General Guidelines:

- Make it easy to understand, clear and concise.
- Structure the report into chapters and sections. Use a consistent referencing style.
- User manual should have task-oriented descriptions and walkthroughs.
- Maintenance manual should include installation instructions, a description of the third party software and where to download it. A list of the packages and files and their uses. A description of how the system can be changed and updated.
-
- In order to get an A, must showcase an understanding beyond what is taught in the course.
-
- Code should be clear, and well commented. Ample testing.
-
- Report must show breadth and depth of knowledge.

White Paper:

- Pages: 9-15
- Vision (1-2pp)
  - Key features and major constraints
  - Must have a clear vision of the product, why we chose it, what's unique, why is it needed
- Initial Business Case (2-3pp)
  - Detail the existing competition, business context, product impact, criteria for success, strategies for commercialisation
  - what problem does this solve for users compared to the existing solutions.
  - Should be clear on who the paying customer is, are the users customers? Or does someone else pay
- Technical Details (3-5pp)
  - Explain product requirements and architecture.
  - Are they clear and adequate?
  - Explain the adopted/needed technologies without giving away critical secrets
- The Future(3-5pp)
  - How do you take the app from a prototype to the finished product
- White paper should include images and diagrams, design flair.

Preliminary Findings Report/Initial Technical Report:

- Pages: 9-15

- Use Case Model (2-3pp)
    - Clearly describe critical functionalities for the functional and non-functional systems
- Candidate System Architecture (2-3pp)
    - Describe the significant decisions about software components and their organisation
- Initial Risk Assessment (2-3pp)
    - Identify the most significant risks and obstacles to successful project completion. Discuss and prioritise these.
    - Explain how these risks were mitigated or removed by early prototypes
- Initial Project Plan (2-3pp)
    - Is the sequence of functionalities adequately described.
    - Include a proposed release plan, sprint plan or roadmap
- Proof of concept (2-3pp)
    - Explain the software development so far, problems, lessons learnt, things that work as expected, things that need to be changed or redone

<u>Proof of concept software:</u>

- Graded on the amount of work done and complexity.
- Implementation contains the core and critical components and algorithms of the application.
- How well does the prototype satisfy client requirements?