ESCOLA SENAI

“PROF. DR. EURYCLIDES DE JESUS ZERBINI”

Vinicius Fernando Piantoni

Vinícius Raphael Ferreira Cruz da Silva

Felipe Carradori de Oliveira

Kauan Silva de Sousa

**1st Help**

Campinas SP

2020Vinicius Fernando Piantoni

Vinícius Raphael Ferreira Cruz da Silva

Felipe Carradori de Oliveira

Kauan Silva de Sousa

**1st Help**

Projeto apresentado à Escola SENAI “Prof. Dr. Euryclides de Jesus Zerbini” para obtenção do certificado de conclusão do Curso Técnico de Desenvolvimento de Sistemas.

Orientador: Fernando José Ignácio e Paulo Henrique Pansani

Vinicius Fernando Piantoni

Vinícius Raphael Ferreira Cruz da Silva

Felipe Carradori de Oliveira

Kauan Silva de Sousa

**1st Help**

Trabalho de conclusão de curso aprovado como requisito parcial para obtenção do grau de técnico, do curso Técnico de Desenvolvimento de Sistemas da Escola SENAI “Prof. Dr. Euryclides de Jesus Zerbini”.

**BANCA EXAMINADORA**

1º Examinador

2º Examinador

3º Examinador

Campinas SP

2020

**DEDICATÓRIA**

Dedicamos esse trabalho primeiramente para nossa família, que sempre nos ofereceu todo suporte possível para que realizássemos o curso, dedicamos também aos nossos professores que foram responsáveis por gerenciar e aplicar todo conteúdo do curso Técnico de Desenvolvimento de Sistemas e ao grupo que teve grande empenho e disposição para realizar o projeto de maneira respeitosa e agradável.

**AGRADECIMENTOS**

Temos um sentimento intenso de agradecimento pela escola SENAI Prof. Dr. Euryclides de Jesus Zerbini que nos proporcionou uma oportunidade de evoluir e aprender.

Agradecemos aos nossos professores que atribuíram um bom ensino para nosso crescimento profissional, além, de sempre buscar a melhor maneira e forma de tirar nossas dúvidas e aplicar o ensinamento de maneira correta.

E também prestamos nossos agradecimentos aos nossos companheiros de aprendizagem, que sempre mantiveram um ambiente agradável e divertido para aprender.

**EPÍGRAFE**

**“Sonhos determinam o que você quer. Ação determina o que você conquista.”**

**- Aldo Novak**

**SUMÁRIO**

[1 INTRODUÇÃO 10](#_Toc14160040)

[2 JUSTIFICATIVA 11](#_Toc14160041)

[3 OBJETIVOS 14](#_Toc14160042)

[3.1. Objetivos Gerais 14](#_Toc14160043)

[3.2. Objetivos Específicos 14](#_Toc14160044)

[4 PRODUCT BACKLOG 15](#_Toc14160045)

[5 REQUISITOS NÃO FUNCIONAIS 15](#_Toc14160046)

[6 PREMISSAS 16](#_Toc14160047)

[7 RESTRIÇÕES 17](#_Toc14160048)

[8 ANÁLISE DE RISCOS DE UM PROJETO 19](#_Toc14160049)

[8.1. Nível e Planos de Ação para os Riscos 19](#_Toc14160050)

[8.2. Planos de ação 20](#_Toc14160051)

[9 SPRINTS 22](#_Toc14160052)

[9.1. Primeiro Sprint 22](#_Toc14160053)

[9.1.1. Product Backlog 22](#_Toc14160054)

[9.1.2. Sprint Backlog 22](#_Toc14160055)

[9.1.3. Burn Down Chart 23](#_Toc14160056)

[9.1.4. Diagramas 23](#_Toc14160057)

[9.1.5. Plano de testes 23](#_Toc14160058)

[9.1.5.1. Resultados 23](#_Toc14160059)

[9.1.6. Kanban e Retrospectiva 23](#_Toc14160060)

[10 Modelo de Dados 24](#_Toc14160061)

[10.1. Diagrama de Entidade e Relacionamento 24](#_Toc14160062)

[10.2. Modelo lógico do banco de dados 24](#_Toc14160063)

[10.3. Dicionário de dados 24](#_Toc14160064)

[11 PRINCIPAIS TELAS DO SISTEMA 25](#_Toc14160065)

[12 CONCLUSÃO 26](#_Toc14160066)

[12.1. Escreva os resultados obtidos 26](#_Toc14160067)

[12.2. Constatações 26](#_Toc14160068)

[12.3. Sugestões de possíveis aperfeiçoamentos técnicos 26](#_Toc14160069)

[13 REFERÊNCIAS 27](#_Toc14160070)

[14 GLOSSÁRIO 28](#_Toc14160071)

[15 ANEXOS 29](#_Toc14160072)

# INTRODUÇÃO

Atualmente as pessoas que precisam de uma determinada ajuda médica passam por diversas dificuldades e barreiras para que o atendimento seja realizado com sucesso e de maneira positiva. Gastam muito tempo indo em vários hospitais por falta de informações sobre disponibilidade de médicos, leitos e equipamentos. Há também a questão de, se o paciente precisar de uma ambulância, a mesma normalmente tende a demorar.

Nossa plataforma online disponibiliza para o usuário que solicite rapidamente a ajuda de um hospital, assim, o hospital recebe uma requisição com os dados do paciente e sua localização, desta forma, eles verificaram se contém disponibilidade de atendê-lo, caso haja, o hospital separa os procedimentos necessários para o atendimento e se necessário solicitam uma ambulância. Economizando tempo para atendimento do paciente e tendo noção dos casos que estão por vir no hospital.

Há aplicativos que podem complementar nosso sistema. Por exemplo, o aplicativo do sus “Meu digiSUS” tem funcionalidades como: agendar uma consulta, lista de vacinas recebidas pelo paciente e poder adicionar aos favoritos os hospitais por perto. Outro exemplo é o “Agenda Fácil” que tem a principal função fazer agendamentos para consultas e exames na rede municipal de saúde de São Paulo. Entretanto, esses aplicativos não tem uma comunicação em tempo real do paciente com o hospital ou ao contrário. Com um sistema que permite essa comunicação em tempo real, diminuiria o tempo de espera dos pacientes e os mesmos não iam precisar fazer tantas transferências em vários hospitais.

# JUSTIFICATIVA

Através de uma pesquisa de campo, foi detectado um problema de comunicação entre os hospitais, que tem como base o alarde com o atendimento ao paciente. Ele se baseia com a demora de verificação de disponibilidade e recursos para executar o devido atendimento.

*Figura 1 – Dados da pesquisa de campo*

O que contribuiu para as respostas “sim”, foi o fato de tentar passar por um determinado atendimento, o principal critério utilizado pelos consultados foi o intenso desgaste que passam no processo de atendimento (a intensa demora; verificação se o hospital tem recursos necessários para atendê-los; se há médicos à disposição).

*Figura 2 – Dados da pesquisa de campo*

A dificuldade para achar os equipamentos necessários para o atendimento com o paciente foi intensa. A maioria dos relatos apresentados foi o transtorno de ficar se locomovendo entre os hospitais que tem os recursos necessários.

Por conta desses dados, com o intuito de melhorar o atendimento e comunicação (hospitais com outros hospitais e pacientes com hospitais), foi desenvolvido um aplicativo mobile e web, no qual o paciente deixa de enfrentar os problemas de transição entre hospitais e cada hospital sabe se pode ou não suportar o paciente que solicitará o atendimento. Tudo isso por conta de um gerenciamento que os hospitais possuem que é capaz de receber todos os dados necessários do usuário e obter uma prévia do tipo de atendimento que será necessário, com isto, o hospital poderá aceitar a requisição de atendimento do usuário. Com a aceitação positiva da requisição, o paciente será informado ao devido hospital que ele irá e automaticamente ele fica adicionado na lista de espera, ou seja, economizando tempo e facilitando para quem precisa de atendimento.

# OBJETIVOS

A seguir será apresentado os objetivos gerais exibindo resumidamente a ideia central e os objetivos específicos representando as fases para o desenvolvimento do sistema.

# Objetivos Gerais

Desenvolver um sistema mobile e web de conversação entre hospitais e pacientes, melhorando a agilidade no atendimento dos hospitais e uma melhor organização.

# Objetivos Específicos

*Web*:

* Integrar *front-end* com *back-end* através de web service;
* Desenvolver *front-end* em javascript (reactjs);
* Realizar teste unitário;

*Mobile*:

* Desenvolver *Web Service*;
* Desenvolver *front-end* em javascript (react-native-js);
* Integrar *front-end* com *back-end* através de web service;
* Armazenar os dados, se logado, no armazenamento interno do celular;

*Back-end:*

* Criar banco de dados através do mongoDB;
* Criar CRUD do hospital e paciente;
* Realizar autenticação do hospital e paciente;
* Login e logout de hospital e paciente;
* Criar rotas de navegação;
* Desenvolver o back-end em javascript (nodejs);
* Enviar imagens cadastradas para o servidor da AmazonS3
* Realizar a busca de hospitais em torno de 10km;

Testes:

* Testar todas as funcionalidades do sistema *web*, *mobile* e *back-end*;
* Procurar possíveis erros no sistema.
* Procurar possíveis dificuldades de usabilidade para o usuário.

# PRODUCT BACKLOG

O produto *backlog* refere-se as funcionalidades que o software deverá possuir, de forma atender as necessidades do cliente.

* Realizar cadastro paciente e hospital;
* Realizar login do paciente hospital;
* Realizar logout do paciente e hospital;
* Realizar pesquisa de hospitais próximos em 10km;
* Realizar instância de dados do paciente;
* Transferência de requisições entre os hospitais;

# REQUISITOS NÃO FUNCIONAIS

Os requisitos não funcionais se referem às propriedades do sistema, não definido suas funcionalidades, mas suas peculiaridades.

* Desenvolver um layout responsivo;
* Renderização de mapa;
* Listagem de usuários e hospitais;
* Desenvolver um layout autodidata;
* Splash screen no mobile;

# PREMISSAS

Premissas são os fatores associados ao escopo do projeto que, para fins de planejamento, são assumidos como verdadeiros, reais ou certos sem a necessidade de prova ou demonstração. Ou seja, são hipóteses ou pressupostos.

* O projeto será realizado por quatro pessoas;
* O projeto será realizado em todos os dias úteis da semana, exceto feriados e mudanças de cronograma escolar/pessoal.
* O projeto será realizado no ambiente escolar e fora dele;
* Serão disponibilizados professores para maiores dúvidas e conselhos;
* O projeto tem previsão de termino no fim de maio no ano de 2020;
* Será disponibilizado um computador no ambiente escolar por integrante do grupo
* Os sistemas web, mobile e back-end serão programados em *ReactJS, React Native e NodeJS*, respectivamente;
* O mobile estará disponível tanto para celulares *IOS* quanto *Android*;
* Será utilizado o programa *Visual Studio Code* para programação em javascript.
* Será utilizado o *mongoDB* para o armazenamento dos dados da plataforma e *awsS3* para armazenamento de imagens;
* A plataforma depende de conexão à internet;
* O sistema depende de ligação ao banco de dados.

# RESTRIÇÕES

Restrições são limitações impostas interna ou externamente ao projeto. Restrições podem ser a obrigatoriedade de se executar determinadas ferramentas e a forma de trabalho da equipe.

**Restrições Técnicas:**

**Banco de dados:**

* Os modelos de banco de dados devem ser elaborados através do Visio;
* O sistema back-end deverá ser feito através do *Visual Studio Code;*
* O back-end será programado em javascript (nodeJS);
* O modelo conceitual do projeto deve ser desenvolvido inicialmente;
* O modelo lógico será desenvolvido após a construção do modelo conceitual;
* O modelo físico deve ser iniciado após a finalização dos modelos acima;
* O banco de dados deve ser produzido através do *mongoDB*;
* Todos os testes serão efetuados todos os dias antes do término da aula;
* O teste unitário será efetuado no final de cada Sprint;

**Plataforma Web:**

* O sistema web deverá ser feito através do *Visual Studio Code*;
* Para o *front-end* da web utilizaremos a linguagem *javascript (reactjs)*;
* A conexão com o banco externo deve ser feita por *web service (axios)*;
* O teste unitário será efetuado no final de cada Sprint;

**Plataforma Mobile:**

* No *front-end* mobile deverá seguir o modelo feito no Adobe XD;
* O sistema mobile deverá ser feito através do *Visual Studio Code*;
* Para o *front-end* da web utilizaremos a linguagem *javascript (React Native JS)*;
* A conexão com o banco externo deve ser feita por *WebService (axios)*;
* O teste unitário será efetuado no final de cada Sprint;

**Restrições Administrativas:**

* Verificar as tarefas para o desenvolvimento do projeto no *Trello;*
* Compartilhamento e extração dos conteúdos através da plataforma GitHub;
* Realizar documentação do sistema utilizando Microsoft Word 2016;

**Restrições finais:**

* O sistema deve ser entregue no final do mês de maio de 2020;

# ANÁLISE DE RISCOS DE UM PROJETO

Na análise de risco são listados todos os prováveis riscos que possam acontecer durante o projeto. Em qual processo, qual é a ameaça, o impacto que o mesmo pode causar no desenvolvimento do projeto, sua probabilidade de acontecer e o risco (produto da multiplicação de “Impacto” com “Probabilidade”)

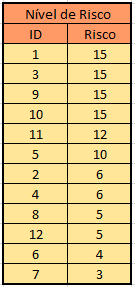


# Nível e Planos de Ação para os Riscos

Definimos uma hierarquia do nível dos riscos, do mais grave para o menos grave. Assim, damos uma maior atenção ás ameaças com maior impacto e probabilidade de acontecer.







# Planos de ação

Planos de ação para os riscos referente à tabela de riscos gerais.

* **RISCO 1 –** Falta de Energia.
* **PLANO 1-** Rever o que foi feito, reorganizar o projetoe fazer em casa se possível.
* **RISCO 2 –** Problemas de Hardware.
* **PLANO 2-** Repor o hardware comprometido
* **RISCO 3 –** Falta de Internet.
* **PLANO 3-** Revisão da documentação refatoração do código.
* **RISCO 4 –** Ausência de algum integrante.
* **PLANO 4-** Seguir o cronograma e se possível adiantar atividades.
* **RISCO 5 –** Perda de dados.
* **PLANO 5-** Realizar backup diariamente em todos os computadores.
* **RISCO 6 –** Conflito entre integrantes do grupo.
* **PLANO 6-** Resolver os conflitos com o grupo e se persistir, com o auxílio do professor.
* **RISCO 7 –** Atraso na elaboração do projeto.
* **PLANO 7-** Reorganizar o cronograma e adiantar a atividade fora do horário de aula.
* **RISCO 8 –** Ausência de integrantes durante a apresentação.
* **PLANO 8-** Dividir novamente o as funções e apresentar sem o integrante ausente.
* **RISCO 9 –** Falha no projetor.
* **PLANO 9-** Buscar uma nova sala para apresentação.
* **RISCO 10 –** Eventos do SENAI.
* **PLANO 10-** Reorganizar o cronograma e adiantar a atividade fora do horário de aula.
* **RISCO 11 –** Problemas familiares.
* **PLANO 11-** Ter comunicação e explicar a situação e não levar para o pessoal e afetar o grupo.
* **RISCO 12 –** Feriados.
* **PLANO 12-** Reorganizar o cronograma e adiantar a atividade fora do horário de aula.

# SPRINTS

Apresenta-se nesta seção a documentação referente ao desenvolvimento do software.

# Primeiro Sprint

Durante o primeiro Sprint, o foco é a conclusão de todo o banco de dados e o *back-end* com o início do *front-end* para teste e o início do *front-end* mobile (*splash screen* e tela de login). Adiantar ao máximo a documentação.

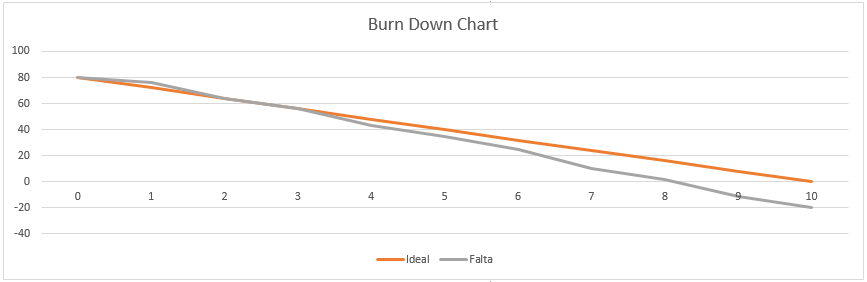
# Product Backlog

Não houve modificações do Product Backlog no primeiro Sprint.

# Sprint Backlog

* Configuração do mongo para o banco de dados.
* Criação de métodos no web service (node.js) para a conexão da web com o back-end.
* Realização do login e logout na web.
* Realização do CRUD dos hospitais e dos pacientes.
* Páginas web com estilização de Home Page, suporte para os usuários, sobre a empresa, cadastro e login.
* Splash Screen com animação.

Burn Down Chart



# Diagramas

[Apresenta](http://www.devmedia.com.br/curso/introducao-a-uml/128) os principais diagramas realizados.

# Plano de testes

O plano de teste é uma maneira de encontrar defeitos e bugs no sistema para serem futuramente corrigidos. Mesmo que durante a programação sempre foram realizados alguns, ainda sim no plano são realizados testes finais passo a passo.

# Resultados

Obtivemos sucesso em todas nossas funcionalidades construídas conforme nosso planejamento sem termos casos extremos de problemas na realização das funções

# Kanban e Retrospectiva

[Apresenta](http://www.devmedia.com.br/curso/introducao-a-uml/128) imagens do quadro de Kanban e as conclusões da retrospectiva.

# Modelo de Dados

Esta parte do planejamento traz informações necessárias para a construção de um banco de dados para o Sistema de Gerenciamento de Acessos.

# Diagrama de Entidade e Relacionamento

Nessa etapa se define: as entidades necessárias para a construção do Banco de Dados; Os relacionamentos e o seu grau, ou seja, a quantidade de entidades que estão ligadas ao relacionamento.

# Modelo lógico do banco de dados

Nessa etapa se define: os atributos pertencentes a cada entidade; as chaves primárias e estrangeiras; o tipo de cada campo e valor de determinados campos.

# Dicionário de dados

Nessa etapa é elaborada uma organização básica dos dados do banco. Aqui são informadas as entidades, com seus respectivos campos, tipos e descrições. O banco foi desenvolvido no servidor de banco de dados SQL Server 2012.

# PRINCIPAIS TELAS DO SISTEMA

Descreve de maneira simples as principais telas do sistema

# CONCLUSÃO

# Escreva os resultados obtidos

Resultados obtidos

# Constatações

Constatações

# Sugestões de possíveis aperfeiçoamentos técnicos

Sugestões

# REFERÊNCIAS

Aqui vai o texto de referências (MORE)

# GLOSSÁRIO

Se houver necessidade

# ANEXOS

Se houver necessidade