

Engenharia de software: Modelo de ciclo de vida RAD.

Claudio Jorge Eckert Junior¹, Iuri Crisnei Kunz¹, Tiago Rios Rocha^{1*}

*Orientador

¹Instituto Federal de Educação, Ciência e Tecnologia do Rio Grande do Sul (IFRS) - *Campus Ibirubá*. Ibirubá, RS, Brasil.

Estudos da área de Engenharia de Software tiveram início na década de 60, a denominada crise do software em que foram expostos recorrentes problemas no desenvolvimento. Pesquisas da *Chaos Report* da *Standish Group* indicaram que em 2013 os projetos analisados obtiveram 48% de atrasos ou prejuízos e 16% de fracassos ou falhas. A engenharia de software tem por objetivo organizar o desenvolvimento através da integração entre pessoas, ferramentas e processos, sendo fragmentada em fases com objetivos definidos, a intercalação das fases são definidas a partir dos modelos de ciclo de vida, estes são estágios percorridos pelo software durante a sua criação, denominados análise, projeto, desenvolvimento, teste, implementação e manutenção. O modelo de ciclo de vida RAD (*Rapid Application Development*) possui as fases de modelagem do negócio, modelagem de dados, modelagem do processo, geração da aplicação e teste. O modelo RAD é apropriado para o desenvolvimento de software que faça uso de classes preexistentes (APIs), que a distribuição do produto seja pequena, o âmbito do projeto seja restrito e possa ser dividido em módulos independentes. Como vantagens ele permite o desenvolvimento rápido entre 60 e 90 dias e dividido em equipes, na qual cada uma recebe parte do software que no final são integradas para formar o todo. Permite a criação e reutilização de componentes, onde o desenvolvimento é conduzido em um nível alto de abstração com maior flexibilidade e redução de codificação manual como atualmente vários *frameworks* suportam, proporcionando um maior envolvimento do usuário com custos reduzidos. Em contrapartida, se tiver altos riscos técnicos, teste de novas tecnologias, interação com outros softwares, dificuldades no gerenciamento do projeto e se uma aplicação não puder ser modularizada com cada função principal completada em menos de três meses, não é aconselhado o uso, para projetos grandes, porém escaláveis, exige-se recursos humanos suficientes para criar o número correto de equipes, aumentando os custos do projeto.

Palavras-chave: *Framework*. Ian Sommerville. Conjunto de vários artefatos. Definição de métodos.