



Mizan-Tepi University
School of Computing and Informatics

Industrial Project Guidelines

Prepared By:

1. Dr. Khalid Mohamed
2. Mr. Fikru Kebede
3. Mr. Abrham Chalachew
4. Mr. Dessalegn Ashebir
5. Mr. Kiros Shiferaw

December, 2019
Tepi, Ethiop

Section-One: Preliminary Section

This section includes: Cover page, Title Page, Approval Sheet, Declaration of Student and advisor, acknowledgement, table of contents, list of tables, list of figures and illustrations, list of acronyms and abbreviations, and abstract.

I. Cover page

The cover page is the first page of the project document. It is preferably printed on hard paper. It includes the title of the project, name of the candidate/s (full name), logo of the university, University, name of the school, name of the department, town(place) of the university, month and year, in that order is formally submitted. This is the only page of a project for which a page number is not assigned.

II. Title Page

The title page is the second page of the project document, the first page for which a page number is assigned although it does not have a number typed on it. It includes the title of the thesis, name of the candidate (full name), name of the department, name of the school, University, name of advisor(s), town (place) of the university, month and year, in that order is formally submitted.

III. Approval Sheet

The final approval page will be incorporated into the student's project after being signed by the board of examiners and members of the advisory committee or advisor(s). The signing of the document will occur after a successful open defense and all required revisions to the document arising from the defense.

IV. Declaration of Student and Advisor

The project document shall contain a declaration of both student and advisor(s) to the effect that the work is the result of the student own investigation and that it has not been already submitted in candidature for a degree of this or any other university.

V. Acknowledgement

Acknowledgement is a brief account of the report or the origin and the utility of the study for which the project is presented. It also includes the acknowledgement to the persons and sources that have been helpful to the investigator. The title ACKNOWLEDGEMENT should be typed in capital letters.

VI. Table of Contents

This section lists all the main chapter headings and the essential sub-headings with the appropriate page numbers against each heading or sub-heading. The listing of the main chapters is generally preceded by some preliminaries like preface or acknowledgement, list of tables, list of figures, acronyms, abbreviations, abstract and their respective pages in small Roman numbers and followed at the end by appendices, and Indexes. Contents should neither be too detailed nor should too sketchy.

The table of contents serves as an important purpose in providing an outline of the contents of the report. The capitalized title 'Contents' should be the central heading of the page and the capitalized word 'CHAPTER' and 'PAGE' should lead to the numbers of chapters and those of pages respectively on the left and right margins.

VII. List of Tables

The table of contents is followed by the list of tables on a separate page. This list of tables consists of the titles or captions of the tables included "in the document along with the page number where these can be located. The capitalized title 'LIST OF TABLES' should be the central heading of the page and the capital words 'TABLE' and 'PAGE' should lead to the numbers and those of pages respectively at left and right margins. The statistical data are presented in vertical columns and horizontal row, according to some classification of subject matter. In the main body of the report any table should be completed within a page. Numbering tables shall be sequentially arranged through out project documentation.

VIII. List of Figures and Illustrations

A figure is a device that presents figurative data in pictorial or visual form. The figure is used to a variety of graphs, charts, maps, sketches, diagrams and drawings. It helps to understand the aspects of data clearly and easily. One idea or fact should be presented in each figure. The description of the figure must be given in the textual body. 'FIGURE' should be written

in the center of the page at the bottom of the figure. The title of the figure should be written in capital letters two spaces below the figure. In the main body of the report any figure should be completed within a page. A list of figures on a separate page is prepared in the same form as the list of tables except that they are numbered with Arabic numbers. Numbering figures and illustrations shall be sequentially arranged throughout the document.

IX. List of Acronyms

List of acronyms and abbreviation shall be included on a separate page.

X. Abstract

It is a brief summary of approximately 400 words (more or less). It should include the project objective, methods used, major contribution & conclusions and recommendations of the project. It should also be one page, one paragraph, single space and Italic in format. It should include key words.

Section-Two: Main Body Of The Report

The text of the project documentation is the most important section in the organization of your final report. The quality of worth of thesis is mainly examined in this section. It is the original production of the researcher. The main body of the report serves the function of demonstrating the competence of the student. If any sentence, paragraph, concept fails to serve the single function within a given section or chapter, it is irrelevant. The subject matter of any chapter should be relevant to that point.

CHAPTER ONE

1. INTRODUCTION

This section includes the background information of the subject, statement of the problem, significance and objectives of the project

1.1 Background of the company [Optional]

This section is optional. But if the project is proposed to a specific organization, in the beginning of the organizational background section, write a description of the mission of your organization in one or two sentences. Identify your organization's constituents and services. Include your organization's long-term goals, as well as what achieving these goals makes possible on a larger scale.

1.2 Background of the project

In background of the project, the team should:

- ❖ Create reader interest in the project topic,
- ❖ Lay the broad foundation for the problem that leads to the project,
- ❖ Place the project within the larger context of the field, and
- ❖ Reach out to a specific audience.

1.3 Motivation

Briefly justify what motivates you to do this project under this title.

1.4 Statement of the problem

The problem statement describes the context for the proposed system, and it also identifies the general analysis approach. It is important in a proposal that the problem stands-out that readers can easily recognize it. A problem statement should be presented within a context, and that context should be provided and briefly explained. This section should show what is already known and the gaps to be filled by the project.

1.5 Proposed System

This section clearly identifies and defines the central concepts or ideas of the project. It specifies the proposed solution for the specified problems under the statement of problem section.

1.6 Objective of the project

The objective of the project can be grouped into general objectives and specific objectives. The specific objectives are arising directly from the general objectives of the project. For each specific objective you must have a method to attempt to achieve it.

1.6.1. General objective

Are broad goals to be achieved. The general objectives of the study state what the project expects to achieve by the study in general terms. General objectives can be broken down into small logically connected parts to form specific objectives. General objective is met through accomplishing the entire specific objective.

1.6.2. Specific objective

Are short terms and narrow in focus? The specific objectives are more in number and they systematically address various aspects of problem as defined under the statement of the problem.

1.7 Scope and Limitation of the project

This section defines the specific area of the project. Limitation addresses how the project will be narrowed in scope and how it is bounded. This is the place to explain the things that the team is doing and why he/she has chosen not to do them.

1.8 Methodology and Tools

1.8.1. Methodologies

In this section the students should give clear, specific, appropriate and credible procedures that shall be followed to attain the proposed objectives of the project. The research/project design planned for use should be clearly stated. The method should be appropriate to the problem area i.e. the statement of the problem, objective and proposed system. The method will include,

- A. Requirement gathering and analysis methodology:(you can use any quantitative and qualitative data gathering methods)
- B. Design Methodology:(You can use one the system design method/SDLC models)
- C. Implementation Methodology: (The approach to be used on implementation/coding such as object oriented or procedural.)
- D. Testing Methodology: (such as unit testing, integration testing and system testing / white-box testing, black-box testing and gray-box).

1.8.2. Tools

This section is about the materials/(software and hardware) tools for each and every task with a reasonable selection criteria.

1.9 Feasibility Analysis

1.9.1. Economical feasibility

In economic feasibility, cost benefit analysis is done in which expected costs and benefits are evaluated. Economic analysis is used for evaluating the effectiveness of your proposed system.

In economic feasibility, the most important is cost-benefit analysis. As the name suggests, it is an analysis of the costs to be incurred in the system and benefits derivable out of the system

In case of a new project, financial viability can be judged on the following parameters:

- ✓ Total estimated cost of the project
- ✓ Financing of the project in terms of its capital structure, debt to equity ratio and promoter's share of total cost
- ✓ Existing investment by the promoter in any other business
- ✓ Projected cash flow and profitability

1.9.2. Technical feasibility

In technical feasibility the following issues are taken into consideration in your proposed project.

- Whether the required technology is available or not
- Whether the required resources are available:
 - Manpower- programmers, testers & debuggers
 - Software and hardware

Once the technical feasibility is established, it is important to consider the monetary factors also. Since it might happen that developing a particular system may be technically possible but it may require huge investments and benefits may be less. For evaluating this, economic feasibility of the proposed system is carried out

1.9.3. Operational Feasibility

Operational feasibility is mainly concerned with issues like whether the system will be used if it is developed and implemented. Whether there will be resistance from users that will affect the possible application benefits? The essential questions that help in testing the operational feasibility of a system are following.

- Does management support the project?
- Are the users not happy with current business practices? Will it reduce the time (operation) considerably? If yes, then they will welcome the change and the new system.
- Have the users been involved in the planning and development of the project? Early involvement reduces the probability of resistance towards the new system.

- Will the proposed system really benefit the organization? Does the overall response increase? Will accessibility of information be lost? Will the system effect the customers in considerable way?

1.9.4. Scheduling feasibility

A scheduling feasibility study will take into account the period in which the project is going to take up to its completion. A project will fail if it takes too long to be completed before it is useful. Typically, this means estimating how long the system will take to develop, and if it can be completed in a given time period using some methods like payback period. Time feasibility is a measure of how reasonable the project timetable is. Given our technical expertise, are the project deadlines reasonable? Some projects are initiated with specific deadlines. It is necessary to determine whether the deadlines are mandatory or desirable.

1.10 Benefits and Beneficiaries of the project

Indicates that, what and how the project will be significant and who will be benefited from the implementation of your project.

1.11 Risk and Constraints

In this section you are expected to discuss a risk, which is an event that may or may not happen, resulting in unwanted consequences or losses. And a constraint is a real-world limit on the possibilities for your project.

It describe any items or issues that will limit the options available to the developers. These might include: corporate or regulatory policies; hardware limitations (timing requirements, memory requirements); interfaces to other applications; specific technologies, tools, and databases to be used; parallel operations; language requirements; communications protocols; security considerations; design conventions or programming standards (for example, if the customer's organization will be responsible for maintaining the delivered software. So, you need to manage both risk and constraints carefully.

CHAPTER TWO

2. OVERVIEW OF EXISTING SYSTEMS

2.1 Introduction

This section describes about the current system such as the tools, methods and techniques what the company is currently using to handle their transactions.

- Structuring, classification and analysis of existing solutions together with their deficiencies
- Identification of key requirements and constraints
- Identification of performance indicators/test metrics
- Motivation and summary of open issues to be addressed

2.2 Literature study

It should be a critical analysis of relevant existing knowledge on the proposed project topic. It includes the strengths, the problem and flow of work of previous system related to your project. The literature study should be relevant with recent citations on the topic. Citations within the past five years are ideal and generally considered current. Citations ten years and older should be used sparingly and only when necessary. Unpublished documents and lay sources like encyclopedias are strictly discouraged. This is done by the student carefully tracking and referencing each and every document used

2.3 Actors in the current system

Identify the primary actors currently interacting with the existing system and identify the role of the actors and use case of the existing system.

2.4 Workflow

Work-flow describes the series of steps that need to be taken to complete a specific tasks in the existing system. You have to identify the flow of work between different actors and their role in the existing system.

2.5 Business rules

In this section, you have to identify the key rules and regulations of the organization about the task you are automating. The business rules will be gathered through interview and observation. Business rules can be developed by a broad range of approaches, for examples:

- A decision-making hierarchy for invoice processing, where the values of certain invoices are tiered to determine which managers can approve
- Calculations to determine bonus potential and payouts for sales personnel
- A series of required questions that if an answer is no, a certain vendor is excluded from a particular partnership opportunity

A set of questions and answers for a guest booking a hotel room, to determine availability and rates

CHAPTER THREE

3. REQUIREMENT ANALYSIS

3.1 Introduction

This chapter formally defines the detailed functional user requirements using high-level requirements identified in the initiation, system concept, and planning phases. It also defines the requirements in terms of data, system performance, security, and maintainability requirements for the proposed project. The requirements are defined in this part to a level of detail sufficient for systems design to proceed. The requirements that will be used to determine acceptance of the system are captured in the test and evaluation master plan. The purposes of this chapter are to:

- Further define and refine the functional and data requirements and document them in the requirements document,
- Verify what information drives the business process, what information is generated, who generates it, where does the information go, and who processes it.
- Develop detailed data and process models (system inputs, outputs, and the process)
- Develop the test and evaluation requirements that will be used to determine acceptable system performance.

3.2 Systems Requirement specification

The final output of this chapter is an SRS(Systems Requirement specification) which is a specification for a particular software product, program, or set of programs that performs certain functions in a specific environment.

3.3 Functional requirements

Functional requirements describe the interactions between the system and the users independent of its implementation. The aspects of a functional requirement is belongs to the system's behavior. Each requirement should be uniquely identified with a sequence number or a meaningful tag of some kind so you need to list all the functional requirement of the system that you want to developed.

3.3.1. Actor identification

In this section, the actors that have any kinds of interaction with the system usecase will be identified with their role. An actor describes any entity that interacts with the system.

3.3.2. System Use case (Use case identification)

A use case describes a function provided by the system that yields a visible result for an actor.

3.3.3. Use case scenario/ narrative (Requirements validation)

The use case narrative/scenario is a detailed scenario about the specific use case. The scenario should include name of the use case, primary actor, description, precondition, event flow, alternative flows

3.3.4. Use Case Diagram

Use case diagram is the representation of the functionality of the system. This section presents functionality of the system in terms of actors and use cases. Based on the identified use cases and actors you are expected to design a use case diagram.

3.4 Non-functional requirements

Nonfunctional requirements describe user-visible aspects of the system that are not directly related with the functional behavior of the system Which is a non-behavioral aspects such as real-time behaviour, performance, error tolerance etc.

3.5 Dynamic Modeling

Dynamic models are generally models that contain or depend upon an element of time, especially allowing for interactions between variables over time. A separate idea with the same name is models that are updated over time with new data. In the case of systems dynamics, we often have variables that are looped.

3.5.1. Sequence diagram

Sequence diagram describe behavior of the system as a sequence of messages exchanged among a set of objects. It is used to formalize the behavior of the system and to visualize the communication among objects of the system.

3.5.2. Activity Diagram

An activity diagram describes a system in terms of activities. Which is essentially an advanced version of flow chart that modeling the flow from one activity to another activity.

Activities are states that represent the execution of a set of operations. The completion of these operations triggers a transition to another activity. It is a flow diagram used to represent the data flow or the control flow through a system.

CHAPTER FOUR

4. SYSTEM DESIGN

In this chapter, the system is described by discussing the design goals of the project, by disintegrating the system into smaller subsystems that can be simply understood (proposed system architecture) and by selecting approaches for building the system, such as:

- The hardware/software platform on which the system will run
- The persistent data management strategy
- The global control flow
- The access control policy and
- The handling of boundary conditions.

The outcome of system design is a clear description of all of these approaches, subsystem decomposition, and a deployment diagram representing the hardware/software representing of the system.

4.1 Design Goals

Goals and objectives are statements that describe what the project will accomplish, or the business value the project will achieve.

- **Goals** are high level statements that provide overall context for what the project is trying to achieve, and should align to business goals.
- **Objectives** are lower level statements that describe the specific, tangible products and deliverables that the project will deliver.

4.2 Proposed System Architecture

The system architecture provides a comprehensive overview of the system. Depending on project outcome and scope, it consists of several architectural elements and modules, including business process model, function model (e.g. with use cases/user stories), data architecture, data model, and security architecture

4.2.1. System Hardware Architecture

The hardware architecture is the central document for the preparation of additional products. It specifies all hardware components and hardware modules of the hardware unit. The

individual elements and their specifications will be developed in accordance with these architectural requirements.

4.2.2. System Software Architecture

Software architecture refers the fundamental structures of a software system and the discipline of creating such structures and systems. It functions as a blueprint for the system and the developing project, laying out the tasks necessary to be executed by the design teams.

4.3 Hardware/Software Mapping

Hardware/software mapping describes how subsystems are assigned to hardware and off-the-shelf components. It also lists the issues introduced by multiple nodes and software reuse.

4.4 Persistent Mapping

Persistent data management describes the persistent data stored by the system and the data management infrastructure required for it. In this section the main modeling tool is ER(Entity Relationship) diagram.

4.5 Class Mapping/Class Diagram

Class diagrams are one of the most useful types of diagrams in UML as they clearly map out the structure of a particular system by modeling its classes, attributes, operations, and relationships between objects. The class mapping is used when the proposed system development follows an object oriented approach

.

4.6 State Modeling

A state model describes the sequences of operations that occur in response to external stimuli. A state model consists of multiple state diagrams, one for each class with temporal behavior that is important to an application. A state diagram relates events and states.

4.7 Component Modeling

A component model is a model that is used to encapsulate equations into a reusable form. By creating such a model, an instance of the component can be used in place of the equations it contains. A subsystem model is a model that is composed of components or other subsystems

4.8 Deployment Modeling

A UML deployment diagram is a diagram that shows the configuration of run time processing nodes and the components that live on them. Deployment diagrams is a kind of structure diagram used in modeling the physical aspects of an object-oriented system. They are often being used to model the static deployment view of a system (topology of the hardware).

4.9 Object Modeling

An object model is a logical interface, software or system that is modeled through the use of object-oriented techniques. It enables the creation of an architectural software or system model prior to development or programming. An object model is part of the object-oriented programming (OOP) life-cycle.

CHAPTER FIVE

5. IMPLEMENTATION

5.1 Introduction

Implementation is the part of the process where software engineers actually program the code for the project.

5.2 Environment setup

The environment setup section is about the configuration of hardware and software platform for the purpose of software development. For example:

- Hardware configuration is about: RAM, Processor, Hard disk etc.
- Software configuration: Operating system, IDE(Integrated Development Environment) for back-end as well as front-end, DBMS (Database Management System), programming language used etc.

Each and every hardware and software entity should be described with a detailed specification.

5.3 Algorithm design and coding

5.3.1. *Algorithm*

It always leads to a solution and tries to be the most efficient solution we can think up. It's often a good idea to number the steps, but you don't have to. Instead of numbered steps, some folks use indentation and write in pseudocode, which is a semi-programming language used to describe the steps in an algorithm. But, we won't use that here since simplicity is the main thing. Other folks just use a diagram called a flowchart.

Implementing the algorithm is the next step in the process. In order to know how to write the algorithm efficiently, you have to know exactly what each line of code is going to accomplish when the program is executed.

5.3.2. Sample code

For some of the main business logic, you have to put sample codes with a descriptive comment in the code and a caption that describes the functionality of the code.

5.4 User Interface (GUI) design

GUI(graphical user interface) is a program interface that allows a user to interact with an electronic devices through graphical icons. So, it is expected to design a GUI at least for the main system use cases. And a graphical user interface needs to follow a standard and should be attractive for the system user.

CHAPTER SIX

6. TESTING

6.1 Introduction

Testing, here also referred as Software testing is a process, to evaluate the functionality of a software application with an intent to find whether the developed project met the specified requirements or not and to identify the defects to ensure that the implementation is defect free in order to produce the quality project.

6.2 Test specification

A test specification is a document determining what functions and features of the software are to be tested during the test procedure, in what way, how often they must be verified, etc. All the project test specifications are included in the test plan.

6.3 Test cases

A test case is a set of conditions or variables under which a tester will determine whether a system under test satisfies requirements or works correctly.

6.4 Functional verification

It is a type of software testing which verifies that each function of the software application operates in conformance with the functional requirements. Each and every functionality of the system is tested by providing appropriate input, verifying the output, and comparing the actual results with the expected results. Verification means static testing and Validation means dynamic testing.

6.5 Systems integration testing

System integration testing (SIT) is a high-level software testing process in which testers verify that all related systems maintain data integrity and can operate in coordination with other systems in the same environment. The testing process ensures that all subcomponents are integrated successfully to provide expected results.

6.6 Sample test scripts

It is a set of instructions (written using a scripting/programming language) that is performed on a system under test to verify that the system performs as expected. Test scripts are used in automated testing.

Chapter Seven

7. Conclusion and Recommendation

7.1 Conclusion

Conclusion is the last paragraph of the project, which contains summary of the whole work and predictions for the future.

7.2 Recommendation

Recommendation can take two forms: recommendations for further study, or recommendations for change, or both. Each recommendation should trace directly to a conclusion.

Section-III: Reference and Appendices

This is the third section of the report. It consists of generally the references and appendices. The references and appendices are written on a separate page in the center with capital letters.

Reference

References are a list of the printed sources utilized in the project work. If the sources in the text are numbered to refer to the source in the references, the entries must be numerically listed in the order of appearance in the text. The various format manuals include information on form for the references. If the list of sources is too large the references should be categorized in the following sections: Books, monographs, documents and reports, periodicals and journals, essay and articles, unpublished thesis and material and newspapers. In writing references the surname is written first than initials, year of publication, title of the book, publishers name, place and total number of pages.

Different authors use different styles for citation. The commonly used style in the Science and Engineering fields is the one by the Institute of Electrical and Electronics Engineers (IEEE) where a citation number is enclosed within square brackets and the reference list is arranged by the order of citation in the project, not by alphabetical order.

All entries in the reference list must have been cited in the text at least once. Don't put too many references at once such as [2, 12, 14, 16, 18, 19, 33, 43] since it will be difficult for the reader to find out which information is taken from which source. Unless there are exceptional cases, don't use more than three references at once. When there are more than one reference, write them in ascending order and within 2 square brackets such as [3, 6, 10] and not as [10, 3, 6] or [3], [6], [10]. Use a single space after the comma.

References have two major objectives. Firstly, the project developers is acknowledging the works of others thereby avoiding plagiarism. Secondly, readers who need more information can access the referred material. Hence, all references must be observable. Arrangements vary, but an entry for a book usually contains the following information to be noticeable:

Author, Title, Publisher and Date of publication. An entry for a journal or a conference proceeding paper usually contains: Author(s), Article title, Journal title or conference details, Volume and number, Pages on which the paper is printed in the journal or conference proceeding and Date of publication. If there are more than three authors, provide et al. (meaning 'and others').

Example: J. D. Bellamy et al., Computer Telephony Integration, New York: Wiley, 2010.

In the Reference list, provide page numbers if you are referencing a section or chapter of the source:

Example: W. Brown, "Electrical Design Considerations," in Advanced Electronic Packaging: With Emphasis on Multichip Modules: Wiley-IEEE Press, 2013, pp. 51-74.

To reference a web page, use the following template:

Author Initial. Author Surname, "Title", Year Published. [Online]. Available: <http://Website URL>. [Accessed: 10- Oct- 2013].

Example:

Emarketer.com, 'Social Networking Reaches Nearly One in Four Around the World', 2014.

[O n l i n e] . A v a i l a b l e : <http://www.emarketer.com/Article/Social-Networking-Reaches-Nearly-One-Four-Around-World/1009976> . [A c c e s s e d : 23 - J u n - 2014] .

Appendices

An appendix is the important reference materials category. It includes the material which cannot be logically included in the main body or textual body of the project report or the relevant materials too unwieldy to include in the main body. The appendix usually includes: statistical tables and sometime raw-data (when data were processed through computer), source code listing. Even the material of minor importance e.g. forms, letters, reminders, interview sheets, blank questionnaires, charts, tables, lengthy questions, report of cases (if follow-up or case studies have been conducted). The appendix serves the function of providing greater clarity and authenticity for the readers or consumers of the document.

Appendix I: Standard Format for Reporting

Language	English
Paper Specifications:	
Color	White
Size	21 cmx29.7 cm (A4)
Weight	> 80 gm
Typing:	
Left margin	3 cm
Right margin	2.5 cm
Top margin	2.5 cm
Bottom margin	2.5cm
Spacing	1.5
Side	Front Single
Font size	12
Font type	Times New Roman,
Font style	Regular
Font color	Black
Breaking a word on 2 lines	Not allowed
Corrections with fluid	Not allowed
Overwriting	Not allowed
Crossing out words	Not allowed
Typing machine	Computer
Printing quality	Laser or better quality
Copies	High quality photocopy

A. Headings

Generally a project document is divided into chapters; each chapter begins from a new page. The title of a chapter is called the chapter heading. The word ‘CHAPTER’ is written in capital letters, in the center of the page and title is placed three spaces of the chapter. The following is an example:

- **Major Heading:** A chapter of the report is divided into major chairs. The major heading is written in capital letters, bold face and at the center of the page.
- **Sub-heading:** A major heading is sometimes divided into sub headings which are known as minor heading and it starts with left margin of a page in lower-upper letters.
- **Paragraph Heading:** If the minor heading is further divided, the paragraph is used. It must be indented five spaces and underlined. A full stop and dash is marked after such a heading. The written matter starts on the same line. These headings are also specified by using the numbers. For the Main headings 1, 2, 3, 4...so on are assigned in a chapter. The minor headings or sub-headings are shown in decimal numbers e.g. 2.1, 2.2, 2.3, it indicates that 1, 2, 3 are the minor headings of second main headings. Similarly paragraph headings are indicated in further decimal numbers e.g. 2.1.1, 2.1.2, 2.1.3 last numbers, 1, 2, 3 are paragraph headings of first minor heading of second major heading.

B. Pagination

Assigning page numbers of the report is very essential. The title page or initial page of any section does not have a page number typed on it, but a number is allotted to it in the series of pages. Page numbers are typed in the bottom right hand corner, one inch below the top edge of the page. The small or lower Roman numerals (i, ii, iii, iv,) are assigned for the pages of preliminary section. The serial Arabic numbers. 1, 2, 3, 4.....so on are assigned for the pages of textual body or main body of the report i.e. Chapter 1 to last references and annex or appendix.

C. Proof Reading

A project report should not have errors. It requires that final typed copies must be checked carefully. All types of errors should be deleted before submission.

D. Binding and Submission

It is the last activity for preparing project report. Before giving to the binder it should be arranged properly and systematically and the serial number of pages are checked carefully and should be approved by project advisor.

Appendix-II: Title/Cover Page



Mizan-Tepi University

School of computing and informatics

Department of -----

Industrial project on [**Title of the project**]

Submitted to the department of ----- in partial
fulfillment of the requirements for the Degree of Bachelor of
Science in -----

By:
Full name of the students

Advisor (s)

Tepi, Ethiopia

Month, Year

Appendix-III: Declaration Page

The Project is our own and has not been presented for a degree in any other university and all the sources of material used for the project have been duly acknowledged.

-----	-----	-----
Name	Signature	Date
-----	-----	-----
Name	Signature	Date
-----	-----	-----
Name	Signature	Date
-----	-----	-----
Name	Signature	Date
-----	-----	-----
Name	Signature	Date

Appendix-IV: Supervisor/Advisor approval sheet

To: department

Subject: Industrial Project Submission

This is to certify that the industrial project entitled -----
-----” submitted in partial fulfillment of the requirements
for the degree of bachelor of science in -----, has been
carried out by the group members under my supervision. Therefore, I recommend that the
students has fulfilled the requirements and hence hereby they can submit the project to the
department.

Name of advisor

Signature

Date

Appendix-V: Approval Page

It is approved that this project has been written in compliance with the formatting rules laid down by the faculty.

Examiners

-----	-----	-----
Examiner-1	Signature	Date
-----	-----	-----
Examiner-2	Signature	Date
-----	-----	-----
Examiner-3	Signature	Date

Chairman

-----	-----	-----
Chairman	Signature	Date

Appendix-VI: Use-case and Test-case samples

I. Use case:

Use Case ID:	<i>[Identifier of each of the use case. It is a running number to identify the bureau and the sequence of the use case. Format: XX(bureau identifier).XXXX (Sequence identifier) e.g. 08.0005]</i>
Use Case Name:	<i>[A concise, results-oriented (showing the goal of the use case) name for the use case. These reflect the tasks the user needs to be able to accomplish using the system. The use case name should start with an active verb. Example: Maintain Land Requesters]</i>
Created By:	<i>[The name of the person or the team that originally created the use case.]</i>
Date Created:	<i>[The date on which the use case is initially created] Format dd/mm/yy]</i>
Updated By:	<i>[The name of the person or the team modified the use case.]</i>
Date Updated:	<i>[Date of modification. Format dd/mm/yy]</i>
Actors:	<i>[An actor is a person or other entity external to the software system being specified who interacts with the system and performs use cases to accomplish tasks. Different actors often correspond to different user classes, or roles, identified from the customer community that will use the product. Name the actor that will be initiating this use case and any other actors who will participate in completing the use case.]</i>
Actors:	<i>[An actor is a person or other entity external to the software system being specified who interacts with the system and performs use cases to accomplish tasks. Different actors often correspond to different user classes, or roles, identified from the customer community that will use the product. Name the actor that will be initiating this use case and any other actors who will participate in completing the use case.]</i>
Description:	<i>[Provide a brief description of the reason for and outcome of this use case, or a high-level description of the sequence of actions and the outcome of executing the use case. Use not more than three statements.]</i>
Trigger:	<i>[Identify the event that initiates the use case. This could be an external business event or system event that causes the use case to begin, or it could be the first step in the normal flow.]</i>
Preconditions:	<i>[List any activities that must take place, or any conditions that must be true, before the use case can be started. Number each precondition. Examples: 1. User's identity has been authenticated. 2. All required files are loaded on the memory.]</i>
Normal Flow:	<i>[Provide a detailed description of the user actions and system responses that will take place during execution of the use case under normal, expected conditions. This dialog</i>

	<i>sequence will ultimately lead to accomplishing the goal stated in the use case name and description. This description may be written as an answer to the hypothetical question, “How do I <accomplish the task stated in the use case name>?” This is done as a numbered list of actions performed by the actor, alternating with responses provided by the system.]</i>
<i>Post conditions:</i>	<i>[Describe the state of the system at the conclusion of the use case execution. Number each postcondition. Examples: 1. A file is opened for the customer. 2. Customer is registered.]</i>
<i>Alternative Flows:</i>	<i>[Document other, legitimate usage scenarios that can take place within this use case separately in this section. State the alternative flow, and describe any differences in the sequence of steps that take place. Number each alternative flow in the form “X.Y”, where “X” is the number in the normal flow and Y is a sequence number for the alternative flow. For example, “5.3” would indicate the third alternative flow line number 5 in the normal flow.]</i>
<i>Exceptions:</i>	<i>[Describe any anticipated error conditions that could occur during execution of the use case, and define how the system is to respond to those conditions. Also, describe how the system is to respond if the use case execution fails for some unanticipated reason. If the use case results in a durable state change in a database or the outside world, state whether the change is rolled back, completed correctly, partially completed with a known state, or left in an undetermined state as a result of the exception.</i>
<i>Priority:</i>	<i>[The relative priority of the service provided by the given use case. Three categories of priority values can be assigned: High, Intermediate or Low]</i>
<i>Frequency of Use:</i>	<i>[How frequently is the use case used – every five minutes, once in a day, every other day, once in a week, etc]</i>
<i>Business Rules:</i>	<i>[Write any business rule of the organization that directly or indirectly affect the use case]</i>
<i>Special Requirements:</i>	<i>[Identify any additional requirements, such as nonfunctional requirements, for the use case that may need to be addressed during design or implementation. These may include performance requirements or other quality attributes.]</i>
<i>Assumptions:</i>	<i>[List any assumptions that were made in the analysis that led to accepting this use case into the product description and writing the use case description.]</i>
<i>Notes and Issues:</i>	<i>[List any additional comments about this use case or any remaining open issues or TBDs (To Be Determined) that must be resolved. Identify who will resolve each issue, the due date, and what the resolution ultimately is.]</i>

II. Test Case

Test Case Template

A test case can have the following elements. Note, however, that a test management tool is normally used by companies and the format is determined by the tool used.

Test Suite ID	The ID of the test suite to which this test case belongs.
Test Case ID	The ID of the test case.
Test Case Summary	The summary / objective of the test case.
Related Requirement	The ID of the requirement this test case relates/traces to.
Prerequisites	Any prerequisites or preconditions that must be fulfilled prior to executing the test.
Test Procedure	Step-by-step procedure to execute the test.
Test Data	The test data, or links to the test data, that are to be used while conducting the test.
Expected Result	The expected result of the test.
Actual Result	The actual result of the test; to be filled after executing the test.
Status	Pass or Fail. Other statuses can be 'Not Executed' if testing is not performed and 'Blocked' if testing is blocked.
Remarks	Any comments on the test case or test execution.
Created By	The name of the author of the test case.
Date of Creation	The date of creation of the test case.
Executed By	The name of the person who executed the test.
Date of Execution	The date of execution of the test.
Test Environment	The environment (Hardware/Software/Network) in which the test was executed.