

CENTRO ESTADUAL DE EDUCAÇÃO TECNOLÓGICA PAULA SOUZA
FACULDADE DE TECNOLOGIA DE INDAIATUBA
DR. ARCHIMEDES LAMMOGLIA

CAIO SIMONASSI

Modelo de Processo de Desenvolvimento de Software:
Modelo Espiral

Indaiatuba

2024

Modelo Espiral

O Modelo Espiral de desenvolvimento de software foi proposto por Barry Boehm em 1986. Ele combina elementos do modelo cascata e da prototipação para abordar os riscos associados a projetos complexos de software. Seu foco principal é a gestão de riscos em cada fase do desenvolvimento, garantindo que potenciais problemas sejam identificados e tratados antes que afetem o produto final.

Funcionamento do Modelo Espiral:

O modelo é estruturado em ciclos ou espirais, e cada ciclo consiste em quatro fases principais:

1. Planejamento: Identificação dos objetivos do projeto, alternativas e restrições. Aqui são feitos os levantamentos iniciais de requisitos e a viabilidade técnica e econômica é avaliada.
2. Análise de riscos: Em cada ciclo, os riscos associados ao projeto são analisados detalhadamente. Se o risco for considerado muito alto, é possível redirecionar o projeto ou até mesmo cancelá-lo. É uma fase crítica, pois a mitigação de riscos evita falhas futuras.
3. Desenvolvimento e verificação: Com os riscos minimizados, é feita uma fase de desenvolvimento que pode incluir desde a criação de protótipos até partes do sistema final. O produto é testado e verificado ao longo do processo para garantir que os requisitos estão sendo atendidos.
4. Avaliação do cliente e planejamento do próximo ciclo: Após a entrega de uma versão parcial ou um protótipo, o feedback do cliente é coletado para determinar se o processo segue para a próxima iteração. A partir desse feedback, o ciclo recomeça com novos ajustes e refinamentos.

Ele usa uma abordagem “evolucionária” à engenharia de software, capacitando o desenvolvedor e o cliente a entender e reagir aos riscos em cada fase evolutiva. O modelo espiral usa a prototipação como um mecanismo de redução de riscos, mas, o que é mais importante, possibilita que o desenvolvedor aplique a abordagem de prototipação em qualquer etapa da evolução do produto. Ele mantém a abordagem de passos sistemáticos sugerida pelo ciclo de vida clássico, mas incorpora-a numa estrutura iterativa que reflete mais realisticamente o mundo real. O modelo espiral exige uma consideração direta dos riscos técnicos em todas as etapas do projeto e, se adequadamente aplicado, deve reduzir os riscos antes que eles se tornem problemáticos (Pressman, 2006).

A prototipação é vista como um meio de redução de riscos, a permitir que se descubram os problemas potenciais antes de se comprometer com um sistema completo. O modelo caracteriza-se como um gerador de modelo de processo. Cada ciclo do modelo em espiral possui quatro atividades principais:

- Primeiro, os requisitos gerais (login seguro, interface de usuário, etc.) seriam levantados no planejamento.
- Em seguida, os riscos associados, como falhas de segurança, seriam analisados e priorizados.
- Um protótipo inicial de login e algumas funcionalidades de segurança seriam desenvolvidos e testados.
- Após o feedback do cliente, novas funcionalidades, como transferência de dinheiro, seriam adicionadas, e o ciclo seria repetido até o sistema completo estar pronto.

Dessa forma, o Modelo Espiral oferece flexibilidade e permite incorporar mudanças gradualmente, o que é vantajoso para projetos com requisitos não completamente definidos ou suscetíveis a mudanças ao longo do tempo.



Figura Modelo Espiral

Referencias

JOSÉ ARAÚJO. **Paradigmas da Engenharia de Software: Modelo Espiral.** Disponível em: <<https://jaraujof.medium.com/paradigmas-da-engenharia-de-softwaremodel-espiral-843a4cb42feb>>. Acesso em: 18 set. 2024.

LESSA, R.; ORIVALDO LESSA JUNIOR, E. **Modelos de Processos de Engenharia de Software.** [s.l: s.n.]. Disponível em: <<https://ead.uepg.br/apl/sigma/assets/editais/PS0059E0080.pdf>>.

SHARMA, P. **Os 9 principais modelos de desenvolvimento de software para escolher: fases e aplicativos.** Disponível em: <<https://cynoteck.com/pt/blog-post/top-software-development-models-to-choose-from/>>. Acesso em: 18 set. 2024.