- **Abstract**
  - All in one paragraph, if someone, he should know all your work
  - it contains: Introduction, problem statement, procedure, results, limitation and conclusion.
- **Acknowledgment**
- **Table of contents/ figures**
- **Chapter 1 : Introduction**
  - Problem statement (Cite some evidence here)
  - Contribution and objective : what is novel about your work
  - Background information about the topic
- **Chapter 2 : Literature review**
  - Related systems (minimum 3 systems)
  - Research paper (minimum 3 papers)
    - Remember the more, the better. **Related systems or research papers or both.**
  - Compare them and evaluate them to see the missing functionality, or feature and address how your system will overcome them. (For example, some students said the system is not friendly , … You need to show an evidence of this. Not just saying).
- **Chapter 3 : Project management**
  - List of tasks with schedule and deadline.
  - Software development life cycle (SDLC):
    - Waterfall, agile. Justify why you select it?
    - Describe for each process what you did in summary.
      - Requirement, Analysis, Design, implantation and testing
- **Chapter 4 : Analysis**
  - Collection requirement
    - Choose 2:
      - Interview (Highley recommanded),
      - Survey, Questionnaire,
      - Brainstorming (less recommanded) :
        - Remember ask good question: not very general (asking about age and your system for university students)
      - Document analysis (For theme-based projects e.g. TAX system, online payment)
- **Chapter 5 : Design**
  - Archetercual design : System components (server, database ..etc)
  - Data flow diagrams : DFD, UML, ERD for the database. How the system protocol works? Messages sent from and to.. etc.
- **Chapter 6 : Implementation**
  - Describe the tools that used
  - System screenshots, menus, reports with description (core processes)
  - Database implementation: Creation of tables , core queries.
  - Coding: Sample code (core code)
- **Chapter 7 : Evaluation and testing**
  - Testing: Validation testing, UI testing (responsive design), Security testing (against some known attacks).
  - Performance (How fast each process in the system). What was the device spec that run the system.
- **Conclusion**
  - Conclusion
  - limitation (What are the weakness in your work, what has not been finished)
  - Future work

Add introduction before each chapter. It is good to know what this chapter is about.