Project Plan for Comparative Reproducibility Study

Project Overview

- **Project Name**: CodeRunners - Reproducibility Showdown
- **Objectives**: To evaluate and compare the reproducibility of multiple papers from ICSE 2023 and SC24 on large language models (LLMs) for code understanding.
- **Scope**:
- Analyze and compare reproducibility across several papers.
- Create a comparative scorecard for each paper.
- Build a web portal that presents reproducibility metrics for each paper.
- Submit a poster summarizing methodology and results to Gateways 2025.
- **Timeline**: 5 days

Key Milestones and Deliverables

Milestone	Description	Due Date	Deliverables
Milestone 1: Project Kickoff	Team meeting to pick multiple papers, assign roles, and define evaluation criteria.	Day 1	Team roles, paper list, reproducibility criteria framework.
Milestone 2: GitHub Repo Setup	Create project repo with README, scorecard templates, and paper-specific folders.	Day 2	GitHub repo, initial data/ code extraction logs.
Milestone 3: Paper Evaluation	Team members reproduce results independently for each paper.	Day 3	Evaluation notes, issue logs, early scorecard entries.
Milestone 4: Scorecard + Portal	Populate comparative scorecard; build portal to display metrics and visuals.	Day 4	Web portal, final scorecards, screenshot-ready summary.
Milestone 5: Poster + Presentation	Finalize PDF poster and prepare 10-minute team presentation.	Day 5	Poster PDF, presentation PDF, team bios/photos/links.

Resource Requirements

1. **Human Resources**:
2. Paper Analysts (1 per paper)
3. Reproduction Engineers (code testers)
4. Portal/Web Developer
5. Documentation Lead / Scribe

6. Designer for Poster

7. **Technical Resources**:

8. GitHub for source code and versioning
9. Jupyter/Colab for experimentation
10. Streamlit/Flask for portal

11. Canva/Slides for poster

12. **Communication Tools**:

13. Zoom or Meet for daily check-ins
14. Google Drive for document sharing
15. Slack/Discord for real-time messaging

Potential Risks

Risk	Description	Mitigation Strategy
Conflicting Paper Requirements	Different dependencies or model APIs may conflict	Use virtual environments and isolate per-paper experiments
Incomplete Artifacts	Paper may lack full code or data	Document gaps, try alternatives, contact authors if needed
Coordination Overload	Multiple papers = more parallel tasks	Assign 1–2 papers per member, stagger deadlines
Reproducibility Ambiguity	Unclear instructions from papers	Add notes to portal, provide interpretation commentary

Conclusion

This expanded plan allows the CodeRunners team to tackle multiple papers in a structured, parallel way. By assigning papers to individuals, standardizing evaluation criteria, and using a comparative scorecard and portal, the project will contribute meaningfully to the conversation around reproducibility in code-focused research.