

Virtual Lego

By

Arfah Md Nor

Dissertation submitted in partial fulfillment of
the requirements for the
Bachelor of Technology (Hons)
(Information Communication Technology)

JUNE 2006

Under supervision of Mr. Mohamed Nordin Zakaria

Universiti Teknologi PETRONAS

Bandar Seri Iskandar

31750 Tronoh

Perak Darul Ridzuan

t

T

385

A685

2006

1) Computer graphics
2) virtual reality

CERTIFICATION OF APPROVAL

Virtual Lego

By

Arfah Md Nor

A project dissertation submitted to the
Information Communication Technology Programme
Universiti Teknologi PETRONAS
in partial fulfillment of the requirement for the
BACHELOR OF TECHNOLOGY (Hons)
(INFORMATION COMMUNICATION TECHNOLOGY)

Approved by,



(Mr. Mohamed Nordin Zakaria)

UNIVERSITI TEKNOLOGI PETRONAS
TRONOH, PERAK
JUNE 2006

CERTIFICATION OF ORIGINALITY

This is to certify that I am responsible for the work submitted in this project, that the original work is my own except as specified in the reference and acknowledgments, and that the original work contained herein has not been undertaken or done by unspecified sources or persons.



ARFAH MD NOR

ABSTRACT

The report, entitled Virtual Lego, reveals the route taken of exploring the technology of Virtual Reality (VR) in implementing a 3D virtual reality toy, begins with the project planning and ends with the results and recommendations of the end-product. The main objective of this project is to develop a prototype of Virtual Lego that is capable of generating and sharing users' ideas in assembling, disassembling and reassembling the basic Lego blocks inside a virtual world via a networked virtual reality. The greatest challenge to Virtual Lego is to provide the users with a sense of immersion and realism throughout their navigation and manipulation processes of the virtual objects inside the 3D scene. Not only a mean in experiencing the beauty of virtual reality, Virtual Lego is also introducing the VR tools and giving opportunities for its users to hands-on the special and high technology VR input device. The methodology consists of four phases including the research and fact-finding, system planning and design, development and testing, and project closing and submission. Virtual Lego is seen to be capable of exploring and addressing the need of new and effective approaches for educational and entertainment, regardless to the level of the ages.

ACKNOWLEDGEMENT

Virtual Lego would not have been accomplished successfully without the helps and guidance from certain individuals throughout the process of its development and implementation. Here by, I would like to take this opportunity to express my appreciations to those who have contributed tremendously by pouring suggestions, positive criticisms and any means of help for the success of this project.

First and foremost, utmost gratitude goes to the Almighty God for giving me the blesss and strength to complete this project with flying colors. I also would like to thank my parents, Mr. Md Nor Abdullah and Mrs. Zaharah Zainal Abidin, for their full encouragement and continuous support during the time I spent in developing this project.

I am also grateful to my supervisor, Mr. Mohamed Nordin Zakaria, who constantly monitors my progress throughout the project's lifetime, for his valuable knowledge and guidance in making this project a reality. Without his helps and courage, I would have faced difficulties to reach up into this stage.

Special thanks to Mr. Raski, the IT Technician, for helping me a lot in handling and setting up the VR equipments especially during the time I was completing my project inside the VR Lab.

Last but not least, to my very supportive housemates and fellow friends who have given me the greatest support until the completion of this project, thank you very much to all. I really appreciate their concerns and kindness, which indirectly constitute to the success of Virtual Lego.

TABLE OF CONTENTS

CERTIFICATION OF APPROVAL	i
CERTIFICATION OF ORIGINALITY	ii
ABSTRACT	iii
ACKNOWLEDGEMENT	iv
LIST OF FIGURES AND TABLE.	vii
ABBREVIATIONS	ix
CHAPTER 1 :	INTRODUCTION								
	1.1	LEGO	1
	1.2	What is Virtual Reality?	4
	1.3	Introduction to Virtual Lego	7
	1.4	Report Overview	9
	1.5	Project Timeline	10
CHAPTER 2 :	LITERATURE REVIEW / THEORY								
	2.1	Introduction to Virtual Reality	11
	2.2	Immersion and Presence	11
	2.3	Interactive Design of Virtual Reality	12
	2.4	Virtual Displays and Interfaces	13
	2.5	Structure Elements in virtual Environment	14
	2.6	Stereoscopic View and HMD	15
	2.7	Force and Haptic Feedback	16
	2.8	Navigational Issues	17

2.9	Communication Support in Virtual Reality	18
-----	--	----

CHAPTER 3 : SYSTEM ARCHITECTURE

3.1	System and Software Design	19
3.1.1	System Architecture	19
3.1.2	System flow diagram	21
3.1.3	Unified Modeling Language (UML) Use Case Diagram	24
3.1.4	User Interface Design	26

CHAPTER 4 : RESULT AND DISCUSSION

4.1	Primitive and Virtual Environment	27
4.1.1	Development of Primitive	28
4.1.2	Development of Virtual Environment	29
4.2	Networked Virtual Reality	30
4.3	Collision Detection	33
4.4	User Interaction and System Navigation	34
4.4.1	Virtual Task Listing	35
4.5	Development of Meaningful Objects	36

CHAPTER 5 : CONCLUSION AND RECOMMENDATION

5.1	Conclusion	37
5.2	Recommendations	38

REFERENCES	41
-------------------	-----------	----

LIST OF FIGURES AND TABLE

List of Figures

- Figure 1.1 Varieties of LEGO bricks and minifigures in the market
- Figure 1.2 Big and light-weight LEGO bricks for children
- Figure 1.3 A LEGO Construction Plan
- Figure 1.4 A head-mounted display
- Figure 1.5 A Binocular Omni-Orientation Monitor (BOOM)
- Figure 1.6 A Cave Automatic Virtual Environment (CAVE)
- Figure 1.7 A Data Glove and a Joystick
- Figure 1.8 A stadium created by VR
- Figure 1.9 An application used by doctors to treat patients who are phobia to height
- Figure 1.10 The portion of the Gantt Chart for Virtual Lego
-
- Figure 2.1 The Lego components lay out on the parts grid
-
- Figure 3.1 System Architecture that illustrates the client side and server side for Virtual Lego
- Figure 3.2 System Flow Diagram for Virtual Lego
- Figure 3.3 UML Use Case Diagram for Virtual Lego
- Figure 3.4 User Interface Design for Virtual Lego
-
- Figure 4.1 The Lego blocks
- Figure 4.2 The Landing Page of Virtual Lego
- Figure 4.3 Concurrent view observed by each user who is dispersed in locations
- Figure 4.4 The portion of code that contain the structure of message to be send by each user to differentiate the actions made inside the application

- Figure 4.5 Rotation made on the Base block only can be viewed by the user who performs the action
- Figure 4.6 Collision detections successfully avoiding the Lego blocks from interpenetrating with each other.
- Figure 4.7 1 x 2 Block collides with 2 x 3 Block
- Figure 4.8 The portion of code for collision detection
- Figure 4.9 The portion of code that generate the keyboard functions
- Figure 4.10 The house that has been successfully developed inside Virtual Lego

List of Table

- Table 1 The virtual task listing of keyboard and pinch gloves for Virtual Lego

ABBREVIATIONS

VE	Virtual Environment
VR	Virtual Reality
HMD	Head Mounted Display

CHAPTER 1

INTRODUCTION

1.1 LEGO

The world's favorite construction toy, LEGO, is characterized by plenty of standard bricks, minifigures, gears and other pieces to create varieties of models such as buildings, ships, planes, trains and almost anything imaginable. The bricks are a wide range of special elements in lots of lively colors (*Figure 1.1*). LEGO has become a medium to express creative ideas and imaginations, not only for children, but also for the adult fans from each part of the world, who appreciate the uniqueness in assembling the bricks and means that lay behind the constructed models.

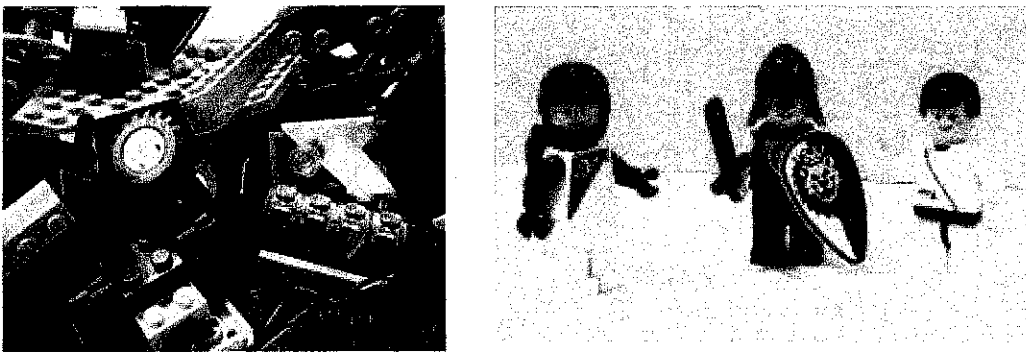


Figure 1.1 : Varieties of LEGO bricks and minifigures in the market

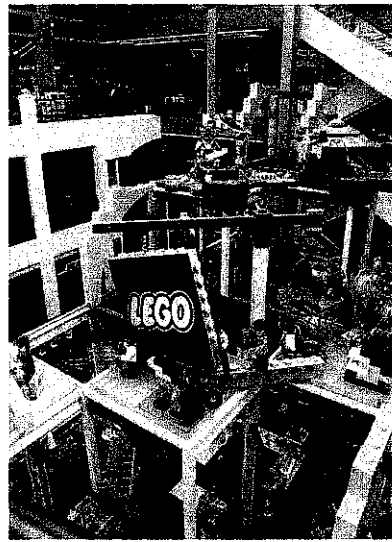
Each piece of the LEGO is manufactured from a strong and resilient plastic known as *acrylonitrile butadiene styrene*, or ABS. According to the history, this famous plastic brick was first created in the year of 1949 by LEGO Group who claims that the word 'LEGO', is derived from Danish phrase, *leg godt*, which means 'play well'.

The LEGO Group has, and is continuously develop pieces of bricks that ideally address the different needs of LEGO models' constructions for the youngsters and adults. Apparently, a range of activities in which all of senses are used for stimulating the creative and expressive skills, is needed for young children. Big and light-weight LEGO bricks allow the youngsters to build life-size models and use their bodies as they explore, create and play with the LEGO sets (*Figure 1.2*). Thus, these LEGO bricks are flexible, make them harmless and noiseless to the children.

By far, tiny and complex pieces of bricks are more preferable by the adults (*Figure 1.3*). They desire complex and highly imaginative construction processes of LEGO bricks that allow the creations of meaningful models, for an example, an airplane, completes with the tires and fixed-wing. In a mean time, the LEGO Group has released thousands of play sets themed around space, robots, pirates, dinosaurs, cities, holiday locations, racing cars, Star Wars, Harry Potter and more.



*Figure 1.2 : Big and light-weight LEGO bricks
for children*



*Figure 1.3 : A LEGO Construction
Plan*

LEGO is a suitable learning tool for children and a mean of entertainment of all ages. Yet, there is a question on how far do the users satisfied in constructing models by which some parts of the LEGO bricks are missing? Certain LEGO models need to be developed by using special pieces of LEGO bricks that are distinguishable and not interchangeable to the bricks of other model. The lost of a piece means the user will not be able to develop completely a LEGO model as it used to be.

Thus, the traditional colourful plastic bricks are fixed in colours limiting users' expressionisms towards the models. The users need to have greater numbers of LEGO bricks of different colours, so that they can have better controls on the LEGO models. Unfortunately, the users need to spend higher amount of money to purchase more LEGO bricks.

The current LEGO construction only allows the user to play and express skills between users of a similar place and situation. Different users from disperse locations are restricted to join, share the creative imaginations, and construct the LEGO models together.

As far as these matters is concern, an exploration of a new approach, Virtual Lego, is believed would give an exposure and totally new experience to the audience or users in participating freely within a three-dimensional environment. Before going deeply into Virtual Lego, the concept of Virtual Reality needs to be clearly understood.

1.2 What is Virtual Reality?

Virtual reality (abbreviated VR) is a computer simulation of a real 3-dimensional world that gives an illusion to one or more users as if they are in a place and time different from their actual real-world location. Differs from computer simulations, VR simulations require special devices for not only to transmit the sensations and sound of the virtual environment, these devices also send the speech and movements of the users to the simulation program.

Users observe objects in the virtual world using a head-mounted display (HMD) that can be a pair of goggles or a full helmet (*Figure 1.4*). The two monitors in-front of the eyes will let the objects appear as 3-dimensional. Most HMDs include a head tracker that responds to the head movements. Users will be looking at different images inside virtual environment as they turn their head left and right. Users are also able to hear sounds of the environment through earphones in the HMD.



Figure 1.4 : A head-mounted display

To overcome the often uncomfortable intrusiveness of a head-mounted display, alternative concepts such as BOOM and CAVE are used for immersive viewing of virtual environments. The Binocular Omni-Orientation Monitor (BOOM) is a head-coupled stereoscopic display device that is used to see the virtual world by looking through two holds of a special box (*Figure 1.5*). Users can guide the box to any position within the operational volume of the device.

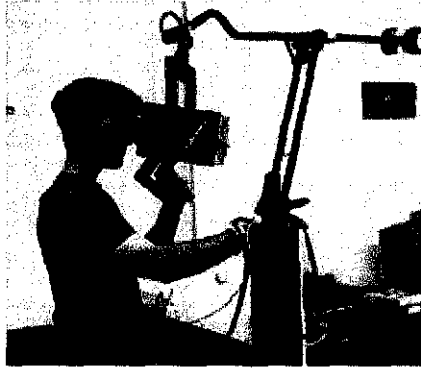


Figure 1.5 : A Binocular Omni-Orientation Monitor (BOOM)

On the other hand, the Cave Automatic Virtual Environment (CAVE) provides the illusion of immersion by projecting stereo images on the walls and floor of a room-sized cube (*Figure 1.6*). Several persons wearing lightweight stereo glasses can enter and walk freely inside the CAVE where the current position of the leading viewer is determined by a head tracking system.

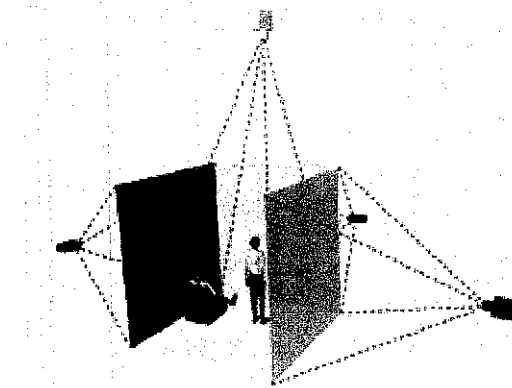
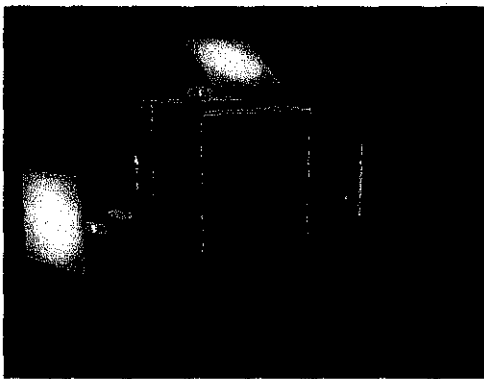


Figure 1.6 : A Cave Automatic Virtual Environment (CAVE)

To replicate the way of moving objects with hands, data gloves, joysticks, and hand-held wands allow the users reach into the virtual world and handle objects (*Figure 1.7*). Directional sound, tactile and force feedback devices and voice recognition are being employed to enrich the immersive experience and to create more sensualized interfaces.



Figure 1.7 : A Data Glove and a Joystick

Virtual reality is currently used to explore and manipulate experimental data in ways that were not possible before. A virtual environment can represent any three-dimensional world such as buildings, landscapes, human anatomy, crime scene reconstructions and even molecular models (*Figure 1.8*). Thus, VR applications can be used for teaching and training in a variety of areas such as medical, equipment operation and military, education, architectural walk-through and entertainment.

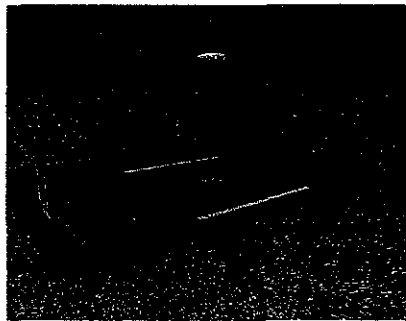


Figure 1.8 : A stadium created by VR

Hence, VR is also possible to be used in studies and treatment of phobias, for example fear of height and much more (*Figure 1.9*).

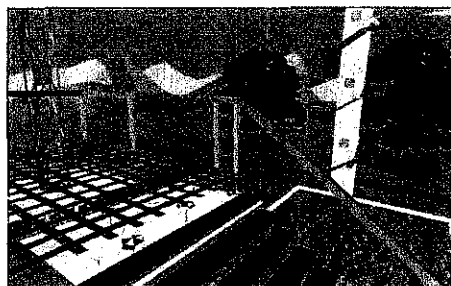


Figure 1.9 : An application used by doctors to treat patients who are phobia to height

1.3 Introduction to Virtual Lego

Virtual Lego, as its name implies, is a 3D virtual reality toy for building 3D shapes. Basically, the system will be based on the idea of LEGO, where it uses blocks as primitives that behave similar to real LEGO. Lego model construction is essentially an iterative method of connecting object pairs, where blocks can be assembled, disassembled and reassembled in enough new shapes and forms to tickle the imagination and stretch the user's creativity. In other words, it allows users to build models by securely attaching together a great variety of parts that interlock with each other in several different ways.

In general, the objectives that lie behind the Virtual Lego development are:

- To provide a scene construction system that supports an iterative and progressive modeling process.
- To let users perform meaningful rearrangement task by manipulating 3D object in a desktop VR system with input devices.
- To let users gain experience on virtual reality application that is fun to use and support their creativity and imagination in generating and applying the ideas. It is truly amazing what few bricks and a full measure of imagination will produce.
- To let users understand and appreciate the multi-step process involved when documenting creations on the computer.
- To allow collaboration between users inside a single application in completing a task. Thus, the teamwork activities will give experiences to users in working and combining ideas with others as if they are collaborate in a real world.

- To achieve realism, what is built in Virtual Lego can be satisfactorily replicated from the real world.

The system will support two modes of operation, view and model. In modeling mode, meaningful objects can be created by stacking the various Lego blocks on top of each other constituting movements in a fixed grid. In viewing mode, the models are presented in a more visually appealing mode, allowing the user to zoom and move around the scene.

In many virtual reality systems, objects can interpenetrate. This is something that almost all naive users find hard to understand. Consequently, this system will apply collision detection to prevent objects from interpenetrating each other. Besides, blocks that are moved over the background will remain floating in the air to increase the realism of the whole system.

The navigation processes of Lego blocks is made possible by the use of a traditional input device, a keyboard, and special VR input device, pinch gloves and tracker. The intended users of Virtual Lego are basically children, who are able to work together in assembling the bricks, share creative ideas and come up with great solutions. However, as the system is uniquely developed using VR technology, it is a great opportunity for peoples of different ages to try this new approach of VR application and experience the beauty that lies inside virtual reality.

1.4 Report Overview

This report is consistently documented into five main chapters that cover the overall issues of Virtual Lego and its development. Ideally, this chapter is consist of a brief overview on the LEGO's history and its products, including a general view on the new evolving technology, Virtual Reality (VR) of which it is the main platform to the development of Virtual Lego. An introduction of what is Virtual Lego, the system's uniqueness and timeline also covered inside this chapter.

The information obtained from the conducted research and analysis on the related fields prior to LEGO and VR is gathered and summarized into the Literature Review, covered under the Chapter 2.

Next, the physical and logical structures of the system architectures, which are specifically designed to assist the development of Virtual Lego is presented inside the Chapter 3. Brief explanations of the system architecture, the system flow, the UML diagram and the user interface design are provided for a better understanding of the visual aids. In addition, the hardware and software tools that are used to successfully develop the system are also covered at the end of the chapter.

The captured print screens of the working system application are properly presented inside the Result and Discussion's chapter. It reveals the success of the system implementations and discussions pertaining to the system development. The results obtained from the conducted user study are also available inside this chapter.

Last but not least, a conclusion of the report and the system is prepared under the Chapter 5. Strategic recommendations for Virtual Lego's future enhancements are listed at this very last chapter.

1.5 Project Timeline

This project has successfully undergone the planning, research and analysis, and design phases within the time frame of 14 weeks, and consequently taken another few months, roughly four, to implement and test the solutions obtained from the analysis done at the earlier stage, and come up with the working system application.

The system timeline is roughly planned and sketched before the Gantt chart is developed to become a tool in assisting the progress of Virtual Lego development, so that the project is delivered on time (*Figure 1.9*). The Gantt chart below only represents a portion of the complete chart, by which is supplemented inside the Appendixes section at the end of the report.

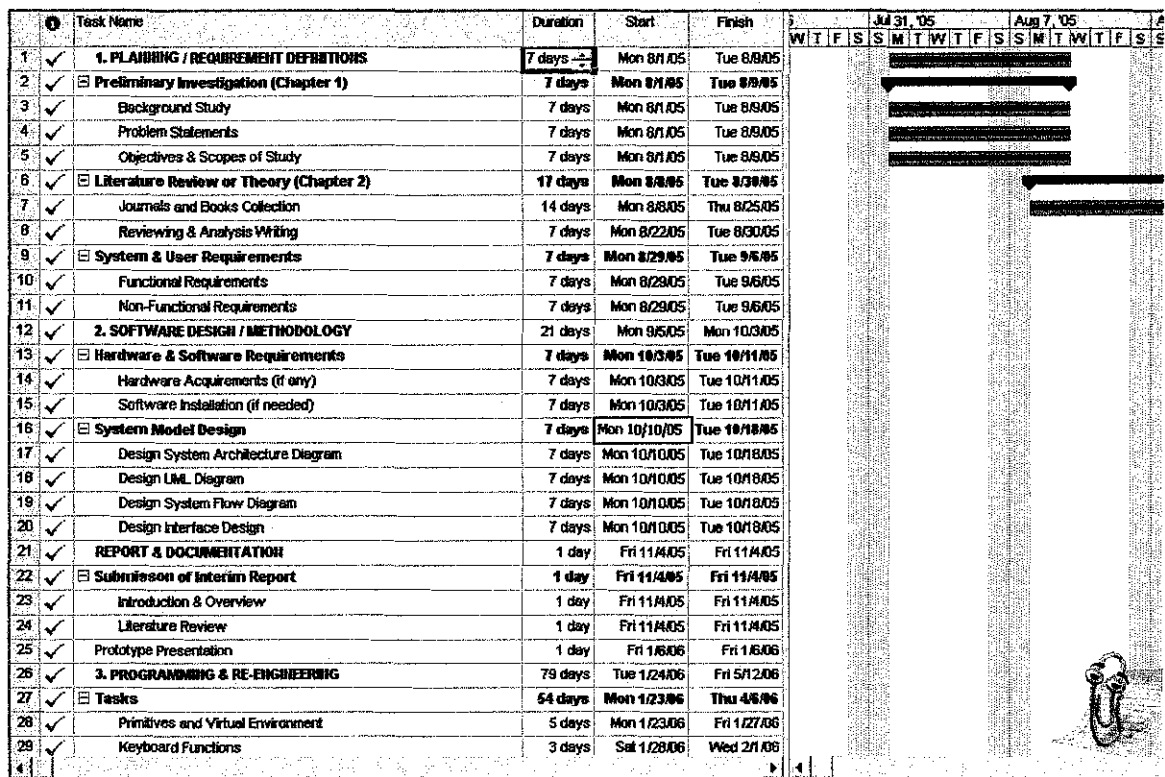


Figure 1.10 : The portion of the Gantt Chart for Virtual Lego

CHAPTER 2

LITERATURE REVIEW / THEORY

2.1 Introduction to Virtual Reality

Frederick had paraphrased Ivan Sutherland's vision on virtual reality evolution that was laid out in a lecture in 1965, "The Ultimate Display" as *'Don't think of that thing as a screen, think of it as a window, a window through which one looks into a virtual world. The challenge to computer graphics is to make that virtual world look real, sound real, move and respond to interaction in real time, and even feels real.'* [Frederick 1999]. Generally, virtual reality (VR) is defined as *'a medium composed of interactive computer simulations that sense the participant's position and actions and replace or augment the feedback to one or more senses, giving the feeling of being mentally immersed or present in the simulation.'* [William 2003].

2.2 Immersion and Presence

In the broader context, a VR system allows a participant to become immersed in a completely synthetic computer generated environment. *'The greatest impact virtual reality will have on science is how it will change our thinking. Virtual reality encourages viewers to be participants immersed in the data rather than passive observers watching from a distance.'* [Nigel 2001]. Immersive is *'the extent of the peripheral display imagery'* [Kalawsky 2000] and *'the extent in which the computer displays are extensive, surrounding, inclusive, vivid and matching.'* [Slater 1996]. Hence, achieving a sense of presence in the environment can be made possible with VR system. According to a journal written by Sheridan, presence is defined as *'experienced by a person when the*

sensory information generated only by and within a computer compels a feeling present in an environment other than the one person is actually in. [Sheridan 1992].

2.3 Interactive Designs of Virtual Reality

A VR system has *'becomes un-inhibited where conventional laws of physics can be controlled in a way that assists greater understanding.'* [Exploiting Virtual Reality Techniques in Education and Training: Technological Issues 2005]. With the system, a participant can achieve a sense of non-real time, where situations can be presented in slow or fast time. Furthermore, a participant also can enjoy a high degree of interaction that can equal or exceed that achievable in the real world.

Four technologies are crucial for VR [Frederick 1999], which are:

- *the visual (and aural and haptic) displays that immerse the user in the virtual world and that block out contradictory sensory impressions from the real world;*
- *the graphics rendering system that generates, at 20 to 30 frames per second, the ever-changing images;*
- *the tracking system that continually reports the position and orientation of the user's head and limbs; and*
- *the database construction and maintenance system for building and maintaining detailed and realistic models of the virtual world.*

In the introduction to his paper reporting on the Workshop on the Challenges of 3D Interaction, Herndon stated that *'The most effective ways for humans to interact with synthetic 3D environments are still not clear'* [Herndon 1994]. On the other hand, Isaacs et al in a section on user interface design in the same paper, contend that user interfaces to 3D graphics applications must be tailored to suit their particular user communities (architects, surgeons, designers etc, also novice or expert user of 3D graphics interfaces) [Isaacs et al 1994]. Consequently, there are some issues that need to be resolved with respect to virtual interfaces and VEs [Barfield 1995), include:

- the ‘*need to develop a theoretical basis for the work done in VEs, and a need to develop conceptual models to assist the designers of virtual worlds*’.
- the ‘*need to develop a solid understanding of the human factors design implications of virtual interfaces*’.
- the ‘*need to develop languages, spatial and state representations, and interactive heuristics for constructing virtual worlds*’.

They conclude by stating that there is a ‘*need to design more natural and intuitive interfaces to virtual environments*’.

2.4 Virtual Displays and Interfaces

To achieve interactive VR applications, ‘*it had to be ensured that whatever was built in the VE could be replicated in the real world, and conversely, that everything that was put into the real world scenario could be built satisfactorily in the VE.*’ [Eastgate 2001]. In setting up the real world scenario, menu selections can be integrated inside the user interface, so that the participant can use the facility to navigate as in a real world inside the VE (*Figure 2.1*).

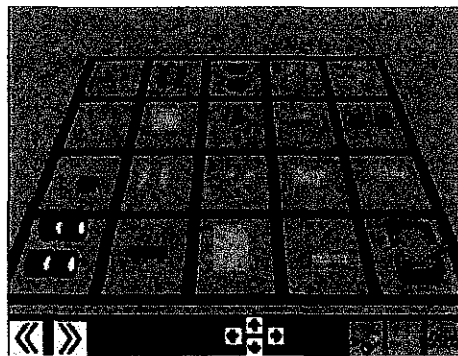


Figure 2.1: The Lego components lay out on the parts grid

In the real world the partially assembled model can be rotated and viewed from any angle. To facilitate this in the VE, clicking on any four buttons with arrows that be arranged at the bottom of the display would rotate the direction indicated. Hence, in the

real world the user can, at any time, decide to remove components from the partially completed model, and then rebuild it. This would be made possible in the VE by providing a 'go-back' button, each click on which would remove one component (most recently added, first removed).

In the other hand, some of VR applications do not need any menus or buttons inside the user interface. The modeling tool "Sculptor" for architectures has '*enabled integrations of direct specifications and manipulation of intelligent objects and scenes in 3D.*' [Engeli 1996]. As Sculptor's interface is almost widget-less, it is very important that the view is undisturbed, while participant looks at a virtual building. Therefore menus and buttons have to be avoided and been replaced by a pop-up menu during the course of designing.

2.5 Structure Elements in Virtual Environment

In the year of 2001, Eastgate had written a thesis that covered the essential elements in VR that give structure to the development of VEs [Eastgate 2001]. He had stated that it is necessary to understand what the components of a VE are and how they relate to each other in order to understand how a VE is built. Eastgate, in his paper also had concluded that VE consists of four building blocks namely topography, objects, behaviors and viewpoints. These four building blocks are not processed in any fixed sequence.

The term topography is being used to refer to all aspects of the layout of the VE. '*VEs have naturalistic topographies in which the real world characteristics of 3D space are reproduced.*' [Eastgate 2001]. However according to Greenhalgh, '*It is possible in a VE however, to create unnatural topographies such as 'portals' or 'teleports' to distant parts of the environment, or 'tardis' type structures, all of which are not currently possible in the real world.*' [Greenhalgh 2000].

The objects within the layout are made up of combinations of shapes, colors and textures. *'More complex visual effects can be achieved by applying textures to the facets to create visually realistic objects.'* [Eastgate 2001]. He goes on stated that by applying gradual, smooth shading across several faces can simulate curved surfaces realistically, although the geometry is still made up of flat facets.

Some of these objects may have dynamic or transient characteristics, which can be described as behaviors that be categorized as interactive, autonomous, linked, or 'no behaviors'. *'As well as inherent properties, objects can also be given function, such as a door that opens when the handle is turned.'* [Eastgate 2001]. According to the thesis, these functions will often be in response to user interactions, but they could also be in response to the behaviors of other objects in the VE, or entirely autonomous. *'The designer observes this autonomous movement and can stop it any point she likes it most. Therefore we denoted this expansion of the architectural design process through autonomous behavior the 'I like it' principle.'* [Engeli 1996].

2.6 Stereoscopic View and HMD

One of the most important features of any display of environments that are three-dimensional is the perception of the third dimension, depth. Consequently, central to many of the depth cues and three dimensional vision in general is the phenomenon of stereoscopic vision. *'Stereoscopic imagery uses two images of the same scene taken from slightly different viewpoints. Using one of many viewing technologies, the images are combined in your mind producing a third dimension—depth.'* [All About Stereo Photography 2005]. According to the article, 3D effects arise from the fact that each of eyes sees from a slightly different perspective.

'In both real-world and computer-generated imaging applications, stereoscopic 3D display technologies can enhance the user's ability to perceive objects in their correct spatial locations, to move through display space easily, and to manipulate objects efficiently and accurately.' [Stereoscopic Displays and Applications XVII 2005]. Head-

Mounted Displays (HMDs) are, probably, the most broadly used visual displays in VR systems. *'Unlike a movie, where the director controls what the viewer sees, including the angle and field of view, the HMD allows viewers to look at an image from various angles or change their field of view by simply moving their heads.'* [Virtual Reality 2005].

2.7 Force and Haptic Feedback

Virtual objects are manipulated and behave quite differently from real objects. *'In order to provide a user with a natural manipulation environment composed of virtual and real objects, it is necessary to bridge the gap between the differences in the physical laws that operate on both virtual and real object.'* [Kitamura 2002]. In the paper, these authors have introduced “magnetic metaphor”, a method to enable a participant to manipulate virtual and real objects in the same manner by expecting the same behaviors from objects according to the same physical laws. Their approach is *'to give the user a feeling as if the corresponding virtual object has an attracting power, and also to give the user a feeling as if the pseudo magnet has been detached when separated from other objects.'* For this purpose, it is possible to use a force feedback display to represent a magnetic power in the user's hand. *'Force feedback reproduces directional forces that can result from solid boundaries, the weight of grasped virtual objects, mechanical compliance of an object and inertia.'* [Jeffrey 2003].

The precise manipulation of virtual objects is complicated by lack of haptic feedback [Mark]. *'Haptics is a recent enhancement to virtual environments allowing users to “touch” and feel the simulated objects with which they interact. Haptics is the science of touch.'* [Jeffrey J, 2003]. Most common computer interface devices, such as mouse and joysticks do not provide realistic touch sensations coordinated with on-screen events, thus, allow only one direction of information flows, from the peripheral to computer. Examples of haptic devices include consumer peripheral devices equipped with special motors and sensors, such as force feedback joysticks and steering wheels. PHANTOM device is an example of more sophisticated devices designed for industrial, medical or scientific applications.

2.8 Navigational Issues

User problems when navigating in VEs were summed up by Ruddle as being users' lack of knowledge of their position, their orientation and a VE's structure, and a general lack of familiarity with using VEs [Ruddle 1998]. As a result of an experimental study looking into the use of a virtual compass as a navigational aid, he concluded that desktop VR may cause specific navigation problems due to the fact that the user does not physically turn their body when they change direction in a desktop VE as they would in the real world, in an HMD or in a cave. However, *'one of the advantages of desktop VR is that the hardware used can be a standard desktop PC system with a few optional extras. This results in a low cost way of implementing VR technology within an organization.'* [Eastgate 2001].

In their journal paper, Mark, Frederick and Carlo propose that proprioception can be used to develop a unified set of interaction techniques that allow user to interact with a virtual world intuitively, efficiently, precisely and lazily [Mark]. Here, proprioception is defined as a person's sense of the position and orientation of his body and its several parts. In the user studies, these authors compared the efficiencies between manipulating virtual objects that are co-located with one's hand and manipulating objects at a distance. Analysis in each experiment revealed a preference for the manipulation of an object using hand rather than the manipulation of an object with an offset.

Mapping from a 2D mouse input to a 3D position is difficult problem in desktop VR [Issacs 2002] as the system needs to provide for 3D motion of the objects, as well as afford object group selection and manipulation. In addressing this issue, Oh and Stuerzlinger have presented two new techniques for grouping of objects, which are intelligent separation and separation with anchoring [Oh 2004]. To verify the effectiveness of the proposed techniques, a series of user test were performed where the participants were asked to rearrange a 3D floor plan under three conditions; intelligent separation, separation with anchoring with one mouse, and with two mice. Overall, the results obtained suggest that intelligent separation requires a longer learning period than

separation with anchoring. However, this technique does not require the extra anchoring or step. Both authors had indicated that the performance will improve once participants master this technique. The authors also concluded that the 2 mice condition did not provide enough cues for the left hand's physical position in relation to the right hand that made participants feel awkward in moving the cursor using that hand.

Besides, Neale suggest a number of ways in which the design of the VE can be adapted to aid navigation especially for users with real world disabilities [Neale 2000]. These methods include adding more cues such as arrows and signs to indicate the direction to go to reach a specific target, and changing the layout of a VE to make reaching the target easier.

2.9 Communication Support in Virtual Reality

Last but not least, virtual reality is not only standalone virtual worlds. Although a virtual world is a place without physical substance it is a place where real people meet and engage in activities that have real consequences [Jakobsson 1999]. The Thanmanns' advocate *'needs for shared environment to provide a comfortable interface for gestural communication, support awareness of other users in the environment, provide mechanisms for different modes of interaction (synchronous vs. asynchronous, allowing to work in different times in the same environment), supply mechanisms for customized tools for data visualization, protection and sharing.'* [Thalmann].

CHAPTER 3

SYSTEM ARCHITECTURE

3.1 System and Software Design

Proper system and software designs with regards to the system inputs, work processes, system outputs and interface design are essential to be prepared in order to understand the interaction between users and the system. Thus, the design documents that consist of system architecture, system flow diagram, UML use case diagram and user interface design are made to assist the entire development of the end-product.

3.1.1 System Architecture

The design of the system architecture can be used as a key to obtain a clearer understanding on the overall structures of Virtual Lego (*Figure 3.1*). Practically, the system architecture is divided into two parts, which are a server side and a client side, connected via the TCP/IP network. Both users from each side are allowed to share their valuable ideas and together assemble the Lego blocks within a single shared environment. To initiate these actions, the users are provided with two types of input devices that are a keyboard and pinch gloves.

A special virtual reality input device, pinch gloves are used together with the magnetic tracker to track the motion of the users' fingers without degradation in update rate or an increase in lag. A Flock mid-range transmitter is used to provide enough coverage for the users to be tracked throughout a full-sized room

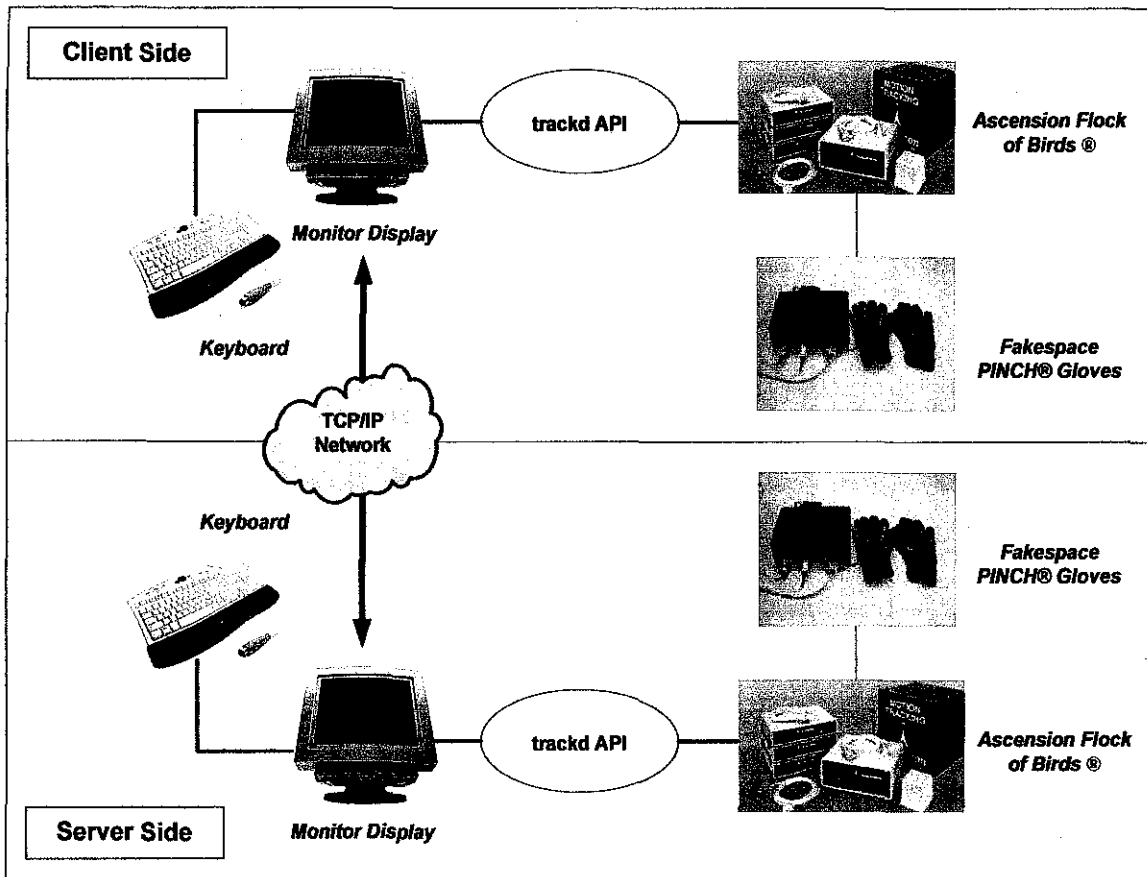


Figure 3.1 : System Architecture that illustrates the client side and server side for Virtual Lego

Whenever the user pinches the glove, a set of binary values that is generated corresponding to the pinch is send to the Flock receiver for further initializations Basically, these binary values will be processed and interpreted inside the Application Programmers Interface, called the trackd API, before the new computed value that is understandable to the application is send to the user's computer. This generic interface allows applications to support a range of hardware without having to modify source code, or write specific hardware drivers. The application program will then process the information and concurrently displays the results of the user's action on the monitor display. Thus, the new actions will be concurrently updated to the other computer display of the other user.

3.1.2 System Flow Diagram

Networked Virtual Reality

To incorporate the total experience, this system supports communications between two different users by allowing them to share the same virtual world. This ‘networked virtual reality’ permits these group of users to virtually interact by generating and combining ideas, and working together in assembling and manipulating the Lego bricks even though there are disperse in locations.

The flow of the system begins by determining which entity are the server and client to permit the concurrent interactions between different computers (*Figure 3.2*). After compilation program is made by a user, the system will check the availability of a server through out the TCP/IP network, and if a server is not up yet that user will automatically becomes one. The new server will wait for another user, who will automatically become a client, to start the application.

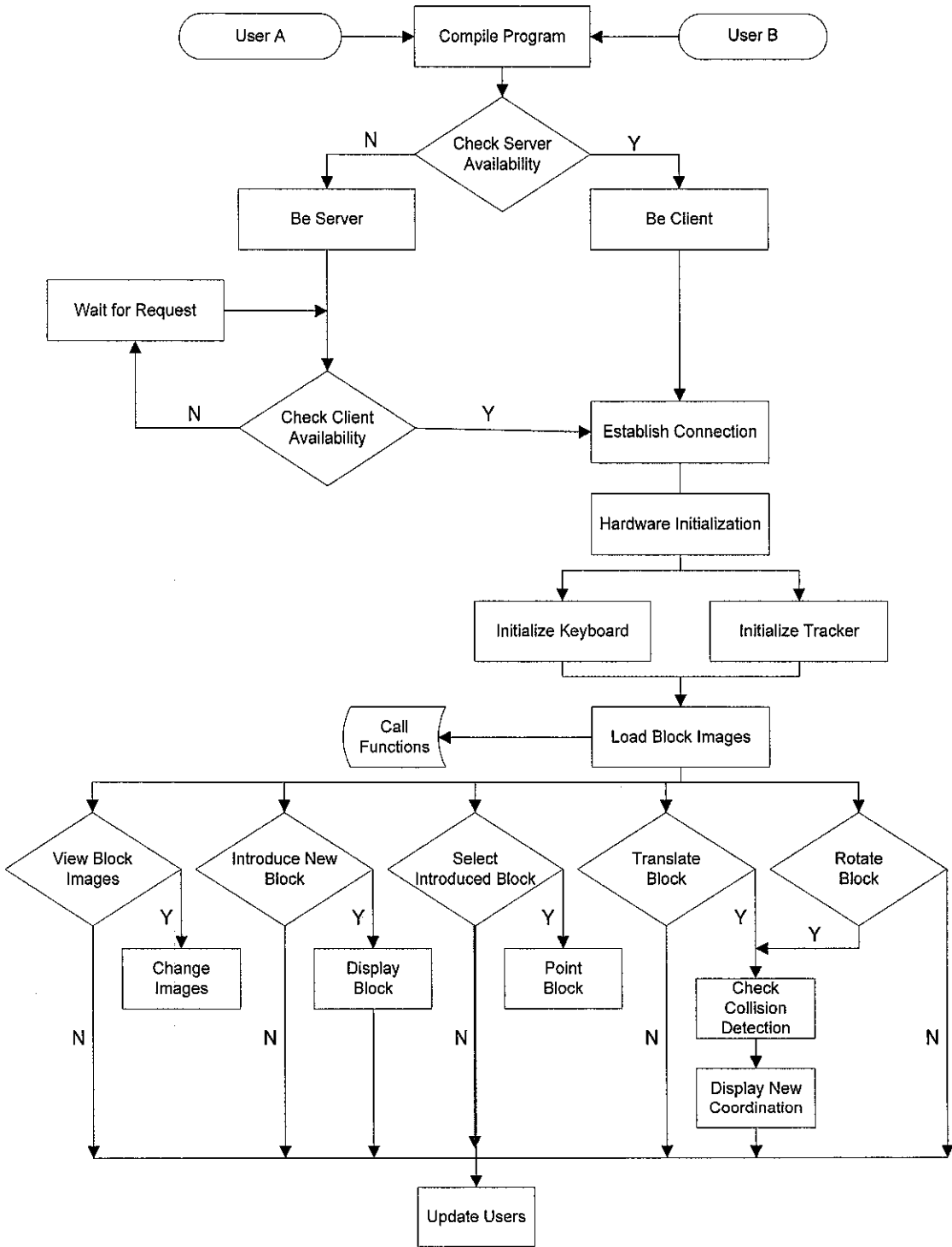


Figure 3.2 : System Flow Diagram for Virtual Lego

User Interaction and System Navigation

After the Client-Server connection is established, the initialization of input hardware is taken place. The Lego blocks are laid out on a Desktop VR display and navigated using the traditional input device, a keyboard or pinch gloves that measures the flexure or touch of the users' fingers. Discrete actions, such as viewing and adding new blocks, and also continuous actions, such as translations and rotations, are assigned to keyboard's button clicks and pinches of the glove.

Success in loading the images of the Lego blocks onto the system interface will allow the users to view each of the block that are available inside the system. The users will need to introduce new Lego blocks inside the 3D scene before using the blocks to creatively construct a meaningful object that is reflected by their imaginations.

Thus, each user is allowed to manipulate the Lego blocks by the means of translations and rotations according to their preferences. The manipulations do not only applicable to be performed to the new introduced block of the scene, but can also be carried out to the entire previous selected blocks. A further explanation on this is available under UML Use Case Diagram section.

Collision Detection

To provide the users with a sense of realism during the development of 3D objects, the system detects whether or not collision between Lego blocks has happened, and if it is so, the system responds to the collision by stopping on its current positions. It means a Lego block that collides to one another will not be interpenetrated.

All the actions made by the users will be concurrently updated to each other that make the 3D scene view of each computer display is currently the same.

3.1.3 Unified Modeling Language (UML) Use Case Diagram

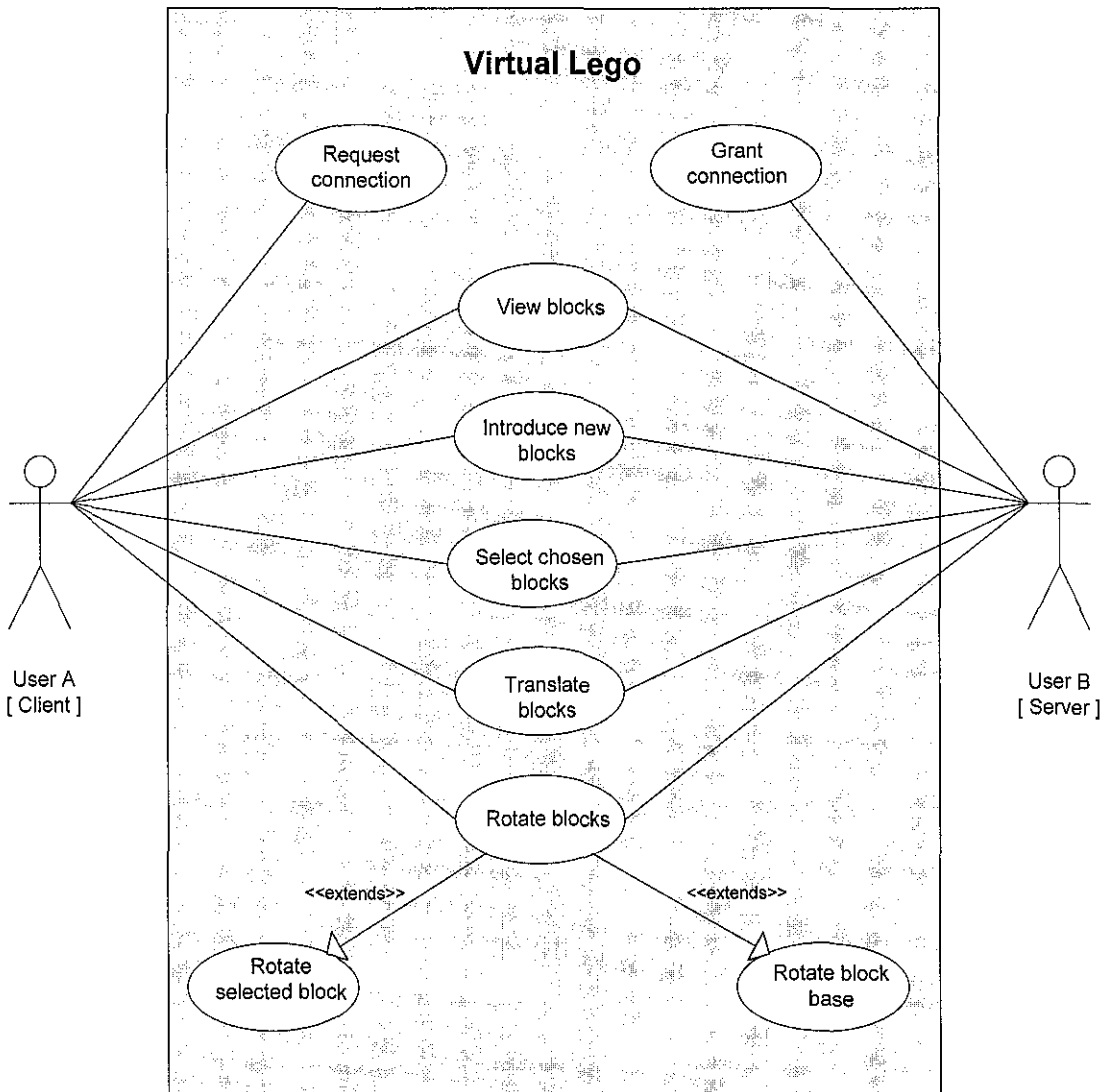


Figure 3.3 : UML Use Case Diagram for Virtual Lego

As shown above is the UML Use Case Diagram that includes all system's functions that are involved inside Virtual Lego (Figure 3.3). Consequently, it gives an overall view of what the system does and who are the intended users that will be using the system. The application system will start immediately as when User B who represents the Server grant the connections requested by the Client who is User A.

Both of the users are able to view all the pre-defined Lego blocks of the system and introduce new Lego blocks into the 3D scene, by which the actions can be concurrently observed by the two of them. The users are also able to select any Lego blocks that are already allocated inside the 3D scene for further manipulations.

In a way of constructing meaningful objects, the users are able to move the selected Lego block freely inside the scene by translating the coordination of the x-axis, y-axis and z-axis of the blocks. Thus, the block can also be rotated up to 360° degrees to provide the users with better views of the blocks' structures. The block base, which is used as the ground for the Lego blocks to be stacked onto, is also able to be rotated so that the users can view around the object that is together constructed by them.

3.1.4 User Interface Design

The system interface is a medium to be used by the users to communicate with the system. The main and the only user interface of Virtual Lego is consists of a user guideline that displays the keyboard functions on the upper-left-hand-side and Lego block-selection palette on the right (*Figure 3.4*). Each possible actions of the user are determined by the pressed of the special keys listed on the user guideline, while the palette is designed to be used for displaying the entire Lego block images for the views of the users. A 3D scene view that is on the center of the system interface displays the results of the key pressings performed by the users. In another words, the 3D scene view is where each user can observed how well they manipulate and stack the Lego blocks, as well of being informed and updated of another user actions within the same application.

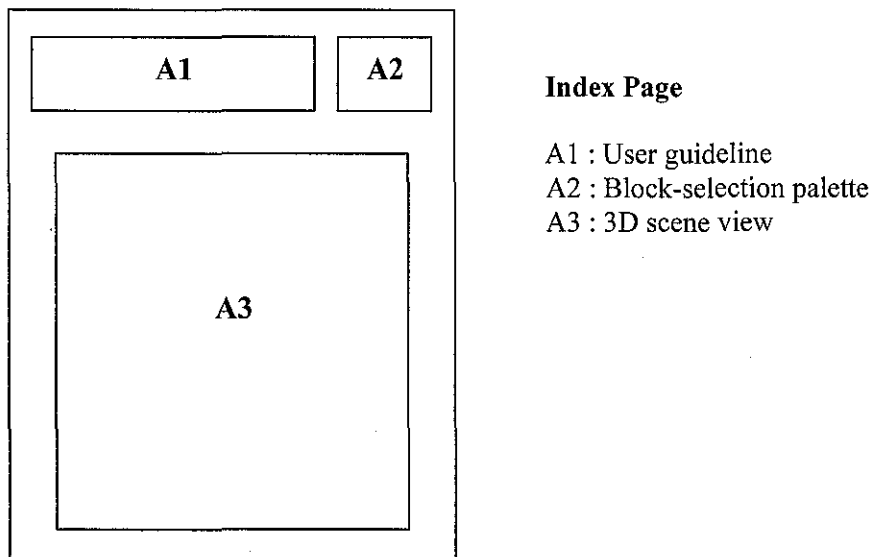


Figure 3.4 : User Interface Design for Virtual Lego

CHAPTER 4

RESULT AND DISCUSSION

4.1 Primitive and Virtual Environment

The system presented for this project is fully developed using C++ and OpenGL, the premier environment for developing interactive 2D and 3D graphics applications. The whole fundamental primitives in the system, which are the basic Lego blocks, are also coded using the program and graphic engine.

OpenGL that stands for Open Graphics Library is a cross platform graphics programming API (application programming interface). OpenGL is used because its implementations are available on a wide variety of platforms. Thus, OpenGL is truly open and stable graphics standard where most video cards have native OpenGL support. As an industry standard, additions to the specification are well-controlled and backward-compatibility requirements ensure that existing applications do not become obsolete.

By the use of OpenGL, Virtual Lego is reliable in producing consistent visual display result, regardless of operating system or windowing system. In addition, Virtual Lego based on OpenGL API can run on systems ranging from PCs to workstations, and supercomputers. As a result, the application can scale to any class of machine that the developer chooses to target.

The other tools required to facilitate the system development and stimulate the environment for the end-product are listed below.

Software

- a. Microsoft Windows XP Professional Edition
- b. Microsoft Visual Studio .Net 2003
- c. Adobe Photoshop
- d. trackd by VRCO

Hardware

- a. Desktop VR
- b. Keyboard
- c. Fakespace PINCH® Gloves
- d. Ascension Flock of Birds®

4.1.1 Development of Primitive

The fundamental primitives of the system are rectangular block shapes with different sizes and colors of surfaces (*Figure 4.1*). The connectivity layouts or joins on top of the Lego blocks are created with the same predefined diameter and height to ensure that the Lego blocks will be stacked properly on top of each other. Each of the Lego blocks is able to be rotated on x-axis, y-axis, and z-axis, which will allow it to be viewed in any angles by the users.

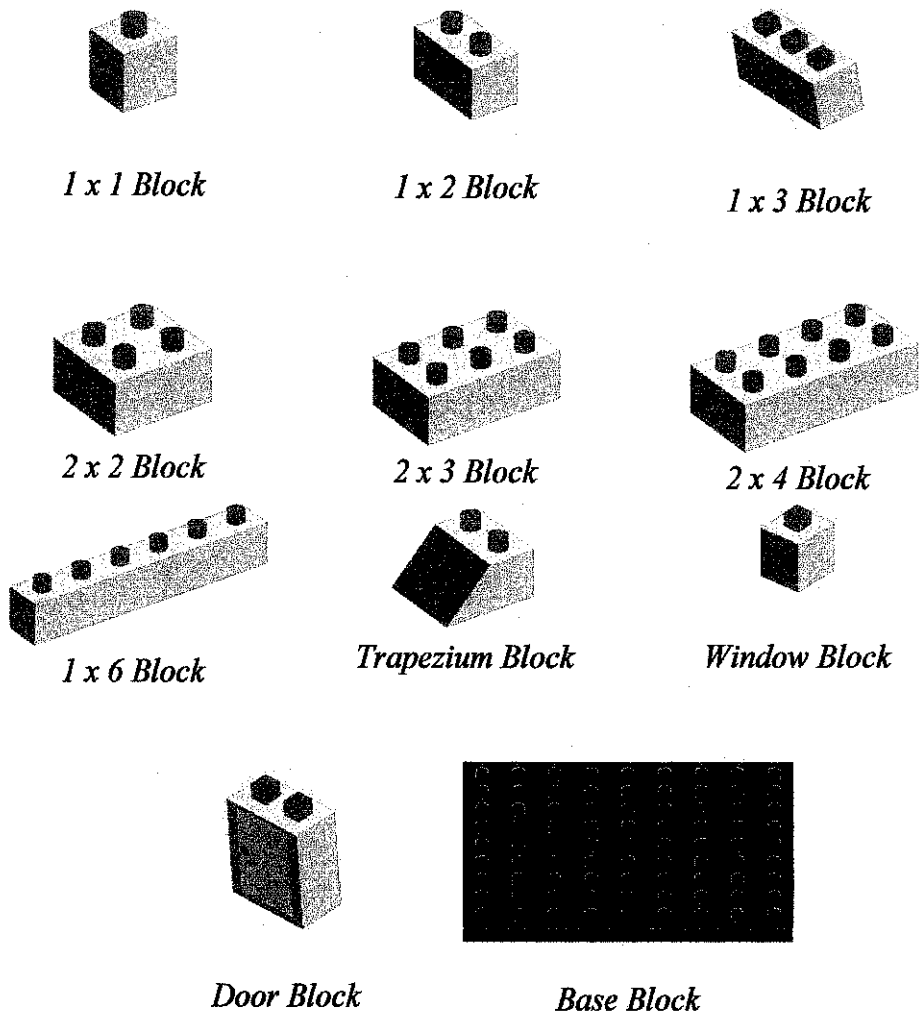


Figure 4.1 : The Lego blocks

4.1.2 Development of Virtual Environment

The user interface of the Virtual Lego system is designed in Adobe Photoshop and textured into the environment using Bitmap functions. The images of the Lego blocks are captured and saved as bitmap files with both of its heights and widths are equal in sizes. The *bitmap.cpp* and *bitmap.h* files that are allocated inside the system program are called to map the user interface with the Lego blocks' images on top of the block-selection palette. All the bitmap images are displayed on the interface based on its pre-defined coordination (*Figure 4.2*).

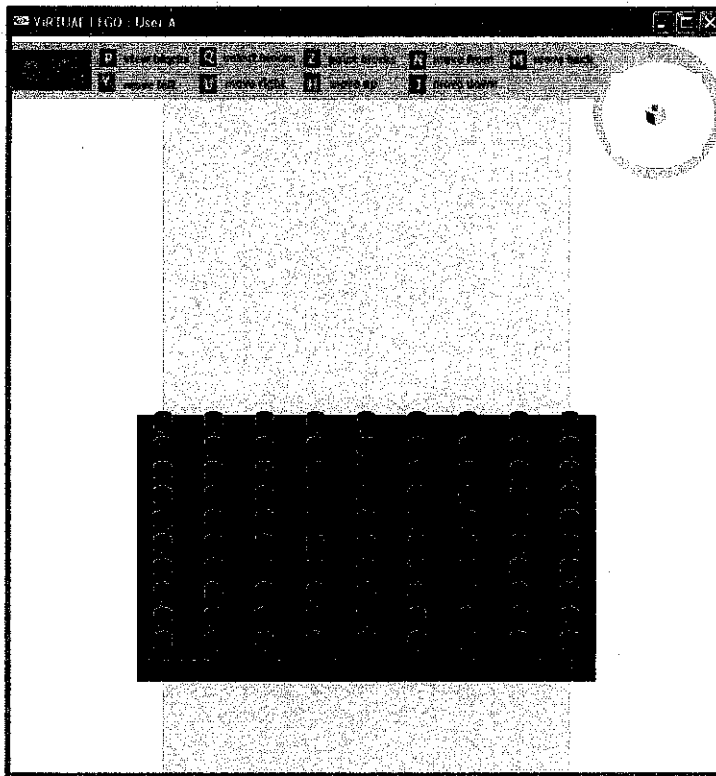


Figure 4.2 : The Landing Page of Virtual Lego

4.2 Networked Virtual Reality

Virtual Lego has successfully integrates the networked virtual reality into the system that allow two users from different locations, to combine ideas and cooperate between each other in constructing 3D shapes using Lego blocks within the same virtual world. This is achieved by connecting the host computers to a computer network using the computer's IP address that is defined inside the main program.

As soon as the Client-Server connection is established, the application program will starts and the landing page of Virtual Lego will be concurrently displayed for both users. The application is now ready to be used, where the users can view all the Lego blocks' images before introduce the actual blocks into the virtual environment. Virtual Lego has made it possible for its users to obtain similar views whenever each of the user making some manipulations, translation and rotations on the selected Lego blocks (*Figure 4.3*). By this,

the users will have the feelings of constructing together the Lego blocks as if as they are in a similar locations and time.

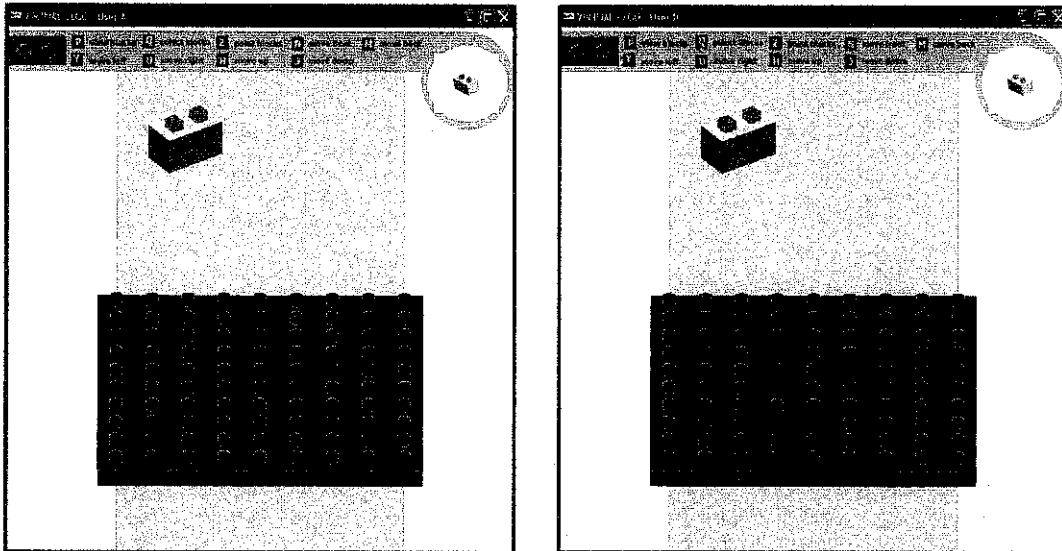


Figure 4.3 : Concurrent view observed by each user who is dispersed in locations

The update processes commence through the process of sending messages between the users' computers via the network. Basically, the message contains of three different parts of information. The first portion of the message is the information of user's actions, to inform whether the user is introduces a new Lego block into the scene, or translates and rotates the selected Lego blocks, or the user is performs nothing (*Figure 4.4*). The second and the third portions of the message consist of the new coordination for the Lego block's translation and rotation.

```
void TellFriend
{
    if (introduce_new_block)
    {
        sprintf(buffer, "s-%d -18 25 -50 0 0 0x", (lego+1));
        introduce_new_block = false;
    }
    else if (block == NULL) // nothing is done
        sprintf(buffer, "s0x");
    :
    .
}
```

Figure 4.4 : The portion of code that contain the structure of message to be send by each user to differentiate the actions made inside the application

To let the users view the Lego blocks and the whole virtual environment properly, the Base block on the center of 3D scene view can be rotated 360° degrees on its y-axis. However, as different people have different way in perceive things and view of preference, the rotation made by a single user will not affect the coordination of the Base block on the other user's application display (*Figure 4.5*).

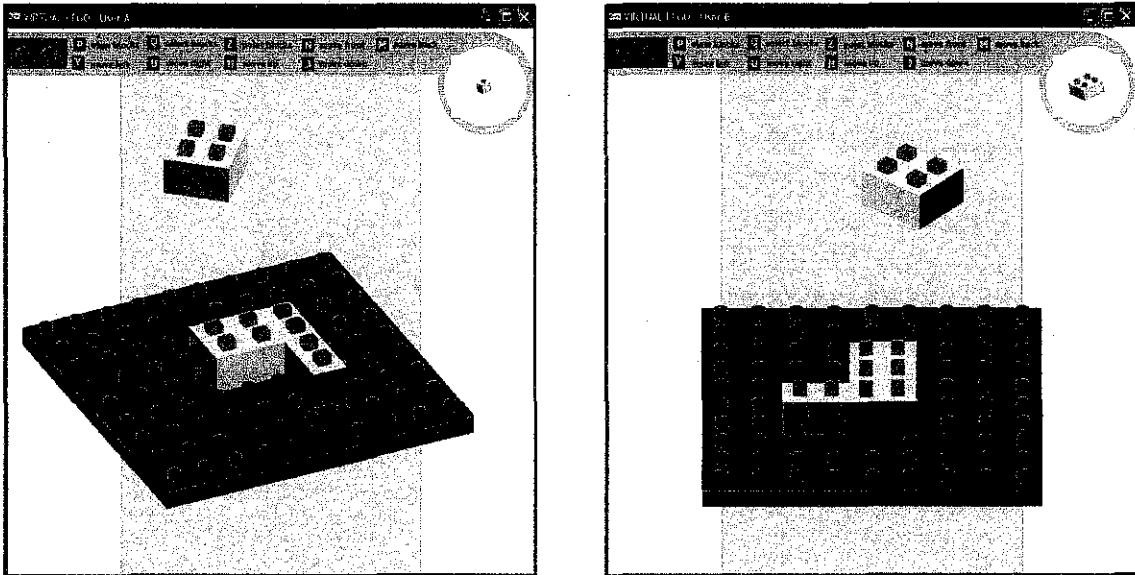


Figure 4.5 : Rotation made on the Base block only can be viewed by the user who performs the action

4.3 Collision Detection

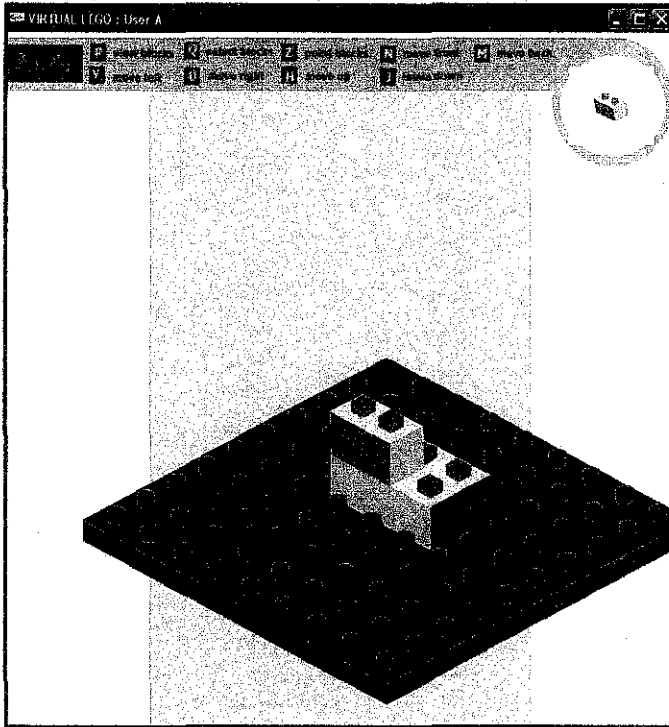


Figure 4.6 : Collision detections successfully avoiding the Lego blocks from interpenetrating with each other.

Precise collision detection can give a higher experience of realism as it limit the Lego block from intercept to one another if the occurrence of collision is detected. Virtual Lego is managed to incorporate collision detection into the program so the Lego blocks can be stacked properly on top of the Base and other Lego blocks (*Figure 4.6*). Lego blocks that are intercepting with each other will produce an unreal perception for the users (*Figure 4.7*).

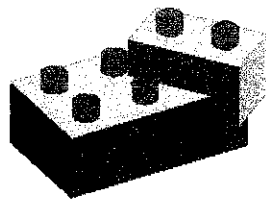


Figure 4.7 : 1 x 2 Block collides with 2 x 3 Block

By far, attaching the collision detection onto the Lego blocks and Base block are the most difficult and complex tasks throughout the whole system development. Each single point constitutes the external structure of the blocks need to be shielded with collision detection to avoid the blocks from collide between each other (*Figure 4.7*).

```
int curr_index = blockList.size();

    colModel[curr_index] = newCollisionModel3D();

colModel[curr_index]->
    addTriangle(4*0,2*4.5,3*0, 4*2.4,2*0,3*0, 4*0,2*0,3*0);
colModel[curr_index]->
    addTriangle(4*0,2*4.5,3*0, 4*2.4,2*0,3*0, 4*2.4,2*4.5,3*0);
    .
    .
```

Figure 4.8 : The portion of code for collision detection

4.4 User Interaction and System Navigation

Options are given to the user to choose their most preferable input devices to be used in manipulating the Lego blocks inside Virtual Lego. The old-time-favorite, a keyboard, is a common and an easy-to-learn input device with a low price range in the market (*Figure 4.8*). However, the hands of the users will be constrained to a specific place through out the whole navigation processes.

```
void Keyboard (unsigned char k, int x, int y)
{
    .
    .

    case 'u':
        blk->x += 1.0;
        printf("move left\n");
        break;

    case 'y':
        blk->x -= 1.0;
        printf("move right\n");
        break;
```

Figure 4.9 : The portion of code that generate the keyboard functions

On the other hand, the pinch gloves are more convenient to the users because it allow them to assemble and construct the Lego models even when they are far from the monitor display. Thus, the users' hands can be moved and flexed around without limitations while they are pinching the gloves. Unfortunately the cost of pinch gloves is much greater to be compared to a cost of a keyboard.

4.4.1 Virtual Task Listing

The virtual task listing below is based on the 'real' task analysis, which is now being carried out via the keyboard and the pinch gloves.

Tasks	Keyboard	Pinch Glove
View Block	P-key	L1L2
Introduce Block	Q-key	L1L3
Choose Selected Block	Z-key	L1L4
Move Down	J-key	L1R1
Move Up	H-key	L2R2
Move Right	U-key	L3R3
Move Left	Y-key	L4R4
Move Front	N-Key	L5R5
Move Back	M-key	R1R2
X Rotation	Key-Up	R1R3
Y Rotation	Key-Left	R1R4
Z Rotation	Key-Right	R1R5
Move Base block to Right	3-key	L2R3
Move Base block to Left	4-key	L2R4
Move Base block to Default Position	5-key	L2R5

Table 1 : The virtual task listing of keyboard and pinch gloves for Virtual Lego

4.5 Development of Meaningful Objects

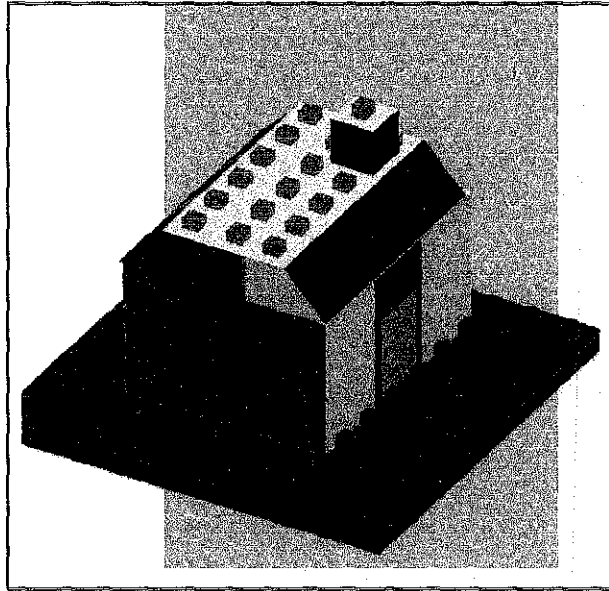


Figure 4.10 : The house that has been successfully developed inside Virtual Lego

It is amazing with a few basic Lego blocks of Virtual Lego, a lot of meaningful objects that reflect the real world objects can be constructed (*Figure 4.9*). The possibility is even higher as Virtual Lego allows different users from different location to generate and share creative ideas, and together they navigate the Lego blocks and manipulate the shared environment to come up with different types of Lego models.

CHAPTER 5

CONCLUSION AND RECOMMENDATION

5.1 Conclusion

As far as the system is concerned, the prototype of Virtual Lego is able to be developed where it works with several virtual reality hardware such as VR display, tracker and pinch gloves. The development of Virtual Lego is an achievement to the target of exploring the powerful of virtual reality and also the tools it possessed that contribute in enhancing the quality of life of the human beings.

The author would like to clarify that all the objectives of Virtual Lego, which are clearly defined in the earlier stage of the system development are successfully met. At this stage, Virtual Lego is able to be used to perform meaningful rearrangement task by manipulating the Lego blocks in a desktop VR system with input devices such as keyboard and pinch glove. Another objective of which Virtual Lego is able to achieve is the ability to become a medium for the user to gain experience on virtual reality that is fun to use. Thus, Virtual Lego has successfully added-in the element of networked virtual reality that permits collaboration between users inside a single application in completing a task. Last but not least, Virtual Lego is likely to achieve the realism, where it is satisfactorily manage to be replicated from the real world.

However, due to certain limitations such as resources and time constraints, Virtual Lego also has some limitations. The main issues such as collision detection, the number of system users and certain system functions are need to be addressed to improve the reliability and efficiency of the system as a whole.

Last but not least, Virtual Lego is a mean of a new exploration towards VR technology for entertainment and education purposes, which is not only intended for children but also to the whole standard of ages.

5.2 Recommendations

In this section, some recommendations that can be implemented to the system are discussed. All of the recommendations are gathered throughout the system development and collected from the comments and suggestions given by the lecturers and colleagues who have tried out the system. All these recommendations should be highly taken into consideration as they are important for future Virtual Lego's improvements.

a) Use a high technology of virtual reality device such as StereoGraphics Crystal Eyes

In conjunction with a pair of StereoGraphics Crystal Eyes, an active-stereo shutter glasses, Virtual Lego will be capable of presenting users with stereo images that allow them to view the Lego bricks as solid in three spatial dimensions. Furthermore, the stereoscopic vision will able the users to see the bricks moving toward or away from them in the depth dimension, which making the scene looks real.

b) Precisely improve collision detection and incorporate lighting effects into the 3D environment

It will be more efficient for the system to have precise collision detection for the whole of the Lego blocks to ensure the virtual world created inside Virtual Lego can successfully be replicated from the real world. For a better improvement of Virtual Lego, collision detection for the connection layouts needs to be developed other than enhancing the existing collision detections of the rectangular blocks.

c) Expand the number of users to join in the application

Stacking and manipulating the Lego blocks can be more fun if many users can collaborate together and pouring more ideas and creative imaginations in developing meaningful objects. Instead of having only two persons at a time, the network system architecture can be expanded to welcome more users with a better technology and quality of network services.

d) Add in more meaningful functions to navigate the Lego blocks

At this moment, Virtual Lego only permits the user to view the pre-defined Lego blocks, introduce the new block into the 3D scene, select the chosen Lego block, do translations and rotations. The system will be more worth if the user is not only able to select their preference Lego blocks to be inserted in the 3D scene, but also have the ability to delete any Lego blocks as they wish. Thus, the user should also be able to change colors of the Lego interfaces. These will permit users a better control of the system and the way of which they preferred in completing the tasks.

e) Create varieties of Lego blocks so that can develop more specific and meaningful objects

To add more sparks into the system, varieties of Lego blocks' shapes and sizes can be created to provide the users with better choices of blocks in a way for them to assemble and create more meaningful 3D objects. It will be more interesting if the Lego blocks are meant for certain entities such as tree or Lego man that reflects the elements of the real world.

f) Make further improvements to the system interface layout

The system interface for Virtual Lego can be enhanced by adding-in more user guidelines especially for the pinch gloves navigation inside the system interface. Currently, the system only provides the guidelines of the keyboard functions. Providing the users with the keyboard and pinch gloves key functions will probably guide and ease their navigational process of the system application.

REFERENCES

[Exploiting Virtual Reality Techniques in Education and Training: Technological Issues 2005]

<http://www.agocg.ac.uk/reports/virtual/vrtech/intro1.htm>

[All About Stereo Photography 2005]

<http://www.shortcourses.com/how/stereo/stereoimages.htm>

[Stereoscopic Displays and Applications XVII 2005]

<http://www.stereoscopic.org/2006/>

[Virtual Reality 2005]

<http://www.bookrags.com/sciences/sciencehistory/virtual-reality-woi.html>

[LEGO]

<http://en.wikipedia.org/wiki/LEGO>

[Frederick 1999] Frederick P. "What's Real About Virtual Reality?" *Special Report November/December 1999*, Chapel Hill: Univ. of North Carolina.

[William 2003] William R & Alan B. "Understanding Virtual Reality: Interface, Application, and Design." United States : Morgan Kaufmann Publishers.

[Kalawsky 2000] Roy. S. Kalawsky. "The Validity of Presence as a Reliable Human Performance Metric in Immersive Environments". In proceedings of *3rd International Workshop on Presence*, Delft, Netherlands.

[Slater 1996]. Slater, Usoh, Linakis, Kooper, "Immersion, Presence and Performance in Virtual Environments: An Experiment with Tri-Dimensional Chess".

[Sheridan 1992] Sheridan, "Musings on Telepresence and Virtual Presence", *Presence: Teleoperators. And Virtual Environments*, 1(1),120-126, 1992

[Eastgate 2001] Eastgate R. "The Structured Development of Virtual Environments: Enhancing Functionality and Interactivity." *Thesis*. University of Nottingham: Doctor of Philosophy.

[Herndon 1994] Herndon, K. P., Van Dam, A., & Gleicher, M. Workshop on the Challenges of 3D Interaction. *SIGCHI Bulletin*, 26(4).

[Barfield 1995] Barfield, W., & Furness III, T. "Virtual Environments and Advanced Interface Design." Oxford: Oxford University Press.

[Engeli 1996] Engeli M. & Kurmann D. "A Virtual Reality Design Environment with Intelligent Objects and Autonomous Agents." In proceedings of *Design and Decision Support Systems Conference*, Spa Belgium.

[Greenhalg 2002] Greenhalgh, C., Purbrick, J., Benford, S., Craven, M., Drozd, A., & Taylor, I. "Temporal links: recording and replaying virtual environments." In proceedings of *Multimedia 2000*, Los Angeles, California, USA.

[Ruddle 1998] Ruddle, R., Payne, S., & Jones, D. "Navigating Large-Scale "Desk-Top" Virtual Buildings." *Presence*, 7(2), 179-192.

[Mark] Mark R., Frederick P. & Carlo H.. "Moving Objects In Space: Exploiting Proprioception In Virtual-Environment Interaction." Chapel Hill, Berkeley.

[Issacs 2002] Issacs P, Shrag J & Strauss PS. "The Design and Implementation of Direct Manipulation in 3D." In: Course notes from the *SIGGRAPH 2002 conference*, San Antonio, Texas, July 2002.

[Neale 2002] Neale, H., Cobb, S., & Wilson, J. "Designing Virtual Learning Environments for People with Learning Disabilities: Usability Issues." In proceedings of *ICDVRAT 2000*, Alghero, Sardinia, University of Reading, 265-272.

[Kitamura 2002] Kitamura Y., Ogata S. & Kishino F. "A Manipulation Environment of Virtual and Real Objects using a Magnetic Metaphor." In proceedings of *ACM Symposium on Virtual Reality Software and Technology*, Hong Kong, China.

[Jeffrey 2003] Jeffrey J. "Haptic Devices." *White paper*, Seattle, WA.

[Thalmann] Thalmann N & Thalmann D. "Virtual Reality Software and Technology." Geneva, Lausanne, Switzerland.

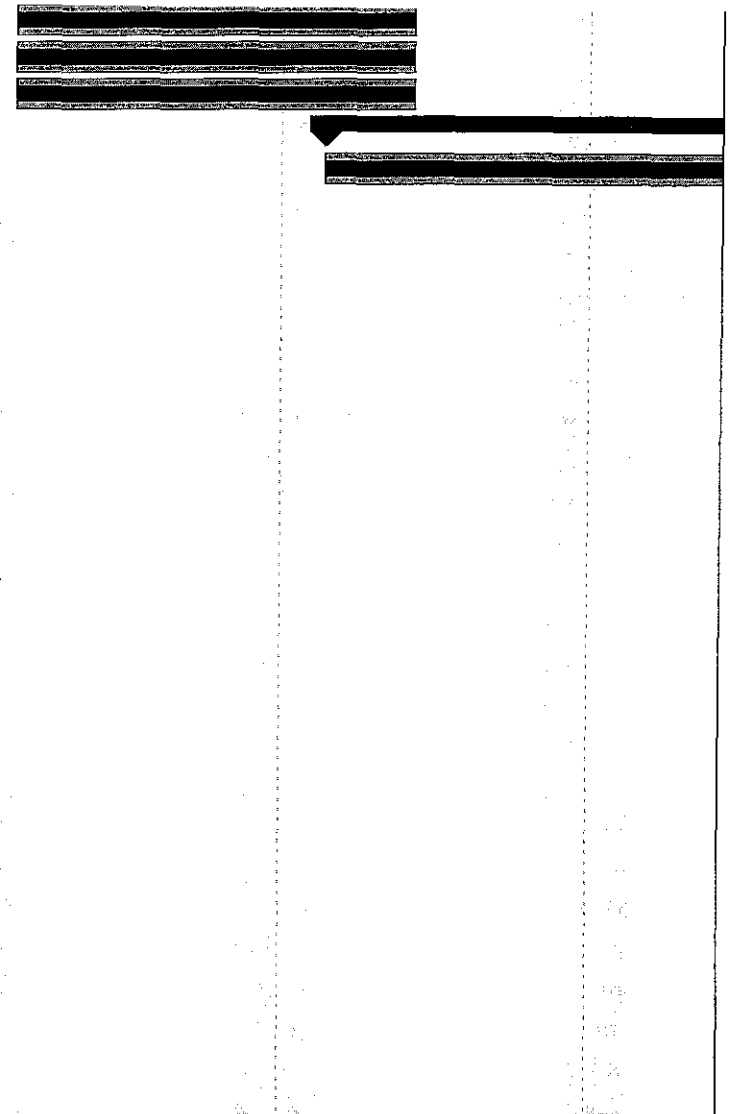
[Jakobsson 1999] Jakobsson M. "Why Bill was killed - understanding social interaction in virtual worlds." In proceedings of the fifteenth *Twente workshop on language technology*, Twente University, Enschede.

[Nigel 2001] Nigel W. & Leng J. "Scientific Applications of Visualization, Virtual Reality and High Performance Visualization Computers."

APPENDIX

GANTT Chart

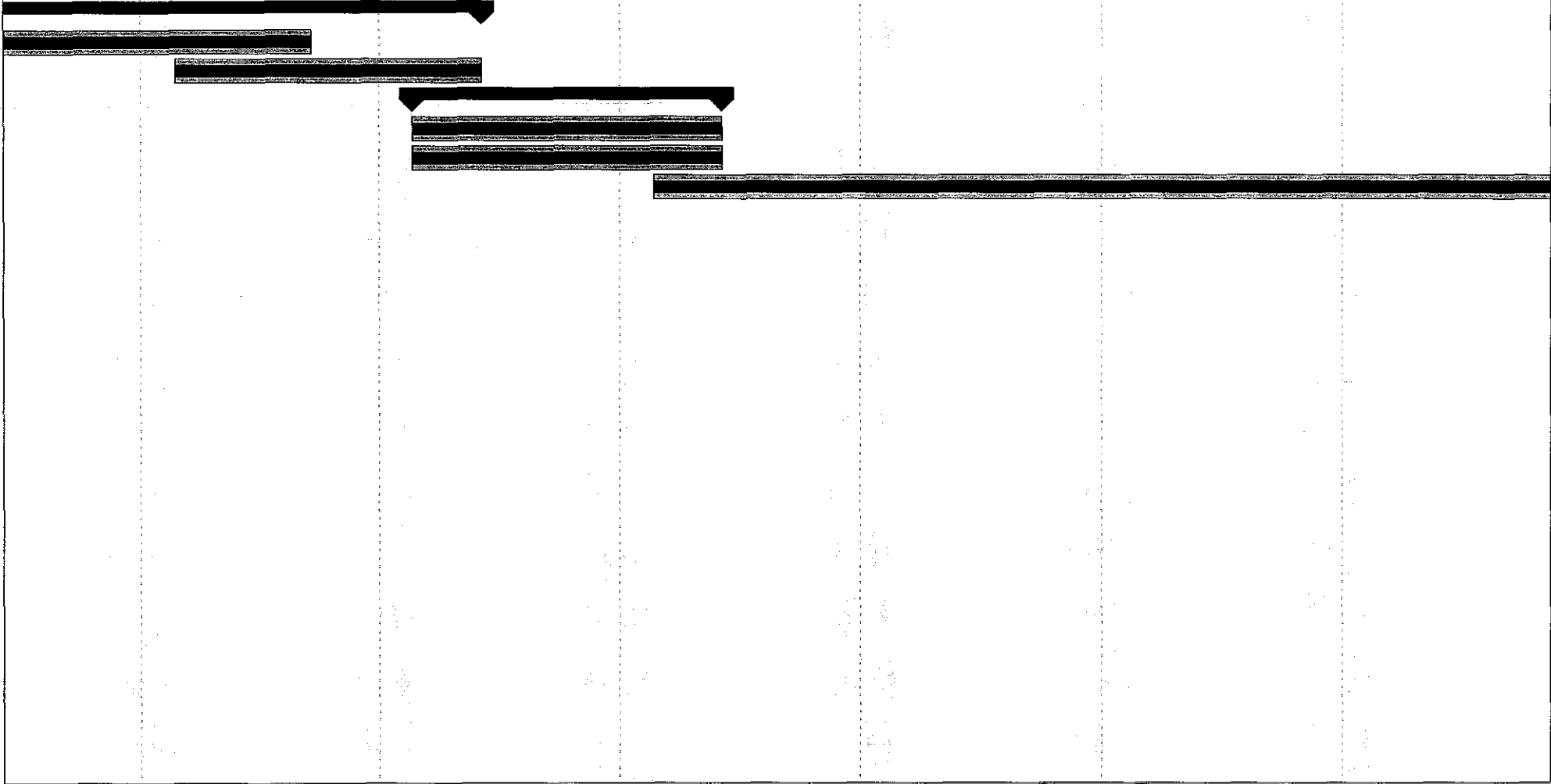
3	✓	Background Study	7 days	Mon 8/1/05
4	✓	Problem Statements	7 days	Mon 8/1/05
5	✓	Objectives & Scopes of Study	7 days	Mon 8/1/05
6	✓	Literature Review or Theory (Chapter 2)	17 days	Mon 8/8/05
7	✓	Journals and Books Collection	14 days	Mon 8/8/05
8	✓	Reviewing & Analysis Writing	7 days	Mon 8/22/05
9	✓	System & User Requirements	7 days	Mon 8/29/05
10	✓	Functional Requirements	7 days	Mon 8/29/05
11	✓	Non-Functional Requirements	7 days	Mon 8/29/05
12	✓	2. SOFTWARE DESIGN / METHODOLOGY	21 days	Mon 9/5/05
13	✓	Hardware & Software Requirements	7 days	Mon 10/3/05
14	✓	Hardware Acquirements (if any)	7 days	Mon 10/3/05
15	✓	Software Installation (if needed)	7 days	Mon 10/3/05
16	✓	System Model Design	7 days	Mon 10/10/05
17	✓	Design System Architecture Diagram	7 days	Mon 10/10/05
18	✓	Design UML Diagram	7 days	Mon 10/10/05
19	✓	Design System Flow Diagram	7 days	Mon 10/10/05
20	✓	Design Interface Design	7 days	Mon 10/10/05
21	✓	REPORT & DOCUMENTATION	1 day	Fri 11/4/05
22	✓	Submission of Interim Report	1 day	Fri 11/4/05
23	✓	Introduction & Overview	1 day	Fri 11/4/05
24	✓	Literature Review	1 day	Fri 11/4/05
25	✓	Prototype Presentation	1 day	Fri 1/6/06
26	✓	3. PROGRAMMING & RE-ENGINEERING	79 days	Tue 1/24/06
27	✓	Tasks	54 days	Mon 1/23/06
28	✓	Primitives and Virtual Environment	5 days	Mon 1/23/06
29	✓	Keyboard Functions	3 days	Sat 1/28/06
30	✓	Collision Detection	18 days	Thu 2/2/06
31	✓	Networked Virtual Reality	25 days	Tue 2/28/06
32	✓	Pinch Gloves Functions	3 days	Tue 4/4/06



Project: VIRTUAL LEGO Date: Tue 6/20/06	Task		Milestone		External Tasks	
	Split		Summary		External Milestone	
	Progress		Project Summary		Deadline	

34	✓	Submission of Progress Report	1 day	Fri 2/17/06
35	✓	Methodology (Chapter 3)	1 day	Fri 2/17/06
36	✓	4. TESTING & DEBUGGING	22 days	Thu 2/23/06
37	✓	Alpha Testing	7 days	Thu 2/23/06
38	✓	Beta Testing	10 days	Mon 3/13/06
39	✓	5. OPERATION & MAINTENANCE	35 days	Mon 3/27/06
40	✓	Report & Documentation Writing	5 days	Mon 5/8/06
41		Submission of Dissertation Final Draft	1 day	Fri 5/26/06
42	☐	Result & Discussion (Chapter 4)	1 day	Fri 5/26/06
43	☐	Conclusion & Recommendation (Chapter 5)	1 day	Fri 5/26/06
44	☐	Oral Presentation	2 days	Fri 6/16/06
45	☐	Submission of Project Dissertations	1 day	Tue 6/20/06

Project: VIRTUAL LEGO Date: Tue 6/20/06	Task		Milestone		External Tasks	
	Split		Summary		External Milestone	
	Progress		Project Summary		Deadline	



Project: VIRTUAL LEGO
Date: Tue 6/20/06

Task



Milestone



External Tasks



Split



Summary



External Milestone



Progress



Project Summary



Deadline



[Redacted]

[Redacted]

[Redacted]

[Redacted]

Project: VIRTUAL LEGO
Date: Tue 6/20/06

Task [Redacted]
Split [Redacted]
Progress [Redacted]

Milestone [Redacted]
Summary [Redacted]
Project Summary [Redacted]

External Tasks [Redacted]
External Milestone [Redacted]
Deadline [Redacted]

Project: VIRTUAL LEGO
Date: Tue 6/20/06

Task



Split



Progress



Milestone



Summary



Project Summary



External Tasks



External Milestone



Deadline



Project: VIRTUAL LEGO
Date: Tue 6/20/06

Task

Split

Progress



Milestone



External Tasks



External Milestone

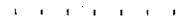
Deadline

Project: VIRTUAL LEGO
Date: Tue 6/20/06

Task



Split



Progress



Milestone



Summary



Project Summary



External Tasks



External Milestone

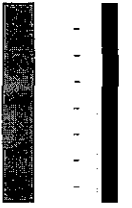


Deadline



Project: VIRTUAL LEGO
Date: Tue 6/20/06

Task
Split
Progress



Milestone
Summary
Project Summary



External Tasks
External Milestone
Deadline

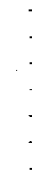


Project: VIRTUAL LEGO
Date: Tue 6/20/06

Task

Split

Progress



Milestone

Summary

Project Summary



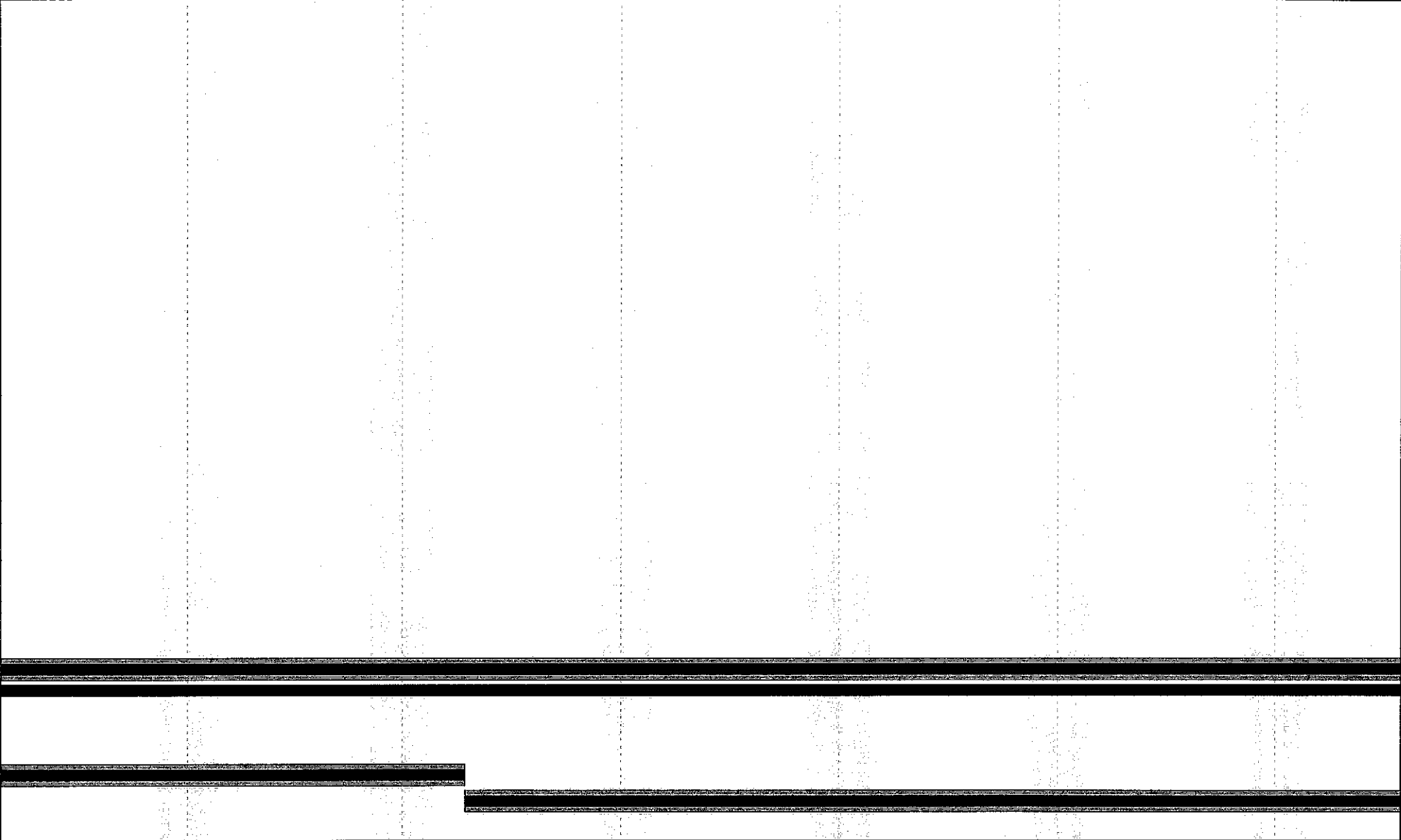
External Tasks

External Milestone

Deadline






M T W T F S S M T W T F S S M T W T F S S M T W T F S S M T W T F S S M T W



Project: VIRTUAL LEGO
Date: Tue 6/20/06

Task 
Split 
Progress 

Milestone 
Summary 
Project Summary 

External Tasks 
External Milestone 
Deadline 



Project: VIRTUAL LEGO
 Date: Tue 6/20/06

Task



Milestone



External Tasks



Split



Summary



External Milestone



Progress



Project Summary



Deadline



Project: VIRTUAL LEGO
Date: Tue 6/20/06

Task



Milestone



External Tasks



Split



Summary



External Milestone



Progress



Project Summary



Deadline




Project: VIRTUAL LEGO
Date: Tue 6/20/06

Task 

Split 

Progress 


Milestone 

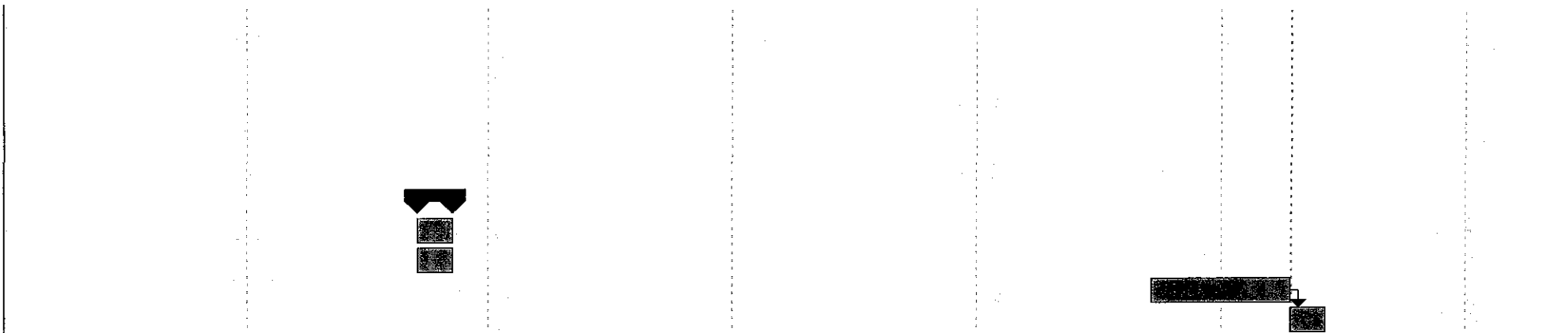
Summary 

Project Summary 

External Tasks 

External Milestone 

Deadline 



Project: VIRTUAL LEGO Date: Tue 6/20/06	Task		Milestone		External Tasks	
	Split		Summary		External Milestone	
	Progress		Project Summary		Deadline	