DOKUZ EYLÜL UNIVERSITY
GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

MEDICAL DATABASE DESIGN

by

Mert AYTİMUR

January, 2006
İZMİR

MEDICAL DATABASE DESIGN

A Thesis Submitted to the
Graduate School of Natural and Applied Sciences of Dokuz Eylül University
In Partial Fulfillment of the Requirements for the Degree of Master of Science
in Electrical & Electronics Engineering

by
Mert AYTİMUR

January, 2006
İZMİR

M.Sc THESIS EXAMINATION RESULT FORM

We have read the thesis entitled **"MEDICAL DATABASE DESIGN"** completed by **MERT AYTİMUR** under supervision of **ASST. PROF. DR. MEHMET KUNTALP** and we certify that in our opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

....

Asst. Prof.Dr.Mehmet KUNTALP

....

Asst. Prof.Dr.Damla Gürkan KUNTALP Asst. Prof.Dr.Adil ALPKOÇAK

Prof.Dr. Cahit HELVACI
Director
Graduate School of Natural and Applied Sciences

ACKNOWLEDGMENTS

First I want to express my gratitude to my friends Mirkan ALTAY, Mesture KAYHAN, and Hürriyet KESKIN for their endless patience and for their moral support they give me with their friendship. I also want to show my gratitude to my family for their endless support.

Of course, I want to thank to my adviser Mehmet KUNTALP for his patience, his understanding, his helpful personality, and for his support during my work.

Mert AYTİMUR

MEDICAL DATABASE DESIGN

ABSTRACT

This work covers a Hospital Information System application developed for patient information monitoring, better patient treatment and more efficient medical record purposes. The application in question uses the benefits of the client-server architecture and the benefits of the world-wide-web.

First a brief explanation of Medical Informatics and Hospital Information Systems are given to give one a general understanding what the application deals with. The main architecture of such systems and the expected role in the future of applications in that category are presented to the reader.

As getting more into detail the used development platform, and other tools are given to make the reader aware of how and by which means this application was developed. Also the information of how the design for the storage side should be made is given in more details as a whole chapter. The modeling steps for the database of the application are explained.

The whole application covering the pages (each page and its sub pages are separately explained), the database model and how it works (containing the tables representing the entities and relations between them), the 'imaged history' file system model (folder tree and all its sub folders with files containing) are investigated and one by one explained in the next chapter.

Keywords : Hospital Information Systems, Entity-Relationship Model, Client-server Architecture, Medical Informatics Applications, Database Design.

TIBBİ VERİTABANI TASARIMI

ÖZ

Bu çalışma, hasta takibi, daha iyi tedavi süreçleri yaratma ve tıbbi data kayıtlarının daha verimli kayıt altında tutulması amaçlarıyla hazırlanmış bir ‘hastane bilgi sistemi’ uygulamasını anlatmaktadır. Söz konusu uygulama sunucu-istemci yapısı ve dünya çapında ağ teknolojisinin nimetlerinden faydalanılarak hazırlanmıştır.

İlk bölümde uygulamanın ne tür bir alanda hizmetler verebileceğini vurgulamak için tıp bilişimi ve hastane bilgi sistemleri hakkında genel bir bakışa yer verilmiştir. Benzer sistemlerin genel yapısı ve bu sistemlerin gelecekteki yerlerinden bahsedilmiştir.

Uygulamanın geliştirildiği platform ve kullanılan araçlar bir sonraki bölümde ele alınmış, ayrıntılı bilgiler verilmiştir. Uygulamanın bilgi depolama kısmının nasıl tasarlanması gerektiği ayrı bir bölüm olarak okuyuculara sunulmuştur. Bu bölüm bilgi depolama modellemesi ve adımları ile ilgili bilgiler içermektedir.

Bir sonraki bölümde ise uygulama anlatılmıştır. Uygulamadaki tüm sayfaların ve alt sayfaların işlevleri ile birlikte ayrıntılı açıklamalarına yer verilmiştir. Ayrıca uygulamada kullanılan veritabanı modeli, veritabanındaki girdi ve ilişkileri temsil eden veritabanı tabloları gösterilmiştir. Son olarak da resimli kayıtların depolanacağı dosyalama sistemi, alt klasörler ve içerdikleri dosyalarla birlikte ayrıntısıyla anlatılmaya çalışılmıştır.

Anahtar sözcükler : Hastane Bilgi Sistemi, İlişkisel Veritabanı Modeli, Sunucu-istemci Yapısı, Tıp Bilişimi Uygulamaları, Veritabanı Tasarımı.

CONTENTS

Page

CHAPTER ONE
INTRODUCTION

The most important issue of the human being is the wellness of the health and how it can be improved. For the last few decades, many researches are deeply focused on this issue. Scientists and physicians have reached a number of results and are still busy on it to put better ones in service. “Medical Informatics” is one of those research areas. Recently, it was just a dream to transfer huge amount of patient data from one place to another. The development of the World Wide Web lit a light for the future, but the transfer rates were still not enough. After the third generation of wireless communication (3G), and the high-speed data networks like ATM and SONET became useable worldwide, this dream became true.

1. Medical Informatics and Hospital Information Systems

Medical Informatics is the name given to the application of information technology to healthcare. It is the:

"Understanding, skills and tools that enable the sharing and use of information to deliver healthcare and promote health" (British Medical Informatics Society).

Medical informatics is often called healthcare informatics or biomedical informatics, and forms part of the domain of eHealth. These later-generation terms reflect the substantive contribution of the citizen & non-medical professions to the generation and usage of healthcare data and related information. Additionally, medical informaticians are active in bioinformatics and other fields not strictly defined as health care.

The aspects of medical informatics include:

- Architectures for electronic medical records and other health information systems used for billing, scheduling or research.
- Decision support systems in healthcare

- Messaging standards for the exchange of information between health care information systems (e.g. through the use of the HL7 data exchange standard) - these specifically define the means to exchange data, not the content
- Controlled medical vocabularies such as the Standardized Nomenclature of Medicine, Clinical Terms (SNOMED-CT) or Logical Observation Identifiers Names and Codes (LOINC) - used to allow a standard, accurate exchange of data content between systems and providers.
- Use of hand-held or portable devices to assist providers with data entry/retrieval or medical decision-making

Medical informatics began in the 1950s with the rise of useable computation devices, computers. Early names for medical informatics included medical computing, medical computer science, computer medicine, medical electronic data processing, medical automatic data processing, medical information processing, medical information science, medical software engineering and medical computer technology.

The earliest use of computation for medicine was for dental projects in the 1950's at the National Bureau of Standards by Robert Ledley.

The next steps in the mid 1950s were the development of expert systems such as MYCIN and INTERNEST-I. In 1965, the National Library of Medicine started to use MEDLINE and MEDLARS. At this time, Neil Pappalardo, Curtis Marble, and Robert Greenes developed MUMPS (Massachusetts General Hospital Utility Multi-Programming System) in Octo Barnett's Laboratory of Computer Science at Massachusetts General Hospital in Boston. In the 1970s and 1980s it was the most commonly used programming language for clinical applications. The MUMPS operating system was used to support MUMPS language specifications. As of 2004, a descendent of this system is being used in the United States Veterans Affairs hospital system.

1970-1980: Large-scale clinical information systems, such as hospital information systems (e.g., HELP at LDS Hospital in Utah) and electronic medical record systems (e.g., PROMIS at the University of Vermont) began to appear at pioneering academic institutions, followed later by commercial products with limited capabilities. Several prototypic demonstration systems employ symbolic artificial intelligence methods to support diagnostic and therapeutic decision support. DENDRAL at Stanford is used to help deduce biochemical structures from mass spectrometry data.

In the United States in 1996, HIPAA regulations concerning privacy and medical record transmission created the impetus for large numbers of physicians to move towards using EMR software, primarily for the purpose of secure medical billing.

In the US, progress towards a standardized health information infrastructure is underway. In 2004, the US Department of Health and Human Services (HHS) formed the Office of the National Coordinator for Health Information Technology (ONCHIT), headed by David J. Brailer, M.D., Ph.D. The mission of this office is to achieve widespread adoption of interoperable electronic health records (EHRs) in the US within 10 years.

A **hospital information system** (HIS) is a comprehensive information system dealing with all aspects of information processing in a hospital. This encompasses human (and paper-based) information processing as well as data processing machines.

As an area of Medical Informatics the aim of an HIS is to achieve the best possible support of patient care and administration by electronic data processing. It can be composed of few software components with specialty specific extensions as well as of a large variety of sub-systems in medical specialties (e.g. Laboratory Information System, Radiology Information System).

1.1 HIS Architectures

Hospital Information Systems (HIS): A Hospital Information System is a combination of computer hardware and software with the necessary communications that aid the management of a hospital's clinical and administrative information. They allow end-users access to patient records, laboratory results, staffing and scheduling, supplies and inventory, amongst other things.

1.1.1 General Architecture of Hospital Information Systems

Wiederhold and Perreault (1990) have mentioned, that the purpose of HIS is to manage the information that health professionals need to perform their jobs effectively and efficiently. Every hospital employee however needs more than only patient related information - thus a HIS is more than an electronic medical record system. To illustrate, why a HIS is more than just the sum of all departmental systems, integrated into a hospital communication system, we shall emphasize the difference between data and information (Prokosch & Dudeck., 1995).

The large amounts of data collected in different hospital areas, are of no value for the health care staff, if no tool is available to transform these data into exactly that piece of information, which is needed for every single decision. Thus, we may distinguish between two levels of qualitative enhancements characterizing a HIS:

- The first level comprises information processing capabilities, which apply data retrieval, data aggregation and data transformation on all data arising in the hospital, independently of its original source. Such capabilities provide large benefits for a variety of management decisions (clinical and administrative ones, as well as short term and strategic long term decisions) and for medical research.
- The second level comprises knowledge processing capabilities, which may be applied in order to support the failing memory of human beings. This means that decision support or decision monitoring functions can actively analyze the data captured during the daily communication and documentation

processes in order to provide proposals in certain data entry situations or raise alerts if they detect critical situations for any patient (Prokosch & Dudeck., 1995).

In my understanding,

a hospital communication system that further includes some level of the above described information processing or knowledge processing functionality shall be named hospital information system.

1.1.2 Advantages of Hospital Information Systems

There are several benefits of HIS compared with efforts and costs.

- *Accessibility:* Computer-stored data can be viewed at multiple locations at all times. When required permissions and proper infrastructure are present, co-treating clinicians can directly view data in each other's patient records. (Chae, p.14)
- *Readability:* Scanned documents can be made available at multiple locations, but freehand may be difficult to read. Typed information, often acquired through transcription, is easy to read, but susceptible to errors. Several authors report on studies where the elimination of transcription not only improved reliability, but also saved time and money. (Chae, p.14)
- *Reporting:* Data in well-organized CPR scan be used to generate reports for institutional, regional or national repositories, and reduces the need for redundant recording. Correspondence, such as progress, referral, and discharge letters can also be generated with data in the CPR. (Chae, p.14)
- *Completeness:* Computers can actively prompt for data. This is useful for improvement of the quality of data in CPRs, especially in the context of decision support, data analysis, and reporting. (Chae, p.14)
- *Decision Support:* This is a broad area of functions that support diagnosis making and treatment policy, which often involve both assessment of health parameters and treatment. (Chae, p.15)
- *Access to external knowledge sources:* Searches of databases with reference knowledge can be performed on the basis of CPR contents. Examples are literature

databases, such as Medline, but also other healthcare related sites via the Internet, such as HON (Health on the Net), Cliniweb, and Medweb. (Chae, p.15)

- *Admissions:* While the initial registration can be lengthy, subsequent visits are quick to register and demographic data is easily updated.

- *Patients:* There is less repetition of personal information. Computerized patient records allow relevant information to be available where and when it is needed.

- *Laboratory:* Specimens are labeled at their source using bar codes and all processing uses the bar codes. This improves specimen tracking, patient safety and utilization of laboratory resources.

- *Laboratory:* Standardization among all provincial laboratories has made ordering more consistent, comparisons easier and more meaningful.

- *Pharmacy:* Automated drug interaction checking prevents errors and results in better patient care.

- *Nursing:* The use of bar codes for patient identification has resulted in fewer errors, not just transcription errors, with automated data capture.

- *Physicians:* The integrated view of patient data, with quick and easy access to up-to-date patient information results in more timely diagnosis and treatment.

- *Diagnostic Imaging:* Report distribution is greatly improved as the nursing units get their reports immediately after they have been signed.

- *Management:* Reporting can be used to make evidence based decisions and maximize the utilization of resources.

- *Management:* Measures of quality of care and other performance measures are available to help identify opportunities for process improvement and ensure compliance with accreditation standards.

1.2 Existing HIS Applications

Recently, as the latest generation of hospital information systems there are many applications adopted. The developer or development groups to be formed took some architectures and models into to account to build their target applications, and to take advantage of various design and development infrastructures.

During this work some of these applications were considered and given special attention. As examples of these recent hospital information systems: the Son Llatzer HIS, the Birim HIS, and the Care2X HIS may be given. Even though they are all hospital information systems and they all have things in common, they differ in many ways. Each one has superiorities regarding the other one. Nevertheless they all have the common goal: To serve for the wellness of patients.

Hospital Son Llatzer (HSLL) in Palma de Mallorca has nearly 500 beds and belongs to the public health service network in Spain. Its HIS was created with the idea of a centralized unique hospital portal. All data and images (digital radiology, endoscopic images, digital ECGs, patient treatment information etc.) are stored in a unique electronic patient record (EPR). This system is specialized to the hospital and is designed to satisfy the needs like laboratory and radiology requests a.o. appearing within the borders of the hospital. To the systems are also wireless access integrated to fulfill the requests made via laptops and PDA's of health professionals and patients.

The Birim Hospital Information System whereas adopted the modular structure. Birim Bilgi İşlem Müşavirlik ve Ticaret Limited Şirketi, the developer firm of the Birim HIS is a leading company in Hospital Information Systems in Turkey, founded in Izmir. The main principle of its HIS structure is the collection of modules, as stated before. They have developed modules for each department a hospital needs, and have brought them together as an entire system, unlike the Son Llatzer Hospital has done. The system is reachable from everywhere within de hospital, but it is said that they are ready to be reachable from outside if needed. A strong security policy is being applied, to prevent data loss and unauthorized access. The database access layer consists of Oracle database systems. They have a 2 and 3-layered design, and they use the 4th generation Oracle, Borland and third party firms' development platforms for their hospital information system. Their software solutions have a strong relationship with the Microsoft Active Directory structure.

On the other hand the Care2X HIS also adopts the modular structure. They claim that they have the ability to integrate any type of application within the Caer2X HIS, even if they are non-compatible with each other.

“Care2x HIS is an open source development (OSD) project and distributed with a "GNU General Public License". Its source code is freely distributed and available to the general public. It is a web-based software and all its functions can be accessed with a common web browser thus there is no need for special user interface software. All program modules are processed on the server side. Module updates and extensions do not require changes on the browsers thus there are no network interruptions and downtimes. Its design supports multiple server configuration to distribute traffic and improve speed and efficiency.” they say in their web sites. The Care2X HIS has a simple development and database infrastructure. They use MySQL as database and C++ as the coding language. While it is an open source project, it is easy to join and add new features to the developed system.

While the SMYRNA HIS has been developed the hospital information systems mentioned above were considered. Their benefits have been taken into account and their weak sides have been searched to bring a new design to life.

1.3 HIS and Future

In the near future hospitals will have to extend their old fashioned systems on newly developed multi-functional clinical systems. Providers try to convince their customers to investigate for such systems.

On the customer side there is a clear understanding on how valuable such systems are. The bureaucratic barriers to direct the funds to such systems stand in the way. Nevertheless there is a promising future in the HIS market. There is an expected growth in Europe from USD 3,131.0 million (in 2003) to USD 6,343.0 million in 2010. The pioneer countries will be the UK, Germany, and France.

Increasing evidence of the cost of medical errors and inefficiencies in healthcare delivery is also expected to contribute to market growth. Concerns about patient safety as well as data security are on the rise, causing providers to seek information systems that are able to reduce errors and eliminate inefficiencies.

In detail, the clinical part of the HIS market will be expected to grow the most. This is because the healthcare industry is increasingly banking on clinical decision support systems that will reduce their costs. The estimation in clinical area is to grow at a compound annual growth rate of 12.3 per cent between 2003 and 2010.

Improving healthcare revenues through automized hospital information systems, and the advantages on patient health of such systems are the driving factors of the market.

CHAPTER TWO
MODELING AND DEVELOPMENT INFRASTRUCTURE

2.1 .NET Framework

The .NET framework created by Microsoft is a software development platform focused on rapid application development, platform independence and network transparency. .NET is Microsoft's strategic initiative for server and desktop development for the next decade. According to Microsoft, .NET includes many technologies that are designed to facilitate rapid development of Internet and intranet applications.

.NET has brought new functionalities and tools to the API (Application programming interface). These innovations allow programmers to develop applications for Windows and the web as well as components and services (web services). .NET provides a new object oriented API. .NET is designed to be sufficiently generic that many different high-level languages can be compiled.

The .NET framework consists of various technologies, which will be given in detail later in this chapter.

- Common Language Infrastructure (CLI) & Common Language Runtime (CLR)
- .NET Languages
- .NET Class Library
- ASP.NET
- Visual Studio.NET

2.1.1 Common Lang. Infrastructure (CLI) & Common Lang. Runtime (CLR)

The common language infrastructure (CLI) is used in Microsoft .NET as the basis for running programs written in different programming languages. Before this, each language required its own runtime module for the compiled programs to run - Visual Basic required VBVM; Visual C++ versions prior to Visual C++.NET needed

MSVCRT DLL. The common language infrastructure is an effort to unify the different runtime modules needed.

The CLI uses a class library and virtual machine, the Common Language Runtime (CLR). Many compilers are in development to produce code for this virtual machine. The code the virtual machine uses is expressed in a Common Intermediate Language (CIL), also known as Microsoft Intermediate Language (MSIL), a CPU-independent set of instructions that can be efficiently converted to native code. CIL can be thought of as a high level assembly language.

As stated before, Common Language Runtime (CLR) is the name chosen by Microsoft for the virtual machine plus runtime library underlying their .NET initiative. The CLR is a generalized multi-language, reflective execution engine on which code originally written in various languages runs. As of 2005, over 40 languages were supported. The runtime environment is mainly used on the Microsoft operating systems, but there exist less capable open source implementations for Unix machines.

2.1.2 .NET Languages

The CLI is designed to provide support for any object-oriented programming language, sharing a common object model and a large common class library. .NET supports over 40 programming languages. Many of these compilers are free.

Most languages have significant adjustments to fit into the .NET Framework. The vendors have often used this as an opportunity to change other features of the languages at the same time. The built-in languages are as follows:

- C#, an object-oriented language.
- JScript .NET, a compiled version of Microsoft's JScript.
- J#, a Java and J++ (the Microsoft variant of Java).
- Managed C++, a variant of the C++ programming language for the .NET platform
- Visual Basic .NET, an improved, object-oriented, multi-threaded version of the classic Visual Basic programming language.

Some available third party languages are:

- A#
- APL
- Boo, based on Python
- COBOL
- Component Pascal
- Delphi 8 and Delphi 2005
- Eiffel
- F#, a member of the ML programming language family.
- Forth
- FORTRAN
- IKVM
- Iron Python, Python
- Lisp
- Mercury
- Oberon
- Pearl
- RPG
- Smalltalk

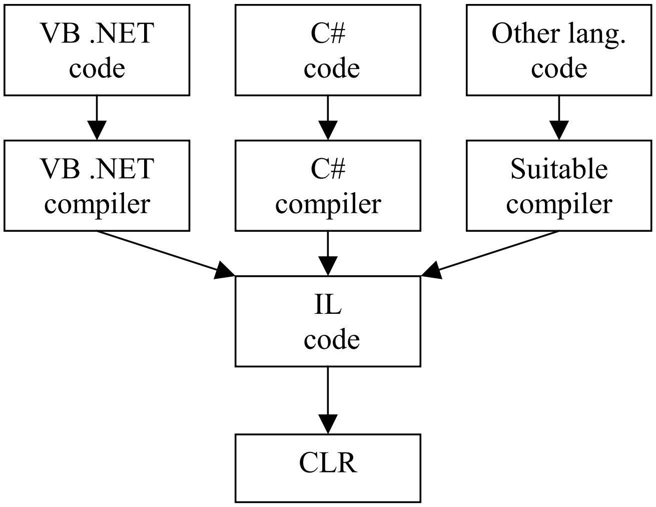


Figure 2.1 Compilation in .NET

2.1.3 .NET Class Library

The .NET Framework class library is a library of classes, interfaces, and value types that are included in the Microsoft .NET Framework SDK. This library provides access to system functionality and is designed to be the foundation on which .NET Framework applications, components, and controls are built.

The class library is object oriented, easy to use, and accelerates software development. For example, the .NET Framework collection classes implement a set of APIs that you can use to develop your own collection classes. The .NET Framework class library enables you to accomplish a range of common programming tasks such as string manipulation, data collection, database connectivity, and file access. One can also use this library to develop console applications, Windows Forms, ASP.NET applications, XML Web services, and Windows services.

2.1.4 ASP.NET

ASP.NET is a set of web development technologies marketed by Microsoft. Programmers can use this set of technologies to build web applications and XML web services.

Even though ASP.NET takes its name from Microsoft's old web development technology, ASP (Active Server Pages), the two differ widely. Microsoft has completely rebuilt ASP.NET, based on the CLR shared by all Microsoft .NET applications. ASP.NET is faster because the entire web site is compiled to one or a few dll files on a Web Server and the Web Site runs faster compared to the previous technology.

ASP.NET attempts to simplify developers' transition from Windows application development to web development by allowing them to build pages composed of controls similar to a Windows user interface. A web control, such as a button or label, operates in the same way as its equivalent in Windows controls. The code can assign its properties and respond to its events. Controls know how to render themselves: whereas Windows controls draw themselves to the screen, web controls produce segments of HTML which form part of the resulting page sent to the end-user's browser.

ASP.NET encourages the programmer to develop applications using an event-driven GUI, rather than in the conventional web scripting style. The framework combines existing technologies such as JavaScript with internal components like "View state" to bring persistent state to the inherently stateless web environment.

ASP.NET uses the .NET Framework as an infrastructure. The .NET Framework offers a managed runtime environment (like Java), providing a virtual machine with JIT and a class library.

The numerous .NET controls, classes and tools can cut down on development time by providing a rich set of features for common programming tasks. Data access provides one example, and comes tightly coupled with ASP.NET. A developer can make a page to display a list of records in a database.

ASP.NET offers several important advantages over previous Web development models:

- *Enhanced Performance:* ASP.NET is compiled common language runtime code running on the server. ASP.NET can take advantage of early binding, just-in-time compilation, native optimization, and caching services. This affects to a remarkable better performance.
- *World-Class Tool Support:* A rich toolbox and designer in the Visual Studio integrated development environment complement the ASP.NET framework. WYSIWYG editing, drag-and-drop server controls, and automatic deployment are just a few of the features this powerful tool provides.
- *Power and Flexibility:* Because ASP.NET is based on the common language runtime, the power and flexibility of that entire platform is available to Web application developers. Also its language independent structure makes it valuable.

Further, common language runtime interoperability guarantees that one's existing applications in COM-based development is preserved when migrating to ASP.NET.

- *Simplicity:* ASP.NET makes it easy to perform common tasks, from simple form submission and client authentication to deployment and site configuration. Additionally, the common language runtime simplifies development, with managed code services such as automatic reference counting and garbage collection.
- *Manageability:* ASP.NET employs a text-based, hierarchical configuration system, which simplifies applying settings to your server environment and Web applications. Because configuration information is stored as plain text, new settings may be applied without the aid of local administration tools. An ASP.NET Framework application is deployed to a server simply by copying the necessary files to the server. No server restart is required, even to deploy or replace running compiled code.
- *Scalability and Availability:* ASP.NET has been designed with scalability in mind, with features specifically tailored to improve performance in clustered and multiprocessor environments. Further, processes are closely monitored and managed by the ASP.NET runtime, so that if one misbehaves (leaks, deadlocks), a new process can be created in its place, which helps keep your application constantly available to handle requests.
- *Security:* With built in Windows authentication and per-application configuration, one can be ensured that developed applications are secure. In more detail: ASP.NET offers a set of strict rules to ensure the security of developed web sites.

2.1.5 Visual Studio.NET

Visual Studio .NET is an Integrated Development Environment (IDE) developed by Microsoft. It is for the Microsoft Windows operating system.

Visual Studio .NET supports the new .NET languages C#, Visual Basic .NET and Managed C++ in addition to C++. You can use Visual Studio .NET to make applications targeting Windows (using Windows Forms), Web (using ASP.NET and Web Services) and portable devices (using .NET Compact Framework).

The look-and-feel of Visual Studio .NET is nearly identical to previous versions of the IDE (Microsoft Visual Studio). Some notable exceptions include the cleaner interface and greater cohesiveness. It is also more customizable with status windows that automatically hide when not in use. All versions of Visual Studio, including its predecessor Visual C++, include a debugger that is integrated with the editing environment.

The most notable feature of the IDE is its support of the new .NET languages. Programs developed in these languages do not compile to machine language (like C++ does, for example) but instead compile into something called CIL. When programs execute the CIL application, it is converted into the appropriate machine language for the platform it is being executed on.

For some features Visual Studio.NET is a very comfortable development platform for developers who like all in one. What Visual Studio.NET offers:

- *Integrated Error Checking:* Visual Studio.NET can recognize problems like casting errors, lacking namespaces, undefined variables etc. during development. While the developer writes the code errors are extracted, underlined and for subsequent debugging listed.
- *Web Form Designer:* One can add easily web controls by dragging and dropping them to web forms. Visual Studio.NET does the background work for the developer: adds the needed scripts into the .aspx file and constructs the suitable background code for the corresponding control.
- *Code Writing Efficiency:* Automatic code completion, colored syntax, and foldable code blocks features makes the development easy, fast and efficient. To make things faster one can also built its own macros.
- *Easy to start:* Starting a web project is very easy. No struggling with one or more configuration files. Visual Studio.NET does all background work for the developer. All classes inside the project will be compiled into a single DLL file.

- *Scalability:* Through this feature the developer has the ability to construct its special controls, dynamic help files etc. Many features can be changed by the desire of the project owner.
- *Integrated Debugging:* The built-in debugging tool makes debugging easy. One can track the code, stop, check variables, trace the stack, and do anything other to make one's work more efficient.

2.2 Database Design, the Entity – Relationship Model

2.2.1 The Entity – Relationship Model

2.2.1.1 Overview of Database Management Systems

The amount of information has become increasingly large and the storage of data for organizations has become very valuable. To get data out of their large and complex datasets, users require tools that simplify the tasks of managing the data and extracting useful information.

A **database** is a collection of data where data can be manipulated, in a desired manner. It stores information distinct to each application. For example, a university database may contain following information:

- Entities such as students, faculty, and courses
- Relationships between entities, such as enrollment in courses, etc.

A **database management system**, or DBMS, is a designed software that helps maintaining and managing the large amount of information. The need for such systems has become very large. Using a DBMS has many advantages:

- Data Independence
- Data Security
- Efficient Data Access
- Data Integrity
- Concurrent Access and Crash Recovery
- Reduced Application Development Time

The user of a database is concerned with real-world structures and the data that is stored and manipulated describes the aspects of these structures.

A **data model** is a collection of high-level data description constructs that hide many low-level storage details. In a DBMS, data is allowed to be stored in a manner that this model has previously described. Most data models are based on relational data models that are the concern of this document. A database design in terms of a semantic model serves as a useful starting point and is translated into a database design in terms of the data model the DBMS supports.

A widely used semantic model called the entity-relationship (ER) model allows us to denote entities and relations, pictorially.

2.2.1.2 Database Design And The E-R Model

The **Entity-relationship model** or **Entity-relationship diagram (ERD)** is a data model or diagram for high-level descriptions of conceptual data models, and it provides a graphical notation for representing such data models in the form of entity-relationship diagrams. Such data models are typically used in the first stage of information-system design; they are used, for example, to describe information needs and/or the type of information that is to be stored in the database during the requirements analysis.

The Entity-Relationship model allows one to describe entities and relations and is used as the basis in many database management systems. It provides useful concepts in extracting data that is needed no more no less.

The database design process can be divided into six individual steps:

1. Requirement Analysis: This is the first step of database design. This is one of the most important steps. This step decides what data to be stored and how the data is used. It must be searched which data must be extracted from the

database that the users will use. This is the step in which discussions must be made with users and groups, and documentations of applications must be prepared.

2. Conceptual Database Design: The next step and the concern of this document. The previously collected data has been converted into a schema. This step is often carried out using the ER model. The goal is to create a simple description of the data. This step must show a simple look of the desired design. Even a user with no technical knowledge must extract the meaning and purpose of the schema.
3. Logical Database Design: A DBMS system must be chosen to perform this step in database design. This step covers the translation of the constructed schema into the DBMS system. Any query language is used to perform this conversion. The most efficient query language is the Structured Query Language (SQL).
4. Schema Refinement: In this step the designed system is looked trough to identify possible problems and to overcome these problems. The database design is refined from these possible problems using normalization and scheme refinement methods.
5. Physical Database Design: This step identifies how and where the data is stored. Any indexing and storage methods are searched to find the most suitable solution.
6. Application and Security Design: Each database has its users. The last step determines the user access and manipulation rights. Roles are maintained for each user to ensure the security and integrity of the database used.

The focus point of this work is the second step: “Conceptual Design”

2.2.1.2.1. Conceptual Design: The Conceptual Design Step is one of the most important steps in database design. Any wrong design in this step will lead to possible failures of the needed application, because this is the semantic step of the design procedure. Some questions must be answered to fulfill this step, such as:

- What are the entities and relationships?

- What information about these entities and relationships should we store in the database?
- What are the integrity constraints?
- Can the model be represented pictorially? (ER diagram)
- Can map an ER diagram into a relational schema

An **entity** represents a discrete object. Entities can be thought of as nouns. Examples: a computer, an employee, a song, a mathematical theorem. A **relationship** captures how two or more entities are related to one another. Relationships can be thought of as verbs. Examples: an owns relation between a company and a computer, a supervises relation between an employee and a department, a performs relation between an artist and a song, a proved relation between a mathematician and a theorem. Entities are drawn as rectangles, relationships as diamonds.

Entities and relationships can both have attributes. Examples: an employee entity might have a social security number attribute; the proved relation may have a date attribute. Attributes are drawn as ovals connected to their owning entity sets by a line. Every entity (unless it is a weak entity) must have a minimal set of uniquely identifying attributes. This set is called the entity's primary key.

Entity-relationship diagrams don't show single entities or single instances of relations. Rather, they show entity sets and relationship sets (displayed as rectangles and diamonds respectively). Example: a particular song is an entity. The collection of all songs in a database is an entity set. The proved relationship between Andrew Wiles and Fermat's last theorem is a single relationship. The set of all such mathematician-theorem relationships in a database is a relationship set.

Lines are drawn between entity sets and the relationship sets they are involved in. If all entities in an entity set must participate in a relation in the relationship set, a thick line is drawn. This is called a participation constraint. If each entity of the entity set can participate in at most one relationship in the relationship set, an arrow is drawn from the entity set to the relationship set. This is called a key constraint. To indicate that each entity in the entity set is involved in exactly one relationship, a thick arrow is drawn.

Associative entity is used to solve the problem of two entities with a many-to-many relationship.

Unary Relationships - a unary relationship is a relationship between the rows of a single table.

2.2.1.2.2. Conceptual Design Using the E-R Model: During the design steps in a conceptual design some considerations must be made, some choices must be made, and some questions must be answered. Such include:

Design Choices

- Should a concept be modeled as an entity or attribute?
- Should a concept be modeled as an entity or relationship?
- Identifying relationships: Binary or ternary relationship? Aggregation?

Note Constraints of the ER Model

- A lot of data semantics can (and should) be captured
- But some constraints cannot be captured. Refinement will be made in the logical design step.

Entity vs. Attribute

One of the design choices is to identify a property as an entity or an attribute. For example, consider the property “address”. One option is to use it as an attribute. Using it as an attribute denotes that the address information is not an important entry. When one wants to know only the address, say for en employee that it is used as an attribute.

The other option is that the address is an important property. When one wants to store more than one addresses, say home address and work address; or the details of an address will be important, say city, street, and then it is used as an entity.

Entity vs. Relationship

Consider the manages relation given in Figure 2.2. We know the manager, its starting date and its budget given. What if the manager manages more than one department, and the budget covers all departments? Then the same information will be stored for each department, which is not true. One can solve the problem by adding another entity set called Manager. The attributes since and dbudget now is describes a manager entity set not a relation. This shows that every manager has a distinct budget and has a distinct budget for each department. Meanwhile it is prevented to store redundant information.

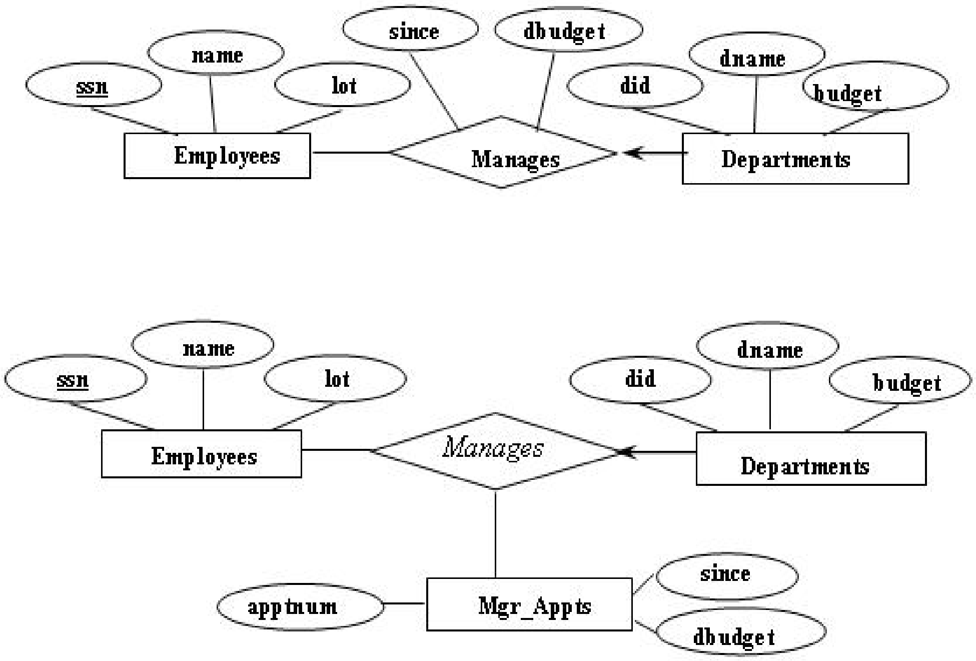


Figure 2.2 Entity vs. Relationship

Binary vs. Ternary Relationship

Consider the ternary relationship in Figure 2.3. It models a situation in which an employee can own several policies. Each policy can be owned by several employees, and each dependant can be covered by several policies.

The requirements:

- A policy cannot be owned jointly by two or more employees.
- Every policy must be owned by some employee.

Dependants are a weak entity set, and each dependent entity is uniquely identified by taking pname in conjunction with policyid of a policy entity.

The first constraints identifies that (which is a side effect) a policy can cover only one dependant. The second identifies we impose a total participation on Policies. It is true if each policy covers at least one dependant. The third requirement forces us to introduce an identifying relationship that is binary. Even ignoring the 3rd one, the best way to model this situation is to use two binary relationships as shown below.

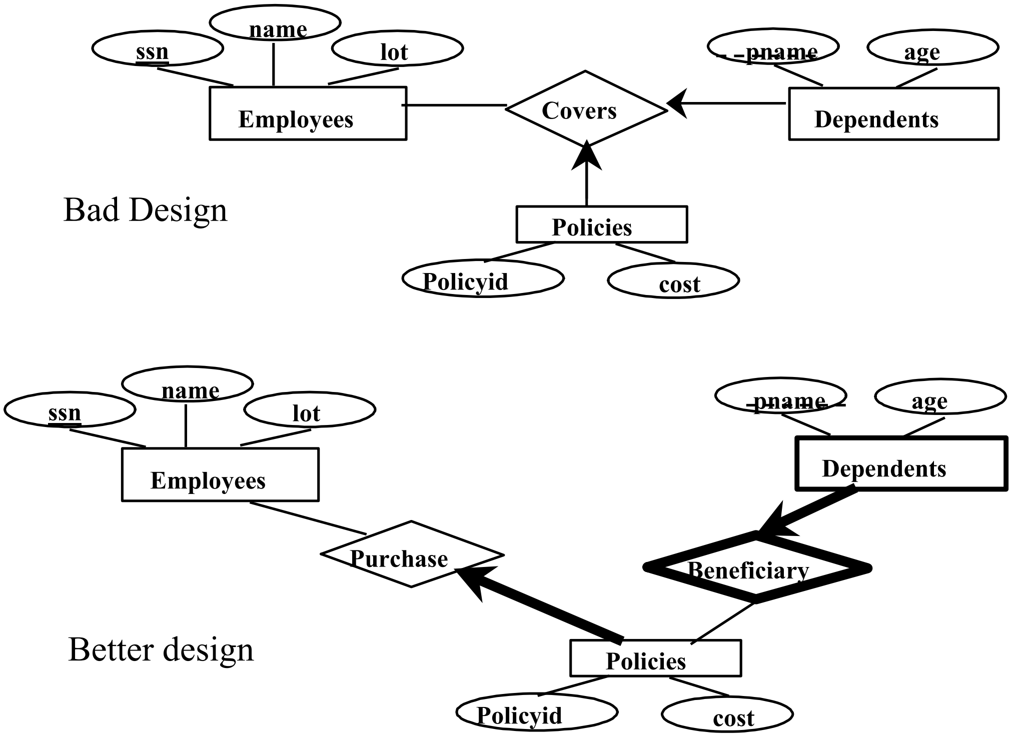


Figure 2.3 Binary vs. Ternary Relationship

2.2.1.2.3. Logical Database Design: The entire schema constructed in the conceptual database design must be converted into tables to ensure the usability of the design. This can be ensured through a query language. Some examples will be given to make give knowledge of the concept by means of the Structured Query Language (SQL).

Entity Sets to Tables

The Entity Set in Figure 2.2. is used to translate it to a database table.

```
CREATE TABLE Employees
        (ssn CHAR(11),
        name CHAR(20),
        lot  INTEGER,
        PRIMARY KEY  (ssn))
```

Table 2.1 An Instance of the Employees Entity Set

	ssn	name	lot
1	300	22	20
2	300	26	20
3	200	13	7

Relations to Tables

The same procedure is applied as in the Entity to Tables section. The main difference is that Foreign Keys must be identified to show the connection (relation) between two entity sets.

```
CREATE TABLE Works_In(
                ssn  CHAR(1),
               did  INTEGER,
                since  DATE,
                PRIMARY KEY (ssn, did),
                FOREIGN KEY (ssn)
                    REFERENCES Employees,
                FOREIGN KEY (did)
                    REFERENCES Departments)
```

Here the foreign key "ssn" refers to the primary key ssn in the "Employees" entity set, and the foreign key "did" refers to the primary key did in the "Departments" entity set.

2.2.1.2.4. Further Step: After all the given steps above a database design will be completed by applying some further steps as follows: These are steps out of subject and may be covered in another detailed research. Anyway let us give these steps to as a conclusion.

- Schema Refinement
    - Allows deciding whether it is a good design or not. (Normalization)
    - If not, the schema will be decomposed into smaller parts. (Decomposition)
- Physical Design
    - Decision of how to store the database. (File Storage, Indexing)
- Security Design
    - Security & Authorization:
    - Three main security objectives: secrecy, integrity, and availability.
    - DB Admin is responsible for overall security
- Designs security policy, maintains history of users' accesses to the database

CHAPTER THREE
THE APPLICATION

3. The SMYRNA Hospital Information System

The aim of this work is to represent the newly developed SMYRNA Hospital Information System that uses the benefits of ASP.NET to make urgent patient data available from everywhere.

The infrastructure is based on the web-technologies, and as the interface the well-known web browser is used. All the required patient data is collectible on the web browser, thanks to the interoperability between the web server and the database server. Searching is enabled by using query and by browsing hierarchical classified data chunks.

Web based information is implemented by use of the simple Hypertext Markup Language (HTML). This makes the system simple as possible, whereas the operation behind can be as complex as possible.

In the design step of the system application, a 3-layered architecture has been considered.

The first (bottom) layer is the data access layer. This layer consists of a collection of databases. It contains collected patient/physician/personnel records, laboratory results, patient scanned images, drug information, etc. An SQL2000 Server manages the data collected and stored. User defined queries enables the access to the required data. This layer also contains a patient folder system, which stores the 'imaged' history of the patients. The previously scanned images, lab reports, diagnosis and others.

The second (middle) layer is the web access layer. This layer makes the information available outside. Data communication between the data access layer and the user layer flows through this layer. Therefore this layer has to be responsible for the security of the whole system. Authorization, authentication and auditing are being held on this layer. Once a user is authorized the answers for its queries is transformed into HTML format by this layer to make the results visible on the top layer (user interaction layer). This layer operates on ASP.NET platform to make the client-server interaction more efficient.

The last (top) layer is the user interaction layer, which is the user application level. The users may be any physician, nurse, patient, or even any other user that have Internet access. At this layer any device like desktop computers or notebooks on which any web browser is enabled can access the application by just writing the address of the web server to the address bar.

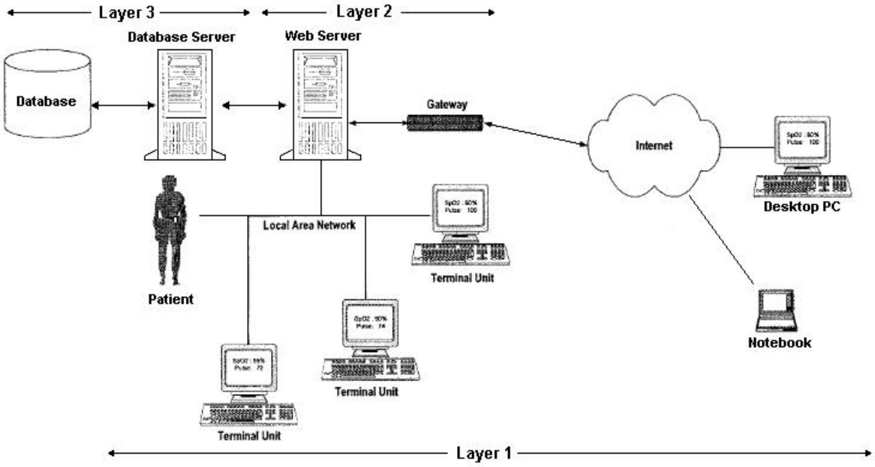


Figure 3.1 The 3-layered architecture of the HIS application

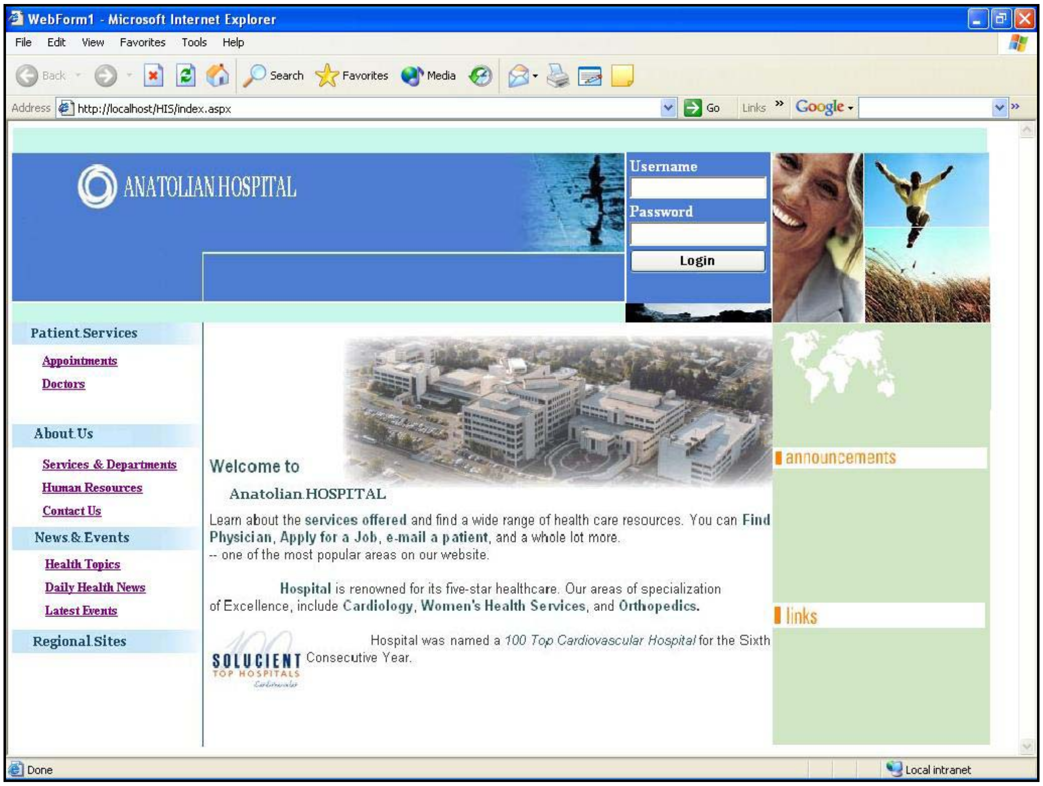
3.1 The Forms

The SMYRNA HIS consists of 9 main web forms. Some include sub-forms to realize user navigation. This chapter investigates the role of each main form and its sub-forms in the Hospital Information System.

3.1.1 Index

This form is the first form a user reaches. From this form each user will be directed to his / her form in interest. 6 types of users are assigned to use the application:

1. Physician
2. Nurse
3. Laboratory Personnel
4. Radiology Personnel
5. Admission Personnel
6. Patient Relative



Welcome to

Anatolian HOSPITAL

Learn about the **services offered** and find a wide range of health care resources. You can **Find Physician**, **Apply for a Job**, **e-mail a patient**, and a whole lot more.
-- one of the most popular areas on our website.

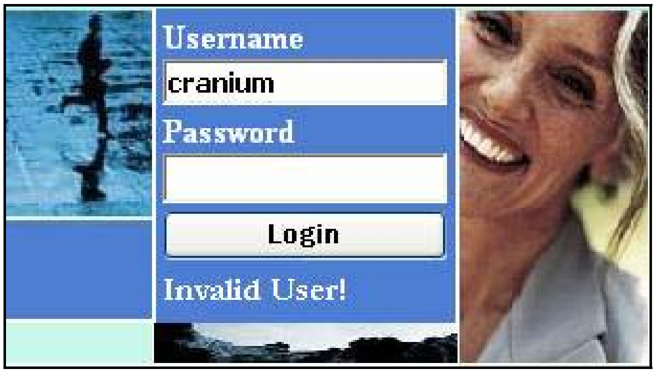
Hospital is renowned for its five-star healthcare. Our areas of specialization of Excellence, include **Cardiology**, **Women's Health Services**, and **Orthopedics.**

Hospital was named a *100 Top Cardiovascular Hospital* for the Sixth Consecutive Year.

Figure 3.2 The main web page of the HIS application

Each user is assigned a login name and a password, which should be entered into the login part of the index page. See Figure 3.2. Entering this information into the boxes and hitting the login button enables a search in the users database previously constructed. If the username and password could be found (if valid) in the database its user type will be extracted and through this information the user will be redirected to his / her allowed web page.

In case if the user is not valid or the user is valid but the password does not match a warning message appears suitable for the situation. Figure 3.3 shows the invalid user message, indicating a mismatch of password or username.


Username
cranium
Password

Invalid User!

Figure 3.3 Login screen and warning message indicating invalid users.

The main page is used to give information about the hospital. A brief explanation of the main tasks the hospital covers, departments, etc. may be given in the front. The left-hand-side column is used like any other portal as an info column. Here one can reach:

Patient Service: Users can use this column to make an online appointment with physicians. Also each physician info may be obtainable through this section.

About Us: Detailed information about the hospital, the human resources section, and the contact information are reachable in this section.

News & Events: Latest information, health topics, and daily health news may be obtainable here.

Regional Sites: Other sites of the hospital if available may be put here.

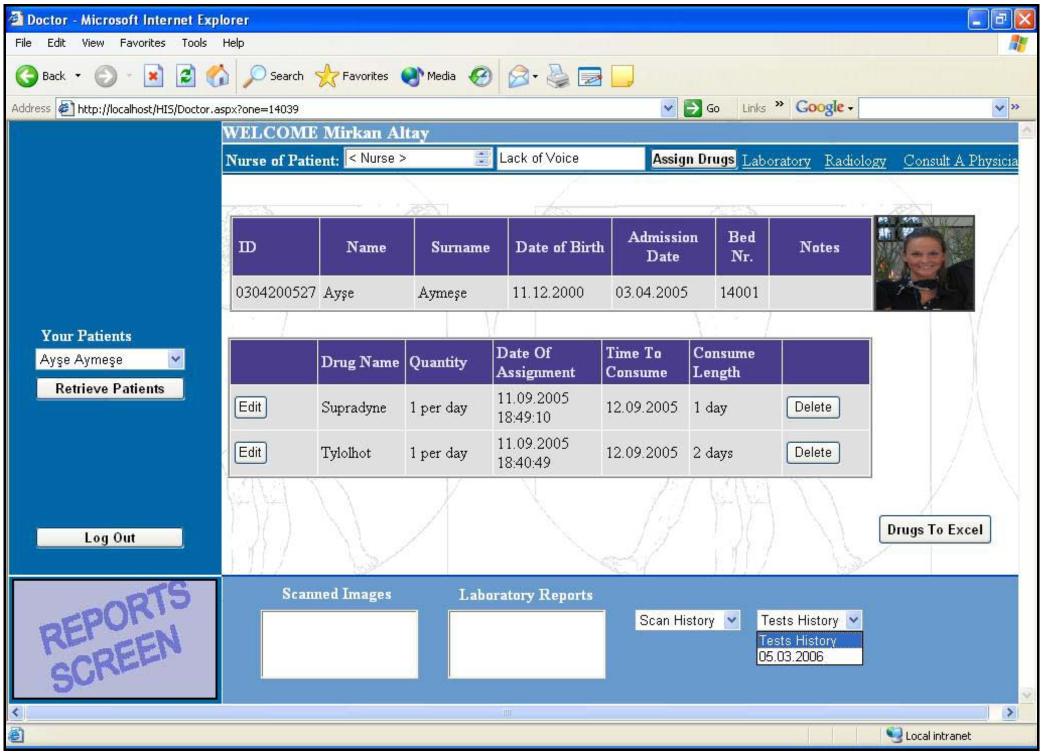
If desired, this column and any other info-section of the main page can be changed. So it can be used as a hospital information portal to give knowledge about health and medicine.

3.1.2 Physician

This is the form where physicians take the entire control of his / her patients. The physicians' form consists of four main information blocks:

1. Physician Information Block
2. Patient Information Block
3. Assignment Block
4. History Block

Figure 3.4 shows the four main blocks of the physician form.


Doctor - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites Media
Address http://localhost/HIS/Doctor.aspx?one=14039
Go Links Google
Assign Drugs Laboratory Radiology Consult A Physicia
Your Patients
Retrieve Patients
Log Out
Drugs To Excel
REPORTS SCREEN
Scanned Images
Laboratory Reports
Scan History
Tests History
Local intranet

WELCOME Mirkan Altay

Nurse of Patient: < Nurse > Lack of Voice

ID	Name	Surname	Date of Birth	Admission Date	Bed Nr.	Notes
0304200527	Ayşe	Aymeşe	11.12.2000	03.04.2005	14001	

Your Patients: Ayşe Aymeşe

	Drug Name	Quantity	Date Of Assignment	Time To Consume	Consume Length	
Edit	Supradyne	1 per day	11.09.2005 18:49:10	12.09.2005	1 day	Delete
Edit	Tylolhot	1 per day	11.09.2005 18:40:49	12.09.2005	2 days	Delete

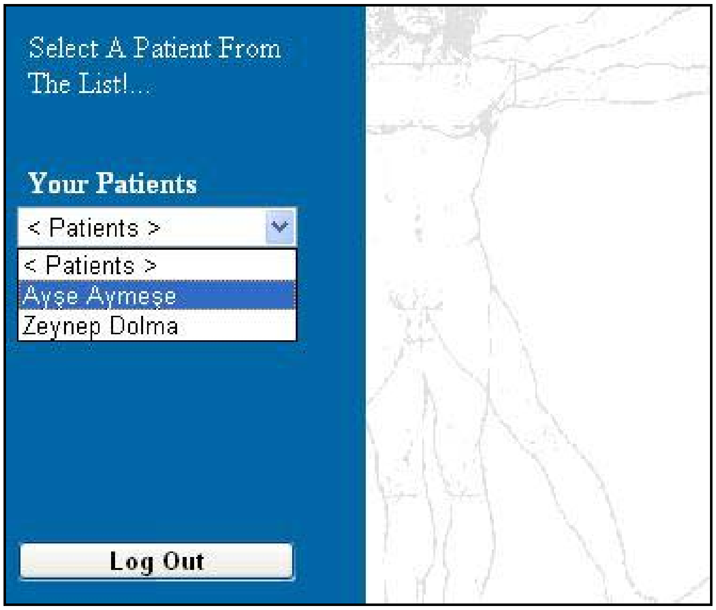
Tests History: Tests History, 05.03.2006

Figure 3.4 A physician's user interface.

3.1.2.1 Physician Information Block

This is the area where information about the physician is being held.

The physician has to just to hit the “Retrieve Patients” button. After hitting the button a query searches through the physician and patient tables of the database. It collects the physician’s patients and fills the patient dropdown list just above the button. Figure 3.5 shows patients assigned to a physician.


Select A Patient From The List!...

Your Patients

< Patients >

< Patients >
Ayşe Aymeşe
Zeynep Dolma

Figure 3.5 ‘Physician Information Block’. Physician with Id 14039 chooses one of his / her patients from the patient dropdown list.

Hitting the “Retrieve Patients” button changes also the on top written ‘PHYSICIAN’ label and welcomes the physician who is using the form at the moment.

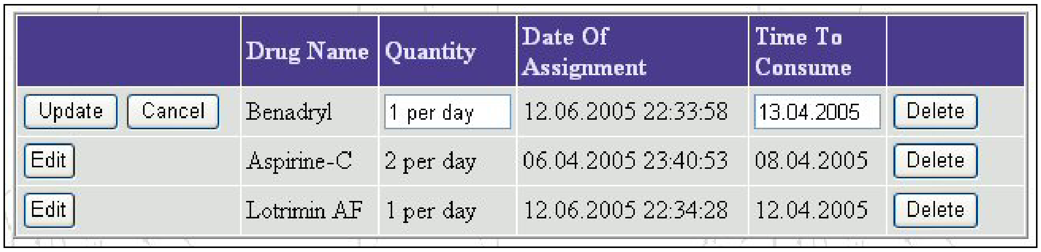
Another and the last button in this block is the logout button. Hitting this button closes the physician’s session and redirects the browser to the hospital main page. At this situation one has to login again to reopen a session.

3.1.2.2 Patient Information Block

This block resides in the middle of the form and is the main information block. Here the information about the selected patient and his / her assigned drug will be shown. After selecting a patient from the dropdown list all information will be collected available from the database. This information is filled into two main data grids residing in this block.

The first data grid shows general info (patient Id, name, surname, date of birth, admission date, bed number, and notes about the patient such as allergies, previous illnesses etc.) of the selected patient. This data grid also shows a picture of the patient. From this data grid the physician can easily see a brief info of his / her patient. No content in this data grid can be changed by anyone.

The second data grid is the drug data grid. Here the drug information (drug name, quantity to consume, date of assignment, time to consume) assigned to the patient will be shown. This is a dynamic data grid where the physician has the ability to change drug data assigned. Pressing the Edit button inside the data grid opens text boxes inside. (And also the edit button vanishes and two other button, update and cancel buttons, appear). This means that the physician can change the contents. Any unchangeable content will not turn into a text box. Figure 3.6. The physician can update the quantity and the time to consume content. Hitting the update button updates the changed content of the drugs and consumes data tables of the hospital database. Now the new content will be visible. A successive update can be realized from the warning label writing on it: “1 records updated”. A failure of an update will be indicated with this label writing on it: “Error Updating Patient!…” For a non-update, hitting the cancel button will be enough. The update and cancel buttons vanishes and the edit button reappears inside the data grid.

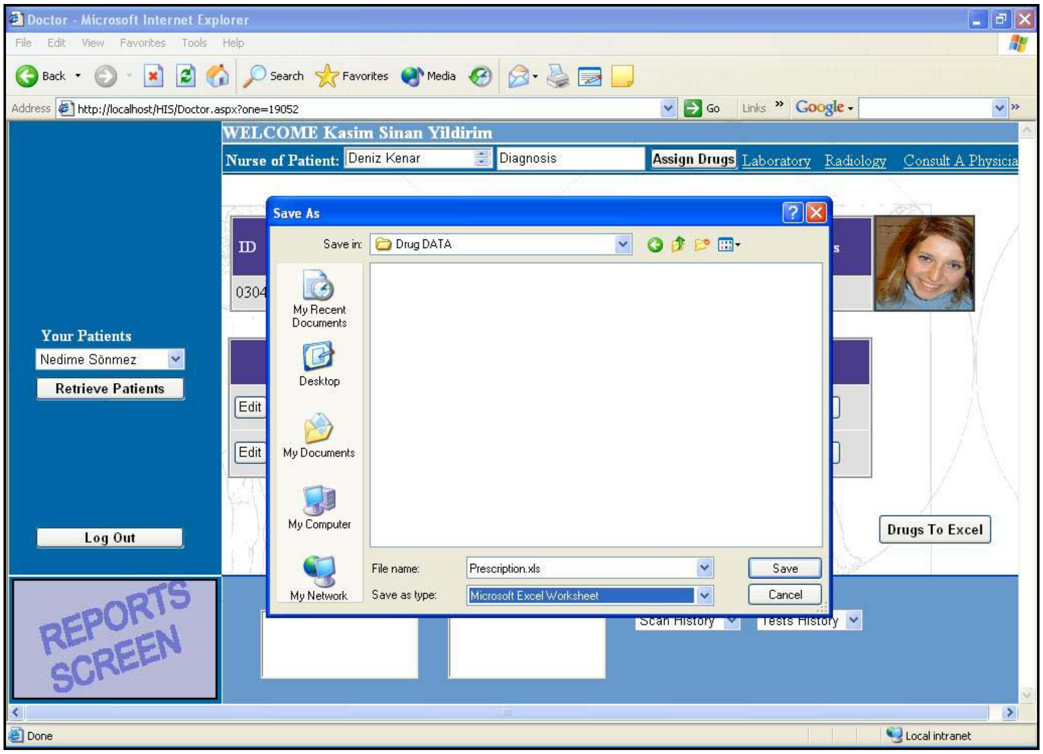


	Drug Name	Quantity	Date Of Assignment	Time To Consume	
Update Cancel	Benadryl	1 per day	12.06.2005 22:33:58	13.04.2005	Delete
Edit	Aspirine-C	2 per day	06.04.2005 23:40:53	08.04.2005	Delete
Edit	Lotrimin AF	1 per day	12.06.2005 22:34:28	12.04.2005	Delete

Figure 3.6 ‘Drug data grid’ in Edit mode.

The physician may decide that a previously given drug should not be consumed anymore. Hitting just the delete button inside right of the data grid will delete the selected drug both from the database and from the data grid. The physician may also add new drugs to the data grid. This procedure will be explained later in the “Assignment Block” of this chapter.

The drugs assigned to a patient may be printable due to the “print to excel” button. If this button is hit all the assigned drugs are exported to an .xls file. On any type of prescription paper this .xls file can be printed. This avoids wrong drug consumptions due to unreadable prescriptions. Figure 3.7 and Figure 3.8.



Address: http://localhost/HIS/Doctor.aspx?one=19052

Nurse of Patient: Deniz Kenar

Diagnosis

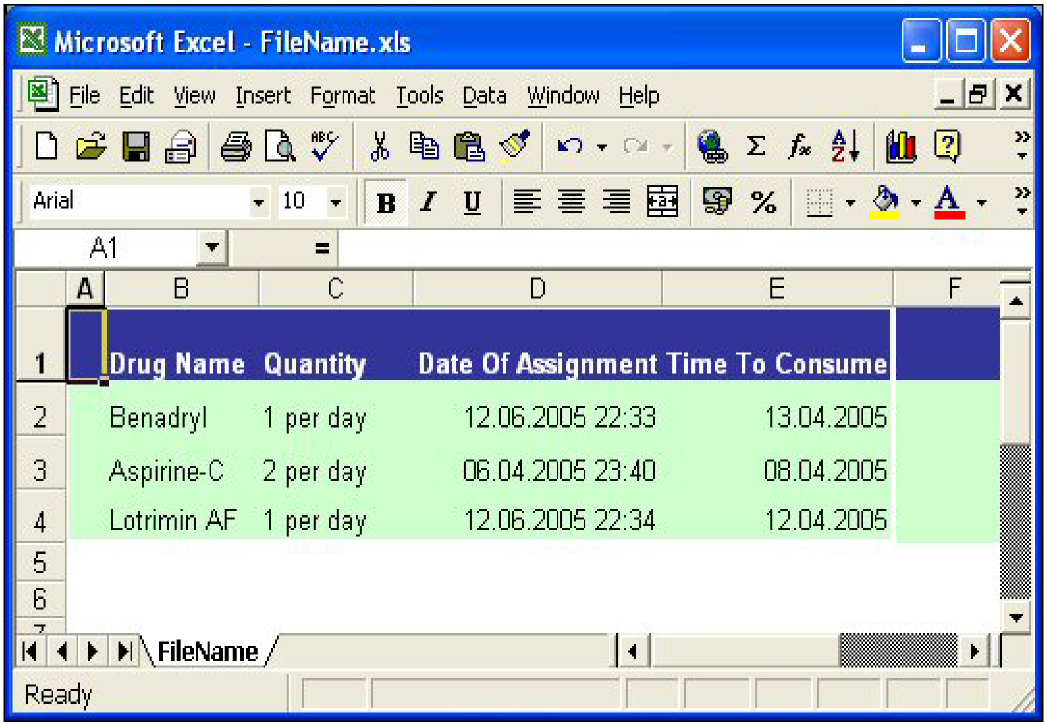
Your Patients: Nedime Sönmez

Save in: Drug DATA

File name: Prescription.xls

Save as type: Microsoft Excel Worksheet

Figure 3.7 A physician can print the drugs assigned into an excel file for subsequent usage purposes.

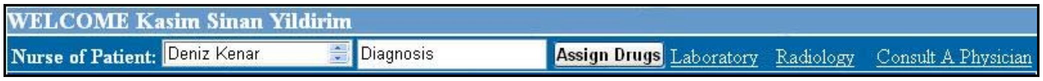

Microsoft Excel - FileName.xls
File Edit View Insert Format Tools Data Window Help
Arial
10
A1
=
FileName
Ready

	A	B	C	D	E	F
1		Drug Name	Quantity	Date Of Assignment	Time To Consume	
2		Benadryl	1 per day	12.06.2005 22:33	13.04.2005	
3		Aspirine-C	2 per day	06.04.2005 23:40	08.04.2005	
4		Lotrimin AF	1 per day	12.06.2005 22:34	12.04.2005	
5						
6						

Figure 3.8 Drugs saved into an excel file.

3.1.2.3 Assignment Block

The ‘Assignment Block’ is the block residing on the top of the form. This block is created for assignment procedures. Physicians may assign drugs, laboratory tests, and radiology scans to his / her patients. (Figure 3.9) On this block the physician also determines the diagnosis for his / her patients.


WELCOME Kasim Sinan Yildirim

Nurse of Patient: Deniz Kenar

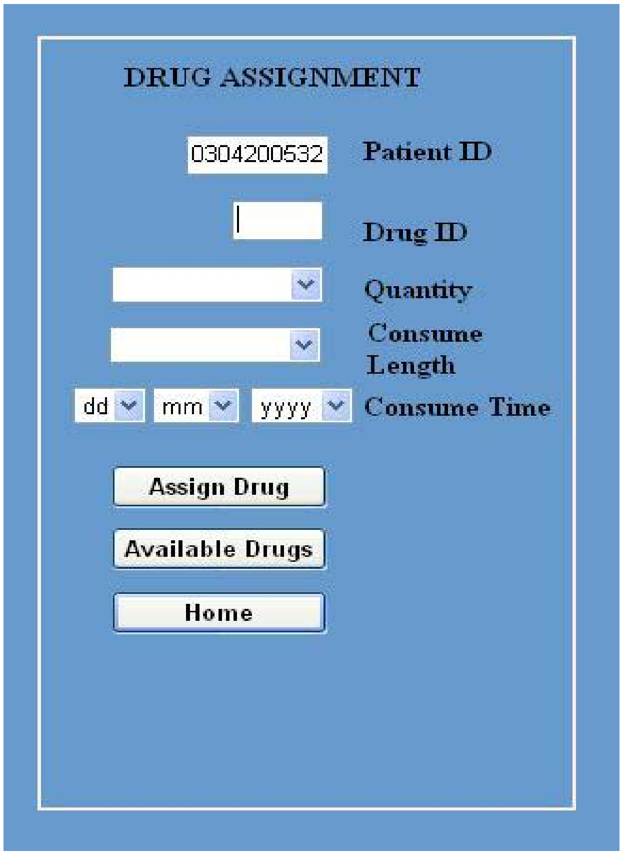
Diagnosis

Figure 3.9 The ‘Assignment Block’ of the ‘Physician’ web page.

3.1.2.4 Drug Assignment Block

The first assignment is the drug assignment button. By clicking this button the physician will be directed to the drug assignment page. Here the physician has the ability to assign drugs to his / her patients. The physician has to fill the Patient Id, Drug Id, Quantity, Consume Length, and Consume Time fields to assign a drug. The Patient Id comes written when entered this page.

Through the available drugs button the physician chooses the best drug to his / her patients. Clicking this button opens the drug form, which will be discussed later in detail. If at least one of the entries is left empty the validation exceptions will be thrown and the required field will be indicated through warning labels. At such a situation the drug assignment button becomes inactive. So no drug assignment will be made accidentally. Even if there is something wrong with the Patient Id or Drug Id no assignment will be made. A "Drug Rejected!…" warning message will be shown to indicate the mismatch. For valid patient and drug Id's a "Drug Accepted!…" message appears. Meanwhile a new row has been created in the Consumes Relation of the hospital database. The "Home" button returns the physician to his/her initial page where he can check if the drug is assigned or not.



DRUG ASSIGNMENT

0304200532 Patient ID

Figure 3.10 The 'Drug Assignment' page

3.1.2.5 Sign In to Laboratory

The second assignment is the laboratory test assignment hyperlink. By clicking this link a sub window opens inside left of the same window. Here the physician has the ability to assign test to his / her patients. The physician fills just the Patient Id and chooses the test he / she wants to assign for the patient.

Not entering the Patient Id results in not assigning a test. A validation exception will be fired and this triggers the warning label to show the text: “Enter Patient Id”. For a wrong patient Id, tests do not assign and the warning label text turns to “0 Tests Accepted!…”

After a successful test assignment the label shows how many tests are accepted. e.g.: “3 Tests Accepted!…” These assigned tests will then be visible on the laboratory form. The laboratory personnel then make the desired tests.

3.1.2.6 Sign In to Radiology

The third and last assignment is the radiology scans assignment hyperlink. By clicking this link a subwindow opens inside left of the same window. Here the physician has the ability to assign scans to his / her patients. The physician fills just the Patient Id and chooses the scans he / she wants to assign for the patient.

Not entering the Patient Id results in not assigning scans. A validation exception will be fired and this triggers the warning label to show the text: “Enter Patient Id”. For a wrong patient Id, scans do not assign and the warning label text turns to “0 Tests Accepted!”.

After a successful scan assignment the label shows how many scans are accepted. e.g.: “3 Tests Accepted!…” These assigned tests will then be visible on the radiology form. The radiology personnel then make the desired tests.

Here at the ‘Assignment Block’ the diagnosis of the patient can be entered. Each time the patient data is shown on the ‘Patient Information Block’ the diagnosis also appears here in the textbox. This is made on purpose as a textbox to give the physician the ability to change the diagnosis if desired. The diagnosis information is added to the history records for later use.

The last feature of this block is that it shows the nurses assigned to the patient. Two nurses to each patient is assigned during admission.

3.1.2.7 History Block

The ‘History Block’ is at the bottom of the physician form. As its name implies physicians can check previous laboratory and radiology results of the selected patient.

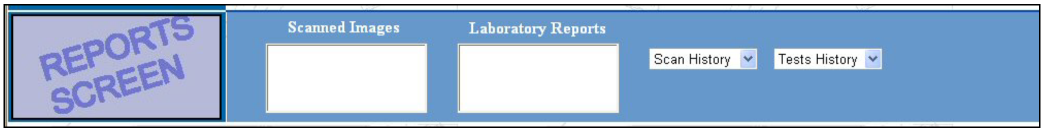
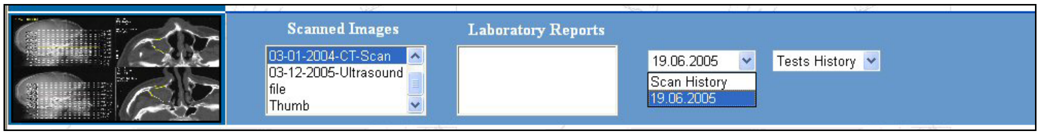


Figure 3.11 The ‘History Block’ at the bottom of the physician web page.

Tiling the block vertically into three: The right part is the one (date screen) where the desired date selectable is. Two dropdown lists are placed here. The physician selects here the date whether from the laboratory results or from the radiology results dropdown list. The middle part (test results screen) is the one where the names for that day of results appear. Two lists are present. One placed for the laboratory and the other one for the radiology results. Selecting the date enables a query that fills the lists. Selecting one of them shows a preview of the result on the left part (report preview screen). Clicking on the preview opens the test result in a separate window. The physician can investigate the result or even can copy them to his / her local folders. This feature prevents excessive paper staying in the way. Storing results will be more accurate and efficient for the physician.



Scanned Images

03-01-2004-CT-Scan
03-12-2005-Ultrasound
file
Thumb

Laboratory Reports

19.06.2005
Scan History
19.06.2005

Tests History

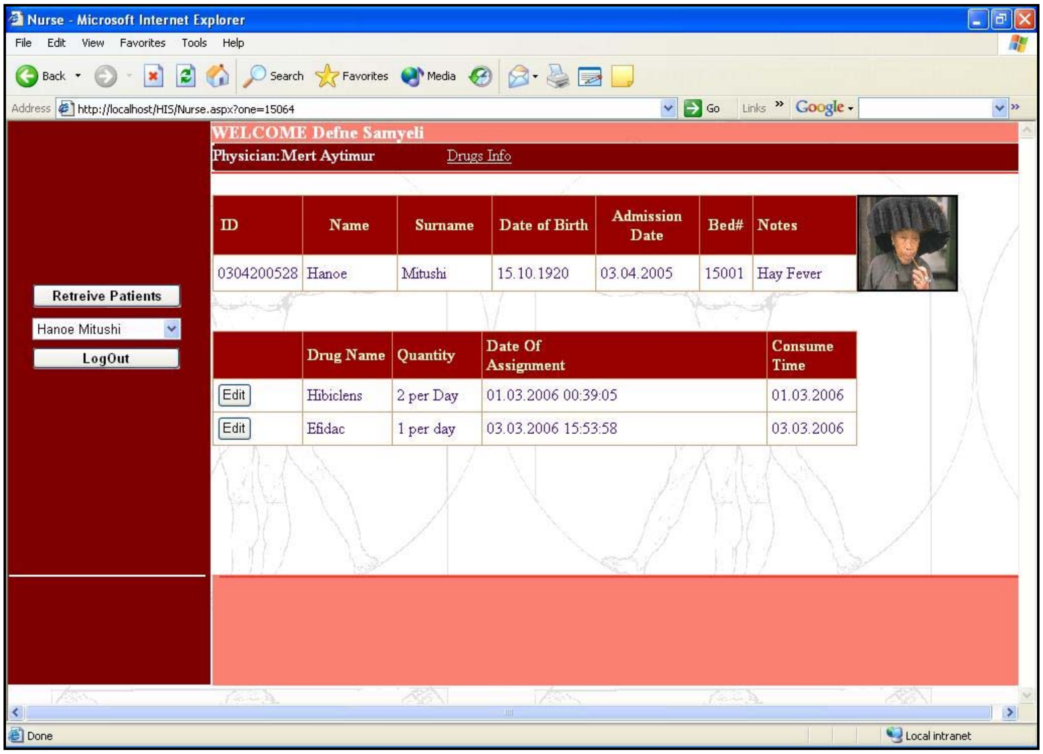
Figure 3.12 The physician may choose from the history dropdown lists a date. According to the chosen date the scanned image and laboratory reports lists will be filled with the reports. A chosen one will be shown on the left of the block as a preview.

3.1.3 Nurse

This is the form where nurses take the entire control of his / her patients. The nurse form consists of two main information blocks:

1. Nurse Information Block
2. Patient Information Block

Figure 3.13 shows the two main blocks of the nurse form.



Nurse - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites Media
Go Links Google
Done
Local intranet

Address http://localhost/HIS/Nurse.aspx?one=15064

WELCOME Defne Samyeli

Physician:Mert Aytimur Drugs Info

ID	Name	Surname	Date of Birth	Admission Date	Bed#	Notes
0304200528	Hanoe	Mitushi	15.10.1920	03.04.2005	15001	Hay Fever

Retreive Patients

Hanoe Mitushi

LogOut

	Drug Name	Quantity	Date Of Assignment	Consume Time
Edit	Hibiclens	2 per Day	01.03.2006 00:39:05	01.03.2006
Edit	Efidac	1 per day	03.03.2006 15:53:58	03.03.2006

Figure 3.13 The ‘Nurse’ page of the HIS residing two main information blocks.

3.1.3.8 Nurse Information Block

This is the area where information about the nurse is being held.

The nurse has to hit the “Retrieve Patients” button to reach patient information. After hitting the button a query searches through the nurse and patient tables of the database. It collects patients assigned to the nurse and fills the patient dropdown list just below the id box. Figure 3.14 shows patients assigned to a nurse.

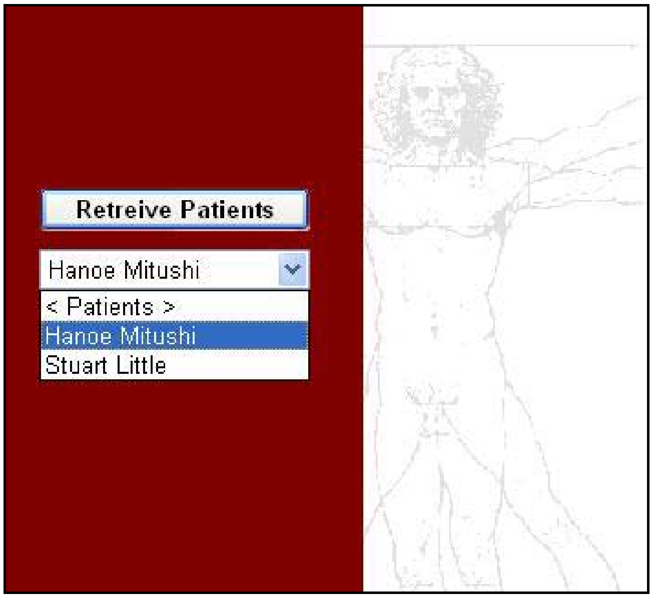


Figure 3.14 Patients assigned to nurse with Id 15064 listed in the patient dropdown list.

Hitting the “Retrieve Patients” button changes also the on top written ‘NURSE’ label and welcomes the nurse who is using the form at the moment. Also the name of the assigned physician appears on top of the screen.

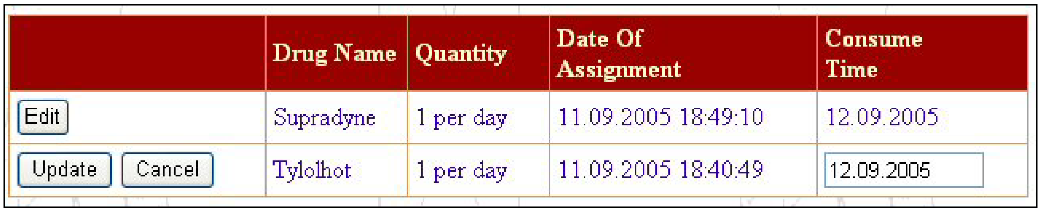
Another and the last button in this block is the logout button. Hitting this button closes the nurse’s session and redirects the browser to the hospital main page. At this situation one has to login again to reopen a session.

3.1.3.9 Patient Information Block

This block resides in the middle of the form and is the main information block. Here the information about the selected patient and his / her assigned drug will be shown. After selecting a patient from the dropdown list all information will be collected available from the database. This information is filled into two main data grids residing in this block.

The first data grid shows general info (patient Id, name, surname, date of birth, admission date, bed number, and notes about the patient such as allergies, previous illnesses etc.) of the selected patient. This data grid also shows a picture of the patient. From this data grid the nurse can easily see a brief info of his / her patient. No content in this data grid can be changed by anyone.

The second data grid is the drug data grid. Here the drug information (drug name, quantity to consume, date of assignment, time to consume) assigned to the patient will be shown. This is a dynamic data grid where the physician only has the ability to change drug data assigned. Nurse can just change the time to consume field. Pressing the Edit button inside the data grid opens text boxes inside. (And also the edit button vanishes and two other button, update and cancel buttons, appear). This means that the nurse can change the contents. Any unchangeable content will not turn into a text box, shown Figure 3.15. The nurse can update only the time to consume content. Hitting the update button updates the changed content of the drugs and consumes data tables of the hospital database. Now the new content will be visible. A successive update can be realized from the warning label writing on it: “1 records updated”. A failure of an update will be indicated with this label writing on it: “Error Updating Patient!…” For a non-update, hitting the cancel button will be enough. The update and cancel buttons vanishes and the edit button reappears inside the data grid.



	Drug Name	Quantity	Date Of Assignment	Consume Time
Edit	Supradyne	1 per day	11.09.2005 18:49:10	12.09.2005
Update Cancel	Tylolhot	1 per day	11.09.2005 18:40:49	12.09.2005

Figure 3.15 A nurse can change just the consume time of the assigned drugs. The nurse enters the time when the patient has consumed the drug.

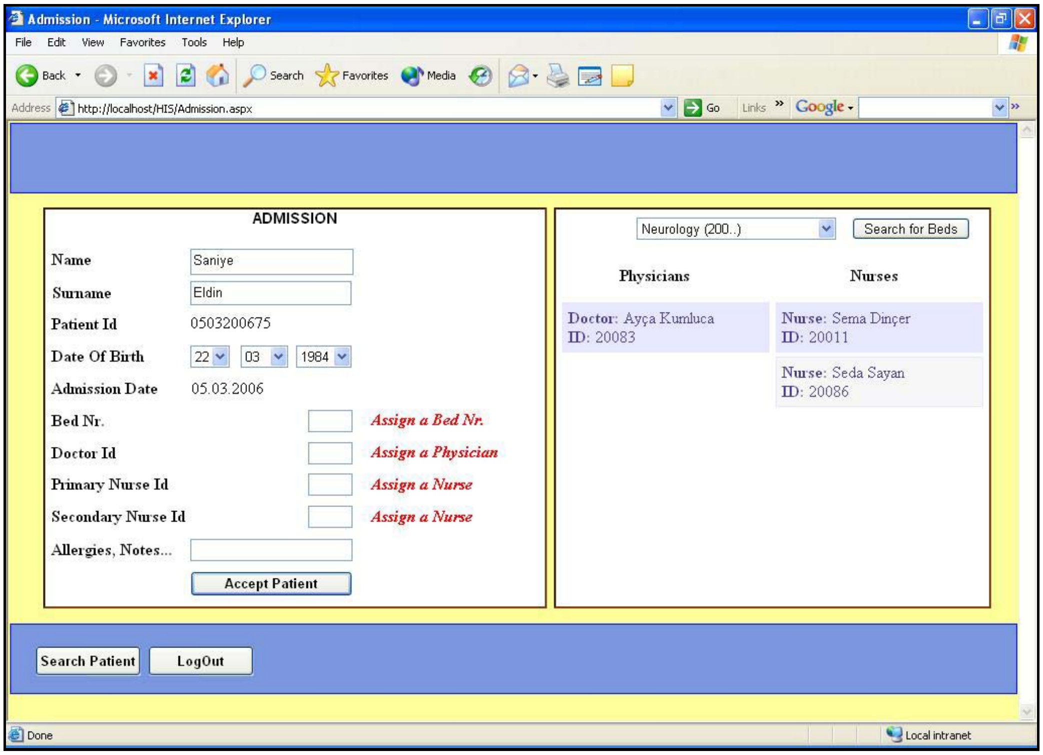
There is just a hyperlink for the nurse linking to the drugs form, where the nurse may take any information about the drugs assigned to a patient.

3.1.4 Admission

From the main page of the application the admission personnel will be directed to this form. It consists of a main window and its sub windows. The main window as shown in Figure 3.16 is tiled into two parts. On the left located part is the admission part whereas on the right located part is the department information part. At the admission part new patient records are made. The desired information is name, surname, date of birth, bed number, physician Id, and nurse Id (for both primary and secondary nurse). The allergies and notes information is optional. All patients are given a unique patient Id. This patient Id is updated each time a new record is made. The patient Id is in the format of:

date + count

The first 8 digits indicate the admission date and the next digits (from 0 to 9999) indicate the count of patient this day. Each reload of the page, which means that a record was made, updates the number by adding 1 to it. This format ensures the uniqueness of the Id, because no duplicate of date will occur.


Admission - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites Media
Address http://localhost/HIS/Admission.aspx
Go
Links
Google
Search for Beds
Accept Patient
Search Patient
LogOut
Done
Local intranet

ADMISSION

Name: Saniye
Surname: Eldin
Patient Id: 0503200675
Date Of Birth: 22 03 1984
Admission Date: 05.03.2006
Bed Nr. *Assign a Bed Nr.*
Doctor Id *Assign a Physician*
Primary Nurse Id *Assign a Nurse*
Secondary Nurse Id *Assign a Nurse*
Allergies, Notes...

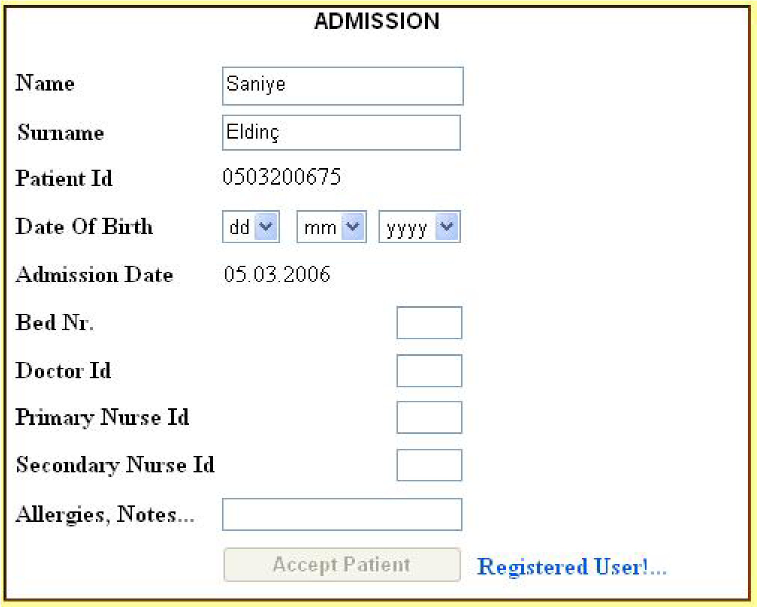
Neurology (200..)

Physicians	Nurses
Doctor: Ayça Kumluca ID: 20083	Nurse: Sema Dinçer ID: 20011
	Nurse: Seda Sayan ID: 20086

Figure 3.16 ‘Admission’ page consisting of two parts. Left: Admission part, right: department information part.

During admission the required fields, as mentioned before, must be filled. Any empty required field triggers its warning label and tells the user to enter the required data. Otherwise no admission will be made.

During Admission, writing the name and the surname of the patient triggers a database search. If such a user is previously registered the “Accept Patient” button becomes inactive to prevent a duplicate admission. A warning also appears that this user is a “registered” user. (Figure 3.17)



ADMISSION

Name	Saniye
Surname	Eldinç
Patient Id	0503200675
Date Of Birth	dd mm yyyy
Admission Date	05.03.2006
Bed Nr.	
Doctor Id	
Primary Nurse Id	
Secondary Nurse Id	
Allergies, Notes...	

Figure 3.17 A registered user admission. “Accept Patient” button inactive.

Two required fields are the Dr. Id, primary nurse Id, and the secondary nurse Id fields. The admission personnel have to fill these forms. The required information can be obtained from the department information part of the admission page (On the right located part). There is a department dropdown list. Here one selects any department to get the names and id’s of the physicians and nurses working in that department. The admission personnel easily get both physician and nurse id’s and complete the form. Also for inpatient situations there should be a bed prepared for the patient. The ‘Search for bed’ button shows the selected department and its allocation status. Figure 3.18 shows the status of beds in the critical care department. Green beds show that they are allocated, whereas gray beds indicate that they are free. The admission personnel choose a suitable one and finish to enter the admission data.

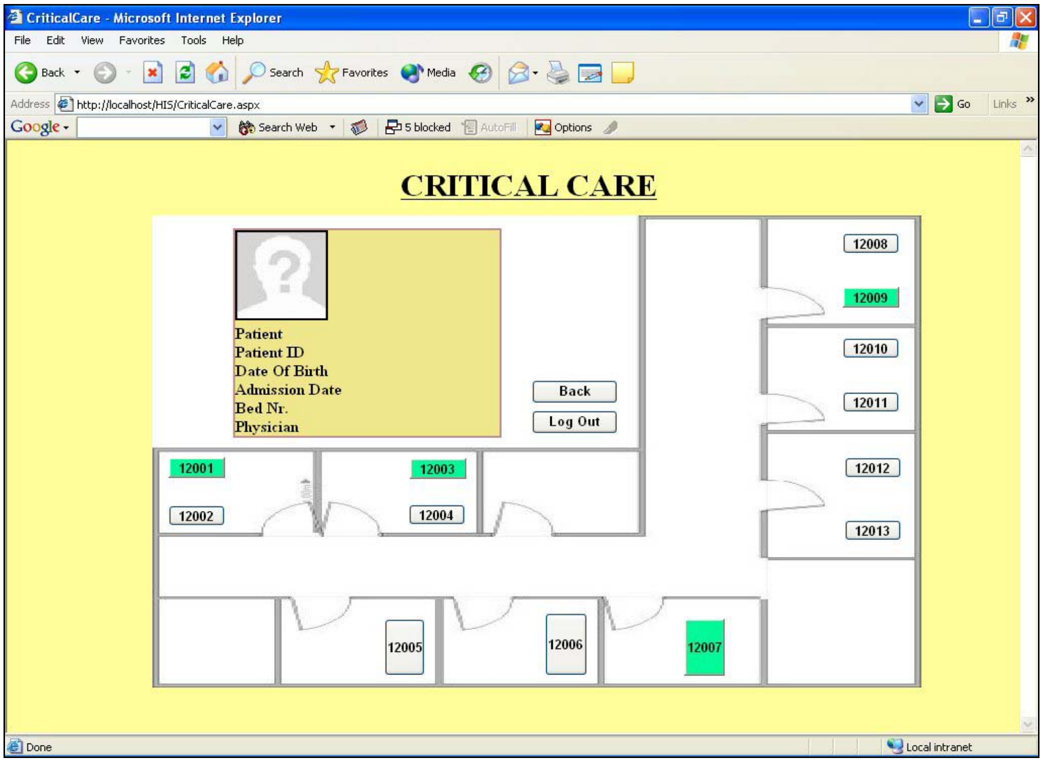
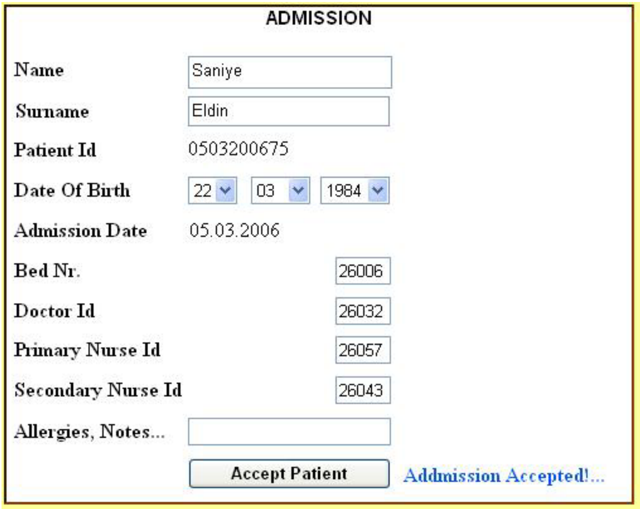

CriticalCare - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites Media
Address http://localhost/HIS/CriticalCare.aspx Go Links
Google Search Web 5 blocked AutoFill Options
CRITICAL CARE
Patient
Patient ID
Date Of Birth
Admission Date
Bed Nr.
Physician
Back
Log Out
12008
12009
12010
12011
12012
12013
12001
12002
12003
12004
12005
12006
12007
Done
Local intranet

Figure 3.18 The admission personnel can reach the bed information from the floor plans. Green beds indicate that the beds are full.

After have been filled all fields one hits the admission button just below the form to create a new record. A successful admission will be indicated with the message "Admission Accepted!" appearing beneath the button. Figure 3.19 shows an accepted admission.



ADMISSION

Name	Saniye
Surname	Eldin
Patient Id	0503200675
Date Of Birth	22 03 1984
Admission Date	05.03.2006
Bed Nr.	26006
Doctor Id	26032
Primary Nurse Id	26057
Secondary Nurse Id	26043
Allergies, Notes...	

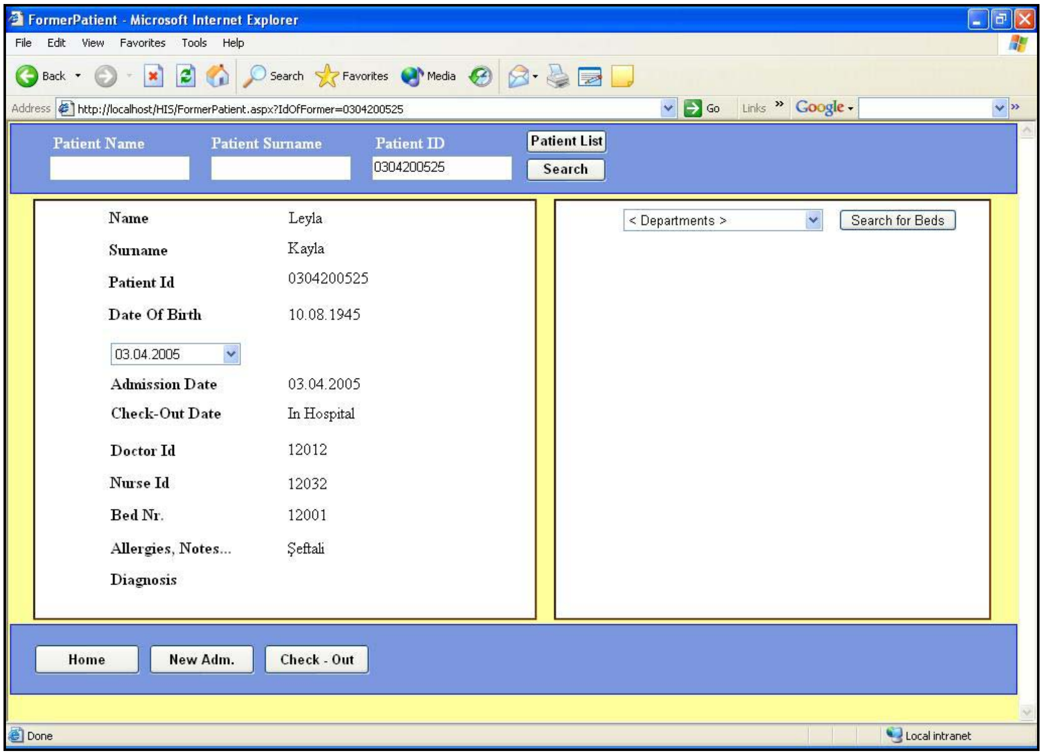
Accept Patient Admission Accepted!...

Figure 3.19 A successful admission.

3.1.4.10 Former Patients

In case if the patient admitting is a previously recorded patient (i.e. he / she just owns a patient Id) then the on the bottom-left located search button should be pressed to reach the 'Former Patients' form.

This form as can be seen in Figure 3.20 searches for previous patient information. The upper most located search block is where one has to enter the name & surname or just the Patient Id to search. No match for the search will be indicated with a "No Patient Found!" message. If no search data is entered the user will be asked for enter at least one parameter.



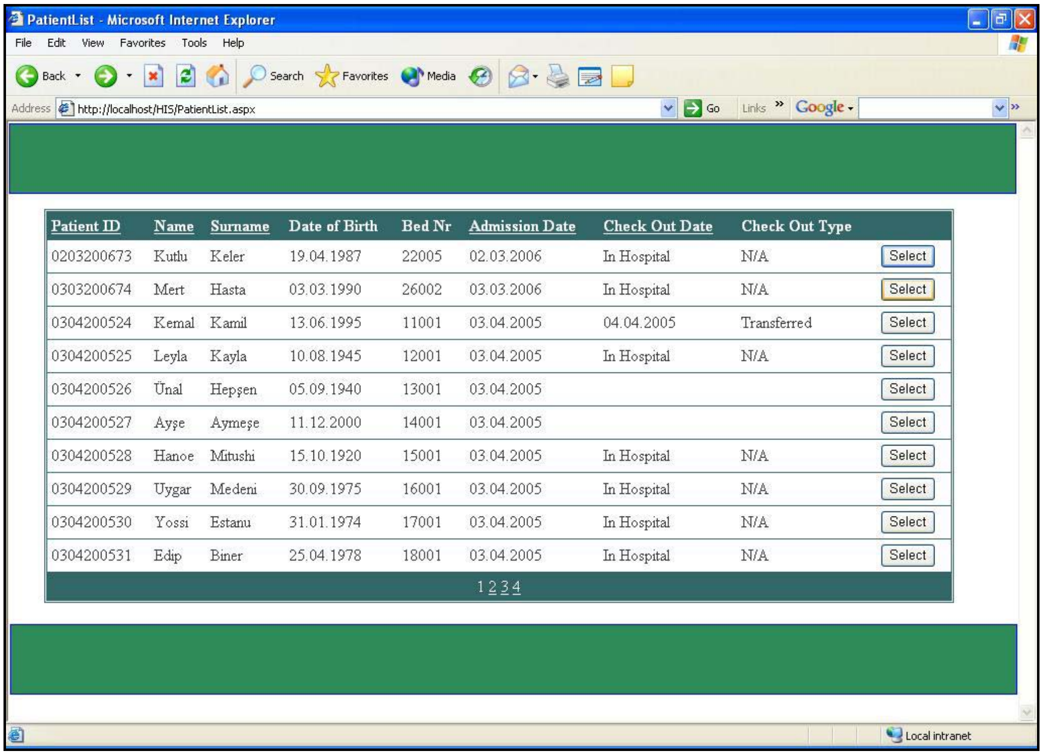
http://localhost/HIS/FormerPatient.aspx?IdOfFormer=0304200525

Patient Name	Patient Surname	Patient ID
		0304200525

Name	Leyla
Surname	Kayla
Patient Id	0304200525
Date Of Birth	10.08.1945
03.04.2005	
Admission Date	03.04.2005
Check-Out Date	In Hospital
Doctor Id	12012
Nurse Id	12032
Bed Nr.	12001
Allergies, Notes...	Şeftali
Diagnosis	

Figure 3.20 'Former Patient' page. The admission personnel may take a look at previous admissions of the same patient.

Another way of searching a former patient is to hit the "Patient List" button. A page opens where a complete list of patients is shown. The personnel may search through this table to obtain the desired patient if available. Selecting the row will return the patient Id into the Patient ID Search box in the Former Patient page. The search for previous admission data may begin.



PatientList - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites Media
Address http://localhost/HIS/PatientList.aspx
Go Links Google
Local intranet

Patient ID	Name	Surname	Date of Birth	Bed Nr	Admission Date	Check Out Date	Check Out Type	
0203200673	Kutlu	Keler	19.04.1987	22005	02.03.2006	In Hospital	N/A	Select
0303200674	Mert	Hasta	03.03.1990	26002	03.03.2006	In Hospital	N/A	Select
0304200524	Kemal	Kamil	13.06.1995	11001	03.04.2005	04.04.2005	Transferred	Select
0304200525	Leyla	Kayla	10.08.1945	12001	03.04.2005	In Hospital	N/A	Select
0304200526	Ünal	Hepşen	05.09.1940	13001	03.04.2005			Select
0304200527	Ayşe	Aymeşe	11.12.2000	14001	03.04.2005			Select
0304200528	Hanoe	Mitushi	15.10.1920	15001	03.04.2005	In Hospital	N/A	Select
0304200529	Uygar	Medeni	30.09.1975	16001	03.04.2005	In Hospital	N/A	Select
0304200530	Yossi	Estanu	31.01.1974	17001	03.04.2005	In Hospital	N/A	Select
0304200531	Edip	Biner	25.04.1978	18001	03.04.2005	In Hospital	N/A	Select

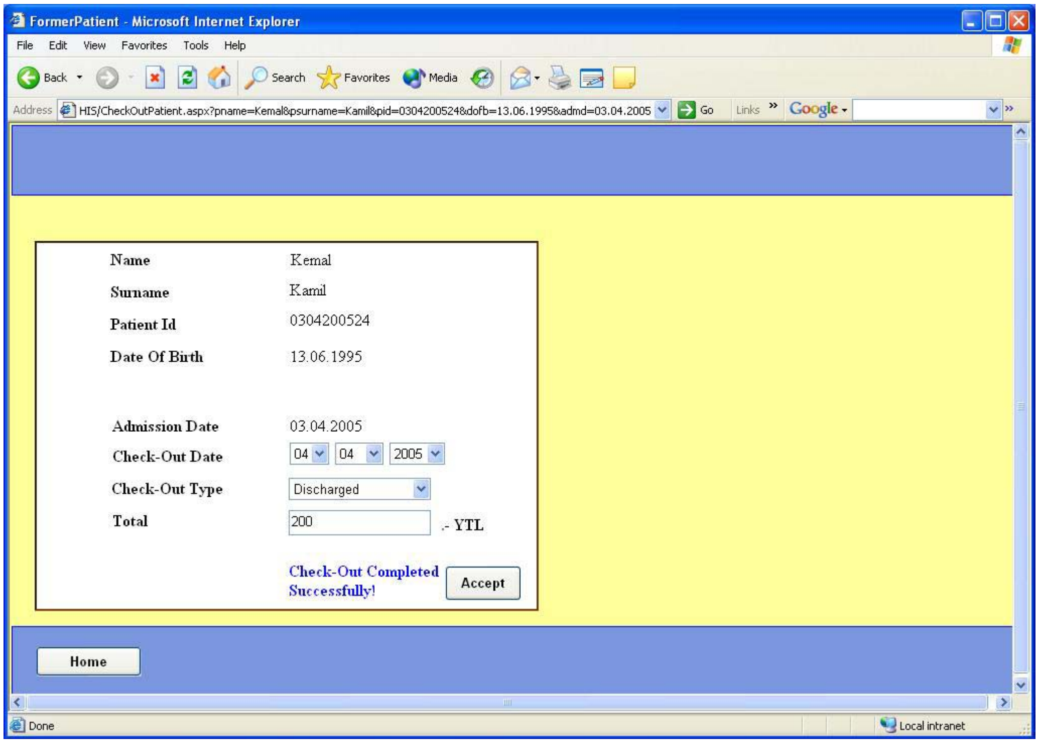
1 2 3 4

Figure 3.21 ‘Patient List’ page. The admission personnel may sort the table by patientID, name, surname, admission date or check out date.

When the searched patient has been found the first 5 data will be filled. These are: name, surname, patient Id, date of birth, and admission date. The admission date(s) will be inserted into the ‘Admission history’ dropdown list. Here one can select, if available, one of the previous admission dates to get the data at this date. After selecting a date the subsequent 7 data will be filled. These are: admission date, check out date (if available) dr. Id, nurse Id, bed number, allergies & notes, and diagnosis at this admission.

If a new admission should be made, the on the bottom located ‘new admission’ button will be pressed. So 4 textboxes appear requesting for dr. Id, nurse Id, bed number, and allergies & notes information. As like a regular admission form the right located block the dr. Id and nurse Id, also the bed number can be obtained and inserted into their corresponding textboxes to complete the admission. A successful admission will be reached after hitting the ‘Accept’ button, and getting an ‘Admission Accepted!’ message.

The nest button is the “Check Out Button”. Hitting this button redirects one to the patient check out page. (Figure 3.22) At this page the checkout of the patient can be made. Entering the date, check out type, and total afterwards hitting the “Accept” button finishes the check out procedure.


FormerPatient - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites Media
Go Links Google
Accept
Home
Done
Local intranet

Address: HIS/CheckOutPatient.aspx?pname=Kemal&psurname=Kamil&pid=0304200524&dofb=13.06.1995&admd=03.04.2005

Name	Kemal
Surname	Kamil
Patient Id	0304200524
Date Of Birth	13.06.1995
Admission Date	03.04.2005
Check-Out Date	04 04 2005
Check-Out Type	Discharged
Total	200 .- YTL

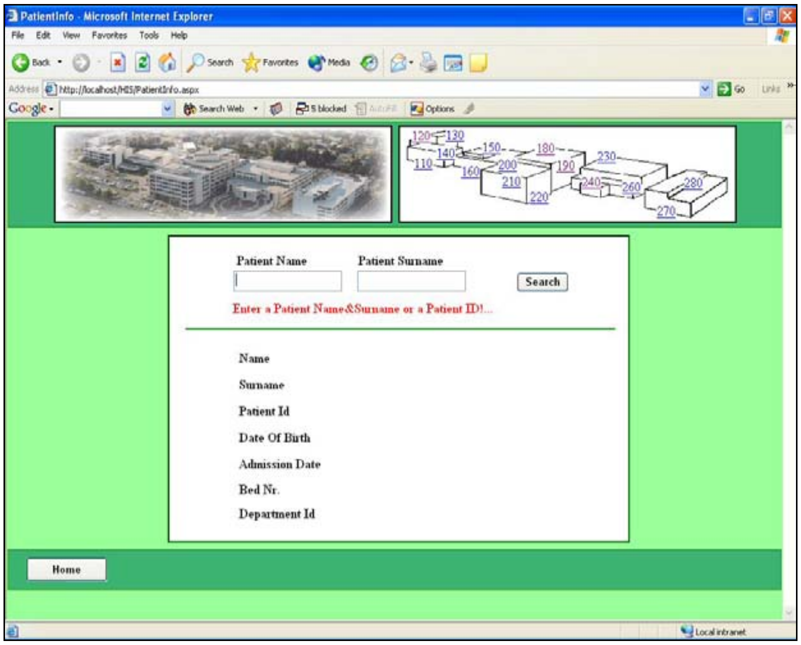
Check-Out Completed Successfully!

Figure 3.22 ‘Check Out’ sub-page.

The last button to explain is the ‘home’ button. This button returns the admission personnel to the admission main page. Also the logout button on the admission main page returns one to the application main page and closes the session. The user has to login again to open a new session to the application.

3.1.5 Patient Info

The ‘Patient Info’ form is for registered patient relatives. At admission the patient is given a username and password for his / her relatives. Through this data the patient relative can reach this page. Logging in and be directed to this page gives one the ability to search for patients staying in at the hospital. As can be seen from Figure3.23 a search block is available. One can search the patient with both its name and surname. If the patient has been found information about him / her will be inserted below the search area. Data given are: Name, surname, patient Id, date of birth, admission date, and bed number. From the bed number the patient relative can find the department as well as the location of the bed of his / her relatives. The on the top located 2 images shows the hospital and its departments. On the right positioned image includes hyperlinks to departments. The first three digits of the bed number shows the department id.



PatientInfo - Microsoft Internet Explorer

Address http://localhost/HIS/PatientInfo.aspx

Patient Name

Patient Surname

Enter a Patient Name&Surname or a Patient ID!...

Name

Surname

Patient Id

Date Of Birth

Admission Date

Bed Nr.

Department Id

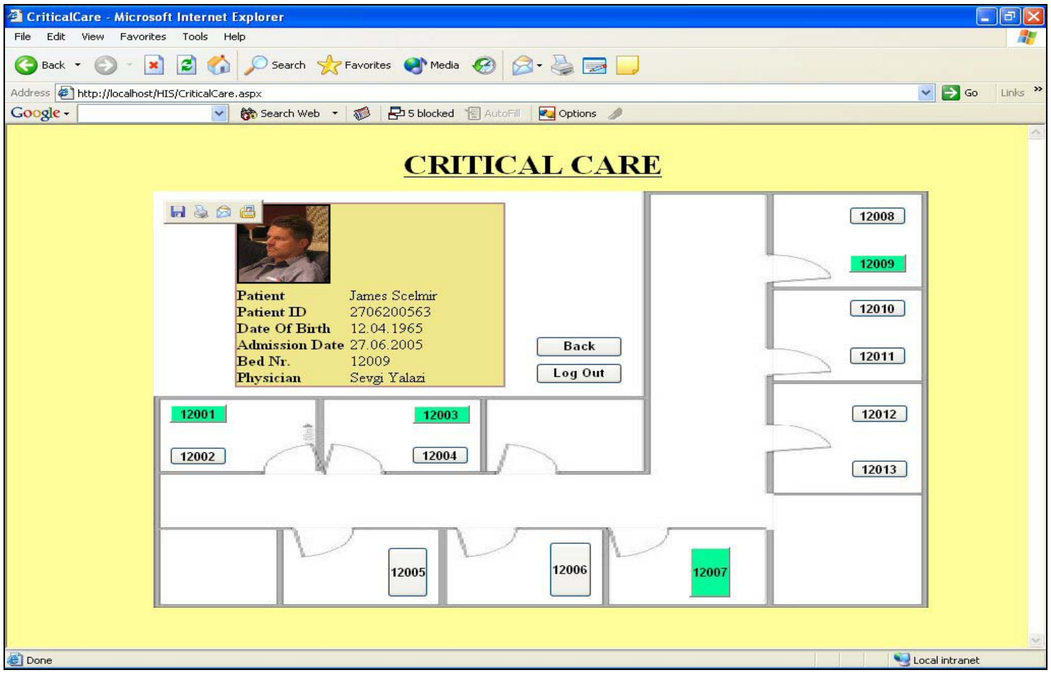
Figure 3.23 A patient search page for non-stuff users like patient relatives. From the department Id one can reach the floor plans and identify the place where the patient in question is.

e.g.: The bed number 12009 tells us that the patient is at the department with department id 120. From the upper right positioned hospital image has to click to the hyperlink named 120. The user will then be directed to the floor plan.

3.1.5.11 Hospital Floor Plans

The hospital floor plans are forms where patient locations (department and bed) can be obtained. There exist 17 departments in our HIS; therefore 17 different floor plans have been constructed.

Returning to our example: The patient relative have searched for patient 'James Scelmir' and has found that he stays in department 120 and the bed number is 12009. Clicking to the hyperlink '120' redirects the user to the floor plan of the critical care department. The floor plan of critical care is shown in Figure 3.24. On the floor plan there are buttons representing beds. On each bed is the bed number written. Beds with white color indicate that they are empty. Green beds indicate that they are allocated.



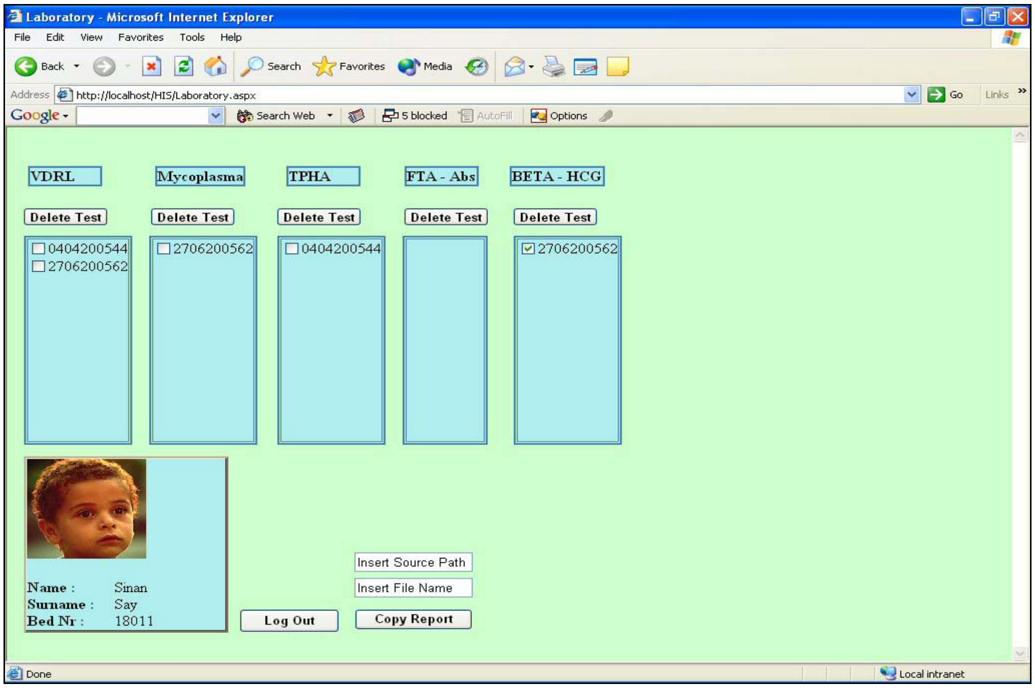
Patient	James Scelmir
Patient ID	2706200563
Date Of Birth	12.04.1965
Admission Date	27.06.2005
Bed Nr.	12009
Physician	Sevgi Yalazi

Figure 3.24 Patient 'James Scelmir' and his information shown on the patient information data grid.

James Scelmir’s relative knows that James lies in bed 12009. Clicking on the bed 12009 returns the information of James Scelmir and will be shown in the information data grid. Name, surname, date of birth, admission date, bed number, and name of the physician information are given. The picture of the patient is also shown. Next to the information data grid there are two buttons. The ‘Back’ button returns the user to the ‘Patient info’ page. The ‘Logout’ button closes the session and redirects the user to the hospital main page.

3.1.6 Laboratory

Users who are members of the laboratory staff are directed to this form after logging into the application from the main page. Each column, as can be seen in Figure 3.25 corresponds to a laboratory test. Each test contains some numbers inside. These are the patient Id’s and indicate that this test is assigned to these patients. The laboratory personnel should check the checkbox of the id to see the data of the patient. For a checked checkbox the name, surname, and bed number are shown in the left-hand side corner located data grid. Also the picture of the patient will be shown here.



http://localhost/HIS/Laboratory.aspx

VDRL	Mycoplasma	TPHA	FTA - Abs	BETA - HCG
☐ 0404200544 ☐ 2706200562	☐ 2706200562	☐ 0404200544		☑ 2706200562

Name : Sinan
Surname : Say
Bed Nr : 18011

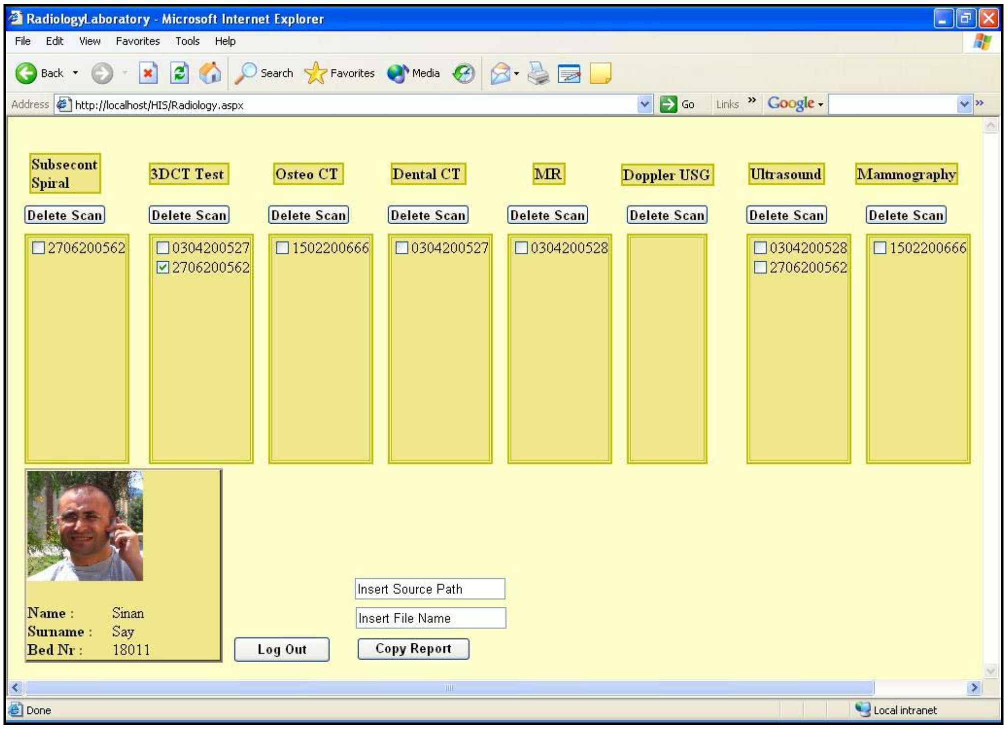
Figure 3.25 The ‘Laboratory’ page. Test that are made and patients assigned to tests.

After completing the test the results should be copied for a subsequent use. It is just a simple procedure to copy the test results. There are two textboxes. One has to insert the source path into the first textbox, to determine where the test result in the local computer is. Also into the next textbox, just below, one has to determine a name for the test result. After hitting the 'Copy Report' button the result will be copied into the patient's history results under a directory with the date. Now the physician can get from his / her own page the test results from the 'History Block' as explained before. The 'Delete Test' button deletes the patient checked, from the assigned test column.

To logout from the application one has just to hit the 'Logout Button', and he / she will be redirected to the hospital main page.

3.1.7 Radiology

Users who are members of the radiology staff are directed to this form after logging into the application from the main page. Each column, as can be seen in Figure 3.26, corresponds to a scan type. Each scan contains some numbers inside. These are the patient Id's and indicate that this scan is assigned to these patients. The radiology personnel should check the checkbox of the id to see the data of the patient. For a checked checkbox the name, surname, and bed number are shown in the left-hand side corner located data grid. Also the picture of the patient will be shown here.



Address: http://localhost/HIS/Radiology.aspx

Subsecont Spiral	3DCT Test	Osteo CT	Dental CT	MR	Doppler USG	Ultrasound	Mammography
□ 2706200562	□ 0304200527 ☑ 2706200562	□ 1502200666	□ 0304200527	□ 0304200528		□ 0304200528 □ 2706200562	□ 1502200666

Name : Sinan
Surname : Say
Bed Nr : 18011

Insert Source Path
Insert File Name

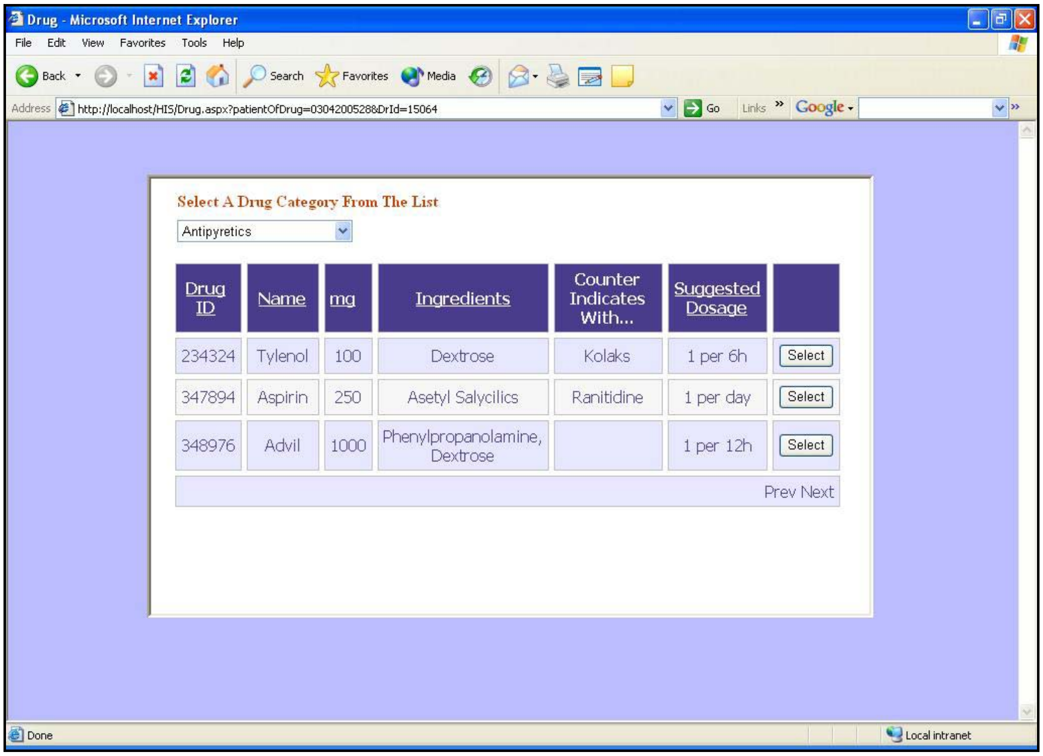
Figure 3.26 The ‘Radiology’ page. Scans that are made and patients assigned to them.

After completing the scan the results should be copied for a subsequent use. It is just a simple procedure to copy the scan results. There are two textboxes. One has to insert the source path into the first textbox, to determine where the scan result in the local computer is. Also into the next textbox, just below, one has to determine a name for the test result. After hitting the ‘Copy Report’ button the result will be copied into the patient’s history results under a directory with the date. Now the physician can get from his / her own page the scan results from the ‘History Block’ as explained before. The ‘Delete Scan’ button deletes the patient checked, from the assigned scan column.

To logout from the application one has just to hit the ‘Logout Button’, and he / she will be redirected to the hospital main page.

3.1.8 Drugs

As stated before a physician may assign drugs by opening from his / her page a drug assignment page. There he / she can click a button to get information about drugs available. This Button leads to the 'Drugs' form. The 'Drugs' form consists of a single block from which drug data can be obtained. As can be seen from Figure3.27 there is a Drugs dropdown list. From this list the physician can select one from various drug categories. For a selected category the drug data grid will be filled with available drugs information.



http://localhost/HIS/Drug.aspx?patientOfDrug=0304200528&DrId=15064

Select A Drug Category From The List

Antipyretics

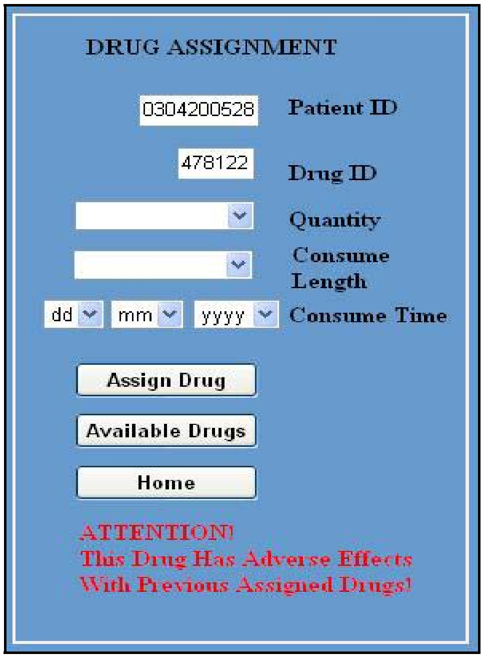
Drug ID	Name	mg	Ingredients	Counter Indicates With...	Suggested Dosage	
234324	Tylenol	100	Dextrose	Kolaks	1 per 6h	Select
347894	Aspirin	250	Asetyl Salycilics	Ranitidine	1 per day	Select
348976	Advil	1000	Phenylpropanolamine, Dextrose		1 per 12h	Select

Prev Next

Figure 3.27 The 'Drugs' page. All needed information available for various drug types.

The available information is: Drug Id, drug name, mg, ingredients, counter indications, and suggested dosage. From these information the physician chooses a suitable drug for his / her patients.

Selecting any drug by clicking the "Select" button sends the drug id to the assignment page. If any adverse effects with previously assigned drugs occur the physician will be warned with a warning label. (Figure 3.28) The physician has the right to continue to assign the drug anyway.



DRUG ASSIGNMENT

0304200528 Patient ID

478122 Drug ID

Quantity

Consume Length

dd mm yyyy Consume Time

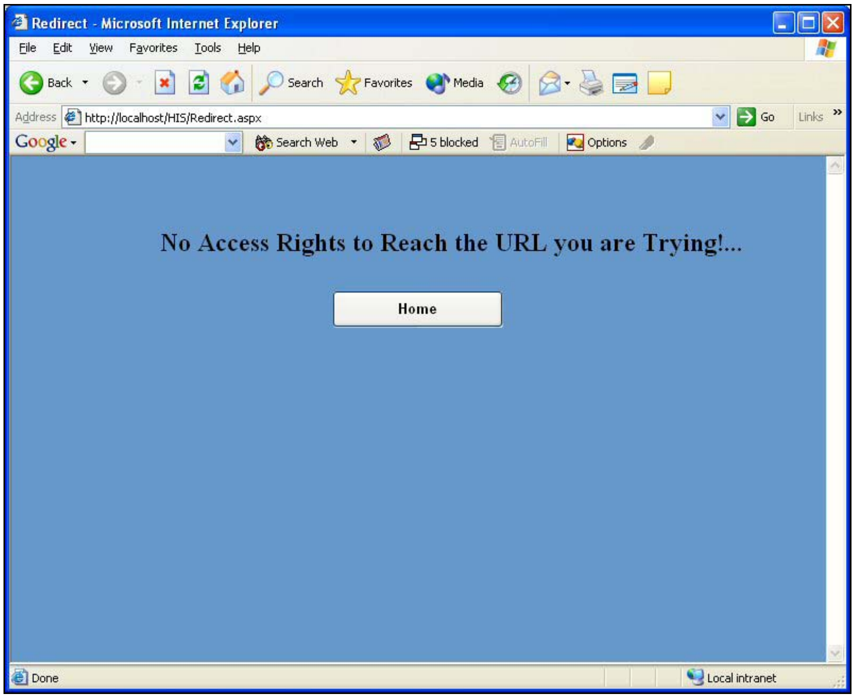
ATTENTION!
This Drug Has Adverse Effects
With Previous Assigned Drugs!

Figure 3.28 The ‘Drugs’ page. All needed information available for various drug types

For more than 5 drugs in a category the list tiles itself into pages and the user may search through more than one page for the suitable drug. If desired, the whole list can be sorted by drug name or by drug Id, just by clicking the title.

3.1.9 Redirect

This page is for unwanted access constructed. If an unauthorized user wants to enter the application without a legal password or just trying to guess the address (inserting random addresses into the browser address bar) this type of user will be redirected to this page. This page indicates that the user is unauthorized and has to be sign in. Reaching this page means also that the constructed session mechanism works and the entire system is secure.


Redirect - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites Media
Address http://localhost/HIS/Redirect.aspx
Go Links
Google Search Web 5 blocked AutoFill Options
Done
Local intranet

No Access Rights to Reach the URL you are Trying!...

Home

Figure 3.29 The 'Redirect' page

3.2 The E-R Model

As stated before the **Entity-relationship** model is a data model or diagram for high-level descriptions of conceptual data models, and it provides a graphical notation for representing such data models in the form of entity-relationship diagrams.

The Entity-Relationship is used as the basis in much database management, and allows one to describe entities and relations.

The HIS application database is modeled according the six E-R modeling steps.

1. Requirement Analysis: The needs of the HIS are discussed, and all requirements are extracted to get a whole knowledge of what the database should support.
2. Conceptual Database Design: In this step the determination process took place. The answers for the questions like " What will be the entities and relations? " " What information should be stored? ", " Can the model

represented pictorially? ", and " Are there any integrity constraints? " are given.

Considering the results from the requirement analysis led us to the answers.

In the hospital database there have to be 5 main entities: Physician (Doctor), Nurse, Patient, Drug, and Department.

There also have to be relations in the design to make entities interact among others. The relations to be used are determined as: Treatment, Consumes, and WorksIn.

The first 3 entities (doctor, nurse, patient) take actively part in the application, using all the relations. The entity patient is the data-collection-side of the system. Each collectible data like scanned images, laboratory results, or any other medical information is assigned to the patient. The entities physician and the nurse interact between each other by using the system to give the required care to the patients they are responsible. The physician evaluates the collected data of the patient and makes some orders that the nurse has to obey.

For the answer of what all the entities and relations must store should be as follows:

For each entity and relation a database table have been created, and primary keys are assigned. The primary keys are the key information that enables the user to search through each table.

Tables of entities and information they carry:

Patient

Table 3.1 Patient Table. Patient Id is the primary key.

Pname	psurname	Pid	dofb	admd	bedno	photo	notes
Patient name	Patient surname	Patient Id	Date of birth	Admission date	Bed number	Link of the photo	Any notes

Doctor

Table 3.2 Doctor Table. Doctor Id is the primary key.

doname	dosurname	doid	phone
Physician name	Physician surname	Physician Id	Physician phone number

Nurse

Table 3.3 Nurse Table. Nurse Id is the primary key.

nname	nsurname	nid
Nurse name	Nurse surname	Nurse Id

Drug

Table 3.4 Drug Table. Drug Id is the primary key.

did	dname	mg	ingredient1	cindic	dosage	type
Drug Id	Drug name	Weight	Ingredients	Counter indications	Suggested dosage	Drug type

Department

Table 3.5 Department Table. Department Id is the primary key.

dpname	dpid
Department name	Department Id

Tables of relations and information they carry:

Treatment

Table 3.6 Treatment Table. Patient Id, doctor Id, and nurse Id together form the primary key.

diag	pid	doid	nid	startd
Diangnosis	Patient Id	Physician Id	Nurse Id	Admission date

Consumes

Table 3.7 Consumes Table. Patient Id, and drug Id together form the primary key.

pid	did	consumetime	quantity	assigndate
Patient Id	Drug Id	Time to consume	Quantity to consume	Drug assignment date

WorksIn

Table 3.8 WorksIn Table. Nurse Id, doctor Id, and department Id form the primary key.

nid	doid	dpid	since
Nurse Id	Physician Id	Department Id	Working since…

Additional tables created, but not assigned to the ER-Model.

(They need no interaction)

Laboratory

Table 3.9 Laboratory Table. Patient Id is the primary key.

PID	bedno	VDRL	MycoPl	TPHA	FTA	BHCG
Patient Id	Bed Number	Type of test	Type of test	Type of test	Type of test	Type of test

Tests

Table 3.10 Tests Table. Patient Id is the primary key.

PatientID	BedNr	SSpiral	ThreeDCT	OsteoCT	DentalCT
Patient Id	Bed number	Type of scan	Type of scan	Type of scan	Type of scan

BoneDens	DopplerUSG	Ultrasound	Mammogr	Rontgen
Type of scan	Type of scan	Type of scan	Type of scan	Type of scan

PatientRelatives

Table 3.11 PatientRelatives Table. No primary key assigned.

Name	Surname	Password	Address	Phone	Email	Username
Name of relative	Surname of relative	Password of relative	Address of relative	Phone of relative	e-mail of relative	Username

Users

Table 3.12 Users Table. username is the primary key.

Name	surname	username	password	mail	Address	tel	userType	crCardNr	birth	SSNr
Name of user	Surname of user	Username	Password	e-mail of user	Address of user	Phone of user	User type	Credit Card Number of user	Birth date of user	Social Security Number of user

Each table has at least one column set as the primary key. Primary keys are the data at which queries jump from one table to the other. Relations have more than one column set as single primary key, because they are junctions of data flow. So it is prevented to store any excessive data twice. From the tables above each primary key is written in red.

As a whole, the entities and relations are determined and the information they will carry are obtained. These all should be collected together to represent the whole design pictorially. Figure 3.30 shows the ER-Model of the SMYRNA Hospital Information System.

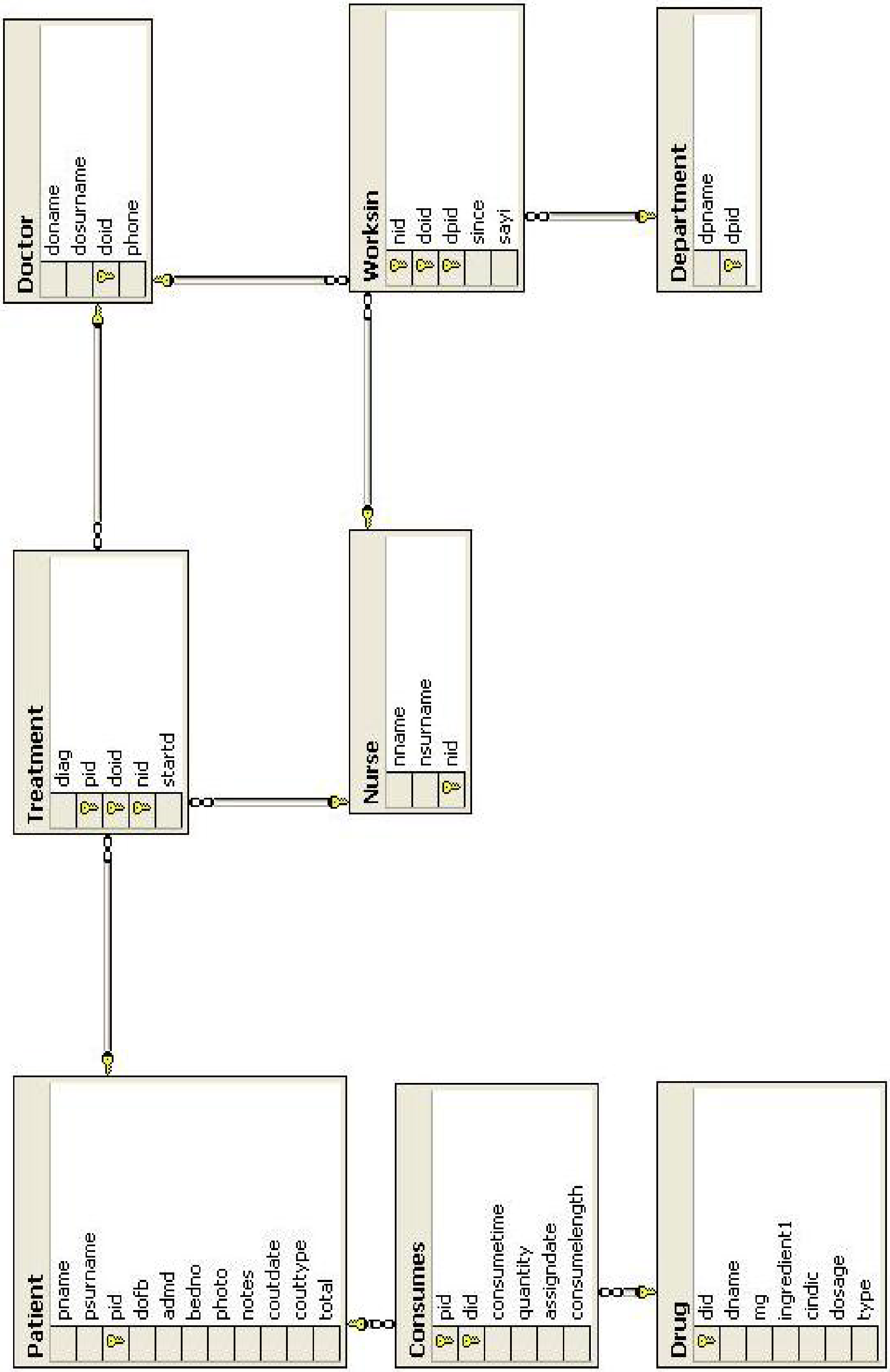


Figure 3.30 The Entity - Relationship model of the HIS.

3. Logical Database Design: To convert the constructed schema into the DBMS system the Structured Query Language (SQL) is used. For a patient search, a physician makes, a query searches through the entities and relations to get the information. For patient information the query uses the primary key of the physician, reaches the patient table through the treatment relation and gets all information from this table.
4. Schema Refinement: Any refinements are made to make the design as simple as possible. Therefore some tables are extracted from the model to prevent data excess.
5. Physical Database Design: The data is stored in a SQL Server 2003 database as tables. Any other data like images and other raw material is stored in a file system that will be discussed in the next section.
6. Application and Security Design: For security reasons the access to the database and to the application is limited. The database can only be reached by queries and not all users have the write access, except a limited write access for physicians, admission personnel and administrators.

3.3 The Patient Folder System

Beneath the database there is also a patient folder system integrated into the system. This folder tree is for storing image like patient information such as laboratory test reports, scanned images, and patient photo.

Figure 3.31 shows the file tree structure for a single patient.

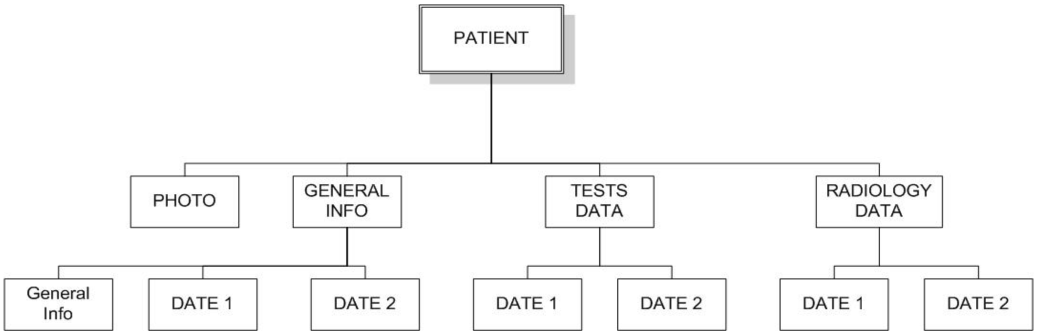


Figure 3.31 File tree structure of the HIS.

Under a previously defined storage device like a generic server this file structure can be stored. For explanation purposes the file tree will be shown to be stored under the path C:\HospitalData. Each time an admission is completed a new folder will be created under the HospitalData folder. The name of the new folder will be the PatientID given to the patient. Say the patient Id is 0302200566, then a folder named 0302200566 will be created under C:\HospitalData.

As from Figure 3.31 can be seen sub folders

Foto

GeneralInfo

TestsData

RadiologyData

will be created initially under C:\HospitalData\0302200566. The info file GeneralInfo.txt will then also created under the path C:\HospitalData\0302200566\GeneralInfo.

The ***Foto*** folder contains just the photo of the patient. Any scanned photo of the patient will be put under this folder manually by the administrator, or if desired by the admission personnel.

The ***GeneralInfo*** folder is the place where, as its name implies, general info about the patient is stored. This folder is just used by admission procedures and does not take part of any SQL query search within the application. The GeneralInfo.txt file stores just name, surname, patient Id, and date of birth information. This folder may also contain many sub folders. Subfolders inside this folder are history folders. e.g. if patient with id 0302200566 is a former patient and admitted twice before, say on dates 02.01.2003 and 21.03.2004 then the C:\HospitalData\0302200566\GeneralInfo folder will contain two subfolders 02012003 and 21032004. Inside these folders the information physician Id, nurse Id, diagnosis, and notes will be stored. A history search for patients is enabled, owing to these folders and files. One can see that if the patient was previously in the hospital, who was his / her physician/nurse and what diagnosis was made.

The ***TestsData*** folder is created during admission automatically, but will be filled after a test is assigned to the patient. The test results will automatically be copied under this folder. Say a test is made on 03.11.2002 then a subfolder named 03112002 will be created and the test results will be copied under C:\HospitalData\0302200566\TestsData\03112002\. If a physician lets the patient with Id 0302200566 be shown on his page, the history dropdown list will be filled indicating that there is a test result on date 03.11.2002. The physician may get the test reports and may copy them to his / her local folders.

The ***RadiologyData*** folder is created during admission automatically, but will be filled after a scan is assigned to the patient. The scan results will automatically be copied under this folder. Say a test is made on 04.11.2002 then a subfolder named 04112002 will be created and the test results will be copied under C:\HospitalData\0302200566\TestsData\04112002\. If a physician lets the patient with Id 0302200566 be shown on his page, the history dropdown list will be filled indicating that there is a scanned image on date 04.11.2002. The physician may get the scanned images and may copy them to his / her local folders.

CHAPTER FOUR
CONCLUSION

4.1 Discussion

A Hospital Information System is a combination of computer hardware and software with the necessary communications that aid the management of a hospital's clinical and administrative information. It is of high importance that the software must have the right type of modularity so that it is manageable, maintainable, and upgradeable. The hardware should also be reliable, available, and have the necessary performance capacity. This work presents a www-based Hospital Information System, which is assumed to provide the needs given in the definition. The architecture and a brief explanation of the application throughout show details that support the assumption.

On the hardware side the application may be run on any server specified and equipped computer irrespective of whether the target application is chosen for a large scale or a smaller hospital. On the software side, the SMYRNA Hospital Information System has a web page-based design architecture. It accepts neither a modular nor an all-in-one portal structure. Everything is made in a web design manner, so the modules are replaced with simple web forms. The web-based architecture ensures the easiness of implementing new features and scalability. This feature has the advantage to implement both the graphical user interface (GUI) and the code in parallel. In a modular case, the module was developed and the GUI was written on top of the module it belongs to. In our case, the code was written as each web form was designed and created, owing to the visual coding platform used. So there is no need to add module code and afterwards the GUI code, i.e. to spend extra time for both.

The use of the ASP.NET platform makes the application reliable, owing to the client-server architecture. Web based information is implemented by use of the simple Hypertext Markup Language (HTML). This makes the system as simple as possible; however, the operation behind can be as complex as possible. Accessing the application through a simple web browser makes it user friendly. Taking a look to recent applications in literature, one can see the tendency for being reachable worldwide. Most hospital information systems have been closed to the world for security and other reasons. But recently we notice that some recent ones open their doors to the world. So does the SMYRNA HIS. While the SMYRNA HIS is based on a web page architecture, it is reachable from any browser-installed-computer throughout the world. No additional software has to be installed. Any physician has the comfort to work from home or any other location outside the hospital, in contrast to other known applications.

Being reachable worldwide brings security questions into minds. In previous applications each user was assigned a security level for accessing data within the application. One could enter the application with this security level and read/write data this security level allows. The SMYRNA HIS uses a user type security system. Types of users are defined and the web forms are constructed for each of these types. So any user can reach only the web pages that its user type allows. No other data or page can be reachable. Any guess of other webpage addresses within the application is prevented through sessions.

Also in other applications, especially in portal like applications, each physician has the access right for patients (whether assigned or not) and its data (laboratory, radiology, drugs etc.). In our system if a patient enters the system, he/she is assigned to the related physician and/or nurse. Only the assigned nurse and physician have the rights to reach and change the data for this patient. If necessary, the physician shares the information with other physicians for consulting purposes. There is also no need to construct user oriented security levels.

The application in question also benefits from its connection type. When the application changes its state and requests for another page or some information, it connects to the web/database server. It creates a temporary connection, retrieves the information, and closes the connection. This short-term connection lowers the network traffic that results in more connection speed and higher available bandwidth. No long term or permanent connections are allowed.

The capability of using distributed databases makes it usable widespread. The database of the Hospital Information System may be set in any place. Only the connection string should be changed to make the information available for the application. Also the file system containing the image history may be transported to any place. This feature makes the application portable and adjustable without any difficulty.

One main difference to other HIS applications is that the SMYRNA HIS focuses on the patient treatment implementing an interaction between physician and nurse. Any change into patients' drug chart made by the physician can be seen on the nurse side and vice versa.

Even the patient's relatives could take part in the interaction system. Any patient relative has the ability to search the patient and find the place where he/she stays, a feature not commonly seen in other similar applications. This feature reduces the waiting time and workload in the hospitals.

The web application side of the HIS is also portable and very easy to construct on any other location. One who wants to make the application run on another hospital site has just to copy a single folder (and its underlying subfolders) to the other location where he/she desires.

Using the benefits of the development platform the HIS in question is scalable at any time. New pages and new features may be added without disturbing the entire system. It is as simple as designing a web page.

As can be seen from all these positive features, this application has implemented many advantageous benefits making it a cost-effective, user-friendly, and powerful tool. It is expected to meet the needs of the medical informatics.

4.2 Future Work

During development the main focus of the SMYRNA HIS project was the interaction between patient, physician, and the nurse. Therefore the application was built to satisfy the needs a patient, its relatives, its physician, its nurse, and administration personnel need. This hospital information system may need also to be improved by adding new features, as any other application does. Therefore, new web pages may be implemented as needed to integrate new features. Also some other features could be added to satisfy the needs that will appear during usage. It should be noted that adding new features will be easy, thanks to the web page architecture and the scalability feature.

REFERENCES

Birim HBYS, Retrieved May, 2005, from http://www.birim.com.tr/

Care2X Integrated Healthcare Environment, Retrieved May, 2005, from http://www.care2x.org/

Chae, Y. M. Ph.D. *Hospital Information Systems*. Graduate School of Public Health, Yonsei University, Seoul, Korea

Common Language Infrastructure. Retrieved March 03, 2005, from http://en.wikipedia.org/wiki/Common_Language_Infrastructure

Common Language Runtime. Retrieved March 07, 2005, from http://en.wikipedia.org/wiki/Common_Language_Runtime

Date, C. J. (1986). *An Introduction to Database Systems.* Volume I, Fourth Edition Addison-Wesley Publishing Company

In the fragmented European hospital information systems market, established vendors have a distinct advantage. (18 Aug 2004) Retrieved Mai, 2005, from http://www.news-medical.net/?id=4139

IT & Communications, Integrated Hospital and Data Systems. *Son Llatzer's integrated data systems project creates the e-hospital.* Retrieved Nov. 25, 2005, from http://www.medinfo.netbib.de/images/SonLlatzer.pdf

Lærum, H., Karlsen, T. H. & Faxvaag , A. *Use of and attitudes to a hospital information system by medical secretaries, nurses and physicians deprived of the paper-based medical record.*

MacDonald , M. (2002). *The Complete Reference: ASP.NET.* The McGraw Hill Companies.

Microsoft .NET. Retrieved March 03, 2005, from http://en.wikipedia.org/wiki/Microsoft.net

Prokosch, H.-U. & Dudeck , J. (1995). *Hospital Information Systems: Design and Development Characteristics; Impact and Future Architecture.* Elsevier Science B.V., 1995

Ramakrishnan, R. & Gehrke, J. *Database Management Systems* Third Edition. The McGraw Hill Companies.

VisualStudio.NET. Retrieved March 07, 2005, from http://en.wikipedia.org/wiki/Visual_Studio_. NET

Winterı, A., Briglı, B. & Wendtı, T. (). *Modeling Hospital Information Systems (Part1): The Revised Three-Layer Graph-Based Meta Model 3LGM2.* 1Institute for Medical Informatics, Statistics and Epidemiology, University of Leipzig, Leipzig, Germany