

# AAiT

Addis Ababa institute of Technology  
አዲስ አበባ ቴክኖሎጂ ኢንስቲትዩት  
Addis Ababa University  
አዲስ አበባ ዩኒቨርሲቲ

March, 2015

## DESIGN AND ANALYSIS OF GRAPHICAL USER INTERFACE FOR TRAIN TRAFFIC CONTROL FOR CASE OF ERC

### Thesis Report

**By:** Yitagessu H/Michael

**Id:** GSR/3882/05

**Advisor:** Abi Abate

**Faculty:** Addis Ababa Institute of Technology

**Department:** Electrical and Computer Engineering

**Major:** Electrical Engineering for Railway Systems

March, 2015

**Thesis Research on**

# **Design and Analysis of GUI**

---

## **For Train Traffic Control Case of ERC**

**By: Yitagessu H/Michael**

**Advisor: Abi Abate**

**Addis Ababa Institute of Technology Department of Electrical and Computer Engineering, MSc in Railway Electrical Engineering March 2015.**

# Abstract

Centralized Traffic Control, commonly named CTC, is the most common way nowadays of controlling train traffic. It is based on the principle of centralizing all information concerning railroad tracks over a given area and to control trains running on it by the mean of a single person, the train dispatcher. This technology needs efficient telecommunications as the employee who dispatches trains is not physically present at a station but more commonly at a remote place, namely the CTC office. The CTC retrieves railway signals and gives useful information to the train dispatcher using good design Graphical User Interface.

Although it was a completely hardware device, CTC is now most of the times a software which summarizes signals, headways and trains to allow the train dispatcher taking good decisions, particularly when some trains stop or have delay. Some of CTC are even partially automatic and can take routine decisions by themselves, letting only critical decisions for human control. It also most of the times prevent human errors by avoiding decisions which could result in train collisions.

This thesis objective is to design and analysis of Graphical User Interface for Train traffic control system for Ethiopian Railway Corporation. In this context, the CTC is the run time user interface and alarm handling facility for the stationary part of the system, subsystem for Centralized Traffic Control. Therefore, based on international standard graphical user interface design process, this thesis will provide analysis by compare and contrast selected candidate frameworks. One of the candidate framework will be current ERC GUI framework. And based on its merits and demerits the report will recommend an appropriate framework using standard specifications.

# Acknowledgments

I would like to thank all the people who support me during this thesis, Abi Abate my advisor in Addis Ababa Institute of Technology Electrical and Computer Engineering Department and ERC signaling and railway traffic controlling staffs and all my friends and family.

# List of Figures

Figure number	Description	Page number
<b>Figure 1.1</b>	Typical Graphical User Interface System	2
<b>Figure 1.2</b>	System Architecture	3
<b>Figure 5.1</b>	Wenzel-MACS-R architecture	11
<b>Figure 5.2</b>	GSM-R screen optimized for touch operation	11
<b>Figure 5.3</b>	Audio and GUI Interfaces	12
<b>Figure 5.4</b>	GSM-R screen for mouse operation	13
<b>Figure 6.1</b>	Java Cycle	18
<b>Figure 6.2</b>	Java Platform Standard Edition	19
<b>Figure 6.3</b>	.NET Framework Platform Architecture	20
<b>Figure 6.3</b>	C# Code Cycle	21
<b>Figure 6.4</b>	C/C++ Building Process	22
<b>Figure 6.5</b>	Java Multiple Document Interface Design	25
<b>Figure 6.6</b>	C# Multiple Document Interface Design	26
<b>Figure 6.7</b>	C++ Multiple Document Interface Design	27
<b>Figure 6.8</b>	XML Processing Layers in .NET	30
<b>Figure 7.1</b>	Design Process	39
<b>Figure 7.2</b>	Kality Depot Schematic Schema	42
<b>Figure 7.3</b>	Railway Track Layout Class flow chart	44
<b>Figure 7.4</b>	Current Information Updater Class flow chart	45
<b>Figure 7.5</b>	Train Information Class flow chart	46
<b>Figure 7.6</b>	Execution Result When The Database Connection is exist	48
<b>Figure 7.7</b>	Execution Result When The Database Connection is doesn't exist	49
<b>Figure 7.8</b>	Execution Result and updated Database information	50

# Abbreviations

<b>ATS</b>	Automatic Train Supervision
<b>CATS</b>	Central Automatic Train Supervision
<b>ERC</b>	Ethiopian Railway Corporation
<b>CREC</b>	China Railway Engineering Corporation
<b>CMI</b>	Command Machine Interface
<b>MDI</b>	Multiple Document Interface
<b>CTC</b>	Central Train Control
<b>AWT</b>	Abstract Window Toolkit
<b>API</b>	Application Programming Interface
<b>XML</b>	Extensible Markup Language
<b>CLR</b>	Common Language Runtime
<b>MSIL</b>	Microsoft Intermediate Language
<b>RIA</b>	Rich Internet Application
<b>WPF</b>	Windows Presentation Foundation
<b>RAD</b>	Rapid Application Development

# Table of Contents

<b>Abstract</b> .....	i
<b>Acknowledgements</b> .....	ii
<b>List of Figures</b> .....	iii
<b>Abbreviations</b> .....	iv
<b>1. Introduction</b> .....	1
1.1. Background.....	1
1.2. System Architecture.....	3
<b>2. Context and motivation</b> .....	4
<b>3. Problem statement</b> .....	5
<b>4. Contribution and Objectives</b> .....	6
<b>5. Literature Review</b> .....	8
<b>6. Framework Evaluation</b> .....	17
6.1 Overview.....	17
6.2 Target Frameworks.....	17
6.2.1 Java .....	17
6.2.2 C#.NET .....	20
6.2.3 C++ .....	22
6.3 Requirements Analysis.....	24
6.3.1 Requirements Specification.....	24
6.3.2 Support for Local Languages.....	27
6.3.3 Socket Communication .....	28
6.3.4 XML Parsing.....	29
6.3.5 Database Connectivity.....	31

6.3.6	Portability.....	32
6.3.7	Ease of Use for Developer.....	33
6.3.8	Widely Common in Industry.....	34
6.3.9	Maturity.....	34
6.3.10	Maintainability in Future.....	34
6.4	Conclusion.....	35
<b>7.</b>	<b>GUI Design.....</b>	<b>38</b>
7.1.	Overview.....	38
7.2.	Application Design Process.....	38
7.2.1	Design Theories and Methodology.....	38
7.2.1.1	Data Collection.....	39
7.2.1.2	Analysis of Data .....	40
7.2.1.3	Focusing on Users.....	41
7.2.1.4	Prototyping.....	41
7.2.1.4.1	Application Discussion .....	43
7.2.1.5	Application Result.....	47
<b>8.</b>	<b>Conclusion and Recommendation.....</b>	<b>52</b>
	<b>Appendix</b>	

# 1. Introduction

This research paper has been divided into four chapters. The first chapter provides the background, motivation, problem statement, contribution, reviewed literatures, system architecture and objectives of the thesis. The second chapter describes the evaluation of candidate frameworks considering the requirements in order to choose the most appropriate framework for this application. The third chapter explains the design process of different parts of the application. At the end of each section within the chapters the results has been explained. The final chapter provides the conclusion and recommendation.

## 1.1 Background

The Ethiopian Government has been undertaking mega railway transportation projects, which aims to enhance the railway transportation network within the country by connecting to adjacent countries, cities and ports. Finalization of this project will provide efficient mobility; improve the export and import activities and boosting the economic development. Addis Ababa LRT and National Railway Networks are the major projects constituted in the plan. Hence, Ethiopian Railway Corporation (ERC) is the only government corporation responsible to manage all these projects. According to the ERC construction agreement, the major companies those under construct more than 60 % of the systems are held by Chinese companies [1]. Most of the equipment those are need to operate, control and monitor of the overall railway systems are manufactured and imported from different countries (specially: from China) [1].

Train traffic control with the support of graphical user interface is the backbone for proper and safe operation of railway transportation system [2]. The perfect integration of control machine (CM) and control command (CC) is a guarantee for appropriate functionality of train traffic flow. The speed profile, position and current truck of the given train are represented as program variables and transmitted over a communication media using standard protocol and they displayed on the monitor screen which is located in Centralized Traffic Control (CTC). Representation of the train or an object on the screen is depends on the framework that the system use. Currently we can find

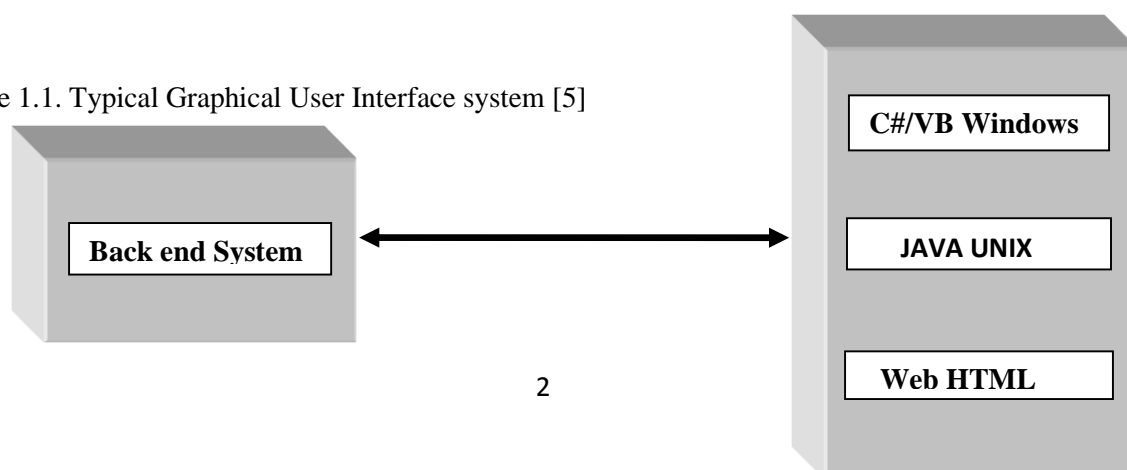
different kinds programming languages those are used to perform the same task but they do have their own merits and demerits, according to the system which they are applied [3].

Therefore it is very intentional to choose essential framework for graphical user interface (GUI) train traffic control system using proper evaluation techniques and technical experiences. Interfacing is one of the major systems that have responsible for:-

- Safe operation of the train
- Safe transportation of fright and passengers
- Existence of efficient and reliable railway transportation system
- Precise and fast train traffic control

It can also provide means of communication between onboard signaling and CTC. CTC is the most common way nowadays of controlling train traffic [4]. It is based on the principle of centralizing all information concerning railroad tracks over a given area and to control trains running on it by the mean of a single person, the train dispatcher. This technology needs efficient telecommunications as the employee who dispatches trains is not physically present at a station but more commonly at a remote place, namely the CTC office. The CTC retrieves railway signals and gives useful information to the train dispatcher with the help of a good design GUI. In this regard, this thesis focuses on investigation of the feature of GUI and finally to recommend appropriate GUI based Train Traffic Control system for ERC.

According to [5], “Graphical User Interface (GUI) is a graphical software layer exists between the human and the system, to provide an easier way to communicate with the back end system for administration, configuration, information exchange, etc”. For a typical GUI system, the GUIs may be created from the implementation of C# or Visual Basic of the Windows platform, Java or UNIX platforms, or HTML or web, which are strongly coupled with the back end system. The figure below illustrates this



## 1.2 System Architecture

The Central Train Control (CTC) is the run time user interface and alarm handling facility for the stationary part of the system. CTC GUI provides the train dispatcher with means of controlling and supervising the system. CTC communicates with all train control functions. Alarms are read from and manipulated in the database, where they may be placed by any subsystem. Figure 1.2 shows the detail interconnection between each functional parts of interfacing system.

The onboard unit (Train) controlled and communicate via GSM-R Wireless from CTC. Every information signal which is came from the onboard unit is stored in the central station server. CTC PC will retrieve every train's parameter such as train position, speed, distance, status and track occupation and status etc. will continually accessed from the database using the application on the PC called graphical user interface.

The database also possible to provide train information for outside world using internet by making some sort of configuration.

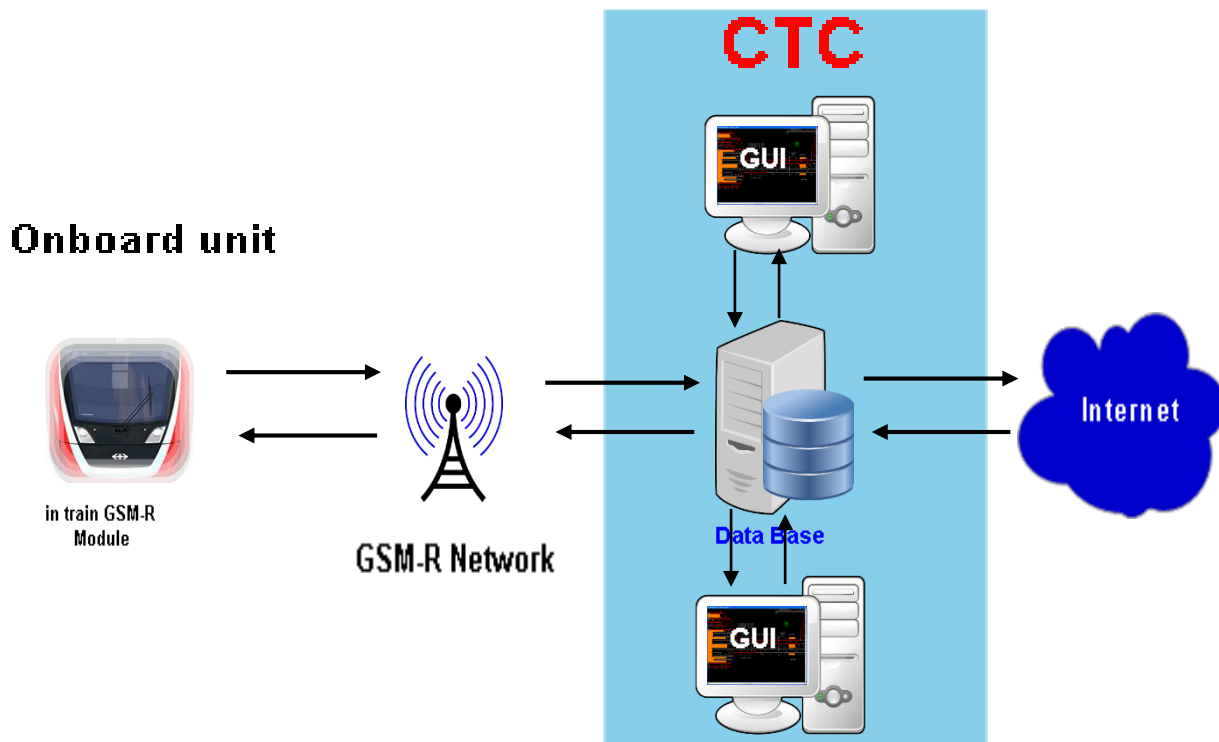


Figure 1.2. System Architecture [7]

## 2. Context and Motivation

Run time train traffic control system is the major functional system that guarantees the safe and reliable train operation and passenger safety. This system will be installed and configured by the responsible Chinese Company. If there is no proper technology transformation between the client (ERC) and contractor (CREC) it may be exist some sort of operational mismatches between the dispatcher and the graphical user interface (GUI).

The object of the thesis project is to design and analyze of GUI for train traffic control for the new implemented railway system using state of the art technologies. The application will have potential to control and monitor train traffic by visualize track layout on CTC screen using local (Amharic) and international languages.

### 3. Problem Statement

As mentioned before run time traffic control system is the major functional system that guarantees the safe and reliable train operation and passenger safety. This system will be installed and configured by the responsible China Company. If there is no proper technology transformation between the client (ERC) and contractor (CRC) it may be exist some sort of operational mismatches between the dispatcher and the graphical user interface (GUI). Hence, the problem is to investigate the dispatcher and the GUI based the existence of such mismatches. In this regard, this thesis focuses on investigation of:

- Which kind of framework (java, C#, C++ ..... ) use
- What kind of standard it follows
- How represent the train on dispatcher screen
- How support local languages
- How support for socket communication
- How support for database connectivity
- How capable to integrate different graphical components
- How the framework ease of use for local developer
- How the framework portable

## 4. Contribution and Objectives

### 4.1 Contribution

According to the Ethiopian government announcement the costs of all railway projects across the country will be more than 3 billion US dollars. Such kind of mega projects will invite different foreign companies. Due to the involvement of such companies the system will introduce technology transformation between them and responsible local staffs. Such that, responsibility of handling of operation and maintenance of the system will be reside on the client company (Ethiopian Railway Corporation). After finalized of the system, in the economic or technical point of view, minimizing of foreign professionals intervention will have different benefits on economic and technological development for Ethiopian people. Therefore, taking this idea into consideration, this thesis paper will have a vital contribution on the process of managing, maintaining and controlling of ERC GUI for train traffic control system as well as other software based railway systems by providing the following core issues. These are:-

- Encouraging local software developers to join and participate in the railway system
- Minimizing foreign currency because of advance maintenance
- Develop confidence on citizens to work on high-tech systems
- Showing the way that how to evaluate available frameworks for specific task
- Providing information about candidate frameworks and their differences and relations
- Introducing GUI design process
- Showing the way that, how to use local language on the selected framework

### 4.2 Objectives

#### General Objective

In general, the objective of this thesis is to investigate run time graphical user interface for our new implemented train traffic control system. Additionally it also design prototype application for Train Control Center (TCC) using local languages according to the available communication protocol, with the aim of efficient and user friendly.

### **Specific Objectives**

Specifically; the aim of this thesis is to:

- Review the conventional GUI platform standards and analyze their role in railway system.
- Theoretical understanding of the event and influencing parameters on GUI platform.
- Investigate the Addis Ababa light rail transit GUI platform and framework
- Evaluate frameworks based on the selected platform.
- Build a prototype simulation using local language and conduct tests and experiments.

## 5. Literature Review

Because of its importance on railway system, GUI has got wide research attentions in the last few years. By applying recent communication technologies it will provide good controlling scheme. According to this the researchers conducted on the area GUI and related to this work are briefly reviewed below. These are:-

- Railway communication systems
- Dispatcher system for GSM-R
- GUI (Graphical User Interface)
- Framework characteristics for GUI

### **Railway communication systems**

In [6], it is well reviewed that how railway communications emerged almost exclusively from the communication between fixed elements to carry out traffic management and circulation regulation. By continuing his observation the author explain about the technologies that communicate fixed elements with mobile elements (trains), how relatively recent, and they have contributed to improve and simplify the work required for rail service exploitation. Therefore, according to this he focused on the network topologies by categorized and identified in to two within the field of railway communications: a first one involving only fixed elements, and a second one involving both, fixed and mobile elements (called "train-to-earth" communications).

In another book [7], these two important topics are reviewed as follows.

- The communication between fixed elements and trains has been established using analogical communication systems, such as the traditional telephone or PMR (Private Mobile Radio) based on radio systems. These analogical systems are still used for voice communications and issues related with signaling. However, their important limitations in terms of bandwidth are causing the migration to digital systems, which offer a higher bandwidth.
- Among the technologies of communication "train-to-earth", one of the most important advances of the last decade has been the GSM-R (Global System for Mobile Communications - Railway). This system is based on the GSM telephony, but has been

adapted to the field of railways. GSM-R is designed to exchange information between trains and control centers, and has as key advantages its low cost, and worldwide support.

### **Dispatcher system for GSM-R**

In the railway transportation environment there are a variety of companies those provide railway controlling and signaling systems. Among the different Wenzel Electronics [8] is the most legend and popular company which provide standard dispatcher system for GSM-R. According to [9] graphical user interface requirement the system has many feature for dispatcher station control.

As stated in [8] the name of the system is called Wenzel-MACS-R it is a modular communication system for railway applications, outstandingly suited for use in Train Control Centers (TCC). It permits access to a wide range of communication technologies used in railway infrastructures such as legacy technologies (e.g. like local battery LB, central battery CB, ISDN and analogue train radio) as well as GSM-R Voice over IP.

Integration of the various communication services via a uniform system and unified dispatcher terminals leads to a significant reduction of operating costs.

Legacy systems can be connected and functionally integrated to Wenzel-MACS-R. The versatile interface modules for a variety of subscriber and network interfaces guarantee high protection of investments because of the integration of existing communication equipment and infrastructure.

Thanks to the modular architecture the Wenzel- MACS-R can be scaled from a system with a few dispatchers controlling a single line up to a country wide network with several train control centers and many stations at the lines

Wenzel-MACS-R ensures high reliability due to full redundant system architecture. The system redundancy comprises self-restoring network topologies ensuring the redundancy of central control components and different default routing strategies in case of failure.

As part of the Wenzel-MACS-R the MACS-R Role Server provides an efficient role management where calls are notified to all dispatchers registered for the related role. This ensures role sharing by several dispatcher terminals during peak hours and handling of several roles by one dispatcher. The role management secures that no role can be left unmanned in the system and no calls get lost. The Wenzel-MACS-R central Voice Recording System (VRS), connected at the interface to the MSC, records all GSM-R calls including extensive Call Related Data (CRD) like functional numbers etc.

The Wenzel-MACS-R switch is connected to the GSM-R network via redundant ETSI PRI (S2m) EIRENE compliant fixed connections to the Mobile Switching Centre (MSC). Wenzel-MACS-R Dispatcher terminals can be connected to the switch via basic rate interface (S0), U-interface (Up0), LAN (VoIP) or GSM-R air interface.

The Wenzel-MACS-R dispatcher terminal allows the use of various types of dispatcher consoles for GSM-R as well as for legacy train control voice applications. From desktop dispatcher terminals like the Wenzel-MACS-GeFo particularly suited for local train stations up to screen based workstations with a graphic user interface (GUI) for train control centers a wide range of dispatchers are available.

The GUI dispatcher terminals are based on a low power embedded Linux PC without any moving parts. This includes an optional 2 Watt GSM-R radio module and the EIRENE featured optional VoIP adapter.

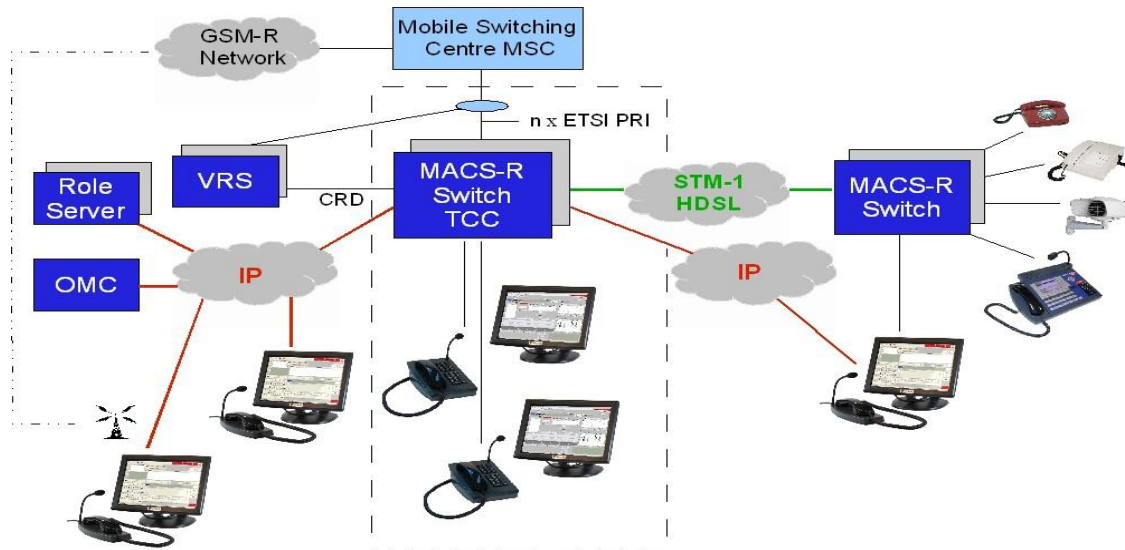


Figure 5.1. Wenzel-MACS-R architecture [8]

The ergonomic design of the user interfaces ensures a smooth, convenient operation even in critical situations. The dispatcher terminals are fully compliant to the EIRENE specification. Administration, configuration, and maintenance is done remotely from the central OMC.

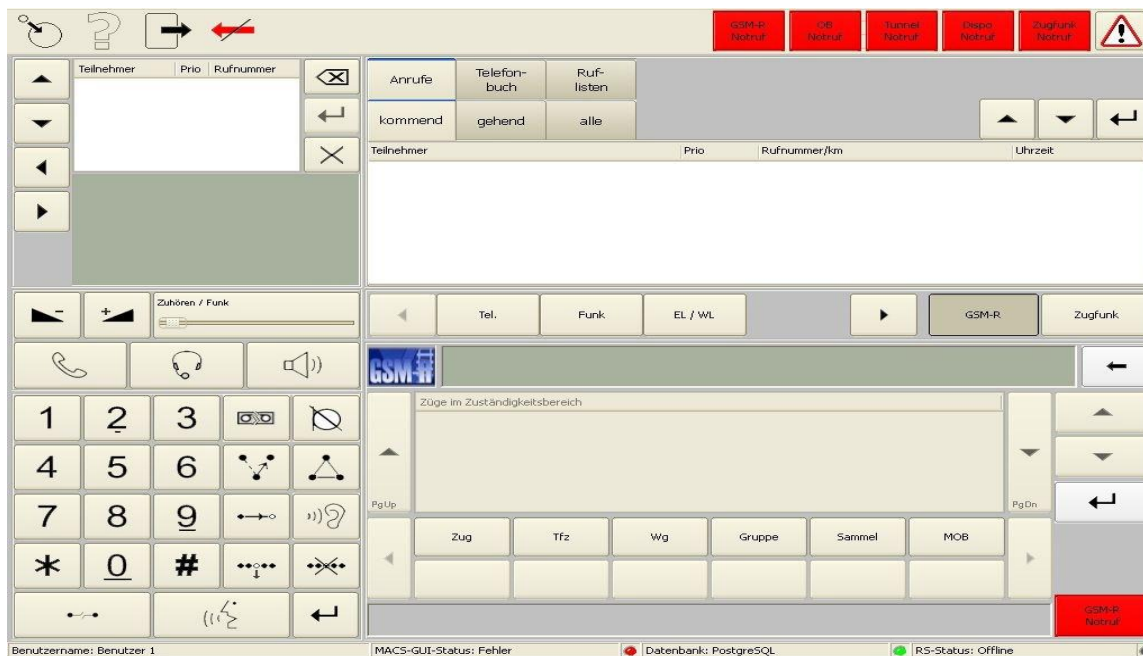


Figure 5.2. GSM-R screen optimized for touch operation [8]

For the Wenzel-MACS-R system several operator consoles are available. They can be combined with touch panel devices (12"-19") or monitors with mouse operation. The **audio device** is best

sited for operator stations with extremely limited desk space all functionalities like call acceptance and dialing are managed via mouse or touch panel.



Figure 5.3. Audio and GUI interfaces [8]

The **MFTT** is optimized in desk space and functionalities. Due to programmable keys for most used functions as well as for RECs and navigation the handling is easy. Arbitrary calls or group calls can put onto the listen-in loudspeaker. The MFTT supports call acceptance and dialing functional numbers by special keypad. The **MFTK** comprise 48 line keys, 8 programmable LCD keys, dialing keys as well as functional keys. All incoming calls are queued according to time and priority onto the LCD keys. Emergency calls are lighted in red. The key shows mnemonic, functional address or number of incoming calls. The operator can answer the call by simply pressing the LCD key.

Meanwhile all operator consoles mentioned above include a goose neck microphone, hands free loudspeaker, hand set with integrated PTT button, and connectors for an optional headset.

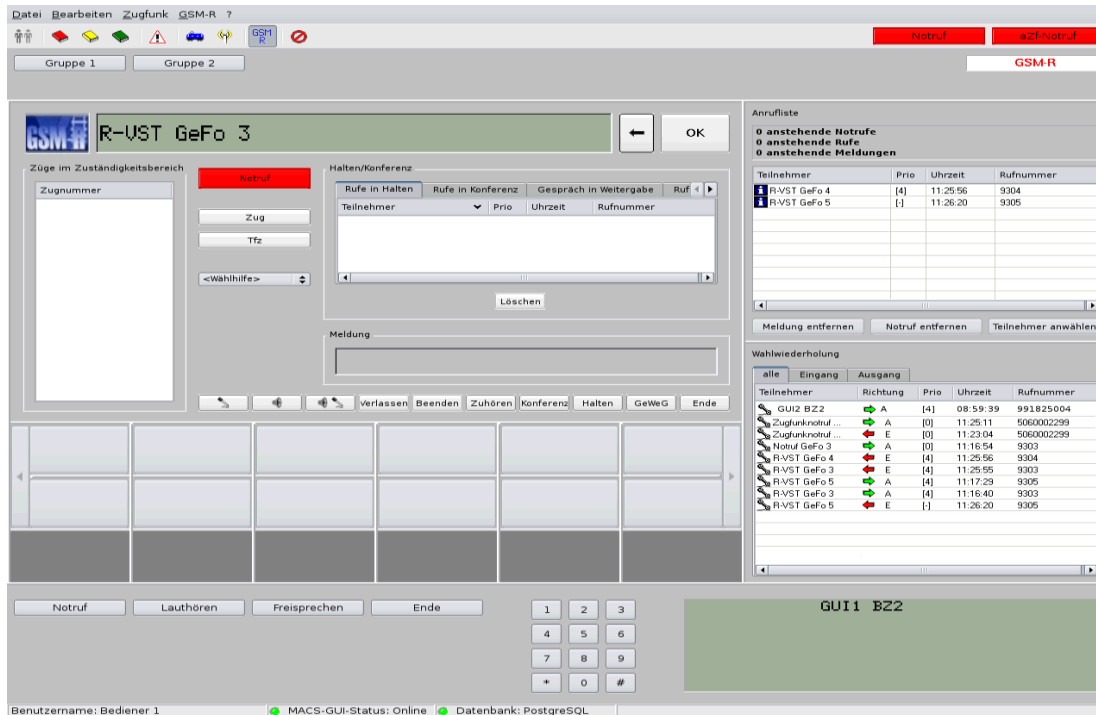


Figure 5.4. GSM-R screen for mouse operation [8]

## GUI (Graphical User Interface)

The invention of GUI started back in the late year of 1970s. This was originated at the Xerox Palo Alto Research Laboratory before the Apple Company used it in their first Macintosh computers. Microsoft later used the same ideas in their first version of the Windows operating system [9][10].

In [11], the authors investigated that is gathering as much information about railway system GUIs, GUI builders and good UI design. However, the review will emphasize more on the areas of GUI builders and UI design. The reason being is, both are specified to be the problematic domains that need to be carefully identified, as their roles in contributing to the production of the generic guidelines to design GUI for railway system.

In reference [12], the author clearly distinguishes GUI builders into four kinds of tools. These are:-

- *Language-based tools*
- *Application framework*
- *Model-based generation and*
- *Interactive tools.*

In Getting Started GUI Development with Java [13] the author outlined that, there are two basic types of GUI program in Java: these are, *Applets* and *Stand-alone applications*.

### **Applets**

An applet is a program that runs in a rectangular area on a Web page. Applets are generally small programs, meant to do fairly simple things, although there is nothing to stop them from being very complex. Applets were responsible for a lot of the initial excitement about Java when it was introduced, since they could do things that could not otherwise be done on Web pages. However, there are now easier ways to do many of the more basic things that can be done with applets, and they are no longer the main focus of interest in Java. Nevertheless, there are still some things that can be done best with applets, and they are still fairly common on the Web.

### **Stand-alone applications**

A stand-alone application is a program that runs on its own, without depending on a Web browser. Any class that has a *main()* routine defines a stand-alone application; running the program just means executing this *main()* routine. And also can be executed independently and would execute and produce some output either as a UI or on the JVM console. Any java class with a main method can be considered a mini standalone java application. However, “command-line” programs, where the user and computer interact by typing things back and forth to each other. Therefore by continuing his explanation the author In [14] asserted that, how GUI program offers a much richer type of user interface, where the user uses a mouse and keyboard to interact with GUI components such as windows, menus, buttons, check boxes, text input boxes, scroll bars, and so on. And also, the main routine of a GUI program creates one or more such components and displays them on the computer screen. Very often, that’s all it does. Once a GUI component has been created, it follows its own programming—programming that tells it how to draw itself on the screen and how to respond to events such as being clicked on by the user. Meanwhile, he advised that a GUI program doesn’t have to be extremely complex.

To conclude, in order to make this system more efficient, it is valuable to consider the performance of the system using an appropriate framework. And the selection and implementation method is must be follow standard procedures. Therefore, these and other related reviewed references have a vital rule to design and analyze GUI for railway system. Even though, the above literatures

focused on the applications those need to use in the user interface side, it must be consider that there are other related issues exist between onboard equipment and GUI such as, communication equipment , protocols and medium (or *transmission medium*).

### **Framework characteristics for GUI**

In [15] the author emphasize that how portability is one of the biggest obstacles for desktop and enterprise applications is at GUI level. According to his observation, traditionally GUIs have been application specific both in characteristics and in terms of the API. By continuing his analysis he stated that, a measure of portability can be achieved by building an abstraction layer that is consistent across platforms. Underneath the abstracted API binds to the underlying graphics engine, such as the Win32 API for Windows, or a flavor of X11 on UNIX or Linux platforms. The GUI has been considered to be tightly tied to the operating system because that operating systems' users are sensitive to particular look and feel and expect a consistent functionality across applications on their platform.

He also compare some frameworks based on the portability performance and he stated that neither C nor C++ provide a standard GUI API, leaving questions of GUI –specific bindings outside the scope of the language. In contrast, Java incorporates GUI APIs (Swing and AWT) directly in the standard. Applications written to the Swing standard ought to have high portability between confirming platforms through the use of a common API. As mentioned previously, the SWT API, defined as part of the open sources Eclipse libraries, offers another portable graphics that is widely used by Java programmers. Whereas the goal of SWT is to offer a consistent cross-platform look and feel, the goal of SWT is to offer cross-platform portability with a look and feel matching that expected on the target platform. Which is better is a matter of opinion, but both offer a means to source code portability.

With the same topic another researcher [16] tried to point out frameworks performance in Real-time applications. What he pointed about C language is, the language used most frequently in embedded programming. Compared to assembly language, C has a more readable format and allows the compiler take control of the registers so that the programmer cannot modify the register directly. However, C is not a safe language for several reasons. First, C uses pointers, a mechanism that allows the programmer to accidentally overwrite or corrupt crucial data. Second, C has no

internal support for exceptions. All errors are handled by return values, which results in inefficient and complex error handling code. Also, C does not have a run-time environment and the programmer needs to take responsibility of memory management and watch for misbehavior of data structures. When he state his observation about Java he explained that, Java is designed to be safe and robust. Java programs are run by a Java Virtual Machine, which handles a lot of low-level features. Hence, Java has a lot of advantages over other programming languages. However, safety has a cost in increased execution time. First, Java uses reference instead of pointers and all memory management is handled in the JVM. The programmer can still access memory location by using object reference instead of pointers. Also, Java supports run-time data structure checking and dynamic memory allocation to data structures. Java can use assertions to verify data type consistency. Besides these, Java has full support for exception handling. Programmers can define exceptions that need to be caught by the JVM. Lastly, Java has run-time garbage collection to restore the memory that was used by objects and will never be accessed again to avoid memory leaks. Although the above features make Java to be a safe language, there are still some problems to be handled in order to use Java in a real-time application. Java is designed to be just-in-time compiled. This feature is important for distribution of small Java applications or applets. However, a real-time system is normally compiled once and runs on the same platform for many times. This difference means that the Java compiler will not produce optimized code for a real-time system. This difference means that the Java compiler is optimized to compile fast rather than produce code that executes fast. Another issue with Java is its thread. Although threading mechanism in Java is easier to use and functions well, threads can result in indeterminism in concurrent programming. As the programmer does not know when a thread is running, it is may be fatal for a hard real-time task to meet time requirement.

# 6. Framework evaluation

## 6.1 Overview

This chapter as explained earlier in the paper is dedicated to the evaluation of frameworks. According to the first objective of the thesis there was an evaluation on different candidates for frameworks. The requirements to be considered includes capability to meet the application requirements specification, support for local languages, support for socket communication, support for XML parsing, support for database connectivity (MYSQL), integration with available graphical component, portability, ease of use for developer, being widely common in industry, maturity and finally maintainability in future.

## 6.2 Target Frameworks

Main candidates for frameworks were Java and C sharp, besides the upgraded version of PowerBuilder should be considered as well. In order to do the evaluation, first I provide an overall description for each of these candidate frameworks considering their advantages and drawbacks, and then provide analysis of the each framework remarking the framework requirement for this thesis.

### 6.2.1 Java

Java platform and language has been started in December 1990 by Sun Microsystems and for the first time it was released in 1994 and the current version is Java SE8 released on March 18 2014. The target was as an alternative to available programming languages such as C and C++ languages, while being object oriented language, get rid of memory managing issues and being portable cross different platforms [17].

Java programming language is a general-purpose language and it gives the possibility to write code on wide range of areas from a piece of code for mobile phones to enterprise computer network applications, “expecting low level code which deals directly with underlying hardware.” [18] It

provides some mechanisms to perform platform dependent tasks, although Java is not made for this purpose [18].

There are several advantages which make Java one of the most common used programming technologies in industry which some of the most important one are explained here

It is an object oriented language in which application components are treated as objects. The developers need to understand the concept of object that is quite straight forward. Then can start coding by creating objects and manipulating them and reuse the objects in different pieces of code. Regarding the fact that The Java core libraries provide programmers a well-designed and intuitive set of APIs containing well-known set of classes with proper set of methods to manipulate common objects and perform common tasks on them

The other major advantage of Java is the portability. Put another way, Java is on a higher level of abstraction in compare to many other languages in order to make it portable regardless of underlying platform. This property enables the developers to create software application once and run it on several different environments regardless of the platform. The portability has been possible using software called Java Virtual Machine or JVM which intermediates between Java program and underlying platform. It comes up with some disadvantages as well for instance lower speed of execution although it has been improved by introducing Just-in-time compiler later on.

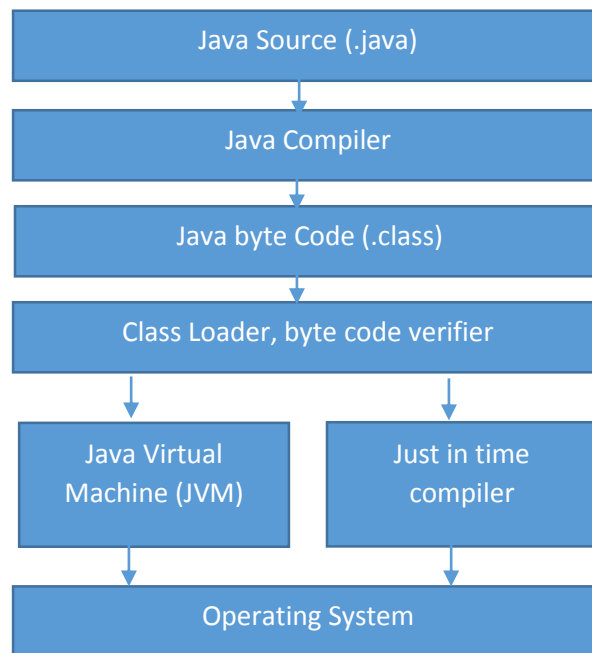


Figure 6.1 Java cycle

The other major facility providing by Java for the developers is automatic memory management using the Garbage Collector. It frees the memory from the created object when they are no longer used which is a tough and error-prone task for developers in some other languages like C or C++. Java is secure and highly reliable in the sense of error detection providing in compile time and also try-catch facility in run time to help the developer to point out the faults. These properties make Java an easy and elegant language for programmers to be more efficient and productive, compare to other programming languages.

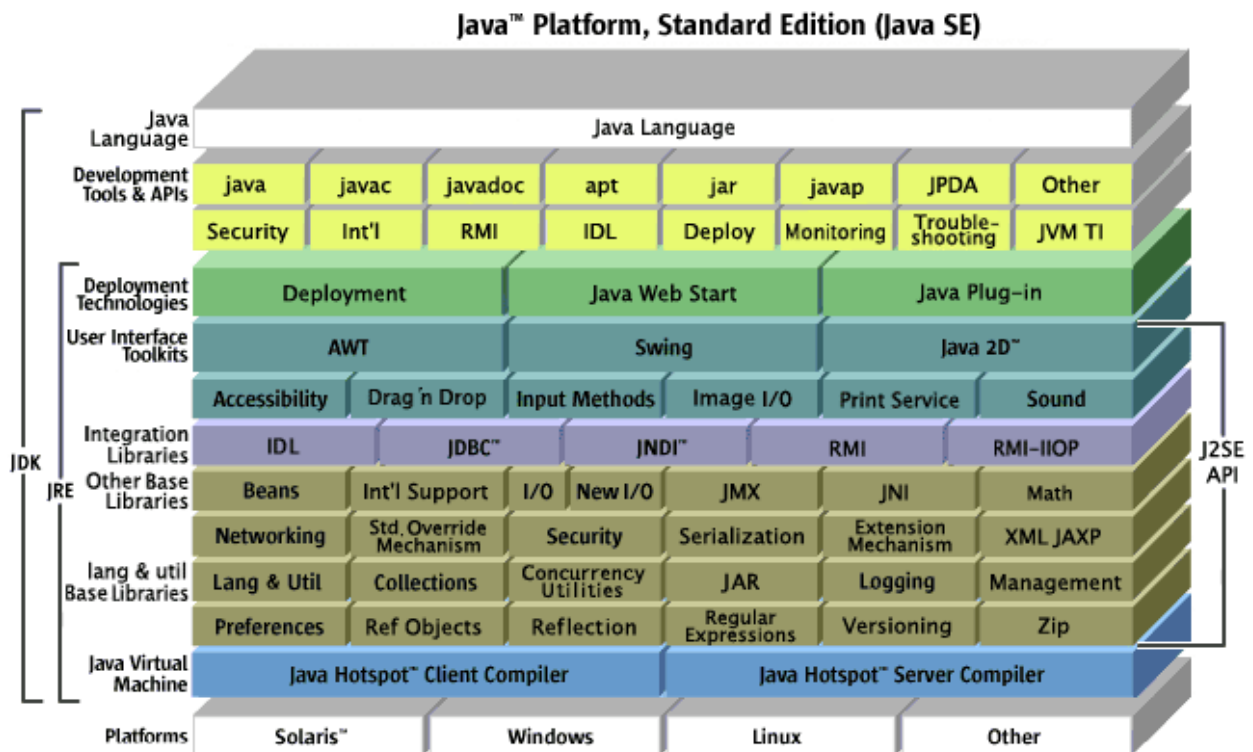


Figure 6.2 Java Platform, Standard Edition [17].

## 6.2.2 C#.NET

C Sharp or C# programming language had been started in January 1999 by Microsoft for .NET platform and for the first time published in 2000. The current version is C# 6.0 and it was released April 2014.

C# is common to be a multi-paradigm programming language, covering the concepts of object oriented, functional, component oriented paradigms and etc. C# is general purpose language which suited for developing a wide variety of robust applications for .NET platform from Windows-based to Web-based applications, particularly suits to develop software piece to use in distributed systems [19].

.NET Framework is a software component running on Microsoft Windows. This framework includes two main components; a virtual execution system called the Common Language Runtime (CLR) and a set of class libraries. Common Language Runtime is in charge of handling the code execution issues such as memory management, threading, exception handling and type safety. And the class libraries which are organized in to namespaces are containing a comprehensive set of predefined useful classes that enable developers to develop all kinds of applications ranging from CONSOL to Web application [19].

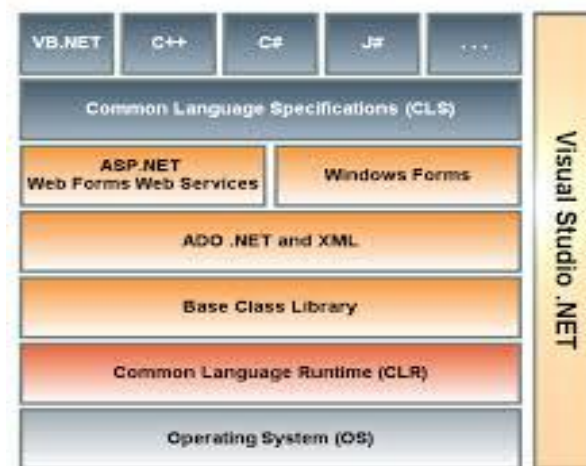


Figure 6.3 .NET Framework platform Architecture [19].

C# is a modern programming language which among the other programming languages is mostly compared with Java. The reason of this comparison is for having many common features similar to Java programming language. But since C# has been invented later than java, the creators have inspired from both the strengths and weaknesses of Java.

C# is object oriented programming language with strong type checking. Therefore developers can take advantage of creating reusable codes. One of the important design goals of C# is to support internationalization. It supports automatic memory management using Garbage Collector as well [20].

Despite the fact that C# source code is converted to an intermediate code called Microsoft Intermediate Language or MSIL which in runtime will be converted to system-purposed code, but this intermediate code currently only runs on few operating systems which the main one is Microsoft Windows platform. Therefore in the sense of portability Java is preferred than C#.NET since it supported on more operating systems [19].

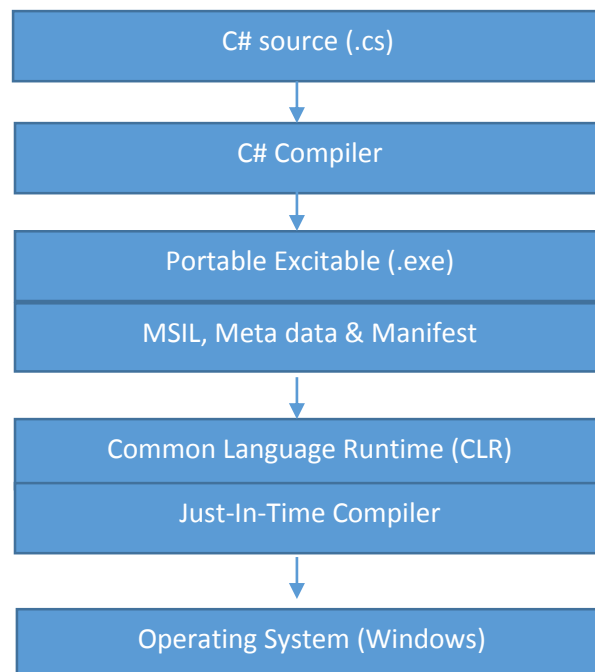


Figure 6.4 C# code cycle

### 6.2.3 C++

C++ is an object-oriented programming language. It was developed by Bjarne Stroustrup at AT&T Bell Laboratories in Murray Hill, New Jersey, USA, in the early 1980's. Stroustrup, an admirer of Simula67 and a strong supporter of C, wanted to combine the best of both the languages and create a more powerful language that could support object-oriented programming features and still retain the power and elegance of C. The result was C++. Therefore, C++ is an extension of C with a major addition of the class construct feature of Simula67. Since the class was a major addition to the original C language, Stroustrup initially called the new language 'C with classes'. However, later in 1983, the name was changed to C++. The idea of C++ comes from the C increment operator ++, thereby suggesting that C++ is an augmented version of C [21].

C++ is a superset of C. Almost all C programs are also C++ programs. However, there are a few minor differences that will prevent a C program to run under C++ compiler.

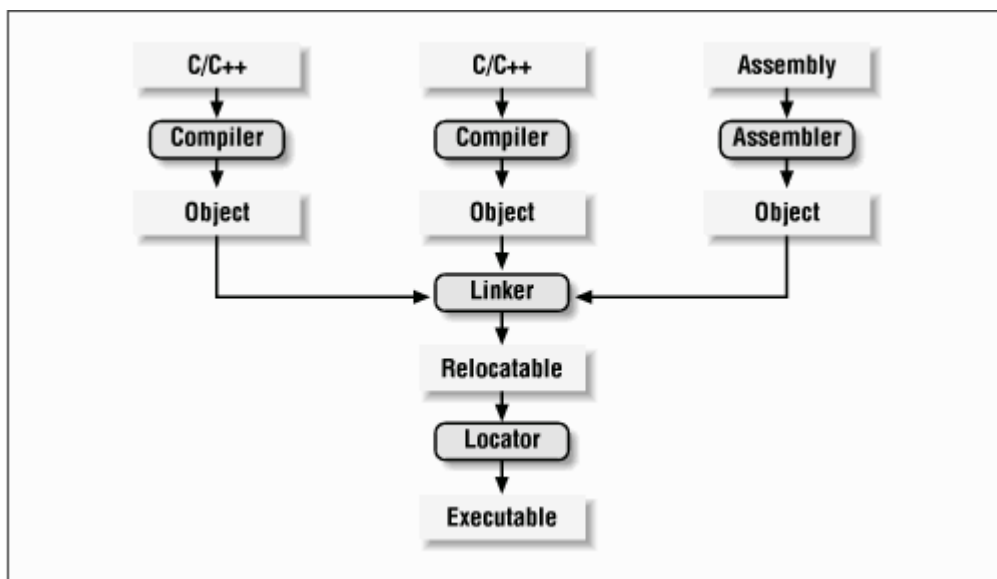


Fig.6.5 C/C++ building process

The most important facilities that C++ adds on to C are classes, inheritance, function overloading and operator overloading. These features enable creating of abstract data types, inherit properties

from existing data types and support polymorphism, thereby making C++ a truly object-oriented language.

C is in many ways hard to categorize. Compared to *assembly* language it is high-level, but it nevertheless includes many low-level facilities to directly manipulate the computer's memory. It is therefore an excellent language for writing efficient "systems" programs. But for other types of programs, C code can be hard to understand, and C programs can therefore be particularly prone to certain types of error. The extra object-oriented facilities in C++ are partly included to overcome these shortcomings. The current version is C++14.0 and released in December 15, 2014 [22].

One of the major features of C++ is classes. They provide a method of binding together data and functions which operate on them. Like structures in C, classes are user-defined data types.

It is a versatile language for handling very large programs; it is suitable for virtually any programming task including development of editors, compilers, databases, communication systems and any complex real life applications systems [21].

Since C++ allow us to create hierarchy related objects, we can build special object-oriented libraries which can be used later by many programmers. While it is able to map the real- world problem properly, the C part of C++ gives the language the ability to get closed to the machine-level details. C++ programs are easily maintainable and expandable. When a new feature needs to be implemented, it is very easy to add to the existing structure of an object. It is expected that C++ will replace C as a general-purpose language in the near future [21].

Since C++12, this strategy can make C++ language comparable with C#, Java or VB.NET, in the sense of additional programming features supported by that such as Arrays, Delegates, Parameterized Constructors, User-defined Enumerations and Generics. In the current version C++14 adds support for some more new features such as Multithreading. It has less flexibility, performance and more limitations comparing for instance Java, by the way in current version of C++ they overcame the extensibility issue somehow [23].

## 6.3 Requirement Analysis

According to programming experiences, it is difficult to conclude which language is the best and which one is the worst, because it pretty much depends on type of the target application. For instance some experiences show that in the case of Graphical User Interface development, C# Windows Forms are a lot better than Java Swing/AWT, because it is faster to develop, the applications run faster and look better while for developing Web application Java is great particularly on Linux servers. In the following section this report takes a look at the requirements of this thesis and then provides analysis of candidate frameworks' capability to fulfill the requirements.

### 6.3.1 Requirements Specification

The requirements specification for the controller machine interface (CMI) application was the old version that has been used for the earlier versions of the application. This specification mostly proceeds with the CMI graphical user interface requirements and provides very detailed description of interface components. After precise studies and considering the limited time, some parts of it ignored through some revision for this thesis.

Graphical User Interface of the CMI is expected to be designed as Multiple Document Interface or MDI. MDI is a Microsoft Windows programming user interface which enables user to work with multiple documents at the same time. MDI encompasses a parent window used as back or desktop window which can embed arbitrary number of child windows, secondary windows can be added to provide additional information as well. Through each window user can access different data through either menu bar or tool bar or both of them and viewing some status regarding window via status bar.

Almost all graphical user interface APIs provide some components proper to develop and manipulate MDI user interface. In the following part we provide an analysis of available facility provided by the candidate frameworks to fulfill the requirement specification. Java Swing toolkit provides some components to implement multiple-document interfaces, including JDesktopPane component proper to be used as the parent window, JInternalFrame class as child windows and

JDialog component as secondary windows. Classes JDesktopPane and JInternalFrame provide many methods for handling child windows.

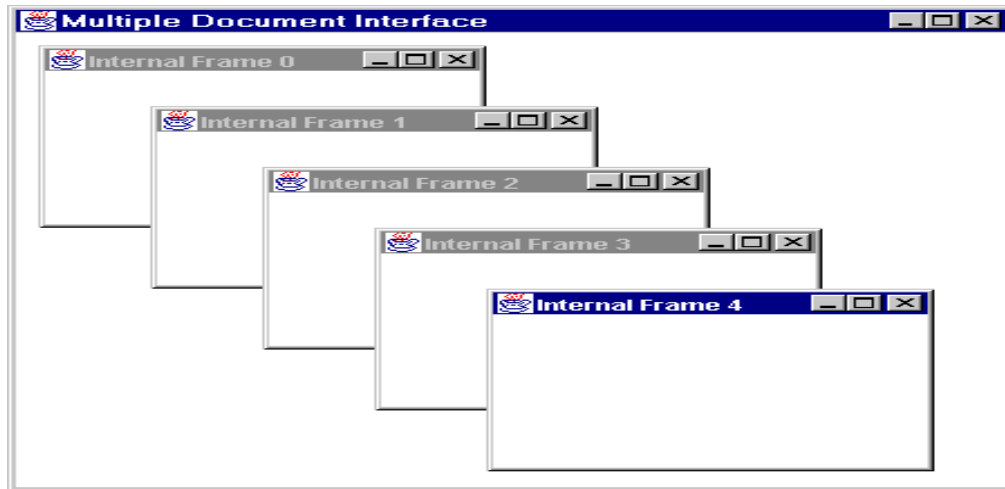


Figure 6.6 Java Multiple Document Interface Design [24].

Here we explain a simple piece of Java code to indicate syntax for implementing multiple-document interface.

**To create MDI Parent window;**

```
JFrame parentWindow = new JFrame ();
JDesktopPane desktop = new JDesktopPane ();
parentWindow.add(desktop);
```

**And To create MDI Child windows;**

```
JInternalFrame childWindow = new JInternalFrame ();
parentWindow.add(childWindow);
```

C#.NET programming language provides several properties through its graphical interface component, Form, to create and manipulate multiple-document interfaces. As we see in the following piece of code the C# solution is even shorter and straighter forward than Java.

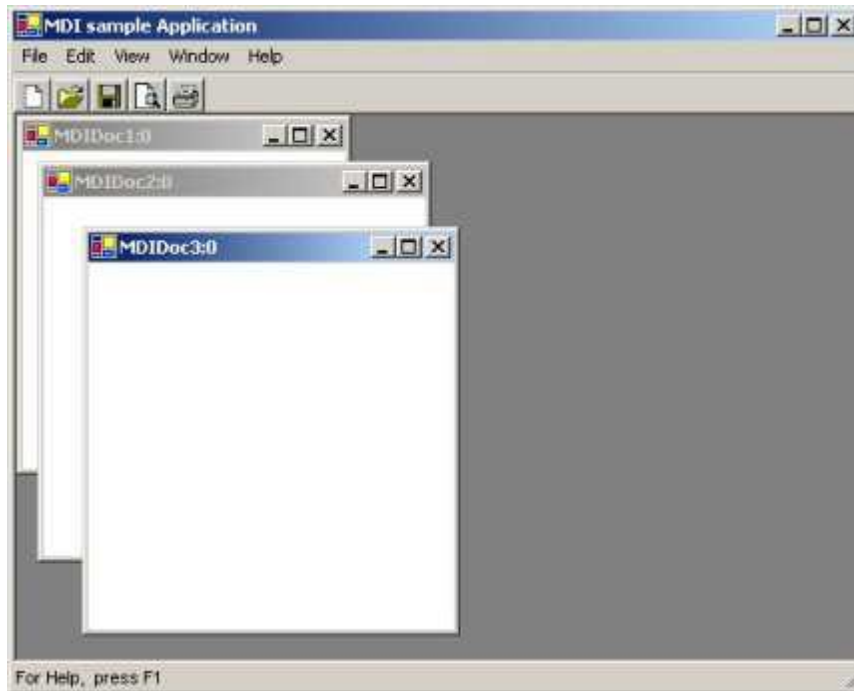


Figure 6.7 C# Multiple Document Interface Design [26]

**To create MDI Parent window;**

```
Form parentWindow = new Form ();
parentWindow.IsMdiContainer = true;
```

**And To create MDI Child windows;**

```
Form childWindow = new Form();
childWindow.MdiParent = parentWindow;
```

And manipulating child windows for instance keep them focused, is performed using ActiveForm property of an MDI Form[25][26].

In C++, to build multiple-document interfaces, either MDI Frame or MDI Frame with Visual C++ component are used as main or parent window.

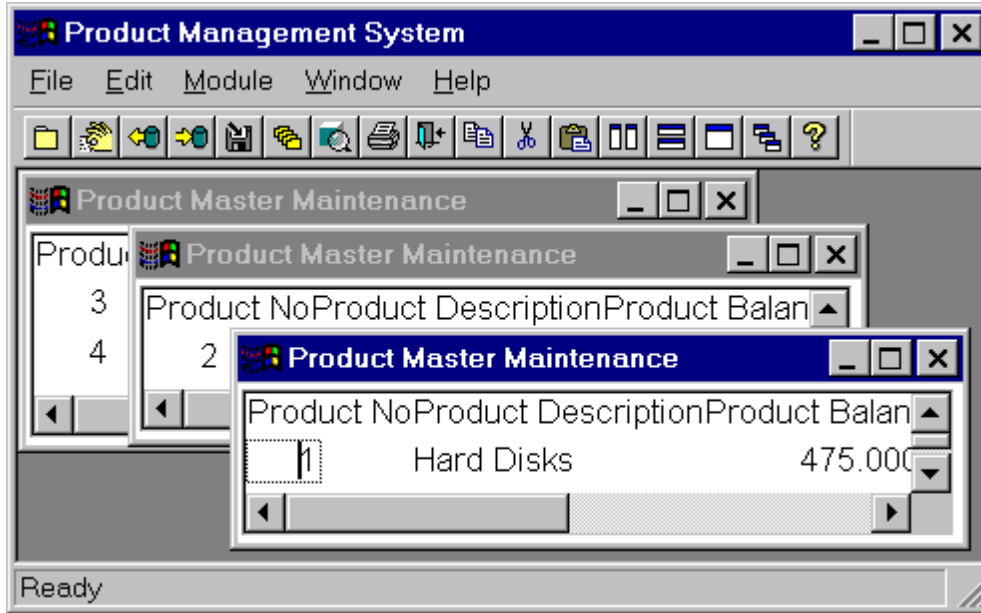


Figure 6.8 C++ Multiple Document Interface Application [27]

The MDI frame window consists of three parts, Frame, Client Area and Sheets. The Frame consists of components including menu bar, toolbar, window title and the status bar to display Visual C++, a brief description of the current menu item or current activity. Client Area is the area between MDI Frame title bar and the Visual status bar in which the sheets are embedded. Sheets correspond to the child windows to perform different user activities. These child windows are opened within the client area, using different functions provided by Visual C++ [27].

### 6.3.2 Supporting Local Languages

The GUI application is often subject to specific customer requirements. One of these requirements is the capability to adapt the application to different languages according to customers around the world.

This process of adapting application is called Internationalization. In order to internationalize an application, it required to be designed in such a way that enables the user to change the language according to region, easily and quickly without software engineering changes. The process of customizing the application to specific locale is called Localization which is accomplished using

locale and culture specific constituents such as translated text, fonts, dates and currencies. Since the software applications which only support English are getting old-fashioned, state-of-the-art software technologies support internationalization.

Java SE platform fully supports internationalization within its libraries facilitating language or culture-specific functionalities. A Java internationalization facility enables developers to fast and easy development of multi-lingual applications. For instance in java.text package, MessageFormat class provides local specific languages, or SimpleDateFormat class supports calendar specific eras and date formats for calendar systems different than Gregorian [28].

In .NET framework this language adaptation capability is called Globalization of which both concept of Internationalization and Localization as explained above, are considered as two aspects. In C# System.Globalization namespace is dedicated to Globalization using CultureInfo class. CultureInfo is the main class providing a set of different properties and methods to customize the application according to specific cultures [29].

Compared with the libraries provided with Java and C#, the Standard C++ library is quite narrow in scope of internationalization for GUI programming. The C and C++ languages and many operating system environments do not provide full support for Unicode and standards-compliant text handling services. Even though some platforms do provide good Unicode text handling services, portable application code cannot make use of them. To develop in these areas, C++ programmers are expected to use various (often platform-specific) third-party libraries [30].

### **6.3.3 Socket Communication**

One of the main purposes of the CMI application is to connect to TCC server in order to exchange data over TCP/IP socket. TCP/IP is complement suite of the Transmission Control Protocol and Internet Protocol, providing a reliable ordered delivery of data between computer applications on internet. It brings the requirement for the target framework to support TCP/IP socket, application end-point for the communication over internet using TCP/IP protocols.

Today in a world of connected computers is becoming more and more difficult to avoid the need of communication in software applications. Therefore it is necessary to every common programming language provides an API for socket communication.

Java facilitates socket communication through java.net package in which ServerSocket and Socket classes are providing properties and methods to access and manipulate socket for server and client applications, respectively. ServerSocket functionality is to wait for client request over the network. Upon receiving such a request will process the request by performing required operations and providing responses. The Socket functionality is to try to initialize the communication by sending connection request to server and then receives its response and so on [31].

C# programming language provides the socket communication through System.Net and System.Net.Sockets namespaces, Socket class provides methods to create client and server sides' applications. Server program is listening to clients request using an instance of TCPListener class and client program connects to server using an instance of TCPClient class [32].

C++ provides socket communication through C++ libraries, TCP/IP Toolkit for C++. There are different available sources of these libraries and most of them are third-party libraries, so there is the probability that is choosing suitable platforms. There are a variety of solutions to use networking in C++, but they depend on the type, flexibility and performance of the specified library. And most of them are platform dependent. The most suitable platform that support C++ libraries for network programming is Microsoft Windows platforms which can be used with a range of programming languages including C and C++ [33][34].

### **6.3.4 XML Parsing**

As explained earlier, one of the objectives of this thesis is to design and implement the communication protocol between CMI application and TCC server. The communication protocol is an XML-based protocol or using XML-formatted messages. It requires that both sides of the communication be able to do XML parsing.

Java provides several different libraries for applications in order to process XML documents or messages, such as SAX; the Simple API for XML, DOM; the Document Object Model API from W3C, XSLT; the XML Style Sheet Language Transformations from W3C and etc.

The most common of them are SAX and DOM. DOM parser creates a tree structure of the XML source document in memory which enables random access to the arbitrary nodes of the DOM tree. SAX parser works with event-driven fashion, tokens of XML document are caught and thrown by the parser as events to be handled [35].

The .NET framework provides a wide variety of API options for reading and writing XML documents which may offer different efficiency and productivity. To process XML documents, C#.NET programming interface provides System.Xml namespace which is built on key XML industry standards, such as DOM, XPath, XSLT, XML Schemas (XSD) and etc. According to .NET framework, processing XML document is including several steps or layers, as shown in the following figure [36].

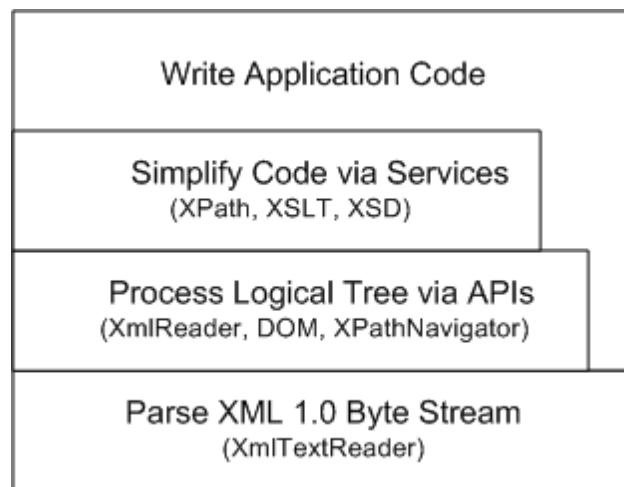


Figure 6.9 XML Processing Layers in .NET [36]

XmlTextReader provides XML 1.0 byte stream functionality which is the most basic step in processing XML documents. After this step, the higher layers are presenting XML APIs choices in order to process the XML documents by treating them as logical tree structures. The APIs performs with two main strategies, streaming and traversal oriented. The most common API for streaming strategy is SAX, but because of the difficulties with SAX-based XML processing, it has been alternated by Microsoft with simpler and more intuitive streaming API through the

XmlReader class library. XML schema and DTD validation facility is also provided through XmlValidatingReader class, which can be used with XmlReader implementation including XmlTextReader. Provided APIs for traversal-oriented strategy by .NET frameworks are including Document Object Model or DOM and XPathNavigator. DOM is the most common traversal-oriented API and is provided through the XmlNode class hierarchy. XPathNavigator API gives the Possibility to traverse XML logical tree using a cursor model [36].

The XML C++ class generator creates source files from an XML DTD or XML Schema. The class generator takes the Document Type Definition (DTD) or the XML Schema, and generates classes for each defined element. Those classes are then used in a C++ program to construct XML documents conforming to the DTD.

This is useful when an application wants to send an XML message to another application based on an agreed-upon DTD or XML Schema, or as the back end of a Web form to construct an XML document. Using these classes, C++ applications can construct, validate, and print XML documents that comply with the input.

The class generator works in conjunction with the Oracle XML parser for C++, which parses the input and passes the parsed document to the class generator [37].

### **6.3.5 Database Connectivity**

Besides providing the user interface, another facility of the CMI application is Alarm Handling. Alarms are read from and manipulated in the database, where they may be placed by any subsystem. In order to accomplish this facility the CMI application target framework is required to support database connectivity for today common databases particularly MySQL.

In Java platform, database connection facility is provided by Java Database Connectivity or JDBC API through java.sql and javax.sql packages. JDBC makes it possible for Java programs to virtually connect to any database such as MySQL, Oracle, postgresql, JavaDB and etc, and then

manipulates data using SQL queries. The only challenge left is to connect with appropriate connection string [38].

In C# applications, database connectivity is performed by a set of components called ADO.NET provided by Microsoft.NET framework. ADO.NET classes are contained in System.data namespace, including System.Data.SqlClient, System.Data.Odbc, System.Data.OleDb and System.Data.Oracle which are used to connect to different databases. System.Data.Odbc class is used to communicate with the Sql Server database, System.Data.SqlClient class is used to connect and manipulate MySQL databases, System.Data.OleDb class is used to perform operations on the Access Database and System.Data.Oracle class is used to perform operations on the Oracle database. Another requirement for C# applications to connect to mysql database is a small program called mysql connector .net which can be found in MySQL official website [39].

MySQL Connector/C++ is one of the latest connectors for MySQL, developed by Sun Microsystems. The MySQL connector for C++ provides an object-oriented application programming interface (API) and a database driver for connecting C++ applications to the MySQL Server.

The development of Connector/C++ took a different approach compared to the existing drivers for C++ by implementing the JDBC API in C++ world. In other words, Connector/C++ driver's interface is mostly based on Java programming language's JDBC API. Java Database Connectivity (JDBC) API is the industry standard for connectivity between the Java programming language and a wide range of databases. Connector/C++ implemented a significant percentage of the JDBC 4.0 specification. C++ application developers who are familiar with JDBC programming may find this useful, and as a result, it could improve application development time [23].

### **6.3.6 Portability**

As mentioned earlier, one of the most significant advantages of Java is portability, provided by its virtual machine which takes care of the complexities between Java applications and underlying

operating system regarding to the type of operating system. C# source code converts to a common intermediate language, CIL or MSIL which can be performed on platforms supporting Common Language Infrastructure, such as the .NET runtime on Windows, or the cross- platform using Mono runtime software.

C++'s approach to portability is different. On the one hand, it's a compiled language, and those binaries are almost always platform specific. So C++ executable will never be portable (unlike Java). On the other hand, porting the compiler can sometimes be enough. The community has found that by porting the compiler as well as some core libraries of the language, **source codes** (and not binaries) could be portable.

However, C++ is widely used in critical systems like compilers, kernels, real-time systems, embedded systems. There's a "low level" aspect of C++ that cannot be overlooked, when talking about portability [40].

### **6.3.7 Ease of Use for Developer**

Providing facilities such as object oriented features, automatic memory management, intuitive set of well-designed libraries, reusable codes, high level of abstraction hiding the platform-dependency complexities besides plenty of available resources makes Java programming language an elegant and easy target for developers in order to be more efficient and productive.

C# programming language is an object-oriented language which compasses a bit more complex features than Java. It is facilitating automatic memory management, high level of abstraction, intuitive and comprehensive set of libraries, very expressive syntax which makes it easy to learn and improves the developers' productivity [41]

The newest versions of C++ are easier than the old ones on the issue of easiness for developers. But compared with Java and C#, C++ is much harder for developers

### **6.3.8 Widely Common In Industry**

According to the ranking provided by The LangPop Community index in October 2014, Java is the most popular programming language with more than 20% usage and C++ with more than 10% is in the fourth rank among 20 first most popular languages and C# with more than 5% from other among first 50 programming languages. It is remarkable that this ranking is based on the number of skilled engineers worldwide, courses and third party vendors [41].

I pointed out the efforts to make C++ competing among other available programming languages. One of the most major efforts was the effort to make C++ libraries compliant with common language specification (CLS) of .NET framework and also enhancing libraries by adding new programming features such as internationalization, portability and etc. to make it comparable with Java or C#. Even if lacking all these efforts, according to the programming languages popularity ranking, C++ is much better popularity than C#.NET [40].

### **6.3.9 Maturity**

There are some attributes that are predicated for a programming language as a mature programming language. For instance the programming language should be in public domain, it should be supported by a community such as forums. It should be used by several groups to develop enterprise successful projects, and there should be set of stable libraries or APIs and several other attributes which according to these attributes all of the candidate frameworks are mature [42].

### **6.3.10 Maintainability in the future**

Software maintenance is group of efforts to improve the software functionality, such as debugging, improving performance, modifying current functionalities, adding new functionalities and so on. Maintainability is not a standard and can be defined by relatively different factors. Some of common factors counted as properties of maintainable code are; clarity or readability of the code, modularity, being easy to understand, less complexity, flexible in front of new changes and etc.[43].

Considering the factors that improves Software maintainability and Java attributes, such as object oriented paradigm which improves the modularity and understanding the code, Garbage Collector which decreases the complexity and probable defect and etc. I can conclude that Java is one of good option to have maintainable application.

The conditions are pretty much verifiable for C# programming language, even better by having a very expressive syntax which improves readability of the code significantly.

C++ besides providing facilities to reuse codes, by minimizing the designing time and maximizing the development time decreases the maintenance costs.

Maintainability of an application developed in a specific programming language is affected by popularity of the programming language in industry. In one hand as explained earlier, there have been significant efforts in upgrading C++ in order to keep it in track versus other popular technologies. In the other hand there are lots of efforts encouraging industries to migrate their C++ applications to state-of-the-art and reliable environment such as Java, VB.NET and ASP.NET, to decrease the risk of getting pushed out the market completely by modern technologies.

The result of programming languages popularity ranking regarding C++ shows that more people know or learn about this language than C# which may good to maintainability issues for C++ applications in future.

## **6.4 Conclusion**

The requirement analysis provided in this chapter enable me to conclude with one of these candidates as the target framework. To accomplish the final evaluation I make a summary of the result for each candidate framework as shown in Table 6.1.

Java framework fulfills the requirements specification. Java is fully supporting application internationalization and localization facilities. Socket communication is supported through Socket and Server Socket class hierarchies in Java API. It is providing several APIs to process XML documents including Java DOM, SAX and StAX interfaces. Java Database Connectivity or JDBC programming interface enables java applications to connect to any database including MySQL.

Integration with Adobe Flex component is possible through several solutions which the appropriate one for the case of this application is to embed compiled or SWF file of Flex component within Java component. Java is portable, maintainable, mature and widely used in industry. Java providing object oriented features, high level of abstraction and etc. is an easy choice to for developers.

C#.NET meets the requirements specification. .NET framework provides internationalization and localization facility through Globalization API. It supports socket communication through System.Net and System.Net.Socket libraries. It provides comprehensive set of API for XML processing through System.XML namespace. Connectivity to MySQL database in .NET framework is offered through System.Data.SqlClient class hierarchy. The integration with Adobe Flex component is possible by embedding SWF within Windows Form application as well. C# applications can be cross- platform using Mono runtime software. It is maintainable, mature and widely used in software industry projects.

C++ meets the requirement specifications as well. C++ facilitates internationalization and localization through translation toolkit and Unicode support. It supports socket communication through C++ libraries, TCP/IP Toolkit for C++ software component. C++/DOM application programming interface provides XML processing for C++ applications. C++ applications can connect to MYSQL database using MySQL C++ connector connection. It is may be possible to integrate Flex component with C++ through loading Flex executable or SWF file in Internet Explorer control embedded within C++ window. Applications developed in C++ are mostly only supported by Microsoft Windows platforms. It has not been commonly involved in industrial projects recently. It is mature programming language but since it has not been state-of-the-art in recent years, it may affect the maintainability.

The final conclusion I could come up with, considering all the requirements analysis, C++ is not a suitable choice for the target framework since it doesn't fulfill some of the requirements. It is not cross-platform and widely used in industry, the maintainability is with difficulties and the integration with Flex is not guaranteed. After that, between Java and C#.NET frameworks, both of them seem appropriate for my purpose. Considering and comparing more precisely indicates that

portability and connection to MySQL database for C#.NET is not as straight forward as Java which made me to pick Java SE as the most suitable framework.

Requirements	Java	C#.NET	C++
Meet Requirements Specification	Yes	Yes	Yes
Support for Local Languages	Yes, Using Java Internationalization Java.Text.Message-Format Library	Yes, Using System.Globalization. CultureInfo Library	May be, by using different Libraries it can be possible
Support for Socket Communication	Yes, Using java.net.Socket and java.net.ServerSocket class hierarchies	Yes, Using System.Net.Socket class hierarchy	Yes, Using TCP/IP Toolkit for C++ software component, but it is complex
XML Parsing	Yes, Using Java DOM SAX, StAX libraries	Yes, Using System.XML namespace	Yes, Using C++ XML class generators ( xml DTD or xml schema)
Database Connectivity (MYSQL)	Yes, Using JDBC library	Yes, Using System.Data.SqlClient class hierarchy provided by ADO.NET	Yes, Using MySQL Connector/C++, but it is complex
Portability	Yes	Yes, Using Mono runtime software	No, it is platform dependent
Ease of Use for Developer	Yes, providing object oriented features and high-level of abstraction	Yes, providing object oriented features and high-level of abstraction	Yes, providing object oriented features and high-level of abstraction
Widely Common in Industry	Yes	Yes	Yes
Maturity	Yes	Yes	Yes
Maintainability in Future	Yes	Yes	Yes

Table 6.1. Summary of Framework Evaluation

# 7. GUI Design

## 7.1 Overview

This chapter describes the process of the GUI application design according to the second objective of the thesis within two different sections. The first part is dedicated to overall design of the GUI application, considering different modules. The second part describes the design process of the interface of GUI to CTC server based on the latest version of the communication protocol using XML formatted messages.

## 7.2 Application Design Process

This section is dedicated to the Graphical User Interface design process according to the requirements specification. The requirements specification has been considered during the first phase of the thesis in order to accomplish the framework evaluation.

### 7.2.1 Design Theories and Methodology

As I mentioned earlier, at the beginning I had to consider the requirement specifications which to design this application by adding local language feature. One of my objectives is to design a graphical user interface (GUI) that can has new feature and the interaction between the users and the system in the sense of being user friendly. The steps involved within the design process I followed to achieve this goal are explained here as indicated in Figure 3.1.

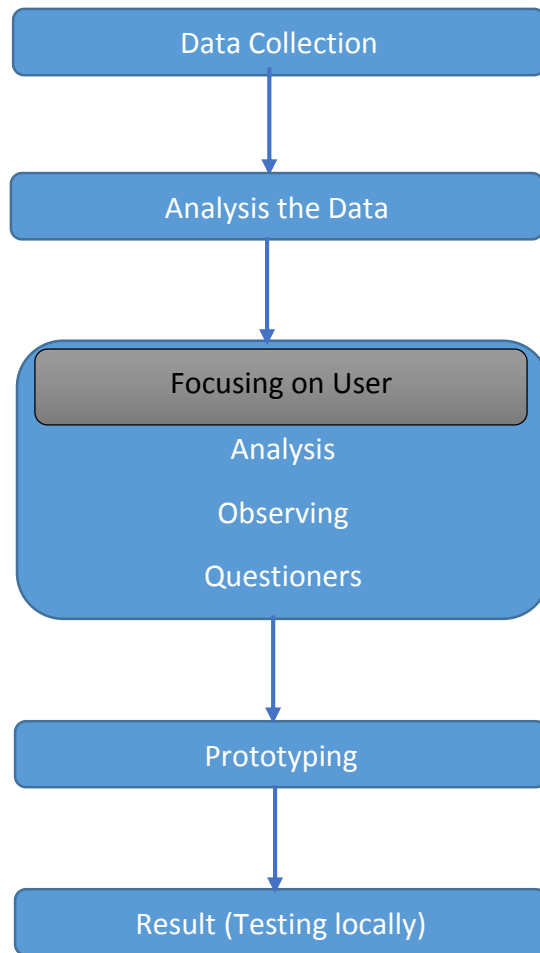


Figure 7.1 Design Process

### 7.2.1.1 Data Collection

As I mentioned earlier at the beginning of this thesis the project is supervise and monitor by the Clint (ERC) and the design and system installation and configuration work by CREC. Therefore options to get necessary data was:-

- Asking specifications and other related issues of the new railway system to CREC staffs by a latter:- this was done first writing a letter to ERC's responsible staff, then after some process the office wrote another letter for the responsible CREC's department. But due to unknown reason they didn't have willing to give detail information. Although, by finding contract agreement document I have got necessary information.

- Gathering information from previous CREC railway projects in different countries:- the company was built different railway projects in different countries Saudi Arabia and Serilanka are the resent ones.
- Browsing from CREC website
- Based on the above specifications search and evaluate candidate frameworks

According to the CREC contract agreement with ERC, CATS (Central Automatic Train Supervision) application server software runs on the CATS application servers with **Windows Server 2008 operating system**. The CATS software is coded with C/C++ language.

CATS' application server is the data process center of ATS sub-system. After it obtains stations, depot/stabling yard and external system data, it will send track layout display information, alarm and train state to ATS (Automatic Train Supervision) workstation and display panel. It takes the responsibility of train schedule relative event. It processes the operation from the dispatcher, maintainer and staff in the depot/stabling yard and sends the information into relevant system. It stores all kinds of data in the system's daily running for later analysis and playback. CATS' application server also takes the responsibility of sending relevant information to radio system and other external systems [12]. Therefore, based on this document I started my analysis.

### 7.2.1.2 Analysis of the Data

After the data collection phase I had been able to define the related data for each part and analyze the relations between them and gather the data for my research. The first part to analyze the data was gathering information about the collected frameworks' in the context of railway traffic control. On the second part , as I mentioned earlier the objective of this research paper is to recommend a good GUI for ERC by selecting appropriate frameworks to control train traffic. Therefore, based on gathered data and following international standard requirements, in the framework evaluation process part (Analysis part), I made my analysis to select the most preferable framework. During the process I tried to focus on implementation of newly designed GUI with the existing platforms.

The application will have different additional features without any change of existing GUI system and these are:-

- Not necessary installing in all dispatcher work PCs
- Only installed on the server and can be accessed by all dispatcher work PCs
- Display all necessary information in local languages
- Easy to be advanced by different developers
- Working on different platforms or operating systems
- Etc.....

### **7.2.1.3 Focusing on Users**

The application must be understandable for all these different kind of users, the main focus was on Controllers and servers and as I did not have access to them. I did the observations of framework on Addis Ababa LRT Project and previous company projects those are finalized and on operation. This phase had been held in a circle of analysis, observations and questioners in order to define the things that I needed to avoid and the functions that I needed to focus more.

### **7.2.1.4 Prototyping**

When I finished doing the previous phase, it was time to come up with some solutions through the brainstorming. And the next step is to design prototypes. Designing different prototypes would lead me to get more close to the final solution. I started to first writing a code on **NetBeans 8.0** to draw a piece of lines and then by using Java built-in functions I did a detailed track layout, trains, signals, track fault indicators, database connectivity indicator etc. design.

The track layout design was taken from the exact layout of Kality Depot as shown in Figure 7.2.



#### 7.2.1.4.1 Discussion

The application has four different classes. Each class has their own tasks for instance:-

- Class ***RailwayTrackLayout*** : the task of this class is to draw the layout, representing the train, signals, database connectivity indicator, fault indicators on the screen. It has more than 45 methods or functions. Each functions has specific jobs to fulfil the operation of GUI during runtime process. When run or execute this class or when it is called by the main method, which is in another class called ***DisplayTrackLayout***.

#### Algorism

- First it check the database connection
  - ✚ If it is connected, according to retrieved data from the database it display every information
  - ✚ If it is not it simply display track layout, fault indicators and database connectivity indicator with red color
- Assume the database connection is good, retrieve database data
- Indicate the trains, track status, database connection status, signal status and track layout on specified position of the screen
- Wait until the database is updated
- Retrieve the new parameters
- Repeat

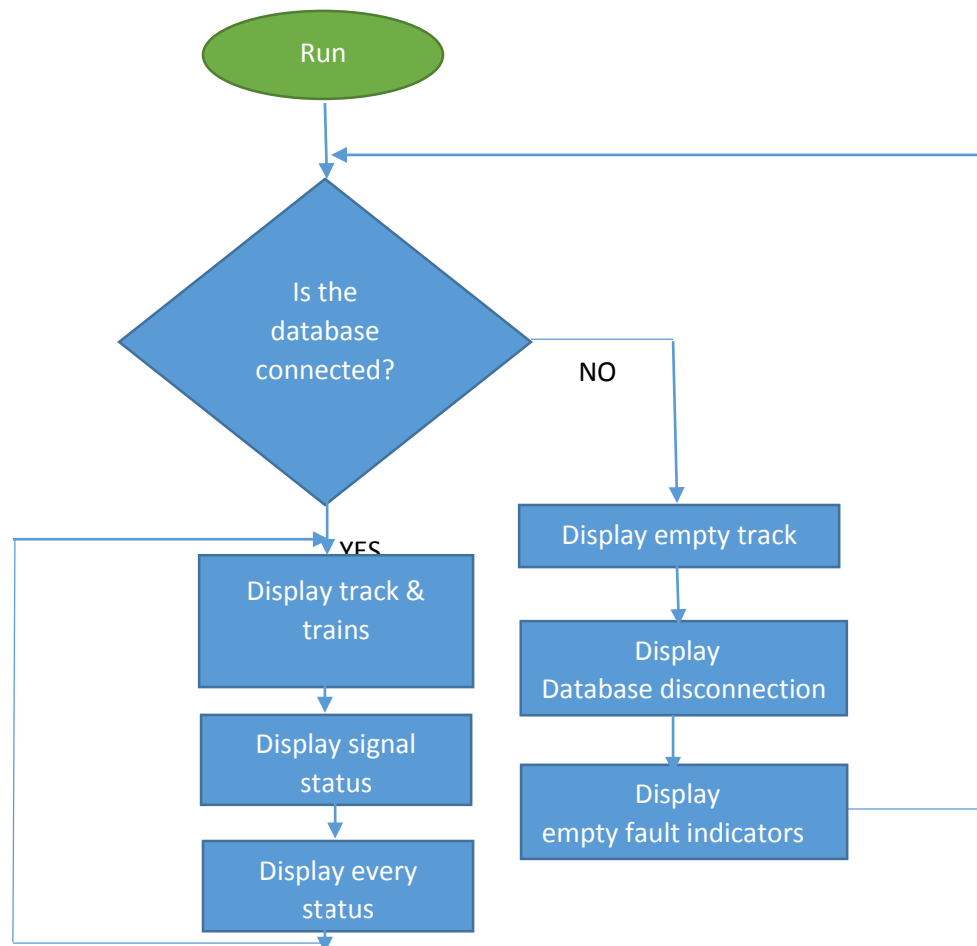
Flow chart for *RailwayTrackLayout*

Figure 7.3 Railway track layout class flow chart

- Class ***CurrentInformationUpdater***: this class is responsible to update the database to display stored data in local languages by other class.

**Algorism**

- Check database connectivity
- Retrieve data from database specified tables
- Store data in another table in local language
- Repeat

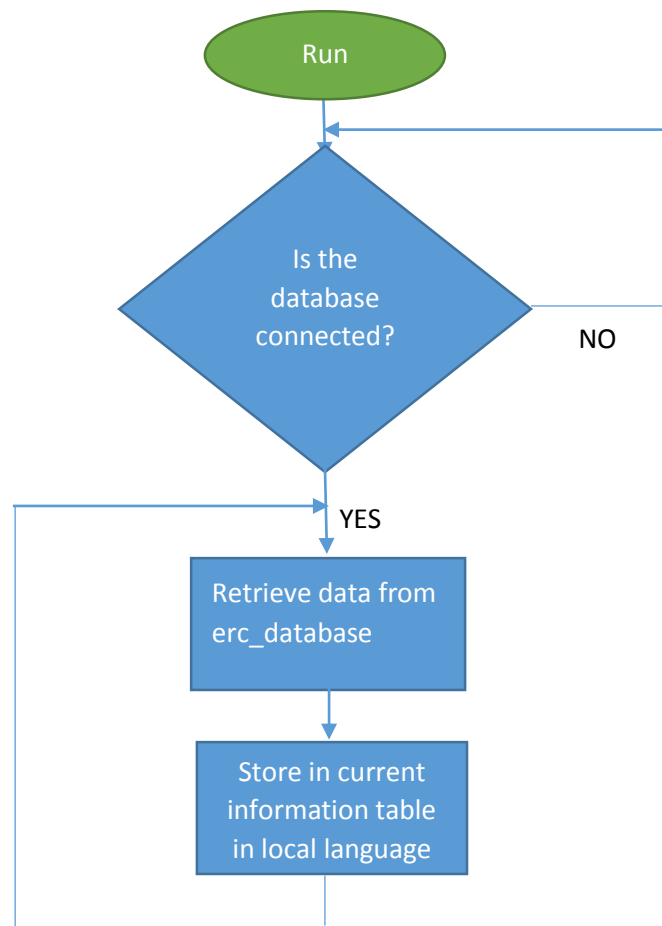
Flow chart for *CurrentInformationUpdater*

Figure 7.4 Current information updater class flow chart

- Class ***TrainInformation*** : displaying of current information of every information in the table format will be done by this class

**Algorism**

- Check database connectivity
- Retrieve data from database *Current Information* table
- Display retrieved data in local language
- Repeat

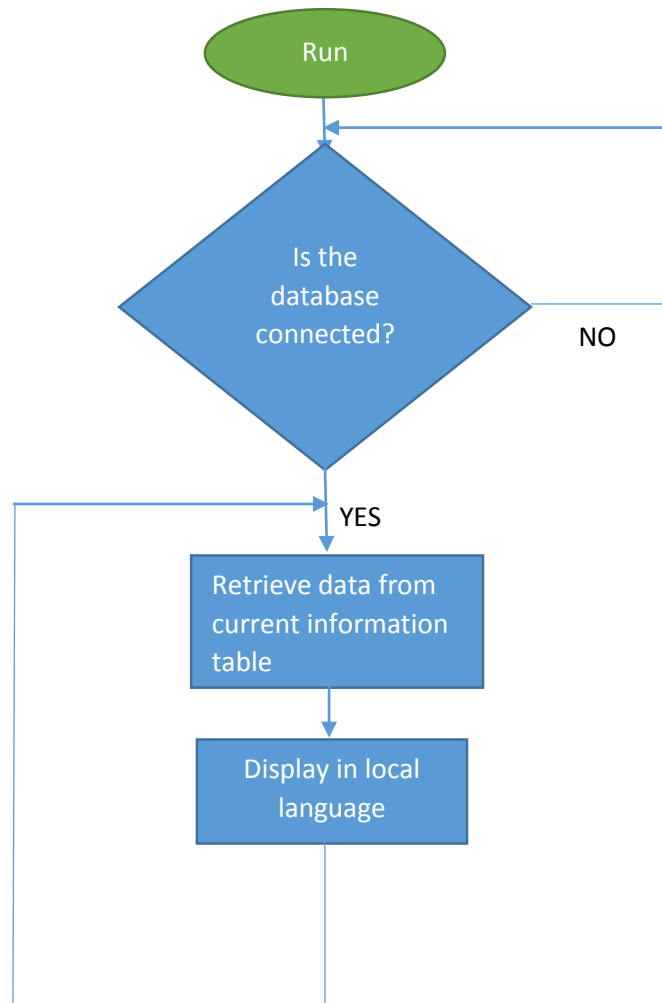
Flow chart for *TrainInformation*

Figure 7.5 Train information class flow chart

- Class *DisplayTrackLayout* : this class has a main method of *RailwayTrackLayout* to display the GUI.

**The detail prototype application is included in the CD!!!**

### 7.2.1.5 Result (Testing locally)

Testing of the application was done on local server. And it is called EasyPHP DeveServer. It is a WAMP package including the server-side scripting language PHP, the web server Apache, the SQL server MySQL, as well as development tools such as the database manager PhpMyAdmin, the debugger Xdebug and many others. Nothing to configure. It's already done! It is just need to download, install ... and code. The administration page allows to list the doc root, extensions, change the Apache port, the time zone, max execution time, error reporting, upload max file size, add/remove alias, manage modules etc ....

By using this powerful application, NetBeans 8.0 and Java SE Development kit 8.0 I successfully tested and analyzed it. Netbeans IDE provides a GUI building feature that enables user to create graphical user interface by dragging and positioning the components from a palette containing AWT/Swing components on to a canvas. Indeed using GUI Builder components with the default setting are added to the interface and the user can easily edit the properties later on. Besides GUI Builder takes care of the alignments and spacing of the components which is complicated to set by manually coding [45].

As the figures shown below during execution of the program the GUI display its component according to the data stored in the database.



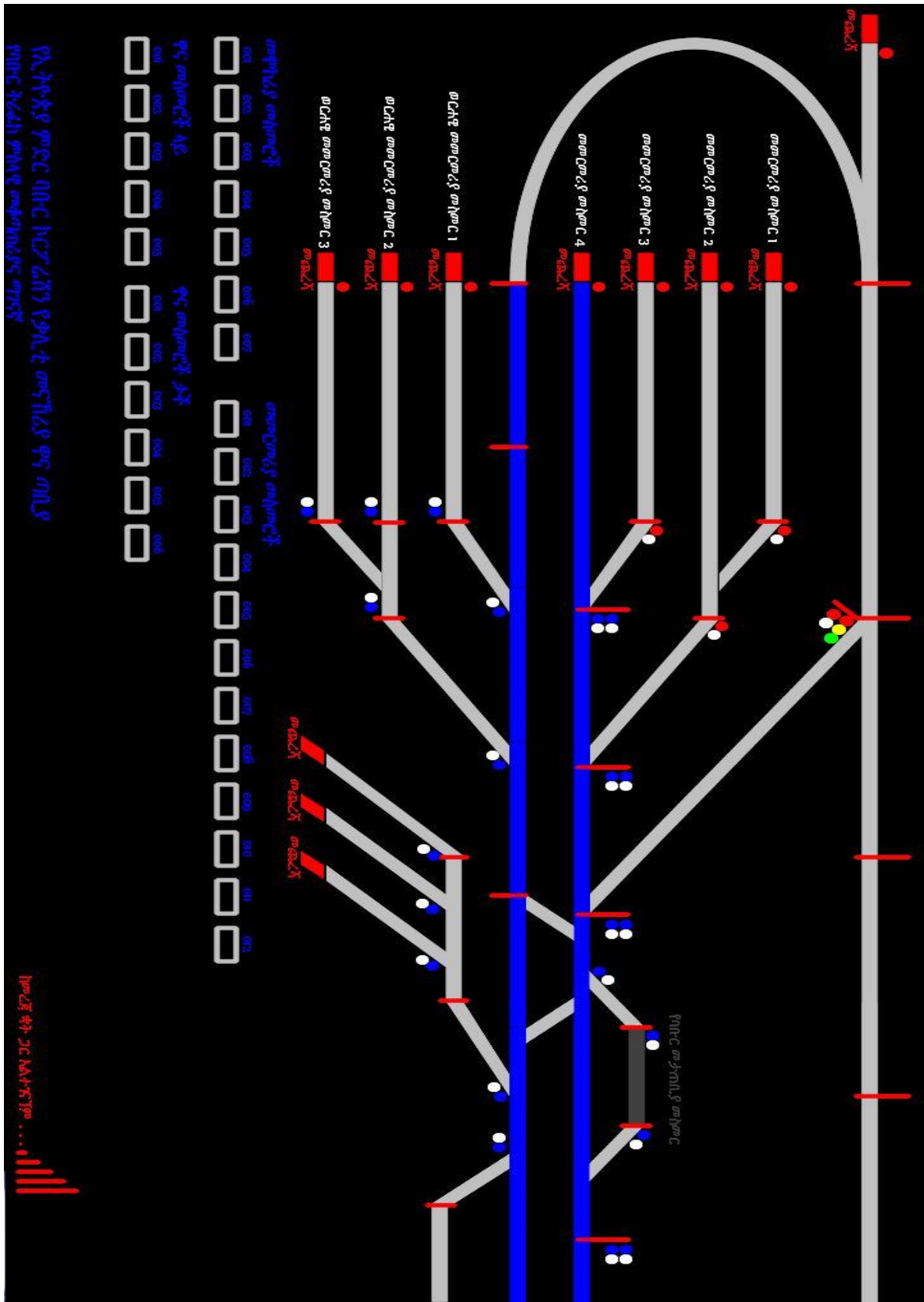


Figure 7.7 Execution result when the database connection doesn't exist

id	Name	Position	Track	Speed	Start	Destination	Train_Status	Current_Track	Track_Status
1	ባቡር 001	መሐጃ ዋና መስመር 005	መሐጃ ዋና መስመር	10 ኪሎ.ሜ/በሰአት	መሐጃ ዋና መስመር 001	ከቃሊቲ መናኸሪያ ለመውጣት	ባቡር ጥሩ ሁኔታ ላይ ነው!	መሐጃ ዋና መስመር 005	jhwkjhvj
2	ባቡር 002	መሞከሪያ መስመር 001	መሞከሪያ መስመር	15 ኪሎ.ሜ/በሰአት	መሞከሪያ መስመር 001	ለሙከራ	ባቡር ጥሩ ሁኔታ ላይ ነው!	መሞከሪያ መስመር 001	kjhcbkjZ
4	ባቡር 004	መመርመሪያ መስመር 001	መመርመሪያ መስመር	10 ኪሎ.ሜ/በሰአት	መመርመሪያ መስመር 001	መመርመሪያ መስመር	ባቡር ጥሩ ሁኔታ ላይ ነው!	መመርመሪያ መስመር 001	HBUKUIKUIVKUI
5	ባቡር 005	መመርመሪያ መስመር 004	መመርመሪያ መስመር	10 ኪሎ.ሜ/በሰአት	መመርመሪያ መስመር 004	መመርመሪያ መስመር	ባቡር ጥሩ ሁኔታ ላይ ነው!	መመርመሪያ መስመር 004	UYGFUYFUYFIU
6	ባቡር 006	መመርመሪያ መስመር 003	መመርመሪያ መስመር	10 ኪሎ.ሜ/በሰአት	መመርመሪያ መስመር 003	መመርመሪያ መስመር	ባቡር ጥሩ ሁኔታ ላይ ነው!	መመርመሪያ መስመር 003	JBKLJNLKLN
7	ባቡር 007	መመለሻ ዋና መስመር 001	መመለሻ ዋና መስመር	10 ኪሎ.ሜ/በሰአት	መመለሻ ዋና መስመር	መመለሻ ዋና መስመር መጨረሻ	ባቡር ጥሩ ሁኔታ ላይ ነው!	መመለሻ ዋና መስመር 001	JNLKLNKL
8	ባቡር 008	መመለሻ ዋና መስመር 003	መመለሻ ዋና መስመር	10 ኪሎ.ሜ/በሰአት	መመለሻ ዋና መስመር	ለሙከራ	ባቡር ጥሩ ሁኔታ ላይ ነው!	መመለሻ ዋና መስመር 003	JNLKLNKL
9	ባቡር 009	ባቡር መታጠቢያ መስመር ላይ	ባቡር መታጠቢያ መስመር ላይ	10 ኪሎ.ሜ/በሰአት	መመለሻ ዋና መስመር	መመለሻ ዋና መስመር መጨረሻ	ባቡር ጥሩ ሁኔታ ላይ ነው!	ባቡር መታጠቢያ መስመር ላይ	JNLKLNKL
10	ባቡር 010	መመርመሪያ መስመር 010	መመርመሪያ መስመር 010	10 ኪሎ.ሜ/በሰአት	መመርመሪያ መስመር 006	መመርመሪያ መስመር መጨረሻ	ባቡር ጥሩ ሁኔታ ላይ ነው!	መመርመሪያ መስመር 010	JNLKLNKL

Figure 7.8 (Pictured by snipping tool) Execution result and updated database information

### Track Layout

It is supposed to represent graphic image of the actual track layout. The components of track layout are including derailleurs, detectors, points, tracks, locations and trains.

### Acknowledge Alarm

It displays a list of unacknowledged alarms with their properties, as stored in the database. User will be able to acknowledge any alarm that selects.

### Locations

Information table list window displays a list of all location with their properties, as available in the track layout. The locations are supposed to be read from the CTC database (see Fig 7.7).

## **Trains**

Trains are displayed as a small colored circle and all of them are registered with their information including name or identity, status and destination. CTC receives this information from the database. The status of each train can be identified through different colors Java support more than 50 colors combinations (see Fig. 7.6, 7.7 & 7.8).

## **Runtime Speed**

Runtime Speed list of the trains will be displayed with local language description, as they have been set in the application. CTC also receives this information from the database (see Fig. 7.8).

## **Location Information**

This information is displayed based on the location of individual trains. The data will be retrieved from CTC database. The axel counter is the main sources of information signal which is stored in the database. Therefore, the position of every train will be well displayed if the database connection is exist (see Fig. 7.8).

## **Track Information**

Due to different reason the track may damage or disconnected track circuit such that to indicate any fault on the dispatcher screen it use different colors as well as displaying the description on the table (see Fig. 7.6, 7.7 & 7.8).

## 8. Conclusion and Recommendation

### Conclusion

I have tested the application locally using MySQL and Netbeans Java framework as the most suitable target framework. Considering all the requirements analysis provided in chapter 2, C++ was not a suitable choice for the target framework since it doesn't fulfill some of the requirements, such as portability, internationalization and complexity of database connectivity and socket communication. Furthermore, more precise considering and comparing between Java and C#.NET frameworks indicated that portability and connection to MySQL database using Java is pretty straighter forward than using C#.NET framework.

During design process of this application, I tried to encompass the login security using PHP. The login system will prevent the user according to their level of access of the GUI. Based on the standard design process the paper also divide the application classes & methods in to different modules. Furthermore within each module there have been some divisions based on the functionality and class hierarchies have been designed according to Object-Oriented design principles.

As I stated earlier the most challenging part of this thesis was getting the information about the new ERC railway system GUI. But, the company, which is responsible to install and configure this system, has experience in different countries. Therefore, based on its previous projects and current contract agreement document between ERC and CREC, I tried to gather the necessary data to fulfil this document.

Meanwhile, this application is a prototype application which will help for local developers to make it advance and knocking a door to prepare GUI for train traffic control using our languages. To do this NetBeans IDE is a powerful software and it provides a GUI building feature that enables user to create graphical user interface by dragging and positioning the components from a palette containing AWT/Swing components on to a canvas. Indeed using GUI Builder components with the default setting are added to the interface and the user can easily edit the properties later on [45].

## Recommendation

➤ I have seen from this thesis paper preparation process the framework which used by the new implemented railway system GUI is C++ programming language. As followed by standard framework evaluation process for GUI design C++ is the least candidate than Java and C#.NET. The most important issues that less competitiveness of this framework was:-

- Internationalization (using local languages)
- Portability (Platform independence)
- Less complexity of socket communication
- Less complexity of database communication
- Easiness for local developers

Therefore, I recommend that the new system of Ethiopian Railway Corporation GUI for train traffic control to use a Java framework.

- As I stated on candidate framework evaluation process, it is very easy than other candidates to integrate Java with Adobe Flex to make the GUI response fast and see in SWF format. Such that, in the future if the system use java framework I recommend in the future to design and work on integrating with this powerful software environment.
- The design process was focused on internationalization, which is capable to use in different local languages, such that, the idea will encourage local developers to add different features. Therefore, I recommend that, by inviting different skillful students and other interested local sectors, ERC to work on this advanced area.

# References

- [1] *ERC*, Construction agreements
- [2] *Aravid Kauppi*, “A Human-Computer Interaction Approach to Train Traffic Control”, Division of human-computer interaction department of information technology Uppsala University ,Uppsala Sweden
- [3] “.NET Framework Conceptual Overview”, Microsoft, 2011,  
<http://msdn.microsoft.com/en-us/library/zw4w595w.aspx>
- [4] *Tetsuo Takashige*, edited by *Kanji Wako*, “Railway Technology today, and Signalling System for Safe Railway Transport”, Japan Railway Transport Review September, 1999
- [5] *M. Pastore*, “Java Developers Finding a Home at Adobe Flex”, *QuinStreet Inc.*, November 2009 <http://www.devx.com/HotList/HotList-Adobe/Article/42707>
- [6] *Aguado, M.; Onandi, O.; Agustin, P.S.; Higuero, M. & Jacob Taquet, E.* “A Broadband Communication Architecture for CBTC Systems”, *IEEE Vehicular Technology Magazine*, Vol. 3, No. 3, pp. 47-56, ISSN: 1556-6072, Nov 2008.
- [7] *Webopedia*, “Graphical User Interface” Available HTTP:  
[http://www.webopedia.com/TERM/G/Graphical\\_User\\_Interface\\_GUI.html](http://www.webopedia.com/TERM/G/Graphical_User_Interface_GUI.html)
- [8] *Wenzel-MACS-R* “Dispatcher System for GSM-R” *WENZEL Elektronik Vehicular Technology Magazine*, Germany, 2012.
- [9] *What is? com*, “Graphical User Interface” Available HTTP:  
[http://whatis.techtarget.com/definition/0,289893,sid\\_gci213989,00.html](http://whatis.techtarget.com/definition/0,289893,sid_gci213989,00.html)
- [10] *B. A. Myers*, “User Interface Software Tools”, Computer Science Department, CMU University, Pittsburgh, August 1994
- [11] *Microsoft*, “MDI Applications in Visual C++”, *Visual Studio 6.0*,  
<https://msdn.microsoft.com/en-us/library/aa733747%28v=vs.60%29.aspx>
- [12] *ERC vs CREC agreement document* “Technical Signaling Specification” Chapter 4 Detail Design 2 ATS Subsystem Document
- [14] *A.Dhar*, “Socket Programming in C#”, July 2003,  
<http://www.devarticles.com/c/a/CSharp/Socket-Programming-in-C-Part-I/>

- [15] **Dave Wood**, “Java technology trends offer renewed promise for portable embedded applications”, Software Engineering Institute, Carnegie Mellon University Sandiego 2007
- [16] **Li Chen & Qinghua Lu**, “Developing Real-time Applications with Java Based Sun SPOT” School of Computer Science and Software Engineering University of Wollongong, Australia, 2007
- [17] **Wikipedia**, “Java Programming Language”, March, 2014,  
[http://en.wikipedia.org/wiki/Java\\_version\\_history#Java\\_SE\\_8\\_.28March\\_18.2C\\_2014.29](http://en.wikipedia.org/wiki/Java_version_history#Java_SE_8_.28March_18.2C_2014.29)
- [18] **D. Harms**, “Industry Trends: How Important Is Java”, *Clarion Magazine*, September 1999,  
<http://www.clarionmag.com/cmag/v1/v1n8understandingjava.html>
- [19] **K. Radeck** “C# and Java: Comparing Programming Languages”, Microsoft, October 2003, <http://msdn.microsoft.com/en-us/library/ms836794.aspx>
- [20] **Microsoft**, “Introduction to the C# Language and the .NET Framework”, 2011  
<http://msdn.microsoft.com/en-us/library/z1zx9t92.aspx>
- [21] **Pooja Chawla Vetter & Prof. Dharminder Kumar** “OOP with C++”  
<http://www.ddegjust.ac.in/studymaterial/mca-3/ms-17.pdf>
- [22] **Infoq.com** “C++14 Is Here: Summary of New Features” August 23 2014  
<http://www.infoq.com/news/2014/08/cpp14-here-features>
- [23] **E Balagurusamy** “Object –Oriented –Programming in C++”
- [24] **Open Standard Organization**, “Standard for Programming Language C++”, October 2013,  
<http://www.open-std.org/jtc1/sc22/wg21/docs/papers/2013/n3797.pdf>
- [25] **M.Hall**, “InternalFrames”, 1999,  
<http://www.apl.jhu.edu/~hall/java/Swing-Tutorial/Swing-Tutorial-JInternalFrame.html>
- [26] **G. Gnana Arun Ganesh**, “Developing MDI Application in C#”, January 2012,  
<http://www.csharpcorner.com/UploadFile/ggaganesh/DevelopingMDIApplicationsinCSharp11272005225843PM/DevelopingMDIApplicationsinCSharp.aspx>
- [27] **S. Rodriguez**, “C# SDI/MDI Application Wizards”, November 2013,  
<http://www.codeproject.com/KB/macros/sdimdiwizards.aspx>
- [28] **Microsoft** “MDI Applications in Visual C++”, *Visual Studio 6.0*,  
<https://msdn.microsoft.com/en-us/library/aa733747%28v=vs.60%29.aspx>

- [29] “Java Internationalisation”, *Oracle Corporation*, 2013,  
<http://java.sun.com/javase/technologies/core/basic/intl/>
- [30] **Kumar, Ravikant**, “Globalization, Internationalization, and Localization using C# and .NET2.0”, November, 2009, <http://www.codeproject.com/KB/locale/Internationalization.aspx>
- [31] “International Component for Unicode”, *ICU*, 2014, <http://site.icu-project.org/>
- [32] **A.P.Rajshekhhar**, “Socket Programming in Java”, April 2012,  
<http://www.devarticles.com/c/a/Java/Socket-Programming-in-Java/>
- [33] **A.Dhar**, “Socket Programming in C#”, *Socket-Programming-in-C-Part-I/* July 2003,  
<http://www.devarticles.com/c/a/CSharp/>
- [34] “The PowerSocket Library”, January 2011,  
<http://www.level5software.net/documents/Pslib21.htm>
- [35] “SocketWrench Freeware Edition”, *Catalyst Development Corporation*, 2011,  
<http://www.catalyst.com/products/socketwrench/freeware/index.html>
- [36] **T. Violleau**, “Java Technology and XML”, *Oracle Corporation*, November 2012,  
<http://java.sun.com/developer/technicalArticles/xml/JavaTechandXML/>
- [37] **A.Skonnard**, “.NET XML Best Practices”, *SoftArtisans, Inc.*, July 2003,  
[http://support.softartisans.com/kbview\\_673.aspx](http://support.softartisans.com/kbview_673.aspx)
- [38] “Oracle XML Developer’s kit Programmer’s guide for C++ ”, *ORACLE.*, 2013,  
[http://docs.oracle.com/cd/E12839\\_01/appdev.1111/b28394/adx\\_cp\\_classgen.htm](http://docs.oracle.com/cd/E12839_01/appdev.1111/b28394/adx_cp_classgen.htm)
- [39] “JDBC Overview”, *Oracle*, <http://www.oracle.com/technetwork/java/overview-141217.html>
- [40] “Accessing Data with C#”, *Exforsys Inc.*, March 2013,  
<http://www.exforsys.com/tutorials/asp.net/accessing-data-with-csharp.html>
- [41] “C++ Portability issue”, *Code Miles*, 2013,  
<http://www.codemiles.com/java/c-java-portability-and-performance-t474.html>
- [42] “LangPop Programming Community Index for October 2013”, *LangPop.com*,  
<http://www.langpop.com>
- [43] **M. Kosyakov**, “Long way to Maturity”, <http://www.worksforweb.com/publications/PHP-mature-platform/>

[44] “Swing GUI Builder (formerly *Project Matisse*)”, Oracle Corporation and/or its affiliates, 2015, <http://netbeans.org/features/java/swing.html>

# Appendix

## 1. Some names and their choice of candidate frameworks for GUI

No	Name	Education level	Seggusted Frameworks for GUI		
			C	C#	Java
1	Aschenaki Aschalew	Electrical Engineer			✓
2	Andualem Legesse	Computer Engineer			✓
3	Birra Kao	Computer Engineer		✓	
4	Mengistu Adum	Electrical Engineer			✓
5	Ragi Befekadu	Computer Engineer		✓	
6	Seyfe Shiferaw	Electrical Engineer			✓
7	Samrawit Kassaye	Electrical Engineer			✓
8	Solomon Feriew	Electrical Engineer		✓	
9	Tibebu Tiruneh	Electrical Engineer			✓
10	Yednekachew Denbel	Control Engineer			✓

## 2. Some comments during the desgin process

This is so fantastic idea, because if the application is develop properly and follow standard procedure due to language variety, it will be so applicable and easy for our local dispatchers.

*Seyfe Shiferaw, he is electrical engineer, founder and general manager of Yeki computer*

The most important thing is that, trying to develop such kind of advance application will promote local developers to create their own applications

*Aschenaki Aschalew, he is electrical engineer at Colombian Shipping line*

It is nice but it need more features to be GUI for train traffic control.

*Ragi Befekadu, he works in his own Computer maintenance shop*

**Submitted by:**

Yitagessu Hailemichael \_\_\_\_\_  
Student Signature Date

**Approved by:**

1. Abi Abate \_\_\_\_\_  
Advisor Signature Date

2. \_\_\_\_\_  
Chairman, Dept.'s Signature Date  
Graduate Committee

3. \_\_\_\_\_  
Chairman, Faculty's Signature Date  
Graduate Committee

4. \_\_\_\_\_  
Dean, Graduate school Signature Date