

Escola Técnica Estadual de São Paulo

Análise e Projetos de Sistemas

Raphael Santos da Silva

Rodrigo Martines de Moraes

Sabrina Barros Bueno

Sebastian Angel Gaona Silva

Theo Littig Tesch Barbosa

TRABALHO DE APS

Gerenciamento de Pessoal e Estimativas de Custo de Software

1ºAno – Desenvolvimento de Sistemas

São Paulo-SP

2021

Raphael Santos da Silva
Rodrigo Martines de Moraes
Sabrina Barros Bueno
Sebastian Angel Gaona Silva
Theo Littig Tesch Barbosa

TRABALHO DE APS

Gerenciamento de Pessoal e Estimativas de Custo de Software

Trabalho do Grupo F sobre Gerenciamento de
Pessoal e Estimativas de Custo de Software
apresentado ao curso de Desenvolvimento de
Sistemas.

Orientador: Professor Luiz Ricardo

1ºAno – Desenvolvimento de Sistemas

São Paulo-SP

2021

Sumário

Introdução	4
1-Gerenciamento de pessoal	5
1.1-Pessoas	6
1.2-Equipe	7
1.3-Questões de comunicação e coordenação	8
2-Estimativa de custo de software	9
2.1Principais atividades na estimativa de custo de software	9
2.2Planejamento para Recursos e Dados Requeridos	11
3-Software comercial	13
3.1-Custo de software comercial	13
4-Softwares científicos e de engenharia	13
4.1-Custo de software científico e de engenharia	14
5-Software de inteligência artificial	14
5.1-Custo de software de inteligência artificial	14
6-Software embarcado	15
6.1-Custo de software embarcado	15
7-Software em tempo real	15
7.1-Custo de software em tempo real	15
8-Software básico	16
8.1-Custo de software básico	16
9-Software de computador pessoal	16
9.1-Custo de software de um computador pessoal	16
10-Conclusão	17
12-Referências bibliográficas	18
11-Anexos	19

Introdução

Quando a questão é produzir ou reformar um software, deve ser levado em conta muitos fatores essenciais durante sua feitoria, desde custos do próprio software e correlacionados, até os gastos com os profissionais e equipe que efetuam.

Em gerenciamento de pessoal muitos temas foram levados em conta, desde o momento de recrutamento de pessoa, até a profissionalização e capacitação, e principalmente suas atividades e custos dentro da companhia, procurando sempre atingir o melhor e mais eficiente resultado.

Em estimativas de custo de software, sabemos o quão importante é estar informado do que acontece no mercado e o que está por vir, saber calcular a estimativa de custo é um processo delicado e complexo, já que software se trata de algo abstrato, mas é importante saber a estimativa de custo, para assim entrar no mercado, calculando tudo antes mesmo de começar a produzir, Afinal planejamento é tudo!

1-Gerenciamento de pessoal

Projetos bem-sucedidos ocorrem quando o gerenciamento de pessoas bem treinadas e motivadas acontece de maneira eficaz. O gerenciamento de pessoal é prioridade para criação de um software competente, portanto qualificar sua equipe para a criação de um software é essencial, sem o entendimento completo do problema a ser realizado e sem um planejamento em mãos, o projeto se perde e a consequência são inserções de erros logo ao início, dos quais serão descobertos somente mais tarde, por estes fatores e muitos outros ocorrem os insucessos de projetos.

O plano de gerenciamento de pessoal é feito conforme as necessidades do projeto, como a quantidade de pessoas, de seus níveis de habilidade, grau de dificuldade do projeto, tamanho do produto, tempo em que uma equipe vai permanecer reunida no projeto (vida da equipe), qualidade e confiabilidade exigidas do sistema a ser construído, flexibilidade da data de entrega e grau de comunicação que o projeto exige. O plano deve ser continuamente atualizado durante o processo para administrar a equipe e as ações de desenvolvimento, alguns importantes itens a serem considerados são:

- **Recrutamento.** Tendo como base a verba disponível, planejar o recrutamento das pessoas é fundamental. Estabelecer as necessidades da equipe, e/ou qual será o custo para a especialização das pessoas para o projeto, a equipe fará trabalho presencial ou a distância? E os salários? Há diversas considerações a se fazer durante um recrutamento.
- **Necessidades de treinamento.** Caso não for esperado que membros contratados tenham as competências para produzir o projeto, será possível desenvolver um plano de treinamento como parte do projeto.
- **Segurança.** Políticas devem ser implementadas para proteger os membros do projeto de perigos de segurança.
- **Prazos.** O plano de gerenciamento deve conter prazos a serem cumpridos de forma individual ou coletiva para os membros do projeto, de forma que metas sejam estabelecidas e cumpridas conforme a tabela de horários, já que também durante o planejamento é preciso pensar o número de horas em que uma pessoa, um departamento ou toda a equipe do projeto será necessária a cada semana ou mês.

- **Liberação de pessoal.** Determinar o momento ideal em que uma pessoa que já concluiu suas tarefas possa sair do projeto beneficia tanto o trabalho quanto os membros da equipe. Quando isso acontece é possível eliminar os pagamentos feitos e reduzir os custos, dando maior atenção a uma pessoa que ainda está com sua tarefa em andamento e melhores transições para futuros projetos.
- **Premiações e reconhecimento.** Premiações são importantes para motivar a equipe. Com um sistema planejado para as premiações sua utilização irá reforçar e promover os comportamentos desejados.

1.1-Pessoas

O projeto de seu software é formado por uma equipe da qual muitas vezes as pessoas que a compõe são desconhecidas uma das outras e tem de colaborar entre si para que o objetivo do projeto seja cumprido dentro do prazo e do orçamento. Os envolvidos podem ser caracterizados em um dos cinco grupos:

1. *Gerentes seniores*, definem os problemas do negócio.
2. *Gerentes de projeto*, responsáveis por planejar, motivar, organizar e controlar os programadores que executam o trabalho de software.
3. *Profissionais*, responsáveis por ter as habilidades técnicas necessárias para desenvolver a engenharia do software.
4. *Clientes*, que especificam os requisitos para o software a serem submetidos ao processo de engenharia, e outros que tenham interesses no produto.

Todo projeto possui pessoas que se enquadram nesses itens descritos. Para maximizar eficiência, a equipe deve estar organizada. Essa é a tarefa do líder da equipe.

Líderes de equipe, em todo lugar há um líder, e em projetos de software o seu papel é fundamental, mas esse trabalho não é para todos, gerenciar projetos é uma atividade que envolve muitas pessoas e por essa razão programadores competentes geralmente resultam maus líderes por não possuírem a certa habilidade com pessoas. Edgemom afirma “infelizmente e, com demasiada frequência, parece que os indivíduos só assumem o papel de gerente de projeto e se torna gerentes de projetos por acidente”.

Um livro sobre liderança técnica de Jerry Weinberg cita alguns tópicos de liderança:

Motivação. Consiste em ter a capacidade de estimular o pessoal a produzir com o melhor da sua habilidade

Organização. Capacidade de organizar processos já existentes ou moldar novos, que vão capacitar o conceito inicial a ser transformado em um produto final.

Inovar e idealizar. Consiste em estimular pessoas a criar e serem criativas, mesmo trabalhando de acordo com os padrões do projeto.

Solucionar problemas. Para se tornar eficaz, um gerente deve saber observar os problemas técnicos e organizacionais e diante disso montar uma forma de solução, podendo também motivar outros desenvolvedores a buscar uma solução. Por em prática problemas solucionados em outros projetos para novas situações, pode ajudar caso as tentativas iniciais de solucionar não tenham sido bem-sucedidas.

Identidade gerencial. Assumir a responsabilidade do projeto é característico de um bom gerente. Deve ter confiança em assumir o controle quando necessário e assegurar o pessoal técnico.

Realizações. Recompensar Iniciativas e realizações na hora certa, otimizando a produtividade de uma equipe é a cara de um gerente competente.

Influência e formação da equipe. A capacidade de olhar para as pessoas e compreendê-las e tomar uma atitude diante desses sinais é uma capacidade de um bom gerente. O gerente deve permanecer controlado em situações difíceis e de alto estresse.

1.2-Equipe

Diversas estruturas existem para gerenciar a organização da equipe durante o desenvolvimento de software. A estrutura não pode ser facilmente modificada, pois quando modificada os efeitos práticos e políticos são afetados, portanto essa estrutura deve ser cuidadosamente analisada no início do projeto o que está ao alcance do gerente de projeto.

A estrutura a ser definida dependerá dos fatores existentes na organização como a quantidade de pessoas na equipe, suas habilidades, grau de dificuldade do problema, tamanho do projeto, dentre outros.

Constantine sugere quatro “paradigmas organizacionais” para equipes de engenharia de software:

1. *Paradigma fechado* consiste em uma equipe de hierarquia tradicional, que podem trabalhar bem em produção de softwares, bastantes similares a esforços

de épocas passadas, porém se mostram menos promissoras em ser inovadoras ao trabalhar com o este paradigma.

2. *Paradigma randômico* é uma organização que depende da iniciativa individual de seus membros, e quando necessária uma inovação essas equipes se destacarão. Mas essas equipes estão mais propensas a brigarem quando for exigido um “desempenho ordenado”.
3. *Paradigma aberto* com o fundamento de ideias dos paradigmas anteriores, este paradigma inova com o trabalho feito de maneira colaborativa, com forte comunicação e tomada de decisões conjuntas. O paradigma aberto se destaca em solucionar problemas complexos, mas não desempenha tão bem quantos as outras equipes.
4. *Paradigma sincronizado* esse modelo organiza os membros para trabalhar com pouca comunicação e em tratar os problemas em partes.

Para se obter uma equipe com bom desempenho, os membros devem confiar uns nos outros, as distribuições de habilidades devem ser adequadas ao problema para atingir e manter a harmonia da equipe.

Alguns dos fatores que devem ser controlados é a “toxicidade da equipe” que consiste em um ambiente frenético, com alto grau de frustração que causa atrito entre os membros da equipe, um plano de projeto mal coordenado ou fragmentado, planejamento falho das funções dentro da equipe de software e a exposição de muitos erros durante o processo de criação.

Além desses fatores que prejudicam a convivência, uma equipe é formada por pessoas com diferentes características que dispõem esforços e colaboram entre si em um projeto, portanto planejar também a psicologia da equipe é a chave. Devido a variedade de características, o líder de equipe deve saber auxiliar essas pessoas que podem ter traços opostos de personalidade e orientar a trabalhar juntas de forma que elas não percam seu desempenho no projeto, o reconhecimento das forças humanas é primordial.

1.3-Questões de comunicação e coordenação

A falta de comunicação é um grande fator para que a criação do software tenha problemas. Juntando a complexidade, confusão e dificuldades significativas na coordenação dos membros da equipe o projeto está propenso a erros, para lidar efetivamente deve-se estabelecer mecanismos de comunicação formal e informal, por meio de reuniões,

comunicação escrita e outros canais de comunicação, os membros do grupo precisam compartilhar suas ideias entre si e solicitar ajuda conforme os problemas surgem.

2-Estimativa de custo de software

Saber o custo estimado de um software é um processo muito complexo e delicado, por se tratar de algo principalmente abstrato, não se pode simplesmente calcular o material, tempo e mão de obra, são muitos outros fatores envolvidos, também não existe um padrão ou uma tabela exata para tal, a verdade é que o custo de um software pode variar muito, dependendo da sua função, exigências e especificações.

Para se calcular o custo de um serviço temos que levar em conta:

- **Custos de Matéria-Prima (MP):** na prática, apenas o custo dos materiais mais relevantes é considerado (em termos de custo). Os materiais de custo reduzido tendem a ser considerados como materiais de consumo e são analisados simplificadaamente.
- **Custos de Mão-de-Obra Direta (MOD):** são os custos relacionados aos trabalhadores (por exemplo, salários, encargos sociais, provisões de férias e décimo-terceiro salário) envolvidos diretamente com o processo produtivo.
- **Custos Indiretos de Fabricação (CIF):** são todos os custos de fabricação que não puderam ser classificados como MP, nem como MOD (por exemplo, materiais de consumo, mão-de-obra indireta, depreciação, água, energia elétrica, telefone, entre outros) e, normalmente, são alocados por intermédio da utilização de algum critério de rateio.

2.1 Principais atividades na estimativa de custo de software

Quando se trata de software, existem algumas atividades básicas para começar a calcular o custo do seu sistema:

Estabelecimento de objetivos e análise do projeto:

É importante estabelecer os objetivos da estimativa de custo como primeiro passo, logo usar estes objetivos para direcionar o nível de detalhamento e o esforço requerido para desenvolver os passos subsequentes, mantendo uma comunicação técnica e detalhada do

projeto com o seu cliente e\ou empresa requerente do produto, lembrando que durante o processo do desenvolvimento é comum a adição e exclusão de funções, exigências e especificações do sistema.

Na especificação do software, algumas informações são essenciais para o bom desenvolvimento do mesmo, dentre elas estão:

- **Proposito:**
Por que a estimativa está sendo feita?
- **Produtos e prazos**
O que vai ser fornecido e quando?
- **Responsabilidades**
Quem é o responsável por cada produto? Onde eles vão fazer o trabalho organizacionalmente? Geograficamente?
- **Procedimentos**
Como o trabalho vai ser feito? Quais ferramentas e técnicas de estimativa de custo serão usadas?
- **Recursos requeridos**
Quantos dados, tempo, dinheiro e\ou esforços são necessários para fazer o trabalho?
- **Suposições**
Sob que condições se está prometendo entregar as estimativas, considerando os recursos anteriores (disponibilidade de pessoal fundamental, tempo de computador, dados de usuário)?
- **Outras**
Perguntas que surjam no decorrer da especificação e\ou desenvolvimento do projeto.

Com todas essas informações em mãos é preciso saber sobre qual **tipo de software** melhor se adapta ao cliente, e qual o seu **tamanho**, analisando os seguintes componentes:

- **A complexidade dos recursos do software**
Analise todas as funções e recursos que o projeto dever possuir, para assim poder atender as exigências do requerente.
- **A complexidade tecnológica**

Análise o sistema de heterogeneidade, e a compatibilidade com outros módulos e sistemas.

- **A complexidade do design**

Análise o público\usuário que o software vai atender, assim junto ao requerente criar um sistema de design compatível e acessível para o mesmo.

Logo após analisar e definir os mesmos, defina o tipo e tamanho do software a ser criado.

2.2Planejamento para Recursos e Dados Requeridos:

Nesta etapa, é necessário calcular o custo do **MP**, **MOD**, e **CIF**, do seu sistema, sempre levando como base o **Custo perceptível** pelo requerente, e o **Custo de investimento**, assim criando um intermédio entre ambos.

Custo de investimento

- **Implementação:**

Se baseia em calcular as horas investidas no projeto e multiplicar pelo valor da hora técnica, da equipe e/ou sua.

- **Manutenção:**

Se baseia em calcular o custo de manutenção e atualizações do sistema, tomando como base a adição e exclusão de novos recursos, funções e manutenções obrigatórias, é importante saber que esse custo deve ser fixo e programado, não um custo acumulado como a implementação.

- **Treinamento:**

Se baseia na custo de treinamento do(s) usuário(s) final(is), nele é necessário a prática para ser calculado, alias há chance de se superestimar ou subestimar o(s) usuário(s), é importante lembrar que tal custo pode ser reduzido ou ampliado a longo prazo.

- **Suporte:**

Se baseia em calcular o custo da equipe de suporte do cliente para com o sistema, nele é possível tanto calcular horas avulsas, mensais ou ilimitadas, esse é muito variável, pois depende do projeto e quantidade de requerimentos para com o cliente.

- **Outras despesas:**

Neste deve se calcular despesas como: funcionários, contadores, RH, água, luz, telefone, internet e outras despesas indiretas do serviço.

Custo perceptível

- **Utilidade:**

Se trata da visão do usuário sobre o projeto, e o quão “atrativo” é para ele, o sistema é útil o bastante? Qual o tempo de resposta? Vale a pena? e outras questões que variam dependendo do projeto.

- **Facilidade e implantação de uso:**

Os usuários odeiam mudanças que custem dinheiro, por isso é de extrema importância um sistema que seja de fácil manutenção e implantação para o uso, caso o projeto seja muito complexo e de difícil entendimento a chances de o mesmo perder a sua credibilidade.

- **Integrações:**

Fazer um software que seja flexível a mudanças e a adição de outros módulos e atualizações faz toda a diferença.

- **Modelo comercial:**

Os usuários gostam de se sentir seguros em negociações, evite usar palavras difíceis e/ou taxas com nomes complexos.

- **Mercado:**

Um dos principais pontos é analisar e identificar se o seu custo está semelhante e dentro da média do mercado, estar sempre “por dentro” pode ser muito útil para não perder cliente para a concorrência.

- **Credibilidade da marca:**

Quando se tem um bom “nome” ou é indicado por boas fontes se passa uma maior segurança para o cliente, além de poder acelerar suas negociações.

Logo após esse processo crie um plano de marketing atrativo, que seja de fácil visualização e entendimento e leve ao seu requerente de software, lembrando que durante o desenvolvimento do mesmo pode ocorrer mudanças e contratempos, assim podendo alterar o custo final.

3-Software comercial

O Software comercial é software desenvolvido para comercializar ou com interesses empresariais. O software comercial e proprietário não são o mesmo. A maioria dos softwares comerciais são proprietários, mas existem softwares livres que são comerciais.

Os conceitos de "Livre" e "proprietário" representam apenas atributos de licença de software. Eles São modalidades de relações jurídicas que se pode estabelecer entre um privado e o fornecedor.

No caso do "Software Proprietário" significa que a distribuição é realizada por comercialização e se dará no regime jurídico clássico comercial no qual a relação é baseada em restrições e permissões, tendo a propriedade a autoria, e utilização do mesmo. isto é, neste tipo de software é algo feito para empresa e ela pode decidir se tornará isso público, ou não.

Já em relação ao "Software Livre", ele significa que a distribuição é realizada em um regime jurídico de colaboração no qual a relação se baseia, ao contrário, em liberdades, tutelando-se tão somente a autoria e a permanência desse mesmo regime nas distribuições subsequentes do software. Este tipo de software é livre, mas pertence uma área comercial.

3.1-Custo de software comercial

Dependendo do tamanho da empresa cada software tem um valor diferente e quando falamos de um software para pequenas e outro para grandes, o valor difere muito. Alguns softwares simples **possuem preços muito mais acessíveis e dependendo do projeto não possuem custo de aquisição**. Já um software complexo e cheio de funcionalidades tem um custo de aquisição altíssimo e uma mensalidade também muito alta.

Softwares para médias empresas tem um custo de aquisição de 10 a 100 mil reais. Softwares para grandes empreendimentos tem, muitas vezes, um custo de implantação que chega a 1 milhão de reais...

4-Softwares científicos e de engenharia

Os softwares científicos e de engenharia são aqueles que auxiliam as aplicações científicas. têm sido caracterizados por algoritmos de processamentos de números. Esse tipo

de software tende a atender um público alvo concentrado, tendo em vista que cada produto feito atende à necessidade de um determinado nicho de mercado.

4.1-Custo de software científico e de engenharia

Nos softwares científicos e de engenharia os preços variam muito, assim como nos outros tipos de softwares, mas de maneira bruta 60% dos custos são custos de desenvolvimento, 40% são custos de teste. Para um software personalizado, o custo de evolução frequentemente ultrapassa os custos de desenvolvimento.

Alguns custos podem variar dependendo do tipo de sistema que está sendo desenvolvido e dos requisitos dos sistemas como a performance e a confiabilidade do sistema. Esta distribuição de custos depende do modelo de desenvolvimento, e de muitos outros fatores na feitoria do projeto.

5-Software de inteligência artificial

Os softwares de inteligência artificial estão se tornando cada dia mais famosos e concorridos pelas companhias ao redor do mundo, e por esse motivo o seu custo é altíssimo.

Muitas empresas famosas como o Google, a IBM, a Apple e até algumas Startups estão na corrida para ganhar espaço nesta grande indústria da inteligência artificial que revela o futuro.

5.1-Custo de software de inteligência artificial

Muitas companhias em todo o mundo estão investindo bilhões de dólares em busca das melhores estruturas, ferramentas e profissionais para o desenvolvimento destes softwares, e tudo indica que essa tendência irá perpetuar pelos próximos anos. Segundo a Gartner, o valor global de negócios de inteligência artificial será de \$3,9 trilhões até 2022.

Nestas tecnologias são usados muitos chips, hardwares e controladores, tudo de ótima qualidade e especificação, por isso os valores estimados para esses softwares são exorbitantemente caros, mas não é possível estipular custos corretos.

6-Software embarcado

Reside na memória de leitura e, é usado para controlar produtos e sistemas para os mercados industriais e de consumo. Executam funções limitadas e particulares. este tipo de software também é conhecido como **Firmware**.

6.1Custo de software embarcado

Um estudo interno feito pela Bell Labs revelou que para se obter de 1 a 2 defeitos a cada 1.000 linhas de código de seus projetos, precisa-se produzir de 150 a 300 linhas de código por mês. Considerando que os salários e gastos vindos do desenvolvimento, gera-se um custo de aproximadamente \$25~\$50 por linha de código.

Um exemplo disso é o **software de controle de uma missão com ônibus espacial da NASA**, feito pelos engenheiros da IBM. Pode ser considerado um software com “zero erros”, já que apresenta um custo aproximado de \$1.000 por linha de código.

7-Software em tempo real

Um software em tempo real é um software que implementa um recurso que funciona em um intervalo de tempo que o usuário percebe como imediato. O termo é mais comumente usado para aplicativos que realizam conexões em tempo real com o servidor ou entre usuários para fornecer uma experiência de conexão instantânea.

7.1-Custo de software em tempo real

As estimativas de um software em tempo real, é algo difícil de calcular assim como os outros tipos de softwares.

8-Software básico

São os programas necessários para que o computador seja funcional, onde esses dão suporte e apoio a diversos programas. É responsável de criar o ambiente onde tem a interação entre o usuário e a máquina.

Um exemplo de Software Básico é o BIOS (Basic Input/Output System), que em português significa Sistema Básico de Entrada e Saída, que é responsável por reconhecer hardwares que são instalados, além de iniciar o sistema operacional. Outros exemplos são os compiladores, as linguagens de programação, utilitários e editores para o gerenciamento de arquivos.

8.1-Custo de um software básico

Para dar um valor de custo ao Software Básico é necessário levar em conta a complexidade do sistema, o fornecedor do produto, os testes e a implementação, assim podendo haver uma variação de R\$ 20,00 até mais de R\$ 5000,00.

9-Software de computador pessoal

Primeiramente um computador pessoal ou PC (do inglês *Personal Computer*) é um computador de pequeno porte, que se destina ao uso pessoal ou por um pequeno grupo de indivíduos. A expressão é utilizada para denominar *desktops*, computadores portáteis, *tablets*, entre outros, executando vários Sistemas Operacionais em várias arquiteturas. Os Sistemas Operacionais predominantes são Microsoft Windows, Mac OS X e os sistemas baseados em Linux, e as principais arquiteturas são as baseadas nos processadores x86, x64 e PowerPC.

9.1-Custo de software de um computador pessoal

Os PCs representam atualmente um dos maiores mercados mundiais. As vendas de PCs crescem constantemente e batem recordes extraordinários. De acordo com a empresa de pesquisa IDC, 66,9 milhões de PCs foram comercializados em 2007, contra 57,9 milhões um ano antes. As vendas de computadores portáteis contaram muito para o aumento de 15,5% nas vendas globais de PCs de julho até setembro de 2007, em relação ao mesmo período do ano passado, especialmente na Europa, segundo a empresa de pesquisa IDC. A pesquisa também aponta um crescimento maior nas vendas fora dos Estados Unidos.

Hoje com uma evolução bastante grande na tecnologia e nos computadores, segundo pesquisa, no Brasil, podemos achar PCs a partir de R\$800 os valores variam até R\$3,000.

OBS: Lembrando que estas informações de custos são estimativas de mercado.

10-Conclusão

O saber do gerenciamento de pessoal é de extrema importância para assim coordenar de maneira fluida e eficiente um grupo de pessoas e/ou uma equipe, assim aumentando a porcentagem de sucesso e projetos bem sucedidos, na mesma, se trata de alguns fatores essenciais como o recrutamento e a capacitação dos integrantes, e os métodos de liderança eficazes, quais motivem e melhorem o desempenho da equipe e do grupo em si, além da comunicação entre os membro(s) e líder(s), que é de extrema importância para o próprio bem estar dos integrantes e a efetuação correta do projeto e alcance do objetivo.

Quando se trata de estimar o custo de um software são diversos fatores que tem de serem levados em contas, um dos principais são o tipo e tamanho de software, modelo de processamento, tempo técnico, valor do mercado e outros... não existe uma tabela ou um quadro que de um valor exato para o software ou projeto, mas é possível por meio de observações de outros trabalhos e cálculos de custo de investimento e custo perceptível ter uma estimativa válida para assim trazer um plano de projeto para o seu cliente\empresa requerente do serviço ou montar seu portfólio de serviço.

11- Referências bibliográficas

Link: <https://centraldefavoritos.com.br/2018/05/10/software-software-basico-nocoes-de-sistemas-operacionais-utilitarios-antivirus/>

Link: <https://techenter.com.br/o-que-e-software/>

Link: <https://www.techtudo.com.br/noticias/noticia/2014/05/o-que-e-bios.html>

Link: <https://www.mxm.com.br/blog/quanto-custa-um-erp-conheca-7-principais-custos/>

Link: <https://fia.com.br/blog/desenvolvimento-de-sofware/>

Link: <https://codificar.com.br/os-custos-envolvidos-no-desenvolvimento-de-um-software>

Link: <https://www.opus-software.com.br/mercado-inteligencia-artificial/>

Link: <https://www.embarcados.com.br/editorial-custo-do-firmware/>

Link: <https://www.techtudo.com.br/dicas-e-tutoriais/noticia/2014/04/firmware-o-que-e-e->

Link: <https://www.devmedia.com.br/modelagem-da-gestao-de-pessoas/26916>

Link: <http://www.linhadecodigo.com.br/artigo/1722/gestao-de-pessoas-em-projetos-de-software.aspx>

Link: <http://wpm.wikidot.com/artefato:plano-de-gerenciamento-de-pessoal>

Link: <https://blog.tecnospeed.com.br/precificacao-de-software/>

Link: <https://smartqualityglobal.com/blog/quanto-custa-desenvolver-um-sistema-sob-medida/>

12- Anexos

ARTIGO:MODELOS PARA ESTIMAR CUSTOS DE SOFTWARE: ESTUDO COMPARATIVO COM SOFTWARES DE PEQUENO PORTE-MARISTELA CORRÊA MELLER

LIVRO:ENGENHARIA DE SOFTWARE\UMA ABORDAGEM PROFISSIONAL\8ª EDIÇÃO\AMGH EDITORA LTDA- ROGER S. PRESSMAN-BRUCE R. MAXIM

ARTIGO: TÉCNICAS DE ESTIMATIA DE CUSTO DE SOFTWARE:CONTRIBUIÇÕES PARA O GERENCIAMENTO DE CUSTO EM PROJETOS- PEDRO HENRIQUE-FERNANDO RODRIGUES

LIVRO: ENGENHARIA DE SOFTWARE-IAN SOMMERVILLE-9ª EDIÇÃO

ARTIGO: ESTIMATIVA DE CUSTO DE SOFTWARE: ROTEIRO E DICAS PARA ESTIMATIVAS DE PROJETO- ANTONIO MENDES DA SILVA FILHO