#as referências citadas estão na documentação que já foi entregue, usarei as mesmas..

**4 Fundamentação Teórica**

A Web 1.0, era uma plataforma, onde usuários, no caso espectadores, podiam apenas navegar entre as páginas e visualizar seu conteúdo; não sendo permitido nenhum tipo de interação com o serviço, como por exemplo, reeditar ou alterar algum conteúdo exposto na página.

        Além disso, a grande maioria de serviços prestados na Internet, eram liberados mediante a pagamentos, ou seja, para usufruir de algum conteúdo, era necessário realizar um pagamento e uma licença única e restrita àquele usuário era gerada para que o mesmo tivesse acesso a tais serviços.

         Apesar do vasto conteúdo de informações transmitidas para o usuário, a Internet era aberta para todos, sem  nenhum dono para controlar os dados acessados.

Eis que surge a segunda fase da Internet que foi o marco da evolução da Internet, onde Tim O’Reilly no ano de 2005 diz as seguintes considerações:

*A web 2.0 é a mudança para uma Internet como plataforma, e um entendimento     das regras para obter sucesso nesta nova plataforma. Entre outras, a regra mais importante é desenvolver aplicativos que aproveitem os efeitos de rede para se tornarem melhores quanto mais são usados pelas pessoas, aproveitando a inteligência coletiva. (REILLY, 2005). [5]*

      Nesta mudança, obteve-se um enorme avanço em relação à interação do usuário, publicações de conteúdo, edições, criações de páginas online sem nenhuma dificuldade em nível de programação e também pode-se citar que o utilizador disponibilizava servidores gratuitamente.

Blogs, PodCast, Hi5, De.li.cio.us são alguns termos que podem ser citados como exemplo da grande variedades de serviços que estão hoje disponíveis graças ao avanço que a Web 2.0 propôs. [6]

       Outras características importantes, são as informações que se atualizam constantemente a fim de diminuir e obter correções para maior comodidade do utilizador, interações com pessoas do mundo inteiro, nisso, incluímos aplicações como Messenger, Googletalk, Skype e Voip, algumas ferramentas de escrita online como Google Docs e Podcast e o avanço do acesso a vídeos que cresceu muito nos últimos anos com o surgimento do Youtube, GoogleVideos e YahooVideos.

#começar falando aqui sobre tecnologias de condensação de código

#começando daqui

Além da evolução em como a internet era vista pelos navegadores, o desenvolvimento também sofreu uma evolução, a maioria dos sites antigos eram todos escritos em HTML.

HTML foi projetado para definir a estrutura de um documento da web, onde o código <p> é um elemento estrutural que se refere à "parágrafo", <LI> também é um elemento estrutural que se refere à "listar item" enquanto o HTML era expandia-se, mais elementos foram adicionados. Estes elementos, definem como o texto que será exibido, será formatado.

Em outras palavras, forma e conteúdo se tornaram inseparáveis em HTML, porém o XML foi projetado para fazer exatamente isso onde <title> não define a forma e sim o conteúdo, o mesmo com <link> e <description>, assim os dados podem ser exportados livres de restrições de formatação.

Com a forma separada, usuários não precisavam saber linhas de código para publicar informações na internet, nascem então os blogs, onde a programação já era toda definida pelo servidor em que hospedava os blogs, o usuário a partir de sua conta, somente tinha que enviar informações para ser exibida.

Mesmo sendo um avanço tecnológico grande, para se construir um site, o programador dedicava muito tempo do seu tempo para digitar inúmeras linhas de comando, as vezes para se construir pouca coisa, necessitava de um grande número de linhas.

Com o aprimoramento das tecnologias, novas linguagens de programação para web foram surgindo, e que satisfazem essa questão da necessidade do programador digitar várias linhas de comando.

Exemplo dessas tecnologia de desenvolvimento ágil é a linguagem Ruby. # fazendo pesquisas sobre ruby

#falando sobre agile

Antes do manifesto, vários fatores agravavam o não cumprimento de metas e prazos de entrega dos softwares aos clientes, tais como poucas reuniões com os clientes, onde ficavam faltando informações sobre o projeto, advogados que brigavam por causa de contratos, documentação muito extensa que acarretava o desentendimento dos desenvolvedores na hora de analisar a documentação

O manifesto agile, que significa desenvolver com agilidade teve como precursor os próprios desenvolvedores ao tentar dominar um pouco o processo produtivo nas empresas em que trabalhavam, não aceitando as regras impostas de algum gerente que muitas vezes não sabiam como funcionava o processo da fábrica de software, então eles começaram a mudar a maneira como trabalhavam, fazendo o que achavam que seriam melhor para os projetos em que trabalhavam e a partir dessas ideias, outros desenvolvedores se juntaram ao manifesto.

O manifesto agile tem como seus valores, quatro itens. Sendo eles:

1. Os indivíduos e suas interações acima de processos e ferramentas.
2. O software em funcionamento acima da documentação abrangente.
3. A colaboração dos clientes acima da negociação de contratos.
4. A capacidade de resposta a mudanças acima de se seguir um novo plano.

O que tem de errado com o formato antigo de desenvolvimento de software? waterfall?

Poucas reuniões com os clientes, advogados brigando por causa de contratos, documentos muito extensos, e desenvolvedores que se perdem no meio dessa extensão de documentação.

Como começou?

começou com programadores tentando dominar um pouco o processo produtivo, não aceitando qualquer coisa que vinha imposta de algum gerente que achava que sabia como funcionava o processo da fábrica de software, começaram a mudar começando a mudar a maneira como elas trabalhavam, fazendo coisas que elas achavam que dariam mais certo para o seu projeto, e dessas idéias as pessoas começaram a se juntar e escreveram o manifesto ágil.

Quais as vantagens?

Resultados vão aparecendo aos poucos, sendo assim cliente fica satisfeito ao ver todos os processos de desenvolvimento do projeto

Técnicas que os desenvolvedores podem usar para aplicar a metodologia ágil.

técnicas de designe, orientação a objeto, programação pareada é mais produtiva do que se duas pessoas estivessem programando separadamente, refaturar o código que é deixar a sintaxe "mais bonita" tornando a base de código muito mais legível. e fazer os testes no programa.