

# Systems Paper Writing

Nofel Yaseen

## Abstract:

- What needs to be achieved and what will allow it? Give a one line example.
- Current setting, problem (even after taking precautions)
- Current approach, our point of view on current approach
- Our Solution: features, differences, and results
- What drove the solution?

## Introduction:

- Current Setting in detail. Provide Scale but do not put limits unless can be cited/justified.
- Why solve the Problem? List Applications and why they require the problem to be solved.
- Other approaches and limitation.
- Give an example too why current approach is insufficient.
- The trend going in the future and how the limitations will hinder.
- Current requirements. If any complex requirement, how can we make it easier?
- Introduce new technologies.
- Our approach, how it is inspired by the past work.
- Any guarantees system provides? Why are these guarantees needed?
- Throw in a figure to give an overview of the system.
- Features, difference from current from current approach. If there are any drawbacks with our approach, how to overcome?
- Any optimisations performed and when they are useful/useless?
- What do you achieve?
- Challenges faced and solution (also include limitation of the solutions)
- Installation and setup
- How evaluation is done and what do the results show.
- Any interesting observations
- List of contributions

## Background and Motivation:

- Traditional/Dominant design: Show comparison of both.
- Go by problem by problem
- Include the assumptions you take to generalise for your research.
- Include definitions.
- Show your testbed and how it is a reasonable equivalent of the industry.
- Trends observed in the test-bed. Show using graphs. Give reason for these trends.
- Mention features, scale, limitations, throw in figures and examples to illustrate it
- Limitation of features, size, and cost.

- Benefits of our approach

#### Design:

- What do you want to achieve? How does it help? (Design Principles). This also may structure the design section.
- Give an overview of the solution: Key insight, define some definitions, how the solution is different from others
- Any changes to the regular system.
- Divide the solution into stages: Any variables you store and how is each stage different from other works, any assumptions you make. Show figure for edge cases.
- For each stage: What benefits it bring compare to related designs. Why it is well suited in the full design. How this stage will stay relevant in future.
- Show tables to illustrate the impact of design changes.
- Relate to current practices and how it will be useful.
- Nice to show in algorithmic form. Describe the steps too.
- Architectural constraints.
- Extensions

#### Implementation:

- What are you validating?
- Hardware/Software used?
- Resources used in hardware/software?
- Make separate subsection on how did you implement each component of design.
- Include any optimisations you made.

#### Evaluation:

- What are you measuring?
- What are you comparing?
- Resources/Systems used?
- Configuration?
- Setup?
- Workloads
- Results and reasons for the results.
- Limitation of experiments

#### Related Work:

- What work do you build on?
- Any papers, which attempt to solve similar problem?

#### Conclusion:

- Setting, problem, solution, findings and potential.