

INSTITUTO FEDERAL DE EDUCAÇÃO SUL-RIO-GRANDENSE
CURSO SUPERIOR DE TECNOLOGIA EM SISTEMAS PARA INTERNET

2º MÓDULO

GÈRE

PELOTAS

2011

2º MÓDULO

GÈRE

Projeto de conclusão do segundo módulo do Curso Superior de Sistemas para Internet do Instituto Federal de Educação, Ciência e Tecnologia Sul-Rio-Grandense .

Orientadores: Márcia Froehlich, Michele Schmidt, Simone Carbone e Raquel Godinho.

Pelotas

2011

2º MÓDULO

GÈRE

Projeto de conclusão do segundo módulo do Curso Superior de Sistemas para Internet do Instituto Federal de Educação, Ciência e Tecnologia Sul-Rio-Grandense .

Aprovado em ____ de _____ de ____.

BANCA EXAMINADORA

Profª. Michele de Almeida Schmidt– IFSUL

Profª. Márcia Froehlich – IFSUL

Profª. Simone Carbone – IFSUL

Profª. Raquel Godinho – IFSUL

AGRADECIMENTO

É com muita honra que a turma do 2º módulo, agradece aos seus professores do curso TSI pelos seus ensinamentos, apoio e conselhos, principalmente as professoras Márcia Froehlich, Michele Schmidt, Simone Carbone e Raquel Godinho.

Aos colegas do Projeto Gêre, estão todos de parabéns pela colaboração na realização do mesmo pela paciência e dedicação.

“O único lugar onde o sucesso vem antes do trabalho é no dicionário.”

(Albert Einstein)

RESUMO

O presente trabalho tem como objetivo fazer a documentação do Sistema Gère. Este sistema visa gerenciar a parte administrativa do curso do Curso TSlaD, com o objetivo de facilitar o controle e manutenção de cronogramas de aulas e de viagens, permitir o acesso de alunos ao sistema possibilitando a integração deste com as empresas que ofertarão vagas, permitindo a inclusão do aluno no mercado de trabalho. As empresas poderão conhecer o perfil dos candidatos já que estes irão postar seus currículos. Além disso, permitirá que a coordenação e administração do TSlaD controle os bens patrimoniais e de consumo do curso. A modelagem do sistema foi documentada a partir dos conteúdos estudados pelos alunos do Segundo Módulo do TSI. A criação do sistema Gère foi sugerida pelos professores do curso, pois estes perceberam a necessidade de criar um sistema que apoiasse a parte administrativa do TSlaD e que os alunos pudessem colocar em prática os conhecimentos construídos ao longo do semestre.

Palavras-chave: Sistema, Gère, TSlaD, módulos.

ABSTRACT

This study aims to make the system to generate documentation. This system aims to manage the administrative part of the course Course TSlaD, in order to facilitate the control and maintenance schedules of classes and trips, allow students access to the system allowing the integration of this with companies that ofertarão vague, allowing inclusion of students in the labor market. Companies can know the backgrounds of the candidates as they will post their resumes. Also, allow the TSlaD Coordination and administration of assets and control the consumption of the course. The modeling system has been documented from content studied by students of the second module of the IST. This system was suggested by the teacher manages the course as they realized the need to create a system that supports the administrative part of TSlaD and that students could put into practice the knowledge built during the semester.

Keywords: System, Gère TSlaD, modules.

LISTA DE ILUSTRAÇÕES

Figura 1: Diagrama de Contexto do Sistema Gère.....	35
Figura 2: Diagrama de nível 1 do Módulo Aluno.....	36
Figura 3: Diagrama de nível 1 do Módulo Empresa.....	37
Figura 4: Diagrama de nível 1 do Módulo Patrimônio	37
Figura 5: Diagrama de nível 1 do Módulo Viagens.....	38
Figura 6: Diagrama de nível 1 do Módulo Professor.....	39
Figura 7: Diagrama de nível 1 do Módulo Cronograma	40
Figura 8: Diagrama de nível 2 gerenciar vagas do Módulo Empresa	41
Figura 9: Diagrama de nível 2 gerenciar patrimônio do Módulo Patrimônio ...	42
Figura 10: Diagrama de nível 2 consultar viagens do Módulo Viagens.....	42
Figura 11: Diagrama de nível 2 protocolar viagens do Módulo Viagens.....	43
Figura 12: Diagrama de nível 2 cadastro de usuarios do Módulo Viagens.....	43
Figura 13: Diagrama de nível 2 gerenciar viagem do Módulo Viagens	44
Figura 14: Diagrama de nível 2 gerenciar cronograma do Módulo Cronograma	45
Figura 15: Modelo conceitual do Sistema Gère.....	108
Figura 16: Modelo Lógico - Administradores	111
Figura 17: Modelo Lógico - Alunos.....	111
Figura 18: Modelo Lógico – Classificação	111
Figura 19: Modelo Lógico – Concorre	112
Figura 20: Modelo Lógico – Diárias.....	112
Figura 21: Modelo Lógico – Disciplinas	112
Figura 22: Modelo Lógico – Empresas	113
Figura 23: Modelo Lógico – Eventos	113

Figura 24: Modelo Lógico – Frequenta	113
Figura 25: Modelo Lógico – Módulos	114
Figura 26: Modelo Lógico – Módulo_polos	114
Figura 27: Modelo Lógico – Motoristas.....	114
Figura 28: Modelo Lógico – Participa.....	115
Figura 29: Modelo Lógico – Patrimônio	115
Figura 30: Modelo Lógico – Pedido	115
Figura 31: Modelo Lógico – Pedido_patrimônio	116
Figura 32: Modelo Lógico – Plano_ensino.....	116
Figura 33: Modelo Lógico – Polos	116
Figura 34: Modelo Lógico – Professores.....	117
Figura 35: Modelo Lógico – Telefones	117
Figura 36: Modelo Lógico – Tipo	117
Figura 37: Modelo Lógico – Tipo_eventos.....	118
Figura 38: Modelo Lógico – Transporte_motorista	118
Figura 39: Modelo Lógico – Transportes.....	118
Figura 40: Modelo Lógico – Usuários	119
Figura 41: Modelo Lógico – Vagas.....	119
Figura 42: Modelo Lógico – Viagens	120
Figura 43: Leituras do Logotipo (marcação)	135
Figura 44: Tela "Index" do sistema	136
Figura 45: Tela interna do Módulo Aluno	137
Figura 46: Tela Interna cronograma do Módulo Alunos	138
Figura 47-Tela Interna cronograma de viagens do Módulo Viagens	139
Figura 48: Tela interna cadastrar viagem do Módulo Viagens	140
Figura 49: Tela interna consultar viagem do Módulo Viagens.....	141

Figura 50:Tela interna alterar viagem do Módulo Viagens.....	142
Figura 51: Tela interna cronograma do Módulo Cronograma	143
Figura 52: Tela interna cronograma – alunos	144
Figura 53: Tela interna professores cronograma do Módulo Professor.....	145
Figura 54: Tela interna do Módulo Patrimônio	146
Figura 55: Tela interna alterar do Módulo Patrimônio	147
Figura 56: Tela interna cadastrar do Módulo Patrimônio.....	148
Figura 57: Tela interna pedir do Módulo Patrimônio	149
Figura 58: Tela interna do Módulo Empresa	150
Figura 59: Tela interna vagas-alunos do Módulo Empresa	151
Figura 60: Tela interna vagas-curriculo-geral do Módulo Empresa.....	152
Figura 61: Tela interna vagas-cadastradas do Módulo Empresa	153
Figura 62: Tela interna do Módulo Professores.....	154

LISTA DE TABELAS

Tabela 1: Notação utilizada no Dicionário de Dados.....	46
Tabela 2: Entidade do Módulo Viagens	47
Tabela 3: Processos do Módulo Viagens	47
Tabela 4: Fluxo de Dados do Módulo Viagens	49
Tabela 5: Depósito de Dados do Módulo Viagens	56
Tabela 6: Entidade Empresa.....	61
Tabela 7: Processos do Módulo Empresas	61
Tabela 8: Depósito do Módulo Empresas	62
Tabela 9: Fluxo de Dados Módulo Empresas	62
Tabela 10: Fluxo de Dados do Módulo Cronograma.....	67
Tabela 11: Entidade do Módulo Aluno	81
Tabela 12: Processos do Módulo Aluno	82
Tabela 13: Fluxo de Dados do Módulo Aluno	82
Tabela 14: Depósito do Módulo Aluno	87
Tabela 15: Fluxo de Dados do Módulo Professor	89
Tabela 16: Depósito de dados do Módulo Professor	93
Tabela 17: Processos do Módulo Professor.....	94
Tabela 18: Entidade do Módulo Professor.....	96
Tabela 19: Entidade do Módulo Patrimônio	96
Tabela 20: Processos do Módulo Patrimônio	96
Tabela 21: Fluxo de Dados do Módulo Patrimônio	97
Tabela 22: Depósito do Módulo Patrimônio	104
Tabela 23 Legenda para simbologia utilizada nas colunas nome do campo das tabelas	56

Tabela 24: para simbologia utilizada nas colunas tipo de dados das tabelas . 56

LISTA DE SIGLAS

BD: Banco de Dados

CASE: Computer-Aided Software Engineering

CEP: Código de Endereçamento Postal

CNPJ: Cadastro Nacional de Pessoa Física

CPF: Cadastro Pessoa Física

DD: Dicionário de Dados

DER: Diagrama Entidade Relacionamento

EAD: Ensino a Distância

ER: Entidade Relacionamento

IFSUL: Instituto Federal De Educação, Ciência E Tecnologia Sul-Rio-Grandense

MLD: Modelo Lógico de Dados

RG: Registro Geral

SGBD: Sistema Gerenciador de Banco de Dados

SIAPE: Sistema Integrado de Administração de Pessoal

SIAPE: Sistema Integrado de Administração de Pessoal

TSlad: Tecnologia em Sistema para Internet à Distância

UAB: Universidade Aberta do Brasil

SUMÁRIO

1. INTRODUÇÃO	12
2. ANÁLISE DE DADOS.....	13
2.1 Avaliações de Requisitos	14
2.2 Análise Funcional.....	16
3. ESTRUTURA DO SISTEMA.....	18
3.1 Objetivos específicos do sistema:	19
3.2 Descrição e funcionalidades dos módulos	21
3.2.1. Módulo de Viagens.....	21
3.2.2 Módulo Cronograma.....	26
3.2.3 Módulo Patrimônio	27
3.2.4 Módulo de Professores	29
3.2.5 Módulo Empresas	30
3.2.6 Módulo Alunos.....	31
3.3 Usuários Envolvidos.....	32
3.3 Resultados Esperados	33
4. DIAGRAMA DE FLUXO DE DADOS.....	34
4.1 Diagrama de contexto	34
4.2 Diagramas de nível um	36
4.3 Diagramas de nível dois.....	41
5. DICIONÁRIO DE DADOS.....	46
6. PROJETO DE BANCO DE DADOS	106
61 Modelagem Conceitual	106
62 Modelo Lógico.....	109
6.3 Modelo Físico : Linguagem de Consulta Estruturada ou Structured Query Language (SQL).....	120

7.INTERFACE DO SISTEMA	134
7.1Nome e Logotipo.....	134
7.2 Interface do Sistema	135
8.CONCLUSÃO	155

1. INTRODUÇÃO

O projeto foi desenvolvido com a finalidade de suprir as necessidades administrativas do curso de Tecnologia em Sistema para Internet a distância (TSlad), ofertado pela Universidade Aberta do Brasil (UAB). O curso é ministrado em cinco polos situados em cidades diferentes. Atualmente a administração destes polos é feita manualmente através de documentos impressos e a comunicação entre os administradores é feita da mesma forma. Assim, visando integrar estas informações o sistema irá agrupar todos os dados em único banco de dados e para facilitar a operacionalidade será dividido em módulos.

Este projeto foi construído ao longo do Segundo Módulo do curso TSI proporcionando a aplicação dos conteúdos trabalhados pelas disciplinas de Análise e Projeto de Sistemas, Projeto de Banco de Dados Relacional, Design e Projeto de Sites e Comunicação e Redação. Além disso, permitiu desenvolver habilidades de trabalho em equipe fortalecendo as relações humanas promovendo uma experiência prática que não se difere da encontrada no mercado de trabalho.

Os capítulos, a seguir, descreverão as atividades desenvolvidas, com embasamento teórico e prático.

No primeiro capítulo buscamos descrever o estado da arte que é o nível mais alto de desenvolvimento, de uma técnica científica embasado no conhecimento fornecido por autores como Roger Pressman, Carlos Heuser, Anielle Damasceno, Edward Yourdon, entre outros.

Na estrutura do sistema, esboçamos os objetivos da construção, os motivos pelos quais ele foi desenvolvido e ainda os usuários que serão beneficiados com a criação do Gère. Ainda neste capítulo descrevemos cada módulo que o compõe com uma descrição das suas funcionalidades.

Nos demais capítulos, apresentamos a modelagem do sistema, com os diagramas de fluxos de dados, projeto de banco de dados, dicionário de dados e o projeto de interface.

2. ANÁLISE DE DADOS

A utilização de computadores, nas mais diversas áreas, tem gerado uma crescente demanda por soluções computadorizadas. O software tornou-se elemento chave na evolução de sistemas e é uma das tecnologias mais importantes em todo o mundo. Ao longo do tempo o software evoluiu, passando por várias alterações na maneira de se pensar o cliente e o produto, mas ainda existem muitos problemas na construção de software de qualidade entregues ao cliente dentro do prazo e dentro do orçamento disponibilizado pelo cliente.

Visando melhorar a qualidade dos produtos de software e aumentar a produtividade no processo de desenvolvimento, surgiu a engenharia de software. Ela trata de aspectos relacionados ao estabelecimento de processos, métodos, técnicas, ferramentas e ambientes de suporte ao desenvolvimento de sistemas.

Com a intenção de apoiar a construção de um software antes da programação propriamente dita é preciso projetar cada detalhe do aplicativo antes que ele seja desenvolvido. PRESSMAN (2010) diz que as essências da prática da engenharia são: a comunicação e análise, a modelagem e projeto de software, a geração do código de programação e o teste, buscando a garantia de qualidade.

Seguindo os conceitos deste mesmo autor, aplicamos à primeira prática de comunicação e análise construindo modelos para o desenvolvimento do sistema, tais modelos podem ser divididos em:

- Diagramação dos eventos e atividades: Avaliação de requisitos;
- Análise funcional do sistema: Onde são construídos diagramas de fluxos de dados que serão utilizados pelos desenvolvedores no momento da programação do sistema e para que o cliente possa entender o que será desenvolvido;
- Construção de modelos de dados: Baseados nas técnicas de geração de modelos lógicos e conceituais para armazenamento e processamentos de dados visam à melhor estruturação dos dados que serão armazenados em bancos de dados;
- Projeto de interface: Criando uma prototipagem do software que será desenvolvido fornecendo ao cliente uma visão real do que será projetado permitindo

assim que ele avalie se aplicativo atenderá a suas necessidades em quanto à usabilidade e funcionalidade prática.

2.1 Avaliações de Requisitos

Os requisitos constituem um conjunto de necessidades definidas pelo cliente, o qual é responsável por definir a estrutura e o comportamento do software que será desenvolvido. Os requisitos podem ser dados, processos, funcionalidades ou restrições que variam de acordo com as regras do negócio. É a base para o projeto e desenvolvimento do software, sua importância é tamanha que são considerados responsáveis pelo sucesso ou fracasso do mesmo. A engenharia de requisitos apresenta duas atividades principais, o levantamento e a análise de requisitos.

Nesta etapa verificaram-se os requisitos funcionais e não funcionais do sistema. Os requisitos funcionais são aqueles que descrevem o comportamento do sistema, ou seja, eles declaram como o sistema deve reagir a entradas específicas e como devem ser geradas as saídas para o usuário, ou seja, é aquilo que descreve o que tem que ser feito pelo sistema.

Durante a análise feita junto com o cliente, através da aplicação de técnicas de entrevistas abertas, foi levantada a necessidade de que o sistema tivesse uma divisão operacional em módulos na qual, cada um teria requisitos funcionais específicos que são descritos no capítulo 3.2.

Já os requisitos não funcionais são as qualidades de um software. Eles descrevem apenas atributos do sistema ou atributos do ambiente. Dizem respeito a questões de viabilidade de implementação.

O estudo de viabilidade levou em consideração a análise custo x benefício, os riscos do projeto e o cronograma proposto. Neste estudo, foram identificados os seguintes elementos:

Benefícios da implementação:

- Agilidade no processamento de requisições de viagem, controle de patrimônio, consultas de viagens e eventos nos polos de ensino;

- Maior controle sobre as aulas ministradas nos polos;
- Diminuição do número de erros tanto na liberação de verbas quanto na organização e controle de eventos realizados nos polos;
- Melhoria na qualidade das informações gerenciais com redução no tempo de obtê-las, facilitando a professores e administradores a consulta de suas agendas;
- Cria uma porta de comunicação entre empresas e alunos buscando facilitar a inserção destes no mercado de trabalho.
- Permite a coordenação um controle maior sobre o patrimônio vinculado a universidade;
- Permite que professores tenham uma comunicação simplificada com os alunos podendo visualizar suas avaliações.

Custos de desenvolvimento, implantação e utilização do sistema:

- Não existem custos relativos ao uso de computadores, pois por tratar-se de projeto acadêmico tais recursos físicos são fornecidos pela União;
- Não há custos com salários e encargos trabalhistas, visto que, os envolvidos no projeto são alunos da instituição;
- Com relação ao material de consumo no desenvolvimento do sistema (papel, suprimentos, mídias, etc.) são fornecidos pelos alunos engajados no projeto e fazem parte dos bens de uso pessoal de cada envolvido.

Riscos legais e econômicos:

- Não foram detectados riscos por tratar-se de um aplicativo para fins acadêmicos.

Prazos estabelecidos (planejamento do projeto):

- O prazo estabelecido pelos orientadores do projeto apresenta-se viável para o planejamento do sistema e comporta um semestre letivo.

2.2 Análise Funcional

Baseados na análise dos requisitos funcionais foram criados diagramas de fluxo de dados para definir o relacionamento entre o sistema e o meio externo que são chamados de entidades externas ao sistema.

O Diagrama de Fluxo de Dados (DFD) é uma ferramenta para modelagem de fluxo de dados, através da representação de processos que usam e geram dados. A representação do DFD nos permite imaginar um sistema como uma rede de processos funcionais, interligados por 'dutos' e 'tanques de armazenamento' de dados (YOURDON,1990).

No DFD representamos e projetamos as partes que irão compor o sistema antes de seu desenvolvimento. Desta forma, neste diagrama, enfatizamos os processos que serão realizados e os dados que passam entre eles. Assim, a partir deste diagrama, é possível compreender como será o sistema a ser implementado evitando redundâncias e omissões próprias da análise de dados.

A estrutura do DFD, segundo Yourdon (1990) apresenta quatro elementos para modelar o sistema em termos conceituais e funcionais que são:

- **Entidade externa:** são, usualmente, categorias lógicas de coisas ou pessoas que interagem recebendo ou informando dados. É representado por um retângulo;
- **Fluxo de dados:** são os elementos de ligação de um DFD. É o caminho por onde passam os dados de um ponto a outro do sistema;
- **Processo:** ponto de processamento dos dados (tratamento, alteração ou manuseio de dados) . É identificado por um verbo no imperativo;
- **Depósito de dados:** indica qualquer tipo de armazenamento de dados.

Em relação ao DFD Yourdon diz que:

“O DFD deve ser modelado em uma série de níveis de modo que a cada nível ofereça sucessivamente mais detalhes sobre uma parte do nível que lhe seja superior. O DFD de nível mais alto consiste de uma única bolha, representando o sistema inteiro; os fluxos de dados mostram as interfaces entre o sistema e os terminadores externos. Esse DFD especial é conhecido como diagrama de contexto. O DFD imediatamente abaixo do diagrama de contexto é conhecido como figura 0. Ele representa a visão de mais alto

nível das principais funções do sistema bem como as principais interfaces entre essas funções.” (YOURDON, 1990, p. 159).

Também foram gerados dicionários de dados para que a interpretação dos fluxos sejam claras e precisas. O dicionário de dados pode ser visto como um depósito central que descreve e define o significado de toda a informação usada na construção de um sistema. Ele permite fazer a verificação de consistência entre as informações constantes no DFD.

Após o processo de análise de requisitos surge então a necessidade de se avaliar o uso de um banco de dados e construir a modelagem dos dados que formam a estrutura formal do banco. HEUSER (1998, p. 22) diz que: “um banco de dados é o conjunto de dados integrados que tem por objetivo atender a uma comunidade de usuários”.

Bancos de dados são componentes importantes dos sistemas de informação, pois eles são responsáveis por possibilitar a reutilização das informações do sistema e do cliente. A professora Marina Teresa Pires Vieira da Universidade Federal de São Carlos em seu artigo sobre Projeto de banco de dados diz que:

“O projeto do banco de dados apresenta-se como uma atividade essencial na fase de desenvolvimento dos sistemas. Projetar bancos de dados tem se tornado uma atividade popular, as vezes realizada não somente por profissionais da área de banco de dados, mas também por não especialistas. Frequentemente, a falta de abordagens adequadas para o projeto de um banco de dados pode incorrer em resultados indesejáveis, como ineficiência em atender a demanda de aplicações e problemas com a manutenção do banco de dados. Geralmente a causa disso é a falta de clareza em entender a natureza exata dos dados em um nível conceitual (abstrato). (TERESA, 2003, p.3)”

Existem linguagens de modelagem para descrever modelos de dados em diferentes níveis de abstração e com diferentes objetivos. Geralmente são considerados dois níveis de abstração: o modelo conceitual e modelo lógico e ainda o modelo físico do banco de dados que traz as características da estrutura do banco.

Trataremos mais detalhadamente este projeto no capítulo 5.

3. ESTRUTURA DO SISTEMA

Um sistema segundo Laudon & Laudon (2007, p. 9) “pode ser definido tecnicamente como um conjunto de componente inter-relacionados que coletam, processam, armazenam e distribuem informações destinadas a apoiar a tomada de decisão, a coordenação e o controle de uma organização”. Neste sentido, o presente sistema, terá módulos interdependentes que irão interagir com objetivos comuns formando um todo. Cada um dos módulos comporta-se, por sua vez, como um sistema cujo resultado é maior do que o resultado que as unidades poderiam ter se funcionassem independentemente (ALVAREZ, 1990, p. 17).

Assim, a partir dos conceitos trazidos por estes autores pensou-se na construção do sistema Gère. A fim de atender as necessidades de controle e facilitar a comunicação entre os gestores ligados ao curso de TSIaD, surgiu a idéia de se implantar um sistema que facilitasse a integração das informações e que ainda pudesse ser usado para aplicar os conhecimentos didáticos adquiridos pelos desenvolvedores, estes, que são alunos do TSI módulo presencial da instituição.

O Sistema, também, tem como objetivo auxiliar no gerenciamento do curso TSIAD tornando a administração mais simples e intuitiva e terá sua estruturação dividida em módulos, cada módulo com suas respectivas funcionalidades e distinções.

Como Gère trata de um sistema administrativo, assim não se pode deixar de pensar um pouco na teoria clássica da administração idealizada por Henri Fayol. A teoria foi desenvolvida porque Henri estava “preocupado em aumentar a eficiência da empresa por meio de sua organização e da aplicação de princípios gerais da Administração em bases científicas” (2003, p. 74). Desta forma, buscando a eficiência dos processos administrativos da UAB, o sistema Gère, foi desenvolvido para proporcionar a seus usuários a gestão e controle eficaz das funções ligadas ao curso de educação à distância.

A sua criação se justifica:

- Os funcionários da UAB precisam de uma ferramenta de controle das atividades relacionadas à administração do ensino a distância;

- O sistema integra informações sobre as atividades acadêmicas realizadas nos polos de ensino a distância;
- Os alunos necessitam de acesso a informações detalhadas não disponíveis no sistema moodle no que diz respeito a eventos que ocorrem nos polos;
- Os professores precisam ter conhecimento antecipado sobre o plano de aulas e viagens para os polos;
- Os coordenadores do TSIaD precisam controlar os bens patrimoniais e de consumo da entidade;
- Os usuários que agendam as viagens para os polos precisam de um sistema que agrupe todas as informações sobre as aulas que, seja fácil e dinâmico para agendar viagens.

A fim de facilitar a organização do sistema e torná-lo mais operacional, o sistema é dividido em seis módulos que terão suas funcionalidades descritas no próximo subcapítulo:

- Módulo de Alunos
- Módulo de Professores
- Módulo de Empresas
- Módulo de Viagens
- Módulo de Cronograma
- Módulo de Patrimônio

3.1 Objetivos específicos do sistema:

Embora o sistema esteja dividido em módulos é importante esclarecer os objetivos específicos do sistema, para que fique clara a sua implementação:

- Permitir, ao professor, a solicitação de cancelamento e/ou re-agendamento da viagem;

- Permitir o cadastramento de polos e cidades com endereço e informações sobre tutores responsáveis pelo polo;
- Possibilitar o cadastro dos meios de transporte com informações de motoristas;
- Visualizar a quantidade de diárias a receber;
- Possibilitar aos professores a visualização das viagens agendadas;
- Facilitar a administração do transporte; Proporcionar, ao administrador, o gerenciamento rápido e claro do cronograma de viagens;
- Permitir que cada usuário específico verifique o cronograma de acordo com as suas necessidades;
- Possibilitar aos diferentes usuários do cronograma, um sistema dinâmico, efetivo e interativo;
- Possibilitar aos professores a solicitação de cadastro e alteração das atividades letivas no cronograma;
- Viabilizar o controle de patrimônio para os coordenadores (administradores do curso), possibilitando cadastro, consulta alteração de bens patrimoniais e de consumo;
- Facilitará a emissão de pedidos de bens de consumo e, ainda, controlará o recebimento destes pedidos, possibilitando o controle de estoque destes itens;
- Tornar possível o cadastro de notas e faltas dos alunos para fins de controle destes;
- Possibilitar, ao professor, a publicação e atualização do seu plano de ensino todo o semestre;
- Possibilitar a todos os usuários a solicitação de cadastro para o acesso ao sistema;
- Tornar flexível a comunicação entre empresas e alunos; Permitir que qualquer empresa, desde que, previamente, autorizada cadastre vagas para serem preenchidas por alunos;
- Disponibilizará currículos de alunos para que as empresas possam apreciar;

- Permitir que os alunos candidatem-se as vagas disponibilizadas possibilitando ainda, a postagem de mensagens de comunicação entre ele e a empresa;
- Auxiliará ao aluno na formulação de currículos já que o sistema fornecerá um modelo padrão;
- Fornecerá, ao aluno, um controle de frequência.

3.2 Descrição e funcionalidades dos módulos

Neste subcapítulo estarão descritas, detalhadamente, as funções disponíveis no sistema, permitindo o entendimento do potencial funcional de cada módulo.

3.2.1. Módulo de Viagens

O módulo de viagens será criado com a função de fazer o controle das viagens realizadas por professores vinculados ao Curso TSIaD do Instituto Federal de Educação, Ciência e Tecnologia Sul-Rio-Grandense(IFSUL), a serviço da instituição para encontros presenciais. O módulo deverá controlar os dados relativos a cadastramento, autorização das viagens agendadas e, ainda, viabilizará a consulta e a geração de relatórios.

Os requisitos funcionais especificam, detalhadamente, como o módulo de viagens irá funcionar:

a) Cadastrar viagens:

Para cadastrar uma viagem, o administrador, irá consultar o cronograma de aulas presenciais para saber informações como: data, horário, disciplina, professor e pólo onde as aulas irão acontecer. Neste momento, o sistema irá buscar, automaticamente, no cadastro de usuários, as informações pessoais deste professor (a) tais como: nome, rua, número, complemento, bairro, cep e cidade, e-mail, telefone e a instituição vinculada.

Durante o agendamento, o administrador informará o destino, local de partida, local de chegada, data e horário de partida e de retorno, meio de transporte a ser utilizada, e breve descrição do objetivo da viagem. O conjunto destas informações irá resultar num código seqüencial e único, de uma viagem cadastrada, que será apresentada em todos os relatórios gerados.

A partir do referido código da viagem será possível, ao funcionário do setor de transportes, consultar as viagens agendadas e assim, completar e/ou alterar os dados do motorista e transporte a ser utilizado. Ao colocar esta informação, o sistema permitirá a visualização dos dados da viagem. Sendo assim, o referido usuário, poderá protocolar a viagem através de um campo liberado para esta finalidade.

b) Calculo de Diárias

As informações como horário/local de partida e horário/local de chegada permitirão que o sistema calcule, automaticamente, a quantidade de diárias que determinado professor terá para receber. Desta forma, o sistema apresentará, a título de informação, a visualização do número de diárias que um professor terá para receber. Estas diárias serão registradas no mesmo depósito de dados das viagens.

O professor terá acesso a esta funcionalidade visualizando suas diárias e podendo, o sistema não controlará o pagamento das diárias.

c) Alteração das Viagens

O administrador poderá alterar determinada viagem cancelando a atual e/ou re-agendando outra data e horário para a viagem cadastrada quando, esta, ainda não foi realizada. A informação de data será validada no momento em que o administrador entrar na respectiva funcionalidade de alterações.

d) Cancelamentos

- Para cancelar uma viagem o administrador deverá informar o SIAPE do servidor que autorizou o cancelamento e o motivo do cancelamento.

Toda viagem cancelada não será excluída do banco ficando, apenas, terá um registro no campo data de cancelamento que deixará de ser nulo.

e) Re-agendamento

O sistema só permitirá o re-agendamento de uma viagem até uma semana antes da viagem programada. Desta forma, nesta funcionalidade, só será permitido à troca de data, horário de viagem, motorista e transporte.

f) Protocolar viagem

O protocolo é um número de identificação que uma determinada viagem receberá. Este será inserido pelo usuário responsável que tiver a permissão de acesso no seu cadastro para esta finalidade. O número de protocolo será validado não permitindo que o mesmo seja inserido outras vezes. Por outro lado, o administrador, com permissão de agendar viagens, não poderá protocolar viagem, apenas, visualizar se as viagens foram protocoladas.

g) Cadastrar pólo

O administrador irá cadastrar as cidades e seus respectivos polos de ensino com dados de coordenador, tutor presencial, telefone, endereço e instituição vinculada.

h) Cadastrar transporte

O administrador irá armazenar os transportes definindo o meio de transporte pelos tipos: veículo particular, avião, ônibus e veículo da instituição.

Se durante o cadastro for informado o tipo veículo da instituição, deverá ser informado à instituição e o campus que o servidor pertence, placa e motorista que dirige o veículo e, além disso, os respectivos dados do motorista, tais como: CPF e instituição vinculada. Estes cadastros permitirão alterações futuras.

Ao selecionar a opção de ônibus ou veículo particular ou avião não será necessário informar o motorista.

i) Cadastrar usuário

Todas as pessoas que acessarem a página inicial do sistema poderão se cadastrar, porém o acesso só será permitido, após, a aprovação do administrador. Entende-se por usuário professores, administradores, coordenadores, estagiários, alunos e empresas.

Ao preencher a ficha de cadastro o usuário deverá informar seus dados pessoais como: nome, endereço, telefone, CPF, RG, data de nascimento, e-mail, login, senha e o tipo de acesso que deseja ter, podendo optar por professor, aluno, empresa e administrador.

- Acesso professor: O usuário deverá obrigatoriamente informar o seu SIAPE, , instituição e campus que está vinculada.
- Acesso aluno: O usuário deverá informar seu número de matrícula, instituição que está matriculado, módulo e curso que está cursando.
- Acesso empresa: O usuário deverá informar o CNPJ, endereço,site,nome fantasia , telefone e nome do responsável pela empresa que quer se cadastrar.
- Acesso administrador: O usuário deverá informar seu SIAPE, instituição vinculada e o setor que deseja ter acesso.

Após a conclusão do cadastro, o usuário, receberá um e-mail informando que deverá aguardar a aprovação do seu cadastro.

Enquanto isso, o administrador do sistema receberá um e-mail informando que há uma nova solicitação de cadastro de usuários e deverá confirmar os dados passados pelo futuro usuário antes da aprovação do cadastro. Para aprovar um cadastro o administrador deverá informar as permissões de acesso que cada usuário terá.

Após a confirmação do cadastro o usuário receberá um e-mail confirmando seu acesso e uma senha para o primeiro acesso que poderá ser trocada após entrar no sistema.

O sistema terá um usuário padrão chamando de Gerenciador com permissão total de acesso a todos os módulos para fins de manutenção. A partir do acesso do Gerenciador que serão criados os usuários administradores designando-se quais aprovarão ou não cadastros de novos usuários.

O acesso de estagiários será realizado da seguinte forma: Serão cadastrados com a função de administrador, porém este cadastro só poderá ser feito pelo usuário que tem permissão para acessar o módulo em que o estagiário irá trabalhar no momento do cadastro o administrador deve decidir que permissões de acesso serão fornecidas a este estagiário. Seu cadastro fica armazenado no depósito de administradores.

O coordenador também acessa o sistema sendo identificados como administradores.

j) Consultas e Relatórios

O acesso aos relatórios de viagens só será permitido a professores, administradores Estes poderão pesquisar e emitir relatórios de viagens por período, protocolo, código, disciplina, pólo, campus e transporte.

Os professores terão a sua disposição a possibilidade de imprimir a ficha de viagens que conterà seus dados pessoais, data da viagem, destino, diárias a serem pagas, previsão de retorno, transporte utilizado.

3.2.2 Módulo Cronograma

O módulo cronograma irá organizar as atividades acadêmicas . O cronograma gerencia as informações de datas e horários de eventos. Estes eventos são distribuídos de forma a facilitar a interpretação de cada usuário. Os requisitos funcionais deste módulo serão:

a) Incluir evento no cronograma

Baseado nas informações de cronograma semestral do modulo o administrador irá cadastrar um evento.Os professores terão a permissão de cadastrar eventos no cronograma. No evento deverá ser estabelecido o tipo de evento, data de inicio, data de termino, professor responsável, disciplina, módulo e pólo.

b) Cancelamento de um evento

Quando cancelado um evento deve-se justificar o motivo Apenas o administrador possui o direito de cancelar um evento. O professor pode apenas solicitar o cancelamento ou encaminhar um pedido de re-agendamento do evento ao administrador. O aluno não tem permissão para realizar esse procedimento. Caso cancelado um evento, o cancelamento do mesmo deve ser imediatamente comunicado aos participantes do evento, também sendo automaticamente enviado à tela do cronograma um relatório do cancelamento.

c) Re-agendamento

O processo de re-agendamento ocorre de modo semelhante ao processo de cancelamento, sendo realizado apenas pelo administrador e solicitado pelo

professor o que difere os dois processos é que no re-agendamento a data do evento é alterada no ato.

d) Cadastrar informações no cronograma

No cronograma deverão ser cadastrados inicialmente os módulos e as disciplinas que constituem a grade de cada semestre. O administrador é encarregado de cadastrar estas informações e do mesmo modo alterar e excluir estes dados.

Segue a descrição de cada um dos depósitos necessários para o cadastramento do cronograma:

- **Módulo:** O administrador irá cadastrar os módulos de cada pólo no cronograma. Os módulos deverão possuir especificações tais quais: código do módulo, data de início e data estimada de término, pólo referente.
- **Disciplina:** O administrador irá cadastrar todas as disciplinas referentes ao curso. Estas disciplinas deverão possuir especificações tais quais: código, nome da disciplina, sigla e carga horária.
- **Disciplina-Módulo:** Haverá no banco de dados uma tabela que relaciona os módulos e as disciplinas referentes a cada módulo, pois uma disciplina estar em vários módulos assim como um módulo tem várias disciplinas

3.2.3 Módulo Patrimônio

O módulo do patrimônio tem como principal objetivo gerenciar e automatizar o processo de administração do patrimônio do TSIaD. Os bens patrimoniais e bens de consumo diferem, já que um é definitivo e o outro é consumido, os bens patrimoniais nunca são excluídos da base de dados do banco, isso é, o N° cadastrado para tal patrimônio não é reutilizado mesmo ele tendo sido extraviado ou baixado. O patrimônio em geral será classificado por dois tipos: 1(patrimônio) e 2(bens de consumo), haverá também uma classificação para identificar o patrimônio

cadastrado um exemplo de classificação “móveis”.O módulo deverá controlar os dados relativos a cadastramento, consulta, e alteração das informações patrimoniais e de bens de consumo controlados pelos administradores. Ainda, será permitido a estes, a formulação de pedido de novos bens de consumo, patrimônio e troca de responsável pelo bem. Os requisitos funcionais deste módulo serão:

a) Cadastrar patrimônio

Para cadastrar o patrimônio o usuário deverá preencher um formulário com informações sobre o bem patrimonial, tais como: descrição, sala, valor, responsável, número do patrimônio, setor, tipo, classificação, data de compra.

b) Alteração do patrimônio

Para alterar o patrimônio o usuário deverá consultar a relação do patrimonio com o número do bem patrimonial, e assim serão listados todos os itens, e para selecionar determinado produto, o usuário deverá clicar no ícone de alterar e trocar as informações desejadas.

Na tela de alteração do patrimonio, o usuário poderá alterar todas as informações sobre os itens cadastrados, assim como deverá dar baixa no estoque de produtos que já foram usados ou inutilizados quando necessário.

c) Emitir pedido

O sistema irá gerar um pedido do material em falta, e o usuário poderá imprimir para que o pedido seja encaminhado ao setor responsável pela compra, Deste modo a compra de bens de consumo como: canetas, borrachas, lápis, apagadores e outros serão agilizados.

d) Avisar sobre falta de bens de consumo

O sistema controlará a quantia disponível de cada bem de consumo cadastrado, e assim que for feita uma consulta nos bens de consumo o sistema irá identificar a falta de um bem então será enviado ao usuário um alerta informando a falta do item.

e) Consultar patrimônio

Os usuários do módulo do patrimônio terão ao seu dispor uma tela para consultas, onde deverá ser selecionada a área de pesquisa como: número do patrimônio, sala, valor, data da compra, descrição, e responsável.

3.2.4 Módulo de Professores

O Módulo de Professores terá a função de organizar as aulas, visualizar viagens agendadas e consultar cronogramas. Neste módulo o professor poderá inserir informações destinadas aos alunos, como notas e faltas, e plano de ensino. Os requisitos funcionais deste módulo são:

a) Registrar notas

O sistema fornecerá ao professor a possibilidade de inserir as notas das avaliações dos alunos no sistema permitindo que os alunos tenham acesso aos resultados destas avaliações ao mesmo tempo em que o professor executa inserção da informação.

b) Registrar frequencia

O professor poderá inserir a frequência dos alunos e o total de aulas que devem ser ministradas, deste modo o aluno poderá saber como esta o seu percentual de frequência.

c) Registrar avaliações

O professor poderá fazer o registro de datas das avaliações Inserindo informações no cronograma.

d) Cadastrar plano de ensino

O professor poderá cadastrar seu plano de ensino semestralmente.

3.2.5 Módulo Empresas

O módulo Empresas irá proporcionar uma interação entre alunos e empresas, pois as empresas poderão cadastrar vagas de emprego disponíveis, verificar currículos previamente cadastrados por alunos.Os requisitos funcionais serão:

a) Cadastrar vagas

As empresas poderão oferecer aos alunos vagas de emprego.Para cadastrar uma vaga a empresa deverá informar :função,quantidade e descrição .

b) Alterar vagas

As empresas poderão alterar as vagas previamente cadastradas por elas, a principal alteração será na quantidade de vagas disponíveis, podendo assim, retirar

a disponibilidade dela já que as vagas não serão excluídas do sistema, e sim armazenadas para ter a possibilidade de reativá-la posteriormente.

c) Consultar aluno-vagas

Os alunos poderão consultar as vagas de emprego disponíveis previamente adicionadas pelas empresas através da opção “consultar vagas” presente no sistema.

d) Consultar Empresa-Aluno

A empresa poderá conhecer o perfil profissional do aluno. Através da consulta de currículos serão divididos nas categorias: visualizar currículo geral, por curso, por turno e por vaga.

3.2.6 Módulo Alunos

O módulo aluno tem o objetivo de: consultar cronograma, consultar vagas, cadastrar currículo e consultar desempenho. O requisitos funcionais são:

a) Cadastro de Aluno

É o cadastramento de usuário do tipo aluno, que tem credenciais de acesso restritivo ao sistema. Os dados referentes a esses alunos vão para uma base de dados e posteriormente podem ser consultados, dispostos em um perfil e acrescentados automaticamente ao currículo. Tal cadastro é disponibilizado ao usuário tão logo ele acessa o sistema pela primeira vez, sob a forma de um formulário, contendo nome, sobrenome, CPF, RG, número da matrícula, e-mail, módulo que cursa, data de nascimento, sexo e permite o envio de foto. O cadastro pode ser alterado

posteriormente quando for mais conveniente.

b) Consultar Cronograma

Os alunos poderão consultar o cronograma de eventos do polo e do modulo que participam.

c) Consultar Vagas

Os alunos poderão consultar as vagas disponibilizadas pela empresas.

d) Cadastrar Currículo

O aluno poderá cadastrar seu currículo inserindo informações como: formação acadêmica, atividades complementares, experiências profissionais, dados adicionais, objetivo, disponibilidade e qualificações

e) Consultar Desempenho

A consulta os dados de desempenho, é importante para que se tenha uma parcial do andamento disciplinar do aluno no curso. Notas, frequência, resultados de atividades e nota final, podem ser consultadas nessa área. Esses dados são previamente inseridos pelos professores.

3.3 Usuários Envolvidos

O **administrador** será o principal usuário do sistema, dependendo de suas permissões ele poderá fazer qualquer acesso e ainda aprovar novos usuários

Os **professores** terão acesso aos módulos: viagem, cronograma, alunos, com restrições de acesso ao patrimônio e empresas.

Os **alunos** terão acesso aos módulos: cronograma (restrição específica para somente visualizar o cronograma), ao seu próprio ambiente no módulo aluno e a consulta de vagas disponibilizadas no módulo das empresas.

As **empresas** terão acesso ao módulo de empresas para cadastrar vagas e ainda consultar currículos de alunos.

3.3 Resultados Esperados

Espera-se excelentes resultados com a implementação deste sistema ágil, seguro e eficiente e que possa trazer bons frutos para seus futuros usuários, e assim tornando a administração dos mais variados módulos mais fáceis.

4. DIAGRAMA DE FLUXO DE DADOS

Dentre as técnicas estruturadas para análise estruturada, o DFD é uma das mais utilizadas, pois fornece a visão estruturada das funções do sistema que permite ao desenvolvedor “imaginar” um sistema como uma rede de processos funcionais interligados. Neste sentido, este capítulo apresenta a modelagem do DFD's do sistema Gère seguindo as definições da teoria de Yourdon(1990).

Para a construção dos DFDs, utilizou-se a ferramenta *CASE*¹ (Computer-Aided Software Engineering) “Case Studio”. A ferramenta *CASE*, para Bortolin (2008), tem como objetivo automatizar atividades manuais de pré-codificação, facilitando as correções e a documentação do sistema.

Sendo assim, o Case Studio é uma ferramenta que permite que se faça um mapa dos bancos de dados e seus vínculos, de uma forma muito intuitiva. Durante a utilização do Case Studio, percebeu-se que sua interface é arrojada, espaçosa, permitindo fazer a engenharia reversa de banco de dados existentes, além de ser compatível com todos os grandes formatos de bancos de dados.

Um DFD pode ser representado em vários níveis, dependendo da complexidade e do detalhamento do sistema. Assim, a seguir, apresentaremos os níveis de representação do DFD: Diagrama de Contexto, Diagrama de nível um e Diagrama de nível dois.

4.1 Diagrama de contexto

O diagrama de contexto, conforme figura 1, é um diagrama de fluxos de dados, que representa um processo do sistema inteiro. Esta modelagem é cercada por entidades externas que fazem a interação com o sistema por meio de fluxo de dados de entrada e saída e as origens e destinos destes dados.

¹ Case significa Engenharia de Software auxiliada por computador. Esta tecnologia é utilizada para facilitar o trabalho dos analistas.

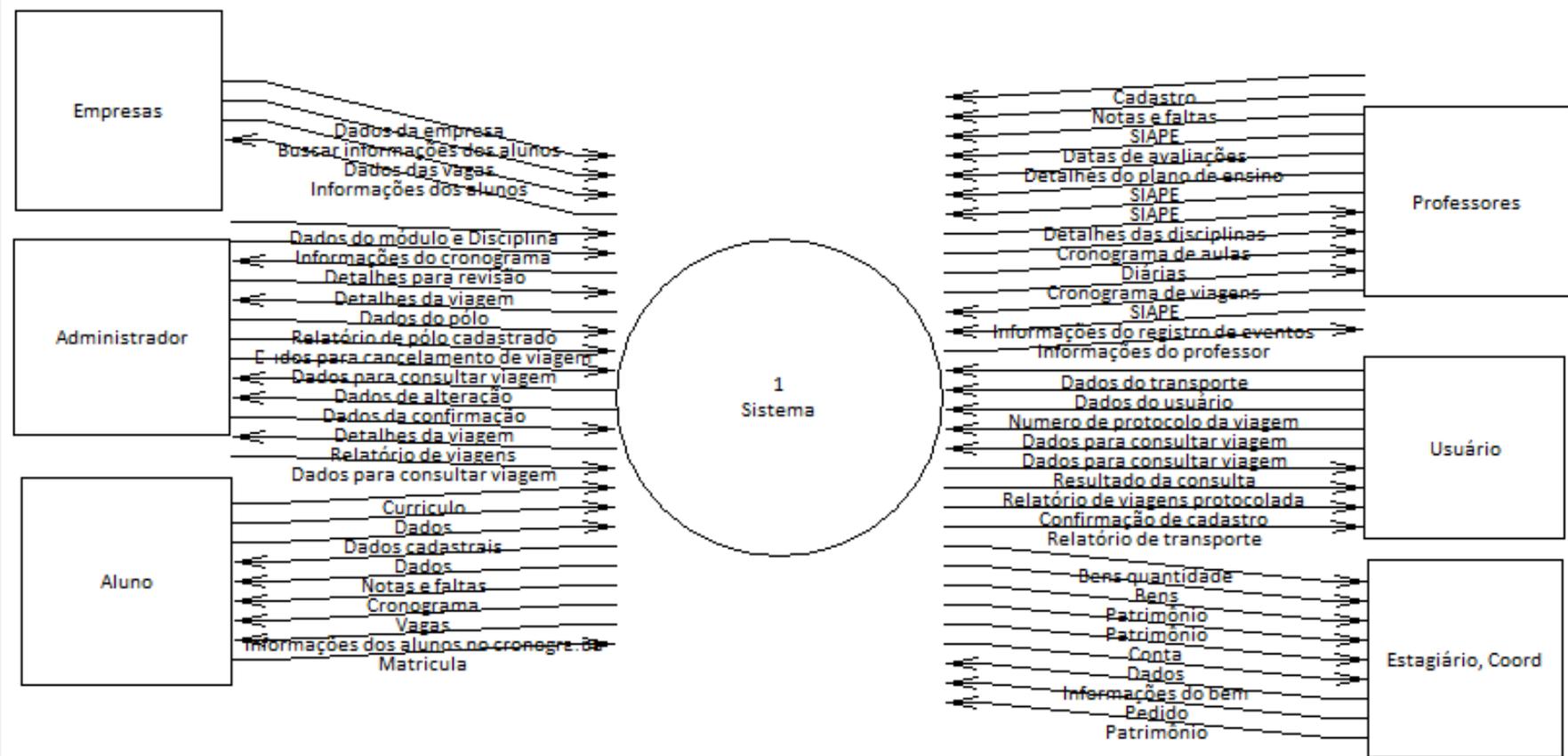


Figura 1: Diagrama de Contexto do Sistema Gère

4.2 Diagramas de nível um

Em seguida, é feita uma decomposição do Diagrama de Contexto, mantendo as mesmas entradas e saídas e as mesmas origens e destinos de cada dado. Neste nível do diagrama, acrescentam-se os processos, que operam sobre os fluxos de dados e os depósitos de dados, que representam o armazenamento de dados.

Neste momento, é possível ter uma visão detalhada das principais funcionalidades que o Sistema Gêre irá apresentar. Assim, a seguir, iremos apresentar o diagrama de todos os módulos que compõem o sistema:

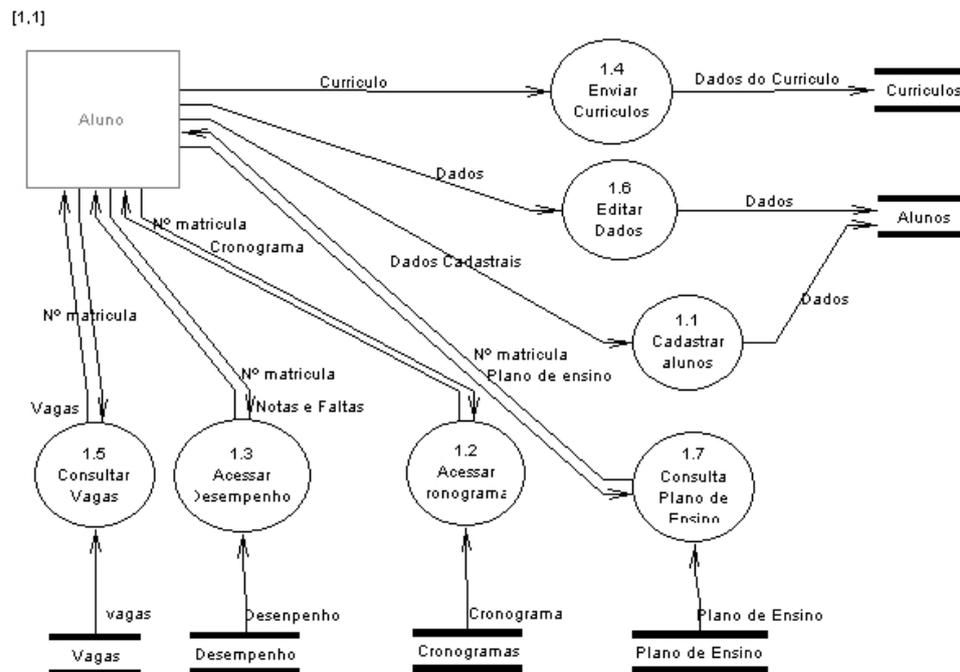


Figura 2: Diagrama de nível 1 do Módulo Aluno

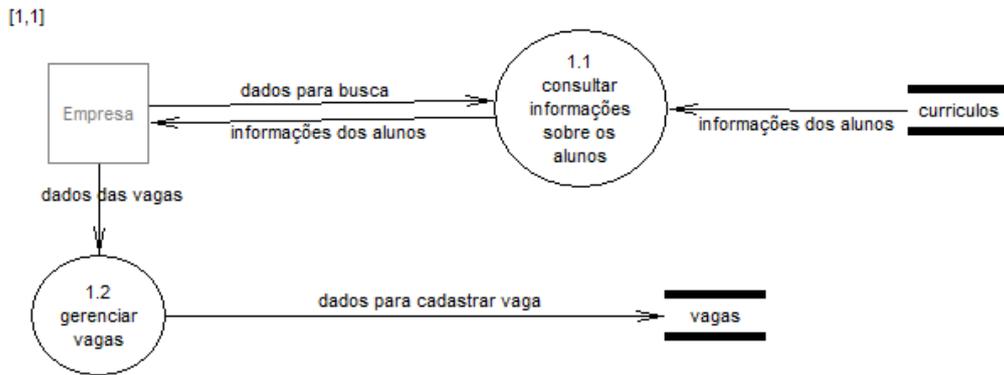


Figura 3: Diagrama de nível 1 do Módulo Empresa

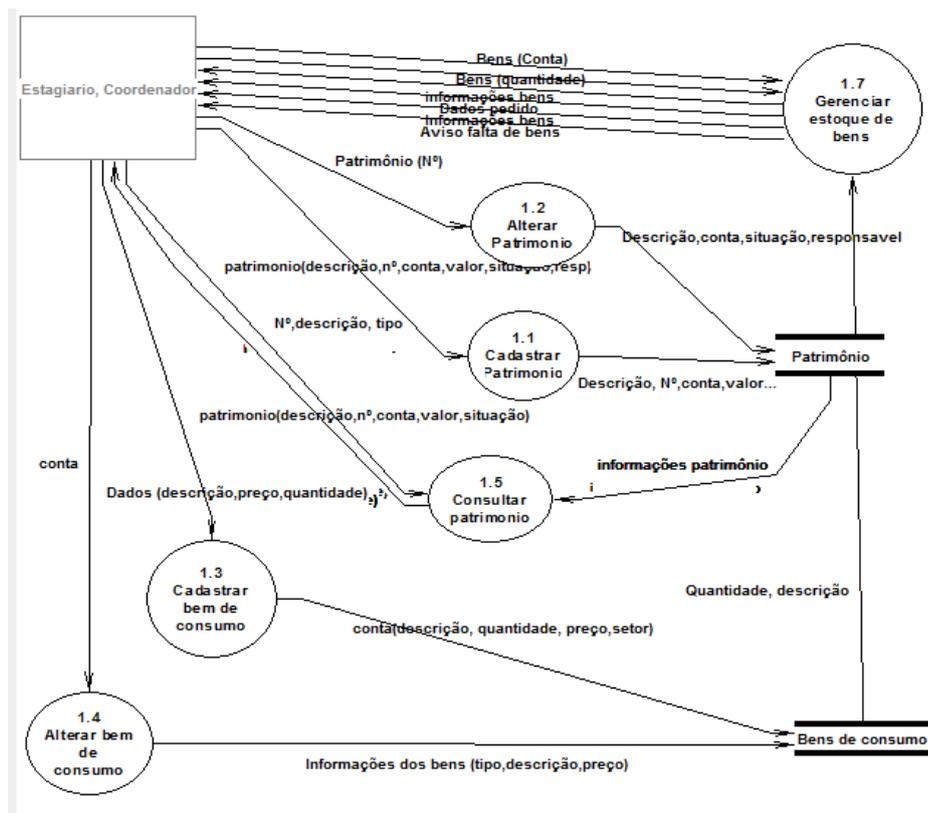


Figura 4: Diagrama de nível 1 do Módulo Patrimônio

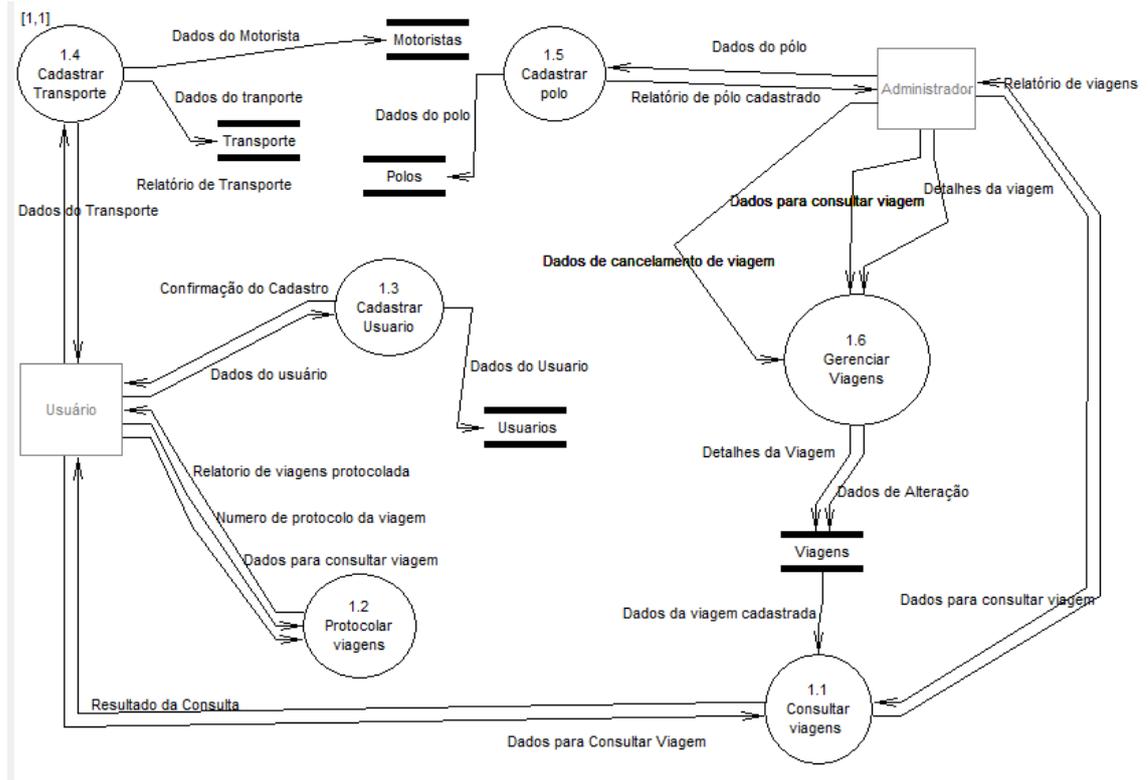


Figura 5: Diagrama de nível 1 do Módulo Viagens

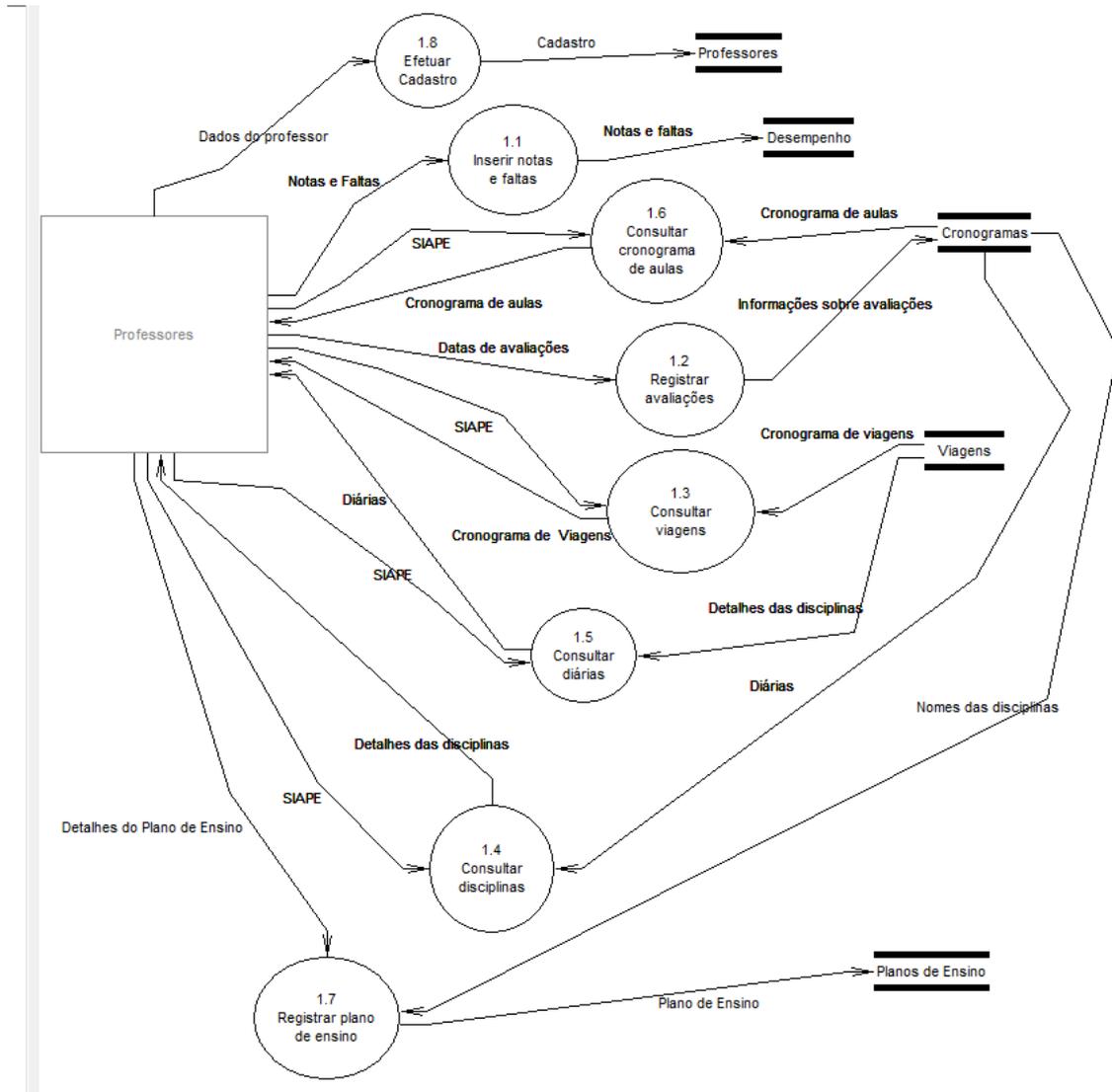


Figura 6: Diagrama de nível 1 do Módulo Professor

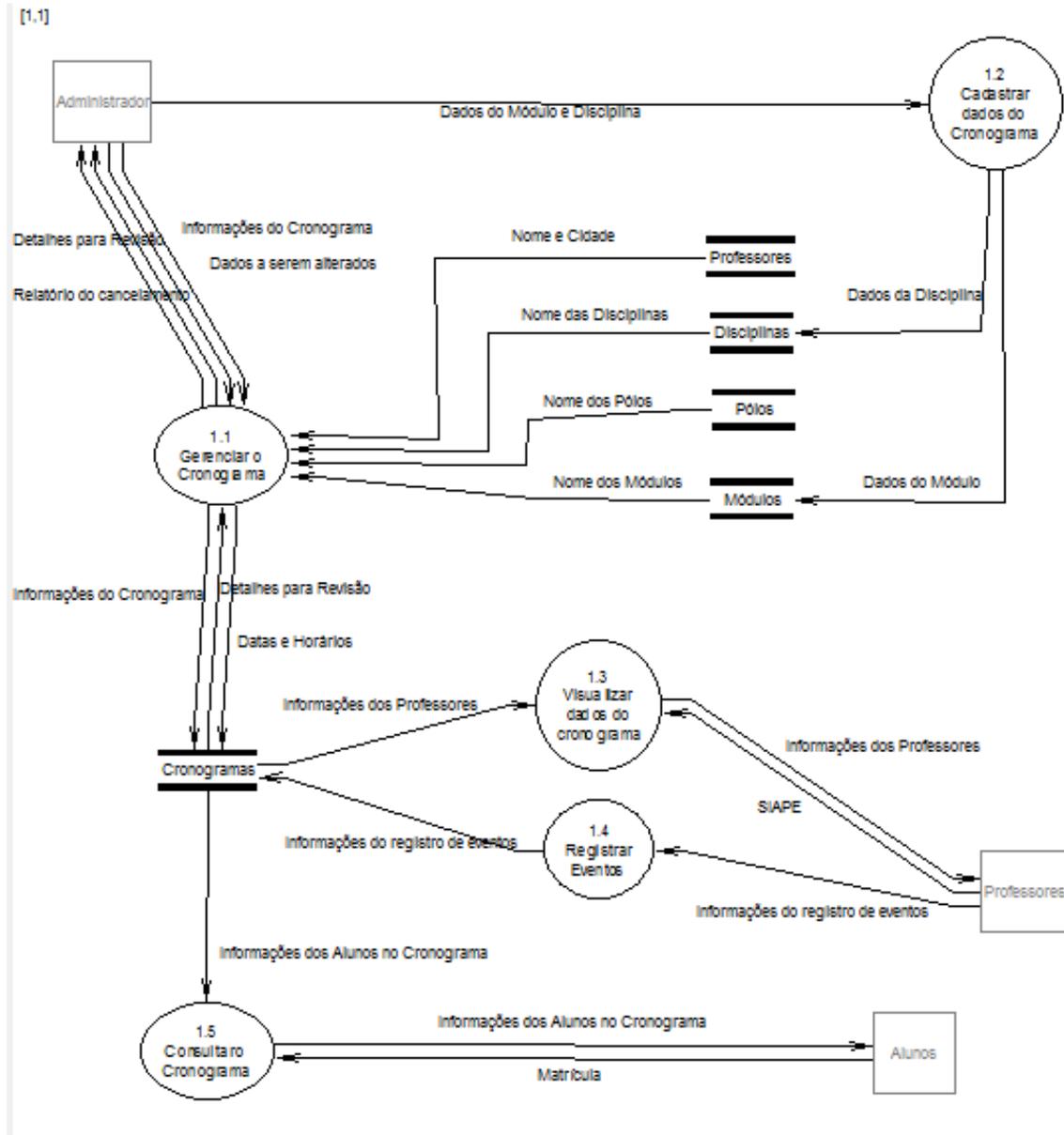


Figura 7: Diagrama de nível 1 do Módulo Cronograma

4.3 Diagramas de nível dois

O diagrama de nível dois representa a visão do interior dos processos do nível um, explicando melhor cada uma de suas funcionalidades e, detalhando mais, especificadamente, o que acontece entre cada processo de nível dois. As figuras abaixo demonstram o desmembramento de processos do nível 1 para o nível 2 de alguns módulos.

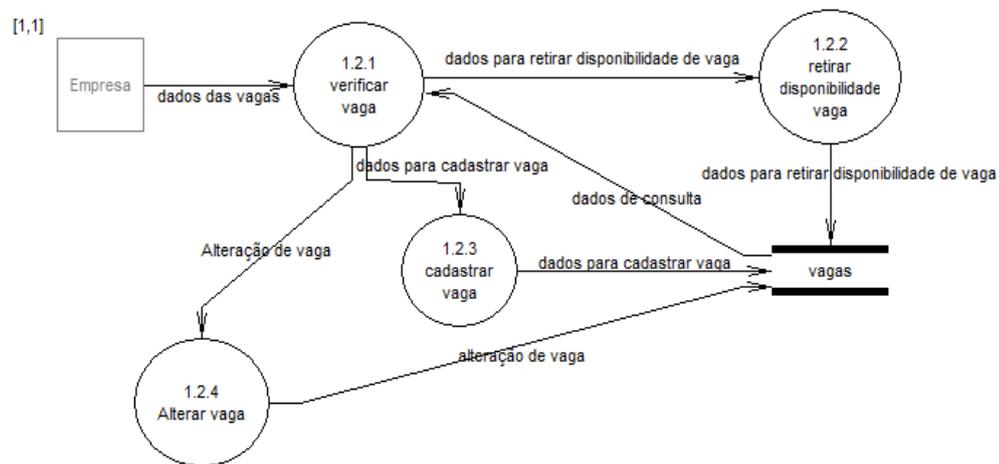


Figura 8: Diagrama de nível 2 gerenciar vagas do Módulo Empresa

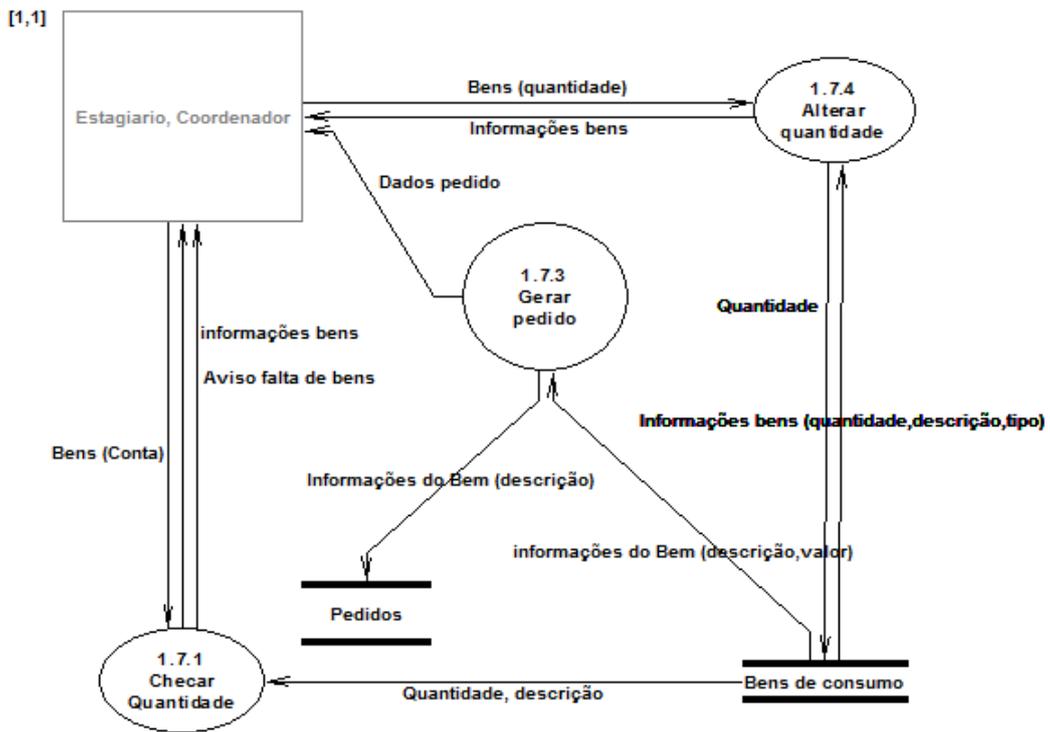


Figura 9: Diagrama de nível 2 gerenciar patrimônio do Módulo Patrimônio

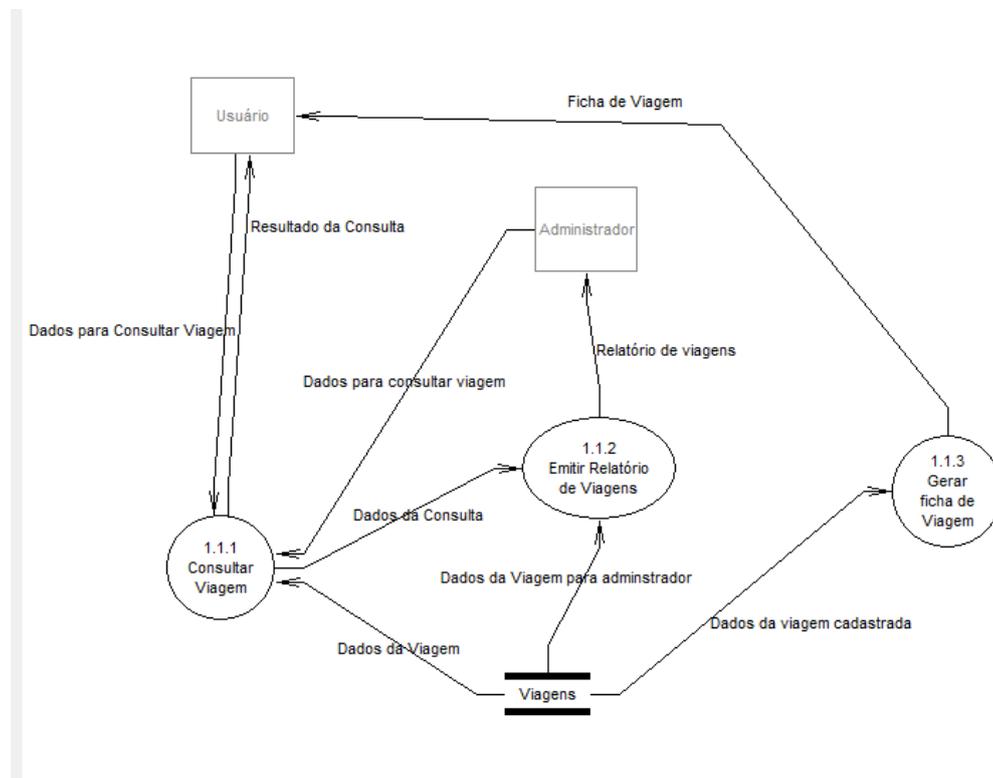


Figura 10: Diagrama de nível 2 consultar viagens do Módulo Viagens

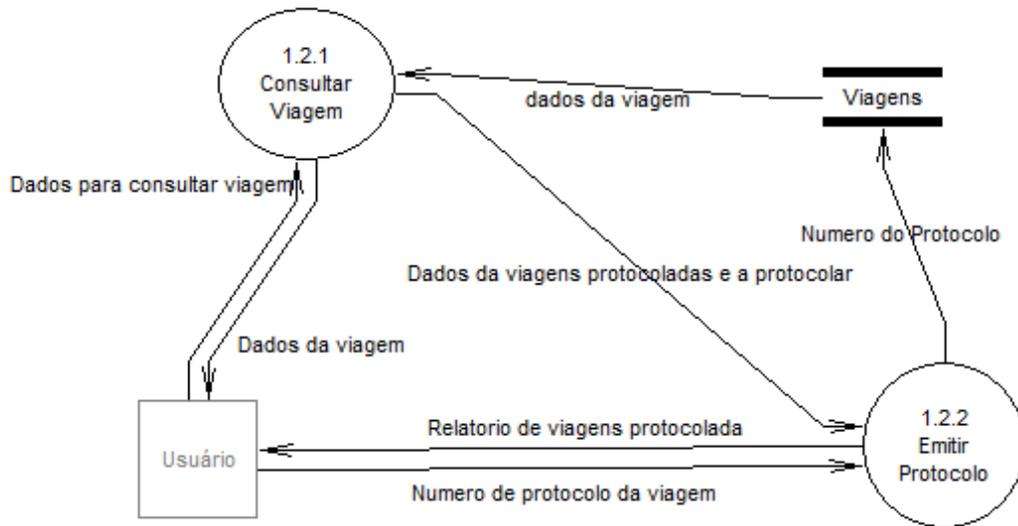


Figura 11: Diagrama de nível 2 protocolar viagens do Módulo Viagens

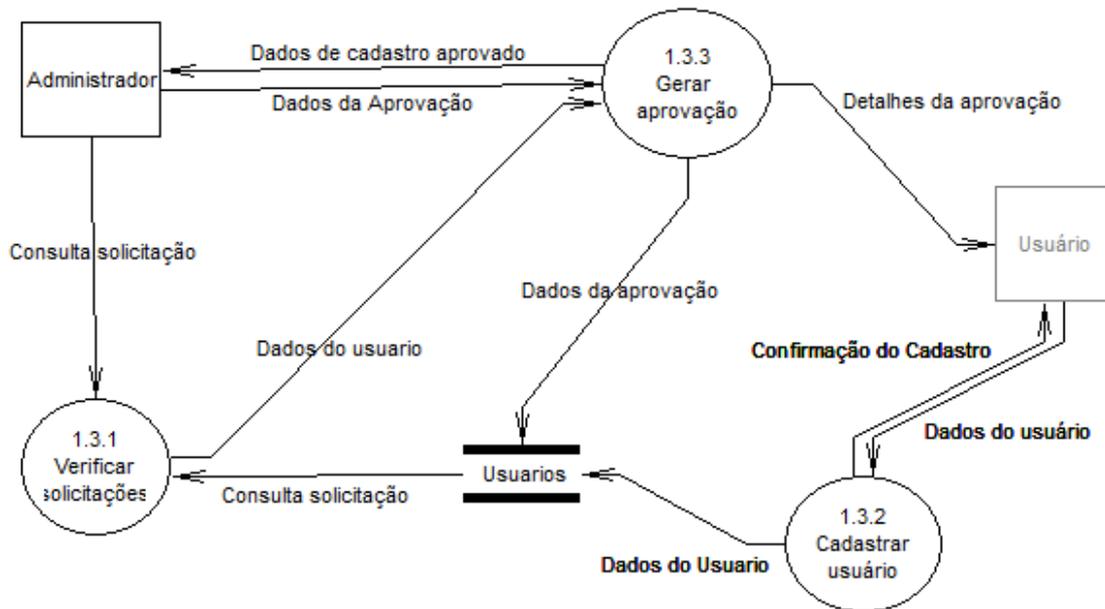


Figura 12: Diagrama de nível 2 cadastro de usuarios do Módulo Viagens

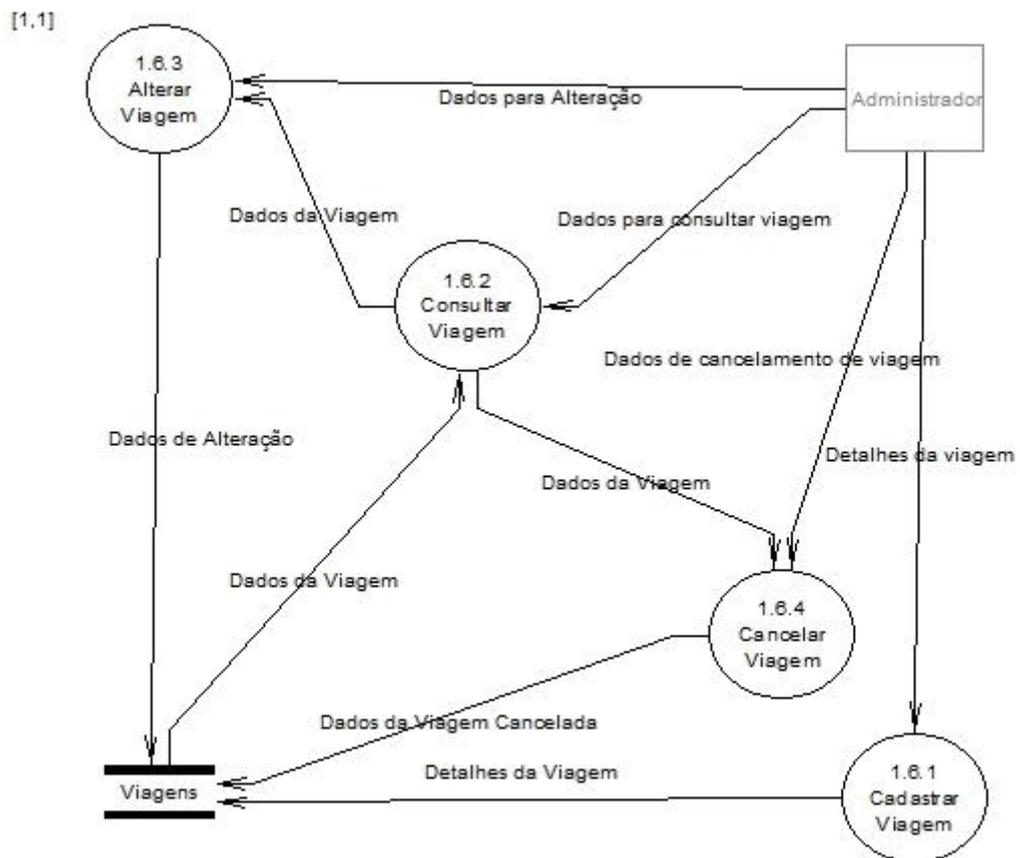


Figura 13: Diagrama de nível 2 gerenciar viagem do Módulo Viagens

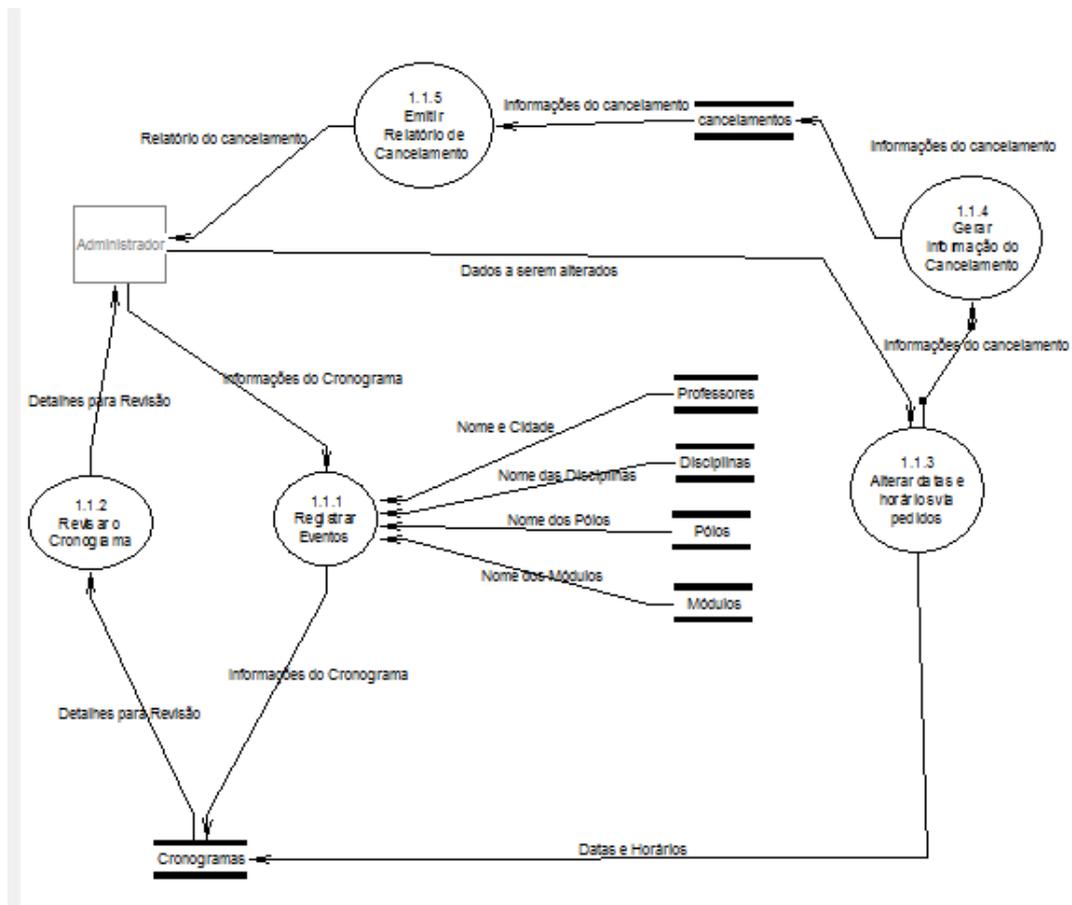


Figura 14: Diagrama de nível 2 gerenciar cronograma do Módulo Cronograma

A representação dos diagramas em níveis permite um fracionamento da complexidade de sistemas. Assim sendo, para completar a modelagem de análise, o próximo subcapítulo apresentará a expansão da compreensão dos dados do DFD a partir da ferramenta “Dicionário de Dados (DD)”.

5. DICIONÁRIO DE DADOS

O Dicionário de Dados, segundo Rezende (2005) é uma ferramenta de Análise Estruturada para descrever, detalhadamente, os conteúdos dos processos, dos fluxos de dados, dos depósitos de dados e das entidades externas do sistema. Estes dados são representados no DD da mesma forma como são referidas no DFD.

O Dicionário de Dados é uma ferramenta, extremamente, útil para definir os dados, pois, à medida que, se faz a sua edição do DD podemos perceber as redundâncias, ambigüidades e contradições nos dados apresentação pelo DFD. Durante a modelagem, esta ferramenta, proporcionou-nos a descrição compreensível e padronizada de todos os dados envolvidos no Sistema Gère.

Para a descrição do DD utilizamos a notação proposta por YOURDON (1990), que pode ser observada na tabela abaixo:

Tabela 1: Notação utilizada no Dicionário de Dados

Notação	Significado
=	é composto de
+	E
()	Opcional
{ }	iteração ou repetição
[]	alternativas possíveis de valores
**	Comentário
@	identificador de um depósito de dados
	ou (usado exclusivamente para separar as alternativas dentro de [])

A construção do DD do Sistema Gère (veja nas próximas tabelas) surgiu da descrição do DFD de todos os fluxos de dados, depósitos de dados, entidades e processos. Os itens de dados que fazem parte de outros itens também foram definidos num padrão em todo documento. Os depósitos de dados foram descritos utilizando-se a construção de iteração ou repetição. Os itens atômicos foram especificados pelas combinações de dados tipo estrutura, tais como, “número” ([0-9]), “caracter” ([A-Z | 0-9 | | , | . | - | ']) ou “data” (dia + mês + ano), ou ainda, por faixas ou alternativas de valores.

Tabela 2: Entidade do Módulo Viagens

Entidade Externa	Usuários
Descrição Detalhada	Usuários são todas as pessoas que interagem com o sistema, como professores, administradores, empresas e alunos. Inclui-se ainda pessoas ligadas a reitoria e ao setor de transportes da instituição.
Fluxos de Entrada	Confirmação de cadastro , relatórios de viagens protocoladas , resultados de consultas, relatórios de transportes.
Fluxos de Saída	Dados do transporte, dados do motorista , números de protocolo de viagens , dados para consultar viagens.
Entidade Externa	Administrador
Descrição Detalhada	Usuários responsáveis pela administração do sistema com privilégios de acesso diferenciado aos demais usuários. Pessoas que podem cadastrar viagens ou alterar viagens cadastradas ou consultar viagens agendadas ou cadastrar os pólos de ensino .
Fluxos de Entrada	Relatórios de pólos cadastrados, dados alterados , dados da confirmação da viagem cadastrada, relatórios de viagens.
Fluxos de Saída	Dados do pólo, dados de cancelamento da viagem, dados para consulta, detalhes da viagem, dados da consulta.

Tabela 3: Processos do Módulo Viagens

Processo	Consultar viagens agendadas
Descrição Detalhada	Realiza tarefa de consultar no depósito de dados as viagens cadastradas baseado nos dados inseridos pelo administrador ou pelo usuário e fornece informações para geração de relatórios.
Fluxos de entrada	Dados para consultar viagem.
Fluxos de Saída	Dados da consulta, resultados da consulta.
Processo	Emitir relatório de viagens
Descrição Detalhada	Realiza a geração de relatórios em formatação padrão para que possa ser visualizado e impresso pelos usuários e administradores.
Fluxos de entrada	Dados da consulta, dados da viagem para o administrador.
Fluxos de Saída	Relatório de viagens.
Processo	Gerar ficha de viagem
Descrição Detalhada	Gerar ficha impressa com os detalhes da viagem cadastrada .
Fluxos de entrada	Dados da viagem cadastrada.
Fluxos de Saída	Ficha de viagem.
Processo	Gravar viagem
Descrição Detalhada	Realiza a gravação no depósito de viagens as viagens agendadas baseado nas informações inseridas pelo administrador e gera a confirmação da viagem cadastrada e gera informações para o cálculo de diárias.
Fluxos de entrada	Dados da viagem cadastrada, detalhes da viagem
Fluxos de Saída	Dados da viagem, destino, data e hora de saída e retorno, dados da confirmação da viagem cadastrada.
Processo	Calcular diárias

Descrição Detalhada	Baseado no período entre saída e chegada e destino o sistema irá calcular a quantidade de diárias que o professor irá receber.
Fluxos de entrada	Destino, data e hora de saída e retorno.
Fluxos de Saída	Resultado do cálculo.
Processo	Consultar cronograma
Descrição Detalhada	Consulta no depósito de dados cronograma as aulas presenciais e informações sobre data, hora, dia e disciplina que será ministrada as aulas para que se possa agendar a viagem baseada nestas informações.
Fluxos de entrada	Dados da aula(tipo, data, e disciplina), dados da consulta.
Fluxos de Saída	Dados do cronograma.
Processo	Protocolar viagens
Descrição Detalhada	Realiza a consulta das viagens cadastradas para que o usuário possa protocolar viagens realizadas e emitir relatórios de viagens protocoladas.
Fluxos de entrada	Dados para pesquisar viagem, número do protocolo.
Fluxos de Saída	Relatório de viagens protocoladas.
Processo	Emitir protocolo
Descrição Detalhada	Realiza a geração de relatórios em formatação padrão para que possa ser visualizado e impresso pelos usuários e administradores com informações sobre a viagem protocolada.
Fluxos de entrada	Número do protocolo, dados da viagem protocolada, a protocolar .
Fluxos de Saída	Relatório de viagens protocoladas, número do protocolo.
Processo	Alterar viagem
Descrição Detalhada	Realiza o reagendamento, a consulta e o cancelamento das viagens cadastradas.
Fluxos de entrada	Dados da viagem cadastrada, dados do cancelamento da viagem, dados da viagem para reagendar.
Fluxos de Saída	Dados alterados .
Processo	Reagendar viagem
Descrição Detalhada	Realiza a alteração das informações das viagens cadastradas para fins de reagendamento.
Fluxos de entrada	Dados da viagem para reagendar, dados da viagem.
Fluxos de Saída	Dados alterados.
Processo	Cancelamento de viagem
Descrição Detalhada	Realiza o cancelamento da viagem com informações gravadas no depósito de viagens.
Fluxos de entrada	Dados do cancelamento da viagem, dados da viagem.
Fluxos de Saída	Dados da confirmação de cancelamento.
Processo	Cadastrar usuário
Descrição Detalhada	Recebe os dados cadastrais fornecidos pelo usuário e os detalhes da aprovação do cadastro fornecidos pelo administrador.
Fluxos de entrada	Dados para cadastro.
Fluxos de Saída	Confirmação do cadastro.

Processo	Verificar solicitações
Descrição Detalhada	Verifica se há solicitações de novos cadastro de usuário. Caso afirmativo, gera o processo de aprovação.
Fluxos de entrada	Consulta solicitações.
Fluxos de Saída	Dados do usuário.
Processo	Gerar aprovação
Descrição Detalhada	Realiza a gravação do depósito de dados das permissões de acesso do novo usuário e envia ao usuário uma mensagem de confirmação do cadastro realizado.
Fluxos de entrada	Dados da aprovação, dados do usuário.
Fluxos de Saída	Dados do cadastro aprovado, dados do usuário(para aprovação) e detalhes da aprovação.
Processo	Cadastrar transporte
Descrição Detalhada	Realiza o cadastro dos meios de transporte utilizados nas viagens e informações sobre motoristas.
Fluxos de entrada	Dados do transporte, Dados do Motorista
Fluxos de Saída	Relatório de transporte.
Processo	Cadastrar pólo
Descrição Detalhada	Realiza o cadastro dos pólos de ensino de cada instituição.
Fluxos de entrada	Dados do pólo.
Fluxos de Saída	Relatório de pólo cadastrado, dados do pólo.
Processo	Gerenciar viagens
Descrição Detalhada	Realiza a organização das informações das viagens cadastradas.
Fluxos de entrada	Dados do cancelamento de viagem, dados para consulta de viagens, detalhes da viagem.
Fluxos de Saída	Dados da alteração, detalhes da viagem

Tabela 4: Fluxo de Dados do Módulo Viagens

Fluxo de Dados	Confirmação do Cadastro
Descrição detalhada	Mensagem enviada ao usuário com os detalhes da aprovação do cadastro do usuário como as suas permissões de acesso.
Origem	Cadastrar usuário (processo).
Destino	Usuário (entidade).
Estrutura de dados	Nome do Usuário= nome+sobrenome
	Email={caracter_válido}
	Tipo Acesso= [Professor aluno administrador empresa]
	CPF=numero* 11 dígitos*
	Data aprovação=dia+mês+ano
	Login=letra+numero
Fluxo de Dados	Dados do Motorista
Descrição detalhada	Transporta dados sobre o motorista para o cadastro no depósito de motoristas.

Origem	Usuário (entidade) e Cadastrar Transporte(processo).
Destino	motoristas (depósito).
Estrutura de dados	Instituição Vinculada =nome
	CNH =letra+numero *carteira nacional de habilitação
	Endereço= rua+complemento+cidade+bairro+numero+estado+CEP
	Nome do Motorista =nome+sobrenome
	Telefone = código de área + prefixo + número de acesso
	CPF=numero 11 dígitos*
Fluxo de Dados	Dados do Transporte
Descrição detalhada	Transporta dados do transporte para o depósito de transportes.
Origem	Usuário (entidade) e Cadastrar Transporte(processo).
Destino	transportes(depósito).
Estrutura de dados	Código do motorista =numero *código do motorista proveniente do depósito de motoristas*
	Placa=letra+numero
	Tipo do Veículo=[Onibus Carro Particular Avião Carro da instituição]
	Nome da Instituição =nome
Fluxo de Dados	Relatório de Transporte
Descrição detalhada	Relatório gerado pelo sistema constando a relação de transportes cadastrados e motoristas que dirigem carros de instituições.
Origem	Cadastro de Transporte(processo)
Destino	Usuário(entidade)
Estrutura de dados	Código do transporte=numero*número sequencial gerado na hora do cadastro*
	Placa=letra+numero
	Tipo do Veículo=[Onibus Carro Particular Avião Carro da instituição]
	Nome da Instituição =nome
	Nome do Motorista =nome+sobrenome
	CNH =numero+letras*carteira nacional de habilitação
	Nome do Motorista =caracter_valido
	Telefone =código de área + prefixo + número de acesso
CPF=numero 11 dígitos*	
Fluxo de Dados	Dados do Pólo
Descrição detalhada	Transporta dados do pólo de ensino cadastrado
Origem	Administrador (entidade) e cadastrar polo (Processo)
Destino	Pólos (depósito)
Estrutura de dados	Endereço= rua+complemento+cidade+bairro+numero+estado+CEP
	Nome do Pólo =nome
	Instituição Vinculada =nome
	Tutor =nome+sobrenome
	Coordenador =nome+sobrenome
	Telefone = código de área + prefixo + número de acesso
Observações=caracter_valido	
Fluxo de Dados	Relatório de Polo Cadastrado
Descrição detalhada	Relatório gerado pelo sistema constando a relação de Polos cadastrados.

Origem	Cadastrar Pólo (processo)
Destino	Administrador(entidade)
Estrutura de dados	Código da pólo= numero *código sequencial gerado na hora do cadastro*
	Endereço= rua+complemento+cidade+bairro+numero+estado+CEP
	Nome do Pólo =nome
	Instituição Vinculada =nome
	Tutor =nome+sobrenome
	Coordenador =nome+sobrenome
	Data Cadastro=dia+mês+ano
	Data inativo=dia+mês+ano
	Telefone = código de área + prefixo + número de acesso
	Observações=caracter_valido
Fluxo de Dados	Dados do usuário
Descrição detalhada	Transporta os dados pessoais do usuario que deseja se cadastrar no sistema.
Origem	Usuário (entidade)
Destino	Usuários (depósito)
Estrutura de dados	Nome do Usuário= nome+sobrenome
	Telefone = código de área + prefixo + número de acesso*
	Email=caracter_valido
	Tipo Acesso= [Professor aluno administrador empresa]
	CPF=numero* 11 dígitos*
	Setor=caracter_valido*opcional preenchido quando tratar-se de cadastro de administradores*
	SIAPE do professor=numero
	Matrícula=numero
	CNPJ=numero* 14 dígitos*
	Senha=letra+numero
	Login=letra+numero
	Razão Social=nome
	Ramo de Atividade=nome
Fluxo de Dados	Número de Protocolo
Descrição detalhada	Número fornecido pelo usuario autorizado confirmando que a viagem foi realizada.
Origem	Usuário (entidade)
Destino	Protocolar Viagem (processo) e Viagem (deposito)
Estrutura de dados	Protocolo=numero
Fluxo de Dados	Relatório de Viagens Protocoladas
Descrição detalhada	Relatório gerado pelo sistema constando todas as informações sobre a viagem que foi protocolada
Origem	Protocolar viagem (processo)
Destino	Usuário (entidade)
Estrutura de dados	Código da viagem=numero*código sequencial gerado na hora do cadastro*
	Data Partida=dia+mês+ano

	Data Chegada=dia+mês+ano
	Meio de Transporte=numero*proveniente do cadastro de motoristas*
	Hora de saída=hora+minutos
	Hora de chegada=hora+minutos
	Total de Diárias=numero
	Protocolo=numero
	SIAPE do professor=numero
	Local Partida= rua+complemento+cidade+bairro+numero+estado+CEP* não obrigatório preenchido somente quando o ponto de partida não seja o pólo de origem*
	Local Destino= rua+complemento+cidade+bairro+numero+estado+CEP* não obrigatório preenchido somente quando o ponto de chegada não seja o pólo de destino*
	Origem=numero *pólo de onde o professor irá partir*
	Destino=numero*pólo onde aula será ministrada*
Fluxo de Dados	Dados para consultar Viagens
Descrição detalhada	Armazena todas as informações que a entidade achar necessárias para consultar uma viagem cadastrada
Origem	Usuarios (entidade) e Administrador(entidade)
Destino	Consultar viagens agendadas(processo)
Estrutura de dados	Periodo=dia+mês+ano
	Meio de Transporte={numero}*proveniente do cadastro de motoristas*
	Intervalo de tempo=hora+minutos
	Total de Diárias=Número
	Disciplina=nome
	Data cancelamento=dia+mês+ano
	Data reagendamento=dia+mês+ano
	Protocolo=numero
	SIAPE do professor={numero}
	Origem=nome*pólo de onde o professor irá partir*
	Destino=nome*pólo onde aula será ministrada*
Fluxo de Dados	Dados do cancelamento da viagem
Descrição detalhada	Transporta informações de motivo do cancelamento da viagem agendada , data e hora do cancelamento
Origem	Administrador(entidade)
Destino	Cancelar Viagem (processo)
Estrutura de dados	Código da viagem=numero*código sequencial gerado na hora do cadastro*
	Observações=caracter_valido *todas as observações que o administrador julgar necessário acrescentar ao cadastro + data de Alterações
	Data cancelamento=dia+mês+ano
	Codigo Administrador=numero*código que informa quem cadastrou ou alterou a viagem.

Fluxo de Dados	Dados de Alteração
Descrição detalhada	Transporta todas as informações que o Administrador desejar alterar no cadastro de uma viagem
Origem	Administrador (entidade)
Destino	Alterar Viagem (processo), Reagendar viagem (processo) e viagens (depósito)
Estrutura de dados	Data Partida=dia+mês+ano
	Data Chegada=dia+mês+ano
	Meio de Transporte={numero}*proveniente do cadastro de motoristas*
	Hora de saída=hora+minutos
	Hora de chegada=hora+minutos
	Total de Diárias=numero
	Disciplina=caracter_valido
	Observações=caracter_valido *todas as observações que o administrador julgar necessário acrescentar ao cadastro + data de Alterações
	Data reagendamento=dia+mês+ano
	Protocolo=numero
	SIAPE do professor={numero}
	Local Partida=caracter_valido* não obrigatório preenchido somente quando o ponto de partida não seja o pólo de origem*
	Local Destino=caracter_valido* não obrigatório preenchido somente quando o ponto de chegada não seja o pólo de destino*
	Origem=caracter_valido*pólo de onde o professor irá partir*
Destino=caracter_valido*pólo onde aula será ministrada*	
Fluxo de Dados	Dados da Viagem
Descrição detalhada	Transporta os dados da viagem que foi inserida no depósito de viagens para confirmação de cadastro concluído
Origem	viagem (depósito)
Destino	Gravar viagem (processo)
Estrutura de dados	Código da viagem=numero*código sequencial gerado na hora do cadastro*
	Data Partida=dia+mês+ano
	Data Chegada=dia+mês+ano
	Meio de Transporte={numero}*proveniente do cadastro de motoristas*
	SIAPE do professor={numero}
	Código Administrador=numero*código que informa quem cadastrou ou alterou a viagem.
Fluxo de Dados	Relatório de Viagens
Descrição detalhada	armazena informações que serão passadas ao administrador ou usuário através de relatórios que podem ser impressos ou consultados em tela
Origem	Emitir relatorios de viagem (processo)
Destino	Administrador
Estrutura de dados	Código da viagem=numero*código sequencial gerado na hora do cadastro*
	Data Partida=dia+mês+ano
	Data Chegada=dia+mês+ano
	Meio de Transporte={numero}*proveniente do cadastro de

	motoristas*
	Hora de saída=hora+minutos
	Hora de chegada=hora+minutos
	Total de Diárias=numero
	Disciplina=caracter_valido
	Observações=caracter_valido *todas as observações que o administrador julgar necessário acrescentar ao cadastro + data de Alterações
	Data cancelamento=dia+mês+ano
	Data reagendamento=dia+mês+ano
	Protocolo=numero
	SIAPE do professor={numero}
	Local Partida=caracter_valido* não obrigatório preenchido somente quando o ponto de partida não seja o pólo de origem*
	Local Destino=caracter_valido* não obrigatório preenchido somente quando o ponto de chegada não seja o pólo de destino*
	Origem=caracter_valido*pólo de onde o professor irá partir*
	Destino=caracter_valido*pólo onde aula será ministrada*
	Codigo Administrador=numero*código que informa quem cadastrou ou alterou a viagem.
Fluxo de Dados	Ficha de Viagem
Descrição detalhada	Ficha gerada para professores com o objetivo de informa-los sobre a viagem que foi realizada.
Origem	Gerar ficha de Viagem (processo)
Destino	Usuário (entidade)
Estrutura de dados	SIAPE do professor={numero}
	Destino=caracter_valido*pólo onde aula será ministrada*
	Código da viagem=numero*código sequencial gerado na hora do cadastro*
	Meio de Transporte={numero}*proveniente do cadastro de motoristas*
	Total de Diárias=numero
	Protocolo=numero
Fluxo de Dados	Dados da Aula
Descrição detalhada	transporta as informações sobre o cronograma de aulas para que se possa agendar a vaigem
Origem	Cronogramas (depósitos)
Destino	Consultar cronograma (processo)
Estrutura de dados	Hora=hora+minutos
	Data =dia+mês+ano
	Disciplina=caracter_valido
Fluxo de Dados	Resultado de Calculo
Descrição detalhada	Armazena o resultado do cálculo de diárias para que seja armazenado no depósito.
Origem	Calcular Diárias (processo)

Destino	viagens(depósito)
Estrutura de dados	@Código da viagem=numero*código sequencial gerado na hora do cadastro*
	Total de Diárias=numero
Fluxo de Dados	Destino
Descrição detalhada	transporta informações para o calculo de diárias
Origem	Gravar viagem(processo)
Destino	Calcular diárias(processo)
Estrutura de dados	Local Partida=caracter_valido* não obrigatório preenchido somente quando o ponto de partida não seja o pólo de origem*
	Local Destino=caracter_valido* não obrigatório preenchido somente quando o ponto de chegada não seja o pólo de destino*
	Origem=caracter_valido*pólo de onde o professor irá partir*
	Destino=caracter_valido*pólo onde aula será ministrada*
	@Código da viagem= numero*código sequencial gerado na hora do cadastro*
Fluxo de Dados	Data hora de saída e retorno
Descrição detalhada	Trasnporta informações sobre a viagem.
Origem	Gravar viagem(processo).
Destino	Calcular diárias(processo).
Estrutura de dados	Hora de saída=hora+minutos
	Hora de chegada=hora+minutos
	Data Partida=dia+mês+ano
	Data Chegada=dia+mês+ano
Fluxo de Dados	Dados da aprovação
Descrição detalhada	Transporta informações sobre a aprovação do cadastro dos usuários
Origem	gerar aprovação(processo)
Destino	usuário (entidade)
Estrutura de dados	@Código do usuário= numero*código sequencial gerado na hora do cadastro*
	Permissões=[consultar cadastrar alterar todas]
Fluxo de Dados	Dados do cadastro aprovado
Descrição detalhada	Transporta os dados do usuário aprovado.
Origem	gerar aprovação(processo).
Destino	administrador(entidade).
Estrutura de dados	@Código do usuário=numero*código sequencial gerado na hora do cadastro*
	Nome do Usuário= nome+sobrenome
	Tipo Acesso= [Professor aluno administrador empresa]
	CPF=numero* 11 dígitos*

	Data aprovação=dia+mês+ano
	Código do Aprovador=numero*código do usuário que aprovou o cadastro
	Permissões=[consultar cadastrar alterar todas]
Fluxo de Dados	Consultar solicitações
Descrição detalhada	Transporta dados para localizar novas solicitações de aprovação.
Origem	administrador(entidade).
Destino	verificar solicitações(processo).
Estrutura de dados	@Código do usuário=numero*código sequencial gerado na hora do cadastro*
	Nome do Usuário= nome+sobrenome
	Data aprovação=dia+mês+ano

Elementos de dados de Fluxos	
Elementos de dados	rua={caracter_valido}
	complemento={caracter_valido}
	cidade={caracter_valido}
	estado={caracter_valido}
	cep={numero_válido}
	numero={numero_válido}
	bairro={caracter_valido}
	numero={numero_válido}
	nome={caracter_valido}
	letras={caracter_valido}
	sobrenome={caracter_valido}
	codigo de area={numero_valido} * dois digitos*
	prefixo={numero_valido} * dois digitos*
	numero de acesso={numero_valido} * oito digitos*
	dia={numero_válido [0_31]}
	mes={numero_válido [0_12]}
ano={numero_válido [1900_2099]}	
hora={numero_valido[0_24]}	
minutos={numero_valido[0_59]}	

Tabela 5: Depósito de Dados do Módulo Viagens

Depósitos de Dados	Transportes
Descrição detalhada	Armazena dados de transportes utilizados nas viagens e informações sobre os motoristas que fazem o transporte
Fluxos de Entrada	Dados do Transporte
Fluxos de Saída	-----

Estrutura de Dados	Tipo do Veículo=[Onibus Carro Particular Avião Carro da instituição]
	Nome da Instituição =nome
	@Codigo do transporte=número *número sequencial gerado na hora do cadastro*
	Código do motorista =numero *código do motorista proveniente do depósito de motoristas*
	Placa=letras+numero
	numero={numero_válido}
	nome={caracter_valido}
	letras={caracter_valido}
Depósitos de Dados	Motoristas
Descrição detalhada	Armazena dados sobre os Motoristas
Fluxos de entrada	Dados do motorista
Fluxos de Saída	-----
Estrutura de Dados	@Código do Motorista=numero *código sequencial gerado na hora do cadastro*
	Instituição Vinculada =nome
	CNH =letra+numero *carteira nacional de habilitação
	Nome do Motorista =nome+sobrenome
	Telefone = código de área + prefixo + número de acesso
	CPF=numero* 11 dígitos*
	numero={numero_válido}
	nome={caracter_valido}
	letras={caracter_valido}
	sobrenome={caracter_valido}
	codigo de area={numero_valido} * dois digitos*
	prefixo={numero_valido} * dois digitos*
	numero de acesso={numero_valido} * oito digitos*
Depósitos de Dados	Polos
Descrição detalhada	Armazena dados dos polos de ensino da instituição
Fluxos de entrada	Dados do Polo
Fluxos de Saída	-----
Estrutura de Dados	@Código da polo= numero*código sequencial gerado na hora do cadastro*
	Endereço= rua+complemento+cidade+bairro+numero+estado+CEP
	Nome do Polo =nome
	Instituição Vinculada =nome
	Tutor =nome+sobrenome
	Coordenador =nome+sobrenome
	Data Cadastro=dia+mês+ano
	Data inativo=dia+mês+ano
	Telefone = código de área + prefixo + número de acesso*

	Observações=texto
	rua={caracter_valido}
	complemento={caracter_valido}
	cidade={caracter_valido}
	estado={caracter_valido}
	cep={numero_válido}
	numero={numero_válido}
	bairro={caracter_valido}
	numero={numero_válido}
	nome={caracter_valido}
	letras={caracter_valido}
	sobrenome={caracter_valido}
	codigo de area={numero_valido} * dois digitos*
	prefixo={numero_valido} * dois digitos*
	numero de acesso={numero_valido} * oito digitos*
	dia={numero_válido [0_31]}
	mes={numero_válido [0_12]}
	ano={numero_válido [1900_2099]}
Depósitos de Dados	Viagens
Descrição detalhada	Armazena dados das viagens agendadas
Fluxos de entrada	Dados da viagem, Dados da alteração,
Fluxos de Saída	Dados da Consulta,dados da Viagem Cadastrada
Estrutura de Dados	@Código da viagem= {número_válido B72}*código sequencial gerado na hora do cadastro*
	Data Partida=dia)+mês+ano
	Data chegada=dia+mês+ano
	Meio de Transporte={número_válido}*proveniente do cadastro de motoristas*
	Hora de saída=hora+minuto
	Hora de chegada=hora+minuto
	Total de Diárias=número_valido
	Disciplina={caracter_válido}
	Observações={caracter_válido} *todas as observações que o administrador julgar necessário acrescentar ao cadastro + data de Alterações
	Data cancelamento=dia+mês+ano
	Data reagendamento=dia+mês+ano
	Protocolo= {número_válido}
	SIAPE do professor={número_válido}
	Local Partida=rua+numero+complemento+cidade+estado+cep+bairro* não obrigatório preenchido somente quando o ponto de partida não seja o pólo de origem*

	Local Destino=rua+numero+complemento+cidade+estado+cep+bairro
	Origem={numero_valido}*pólo de onde o professor irá partir*
	Destino={numero_valido}*pólo onde aula será ministrada*
	Codigo Administrador= {numero_valido}*código que informa quem cadastrou ou alterou a viagem.
Depósitos de Dados	Usuários
Descrição detalhada	Armazena todas as informações a respeito do usuário cadastrado no sistema
Fluxos de entrada	Dados do usuário, dados da aprovação
Fluxos de Saída	Consultas da Solicitação
Estrutura de Dados	@Código do usuário=numero*código sequencial gerado na hora do cadastro*
	Nome do Usuário= nome+sobrenome
	Telefone = código de área + prefixo + número de acesso*
	Email={caracter_valido}
	Tipo Acesso= [Professor aluno administrador empresa]
	CPF=numero* 11 dígitos*
	Setor=nome*opcional preenchido quando tratar-se de cadastro de administradores*
	SIAPE do professor=numero
	Matrícula=numero
	Data cadastro=dia+mês+ano
	Data aprovação=dia+mês+ano
	Código do Aprovador= numero*código do usuário que aprovou o cadastro
	Data inativo=dia+mês+ano*preenchido quando cadastro do usuário for inativado*
	Endereço= rua+complemento+cidade+bairro+numero+estado+CEP
	CNPJ={número_válido [0-9]}* 11 dígitos*
	Senha=numero+letras
	Login=nome+letras
	Razão Social=nome
	rua={caracter_valido}
	complemento={caracter_valido}
	cidade={caracter_valido}
	estado={caracter_valido}
	cep={numero_válido}
	numero={numero_válido}
	bairro={caracter_valido}
	numero={numero_válido}
	nome={caracter_valido}
letras={caracter_valido}	
sobrenome={caracter_valido}	
codigo de area={numero_valido} * dois digitos*	

prefixo={numero_valido} * dois digitos*
numero de acesso={numero_valido} * oito digitos*
dia={numero_válido [0_31]}
mes={numero_válido [0_12]}
ano={numero_válido [1900_2099]}

Tabela 6: Entidade Empresa

Entidade Externa	empresa
Descrição Detalhada	Pode realizar o cadastro de vagas, buscar informações dos alunos com o propósito de encaixa-lo em uma vaga disponível.
Fluxos de Entrada	Informações dos alunos
Fluxos de Saída	dados das vagas dados para busca dados da empresa

Tabela 7: Processos do Módulo Empresas

Processo	consultar informações sobre os alunos
Descrição Detalhada	realiza a tarefa de buscar as informações dos alunos mediante consulta ao deposito "alunos" para as empresas
Fluxos de entrada	buscar informações dos alunos informações dos alunos
Fluxos de Saída	informações dos alunos
Processo	gerenciar vagas
Descrição Detalhada	tem a função de cadastrar as vagas que a empresa disponibilizou para que qualquer aluno possa consultar
Fluxos de entrada	dados da vaga
Fluxos de Saída	dados para cadastrar vaga
Processo	verificar vaga
Descrição Detalhada	tem a função de verificar o estado da vaga no banco.
Fluxos de entrada	dados das vagas dados de consulta
Fluxos de Saída	alteração de vaga dados para cadastrar vaga dados para retirar disponibilidade de vaga
Processo	retirar disponibilidade de vaga
Descrição Detalhada	tem a função de "zerar" o numero de vagas
Fluxos de entrada	dados para retirar disponibilidade de vaga

Fluxos de Saída	dados para retirar disponibilidade de vaga
Processo	cadastrar vaga
Descrição Detalhada	recebe as informações da vaga para cadastrá-la no sistema.
Fluxos de entrada	dados para cadastrar vaga
Fluxos de Saída	dados para cadastrar vaga
Processo	alterar vaga
Descrição Detalhada	recebe as informações da vaga para alterar suas informações.
Fluxos de entrada	alteração de vaga
Fluxos de Saída	alteração de vaga

Tabela 8: Depósito do Módulo Empresas

Depositos de Dados	vagas
Descrição detalhada	armazena as vagas cadastradas pela empresa, deixando para ela disponível fazer alterações. E para os alunos consultar as vagas
Fluxos de entrada	dados para cadastrar vagas
Fluxos de Saída	.
Estrutura de Dados e elementos de dados	<p>vagas = @codigo + descrição + quantidade + nome codigo = {numero_valido} descrição = {caracter_valido} quantidade = {numero_valido} nome = {caracter_valido} caracter_valido = [A-Z a-z 0-9 , . ~ ^] numero_valido = [0-9]z codigo = *numero de registro atribuído um diferente para cada vaga cadastrada* quantidade = *pode ser atribuído um valor 0 para quando a vaga não estiver mais disponível*</p>

Tabela 9: Fluxo de Dados Módulo Empresas

Fluxo de Dados	dados das vagas
Descrição detalhada	Responsável pela transferência dos dados das vagas que a empresa irá disponibilizar para os alunos
Origem	Empresa (entidade externa)

Destino	Gerenciar vagas (processo)
Estrutura de dados e elementos de Dados	<p>Dados das vagas = Nome da empresa + código + descrição + quantidade + nome</p> <p>Nome da empresa = {caracter_valido}</p> <p>código = {numero_valido}</p> <p>descrição = {caracter_valido}</p> <p>quantidade = {numero_valido}</p> <p>nome = {caracter_valido}</p> <p>caracter_valido = {A-Z a-z , . ' ~ ^}</p> <p>numero_valido = { 0-9 }</p> <p>quantidade = * número de dois dígitos *</p> <p>código = * número gerado automaticamente ao cadastrar uma nova vaga *</p>
Fluxo de Dados	dados para cadastrar vaga
Descrição detalhada	responsável pela transferência dos dados da vaga para realizar o cadastro no sistema
Origem	gerenciar vagas (processo)
Destino	vagas (depósito)
Estrutura de dados e elementos de Dados	<p>Dados das vagas = Nome da empresa + código + descrição + quantidade + nome</p> <p>Nome da empresa = {caracter_valido}</p> <p>código = {numero_valido}</p> <p>descrição = {caracter_valido}</p> <p>quantidade = {numero_valido}</p> <p>nome = {caracter_valido}</p> <p>caracter_valido = {A-Z a-z , . ' ~ ^}</p> <p>numero_valido = { 0-9 }</p> <p>quantidade = * número de dois dígitos *</p> <p>código = * número gerado automaticamente ao cadastrar uma nova vaga *</p>
Fluxo de Dados	dados para busca
Descrição detalhada	responsável por buscar informações dos alunos para que se possa realizar um contato ou uma entrevista de emprego
Origem	empresa (entidade externa)
Destino	consultar informações sobre os alunos (processo)
Estrutura de dados e elementos de Dados	<p>buscar informações dos alunos = {nome + cod + turma + turno}</p> <p>nome = {caracter_válido}</p> <p>cod = {numero_valido}</p> <p>turma = {caracter_valido}</p> <p>turno = {caracter_valido}</p> <p>buscar informações dos alunos = *pode ser buscado por qualquer um dos campos atribuídos*</p> <p>caracter_valido = {A-Z a-z 0-9 , . ' ~ ^}</p> <p>numero_valido = {0-9 }</p>

Fluxo de Dados	informações dos alunos
Descrição detalhada	responsável por transferir os dados dos alunos para a empresa
Origem	consultar informações sobre alunos (processo)
Destino	empresa (entidade externa)
Estrutura de dados e elementos de Dados	informações dos alunos = {nome + telefone + e-mail + curriculo } nome = {caracter_Valido} telefone = {numero_valido} e-mail = {caracter_valido} curriculo = *curriculo cadastrado pelo alunos* caracter_valido = {A-Z a-z 0-9 @ . } numero_valido = {0-9}
Fluxo de Dados	informações dos alunos
Descrição detalhada	responsável por transferir os dados dos alunos para a empresa
Origem	curriculos (depósito)
Destino	consultar informações sobre os alunos (processo)
Estrutura de dados e elementos de Dados	informações dos alunos = {nome + telefone + e-mail + curriculo } nome = {caracter_Valido} telefone = {numero_valido} e-mail = {caracter_valido} curriculo = *curriculo cadastrado pelo alunos* caracter_valido = {A-Z a-z 0-9 @ . } numero_valido = {0-9}
Fluxo de Dados	dados das vagas
Descrição detalhada	responsável por transferir os dados das vagas para o processo determinar se ela existe ou não.
Origem	empresa (entidade)
Destino	verificar vaga (processo)
Estrutura de dados e elementos de Dados	Dados das vagas = Nome da empresa + código + descrição + quantidade + nome Nome da empresa = {caracter_valido} código = {numero_valido} descrição = {caracter_valido} quantidade = {numero_valido} nome = {caracter_valido} caracter_valido = {A-Z a-z , . ' ~ ^} numero_valido = { 0-9 } quantidade = * número de dois dígitos * código = * número gerado automaticamente ao cadastrar uma nova vaga *
Fluxo de Dados	dados para retirar disponibilidade de vagas

Descrição detalhada	responsável por transferir os dados para retirar a disponibilidade de uma vaga
Origem	verificar vaga (processo)
Destino	retirar disponibilidade da vaga(processo)
Estrutura de dados e elementos de Dados	<p>para retirar disponibilidade de vagas = Nome da empresa + código + quantidade</p> <p>Nome da empresa = {caracter_valido}</p> <p>código = {numero_valido}</p> <p>quantidade = {numero_valido}</p> <p>caracter_valido = {A-Z a-z , . ' ~ ^}</p> <p>numero_valido = { 0-9 }</p> <p>quantidade = * número de dois dígitos *</p> <p>código = * número gerado automaticamente ao cadastrar uma nova vaga *</p>
Fluxo de Dados	alteração de vaga
Descrição detalhada	Responsável pela transferência dos dados das vagas para alterá-las
Origem	verificar vaga (processo)
Destino	alterar vaga (processo)
Estrutura de dados e elementos de Dados	<p>alteração de vaga = Nome da empresa + código + descrição + quantidade + nome</p> <p>Nome da empresa = {caracter_valido}</p> <p>código = {numero_valido}</p> <p>descrição = {caracter_valido}</p> <p>quantidade = {numero_valido}</p> <p>nome = {caracter_valido}</p> <p>caracter_valido = {A-Z a-z , . ' ~ ^}</p> <p>numero_valido = { 0-9 }</p> <p>quantidade = * número de dois dígitos *</p> <p>código = * número gerado automaticamente ao cadastrar uma nova vaga *</p>
Fluxo de Dados	alteração de vaga
Descrição detalhada	Responsável pela transferência dos dados das vagas para alterá-las
Origem	alterar vaga (processo)
Destino	vagas (depósito)

Estrutura de dados e elementos de Dados	<p>alteração de vaga = Nome da empresa + código + descrição + quantidade + nome Nome da empresa = {caracter_valido} código = {numero_valido} descrição = {caracter_valido} quantidade = {numero_valido} nome = {caracter_valido} caracter_valido = {A-Z a-z , . ' ~ ^} numero_valido = { 0-9 } quantidade = * número de dois dígitos * código = * número gerado automaticamente ao cadastrar uma nova vaga *</p>
Fluxo de Dados	dados para cadastrar vaga
Descrição detalhada	responsável pela transferência dos dados da vaga para realizar o cadastro no sistema
Origem	verificar vaga (processo)
Destino	cadastrar vaga (processo)
Estrutura de dados e elementos de Dados	<p>Dados das vagas = Nome da empresa + código + descrição + quantidade + nome Nome da empresa = {caracter_valido} código = {numero_valido} descrição = {caracter_valido} quantidade = {numero_valido} nome = {caracter_valido} caracter_valido = {A-Z a-z , . ' ~ ^} numero_valido = { 0-9 } quantidade = * número de dois dígitos * código = * número gerado automaticamente ao cadastrar uma nova vaga *</p>
Fluxo de Dados	dados para cadastrar vaga
Descrição detalhada	responsável pela transferência dos dados da vaga para realizar o cadastro no sistema
Origem	cadastrar vaga (processo)
Destino	vagas (depósito)
Estrutura de dados e elementos de Dados	<p>Dados para cadastrar vagas = Nome da empresa + código + descrição + quantidade + nome Nome da empresa = {caracter_valido} código = {numero_valido} descrição = {caracter_valido} quantidade = {numero_valido} nome = {caracter_valido} caracter_valido = {A-Z a-z , . ' ~ ^} numero_valido = { 0-9 } quantidade = * número de dois dígitos * código = * número gerado automaticamente ao cadastrar uma nova vaga *</p>

Fluxo de Dados	dados para cadastrar vaga
Descrição detalhada	responsável pela transfêrencia dos dados da vaga para realizar o cadastro, alteração ou retirar a disponibilidade de vagas no sistema
Origem	vagas (depósito)
Destino	verificar vaga (processo)
Estrutura de dados e elementos de Dados	<p>Dados das vagas = Nome da empresa + código + descrição + quantidade + nome</p> <p>Nome da empresa = {caracter_valido}</p> <p>código = {numero_valido}</p> <p>descrição = {caracter_valido}</p> <p>quantidade = {numero_valido}</p> <p>nome = {caracter_valido}</p> <p>caracter_valido = {A-Z a-z , . ' ~ ^}</p> <p>numero_valido = { 0-9 }</p> <p>quantidade = * número de dois dígitos *</p> <p>código = * número gerado automaticamente ao cadastrar uma nova vaga *</p>
Fluxo de Dados	dados para retirar disponibilidade de vagas
Descrição detalhada	responsável por transferir os dados para retirar a disponibilidade de uma vaga
Origem	retirar disponibilidade da vaga (processo)
Destino	vagas (depósito)
Estrutura de dados e elementos de Dados	<p>para retirar disponibilidade de vagas = Nome da empresa + código + quantidade</p> <p>Nome da empresa = {caracter_valido}</p> <p>código = {numero_valido}</p> <p>quantidade = {numero_valido}</p> <p>caracter_valido = {A-Z a-z , . ' ~ ^}</p> <p>numero_valido = { 0-9 }</p> <p>quantidade = * número de dois dígitos *</p> <p>código = * número gerado automaticamente ao cadastrar uma nova vaga *</p>

Tabela 10: Fluxo de Dados do Módulo Cronograma

Fluxo de Dados	Dados do Módulo e Disciplina
Descrição detalhada	Informações necessárias para o cadastro e utilização do cronograma.
Origem	Administrador(Entidade)
Destino	Cadastrar dados do Cronograma(Processo)

Estrutura de Dados e elementos de dados	dados= código_disciplina + nome_disciplina + CH_disciplina + número modulo + nome do módulo; Número modulo={valor válido}; Nome do módulo={caractere válido}; valor válido=[0-9] *número de no máximo 2 dígitos*; caractere válido=[A-Z a-z 0-9]. Código_disciplina={número válido}; Nome_disciplina={caractere válido}; CH_disciplina={número válido};
Fluxo de Dados	Dados do Módulo
Descrição detalhada	Transmite as informações cadastradas pelo administrador ao depósito Módulo, contendo as informações referentes ao módulo.
Origem	Cadastrar dados do Cronograma(Processo)
Destino	Módulo(Depósito)
Estrutura de Dados e elementos	Módulo=número modulo + nome do módulo; Número modulo={valor válido}; Nome do módulo={caractere válido}; valor válido=[0-9] *número de no máximo 2 dígitos*; caractere válido=[A-Z a-z 0-9].
Fluxo de Dados	Dados da Disciplina
Descrição detalhada	Transmite as informações cadastradas pelo administrador ao depósito Disciplinas, contendo as informações referentes à disciplina.
Origem	Cadastrar dados do Cronograma(Processo)
Destino	Disciplinas(Depósito)
Estrutura de Dados e elementos	disciplinas= @código_disciplina + nome_disciplina + CH_disciplina; Código_disciplina={número válido}; Nome_disciplina={caractere válido}; CH_disciplina={número válido};
Fluxo de Dados	Nome dos Módulos
Descrição detalhada	O fluxo irá transmitir as informações para o um outro processo, que por sua vez enviará
Origem	Módulos(Depósitos)
Destino	Gerenciar o cronograma (Processo)

Estrutura de Dados e elementos	modulo= número modulo + nome do módulo; Número modulo={valor válido}; Nome do módulo={caractere válido}; valor válido=[0-9] *número de no máximo 2 dígitos*; caractere válido=[A-Z a-z 0-9].
Fluxo de Dados	Nome dos Polos
Descrição detalhada	Este fluxo de dados carrega as informações a serem escritas no cronograma a partir do processo 'Gerenciar o cronograma'.
Origem	Polos(Depósitos)
Destino	Gerenciar o cronograma (Processo)
Estrutura de Dados e elementos	polo= código polo + nome do polo; Código modulo={valor válido}; Nome do módulo={caractere válido}; valor válido=[0-9] *número de no máximo 2 dígitos*; caractere válido=[A-Z a-z 0-9].
Fluxo de Dados	Nome da Disciplina
Descrição detalhada	O fluxo Nome da Disciplina leva consigo os detalhes relacionados à disciplina que serão escritos no cronograma a partir do processo 'Gerenciar o Cronograma'.
Origem	Disciplinas(Depósitos)
Destino	Gerenciar o cronograma (Processo)
Estrutura de Dados e elementos	disciplinas=código da disciplina + nome da disciplina; Código da disciplina={valor válido}; Nome da disciplina={caractere válido}; valor válido=[0-9]; caractere válido=[A-Z a-z 0-9].
Fluxo de Dados	Nome e Cidade
Descrição detalhada	Envia ao cronograma informações sobre o professor a fim de que sejam inseridas no cronograma.
Origem	Professores(Depósitos)
Destino	Gerenciar o cronograma (Processo)
Estrutura de Dados e elementos	Professores=código do professor + nome do professor; Código do professor={valor válido}; Nome do professor={caractere válido}; valor válido=[0-9]; caractere válido=[A-Z a-z 0-9].

Fluxo de Dados	Informações do Cronograma
Descrição detalhada	Contém informações que serão passadas ao cronograma sem que retire as mesmas de um depósito. São informações adicionais porém fundamentais para o agendamento dos eventos.
Origem	Administrador (Entidade)
Destino	Gerenciar o cronograma (Processo)
Estrutura de Dados e elementos	informações do cronograma= código do evento + data de início do evento + data de término + tipo de evento; data de início do evento={número válido}; data de término={número válido}; Tipo de evento={número válido}; Código do evento={número válido}; Número válido=[0-9];
Fluxo de Dados	Informações do Cronograma
Descrição detalhada	Neste fluxo as informações já adicionadas pelo usuário e as informações chamadas dos depósitos são juntamente ue serão passadas ao cronograma sem que retire as mesmas de um depósito. São informações adicionais porém fundamentais para o agendamento dos eventos.
Origem	Gerenciar o cronograma (Processo)
Destino	Cronogramas (Depósito)
Estrutura de Dados e elementos	informações do cronograma=código do evento + data de início do evento + data de término + tipo de evento + disciplina + professor responsável; data de início do evento={número válido}; data de término={número válido}; Tipo de evento={número válido}; Disciplina={número válido}; Professor responsável={número válido}; *o professor é identificado pelo SIAPE*; Código do evento={número válido}; Número válido=[0-9];
Fluxo de Dados	Detalhes para revisão
Descrição detalhada	Neste fluxo o administrador pode visualizar as informações registradas no cronograma com o intuito de revisá-las.
Origem	Gerenciar o cronograma (Processo)

Destino	Administrador (Entidade)
Estrutura de Dados e elementos	<p>Detalhes para revisão=código do evento + data de início do evento + data de término + tipo de evento + disciplina + professor responsável; data de início do evento={número válido}; data de término={número válido}; Tipo de evento={número válido}; Disciplina={número válido}; Professor responsável={número válido}; *o professor é identificado pelo SIAPE*; Código do evento={número válido}; Número válido=[0-9];</p>
Fluxo de Dados	Detalhes para revisão
Descrição detalhada	Neste fluxo o administrador pode visualizar as informações registradas no cronograma com o intuito de revisá-las.
Origem	Cronogramas (Depósito)
Destino	Gerenciar o cronograma (Processo)
Estrutura de Dados e elementos	<p>Detalhes para revisão=código do evento + data de início do evento + data de término + tipo de evento + disciplina + professor responsável; data de início do evento={número válido}; data de término={número válido}; Tipo de evento={número válido}; Disciplina={número válido}; Professor responsável={número válido}; *o professor é identificado pelo SIAPE*; Código do evento={número válido}; Número válido=[0-9];</p>
Fluxo de Dados	Informações do Professores
Descrição detalhada	As informações do fluxo de dados são passadas do depósito Cronogramas para o professor a fim de que ele possa visualizar as informações do cronograma no sistema.
Origem	Cronogramas (Depósito)
Destino	Visualizar dados no cronograma (Processo)

Estrutura de Dados e elementos	<p>Informações do professor = código do evento + data de início do evento + data de término + tipo de evento + disciplina + professor responsável + código do polo + código do módulo; data de início do evento={número válido}; data de término={número válido}; Tipo de evento={número válido}; Disciplina={número válido}; Professor responsável={número válido} *o professor é identificado pelo SIAPE*; Código do Polo={número válido}; Código do Módulo={número válido}; Código do evento={número válido}; Número válido=[0-9];</p>
Fluxo de Dados	Informações do Professores
Descrição detalhada	As informações do fluxo de dados são passadas do depósito Cronogramas para o professor a fim de que ele possa visualizar as informações do cronograma no sistema.
Origem	Visualizar dados no cronograma (Processo)
Destino	Professores (Entidade)
Estrutura de Dados e elementos	<p>Informações do professor = código do evento + data de início do evento + data de término + tipo de evento + disciplina + professor responsável + código do polo + código do módulo; data de início do evento={número válido}; data de término={número válido}; Tipo de evento={número válido}; Disciplina={número válido}; Professor responsável={número válido} *o professor é identificado pelo SIAPE*; Código do Polo={número válido}; Código do Módulo={número válido}; Código do evento={número válido}; Número válido=[0-9];</p>
Fluxo de Dados	SIAPE
Descrição detalhada	Neste fluxo de dados o professor autentica a sua entrada para visualizar o cronograma. O professor deve inserir o SIAPE para ter acesso ao cronograma.
Origem	Professores (Entidade)
Destino	Visualizar dados no cronograma (Processo)

Estrutura de Dados e elementos	<p>SIAPE= número do siape; número do siape={número válido}; número válido=[0-9];</p>
Fluxo de Dados	Informações do registro de eventos
Descrição detalhada	Informa datas de eventos e avaliações registradas pelo professor
Origem	Professores (Entidade)
Destino	Registrar eventos (processo)
Estrutura de Dados e elementos	<p>Registro=Tipo de evento + data de inicio + data de termino do evento + nome do evento + polo + módulo + disciplina; tipo de evento= {numero valido}; Data de inicio= {numero valido}; Data de termino={numero valido}; Nome do evento={Caractere valido}; Polo={numero valido}; Módulo={numero valido}; Disciplina={numero valido}; Numero valido=[0-9]; Caractere valido=[A-Z a-z 0-9 .]</p>
Fluxo de Dados	Informações do registro de eventos
Descrição detalhada	Informa datas de eventos e avaliações registradas pelo professor
Origem	Registrar eventos (Processo)
Destino	Cronogramas (Deposito)
Estrutura de Dados e elementos	<p>Registro=Tipo de evento + data de inicio + data de termino do evento + nome do evento + polo + módulo + disciplina; tipo de evento= {numero valido}; Data de inicio= {numero valido}; Data de termino={numero valido}; Nome do evento={Caractere valido}; Polo={numero valido}; Módulo={numero valido}; Disciplina={numero valido}; Numero valido=[0-9]; Caractere valido=[A-Z a-z 0-9 .]</p>
Fluxo de Dados	Informações dos alunos no cronograma
Descrição detalhada	Transmite ao alunos as informações do cronograma para o sistema para que seja processado no processo Consulta ao cronograma dos Alunos.
Origem	Cronogramas(Deposito)

Destino	Consultar o Cronograma (Processo)
Estrutura de Dados e elementos	<p>Informações do cronograma= Tipo de evento + data de inicio + data de termino do evento + nome do evento + polo + módulo + disciplina; tipo de evento= {numero valido}; Data de inicio= {numero valido}; Data de termino={numero valido}; Nome do evento={Caractere valido}; Polo={numero valido}; Módulo={numero valido}; Disciplina={numero valido}; Numero valido=[0-9]; Caractere valido=[A-Z a-z 0-9 .]</p>
Fluxo de Dados	Informações dos alunos no cronograma
Descrição detalhada	Transmite ao alunos as informações específicas do aluno, que o visualiza identificando-o através do Número da Matrícula.
Origem	Consultar o Cronograma (Processo)
Destino	Alunos (Entidade)
Estrutura de Dados e elementos	<p>Informações do cronograma= Tipo de evento + data de inicio + data de termino do evento + nome do evento; tipo de evento= {numero valido}; Data de inicio= {numero valido}; Data de termino={numero valido}; Nome do evento={Caractere valido}; Numero valido=[0-9]; Caractere valido=[A-Z a-z 0-9 .]</p>
Fluxo de Dados	Matrícula
Descrição detalhada	A Matrícula é o valor que identifica o aluno, através da matrícula o aluno tem acesso único e exclusivo ao seu cronograma contendo as informações que lhe são necessárias.
Origem	Alunos (Entidade)
Destino	Consultar o cronograma (processo)
Estrutura de Dados e elementos	<p>matrícula= número da matrícula; número da matrícula={número válido}; número válido=[0-9];</p>
Fluxo de Dados	Dados a serem alterados
Descrição detalhada	Através deste fluxo o adminstrador passa as informações que desejam ser alteradas no cronograma requisitadas previamente pelos professores.

Origem	Administrador (Entidade)
Destino	Gerenciar o Cronograma (processo)
Estrutura de Dados e elementos	Dados a serem alterados=data de início do evento + data de término + tipo de evento + disciplina + professor responsável; data de início do evento={número válido}; data de término={número válido}; Tipo de evento={número válido}; Disciplina={número válido}; Professor responsável={número válido}; Número válido=[0-9];
Fluxo de Dados	Datas e horários
Descrição detalhada	Neste fluxo os dados já inseridos pelo administrador são enviados ao depósito cronogramas.
Origem	Gerenciar o cronograma (processo)
Destino	Cronogramas (Depósito)
Estrutura de Dados e elementos	Dados a serem alterados=data de início do evento + data de término + tipo de evento + disciplina + professor responsável; data de início do evento={número válido}; data de término={número válido}; Tipo de evento={número válido}; Disciplina={número válido}; Professor responsável={número válido}; Número válido=[0-9];
Fluxo de Dados	Relatório de cancelamento
Descrição detalhada	Neste fluxo o relatório é emitido ao administrador.
Origem	Gerenciar o Cronograma (processo)
Destino	Administrador (Entidade)
Estrutura de Dados e elementos	Cancelamentos=código do cancelamento + data do cancelamento + código do evento + cancelador + motivo do cancelamento; Código do Cancelamento={número válido}; Data do Cancelamento={número válido}; Código do evento={número válido}; Cancelador={número válido}; Motivo do Cancelamento={caractere válido}; Número válido=[0-9]; Caractere válido=[A-Z a-z _ . : ;- '];
Fluxo de Dados	Informações do cancelamento

Descrição detalhada	No fluxo Informações do cancelamento, o administrador encaminha os cancelamentos para o depósito. Agora já processados e organizados no sistema
Origem	Gerar informações do cancelamento(processo)
Destino	Cancelamentos (depósitos)
Estrutura de Dados e elementos	Cancelamentos=código do cancelamento + data do cancelamento + código do evento + cancelador + motivo do cancelamento; Código do Cancelamento={número válido}; Data do Cancelamento={número válido}; Código do evento={número válido}; Cancelador={número válido}; Motivo do Cancelamento={caractere válido}; Número válido=[0-9]; Caractere válido=[A-Z a-z , . : - ']];
Fluxo de Dados	Informações do cancelamento
Descrição detalhada	Depois de passar ao depósito de dados as informações são encaminhadas para ser emitido um relatório.
Origem	Cancelamentos (depósitos)
Destino	Emitir relatório de cancelamento (processo)
Estrutura de Dados e elementos	Cancelamentos=código do cancelamento + data do cancelamento + código do evento + cancelador + motivo do cancelamento; Código do Cancelamento={número válido}; Data do Cancelamento={número válido}; Código do evento={número válido}; Cancelador={número válido}; Motivo do Cancelamento={caractere válido}; Número válido=[0-9]; Caractere válido=[A-Z a-z , . : - ']];
Fluxo de Dados	Relatório de cancelamento
Descrição detalhada	Neste fluxo o relatório é emitido ao administrador.
Origem	Emitir relatório de cancelamento (processo)
Destino	Administrador (Entidade)

Estrutura de Dados e elementos	<p>Cancelamentos=código do cancelamento + data do cancelamento + código do evento + cancelador + motivo do cancelamento;</p> <p>Código do Cancelamento={número válido}; Data do Cancelamento={número válido}; Código do evento={número válido}; Cancelador={número válido}; Motivo do Cancelamento={caractere válido}; Número válido=[0-9]; Caractere válido=[A-Z a-z _ . : ; - '];</p>
Fluxo de Dados	Informações do cancelamento
Descrição detalhada	Depois de passar ao depósito de dados as informações são encaminhadas para ser emitido um relatório.
Origem	Alterar datas e horários via pedidos (processo)
Destino	Gerar informações do cancelamento (processo)
Estrutura de Dados e elementos	<p>Cancelamentos=código do cancelamento + data do cancelamento + código do evento + cancelador + motivo do cancelamento;</p> <p>Código do Cancelamento={número válido}; Data do Cancelamento={número válido}; Código do evento={número válido}; Cancelador={número válido}; Motivo do Cancelamento={caractere válido}; Número válido=[0-9]; Caractere válido=[A-Z a-z _ . : ; - '];</p>
Fluxo de Dados	Dados a serem alterados
Descrição detalhada	Depois de passar ao depósito de dados as informações são encaminhadas para ser emitido um relatório.
Origem	Administrador (Entidade)
Destino	Alterar datas e horários via pedido (processo)

Estrutura de Dados e elementos	<p>dados=código do evento + data de início do evento + data de término + tipo de evento + disciplina + professor responsável; data de início do evento={número válido}; data de término={número válido}; Tipo de evento={número válido}; Disciplina={número válido}; Professor responsável={número válido}; *o professor é identificado pelo SIAPE*; Cancelado={estado válido} Código do evento={número válido}; Número válido=[0-9]; Estado Válido=[Sim Não];</p>
Fluxo de Dados	Informações do Cronograma
Descrição detalhada	Contém informações que serão passadas ao cronograma sem que retire as mesmas de um depósito. São informações adicionais porém fundamentais para o agendamento dos eventos.
Origem	Administrador (Entidade)
Destino	Registrar Eventos (Processo)
Estrutura de Dados e elementos	<p>informações do cronograma= código do evento + data de início do evento + data de término + tipo de evento; data de início do evento={número válido}; data de término={número válido}; Tipo de evento={número válido}; Código do evento={número válido}; Número válido=[0-9];</p>
Fluxo de Dados	Informações do Cronograma
Descrição detalhada	Neste fluxo as informações já adicionadas pelo usuário e as informações chamadas dos depósitos são juntamente ue serão passadas ao cronograma sem que retire as mesmas de um depósito. São informações adicionais porém fundamentais para o agendamento dos eventos.
Origem	Registrar Eventos (Processo)
Destino	Cronogramas (Depósito)

Estrutura de Dados e elementos	informações do cronograma=código do evento + data de início do evento + data de término + tipo de evento + disciplina + professor responsável; data de início do evento={número válido}; data de término={número válido}; Tipo de evento={número válido}; Disciplina={número válido}; Professor responsável={número válido}; *o professor é identificado pelo SIAPE*; Código do evento={número válido}; Número válido=[0-9];
Fluxo de Dados	Detalhes para revisão
Descrição detalhada	Neste fluxo o administrador pode visualizar as informações registradas no cronograma com o intuito de revisá-las.
Origem	Revisar o Cronograma (Processo)
Destino	Administrador (Entidade)
Estrutura de Dados e elementos	Detalhes para revisão=código do evento + data de início do evento + data de término + tipo de evento + disciplina + professor responsável; data de início do evento={número válido}; data de término={número válido}; Tipo de evento={número válido}; Disciplina={número válido}; Professor responsável={número válido}; *o professor é identificado pelo SIAPE*; Código do evento={número válido}; Número válido=[0-9];
Fluxo de Dados	Detalhes para revisão
Descrição detalhada	Neste fluxo o administrador pode visualizar as informações registradas no cronograma com o intuito de revisá-las.
Origem	Cronogramas (Depósito)
Destino	Revisar o Cronograma (Processo)

Estrutura de Dados e elementos	<p>Detalhes para revisão=código do evento + data de início do evento + data de término + tipo de evento + disciplina + professor responsável; data de início do evento={número válido}; data de término={número válido}; Tipo de evento={número válido}; Disciplina={número válido}; Professor responsável={número válido}; *o professor é identificado pelo SIAPE*; Código do evento={número válido}; Número válido=[0-9];</p>
Fluxo de Dados	Nome dos Módulos
Descrição detalhada	O fluxo irá transmitir as informações para o um outro processo, que por sua vez enviará
Origem	Módulos(Depósitos)
Destino	Registrar Eventos (Processo)
Estrutura de Dados e elementos	<p>modulo= número modulo + nome do módulo; Número modulo={valor válido}; Nome do módulo={caractere válido}; valor válido=[0-9] *número de no máximo 2 dígitos*; caractere válido=[A-Z a-z 0-9].</p>
Fluxo de Dados	Nome dos Polos
Descrição detalhada	Este fluxo de dados carrega as informações a serem escritas no cronograma a partir do processo 'Gerenciar o cronograma'.
Origem	Polos(Depósitos)
Destino	Registrar Eventos (Processo)
Estrutura de Dados e elementos	<p>polo= código polo + nome do polo; Código modulo={valor válido}; Nome do módulo={caractere válido}; valor válido=[0-9] *número de no máximo 2 dígitos*; caractere válido=[A-Z a-z 0-9].</p>
Fluxo de Dados	Nome da Disciplina
Descrição detalhada	O fluxo Nome da Disciplina leva consigo os detalhes relacionados à disciplina que serão escritos no cronograma a partir do processo 'Gerenciar o Cronograma'.
Origem	Disciplinas(Depósitos)
Destino	Registrar Eventos (Processo)

Estrutura de Dados e elementos	disciplinas=código da disciplina + nome da disciplina; Código da disciplina={valor válido}; Nome da disciplina={caractere válido}; valor válido=[0-9]; caractere válido=[A-Z a-z 0-9].
Fluxo de Dados	Nome e Cidade
Descrição detalhada	Envia ao cronograma informações sobre o professor a fim de que sejam inseridas no cronograma.
Origem	Professores(Depósitos)
Destino	Registrar Eventos (Processo)
Estrutura de Dados e elementos	Professores=código do professor + nome do professor; Código do professor={valor válido}; Nome do professor={caractere válido}; valor válido=[0-9]; caractere válido=[A-Z a-z 0-9].
Fluxo de Dados	Datas e Horários
Descrição detalhada	Envia ao cronograma informações sobre o professor a fim de que sejam inseridas no cronograma.
Origem	Alterar datas e horários via pedidos(Processo)
Destino	Cronogramas (Depósitos)
Estrutura de Dados e elementos	dados=código do evento + data de início do evento + data de término + tipo de evento + disciplina + professor responsável; data de início do evento={número válido}; data de término={número válido}; Tipo de evento={número válido}; Disciplina={número válido}; Professor responsável={número válido}; *o professor é identificado pelo SIAPE*; Cancelado={estado válido} Código do evento={número válido}; Número válido=[0-9]; Estado Válido=[Sim Não];

Tabela 11: Entidade do Módulo Aluno

Entidade Externa	Aluno
Descrição Detalhada	A Pessoa que pode solicitar cadastro e currículo.
Fluxo De Entrada	Dados, Plano de ensino, cronograma, Notas e Faltas, Vagas
Fluxo De Saída	Nº matrícula, Dados Cadastrais, Dados, Currículo

Tabela 12: Processos do Módulo Aluno

Processo	Enviar Currículos
Descrição Detalhada	Realiza a tarefa de enviar currículo de alunos
Fluxo De Entrada	Currículo
Fluxo De Saída	Dados Currículo
Processo	Editar Dados
Descrição Detalhada	Realiza a tarefa de editar dados dos alunos
Fluxo De Entrada	Dados
Fluxo De Saída	Dados, Dados
Processo	Cadastrar Alunos
Descrição Detalhada	Realiza a tarefa de cadastrar os alunos
Fluxo De Entrada	Dados Cadastrais
Fluxo De Saída	Dados
Processo	Acessar Cronogramas
Descrição Detalhada	Realiza a tarefa de acessar o cronograma de alunos
Fluxo De Entrada	cronograma, N ^o matricula
Fluxo De Saída	Cronograma
Processo	Acessar Desempenho
Descrição Detalhada	Realiza a tarefa de acessar as notas de alunos
Fluxo De Entrada	Desempenho, N ^o matricula
Fluxo De Saída	Notas e Faltas
Processo	Consultar Vagas
Descrição Detalhada	Realiza a tarefa de consultar vagas de alunos
Fluxo De Entrada	Vagas, N ^o matricula
Fluxo De Saída	Vagas
Processo	Consulta Plano de ensino
Descrição Detalhada	Realiza a tarefa de consultar o plano de ensino
Fluxo De Entrada	Plano de ensino, N ^o matricula
Fluxo De Saída	Plano de ensino

Tabela 13: Fluxo de Dados do Módulo Aluno

Fluxo De Dados	Currículo
Descrição Detalhada	Informações do currículo de alunos cadastrados
Origem	Aluno(Entidade Externa)
Destino	Enviar Currículo (Processo)

Estrutura e Elementos de Dados	Curriculo = Nome de aluno + RG do Aluno + Sexo + Telefone Residencial + Telefone Celular + E-mail + Formação + Experiência Profissional + Informações Adicionais.
	Nome de aluno = {caracter_valido}
	RG do Aluno = {caracter_valido}
	Sexo = [M F]
	Telefone Residencial = [caracter_valido]
	Telefone Celular = [caracter_valido]
	E-mail = [caracter_valido]
	Caracter_válido = [A-Z a-z 0-9 ' - @]
Fluxo De Dados	Dados do currículo
Descrição Detalhada	Informações do currículo de alunos cadastrados
Origem	Enviar currículos(processo)
Destino	Currículos(deposito)
Estrutura e Elementos de Dados	Dados do Currículo = Nome de aluno + RG do Aluno + Sexo + Telefone Residencial + Telefone Celular + E-mail + Formação + Experiência Profissional + Informações Adicionais.
	Nome de aluno = {caracter_valido}
	RG do Aluno = {caracter_valido}
	Sexo = [M F]
	Telefone Residencial = {caracter_valido}
	Telefone Celular = {caracter_valido}
	E-mail = {caracter_valido }
	Caracter_válido = { A-Z a-z 0-9 ' - @}
Fluxo De Dados	Dados
Descrição Detalhada	Alunos visualiza seus dados cadastrados no sistema
Origem	Aluno(Entidade externa)
Destino	Editar dados(Processo)
Estrutura e Elementos de Dados	Dados = Nome de aluno + RG + Sexo + Telefone Residencial + Telefone Celular + E-mail + Modulo+senha+data de nascimento.
	Nome de aluno = {caracter_valido}
	RG do Aluno = {Caracter_válido}
	Sexo = [M F]
	Telefone Residencial = {Caracter_válido}
	Telefone Celular = {Caracter_válido}
	E-mail = [caracter_valido]
	Modulo = {caracter_valido}
	senha = {caracter_valido}
	Data de nascimento = [caracter_valido]
Caracter_válido = [A-Z a-z 0-9 ' - @]	
Fluxo De Dados	Dados
Descrição Detalhada	Alunos insere seus dados no sistema
Origem	Editar dados(Processo)

Destino	Aluno(Entidade externa)
Estrutura e Elementos de Dados	Dados = Nome de aluno + RG + Sexo + Telefone Residencial + Telefone Celular + E-mail + Modulo+senha+data de nascimento.
	Nome de aluno = {caracter_valido}
	RG do Aluno = {Caracter_válido}
	Sexo = [M F]
	Telefone Residencial = {Caracter_válido}
	Telefone Celular = {Caracter_válido}
	E-mail = [caracter_valido]
	Modulo = {caracter_valido}
	senha = {caracter_valido}
	Data de nascimento = [caracter_valido]
	Caracter_válido = [A-Z a-z 0-9 ' - @]
Fluxo De Dados	Dados
Descrição Detalhada	Alunos insere seus dados no sistema
Origem	Editar dados(Processo)
Destino	Alunos(deposito
Estrutura e Elementos de Dados	Dados = Nome de aluno + RG + Sexo + Telefone Residencial + Telefone Celular + E-mail + Modulo+senha+data de nascimento.
	Nome de aluno = {caracter_valido}
	RG do Aluno = {Caracter_válido}
	Sexo = [M F]
	Telefone Residencial = {Caracter_válido}
	Telefone Celular = {Caracter_válido}
	E-mail = [caracter_valido]
	Modulo = {caracter_valido}
	senha = {caracter_valido}
	Data de nascimento = [caracter_valido]
	Caracter_válido = [A-Z a-z 0-9 ' - @]
Fluxo De Dados	Dados cadastrais
Descrição Detalhada	Aluno insere dados, para se cadastrar no sistema
Origem	Aluno(entidade externa)
Destino	Cadastrar alunos (processo)
Estrutura e Elementos de Dados	Dados cadastrais = Nome de aluno + RG + Sexo + Telefone Residencial + Telefone Celular + E-mail + Modulo+senha+data de nascimento.
	Nome de aluno = {caracter_valido}
	RG do Aluno = {Numero_valido}
	Sexo = [M F]
	Telefone Residencial = {Numero_valido}
	Telefone Celular = {Numero_valido}
	E-mail = {caracter_valido }
	Modulo = {caracter_valido}

	senha = {caracter_valido}
	Data de nascimento = {caracter_valido}
	Caracter_válido = [A-Z a-z 0-9 ' - @]
Fluxo De Dados	Dados
Descrição Detalhada	insere dados do aluno, para guardar dados cadastrais do aluno
Origem	Cadastrar alunos (processo)
Destino	Alunos(deposito)
Estrutura e Elementos de Dados	Dados cadastrais = Nome de aluno + RG + Sexo + Telefone Residencial + Telefone Celular + E-mail + Modulo+senha+data de nascimento.
	Nome de aluno = {caracter_valido}
	RG do Aluno = {Numero_valido}
	Sexo = [M F]
	Telefone Residencial = {Numero_valido}
	Telefone Celular = {Numero_valido}
	E-mail = {caracter_valido }
	Modulo = {caracter_valido}
	senha = {caracter_valido}
	Data de nascimento = {caracter_valido}
Caracter_válido = [A-Z a-z 0-9 ' - @]	
Fluxo De Dados	Plano de ensino
Descrição Detalhada	Consulta planos de ensino
Origem	Plano de Ensino(deposito)
Destino	Consulta Plano de ensino(processo)
Estrutura e Elementos de Dados	Plano de ensino = Nome_da_materia + codigo + data+ descricao_da_materia.
	Nome_da_materia = {caracter_valido}
	codigo = {caracter_valido}
	data = {caracter_valido}
	descricao_da_materia = {caracter_valido}
Caracter_válido = [A-Z a-z 0-9 ' - @]	
Fluxo De Dados	Plano de ensino
Descrição Detalhada	Mostra informações de plano de ensino ao aluno
Origem	Consulta Plano de ensino(processo)
Destino	Aluno(entidade externa)
Estrutura e Elementos de Dados	Plano de ensino = Nome_da_materia + codigo + data+ descricao_da_materia.
	Nome_da_materia = {caracter_valido}
	codigo = {caracter_valido}
	data = {caracter_valido}
	descricao_da_materia = {caracter_valido}
Caracter_válido = [A-Z a-z 0-9 ' - @]	
Fluxo De Dados	Nº matricula
Descrição Detalhada	Informa o numero da matricula do aluno para identificar ele.

Origem	Aluno(entidade externa)
Destino	Consulta plano de ensino(processo)
Estrutura e Elementos de Dados	Nº matricula = numero_matricula
	numero_matricula = {numero_valido}
	numero_valido = [0-9]
Fluxo De Dados	Nº matricula
Descrição Detalhada	Informa o numero da matricula do aluno para identificar ele.
Origem	Aluno(entidade externa)
Destino	Consulta plano de ensino(processo)
Estrutura e Elementos de Dados	Nº matricula = numero_matricula
	numero_matricula = {numero_valido}
	numero_valido = [0-9]
Fluxo De Dados	cronograma
Descrição Detalhada	Informa ao usuario os seus horarios e datas
Origem	cronogramas(deposito)
Destino	Acessar cronograma(processo)
Estrutura e Elementos de Dados	cronogramas = horario + data inicio + data fim
	horario = {caracter_valido}
	data_inicio = {caracter_valido}
	data_fim={caracter_valido}
	Caracter_válido = [A-Z a-z 0-9 ' - @]
Fluxo De Dados	cronograma
Descrição Detalhada	Informa ao usuario os seus horarios e datas
Origem	Acessar cronograma(processo)
Destino	Aluno(deposito)
Estrutura e Elementos de Dados	cronogramas = horario + data inicio + data fim
	horario = {caracter_valido}
	data_inicio = {caracter_valido}
	data_fim={caracter_valido}
	Caracter_válido = [A-Z a-z 0-9 ' - @]
Fluxo De Dados	Nº matricula
Descrição Detalhada	Informa o numero da matricula do aluno para identificar ele.
Origem	Aluno(entidade externa)
Destino	Acessar Desempenho(processo)
Estrutura e Elementos de Dados	Nº matricula = numero_matricula
	numero_matricula = {numero_valido}
	numero_valido = [0-9]
Fluxo De Dados	Nº matricula
Descrição Detalhada	Informa o numero da matricula do aluno para identificar ele.
Origem	Aluno(entidade externa)
Destino	vagas(processo)
Estrutura e Elementos de Dados	Nº matricula = numero_matricula
	numero_matricula = {numero_valido}

	numero_valido = [0-9]
Fluxo De Dados	Desempenho
Descrição Detalhada	Informa o desempenho do aluno
Origem	Desempenho(deposito)
Destino	Acessar Desempenho(processo)
Estrutura e Elementos de Dados	Desempenho = notas + frequencia
	notas= {carcter_valido}
	frequencia={caracter_valido}
	Caracter_válido = [A-Z a-z 0-9 ' - @]
Fluxo De Dados	Desempenho
Descrição Detalhada	Informa o desempenho do aluno
Origem	Acessar Desempenho(processo)
Destino	Aluno(entidade externa)
Estrutura e Elementos de Dados	Desempenho = notas + frequencia
	notas= {carcter_valido}
	frequencia={caracter_valido}
	Caracter_válido = [A-Z a-z 0-9 ' - @]
Fluxo De Dados	vagas
Descrição Detalhada	informa as vagas para o aluno
Origem	vagas(deposito)
Destino	consultar vagas(processo)
Estrutura e Elementos de Dados	vagas = quantidade + nome+descricao
	quantidade={caracter_valido}
	nome = {caracter_valido}
	descricao={caracter_valido}
	Caracter_válido = [A-Z a-z 0-9 ' - @]

Tabela 14: Depósito do Módulo Aluno

Deposito de dados	Curriculos
Descricao detalhada	Armazena dados do curriculo dos alunos que estao cadastrados no portal
Fluxo de entrada	Dados do curriculo
Fluxo de saida	(nenhum)
Estrutura de dados e	Curriculos = Nome de aluno + @RG do Aluno + Sexo + Telefone Residencial + Telefone Celular + E-mail + Formação + Experiência Profissional + Informações Adicionais.
Elementos de dados	Nome de aluno = {caracter_valido}
	RG do Aluno = {Numero_valido}
	Sexo = [M F]
	Telefone Residencial = [Numero_valido]
	Telefone Celular = [Numero_valido]
	E-mail = [caracter_valido + Numero Valido]

	Caracter_válido = [A-Z a-z 0-9 ' - @]
	Numero_valido = [0-9]
Deposito de dados	Alunos
Descricao detalhada	Armazena informações de alunos que estão cadastradas no sistema
Fluxo de entrada	Dados,Dados
Fluxo de saída	(nenhum)
Estrutura de dados e	Alunos = Nome de aluno + @ RG + Sexo + Telefone Residencial + Telefone Celular + E-mail + Modulo+senha+data de nascimento
Elementos de dados	Nome de aluno = {caracter_valido}
	RG do Aluno = {Numero_valido}
	Sexo = [M F]
	Telefone Residencial = [Numero_valido]
	Telefone Celular = [Numero_valido]
	E-mail = [caracter_valido + Numero Valido]
	Modulo = {caracter_valido}
	senha = {caracter_valido}
	Data de nascimento = caracter_valido
	Caracter_válido = [A-Z a-z 0-9 ' - @]
	Numero_valido = [0-9]
Deposito de dados	vagas
Descricao detalhada	Armazena vagas que empresas disponibilizam
Fluxo de entrada	nenhum
Fluxo de saída	Vagas
Estrutura de dados e	Vagas = nome + descrição + quantidade
Elementos de dados	nome = {caracter_válido}
	descrição = {caracter_valido + numero+valido}
	quantidade = {numero_valido}
	caracter_válido = [A-Z a-z 0-9 ' -]
	número_valido = [0-9]
Deposito de dados	Cronogramas
Descricao detalhada	Armazena horários de aulas e datas
Fluxo de entrada	nenhum
Fluxo de saída	Dados do cronograma
Estrutura de dados e Elementos de dados	Cronogramas = horário + data_inicio + data_fim
	horario = {hh/mm}
	hh = {numero_valido}
	mm = {numero_valido}
	disciplina = {caracter_válido}
	data_inicio = {dd/mm/aa}
	data_fim = {dd/mm/aa}
dd = {numero_valido}	
	mm = {numero_válido}

	aa = {numero_válido}
	caracter_válido = [A-Z a-z 0-9 ' -]
	número_valido = [0-9]
Deposito de dados	Desempenho
Descricao detalhada	Armazena notas e frequência do aluno
Fluxo de entrada	nenhum
Fluxo de saída	Dados do Desempenho
Estrutura de dados e Elementos de dados	Desempenho = notas + frequencia
	notas = {caracter_válido}
	frequencia = {número_válido}
	caracter_válido = [A-Z a-z 0-9 ' -]
	número_valido = [0-9]
Deposito de dados	Plano de Ensino
Descricao detalhada	Armazena matérias e datas
Fluxo de entrada	nenhum
Fluxo de saída	Dados plano de ensino
Estrutura de dados e	Plano de Ensino = matéria + data
Elementos de dados	Matéria = {caracter_válido}
	data= {dd/mm/aa}
	dd = {numero_valido}
	mm = {numero_válido}
	aa = {numero_valido}
	caracter_válido = [A-Z a-z 0-9 ' -]
	número_valido = [0-9]

Tabela 15: Fluxo de Dados do Módulo Professor

Fluxo de Dados	Dados do professor
Descrição detalhada	Dados do professor para efetuar cadastro
Origem	Professores(Entidade Externa)
Destino	Efetuar cadastro(Processo)
Estrutura de dados e Elementos de Dados	Professores = título_cortesia + @nome do professor + SIAPE + endereço + telefone
	título_cortesia = [Sr. Srta. Sra.]
	nome do professor = {caracter_válido}
	SIAPE = {número_válido}
	endereço = {cidade + estado}
	telefone = {código de área + prefixo + número de acesso}
	cidade = {caracter_válido}
	estado = {caracter_válido}
	caracter_válido = [A-Z a-z 0-9 ' -]
	número_valido = [0-9]
	Código de área = * número de dois dígitos *
	Prefixo = * número de quatro dígitos que nunca começa com 0 ou 1 *

	Número de acesso = * qualquer cadeia de quatro números *
Fluxo de Dados	Cadastro
Descrição detalhada	Cadastro do professor
Origem	Efetuar cadastro(Processo)
Destino	Professores(Depósito de Dados)
Estrutura de dados e Elementos de Dados	Cadastro = título_cortesia + @nome do professor + SIAPE + endereço + telefone
	título_cortesia = [Sr. Srta. Sra.]
	nome do professor = {caracter_válido}
	SIAPE = {número_válido}
	endereço = {cidade + estado}
	telefone = {código de área + prefixo + número de acesso}
	cidade = {caracter_válido}
	estado = {caracter_válido}
	caracter_válido = [A-Z a-z 0-9 ' -]
	número_valido = [0-9]
	Código de área = * número de dois dígitos *
	Prefixo = * número de quatro dígitos que nunca começa com 0 ou 1 *
	Número de acesso = * qualquer cadeia de quatro números *
Fluxo de Dados	Notas e faltas
Descrição detalhada	Desempenho do aluno
Origem	Professores, Inserir notas e faltas(Entidade Externa,Processo)
Destino	Inserir notas e faltas, Desempenho(Processo,Depósito de Dados)
Estrutura de dados e Elementos de Dados	Notas e faltas=código_aluno+datas+notas+faltas
	código_aluno={número_válido}
	datas={número_válido + caracter_válido}
	notas={número_válido + caracter_válido}
	faltas={número_válido}
	caracter_válido = [A-Z a-z 0-9 ' -]
número_valido = [0-9]	
Fluxo de Dados	SIAPE
Descrição detalhada	Código do professor
Origem	Professores(Entidade Externa)
Destino	Consultar cronograma de aulas, Consultar viagens, Consultar diárias,Consultar disciplinas(Processos)
Estrutura de dados e Elementos de Dados	SIAPE=SIAPE_professor
	SIAPE_professor={número_válido}
	número_válido=[0-9]
Fluxo de Dados	Cronograma de aulas
Descrição detalhada	Datas e eventos de aulas

Origem	Cronogramas, Consultar cronograma de aulas (Depósito de Dados, Processo)
Destino	Consultar cronograma de aulas, Professores (Processo, Entidade Externa)
Estrutura de dados e Elementos de Dados	Cronograma de aulas=horários+data+código
	horários={número_válido + caracter_válido}
	datas={número_válido + caracter_válido}
	código={número_válido}
	caracter_válido = [A-Z a-z 0-9 ' -]
número_válido = [0-9]	
Fluxo de Dados	Datas de avaliações
Descrição detalhada	Datas de avaliações
Origem	Professores (Entidade Externa)
Destino	Registrar avaliações (Processo)
Estrutura de dados e Elementos de Dados	Datas de avaliações=datas
	datas={número_válido + caracter_válido}
	caracter_válido = [A-Z a-z 0-9 ' -]
	número_válido = [0-9]
Fluxo de Dados	Informações sobre avaliações
Descrição detalhada	Informações sobre avaliações
Origem	Registrar avaliações (Processo)
Destino	Cronogramas (Depósito de Dados)
Estrutura de dados e Elementos de Dados	Informações sobre avaliações=datas+horários
	datas={número_válido + caracter_válido}
	horários={número_válido + caracter_válido}
	caracter_válido = [A-Z a-z 0-9 ' -]
	número_válido = [0-9]
Fluxo de Dados	Cronograma de viagens
Descrição detalhada	Datas e horários de viagens
Origem	Viagens, Consultar viagens (Depósito de Dados, Processo)
Destino	Consultar viagens, Professores (Processo, Entidade Externa)
Estrutura de dados e Elementos de Dados	Cronograma de viagens=data_saída+local_de_partida+hora_saída+observações+data_cancelamento+hora_chegada+polo_origem+data_reagendamento+data_chegada+polo_destino+código+local_de_chegada
	data_saída={número_válido + caracter_válido}
	local_de_partida={caracter_válido}
	hora_saída={número_válido + caracter_válido}
	observações={caracter_válido}
	data_cancelamento={número_válido + caracter_válido}
	hora_chegada={número_válido +

	caracter_válido}
	polo_origem={caracter_válido}
	data_reagendamento={número_válido + caracter_válido}
	data_chegada={número_válido + caracter_válido}
	polo_destino={caracter_válido}
	código={número_válido}
	local_de_chegada={caracter_válido}
	caracter_válido = [A-Z a-z 0-9 ' -]
	número_valido = [0-9]
Fluxo de Dados	Diárias
Descrição detalhada	Valor que o professor tem a receber
Origem	Viagens, Consultar diárias(Depósito de Dados,Processo)
Destino	Consultar diárias, Professores(Processo,Entidade Externa)
Estrutura de dados e Elementos de Dados	Diárias=valor+data
	valor={número_válido+caracter_válido}
	data={número_válido+caracter_válido}
	caracter_válido = [A-Z a-z 0-9 ' -]
	número_valido = [0-9]
Fluxo de Dados	Detalhes das disciplinas
Descrição detalhada	Detalhes das disciplinas
Origem	Cronogramas, Consultar disciplinas(Depósito de Dados,Processo)
Destino	Consultar disciplinas, Professores(Processo,Entidade Externa)
Estrutura de dados e Elementos de Dados	Detalhes das disciplinas=nome_da_matéria+código+desc rição_da_matéria
	nome_da_matéria={caracter_válido}
	código={número_válido}
	descrição_da_matéria={caracter_válido}
	caracter_válido = [A-Z a-z 0-9 ' -]
número_valido = [0-9]	
Fluxo de Dados	Detalhes do Plano de Ensino
Descrição detalhada	Detalhes do conteúdo programático do semestre
Origem	Professores(Entidade Externa)
Destino	Registrar plano de ensino(Processo)
Estrutura de dados e Elementos de Dados	Detalhes do Plano de Ensino=nome_da_matéria+código+descriçã o_da_matéria+data
	nome_da_matéria={caracter_válido}
	código={número_válido}
	descrição_da_matéria={caracter_válido}
	data={número_válido+caracter_válido}

	caracter_válido = [A-Z a-z 0-9 ' -]
	número_valido = [0-9]
Fluxo de Dados	Plano de Ensino
Descrição detalhada	Detalhes do Plano de Ensino
Origem	Registrar plano de ensino(Processo)
Destino	Planos de Ensino(Depósito de Dados)
Estrutura de dados e Elementos de Dados	Plano de Ensino=nome_da_materia+código+descrição_da_materia+data
	nome_da_materia={caracter_válido}
	código={número_válido}
	descrição_da_materia={caracter_válido}
	data={número_válido+caracter_válido}
	caracter_válido = [A-Z a-z 0-9 ' -]
	número_valido = [0-9]
Fluxo de Dados	Nomes das disciplinas
Descrição detalhada	Detalhes das disciplinas do semestre
Origem	Cronogramas(Depósito de Dados)
Destino	Registrar plano de ensino(Processo)
Estrutura de dados e Elementos de Dados	Nomes das disciplinas=nome_da_materia+código+descrição_da_materia
	nome_da_materia={caracter_válido}
	código={número_válido}
	descrição_da_materia={caracter_válido}
	caracter_válido = [A-Z a-z 0-9 ' -]
	número_valido = [0-9]

Tabela 16: Depósito de dados do Módulo Professor

Depósitos de Dados	Professores
Descrição detalhada	Armazena dados do cadastro do professor
Fluxos de entrada	Cadastro
Fluxos de Saída	--
Estrutura de Dados e elementos de dados	Professores = {título_cortesia + @nome do professor + SIAPE + endereço + telefone}
	título_cortesia = [Sr. Srta. Sra.]
	nome do professor = {caracter_válido}
	SIAPE = {número_válido}
	endereço = {cidade + estado}
	telefone = {código de área + prefixo + número de acesso}
	cidade = {caracter_válido}
	estado = {caracter_válido}
	caracter_válido = [A-Z a-z 0-9 ' -]
	número_valido = [0-9]

	Código de área = * número de dois dígitos *
	Prefixo = * número de quatro dígitos que nunca começa com 0 ou 1 *
	Número de acesso = * qualquer cadeia de quatro números *
Depósitos de Dados	Desempenho
Descrição detalhada	Armazena desempenho do aluno
Fluxos de entrada	Notas e faltas
Fluxos de Saída	--
Depósitos de Dados	Cronograma
Descrição detalhada	Armazena datas de eventos
Fluxos de entrada	Informações de avaliações
Fluxos de Saída	Cronograma de aula, detalhes das disciplinas, nome das disciplinas
Estrutura de Dados e elementos de dados	--
Depósitos de Dados	Viagens
Descrição detalhada	Armazena datas das viagens dos professores aos pólos
Fluxos de entrada	--
Fluxos de Saída	Cronograma de viagens, diárias
Estrutura de Dados e elementos de dados	--
Depósitos de Dados	Plano de Ensino
Descrição detalhada	Armazena o conteúdo programático do semestre
Fluxos de entrada	Plano de Ensino
Fluxos de Saída	--
Estrutura de Dados e elementos de dados	Plano de ensino={Nome da matéria+descrição da matéria+data+código}
	Nome da matéria={caracter_válido}
	Descrição da matéria={caracter_válido}
	Data={número_válido + caracter_válido}
	Código={número_válido}
	caracter_válido = [A-Z a-z 0-9 ' -]
	número_válido = [0-9]

Tabela 17: Processos do Módulo Professor

Processo	Efetuar cadastro
Descrição Detalhada	O professor efetua seu cadastro no sistema.
Fluxos de entrada	Dados do professor

Fluxos de Saída	Cadastro
Processo	Inserir notas e faltas
Descrição Detalhada	O professor insere todas as notas e faltas/presenças do aluno no sistema.
Fluxos de entrada	Notas e faltas
Fluxos de Saída	Notas e faltas
Processo	Consultar cronograma de aulas
Descrição Detalhada	O professor consulta o cronograma de aula com as datas de seus compromissos.
Fluxos de entrada	SIAPE, cronograma de aulas
Fluxos de Saída	Cronograma de aulas
Processo	Registrar avaliações
Descrição Detalhada	O professor registra as datas em que haverá eventos que serão avaliados.
Fluxos de entrada	Datas de avaliações
Fluxos de Saída	Informações sobre avaliações
Processo	Consultar viagens
Descrição Detalhada	O professor consulta quais os dias em que precisará viajar até outro Pólo para realizar aula presencial.
Fluxos de entrada	SIAPE, cronograma de viagens
Fluxos de Saída	Cronograma de viagens
Processo	Consultar diárias
Descrição Detalhada	O professor consulta quais e quantas diárias tem a receber.
Fluxos de entrada	Diárias, SIAPE
Fluxos de Saída	Diárias
Processo	Consultar disciplinas
Descrição Detalhada	O professor consultará quais disciplinas lecionará em cada módulo.
Fluxos de entrada	Detalhes das disciplinas
Fluxos de Saída	Detalhes das disciplinas
Processo	Registrar plano de ensino
Descrição Detalhada	O professor registra o seu plano de ensino com todas as datas de atividades que serão realizadas durante o semestre.
Fluxos de entrada	Nomes das disciplinas, detalhes do plano de ensino
Fluxos de Saída	Plano de ensino

Tabela 18: Entidade do Módulo Professor

Entidade Externa	Professores
Descrição Detalhada	Pessoa que realiza aulas presenciais e a distância, viajando até o Pólo da aula presencial quando necessário.
Fluxos de Entrada	Cronograma de aulas, cronograma de viagens, diárias, detalhes das disciplinas.
Fluxos de Saída	Dados do professor, notas e faltas, SIAPE, datas de avaliações, detalhes do plano de ensino.

Tabela 19: Entidade do Módulo Patrimônio

Entidade Externa	Estagiário, Coordenador
Descrição detalhada	Pessoa que pode gerenciar o controle de patrimonio, assim como o controle de estoque e pedidos.
Fluxo de entrada	patrimonio(descrição, nº, conta, valor, situação), informações bens, Dados pedido, informações bens, Aviso falta de bens
Fluxo de saída	Bens(conta), Bens(quantidade), Patrimônio(nº), patrimônio(descrição, nº, conta, valor, situação, resp), Dados(descrição, preço, quantidade), Conta

Tabela 20: Processos do Módulo Patrimônio

Processo	Gerenciar estoque de bens
Descrição detalhada	É o sistema que controla o patrimônio do TSlad
Fluxos de entrada	Bens (quantidade) , Bens (Conta), Quantidade, descrição
Fluxos de saída	informações bens, Dados pedidos, informações bens, Aviso falta de bens

Processo	Cadastrar bem de consumo
Descrição Detalhada	Realiza tarefa de cadastrar os bens de consumo
Fluxos de entrada	Dados (descrição, preço, quantidade)
Fluxos de Saída	conta(descrição, quantidade, preço, setor)

Processo	Alterar bem de consumo
Descrição Detalhada	Realiza a tarefa de alterar os bens de consumo.
Fluxos de entrada	Conta.
Fluxos de Saída	Informações dos bens

Processo	Cadastrar patrimonio
Descrição Detalhada	Realiza a tarefa de cadastrar o patrimonio
Fluxos de entrada	Patrimonio (descrição, numero, sala, valor, situação, resp).
Fluxos de Saída	Descrição, nº, conta, valor, Situação, Resp

Processo	Alterar patrimonio
Descrição Detalhada	Realiza a alteração do patrimonio
Fluxos de entrada	Patrimonio (Nº).
Fluxos de Saída	Descrição, Conta, Situação, Responsavel.

Processo	Consulta patrimonio
Descrição Detalhada	Realiza a tarefa de consultar as informações do patrimonio.
Fluxos de entrada	Informações do patrimonio.
Fluxos de Saída	Patrimonio (descrição, nº, Conta, valor, situação).

Processo	Sistema Patrimônio
Descrição Detalhada	É o sistema que controla o patrimônio do TSIad
Fluxos de entrada	patrimonio (descrição, nº, conta, valor, situação, resp), patrimônio (nº), bens (quantidade), bens (conta)
Fluxos de saída	patrimonio (descrição, nº, conta, valor, situação)

Processo	Alterar quantidade
Descrição Detalhada	Realiza a tarefa de alterar as quantidades dos bens de consumo
Fluxos de entrada	bens (quantidade), informações bens
Fluxos de Saída	informações bens, quantidade

Processo	Gerar pedido
Descrição Detalhada	Realiza a tarefa de gerar pedidos para o estagiario, coordenador
Fluxos de entrada	informações do bem (descrição, valor), dados pedido
Fluxos de Saída	informações do bem (descrição)

Processo	Checar quantidade
Descrição Detalhada	Realiza a tarefa de checar as quantidades dos bens de consumo
Fluxos de entrada	bens (conta); quantidade, descrição
Fluxos de Saída	informações bens, aviso falta de bens

Tabela 21: Fluxo de Dados do Módulo Patrimônio

Fluxo de dados	Informações dos bens
Descrição detalhada	Envia do sistema ao funcionario a descrição do bem.
Origem	Gerenciar estoque de bens (Processo)
Destino	Estagiário, Coordenador (Entidade externa)

Estrutura de dados e Elementos de dados	informações do bem = descrição descrição = [caractéres válidos] *Nome do patrimonio com sua respectiva descrição caracter valido = [a-z A - Z 0 - 9]
Fluxo de dados	Patrimonio Nº
Descrição detalhada	Envia Número do patrimonio para posterior consulta no sistema
Origem	Estagiário, Coordenador(Entidade externa)
Destino	Alterar Patrimonio(Processo)
Estrutura de dados e Elementos de dados	Patrimonio = Nº Nº = {numero valido} *Numeros para registro do patrimonio numero valido = [0 -9]
Fluxo de dados	Bens (quantidade)
Descrição detalhada	Atualiza o sistema com a quantidade de bens de consumo disponiveis
Origem	Estagiário, Coordenador(Entidade externa)
Destino	Gerenciar estoque de bens(Processo)
Estrutura de dados e Elementos de dados	Bens = quantidade quantidade = [0-9] *Registra quantidade de bens
Fluxo de Dados	Dados (descrição, preço, quantidade)
Descrição detalhada	Envia os dados de cadastrar para o sistema.
Origem	Estagiario, Coordenador (entidade externa)
Destino	Cadastrar bem de consumo (processo)
Estrutura de dados e Elementos de dados	Dados = descrição + preço + quantidade descrição = [caracter válido] *Nome do patrimonio com sua respectiva descrição preço={numero valido} quantidade={numero valido} caracter valido = [a-z A - Z 0 - 9] numero valido = [0 -9]
Fluxo de Dados	Conta
Descrição detalhada	Envia os dados de alterar dos bens de consumo
Origem	Estagiario, Coordenador (entidade externa)
Destino	Alterar bem de consumo (processo)
Estrutura de dados e Elementos de dados	Conta conta= [caracter válido] caracter valido = [a-z A - Z 0 - 9]
Fluxo:	Informações de bens

Descrição detalhada:	Faz alterações de informações mantidas no depósito de dados.
Origem:	Alterar bem de consumo (processo)
Destino:	Bem de consumo (depósito de dados)
Estrutura de elementos e elementos de dados:	<p>tipo+descrição+quantidade+valor tipo = [0-9] descrição = { caracter válido } descrição = *texto obrigatorio* quantidade = { numero valido } valor = { numero valido } caracter valido = [a-z A - Z 0 - 9] numero valido = [0 -9]</p>
Fluxo:	Quantidade, Descrição
Descrição detalhada:	Tem como função mostrar a quantidade e descrição dos elementos mantidos no depósito de dados de "Bens de consumo"
Origem:	Bens de consumo (depósito de dados)
Destino:	Gerenciar estoque de Bens (processo)
Estrutura de elementos e dados:	<p>quantidade+descrição quantidade = {numero valido} descrição = {caracter valido} descrição = *texto obrigatorio* numero valido = [0-9] caracter valido = [a-z A-Z 0-9]</p>
Fluxo:	Descrição, Conta, Situação, Responsavel
Descrição detalhada:	Fluxo responsavel em cadastrar informações no depósito de dados de "bens de consumo"
Origem:	Alterar patrimonio (processo)
Destino:	Patrimonio (depósito de dados)
Estrutura de elementos e dados:	<p>Descrição+Conta+Situação+ Responsavel descrição= { caracter valido } descrição = * texto obrigatorio* numero = {numero valido} situação= {caracter valido} responsavel = {caracter valido} numero valido = [0-9] caracter valido = [a-z A-Z 0-9]</p>
Fluxo:	Informação de patrimonio
descrição detalhada:	Fluxo que mostra para o gerenciador do sistema as informações mantida do depósito de dados " Patrimonio"
Origem:	Patrimonio (depósito de dados)
Destino:	Consultar patrimonio (processo)

Estrutura de elementos e dados:	Descrição+Numero+Sala+Situação+ Responsavel descrição= { caracter valido } descrição = * texto obrigatorio* numero = {numero valido} sala = {numero valido} situação= {caracter valido} responsavel = {caracter valido} numero valido = [0-9] caracter valido = [a-z A-Z 0-9]
Fluxo:	Dados pedidos
descrição detalhada:	Fluxo que envia informações sobre pedidos para o administrador
Origem:	Gerenciar estoque de bens(Processo)
Destino:	Estagiario, Coordenador(Entidade externa)
Estrutura de elementos e dados:	Descrição+quantidade descrição= { caracter valido } descrição = * texto obrigatorio* quantidade= {numero valido} numero valido = [0-9] caracter valido = [a-z A-Z 0-9]

Fluxo:	Aviso falta de bens
descrição detalhada:	Fluxo que envia informações sobre falta de bens para o administrador
Origem:	Gerenciar estoque de bens(Processo)
Destino:	Estagiario, Coordenador(Entidade externa)
Estrutura de elementos e dados:	Descrição+quantidade descrição= { caracter valido } descrição = * texto obrigatorio* quantidade= {numero valido} numero valido = [0-9] caracter valido = [a-z A-Z 0-9]
Fluxo:	patrimonio(descrição, nº, conta, valor, situação, resp)
descrição detalhada:	Fluxo que envia informações para cadastrar patrimonio
Origem:	Estagiario, Coordenador(Entidade externa)
Destino:	Cadastrar Patrimonio(Processo)
Estrutura de elementos e dados:	Descrição+Nº+conta+valor+situação+resp descrição= { caracter valido } descrição = * texto obrigatorio* nº= {numero valido} conta={numero valido} Valor={numero valido} situação= { caracter valido } resp= { caracter valido } numero valido = [0-9] caracter valido = [a-z A-Z 0-9]

Fluxo:	descrição, nº, conta, valor, situação, resp
descrição detalhada:	Fluxo que envia informações para o banco
Origem:	Cadastrar Patrimonio(Processo)
Destino:	Patrimonio(Deposito de dados)
Estrutura de elementos e dados:	<p>Descrição+Nº+conta+valor+situação+resp descrição= { caracter valido } descrição = * texto obrigatorio* nº= {numero valido} conta={numero valido} Valor={numero valido} situação= { caracter valido } resp= { caracter valido } numero valido = [0-9] caracter valido = [a-z A-Z 0-9]</p>
Fluxo:	patrimonio(descrição, nº, conta, valor, situação)
descrição detalhada:	Fluxo que envia informações para a entidade estagiario, coordenador
Origem:	Consultar patrimonio(Processo)
Destino:	Estagiario, Coordenador(Entidade Externa)
Estrutura de elementos e dados:	<p>Descrição+Nº+conta+valor+situação descrição= { caracter valido } descrição = * texto obrigatorio* nº= {numero valido} conta={numero valido} Valor={numero valido} situação= { caracter valido } numero valido = [0-9] caracter valido = [a-z A-Z 0-9]</p>
Fluxo:	conta(descrição, quantidade, preço, setor)
descrição detalhada:	Fluxo que envia informações para cadastrar no deposito de dados "Bens de consumo"
Origem:	Cadastrar bens de consumo(Processo)
Destino:	Bens de consumo(Deposito de dados)
Estrutura de elementos e dados:	<p>Descrição+quantidade+preço+setor descrição= { caracter valido } descrição = * texto obrigatorio* quantidade= {numero valido} preço={numero valido} setor= { caracter valido } numero valido = [0-9] caracter valido = [a-z A-Z 0-9]</p>

Fluxo:	Bens (quantidade)
descrição detalhada:	Envia as quantidades de bens para ser alterada
Origem:	Estagiario,Coordenador (Entidade Externa)
Destino:	Alterar quantidade (Processo)

Estrutura de elementos e dados:	
---------------------------------	--

Fluxo:	Informações bens
descrição detalhada:	Envia as informações dos bens para o Estagiario,Coordenador
Origem:	Alterar quantidade (Processo)
Destino:	Estagiario,Coordenador (Entidade Externa)
Estrutura de elementos e dados:	informações do bem = descrição descrição = [caractères válidos] *Nome do patrimonio com sua respectiva descrição caracter valido = [a-z A - Z 0 - 9]
Fluxo:	Bens (Conta)
descrição detalhada:	
Origem:	Estagiario,Coordenador (Entidade Externa)
Destino:	Checar Quantidade (Processo)
Estrutura de elementos e dados:	Conta conta= [caracter válido] caracter valido = [a-z A - Z 0 - 9]
Fluxo:	Informações bens
descrição detalhada:	Envia as informações dos bens para o Estagiario,Coordenador
Origem:	Checar Quantidade (Processo)
Destino:	Estagiario,Coordenador (Entidade Externa)
Estrutura de elementos e dados:	informações do bem = descrição descrição = [caractères válidos] *Nome do patrimonio com sua respectiva descrição caracter valido = [a-z A - Z 0 - 9]
Fluxo:	Aviso falta de bens
descrição detalhada:	Envia informações sobre falta de bens para o estagiario,coordenador
Origem:	Checar Quantidade (Processo)
Destino:	Estagiario,Coordenador (Entidade Externa)
Estrutura de elementos e dados:	Descrição+quantidade descrição= { caracter valido } descrição = * texto obrigatorio* quantidade= {numero valido} numero valido = [0-9] caracter valido = [a-z A-Z 0-9]
Fluxo:	Dados pedido
descrição detalhada:	Fluxo que envia informações sobre pedidos para o administrador
Origem:	Gerar pedido (Processo)
Destino:	Estagiario,Coordenador (Entidade Externa)

Estrutura de elementos e dados:	<p>Descrição+quantidade descrição= { caracter valido } descrição = * texto obrigatorio* quantidade= {numero valido} numero valido = [0-9] caracter valido = [a-z A-Z 0-9]</p>
Fluxo:	Informações do bem (descrição)
descrição detalhada:	Envia as informações dos bens para os pedidos
Origem:	Gerar pedido (Processo)
Destino:	Pedidos (Deposito de dados)
Estrutura de elementos e dados:	<p>informações do bem = descrição descrição = [caractéres válidos] *Nome do patrimonio com sua respectiva descrição caracter valido = [a-z A - Z 0 - 9]</p>
Fluxo:	Informações do Bem (descrição, valor)
descrição detalhada:	Envia as informações dos bens para ser depositadas
Origem:	Gerar pedido (Processo)
Destino:	Bens de consumo (Deposito de dados)
Estrutura de elementos e dados:	<p>informações do bem = descrição+valor descrição = [caractéres válidos] *Nome do patrimonio com sua respectiva descrição valor={numero valido} caracter valido = [a-z A - Z 0 - 9]</p>
Fluxo:	Quantidade, Descrição
descrição detalhada:	Envia a quantidade e a descrição dos elementos mantidos no deposito de dados
Origem:	Bens de consumo (Deposito de dados)
Destino:	Checar quantidade (Processo)
Estrutura de elementos e dados:	<p>tipo+descrição+quantidade+valor tipo = [0-9] descrição = { caracter válido } descrição = *texto obrigatorio* quantidade = { numero valido } valor = { numero valido } caracter valido = [a-z A - Z 0 - 9] numero valido = [0 -9]</p>
Fluxo:	Quantidade
descrição detalhada:	Envia a quantidade e a descrição dos elementos mantidos no deposito de dados
Origem:	Alterar quantidade (Processo)
Destino:	Bens de consumo (Deposito de dados)
Estrutura de elementos e dados:	
Fluxo:	Informações bens
descrição detalhada:	Envia a quantidade e a descrição dos elementos mantidos no deposito de dados
Origem:	Bens de consumo (Deposito de dados)
Destino:	Alterar quantidade (Processo)

Estrutura de elementos e dados:	informações do bem = descrição descrição = [caractères válidos] *Nome do patrimonio com sua respectiva descrição caracter valido = [a-z A - Z 0 - 9]
---------------------------------	---

Tabela 22: Depósito do Módulo Patrimônio

Deposito de dados:	Bens de consumo
Descrição detalhada:	Deposito de dados usado para guardar as informações de bens de consumo
Fluxo de entrada:	Informações de bens, conta(descrição, quantidade, preço, setor)
Fluxo de saída:	Quantidade, descrição; informações bens
estrutura de elementos e dados:	tipo+descrição+quantidade+valor tipo = [0-9] descrição = { caracter válido } descrição = *texto obrigatorio* quantidade = { numero valido } valor = { numero valido } caracter valido = [a-z A - Z 0 - 9] numero valido = [0 -9]
Deposito de dados:	Patrimonio
Descrição detalhada:	Local onde são armazenadas informações sobre todo o patrimonio
Fluxo de entrada:	Descrição, conta, situação, responsavel; descrição, nº, conta, valor, situação, resp
Fluxo de saída:	Informações patrimonio
Estrutura de elementos e dados:	tipo+descrição+quantidade+valor tipo = [0-9] descrição = { caracter válido } descrição=*texto obrigatorio* quantidade = { numero valido } valor = { numero valido } caracter valido = [a-z A - Z 0 - 9] numero valido = [0 -9]
Depositos de Dados	Bens de Consumo
Descrição detalhada	Guarda as informações dos bens de consumo
Fluxos de entrada	Quantidade
Fluxos de Saída	Quantidade, descrição; informações bens
Estrutura de Dados e elementos de dados	Bens de consumo = quantidade+descrição+tipo quantidade = {numero válido} descrição = {caracter válido} *obrigatorio* tipo = {caracter válido}*obrigatorio* numero valido = [0-9] caracter valido = [a-z A-Z 0-9]
Depositos de Dados	Pedidos

Descrição detalhada	É o lugar onde ficam armazenados as informações sobre os pedidos que devem ser realizados
Fluxos de entrada	informações do bem (descrição)
Fluxos de saída	
Estrutura de Dados e elementos de dados	<p>Bens de consumo = quantidade+descricao+tipo</p> <p>quantidade = {numero válido}</p> <p>descrição = {caracter válido} *obrigatorio*</p> <p>tipo = {caracter válido}*obrigatorio*</p> <p>numero valido = [0-9]</p> <p>caracter valido = [a-z A-Z 0-9]</p>

6. PROJETO DE BANCO DE DADOS

O presente capítulo apresentará o projeto de banco de dados em três fases: Modelagem conceitual e Modelo lógico e Físico. O banco de dados foi definido com base nos tipos de dados do padrão Sistema Gerenciador de Banco de Dados (SGBD)² PostgreSQL. A criação do banco de dados foi feita através da linguagem SQL.

6.1 Modelagem Conceitual

Nesta fase do projeto, temos a visão global do sistema, pois a modelagem conceitual traz a abstração do sistema de mais alto nível e tem como objetivo a representação dos requisitos de dados do sistema (HEUSER, 1998, p.25 -26). Deve-se, então, observar com atenção o fluxo de dados do sistema e como estes estão inter-relacionados. De acordo com o autor acima citado:

“O modelo conceitual é uma descrição de quais dados serão armazenados no banco de dados, entretanto, não registra como estes dados estão armazenados a nível de um sistema de gerência de banco de dados (SGBD). Um modelo lógico é uma descrição de um banco de dados no nível de abstração visto pelo usuário do SGBD. Assim, o modelo lógico é dependente do tipo do tipo de SGBD que está sendo usado.” (HEUSER, 1998, p.25 -26).

A técnica de modelagem conceitual é a abordagem de Entidade de Relacionamento (ER), que é representada através do Diagrama de Entidade de Relacionamento (DER). Este modelo, de acordo com Heuser (1998, p. 19) “captura as necessidades da organização em termos de armazenamento de dados de forma independente de implementação”. Desta forma, este diagrama representa as

² **SGBD** - Software que incorpora as funções de definição, recuperação e alteração de dados em um banco de dados (HEUSER, 1998, p.23).

informações detalhadas das associações existentes entre as entidades de dados e usa componentes semânticos específicos apresentados a seguir:

- Entidade: são os componentes físicos e abstratos utilizados para representar a realidade modelada;
- Entidade associativa: é a redefinição de um relacionamento, que passa a ser tratado como se fosse também uma entidade
- Relacionamento: É uma correspondência entre duas entidades;
- Atributos: São os dados quem correspondem a cada ocorrência de uma entidade ou de um relacionamento;
- Ocorrência: Conjunto de atributos de uma determinada entidade;
- Especialização/Generalização: É um subconjunto das ocorrências (especializadas) de uma entidade genérica;
- Cardinalidade: Quantas ocorrências de cada entidade estão envolvidas no relacionamento. Esta pode ser: 1:1 (um para um), 1:N (um para muitos) ou N:N (muitos para muitos).

Neste contexto, a partir dos requisitos levantados e para apoiar o desenvolvimento do banco de dados, tanto em nível de modelagem quanto em nível de projeto de banco, utilizamos a ferramenta CASE “BR-Modelo³”. O BR-modelo é uma ferramenta freeware⁴ voltada para ensino de modelagem em banco de dados relacional com base na metodologia defendida por Carlos A. Heuser, sendo a mais usada na modelagem conceitual.

A seguir, na figura 17, elaboramos o Diagrama de Entidade-Relacionamento do Sistema Gère:

³ O brModelo é uma ferramenta desenvolvida para modelagem de um banco de dados criada num curso de pós-graduação orientado por Carlos Heuser, autor do livro “Projeto de Banco de Dados”.

⁴ Freeware é qualquer programa de computador cuja utilização não implica o pagamento de licenças de uso ou *royalties*.

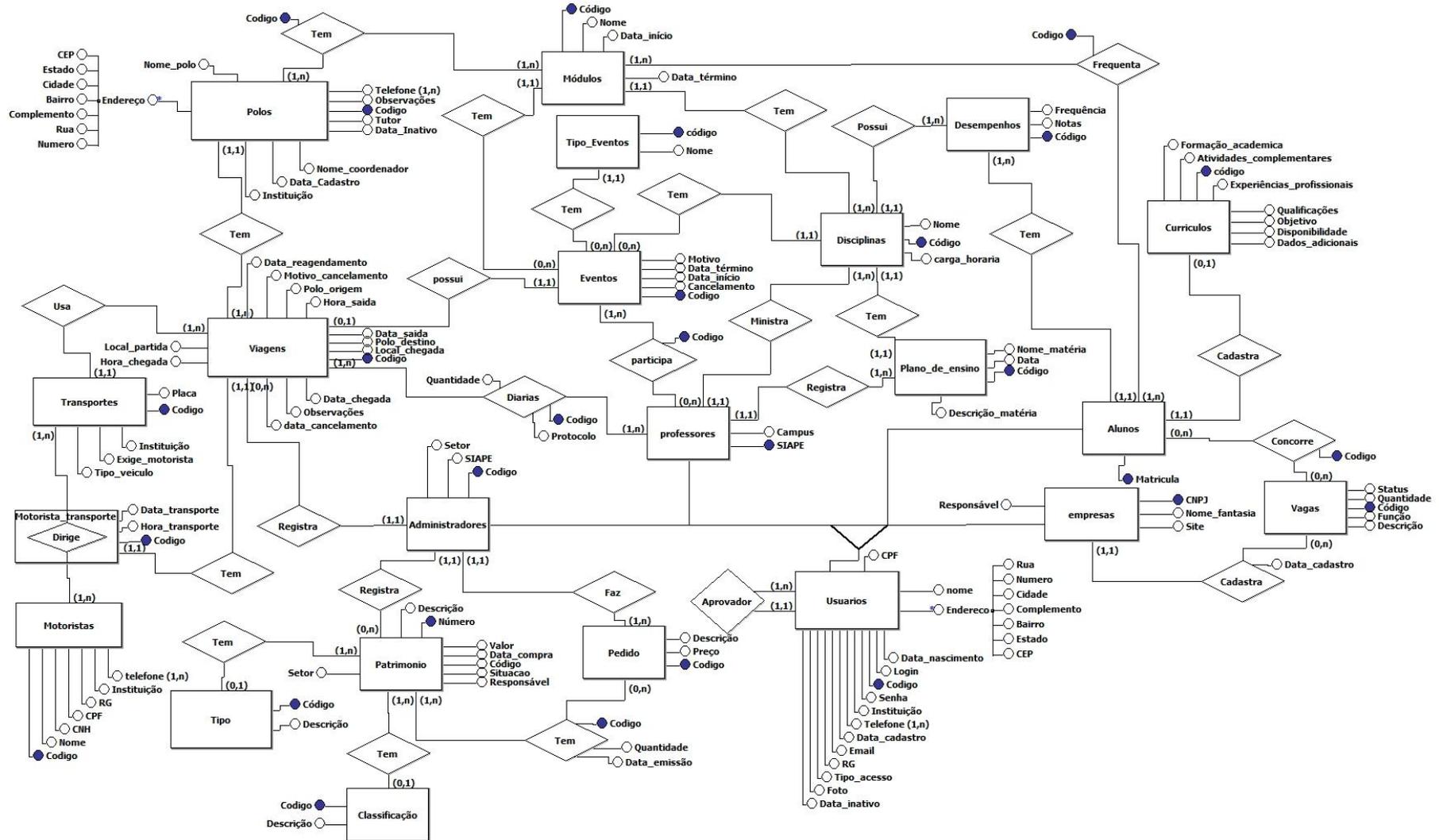


Figura 15: Modelo conceitual do Sistema Gèrè

62 Modelo Lógico

O modelo lógico apresenta algumas limitações do banco e implementa recursos como adequação de padrão e nomenclatura. A análise da integridade do banco é feita definindo-se regras com a criação de chaves primárias e estrangeiras que são criadas a partir da análise das cardinalidades do modelo conceitual.

Assim, a seguir apresentamos a estrutura do banco de dados e os tipos de dados que foram definidos nas duas primeiras tabelas como legendas:

Tabela 23 Legenda para simbologia utilizada nas colunas nome do campo das tabelas

Nome do campo	
#	Chave primária
@	Chave estrangeira

Tabela 24: para simbologia utilizada nas colunas tipo de dados das tabelas

Tipo de dados	Descrição
Serial	Inteiro de oito bytes com auto-incremento
Varchar	Permite armazenar cadeias de caracteres, mas com tamanho variável.
Char	Permite armazenar cadeias de caracteres (letras, símbolos e números).
Date	Permite armazenar datas.
Integer	Permite armazenar número positivo ou negativo.
Time	Permite armazenar horários.

Modelo Logico - administradores					
Generated: 13/07/2011 09:43:05					
Server: PostgreSQL Database Server 8.2 (localhost:5432)					
Database: gere_final					
Schema: public					
Colunas					
Campo	Tipo de dados	Not Null?	Primary key?	Default	Comment
codigo	integer	Yes	Yes	nextval('administradores_codigo_seq'::regclass)	
siape	integer	Yes	No		
setor	character varying(150)	No	No		
usuario	integer	Yes	No		
Constraints					
Campo	Type	Definition	Comment		
administradores_pkey	Primary key	(codigo)			
administradores_usuario_fkey	Foreign key	(usuario) REFERENCES usuarios (codigo) MATCH SIMPLE ON UPDATE NO ACTION ON DELETE NO ACTION			

Figura 16: Modelo Lógico - Administradores

Modelo Logico - alunos					
Generated: 13/07/2011 09:43:34					
Server: PostgreSQL Database Server 8.2 (localhost:5432)					
Database: gere_final					
Schema: public					
Colunas					
Campo	Tipo de dados	Not Null	Primary key?	Default	Comment
matricula	integer	Yes	Yes		
usuario	integer	Yes	No		
Constraints					
Campo	Type	Definition	Comment		
alunos_pkey	Primary key	(matricula)			
alunos_usuario_fkey	Foreign key	(usuario) REFERENCES usuarios (codigo) MATCH SIMPLE ON UPDATE NO ACTION ON DELETE NO ACTION			

Figura 17: Modelo Lógico - Alunos

Modelo Logico - classificacao					
Generated: 13/07/2011 09:43:52					
Server: PostgreSQL Database Server 8.2 (localhost:5432)					
Database: gere_final					
Schema: public					
Colunas					
Campo	Tipo de dados	Not Null	Primary key?	Default	Comment
codigo	integer	Yes	Yes	nextval('classificacao_codigo_seq'::regclass)	
descricao	character varying(150)	Yes	No		
Constraints					
Campo	Type	Definition	Comment		
classificacao_pkey	Primary key	(codigo)			

Figura 18: Modelo Lógico – Classificação

Modelo Logico - concorre

Generated: 13/07/2011 09:44:08
 Server: PostgreSQL Database Server 8.2 (localhost:5432)
 Database: gere_final
 Schema: public

Colunas

Campo	Tipo de dados	Not Null	Primary key?	Default	Comment
codigo	integer	Yes	Yes		
vaga	integer	No	No		
aluno	integer	No	No		

Constraints

Campo	Type	Definition	Comment
concorre_pkey	Primary key	(codigo)	
concorre_aluno_fkey	Foreign key	(aluno) REFERENCES alunos (matricula) MATCH SIMPLE ON UPDATE NO ACTION ON DELETE NO ACTION	
concorre_vaga_fkey	Foreign key	(vaga) REFERENCES vagas (codigo) MATCH SIMPLE ON UPDATE NO ACTION ON DELETE NO ACTION	

Figura 19: Modelo Lógico – Concorre

Modelo Logico - diarias

Generated: 13/07/2011 09:44:25
 Server: PostgreSQL Database Server 8.2 (localhost:5432)
 Database: gere_final
 Schema: public

Colunas

Campo	Tipo de dados	Not Null	Primary key?	Default	Comment
codigo	integer	Yes	Yes	nextval('diarias_codigo_seq'::regclass)	
protocolo	character(50)	No	No		
quantidade	numeric(2,1)	Yes	No		
viagem	integer	Yes	No		
professor	integer	Yes	No		

Constraints

Campo	Type	Definition	Comment
diarias_pkey	Primary key	(codigo)	
diarias_professor_fkey	Foreign key	(professor) REFERENCES professores (siape) MATCH SIMPLE ON UPDATE NO ACTION ON DELETE NO ACTION	
diarias_viagem_fkey	Foreign key	(viagem) REFERENCES viagens (codigo) MATCH SIMPLE ON UPDATE NO ACTION ON DELETE NO ACTION	

Figura 20: Modelo Lógico – Diárias

Modelo Logico - disciplinas

Generated: 13/07/2011 09:44:39
 Server: PostgreSQL Database Server 8.2 (localhost:5432)
 Database: gere_final
 Schema: public

Colunas

Campo	Tipo de dados	Not Null	Primary key?	Default	Comment
codigo	integer	Yes	Yes	nextval('disciplinas_codigo_seq'::regclass)	
carga_horaria	character varying(15)	Yes	No		
nome	character varying(150)	Yes	No		
modulo	integer	Yes	No		
professor	integer	Yes	No		

Constraints

Campo	Type	Definition	Comment
disciplinas_pkey	Primary key	(codigo)	
disciplinas_modulo_fkey	Foreign key	(modulo) REFERENCES modulos (codigo) MATCH SIMPLE ON UPDATE NO ACTION ON DELETE NO ACTION	
disciplinas_professor_fkey	Foreign key	(professor) REFERENCES professores (siape) MATCH SIMPLE ON UPDATE NO ACTION ON DELETE NO ACTION	

Figura 21: Modelo Lógico – Disciplinas

Modelo Logico - empresas

Generated: 13/07/2011 09:44:56
 Server: PostgreSQL Database Server 8.2 (localhost:5432)
 Database: gere_final
 Schema: public

Colunas

Campo	Tipo de dados	Not Null	Primary key?	Default	Comment
cnj	integer	Yes	Yes		
nome_fantasia	character varying(150)	Yes	No		
site	character varying(150)	No	No		
responsavel	character varying(150)	Yes	No		
usuario	integer	Yes	No		

Constraints

Campo	Type	Definition	Comment
empresas_pkey	Primary key	(cnj)	
empresas_usuario_fkey	Foreign key	(usuario) REFERENCES usuarios (codigo) MATCH SIMPLE ON UPDATE NO ACTION ON DELETE NO ACTION	

Figura 22: Modelo Lógico – Empresas

Modelo Logico - eventos

Generated: 13/07/2011 09:45:09
 Server: PostgreSQL Database Server 8.2 (localhost:5432)
 Database: gere_final
 Schema: public

Colunas

Campo	Tipo de dados	Not Null	Primary key?	Default	Comment
codigo	integer	Yes	Yes	nextval('eventos_codigo_seq'::regclass)	
cancelamento	character(1)	No	No		S= Sim, N =NÃO
motivo	character varying(150)	No	No		
professor	integer	Yes	No		
data_cancelamento	date	No	No		
data_termino	date	Yes	No		
data_inicio	date	Yes	No		
modulo	integer	Yes	No		
tipo_evento	integer	Yes	No		
disciplina	integer	Yes	No		

Constraints

Campo	Type	Definition	Comment
eventos_pkey	Primary key	(codigo)	
eventos_disciplina_fkey	Foreign key	(disciplina) REFERENCES disciplinas (codigo) MATCH SIMPLE ON UPDATE NO ACTION ON DELETE NO ACTION	
eventos_modulo_fkey	Foreign key	(modulo) REFERENCES modulos (codigo) MATCH SIMPLE ON UPDATE NO ACTION ON DELETE NO ACTION	
eventos_tipo_evento_fkey	Foreign key	(tipo_evento) REFERENCES tipo_eventos (codigo) MATCH SIMPLE ON UPDATE NO ACTION ON DELETE NO ACTION	
eventos_cancelamento_check	Check	(cancelamento = 'S'::bpchar OR cancelamento = 'N'::bpchar)	

Figura 23: Modelo Lógico – Eventos

Modelo Logico - frequenta

Generated: 13/07/2011 09:45:22
 Server: PostgreSQL Database Server 8.2 (localhost:5432)
 Database: gere_final
 Schema: public

Colunas

Campo	Tipo de dados	Not Null	Primary key?	Default	Comment
codigo	integer	Yes	Yes	nextval('frequenta_codigo_seq'::regclass)	
modulo	integer	No	No		
aluno	integer	No	No		

Constraints

Campo	Type	Definition	Comment
frequenta_pkey	Primary key	(codigo)	
frequenta_aluno_fkey	Foreign key	(aluno) REFERENCES alunos (matricula) MATCH SIMPLE ON UPDATE NO ACTION ON DELETE NO ACTION	
frequenta_modulo_fkey	Foreign key	(modulo) REFERENCES modulos (codigo) MATCH SIMPLE ON UPDATE NO ACTION ON DELETE NO ACTION	

Figura 24: Modelo Lógico – Frequenta

Modelo Logico - modulos

Generated: 13/07/2011 09:45:56
 Server: PostgreSQL Database Server 8.2 (localhost:5432)
 Database: gere_final
 Schema: public

Colunas

Campo	Tipo de dados	Not Null	Primary key?	Default	Comment
codigo	integer	Yes	Yes	nextval('modulos_codigo_seq'::regclass)	
data_inicio	date	Yes	No		
data_termino	date	Yes	No		
nome	character varying(25)	Yes	No		

Constraints

Campo	Type	Definition	Comment
modulos_pkey	Primary key	(codigo)	

Figura 25: Modelo Lógico – Módulos

Modelo Logico - modulo_polos

Generated: 13/07/2011 09:45:36
 Server: PostgreSQL Database Server 8.2 (localhost:5432)
 Database: gere_final
 Schema: public

Colunas

Campo	Tipo de dados	Not Null	Primary key?	Default	Comment
codigo	integer	Yes	Yes	nextval('modulo_polos_codigo_seq'::regclass)	
polo	integer	Yes	No		
modulo	integer	Yes	No		

Constraints

Campo	Type	Definition	Comment
modulo_polos_pkey	Primary key	(codigo)	
modulo_polos_modulo_fkey	Foreign key	(modulo) REFERENCES modulos (codigo) MATCH SIMPLE ON UPDATE NO ACTION ON DELETE NO ACTION	
modulo_polos_polo_fkey	Foreign key	(polo) REFERENCES polos (codigo) MATCH SIMPLE ON UPDATE NO ACTION ON DELETE NO ACTION	

Figura 26: Modelo Lógico – Módulo_polos

Modelo Logico - motoristas

Generated: 13/07/2011 09:46:09
 Server: PostgreSQL Database Server 8.2 (localhost:5432)
 Database: gere_final
 Schema: public

Colunas

Campo	Tipo de dados	Not Null	Primary key?	Default	Comment
codigo	integer	Yes	Yes	nextval('motoristas_codigo_seq'::regclass)	
nome	character varying(150)	Yes	No		
cnh	integer	Yes	No		
cpf	integer	Yes	No		
rg	integer	Yes	No		
instituicao	character varying(150)	Yes	No		

Constraints

Campo	Type	Definition	Comment
motoristas_pkey	Primary key	(codigo)	

Figura 27: Modelo Lógico – Motoristas

Modelo Logico - participa						
Generated: 13/07/2011 09:46:21						
Server: PostgreSQL Database Server 8.2 (localhost:5432)						
Database: gere_final						
Schema: public						
Colunas						
Campo	Tipo de dados	Not Null	Primary key?	Default	Comment	
codigo	integer	Yes	Yes	nextval('participa_codigo_seq'::regclass)		
evento	integer	No	No			
professor	integer	No	No			
Constraints						
Campo	Type	Definition	Comment			
participa_pkey	Primary key	(codigo)				
participa_evento_fkey	Foreign key	(evento) REFERENCES eventos (codigo) MATCH SIMPLE ON UPDATE NO ACTION ON DELETE NO ACTION				
participa_professor_fkey	Foreign key	(professor) REFERENCES professores (siae) MATCH SIMPLE ON UPDATE NO ACTION ON DELETE NO ACTION				

Figura 28: Modelo Lógico – Participa

Modelo Logico - patrimonio						
Generated: 13/07/2011 09:46:34						
Server: PostgreSQL Database Server 8.2 (localhost:5432)						
Database: gere_final						
Schema: public						
Colunas						
Campo	Tipo de dados	Not Null	Primary key?	Default	Comment	
numero	integer	Yes	Yes			
codigo	integer	Yes	No	nextval('patrimonio_codigo_seq'::regclass)		
respopnstavel	character varying(150)	Yes	No			
preco	numeric(5,2)	Yes	No			
setor	character varying(150)	Yes	No			
data_compra	date	Yes	No			
descricao	character varying(150)	Yes	No			
situacao	character varying(1)	Yes	No			
tipo	integer	No	No			
classificacao	integer	No	No			
administrador	integer	No	No			
Constraints						
Campo	Type	Definition	Comment			
patrimonio_pkey	Primary key	(numero)				
patrimonio_administrador_fkey	Foreign key	(administrador) REFERENCES administradores (codigo) MATCH SIMPLE ON UPDATE NO ACTION ON DELETE NO ACTION				
patrimonio_classificacao_fkey	Foreign key	(classificacao) REFERENCES classificacao (codigo) MATCH SIMPLE ON UPDATE NO ACTION ON DELETE NO ACTION				
patrimonio_tipo_fkey	Foreign key	(tipo) REFERENCES tipo (codigo) MATCH SIMPLE ON UPDATE NO ACTION ON DELETE NO ACTION				
patrimonio_situacao_check	Check	(situacao::text = 'A'::text OR situacao::text = 'B'::text OR situacao::text = 'E'::text)				

Figura 29: Modelo Lógico – Patrimônio

Modelo Logico - pedido						
Generated: 13/07/2011 09:46:48						
Server: PostgreSQL Database Server 8.2 (localhost:5432)						
Database: gere_final						
Schema: public						
Colunas						
Campo	Tipo de dados	Not Null	Primary key?	Default	Comment	
codigo	integer	Yes	Yes	nextval('pedido_codigo_seq'::regclass)		
preco	numeric(5,2)	Yes	No			
descricao	character varying(150)	Yes	No			
administrador	integer	No	No			
Constraints						
Campo	Type	Definition	Comment			
pedido_pkey	Primary key	(codigo)				
pedido_administrador_fkey	Foreign key	(administrador) REFERENCES administradores (codigo) MATCH SIMPLE ON UPDATE NO ACTION ON DELETE NO ACTION				

Figura 30: Modelo Lógico – Pedido

Modelo Logico - pedido_patrimonio

Generated: 13/07/2011 09:47:02
 Server: PostgreSQL Database Server 8.2 (localhost:5432)
 Database: gere_final
 Schema: public

Colunas

Campo	Tipo de dados	Not Null	Primary key?	Default	Comment
codigo	integer	Yes	Yes	nextval('pedido_patrimonio_codigo_seq'::regclass)	
quantidade	integer	Yes	No		
data_emissao	integer	Yes	No		
patrimonio	integer	No	No		
pedido	integer	No	No		

Constraints

Campo	Type	Definition	Comment
pedido_patrimonio_pkey	Primary key	(codigo)	
pedido_patrimonio_patrimonio_fkey	Foreign key	(patrimonio) REFERENCES patrimonio (numero) MATCH SIMPLE ON UPDATE NO ACTION ON DELETE NO ACTION	
pedido_patrimonio_pedido_fkey	Foreign key	(pedido) REFERENCES pedido (codigo) MATCH SIMPLE ON UPDATE NO ACTION ON DELETE NO ACTION	

Figura 31: Modelo Lógico – Pedido_patrimônio

Modelo Logico - plano_ensino

Generated: 13/07/2011 09:47:16
 Server: PostgreSQL Database Server 8.2 (localhost:5432)
 Database: gere_final
 Schema: public

Colunas

Campo	Tipo de dados	Not Null	Primary key?	Default	Comment
codigo	integer	Yes	Yes	nextval('plano_ensino_codigo_seq'::regclass)	
data	date	Yes	No		
nome_materia	character varying(150)	Yes	No		
descricao_materia	character varying(150)	No	No		
professor	integer	No	No		
disciplina	integer	No	No		

Constraints

Campo	Type	Definition	Comment
plano_ensino_pkey	Primary key	(codigo)	
plano_ensino_disciplina_fkey	Foreign key	(disciplina) REFERENCES disciplinas (codigo) MATCH SIMPLE ON UPDATE NO ACTION ON DELETE NO ACTION	
plano_ensino_professor_fkey	Foreign key	(professor) REFERENCES professores (siae) MATCH SIMPLE ON UPDATE NO ACTION ON DELETE NO ACTION	

Figura 32: Modelo Lógico – Plano_ensino

Modelo Logico - polos

Generated: 13/07/2011 09:47:32
 Server: PostgreSQL Database Server 8.2 (localhost:5432)
 Database: gere_final
 Schema: public

Colunas

Campo	Tipo de dados	Not Null	Primary key?	Default	Comment
codigo	integer	Yes	Yes	nextval('polos_codigo_seq'::regclass)	
observacao	character varying(255)	No	No		
nome_coordenador	character varying(150)	Yes	No		
nome_polo	character varying(25)	Yes	No		
data_inativo	date	No	No		
tutor	character varying(25)	Yes	No		
data_cadastro	date	Yes	No		
instituicao	character varying(150)	Yes	No		
rua	character varying(150)	Yes	No		
numero	character varying(10)	Yes	No		
cep	integer	Yes	No		
bairro	character varying(50)	Yes	No		
cidade	character varying(50)	Yes	No		
estado	character(2)	Yes	No		

Constraints

Campo	Type	Definition	Comment
polos_pkey	Primary key	(codigo)	

Figura 33: Modelo Lógico – Polos

Modelo Logico - professores

Generated: 13/07/2011 09:48:24
 Server: PostgreSQL Database Server 8.2 (localhost:5432)
 Database: gere_final
 Schema: public

Colunas

Campo	Tipo de dados	Not Null	Primary key?	Default	Comment
siape	integer	Yes	Yes		
usuario	integer	Yes	No		
campus	character varying(150)	Yes	No		

Constraints

Campo	Type	Definition	Comment
professores_pkey	Primary key	(siape)	
professores_usuario_fkey	Foreign key	(usuario) REFERENCES usuarios (codigo) MATCH SIMPLE ON UPDATE NO ACTION ON DELETE NO ACTION	

Figura 34: Modelo Lógico – Professores

Modelo Logico - telefones

Generated: 13/07/2011 09:48:31
 Server: PostgreSQL Database Server 8.2 (localhost:5432)
 Database: gere_final
 Schema: public

Colunas

Campo	Tipo de dados	Not Null	Primary key?	Default	Comment
codigo	integer	Yes	Yes	nextval('telefones_codigo_seq'::regclass)	
tipo_telefone	character(1)	Yes	No		C= CELULAR, F=FIXO
numero_telefone	integer	Yes	No		
ddd	character(2)	Yes	No		
cod_polo	integer	No	No		
cod_usuario	integer	No	No		
cod_motorista	integer	No	No		

Constraints

Campo	Type	Definition	Comment
telefones_pkey	Primary key	(codigo)	
telefones_cod_motorista_fkey	Foreign key	(cod_motorista) REFERENCES motoristas (codigo) MATCH SIMPLE ON UPDATE NO ACTION ON DELETE NO ACTION	
telefones_cod_polo_fkey	Foreign key	(cod_polo) REFERENCES polos (codigo) MATCH SIMPLE ON UPDATE NO ACTION ON DELETE NO ACTION	
telefones_cod_usuario_fkey	Foreign key	(cod_usuario) REFERENCES usuarios (codigo) MATCH SIMPLE ON UPDATE NO ACTION ON DELETE NO ACTION	
telefones_tipo_telefone_check	Check	(tipo_telefone = 'C'::bpchar OR tipo_telefone = 'F'::bpchar)	

Figura 35: Modelo Lógico – Telefones

Modelo Logico - tipo

Generated: 13/07/2011 09:48:46
 Server: PostgreSQL Database Server 8.2 (localhost:5432)
 Database: gere_final
 Schema: public

Colunas

Campo	Tipo de dados	Not Null	Primary key?	Default	Comment
codigo	integer	Yes	Yes	nextval('tipo_codigo_seq'::regclass)	
descricao	character varying(150)	Yes	No		

Constraints

Campo	Type	Definition	Comment
tipo_pkey	Primary key	(codigo)	

Figura 36: Modelo Lógico – Tipo

Modelo Logico - tipo_eventos					
Generated: 13/07/2011 09:48:59					
Server: PostgreSQL Database Server 8.2 (localhost:5432)					
Database: gere_final					
Schema: public					
Colunas					
Campo	Tipo de dados	Not Null	Primary key?	Default	Comment
codigo	integer	Yes	Yes	nextval('tipo_eventos_codigo_seq'::regclass)	
nome	character varying(150)	Yes	No		
Constraints					
Campo	Type	Definition	Comment		
tipo_eventos_pkey	Primary key	(codigo)			

Figura 37: Modelo Lógico – Tipo_eventos

Modelo Logico - transporte_motorista					
Generated: 13/07/2011 09:49:09					
Server: PostgreSQL Database Server 8.2 (localhost:5432)					
Database: gere_final					
Schema: public					
Colunas					
Campo	Tipo de dados	Not Null	Primary key?	Default	Comment
codigo	integer	Yes	Yes	nextval('transporte_motorista_codigo_seq'::regclass)	
transporte	integer	Yes	No		
motorista	integer	Yes	No		
data_transporte	date	Yes	No		
hora_transporte	time without time zone	Yes	No		
Constraints					
Campo	Type	Definition	Comment		
transporte_motorista_pkey	Primary key	(codigo)			
transporte_motorista_motorista_fkey	Foreign key	(motorista) REFERENCES motoristas (codigo) MATCH SIMPLE ON UPDATE NO ACTION ON DELETE NO ACTION			
transporte_motorista_transporte_fkey	Foreign key	(transporte) REFERENCES transportes (codigo) MATCH SIMPLE ON UPDATE NO ACTION ON DELETE NO ACTION			

Figura 38: Modelo Lógico – Transporte_motorista

Modelo Logico - transportes					
Generated: 13/07/2011 09:49:26					
Server: PostgreSQL Database Server 8.2 (localhost:5432)					
Database: gere_final					
Schema: public					
Colunas					
Campo	Tipo de dados	Not Null	Primary key?	Default	Comment
codigo	integer	Yes	Yes	nextval('transportes_codigo_seq'::regclass)	
tipo_veiculo	character(1)	No	No		C= CARRO, O = ONIBUS, A= AVIAO,P= VEICULO PARTICULAR
placa	character(15)	No	No		
instituicao	character varying(150)	No	No		
exige_motorista	character(1)	No	No		S=Sim N=NÃO
Constraints					
Campo	Type	Definition	Comment		
transportes_pkey	Primary key	(codigo)			
transportes_exige_motorista_check	Check	(exige_motorista = 'S'::bpchar OR exige_motorista = 'N'::bpchar)			
transportes_tipo_veiculo_check	Check	(tipo_veiculo = 'C'::bpchar OR tipo_veiculo = 'O'::bpchar OR tipo_veiculo = 'A'::bpchar OR tipo_veiculo = 'P'::bpchar)			

Figura 39: Modelo Lógico – Transportes

Modelo Logico - usuarios

Generated: 13/07/2011 09:49:41
 Server: PostgreSQL Database Server 8.2 (localhost:5432)
 Database: gere_final
 Schema: public

Colunas

Campo	Tipo de dados	Not Null	Primary key?	Default	Comment
codigo	integer	Yes	Yes	nextval('usuarios_codigo_seq'::regclass)	
instituicao	character varying(150)	Yes	No		
rg	integer	Yes	No		
email	character varying(15)	Yes	No		
cpf	character(11)	Yes	No		
data_inativo	date	No	No		
data_nascimento	date	Yes	No		
login	character(6)	Yes	No		
senha	character(6)	Yes	No		
nome	character varying(150)	Yes	No		
data_cadastro	date	Yes	No		
aprovador	integer	No	No		
rua	character varying(150)	Yes	No		
numero	character varying(10)	Yes	No		
complemento	character varying(50)	Yes	No		
bairro	character varying(50)	Yes	No		
cidade	character varying(50)	Yes	No		
estado	character(2)	Yes	No		
cep	integer	Yes	No		
foto	character varying(150)	No	No		
tipo_acesso	character(2)	No	No		AD= administrador AL=aluno PO= professor e EP empresas

Constraints

Campo	Type	Definition	Comment
usuarios_pkey	Primary key	(codigo)	
usuarios_tipo_acesso_check	Check	(tipo_acesso = 'AD'::bpchar OR tipo_acesso = 'AL'::bpchar OR tipo_acesso = 'PO'::bpchar OR tipo_acesso = 'EP'::bpchar)	

Figura 40: Modelo Lógico – Usuários

Modelo Logico - vagas

Generated: 13/07/2011 09:49:56
 Server: PostgreSQL Database Server 8.2 (localhost:5432)
 Database: gere_final
 Schema: public

Colunas

Campo	Tipo de dados	Not Null	Primary key?	Default	Comment
codigo	integer	Yes	Yes		
funcao	character varying(150)	Yes	No		
quantidade	integer	Yes	No		
descricao	character varying(150)	Yes	No		
empresa	integer	Yes	No		
status	character(1)	Yes	No		
data_cadastro	date	Yes	No		

Constraints

Campo	Type	Definition	Comment
vagas_pkey	Primary key	(codigo)	
vagas_empresa_fkey	Foreign key	(empresa) REFERENCES empresas (cnpj) MATCH SIMPLE ON UPDATE NO ACTION ON DELETE NO ACTION	
vagas_status_check	Check	(status = 'A'::bpchar OR status = 'T'::bpchar)	

Figura 41: Modelo Lógico – Vagas

Modelo Logico - viagens					
Generated: 13/07/2011 09:50:09					
Server: PostgreSQL Database Server 8.2 (localhost:5432)					
Database: gere_final					
Schema: public					
Colunas					
Campo	Tipo de dados	Not Null	Primary key?	Default	Comment
codigo	integer	Yes	Yes	nextval('viagens_codigo_seq'::regclass)	
polo_origem	integer	Yes	No		
polo_destino	integer	Yes	No		
local_chegada	character varying(150)	Yes	No		
local_partida	character varying(150)	Yes	No		
data_chegada	date	Yes	No		
data_saida	date	Yes	No		
hora_saida	time without time zone	Yes	No		
hora_chegada	time without time zone	Yes	No		
data_reagendamento	date	No	No		
observacoes	character varying(150)	Yes	No		
data_cancelamento	date	No	No		
administrador	integer	No	No		
motorista	integer	No	No		
transporte	integer	No	No		
evento	integer	Yes	No		
Constraints					
Campo	Type	Definition	Comment		
viagens_pkey	Primary key	(codigo)			
viagens_administrador_fkey	Foreign key	(administrador) REFERENCES administradores (codigo) MATCH SIMPLE ON UPDATE NO ACTION ON DELETE NO ACTION			
viagens_evento_fkey	Foreign key	(evento) REFERENCES eventos (codigo) MATCH SIMPLE ON UPDATE NO ACTION ON DELETE NO ACTION			
viagens_motorista_fkey	Foreign key	(motorista) REFERENCES transporte_motorista (codigo) MATCH SIMPLE ON UPDATE NO ACTION ON DELETE NO ACTION			
viagens_polo_destino_fkey	Foreign key	(polo_destino) REFERENCES polos (codigo) MATCH SIMPLE ON UPDATE NO ACTION ON DELETE NO ACTION			
viagens_polo_origem_fkey	Foreign key	(polo_origem) REFERENCES polos (codigo) MATCH SIMPLE ON UPDATE NO ACTION ON DELETE NO ACTION			
viagens_transporte_fkey	Foreign key	(transporte) REFERENCES transportes (codigo) MATCH SIMPLE ON UPDATE NO ACTION ON DELETE NO ACTION			

Figura 42: Modelo Lógico – Viagens

6.3 Modelo Físico : Linguagem de Consulta Estruturada ou Structured Query Language (SQL)

No modelo físico faz-se a modelagem física do modelo de banco de dados. Levam-se em conta as limitações impostas pelo SGBD escolhido e deve ser criado sempre com base nos exemplos de modelagem de dados produzidos no item anterior, modelo lógico.

Iremos utilizar o SGBD PostgreSQL. PostgreSQL é um sistema de banco de dados relacional, que possui diversos recursos, sendo totalmente código aberto, robusto. Este software é totalmente baseado na linguagem SQL, permitindo consultas avançadas, como agregação e agrupamento sem muitas dificuldades.

SQL é, segundo Oliveira (2002, p.18), “um conjunto de comandos de manipulação de banco de dados utilizado para criar e manter a estrutura desse banco de dados”. Desta forma, é possível através de um comando pesquisar dados em diversas tabelas ou atualizar ou excluir diversas linhas de tabelas.

A linguagem SQL, para o autor supracitado, é agrupada segundo sua definição em:

- Data Definition Language (DDL) permite a criação dos componentes do banco de dados, tais como tabelas, índices;
- Data Manipulation Language (DML) permite a manipulação dos dados armazenados no banco de dados;
- Data Query Language permite extrair dados do banco de dados;
- Data Control Language (DCL): provê a segurança interna do banco de dados.

Neste momento, pretende-se apresentar os comandos utilizados para criar as tabelas do Modelo Lógico:

```
CREATE DATABASE GERE
```

```
CREATE TABLE USUARIOS (
```

```
    codigo serial PRIMARY KEY,
```

```
    instituicao varchar(150) not null,
```

```
    rg integer not null,
```

```
    email varchar(15)not null,
```

```
    cpf char(11)not null,
```

```
    data_inativo date ,
```

```
    data_nascimento date not null,
```

```
    login char(6) not null,
```

```
    senha char(6) not null,
```

```
nome varchar (150) not null,  
data_cadastro date not null,  
aprovador integer,  
rua varchar(150) not null,  
numero varchar(10) not null,  
complemento varchar(50) not null,  
bairro varchar(50) not null,  
cidade varchar(50) not null,  
estado char(2) not null,  
cep integer not null,  
foto varchar(150) ,  
tipo_acesso char (2) check (tipo_acesso='AD' OR tipo_acesso= 'AL' OR  
tipo_acesso= 'PO' OR tipo_acesso= 'EP'));  
  
COMMENT ON COLUMN usuarios.tipo_acesso IS ' AD= administrador  
AL=aluno PO= professor e EP empresas';
```

```
CREATE TABLE PROFESSORES(  
  
    siape integer PRIMARY KEY,  
  
    usuario integer not null,  
  
    campus varchar(150) not null,  
  
    FOREIGN KEY (usuario) REFERENCES usuarios(codigo));
```

```
CREATE TABLE ADMINISTRADORES(  

```

```
codigo serial PRIMARY KEY,  
siape integer not null,  
setor varchar(150),  
usuario integer not null,  
FOREIGN KEY (usuario) REFERENCES usuarios(codigo));
```

```
CREATE TABLE ALUNOS(  
matricula integer PRIMARY KEY,  
usuario integer not null,  
FOREIGN KEY (usuario) REFERENCES usuarios(codigo));
```

```
CREATE TABLE EMPRESAS(  
cnpj integer PRIMARY KEY,  
nome_fantasia varchar(150) not null,  
site varchar(150),  
responsavel varchar(150) not null,  
usuario integer not null,  
FOREIGN KEY (usuario) REFERENCES usuarios(codigo));
```

```
CREATE TABLE VAGAS (  
codigo integer PRIMARY KEY,  
funcao varchar(150) not null,  
quantidade integer not null,
```

```
descricao varchar(150) not null,  
empresa integer not null,  
status char (1) not null check(status='A' or status='I'),  
data_cadastro date not null,  
FOREIGN KEY (empresa) REFERENCES empresas(cnpj));
```

```
CREATE TABLE CONCORRE (  
    codigo integer PRIMARY KEY,  
    vaga integer,  
    aluno integer,  
    FOREIGN KEY (vaga) REFERENCES vagas(codigo),  
    FOREIGN KEY (aluno) REFERENCES alunos(matricula));
```

```
CREATE TABLE MODULOS(  
    codigo serial PRIMARY KEY,  
    data_inicio date not null,  
    data_termino date not null,  
    nome varchar(25) not null);
```

```
CREATE TABLE FREQUENTA(  
    codigo serial PRIMARY KEY,  
    modulo integer,  
    aluno integer,
```

```
FOREIGN KEY (modulo) REFERENCES modulos(codigo),  
FOREIGN KEY (aluno) REFERENCES alunos(matricula));
```

```
CREATE TABLE POLOS(  
    codigo serial PRIMARY KEY,  
    observacao varchar(255) ,  
    nome_coordenador varchar(150) not null,  
    nome_polo varchar(25) not null,  
    data_inativo date ,  
    tutor varchar(25) not null,  
    data_cadastro date not null,  
    instituicao varchar(150) not null,  
    rua varchar(150) not null,  
    numero varchar(10) not null,  
    CEP integer not null,  
    bairro varchar(50) not null,  
    cidade varchar(50) not null,  
    estado char(2) not null);
```

```
CREATE TABLE MODULO_POLOS(  
    codigo serial PRIMARY KEY,  
    polo integer not null,  
    modulo integer not null,
```

```
FOREIGN KEY (polo) REFERENCES polos(codigo),  
FOREIGN KEY (modulo) REFERENCES modulos(codigo));
```

```
CREATE TABLE TRANSPORTES(  
    codigo serial PRIMARY KEY,  
    tipo_veiculo char (1) check (tipo_veiculo= 'C' OR tipo_veiculo='O' OR  
    tipo_veiculo='A' OR tipo_veiculo='P'),  
    placa char (15),  
    instituicao varchar (150),  
    exige_motorista char(1) check ( exige_motorista='S' OR  
    exige_motorista='N'));  
  
COMMENT ON COLUMN TRANSPORTES.tipo_veiculo IS 'C= CARRO, O  
=ONIBUS, A= AVIAO,P= VEICULO PARTICULAR';  
  
COMMENT ON COLUMN TRANSPORTES.exige_motorista IS 'S=Sim  
N=Não';
```

```
CREATE TABLE MOTORISTAS(  
    codigo serial PRIMARY KEY,  
    nome varchar(150)not null,  
    cnh integer not null,  
    cpf integer not null,  
    rg integer not null,  
    instituicao varchar(150) not null);
```

```
CREATE TABLE TELEFONES(  
    codigo serial PRIMARY KEY,  
    tipo_telefone char(1) not null check (tipo_telefone='C' or tipo_telefone='F'),  
    numero_telefone integer not null,  
    ddd char(2) not null,  
    cod_polo integer,  
    cod_usuario integer,  
    cod_motorista integer,  
    FOREIGN KEY (cod_polo) REFERENCES polos(codigo),  
    FOREIGN KEY (cod_usuario) REFERENCES usuarios(codigo),  
    FOREIGN KEY (cod_motorista) REFERENCES motoristas(codigo));  
COMMENT ON COLUMN telefones.tipo_telefone IS 'C= CELULAR, F=FIXO';
```

```
CREATE TABLE DISCIPLINAS (  
    codigo serial PRIMARY KEY,  
    carga_horaria varchar(15) not null,  
    nome varchar(150) not null,  
    modulo integer not null,  
    professor integer not null,  
    FOREIGN KEY (modulo) REFERENCES modulos(codigo),  
    FOREIGN KEY (professor) REFERENCES professores(siape));
```

```
CREATE TABLE PLANO_ENSINO(  
    codigo serial PRIMARY KEY,  
    nome varchar(150) not null,  
    modulo integer not null,  
    professor integer not null,  
    FOREIGN KEY (modulo) REFERENCES modulos(codigo),  
    FOREIGN KEY (professor) REFERENCES professores(siape));
```

```
codigo serial PRIMARY KEY,  
data date not null,  
nome_materia varchar(150) not null,  
descricao_materia varchar(150),  
professor integer,  
disciplina integer,  
FOREIGN KEY (disciplina) REFERENCES disciplinas(codigo),  
FOREIGN KEY (professor) REFERENCES professores(siape));
```

```
CREATE TABLE TIPO_EVENTOS (
```

```
    codigo serial PRIMARY KEY,  
    nome varchar(150) not null);
```

```
CREATE TABLE EVENTOS (
```

```
    codigo serial PRIMARY KEY,  
    cancelamento char(1) check (cancelamento='S' or cancelamento='N'),  
    motivo varchar(150),  
    professor integer not null,  
    data_cancelamento date,  
    data_termino date not null,  
    data_inicio date not null,  
    modulo integer not null,  
    tipo_evento integer not null,
```

```
disciplina integer not null,  
  
FOREIGN KEY (modulo) REFERENCES modulos(codigo),  
  
FOREIGN KEY (tipo_evento) REFERENCES tipo_eventos(codigo),  
  
FOREIGN KEY (disciplina) REFERENCES disciplinas(codigo));  
  
COMMENT ON COLUMN EVENTOS.cancelamento IS 'S= Sim, N =Não';
```

```
CREATE TABLE PARTICIPA(  
  

```

```
    codigo serial PRIMARY KEY,  
  
    evento integer,  
  
    professor integer,  
  
    FOREIGN KEY (evento) REFERENCES eventos(codigo),  
  
    FOREIGN KEY (professor) REFERENCES professores(siape));
```

```
CREATE TABLE TRANSPORTE_MOTORISTA(  
  

```

```
    codigo serial PRIMARY KEY,  
  
    transporte integer not null,  
  
    motorista integer not null,  
  
    data_transporte date not null,  
  
    hora_transporte time not null,  
  
    FOREIGN KEY (transporte) REFERENCES transportes(codigo),  
  
    FOREIGN KEY (motorista) REFERENCES motoristas(codigo));
```

```
CREATE TABLE VIAGENS(  
  

```

codigo serial PRIMARY KEY,
polo_origem integer not null,
polo_destino integer not null,
local_chegada varchar(150) not null,
local_partida varchar(150) not null,
data_chegada date not null,
data_saida date not null,
hora_saida time not null,
hora_chegada time not null,
data_reagendamento date,
observacoes varchar(150) not null,
data_cancelamento date ,
administrador integer,
motorista integer,
transporte integer,
evento integer not null,
FOREIGN KEY (motorista) REFERENCES transporte_motorista(codigo),
FOREIGN KEY (transporte) REFERENCES transportes(codigo),
FOREIGN KEY (administrador) REFERENCES administradores(codigo),
FOREIGN KEY (evento) REFERENCES eventos(codigo),
FOREIGN KEY (polo_origem) REFERENCES polos(codigo),
FOREIGN KEY (polo_destino) REFERENCES polos(codigo));

```
CREATE TABLE DIARIAS(  
    codigo serial PRIMARY KEY,  
    protocolo char(50),  
    quantidade numeric(2,1) not null,  
    viagem integer not null,  
    professor integer not null,  
    FOREIGN KEY (viagem) REFERENCES viagens(codigo),  
    FOREIGN KEY (professor) REFERENCES professores(siape));
```

```
CREATE TABLE TIPO(  
    codigo serial PRIMARY KEY,  
    descricao varchar(150) not null);
```

```
CREATE TABLE CLASSIFICACAO(  
    codigo serial PRIMARY KEY,  
    descricao varchar(150) not null);
```

```
CREATE TABLE PATRIMONIO(  
    numero integer PRIMARY KEY,  
    codigo serial not null,  
    respopnstavel varchar (150) not null,  
    preco numeric(5,2) not null,  
    setor varchar (150) not null,
```

```
data_compra date not null,  
  
descricao varchar (150) not null,  
  
situacao varchar(1) not null check (situacao = 'A' or situacao = 'B' or situacao  
= 'E'),  
  
tipo integer,  
  
classificacao integer,  
  
administrador integer,  
  
FOREIGN KEY (tipo) REFERENCES tipo(codigo),  
  
FOREIGN KEY (classificacao) REFERENCES classificacao(codigo),  
  
FOREIGN KEY (administrador) REFERENCES administradores(codigo));
```

```
CREATE TABLE PEDIDO(
```

```
    codigo serial PRIMARY KEY,  
  
    preco numeric(5,2) not null,  
  
    descricao varchar (150) not null,  
  
    administrador integer,  
  
    FOREIGN KEY (administrador) REFERENCES administradores(codigo));
```

```
CREATE TABLE PEDIDO_PATRIMONIO(
```

```
    codigo serial PRIMARY KEY,  
  
    quantidade integer not null,  
  
    data_emissao integer not null,  
  
    patrimonio integer,
```

pedido integer,

FOREIGN KEY (patrimonio) REFERENCES patrimonio(numero),

FOREIGN KEY (pedido) REFERENCES pedido(codigo));

7.INTERFACE DO SISTEMA

A interface do sistema constitui-se a comunicação com o usuário. Conforme Carrion (2008, p 67) “a arte se define como expressão. A filosofia da arte consiste em desdobrar tudo o que ela é como expressão”. Neste contexto, o projeto da interface levou em consideração a funcionalidade e a perspectiva estética na construção do design.

Assim, pretendeu-se articular os módulos do sistema apresentando elementos visuais e a funcionalidade do design final de maneira coesa. Desta forma, apresentando, aos usuários finais do sistema, a articulação da satisfação do conteúdo e do design.

7.1 Nome e Logotipo

O nome *Gère – Gerência e Integração Acadêmica* foi dado ao sistema por se tratar de um sistema administrativo. A palavra *gère* vem do francês e significa *gerência*, termo que está diretamente relacionado com a administração. A língua francesa foi escolhida fazendo referência à nacionalidade de Jules Henri Fayol, um estudioso nascido no século XIX, responsável por idealizar a teoria clássica da administração, caracterizada pela ênfase na estrutura organizacional, pela visão do homem econômico e pela busca da máxima eficiência.

O logotipo é a forma única de representação gráfica da uma marca ou produto. Para o sistema *Gère* foi criado um logotipo, possuindo tipografia e elemento gráfico. O elemento gráfico é um conjunto de cubos, onde os três cubos maiores centrais representam o sistema e os três cubos menores e mais externos, representam os usuários principais, alunos, professores e empresas, tendo dois cubos em contraste de cor, representando um processo de interação entre usuário e sistema.



Figura 43: Leituras do Logotipo (marcação)

Para a tipografia do logotipo foram usadas duas fontes diferentes em caixa alta, para a palavra Gère foi usada a fonte Barmeno® Extra Bold e para o texto de apoio foi usado a fonte Calibri.

Após alguns testes com outras cores, foram utilizadas o laranja e o cinza por possuírem bom contraste, cada uma com outras duas variações, em notação hexadecimal são elas: laranjas #FF8300, #E26000 e #FEBD00 e cinzas #838BA0, #545F7D e #393945. No texto complementar foi utilizado um barra em tons de cinza como proteção, para que mesmo em tamanho menor não ocorra a perda da leitura.

7.2 Interface do Sistema

A interface do sistema foi criada visando principalmente a usabilidade do usuário, sendo simples e intuitiva, tendo uma aparência *clean*. Foi utilizada a cor azul em todo sistema por ser uma cor harmônica e que faz referência à segurança e confiabilidade.

A página inicial é a página de acesso do sistema, possui o logotipo do sistema e abaixo o logotipo da instituição que está utilizando o sistema, também possui uma breve descrição do sistema e um box de login e senha onde o usuário deve escolher também qual seu tipo de acesso: aluno, professor, empresa ou admin.



Figura 44: Tela "Index" do sistema

As páginas internas seguem uma unidade, possuem na lateral esquerda um caixa em azul onde encontram-se o logo do sistema, as informações do usuário e o sub-menu principal diferente para cada tipo de acesso. Abaixo a miniatura do cronograma em forma de calendário com destaque nas datas que possuem eventos marcados e logo em seguida um box com as 3 últimas notícias.

Na parte superior do layout foi colocada uma barra de busca para o usuário ter acesso rápido ao que está procurando e abaixo o dia, mês e ano. Em seguida está o menu global, padrão para todos os tipos de acesso, com quatro botões: home, contato, ajuda e logout. Mais abaixo a *breadcrumb* mostrando ao usuário em que seção do sistema está. Seguindo a estrutura vertical temos o logotipo de cada seção, com o nome da seção e um elemento gráfico que a representa. Depois temos o sub-menu secundário com os botões que variam conforme o item que está selecionado no sub-menu principal. Abaixo temos a área de interesse onde tudo que é solicitado ao sistema é mostrado nessa área, ela varia de altura conforme a quantidade de informações que o sistema exibirá. E no rodapé o layout possui o menu global para acesso rápido, para seções com várias rolagens.

Sexta-feira | 27 de maio de 2011

[Home](#) | [Contato](#) | [Ajuda](#) | [Logout](#)

você está em: [alunos](#) > [desempenho](#) > [consultar](#)

Desempenho

consultar

matéria	1ª avaliação	2ª avaliação	conceito final	frequência	enviar email ao prof.
PBDR	B	B	B	90%	
LPW	C	C	C	100%	
DPS	A	A	A	85%	
APS	D	D	D	100%	
ING	B	B	B	95%	
RED	A	A	A	100%	
RHT	A	A	A	95%	
RED	A	A	A	100%	

cronograma

JUNHO DE 2011

Dom	Seg	Ter	Qua	Qui	Sex	Sab
29	30	31	1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	1	2

notícias

24/06/2011
- aula de LPW remarcada

24/06/2011
- aula de LPW remarcada

24/06/2011
- aula de LPW remarcada

ver últimas notícias

Copyright © 2011 Instituto Federal Sul-rio-grandense

[Home](#) | [Contato](#) | [Ajuda](#) | [Logout](#)

Figura 45: Tela interna do Módulo Aluno



GÈRE
GERÊNCIA E INTEGRAÇÃO ACADÊMICA

Busca:

Sexta-feira | 27 de maio de 2011



Cronograma de Aulas

consultar



alterar foto

Bem-vindo
Fulano da Silva

Curso Sup. de Tec. para Internet
E-mail: aluno@gmail.com
Função: Aluno

Cronograma de Aulas

Desempenho

Currículo

Vagas

Perfil

JUNHO DE 2011

Dom	Seg	Ter	Qua	Qui	Sex	Sab
29	30	31	1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	1	2

04/06/2011 - Atividades do dia

Avaliação - Profa. Michele
Disciplina: PBDR
Pólo: Picada Café
Módulo 4
Horário da aula: 08h

Web Conferência - Profa. Simone
Disciplina: APS
Pólo: Picada Café
Módulo 4
Horário da conferência: 14h

data	matéria	módulo	professor	enviar email aos professores
04/06/2011	PBDR	Módulo 4	Fulano	
12/06/2011	LPW	Módulo 4	Fulano	
25/06/2011	PBDR	Módulo 4	Beltrano	
02/07/2011	LPW	Módulo 4	Beltrano	
04/06/2011	PBDR	Módulo 4	Fulano	
12/06/2011	LPW	Módulo 4	Ciclano	
25/06/2011	PBDR	Módulo 4	Fulano	
02/07/2011	LPW	Módulo 4	Beltrano	
04/06/2011	PBDR	Módulo 4	Ciclano	
12/06/2011	LPW	Módulo 4	Fulano	
25/06/2011	PBDR	Módulo 4	Fulano	
02/07/2011	LPW	Módulo 4	Ciclano	
04/06/2011	PBDR	Módulo 4	Beltrano	
12/06/2011	LPW	Módulo 4	Beltrano	
25/06/2011	PBDR	Módulo 4	Fulano	
02/07/2011	LPW	Módulo 4	Ciclano	
04/06/2011	PBDR	Módulo 4	Beltrano	
12/06/2011	LPW	Módulo 4	Fulano	
25/06/2011	PBDR	Módulo 4	Fulano	
02/07/2011	LPW	Módulo 4	Ciclano	
04/06/2011	PBDR	Módulo 4	Ciclano	
12/06/2011	LPW	Módulo 4	Fulano	

cronograma

JUNHO DE 2011

Dom	Seg	Ter	Qua	Qui	Sex	Sab
29	30	31	1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	1	2

noticias

24/06/2011
- aula de LPW remarcada

24/06/2011
- aula de LPW remarcada

24/06/2011
- aula de LPW remarcada

ver últimas noticias

1 2 3 4 >



Copyright © 2011 Instituto Federal Sul-rio-grandense
Home | Contato | Ajuda | Logout

Figura 46: Tela Interna cronograma do Módulo Alunos

Sexta-feira | 27 de maio de 2011

Home | Contato | Ajuda | Logout

você está em: [professores](#) > [cronograma de viagens](#)



Viagens

Cronograma de Viagens



Bem-vindo [alterar foto](#)
Fulano da Silva

Curso Sup. de Tec. para Internet
E-mail: professor@ifsul.edu.br
Função: Professor

- [Cronograma de Aulas](#)
- [Cronograma de Viagens](#)
- [Plano de Ensino](#)
- [Disciplinas](#)
- [Perfil](#)

← JUNHO DE 2011 →

Dom	Seg	Ter	Qua	Qui	Sex	Sab
29	30	31	1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	1	2

04/06/2011 - Campus Passo Fundo

Matéria: PBDR

Horário/local de saída:
- Campus Pelotas as 11h

>Lorem ipsum dolor sit amet, consectetur as adipiscing elit, sed do eiusmod tempor. Em ipsum dolor sit amet, consectetur aseg adipiscing elit, sed do eiusmod tempor.

[solicitar cancelamento](#)

data	matéria	local/horário de saída	cidade	enviar email aos alunos
04/06/2011	PBDR	Campus Pelotas - 11h	Sta. Vitória	
12/06/2011	LPW	Campus Pelotas - 8h	Charqueadas	
25/06/2011	PBDR	Campus Pelotas - 11h	Camaquã	
02/07/2011	LPW	Campus Pelotas - 8h	Bagé	
04/06/2011	PBDR	Campus Pelotas - 11h	Sta. Vitória	
12/06/2011	LPW	Campus Pelotas - 8h	Charqueadas	
25/06/2011	PBDR	Campus Pelotas - 11h	Camaquã	
02/07/2011	LPW	Campus Pelotas - 8h	Bagé	
04/06/2011	PBDR	Campus Pelotas - 11h	Sta. Vitória	
12/06/2011	LPW	Campus Pelotas - 8h	Charqueadas	
25/06/2011	PBDR	Campus Pelotas - 11h	Camaquã	
02/07/2011	LPW	Campus Pelotas - 8h	Bagé	
04/06/2011	PBDR	Campus Pelotas - 11h	Sta. Vitória	
12/06/2011	LPW	Campus Pelotas - 8h	Charqueadas	
25/06/2011	PBDR	Campus Pelotas - 11h	Camaquã	
02/07/2011	LPW	Campus Pelotas - 8h	Bagé	
04/06/2011	PBDR	Campus Pelotas - 11h	Sta. Vitória	
12/06/2011	LPW	Campus Pelotas - 8h	Charqueadas	

1 2 3 4 >

cronograma

← JUNHO DE 2011 →

Dom	Seg	Ter	Qua	Qui	Sex	Sab
29	30	31	1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	1	2

notícias

24/06/2011
- aula de LPW remarcada

24/06/2011
- aula de LPW remarcada

24/06/2011
- aula de LPW remarcada

[ver últimas notícias](#)

Copyright © 2011 Instituto Federal Sul-rio-grandense

Home | Contato | Ajuda | Logout





Figura 47-Tela Interna cronograma de viagens do Módulo Viagens



GÈRE
GERÊNCIA E INTEGRAÇÃO ACADÊMICA

Busca:

Sexta-feira | 27 de maio de 2011

Home | Contato | Ajuda | Logout

você está em: [viagens](#) > [cronograma de viagens](#) > [Cadastrar](#)



Viagens

Cadastrar | Consultar | Alterar



alterar foto

Bem-vindo
Fulano da Silva

Curso Sup. de Tec. para Internet
E-mail: professor@ifsul.edu.br
Função: Professor

Cronograma de Aulas

Cronograma de Viagens

Plano de Ensino

Disciplinas

Enviar Email

Perfil

JUNHO DE 2011

Dom	Seg	Ter	Qua	Qui	Sex	Sab
29	30	31	1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	1	2

04/06/2011 - Campus Passo Fundo

Matéria: PBDR

Horário/local de saída:
- Campus Pelotas as 11h

Aula presencial sobre Modelo conceitual ministrada pela Professora Michele Schmitz

Transporte: Carro com motorista do IFSUL

Cadastrar Viagens

Data Evento: **12/06/2011** Professor: **Maria da Silva** Modulo: **Modulo II**

Disciplina: **PBR-Projeto de Banco de dados**

Preencha o formulário para cadastrar viagem:

Polo de origem: Polo de destino:

Local de Partida:

Data Partida: Hora saída:

Tipo Transporte: Motorista:

Local de Chegada:

Data Chegada: Hora chegada:

cronograma

JUNHO DE 2011

Dom	Seg	Ter	Qua	Qui	Sex	Sab
29	30	31	1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	1	2

notícias

24/06/2011
- aula de LPW remarcada

24/06/2011
- aula de LPW remarcada

24/06/2011
- aula de LPW remarcada

ver últimas notícias

Copyright © 2011. Instituto Federal Sul-rio-grandense





Home | Contato | Ajuda | Logout

Figura 48: Tela interna cadastrar viagem do Módulo Viagens



GÈRE
GESTÃO E INTEGRAÇÃO ACADÊMICA

Busca:

Sexta-feira | 27 de maio de 2011

Home | Contato | Ajuda | Logout

você está em: viagens > cronograma de viagens > consultar



Fulano da Silva

Bem-vindo

Curso Sup. de Tec. para Internet
E-mail: professor@ifrsul.edu.br
Função: Professor

- Cronograma de Aulas
- Cronograma de Viagens**
- Plano de Ensino
- Disciplinas
- Enviar Email
- Perfil



Viagens

Cadastrar | Consultar | Alterar

JUNHO DE 2011

Dom	Seg	Ter	Qua	Qui	Sex	Sab
29	30	31	1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	1	2

04/06/2011 - Campus Passo Fundo

Matéria: PBDR

Horário/local de saída:
- Campus Pelotas as 11h

Aula presencial sobre Modelo conceitual ministrada pela Professora Michele Schmitd
Transporte: Carro com motorista do IFSUL

Cronograma de Viagens de 01/06/2011 à 30/06/2011

data	matéria	local/horário de saída	cidade	enviar email aos alunos
04/06/2011	PBDR	Campus Pelotas - 11h	Sta. Vitória	✉
12/06/2011	LPW	Campus Pelotas - 8h	Charqueadas	✉
25/06/2011	PBDR	Campus Pelotas - 11h	Camaquã	✉
02/07/2011	LPW	Campus Pelotas - 8h	Bagé	✉
04/06/2011	PBDR	Campus Pelotas - 11h	Sta. Vitória	✉
12/06/2011	LPW	Campus Pelotas - 8h	Charqueadas	✉
25/06/2011	PBDR	Campus Pelotas - 11h	Camaquã	✉
02/07/2011	LPW	Campus Pelotas - 8h	Bagé	✉
04/06/2011	PBDR	Campus Pelotas - 11h	Sta. Vitória	✉
12/06/2011	LPW	Campus Pelotas - 8h	Charqueadas	✉

1 2 3 4 >

Período de: à:

Polo: Tipo Transporte:

SIAPE: Nº de protocolo:

cronograma

JUNHO DE 2011

Dom	Seg	Ter	Qua	Qui	Sex	Sab
29	30	31	1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	1	2

noticias

- 24/06/2011 - aula de LPW remarcada
- 24/06/2011 - aula de LPW remarcada
- 24/06/2011 - aula de LPW remarcada

[ver últimas notícias](#)

Copyright © 2011 Instituto Federal Sul-rio-grandense

Home | Contato | Ajuda | Logout

Figura 49: Tela interna consultar viagem do Módulo Viagens



GÈRE
GERÊNCIA DE INTEGRAÇÃO ACADÊMICA

Busca:

Sexta-feira | 27 de maio de 2011

Home | Contato | Ajuda | Logout

você está em: [viagens](#) > [cronograma de viagens](#) > [Alterar](#)



Viagens

Cadastrar
Consultar
Alterar



alterar foto

Bem-vindo
Fulano da Silva

Curso Sup. de Tec. para Internet
E-mail: professor@ifsul.edu.br
Função: Professor

Cronograma de Aulas

Cronograma de Viagens

Plano de Ensino

Disciplinas

Enviar Email

Perfil

JUNHO DE 2011

Dom	Seg	Ter	Qua	Qui	Sex	Sab
29	30	31	1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	1	2

04/06/2011 - Campus Passo Fundo

Matéria: PBDR

Horário/local de saída:
- Campus Pelotas as 11h

Aula presencial sobre Modelo conceitual ministrada pela Professora Michele Schmitd
Transporte: Carro com motorista do IFSUL

Alterar Viagens

Data Evento: **12/06/2011** Professor: **Maria da Silva** Modulo: **Modulo II**

Disciplina: **PBR-Projeto de Banco de dados** Codigo da Viagem: **9800/11**

Altere os dados necessários:

Polo de origem: Polo de destino:

Local de Partida:

Data Partida: Hora saída:

Tipo Transporte: Motorista:

Local de Chegada:

Data Chegada: Hora chegada:

Nº de Protocolo: Total Diárias:

Motivo de cancelamento: Motivo Re-agendamento:

cronograma

JUNHO DE 2011

Dom	Seg	Ter	Qua	Qui	Sex	Sab
29	30	31	1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	1	2

noticias

24/06/2011
- aula de LPW remarcada

24/06/2011
- aula de LPW remarcada

24/06/2011
- aula de LPW remarcada

ver últimas notícias

Copyright © 2011 Instituto Federal Sul-rio-grandense

Home | Contato | Ajuda | Logout

Figura 50:Tela interna alterar viagem do Módulo Viagens



GÈRE
GERÊNCIA E INTERMEDIÇÃO ACADÊMICA

Busca:

Sexta-feira | 27 de maio de 2011

Home | Contato | Ajuda | Logout

você está em: [admin](#) > [cronograma de aulas](#)



Bem-vindo
Fulano da Silva

Administrador Gère
E-mail: fulano.admin@ifsul.edu.br
Função: Administrador

- [Cronograma de Aulas](#)
- [Cronograma de Viagens](#)
- [Currículos](#)
- [Vagas](#)
- [Patrimônio](#)
- [Perfil](#)

Cronograma de Aulas

Cadastrar | Pedido | Consultar | Alterar

04/06/2011 - Solicitações do dia

Aula Presencial 2 - Profa. Michele
Disciplina: PBDR
Pólo: Picada Café
Módulo 4
De: 08h - 15/07/2011
Para: 10h - 17/07/2011
Motivo:
Ausente por problemas de saúde.

JUNHO DE 2011

Dom	Seg	Ter	Qua	Qui	Sex	Sab
29	30	31	1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	1	2

data	matéria	módulo	professor	alterar
04/06/2011	PBDR	Módulo 4	Fulano	
12/06/2011	LPW	Módulo 4	Fulano	
25/06/2011	PBDR	Módulo 2	Beltrano	
02/07/2011	LPW	Módulo 1	Beltrano	
04/06/2011	PBDR	Módulo 1	Fulano	
12/06/2011	LPW	Módulo 2	Ciclano	
25/06/2011	PBDR	Módulo 6	Fulano	
02/07/2011	LPW	Módulo 1	Beltrano	
04/06/2011	PBDR	Módulo 2	Ciclano	
12/06/2011	LPW	Módulo 6	Fulano	
25/06/2011	PBDR	Módulo 1	Fulano	
02/07/2011	LPW	Módulo 6	Ciclano	
04/06/2011	PBDR	Módulo 2	Beltrano	
12/06/2011	LPW	Módulo 1	Beltrano	
25/06/2011	PBDR	Módulo 1	Fulano	
02/07/2011	LPW	Módulo 2	Ciclano	
04/06/2011	PBDR	Módulo 1	Beltrano	
12/06/2011	LPW	Módulo 6	Fulano	
25/06/2011	PBDR	Módulo 1	Fulano	
02/07/2011	LPW	Módulo 2	Ciclano	
04/06/2011	PBDR	Módulo 2	Ciclano	
12/06/2011	LPW	Módulo 6	Fulano	

notícias

24/06/2011
- aula de LPW remarcada

24/06/2011
- aula de LPW remarcada

24/06/2011
- aula de LPW remarcada

[ver últimas notícias](#)

1 2 3 4 >



Copyright © 2011 Instituto Federal Sul-rio-grandense

Home | Contato | Ajuda | Logout

Figura 51: Tela interna cronograma do Módulo Cronograma



GÈRE
GERÊNCIA E INTEGRAÇÃO ACADÊMICA

Busca:

Sexta-feira | 27 de maio de 2011

Home
Contato
Ajuda
Logout

você está em: [alunos](#) > [cronogramas de aulas](#)



Cronograma de Aulas



alterar foto

Bem-vindo
Fulano da Silva

Curso Sup. de Tec. para Internet
E-mail: aluno@gmail.com
Função: Aluno

Cronograma de Aulas

Desempenho

Currículo

Vagas

Perfil

JUNHO DE 2011

Dom	Seg	Ter	Qua	Qui	Sex	Sab
29	30	31	1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	1	2

04/06/2011 - Atividades do dia

Avaliação - Profa. Michele
Disciplina: PBDR
Pólo: Picada Café
Módulo 4
Horário da aula: 08h

Web Conferência - Profa. Simone
Disciplina: APS
Pólo: Picada Café
Módulo 4
Horário da conferência: 14h

data	matéria	módulo	professor	enviar email aos professores
04/06/2011	PBDR	Módulo 4	Fulano	
12/06/2011	LPW	Módulo 4	Fulano	
25/06/2011	PBDR	Módulo 4	Beltrano	
02/07/2011	LPW	Módulo 4	Beltrano	
04/06/2011	PBDR	Módulo 4	Fulano	
12/06/2011	LPW	Módulo 4	Ciclano	
25/06/2011	PBDR	Módulo 4	Fulano	
02/07/2011	LPW	Módulo 4	Beltrano	
04/06/2011	PBDR	Módulo 4	Ciclano	
12/06/2011	LPW	Módulo 4	Fulano	
25/06/2011	PBDR	Módulo 4	Fulano	
02/07/2011	LPW	Módulo 4	Ciclano	
04/06/2011	PBDR	Módulo 4	Beltrano	
12/06/2011	LPW	Módulo 4	Fulano	
25/06/2011	PBDR	Módulo 4	Fulano	
02/07/2011	LPW	Módulo 4	Ciclano	
04/06/2011	PBDR	Módulo 4	Ciclano	
12/06/2011	LPW	Módulo 4	Fulano	
25/06/2011	PBDR	Módulo 4	Fulano	
02/07/2011	LPW	Módulo 4	Ciclano	
04/06/2011	PBDR	Módulo 4	Ciclano	
12/06/2011	LPW	Módulo 4	Fulano	

cronograma

JUNHO DE 2011

Dom	Seg	Ter	Qua	Qui	Sex	Sab
29	30	31	1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	1	2

notícias

24/06/2011
- aula de LPW remarcada

24/06/2011
- aula de LPW remarcada

24/06/2011
- aula de LPW remarcada

ver últimas notícias

1 2 3 4 >



Copyright © 2011 Instituto Federal Sul-rio-grandense

Home
Contato
Ajuda
Logout

Figura 52: Tela interna cronograma – alunos do Módulo Cronograma



GÈRE
GESTÃO E INTEGRAÇÃO ACADÊMICA

Busca:

Sexta-feira | 27 de maio de 2011

Home | Contato | Ajuda | Logout

você está em: [professores](#) > [cronogramas de aulas](#)



Cronograma de Aulas



alterar foto

Bem-vindo
Fulano da Silva

Curso Sup. de Tec. para Internet
E-mail: professor@ifsul.edu.br
Função: Professor

- Cronograma de Aulas
- Cronograma de Viagens
- Plano de Ensino
- Disciplina
- Perfil

JUNHO DE 2011

Dom	Seg	Ter	Qua	Qui	Sex	Sab
29	30	31	1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	1	2

04/06/2011 - Atividades do dia

Aula Presencial 2 - Profa. Michele
Disciplina: PBDR
Pólo: Picada Café
Módulo 4
Horário da Saída: 08h

Web Conferência - Profa. Simone
Disciplina: APS
Pólo: Picada Café
Módulo 2
Horário da conferência: 14h

data	matéria	módulo	professor	enviar email aos alunos
04/06/2011	PBDR	Módulo 4	Fulano	<input type="button" value="✉"/>
12/06/2011	LPW	Módulo 4	Fulano	<input type="button" value="✉"/>
25/06/2011	PBDR	Módulo 2	Beltrano	<input type="button" value="✉"/>
02/07/2011	LPW	Módulo 1	Beltrano	<input type="button" value="✉"/>
04/06/2011	PBDR	Módulo 1	Fulano	<input type="button" value="✉"/>
12/06/2011	LPW	Módulo 2	Ciclano	<input type="button" value="✉"/>
25/06/2011	PBDR	Módulo 6	Fulano	<input type="button" value="✉"/>
02/07/2011	LPW	Módulo 1	Beltrano	<input type="button" value="✉"/>
04/06/2011	PBDR	Módulo 2	Ciclano	<input type="button" value="✉"/>
12/06/2011	LPW	Módulo 6	Fulano	<input type="button" value="✉"/>
25/06/2011	PBDR	Módulo 1	Fulano	<input type="button" value="✉"/>
02/07/2011	LPW	Módulo 6	Ciclano	<input type="button" value="✉"/>
04/06/2011	PBDR	Módulo 2	Beltrano	<input type="button" value="✉"/>
12/06/2011	LPW	Módulo 1	Beltrano	<input type="button" value="✉"/>
25/06/2011	PBDR	Módulo 1	Fulano	<input type="button" value="✉"/>
02/07/2011	LPW	Módulo 2	Ciclano	<input type="button" value="✉"/>
04/06/2011	PBDR	Módulo 1	Beltrano	<input type="button" value="✉"/>
12/06/2011	LPW	Módulo 6	Fulano	<input type="button" value="✉"/>
25/06/2011	PBDR	Módulo 1	Fulano	<input type="button" value="✉"/>
02/07/2011	LPW	Módulo 2	Ciclano	<input type="button" value="✉"/>
04/06/2011	PBDR	Módulo 2	Ciclano	<input type="button" value="✉"/>
12/06/2011	LPW	Módulo 6	Fulano	<input type="button" value="✉"/>

cronograma

JUNHO DE 2011

Dom	Seg	Ter	Qua	Qui	Sex	Sab
29	30	31	1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	1	2

notícias

24/06/2011
- aula de LPW remarcada

24/06/2011
- aula de LPW remarcada

24/06/2011
- aula de LPW remarcada

[ver últimas notícias](#)

1 2 3 4 >



Copyright © 2011 Instituto Federal Sul-rio-grandense

Home | Contato | Ajuda | Logout

Figura 53: Tela interna professores cronograma do Módulo Professor

Busca:

Sexta-feira | 27 de maio de 2011

Home | Contato | Ajuda | Logout

você está em: [admin](#) > [patrimônio](#) > [cadastrar](#)

Patrimônio

Cadastrar | Pedido | Consultar | Alterar

Buscar por Classificar por

Setor:

Descrição:

Número:

Responsável:

Valor:

Sala:

Situação:

Data da Compra:

Tipo:



GÈRE
GERÊNCIA DE INTEGRAÇÃO ACADÊMICA



[alterar foto](#)

Bem-vindo
Fulano da Silva

Administrador Gêre
E-mail: fulano.admin@ifsul.edu.br
Função: Administrador

- [Cronograma de Aulas](#)
- [Cronograma de Viagens](#)
- [Currículos](#)
- [Vagas](#)
- [Patrimônio](#)
- [Perfil](#)

notícias

24/06/2011
- aula de LPW remarcada

24/06/2011
- aula de LPW remarcada

24/06/2011
- aula de LPW remarcada

[ver últimas notícias](#)

Copyright © 2011 Instituto Federal Sul-rio-grandense

Home | Contato | Ajuda | Logout



Figura 56: Tela interna cadastrar do Módulo Patrimônio

Busca:

Sexta-feira | 27 de maio de 2011

Home | Contato | Ajuda | Logout

Patrimônio

voce está em: [admin](#) > [patrimonio](#) > [cadastrar](#)

Cadastrar | **Pedido** | Consultar | Alterar




[alterar foto](#)

Bem-vindo
Fulano da Silva

Administrador Gêre
E-mail: fulano.admin@ifsul.edu.br
Função: Administrador

- [Cronograma de Aulas](#)
- [Cronograma de Viagens](#)
- [Currículos](#)
- [Vagas](#)
- [Patrimônio](#)
- [Perfil](#)

noticias

- 24/06/2011
- aula de LPW remarcada
- 24/06/2011
- aula de LPW remarcada
- 24/06/2011
- aula de LPW remarcada

[ver últimas notícias](#)

Nome:

Descrição:

Quantidade:

Valor Lote:

Valor Unitário:





Copyright © 2011 Instituto Federal Sul-rio-grandense

Home | Contato | Ajuda | Logout

Figura 57: Tela interna pedir do Módulo Patrimônio

8.CONCLUSÃO

O sistema Gère irá contribuir para melhor administração das operações realizadas nos polos que atendem ao curso de TSlaD. Viabilizará também a comunicação entre professores, alunos permitindo a integração em um só local de informações como aulas, planos de ensino, frequências avaliações e datas de eventos que ocorrerão nos polos. Para os alunos a equipe desenvolvedora atendendo a necessidade do cliente criou o modulo de empresas que permitira ao alunos o conhecimento das necessidades do mercado de trabalho e ás empresas a possibilidade de conhecer novos profissionais formados pela instituição.

Com a elaboração do sistema os alunos do módulo II conseguiram aplicar todos os conhecimentos teóricos adquiridos durante o semestre. Não podemos deixar de salientar ainda que, mais do que a teoria, a prática mostrou á equipe o quão difícil é a elaboração de um sistema, principalmente quando precisamos entender a necessidade usuário e transformar esta necessidade em um sistema que realmente atenda ao cliente.

Aprendemos ainda lições valiosas de como a comunicação entre as equipes é fundamental, o trabalho em grupo é muito mais proveitoso que o individual e que horas sem dormir são horas empenhadas em um trabalho nos trará orgulho.

Por fim entendemos que nenhum conhecimento teórico é suficientemente, útil se não soubermos como aplicá-lo. Certamente concluiremos o segundo semestre carregando o lema de que não há sucesso sem trabalho, esforço luta e dedicação.

REFERÊNCIAS BIBLIOGRÁFICAS

WIKIPÉDIA. Disponível em: <http://pt.wikipedia.org/wiki/Teoria_de_sistemas>
Acessado em jun de 2011.

BAIXAKI. Disponível em: <<http://www.baixaki.com.br/linux/download/postgresql.htm>>
Acessado em jun de 2011.

BORTOLIN, Sérgio A. M. **Ferramenta Case**. Disponível em
<<http://www.al.urcamp.tche.br/infocamp/edicoes/nov05/Ferramentas%20Case.pdf>>Acessad
o em jun 2011.

CARRION ,wellington. **Design para Web Designers**. Rio de Janeiro: Brasport, 2008.

CHIAVENATO, Idalberto. **Introdução à teoria geral da administração**: uma visão abrangente da moderna administração das organizações. 7. ed. rev. e atual. - Rio de Janeiro: Elsevier, 2003.

COSTA, Rogério Luis da. **SQL**: Guia prático. 2 ed. Rio de Janeiro: Brasport, 2006.

DAMASCENO, Anielle. **Web Design, Teoria e Prática**. Florianópolis: Ed. Visual Books, 2004.

HEUSER, Carlos Alberto. **Projeto de bancos de dados**. 4 ed. Porto alegre: sagra luzzatto.

LAUDON, Kenneth C.; LAUDON, Jane Price. **Sistemas de informações gerenciais**. Tradução Thelma Guimarães; 7. ed. - São Paulo: Pearson Prentice Hall, 2007.

MARCO, Marina. **Projeto Banco de dados**. Disponível em:
<<http://pt.scribd.com/doc/6795108/Projeto-de-Banco-de-Dados>>. Acessado em jun de 2011.

OLIVEIRA, Celso Henrique Poderoso de, **SQL**: Curso Prático. São Paulo: Novatec,2002.

REZENDE, Dennis. Engenharia de software e sistemas de informação. 3ed rev. E ampl. Rio de Janeiro: Brasport, 2005.

WILLIAMS, Robin. **Design para quem não é designer**: noções básicas de planejamento visual. 2ª edição. São Paulo: Callis, 2005.

YOURDON, E. **Análise estruturada moderna**. 3.ed. Rio de Janeiro: Elsevier, 1990. 856p.