

**MODELO DE DOCUMENTAÇÃO**

**ENGENHARIA DE REQUISITOS**

**Brasília - DF**

**2019**

**Autor: Prof. Dr. Milton Pombo da Paz**

**Pró-Reitoria Acadêmica**

**Escola de Educação, Tecnologia e Comunicação**

**Curso de Bacharelado em Engenharia de Software**

**Trabalho de Disciplina**

**MODELO DE DOCUMENTAÇÃO DE PROJETO DE ENGENHARIA DE REQUISITOS**

Este **Modelo de Documentação de Projeto de Engenharia de Requisitos** foi desenvolvido pelo Prof. Dr. Milton Pombo da Paz na Universidade Católica de Brasília para a disciplina de Engenharia Requisitos dos Cursos de Bacharelado em Engenharia de Software, Bacharelado em Sistema de Informação, Bacharelado em Ciência da Computação, Gestão da Tecnologia da Informação e Análise e Desenvolvimento de Sistemas.

*Instruções gerais de formatação:*

1. *Seguir o Manual de documentação da UCB no site da biblioteca*
2. *Referências bibliográficas – ver no manual do site da biblioteca.*
3. *Citações: direta e indireta – ver no manual do site da biblioteca.*
4. *Figuras, tabelas, quadros e gráficos: colocar legenda e fonte – ver a partir da página 78 do manual.*
5. *Ilustrações: ver no manual do site da biblioteca como fazer legenda de tabela, quadro, figuras e gráficos.*
6. *Parágrafo: usar uma tabulação.*
7. *Palavras em outro idioma: itálico.*
8. *Conceitos: referenciar – ver referências diretas e indiretas – ver a partir da página 78 do manual.*
9. *Itens e subitens: numerar e formatar como título1,2,3,4 – ver a partir da página 78 do manual.*
10. *Ver a diferença gráfica de tabelas e quadros.*
11. *Sigla: na primeira vez colocar a sigla e depois pode usar somente a sigla.*
12. *Primar pela correção ortográfica e gramatical.*
13. *Não usar palavras na primeira pessoa do singular, ou do plural.*
14. *Não usar marcadores para listas: usar lista numerada ou alfabética.*
15. *Formatar corretamente o Sumário de acordo com a ABNT e Manual de Trab. Acadêmico UCB*
16. *A formatação do texto e pós-texto do Documento seguirá o mesmo padrão do Artigo, pag. 104 do Manual da UCB – no caso do PP use espaço 1 entre as linhas.*

****

**SISTEMA DE CONTROLE DE ALUNOS**

**Pró-Reitoria Acadêmica**

**Escola de Educação, Tecnologia e Comunicação**

**Curso de Bacharelado em Engenharia de Software**

**Trabalho de Disciplina de Engenharia de Requisitos**

**Autor: João Victor Bueno Carneiro**

**Orientador: Prof. Dr. Milton Pombo da Paz**

**Brasília - DF**

**2019**

**João Victor Bueno Carneiro**

**SISTEMA DE CONTROLE DE ALUNOS**

Documento apresentado ao Curso de graduação de Bacharelado em Engenharia de Software, da Universidade Católica de Brasília, como requisito parcial para obtenção da aprovação na disciplina de Engenharia Requisitos.

Orientador: Prof. Dr. Milton Pombo da Paz

**Brasília**

**2019**

|  |
| --- |
| BUENO, João Victor Carneiro.  Título do Trabalho. João Victor Bueno Carneiro. 11 de 11 de 2019.  Paginação: número.  Universidade Católica de Brasília, 11 de 11 de 2019.  Orientação: Prof. Dr. Milton Pombo da Paz. |



Projeto Engenharia Requisitos de autoria de João Victor Bueno Carneiro, intitulado “sistema de controle de alunos”, apresentado como requisito parcial para obtenção da aprovação na disciplina de Engenharia Requisitos do Curso de Bacharelado em Engenharia de Software da Universidade Católica de Brasília, defendido e aprovado pela banca examinadora abaixo assinada:

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

Prof. Dr. Milton Pombo da Paz

Engenharia de Requisitos

**Brasília**

**2019**

**DEDICATÓRIA**

Dedico este trabalho a todos que me ajudaram e me apoiaram durante todo o processo de desenvolvimento, em especial aos professores e meus pais pelo apoio.

**AGRADECIMENTOS**

Agradeço primeiramente a Deus, por me mostrar o caminho correto na realização desse trabalho.

Aos meus pais por todo apoio possível, amigos, familiares e professor pelo Milton Pombo da Paz que teve um papel fundamental na conclusão deste trabalho.

“O sucesso é a soma de pequenos esforços repetidos dia após dia.’’ Robert Collier

RESUMO

Referência: BUENO, João Victor Bueno Carneiro. Sistema de Controle de Alunos, 2019. nr p. Bacharelado em Engenharia de Software, ou Bacharelado em Sistemas de Informação, ou Bacharelado em Ciência da Computação, ou Gestão da Tecnologia da Informação ou Análise e Desenvolvimento de Sistemas – UCB – Universidade Católica de Brasília, Taguatinga – DF, 2019.

Palavras-chave: Engenharia de Requisitos, Educação, Faculdade, Registro, Cursos.

*ABSTRACT*

*Esse trabalho se trata de um sistema que fornece a organização de dados de todos os milhares de alunos da instituição de ensino Universidade Católica de Brasília. O sistema de controle de alunos será utilizado para organizar a matrícula dos alunos, situação de cada um e o mais importante, acompanhar o desempenho de cada um. A realização desse sistema é devido a uma pesquisa feita recentemente na instituição de que seria necessário um sistema para controlar a situação de cada estudante e informar se esses alunos estão gostando de estudar na UCB para além de levantar o mérito de ensino da Universidade, também perceber que o ensino realizado está sendo realmente bom para os estudantes. Os Resultados serão o fácil registro de novos alunos, informações melhores a respeito do ensino, fácil acesso ao sistema e mais informações sobre a instituição, controle de alunos formandos e alunos com a matrícula trancada.*

*Keywords:* Engenharia de Requisitos, Educação, Faculdade, Registro, Cursos.

LISTA DE FIGURAS

[Figura 1. Cronograma. 52](#_Toc22120631)

[Figura 2 - Diagrama de Caso de Uso de Negócio. 55](#_Toc22120632)

[Figura 3 - Diagrama de classe do Negócio 56](#_Toc22120633)

[Figura 5 - Diagrama de Atividades. 57](#_Toc22120635)

[Figura 6 - Qualidade de *Software.* 58](#_Toc22120636)

[Figura 7 - Diagrama de Casos de Uso de *Software*. 67](#_Toc22120637)

[Figura 8 - Diagrama de classe de análise – efetuar login. 71](#_Toc22120638)

[Figura 9 - Diagrama de interação de objetos – efetuar login. 72](#_Toc22120639)

[Figura 14 – Diagrama de Classe. 73](#_Toc22120644)

[Figura 17 – Diagrama de Componentes. 73](#_Toc22120647)

[Figura 18 - Diagrama de Implantação - a. 75](#_Toc22120648)

[Figura 19 - Diagrama de Implantação - b. 75](#_Toc22120649)

[Figura 20 - Diagrama de Implantação -c. 76](#_Toc22120650)

[Figura 21 - Diagrama de Implantação - d. 76](#_Toc22120651)

[Figura 28 - Diagrama de Atividades - a. 77](#_Toc22120658)

[Figura 29 - Diagrama de Atividades - b. 78](#_Toc22120659)

[Figura 30 - Diagrama de Sequência. 78](#_Toc22120660)

[Figura 39 - Diagrama de Casos de Uso - a. 80](#_Toc22120669)

[Figura 40 - Diagrama de Casos de Uso - b. 81](#_Toc22120670)

[Figura 41 - Diagrama de Máquina de Estado ou Transição de Estado - a. 83](#_Toc22120671)

[Figura 42 - Diagrama de Máquina de Estado ou Transição de Estado - b. 83](#_Toc22120672)

[Figura 43 - Diagrama de Máquina de Estado ou Transição de Estado - c. 84](#_Toc22120673)

[Figura 44 - MER: Modelo de Entidade-Relacionamento. 85](#_Toc22120674)

[Figura 45 - MFD: Modelo Físico de Dados. 86](#_Toc22120675)

LISTA DE TABELAS

[Tabela 1. Recursos Humanos. 48](#_Toc22120676)

[Tabela 2. Recursos de Hardware*.* 49](#_Toc22120677)

[Tabela 3. Recursos Financeiros. 51](#_Toc22120678)

LISTA DE QUADROS

[Quadro 1. Marcos do Projeto. 42](#_Toc22120679)

[Quadro 2. Riscos identificados que podem afetar o projeto. 43](#_Toc22120680)

[Quadro 3. Documentação a ser entregue no projeto. 44](#_Toc22120681)

[Quadro 4 - Requisitos de *Software.* 63](#_Toc22120682)

[Quadro 5 - Requisitos Não-Funcionais. 64](#_Toc22120683)

[Quadro 6 – Restrições. 65](#_Toc22120684)

**SUMÁRIO**

RESUMO 28

*ABSTRACT* 29

LISTA DE FIGURAS 30

LISTA DE TABELAS 31

LISTA DE QUADROS 32

1 INTRODUÇÃO 36

1.1 MOTIVAÇÃO 36

1.2 SURGIMENTO DA NECESSIDADE DO TRABALHO 36

1.3 USUÁRIOS DO PROJETO 36

1.4 ORGANIZAÇÃO INTERESSADA 36

1.5 ORGANOGRAMA DA ORGANIZAÇÃO 37

1.6 REQUISITOS DE ALTO NÍVEL 37

1.7 DIAGNÓSTICO DA EMPRESA 37

1.7.1 Histórico 37

1.7.2 Negócio 37

2 OBJETIVOS 38

2.1 OBJETIVO GERAL 38

2.2 OBJETIVOS ESPECÍFICOS 38

3 PROPOSTA DO SISTEMA 39

3.1 DESCRIÇÃO DO SISTEMA PROPOSTO 39

3.2 JUSTIFICATIVA DA PROPOSTA 39

3.3 RESULTADOS ESPERADOS 39

3.4 RESTRIÇÕES DO SISTEMA PROPOSTO 40

3.5 RELAÇÃO CUSTO X BENEFÍCIO: ANÁLISE DA VIABILIDADE ECONÔMICA DO NOVO SISTEMA 40

3.6 ÁREAS AFETADAS PELO NOVO SISTEMA: ÓRGÃOS OU FUNÇÕES ALTERADAS PELO SISTEMA. ESPECIFICAR AS ALTERAÇÕES PREVISTAS 40

4 PLANEJAMENTO DO PROJETO 41

4.1 PLANO DO PROCESSO DE DESENVOLVIMENTO 41

4.1.1 Ciclo de Vida do Projeto 41

4.1.2 Métodos de Desenvolvimento e Ferramentas CASE 42

4.2 PLANO DE ACOMPANHAMENTO 42

4.2.1 Marcos e Pontos de Controle 42

4.2.2 Métodos de Acompanhamento e Controle 43

4.2.3 Análise e Gerência de Riscos 43

4.3 PLANO DE DOCUMENTAÇÃO 44

4.3.1 Documentos do Projeto 44

4.4 PLANO DE RECURSOS E PRODUTOS 46

4.4.1 Plano de Pessoal 47

4.4.2 Recursos de Hardware para o Desenvolvimento 48

4.4.3 Recursos de Software 49

4.4.4 Configuração da rede 50

4.4.5 Configuração de Banco de Dados 50

4.4.6 Descrição das instalações 50

4.4.7 Recursos Financeiros 51

4.5 CRONOGRAMA DO PROJETO 51

5 ANÁLISE DE NEGÓCIO 53

5.1 MAPA MENTAL DO NEGÓCIO 53

5.2 REGRAS DE NEGÓCIO 53

5.3 DIAGRAMA DE CASOS DE USO DE NEGÓCIO 54

5.4 DIAGRAMAS DE CLASSE DO NEGÓCIO (MODELO DE DOMÍNIO) 55

5.6 DIAGRAMA DE ATIVIDADES 56

6 ANÁLISE DE SISTEMAS 58

6.1 DESCRIÇÃO DO PROBLEMA 58

6.2 IDENTIFICAÇÃO DOS INTERESSADOS 59

6.3 DESCRIÇÃO DAS NECESSIDADES DOS INTERESSADOS 60

6.4 DESCRIÇÃO DAS CARACTERÍSTICAS DO SISTEMA 61

7 ANÁLISE DE REQUISITOS 62

7.1 ESPECIFICAÇÃO DOS REQUISITOS DO SISTEMA 62

7.2 REQUISITOS DO SOFTWARE 62

7.2.1 Requisitos Funcionais 62

7.2.2 Requisitos Não-Funcionais (Suplementares) 64

7.3 RESTRIÇÕES 65

7.4 DIAGRAMA DE CASOS DE USO DA SOLUÇÃO 66

7.4.1 Visão Geral dos Casos de Uso e Atores 66

7.4.2 Descrição dos Cenários de Casos de Uso 67

8 DOCUMENTAÇÃO DE PROJETO 72

8.1 MEMORIAL DESCRITIVO DAS DECISÕES DE PROJETO 74

8.4 DIAGRAMA DE CLASSE 75

8.6 DIAGRAMA DE COMPONENTES 73

8.7 DIAGRAMA DE IMPLANTAÇÃO 74

8.10 DIAGRAMA DE ATIVIDADES 78

8.11 DIAGRAMA DE SEQUÊNCIA 79

8.15 DIAGRAMA DE CASOS DE USO 80

8.16 DIAGRAMA DE MÁQUINA DE ESTADO OU TRANSIÇÃO DE ESTADO

9 MODELAGEM DO BANCO DE DADOS 83

9.1 MODELO ENTIDADE RELACIONAMENTO 83

9.2 MODELO FÍSICO DE DADOS 84

10 DICIONÁRIO DE DADOS 85

11 TELAS DO PROTÓTIPO 86

12 CONCLUSÃO 87

12.1 TRABALHOS FUTUROS 87

REFERÊNCIAS 88

GLOSSÁRIO 89

# INTRODUÇÃO

*TI, informática nas organizações de maneira genérica. Efetuar login, sistema de controle de alunos.*

*Sistema de Controle de Alunos – Cadastro de alunos, controle boletos pagos e não pagos, controle de alunos bolsistas, controle de alunos com a matrícula trancada, mantendo o controle da situação de todos os alunos da instituição.*

*Organizar uma grande quantidade de cadastro de alunos e a situação em que cada um propõe.*

*Organizar o controle de cada aluno na instituição com objetivo de incentivar os demais a concluírem seus respectivos cursos até seu final. Informar a situação de cada aluno, controle de alunos com a matricula trancada, pagamentos pendentes, pagamentos em dia, bolsistas.*

## MOTIVAÇÃO

*A importância de implementar o conhecimento da engenharia de requisitos para poder oferecer um sistema que efetue um controle de cadastros de alunos em uma instituição.*

## SURGIMENTO DA NECESSIDADE DO TRABALHO

*Segurança no controle de alunos, situação cadastral de cada aluno, levantamento de boleto pagos e não pagos, alunos com a matrícula trancada, alunos aprovados e reprovados.*

## USUÁRIOS DO PROJETO

*Todos os alunos matriculados na instituição.*

## ORGANIZAÇÃO INTERESSADA

*Instituições De Ensino Superior, pós-Graduação, Mestrado, Doutorado.*

*Universidade Católica de Brasília.*

## ORGANOGRAMA DA ORGANIZAÇÃO

*Não há organograma para o tema aplicado.*

## REQUISITOS DE ALTO NÍVEL

*Sustentabilidade.*

## DIAGNÓSTICO DA EMPRESA

*Controle de Cadastro de alunos, para manter a sustentabilidade dos dados de todos os alunos da Universidade Católica de Brasília.*

### Histórico

*A Universidade de Brasília necessita de um sistema que sustenta os cadastros dos mais de 1000 alunos matriculados na instituição, onde esse sistema irá controlar cada dado do aluno onde irá verificar seu histórico durante o tempo em que estiver matriculado na Universidade.*

### Negócio

*Educação.*

# OBJETIVOS

*Objetivos Gerais e Específicos do Projeto.*

## OBJETIVO GERAL

*Desenvolver um sistema de informação para Controle de Alunos.*

## OBJETIVOS ESPECÍFICOS

*Organizar o gerenciamento de controle de alunos na instituição.*

# PROPOSTA DO SISTEMA

A seguir será apresentada a proposta do sistema, visando detalhar os principais pontos a serem seguidos.

## DESCRIÇÃO DO SISTEMA PROPOSTO

O sistema de controle de alunos, irá manter o controle de cadastro dos milhares de alunos que estudam na instituição. Esse sistema constitui em organizar alunos com direito a descontos, alunos pendentes em pagamentos, bolsistas.

Com esse projeto de sistema, é possível manter o controle geral da situação de cada aluno na instituição, além de organizar as situações ele permite informar o caminho de projeto de cada aluno em que fizer a escolha de seu curso, permitindo ver a possibilidade em atuar na área de trabalho, pois um dos objetivos da instituição é manter o foco do estudante até o final de seu curso escolhido.

## JUSTIFICATIVA DA PROPOSTA

Hoje é visto que, em muitos condomínios, os serviços realizados são feitos de forma manual e sem recursos informatizados, e, quando possuem algum tipo de automatização, não atendem às necessidades dos usuários. Por isso, é preciso o desenvolvimento de um sistema que permita informatizar as transações realizadas em um condomínio e que possa atender aos requisitos dos moradores e síndico de forma transparente e interativa entre os usuários.

O Sistema X é uma ferramenta que ajuda a resolver um problema muito comum nos sistemas de comunicação via *Internet* para condomínios, ou seja, sites dinâmicos com gerenciamento de conteúdo e funcionalidades específicas.

É visto que, muitas instituições têm problemas para informar os alunos como funciona a instituição, ou seja o problema é a informação onde irá atrair o aluno a querer estudar nessa instituição.

Esse sistema irá ajudar a instituição Universidade Católica de Brasília a mostrar um projeto de vida de cada curso em que o aluno possivelmente vá escolher, sendo assim, aproveitando o bom desempenho no MEC da instituição, o objetivo é usar esse desempenho para mostrar o quão eficiente é a Universidade Católica de Brasília.

## RESULTADOS ESPERADOS

Com a implantação do Sistema de controle de alunos, esperam-se os seguintes resultados:

1. Melhorar as situações dos alunos em caso de dúvidas sobre os cursos em que estão realizando;
2. Aumentar a credibilidade da instituição, afim de expandir o mercado de trabalho com alunos que cursaram nessa instituição e aumentar o índice de educação no MEC;
3. Informar a quantidade de alunos novos na instituição;
4. Informar um projeto de vida de cada curso em que o aluno irá escolher;
5. Controle de alunos bolsistas, desistentes, pagamentos em dia, pagamentos pendentes;
6. Informação dos eventos propostos pela instituição;
7. Controle alunos formandos, para a organização de cerimônia de formatura; e
8. Satisfação dos alunos a respeito das informações passadas pelo sistema da instituição.

## RESTRIÇÕES DO SISTEMA PROPOSTO

O sistema não irá informar a possibilidade de os alunos garantirem emprego, após o término do curso. O sistema não oferece descontos mensais para os alunos que pretendem entrar na instituição. A instituição não permite que alunos tenha acesso a dados da instituição após o término do curso.

Mesmo o aluno tendo feito parte da instituição, o aluno não poderá ter acesso aos dados utilizados nos trabalhos do curso inseridos no moodle da instituição.

## RELAÇÃO CUSTO X BENEFÍCIO: ANÁLISE DA VIABILIDADE ECONÔMICA DO NOVO SISTEMA

O processo de desenvolvimento desse software é muito trabalhoso e caro, pois o sistema consiste em diversas atividades de monitoramento e registro de milhares de alunos da instituição. Mesmo com todo o trabalho realizado, o sistema irá beneficiar tanto o desempenho da instituição quanto a satisfação dos alunos em saber que tudo em que o sistema informou, irá estar acontecendo no curso, sendo assim esse desempenho irá abranger os custos financeiros investidos no sistema com agilidade e qualidade no produto final do processo de desenvolvimento.

## ÁREAS AFETADAS PELO NOVO SISTEMA: ÓRGÃOS OU FUNÇÕES ALTERADAS PELO SISTEMA. ESPECIFICAR AS ALTERAÇÕES PREVISTAS

A ferramenta atuará diretamente nas funções desempenhadas pela equipe de desenvolvimento de *software* e/ou desenvolvedores autônomos.

As áreas afetadas pelo novo sistema são: Financeira e Administrativas.

# PLANEJAMENTO DO PROJETO

Ao longo desse capitulo, abordaremos os detalhes do plano do processo de desenvolvimento, de organização, de acompanhamento, de documentação, de recursos e produtos para execução e o cronograma do projeto.

## PLANO DO PROCESSO DE DESENVOLVIMENTO

Nesta seção serão descritos os processos, métodos, linguagem de programação e *hardware* que serão utilizados para desenvolvimento do *software*.

### Ciclo de Vida do Projeto

O clico de vida adotada no projeto será o modelo em cascata[[1]](#footnote-1), segundo Pressman (2006) *“o modelo em cascata é o mais tradicional entre os diversos tipos utilizados no mercado, chamado de ciclo de vida clássico.”* Esse modelo sugere um desenvolvimento de *software* de maneira sequencial que progride ao longo do projeto deixando as etapas dependentes uma da outra.

Foi decidido utilizar o modelo em cascata porque tem-se os requisitos do sistema bem compreendidos e estáveis.

As atividades do projeto estarão dentro do arcabouço de processo genérico, esse arcabouço é uma abordagem sistemática e utilizada como base para maioria dos processos de *software*. O arcabouço está dividido em etapas de comunicação, planejamento, modelagem, construção e implantação.

1. **Análise**: Análise de Negócio, Análise de sistema, Análise de Requisitos;
2. **Implementação**: Implementação BD, Implementação, Aplicação, Implementação REDE;
3. **Teste**: Teste BD, Teste Aplicação, Teste Rede, Teste Segurança;
4. **Projeto**: Projeto BD, Projeto Aplicação, Projeto REDE, Projeto Segurança;
5. **Implantação**: Implantação BD, Implantação, Aplicação, Implantação Rede, Segurança;
6. **Documentação;**
7. **Homologação; e**
8. **Encerramento.**

Por ser um modelo que nos permite adaptação à necessidade dos projetos, terá uma atividade de controle, que tem a função de monitorar e acompanhar o andamento do cronograma, além de estabelecer metas a serem cumpridas pelos envolvidos.

### Métodos de Desenvolvimento e Ferramentas CASE

O método de desenvolvimento do *software* terá como referência a programação orientado a objeto, utilizando a Linguagem de Modelagem Unificada (*Unified Modeling Language* - UML) para guiar a especificação, construção, visualização e documentação do sistema.

## PLANO DE ACOMPANHAMENTO

O plano de acompanhamento do projeto é a tarefa que tem a missão de garantir que o seu desenvolvimento ocorra dentro do cronograma estabelecido entre os interessados.

### Marcos e Pontos de Controle

Os marcos e pontos de controle são os prazos definidos entre os interessados para a entrega dos artefatos definidos. Os pontos de controle normalmente acontecem através de reuniões e também podem ser usadas para discutir as próximas etapas do projeto, alinhando novos prazos (Quadro 1).

Quadro 1. Marcos do Projeto.

| **Sequência** | **Ponto de Controle** | **Artefatos a serem entregues** | **Previsão de entrega** |
| --- | --- | --- | --- |
| 1 | Planejamento | Definir Proposta do Projeto | 15/03/2020 |
| 2 | Análise | Proposta refinada Plano de projeto elaborado Levantamento de custos e riscos Definição de um cronograma | 30/04/2020 |
| 3 | Projeto | Especificação dos requisitos do sistema Diagramas de casos de uso de negócio Diagramas de casos de uso de *software* Diagramas de atividades Especificação de caso de uso Descrição sistêmica do *software* | 31/05/2020 |
| 4 | Implementação | Codificação Testes | 31/10/2020 |
| 5 | Implantação | Produto Final - *Software* e Documentação | 10/11/2020 |

Fonte: Elaborado pelo autor.

### Métodos de Acompanhamento e Controle

O acompanhamento do projeto será através de reuniões semanalmente aos sábados pela nas dependências da instituição no turno da manhã, eventualmente os encontros podem acontecer durante a semana, preferencialmente com a presença do professor e de todos os membros da equipe. Troca de mensagens eletrônicas como *e-mails*, *sms* ou ligações durante a semana para satisfazer dúvidas, questionamentos ou validações de etapas do projeto.

O método de acompanhamento irá seguir com auxílio de alguns profissionais em outras áreas, como professores dos cursos específicos da instituição, para a construção do projeto de vida do aluno no sistema.

### Análise e Gerência de Riscos

Segundo o RUP o risco está associado a relação de altos e baixos com possibilidade de afetar o sucesso dos marcos principais e do projeto como um todo (Quadro 2).

Quadro 2. Riscos identificados que podem afetar o projeto.

| **Risco** | **Impacto** | **Probabilidade** | **Gravidade** | **Ação** |
| --- | --- | --- | --- | --- |
| Não cumprimento do prazo | Alto | Média | Alto | Prazos bem definidos. Reunião semanal de ponto de controle para acompanhar e definir metas. |
| Falta de comunicação entre os membros da equipe | Alto | Baixa | Alto | Reuniões objetivas, claras e com assuntos definidos e informados previamente. |
| Inexperiência na solução da proposta | Média | Média | Alto | Definir reuniões extras com os membros da equipe e com pessoas que dominam o assunto para troca de experiência. |
| Crescimento nas funcionalidades e características do *software* | Média | Baixa | Alto | Estabelecer o escopo e o não escopo do projeto. Deixar os requisitos bem definidos. |
| Perda do material já produzido | Alto | Baixa | Alto | Manter backup dos arquivos já produzidos em locais distintos. |
| Desfalque na equipe | Alto | Baixa | Alto | Deixar todos os integrantes cientes de suas responsabilidades e importância para o sucesso do projeto. |
| Falta de controle do gerente sobre a equipe. | Alto | Baixa | Alto | Indicar um membro da equipe para ser o ponto focal do professor orientador. |

Fonte: Elaborado pelo autor.

## PLANO DE DOCUMENTAÇÃO

Neste item serão apresentados os documentos que compõem o projeto de desenvolvimento do *Sistema de controle de Alunos* e suas respectivas fases de desenvolvimento.

### Documentos do Projeto

O Quadro 3 a seguir identifica cada artefato a ser entregue e suas respectivas fases de desenvolvimento segundo o RUP, descreve ainda o objetivo a ser atingido com a elaboração do documento.

Quadro 3. Documentação a ser entregue no projeto.

| **Fase** | **Artefato** | **Objetivo do Artefato** |
| --- | --- | --- |
| Planejamento | Proposta refinada | Detalhar e justificar com embasamento teórico as necessidades do *software*. |
| Plano de projeto elaborado | Definir objetivos, recursos e diretrizes a serem seguidos e necessários ao longo do projeto. |
| Levantamento de custos e riscos | Identificar os custos no projeto, os riscos e seus impactos no sucesso do sistema. |
| Definição de um cronograma | Estabelecer cronograma conforme os prazos para entrega e o tempo e esforço dos envolvidos. |
| Análise de negócio | ATA de Reunião, Regras de negócios, diagrama UC de negócio, diagrama UC classes de negócios. |  |
| Análise de Sistemas | Proposta de sistematização sistêmica. |  |
| Análise de Requisitos | Requisitos funcionais, Requisitos não-funcionais, diagrama UC solução, diagrama de classe da solução, documentação dos cenários dos casos de uso. |  |
| Projeto BD | Modelo entidade-relacionamento (MER), modelo logico de dados, modelo físico de dados, schema do banco de dados (script). |  |
| Projeto Aplicação | Arquitetura de funcionamento, diagrama de navegação de telas via MENU, protótipo das telas, projeto NVC. |  |
| Projeto REDE | Arquitetura da rede, topologia da rede, softwares componentes da rede. |  |
| Projeto Segurança | Controle de acesso, matriz de responsabilidade, criptografia, matriz de acessibilidade. |  |
| Implementação BD | Schema do banco de dados (script). |  |
| Implementação Aplicação | Diagrama estruturar modular dos programas, Codificação da aplicação. |  |
| Implementação REDE | Projeto físico da rede. |  |
| Implementação Segurança | Todo projeto de segurança da aplicação, plano de segurança da aplicação. |  |
| Teste BD | Relatório de avaliação do BD. |  |
| Teste Aplicação | Relatório de avaliação da aplicação. |  |
| Teste REDE | Relatório de avaliação da rede. |  |
| Teste Segurança | Relatório de avaliação da segurança. |  |
|  |  |  |
| Implantação BD | Relatório de avaliação da implantação. |  |
| Implantação Aplicação | Relatório de avaliação da implantação. |  |
| Implantação REDE | Relatório de avaliação da implantação. |  |
| Implantação Segurança | Relatório de avaliação da implantação. |  |
| Homologação | Documento de homologação dos resultados do projeto |  |
| Encerramento | Aceite dos resultados do projeto. |  |

Fonte: Elaborado pelo autor.

## PLANO DE RECURSOS E PRODUTOS

Ao longo desse tópico será abordado os recursos necessários para a obter sucesso no projeto de desenvolvimento do sistema de controle de alunos.

### Plano de Pessoal

A organização da equipe para produção desse projeto será dividida em duas: gerência e desenvolvimento.

#### Equipe de Gerência

A equipe de gerencia responsável por controlar o cronograma e recomendar melhorias e indicar correções durante o desenvolvimento do *software* será composta por:

1. Um Gerente de Projeto; e
2. Um Analista de Negócio.

#### Equipe de Desenvolvimento

A equipe de desenvolvimento responsável pela produção de todos os artefatos da documentação e da codificação e de testes do *software* será composta por:

1. Três Programadores Java;
2. Um Gerente de Projeto;
3. Um Analista de Sistemas;
4. Um Analista de requisitos;
5. Um DBA;
6. Um Analista de Testes;
7. Um Web Designer; e
8. Um Projetista.

OU

Os recursos humanos é o essencial em qualquer projeto, e na Tabela 1 será discriminado todos os recursos, quantidade, papeis, quantidade de horas, valor hora, valor unitário e total de cada recurso e o valor total gasto com os recursos humanos necessários para o desenvolvimento do projeto.

Tabela 1. Recursos Humanos.

| **ITEM** | **RECURSO** | **PAPEL** | **QTD** | **QTD DE HORAS** | **VALOR HORA** | **VALOR UNITÁRIO** | **TOTAL** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | Gerente de Projetos | Gerenciar e controlar o andamento do projeto. | 1 | 240 | R$ 50,00 | R$ 12.000,00 | R$ 12.000,00 |
| 2 | Analista de Requisitos | Levantar e elicitar os requisitos do *software* a ser desenvolvido | 3 | 40 | R$ 30,00 | R$ 1.200,00 | R$ 3.600,00 |
| 3 | Analista de Negócios | Definir soluções que permitam que o *Software* atinja seus objetivos. | 1 | 40 | R$ 50,00 | R$ 2.000,00 | R$ 2.000,00 |
| 4 | DBA | Assegurar que o banco de dados estejam no ar todo o tempo que o sistema necessita, com rapidez e confiabilidade. | 3 | 40 | R$ 40,00 | R$ 3.200,00 | R$ 9.600,00 |
| 5 | Programador | Codificar os requisitos levantados pelo analista de requisitos. | 3 | 120 | R$ 30,00 | R$ 3.600,00 | R$ 10.800,00 |
| 6 | Analista de Teste | Testar os códigos feitos pelo programador e comprovar que estão atendendo os requisitos validados pelo cliente | 3 | 40 | R$ 30,00 | R$ 1.200,00 | R$ 3.600,00 |
|  | **TOTAL** | | | | | | R$ 41.600,00 |

Fonte: Elaborado pelo autor.

### Recursos de Hardware para o Desenvolvimento

A Tabela 2 a seguir representa a configuração de *hardware* dos computadores pessoais da equipe de desenvolvimento, podendo essas configurações sofrer modificações durante o projeto.

Os recursos de *Hardware* utilizados para o desenvolvimento do projeto são discriminados na Tabela 2 com a descrição, quantidade, valor unitário e total.

Tabela 2. Recursos de Hardware.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Item** | **Descrição** | **Unidade** | **Quantidade** | **PU** | **PT** |
| 1 | Notebook Dell - Intel Core I5-3210M CPU @ 2.50GHz - 6 GB RAM - 1 TB HD - Placa de Vídeo Nvidia GeForce GT 630M 128-bit 1GB | Un | 1 | 1000,00 | 1000,00 |
| 2 | Notebook Dell - Intel Core I5-4200M CPU @ 3.10GHz - 6 GB RAM - 1 TB HD - Placa de Vídeo Nvidia GeForce GT 6200 128-bit 1GB | Un | 1 | 1000,00 | 1000,00 |
| 3 | Notebook Samsung- Intel Core I3 CPU @ 2.50GHz –   1. GB RAM - 720 HD | Un | 1 | 1000,00 | 1000,00 |
|  |  |  |  | Total | 3000,00 |

Fonte: Elaborado pelo autor.

### Recursos de Software

#### Servidor: Sistema Operacional

* *Microsoft® Windows XP Professional SP2;*
* *Java/ JSP*;
* Servidor *Web*: *Apache TomCat*; e
* Sistema Gerenciador de Banco de Dados: *Microsoft SQL 2000*.

**Cliente:**

* Sistema operacional: *Windows XP*, *Vista* ou *Seven*, *Unix* e *Linux*;
* *Plugins*: Máquina Virtual (*JRE*), *Adobe Acrobat Reader, Macromedia Flash, Real Player e QuickTime*; e
* *Browser* compatível com a tecnologia *Java*.

#### Linguagens de Programação

O desenvolvimento da aplicação *CodeGen* para a plataforma *Desktop* será baseada na linguagem de programação orientada objetos *Java*, utilizando muito dos conceitos JPA[[2]](#footnote-2)A (Java *Persistence* API), *Annotation*[[3]](#footnote-3)e *Reflection*[[4]](#footnote-4)para identificar estruturas de Banco de Dados relacional.

### Configuração da rede

* Conexão com a *internet* de no mínimo 1*Mbp*s;
* Todas as estações cliente deverão estar conectadas na rede *(Internet*) e ter acesso ao servidor de arquivos e a base de dados;
* *Modem ADSL* Externo ou Placa de *Modem Interno PCI*;
* Placas de rede *Ethernet* 10/100 *Mbps* – RJ45 ou *USB* (conforme modem) em quantidade igual ou superior à quantidade de computadores;
* *ADSL* (Recomendado) ou Conexão discada;
* RJ – 45 ou RJ – 25; e
* Cabo *UTP* Categoria – 5.

### Configuração de Banco de Dados

Sistema Gerenciador de Banco de Dados Relacional *MYSQL* 5.0, pois ele é *open source,* possui consistência, alta [performance](http://www.oficinadanet.com.br/artigo/484/por_que_usar_mysql), confiabilidade e é fácil de usar.

### Descrição das instalações

O portal será hospedado em um servidor *Web Hosting*, que contenha a seguinte infraestrutura.

1. Conexão ininterrupta à *internet*;
2. Links acima de 20 *Giga Bits* por minuto;
3. Pontos de interconexão (PIX) do Ponto de Troca de Tráfego (PTT) homologado pelo CGI.Br (Comitê Gestor da *Internet* no Brasil);
4. Fornecimento ininterrupto de energia;
5. No *Breaks* e geradores de energia;
6. Sistemas de *IPS* (*Intrusion Prevention System*), para identificação e bloqueio de tentativas de intrusão aos servidores;
7. *Firewalls* redundantes *Backup* ativo;
8. Sistema de refrigeração/climatização;
9. Sistema contra incêndio; e
10. Instalações com espaço e organização para equipamentos e equipe técnica.

As máquinas dos usuários (clientes) acessarão o serviço a partir do navegador *Web* padrão utilizado, não sendo necessária a instalação de aplicações clientes nas estações, podendo assim, consequentemente, ser acessado de qualquer local que possua *internet*.

### Recursos Financeiros

O investimento inicial para o desenvolvimento do projeto é a soma dos valores necessário para alocação de recursos humanos, *software* e *hardware* (Tabela 3).

Tabela 3. Recursos Financeiros.

|  |  |  |
| --- | --- | --- |
| **Item** | **Descrição** | **Total** |
| 1 | Recursos Humanos | R$ 41.600,00 |
| 2 | Recursos de *Hardware* | R$ 6.400,00 |
| 3 | Recursos de *Software* | R$ 370,50 |
| **Total** | | R$ 48.370,50 |

Fonte: Elaborado pelo autor.

## CRONOGRAMA DO PROJETO

O cronograma é parte essencial de qualquer projeto, e faz parte tanto do sucesso quanto do fracasso de um resultado. O cronograma deve ser controlado e monitorado através das reuniões de ponto de controle e só ser alterado se todos os envolvidos aprovarem.

A Figura 1 apresenta o cronograma elaborado para o desenvolvimento desse projeto.

**Uma imagem contendo texto

Descrição gerada automaticamente**

Figura 1. Cronograma.

Fonte: Elaborado pelo autor.

# ANÁLISE DE NEGÓCIO

Neste capítulo será descrito, através de diagramas e especificações, o processo do negócio em que o *software* em questão será inserido, sendo estes o diagrama do modelo de caso de uso de negócio, diagrama do modelo de classes do negócio, e, por fim, o diagrama de atividades.

## MAPA MENTAL DO NEGÓCIO

Uma imagem contendo captura de tela

Descrição gerada automaticamente

## REGRAS DE NEGÓCIO

*São as regras que fazem o negócio existir.*

| **Número** | **Nome** | **Descrição** | **Setor** |
| --- | --- | --- | --- |
| RN1 | Registro de alunos | Registro de novos alunos, Matrícula, alunos que estão terminando curso (formando). |  |
| RN2 | Informações gerais | Informações sobre as matérias do curso, informações do recebimento do boleto e notas fiscais. |  |
| RN3 | Realização de eventos | Encontros Presenciais (EaD), formatura, pesquisas de campo. |  |
| RN4 | EaD | Opções de matérias, informações das matérias e agendamento de encontros presenciais. |  |
| RN5 | Situação dos Alunos | Alunos com o pagamento em dia, alunos com pagamentos em atraso, alunos com direito a descontos especiais. |  |
| RN6 | Alunos Bolsistas | Alunos com bolsa de 100%;  Alunos com bolsa de 75%;  Alunos com bolsa de 50%;  Alunos com bolsa de 25%; |  |
| RN7 | Transferência de curso | Transferência de curso. |  |
| RN8 | Transferência de Instituição | Confirmar curso e semestre, confirmar dados de transferência e a transferência. |  |
| RN9 | Portal do aluno | Notas, 2º via do boleto, faltas, matérias, fazer a grade de matérias. |  |

## DIAGRAMA DE CASOS DE USO DE NEGÓCIO

O diagrama de casos de uso de negócio demonstra as principais funções que são executadas por cada ator dentro do processo de desenvolvimento de *software*, sendo que o diagrama abaixo foi desenvolvido baseado no processo unificado (RUP). Neste diagrama são representadas as principais atividades desenvolvidas dentro de uma iteração do RUP, sendo que o *software* resultado deste trabalho será baseado, especificamente, na etapa de codificação.

A Figura 2 apresenta o Diagrama de Casos de Uso de Negócio com a visão de cada ator no sistema de controle de alunos.

Uma imagem contendo texto, mapa

Descrição gerada automaticamente

Figura 2 - Diagrama de Caso de Uso de Negócio.

Fonte: Elaborado pelo autor.

## DIAGRAMAS DE CLASSE DO NEGÓCIO (MODELO DE DOMÍNIO)

A Figura 3 apresenta o Diagrama de Classe de Negócio, com a visão de cada ator no sistema de controle de alunos.

Uma imagem contendo texto, mapa

Descrição gerada automaticamente

Figura 2 - Diagrama de classe do Negócio

Fonte: Elaborado pelo autor.

## DIAGRAMA DE ATIVIDADES

O diagrama a seguir, conforme Figura 5, mostra o fluxo das atividades realizadas no negócio do cliente.

Uma imagem contendo captura de tela

Descrição gerada automaticamente

Figura 3 - Diagrama de Atividades.

Fonte: Elaborado pelo autor.

# ANÁLISE DE SISTEMAS

Esse sistema irá solucionar problemas como, a organização de dados dos diversos alunos matriculados na UCB e tornar esse sistema mais fácil de ser acessado e de encontrar as necessidades que o usuário desejar.

Neste capítulo serão descritos os problemas que aplicação irá solucionar e as funcionalidades que o *software* deverá atender.

## DESCRIÇÃO DO PROBLEMA

O desenvolvimento de *software* vem crescendo de forma exponencial, juntamente com aumento da diversidade de aplicações do mesmo em diversas tarefas. Para suprir toda esta demanda, com eficiência e eficácia, surge a todo o momentos diversos métodos, técnicas e tecnologias para que se produza o *software* cada vez mais rápido e com maior qualidade. Para ratificar esta informação, pode-se citar como exemplos o modelo de processo *Rapid Application Development* (RAD), o uso do conceito iterativo incremental nos modelos de processo de desenvolvimento de *software*, além de ferramentas CASE para auxiliar a equipe de desenvolvimento nas diversas etapas do desenvolvimento. Como pode ser visto na Figura 6 qualidade do *software* depende de vários fatores que quando combinados de forma distribuída torna o produto final mais estável. A utilização de ferramentas na etapa de codificação (geradores) auxilia no equilíbrio da variável tempo, tornando o processo mais ágil.

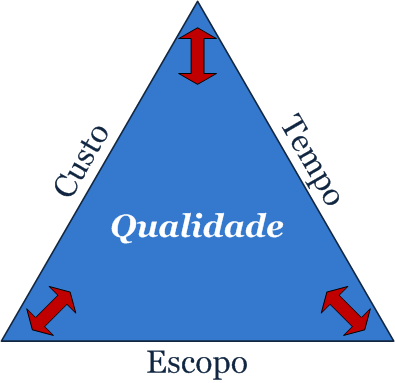


Figura 4 - Qualidade de *Software.*

Fonte:Luiz Ricardo, 2013.

A eficiência e eficácia do processo de desenvolvimento de *software* dependem de cada etapa do mesmo. Uma etapa que é de vital importância para a qualidade do processo é a etapa de codificação. Para esta etapa existem diversas ferramentas e técnicas para torná-la mais produtiva, dentre elas pode-se citar os geradores de aplicações baseadas no conceito RAD e geradores de código para uma plataforma específica. Estas ferramentas trazem à etapa de codificação de *software* agilidade e padronização que são elementos fundamentais para a qualidade do produto gerado. Porém os geradores, tanto de aplicação como de código, possuem limitações que prejudicam a escalabilidade do projeto, ou seja, dificulta a evolução da aplicação para as regras específicas do *software* que está sendo desenvolvido. Pode-se citar como exemplo destas limitações:

1. Código acoplado com arquitetura do gerador;
2. Código de difícil entendimento;
3. Opção de geração em uma arquitetura específica;
4. Dificuldade na manipulação da ferramenta (*design*); e
5. Falta de suporte à escalabilidade no código gerado.

Estas limitações citadas são comumente utilizadas como motivo para evitar o uso deste tipo de *software*. Para resolver este problema será desenvolvido um *software* de geração de código que auxiliará na implementação inicial do sistema, oferecendo ao usuário um produto final de qualidade, legível e escalável.

## IDENTIFICAÇÃO DOS INTERESSADOS

A área de tecnologia da informação em uma empresa é de vital importância para o sucesso da mesma. Quando os recursos oferecidos pela TI são empregados de forma coerente se tem como resultado a vantagem competitiva. Diante disto, todas melhorias que algum processo contido na área de Tecnologia da Informação irá refletir em todo o negócio. Sendo o processo de desenvolvimento de *software* pertencente à esta área, torná-lo mais ágil e padronizado irá trazer melhorias que irão ultrapassar os limites da área de Tecnologia da Informação. Pode-se citar como interessados diretamente:

1. Programador;
2. Gerente de Projetos;
3. Arquiteto de *software*;
4. Testador;
5. Analista de negócio; e
6. Usuário.

Indiretamente, surgem outros interessados, que mesmo não tendo envolvimento direto com processo de desenvolvimento de *software* irá se beneficiar de alguma forma destas melhorias:

1. Dono da empresa; e
2. Executivos.

## DESCRIÇÃO DAS NECESSIDADES DOS INTERESSADOS

O grande alcance dos interessados do *software* é ratificado pela quantidade de necessidades que o mesmo soluciona. Estas necessidades podem ser caracterizadas pela dificuldade de estruturar o início de um projeto, padronização do código, utilização de ferramentas CASE por pessoas com menos experiência e a demora em entregar um produto funcional para o cliente.

A fase inicial da etapa de codificação de *software* é caracterizada, na maioria das vezes, por atividades repetitivas que ocupam boa parte do tempo do programador. Pode-se citar como exemplo destas atividades a criação das classes de domínio, as operações básicas (CRUD) que são realizadas sobre as classes de domínio assim como as validações básicas que são necessárias para manter a integridades dos dados fornecidos pelo *software* em questão. Muitas vezes essas atividades executadas no início do desenvolvimento são auxiliadas por ferramentas CASE, porém a utilização destas ferramentas costuma ser complexa sendo que para algum membro da equipe manipular deverá ter muito conhecimento técnico impedindo de um estagiário ou profissional com menos conhecimento técnico manipulá-la.

Outra situação que cerca todo o processo de desenvolvimento de *software* é a demora em demonstrar algum resultado funcional da aplicação para o cliente, causando muitas vezes desconfiança por parte do contratante e discussões desnecessárias. Este fato dificulta a validação do sistema pelo cliente, considerando que o cliente consegue verificar de forma mais eficaz se o sistema atende as suas analisando uma aplicação já funcional.

## DESCRIÇÃO DAS CARACTERÍSTICAS DO SISTEMA

O sistema a ser desenvolvido deverá conter diversas características para que as necessidades de seus usuários sejam solucionadas. Dentre as principais características que o *software* deverá atender estão: a facilidade de manipulação da aplicação (interface gráfica), agilidade no processo de geração sendo que a aplicação irá disponibilizar uma geração padrão ou uma geração customizada, abstração do processo de codificação de *software*, ou seja, o usuário não precisará ter conhecimento avançados em codificação tendo apenas conhecimento da estrutura de dados e das principais regras de negócio.

Outra característica que cerca o produto final do gerador de código é que seu resultado seja uma aplicação funcional (CRUD) com as tabelas de domínio para que possa solucionar a necessidade de demonstração de um produto funcional para o cliente, trazendo confiabilidade e segurança para o mesmo.

Facilidade de acesso e pesquisa de suas necessidades e forte segurança de dados cadastrados.

# ANÁLISE DE REQUISITOS

Esse sistema irá solucionar problemas como, a organização de dados dos diversos alunos matriculados na UCB e tornar esse sistema mais fácil de ser acessado e de encontrar as necessidades que o usuário desejar.

Neste capítulo serão descritos os problemas que aplicação irá solucionar e as funcionalidades que o *software* deverá atender, assim como os requisitos suplementares e as restrições do mesmo.

## ESPECIFICAÇÃO DOS REQUISITOS DO SISTEMA

Serão abordados neste capítulo aspectos relacionados aos requisitos funcionais e não funcionais do sistema. Realizar-se-á uma descrição resumida do problema cuja aplicação deverá solucionar. Após a descrição do problema serão definidos os interessados pelo sistema com suas respectivas necessidades. Após a identificação de todos os interessados, se definirão os requisitos de *software,* ou seja, os requisitos funcionais do sistema, suas funcionalidades serão classificadas de acordo com as necessidades que cada interessado possui. Após a identificação dos requisitos funcionais, será desenvolvida a especificação suplementar, que define os requisitos não-funcionais do sistema. E, por último, se definirá as restrições de *software* que a aplicação possui tanto as relacionadas à infraestrutura quanto ao ambiente de *software*.

## REQUISITOS DO SOFTWARE

### Requisitos Funcionais

*São os requisitos da solução sistêmica.*

| **Número** | **Nome Requisito** | **Descrição** | **RN** |
| --- | --- | --- | --- |
| RF1 | EaD | Acessar no moodle de matérias de ensino a distância e ter acesso além das matérias, eventos, provas práticas dadas aos professores de EaD. |  |
| RF2 | Alunos Bolsistas | Alunos com direito a descontos especiais nas mensalidades para os que fizeram o vestibular ou o ENEM. |  |
| RF3 | Situação dos alunos | Acessar os dados do aluno e ver o seu desempenho, pagamentos atrasados e pagamentos em dia. |  |
| RF4 | transferência de curso | Consultar os dados do aluno e ver as matérias concluídas, para poder fazer a transferência (caso for um curso ligado ao curso em que o aluno estiver cursando). |  |
| RF5 | Transferência de instituição | Consultar dados de curso e matérias concluídas. Acessar o requisito de Registros e fazer a nova matrícula do aluno. |  |
| RF6 | Registro de Alunos | Registrar dados necessários para a matrícula do aluno, organizar a quantidade de alunos novos e alunos que concluíram seus cursos ou saíram da instituição. |  |
| RF7 | Informações Gerais | Aluno poderá acessar aos seus dados de matrícula e de seu curso, terá acesso as suas notas, número de faltas, mensalidades pagas em dia e em atraso. |  |
| RF8 | Realização de eventos | Consultar as datas de eventos como palestras, formaturas e pesquisas de campo, atividades presenciais que irá precisar do agendamento do aluno no moodle (EaD). |  |
| RF9 | Portal do Aluno | O aluno pode consultar seu histórico no curso e seu desempenho, número de faltas em cada matéria e notas. Pode também fazer sua grade de matérias como desejar de acordo com as matérias que o curso oferece e aprovações. |  |

Todas as funcionalidades do SIGESC deverão ser disponibilizadas de acordo com o perfil do usuário *logado* no momento.

Os requisitos de *software* descrevem as características que o sistema deve atender para solucionar as necessidades do usuário. Os requisitos de *software* da aplicação CodeGen estão listados no Quadro 4.

Quadro 4 - Requisitos de *Software.*

| **Prioridade** | **Requisito** | **Descrição** |
| --- | --- | --- |
| Alta | Configurar novo projeto | Esta funcionalidade é responsável por informar os principais dados do projeto. |
| Alta | Configurar banco de dados | Esta funcionalidade é responsável pela configuração das credenciais do banco de dados que será usado pelo gerador. |
| Alta | Configurar fonte de dados | Esta funcionalidade é responsável por configurar a fonte de dados que o sistema irá executar como base para a geração do código-fonte. |
| Alta | Realizar *upload* de arquivo | Esta funcionalidade é responsável por realizar o upload de um arquivo. |
| Alta | Download do código-fonte | Esta funcionalidade permite ao usuário realizar o download do código fonte gerado pelo sistema. |
| Alta | Gerar código-fonte | Esta funcionalidade é a geração do CRUD e das camadas MVC do projeto. |
| Alta | Visualizar estrutura de dados importada. | Esta funcionalidade permite ao usuário visualizar estrutura de dados do banco e seus relacionamentos. |
| Média | Download da imagem do MER | Esta funcionalidade permite ao usuário realizar Download da imagem da estrutura do BD importada. |
| Alta | Navegar entre as etapas do processo. | Esta funcionalidade é responsável pela navegação entre as diferentes etapas do processo. |
| Média | Carregar um projeto | Esta funcionalidade é responsável por carregar um projeto que já havia sido cadastrado anteriormente. |
| Alta | *Download* de uma versão anterior do projeto. | Esta funcionalidade é responsável por carregar um projeto que já havia sido cadastrado anteriormente. |
| Alta | Selecionar tipo de geração | Esta funcionalidade permite ao usuário escolher o tipo de geração que deseja. Se um projeto padrão ou customizado. |
| Alta | Editar configurações da interface | Esta funcionalidade permite ao usuário buscar o elemento que se deseja alterar e editar as suas configurações da interface das páginas. |
| Alta | Editar configurações do projeto | Esta funcionalidade permite ao usuário editar as configurações de arquitetura, Gerência de dependência e estrutura do projeto. |

Fonte: Elaborado pelo autor.

### Requisitos Não-Funcionais (Suplementares)

Os requisitos suplementares, ou não-funcionais, tem como objetivo manter a qualidade do *software* desejada, sendo classificados de diversas formas, tais como: performance, segurança, comunicabilidade, usabilidade entre outros. Os requisitos suplementares estão listados no **Erro! Autoreferência de indicador não válida.**.

Quadro 5 - Requisitos Não-Funcionais.

| **Número** | **Nome RNF** | **Descrição** | **RF** |
| --- | --- | --- | --- |
| RNF1 | Usabilidade | As telas do sistema devem seguir o padrão *wizard.* |  |
| RNF2 | Usabilidade | Cada tela da ferramenta deverá conter uma breve explicação da sua função. |  |
| RNF3 | Usabilidade | Cada campo do formulário deverá ter mensagens explicativas de sua função. |  |
| RNF4 | Usabilidade | A usabilidade do sistema será de total aplicabilidade, impactando em diversos aspectos, que vão desde a facilidade de uso, pois o sistema possuíra um menu “Ajuda” com as informações de uso e “dúvidas” comuns entre os usuários sobre a utilização de modo geral da navegabilidade do SIGESC até a poder participar de reuniões sem sair de sua comodidade de forma instantânea através do menu “chat”, entre outras características específicas deste sistema. |  |
| RNF5 | Interoperabilidade | O sistema deve executar nas plataformas *Windows*, *Linux* e *MacOS*. |  |
| RNF6 | Ambiente | O sistema deve executar em plataforma *desktop.* |  |
| RNF7 | Padronização | As estruturas de fontes de dados fornecidas devem estar em conformidade com os padrões do *CodeGen*. |  |
| RNF8 | Padronização | O código gerado pelo sistema deve seguir os padrões da linguagem de programação. |  |
| RNF9 | Disponibilidade | O sistema deverá estar disponível 24 horas por dia durante sete dias por semana |  |
| RNF10 | Desempenho | O tempo médio de consultas não deverá ultrapassar 5 segundos. Se esse tempo exceder é porque o banco de dados está no limite de seu tamanho, o que indica que será necessária uma evolução de hardware para rede ou processadores e memória para o servidor; e  O sistema deverá agilizar o processo de atendimento ao síndico, subsíndico e condôminos. |  |
| RNF11 | Segurança | Todo acesso à informação do sistema será controlado de acordo com o perfil do usuário mediante autenticação; e  O sistema possuirá a geração de arquivos de log. |  |
| RNF12 | Suportabilidade | O sistema possuirá codificação padronizada, para facilitar em futuras modificações. |  |

Fonte: Elaborado pelo autor.

## RESTRIÇÕES

As restrições do sistema caracterizam as funcionalidades que o *software*, em sua versão atual, não irá atender. As restrições do *software* estão listadas no Quadro 6.

Quadro 6 – Restrições.

| **Restrições** |
| --- |
| A geração do código será baseada na linguagem de programação *Java*. |
| O Sistema não irá abranger as regras de negócio. |
| O código gerado pelo sistema não realiza validações como CPF e Datas. |

Fonte: Elaborado pelo autor.

## DIAGRAMA DE CASOS DE USO DA SOLUÇÃO

Nesta seção serão definidos os modelos de casos de uso. Primeiramente será mostrada uma visão geral dos casos de uso que definem as funcionalidades do sistema, com seus respectivos atores. Posteriormente será feita a descrição de cada caso de uso que deverá ser implementado no sistema.

Uma imagem contendo texto, mapa

Descrição gerada automaticamente

Figura 7 - Diagrama de Casos de Uso de *Software*.

Fonte: Elaborado pelo autor.

### Visão Geral dos Casos de Uso e Atores

A seguir será apresentado o Diagrama de Casos de Uso de *Software* com a visão de cada ator no sistema de controle de alunos, abrangendo assim todas as funcionalidades previstas para a implementação.

Uma imagem contendo mapa, texto

Descrição gerada automaticamente

Figura 8 - Diagrama de Casos de Uso de *Software*.

Fonte: Elaborado pelo autor.

### Descrição dos Cenários de Casos de Uso

*Caso de uso efetuar login no portal do aluno.*

Uma imagem contendo texto

Descrição gerada automaticamente Nesta seção serão descritos todos os casos de uso apresentados no diagrama de caso uso de *software*. Esta descrição irá conter o nome do caso de uso, objetivo, atores, pré-condições, fluxo principal, fluxos alternativos, fluxos de exceção, pós-condições e características suplementares.

#### Descrição do caso de uso UC01 – Efetuar Login Portal do Aluno

A descrição detalhada do caso de uso é responsável por apresentar os fluxos principal, alternativos e de exceção do caso de uso em questão além de apresentar as pré-condições e pós-condições que existem antes e após a execução do mesmo, respectivamente.

Histórico de Revisão

|  |  |  |  |
| --- | --- | --- | --- |
| **Nome** | **Data** | **Razão da mudança** | **Versão** |
| Moisés  Moisés  Moisés | 15/11/2019 | Criação do documento  Revisão do documento  Ajustes no fluxo principal | 1.0  1.1  1.2 |

|  |  |  |  |
| --- | --- | --- | --- |
| ID do Caso de Uso: | UC-01 | | |
| Nome do Caso de Uso: | Efetuar *Login Portal do aluno* | | |
| Criado por: | João | Última atualização: |  |
| Data da Criação: | 15/11/2019 | Data da última atualização: |  |

|  |  |
| --- | --- |
| Ator: | Aluno. |
| Descrição | Permitir o autor ter acesso as matérias, notas, número de faltas e mudanças na grade horária em mudança de semestre. |
| Pré-condições | Usuário deverá estar cadastrado no banco de dados do sistema com a sua matrícula. |
| Pós-condições | Usuário *Logado* |
| Prioridade | Alta |
| Frequência de Uso | Alta |
| Fluxo Principal | P1. O caso de uso se inicia quando o usuário acessa o sistema por meio da *internet*.  P2. O sistema disponibiliza uma tela com dois campos para serem preenchidos com o usuário e a senha do ator. (Tela01) [E2].  P3. O ator preenche os campos disponibilizados na tela e seleciona o botão de entrar no sistema [A1] [A2].  P4. O sistema procura o usuário na base de dados e, caso exista, verifica se a senha informada é a mesma senha do usuário encontrado [E1].  P5. O sistema autentica o usuário. (Tela03)[E2]  P6. O caso de uso se encerra. |
| Fluxo Alternativo | A1. O ator seleciona a opção “Recuperar senha”  A1.1 O sistema redireciona o ator para uma nova tela de recuperação de senha (Tela02).  A1.2 O ator digita o seu *e-mail* e clica no botão “Obter nova senha por *e-mail*.  A1.3 O sistema verifica existência do *e-mail* na base de dados [E1].  A1.4 O sistema emite uma nova senha por *e-mail*, e retorna para a tela de *login* (Tela01).  A1.5 O sistema retorna para o passo P3.  A2 Sair  A2.1 O usuário fecha a tela do *browser*.  A2.3 Segue para o passo P6. |
| Exceções | E1. O sistema não encontra nenhum usuário com os dados informados.  E1.1 O sistema informa uma mensagem de erro(M001).  E1.2 Segue para o passo A1.1.  E2 O sistema não consegue redirecionar o ator para a tela desejada.  E2.1 O sistema informa uma mensagem de erro(M004).  E2.2 Segue para o passo P6.  E3. O sistema não encontra o *e-mail* no banco de dados.  E3.1 O sistema informa uma mensagem de erro(M002).  E3.2 Segue para o passo A1.2. |
| Requerimentos Especiais | Usuário deve ter privilégios de acesso a essas funções específicas |
| Suposições |  |
| Notas e casos | As mensagens estão especificadas na Tabela Mensagens do Sistema (anexo 01). |

Anexo 01 – Mensagens do Sistema

|  |  |
| --- | --- |
| **Código da mensagem** | **Mensagem** |
| M001 | “Usuário ou senha digitados está invalido! Tente novamente” |
| M002 | “*E-mail* informado invalido, verifique a digitação do seu e-*mail”* |
| M003 | “Erro ao redirecionar a página selecionada. Favor entrar em contato com o administrador do sistema” |
| M004 | “Erro inesperado. Favor entrar em contato com o administrador do sistema” |

A Figura 8 e Figura 9 representa um protótipo de tela que apresenta o diagrama de classe e diagrama de análise respectivamente, referente ao caso de uso UC01 – Efetuar *login*.



##### **Diagrama de classe de análise**



Figura 5 - Diagrama de classe de análise – efetuar login.

Fonte: Elaborado pelo autor.

##### **Diagrama de interação de objetos**

Uma imagem contendo captura de tela

Descrição gerada automaticamente

Figura 6 - Diagrama de interação de objetos – Transferência de curso

Fonte: Elaborado pelo autor.

# DOCUMENTAÇÃO DE PROJETO

O sucesso para a aplicação do processo com tecnologias orientadas a objetos está ligado diretamente à arquitetura em camadas e principalmente às observações do mercado atual.

Esta organização em camadas nos permitirá independências e tem como principais objetivos: Atingir a eficiência; Escalabilidade; Reutilização e Facilidade em Manutenção.

## MEMORIAL DESCRITIVO DAS DECISÕES DE PROJETO

Neste item, serão expostos os modelos de: Mapa de Navegação do Sistema, Diagrama de dados do Projeto, Topologia do Ambiente de Operação (diagrama de Implantação) e Diagrama de Componentes.

## DIAGRAMA DE CLASSE

Algumas opções do aluno que pode fazer no sistema. Alteração de dados (Email, telefone e etc), solicitar mudanças no curso, turno, campus, instituição, acesso a biblioteca pelo site, acesso ao portal do aluno.

Uma imagem contendo captura de tela

Descrição gerada automaticamente

Figura 7 – Diagrama de Classe.

Fonte: Elaborado pelo autor.

## DIAGRAMA DE COMPONENTES SOFTWARE

*Diagrama sobre o acesso do aluno ao site da UCB e as opções de acesso e mais sobre a instituição e cursos.*

*Uma imagem contendo captura de tela

Descrição gerada automaticamente*

Figura 8 – Diagrama de Componentes.

Fonte: Elaborado pelo autor.

## DIAGRAMA DE IMPLANTAÇÃO hardware

*Diagrama definindo a trajetória de acesso do sistema através da internet com auxílio do navegador e site de pesquisa onde também mostra uma das funções externas que é gerar boletos de pagamentos.*

Uma imagem contendo captura de tela

Descrição gerada automaticamenteA implantação do projeto deverá atender à infraestrutura definida para o sistema. Esta, por sua vez, deverá se basear em um servidor de aplicação, onde serão disponibilizadas as páginas e demais classes que compõem a aplicação.

O sistema de gerenciamento de Banco de Dados *MySQL* armazenará as tabelas utilizadas no Sistema SIGESC.



Figura 9 - Diagrama de Implantação - a.

Fonte: Elaborado pelo autor.

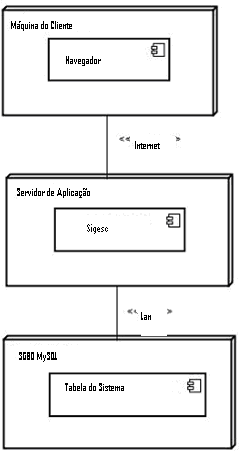


Figura 10 - Diagrama de Implantação - b.

Fonte: Elaborado pelo autor.

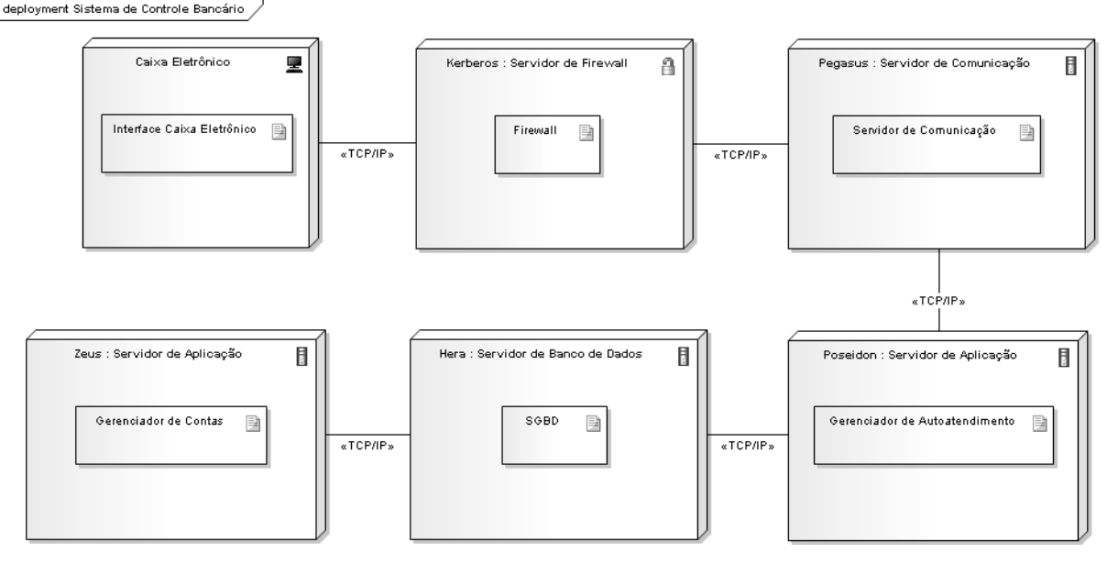


Figura 11 - Diagrama de Implantação -c.

Fonte: Elaborado pelo autor.

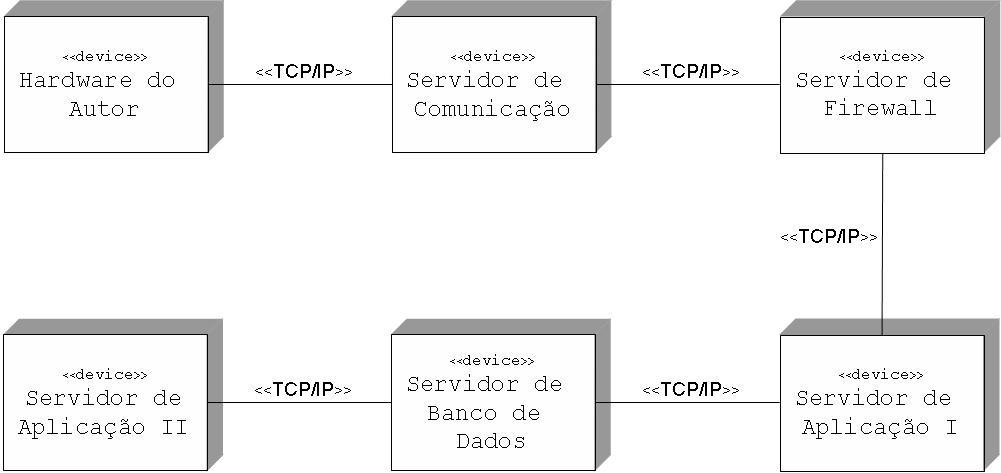


Figura 12 - Diagrama de Implantação - d.

Fonte: Elaborado pelo autor.

## DIAGRAMA DE ATIVIDADES

*Diagrama explicando a função de consultar matérias dentro do moodle, seja em EaD ou Matérias presenciais, onde pode encontrar as notas, número de faltas e desempenho nas disciplinas e no curso em geral.*

Uma imagem contendo captura de tela

Descrição gerada automaticamente

Figura 13 - Diagrama de Atividades – a consultar matérias.

Fonte: Elaborado pelo autor.

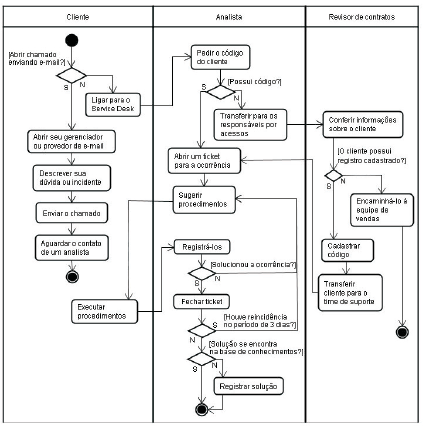


Figura 14 - Diagrama de Atividades - b.

Fonte: Elaborado pelo autor.

## DIAGRAMA DE SEQUÊNCIA

*Dois diagramas de sequência que explica a forma de o aluno solicitar a mudança de curso e como utilizar os pontos facultativos. O segundo explica a forma do aluno acessar o portal do aluno e utilizar as funções do moodle.*

*Uma imagem contendo captura de tela

Descrição gerada automaticamente*

Figura 15 - Diagrama de Sequência.

Fonte: Elaborado pelo autor.

*Uma imagem contendo captura de tela

Descrição gerada automaticamente*

Figura 31 - Diagrama de Sequência.

Fonte: Elaborado pelo autor.

## DIAGRAMA DE CASOS DE USO

*Diagrama onde mostra o contexto geral do sistema.*

*Uma imagem contendo texto, mapa

Descrição gerada automaticamente*

Figura 16 - Diagrama de Casos de Uso - a.

Fonte: Elaborado pelo autor.

Uma imagem contendo texto

Descrição gerada automaticamente

Figura 17 - Diagrama de Casos de Uso - b.

Fonte: Elaborado pelo autor.

## DIAGRAMA DE MÁQUINA DE ESTADO OU TRANSIÇÃO DE ESTADO

*Diagrama que mostra como o sistema efetua a cobrança em caso de atraso de pagamento da mensalidade.*

Uma imagem contendo captura de tela

Descrição gerada automaticamente

Figura 41 - Diagrama de Casos de Uso Mensalidade.

Fonte: Elaborado pelo autor.

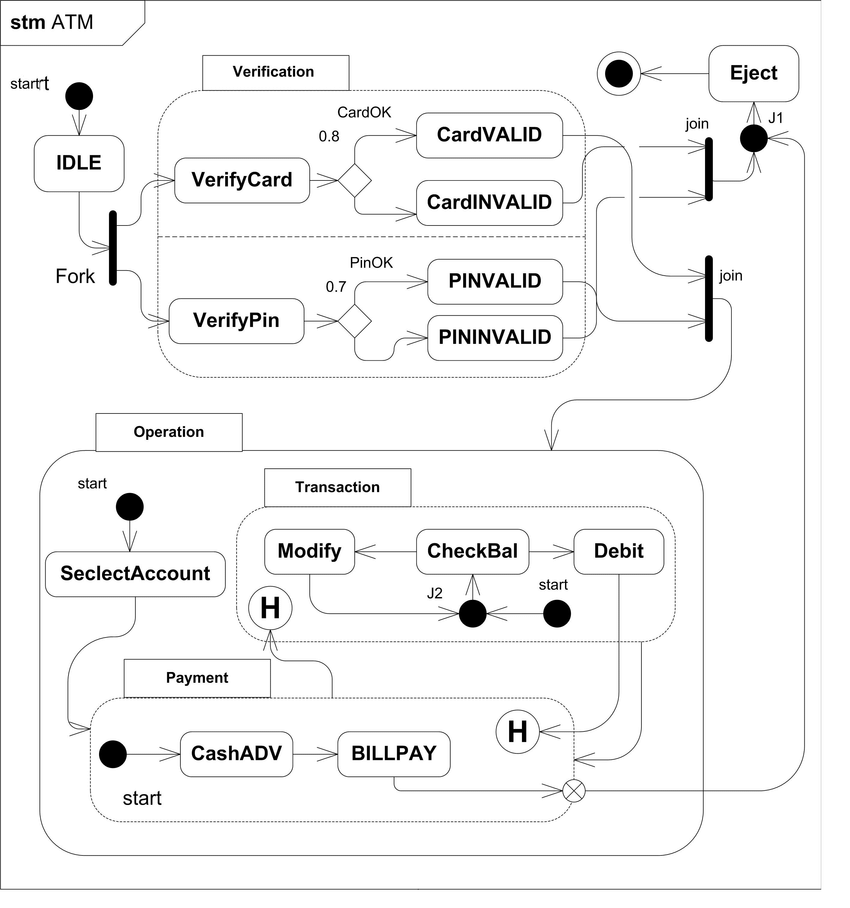


Figura 42 - Diagrama de Máquina de Estado ou Transição de Estado - b.

Fonte: Elaborado pelo autor.

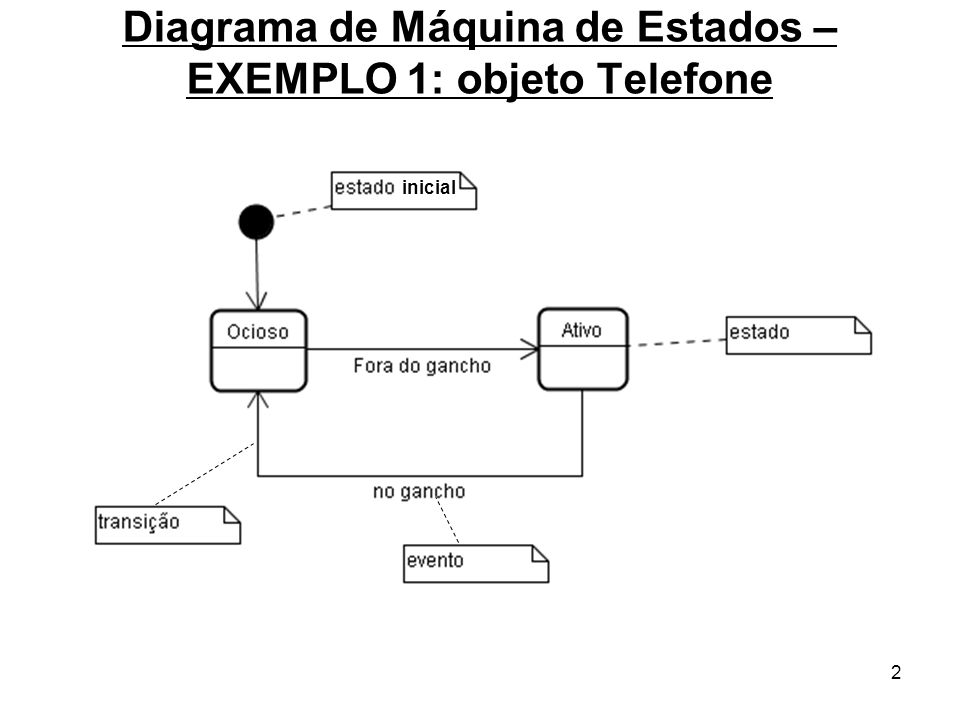


Figura 43 - Diagrama de Máquina de Estado ou Transição de Estado - c.

Fonte: Elaborado pelo autor.

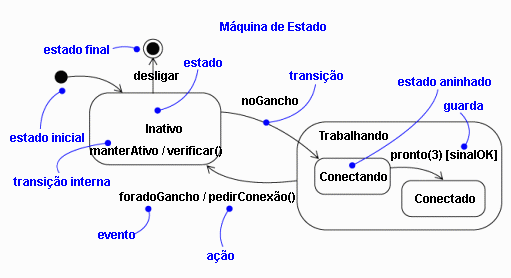


Figura 44 - Diagrama de Máquina de Estado ou Transição de Estado - d.

Fonte: Elaborado pelo autor.

# MODELAGEM DO BANCO DE DADOS

## MODELO ENTIDADE RELACIONAMENTO

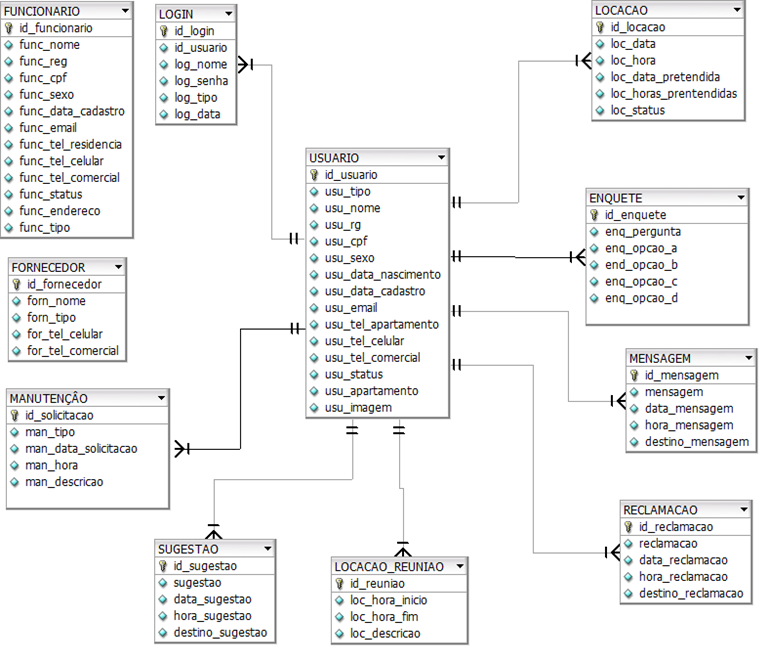


Figura 45 - MER: Modelo de Entidade-Relacionamento.

Fonte: Elaborado pelo autor.

## MODELO FÍSICO DE DADOS

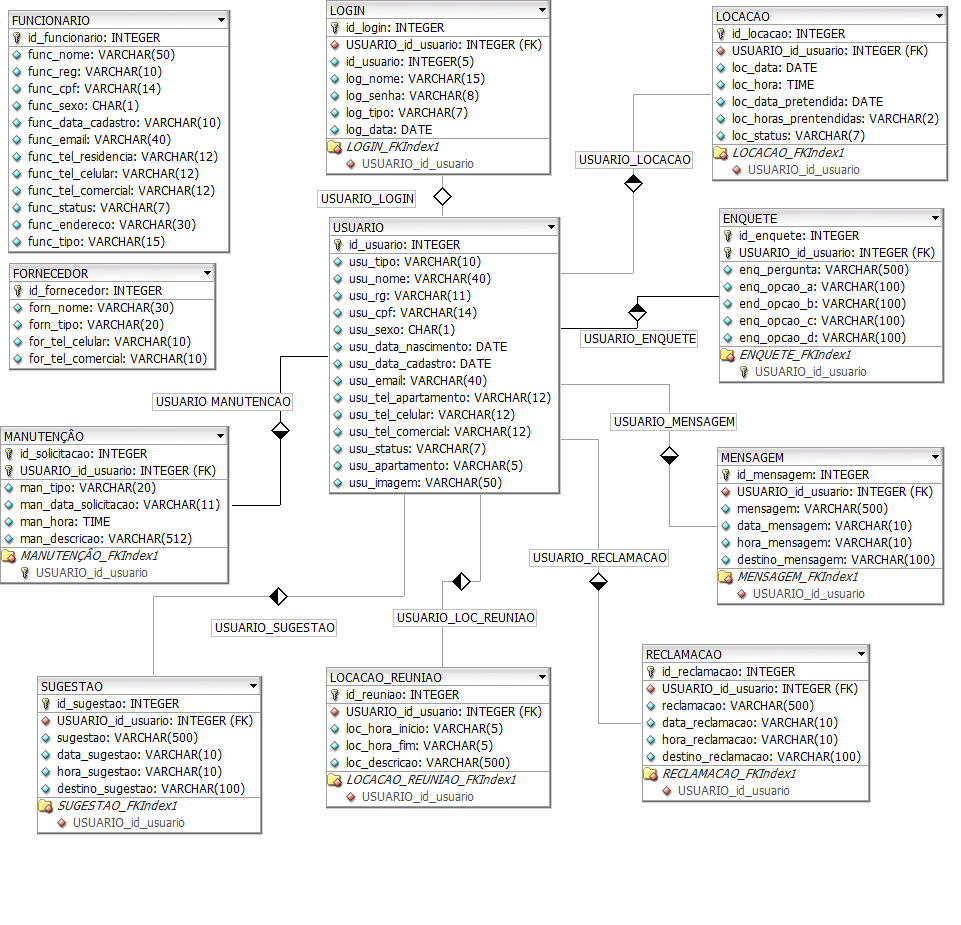


Figura 46 - MFD: Modelo Físico de Dados.

Fonte: Elaborado pelo autor.

# DICIONÁRIO DE DADOS

Contém características dos dados que serão utilizados no banco de dados do sistema SIGESC.

|  |
| --- |
| **ENQUETE** |
| |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | | ***ColumnName*** | ***DataType*** | ***PrimaryKey*** | ***NotNull*** | ***Flags*** | ***Default Value*** | ***Comment*** | ***AutoInc*** | | **id\_enquete** | ***INTEGER*** | *PK* | *NN* | *UNSIGNED* |  |  | AI | | **USUARIO\_id\_usuario** | ***INTEGER*** | *PK* | *NN* | *UNSIGNED* |  |  |  | | enq\_pergunta | VARCHAR(500) |  |  |  |  |  |  | | enq\_opcao\_a | VARCHAR(100) |  |  |  |  |  |  | | end\_opcao\_b | VARCHAR(100) |  |  |  |  |  |  | | enq\_opcao\_c | VARCHAR(100) |  |  |  |  |  |  | | enq\_opcao\_d | VARCHAR(100) |  |  |  |  |  |  | |
| |  |  |  | | --- | --- | --- | | ***IndexName*** | ***IndexType*** | ***Columns*** | | *PRIMARY* | *PRIMARY* | id\_enquete USUARIO\_id\_usuario | | *ENQUETE\_FKIndex1* | *Index* | USUARIO\_id\_usuario | |

|  |
| --- |
| **FORNECEDOR** |
| |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | | ***ColumnName*** | ***DataType*** | ***PrimaryKey*** | ***NotNull*** | ***Flags*** | ***Default Value*** | ***Comment*** | ***AutoInc*** | | **id\_fornecedor** | ***INTEGER*** | *PK* | *NN* | *UNSIGNED* |  |  | AI | | forn\_nome | *VARCHAR(30)* |  |  |  |  |  |  | | forn\_tipo | *VARCHAR(20)* |  |  |  |  |  |  | | for\_tel\_celular | *VARCHAR(10)* |  |  |  |  |  |  | | for\_tel\_comercial | *VARCHAR(10)* |  |  |  |  |  |  | |
| |  |  |  | | --- | --- | --- | | ***IndexName*** | ***IndexType*** | ***Columns*** | | *PRIMARY* | *PRIMARY* | id\_fornecedor | |

# TELAS DO PROTÓTIPO

*Modelo simples do acesso ao login do sistema em portal do aluno.*

*Uma imagem contendo captura de tela

Descrição gerada automaticamente*

# CONCLUSÃO

*Esse sistema tem como objetivo satisfazer as necessidades do cliente em organizar os dados de todos os alunos matriculados na instituição e fazer um levantamento de dados dos alunos com dificuldades nas matérias e que com intuito de motivar a não desistir com dicas e funções do sistema que facilitam o acesso com o conteúdo das aulas.*

*Com os objetivos atingidos o sistema terá cumprido o dever de facilitar o acesso dos estudantes as matérias e tirar as principais dúvidas com os professores através moodle com o chat de conversas.*

*Com os resultados atingidos, a instituição terá grandes chances de receber novos alunos com o intuito de se sentirem como os antigos estudantes que tiveram passagem na UCB, ou seja, aprender com o fácil acesso ao sistema e tirar as principais dúvidas com os professores online ou o material deixado em anexo.*

## TRABALHOS FUTUROS

*Trazer novos projetos inovadores para o sistema com o intuito de sempre manter atualizado com os avanços tecnológicos no mundo.*

# REFERÊNCIAS

LAUREANO, MARCOS A. P.; MORAES, PAULO E. S.. **Segurança como estratégia de gestão da informação**. Revista Economia & Tecnologia – ISSN 1415-451X, Vol. 8 – Fascículo 3 – P. 38-44. 2005.

# GLOSSÁRIO

*Ferramentas: Pencil, Astah, XMind.*

1. Cascata: Dentre os paradigmas da engenharia de *software* é o mais antigo, conhecido como clássico, utilizado quando o projeto flui de maneira linear desde sua concepção até a sua implantação. [↑](#footnote-ref-1)
2. JPA: é um *framework* para persistir objetos Java em um banco de dados. [↑](#footnote-ref-2)
3. *Annotation:* recurso disponível a partir da versão 5 do Java que possibilita a declaração de metadados nos objetos. [↑](#footnote-ref-3)
4. *Reflection:* pacote do Java que possibilita a criação de chamadas em tempo de execução do programa. [↑](#footnote-ref-4)