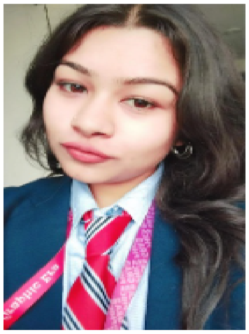
PROJECT AND TEAM INFORMATION

Project Title

(Try to choose a catchy title. Max 20 words).

Optimization of Scheduling Problems using Graph Coloring Algorithms

Student / Team Information

Team Name: *Team #*	*Edge Navigators*
Team member 1 (Team Lead) *(Last Name, name: student ID: email, picture):*	*Anshika Sakalni – 23012076* *anshikasaklani894@gmail.com* 
Team member 2 *(Last Name, name: student ID: email, picture):*	*Akriti Rawat – 23018263* *akritirawat12345@gmail.com*

Team member 3 *(Last Name, name: student ID: email, picture):*	*Manya –230125036* *manyachauhan04@gmail.com*
Team member 4 *(Last Name, name: student ID: email, picture):*	*Rakhi Bisht – 23012177* *rakhibisht24122004@gmail.com*

PROPOSAL DESCRIPTION (10 pts)

Motivation (1 pt)

(Describe the problem you want to solve and why it is important. Max 300 words).

*Scheduling problems, such as exam timetabling and frequency assignment in networks, require efficient resource allocation to avoid conflicts. Graph coloring provides a robust solution by minimizing clashes using distinct color assignments to conflicting entities. This proposal focuses on implementing and optimizing graph coloring techniques, specifically the **Greedy Algorithm**, **Welsh-Powell Algorithm**, and **DSATUR Algorithm**, to enhance scheduling efficiency.*

State of the Art / Current solution (1 pt)

(Describe how the problem is solved today (if it is). Max 200 words).

*Scheduling problems have been extensively studied, and several **state-of-the-art solutions** exist using **graph coloring algorithms, metaheuristics, and AI-based approaches**. These methods optimize resource allocation, minimize conflicts, and improve efficiency.*

Project Goals and Milestones (2 pts)

(Describe the project general goals. Include initial milestones as well any other milestones. Max 300 words).

1. ***Phase 1: Research & Planning***
    - *Conduct literature review on scheduling problems and graph coloring.*
    - *Identify real-world datasets for testing.*
2. ***Phase 2: Algorithm Development***
    - *Implement **Greedy Coloring**, **Welsh-Powell**, and **DSATUR** algorithms.*
    - *Optimize algorithms for improved performance.*
3. ***Phase 3: Testing & Validation***
    - *Apply algorithms to sample scheduling problems (exam timetabling, frequency assignment).*
    - *Compare performance metrics such as execution time and chromatic number.*
4. ***Phase 4: Optimization & Enhancements***
    - *Fine-tune algorithms for scalability and efficiency.*
    - *Implement additional constraints for more realistic scheduling.*
5. ***Phase 5: Documentation & Finalization***
    - *Prepare detailed report including methodology, results, and analysis.*
    - *Develop visualization tools for better result interpretation.*
    - *Present final project outcomes.*

Project Approach (3 pts)

(Describe how you plan to articulate and design a solution. Including platforms and technologies that you will use. Max 300 words).

*The solution will be designed using a **modular approach**, ensuring scalability and flexibility. The system will model scheduling problems as graphs, where tasks are represented as **nodes** and constraints as **edges**. The core implementation will involve three key algorithms: **Greedy Coloring, Welsh-Powell, and DSATUR**, optimized for different scenarios.*

Technologies & Platforms:

- ***Programming Language:** Python (for algorithm implementation)*
- ***Libraries & Tools:** NetworkX (graph processing), Matplotlib (visualization), NumPy & Pandas (data handling)*
- ***Development Environment:** Jupyter Notebook / VS Code*
- ***Dataset Handling:** CSV, JSON files (for real-world scheduling datasets)*
- ***Performance Evaluation:** Time complexity analysis using benchmarking tools*

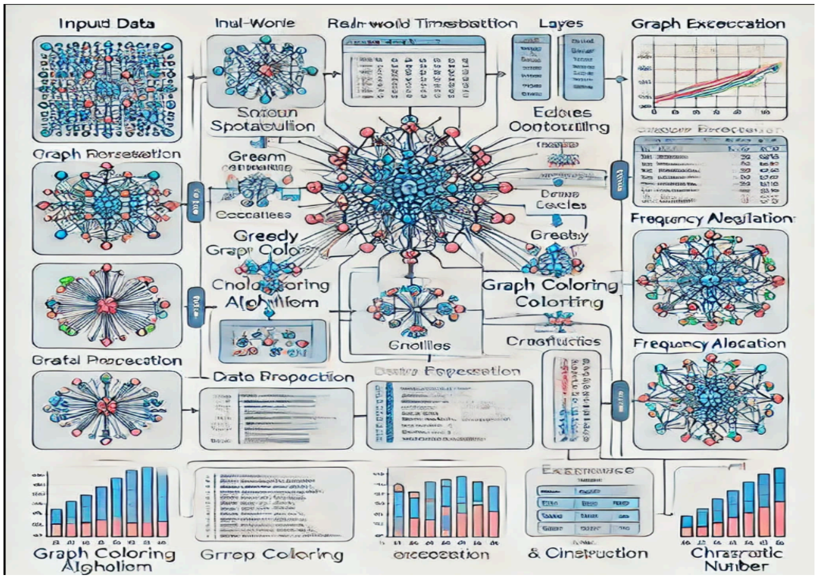
*The system will provide a **command-line and GUI-based interface** for visualization and usability, ensuring accessibility for researchers and educators. Future enhancements may integrate AI-based optimization techniques for dynamic scheduling.*

System Architecture (High Level Diagram)(2 pts)

(Provide an overview of the system, identifying its main components and interfaces in the form of a diagram using a tool of your choice).

The architecture will include:

- ***Input Data Layer** (Real-world scheduling datasets)*
- ***Graph Processing Layer** (Graph representation and algorithm processing)*
- ***Algorithm Execution Layer** (Greedy, Welsh-Powell, DSATUR)*
- ***Output & Visualization Layer** (Schedule generation and visualization tools)*



It outlines the key components, including ***input data, graph processing, algorithm execution, output visualization, and user interaction layers****.*

Project Outcome / Deliverables (1 pts)

(Describe what are the outcomes / deliverables of the project. Max 200 words).

- *A comparative study of* ***graph coloring algorithms*** *in scheduling problems.*
- *A working implementation with* ***visualizations*** *of scheduling scenarios.*
- *Identification of the* ***most efficient*** *algorithm for different use cases.*
- *Insights into further optimization for large-scale scheduling problems.*

Assumptions

(Describe the assumptions (if any) you are making to solve the problem. Max 100 words)

1. *The scheduling problem can be accurately represented as a* ***graph****, where nodes represent tasks and edges represent conflicts.*
2. *The dataset used for scheduling problems is* ***well-defined and structured****, ensuring that conflicts can be appropriately modeled.*
3. *The algorithms will be evaluated under* ***consistent constraints****, assuming that the number of tasks, conflict density, and available resources remain within a defined range.*
4. *There are* ***no hard constraints*** *that prohibit graph coloring from being applied (e.g., specific time slots that must be assigned manually).*
5. *The system assumes that* ***task dependencies*** *can be resolved using graph-based heuristics.*

References

(Provide a list of resources or references you utilised for the completion of this deliverable. You may provide links).

- *Relevant research papers and algorithms on graph coloring*
- *NetworkX library for graph modeling*
- *Applications of scheduling problems in real-world scenarios*