SmartSDLC – AI-Enhanced Software Development Lifecycle

Submitted by: Rayani TulasiRam

Institution: St. Ann's College of Engineering and Technology

Course: BTech

Instructor: Ramesh

Date: 25/06/2025

# Abstract

SmartSDLC is an AI-enhanced evolution of SDLC integrating AI into every phase of development, enabling efficiency, accuracy, and automation.

# Introduction

SDLC is a structured approach to software creation including phases such as requirement analysis, design, development, testing, deployment, and maintenance.

# Limitations of Traditional SDLC

- Manual errors  
- Slow requirement gathering  
- Inflexible designs  
- Isolated development/testing  
- Lack of predictive insights

# SmartSDLC: A New Paradigm

SmartSDLC leverages AI and ML to enhance SDLC with analytics, automation, and prediction, ensuring adaptive and accurate development.

# Integration of AI in SDLC Phases

Requirement Analysis:  
- NLP for communication analysis  
- Automatic user story extraction  
- Predictive requirement forecasting  
  
Design:  
- AI-suggested architectures  
- Auto UML diagram creation  
- Simulation of alternatives  
  
Development:  
- Code suggestions (e.g., Copilot)  
- Auto code generation & refactor  
- Ensures coding standards  
  
Testing:  
- Dynamic test case creation  
- Bug detection with root cause  
- Self-healing test scripts  
  
Deployment:  
- Smart CI/CD optimization  
- Predictive scaling & rollback  
- AI monitoring for anomalies  
  
Maintenance:  
- Predictive issue detection  
- Chatbots for support  
- Auto patching suggestions

# Benefits of SmartSDLC

- Faster development  
- Better quality  
- Lower costs  
- High adaptability

# Challenges and Risks

- Privacy/security  
- Setup costs  
- Data quality  
- Ethical concerns

# Real-World Tools & Case Studies

- GitHub Copilot  
- Testim  
- DeepCode  
- AWS CodeGuru

# Conclusion

SmartSDLC is the future, embedding intelligence throughout software creation for faster, smarter, and better results.

# References

- IEEE Software Engineering Standards  
- GitHub Copilot documentation  
- Testim.io case studies  
- AWS CodeGuru whitepapers  
- Research papers on AI in software engineering