

**UNIVERSIDADE DO EXTREMO SUL CATARINENSE - UNESC  
CURSO DE CIÊNCIA DA COMPUTAÇÃO**

**CARLOS RAFAEL MAGALHÃES FERNANDES**

**INTEGRAÇÃO DE DADOS PARA COMPARTILHAMENTO DE INFORMAÇÕES  
RELACIONADAS AOS SERVIÇOS DISPONÍVEIS EM UNIDADES DE SAÚDE**

**CRICIÚMA**

**2019**

**CARLOS RAFAEL MAGALHÃES FERNANDES**

**INTEGRAÇÃO DE DADOS PARA COMPARTILHAMENTO DE INFORMAÇÕES  
RELACIONADAS AOS SERVIÇOS DISPONÍVEIS EM UNIDADES DE SAÚDE**

Trabalho de Conclusão de Curso, apresentado para obtenção do grau de Bacharel no curso de Ciência da Computação da Universidade do Extremo Sul Catarinense, UNESC.

Orientador: Prof. Esp. Gilberto Vieira da Silva

**CRICIÚMA**

**2019**

**CARLOS RAFAEL MAGALHÃES FERNANDES**

**INTEGRAÇÃO DE DADOS PARA COMPARTILHAMENTO DE INFORMAÇÕES  
RELACIONADAS AOS SERVIÇOS DISPONÍVEIS EM UNIDADES DE SAÚDE**

Trabalho de Conclusão de Curso aprovado pela Banca Examinadora para obtenção do Grau de Bacharel, no Curso de Ciência da Computação da Universidade do Extremo Sul Catarinense, UNESC, com Linha de Pesquisa em Desenvolvimento Web

Criciúma, 02 de dezembro de 2019.

**BANCA EXAMINADORA**



Prof. Gilberto Vieira da Silva - Especialista - (UNESC) - Orientador



Profa. Ana Claudia Garcia Barbosa - Mestra - (UNESC)



Prof. Matheus Leandro Ferreira- Especialista - (UNESC)

**Aos meus queridos pais, meus irmãos,  
meus amigos que sempre me apoiaram.**

## **AGRADECIMENTOS**

Primeiramente a Deus, que me abençoou durante todo o tempo que estive cursando a faculdade, a minha família que sempre me ajudou e incentivou nos momentos difíceis, meus amigos pelo apoio, as empresas EloVerde e Duapi Sistemas, pela oportunidade de fazer parte do seu time.

Agradeço também ao meu orientador Gilberto Vieira da Silva por me orientar e direcionar os caminhos certos para que pudesse desenvolver o trabalho de conclusão de curso, agradeço a todos professores por terem passado seus conhecimentos, e também a todos os colegas que sempre estiveram juntos comigo apoiando, agradeço também ao Vicente, Isabel, Cardoso, Augusto, Nicolau, Deolinda, Osvaldo, Iracema, Rosa e Rafael que sempre estiveram comigo.

**“Comunicação não é o que você fala, mas o que outro compreende do que foi dito.”**

**Claudia Belucci**

## RESUMO

A aplicação *web* é uma das tecnologias que está em constante evolução. Diariamente surgem padrões e tecnologias que auxiliam no desenvolvimento das mesmas. Navegadores *webs* como *Safari*, *Google Chrome* e *Mozilla Firefox* estão cada vez melhores, possibilitando a sua usabilidade, ampliando recursos e funcionalidades que facilitam seu uso, quer seja no seio dos desenvolvedores *webs*, quer seja dos usuários comuns. Essas tecnologias buscam trazer melhor qualidade de vida e mais acesso aos médicos que precisam obter os históricos de um paciente que foi atendido em outra unidade de saúde. Baseado nesse pressuposto, este trabalho teve como objetivo desenvolver um protótipo de aplicação *web* para compartilhamento de informações entre unidades de saúde que apresentam dificuldades em comunicar-se. Com o intuito máximo de auxiliar pacientes a localizarem quais unidades de saúde estão disponibilizando os exames que os mesmos precisam fazer. Para isso, foi realizada uma fundamentação teórica referente às técnicas de compartilhamento de informações entre as unidades de saúde e metodologias de comunicação em aplicações *web*. Com os conhecimentos obtidos a partir da pesquisa, foi possível elaborar um protótipo de aplicação *web*, utilizando a ferramenta *Laravel* para criação de uma API, a biblioteca de análise, geração e envio de mensagem *HL7 v2.x* baseado em *PHP*, inspirado no pacote *Perl Net-HL7*, para análise e criação da mensagem no padrão *hl7 v2.x*.

**Palavras-chave:** *HL7 v2.x*; Aplicações *web*; *Perl Net-HL7* e Comunicação;

## ABSTRACT

Web application is one of the constantly evolving technologies. Patterns and technologies that help their development appear daily. Web browsers such as Safari, Google Chrome, and Mozilla Firefox are getting better and better, enabling usability, extending features and functionality that make it easy to use, whether it's within web developers or ordinary users. These technologies seek to bring better quality of life and more access to doctors who need to obtain the histories of a patient who was seen at another health center. Based on this assumption, this paper aimed to develop a web application prototype for information sharing among health facilities that have difficulties in communicating. In order to help patients, locate which health facilities are providing the tests they need to do. For this, a theoretical foundation was made regarding the techniques of information sharing between health centers and communication methodologies in web applications. With the knowledge gained from the research, it was possible to develop a web application prototype, using the Laravel tool to create an API, the PHP-based HL7 v2.x analysis, message generation and analysis library, inspired by the Perl package. Net-HL7, for analysis and message creation in the hl7 v2.x standard.

**Keywords:** HL7 v2.x; Web applications; Perl Net-HL7 and Communication;

## LISTA DE ILUSTRAÇÕES

Figura 1 - Interoperabilidade entre sistemas hospitalares .....	11
Figura 2 - Estrutura conceitual de um documento CCR.....	17
Figura 3 - As sete camadas do modelo ISO/OSI .....	18
Figura 4 - Esquema de comunicação HL7 v2 .....	22
Figura 5 - Componentes de uma mensagem HL7 v2.x.....	22
Figura 6 - Definição de aplicação web .....	24
Figura 7 Arquitetura de Uma Aplicação Web.....	26
Figura 8 - Definição de interface <i>preparation application</i> .....	28
Figura 9 Representação de dados XML.....	28
Figura 10 Representação de dados JSON .....	29
Figura 11 Representação de YAML .....	29
Figura 12 - Monolítica e Microservice Banco de dados. ....	31
Figura 13 – Mensagem hl7.....	37
Figura 14 – Representação de análise de mensagem.....	43
Figura 15 – Representação da criação de mensagem. ....	44
Figura 16 – Método para salva a mensagem no padrão hl7 .....	45
Figura 17 – Atendimento do paciente. ....	46
Figura 18 – Atendimento do paciente. ....	47
Figura 19 – informações dos exames realizado nas unidades de saúde. ....	48
Figura 20 – Informações do paciente.....	49

## LISTA DE TABELAS

Tabela 1 – Campos do segmento MSH. ....	38
Tabela 2– Campos do segmento PID. ....	39
Tabela 3 – Campos do segmento EVN.....	40
Tabela 4 – Campos do segmento NTE.....	41
Tabela 5 – Campos do segmento OBR. ....	41

## LISTA DE ABREVIATURAS E SIGLAS

AAFP	American Academy of Family Physicians
AAFP	Academia Americana de Médicos de Família
ABRAMGE	Associação Brasileira de Medicina de Grupo
AMB	Associação Médica Brasileira
ANS	Agência Nacional de Saúde
ANSI	<i>National Standards Institute</i>
API	Interface de Programação de Aplicações
ASP	Active Server Page
ASTM	American Society for Testing and Materials
CCR	Continuity of Care Record
CDA	Clinical Document Architecture
CID	Código Internacional de Doenças
DATASUS	Departamento de Informação e Informática do Sistema Único de Saúde
EDIFACT	Electronic Data Interchange for Administration, Commerce and Transport
EHR	<i>Electronic Health Records</i>
EMR	Electronic Medical Records
HIMSS	Health Information and Management Systems Society
HL7	Health Level 7
HTTP	HyperText Transfer Protocol
ISSN	International Standard Serial Number
ISO	International Organization for Standardization
NGS	Nível de Garantia de Segurança
PEP	Prontuário Eletrônico do Paciente
RES	Registro Eletrônico em Saúde
RIM	Reference Information Model
SNOMED	Systematized Nomenclature of Medicine
SUS	Sistema Único de Saúde
TCP/IP	Transmission Control Protocol / Internet Protocol
TIC	Tecnologia da Informação e Comunicação
TISS	Troca de Informações na Saúde Suplementar

UX	User experience
XML	Extensible Markup Language
YAML	Ain't Markup Language
REST	Transferência de Estado Representacional

## SUMÁRIO

<b>1 INTRODUÇÃO</b> .....	<b>6</b>
1.1 OBJETIVO GERAL.....	7
1.2 OBJETIVOS ESPECÍFICOS .....	7
1.3 JUSTIFICATIVA .....	8
1.4 ESTRUTURA DO TRABALHO.....	9
<b>2 INTEROPERABILIDADE ENTRE SISTEMAS HOSPITALARES</b> .....	<b>10</b>
2.1 PADRÕES EM UNIDADES DE SAÚDE.....	12
2.2 TECNOLOGIA E PADRÕES .....	14
<b>2.2.1 Padrão CCR</b> .....	<b>14</b>
<b>2.2.2 Aplicações do CCR</b> .....	<b>16</b>
<b>2.2.3 HL7</b> .....	<b>17</b>
<b>2.2.3 Funcionamento padrão HL7</b> .....	<b>19</b>
2.2.3.1 Tipo de mensagem e formato .....	19
2.2.3.2 Eventos .....	19
2.2.3.3 Confirmação de erro.....	20
<b>2.2.4 Arquitetura física do sistema</b> .....	<b>20</b>
2.2.4.1 Versões publicadas .....	20
2.2.4.1.1 <i>HL7 Versão 1.</i> .....	20
2.2.4.1.2 <i>Versões 2.x.</i> .....	21
2.2.4.1.3 <i>Versão 3</i> .....	23
2.2.4.1.4 <i>Tipos de dados utilizado pela versão 3</i> .....	23
<b>3 TECNOLOGIA PARA DESENVOLVIMENTO WEB</b> .....	<b>24</b>
3.1 APLICAÇÕES WEB .....	24
<b>3.1.2 Arquitetura das aplicações web</b> .....	<b>25</b>
3.1.2.1 Server side .....	26
3.1.2.2 Client side.....	27
3.2. APPLICATION PROGRAMMING INTERFACE (API).....	27
<b>3.2.1 Microservice</b> .....	<b>30</b>
<b>4 TRABALHOS CORRELATOS</b> .....	<b>32</b>
4.1 ANÁLISE COMPARATIVA DE PADRÕES DE INTEROPERABILIDADE EM SISTEMAS HOSPITALARES.....	32

4.2 MODELOS PARA INTEROPERABILIDADE DE SISTEMAS HOSPITALARES UTILIZANDO PADRÃO HL7 .....	33
4.3 ESTUDO DA PADRONIZAÇÃO VISANDO À INTEROPERABILIDADE: O CASO DAS ORGANIZAÇÕES DE SAÚDE NA CIDADE DE MARÍLIA - SÃO PAULO.....	34
4.4 PLATAFORMA PARA M-HEALTH BASEADA NO PADRÃO OPENEHR, EM COMUNICAÇÕES M2M E EM COMPUTAÇÃO EM NUVEM.....	35
<b>5 FERRAMENTA DE INTEGRAÇÃO DE DADOS PARA COMPARTILHAMENTO DE INFORMAÇÕES RELACIONADAS AOS SERVIÇOS DISPONÍVEIS EM UNIDADES DE SAÚDE .....</b>	<b>36</b>
5.1 METODOLOGIA.....	36
<b>5.1.1 Biblioteca Php para Analisar, Gerar e Enviar Mensagens hl7 v2.x. ....</b>	<b>36</b>
5.1.1.1 Demonstração de análise e criação da mensagem hl7 .....	38
5.1.1.2 Construção da API .....	44
<b>5.1.2 Desenvolvimento do Protótipo .....</b>	<b>46</b>
5.1.2.1 Etapa de atendimento de um paciente.....	47
5.1.2.1 Etapa de consulta de um paciente .....	48
5.2 RESULTADOS OBTIDOS.....	49
<b>6 CONCLUSÃO .....</b>	<b>51</b>
<b>REFERÊNCIAS.....</b>	<b>54</b>

## 1 INTRODUÇÃO

O problema de comunicação sempre acarreta um custo, que gera a baixa produtividade e a perda de tempo, em função disso, as tecnologias de informação existentes colaboram para a complexidade da comunicação, para tal, necessita-se de maior integração entre as tecnologias que abrangem os sistemas de informações (ANDRADE, 2014).

Segundo Moreira (2018), nas unidades de saúde, principalmente pública, há ainda inúmeras dificuldades no sistema de informação, conseqüentemente acabam afetando diretamente os técnicos de saúde, que buscam por informações de quais medicamentos, equipamentos e serviços estão disponíveis na sua unidade de saúde ou em outras para encaminhamento de paciente.

O surgimento, bem como início da história da informática, é visto como um passo fundamental para se alcançar a era do conhecimento. A priori, foi possível guardar, armazenar grandes massas de informações, em seguida, permitiu-se analisar os mais variados dados para gerar diferentes tipos de conhecimentos.

Sendo possível armazenar dados, é relevante que se possa compartilhá-los com outros sistemas, neste sentido, diferentes mecanismos foram desenvolvidos para garantir a comunicação entre os vários tipos de sistemas, envolvendo dispositivos, sistemas operacionais, aplicativos, bases de dados e deste modo, foram projetados padrões como o HL7, CCR a nível internacional.

O HL7 tem o objetivo de definir normas para a transmissão de dados, não restringe o sistema operacional, nem a linguagem de programação a ser utilizada para a troca de informação, permitindo ainda que o implementador defina suas próprias mensagens, o que mais uma vez, torna-o flexível.

CCR foi desenvolvido para resolver um problema clínico muito real: os médicos necessitam de uma forma simples e rápida de coletar um núcleo importante de informações de atendimento ao paciente em um formato estruturado, legível e facilmente transferível.

Com o desenvolvimento da tecnologia e a crescente incorporação da informática nos ambientes hospitalares têm colaborado para a agilidade na produção, organização e compartilhamento das informações.

Essa comunicação torna-se muito mais fácil quando as informações são enviadas em tempo real. Segundo Young (1982), define um sistema de tempo real

como qualquer ação ou sistema de processamento de informações que dispõe atender a motivação de acesso externa dentro de um período finito e especificado. A informação em tempo real assegura rapidez e eficiência nas tomadas de soluções, criando assim muitas vantagens que resultarão na solução de um determinado problema aumentando o valor das organizações, desta forma, a interoperabilidade entre os sistemas hospitalares tornou-se o principal eixo de estudos para a disseminação de informações médicas (BATISTA, 2016).

Este trabalho tem como objetivo utilizar tecnologias de integração de dados, para a elaboração de um protótipo de aplicação *web*, voltado para o apoio das unidades de saúde que têm dificuldade em comunicar-se entre si, com o intuito de facilitar o acesso à informação como medicamentos, consultas e situações dos equipamentos.

## 1.1 OBJETIVO GERAL

Desenvolver um protótipo de aplicação para integração de dados nas unidades de saúde que apresentam dificuldades em comunicar-se entre si.

## 1.2 OBJETIVOS ESPECÍFICOS

Os objetivos específicos desta pesquisa consistem em:

- a) pesquisar formas de interação com as unidades de saúde e usuário da aplicação;
- b) entender o funcionamento de envio de mensagem em tempo real;
- c) compreender o funcionamento da comunicação entre sistemas heterogêneos;
- d) desenvolver um protótipo de aplicação na *web* para integração de dados nas unidades de saúde que apresentam dificuldades em comunicar-se entre si;
- e) avaliar a eficiência do protótipo.

### 1.3 JUSTIFICATIVA

A integração de dados funciona como um facilitador, fornecendo informações que ajudam na escolha da melhor alternativa a ser seguida. Na saúde os benefícios serão muitos, os técnicos economizaram tempo, esforço, mantendo-se atualizados em uma única fonte de informação.

A integração de dados nas unidades de saúde é importante, pois pode proporcionar agilidade e auxílio na tomada de decisões, aumentando a produtividade e dando maior segurança nos dados, fazendo assim uma gestão inteligente do estoque dos hospitais (SANNA, 2014).

As unidades de saúde precisam de integração de dados porque a mesma simplifica consideravelmente as rotinas operacionais. O cotidiano dos técnicos de saúde e dos pacientes já é preenchido com uma infinidade de tarefas e buscar informações entre múltiplas fontes fará com que eles fiquem menos produtivos na execução de cada uma delas (SILVA, 2017). Ao manter todos os dados essenciais em um só lugar, os técnicos de saúde pouparão tempo e esforços nas buscas de uma unidade de pronto atendimento. Com um sistema de integração de dados, todas essas informações que são armazenadas, utilizando *softwares* diferentes e muitas vezes incompatíveis, podem ser encontradas em um só lugar usando a mesma plataforma para consolidar as informações. Os pacientes ganham praticidade no acesso às informações e as unidades de saúde estariam mais organizadas (PIERRO, 2011).

Quando se sabe exatamente o que acontece nas unidades de saúde, é muito mais fácil tomar a decisão certa. A informação em tempo real pode dar todas as entradas necessárias para que os técnicos de saúde e os pacientes decidam conscientemente sobre o melhor rumo a tomar referente ao atendimento do hospital (GUIMARÃES, 2000).

A interoperabilidade de sistemas na área da saúde contribui para o desenvolvimento da qualidade e produtividade da assistência médica, com o objetivo de permitir a realização da pesquisa de serviço, entre as unidades de pronto atendimento. Por esse motivo, o controle da informação permitirá que os técnicos de saúde realizem as atividades com efetividade e eficiência, integrando assim as informações paralelas a comunicação, de modo a coordenar as atividades entre os diferentes membros das equipes (INE, 2014).

Está ficando cada vez mais claro que o intercâmbio coerente de informações na saúde vem sendo tomada como primordial para o sucesso de um tratamento, e até mesmo para salvar vida (CM TECNOLOGIA, 2016).

Desse modo os hospitais passarão a ter acesso aos históricos de algumas informações disponibilizadas pelas as unidades de saúde em que deve ser interpretada como forma de enriquecer propostas de aquisição, eficiência, revolução e qualidade para assegurar melhores serviços e a universalidade do acesso de dados que os pacientes terão ao seu dispor.

#### 1.4 ESTRUTURA DO TRABALHO

Este trabalho está dividido em uma estrutura com 5 capítulos. Sendo que o primeiro capítulo apresenta uma breve descrição sobre o tema do trabalho, contendo a introdução, o objetivo geral e os específicos e ainda a justificativa do mesmo.

O segundo capítulo trata dos assuntos relacionados a interoperabilidade entre sistemas hospitalares, incluindo os padrões usados nas unidades de saúde, tecnologias para o funcionamento do padrão HL7 e arquitetura física do sistema.

O terceiro capítulo aborda o levantamento teórico referente a todas as tecnologias aplicadas no desenvolvimento do trabalho. Dois grandes grupos se destacam nesse capítulo: Aplicações Web, com sua estrutura, arquitetura e funcionamento.

No quarto capítulo são apresentados trabalhos correlatos ao estudo desenvolvido.

O quinto capítulo traz a estrutura do trabalho desenvolvido, onde aborda-se a metodologia aplicada para o desenvolvimento do protótipo e onde são avaliados os resultados obtidos com a elaboração do mesmo.

## 2 INTEROPERABILIDADE ENTRE SISTEMAS HOSPITALARES

A Interoperabilidade refere-se ao processo de comunicação de dois ou mais sistemas que podem ser computadores, redes, *software* e outros componentes de tecnologia de informação, sem a geração de uma dependência tecnológica entre os mesmos (BRASIL, 2017).

Não é apenas a integração entre sistemas e redes de computadores, não considera simplesmente definição de tecnologia, mas a soma de todos esses fatores, considerando, também, a existência de um legado de sistemas de plataforma, de *hardware* e *software* instalados (LOPES, 2018).

Segundo ISNN (2018) o setor da saúde no Brasil guarda grandes desafios, entre eles o próprio uso da tecnologia da informação, nesse contexto a interoperabilidade na saúde está deixando de ser uma tendência para ser uma exigência do mercado, visto que, além de diminuir custo e subtrair deficiências através da otimização de tarefas, oferece maior administração e agilidade de fluxo de trabalho e otimização do tempo garantindo ao paciente uma melhor experiência de entendimento.

Muitas organizações de saúde ainda não adotaram a interoperabilidade de sistemas por falta de conhecimento do processo e investimento (SCIARRA; RONDINA, 2018).

Na saúde, a interoperabilidade é a capacidade de diferentes sistemas de informação de se comunicarem, trocar dados, e usar as informações utilizando padrões e modelos que permitem que os dados sejam compartilhados entre profissionais de saúde, pacientes, farmácias, hospitais, laboratórios, etc., independentemente do aplicativo a ser utilizado como pode ser visto na figura 1 (EHCOS, 2018).

Figura 1 - Interoperabilidade entre sistemas hospitalares



Fonte: Ehcos (2018).

Com o desenvolvimento da tecnologia e a crescente informatização em ambientes clínicos e hospitalares, no início dos anos noventa, começou-se a cogitar o compartilhamento das informações nestes sistemas. Conseqüentemente, a interoperabilidade entre os sistemas hospitalares passou a ser o grande foco de pesquisa para disseminação de informações médicas. Contudo, quando se imagina em interoperabilidade na qual sistemas compartilham informações com outros sistemas não se pode deixar de pensar em padrões, sendo fundamental que os dados sejam estruturados e não dúbios. Além disso, é importante ter dados com uma semântica que possa ser compreendida pelos outros sistemas (COSTA, 2001).

Diversos hospitais e clínicas, de vários países, disponibilizam API's para o consumo de informações clínicas. O desafio da integração das unidades de saúde está no fato de que cada uma delas trabalha de forma autônoma e como ilha isolada de informação, qualquer uma destas organizações prestam diferentes serviços de assistência médica utilizando formas e processos semelhantes, porém, com diferenças. A maioria destas organizações possuem os recursos necessários e capacidade para personalizar o software que utiliza para adequá-lo a seus processos, ao invés de ajustar o processo, de acordo com o software (INTEROPERA, 2018, tradução nossa).

Por exemplo, considere a entrada de um paciente em dois hospitais diferentes sendo eles A e B.

A informação usada para distinguir os pacientes é basicamente a mesma, mas é possível que o prontuário eletrônico do hospital B armazene dados diferentemente do hospital A (exemplo: o Hospital B não utiliza o campo 'nome' dos

pacientes). Agora, considere a possibilidade de um surto de Ebola, semelhante ao que ocorreu em 2014.

O hospital A já possui em seu sistema um elemento de dados definido para casos deste tipo, denominado de 'surto', conseqüentemente, eles não precisam mudar nada em seu sistema de registro eletrônico em saúde.

O hospital B precisa de alguma maneira manter o controle e a vigilância sobre os casos confirmados de seus pacientes. Eles decidem utilizar o elemento 'nome' (visto que este é um elemento de dados não utilizado nesta implantação) do paciente como elemento identificador dos casos em suspeita e dos confirmados, assim, digitam a palavra "Ebola" neste campo.

Para cada um destes cenários as API's que os sistemas utilizam para se comunicar continuarão funcionando, porém a 'semântica' ou o 'significado' destes campos foi comprometido, pois seu contexto foi modificado. O caso é que quando utilizamos API's, não basta apenas ter a API. Você precisa entender o que a API fornece para você e como isto se relaciona com as informações com as quais você trabalha (PARKER, 2016).

Existindo uma quantidade significativa de padrões de interoperabilidade de sistemas constatou-se a necessidade de efetuar uma análise desses padrões.

## 2.1 PADRÕES EM UNIDADES DE SAÚDE

A necessidade de padronizar a troca eletrônica de informações em saúde trazem inúmeros benefícios e deve-se a vários fatores (VAN BEMMEL,1997; WATERS,1999):

- a) facilitar a busca e a comunicação de informações;
- b) melhorar a qualidade da assistência à saúde;
- c) viabilizar o uso de sistemas de apoio à decisão e alerta, que são indispensáveis para a interoperabilidade entre os sistemas;
- d) aprimorar a comunicação entre os profissionais do setor;
- e) devido a pontos importantes para a área de saúde como estatísticas, prestação de contas (faturamento), indexação de documentos e pesquisa clínicas.

Segundo Costa (2001), os padrões realizados na área da saúde podem ser apresentados como:

- a) identificação: para pacientes *Social Security Number* - nos Estados Unidos, Cartão Nacional de Saúde no Brasil, médicos (Número no Conselho Regional de Medicina);
- b) comunicação: padrão para mensagens entre sistemas (HL7, X12, EDIFACT *Electronic Data Interchange For Administration, Commerce and Transport*, XML);
- c) conteúdo e estrutura: padronização do Registro Clínico do *DATASUS* (Departamento de Informação e Informática do Sistema Único de Saúde), *ABRAMGE* (Associação Brasileira de Medicina de Grupo);
- d) representação de dados clínicos (códigos): Código Internacional de Doenças (CID), *Systematized Nomenclature of Medicine* SNOMED, Associação Médica Brasileira (AMB);
- e) confidencialidade, segurança e autenticação.

Por contornar dados que devem ser preservados sob sigilo, a implantação da troca de informações em saúde pressupõe a observância de normas já existentes originais de órgãos competentes para tais fins (ANS,2017).

A medida do processo de segurança apresentados pela Agência Nacional de Saúde (ANS) para a implantação da Troca de Informação em Saúde Suplementar (TISS), estão as normas técnicas estabelecidas Resolução CFM n.º 1639, de 10 de julho de 2002, e nas resoluções da ANS (RN n.º 21, de 12 de dezembro de 2002, e na RDC n.º 64, de 10 de abril de 2001). Tais medidas proporcionam as garantias administrativas, técnicas e físicas de proteção ao acesso à informação trocada.

Recomenda-se, para o alcance das finalidades de segurança e privacidade, a análise dos requisitos do Nível de Garantia de Segurança (NGS-1), descritos no "Manual de Requisitos de Segurança, Conteúdo e Funcionalidades para Sistemas de Registro Eletrônico em Saúde" (RES), em conformidade com a norma NBR ISO/IEC 17799 - Código de Prática para a Gestão da Segurança da Informação.

## 2.2 TECNOLOGIA E PADRÕES

A fim de produzir modelos de interoperabilidade de sistemas hospitalares, é fundamental compreender o que é o padrão HL7, seu princípio, suas versões publicadas, suas potencialidades para que possa ser feita a construção de sua gramática. Já para a implementação de uma arquitetura cliente servidor é necessário conhecer quais recursos e funcionalidades que a linguagem selecionada proporciona. São as tecnologias selecionadas para concepção deste modelo de interoperabilidade que serão abordadas a seguir.

### **Padrão CCR**

O padrão *Continuity-of-Care Record* (CCR) foi desenvolvido para resolver um problema clínico muito real: os médicos necessitam de uma forma simples e rápida de coletar um núcleo importante de informações de atendimento ao paciente em um formato estruturado, legível e facilmente transferível. Para esse fim, diferentes organizações médicas de alto nível, envolvendo a Academia Americana de Pediatria, a Sociedade Médica de Massachusetts, a Academia Americana de Médicos de Família (AAFP), a Sociedade de Sistemas e Gestão da Informação em Saúde (HIMSS) e a *American Health Care Association*, uniram forças com a *ASTM International* para desenvolver o que hoje é célebre como CCR. De acordo com a AAFP, a CCR é uma forma de criar documentos que contêm o núcleo mais importante e oportuno das informações de saúde sobre um paciente e enviá-los de uma unidade para outra (DAVID, 2010).

O CCR é baseado na comunicação eletrônica, além de ter como objetivo permitir que os médicos, ao utilizarem o Prontuário Eletrônico do Paciente (PEP), podendo ter acesso rápido às informações dos mesmos, facilitando, um diagnóstico prévio, sem necessariamente visualizar todos os dados, seu intuito é melhorar a qualidade dos cuidados de saúde e reduzir os erros médicos, disponibilizando informações atualizadas.

De acordo com Fábio Costa (2012), o CCR não tem como objetivo criar um prontuário completo, de acesso universal. Os principais objetivos são melhorar os cuidados médicos, reduzir o número de erros e assegurar um padrão mínimo de segurança na transferência das informações médicas. Além de criar documentos

que contenham as mais relevantes e atualizadas informações sobre o estado clínico do paciente.

Com esse padrão, é possível criar documentos contendo informações de saúde que podem ser enviados de um dispositivo para outro. Por ser expresso em XML, o CCR pode ser criado, lido e interpretado por qualquer software *Electronic Health Records* (EHR) ou *Electronic Medical Records* (EMR). Contém *tags* específicas para descrever diagnósticos e listas de problemas, medicamentos, alergias e plano de cuidados.

De acordo com Guerra (2010), o registro possui um conjunto dos mais relevantes dados administrativos, demográficos e informações clínicas sobre a saúde do paciente, contendo as informações médicas necessárias para a continuidade de um tratamento em outros centros médicos.

O conjunto de dados do núcleo CCR é composto pelos seguintes elementos:

- a) CCR informações de identificação: referindo-se, praticamente, a datas de consultas e fins de encaminhamento;
- b) informações de identificação do paciente: usadas para criar identificadores únicos do paciente sem a necessidade de um sistema centralizado;
- c) paciente seguros / informações financeiras: informações básicas sobre a elegibilidade para os benefícios;
- d) diretivas antecipadas: instruções de reanimação, como pode ser encontrado em testamentos e procurações de saúde;
- e) estado de saúde do paciente: informações detalhadas sobre a condição de saúde do paciente;
- f) cuidados com documentação: detalhes sobre o encontro paciente-médico;
- g) cuidados recomendação plano: testes, procedimentos ou regimes de atendimento planejadas ou programadas;
- h) lista de profissionais de saúde: informações sobre os profissionais da área que realizaram atendimento a esse paciente.

Segundo ASTM (2013) o CCR facilita na transferência de informações entre unidades de saúde melhorando a qualidade do atendimento. CCR ajuda a

evitar medicações erradas, fornecendo uma base para evitar interações medicamentosas e prescrições duplicadas, reduzindo os testes laboratoriais redundantes. Além disso, futuras aplicações podem “capacitar os pacientes a participarem da gestão de sua própria saúde”. Por exemplo, os pacientes poderiam usar o CCR em seu computador doméstico para rever medicações e a dosagem do mesmo sincronizando sua programação com seu smartphone (SANDISK CORPORATION, 2011).

O CCR deve ser preparado em um formato eletrônico estruturado que é intercambiável entre os Registros dos sistemas Eletrônicos de Saúde (RES.). Para garantir a interoperabilidade dos CCRs eletrônicos, esta norma especifica que a codificação XML é necessária quando o CCR é criado em um formato eletrônico estruturado. A codificação XML proporciona flexibilidade que permitirá aos usuários se prepararem, transmitirem e exibirem o CCR de várias maneiras.

### **Aplicações do CCR**

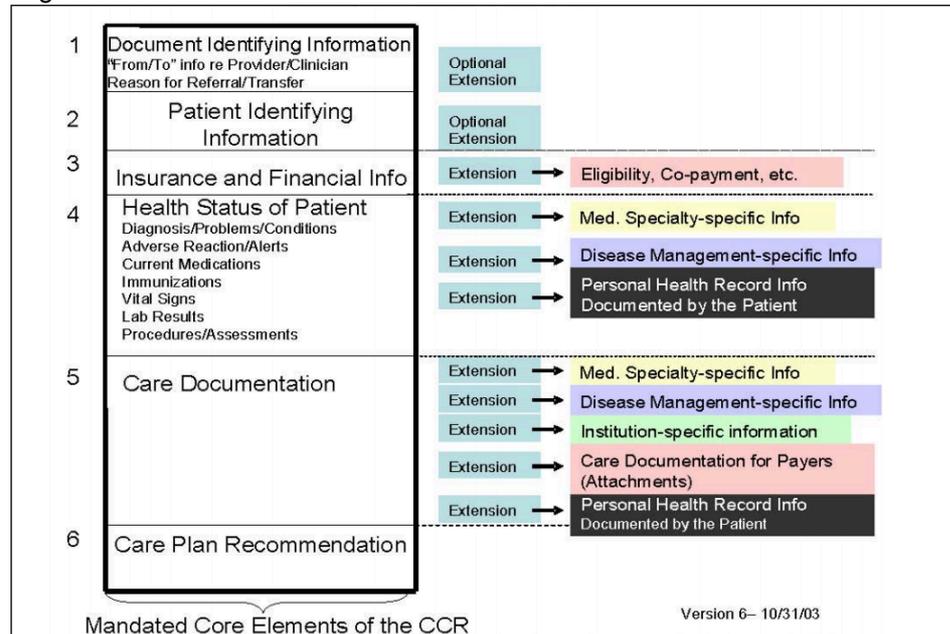
Atualmente, o CCR consiste em três aplicações de elementos centrais obrigatórios. Cada uma dessas aplicações especifica o seu funcionamento.

As aplicações são compostas pelos seguintes elementos:

- a) **referência**: as unidades de saúde devem transferir as informações CCR para o provedor que receberá em formato eletrônico, o mais provável para transmissão é a utilização de e-mail seguro ou HL7, e incluindo o motivo do encaminhamento juntamente com o mínimo de informações propostas;
- b) **transferir**: a partir de uma internação ou consulta institucional: Os usuários ou médicos devem transmitir o CCR para o provedor, as informações com os novos cuidados que o paciente ou a instituição de saúde deverá seguir, estes dados devem ser enviados imediatamente para estarem disponíveis;
- c) **descarga** (sem referência obrigatória ou transferência): O CCR deve ser fornecido ao paciente em papel ou em formato digital para uso futuro incluindo visitas ao Departamento de Atendimento de Urgência ou Emergência.

Conforme a figura 2, a estrutura conceitual de um documento CCR.

Figura 2 - Estrutura conceitual de um documento CCR



Fonte: Continuity of Care Records (2005).

### 2.2.3 HL7

O HL7 é um protocolo internacional utilizado pelo setor de saúde para intercâmbio de dados eletrônicos entre aplicações como LIS para EHR, HIS para RIS (INTEROPERA, 2017).

O padrão HL7 é desenvolvido por uma organização sem fins lucrativos denominada *Health Level Seven* (HL7), fundada em 1987 nos Estados Unidos, mas conta a contribuição de vinte e sete países como Alemanha, Argentina, Austrália, Brasil, Canadá, China, Coréia, Croácia, Dinamarca, Espanha, Finlândia, França, Grécia, Holanda, Índia, Irlanda, Itália, Japão, Lituânia, México, Nova Zelândia, Polônia, Reino Unido, República Tcheca, Suíça, Taiwan e Turquia. O *Health Level Seven International* foi Certificado pela *National Standards Institute* (ANSI) para desenvolver padrões para a área da saúde desde 1994 possui como membros, médicos, técnicos, engenheiros de sistemas de informação, fabricantes de equipamentos, que tem como objetivo promover normas para a interoperabilidade de sistemas em saúde, de forma a aperfeiçoar os atendimentos clínicos, otimizar o fluxo de trabalho, reduzir ambiguidades e permitir a transferência de conhecimento entre profissionais, agências governamentais, indústrias e pacientes (BATISTA, 2016).

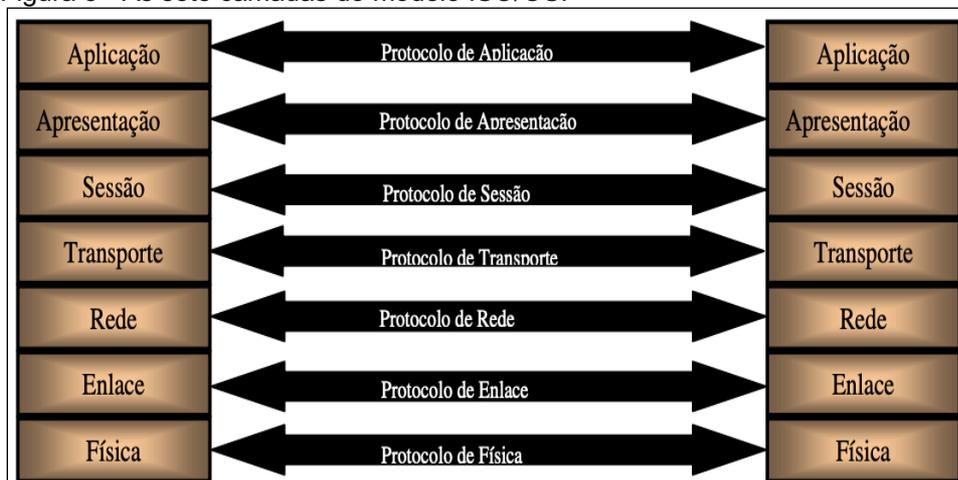
O HL7 não restringe o sistema operacional e nem linguagem de programação a ser utilizada para a troca de informações. Além disso, ele é

independente do protocolo de comunicação e do meio físico e ainda permite que o desenvolvedor defina suas próprias mensagens, o que torna, mais flexível (CARVALHO, 2007a).

Segundo *Health Level 7* (2008a) a missão do HL7 é prover sistemas e padrões relacionados para a troca, integração, compartilhamento e recuperação de informações, para apoio da prática médica e administrativa, permitindo maior controle dos serviços de saúde. Criar metodologias, padrões e diretrizes que sejam compreensivos, viáveis economicamente e que permitam a interoperabilidade e o compartilhamento de informações clínicas armazenadas.

A origem do termo *Level seven* está relacionada à camada de aplicação, que representa o mais alto nível do modelo de comunicação OSI como pode ser vista na figura 3. A camada de aplicação relaciona-se com a implementação de sistemas abertos, sendo assim não é necessária nenhuma restrição quanto ao tipo de rede, comunicação e meio físico. Dessa forma estabeleceu-se que o padrão HL7 tem a finalidade de definir o conteúdo e formato das mensagens que poderão ser trocadas na camada de aplicação (HENRIQUES; CARVALHO, 2005).

Figura 3 - As sete camadas do modelo ISO/OSI



Fonte: ISO (2018).

Segundo HL7 (2008a) o padrão HL7 fornece especificações técnicas nas categorias informativo, rascunho de padrão conforme seu estado evolutivo. As especificações oferecem soluções à comunidade que gerencia informações médicas, visando transpassar as barreiras associadas à interoperabilidade dos sistemas de informação hospitalares. O escopo das questões abordadas pelo HL7

continuará a evoluir na mesma velocidade em que a complexidade de informação médica evoluir.

### **Funcionamento padrão HL7**

A unidade básica de informação, a ser trocada entre os diferentes usuários, é definida no padrão HL7 por mensagem. O padrão especifica além dos atributos principais da troca de mensagem entre sistemas distintos, os vários tipos de mensagem e respectivos dados. Em geral, uma mensagem é formada por segmentos, que por sua vez são formados por campos, sendo estes últimos organizados em componentes (BATISTA, 2016).

#### 2.2.3.1 Tipo de mensagem e formato

Segundo Lopes (2005) o padrão especifica quais são as informações a serem trocadas entre os diferentes setores do hospital. Dentre os diferentes tipos de mensagens estabelecido pelo HL7 podem ser citadas:

- a) controle de pacientes ao nível de admissão, transferências e saídas, pedidos, resultados, observações clínicas e contabilidade;
- b) o padrão HL7 apresenta o formato das mensagens, ou seja, descreve como os dados devem ser representados, o seu tipo e quais os caracteres usados para delimitar os vários segmentos numa mensagem.

#### 2.2.3.2 Eventos

O padrão HL7 especifica as condições em que as mensagens precisam ocorrer, regras como consequência de eventos (*trigger event*). Por exemplo, se for determinado que os dados pessoais de um paciente devem ser disponibilizados, assim quando houver uma admissão de paciente, imediatamente deve ser desencadeada uma mensagem que transmita a informação a todos os interessados (LAJAS, 2015).

### 2.2.3.3 Confirmação de erro

O padrão HL7 também define as ações que devem ser feitas na ocorrência de certos erros entre as aplicações. Por exemplo, caso uma aplicação não receba as informações que esperava, esta pode comunicar a aplicação emissora por meio de uma mensagem de erro. Por outro lado, se a transmissão das informações foi efetuada corretamente, então uma mensagem de confirmação (*ACK-acknowledge*) pode igualmente ser enviada. Garantindo, portanto, segurança na troca de mensagens (LAJAS, 2015).

### **Arquitetura física do sistema**

Segundo HL7 (2014) quanto a arquitetura física do sistema definida pelo padrão HL7 tem-se:

- a) sistema computacional pode estar organizado de uma forma central ou distribuído;
- b) a totalidade das especificações definidas na norma HL7 não necessita obrigatoriamente ser implementada;
- c) a troca de informação pode ser efetuada com base em diversos sistemas operacionais ou linguagens de programação;
- d) o padrão HL7 é independente do meio físico e protocolo de comunicação, podendo, por exemplo, ser usado o protocolo TCP/IP.

### 2.2.4.1 Versões publicadas

Para preencher estas lacunas, o HL7 se empenhou em desenvolver um novo padrão que incorpora o melhor das versões anteriores do HL7, apoiando-se nas lições aprendidas sobre as dificuldades enfrentadas ao longo de seus ciclos de vida, e reaproveitando os melhores, mais modernos e consolidados padrões web como JSON, XML, RES, HTTP dentre outros (INTEROPERA, 2014).

#### 2.2.4.1.1 HL7 Versão 1.

A primeira versão do padrão HL7 (versão 1.0) foi lançada em 1987 e representava um layout de mensagens entre duas ou mais aplicações, disparadas por meio de um evento (*triggered*) particular. Ele especificava o dado a ser enviado, o tipo e o tamanho de cada campo. Mostrava também, se é uma informação é requerida ou opcional no processo, ou se pode ser repetido (HL7, 2014).

#### 2.2.4.1.2 Versões 2.x.

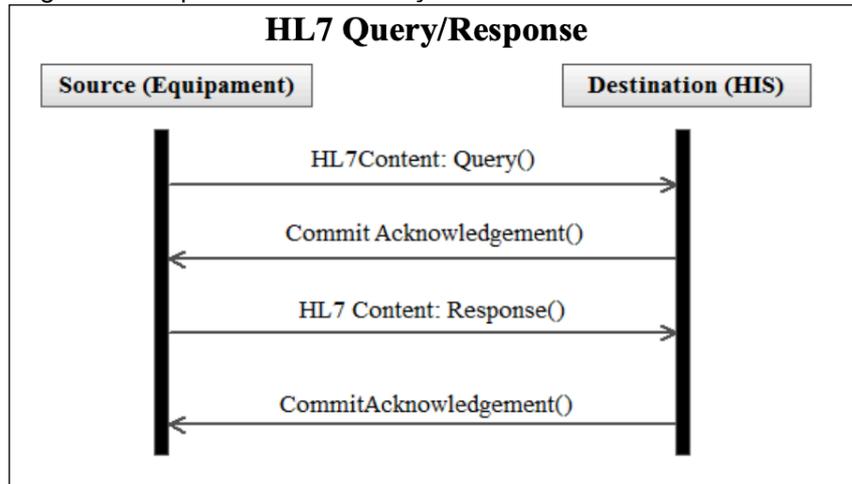
As versões criadas pela família 2.X surgiu em setembro de 1988 e depois desta uma série de atualizações foram feitas. A versão 2.1, apresentada em junho de 1990, constituiu a primeira versão a ser reconhecida e efetivamente adotada (nos Estados Unidos). Em junho de 1994 o HL7 torna-se membro certificado pela *American National Standards Institute* (ANSI) e em dezembro desse mesmo ano surge à versão 2.2, e a sua aplicação começa na prática a ser implementada. Nestas versões, as mensagens são compostas por campos de comprimento variável e separadas por um caractere (habitualmente “[|]”). Cada tipo de dado tem sua forma de codificação especificada e suas condições para repetição. Os campos são agrupados segundo à lógica pré-estabelecida, em uma estrutura denominada segmento. Cada segmento começa com um código de três caracteres que define sua função. Vários segmentos, divididos pelos respectivos separadores (usualmente, caractere de mudança de linha), compõem uma mensagem (HENRIQUES, CARVALHO, 2005).

Alguns aspectos negativos das versões 2.x deve a vários fatores:

- a) algumas especificações permitem interpretações distintas;
- b) processo de integração é complexo;
- c) falta de suporte novas tecnologias, tais como:
- d) web
  - Orientação a objetos,
  - XML - Extensible Markup language,
  - Segurança.

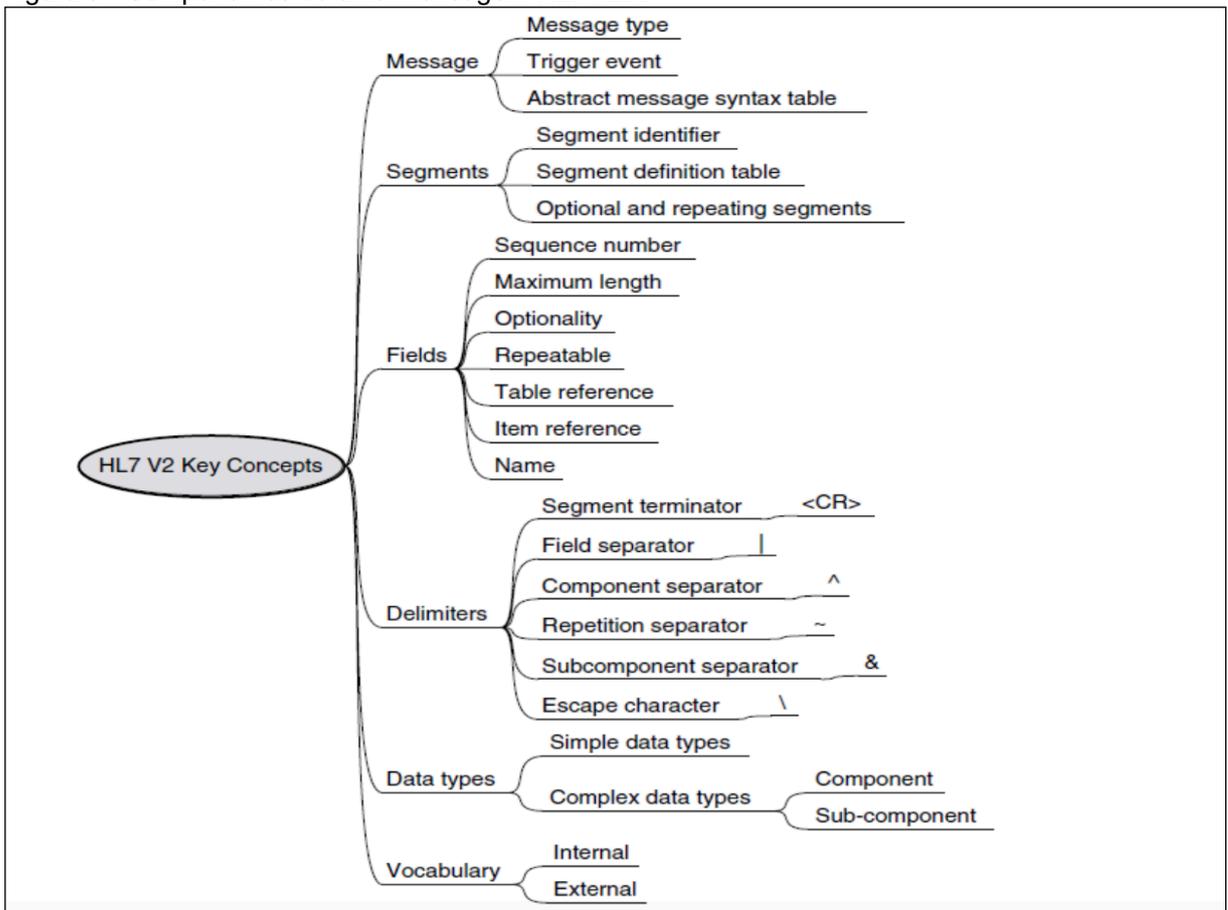
As figuras abaixo, mostram o esquema dos componentes ontológicos da versão 2 de uma mensagem hl7.

Figura 4 - Esquema de comunicação HL7 v2



Fonte: Santos (2014).

Figura 5 - Componentes de uma mensagem HL7 v2.x.



Fonte: Santos (2014).

#### 2.2.4.1.3 Versão 3

A versão 3.0 trouxe uma mudança de paradigma, buscando eliminar limitações das versões anteriores à v.3 utiliza a tecnologia XML, orientação a objetos e segurança para formatação das informações nas suas mensagens, e seus componentes são: XML Schema para validação, HL7 *Reference Information Model* (RIM) para conexões semânticas, ao fornecer um vocabulário padrão e mensagens estruturadas em XML (HL7, 2014).

Ainda no contexto do HL7, se destaca o *Clinical Document Architecture* (CDA), que se trata de um padrão para a representação que especifica a estrutura e semântica de documentos clínicos, baseada em XML, permitindo a compreensão dos mesmos tanto por humanos quanto por máquinas. Além de permitir a utilização de ontologias associadas a estes documentos. Os documentos criados em CDA podem conter textos, imagens, sons e outros conteúdos multimídia (especificados no RIM). Apesar de fazer parte da versão 3.0, suas mensagens não precisam ser compartilhadas nesta mesma versão.

#### 2.2.4.1.4 Tipos de dados utilizado pela versão 3

A forma de comunicação entre ambientes de Saúde é complexa devido à existência de tipos variados de terminologia e conceitos, tornando-se necessário o conhecimento da definição dos valores transportados. Devido a esse motivo, os elementos baseados em HL7 versão 3 possuem um dado específico. De acordo com HL7(2011), o padrão HL7 possui seu próprio conjunto de tipos de dados, pois segundo o próprio comitê, os dados usados pelas diferentes linguagens de programação não são suficientemente poderosos para a representação das informações neste domínio de aplicação. Porém, um número de tipos de dados empregados em diferentes linguagens de programação como: BASIC, Pascal, C, C++ e Java, foram usados como base de entrada para a especificação do conjunto de dados do padrão HL7.

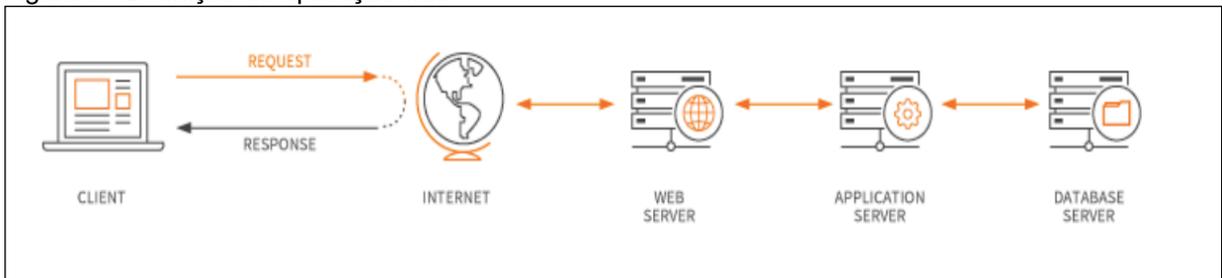
### 3 TECNOLOGIA PARA DESENVOLVIMENTO WEB

Este capítulo traz uma breve explicação dos recursos de tecnologias utilizadas para a elaboração do trabalho proposto, como as Aplicações *Web*, e Interoperabilidade entre sistemas hospitalares.

#### 3.1 APLICAÇÕES WEB

Uma aplicação *web* ou "*Web App*" é um programa de *software* que é executado em um servidor da *web*, utiliza navegadores da *web* e tecnologia para efetuar tarefas pela *internet*. Representação abaixo na figura 6.

Figura 6 - Definição de aplicação web



Fonte: StackPath (2018).

Os aplicativos *webs* usam uma combinação de *scripts* do lado servidor, como por exemplo, PHP e ASP, para manipular o armazenamento e a recuperação das informações e os *scripts* do lado cliente (*JavaScript* e HTML) para apresentar as informações aos usuários. Ao contrário dos aplicativos de *desktop* tradicionais, que são lançados para um sistema operacional, os aplicativos da *Web* devem ser acessados por meio de um navegador da *web*, permitindo que os usuários interajam com a empresa usando formulários *on-line*, sistema de gerenciamento de conteúdo, carrinho de compras e muito mais. Assim os desenvolvedores não precisam criar aplicativos da *web* para várias plataformas, por exemplo, o mesmo aplicativo funciona no *Chrome*, OS X, *Linux* e *Windows*. Além disso, os aplicativos permitem que os usuários criem documentos, compartilhem informações, colaborem em projetos e trabalhem em documentos comuns, independente do local ou dispositivo (CHRISTENSSON, 2014, tradução nossa).

Os aplicativos da *web* geralmente são codificados em linguagem suportada pelo navegador, como HTML, *JavaScript* pois esses idiomas dependem do navegador para tornar o programa executável. Quando você usa um aplicativo

*web*, independentemente se esteja em um aplicativo móvel ou em um computador, a maior parte do processamento é realizado mediante uma rede de servidores. O aplicativo da *web* requer um servidor *web* para gerenciar solicitações do cliente (NDEGWA, 2016).

Fluxo típico de aplicativo da *web*:

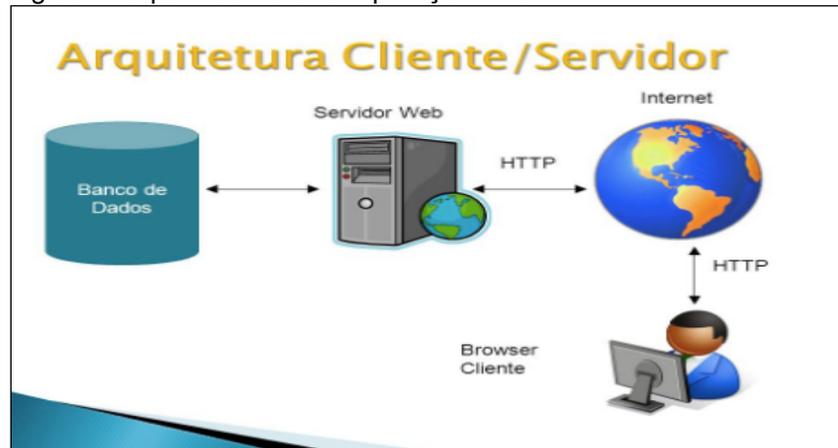
- a) o usuário dispara uma requisição para o servidor da *web* pela *internet*, seja por meio de um navegador da *web* ou da interface do usuário do aplicativo;
- b) o servidor da *web* envia essa solicitação para o servidor de aplicativos da *web* apropriado;
- c) o servidor de aplicativos da *web* executa a tarefa solicitada e regeira os resultados dos dados solicitado;
- d) o servidor de aplicativos da *web* envia resultados para o servidor da *web* com os dados processados;
- e) o servidor da *web* responde de volta ao cliente com as informações solicitadas que aparecem na tela do usuário.

### 3.1.2 Arquitetura das aplicações *web*

A arquitetura de aplicativo *web* é uma estrutura composta pelas relações e interações entre componentes de aplicativos, como sistemas de *middleware*, interface com usuários e banco de dados. No entanto, essa definição é um pouco ambígua, já que quando falamos sobre componentes de um aplicativo da *web*, podem significar duas coisas, UX, mas não tem nada a ver com uma arquitetura de aplicativo da *web* que pretendemos abordar mais sim como *design* de interface do usuário (SCIENCESOFT, 2018, tradução nossa).

Conforme demonstra a figura 7, a arquitetura das aplicações *web* estão divididas em dois componentes principais: Lado do Cliente e o Lado do Servidor.

Figura 7 Arquitetura de Uma Aplicação Web



Fonte: SlidePlayer (2019).

A arquitetura de aplicativos da *web* pode ser definida com a representação desse processo:

- a) o usuário procura por uma URL específica, que o navegador localize e solicita.
- b) na rede, os dados são enviados dos servidores para o navegador e em seguida executado, assim exibindo a página solicitada.
- c) o usuário visualiza e interage com a página.

### 3.1.2.1 Server side

O *script* do lado do servidor, é a tecnologia de desenvolvimento *web* de *back-end*, pode ser definido como um método usado no desenvolvimento *web* para produzir uma resposta customizada para cada requisição do usuário no *website*. O Google usa esses *scripts* para preencher seu termo de pesquisa, colocar anúncios ou descobrir o que você está procurando, ou seja, é a parte mecânica da aplicação, onde são usadas aquelas linguagens como PHP, *Java*, ASP e *Javascript (Node.js)* que de uns tempos pra cá vem ganhando um grande espaço em desenvolvimento de aplicações *web*.

Este lado consiste em pelo menos mais duas partes: lógica da aplicação e banco de dados, onde todos os dados são armazenados.

No *script* do lado do servidor, as respostas podem ser personalizadas exclusivamente com base no requisito do usuário. Para os navegadores da *web* se

comunicarem com os servidores, usam o *Hyper Text Transfer Protocol* (HTTP). O Usuário faz uma solicitação HTTP que é enviada do navegador para o servidor de destino e lhe é retornado um resultado (MOZILLA DEVELOPER, 2018a, tradução nossa).

### 3.1.2.2 *Client side*

O componente mais comum nos olhos do usuário, é a camada da aplicação que interage com os usuários das aplicações, também pode ser definido como um código presente na página HTML de um cliente, que geralmente é anexado ao navegador em um idioma compatível com *browser*.

O navegador faz *download* deste código temporariamente e o processa sem o servidor. Se informações adicionais forem necessárias, uma solicitação será levantada e enviada ao servidor. Na maioria das vezes a linguagem mais usada para *script* no lado do cliente é *JavaScript*. É compatível e tem a capacidade de ser executado em qualquer navegador como (*Chrome*, *IE*, *Firefox*, entre outros).

O *Javascript* no lado do cliente (sabendo que também está no lado do servidor com o *Node.js*) e *Ajax*, estas são duas das tecnologias de *client-side* que mais vem crescendo na área de desenvolvimento *web*, como resultado, agora é mais rápido e mais fácil fazer *scripts* no lado do cliente, deixando menos trabalho para o servidor.

## 3.2. APPLICATION PROGRAMMING INTERFACE (API)

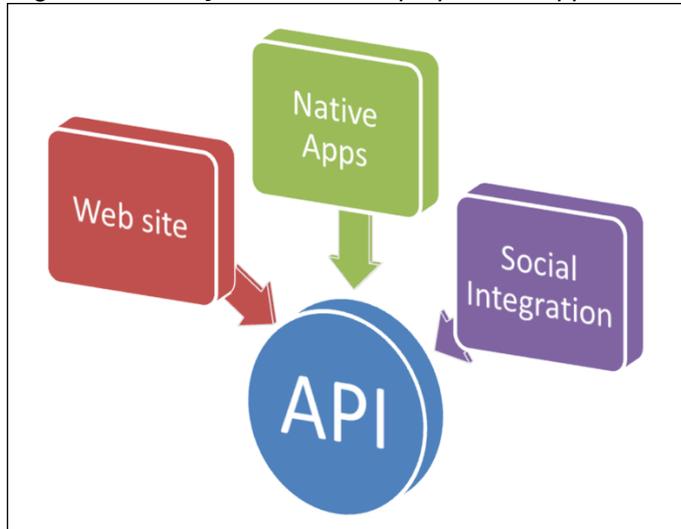
Uma interface de programação de aplicativo (API) é uma especificação destinada a ser usada como uma interface por *software* componentes para se comunicarem uns com os outros. Uma API pode incluir especificações para rotinas, estruturas de dados, objeto classes e variáveis.

Interfaces de programação de aplicativos (APIs) expõem serviços ou dados fornecidos por uma aplicação de *software* através de um conjunto de recursos pré-definidos, tais como métodos, objetos ou URIs (Stylos et al, 2009).

Usando estes recursos, outros aplicativos podem acessar os dados ou serviços sem ter que implementar os objetos e procedimentos subjacentes. APIs são centrais para muitas arquiteturas de *software* modernas como mostra a figura 8, já

que elas fornecem abstrações de alto nível ações que facilitam as tarefas de programação e a reutilização de código (ROBILLARD, 2009).

Figura 8 - Definição de interface *preparation application*



Fonte: Secode (2017).

API permite a interoperabilidade entres aplicações e usuários, reforçando ainda mais a importância de pensarmos em algo padronizado e de preferência, de fácil representação e compreensão humana, conforme as figuras 9,10 e 11 (PIRES, 2017).

Alguns exemplos de representações de dados de APIs:

a) xml

XML (*Extensible Markup Language*) é um formato de texto simples e muito flexível derivado do SGML (ISO 8879), originalmente projetado para enfrentar os desafios da publicação eletrônica em larga escala, o XML também está desempenhando um papel cada vez mais importante na troca de uma ampla variedade de dados na *Web* e em outros lugares (XML,2019), na figura 9 temos a representação de uma mensagem xml.

Figura 9 Representação de dados XML

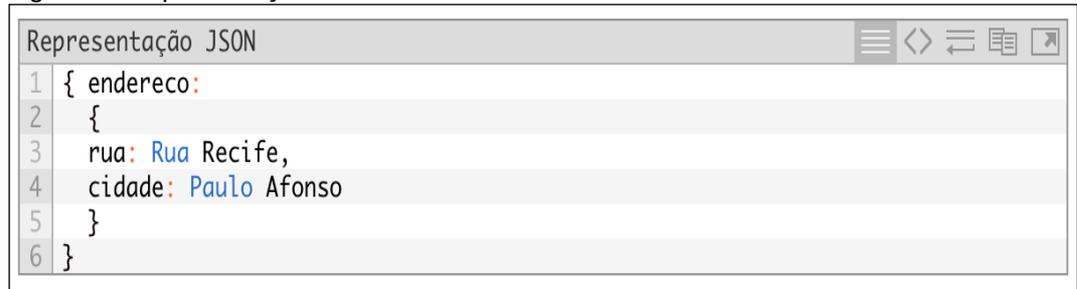
Representação XML	
1	<endereco>
2	<rua>
3	Rua Recife
4	</rua>
5	<cidade>
6	Paulo Afonso
7	</cidade>
8	</endereco>

Fonte: Secode (2017).

## b) Json

Json (*JavaScript Object Notation*) é um formato leve de intercâmbio de dados. É legível e de fácil escrita, tanto para humanos quanto para as máquinas analisarem, é baseado em um subconjunto do ECMA-262, json em um formato de texto que é completamente independente da linguagem, mas usa convenções familiares aos programadores da família da linguagem C incluindo C ++, C #, Java, Perl, *Python* e muitos outros. Essas propriedades tornam o JSON uma linguagem ideal de intercâmbio de dados (JSON, 2019), na figura 10 temos a representação de uma mensagem em json

Figura 10 Representação de dados JSON

A screenshot of a code editor window titled "Representação JSON". The editor shows a JSON object with the following content:

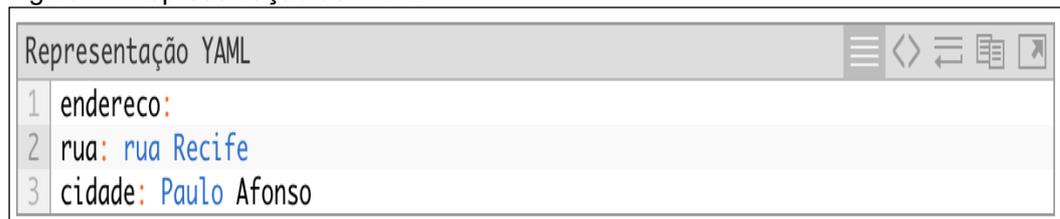
```
1 { endereco:  
2   {  
3     rua: Rua Recife,  
4     cidade: Paulo Afonso  
5   }  
6 }
```

Fonte: Secode (2017).

## c) Yaml

YAML é um formato de dados estruturado legível por humanos. É menos complexo e desagradável que XML ou JSON, mas fornece recursos semelhantes. Essencialmente, permite fornecer definições de configurações poderosas, sem a necessidade de aprender um tipo de código mais complexo, como CSS, *JavaScript* e PHP (LEARN,2019), na figura 11 temos a representação de uma mensagem em yaml.

Figura 11 Representação de YAML

A screenshot of a code editor window titled "Representação YAML". The editor shows a YAML object with the following content:

```
1 endereco:  
2   rua: rua Recife  
3   cidade: Paulo Afonso
```

Fonte: Secode (2017).

Segundo Myers e Stylos (2016) apontam que todo *software* moderno faz uso de APIs, em vez de programar as funcionalidades a partir do zero, o fundamental da tarefa dos desenvolvedores de *software* agora muitas vezes é costurar, funcionalidades que as APIs existentes fornecem.

Portanto, aprender a usar novas APIs é uma tarefa diária que muitos desenvolvedores enfrentam. Para ajudar nessa tarefa, as APIs são normalmente enviadas com documentação, como guias do programador, livros, tutoriais e documentação de referência da API (MIHALY, 2011; WATSON, 2015).

### 3.2.1 Microservice

Segundo Newman (2015), *microservice* são serviços com poucas responsabilidades, que podem trabalhar de forma independente ou em conjunto com outros serviços. Ao contrário de sistemas monolítico, cada *microservice* é empacotado em um artefacto separado, e pode-se realizar a implantação de cada um independentemente.

Quanto menor o serviço, maior a maximização dos benefícios e minimização das desvantagens da arquitetura orientada a *microservice*. Serviços extremamente pequenos tornam-se mais independentes, porém a complexidade de desenvolvimento e *deploy* aumentam consideravelmente.

Cada *microservice* é uma entidade separada e autônoma, eles podem ser implantados em diferentes máquinas, e ainda sim trabalhem de forma coletiva, sendo capaz de executar seus processos de forma independente caso algum outro serviço falhe. Para garantir a separação entre eles, todas as comunicações e realizações por chamadas de rede (FARCIC, 2015).

Na arquitetura *microservice*, um aplicativo é dividido em serviços. Cada serviço executa um processo exclusivo e normalmente gerencia seu próprio banco de dados. Um serviço pode gerar alertas, registrar dados, suportar Interfaces de Usuário, manipular a identificação ou autenticação do usuário e executar várias outras tarefas.

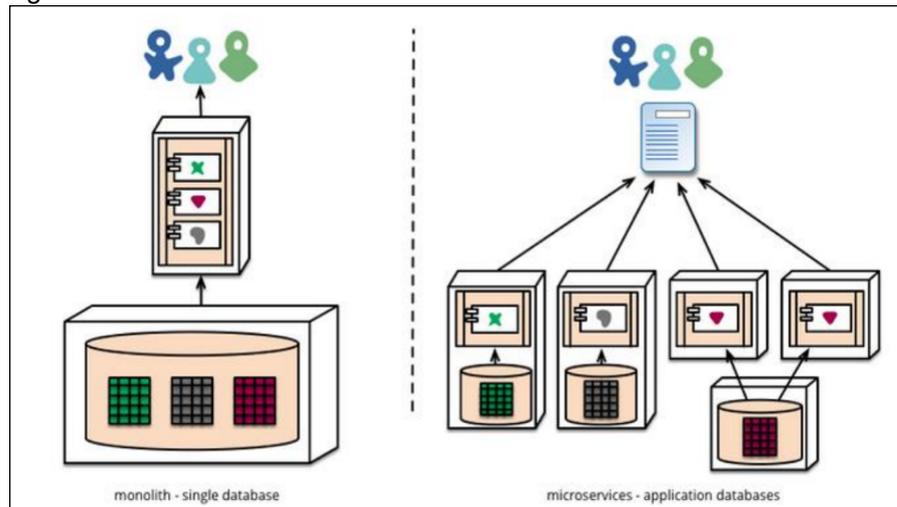
O paradigma de *microservice* fornece às equipes de desenvolvimento uma abordagem mais descentralizada para a criação de *software*.

Os *microservice* permitem que cada serviço seja isolado, reconstruído, implementado e gerenciado independentemente. Por exemplo, se um programa não

estiver gerando relatórios corretamente, pode ser mais fácil rastrear o problema para esse serviço específico, poderia então ser testado, reiniciado, corrigido e implementado conforme necessário, independentemente de outros serviços (ROUSE, 2018, tradução nossa).

A descentralização do gerenciamento de dados também segue a mesma ideia, pois para cada *microservice* existe sua própria base de dados conforme demonstra a figura 12.

Figura 11 - Monolítica e Microservice Banco de dados.



Fonte: Viana (2017).

## 4 TRABALHOS CORRELATOS

Diante da evolução da tecnologia, diversas pesquisas e desenvolvimentos são realizados no mundo inteiro. Algumas áreas como tecnologia móvel e recursos *Web* são muito exploradas, devido a grande quantidade de usuários a qual essas plataformas atingem e a variedade de recursos para o desenvolvimento que elas disponibilizam.

Nos tópicos subsequentes são abordadas algumas dessas pesquisas relacionadas ao presente trabalho.

### 4.1 ANÁLISE COMPARATIVA DE PADRÕES DE INTEROPERABILIDADE EM SISTEMAS HOSPITALARES

Em 2016, a acadêmica Tatiane Aparecida Batista, do programa de graduação, do Curso de Ciência da Computação do Departamento de Ciências Exatas e Engenharia (DCEEEng), da Universidade Regional do Noroeste do Estado do Rio Grande do Sul (UNIJUI), elaborou um estudo sobre interoperabilidade em sistemas hospitalares.

Havendo uma quantidade significativa de padrões de interoperabilidade de sistemas, constatou-se a necessidade de efetuar uma análise e avaliação comparativa desses padrões. Justamente por haver uma quantidade expressiva de padrões que podem ser utilizados para a interoperabilidade de sistemas, o presente trabalho busca auxiliar os desenvolvedores de sistemas de informação em saúde para a escolha do mais adequado e que sustente as necessidades da implementação. Torna-se, portanto, imprescindível que seja definido com clareza que tipo de informação deve ser documentado, e quais tipos de dados devem ser compartilhados para que se consolide um Sistema de Informação adequado a todas as instituições que venham a utilizá-los.

## 4.2 MODELOS PARA INTEROPERABILIDADE DE SISTEMAS HOSPITALARES UTILIZANDO PADRÃO HL7

Em 2005, as acadêmicas Karine Petry e Paula Marien Albrecht Lopes, do programa de graduação, do Curso de Sistemas de Informação do Departamento de Informática e Estatística, da Universidade Federal de Santa Catarina elaborou um estudo sobre interoperabilidade em sistemas hospitalares utilizando padrão HL7.

A aceitação em grande escala da informática proporcionou maior investimento no setor. Em decorrência disto, houve um aprimoramento tecnológico em *hardware* e *software*.

Dessa forma, diversos sistemas de informação foram desenvolvidos. Surge então, o problema da interoperabilidade - sistemas necessitam se comunicar para permitir o acesso às informações. Assim, torna-se indispensável o estabelecimento de padrões com o intuito de promover esta tão esperada interoperabilidade.

A área de saúde por manipular com informações de elevada importância deve receber um tratamento especial, dessa forma, em 1987, um conjunto de membros de diversos países reuniram-se com o objetivo de elaborar um padrão de caráter internacional que promova a interoperabilidade de sistemas da área de saúde. Este padrão foi denominado de HL7.

O HL7 é um padrão reconhecido pela ANSI que permite a comunicação entre sistemas através do envio de mensagens, sendo que sua flexibilidade permite a interoperabilidade de sistemas independente dos equipamentos, sistemas operacionais, linguagens e banco de dados utilizados.

Explorando ainda mais esta flexibilidade, verificou-se que é possível o desenvolvimento de um *middleware* projetado segundo as especificações do padrão HL7 capaz de facilitar a integração de sistemas legados com os sistemas de última geração.

### 4.3 ESTUDO DA PADRONIZAÇÃO VISANDO À INTEROPERABILIDADE: O CASO DAS ORGANIZAÇÕES DE SAÚDE NA CIDADE DE MARÍLIA - SÃO PAULO

Em 2015, o acadêmico Nelson Júlio de Oliveira Miranda do Programa de Pós-Graduação em Ciência da Informação da Universidade Estadual Paulista, Campus de Marília elaborou um estudo sobre Padronização de interoperabilidade das organizações de saúde.

Nas organizações, as TIC adentraram automatizando os processos de gestão, a preocupação com padrões só ocorreu com a necessidade de troca de dados e informações entre sistemas intra e interorganizacionais.

A interoperabilidade entre sistemas é entendida como a capacidade de dois ou mais sistemas trocarem dados e informações, e deve ocorrer nos níveis sintático e semântico. Para que isso ocorra, acordos precisam ser firmados, no sentido de definir uma padronização na escrita e no significado dos dados e informações a serem trocados.

Nas organizações de saúde já se convive com padrões específicos (HL7, DICOM, openEHR, CID etc.). Além desses, também se destacam o MeSH e o DeCS. Ademais, também são adotados os padrões genéricos de infraestrutura e comunicação (Ethernet, TCP/IP, HTML, XML, JSON etc.). Modelos de referência da informação como o HL7 RIM e openEHR RIM definem qual a arquitetura da informação em saúde ideal.

Terminologias e nomenclaturas, representadas pelo SNOMED-CT, CID, LOINC etc., procuram garantir o entendimento do significado do que é registrado e transferido pelos sistemas. Somado a esses padrões, estão os aspectos éticos e legais inerentes aos dados e informações para a saúde do indivíduo, que devem ser levados em consideração quando interoperáveis.

#### 4.4 PLATAFORMA PARA M-HEALTH BASEADA NO PADRÃO OPENEHR, EM COMUNICAÇÕES M2M E EM COMPUTAÇÃO EM NUVEM.

Em 2016, o acadêmico Jesús Noel Suárez Rubí, do programa de Dissertação de Mestrado em Engenharia Elétrica, Departamento de Engenharia Elétrica da Universidade de Brasília Elétrica Faculdade de Tecnologia, elaborou um estudo baseado no padrão OPENEHR, em comunicação M2M e computação em nuvem.

Os recentes desenvolvimentos das comunicações móveis e da Internet e sua subsequente disseminação nos mais diversos setores da sociedade têm favorecido o surgimento de novos serviços, acessíveis a partir de terminais móveis, para o provimento de informações médicas, monitoramento, vigilância e tratamento de saúde.

O monitoramento de parâmetros fisiológicos por meio de equipamentos eletrônicos tem permitido aos profissionais da saúde a realização de diagnósticos de diferentes tipos de doenças, tanto de forma presencial em hospitais, laboratórios e clínicas especializadas, como também por meio de tele monitoramento. Por outro lado, em termos de políticas públicas, verifica-se que há iniciativas importantes, como o e-SUS, o Prontuário Eletrônico do Paciente e a possível adoção do *Open Electronic Health Record* (OpenEHR) no Brasil.

Tais iniciativas caminham na direção de uma progressiva implantação de *e-Health* em nosso Sistema Único de Saúde (SUS) e devem ser consideradas para a efetiva informatização dos serviços de saúde em nosso país.

## **5 FERRAMENTA DE INTEGRAÇÃO DE DADOS PARA COMPARTILHAMENTO DE INFORMAÇÕES RELACIONADAS AOS SERVIÇOS DISPONÍVEIS EM UNIDADES DE SAÚDE**

Este trabalho tem como o objetivo desenvolver um protótipo de aplicação *web* usando a integração de dados.

A integração de dados que será utilizado neste protótipo será proveniente de uma API que terá como finalidade integrar os dados das unidades de saúde. Quando acessado à API será capaz de fornecer informações para auxiliar nas unidades de saúde que possuem dificuldades em comunicar-se umas às outras.

### **5.1 METODOLOGIA**

Para a concretização deste trabalho foi realizado o levantamento bibliográfico relacionado às unidades de saúde que apresentam dificuldade em comunicar-se umas às outras. Sendo assim, foi proposto o desenvolvimento de um protótipo de uma aplicação *Web* para a integração de dados para o compartilhamento de serviços disponíveis nas unidades de saúde.

As operações relacionadas a integração de dados será responsabilidade da API a ser desenvolvida com integração de uma biblioteca de análise para gerar, enviar e criar uma mensagem no padrão hl7, já na implementação será usando a linguagem de programação PHP e ECMAScript. Primeiramente será feito uma análise do funcionamento dos sistemas de integração de dados existentes e em seguida será feito uma outra análise com relação ao envio das informações para que as outras unidades de saúde possam ter aceso, após essa análise será dado início ao desenvolvimento do protótipo

#### **5.1.1 Biblioteca Php para Analisar, Gerar e Enviar Mensagens hl7 v2.x.**

As mensagens HL7 v2.x usam uma sintaxe para sua codificação que não é baseada em XML, mas sim em segmentos (linhas) e delimitadores de caracteres. Os segmentos possuem campos separados pelo delimitador. Um campo composto pode ter componentes separados pelos subcomponentes.

Os delimitadores padrão são barra vertical ou pipe ( | ) serve como separador de campos, o caractere ( ^ ) corresponde com separador de componentes e o e comercial ( & ) como separador de subcomponentes. O caractere de til ( ~ ) é o separador de repetições. Todo segmento começa com uma *string* de 3 caracteres que identifica seu tipo.

Cada segmento da mensagem contém uma categoria específica de informações. Toda mensagem tem o MSH como primeiro segmento, e inclui um campo que identifica o tipo de mensagem, definindo assim a interação, origem, destino e algumas especificidades da sintaxe de uma mensagem, como podemos ver na figura 13 (RADES, 2018).

Figura 13 – Mensagem hl7.

```
MSH|^~\&|ADT1|MCM|LABADT|MCM|198808181126|SECURITY|ADT^A01|MSG00001-|P|2.3
EVN|A01|198808181123
PID|||PATID1234^5^M11||JONES^WILLIAM^A^III||19610615|M-||C|1200 N ELM
STREET^^GREENSBORO^NC^27401-1020|GL|(91-9)379-1212|(919)271-
3434||S||PATID12345001^2^M10|123456789|9-87654^NC
NK1|1|JONES^BARBARA^K|WIFE|||||NK
PV1|1|I|2000^2012^01|||004777^LEBAUER^SIDNEY^J.||SUR||-||ADM|A0-
AL1|1||^Penicillin||Produces hives
AL1|2||^Cat dander|Respiratory distress
```

Fonte: Do Autor (2019).

A biblioteca tem como princípio básico de funcionamento receber uma mensagem no padrão hl7 v2.x, e submetê-la ao processo de decodificação da mensagem e armazená-la em uma base de dados preparada com os segmento do hl7v2.x.

O reconhecimento dos dados é acessado por meio de classe *Message*, que proporciona a capacidade de reconhecimento da mensagem e responder apropriadamente. Geralmente, é usado o construtor da classe para criar um novo objeto *Message*.

Para os serviços disponíveis em unidades de saúde, não existe um segmento com o protocolo de mensagem hl7v2, então optou-se no uso do *JavaScript Object Notation* por ser um padrão de fácil compreensão que a maioria dos sistemas usam.

Com isso os dados relacionados ao serviço disponível, são enviados para a API no formato JSON onde será informado o nome do hospital, o código do exame universal e o status do exame, onde a classe *ServicesAvailableHospitals* fará todo processo, para armazenar na base de dados e disponibilizar para outras unidades de saúde.

#### 5.1.1.1 Demonstração de análise e criação da mensagem hl7

O segmento HL7 MSH (cabeçalho da mensagem) está presente em todos os tipos de mensagens HL7 e define a origem, a finalidade, o destino e determinadas especificidades de sintaxe da mensagem, como delimitadores (caracteres, separadores). É sempre o primeiro segmento na mensagem HL7, com a única exceção sendo as mensagens em lote HL7 (COREPINT, 2019).

Existem 19 campos no segmento MSH, seis dos quais (separador de campo, caracteres de codificação, tipo de mensagem, ID de controle de mensagem, ID de processamento e ID da versão) são necessários para todas as mensagens processadas usando o padrão HL7. O mais importante dos campos MSH, e talvez o mais importante em toda a mensagem é o MSH-9 (Tipo de Mensagem). Este campo especifica que tipo de mensagem está sendo transmitida e qual é o evento de disparo. Quando uma mensagem é carregada, geralmente o primeiro campo examinado para determinar o processamento é o valor neste campo.

Segue-se na tabela 1 com os campos do segmento MSH.

Tabela 1 – Campos do segmento MSH.

Seq	Tam	Td	Opc	Nome do elemento
1	1	ST	R	Separador de campo
2	4	ST	R	Caracteres de codificação
3	180	HD	O	Aplicativo de envio
4	180	HD	O	Instalação de envio
5	180	HD	O	Aplicativo de recebimento
6	180	HD	O	Instalação receptora
7	26	TS	O	Data / Hora da Mensagem
8	40	ST	O	Segurança
9	7	CM_MSG	R	Tipo de mensagem

<b>Seq</b>	<b>Tam</b>	<b>Td</b>	<b>Opc</b>	<b>Nome do elemento</b>
10	20	ST	R	ID do controle de mensagens
11	3	PT	R	ID de processamento
12	8	ID	R	ID da versão
13	15	NM	O	Número sequencial
14	180	ST	O	Ponteiro de Continuação
15	2	EU IRIA	O	Aceitar tipo de reconhecimento
16	2	ID	O	Tipo de reconhecimento de aplicativo
17	2	ID	O	Código do país
18	6	ID	O	Conjunto de caracteres
19	3	CE	O	Idioma principal da mensagem

Fonte: Corepointhealth (2019).

O segmento PID HL7 é encontrado em todos os tipos de mensagem e contém 39 campos diferentes com valores que variam de número de identificação do paciente, sexo, endereço, estado civil, cidadania. O segmento PID fornece informações importantes de identificação sobre o paciente e, de fato, é usado como o principal meio de comunicação das informações demográficas e de identificação sobre um paciente entre os sistemas. Devido à natureza das informações encontradas no segmento PID, é improvável que mude com frequência (COREPINT, 2019).

Segue-se na tabela 2 com os campos do segmento PID.

Tabela 2– Campos do segmento PID.

<b>Seq</b>	<b>Tam</b>	<b>Td</b>	<b>Opc</b>	<b>Nome do elemento</b>
1	4	SI	O	ID do conjunto - ID do paciente
2	20	CX	O	ID do paciente (ID externo)
3	20	CX	R	ID do paciente (ID interno)
4	20	CX	O	Identificação alternativa do paciente - PID
5	48.	XPN	R	Nome do paciente
6	48.	XPN	O	nome de solteira da mãe
7	26	TS	O	Data / Hora de Nascimento
8	1	IS	O	Sexo

<b>Seq</b>	<b>Tam</b>	<b>Td</b>	<b>Opc</b>	<b>Nome do elemento</b>
9	48.	XPN	O	Alias do paciente
10	1	IS	O	Raça
11	106	XAD	O	Endereço do paciente
12	4	IS	B	Código do país
13	40.	XTN	O	Número de telefone - Página inicial
14	40.	XTN	O	Número de telefone - Empresas
15	60	CE	O	Idioma principal
16	1	IS	O	Estado civil
17	3	IS	O	Religião
18	20	CX	O	Número da conta do paciente
19	16	ST	O	Número SSN - Paciente
20	25	DLN	O	Número da carteira de motorista - Paciente
21	20	CX	O	Identificador da Mãe
22	3	IS	O	Grupo étnico
23	60	ST	O	Local do nascimento
24	2	ID	O	Indicador de nascimento múltiplo
25	2	NM	O	Ordem de nascimento
26	4	IS	O	Cidadania
27	60	CE	O	Veteranos Military Status
28.	80	CE	O	Nacionalidade
29	26	TS	O	Data e Hora da Morte do Paciente
30	1	ID	O	Indicador de Morte do Paciente

Fonte: Caristix (2019).

O segmento EVN é usado para comunicar informações de eventos de acionamento a aplicativos receptores.

Segue-se na tabela 3 com os campos do segmento EVN.

Tabela 3 – Campos do segmento EVN.

<b>seq</b>	<b>len</b>	<b>dt</b>	<b>optar</b>	<b>Nome do elemento</b>
1	3	ID	B	Código do tipo de evento
2	26	TS	R	Data / Hora Gravada

<b>seq</b>	<b>len</b>	<b>dt</b>	<b>optar</b>	<b>Nome do elemento</b>
3	26	TS	O	Data / hora do evento planejado
4	3	IS	O	Código do motivo do evento
5	60	XCN	O	ID do Operador
6	26	TS	O	Evento ocorrido

Fonte: Caristix (2019)

O segmento HL7 NTE é usado para enviar notas e comentários em uma mensagem.

Segue-se na tabela 4 com os campos do segmento NTE.

Tabela 4 – Campos do segmento NTE.

<b>Seq</b>	<b>Len</b>	<b>Td</b>	<b>opc</b>	<b>Nome do elemento</b>
1	4	SI	O	Definir ID NTE
2	8	ID	O	Fonte do comentário
3	64k	FT	O	Comente
4	60	CE	O	Tipo de comentário

Fonte: Caristix (2019).

O segmento HL7 OBR transmite informações sobre um exame, estudo observação de diagnóstico ou avaliação específica de um pedido ou resultado. É usado com mais frequência nas mensagens ORM e ORU e tem um papel importante no transporte dos números de pedidos de lugares e preenchimento (CARISTIX, 2019).

Um segmento OBR sempre deve ser incluído nos pedidos em que uma solicitação é feita para um determinado conjunto de observações. Para testes de laboratório e estudos de diagnóstico, cada segmento OBR geralmente se aplica a uma única amostra ou estudo de diagnóstico.

Segue-se na tabela 5 com os campos do segmento OBR.:

Tabela 5 – Campos do segmento OBR.

<b>Seq</b>	<b>Tam</b>	<b>Td</b>	<b>Opc</b>	<b>Nome do elemento</b>
1	4	SI	C	ID do conjunto – OBR
2	75	EI	C	Número do pedido do Placer

<b>Seq</b>	<b>Tam</b>	<b>Td</b>	<b>Opc</b>	<b>Nome do elemento</b>
3	75	EI	C	Número do pedido de preenchimento
4	200	CE	R	ID do serviço universal
5	2	ID	B	Prioridade
6	26	TS	B	Data / hora solicitada
7	26	TS	C	Data / Hora da Observação
8	26	TS	O	Data / Hora de término da observação
9	20	CQ	O	Volume da coleção
10	60	XCN	O	Identificador do coletor
11	1	ID	O	Código de ação da amostra
12	60	CE	O	Código de Perigo
13	300	ST	O	Informações clínicas relevantes.
14	26	TS	C	Data / Hora da Amostra Recebida
15	300	CM	O	Fonte da amostra
16	80	XCN	O	Provedor de pedidos
17	40.	XTN	O	Número de telefone para retorno de chamada do pedido
18	60	ST	O	Campo 1 do marcador
19	60	ST	O	Campo 2 do marcador
20	60	ST	O	Campo de preenchimento 1
21	60	ST	O	Campo de preenchimento 2
22	26	TS	C	Rpt Resultados / Status Status – Data / Hora
23	40.	CM	O	Carregar para praticar
24	10	ID	O	ID da Seita de Serviço de Diagnóstico
25	1	ID	C	Status do resultado
26	400	CM	O	Resultado dos Pais
27	200	TQ	O	Quantidade / Tempo
28.	150	XCN	O	Cópias de resultado para
29	150	CM	O	Pai
30	20	ID	O	Modo de transporte
31	300	CE	O	Motivo do Estudo
32.	200	CM	O	Intérprete de resultado principal
33	200	CM	O	Intérprete Assistente de Resultado

Seq	Tam	Td	Opc	Nome do elemento
34	200	CM	O	Técnico
35	200	CM	O	Transcriptionist
36.	26	TS	O	Data / Hora Programada
37.	4	NM	O	Número de recipientes para amostras
38.	60	CE	O	Logística de transporte de amostra coletada
39.	200	CE	O	Comentário do Colecionador
40.	60	CE	O	Responsabilidade do Acordo de Transporte
41	30	ID	O	Transporte organizado
42.	1	ID	O	Acompanhamento obrigatório
43	200	CE	O	Comentário sobre o transporte planejado de pacientes

Fonte: Caristix (2019).

Com a biblioteca de análise, geração e envio de mensagem HL7 v2.x baseado em PHP, inspirado no pacote Perl Net-HL7, conseguimos ver alguns exemplos de representação de análise e criação de uma mensagem.

a) Análise da mensagem

Na figura 14 há uma representação de análise de uma mensagem no padrão HL7 v2X.

Figura 14 – Representação de análise de mensagem.

```

$msg = new Message("MSH|^~\&|1|\rPID|||abcd|\r");
$pid = $msg->getSegmentByIndex(1);
echo $pid->getField(3);
echo $msg->toString(true);
$msg->getFirstSegmentInstance('PID');
$msg->hasSegment('PID');

```

Fonte: Do Autor (2019).

b) Criação de uma nova mensagem

Já na figura 15 há a representação da criação de uma mensagem no padrão HL7 v2X.

Figura 15 – Representação da criação de mensagem.

```

$msg = new Message();
$msh = new MSH();
$msg->addSegment($msh);
$abc = new Segment('ABC');
$abc->setField(1, 'xyz');
$abc->setField(4, ['']);
$msg->setSegment($abc, 1);

$pid = new PID();
$pid->setPatientName([$lastname, $firstname, $middlename, $suffix]);
$pid->setField('abcd', 5);
unset($pid);

$msg = new Message("MSH|^~\&|||||ORM^001|P|2.3.1|", null, true, true);
$h17String = "MSH|^~\&|||||ORU^R01|00001|P|2.3.1|\n" . "OBX|1||11^AA|\n" . "OBX|1||22^BB|\n";
$msg = new Message($h17String, null, true, true, false);

$msg = new Message("MSH|^~\&|1|\rPV1|1|0|^AAAA1^^^BB|", null, true);
$pv1 = $msg->getSegmentByIndex(1);
$fields = $pv1->getField(3);

$msg = new Message("MSH|^~\&|1|\nABC|||xxx\n", ['SEGMENT_ENDING_BAR' => false]);
$msg->toString(true);
(new Connection($ip, $port))->send($msg);
$msg = new Message("MSH|^~\&|1|\rPV1|1|0|^AAAA1^^^BB|",
    ['SEGMENT_SEPARATOR' => '\r\n',
     'HL7_VERSION' => '2.3']);

```

Fonte: Do Autor (2019)

### 5.1.1.2 Construção da API

Como foi citado no ponto de desenvolvimento do protótipo, o *hl7-api* foi desenvolvido usando o *framework laravel*, por ser rápido, livre, de código aberto, e por usar o padrão *Model, View, Controller* (MVC), mais para a construção da api usou-se simplesmente o *model* juntamente com o *Eloquent* do laravel que facilitou nas consultas e manipulação dos dados na base de dados, e o *controller* que foi o responsável por receber todas requisições do usuário, instanciando assim a classe *Message* onde será validade a mensagem da enviada.

O *laravel* mostrou-se uma tecnologia propícia para o desenvolvimento em aplicações *web*, visto que ela é gratuita e possui uma ótima documentação e uma boa comunidade. E um dos pontos que fez com que o *laravel* fosse selecionado, foi por ser o *framework* PHP mais usado atualmente (TRENDS, 2019).

O mesmo usa o *composer* para gerenciar suas dependências, o *composer* é uma ferramenta para gerenciamento de dependências em PHP. Ele permite que se declarem as bibliotecas das quais seu projeto depende e as gerencia (instala / atualiza), o *composer* requer o PHP 5.3.2+ para que seja executado,

algumas configurações sensíveis de php e sinalizadores de compilação também são necessárias. (COMPOSER,2019).

Com o *framework* instalado e a configuração certa, iniciou-se a instalação da biblioteca usando o *composer*, e em seguida foi construído uma base de dados usando os campos dos segmentos PID, MSH, EVN, NTE, e OBR para que se tenha sempre o histórico dos dados do paciente.

A princípio foi criada uma classe, ou seja, um *Controller* denominado *SaveMessage*, visto que usamos esse *controller* para receber todas as requisições enviadas contendo a mensagem no padrão hl7 v2x, neste método declaramos os métodos responsáveis para manipulação dos campos de um determinado segmento óbitos na mensagem sendo eles necessários no desenvolvimento do protótipo como podemos ver na figura 16.

Figura 16 – Método para salva a mensagem no padrão hl7

```

use Aranyasen\HL7\Message;
use Aranyasen\HL7\Segment;
use Exception;
use Illuminate\Support\Facades\DB;
use App\Http\Controllers\V1\SaveMessageMsh;

class SaveMessage extends Controller
{
    static function saveMessage(string $message)
    {
        $msg = new Message($message);

        DB::beginTransaction();
        try {
            $pidResponse = SaveMessagepid::saveMessagepid($msg);
            $mshResponse = SaveMessageMsh::saveMessageMsh($msg, $pidResponse->id);
            $evnResponse = SaveMessageevn::saveMessageEvn($msg, $pidResponse->id);
            $nteResponse = SaveMessagnte::SaveMessagnte($msg, $pidResponse->id);
            $obrResponse = SaveMessageObr::saveMessageObr($msg, $pidResponse->id);
            DB::commit();
            return ['msg' => $msg];
        } catch (Exception $ex) {
            DB::rollBack();
            return ['msg_erro' => $ex->getMessage()];
        }
    }
}

```

Fonte: Do Autor (2019).

Para realizar a conversão dos dados com a API usando a biblioteca, se faz necessário seguir alguns passos básicos. Com as instâncias das classes *SaveMessagepid*, *SaveMessageMsh*, *SaveMessageevn*, *SaveMessagnte* e *SaveMessageObr* criadas, é necessário chamar os métodos *saveMessagepid()*, passando com parâmetro a mensagem já para o os métodos *saveMessageMsh()*,

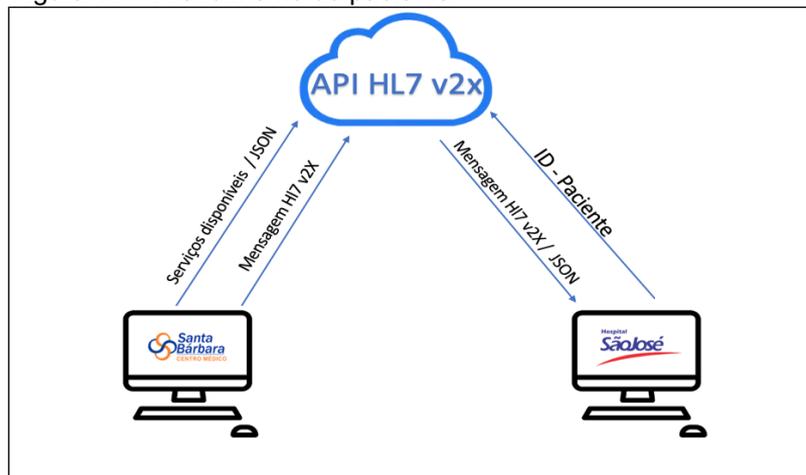
*saveMessageEvn()*, *SaveMessagnte()*, *saveMessageObr()* será passado a mensagem e o identificador do paciente, em seguida é identificado dentro do segmento o seu tipo, e armazenado os dados em uma tabela específica na base de dados.

Para identificar um paciente e encontrar as unidade de saúde que estão realizando os exames do mesmo, é necessário fazer uma requisição enviando o código do paciente para, API que vai acessar o método *show()*, que terá acesso ao segmento OBR da mensagem do pacientes recuperando o código do exame, verificado assim em uma tabela da base de dados denominada *services* onde será identificada as unidades de saúde que estão realizando os exames que foi passado para um determinado paciente.

### 5.1.2 Desenvolvimento do Protótipo

Com base na aplicação desenvolvida já com a implementação da biblioteca de conversão de mensagem, iniciou-se o desenvolvimento do protótipo, na figura 17 podemos ver a imagens da arquitetura REST usada.

Figura 17 – Atendimento do paciente.



Fonte: Do Autor (2019).

Na criação da API usando o *framework laravel*, na versão 5.8 baseada na linguagem de programação PHP.

Baseado na experiência obtida no uso normal do protótipo, foram relacionados os passos para a criação do protótipo, iniciando uma nova etapa, onde

foram aplicadas as implementações utilizadas de conversão de uma mensagem no padrão hl7.

### 5.1.2.1 Etapa de atendimento de um paciente

É nesta fase da metodologia que se explora a etapa de atendimento de um paciente para a realização de um determinado exame. Ainda nessa etapa emprega-se as técnicas de análise da mensagem enviada para a API.

O *software* para a simulação de atendimento da unidade de saúde é denominado como (Santa Bárbara Centro Médico) é um *software* desenvolvido pelo autor para a simulação de um do processo de atendimento de um paciente onde será informado o exame a ser realizado. Nela contém opções de atendimento do paciente e também para a unidade de saúde informar quais exames estão sendo realizado no momento.

Este *software* envia as informações do paciente para uma API, tornando assim as informações desse compartilhadas com as outras unidades de saúde, é um produto de código aberto desenvolvido na linguagem de programação PHP usando o *framework Laravel* e a biblioteca JQuery, nas figuras 18 e 19 podemos ver as imagens do protótipo.

Figura 18 – Atendimento do paciente.

Ficha do Paciente		
Dados do Paciente	Notas e comentários	Pedido de observação
N. do Pedido	ID do serviço universal	O Nome Textual
Prioridade	Data Solicitada	Data da Observação
Data do término da observação	Identificador do Provedor	Nome Do Provedor
Telefone		

Fonte: Do Autor (2019)

Figura 19 – informações dos exames realizado nas unidades de saúde.

Código	Hospital	Ação
3	Hospital Militar	[List] [Edit] [Delete]
4	Maria Pia	[List] [Edit] [Delete]
1	Santa Barbara	[List] [Edit] [Delete]

Fonte: Do Autor (2019).

#### 5.1.2.1 Etapa de consulta de um paciente

Nesta fase se explora a etapa da consulta de quais unidades de saúde estão realizando os exames para um determinado paciente. Continuado nessa etapa emprega-se a técnicas de algoritmo para a conversão da mensagem no padrão hl7, para que possa ser apresentada de forma legível.

O *software* para a simulação de pesquisa de atendimento é o denominado (São José Centro Médico). Nele contém uma opção onde deve ser inserido o número do paciente em atendimento, caso o mesmo seja identificado, o sistema retornará todas as informações do paciente em atendimento, em seguida as unidades de saúde que estão realizando os exames solicitados.

Este *software* foi desenvolvido de início, para a simulação de um *software* hospitalar que encontrasse quais unidades de saúde estariam realizando exames consoante o pedido do médico para um determinado paciente, é um produto de código aberto desenvolvido na linguagem de programação PHP usando o *framework* Laravel e a biblioteca JQuery, a imagem do protótipo está na figura 20.

Figura 20 – Informações do paciente

Hospital **São José** São José

Nome: Fernandes      Sobrenome: carlos R      Data de Nascimento: 3/9/2019      Cidade: Luanda  
 País: Angola      Celular: 908489968      Comentário Médico: Fazer teste completo      Exame: Raio X  
 Código de Exame: 20101171      Prioridade: baixa      Título: DOCTOR      Médico: Gian

Código hospitalar	Hospital com à realização do exame
3	Hospital Militar
4	Maria Pia
1	Santa Barbara

Fonte: Do Autor (2019).

## 5.2 RESULTADOS OBTIDOS

Com a pesquisa realizada neste trabalho e a aplicação dos fundamentos obtidos através das pesquisas bibliográfica, foi possível obter um protótipo de aplicação *web* capaz de converter mensagem hl7, armazená-la em uma base de dados e compartilhá-la com outras unidades de saúde, usando uma API (hl7-api) que foi desenvolvida pelo autor, integrando com a biblioteca de análise hl7.

Foi possível constatar que API, juntamente com a biblioteca de conversão de mensagem usada, possuem precisão e funcionamento perfeitos em conjunto com a plataforma *Web*.

Sendo assim, o processo apresentou-se ideal para uma possível utilização em unidades de saúde que apresentam dificuldades em comunicar-se umas com as outras.

Os recursos de cada segmento foram muito importantes para que isso fosse possível, campos e subcampos facilitaram o armazenamento das informações na nossa base de dados.

O protótipo se mostrou uma solução acessível e viável, além disso, é válido ressaltar que se trata de uma aplicação *web*, o que sugere uma facilidade de acesso ainda maior.

Sendo assim, os objetivos propostos foram atingidos - em suma, resultando em um protótipo de aplicativo com diversas possibilidades de aprimoramento e em uma linha de pesquisa que permite a aplicação de diversos recursos tecnológicos.

No decorrer do trabalho algumas dificuldades foram encontradas como a impossibilidade de obter os dados relacionados à quais unidades de saúde estariam realizando os exames na arquitetura do HL7, devido à falta de um segmento referente a esse assunto.

Para suprir essa necessidade, desenvolveu-se uma api em paralelo usando dois tipos de dados, o HL7 para comunicação entre as informações do paciente, e o json para comunicação entre os dados dos exames relacionados às unidades de saúde.

Desse modo a api passou a atender duas entradas de dados, mudando assim a modelagem da base de dados que passou a ter os segmentos PID, MSH, EVN, NTE e OBR usando os dados no padrão hl7 para armazenar as informações do paciente nesses segmentos, nesse caso criou-se o *services* que passou a armazenar os exames dos hospitais, recebendo uma requisição dos dados no padrão json criando uma relação de um para n podendo, assim, um hospital informar vários exames.

Com isso, no momento da pesquisa de um paciente a api responde para o sistema que solicitou um *array* com duas posições, na primeira os dados do paciente que estará estruturado no padrão hl7 e na segunda posição serão os hospitais com os exames que estão sendo realizados referente aquele paciente.

Ainda no desenvolvimento do protótipo, pensou-se em ferramentas que pudessem facilitar na construção do mesmo, desse modo obteve-se um estudo do funcionamento do framework laravel para a construção da Api e dos dois sistemas hospitalares para a simulação, a fim de entender também o funcionamento da biblioteca de análise da mensagem no padrão hl7, as ferramentas citadas usam a linguagem de programação php, que proporcionou a construção do protótipo.

## 6 CONCLUSÃO

Em se tratando da evolução da integração e alta demanda de dados, foi possível entender que determinadas áreas da saúde que armazenam informações já compartilham esses dados com outros sistemas, agilizando o processo de trabalho e na escolha da melhor alternativa a ser seguida. Este trabalho apresentou uma fundamentação sobre uma ferramenta de integração de dados entre sistemas hospitalares.

Ao transcorrer da introdução este trabalho apresentou-se pesquisas para o desenvolvimento de um protótipo de integração de dados entre sistemas hospitalares usando o padrão HL7 v2x, os dados estarão disponíveis numa base dados HL7\_INFO, que futuramente só será acessada com a autenticação das unidades de saúde. Foi apresentado um referencial teórico sobre os temas tratados assim como as peculiaridade e conceitos que foram indicados para ajudar nas técnicas para o alcance aos objetivos propostos.

O HL7 traz um diferencial interessante relacionados à troca, a integração, o compartilhamento e a recuperação de informações, para apoio da prática médica e controle administrativo, sendo desse jeito o *feedback* favorece o atendimento de um paciente.

A ferramenta *Laravel*, teve uma grande importância para construção da Api, a biblioteca hl7 contribuiu para a análise e a criação das mensagens no padrão HL7 v2x.

A execução do presente trabalho de conclusão de curso proporcionou a aquisição de conhecimentos em diversas áreas. Foi possível perceber a evolução dos padrões de compartilhamento de dados usado nos sistemas da área da saúde, e o potencial das aplicações *web* atualmente, tal como uma breve disposição dos recursos por ela suportada. A comunicação foi o principal conceito que direcionou essa pesquisa, levando a aplicação das diversas outras técnicas abordadas neste trabalho.

Este protótipo em desenvolvimento, proporciona grandes ganhos para os pacientes e médicos, na qual depois de uma consulta será possível direcionar o paciente para os hospitais que estão realizando os exames solicitados. Com isso será reduzido o tempo que um paciente leva para encontrar uma unidade de saúde que tenha o exame disponível, com o *software* em funcionamento todo esse

processo em que o paciente passa por consulta e aguarda o hospital localizar uma unidade de saúde para marcar o exame não será mais necessário, visto que, o processo de busca é imediato e instantâneo.

O trabalho facilitará o médico a obter o histórico de um paciente, sabendo assim qual foi o seu exame anterior, em qual hospital se deu e quem foi o médico que autorizou o mesmo. Com essas informações o médico poderá ter um ponto de partida em relação ao paciente. Os hospitais terão maior organização com aos pacientes visto que eles terão o acesso facilitado de seus respectivos exames.

Atualmente para realização de um exame o paciente precisa pesquisar laboratórios para obter a informação para a possível realização de um determinado exame, sendo assim, o paciente estará perdendo tempo e agravando mais a sua saúde.

Em produção a aplicação preencherá o tempo do paciente uma vez que as unidades vão disponibilizar a lista dos exames que estão sendo realizados no momento, podendo ainda atualizar a lista a qualquer momento, no atendimento o médico poderá informar qual a prioridade do paciente, para que quando chegar a uma unidade de saúde ser atendido conforme a sua prioridade indicada, assim será possível diminuir o tempo que o médico levaria para obter o histórico do paciente.

As técnicas e tecnologias abordadas foram aplicadas buscando atender um objetivo principal: criar um protótipo de aplicação *web* para as unidades de saúde que apresentam dificuldades em comunicar-se umas com as outras, visando auxiliar os técnicos da saúde para que se possam informar com precisão aos pacientes quais unidades de saúde estão realizando os exames solicitados pelo médico.

Os objetivos específicos relacionados para a execução desse trabalho foram todos atingidos, os quais foram responsáveis por orientar a pesquisa e o desenvolvimento, além de permitir que o objetivo geral do trabalho também fosse alcançado.

Como foi abordados no capítulo 5.1.2 do presente trabalho o protótipo possui duas funcionalidades principais, sendo elas o atendimento de um paciente onde será enviada uma requisição para API juntamente com a biblioteca hl7, e estas serão capazes de analisar a mensagem enviada e armazenar em uma base de dados, como última etapa, será retornada uma mensagem ao paciente, contendo todos dos dados do mesmo no formato hl7 v2x.

Como sugestão para trabalhos futuros, pode se aplicar técnicas de funcionamento em dispositivos móveis, para que os pacientes possam também ter acesso de quais unidades de saúde mais próximas estão realizando um determinado exame. Pode-se ainda atribuir diversas funcionalidades, alguns exemplos possíveis:

- a) ser capaz de reconhecer diversos padrões de comunicações entre sistemas hospitalares;
- b) aplicar métodos de autenticação para garantir segurança nas informações;
- c) criar opção de exportar a consulta realizada no protótipo e compartilhar com o paciente por meio de em um arquivo de texto;

## REFERÊNCIAS

ANDRADE, André Queiróz de. **A tomada de decisão e os sistemas de informação em saúde**. 2008. Dissertação (Mestrado em Ciência da Informação) – Universidade Federal de Minas Gerais, acesso em: 06 out. 2018.

BASTOS, Maria Pappaterra. **Sistemas de informação em saúde: o seu uso no acompanhamento de pacientes hipertensos e diabéticos: Um estudo de caso do Sistema Remédio em Casa utilizado no município do Rio de Janeiro**. (2009), acesso em: 06 out. 2018.

BATISTA. **Análise Comparativa de Padrões de Interoperabilidade em Sistemas Hospitalares**. 2016. 64 f. Curso de Ciência da Computação, – Departamento de Ciências Exatas e Engenharias, Universidade Regional do Noroeste do Estado do Rio Grande do Sul, Santa Rosa, 2016.

CANALTECH. **API**. Disponível em: <<https://canaltech.com.br/software/o-que-e-api/>>. Acesso em: 21 out. 2018.

CHRISTENSSON, Per. **Web Applications Definition**. Acesso em: 05 maio. 2018. CM Tecnologia. Brasil, Comitê Executivo de Governo Eletrônico. e **PING- Padrões de Interoperabilidade de Governo Eletrônico**. Documento de Referência. Versão 1.0. julho de 2005. Disponível em: <[www.governoeletronico.gov.br](http://www.governoeletronico.gov.br)> Acesso em 12 de março de 2019.

CMTECNOLOGIA **As vantagens e os desafios da interoperabilidade em saúde**. Disponível em: <<https://www.cmtecnologia.com.br/interoperabilidade-em-saude/>>. Acesso em: 21 out. 2018.

COSTA, Cláudio Giulliano Alves da. **Desenvolvimento e Avaliação Tecnológica de um Sistema de Prontuário Eletrônico do Paciente, Baseado nos Paradigmas da World Wide Web e da Engenharia de Software**. Campinas, 2001. Disponível em:<[http://www.medsolution.com.br/claudio/dissertacao/Dissertacao\\_Claudio\\_Giulliano\\_PEP.pdf](http://www.medsolution.com.br/claudio/dissertacao/Dissertacao_Claudio_Giulliano_PEP.pdf)>. Acesso em: 07 jun. 2019.

GUIMARÃES, Marcelo de Paiva. **Sistemas de Tempo Real**. Disponível em: <http://www.lsi.usp.br/~paiva/sd/Sistemas%20de%20Tempo%20Real.pdf>. Acesso em: 28 out. 2018.

**HEALTH LEVEL SEVEN INTERNATIONAL**. Ann Arbor (MI): HL7  
HENRIQUES, Jorge; CARVALHO, Paulo de. **HL7 Health Level Seven**. 2005. Instituto Nacional de Estatística. Inquérito à **utilização das tecnologias de informação e da comunicação nos hospitais**, Dezembro 2014. Disponível em <http://www.peprobe.com/wp-content/uploads/2014/12/16IUTICHosp.2014.pdf>, acesso em: 04 out. 2018.

International, 2014. Disponível em: < <http://www.hl7.org/index.cfm>>. Acesso em: 15 IRENE. **Mais de 60% dos hospitais públicos estão sempre superlotados.** 2014. Disponível em: <<http://g1.globo.com/bom-dia-brasil/noticia/2014/03/mais-de-60-dos-hospitais-publicos-estao-sempre-superlotados.html>>. Acesso em: 17 out. 2018.

**Linguagens de Programação.** Março de 2001. Disponível em: <<http://www.urcamp.tche.br/ccei/revista7.pdf>>. Acesso em: 20 out. 2019.

LOPES, karine petry paula marien albrecht **MODELOS PARA INTEROPERABILIDADE DE SISTEMAS HOSPITALARES UTILIZANDO PADRÃO HL7.** 2005. 189 f. Curso de Sistemas de Informação, Universidade Federal de Santa Catarina, Florianópolis, 2005. Disponível em: [http://www.inf.ufpr.br/Imperes/2017\\_2/ci167/interoperabilidadeHL7.pdf](http://www.inf.ufpr.br/Imperes/2017_2/ci167/interoperabilidadeHL7.pdf)

MARÍLIA, BAURU (Org.). **Morre menina picada por escorpião no quintal de casa no interior de SP.** 2018. Disponível em: <<https://g1.globo.com/sp/bauru-marilia/noticia/morre-menina-picada-por-escorpiao-no-quintal-de-casa-no-interior-de-sp.ghtml>>. Acesso em: 16 out. 2018.

MELLO Fabrício de Royes; LACERDA, Guilherme Silva de. **Aspectos Formais Sobre**

MIRANDA. **Estudo da padronização visando à interoperabilidade: O caso das organizações de saúde na cidade de Marília - São Paulo.** 2015. 2010 f. Monografia (Especialização) - Curso de em Ciência da Informação, Universidade Estadual Paulista, Marília, 2015.

MOREIRA, Rene. **Menina de 4 anos morre após ser picada por escorpião.** 2018. Disponível em: <<https://brasil.estadao.com.br/noticias/geral,menina-de-4-anos-morre-apos-ser-picada-por-escorpiao,70002398631>>. Acesso em: 03 nov. 2018.

MOZILLA DEVELOPER. **Introdução ao lado do servidor.** 2018. Disponível em: <[https://developer.mozilla.org/en-US/docs/Learn/Server\\_side/First\\_steps/Introduction](https://developer.mozilla.org/en-US/docs/Learn/Server_side/First_steps/Introduction)>. Acesso em: 19 maio 2018

OMS. ORGANIZAÇÃO MUNDIAL DE SAÚDE. **Classificação estatística internacional de doenças e problemas relacionados à saúde.** 10. Rev. São Paulo: CBCD, 1995. Acesso em: 06 out. 2018.

PIERRO, Bruno de. **A integração dos dados sobre saúde.** 2011. Disponível em: <<https://jornalgggn.com.br/materia-artigo/a-integracao-dos-dados-sobre-saude>>. Acesso em: 03 nov. 2018.

SANNA, Fernanda Amorim. **A IMPORTÂNCIA DA INTEGRAÇÃO DE DADOS PARA A QUALIDADE DA DECISÃO DO GESTOR PÚBLICO.** 2014. Disponível em: <[http://repositorio.enap.gov.br/bitstream/1/1940/1/Fernanda\\_Sanna\\_TCC\\_EGP9.pdf](http://repositorio.enap.gov.br/bitstream/1/1940/1/Fernanda_Sanna_TCC_EGP9.pdf)>. Acesso em: 28 out. 2018.

SCIARRA; RONDINA2. Informática em Saúde e a Interoperabilidade nos Sistemas Hospitalares. **Informática em Saúde e A Interoperabilidade nos Sistemas Hospitalares**, São Paulo, jan. 2018.

SILVA, Rafael Ferreira, **A importância da interoperabilidade**, disponível em: <http://phpbrasil.com/articles/article.php/id/851>, acesso em: 16 out. 2018.

UNIÃO, Tribunal de Contas da; UNIÃO, Tribunal de Contas da. **Mais de 60% dos hospitais públicos estão sempre superlotados**. 2014. Disponível em: <http://g1.globo.com/bom-dia-brasil/noticia/2014/03/mais-de-60-dos-hospitais-publicos-estao-sempre-superlotados.html>. Acesso em: 03 nov. 2018.

W3C et al (Org.). **Abstract**. Disponível em: <https://www.w3.org/TR/push-api/>. Acesso em: 16 out. 2018.

YOUNG. **Sistemas de Tempo Real**. 1982. Disponível em: <http://home.ufam.edu.br/lucascordeiro/str/slides/01-introducao-sistemas-de-tempo-real.pdf>. Acesso em: 03 nov. 2018.

## APÊNDICE A – ARTIGO CIENTÍFICO

**INTEGRAÇÃO DE DADOS PARA COMPARTILHAMENTO DE INFORMAÇÕES  
RELACIONADAS AOS SERVIÇOS DISPONÍVEIS EM UNIDADES DE SAÚDE****Carlos Rafael Magalhães Fernandes<sup>1</sup>, Gilberto Vieira da Silva<sup>2</sup>**<sup>1</sup> Curso de Ciência da Computação

Universidade do Extremo Sul Catarinense (UNESC) – Criciúma, SC – Brazil

{carlosr.m.fernandes, gilbert.vieira.silva}@gmail.com

**Abstract.** *This article aims to develop a web application prototype for information sharing among health facilities that have difficulties in communicating. In order to help patients, locate which health facilities are providing the tests they need to do. For this, a theoretical foundation was made regarding the techniques of information sharing between health centers and communication methodologies in web applications. With the knowledge gained from the research, it was possible to develop a web application prototype, using the Laravel tool to create an API, the PHP-based HL7 v2.x analysis, message generation and analysis library, inspired by the Perl package. Net-HL7, for analysis and message creation in the hl7 v2.x standard.*

**Resumo.** *Este artigo tem como objetivo desenvolver um protótipo de aplicação web para compartilhamento de informações entre unidades de saúde que apresentam dificuldades em comunicar-se. Com o intuito máximo de auxiliar pacientes a localizarem quais unidades de saúde estão disponibilizando os exames que os mesmos precisam fazer. Para isso, foi realizada uma fundamentação teórica referente às técnicas de compartilhamento de informações entre as unidades de saúde e metodologias de comunicação em aplicações web. Com os conhecimentos obtidos a partir da pesquisa, foi possível elaborar um protótipo de aplicação web, utilizando a ferramenta Laravel para criação de uma API, a biblioteca de análise, geração e envio de mensagem HL7 v2.x baseado em PHP, inspirado no pacote Perl Net-HL7, para análise e criação da mensagem no padrão hl7 v2.x..*

**1. INTRODUÇÃO**

O problema de comunicação sempre acarreta um custo, que gera a baixa produtividade e a perda de tempo, em função disso, as tecnologias de informação existentes colaboram para a complexidade da comunicação, para tal, necessita-se de maior integração entre as tecnologias que abrangem os sistemas de informações.

nas unidades de saúde, principalmente pública, há ainda inúmeras dificuldades no sistema de informação, conseqüentemente acabam afetando diretamente os técnicos de saúde, que buscam por informações de quais medicamentos, equipamentos e serviços estão disponíveis na sua unidade de saúde ou em outras para encaminhamento de paciente.

O surgimento, bem como início da história da informática, é visto como um passo fundamental para se alcançar a era do conhecimento, Sendo possível armazenar dados, é relevante que se possa compartilhá-los com outros sistemas, neste sentido, diferentes mecanismos foram desenvolvidos para garantir a comunicação entre os vários tipos de sistemas, envolvendo dispositivos, sistemas operacionais, aplicativos, bases de dados e deste modo, foram projetados padrões como o HL7, CCR a nível internacional.

O HL7 tem o objetivo de definir normas para a transmissão de dados, não restringe o sistema operacional, nem a linguagem de programação a ser utilizada para a troca de informação, permitindo ainda que o implementador defina suas próprias mensagens, o que mais uma vez, torna-o flexível.

Este trabalho tem como objetivo utilizar tecnologias de integração de dados, para a elaboração de um protótipo de aplicação *web*, voltado para o apoio das unidades de saúde que têm dificuldade em comunicar-se entre si, com o intuito de facilitar o acesso à informação como medicamentos, consultas e situações dos equipamentos.

## 2. METODOLOGIA

Para a concretização deste trabalho foi realizado o levantamento bibliográfico relacionado às unidades de saúde que apresentam dificuldade em comunicar-se umas às outras. Sendo assim, foi proposto o desenvolvimento de um protótipo de uma aplicação *Web* para a integração de dados para o compartilhamento de serviços disponíveis nas unidades de saúde.

As operações relacionadas a integração de dados será responsabilidade da API a ser desenvolvida com integração de uma biblioteca de análise para gerar, enviar e criar uma mensagem no padrão hl7, já na implementação será usando a linguagem de programação PHP e ECMAScript. Primeiramente será feito uma análise do funcionamento dos sistemas de integração de dados existentes e em seguida será feito uma outra análise com relação ao envio das informações para que as outras unidades de saúde possam ter aceso, após essa análise será dado início ao desenvolvimento do protótipo.

### 2.1 A BIBLIOTECA PHP PARA ANALISAR, GERAR E ENVIAR MENSAGENS HL7 V2.X.

As mensagens HL7 v2.x usam uma sintaxe para sua codificação que não é baseada em XML, mas sim em segmentos (linhas) e delimitadores de caracteres. Os segmentos possuem campos separados pelo delimitador. Um campo composto pode ter componentes separados pelos subcomponentes.

Os delimitadores padrão são barra vertical ou pipe ( | ) serve como separador de campos, o caractere ( ^ ) corresponde com separador de componentes e o e comercial ( & ) como separador de subcomponentes. O caractere de til ( ~ ) é o separador de repetições. Todo segmento começa com uma *string* de 3 caracteres que identifica seu tipo.

A biblioteca tem como princípio básico de funcionamento receber uma mensagem no padrão hl7 v2.x, e submetê-la ao processo de decodificação da mensagem e armazená-la em uma base de dados preparada com os segmento do hl7v2.x.

O reconhecimento dos dados é acessado por meio de classe *Message*, que proporciona a capacidade de reconhecimento da mensagem e responder apropriadamente. Geralmente, é usado o construtor da classe para criar um novo objeto *Message*.

Para os serviços disponíveis em unidades de saúde, não existe um segmento com o protocolo de mensagem hl7v2, então optou-se no uso do *JavaScript Object Notation* por ser um padrão de fácil compreensão que a maioria dos sistemas usam.

Com isso os dados relacionados ao serviço disponível, são enviados para a API no formato JSON onde será informado o nome do hospital, o código do exame universal e o status do exame, onde a classe *ServicesAvailableHospitals* fará todo processo, para armazenar na base de dados e disponibilizar para outras unidades de saúde.

### 2.1.1 Demonstração de análise e criação da mensagem hl7

Com a biblioteca de análise, geração e envio de mensagem HL7 v2.x baseado em PHP, inspirado no pacote Perl Net-HL7, conseguimos ver alguns exemplos de representação de análise e criação de uma mensagem.

Na figura 14 há uma representação de análise de uma mensagem no padrão HL7 v2X.

Figura 14 – Representação de análise de mensagem.

```

$msg = new Message("MSH|^~\&|1|\rPID|||abcd|\r");
$pid = $msg->getSegmentByIndex(1);
echo $pid->getField(3);
echo $msg->toString(true);
$msg->getFirstSegmentInstance('PID');
$msg->hasSegment('PID');

```

Fonte: Do Autor (2019).

### 2.1.2 Construção da API

O *hl7-api* foi desenvolvido usando o *framework laravel*, por ser rápido, livre, de código aberto, e por usar o padrão *Model, View, Controller* (MVC), mais para a construção da api usou-se simplesmente o *model* juntamente com o *Eloquent* do laravel que facilitou nas consultas e manipulação dos dados na base de dados, e o *controller* que foi o responsável por receber todas requisições do usuário, instanciando assim a classe *Message* onde será validade a mensagem da enviada.

O *laravel* mostrou-se uma tecnologia propícia para o desenvolvimento em aplicações *web*, visto que ela é gratuita e possui uma ótima documentação e uma boa comunidade. E um dos pontos que fez com que o *laravel* fosse selecionado, foi por ser o *framework* PHP mais usado atualmente (TRENDS, 2019).

O mesmo usa o *composer* para gerenciar suas dependências, o *composer* é uma ferramenta para gerenciamento de dependências em PHP. Ele permite que se declarem as bibliotecas das quais seu projeto depende e as gerencia (instala / atualiza), o *composer* requer o PHP 5.3.2+ para que seja executado, algumas configurações sensíveis de php e sinalizadores de compilação também são necessárias. (COMPOSER,2019).

Para realizar a conversão dos dados com a API usando a biblioteca, se faz necessário seguir alguns passos básicos. Com as instâncias das classes *SaveMessagepid*, *SaveMessagehsh*, *SaveMessageevn*, *SaveMessagnte* e *SaveMessageObr* criadas, é necessário chamar os métodos *saveMessagepid()*, passando com parâmetro a mensagem já para o os

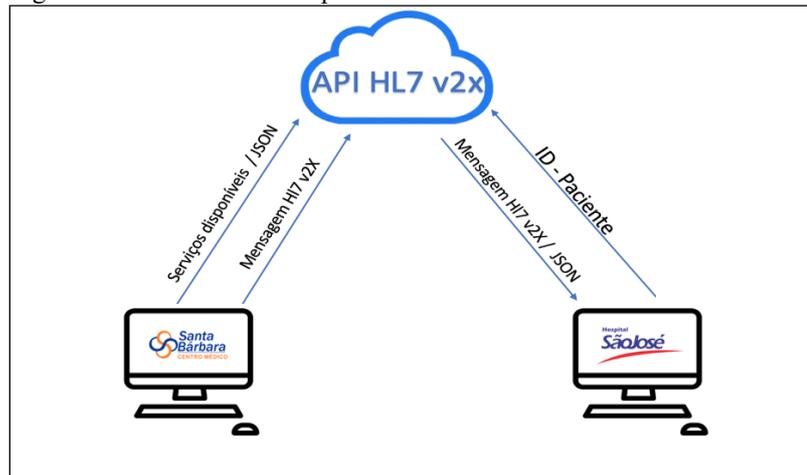
métodos *saveMessageMsh()*, *saveMessageEvn()*, *SaveMessagnte()*, *saveMessageObr()* será passado a mensagem e o identificador do paciente, em seguida é identificado dentro do segmento o seu tipo, e armazenado os dados em uma tabela especifica na base de dados.

Para identificar um paciente e encontrar as unidade de saúde que estão realizando os exames do mesmo, é necessário fazer uma requisição enviando o código do paciente para, API que vai acessar o método *show()*, que terá acesso ao segmento OBR da mensagem do pacientes recuperando o código do exame, verificado assim em uma tabela da base de dados denominada *services* onde será identificada as unidades de saúde que estão realizando os exames que foi passado para um determinado paciente

### 2.1.3 Desenvolvimento do Protótipo

Com base na aplicação desenvolvida já com a implementação da biblioteca de conversão de mensagem, iniciou-se o desenvolvimento do protótipo, na figura 17 podemos ver a imagens da arquitetura REST usada.

Figura 17 – Atendimento do paciente.



Fonte: Do Autor (2019).

Na criação da API usando o *framework laravel*, na versão 5.8 baseada na linguagem de programação PHP.

Baseado na experiência obtida no uso normal do protótipo, foram relacionados os passos para a criação do protótipo, iniciando uma nova etapa, onde foram aplicadas as implementações utilizadas de conversão de uma mensagem no padrão hl7.

## 2.2 ETAPA DE ATENDIMENTO DE UM PACIENTE

É nesta fase da metodologia que se explora a etapa de atendimento de um paciente para a realização de um determinado exame. Ainda nessa etapa emprega-se as técnicas de análise da mensagem enviada para a API.

O *software* para a simulação de atendimento da unidade de saúde é denominado como (Santa Bárbara Centro Médico) é um *software* desenvolvido pelo autor para a simulação de um do processo de atendimento de um paciente onde será informado o exame a ser realizado. Nela contém opções de atendimento do paciente e também para a unidade de saúde informar quais exames estão sendo realizado no momento.

Este *software* envia as informações do paciente para uma API, tornando assim as informações desse compartilhadas com as outras unidades de saúde, é um produto de código aberto desenvolvido na linguagem de programação PHP usando o *framework Laravel* e a biblioteca JQuery, nas figuras 18 e 19 podemos ver as imagens do protótipo

Figura 18 – Atendimento do paciente.

The screenshot shows a web application interface for patient records. A modal window titled "Ficha do Paciente" is open, displaying a form with the following fields:

- N. do Pedido: Text input field.
- ID do serviço universal: Text input field.
- O Nome Textual: Text input field.
- Prioridade: Dropdown menu.
- Data Solicitada: Date input field (format: dd/mm/aaaa).
- Data da Observação: Date input field (format: dd/mm/aaaa).
- Data do término da observação: Date input field (format: dd/mm/aaaa).
- Identificador do Provedor: Text input field.
- Nome Do Provedor: Text input field.
- Telefone: Text input field.

At the bottom right of the form are two buttons: "Fechar" (grey) and "Salvar" (blue).

Fonte: Do Autor (2019)

Figura 19 – informações dos exames realizado nas unidades de saúde.

The screenshot shows a web application interface for health centers. A table titled "Centro de Saúde" is displayed, with a "+ Novo Centro de Saúde" button in the top right corner. The table has the following structure:

Código	Hospital	Ação
3	Hospital Militar	[Menu] [Edit] [Delete]
4	Maria Pia	[Menu] [Edit] [Delete]
1	Santa Barbara	[Menu] [Edit] [Delete]

Fonte: Do Autor (2019).

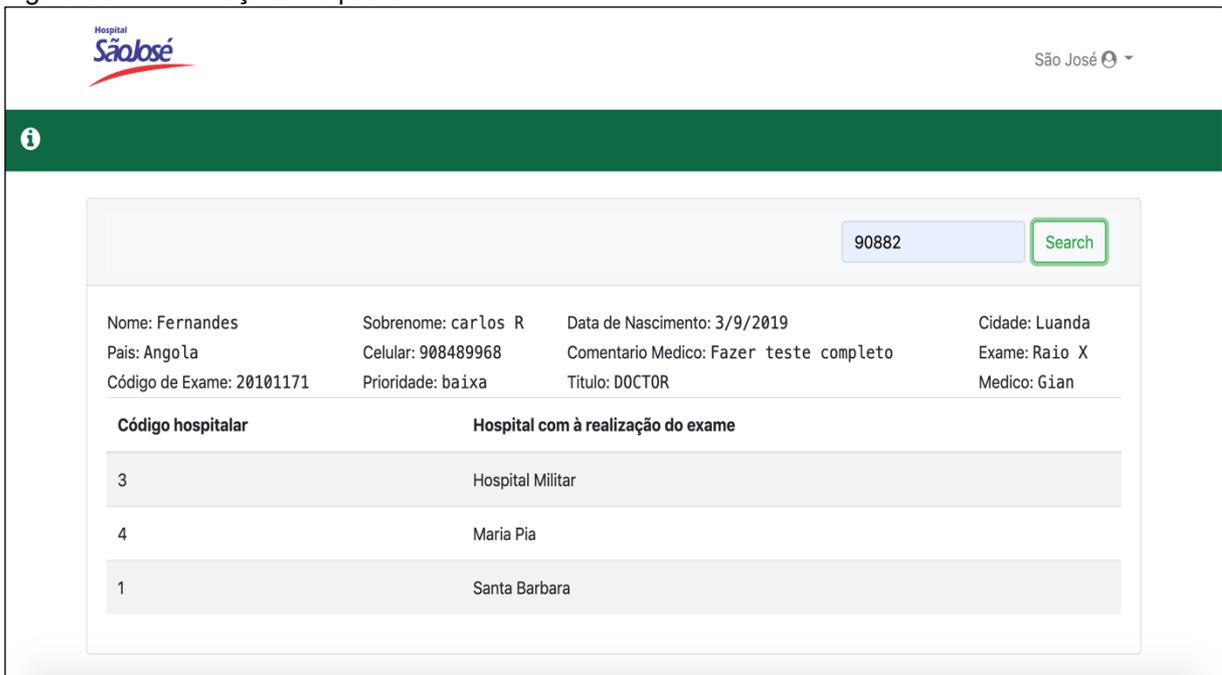
## 2.3 ETAPA DE CONSULTA DE UM PACIENTE

Nesta fase se explora a etapa da consulta de quais unidades de saúde estão realizando os exames para um determinado paciente. Continuado nessa etapa emprega-se a técnicas de algoritmo para a conversão da mensagem no padrão hl7, para que possa ser apresentada de forma legível.

O *software* para a simulação de pesquisa de atendimento é o denominado (São José Centro Médico). Nele contém uma opção onde deve ser inserido o número do paciente em atendimento, caso o mesmo seja identificado, o sistema retornará todas as informações do paciente em atendimento, em seguida as unidades de saúde que estão realizando os exames solicitados.

Este *software* foi desenvolvido de início, para a simulação de um *software* hospitalar que encontrasse quais unidades de saúde estariam realizando exames consoante o pedido do médico para um determinado paciente, é um produto de código aberto desenvolvido na linguagem de programação PHP usando o *framework* Laravel e a biblioteca JQuery, a imagem do protótipo está na figura 20.

Figura 20 – Informações do paciente



Hospital São José

São José

90882 Search

Nome: Fernandes	Sobrenome: carlos R	Data de Nascimento: 3/9/2019	Cidade: Luanda
Pais: Angola	Celular: 908489968	Comentário Medico: Fazer teste completo	Exame: Raio X
Código de Exame: 20101171	Prioridade: baixa	Título: DOCTOR	Medico: Gian

Código hospitalar	Hospital com à realização do exame
3	Hospital Militar
4	Maria Pia
1	Santa Barbara

Fonte: Do Autor (2019).

### 3 RESULTADOS OBTIDOS

Com a pesquisa realizada neste trabalho e a aplicação dos fundamentos obtidos através das pesquisas bibliográfica, foi possível obter um protótipo de aplicação *web* capaz de converter mensagem hl7, armazená-la em uma base de dados e compartilhá-la com outras unidades de saúde, usando uma API (hl7-api) que foi desenvolvida pelo autor, integrando com a biblioteca de análise hl7.

Foi possível constatar que API, juntamente com a biblioteca de conversão de mensagem usada, possuem precisão e funcionamento perfeitos em conjunto com a plataforma *Web*.

Sendo assim, o processo apresentou-se ideal para uma possível utilização em unidades de saúde que apresentam dificuldades em comunicar-se umas com as outras.

Os recursos de cada segmento foram muito importantes para que isso fosse possível, campos e subcampos facilitaram o armazenamento das informações na nossa base de dados.

O protótipo se mostrou uma solução acessível e viável, além disso, é válido ressaltar que se trata de uma aplicação *web*, o que sugere uma facilidade de acesso ainda maior.

Sendo assim, os objetivos propostos foram atingidos - em suma, resultando em um protótipo de aplicativo com diversas possibilidades de aprimoramento e em uma linha de pesquisa que permite a aplicação de diversos recursos tecnológicos.

No decorrer do trabalho algumas dificuldades foram encontradas como a impossibilidade de obter os dados relacionados à quais unidades de saúde estariam realizando os exames na arquitetura do HL7, devido à falta de um segmento referente a esse assunto.

Para suprir essa necessidade, desenvolveu-se uma api em paralelo usando dois tipos de dados, o HL7 para comunicação entre as informações do paciente, e o json para comunicação entre os dados dos exames relacionados às unidades de saúde.

Desse modo a api passou a atender duas entradas de dados, mudando assim a modelagem da base de dados que passou a ter os segmentos PID, MSH, EVN, NTE e OBR usando os dados no padrão hl7 para armazenar as informações do paciente nesses segmentos, nesse caso criou-se o *services* que passou a armazenar os exames dos hospitais, recebendo uma requisição dos dados no padrão json criando uma relação de um para n podendo, assim, um hospital informar vários exames.

Com isso, no momento da pesquisa de um paciente a api responde para o sistema que solicitou um *array* com duas posições, na primeira os dados do paciente que estará estruturado no padrão hl7 e na segunda posição serão os hospitais com os exames que estão sendo realizados referente aquele paciente.

Ainda no desenvolvimento do protótipo, pensou-se em ferramentas que pudessem facilitar na construção do mesmo, desse modo obteve-se um estudo do funcionamento do framework laravel para a construção da Api e dos dois sistemas hospitalares para a simulação, a fim de entender também o funcionamento da biblioteca de análise da mensagem no padrão hl7, as ferramentas citadas usam a linguagem de programação php, que proporcionou a construção do protótipo.

## 4 CONCLUSÃO

Em se tratando da evolução da integração e alta demanda de dados, foi possível entender que determinadas áreas da saúde que armazenam informações já compartilham esses dados com outros sistemas, agilizando o processo de trabalho e na escolha da melhor alternativa a ser seguida. Este trabalho apresentou uma fundamentação sobre uma ferramenta de integração de dados entre sistemas hospitalares.

Ao transcorrer da introdução este trabalho apresentou-se pesquisas para o desenvolvimento de um protótipo de integração de dados entre sistemas hospitalares usando o padrão HL7 v2x, os dados estarão disponíveis numa base dados HL7\_INFO, que futuramente só será acessada com a autenticação das unidades de saúde. Foi apresentado um referencial teórico sobre os temas tratados assim como as peculiaridade e conceitos que foram indicados para ajudar nas técnicas para o alcance aos objetivos propostos.

O HL7 traz um diferencial interessante relacionados à troca, a integração, o compartilhamento e a recuperação de informações, para apoio da prática médica e controle administrativo, sendo desse jeito o *feedback* favorece o atendimento de um paciente.

A ferramenta *Laravel*, teve uma grande importância para construção da Api, a biblioteca hl7 contribuiu para a análise e a criação das mensagens no padrão HL7 v2x.

A execução do presente trabalho de conclusão de curso proporcionou a aquisição de conhecimentos em diversas áreas. Foi possível perceber a evolução dos padrões de compartilhamento de dados usado nos sistemas da área da saúde, e o potencial das aplicações

*web* atualmente, tal como uma breve disposição dos recursos por ela suportada. A comunicação foi o principal conceito que direcionou essa pesquisa, levando a aplicação das diversas outras técnicas abordadas neste trabalho.

Este protótipo em desenvolvimento, proporciona grandes ganhos para os pacientes e médicos, na qual depois de uma consulta será possível direcionar o paciente para os hospitais que estão realizando os exames solicitados. Com isso será reduzido o tempo que um paciente leva para encontrar uma unidade de saúde que tenha o exame disponível, com o *software* em funcionamento todo esse processo em que o paciente passa por consulta e aguarda o hospital localizar uma unidade de saúde para marcar o exame não será mais necessário, visto que, o processo de busca é imediato e instantâneo.

O trabalho facilitará o médico a obter o histórico de um paciente, sabendo assim qual foi o seu exame anterior, em qual hospital se deu e quem foi o médico que autorizou o mesmo. Com essas informações o médico poderá ter um ponto de partida em relação ao paciente. Os hospitais terão maior organização com aos pacientes visto que eles terão o acesso facilitado de seus respectivos exames.

Atualmente para realização de um exame o paciente precisa pesquisar laboratórios para obter a informação para a possível realização de um determinado exame, sendo assim, o paciente estará perdendo tempo e agravando mais a sua saúde.

Em produção a aplicação preencherá o tempo do paciente uma vez que as unidades vão disponibilizar a lista dos exames que estão sendo realizados no momento, podendo ainda atualizar a lista a qualquer momento, no atendimento o médico poderá informar qual a prioridade do paciente, para que quando chegar a uma unidade de saúde ser atendido conforme a sua prioridade indicada, assim será possível diminuir o tempo que o médico levaria para obter o histórico do paciente.

As técnicas e tecnologias abordadas foram aplicadas buscando atender um objetivo principal: criar um protótipo de aplicação *web* para as unidades de saúde que apresentam dificuldades em comunicar-se umas com as outras, visando auxiliar os técnicos da saúde para que se possam informar com precisão aos pacientes quais unidades de saúde estão realizando os exames solicitados pelo médico.

Os objetivos específicos relacionados para a execução desse trabalho foram todos atingidos, os quais foram responsáveis por orientar a pesquisa e o desenvolvimento, além de permitir que o objetivo geral do trabalho também fosse alcançado.

Como foi abordados no capítulo 5.1.2 do presente trabalho o protótipo possui duas funcionalidades principais, sendo elas o atendimento de um paciente onde será enviada uma requisição para API juntamente com a biblioteca hl7, e estas serão capazes de analisar a mensagem enviada e armazenar em uma base de dados, como última etapa, será retornada uma mensagem ao paciente, contendo todos dos dados do mesmo no formato hl7 v2x.

Como sugestão para trabalhos futuros, pode se aplicar técnicas de funcionamento em dispositivos móveis, para que os pacientes possam também ter acesso de quais unidades de saúde mais próximas estão realizando um determinado exame. Pode-se ainda atribuir diversas funcionalidades, alguns exemplos possíveis: ser capaz de reconhecer diversos padrões de comunicações entre sistemas hospitalares, aplicar métodos de autenticação para garantir segurança nas informações, criar opção de exportar a consulta realizada no protótipo e compartilhar com o paciente por meio de em um arquivo de texto.

## 5. REFERÊNCIAS

SCIARRA; RONDINA2. Informática em Saúde e a Interoperabilidade nos Sistemas Hospitalares. **Informática em Saúde e A Interoperabilidade nos Sistemas Hospitalares**, São Paulo, jan. 2018.

LOPES, karine petry paula marien albrecht **MODELOS PARA INTEROPERABILIDADE DE SISTEMAS HOSPITALARES UTILIZANDO PADRÃO HL7**. 2005. 189 f. Curso de Sistemas de Informação, Universidade Federal de Santa Catarina, Florianópolis, 2005. Disponível em:  
[http://www.inf.ufpr.br/Imperes/2017\\_2/ci167/interoperabilidadeHL7.pdf](http://www.inf.ufpr.br/Imperes/2017_2/ci167/interoperabilidadeHL7.pdf)