

FACULDADES INTEGRADAS NORTE DO PARANÁ - UNOPAR
TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO DE SISTEMAS

WALDIR BORBA JUNIOR

**STARTUP DE CONSULTORIA - CONSULTORIA A EMPRESA DE
ROUPAS T-SHIRT**

CONSULTORIA DE TECNOLOGIA COMPUTACIONAL

COLOMBO
2021

WALDIR BORBA JUNIOR

STARTUP DE CONSULTORIA - CONSULTORIA A EMPRESA DE ROUPAS T-SHIRT

Portfólio Interdisciplinar Individual, do Curso de graduação, do Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas – da Faculdades Integradas Norte do Paraná - UNOPAR – EAD Colombo, como requisito parcial para conclusão do segundo semestre.

Orientador: Vanice Dalto
UNOPAR - EAD

COLOMBO
2021

SUMÁRIO

1 – INTRODUÇÃO	1
2 – TAREFA 1: METODOLOGIA DE DESENVOLVIMENTO	2
2.1 Metodologia Tradicionais	2
2.2 Métodos Ágeis	3
2.3 Metodologia de Trabalho Sugerida	4
3 – TAREFA 2: SISTEMAS DISTRIBUÍDOS	6
3.1 Distribuição de Dados	6
3.2 Características dos sistemas distribuídos	6
3.3 Arquitetura Sugerida	8
4 – TAREFA 3: DESENVOLVIMENTO WEB	9
4.1 A Importância da <i>WEB</i>	9
4.2 Funcionamento do Acesso Web	10
4.3 HTML e CSS - Linguagens de Marcação	12
4.4 HTML5	13
4.5 CSS	13
4.6 CSS3	14
4.7 HTML5 e CSS3	14
4.8 Sugestão	14
5 – TAREFA 4: DESENVOLVIMENTO MOBILE	16
5.1 O que é a Tecnologia Mobile	16
5.2 A Importância do Desenvolvimento de Software Móvel	16
5.3 Linguagens de Programação para Desenvolvimento Android	16
5.4 Como funciona o sistema Android	17
5.5 Desenvolvimento Nativo	17
5.6 Desenvolvimento Híbrido	17
5.7 Linguagens de Programação para Android	18
5.7.1 Java	18
5.7.2 LUA (Corona SDK)	18
5.7.3 Kotlin	19
5.7.4 C#	20
5.7.5 JavaScript + CSS + HTML (PhoneGap)	20
5.8 Tendências no Desenvolvimento Móvel	21
5.8.1 Inteligência artificial e aprendizado da máquina	21

5.8.2	Integração com IoT e Dispositivos (<i>wearables</i>)	21
5.8.3	Application Performance Measurement (APM)	21
5.8.4	Enterprise Mobile Management (EMM)	22
5.8.5	Realidade Aumentada	22
5.8.6	Gateway de Pagamentos	22
6	– CONCLUSÃO	23
	Referências	24

1 INTRODUÇÃO

Startup significa o ato de começar algo, normalmente relacionado com companhias e empresas que estão no início de suas atividades e que buscam explorar atividades inovadoras no mercado.

Empresas startup são jovens e buscam a inovação em qualquer área ou ramo de atividade, procurando desenvolver um modelo de negócio escalável e que seja repetível. (SIGNIFICADOS, acessado em: Março 2021)

Em uma sociedade cada vez mais dependente das tecnologias de informação, em que a fazer a informação ser disponibilizada de uma forma mais rápida chega a ser uma questão de sobrevivência. Com isso, saber fazer uso correto da informação é necessário para alcançar objetivos.

Para poder empreender na web de sucesso hoje em dia é uma tarefa um tanto quanto simples, mas ao mesmo tempo trabalhosa, já que demanda os mais diversos fatores, como por exemplo, dedicação, total comprometimento, responsabilidade, entre outros, por parte da pessoa que resolve ter um projeto online e levar adiante com sucesso.

Só assim o empreendedor web consegue fazer o seu trabalho com total sucesso e dessa forma garantir o seu tão almejado reconhecimento e renome no meio online, sempre muito concorrido e disputado, já que a internet conta com sites e blogs dos mais diversos nichos e ramos, e em grande quantidade, o que deixa a concorrência ainda mais acirrada.

2 TAREFA 1: METODOLOGIA DE DESENVOLVIMENTO

Metodologias de desenvolvimento são uma série de técnicas aplicadas durante um projeto de desenvolvimento de software com o objetivo de auxiliar na realização do projeto. Devido a natureza complexa da tarefa, a ausência de uma metodologia pode transformar um projeto de software em um verdadeiro caos. No início se utilizavam o que chamamos hoje de metodologias tradicionais, mais burocráticas e sequenciais, a exemplo do que era feito na engenharia. Com o tempo começaram a ganhar espaço as metodologias ágeis, voltadas para projetos mais dinâmicos, melhorando a resposta a mudanças durante o projeto.

2.1 Metodologia Tradicionais

Quando a computação surgiu, os programas desenvolvidos eram criados para resolver problemas pontuais, como ler informações do disco ou realizar cálculos matemáticos. Com o crescimento da tecnologia, surgimento de sistemas operacionais e sistemas corporativos, criou-se a necessidade de gerenciar estes grandes projetos. As metodologias tradicionais de desenvolvimento apareceram como uma resposta a essa necessidade.

Na criação destas metodologias, os projetos de engenharia foram utilizados como referência. Portanto as metodologias tradicionais seguem o paradigma de coletar requisitos, desenhar, construir e dar manutenção. Sequencialmente. Por isso a primeira metodologia utilizada foi a metodologia em cascata, ou waterfall. Devido a óbvia limitação com relação a mudanças nos requisitos, logo surgiram as metodologias Unified Process (processo unificado) e Espiral.

A metodologia de processo unificado agregava diversas técnicas de construção de software, utilizando diagramas UML. Nesta metodologia, as fases não eram finalizadas antes do início da próxima fase. As fases eram sobrepostas umas as outras, permitindo com que alterações ocorressem durante o projeto. Já a metodologia em espiral era semelhante a metodologia em cascata. Mas uma vez finalizado o desenvolvimento, voltava-se para o início para que novas funcionalidades fossem agregadas ao projeto.

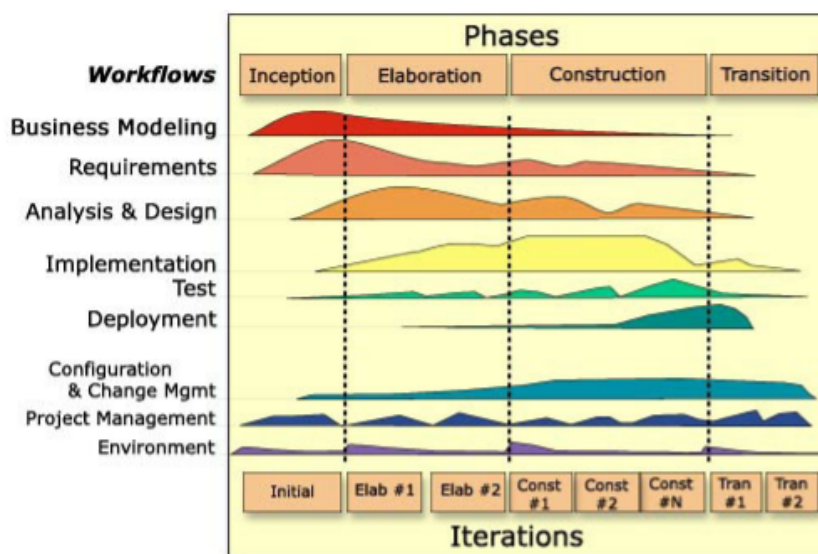


Figura 1 – Ciclo do processo unificado

2.2 Métodos Ágeis

Com a disseminação da computação e a utilização de sistemas para diversas atividades, surgiu a necessidade de tornar o processo de criação de softwares mais dinâmico. O momento chave para a criação dos chamados métodos ágeis foi a criação do Manifesto para Desenvolvimento Ágil de Software, em 2001. A partir do manifesto a ideia de que um software de qualidade e, principalmente, a satisfação do cliente só seriam atingidos com metodologias mais ágeis e menos restritivas ganhou força. Diversas técnicas surgiram, entre elas a programação extrema, o Scrum e o Kanban.

A programação extrema, ou eXtreme Programming (XP), é o oposto aos longos ciclos de desenvolvimento das metodologias tradicionais. O processo XP é caracterizado por ciclos de desenvolvimento curtos, planejamento incremental, feedback e revisões contínuas, além de um design em constante evolução. Com essas características, os desenvolvedores conseguem responder as mudanças com muito mais qualidade. Entre outras técnicas a XP introduziu a técnica de programação em pares, onde dois programadores trabalham em apenas um computador), a técnica de refatoração (melhorias no código, mesmo que ele esteja funcionando) e a simplificação do design (colocando a ênfase no design da solução mais simples para resolver o problema).

O Scrum é um dos métodos ágeis mais conhecidos e utilizados. Ele é um processo iterativo e incremental de desenvolvimento de software. As atividades a serem desenvolvidas no projeto ficam alocadas em um conjunto chamado backlog – nesse caso o backlog do produto. O trabalho é dividido em iterações chamadas sprints. As tarefas que serão executadas em um determinado sprint são definidas em uma reunião de planejamento, sendo movidas do backlog do produto para o backlog do sprint. O Scrum também define reuniões diárias, de apenas 15 minutos, onde todos devem relatar o andamento e se possuem alguma dificuldade que pode

bloquear o processo.

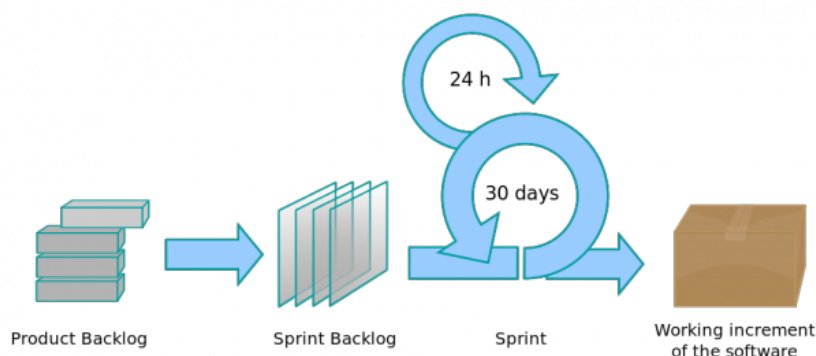


Figura 2 – Processo Scrum

Outra metodologia ágil bastante utilizada é o Kanban. Originalmente criado para a indústria, o Kanban se baseia em três princípios básicos: visibilidade do que deve ser feito, limite do trabalho em execução e fluxo de trabalho melhorado. Um quadro com as tarefas é utilizado para dar a visibilidade e controlar o andamento das tarefas. Pelo dinamismo do Kanban, ele é apropriado para projetos onde as prioridades são constantemente alteradas. (??)

2.3 Metodologia de Trabalho Sugerida

Para a empresa de roupas T-Shirt a metodologia de trabalho escolhida será a metodologia ágil, abaixo serão mencionados algumas das principais vantagens da escolha desta metodologia.

1. Mais independência e produtividade para a equipe

Estruturas ultrapassadas inibem o pensamento criativo e impactam negativamente na produtividade de uma equipe de TI. Quando as Metodologias Ágeis são utilizadas, há uma otimização do tempo e rentabilidade da gestão, visto que não será necessário se preocupar com contratos e documentações extensas e complicadas, ou com ferramentas rígidas que impedem uma resposta rápida para imprevistos.

Assim, a equipe torna-se mais independente e com um melhor preparo para lidar com as resoluções de problemas. Dessa maneira, eles não perdem mais tempo como perdiam utilizando as metodologias tradicionais, e aumentam a produtividade.

2. Melhorias na comunicação

Essas metodologias visam manter um bom relacionamento com os desenvolvedores e clientes. Para que isso aconteça, tudo é pensado para facilitar e otimizar este diálogo, sendo que muitos adeptos preferem as conversas pessoais, a outras formas de comunicação.

Além do mais, essa técnica conta com uma boa prática de feedback. Isso significa que o desenvolvedor sempre estará por dentro das informações do cliente e do código. Essa informação se dá por testes que apontam erros e possíveis formas de melhorias.

3. Melhor definição do objetivo

Com uma maior integração, colaboração do cliente, feedback e softwares, torna-se possível ter uma melhor definição do objetivo, visto que todas as questões principais de uma gestão de TI passam a ser mais claras.

Dessa forma, as Metodologias Ágeis garantem uma nova abordagem, sendo mais adaptada às constantes mudanças que são presenciadas atualmente. Tudo isso colabora para se fazer um planejamento mais eficaz para a gestão de TI, já que os projetos dessa área precisam de soluções muito rápidas e um bom preparo para imprevistos.

4. Melhor atendimento ao cliente

As Metodologias Ágeis acreditam que a atenção deve ser focada nas pessoas, uma vez que são elas que tornam possível que os processos e projetos aconteçam. Por causa disso, fazem de tudo para impulsionar os agentes relacionados aos projetos, principalmente os clientes. O atendimento será melhor, porque a busca pela satisfação do consumidor estará em primeiro lugar.

Como podemos observar, as Metodologias Ágeis proporcionam muitos benefícios para a gestão de TI. Por isso, sua utilização torna-se um importante investimento a ser feito pela empresa T-Shirt.

3 TAREFA 2: SISTEMAS DISTRIBUÍDOS

Os sistemas distribuídos são peças de software usadas para coordenar as ações de vários computadores. Essa coordenação é alcançada através da troca de mensagens, isto é, com pedaços de dados que transmitem informações.

Os sistemas distribuídos requerem componentes simultâneos, uma rede de comunicação e um mecanismo de sincronização. Eles permitem que recursos, incluindo software, sejam compartilhados por sistemas conectados a uma rede. Portanto, o sistema é baseado em uma rede que conecta computadores e lida com o roteamento de mensagens.

A computação distribuída é uma área da computação responsável pela análise de sistemas distribuídos. O programa de computador executado em um sistema distribuído é chamado de programa distribuído.

Em um contexto no qual pode haver centenas ou milhares de computadores, o que é uma proporção comum em grandes empresas de Internet, é muito comum que haja falhas de componentes, sejam elas hardware, rede, discos, etc., e o sistema deve ser preparado para enfrentá-los em todos os momentos.

3.1 Distribuição de Dados

A distribuição é essencial para lidar com *clusters* de dados muito grandes. É necessário obter escalabilidade, que é o meio de manter um desempenho estável quando os conjuntos de dados crescem adicionando novos recursos ao sistema.

Por outro lado, a distribuição apresenta uma série de problemas técnicos que tornam importante o design e a implementação de armazenamento e computação distribuídos. Um ponto a ser levado em consideração é o risco de possíveis falhas.

3.2 Características dos sistemas distribuídos

Compatibilidade

Os dispositivos podem funcionar com diferentes sistemas operacionais. Isso não os impede de oferecer sempre os mesmos serviços aos usuários. Por esse motivo, todos os dispositivos conectados são compatíveis entre si.

Outra questão fundamental é o design do software, porque também é compatível com todos os sistemas e usuários que estão em cada computador.

Tolerância ao erro

Sendo uma rede única com muitos computadores, se algum de seus componentes falhar, os outros poderão continuar executando sua função completamente, evitando erros rapidamente.

Por esse motivo, os sistemas distribuídos tendem a fornecer muita confiança ao trabalhar com eles, porque é muito raro o sistema falhar completamente, porque as tarefas não residem em um único dispositivo, mas em computadores diferentes.

Middleware e API

Processadores diferentes usam *middleware* de distribuição, ajudando a compartilhar diferentes recursos e capacidades para fornecer aos usuários uma rede consistente e integrada. Ele também oferece aos aplicativos diversos serviços, como segurança e recuperação de falhas.

Atualmente, mais se ouve sobre APIs (interfaces de programação de aplicativos), que funcionam como um *gateway* através do qual os aplicativos podem se comunicar. Os aplicativos não precisam saber nada sobre outros aplicativos, exceto sua API.

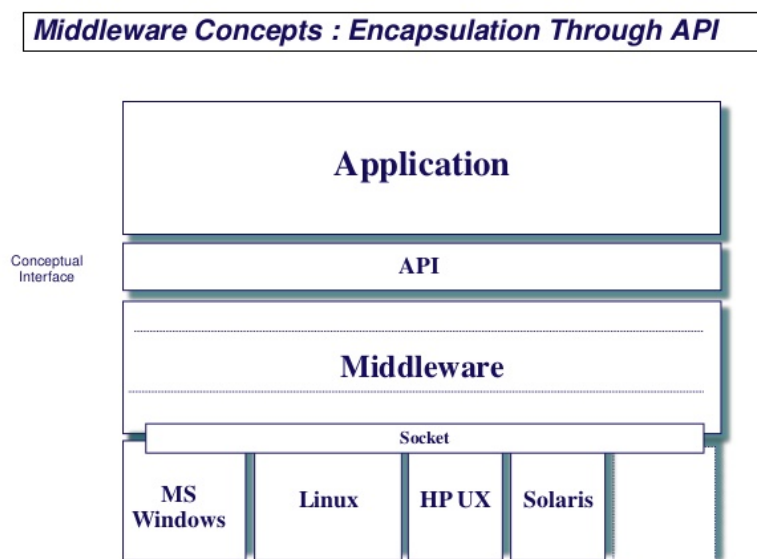


Figura 3 – Middleware

Exemplos comuns de *middleware* incluem *middleware* de banco de dados, *middleware* de servidor de aplicativos, *middleware* orientado a mensagens, *middleware* de web e monitores de processamento de transações. Normalmente, cada programa oferece serviços de sistemas de mensagens para que diversos aplicativos possam se comunicar utilizando estruturas de mensagens como protocolo SOAP, serviços Web, REST (representational state transfer) e JSON (JavaScript Object Notation).

Embora todos os tipos de *middleware* executem funções de comunicação, o tipo que uma empresa escolherá depende de qual serviço está sendo utilizado e qual tipo de informação deve ser comunicado. Isso pode incluir autenticação de segurança, gerenciamento de transações, consultas de mensagens, servidores de aplicativos, servidores da web e diretórios. O *middleware* também pode ser utilizado para processamento distribuído com ações que ocorrem em tempo real em vez de envio e recebimento repetitivo de dados.

3.3 Arquitetura Sugerida

Para a Empresa de Roupas T-Shirt, decidiu-se utilizar a arquitetura distribuída, por ela ser mais dinâmica, no permite otimizar todo o sistema da empresa. Abaixo citaremos algumas das vantagens em optar por esta arquitetura distribuída.

- melhor relação custo/benefício;
- maior capacidade de processamento; maior domínio de aplicações;
- maior confiabilidade;
- maior disponibilidade;
- crescimento gradativo de sua capacidade de processamento.

4 TAREFA 3: DESENVOLVIMENTO WEB

A palavra *web* pertence ao vocabulário da língua inglesa e significa “rede”. Mesmo pertencendo ao léxico inglês o termo é amplamente utilizado em português como muitos outros com a mesma origem. Este conceito é utilizado dentro do espaço da tecnologia para referir-se a um sistema de interligação de documentos e recursos de forma geral, através da internet. O termo tanto pode se referir a uma página de internet, a um lugar virtual (*website*) ou a um servidor. Este documento eletrônico contém informações adaptadas ao espaço em que se encontram de forma que possam ser acessadas por usuários do mundo todo que utilizam um determinado navegador.

Hoje em dia este acesso pode se dar a partir de um computador ou de qualquer outro dispositivo móvel. Uma página web é o nome que se dá a um documento eletrônico. Web é a forma resumida e adaptada de **WWW** (*World Wide Web*). A estas páginas se podem ter acesso através de uma conexão de internet e estão compostas por textos, informações, links e aplicações informáticas. As páginas *web* normalmente funcionam como uma espécie de cartão de apresentação digital. Empresas, pessoas particulares ou organizações, usam este recurso para apresentar suas idéias ou para fornecer informações. As páginas web geralmente se encontram em formato HTML, que é a linguagem utilizada nelas e que permite operar com links, etc.

Por meio delas se pode acessar outras páginas, “navegando” entre *links*. Uma *website* (palavra feita da união dos termos ingleses *web*+*site*) é o lugar onde se reúnem várias páginas webs dentro de um mesmo domínio e relacionadas entre si. O servidor web, por outro lado, é o computador ou a unidade onde se executam determinados software. Ou seja, é um programa utilizado para transferir páginas web. A *world wide web* (*web*) tem sua origem (embora apenas a idéia à principio) em 1980 com o desenvolvimento de um projeto usado para armazenar informações associadas entre elas.

O projeto se denominava ENQUIRE e foi desenvolvido pelo inglês Tim Bernes-Lee. A idéia foi sendo melhorada ao longo dos anos e em 1991 foi publicado oficialmente o primeiro *website*, mas só em 1994 foi dado o grande impulso á comercialização da *web*. Hoje em dia a WWW é um espaço que pode ser acessado de forma globalizadas onde as pessoas podem formar, informar ou informar-se através de dispositivos conectados a Internet. (??)

4.1 A Importância da WEB

Para muitos, a grande conquista do milênio foi o surgimento da Rede Mundial de Computadores. Durante duas décadas a Internet ficou restrita ao ambiente acadêmico e científico e só em 1987 que pela primeira vez, foi liberado o seu uso comercial nos EUA. Mas só 5 anos depois, em 1992, que a rede virou “moda”. Começaram a aparecer, nos EUA, várias empresas provedoras de acesso e no ano seguinte já era comum em universidades, estudantes

criarem suas "páginas" com informações pessoais.

No Brasil, a primeira espinha dorsal conectada à internet eram alguns centros de pesquisas e universidades, além de algumas organizações não governamentais. Em 1994 começaram a funcionar os primeiros serviços web do Brasil e começou a ser testado o acesso discado com fins comerciais. No ano seguinte foi liberado o uso comercial da internet no Brasil e a partir de recursos à disposição dos usuários como facilidade de acesso, o surgimento de provedores e portais de serviços, possibilitaram um aumento considerável de usuários.

A internet passou a ser utilizada por vários segmentos sociais. Os estudantes passaram a buscar informações para pesquisas escolares, enquanto jovens utilizavam para pura diversão e entretenimento. As salas de chat tornaram-se pontos de encontro para um bate papo virtual a qualquer instante, desempregados iniciavam buscas de emprego por meio de sites de agências de emprego ou enviando currículos por *e-mail* e então as empresas, de médio e grande porte, entenderam que a Internet era um excelente caminho para melhorar seus lucros e resultados.

A internet revolucionou o "mundo dos negócios", sendo uma ferramenta estratégica vital para as empresas dos mais variados setores. A cultura empresarial mudou drasticamente, com a comunicação em tempo real através da internet as organizações passaram a responder rápido as novas tendências do mercado, além de encontrar e conquistar seu público alvo com relativa eficiência. Isso se dá através da imensa gama de informação que se pode conseguir na rede, ou mais especificamente nas redes sociais, é possível identificar grupos étnicos, tendências, preferências, opiniões e muito mais, além de levantar dados de qualquer relevância para qualquer negócio que atue no setor comercial. (??)

4.2 Funcionamento do Acesso Web

A *Internet* é uma rede que interconecta computadores e outros dispositivos como o seu celular em escala global para a transferência de dados entre eles. Já a *World Wide Web* é uma aplicação onde páginas são interligadas através de links e que se utiliza da *Internet* para funcionar.

Antes de abordar sobre a Web, precisamos conhecer um pouco sobre rede de computadores. Uma rede de computadores é a interconexão entre computadores que permite a comunicação de dados entre si. Esta comunicação pode ser feita através de cabos ou sem fios.

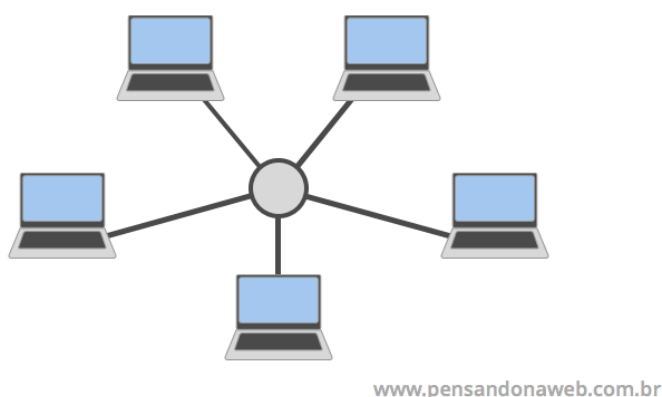


Figura 4 – Rede de computadores

Partindo do princípio que iremos acessar o site da Empresa T-Shirt, devemos abrir o navegador *internet* de preferência e na barra de endereço digitar: www.tshirt.com.br passados alguns segundos a página inicial da empresa é exibida.

O acesso ao *site* da empresa, só foi possível, pois os computadores possuem um endereço numérico único chamado endereço IP. Para que o acesso à página desejada, seja realmente realizado, o computador do cliente, precisa antes estabelecer uma conexão com o computador onde a página solicitada está hospedada.

Para simplificar, iremos chamar **cliente**, o computador utilizado para acesso ao site e **servidor**, o computador aonde a página esta hospedada.

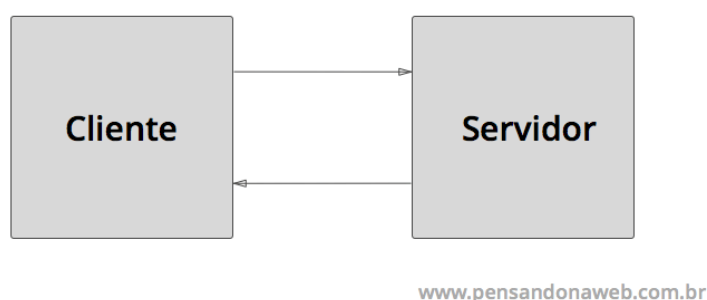


Figura 5 – Diagrama cliente x servidor

Uma vez conhecido o endereço IP do destino a qual deseja se conectar, o cliente precisa estabelecer uma conexão com o servidor.

Este tipo de conexão utiliza o protocolo TCP ou *Transmission Control Protocol* e é através deste protocolo que o cliente e o servidor conversam entre si. Através desta conexão ocorre o envio de pacotes, fragmentos menores dos dados que serão trafegados que contém informações como a porta de origem, a porta de destino e a sequência que devem ser reconstruídos ao chegar no destino.

O que acontece quando a conexão é estabelecida? Existe uma aplicação conhecida como servidor web que recebe e manipula todos os pacotes recebidos.

o HTTP ou *Hypertext Transfer Protocol* e é o idioma que os navegadores e os servidores web conversam. É por meio deste idioma que o seu navegador informa ao servidor *web* qual a sua versão, qual o seu idioma, se aceita conteúdo compactado ou não e qual página foi solicitada. Da mesma forma, é através deste idioma que o servidor web informa ao seu navegador se a página solicitada existe, qual o seu formato.

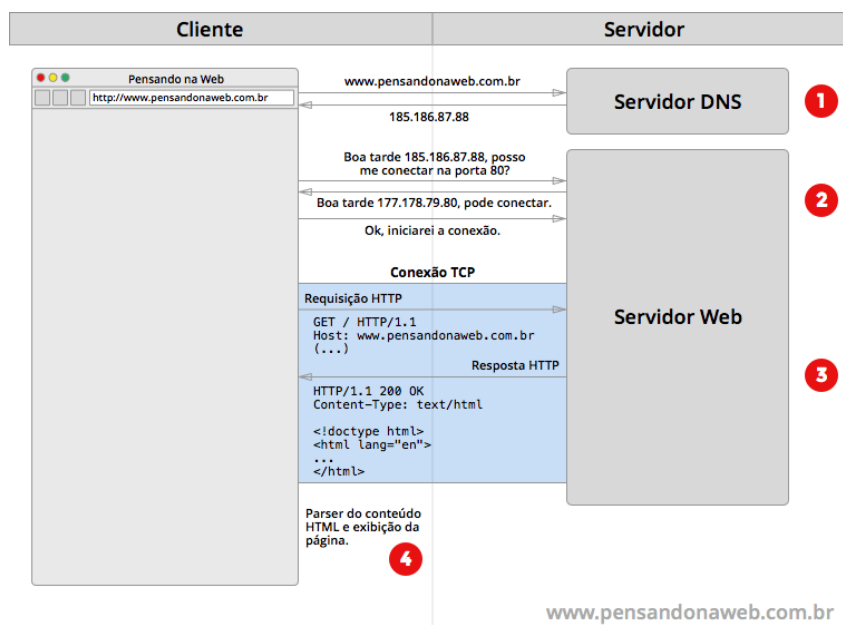


Figura 6 – Arquitetura cliente servidor web

O HTML ou *Hypertext Markup Language* é uma linguagem de marcação utilizada para criar páginas web. Ela foi criada por Tim Berners-Lee, o também criador do protocolo HTTP e da *World Wide Web*, e no início suportava apenas elementos de textos e *links*.

4.3 HTML e CSS - Linguagens de Marcação

Antes de falarmos em HTML5, é necessário detalhar um pouco mais o HTML. HTML significa "Linguagem de Marcação de Hipertexto". Consiste em uma linguagem de marcação utilizada para produção de páginas na web, que permite a criação de documentos que podem ser lidos em praticamente qualquer tipo de computador e transmitidos pela internet.

Para escrever documentos HTML não é necessário mais do que um editor de texto simples e conhecimento dos códigos que compõem a linguagem. Os códigos (conhecidos como *tags*) servem para indicar a função de cada elemento da página *Web*. Os tags funcionam como comandos de formatação de textos, formulários, links (ligações), imagens, tabelas, entre outros.

Os *browsers* (navegadores) identificam as *tags* e apresentam a página conforme está especificada. Um documento em HTML é um texto simples, que pode ser editado no Bloco de Notas (Windows) ou Editor de Texto (Mac) e transformado em hipertexto.

4.4 HTML5

Entre 1993 e 1995, o HTML ganhou as versões HTML+, HTML2.0 e HTML3.0, onde foram propostas diversas mudanças para enriquecer as possibilidades da linguagem. Contudo, até aqui o HTML ainda não era tratado como um padrão.

Quando o HTML4 foi lançado, o W3C alertou os desenvolvedores sobre algumas boas práticas que deveriam ser seguidas ao produzir códigos client-side. Desde este tempo, assuntos como a separação da estrutura do código com a formatação e princípios de acessibilidade foram trazidos para discussões e à atenção dos fabricantes e desenvolvedores.

Contudo, o HTML4 ainda não trazia diferencial real para a semântica do código. O HTML4 também não facilitava a manipulação dos elementos via Javascript ou CSS. Se você quisesse criar um sistema com a possibilidade de Drag'n Drop de elementos, era necessário criar um grande script, com bugs e que muitas vezes não funcionavam de acordo em todos os browsers.

O HTML5 é a nova versão do HTML4. Enquanto o WHATWG define as regras de marcação que usaremos no HTML5 e no XHTML, eles também definem APIs que formarão a base da arquitetura web. Essas APIs são conhecidas como DOM Level 0.

Um dos principais objetivos do HTML5 é facilitar a manipulação do elemento possibilitando o desenvolvedor a modificar as características dos objetos de forma não intrusiva e de maneira que seja transparente para o usuário final.

Ao contrário das versões anteriores, o HTML5 fornece ferramentas para a CSS e o Javascript fazerem seu trabalho da melhor maneira possível. O HTML5 permite por meio de suas APIs a manipulação das características destes elementos, de forma que o website ou a aplicação continue leve e funcional.

O HTML5 também cria novas tags e modifica a função de outras. As versões antigas do HTML não continham um padrão universal para a criação de seções comuns e específicas como rodapé, cabeçalho, sidebar, menus e etc. Não havia um padrão de nomenclatura de IDs, Classes ou tags. Não havia um método de capturar de maneira automática as informações localizadas nos rodapés dos websites.

O HTML5 modifica a forma de como escrevemos código e organizamos a informação na página. Seria mais semântica com menos código. Seria mais interatividade sem a necessidade de instalação de plugins e perda de performance. É a criação de código interoperável, pronto para futuros dispositivos e que facilita a reutilização da informação de diversas formas.

4.5 CSS

CSS é chamado de linguagem *Cascading Style Sheet* e é usado para estilizar elementos escritos em uma linguagem de marcação como HTML. O CSS separa o conteúdo da representação visual do site. Pense na decoração da sua página. Utilizando o CSS é possível alterar a cor do texto e do fundo, fonte e espaçamento entre parágrafos. Também pode criar tabelas,

usar variações de *layouts*, ajustar imagens para suas respectivas telas e assim por diante.

CSS foi desenvolvido pelo W3C (*World Wide Web Consortium*) em 1996, por uma razão bem simples. O HTML não foi projetado para ter tags que ajudariam a formatar a página. Você deveria apenas escrever a marcação para o site.

4.6 CSS3

Apesar de lançada em 2010, CSS3 é a última versão da Folha de Estilo em Cascata e veio para acrescentar de forma melhorada das versões anteriores.

A melhor novidade é em relação a flexibilidade na criação de layouts, trazendo mais autonomia para os webdesigners e também desenvolvedores, que de certa forma estão ligados ao visual do site.

Com o CSS3, é possível elaborar cantos arredondados, sombras, efeitos gradientes, animações e efeitos de transição, dentre outras opções.

4.7 HTML5 e CSS3

O HTML5 foi desenvolvido para atender as crescentes demandas apresentadas pelas necessidades atuais da mídia, cross-device e internet móvel. Podemos dizer que é uma ótima ferramenta para desenvolvimento de aplicativos móveis multiplataforma porque muitos dos seus recursos foram adaptados com a consideração de acesso em dispositivos de baixa potência, incluindo Tablets e Smartphones.

Para uma adição, o HTML5 oferece uma interface comum para tornar os componentes de carregamento mais simples. Por exemplo, o HTML5 não requer o plugin Flash porque o elemento será executado sozinho.



Figura 7 – HTML5 CSS3 JS6 - adaptada pelo autor

4.8 Sugestão

Com o intuito de permitir uma rápida expansão Empresa de Roupas T-Shirt, definiu-se em utilizar o HTML5 e o CSS3, que irá propor um desenvolvendo e expandindo do conteúdo na

web e aplicativos mobile, que podem acessar em diferentes dispositivos, navegadores e sistemas operacionais.

5 TAREFA 4: DESENVOLVIMENTO MOBILE

O desenvolvimento mobile é um dos nichos que mais crescem no mercado da tecnologia.

Dentro das corporações, a tecnologia mobile proporciona uma série de benefícios, entre eles o acesso mais rápido e prático às informações — o que facilita a análise de dados, o gerenciamento e a tomada de decisão.

5.1 O que é a Tecnologia Mobile

A tecnologia *mobile* é aquela que permite a movimentação do usuário durante o seu uso.

No meio corporativo, essa mobilidade possibilita uma gestão mais assertiva sobre os negócios uma vez que permite ao usuário administrar a empresa a qualquer hora e em qualquer lugar.

Então, o desenvolvimento mobile está relacionado a criação dessa tecnologia.

Por ser cada vez mais importante no dia a dia de pessoas e empresas, a tecnologia mobile deixou de ser vista apenas como um recurso e passou a ser uma necessidade.

Hoje é possível integrar o *desktop* ao celular, de forma inovadora, e assim aumentar a produtividade e oferecer uma experiência personalizada para o cliente.

5.2 A Importância do Desenvolvimento de Software Móvel

A popularização dos *smartphones* tem contribuído para mais pessoas utilizarem dispositivos móveis.

Como a tendência é que os usuários se tornem cada vez mais mobile e as empresas recorram a tecnologia para otimizar funções, caberá aos desenvolvedores criar apps capazes de atender essa demanda.

Não só para otimizar tarefas e ajudar na gestão corporativa, o desenvolvimento de *software* móvel também é importante para manter o cliente um passo à frente da concorrência.

5.3 Linguagens de Programação para Desenvolvimento Android

Muitas empresas têm feito dos aplicativos o seu principal negócio e meio de comunicação com o cliente. Mas quais são as habilidades necessárias para se desenvolver um aplicativo Android do zero? Há algumas linguagens de programação que são mais indicadas do que outras para a tarefa, ao mesmo tempo, vale lembrar que é preciso um roteiro bem estruturado para não cometer os erros mais comuns ao desenvolvê-lo.

5.4 Como funciona o sistema Android

O Sistema operacional Android, criado pela Android Inc em 2003 e adquirida pela Google em 2005, usa um típico Kernel Linux.

Este Kernel conta com algumas modificações, principalmente quando se trata de gerenciamento agressivo de memória, consumo de energia e outras melhorias específicas para dispositivos móveis.

Todas as linguagens de programação para Android contam com, ao menos, um de dois kits de desenvolvimento. O primeiro é o NDK (Kit de Desenvolvimento Nativo).

Este pode ser manipulado utilizando C/C++ ou Java.

O segundo kit de desenvolvimento é o Android SDK (Kit de Desenvolvimento de Software ou DevKit).

Trata-se de uma plataforma de mais alto nível, responsável pelo ecossistema de desenvolvimento da plataforma.

O Android SDK inclui projetos de exemplo com código fonte, ferramentas de desenvolvimento, emulador e outras bibliotecas que auxilia os desenvolvedores a utilizarem a plataforma.

Antes de continuar, precisamos definir as duas formas possíveis de desenvolvimento: desenvolvimento nativo e desenvolvimento híbrido.

5.5 Desenvolvimento Nativo

Em desenvolvimento nativo, é necessário um especialista na linguagem específica daquela plataforma que se deseja gerar um aplicativo, seja Android ou iOS.

Também possui acesso à todos os recursos do dispositivo, com bom desempenho (acelerômetro, giroscópio, geolocalização, etc), como são desenvolvidos especificamente para cada plataforma, explora muito bem toda a UI.

5.6 Desenvolvimento Híbrido

Já em desenvolvimento híbrido, não é necessário um especialista em linguagem nativa, muitas vezes utiliza-se linguagens que não são específicas daquela plataforma, e com um único código fonte, pode-se gerar aplicativos para plataformas diferentes (Android e IOS).

Respeita-se a UI do sistema operacional do dispositivo, porém, utiliza um navegador embutido no aplicativo para demonstrar ao usuário.

Cada uma dessas modalidades tem seu espaço no mercado, basicamente esse pode ser considerado mais um indicador, busque também por vagas em mobile, só que levando em conta que produto te agradaria mais trabalhar.

5.7 Linguagens de Programação para Android

5.7.1 Java



Figura 8 – Java

É uma linguagem que utiliza o paradigma Orientado a Objetos, e mais recente, tem trabalhado forte em inserir elementos de programação Funcional em seu escopo.

É tido como a linguagem oficial para o desenvolvimento do Android e é suportado pelo Android Studio.

Pontos positivos do Java

- Muitos tutoriais e dicas na internet sobre a mais variada gama de assuntos da linguagem;
- Uma das linguagens mais usadas para fazer aplicativos Android;
- Suporta Android Studio;
- É versátil.

Pontos negativos do Java

- Curva de aprendizado é bastante íngreme;
- Sobrecarga de conteúdo pode mais atrapalhar que ajudar se você não sabe o que procura;
- Não é a linguagem de computador mais amigável para leitura;
- Atualizações frequentes dificultando o processo das atualizações.

5.7.2 LUA (Corona SDK)



Figura 9 – LUA (Corona SDK)

Esta é uma das opções mais simples para quem quer desenvolver aplicações para Android.

Estamos falando da linguagem de *scripting* Lua, poderosa e fácil de aprender, com Corona SDK.

Caso não conheça, Corona é um *framework* que lhe deixa criar jogos e aplicativos para *mobile*, desktop, TVs e tudo mais que rode Android.

Existem alguns benefícios em usar o Corona SDK para desenvolver aplicativos Android. Os devs por trás da Corona dizem que seu produto permite que os desenvolvedores trabalhem em aplicativos móveis com uma velocidade dez vezes maior.

Pontos positivos de usar LUA com Corona SDK

- Simples, fácil de aprender e poderoso;
- É uma linguagem de programação muito rápida de usar;
- Suporte para todas as bibliotecas nativas, podendo publicar em várias plataformas;
- Emulador que permite ver sem compilar o código;

Pontos negativos de usar LUA com Corona SDK

- Precisa de um editor para o código;
- É limitado em comparação com outras linguagens;
- Exige pagamento para a utilização de alguns recursos.

5.7.3 Kotlin



Figura 10 – Kotlin

Kotlin é uma linguagem de programação reconhecida pela Google que pode ser usada como alternativa ao Java para desenvolvimento de aplicativos Android. E geralmente é isso mesmo que ocorre.

Ele pode interoperar com Java e é executado na Java Virtual Machine.

Pontos positivos do Kotlin

- Suporte a Java Virtual Machine;
- Não faz com que o tamanho dos arquivos aumente;
- Não causa desaceleração;
- É simplificado;
- Ideal para começar com Android;
- Suporta Android Studio.

Pontos negativos do Kotlin

- Embora seja fácil de aprender, não é tão fácil quanto outras linguagens citadas;
- O suporte da comunidade é bem menor, por não ser tão popular.

5.7.4 C#



Figura 11 – C#

C# é bastante semelhante ao Java e por isso é indicado para o desenvolvimento de aplicativos Android.

Outro motivo para aprender C# se dá pelo fato de que hoje é a uma das mais importantes linguagens do mercado *enterprise*.

É também uma linguagem Orientada a Objetos, mas com uma sintaxe mais simples do que Java, o que torna a codificação aparentemente mais fácil.

Pontos positivos do C#

- Simples de programar;
- Fácil de ler e entender;
- Orientado a objetos;
- Similar ao C++;
- Sem problemas de memória graças à coleta de lixo;

Pontos negativos do C#

- C# e Unity são excelentes para jogos 3D, mas não tão bons para desenvolver aplicativos padrão juntos;
- Não está de acordo com a linguagem do Material Design do Google;
- Existe menos liquidez no mercado para desenvolvedores profissionais de Android em C#.

5.7.5 JavaScript + CSS + HTML (PhoneGap)



Figura 12 – JavaScript + CSS + HTML (PhoneGap)

Os aplicativos Android também podem ser criados com HTML, CSS e JavaScript usando Adobe PhoneGap no Apache Cordova.

O framework PhoneGap permite criar aplicativos híbridos que são mostrados por meio do “WebView”, mas empacotados como um aplicativo.

A estrutura do PhoneGap dificilmente requer muita programação, exceto para JavaScript.

O PhoneGap, basicamente como o nome sugere, apenas preenche uma lacuna, dando aos desenvolvedores de aplicativos Android acesso a recursos do telefone, como a câmera ou o acelerômetro.

Pontos positivos do JavaScript + CSS + HTML com PhoneGap

- Permite aos designers, front-ends ou qualquer pessoa que crie sites interativos usando HTML, JavaScript e CSS usar recursos mais simples do Android;
- Pode ‘quebrar um galho em situações’ bem específicas.

Pontos negativos do JavaScript + CSS + HTML com PhoneGap

- Para tarefas básicas, ele dará conta, mas não para a maior parte das funções;
- Esta opção não está presente no ‘cinto de utilidades’ do desenvolvedor Android profissional.

5.8 Tendências no Desenvolvimento Móvel

Acompanhar todas as tendências também pode ser um desafio. Afinal, a tecnologia passa por mudanças constantes e toda essa oscilação vai exigir conhecimento para a software house não correr o risco de levar um produto ultrapassado para o cliente.

Para se manter atualizado, veja agora algumas tendências no desenvolvimento mobile:

5.8.1 Inteligência artificial e aprendizado da máquina

Oferecer funcionalidades inteligentes é o que vai agregar valor a sua solução mobile. Portanto, investir em inteligência artificial e aprendizado da máquina é importante para deixar o seu produto mais moderno e valorizado em relação ao dos concorrentes

5.8.2 Integração com IoT e Dispositivos (*wearables*)

Wearables são dispositivos que podem ser “vestidos” ou integrados ao corpo do usuário (relógios inteligentes, pulseiras e óculos de realidade virtual).

O sucesso dessa tecnologia está atrelado à integração do sistema operacional com a Internet das coisas, que permite o gerenciamento de diversos objetos digitais.

5.8.3 Application Performance Measurement (APM)

Application Performance Measurement está relacionado ao monitoramento e gestão de desempenho e disponibilidade de aplicativos. Se o desenvolvedor não fizer esse gerenciamento, ele não consegue garantir a disponibilidade e nem a qualidade do *software*.

5.8.4 Enterprise Mobile Management (EMM)

Enterprise Mobile Management é o gerenciamento de mobilidade corporativa. À medida que mais empresas compram dispositivos e buscam suporte para o uso dessas ferramentas, o EMM se torna cada vez mais necessário.

5.8.5 Realidade Aumentada

Realidade aumentada está relacionada à integração de elementos ou informações virtuais ao dispositivo móvel. Essa tecnologia oferece experiências mais ricas ao usuário uma vez que é usada para otimizar ambientes ou situações do cotidiano.

5.8.6 Gateway de Pagamentos

Gateway de pagamento é um serviço destinado a empresas de grande porte, lojas virtuais e sistemas SaaS. No dispositivo, uma operadora financeira é responsável por autorizar pagamentos de transações online em *websites* de empresas ou pessoas físicas.

6 CONCLUSÃO

Através do estudo de caso observou-se que uma definição correta de startup é uma empresa que possui um modelo de negócios repetível e escalável. Elas são inovadoras e, normalmente, usam a tecnologia para o seu funcionamento. Também são compostas por um grupo de pessoas com perfil e mindset empreendedor.

A Empresa de Roupas T-Shirt com a assessoria Startup, observou que existe um esquema chamado Ciclo de Feedback, que consiste em um processo de três etapas: construir, medir e aprender. Assim, o primeiro passo é identificar um nicho de mercado e construir uma solução na forma de MVP; então, medir os resultados com base nas opiniões dos clientes, em seguida, é preciso aprender com os erros e construir algo melhor e assim sucessivamente.

Observou-se que a análise de sistemas sendo uma atividade que tem como finalidade a realização de estudos de processos a fim de encontrar o melhor caminho racional para que a informação possa ser processada foi primordial para a Empresa. O analista de sistemas deve servir como um tradutor entre as necessidades do usuário e o programa a ser desenvolvido pelo programador.

Referências

GEEKHUNTER. **Android Melhores Languages de Programação**. acessado em: Março 2021. <<https://blog.geekhunter.com.br/linguagens-de-programacao-para-android/>>. Nenhuma citação no texto.

ILUSTRADDEV. **O que é CSS3?** acessado em: Março 2021. <<https://ilustradev.com.br/o-que-e-css3-como-funciona-estilizacao-com-css/>>. Nenhuma citação no texto.

PAULO, E. Q. S. **Metodologias de Desenvolvimento**. acessado em: Março 2021. <<https://queconceito.com.br/web>>. Nenhuma citação no texto.

RAMOS, A. D. S. **A Importância e a Evolução no Desenvolvimento Web**. acessado em: Março 2021. <<https://www.scribd.com/document/215199079/A-IMPORTANCIA-E-EVOLUCAO-DO-DESENVOLVIMENTO-WEB>>. Nenhuma citação no texto.

SIGNIFICADOS. **Significado de Startup**. acessado em: Março 2021. <<https://www.significados.com.br/startup/>>. Citado na página 1.

SOFTERIZE. **Metodologias de Desenvolvimento**. acessado em: Março 2021. <<https://www.softerize.com.br/blog/metodologias-de-desenvolvimento/>>. Nenhuma citação no texto.