|  |
| --- |
| School of Science and TechnologyComputing |
| SOFT10101 |
| Project |



Contents

[Project Specification 2](#_Toc30690818)

[Program 2](#_Toc30690819)

[Demonstration 3](#_Toc30690820)

[Report 3](#_Toc30690821)

[Marking Scheme 3](#_Toc30690822)

[Deliverables 4](#_Toc30690823)

[Project proposal 4](#_Toc30690824)

[Project Submission 4](#_Toc30690825)

[Demonstration 4](#_Toc30690826)

[Project Ideas 5](#_Toc30690827)

[Financial applications 5](#_Toc30690828)

[Games 6](#_Toc30690829)

[Language 6](#_Toc30690830)

[Graphics 6](#_Toc30690831)

[Data 6](#_Toc30690832)

[Simulations 6](#_Toc30690833)

[Commerce 7](#_Toc30690834)

[Health 7](#_Toc30690835)

[Maths Applications 7](#_Toc30690836)

[Computer Science 7](#_Toc30690837)

[Management Systems 8](#_Toc30690838)

[Using Libraries 8](#_Toc30690839)

SOFT10101 Project 2022\_23

# Project Specification

Your project is an opportunity to write a more complex program.

The program should

* Exhibit skill in breaking down a complex problem into individual elements
* Specify clear outcomes
* Include a reasonable amount of design thought
* Limit the amount of repetitive obvious coding

Although we would expect the project to have elements that you find challenging, it is also important to work within your own abilities. If you have found the course difficult so far, then focus on the code elements that you are confident in, before moving on to more complex areas.

The degree of sophistication and intrinsic complexity in the problem will be considered during assessment. The program should relate directly to your course title. It will also provide you with additional materials to act as portfolio or discussion topic with interviewers

The Topic for your project can come from ideas that we have provided or idea that you have proposed. The topic must be discussed with your lab tutor and be agreed before you start work on your project.

## Program

You must produce a working program with all source code, executable file and any required other files (e.g. images).

It is expected that your code will include elements that have been taught as part of this module, i.e.

* Appropriate data types
* Conditional statements
* Loops
* Arrays or other collections of data items
* Structs where appropriate
* Functions and Classes
* File handling

Features added purely to give evidence of that feature rather than as an intrinsic part of the design will have a negative effect!

You may use software and libraries from other sources for specific elements, but this software must be fully and correctly attributed in the program source files and in the declaration of ownership.

The degree to which you have relied on existing code, and the ‘added value’ you have put in will form part of the assessment.

## Demonstration

You will be required to demonstrate your code in the labs after the work has been submitted.

Your ability to explain any code you have used in a professional manner will be included in the demonstration as part of the assessment.

## Report

The report must be no more than 8 pages (font no smaller than 9 point). It must include sections on the following:

1. Specification of the aims of the program and the proposed system breaking down into key features required.

2. Design/Implementation – description of the software elements that have to be written to achieve the system (this should include some diagrams eg flow charts, UML such as use case diagram, activity diagram, etc). Any particular implementation problems or issues should be commented on.

3. Test – details of tests that would confirm that the software performs according to the specification (including user faulty input where appropriate). NOTE this is not a transcript of the tests you have undertaken, but a specification of tests that could be applied to your program.

4. Evaluation - what worked, what didn’t and what could have been improved

## Marking Scheme

The GBA Marking Scheme is available in the Assessment Specification.

Marks are awarded for the following categories

* Functionality (55%)
* Object Oriented programming (20%)
* Modularity (10%)
* Comments and documentation (5%)
* Testing (10%)

# Deliverables

All submissions are made to the relevant Dropbox on the module Learning Room.

## Project proposal

You must write a program proposal as agreed with your lab tutor before starting.

The proposal should contain:

* + Title
  + Description
  + Aims and Objectives
  + Software requirements

Upload your proposal to dropbox ( Project Proposal) by Friday 19th Feb 2023.

## Project Submission

You must submit the following:

1. Report

2. Declaration of ownership (which will include specific details of any code sourced from elsewhere)

3. Executable and source code, including .sln and .proj files (to enable the tutor to browse the project structure), and associated text, image files where necessary. Any code sourced from elsewhere must be clearly identified within the source files.

Upload your proposal to dropbox ( Project Submission) by 14:30 on Sunday 16th April 2023

## Demonstration

You will need to demonstrate your project to your lab tutor during labs in the weeks following the submission date.

This Demonstration forms part of the assessment for your project.

# Project Ideas

The following ideas are suggestions that might help in deciding which project to choose. You are welcome to propose other project ideas and discuss these with your lab tutor.

Projects can be written as a console program (i.e. no graphics) or using a graphics library.

Console programs can produce data that could be saved as a file such as a CSV then be examined e.g. via excel. Or as a bitmap file to be displayed in a graphics program.

The best programs will show a lot of thought on the way data is handled, how user errors are coped with, and how the program is structured to make enhancements easier.

Your project should also be something different to your Python Project.

The following ideas are only suggestions and they include project titles from previous years submissions.

## Financial applications

A program to do financial management including mortgage accounts, with compound interest

A program to help manage personal finances

Loan Management Application

Virtual Stock exchange

Banking management system

ATM simulator

Program that allows the user to calculate their retirement funds as well as plan their goals

Virtual Currency Trading platform

## Games

Spelling game

Blackjack game

Adventure game

User input game / test, specified for different ages

2D platforming game

Sudoku

Draughts

Word search

Casino game

A multigame package with the choice of 3 different games

## Language

Constructs stories from semi-random selections of words and numbers (better version creates a document with images too)

Program produces a story from select words and phrases depending on the users input.

Something that parses text from a file and works out some information to help the writer e.g. typical sentence length, checks for bad style e.g. starting with a preposition ( see Gunning Fog Index)

## Graphics

Something that generates a bitmap (raw) image file of something mathematical e.g. fractals, where the user can change the parameters, colour scheme etc. – this can be done via a console program and the file can be separately displayed

## Data

A sorting algorithm is a method for reorganizing a large number of items into a specific order, such as alphabetical, highest-to-lowest value or shortest-to-longest distance.

Student teacher Database

## Simulations

Network simulator that uses a list of nodes and bandwidths and a set of messages of different sizes and routes and generates some performance figures

Football game simulator

## Commerce

Menu ordering system for a restaurant

A database system for a Taxi Company

A sports club management system.

Driving Lesson Booking System

Train Railway Reservation System

Car Renting Software

Property Services Booking System

DVD rental system

Maintain the stock of a store that sells books, music and movies.

Pharmacy Management System

Planning a trip

View current movies at the cinema, view available seats, prices and book tickets

Book a bus ticket to allow the user to travel from one city to another.

## Health

A program that tracks and helps improve the user’s strength at the gym

Nutrition management

Food Nutrition Labelling

gym membership management program

## Maths Applications

Revision aid for geometry

Mathematical Graphing Tool

This program will allow the user to choose different mathematical equations and to input the variables and it will automatically work it out for them

## Computer Science

A neural network that can adapt and learn

Chatbot

Implementation of a functional programming language

A Nintendo Entertainment System emulator

## Management Systems

Harvard reference system

Program to allow company owner to view the profits of the business and the payments of the customers.

Program to allow library owner to view and handle the details of all books that are stored in the system

HR Management System

Program to allocate staffing to each job in a hotel

Phonebook

School Management system

An hourly pay calculator for staff who work on zero hour casual contracts

Work place rota

This system to help manage university data

Employee management system

Music player

# Using Libraries

You can use libraries for your project but you must make this clear in your declaration of ownership.

Remember that you will get very little credit (and a low grade) if all you do is take a standard graphics library tutorial, change background, change the sprites a little bit…

The equivalent of Python module, PyGame, is “Simple DirectMedia Layer” ( SDL)

<https://www.libsdl.org/>

<https://lazyfoo.net/tutorials/SDL/index.php>