PROCESSOS DE CRIAÇÃO DE APLICATIVOS

Joaldo de Carvalho
Wesley Oliveira
Irlei Rodrigo Ferraciolli da Silva
Rodrigo Clemente Thom de Souza

INTRODUÇÃO

O mundo está dominado pelos dispositivos móveis. A cada dia cresce os usuários de aplicativos, como *WhatsApp*, Agenda, *Doulingo* entre outros. Com o aumento dos *downloads* dessas aplicações, cresce também a procura por *smartphones*, segundo estudo feito pela Teleco em 2013 existia cerca de 271 milhões de celulares no Brasil, ou seja, em média mais de 1 celular por pessoa. Os números são mais assustadores ainda quando falamos do mundo todo, estimasse que exista mais de 6,8 bilhões de celulares em todo planeta Terra. Entra na categoria de dispositivos móveis ainda os *tablets*, relógios *wear,* TVs e carros inteligentes, sendo que os 3 últimos tipos estão em fase de aperfeiçoamento, dessa forma, já temos idéia do tamanho do campo em que estamos entrando, para tudo isso funcionar perfeitamente é preciso muita análise, um dos modelos mais utilizados para o desenvolvimentos de aplicativos em geral, é o planejamento com o Modelo em Cascata (*Waterfall),* apresentado na figura 1.

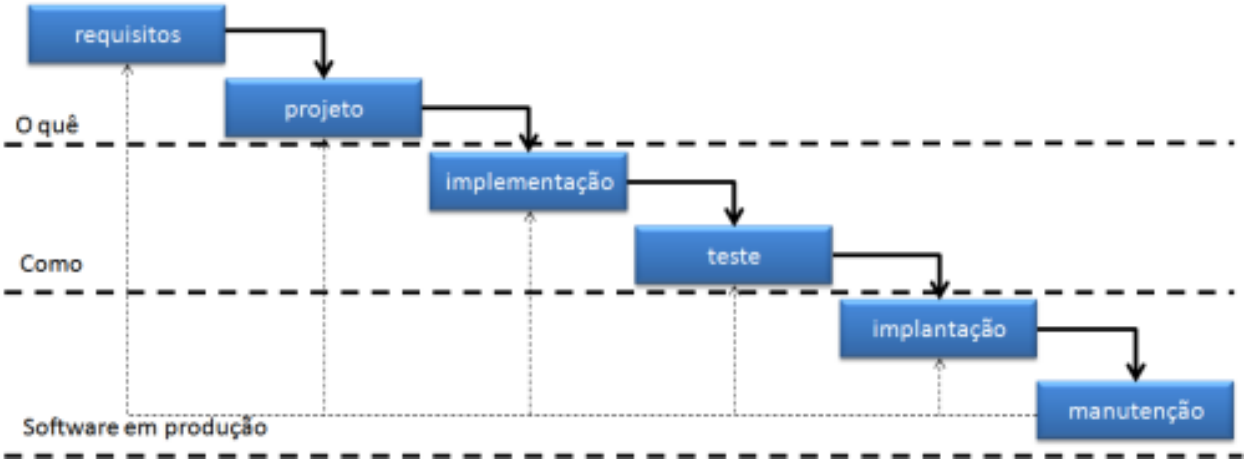

Figura 1 – Ciclo de vida do *software* (Modelo Cascata)

O ciclo de vida de *software* em cascata possui seis fases: a primeira chamamos de Requisitos, é quando os envolvidos se reúnem e definem as funcionalidades do produto, aqui é essencial ficar claro o que realmente o cliente necessita.. A segunda fase é o Projeto (*Design*) que tem como sua finalidade definir detalhes de desempenho, interface, usabilidade, banco de dados, hardware entre muitos outros. Implementação ou codificação, podemos dizer que é o desenvolvimento do aplicativo, ou seja, quando os programadores codificam o que foi requisitado, é de suma importância que as fases anteriores a essa tenham sido realizadas de maneira eficaz, para chegar aos desenvolvedores algo claro. Depois de implementado os requisitos, temos a fase de Testes para detectar possíveis falhas, após isso é feito a Implantação do produto final no ambiente do cliente e ao longo de toda a vida do software é feito manutenções e melhorias solicitadas pelos clientes e identificadas pela equipe de desenvolvedores.

Para deixar essa tarefa de análise mais fácil, é utilizada mundialmente a notação *Unified Modeling Language (UML)*. A UML é uma maneira gráfica de modelar sistemas, facilita no entendimento de todo processo, tanto para o cliente, programadores e gerentes de projeto, é importante também porque toda a documentação do projeto é gerada de forma segura e clara, podendo assim ser consultada por todos. É dividido em diagramas de estrutura e diagramas de comportamento como demonstrado na Figura 2.

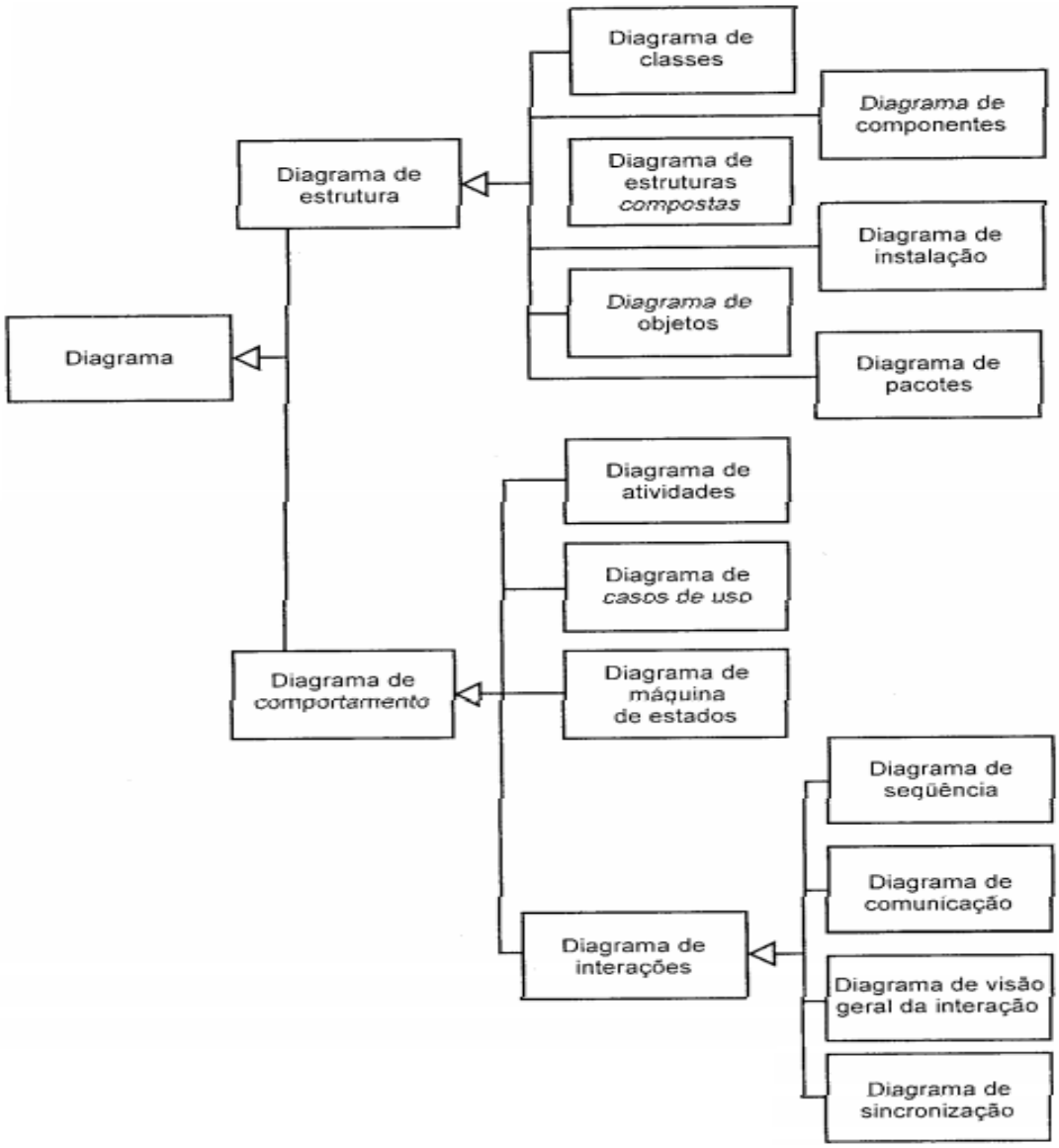

Figura 2 – Diagramas da *UML* 2.0

Para um melhor entendimento sobre UML, abaixo contém uma breve explicação sobre cada um dos diagramas:

Diagramas estruturais:

- Diagrama de Classes: provavelmente é o mais utilizado nos projetos de levantamento de requisitos, sendo assim muito importante, ele é quem define a estrutura de todo sistema, determinando assim as características e ações.

- Diagrama de Componentes: usado para mostrar os relacionamentos entre os componentes e as classes.
- Diagrama de Estruturas Compostas: descreve a estrutura interna de um componente ou de uma classe, detalha como elas se comunicam e apoiam.
- Diagrama de Instalação: determina as necessidades físicas do sistema, como servidores, estações e protocolos de comunicação.
- Diagrama de Objetos: é um complemento do diagrama de classes, ele fornece uma visão dos valores guardados pela cópia das classes na execução do *software*.
- Diagrama de Pacotes: o objetivo é determinar as partes de um sistema ou ainda definir camadas no processo de desenvolvimento, basicamente é a organização feita por meio de pastas.

Diagramas comportamentais:

- Diagrama de Atividades: utilizada para descrever os passos para concluir uma atividade específica.
- Diagrama de Casos de Uso: pode ser considerado o diagrama mais informal da *UML,* ele é utilizado com mais frequência nas fases de requisito e análise, costuma ser consultado durante todo o projeto.
- Diagrama de Máquina de Estados: utilizado para detalhar o comportamento de uma parte do sistema, pode ser utilizado com casos de uso.
- Diagrama de Sequência: identifica um evento no sistema e também o responsável pelo evento, determinando assim como o processo deve ser feito até ser concluído.

- Diagrama de Comunicação: está ligada ao diagrama de sequência, é focado em identificar as mensagens trocadas entre os envolvidos durante o processo.

- Diagrama de Visão Geral da Interação: é uma visão geral dentro de como funciona um sistema ou processo.

- Diagrama de Sincronização: descreve a mudança de uma instância de uma classe, utilizado para demonstrar a mudança de um objeto em resposta a eventos internos.

DESENVOLVIMENTO

No planejamento para dispositivos móveis é importante definir o sistema operacional (S.O), assim como os sistemas *Windows* e *Linux*, presentes em computadores *desktops* e *notebooks*, os dispositivos móveis e portáveis também possuem S.O. Dentre os mais conhecidos estão:

a) *Android*: sistema produzido pela *Google*. Lançado em 2008, atualmente é o S.O para *mobile* mais utilizado no mundo, por conta do crescimento e por ser código aberto, é o ambiente com maior enfoque nesse trabalho.
b) *Windows Phone*: criado pela *Microsoft* com foco na atualização constante de aplicativos conectados a redes sociais e a integração com os aparelhos com o mesmo sistema operacional da empresa;
c) *BlackBerry*: base para os celulares da marca desde 1999, sua principal função é permitir separar as atividades profissionais e pessoais no mesmo celular;
d) *iOS*:S.O dos *iPhones*, produzido pela *Apple*, sua característica é explorar ao máximo a sua potencialidade

A Figura 3 apresenta uma amostra de Janeiro a Outubro de 2013, o sistema operacional *Android* é o mais utilizado com 52%, seguido de perto pelo*iOS*com 41%.

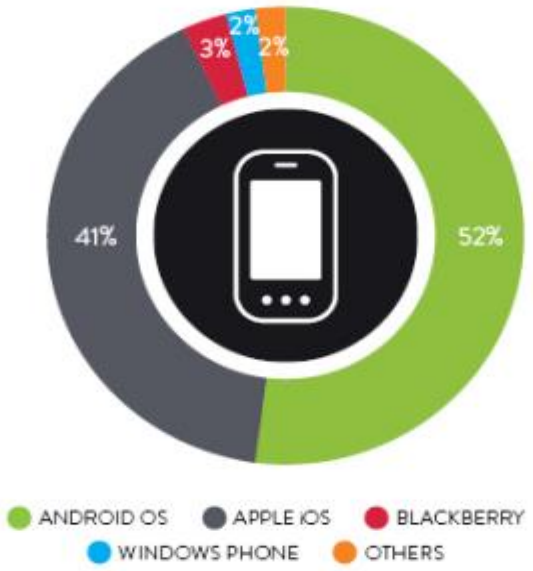

Figura 3 - *Marketshare* de sistemas operacionais *mobile*.

Outro ponto importante no desenvolvimento de aplicativos é a parte de novas tecnologias, onde os envolvidos no processo devem ficar atentos para essas novidades, de forma que possam desenvolver algo com um diferencial, que é essencial na disputa que hoje existe no mercado de dispositivos móveis, aplicativos ficaram obsoletos por não acompanhar as inovações, alguns casos recentes foram o MSN e Orkut. Buscamos listar algumas das tecnologias emergentes que deve ser direcionado uma atenção especial no momento de análise do aplicativo.

- *SmartStay:* é um recurso onde o telefone é inteligente o suficiente para manter a tela ativada por detectar se o usuário continua olhando para a tela. O ecrã desliga-se automaticamente se o usuário não estiver olhando na tela mais.;
- *Android Wear*/TV/Auto: é um sistema operacional gratuito, desenvolvido pelo Google, ele é destinado a dispositivos vestíveis, como os *Samsung Gear Live, LG G Watch* e *Moto* 360, só pra citar alguns. O aplicativo que conecta o relógio inteligente ao celular com *Android* permite executar uma série de tarefas
- Internet das coisas: se refere a uma revolução tecnológica que tem como objetivo conectar os itens usados do dia a dia à rede mundial de

computadores. Cada vez mais surgem eletrodomésticos, meios de transporte e até mesmo tênis, roupas e maçanetas conectadas à Internet e a outros dispositivos, como computadores e smartphones.

- *Cloud*: refere-se à utilização da memória e das capacidades de armazenamento e cálculo de computadores e servidores compartilhados e interligados por meio da Internet.

CONCLUSÃO

O desenvolvimento de aplicativos tem se tornado um mercado muito rentável, crescendo mais a cada dia, com isso a preocupação em desenvolver algo bem documentado e de forma ágil aumenta. Para suprir essa necessidade é utilizado alguns dos assuntos abordados nesse trabalho, a começar pela condução de todo o projeto, utilizando o ciclo de vida de software em cascata, que trata os requisitos, codificação e implementação em etapas distintas. Outra ajuda nesse processo é a *UML* que é composto por diversos diagramas que ajudam desde a especificação dos requisitos até a fase final, onde o aplicativo produzido será liberado para o cliente, é de suma importância um processo de análise bem definido.

Outro ponto importante para se destacar é as novas tecnologias que estão surgindo, algo essencial para aqueles que esperam lucrar com essa área, para ter um aplicativo de sucesso é preciso realizar um estudo minucioso sobre o mercado e tendências, pois como destacamos os aplicativos devem conter novas tecnologias para não ficar obsoleto.

Para melhorar todo esse processo pode ser tomado algumas ações como por exemplo adaptar o modelo de ciclo de vida de software para a realidade de *mobile*, propor alterações nos diagramas da UML ou propor novos específicos para esse tipo de tecnologia, estudo esse que está sendo desenvolvido pelos autores desse artigo.