# Game Graduation Project Document Format

**Prepared By:**

**Eng. Neama Abdulaziz**

**Jan. 2022**

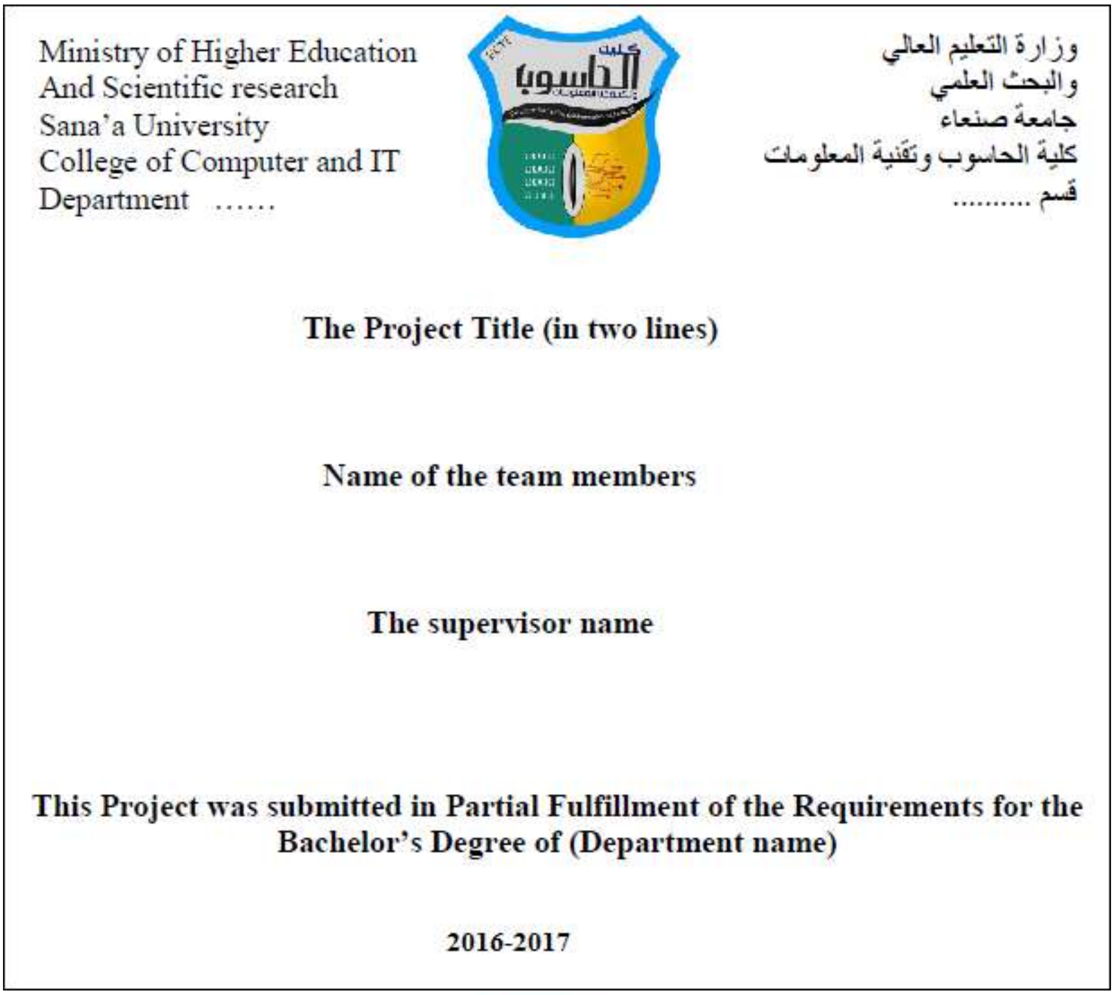
**Faculty of Computer and IT – Sanaa University**

# Game 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**

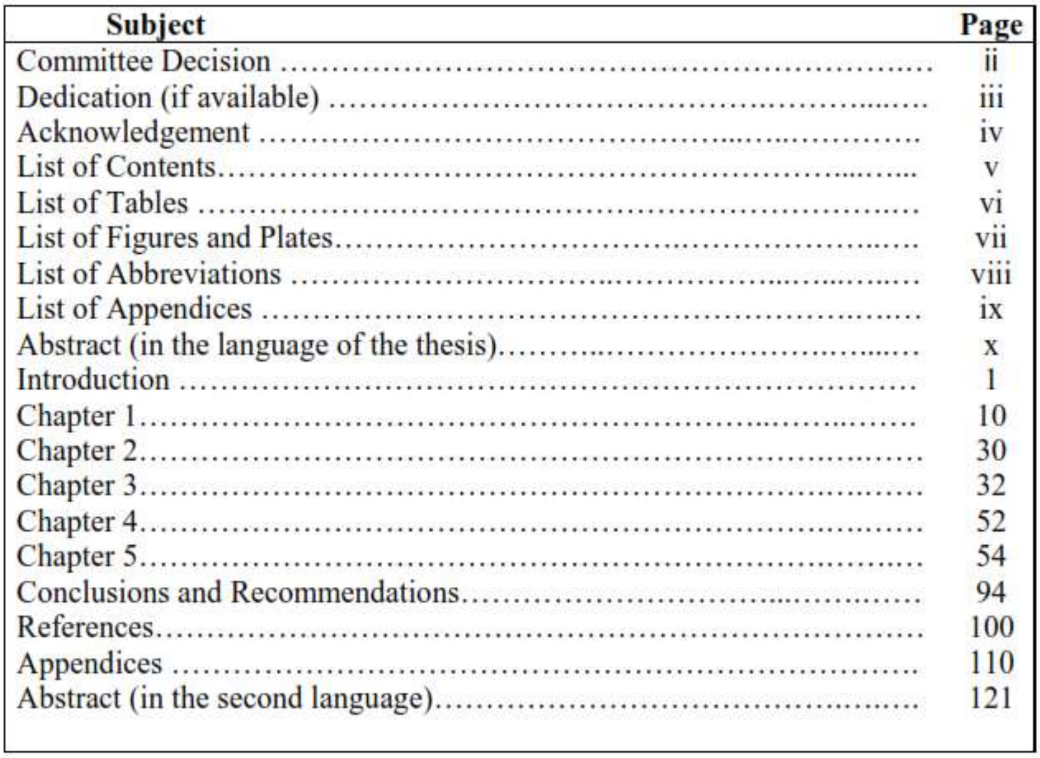
*This page should be replicated in the second page and should be signed by the supervisor.

**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.

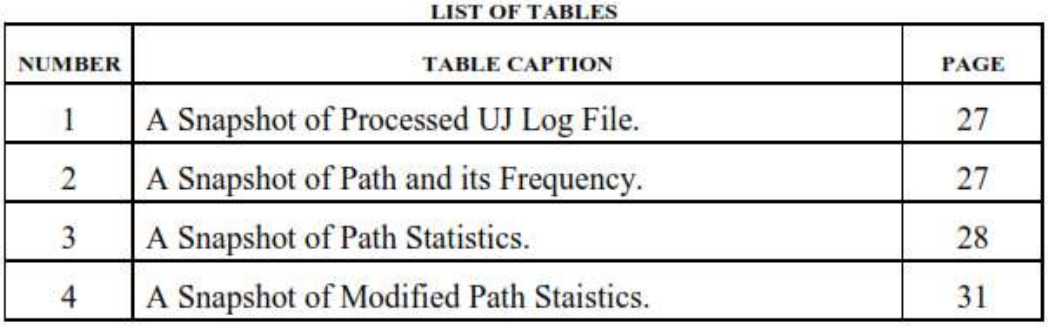
## Table of Contents

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.

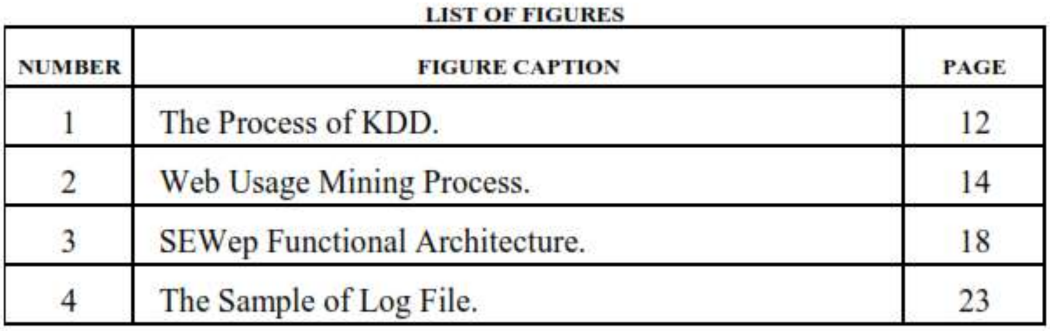


LIST OF TABLES

* 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

## 1.1 Background/Project Introduction

Introduce your idea as a game and the reasons behind this type of project and idea. In general, mention the specific fields this game is coming from and the necessity of this type of game.

⇒

## 1.2 Game Overview

Discuss the specific dimensions should be covered in this game. Where does the story come from? Why do you need to implement this story? What are the needs, lacks and goals, tasks, missions, and final rounds (in general)?

⇒

## 1.3 Game Specifications

You need to specify some characteristics to determine the type, audience and many other characteristics of your game. These characteristics should be as the following examples:
**Type**: serious
**Game graphics type**: 2D
**Group**: all Yemeni
**Target age**: EGPA
**Topic**: Co-existence
**Time**: current
**PLACE**: fictional
**Seasons**: unlimited
**Winning by season**: making resources available equally for all people in all places (at this season there are 6 places/6 missions)
**Task**: Stop fighting and providing resources equally
Why this character: it was bullied and marginalized
**Problem**: people are fighting each other for getting resources for themselves alone
**How to play**: you should state the ways/choices provided to play the game according to the story
**Story**: state the story type, characters and brief paragraph about it
⇒

## 1.4 Game ending

Here you have some points to be shown:

a. **Prototype**:
b. **Implementation**:
c. **Posters**:

in each one of these you should specify an ending for the game that is making the user excited and not feeling unsatisfied.
⇒

## 1.5 Game Definition as a software:

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

Faculty of Computer & Information technology

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/player Characteristics

Describe the users to whom this project is introduced.

| User group name | Type | Age-period (if necessary) | Needed skills | Tasks |
|---|---|---|---|---|
| Eg: student | Player/character | 18-30 | Could use the PC and search via the Internet | Create an account … etc<br>Note: All the tasks must be mentioned here. |

In 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. How many missions, and level will be delivered for the first trial, time duration and any other types of constraints.

⇒

## 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

FCIT
كلية الحاسوب وتكنولوجيا المعلومات
Faculty of Computer & Information technology

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:**<br>Manager<br>Employee<br>Analyst<br>Designer<br>Developer<br>Team leader<br>Guard | | | | |
| **HW:**<br>PC<br>Server<br>Wires<br>Switch<br>Modem | | | | |

|  |  |  |  |  |
|---|---|---|---|---|
| Router |  |  |  |  |
| **SW:**<br>Windows 8<br>Windows server 2008<br>SQL server 2008<br>Visual studio.net 2010<br>Antivirus and internet protection<br>Office 2013 and Project<br>E-draw Max<br>Adobe Photoshop<br>Adobe Acrobat<br>Server Hosting and distribution System<br>additional helping programs |  |  |  |  |
| **Other sets:**<br>Printer<br>Printer invoices and receipts<br>Scanner<br>A fingerprint system |  |  |  |  |
| **Risks:**<br>Alert sensors<br>Fire-extinguisher<br>Monitoring screens<br>Surveillance Cameras |  |  |  |  |
| **Stationary:**<br>Printer paper drafts<br>Printer invoices and receipts paper<br>Pens<br>Folders<br>Disclosures |  |  |  |  |
| **Others:**<br>Wrong products delivered<br>Bad taste<br>Maintenance |  |  |  |  |

FCIT
كلية الحاسوب وتكنولوجيا المعلومات
Faculty of Computer & Information technology

| | | | | |
|---|---|---|---|---|
| Renewing<br>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. Are users are satisfied with current system?***

***C. 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**

FCIT
كلية الحاسوب وتكنلوجيا المعلومات
Faculty of Computer & Information technology

# Chapter 2: Theoretical side

In this chapter you need to mention the following:

## 2.1 Game Development

Briefly, describe the process of game development, history and stages. Add photos and references if possible.
⇒

## 2.2 Your game story

Write your story here, if possible. If not, give us a brief summary of what will be done during each level or mission of the game.
⇒

## 2.3 Game Analysis

Briefly, describe the process of game development, history and stages. Add photos and references if possible.

### 2.3.1 Levels of difficulty

“Trouble in the cave” is a serious game made for primary school students. The level of difficulty depends on the students’ previous knowledge about geometry, measurement and basic ICT skills such as clicking, navigating on a computer and moving elements. The game in itself is therefore easy to manage, and the level of difficulty is considered to be easy or easy/normal.

### 2.3.2 Number of players

This is a one-player game, without any opponents.

### 2.3.3 Positive stimulus of the players

The player has to complete two mathematical tasks. For each task the player will receive a key. These keys must be collected in order for the player to complete the game. Also, after completing a task, the player will receive positive feedback from different computer characters. The keys and the feedback are the driving force and the positive stimulus of the player.

FCIT
كلية الحاسوب وتكنولوجيا المعلومات
Faculty of Computer & Information technology

### 2.3.4 Tracking/ counting of achieved results

The player will receive a key for each task that he completes. The number of keys earned, will be shown at the top of the screen.

### *2.3.5 Place of the game in educational process*

*Describe how the game can be used:*

### 2.3.6 Used software

Our game is made with the visual programming environment Scratch. Some of the pictures were refined with Paint and PhotoScape.

### 2.3.7 Minimum system requirements

The game can be played offline but you need to have installed Scratch 2.0. This version will work on Mac, Windows and some versions of Linux (32 bit). But for opening Scratch, you need to have Adobe Air installed.
The game can also be played online.

### 2.3.8 Player Experience

…

### 2.3.9 Game ethics:

Bnm,./
⇒

## 2.4 Game plot

You can use the points of the following example:
<u>Exposition:</u> We are situated outside a cave in Slovenia. Our main characters, the protagonists, are being introduced. We meet a Bulgarian girl named Ivanka, and a Norwegian Viking named Ole. They are having a conversationoutside the cave. Ivanka is a bit scared of caves, but because Ole is an experienced cave explorer, he convinces her to enter the cave.

FCIT
كلية الحاسوب وتكنولوجيا المعلومات
Faculty of Computer & Information technology
0010
0000
0000
0111

Rising action: After entering the cave, the Bulgarian girl faces her fear of caves. She discovers that the cave is not so frightening after all. After a suggestion from Ole, she decides to sing a song to listen to the echo.

Climax: The song wakes up a bat in the cave. Ole, who apparently is afraid of bats, is frightened by this and runs into the next hole in the cave. Ivanka, who has conquered her fear of caves, runs after him. He runs in to an opening in the cave which actually is a room with a door. The door closes behind him, and he is trapped. The bat appears again. The bat is, in fact, rather helpful and kind. He tells Ivanka to complete two tasks in the cave to receive two keys to unlock the door and save Ole. This is the turning point in the story.

Falling action: Ivanka meet many scary creatures on her way, but she is able to complete the two tasks and receive the two keys. There is actually a minor “climax” on each task as well, as there is a peak each time she complete a task.

Denouement: After successfully finishing the tasks, she uses the keys to unlock the door and save Ole. Ole is so relieved and thankful to be saved, that he gives Ivanka his Viking helmet. They walk happily together out of the cave.

⇒

# Chapter 3.

3.1 Requirement Analysis:

3.1.1 System Requirements:

**3.1.1.1 System Environment**

**3.1.1.2 Software Information**

**3.1.1.3 System Constraints**

**3.1.1.4 System Modeling**

**3.1.2 Player Requirements (user preferences)**

## 3.1.3 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.

### 3.1.3.1. 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.

⇒

### 3.1.3.2. *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.

⇒

### 3.1.3.3. 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.

⇒

Faculty of Computer & Information technology

### 3.1.3.4. 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.

⇒

### 3.1.3.5. 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.

⇒

### 3.1.3.6. 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. ⇒

**3.2 DETECTION**
**3.2.1 Image acquisition**
**3.2.2 Image processing and hand detection**
**3.2.3 Image Processing Life Cycle**
**3.2.4 Detection Problems and Solutions**

**3.3 RECOGNITION**
**3.3.1 Recognition Notes**
**3.3.2 Hand Gesture Recognition**

**3.3.3 Recognition Problems and Solutions**

**3.4 Event Generation and Applications**
**3.5 Story modeling**

## 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.

FCIT
كلية الحاسوب وتكنولوجيا المعلومات
Faculty of Computer & Information technology

### 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

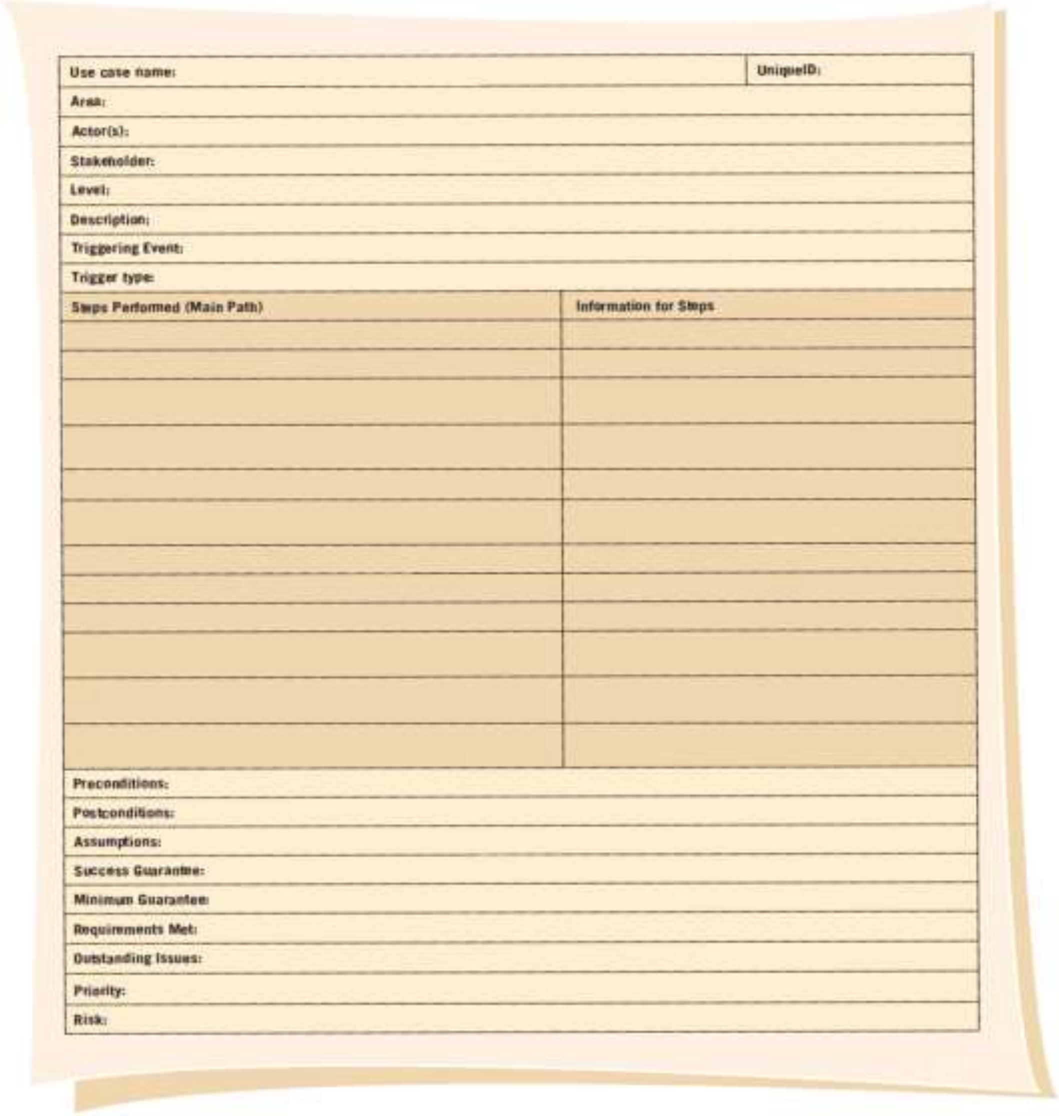
Identify and describe the system's actors

**⇒<Actor1>**

⇒ [Actor 1's description]

كلية الحاسوب وتكنولوجيا المعلومات
Faculty of Computer & Information technology

### 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: | |

## 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

### 3.6 Data modeling (if any)

Add in your ER schema and ERD with their description
⇒

### C. Any other models

FCIT
كلية الحاسوب وتكنولوجيا المعلومات
Faculty of Computer & Information technology

# Chapter 4. Implement the System

## 4.1 Overview

Provide a brief overview of how to develop this project
⇒

## 4.2 Database building (if any)

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 Story Development

Provide the main and essential information about the story development
⇒

## 4.5 Game Playing

Identify the game playing approach and your AI unit and how this will insure the integrated flow of your game
⇒

## 4.6 System Manual

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

## 4.7 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

⇒

## 4.8 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
⇒

كلية الحاسوب وتكنولوجيا المعلومات
Faculty of Computer & Information technology

# 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.