

Engenharia de Software 2

Projeto – Escalonamento de Enfermeiros por Bloco Operatório

Bruno Santos Nº1012388

Leonardo Lourenço Nº1012395

Luciano Brido Nº 1011881

Índice

>Descrição do tema do projeto……………………………………………………….pag.3

>Diagrama de Contexto………………………………………………………………pag.3

Descrição dos três padrões ……………………………………………….……….pág.4,5

Tabela de Atores, objetivos e respetivos Casos de Uso ………………………..…...pág.6

Descrição dos atores………………………………………………………………...pág.7

Diagrama de Casos de Uso………………………………………………………….pág.8

>Descrição dos Casos de Uso…………………….…...pág.9,10,11,12,13,14,15,16

>Diagramas de Sequência dos Casos de Uso……………………………….pág.18,19,20

<< Fazer os restantes >>

>Diagrama de Classes……………………………………………………………….pág.21

>>Semântica de Classes:

>>Dicionário de Dados…………………………………………………..pág.

>>Operações……………………………………………………………..pág.

>>Diagramas de sequencia a que pertence………………………………pág.

>>Diagrama de Atividades ……………………………………………………..pág.

Diagrama de Estados………………………………………………………………pág.

Diagrama de Componentes……………………………………………………….pág.

Diagrama de Instalação……………………………………………………………pág.

>Algoritmo(s) para gerar horários…………………………………………………..pág.17

>Protótipos ………………………………………………………………..pág.22,23,24,25

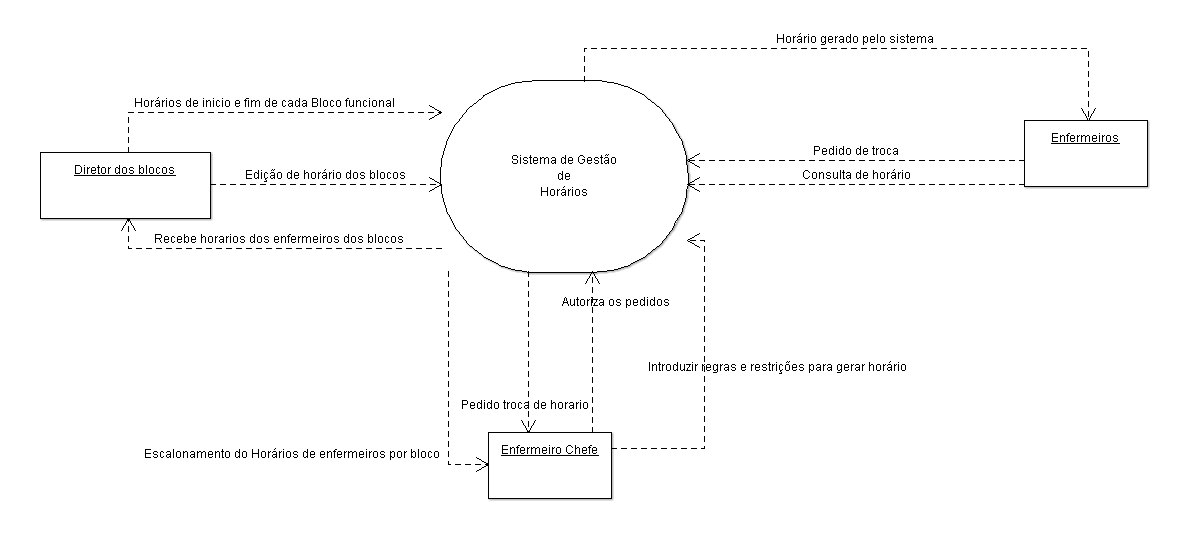
Atividades e tempos gastos em horas por elemento de grupo …………………….pág.26

Descrição do tema do projeto

O objetivo do nosso projeto é criar um software que realize o escalonamento de enfermeiros por bloco operatório.

Os pontos alvo do nosso projeto são a criação dos horários dos enfermeiros segundo o bloco operatório escolhido e a possibilidade dos enfermeiros poderem realizar trocas entre si, isto tudo atendendo sempre à legislação.

Diagrama de Contexto



Descrição dos três padrões

BreathBeforeDepth:

O processo de desenvolvimento de casos de uso é um processo em que se está constantemente a aprender e a reavaliar o nosso modelo.

Este processo é muito parecido á forma como antigamente se faziam as casas, não tinham planos detalhados de como seria a casa, iam construído os quartos segundo as necessidades, tanto que nas casas mais antigas existem quartos que apenas podemos entrar neles através de um outro quarto.

É preciso pensar nos casos de uso mas não ao mais afincado pormenor.

A equipa de desenvolvimento esgotar-se-á facilmente se detalharmos as coisas cedo e de forma complexa, depressa perdem o interesse em faze-lo. Se começarmos com algo muito complexo e necessitarmos de mudar algo mais tarde, teremos de mudar mais coisas nos nossos casos de uso. Só quando a nossa visão geral estiver completa e conhecermos mais o sistema que estamos a trabalhar, devemos a começar a aumentar a precisão do delineamento dos casos de uso.

O BreadthBeforeDepth indica a ordem para o processo de escrita dos casos de uso.

Às vezes, é necessário concluir alguns casos de uso antes dos outros. Nesse caso, precisamos de uma abordagem incremental com base na prioridade do projeto. Identificamos os casos de uso mais importantes, como aqueles que definem o projeto ou aqueles que precisam ser feitos primeiro, com base em problemas comerciais ou técnicos. É necessário a explicitação de cada caso de uso para que se este não for realmente necessário eliminá-lo.

O BreadthBeforeDepth, é isso mesmo, como o nome indica Respirar Antes de Aprofundar, ou seja, pensar primeiro de forma calma no aspeto global dos nossos casos de uso e depois adicionar detalhes progressivamente.

A visão geral permite ver o todo sem ficar preso nos detalhes. Trabalhar em alto nível facilita o esboço do sistema e dos seus limites, além de verificar se os seus casos de uso satisfazem todas as funcionalidades do sistema. Esta visão de alto nível também permite que iniciemos o planeamento do projeto, o estabelecimento das prioridades do nosso trabalho, e o desenvolvimento dos casos de uso mais importantes primeiro e os menos importantes depois.

PreciseAndReadable:

Um caso de uso deve ser compreensível para os stakeholders e para os programadores, isto porque o caso de uso pode ser para vários tipos de nível de habilidades.

Um caso de uso descrito muito tecnicamente e ou muito detalhadamente pode ser útil para os programadores, mas os próprios clientes podem não compreender e podem-no descartar por não perceberem do que se trata.

Portanto um caso de uso também pode ser o oposto ser pouco detalhado mas de fácil compreensão, o que pode tornar o caso difícil de implementar, já que o programador não sabe exatamente o que é pedido.

Por estas duas razões, um caso de uso para ser bem escrito tem de satisfaz todas as audiências sem favorecer nenhum dos grupos).

Quando um caso de uso está incompleto para os programadores, eles têm a tendência de adicionar detalhe e soluções, o que ajuda a complicar o caso de uso, mas ao mesmo tempo isto torna-o difícil de entender para o cliente, criando desacordo por não entenderem a racionalidade da solução e ou detalhe.

A solução apresentada pelo autor é que haja o diálogo entre os envolvidos, para que todos compreendem como funciona, pois o objetivo dos casos de uso é determinar o que o sistema deve fazer para o cliente e que como o sistema deve funcionar para os programadores implementarem.

A conclusão a que chega é que um caso de uso preciso e compreensível é um caso em que há um consenso de todos os envolvidos.

Para a construção do próprio caso de uso, o autor propôs que seja escrito com um único objetivo em mente, outra proposta para o escritor do caso é saber a que vai ler e usar a linguagem do leitor. O caso de uso não deve deixar espaço para interpretação, usar vários exemplos de como o sistema deve funcionar.

LeveledSteps:

Etapas de casos de uso excessivamente grandes ou excessivamente pequenas ocultam o objetivo e dificultam a leitura e compreensão do caso de uso. O padrão LeveledSteps descreve como escrever casos de uso equilibrados e razoavelmente dimensionados.

Qualquer coisa pode ser descrita em etapas pequenas que tornam o caso de uso cada vez menor perdendo o sentido e a intenção de detalhe.

O oposto também poderá acontecer qualquer coisa descrita em etapas excessivamente grandes podem ocultar um comportamento importante, omitindo ações chave que os desenvolvedores devem conhecer.

Misturar níveis de detalhe pode levar a uma distração do que é suposto estar a acontecer, e dificulta a interpretação correta das instruções.

Cada cenário deve ter de três a nove etapas. Idealmente, eles estão todos em níveis semelhantes, um nível de abstração logo abaixo do objetivo do caso de uso

Tabela de Atores, objetivos e respetivos Casos de Uso

|  |  |  |
| --- | --- | --- |
| **Ator** | **Casos de Uso** | **Objetivo** |
| **Diretor**  **Dos**  **Blocos** | Gerar Horário | Realizar a criação do horário do bloco x segundo as regras impostas pelo Enfermeiro Chefe. |
| Visualizar horários | Visualizar horários dos blocos operatórios. |
| Gerir enfermeiros | Criar, editar, visualizar e eliminar dados pessoais dos enfermeiros. |
| Gerir Especialidades | Criar, editar, visualizar e eliminar especialidades. |
| Gerir Turnos | Criar, editar, visualizar e eliminar turnos. |
| Gerir Blocos Operatórios | Criar, editar, visualizar e eliminar turnos. |
| Gerir a ligação entre Blocos Operatórios e Turnos | Criar, editar, visualizar e eliminar a ligação entre blocos operatórios e turnos. |
| **Enfermeiro**  **Chefe** | Introduzir regras e restrições | Para gerar horário corretamente. Ex: enfermeiros escolhidos, etc. |
| Pedir trocas de turnos | Realizar o pedido de trocas de turnos no horário x. |
| Validar trocas | Autorizar ou não os pedidos de trocas. |
| Visualizar horários | Visualizar horários dos blocos operatórios. |
| **Enfermeiros** | Pedir trocas de turnos | Realizar o pedido de trocas de turnos no horário x. |
| Visualizar horário | Visualizar horários dos blocos operatórios. |

Descrição dos atores

No nosso software temos 3 tipos de utilizadores:

Diretor dos Blocos: A quem cabe gerar os horários e certificar que as trocas entre enfermeiros são feitas segundo a legislação, nível da hierarquia alto.

Enfermeiro Chefe: A quem cabe introduzir as regras e restrições e validar as trocas, nível de hierarquia médio.

Enfermeiro: Pode visualizar o seu horário e pedir trocas de turnos, nível de hierarquia baixo.

Diagrama de Casos de Uso

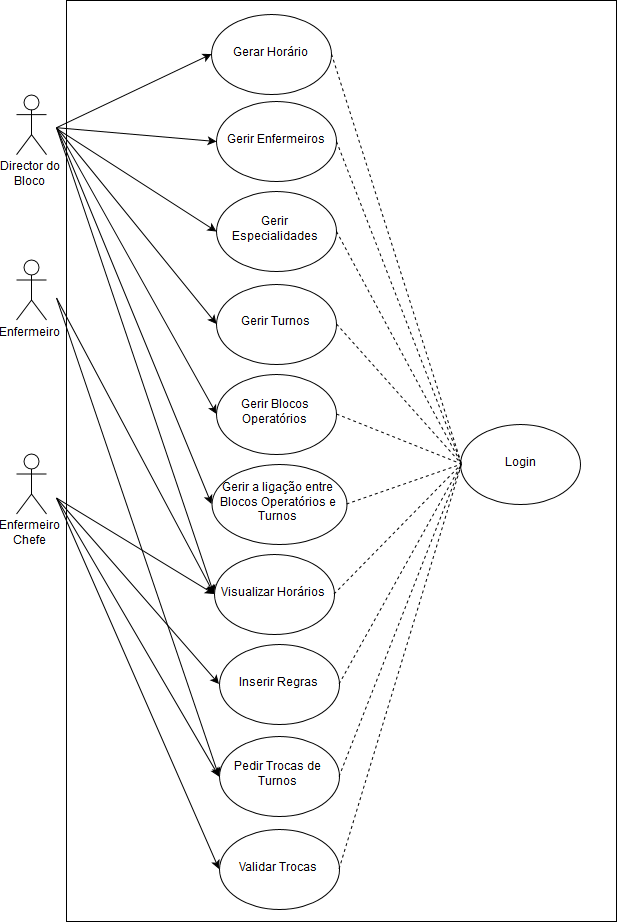
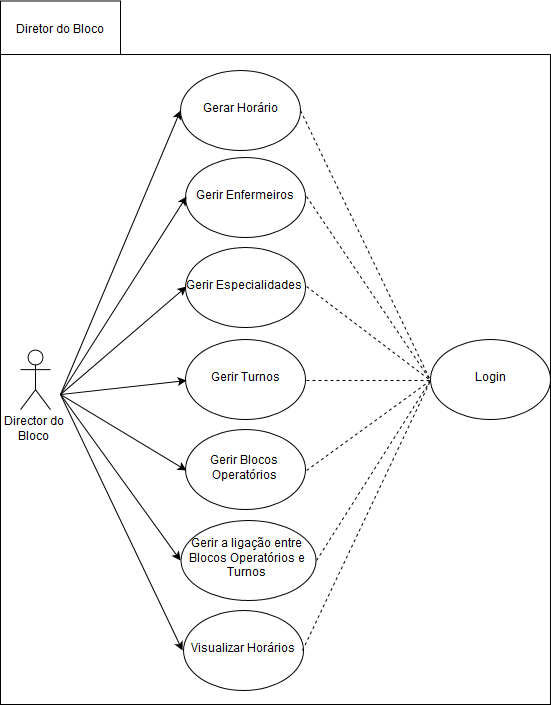
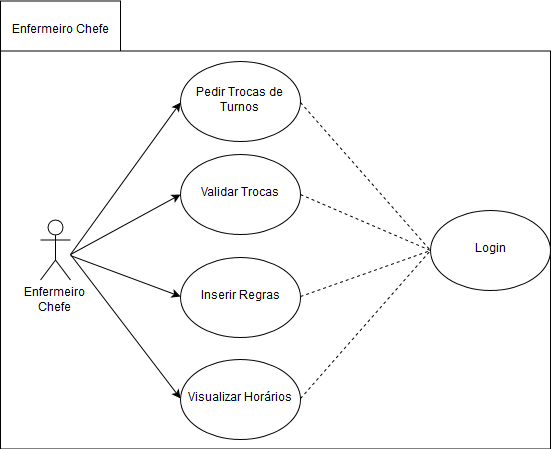
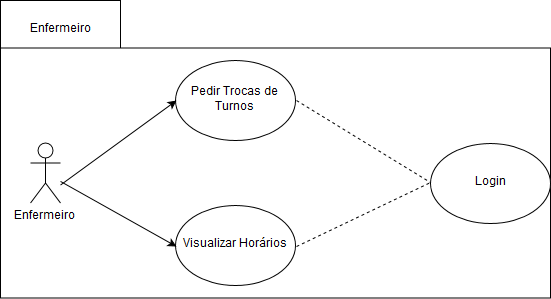


Diagrama de Pacotes de Casos de Uso







Descrição de todos os Casos de Uso

|  |  |
| --- | --- |
| Nome do caso de uso | RF01, Autenticar Ator (Login) |
| Descrição do requisito | O objetivo deste caso de uso é controlar o acesso ao sistema de software |
| Tamanho | 21 |
| Pré-condição |  |
| Cenário Principal |  |
|  | 1.Acede ao software  2.Pede a sua identificação (login e password)  3.Introduz login e password  4.Apresenta o ambiente de trabalho |
| Cenário Secundário | 4a. Se o email de utilizador estiver errado o sistema volta a pedir identificação.  4b. Se a password estiver incorreta o sistema volta a pedir a sua password. |
| Requisitos não funcionais |  |

|  |  |
| --- | --- |
| Nome do caso de uso | RF02, Criar Enfermeiro |
| Descrição do requisito | O objetivo deste caso de uso é criar novo enfermeiro |
| Tamanho | 8 |
| Pré-condição | O diretor tem o login válido. |
| Cenário Principal |  |
|  | 1-O diretor seleciona a opção criar novo enfermeiro  2-O sistema abre o formulário com os dados a introduzir  3-O diretor insere os atributos (inserir nome, email, especialidade, tipo de enfermeiro, número de telefone, numero de identificação de identidade, data de nascimento e data do filho mais novo).  4-O diretor carrega na opção criar.  5-O sistema atribui automaticamente o ID ao novo enfermeiro.  6-O sistema dá sucesso. |
| Cenário Alternativo | 2a-O sistema tem opção de voltar à lista  3a-Se os atributos (nome, email, especialidade, tipo de enfermeiro, número de telefone, numero de identificação de identidade, data de nascimento e data do filho mais novo), não forem bem preenchidos o sistema dá erro e pede para inserir os dados. (Colocar aqui que verificações fazemos a cada campo, no editar é igual) |
| Requisitos não funcionais | * Testar se o sistema deixa registar o enfermeiro sem os campos obrigatórios estarem preenchidos. * Testar campo IDCard através do algoritmo Dígito de controlo. * Teste1-Validar email   Utilizador insere no Email = skgeldgjeç  O sistema notifica – O Email tem de ter um domínio e um subdomínio válido (exemplo [nomedoutilizador@nomedosubdominio.nomedodominio](mailto:nomedoutilizador@nomedosubdominio.nomedodominio));   * Teste2– Validação do Contacto   Utilizador insere – Contacto = 8154866848egege84e  O sistema notifica – O contacto apenas pode conter caracteres numéricos.   * Teste3– Validação do Nome   Utilizador insere – Name = Alfredo012  O sistema notifica – O Nome apenas pode conter caracteres. |

|  |  |
| --- | --- |
| Nome do caso de uso | RF03, Editar perfil dos Enfermeiros |
| Descrição do requisito | O objetivo deste caso de uso é editar perfil do enfermeiro |
| Tamanho | 8 |
| Pré-condição | O diretor tem de ter o login válido. |
| Cenário Principal |  |
|  | 1-O diretor seleciona a opção editar enfermeiros no enfermeiro que pretende editar.  2-O sistema abre o formulário com os dados já preenchidos.  4-O diretor altera os atributos (inserir nome, email, especialidade, tipo de enfermeiro, número de telefone, numero de identificação de identidade, data de nascimento e data do filho mais novo) desejados.  5-O ator carrega na opção guardar.  6-O sistema devolve uma mensagem de sucesso. |
| Cenário Alternativo | 1A-O sistema tem opção de voltar á lista  4a-Se os atributos (nome, email, especialidade, tipo de enfermeiro, número de telefone, numero de identificação de identidade, data de nascimento e data do filho mais novo), não forem bem preenchidos o sistema dá erro e pede para inserir os dados. |
| Requisitos não funcionais | * Testar se o sistema deixa registar o enfermeiro sem os campos obrigatórios estarem preenchidos. * Testar campo IDCard através do algoritmo Dígito de controlo. * Teste1-Validar email   Utilizador insere no Email = skgeldgjeç  O sistema notifica – O Email tem de ter um domínio e um subdomínio válido (exemplo [nomedoutilizador@nomedosubdominio.nomedodominio](mailto:nomedoutilizador@nomedosubdominio.nomedodominio));   * Teste2– Validação do Contacto   Utilizador insere – Contacto = 8154866848egege84e  O sistema notifica – O contacto apenas pode conter caracteres numéricos.   * Teste3– Validação do Nome   Utilizador insere – Name = Alfredo012  O sistema notifica – O Nome apenas pode conter caracteres. |

|  |  |
| --- | --- |
| Nome do caso de uso | RF04, Eliminar Enfermeiro |
| Descrição do requisito | O objetivo deste caso de uso é eliminar enfermeiro. |
| Tamanho | 15 |
|  |  |
| Pré-condição | O diretor tem de ter o login válido.  Tem que existir uma lista de enfermeiros no sistema. |
| Cenário Principal |  |
|  | 1-O diretor seleciona eliminar no enfermeiro que pretende apagar.  2-O sistema abre uma janela com os detalhes do enfermeiro.  3-O diretor seleciona o botão delete para eliminar o enfermeiro.  4- O sistema abre uma janela para confirmar que pretende eliminar o enfermeiro.  5-O sistema devolve uma mensagem de sucesso. |
| Cenário Alternativo | 2a-O sistema tem um botão de voltar à lista.  3a- O diretor do bloco pode cancelar  3b- Se o enfermeiro tiver um horário não pode ser eliminado, o sistema devolve uma mensagem de erro. |
| Requisitos não funcionais |  |

|  |  |
| --- | --- |
| Nome do caso de uso | RF05, Criar Especialidade |
| Descrição do requisito | O objetivo deste caso de uso é criar uma nova especialidade |
| Tamanho | 8 |
| Pré-condição | O diretor tem o login válido. |
| Cenário Principal |  |
|  | 1-O diretor seleciona a opção criar nova especialidade  2-O sistema abre o formulário com os dados a introduzir  3-O diretor insere o atributo nome.  4-O diretor carrega na opção criar.  5-O sistema atribui automaticamente o ID à nova especialidade e devolve uma mensagem de sucesso. |
| Cenário Alternativo | 2a-O sistema tem opção de voltar à lista  3a-Se o atributo nome estiver nulo o sistema pede para inserir um nome. |
| Requisitos não funcionais | * Teste1– Validação do Nome   Utilizador insere – Name = Alfredo012  O sistema notifica – O Nome apenas pode conter caracteres. |

|  |  |
| --- | --- |
| Nome do caso de uso | RF06, Editar Especialidade |
| Descrição do requisito | O objetivo deste caso de uso é editar uma especialidade |
| Tamanho | 8 |
| Pré-condição | O diretor tem o login válido. |
| Cenário Principal |  |
|  | 1-O diretor seleciona a opção editar na especialidade que pretende editar.  2-O sistema abre o formulário com os dados já preenchidos.  3-O diretor altera os atributos que pretende alterar.  4-O diretor carrega na opção guardar.  5-O sistema devolve uma mensagem de sucesso. |
| Cenário Alternativo | 2a-O sistema tem opção de voltar à lista  3a-Se o atributo nome estiver nulo o sistema pede para inserir um nome. |
| Requisitos não funcionais | * Teste1– Validação do Nome   Utilizador insere – Name = Alfredo012   * O sistema notifica – O Nome apenas pode conter caracteres. |

|  |  |
| --- | --- |
| Nome do caso de uso | RF07, Eliminar Especialidade |
| Descrição do requisito | O objetivo deste caso de uso é eliminar uma especialidade |
| Tamanho | 8 |
| Pré-condição | O diretor tem o login válido. |
| Cenário Principal |  |
|  | 1-O diretor seleciona eliminar na especialidade que pretende apagar.  2-O sistema abre uma janela com os detalhes da especialidade.  3-O diretor seleciona o botão delete para eliminar a especialidade.  4- O sistema abre uma janela para confirmar que pretende eliminar a especialidade.  5-O sistema devolve uma mensagem de sucesso. |
| Cenário Alternativo | 2a-O sistema tem um botão de voltar à lista.  3a- O diretor do bloco pode cancelar  3b- Se a especialidade estiver atribuída a pelo menos um enfermeiro não pode ser eliminada, o sistema devolve uma mensagem de erro. |
| Requisitos não funcionais |  |

|  |  |
| --- | --- |
| Nome do caso de uso | RF08, Criar Turno |
| Descrição do requisito | O objetivo deste caso de uso é criar um novo turno |
| Tamanho | 8 |
| Pré-condição | O diretor tem o login válido. |
| Cenário Principal |  |
|  | 1-O diretor seleciona a opção criar novo turno.  2-O sistema abre o formulário com os dados a introduzir  3-O diretor insere o atributo nome, hora de início e duração.  4-O diretor carrega na opção criar.  5-O sistema atribui automaticamente o ID ao novo turno e devolve uma mensagem de sucesso. |
| Cenário Alternativo | 2a-O sistema tem opção de voltar à lista  3a-Se o atributo nome estiver nulo o sistema pede para inserir um nome.  3b- Se o atributo hora de início estiver nulo o sistema pede para inserir uma hora de início.  3c- Se o atributo hora de início tiver sido introduzido num formato incorreto o sistema pede para inserir no formato 00:00.  3d- Se o atributo duração estiver nulo o sistema pede para inserir uma duração.  3e- Se o atributo duração tiver sido introduzido num formato incorreto o sistema pede para inserir no formato 00:00. |
| Requisitos não funcionais | * Testes |

|  |  |
| --- | --- |
| Nome do caso de uso | RF09, Editar Turno |
| Descrição do requisito | O objetivo deste caso de uso é editar um turno |
| Tamanho | 8 |
| Pré-condição | O diretor tem o login válido. |
| Cenário Principal |  |
|  | 1-O diretor seleciona a opção editar no turno que pretende editar.  2-O sistema abre o formulário com os dados já preenchidos.  3-O diretor altera os atributos que pretende alterar.  4-O diretor carrega na opção guardar.  5-O sistema devolve uma mensagem de sucesso. |
| Cenário Alternativo | 2a-O sistema tem opção de voltar à lista  3a-Se o atributo nome estiver nulo o sistema pede para inserir um nome.  3b- Se o atributo hora de início estiver nulo o sistema pede para inserir uma hora de início.  3c- Se o atributo hora de início tiver sido introduzido num formato incorreto o sistema pede para inserir no formato 00:00.  3d- Se o atributo duração estiver nulo o sistema pede para inserir uma duração.  3e- Se o atributo duração tiver sido introduzido num formato incorreto o sistema pede para inserir no formato 00:00. |
| Requisitos não funcionais |  |

|  |  |
| --- | --- |
| Nome do caso de uso | RF10, Eliminar Turno |
| Descrição do requisito | O objetivo deste caso de uso é eliminar um turno |
| Tamanho | 8 |
| Pré-condição | O diretor tem o login válido. |
| Cenário Principal |  |
|  | 1-O diretor seleciona eliminar no turno que pretende apagar.  2-O sistema abre uma janela com os detalhes do turno.  3-O diretor seleciona o botão delete para eliminar o turno.  4- O sistema abre uma janela para confirmar que pretende eliminar o turno.  5-O sistema devolve uma mensagem de sucesso. |
| Cenário Alternativo | 2a-O sistema tem um botão de voltar à lista.  3a- O diretor do bloco pode cancelar  3b- Se o turno estiver atribuído a pelo menos uma ligação entre um bloco operatório e um turno, o turno não pode ser eliminado, o sistema devolve uma mensagem de erro. |
| Requisitos não funcionais |  |

|  |  |
| --- | --- |
| Nome do caso de uso | RF11, Criar Bloco Operatório |
| Descrição do requisito | O objetivo deste caso de uso é criar um novo bloco operatório |
| Tamanho | 8 |
| Pré-condição | O diretor tem o login válido. |
| Cenário Principal |  |
|  | 1-O diretor seleciona a opção criar um novo bloco operatório.  2-O sistema abre o formulário com os dados a introduzir  3-O diretor insere o atributo nome.  4-O diretor carrega na opção criar.  5-O sistema atribui automaticamente o ID ao novo bloco operatório e devolve uma mensagem de sucesso. |
| Cenário Alternativo | 2a-O sistema tem opção de voltar à lista  3a-Se o atributo nome estiver nulo o sistema pede para inserir um nome. |
| Requisitos não funcionais |  |

|  |  |
| --- | --- |
| Nome do caso de uso | RF12, Editar Bloco Operatório |
| Descrição do requisito | O objetivo deste caso de uso é editar um bloco operatório |
| Tamanho | 8 |
| Pré-condição | O diretor tem o login válido. |
| Cenário Principal |  |
|  | 1-O diretor seleciona a opção editar no bloco operatório que pretende editar.  2-O sistema abre o formulário com os dados já preenchidos.  3-O diretor altera os atributos que pretende alterar.  4-O diretor carrega na opção guardar.  5-O sistema devolve uma mensagem de sucesso. |
| Cenário Alternativo | 2a-O sistema tem opção de voltar à lista  3a-Se o atributo nome estiver nulo o sistema pede para inserir um nome. |
| Requisitos não funcionais | * Testes |

|  |  |
| --- | --- |
| Nome do caso de uso | RF13, Eliminar Bloco Operatório |
| Descrição do requisito | O objetivo deste caso de uso é eliminar um bloco operatório |
| Tamanho | 8 |
| Pré-condição | O diretor tem o login válido. |
| Cenário Principal |  |
|  | 1-O diretor seleciona eliminar no bloco operatório que pretende apagar.  2-O sistema abre uma janela com os detalhes do bloco operatório.  3-O diretor seleciona o botão delete para eliminar o bloco operatório.  4- O sistema abre uma janela para confirmar que pretende eliminar o bloco operatório.  5-O sistema devolve uma mensagem de sucesso. |
| Cenário Alternativo | 2a-O sistema tem um botão de voltar à lista.  3a- O diretor do bloco pode cancelar  3b- Se o bloco operatório estiver atribuído a pelo menos uma ligação entre um bloco operatório e um turno, o bloco operatório não pode ser eliminado, o sistema devolve uma mensagem de erro. |
| Requisitos não funcionais |  |

|  |  |
| --- | --- |
| Nome do caso de uso | RF14, Criar ligação entre Bloco Operatório e Turno |
| Descrição do requisito | O objetivo deste caso de uso é criar uma nova ligação entre bloco operatório e turno |
| Tamanho | 8 |
| Pré-condição | O diretor tem o login válido. |
| Cenário Principal |  |
|  | 1-O diretor seleciona a opção criar uma nova ligação entre bloco operatório e turno.  2-O sistema abre o formulário com todos os blocos operatórios e turnos.  3-O diretor escolhe um bloco operatório e um turno.  4-O diretor carrega na opção criar.  5-O sistema atribui automaticamente o ID à nova ligação entre bloco operatório e turno e devolve uma mensagem de sucesso. |
| Cenário Alternativo | 2a-O sistema tem opção de voltar à lista  3a-Se não existir nenhum bloco operatório para fazer ligação o sistema dá erro.  3b-Se não existir nenhum turno para fazer ligação o sistema dá erro. |
| Requisitos não funcionais | * Testes |

|  |  |
| --- | --- |
| Nome do caso de uso | RF15, Editar ligação entre Bloco Operatório e Turno |
| Descrição do requisito | O objetivo deste caso de uso é editar uma ligação entre bloco operatório e turno |
| Tamanho | 8 |
| Pré-condição | O diretor tem o login válido. |
| Cenário Principal |  |
|  | 1-O diretor seleciona a opção editar na ligação entre bloco operatório e turno que pretende editar.  2-O sistema abre o formulário com o bloco operatório e o turno já selecionados.  3-O diretor escolhe o bloco operatório e/ou o turno que pretende alterar.  4-O diretor carrega na opção guardar.  5-O sistema devolve uma mensagem de sucesso. |
| Cenário Alternativo | 2a-O sistema tem opção de voltar à lista |
| Requisitos não funcionais | * Testes |

|  |  |
| --- | --- |
| Nome do caso de uso | RF16, Eliminar ligação entre Bloco Operatório e Turno |
| Descrição do requisito | O objetivo deste caso de uso é eliminar uma ligação entre bloco operatório e turno |
| Tamanho | 8 |
| Pré-condição | O diretor tem o login válido. |
| Cenário Principal |  |
|  | 1-O diretor seleciona eliminar na ligação entre bloco operatório e turno que pretende apagar.  2-O sistema abre uma janela com os detalhes da ligação entre bloco operatório e turno  3-O diretor seleciona o botão delete para eliminar a ligação entre bloco operatório e turno.  4- O sistema abre uma janela para confirmar que pretende eliminar a ligação entre bloco operatório e turno.  5-O sistema devolve uma mensagem de sucesso. |
| Cenário Alternativo | 2a-O sistema tem um botão de voltar à lista.  3a- O diretor do bloco pode cancelar  3b- Se a ligação entre bloco operatório e turno estiver atribuído a pelo menos um horário não pode ser eliminado, o sistema devolve uma mensagem de erro. |
| Requisitos não funcionais | * Testes |

|  |  |
| --- | --- |
| Nome do caso de uso | RF17, Gerar Horário |
| Descrição do requisito | O objetivo deste caso de uso é gerar um novo horário |
| Tamanho | 8 |
| Pré-condição | O diretor tem o login válido.  Tem de existir enfermeiros e pelo menos uma ligação entre bloco operatório e turno |
| Cenário Principal |  |
|  | 1-O diretor seleciona a opção criar horários  2-O sistema abre a janela de criação de horários  5-O diretor insere o atributo data e seleciona o nome do enfermeiro.  6-O sistema atribui automaticamente o ID ao horário  7-O diretor carrega na opção criar.  8-O sistema devolve uma mensagem de sucesso. |
| Cenário Alternativo | 3a-O sistema tem um botão de voltar à lista.  4a-Se o atributo não estiver preenchido, o sistema dá erro. |
| Requisitos não funcionais |  |

|  |  |
| --- | --- |
| Nome do caso de uso | RF18, Visualizar Horários |
| Descrição do requisito | O objetivo deste caso de uso é visualizar horários dos enfermeiros por bloco operatório |
| Tamanho | 21 |
| Pré-condição | O ator tem o login válido. |
| Cenário Principal |  |
|  | 1-O ator seleciona a opção horários  2-O sistema apresenta uma lista de horários |
| Cenário Alternativo |  |
| Requisitos não funcionais |  |

|  |  |
| --- | --- |
| Nome do caso de uso | RF19, Criar Pedido de Troca |
| Descrição do requisito | O objetivo deste caso de uso é criar troca entre enfermeiros |
| Tamanho | 17 |
| Pré-condição | O enfermeiro ou enfermeiro chefe tem o login válido. |
| Cenário Principal |  |
|  | 1-O enfermeiro seleciona a opção trocas  2-O sistema abre a janela das trocas  3-O enfermeiro seleciona a opção criar troca  4-O sistema abre a janela de criação de trocas  5-O enfermeiro escolhe a dia A e o dia B e o turno A e o turno B para a troca de turno. (A Horário atual, B Horário com o qual quer trocar)  6-O sistema atribui automaticamente o ID à troca.  7-O enfermeiro carrega na opção guardar.  8-O sistema devolve uma mensagem de sucesso. |
| Cenário Alternativo | 4a- O sistema tem a opção de voltar ao menu anterior  5a -Se os atributos (Data A, Data B, turno A, turno B, Horário A, Horário B) não estiverem bem preenchidos o sistema dá erro.  7a – O sistema tem uma opção de cancelar troca |
| Requisitos não funcionais |  |

|  |  |
| --- | --- |
| Nome do caso de uso | RF20, Visualizar Trocas |
| Descrição do requisito | O objetivo deste caso de uso é, visualizar todas as trocas dos enfermeiros. |
| Tamanho | 5 |
|  |  |
| Pré-condição | O ator tem o login válido.  Tem de existir trocas validadas no sistema. |
| Cenário Principal |  |
|  | 1-O ator seleciona a opção trocas  2-O sistema apresenta uma lista de trocas validadas |
| Cenário Alternativo | 1a-O sistema devolve uma mensagem de confirmação. |
| Requisitos não funcionais |  |

|  |  |
| --- | --- |
| Nome do caso de uso | RF21, Validar Pedidos de Trocas |
| Descrição do requisito | O objetivo deste caso de uso é validar todas as trocas dos enfermeiros. |
| Tamanho | 21 |
|  |  |
| Pré-condição | O enfermeiro chefe tem o login válido. |
| Cenário Principal |  |
|  | 1-O enfermeiro chefe seleciona a janela com a lista de trocas  2-O sistema apresenta uma lista de pedidos de trocas  3-O enfermeiro chefe seleciona as trocas que pretende aprovar ou não aprovar  4-O sistema dá uma mensagem da troca validada. |
| Cenário Alternativo | 3a- O sistema dá uma mensagem de confirmação. |
| Requisitos não funcionais |  |

|  |  |
| --- | --- |
| Nome do caso de uso | RF22, Criar Regras e Restrições |
| Descrição do requisito | O objetivo deste caso de uso é criar novas regras e restrições para o horário. |
| Tamanho | 21 |
| Pré-condição | O diretor tem o login válido. |
| Cenário Principal |  |
|  | 1-O diretor seleciona a opção criar novas regras  2-O sistema abre o formulário com os dados a introduzir  3-O diretor insere os atributos (inserir idade do enfermeiro, idade do filho mais novo, horas semanais, tempo entre turnos).  4--O sistema dá sucesso. |
| Cenário Alternativo | 2a-O sistema tem opção de voltar à lista  3a-Se os atributos (nome, email, especialidade, tipo de enfermeiro, número de telefone, numero de identificação de identidade, data de nascimento e data do filho mais novo), não forem bem preenchidos o sistema dá erro e pede para inserir os dados. |
| Requisitos não funcionais | * Casos de teste |

|  |  |
| --- | --- |
| Nome do caso de uso | RF23, Editar Regras e Restrições |
| Descrição do requisito | O objetivo deste caso de uso é editar regras e restrições |
| Tamanho | 8 |
| Pré-condição | O enfermeiro chefe tem de ter o login válido. |
| Cenário Principal |  |
|  | 1-O enfermeiro chefe seleciona a opção editar regras.  2-O enfermeiro chefe seleciona a regra desejada para editar.  3-O sistema abre a janela com a informação da regra que deseja editar.  4-O enfermeiro chefe altera os atributos (inserir nome, email, especialidade, tipo de enfermeiro, número de telefone, numero de identificação de identidade, data de nascimento e data do filho mais novo) desejados.  5-O enfermeiro chefe carrega na opção guardar.  6-O sistema devolve uma mensagem de sucesso. |
| Cenário Alternativo | 1A-O sistema tem opção de voltar á lista  4a-Se os atributos (inserir nome, email, especialidade, tipo de enfermeiro, número de telefone, numero de identificação de identidade, data de nascimento e data do filho mais novo), não forem bem preenchidos o sistema dá erro e pede para inserir os dados. |
| Requisitos não funcionais |  |

|  |  |
| --- | --- |
| Nome do caso de uso | RF24, Eliminar Regras e Restrições |
| Descrição do requisito | O objetivo deste caso de uso é eliminar regras e restrições. |
| Tamanho | 15 |
|  |  |
| Pré-condição | O diretor tem de ter o login válido.  Tem que existir regras e restrições no sistema. |
| Cenário Principal |  |
|  | 1-O diretor seleciona eliminar na regra que pretende apagar.  2-O sistema abre uma janela com os detalhes da regra que selecionou.  3-O diretor seleciona o botão delete para eliminar a regra.  4-O sistema devolve uma mensagem de sucesso. |
| Cenário Alternativo | 2a-O sistema tem um botão de voltar à lista.  3a-O sistema devolve uma mensagem de confirmação. |
| Requisitos não funcionais |  |

5 Diagramas de Sequência dos Casos de Uso mais importantes

Diagrama de Sequência Login

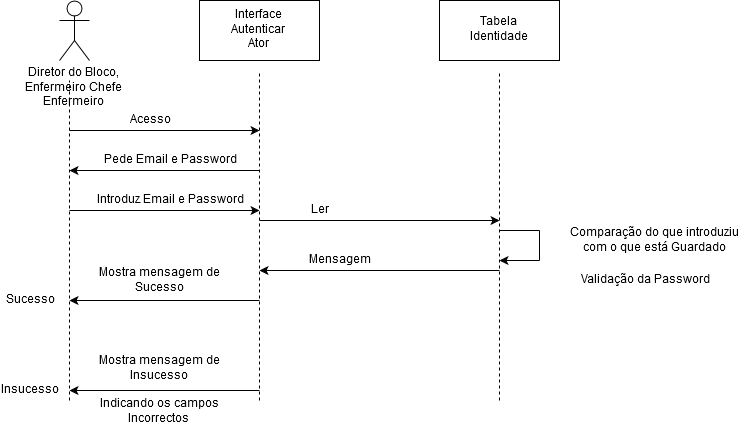


Diagrama de Sequência Introduzir Regras

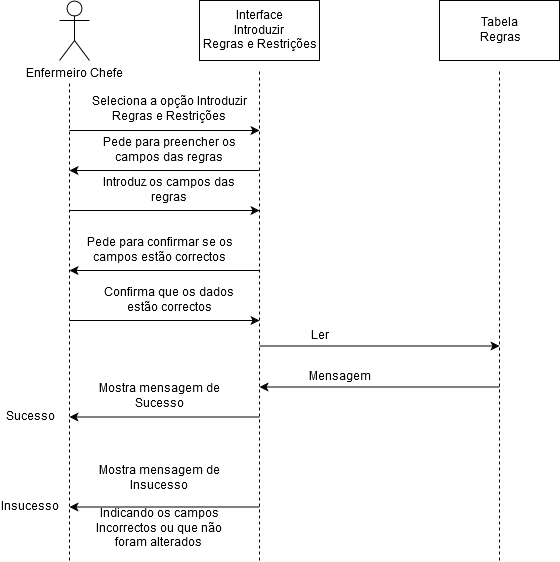


Diagrama de Sequência Visualizar Horário (updated)

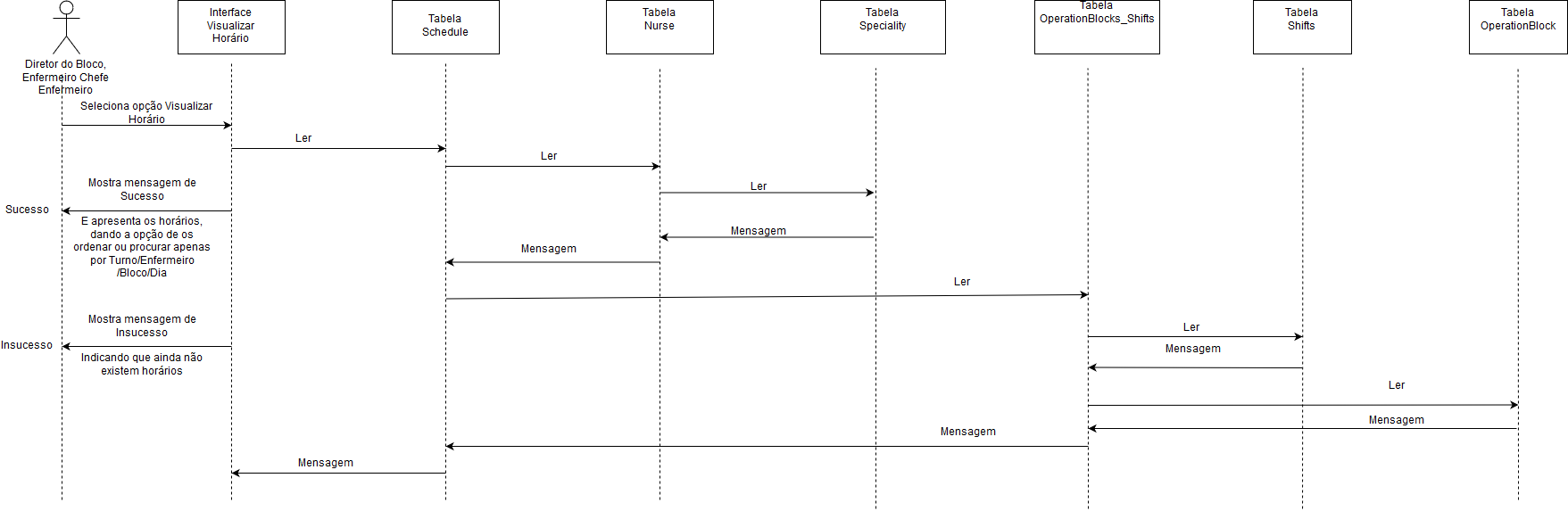


Diagrama de Sequência Gerar Horário(updated)

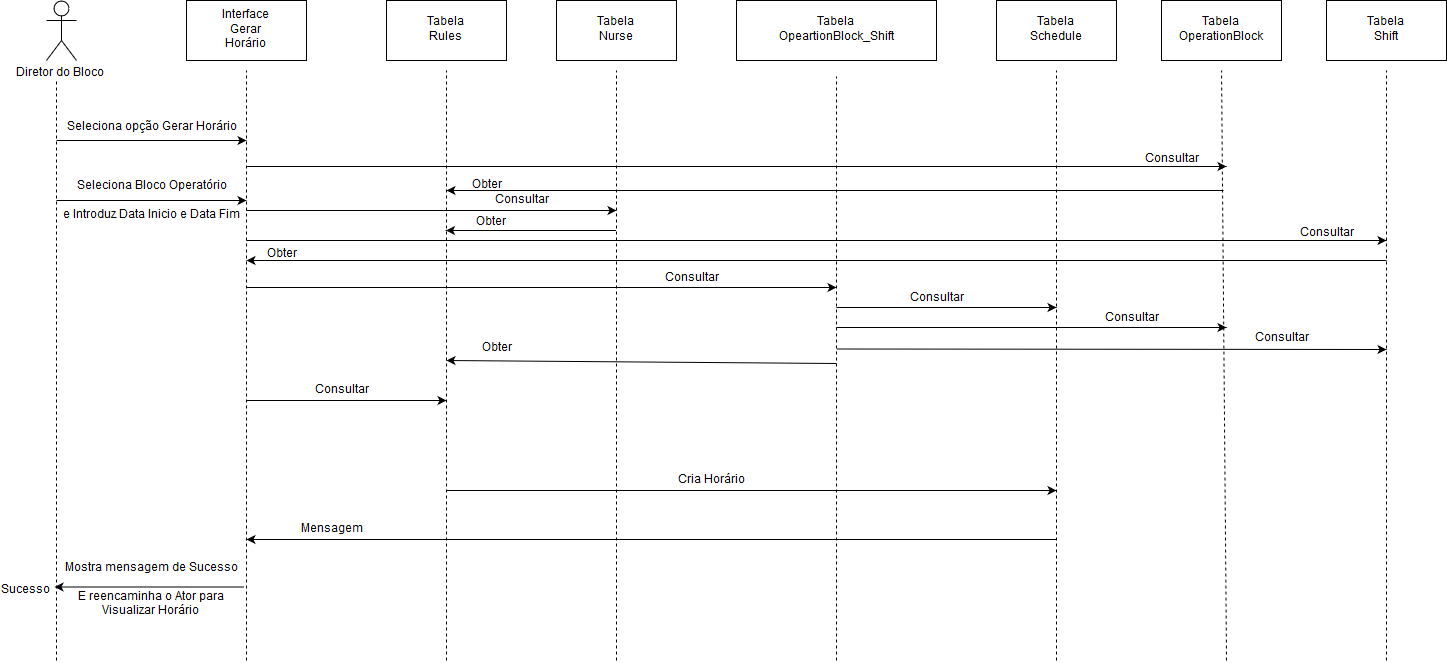


Diagrama de Sequência Pedir Troca de Turnos (Updated)

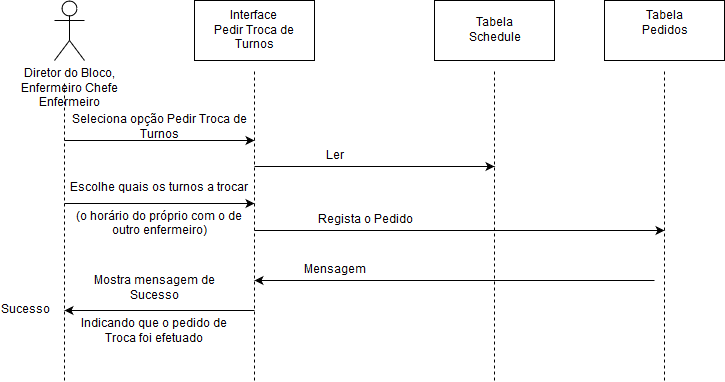
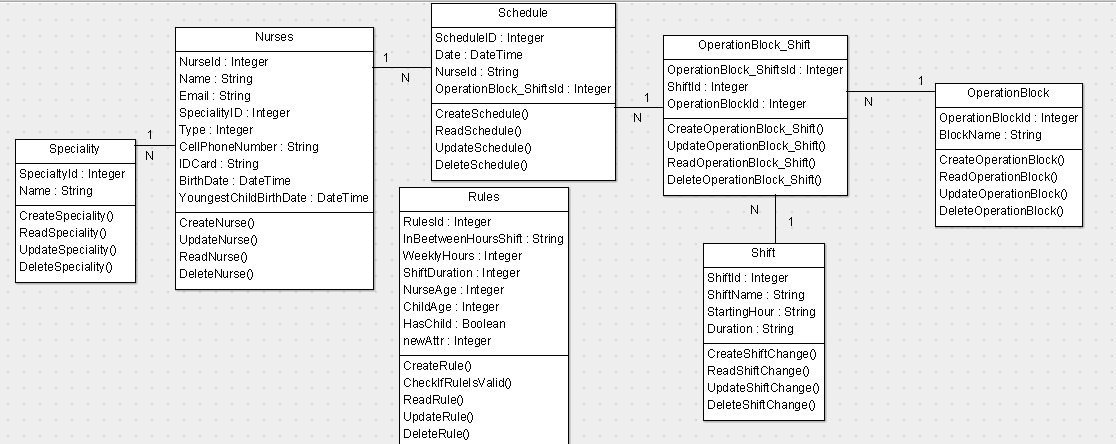


Diagrama de Classes(colocar tabela dos pedidos?)



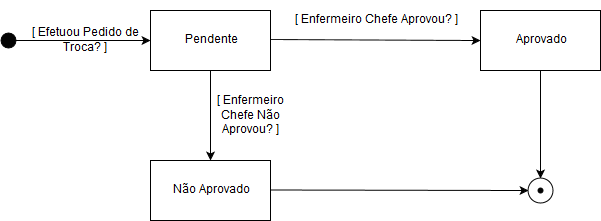
Semântica de Classes:

Classe Nurse

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Nurse | | | | | | |
| Nome do campo | Tipo de Dados | Descrição | Valores Válidos | Formato | Restrições | |
| NurseId | Inteiro | Chave Primária. | Números/Maior que zero | Ate 5 dígitos | Gerado pelo Sistema/Obrigatório/não alterável | |
| Name | String | Identifica o nome do enfermeiro. | Caracteres | Até 20 caracteres | Obrigatório/Alterável | |
| Email | Sring | Email do enfermeiro. | Caracteres | Até 256 caracteres | Obrigatório/Alterável | |
| Type | Inteiro | Indica o tipo de enfermeiro (enfermeiro=0; enfermeirChefe=1).Deixamos do tipo inteiro para permitir expansibilidade. | Números | Um dígito (0/1) | Obrigatório/Alterável | |
| CellPhoneNumber | Inteiro | Indica o contacto do enfermeiro. | Caracteres | 9 caracteres | Obrigatório/Alterável | |
| IDCard | Bool | Identificação do enfermeiro. | Carateres | 12 caracteres | Obrigatório/Alterável | |
| BirthDate | Date | Data de nascimento do enfermeiro. | Números | dd/MM/yyyy | Obrigatório/Alterável |
| YoungestChildBirthDate | Date | Data do filho mais novo do enfermeiro. | Números | dd/MM/yyyy | Alterável |

Diagrama de Atividades

Diagrama de Estados



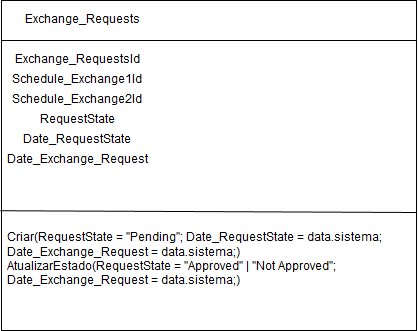


Diagrama de Componentes

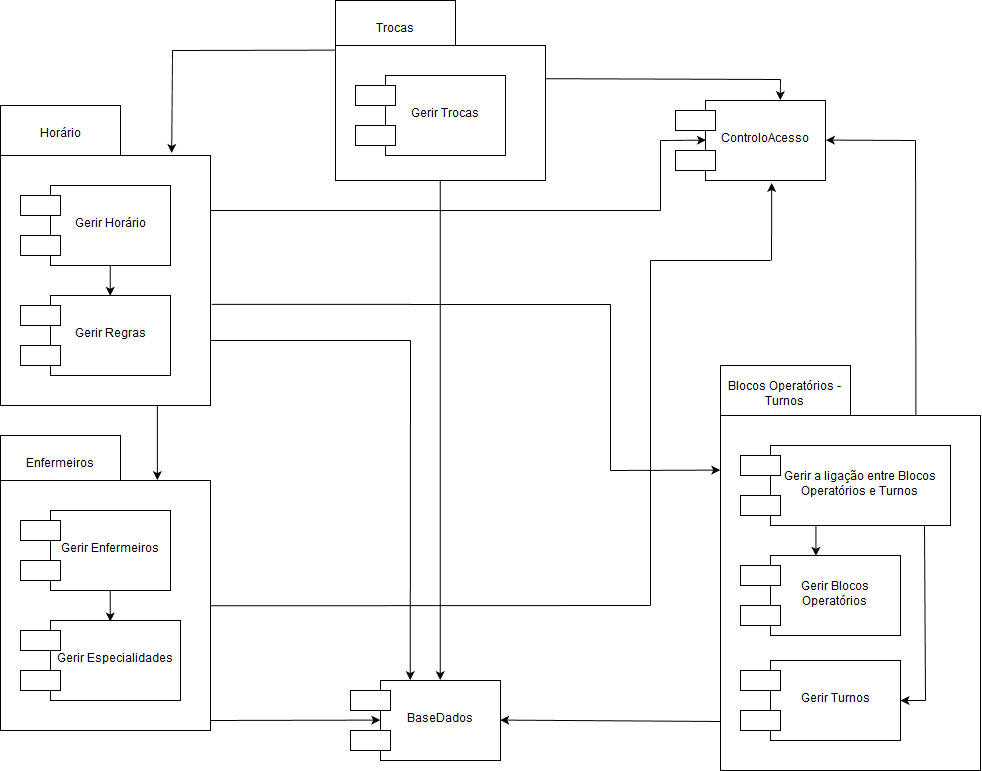
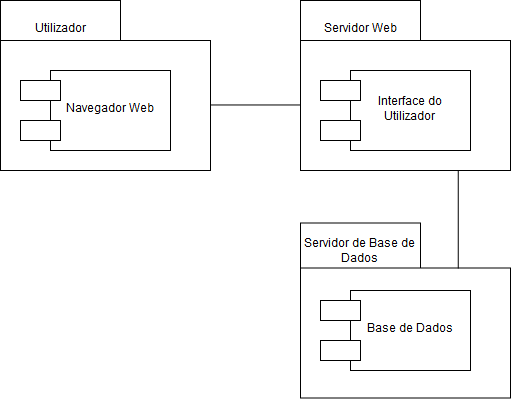


Diagrama de Instalação

Este diagrama descreve as vertentes de software e de hardware do sistema.



Na Interface do Utilizador estão situados todos os componentes do diagrama de componentes acima.

Na Base de Dados estão situadas as seguintes classes: Schedule, Nurse, Specialty, Operation Block, Shift, Operation Block Shift, Exchange Request, Schedule Exchange1, Schedule Exchange2.

Algoritmo(s) para gerar horários

1.Obter o nome do bloco a escalonar.

2.Obter o nº de enfermeiros disponíveis.

3.Obter a duração do turno.

4.Obter as horas semanais de cada enfermeiro.

5.Obter o nº de horas entre turnos.

6.Obter a idade limite do filho mais novo.

7.Calcular o nº de turnos(pela divisão inteira de 24h com a duração de cada turno).

8.Fazer se:

8.1.Bloco não estiver cheio e Turno for disponivel e Enfermeiros disponiveis:

8.1.Para cada enfermeiro, fazer:

8.1.1.Obter a idade e a idade do filho mais novo e o cargo//Pelo cálculo da data de nascimento do enfermeiro.

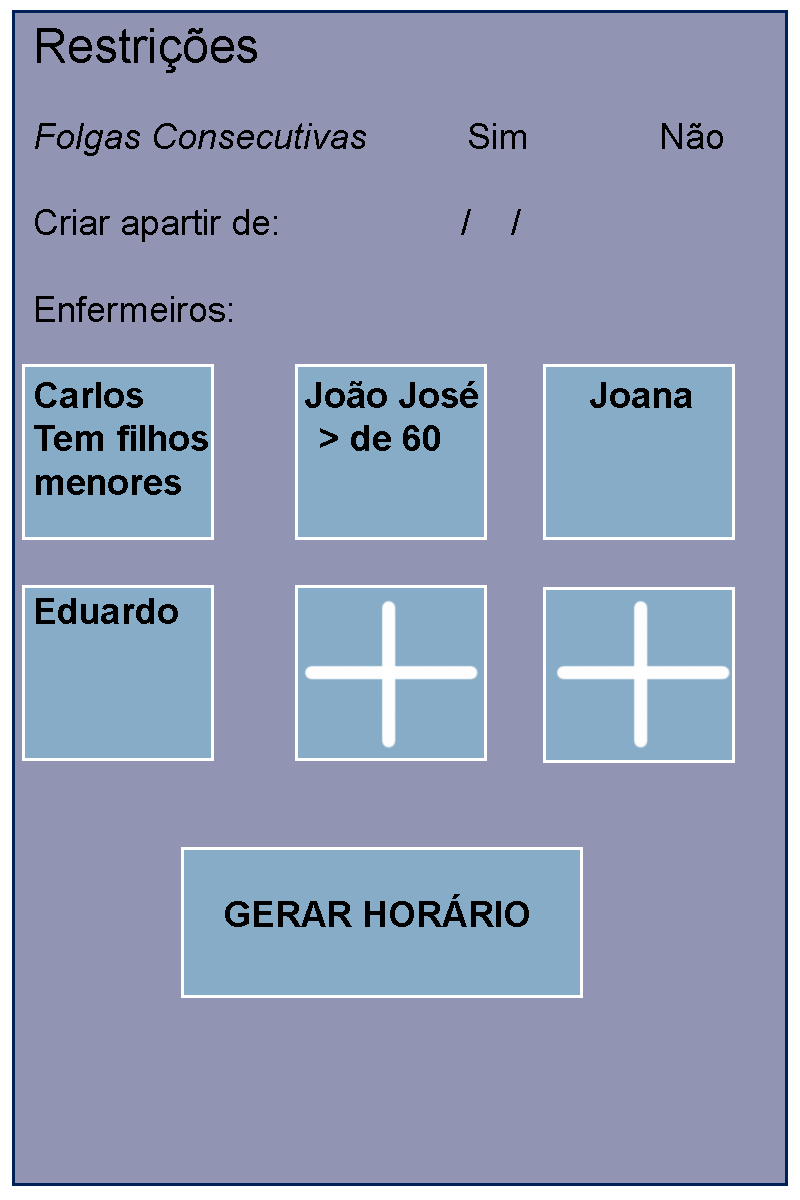
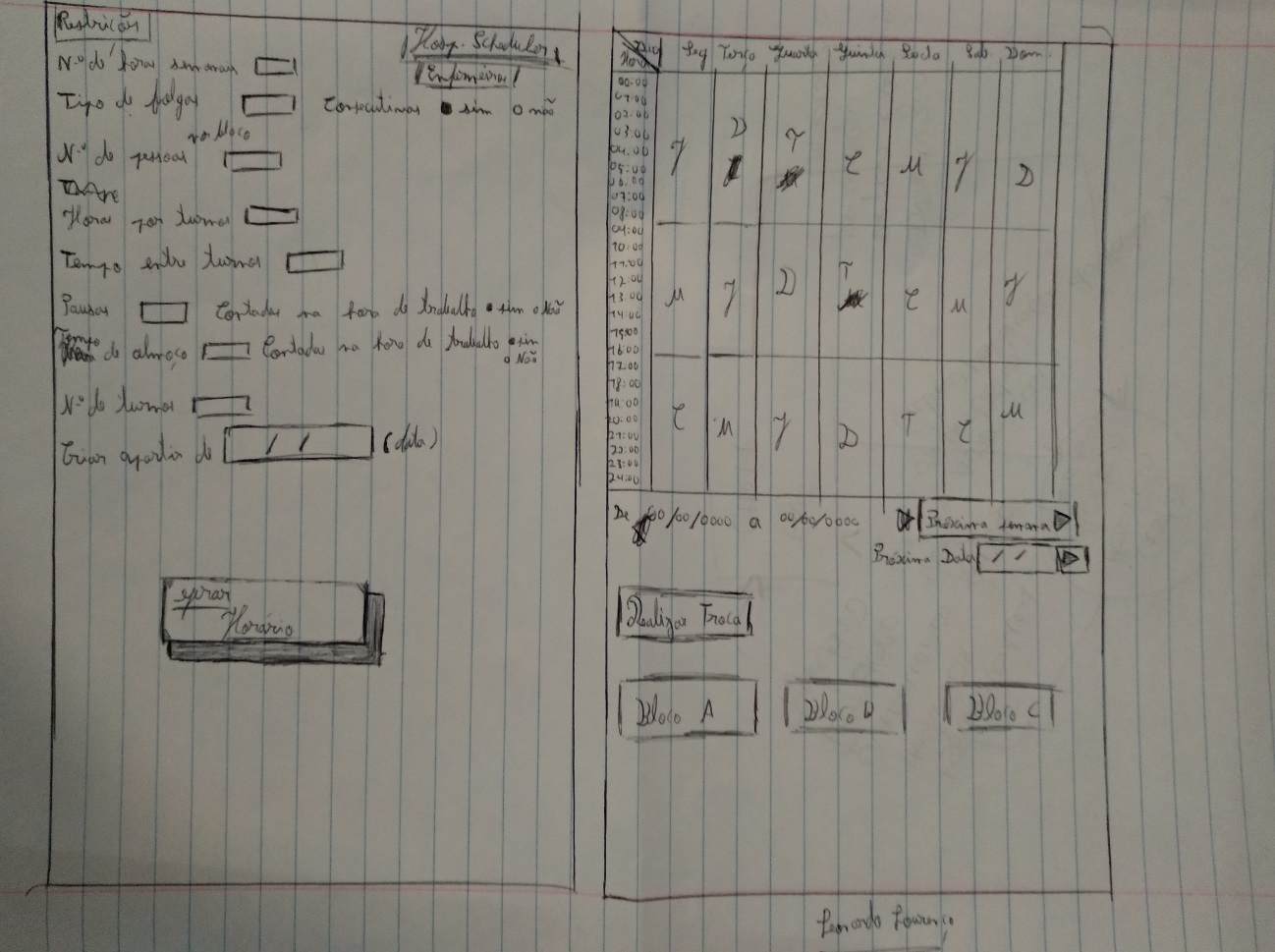
8.1.1.1.Se o enfermeiro tiver mais de x anos e ou se o enfermeiro tiver o filho menor que x anos e ou se o enfermeiro for o enfermeiro-chefe , então:

8.1.1.1.1.Não faz turnos de noite.

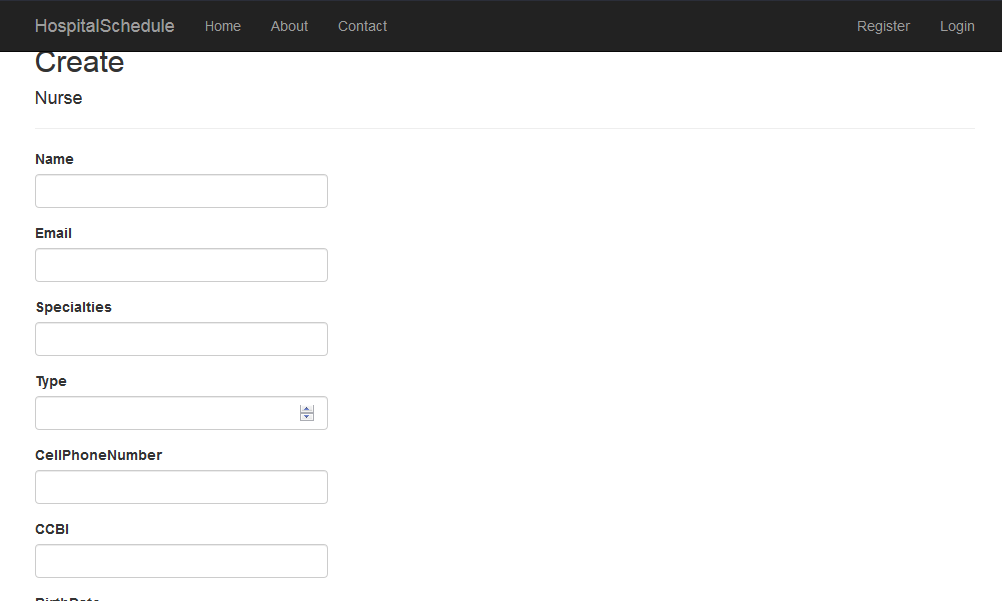
8.1.1.2.Se não:

8.1.1.2.1.Escalona o enfermeiro num turno disponível(por ordem da hora do turno).

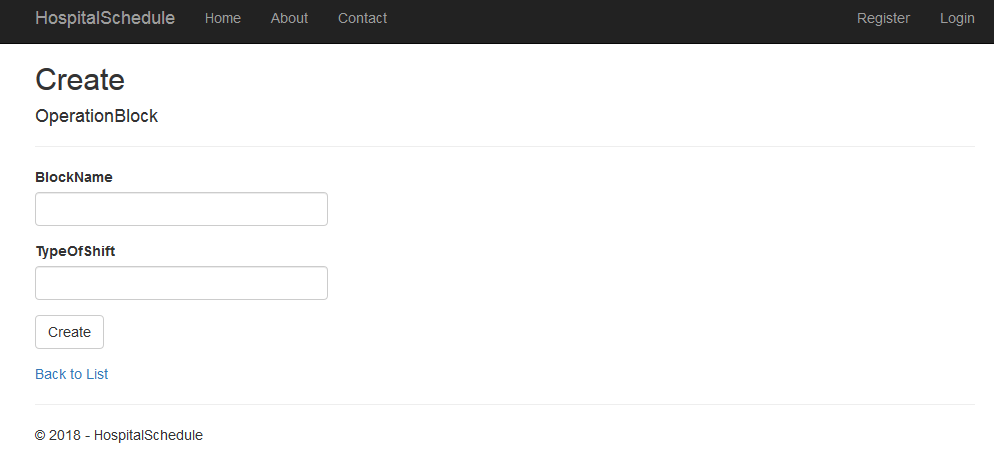
9.Adicionar o nome do bloco no início da tabela.

Protótipos

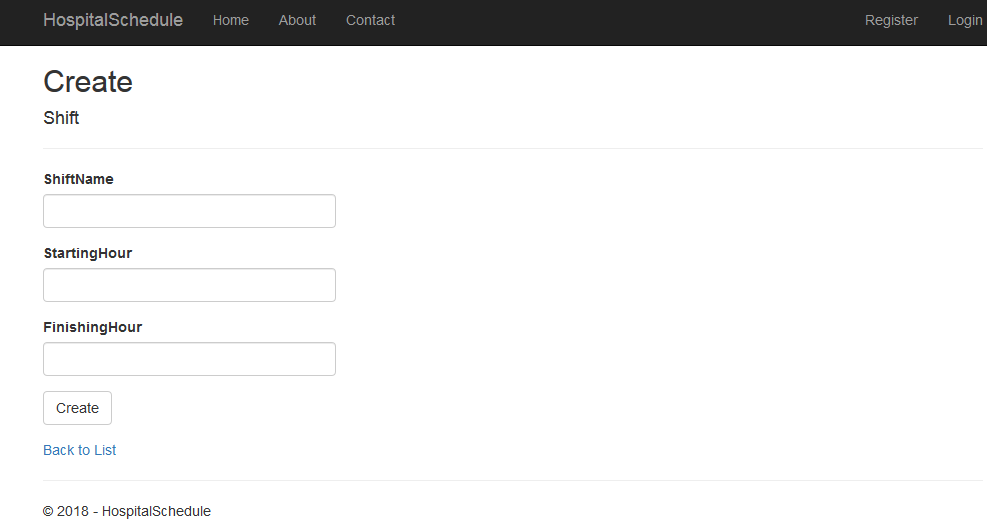
Protótipo Inserir enfermeiros



Protótipo Inserir Bloco Operatório



Protótipo Inserir Turno



Atividades e tempos gastos em horas por elemento de grupo

Leonardo Lourenço:

Organização e Planeamento – 10H

Desenvolvimento de Protótipos – 5H

Desenvolvimento de Diagramas de caso de uso – 5H

Desenvolvimento de Diagramas de Sequência – 4H

Desenvolvimento dos Atores e respetivos casos de uso – 3H

Padrão do caso de Uso: BreathBeforeDepth – 4H

CRUD Horário – 20H

CRUD Blocos Operatórios – 8H

CRUD Horário\_Turnos\_BlocosOperatórios – 6H

CRUD Turnos – 3H

CRUD Enfermeiros – 3H

CRUD Regras – 3H

Expressões Regulares em diversas tabelas – 2H

Bruno Santos:

CRUD Horário-10H

CRUD Turnos-10H

CRUD Blocos Operatórios –6H

CRUD Turnos – 2H

CRUD Enfermeiros – 1H

Luciano Brido:

Desenvolvimento de Diagramas de contexto – 2H

Desenvolvimento de Diagramas de caso de uso – 6H  
CRUD Enfermeiros – 15H

CRUD Regras – 2H