

Advantages & Disadvantages

✓ Advantages:

- AI automates requirement analysis, code, and test generation
- Supports multiple languages and frameworks
- Works offline with IBM Granite model (privacy-first)
- Saves developer time and improves accuracy
- User-friendly interface with prompt and PDF support
- Ideal for startups, educational, and enterprise use

✗ Disadvantages:

- Requires system with decent memory/compute for local model
- Limited to predefined prompts and templates (customization needed)
- No live multi-user collaboration support (yet)
- Streamlit UI may lack deep design flexibility
- Initial setup for model loading may take time

Conclusion

The SmartSDLC project successfully demonstrates how AI can transform traditional software development workflows. By automating requirement gathering, code generation, and test creation, it addresses real-world inefficiencies and streamlines the SDLC process. The use of IBM Granite ensures enterprise-grade privacy and reliability, while the modular design provides a scalable foundation for future enhancements.

Future Scope

- Integration with GitHub, JIRA, and CI/CD tools
- Multi-user role support (Developer, QA, Admin dashboards)
- Real-time collaboration and chat history
- Enhanced prompt templates and customization
- IDE plugin for VS Code or IntelliJ
- Cloud deployment with containerization (Docker/Kubernetes)

Appendix

 Source Code

Available in /smartsdlc_app/ folder

 Model link: <https://huggingface.co/ibm-granite/granite-3.3-2b-instruct>

GitHub Repo:

<https://github.com/kesamSravani/SDLC>