Cinema E-Ticketing Self-Service System

By

Maatlou Precious Radingwana

Dissertation submitted in partial fulfillment of
the requirements for the
Bachelor of Technology (Hons)
(Business Information Systems)
JULY 2007

Universiti Teknologi PETRONAS
Bandar Seri Iskandar
31750 Tronoh
Perak Darul Ridzuan

CERTIFICATION OF APPROVAL

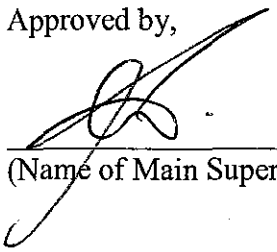
Cinema E-Ticketing Self Service System

By

Maatlou Precious Radingwana

A project dissertation submitted to the
Business Information Systems Programme
Universiti Teknologi PETRONAS
In partial fulfillment of the requirement for the
BACHELOR OF TECHNOLOGY (Hons)
(BUSINESS INFORMATION SYSTEMS)

Approved by,



(Name of Main Supervisor)

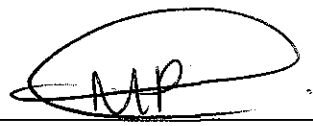
UNIVERSITI TEKNOLOGI PETRONAS

TRONOH, PERAK

July 2007

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 references and acknowledgements, and that the original work contained herein have not been undertaken or done by unspecified sources or persons.



MAATLOU PRECIOUS RADINGWANA

ABSTRACT

In a world of wide-spread technology usage, standing in a queue for half-an-hour or more is what any customer would hate. There really isn't any enjoyment in waiting in a queue for hours on end. Eventually customers become irritable of all that waiting they decide to leave The Cinema E-Ticketing Self-Service Project commenced in January 2007 and its work was completed in September 2007. The original objective of the project was to investigate and hopefully develop a self-service ecommerce-based system in particular a cinema self service system. The objectives of self-service systems are relative to the kind of self-service system being created. The most common example would include many gas stations, where the customer pumps their own gas rather than have an attendant do it. In this context, the Cinema E-ticketing Self-Service System's specific objectives were relative i.e. to make the system user friendly, to be easily accessible, and to provide movie information and slots also to provide a seating schema for selection. Lastly it should provide a payment process simulation and print a ticket. The Cinema E-ticketing Self Service system will therefore encompass the efficiency of a self-service system. The ticket printing functionality will give more convenience to the user.

The methodology includes the procedure identified to implement the system. The waterfall model was used. The model carefully and gradually refines each phase of the system's development lifecycle. These are requirements gathering, analysis, design, development, the testing and maintenance. The report concludes with recommendations such as: There is a need for self-service systems to be developed, to assist staff or workers i.e. to reduce the load of work. Serious consideration must be taken into security issues and providing consumers with some level of security should be a priority. Steps should be taken to encourage better physical design of self-service systems. Instead of designs that repel customers and complex interfaces. Since self service is growing rapidly, everyone should be catered for especially the handicapped. Disabled users, especially the blind should be considered and steps should be taken in future to adapt self-service systems for their use. Encouragement should be given to the development of more self-service systems, which would promote the usage of such systems by the public in order to familiarize society with electronic commerce systems.

ACKNOWLEDGEMENTS

The author would first like to thank God for the strength and support throughout the creation of this project, from the conception to the development and delivery of this system. I would also like to acknowledge and thank the following individuals that provided an enormous amount of help towards the completion of this project. These individuals have provided their time and knowledge in order to lend a hand in this project.

I would like to thank Ms Emy Elyanee Mustapha for her supervision and guidance throughout the whole final year project process. Mr. Dicky Ekklesias, a fellow postgraduate student and friend, who unselfishly used his vast knowledge and skills to guide me through the development of the system and with the greatest gratitude I thank him.

Not forgetting fellow colleagues, friends and family, for being my pillars of support and my source of motivation throughout all the semesters.

TABLE OF CONTENTS

Page

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

1. INTRODUCTORY SUMMARY

E-ticketing is evolving at a rapid speed and a lot more systems will be seen in development. It has become the new venture for businesses to provide service to their customers as it is considered as saving time and costs. The Cinema E-ticketing Self Service system will therefore encompass the efficiency of a self-service system. The ticket printing functionality should be able to give more convenience to the user. The flow of the system will be direct and user-friendly; and hopefully users will not experience any difficulties in learning how to utilize the system for their effectiveness.

1.1. BACKGROUND OF STUDY

From time to time we hear the constant complaints of customers who demand fast and quick service. In this technology driven world, no individual wants to stand in line for more than 15 minutes awaiting service. People especially customers prefer convenience and time-saving service. This is where the self-service system comes in, it is a system that provides quick and easy service that you do yourself instead of having to wait in a queue. The Cinema E-Ticketing Self-service system is one kind of those systems, one that could make the experience of going to buy a movie ticket much more pleasant. The system provides a user friendly interface that's also easy to navigate during interaction. Self-service systems are now being created all over the world. Major companies are beginning to see the benefits such systems bring forth to both the service provider and the customer. Therefore, this study further provides an in-depth into how and exactly why are self service systems can be developed and particularly there s a need for them.

1.2. PROBLEM STATEMENT

In a world of wide-spread technology usage, standing in a queue for half-an-hour or more is what any customer would hate. There really isn't any enjoyment in waiting in a queue for hours on end. Eventually customers become irritable of all that waiting that they decide to leave. Wouldn't a self-service be efficient in addressing this problem? In order to curb such a problem, a self-service system would be an appropriate solution. The project's identification was a result of the problems that customers face in a receiving service, particularly in ticket purchasing. In this project the author focuses on the benefits the project will give, therefore saw the significance in providing a fast and easier method of creating convenience for the customer.

1.3. OBJETIVES OF SELF-SERVICE

The common definition found of the self-service concept was that it is the practice of serving oneself, usually when purchasing items; may be it over the internet or on a remote system.

Objectives of self-service systems are relative to the kind of self-service system being created. The most common example would include many gas stations, where the customer pumps their own gas rather than have an attendant do it. The Cinema E-ticketing Self service system will be a remote ecommerce system that will be utilized to assist the ticketing counter personnel.

Below are the summary of some of the objectives of Cinema E-ticketing Self- Service Systems: these can be identified as the primary objectives for the provision of a new self-service system:

- To be user-friendly.
- To be easily accessible.
- To make available movie information and slots.
- To provide a seating schema for selection.
- To provide payment process simulation.
- Should print a ticket.

1.4. SCOPE OF STUDY

The scope focused on self-service particularly cinema hence being called a Cinema E-ticketing self-service system. The system's focus was on cinema e-ticketing through self-service i.e. providing tickets on a remote system. It would be a self-service system that allows the user to choose a movie which they'd like to watch, select a movie slot and preferred seat, the user can then choose to pay using cash, credit card or debit card and finally when the transaction is made, the system should print out a ticket.

1.5. PROJECT RELEVANCY

The purchasing of tickets through this self-service system would make buying a ticket more convenient since the tickets would be printed at the point of sale i.e. immediately after purchase. The Cinema E-ticketing self-service system was meant to serve as a ticket printing self-service kiosk located close to the ticket counters in-house. The tickets can only be purchased in-house and not on the internet. For an existing example of this kind of system see Figure 4 and 5, below the Literature Review.

1.6. FEASIBILITY OF THE PROJECT WITHIN SCOPE AND TIME FRAME

Since the project will focus on the design, functionality and the service provision. The author worked within the Time Frame in shown in **Figure 1**.

Table 1: Monthly Project Time Frame

TASK	DESCRIPTION	PERIOD / DURATION
Planning	Kick-off project with requirements gathering	**1 Month**
Design	User friendly interface	**2 Month**
Implementation, Programming and Development	Coding of system back-end Interface Development Complete project	**3 Months**
System Testing	Run integration testing before handover	**2 Months**
System Presentation	Present Final product	**Week 9**
TOTAL		**6 Months**

CHAPTER 2

2. LITERATURE REVIEW/THEORY

2.1. SELF-SERVICE TECHNOLOGY

Lee and Allaway compiled the results of a study on the emerging role of self-service technologies such as airline ticketing machines, automatic teller machines, and computer-based shopping services in the strategic offering of service providers. (Lee J., & Allaway A., 2002)

SSTs allow (or force) consumers to help produce their own service encounters via machine interaction rather than by interacting with a firm's service personnel. Firms which introduce SSTs wish to gain rapid acceptance and usage of these technologies by potential consumers. (Lee J., & Allaway A., 2002)

The study investigated whether the provision of more personal control to consumers can reduce their perceived risk, enhance the perceived value of the SST, and encourage greater adoption intention associated with the innovation. A set of managerial implications and recommendations were provided as well. (Lee J., & Allaway A., 2002)

2.1.1. What drives consumers to shop online?

Monsuwe, Dellaert and Ruyter tell of how a large number of consumers in the US and Europe frequently shop on the Internet, it is a research on what drives consumers to shop online. It was produced to propose a framework to increase researchers' understanding of consumers' attitudes on and their intention toward making online purchases. The framework uses the constructs of the Technology Acceptance Model (TAM) as a basis, extended by exogenous factors and applies it to the online shopping context. (Monsuwe et al., 2004)

This report indicates that attitudes toward online shopping and intention to shop online are not only affected by ease of use, usefulness, and enjoyment, but also by exogenous factors: like consumer traits, situational factors, product characteristics, previous online shopping experiences, and trust in online shopping. (Monsuwe et al., 2004)

2.1.2. Self-service cash register system

The self-service systems have already gained a great success in Western-Europe. In the countries where this type of service is used, this method has already become a part of shopping. The system combines the functions of the traditional cash registers, containing all their elements, supplemented with the most up-to-date achievements of our time. The essence of this technology is that the buyers themselves are operating the self-service cash register, reading in separately (by means of intelligent bar codes) the articles bought by them, then finally pay the amount indicated by the machine in whichever way of payment.

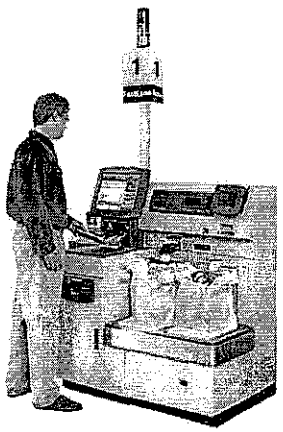


Figure 2.1: Example of a self service terminal

The self-service terminals have several advantages in comparison to the traditional self-service system. The solution applies not only the most up-to-date technique of the age (color touch screen) but it is of a favorable design in respect of ergonomy. Various

payment possibilities (banknotes, coins, cards), error-tolerant operation, control station, scales, spatial bar code reader, all these form integrated parts of the system.

Easy manageability (informatics skill, computer operation knowledge shall not be required) is also important for the developers and users, since this exerts a great influence to the speed of the service. Surveys have confirmed that one of the most frequent complaints of customers is the queuing up during shopping and/or purchasing. It also goes without saying that the possibility of traditional payment in the store is maintained for those who prefer it.

The self-service system is an excellent solution for the commercial chains, to whom the buyers' satisfaction is important and who would like to attract further masses of customers serving them quickly and smoothly. It is evident that self-service systems make any e-transactions easier for buyers.

2.2. E-TICKETING

In April 2007, an article was written on Malaysia Airlines, which is said to be on track to roll out electronic ticketing (eTicketing) by September this year with the completion of its customer airlines' check-in system cutover to the new SITA Departure Control System. ()

This allows the 15 airlines including Jet Airways, Royal Brunei and Kuwait Airways, which are served by Malaysia Airlines as the ground handling agent, to check in eTicket passengers along with paper ticket customers. This cutover for the customer airlines represents phase one of Malaysia Airlines' RM400 million Passenger Services System rollout to upgrade all IT infrastructure to facilitate the airline's move to the International Air Transport Association's (IATA) standard eTicketing.

eTicketing will eventually simplify traveling experience, offering customers a number of check-in options including via the internet and self-service kiosks located at the airport. The next phase to enable Malaysia Airlines to be fully eTicketing capable involves the

system cutover for its 16 domestic stations beginning with Alor Setar. The completion date is targeted for early July after which the international stations will begin their rollout and complete it by August.

Work on this venture will begin on implementing the new eTicketing platform for Malaysia Airlines' travel agents in Malaysia, and across the globe with completion targeted for early September 2007.

2.2.1. Problems with Self-Service Checkout Systems

Self-service checkout systems have to be flawless. It only takes one bad experience to create an ex-customer. Self-checkout terminals are increasingly employed by retailers—particularly grocers. But many consumers remain unconvinced about the convenience of checking themselves out. How well an organization fields the technology can make the difference between whether customers walk away from the experience enthusiastic or discouraged. There following our problems that could be detrimental to the success of a self-service system. (Carr .F.D, 2004)

Erroneous or missing data can be a show-stopper for a self-service consumer. This can be resolved by ironing out the kinks at manned cash registers, before you open up self-checkout lanes. Self-checkout scanners read a bar code on every product; and for every bar code there's a price in the store database. "It's a lot easier for a cashier to overcome a 'not on file' error than it is for a self-service [customer]," says Dusty Lutz, product manager for NCR's FastLane self-checkout systems. (Carr .F.D, 2004)

Analyze also how well the technology will fit into existing store operations, says Roy A. Garver, a consultant who has been involved in self-checkout implementations at Wal-Mart, Kroger, Home Depot and other companies. Retailers that fail to make that effort upfront will suffer through more problems after the technology is deployed; Garver says that in the process this will damage the customers' perceptions of the technology. (Carr .F.D, 2004)

Ease-of-use is critical—and largely under the control of the vendor. This can be resolved by doing everything possible to make your consumer's first encounter positive. "It only takes one bad experience to turn a user into a non-user of self-service," NCR's Lutz says. Consumers who become confused or embarrassed about their inability to complete a transaction may never come back. Stores can make systems friendlier with clear signs describing the steps in the process particularly those steps that aren't obvious, such as placing produce on a scale during checkout. (Carr .F David, 2004)

Jakob Nielsen, a human-factors designer, says brick-and-mortar stores ought to offer more clues as to how things work. But sometimes the physical design of a checkout unit undermines the application, he says. "Like the screen will say, 'Press "OK"' to complete your purchase,' but there's no 'OK' button it says 'Yes' or 'Enter' instead." (Carr .F.D, 2004)

Your operations may not work the way your customers do. To resolve this problem you should think through what the customer and the system has to do. Self-checkout requires each item to be individually scanned. Deli items will have to be identified by bar codes instead of hand scrawl. Shortcuts such as scanning a single jar of baby food 20 times to represent 20 items will have to be rethought. (Carr .F.D, 2004)

Don't let store personnel ignore customers in the self-service lane. The resolution to this problem is to assign a friendly face to the technology, says Greg Buzek, principal analyst with IHL Consulting Group in Franklin, Tenn. Kroger employs some of its most experienced cashiers to greet customers, show them how to get through the process and overcome any obstacles. After all, showing customers how easy self-checkout can be is a good investment. "People don't like to feel stupid or [as if] they did something wrong," she says, so you have to get them over that hurdle. (Carr .F.D, 2004)

2.2.2. Malaysian Airlines investing in electronic ticketing

Malaysia Airlines (MAS) is investing RM200mil in the next three years in a new passenger service system that will - among other things - enable the airline to be fully electronic ticketing (eTicketing) capable by September. (Tahir .S, 2007)

MAS senior general manager Dr Amin Khan said the outlay was mainly for putting up the infrastructure for eTicketing reservation and electronic check-in infrastructure.

Figure 2.3 :Service check-in kiosk at the KL International Airport

"By September, MAS customers will have two check-in options - via the self-service check-in kiosks at the airport or the normal check-in counters," he said at a media briefing yesterday. (Tahir .S, 2007)

"Customers will also enjoy the ease of printing their own tickets and the option of making changes to their travel itinerary themselves, thus not having to visit the ticketing office or travel agent," he added. (Tahir .S, 2007)

eTicketing and electronic check-ins are the essence of MAS' multi-million ringgit information technology programme to comply with International Air Transport Association's mandate that calls for all air travel to be ticketless by 2007.

The national airline expected savings of RM70mil a year by implementing eTicketing.

"Check-ins will also be easier as passengers will just have to show their e-ticket number and a form of identification. There will be no more problems associated with misplaced, damaged or forgotten tickets," he said. (Tahir .S, 2007)

MAS eTicketing's introduction is on track following its successful cutover to the new user-friendly SITA Departure Control System (DCS) for its 16 domestic stations, including the KL International Airport (KLIA), to hasten passenger check-in and boarding process, replacing the system the airline has been using for the last 20 years.

Amin said the airline had also introduced the new check-in system at the Heathrow Airport in London and Changi Airport in Singapore. The system cutover for the rest of the international stations started yesterday and will be completed by September. (Tahir .S, 2007)

2.2.3. How Ticketing kiosks are changing the cinema experience

The line to buy movie tickets from the cashier is long and moving at a snail's pace, but the self-service ticketing kiosk goes unused. (Bickers J. & Richmond F., 2006)

"In the overall scheme of things, the vast majority of tickets are sold at the theater to people waiting at the box office," said Terrell Falk, vice president of marketing and communications at Cinemark, which operates 2,700 cinema screens in 13 countries. "It is a growing percentage of the tickets that are sold on the Internet or via the kiosks, but it is still not the majority way yet. You need to look at this in the future. Eventually these (kiosks) may replace box offices totally." (Bickers J. & Richmond F., 2006)

But even if they don't replace the box office, they are still serving their purpose — moving more patrons through the lines at a faster rate, especially when it matters the most. (Bickers J. & Richmond F., 2006)

Falk said she is satisfied with kiosk usage at Cinemark, understanding that it will continue to grow over time. Today, Fridays and Saturdays are the peak times for kiosk use because long lines prompt customers to try something new. But over time, users will become familiar with the machines and will increasingly use them throughout the week. (Bickers J. & Richmond F., 2006)

2.3. RELATED IMPLEMENTED SYSTEMS

Computicket is a South African company that sells tickets for almost every cultural and sporting event in the country (including movies). These tickets can be purchased through Computicket online. This system can be found at www.computicket.co.za.

Meanwhile, Ster-Kinekor is another cinemas site that offers a service that allows customers to reserve movie tickets over the Internet. Once reserved, the tickets can be collected from the appropriate cinema before the show. It can be found if users surf www.sterkinekor.com.

Customers that visit the Ster-Kinekor web site find relevant information on the movies currently being screened at Ster-Kinekor cinemas. Those interested in reserving tickets for one of these movies visit the ticket sales page. From this page the customer selects the relevant cinema complex, movie, date and time and the number of tickets. The system attempts to automatically select the best available seats for that show. The customer is then called to enter personal and credit card details that are submitted to the ticketing system. If authorization is received from the card-issuing bank, the customer is informed of the successful transaction.

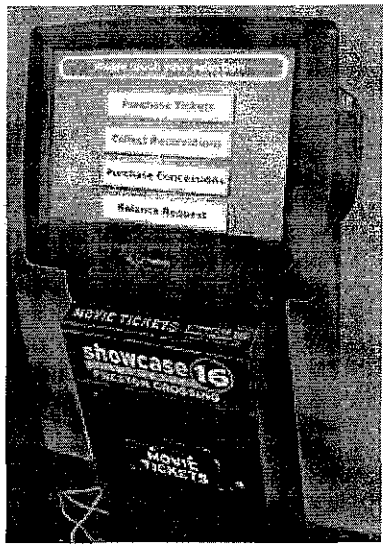


Figure 2.2: Cinema E-ticketing kiosk

Matt McNeill, chief executive of online ticketing provider eTickets.to, said education is key to driving the adoption of self-service ticketing.

"The natural users of those kinds of self-service systems seem to be of the younger generation that is used to using computer-based systems. They tend to naturally gravitate toward (self-service) systems," McNeill said. He stressed the importance of "educating the people that don't necessarily grasp ... the benefits, like lack of queuing and the level of choice that (self-service) provides."

While McNeill concedes that older customers may have more of a tendency to avoid self-service technology, he also believes that "where you have more effective client education going on, that barrier kind of disappears. It's a matter of talking up all the benefits through all the standard marketing channels."

According to Dave McCracken, president of self-service software company Livewire International, general venues like supermarkets prepare the client population and pave the way for more specialized venues.

McCracken described a Livewire application that involves clients who are "more advanced in years," a couple who run a ski lift. "They were reluctant to put a kiosk in to

begin with, because they weren't that up on technology, but they learned pretty quickly the benefit of it, and they're sold on it today. Because they're older, they recognized that a lot of the older people are unfamiliar with it, and a little sensitive to using a kiosk, and the owners would actually go over, and walk customers through the process of learning it. There, for example, where you actually have people year after year going skiing, they're familiar with it, they have no problem utilizing it."

The design of the interface display is important, McCracken said, in making sure systems are user-friendly. "We have a nice clean interface, trying to keep it nice and simple, always displaying the shopping cart, all the things we did to make it easy to walk people through the process. We've stayed consistent with that from system to system, and it's just proven very successful."

Venue operators themselves agree that education is key. "We really haven't experienced any resistance or difficulty using these (kiosks) at all as long as when you introduce them you have somebody there to help people," said Cinemark's Falk.

As with any other self-service technology, placement and logistics are key determinants of a project's success or failure. Whitney pointed out a key rule of thumb that seems intuitive, and yet is ignored by some theater owners — the kiosks must be placed in an area that is trafficked by patrons between the parking lot and the box office.

"In fact, the more the kiosk is visible to the consumer the more it is used," he said. "We turned one kiosk 90 degrees so that instead of the side of the unit facing the traffic flow, the screen faced the traffic flow, and without any other changes the usage for that device improved 134 percent."

He said cinema owners are also wise to deploy personnel to stand by the machines during the first few weekends, not only to teach customers how to use the machine but to draw attention to them in the first place.

The value proposition for self-service ticketing is strong, and will only become more so over time: According to a joint study between Radiant Systems and Georgia Tech University, three kiosks replace the output of two traditional, staffed box office locations.

2.4. INFORMATION ON KIOSK APPLICATIONS

A kiosk project is a user application, which focuses on user interface and design. Application development is also dependent upon the purpose of the kiosk. Many deployers will have the option of deploying an existing application such as a company website, or creating an entirely new application.

Whether the application exists or new, it should be short and simple as typical kiosks require the user to stand during use. Users should be able to easily navigate to the information needed within a relatively short period of time.

The design of the application should also consider whether the kiosk utilizes a touch screen or a trackball. If the kiosk uses a touch screen, then links and buttons should be large enough for a human finger to touch. This is especially important if deployers are using an existing application where the links are currently only large enough for a computer user's mouse to click.

If you choose to create a new application, experienced kiosk developers can custom-develop a kiosk application specific to your needs. Consulting with a kiosk application developer will help you uncover the user interface issues related only to kiosks. Additionally, there are numerous types of kiosk applications, be sure to find a developer that has detailed experience for the intended use of your kiosk. For example, if your kiosk is a ticket-dispensing kiosk, you would not want to work with a company that has only developed applications for digital signage kiosks.

There are many good reasons for porting a web application to a kiosk. These include the ability to re-use an existing web application and the ease of finding development resources for web applications. Web applications also provide the flexibility to place the processing power either locally or remotely based on the need.

2.4.1. Kiosks and Customer Self Service

Touch screen technology, by enabling kiosks that are user friendly, reliable, robust and attractive can help businesses in many markets realize their objectives and meet obligations to shareholders and stakeholder groups. Potential applications include:

- Presenting product information
- Ticket issue
- Check-in
- ATM and cash dispenser
- Bill payment and financial transactions
- Gaming and betting
- Internet access
- Food service and ordering
- Photo booths
- Voting
- Loyalty cards and special offers

Perhaps the most fundamental advantage of the touch screen in such applications is its ability to offer an exciting and rewarding user experience. A well designed touch screen interface can present a compelling proposition to potential customers passing by a kiosk, whether located in a busy shopping centre, cinema multiplex, convenience store, railway station or air terminal, a booking hall, a fuel station forecourt; the possibilities are endless. Even those who are not familiar with computer technology, or have had previous bad experiences, can be attracted to a bright, friendly, ergonomically-designed touch screen; the chances are that they will not even realize they are interacting with a computer.

While the chief concerns lie with application development, to create slick, attractive user interfaces, kiosk designers also need to select the optimum touch screen technology for the application.

Kiosk designers can select technologies that are suitable for outdoor deployment. They have high resistance to damage from key scratches, spilled drinks or other accidental or malicious damage, are able to operate continuously over extended periods without maintenance, can be activated by operators wearing gloves or tapping the screen with a ball pen, and are clearly visible under any lighting level from direct sunlight through artificial lighting to night time use.

In fact, the touch screen technologies now available allow designers to conceive kiosk solutions to virtually any retail or information delivery requirement. Low maintenance is particularly important to businesses that operate kiosks, as the cost of field service visits is high and can significantly damage the profitability of the enterprise.

Kiosks featuring a touch screen instead of a keyboard or mouse are also inherently more reliable, with fewer failure mechanisms and less frequent maintenance or replacement intervals. And, just as in the POS application, the touch screen also prevents access to barred functions. This is particularly important where the kiosk provides secure payment facilities, online booking services, or Internet access, for example.

2.4.2. Usage of Special Printers in E-ticketing

Printers are one of the most mechanically complex components of a self-service device - and one of the most important.

"The essential component of any self-service device is leaving the customer with some tangible result of their experience," said Frank Beurskens, co-founder and chief executive of ShoptoCook, provider of a standalone self-service solution that gives grocery shoppers "fast, fun and fulfilling meal ideas relevant to each shopper's individual needs while they are in the store." When Beurskens began evaluating printers for ShoptoCook's recipe kiosk, he had three criteria:

1) Durability of the printer in terms of its mechanical parts,
2) Sophisticated drivers that allow remote management of the printer and the ability to have a log file associated with the printer, and
3) Minimal required maintenance.

Experts say the benefits of printers at kiosks must be balanced against the necessity and expense of having maintenance and service available. One undisputed fact emerges from conversations about printers – they break down. Therefore when choosing a printer for a kiosk system, it is important to know which is best suited for your particular system.

In the case of the author s system, a kiosk printer couldn't be used because of the lack of availability. The author had to resort to using a Hewlett Packard document printer to provide the user with a tangible result of his or her experience.

"The vast majority of printers deployed in kiosks are thermal printers, though certain applications lend themselves well to laser printers, especially those in which higher quality and/or color printing is required."

- Francie Mendelsohn, president, Summit Research Associates Inc.

CHAPTER 3

3.1 RESEARCH METHODOLOGY

The self-service project operated mainly through a kiosk interface, involving both the customer i.e. the user and the system. However, the first part of the project also involved research on self-service systems, providing the opportunity to examine specifications, requirements and functionalities of such a system at first hand.

Whilst researching on the project the author came across a number of consumer issues with regards to self-service systems. Consumers are concerned about the following issues:

- **Ease of Use:** a general finding is that unless a self-service system is easy to use it will not be used. Customers will seldom try a system more than a very few times before giving up and it will be difficult to get them to try again even if the system is modified.

- **Reliability:** self-service systems must operate reliably. If the machine is frequently out of order then they will avoid using it.

- **Privacy:** most users want to operate the system in privacy, not only to protect their private data but to avoid being overlooked should they make a 'mistake'.

- **Data Protection:** customers are now aware of data protection issues and are concerned that no unauthorized person should be able to access their personal details.

- **Public Awareness:** there is a threshold to be achieved where public awareness and acceptance of a new self-service system makes the proportion of use of that channel economic. Steps have to be taken to raise awareness to and beyond this level. System designers need to be aware of existing levels of public awareness so as to exploit them - for example by making a ticket issuing machine resemble the design of an ATM like the Cinema E-ticketing Self-service system.

- **Costs and Charges:** great care has to be taken over charging regimes, which can fundamentally alter user behaviour.

3.2. PROCEDURE IDENTIFICATION

The waterfall model derives its name due to the cascading effect from one phase to the other as is illustrated in **Figure 3.1**. In this model each phase well defined starting and ending point, with identifiable deliveries to the next phase.

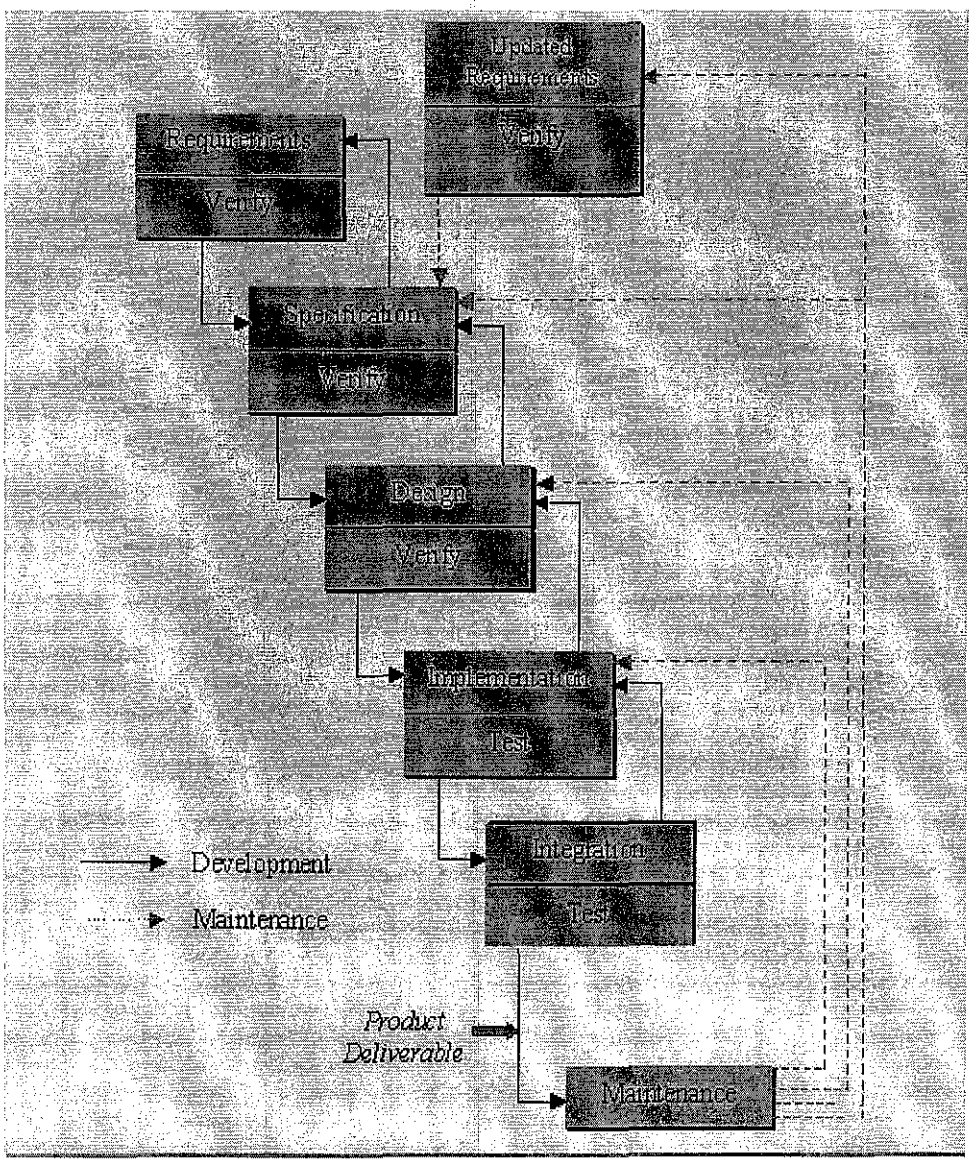


Figure 3.1: Graphical Illustration of the Waterfall Model

The methodology that'll be used will involve the SDLC:

1. Requirements gathering
2. Analysis
3. Design – **must be a User-Centered Design**
4. Development / Coding
5. Testing
6. Error checking / Maintenance

3.3. PROJECT ACTIVITIES

3.3.1. Requirements Gathering

First, requirements of the system were gathered through the author's research method, which involved mostly the internet as the primary source and the author happened to come across the Fuji Photo Printing kiosk in Genting Highlands, Malaysia whereby the staff on-duty was questioned about that specific system, then a demonstration of the system's functionalities and navigation was given. The aforementioned experienced helped provided a great deal of insight and became crucial information before the start of the design process. Requirement gathering gave a lot of what the system needed to have and what functionalities were important to include, in order to achieve the objectives set out, the system's requirements needed to be clear with no ambiguity. Thereafter the author formulated a list of the system's requirements which are listed below:

- User should be able to purchase tickets on the self-service system.
- System should be easy to navigate through.
- System should be user friendly.
- User shouldn't spend more that 10 to 15 minutes during use.
- System should print out a ticket for the user.
- System shouldn't be hard to implement for the developer.

3.3.2. Analysis

Following the requirements gathering and the analysis of existing systems, a visit to the local cinema was imminent in order to take a closer look at the as-is system, which included the counter personnel. Cinema websites that offered online ticket booking or purchasing services such as Sterkinekor, and TGV were analyzed.

Though the Cinema E-ticketing Self Service system sells tickets electronically, it does not share the same objectives as these website systems. This project provides its services on a remote system. Wherelse the above mentioned sites operate online i.e. over the internet but they were a food source of information and guidance towards the refining of the author s system's processes. Figure 3.2 below shows the cinema e-ticketing self service system's use case diagram.

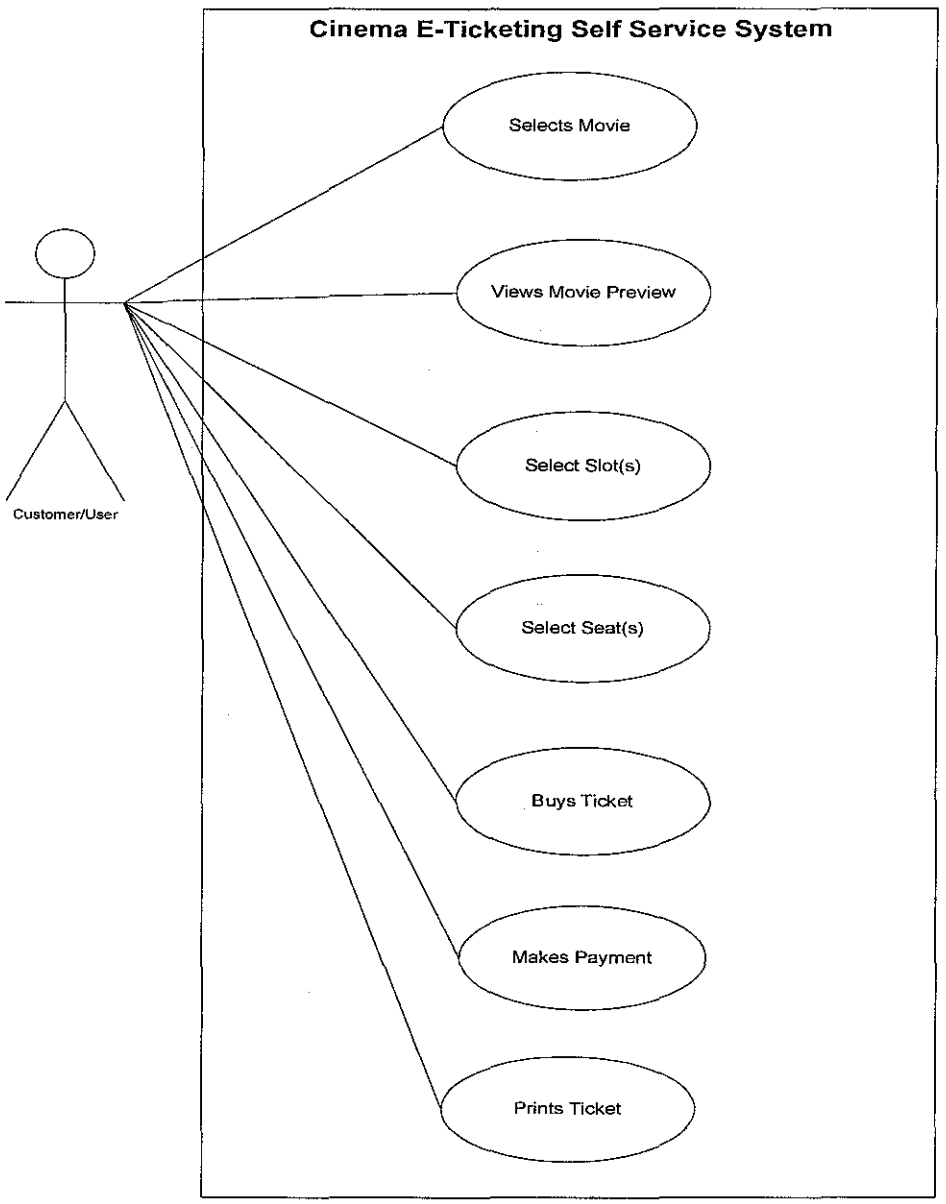


Figure 3.2: System Use Case Diagram

3.3.3. Design

Design focuses on high level design, in terms of what programs are needed and how are they going to interact, low-level design (how the individual programs are going to work), interface design (what are the interfaces going to look like) and data design (what data will be required). Much care is taken during this phase. The logical system of the product is developed in this phase. During the design of the system, it was realized that the system's interface needed to be a user-centered interface. This means approaching the system's interface design from the user perspective.

This project s interface was designed with Visual Basic.Net and Adobe Photoshop in mind. Adobe Photoshop CS2 assisted in the system s appearance because it has been realized that VB.Net does not contain a dynamic look of colors. The interface needed to attract the customer (user) visually. It had to go easy on the eyes in order to strike interest meaning it had to at least appear professional and make an impression on the user. One of the most important aspects of a kiosk/self-service interface standards is that, a kiosk interface had to have large buttons. An example of such an interface is depicted in **Figure: 3.3** below.

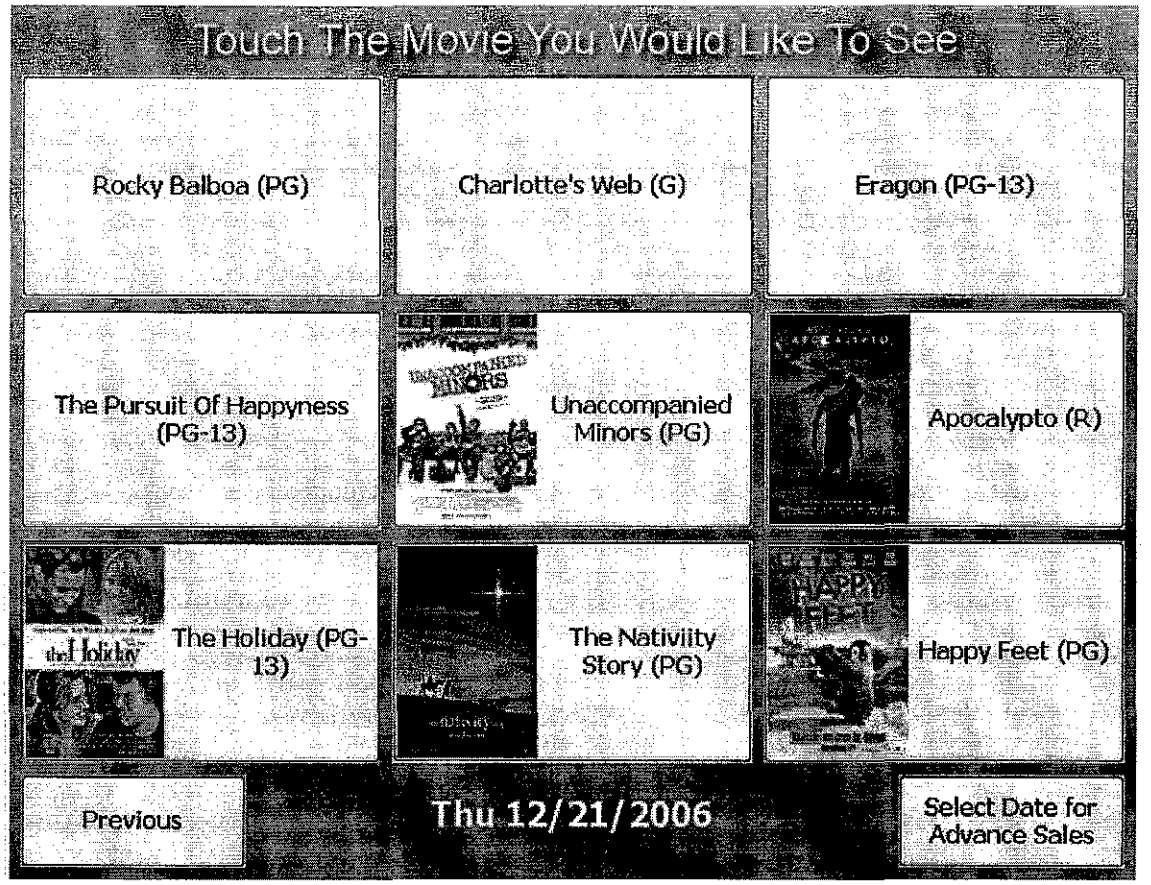


Figure 3.3: Example of a Movie Kiosk Interface

The above interface is actually on a touch screen. The easy-to-use interface allows customers to easily find the movie and show time that they want to see, in addition to seeing movie posters for the current titles. Due to the unavailability of a touch screen for the Cinema E-ticketing project, the normal mouse-click interaction will be used.

3.3.4. Development / Coding

The development phase involved gradually refining the design specifications into programming code. The programming of the Cinema e-ticketing self service system was undertaken utilizing the programming language called Microsoft Visual Basic.Net. Programming tools like Compilers, Interpreters, and Debuggers can also be used to

generate code. Though there is a vast array of suitable programming languages for development, Visual Basic.Net was the most suitable due to its simultaneous support of both design and code.

During the project's development the author experienced a few setbacks, though it was not complex, like any other system it had errors and a lot of reiteration occurred in order to get it to the best quality.

3.3.5. Payment Process Simulation

The author created a simulation of a payment system, for the ticket payment. The system requests the user's bank card number with the option of either a credit or debit card. The entering of the card number should follow a card swipe. The entering of the card number is executed by utilizing the onscreen numerical buttons, which simulate a touch screen on-screen numerical keyboard. Which in a real based system, the user would use a touch screen instead of mouse-click interaction. This simulation has three payment options, the credit card, debit card and cash respectively. Each one when chosen displays a screen relevant to that specific choice. When the user chooses the debit card option, the card expiration date is not requested where else for the credit card option, a card expiration date is requested.

The cash option only displays the seats selected and the numbers of seats selected are shown. The simulation assumes to automatically calculate the number of seats selected to the price of the movie chosen which the total will appear on the Ticket. Succeeding the submission from the Ticket Purchase page, the user can either choose to print the ticket or return to the main page if he/she has decided not purchase the particular ticket anymore.

3.3.6. Testing

During the testing phase, the system was tested for successful execution. The program was written as a series of individual forms, these were subjected to separate debugging. Thereafter the system was tested as a whole, a number of errors were identified therefore modifications had to be made to the code.

The separate forms were later on brought together and tested as a complete system. The system was to be tested to ensure that interfaces between forms work i.e. integration testing, the system works on the intended interface/platform and will do what the user requires.

3.3.7. Maintenance/Error Checking

Occasionally, the system needed maintenance. At times during testing the system would give error alerts which called for error-checking. Sometimes the system requirements changed, this change could happen because of some unexpected values that the user inputs into the system.

3.4. TOOLS IDENTIFIED

3.4.1. Software

- **Microsoft Visual Basic.Net**

VB.Net was used to develop the backend i.e. the Coding and Design. It is an object-oriented computer language that can be viewed as an evolution of Microsoft's Visual Basic (VB) implemented on the Microsoft .NET framework.

- **Adobe Photoshop CS2**

Photoshop was used to manipulate the colors used for the interface. Adobe Photoshop, or simply Photoshop, is a graphics editor developed and published by Adobe Systems. It is the current market leader for commercial bitmap and image manipulation, and is the flagship product of Adobe Systems. Although originally designed to edit images for paper-based printing, Photoshop can also be used for a wide range of other professional and amateur purposes.

3.4.2. Hardware

- Acer TravelMate 4001LCi Laptop,
  Intel Pentium ® M 705 processor
  (1.5 GHz, 400MHz FSB, 1MB L2 cache)
- Mouse
- Hp psc 1310 all-in-one. Printer, scanner, copier.
  - The printer was used as the ticket printer.

3.5. SYSTEM FLOWCHART

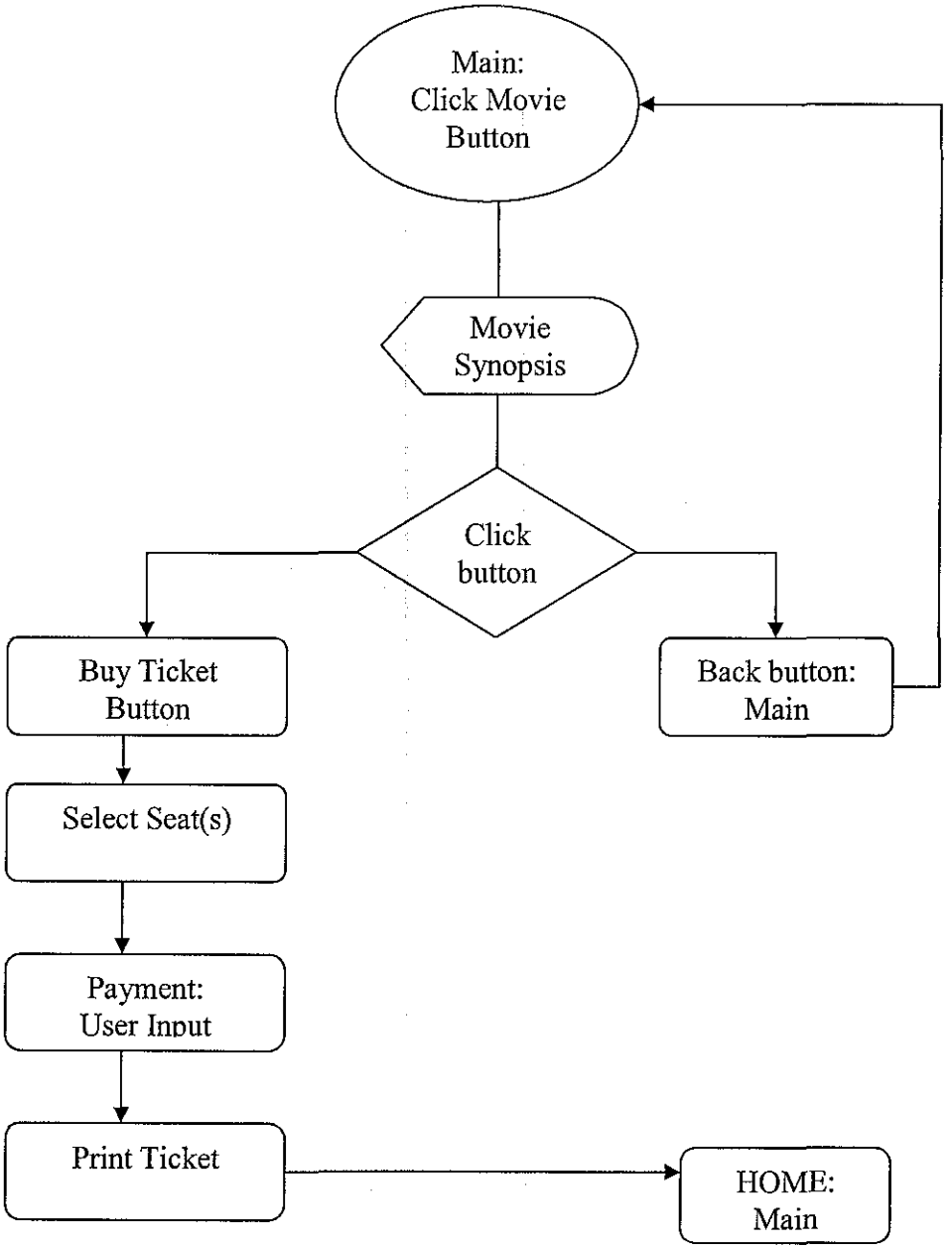


Figure 3.4: Cinema E-ticketing Self-Service System flowchart

CHAPTER 4

4.1. RESULTS

There are self-service systems in libraries, in public employment services, in stores and some are being developed for airport luggage check-in. These systems are all being utilized for the convenience of the customer which in corporate terms would improve customer relationship management resulting in customer retention. A system flow diagram was drafted and is clearly shown in Figure 3.3 above, the flow shows in detail the flow of the system, from a user s perspective.

The system was developed using Microsoft Visual Basic.Net; the system had to be designed using VB.Net s Design form to begin design process. Buttons had to be placed on the main page so as to achieve the "large" buttons requirements. The buttons were given equal sizes in order to be consistent. The technicalities of the system were not a complex as expected. The coding followed the design part. Code had to be searched on the internet, especially towards the end of the development process.

This project consists of 5 pages. Images and text files were first placed into a data folder, in the Project folder. The contents of the data folder were copied into another data folder within a bin folder. This was to ensure that the values can be called by the code during runtime.

For the respective forms and their corresponding code please refer to **Appendix 1-1.**

When developing a self service kiosk system, one should avoid the mistake of thinking they can take an existing website and port it to a kiosk. Web applications are designed to be accessed via a mouse, and usually are poorly suited to the kiosk interface of choice, a touch screen. Unfortunately due to being unable to acquire a touch screen the author developed the system having all the aspects of a touch screen in mind but utilizing a mouse.

The Cinema E-ticketing system has an interface of a kiosk system which is different to a website s interface, which was the aim in the initially. Hence, although the code behind the scenes still might be delivering the same information, the front end of a kiosk system needs to look very different from a website. A self service kiosk interface must have larger buttons and fewer details on one screen as compared to the usually smaller buttons and elaborate text on a website. Refer to Appendix to view the System s interface which meets the desired criteria.

The design of the system sticks to the old adage-turned-cliché that rings true: Less is more. The user interface prototyping process shown in **Figure 4.2** helped in approaching this project s interface prototyping phase. Security is also a vital component of any system therefore creating trust and peace of mind in the user is equally important. The Cinema E-ticketing Self service system s payment process is only a simulation; it does not execute a thorough payment process.

4.2. DISCUSSIONS

Throughout the researching and developing of this system, the author came across a number of different self-service systems being developed in the world for example, self-service airport check-ins, in-prison kiosks in the United States, and others. Self service systems are globally demanded especially by corporations trying to lower costs therefore the development of the Cinema E-ticketing Self-service system is one step towards creating electronic business solutions.

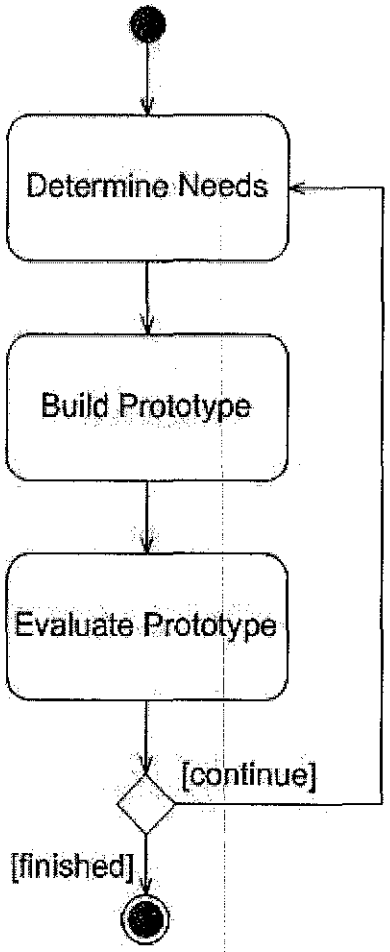


Figure 4.1: User Interface prototyping process

The author first determined the needs stage which is similar to the requirements gathering phase, the needs determine the approach to building the prototype since these are the processes. Thereafter the evaluation began right after the completion of the prototype.

CHAPTER 5

5.1. CONCLUSION

The objectives that have been set out clearly in the initial stages of the system indicated that the system should be more user-centered and beneficial to them in both usability and accessibility i.e. they should take the needs of the user into consideration. The development of the system has seen the author develop, modify and test the system in order for it to reach the desired standards. Simplicity was emphasized on as users wouldn't prefer a system that's complicated to use. Self service providers should further keep in mind that kiosks featuring a touch screen instead of a keyboard or mouse are also inherently more reliable, with fewer failure mechanisms and less frequent maintenance or replacement intervals.

The Cinema E-ticketing self-service system created had simplicity for the easy use and satisfaction of the user. With non-complex self-service system the user will find it easier to access the system and find tickets easily available. The efficiency of the system will allow an increase in passing-through speed. The completion of the system has brought great satisfaction to the author in terms of the system functions.

5.2. RECOMMENDATIONS

- There is a need for self-service systems to be developed, to assist staff or workers i.e. to reduce the load of work.
- Serious consideration must be taken into security issues and providing consumers with some level of security should be a priority.
- Steps should be taken to encourage better physical design of self-service systems. Instead of designs that repel customers and complex interfaces.
- Since self service is growing rapidly, everyone should be catered for including the handicapped. Disabled users especially the blind should be considered and steps should be taken in future to adapt self-service systems for their use.

REFERENCES

1. James Bicker (2006). Intelligent Design. *Self-service World and Kiosk Marketplace Guide,* 11, 18, 25-26.

2. Cecilia Crosby (2006), contributing editor. Printers. How to choose a printer for your self service device. *Self-service World and Kiosk Marketplace Guide*, 4, 9.

3. Sabry Tahir, (July 17 2007).MAS invests in electronic ticketing. *The Star Online.* Retrieved July 21, 2007, from http://biz.thestar.com.my/news/story.asp?file=/2007/7/17/business/18323113&sec=business

4. James Bickers and Frank Richmond (05 September 2006). Ticketing kiosks changing the cinema experience. Retrieved July 27, 2007, from http://www.kioskmarketplace.com/article.php?id=16124

5. Jungki Lee and Arthur Allaway (2002), "Effects of Personal Control on Adoption of Self-Service Technology Innovations," *Journal of Services Marketing*, 16 (6) and 553-572.

6. Ton˜ ita Perea y Monsuwe´, Benedict G.C. Dellaert and Ko de Ruyter (2004), What drives consumers to shop online? *A literature review, Journal: International Journal of Service Industry Management,* Maastricht University, Maastricht, The Netherlands.

7. Kerry Bodine (30 Mar 2007), Designing self-service around customer persona Retrieved August 21, 2007.

8. M.C. Maguire, A *Review of User-Interface Design Guidelines for Public Information Kiosk Systems*, HUSAT Research Institute.

9. *Self service cash register system.* Retrieved on May 24,2007,from http://www.synergon.hu/en/solutions_services/retail/ss_cash_registers.php

10. *Malaysia Air in e-ticketing push.* (April 20 2007) .Retrieved 23 April, 2007, from http://www.breakingtravelnews.com/article/20070420090101424

11. Last accessed September 22, 2007.*www.sterkinekor.co.za*

12. Last accessed September 18, 2007. *www.computicket.co.za*

13. *Kiosk.* Retrieved June 12, 2007 from http://www.pfasoft.com/kiosk.htm

14. *The Waterfall Model.* Retrieved September 15, 2007, from http://scitec.uwichill.edu.bb/cmp/online/cs22l/waterfall_model.htm

15. Kiosk and Customer Self Service. Retrieved September 15, 2007, from http://solutions.3m.com/wps/portal/3M/en_GB/TouchScreens/Home/PIA/Kiosks/

Figure 1: Main Page / Home

Large buttons were placed on the screen for visibility, required by interface standards. Each button takes you to the Movie Preview of the respective movie showed on that button. Upon selecting and clicking on that button, it will direct you to the Movie Preview Page.

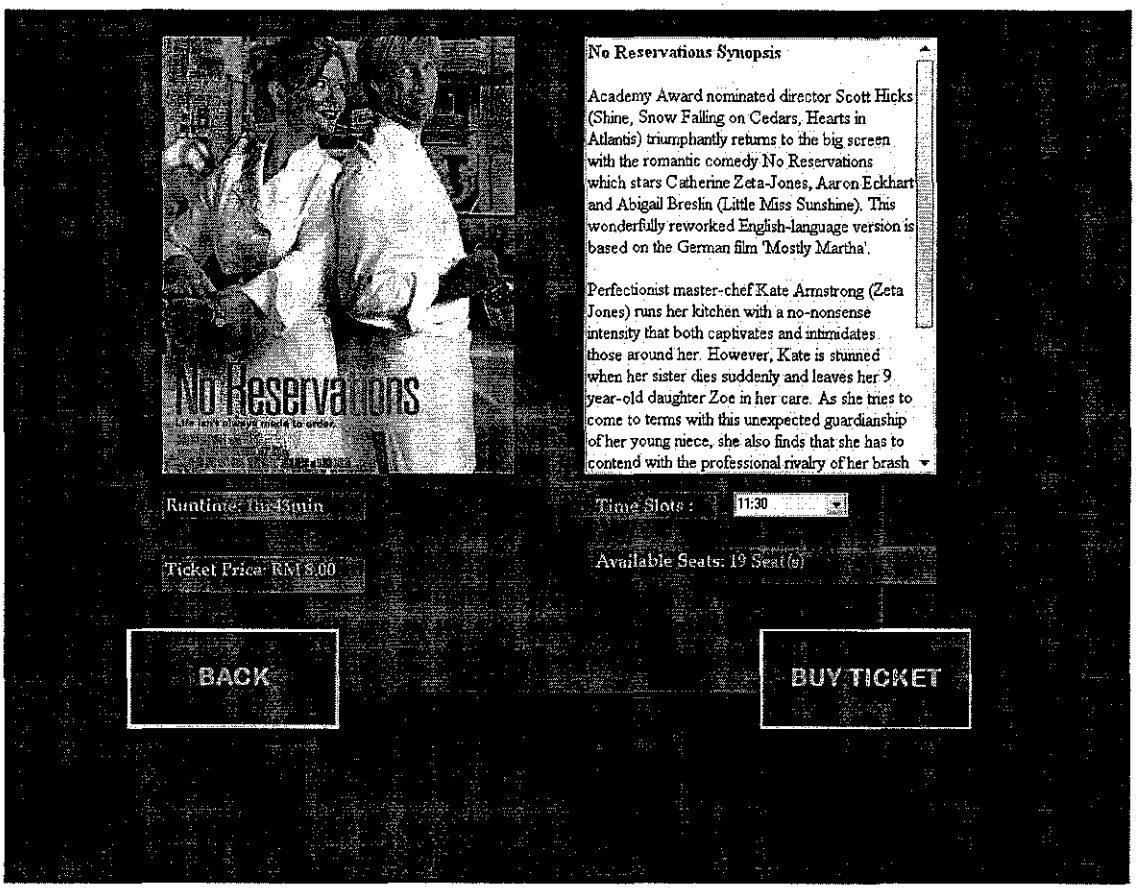


Figure 2: Movie Previews Page – No Reservations

Once the Movies Preview page appears, it displays the movie picture and its synopsis; the user can read through the synopsis before making the decision to buy the ticket or go back to the Main Page using buttons: BACK and BUY TICKET. This page also displays the Runtime, Ticket price, Time slots and Available seats for that movie. The user can select a slot.

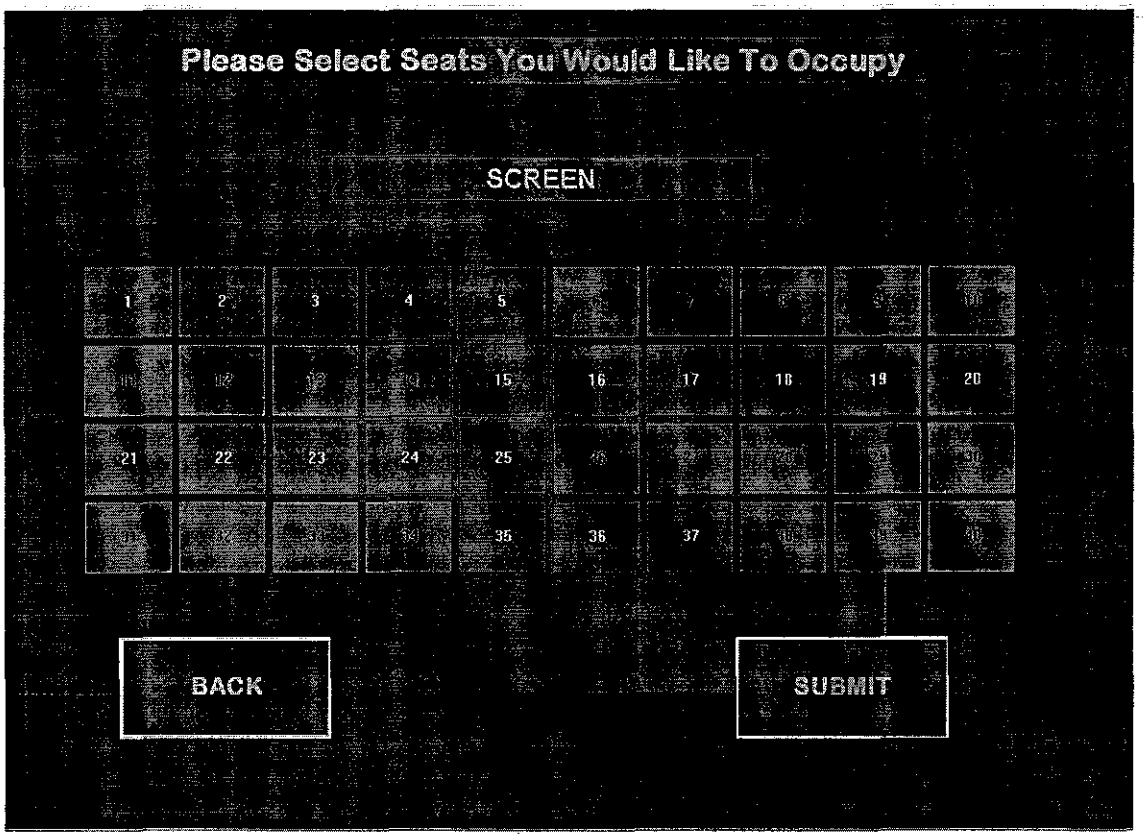


Figure 3: Seats Page

This page is displayed after the user has selected a slot and clicked on the Buy Ticket button. It displays the seats schema, the seats are numbered, the red are the unavailable or occupied seats and the green are the available seats. The user can only click on the green to select a seat. Then he/she can proceed using the Submit button.

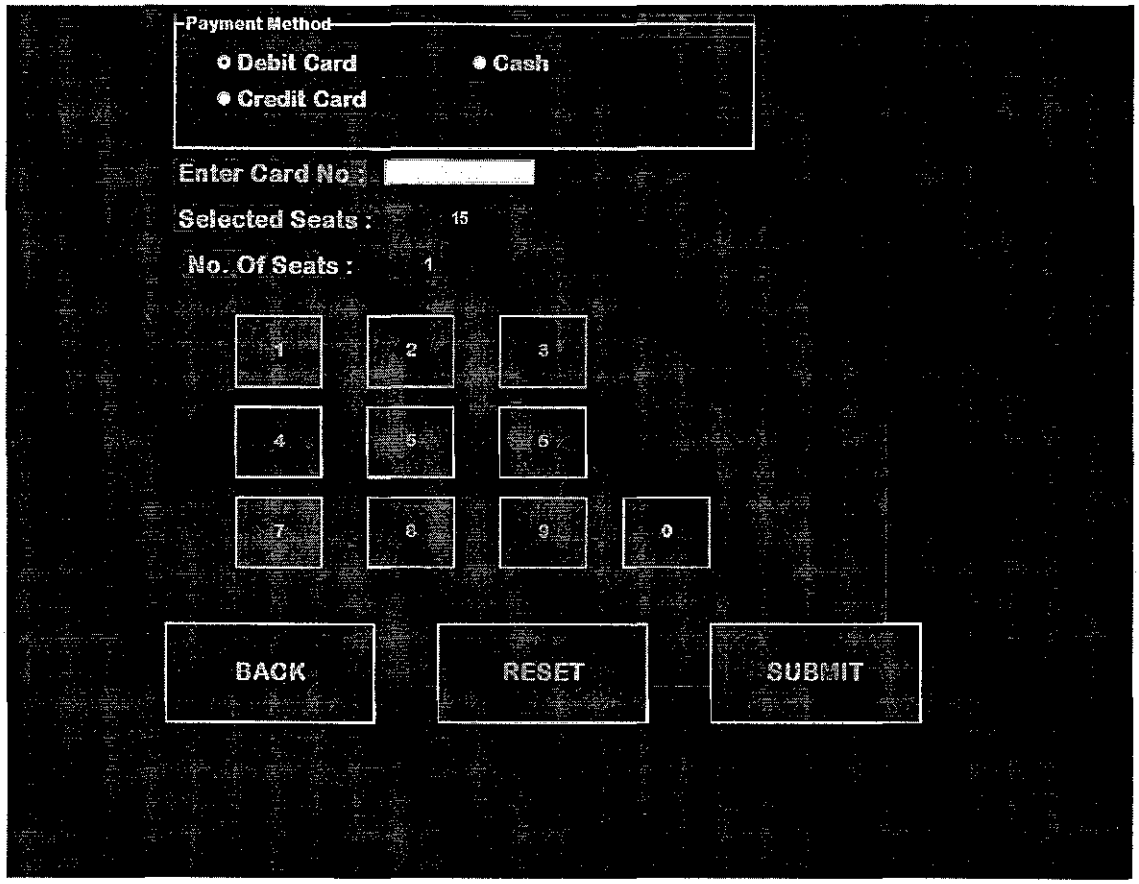


Figure 4: Ticket Purchase Page – Debit Card

The Ticket Purchase page will display a number of fields that need to be filled by the user. The user's payment method option, card number (this should be done using the numeric buttons on the screen), quantity or number of tickets the user would like to purchase. This page also displays the selected number of seats and the exact seat numbers. Upon completion the user can either click the SUBMIT button or the RESET button if changes need to be made and the Back button, to return to the seats page. When the Debit Card radio button is selected, the screen does not display a card expiry calendar option.

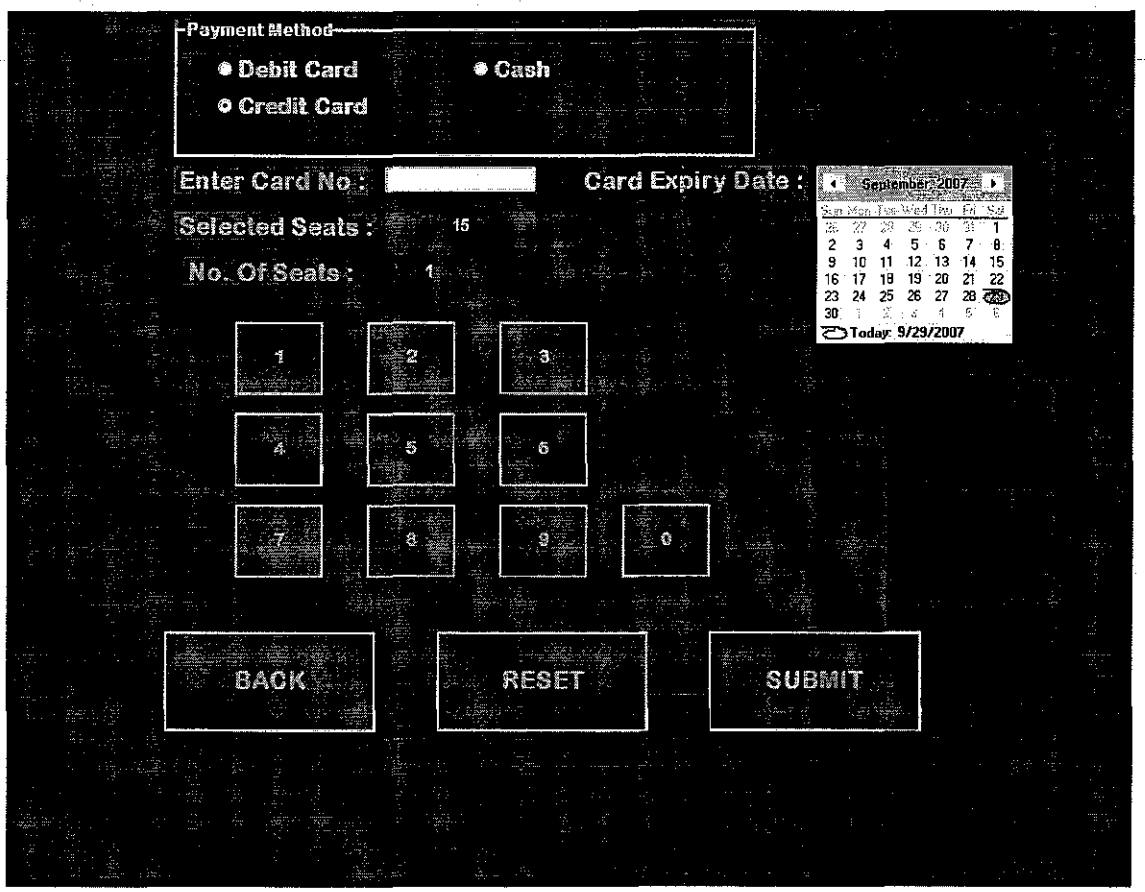


Figure 5: Ticket Purchase Page - Credit Card

The Ticket Purchase page will display a number of fields that need to be filled by the user. The user's payment method option, card number (this should be done using the numeric buttons on the screen), quantity or number of tickets the user would like to purchase. This page also displays the selected number of seats and the exact seat numbers. Upon completion the user can either click the SUBMIT button or the RESET button if changes need to be made and the Back button, to return to the seats page. When the Credit Card radio button is selected, the screen displays a card expiry calendar option.

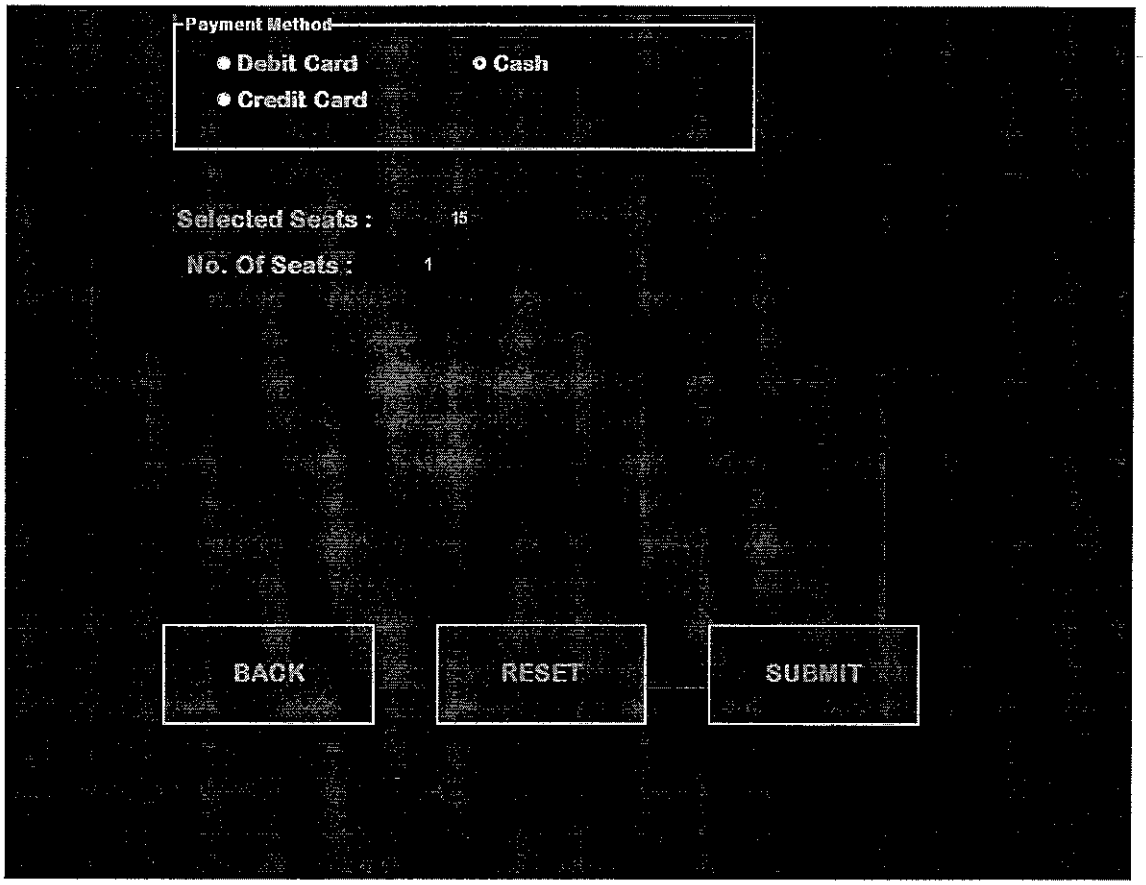


Figure 6: Ticket Purchase Page - Cash

The Ticket Purchase page will display a number of fields that need to be filled by the user. The user's payment method option, card number (this should be done using the numeric buttons on the screen), quantity or number of tickets the user would like to purchase. This page also displays the selected number of seats and the exact seat numbers. Upon completion the user can either click the SUBMIT button or the RESET button if changes need to be made and the Back button, to return to the seats page. When the Cash radio button is selected, the screen does not display a card expiry calendar option, the card number input box, the numeric buttons.

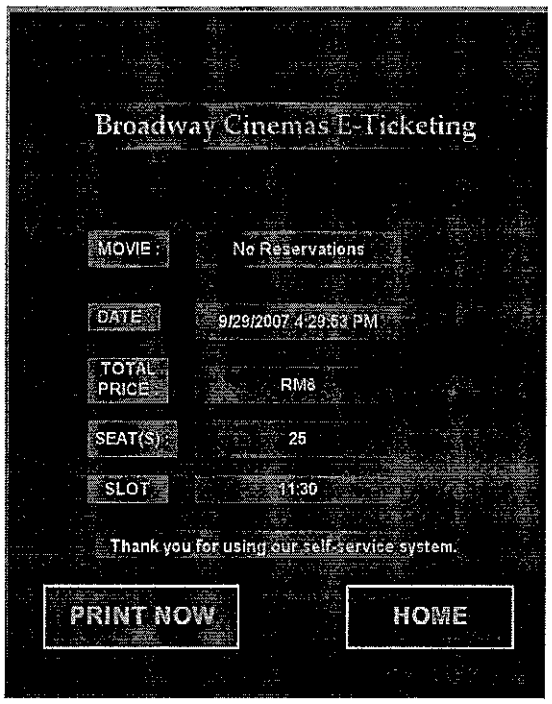


Figure 7: Ticket Print Page

When the SUBMIT is clicked, the Ticket Print Page will be displayed with the user's entered and selected details except for the pin number and payment method option because the payment process is only a simulation therefore the card details cannot be processed. This page also serves as the ticket to be printed, the user can either click the PRINT NOW button which prints out a document that represents a Ticket or the HOME button which closes the page and goes back to the Main page.

The tangible result of this user experience is the ticket printed which is the core of this project s objective. Part of the system was to print out a ticket or at least a representation of what the ticket should look like and be. When the ticket is printed, the buttons are not visible on the printed ticket.

PROJECT SOURCE CODE

MAIN PAGE

```
Imports System.Configuration
Public Class frmMain
  Inherits System.Windows.Forms.Form
    Public myMoviePreview As New frmMoviePreview(Me)
    Public myTicketPurchase As Object
    Public mySeats As Object
    Public myTicketPrint As Object
    Public movieId As Integer
    Public tickets(6) As Integer
    Public nMovie As Integer
    Public nTickets As Integer
    Public slotId As Integer
    Public selectedSeats As String
    Public Const NUMSEATS As Integer = 40
    'MAXIMUM: 6 theaters, 6 slots, and 40 seats
    Public seats(6, 6, NUMSEATS) As Boolean 'true = available, false = not available


#Region " Windows Form Designer generated code "

  Public Sub New()
    MyBase.New()

    'This call is required by the Windows Form Designer.
    InitializeComponent()

    'Add any initialization after the InitializeComponent() call

    End Sub
    'Form overrides dispose to clean up the component list.
    Protected Overloads Overrides Sub Dispose(ByVal disposing As Boolean)
        If disposing Then
            If Not (components Is Nothing) Then
                components.Dispose()
            End If
        End If
        MyBase.Dispose(disposing)
    End Sub

    'Required by the Windows Form Designer
    Private components As System.ComponentModel.IContainer

    'NOTE: The following procedure is required by the Windows Form Designer
    'It can be modified using the Windows Form Designer.
    'Do not modify it using the code editor.
    Friend WithEvents btnMovie1 As System.Windows.Forms.Button
    Friend WithEvents btnMovie2 As System.Windows.Forms.Button
    Friend WithEvents btnMovie3 As System.Windows.Forms.Button
    Friend WithEvents btnMovie4 As System.Windows.Forms.Button
    Friend WithEvents btnMovie5 As System.Windows.Forms.Button
    Friend WithEvents btnMovie6 As System.Windows.Forms.Button
    Friend WithEvents lblTitle As System.Windows.Forms.Label
    <System.Diagnostics.DebuggerStepThrough()> Private Sub InitializeComponent()
        Dim resources As System.Resources.ResourceManager = New
System.Resources.ResourceManager(GetType(frmMain))
        Me.btnMovie1 = New System.Windows.Forms.Button
        Me.btnMovie2 = New System.Windows.Forms.Button
        Me.btnMovie3 = New System.Windows.Forms.Button
        Me.btnMovie4 = New System.Windows.Forms.Button
        Me.btnMovie5 = New System.Windows.Forms.Button
        Me.btnMovie6 = New System.Windows.Forms.Button
        Me.lblTitle = New System.Windows.Forms.Label
        Me.SuspendLayout()
        '
        'btnMovie1
```

```
        '
        Me.btnMovie1.AccessibleRole = System.Windows.Forms.AccessibleRole.Sound
        Me.btnMovie1.Cursor = System.Windows.Forms.Cursors.Arrow
        Me.btnMovie1.Image = CType(resources.GetObject("btnMovie1.Image"),
System.Drawing.Image)
        Me.btnMovie1.Location = New System.Drawing.Point(32, 48)
        Me.btnMovie1.Name = "btnMovie1"
        Me.btnMovie1.Size = New System.Drawing.Size(296, 280)
        Me.btnMovie1.TabIndex = 1
        '
        'btnMovie2
        '
        Me.btnMovie2.Image = CType(resources.GetObject("btnMovie2.Image"),
System.Drawing.Image)
        Me.btnMovie2.Location = New System.Drawing.Point(352, 48)
        Me.btnMovie2.Name = "btnMovie2"
        Me.btnMovie2.Size = New System.Drawing.Size(296, 280)
        Me.btnMovie2.TabIndex = 2
        '
        'btnMovie3
        '
        Me.btnMovie3.Image = CType(resources.GetObject("btnMovie3.Image"),
System.Drawing.Image)
        Me.btnMovie3.Location = New System.Drawing.Point(680, 48)
        Me.btnMovie3.Name = "btnMovie3"
        Me.btnMovie3.Size = New System.Drawing.Size(296, 280)
        Me.btnMovie3.TabIndex = 3
        '
        'btnMovie4
        '
        Me.btnMovie4.Image = CType(resources.GetObject("btnMovie4.Image"),
System.Drawing.Image)
        Me.btnMovie4.Location = New System.Drawing.Point(40, 400)
        Me.btnMovie4.Name = "btnMovie4"
        Me.btnMovie4.Size = New System.Drawing.Size(296, 280)
        Me.btnMovie4.TabIndex = 4
        '
        'btnMovie5
        '
        Me.btnMovie5.Image = CType(resources.GetObject("btnMovie5.Image"),
System.Drawing.Image)
        Me.btnMovie5.Location = New System.Drawing.Point(352, 400)
        Me.btnMovie5.Name = "btnMovie5"
        Me.btnMovie5.Size = New System.Drawing.Size(296, 280)
        Me.btnMovie5.TabIndex = 5
        '
        'btnMovie6
        '
        Me.btnMovie6.BackgroundImage =
CType(resources.GetObject("btnMovie6.BackgroundImage"), System.Drawing.Image)
        Me.btnMovie6.Image = CType(resources.GetObject("btnMovie6.Image"),
System.Drawing.Image)
        Me.btnMovie6.Location = New System.Drawing.Point(688, 400)
        Me.btnMovie6.Name = "btnMovie6"
        Me.btnMovie6.Size = New System.Drawing.Size(296, 280)
        Me.btnMovie6.TabIndex = 6
        '
        'lblTitle
        '
        Me.lblTitle.Font = New System.Drawing.Font("Arial Rounded MT Bold", 18.0!,
System.Drawing.FontStyle.Bold, System.Drawing.GraphicsUnit.Point, CType(0, Byte))
        Me.lblTitle.ForeColor = System.Drawing.Color.GreenYellow
        Me.lblTitle.Image = CType(resources.GetObject("lblTitle.Image"),
System.Drawing.Image)
        Me.lblTitle.Location = New System.Drawing.Point(120, 336)
        Me.lblTitle.Name = "lblTitle"
        Me.lblTitle.Size = New System.Drawing.Size(664, 64)
        Me.lblTitle.TabIndex = 7
        Me.lblTitle.Text = "Please Click Button to Select Movie"
        Me.lblTitle.TextAlign = System.Drawing.ContentAlignment.MiddleCenter
        '
```

```
        'frmMain
        '
        Me.AutoScaleBaseSize = New System.Drawing.Size(5, 13)
        Me.BackgroundImage = CType(resources.GetObject("$this.BackgroundImage"),
System.Drawing.Image)
        Me.ClientSize = New System.Drawing.Size(1002, 698)
        Me.Controls.Add(Me.btnMovie2)
        Me.Controls.Add(Me.lblTitle)
        Me.Controls.Add(Me.btnMovie6)
        Me.Controls.Add(Me.btnMovie5)
        Me.Controls.Add(Me.btnMovie4)
        Me.Controls.Add(Me.btnMovie3)
        Me.Controls.Add(Me.btnMovie1)
        Me.Font = New System.Drawing.Font("Microsoft Sans Serif", 8.25!,
System.Drawing.FontStyle.Regular, System.Drawing.GraphicsUnit.Point, CType(0, Byte))
        Me.FormBorderStyle = System.Windows.Forms.FormBorderStyle.None
        Me.MaximizeBox = False
        Me.Name = "frmMain"
        Me.StartPosition = System.Windows.Forms.FormStartPosition.CenterScreen
        Me.Text = "Cinema Eticketing Self Service - Home"
        Me.WindowState = System.Windows.Forms.FormWindowState.Maximized
        Me.ResumeLayout(False)

    End Sub

#End Region

  Private Sub frmMain_Load(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles MyBase.Load
        nMovie = 6
        Dim i, j, k As Integer
        Dim tmpString As String() = Nothing
        Dim tmpS As String
        Dim valS As Integer
        For i = 0 To nMovie
            'initialize all the seats to false first
            For j = 0 To 6
                For k = 0 To NUMSEATS - 1
                    seats(i, j, k) = False
                Next
            Next
            'initialize all the available seats
            For j = 1 To CInt(ConfigurationSettings.AppSettings("Movie" + CStr(i + 1) +
"Slots"))
                tmpString = ConfigurationSettings.AppSettings("Movie" + CStr(i + 1) +
"Slot" + CStr(j) + "Seats").Split(",")
                'Console.Write("Movie" + CStr(i + 1) + "Slot" + CStr(j) + "Seats: ")
                'Console.WriteLine(tmpString)
                For Each tmpS In tmpString
                    valS = CInt(tmpS)
                    seats(i, j - 1, valS - 1) = True
                Next
            Next
        Next
  End Sub

  Private Sub btnMovie1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles btnMovie1.Click
    Me.Hide()
        movieId = 1
        slotId = 1
    prepareMoviePreview()
    myMoviePreview.Show()
  End Sub

  Private Sub btnMovie2_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles btnMovie2.Click
    Me.Hide()
        movieId = 2
        slotId = 2
    prepareMoviePreview()
```

```
    myMoviePreview.Show()
  End Sub

  Private Sub prepareMoviePreview()
    myMoviePreview.picBox.Image =
Image.FromFile(System.Windows.Forms.Application.StartupPath + "\data\Movie" +
CStr(movieId) + "Pic.jpg")
    myMoviePreview.txtMoviePreview.LoadFile(System.Windows.Forms.Application.StartupPath
+ "\data\Movie" + CStr(movieId) + "Preview.rtf")
    myMoviePreview.lblPrice.Text = "Ticket Price: RM " +
ConfigurationSettings.AppSettings("Movie" + CStr(movieId) + "Price")
        myMoviePreview.lblRuntime.Text = "Runtime: " +
ConfigurationSettings.AppSettings("Movie" + CStr(movieId) + "Runtime")
        myMoviePreview.cbTimesSlot.Items.Clear()
        Dim i As Integer
        For i = 1 To CInt(ConfigurationSettings.AppSettings("Movie" + CStr(movieId) +
"Slots"))

myMoviePreview.cbTimesSlot.Items.Add(ConfigurationSettings.AppSettings("Movie" +
CStr(movieId) + "Slot" + CStr(i) + "Time"))
        Next
        myMoviePreview.cbTimesSlot.SelectedIndex = 0
        myMoviePreview.lblSeats.Text = "Available Seats: " + CStr(countSeats(seats,
movieId - 1, 0)) + " Seat(s)"
  End Sub

  Private Sub frmMain_Closed(ByVal sender As Object, ByVal e As System.EventArgs) Handles
MyBase.Closed
        Dim tmpName As String
        Dim a As New clsAppSettings(clsAppSettings.Config.DebugFile)
        Dim i As Integer
        Dim j As Integer
        Dim k As Integer
        Dim tmpString As String
        For i = 1 To nMovie
            '--- update the database ---
            For j = 0 To 6
                tmpString = ""
                For k = 0 To NUMSEATS - 1
                    If seats(i - 1, j, k) Then
                        If tmpString.Length > 0 Then
                            tmpString = tmpString + ", " + CStr((k + 1))
                        Else
                            tmpString = CStr((k + 1))
                        End If
                    End If
                Next
                tmpName = "Movie" + CStr(i) + "Slot" + CStr((j + 1)) + "Seats"
                a.SaveSetting(tmpName, tmpString)
            Next
            '--- end of update the database ---
        Next
  End Sub

    Private Sub btnMovie3_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnMovie3.Click
        Me.Hide()
        movieId = 3
        slotId = 3
        prepareMoviePreview()
        myMoviePreview.Show()
    End Sub

    Private Sub btnMovie4_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnMovie4.Click
        Me.Hide()
        movieId = 4
        slotId = 4
        prepareMoviePreview()
        myMoviePreview.Show()
    End Sub
```

```
    Private Sub btnMovie5_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnMovie5.Click
        Me.Hide()
        movieId = 5
        slotId = 5
        prepareMoviePreview()
        myMoviePreview.Show()
    End Sub

    Private Sub btnMovie6_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnMovie6.Click
        Me.Hide()
        movieId = 6
        slotId = 6
        prepareMoviePreview()
        myMoviePreview.Show()
    End Sub

    Public Function countSeats(ByVal tmpSeats(,,) As Boolean, ByVal movieId As Integer,
ByVal slotId As Integer) As Integer
        Dim i As Integer
        Dim cnt As Integer
        cnt = 0
        For i = 0 To NUMSEATS - 1
            If tmpSeats(movieId, slotId, i) = True Then
                cnt = cnt + 1
            End If
        Next
        Return cnt
    End Function
End Class
```

MOVIE PREVIEW PAGE

```
Imports System.Configuration

Public Class frmMoviePreview
    Inherits System.Windows.Forms.Form
    Dim myMain As Object

#Region " Windows Form Designer generated code "

    Public Sub New()
        MyBase.New()

        'This call is required by the Windows Form Designer.
        InitializeComponent()

        'Add any initialization after the InitializeComponent() call

    End Sub

    Public Sub New(ByVal caller As Object)
        MyBase.New()

        'This call is required by the Windows Form Designer.
        InitializeComponent()

        'Add any initialization after the InitializeComponent() call
        myMain = caller
    End Sub

    'Form overrides dispose to clean up the component list.
    Protected Overloads Overrides Sub Dispose(ByVal disposing As Boolean)
        If disposing Then
            If Not (components Is Nothing) Then
                components.Dispose()
```

```
            End If
        End If
        MyBase.Dispose(disposing)
    End Sub

    'Required by the Windows Form Designer
    Private components As System.ComponentModel.IContainer

    'NOTE: The following procedure is required by the Windows Form Designer
    'It can be modified using the Windows Form Designer.
    'Do not modify it using the code editor.
    Friend WithEvents picBox As System.Windows.Forms.PictureBox
    Friend WithEvents txtMoviePreview As System.Windows.Forms.RichTextBox
    Friend WithEvents btnBack As System.Windows.Forms.Button
    Friend WithEvents lblPrice As System.Windows.Forms.Label
    Friend WithEvents lblSeats As System.Windows.Forms.Label
    Friend WithEvents btnBuy As System.Windows.Forms.Button
    Friend WithEvents lblRuntime As System.Windows.Forms.Label
    Friend WithEvents lblTimeslots As System.Windows.Forms.Label
    Friend WithEvents cbTimesSlot As System.Windows.Forms.ComboBox
    <System.Diagnostics.DebuggerStepThrough()> Private Sub InitializeComponent()
        Dim resources As System.Resources.ResourceManager = New
System.Resources.ResourceManager(GetType(frmMoviePreview))
        Me.btnBack = New System.Windows.Forms.Button
        Me.picBox = New System.Windows.Forms.PictureBox
        Me.txtMoviePreview = New System.Windows.Forms.RichTextBox
        Me.lblPrice = New System.Windows.Forms.Label
        Me.lblSeats = New System.Windows.Forms.Label
        Me.btnBuy = New System.Windows.Forms.Button
        Me.lblRuntime = New System.Windows.Forms.Label
        Me.lblTimeslots = New System.Windows.Forms.Label
        Me.cbTimesSlot = New System.Windows.Forms.ComboBox
        Me.SuspendLayout()
        '
        'btnBack
        '
        Me.btnBack.BackgroundImage =
CType(resources.GetObject("btnBack.BackgroundImage"), System.Drawing.Image)
        Me.btnBack.Font = New System.Drawing.Font("Arial Rounded MT Bold", 15.75!,
System.Drawing.FontStyle.Bold, System.Drawing.GraphicsUnit.Point, CType(0, Byte))
        Me.btnBack.Location = New System.Drawing.Point(112, 544)
        Me.btnBack.Name = "btnBack"
        Me.btnBack.Size = New System.Drawing.Size(192, 88)
        Me.btnBack.TabIndex = 1
        Me.btnBack.Text = "BACK"
        '
        'picBox
        '
        Me.picBox.Image = CType(resources.GetObject("picBox.Image"),
System.Drawing.Image)
        Me.picBox.Location = New System.Drawing.Point(144, 24)
        Me.picBox.Name = "picBox"
        Me.picBox.Size = New System.Drawing.Size(320, 384)
        Me.picBox.SizeMode = System.Windows.Forms.PictureBoxSizeMode.CenterImage
        Me.picBox.TabIndex = 2
        Me.picBox.TabStop = False
        '
        'txtMoviePreview
        '
        Me.txtMoviePreview.Font = New System.Drawing.Font("Verdana", 8.25!,
System.Drawing.FontStyle.Regular, System.Drawing.GraphicsUnit.Point, CType(0, Byte))
        Me.txtMoviePreview.Location = New System.Drawing.Point(528, 24)
        Me.txtMoviePreview.Name = "txtMoviePreview"
        Me.txtMoviePreview.ReadOnly = True
        Me.txtMoviePreview.Size = New System.Drawing.Size(320, 384)
        Me.txtMoviePreview.TabIndex = 4
        Me.txtMoviePreview.Text = ""
        '
        'lblPrice
        '
```

```
        Me.lblPrice.Font = New System.Drawing.Font("Palatino Linotype", 12.0!,
System.Drawing.FontStyle.Bold, System.Drawing.GraphicsUnit.Point, CType(0, Byte))
        Me.lblPrice.Image = CType(resources.GetObject("lblPrice.Image"),
System.Drawing.Image)
        Me.lblPrice.Location = New System.Drawing.Point(144, 480)
        Me.lblPrice.Name = "lblPrice"
        Me.lblPrice.Size = New System.Drawing.Size(184, 32)
        Me.lblPrice.TabIndex = 5
        '
        'lblSeats
        '
        Me.lblSeats.Font = New System.Drawing.Font("Palatino Linotype", 12.0!,
System.Drawing.FontStyle.Bold, System.Drawing.GraphicsUnit.Point, CType(0, Byte))
        Me.lblSeats.Image = CType(resources.GetObject("lblSeats.Image"),
System.Drawing.Image)
        Me.lblSeats.Location = New System.Drawing.Point(536, 472)
        Me.lblSeats.Name = "lblSeats"
        Me.lblSeats.Size = New System.Drawing.Size(312, 32)
        Me.lblSeats.TabIndex = 6
        '
        'btnBuy
        '
        Me.btnBuy.BackgroundImage = CType(resources.GetObject("btnBuy.BackgroundImage"),
System.Drawing.Image)
        Me.btnBuy.Font = New System.Drawing.Font("Arial Rounded MT Bold", 15.75!,
System.Drawing.FontStyle.Bold, System.Drawing.GraphicsUnit.Point, CType(0, Byte))
        Me.btnBuy.ForeColor = System.Drawing.Color.GreenYellow
        Me.btnBuy.Location = New System.Drawing.Point(688, 544)
        Me.btnBuy.Name = "btnBuy"
        Me.btnBuy.Size = New System.Drawing.Size(192, 88)
        Me.btnBuy.TabIndex = 7
        Me.btnBuy.Text = "BUY TICKET"
        '
        'lblRuntime
        '
        Me.lblRuntime.Font = New System.Drawing.Font("Palatino Linotype", 12.0!,
System.Drawing.FontStyle.Bold, System.Drawing.GraphicsUnit.Point, CType(0, Byte))
        Me.lblRuntime.Image = CType(resources.GetObject("lblRuntime.Image"),
System.Drawing.Image)
        Me.lblRuntime.Location = New System.Drawing.Point(144, 424)
        Me.lblRuntime.Name = "lblRuntime"
        Me.lblRuntime.Size = New System.Drawing.Size(184, 24)
        Me.lblRuntime.TabIndex = 8
        '
        'lblTimeslots
        '
        Me.lblTimeslots.Font = New System.Drawing.Font("Palatino Linotype", 12.0!,
System.Drawing.FontStyle.Bold, System.Drawing.GraphicsUnit.Point, CType(0, Byte))
        Me.lblTimeslots.Image = CType(resources.GetObject("lblTimeslots.Image"),
System.Drawing.Image)
        Me.lblTimeslots.Location = New System.Drawing.Point(536, 424)
        Me.lblTimeslots.Name = "lblTimeslots"
        Me.lblTimeslots.Size = New System.Drawing.Size(112, 24)
        Me.lblTimeslots.TabIndex = 9
        Me.lblTimeslots.Text = "Time Slots :"
        '
        'cbTimesSlot
        '
        Me.cbTimesSlot.Location = New System.Drawing.Point(664, 424)
        Me.cbTimesSlot.Name = "cbTimesSlot"
        Me.cbTimesSlot.Size = New System.Drawing.Size(104, 21)
        Me.cbTimesSlot.TabIndex = 10
        '
        'frmMoviePreview
        '
        Me.AutoScaleBaseSize = New System.Drawing.Size(5, 13)
        Me.BackgroundImage = CType(resources.GetObject("$this.BackgroundImage"),
System.Drawing.Image)
        Me.ClientSize = New System.Drawing.Size(1000, 662)
        Me.Controls.Add(Me.cbTimesSlot)
        Me.Controls.Add(Me.lblTimeslots)
```

```
        Me.Controls.Add(Me.lblRuntime)
        Me.Controls.Add(Me.btnBuy)
        Me.Controls.Add(Me.lblSeats)
        Me.Controls.Add(Me.lblPrice)
        Me.Controls.Add(Me.txtMoviePreview)
        Me.Controls.Add(Me.picBox)
        Me.Controls.Add(Me.btnBack)
        Me.ForeColor = System.Drawing.Color.GreenYellow
        Me.FormBorderStyle = System.Windows.Forms.FormBorderStyle.None
        Me.MaximizeBox = False
        Me.Name = "frmMoviePreview"
        Me.StartPosition = System.Windows.Forms.FormStartPosition.CenterScreen
        Me.Text = "MoviePreview"
        Me.WindowState = System.Windows.Forms.FormWindowState.Maximized
        Me.ResumeLayout(False)

    End Sub

#End Region
    'Close movie preview page to open main form
    Private Sub frmMoviePreview_Closed(ByVal sender As Object, ByVal e As
System.EventArgs) Handles MyBase.Closed
        myMain.Show()
    End Sub
    ' To close the current page to show main form when back button is clicked
    Private Sub btnBack_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles btnBack.Click
        Me.Hide()
        myMain.Show()
    End Sub
    'Load movie preview form
    Private Sub frmMoviePreview_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
    End Sub
    ' Click on Buy button, will close the movie preview form and display the Seat
Selection form.
    Private Sub btnBuy_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles btnBuy.Click
        Me.Hide()
        myMain.slotId = cbTimesSlot.SelectedIndex
        myMain.mySeats = New frmSeats(myMain)
        myMain.mySeats.Show()
    End Sub

    Private Sub cbTimesSlot_SelectedIndexChanged(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles cbTimesSlot.SelectedIndexChanged
        lblSeats.Text = "Available Seats: " + CStr(myMain.countSeats(myMain.seats,
myMain.movieId - 1, cbTimesSlot.SelectedIndex)) + " Seat(s)"
    End Sub
End Class
```

SEAT SELECTION PAGE

```
Imports System.Configuration

Public Class frmSeats
    Inherits System.Windows.Forms.Form
    Dim myMain As Object

#Region " Windows Form Designer generated code "
    Dim btnSeats() As Button
    Public Sub New()
        MyBase.New()

        'This call is required by the Windows Form Designer.
        InitializeComponent()
        InitializeSeats()
        'Add any initialization after the InitializeComponent() call
```

```
    End Sub
    Public Sub New(ByVal caller As Object)
        MyBase.New()

        myMain = caller

        'This call is required by the Windows Form Designer.
        InitializeComponent()
        InitializeSeats()
        'Add any initialization after the InitializeComponent() call
    End Sub


    'Form overrides dispose to clean up the component list.
    Protected Overloads Overrides Sub Dispose(ByVal disposing As Boolean)
        If disposing Then
            If Not (components Is Nothing) Then
                components.Dispose()
            End If
        End If
        MyBase.Dispose(disposing)
    End Sub

    Private Sub InitializeSeats()
        Dim i, j, k, seatWidth, seatHeight, seatX, seatY, seatGapX, seatGapY As Integer
        ReDim btnSeats(40)

        j = 0
        k = 0
        seatWidth = 80
        seatHeight = 64
        seatX = 72
        seatY = 224
        seatGapX = 85
        seatGapY = 69

        For i = 0 To myMain.NUMSEATS - 1
            btnSeats(i) = New System.Windows.Forms.Button
            With btnSeats(i)
                If myMain.seats(myMain.movieId - 1, myMain.slotId, i) Then
                    .BackColor = System.Drawing.Color.Green
                Else
                    .BackColor = System.Drawing.Color.Red
                    .Enabled = False
                End If
                .FlatStyle = System.Windows.Forms.FlatStyle.Popup
                .Font = New System.Drawing.Font("Microsoft Sans Serif", 9.75!,
System.Drawing.FontStyle.Bold, System.Drawing.GraphicsUnit.Point, CType(0, Byte))
                .Text = (i + 1)
                .Size = New Size(seatWidth, seatHeight)
                .Location = New Point(seatX + (seatGapX * j), seatY + (seatGapY * k))
                .Tag = i
                AddHandler .Click, AddressOf Me.btnSeatsClick
            End With
            j = j + 1
            If (i + 1) Mod 10 = 0 Then
                j = 0
                k = k + 1
            End If
        Next
        Me.Controls.AddRange(btnSeats)
    End Sub
    'Required by the Windows Form Designer
    Private components As System.ComponentModel.IContainer

    'NOTE: The following procedure is required by the Windows Form Designer
    'It can be modified using the Windows Form Designer.
    'Do not modify it using the code editor.
    Friend WithEvents lblSreen As System.Windows.Forms.Label
    Friend WithEvents btnBack As System.Windows.Forms.Button
```

```
    Friend WithEvents btnSubmit As System.Windows.Forms.Button
    Friend WithEvents lblTitleSeats As System.Windows.Forms.Label
    <System.Diagnostics.DebuggerStepThrough()> Private Sub InitializeComponent()
        Dim resources As System.Resources.ResourceManager = New
System.Resources.ResourceManager(GetType(frmSeats))
        Me.lblSreen = New System.Windows.Forms.Label
        Me.btnBack = New System.Windows.Forms.Button
        Me.btnSubmit = New System.Windows.Forms.Button
        Me.lblTitleSeats = New System.Windows.Forms.Label
        Me.SuspendLayout()
        '
        'lblSreen
        '
        Me.lblSreen.BorderStyle = System.Windows.Forms.BorderStyle.FixedSingle
        Me.lblSreen.Font = New System.Drawing.Font("Arial", 18.0!,
System.Drawing.FontStyle.Bold, System.Drawing.GraphicsUnit.Point, CType(0, Byte))
        Me.lblSreen.Image = CType(resources.GetObject("lblSreen.Image"),
System.Drawing.Image)
        Me.lblSreen.Location = New System.Drawing.Point(296, 128)
        Me.lblSreen.Name = "lblSreen"
        Me.lblSreen.Size = New System.Drawing.Size(384, 40)
        Me.lblSreen.TabIndex = 0
        Me.lblSreen.Text = "SCREEN"
        Me.lblSreen.TextAlign = System.Drawing.ContentAlignment.MiddleCenter
        '
        'btnBack
        '
        Me.btnBack.BackgroundImage =
CType(resources.GetObject("btnBack.BackgroundImage"), System.Drawing.Image)
        Me.btnBack.Font = New System.Drawing.Font("Arial Rounded MT Bold", 15.75!,
System.Drawing.FontStyle.Bold, System.Drawing.GraphicsUnit.Point, CType(0, Byte))
        Me.btnBack.ForeColor = System.Drawing.Color.GreenYellow
        Me.btnBack.Location = New System.Drawing.Point(104, 552)
        Me.btnBack.Name = "btnBack"
        Me.btnBack.Size = New System.Drawing.Size(192, 88)
        Me.btnBack.TabIndex = 20
        Me.btnBack.Text = "BACK"
        '
        'btnSubmit
        '
        Me.btnSubmit.BackgroundImage =
CType(resources.GetObject("btnSubmit.BackgroundImage"), System.Drawing.Image)
        Me.btnSubmit.Font = New System.Drawing.Font("Arial Rounded MT Bold", 15.75!,
System.Drawing.FontStyle.Bold, System.Drawing.GraphicsUnit.Point, CType(0, Byte))
        Me.btnSubmit.ForeColor = System.Drawing.Color.GreenYellow
        Me.btnSubmit.Location = New System.Drawing.Point(664, 552)
        Me.btnSubmit.Name = "btnSubmit"
        Me.btnSubmit.Size = New System.Drawing.Size(192, 88)
        Me.btnSubmit.TabIndex = 21
        Me.btnSubmit.Text = "SUBMIT"
        '
        'lblTitleSeats
        '
        Me.lblTitleSeats.Font = New System.Drawing.Font("Arial Rounded MT Bold", 20.25!,
System.Drawing.FontStyle.Bold, System.Drawing.GraphicsUnit.Point, CType(0, Byte))
        Me.lblTitleSeats.ForeColor = System.Drawing.Color.GreenYellow
        Me.lblTitleSeats.Image = CType(resources.GetObject("lblTitleSeats.Image"),
System.Drawing.Image)
        Me.lblTitleSeats.Location = New System.Drawing.Point(120, 24)
        Me.lblTitleSeats.Name = "lblTitleSeats"
        Me.lblTitleSeats.Size = New System.Drawing.Size(736, 40)
        Me.lblTitleSeats.TabIndex = 22
        Me.lblTitleSeats.Text = "Please Select Seats You Would Like To Occupy"
        Me.lblTitleSeats.TextAlign = System.Drawing.ContentAlignment.MiddleCenter
        '
        'frmSeats
        '
        Me.AutoScaleBaseSize = New System.Drawing.Size(5, 13)
        Me.BackgroundImage = CType(resources.GetObject("$this.BackgroundImage"),
System.Drawing.Image)
        Me.ClientSize = New System.Drawing.Size(992, 664)
```

```
        Me.Controls.Add(Me.lblTitleSeats)
        Me.Controls.Add(Me.btnSubmit)
        Me.Controls.Add(Me.btnBack)
        Me.Controls.Add(Me.lblSreen)
        Me.Font = New System.Drawing.Font("Arial", 8.25!,
System.Drawing.FontStyle.Regular, System.Drawing.GraphicsUnit.Point, CType(0, Byte))
        Me.ForeColor = System.Drawing.SystemColors.ControlLight
        Me.FormBorderStyle = System.Windows.Forms.FormBorderStyle.None
        Me.Name = "frmSeats"
        Me.Text = "Seat Selection"
        Me.WindowState = System.Windows.Forms.FormWindowState.Maximized
        Me.ResumeLayout(False)

    End Sub

#End Region

    Private Sub frmSeats_Closed(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Closed
        myMain.Show()
    End Sub

    Private Sub btnSubmit_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnSubmit.Click
        'myMain.tickets(myMain.movieId - 1) = myMain.tickets(myMain.movieId - 1) -
(cbQuantity.SelectedIndex + 1)
        Me.Hide()
        'myMain.nTickets = cbQuantity.SelectedIndex + 1
        Dim i As Integer
        Dim cnt As Integer
        cnt = 0
        myMain.selectedSeats = ""
        For i = 0 To myMain.NUMSEATS - 1
            With btnSeats(i)
                If Equals(.BackColor, System.Drawing.Color.Yellow) Then
                    If myMain.selectedSeats.Length > 0 Then
                        myMain.selectedSeats = myMain.selectedSeats + ", " + CStr(i + 1)
                    Else
                        myMain.selectedSeats = CStr(i + 1)
                    End If
                    cnt = cnt + 1
                End If
            End With
        Next
        myMain.nTickets = cnt
        myMain.myTicketPurchase = New frmTicketPurchase(myMain)
        myMain.myTicketPurchase.Show()
    End Sub

    Private Sub btnBack_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles btnBack.Click
        Me.Hide()
        myMain.myMoviePreview.show()
    End Sub

    Private Sub btnSeatsClick(ByVal sender As System.Object, ByVal e As EventArgs)
        ' see which button was clicked, check out its index in the array
        With btnSeats(sender.Tag)
            If Equals(.BackColor, System.Drawing.Color.Green) Then
                .BackColor = System.Drawing.Color.Yellow
            Else
                .BackColor = System.Drawing.Color.Green
            End If
        End With
    End Sub

    Private Sub frmSeats_Load(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles MyBase.Load

    End Sub
End Class
```

TICKET PURCHASE PAGE

```
Imports System.Configuration
Public Class frmTicketPurchase
  Inherits System.Windows.Forms.Form
  Dim myMain As Object

#Region " Windows Form Designer generated code "

  Public Sub New()
    MyBase.New()

    'This call is required by the Windows Form Designer.
    InitializeComponent()

    'Add any initialization after the InitializeComponent() call

  End Sub

  Public Sub New(ByVal caller As Object)
    MyBase.New()

    'This call is required by the Windows Form Designer.
    InitializeComponent()

    'Add any initialization after the InitializeComponent() call
    myMain = caller
  End Sub

  'Form overrides dispose to clean up the component list.
  Protected Overloads Overrides Sub Dispose(ByVal disposing As Boolean)
    If disposing Then
      If Not (components Is Nothing) Then
        components.Dispose()
      End If
    End If
    MyBase.Dispose(disposing)
  End Sub

  'Required by the Windows Form Designer
  Private components As System.ComponentModel.IContainer

  'NOTE: The following procedure is required by the Windows Form Designer
  'It can be modified using the Windows Form Designer.
  'Do not modify it using the code editor.
    Friend WithEvents btnSubmit As System.Windows.Forms.Button
    Friend WithEvents Button1 As System.Windows.Forms.Button
    Friend WithEvents btnClear As System.Windows.Forms.Button
    Friend WithEvents Button2 As System.Windows.Forms.Button
    Friend WithEvents Button3 As System.Windows.Forms.Button
    Friend WithEvents Button4 As System.Windows.Forms.Button
    Friend WithEvents Button5 As System.Windows.Forms.Button
    Friend WithEvents Button6 As System.Windows.Forms.Button
    Friend WithEvents Button7 As System.Windows.Forms.Button
    Friend WithEvents Button8 As System.Windows.Forms.Button
    Friend WithEvents Button9 As System.Windows.Forms.Button
    Friend WithEvents Button10 As System.Windows.Forms.Button
    Friend WithEvents grbPaymethod As System.Windows.Forms.GroupBox
    Friend WithEvents rbtDebit As System.Windows.Forms.RadioButton
    Friend WithEvents rbtCredit As System.Windows.Forms.RadioButton
    Friend WithEvents rbtCash As System.Windows.Forms.RadioButton
    Friend WithEvents txtCardNo As System.Windows.Forms.TextBox
    Friend WithEvents lblCardNo As System.Windows.Forms.Label
    Friend WithEvents lblCardExpiry As System.Windows.Forms.Label
    Friend WithEvents MonthCalendar1 As System.Windows.Forms.MonthCalendar
    Friend WithEvents lblQuantity As System.Windows.Forms.Label
    Friend WithEvents lblNumSeats As System.Windows.Forms.Label
    Friend WithEvents lblSeats As System.Windows.Forms.Label
    Friend WithEvents lblSelectedSeats As System.Windows.Forms.Label
    Friend WithEvents btnBack As System.Windows.Forms.Button
```

```
    <System.Diagnostics.DebuggerStepThrough()> Private Sub InitializeComponent()
        Dim resources As System.Resources.ResourceManager = New
System.Resources.ResourceManager(GetType(frmTicketPurchase))
        Me.btnSubmit = New System.Windows.Forms.Button
        Me.txtCardNo = New System.Windows.Forms.TextBox
        Me.Button1 = New System.Windows.Forms.Button
        Me.btnClear = New System.Windows.Forms.Button
        Me.lblCardNo = New System.Windows.Forms.Label
        Me.Button2 = New System.Windows.Forms.Button
        Me.Button3 = New System.Windows.Forms.Button
        Me.Button4 = New System.Windows.Forms.Button
        Me.Button5 = New System.Windows.Forms.Button
        Me.Button6 = New System.Windows.Forms.Button
        Me.Button7 = New System.Windows.Forms.Button
        Me.Button8 = New System.Windows.Forms.Button
        Me.Button9 = New System.Windows.Forms.Button
        Me.Button10 = New System.Windows.Forms.Button
        Me.grbPaymethod = New System.Windows.Forms.GroupBox
        Me.rbtCash = New System.Windows.Forms.RadioButton
        Me.rbtCredit = New System.Windows.Forms.RadioButton
        Me.rbtDebit = New System.Windows.Forms.RadioButton
        Me.btnBack = New System.Windows.Forms.Button
        Me.lblCardExpiry = New System.Windows.Forms.Label
        Me.MonthCalendar1 = New System.Windows.Forms.MonthCalendar
        Me.lblSeats = New System.Windows.Forms.Label
        Me.lblSelectedSeats = New System.Windows.Forms.Label
        Me.lblQuantity = New System.Windows.Forms.Label
        Me.lblNumSeats = New System.Windows.Forms.Label
        Me.grbPaymethod.SuspendLayout()
        Me.SuspendLayout()
        '
        'btnSubmit
        '
        Me.btnSubmit.BackgroundImage =
CType(resources.GetObject("btnSubmit.BackgroundImage"), System.Drawing.Image)
        Me.btnSubmit.Font = New System.Drawing.Font("Arial Rounded MT Bold", 15.75!,
System.Drawing.FontStyle.Bold, System.Drawing.GraphicsUnit.Point, CType(0, Byte))
        Me.btnSubmit.ForeColor = System.Drawing.Color.GreenYellow
        Me.btnSubmit.Location = New System.Drawing.Point(640, 544)
        Me.btnSubmit.Name = "btnSubmit"
        Me.btnSubmit.Size = New System.Drawing.Size(192, 88)
        Me.btnSubmit.TabIndex = 13
        Me.btnSubmit.Text = "SUBMIT"
        '
        'txtCardNo
        '
        Me.txtCardNo.Font = New System.Drawing.Font("Microsoft Sans Serif", 9.75!,
System.Drawing.FontStyle.Bold, System.Drawing.GraphicsUnit.Point, CType(0, Byte))
        Me.txtCardNo.Location = New System.Drawing.Point(344, 136)
        Me.txtCardNo.MaxLength = 10
        Me.txtCardNo.Name = "txtCardNo"
        Me.txtCardNo.Size = New System.Drawing.Size(136, 22)
        Me.txtCardNo.TabIndex = 3
        Me.txtCardNo.Text = ""
        Me.txtCardNo.TextAlign = System.Windows.Forms.HorizontalAlignment.Center
        '
        'Button1
        '
        Me.Button1.BackgroundImage =
CType(resources.GetObject("Button1.BackgroundImage"), System.Drawing.Image)
        Me.Button1.Font = New System.Drawing.Font("Arial Rounded MT Bold", 12.0!,
System.Drawing.FontStyle.Bold, System.Drawing.GraphicsUnit.Point, CType(0, Byte))
        Me.Button1.ForeColor = System.Drawing.Color.GreenYellow
        Me.Button1.Location = New System.Drawing.Point(208, 272)
        Me.Button1.Name = "Button1"
        Me.Button1.Size = New System.Drawing.Size(80, 64)
        Me.Button1.TabIndex = 6
        Me.Button1.Text = "1"
        '
        'btnClear
        '
```

```
        Me.btnClear.BackgroundImage =
CType(resources.GetObject("btnClear.BackgroundImage"), System.Drawing.Image)
        Me.btnClear.Font = New System.Drawing.Font("Arial Rounded MT Bold", 15.75!,
System.Drawing.FontStyle.Bold, System.Drawing.GraphicsUnit.Point, CType(0, Byte))
        Me.btnClear.ForeColor = System.Drawing.Color.GreenYellow
        Me.btnClear.Location = New System.Drawing.Point(392, 544)
        Me.btnClear.Name = "btnClear"
        Me.btnClear.Size = New System.Drawing.Size(192, 88)
        Me.btnClear.TabIndex = 16
        Me.btnClear.Text = "RESET"
        '
        'lblCardNo
        '
        Me.lblCardNo.BackColor = System.Drawing.SystemColors.ActiveCaptionText
        Me.lblCardNo.Font = New System.Drawing.Font("Arial Rounded MT Bold", 15.75!,
System.Drawing.FontStyle.Bold, System.Drawing.GraphicsUnit.Point, CType(0, Byte))
        Me.lblCardNo.ForeColor = System.Drawing.Color.GreenYellow
        Me.lblCardNo.Location = New System.Drawing.Point(152, 136)
        Me.lblCardNo.Name = "lblCardNo"
        Me.lblCardNo.Size = New System.Drawing.Size(176, 24)
        Me.lblCardNo.TabIndex = 2
        Me.lblCardNo.Text = "Enter Card No :"
        '
        'Button2
        '
        Me.Button2.BackgroundImage =
CType(resources.GetObject("Button2.BackgroundImage"), System.Drawing.Image)
        Me.Button2.Font = New System.Drawing.Font("Arial Rounded MT Bold", 12.0!,
System.Drawing.FontStyle.Bold, System.Drawing.GraphicsUnit.Point, CType(0, Byte))
        Me.Button2.ForeColor = System.Drawing.Color.GreenYellow
        Me.Button2.Location = New System.Drawing.Point(328, 272)
        Me.Button2.Name = "Button2"
        Me.Button2.Size = New System.Drawing.Size(80, 64)
        Me.Button2.TabIndex = 7
        Me.Button2.Text = "2"
        '
        'Button3
        '
        Me.Button3.BackgroundImage =
CType(resources.GetObject("Button3.BackgroundImage"), System.Drawing.Image)
        Me.Button3.Font = New System.Drawing.Font("Arial Rounded MT Bold", 12.0!,
System.Drawing.FontStyle.Bold, System.Drawing.GraphicsUnit.Point, CType(0, Byte))
        Me.Button3.ForeColor = System.Drawing.Color.GreenYellow
        Me.Button3.Location = New System.Drawing.Point(448, 272)
        Me.Button3.Name = "Button3"
        Me.Button3.Size = New System.Drawing.Size(80, 64)
        Me.Button3.TabIndex = 8
        Me.Button3.Text = "3"
        '
        'Button4
        '
        Me.Button4.BackgroundImage =
CType(resources.GetObject("Button4.BackgroundImage"), System.Drawing.Image)
        Me.Button4.Font = New System.Drawing.Font("Arial Rounded MT Bold", 12.0!,
System.Drawing.FontStyle.Bold, System.Drawing.GraphicsUnit.Point, CType(0, Byte))
        Me.Button4.ForeColor = System.Drawing.Color.GreenYellow
        Me.Button4.Location = New System.Drawing.Point(208, 352)
        Me.Button4.Name = "Button4"
        Me.Button4.Size = New System.Drawing.Size(80, 64)
        Me.Button4.TabIndex = 8
        Me.Button4.Text = "4"
        '
        'Button5
        '
        Me.Button5.BackgroundImage =
CType(resources.GetObject("Button5.BackgroundImage"), System.Drawing.Image)
        Me.Button5.Font = New System.Drawing.Font("Arial Rounded MT Bold", 12.0!,
System.Drawing.FontStyle.Bold, System.Drawing.GraphicsUnit.Point, CType(0, Byte))
        Me.Button5.ForeColor = System.Drawing.Color.GreenYellow
        Me.Button5.Location = New System.Drawing.Point(328, 352)
        Me.Button5.Name = "Button5"
```

```
        Me.Button5.Size = New System.Drawing.Size(80, 64)
        Me.Button5.TabIndex = 9
        Me.Button5.Text = "5"
        '
        'Button6
        '
        Me.Button6.BackgroundImage =
CType(resources.GetObject("Button6.BackgroundImage"), System.Drawing.Image)
        Me.Button6.Font = New System.Drawing.Font("Arial Rounded MT Bold", 12.0!,
System.Drawing.FontStyle.Bold, System.Drawing.GraphicsUnit.Point, CType(0, Byte))
        Me.Button6.ForeColor = System.Drawing.Color.GreenYellow
        Me.Button6.Location = New System.Drawing.Point(448, 352)
        Me.Button6.Name = "Button6"
        Me.Button6.Size = New System.Drawing.Size(80, 64)
        Me.Button6.TabIndex = 10
        Me.Button6.Text = "6"
        '
        'Button7
        '
        Me.Button7.BackgroundImage =
CType(resources.GetObject("Button7.BackgroundImage"), System.Drawing.Image)
        Me.Button7.Font = New System.Drawing.Font("Arial Rounded MT Bold", 12.0!,
System.Drawing.FontStyle.Bold, System.Drawing.GraphicsUnit.Point, CType(0, Byte))
        Me.Button7.ForeColor = System.Drawing.Color.GreenYellow
        Me.Button7.Location = New System.Drawing.Point(208, 432)
        Me.Button7.Name = "Button7"
        Me.Button7.Size = New System.Drawing.Size(80, 64)
        Me.Button7.TabIndex = 11
        Me.Button7.Text = "7"
        '
        'Button8
        '
        Me.Button8.BackgroundImage =
CType(resources.GetObject("Button8.BackgroundImage"), System.Drawing.Image)
        Me.Button8.Font = New System.Drawing.Font("Arial Rounded MT Bold", 12.0!,
System.Drawing.FontStyle.Bold, System.Drawing.GraphicsUnit.Point, CType(0, Byte))
        Me.Button8.ForeColor = System.Drawing.Color.GreenYellow
        Me.Button8.Location = New System.Drawing.Point(328, 432)
        Me.Button8.Name = "Button8"
        Me.Button8.Size = New System.Drawing.Size(80, 64)
        Me.Button8.TabIndex = 12
        Me.Button8.Text = "8"
        '
        'Button9
        '
        Me.Button9.BackgroundImage =
CType(resources.GetObject("Button9.BackgroundImage"), System.Drawing.Image)
        Me.Button9.Font = New System.Drawing.Font("Arial Rounded MT Bold", 12.0!,
System.Drawing.FontStyle.Bold, System.Drawing.GraphicsUnit.Point, CType(0, Byte))
        Me.Button9.ForeColor = System.Drawing.Color.GreenYellow
        Me.Button9.Location = New System.Drawing.Point(448, 432)
        Me.Button9.Name = "Button9"
        Me.Button9.Size = New System.Drawing.Size(80, 64)
        Me.Button9.TabIndex = 13
        Me.Button9.Text = "9"
        '
        'Button10
        '
        Me.Button10.BackgroundImage =
CType(resources.GetObject("Button10.BackgroundImage"), System.Drawing.Image)
        Me.Button10.Font = New System.Drawing.Font("Arial Rounded MT Bold", 12.0!,
System.Drawing.FontStyle.Bold, System.Drawing.GraphicsUnit.Point, CType(0, Byte))
        Me.Button10.ForeColor = System.Drawing.Color.GreenYellow
        Me.Button10.Location = New System.Drawing.Point(560, 432)
        Me.Button10.Name = "Button10"
        Me.Button10.Size = New System.Drawing.Size(80, 64)
        Me.Button10.TabIndex = 14
        Me.Button10.Text = "0"
        '
        'grbPaymethod
        '
```

```
        Me.grbPaymethod.BackgroundImage =
CType(resources.GetObject("grbPaymethod.BackgroundImage"), System.Drawing.Image)
        Me.grbPaymethod.Controls.Add(Me.rbtCash)
        Me.grbPaymethod.Controls.Add(Me.rbtCredit)
        Me.grbPaymethod.Controls.Add(Me.rbtDebit)
        Me.grbPaymethod.Font = New System.Drawing.Font("Arial Rounded MT Bold", 11.25!,
System.Drawing.FontStyle.Bold, System.Drawing.GraphicsUnit.Point, CType(0, Byte))
        Me.grbPaymethod.ForeColor = System.Drawing.Color.GreenYellow
        Me.grbPaymethod.Location = New System.Drawing.Point(152, 8)
        Me.grbPaymethod.Name = "grbPaymethod"
        Me.grbPaymethod.Size = New System.Drawing.Size(528, 120)
        Me.grbPaymethod.TabIndex = 18
        Me.grbPaymethod.TabStop = False
        Me.grbPaymethod.Text = "Payment Method"
        '
        'rbtCash
        '
        Me.rbtCash.BackgroundImage =
CType(resources.GetObject("rbtCash.BackgroundImage"), System.Drawing.Image)
        Me.rbtCash.Font = New System.Drawing.Font("Arial Rounded MT Bold", 14.25!,
System.Drawing.FontStyle.Bold, System.Drawing.GraphicsUnit.Point, CType(0, Byte))
        Me.rbtCash.Location = New System.Drawing.Point(272, 32)
        Me.rbtCash.Name = "rbtCash"
        Me.rbtCash.Size = New System.Drawing.Size(144, 24)
        Me.rbtCash.TabIndex = 2
        Me.rbtCash.Text = "Cash"
        '
        'rbtCredit
        '
        Me.rbtCredit.BackgroundImage =
CType(resources.GetObject("rbtCredit.BackgroundImage"), System.Drawing.Image)
        Me.rbtCredit.Font = New System.Drawing.Font("Arial Rounded MT Bold", 14.25!,
System.Drawing.FontStyle.Bold, System.Drawing.GraphicsUnit.Point, CType(0, Byte))
        Me.rbtCredit.ForeColor = System.Drawing.Color.GreenYellow
        Me.rbtCredit.Location = New System.Drawing.Point(40, 64)
        Me.rbtCredit.Name = "rbtCredit"
        Me.rbtCredit.Size = New System.Drawing.Size(144, 24)
        Me.rbtCredit.TabIndex = 1
        Me.rbtCredit.Text = "Credit Card"
        '
        'rbtDebit
        '
        Me.rbtDebit.BackgroundImage =
CType(resources.GetObject("rbtDebit.BackgroundImage"), System.Drawing.Image)
        Me.rbtDebit.Font = New System.Drawing.Font("Arial Rounded MT Bold", 14.25!,
System.Drawing.FontStyle.Bold, System.Drawing.GraphicsUnit.Point, CType(0, Byte))
        Me.rbtDebit.ForeColor = System.Drawing.Color.GreenYellow
        Me.rbtDebit.Location = New System.Drawing.Point(40, 32)
        Me.rbtDebit.Name = "rbtDebit"
        Me.rbtDebit.Size = New System.Drawing.Size(144, 24)
        Me.rbtDebit.TabIndex = 0
        Me.rbtDebit.Text = "Debit Card"
        '
        'btnBack
        '
        Me.btnBack.BackgroundImage =
CType(resources.GetObject("btnBack.BackgroundImage"), System.Drawing.Image)
        Me.btnBack.Font = New System.Drawing.Font("Arial Rounded MT Bold", 15.75!,
System.Drawing.FontStyle.Bold, System.Drawing.GraphicsUnit.Point, CType(0, Byte))
        Me.btnBack.ForeColor = System.Drawing.Color.GreenYellow
        Me.btnBack.Location = New System.Drawing.Point(144, 544)
        Me.btnBack.Name = "btnBack"
        Me.btnBack.Size = New System.Drawing.Size(192, 88)
        Me.btnBack.TabIndex = 19
        Me.btnBack.Text = "BACK"
        '
        'lblCardExpiry
        '
        Me.lblCardExpiry.BackColor = System.Drawing.SystemColors.ActiveCaptionText
        Me.lblCardExpiry.Font = New System.Drawing.Font("Arial Rounded MT Bold", 15.75!,
System.Drawing.FontStyle.Bold, System.Drawing.GraphicsUnit.Point, CType(0, Byte))
```

```
        Me.lblCardExpiry.ForeColor = System.Drawing.Color.GreenYellow
        Me.lblCardExpiry.Location = New System.Drawing.Point(520, 136)
        Me.lblCardExpiry.Name = "lblCardExpiry"
        Me.lblCardExpiry.Size = New System.Drawing.Size(208, 32)
        Me.lblCardExpiry.TabIndex = 20
        Me.lblCardExpiry.Text = "Card Expiry Date :"
        '
        'MonthCalendar1
        '
        Me.MonthCalendar1.Location = New System.Drawing.Point(736, 136)
        Me.MonthCalendar1.Name = "MonthCalendar1"
        Me.MonthCalendar1.TabIndex = 21
        '
        'lblSeats
        '
        Me.lblSeats.BackColor = System.Drawing.SystemColors.ActiveCaptionText
        Me.lblSeats.Font = New System.Drawing.Font("Arial Rounded MT Bold", 15.75!,
System.Drawing.FontStyle.Bold, System.Drawing.GraphicsUnit.Point, CType(0, Byte))
        Me.lblSeats.ForeColor = System.Drawing.Color.GreenYellow
        Me.lblSeats.Location = New System.Drawing.Point(152, 176)
        Me.lblSeats.Name = "lblSeats"
        Me.lblSeats.Size = New System.Drawing.Size(184, 24)
        Me.lblSeats.TabIndex = 22
        Me.lblSeats.Text = "Selected Seats :"
        '
        'lblSelectedSeats
        '
        Me.lblSelectedSeats.BackColor = System.Drawing.SystemColors.ActiveCaptionText
        Me.lblSelectedSeats.Font = New System.Drawing.Font("Arial", 11.25!,
System.Drawing.FontStyle.Bold, System.Drawing.GraphicsUnit.Point, CType(0, Byte))
        Me.lblSelectedSeats.ForeColor = System.Drawing.SystemColors.ControlLight
        Me.lblSelectedSeats.Location = New System.Drawing.Point(344, 176)
        Me.lblSelectedSeats.Name = "lblSelectedSeats"
        Me.lblSelectedSeats.Size = New System.Drawing.Size(136, 24)
        Me.lblSelectedSeats.TabIndex = 23
        Me.lblSelectedSeats.TextAlign = System.Drawing.ContentAlignment.MiddleCenter
        '
        'lblQuantity
        '
        Me.lblQuantity.BackColor = System.Drawing.SystemColors.ActiveCaptionText
        Me.lblQuantity.Font = New System.Drawing.Font("Arial", 11.25!,
System.Drawing.FontStyle.Bold, System.Drawing.GraphicsUnit.Point, CType(0, Byte))
        Me.lblQuantity.ForeColor = System.Drawing.SystemColors.ControlLight
        Me.lblQuantity.Location = New System.Drawing.Point(344, 216)
        Me.lblQuantity.Name = "lblQuantity"
        Me.lblQuantity.Size = New System.Drawing.Size(80, 24)
        Me.lblQuantity.TabIndex = 24
        Me.lblQuantity.TextAlign = System.Drawing.ContentAlignment.MiddleCenter
        '
        'lblNumSeats
        '
        Me.lblNumSeats.BackColor = System.Drawing.SystemColors.ActiveCaptionText
        Me.lblNumSeats.Font = New System.Drawing.Font("Arial Rounded MT Bold", 15.75!,
System.Drawing.FontStyle.Bold, System.Drawing.GraphicsUnit.Point, CType(0, Byte))
        Me.lblNumSeats.ForeColor = System.Drawing.Color.GreenYellow
        Me.lblNumSeats.Location = New System.Drawing.Point(160, 216)
        Me.lblNumSeats.Name = "lblNumSeats"
        Me.lblNumSeats.Size = New System.Drawing.Size(160, 24)
        Me.lblNumSeats.TabIndex = 25
        Me.lblNumSeats.Text = "No. Of Seats :"
        '
        'frmTicketPurchase
        '
        Me.AutoScaleBaseSize = New System.Drawing.Size(5, 13)
        Me.BackgroundImage = CType(resources.GetObject("$this.BackgroundImage"),
System.Drawing.Image)
        Me.ClientSize = New System.Drawing.Size(1000, 662)
        Me.Controls.Add(Me.lblNumSeats)
        Me.Controls.Add(Me.lblQuantity)
        Me.Controls.Add(Me.lblSelectedSeats)
        Me.Controls.Add(Me.lblSeats)
```

```
        Me.Controls.Add(Me.MonthCalendar1)
        Me.Controls.Add(Me.lblCardExpiry)
        Me.Controls.Add(Me.btnBack)
        Me.Controls.Add(Me.grbPaymethod)
        Me.Controls.Add(Me.Button10)
        Me.Controls.Add(Me.Button9)
        Me.Controls.Add(Me.Button8)
        Me.Controls.Add(Me.Button7)
        Me.Controls.Add(Me.Button6)
        Me.Controls.Add(Me.Button5)
        Me.Controls.Add(Me.Button4)
        Me.Controls.Add(Me.Button3)
        Me.Controls.Add(Me.Button2)
        Me.Controls.Add(Me.lblCardNo)
        Me.Controls.Add(Me.btnClear)
        Me.Controls.Add(Me.Button1)
        Me.Controls.Add(Me.txtCardNo)
        Me.Controls.Add(Me.btnSubmit)
        Me.FormBorderStyle = System.Windows.Forms.FormBorderStyle.None
        Me.MaximizeBox = False
        Me.Name = "frmTicketPurchase"
        Me.StartPosition = System.Windows.Forms.FormStartPosition.CenterScreen
        Me.Text = "TicketPurchase"
        Me.WindowState = System.Windows.Forms.FormWindowState.Maximized
        Me.grbPaymethod.ResumeLayout(False)
        Me.ResumeLayout(False)

    End Sub

#End Region

  Private Sub frmTicketPurchase_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
        lblSelectedSeats.Text = myMain.selectedSeats
        lblQuantity.Text = CStr(myMain.ntickets)
  End Sub
    'Close Ticket purchase form to show the Main page
  Private Sub frmTicketPurchase_Closed(ByVal sender As Object, ByVal e As
System.EventArgs) Handles MyBase.Closed
    myMain.Show()
  End Sub
    Private Sub btnSubmit_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnSubmit.Click
        Me.Hide()
        '--- update seats availability ---
        If myMain.selectedSeats.indexOf(",") > 0 Then
            Dim tmpString As String()
            tmpString = myMain.selectedSeats.split(",")
            Dim tmpS As String
            Dim valS As Integer
            For Each tmpS In tmpString
                valS = CInt(tmpS)
                myMain.seats(myMain.movieId - 1, myMain.slotId, valS - 1) = False
            Next
        Else
            myMain.seats(myMain.movieId - 1, myMain.slotId, CInt(myMain.selectedSeats) -
1) = False
        End If
        '--- end of update seats availability ---

        myMain.myTicketPrint = New frmPrintTicket(myMain)
        myMain.myTicketPrint.Show()
    End Sub

    'Set Pin values to asterisks to create pin number privacy.

    Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles Button1.Click
        txtCardNo.Text = txtCardNo.Text + "*"

    End Sub
```

```
    Private Sub Button2_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles Button2.Click
        txtCardNo.Text = txtCardNo.Text + "*"
    End Sub

    Private Sub Button3_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles Button3.Click
        txtCardNo.Text = txtCardNo.Text + "*"
    End Sub


    Private Sub Button4_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles Button4.Click
        txtCardNo.Text = txtCardNo.Text + "*"
    End Sub

    Private Sub Button5_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles Button5.Click
        txtCardNo.Text = txtCardNo.Text + "*"
    End Sub

    Private Sub Button6_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles Button6.Click
        txtCardNo.Text = txtCardNo.Text + "*"
    End Sub

    Private Sub Button7_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles Button7.Click
        txtCardNo.Text = txtCardNo.Text + "*"
    End Sub

    Private Sub Button8_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles Button8.Click
        txtCardNo.Text = txtCardNo.Text + "*"
    End Sub

    Private Sub Button9_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles Button9.Click
        txtCardNo.Text = txtCardNo.Text + "*"
    End Sub

    Private Sub Button10_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button10.Click
        txtCardNo.Text = txtCardNo.Text + "*"
    End Sub
    ' Reset the Pin text box to clear
    Private Sub btnClear_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnClear.Click
        txtCardNo.Text = " "
    End Sub

    Private Sub btnBack_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles btnBack.Click
        Me.Close()
        myMain.mySeats.Show()
    End Sub
    'Select Cash
    Private Sub rbtCash_CheckedChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles rbtCash.CheckedChanged
        lblCardNo.Visible = False
        txtCardNo.Visible = False
        Button1.Visible = False
        Button2.Visible = False
        Button3.Visible = False
        Button4.Visible = False
        Button5.Visible = False
        Button6.Visible = False
        Button7.Visible = False
        Button8.Visible = False
        Button9.Visible = False
        Button10.Visible = False
```

```
        lblCardExpiry.Visible = False
        MonthCalendar1.Visible = False

    End Sub

    Private Sub rbtCredit_CheckedChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles rbtCredit.CheckedChanged
        lblCardNo.Visible = True
        txtCardNo.Visible = True
        Button1.Visible = True
        Button2.Visible = True
        Button3.Visible = True
        Button4.Visible = True
        Button5.Visible = True
        Button6.Visible = True
        Button7.Visible = True
        Button8.Visible = True
        Button9.Visible = True
        Button10.Visible = True
        lblCardExpiry.Visible = True
        MonthCalendar1.Visible = True
    End Sub

    Private Sub rbtDebit_CheckedChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles rbtDebit.CheckedChanged
        lblCardNo.Visible = True
        txtCardNo.Visible = True
        Button1.Visible = True
        Button2.Visible = True
        Button3.Visible = True
        Button4.Visible = True
        Button5.Visible = True
        Button6.Visible = True
        Button7.Visible = True
        Button8.Visible = True
        Button9.Visible = True
        Button10.Visible = True
        lblCardExpiry.Visible = False
        MonthCalendar1.Visible = False
    End Sub
End Class
```

TICKET PRINT PAGE

```
Imports System.Configuration

Public Class frmPrintTicket
    Inherits System.Windows.Forms.Form
    Dim myMain As Object
    '-----printing code-----
    'create a printing document
    Private WithEvents pd As Printing.PrintDocument
    'storage for form image
    Dim formImage As Bitmap
    'create API prototoype
    Private Declare Function BitBlt Lib "gdi32.dll" Alias _
        "BitBlt" (ByVal hdcDest As IntPtr, _
        ByVal nXDest As Integer, ByVal nYDest As _
        Integer, ByVal nWidth As Integer, _
        ByVal nHeight As Integer, ByVal _
        hdcSrc As IntPtr, ByVal nXSrc As Integer, _
        ByVal nYSrc As Integer, _
        ByVal dwRop As System.Int32) As Long
    ' Callback from PrintDocument component to
    ' do the actual printing
    Private Sub pdTicket_PrintPage(ByVal sender As Object, _
       ByVal e As System.Drawing.Printing.PrintPageEventArgs) _
       Handles pdTicket.PrintPage
        e.Graphics.DrawImage(formImage, 100, 100)
```

```
    End Sub
    Private Sub GetFormImage()
        Dim g As Graphics = Me.CreateGraphics()
        Dim s As Size = Me.Size
        formImage = New Bitmap(s.Width, s.Height, g)
        Dim mg As Graphics = Graphics.FromImage(formImage)
        Dim dc1 As IntPtr = g.GetHdc
        Dim dc2 As IntPtr = mg.GetHdc
        ' added code to compute and capture the form
        ' title bar and borders
        Dim widthDiff As Integer = _
           (Me.Width - Me.ClientRectangle.Width)
        Dim heightDiff As Integer = _
           (Me.Height - Me.ClientRectangle.Height)
        Dim borderSize As Integer = widthDiff \ 2
        Dim heightTitleBar As Integer = heightDiff - borderSize
        BitBlt(dc2, 0, 0, _
           Me.ClientRectangle.Width + widthDiff, _
           Me.ClientRectangle.Height + heightDiff, dc1, _
           0 - borderSize, 0 - heightTitleBar, 13369376)

        g.ReleaseHdc(dc1)
        mg.ReleaseHdc(dc2)
    End Sub
    '-----end of printing code-----

#Region " Windows Form Designer generated code "

    Public Sub New()
        MyBase.New()

        'This call is required by the Windows Form Designer.
        InitializeComponent()

        'Add any initialization after the InitializeComponent() call

    End Sub
    Public Sub New(ByVal caller As Object)
        MyBase.New()

        'This call is required by the Windows Form Designer.
        InitializeComponent()

        'Add any initialization after the InitializeComponent() call
        myMain = caller
    End Sub

    'Form overrides dispose to clean up the component list.
    Protected Overloads Overrides Sub Dispose(ByVal disposing As Boolean)
        If disposing Then
            If Not (components Is Nothing) Then
                components.Dispose()
            End If
        End If
        MyBase.Dispose(disposing)
    End Sub

    'Required by the Windows Form Designer
    Private components As System.ComponentModel.IContainer

    'NOTE: The following procedure is required by the Windows Form Designer
    'It can be modified using the Windows Form Designer.
    'Do not modify it using the code editor.
    Friend WithEvents btnPrintPreview As System.Windows.Forms.Button
    Friend WithEvents dlgPrintPreview As System.Windows.Forms.PrintPreviewDialog
    Friend WithEvents btnHome As System.Windows.Forms.Button
    Friend WithEvents btnPrintTicket As System.Windows.Forms.Button
    Friend WithEvents lblDate As System.Windows.Forms.Label
    Friend WithEvents lblPrice As System.Windows.Forms.Label
    Private WithEvents pdTicket As System.Drawing.Printing.PrintDocument
    Friend WithEvents lblTitle As System.Windows.Forms.Label
```

```
    Friend WithEvents lblDay As System.Windows.Forms.Label
    Friend WithEvents lblName As System.Windows.Forms.Label
    Friend WithEvents lblMovieName As System.Windows.Forms.Label
    Friend WithEvents lblCharge As System.Windows.Forms.Label
    Friend WithEvents lblNote As System.Windows.Forms.Label
    Friend WithEvents lblSelectedSeats As System.Windows.Forms.Label
    Friend WithEvents lblSeats As System.Windows.Forms.Label
    Friend WithEvents lblSlot As System.Windows.Forms.Label
    Friend WithEvents lblblslotTime As System.Windows.Forms.Label
    <System.Diagnostics.DebuggerStepThrough()> Private Sub InitializeComponent()
        Dim resources As System.Resources.ResourceManager = New
System.Resources.ResourceManager(GetType(frmPrintTicket))
        Me.btnPrintPreview = New System.Windows.Forms.Button
        Me.btnPrintTicket = New System.Windows.Forms.Button
        Me.dlgPrintPreview = New System.Windows.Forms.PrintPreviewDialog
        Me.btnHome = New System.Windows.Forms.Button
        Me.lblDate = New System.Windows.Forms.Label
        Me.lblPrice = New System.Windows.Forms.Label
        Me.lblName = New System.Windows.Forms.Label
        Me.pdTicket = New System.Drawing.Printing.PrintDocument
        Me.lblTitle = New System.Windows.Forms.Label
        Me.lblDay = New System.Windows.Forms.Label
        Me.lblMovieName = New System.Windows.Forms.Label
        Me.lblCharge = New System.Windows.Forms.Label
        Me.lblNote = New System.Windows.Forms.Label
        Me.lblSelectedSeats = New System.Windows.Forms.Label
        Me.lblSeats = New System.Windows.Forms.Label
        Me.lblblslotTime = New System.Windows.Forms.Label
        Me.lblSlot = New System.Windows.Forms.Label
        Me.SuspendLayout()
        '
        'btnPrintPreview
        '
        Me.btnPrintPreview.BackgroundImage =
CType(resources.GetObject("btnPrintPreview.BackgroundImage"), System.Drawing.Image)
        Me.btnPrintPreview.Font = New System.Drawing.Font("Arial Rounded MT Bold",
15.75!, System.Drawing.FontStyle.Bold, System.Drawing.GraphicsUnit.Point, CType(0, Byte))
        Me.btnPrintPreview.ForeColor = System.Drawing.Color.GreenYellow
        Me.btnPrintPreview.Location = New System.Drawing.Point(144, 104)
        Me.btnPrintPreview.Name = "btnPrintPreview"
        Me.btnPrintPreview.Size = New System.Drawing.Size(0, 0)
        Me.btnPrintPreview.TabIndex = 0
        Me.btnPrintPreview.Text = "PRINT PREVIEW"
        '
        'btnPrintTicket
        '
        Me.btnPrintTicket.BackgroundImage =
CType(resources.GetObject("btnPrintTicket.BackgroundImage"), System.Drawing.Image)
        Me.btnPrintTicket.Font = New System.Drawing.Font("Arial Rounded MT Bold", 15.75!,
System.Drawing.FontStyle.Bold, System.Drawing.GraphicsUnit.Point, CType(0, Byte))
        Me.btnPrintTicket.ForeColor = System.Drawing.Color.GreenYellow
        Me.btnPrintTicket.Location = New System.Drawing.Point(32, 504)
        Me.btnPrintTicket.Name = "btnPrintTicket"
        Me.btnPrintTicket.Size = New System.Drawing.Size(176, 56)
        Me.btnPrintTicket.TabIndex = 2
        Me.btnPrintTicket.Text = "PRINT NOW"
        '
        'dlgPrintPreview
        '
        Me.dlgPrintPreview.AutoScrollMargin = New System.Drawing.Size(0, 0)
        Me.dlgPrintPreview.AutoScrollMinSize = New System.Drawing.Size(0, 0)
        Me.dlgPrintPreview.ClientSize = New System.Drawing.Size(400, 300)
        Me.dlgPrintPreview.Enabled = True
        Me.dlgPrintPreview.Icon = CType(resources.GetObject("dlgPrintPreview.Icon"),
System.Drawing.Icon)
        Me.dlgPrintPreview.Location = New System.Drawing.Point(44, 54)
        Me.dlgPrintPreview.MinimumSize = New System.Drawing.Size(375, 250)
        Me.dlgPrintPreview.Name = "dlgPrintPreview"
        Me.dlgPrintPreview.TransparencyKey = System.Drawing.Color.Empty
        Me.dlgPrintPreview.Visible = False
        '
```

```
        'btnHome
        '
        Me.btnHome.BackgroundImage =
CType(resources.GetObject("btnHome.BackgroundImage"), System.Drawing.Image)
        Me.btnHome.Font = New System.Drawing.Font("Arial Rounded MT Bold", 15.75!,
System.Drawing.FontStyle.Bold, System.Drawing.GraphicsUnit.Point, CType(0, Byte))
        Me.btnHome.ForeColor = System.Drawing.Color.GreenYellow
        Me.btnHome.Location = New System.Drawing.Point(304, 504)
        Me.btnHome.Name = "btnHome"
        Me.btnHome.Size = New System.Drawing.Size(152, 56)
        Me.btnHome.TabIndex = 3
        Me.btnHome.Text = "HOME"
        '
        'lblDate
        '
        Me.lblDate.Font = New System.Drawing.Font("Arial", 11.25!,
System.Drawing.FontStyle.Bold, System.Drawing.GraphicsUnit.Point, CType(0, Byte))
        Me.lblDate.ForeColor = System.Drawing.SystemColors.ControlLight
        Me.lblDate.Image = CType(resources.GetObject("lblDate.Image"),
System.Drawing.Image)
        Me.lblDate.Location = New System.Drawing.Point(168, 256)
        Me.lblDate.Name = "lblDate"
        Me.lblDate.Size = New System.Drawing.Size(184, 32)
        Me.lblDate.TabIndex = 6
        Me.lblDate.TextAlign = System.Drawing.ContentAlignment.MiddleCenter
        '
        'lblPrice
        '
        Me.lblPrice.Font = New System.Drawing.Font("Arial", 11.25!,
System.Drawing.FontStyle.Bold, System.Drawing.GraphicsUnit.Point, CType(0, Byte))
        Me.lblPrice.ForeColor = System.Drawing.SystemColors.ControlLight
        Me.lblPrice.Image = CType(resources.GetObject("lblPrice.Image"),
System.Drawing.Image)
        Me.lblPrice.Location = New System.Drawing.Point(168, 312)
        Me.lblPrice.Name = "lblPrice"
        Me.lblPrice.Size = New System.Drawing.Size(184, 32)
        Me.lblPrice.TabIndex = 7
        Me.lblPrice.TextAlign = System.Drawing.ContentAlignment.MiddleCenter
        '
        'lblName
        '
        Me.lblName.Font = New System.Drawing.Font("Arial", 11.25!,
System.Drawing.FontStyle.Bold, System.Drawing.GraphicsUnit.Point, CType(0, Byte))
        Me.lblName.ForeColor = System.Drawing.SystemColors.ControlLight
        Me.lblName.Image = CType(resources.GetObject("lblName.Image"),
System.Drawing.Image)
        Me.lblName.Location = New System.Drawing.Point(168, 192)
        Me.lblName.Name = "lblName"
        Me.lblName.Size = New System.Drawing.Size(184, 32)
        Me.lblName.TabIndex = 8
        Me.lblName.TextAlign = System.Drawing.ContentAlignment.MiddleCenter
        '
        'pdTicket
        '
        '
        'lblTitle
        '
        Me.lblTitle.Font = New System.Drawing.Font("Palatino Linotype", 18.0!,
System.Drawing.FontStyle.Bold, System.Drawing.GraphicsUnit.Point, CType(0, Byte))
        Me.lblTitle.ForeColor = System.Drawing.Color.Gold
        Me.lblTitle.Image = CType(resources.GetObject("lblTitle.Image"),
System.Drawing.Image)
        Me.lblTitle.Location = New System.Drawing.Point(64, 80)
        Me.lblTitle.Name = "lblTitle"
        Me.lblTitle.Size = New System.Drawing.Size(368, 40)
        Me.lblTitle.TabIndex = 9
        Me.lblTitle.TextAlign = System.Drawing.ContentAlignment.MiddleCenter
        '
        'lblDay
        '
```

```
        Me.lblDay.Font = New System.Drawing.Font("Arial", 11.25!,
System.Drawing.FontStyle.Bold, System.Drawing.GraphicsUnit.Point, CType(0, Byte))
        Me.lblDay.ForeColor = System.Drawing.SystemColors.ControlLight
        Me.lblDay.Image = CType(resources.GetObject("lblDay.Image"),
System.Drawing.Image)
        Me.lblDay.Location = New System.Drawing.Point(72, 256)
        Me.lblDay.Name = "lblDay"
        Me.lblDay.Size = New System.Drawing.Size(64, 24)
        Me.lblDay.TabIndex = 10
        Me.lblDay.Text = "DATE :"
        Me.lblDay.TextAlign = System.Drawing.ContentAlignment.MiddleCenter
        '
        'lblMovieName
        '
        Me.lblMovieName.BackColor = System.Drawing.SystemColors.Desktop
        Me.lblMovieName.Font = New System.Drawing.Font("Arial", 11.25!,
System.Drawing.FontStyle.Bold, System.Drawing.GraphicsUnit.Point, CType(0, Byte))
        Me.lblMovieName.ForeColor = System.Drawing.SystemColors.ControlLight
        Me.lblMovieName.Image = CType(resources.GetObject("lblMovieName.Image"),
System.Drawing.Image)
        Me.lblMovieName.Location = New System.Drawing.Point(72, 192)
        Me.lblMovieName.Name = "lblMovieName"
        Me.lblMovieName.Size = New System.Drawing.Size(72, 32)
        Me.lblMovieName.TabIndex = 11
        Me.lblMovieName.Text = "MOVIE :"
        Me.lblMovieName.TextAlign = System.Drawing.ContentAlignment.MiddleCenter
        '
        'lblCharge
        '
        Me.lblCharge.Font = New System.Drawing.Font("Arial", 11.25!,
System.Drawing.FontStyle.Bold, System.Drawing.GraphicsUnit.Point, CType(0, Byte))
        Me.lblCharge.ForeColor = System.Drawing.SystemColors.ControlLight
        Me.lblCharge.Image = CType(resources.GetObject("lblCharge.Image"),
System.Drawing.Image)
        Me.lblCharge.Location = New System.Drawing.Point(72, 304)
        Me.lblCharge.Name = "lblCharge"
        Me.lblCharge.Size = New System.Drawing.Size(72, 40)
        Me.lblCharge.TabIndex = 12
        Me.lblCharge.Text = "TOTAL PRICE :"
        Me.lblCharge.TextAlign = System.Drawing.ContentAlignment.MiddleCenter
        '
        'lblNote
        '
        Me.lblNote.Font = New System.Drawing.Font("Arial", 11.25!,
System.Drawing.FontStyle.Bold, System.Drawing.GraphicsUnit.Point, CType(0, Byte))
        Me.lblNote.ForeColor = System.Drawing.SystemColors.ControlLight
        Me.lblNote.Image = CType(resources.GetObject("lblNote.Image"),
System.Drawing.Image)
        Me.lblNote.Location = New System.Drawing.Point(80, 456)
        Me.lblNote.Name = "lblNote"
        Me.lblNote.Size = New System.Drawing.Size(336, 24)
        Me.lblNote.TabIndex = 13
        Me.lblNote.Text = "Thank you for using our self-service system."
        Me.lblNote.TextAlign = System.Drawing.ContentAlignment.BottomCenter
        '
        'lblSelectedSeats
        '
        Me.lblSelectedSeats.Font = New System.Drawing.Font("Arial", 11.25!,
System.Drawing.FontStyle.Bold, System.Drawing.GraphicsUnit.Point, CType(0, Byte))
        Me.lblSelectedSeats.ForeColor = System.Drawing.SystemColors.ControlLight
        Me.lblSelectedSeats.Image = CType(resources.GetObject("lblSelectedSeats.Image"),
System.Drawing.Image)
        Me.lblSelectedSeats.Location = New System.Drawing.Point(72, 360)
        Me.lblSelectedSeats.Name = "lblSelectedSeats"
        Me.lblSelectedSeats.Size = New System.Drawing.Size(80, 32)
        Me.lblSelectedSeats.TabIndex = 14
        Me.lblSelectedSeats.Text = "SEAT(S) :"
        Me.lblSelectedSeats.TextAlign = System.Drawing.ContentAlignment.MiddleCenter
        '
        'lblSeats
        '
```

```
        Me.lblSeats.Font = New System.Drawing.Font("Arial", 11.25!,
System.Drawing.FontStyle.Bold, System.Drawing.GraphicsUnit.Point, CType(0, Byte))
        Me.lblSeats.ForeColor = System.Drawing.SystemColors.ControlLight
        Me.lblSeats.Image = CType(resources.GetObject("lblSeats.Image"),
System.Drawing.Image)
        Me.lblSeats.Location = New System.Drawing.Point(168, 360)
        Me.lblSeats.Name = "lblSeats"
        Me.lblSeats.Size = New System.Drawing.Size(184, 32)
        Me.lblSeats.TabIndex = 15
        Me.lblSeats.TextAlign = System.Drawing.ContentAlignment.MiddleCenter
        '
        'lblblslotTime
        '
        Me.lblblslotTime.Font = New System.Drawing.Font("Arial", 11.25!,
System.Drawing.FontStyle.Bold, System.Drawing.GraphicsUnit.Point, CType(0, Byte))
        Me.lblblslotTime.ForeColor = System.Drawing.SystemColors.ControlLight
        Me.lblblslotTime.Image = CType(resources.GetObject("lblblslotTime.Image"),
System.Drawing.Image)
        Me.lblblslotTime.Location = New System.Drawing.Point(72, 408)
        Me.lblblslotTime.Name = "lblblslotTime"
        Me.lblblslotTime.Size = New System.Drawing.Size(72, 24)
        Me.lblblslotTime.TabIndex = 16
        Me.lblblslotTime.Text = "SLOT"
        Me.lblblslotTime.TextAlign = System.Drawing.ContentAlignment.MiddleCenter
        '
        'lblSlot
        '
        Me.lblSlot.Font = New System.Drawing.Font("Arial", 11.25!,
System.Drawing.FontStyle.Bold, System.Drawing.GraphicsUnit.Point, CType(0, Byte))
        Me.lblSlot.ForeColor = System.Drawing.SystemColors.ControlLight
        Me.lblSlot.Image = CType(resources.GetObject("lblSlot.Image"),
System.Drawing.Image)
        Me.lblSlot.Location = New System.Drawing.Point(168, 408)
        Me.lblSlot.Name = "lblSlot"
        Me.lblSlot.Size = New System.Drawing.Size(184, 24)
        Me.lblSlot.TabIndex = 17
        Me.lblSlot.TextAlign = System.Drawing.ContentAlignment.MiddleCenter
        '
        'frmPrintTicket
        '
        Me.AutoScaleBaseSize = New System.Drawing.Size(5, 13)
        Me.BackgroundImage = CType(resources.GetObject("$this.BackgroundImage"),
System.Drawing.Image)
        Me.ClientSize = New System.Drawing.Size(488, 600)
        Me.Controls.Add(Me.lblSlot)
        Me.Controls.Add(Me.lblblslotTime)
        Me.Controls.Add(Me.lblSeats)
        Me.Controls.Add(Me.lblSelectedSeats)
        Me.Controls.Add(Me.lblNote)
        Me.Controls.Add(Me.lblCharge)
        Me.Controls.Add(Me.lblMovieName)
        Me.Controls.Add(Me.lblDay)
        Me.Controls.Add(Me.lblTitle)
        Me.Controls.Add(Me.lblName)
        Me.Controls.Add(Me.lblPrice)
        Me.Controls.Add(Me.lblDate)
        Me.Controls.Add(Me.btnHome)
        Me.Controls.Add(Me.btnPrintTicket)
        Me.Controls.Add(Me.btnPrintPreview)
        Me.FormBorderStyle = System.Windows.Forms.FormBorderStyle.None
        Me.Name = "frmPrintTicket"
        Me.Text = "Print_Ticket"
        Me.ResumeLayout(False)

    End Sub

#End Region


    'Prints Immediately
```

```
    Private Sub btnPrintTicket_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnPrintTicket.Click
        btnHome.Visible = False
        '-----printing code-----
        btnPrintTicket.Visible = False
        GetFormImage()
        pdTicket.Print()
        btnPrintTicket.Visible = True
        '-----end of printing code-----
        btnHome.Visible = True
    End Sub

    Private Sub btnHome_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles btnHome.Click
        Me.Close()
        myMain.show()
    End Sub

    Private Sub frmPrintTicket_Closed(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Closed
        myMain.show()
    End Sub

    Private Sub frmPrintTicket_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
    End Sub

    Private Sub frmTicketPrint_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
        Dim i As Integer
        lblName.Text = ConfigurationSettings.AppSettings("Movie" + CStr(myMain.movieId) +
"Name")
        lblDate.Text = Now
        lblPrice.Text = "RM" + CStr(myMain.nTickets *
ConfigurationSettings.AppSettings("Movie" + CStr(myMain.movieId) + "Price"))
        lblTitle.Text = "Broadway Cinemas E-Ticketing"
        lblSeats.Text = myMain.selectedSeats
        lblSlot.Text = ConfigurationSettings.AppSettings("Movie" + CStr(myMain.movieId) +
"Slot" + CStr(myMain.slotId + 1) + "Time")

    End Sub
End Class
```

APPLICATION CONFIGURATION (DATABASE)

```
<?xml version="1.0" encoding="utf-8" ?>
<configuration>
	<appSettings>
		<add key="Movie1Name" value="Shrek 3" />
		<add key="Movie2Name" value="Hairspray" />
		<add key="Movie3Name" value="Becoming Jane" />
		<add key="Movie4Name" value="Fantastic Four" />
		<add key="Movie5Name" value="Happy Feet" />
		<add key="Movie6Name" value="No Reservations" />

		<add key="Movie1Price" value="7.00" />
		<add key="Movie2Price" value="8.00" />
		<add key="Movie3Price" value="8.00" />
		<add key="Movie4Price" value="7.00" />
		<add key="Movie5Price" value="7.00" />
		<add key="Movie6Price" value="8.00" />

		<add key="Movie1Runtime" value="1hr33min" />
		<add key="Movie2Runtime" value="1hr53min" />
		<add key="Movie3Runtime" value="1hr53min" />
		<add key="Movie4Runtime" value="1hr29min" />
		<add key="Movie5Runtime" value="1hr49min" />
		<add key="Movie6Runtime" value="1hr43min" />

		<add key="Movie1Slots" value="5" />
		<add key="Movie1Slot1Time" value="09:15" />
		<add key="Movie1Slot1Seats" value="1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12,
13, 14 , 15, 23, 25, 27, 28" />
		<add key="Movie1Slot2Time" value="11:30" />
		<add key="Movie1Slot2Seats" value="2, 3, 4, 5, 6, 7, 8, 9, 10, 12, 13, 14,
18, 26, 27, 28, 30, 35, 36" />
		<add key="Movie1Slot3Time" value="14:15" />
		<add key="Movie1Slot3Seats" value="3, 4, 5, 6, 7, 8, 9, 10, 11, 13, 15,
16, 17, 18, 20, 22, 23, 25, 28" />
		<add key="Movie1Slot4Time" value="17:00" />
		<add key="Movie1Slot4Seats" value="4, 5, 6, 7, 8, 9, 10, 11, 22, 23, 24,
25, 26, 34, 35, 36, 37, 38, 40" />
		<add key="Movie1Slot5Time" value="19:30" />
		<add key="Movie1Slot5Seats" value="11, 12, 13, 14, 15, 16, 17, 18, 19, 20,
31, 32, 33, 34, 35, 36, 37, 29" />

		<add key="Movie2Slots" value="5" />
		<add key="Movie2Slot1Time" value="11:30" />
		<add key="Movie2Slot1Seats" value="1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 14 ,
16, 17, 18, 23, 25, 28, 29" />
		<add key="Movie2Slot2Time" value="14:15" />
		<add key="Movie2Slot2Seats" value="2, 3, 4, 5, 6, 7, 8, 9, 10, 12, 13, 14,
18, 19, 26, 27 , 28, 29, 30" />
		<add key="Movie2Slot3Time" value="17:00" />
		<add key="Movie2Slot3Seats" value="3, 4, 5, 6, 7, 8, 9, 10, 11, 13, 15,
16, 17, 18, 20, 22, 31, 32, 33" />
		<add key="Movie2Slot4Time" value="19:30" />
		<add key="Movie2Slot4Seats" value="4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14,
15, 22, 23, 24, 25, 26, 36, 37" />
		<add key="Movie2Slot5Time" value="22:00" />
		<add key="Movie2Slot5Seats" value="11, 12, 13, 14, 15, 16, 17, 18, 19, 20,
31, 32, 33, 34, 35, 36, 38" />

		<add key="Movie3Slots" value="5" />
		<add key="Movie3Slot1Time" value="11:30" />
		<add key="Movie3Slot1Seats" value="1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 14 ,
23, 25, 28, 29, 30, 32, 34, 36" />
		<add key="Movie3Slot2Time" value="14:15" />
		<add key="Movie3Slot2Seats" value="2, 3, 4, 5, 6, 7, 12, 13, 14, 15, 16,
17, 18,19, 20, 31, 32, 32, 34, 35" />
		<add key="Movie3Slot3Time" value="17:00" />
		<add key="Movie3Slot3Seats" value="1, 2, 3, 4, 5, 15, 16 , 17, 18, 19, 20,
21, 22, 23, 24, 25, 35, 36, 37" />
```

```
            <add key="Movie3Slot4Time" value="19:30" />
            <add key="Movie3Slot4Seats" value="4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 15,
21, 22, 23, 24, 25, 26, 34, 37, 40" />
            <add key="Movie3Slot5Time" value="22:00" />
            <add key="Movie3Slot5Seats" value="1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 21, 22,
23, 24, 25, 26, 27, 28, 29, 30, 35" />

            <add key="Movie4Slots" value="5" />
            <add key="Movie4Slot1Time" value="09:15" />
            <add key="Movie4Slot1Seats" value="1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 14,
15, 16, 17, 21, 22 , 23, 25, 28, 29" />
            <add key="Movie4Slot2Time" value="11:30" />
            <add key="Movie4Slot2Seats" value="7, 8, 9, 10, 11, 12, 13, 15, 16, 17,
18, 20, 21, 22, 23, 24, 31, 32, 33, 34" />
            <add key="Movie4Slot3Time" value="14:15" />
            <add key="Movie4Slot3Seats" value="4, 5, 6, 7, 8, 9, 10, 14, 15, 21, 22,
23, 24, 25, 26, 35, 36, 37, 38, 39, 40" />
            <add key="Movie4Slot4Time" value="17:00" />
            <add key="Movie4Slot4Seats" value="1, 2, 3, 4, 5, 15, 16 , 17, 18, 19, 20,
21, 22, 23, 24, 25, 35, 36, 37, 38" />
            <add key="Movie4Slot5Time" value="19:30" />
            <add key="Movie4Slot5Seats" value="1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 21, 22,
23, 24, 25, 26, 27, 28, 29, 30, 31, 32"/>

            <add key="Movie5Slots" value="5" />
            <add key="Movie5Slot1Time" value="09:15" />
            <add key="Movie5Slot1Seats" value="1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 14 ,
15, 16, 17, 18, 23, 25, 27, 28, 29" />
            <add key="Movie5Slot2Time" value="11:30" />
            <add key="Movie5Slot2Seats" value="2, 3, 4, 5, 6, 7, 8, 9, 10, 22, 23, 24,
26, 27, 28, 30, 32, 33, 34, 35, 36" />
            <add key="Movie5Slot3Time" value="14:15" />
            <add key="Movie5Slot3Seats" value="7, 8, 9, 10, 11, 12, 13, 15, 16, 17,
18, 20, 21, 22, 23, 24, 31, 32, 33, 34" />
            <add key="Movie5Slot4Time" value="17:00" />
            <add key="Movie5Slot4Seats" value="4, 5, 6, 7, 8, 9, 10, 14, 15, 21, 22,
23, 24, 25, 26, 35, 36, 37, 38, 39, 40" />
            <add key="Movie5Slot5Time" value="19:30" />
            <add key="Movie5Slot5Seats" value="11, 12, 13, 14, 15, 16, 17, 18, 19, 20,
21, 22, 23, 24, 25, 26, 27, 28, 29" />

            <add key="Movie6Slots" value="5" />
            <add key="Movie6Slot1Time" value="11:30" />
            <add key="Movie6Slot1Seats" value="1, 2, 3, 4, 5, 15, 16 , 17, 18, 19, 20,
21, 22, 23, 24, 25, 35, 36, 37" />
            <add key="Movie6Slot2Time" value="14:15" />
            <add key="Movie6Slot2Seats" value="2, 3, 4, 5, 6, 7, 12, 13, 14, 15, 16,
17, 18,19, 20, 31, 32, 32, 34, 35" />
            <add key="Movie6Slot3Time" value="17:00" />
            <add key="Movie6Slot3Seats" value="3, 4, 5, 6, 7, 8, 9, 10, 11, 13, 15,
16, 17, 18, 19, 20, 2ø, 23, 24, 25" />
            <add key="Movie6Slot4Time" value="19:30" />
            <add key="Movie6Slot4Seats" value="11, 12, 13, 14, 15, 16, 17, 18, 19, 20,
21, 22, 23, 24, 25, 26, 27, 28, 29" />
            <add key="Movie6Slot5Time" value="22:00" />
            <add key="Movie6Slot5Seats" value="1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 21, 22,
23, 24, 25, 26, 27, 28, 29, 30, 32"/>

            </appSettings>
</configuration>
```