**** 

**Lebanese Canadian University – LCU**

Faculty of Arts and Science – FSA

Bachelor in Interior Design

Final project

**Subject Title**

Presented by

**Student name**

Supervised by

**Mme … and M. ..**

May ??, 2019

Table of Contents

[1 Introduction 4](#_Toc16586292)

[1.1 Overview 4](#_Toc16586293)

[1.2 Scope 4](#_Toc16586294)

[1.3 Goals and Objectives 5](#_Toc16586295)

[1.4 Business Statements 5](#_Toc16586296)

[1.5 Acronyms and Abbreviations 5](#_Toc16586297)

[1.6 Points of Contact 6](#_Toc16586298)

[2 General Design Constraints 7](#_Toc16586299)

[2.1 Mandated Constraints 7](#_Toc16586300)

[2.2 Potential System Evolution 7](#_Toc16586301)

[3 Nonfunctional Requirements 8](#_Toc16586302)

[3.1 Performance Requirements 8](#_Toc16586303)

[3.2 Security Requirements 8](#_Toc16586304)

[3.3 Safety Requirements 8](#_Toc16586305)

[3.4 Legal Requirements 8](#_Toc16586306)

[3.5 Documentation and Training 8](#_Toc16586307)

[4 System Architecture 9](#_Toc16586308)

[4.1 Software Interface 9](#_Toc16586309)

[4.2 Process Diagram 9](#_Toc16586310)

[5 Technical Specifications 10](#_Toc16586311)

[5.1 Admin Module 10](#_Toc16586312)

[5.1.1 Flow Chart 10](#_Toc16586313)

[5.1.2 Features 10](#_Toc16586314)

[5.1.3 ERD Diagram 10](#_Toc16586315)

[5.2 User Module 10](#_Toc16586316)

[5.2.1 Flow Chart 10](#_Toc16586317)

[5.2.2 Features 10](#_Toc16586318)

[5.2.3 ERD Diagram 10](#_Toc16586319)

[5.3 Client Module 10](#_Toc16586320)

[5.3.1 Flow Chart 11](#_Toc16586321)

[5.3.2 Features 11](#_Toc16586322)

[5.3.3 ERD Diagram 11](#_Toc16586323)

[6 Contract 12](#_Toc16586324)

[6.1 Introduction 12](#_Toc16586325)

[6.2 The authors of the software license 12](#_Toc16586326)

[6.3 Preamble 12](#_Toc16586327)

[6.4 Objects delivered to the client 12](#_Toc16586328)

[6.5 Objects not delivered to the client 12](#_Toc16586329)

[6.6 Contract effective date 12](#_Toc16586330)

[6.7 Associated services 12](#_Toc16586331)

[6.8 Additional services 12](#_Toc16586332)

[6.9 Warranty and maintenance 12](#_Toc16586333)

[6.10 Languages 12](#_Toc16586334)

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Date** | **Name** | **Description** |
|  |  |  |  |
|  |  |  |  |

# Introduction

## Overview

This section provides an overview of the requirements document and the system specified. After reading this section the reader should understand the purpose of the document and have a general idea what the proposed system will do.

The design of this template assumes there will be a separate project document and quality assurance test plan that will address issues such as schedule, cost, development methods, development phases, deliverables and testing procedures. If there are other sources of product requirements such as prototypes, user guides or user interface storyboards they should be mentioned and referenced in this section.

You may want to identify the intended audience for the document and describe the individual sections of the document.

***Example:***

This document defines the requirement for the Innovative Publishing system that is being developed for UMKC Faculty. The purpose of this document is to represent the system requirements in a readable way so that clients and stakeholders can understand them and verify them for correctness but with enough detail that developers can design and implement a software system from them.

This document doesn’t address *project* issues such as schedule, cost, development methods, development phases, deliverables and testing procedures. Those are addressed in a separate project document and quality assurance test plan.

The Innovative Publishing system is a web-based tool for publishing content on the Internet. It provides a way for authors to get direct and specific feedback from readers while imposing minimal additional work for authors and readers.

## Scope

The scope defines the boundaries of the product – what it will and will not do. Clients and other stakeholders need a clear understanding what to expect. It’s at the boundaries of the system where there is the most opportunity for misunderstandings regarding what is and is not going to be implemented.

System features are described below so The system features section below does specify exactly what will be included in the system; however, it is not presented in a way that makes clear functionality at the boundaries of the system.

***Example:***

The innovative publishing system will solicit feedback from readers including written comments. Aggregate feedback from readers may be offered directly to other readers (i.e. articles might be rated), but unedited comments from readers will not automatically be made available to other readers. The reason for this is the quality of unedited comments is hard to control.

## Goals and Objectives

Making the goals and objectives of the product explicit guides developers and give direction to the project. During the development of even a small system developers make hundreds of tiny decisions (consciously and unconsciously) not specifically addressed by the formal requirements. Knowing the goals and objectives will help them make the best decisions for the product.

Goals and objectives also help keep the project on track. Without a clear set of goals and objectives the scope of the project can grow or shift as new information arrives. Shifting the focus of the project isn’t bad as long as it is done open and explicitly with everyone in agreement.

***Example:***

The three main goals of the innovative publishing product are:

1. Provide a simple mechanism for readers to offer feedback to authors on content. The rational is that more readers will offer feedback if the process is easy.
2. Use feedback from readers to add value to the content and improve the experience of other readers without intervention from the author. For example, book reviews on Amazon are instant free content that increases the value of the site.
3. The system shouldn’t require any extra effort on the part of the author or other publishing agent.

## Business Statements

## Acronyms and Abbreviations

This section defines potentially unfamiliar or ambiguous words, acronyms and abbreviations. If terms such as “shall”, “should” and “may” are used to indicate importance the meaning of these terms should be defined here.

***Example:***

**Use case** – describes a goal-oriented interaction between the system and an actor. A use case may define several variants called scenarios that result in different paths through the use case and usually different outcomes.

**Scenario** – one path through a use case

**Actor** – user or other software system that receives value from a use case.

**Role** – category of users that share similar characteristics.

**Product** – what is being described here; the software system specified in this document.

**Project** – activities that will lead to the production of the product described here. Project issues are described in a separate project plan.

**Shall** – adverb used to indicate importance; indicates the requirement is mandatory. “Must” and “will” are synonyms for “shall”.

**Should** – adverb used to indicate importance; indicates the requirement is desired but not mandatory.

**May** – adverb used to indicate an option. For example, “The system may be taken offline for up to one hour every evening for maintenance.” Not used to express a requirement, but rather to specifically allow an option.

**Controls** – the individual elements of a user interface such as buttons and check boxes.

## Points of Contact

Provide a list of the point(s) of organizational contact (POCs) that may be needed by the prospective solution provider for informational and coordination purposes.

Include type of contact, contact name, department, telephone number, and e-mail address (if applicable), and project oversight function

Below is a list of Point of Contacts relevant to this project:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Contact Name | Contact Type | Department | Telephone Number | Email | Oversight Function |
|  | HMIS Lead Agency |  |  |  | Implementation  Financial  Training |
|  | CoC Lead Agency |  |  |  | General Governance  High Level Decisions  Problem Resolution |

# General Design Constraints

## Mandated Constraints

Ideally requirements will be specified in terms of functionality needed and developers will have free rein to design and implement a solution. In practice there are constraints on the eventual design and implementation.

Constraints may be mandated technologies. For example, the client may specify that a specific database management system, programming language, and/or operating system be used.

Constraints limit design and implementation options.

Constraints might be absolute, desirable or optional. If constraints aren’t absolute the motivation for the constraint should also be given.

## Potential System Evolution

The resulting software system should be maintainable and extensible. Knowing the types of anticipated changes aids significantly in establishing an architecture that will accommodate the types of expected changes. This section suggests ways the system is likely to be extended or modified in the future.

# Nonfunctional Requirements

Nonfunctional requirements are properties the system must have. Nonfunctional requirements tend to be orthogonal to functional requirements. For example a system may have the nonfunctional requirement that it be offline no more than 15 minutes at a time and not more than ½ hour each week. The realization of this requirements isn’t limited to one spot in the code. This nonfunctional requirement crosscuts some or all functional requirements.

## Performance Requirements

The main performance characteristics are speed and capacity (memory). Performance requirements are usually stated as a function of the number of concurrent users. Use this section to state the performance requirements of the system as a whole. If specific transactions have their own performance requirements state these requirements below along with the description of the feature.

***Example:***

System startup time should be less than 3 seconds. With 30 concurrent users no operation should take more than 5 seconds and 95% of the operations should take less than 2 seconds.

## Security Requirements

Access to data and features may be limited to specific users. There may also be a requirement to keep an audit trail of system use. This section describes the security requirements including the levels and what needs to be protected.

## Safety Requirements

The system may affect the safety of the larger environment. For example, there are limits on the intensity of stray electromagnetic radiation from electronic devices used in hospitals. Potential safety concerns should be investigated and documented in this section.

## Legal Requirements

Some security and safety requirements may also be legal requirements. For example, federal law protects confidentiality of medical records.

***Example:***

Student social security numbers will not be visible to other students.

## Documentation and Training

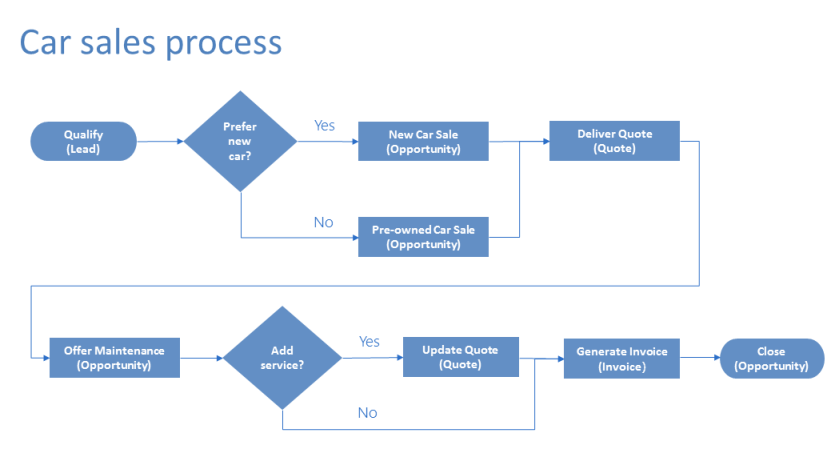
An important part of the total system is the documentation and training that is provided with the system. This section should describe the types and quantity of documentation and training that will be provided with the product.

# System Architecture

## Software Interface

The software interfaces may be locally addressable API’s or remote interfaces using technologies such as web services.

## Process Diagram



# Technical Specifications

The core requirements of the system are listed in this section. This template recommends organizing requirements by features rather than use cases. Features are system behaviors from the user’s point-of-view. The requirements of a feature are described by one or more use cases plus any additional narration that is necessary

Features should be ranked and listed in priority order. Priority is determined by cost, risk and value. To prevent arguments over the exact values of these measures this template recommends using the values: high, medium and low. There should be a written understanding how the priorities listed here are used to determine what order features are delivered and what determines essential features, desirable features and optional features.

## Admin Module

A short description provides an introduction helpful for understanding the module functionality

### Flow Chart

### Features

### ERD Diagram

## User Module

A short description provides an introduction helpful for understanding the module functionality

### Flow Chart

### Features

### ERD Diagram

## Client Module

A short description provides an introduction helpful for understanding the module functionality

### Flow Chart

### Features

### ERD Diagram

# Contract

## Introduction

## The authors of the software license

## Preamble

The introductory part of a statute or deed, stating its purpose, aims, and justification.

## Objects delivered to the client

## Objects not delivered to the client

## Contract effective date

## Associated services

## Additional services

## Warranty and maintenance

## Languages