FORTEX36 PROJECT REPORT FORMAT

1. Cover Page

The cover page must include the project title, team name, hackathon name, date of submission, and the complete list of team members with their registration numbers. It must also prominently display the GitHub Repository link. This page acts as the official identity of the project and must look formal and organized.

The GitHub Repository link must be written clearly and be accessible.

2. Abstract

The abstract provides a concise overview of the entire project. It should briefly describe the problem being addressed, the motivation behind solving it, the proposed solution, the technologies used, and the outcome achieved within the hackathon duration. The abstract should allow a reader to understand the essence of the project without reading the full report.

At the end of the abstract section, the GitHub Repository link must be mentioned again.

3. Problem Statement

This section must clearly define the real-world problem the team is attempting to solve. It should explain the context of the problem, who is affected by it, why it matters, and what limitations exist in current solutions. The explanation should highlight the gap your project addresses and justify why the problem is relevant in today's scenario.

The GitHub Repository link must be included in this section.

4. Proposed Solution

The proposed solution section explains your idea in detail. It should describe how the system works from the user's perspective, what major features are implemented, and what makes the solution unique compared to existing approaches. This section must communicate the workflow of the system, interaction between components, and overall project logic.

If applicable, conceptual explanations of automation, AI logic, security mechanisms, or processing pipelines should be included.

The GitHub Repository link must be referenced in this section.

---

5. Technology Stack

This section must describe all the technologies used in the project. It should include the programming languages, frameworks, libraries, databases, APIs, platforms, and tools. Along with listing them, the report must explain why each technology was selected and how it contributes to the project's performance, scalability, and usability.

The GitHub Repository link must be mentioned here as well.

---

6. System Architecture and Design

This section explains the architectural design of the system. It should describe how the frontend, backend, database, APIs, and external services communicate with each other. The data flow, user flow, and control flow should be clearly explained. If diagrams are used, they must be relevant and labeled properly.

If the project includes automation, AI models, security layers, or cloud services, their role in the architecture should be described.

The GitHub Repository link must be included.

---

7. Implementation Details

This section focuses on how the project was built. It should explain the main modules, core logic, important algorithms, API integrations, authentication mechanisms, storage handling, and processing steps. Students should explain how the code is structured in the repository and how different components interact with each other.

The GitHub Repository link must be referenced clearly.

---

8. Results and Demonstration

This section explains what the team successfully achieved. It should describe the working features of the system, outputs produced, performance observations, and screenshots or sample results if applicable. The objective is to show that the proposed solution is actually implemented and functional.

The GitHub Repository link must appear in this section.

---

9. Challenges Faced

This section reflects on the difficulties encountered during the hackathon. It should describe technical challenges, integration issues, debugging problems, time constraints, and learning experiences. Teams should explain how they overcame these issues and what they learned during development.

The GitHub Repository link must be included.

---

10. Future Scope

This section must describe possible enhancements beyond the hackathon. It should explain new features that can be added, scalability improvements, performance optimizations, security upgrades, and real-world deployment possibilities. The aim is to show long-term thinking and innovation.

The GitHub Repository link must be mentioned again.

---

11. Conclusion

The conclusion summarizes the project. It should briefly restate the problem, the solution, the impact created, and the overall achievement during the hackathon. This section should close the report in a professional and confident manner.

The GitHub Repository link must be included.

---

12. References

This section lists all external sources used, including APIs, libraries, documentation, research papers, tools, datasets, and frameworks. Proper acknowledgment of resources is mandatory.

The GitHub Repository link must also be written here.