Graduation Project Document Format

اعداد

م. نعمة عبد العزيز

مراجعة وتنقيح

د. ناجي الشيباني

أبريل 2016

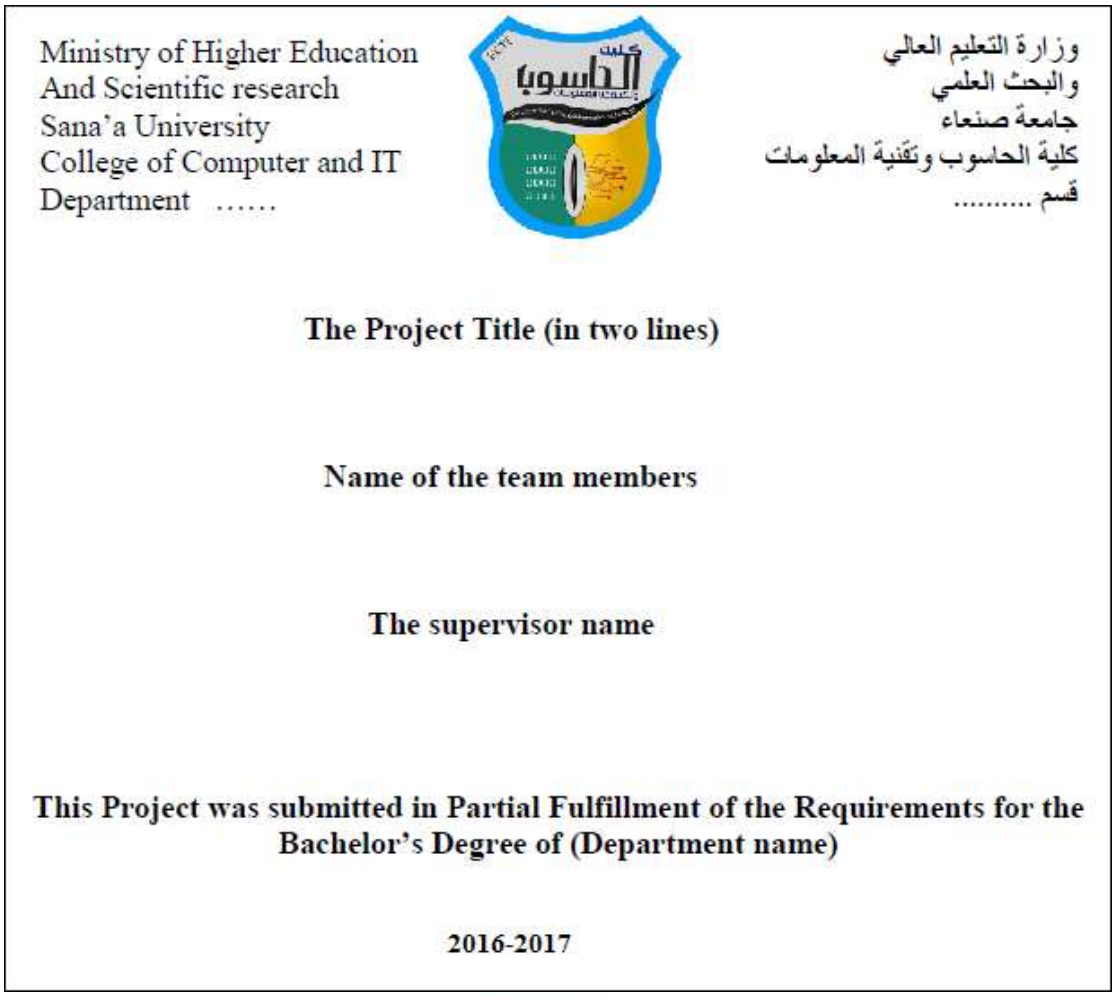
كلية الحاسوب وتكنولوجيا المعلومات

Graduation Document Format

Please read that document for a general discussion of how to prepare a proposal document.

Cover Page

Team Members, logo (optional), Deliverable name, Date. As the following Example:


Ministry of Higher Education
And Scientific research
Sana'a University
College of Computer and IT
Department .....

وزارة التعليم العالي
والبحث العلمي
جامعة صنعاء
كلية الحاسوب وتقنية المعلومات
قسم .....

The Project Title (in two lines)

Name of the team members

The supervisor name

This Project was submitted in Partial Fulfillment of the Requirements for the Bachelor's Degree of (Department name)

2016-2017

*This page should be replicated in the second page and should be signed by the supervisor.

Dedication: List the names of anyone who you may want to dedicate your work to.

Acknowledgements: The Acknowledgements lists the names of anyone who may have given you valuable assistance in your project.

Table of Contents

The Table of Contents outlines the different sections of the report, and shows the reader where to find them. It contains a list of all the chapters, sections and

sub-sections and their corresponding page numbers. The Table of Contents can be generated electronically using Microsoft Word.

Table of Tables

In this section, all figures (including screenshots, photographs, drawings and graphs) in the report are to be listed together with respective page numbers.

LIST OF TABLES

NUMBER	TABLE CAPTION	PAGE
1	A Snapshot of Processed UJ Log File.	27
2	A Snapshot of Path and its Frequency.	27
3	A Snapshot of Path Statistics.	28
4	A Snapshot of Modified Path Staistics.	31

* The numerals and **title (legend)** for a figure are **placed below it** and start at the left margin at the bottom of the page.

Table of Figures

In this section, all tables in the report are to be listed together with respective page numbers.

LIST OF FIGURES

* Tables and figures should be placed as close as possible to the text where they are first cited. Tables and figures in the appendices should be numbered consecutively following those in the text.

Abstract

This gives the reader a general *overview/summary* of the *whole* report without them having to read the entire document. It should be able to stand alone as a separate document if required.

* It's an abbreviation for your whole project, must be written at the end of documentation phase, and it must not be more than 3 paragraphs. It contains the general goal of your project and its final expected result(s). It is the part most readers of the thesis/project will read it first. The abstract should be very well written. It should be clear, easy to read, and to-the point. The abstract conveys the most important messages regarding your project, such as: what you set out to do? How did you do it? What results were obtained? You will have a much better shot at writing a good abstract after you have completed all the other parts of the thesis.

Chapter 1. Introduction *mandatory

1.1 Background

Present any domain-specific background necessary to understand the problem. This part of this chapter is to prepare the reader's mind to understand what is the targeted system, people or organization and what the main problem is, for which this project will be done to solve. The main problem should be mentioned in the last paragraph of this point

Note: it's not necessary all the time to have a problem, sometimes we have an opportunity and would like to have a system to benefit this opportunity.

⇒

1.2 Problem Statements

Discuss the specific problems (the main problem in details) that your software product will attempt to solve. You should mention (just) your project's problems those can be solved using computer sciences.

Note: The same must be done if we have an opportunity, it must be explained in details to provide a way to discover the objectives.

⇒

1.3 Objectives

Describe – in points preferably –the objectives(the partial solutions for the detailed problem statements) of your project, Include a measurable statement of the intended benefits or objectives to be achieved by developing this system.

⇒

1.4 Acceptance Criteria

List a number of success criteria necessary for the project (the most important parts of the proposed system which must be found in the final delivered system).

⇒

1.5 System Definition:

Give a clear definition about the project (type: application or thesis, if it is an application, identify its type (desktop, web, or android app)), that you will implement and explain its new features.

⇒

1.6 Purpose

Describe the purpose of this project. You can explain that when you specify who can benefit your project, i.e. the most interested benefits in future that all of customers, employees or the whole society will gain.

⇒

1.7 Goal

Give a brief summary of the major features that the system will perform. The main goal that will solve the main problem. It must be written as one paragraph and not more than one sentence.

⇒

1.8 User Characteristics

Describe the users to whom this project is introduced.

User group name	Qualifications	Age-period (if necessary)	Needed skills	Tasks
Eg: student	Scientific Secondary	18-30	Could use the PC and search via the Internet	Create an account … etc Note: All the tasks must be mentioned here.

In Qualifications, Age-period and needed skills just show what will serve you in your work.
⇒

1.9 Limitations

List any conditions or constraints that prevent you from developing or implementing the system.
⇒

1.10 Assumptions and dependencies

List any factors or dependencies that the developers may assume will exist that may affect the software product (limitation solving).
⇒

1.11 Scope

Describe the scope of the software to be produced. The scope here is two parts: the first is the actual domain, eg: Student Affairs Departments, FCIT, Sana'a University. The second is the operational borders of the system, i.e. the operations that will be implemented in the final deliverable system. The final writing of this point must start with the operational borders then the actual domain.
⇒

1.12 Life Cycle Model

Choose an appropriate life cycle model that you see it may be appropriate for your system. You should mention here why you choose to work using this model and give a brief explanation of your chosen methodology.
Note: the choosing of the model will affect the order of the next chapters of this documentation.
⇒

1.13 Related Work

Mention the names of previous related work in the same domain or those using the same technology. Those names must have a citation to make the reader know more information about them from the references page.
Note: the details about these names will be written in the Fact Finding part(s).
⇒

1.14 Project Plan

Mention the steps you will follow up to complete your project.
Note: these steps must be written according to the chosen/used life cycle model.
⇒

1.15 Feasibility Study Is the Problem Worth Solving?

1.15.1 Technical Feasibility

A. Technical Feasibility – Needed Hardware

B. Technical Feasibility – available Hardware

C. Technical Feasibility – Needed Software

D. Is it practical or not?

E. Needed Experts

F. Can current technical resources be upgraded or added to in a manner that fulfills the request under consideration?

1.15.2 Financial Feasibility

It must be written in a table such as the following one:

Requirements	Cost			
	Manual	System		
		build	buy	rent
HR: Manager Employee Analyst Designer Developer Team leader Guard				
HW: PC Server Wires Switch Modem Router				
SW: Windows 8 Windows server 2008 SQL server 2008 Visual studio.net 2010 Antivirus and internet protection Office 2013 and Project E-draw Max				

Adobe Photoshop Adobe Acrobat Server Hosting and distribution System additional helping programs				
Other sets: Printer Printer invoices and receipts Scanner A fingerprint system				
Risks: Alert sensors Fire-extinguisher Monitoring screens Surveillance Cameras				
Stationary: Printer paper drafts Printer invoices and receipts paper Pens Folders Disclosures				
Others: Wrong products delivered Bad taste Maintenance Renewing Renting				
Total:				

1.15.3 Operational Feasibility

A. According to PIECES standard, we can talk about:

i. Performance:

i.i throughput:

i.ii response time:

ii. Information:

ii.i input:

ii.ii stored data:.

ii.iii output:.

iii. Economic:

iii.i cost:
iii.ii profit:

iv. Control/ security:

iv.i too low:.
iv.ii too high:.

v. Efficiency:

v.i waste time:.
v.ii waste materials:
v.iii effort:
v.iv required materials:

vi. Services:

vi.i inaccurate, inconsistent or unreliable result:
vi.ii easy to learn/to use
vi.iii inflexible.
vi.iv incompatible

B. Available human resources
C. Users that do not want a new system may prevent it from becoming operationally feasible.
D. Is users are satisfied with current system?
E. Legal system

1.15.4 Scheduling Feasibility (planning the time)

1.15.5 Identifying Benefits and Costs

A. Tangible benefits

B. Intangible benefits

C. Tangible costs

D. Intangible costs

Chapter 2: Current System (Theoretical side)

Describe and analyze the business logic and flow of the ***current*** system.
⇒

Chapter 3.

Note: From this chapter each documentation may be different. This depends on the life cycle model type and the project type (Applicable project or thesis).

A. Thesis Research

The following two or three chapters must have the description of the proposed thesis, its implementation and result. It must be declared clearly and all sides must be shown within the test of the thesis before generalizing the final result.

Chapter on Review of Literature or previous work

1. This chapter is different from the background provided in the Introduction. The background provides general information. The literature review focuses on issues that are more specifically related to the work in your project.
2. Describe similar work done by others in the past and described in the literature. If you cannot find prior work in the literature, then it is most likely that the work you are describing is too simple to qualify as a graduate project.
3. Your document needs to demonstrate that you have done a literature search and completed a critical analysis of the relevant literature describing prior work in the field. Demonstrate this by writing some discussions on what others have done, what they have achieved, and limitations of their work. If they exist, then provide reviews of prior work in the literature, this shows that you have done a comprehensive literature search.
4. Do not copy and paste text from the literature; paraphrase the contents in your own words.

Chapter on designing the proposed system

Describe here your proposed system/thesis and discuss why you built it this way and the way it must work.

Chapter on Implementation and testing

Describe here your proposed system/thesis implementation and test it using many test cases then discuss and evaluate the final results.

B. Agile (Rapid Application Development or Iterative) projects

In this type of projects the chapter 3 and 4 below must be written more than once, at least three times. In each time it must be one chapter. This chapter name must be "Release X" where x is the number of release.

C. Structural projects

Proposed System

3.1 Overview

Provide a brief overview of the project you are intending to develop

⇒

3.2 Fact Finding Tools

Some of these tools are mandatory and others are optional. You must use all mandatory tools and one or some of optional tools

⇒
Mandatory:
3.2.1 Sampling of existing documentations, forms, and database or file.
3.2.2 Research and site visits
Optional:
3.2.3 Observation of work environment.
3.2.4 Questionnaires.
3.2.5 Interviews.
3.2.6 Prototyping.
3.2.7 Joint Requirement Planning
3.2.8 CASE Tools.
3.2.9 Any other interactive technology (needed according to the chosen model)

* Each one of these tools must be mentioned briefly - their results or report of your work - all of their particular cases, their answers, etc must be added in appendices.
* Books' authors, and websites links must be added in references page. In the official document you must write just their names only!

3.3 Requirements Specification

3.3.1 User Requirements (preferences)

Statements, in a natural language, of what services the system is expected to provide as a preferences

[Steps]

3.3.1.1

3.3.1.2

3.3.2 Functional Requirements

Statements, in a natural language, of what services the system is expected to provide as a services or functions. List all the required functionality of the software in complete detail. Number and organize the requirements carefully.

3.3.3 Non-Functional Requirements

This section describes any desired attributes or characteristics required of the system that do not provide a function or capability. Another way of saying it is that this section specifies how much quality is required.

One way to distinguish a functional requirement is that you can point to the source code that implements it, but nonfunctional requirements can't be isolated in the code.

Note: Use just the needed non-functional requirement, to ensure that the chosen requirements will never conflict.

These requirements are subject to the same criteria as the previous section. Special attention must be given to stating the requirements in a manner that is objective and quantifiable; there must be some measurable way to assess whether the requirement has been met.

3.3.3.1. Usability

Is the ease with which a user can learn to operate, prepare inputs for, and interpret outputs of a system or component? This can be done using user interface, online help, and user documentation.

⇒

3.3.3.2. *Dependability*

- ***Reliability.*** is the ability of a system or a component to perform its required functions under stated conditions for a specified period of time, this includes:
    - Acceptable mean time to failure.
    - Ability to detect specified faults.
    - Detect the security attack.
- ***Robustness:*** the degree to which a system or a component can function correctly in the presence of invalid inputs or stressful environment conditions.
- ***Safety:*** a measure of absence of catastrophic consequences to the environment.
- *Security*: a measure of how the system can prevent it self from the external attack.

⇒

3.3.3.3. Performance

Requirements are concerned with quantifiable attributes of the system, such as:

- Response time: how quickly the system reacts to the user input.
- Throughput: how much work the system can accomplish within a specified amount of time.
- Availability: the degree to which a system or component is operational and accessible when required for use.
- Accuracy.

⇒

3.3.3.4. Supportability

The ease of change to the system after deployment:

- Adaptability: the ability to change the system to deal with additional application domain concepts.
- Maintainability: the ability to change the system to deal with new technology or to fix defects.
- Internationalization: ability to change the system to deal with additional international conventions, Such as language unit, number format.
- Portability: the ease with which a system or a component can be transferred from one hardware or software to another.

⇒

3.3.3.5. Validation

Is a critical step in the development process? In the requirements validation we check if the specification is:

- Complete: all possible scenarios are described including exceptional behavior in requirement model.
- Consistent: if the requirement specification does not contradict itself.
- Unambiguous: if the requirement specification define exactly one system.
- Correct: if the requirement specification represent accurately the system that the client need, and the developers intend to build.

⇒

3.3.3.6. Other

- **Realistic**: if the system can be implemented within constraints.
- **Verifiable**: once the system is built, repeatable test can be designed to demonstrate that the system fulfills the requirement specification.
- **Traceable**: if each requirement can be traced through the system development to its corresponding system functions, and if each system function can be traced back to its corresponding set of requirements.

⇒

A.

3.5 Modeling System Requirements (Structural process modeling)

3.5.1 DFD

List below the project's main usage scenarios

3.5.1.1 ⇒<Scenario >

⇒ [The DFD description, written in simple and easy to understand language and particularly]

3.5.1.2 ⇒<Diagrams >

3.5.2 DFD descriptions:

⇒ [Using forms provided by your systems analysis and design book]

3.5.2.1 Data flow data dictionary

3.5.2.2 Data Structure

3.5.2.3 Data elements

3.5.3 Structured System Analysis

3.5.3.1 Process Model Description (Process specification)

3.5.3.2 Logical Process Modeling

3.5.3.2.1 Structured English

3.5.3.2.2 Decision Tables

Include a matrix representation of the logic of a decision, which specifies the possible conditions for the decision and the resulting actions. Choose two processes to model their logic requirements using decision table.

3.5.3.2.3 Decision Trees

Include a graphical representation of a decision situation in the form of a tree. . Choose two processes to model their logic requirements using decision trees.

3.5.3.3 Data Model

Include visual depictions of the data and relationships in the problem domain using the standard notation given in the **Entity-Relationship Model.**

B.

3.5 System Models (UML)

System Models (Except for the scenarios subsection) are to be developed during the Analysis phase.

3.5.1. Scenarios

[List below the project's main usage scenarios, written in simple and easy to understand language and particularly]

3.5.1.1 ⇒<Scenario 1>

⇒ [The scenario description]

3.5.2. Use case model

Insert the diagrams depicting your use case model

⇒

3.5.2.1 Actors

Identify and describe the system's actors

⇒<Actor1>

⇒ [Actor 1's description]

3.5.2.2 Use case details

Use case name:	UniqueID:
Area:	
Actor(s):	
Stakeholder:	
Level:	
Description:	
Triggering Event:	
Trigger type:	
Steps Performed (Main Path)	Information for Steps
Preconditions:	
Postconditions:	
Assumptions:	
Success Guarantee:	
Minimum Guarantee:	
Requirements Met:	
Outstanding Issues:	
Priority:	
Risk:	

Steps Performed (Main Path)	Information for Steps
Extensions or Alternative Scenarios	

3.5.3. Object model

3.5.3.1 Class diagrams

Add in your analysis class diagrams
⇒
List down your classes, describe them and state its stereotype (Boundary, Control or Entity)

Class Name	Description	Stereotype

3.5.3.2 Sequence diagrams

⇒
Add in your analysis sequence diagrams

3.5.4. Dynamic model

Add in your state charts and describe the flow of every state chart
⇒

3.5.4.1 State Chart diagrams

⇒
Add in your analysis state chart diagrams

3.5.4.2 Activity diagrams

⇒
Add in your analysis Activity diagrams

3.5.5. Database model

Add in your ER schema and ERD with their description
⇒

C. Any other models

Chapter 4. Implement the System

4.1 Overview

Provide a brief overview of how to develop this project
⇒

4.2 Database building

Provide all database tables creation codes, relations and how to connect with interface environment
⇒

4.3 Interface building

Provide the main and essential interfaces in your project
⇒

4.4 System Manual

Provide the manual of your system or at least its critical or essential functions
⇒

4.5 Integration and System Testing

Provide the result of these tests in your documentation
4.5.1 Integration Test

- Big Bang
- Top-Down testing
- Bottom-up testing
- Overall

4.5.2 System Test

- Usability testing – this is how well the user can access the different features in the system and how easy it is to use.
- GUI software testing – this is to check if graphically that the program looks how was intended and the GUI works as intended.
- Security testing – this would be to check if important information is secure and if there are certain access restriction that they work.
- Accessibility – how easy is it for various users including users with disability to use the system.
- Reliability testing – to check that the system works for long period of time and does not constantly crash.
- Overall

⇒

4.6 Results of the project

Provide the main and essential results in your project
⇒

Note: this chapter is a general chapter for all the types of projects

Chapter 5. Recommendations and Suggestions

5.1 Overview

Provide a brief overview of the project you had developed
⇒

5.2 Suggestions (Future Plan)

Provide your suggestions to develop new releases of your program
⇒

5.3 Recommendations

Provide your recommendations to enhance developing new releases of your program or any other program in the same domain
⇒

ACRONYMS AND ABBREVIATIONS

List here all new acronyms or your system abbreviations.

Appendices

Use Appendices to present material that will interrupt the flow if included in the main body of your document. Typical contents of appendices include: Source Code, data tables, detailed analysis and design models, surveys and interviews. If a user manual is called for, then provide it in an appendix.

References

Every citation made in the body of the thesis must appear in the Bibliography. Similarly, every item listed in the Bibliography must be cited in the body of the thesis.

2. The committee may use the list of references as a yard stick to assess how well you have researched the field before setting out to do your project. The committee may look for completeness and also accuracy of the references. Error in the bibliography will need to be corrected before a thesis is approved.

3. Follow a single standard method for citing and listing both the print references and the online references. There are many different formats for citing and listing references, such as: APA, MLA, ACM style, IEEE style, etc. Choose one and follow it consistently throughout the thesis. Note that there is a standard method for listing online references, listing just the URL is not sufficient.

For example, we can use MLA to cite a book and a website:

For Books:

Structure:

Last, First M. Book. City: Publisher, Year Published. Print.

Examples:

- James, Henry. The Ambassadors. Rockville: Serenity, 2009. Print.
- Dickens, Charles. Great Expectations. New York: Dodd, Mead, 1942. Print.

For Websites:

Structure:

Last name, First name. "Article Title." Website Title. Publisher of Website, Day Month Year article was published. Web. Day Month Year article was accessed. <URL>.

Example:

Cain, Kevin. "The Negative Effects of Facebook on Communication." Social Media Today RSS N.P., 29 June 2012. Web. 02 Jan. 2013.

ملاحظات

- يتم كتابة التقرير باللغة العربية او الإنجليزية. وفي حالة اختيار العربية يتم كتابة المصطلحات باللغة الإنجليزية ويتم عمل فهرس بهذه المصطلحات مع شرح موجز لها في نهاية التوثيق.
- يتم كتابة التوثيق بمراجعة ومتابعة المشرف ولن يقبل أي توثيق لا يعتمد ويوقع من المشرف.
- الأخطاء الاملائية والنحوية مسؤولية الطلاب فيجب عليهم مراجعة كاملة وشاملة للتوثيق وتصحيح كل الأخطاء قبل التسليم النهائي.
- يتم تسليم ثلاث نسخ مجلدة من التوثيق. نسخة للمشرف ونسخة للمكتبة ونسخة لرئيس القسم. يجب توقيع المشرف على كل النسخ.
- يجب تسليم قرص مدمج مع نسخ التوثيق يحتوي التوثيق والكود البرمجي والبرنامج او النظام التنفيذي.
- درجة التوثيق تمنح بناء على الالتزام بالمخطط أعلاه وبعدم وجود أخطاء.
- التوثيق مستند بالغ الأهمية وعليه يجب إعطائه وقت وعناية فائقة ليخرج بشكل لائق.