Web Based Distance Learning for Power System Engineering

by

Rainer Helmut Lehr

Supervisor: Professor A. Petroianu

Thesis submitted to

The faculty of Engineering of the University of Cape Town in fulfilment of the requirements for the degree of Masters of Science in Electrical Engineering

September 2000

Acknowledgment

The author takes this opportunity to thank his supervisor Professor Alexander Petroianu for providing the possibility, environment and suggestions for the thesis.

Also thanks to Eskom for their financial support .

My gratitude goes to Dr. Remus Fetea for his invaluable help, suggestions and his dedication for organising the project. Thanks to Sven Willmann for all the helpful discussions and comments about the project.

Special thanks goes to my housemates Helena Broadbridge and Jennifer Wallace for their proof reading and for being such helpful and understanding friends.

I would like to thank my parents for their love and their support through all those years.

Thanks to the mountains, the sea and the sun for keeping me sane.

And finally thanks to all my friends who make life worth living.

Synopsis

This thesis presents the results of research work carried out to develop a Power Engineering Education platform for Web Based Distance Learning at the University of Cape Town. It also reviews the literature on the subject of Web Based Distance Learning and introduces a system that is designed and implemented by the author.

The potential of the World Wide Web (WWW) with the capabilities of animation, audio, chat, graphics, and video makes active learning in distant learning environments possible. Most course-based or learning sites simply post course materials. The asynchronous nature of the WWW allows flexible scheduling for both learner and instructor. When and where to access course material and other resources only depends on one's personal needs.

The concept of a virtual classroom meets these demands. It creates an environment, which enables students to interact with the instructor on an individual basis, and to create discussion groups or learning groups with fellow students. A virtual classroom contains at least the following features:

- Chat-Room
- Discussion Group
- Bulletin Board / News Group
- Email
- Virtual Library

All these features are a useful and a desirable part of any Web Based Distance Learning course. The scope of a course can lead to a design that focuses more on certain features than on others.

New tools and techniques for delivering information on the WWW become very popular and disappear again in a rapid speed. Only a few developments become an industrial standard. The decision of what tools and techniques to use depends mostly on the course requirements, financial constraints, infrastructure and a big part of personal preference of the designer. A good recommendation is: design as sophisticated as necessary by keeping it as simple as possible. Utilise the power of the WWW and all its technical possibilities to create a course that meets all the requirements. Use widely accepted techniques and standards to ensure a problem free and easy maintainable lifetime of the system.

Chapter 5 describes two well-known example sites and concludes that both sites are not utilising the full power of the WWW and the Internet. Including a feature because it is available (like the Chat-Room for example) does not automatically create a virtual community. To provide an incentive to use the features is as necessary as the feature itself. Putting students and a teacher together into a classroom does not result into a learning effort. If all of the participants sit idle at their desks, there will not be a successful learning outcome. These sites are well received in the Internet community but more likely because of the quality of their contents than for their learning environment.

The chapters 6 and 7 introduce the author's design and implementation of a Web Based Distance Learning environment. It focuses on instructing than on just presenting. Communication between students and instructors and between students themselves is the main idea behind this approach. The system is not limited to communication but offers personal guidance through an online tutorial where the instructor tutors online while the students study the course notes or doing exercises. More personal guidance is possible through the dynamically created customised Web Pages. Each student receives a personal learning environment, customised to the student's individual needs and preferences. The instructor watches the student's progress and can individually interfere if help and more guidance are needed.

A Web Based Distance Learning course overcomes the disadvantages of conventional distance learning methods by offering faster interaction, multimedia and by creating a real learning community. It reduces the cost of learning by being independent of location and time. In the case of further education, the costs for travelling, accommodation and the absence from work are saved and the asynchronous nature of a Distance Learning course accommodates the individual time constraints.

The, by the author developed and implemented system meets all requirements for creating an effective learning environment. The essence of the WWW is communication and information exchange and therefore a perfect environment for distance learning.

Table of Contents

Table of Figures

Glossary of terms

ASCII	**A**merican **S**tandard **C**ode for **I**nformation **I**nterchange
DB	A **database** is a collection of data that is organised so that its contents can easily be accessed, managed, and updated.
CGI	**C**ommon **G**ateway **I**nterface
Extranet	An extranet is a private network that uses the Internet protocol and the public telecommunication system to securely share part of a business's information or operations with suppliers, vendors, partners, customers, or other businesses.
GUI	**G**raphical **U**ser **I**nterface
Host	On the Internet, the term "host" means any computer that has full two-way access to other computers on the Internet.
HTML	HTML (**H**ypertext **M**arkup **L**anguage) is the set of "markup" symbols or codes inserted in a file intended for display on a World Wide Web Browser.
HTTP	The **H**ypertext **T**ransfer **P**rotocol (HTTP) is the set of rules for exchanging files (text, graphic images, sound, video, and other multimedia files) on the World Wide Web.
Hypertext	Hypertext is the organization of information units into connected associations that a user can choose to make.
Intranet	An intranet is a private network that is contained within an enterprise.

Java	Java is a programming language specially designed for use in the distributed environment of the Internet. It was introduced by Sun Microsystems in 1995 and instantly created a new sense of the interactive possibilities of the Web. Both of the major Web Browsers include a Java virtual machine.
LAN	A **L**ocal **A**rea **N**etwork (LAN) is a group of computers and associated devices that share a common communications line and typically share the resources of a single processor or server within a small geographic area (for example, within an office building).
Link	A link is a selectable connection from one word, picture, or information object to another.
Mpsec	**M**ultimedia **P**ower **S**ystems **E**ngineering **C**ourse
Petri-Net	Petri-Nets are used to model parallel technical processes and their interaction.
Plug-In	Plug-in applications are programs that can easily be installed and used as part of your Web Browser.
Servlet	A Servlet is a small java program that runs on a Web Server.
TCP/IP	TCP/IP (**T**ransmission **C**ontrol **P**rotocol/**I**nternet **P**rotocol) is the basic communication language or protocol of the Internet.
WBI	**W**eb **B**ased **I**nstruction, using the WWW as a medium for instruction.
WWW	**W**orld **W**ide **W**eb. A technical definition of the World Wide Web is: all the resources and users on the Internet that are using the HTTP.

1 Introduction

This thesis investigates the possibility of a Power Engineering Education platform for Web Based Distance Learning at the University of Cape Town. Distance learning consists of instruction through print or electronic communications media to persons engaged in learning in a place or time different from that of the instructor(s) or other students (Moore, Cookson, & Donaldson, 1990). The potential of the World Wide Web (WWW) with the capabilities of animation, audio, chat, graphics, and video makes active learning in distant learning environments possible. Most course-based sites or learning sites simply post course materials. Use of the WWW as merely an "electronic book" falls far short of the potential for moving instruction away from a repository model to one where active learning can occur (Filipczak, 1995; Howard-Vital, 1995; Shotsberger, 1996). The asynchronous nature of the WWW allows flexible scheduling for both learner and instructor. When and where to access course material and other resources only depends on one's personal needs. A concern often associated with distance learning is the lack of learner interaction. A common complaint mentioned by learners in distance learning environments is that they feel isolated and unconnected. The concept of a virtual classroom meets these demands. It creates an environment, which enables students to interact with the instructor on an individual basis, and to create discussion groups or learning groups with fellow students.

The technical possibilities and the consequential advantages of the WWW as a medium for a distance-learning environment are presented and discussed in this thesis. The research task was to design, develop, implement and test a platform for Web Based Distance Learning. The thesis does not focus on a course content itself and on what the didactical possibilities of presenting a course on the WWW are.

The author, with reference to the existing literature and public domain programs, decided to develop and write his own programs in Java. This was motivated by the

need to gain a deeper understanding of the Internet communication algorithms, network components and their limitations. Instead of using standard software components and inserting the components like black boxes into the system, the developed programs were specially customised to the specific need of the system design.

After the introduction, this thesis starts by presenting a background on distance learning, the Internet, and the Internet application known as the World Wide Web in chapter two.

The third chapter focuses on the design of Web Based instructions and the components to support the learning on the WWW.

The fourth chapter discusses the technical possibilities of delivering Web Based Instructions.

Chapter five describes two other Web Based Instruction sites and their approach of delivering instructions over the Internet.

The chapters six and seven explain the design and the implementation of the developed system. Differences between the developed system and other Web Based Instruction sites, as well as its advantages are discussed.

The last chapter presents the authors conclusions and recommendations for further development.

2 Web Based Distance Learning

2.1 Distance Learning

Traditionally we define learning as a process where one or more instructors are trying to mediate knowledge to one or more students in a classroom-like environment with the aim of increasing the student's knowledge. Distance learning takes place when the instructors and students who are involved in the learning process are physically or time-wise separated from each other. The means of communication range from printed correspondence via television, to the World Wide Web. There are various reasons for engaging in distance learning: constraints of time, distance and finances to name only a few examples. Mostly we think of higher education as distance learning programmes, like a university degree via UNISA (University of South Africa) or the Fern Universitaet Hagen in Germany. But today, with technology rapidly advancing, for most professions a constant, continuous further education and training is a permanent issue.

For both employer and employee, it is very important to be up to date with the newest development in their field of expertise. It is very cost intensive to send employees regularly on courses, the costs of travel, accommodation and the absence from work can be quiet substantial. Distance learning offers an attractive alternative. The asynchronous nature of a distance-learning environment is predestined for this cause. Since the student is able to study whenever it is convenient for him and when his time schedule allows it, the employee is not missing his work and doesn't have to be separated from his accustomed environment and family. In the case of employer-approved studies, studying could even occur at the workplace during or after working hours and utilize the available resources. Distance learning can make a strong contribution to corporate training efforts, especially when using some of the more advanced technology (Clyatt, 1998).

Distance learning raises a range of pedagogical, technological and organizational issues. Certainly these issues have to be addressed quite differently from a "traditional" learning environment. For example, a teacher standing in a classroom is instantly confronted with the response of the students. For example, blank faces after an explanation, students entertaining themselves with conversation or materials not relevant to the course, questions raised immediately, or even if students do not participate at all by not showing up in the classroom. How does an instructor keep the attention of the students or receive feedback if the students are scattered all around the world?

2.2 The Internet and the World Wide Web

The Internet evolved out of the ARPANET of the US Defence Department. In the early 1960 the Advanced Research Project Agency (ARPA) started a small network of computers amongst researchers in the USA. The Universities were the first institutions to utilise the Internet by connecting to each other with the purpose of file transfer and accessing super computers remotely. Email quickly became the most used application on the Internet. In the 1970's the ARPANET slowly moved away from its military roots and grew internationally with its first links to London and Norway.

Today, the Internet is a public, co-operative, and self-sustaining facility accessible to hundreds of millions of people worldwide. Physically, the Internet uses a portion of the total resources of the currently existing public telecommunication networks. Technically, what distinguishes the Internet is its use of a set of protocols called TCP/IP (Transmission Control Protocol/Internet Protocol). Two recent adaptations of Internet technology, the intranet and the extranet, also make use of the TCP/IP protocol. TCP/IP is the basic communication language or protocol of the Internet. When a computer is set up with direct access to the Internet, the computer is provided with a copy of the TCP/IP protocol.

In the mid-80s the availability of personal computers and powerful network-ready servers did allow corporations to join the Internet for communication purposes. In 1991 Tim Berners-Lee, working at CERN in Switzerland, posted the first computer code of the World Wide Web (WWW) in a relatively innocuous newsgroup, "alt.hypertext."

The ability to combine words, pictures, and sounds on Web Pages excited many computer programmers who saw the potential for publishing information on the Internet in a way that can be as easy as using a word processor. The WWW is in principle an "application" using the Internet as its medium. The idea behind it was that most available information was somehow processed or stored on computers somewhere in the world. Linking that information together through a hypertext system, that spans the globe and therefore makes the information accessible to anyone, anywhere, at any time. The power of the WWW is that is not restricted to a certain format. It links any kind of resources in the universe of information, text, images, sound and videos....

The WWW is a hypertext system and uses the Hypertext Transfer Protocol (HTTP) to communicate. In simple words, a client-software (mostly browser) connects to a server and requests a file that the server (if available) sends back to the client. The client software then saves, displays or executes the file. In general the User Interface for the WWW is a HTML document (Hypertext Markup Language) that is displayed in a browser and the document links to other resources. For example, if a page contains pictures and sounds, then these are not included in the document itself. Only links to independent files are included in the document. These files are then separately downloaded. Figure 1 shows the recursive structure of such a hypertext system.

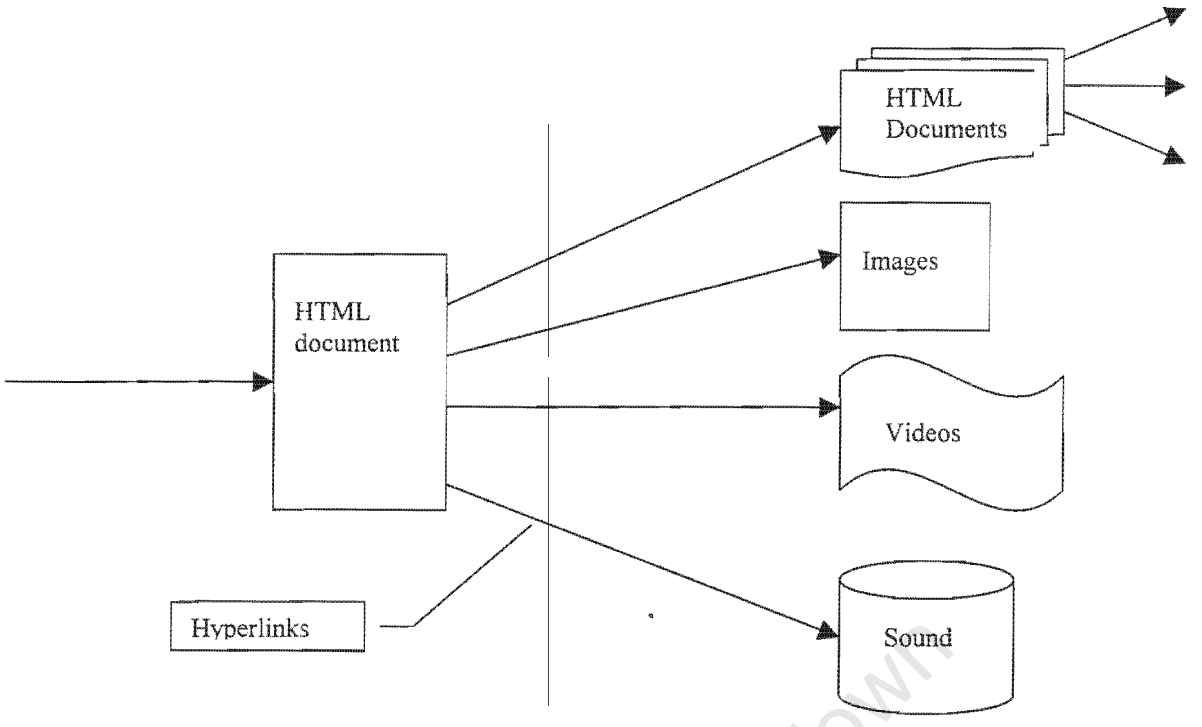


Figure 1: Hypertext Structure

The WWW is growing daily; millions of people are using it for communication, research and information retrieval. Today the WWW comprises the bulk of traffic over the Internet. New standards and technologies emerge at a rapid speed that make the WWW one of the most promising sectors in information technology.

2.3 Web Based Distance Learning

The location, the number and the technological infrastructure available to participating students are major considerations in the decision whether or not to provide a Web based course. If the students are in the proximity of the instructor and the resources, there is no real need for a Web based course. Obviously there is also no need for a Web based course if the students don't have access to the

Internet at all. But what if the technological infrastructure is limited e.g. connection speed or the cost for an Internet service provider? The question is how to design a Web based course that meets these requirements, with an implementation that is still useful? Just because the Internet is very popular doesn't mean it is the solution to all problems. The WWW offers great potential as a distance-learning environment; it has all the advantages of the most common distance learning environments like mail or television/radio and offers the potential to overcome their disadvantages. The Web offers information from anywhere in the world at any point of time to anyone interested. The only constraint: access to the Internet. Television or Radio still have to be scheduled and can only be followed at certain times and instruction through mail is slow and has very limited features. The WWW doesn't have a schedule and is instantaneous. Offers data and multimedia (like graphics and video) as well as a personal asynchronous communication with the instructor via Email. The student is able to study at a convenient time and to ask the instructors for immediate assistance, engage the instructor in a discussion about a topic, or submit assignments directly over the WWW. On the other side, the instructor has similar advantages - no student is disturbing him while he is occupied with work, but he is addressing each student individually.

The WWW also offers the opportunity of a direct, face-to-face communication via Chat-rooms or video conferencing. One complaint often voiced by learners in distance learning environments is that they feel isolated and unconnected (Hill, 1997). With the WWW, creating a learning community where students communicate with other students and not only the instructor, and where students work in teams, is possible. The essence of the WWW is communication and information exchange. A perfect environment for distance-learning.

3 Designing Web Based Instructions

Instruction is the deliberate organisation and presentation of information with the end goal of promoting specific learning (T.F. McManus, 1996). "The questions that do need to be asked relate to making sure distance education efforts meet the course objectives and the students needs" (Schifter, 1999). Using the Internet as a medium for instruction delivery is a widespread idea. HTML provides an efficient way to present instructions on the Web. Publishing course notes as simple copies of books or lecture notes is an easy realised and simple approach, but has little to do with instructing through the Web. In a conventional lecture theatre, the lecturer gives guidance and keeps distractions away. Whereas on the Internet, the students are confronted with a whole range of distractions and a lack of guidance and this has to be compensated by a good instructional design. Consider that it is far easier to change the visiting Web Site or simply switch off the computer than walking out of a lecture theatre.

The goal must be to present the information without overloading the student, using multimedia for instruction where it is appropriate, without boring the student with downloading time and distraction by overusing the functionality of the Web (blinking, moving and flashing...). Web designers, who are new to the medium WWW, are often amazed by the possibilities and features available for designing a page, and overload the page with every possible feature.

Where hyperlinks lead to related topics or information inside and outside their own system and expand the instructions almost infinitely, too many hyperlinks can cause irritation by criss-crossing the pages and losing track of the learning goal.
Everyone who frequently uses the WWW experiences the situation when accessing the WWW to find specific information, but ending up spending hours following links to other interesting topics not related to the original aim.

By creating Web Based Instructions, the designer has to find a balance between keeping the user attracted to the system and overloading the users perception. In this chapter methods are introduced that help keeping this balance and that show that a Web Based Distance Learning system has advantages over conventional Distance Learning methods.

3.1 Creating Virtual Learning Communities via the Web

Just sitting in a classroom inevitably creates a community, however temporary. You're sharing the same room at the same time with people who have the same goal - learning a specific topic. Interaction between students in such a community ranges from discussing the topic presented by the instructor, to sharing results of exercises, to simple small talk to distract and refresh the mind. Many students are more willing to ask fellow students for advice, help or their opinion, as they would be to ask the instructor. Reasons for that could be to save the embarrassment of asking a potential trivial question in public, or to feel more easily understood by their fellow students. On the other side the instructor is able to survey the class with one look and recognise the bored student, who is playing games on the one side and the student with the puzzled look of the totally lost person on the other. A learning community is essential for successful learning. Creating such an environment on the WWW, where students and instructor communicate with each other moves instructions on the WWW away from simple reading matter towards interacting in a virtual classroom.

A virtual classroom creates an environment where online resources are used to facilitate collaborative learning between students and instructors, between a class and a wider academic and non-academic community (Hiltz, 1995).

The following features are essential for a virtual classroom:

- Email environment within the Virtual Classroom
- Bulletin Board / Discussion Group
- Chat-Room

Through Email the students are able to interact with other learners. It enables students to discuss problems, work on assignments or projects together. Most commonly, the student and instructor also interact via Email. A virtual office replaces the office hours where students could come and ask question. The asynchronous nature of Email enables independence between student and instructor and still ensures that interaction takes place. For the instructor, Email is a way of providing guidance and getting feedback in a distance-learning environment. The Email environment could include mailing lists and address books, to offering a Web based Email account like Hotmail (www.hotmail.com), organised for easy access for specific groups and classes within the virtual classroom.

Where Email helps individuals (student-student / student-instructor) to interact, a Bulletin Board enables communication among all course participants. Questions regarding course material, assignments and course announcements are easily placed and accessed on the bulletin board.

An online discussion forum is an ideal place to enhance student inclusion and participation in the course. A discussion forum is a collection of notes, messages, questions or answers published permanently (depending on the administration policy) and can be read at anytime by anyone. Such discussion could go on forever or for the duration of the course. Any question that might arise while studying the course can be published there, or already found there, because other students came across the same problem. Anyone (including the instructor) can place an answer (right or wrong), or a further question and solve or deepen the discussion.

This can save time for the more advanced student because the student does not have to listen to arguments from less advanced students. On the other hand, these students could get involved in discussions that take them further than needed or expected by the instructor. The instructor could also utilise the discussion forum in his course by placing certain questions and then watch the progress of the discussion, give hints and lead it where the instructor wants it. Or even link the course notes to interesting discussions or points made in the forum.

A Chat-Room works in principle like the discussion forum, just online and instantly. Only people participating at a specific time take part in the discussion and can utilise everything said there. It is a convenient tool to get advice from a fellow student or the instructor; the question is typed in and can be read and answered (if the answer is known) instantly. A disadvantage is that if none of the participants know the answer, the question will be lost and may never be solved. If the question is important then it should be placed in the discussion forum so that everyone has the chance of utilising it. The Chat-Room can also act as a virtual office, for example a student makes an appointment with the instructor to meet in the Chat-Room to have an online conversation.

3.2 Incorporating Interactivity and Multimedia into Web Based Instruction

Since the Internet is a highly acclaimed multimedia medium it seems natural to incorporate all available features into an instructional system. Many Web Sites feature all possible multimedia accessories, but this makes the site very slow to download and difficult to orientate in it. The opportunities arising through incorporating different media like text, graphics, video, sound and animation into one system are significant. For example, enhancing a text with a video to deepen the understanding or a recorded lecture of an expert in the field. Or instead of just

looking at a static graph explaining the functionality of, a motor for instance, an animation let the parts move. But it is very important to plan the use of multimedia features carefully, placing multimedia only where it is useful and giving the user the choice of displaying it. For example, a situation where a user who visits a page several times and every time has to wait until a introductory video is downloaded before the user can continue with the purpose of the visit, is more annoying and discouraging than helpful.

The Internet offers student interactivity far beyond other distance learning media where it is mostly restricted to correspondence. Besides student interaction mentioned above with the virtual classroom, it offers the possibility of online exercises, examination, tutorial and even the use of a virtual laboratory. The online exercises as server or client side programs could be in the form of multiple-choice questions or in the form that the student has to enter an answer in easy form (e.g. single words or numbers) and the programs then evaluates the answer automatically and provides either the right answer or the program provides help and hints to get the student onto the right track.

Examinations work along the same principle, except that normally no help and guidance is provided and a time restriction is given. In a virtual laboratory the student is able to access machines or simulation software remotely. This provides the biggest form of interactivity. The student has the opportunity to conduct practical experiments on real experimental facilities at a remote location. Software that is not available to the student (e.g. too expensive) can be accessed remotely by sending the input data to the remote server and receiving the results after the remote server processed the data.

3.3 Virtual Library

The WWW is a huge collection of knowledge; digital versions of books, research papers, and articles are available. Through the cyberspace [Cyberspace is the total

4.1.2 Graphics / Video / Sound

Even simple created HTML documents can contain links to images, graphs in numerous formats that are displayed within the text. All standard browsers are able to display numerous standard file formats including JPEG (Joint Photographic Experts Group) and GIF (Graphic Interface Format). Unusual formats need a so-called Plug-in to be displayed. The same applies to sound, movie or video clips like MPEG (Motion Picture Expert Group) or QuickTime Files (from Apple). It is recommended to refer to the standard formats where it is possible because the downloading and installation of Plug-Ins require time.

4.1.3 Simulations

The use of Java and other software makes it possible to create animations and simulations that are executed on the client side. On the other hand, expensive and powerful simulation software can be made available to the students through remote access over the Internet. There is no doubt that the experience with simulations can greatly enhance the understanding of certain areas

interconnectedness of human beings through computers and telecommunication without regard to physical geography (Whatis.com, 2000)] structure of the WWW, the information incorporated into a course could theoretically grow indefinitely. A conventional textbook offers only as much information as the author actually put into it. In the WWW the information available on a Web Page can be extended with a simple link to include the information on the linked Web Page, and this recursively.

There is a myth that everything is available on the WWW and that everything that is available is correct and useful. The selection of Web Pages to be included into a course needs to be done very carefully. Since the WWW is a free medium and everyone can publish anything without any sort of control, it is necessary to be cautious with any contents on the WWW. Usually a conventional learning environment includes a library where students can chose material (from an already selected range) to clarify and deepen their understanding. Searching the Internet for relevant information is a very time consuming and sometimes very frustrating task. Including a virtual library into the Web Based Distance Learning environment where the information is already filtered and easy to access enhances the efficiency of the learning environment. Students are able to access resources easily and at the same time they can be assured about the value of the contents.

3.4 Conclusion

All the features introduced above, are useful and a desirable part of any Web Based Distance Learning course. The scope of a course can lead to a design that focuses more on certain features then on others. For example, a certain course needs a great amount of communication between students and the instructor but there is not much additional material available. Therefore the virtual library might not be useful to be implemented. Chapter 6 (Design and Implementation) describes the author's design and implementation of a Web Based Distance Learning course and the motivation behind the design and the implemented features.

4 Delivering Web Based Instructions

The questions: "How do I read the information on the WWW?" or "How do I make the information available on the Internet?" seem trivial. Taking one of the common, freely available browsers and making sure to be up to date with the development is a sufficient option on the client side. To present information is much more than just typing it with an editor and storing it on a Web Server. This approach might work for simple home pages for personal use. A bigger system needs more considerations to have a reliable, scalable system that can sufficiently cope with the client and server side demands. Some simple questions like "what material do I publish?" and "how many users do I expect?" lead ultimately to more questions like, "can my internet connection carry the expected traffic?", or "is my server breaking down under the workload?".

This chapter introduces tools and the technical possibilities of how to deliver Web Based Instructions efficiently and successfully.

4.1 The Material

The material to be presented on the WWW is usually a combination of text and pictures/graphs combined with sound and video.

4.1.1 HTML

HTML (Hypertext Markup Language) is the lingua franca for publishing hypertext on the World Wide Web (W3C, 2000). It is a non-proprietary format based upon SGML and can be created and processed by a wide range of tools, from simple plain text editors to sophisticated authoring tools. HTML is the set of "markup" symbols or codes inserted in a file intended for display on a World Wide Web Browser. The Markup tells the Web Browser how to display a Web Page's words and images for the user.

4.2 The Back-end Servers

The Back-end is the part of a distributed network model that contains the business logic of a net application. In a simple Client/Server model, the front-end is the client and the back-end the server. The back-end can be a sophisticated Application Server, a simple Web Server or a set of different servers that run the business logic of the net application. In IT (Information Technology) terminology a Server is a program, which provides a service to other (client) programs (FOLDOC, 2000). A system for Web Based Distance Learning could have several different types of Servers or any combination of these:

- Web Server
- File Server (ftp Server)
- Chat Server
- Bulletin/News Group Server

4.2.1 Web Server

Web Servers are the cornerstones of the WWW. They provide the access to the information on the WWW. Commonly the Web Server is requested by a client to provide a specific resource. The resource could be anything from a simple HTML document to a movie or a sophisticated database search engine. The Web Server sends the requested resource back to the client. In the case of an HTML file, it simply reads it from the storage device and sends it over the Internet. In the case of a database request it passes the request parameters to the search engine, waits for the results and passes them back over the Internet. Web Servers communicate via the HTTP (Hypertext Transfer Protocol) protocol. The Hypertext Transfer Protocol is the set of rules for exchanging files (text, graphic images, sound, video, and other multimedia files) on the World Wide Web.

4.2.2 File Server

The Internet standard for a File Server is the File Transfer Protocol (FTP). FTP is a client-server protocol, which allows a user on one computer to transfer files to and from another computer over a TCP/IP network.

4.2.3 Chat Server

"Chatting" on the Internet is communicating with other people who are using the Internet at the same time. Usually this "talking" is conducted by exchanging typed messages between users. A Chat Server provides the service of exchanging the messages usually by receiving a message from one user and broadcasting this message to all users participating in the Chat. In special cases a private Chat can be arranged so that two parties exchange messages without any other users listening to it. Chat-Rooms or Chat groups are set up on the Chat Server focusing on a particular topic. Users must log into the Chat-Room/group to be able to participate.

4.2.4 Bulletin/News Group Server

The Bulletin Board System originates from the time when computer mostly communicated via modems and they operate independent of the Internet. A Bulletin Board System is a computer that can be reached by computer modem dialling for the purpose of sharing or exchanging messages or other files. The WWW offers its own Bulletin Board Systems. Documents, messages or files are exchanged via HTML. The difference to a Chat is its asynchronous state. A Chat works online. A message is sent and "instantly" received on the other side. A message that is placed on a bulletin board is saved on the server and can be retrieved any time from anywhere. The sender and receiver do not have to be online at the same time.

A Bulletin Board System is an ideal way to publish long lasting announcements or information.

A News-Group is a discussion forum for a particular subject consisting of notes sent through the Internet to a specific site and redistributed via the WWW. Users can read the notes and contribute by sending their answer or comments to the server. Servers are the programs that provide the service of organising, storing and maintaining the information, notes and messages and the service of redistributing.

4.3 Browsers

A browser is an application program that provides a way of looking at and interacting with all the information on the World Wide Web. Technically, a Web Browser is a client program that uses the HTTP protocol to make requests to Web Servers throughout the Internet on behalf of the browser's user. Once the browser receives the information from the Web Server, it displays the information and interacts accordingly with the user. The first widely distributed graphical Web Browser was Mosaic, which was developed at the National Centre for Supercomputing Applications (NCSA) at the University of Illinois in Urbana, Illinois in 1993. All commonly used browsers are based on Mosaic. Today two browsers are primarily used on the Internet, the Microsoft's Internet Explorer and the Netscape Navigator from Netscape Communications. Both browsers implement the standard features but have their own extensions, which are generally not compatible. To accommodate any possible user the contents should be tested with both browsers and in special cases offer two sets of pages, one set for each browser. At this point, it is usually recommended to provide a set of simple HTML pages to accommodate older browser versions. Personally I think supporting older browser versions is unnecessary because both major browsers are freely available (on the WWW) in their latest version for most common platforms.

4.4 Computers and Connections

In principle, any computer could be used as a Web Server. Serving as a simple Web Server for homepage-like contents (simple HTML files, graphics etc) does not require many resources. The computer only needs a permanent Internet connection and has to run 24 hours to be able to deliver a Web Server's contents at any time. However for busy Web Sites a more sophisticated design is necessary. There are two main issues to consider, the traffic that reaches the host and on the amount of computing necessary to serve the request. For example, the design for a host that provides simple HTML pages but has thousands of requests daily should focus on the network connection. A host that receives only a few requests but has to search an extended database or solve complex mathematical calculations should emphasise a fast processor and memory. For a distance learning system the design also varies depending on the contents and the expected traffic. Distance learning sites that offer the basic WBI (Web Based Instruction) components, with contents based on text and simple graphics but having many users, can certainly make do with a standard computer configuration and a fast Internet connection. A site that offers remote simulations or remote software access will however need substantial computing power to complete tasks and return results to the user in an acceptable time period. When designing a Web Site it is crucial to keep in mind that users will not wait too long for a response. The temptation to browse another Web Site by a simple mouse click is very significant. For a distance-learning course where concentration on the topic is important and every distraction should be avoided, this temptation is deadly. While waiting, users tend to find another occupation and this will disturb the learning. The WWW is a fast growing and fast developing medium. It is thus also very important to expect that new technology must be incorporated into the system at a later point in time. To face a total redesign because the number of users exceeds the expectations or because new technology cannot be integrated, is not only annoying but also expensive in time and resources.

4.5 Distributed Computing System

If the workload of the host becomes too big for one machine to handle, the creation of a distributed computing environment is a widely accepted solution. B.W Lampson (Lampson) described the distributed computing environment as "an environment in which some time, when you can least afford it, the failure of some computer, which you did not know existed, situated someplace, will cause your job to fail." Apart from this rather cynical view, a well-designed distributed approach provides an easy interface for the programmer and excellent administration facilities. By dividing one application program into two or more pieces, (client and server), and distributing these resulting programs over two or more computers for processing and computing, a distributed computing system is created.

Having a closer look at the definition above, the WWW is, in principle already a distributed computing environment. The browser as the client, sometimes also called the front end, which provides the user interface, formats the data received from the server or formats a request for data from the server. The server, sometimes called the back end, performs the computation of data or the storage of data for retrieving of the client.

Creating a distributed computing environment for a Web Application is distributing the workload of one Web Server onto two or more computers. The front end and the interface do not change. For example, the Web Site of the University of Cape Town with the address http://www.uct.ac.za offers a number of Web Pages. Whether the Web Pages reside on one computer or on several is not visible to the user. The client requests the data always from the same computer but might get the reply from a totally different computer, residing somewhere else on the Internet. The Web Server receives a request for a specific service and passes the request on to the computer that handles the service. The computer processes the request and either passes the response back to the Web Server or directly back to the client. Figure 2

shows the structure of a typical distributed Application and the communication links between the components.

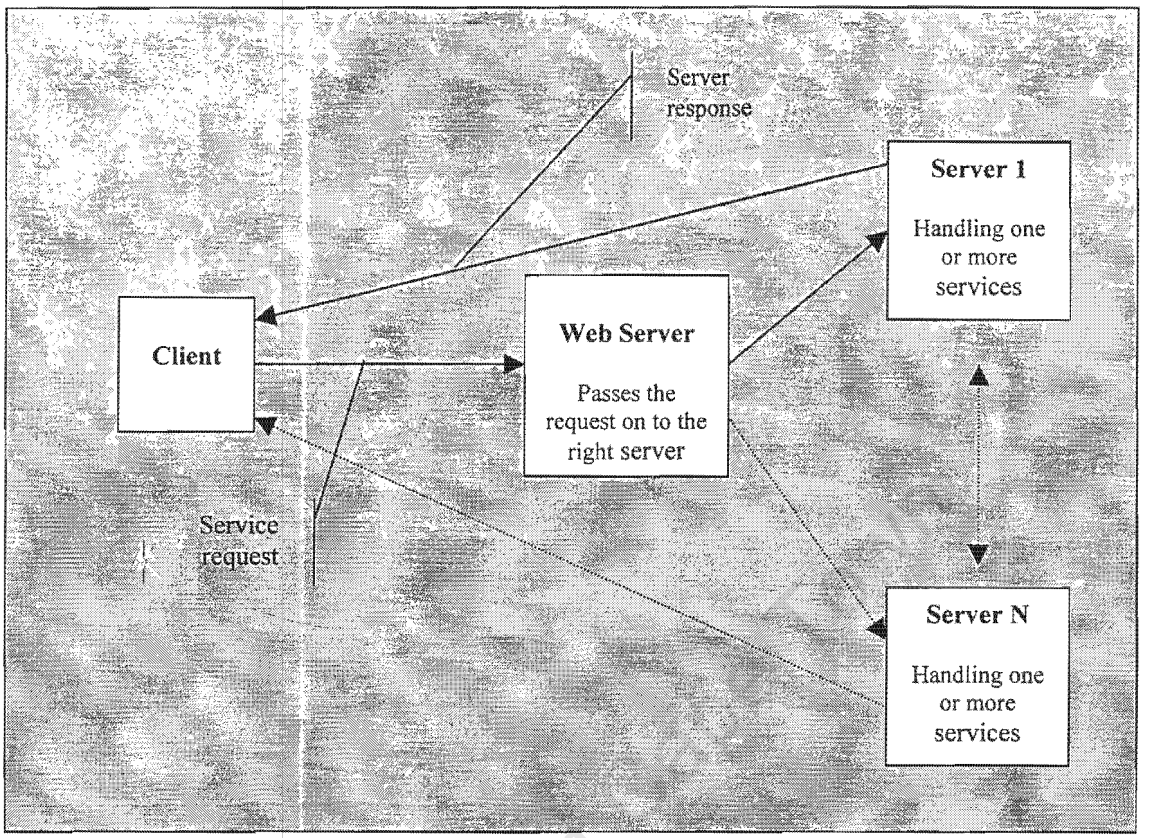


Figure 2: Distributed Web Application

4.6 Conclusion

"What do I use?" is a broad question if applied to a WWW application development. New tools and techniques for delivering information on the WWW become rapidly popular and then disappear just as quickly. Only a few developments become an industrial standard. The decision depends mostly on the course requirements, financial constraints, infrastructure and a big part of personal preference of the designer. A good recommendation is: design as sophisticated as necessary by keeping it as simple as possible. Utilise the power of the WWW and all its technical possibilities to create a course that meets all the requirements. Use widely accepted techniques and standards to ensure a problem free and easy maintainable lifetime of the system. In Chapter 7 the author describes his motivation and choice of tools and techniques to implement a Web Based Distance Learning course.

5 Example Distance Learning Sites

This chapter looks at some examples of distance learning environments and describes their features and functionality.

5.1 PowerLearn

The PowerLearn site is under the joint development of the Iowa State University and the Virginia Tech. USA and was funded through an award from the USA National Science Foundation and the Electric Power Research Institute (PowerLearn, 2000). The site (http://powerlearn.ee.iastate.edu/main/intro.html) provides a set of instructional modules for use in electric power engineering education. Each module consists of three components: presentation, library, and simulator. The presentation contains theory, examples and assignments as text as downloadable files or HTML pages for viewing with a Web Browser. Additionally, slides containing key points to present in a classroom in the form of overheads are available to download. The library is a collection of miscellaneous elements pertaining to the module. Some library elements are textual and others are visual. The simulator is an interactive, highly visual MATLAB program that runs on the student version of MATLAB 5.0. Each simulator provides a structured experience for observing basic concepts related to the module.

5.2 Tensor and Relativity

The site (http://vishnu.mth.uct.ac.za/omei/gr/index.html) offers Introduction to Tensors and Relativity and resides on the UCT network. The aim of this Web resource is to provide a central information point for students taking the course at UCT and a way of delivering material to students taking the course outside of the University of Cape Town.

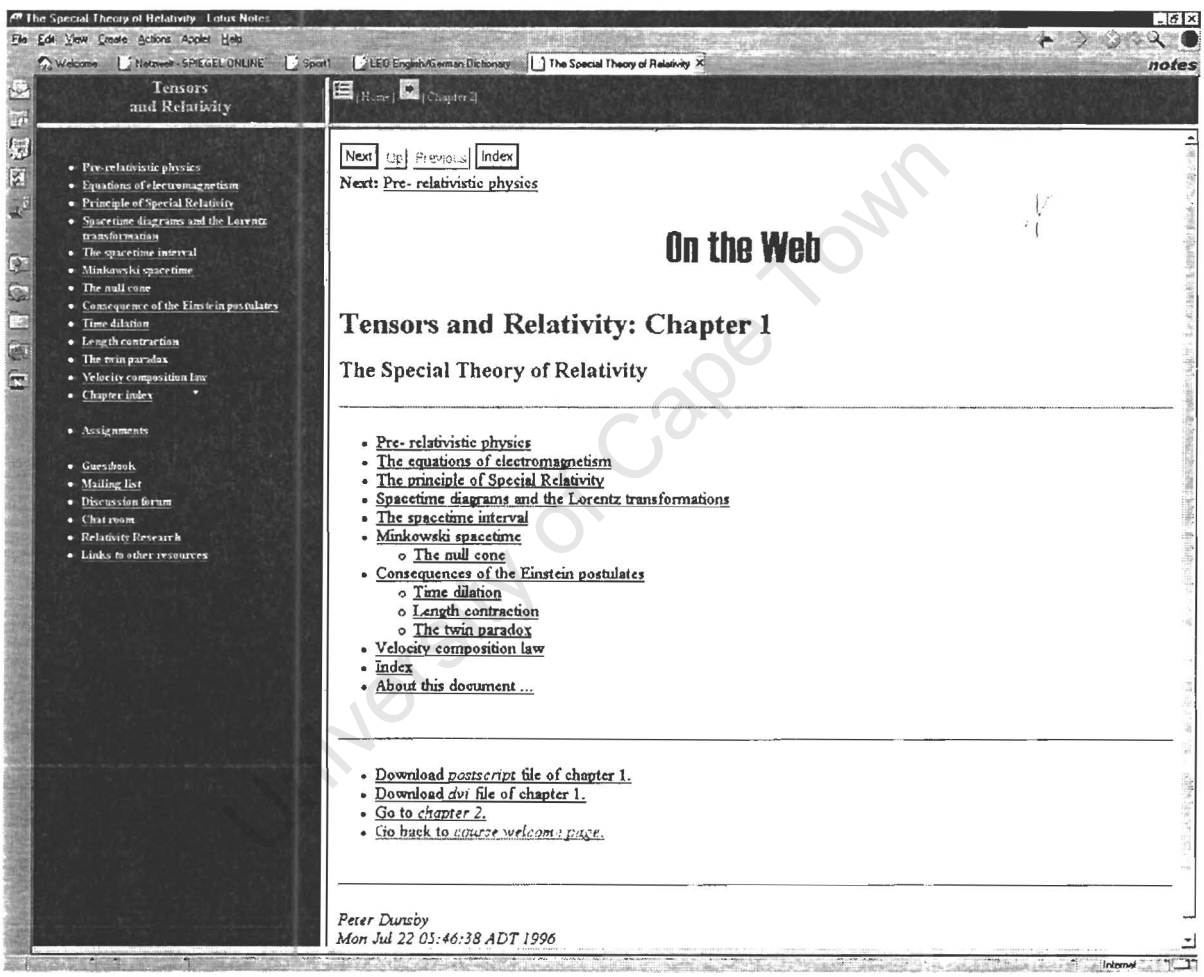


Figure 3: The Tensor And Relativity Web Site

5.2.1 Course Notes

The course notes are simple files enhanced with graphs in HTML format, hierarchically structured into chapters and subchapters. The user interface is the browser display area divided into 3 frames. The left frame features a table of contents that shows the topics of the current chapter. Each entry is a direct link to the specific page, which is then loaded into the main frame. In addition it shows a group of links to advanced features such as discussion group or guest book. The top frame offers buttons to skip through the chapters and a button that leads back to the start page. The main frame displays the course notes. Each chapter starts with a contents page with direct links to all topics and sometimes a little introduction. Each page features, mostly at the top of the page, some buttons (up, next, previous) to navigate through the pages. A button (index) that links to an index page of the chapter completes the navigation features.

Assignments are an integral part of the course. They are designed to help the student learn the material and understand the topic. The assignments are available in HTML format on the WWW and to download in several document formats. After submitting the answer via Email, the student can download the solutions from the WWW.

5.2.2 Additional Features

The site offers several additional features to create a virtual community

- Discussion Group
- Chat-Room
- Mailing List
- Guest Book
- Tutorials

The discussion group is a conventional HTML based system. The "message board archives" page lists all available entries and by clicking on the entry the system displays the message in the main frame. From this point the user has 3 choices. Either to return to the "message board archives" page, post a follow up message or post a new message. Both "post" pages offer simple HTML forms to enter a message and to send it back to the server. A CGI (Common Gateway Interface) script processes the server side business logic for the discussion group.

The system offers a CGI based Chat-Room. The communication is conducted via the HTTP protocol. The typed messages are send to the CGI Script on the server side, and after a time period, the browser requests an update from a CGI. The Server responses with a new HTML page containing all messages. Subscribing to the mailing list adds the user to the public distribution chain of the site. All changes or interesting news will be sent out via Email. The guest book is a little feature that offers occasional visitors to leave their comments about the site. The feedback received through this feature is valuable to improve the contents and the effectiveness of the overall system. Tutorials are currently only available at UCT.

5.3 Conclusion

In principle the PowerLearn site is a simple repository of course notes and tools to help a lecturer design and teach a topic. It is an advanced version of a textbook, which utilises the opportunities of the WWW to distribute and exchange information, enhanced with multimedia, world-wide.

The "Tensor and Relativity" site offers (according to the entries in the guest book) valuable information in the specific field. The structure of the course notes is simply organised by topics. For visitors familiar to the topic the orientation is probably easy. Following the offered navigation tools and the constantly changing table of contents

left the author several times lost in hyperspace. The position in the course notes and the simplest way back out was not clear enough to find. With the additional tools, the Web Site creator tries to incorporate a virtual community into the course.

Whereas the Powerlearn site is just a conventional collection of information published over the Internet, the "Tensor and Relativity" site tries to incorporate features of the WWW into its course. Still both sites are not utilising the full power of the WWW and the Internet. Including a feature because it is available (like the Chat-Room for example) does not automatically create a virtual community. To provide an incentive to use the features is as necessary as the feature itself. Putting students and a teacher together into a classroom does not result into a learning effort. If all of the participants sit idle at their desks, there won't be a successful learning outcome. Only if they start doing something they might learn. The author's design as explained in Chapter 6 (Design and Implementation) tries to "instruct" over the WWW rather then only present information.

6 Design and Implementation Details

The main goal of this project was to create a Web Based Distance Learning environment that incorporates all necessary features of a distance-learning environment with the opportunities of the WWW. The system doesn't focus on the actual contents of the course notes but creates a general platform that supports any instructor to conduct distance learning for his course. This chapter describes the author's design and the implementation of a Web Based Distance Learning course. The design focuses technically on platform independence, free availability and easy maintenance. The decision of what of the above-discussed features and techniques (Chapter 3 and 4) to implement is made with the goal of instructing rather then presenting. The implementation focuses on communication between participants and online help/instructions from the instructor.

6.1 A Definition of the Project Modules

The project follows a modular approach. The modular approach allows the independent development and use of the modules. A module can be changed or even removed without affecting the other modules. A new module is added by linking it into the system structure. Even the location of the module is irrelevant; it could reside anywhere on the Internet.

There are six main modules:

1. Course Module
2. Virtual Classroom
3. Testing and Performance Evaluation
4. Virtual Library
5. Tutorial
6. System database

Most of the modules are subdivided into several sub modules following a hierarchical structure.

6.2 System Structure

As mentioned above, the system is structured into several independent modules. Each module is independent organised, developed and implemented which allows easy maintenance and up/down grading and creates flexibility for the project. Certain modules can be changed or entirely removed without having to change the whole structure and concept of the project. Some modules are divided into several sub-modules, as can be seen in Figure 4.

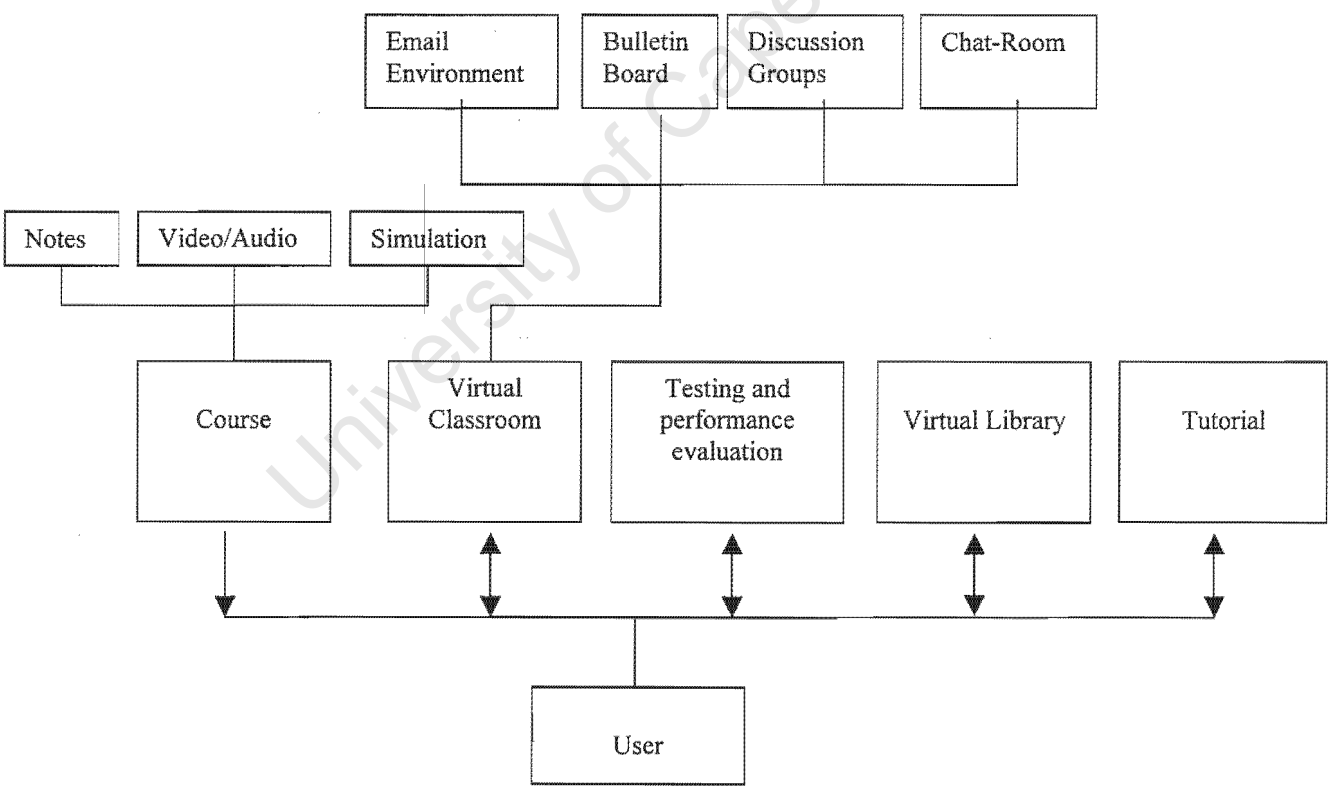


Figure 4: Module Structure

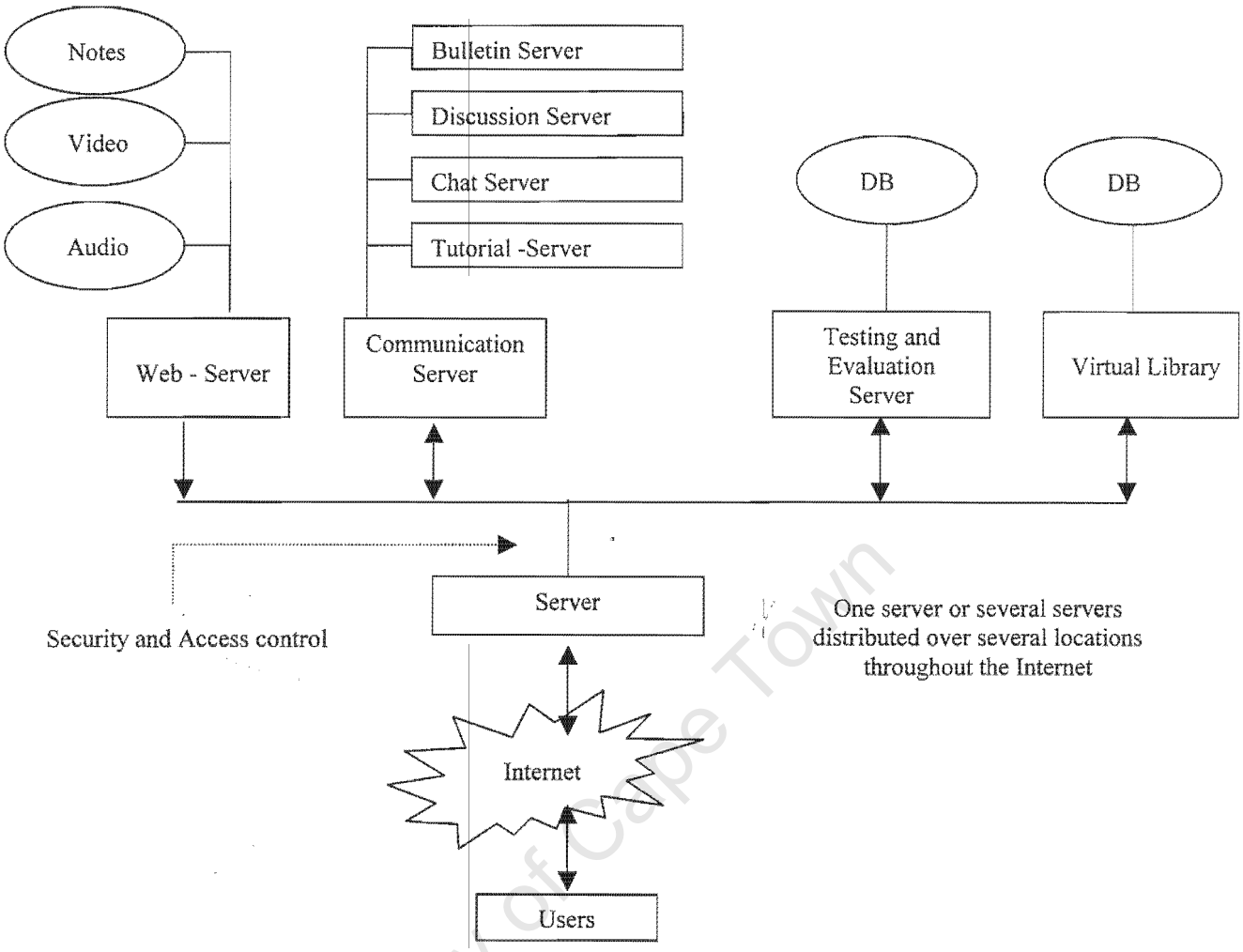


Figure 5: Architecture Of Software Components

The diagram above (Figure 5) shows the organisation of the software components related to the modules. It is a logical 3 Tier structure and if the resources are limited, the 3. Tier can physically be included into the 2. Tier and that makes it a 2 Tier structure. The modular design enables an extension from 2 Tier to 3 Tier very easily. The Web Server is the foundation of the system. It makes all the conventional Web contents (notes, video...) available. The communication server is a collection of small server programs running as Java Servlets that enable communication between participating users. Like the Communication Server, the Testing and Performance Evaluation module and the library are also a set of programs running independently. Each of these programs can be removed

independently or new programs can be added without major changes in the system structure.

6.3 The User Interface

The user interface is designed to support the learning effort and for the greatest efficiency in the use of the system. The system GUI (graphical user interface) is displayed inside the browser environment. The display is divided into 2 parts. The main frame displays all information and the toolbar frame contains buttons for navigation and to access all major facilities of the system. This arrangement does not change throughout the system. Whereas the content of the main frame changes constantly to accommodate the required information, the toolbar is always visible and does not change. By applying this concept continuously throughout the system, the users always feels comfortable and in control instead of having to adjust to a new design and finding their way around every new module.

The toolbar (see Figure 6) offers direct links to the virtual library, a Help System and the following virtual classroom features:

- Chat-Room
- Bulletin Board
- Discussion group
- Email

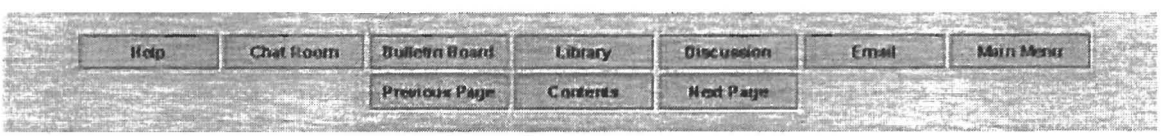


Figure 6: The Toolbar

The other four buttons (Main Menu, Previous Page, Next Page and Contents) are for navigating through the system and will be further explained below. "...in many on-line courses, links to the Chat-Room and discussion forums are provided at the bottom of every page. This is overkill." (Downes, 1997), some argue that access to the virtual classroom facilities should only be at one location in the system. That means, to start the Chat-Room or to read a discussion group entry, the user has to go to the defined starting point and interact at that location. But considering a real life scenario where a student has a problem with the course notes, the student will not read the notes, memorise that part and then go to the instructor or fellow student, quotes and discusses the problem out of his memory. More likely the student will take the notes and present them to the discussion partner. Today, any standard computer performs multi-tasking and permits a Chat-Room and the information to be displayed together. The GUI system utilises this potential by allowing all features interacting simultaneously. Figure 7 shows the typical frame layout.

Additional to the two frames of the display area, Pop-Up windows extend the GUI in certain cases. For example, the Chat-Room has its own window and all interaction is conducted inside that window. This allows the user to browse through other information in the main frame while chatting about a particular topic.

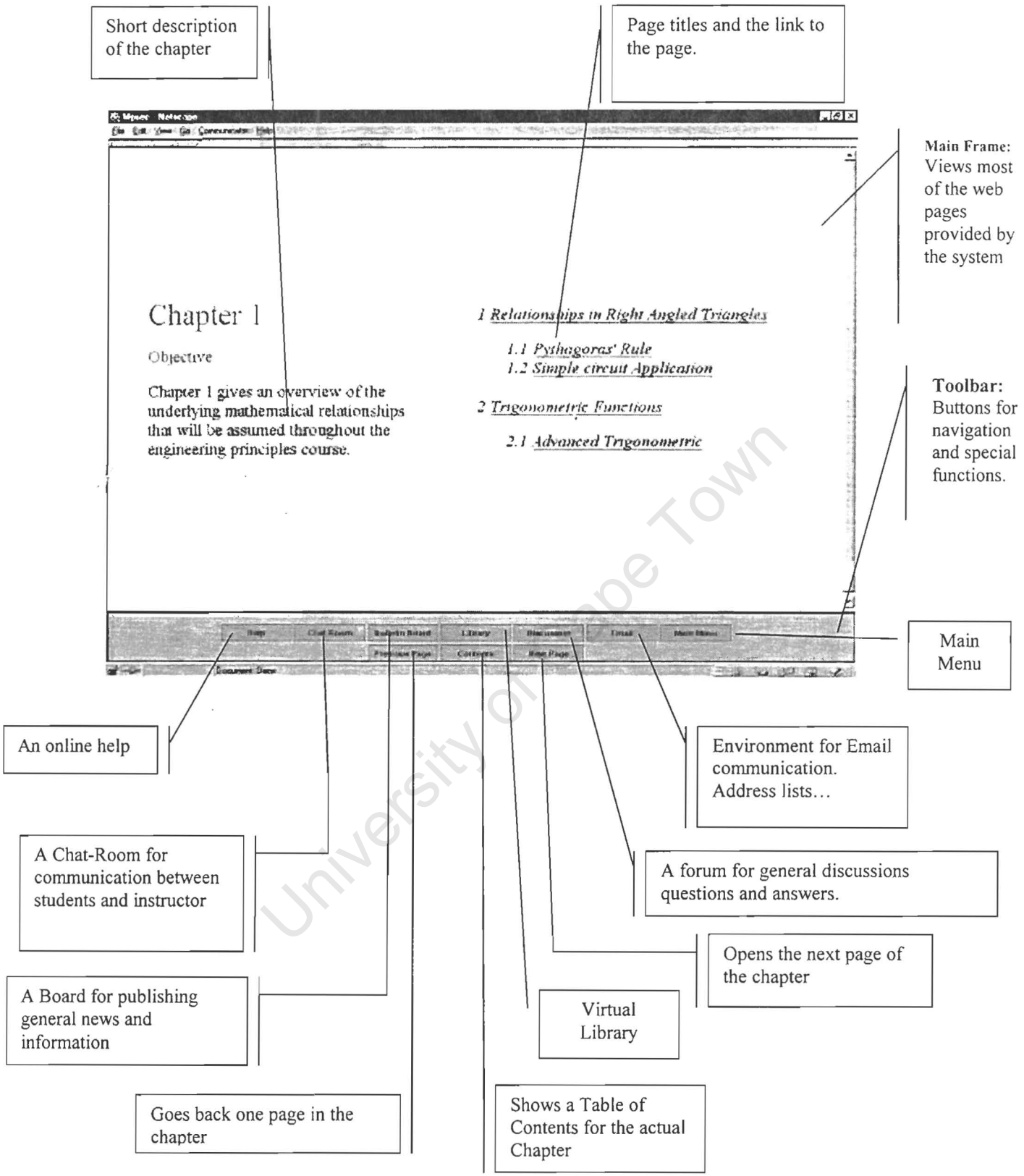


Figure 7: Frame Layout

6.4 Navigation through the Web System

With the Main Menu as the start page and the toolbar always visible, the user can navigate easily through all modules. The main menu offers the links to all course modules and by simply clicking on the links the specific module is started. The toolbar offers access to the virtual classroom module. The first page of each course module is a table of contents for that module. From this point, the navigation buttons "Previous Page", "Next Page" and "Contents" offer easy navigation. The "Next Page" button loads the following page into the main frame, similar to turning a page in a book. The "Previous Page" button does the opposite and returns to the previous page. The "Content" button returns to the chapter table of contents, whereas the "Main Menu"-button returns to the start page and enables the user to choose another module.

Why have an own "Next Page" / "Previous Page" button when every browser has one integrated? The "back" button of the browser loads the pages in exactly the same sequence as they were visited and the "forward"/"Next" button merely loads the pages in the sequence visited before. The system navigation buttons lead the users through the chapters in a pre-set sequence without getting "lost in hyper space".

By creating the start pages dynamically with a Servlet, the sequence of pages can be customised for the individual needs of a single student. For example, to avoid a student skipping certain pages by jumping directly to a topic later in the chapter the *list* array only contains the file names of the pages visited before and the immediate next. The student is forced to read the pages in a set sequence. The responsibility of the sequence is totally with the instructor. Accordingly to the instructor's didactical ideas, the sequence has to be defined. This allows great flexibility for the instructor but the administrational effort is substantial. For each chapter a separate Servlet has to be designed and implemented to meet the instructor's requirements.

6.4.1 Technical Implementation

JavaScripts and the dynamically created table of contents pages of each chapter control the navigation process. The Toolbar contains a JavaScripts with several functions that control what page or component is started, depending on the state of variables. The state of these variables is updated every time an event occurs in the navigation system. In Appendix 10.1 (Navigation System) is a list of these variables with a short explanation.

For example, the JavaScript function that is invoked by pressing either the "next page" or the "previous page" button adds or subtracts 1 from the *index* variable. The updated *index* variable, now points to a new item in the *list* array, which is then loaded into the main frame. Therefore the *index* variable is the only one that changes its state throughout a chapter. The other variables are initiated when the chapter is started and the JavaScript functions utilise the information stored in the variables during the navigation in the chapter.

The dynamically created start pages (table of contents page) contain a JavaScript that handles the selection of the table of contents and updates the Toolbar variables with the chapter specific data.

6.5 Dynamically Created and Customised Web Pages

Like conventional books, the majority of Web Sites are designed for the general user (of the publications targeted audience), the same information in the same form and quantity for every user accessing that publication. Individual needs and personal requirements are not acknowledged with this approach. Each individual user is faced with the task of working through all available information and to filter the useful and the redundant information without guidance or help. For example, a student needs to know about a very specific topic. The normal approach might be to

find a book in the library that covers the topic and to read it. Most likely, the book contains all information about the topic but that does not guaranty that the student actually understood the topic. Maybe the book does not cover some prerequisite information and background knowledge. And if, how can be ensured that those chapters are actually read by the student? To meet the needs of each individual user of the system, user specific customised Web Pages, are a flexible and powerful tool. The instructor decides what route through the available information each individual student has to take, and the instructor also monitors and guides individuals according to their progress. Tests, assignments and personal communication can be utilised to support the instructor in the assessment of the individual student. At university for example, if a student fails the mathematics course in the first year, the student might not be allowed to participate the second year mathematics course because the student clearly did not understand the basics. With user specific customised Web Pages the same principle is used on a chapter-to-chapter or even page-to-page basis. Depending on the policy and individual involvement, the administrational expenses and the effort that the instructor has to make, can be very extensive.

The author's implemented system focuses on individually customised "Table of Contents" pages. They are dynamically created when the student requests the specific chapters "Table of Content" page. According to the student's previous achievements, only the topics necessary and useful (according to the instructor) are made accessible. The instructor can predefine even the sequence of the pages. After each step the next step is automatically made accessible to the student. This approach differs significantly from the general perception of a Web Based Distance Learning Course. It enables a high level of personal guidance and individual attention apart from the communication features. The information presented is customised for the individual user.

6.5.1 Technical Implementation

A server side program creates dynamic Web Pages, right at the time of the resource request. Normally a request to a Web Server to return a HTML-page will cause the server to read the HTML-file byte by byte from the storage device and send these bytes to the requesting client. For a dynamically created HTML-page, the server follows a program algorithm to create these sequences of bytes according to data and the requirements. After the algorithm is completed, the sequence of bytes are exactly like the ones read from a storage device.

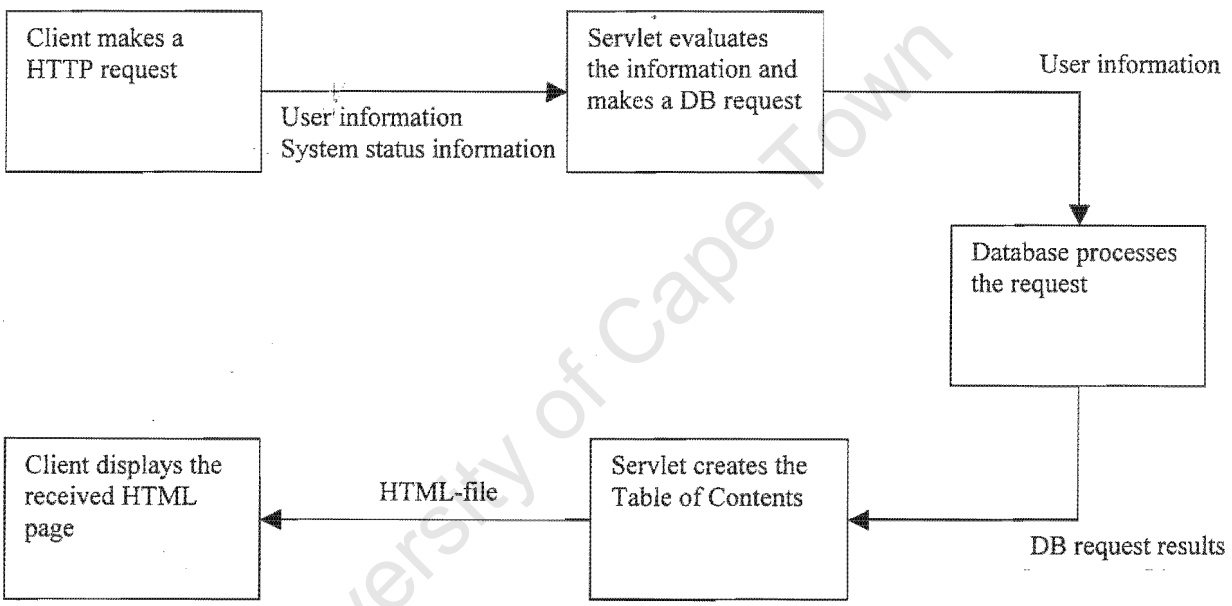


Figure 8: Data Flow Diagram

Java Servlets are implemented as the Server Side programs. The Servlet receives the request for a Table of Contents page through the HTTP-protocol. The request (HTTP-Header) contains the information of the user making the request. According to the information stored in the system database for this particular user, the Servlet creates an HTML-page. If the user has access to a page in the chapter, a HTML hyperlink to this page will be included, otherwise not. The data flow between components can be seen in the data flow diagram in Figure 8. Also JavaScript

information (see 6.4.1Technical Implementation) like the allowed sequence for the navigation is included, up to date with the newest information in the system database.

6.6 Course Module

The course module is subdivided into several modules that are equivalent to chapters. Each of these sub-modules is independently designed. Through the hierarchical structure of the system, each module has its own sub-directory in the file system. All notes, applet files, pictures, graphs etc. are stored exclusively there. The following Figure 9 shows the file system structure:

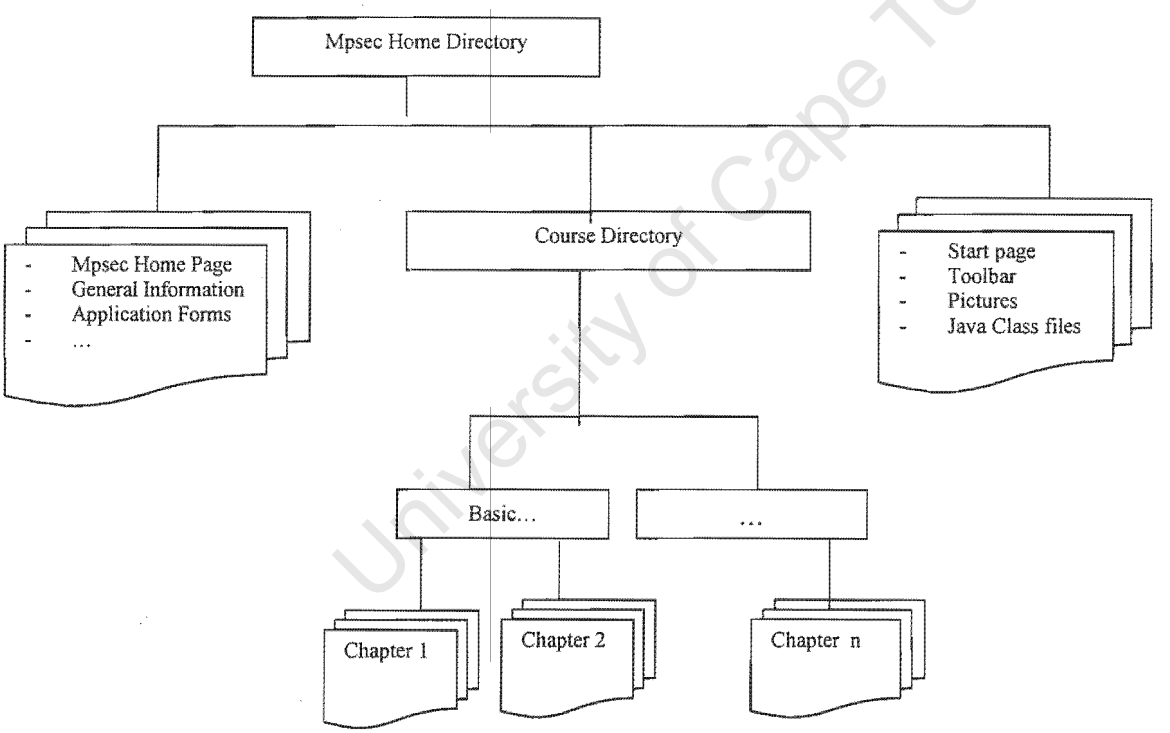


Figure 9: File System Structure

6.6.1 Course Notes

The notes of each chapter are divided into several subtopics, which consists of one or more pages. Each page is designed to fit entirely into the display area of the browser (depending on the browsers window size) to minimise the need of scrolling up and down. The browser software is certainly able to display long and extensive documents by supporting the scrolling feature. But this can be very disruptive, and it can be difficult to judge from the position of the scrollbar how long the document is and how much information is still to come. Displaying the whole contents of a page at once allows a fast recognition of the scope of the page and eases the eyes by not constantly having to adjust to a moving text. Figure 10 shows a typical page layout. Pop-up windows provide additional information or further details, which are not immediately relevant. Less relevant information does not distract from the main goal but can enhance the learning if needed. Example exercises and their solutions are also displayed in pop-up windows. Some pages contain hyperlinks to relevant information at some other location in the course notes, library or in the www. These hyperlinks expand the information offered on a page almost indefinitely.

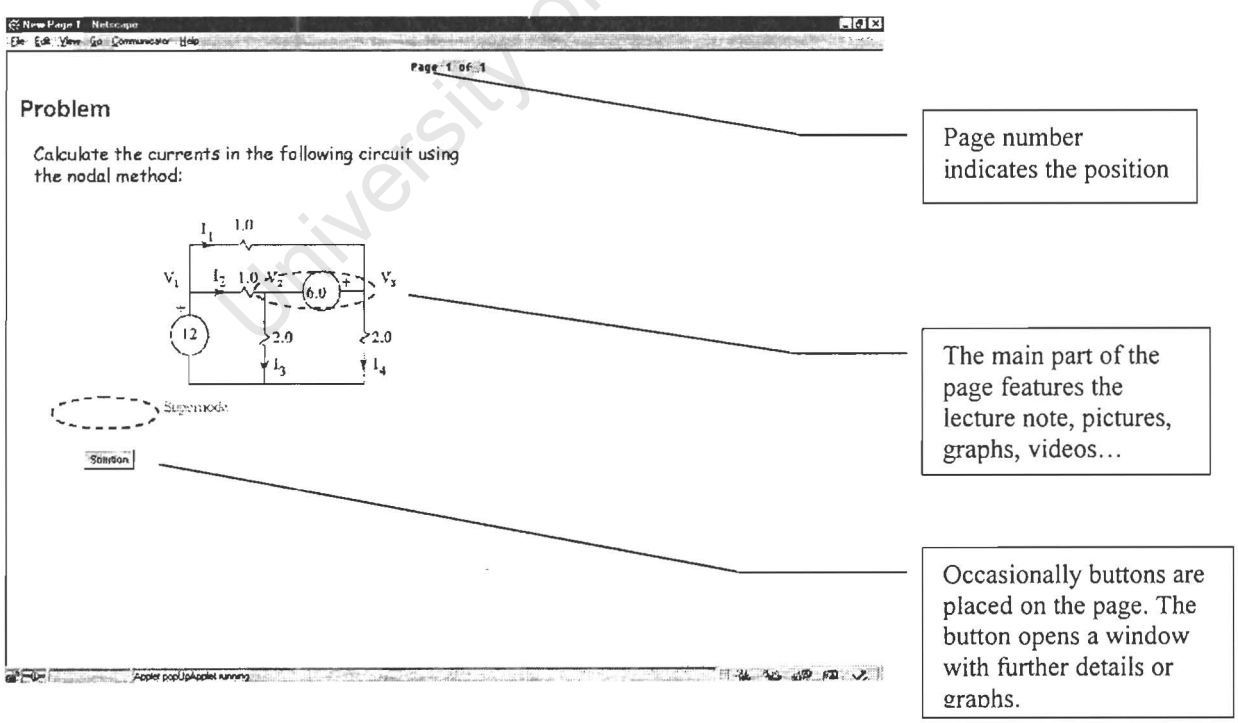


Figure 10: Page Layout

6.7 *Virtual Classroom Module*

6.7.1 The Chat-Room Software

The Chat-Room enables online communication between users. Messages posted to the Chat Server will be instantly broadcasted to all users who are logged on to the Chat-Room. The Chat-Room and also the Tutorial module (explained below) move away from the asynchronous state of the Web Based Distance Learning system towards a time dependent synchronous state. But since these components are additional features and not compulsory to use, the components add great communication capability to the system. The Chat-Room software consists of a server side component, the Chat Server and a user interface as the client side component. The software is implemented in Java, the client side as applet and the server side as a stand-alone Java application. The user interface is a Java applet, which is downloaded from the Web Server and runs inside the browser environment. A simple HTML tag requests the Java applet files from the Web Server and executes the Java files after the downloading is completed. Figure 11 shows the communication protocol between the Client side and the Server side.

The GUI of the applet offers simple input/output facilities to the user. There are two text fields; the upper one is the display area that shows all messages received from the server and the lower one is the user input field. By pressing the ENTER key, the contents of the input field is sent to the server for broadcasting. The EXIT button closes the connection to the server and closes the Chat-Window. The design of the GUI (see Figure 12) is very simple and straightforward for the purpose of communication. There are no user distractions and additional features to play around with.

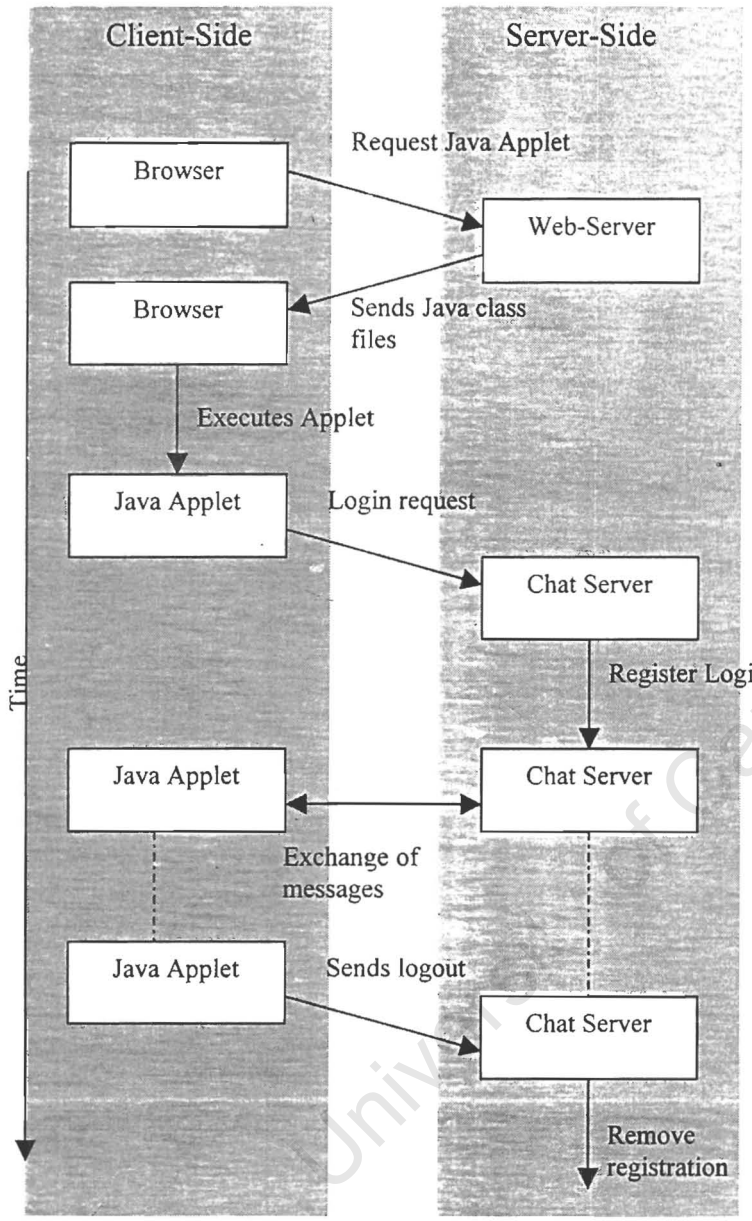


Figure 11: Chat-Room Communication Protocol

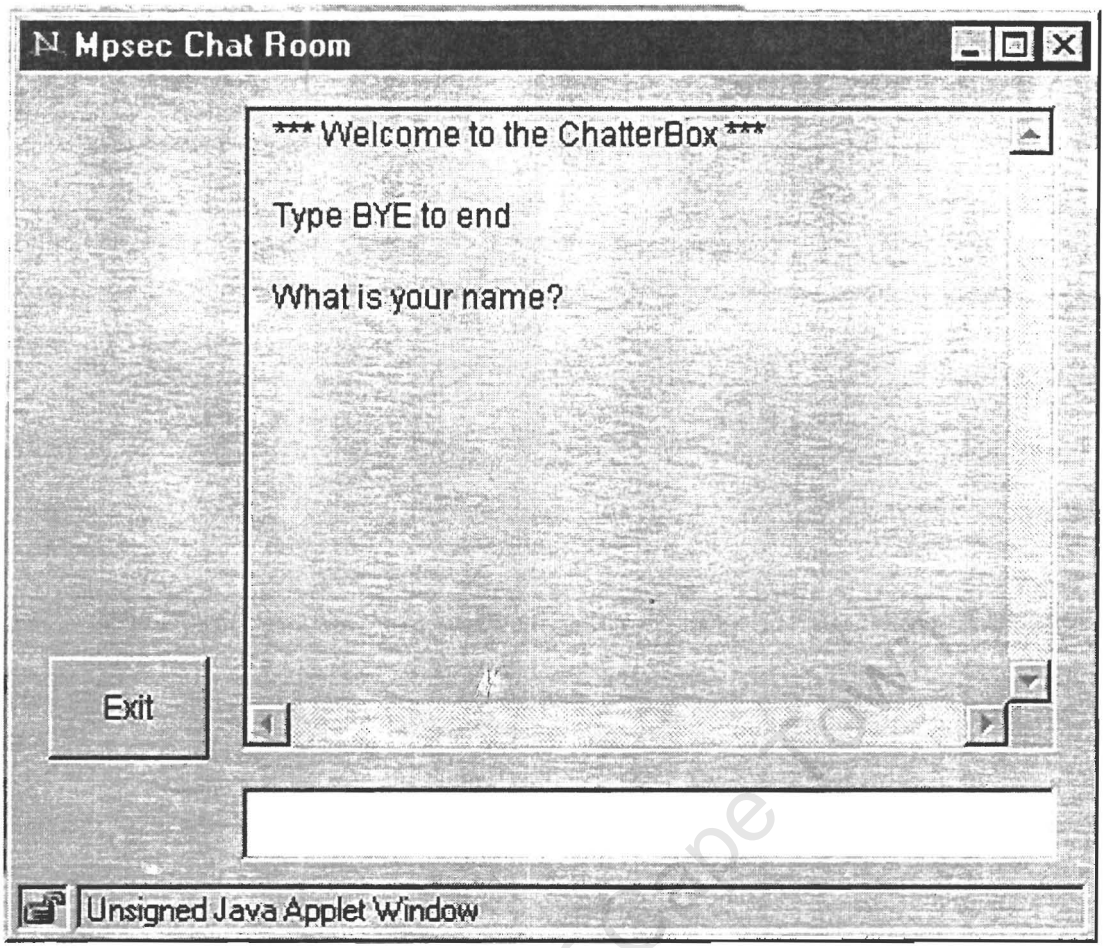


Figure 12: Chat-Room GUI

The Chat-room software is divided into two main processes, Chat Client and Chat Server and the two main processes run totally independently on different computers. There is only one Chat Server active on the system, but for each user (the number of users is indefinite) there is one Chat Client, one Chat Receiver and one Chat-handler running. The Chat Server listens constantly on its port for clients. When a client connects (via a socket connection) to Chat Server's port the server adds the client to a list and creates a new thread (Chat handler) for the client. This thread listens on a port dedicated to one client and passes all input received back to the main process. All input that the Chat Server receives from any of the connected clients (received through the thread) is broadcast immediately to all clients in the list, including the sender. Since a new Chat-handler thread is started every time a new client logs on, the number of sub threads (Chat-handler) for the Chat Server

process is indefinite. Because the occurrence of the event of receiving a message from the server is not defined, the client must be listening constantly to the server. To be able to have the user input independent from the receiving part, the Chat Client process is forked into two threads, the main process (Chat Client) and the Client-receiver thread. The Chat Client is running the GUI (Graphical User Interface) on the user side and sends all input to the Chat-handler (Server side). The Chat Receiver is waiting for messages from the Chat-handler and passes it on to the Chat Client for display. See the Chat-Room Petri-Net in the Appendix.

6.7.2 The Bulletin Board Software

The main goal for a Bulletin Board Software component is to make the publication of information easy, convenient and accessible for all users and administrators. Notices posted on the Bulletin Board have to be instantly available to everyone. The implemented Bulletin Board consist of the Bulletin Board Server on the server side and a user interface on the client side. The User Interface of the Bulletin Board is designed differently to the user interface of the Chat-room or Discussion group. Because the Bulletin Board is designed to feature general information, such as administrative news or course information, which is not considered part of the course notes, it is not necessary to have the information displayed in a separate window. The user interface is a set of custom made, dynamically created HTML pages. Depending on the user request, a specific HTML page with the requested contents is created by the Bulletin Board server. The main advantage of this design is that additional contents like tables, pictures and documents can easily (via HTML-tag) be included in the message. For example, course assignments or timetables are very simply inserted into the HTML page and therefore distributed to anyone interested. The differences in distributing information on a Bulletin Board and not, for example, by bulk Email is that the receiver decides if the information is needed rather than only the sender. The selection is done on both sides of the distribution chain and the user knows exactly where to find the information if needed.

Distribution via Email might result in somebody losing the information and then being faced with the problem of having to demand it again from the source.

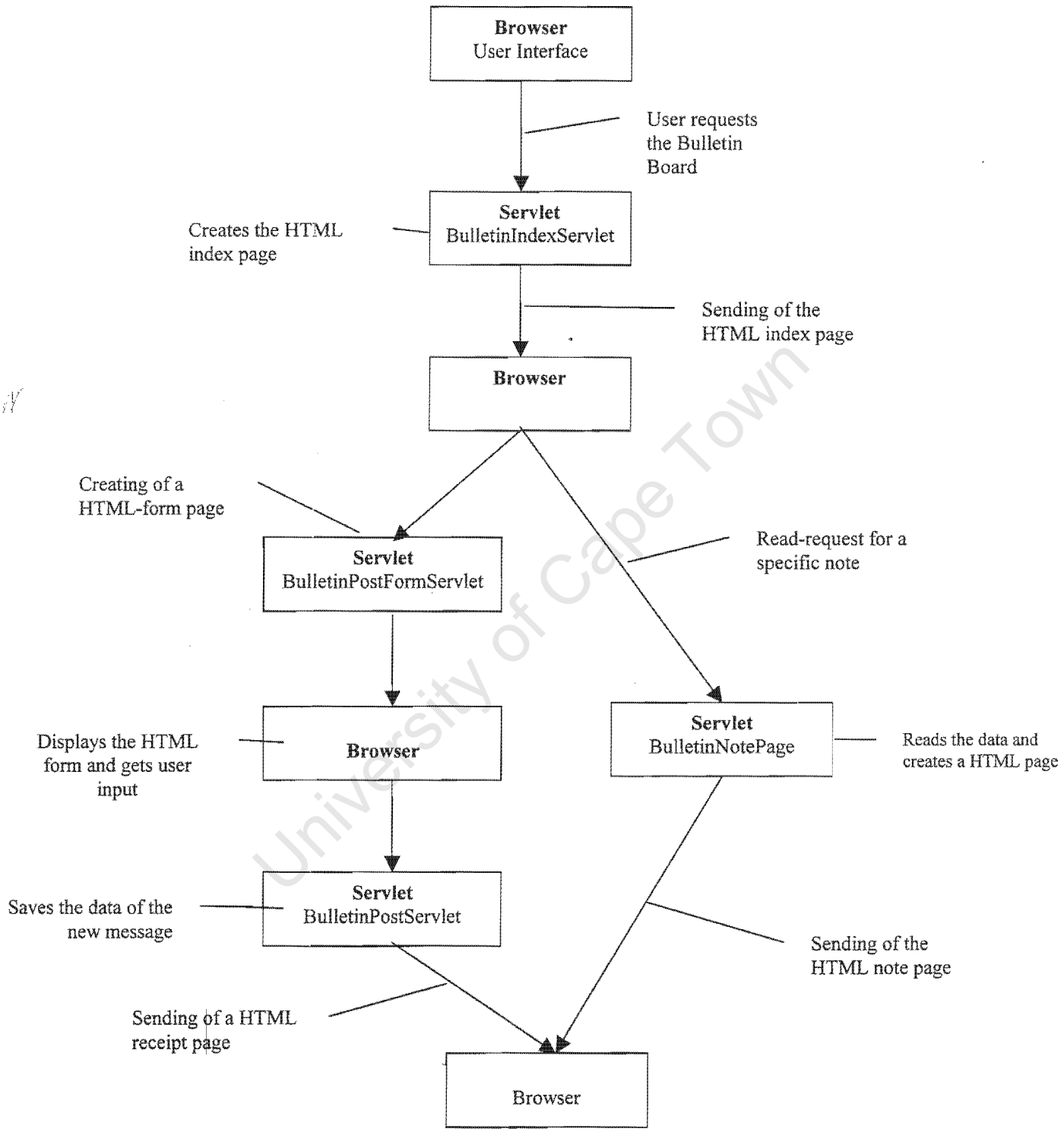


Figure 13: A State Model For The Bulletin Board

The Bulletin Board Server is a collection of 4 Servlets written in Java and is accessible through the Web Servers Java Servlet engine. Figure 13 shows the state model of the Bulletin Board software. The user interface passes parameter via the HTTP protocol to the Servlets and, according to the parameters; the Servlets dynamically create an HTML code, which is sent back as the server's response. The starting point is the BulletinIndexServlet, which creates a menu of all notes, sorted under various topics. To access a note, clicking onto the hyperlink starts the BulletinNotePage, a Servlet that creates a HTML page with the data saved on the server.

The Post buttons on the index page are linked to the BulletinPostFormServlet, which creates a HTML-Form page that enables the user to fill in the information and send it back to the server. The BulletinPostServlets receive the input and save it into the appropriate files. The update of the information is instantaneous so that access to the index page will show the new notes.

6.7.3 The Discussion Forum Software

The Discussion Forum serves as a medium to discuss problems and questions with a broad range of users. The service is time and user independent. A message (user contribution to a discussion) is posted and if a user responds to it, the messages are linked together. The discussion forum is different from the Bulletin board where there is only a one-way communication (a user posts a notice) and from Email, which is a personal one-to-one communication, and from the Chat-Room that is time dependent. The discussion forum thus offers any user the opportunity to reply to a specific message in public.

Similar to the Chat-Room component, the discussion group component is designed as a Java applet. The GUI is a separate window that displays all the main topics of

the discussion group. The submitted messages are organised into groups of related messages. Opening one messages by clicking on the topic, displays the content of the message and all messages related to it. Replying to a certain message, means taking part in the discussion. The newly submitted messages will be saved on the discussion server and linked into the previous submitted messages.

The GUI is designed as a separate window instead of a GUI as an HTML page in the main frame. This design allows the user for example, to browse though the course notes until the discussed topic is found and still having the message open on the screen.

6.7.4 The Email Environment

The Email environment is supposed to be more an address book and distribution list than a separate software component. Communication between all participants of the course, students as well as lecturer, is a main issue in any Web Based Distance Learning environment. Since Email is one of the most popular applications on the Internet, any Web based course should make extensive use of Email. All users are organised into courses and groups, for example a user is a member of the course eee387 and of the tutorial group X. A distribution list for each group is available. Emails directed to this list are automatically distributed to all group members. Email addresses of individuals can be easily located in the respective group or through the search engine.

6.8 The Virtual Library Module

Sometimes it is necessary for students to create their own perspective over the material. To enable the students to explore the material, or just to catch up on certain issues that slipped their mind, the virtual library offers a powerful tool. It offers direct access (e.g. by keyword search) to the page containing the information they are looking for without browsing through all the notes. The library also contains information that is not incorporated into the course notes and offers links to material existing anywhere on the Web. It also enables hyperlinks from the course notes to these materials without having absolute URLs inside the course notes. This avoids the need of working through all course notes in case the targeted contents change or are not available anymore.

Sven Willmann, an exchange student from the Konstanz University of Applied Science, researched and developed the virtual library for this research project in his undergraduate thesis (Willmann, 2000). The author laid out the guidelines and conditions for the interface between the system designed by the author and the virtual library. Sven Willmann has investigated and implemented a concept and design satisfying these constraints.

The virtual library is totally independent of the instructional system. The Toolbar (see "The User Interface") links to the virtual library and the start page is displayed in the main frame of the system user interface.

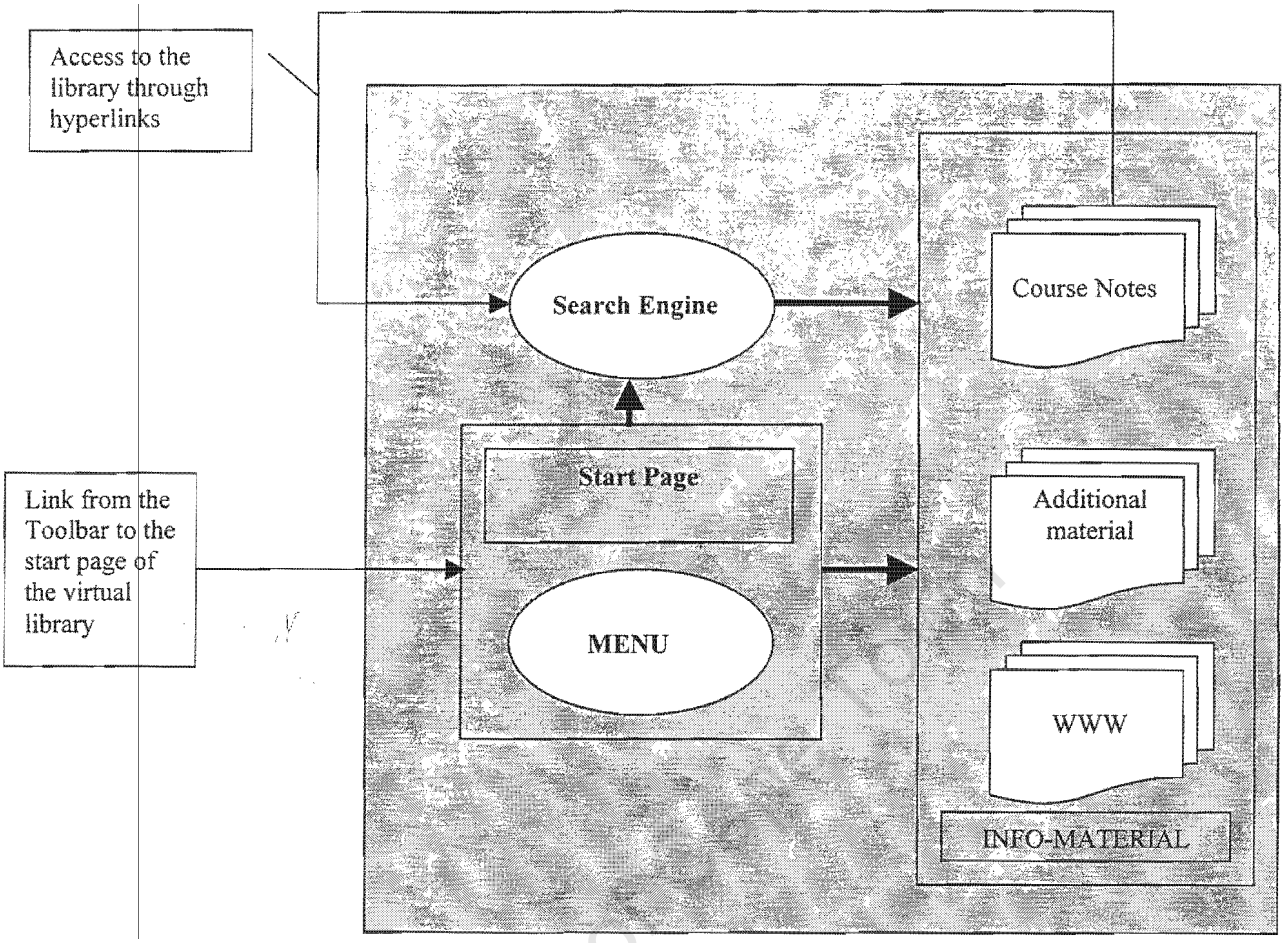


Figure 14: Library Access Structure

As can be seen in Figure 14, only the search engine and the menu offer access to the provided material. From the start page it is possible to enter a query for the search engine, or search through the menu structure for the desired information. In addition, there is the possibility of offering links in the course notes directly to material in the library. These links are addressed directly to the search engine and have certain keywords as parameters that are pass on to the search engine. The search engine returns the required response. More details about the design and implementation can be found in the thesis (Willmann, 2000).

6.9 Testing and Performance Evaluation Module

To ensure students have obtained the desired knowledge, it is useful to assess their learning. The students access their assignments or tests through the WWW and submit their work electronically to the instructor. The instructor assesses the work and gives his/her recommendation and remarks back to the students.

The Test Software

The Test software is used to evaluate the student's progress with the course. It automatically marks certain questions (e.g. multiple choice), and sends the results and the remaining questions for further marking back to the server.
The Test Software consists of the Test Server on the server side and a user interface on the client side. The Test Server is a Servlet written in Java and listens constantly for results posted to its port. The Server saves it in the appropriate file where the instructor can access it for further processing.

The user interface (Figure 15) is a Java applet, which is downloaded from the Server and runs inside the browser environment. The Java applet connects directly to the Test Server to post the results.

Options like time constraints for downloading the test applet, or certain time limits for posting the results can be easily implemented. For example, by adding a time constraining component to the code of the Java applet or by restricting the access to the files through the security and access control of the Web Server software.

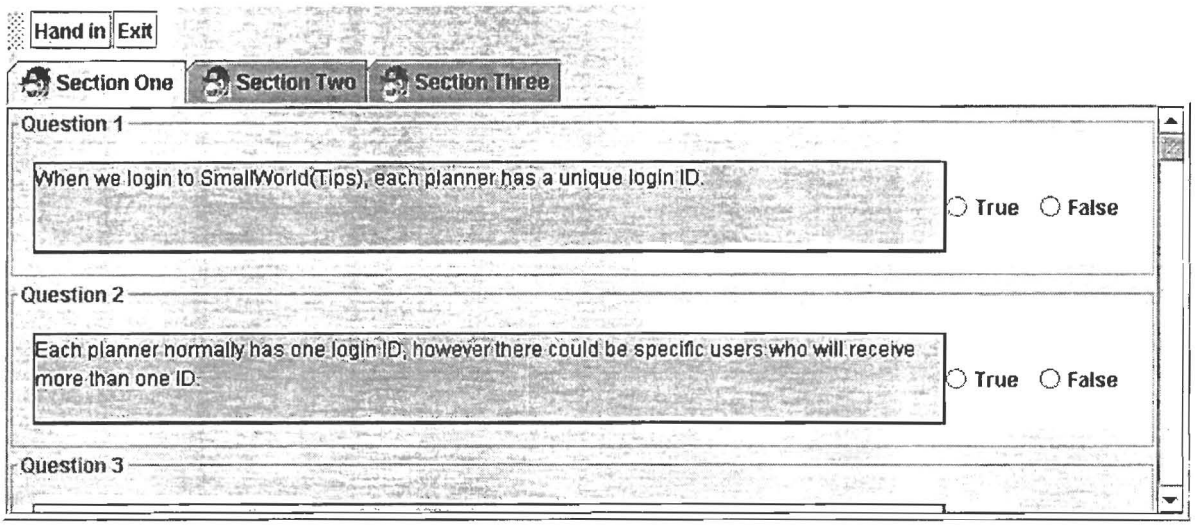


Figure 15: Exam Software GUI

Assignments

There are certain limitations for an online testing environment. The effort to give the student the same possibilities and convenience as a piece of paper for an online assignment is substantial, even excessive. But even for Web Based Distance Learning, not everything has to be done online. The instructor can create classical assignments for the students and distribute the assignments via Email or the bulletin board and receive the results via Email. The students are able to work with standard text processing software like the Microsoft Office suit, programming and simulation software (even through remote access) and create documents that can be submitted electronically. As long as there is a defined standard file format, the lecturer does not have a problem to receive the documents and process them similar to the hand-ins of conventional assignments.

6.10 Tutorial Module

The tutorial module is designed to bring more support to students in their learning effort and bridge the physical distance between student and instructors in the distance-learning environment. The tutorial module is modelled on a conventional tutorial, where students work on material or a problem with the instructor or another appointed person as a tutor to help and give guidance. The tutorial component extends the Chat-Room component of the virtual classroom module to monitor the student's focus on the material. The tutor can choose out of all students participating in the tutorial which one to monitor. If the tutor selects another student, the tutor's browser loads the page the student is currently displaying.

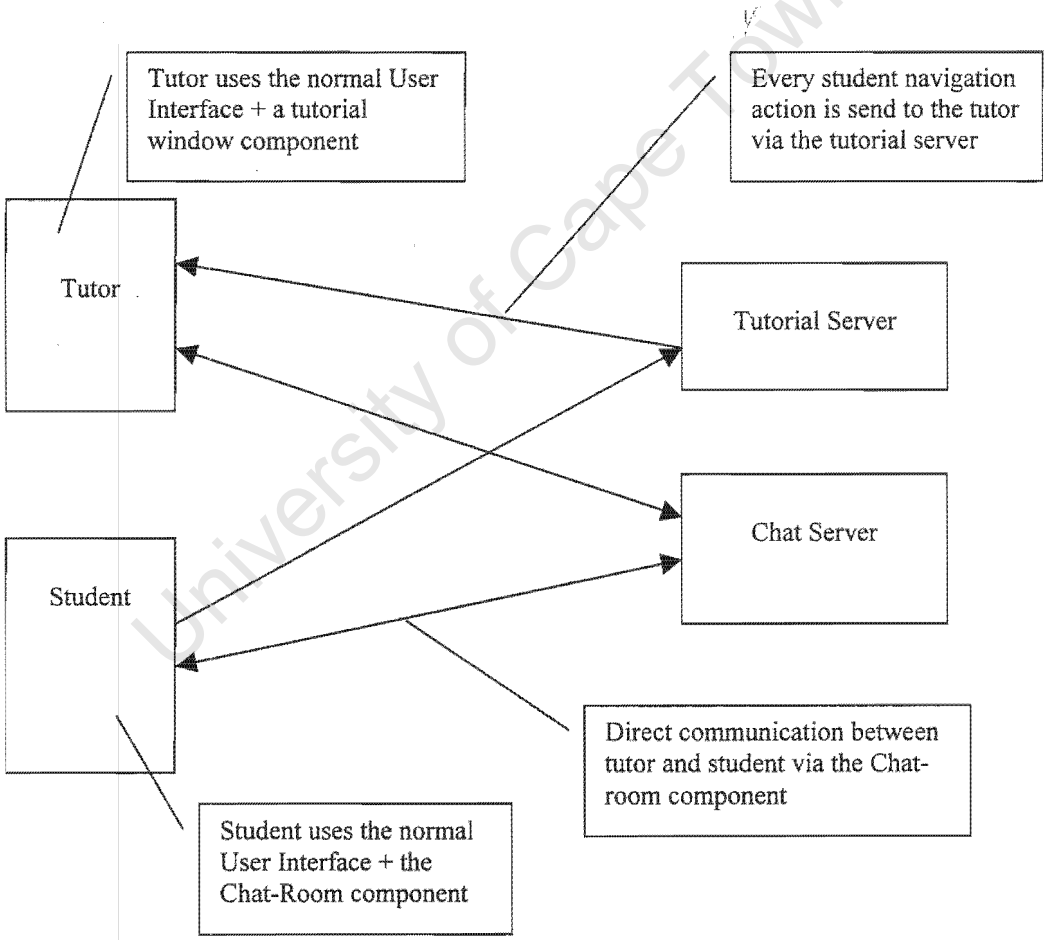


Figure 16: Tutorial-Component Block Model

Figure 16 shows a Block model of the Tutorial software. Each Block represents an independent software program. The arrows indicate which software components are communicating with each other. Every time the student changes the current page (displayed in the student's browser), the tutor is notified and therefore the tutor can monitor the progress of the students. For example, if a student spends more time on a certain topic than usual or more than the rest of the group needs, the tutor can focus on this particular student and give him/her guidance without disturbing the others in their learning. The communication between the tutor and the participating students is conducted through the Chat-Room component. The tutor is able to interact with the whole tutorial group. All students and the tutor are able to address everyone to share observations or information that is valuable for the group or they can address single users privately.

6.10.1 The Student User Interface

The user interface follows the principles of the system, and in addition the Chat-Room user interface of the virtual classroom component is used. The tutorial start page is a normal HTML page accessible through the main menu. All available tutorial groups are listed on that page. Selecting one of the tutorials on that page subscribes the student to the tutorial server and loads the start page of the selected tutorial into the main frame. From now on, every time the student changes the page, the change is sent to the tutorial server by an applet running in the background. The advantage of the Chat-Room's user interface design as a separate window is obvious, as it can be moved around the screen, next to the browser or overlapping and be used parallel to the browser. In this way, reading the course notes and following the conversation in the Chat-Room is made easy. Any change in the window (new message) can easily be detected and evaluated.

6.10.2 The Tutorial Window

The Tutorial Window (shown in Figure 17) is an additional user interface for the tutor. It is based on the Chat-Room window and as an extension features a list of all users logged into the tutorial group. The list is implemented as a text box with a descriptive label and a button. Clicking on the button displays a drop down list of all participating students with the name of their currently active page. Selecting one entry of the list loads into the tutor's browser the same page as that being viewed by the selected student. Activating the checkbox labelled "Private" sends the following messages exclusively to the selected user. Conversation held with the activated checkbox is invisible to all other users of the tutorial.

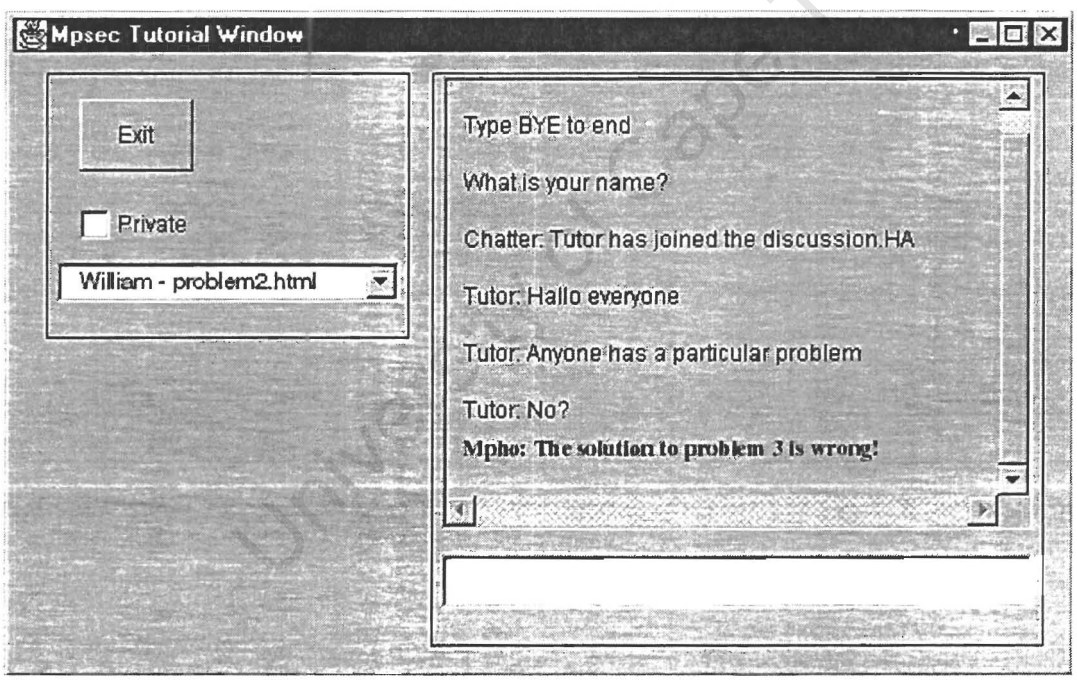


Figure 17: The Tutorial Window

6.10.3 Technical Implementation

The Tutorial software is divided into three main processes, Tutorial-Server, Tutorial-Window and Student-Client. They are running independently on different machines. Like the Chat-Room software, there is only one Tutorial-Server active on the system, one Tutorial-Window for every active tutor and one Student-Client per active student. Additional to these processes comes the Chat-Room component with its own processes. The components, which reside on different computers, communicate through socket connections over the Internet. The Tutorial-Server has two ports to listen to, one for Student-Clients to connect to and one for the Tutorial-Clients. (See the Petri-Net of the Tutorial system in the Appendix.)

The handling of the connecting process is equivalent to that of the Chat Server, only that it is done parallel for both ports. Due to the Java Security model, the Student-Clients cannot connect directly to the Tutorial-Window. Therefore the server is placed in between the two components and is broadcasting all changes received from the Student-clients to all active Tutorial-Windows. Since the receiving side of the server and the sending side are two different processes, a data class handles the exchange of data (see Figure 18) and a semaphore co-ordinates the access. Semaphores are a technique for co-ordinating or synchronising activities in which multiple processes compete for the same resources.

A semaphore (Whatis.com, 2000) is a value in a designated place, which each process can check and then change the value. Depending on the value that is found, the process can use the resource or will find that it is already in use and must wait for a while before trying again.

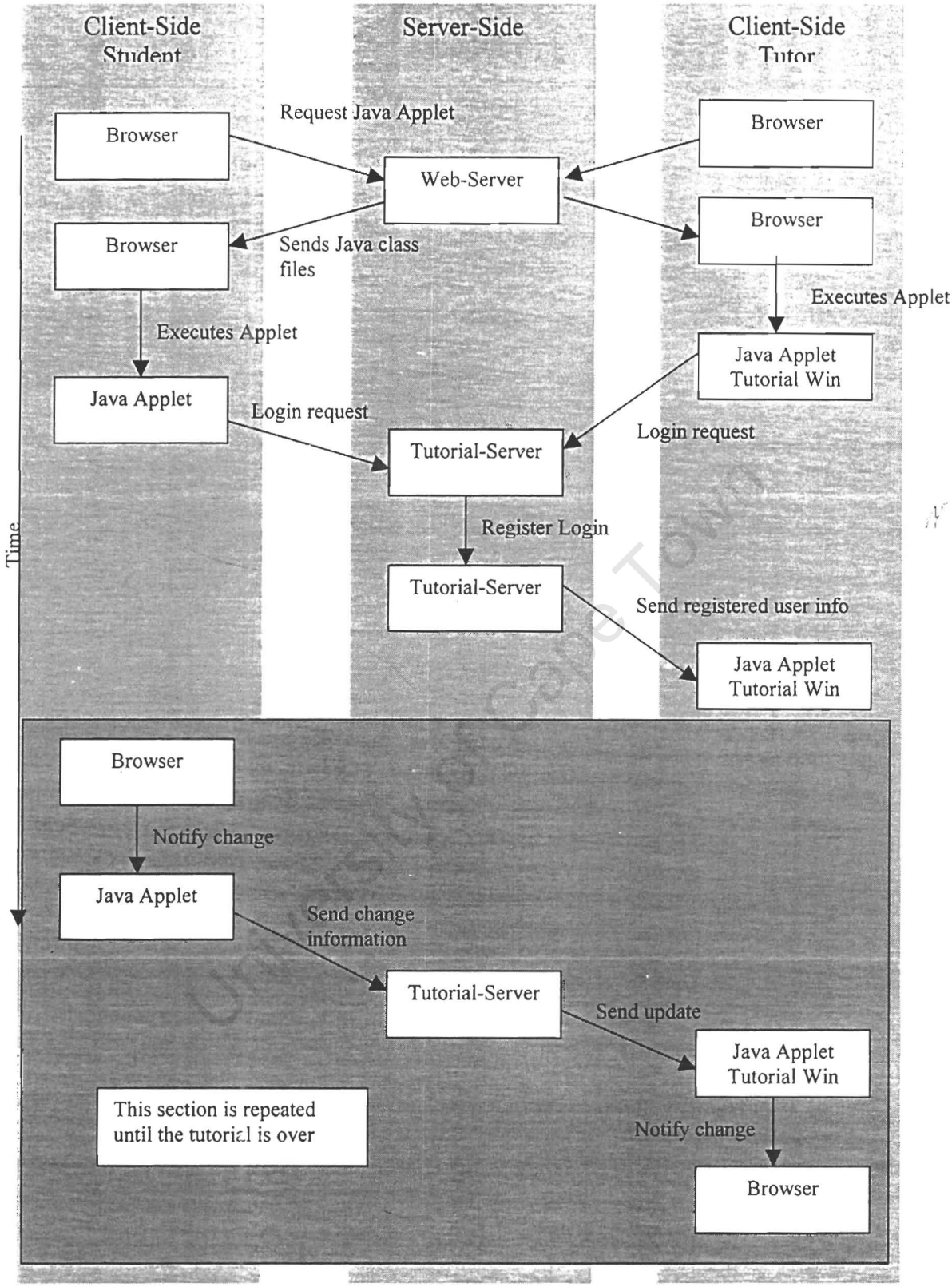


Figure 18:Tutorial Communication Protocol

6.11 Simulations

The simulation module is designed to enable students the access to simulation software that is otherwise not available to them. In this state of the project only an example simulation was implemented because the contents of the course and therefore the precise definition of simulations is not yet completely defined.

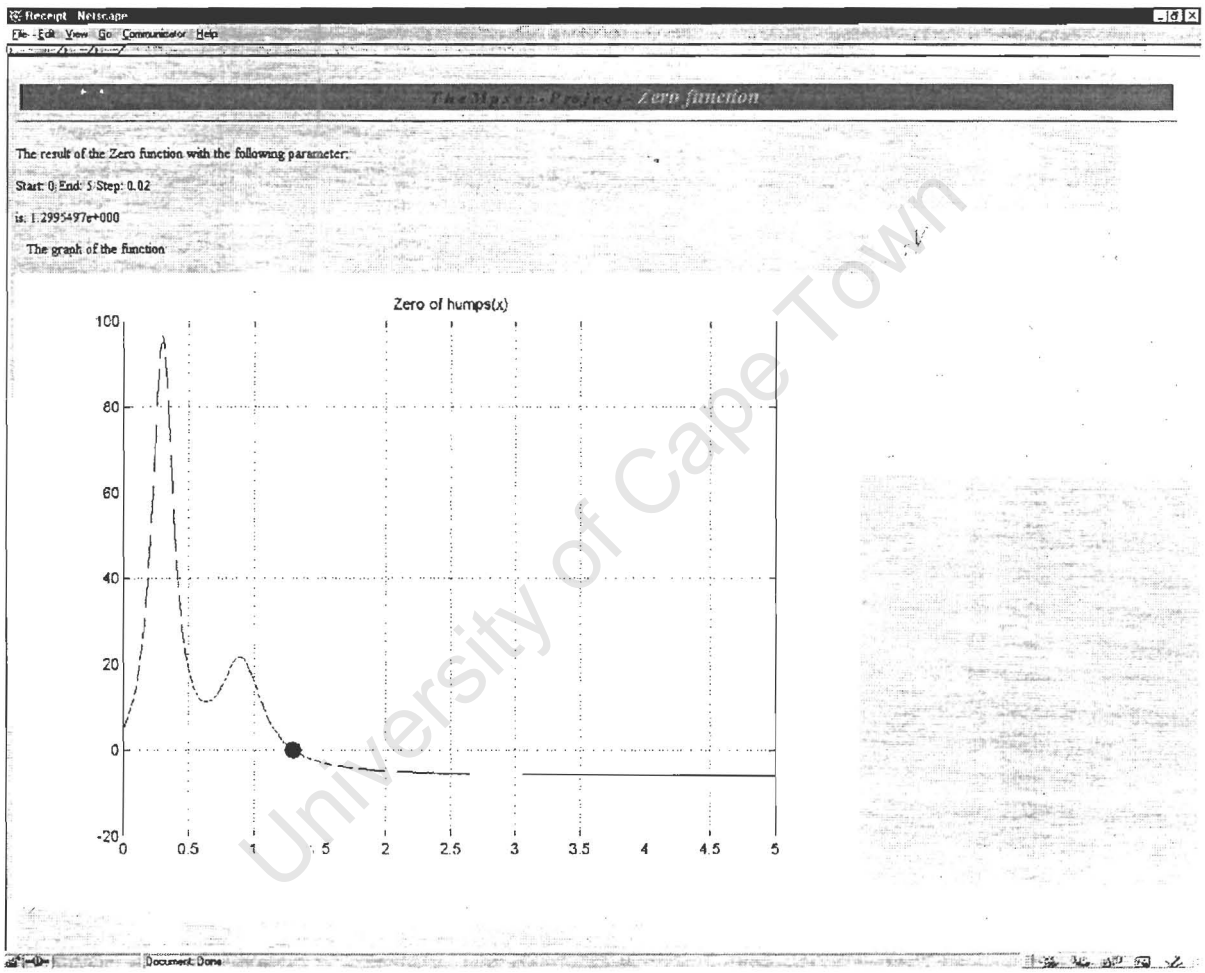


Figure 19: Simulation Results Page

The author implemented some example simulation with the Matlab engine and a WWW interface. The Simulation module consists of three components:

- Client
- Server
- Matlab Engine

The Client Side is simple the HTML pages that are provided by the server side. The Server Side consists of the Web Server that provides the static Web Pages and a Servlet that receives the user input and creates dynamically a result page. An example of a result page is shown in Figure 19.

As can be seen in Figure 20, the user enters the necessary parameters via HTML forms and these parameters get sent to a Servlet on the Server side. The Servlet passes the parameters to the Matlab engine. The Matlab engine passes the results back to the Servlet and the Servlet prepares the results into an HTML page that is send back to the client. If the simulation does not require too much time, the user can wait online for the results. In the case of a more time consuming simulations, the simulations can be started and the results accessed at a later point in time. This enables the student to continue with other parts of the course or even enables the student to start simulations in the evening, run it over night and check it the next morning for the results.

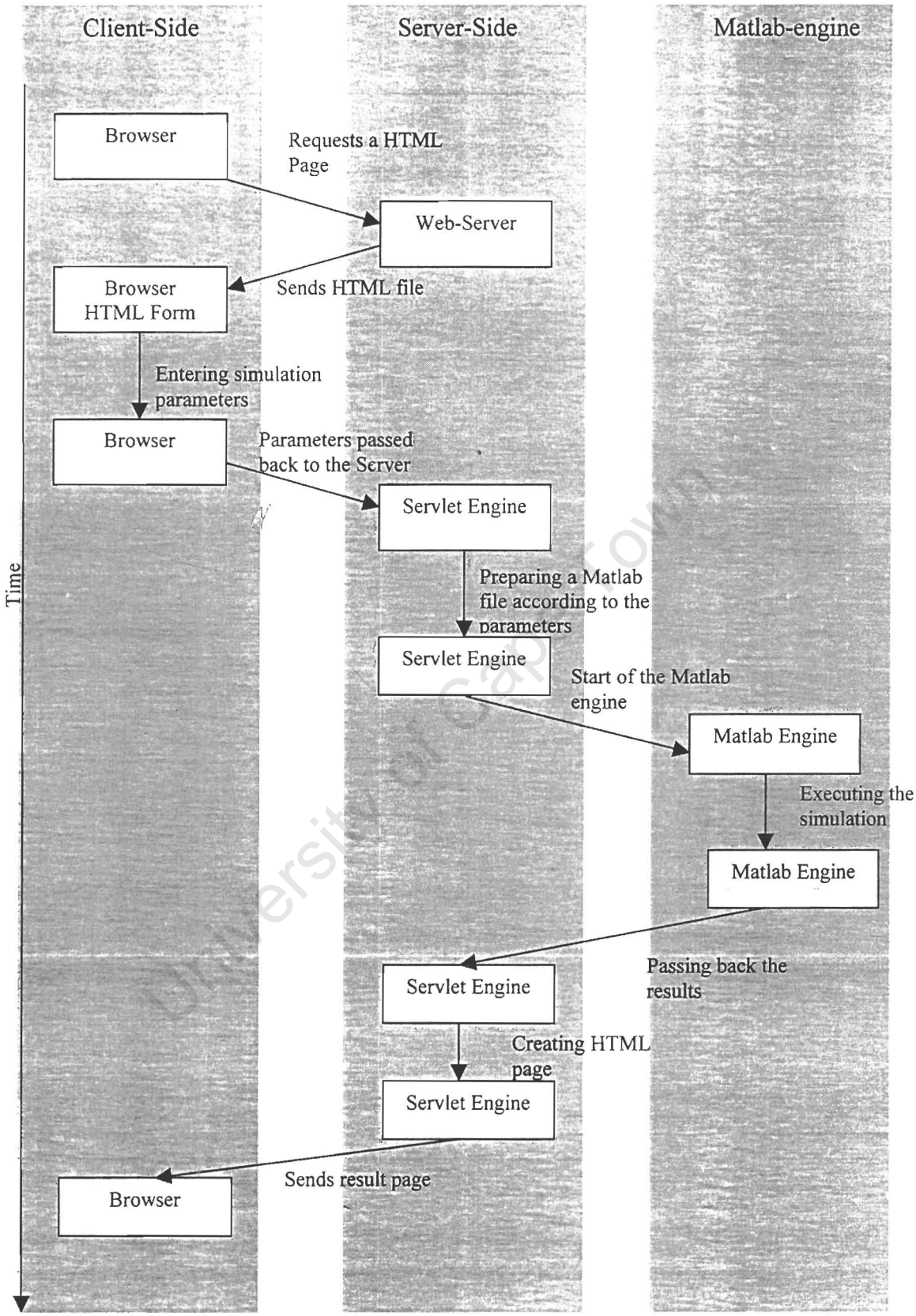


Figure 20: Simulation Communication Protocol

6.11.1 Technical Implementation

The implementation of the above-described procedure is more complicated as it appears. The available version of Matlab (5.3) has no interface to the Java programming language as it has for the programming language C++. It is possible to call the Matlab engine from a C++ program and utilise the Matlab computational power. The same feature is expected for Java in future releases of the Matlab software packet. To work around this disadvantage, the Java API Runtime class is used. Every Java application has a single instance of class Runtime that allows the application to interface with the environment in which the application is running. Therefore it is possible to start any program in that environment and execute it in a separate process. It is possible to write algorithms (so called m-files) in the Matlab programming language and execute them with the Matlab engine.

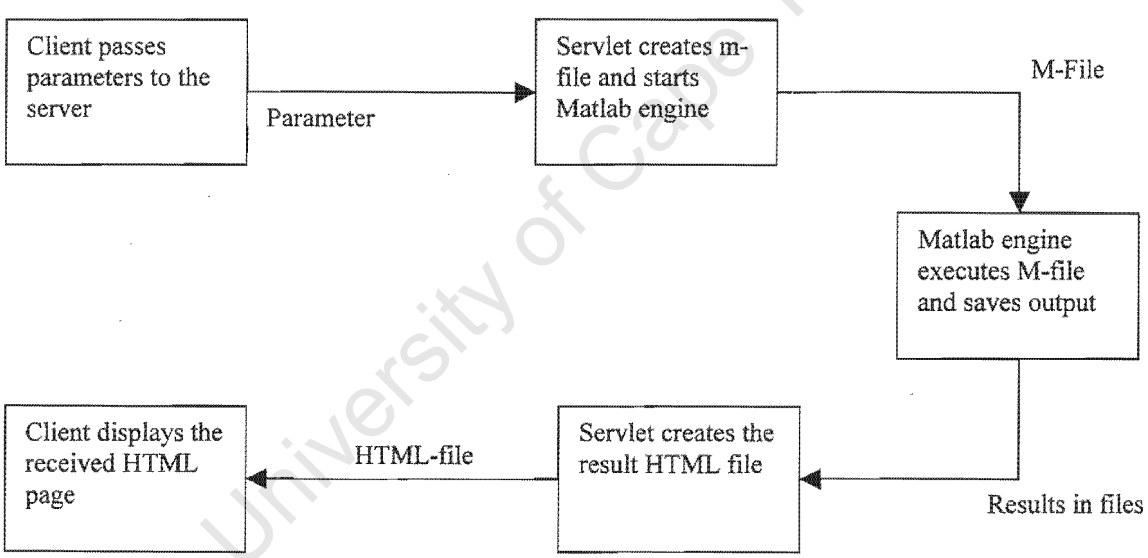


Figure 21: Data Flow Diagram

Utilising this approach, the Servlet is creating a Matlab m-file according to the simulation and the parameters (specified by the user). This Algorithm specifies not only the details of the simulation or the mathematical computation but also defines the format of the results (graphs, data). After saving that algorithm (m-file) to the file

system, the Servlet starts the Matlab engine through the Runtime class. The Matlab engine executes the m-file. The Servlet waits until the process is finished (the process notifies the parent process when it completes) and prepares the results as an HTML file and sends it back to the client. The data flow between components can be seen in the data flow diagram in Figure 21.

Technically this is not the finest design, but with this strategy basically any software program can be included for a remote access and be utilised by students.

6.12 System Database Module

During the development of the system, because of the amount of data, the need for a powerful database was not given. To test the application server prototypes, a simple text file solution was chosen and proofed to be sufficient. For an effective running Web Based Distance Learning system, a database management system is necessary. To keep track of all users, user details (contact details, credits, marks...) and the messages of the bulletin board or discussion forum a form of data management and organisation is required. Also the search and response time is much faster with a sophisticated database management system then the text file option.

During the development period, every server application that had data to store or to access, created its own file on the file system. For example, the Discussion-Group Servlet keeps files for the messages and the access control system keeps files of the users with their passwords. These files are all in ASCII format that guaranties that the access of the files for manipulation through the administration can be done with a simple text editor. Also the transfer of the files into a database format for a specific database product can be done without great problems. With the object oriented design and implementation in the Java programming language, the change of the objects handling the data input and output can be easily implemented without great changes to the program structure.

The choice of database management system is largely dependent on the financial possibilities and the chosen hardware/operating-system environment.

6.13 The Help System

The Help System is based on dynamic Web Pages. According to the system status when a request to the Help System is made, the information on the HTML page contains general information to help in the process of navigation and the use of all features available at this point in the system structure. It also contains specific help to the topic that is being worked on. For example, help where to find more information and who to contact for information about that specific topic.

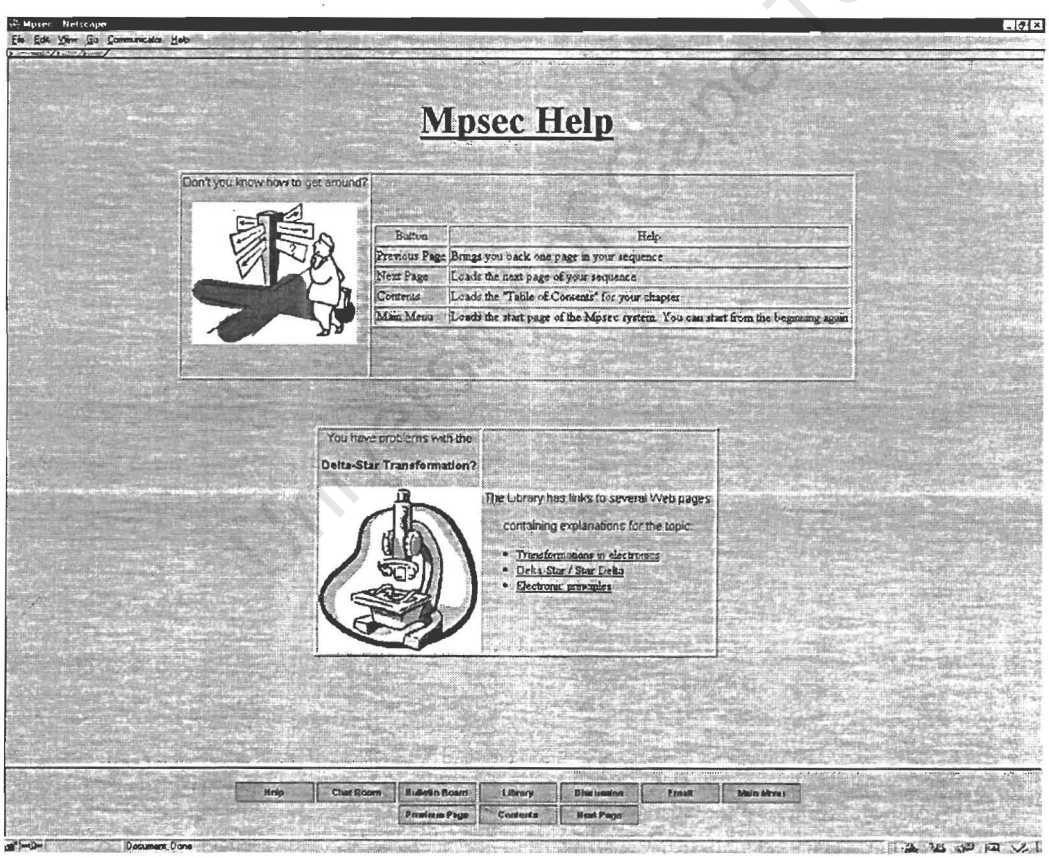


Figure 22: Help System GUI

By activating the Help through the toolbar, a HTTP request to a Servlet with all the system status information is sent to the Web Server. The Servlet processes the status information and creates the help information in HTML format and sends it back to the client. Figure 22 shows an example of a Help System Web Page.

7 Miscellaneous

7.1 The Development Environment

The system is developed and tested in an environment, which is dependent of the available resources for the project. The development environment is certainly not sufficient to run the distance-learning course in full scale; it enables the development and testing of the modules. The structure of the environment changed in three phases:

- Phase 1: Development of prototypes of all modules on a single machine.

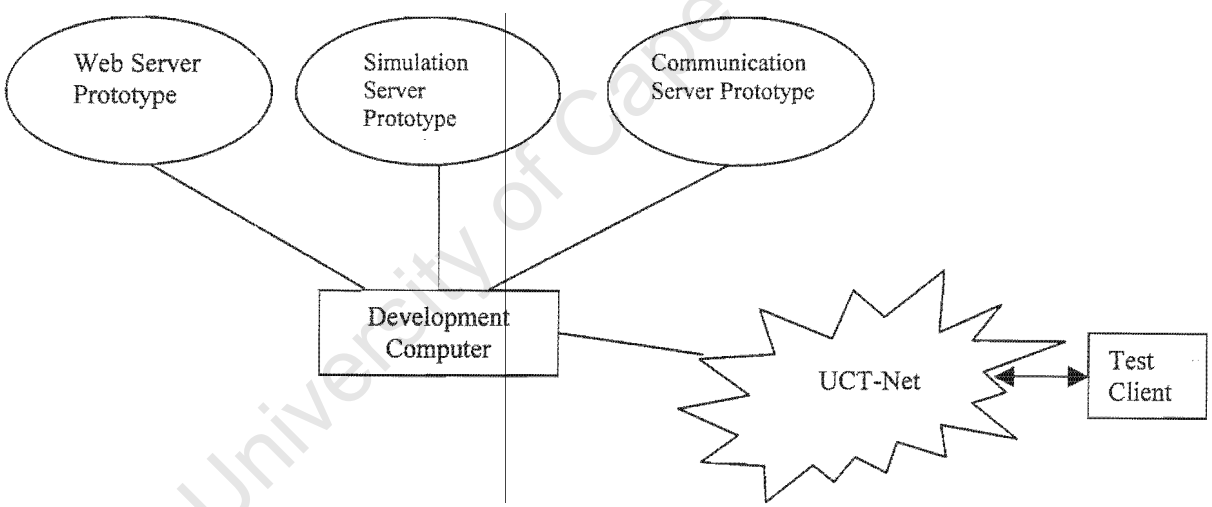


Figure 23:Module Distribution In The Prototype Environment

The Figure 23 shows that all modules are residing on one computer. The development computer and the computer running as Test Client are connected to the UCT network.

- Phase 2: distributing the modules to dedicated machines.

As can be seen in Figure 24, the different modules are residing on independent computers:

Course module --> Faculty Web Server; the faculty Web Server offered a faster access due to the slow Internet connection at the development location.

Simulation module --> Dedicated server; having the simulation engines running on a dedicated machine improves the performance and so the response time.

The other modules --> separate dedicated server

Each computer is directly connected to the UCT network. The Test Client can reside anywhere on the Internet.

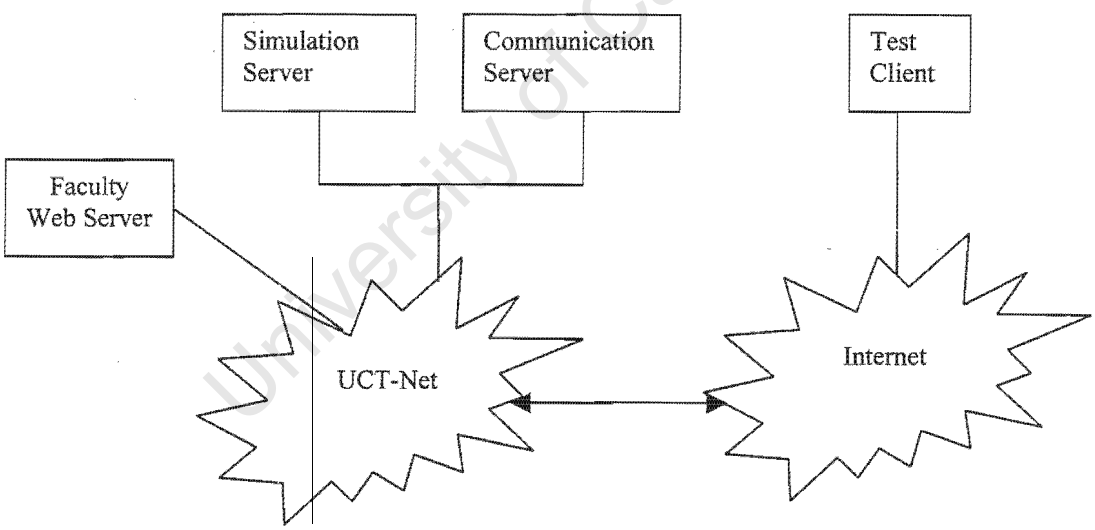


Figure 24: Module Distribution In The Development Environment

- Phase 3: deploy to a runtime environment

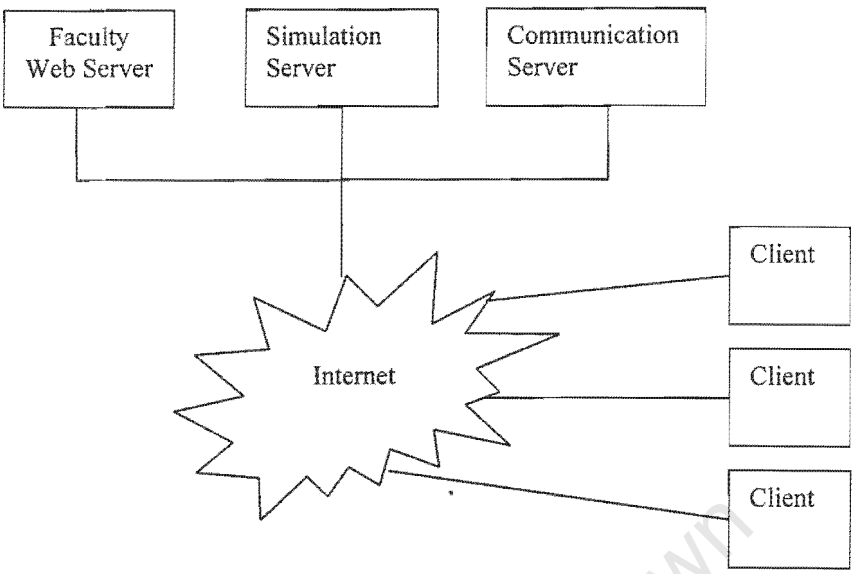


Figure 25: Module Distribution in The Runtime Environment

The difference between Figure 25, which shows a possible runtime environment, and Figure 24 is that in the runtime environment the servers are directly connected to the Internet.

The development was done under the Microsoft NT operating system. The computers were linked to the UCT network and therefore behind the UCT proxy-server. The Web Server is originally developed for the Unix operating system. The release for the Microsoft NT operating system is experimental, and is recommended only for experienced users. The Apache Group does not guarantee that this software will work as documented, or even at all. During the development there were no problems with the system at all and it did run stable and reliable.

7.2 The Runtime Environment

The modular design of the system allows a wide range of possibilities for the runtime environment. The choice of the operating system is a matter of preferences and maintenance. Only 21.81% (Netcraft, May 2000) of all Web Servers were running under the Microsoft NT operating system in May 2000 but the complexity of a Unix system might face the administration with problems. Most Unix system are more stable then Microsoft's NT and the free availability of the Linux system makes Unix an interesting option. The choice of hardware is restricted by the financial possibilities. Certainly there is no upper limit of investing in computational power. The lower limit is defined by the desired performance. If the system is restricted to a small amount of users, simple course notes and virtual Class Room components any standard computer will be sufficient. Any increase of users or components like remote simulations might lead to a demand of upgrading the computer up to a distributed approach of the system.

The network connection is the main issue for the runtime environment. The speed and bandwidth of the connection will decide over the efficiency and popularity of the courses.

7.3 Existing Software

Deciding what software to use for the Web Based Distance Learning system was based on the features available for the system, platform independency and its free availability. For example, the Apache Web Server is the most used Web Server according to the Netcraft Web Server Survey (Netcraft, 1999) and free to download (http://www.apache.org). Java on the other hand is a programming language (also freely available from http://www.java.sun.com), which enables platform independent development of server and client side application customized for the World Wide Web. Since during the software development circle, the final hardware system of the runtime environment was not known and because of the speed systems have to adapt to change (according to the demand and high rate of technology advances), it is very important to have a most flexible and platform independent system design. All implementations follow that idea and can, without or with only little change be transferred to any common hardware and operating system. Below is a list of the most important software components used and a description of them:

General:

Java(R) 2 SDK, Standard Edition Version 1.2.1

Java is a general-purpose, high-level programming language and a powerful software platform. It adds the functionality of a programming language to the Web Pages. For example, simulations could run on the client computer. In this case, the Java class files are downloaded and then executed within the browser instead of constantly downloading new data from the server. If necessary, results can be sent back to the server for further processing or data storage. This approach reduces the amount of network transaction and so is less vulnerable to network errors. Java is platform independent and code developed in Java is therefore able to run on all major computer platforms.

Server Side:

Web Server Software - Apache Web Server

The Apache Project is a collaborative software development effort aimed at creating a robust, commercial-grade, and freely available source code implementation of an HTTP (Web) server (Apache Project, 1999).
The Apache Web Server is available on several platforms including Windows NT and includes all features needed by the system:

- Java Servlet engine
- Security

Java Servlet Development Kid

Servlets are modules that extend request/response-oriented servers, such as Java-enabled Web Servers. For example, a Servlet might be responsible for taking data in an HTML application form and applying the business logic used in the application. Servlets are an effective replacement for CGI (Common Gateway Interface) scripts. They provide a way to generate dynamic documents that is both easier to write and faster to run. Servlets also address the problem of doing server-side programming with platform-specific APIs: they are developed with the Java Servlet API, a standard Java extension. In short, Servlets move the computation from the client side to the server side and adds the functionality of a centralised computer system, e.g. secure database access, less network transmission for sensible data etc.

Client Side:

Web Browser (e.g. Netscape Communicator, Microsoft Internet Explorer)

All facets of the projects Web system are tested with the two most common Web Browsers, to ensure availability of the course to the widest audience possible.

JFC / Java Plug-in 1.2

JFC is short for JavaTM Foundation Classes, which encompass a group of features to help people build graphical user interfaces (GUIs). Java Plug-in 1.2 enables Web Browsers to use the advantages of the JFC.

JavaScript

JavaScript is a script language, developed by Netscape. JavaScript uses some of the same ideas found in Java, the compiled object-oriented language derived from C++. JavaScript code can be imbedded in HTML pages and interpreted by the Web Browser (or client). They are used to add dynamic functionality to the static HTML pages. Both Microsoft and Netscape browsers support JavaScript, but sometimes in slightly different ways.

Email Service (e.g. Netscape Communicator, Microsoft Outlook, Eudora...)

Any common Email service ensures the communication between students and instructors. Where appropriate, links inside the pages invoke directly the Email software.

Multimedia Software (e.g. Quicktime, Mpec player)

These depend on the installation and browser preferences. Multimedia software is available as plug-ins and automatically started by the browser when demanded.

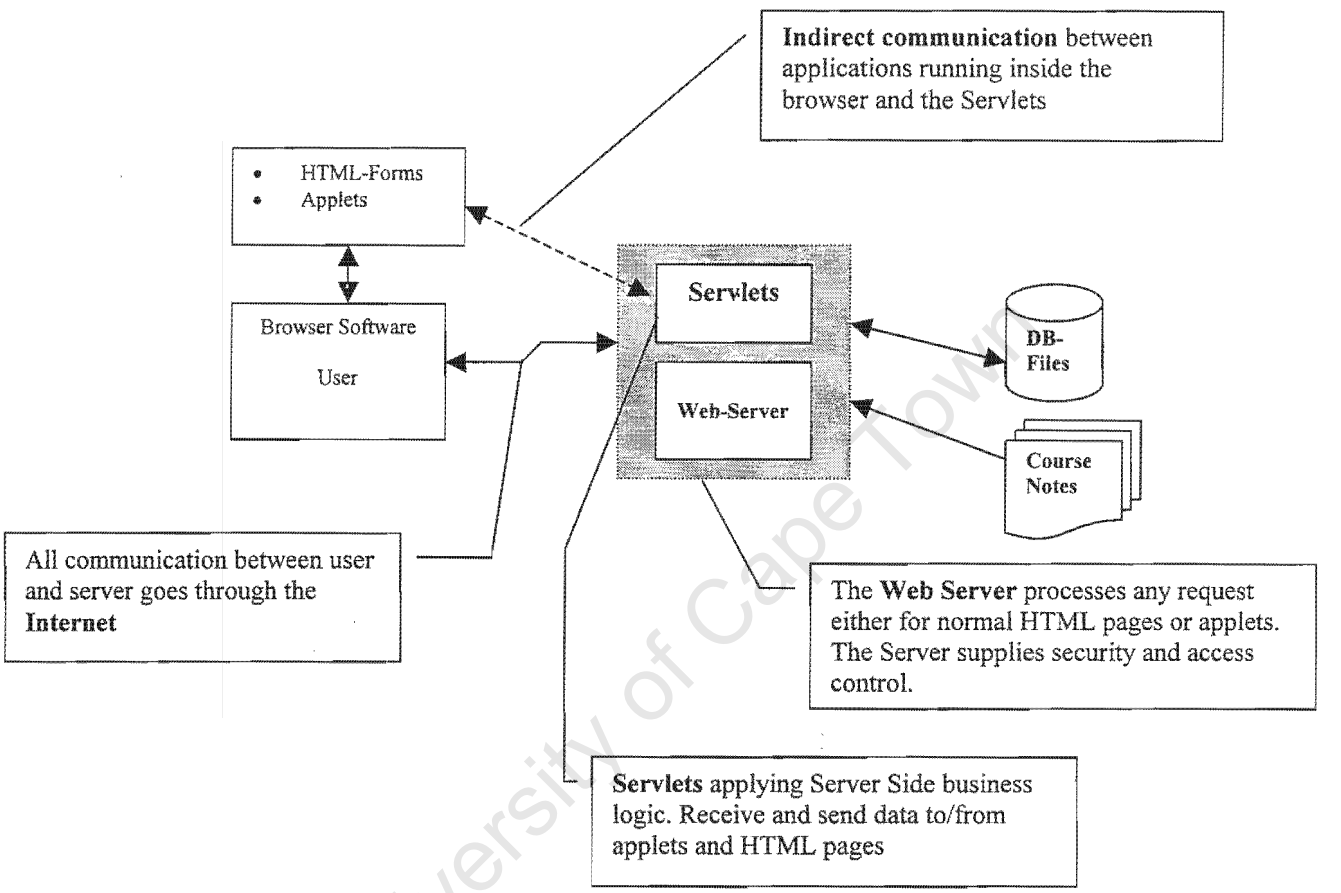


Figure 26: Software Components

The Figure 26 shows the software components, their task and relation to each other.

7.3.1 Problem with Existing Software

Browser - Software:

The two most common browsers (Netscape Communicator, Microsoft Internet Explorer) differ in certain aspects. For example, third party programs (Plug-ins) such as video players show up differently. So the appearance of certain functions will look different on each browser.

Java

The newest Java developments are not directly integrated into the browser software. This means that development with the newest Java standards will create problems with the browsers. For example the browsers from Netscape and Microsoft only support the Java 1.1 version. To run the newest Java standard (version 1.3) the browsers need to install plug-ins, which need to be downloaded and installed. Even if the plug-in needs to be downloaded and installed only once, it creates a risk of problems and errors. The development of the Java GUI for the client side was originally done with the Java 1.2 version with the hope that a new browser generation will incorporate the new standard. By the time of writing there was still no new browser software available and it will not be in the near future. The problems occurring with the Java plug-in to support the new Java version lead to the decision, to recreate the Java GUI's with the Java 1.1 version. Because of the object-orientated design of the software, this problem was easily solved. The new GUI might lack some up to date design features but efficiency and functionality has to come first for a software programme.

Also the Apache Server is only able to run Servlets developed with the JSDK 2.0 and not with the newest JSDK 2.1.

7.4 Security / Access Control

Being connected to the Internet, allowing arbitrary computers to connect to your system is a frightening idea for every system administrator. To secure your own system is a very important factor, restricting access to certain areas another one. A secure system must block any attempt of gaining control over the operating system or parts of the computer not meant to be public but must allow access through an access control service.

Access control-based services must handle a variety of user scenarios and conditions. For example, users with knowledge of the URL who attempt to enter a section of the information space under access control must be informed that they do not have access privileges to the information. Certain areas are accessible for a specific group of users but restricted for everyone else. Users must be given the service to subscribe to the system.

7.4.1 Access Control Service

The Apache Web Server provides two ways of restricting access to documents, either by hostname (IP address) or by demanding the submission of username and password. The first option can be used to limit access within a company. In the scope of a Web Based Distance Learning environment, the users of the documents are widely dispersed and the administrator needs to be able to control access on an individual basis. If a first request for a document in a restricted area occurs, the Web Server will respond with an http-protocol client error (401 "Unauthorised") demanding authorisation. The browser software will ask the user (via pop-up window) for a username and a password and will include these in the repeated request. The Web Server checks the username and password and if they are valid for the restricted area, returns the page. The browser will send the user authentication on subsequent requests.

While authentication does allow resources to be restricted to particular users, there are potential security issues:

- The username and password is as secure as any username/password system, in that end-users should not tell others their password or write it down or make it easy to guess.
- The Apache Server authentication scheme transmits passwords across the Internet unencrypted, so they could be intercepted.
- No control over spreading the username and password to different users to avoid single registration.

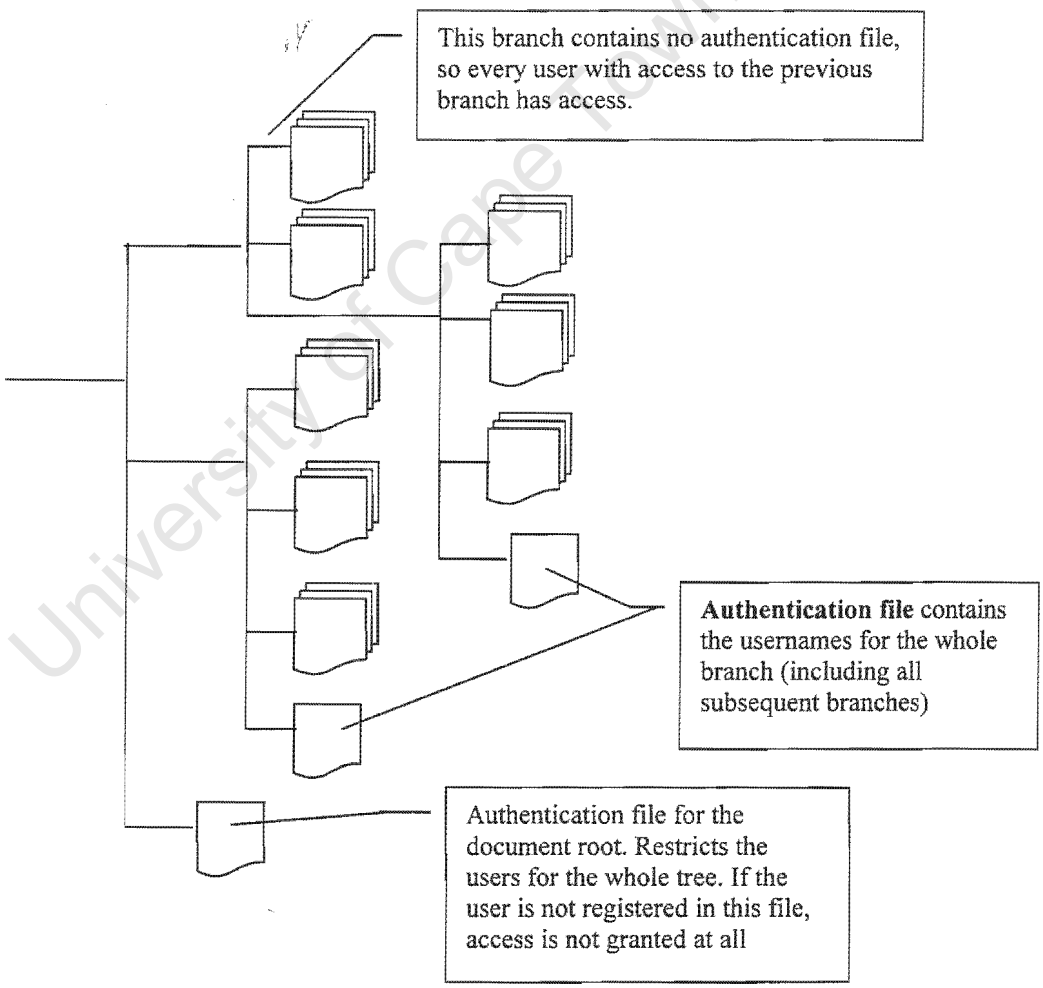


Figure 27: Document Tree Example

By structuring the documents (as can be seen in Figure 27) into a document tree (subsequent series of directories) allows each branch of the tree to be a separate restricted area by placing an authentication file into it. The authentication file contains the valid usernames for the specific branch of the document tree. The server will check every request with this file and allow or reject the request accordingly.

To allow a user access to a restricted area of the system, the system needs to know the identities of all the users who are allowed to the system. This information is stored in the system database. Users have to apply for access via an application form (HTML form stored in a public area of the server). The application has to be processed and accepted or rejected by the system administration.

7.4.2 User Information for Customised Web Pages

To enable customised Web Pages, the system has to know what user is making the request. Since the Web Server software, which handles the Access Control is an independent software tool, accessing the user information is not a simple task. One solution would be to ask the user directly via the GUI and to store the information in the system status variables. But having two independent authorisation processes, one for the Access Control and one for the status information is not acceptable. In this implementation, the server application for the "Main Menu" retrieves the user information from the HTTP header and saves it in the system status variables of the Toolbar (see 6.3 The User Interface). Since the browser software only includes the authentication into the HTTP header when requesting a resource in a restricted area, the Servlet for the "Main Menu" has to be located in a restricted area.

7.5 Distributed Model

Depending on the final system requirements, available infrastructure and funding, two possible solutions for the Web Based Distance Learning System were designed.

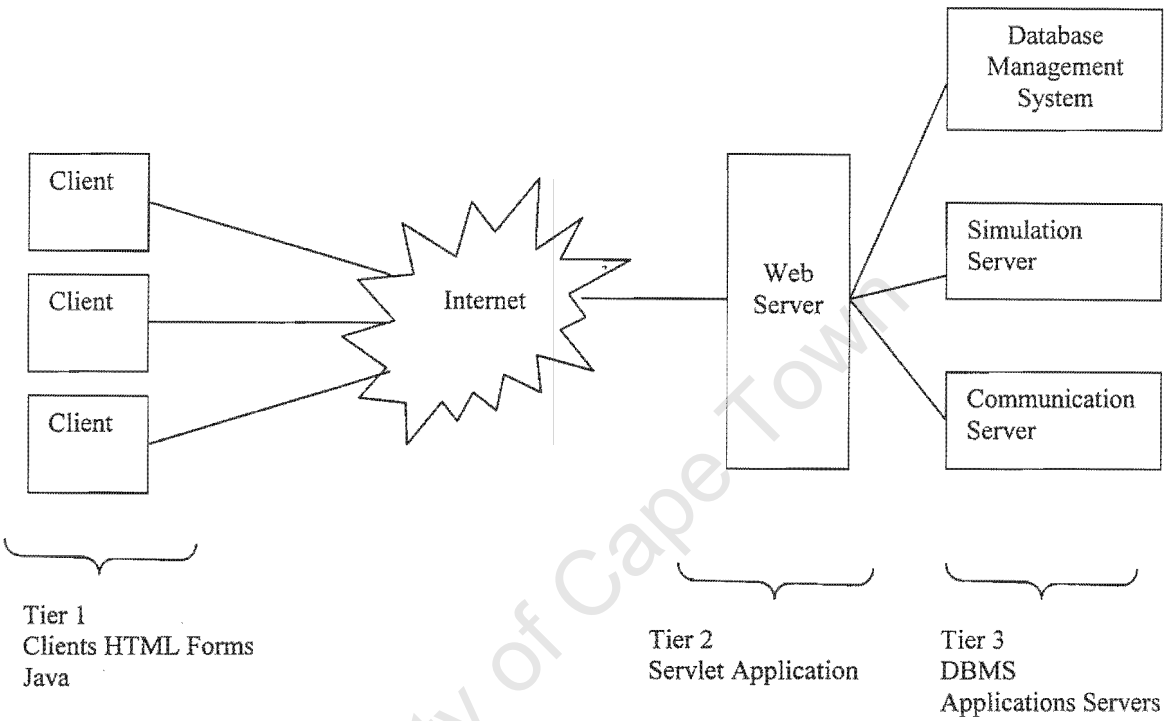


Figure 28: 1. Distributed Model

One efficient design is the distribution of the system modules onto several computers connected by a local area network (LAN) as can be seen in Figure 28. The LAN is connected to the Internet through a gateway. This design focuses on fast communication (depending on the capability of the LAN) between the system modules. The traffic on the network only consists of the necessary communication between system modules. No redundant information is slowing down the network. Dedicated computers running the applications for a specialised task enable a fast response time (depending of the computational power of the computer and algorithm). Only the clients HTTP request and the final HTTP response are going

through the Internet and are depending on outside factors, like service provider and backbone bandwidth. The typical 3-Tier design of the system structure (see 6.2 System Structure) allows an easy implementation on this environment. A 3-Tier Client Server architecture is all about how to split the client server application into functional units and assign the units to the correct place in the environment. The Tier 1 is the front end, containing the GUI. The Tier 2 contains the business logic; the business logic controls the traffic that links the clients with the different server applications and the database. The Tier 3 is the set of server applications and database management system. System performance is not the only advantage of a 3-Tier architecture; scalability, stability, control and administration are also significantly enhanced.

The second design (Figure 29) is the architecture used in the second development phase (see 7.1 The Development Environment), The advantage of this design is the possible incorporation of existent infrastructure and the wide distribution of computers over large distance, possibly world-wide. For example, Web Servers all around the world can be included into the system to host lecture notes of instructors based at the server's location. Application Server for simulations that already exist at universities or in the private sector can be included without the client noticing it. The example is also valid for the first design, but the disadvantage is the traffic of the Internet. All communication between the modules is transmitted over the Internet and the efficiency of the design is depending on the workload of the Internet. This could lead to a very slow response time of a server request. Also maintenance and administration is more complex but a system failure at one location does not lead to a total breakdown of the system. All other modules can continue to work normally. As for the first design, the 3-Tier design of the system structure makes the implementation easy and not transparent for the user.

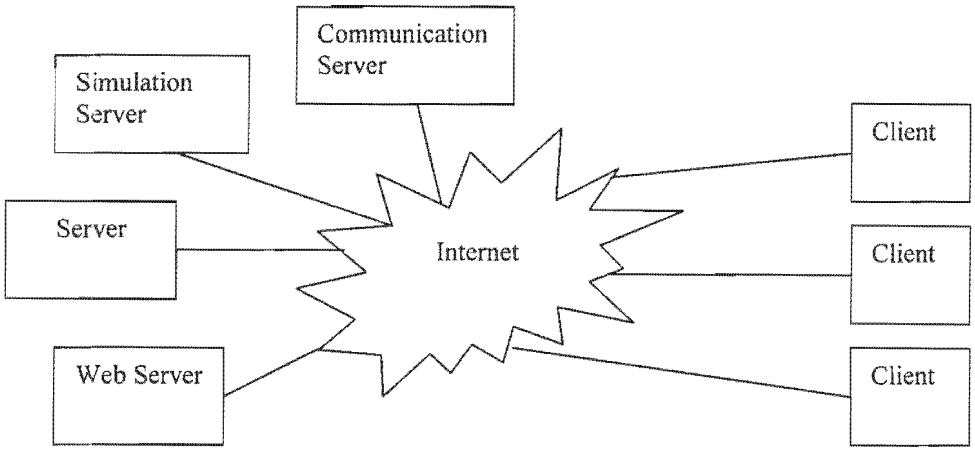


Figure 29: 2. Distributed Model

7.6 Maintenance and Administration

Administration involves more than only keeping the system up and running by rebooting it after a system failure. The Access Control Service needs administration like user registration, updating password files etc. Also do users gain and loose access to certain restricted areas, for example to write exams during a specific time period and the authentication files need to be updated accordingly.

To maintain an up to-date knowledge of the system users and their status, the registration files, which hold the personal information of the users (including the exam results, registration limits, payments etc), have to be updated regularly. Users whose right to access the system or certain areas expired have to be removed from the password file and/or the authentication files. The results of exams or assignments have to be added to the files.

The lecture notes have to be updated and revised on a regular basis. New information and new perceptions have to be added as well as changing the way of presenting the information according to feedback and experience throughout the

course. Also the technology of presenting information on the WWW changes at a rapid speed. Updating the system software according to new technology standards, like a new version of HTTP Protocol or even a new Internet Protocol (IP6), and updating the information towards new HTML standards or other presentation standards is absolutely necessary for having the system functioning and running. Being attractive and convenient for the user is a very important maintenance issue that should not be neglected.

Maintenance of the Web Server software is a continuous task; monitoring and administrating the system log files, reacting on failures and problems. This could involve changing the workload for certain computers by redirecting the task onto other computers or even removing certain components.

8 Conclusion

In this thesis, work was carried out to develop a Web Based Distance Learning environment for power system education. The following contains the conclusion that the author has drawn on the basis of the information contained in this thesis and the readings made by the author.

A Web Based Distance Learning course overcomes the disadvantages of conventional distance learning methods by offering faster interaction, multimedia and by creating a real learning community. It reduces the cost of learning by being independent of location and time. In the case of further education, the costs for travelling, accommodation and the absence from work are saved and the asynchronous nature of a Distance Learning course accommodates the individual time restrictions.

"For many years, visionaries and futurists have been telling us that one-day we would have quick and easy online access to all of the world's information. Well, the future has arrived; [it's] called the World Wide Web (WWW) and its growth in the past few years has been phenomenal. Already the Web is showing us how global networks will transform education" (Greg Kearsley, 1996). The WWW or more precisely the Internet is only a medium that can be utilised to transform education. The Internet is definitely not transforming education on its own. This has to be done by the institution that offers education; the Internet is only offering a wide range of opportunities. A first motivation for offering a course on the WWW might be: "I put it on the Web and then I save time and work". But another medium to present information does not replace teaching. And as it is general knowledge today, a computer does not reduce the workload, it just shifts it towards other tasks. A Web Based Distance Learning course needs a large degree on maintenance work. The WWW is a fast changing medium and to have a successful and effective learning environment the course has to change fast as well. The virtual classroom

components live through the interaction and the accuracy of the information available. Nothing is more disappointing then a question placed in a forum and not receiving an answer for a long time period or browsing through notes and messages that are outdated for a long time. To have a successful Web Based Distance Learning, the involvement of the lecturer is more likely to increase as to decrease. The idea of the virtual community allows even more individual guidance and interaction, and to ensure that, it might need even more attention from the lecturer then a conventional course would need. But putting all the potential together, the learn-effect promises to be greater and worthwhile the effort.

The system developed in this thesis follows the idea of instructing rather then just presenting. Therefore the communication is most important and the implementation pays tribute to it by an extensive virtual classroom and personal guidance tools like the online tutorial and the customised Web Pages. The personal and individual appearance for every user is a major advantage and the difference towards other Web Based Distance Learning sites. Each user experiences a personally customised environment that guides the user through the learning environment according to the users personal needs. If commercial Web Portals use the same principle to target the user with specific products and advertisement, the principle must be good enough for educational systems to target the audience more precisely. The user receives more personal attention from the lecturer through the virtual classroom environment and has more personal interaction with the lecturer through the direct communication, whereas in a conventional classroom the instructor's answers are aimed at the whole group of the attendees and not towards an individual.

"Think big, start smart, scale fast" is a good premeditation for the design of an Internet Application. What is if the demand for the system exceeds all expectations and predictions? The initial demand and success of an Internet site can sink rapidly down to zero if the site is not able to fulfil its task because the system fails under the workload or administration failure. On the other hand, great investment into

hardware and software for a new system is very risky because of the unpredictable outcome. And very likely there is a lack of funding for a start-up project. So it is advisable to design the system for a big demand, implement a reasonable sized environment and be prepared and be able to upgrade the system very fast to accommodate any demand. The modular design of the system as a logical 3-Tier structure follows exactly this premeditation. The tools and languages used for the implementation are all free available and platform independent and therefore easily transported onto any runtime system, changed, adapted to a new environment and new requirements and reproduced by any administrator.

The ultimate effectiveness of the chosen design has still to be proofed. Eskom (electricity supply company of South Africa) initiated the project and supports it financially. Eskom intends to use the developed system to offer further education to their employees and therefore influences the contents of the course. Because the course content was not part of this thesis and as this part of the project only slowly starts to develop at the end of this thesis's timeframe, an effective testing of the system is not feasible. The focus of the system is on communication and that will only reveal its full potential and effectiveness as a learning environment when there is a full learning community involved. Therefore the decisions were based on the author's experience with the WWW as a research medium, his recent experience as a student himself and the research results found in publications.

Finally the author concludes that the implemented system meets all requirements to create an effective learning environment. In spite of project's financial limitations (no funding for commercial software products), the modular design and the 3-Tier architecture allows scalability and the possible expansion if need arises.

9 Biography

Anil Aggarwal — "Web-Based learning and teaching technologies: Opportunities and Challenges", Idea Group Publishing, Hershey USA, 2000

Apache Project — "About the Apache HTTP Server Project". Online. Available:http://www.apache.org/ABOUT_APACHE.html February 1999.

Bishop, Judy — "Java Gently", Addison-Wesley, England, 1997

Clyatt, Bob — "Web-Based Distance Learning: Tool for Change", Online. Available: http://www.comproj.com/wbdl.htm 1998.

Downes, Stephen — "Web-Based Courses: The Assiniboine Model", Online, Available: http://ww.westga.edu/~distance/downes22.html 1997,

Filipczak B. — "Putting the learning into distance learning", Training, 1995

Foldoc — "Free Online Dictionary of Computing", Online Available: http://foldoc.doc.ic.ac.uk/foldoc/index.html , April 2000

Hill, Janette R. — "Distance Learning Environments Via the World Wide Web", Available: Web-Based Instruction by Badrul H. Khan, February 1997

Hilz S.R	"The virtual classroom: Learning without limits via computer networks". Norwood, 1994
Howard-Vital M. R.	"Information technology: Teaching and learning in the twenty-first century", Educational Horizons, 1995
James E. Goldman, Phillip T. Rawles, Julie R. Mariga	"Client/Server Information Systems", John Wiley & Sons, Inc. New York, 1999
Jonassen, David H. and Mandl Heinz	"Designing Hypermedia for Learning", Springer Verlag, New York, July 1989
Kearlsley G, Moore M.G.	"Distance education: a systems view", Wadsworth, New York, 1996
Khan, Badrul H.	"Web-Based Instruction", Educational Technology Publications, Englewood Cliffs, N.J., July 1997
Lampson, B.W.	"Distributed & Client/Server", Lecture notes by Prof. K.J. Macgregor. Page 2, 1998
M. Crow, A. Pahwa, S.Starrett, K. Olejniczak, S. Sudhoff	"Collaborative Distance Education in Power Engineering", IEEE Transaction On Power Systems, Vol. 15, No 1, February 2000

McCalley J, Ajjarapu V, De La Ree J, Phadke A, Sheble G, Venkata S and Vittal V. "Multimedia Courseware Sparks Interest in the Industry", IEEE Computer Application in Power, October 1998

McManus, Thomas Fox "Delivering Instructions on the World Wide Web". Online. Available: http://ecwf.cc.utexas.edu/~mcmanus/wbi.html January 1996.

Moore, Cookson and Donaldson "Contemporary issues in American distance education", Pergamon Press, New York, 1990

Netcraft , "The Netcraft Web Server Survey", Online Available: http://www.netcraft.com/survey/, July 1999

PowerLearn "PowerLearn Intro", Online, Available: http://powerlearn.ee.iastate.edu/main/intro.html, June 2000

Schifter, Catherine C. "Teaching in the 21st Century", Available: The Internet and Higher Education Volume 1 Number 4, 1999

Shotsberger "Instructional uses of the WWW", Exemplars and precautions. Educational Technology, 1996

W3C "Hypertext Markup Language", Online Available: http://www.w3.org/MarkUp/, April 2000

Wang, Paul S. "Java with Object-Oriented Programming and World Wide Web Applications", PWS Publishing, California, 1998

Whatis.com "Whatis.com", Online, Available: http://www.whatis.com, May 2000

Willmann Sven "Development of a Digital Library in content of a Multimedia Power System Engineering Course (Mpsec)", Diplomarbeit at the FH Konstanz, January 2000

10 Appendix

10.1 Navigation System

List	This variable is a JavaScript array. The items in this array are the filenames of all HTML pages of a chapter in the recommended sequence. When the first page (table of contents page) of a chapter is loaded, the array is initiated with the chapter specific items.
Index	The *index* variable points to the item in the *list* array that is currently loaded and visible in the main frame of the system.
codebase	Codebase contains the absolute path to the files in the *list* array. Having the pathnames separate from the filenames in the *list* array supports the maintenance of the modular system structure. Relocating the chapter in the file system structure only demands a change of the *codebase* variable. The adjustment is only necessary once and only in one file.
contents	This variable contains the absolute path plus the file name of the start page (table of contents page) of the current chapter.
tutorial	This variable is a Boolean marker. If the variable is set, the tutorial module is active and any change in the navigation is broadcasted to the Tutorial server.
loginname	Loginname contains the name of the user logged on to the system. The user authentication is requested by the Web Server access control system. Every time a user specific request to the Server is made, the loginname is sent with the HTTP request as a parameter. According to this parameter the Server response is customised for the specific user.

The navigation system control script (toolbar):

```
<script language="JavaScript1.2">

var list, index, codebase, contents, tutorial, loginname;

list = new
Array("http://196.7.12.131/wservlets/LoginServlet");
contents = list[0];
index = 0;
tutorial = 0;

Normal = new Array(10);
Highlight = new Array(10);
Normallist = new Array("help1.gif", "chat1.gif",
"bulletin1.gif", "library1.gif",
"discussion1.gif","Email1.gif", "main1.gif",
"prev1.gif","contents1.gif", "next1.gif" );
Highlightlist = new Array("help2.gif", "chat2.gif",
"bulletin2.gif", "library2.gif",
"discussion2.gif","Email2.gif", "main2.gif",
"prev2.gif","contents2.gif", "next2.gif" );

for ( var i=0; i<10;i++ ){
   Normal[i] = new Image();
   Normal[i].src = Normallist[i];
   Highlight[i] = new Image();
   Highlight[i].src = Highlightlist[i];
}

function pageload( n ) {

  if ( n < 0 ) {
     if ( index > 0 ) {
        index--;
        parent.main.window.location =  codebase +
list[index];
     }
  }
  else {
     if ( index < list.length -1 ) {
        index++;
        parent.main.window.location = codebase + list[index];
     }
     else {
        parent.main.window.location = "end-of-module.html";
```

```
        index = list.length;
      }
   }
   if (tutorial == 1 ) {
      document.StudTut.sendUpDate(
"http://196.7.12.131/mpsec/"+codebase+list[index] );
   }
}

function library(){
   alert( loginname);
   parent.main.window.location =
"http://196.7.12.131/mpsec/bib/bib.html";
}

function mc() {
   list = new
Array("http://196.7.12.131/wservlets/LoginServlet");
   index = 0;
   parent.main.window.location = list[index];
   contents = list[0];

}
function mail() {
//   parent.main.window.location = "http://www.cnn.com";
     document.StudTut.requestLogin("module3");
     tutorial=1;
}
function inhalt() {
   index = 0;
   parent.main.window.location =  contents;

}

function help() {
   parent.main.window.location = "mpsec_help_system.html";
}
function Bildwechsel( Bildnr, Index )
{
   if ( Index == 1 ) {
      document.images[Bildnr].src = Highlight[Bildnr].src;
   }
   else document.images[Bildnr].src = Normal[Bildnr].src;
}

function appletload( n ) {
```

```
  if ( n == 1 ) {
     document.Bulletin.startBulletin();
  }
  if ( n == 0 ) {
     document.Chat.startChat();
  }
}
function bulletinboard(){

parent.main.window.location="http://196.7.12.131/servlets/Bul
letinIndexServlet"
}
function discussion(){

parent.main.window.location="http://196.7.12.131/servlets/Arc
hivePageServlet"
}

</script>
```

10.2 Definition of a Petri-Net

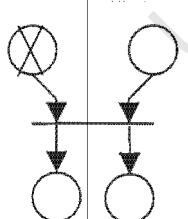
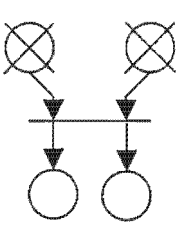
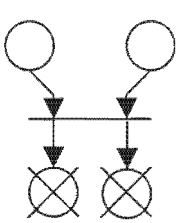
Petri-Nets are used to model parallel technical processes and their interaction. A Petri-Net defines important aspects of a parallel procedure but leaves details open for the later implementation.

Definition:

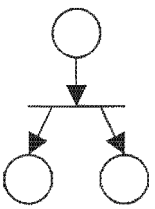
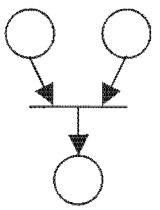
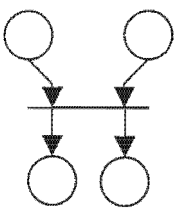
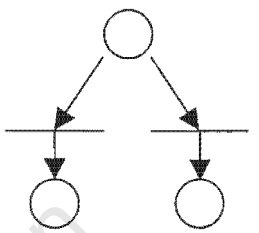
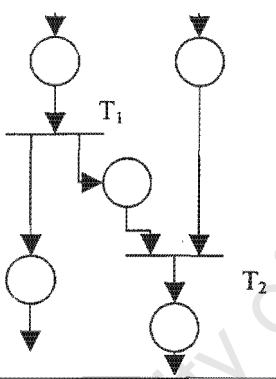
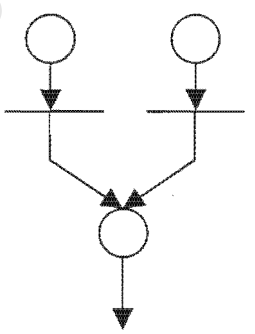
State	
Transition	
Access	
Current state	A marker on a state 

A transition switches if all entry states are set (a Marker in the state). If a transition switches, a marker is set to all following states.

Example:

Not switching:	Switching:	After switching:
		

Elements in a Petri-Net:

Division:		**Unification:** Continueing only if both entry states are set (AND)	
Synchronisation: (Rendezvous)		**Alternative:** (exclusive OR)	
Asymmetric synchronisation: T_2 only switches if both entry states are set		**Unification without wait**	

10.3 Chat-Room Petri-Net

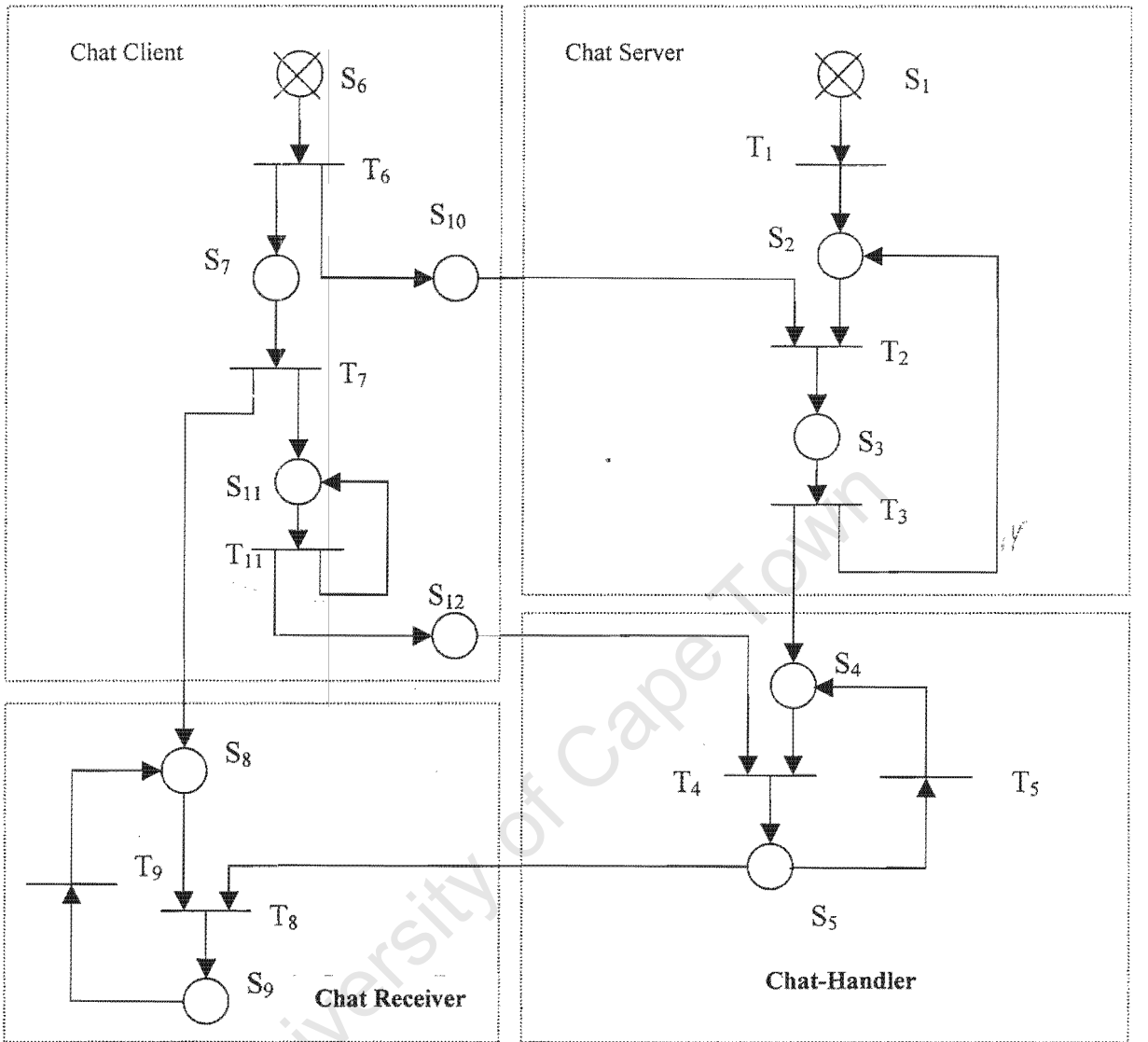


Figure 30: Petri-Net Of The Chat-Room Client/Server Interaction

S1	Start of Chat Server	S9	Display received message
S2	Listening on port for new client	S10	Send login to Chat Server
S3	Register client in list	S11	Wait for input from user
S4	Wait for message from a user	S12	Send input to Chat-handler
S5	Broadcast message to all users	T2	Transition fires if a new user connects
S6	Start of Chat Client	T4	Transition fires if client sends a message
S8	Wait for message from handler		

10.4 Tutorial

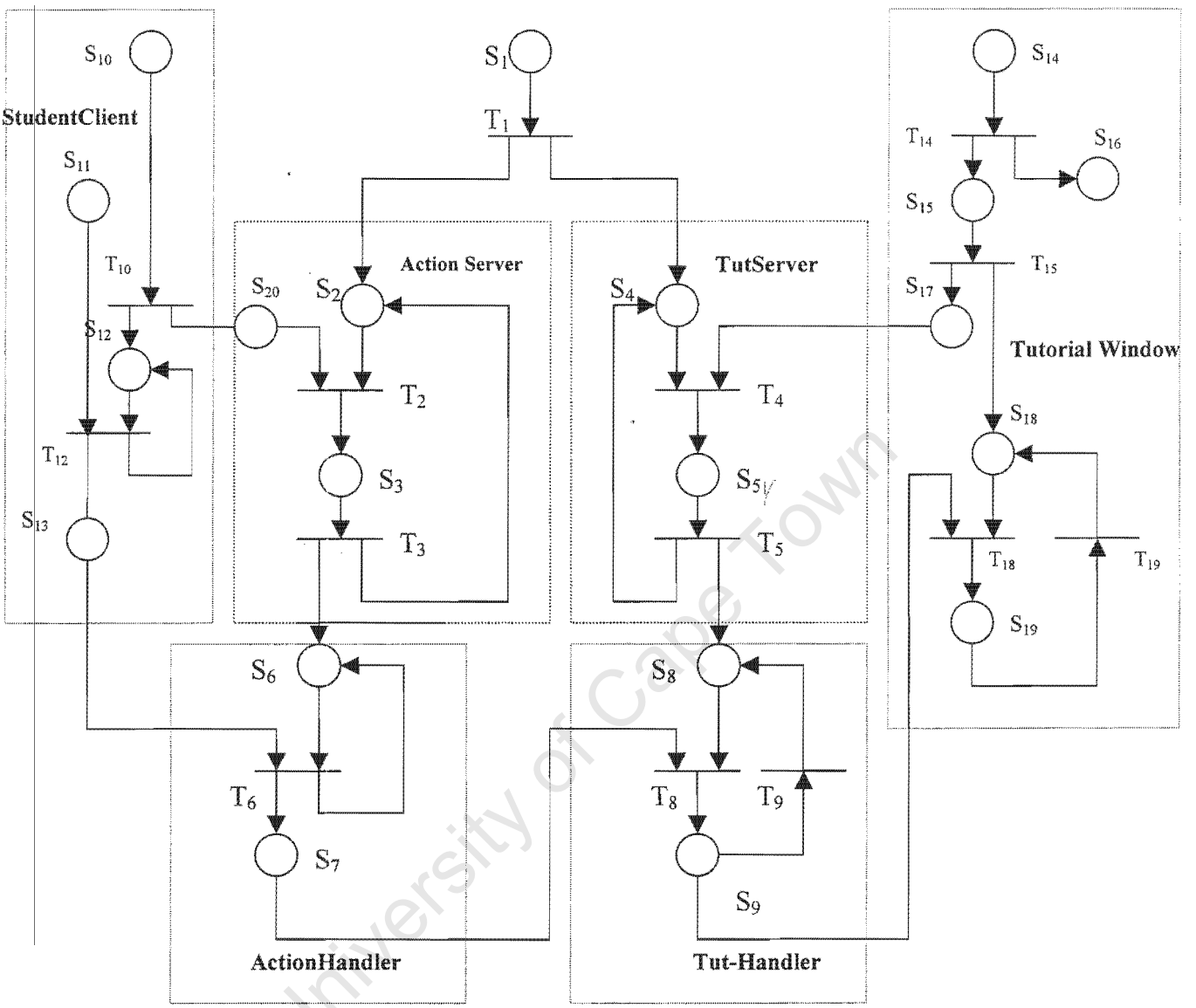


Figure 31: Petri-Net Of The Tutorial Client/Server Interaction

S_1	Start of Tutorial-Server	S_{14}	Start of Tutorial Window
S_2/S_4	Listening on port for new client	S_{16}	Start the Chat Client (see Chat-Room)
S_3/S_5	Register new client in list	S_{17}	Send login to Tutorial-Server
S_6	Wait for notification from client	S_{18}	Wait for update from Server
S_7	Place change into semaphore	S_{19}	Update the information in the GUI
S_8	Wait for semaphore	S_{20}	Send login to Tutorial Server
S_9	Send update to tutorial window	T_2/T_4	Transition fires if a new user connects
S_{10}	Start of the Student-client	T_6	Transition fires if client sends update
S_{11}	Notification from the browser	T_8	Transition fires if semaphore is set
S_{12}	Wait for notification from browser	T_{12}	Transition fires if browser sends notification
S_{13}	Send update to Server	T_{18}	Transition fires if Server sends update