BATTLE OF FRANCE WWII

A Thesis
Presented to the
Faculty of
San Diego State University

In Partial Fulfillment
of the Requirements for the Degree
Master of Science
in
Computer Science

by
Arpitha Rao Gadhath
Fall 2014
SAN DIEGO STATE UNIVERSITY

The Undersigned Faculty Committee Approves the
Thesis of Arpitha Rao Gadhath:

Battle of France WWII

Carl Eckberg, Chair
Department of Computer Science

Tao Xie
Department of Computer Science

J. Carmelo Interlando
Department of Mathematics and Statistics

Nov 12, 2014
Approval Date
Copyright © 2014
by
Arpitha Rao Gadhath
All Rights Reserved
DEDICATION

It is with my sincere gratitude, I dedicate my thesis to Prof Dr. Carl Eckberg, who has been a constant source of knowledge. I also dedicate this thesis to my mother Anuradha and father Anantha Rao for making me believe in myself and pushing me forward every step of my life. I also want to thank my sister Akshatha, brother-in-law Manoj and my husband Ashwin for their moral support.
ABSTRACT OF THE THESIS

Battle of France WWII
by
Arpitha Rao Gadhath
Master of Science in Computer Science
San Diego State University, 2014

The purpose of this thesis is to build an interactive Geographical Information System (GIS) tool, relating to the series of events that occurred during the Battle of France World War II. The tool gives us an insight about the countries involved in the battle, their allies and their strategies.

This tool was created to use it as a one stop source of information regarding all the important battles that took place, which lead to the fall of France. The tool brings together the maps of all the countries involved. Integrated with each map is the data relevant to that map. The data for each country includes the place of attack, the strategies used during the attack, and the kind of warfare. The tool also makes use of HTML files to give all the information, along with the images from the time of the war and a footage which explains everything about the particular battle.

The tool was build using JAVA, along with the use of MOJO (Map Objects Java Objects) to develop Maps of each of the countries. MOJO is developed by ESRI (Environmental Science Research Institute) which makes it easier to add data to the maps. It also makes highlighting important information easier making use of pop-up windows, charts and infographics. HTML files were designed making use of the open-source template developed by Bootstrap.

The tool is built in such a way that the interface is simple and easy for the user to use and understand.
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>ABSTRACT ................................................................. v</td>
</tr>
<tr>
<td>LIST OF FIGURES .......................................................... vii</td>
</tr>
<tr>
<td>ACKNOWLEDGEMENTS ............................................................ ix</td>
</tr>
</tbody>
</table>

## CHAPTER

1 INTRODUCTION ........................................................................... 1
   1.1 Overview ........................................................................... 1
   1.2 About the Application ...................................................... 2

2 SPECIFICATIONS ....................................................................... 3

3 TECHNOLOGIES ......................................................................... 5
   3.1 Java ................................................................................. 5
   3.2 MapObjects - Java Edition ............................................... 6
   3.3 Eclipse IDE ....................................................................... 7
   3.4 HTML5 ........................................................................... 7
   3.5 DBF Explorer ................................................................. 8
   3.6 CDBF-DBF Viewer and Editor ......................................... 8

4 THE FALL OF FRANCE ............................................................... 9
   4.1 Overview ........................................................................... 9
   4.2 Fall Weiss ................................................................. 9
   4.3 Fall Gelb (Luxembourg) ............................................... 10
   4.4 Fall Gelb (Netherlands) ............................................... 11
   4.5 Fall Gelb (Belgium) ........................................................... 12
   4.6 Fall of France ................................................................. 13

5 SOFTWARE PROTOTYPE ............................................................ 14

6 ARCHITECTURAL DESIGN ......................................................... 17

7 SETUP DEVELOPER’S ENVIRONMENT ..................................... 19

8 LAYERS AND SHAPEFILES .................................................... 21
   8.1 Projecting Layers ............................................................. 21
8.2 How to Create Shapefiles ................................................................. 21
8.3 Editing Records of Shapefiles .......................................................... 23
8.4 Table Records and Fields ............................................................... 23
9  TOOLBARS AND THEIR SIGNIFICANCE ........................................ 24
   9.1 ZoomPanToolBar ................................................................. 24
   9.2 Selection Toolbar ................................................................. 25
   9.3 Custom Toolbar ................................................................. 25
   9.4 Hotlink .................................................................................. 26
10 CUSTOM MENUS AND THEIR FUNCTIONS .................................... 31
   10.1 File Menu ........................................................................ 31
   10.2 Battles ............................................................................. 33
   10.3 Resistance ........................................................................ 33
   10.4 Read Me ........................................................................... 33
   10.5 Help .................................................................................. 35
11 CONCLUSION AND CHALLENGES .............................................. 37
12 FUTURE ENHANCEMENTS ............................................................. 38
REFERENCES ........................................................................... 39
LIST OF FIGURES

PAGE

Figure 5.1. Initial page load. .................................................................15
Figure 5.2. Custom menu bar. .................................................................15
Figure 5.3. Prototype of the application ..................................................16
Figure 6.1. Software architecture diagram ...........................................17
Figure 7.1. Adding ESRI library .............................................................19
Figure 7.2. Importing MOJO jar files to Eclipse ......................................20
Figure 8.1. Creating shape file by selecting features. (Theme->create layer from selection) .................................................................22
Figure 8.2. Enter the name of the newly created shape file .....................22
Figure 9.1. Zoom pan toolbar. ...............................................................24
Figure 9.2. Selection toolbar .................................................................25
Figure 9.3. Custom toolbar .................................................................26
Figure 9.4. Clicking on XY tool to browse to a CSV file .......................26
Figure 9.5. Clicking on open would load the shape file .......................27
Figure 9.6. HTML file showing the content related to the battle ..........27
Figure 9.7. HTML File showing the images ...........................................28
Figure 9.8. HTML File showing the video ..............................................28
Figure 9.9. Hotlink tool displaying image gallery when clicked on the point on a map ....29
Figure 9.10. Snippet implementation of Hotlink tool ..........................30
Figure 10.1. Custom menus available ....................................................31
Figure 10.2. Selecting the legend editor ............................................32
Figure 10.3. Using the options from the legend editor ........................32
Figure 10.4. Selecting the Battles menu ............................................33
Figure 10.5. After selecting Luxembourg ............................................34
Figure 10.6. Showing the Resistance menu ...........................................34
Figure 10.7. Displaying the Read Me menu ...........................................35
Figure 10.8. Help menu with the About the Tool option ......................36
ACKNOWLEDGEMENTS

I would like to take this opportunity to thank my advisor and Thesis chairperson Professor Dr. Carl Eckberg, San Diego State University for giving me this opportunity to work on this Thesis. You have been a tremendous mentor for me, encouraging my research and also advising me. I would also like to thank Professor Dr. Tao Xie, San Diego State University and Professor Dr. J. Carmelo Interlando, San Diego State University for serving as my committee members.
CHAPTER 1

INTRODUCTION

1.1 OVERVIEW

The Battle of France, also known as the Fall of France took place during the Second World War between Germany and France. After the First World War, the French which had a lot of casualties, constructed a defensive line along the German border called The Maginot Line to protect themselves from a future German attack. It all started in September 1939 when the Germans attacked and conquered Poland resulting in the French and the British declaring a war against Germany. Even 9 months after declaring a war against Germany the French and The British had done very little to threaten the Germans. This phase of anticipation was known as the “Phoney War”. It all changed on the 10th of May when the Germans decided to attack the Lower Countries leading to the invasion of France. The intention behind developing this tool was to make it an easy interactive method for history enthusiasts to use it to learn more about the series of events that occurred during the Fall of France. The tool holds all the information regarding the strategies they used during these attacks and the kind of warfare which was entirely new and experimented for the very first time.

The GIS (Geographic Information Systems) tool was built using JAVA, along with the use of MOJO (Map Objects Java Objects) to develop maps of each of the countries. MOJO is developed by ESRI (Environmental Science Research Institute) which makes it easier to add data to the maps. It also makes highlighting important information easier making use of pop-up windows, charts and infographics. HTML files are used to give more information regarding each of the events. It also contains a series of images which reminds people about the wrath that took place during these invasions. And to give the user a more in-depth knowledge there is a video embedded at the end of the HTML file. There is also a feature in the tool which makes use of the Google image gallery feature to give the user a glimpse of what all the places look like.
Chapter 2 talks about the requirements for the tool to work in detail. Chapter 3 tells us in detail about the technology used to develop this application, like Java, MOJO, developed by ESRI, HTML (Hyper Text Markup Language), Image sliders, DBF Explorer, CDBF-viewer editor. Chapter 4 talks more about the topic of the thesis the Fall of France. It tells us how the tool makes it easy for the user to explore this piece of history and learn about all the features of the tool which are created explicitly for the topic. Chapter 5 is about the prototype of the software. Chapter 6 is about the architectural design and main components used during the development of this tool. Chapter 7 tells us about the environment that needs to be set to make the tool work. Chapter 8 explains the shape files and layers used in the tool to represent the countries and the places where the events took place. Chapter 9 tells us about the toolbars available and their usage in the application. Chapter 10 tell us about the custom menus made available to the user and their utility in learning about the battles which took place during that time. The Help button which gives more insight in to the usage of the tool. Chapter 11 and 12 talk about the challenges faced during the course of tool development and future enhancements respectively.

1.2 ABOUT THE APPLICATION

This is developed as an informative tool, making use of the JAVA programing language hand in hand with our own custom GIS map developed making use of JAVA Swings and MOJO (Map Objects Java Objects), which is an ESRI (Environmental Science Research Institute) product. The advantage of using MOJO is that it lets you creates map layers dynamically and display them. It also lets you make use of some of the customized features such as zooming in and out of the map, adding new map objects, adding new features to the map objects etc. The Eclipse IDE used for developing this tool is 4.4.1, it also has downward compatibility up to 3.6.1. Eclipse IDE creates a flexible environment to create applications. It becomes a lot easier to customize the tool to perform user defined functions. The HTML files which are created making use of bootstrap template are linked to points on each layer to provide more information to the user regarding the events. The HTML file can also hold images of all the formats such as JPG, JPEG, PNG etc. which are also supported by MOJO. The application also contains a video embedded in each of the HTML files. The working of the application and the tools used are explained in detail in the later chapters.
CHAPTER 2

SPECIFICATIONS

The thesis features an interactive application tool which attracts the attention of the people who want to know more about the historic battles that took place during World War II. The tool concentrates on giving the user information about the series of battles that took place from Sep 1939 through June 1940 and also the countries involved such as France, Germany, Netherlands, Poland and Belgium. It also gives user information regarding the places where the attacks took place along with other useful information. The tool also lets the user witness these events that took place, by making use of images and videos from the past. For the tool to work effectively there are a few specifications below and a list of requirements:

- The tool should hold information regarding all the countries those were involved or led to the Fall of France.
- The tool should also provide information regarding the main events which redefined the course of the battle.
- A Hotlink tool must be made use of to provide information regarding points on each of the map layers.
- The tool should have the capabilities of adding or removing a layer as per the user’s convenience.
- A print tool to help the user print some useful information provided by the tool for future reference.
- The tool must be robust and simple for any user to navigate through out the application without having to go through any hustle.
- A web tool must be used to provide more information about the battle.
- The tool must also contain collective information in the form of pictures and videos to let the user know more about the incidents that took place.
- It must also have all the necessary tools required by the user to interact with the application and to create a flawless experience for the user.
- The tool should have a simple user interface aimed at providing more knowledge to the user.
• The tool is to provide sufficient information about the fall of France and how the Germans attacked the lower countries to get to France.
• The tool should also provide information regarding the political and military strategies used during the time of the invasion.
• JAVA with MOJO (Map Objects Java Objects) was recommended for development of this tool, since they are platform independent.

The motivation behind developing this tool was the interest I had to know more about the battles and wars that took place between some of the most powerful countries in the world. The intended audience for this are the historians looking to know more about the fall of France without having to go through multiple books or places looking for the information required and also high school and college students for whom the information from this tool will help know more about those events in history.
CHAPTER 3

TECHNOLOGIES

As the name implies this chapter deals with the platform and technologies used in developing this tool. It describes the advantages of using the respective technologies in developing the application. The list of technologies used are JAVA Swing, MOJO, HTML5 and a DBF editor. JAVA Swing is a very powerful tool used to develop the Graphical User Interface (GUI). Java is platform independent, which is also one of the main reasons for using it in developing the application. A more detailed explanation regarding all the technologies used are mentioned below.

3.1 JAVA

It is a robust programing language used to develop application software and a cross-platform computing environment. It is being used in a variety of computer platforms ranging from phones to super computers. To run programs written in the JAVA language we make use of a cluster of related programs known as the JAVA Platform. The platform is not specific to any operating system nor a specific hardware but to a virtual machine and a bunch of libraries. JAVA programs run on virtually any hardware. The platform consists of many programs, each one carrying out a different functionality to fulfill its overall capability. The most essential components are the JAVA language compiler, the libraries and the runtime environment. The JAVA Virtual Machine (JVM) converts all the code/instructions in java into API JAVA byte code programs, which are integrated. This byte code remains the same on all the hardware, with any kind of operating system running on it. JVM has in it a Just in Time (JIT) compiler, which converts these byte codes in to the native language, if requested [1].

Use of this JIT makes it execute a little slower than native language, but this is true of any higher-level language. However, a set of well-tuned interpreters, smart compilers, and just-in-time byte code compilers can bring its performance closer to that of native code. Java
is a powerful software platform and a general-purpose, high-level programming language. Using the Java API, we can write many types of programs.

In this thesis we make use of Java Swing, which is a primary Java GUI widget toolkit. Swing provides a more redefined set of GUI Components when compared to Abstract Window Toolkit (AWT) [2]. Some of the more advanced components that we get with Swing are scroll panes, tabbed panel, trees, tables when compared to AWT. Also Swing is platform independent as it is written entirely in Java when compared to AWT, which is platform dependent and hence the name “lightweight”. Some of the important advantages of using Java Swing are mentioned below [1]:

- **Platform Independent**: As the name suggests it is one of the biggest advantage of using Swing, where you do not have to bother about the underlying operating system.

- **Ease to use**: A document with the complete set of classes used in swing is available which makes it easier for the developer to understand the functionality of all the components available to use.

- **Extensible**: Making use of the inheritance property of Java a user can create his own custom implementation of the components which are available.

- **Customizable**: The appearance of a swing component can also be modified by the user as per convenience. The appearance of the components can be programmatically changed to give the UI a completely unique appearance.

- **Configurable**: Using Swing makes it possible to respond to fundamental changes in its setting at runtime, which means a Swing application is capable of hot swapping its user interface at runtime.

- **Consistent look and feel**: As swing is less dependent on the operating system, widgets, swing varies less than AWT on different operating system.

These are some of the advantages of using Swing in our project.

### 3.2 MapObjects - Java Edition

Map Objects Java Objects also known, as MOJO is a product of ESRI. It is a set of mapping software components, which can be integrated into an application. Map Objects can also be combined with other objects such as database objects, multimedia objects and graphing objects [3]. It has the ability to combine these multiple objects to create a custom map, which can be used to develop GIS applications [4]. Many custom cross platform applications can be developed by making use of these mapping components. The applications can be built in an interactive way where the user can query to obtain the information required. MOJO can be described as component architecture similar to visual basic.
Mentioned below are some of the functions we can make use of while developing an application using MapObjects:

- Display map layers with roads and boundaries.
- Zoom in and out of a map.
- Add features on a map such as points, lines or polygons.
- Create new shape files.
- Find locations on the map using coordinates or address.
- Label the features.
- Have an attribute table for all the features and also query those to update or fetch information from the attribute table.
- Click on the features to identify them.
- Add descriptive text.

These are some of the functions available to develop the application. There are many more of these functions using which you can make the application more intuitive.

### 3.3 Eclipse IDE

Eclipse is an Integrated Development Environment (IDE) used to develop applications in programming languages such as Java, C, C++, Java Script etc., and in our case we use it to develop the tool in Java. It creates a workspace to store projects and also supports other plugins to be imported to integrate into the project. We can also import external libraries to use in the project, for example in our thesis we make use of MOJO jar files or library files using which we can use all the functionalities of Map Objects in our program. It also helps the developer debug the code making use of break points and a debugger. This save a huge amount of time for the developer to find issues in the code while writing it and develop an application without any errors. Eclipse has a faster Java Virtual Machine and it also give the user autosuggestions. Hence, eclipse is one of the better choices for developers to use for building an application. Another is Netbeans.

### 3.4 HTML5

Hyper Text Markup Language (HTML) is a markup language that can be used to describe the content of the web. HTML5 is the 5\textsuperscript{th} major version of HTML. Some of the major changes or advantages using HTML5 is that we can integrate images, video, audio,
graphics and interactive documents as a part of the content. It comes with new tags such as `<source>` to define that it’s a media content, `<article>` to define it as an article, `<nav>` to specify navigation within the sections of your page. These are all the stylistic features, which can be made use of. The look and style of the content written in the HTML page can be customized using Cascading Style Sheets, which is a style sheet language. We can modify the font, color and appearance of the content making use of CSS. Java Scripts are also used in the HTML files to add dynamic functionality. In our thesis we are making use of all three (HTML, CSS and Java Script) to represent the content related to the topic.

3.5 DBF EXPLORER

DBF Explorer is an application used to view the database files. Database files can also be edited or modified by making use of this application. This application can be used to perform some of the simple operations such as adding a record, deleting a record, adding rows and columns, creating new DBF files, modifying a structure of the field etc. we can also export the data in a text or a HTML file making use of this application. In our tool we make use of DBF Explorer to create, edit or view a DBF file related to a shape file. One of the best advantages of using this tool is that it does not need a database connection to perform all the operations mentioned above and the application by itself is available for free in the market.

3.6 CDBF-DBF VIEWER AND EDITOR

CDBF-DBF viewer and editor provides a graphical user interface for the user to open, view, edit and modify a DBF file without have to write any code to perform the operations he/she intends to. The built-in designer lets the user generate reports making use of the information available in the DBF files. The product was a trial version and is not available for free.
CHAPTER 4

THE FALL OF FRANCE

4.1 OVERVIEW

It all started in September of 1939 when Germany decided to invade Poland called Fall Weiss. This led to France and its ally Britan to declare a war against Germany. After which there was no imminent threat by any of the countries, which was named the Phoney War. All this ended when the Germans launched the Invasion of France and the lower countries namely Luxemburg, Belgium and the Netherlands. The Germans were able to get past the lower countries with ease as they did not pose any opposition. Then the Germans took 45 days to push the French and British soldiers into the sea, capturing the land of the French.

4.2 FALL WEISS

Fall Weiss also known as the Invasion of Poland, began on September 1st 1939, and lasted for over a month and five days, and by the end of it the Germans and the Soviet Union conquered and divided it among themselves. The initial strategy Adolf Hitler had when he came to power in 1933 was to create puppet states around Germany, which had no economy of their own, and were entirely dependent on Germany for everything. He made some changes in the foreign policies in such a way that it would improve the relationship between Poland and Germany and also weaken the ties between Poland and France. Meanwhile he also tried to persuade Poland in to an Anti-Comintern pact, forming a cooperative opposition against the Soviet Union [5]. He told Poland that if they agreed to wage war against the Soviet Union, they would be given free land in the North-East of Ukraine and Belarus. Poland knew that agreeing to this meant that Poland would be largely dependent on Germany, they felt that it would eventually threaten their independence and decided not to support Germany in a war against the Soviet Union.

This decision taken by Poland did not go down well with the Germans, they provoked a war against Poland using the City of Danzig and the Polish Corridor as a reason [5]. They stated that the Polish had attacked those Germans, who were a majority in the free city of
Danzig and also a minority population in the Polish Corridor. The Germans demand for the City of Danzig had increased by 1937 and they proposed to construct a road to connect East Prussia with Germany. Poland rejected its request, as it did not trust Germany [5]. Poland knew that the Germans were making pacts with the Anti-Polish Ukraine Nationalist, to isolate them. The British and the French were observing this situation and decided that they would help Poland defend its independence, which meant that they would go against the Germans in case of an attack. The British still hoped that there would be some sort of a peace talk to resolve the issue. The talks between the Germans and the Polish broke down, and the Germans realized that the British and the French had failed to gain support of Soviet Union to wage war against the Germans, and also that the Soviet Union was interested in making a deal with the Germans to attack Poland. After getting to know this the Germans made a pact with the Soviet Union that if they were to be a part of Case White (Invasion of Poland) then they would be given 2/3rd of the country after defeating Poland and the Germans would keep the rest of 1/3rd [5]. The Germans decided to attack Poland from all the sides and finally converge at Warsaw. The British and the French forces were able to do very little to stop the Germans. On 6th of October the Polish finally surrendered and Poland was completely invaded by Germany and Soviet Union. This marked the beginning of World War II and the British and French declared that they would be going against the Germans.

4.3 FALL GELB (LUXEMBOURG)

After the British and the French decided to wage war against Germany when they attacked Poland, there was a lack of military operations from both ends. This was phrased The Phoney War during the early phase of World War II. It all changed when Germany decided to invade France and its lower countries in the May of 1940 [6]. The Lower Countries included Luxembourg, Netherlands and Belgium. The reason for the Germans to attack the Lower countries was to make it easier for them to attack France from all sides. The attack on the lower countries was codenamed Case Yellow by the Germans. Fall means ‘plan’ or ‘case’ and Gelb means ‘yellow’ [7].

Luxemburg was a small independent country sandwiched between France and Germany. When the tension between the two neighboring countries grew, the government of Luxemburg was in a fix; it did not want to be a part of the war between the two countries and
decided to stand by the neutrality policy it signed during the Treaty of London in 1867 [8]. On noticing an increase in the military activity in the German border on the 9th of May 1940, Luxembourg decided to block the border using concrete and steel barricades of the Schuster line. The following morning at 4:35, Germans began the invasion of Luxembourg [8]. Germans did not face any kind of resistance. The capital city was under German control by noon.

4.4 FALL GELB (NETHERLANDS)

Netherlands was a part of the operation Fall Gelb. After the invasion of Poland, when the French and British decided to go to war against Hitler, Winston Churchill warned the lower countries of an attack by the Germans [9]. The Netherlands like the rest of the lower countries did not take it seriously, it rather decided to be neutral as it did during the First World War. It thought of Germany a strategic trade partner and did not want to destroy the trade through war. Germany in its intention to protect the Ruhr area decided to invade the lower countries.

The Dutch did not have a strong army; their military troops were undertrained and were not ready to face the German army. All their military equipment was ordered from Germany and was obviously delayed [9]. When the tension grew between the neighboring countries, the Dutch army sealed all the border gates leading to Germany. The attack came as a surprise to the Dutch; on the early hours of 10th May 1940, the Dutch army saw the German aircraft from Luftwaffe violating the Dutch airspace. Initially the Dutch though that the aircrafts were heading towards UK and was not an attack on the Dutch airbase, but it turned out that the aircraft came back to attack the Dutch airfields.

The Germans strategy was to attack and conquer all the airfields held by the Dutch making use of paratroopers. This strategy was used for the very first time in any war. The idea was to attach and occupy the airfields even before the ground troops made it to the area. The attack on The Hague seemed to be a failure at the beginning and at the same time French started to gather their troops near the border of Netherlands. The Dutch couldn’t hold back the German troops; the German Luftwaffe aircraft heavily bombarded their cities. There were heavy civilian casualties when the Germans bombarded Rotterdam [10]. The threat to
bombard more cities by the Germans brought an end to the battle and the Dutch decided to surrender. It took just 6 days for the Germans to conquer the Dutch.

Netherlands was under German control until 1945. The brutality of Germans can be witnessed by reading the diary of Anne Frank, one of the Jewish victims of the Holocaust.

4.5 **Fall Gelb (Belgium)**

Battle of Belgium was a part of case yellow, where the Germans decided to attack the lower countries to get to France. The ruler of Belgium at that point, King Leopard did, try to mediate between Germans and the western allies before the war and after the war broke out in 1939. Even though the Belgians had fought against Germany in the First World War, King Leopard decided to be neutral and not to support any allies. He knew that supporting the allies in any means would bring the wrath of the Germans upon them. The King did not let any of the allied armed forces to get in to the Belgian border.

A German aircraft carrying all the documents regarding the attack was bought down in Belgium, and this came as a prior warning to the Belgians. As soon as the King learnt about the attack he decided to take the help of the allies. The British and the French knew that the attack on Belgium by the Germans needed to be stopped as they knew Germany would use Belgium as a way to attack the French more aggressively, hence they sent the best available allied military force to the Belgium borders to help them hold back the Germans [11].

Fort Eban-Emael toward the German border was a large underground fort, which was guarded by close to 1200 Belgian soldiers. The Germans were able to capture this fort which helped them bring in more troops and secure 3 important bridges. They held the K-W Line, which was guarded by both the Belgian and The French forces, for 3 days, after which with the help of Luftwaffe the Germans broke though the Ardennes Forest. This came as a shock to the French and they started to retreat. The Belgian soldiers then gave up the K-W Line. The British forces BEF (British Expeditionary Force) was sent to the north when they were not ready for the Blitzkrieg in the west by then the Germans had cut off the Belgians and King Leopard.Shortly after that the Belgian King decided to surrender the armed forces, which left the British perplexed. They had to withdraw forces from the Belgian port of
Dunkirk. The Battle of Belgium also included the biggest tank battle till that time called the Battle of Hannut. It took Germany 18 days to conquer Belgium [11].

4.6 FALL OF FRANCE

After the French suffered huge casualties in the First World War they decided to build a barricade along the German border called the Maginot Line. The Germans after the Phoney War decided to attack the French and the lower countries code named Fall Gelb. The Germans came heavy on the Maginot Line and broke in to the French border through Ardennes and then the Somme valley. By doing this the Germans were able to block the allied forces who had entered Belgium from coming back in [12]. During the Dunkirk operation The British and the allied forces were pushed in to the sea and then Winston Churchill ordered an evacuation saving 350,000 soldiers.

The Germans then initiated Fall Rot or case Red. After the allied forces were pushed back the French did still show resistance, but the German air strike caused French a lot of casualties. The Germans cut through deep in to the Maginot Line, attacking the south of France. The Germans then attacked Paris, which was completely unguarded. Germany was able to get Paris under their control in the next three days. Finally the French forces surrendered, it made an agreement where Germany would occupy the North and the West of France. Italy would get a small southeastern Italian occupation zone and a Vichy Government was formed in the unoccupied Zone [13].
CHAPTER 5

SOFTWARE PROTOTYPE

Software Prototyping is one of the important phases of the software development cycle, where the developer creates an initial working version of the application to check its functionalities. The advantage of developing a software prototype is that the software designer and developer can get an early feedback from the user about the application. This would let the client or the end user see if the software specifications are getting fulfilled and if there is a change in the early stages of development it becomes a lot easier to implement. Developing an application till the end and then finding an inconsistency in the design and functionality would lead to higher cost of application development and also an increase in the time frame for the final version.

Agile Software Development was used while developing this tool for my thesis. Software Prototyping is a part of this development process, where I created initial draft version of my application to be approved. This phase continued till the end where a final version of the completely developed application was developed. The prototype of my application had some of the basic and important functionalities that were being built into the tool. Figure 5.1, 5.2 and 5.3 are the screen shots of how my prototype looked like.

- When the project is loaded initially, it shows a shape file with all the countries involved in The Battle of France as shown in Figure 5.1.
- Custom Menu bar, which had a functionality of loading different layers as shown in Figure 5.2.
- Custom Tool bar to link a HTML page to a map layer as shown in Figure 5.3.
Figure 5.1. Initial page load.

Figure 5.2. Custom menu bar.
Figure 5.3. Prototype of the application.
CHAPTER 6
ARCHITECTURAL DESIGN

Software Architecture is a high-level design of the application. It is used to describe how the flow of control needs to be within the application. It can be used as a blue print to identify the functionalities for each of the component within the tool. The flow diagram in Figure 6.1 shows an architectural design of the GIS based application developed as a part of this Thesis:

The architecture of the application is as follows [14]:

- When the application is loaded, it comes up with a map layer which includes all the countries that were involved in the battle. It also has all the built in and custom made tool bars and menu items.

- The UI comprises of four different parts, the left panel, right panel, north panel and the south. The left panel consists of a TOC where all the layers loaded will be listed.

![Image of software architecture diagram.](image-url)
The north panel includes all the custom and build in toolbars and menu items. The right panel has not been made use of in our application and the south panel is used to represent the coordinates in the map layer.

- The addition of custom tool bars and menu items were easier to implement, as Java Swing was used to develop the user interface.
- Shape files were used to represent the map. MOJO made it easier to create these shape files based on the requirement and each shape file was integrated with a corresponding DBF file.
- Html files were created to give a detailed description to the user. Embedded with each HTM file were a set of images and video.
CHAPTER 7

SETUP DEVELOPER’S ENVIRONMENT

As we already know Eclipse IDE let us import libraries to be used in our project. In our thesis we import two of the ESRI MOJO libraries, which let us use some of the inbuilt components in the project. The screen shots below show in steps how to import these jar file or library files in to the project. Figure 7.1 and 7.2 shows how to import/select the file.

Figure 7.1. Adding ESRI library.
Figure 7.2. Importing MOJO jar files to Eclipse.
CHAPTER 8

LAYERS AND SHAPEFILES

8.1 PROJECTING LAYERS

Map Objects Java Objects (MOJO) comes with a predefined set of functionalities that can be used to create shape files from an existing map layer. This can also be listed on the TOC making use of the function addShapeFileToMap() [15]. A new shape file can also be created making use of some of the tools available. The tools used to create the shape files are ‘select feature layer’ and once the selection is done we the ‘create layer form selection’ tool to create the shape file. Shape files for all the countries used in the thesis were created similarly.

ArcView is a GIS Software developed by ESRI, which can be used to create a polygon, line or a point shape file and project it on the map. Once the .shp file is generated the ‘add layer’ tool is used to project that shape file on the map. When projected on the map a new layer also includes the information given in the DBF file along with the corresponding shape file.

8.2 HOW TO CREATE SHAPEFILES

As mentioned earlier MOJO provides some of the tools that can be used to create a shape file, where the user can select he section for which he want to create a new layer and use ‘Create Layers From Selection’ to create the new shape file which only has the selected region. The following steps tells us how to create a shape file form the existing map:

- Select the map listed on the TOC from which you want to create a shape file.
- After selecting the map from the TOC, the tool ‘Select Feature’ gets enabled on the tool bar. When you click on the tool it gives user the option of what kind of shape file he/she would like to create (rectangle, circle, point or a polygon).
- Let’s say that the user selection the polygon option. Then the user can draw a polygon on the area that he wants to create a shape file for.
- Once that is done the user can go to the ‘Theme’ menu and select ‘Create Layer from Selection’ to create a new shape file as shown in Figure 8.1.
- When that is done the user will be prompted with a window to select the name of the shape file to be stored at as shown in Figure 8.2.
Figure 8.1. Creating shape file by selecting features. (Theme->create layer from selection).

Figure 8.2. Enter the name of the newly created shape file.
8.3 Editing Records of Shapefiles

When a shape file is created, an associated DBF file will also be generated as a part of the shape file with a .dbf extension. CDBF – Viewer and Editor or a DBF Editor can be used to view or edit the DBF file associated with the shape file. All DBF files are database files, using a CDBF – Viewer and Editor or a DBF Editor will let the user edit these files without the need of having an actual database connection.

8.4 Table Records and Fields

Each .dbf file will have a table with a specified set of rows and columns in them holding all the data entered by the user, which is associated with the corresponding shape file. The column and the rows in the table can be modified as per the user’s requirement. It is fairly simple to add or remove an existing column or row from a table making use of CDBF – Viewer and Editor or a DBF Editor.
CHAPTER 9

TOOLBARS AND THEIR SIGNIFICANCE

The Tool bars used in the application enhances the way user gets to interact with the tool. The Tool bars can be customized to add new features to the application. We also make use of some of the standard tool bars provided by MOJO in this application. ‘ZoomPan ToolBar’ and ‘Selection ToolBar’ are the two tool bars we are making use of.

9.1 ZOOMPANTOOLBAR

Some of the common functionalities required in an application are provided by the ZoomPan ToolBar, such as zooming in, zooming out, zoom to fill extent etc. Figure 9.1 shows how the ZoomPan ToolBar looks like in the application.

![Zoom Pan Toolbar](image)

Figure 9.1. Zoom pan toolbar.

The list below contains the functionalities of all the available options in the ZoomPan ToolBar, which is an ESRI component:

- **Zoom In**: This can be used to zoom in to the pointed location on the map.
- **Zoom Out**: This is used to zoom out of the map.
- **ZoomToActiveLayer**: If you want to zoom in to the feature selected on the map that is active that point we can make use of this.
- **ZoomToFullExtent**: To zoom the map to an extent which covers all the layers on the map within its extent, this tool can be used.
- **Next Extent**: This tool is used to zoom into the next available extent.
- **Previous Extent**: This is used to zoom into the previous extent stored.
- **Pan**: This tool is used based on the movement of the mouse to pan the map.
- **Pan One Direction**: To pan the map only in one direction this tool can be used.
- **Identify**: To identify a feature on the selected map this tool can be used.
9.2 Selection Toolbar

Selection Toolbar is another ESRI tool bar used in the application. This toolbar is enabled for the user to use only when a layer is selected in the TOC. Once the layer is selected we can use these available tools to make some selections on the map. This toolbar comes in handy when we want to create a new shape file or when we want to view the attribute table in a map as shown in Figure 9.2.

![Figure 9.2. Selection toolbar.]

The list of tools in selection toolbar as mentioned below:

- **Find tool**: Find tool is used to find the feature mentioned.
- **Search tool**: Search tool is used to find the features using a Query.
- **Query Builder tool**: Query Builder tool is used to select a features on the using a query result.
- **Select Feature tool**: Select Feature tool is used to select feature and create a layer form the existing parts of the map.
- **Clear Selection tool**: Clear Selection tool is clear all the selections currently made.
- **Buffer**: Buffer tool is used to develop a buffer polygon on the selected feature.
- **Attribute Table**: Attribute Table is used to view the data stored in .dbf file in the form of tables [14].

9.3 Custom Toolbar

This user created toolbar is used to add a few extra features in the application that makes it more intuitive. Some of the features provided by this tool bar to the user includes adding a new layer, place a point on the map, open an Html file related to the layer etc as shown in Figure 9.3. Each of the tools functionality is as mentioned below:

- **Point tool**: placing a point on the map can be done using this tool.
- **XY tool**: This is the tool user can use to create a new layer. The tool uses a comma separated file (CSV) as an input. The user can store in the CSV the latitude and longitude for each of the feature he wants to place in the file followed by a name associated with that feature. Following steps needs to be followed to create a new layer:
1. Click on the XY tool, which opens a browser window asking the user to select the CSV file he wants to load as shown in Figure 9.4.

2. Once the user hits OK, the corresponding coordinates appear on the map as a dot as shown in Figure 9.5. The true type map symbol (dot) can be changed to any icon of font and selecting some icon therein other fonts available, by changing the value of the true type.

- **Web Tool**: This tool pops up the HTML file related to the shape file that has been loaded. It has all the information, images and a video related to the layer. Figure 9.6, 9.7 and 9.8 shows them respectively.

- **Add layer**: This can be used only to add a .shp shape file.

- **Arrow tool**: After the use of different standard and custom tools, the cursor needs to get back its default functionality which can be achieved using this tool.

**Figure 9.3. Custom toolbar.**

**Figure 9.4. Clicking on XY tool to browse to a CSV file.**

### 9.4 Hotlink

Hotlink is one of the custom tool integrated in the application. The tool, when you select it and click on one of the points or a feature on the active layer, a google image gallery
Figure 9.5. Clicking on open would load the shape file.

Figure 9.6. HTML file showing the content related to the battle.
Figure 9.7. HTML File showing the images.

Figure 9.8. HTML File showing the video.
for that particular point pops up. It has a lightning bolt icon in the toolbar section. In the current application this tool can be used as represented in Figure 9.9:

![Figure 9.9. Hotlink tool displaying image gallery when clicked on the point on a map.](image)

In Figure 9.10 we show some of the code needed to use the hotlink tool feature for polygons. The code to identify which polygon a user is clicking inside is very complex. Fortunately ESRI has solved this problem for us. They use an identify class and a PickListener for this purpose.
**class** MyPickAdapter **implements** PickListener
{
   //implements hotlink
   **public void** beginPick(PickEvent pe){System.out.println("begin pick");}
   // this fires even when you click outside the states layer
   **public void** endPick(PickEvent pe){}
   **public void** foundData(PickEvent pe)
   {
      //fires only when a layer feature is clicked
      selectedShapefileLayer = (FeatureLayer)pe.getLayer();

      //FeatureLayer flayer2 = (FeatureLayer)
      pe.getLayer();

      com.esri.mo2.data.feat.Cursor c = pe.getCursor();
      Feature f = null;
      Fields fields = null;

      mystate="NA";

      if (c != null)
         f = (Feature)c.next();
      fields = f.getFields();
      mystate = (String)f.getValue(1);
      System.out.println("Selected Country: "+mystate);
      System.out.println("Selected Layer: "+selectedShapefileLayer.getName());

      try {
         HotPick hotpick = new HotPick(mystate,
            selectedShapefileLayer); //opens dialog window
         //hotpick.addContentToJDialog(Location11,Date,Rank); //
         //opens dialog window with Duke in it
      } **catch** (Exception e){
         e.printStackTrace();
      }
   }
}

**Figure 9.10. Snippet implementation of Hotlink tool.**
CHAPTER 10

CUSTOM MENUS AND THEIR FUNCTIONS

This chapter deals with the custom menu buttons available to the user and their functionalities. In the application there are about 6 menu buttons such as File, Theme, Battles, Resistance, Help and Read Me. Each of them have their own custom functionalities. We shall be going through each one of them in detail in this chapter. Figure 10.1 shows all the custom menus available.

![Custom Menus](image)

Figure 10.1. Custom menus available.

10.1 **FILE MENU**

The File menu has three options, those are Add Layer, Remove Layer and Legend Editor. Let’s go through the functionality of each one of them:

- **Add Layer:** When the use clicks on the Add Layer button, a window pops up asking the user to select the .shp file. After selecting the file ok button needs to be clicked, then that shape file will be loaded on top of the existing map layer.

- **Remove Layer:** Using this option the user can delete any available layer from the list of layers present in the TOC. Before the user clicks the remove layer button he need to select a layer that he intends to delete.

- **Legend Editor:** Legend editor lets the user change the color and their properties of the existing map. This lets the user read different properties of each layer represented in a different color. The Legend Editor window has three tabs namely symbol, labels and general. The attributes of a layer can be analyzed visually by rending the colors using graduated symbols. If we represent the attribute values on the map using Legend Editor, analyzing the statistical data becomes a lot easier. Using the Labels tab we can label the shape file where ever necessary, the font and color of the label can also be changed making use of the Label tab. The name of the shape file can also be changed using Legend Editor. Figure 10.2 and 10.3 represent the way Legend Editor can be used to label and change the color of the shape files.
Figure 10.2. Selecting the legend editor.

Figure 10.3. Using the options from the legend editor.
10.2 Battles

The Battle menu button is used in the application to load all layers related to each of the battles. At any point of time any of the battles can be loaded and it would load the layers related to that battle and bring it to the foreground. For example if the Luxembourger shape file is loaded currently and you want to look at the layers related to France then click on the Battles menu and select Fall of France. Figure 10.4 and 10.5 shows the working of this menu.

![Figure 10.4. Selecting the Battles menu.](image)

10.3 Resistance

This menu has four one option, one for each of the countries involved in the Battle of France. Poland, Luxembourg, Netherlands, Belgium and France. When you click on any of the options available the user gets to see a window with information regarding all the places those countries offered some resistance against the German attack. Figure 10.6 shows the working of these menu options.

10.4 Read Me

Read Me menu has only one option to select called Get Started. This option is used as a way to give user an insight in to the working of this application. It is a brief description
Figure 10.5. After selecting Luxembourg.

Figure 10.6. Showing the Resistance menu.
about the options available for the user regarding the layers and how to load them on to the foreground whenever necessary. Figure 10.7 shows the same.

**Figure 10.7. Displaying the Read Me menu.**

### 10.5 HELP

The Help menu pops up a window describing all the functionalities of the tools available in the application to use and also how to use them. Figure 10.8 shows how the tips in the Help menu looks like.
Figure 10.8. Help menu with the About the Tool option.
CHAPTER 11

CONCLUSION AND CHALLENGES

The intention of this thesis is to develop an interactive GIS tool, that educates the users about the important events which led to fall of France. This GIS application uses the functionalities offered by ESRI to represent the history of fall of France and the Lower countries on maps. To represent the information in creative way, this application also makes use of web pages.

Some of the features provided by this GIS application to the user are

- Reading CSV files and plotting the places of attacks on the map.
- Creation of shapefiles from existing shapefiles.
- Implementation of a button to provide user a one-click navigation to the web page that contains information, photos and video of the battle.
- Implementation of Hotlink to provide photo gallery of the places plotted on the map.

The challenges involved while developing this application was to make the section of the neighboring countries visible along with the country that was attacked. The application provides a functionality to create shapefile of an entire country and load it as a map. However it was challenging to make only a section of a country appear on the map instead of the entire country.
CHAPTER 12

FUTURE ENHANCEMENTS

This application is developed to provide detail information of all the events that lead to the invasion of France during World War 2. Although it provides information about the battles fought during this period, there are several improvements that can be incorporated in the application. Few of the future improvements are as mentioned below.

- A timeline can be created for this application using slider.
- More information can be provided about the battles and the role of people involved in the battles.
- Additional features like quiz can be implemented to make the learning more interesting and enjoyable.
- More web pages can be created to give user more information about the battles.
REFERENCES


