

## Capítulo II – NOBLEWARE: Estimativas de projeto de software

André Nogueira da Costa

Fernando Hideki Watanabe

Pedro Andrade Galdana

Anderson Paulo Ávila Santos<sup>4</sup>

Lucas Busatta Galhardi<sup>5</sup>

Fábio Takeshi Matsunaga<sup>6</sup>

### RESUMO

Este artigo tem como objetivo expor os resultados obtidos durante o desenvolvimento do projeto proposto na Jornada de Aprendizagem, que consiste em construir um sistema como forma de solução de um problema real do mercado de softwares. Neste contexto, tratou-se dos equívocos e dificuldades encontradas nas estimativas de orçamentos de softwares realizadas por equipes de consultoria, obtendo como solução um sistema que por meio de uma Inteligência Artificial, tendo como base o conhecimento prévio dos participantes, realizasse uma estimativa de horas e custos para a construção de um software, voltado para os usuários que não possuem fácil acesso à uma consultoria presencial ou para empresas interessadas na ferramenta.

**Palavras-chave:** Software. Consultoria. Estimativa. Inteligência Artificial.

### NOBLEWARE: Software Project Estimate

### ABSTRACT

This article aims to expose the results obtained during the development of the project proposed in the Learning Journey, which consists of building a system as a way to solve a real problem in the software market. In this context, the mistakes and difficulties found in the estimates of software budgets carried out by consulting teams were addressed, obtaining as a solution a system that, through an Artificial Intelligence, based on the prior knowledge of the participants, carried out an estimate of hours and costs for the construction of a software, aimed at users who do not have easy access to a face-to-face consultancy or for companies interested in the tool.

**Keywords:** Software. Consultancy. Estimate. Artificial Intelligence.

---

<sup>4</sup> Mestre em Ciência da Computação. E-mail: anderson.avila@sistemafiep.org.br

<sup>5</sup> Mestre em Ciência da Computação. E-mail: lucas.galhardi@sistemafiep.org.br

<sup>6</sup> Mestre em Ciência da Computação. E-mail: fabio.matsunaga@sistemafiep.org.br

## 1. INTRODUÇÃO

O Processo de desenvolvimento de Software é um processo de alta complexidade, pois são diversas fases realizadas durante todo o processo, e uma das principais fases é a fase de Levantamento de Requisitos. Durante esta fase, são realizadas a coleta dos requisitos necessários (Requisitos Funcionais, Requisitos Não Funcionais), e é justamente nesta fase que surgem muitos problemas relacionados ao processo de desenvolvimento do Software, sendo um dos problemas mais comuns, a falta dos requisitos, acarretando assim mais problemas de custos e entrega do projeto.

A equipe analisou estes problemas, e assim, desenvolveram uma ideia de uma possível solução, sendo este um sistema Web de consultoria para estimativas de custo e horas para determinado projeto.

Nossas principais justificativas para realização deste projeto são solucionar os problemas envolvendo o processo de desenvolvimento de Software e trazer uma maior praticidade, acessibilidade e facilitador para aqueles que não possuem conhecimento a respeito do processo (seja tanto no quesito informativo quanto no quesito custos e valores).

O projeto Nobleware contribui para um dos ODS, mais especificamente o objetivo **9 – Indústria, Inovação e Infraestrutura**, que se preocupa com a construção de infraestruturas confiáveis para promover a industrialização e estimular a inovação. De acordo com os tópicos presentes no objetivo, o nosso projeto se enquadra no propósito de elevar o acesso das pessoas às tecnologias.

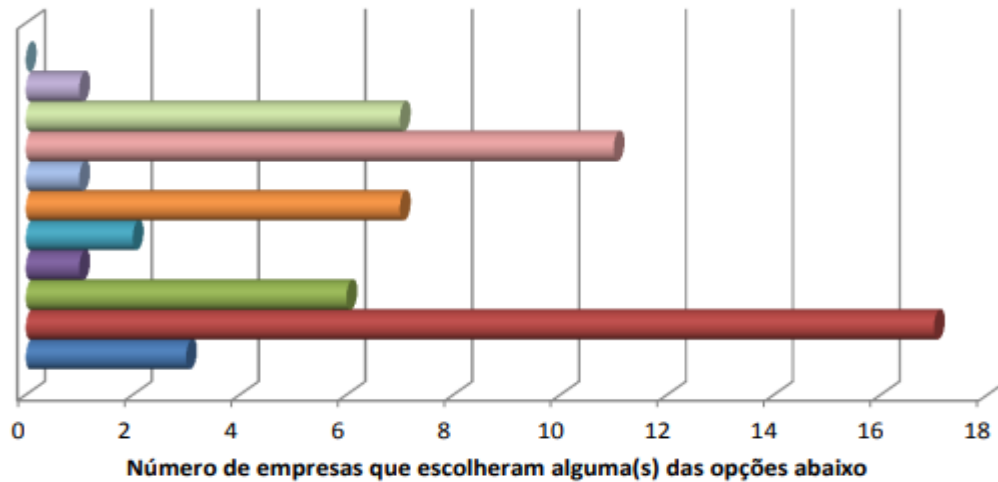
## 2. FUNDAMENTAÇÃO TEÓRICA

Durante nossas pesquisas realizadas, o site Whow! (2020) afirma que muitas empresas, durante o período da quarentena, tiveram a necessidade repentina de utilização de serviços e plataformas como solução para manter seus negócios ativos.

Pelas pesquisas realizadas por FIGUEIRA (2012), este aponta que muito dos desenvolvedores encontram dificuldades durante o processo de levantamento de requisitos como é indicado no gráfico abaixo:

Figura 1 – Dificuldades no levantamento de requisitos.

### Principais dificuldades no levantamento de requisitos estão relacionadas:



- Outro
- Não enfrento dificuldades.
- Ao analista ter que exercer vários papéis durante a atividade.
- Às mudanças nos requisitos.
- À dificuldade em capturar informações no domínio.
- À dificuldade em validar o que o cliente solicitou.
- À análise das informações coletadas do usuário.
- Às técnicas e ferramentas existentes.
- À existência de muitos stakeholders envolvidos com interesses distintos.
- À dificuldade do cliente expressar o que realmente deseja.
- À dificuldade em se relacionar com o cliente.

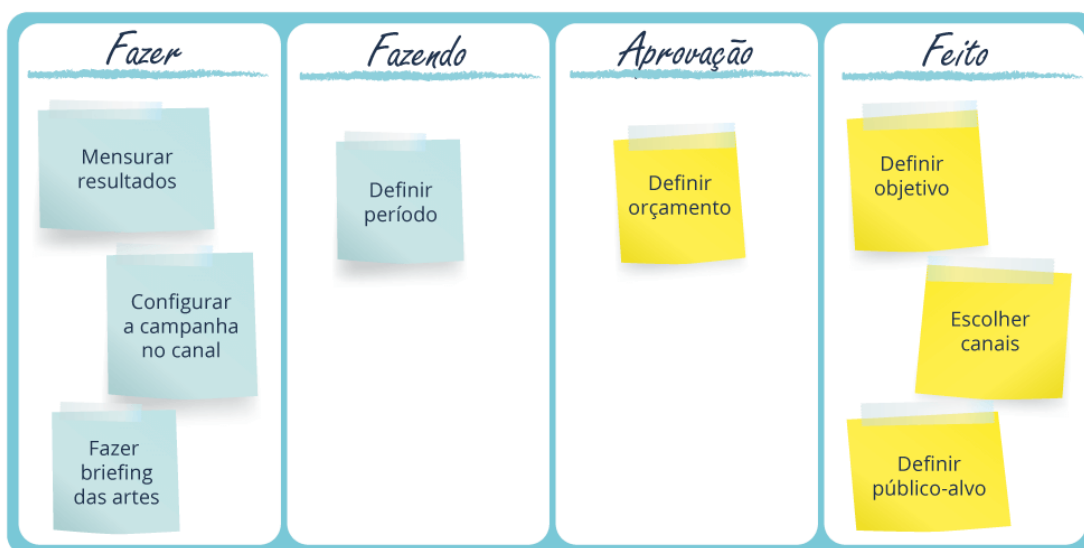
Fonte: Anderson Marques da Silva Figueira (2012)

### 3. METODOLOGIA

A metodologia aplicada para o desenvolvimento do trabalho foi o SCRUM, que consiste em definir backlogs — um conjunto de tarefas a serem realizadas. Além de dividir as etapas de desenvolvimento em ciclos, também chamados de Sprints, normalmente cada Sprint tem um tempo de duração, por exemplo, uma semana ou duas semanas. Para definir uma Sprint, existe uma fase de planejamento para selecionar as atividades que serão realizadas durante o período determinado, esta fase é chamada de Sprint Planning. Após a finalização da Sprint, é realizada uma reunião, para verificar se os objetivos da Sprint — Product Backlog — foram atingidos e um feedback da equipe sobre o andamento do projeto, isto é chamado de Sprint Retrospective.

Para controlar os fluxos de produção é utilizado o Kanban, que é uma metodologia de gerenciamento ágil, que conta com a ajuda de um painel que organiza o progresso do projeto, dividindo-o em etapas: “para fazer”, “em andamento”, “aprovação” e “concluído”. Assim é possível ter uma visão geral de como está o andamento do projeto. Atualmente, existem algumas ferramentas que auxiliam na criação do Kanban, com uma interface digital por meio de um software. A ferramenta escolhida para esta função foi o Trello. Nela é possível criar o planejamento das ações do projeto, determinar prazos e atualizar o andamento de cada tarefa.

Figura 2 – Exemplo tabela Kanban



Fonte: Roberto Gil Espinha, 2019.

Após a definição da metodologia de desenvolvimento, foi iniciado o processo de modelagem do sistema, por meio dos diagramas UML, que servem para determinar os fluxos e as funções que o sistema irá conter. Nesse projeto foram utilizados os diagramas de caso de uso e diagrama de classe, que são a base para começar o desenvolvimento do software. Sendo assim, foi iniciada a construção do sistema por meio de códigos, utilizando as tecnologias escolhidas pela equipe, que no caso foi o ReactJS (JavaScripts) e o Django (Python). Para salvar as mudanças no código, foi utilizado o GitHub, cuja sua função é controlar o que cada integrante da equipe alterou em determinada parte do software, possibilitando o versionamento de código.

Para o desenvolvimento da camada de back-end (Django), foi utilizado a arquitetura API Rest, que é um conjunto de restrições, onde as requisições feitas pela camada client-side, tenham um padrão de comunicação com a camada server-side. E para o desenvolvimento da camada front-end (ReactJS), foi utilizado o padrão de reutilização de componentes, para organizar e facilitar a construção das interfaces.

#### **4. DESENVOLVIMENTO E APRESENTAÇÃO DO SISTEMA**

Com a modelagem do sistema realizada, a equipe decidiu definir para o front-end, o modelo arquitetural com base na reutilização de componentes, como opção para o desenvolvimento ágil e organizado da interface do sistema. E para o back-end, a equipe decidiu utilizar o modelo de API Rest, visando estabelecer um padrão de comunicação com a camada de client-side, possibilitando a identificação do resultado de cada requisição através dos códigos de status de resposta HTTP.

Após a aplicação dos modelos arquiteturais, o desenvolvimento da solução como produto foi iniciado com o objetivo de construir as telas “Home”, “Questionário” e “Orçamento”.

Figura 3 – Tela Home.



Fonte: Elaborado pelo autor (2021).

A figura 3 representa a tela inicial da aplicação, onde é exibido um dashboard contando informações e estatísticas com base nos orçamentos gerados no histórico do sistema.

Figura 4 – Tela de Questionário.

The questionnaire screen includes a progress bar with four stages: "Conhecendo o sistema", "Requisitos", "Detalhes", and "Orçamento". The current stage is "Requisitos".

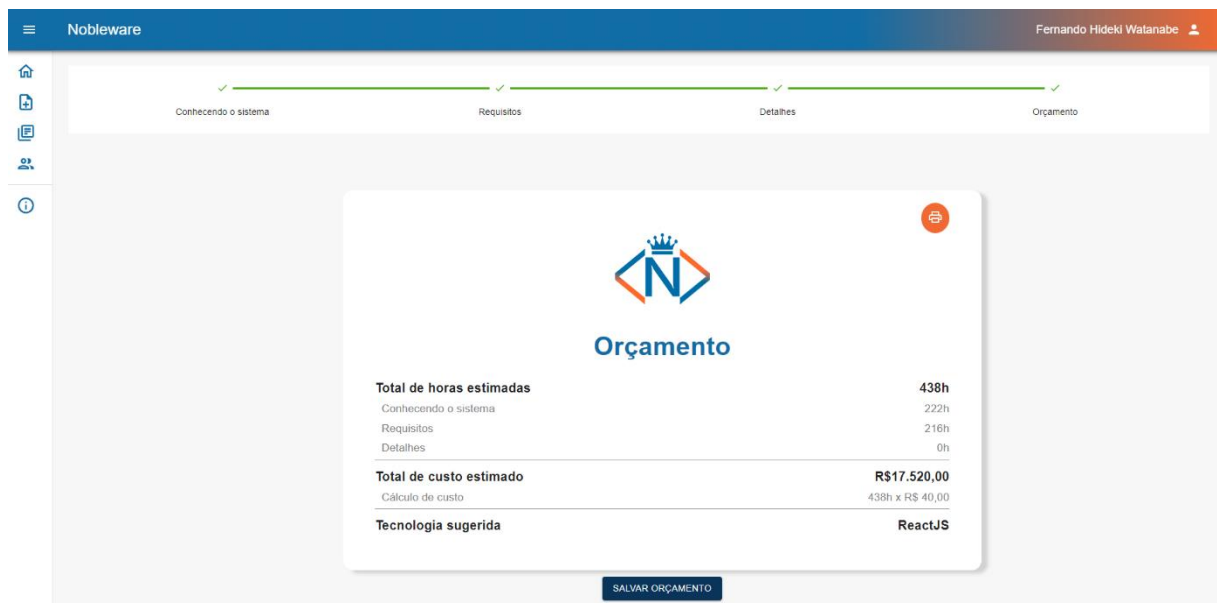
Questions and options:

- Qual a plataforma de desenvolvimento?
  - Web
  - Mobile
  - Desktop
- Deverá rodar em quais navegadores?
  - Opera
  - Mozilla
  - Chrome
  - Internet Edge
- Deverá rodar em quais sistemas operacionais?
  - Android
  - IOS (Apple)
- Deverá ter acesso a internet para utilizar o sistema?
  - Sim
  - Não

Fonte: Elaborado pelo autor (2021).

A figura 4 representa a tela de questionário, uma das principais funcionalidades da aplicação, contendo as questões de múltipla escolha com o objetivo de colher os interesses do usuário em relação ao software que ele deseja. As questões foram divididas em etapas, que simulam as fases de uma entrevista de consultoria com o cliente. Ao final do questionário, as respostas são enviadas para a camada server-side, onde são processadas pela Inteligência Artificial (IA), que por meio de um conjunto de dados de estimativas, executa a predição de tempo de desenvolvimento através de um modelo treinado. O resultado obtido na atuação da IA, é exibido para o usuário na tela de orçamento, conforme figura 5.

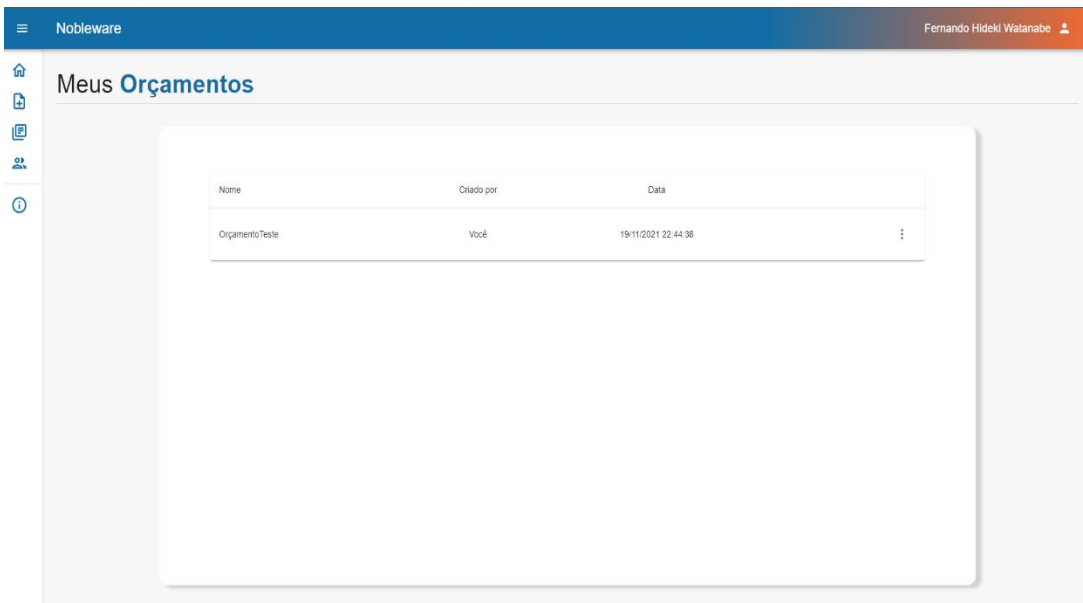
Figura 5 – Tela de Orçamento.



Fonte: Elaborado pelo autor (2021).

A figura 6 representa a tela de orçamentos realizados, contendo todos os orçamentos salvos pelo usuário. Cada orçamento é listado com botões de ação, dando as opções de visualizar e compartilhar o arquivo do orçamento.

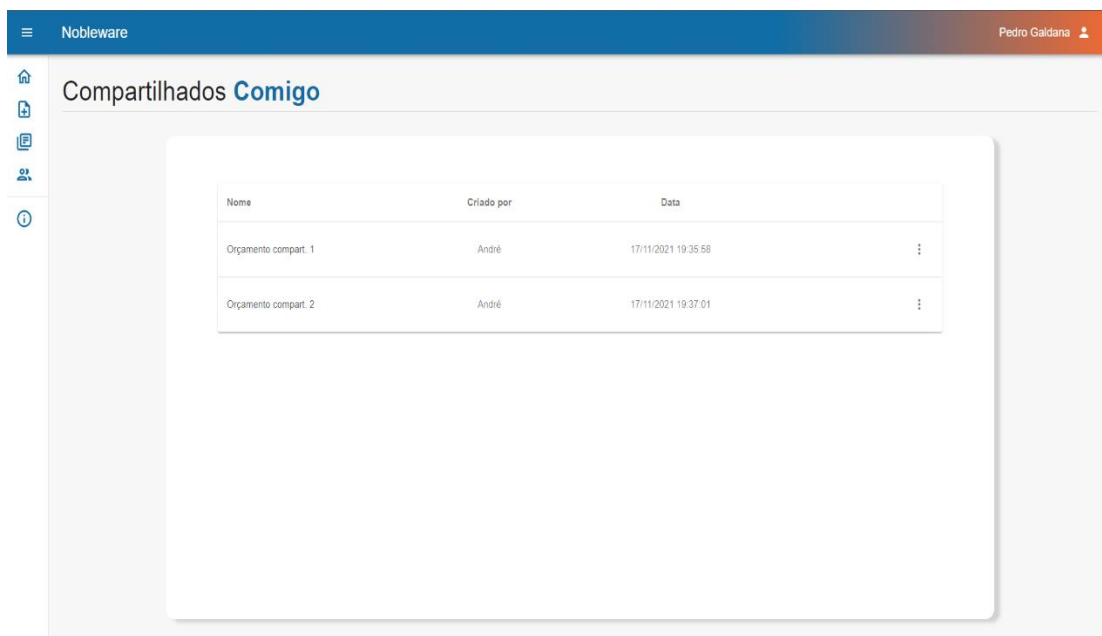
Figura 6 – Tela de Meus Orçamentos.



Fonte: Elaborado pelo autor (2021).

Na figura 7 é apresentada a Tela de Orçamentos Compartilhados Comigo, onde todos os orçamentos que foram compartilhados com aquele determinado usuário serão listados naquela tela, permitindo a visualização mais detalhada do orçamento compartilhado (Figura 5).

Figura 7 – Tela de Orçamentos Compartilhados Comigo



Fonte: Elaborado pelo autor (2021).

## 5. COMPARANDO LINGUAGENS DE PROGRAMAÇÃO

Em nosso projeto, utilizamos de algumas tecnologias disponíveis para o desenvolvimento do Sistema, sendo essas tecnologias utilizadas o Framework ReactJS, feito em JavaScript, para o desenvolvimento do Front-End, e o Framework Django, feito em Python, para o desenvolvimento do Back-End do projeto.

Analizamos alguns comparativos de algumas outras linguagens que são equiparadas as quais utilizamos para o desenvolvimento do nosso projeto, sendo essas linguagens, mais especificamente o C# e o PHP.

## 5.1 VANTAGENS

- **Python:** O Python, por se tratar de uma linguagem Open Source, ela é muito utilizada no mercado de Tecnologia num geral, e esta é de certa forma uma linguagem muito prática e com uma certa facilidade de se aprender pelo fato dela ser dinamicamente tipada, além de ser uma linguagem bem versátil com possibilidade de aplicação para diversas soluções. Essa linguagem é bem consolidada no mercado atual e possui suporte para diversas plataformas diferentes, então esta possui muitas soluções disponíveis tanto por parte da documentação original da linguagem quanto por parte da comunidade de desenvolvedores, o que a torna uma boa opção de uso para desenvolvimento geral.
- **C#:** A linguagem C# é uma linguagem que é fortemente tipada que segue o paradigma orientado à objetos, mas isto não significa que ela é difícil de se aprender ou algo do gênero, muito pelo contrário, caso a pessoa possua experiência com outras linguagens como Java, C++, ela não encontrará dificuldades. Assim como a linguagem Python, o C# também é uma linguagem muito consolidada no mercado de Tecnologia, sendo assim, também possui muitos recursos e soluções disponíveis. Essa linguagem, diferente da Python, possui suporte para diversas plataformas, podendo ser Web, Desktop e até mesmo mobile, o que também possibilita uma possibilidade grande de aplicabilidade em diversas soluções diferentes. No quesito desempenho, C# é uma linguagem compilada,

sendo assim, garante um pouco mais de velocidade e segurança na sua execução.

- **PHP:** Assim como as linguagens Python e C#, o PHP é uma linguagem open source, permitindo sua utilização para qualquer situação, pessoas e/ou empresas. Essa linguagem também possui um certo grau de facilidade, visto que pessoas que já tiveram contato com C, Java e afins também não encontraram grandes dificuldades em aprender essa linguagem, já que o PHP utiliza de nomenclaturas presentes nessas linguagens. A linguagem também é muito versátil, visto que ela é capaz de rodar em diversas plataformas como Linux, Windows, Mac OS etc. Além disso, essa linguagem

## 5.2 DESVANTAGENS

- **Python:** Uma das principais desvantagens que pode ser destacada nesta linguagem é em relação à sua performance. Pelo fato de a linguagem ser interpretada, isso significa que sua execução ocorre com o auxílio de um interpretador ao invés de um compilador, sendo executado linha por linha. Uma das desvantagens que também vale ser citada é a questão do design da linguagem, isso faz com que seja necessário realizar muitos mais testes no código pois os erros aparecem somente no tempo de execução do código. O Python também conta com uma limitação em relação a plataformas mobiles, onde é raramente usada e possui poucos recursos e soluções disponíveis.
- **C#:** Apesar do fato da linguagem C# ser compilada trazer um certo ganho de desempenho, existem algumas complicações que podem ser ocasionadas durante o desenvolvimento. Por exemplo, se caso seja necessário realizar alguma modificação no código, será necessário recompilar a aplicação inteira, o que pode acabar ocasionando um ganho no tempo de execução. A estrutura da linguagem C# é bem estrita se comparada com linguagem que utilizam de tipagem dinâmica, então muita das vezes alguma solução

feita em C# pode acabar se tornando muito maior e mais complexa que soluções realizadas em Python por exemplo.

- **PHP:** A linguagem PHP é um tipo de linguagem interpretada, que é executada pelo lado do servidor, portanto, a chance de algum invasor conseguir o acesso ao código fonte é relativamente alta. A linguagem PHP não possui uma determinada padronização, sendo assim, muitos recursos de versões anteriores das linguagens podem acabar se tornando obsoletos e até mesmo podem parar de funcionar conforme versões novas forem sendo implementadas, além disso, em relação à documentação da linguagem, a documentação dela é bastante incompleta e nem sempre todos os recursos estarão disponíveis na documentação oficial da linguagem, podendo encontrar facilmente recursos implementados que ainda não foram inseridos na documentação oficial.

## 6. APRESENTAÇÃO DOS RESULTADOS E CONSIDERAÇÕES FINAIS

A equipe conseguiu concluir com êxito a realização do desenvolvimento do Sistema, conseguindo realizar a implementação de todas as funcionalidades que estavam previstas além de conseguir realizar a implementação do algoritmo de Inteligência Artificial.

Além da realização do desenvolvimento do Sistema, a equipe realizou testes para garantir que as funcionalidades do sistema estão funcionando de acordo com as especificações que foram documentadas. Com o auxílio da ferramenta TestLink, a equipe pode ser capaz de realizar o controle dos testes que foram realizados sobre a aplicação, além de manter o controle de versões liberadas.

Foram realizados diversos testes sobre a aplicação, sendo os principais os testes funcionais, para garantir que o sistema esteja funcional, testes de usabilidade, para garantir que o sistema possuirá uma usabilidade relativamente simples e sem muitas complicações que possam atrapalhar a experiência de usuário, e alguns testes exploratórios gerais.

A plataforma também permite gerar o relatório da execução dos testes, que permite visualizar os cenários realizados, com o passo a passo, resultados esperados e resultados obtidos.

Concluiu-se que ao final do projeto que, por ser algo incomum no mercado de sistemas Web, a aplicação desenvolvida pela equipe possui um certo potencial de mercado, visto que até o presente momento não existe nenhum produto semelhante ao implementado pela equipe. Vale ressaltar também que, existem inúmeras pesquisas e artigos, como por exemplo, em “Machine learning approaches to estimating software development effort” (K. Srinivasan and D. Fisher, 1995), discutindo sobre esse assunto e até mesmo, Inteligências Artificiais treinadas para realizar o orçamento de um software, porém não encontramos nenhum sistema no âmbito comercial, que realizasse o que o nosso sistema faz.

Para planos futuros, a equipe planeja realizar a implementação de features novas, que irão complementar e melhorar o Sistema como um todo, além de continuar treinando a Inteligência Artificial implementada para que esta esteja funcionando da melhor maneira possível e esteja sempre atualizada.

## REFERÊNCIAS

FIGUEIRA, S. M. A. ANÁLISE DAS TÉCNICAS DE LEVANTAMENTO DE REQUISITOS PARA DESENVOLVIMENTO DE SOFTWARE NAS EMPRESAS DE VITÓRIA DA CONQUISTA – BA. 74f. Monografia (Graduação em Engenharia de Software). Universidade Estadual do Sudoeste da Bahia, Bahia, 2012. Disponível em: < <http://www2.uesb.br/computacao/wp-content/uploads/2014/09/ANÁLISE-DAS-TÉCNICAS-DE-LEVANTAMENTO-DE-REQUISITOS-PARA-DESENVOLVIMENTO-DE-SOFTWARE-NAS-EMPRESAS-DE-VITÓRIA-DA-CONQUISTA---BA.pdf>> Acessado em: 31 mai. 2021.

FONSECA, A. Aumento na demanda por desenvolvedores na quarentena. Disponível em: <<https://www.whow.com.br/tecnologia/aumento-demanda-desenvolvedores-quarentena/>>. Acesso em: 27 mai. 2021.

NETO, N. A. J. C# vs. Python. Disponível em: <[https://linkedin.com/pulse/c-vs-python-jos%C3%A9-antonio-das-neves-neto/?trk=public\\_profile\\_article\\_view](https://linkedin.com/pulse/c-vs-python-jos%C3%A9-antonio-das-neves-neto/?trk=public_profile_article_view)>. Acesso em: 11 out. 2021.

Stack Overflow Developer Survey 2021. Disponível em:  
<<https://insights.stackoverflow.com/survey/2021>>.  
Acesso em: 30 out. 2021.

FERREIRA, K. O que é PHP e por que você precisa conhecer essa linguagem de programação web. Disponível em: <<https://rockcontent.com/br/blog/o-que-e-php/>>.  
Acesso em: 30 out. 2021.

Um tour pelo C#. Disponível em: <<https://docs.microsoft.com/pt-br/dotnet/csharp/tour-of-csharp/>>.  
Acesso em: 30 out. 2021.

PHP: Documentation. Disponível em: <<https://www.php.net/docs.php>>.  
Acesso em: 30 out. 2021.

ROVEDA, U. O que é Python, para que serve e por que aprender. Disponível em:  
<<https://kenzie.com.br/blog/o-que-e-python/>>.  
Acesso em: 30. out. 2021.

O tutorial de Python. Disponível em: <<https://docs.python.org/pt-br/3.10/tutorial/index.html>>.  
Acesso em: 30. out. 2021.

K. Srinivasan and D. Fisher, "Machine learning approaches to estimating software development effort," in IEEE Transactions on Software Engineering, vol. 21, no. 2, pp. 126-137, Feb. 1995, doi: 10.1109/32.345828.