI

FSKTM 2001 21

REGISTRATION SYSTEM FOR THE FACULTY OF SCIENCE – SANA'A UNIVERSITY

BY:

ALI MOHAMMED AL-SHARAFI

A PROJECT SUBMITTED IN PARTIAL FULFILLMENT OF REQUIREMENTSFOR THE DEGREE OF MASTER OF SCIENCE IN THE FACULTY OF COMPUTER SCIENCE AND INFORMATION TECHNOLOGY UNIVERSITY PUTRA MALAYSIA

NOVEMBER 2001

ABSTRACT

Abstract of a project submitted to the Faculty of Computer Science and Information Technology in partial fulfillment of requirement for the degree of Master of Science.

REGISTRATION SYSTEM FOR FACULTU OF SCIENCE – SANA'A UNIVERSITY

By

ALI MOHAMMED AL-SHARAFI

November 2001

Chairman: **Pn. Norhayati Mohd. Ali**

Faculty: **Computer Science & Information Technology**

The purpose of this project is to develop a computerized registration system for Faculty of Science in Sana'a University – Yemen, since the registration system in the faculty still manual.

This system was developed using Oracle as a DBMS, because of the facilities it provides such as, unlimited database and high-speed data access.

The waterfall model was the model that has been chosen to develop this system because it does not require a user feedback like prototyping model or some other models and this is because this project has been developed in a different place (i.e. not in the same country of the Sana'a University).

With this system, the registration process can be easily done accurately, faster, and with more security than the manual process.

ACKNOWLEDGEMENT

First of all, ALHAMDULELLAH.

I would like to express my special appreciation and dedication to my father who has encouraged me to pursue my studies to get master degree, my mother, my wife, my kids, my brothers and sisters, and to all my family.

Special thanks and appreciation to my supervisor, Pn. Norhayati Mohd. Ali for her guidance, advices, help and understanding, and to Dr. Hamidah Ibrahim, graduate program coordinator for her support and help.

I'd like also to thank all my friends who have in one way or another given me support, advice, or guidance during the preparation of this project, especially my dear friend Belal Zaqaibeh.

DECLARATION

I hereby declare that the project is based on my original work except for quotations and citations, which have been duly acknowledged. I also declare that it has not been previously or concurrently submitted for any other degree at UPM or other institutions.

_____

Ali Al-Sharafi

Date:

Table of Contents

LIST OF TABLES

LIST OF FIGURES

Chapter 1
Introduction

1.1 Overview

The registration process is the first thing the student must go through after getting the admission from the university he wants to study in, and this process must be repeated by the student twice or thrice each year –depending on the system of the university, as some universities have only two semesters per year (first and second) and other universities have three semesters (first, second, and summer semester).

The computer has become main part in our life, and as it is observed, it is in all fields around us which is facilitates our work. So, the need of a computerized registration system is very important to make this process easier for the students and the registrars.

1.2 Problem statement

The registration system in most universities all over the world has become computerized but in Sana'a University – Sana'a – Yemen the registration system is still manual until this moment. The main purpose of computerized systems that they always built to eliminate problems in manual systems or to improve the existing systems. The idea of developing this computerized registration system has occurred because there are many problems in this manual process (registration process) such as:

1. The security in this manual system is not provided completely.
2. Many errors and mistakes occur in this process.

3. The registration process takes very long time to finish and to get the complete students list in each subject.
4. Sometimes many registration forms got lost and this makes a big problem for the students because they have to go through this process again.
5. The difficulty in storing and restoring the information and data.

1.3 Objectives

The objective here is to develop a computerized system to make the registration process easier, faster, and even much more safe. The goal is to make this system as comprehensive as possible to cover every detail in this process. The objectives for developing such system are:

1. To eliminate difficulty in storing and restoring the date and information.
2. To provide security and accuracy in registration process.
3. To provide a collection of reports and queries that are required by users and they were not available in the manual system.
4. To save time by making this process faster and easier than the manual process.

1.4 Scope

The scope of this study –the registration system- is Faculty of Science in Sana'a University at this time, and it can be modified to be used in all other faculties in the same university or to be used in all other universities in Yemen, either those belong to the government or private universities because all of them

have the same system of study. It also can be modified to be used in the schools and institutes, which they are using the similar registration system – or have similar study system.

1.5 The registration process

The registration process in the Faculty of Science in Sana'a University is not different from other registration process in other universities except the system of the study that may make a little difference in the process steps.

If the student is new then he must fill in a personal information form includes his high school information to be used as history info. After that he must fill in the form that indicates his level in the university since the study system in the university is by semester not credits, which means he must study the courses arranged for his semester. If the student has failed in more than 3 courses then he must register that year again otherwise he can continue to the next semester and study those courses in any semester in the next year.

Chapter 2

Literature Review

2.1 Introduction

The use of computers is common in many businesses. Most offices use them for word-processing, and many for spreadsheets and databases. In addition, there are a wide number of more specialized software packages available for applications such as accounts, stock control, design and Registration systems. Most computer systems are database systems, especially those not for business use such as systems for clinical records, systems for libraries, and systems for registration either in universities, schools, or in any other organization.

2.2 Why Computerize?

It is important to know exactly why you plan to computerize. What do you expect the system to do? Is it meant to solve a specific business problem? Are you attempting to keep up with, or ahead of, your competition? Is computerization the best option for your firm? Some possible advantages are:

(i) Increased efficiency

(ii) Cost savings

(iii) Easy retrieval of information

(iv) Improved accuracy

(v) Better services for users

2.3 What is a system?

A system is a collection of elements or components that are organized for a common purpose. The word sometimes describes the organization or plans itself (and is similar in meaning to *method*, as in "I have my own little system") and sometimes describes the parts in the system (as in "computer system").

A computer system consists of hardware components that have been carefully chosen so that they work well together and software components or programs that run in the computer.

The main software component is itself an operating system that manages and provides services to other programs that can be run in the computer. A filing system is a group of files organized with a plan (for example, alphabetical by customer).

All of nature and the universe can be said to be a system. We've coined a word, *ecosystem*, for the systems on Earth that affect life systems. The term can be very useful because so many things can be described as systems. It can also be very unuseful when a more specific term is needed.

2.4 What is a Database?

A database management system, or DBMS, gives the user access to their data and helps them transform the data into information. Such database management systems include dBase, Paradox, IMS, and Oracle. These systems allow users to create, update, and extract information from their databases. Compared to a manual filing system, the biggest advantages to a computerized database system are speed, accuracy, and accessibility.

A database is a structured collection of data. Data refers to the characteristics of people, things, and events.

2.4.1 What is a Relational Database?

A relational database stores all its data inside tables, and nothing more. All operations on data are done on the tables themselves or produce another tables as the result. You never see anything except for tables.

A table is a set of rows and columns. This is very important, because a set does not have any predefined sort order for its elements. Each row is a set of columns with only one value for each. All rows from the same table have the same set of columns, although some columns may have NULL values, i.e. the values for that rows was not initialized. Note that a NULL value for a string column is different from an empty string. You should think about a NULL value as an "unknown" value.

The rows from a relational table are analogous to a record, and the columns to a field. There are two basic operations you can perform on a relational table. The first one is retrieving a subset of its columns. The second is retrieving a subset of its rows. You can also perform operations between two tables, treating then as sets: you can make Cartesian product of the tables, you can get the intersection between two tables, you can add one table to another and so on.

2.4.2 Relational Databases versus Database Servers

Not all databases are relational, and not all relational databases are built on the client/server paradigm. But most of the time the user will want a relational database server, so it's important to clarify the distinction.

A relational database manipulates only tables and the result of all operations are also tables. The tables are sets, which are themselves sets of rows and columns. The database itself can be viewed as a set of tables. So a DBF file is not a relational database. We do not manipulate a DBF table as a set (we are always following an index) and we do not perform operation on tables that yield other tables as the result (we are just looping through records from one or more tables, even when we use the "SET RELATION" dBase statement). Most database file formats are not relational databases. Even the BTrieve server NLM is *not* a relational database, because we do not operate on sets tables or sets of tables.

Conversely, a MDB file (from MS Access) is a relational database. Although we can open and manipulate a MDB file just like a DBF file, navigating through records and index, we can also perform all operations through a relational view of the database and using SQL statements.

Actually, most non-relational databases are based on some "navigational" model: a hierarchy, a linked list, a B-Tree, etc. It's common to refer to these as ISAM (Indexed Sequential Access Method) Databases.

Now let's see what is a database server: it's a specialized process that manages the database itself. The applications are clients to the database server and

they never manipulate the database directly, but only make requests for the server to perform these operations.

This allows the server to add many sophisticated features, such as transaction processing, recovery, backup, access control and etc without increasing the complexity of every application. The server also reduces the risk of data file corruption, if only because only the server writes to the database (a crash on any client machine will not leave unflushed buffers).

A nice database server also takes advantage of the client/server architecture to lower network usage. If we open a DBF or MDB file stored on a file server we need to retrieve every record just to filter out which ones we really need. But if we connect to a database server, it filters out the unneeded records and sends to the client only the data that really matters.

Access is a relational database but it is not a database server. mSQL, SQL Anywhere, DB2, Oracle are both relational databases and database servers. The Btrieve NLM is a database server but it is not a relational database.

2.5 Oracle and Database

Oracle stores each data item in its own *field*. For example, a person's first name, date of birth, and their postal code are each stored in separate fields. The name of a field usually reflects its contents. A postal code field might be named POSTAL-CODE or PSTL_CD. Each DBMS has its own rules for naming the data fields.

A field has little meaning unless it is seen within the context of other fields. The postal code T6G 2H1, for example, expresses nothing by itself. To what person or business does it belong? The postal code field is informative only after it is associated with other data. In Oracle, the fields relating to a particular person, thing, or event are bundled together to form a single, complete unit of data, called a record (it can also be referred to as a row or an occurrence). Each record is made up of a number of fields. No two fields in a record can have the same field name.

During an Oracle database design project, the analysis of the business needs identifies all the fields or attributes of interest. If the business needs change over time, we define any additional fields or change the definition of existing fields.

2.5.1Oracle Tables

Oracle stores records relating to each other in a *table*. For example, all the records for employees of a company would be stored by Oracle in one table, the employee table. A table is easily visualized as a tabular arrangement of data, not unlike a spreadsheet, consisting of vertical columns and horizontal rows.

A table consists of a number of records. The field names of each record in the table are the same, although the field values may differ. Every employee record has a salary field, called SALARY. The values in the SALARY field can be different for each employee.

Each field occupies one column and each record occupies one row. In each column of the table, we put a specific category of information for the employees,

such as their ID number, first name, and position. Each row in the table contains the information relating to a specific employee, together as one record. Each record is a unique entry and is independent of any other record in the table. The EMPLOYEE table, for example, contains records for Barney Rubble and Shelly Gravel. Although both records are part of the EMPLOYEE table, the data contained within them is independent of each other. There is no relationship between Barney's and Shelly's salaries.

After the analysis of the business requirements, the database design team defines the necessary tables. Different tables are created for the various groups of information. An EMPLOYEE table is created for employee information, a DEPARTMENT table is created for department information. Related tables are grouped together to form a *database*. For example, a personnel or human resources application database includes both the EMPLOYEE and DEPARTMENT tables and all other tables involved in the application.

2.5.2Personal Oracle7 Security: Database Passwords

The Personal Oracle7 database has many levels of security to keep stored data secure, but it does not have the same level of security as data stored in an Oracle7 Server running on server-level operating systems such as Windows NT.

The Personal Oracle7 starter database has an assigned password. This is known as password protection. When we first install Personal Oracle7 and the starter database, we do not need to provide passwords. The Personal Oracle7 starter database has been configured so that the password is provided for us. To prevent others from accessing the database, however, we should change the

password assigned to the starter database. This will prevent others from starting up or shutting down the database. We can also change the passwords on user accounts so that only we know the password.

2.6 Irbid National University Registration System

The registration system in the Faculty of Science - Irbid National University is quite similar to the system in the Faculty of Science – Sana'a University except that the student in Irbid National University can choose the courses that he wants to study in each semester.

The process of registering the students is starting by giving the student a form to fill in with the personal information. The next step is that the student fill in a courses form with the courses he would like to study in the semester of registering then submit it to the registrar. In case of closing one of courses (i.e. it reached the maximum number of student allowed to take it) then he must change the section or the course if there is only one section for that course.

If the student has failed in any course he can register it any other semester without need to register that semester again even if he has failed in all courses he registered.

Chapter 3

Methodology

3.1 Introduction

There are many models (methodologies) of software development (life cycle). Most of them will be mentioned briefly and the focus will be on the two most widely used which they are:

- Waterfall model.
- Prototyping model.

3.2 System Development Methodologies

3.2.1 Waterfall Model

In this model one development stage should be completed before the next stage begins. To explain more let's consider that we are in the requirements analysis stage, in this stage we have to be sure that all requirements are elicited from the user, analyzed for completeness and consistency, and documented in a requirements document then only after finishing all this we can go on to next stage which is system design.

There are usually five stages in this model of software development:

- *Requirements analysis and definition.* In this stage the requirements of the "to be developed software" are established. These are usually the services it will provide, its constraints and the goals of the software. Once these are established they have to be defined in such a way that they are usable in the next stage. This stage is often preluded by a feasibility study or a

feasibility study is included in this stage. The feasibility study includes questions like: should we develop the software, what are the alternatives? It could be called the conception of a software product and might be seen as the very beginning of the life cycle.

- *System and software design.* In this stage the established requirements, flowing from the first stage, are identified as software or hardware requirements. The software requirements are then translated in such a way that they can be readily transformed into computer programs.
- *Implementation and unit testing.* This is the stage where the computer programs are created. Each program is called a unit, and unit testing is the verification that every unit meets its specification.
- *System testing.* All the units are combined and now the whole is tested. When the combined programs are successfully tested the software product is finished.
- *Operation and maintenance.* Most software products include this stage of the development. It involves correcting errors that have gone undetected before, improvement and other forms of support. This stage is part of the life cycle of a software product, and not of the strict development, although improvements and fixes can still be considered as "development".

Next figure illustrates the sages of the waterfall model:

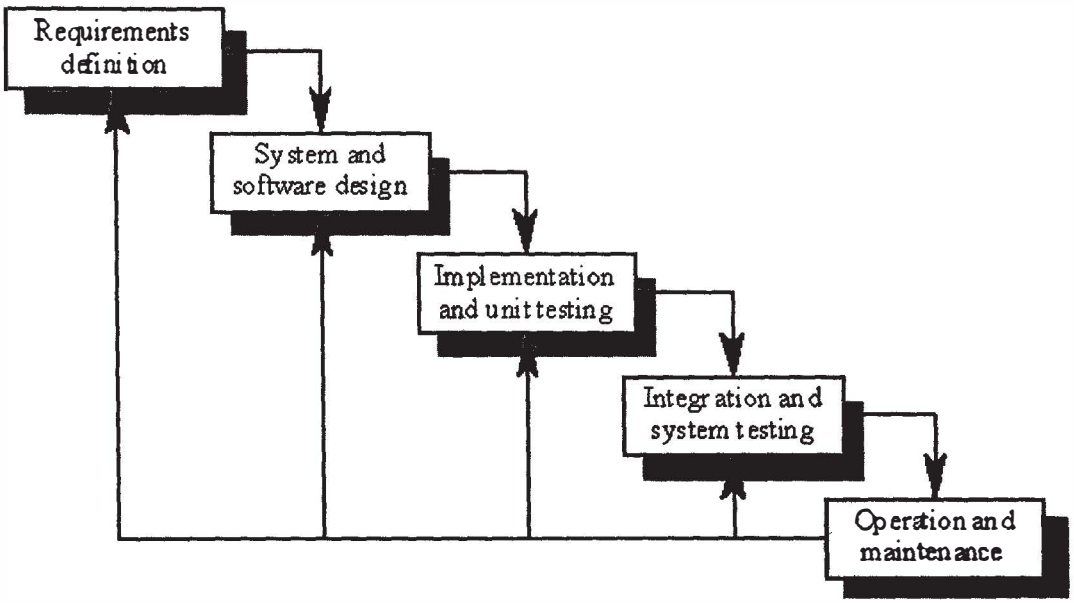

Figure 3.1: Waterfall Model

The waterfall mode can be very useful because of its simplicity that makes it easy to users who are not familiar with software development.

3.2.2 Prototyping Model

This approach is considered as the best offer when the client / user defines a set of general objectives for the software he needs but does not give the detailed input, processing, or output requirements. Some times the developer is unsure of the efficiency of an algorithm, the adaptability of the O.S, or the interaction form between user and machine.

After requirements gathering and the overall objectives have been defined and requirements whish are known have been identified, a quick design occurs - it is a representation of those aspects of the software that will be visible to the client

user. The quick design is considered as a prototype. The prototype then is evaluated by the user who will give his refinement for the requirements for the software to be developed. This iteration continues (i.e. tune prototype) until the client / user is satisfied.

Prototyping steps can be ordered as follows:

1. Requirements gathering:
    - Obtain requirements from client / user.
    - Identify areas of uncertainty.
2. Quick design and build:
    - Quick design of aspects of system visible to user.
    - Quick development of prototype implementation.
3. User evaluation of prototype:
    - Prototype used to refine the requirements of system.

As we mentioned before these steps are repeated until the requirements of system are well defined.

Next figure shows the prototyping model:

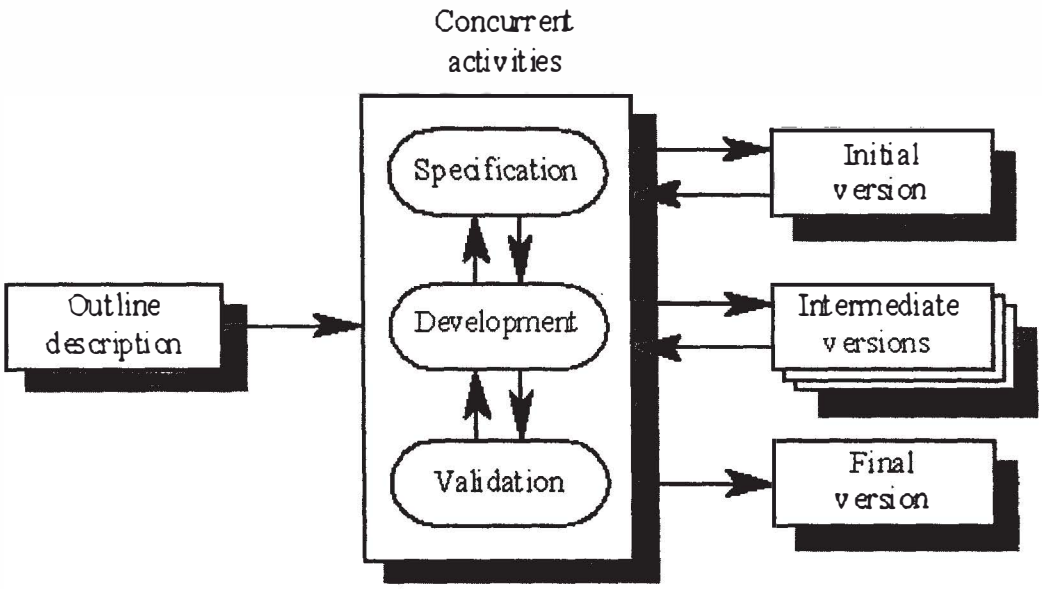

Figure 3.2: Prototyping Model

3.3 Conclusion

Since this project is not going to be developed in the same country that the university is in now (i.e. not in the same place with the client / user) so, the prototype model (methodology) can not be used to develop this system (Registration system), because this model requires feedback from the user.

The best model is waterfall methodology to build this system because it is linear and sequential way and it does not need user feedback. This model has been chosen because it is one of the most widely used methodologies in system building (development) and the other models have some weaknesses that make them not as good as this methodology or because they can not be used unless they are in the same area that user in.