|  |  |
| --- | --- |
| **Computer Science Username:** |  |
| **Full Name:** |  |
| **Student ID:** |  |
| **University Username:** |  |

Important: rename this file by removing the ‘RENAME\_ME\_’ part of the filename before submission. This is to ensure that you submit the correct file.

**G52CPP Coursework Documentation File  
(v1.0)**

(Your chance to tell us what you did and what you want us to give you marks for)

**Each section of this document can be extremely brief! Please just add short bullet points and possibly screenshots, not long explanations.**

**READ ALL OF THE RED TEXT. The red text in each section tells you what to include in that section. You can leave the red text in or delete it, that is entirely up to you, but leaving it in may help you in checking that you did everything.**

# Overview

Please give a very brief summary (in bullet points) of what your program does. e.g. is it a fast game, a puzzle game, an editor of some kind, a display tool, etc. Basically, you are telling us in general what it SHOULD do, so that we can judge how well it does what it should.

<Insert your text here>

# Main Screenshot(s)

Please include a screenshot of the main game screen, choosing something which illustrates the game in progress. You can include multiple screenshots to illustrate the game if you wish. A couple of other screenshots will also be included later. Hint: Just use ALT-Print to get copies of the current window for the running program and the paste them into this document.

<Insert at least one screenshot here >

# Usage

Tell us briefly how to use your program. Do we just run it and it all works or do we need to click on anything on the screen, or press any key to make it do things? i.e. you are telling us how you would like it to be tested. This is your chance to walk us through it if it is complex to get started. In general we will expect your program to just run though.

<Insert your text here>

# Known problems

Please mention any problems that you know about with your program. If you don’t mention them then we will assume that you do not know about them, and/or that your testing was insufficient.

<Insert your text here>

# Files

Please provide a list of the files which you added, or modified in the following table, along with a brief statement of what they are used for. You may have the .h and .cpp on one line if you wish.

Files which I added/are mine:

|  |  |
| --- | --- |
| **File name(s)** | **Purpose** |
| <new filename> | <purpose of file, e.g. “main subclass of BaseEngine”> |
| <new filename> | <purpose of file, e.g. “main subclass of TileMananger”> |
| <create new rows > | <Insert extra files into new table rows> |

If you had to change any of the base classes, please provide details of the changes that you made and why. If this was only to do the STL change then you may ignore this, since you will mention his later. If you make a change to the base classes then you need to justify the change and it should work with all sub-classes without modification to the sub-classes. The most common acceptable changes will be generic additions to provide new generic functionality. You should not put code changes or new code which is specific to your program in the framework classes!

Base class files which were modified, and why:

|  |  |
| --- | --- |
| **File name(s)** | **Changes and reasons (i.e. justification)** |
| BaseEngine.h |  |
| BaseEngine.cpp |  |
| DisplayableObject.h |  |
| DisplayableObject.cpp |  |
| FontManager.h |  |
| FontManager.cpp |  |
| JPGImage.h |  |
| JPGImage.cpp |  |
| TileManager.h |  |
| TileManager.cpp |  |
| MovementPosition.h |  |
| Templates.h |  |

# Specific requirements

Consider each of the requirements one at a time and give a brief (bullet-pointed) summary of how you have met the requirement, and why your implementation of it is so good.

If you have failed to implement a feature, or there are known problems with your implementation, then you should include the details here. That way we will know that your testing was not to blame. e.g. if you know something goes wrong under certain circumstances then please say so. Most professional applications have known bugs or problems so this is not a disaster, but you will lose marks for bugs and problems. If you do not mention it here, we have to assume that you thought that the features worked correctly so you will not only lose marks for features which don't work but will also lose marks for not testing it correctly.

For each requirement that you did implement, you should mention how it has been implemented.

This documentation should be in a format which will allow the person marking the coursework to easily identify the various functions (and data members, where applicable) that you added or modified in order to achieve the functionality. **Please be clear and concise rather than wordy. It will be quicker for you to write and quicker for us to read.**

Note: There is no need to explicitly attempt to use specific C++ features if they are unnecessary. E.g. you should not try to alter your program just so that you can put some exception handling in, to prove that you can do so. However, if using a C++ feature is the most appropriate way to handle a problem, you should consider highlighting that you used it (e.g. casting or exception handling) in the relevant sections for the features, or for the complexity or efficiency marking criteria. Your knowledge of these features will be tested in the exam, so I see no need for you to also prove this in the coursework.

Please include a screenshot of the main game screen where appropriate to illustrate your comments, e.g. about your displayable objects or background appearance, choosing something which illustrates the game in progress. You can include multiple screenshots to illustrate the game if you wish.

## Change the framework to use a container class (requirement 1)

**What I did and why I did it this way?**

Insert your text here, justify your decisions (see requirements document) and mention files/functions which you changed.

What container class did you choose? Why?

Did you store pointers or objects in your class? Why?

Did you make the container class object a global, a class member or something else? Why?

Did you keep a pointer to the container class object or the object itself as a variable? Why?

## Implement BaseEngine sub-class object and draw an appropriate background (requirement 2)

**What I did:**

Include any information about your BaseEngine sub-class and the object of that type that you want us to consider in marking.

Mention any interesting features in your implementation of the background. e.g. what do you think was good that you want us to consider and not to risk missing

**Screenshot:**

Please insert a screenshot here illustrating your background

You may include multiple screenshots if you think that it will better illustrate what you have done

## Tile manager class (requirement 3)

**What I did:**

Explain here what your tile manager sub-class does, any complexities in it and any problems that you have (e.g. bugs).

Which class/classes use it?

What does it do and where is it shown?

What changes the tiles (if anything)?

## Player-controlled displayable object (requirements 4 and 5)

**What I did:**

Explain what your player controller displayable object is supposed to do, mention any problems with it (e.g. bugs you know about) and <Insert your text here>

Did you have to do any special coding to make it work well, that you want us to take into consideration?

How does the user/player control it? (Keyboard or mouse? Where is the code to do this?)

## Second displayable object (requirement 4)

**What I did:**

The requirements involve you creating at least two displayable object classes. Explain your second class, what it does and what the class name is.

If you have multiple additional displayable objects, you may describe each one here

## Automation (automatically controlled object or decision making, rqmt 6)

**What I did:**

Explain what automatically controlled object (it could be the second displayable object above) or automated decision making you have in your program.

Explain how it works and anything special that you want us to take into account in marking.

## Load information from files (half of requirement 7)

**What I did:**

Explain what loading of data from files you have.

## Save information to files (half of requirement 7)

**What I did:**

Explain what saving of data to files you have.

## Support different states (requirement 8)

**What I did:**

Explain what code you added and what method you used to create your states.

**States supported and transition methods:**

List the states that you support and make it clear under what conditions the program will change from each state to each other state.

**Screenshots:**

Please insert screenshots here illustrating the different states.

## Have something react and change (requirement 9)

**What I did:**

What do you have which reacts and changes? Under what circumstance does the change occur? Explain what code you added to do this.

# Marking criteria

Since you know the marking criteria, you may want to make some comments about some of them, to point out something which you would like us to take into account in the marking. E.g. is there a requirement which you think was particularly well implemented?

## Code style and readability

If you want to make any comments about the code style that you selected or to justify any decisions then please do so here.

Do you want to explain anything about your code style of structure?

Maybe you used a consistent naming style which we may not have seen before or maybe you have reasons for what seems to be an odd style? If so then here is your chance to comment.

## Efficiency

What part(s) of your program do you think are especially efficient or inefficient?

Are there any justifications that you want to give for things which may appear to be inefficient but are not, explaining why?

Is there particularly efficient code that you want to highlight to us here?

## Robustness, Compilation, Correctness and Reliability

**Known problems with compilation:**

Do you know of any issues with compiling your program? If so then please explain them here.

**Known problems at runtime:**

Do you know of any issues with running your program? Does it crash or hang under certain circumstances as far as you know? Does anything not work how it should, or sometimes fail?

## Problem/Program Complexity (VERY IMPORTANT)

What do you think was particularly complex about your program?

(You can list multiple things.)

Remember that you get marks for achieving complex tasks, not for writing very complex code to achieve something which should have been simple if you had done it in another way.

## Impact (or appearance)

Is there anything especially good or bad about your program, that should make us go ‘wow’? Is there anything about the appearance of your program that you want to draw our attention to? Please feel free to add screenshots to support your arguments if you wish. Here is your chance to persuade us that your program has the ‘wow factor’.

It is especially important to explain it here if you have some ‘wow’ feature which only happens under specific circumstances.

# Additional information

In this section you should make any comments which are useful but did not fit the previous sections.

E.g. you may wish to make a comment about additional features which you added which you would like to have considered in marking, or a justification of why your implementation of something was particularly good, but which did not fit one of the above sections.

Alternatively, you may wish to explain any unusual behaviour or problems that your program has which were not mentioned elsewhere, or anything else not mentioned in the previous sections.

Please do NOT mention anything which was already mentioned elsewhere in this document.