

**UNIVERSIDADE FEDERAL DE SANTA CATARINA  
CAMPUS ARARANGUÁ  
CENTRO DE CIÊNCIAS, TECNOLOGIAS E SAÚDE  
ENGENHARIA DE COMPUTAÇÃO**

ALE CHAITO

**UM PROTÓTIPO DE UM SISTEMA DE SOFTWARE PARA GERAÇÃO DE  
FORMULÁRIOS PARA COLETA DE DADOS**

Araranguá

2021

Ale Chaito

**UM PROTÓTIPO DE UM SISTEMA DE SOFTWARE PARA GERAÇÃO DE  
FORMULÁRIOS PARA COLETA DE DADOS**

Trabalho de Conclusão de Curso submetido à Universidade Federal de Santa Catarina, como parte dos requisitos necessários para a obtenção do Grau de Bacharel em Engenharia de Computação.

Orientador: Prof. Jim Lau, Dr.

Araranguá

2021

Ficha de identificação da obra elaborada pelo autor,  
através do Programa de Geração Automática da Biblioteca Universitária da UFSC.

CHAITO, ALE

UM PROTÓTIPO DE UM SISTEMA DE SOFTWARE PARA GERAÇÃO DE  
FORMULÁRIOS PARA COLETA DE DADOS / ALE CHAITO ;  
orientador, Jim Lau, 2021.

43 p.

Trabalho de Conclusão de Curso (graduação) -  
Universidade Federal de Santa Catarina, Campus Araranguá,  
Graduação em Engenharia de Computação, Araranguá, 2021.

Inclui referências.

1. Engenharia de Computação. 2. Sistema WEB. 3. Coleta  
de Dados Científicos. 4. Pesquisa de Campo e Grupos de  
Pesquisa. 5. Dispositivos Móveis e Internet. I. Lau, Jim.  
II. Universidade Federal de Santa Catarina. Graduação em  
Engenharia de Computação. III. Título.

**Ale Chaito**

# **UM PROTÓTIPO DE UM SISTEMA DE SOFTWARE PARA GERAÇÃO DE FORMULÁRIOS PARA COLETA DE DADOS**

Este Trabalho de Conclusão de Curso foi julgado adequado para obtenção do Título de "Bacharel em Engenharia de Computação" e aprovado em sua forma final pelo Curso de Graduação em Engenharia



Documento assinado digitalmente  
Fabrício de Oliveira Ourique  
Data: 19/05/2021 11:07:54-0300  
CPF: 916.167.860-00  
Verifique as assinaturas em <https://v.ufsc.br>  
de Computação.

---

**Prof. Dr. Fabrício de Oliveira Ourique**

**Coordenador de curso**

## **Comissão Examinadora**



Documento assinado digitalmente  
Jim Lau  
Data: 19/05/2021 10:39:07-0300  
CPF: 613.464.702-00  
Verifique as assinaturas em <https://v.ufsc.br>

---

**Prof. Dr. Jim Lau**

**Orientador**

Universidade Federal de Santa Catarina



Documento assinado digitalmente  
Fábio Rodrigues de La Rocha  
Data: 19/05/2021 10:54:29-0300  
CPF: 963.007.520-20  
Verifique as assinaturas em <https://v.ufsc.br>

---

**Prof. Dr. Fábio Rodrigues De La Rocha**

Universidade Federal de Santa Catarina



Documento assinado digitalmente  
Vinicius Faria Culmant Ramos  
Data: 20/05/2021 11:08:59-0300  
CPF: 095.717.197-81  
Verifique as assinaturas em <https://v.ufsc.br>

---

**Prof. Dr. Vinicius Faria Culmant Ramos**

Universidade Federal de Santa Catarina

**Araranguá, 13 de maio de 2020**

## RESUMO

A rápida evolução dos dispositivos moveis e com uma população brasileira de 82.7% que tem acesso à internet, estes dados permitiram que a coleta de dados para pesquisa de campo também pudesse se transformar. Os antigos formulários presentes nas coletas tradicionais podem ser transformados em páginas WEB, sendo assim os pesquisadores não são impedidos por custo, tempo ou distância. Alguns projetos desenvolvidos pelos profissionais da área da saúde da UFSC de Araranguá ainda utiliza o sistema de coleta de dados através do modelo tradicional, foi através desta necessidade, foi elaborado em conjunto com os pesquisadores da área da saúde um protótipo que pretende facilitar todo este processo de coleta de dados, passando pelas etapas de cadastro de paciente, criação de formulários, análise e extração de resultados. Este protótipo surge como uma ferramenta de auxílio aos pesquisadores realizar uma coleta de dados de modo remoto, por exemplo, sem gerar aglomeração e contato direto, de modo a evitar a propagação do vírus. Além de ajudar os pesquisadores otimizando no levantamento dos dados em campo.

**Palavras-chave:** Sistema WEB, Coleta de Dados Científicos, Pesquisa de Campo, Dispositivos Móveis, Internet.

## **ABSTRACT**

The rapid evolution of mobile devices and with a 82.7% Brazilian population that has access to the internet, these data allowed the collection of data for field research to also be transformed. The old forms present in traditional collections can be transformed into WEB pages, so researchers are not hindered by cost, time or distance. Some projects developed by health professionals at UFSC in Araranguá still use the data collection system using the traditional model, it was through this need, a prototype was developed together with the researchers that aims to facilitate this entire data collection process. , going through the steps of patient registration, creating forms, analyzing and extracting results. This prototype appears as a tool to help researchers conduct data collection remotely, for example, without generating agglomeration and direct contact, in order to prevent the spread of the virus. In addition to helping researchers optimizing data collection in the field.

**Keywords:** WEB System, Scientific Data collect, Field Research, Mobile Devices, Internet.

## Lista de Figuras

Figura 1- Conceitos envolvidos na POO. ....	6
Figura 2 - Camadas da arquitetura MVC.....	7
Figura 3 - Exemplo de arquitetura Cliente Servidor.....	9
Figura 4 - Cobertura do PHP no mercado. ....	10
Figura 5 - Classificação Anual dos frameworks PHP. ....	11
Figura 6 - Estrutura de comunicação entre os componentes da LNMP. ....	13
Figura 7 - Visão macro do sistema e seus componentes. ....	17
Figura 8 - Ciclo de requisições HTTP entre cliente/servidor. ....	20
Figura 9- Casos de uso do usuário.....	21
Figura 10 - Árvore de requisição PHP-FPM. ....	23
Figura 11- Diagrama de Arquitetura. ....	24
Figura 12 - Diagrama de Classes.....	25
Figura 13 - Tela inicial de boas vindas.....	26
Figura 14 - Tela de login do usuário.....	27
Figura 15 - Tela de registro do usuário. ....	27
Figura 16 - Tela inicial dos projetos.....	27
Figura 17 - Tela de criação de novo projeto.....	28
Figura 18 - Tela visão geral dos projetos. ....	28
Figura 19 - Tela de gerenciamento de projeto.....	29
Figura 20 - Formulário de novo questionário.....	29
Figura 21 - Tela de gerenciamento de questionário. ....	30
Figura 22 - Tela de gerenciamento de um bloco. ....	30
Figura 23 – Questões retirados de um formulário real, utiliza-se apenas letras e números. ....	31
Figura 24 - Questões retirados de um formulário real, utiliza-se apenas tipo de opções já definidas. ....	31
Figura 25 - Questões retirados de um formulário real, utiliza-se apenas números. ....	32
Figura 26 - Questões retirados de um formulário real, opção de sim ou não.....	32
Figura 27 - Tela para adicionar nova questão.....	<b>Error! Bookmark not defined.</b>
Figura 28 - Tela para visualizar participantes de um projeto. ....	32
Figura 29 - Tela para inserir um novo usuário a um projeto. ....	33
Figura 30 - Tela para pesquisar por pacientes cadastrados para avaliar.....	33
Figura 31 - Tela para submeter um paciente a um formulário. ....	34
Figura 32 - Tela de exportar avaliações/levantamento de dados...	<b>Error! Bookmark not defined.</b>

## SUMÁRIO

Ale Chaito.....	3
RESUMO .....	4
ABSTRACT .....	5
1. INTRODUÇÃO .....	1
1.1. JUSTIFICATIVA .....	2
1.2. OBJETIVOS .....	3
1.2.1. Objetivo Geral .....	3
1.2.2. Objetivos Específicos.....	3
1.3. METODOLOGIA.....	4
1.4. ORGANIZAÇÃO DO TRABALHO.....	4
2. FUNDAMENTAÇÃO TEÓRICA.....	6
2.1. POO – PROGRAMAÇÃO ORIENTADA A OBJETOS .....	6
2.2. MVC – DESIGN PATTERN .....	7
2.3. ARQUITETURA CLIENTE/SERVIDOR .....	8
2.4. FERRAMENTAS UTILIZADAS .....	9
2.4.1 PHP framework Laravel.....	9
2.4.2 Framework Bootstrap.....	11
2.4.3 Banco de dados MySQL.....	12
2.4.4 LNMP stack .....	12
2.5. TRABALHOS RELACIONADOS.....	13
3 DESCRIÇÃO DA ARQUITETURA E MODELAGEM DO SISTEMA.....	17
3.1. LEVANTAMENTO DE REQUISITOS .....	18
3.1.1 Requisitos Funcionais.....	18
3.1.2 Requisitos Não Funcionais .....	18
3.2. BLOCO CLIENTE .....	19
3.2.1 Casos de uso.....	20
3.3 BLOCO SERVIDOR .....	22
3.4 DIAGRAMA DE ARQUITETURA.....	24
3.5 DIAGRAMA DE CLASSES.....	25
4 IMPLEMENTAÇÃO .....	26
4.1 REQUISITOS DE HARDWARE.....	26



4.2 PROTÓTIPO.....	26
4.3 VALIDAÇÃO.....	35
5 CONCLUSÃO .....	37
5.2 TRABALHOS FUTUROS.....	37
5.2.1 Política de Controle de Acesso .....	37
5.2.2 Digitalização de Formulários .....	38
5.2.3 Banco de questões .....	38
5.2.3 Formularios Dinâmicos.....	38
REFERÊNCIAS.....	39

# 1. INTRODUÇÃO

Nos últimos anos, o mercado de dispositivos móveis evoluiu de maneira muito rápida, devido a três fatores: a queda dos preços, o lançamento de dispositivos móveis com alta capacidade de processamento e o surgimento de novas tecnologias para o desenvolvimento de Aplicações Móveis (Apps) [Santos, et al.,2004]. Estes dispositivos juntamente com a Internet proporcionaram um meio para a coleta de dados e a difusão das informações que antes não era possível de ser realizada, criando condições para um novo tipo de aplicação de Coleta Móvel de Dados (CMD) [Rezende et al. 2010]. Os pesquisadores não ficam mais limitados pelo tempo de criação dos formulários, custo e distância, podendo acessar o mundo quase que instantaneamente a um custo mais baixo. A coleta de dados através de questionário pode divergir do modelo tradicional impresso, ver o Anexo A para visualizar um exemplo de formulário, permitindo que os pesquisadores utilizem uma interface mais interativa e rica com conteúdo multimídia, tanto na coleta dos resultados quanto na exibição [FREITAS, et al.,2004].

Hoje é possível aplicar um questionário de avaliação de paciente no período da manhã e coletar os dados e suas respectivas análises ao final da tarde. A etapa de análise de dados é uma tarefa que normalmente acontece apenas no final da coleta de dados, agora pode ser acompanhada praticamente em tempo real, além disso o questionário pode ser mantido permanentemente na rede, o que significa que a quantidade de dados coletados nunca para de evoluir. Portanto, deve-se reconsiderar a definição do esquema de cruzamento de dados e a análise online dos resultados, e se preparar com antecedência [FREITAS, et al.,2004].

Em certa medida, esse método requer um entendimento da análise que deve ser realizada *a priori*, ou seja, antes da coleta de dados, o pesquisador precisa ter uma orientação prévia dos resultados que deseja atingir com sua pesquisa para que os dados possam ser apresentados e discutidos em tempo real à medida que são coletados. Isso porque, de fato, os aplicativos de pesquisa realizados pela Internet podem pré-visualizar os resultados das análises e armazená-los em um servidor web, de onde podem ser acessados globalmente [Boni, 2016].

Para aproveitar as possibilidades de pesquisa proporcionadas pela Internet, é necessário repensar os métodos de pesquisa, desenvolvimento e propor novas soluções

mais adequadas a esse ambiente. Também é necessário revisar os papéis dos diferentes participantes no processo de pesquisa, como eles participam, as ferramentas disponíveis para a comunidade de pesquisa e até mesmo para todos os analistas e executivos [Boni, 2016].

## **1.1.JUSTIFICATIVA**

Um levantamento feito pelo IBGE em 2019 mostra que 82,7% da população brasileira possui acesso a internet, desse total 86,7% estão localizados em uma area urbana e 55,6% na area rural [Educa IBGE, 2019]. Estes dados mostram, por exemplo, a viabilidade no desenvolvimento de um sistema de coleta de dados através de dispositivos móveis conectados à internet que além de facilitar o trabalho do entrevistador também facilita para o entrevistado.

Atualmente, muitos os projetos realizados pelos profissionais da fisioterapia e medicina, presentes na Universidade Federal de Santa Catarina (UFSC), campus Araranguá, realizam coleta de dados hoje na forma tradicional através de questionários impressos e se deslocando até locais específicos. Uma pesquisa realizada com estes profissionais buscou identificar as principais queixas ao se utilizar o método tradicional para coleta de dados, foram elas:

- Tempo gasto na criação de questionários;
- Custo e tempo gasto na impressão de questionários em papel;
- Digitalização dos dados coletados;
- Custo com transporte e armazenamento dos questionários;
- Impacto ambiental gerado pelo uso de papel impresso;
- Custo e tempo gasto no deslocamento do entrevistador;

Outro ponto, não menos importante a ser levado em conta é que devido a pandemia de Covid-19 que o Brasil está enfrentando [Rodriguez Morales, 2020], deve-se evitar o contato e aglomeração de pessoas que não sejam essenciais.

Pensando nestes fatores citados anteriormente, já que os dispositivos digitais estão cada vez mais difundidos entre a população em geral [Rangel,2015], e que já existem ferramentas bem sucedidas na área da saúde vislumbra-se a possibilidade de criar uma ferramenta capaz de solucionar ou minimizar os problemas encontrados durante uma

pesquisa de campo. Além disso, esta ferramenta pode ser criada de forma genérica para atender outras áreas além da saúde, já que outros processos de pesquisa de campo compartilham de muitos dos problemas aqui citados.

No Brasil temos ótimos exemplos de avanços na área denominada “Informática Médica”, como o Cartão Nacional de Saúde e o Registro Eletrônico de Saúde, estes são sistemas informatizados governamentais que possuem grande abrangência do território e grande número de usuários que utilizam. Sistemas deste tipo se assemelham muito a ferramenta proposta neste trabalho visto que no Registro Eletrônico de Saúde, por exemplo, os dados do paciente são armazenados para que seja possível prevenir e diagnosticar possíveis doenças, assim como seria feito utilizando um questionário gerado pela ferramenta. Isto mostra que o ecossistema está propício e pronto para receber mais sistemas deste tipo [Barros, et al.,2002].

## **1.2.OBJETIVOS**

### **1.2.1. Objetivo Geral**

Este trabalho tem como objetivo de desenvolver um protótipo de uma plataforma WEB para pesquisadores e acadêmicos da área da saúde interessados em realizar pesquisas de campo através de um sistema de coleta de dados baseados em questionários.

### **1.2.2. Objetivos Específicos**

Buscando atingir o objetivo geral deste trabalho, foram elencados os seguintes objetivos específicos:

- Permitir o cadastro e gerenciamento de pacientes;
- Exportar os dados coletados dos pacientes;
- Habilitar o gerenciamento de grupos de pesquisa;
- Permitir que diferentes usuarios colaborem para criação de formulario;
- Criação de formulários de perguntas e respostas para avaliação;
- Criação de formulários com subcategorias de perguntas;

### **1.3.METODOLOGIA**

Este trabalho possui caráter experimental e exploratório, desenvolvido na Universidade Federal de Santa Catarina - UFSC, com colaboração dos profissionais fisioterapeutas presentes no campus Araranguá. Além disso, todo aparato científico e tecnológico adquirido durante a revisão da literatura foi fundamental para o processo.

Sendo assim nosso software foi construído através de um planejamento que foi disposto e descrito nas seguintes etapas:

- Realizar revisão bibliográfica sobre quais métodos e ferramentas foram utilizadas nos softwares semelhantes;
- Levantamento das necessidades dos profissionais de saúde e seus modelos de questionários;
- Levantamento e análise dos requisitos funcionais e não funcionais;
- Modelagem dos diagramas de classe, caso de uso e diagrama de arquitetura;
- Validar a interface gráfica responsiva em diversos dispositivos;
- Construir a arquitetura cliente servidor para fornecer acesso ao software;
- Análise e discussão do desempenho e robustez do software;

### **1.4.ORGANIZAÇÃO DO TRABALHO**

Esta monografia está organizada em 6 capítulos e representa o trabalho de conclusão de curso desenvolvido para a Graduação de Engenharia de Computação na Universidade Federal de Santa Catarina.

- Capítulo 1 – Este capítulo é apresentado uma introdução da problemática envolvida nesse trabalho e apresentada, além da justificativa, dos objetivos gerais e específicos, metodologia utilizada para o desenvolvimento do protótipo e monografia.
- Capítulo 2 - Este capítulo apresentados alguns conceitos, ferramentas, tecnologias e métodos que foram aplicados para o desenvolvimento do software. Sendo exibidos conceitos de Programação Orientada a Objetos (POO), Design Pattern

MVC e Arquitetura Cliente/Servidor, além disso é apresentado o framework Laravel.

- Capítulo 3 - São apresentadas as modelagens criadas para facilitar e guiar o desenvolvimento do software como diagrama de classes, casos de uso e diagrama de arquitetura. Também são explanados os passos para construção efetiva do software como modelagem do banco de dados, criação da interface utilizando HTML e CSS e do backend utilizando o framework Laravel.
- Capítulo 4 – Neste capítulo é apresentado o protótipo com todas as principais telas do sistema e suas principais funcionalidades, também são mostrados quais foram os requisitos de hardware que foram utilizados para o desenvolvimento.
- Capítulo 5 - Por fim temos as considerações finais e conclusão sobre o trabalho proposto, em seguida são sugeridos trabalhos futuros que podem ser explorados em um outro trabalho. Também são apresentados as referências utilizadas e o código da plataforma desenvolvida.

## 2. FUNDAMENTAÇÃO TEÓRICA

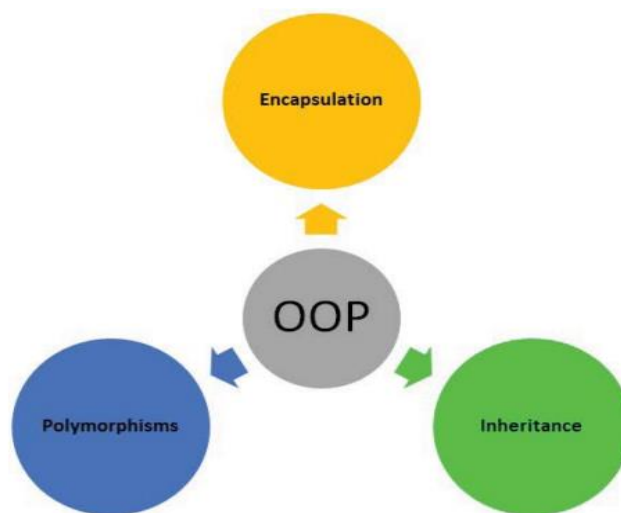
Neste capítulo são apresentados alguns conceitos, ferramentas e métodos que embasaram o desenvolvimento do projeto como a programação orientada a objetos, métodos ágeis e arquitetura de software MVC.

### 2.1.POO – PROGRAMAÇÃO ORIENTADA A OBJETOS

A programação orientada a objetos faz parte da ciência da computação desde o final da década de 1980 e começo de 1990, durante 1990 este conceito teve ampla adoção tanto na indústria como na academia [Berndtsson,2005].

A ideia por trás da POO é derivada da representação humana do conhecimento no cérebro de acordo com o mundo real, a ideia de POO foi desenvolvida por cientistas cognitivos. Este paradigma consiste no objeto que possui atributos (propriedades e métodos (ações) [Hourani,2019]. A Figura 1 evidencia os principais conceitos envolvendo a POO.

Figura 1- Conceitos envolvidos na POO.



Fonte: Hourani, 2019.

- Encapsulamento: “É uma técnica de software que consiste em isolar as funções do sistema ou um conjunto de dados e operações nestes dados com um módulo e fornecer especificações precisas para este módulo” [IEEE,2017].
- Herança: “E uma notação semântica das responsabilidades (propriedades e restrições) das subclasses são consideradas para incluir as responsabilidades de uma superclasse, além da sua própria” [IEEE,2017].

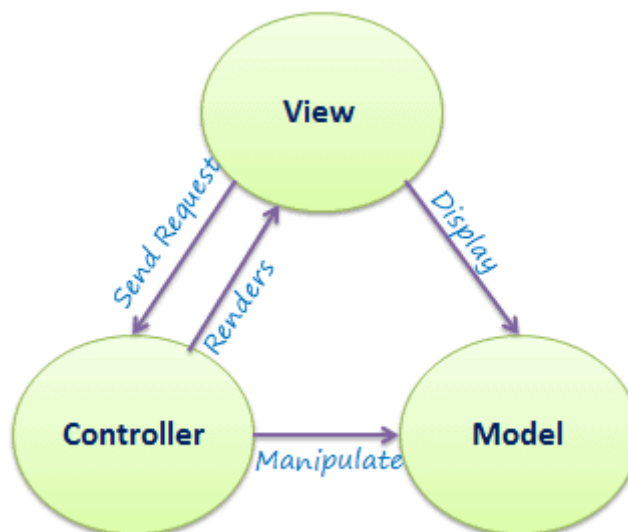
- Polimorfismo: É a implementação dos mesmos métodos e operadores de maneiras diferenciadas ou diferentes [Hourani,2019].

O paradigma orientado a objetos fez surgir a linguagem de modelagem unificada (UML) uma abordagem para especificar, visualizar, construir e documentar sistemas de software. Representa a reunião das melhores práticas de engenharia provendo com sucesso uma base para modelagem de grandes e complexos sistemas [Wang,2012].

## 2.2. MVC – DESIGN PATTERN

O conhecido padrão de design *Model/View/Controller* (MVC) é um caminho muito útil para arquiteturas de sistemas de softwares interativos. No MVC a *View* mostra as informações para o usuário junto com o *Controller* que processa as interações dos usuários. O *Model* é a parte da aplicação que contém a informação representada pela *View* e a lógica que muda essa informação como resposta a interação do usuário [Wang,2012], a Figura 2 exemplifica isto.

Figura 2 - Camadas da arquitetura MVC.



Fonte: O proprio autor.

Este modelo ganhou relevância e amplo uso graças às grandes vantagens que oferece, sendo elas:

- Facilita o reaproveitamento de código;
- Facilidade na manutenção e adição de recursos;



- Maior integração da equipe e/ou divisão de tarefas;
- Diversas tecnologias estão adotando essa arquitetura;
- Facilidade em manter o seu código sempre limpo;

Porém apesar das vantagens este modelo traz um problema à tona quando utilizado para desenvolver aplicações web que surge do fato das aplicações web serem intrinsecamente particionadas entre o cliente e o servidor. A View, é claro, é exibida no cliente; porém o Model e o Controller podem teoricamente ser particionados de muitas formas entre o cliente e o servidor. Isto implica na necessidade de se pensar em como isto será particionado e qual linguagem poderá ser utilizada, como por exemplo a linguagem JavaScript e processada do lado do cliente já a linguagem PHP e processada pelo lado do servidor [Zhang, et al.,2013].

Como o modelo MVC fornece uma linguagem comum para os programadores foram criados diversos Frameworks para as linguagens mais utilizadas como:

- PHP: Laravel, CodeIgniter, Symfony, Cake PHP, Zend Framework;
- JAVA: Struts, Spring;
- Python: Giotto, Django;
- JavaScript: Kendo, Angular, Ember, Backbone, React;

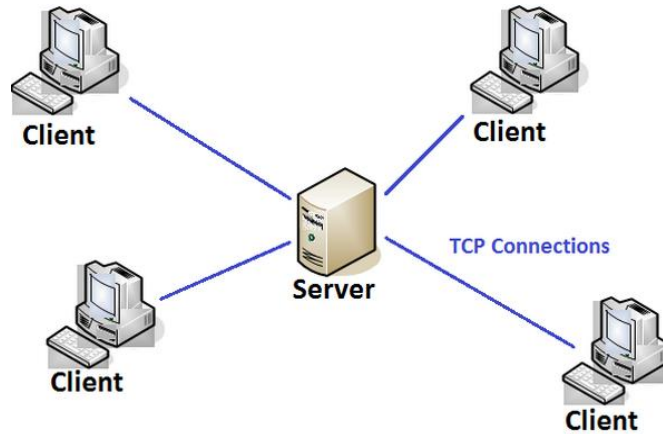
### **2.3. ARQUITETURA CLIENTE/SERVIDOR**

Arquitetura Cliente/Servidor surgiu como uma novidade comparada às antigas arquiteturas antes utilizadas como os Sistemas Centralizados e os Micros Standalone, ou seja, sistemas que trabalham isolados.

Nesta arquitetura temos dois papéis bem definidos que são os clientes e servidores, o cliente é classificado como qualquer equipamento que necessita de outro, melhor dizendo é o consumidor de serviços. O servidor por outra vez é o provedor desses serviços. Quando conectados por redes as requisições podem ser feitas de qualquer local independente de sua localização [Rabelo, et al.1998]. Na

Figura 3 podemos ver um exemplo de uma arquitetura cliente servidor onde o tráfego de dados entre os atores é realizado através de conexões TCP na camada de transporte.

Figura 3 - Exemplo de arquitetura Cliente Servidor.



Fonte: O Autor.

Apesar de seus papéis serem bem definidos, qualquer máquina pode atuar tanto como cliente, como servidor, dependendo da operação que está executando no momento. De forma geral os clientes abrem requisições que são atendidas pelos servidores. As vantagens deste modelo em comparação a outras arquiteturas de rede são:

- Centralizado;
- Fácil manutenção;
- Escalabilidade: podemos aumentar o número de clientes e servidores por separado;

Todavia, a maior desvantagem é que um número maior de clientes acarretará em mais problemas para o servidor, devido ao congestionamento que será gerado pelo tráfego (Lizama, 2016).

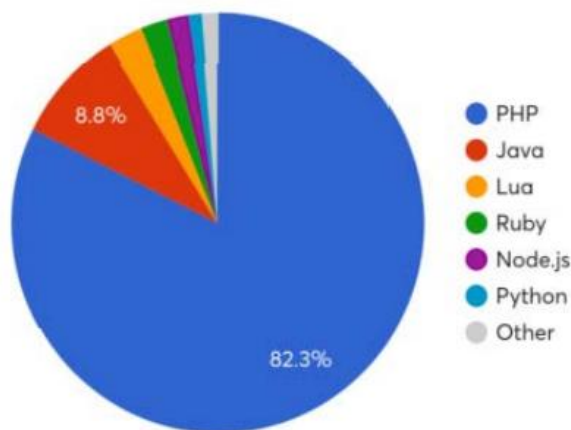
## 2.4.FERRAMENTAS UTILIZADAS

Nesta seção são apresentadas as ferramentas e tecnologias que foram escolhidas para construção da ferramenta e do servidor web responsável por atender as requisições dos clientes. Em cada subseção estão presentes alguns dados gerais sobre a ferramenta e/ou a tecnologia e em seguida é elucidado a motivação por trás de cada escolha.

### 2.4.1 PHP framework Laravel

De acordo com a W3Techs, o PHP é a linguagem de programação server-side mais popular, presente em 81.4% dos websites conhecidos. Muitos sabem que sites como Facebook e Wikipedia também utilizam sistemas gerenciadores de conteúdo escritos em PHP. Devido à sua sintaxe simples e de funções e variáveis de tipagem dinâmica além de uma grande quantia de recursos integrados, a linguagem é fácil de aprender para iniciantes.

Figura 4 - Cobertura do PHP no mercado.



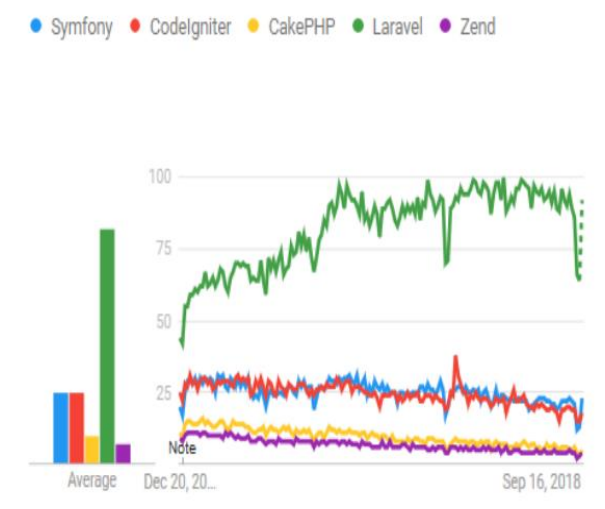
Fonte: N. Yadav, 2019.

O PHP está tomando todo mercado de desenvolvimento web, porém para desenvolver um website deve-se ter conhecimento sobre linguagem PHP e outras linguagens como CSS e Javascript [Xiaosheng, 2010]. A realização de um código PHP é feita em páginas da web HTML. Para construir um site em PHP é demorado e exige muito conhecimento técnico. As vezes recursos como autenticação de usuários que são muito utilizados têm de ser reescritos em diversos projetos por estes e outros motivos os frameworks foram criados. Na moderna indústria de TI, é esperado que os desenvolvedores possuam conhecimento de pelo menos um framework.

O Laravel é um projeto open source WEB desenvolvido em PHP, criado por Taylor Otwell, para criação de aplicativos WEB baseados no padrão de arquitetura MVC e baseado no Symfony. Sua primeira versão beta surgiu em 9 de julho de 2011 (Figura 5). Sendo classificado como um dos mais bem sucedidos frameworks mais bem sucedidos em 2019, com base em uma pesquisa anual conduzida pelo SitePoint. Os frameworks PHP da web oferecem diversas vantagens e benefícios graças a reutilização de componentes. Frameworks são usados para superar os problemas que enfrentamos

durante o desenvolvimento de um aplicativo web com PHP como a falta de segurança ao criar um sistema de autenticação de usuários. Estes frameworks promovem o rápido desenvolvimento de aplicativos (RAD), o que economiza nosso tempo e ajuda na construção de um ambiente mais estável para o aplicativo [N. Yadav, 2019].

Figura 5 - Classificação Anual dos frameworks PHP.



Fonte: Survey of SitePoint, 2019.

Sendo assim o framework oferece um caminho rápido e fácil para construção de aplicativos WEB, fornecendo recursos como autenticação, roteamento, gerenciamento de sessão, caching e containers IoC, além de outros recursos amplamente usados como ferramentas de migração de banco de dados e suporte a teste unitário integrado [Laravel, et al. 2020]. Graças a sua versatilidade este framework foi escolhido para a construção da plataforma.

## 2.4.2 Framework Bootstrap

O Bootstrap é um framework CSS grátis e open source dirigido a responsividade. contém CSS, HTML e JavaScript planejados para designs de tipografias, formulários, botões, navegações e outros componentes de interface. É considerado o framework open source mais usado pelos desenvolvedores, o projeto está disponível no Github e conta com mais de 142.000 estrelas, tornando o projeto com mais estrelas da plataforma [Bootstrap, et al. 2020].

Graças a modularidade de componentes esta ferramenta agiliza e muito a criação de interfaces gráficas responsivas, totalmente padronizadas graças a seus componentes e

validados por usuários de todo mundo, por estes motivos foi o framework escolhido para o desenvolvimento de nossa interface.

### **2.4.3 Banco de dados MySQL**

Um SGDB (Sistema Gerenciador de Banco de Dados) é definido por Silberschatz e Korth [2006] como um conjunto de programas utilitários e de dados inter-relacionados que permitem acesso e manipulação a estes dados. O principal foco de um SGDB é fornecer um local eficiente para recuperação e armazenamento de informações.

Escrito em C e C ++, é um analisador SQL escrito em Yet Another Compiler-Compiler (YACC), mas usa um analisador léxico próprio. O Mysql funciona em muitos sistemas operacionais como FreeBSD, Linux, macOS, Microsoft Windows entre outros. Possui clientes autônomos que permitem que usuários interajam diretamente com o banco de dados MySQL usando SQL, porém mais do que isso é usado com outros programas para implementar aplicações que necessitam de banco de dados relacionais [Mysql, et. al 2011].

Algumas das funcionalidades presentes no MySql são:

- Suporte multi plataforma;
- Gatilhos;
- Cursores;
- Esquema de informação;
- Suporte para SSL;
- Caching de consultas;
- Sub selects;
- Indexação por texto e pesquisa;
- Suporte a UNICODE;
- Motores de armazenamento nativos InnoDB, MyISAM, Merge, Memory (heap);

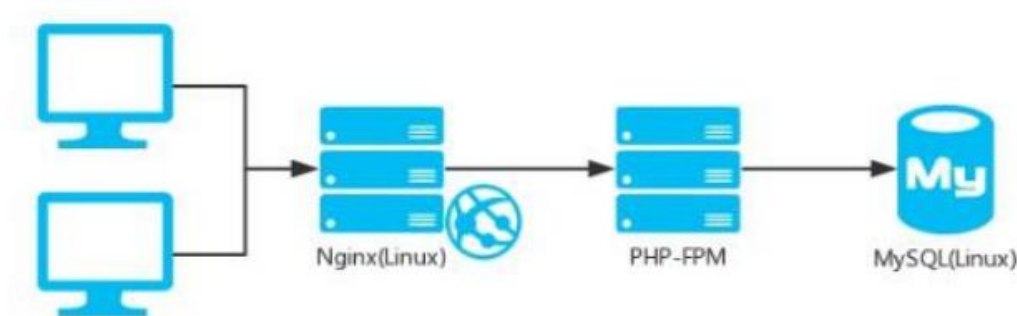
O fato de performar bem em hardwares de baixo desempenho faz com que o MySQL seja o SGDB mais utilizado no mundo, além disso em testes de comparação de desempenho se mostra 30% mais rápido que seus concorrentes FireBird e PostgreSQL como mostra Souza e Pires [2008]. Graças ao seu desempenho muito bom, amplo uso e funcionalidades, este foi o motor de armazenamento escolhido para este trabalho.

### **2.4.4 LNMP stack**

LNMP é um modelo de arquitetura para desenvolvedores web, nomeado como um acrônimo dos itens que o compõem, sendo 4 componentes de código aberto: o sistema operacional Linux, o servidor HTTP Nginx, o MySQL e a linguagem de programação PHP. Como um pacote de soluções, é adequado para construção de sites dinâmicos e aplicações web.

Este pacote de componentes atende muito bem a necessidade de nossa plataforma e será utilizado para compor a arquitetura cliente/servidor deste trabalho. A Figura 6 mostra toda a ligação entre os componentes citados.

Figura 6 - Estrutura de comunicação entre os componentes da LNMP.



Fonte: Xianjun Chen, 2017.

## 2.5. TRABALHOS RELACIONADOS

Na UNIFESP em 2013 foi desenvolvido o projeto Maritaca, uma infraestrutura para a coleta de dados a partir de dispositivos móveis Android, permite que qualquer pessoa sem conhecimento em programação crie uma aplicação para coleta de dados [dos Santos, et al.,2013]. Nesta ferramenta os formulários são transformados em telas de aplicativo por onde o usuário navega e responde às questões.

Um outro projeto desenvolveu um software de coleta de dados através de dispositivos móveis equipados com Microsoft Windows Mobile 2003 e a ferramenta de desenvolvimento Microsoft Visual Studio 2005 com a plataforma .NET e linguagem C#. Utilizaram a abordagem de programação orientada a objetos e micro serviços, os questionários desenvolvidos por este software possuem suporte a questões de múltipla escolha, questões abertas e gravação de voz [Boni, et al.,2006].

Uma solução robusta para coleta de dados de campo usando plataformas móveis chamado e-Phenology Collector foi criado na UNICAMP, este trabalho além de desenvolver o software se preocupou com as possíveis falhas advindas do uso de

dispositivos móveis denominado “Dependability” que são a disponibilidade, confiabilidade, integridade dos dados e performance [Bollis,2016].

Para coleta de dados de pacientes em unidades ambulatoriais em 2015 foi desenvolvido um instrumento informatizado, que foi elaborado utilizando Java e JavaServer Page para o banco de dados persistente que utilizou-se MySQL. Este trabalho, porém, não teve exclusiva preocupação em funcionar em dispositivos móveis [Rangel,2015]. Neste sistema, o usuário precisa cadastrar seus pacientes e criar formulários com questões de texto aberto, sem oferecer outras possibilidades.

Em [Roberto, et al.,2014] diagnosticou que os anestesiológicos muitos dos softwares existentes no mercado não são práticos para o uso diário além do alto custo de aquisição. Tendo isto em vista foi desenvolvido uma plataforma que consiste em um formulário pré-anestésico que inclui questões de múltipla escolha, caixas de texto predefinido e livre. Tudo isto foi feito sem custos adicionais e com compatibilidade com os computadores já existentes, sem a necessidade de instalação de um software adicional.

O CIF Fisio Hospitalar é uma ferramenta que permite a criação de documentos eletrônicos para auxiliar na classificação internacional de funcionalidade e incapacidade (CIF), desenvolvido especialmente para ajudar fisioterapeutas e terapeutas ocupacionais na aplicação desta avaliação de pacientes como mostra [Paschoal, et al.,2019].

Em [NEGREIROS,2010] também desenvolveu um trabalho relacionado a CIF, um sistema de avaliação digital utilizando Java e MySQL, este por sua vez só funciona em computadores desktop, mas também busca auxiliar fisioterapeutas e terapeutas ocupacionais que sem essa ferramenta seria necessário a utilização de formulários de papel para avaliar seus pacientes.

Um sistema digital de avaliação e pesquisa baseado na Classificação Internacional de Doenças (CID) foi desenvolvido por [Atkinson and Nixon-Cave,2011], permite que sejam criados formulários baseados na CID e sejam direcionados aos médicos especializados naquela patologia identificada. Esta ferramenta foi criada especialmente para atender a demanda do hospital da universidade do Chiba, que tinha dificuldades em encontrar especialistas de uma forma eficiente.

Um dos principais temas de pesquisa e desenvolvimento no âmbito da Informática Médica é o Prontuário Eletrônico do Paciente (PEP), seu grande valor tanto para o paciente como para a equipe médica o faz ganhar destaque. Este sistema tem como

objetivo auxiliar e/ou substituir o prontuário em papel que é um agrupamento de informações sobre o histórico e estado do paciente como mostra [Costa,2001].

LimeSurvey é uma ferramenta gratuita e open source online de estatística escrito em PHP com banco de dados MySQL. Um software web que permite que usuários criem interfaces para desenvolver e publicar online, coletar dados, criar estatísticas e exportar o resultado para outras aplicações [LimeSurvey, 2021].

Google Forms é um software de administração de enquetes, oferecido pelo Google. Esta ferramenta permite que o usuário crie e edite enquetes online em colaboração com outros usuários em tempo real. Todos os dados podem ser automaticamente exportados para ferramentas de visualização de dados [Google Forms, 2021].

Todos estes trabalhos citados anteriormente possuem certo grau de semelhança com o que se pretende desenvolver aqui, porém todos são desenvolvidos de maneira muito específica e não possuem uma abordagem mais abrangente, o que dificulta que o sistema seja extrapolado para outros meios. Além disso, a maioria dos sistemas apresentados anteriormente não tem preocupação de funcionar em diferentes dispositivos, fato este que diminui a abrangência do sistema, outro ponto importante é que os questionários são muito limitados e não oferecem diferentes tipos de questões para que o impedimento criativo seja apenas do usuário durante a criação.

Sendo assim, o sistema proposto aqui pode ser utilizado para diversos meios e fins na área da saúde como também para outras áreas, sua principal característica se dá na flexibilidade buscando resolver um problema mais amplo que é a criação e digitalização de formulários de modo geral. Além do escopo aberto do sistema oferece a possibilidade de visualização de dados através da exportação em CSV ou XLSX, com foco em gerenciar grupos de pesquisa.

A tabela a seguir ilustra quais as funcionalidades presentes em nosso software em relação as outras ferramentas existentes na literatura.

Tabela 1 – Comparação de Funcionalidades

Ferramenta	Criação de Questionário	Cadastro de Pacientes	Exportar Dados	Grupos de Pesquisa	Responsividade
<b>Google Docs</b>	X	-	X	-	X
<b>LimeSurvey</b>	X	-	X	-	X
<b>E-Phenology</b>	X	-	-	-	-



<b>Maritaca</b>	X	-	-	-	-
<b>CIF FISIO Hospital</b>	X	-	-	-	-
<b>Trabalho Proposto</b>	X	X	X	X	X

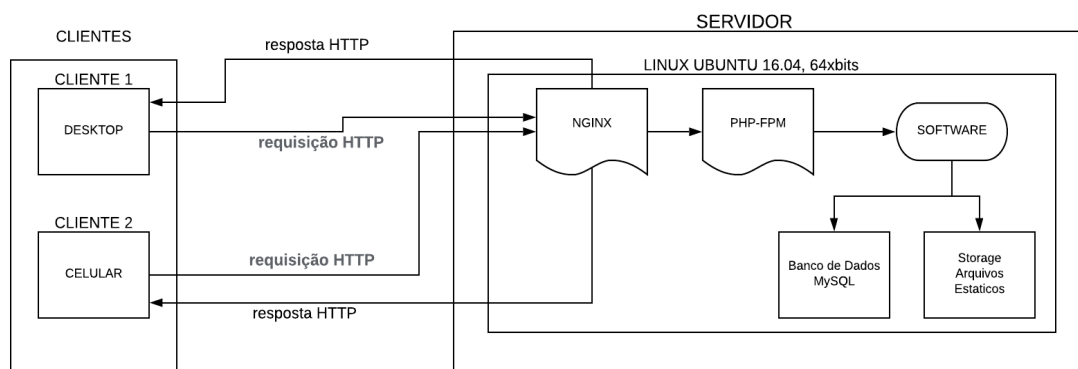
### 3 DESCRIÇÃO DA ARQUITETURA E MODELAGEM DO SISTEMA

Nosso sistema é composto por dois grandes blocos. O primeiro deles é o bloco do cliente e o segundo o bloco do servidor, estes por sua vez representam a nossa arquitetura macro de cliente/servidor cada um com seu respectivo papel. O cliente por sua vez efetua requisições HTTP que são enviadas para o servidor responsável por devolver uma resposta.

Descrevendo com um pouco mais de detalhes a requisição HTTP gerada pelo cliente tem origem no browser instalado no dispositivo, a ação requisitada pelo cliente é empacotada e enviada através da rede de internet até o servidor. Quando chega ao servidor o pacote de dados é recebido pelo Nginx que após o processamento identifica que no conteúdo do pacote existe código PHP, sendo assim então solicita a ajuda do PHP-FPM para interpretar. Por fim, consegue redirecionar o pacote para a rota HTTP correta presente no software. Resumindo, cada pacote de dados enviado pelo cliente tem como origem uma ou mais rotas presentes no software.

O caminho de volta do pacote de dados até o cliente tem como origem o software que executa a ação requisitada e então irá com ajuda do PHP-FPM e o Nginx encapsular a resposta HTTP e enviar novamente através da rede internet até chegar ao cliente para ser interpretado pelo browser, completando assim o ciclo de requisições. A Figura 7 apresenta uma visão geral de nosso sistema e seus componentes.

Figura 7 - Visão macro do sistema e seus componentes.



Fonte: O autor.

### **3.1. LEVANTAMENTO DE REQUISITOS**

O principal objetivo desta etapa é definir os requisitos do software que será desenvolvido, não existe nenhum formato ideal geral para isto. Os requisitos foram elencados com base na bibliografia existente e complementados juntamente com profissionais fisioterapeutas presentes no Campus da Universidade Federal de Santa Catarina em Araranguá. Os requisitos foram descritos em linguagem natural e foi composto pelas necessidades identificadas pelos profissionais de quais serviços o software deve propiciar.

Posteriormente os requisitos devem passar por um refinamento para que as funcionalidades e serviços do software estejam claros e satisfatórios, com o objetivo de manter em harmonia o que se é requisitado e do que pode ser desenvolvido com as ferramentas e linguagens existentes.

#### **3.1.1 Requisitos Funcionais**

- RF01: Autenticação de usuários;
- RF02: Cadastro de usuários;
- RF03: Permitir a criação e gerenciamento de projetos onde estão inseridos os questionários que são compostos por diversos blocos e questões;
- RF04: Permitir que diferentes usuários colaborem entre si para criação e gerenciamento dos formulários;
- RF05: Permitir que usuários sejam inseridos ou excluídos de projetos pelo seu administrador;
- RF06: Permitir o cadastro de pacientes;
- RF07: Permitir submeter os pacientes a avaliação através dos questionários criados;
- RF08: Permitir que sejam exportados os resultados referentes à avaliação dos pacientes;
- RF10: Permitir com que os usuários editem seus dados cadastrados;
- RF11: Ferramenta deve ser responsiva utilizável em diversos tamanhos de tela;

#### **3.1.2 Requisitos Não Funcionais**

- RNF01: Prover a integridade e segurança dos dados que trafegam pela aplicação.
- RNF02: A linguagem utilizada e o PHP 7.0 com framework Laravel;
- RNF03: O banco de dados utilizado deve ser relacional MySQL;
- RNF04: Será necessário um web server para rodar toda ferramenta e linguagens utilizaremos o Nginx;

### 3.2. BLOCO CLIENTE

O bloco do cliente pode ser constituído por um ou diversos clientes, o fato de nosso software ser totalmente responsivo permite que os clientes efetuem acesso ao sistema a partir dos mais variados tipos de dispositivos, como: celulares, desktops, notebooks, laptops ou tablets. Apesar de existir essa heterogeneidade de dispositivos todos possuem um ponto em comum, geralmente estes dispositivos possuem um navegador instalado por padrão no seu sistema operacional, e esta ferramenta permite que o pacote HTTP recebido seja transformado em uma interface gráfica composta por HTML e CSS.

Nossa interface gráfica foi construída utilizando o framework Bootstrap citado anteriormente na seção 2.5.2 a partir desta interface e que são geradas as requisições feitas ao servidor. Por exemplo através dos formulários são enviadas as requisições HTTP do tipo POST, já através dos links são enviadas as requisições do tipo GET ou DELETE.

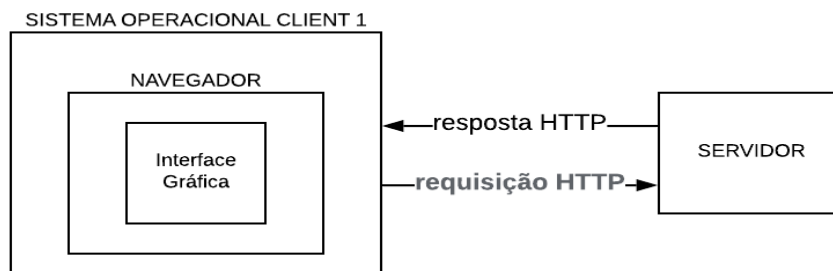
As requisições HTTP são dos mais variados tipos, como:

- **GET:** O método GET solicita a representação de um recurso específico. Requisições utilizando o método GET devem retornar apenas dados.
- **POST:** O método POST é utilizado para submeter uma entidade a um recurso específico, frequentemente causando uma mudança no estado do recurso ou efeitos colaterais no servidor.
- **HEAD:** O método HEAD solicita uma resposta de forma idêntica ao método GET, porém sem conter o corpo da resposta.
- **PUT:** O método PUT substitui todas as atuais representações do recurso de destino pela carga de dados da requisição.
- **DELETE:** O método DELETE remove um recurso específico.
- **CONNECT:** O método CONNECT estabelece um túnel para o servidor identificado pelo recurso de destino.

- **OPTIONS:** O método OPTIONS é usado para descrever as opções de comunicação com o recurso de destino.
- **TRACE:** O método TRACE executa um teste de chamada loopback junto com o caminho para o recurso de destino.
- **PATCH:** O método PATCH é utilizado para aplicar modificações parciais em um recurso.

O ciclo de requisições e os componentes envolvidos com foco no lado do cliente estão presentes na Figura 8.

Figura 8 - Ciclo de requisições HTTP entre cliente/servidor.



Fonte: O Autor.

### 3.2.1 Casos de uso

Em engenharia de software e engenharia de sistemas, os casos de uso são uma lista de ações ou eventos tipicamente definindo interações entre o ator e o um domínio em específico para alcançar um objetivo. O ator pode ser um humano ou outro.

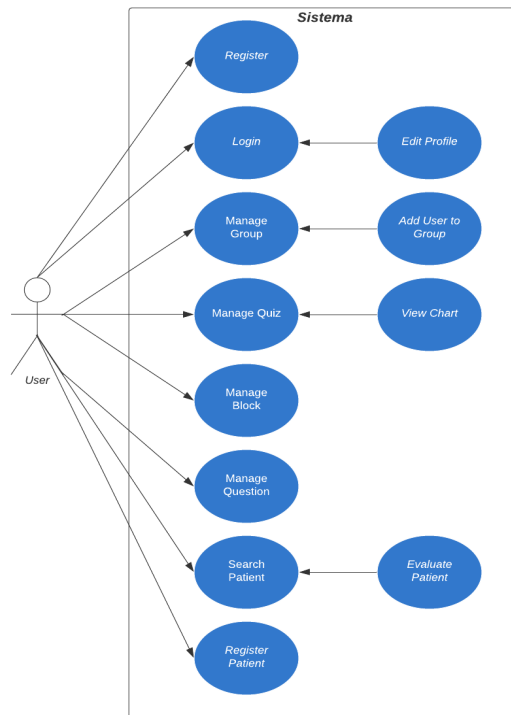
Os casos de uso apresentados na Figura 9 no nosso caso definem as interações possíveis do cliente com o software desenvolvido, as diversas interações contemplam o levantamento dos requisitos funcionais presentes na seção **Error! Reference source not found.**

Tabela 2 – Descrição dos Casos de Uso.

Ação do Ator	Resposta do Sistema
--------------	---------------------

Permitir que o próprio usuário faça o cadastro para utilizar o sistema.	O sistema cadastra o usuário e permite que ele acesse as funcionalidades.
Realizar a autenticação do usuário através de email e senha.	O sistema verifica os dados e autentica se o usuário existe.
Permitir que o usuário gerencie grupos de pesquisa e insira novos participantes.	O sistema insere novos participantes ao projeto e permite que haja colaboração.
Permitir que o usuário insira blocos e questões a um questionário	Inserção de blocos e questões ao questionário selecionado;
Permitir que o usuário exclua projetos, questionários, blocos e questões.	O sistema exclui permanentemente o projeto, questionário, bloco e questão.
Permitir que o usuário submeta um paciente a um questionário.	O sistema armazena as informações inseridas pelo paciente e salva no banco de dados para exportação.
Permitir que o usuário exporte dados em formato CSV e XLSX.	O sistema transforma os dados dos pacientes avaliados no formato solicitado pelo usuário e os disponibiliza para download.
Habilitar o cadastro de pacientes.	O sistema cadastra o paciente e deixa ele disponível para ser avaliado por um questionário.
Permitir que o usuário altere seus dados cadastrados durante a etapa de registro.	O sistema altera os dados cadastrais do paciente.

Figura 9- Diagrama de casos de uso.



### 3.3 BLOCO SERVIDOR

O servidor está utilizando o sistema operacional Linux Ubuntu 16.04 LTS - 64 bits, o motivo da escolha deste servidor foi a facilidade de instalação da stack LNPM além de ser totalmente grátis e open source. O servidor está recebendo requisições dos clientes através do porta padrão 80. Dentro deste bloco temos os seguintes componentes:

- **Nginx**

O Nginx é servidor web conhecido por sua velocidade de processamento, um servidor web por sua vez geralmente recebe solicitações HTTP de rede de entrada e envia respostas HTTP de saída (uma para cada solicitação processada), junto com conteúdo da web, por meio de conexões TCP/IP transparentes e/ou criptografadas (ver também: HTTPS) que são iniciadas por agentes de usuário cliente antes de enviar uma ou mais solicitações HTTP. O objetivo de um servidor web é armazenar e entregar conteúdos e/ou recursos da web. Exemplos de conteúdo da Web podem ser arquivos HTML, arquivos XHTML, arquivos de imagem, folhas de estilo, scripts, outros tipos de arquivos genéricos que podem ser baixados por clientes, etc.

- **PHP-FPM**

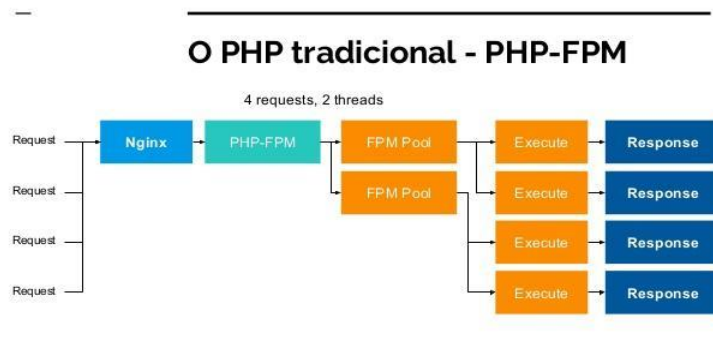
O PHP-FPM é um gerenciador de processos para gerenciar FastCGI SAPI (Server API) em PHP. É um serviço e não um módulo, este serviço executa de forma totalmente independente do servidor da Web em um processo separado, e qualquer servidor da Web

compatível com FastCGI (Fast Common Gateway Interface) oferece suporte a este serviço. Muito mais rápido do que outros métodos de processamento de scripts PHP e escalável, o que significa que pode construir clusters e estender a capacidade do PHP de receber solicitações.

Com o PHP-FPM, os elementos chamados e as instruções são armazenados na memória, ou seja, a implementação do cache no nível do servidor, e se a solicitação for executada novamente, pode ser utilizada diretamente. Portanto, a frequência de solicitação de arquivos PHP é bastante reduzida, o que significa que a carga da máquina (carga média) é reduzida e a disponibilidade de recursos é melhorada, para que outro trabalho possa ser executado.

PHP-FPM pode chamar "subprocessos" no mesmo "pool de trabalho", separando completamente o processamento de um script php do processamento de outro script php. Essas propriedades são configuradas no arquivo "pool", e o pool "www" é definido por padrão. Basicamente, o servidor web envia uma solicitação php para php-fpm, depois a envia para um de seus filhos e a executa até que a resposta seja entregue, veja a Figura 10.

Figura 10 - Árvore de requisição PHP-FPM.



Fonte: Gan, 2018.

- **SOFTWARE**

Esta é a ferramenta que foi criada e que está sendo descrita neste trabalho. Foi construída utilizando linguagem PHP com auxílio do framework Laravel citado na seção 2.5.1. Como citado anteriormente, o Laravel organiza a construção da ferramenta em 3 módulos: View, Model e o Controller. A view é onde está localizado nosso código HTML, Javascript e CSS, ou seja, é a parte responsável por gerar a interface gráfica. A



Model por sua vez corresponde às nossas classes já que utilizamos o paradigma orientado a objetos como descrito na seção 2.1, concentrando o modelo de dados e a estrutura geral dos componentes e como interagem entre si. Por fim temos o controller um conjunto de funções divididas em classes, mas que tem a responsabilidade de concentrar todas regras de negócios e ações presentes na ferramenta.

Um exemplo para entender melhor o papel de cada um deles pode ser o recebimento de uma requisição para visualizar todos usuários cadastrados no sistema, portanto o Controller requisita todos usuários existentes utilizando o modelo de dados do usuário a Model usuário. Após o recebimento, o Controller passa as informações que recebeu para a View que constrói a página HTML e devolve para que o controller possa enviar de volta a requisição. Portanto, o Controller age como se fosse um intermediador entre o Model e a View.

- **BANCO DE DADOS MYSQL**

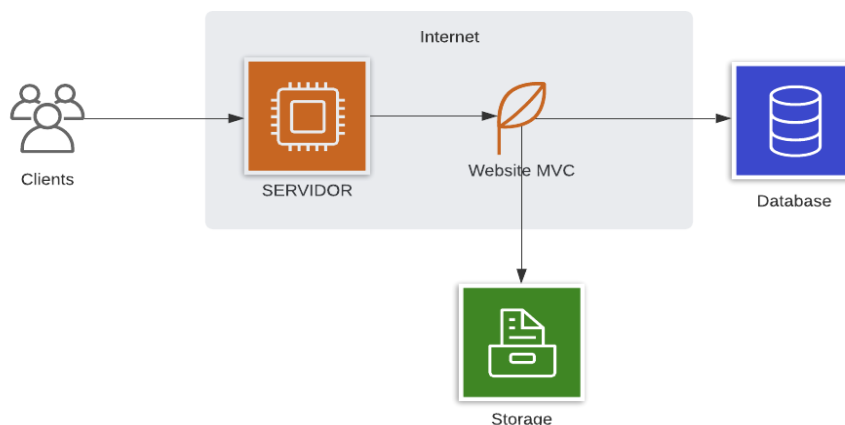
O banco de dados MySQL é da categoria relacional, foi encarregado de armazenar todas as informações relevantes para o software. Além disso, outro papel importante foi a recuperação de dados armazenados através de consultas na linguagem SQL criadas pelo Controller, para que então fosse possível processar estes dados e expor ao usuário final do sistema.

### **3.4 DIAGRAMA DE ARQUITETURA**

Um diagrama de arquitetura é um diagrama do sistema que é usado para abstrair o esboço geral do sistema de software e seus relacionamentos, restrições e limites entre os componentes. É uma importante ferramenta que provê uma visão geral da arquitetura física do sistema de software.

A Figura 11 representa esta visão geral da arquitetura física, no servidor temos todos os componentes citados anteriormente como o Nginx e PHP-FPM. O website MVC representa nosso software que está conectado ao banco de dados MySQL e a Storage para armazenamento de arquivos estáticos. Este diagrama foi utilizado para orientar e facilitar a construção da arquitetura do servidor.

Figura 11- Diagrama de Arquitetura.

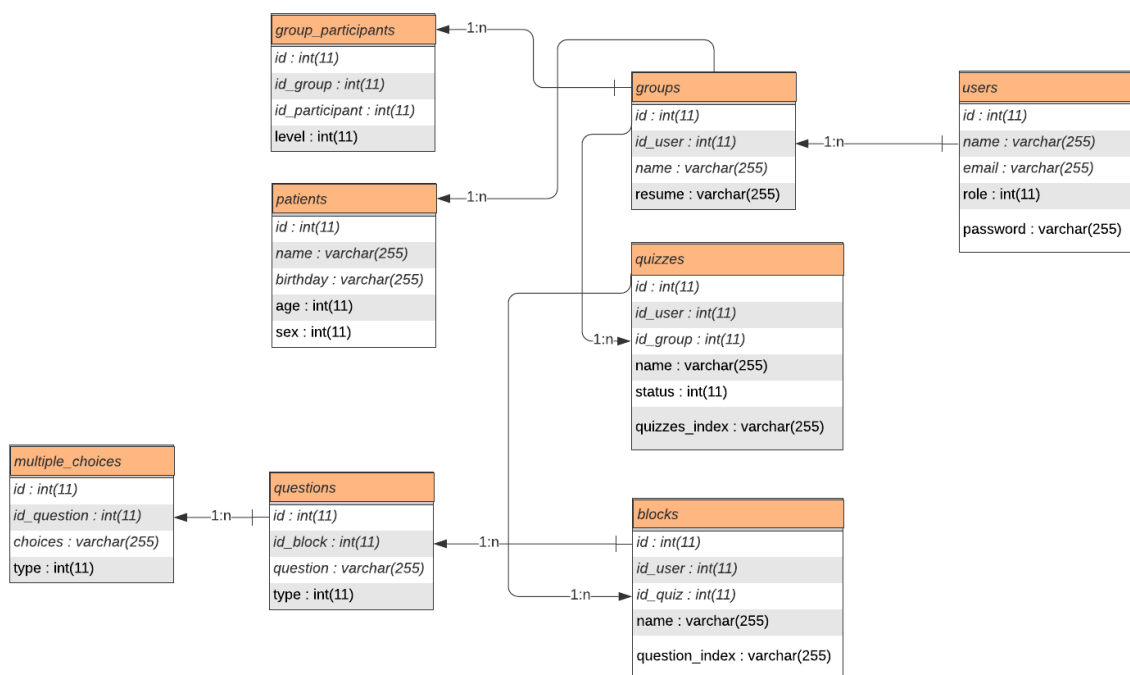


### 3.5 DIAGRAMA DE CLASSES

O diagrama de classes na linguagem unificada de modelagem (UML) é um tipo estático de diagrama de estrutura que descreve a estrutura do sistema mostrando as classes presentes no software, seus atributos, operações e os relacionamentos entre os objetos.

O diagrama da Figura 12 foi utilizado para orientar a ordem em que as classes deveriam ser criadas e a relação existente entre as classes, graças a este diagrama o tempo de desenvolvimento do software pode ser diminuído.

Figura 12 - Diagrama de Classes.



## 4 IMPLEMENTAÇÃO

Neste capítulo será apresentado a implementação da ferramenta proposta neste trabalho, primeiramente são elencados os requisitos de hardware necessários para a implementação e por fim é apresentado o protótipo desenvolvido.

### 4.1 REQUISITOS DE HARDWARE

Para desenvolvimento da ferramenta foram necessários apenas um notebook com processador Intel core i5 3.9Ghz, 4GB RAM e 500GB HD. Foram realizados alguns testes de desenvolvimento com computadores de configuração inferiores como Pentium 4 2.0GHz, 1GB RAM e 40GB HD e mesmo assim houve sucesso no desenvolvimento sem problemas de processamento ou falta de memória.

### 4.2 PROTÓTIPO

Um protótipo da nossa ferramenta foi construído respeitando os padrões de engenharia de software e que também utilizou todas ferramentas e tecnologias que foram descritas nos capítulos anteriores. A seguir são apresentadas algumas telas do software com uma descrição breve de cada uma delas.

Primeiramente desenvolvemos uma estrutura principal composta pelos projetos onde estão inseridos os questionários, que por sua vez, compostos por um ou mais blocos e dentro destes blocos finalmente temos as questões. Os tipos de questões existentes no software podem ser de 4 tipos: texto, numérico, sim/não e de alternativas. Todos os outros requisitos funcionais elencados que não faziam parte desta estrutura foram desenvolvidos posteriormente e vieram como funcionalidades não essenciais e adicionais ao software.

- **Página Inicial, Cadastro e Login de Usuários**

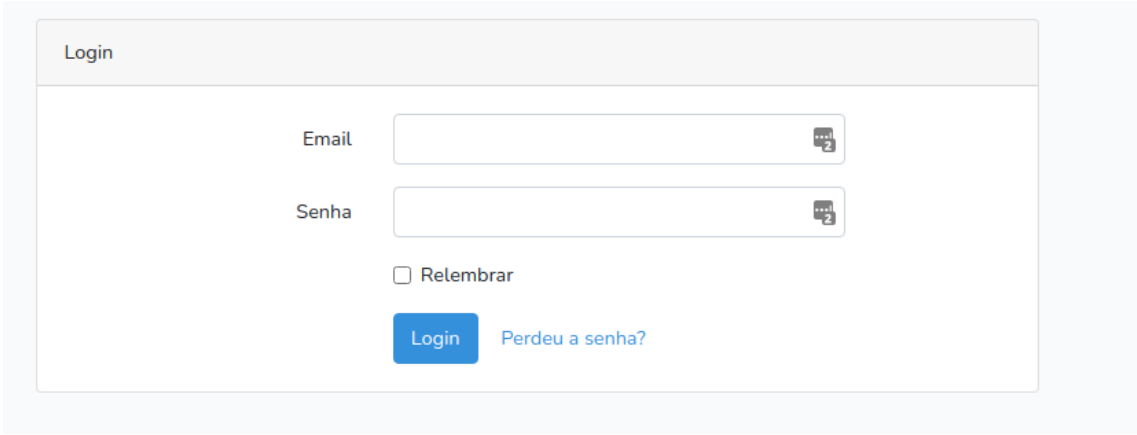
Na Figura 13 podemos conferir como ficou a página inicial com seu nome e uma breve explicação do que se trata a ferramenta, com dois botões no topo esquerdo da tela um para login e outro para se cadastrar.

Figura 13 - Tela inicial de boas vindas.



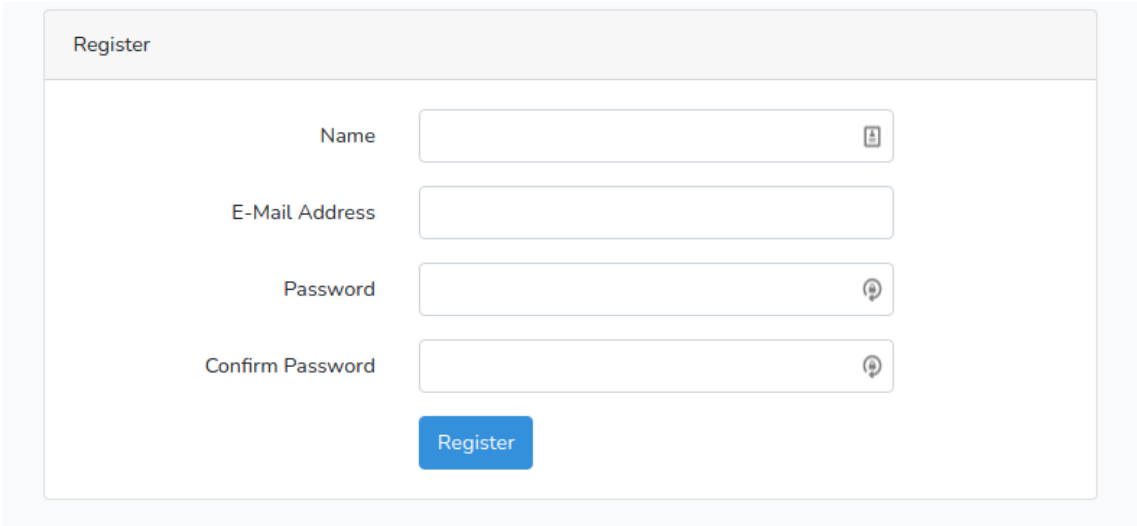
A Figura 14 e a Figura 15 mostram as telas de login e cadastro de usuários no sistema. O cadastro é simples em poucos minutos o usuário já tem total acesso aos recursos presentes na ferramenta.

Figura 14 - Tela de login do usuário.



The screenshot shows a login form titled "Login". It contains two input fields: "Email" and "Senha" (Password). Below the password field is a checkbox labeled "Relembrar" (Remember me). At the bottom, there is a blue "Login" button and a link labeled "Perdeu a senha?" (Forgot password?).

Figura 15 - Tela de registro do usuário.



The screenshot shows a registration form titled "Register". It contains four input fields: "Name", "E-Mail Address", "Password", and "Confirm Password". Below the "Confirm Password" field is a blue "Register" button.

- **Criar e Gerenciar Projetos**

Levando em conta a estrutura já citada anteriormente, o projeto é a camada mais superior que engloba todos os questionários. Na Figura 16 é possível ver a tela que contém o botão para a criação de um novo projeto, esta tela é apresentada assim que o usuário efetua o login no sistema.

Figura 16 - Tela inicial dos projetos.

## Projetos

Novo Projeto

	Nome	Descrição	Ações
<input type="checkbox"/>	Projeto 1 <ul style="list-style-type: none"><li>Questionário 1  </li><li>Questionário 2  </li></ul>	Descrição 1	
<input type="checkbox"/>	Projeto 2 <ul style="list-style-type: none"><li>Questionário 1  </li><li>Questionário 2  </li></ul>	Descrição 2	

Selecionar Todos Projetos

Com os selecionados:

Apagar

Neste contexto acionando o botão de “Novo Projeto”, surgirá para o usuário um formulário para que seja inserido o nome que deve ter o projeto e uma breve descrição dele, como mostra a Figura 17.

Figura 17 - Tela de criação de novo projeto.

Novo Projeto

Nome

Descrição

Fechar Criar

Ao criar um novo projeto o usuário poderá gerenciar este projeto, excluindo ou editando seu conteúdo como mostra a Figura 18. Nesta tela podemos ver que se existirem múltiplos projetos criados, o usuário poderá excluir todos de uma vez só, utilizando a opção de selecionar todos e remover.

Figura 18 - Tela visão geral dos projetos.

Projetos

Novo Projeto

	Nome	Descrição	Ações
<input type="checkbox"/>	Projeto 1 <ul style="list-style-type: none"><li>Questionário 1  </li><li>Questionário 2  </li></ul>	Descrição 1	
<input type="checkbox"/>	Projeto 2 <ul style="list-style-type: none"><li>Questionário 1  </li><li>Questionário 2  </li></ul>	Descrição 2	

Selecionar Todos Projetos

Com os selecionados:

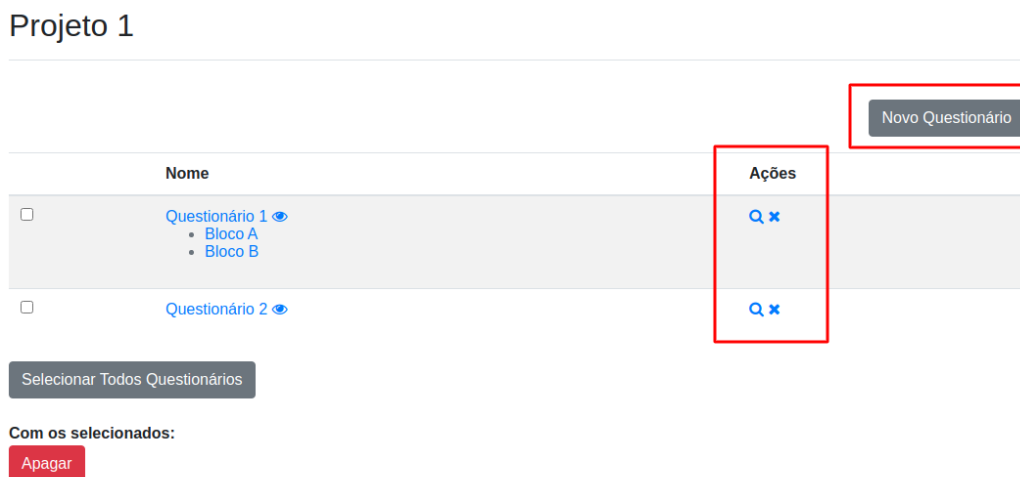
Apagar

- **Criar e gerenciar questionários**

Ao adentrar em um projeto específico, será carregada a tela de gerenciamento de projeto como mostra a Figura 19, sendo assim o usuário pode realizar diversas ações:

- Criar novo questionário;
- Pré visualizar um questionário;
- Excluir um questionário;

Figura 19 - Tela de gerenciamento de projeto.



Ao executar a ação de criar um novo questionário o formulário para criação é apresentado como na Figura 20.

Figura 20 - Formulário de novo questionário.

Novo Questionário

Nome

questionario exemplo

Sair Criar

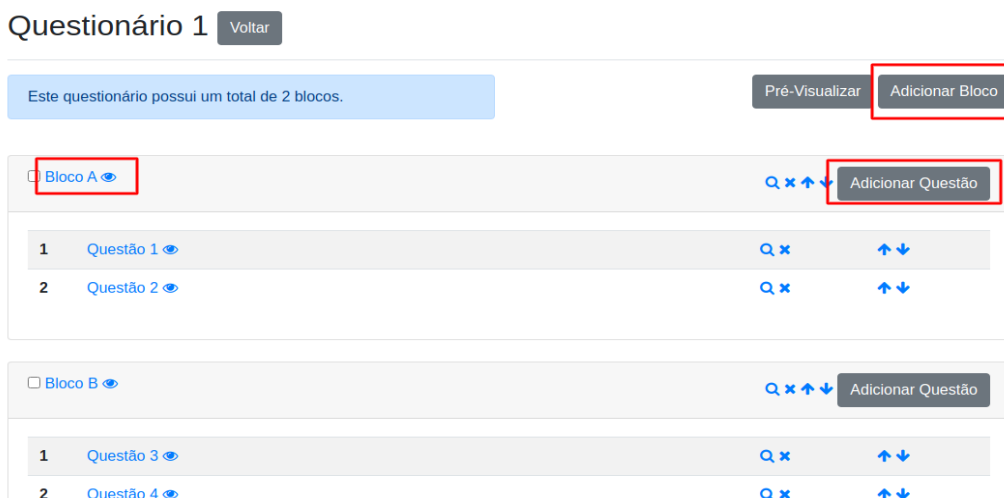
Os questionários são compostos por blocos que agrupam questões dirigidas ao mesmo escopo.

- **Criar e gerenciar blocos**

Na Figura 21 o usuário tem acesso a tela de gerenciamento de questionário, nela estão presentes ações como:

- Adicionar novo bloco;
- Pré visualizar questionário/blocos/questões;
- Adicionar nova questão;
- Mover questões/blocos de posição

Figura 21 - Tela de gerenciamento de questionário.

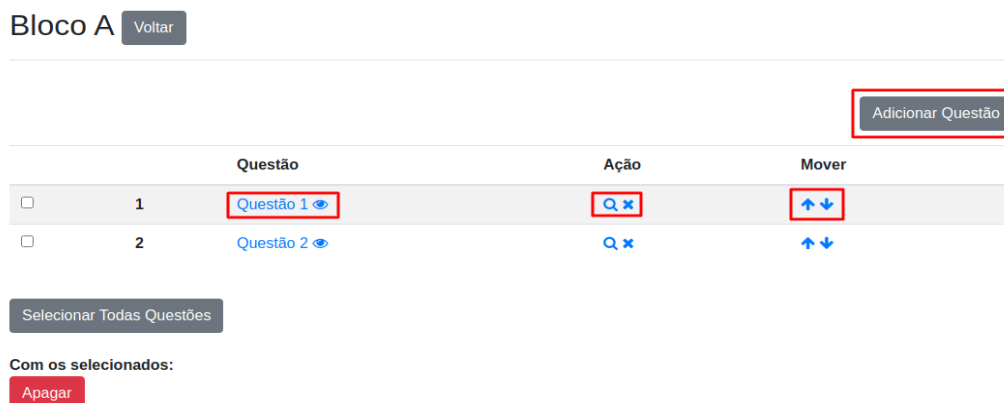


- **Criar e gerenciar questões**

Na tela de gerenciamento de blocos como mostra a Figura 22, o usuário tem diversas ações disponíveis:

- Visualizar questão;
- Adicionar questão;
- Pré visualizar questão;
- Mover questão de posição;
- Remover questão;

Figura 22 - Tela de gerenciamento de um bloco.



A Figura 23 nos mostra a tela para criação de novas questões, alguns exemplos de questões de formulários que poderiam ser criadas estão disponíveis a seguir. Todas as imagens foram baseadas no formulários utilizados na área da saúde como o da Classificação Internacional de Funcionalidade (CIF) [Pericia Medica, 2021] e The Survey of Well-being of Young Children (SWYC) [Tufts Children Hospital, 2021].

Figura 23 - Tela para adicionar nova questão.

- **Resposta curta:** Permite uma resposta de uma ou mais palavras;

Figura 24 – Questões retiradas da CIF, utiliza-se apenas letras e números.

BLOCO F: USO DE MEDICAMENTOS		
Agora vamos falar dos medicamentos que o senhor usa. Aqueles de uso regular, que o médico disse para tomar sempre/ todos os dias: Obs. Conferir essa informação no prontuário/ com a equipe.		
F1a	Qual o nome do remédio?	
F1b	Para qual doença ou problema de saúde o(a) Sr.(a) usa este remédio?	
F2a	Qual o nome do remédio?	
F2b	Para qual doença ou problema de saúde o(a) Sr.(a) usa este remédio?	
F3a	Qual o nome do remédio?	
F3b	Para qual doença ou problema de saúde o(a) Sr.(a) usa este remédio?	
F4a	Qual o nome do remédio?	
F4b	Para qual doença ou problema de saúde o(a) Sr.(a) usa este remédio?	
F5a	Qual o nome do remédio?	
F5b	Para qual doença ou problema de saúde o(a) Sr.(a) usa este remédio?	

- **Múltiplas Escolhas:** Permite a seleção de simples ou múltiplas respostas de uma lista pré definida;

Figura 25 – Questão retirada da SWYC, utiliza-se apenas tipo de opções já definidas.



<b>MARCOS DO DESENVOLVIMENTO (Developmental Milestones)</b>			
As perguntas a seguir são sobre o desenvolvimento de sua criança. Por favor, conte para nós o quanto sua criança faz cada uma destas coisas. Se sua criança já deixou de fazer alguma destas coisas, escolha a resposta que melhor descreve o quanto ele/ela costumava fazer isso antes. Por favor, verifique se respondeu TODAS as perguntas.			
	Ainda Não	Um pouco	Muito
Faz sons que mostram para você que ele ou ela está feliz ou chateado . . . . .	0	1	2
Parece feliz em ver você . . . . .	0	1	2
Segue com os olhos o movimento de um brinquedo . . . . .	0	1	2
Vira a cabeça para achar a pessoa que está falando . . . . .	0	1	2
Mantém a cabeça firme quando puxado para sentar . . . . .	0	1	2
Junta as mãos . . . . .	0	1	2
Ri . . . . .	0	1	2
Mantém a cabeça firme quando você o/a segura na posição sentada . . . . .	0	1	2
Faz sons como "ga", "ma" ou "ba" . . . . .	0	1	2
Olha quando você o/a chama pelo nome . . . . .	0	1	2

- **Numérico:** Permite uma resposta numérica;

Figura 26 – Questão retirada da CIF, utiliza-se apenas números.

D29	Há quanto tempo parou de fumar? <i>SOMENTE SE D27=3, E D28=1 OU D28=2 ou seja, se for ex-fumante.</i>	anos 0 Há menos de 1 ano
-----	---	-----------------------------

- **Sim/Não:** Uma questão simples de múltipla escolha com apenas duas opções "sim" ou "não".

Figura 27 - Questões retirados da CIF, opção de sim ou não.

J20	Tem dificuldades para tomar decisões?	0 Sim 1 Não
J21	Tem dificuldades no serviço (seu trabalho é penoso, cause-lhe sofrimento?)	0 Sim 1 Não
J22	É incapaz de desempenhar um papel útil em sua vida?	0 Sim 1 Não
J23	Tem perdido o interesse nas coisas?	0 Sim 1 Não
J24	Você se sente uma pessoa inútil, sem préstimo?	0 Sim 1 Não
J25	Tem tido ideia de acabar com a vida?	0 Sim 1 Não
J26	Sente-se cansado (a) o tempo todo?	0 Sim 1 Não
J27	Você se cansa com facilidade?	0 Sim 1 Não
J28	Tem sensações desagradáveis no estômago?	0 Sim 1 Não

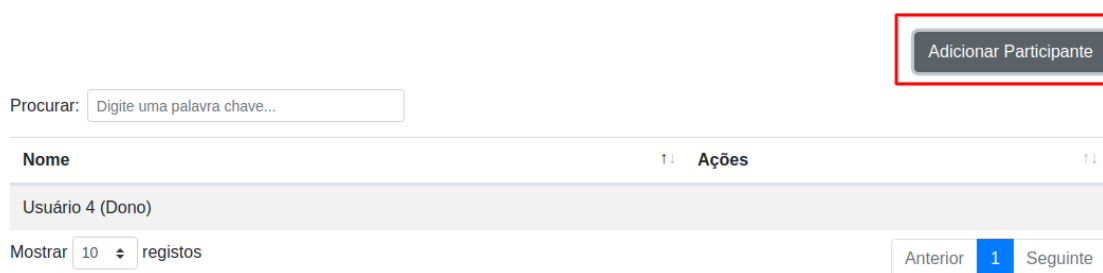
- **Adicionar e gerenciar participantes dos projetos**

Na Figura 28 podemos visualizar participantes de um projeto. Neste contexto o usuário poderá executar algumas funções como:

- Adicionar novo participante ao projeto;
- Pesquisar pelos usuários presentes na plataforma;

Figura 28 - Tela para visualizar participantes de um projeto.

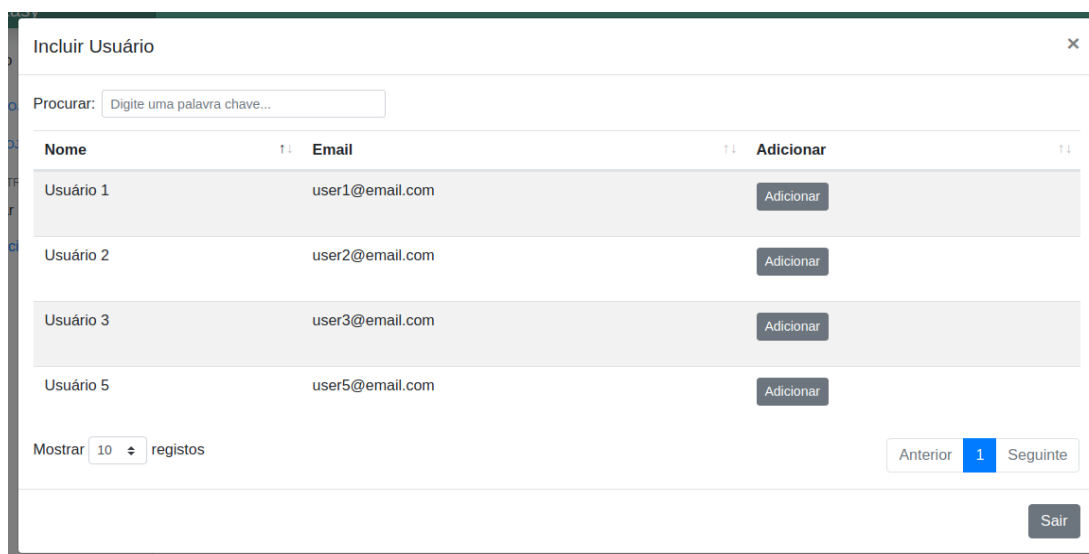
## Projeto 1



Fonte: O autor.

Ao adicionar um novo participante ao projeto, este novo participante poderá colaborar com o gerenciamento do conteúdo existente no projeto como questionários, blocos e questões.

Figura 29 - Tela para inserir um novo usuário a um projeto.



Fonte: O autor.

- **Cadastrar e submeter paciente a um questionário**

A Figura 30 representa a tela para qual o usuário é redirecionado para realizar a avaliação de um paciente, sendo assim o usuário poderá realizar as seguintes ações:

- Buscar por um paciente por nome/idade/cpf/rg;
- Submeter um paciente a um formulário previamente criado;

Figura 30 - Tela para pesquisar por pacientes cadastrados para avaliar.

## Buscar Paciente

Procurar:

Nome	Idade	CPF/RG	Avaliar
Paciente 1	18	45646191755	<input type="button" value="Avaliar"/>
Paciente 2	62	451545432154	<input type="button" value="Avaliar"/>
Paciente 3	32	4645123414535	<input type="button" value="Avaliar"/>

Mostrar  registros Anterior **1** Seguinte

Ao realizar a ação de avaliar um paciente, o entrevistador será direcionado para a tela da Figura 31, onde estão presentes os blocos e as questões que existem no questionário, sendo assim o entrevistador poderá fazer as perguntas e selecionar as respostas ou deixar que o próprio paciente responda ao questionário.

Figura 31 - Tela para submeter um paciente a um formulário.

## Avaliando Paciente

Informações do Paciente  
Nome: Paciente 1  
Idade: 18  
Sexo: 0

---

Bloco A

Questão 1

Selecione uma alternativa:

alternativa 1

alternativa 2

alternativa 3

---

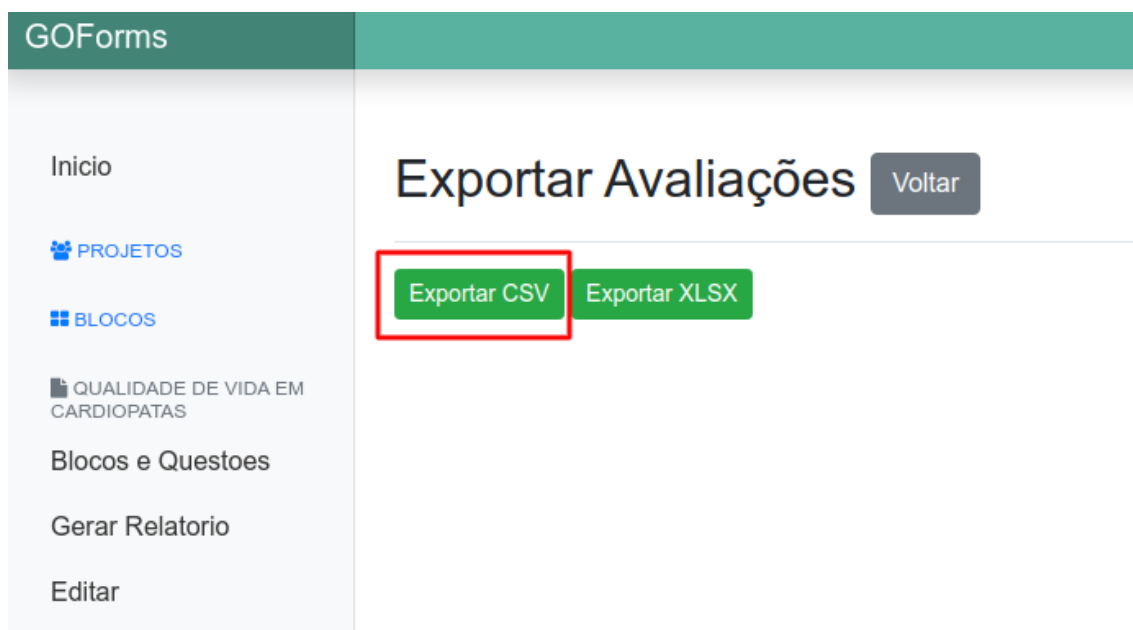
Bloco A

Questão 2

- **Exportar dados de uma avaliação**

Na figura 32 na tela de exportar dados, temos uma das funcionalidades mais importantes presentes no software. Nesta funcionalidade o usuário poderá optar por exportar os dados armazenados resultantes da avaliação de pacientes no formato CSV ou XLSX.

Figura 32 – Exportação de Dados.



Os dados exportados podem ser visualizados na Figura 33, o cabeçalho é formado pelo nome do paciente e as questões presentes no formulário.

Figura 33 – Visualização dos dados em CSV

	A	B	C	D	E
1	Paciente	Tem dormido mal?	Tem tido dificuldade em manter suas relações sociais habituais?	Tem tido dificuldade em relacionar-se com as pessoas?	Sente que não esta exercendo um papel util na vida?
2	Paciente 1	sim	sim	sim	sim
3	Paciente 2	sim	sim	sim	sim
4	Paciente 3	sim	sim	sim	sim
5					

### 4.3 VALIDAÇÃO

Com o objetivo de validar as funcionalidades e a interface do protótipo, foi realizado uma pesquisa qualitativa com 10 usuários. Todos os usuários receberam as mesma instruções de ações a serem realizadas, sendo elas:

- Cadastrar uma conta na ferramenta;
- Criar um projeto/grupo;
- Criar um formulário com no mínimo 2 blocos e 4 questões;
- Cadastrar um paciente;
- Avaliar um paciente com o formulário anteriormente criado;
- Exportar os dados em XLSX e visualizar com Excel;

Após todos usuários realizarem as ações, eles foram submetidos a uma série de perguntas relacionadas às funcionalidades e a interface. As respostas foram padronizadas

para melhor avaliação ao final do processo, foram categorizadas como bom, regular e ruim.

Tabela 3 – Avaliação do protótipo.

ASPECTO	BOM	REGULAR	RUIM
Como foi sua experiência com o cadastro de pacientes?	70%	20%	10%
Como foi sua experiência com o cadastro de usuário?	80%	20%	0%
Como foi sua experiência com o login de usuário?	100%	0%	0%
Como foi sua experiência com a criação de grupos/projetos?	90%	10%	0%
Como foi sua experiência com a criação de questionários/formulários?	90%	0%	10%
Como foi sua experiência com a criação de blocos?	80%	20%	0%
Como foi sua experiência com a criação de questionários?	80%	20%	0%
Como foi sua experiência com a avaliação de pacientes?	80%	20%	0%
Como foi sua experiência com a exportação de dados?	90%	0%	10%

Os resultados de cada avaliação receberão uma porcentagem resultante das 10 avaliações recebidas referentes aos 10 usuários. Um exemplo, 5 pessoas classificaram um aspecto como “bom”, sendo assim bom receberia 50%.

Para um grupo de 80% das classificações recebidas pelos usuários foram categorizadas como bom, isto mostra que as funcionalidades e interfaces presentes no protótipo atendem a expectativa da maioria.

## **5 CONCLUSÃO**

Este trabalho apresenta os resultados obtidos com um protótipo de uma ferramenta de criação de formulários e análise de dados para pesquisas de campo na área da saúde. O protótipo por sua vez foi construído utilizando uma arquitetura cliente/servidor com um framework Laravel (PHP) para o backend e o Bootstrap (HTML/CSS) para o frontend. Através destas tecnologias todos os requisitos funcionais e não funcionais levantados foram atendidos mostrando que este protótipo tem impacto direto nas pesquisas de campo dos profissionais de saúde interessados em utilizar a ferramenta desenvolvida.

As tecnologias escolhidas para o desenvolvimento se mostraram eficientes e capazes de atender ao escopo do projeto, porém devido a ampla gama de linguagens, frameworks e arquiteturas existentes para o desenvolvimento de software entende-se que as escolhas feitas não representam necessariamente as melhores. Porém dado a filosofia de código aberto tudo que foi desenvolvido se encontra disponível no Github (<https://github.com/alechaito/goforms>), ou seja, qualquer pessoa interessada em dar continuidade ao trabalho poderá incrementar com suas próprias alterações, adequar a ferramenta para sua necessidade própria ou até mesmo estudar como a ferramenta foi desenvolvida para a criação de algo similar utilizando outras tecnologias e ferramentas.

### **5.2 TRABALHOS FUTUROS**

Dado o caráter de protótipo do trabalho desenvolvido tem-se a possibilidade de melhorar a ferramenta e fornecer mais funcionalidades. Como trabalhos futuros são citadas algumas das possíveis melhorias a serem feitas a fim de aperfeiçoar este trabalho.

#### **5.2.1 Política de Controle de Acesso**

Como nosso sistema permite que um usuário convide outro para participar de seu projeto seria interessante que o criador do projeto pudesse implementar uma política de controle de acesso para os usuários que fossem participar do projeto. Esta é uma característica presente em muitos sistemas onde um usuário de nível superior fornece as permissões de outro usuário para que seja possível ter um controle mínimo do que tem ou não acesso dentro do seu perfil de usuário.

Um exemplo disso é o sistema usado pela Universidade Federal de Santa Catarina, o Moodle, uma plataforma de gestão de aprendizado online, nele existem diversos papéis como: professor, aluno, editor, participante. Estes papéis, por sua vez, fornecem uma permissão específica, controlando o que cada usuário poderá fazer dentro do escopo delimitado. No presente trabalho diferentes papéis como administrador, editor e leitor poderiam ser inseridos incrementando as funcionalidades já presentes no sistema.

### **5.2.2 Digitalização de Formulários**

Em muitas pesquisas de campo os entrevistadores já possuem seus modelos de formulários prontos às vezes em csv, pdf, json, xml entre outros, nestes casos atualmente o entrevistador terá que construir novamente seu formulário desde o zero em nossa ferramenta. Uma funcionalidade que agilizaria este processo seria utilizar algum mecanismo que fosse capaz de ler estes formulários em diversos formatos e transformá-los no formato do sistema, com essa facilidade a adoção da ferramenta seria muito mais ampla, sendo uma barreira a menos, para que novos pesquisadores se beneficiem do sistema.

### **5.2.3 Banco de Questões**

Um mecanismo importante para ajudar na criação dos formulários pelos pesquisadores seria a existência de um banco de questões, que funcionaria da seguinte maneira, sempre que uma nova questão fosse inserida no sistema pelo usuário ela seria copiada para o banco de questões permitindo então que questões fossem reaproveitadas em outros formulários e além disso que pudessem ser compartilhadas entre outros coordenadores, isso por si só seria uma economia muito grande de tempo envolvido no processo de concepção de formulários.

### **5.2.3 Formularios Dinámicos**

Para criar pesquisas mais eficientes e com mais assertividade, uma funcionalidade importante seria transformar os formulários estáticos e dinâmicos, para permitir que em tempo real conforme a resposta do entrevistado o formulário possa se adaptar e seguir por caminhos diferentes.

## REFERÊNCIAS

APOSTOLICO, M. R.; EGRY, E. Y. Uso da internet na coleta de dados primários na pesquisa em Enfermagem. **Revista Brasileira de Enfermagem**, 66, n. 6, p. 949-955, 2013.

ATKINSON, H. L.; NIXON-CAVE, K. A Tool for Clinical Reasoning and Reflection Using the International Classification of Functioning, Disability and Health (ICF) Framework and Patient Management Model. **Physical Therapy**, 91, n. 3, p. 416-430, 2011.

BARROS, R. S.; FERREIRA, S. M.; HEXSEL, R. A., 2002, **Desenvolvimento de solução única de software para o Sistema Cartão Nacional de Saúde.**

BERNDTSSON, M. Analyzing course configurations for teaching object-oriented modeling and design. **IEEE**, 2005.

BOLLIS, E. R. e-Phenology Collector: uma solução robusta para coleta de dados de campo usando plataformas móveis. 2016.

BONI, G. N.; MORGADO, E. M.; GOULART, L. J.; TOKUNAGA, M. K. *et al.*, 2006, **Desenvolvimento de um Software de Coleta de Dados para Pesquisas de Campo Através de Dispositivos Móveis.**

CALLIYERIS, V.; DE ELUA ROBLE, G. L.; COSTA, C.; DA SILVA SOUZA, W. Pesquisa via Internet como técnica de coleta de dados: um balanço da literatura e os principais desafios para sua utilização. **Revista Brasileira de Marketing**, 14, n. 4, p. 479-491, 2015.

CESAR, C. L.; FIGUEIREDO, G. M.; WESTPHAL, M. F.; CARDOSO, M. R. A. *et al.* Referred morbidity and the utilization of health services in urban areas of south-eastern Brazil: methodology. **Revista de saude publica**, 30, n. 2, p. 153-160, 1996.



CHEN, X. Restful API Architecture Based on Laravel Framework **Journal of Physics**, 910, 2017.

CHI, X.; LIU, B.; NIU, Q.; WU, Q., 2012, **Web load balance and cache optimization design based nginx under high-concurrency environment**. IEEE. 1029-1032.

COSTA, C. G. A. d. Desenvolvimento e avaliação tecnológica de um sistema de prontuário eletrônico do paciente, baseado nos paradigmas da World Wide Web e da engenharia de software. 2001.

DAMASCENO, L. M. d. S.; SILVA, P. G. d.; RAMOS, A. S. M.; CORTEZ, A. E. G. *et al.* Potencialidade e limitações da coleta de dados através de pesquisa online. **XVII Seminário em Administração (SEMEAD)**, 7, 2014.

DE BARROS LIMA, A. M. E.; RODRIGUES, C. A. Q.; HAIKAL, D. S.; SILVEIRA, M. F. *et al.* Desenvolvimento de um programa de computador para levantamentos epidemiológicos sobre condições de saúde bucal. **Unimontes Científica**, 14, n. 1, p. 30-42, 2012.

DE LEMOS, M. F.; OLIVEIRA, P. C.; RUELA, L. C.; DA SILVA SANTOS, M. *et al.* Aplicabilidade da arquitetura MVC em uma aplicação web (WebApps). **RE3C-Revista Eletrônica Científica de Ciência da Computação**, 8, n. 1, 2013.

DE OLIVEIRA SOUZA, M.; MATIOSKI, M. E.; NEVES, L. A. P. ANÁLISE DE DESEMPENHO DOS BANCOS DE DADOS MYSQL, POSTGRESQL E FIREBIRD: UM ESTUDO DE CASO1. **Gestão**.

DOS SANTOS, B. G.; MAMANI-ALIAGA, A. H.; SÁNCHEZ, J. V.; MENDONCA, M. *et al.*, 2013, **Projeto maritaca: Arquitetura e infraestrutura para coleta móvel de dados usando smartphones**. 1076-1083.

DOS SANTOS SOARES, M. Comparação entre metodologias Ágeis e tradicionais para o desenvolvimento de software. **INFOCOMP Journal of Computer Science**, 3, n. 2, p. 8-13, 2004.

FONTANELLA, B. J. B.; CAMPOS, C. J. G.; TURATO, E. R. Coleta de dados na pesquisa clínico-qualitativa: uso de entrevistas não-dirigidas de questões abertas por profissionais da saúde. **Revista Latino-Americana de Enfermagem**, 14, n. 5, 2006.

FREITAS, H.; JANISSEK-MUNIZ, R.; BAULAC, Y. Pesquisa via Web. : Reinventando o papel e a idéia de pesquisa. Canoas: Sphinx 2006.

FREITAS, H.; JANISSEK, R.; MOSCAROLA, J., 2004, **Dinâmica do processo de coleta e análise de dados via web**. 1-13.

FREITAS, H. M. R. d.; JANISSEK-MUNIZ, R.; MOSCAROLA, J. Uso da Internet no processo de pesquisa e análise de dados. **Associação Nacional de Empresas de Pesquisa (2004: São Paulo).[Anais..][São Paulo: ANEP, 2004]**, 2004.

GAN, Y.; DELIMITROU, C. The architectural implications of cloud microservices. **IEEE Computer Architecture Letters**, 17, n. 2, p. 155-158, 2018.

HOURANI, H. A Code Complexity Model of Object Oriented Programming (OOP). **IEEE**, 2019.

IEEE. 24765-2017 - ISO/IEC/IEEE International Standard - Systems and software engineering--Vocabulary. **IEEE**, 28 Aug. 2017 2017.

JENAL, S.; ÉVORA, Y. D. M. Revisão de literatura: implantação de prontuário eletrônico do paciente. **Journal of health informatics**, 4, n. 4, 2012.

KORTH, H. F.; LEVY, E.; SILBERSCHATZ, A. **A formal approach to recovery by compensating transactions**. University of Texas at Austin, Department of Computer Sciences, 1990.

LEFF, A.; RAYFIELD, J. T., 2001, **Web-application development using the model/view/controller design pattern**. IEEE. 118-127.

LIMA, D. F. B.; BRAGA, A. L. d. S.; FERNANDES, J. L.; BRANDÃO, E. d. S. Sistema de informação em saúde: concepções e perspectivas dos enfermeiros sobre o prontuário eletrônico do paciente. **Revista de Enfermagem Referência**, n. 5, p. 113-119, 2011.

LIU, C.; WANG, K., 2012, **An Online Examination System Based on UML Modeling and MVC Design Pattern**. IEEE. 815-817.

LIZAMA, O.; KINDLEY, G.; MORALES, J. I. J.; GONZALES, A. Redes de Computadores: Arquitectura Cliente-Servidor. **Universidad Tecnica Federico Santa Maria**, p. 1-8, 2016.

MUFID, M. R.; BASOFI, A.; AL RASYID, M. U. H.; ROCHIMANSYAH, I. F., 2019, **Design an MVC Model using Python for Flask Framework Development**. IEEE. 214-219.

NASCIMENTO, G. V. **Um modelo de referência para o desenvolvimento ágil de software**. 2008. -, Universidade de São Paulo.

NEGREIROS, I. S. SISTEMA DIGITAL DE AVALIAÇÃO ATRAVÉS CLASSIFICAÇÃO INTERNACIONAL DE FUNCIONALIDADE, INCAPACIDADE E SAÚDE-CIF. : Universidade Federal do Ceará, Fortaleza, Ceará Fortaleza 2010.

PASCHOAL, L. N.; SOUZA, P. N.; BUCHALLA, C. M.; BRITO, C. M. M. *et al.* Identification of relevant categories for inpatient physical therapy care using the International Classification of Functioning, Disability and Health: a Brazilian survey. **Braz J Phys Ther**, 23, n. 3, p. 212-220, May - Jun 2019.

RABELO, N. S. Utilização de Bancos de Dados e Arquitetura Cliente/Servidor em Ambientes Comerciais. **Uberlândia: Centro Universitário do Triângulo–Unit**, 1998.

RANGEL, J. A. Informatização de um instrumento de coleta de dados para pacientes em unidade ambulatorial. 2015.

REZENDE, J.; SILVA, B.; DA CONCEIÇÃO, A., 2010, **Plataforma para desenvolvimento simples e flexível de questionários para Coleta Móvel de Dados (CMD)**.

RIBERTO, M. Core sets da classificação internacional de funcionalidade, incapacidade e saúde. **Revista Brasileira de Enfermagem**, 64, n. 5, p. 938-946, 2011.

RIQUELME, S. E. L. Gestión Documental: buena práctica para reducir el consumo de papel en apoyo al desarrollo sostenible. **GECONTEC: Revista Internacional de Gestión del Conocimiento y la Tecnología**, 7, n. 1, p. 78-92, 2019.

ROBERTO, P.; PINHEIRO, F.; SILVA, C.; LAPA, T. *et al.* formulário interAtivo de AvAliAção pré-AnestésICA desenvolvido Em Portugal. **Revista da Sociedade Portuguesa de Anestesiologia**, 23, n. 1, p. 20-24, 2014.

RODRIGUEZ-MORALES, A. J.; GALLEGRO, V.; ESCALERA-ANTEZANA, J. P.; MÉNDEZ, C. A. *et al.* COVID-19 in Latin America: The implications of the first confirmed case in Brazil. **Travel medicine and infectious disease**, 2020.

SENDIANG, M.; KASENDA, S.; POLII, A.; PUTUNG, Y. R., 2018, **Optimizing Laravel Authentication Process**. IEEE. 247-251.

SINGH, A.; CHAWLA, P.; SINGH, K.; SINGH, A. K., 2018, **Formulating an MVC Framework for Web Development in JAVA**. IEEE. 926-929.

SUDARSHAN, S.; SILBERSCHATZ, A.; KORTH, F. H. Sistemas de Banco de Dados. 3ª. Edição. : Editora Makron Books 1999.

SUN, Q. Toward Understanding Students' Learning Performance in an Object-Oriented Programming Course: The Perspective of Program Quality. **IEEE**, 2020.

TABORDA, M.; RANGEL, M. Pesquisa Quali-quantitativa On-line: Relato de uma experiência em desenvolvimento no campo da saúde. **CIAIQ2015**, 1, 2015.

WACHELKE, J.; NATIVIDADE, J.; DE ANDRADE, A.; WOLTER, R. *et al.* Caracterização e avaliação de um procedimento de coleta de dados online (CORP). **Avaliação Psicológica**, 13, n. 1, p. 143-146, 2014.

WAKU, G. M.; BOLLIS, E. R.; RUBIRA, C. M.; TORRES, R. d. S., 2015, **A robust software product line architecture for data collection in android platform**. **IEEE**. 31-39.

WANG, C. L. K. An Online Examination System Based on UML Modeling and MVC Design Pattern. **IEEE**, 2012.

YADAV, N.; RAJPOOT, D. S.; DHAKAD, S. K., 2019, **LARAVEL: A PHP Framework for E-Commerce Website**. **IEEE**. 503-508.

YU, X.; YI, C., 2010, **Design and Implementation of the Website Based on PHP & MYSQL**. **IEEE**. 1-4.

ZHANG, D.; WEI, Z.; YANG, Y., 2013, **Research on lightweight MVC framework based on spring MVC and mybatis**. **IEEE**. 350-353.

Tufts Children Hospital – SWYC. <https://www.tuftschildrenshospital.org/the-survey-of-wellbeing-of-young-children/overview>. Acesso: 03/05/2021.

Pericia Medica - CIF. [http://www.periciamedicadf.com.br/cif2/cif\\_portugues.pdf](http://www.periciamedicadf.com.br/cif2/cif_portugues.pdf). Acesso: 03/05/2021.

Uso de Internet, Televisao e Celular no Brasil. Educa IBGE, Brasilia, 4, janeiro de 2019. Disponivel em: <https://educa.ibge.gov.br/jovens/materias-especiais/20787-uso-de-internet-televisao-e-celular-no-brasil.html>. Acesso em: 19, maio de 2021.

Google Forms. Sao Paulo, 4, janeiro de 2021. Disponivel em: <https://www.google.com/forms/about/>. Acesso em: 19, maio de 2021.

LimeSurvey. Sao Paulo, 20, janeiro de 2021. Disponivel em: <https://www.limesurvey.org/pt/>. Acesso em: 19, maio de 2021.

## **ANEXO A – FORMULÁRIO TRADICIONAL**



# SWYC™ : 2 meses

1 mês, 0 dias até 3 meses, 31 dias  
1 month, 0 days to 3 months, 31 days  
V1.07, 4/1/17

Nome da Criança (Child name)
Data de Nascimento (DOB):
Idade Gestacional (gestational age)
Data de Hoje (date of administration):
IG Corrigida (ID#):

## MARCOS DO DESENVOLVIMENTO (Developmental Milestones)

As perguntas a seguir são sobre o desenvolvimento de sua criança. Por favor, conte para nós o quanto sua criança faz cada uma destas coisas. Se sua criança já deixou de fazer alguma destas coisas, escolha a resposta que melhor descreve o quanto ele/ela costumava fazer isso antes. Por favor, verifique se respondeu TODAS as perguntas.

	Ainda Não	Um pouco	Muito
Faz sons que mostram para você que ele ou ela está feliz ou chateado . . . . .	<input type="radio"/> 0	<input type="radio"/> 1	<input type="radio"/> 2
Parece feliz em ver você . . . . .	<input type="radio"/> 0	<input type="radio"/> 1	<input type="radio"/> 2
Segue com os olhos o movimento de um brinquedo . . . . .	<input type="radio"/> 0	<input type="radio"/> 1	<input type="radio"/> 2
Vira a cabeça para achar a pessoa que está falando . . . . .	<input type="radio"/> 0	<input type="radio"/> 1	<input type="radio"/> 2
Mantém a cabeça firme quando puxado para sentar . . . . .	<input type="radio"/> 0	<input type="radio"/> 1	<input type="radio"/> 2
Junta as mãos . . . . .	<input type="radio"/> 0	<input type="radio"/> 1	<input type="radio"/> 2
Ri . . . . .	<input type="radio"/> 0	<input type="radio"/> 1	<input type="radio"/> 2
Mantém a cabeça firme quando você o/a segura na posição sentada . . . . .	<input type="radio"/> 0	<input type="radio"/> 1	<input type="radio"/> 2
Faz sons como "ga", "ma" ou "ba" . . . . .	<input type="radio"/> 0	<input type="radio"/> 1	<input type="radio"/> 2
Olha quando você o/a chama pelo nome . . . . .	<input type="radio"/> 0	<input type="radio"/> 1	<input type="radio"/> 2

## LISTA DE SINTOMAS DO BEBÊ (BPSC)

Estas perguntas são sobre o comportamento da sua criança. Pense sobre o que você esperaria de outras crianças da mesma idade e nos conte o quanto cada pergunta descreve o comportamento de sua criança.

	Não	Um pouco	Muito
Sua criança fica incomodada com novas pessoas? . . . . .	<input type="radio"/> 0	<input type="radio"/> 1	<input type="radio"/> 2
Sua criança fica incomodada em lugares novos? . . . . .	<input type="radio"/> 0	<input type="radio"/> 1	<input type="radio"/> 2
É difícil para sua criança lidar com mudanças na rotina? . . . . .	<input type="radio"/> 0	<input type="radio"/> 1	<input type="radio"/> 2
Sua criança fica incomodada de ser carregada por outras pessoas? . . . . .	<input type="radio"/> 0	<input type="radio"/> 1	<input type="radio"/> 2
Sua criança chora muito? . . . . .	<input type="radio"/> 0	<input type="radio"/> 1	<input type="radio"/> 2
É difícil para sua criança se acalmar sozinha? . . . . .	<input type="radio"/> 0	<input type="radio"/> 1	<input type="radio"/> 2
Sua criança fica irritada facilmente? . . . . .	<input type="radio"/> 0	<input type="radio"/> 1	<input type="radio"/> 2
Sua criança continua chorando, mesmo quando você a pega no colo e tenta acalmá-la? . . . . .	<input type="radio"/> 0	<input type="radio"/> 1	<input type="radio"/> 2
É difícil manter sua criança nas rotinas do dia a dia? . . . . .	<input type="radio"/> 0	<input type="radio"/> 1	<input type="radio"/> 2
Sua criança tem dificuldades para pegar no sono? . . . . .	<input type="radio"/> 0	<input type="radio"/> 1	<input type="radio"/> 2
É difícil para você dormir o suficiente por causa da sua criança? . . . . .	<input type="radio"/> 0	<input type="radio"/> 1	<input type="radio"/> 2
Sua criança tem dificuldades para manter o sono? . . . . .	<input type="radio"/> 0	<input type="radio"/> 1	<input type="radio"/> 2

## PREOCUPAÇÕES DOS PAIS (Parent Concerns)

Com relação ao comportamento atual da sua criança:	Não	Um pouco	Muito
Você tem alguma preocupação com o aprendizado ou com o desenvolvimento dela?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Você tem alguma preocupação com o comportamento de sua criança?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>



## PERGUNTAS SOBRE A FAMÍLIA (Family Questions)

		Sim	Não						
1	Alguem que mora com sua criança fuma cigarro?	<input type="radio"/>	<input type="radio"/>						
2	No último ano, alguma vez você consumiu mais álcool ou drogas do que pretendia?	<input type="radio"/>	<input type="radio"/>						
3	No último ano, você sentiu vontade ou necessidade de diminuir o seu consumo de álcool ou drogas?	<input type="radio"/>	<input type="radio"/>						
4	Alguma vez, o uso de álcool ou drogas por algum membro da família trouxe consequências negativas para sua criança?	<input type="radio"/>	<input type="radio"/>						
		Nunca aconteceu	Aconteceu algumas vezes	Frequentemente acontece					
5	Nos últimos 12 meses, ficamos preocupados se nossa comida poderia acabar antes que pudessemos comprar mais.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>					
		Não tem conflito	Com algum conflito	Muito conflito	Não se aplica				
6	Em geral, como você descreveria seu relacionamento com seu/sua marido/companheiro(a)?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>				
		Sem dificuldade de	Com alguma dificuldade de	Com muita dificuldade	Não se aplica				
7	Você e seu/sua marido/companheiro(a) resolvem seus desentendimentos	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>				
8	Na última semana, quantos dias você ou outro membro da família leu para sua criança?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

## Mudanças emocionais com um novo bebê (Emotional Changes with a New Baby\*\*)

Visto que acabou de ter seu bebê, gostaríamos de saber como você se sente. Não apenas como se sente hoje, mas como se sentiu NOS ÚLTIMOS 7 DIAS. Por favor, marque a resposta que corresponde com o que você sente.

Nos últimos 7 dias...				
<b>1 Tenho sido capaz de me rir e ver o lado divertido das coisas</b>				
<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Tanto como dantes	Menos do que antes	Muito menos do que antes	Nunca	
<b>2 Tenho tido esperança no futuro</b>				
<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Tanta como sempre tive	Bastante menos do que costumava ter	Muito menos do que costumava ter	Quase nenhuma	
<b>3* Tenho-me culpado sem necessidade quando as coisas correm mal</b>				
<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Sim, a maioria das vezes	Sim, algumas vezes	Raramente	Não, nunca	
<b>4 Tenho estado ansiosa ou preocupada sem motivo</b>				
<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Não, nunca	Quase nunca	Sim, por vezes	Sim, muitas vezes	
<b>5* Tenho-me sentido com medo, ou muito assustada, sem grande motivo</b>				
<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Sim, muitas vezes	Sim, por vezes	Não, raramente	Não, nunca	
<b>6* Tenho sentido que são coisas demais para mim</b>				
<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Sim, a maioria das vezes não tenho conseguido resolvê-las	Sim, por vezes não tenho conseguido resolvê-las como dantes	Não, a maioria das vezes resolvo-as facilmente	Não, resolvo-as tão bem como dantes	
<b>7* Tenho-me sentido tão infeliz que durmo mal</b>				
<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Sim, quase sempre	Sim, por vezes	Raramente	Não, nunca	
<b>8* Tenho-me sentido triste ou muito infeliz</b>				
<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Sim, quase sempre	Sim, muitas vezes	Raramente	Não, nunca	
<b>9* Tenho-me sentido tão infeliz que choro</b>				
<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Sim, quase sempre	Sim, muitas vezes	Só às vezes	Não, nunca	
<b>10* Tive ideias de fazer mal a mim mesma</b>				
<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Sim, muitas vezes	Por vezes	Muito raramente	Nunca	

\*\*© 1987 The Royal College of Psychiatrists. Cox, J.L., Holden, J.M., & Sagovsky, R. (1987). Detection of postnatal depression. Development of the 10-item Edinburgh Postnatal Depression Scale. *British Journal of Psychiatry*, 150, 782-786. Written permission must be obtained from the Royal College of Psychiatrists for copying and distribution to others or for republication (in print, online or by any other medium).