Coursework – documentation guidance

The Report should be used to:

- describe the key aspects of your submission to your lab tutor, who is assessing the work. We are interested in how helpful it is to us understanding your solution;
- communicate your solution – the design and subsequent implementation of that design (include the reasoning behind the decisions you have made).

The aspect addressed in the GBA grid is

	First	2:1	2:2	3	Fail
Report	High quality report giving useful information for an analyst taking the prototype to final production. Good judgement of what to include/exclude	As 2:2 with explanations of advanced functionality and comment on code design and quality.	As 3rd with more depth on structure and Interface design	Report provides justification for basic design decisions, provides class diagrams.	Report missing or insufficient information to understand the reasoning behind design decisions.

The basic rule is " what would be useful in explaining what I have done", in particular with respect to any of:

- the object model – including design patterns
- the file(s)
- the container(s)
- the functionality
  - daily operations (add/edit/delete)
  - weekly operations (sales + reporting)
  - Search/browse functionality

The table below is a guide, not a recipe – it has some suggestions of the items of interest – they are not exhaustive & you may have functionality or perspectives that is not listed.
We try not to be prescriptive about what you should submit because part of the process requires you to judge what is significant/important to the

reader (Giovanna/Patrick/me). It is often the case that "less is more" (or "shorter is better") as students who are not confident about what would be useful tend to put in anything they can think of, which is confusing.

Area	Explain what you have done/used/chosen & what factors guided the decision; aim to identify the key points (don't write an essay).
Files	How many files, what format, how they were processed (library?) Maybe included a few lines of each, annotated to highlight key etc.
Object model	Class diagram (from VS?) illustrating the Inheritance used Identify the extra data in the derived class(es) Virtual functions – identify any used
design patterns	MVC another pattern
Containers	Include the declarations. What is the element / component type(s)? Why were they chosen? Where are they populated? If custom, contrast them with available std alternatives
Search	What search functionality have you implemented and how; the prototype of the search function may be significant; is the functionality re-used ? Would this approach extend to complex queries (with logical connectives)?
System Operation	add/remove/edit projects /materials report details add box office data
Other	Anything else you think important that doesn't naturally fit anywhere above. (such as aspects that were either much easier or much harder than anticipated, but only If you have something to say – don't include for the sake of it)
best bit	what are you most pleased with?

Please email if you want more clarification – but aim to ask specific questions, such as "is it worth including x?" rather than simply "please

clarify”.