

PRONTUÁRIO ÚNICO UCB: UMA PROPOSTA PARA OS CURSOS DE SAÚDE DA UNIVERSIDADE CATÓLICA DE BRASÍLIA

Pró-Reitoria Acadêmica

Escola de Educação, Tecnologia e Comunicação

Curso de Ciência da Computação

Trabalho de Conclusão de Curso

Autores: Eduardo Edson Batista Cordeiro Alves

Hygor Fellipe Moreira

Orientador: Prof. Me. Mário de Oliveira Braga Filho

Co-orientadora: Prof. Dra. Samara Haddad Simões Machado

Brasília - DF

2018

EDUARDO EDSON BATISTA CORDEIRO ALVES

HYGOR FELLIPE MOREIRA

PRONTUÁRIO ÚNICO UCB: UMA PROPOSTA PARA OS CURSOS DE SAÚDE DA UNIVERSIDADE CATÓLICA DE BRASÍLIA

Monografia apresentada ao curso de graduação em Ciência da Computação da Universidade Católica de Brasília, como requisito parcial para obtenção do Título de Bacharel em Ciência da Computação.

Orientador: Prof. Me. Mário de Oliveira Braga Filho

Co-orientadora: Prof. Dra. Samara Haddad Simões Machado

Brasília

2018



Monografia de autoria de Eduardo Edson, Hygor Fellipe, intitulada “PRONTUÁRIO ÚNICO UCB: UMA PROPOSTA PARA OS CURSOS DE SAÚDE DA UNIVERSIDADE CATÓLICA DE BRASÍLIA”, apresentada como requisito parcial para obtenção do grau de Bacharel em Ciência da Computação da Universidade Católica de Brasília, em 05/07/2018, defendida e aprovada pela banca examinadora abaixo assinada:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
Prof. Me. Mário de Oliveira Braga Filho

Orientador

Bacharelado em Ciências da Computação – UCB

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Prof. Dra. Samara Haddad Simões Machado

Curso de Farmácia - UCB

Co-Orientadora

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Prof. Dr. Eduardo Amadeu Dutra Moresi

Avaliador Interno

Doutorado em Educação – UCB

Brasília-DF

2018

AGRADECIMENTOS

Agradecemos primeiramente a Deus, por ter nos proporcionado chegar até essa etapa em nossas vidas, e conseguir realizar esse trabalho.

A todos os nossos familiares ancestrais, através dos quais adquirimos sabedoria e obtivemos uma boa educação, formando o nosso caráter e nos incentivando a desenvolver nossas habilidades.

Ao nosso orientador, Professor Mário de Oliveira Braga Filho, que nos orientou no desenvolvimento do trabalho e nos recebeu com cordialidade e disponibilidade.

A todos os amigos e colegas que acompanharam, incentivaram e deram opiniões no decorrer do trabalho sempre torcendo e apoiando.

Agradecemos à Profa. Dra. Samara, Coordenadora do Curso de Fármacia e os Fisioterapeutas Raynan e Viviane, pela atenção, gentileza e respeito com que nos atenderam na fase de levantamento de requisitos, contribuindo bastante para nosso projeto

Ao Curso de Ciência da Computação da Universidade Católica de Brasília, е às pessoas com quem convivemos nesse espaço ао longo desses anos. Aos professores e profissionais que possibilitaram o aprendizado e incentivaram a busca pelo conhecimento.

Por fim, agradecemos as dificuldades que passamos para poder chegar até aqui. Sem elas não teríamos aprendido o suficiente para sermos seres humanos capazes de utilizar o conhecimento para criar tecnologias que possam ajudar a todos.

*“Aquele que não é um bom aprendiz não será um bom mestre.”*

*Platão*

RESUMO

CORDEIRO, Eduardo Alves, MOREIRA, Hygor Fellipe. “PRONTUÁRIO ÚNICO UCB: UMA PROPOSTA PARA OS CURSOS DE SAÚDE DA UNIVERSIDADE CATÓLICA DE BRASÍLIA”. 2018. Páginas: 68. Monografia (Bacharelado em Ciência da Computação) – Universidade Católica de Brasília, Brasília, 2018.

O avanço tecnológico registrado nos últimos anos vem trazendo enormes benefícios na melhoria da qualidade da coleta, transformação e distribuição da informação. Na área da saúde não é diferente. Os órgãos responsáveis pela melhoria e transformação de sistemas, serviços e processos de Saúde, destacando-se entre eles a Sociedade Brasileira de Informática na Saúde – SBIS, vêm aumentando sua atuação na promoção e melhoria da qualidade dos sistemas informatizados da saúde. A Universidade Católica de Brasília oferece à comunidade uma grade de cursos da área de Saúde que, além de formar profissionais das mais diversas linhas de atuação no benefício da saúde humana possibilita o atendimento à sociedade em geral, através de seus programas sociais. Os Cursos da área da saúde estão sempre buscando fomentar as ações referentes à valorização do acesso continuado à educação, entendendo que é por meio do atendimento à população que seus alunos, futuros profissionais da saúde, começam desde os primeiros semestres a entender o valor de suas carreiras na melhoria da condição de vida do ser humano. Este trabalho tem por finalidade apresentar uma prova de conceito denominada de Sistema Único de Prontuário da UCB, com a finalidade de permitir que todos os cursos que fazem atendimentos externos, possam reunir em um só sistema as informações do paciente que adentra os portões da Universidade atrás de atendimento médico, fisioterápico, odontológico e de outras áreas afins. Como prova de conceito foram feitos levantamentos de requisitos no curso de fisioterapia, mapeando todos os processos internos de atendimento e avaliação das condições físicas e motoras dos pacientes, informatizando todas as fichas utilizadas nos diversos procedimentos e áreas da fisioterapia. A execução contou com a aplicação da metodologia C*BL (Challenge Based Learning)* aliada ao framework de processos agéis Scrum. O desenvolvimento mostrou que é possível reunir a informação dos pacientes de todos os cursos em uma só plataforma de informação, permitindo assim a utilização de um prontuário único que possa receber as informações advindas de todos os atendimentos recebidos pelo paciente na Universidade.

Palavras-Chave:Prontuário Eletrônico. Sistema de Saúde. Programas sociais.

ABSTRACT

CORDEIRO, Eduardo Alves, MOREIRA, Hygor Fellipe. “UCB SINGLE MEDICAL RECORD: A PROPOSAL FOR THE HEALTH COURSES OF THE CATHOLIC UNIVERSITY OF BRASILIA”. 2018. Pages: 68. Monography (Bachelor in Computer Science) – Catholic University of Brasilia, Brasilia, 2018.

The Technological advances recorded in recent years have brought enormous benefits in improving the quality of information collection, transformation and distribution. In the area of health is no different. The bodies responsible for improving and transforming health systems, services and processes, including the Brazilian Society of Health Informatics (SBIS), have been increasing their activities in promoting and improving the quality of computerized health systems. The Catholic University of Brasília offers the community a range of courses in the area of health that, in addition to training professionals of the most diverse lines of action in the benefit of human health, makes it possible to attend to society in general, through its social programs. The courses in the health area are always seeking to promote actions related to the valorization of continued access to education, understanding that it is through the service to the population that their students, future health professionals, start from the first semesters to understand the value of their careers in improving the human condition of life. The purpose of this paper is to present a proof of concept called the UCB Single Medical Record, with the purpose of allowing all courses that provide external assistance, can gather in a single system the information of the patient that enters the university gates behind of medical, physiotherapeutic, dental and other related areas. As proof of concept, requirements were surveyed in the physical therapy course, mapping all the internal processes of care and evaluation of the physical and motor conditions of the patients, computerizing all the files used in the various procedures and areas of physiotherapy. The implementation was based on the CBL (Challenge Based Learning) methodology coupled with the Scrum agile process framework. The development showed that it is possible to gather patients' information from all courses in a single information platform, thus allowing the use of a single medical record that can receive the information coming from all patient care received at the University.

Key words: Electronic Health Record. Health System. Social Programs.

SUMÁRIO

[1 INTRODUÇÃO 9](#_Toc9741)

[2 MOTIVAÇÃO 10](#_Toc1238)

[3 OBJETIVOS 11](#_Toc30368)

[3.1 GERAL 11](#_Toc29886)

[3.2 ESPECÍFICOS 11](#_Toc17955)

[4 JUSTIFICATIVA 12](#_Toc23598)

[5 DESCRIÇÃO DO SISTEMA PROPOSTO 13](#_Toc29454)

[5.1 RESULTADOS ESPERADOS 13](#_Toc31832)

[5.2 RESTRIÇÕES DO SISTEMA PROPOSTO 13](#_Toc11240)

[5.3 ÁREAS AFETADAS PELO SISTEMA 14](#_Toc23393)

[5.4 FLUXO DE ATENDIMENTO DA CLÍNICA DE FISIOTERAPIA 15](#_Toc28974)

[5.5 MAPA MENTAL DO SISTEMA 15](#_Toc26380)

[6 SISTEMAS UTILIZADOS PELAS CLÍNICAS DA UCB 17](#_Toc13260)

[6.1 CLÍNICA DE BIOMEDICINA 17](#_Toc1519)

[6.2 CLÍNICA DE ENFERMAGEM 17](#_Toc23568)

[6.3 CLÍNICA DE FARMÁCIA 17](#_Toc21244)

[6.4 CLÍNICA DE FISIOTERAPIA 17](#_Toc9595)

[6.5 CLÍNICA DE MEDICINA 17](#_Toc24229)

[6.6 CLÍNICA DE NUTRIÇÃO 17](#_Toc24034)

[6.7 CLÍNICA DE ODONTOLOGIA 17](#_Toc27118)

[6.8 CLÍNICA DE PSICOLOGIA 18](#_Toc19123)

[7 PLANEJAMENTO DO PROJETO 19](#_Toc29468)

[7.1 PLANO DE DESENVOLVIMENTO DO PROJETO 19](#_Toc19812)

[7.1.1 Ciclo de vida do projeto 19](#_Toc26865)

[7.1.2 Método de desenvolvimento e ferramentas case 21](#_Toc26706)

[7.1.3 Linguagens de programação 24](#_Toc16516)

[7.1.4 Ambiente de hardware para desenvolvimento 25](#_Toc3996)

[7.2 PLANO DE ORGANIZAÇÃO 25](#_Toc19266)

[7.2.1 Equipe de gerência 25](#_Toc19892)

[7.2.2 Equipe de desenvolvimento 26](#_Toc21919)

[7.3 PLANO DE ACOMPANHAMENTO 27](#_Toc16842)

[7.3.1 Marcos de ponto e controle 27](#_Toc7280)

[7.3.2 Métodos de acompanhamento e controle 27](#_Toc28219)

[7.3.3 Análise e gerência de riscos 27](#_Toc22032)

[7.4 PLANO DE DOCUMENTAÇÃO 28](#_Toc16286)

[7.5 CRONOGRAMA 29](#_Toc13759)

[8 REFERENCIAL TEÓRICO 30](#_Toc14639)

[8.1 SOBRE O DESENVOLVIMENTO ÁGIL 30](#_Toc17110)

[8.2 SOBRE O MÉTODO SCRUM COM CBL 31](#_Toc22953)

[9 ESPECIFICAÇÃO DE REQUISITOS DO SISTEMA 34](#_Toc27338)

[9.1 DIAGRAMA DE CLASSE DE DOMINIO 34](#_Toc9161)

[9.2 TELAS DO SISTEMA 35](#_Toc11284)

[10 DOCUMENTAÇÃO DO PROJETO 49](#_Toc20498)

[10.1 DICIONÁRIO DE DADOS 49](#_Toc26989)

[11 CONCLUSÃO 55](#_Toc6538)

[REFERÊNCIAS BIBLIOGRÁFICAS 56](#_Toc12307)

[APÊNDICES 57](#_Toc25994)

[APÊNDICE A – DOCUMENTO GERADO PELO CBL 57](#_Toc20706)

[APÊNDICE B – MANUAL DE INSTALAÇÃO DO PRONTUÁRIO ÚNICO UCB 61](#_Toc18129)

# 

# INTRODUÇÃO

A saúde é uma questão de extrema importância para toda a população inserida em qualquer sociedade, pois, a partir de uma gestão bem estruturada nessa área é possível garantir integridade física e/ou mental às pessoas que necessitam de atendimento médico para tratamento preventivo ou corretivo. Tendo em vista tamanha responsabilidade de gerenciar, controlar e organizar uma grande demanda de dados e informações sobre pacientes, a praticidade e robustez proporcionados por sistemas de informação em saúde (SIS) são capazes de auxiliar, de forma eficiente, o controle de prontuários e atendimentos realizados por clínicas, hospitais, entre outras instituições ligadas à saúde, pois, através do processamento de dados, esses sistemas são capazes de produzir informações relevantes.

Este projeto tem por objetivo o desenvolvimento de uma solução para categorização e organização de prontuários de pacientes das diversas clínicas de atendimento da Universidade Católica de Brasília, através de um sistema web, com o intuito de apoiar a tomada de decisão dos supervisores das diversas clínicas presentes no campus, buscando a melhoria da qualidade de vida dos pacientes que recebem atendimento.

Este documento visa fornecer um detalhamento acerca do projeto e está organizado da seguinte maneira: No capítulo 3 serão descritos os objetivos geral e específicos. No capítulo 4 será apresentada a justificativa da proposta. No capítulo 5 serão esclarecidos os resultados esperados, as restrições do sistema proposto e as áreas afetadas pelo sistema. O capítulo 7 é dedicado ao plano de desenvolvimento do projeto, onde são evidenciados os métodos adotados, a análise e gerência de riscos, equipamentos e softwares utilizados e o cronograma. No capítulo 8 encontra-se o referencial teórico sobre o modelo de desenvolvimento utilizado e como o modelo ágil funciona. O capítulo 9 é aderente à modelagem do sistema e diz respeito aos diagramas da Linguagem de Modelagem Unificada (UML), o Diagrama Domínio e a prototipação de telas. O capítulo 10 traz a documentação do projeto como o dicionário de dados. O capítulo 11 é dedicado à conclusão.

# MOTIVAÇÃO

O estudo da tecnologia da informação (TI) e suas diversas áreas de atuação impõe uma série de desafios, dentre eles a maneira de utilizar a tecnologia para melhoria da condição de vida das pessoas.

O grande volume de dados produzidos pelo uso da TI está impactando a maneira das empresas fazerem negócio, e o desafio central, neste caso, é saber exatamente quais dados são importantes e como utilizá-los na tomada de decisão, a fim de criar um diferencial competitivo. A cada dia surgem novos sistemas e aplicativos móveis que, conectados ou não a outros dispositivos eletrônicos, buscam coletar, armazenar e difundir informações com propósitos diversos.

Durante a disciplina de APS (Análise e Projeto de Software), foi sugerida a ideia deste projeto a fim de se verificar a sua aplicação prática na própria Universidade. A elaboração do CBL (Challenge Based Learning) mostrou que a ideia poderia agregar valor aos projetos sociais da Universidade Católica de Brasília. Seguindo essa metodologia, foi identificado a ausência de um sistema único e centralizado, de uso comum a todas as clínicas, que fosse capaz de manter prontuários dos pacientes que recebem atendimento médico dentro da universidade de forma centralizada.

O emprego das melhores práticas da engenharia de software, buscando validar o modelo proposto por Santos (2016) unindo a metodologia do CBL com o framework Scrum de processos agéis, tornou possível o planejamento, a elaboração da documentação, bem como a modelagem do sistema proposto. Poder transformar uma idéia em um produto de software e ter a possibilidade de contribuir de alguma forma com o ambiente acadêmico é um dos fatores mais motivadores deste trabalho.

# OBJETIVOS

## GERAL

Desenvolver o sistema Prontuário Único UCB para possibilitar aos supervisores das clínicas de saúde da Universidade Católica de Brasília a criação, organização e categorização de prontuários dos pacientes que buscam tratamento no campus, registrando informações do paciente, atendimentos realizados, dentre outras informações necessárias a fim de compor os devidos prontuários que estarão disponíveis para visualização por parte das outras clínicas, dentro de um contexto centralizado, envolvendo toda a área de saúde da universidade que realize atendimento ao público. Além disso, o sistema tem como objetivo dar a possibilidade aos supervisores das disciplinas de estágio obrigatório de gerenciar melhor as atividades dos alunos estagiários dentro das clínicas de suas respectivas áreas de atuação.

## ESPECÍFICOS

Os objetivos específicos buscaram principalmente entender o atendimento de pacientes na UCB, compreender a dinâmica de permanência desses pacientes nos diversos programas, conhecer quais os recursos tecnológicos disponíveis para triagem e atendimento em cada curso da área de saúde além de verificar como são salvaguardadas as informações referentes aos prontuários desses pacientes. Para tanto foram desenvolvidos os seguintes requisitos funcionais:

1. Cadastrar: pacientes, alunos, áreas de atendimento, disciplinas, recepcionistas, supervisores, setores, fichas de triagem, fichas de evolução, fichas de inscrição específicas de cada área de atendimento da clínica.
2. Pesquisar: pacientes, alunos, áreas de atendimento, disciplinas, recepcionistas, supervisores, setores, fichas de triagem, fichas de evolução, fichas de inscrição específicas de cada área de atendimento da clínica.
3. Alterar: pacientes, alunos, áreas de atendimento, disciplinas, recepcionistas, supervisores, setores, senha do perfil de usuário autenticado no sistema.

# JUSTIFICATIVA

Dentro do contexto da universidade, buscou-se avaliar problemas que pudessem ser resolvidos ou amenizados em grande parte, com o apoio de um sistema informatizado. Nesse sentido, logo foi percebida uma grande oportunidade de auxiliar os processos de tomada de decisão e de gestão da informação tão necessários no dia a dia das clínicas de saúde da Universidade Católica de Brasília. Com o decorrer do tempo, as diferentes áreas da saúde que prestam atendimento à comunidade acabaram tomando métodos diferentes sobre como realizar o armazenamento de informações referentes à base de pacientes e seus respectivos prontuários. Sendo assim, não existe uma padronização para a coleta, armazenamento e disponibilização de informações, e o paciente passa por procedimentos diferenciados ao ser atendido em áreas distintas. Compreende-se que o fluxo existente no atendimento prejudica o compartilhamento de informações entre diferentes clínicas-escola, quando consideramos a situação em que o paciente pode receber atendimento em diversas áreas e precisa ter uma trajetória mapeada que relate os documentos gerados por todos os procedimentos aos quais ele foi submetido. Diante desse desafio, a solução proposta por este projeto busca dissolver as dificuldades de armazenamento e uso de informações, além de integrar as clínicas-escola em um único ambiente centralizado, proporcionando maior interação entre elas e agilizando o acesso aos dados.

# DESCRIÇÃO DO SISTEMA PROPOSTO

O Prontuário Único UCB é um sistema web, podendo ser acessado a qualquer momento de qualquer computador conectado à internet. O cadastramento dos usuários pode atender três perfis: supervisor, aluno ou recepcionista, e são feitos através do sistema. A Recepcionista está apta a criar o cadastro do paciente e preencher uma ficha de triagem, a fim de iniciar o processo de atendimento. O Aluno fica responsável por cadastrar fichas de evolução e fichas de atendimento nas áreas específicas da clínica e conduzir o paciente a uma conduta de tratamento. O Supervisor é o usuário que mais possui privilégios no sistema, podendo cadastrar novos usuários, pacientes, áreas de atendimento, etc. O foco do sistema é criar, para cada paciente cadastrado, uma interface centralizada onde estarão disponíveis todas as fichas de inscrição preenchidas através do sistema no decorrer dos atendimentos realizados nas diversas clínicas-escola. Dessa forma será possível manter um histórico do paciente que ficará disponível para todas as clínicas que utilizam o sistema.

## RESULTADOS ESPERADOS

Os resultados esperados com o desenvolvimento do sistema são:

1. Otimização do atendimento de pacientes dos programas sociais da área de saúde da Universidade Católica de Brasília.
2. Melhor controle do prontuário do paciente.
3. Centralização das informações do paciente facilitando a tomada de decisão dos profissionais da área de saúde da Universidade Católica de Brasília.
4. Alívio da carga de trabalho gerada pela atualização e armazenamento de dados de forma manual, em meios diversos e descentralizados (papel, documentos avulsos em formato doc, etc).

## RESTRIÇÕES DO SISTEMA PROPOSTO

Algumas ações não poderão ser realizadas no desenvolvimento do protótipo inicial do Prontuário Único UCB. São elas:

1. Laudo Fisioterapêutico de Alta.
2. Tela de informações a respeito dos atendimentos marcados.
3. Esquema de grupos de alunos no sistema, pois na clínica de fisioterapia os estagiários são designados em grupos que realizam rodízio nas áreas de atendimento durante o semestre.
4. Funcionalidade de anexar arquivos externos (doc, pdf) às fichas de inscrição.
5. Funcionalidade de gerenciamento de condutas de tratamento elaboradas pelos alunos ao final de um atendimento.
6. Obtenção de certificação junto aos órgãos competentes para possibilitar o desenvolvimento do sistema no formato de um Prontuário Eletrônico.

## ÁREAS AFETADAS PELO SISTEMA

O Prontuário Único UCB pode afetar de maneira positiva três perfis de usuário:

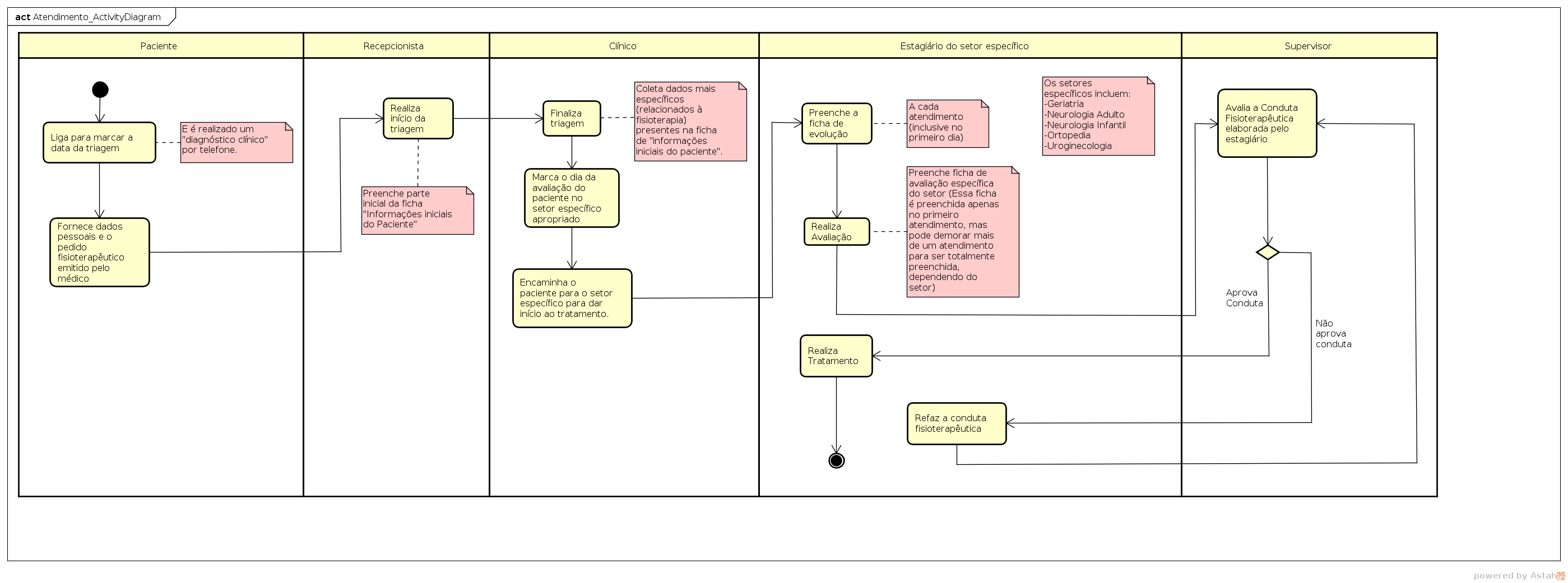
* Usuário Recepcionista: poderá realizar cadastro de novos pacientes e conduzir o processo de triagem do paciente, preenchendo as informações iniciais da ficha de triagem.
* Usuário Aluno: Estagiário da clínica que é responsável por cadastrar fichas de evolução e de atendimento específico do paciente no sistema, e utiliza essas informações como apoio para elaboração da conduta de tratamento fora do escopo do sistema, a ser avaliada pelo supervisor no mundo real.
* Usuário Supervisor: Tem privilégios absolutos para cadastrar usuários, áreas de atendimento, setores, etc. Além disso, pode fazer as mesmas ações que o aluno, no escopo do sistema.

Sendo assim, em qualquer clínica de saúde da universidade que precise de uma solução para organizar os prontuários e informações dos pacientes, o Sistema Prontuário Único UCB poderá ser aplicado.

## FLUXO DE ATENDIMENTO DA CLÍNICA DE FISIOTERAPIA

O paciente, ao procurar atendimento na clínica de fisioterapia, deve primeiro fazer contato por telefone para marcar uma data para a triagem, além de fornecer algumas informações para diagnóstico inicial. Ao comparecer à clínica, o paciente deve fornecer dados pessoais e o pedido fisioterapêutico emitido por um médico externo. A recepcionista é responsável por realizar o início da triagem, preenchendo a ficha de informações iniciais do paciente. O clínico finaliza a triagem, completando o diagnóstico do paciente, e marca a data da primeira avaliação, encaminhando o indivíduo à área específica de tratamento correspondente à patologia identificada. O estagiário recebe o paciente na data marcada, preenche a ficha de evolução ao inicio de cada atendimento realizado, realiza a avaliação do paciente, elabora uma conduta fisioterapêutica (tratamento), que será avaliada pelo supervisor presente. Caso o supervisor reprove a conduta, o estagiário deve refazê-la e apresenta-la novamente. Caso contrário, o estagiário poderá aplicar a conduta e finalizar o atendimento da respectiva data, remarcando outro atendimento, caso haja necessidade.

Figura 1 - Fluxo de atendimento de pacientes na clínica de fisioterapia.

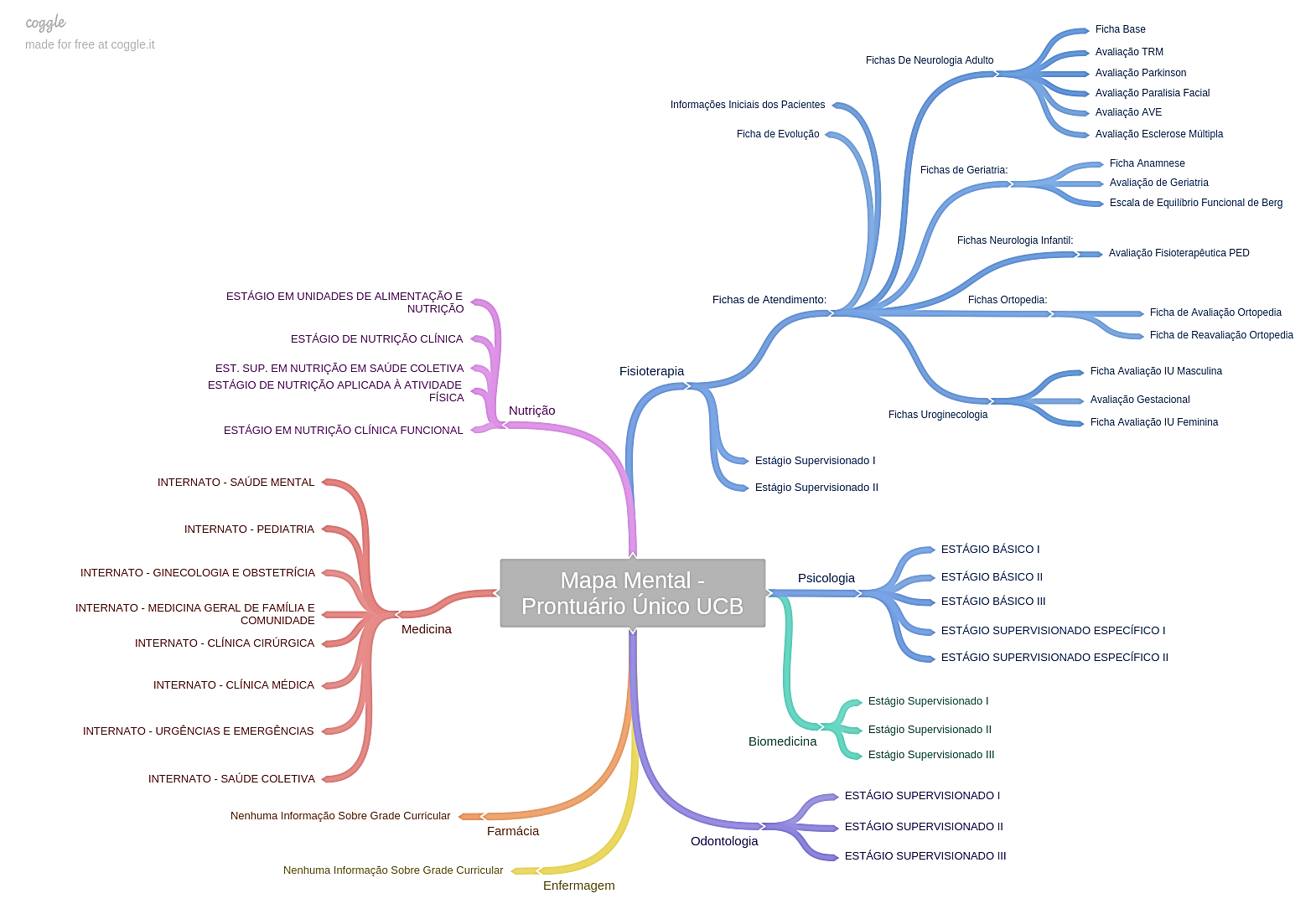


Fonte: Autoria própria

## MAPA MENTAL DO SISTEMA

O mapa mental a seguir mostra todo o escopo de cursos que se desejou atingir. Porém, a prova de conceito foi baseada e aprofundada apenas no curso de fisioterapia, onde realizamos todo o levantamento de requisitos e processos existentes. Para cada curso descrito no mapa estão relacionadas as respectivas disciplinas de estágio obrigatório que o compõe. Para o curso de fisioterapia estão descritas informações adicionais a respeito das áreas de atendimento da clínica, juntamente com as fichas de avaliação utilizadas para coleta de dados em cada uma dessas áreas.

Figura 2 - Mapa mental do sistema Prontuário Único UCB, com esquema detalhado na ramificação do curso de fisioterapia.



Fonte: Autoria Própria

# SISTEMAS UTILIZADOS PELAS CLÍNICAS DA UCB

## ClÍNICA DE BIOMEDICINA

Não possui sistema de controle de prontuários, porém, está em andamento um processo de negociação para contratação do “Sistema de Gestão Laboratorial”, distribuído pela empresa Concent.

## cLÍNICA DE ENFERMAGEM

Sistema: TrakCare – Sistema Unificado de Informações de Saúde.

Empresa: InterSystems.

## CLÍNICA DE FARMÁCIA

Não possui sistema de controle de prontuários.

## CLÍNICA DE FISIOTERAPIA

Sistema: SIS+ Web – Sistema Integrado em Saúde - Fisioterapia.

Empresa: Automatizus.

## ClÍNICA DE MEDICINA

Não foi informada a existência de um sistema para controle de prontuários.

## cLÍNICA DE NUTRIÇÃO

Não possui sistema de controle de prontuários.

## CLÍNICA DE ODONTOLOGIA

Sistema: SISB – Sistema Integrado de Saúde Bucal.

Empresa: Automatizus.

## CLÍNICA DE PSICOLOGIA

Não possui sistema de controle de prontuários.

# PLANEJAMENTO DO PROJETO

## PLANO DE DESENVOLVIMENTO DO PROJETO

O modelo de ciclo de vida a ser adotado pelo projeto é o iterativo e incremental, pois há entrega em prazo específico do produto e além desse fato os requisitos do projeto podem mudar, necessitando validações de como o produto irá se comportar.

Para apoiar o desenvolvimento do produto será utilizado o método *Scrum* em conjunto com o *CBL* proposto pela tese de doutorado de Alan Ricardo dos Santos. De acordo com (SOMMERVILLE, 2011), o *Scrum* é uma metodologia ágil focada em desenvolvimento iterativo, que pode utilizar técnicas ágeis como o XP, estabelecendo - se como um *framework* para gerenciar um projeto. Enquanto o *CBL*, trata-se de uma abordagem multidisciplinar baseada em desafios, esta que proporciona espaço e liberdade para a criatividade de forma a nos manter motivados.

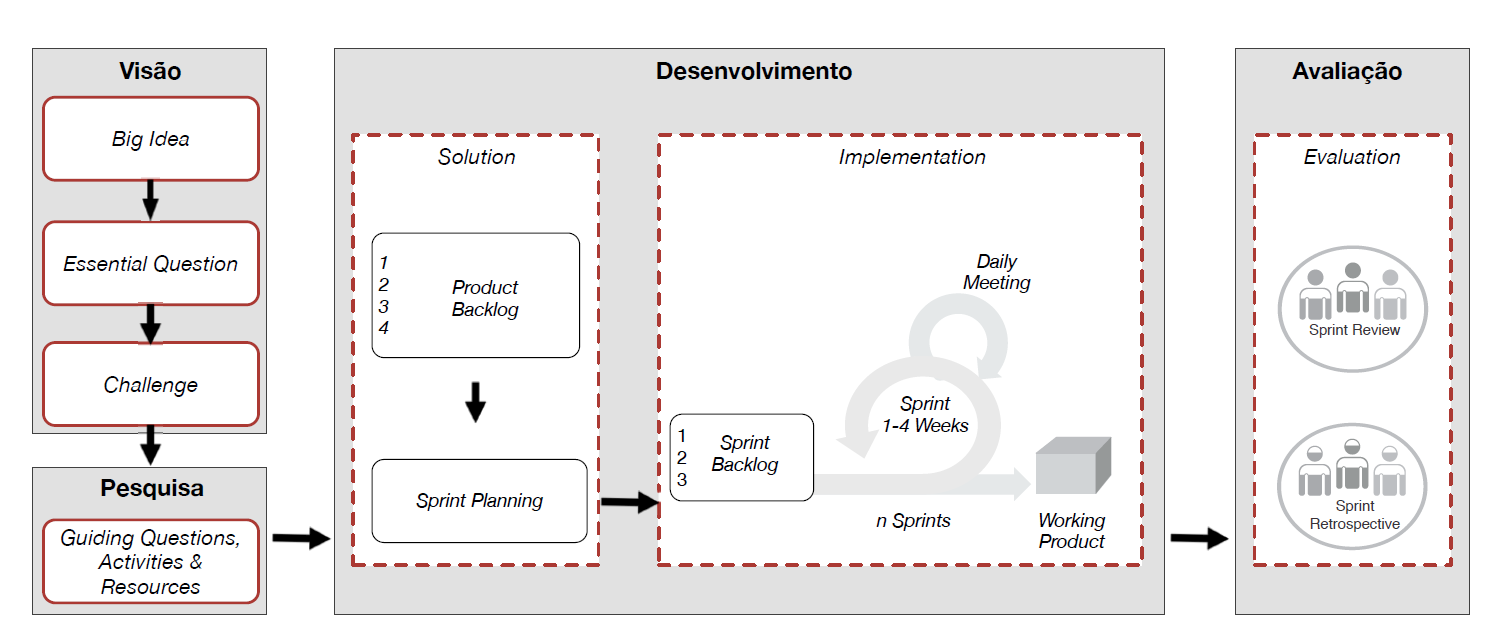
### Ciclo de vida do projeto

O ciclo de vida do SCRUM com CBL é baseado em quatro fases:

Quadro 1 - Ciclo de Vida SCRUM-CBL

|  |  |  |
| --- | --- | --- |
| Fase | Estágios envolvidos | Objetivo |
| Visão | Big Idea, Essential Question,  Challenge | O objetivo desta fase é montar os grupos de estudantes de acordo com afinidades e objetivos em comum, de forma a permitir um processo de aprendizado através da identificação de uma área de interesse e de um problema a ser resolvido. |
| Pesquisa | Guiding Questions, Resources  and Activities, Solution Proposal | Tem como objetivo proporcionar uma oportunidade para que os membros das equipes descubram e aprendam quais os conhecimentos, recursos e atividades são necessários para propor uma solução que seja viável, permitindo também um processo de aprendizado baseado em problemas reais. |
| Desenvolvimento | Product Backlog, Sprint Planning, Sprint development, Testing | Faz com que a proposta de solução encontrada através do processo de pesquisa seja transformada em requisitos transpostos em um product backlog, os quais são desenvolvidos através de curtos ciclos de desenvolvimento proporcionando aprendizado e melhoria contínua. |
| Avaliação | Sprint Review, Sprint Retrospective | Permite que a cada iteração de desenvolvimento o produto desenvolvido seja avaliado em conjunto de instrutores e membros da equipe e entre membros de diferentes equipes, além de permitir também uma retrospectiva sobre o processo de aprendizado e desenvolvimento durante a iteração de desenvolvimento. |

Figura 3 - Ciclo de Vida Scrum com CBL



Fonte: Santos - 2016

### Método de desenvolvimento e ferramentas case

Ferramentas CASE têm como objetivo auxiliar na concepção do sistema e visualizar aspectos da aplicação. Em um sistema desenvolvido seguindo o paradigma OO que provê benefícios em relação à qualidade e reaproveitamento de software (BASILI; BRIAND; MELO, 1996), a modelagem seguindo os princípios da UML é de suma importância.

A escolha das ferramentas CASE leva em consideração que todas as tecnologias são gratuitas, ou fornecem planos gratuitos. Possibilitando assim a redução de custos para do desenvolvimento de projeto.

1. Astah Community

É uma ferramenta de modelagem UML (Unified Modeling Language – Linguagem de Modelagem Unificada) cujo objetivo é auxiliar a modelagem de diagramas importantes para a fase de planejamento do sistema, e possui uma versão gratuita.

1. Atom

É um editor de texto de código aberto disponível para as plataformas Linux, macOS e Microsoft Windows, desenvolvido pelo GitHub sob a licença MIT.

1. PostgreSQL

É um Sistema Gerenciador de Banco de Dados (SGBD), que utiliza a linguagem SQL (Linguagem de Consulta Estruturada, do inglês Structured Query Language) como interface. É atualmente um dos SGBD’s de código aberto mais avançados.

O manual de instalação do sistema Prontuário Único UCB está localizado no Apêndice B, e nele está incluído o comando necessário para instalação do SGBD PostgreSQL.

O Banco de dados neste projeto contém as seguintes tabelas:

* servicos\_areaatendimento
* servicos\_disciplina
* servicos\_setor
* usuarios\_aluno
* usuarios\_paciente
* usuarios\_recepcionista
* usuarios\_supervisor
* usuarios\_fisioterapiaacidentevascularencefalico
* usuarios\_fisioterapiaavaliacaofeminina
* usuarios\_fisioterapiaavaliacaogestacional
* usuarios\_fisioterapiaavaliacaomasculina
* usuarios\_fisioterapiaberg
* usuarios\_fisioterapiaesclerosemultipla
* usuarios\_fisioterapiaevolucao
* usuarios\_fisioterapiageriatriaanamnese
* usuarios\_fisioterapiageriatriaavaliacao
* usuarios\_fisioterapianeurologiainfantilavaliacao
* usuarios\_fisioterapianeurologica
* usuarios\_fisioterapiaortopediaavaliacao
* usuarios\_fisioterapiaortopediareavaliacao
* usuarios\_fisioterapiaparalisiafacial
* usuarios\_fisioterapiaparkinson
* usuarios\_fisioterapiatriagem
* usuarios\_fisioterapiatrm

1. Trello

É um sistema web de quadro virtual para gerenciamento de tarefas que segue o método "kanban", muito usado no desenvolvimento de projetos de software em conjunto com a metodologia Scrum.

Figura 4 - Quadro de documentos construído durante o planejamento do projeto.

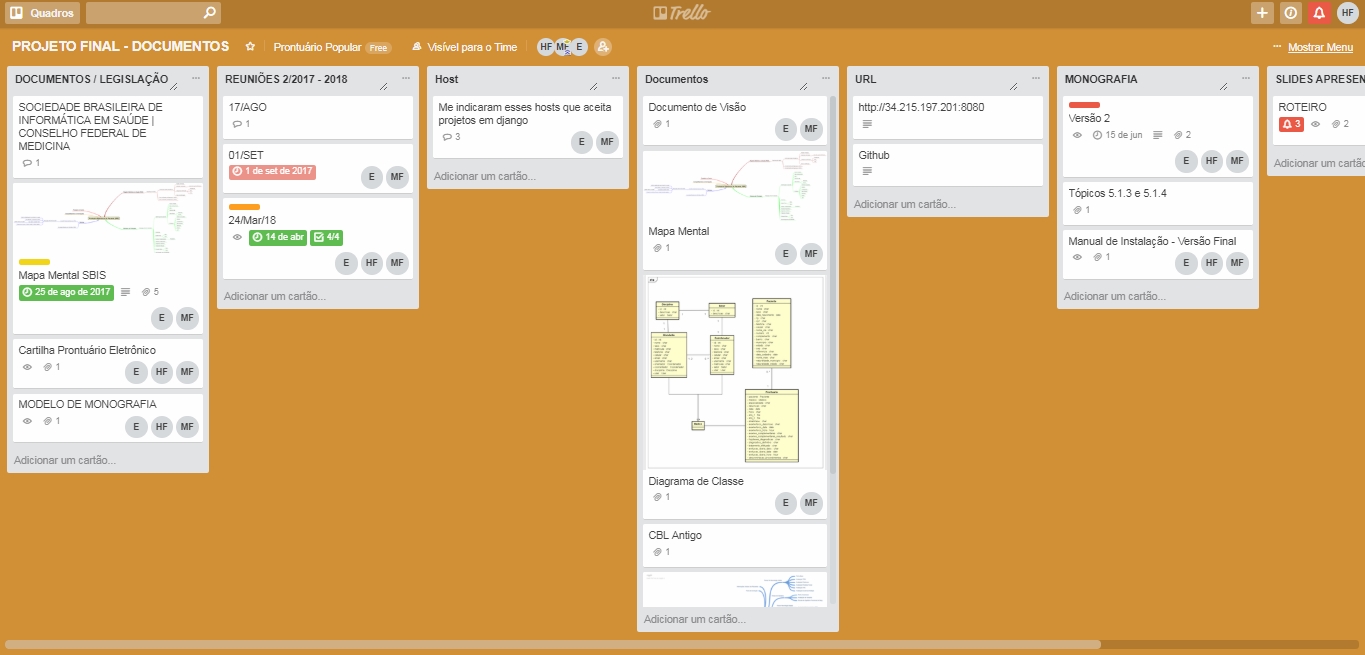
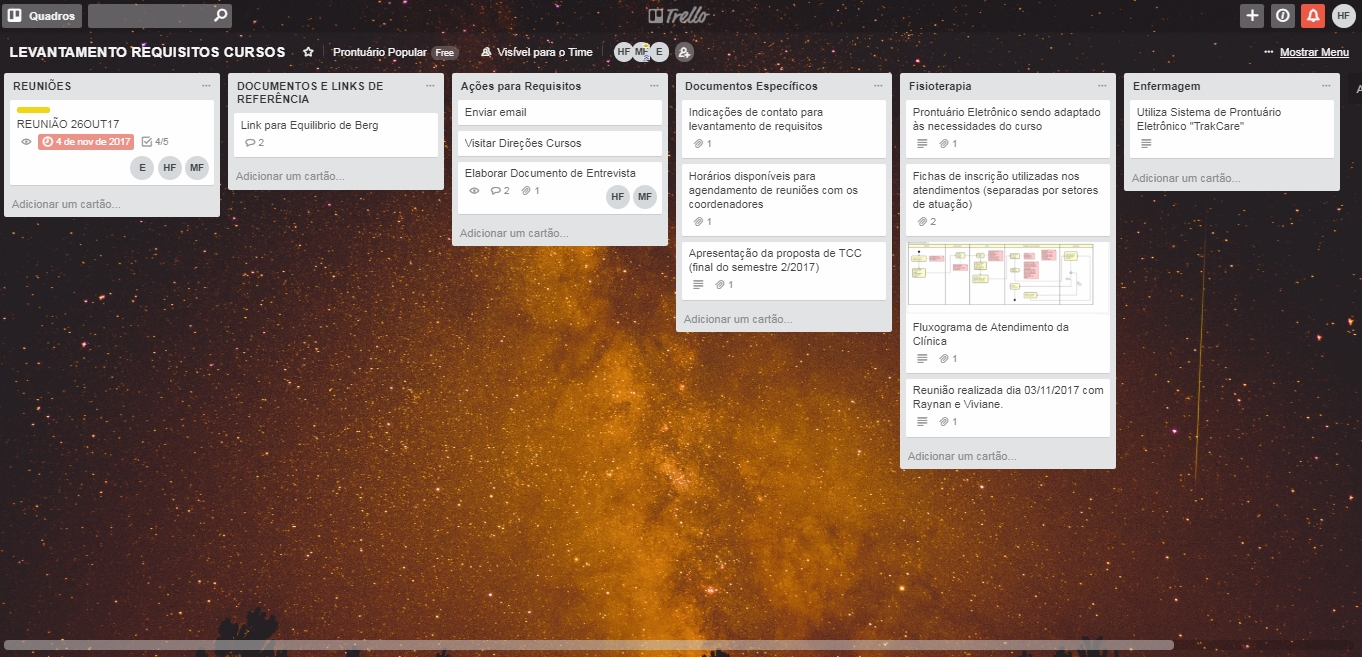


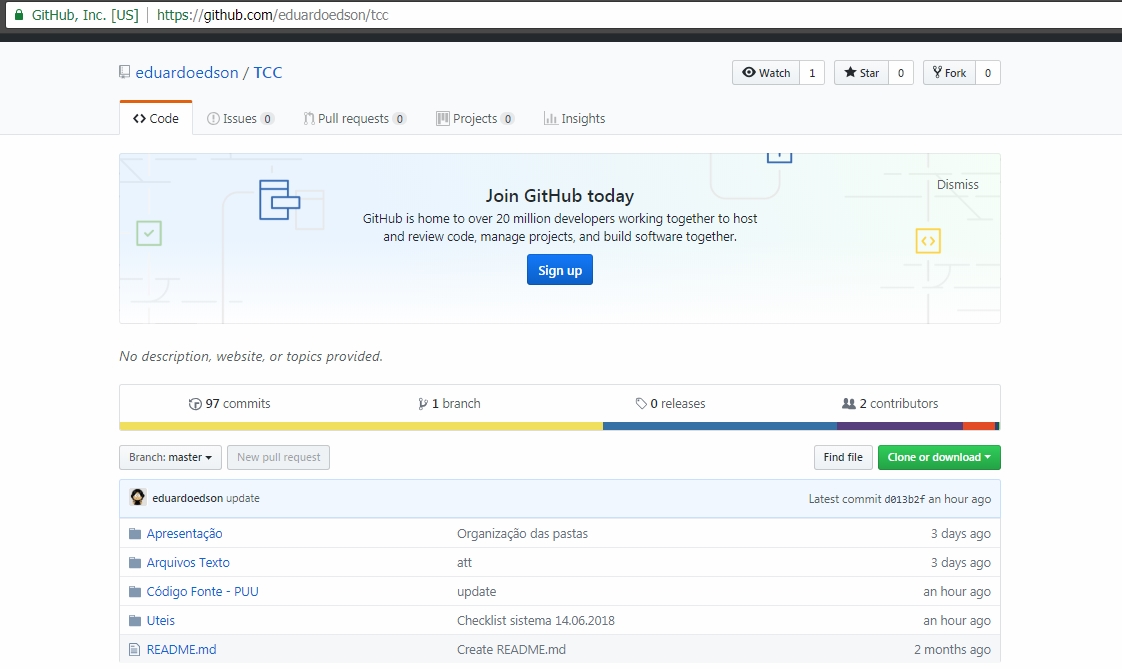
Figura 5 - Quadro de levantamento de requisitos construído durante o planejamento do projeto.



1. Git

É um sistema de controle de versão distribuído e de gerenciamento de código fonte, com ênfase em velocidade.

Figura 6 - Página inicial do projeto no github.



Fonte: <https://github.com/eduardoedson/tcc>

1. Django 1.11.4

É um framework direcionado para desenvolvimento rápido para web. Ele utiliza o padrão model-template-view (MTV) e é escrito em Python. É um projeto de código aberto, publicado sob a licença BSD em 2005.

1. Bower 5.2.0

É um gerenciador de pacotes bastante popular, construído para gerenciar especialmente as dependências do front-end de uma aplicação, como projetos CSS, Javascript, etc.

1. Bootstrap 3 – versão 9.0

É um framework web de código aberto, voltado para o desenvolvimento de componentes de front-end para sites e aplicações web que utilizam HTML, CSS e JavaScript, resultando na construção de sites amigáveis, elegantes e responsivos, a fim de melhorar a experiência do usuário.

### Linguagens de programação

1. Python 3.6.5

É uma linguagem de programação de alto nível, interpretada, de script, imperativa, orientada a objetos, funcional, de tipagem dinâmica e forte. Atualmente possui um modelo de desenvolvimento comunitário, aberto e gerenciado por uma organização sem fins lucrativos. Essa tecnologia compõe o back-end do sistema, que está hospedado no AWS (Amazon Web Services), uma plataforma de serviços de computação em nuvem que possui um plano gratuito com duração de 12 meses.

1. Javascript 1.7

É uma linguagem de programação baseada em scripts e padronizada pela ECMA International (associação especializada na padronização de sistemas de informação).

### Ambiente de hardware para desenvolvimento

#### Computador:

* Processador: i3-6006U @ 2.0 GHz x 4
* Memória RAM: 4GB
* Capacidade: 150 GB
* Sistema Operacional: Ubuntu 17.10

## PLANO DE ORGANIZAÇÃO

### Equipe de gerência

Quadro 2 - Scrum Master – Atividades

|  |  |
| --- | --- |
| Nome | Mário de Oliveira Braga Filho |
| Papel | Scrum Master |
| Atividades | •. Assegurar que a equipe de desenvolvimento funcione plenamente e seja produtiva;  •. Ajudar na cooperação entre todas as funções e papéis do time;  •. Proteger a equipe de interferências externas;  •. Assegurar-se de que a metodologia está sendo seguida, incluindo chamadas para reuniões diárias (Daily Scrum Meetings), revisões de atividade (Sprint Reviews) e reuniões de planejamento das atividades (Sprint Planning). |

Quadro 3 - Product Owner – Atividades

|  |  |
| --- | --- |
| Nome | Eduardo Edson, Mario de Oliveira, Hygor Fellipe |
| Papel | Product Owner |
| Atividades | •. Ele tem as seguintes responsabilidades:  •. Definir as características e conteúdo do produto;  •. Decidir sobre a data de término;  •. Ser responsável pela rentabilidade do produto;  •. Priorizar as funções;  •. Ajustas recursos e priorizar tarefas, como necessário;  •. Aceitar ou rejeitar o resultado do trabalho. |

### Equipe de desenvolvimento

|  |  |
| --- | --- |
| Nome | Eduardo Edson, Hygor Fellipe |
| Papel | Desenvolvimento |
| Atividades | •. Definem metas de cada Sprint, junto ao Scrum Master, e especificam seus resultados de trabalho;  •. Têm o direito de fazer tudo dentro dos limites das diretrizes do projeto para atingir a meta de cada Sprint;  •. Organizam os trabalhos para atingir os objetivos dos Sprints;  •. Trabalham para atingir todos os resultados definidos pelo Proprietário do Produto. |

Quadro 4 - Equipe de Desenvolvimento - Atividades

## PLANO DE ACOMPANHAMENTO

### Marcos de ponto e controle

São definidos como marcos do projeto o término de cada Sprint. No final do período do Sprint, a reunião de revisão do Sprint é feita. Nesta reunião, a equipe de desenvolvimento, junto ao *Scrum Master*, se reúne com o Proprietário do Produto.

### 

### Métodos de acompanhamento e controle

As tarefas serão acompanhadas mensalmente pelos integrantes do grupo e pelo orientador do trabalho através de reuniões com o professor Braga às Quartas-feiras, a fim de garantir o cumprimento do cronograma e para a resolução de possíveis problemas. A implementação será apoiada pelo uso de ferramentas para gerenciamento de *Scrum.*

### Análise e gerência de riscos

Quadro 5 - Riscos

|  |  |  |  |
| --- | --- | --- | --- |
| Etapas | Risco | Contingência | Prioridade |
| Comunicação | -Indisponibilidade da equipe;  - Levantamento de requisitos deficiente;  - Alteração nos requisitos. | - Remarcação de Reuniões;  - Manter constante interação entre os membros do grupo;  - Ajustar prazos, prioridades e custos. | Alta |
| Planejamento | - Atraso no cronograma | - Ajustar cronograma | Alta |
| Modelagem | - Desenvolvimento de casos de usos inadequados ao sistema;  - Escolha de ferramentas inadequadas. | - Revisão dos casos de uso pelo orientador do projeto;  - Solicitar ao orientador, indicação de ferramentas adequadas à modelagem | Alta |
| Construção | - Desenvolvimento incorreto das funcionalidades;  - Testes insuficientes para descobrir falhas. | - Realizar testes, a fim de descobrir tal ocorrência e corrigi-la;  - Realização de testes mais aprofundados no sistema. | Alta |

## PLANO DE DOCUMENTAÇÃO

Com o objetivo de melhorar a qualidade do software produzido, os seguintes documentos foram elaborados durante o desenvolvimento do projeto, os quais foram relacionados nesta documentação, conforme abaixo:

Quadro 6 - Plano de Documentação

|  |  |  |
| --- | --- | --- |
| Documentos gerados | Objetivo | Fase do Ciclo de Vida |
| Proposta de Pré-Projeto | Propor um possível trabalho a ser desenvolvido e sua motivação. | Planejamento |
| Proposta de Processo de desenvolvimento | Especificar a metodologia de desenvolvimentos que se pretende utilizar | Planejamento |
| Plano de Projeto | Planejar a abordagem que se pretende dar ao tema e como se pretende atingir o objetivo final | Planejamento |
| Requisitos do sistema | Identificar o que o sistema necessita para ser desenvolvimento | Planejamento e Análise |
| Documento de visão | Similar ao Plano de projeto, seu objetivo é fornecer uma visão ampla do produto que se pretende desenvolver, sem se aprofundar em detalhes | Planejamento e Análise |
| Modelo entidade relacionamento | Um modelo conceitual que será utilizado para orientar o desenvolvimento propriamente dito, fornecendo informações sobre os aspectos relacionados ao domínio do projeto em questão. | Planejamento e Análise |
| Dicionário de dados | Gerar a documentação de projeto e implementação gerados na ferramenta case | Projeto e implementação |

## CRONOGRAMA

Quadro 7 - Cronograma com macro atividades do Projeto (2017)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Atividade | Ago 17 | Set 17 | Out 17 | Nov 17 | Dez 17 |
| Escolha do Tema (CBL) | ☒ | ☐ | ☐ | ☐ | ☐ |
| Pesquisas sobre o Tema(CBL) | ☐ | ☒ | ☒ | ☐ | ☐ |
| Definição de Requisitos(CBL) | ☐ | ☐ | ☒ | ☒ | ☒ |
| Coleta de Dados(CBL) | ☐ | ☐ | ☒ | ☒ | ☒ |
| Definição das Tecnologias(CBL) | ☐ | ☐ | ☐ | ☒ | ☒ |
| Reuniões | ☒ | ☒ | ☒ | ☒ | ☒ |

Quadro 8 – Cronograma com macro atividades do Projeto (2018)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Atividade | Fev 18 | Mar 18 | Abr 18 | Mai 18 | Jun 18 | Jul 18 |
| Análise de especificações funcionais | ☒ | ☐ | ☐ | ☐ | ☐ | ☐ |
| Desenvolvimento e Codificação | ☐ | ☒ | ☒ | ☒ | ☒ | ☒ |
| Testes de Unidades | ☐ | ☐ | ☒ | ☒ | ☒ | ☒ |
| Construção da documentação | ☐ | ☐ | ☒ | ☒ | ☒ | ☒ |
| Apresentação | ☐ | ☐ | ☐ | ☐ | ☐ | ☒ |
| Reuniões | ☒ | ☒ | ☒ | ☒ | ☒ | ☒ |

# REFERENCIAL TEÓRICO

## SOBRE O desenvolvimento ágil

O desenvolvimento de software adaptativo e incremental que é a base do desenvolvimento ágil vem sendo estudado desde os anos 70. O uso de métodos ágeis no desenvolvimento de software flexibiliza o processo de forma a utilizar melhoria continua, entrega incremental, como também se adapta rapidamente a mudanças de requisitos e tecnologias. A metodologia ágil foca mais em aspectos humanos de engenharia de software do que processos, interações humanas antes de ferramentas e processos. Neste sentido, um mal entendimento comum é que os projetos ágeis em geral não têm equipes que criam e mantém documentação de projeto.

O desenvolvimento ágil é considerado atualmente uma das principais abordagens de desenvolvimento de software. Além disso, os métodos ágeis são considerados adequados para diversos tipos de projetos, e as empresas tem aumentado o reconhecimento sobre a necessidade de agilidade. Entretanto, os métodos ágeis também possuem desafios em alguns casos como falta de planejamento de arquitetura, uma ênfase muito grande em resultados antecipados e um baixo nível de cobertura de teste.

Existem diferentes métodos ágeis de desenvolvimento de software, em 2010, Hasnain (E. Hasnain, 2010) realizou uma revisão da literatura que ajudou pesquisadores a determinar o estado da arte na pesquisa sobre desenvolvimento ágil, reportando que nas conferências da IEEE tem aumentado a cada ano o número de estudos sobre os métodos Scrum e Extreme Programming (XP). Da mesma forma, anteriormente Tore e Torgeir (T. Dybå and T. Dingsør, 2008) também realizaram uma revisão da literatura sobre estudos empíricos de desenvolvimento ágil, fornecendo uma base dos estudos correntes e referências sobre métodos ágeis de desenvolvimento. Sendo assim, existem diferentes métodos ágeis de desenvolvimento e diferentes razões para a adoção do desenvolvimento ágil.

Baseado em um estudo sobre a evolução do desenvolvimento ágil de software no Brasil publicado por Melo et al., foi possível obter um panorama das principais razões na adoção de métodos ágeis, como também a percepção dos benefícios:

* Acelerar o time-to-market.
* Aumento de produtividade.
* Melhorar a disciplina técnica.
* Melhorar a habilidade de mudanças.
* Melhorar a manutenção do software.
* Melhorar a qualidade do software.
* Melhorar o alinhamento entre TI e área de negócios.
* Melhorar o moral da equipe.
* Reduzir custos.
* Reduzir riscos.
* Simplificar o processo de desenvolvimento.
* Visibilidade do projeto.

## sobre o método scrum com cbl

Quadro 9 - Principais características do CBL

|  |  |
| --- | --- |
| Nome | Descrição |
| Big Idea | Um conceito amplo que pode ser explorado de diversas formas, é envolvente, e tem importância para os alunos e a sociedade em geral. |
| Essential Question | Deve identificar o que é importante a ser descoberto da Big Idea e refinar a ideia. |
| Challenge | Os estudantes devem articular um desafio para criar uma resposta específica ou solução que pode resultar em ação concreta. |
| Guiding Questions | Representa o conhecimento que os estudantes precisam descobrir para atingir o desafio. |
| Guiding Activities | Qualquer atividade que possa ajudar os estudantes a responderem as guiding questions. |
| Guiding Resources | Recursos diversos que podem ser podcasts, websites, vídeos, banco de dados, artigos ou especialistas. |
| Solution | Cada desafio é indicado de forma suficientemente ampla para permitir uma variedade de soluções que devem ser concretas, atingíveis e claramente articuladas. |
| Assessment | A solução pode ser avaliada pela sua conexão com o desafio, precisão do conteúdo, clareza de comunicação, aplicabilidade para implementação, etc. |
| Publishing | O desafio proporciona múltiplas oportunidades de documentar a experiência e publicação da mesma. Os estudantes são encorajados a publicar resultados e obter feedback. |

Quadro 10 - Principais características do Scrum

|  |  |
| --- | --- |
| Fase | Descrição |
| Planejamento | Esta fase envolve diversas atividades, entre elas as principais que se destacam são o desenvolvimento de uma lista compreensiva de requisitos, chamada de “product backlog”, definir os objetivos e datas de uma ou mais releases e selecionar a release mais apropriada para o desenvolvimento imediato, de acordo com as prioridades definidas no product backlog. |
| Arquitetura/Design | Revisar os itens do backlog de forma a identificar mudanças necessárias, refinar a arquitetura do sistema para suportar os requisitos a serem implementados. |
| Desenvolvimento (“Sprint”) | Esta fase é um ciclo iterativo de trabalho de desenvolvimento, com Sprints iterativos até que o produto esteja pronto para distribuição. O Sprint é um conjunto de atividades de desenvolvimento que devem ser conduzidas em um período de uma a quatro semanas. Para cada Sprint deve ser realizado uma revisão onde todas as equipes apresentam o seu progresso de trabalho, levantando e resolvendo problemas como também atualizando o backlog, neste momento os riscos são revisados e respostas são definidas. |
| Fechamento | Esta fase prepara o produto desenvolvido para release. |

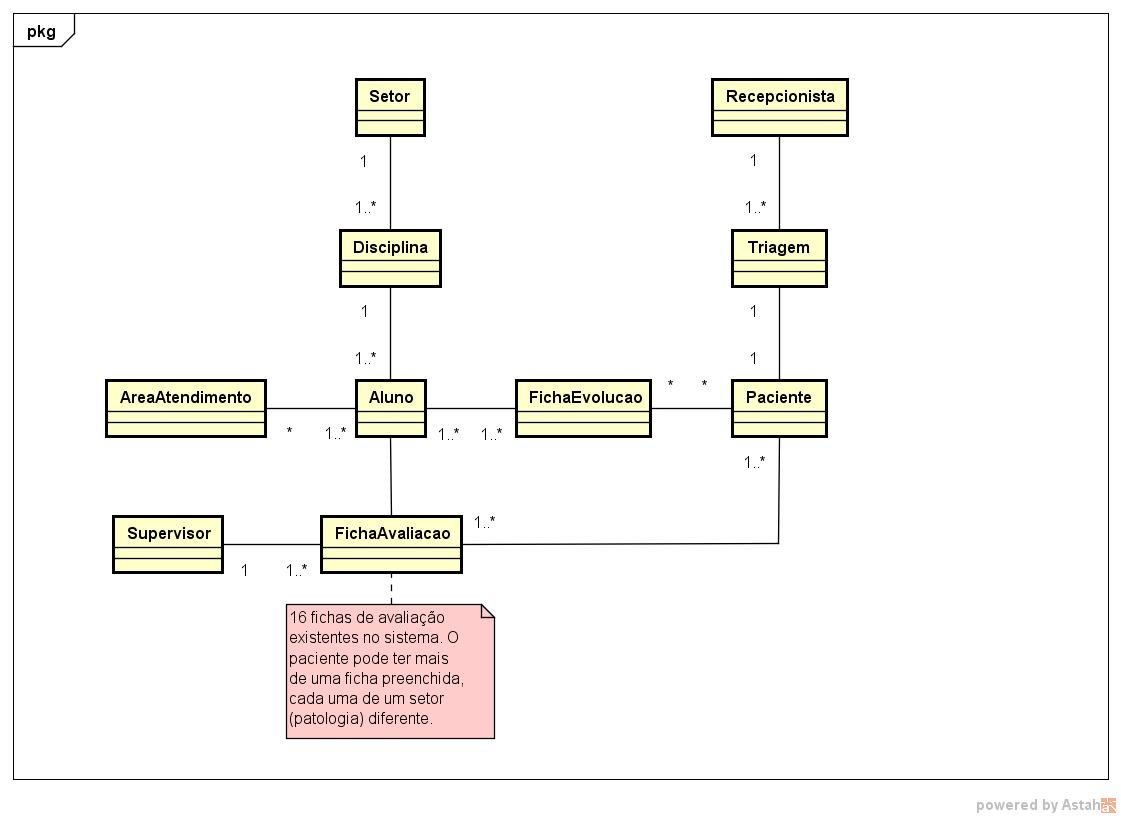
A estrutura base do método basicamente é composta pelos seguintes estágios da abordagem baseada em desafios: Big Idea, Essential Question, Challenge, Implementation, Solution e Evaluation. Assim, analisando estes estágios com os estágios descritos no Quadro 8 os estágios Assessment e Publication foram substituídos pelos estágios Implementation e Evaluation propostos em uma evolução do método CBL, aspecto que vai de encontro a uma maneira de integrar práticas ágeis na proposta CBL.

Este método inclui práticas ágeis nos estágios de Solution, Implementation e Evaluation. O estágio de solution engloba o product backlog, o estágio de implementation contempla a iteração ágil, incluindo planejamento de sprint, reuniões diárias e entrega do produto incremental. Assim, o estágio de evaluation inclui as revisões de sprint e retrospectivas de sprint de forma a consolidar alguns dos principais conceitos do framework scrum, incorporando também práticas ágeis. As fases do método são apresentadas na Figura 2 onde tratamos sobre o ciclo de vida do projeto.

# ESPECIFICAÇÃO DE REQUISITOS DO SISTEMA

## DIAGRAMA DE CLASSE DE DOMINIO

Figura 7 - Diagrama de classe de domínio do sistema



## TELAS DO SISTEMA

Figura 8 - Tela de login

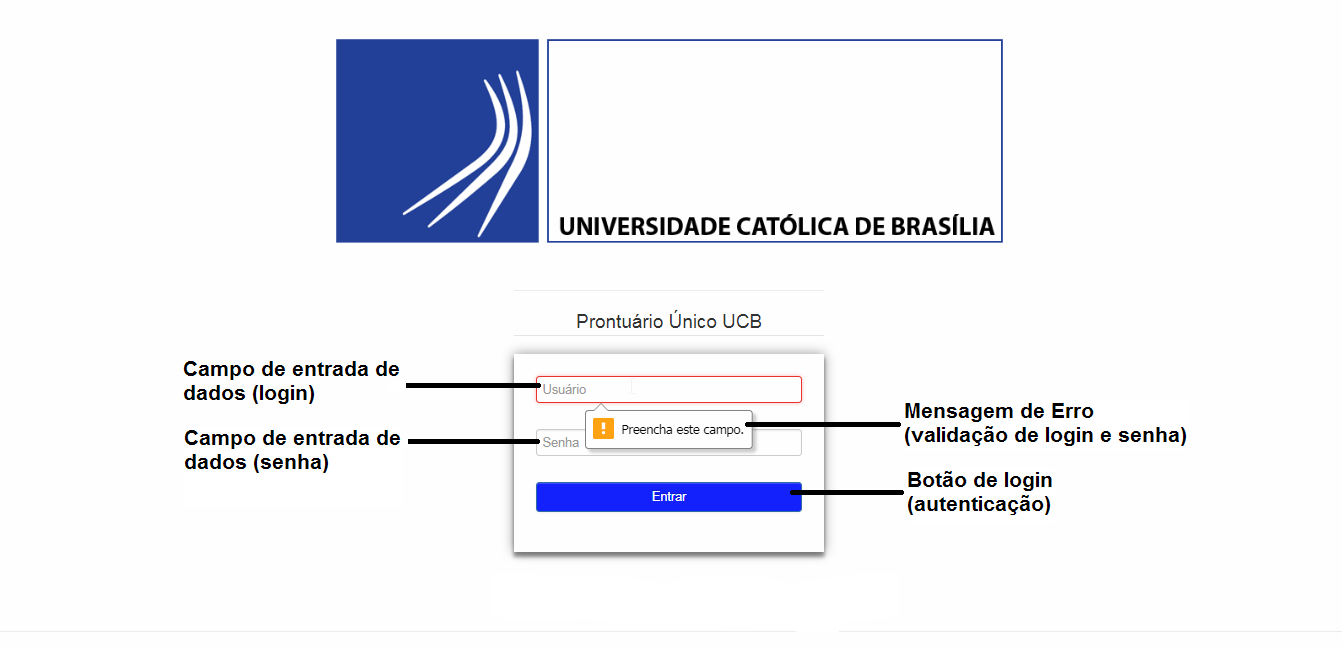


Figura 9 - Tela principal do sistema

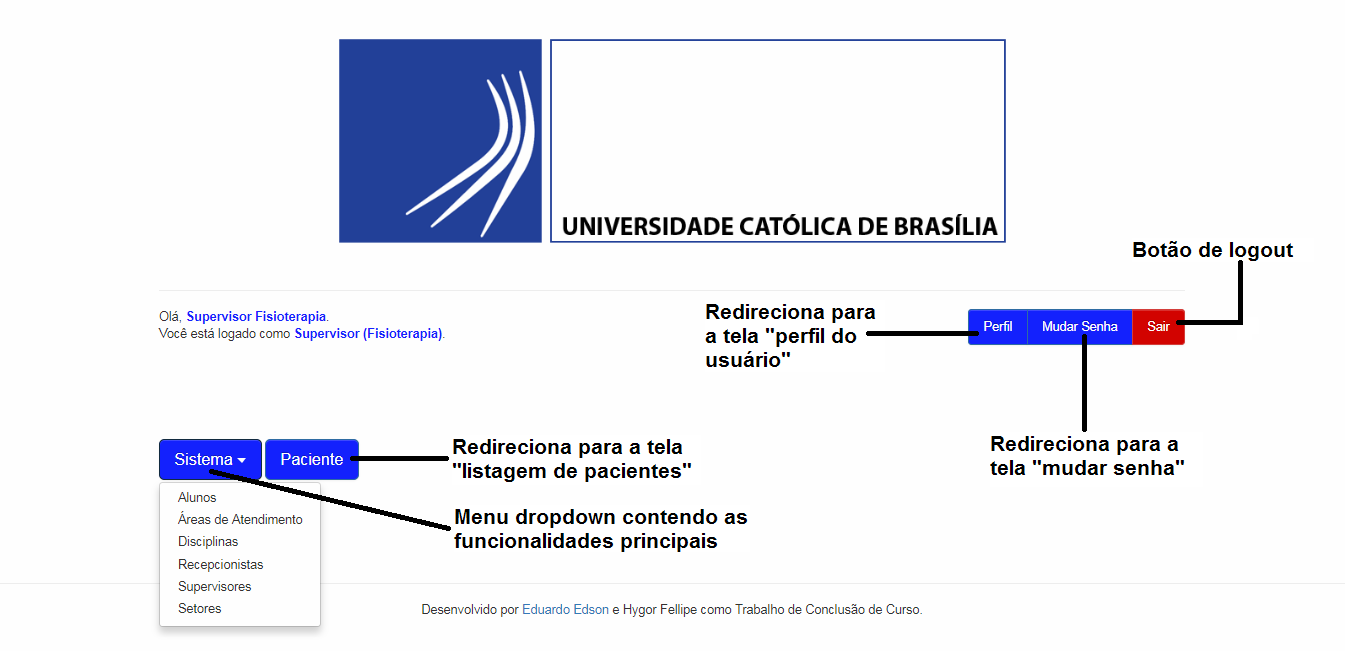


Figura 10 - Tela de perfil do usuário autenticado no sistema (neste exemplo, perfil de supervisor)

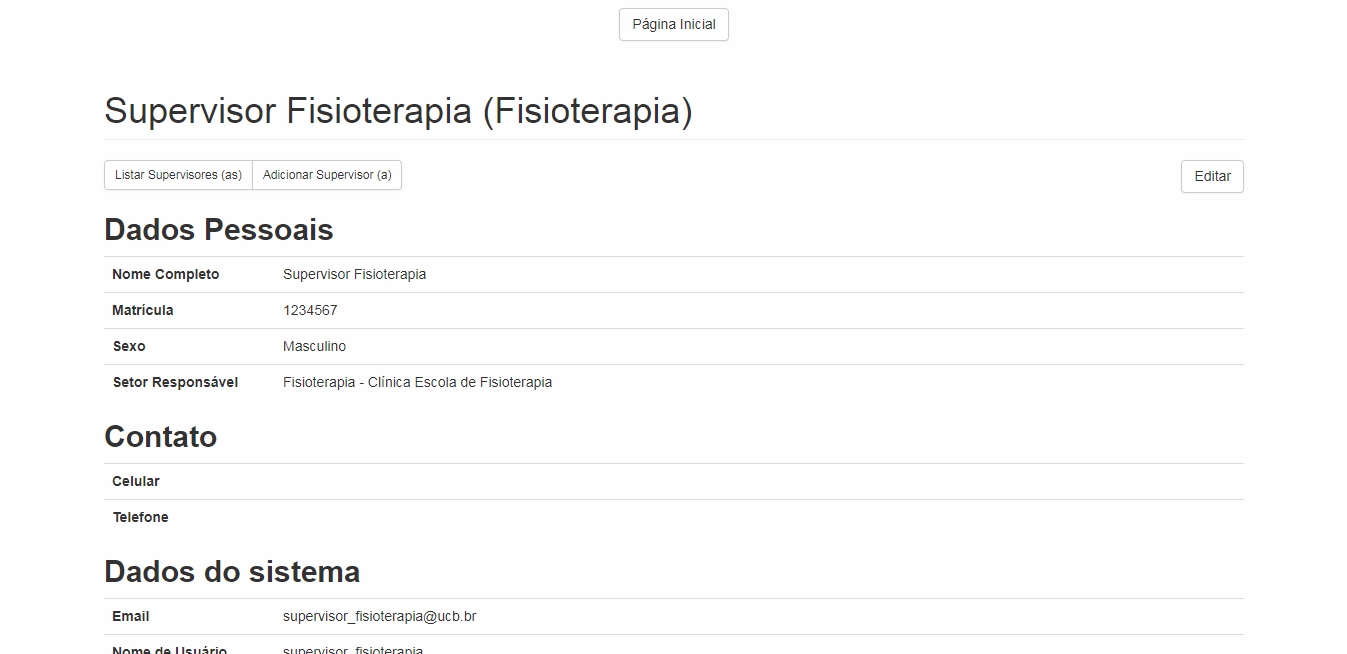


Figura 11 - Tela de alteração de senha do usuário



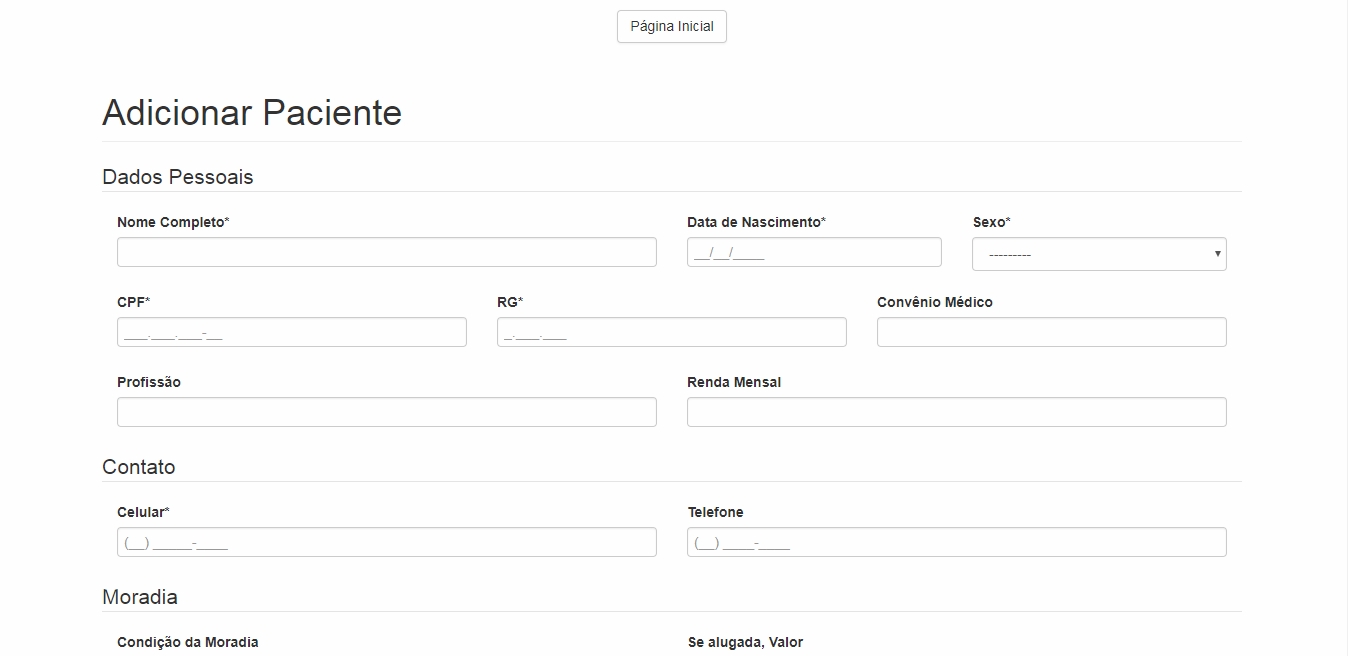
Figura 12 - Tela de cadastro de paciente 

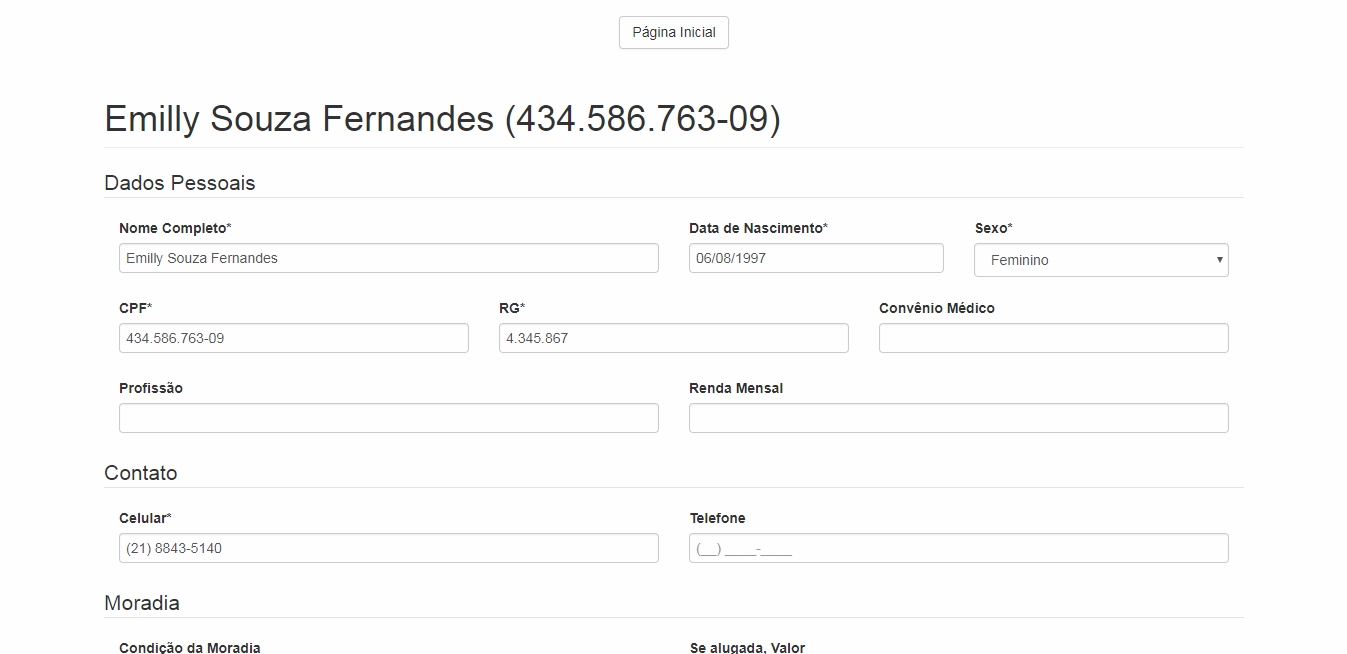
Figura 13 - Tela de alteração de dados do paciente 

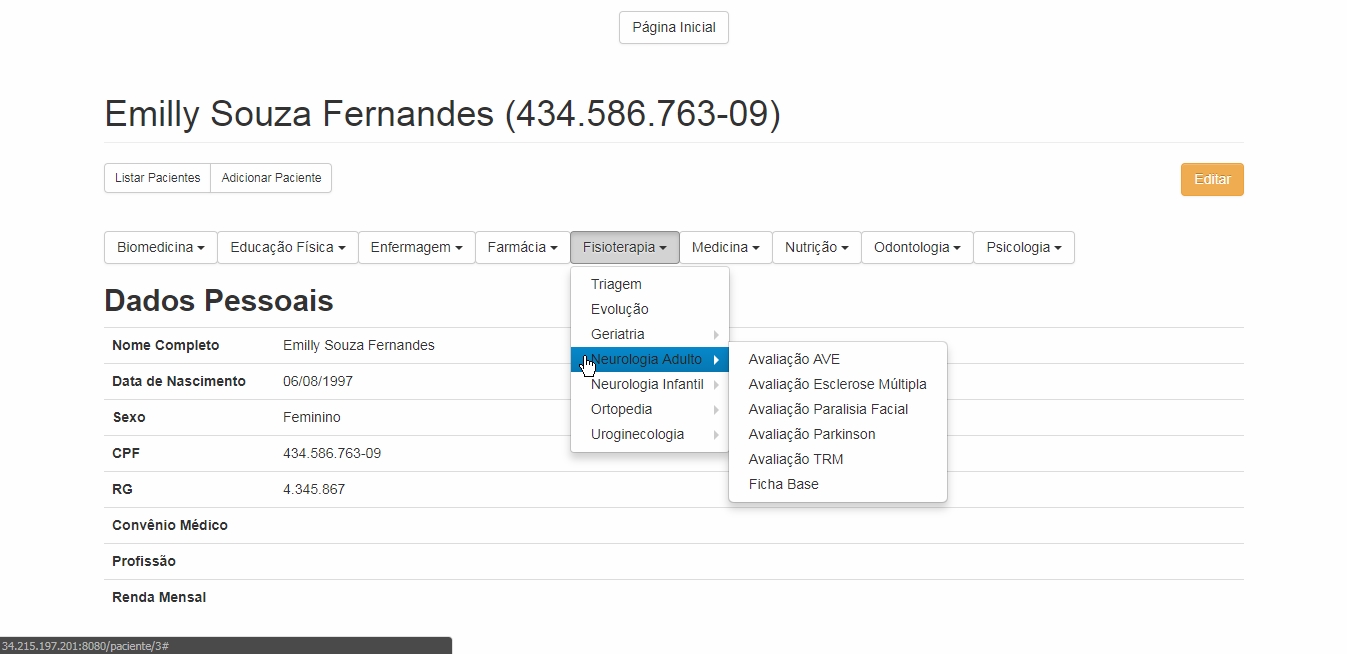
Figura 14 - Tela de detalhamento do paciente, contendo menu em cascata para cada área da saúde, e no menu de fisioterapia o paciente pode ter suas fichas específicas preenchidas. Além disso estão detalhadas as informações pessoais.

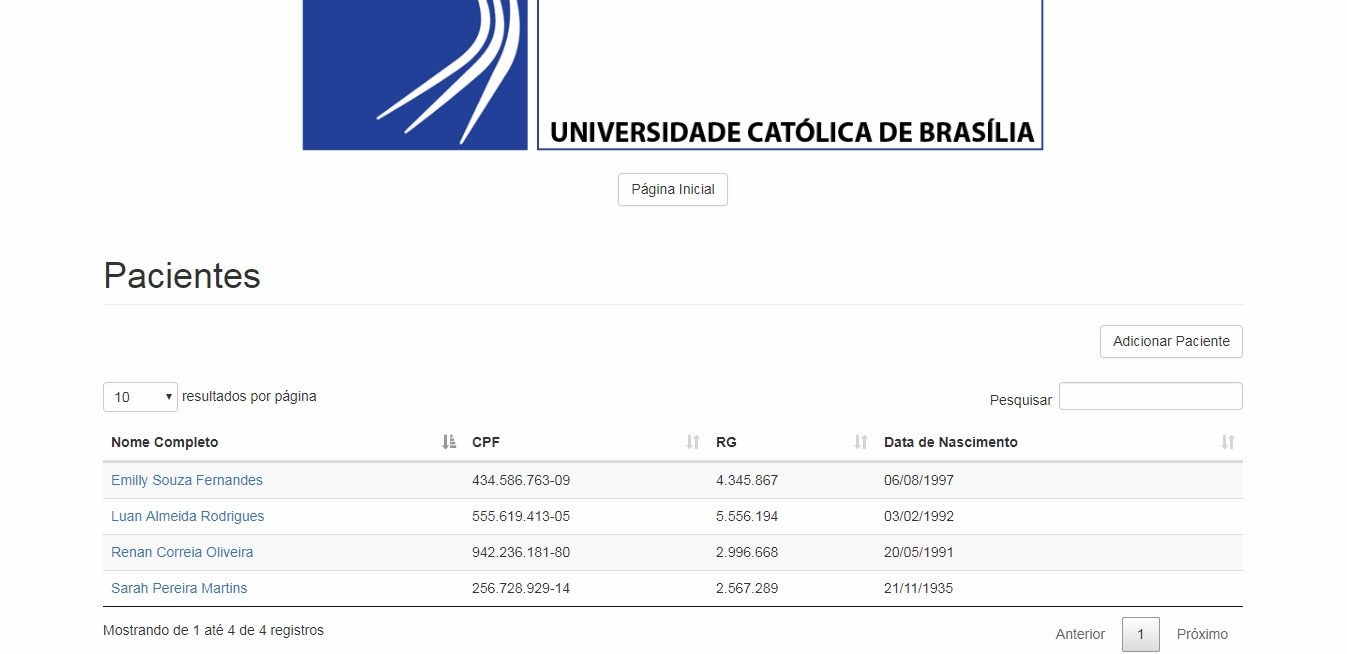
Figura 15 - Tela de listagem de pacientes cadastrados 

Figura 16 - Tela de cadastro de área de atendimento (específica dentro de cada clínica) 

Figura 17 - Tela de alteração de dados da área de atendimento 

Figura 18 - Tela de detalhamento da área de atendimento 

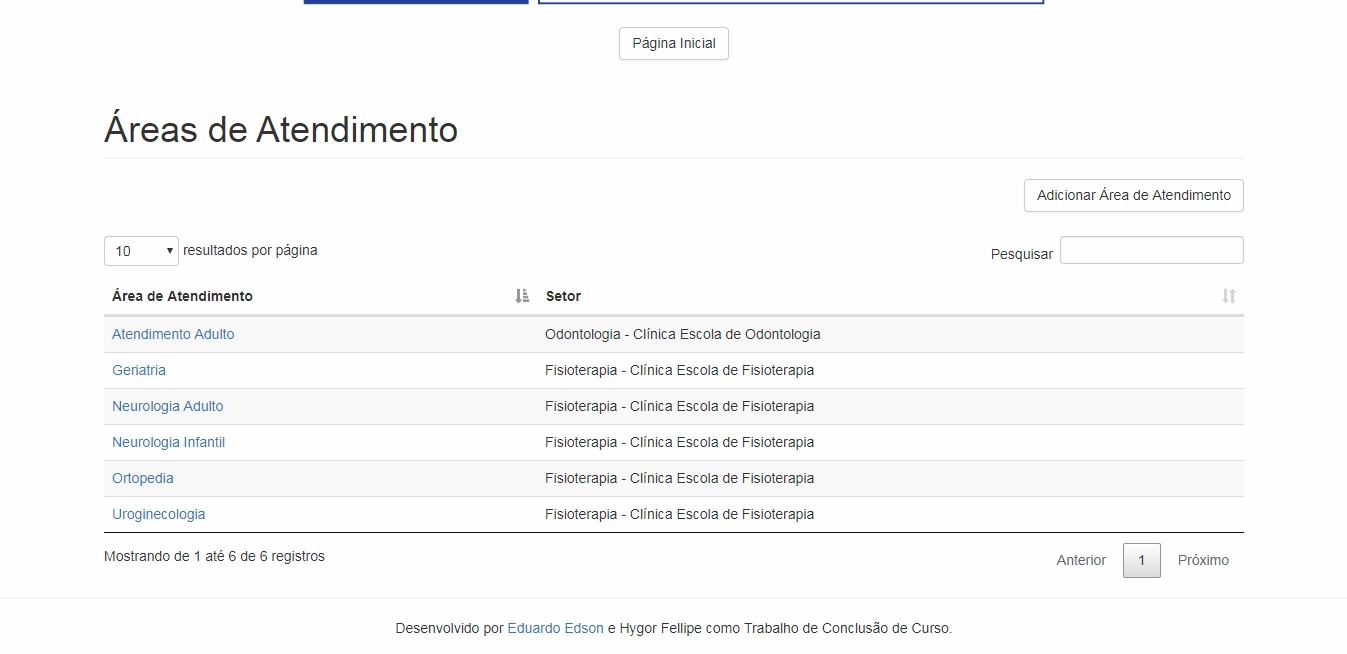
Figura 19 - Tela de listagem das áreas de atendimento 

Figura 20 - Tela de cadastro de disciplina



Figura 21 - Tela de alteração de dados de uma disciplina



Figura 22 - Tela de detalhamento de uma disciplina



Figura 23 - Tela de listagem das disciplinas

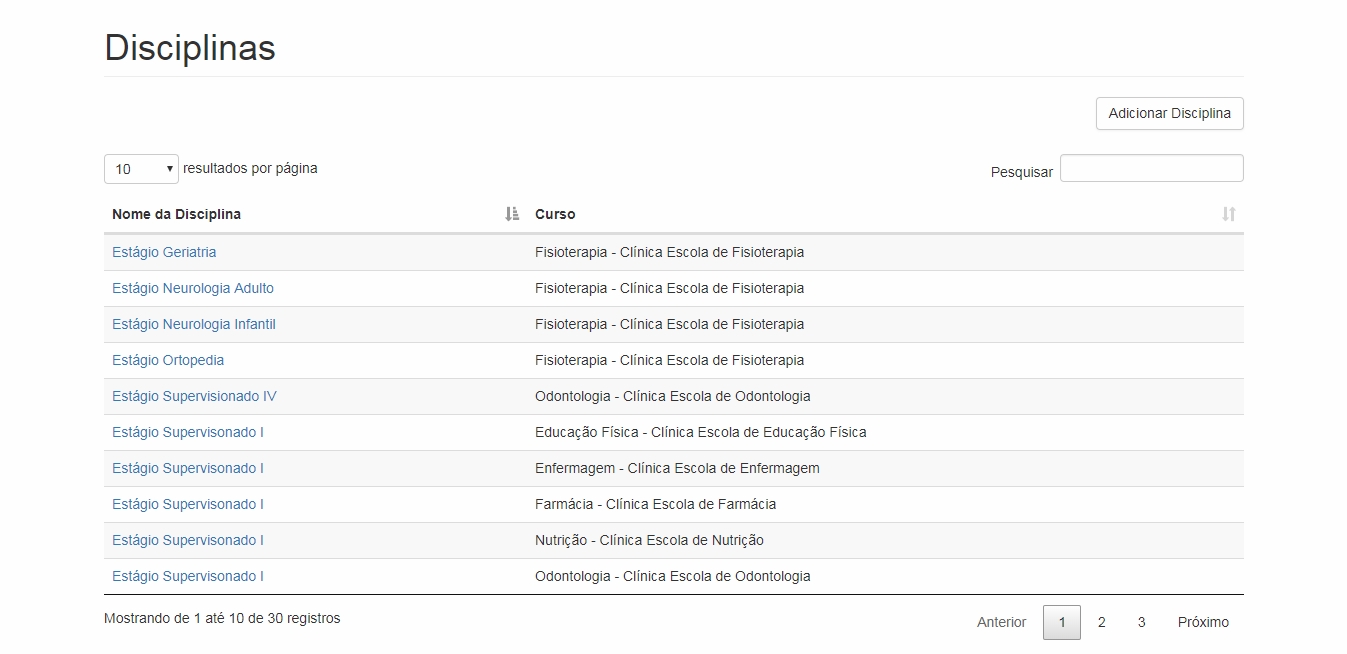


Figura 24 - Tela de cadastro de recepcionista

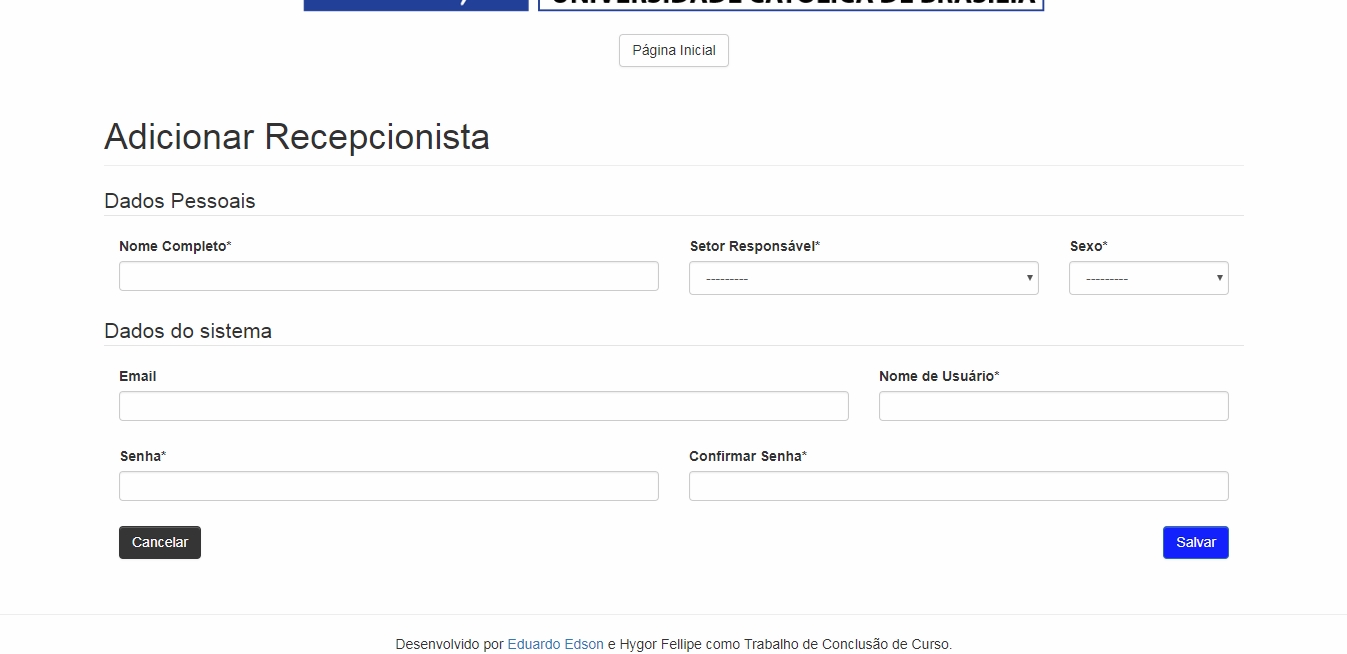


Figura 25 - Tela de alteração de dados de uma recepcionista



Figura 26 - Tela de detalhamento de recepcionista

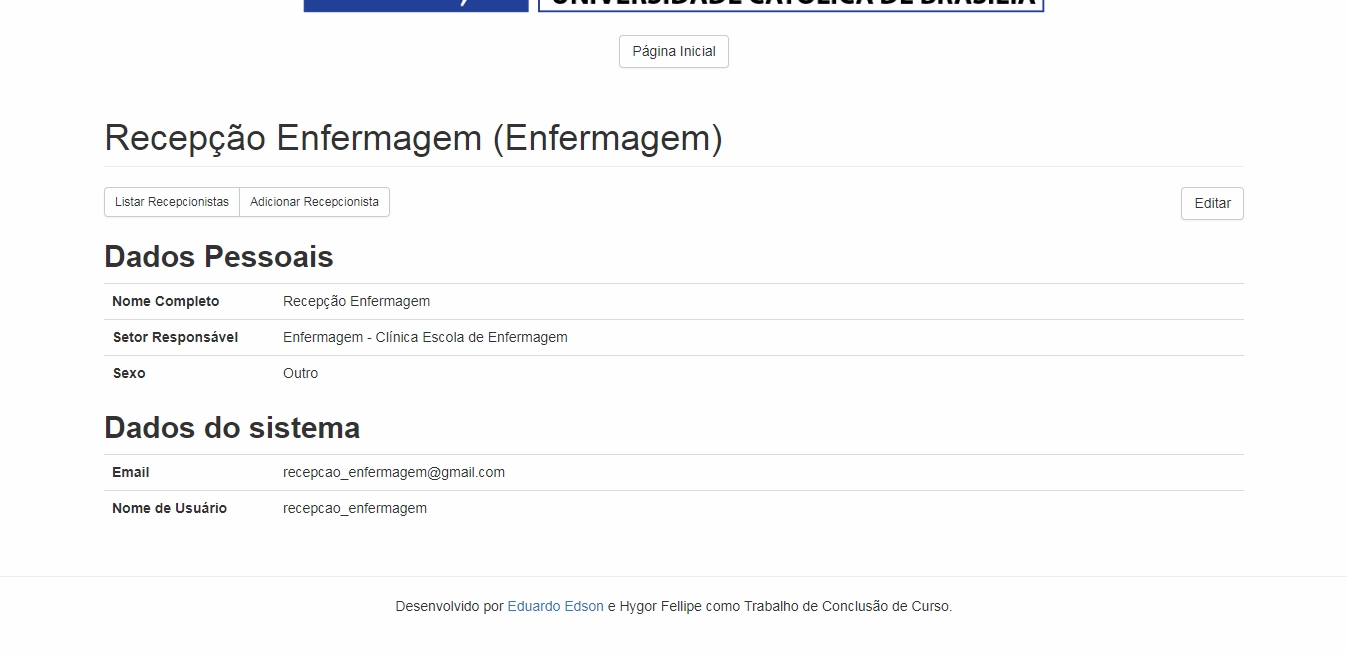


Figura 27 - Tela de listagem de recepcionistas



Figura 28 - Tela de cadastro de supervisor

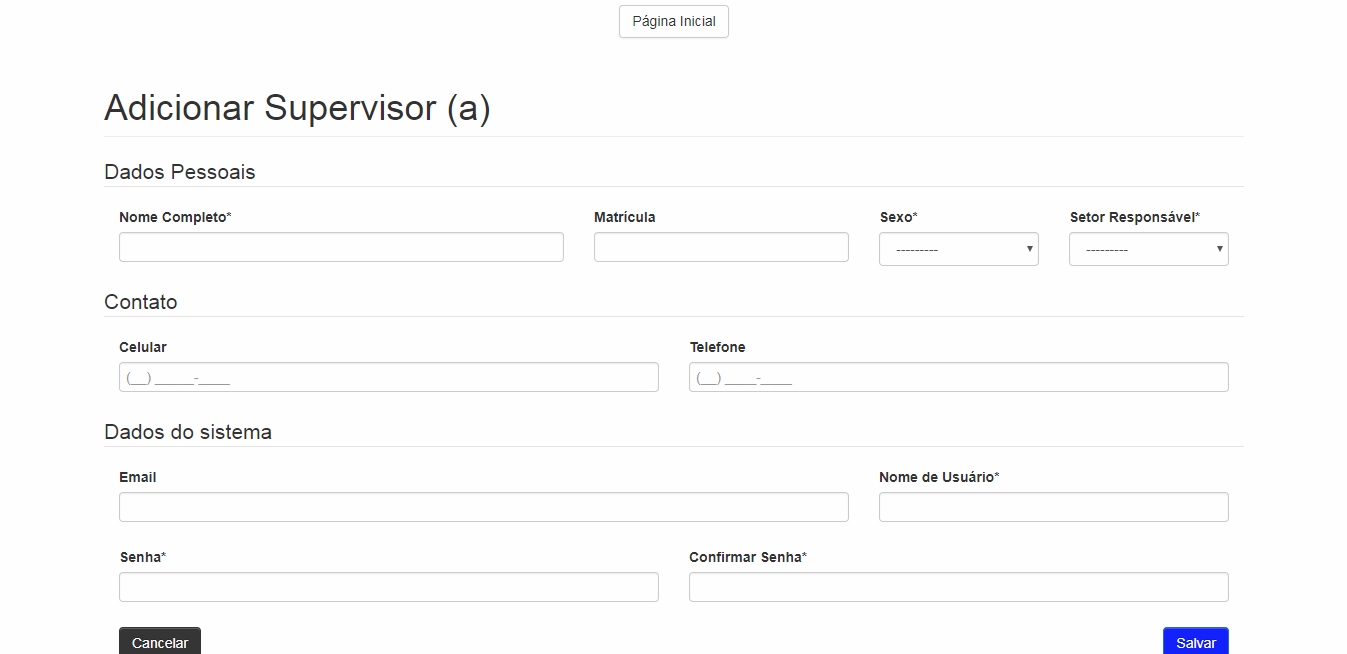


Figura 29 - Tela de alteração de dados de supervisor

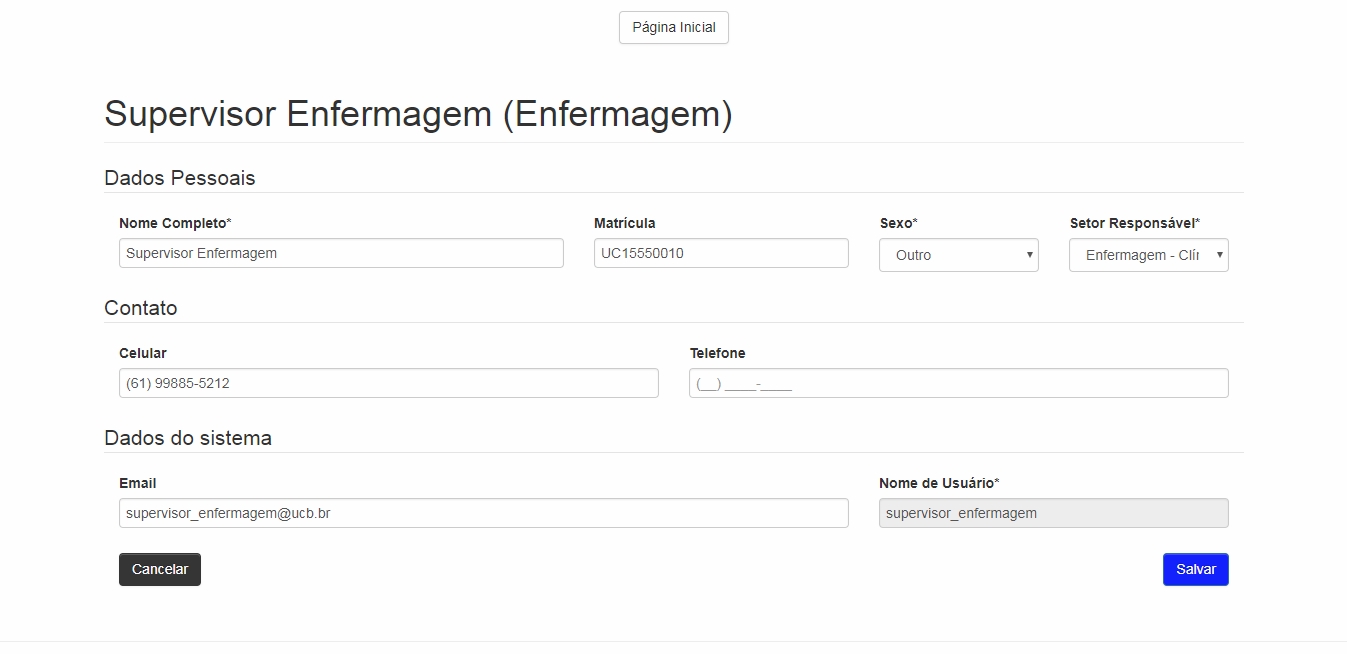


Figura 30 - Tela de detalhamento de supervisor



Figura 31 - Tela de listagem de supervisores



Figura 32 – Tela de cadastro de setor



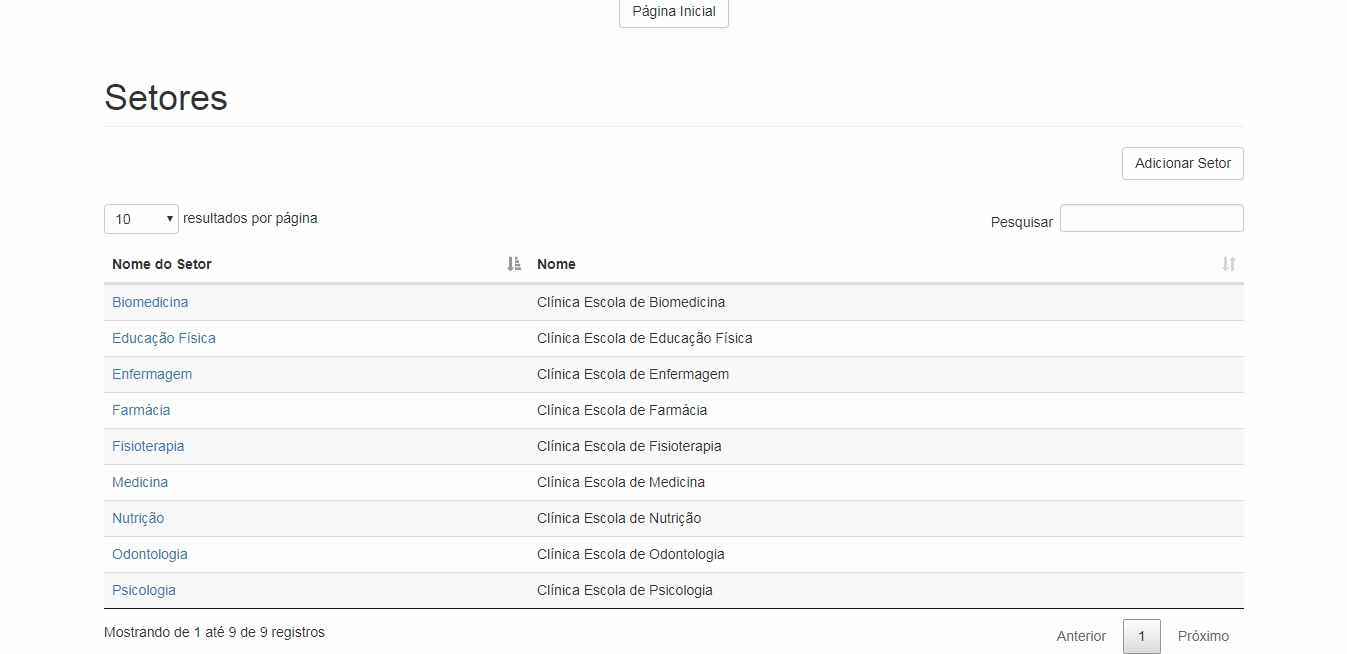
Figura 33 - Tela de alteração de dados de setor



Figura 34 – Tela de detalhamento de setor contendo as disciplinas vinculadas



Figura 35 - Tela de listagem de setores



# DOCUMENTAÇÃO DO PROJETO

## DICIONÁRIO DE DADOS

Quadro 11 - Dicionário de dados – servicos\_areaatendimento

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Tabela | Coluna | Tipo | PK | Null | Descrição |
| servicos\_areaatendimento | id | integer | Sim | Não | Chave Primária. |
| servicos\_areaatendimento | descricao | varchar(30) | Não | Não | Nome do local onde ocorre o atendimento. |
| servicos\_areaatendimento | setor\_id | integer | Sim | Não | Chave primária do setor a qual essa área de atendimento pertence. |

Quadro 12 - Dicionário de dados – servicos\_disciplina

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Tabela | Coluna | Tipo | PK | Null | Descrição |
| servicos\_disciplina | id | integer | Sim | Não | Chave Primária. |
| servicos\_disciplina | descricao | varchar(50) | Não | Não | Nome da disciplina. |
| servicos\_disciplina | setor\_id | integer | Sim | Não | Chave primária do setor a qual essa disciplina pertence. |

Quadro 13 - Dicionário de dados – servicos\_setor

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Tabela | Coluna | Tipo | PK | Null | Descrição |
| servicos\_setor | id | integer | Sim | Não | Chave Primária. |
| servicos\_setor | descricao | varchar(30) | Não | Não | Nome do setor (Curso). |
| servicos\_setor | nome | varchar(50) | Não | Não | Nome dado a clínica do curso (Setor). |

Quadro 14 - Dicionário de dados – usuarios\_aluno

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Tabela | Coluna | Tipo | PK | Null | Descrição |
| usuarios\_aluno | id | integer | Sim | Não | Chave Primária. |
| usuarios\_aluno | nome | varchar(80) | Não | Não | Nome do aluno. |
| usuarios\_aluno | sexo | varchar(1) | Não | Não | Sexo do aluno. |
| usuarios\_aluno | matricua | varchar(13) | Não | Não | Matrícula do aluno. |
| usuarios\_aluno | telefone | varchar(14) | Não | Sim | Telefone para contato. |
| usuarios\_aluno | celular | varchar(15) | Não | Não | Celular para contato. |
| usuarios\_aluno | email | varchar(254) | Não | Não | Email para cadastro. |
| usuarios\_aluno | username | varchar(30) | Não | Não | Usuário para autenticação no sistema. |
| usuarios\_aluno | disciplina\_id | integer | Sim | Não | Chave primária da disciplina que este aluno está cursando para ter direito a acessar o sistema. |
| usuarios\_aluno | supervisor\_id | integer | Sim | Não | Chave primária do supervisor que controla as ações do aluno no sistema. |

Quadro 15 - Dicionário de dados – usuarios\_paciente

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Tabela | Coluna | Tipo | PK | Null | Descrição |
| usuarios\_paciente | id | integer | Sim | Não | Chave Primária. |
| usuarios\_paciente | nome | varchar(80) | Não | Não | Nome do paciente. |
| usuarios\_paciente | sexo | varchar(1) | Não | Não | Sexo do paciente. |
| usuarios\_paciente | data\_nascimento | date | Não | Não | Data de nascimento do paciente. |
| usuarios\_paciente | rg | varchar(9) | Não | Não | RG do paciente. |
| usuarios\_paciente | cpf | varchar(14) | Não | Não | CPF do paciente. |
| usuarios\_paciente | profissao | varchar(30) | Não | Não | Qual profissão o paciente exerce. |
| usuarios\_paciente | renda | varchar(14) | Não | Não | Renda pessoal do paciente. |
| usuarios\_paciente | convenio | varchar(14) | Não | Não | Convênio médico do paciente. |
| usuarios\_paciente | grupo\_familiar | varchar(3) | Não | Não | Quantas pessoas mora com o paciente. |
| usuarios\_paciente | renda\_total | varchar(14) | Não | Não | Renda da soma de todas as pessoas que moram junto ao paciente. |
| usuarios\_paciente | telefone | varchar(14) | Não | Sim | Telefone para contato. |
| usuarios\_paciente | celular | varchar(15) | Não | Nao | Celular para contato. |
| usuarios\_paciente | cep | varchar(10) | Não | Não | Cep da residência do paciente. |
| usuarios\_paciente | endereco | varchar(50) | Não | Não | Endereço completo do paciente. |
| usuarios\_paciente | numero | varchar(5) | Não | Não | Número como complemento do endereço. |
| usuarios\_paciente | complemento | varchar(50) | Não | Não | Complemento do endereço. |
| usuarios\_paciente | bairro | varchar(30) | Não | Não | Bairro da residência do paciente. |
| usuarios\_paciente | referencia | varchar(30) | Não | Não | Referência para fácil localização do endereço do paciente. |
| usuarios\_paciente | uf | varchar(2) | Não | Não | Estado que o paciente mora. |
| usuarios\_paciente | data\_cadastro | date | Não | Não | Data em que o paciente foi cadastrado no sistema. |
| usuarios\_paciente | colaborador | varchar(10) | Não | Não | Informa se o paciente é colaborador da instituição (UCB). |
| usuarios\_paciente | moradia | varchar(30) | Não | Não | Informa o estado da moradia do paciente. |
| usuarios\_paciente | valor\_aluguel | varchar(30) | Não | Não | Valor que o paciente paga de aluguel em sua moradia (caso seja alugada). |
| usuarios\_paciente | tratamento\_sus | varchar(10) | Não | Não | Informa se o paciente recebe algum tratamento pelo SUS. |
| usuarios\_paciente | beneficio | varchar(10) | Não | Não | Informa se o paciente recebe algum benefício do governo. |
| usuarios\_paciente | referencia\_nome | varchar(80) | Não | Não | Nome da pessoa de referência do paciente. |
| usuarios\_paciente | referencia\_telefone | varchar(15) | Não | Não | Telefone da pessoa de referência do paciente. |
| usuarios\_paciente | referencia\_parentesco | varchar(80) | Não | Não | Grau de parentesco da referência do paciente. |

Quadro 16 - Dicionário de dados – usuarios\_recepcionista

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Tabela | Coluna | Tipo | PK | Null | Descrição |
| usuarios\_recepcionista | id | integer | Sim | Não | Chave Primária. |
| usuarios\_recepcionista | nome | varchar(80) | Não | Não | Nome do recepcionista. |
| usuarios\_recepcionista | sexo | varchar(1) | Não | Não | Sexo do recepcionista. |
| usuarios\_recepcionista | email | varchar(254) | Não | Não | E-mail para cadastro. |
| usuarios\_recepcionista | username | varchar(30) | Não | Não | Usuário para autenticação no sistema. |
| usuarios\_recepcionista | setor\_id | integer | Sim | Não | Chave primária do setor (Curso) que o recepcionista é responsável. |

Quadro 17 - Dicionário de dados – usuarios\_supervisor

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Tabela | Coluna | Tipo | PK | Null | Descrição |
| usuarios\_supervisor | id | integer | Sim | Não | Chave Primária. |
| usuarios\_supervisor | nome | varchar(80) | Não | Não | Nome do recepcionista. |
| usuarios\_supervisor | sexo | varchar(1) | Não | Não | Sexo do recepcionista. |
| usuarios\_supervisor | matricula | varchar(13) | Não | Não | Matricula do professor na instituição (UCB). |
| usuarios\_supervisor | telefone | varchar(14) | Não | Sim | Telefone para contato. |
| usuarios\_supervisor | celular | varchar(15) | Não | Não | Celular para contato. |
| usuarios\_supervisor | email | varchar(254) | Não | Não | E-mail para cadastro. |
| usuarios\_supervisor | username | varchar(30) | Não | Não | Usuário para autenticação no sistema. |
| usuarios\_supervisor | setor\_id | integer | Sim | Não | Chave primária do setor (Curso) que o professor é responsável. |

# CONCLUSÃO

Este trabalho mostrou que é factível o desenvolvimento de um projeto através do modelo proposto de *Scrum* e *CBL* em conjunto. Após o estudo realizado, foi possível gerar um protótipo funcional seguindo a metodologia e documentações.

Partindo do princípio que este protótipo servirá como base para uma futura evolução, o trabalho realizado buscou evidenciar a necessidade e a importância de tratar, de forma unificada e organizada, as informações dos pacientes que percorrem cada uma das clínicas da universidade, visto que o ecossistema da saúde dentro do campus é adequado para dar início a uma espécie de trabalho colaborativo entre as clínicas-escola, que poderiam se beneficiar ao compartilhar entre si as informações de prontuários dos pacientes atendidos em suas respectivas áreas para que, em cada ambulatório, os estagiários tenham em mãos o histórico do paciente advindo de outros setores dentro da instituição, e usem esse recurso como auxílio para a tomada de decisões, junto ao supervisor responsável.

Considerando o cenário descrito anteriormente, a prova de conceito cumpriu o objetivo proposto através do desenvolvimento do protótipo, intitulado Prontuário Único UCB, como pontapé inicial para uma evolução na maneira como a universidade e comunidade se relacionam no âmbito da saúde e qualidade de vida.

# REFERÊNCIAS BIBLIOGRÁFICAS

BASILI, V.; BRIAND, L.; MELO, W. A validation of object-oriented design metrics as quality indicators. Software Engineering, IEEE Transactions on, 1996. v. 22, n. 10, p. 751–761, Oct 1996. ISSN 0098-5589.

C. O. Melo, V. Santos, E. Katayama, H. Corbucci, R. Prikladnicki, A. Goldman, and F. Kon. The evolution of agile software development in Brazil. Journal of the Brazilian Computer Society,19(4):523–552, 2013.

COSTA, Claudio Giulliano Alves da. Cartilha sobre Prontuário Eletrônico - A Certificação de Sistemas de Registro Eletrônico de Saúde. 2012. Disponível em: < https://portal.cfm.org.br/crmdigital/Cartilha\_SBIS\_CFM\_Prontuario\_Eletronico\_fev\_2012.pdf>. Acesso em: 26 set. 2017.

E. Hasnain. An Overview of Published Agile Studies: A Systematic Literature Review. In Proceedings of the 2010 National Software Engineering Conference, NSEC ’10, pages 3:1–3:6, Rawalpindi, Pakistan, 2010.

PRESSMAN, Roger S.; TRAVIESO, Mônica Maria G. (Trad.). Engenharia de software. 7. ed. Porto Alegre: AMGH, 2011.

SANTOS, Alan Ricardo dos. Um método de aprendizagem baseada em desafios: Um estudo de caso em ambientes de desenvolvimento de aplicativos. 2016. 179. Tese de Doutorado em Ciência da Computação – Pontifícia Universidade Católica do Rio Grande do Sul, Rio Grande do Sul.

SANTOS, Tamyres Oliveira dos. Implantação de sistemas informatizados na saúde : uma revisão integrativa da literatura. 2013. 58 f. Trabalho de Conclusão de Curso (Graduação em Enfermagem). Universidade Federal do Rio Grande do Sul, Porto Alegre, 2013.

ScrumGuides.org. The Scrum Guide. 2017. Disponível em: <http://www.scrumguides.org/scrum-guide.html>.

SOMMERVILLE, Ian. Software Engineering. 9Th ed. Boston: Pearson Education, 2011.

T. Dybå and T. Dingsør. Empirical studies of agile software development: A systematic review. Information and Software Technology, 50(9:10):833 – 859, 2008.

# APÊNDICES

## APÊNDICE A – DOCUMENTO GERADO PELO CBL

Big Ideia: PRONTUÁRIO ELETRÔNICO

Essential Question:

* O que é um prontuário Eletrônico?
* Existe alguma certificação necessária para implantar um prontuário eletrônico?
* Qual instituição cuida da certificação e homologação?
* Como podemos melhorar a eficiência do gerenciamento dos prontuários dos pacientes dentro da Universidade Católica de Brasília?

Challenge: Centralizar os prontuários dos pacientes de fisioterapia em um sistema informatizado que facilite a avaliação e acompanhamento de cada paciente.

Guiding Questions:

1. O controle eletrônico de prontuários já existe na clínica escola de fisioterapia?

Não, porém existe um software proprietário sendo adaptado para as necessidades da clínica, mas os coordenadores têm interesse em contribuir com a nossa solução, e talvez utilizá-la.

1. Onde serão armazenadas as informações de prontuário de uma pessoa?

Em um servidor, na própria clínica de fisioterapia.

1. O paciente possui acesso ao próprio prontuário?

Não, os prontuários ficarão em poder da clínica, somente. Mas o paciente poderá consultar suas informações junto aos responsáveis pelo seu tratamento.

1. Quem atualiza as informações dos alunos e supervisores?

O(s) administrador(es) do sistema, que serão os coordenadores do curso.

1. Quem atualiza as informações das avaliações do paciente?

As informações são atualizadas pelos estudantes (estagiários supervisionados) em cada consulta realizada.

1. De quanto em quanto tempo os dados do paciente são atualizados?

Dados pessoais: somente quando houver alteração.

Dados que compõem o prontuário (fichas de avaliação/acompanhamento): sempre que o paciente retorna à clínica para realizar uma consulta.

1. Atualmente, como as informações são armazenadas?

O armazenamento de informações é feito em prontuários de papel, e eventualmente são preenchidos alguns documentos em formato doc (word).

1. Possui acesso público para consultas?

Não, o paciente pode ter acesso às suas informações somente junto aos responsáveis pelo seu tratamento, na própria clínica.

1. Como evitar redundância nas informações?

O sistema foi pensado para manter as informações de cada paciente em um só lugar, não havendo risco de duplicidade de informações, pois todo o acesso à informação será direcionado ao lugar correto.

1. Quais dados o paciente precisa fornecer para que seja criada a sua ficha inicial?

* Data agendada para a triagem
* Nome completo
* Data de nascimento do paciente
* RG
* Sexo
* Telefone de contato
* Endereço
* Bairro
* UF
* Se é colaborador da Instituição
* Condições de moradia
* Se alugada- Valor
* Possui benefício do Governo
* Profissão
* Renda mensal do paciente
* Grupo Familiar -QTD
* Quantos do grupo familiar trabalham
* Renda familiar
* Possui Convênio Médico
* Realiza tratamento pelo SUS
* Referência para contato
* Telefone da pessoa de referência
* Grau de parentesco
* Total da renda familiar

Dados fornecidos pelo clínico:

* Relato do paciente
* Data do Laudo Médico
* Médico/CRM
* Diagnóstico Clínico
* CID – Classificação Internacional da Doença
* Queixa principal
* Área de atendimento
* Se faz uso de Medicamentos
* Patologias associadas
* Tempo de lesão
* Tratamento anterior
* Marcha independente
* Independência para AVD'S – Atividade de Vida Diária
* Pratica Atividade física
* Realiza atividade Laboral
* Paciente Atleta
* Exames complementares
* Parecer para atendimento fisioterapêutico

1. Qual a metodologia de desenvolvimento utilizada?

Scrum.

1. O atendimento é interrompido durante as férias/recesso?

Sim, os atendimentos são interrompidos durante férias/recesso dos alunos, nesse período é realizada somente a triagem de novos pacientes, para serem atendidos após o recesso letivo.

1. Quais vantagens e inconvenientes entre prontuário informatizado e tradicional (papel)?

|  |  |  |
| --- | --- | --- |
| Prontuários | Vantagens | Inconvenientes |
| Prontuário em Papel | -Maior liberdade na maneira de escrever;  - Facilidade no manuseio;  - Não requer treinamento para o seu manuseio;  - Nunca fica “fora do ar”. | -Ilegibilidade em decorrência dos “hieróglifos” da equipe de saúde;  - Espaço único;  - Ambiguidade;  - Perda frequente de informações;  - Multiplicidade de pastas;  - Dificuldade de acesso e de pesquisa coletiva;  - Falta de padronização;  - Fragilidade do papel. |
| Prontuário Informatizado | - Redução no tempo de atendimento e custos, eliminação da redundância na demanda de exames;  - Desterritorialização;  - Possibilidades de reconstrução histórica e completa dos casos acerca dos pacientes, registros médicos, tratamentos, laudos...;  - Fim do problema de compreensão dos hieróglifos da equipe de saúde;  - Facilidade na organização e no acesso às informações;  - Racionalidade do espaço de arquivamento de grandes quantidades de documentos; | - Manutenção dos prontuários em papel para fins jurídicos, em virtude da indefinição legal dos documentos eletrônicos;  - Necessidade de grande investimento em hardware, software e treinamento;  - Resistência a mudanças;  - Demora na sua implantação;  - Falhas na tecnologia.  - Falhas no sistema de fornecimento de energia elétrica. |

Guiding Activities: Conversar com profissionais que atuam na clínica escola de fisioterapia da UCB, para obter informações relevantes sobre seu funcionamento (fluxo de atendimento, indivíduos envolvidos no processo de atendimento, processos existentes, dados necessários para desenvolvimento da solução, etc).

Solution: Um sistema informatizado que irá realizar o controle dos prontuários e de todos os dados referentes aos indivíduos envolvidos no processo de tratamento dos pacientes, de forma organizada, visando agilizar o manuseio dos documentos da clínica e prover facilidades que os prontuários de papel não oferecem.

## APÊNDICE B – manual de instalação do prontuário ÚNICO UCB

INTRODUÇÃO

Este manual foi feito para a instalação do sistema Prontuário Único UCB, produzido como projeto final do curso de Ciência da Computação. Nessa instalação, será usado um sistema operacional Linux, mais especificamente [Ubuntu 17.10](https://www.ubuntu.com/desktop/1710), podendo ocorrer divergências em outras distribuições e sistemas operacionais.

INSTALAÇÃO DAS DEPENDÊNCIAS DO SISTEMA

Abra um terminal com o comando *Ctrl+Alt+T* e cole os seguintes comandos:

|  |
| --- |
| sudo apt-get install python-pip python-dev python3-dev libpq-dev postgresql postgresql-contrib npm curl git vim pgadmin3 |

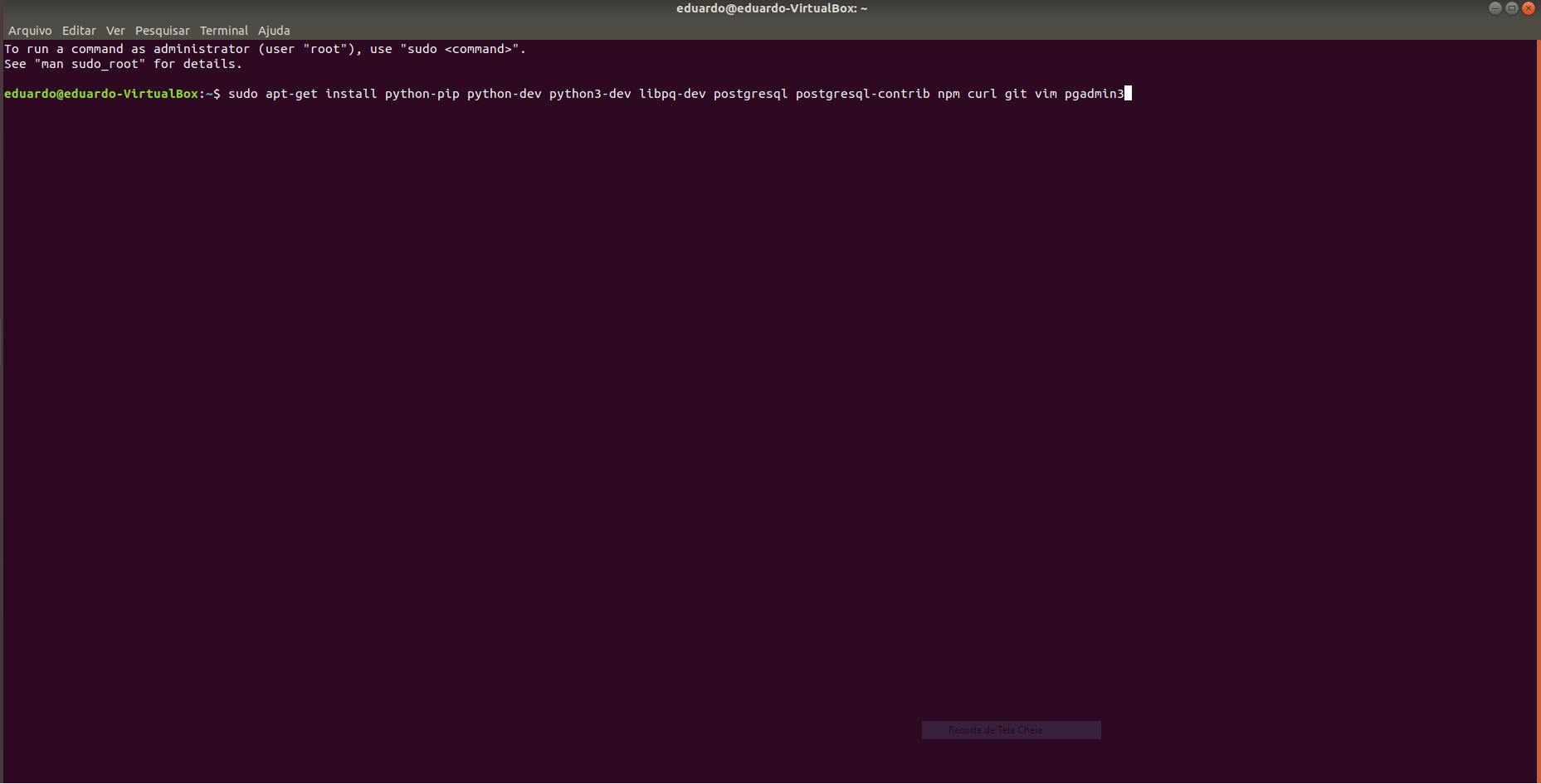


Figura 1

Após concluir a execução desse comando, terá que colar os seguintes comandos e esperar suas execuções:

|  |
| --- |
| sudo curl -sL https://deb.nodesource.com/setup\_7.x | sudo -E bash - |

|  |
| --- |
| sudo apt-get install -y nodejs |

|  |
| --- |
| sudo apt-get install -y build-essential |

|  |
| --- |
| sudo npm install -g bower |

Ao finalizar todas as instalações, as dependências do sistema operacional foram atendidas.

BAIXAR O PROJETO

O projeto encontra-se disponível no GitHub, para poder baixá-lo é necessário rodar o seguinte comando no terminal:

|  |
| --- |
| git clone https://github.com/eduardoedson/TCC.git |

CRIAR BANCO DE DADOS

Primeiro, será preciso entrar no terminal do postgresql usando o comando:

|  |
| --- |
| sudo su - postgres |

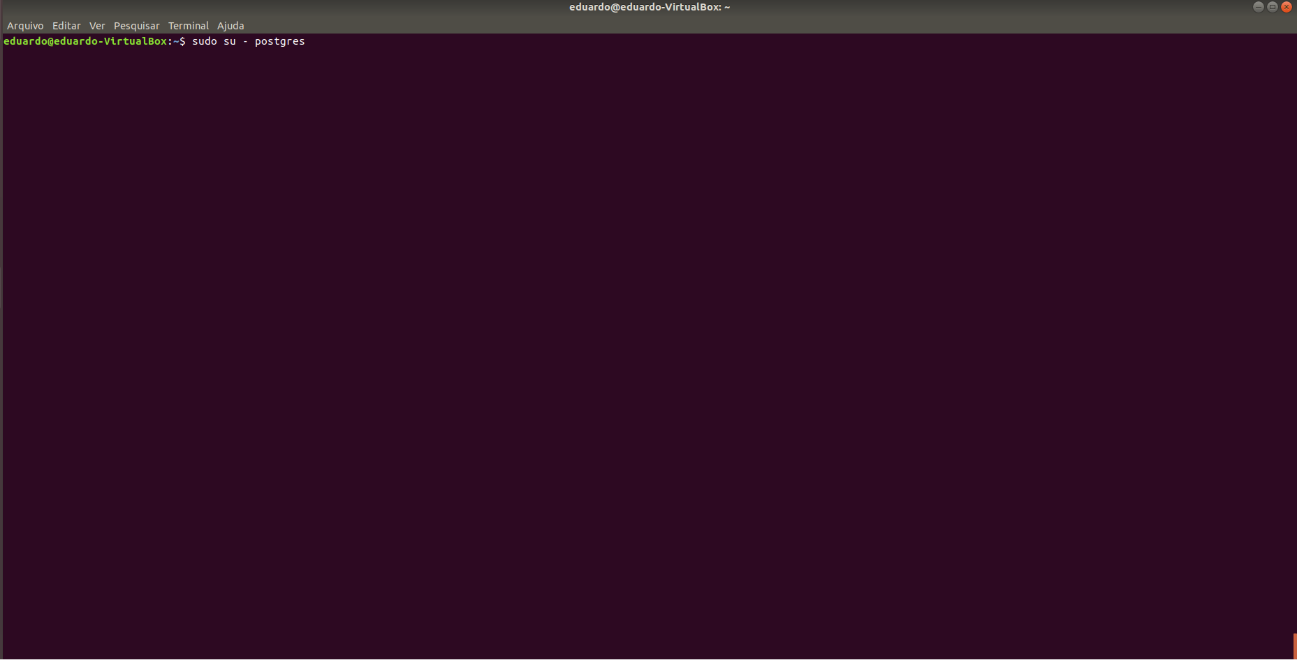


Figura 2

Agora vamos criar o banco com o seguinte comando:

|  |
| --- |
| createdb prontuario |

Após esse comando, é necessário entrar no psql com o seguinte comando e criar o usuário e suas permissões:

|  |
| --- |
| psql |

|  |
| --- |
| CREATE USER root WITH PASSWORD 'root'; |

|  |
| --- |
| GRANT ALL PRIVILEGES ON DATABASE prontuario TO root; |

Para sair do psql e do postgresql use os seguintes comandos respectivamente:

|  |
| --- |
| \q |

# 

|  |
| --- |
| exit |

INSTALAÇÃO DA VIRTUALENV

A virtualenv será usada para instalar as dependências especificas do projeto sem prejudicar as instalações do sistema operacional.

Para instalar, use o seguinte comando no terminal:

|  |
| --- |
| sudo pip install virtualenvwrapper |

Em seguida, deverá configurar o bashrc. Para abri-lo, usaremos o seguinte comando:

|  |
| --- |
| sudo vim ~/.bashrc |

No final do arquivo, será adicionado essas duas novas linhas:

|  |
| --- |
| export WORKON\_HOME=$HOME/.virtualenvs  source /usr/local/bin/virtualenvwrapper.sh |

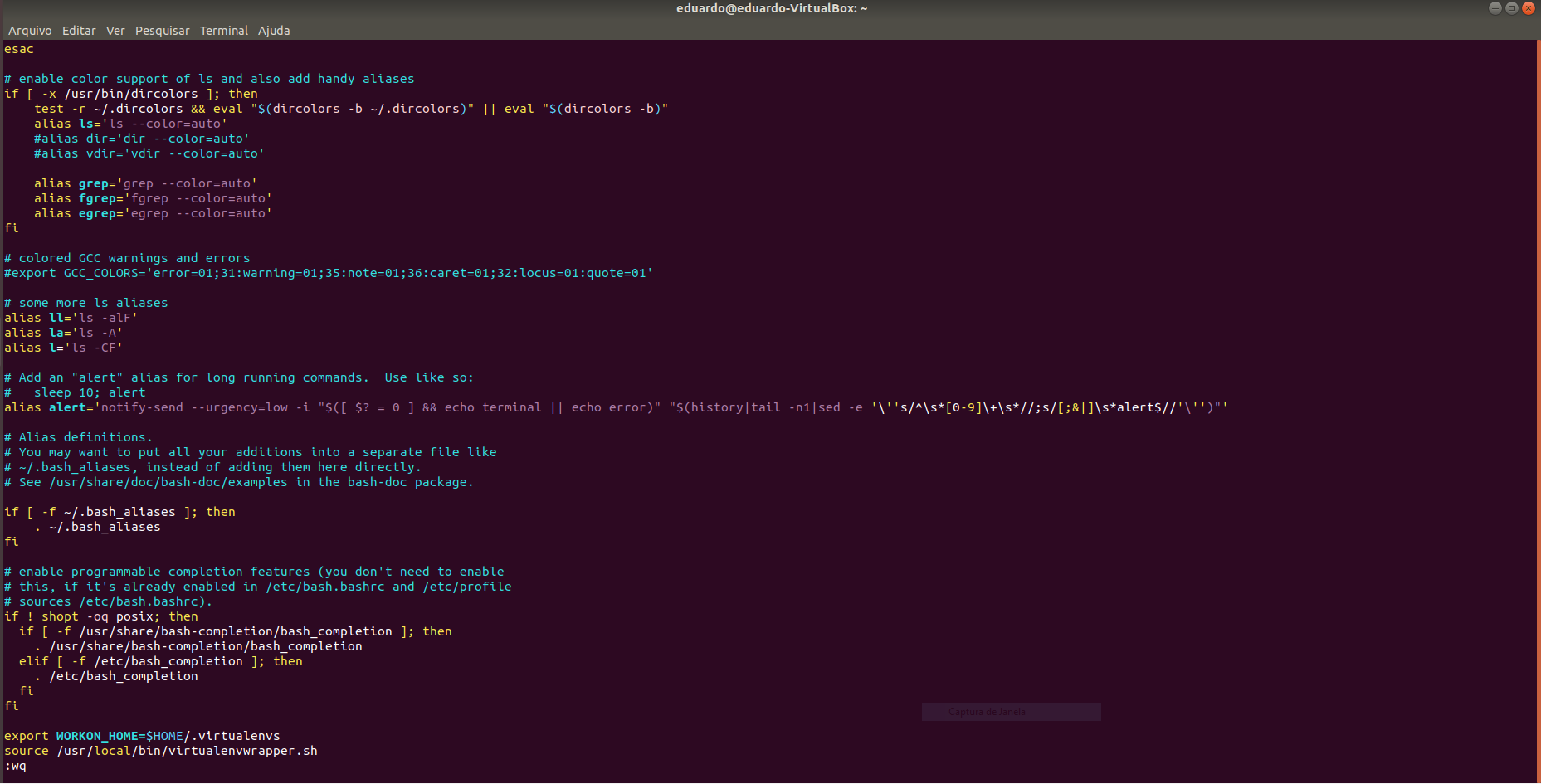


Figura 3

Depois de adicionar as linhas, iremos apertar *ESC* e digitas :wq para fechar e salvar o arquivo.

Nesse momento é preciso recarregar o arquivo com o seguinte comando:

|  |
| --- |
| source ~/.bashrc |

Com o seguinte comando, será criada e executada uma virtualenv configurada para usar python3:

|  |
| --- |
| mkvirtualenv --python=/usr/bin/python3 prontuario |

O (prontuario) *(Figura 4)* escrito no terminal indica que você está dentro da virtualenv, que já foi executada, por ser a primeira vez. Porém nas outras vezes que rodar o sistema, terá que usar o comando *workon prontuário* para ligar a virtualenv.

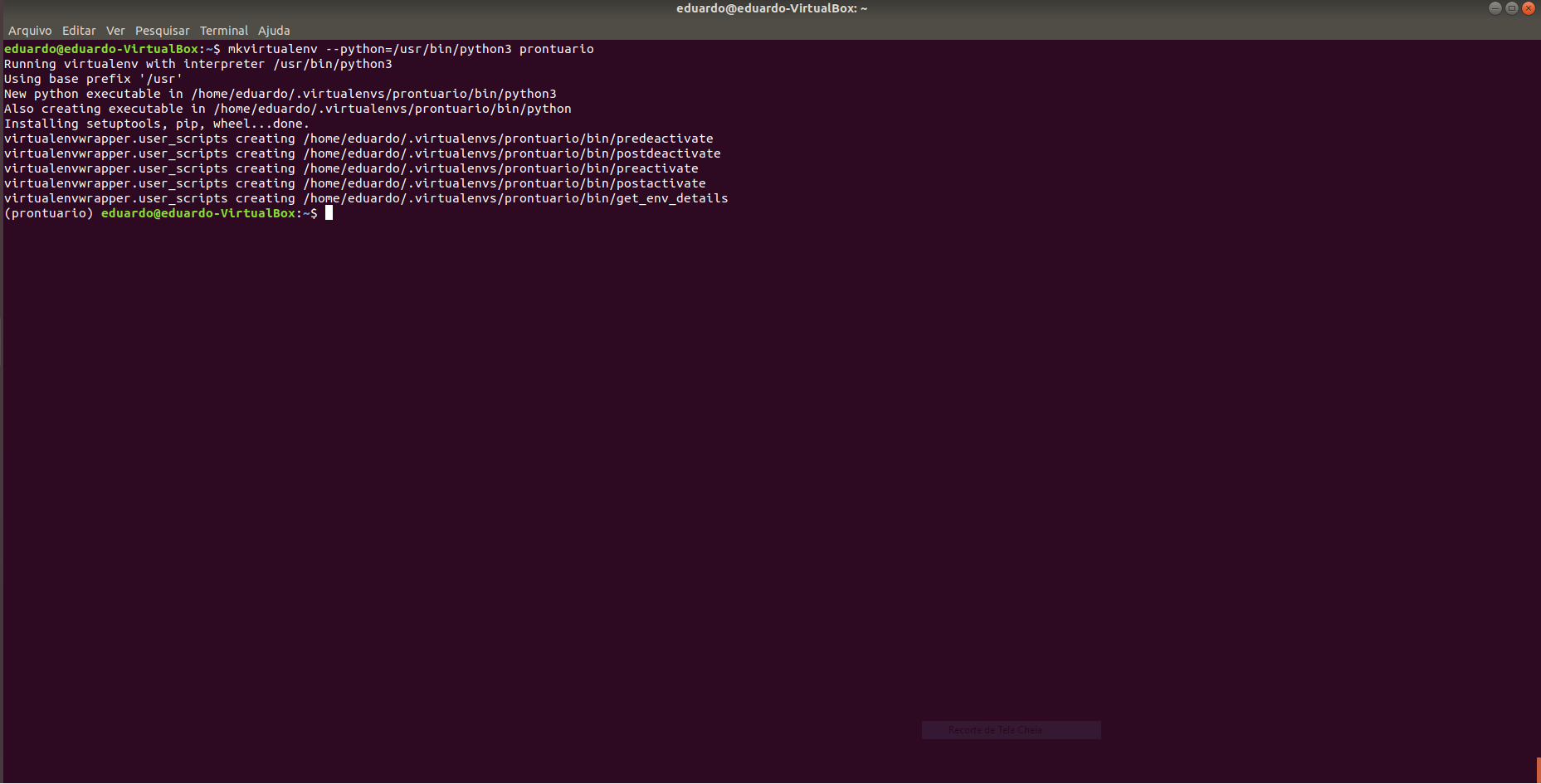


Figura 4

DEPENDÊNCIAS DO PROJETO

A partir deste momento, estando dentro da virtualenv, será preciso entrar pasta do projeto que foi clonado nos passos anteriores:

|  |
| --- |
| cd /TCC/django\_project |

Neste momento será preciso instalar as dependências especificas do projeto com os seguintes comandos:

|  |
| --- |
| pip install -r requeriments.txt |

|  |
| --- |
| ./manage.py bower install |

Caso existam opções na hora de instalação, qualquer uma pode ser selecionada.

Com isso a instalação está completa, faltando somente criar as tabelas do banco de dados com o seguinte comando:

|  |
| --- |
| ./manage.py migrate |

Para rodar o sistema para poder acessá-lo no navegador:

|  |
| --- |
| ./manage.py runserver |

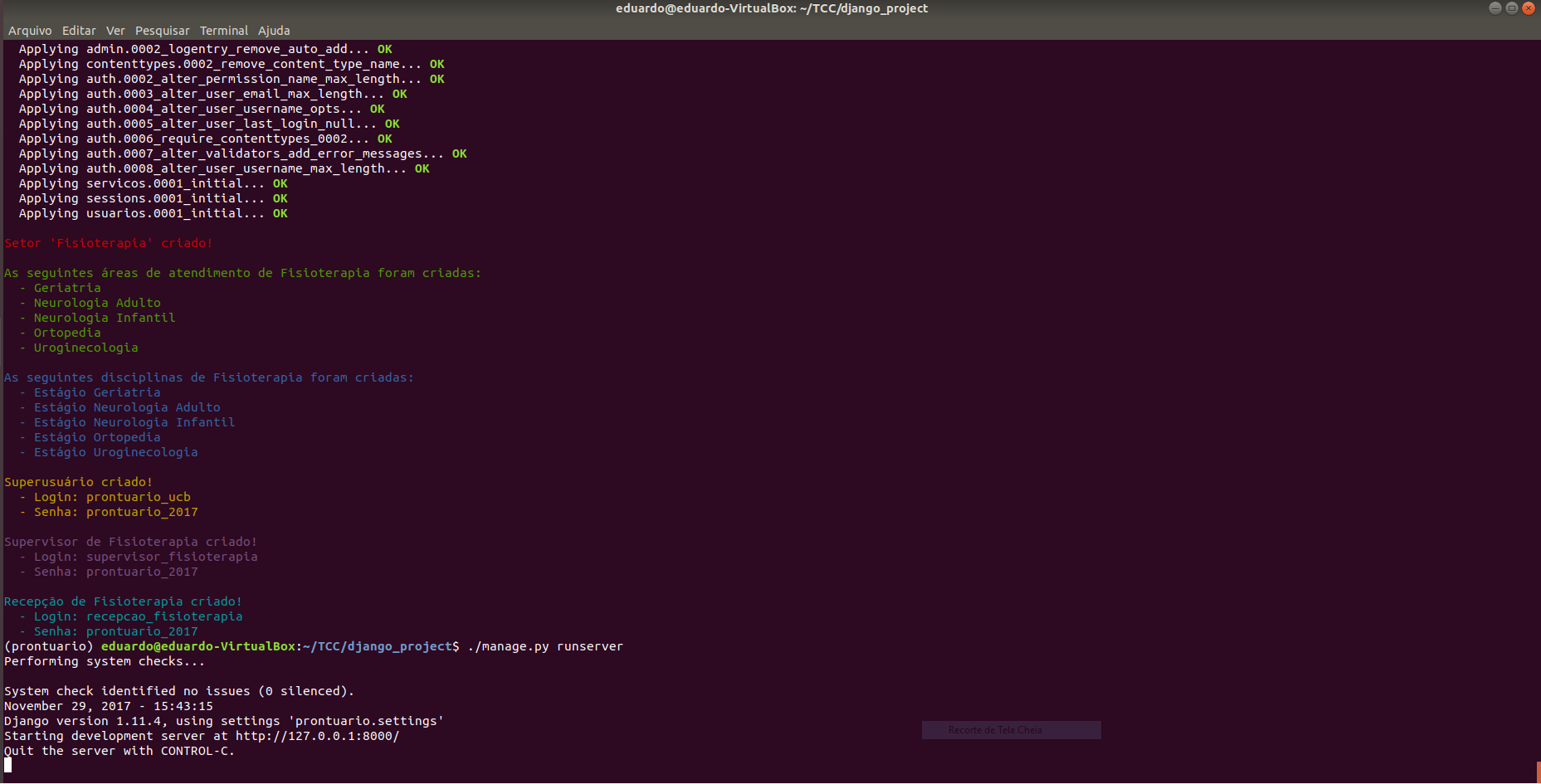


Figura 5

Após todos esses passos, o sistema foi instalado e está rodando.

ACESSANDO O SISTEMA

Para acessar o sistema, basta acessar a url <http://localhost:8000>.



Figura 6

Durante a instalação foram criados usuários padrões para que você possa ter acesso:

|  |  |  |
| --- | --- | --- |
| Usuário | Senha | Tipo |
| prontuário\_ucb | prontuário\_2017 | Superusuário |
| supervisor\_fisioterapia | prontuário\_2017 | Supervisor |
| recepção\_fisioterapia | prontuário\_2017 | Recepcionista |

ACESSANDO O SISTEMA OUTRAS VEZES

Para acessar o sistema outras vezes iremos usar 3 comandos:

|  |
| --- |
| workon prontuario |

|  |
| --- |
| Cd TCC/django\_project |

|  |
| --- |
| ./manage.py runserver |

E, pra finalizar, é só acessar a url <http://localhost:8000> no navegador.