



**Fundação Educacional do Município de Assis
Instituto Municipal de Ensino Superior de Assis
Campus "José Santilli Sobrinho"**

Marcos Vinício Honório

**APLICATIVO ANDROID PARA AVALIAÇÃO DE CONTROLE DE
QUALIDADE UTILIZANDO O PROGRAMA 5S**

Assis - SP

2014

MARCOS VINÍCIO HONÓRIO

**APLICATIVO ANDROID PARA AVALIAÇÃO DE CONTROLE DE
QUALIDADE UTILIZANDO O PROGRAMA 5S**

Trabalho de Conclusão de Curso apresentado ao Curso de Análise e Desenvolvimento de Sistemas do Instituto Municipal do Ensino Superior de Assis – IMESA e Fundação Educacional do Município de Assis – FEMA, como requisito para a obtenção do Certificado de Conclusão.

Orientadora: Esp. Diomara Martins Reigato Barros
Área de Concentração: Desenvolvimento de software

Assis - SP

2014

FICHA CATALOGRÁFICA

HONÓRIO, Marcos Vinício

Aplicativo Android para Avaliação de Controle de Qualidade Utilizando Programa 5S / Marcos Vinício Honório. Fundação Educacional do Município de Assis – FEMA – Assis, 2013.
40Pag.

Orientador: Prof^a. Diomara Martins Reigato Barros

Trabalho de Conclusão de Curso – Instituto Municipal de Ensino Superior de Assis – IMESA.

1. Programa 5s. 2. Google Android. 3. Mobile.

CDD: 001.6

Biblioteca da FEMA

DEDICATÓRIA

Dedico este trabalho a Deus, responsável pelas forças que me guiaram durante os três anos de curso e pelo dom da vida, sem as quais eu não encontraria em mim capacidade de prosseguir.

AGRADECIMENTOS

Em primeiro lugar a Deus, por ter me dado sabedoria necessária e sempre ter estado ao meu lado na realização deste e demais trabalhos no decorrer do curso.

À minha Esposa Luciana Alves Martins Honório, por me apoiar e ter dado forças dentro de suas possibilidades e em momentos difíceis, foi uma peça chave em minha formação.

Agradeço aos meus pais Francisco Honório e Maria de Lourdes. Aos meus amigos por estarem ao meu lado nos bons e maus momentos nos últimos anos.

À professora Diomara Martins Reigato Barros, pela orientação e apoio na execução deste trabalho. A todos os professores do curso de Análise de Sistemas pelos ensinamentos transferidos ao longo dos anos e que somados compõem o conhecimento indispensável à minha formação. Aos colegas de turma no curso de graduação cujo companheirismo eu não poderia deixar de agradecer.

Em especial ao Emerson Marcelino da Silveira, pelo apoio e amizade que foram indispensáveis nos anos decorridos do curso.

RESUMO

Este trabalho tem por objetivo principal ser uma ferramenta de auxílio no critério de avaliação de controle de qualidade utilizando o Programa 5s em setores de empresas sucroenergético. A necessidade de haver um ótimo controle nos processos, melhorias e qualidade nos serviços prestados, a busca pela qualidade total, onde o cenário atual hoje é cada vez maior a concorrência entre às empresas. É apresentado o projeto de desenvolvimento de um software para avaliação de controle de qualidade como alternativa trazer mais rapidez no processo de avaliação e melhorar a qualidade da avaliação feita pelo auditor.

O motivo para escolha do tema é a certeza de que o aplicativo irá satisfazer as necessidades da diretoria da empresa e do auditor. O trabalho de avaliação se tornará completo e contará com a precisão do aplicativo com acompanhamento em tempo real das notas, médias e informações que poderão ser acessadas direto do sistema. Com aplicativo o auditor poderá fotografar os campos dos setores, as imagens capturadas serão armazenadas nas pastas de cada setor avaliado. O aplicativo irá funcionar em dispositivos móveis, como *Tablets* ou *Smartphones* munidos de sistema Android. A interface da aplicação é de fácil compreensão, possibilitando navegação entre as telas de forma intuitiva, com ícones e botões auto compreensíveis. Foi utilizada UML (Unified Modeling Language) para padronização dos diagramas e entendimento do aplicativo. Para programação e desenvolvimento foram utilizadas as tecnologias JAVA e Google Android, por serem tecnologias em alta no mercado de TI e juntas propiciarem as melhores soluções para desenvolvimento móvel.

Palavras – chave: Programa 5s, Google Android, Mobile.

ABSTRACT

This work has as main goal to be a tool to support the evaluation criteria of quality control program using the 5s in the sugarcane business sectors. The need for optimal control processes, improvements and quality services, the search for total quality, where the current scenario today is ever greater competition between the companies. The project to develop software for evaluation of quality control is presented as an alternative to bring faster in the evaluation process and improve the quality of assessment made by the auditor.

The reason for choosing the topic is sure that the application will meet the needs of the board of directors and the auditor. The evaluation work will become full and will include the accuracy of the application with real-time monitoring of the notes, medium and information that can be accessed directly from the system. Application with the auditor can photograph the fields of sectors, the captured images are stored in folders of each sector evaluated. The application will work on mobile devices such as tablets or smartphones equipped with Android system. The application interface is easy to understand, allowing navigation between screens intuitively, with self understandable icons and buttons . UML (Unified Modeling Language) was used for standardization of diagrams and understanding of the application . Programming and development for the Google Android and JAVA technologies were used , because they are in high technologies in the IT market and together they encourage the best solutions for mobile development .

key – words: 5s Program, Google Android , Mobile.

LISTA DE ILUSTRAÇÕES

Figura 1 - Diagrama de Casos de Uso -----	16
Figura 2 - Manter Setor -----	17
Figura 3 - Manter Campo -----	18
Figura 4 - Manter Setor/Campos -----	20
Figura 5 - Manter Normas -----	21
Figura 6 - Incluir Avaliação -----	22
Figura 7 - Manter Campos Avaliados -----	24
Figura 8 - Diagrama de Classes -----	25
Figura 9 - Diagrama de Atividades -----	26
Figura 10 - Work Breakdown Structure -----	28
Figura 11 - Sequenciamento das Atividades -----	29
Figura 12- Custo Analise e Programação -----	30
Figura 13 - Custo de Equipamentos -----	31
Figura 14 - Custo Total do Projeto -----	31
Figura 15 - Organização do Projeto -----	32
Figura 16 - Tela Menu Principal -----	45
Figura 17 - Tela Setor -----	46
Figura 18 - Tela Avaliação -----	47

SUMÁRIO

1. INTRODUÇÃO	10
1.1 OBJETIVO.....	11
1.2 JUSTIFICATIVAS	11
1.3 PÚBLICO ALVO	11
1.4 MOTIVAÇÃO.....	11
1.5 ESTRUTURA DO TRABALHO.....	12
1.6 METODOLOGIA DE DESENVOLVIMENTO	12
2.DEFINIÇÃO CONCEITUAL	13
2.1 GOOGLE ANDROID	13
2.2 ANDROID SDK	13
2.3 JAVA	14
2.4 PROGRAMA 5S	14
3.ANALISE E ESPECIFICAÇÃO	166
3.1 DIAGRAMA DE CASOS DE USO	166
3.2 ESPECIFICAÇÃO DE CASOS DE USO	177
3.3 DIAGRAMA DE CLASSES.....	265
3.4 DIAGRAMA DE ATIVIDADES.....	266
4.ESTRUTURA DO PROJETO	277
4.1 ESTRUTURA ANALÍTICA DO PROJETO.....	288
4.2 SEQUENCIAMENTO DAS ATIVIDADES.....	299
4.3 ORÇAMENTO	29
4.3.1 Análise e Programação.....	30
4.3.2 Equipamentos.....	30
4.3.3 Custo total do Projeto	31
5.IMPLEMENTAÇÃO DO APLICATIVO	322
5.1 ORGANIZAÇÃO DO PROJETO.....	452
5.2 CLASSES	453
5.2.1 Classe PrincipalActivity	33
5.2.2 Classe AvaliacaoActivity	35
5.3 INTERFACE DO APLICATIVO.....	45
6.CONCLUSÃO	48
REFERÊNCIAS	49
ANEXOS	50
CRONOGRAMA.....	50

1. INTRODUÇÃO

Para garantir a sobrevivência de uma empresa nos dias de hoje necessitamos de uma forma precisa de avaliação, onde o fator primordial é a qualidade dos produtos e serviços prestados de um modo que a empresa consiga ganhar a preferência do seu cliente.

A avaliação de controle de qualidade é uma norma padrão adotada por empresas de pequeno e médio porte, com finalidade de alcançar um sistema de padronização e as normas ISO (International Organization for Standardization). Esse sistema ainda está sendo implantado na Usina Água Bonita, uma empresa responsável pela produção de etanol, açúcar e energia, a empresa busca melhorar a qualidade de seus produtos, ou seja, a qualidade total em modo geral a preferência do consumidor.

No caso da empresa Água Bonita ainda utiliza como método de avaliação o programa 5s, que é a primeira fase de adoção de qualidade total que antecede o ISO, o 5s é uma maneira de melhorar o comportamento das pessoas, tanto dentro da empresa como também no seu meio de vida.

O 5s é acompanhado de 5 palavras Japonesas SEIRI (senso de arrumação), SEITON (senso de ordenação), SEISOH (senso de limpeza), SEIKETSU (senso de asseio), SHITSHUKE (senso de autodisciplina). Garante maior produtividade, redução de despesas, melhoria da qualidade de produtos e serviços prestados, menos acidentes e maior satisfação das pessoas em seu ambiente de trabalho. A empresa Água Bonita avalia seus setores com o programa 5s, de maneira manual utilizando planilhas para anotação, fazendo levantamentos dessas informações e aplicando sugestões aos funcionários do setor auxiliando e dando dicas para melhor desempenho.

1.1 OBJETIVO

O objetivo do trabalho é desenvolver um sistema que avalie os setores de uma empresa sucroenergético, utilizando as normas do programa 5s. O sistema irá avaliar através de um dispositivo móvel tablet ou celular, onde o responsável pela auditoria irá averiguar se os setores da empresa estão de acordo com as normas estabelecidas. O sistema também trará informações úteis que poderão auxiliar no critério de avaliação do auditor, usará como método semelhante à pesquisa de satisfação com base de valores constados no sistema, o auditor informará o sistema que automaticamente irá fazer um relatório, que será apresentado no final da auditoria. Esse relatório constará uma nota representada em porcentagem e dados estatísticos, que será entregue a diretoria da empresa.

1.2 JUSTIFICATIVAS

Devido à demanda por competição entre as empresas sucroenergético, ganha o mercado a empresa que oferecer qualidade total em seus produtos e serviços prestados. Como o critério de avaliação de 5S desse tipo de empresa é demorado devido à quantidade de setores. O sistema busca melhorar a qualidade da avaliação feita durante a auditoria realizada, facilitando o critério de avaliação por meio tecnológico que trás maior agilidade e precisão nos resultados de pesquisas e dados estatísticos.

1.3 PÚBLICO ALVO

O público a que se destina a utilização do sistema é composto por empresas do ramo sucroenergético.

1.4 MOTIVAÇÃO

O estudo sobre a linguagem JAVA, bem como o estudo da plataforma Google Android, colocando em prática no desenvolvimento de um aplicativo, foi a grande

motivação para este trabalho, considerando que irá enriquecer os conhecimentos adquiridos na faculdade.

1.5 ESTRUTURA DO TRABALHO

Este trabalho está dividido em capítulos, estruturados de forma clara e organizada, favorecendo ao leitor um amplo entendimento dos conceitos abordados, e metodologias utilizadas para sua realização.

O trabalho tem a seguinte estrutura de capítulos:

1. Introdução
2. Definição Conceitual
3. Análise e Especificação
4. Estrutura do Projeto
5. Implementação do aplicativo
6. Conclusão

1.6 METODOLOGIA DE DESENVOLVIMENTO

A aplicação que será executada utilizará dispositivos móveis e será desenvolvida para a plataforma Google Android. O software vai ser instalado em um tablet ou smartphone que será escrito na linguagem de programação Java.

Para o desenvolvimento do aplicativo serão utilizadas as seguintes tecnologias:

- UML (Unified Modeling Language) para padronização dos diagramas e entendimento do aplicativo.
- Eclipse Indigo Service Release 2, com o plug-in ADT (Android Development Tools): ambiente de desenvolvimento de software.

2.DEFINIÇÃO CONCEITUAL

2.1 GOOGLE ANDROID

O Google android é sistema operacional de código aberto (open source) criado pela Google, onde o seu principal objetivo é o desenvolvimento de aplicativos para dispositivos móveis. Sua plataforma é baseada no *Kernel* do sistema operacional *Linux*, com uma interface rica e amigável.

O Android é uma plataforma completa para dispositivos móveis, pois envolve um pacote com programas para celulares, já inclusos um sistema operacional, aplicativos, middleware e interface de usuário. Esta plataforma foi criada para ser verdadeiramente de código aberto e com o objetivo de permitir aos desenvolvedores a criação de aplicativos móveis capazes de tirar proveito de toda a capacidade que um aparelho móvel pode oferecer. Um exemplo clássico do que foi dito é a capacidade de uma aplicação interagir com as funcionalidades do núcleo do dispositivo como efetuar chamadas, enviar mensagens de texto, utilizar a câmera, permitindo que o desenvolvedor faça adaptações e evoluam ainda mais estas funcionalidades. Por ser uma plataforma de código aberto esta por sua vez permite que o desenvolvedor faça adaptações, visando incorporar novas tecnologias conforme forem surgindo, visto que existem várias comunidades de desenvolvedores trabalhando em conjunto para construção de aplicações cada vez mais inovadoras (PEREIRA; SILVA, 2009).

2.2 ANDROID SDK

O Android SDK é o software adotado para desenvolver aplicações no Android, este por sua vez fornece uma API completa para a linguagem Java, um emulador para simular um celular e ferramentas de desenvolvimento necessárias para construir, testar e depurar aplicativos para o Android (FARTO, 2010).

O Android SDK (Software Development Kit) é o software usado para o desenvolvimento de aplicações Android, ele fornece ferramentas utilitárias e uma API completa para a linguagem Java, com todas as classes necessárias. Com o plug-in ADT para o Eclipse, ocorre a integração do ambiente de desenvolvimento Java com o

emulador disponível no SDK, pois é possível iniciá-lo diretamente do Eclipse, instalar a aplicação e depurar o código-fonte como em qualquer outro software Java. Para facilitar os testes em dispositivos reais e obter maior produtividade de desenvolvimento, é possível plugar um celular ou tablet na porta USB do computador e executar os aplicativos diretamente neste aparelho. Isto proporciona ao desenvolvedor uma análise do desempenho e usabilidade da aplicação ainda na fase de desenvolvimento (LECHETA, 2010).

2.3 JAVA

Java é uma linguagem de programação orientada a objeto. Atualmente umas das linguagens que mais vem crescendo no mercado de TI devido sua robustez, simplicidade e alto nível de segurança. A compilação do código ocorre para JVM (Java Virtual Machine) considerada uma linguagem Universal e Interpretada.

A arquitetura ou plataforma Java tem basicamente dois componentes: a JVM e a interface de programação de aplicações (API – Application Programming Interface). A JVM emula o ambiente computacional e é a responsável pela portabilidade da linguagem, pois, uma vez instalado na máquina, pode-se executar qualquer sistema em Java, independente de sistema operacional. Os “bytecodes” são uma espécie de código assembler para a JVM. Este código é otimizado pela JVM, que o interpreta, gerando e passando ao hardware em que esta instalada, os comandos necessários. Por isso que se diz que os sistemas Java são compilados e interpretados (GRADVOHL, 2008).

2.4 PROGRAMA 5S

Os cinco sensores que dão nome ao Programa 5S derivam de cinco palavras japonesas seiri, seiton, seisou, seiketsu e shitsuke que em português são conhecidas como os sensores de utilização, ordenação, limpeza, asseio e autodisciplina.

O Programa 5s tem como essência a disciplina, foi com esta intenção que os japoneses no final da década de 60, desenvolveram o Programa 5S.

As organizações e empresas devido ao processo de globalização e o aumento da competitividade, necessitaram melhorar o nível de desempenho, buscando implantar

os Sistemas de Qualidade como alternativa para sobreviverem a mudanças e avanços tecnológicos.

3. ANÁLISE E ESPECIFICAÇÃO

Este capítulo apresenta informações obtidas a partir da análise de requisitos do Aplicativo.

3.1 DIAGRAMA DE CASOS DE USO

O Diagrama de Casos de Uso tem como objetivo representar graficamente os requisitos do sistema. Nele se encontram os atores, os processos que serão automatizados e o relacionamento entre ambos (LIMA, 2012).

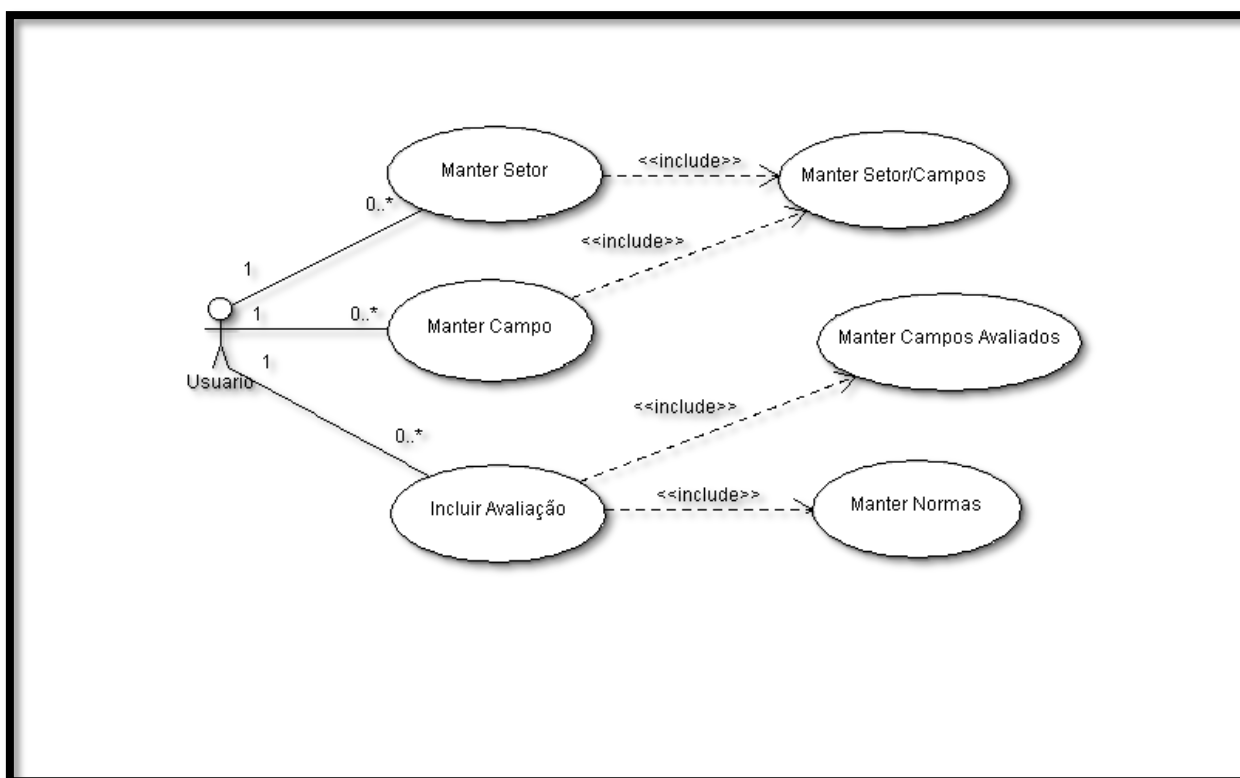


Figura 1 Diagrama de Casos de Uso

Lista de Casos de uso

1. Manter Setor
2. Manter Campo
3. Setor/Campos
4. Manter Normas
5. Iniciar Avaliação
6. Campos Avaliados

3.2 ESPECIFICAÇÃO DE CASOS DE USO

CASO DE USO – Manter Setor

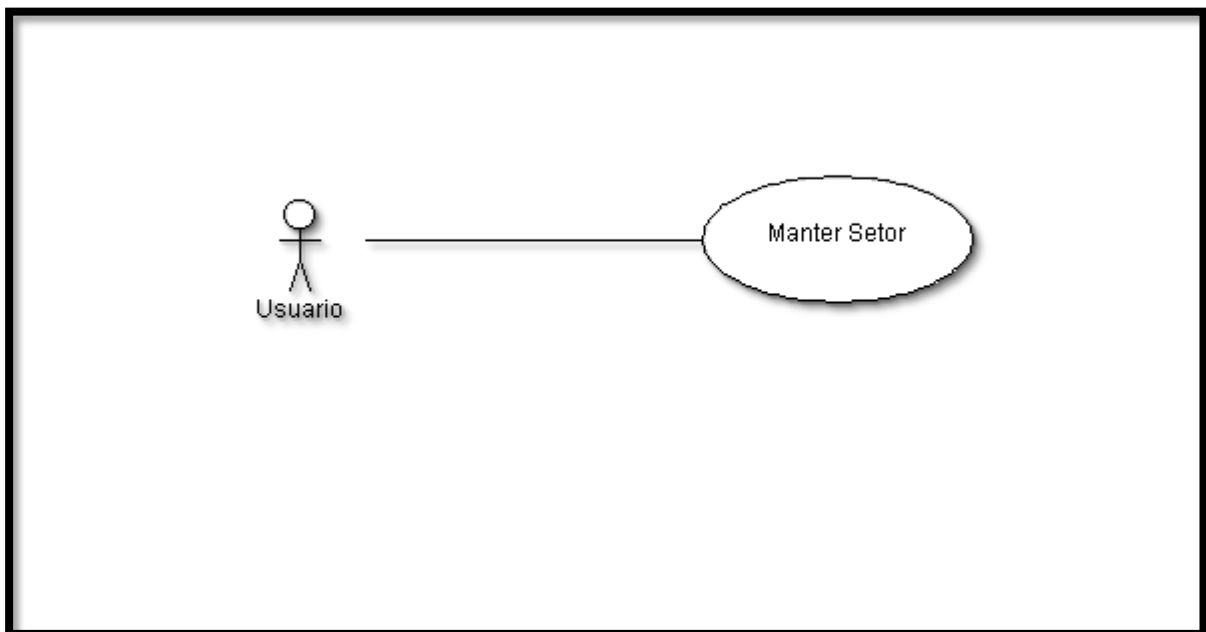


Figura 2 - Manter Setor

1. Finalidade/Objetivo

Cadastrar novos setores no sistema.

2. Atores

Usuário.

3. Pré-Condições

Estar no Menu Principal.

4. Evento Inicial

O usuário deve selecionar a opção “Cadastrar Setor”.

5. Fluxo Principal

- a) O Usuário seleciona a opção “Cadastrar Setor”.
- b) O caso de uso é iniciado.
- c) O usuário cria um novo código que o novo setor passará a ter.
- d) O usuário digita o nome do setor a ser cadastrado.
- e) O usuário salva o novo setor.

6. Casos de Testes

Cancelar o caso de uso.

CASO DE USO – Manter Campo

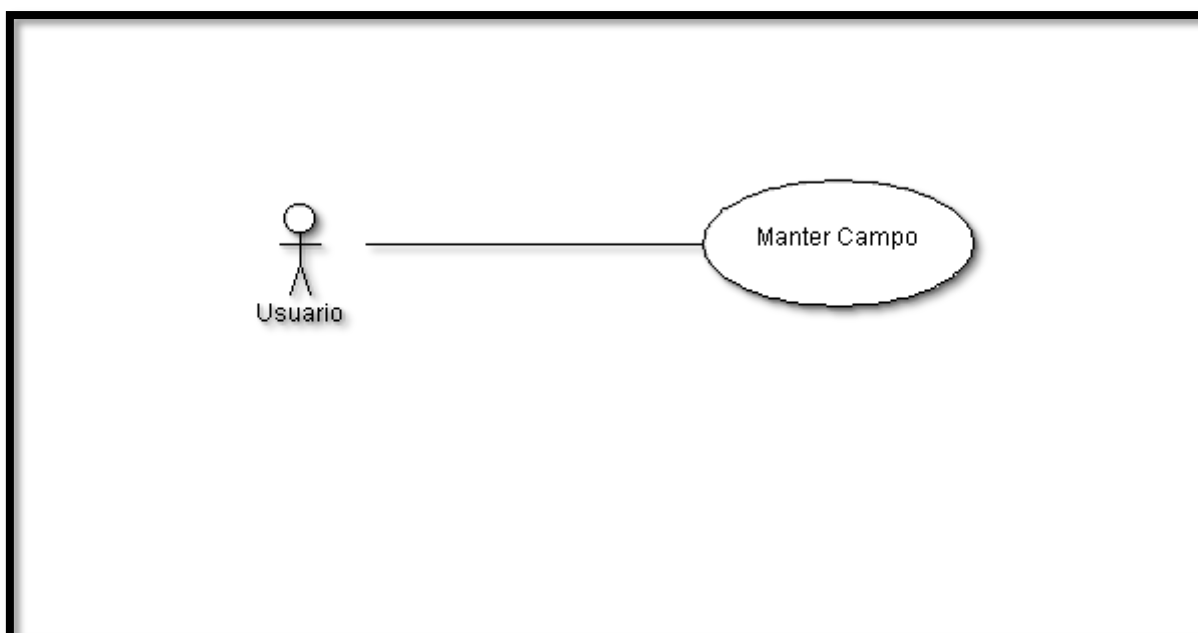


Figura 3 – Manter Campo

1. Finalidade/Objetivo

Cadastrar novos campos no sistema.

2. Atores

Usuário.

3. Pré-Condições

Estar no “Menu Principal”.

4. Evento Inicial

O usuário deve selecionar a opção “Cadastrar Campo”.

5. Fluxo Principal

- a) O Usuário seleciona a opção “Cadastrar Setor”.
- b) O caso de uso é iniciado.
- c) O usuário cria um novo código que o novo campo passará a ter.
- d) O usuário digita o nome do campo a ser cadastrado.
- e) O usuário salva o novo campo.

6. Casos de Testes

Cancelar o caso de uso.

CASO DE USO – Manter Setor/Campos

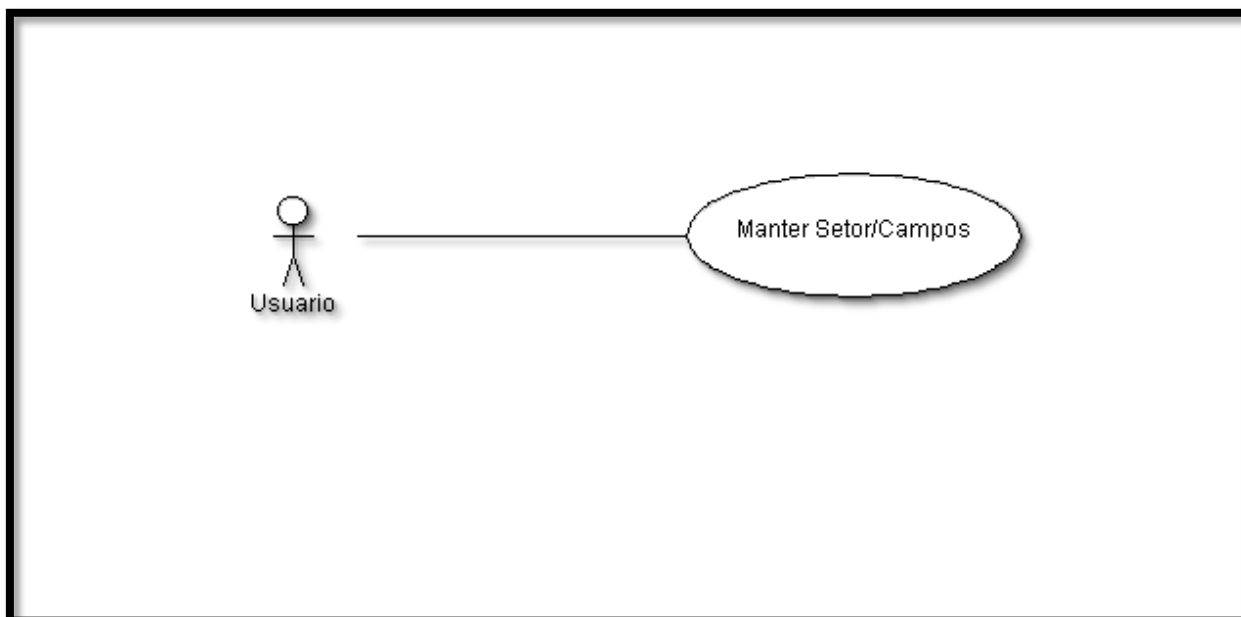


Figura 4 – Manter Setor/Campos

1. Finalidade/Objetivo

Mostrar os nomes e códigos dos setores e campos cadastrados no sistema.

2. Atores

Usuário.

3. Pré-Condições

Estar no Menu Principal.

4. Evento Inicial

O usuário deve selecionar a opção Setor/Campos.

5. Fluxo Principal

- a) O Usuário seleciona o Setor/Campos.
- b) O caso de uso é iniciado.
- c) O usuário visualiza o código e nome dos setores e campos cadastrados.
- d) O usuário pode escolher qual Setor e quantos campos serão avaliados.

7. Casos de Testes

Cancelar o caso de uso.

CASO DE USO – Manter Normas

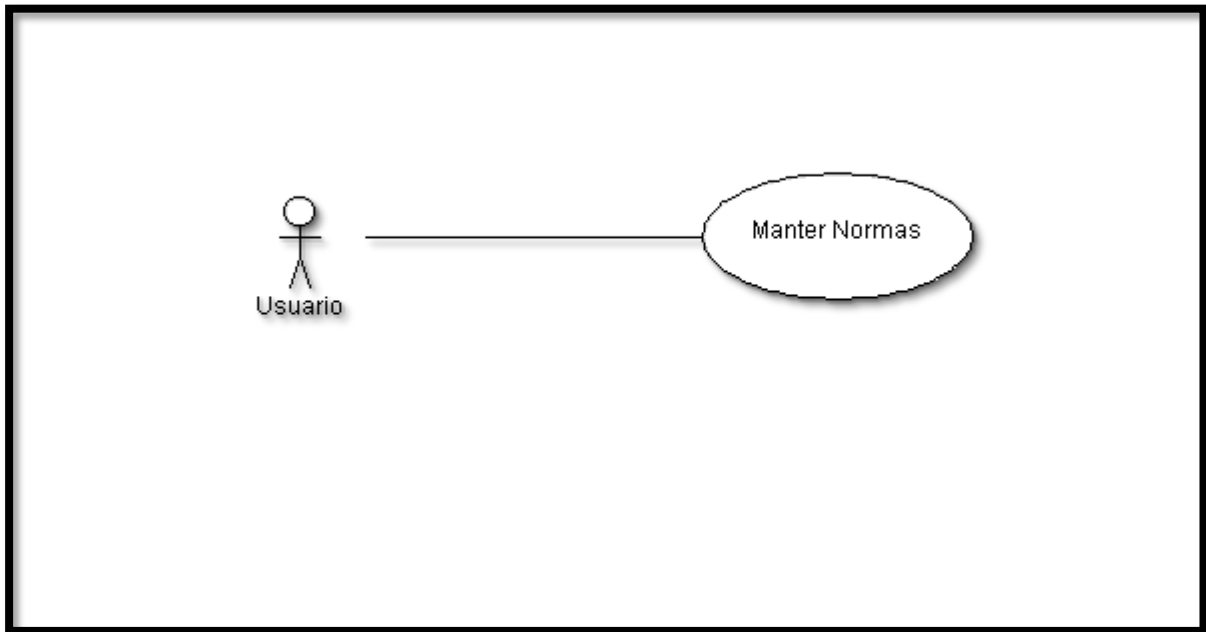


Figura 5 - Manter Normas

1. Finalidade/Objetivo

Tela onde constam normas de avaliação 5s.

2. Atores

Usuário.

3. Pré-Condições

Estar no Menu Principal.

4. Evento Inicial

O usuário deve estar avaliando o Setor.

5. Fluxo Principal

- a) O Usuário seleciona a Informação.
- b) O sistema abre o Menu de Informações.
- c) O usuário pode utilizar as Informações contidas como auxílio na avaliação.
- d) O sistema da opção de inserir, excluir e salvar as normas do 5s.

CASO DE USO – Incluir Avaliação

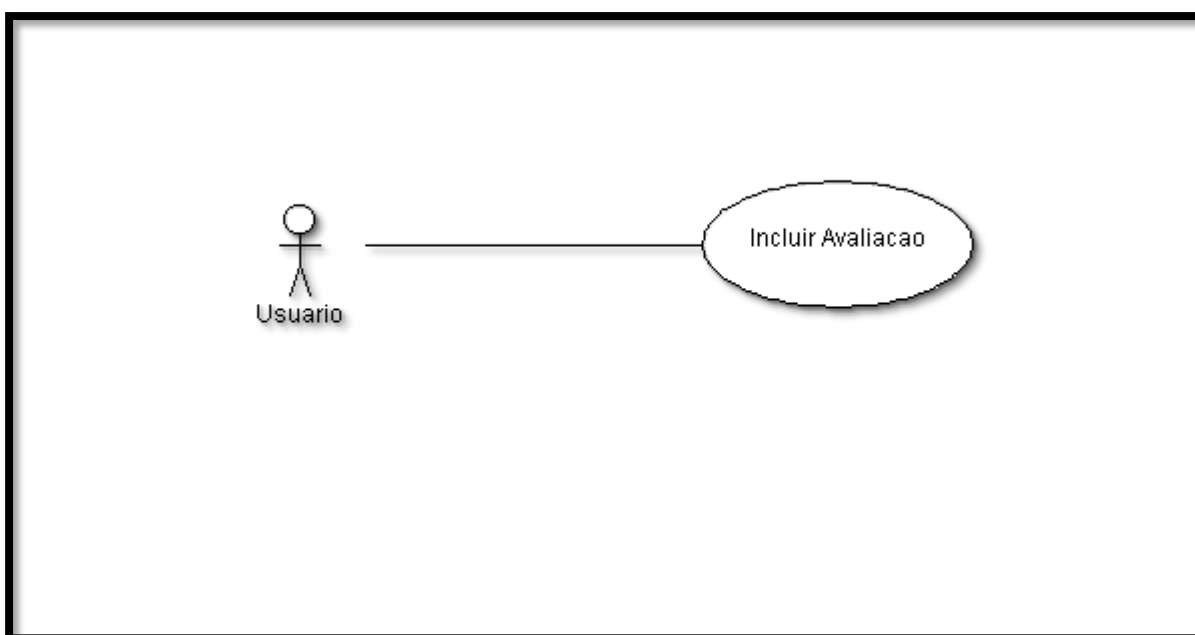


Figura 6 – Incluir Avaliação

1. Finalidade/Objetivo

O usuário cria um código de avaliação, que poderá ser consultado.

2. Atores

Usuário.

3. Pré-Condições

Estar no Menu de Avaliação 5s.

4. Evento Inicial

O usuário deve selecionar a opção "Avaliação".

5. Fluxo Principal

- a) O Usuário informa um novo código de avaliação.
- b) O usuário informa a data da avaliação.
- c) O usuário informa o nome do setor.
- d) O usuário informa o nome ao sistema.

6. Fluxos Alternativos

A1-Cancela a Operação.

- a) O usuário cancela o caso de uso. O Sistema retorna ao Menu Principal.

7. Casos de Testes

Cancelar o caso de uso.

CASO DE USO – Manter Campos Avaliados

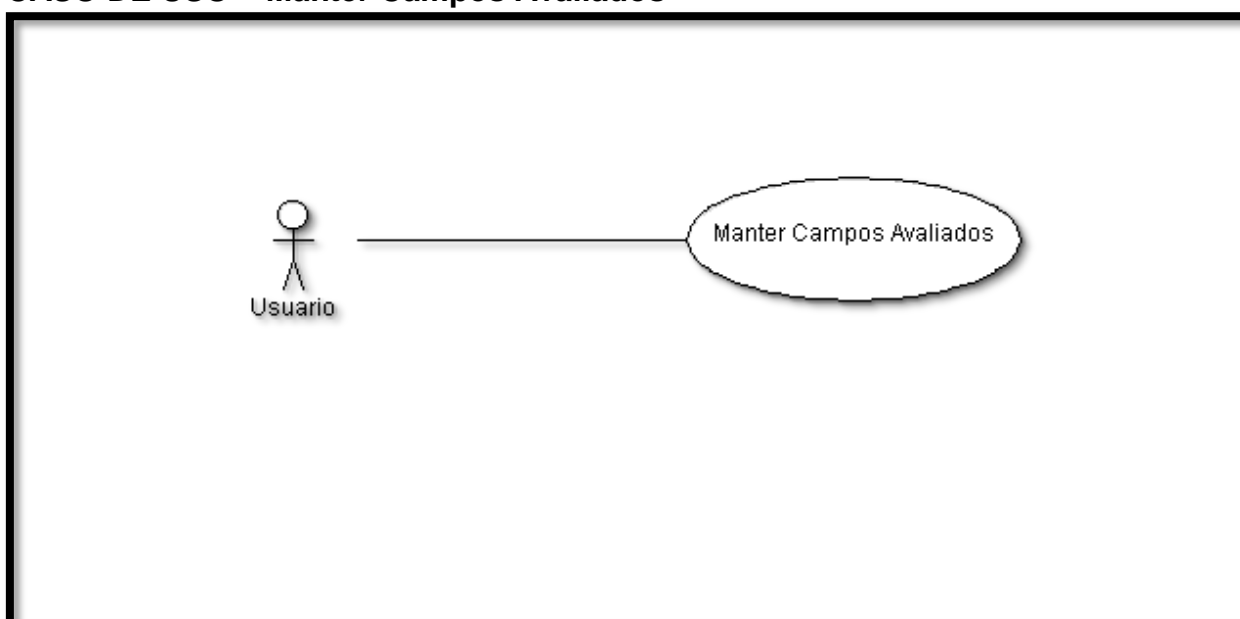


Figura 7 – Manter Campos Avaliados

1. Finalidade/Objetivo

Começa a avaliar utilizando notas de 0,00 a 1,00. O usuário digita a nota de acordo com os critérios de avaliação. Nesta tela também o usuário pode optar para utilização da câmera para registrar pontos dos setores.

2. Atores

Usuário.

3. Pré-Condições

Estar no Menu de Avaliação 5s.

4. Evento Inicial

Informar todas as notas dos campos para gerar a média.

5. Fluxo Principal

- a) O Usuário informa as notas em todos os campos.
- b) O usuário encerra a avaliação e a média é calculada e salva no banco de dados.
- c) O sistema gera um relatório com dados da avaliação realizada.

7. Casos de Testes

O sistema faz um levantamento com os Setores que necessitam de melhorias.

3.3 DIAGRAMA DE CLASSES

O diagrama de classes permite visualização estática do projeto para posterior especificação e documentação do modelo estrutural.

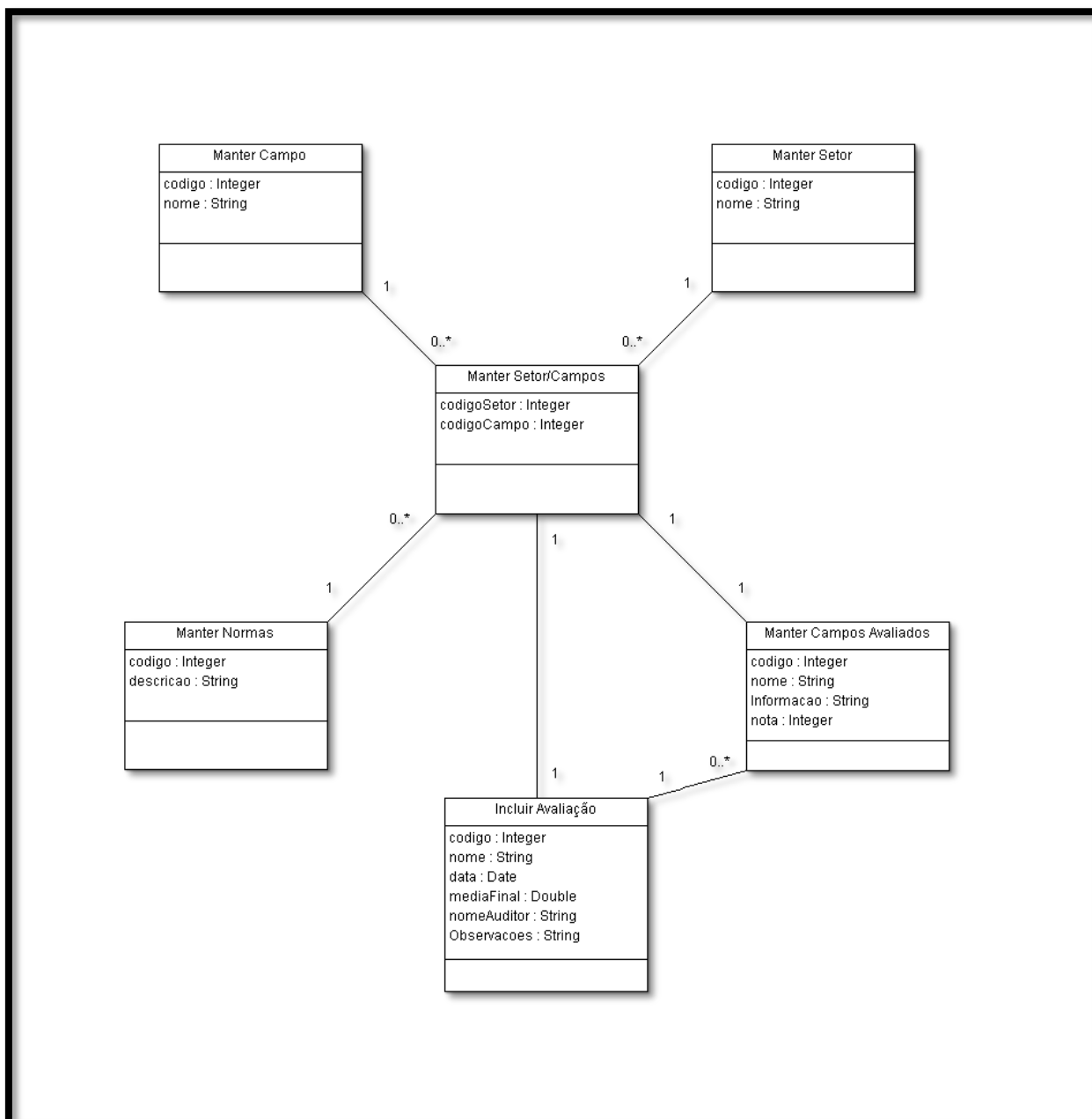


Figura 8 - Diagrama de Classes

3.4 DIAGRAMA DE ATIVIDADES

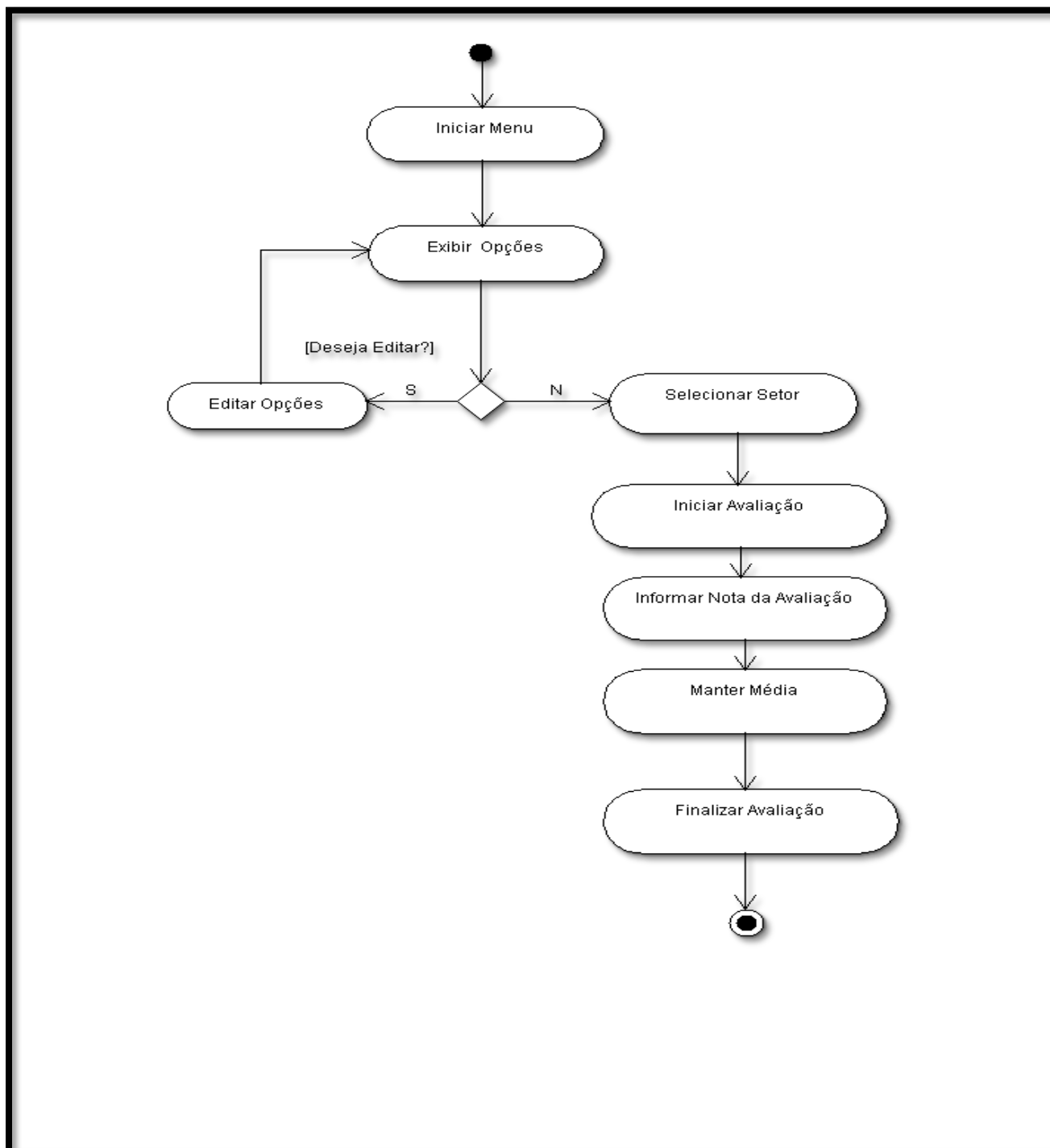


Figura 9 - Diagrama de Atividades

4. ESTRUTURA DO PROJETO

Este capítulo apresenta a metodologia de desenvolvimento do trabalho. As fases e etapas de sua construção serão ilustradas no diagrama da Estrutura Analítica do Projeto, também conhecida como WBS (Work Breakdown Structure). O atual capítulo apresenta também o diagrama de sequência de atividades e orçamento do sistema.

4.1 ESTRUTURA ANALÍTICA DO PROJETO

Através da Estrutura Analítica do Projeto é possível obter uma visão ampla, facilitando a divisão do conjunto de atividades a serem executadas em cada fase.



Figura 10 - Work breakdown structure

4.2 SEQUENCIAMENTO DAS ATIVIDADES

No diagrama de Sequenciamento de Atividades é apresentado o tempo de duração de cada atividade desenvolvida ao longo do projeto.

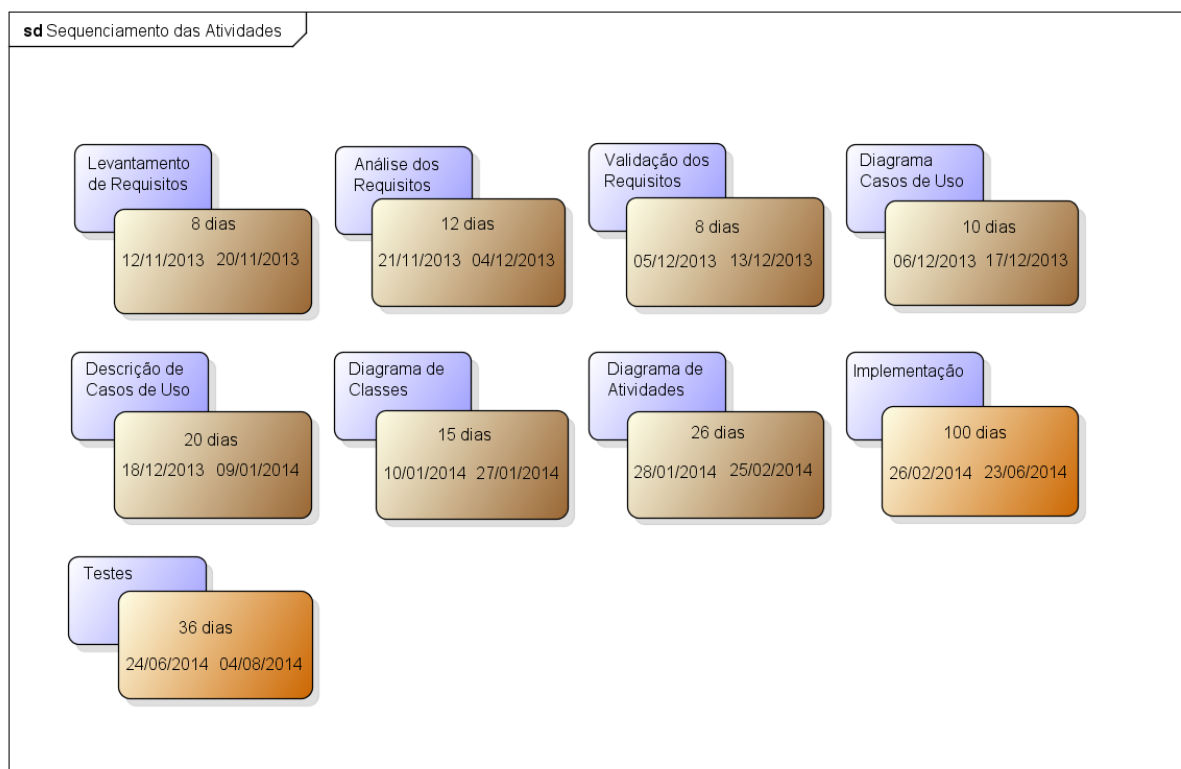


Figura 11 - Sequenciamento das Atividades

4.3 ORÇAMENTO

Os recursos necessários para a análise e desenvolvimento do aplicativo objeto desse trabalho são:

- 01 Analista de Sistema
- 01 Programador
- 01 Notebook
- 01 Tablet

4.3.1 Análise e Programação

Os custos com análise e programação sucederam de acordo com a figura abaixo, onde cada uma das 400 horas investidas em análise custou R\$ 20,00 totalizando o valor de R\$ 8.000,00. Em programação, foram investidas 144 horas, com valor unitário de R\$ 15,00 compondo um total de R\$ 2.160,00. O gasto total em análise e programação foi de R\$ 10.160,00.

Analista de Sistemas			
Analista	Quantidade de Horas	Valor da Hora	Total
Marcos Vinício Honório	400	R\$ 20,00	R\$ 8.000,00
Programador			
Analista	Quantidade de Horas	Valor da Hora	Total
Marcos Vinício Honório	144	R\$ 15,00	R\$ 2.160,00
TOTAL			R\$10.160,00

Figura 12 - Custo Análise e Programação

4.3.2 Equipamentos

Os custos com equipamentos necessários para o desenvolvimento do aplicativo foram um Notebook Acer, no valor de R\$ 2.200,00 com depreciação diária de R\$ 3,05, sendo que o gasto total do equipamento corresponde ao valor de R\$ 727,00. Foi necessário a compra de um tablet no valor de R\$ 290,00, com depreciação diária de R\$ 0,39, totalizando o valor de R\$ 98,50. O total geral gasto em equipamentos é de R\$ 825,50.

Equipamentos			
Equipamento	Valor	Depreciação Diária	Total
Notebook Acer	R\$ 2.200,00	R\$ 3,05	R\$ 727,00
Tablet Genesis	R\$ 290,00	R\$ 0,39	R\$ 98,50
TOTAL			R\$ 825,50

Figura 13 - Custo de Equipamentos

4.3.3 Custo total do Projeto

O custo total para o desenvolvimento do aplicativo é de R\$ 10985,50, considerando-se a somatória de gastos com mão de obra e equipamentos.

Valor Total do Projeto	
Mão de Obra	R\$ 10.160,00
Equipamento	R\$ 825,50
TOTAL	R\$ 10985,50

Figura 14 - Custo Total do Projeto

5. IMPLEMENTAÇÃO DO APLICATIVO

5.1 ORGANIZAÇÃO DO PROJETO

Para a implementação do aplicativo, foi utilizado a ferramenta de desenvolvimento Eclipse IDE. A aplicação é baseada na plataforma Google Android para dispositivos móveis, que fornece ferramentas para criação de aplicativos e utiliza a linguagem Java.

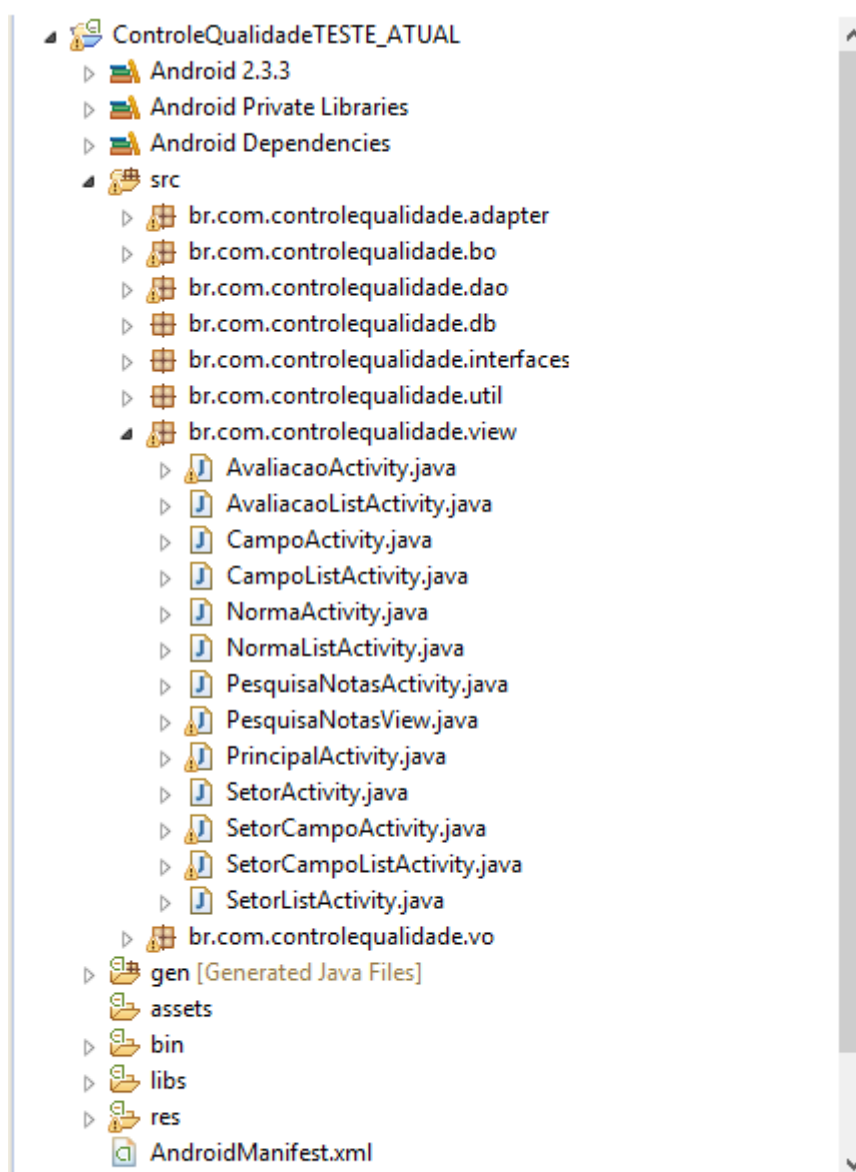


Figura 15 - Organização do Projeto

O projeto está dividido em 5 classes Java: `PrincipalActivity`, classe `SetorActivity`, classe `CampoActivity`, classe `NormaActivity` e `PesquisaNotasActivity`. As classes são organizadas no pacote `com.br.controldequalidade.view`.

PrincipalActivity – É a classe principal, que gerencia o acesso às telas do sistema, ou seja as telas de setores.

SetorActivity – É a classe que gerencia os setores , contém os métodos que mostram os setores já cadastrados e os métodos de validação .

CampoActivity – É a classe que gerencia os campos, contém os métodos que mostram os campos já cadastrados e os métodos de validação .

NormaActivity – É a classe que gerencia as Normas, contém os métodos que mostram as normas já cadastradas e os métodos de validação .

PesquisaNotasActivity – É a classe que faz a pesquisa das médias dos setores.

5.2 CLASSES

5.2.1 Classe `PrincipalActivity`

A classe `PrincipalActivity` apresenta o método `onCreate()` conforme figura a seguir. O método `onCreate()` é o método onde é feita a inicialização dos componentes essenciais de uma atividade. Dentro dele, o layout é invocado em `setContentView()`. Ainda dentro de `onCreate()` é criada as intents para inicializar cada atividade. Segue abaixo, o Método `onCreate()` da classe `PrincipalActivity`.

```
public class PrincipalActivity extends Activity {

    private Button btnSetor = null;
    private Button btnCampo = null;
    private Button btnNorma = null;
    private Button btnSetorCampos = null;
    private Button btnAvaliacao = null;
    private Button btnConsultarMedia = null;
    private Button btnSair = null;

    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        btnSetor = (Button) findViewById(R.id.btnSetor);
        btnCampo = (Button) findViewById(R.id.btnCampo);
        btnNorma = (Button) findViewById(R.id.btnNorma);
        btnSetorCampos = (Button) findViewById(R.id.btnSetorCampos);
    }
}
```

```
btnAvaliacao = (Button) findViewById(R.id.btnAvaliacao);
btnConsultarMedia = (Button) findViewById(R.id.btnConsultarMedia);
btnSair = (Button) findViewById(R.id.btnSair);

btnSetor.setOnClickListener(new View.OnClickListener() {

    @Override
    public void onClick(View arg0) {
        exibirSetor();
    }
});

btnCampo.setOnClickListener(new View.OnClickListener() {

    @Override
    public void onClick(View arg0) {
        exibirCampo();
    }
});

btnNorma.setOnClickListener(new View.OnClickListener() {

    @Override
    public void onClick(View arg0) {
        exibirNorma();
    }
});

btnSetorCampos.setOnClickListener(new View.OnClickListener() {

    @Override
    public void onClick(View arg0) {
        exibirSetorCampos();
    }
});

btnAvaliacao.setOnClickListener(new View.OnClickListener() {

    @Override
    public void onClick(View arg0) {
        exibirAvaliacao();
    }
});

btnConsultarMedia.setOnClickListener(new View.OnClickListener() {

    @Override
    public void onClick(View arg0) {
        exibirPesquisa();
    }
});

btnSair.setOnClickListener(new View.OnClickListener() {

    @Override
    public void onClick(View v) {
        sair();
    }
});
```

```

    }

    private void exibirSetor() {
        Intent iSetor = new Intent(this, SetorListActivity.class);
        startActivity(iSetor);
    }

    private void exibirCampo() {
        Intent iCampo = new Intent(this, CampoListActivity.class);
        startActivity(iCampo);
    }

    private void exibirNorma() {
        Intent iNorma = new Intent(this, NormaListActivity.class);
        startActivity(iNorma);
    }

    private void exibirSetorCampos() {
        Intent iCampo = new Intent(this, SetorCampoListActivity.class);
        startActivity(iCampo);
    }

    private void exibirAvaliacao() {
        Intent iAvaliacao = new Intent(this, AvaliacaoListActivity.class);
        startActivity(iAvaliacao);
    }

    private void exibirPesquisa() {
        Intent iPesquisa = new Intent(this, PesquisaNotasActivity.class);
        startActivity(iPesquisa);
    }

    private void sair() {
        Intent iSair = new Intent(Intent.ACTION_MAIN);
        finish();
    }
}

```

5.2.2 Classe AvaliacaoActivity

A classe AvaliacaoActivity é responsável pela avaliação do setores.

Possui os métodos onCreate, carregarAvaliacao, carregarSetores, tirarFoto, selecionar Data e pesquisarNormas .

No método onCreate() É a primeira função a ser executada em uma Activity, é a responsável por carregar os layouts XML e outras operações de inicialização. É executada apenas uma vez.

carregarAvaliacao é o método responsável por carregar a tela de Avaliação, onde são validados os campos e feito o calculo da média.

carregarSetores é o método responsável por carregar a tela de Setores, onde são validados os mesmos.

tirarFoto é o método responsável por tirar fotos.

selecionarData é o método responsável pelo pop up de data.

pesquisarNormas é o método responsável pela pesquisa das normas.

salvar é o método responsável por salvar a média setorial que será imprimida na tela de Avaliação 5s.

Segue abaixo, o Métodos da classe AvaliacaoActivity.

```
public class AvaliacaoActivity extends Activity {

    private EditText txtCodigo = null;
    private EditText txtNomeAvaliador = null;
    private EditText edData = null;
    private EditText txtObservacao = null;
    private Button btnSalvar = null;
    private Button btnTirarFoto = null;
    private ImageButton btnData = null;
    private ImageButton btnNormas = null;

    private Spinner spnSetor;
    private List<String> setores = new ArrayList<String>();
    private List<SetorVO> setorList = null;
    private String setor1;
    private SetorCampoBO setorCampoBO = new SetorCampoBO();

    private AvaliacaoVO avaliacao = null;
    private boolean isUpdate = false;
    private AvaliacaoBO avaliacaoBO = new AvaliacaoBO();

    private Random random = new Random();

    private LinearLayout ll;
    private LinearLayout ll2;
    Integer count = 0;
    private ArrayList<EditText> edits;
    private ArrayList<TextView> texts;
    private CamposAvaliadosBO camposAvaliadosBO = new CamposAvaliadosBO();

    private List<String> campos = new ArrayList<String>();
    private List<CampoVO> campoList;
    String campo1;

    Long codigoAvaliacao;
    Long codigoSetor;
    ListView lstCamposListaAval1;

    private CamposAvaliadosVO camposAvaliados = null;

    private static NumberFormat format = new DecimalFormat("00");
```

```

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.avalicao_view);

    txtCodigo = (EditText) findViewById(R.id.txtCodigo);
    txtNomeAvaliador = (EditText) findViewById(R.id.txtNomeAvaliador);
    edData = (EditText) findViewById(R.id.edData);
    txtObservacao = (EditText) findViewById(R.id.txtObservacao);
    btnSalvar = (Button) findViewById(R.id.btnSalvar);
    btnTirarFoto = (Button) findViewById(R.id.btnTirarFoto);
    spnSetor = (Spinner) findViewById(R.id.spnSetor);
    btnData = (ImageButton) findViewById(R.id.btnData);
    btnNormas = (ImageButton) findViewById(R.id.btnNormas);

    ll = (LinearLayout) findViewById(R.id.edits_ll);
    ll2 = (LinearLayout) findViewById(R.id.edits_ll2);
    edits = new ArrayList<EditText>();
    texts = new ArrayList<TextView>();

    carregarSetores();
    carregarSetorList();

    ArrayAdapter<String> arrayAdapter = new ArrayAdapter<String>(this,
        android.R.layout.simple_spinner_dropdown_item, setores);
    ArrayAdapter<String> spinnerArrayAdapter = arrayAdapter;

    spinnerArrayAdapter.setDropDownViewResource(android.R.layout.simple_spinner_item);
    spnSetor.setAdapter(spinnerArrayAdapter);

    btnSalvar.setOnClickListener(new View.OnClickListener() {

        @Override
        public void onClick(View arg0) {
            try {
                salvar();
            } catch (ParseException e) {
                // TODO Auto-generated catch block
                e.printStackTrace();
            }
        }
    });

    btnNormas.setOnClickListener(new View.OnClickListener() {

        @Override
        public void onClick(View v) {
            pesquisarNormas();
        }
    });

    spnSetor.setOnItemClickListener(new
    AdapterView.OnItemClickListener() {

        @Override
        public void onItemClick(AdapterView<?> parent, View v, int posicao, long id) {

```

```

setor1 = parent.getItemAtPosition(posicao).toString();

        l1.removeAllViews();
        l12.removeAllViews();

        if(campoList != null){
            campoList.clear();
        }
        carregarCampos();
        carregarCamposList();

        criarNovaEditText();

    }

    @Override
    public void onNothingSelected(AdapterView<?> parent) {

    }

});
btnTirarFoto.setOnClickListener(new View.OnClickListener() {

    @Override
    public void onClick(View v) {
        tirarFoto();
    }

});

btnData.setOnClickListener(new View.OnClickListener() {

    public void onClick(View arg0) {
        selecionarData();
    }

});

Bundle parametros = getIntent().getExtras();

if (parametros != null && parametros.containsKey("avaliacao")) {
    avaliacao = (AvaliacaoVO)
parametros.getSerializable("avaliacao");

    isUpdate = true;

    carregarAvaliacao();
} else {
    avaliacao = new AvaliacaoVO();
}

}

private void carregarAvaliacao() {

    txtCodigo.setText(Long.toString(avaliacao.getCodigoAvaliacao()));

    txtNomeAvaliador.setText(String.valueOf(avaliacao.getNomeAvaliador()));
    edData.setText(String.valueOf(avaliacao.getDataAvaliacao()));
    txtObservacao.setText(String.valueOf(avaliacao.getObservacao()));

}

```

```

private void salvar() throws ParseException {
    Boolean validou = true;

    if(validarCampos() == false){
        validou = false;
        setResult(Activity.RESULT_CANCELED);
    }

    if(validou == true){

        String setor[] = setor1.split(Pattern.quote("-"));
        Long codigoSetor = Long.parseLong(setor[0].replaceAll(" ", ""));
        Long código =Long.parseLong(txtCodigo.getText().toString()
!= null ? txtCodigo.getText().toString() : null);
        String nomeAvaliador = txtNomeAvaliador.getText().toString();
        Date dataAvaliacao =
DateUtil.parse(edData.getText().toString());
        String observacao = txtObservacao.getText().toString();

        avaliacao.setCodigoAvaliacao(codigo);
        avaliacao.setNomeAvaliador(nomeAvaliador);
        avaliacao.setDataAvaliacao(dataAvaliacao);
        avaliacao.setCodSetor(codigoSetor);
        avaliacao.setObservacao(observacao);

        camposAvaliados = new CamposAvaliadosVO();

        camposAvaliados.setCodigoAvaliacao(codigo);

        Integer tamSelecionados = edits.size();
        Integer x = 0;
        Double mediaAval = 0.0;

        while(x < tamSelecionados){
            if (x == tamSelecionados){
                break;
            }

            camposAvaliados.setCodigoCampo(campoList.get(x).getCodigoCampo());

            camposAvaliados.setDescCampo(campoList.get(x).getDescCampo());

            Double nota =
Double.parseDouble(edits.get(x).getText().toString());

            camposAvaliados.setNota(nota);

            mediaAval = mediaAval + nota;
            if(validarNota(edits.get(x)) == false){

                validou = false;
                setResult(Activity.RESULT_CANCELED);
                break;
            }
        }
    }
}

```

```

        try {
            if (!isUpdate) {
camposAvaliadosBO.inserir(this, camposAvaliados);
            } else {
camposAvaliadosBO.alterar(this, camposAvaliados);
            }
        }
        catch (Exception e) {
Toast.makeText(this,
e.getMessage(), Toast.LENGTH_LONG).show();
        }
        x++;
    }

    if(validou == true){
        Double valor = mediaAval/5;

        BigDecimal bd = new BigDecimal(valor);
        bd = bd.setScale(2, BigDecimal.ROUND_HALF_UP);
        Double media = bd.doubleValue();

        avaliacao.setMediaAvaliacao(media);

        try {
            if (!isUpdate) {
                avaliacaoBO.inserir(this, avaliacao);
            } else {
                avaliacaoBO.alterar(this, avaliacao);
            }
        } catch (Exception e) {
            Toast.makeText(this, e.getMessage(),
Toast.LENGTH_LONG).show();
        }

        setResult(Activity.RESULT_OK);

        finish();
    }
}

public void onValidationCancelled() { }

private void carregarSetores() {
    try {
        setorList = setorCampoBO.selecionarTodosSetores(this);
    } catch (Exception e) {
        Toast.makeText(this, e.getMessage(), Toast.LENGTH_LONG).show();
    }
}

```



```

}

    private void carregarSetorList(){

        Integer tam = setorList.size();
        Integer x = 0;

        while(x < tam) {

            if(x == tam){
                break;
            }
            setores.add(x, setorList.get(x).getCodigoSetor() + " - " +
setorList.get(x).getDescSetor());
            x++;
        }
    }

    private void carregarCampos() {

        try {

            String setor[] = setor1.split(Pattern.quote("-"));
            codigoSetor = Long.parseLong(setor[0].replaceAll(" ", ""));

            campoList =
camposAvaliadosBO.selecionarTodosCamposPorSetor(this, codigoSetor);
        } catch (Exception e) {
            Toast.makeText(this, e.getMessage(), Toast.LENGTH_LONG).show();
        }
    }

    private void carregarCamposList(){

        Integer tam = campoList.size();
        Integer x = 0;

        while(x < tam) {

            if(x == tam){
                break;
            }
            campos.add(x, campoList.get(x).getCodigoCampo() + " - " +
campoList.get(x).getDescCampo());
            x++;
        }
    }

    private void tirarFoto() {
        Integer randomId = random.nextInt(999999);

        String photoId = "avaliacao" + randomId;

        File file = new File(Environment.getExternalStorageDirectory()
            .getPath() + "/Avaliacao/" + photoId + ".jpg");

        avaliacao.setFotoAvaliacaoId(photoId);

        Uri imageUri = Uri.fromFile(file);
    }
}

```

```

        Intent takePictureIntent = new
Intent(MediaStore.ACTION_IMAGE_CAPTURE);
        takePictureIntent.putExtra(MediaStore.EXTRA_OUTPUT, imageUri);

        startActivityForResult(takePictureIntent, 123456);
    }

    private Bitmap carregarFoto(String fotoId) throws Exception {
        BitmapFactory.Options opcoes = new BitmapFactory.Options();
        opcoes.inPreferredConfig = Config.RGB_565;
        opcoes.inSampleSize = 2;

        String caminhoFoto = Environment.getExternalStorageDirectory()
            .getPath() + "/Avaliacao/" +
avaliacao.getFotoAvaliacaoId() + ".jpg";

        Bitmap bitmap = BitmapFactory.decodeFile(caminhoFoto, opcoes);

        ExifInterface ei = new ExifInterface(caminhoFoto);
        int orientation = ei.getAttributeInt(ExifInterface.TAG_ORIENTATION,
            ExifInterface.ORIENTATION_NORMAL);

        switch (orientation) {
            case ExifInterface.ORIENTATION_ROTATE_90:
                rotate(bitmap, 90);
                break;
            case ExifInterface.ORIENTATION_ROTATE_180:
                rotate(bitmap, 180);
                break;
        }

        return bitmap;
    }

    public static Bitmap rotate(Bitmap bitmap, int degree) {
        int w = bitmap.getWidth();
        int h = bitmap.getHeight();

        Matrix mtx = new Matrix();
        mtx.setRotate(degree);

        return Bitmap.createBitmap(bitmap, 0, 0, w, h, mtx, true);
    }

    private void selecionarData() {
        Calendar calendar = Calendar.getInstance();

        if (avaliacao.getDataAvaliacao() != null) {
            calendar.setTime(avaliacao.getDataAvaliacao());
        }

        DatePickerDialog dialog = new DatePickerDialog(
            this,
            new DatePickerDialog.OnDateSetListener() {

                @Override
                public void onDateSet(DatePicker dialog, int year,
                    int monthOfYear, int dayOfMonth) {
                    edData.setText(format.format(dayOfMonth) + "/"

```

```

        + format.format((monthOfYear + 1) + "/" + year);
    }
    }, calendar.get(Calendar.YEAR),
calendar.get(Calendar.MONTH),
    calendar.get(Calendar.DAY_OF_MONTH));

    dialog.show();
}
protected void criarNovaEditText() {

    edits = new ArrayList<EditText>();
    texts = new ArrayList<TextView>();

    Integer tam = campoList.size();
    Integer x = 0;
    while (x < tam){

        if(x == tam){
            break;
        }

        count++;
        LinearLayout.LayoutParams params2 = new LinearLayout.LayoutParams
            (LinearLayout.LayoutParams.MATCH_PARENT, LinearLayout.LayoutParams.WRAP_CONTENT);

        TextView text = new TextView(this);
        text.setText(campoList.get(x).getCodigoCampo() + "-" +
campoList.get(x).getDescCampo());

        text.setTextSize(18);
        text.setPadding(0, 20, 0, 0);
        texts.add(text);
        ll.addView(text, params2);

        LinearLayout.LayoutParams params = new LinearLayout.LayoutParams
            (LinearLayout.LayoutParams.MATCH_PARENT, LinearLayout.LayoutParams.WRAP_CONTENT);

        EditText edit = new EditText(this);
        edit.setWidth(180);
        edits.add(edit);
        ll2.addView(edit, params);
        x++;
    }

}

protected Boolean validarNota(EditText nota) throws ParseException{

    Double valorNota = Double.parseDouble(nota.getText().toString());

    if (valorNota < 0.00 || valorNota > 1.00 ) {

        Toast.makeText(AvaliacaoActivity.this, "Valor da Nota tem que
estar entre 0.00 e 1.00", Toast.LENGTH_LONG).show();

        nota.setError("Preencha com valor entre 0.00 e 1.00!");
        nota.requestFocus();
        return false;
    }

}

```

```

        else{
            return true;
        }
    }

    private void pesquisarNormas() {
        Bundle parametros = new Bundle();

        parametros.putBoolean("pesquisar", true);

        Intent iNormas = new Intent(this, NormaListActivity.class);

        iNormas.putExtras(parametros);

        startActivityForResult(iNormas, 12345);
    }

    protected Boolean validarCampos() throws ParseException{

        if ("".equals(txtCodigo.getText().toString())) {

            txtCodigo.setError("Código não pode ser nulo!");
            txtCodigo.requestFocus();
            return false;
        }
        else if ("".equals(txtNomeAvaliador.getText().toString())) {

            txtNomeAvaliador.setError("Nome não pode ser nulo!");
            txtNomeAvaliador.requestFocus();
            return false;
        }
        else if ("".equals(edData.getText().toString())) {

            edData.setError("Data não pode ser nula!");
            edData.requestFocus();
            return false;
        }
        else if ("".equals(txtObservacao.getText().toString())) {

            txtObservacao.setError("Observação não pode ser nula!");
            txtObservacao.requestFocus();
            return false;
        }
        else{
            return true;
        }
    }
}

```

5.3 Interface do Aplicativo

Menu Principal do sistema, aqui o usuário tem a opção de selecionar o setor ou campo que deseja cadastrar, além de poder fazer as consultas e cadastros de normas e Iniciar a avaliação 5s.

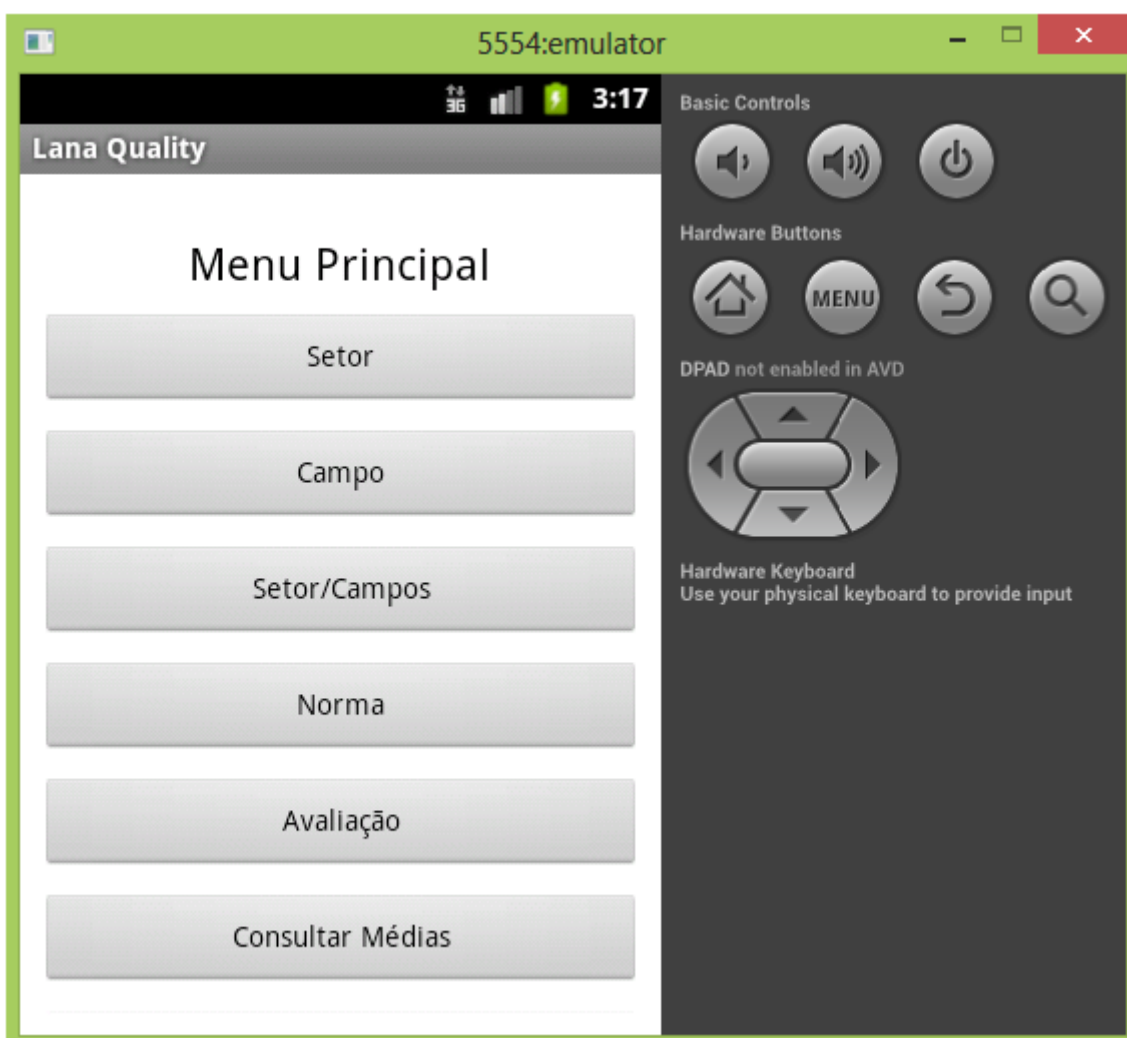


Figura 16 - Tela Menu Principal

Tela setor, onde é possível cadastrar novos setores.

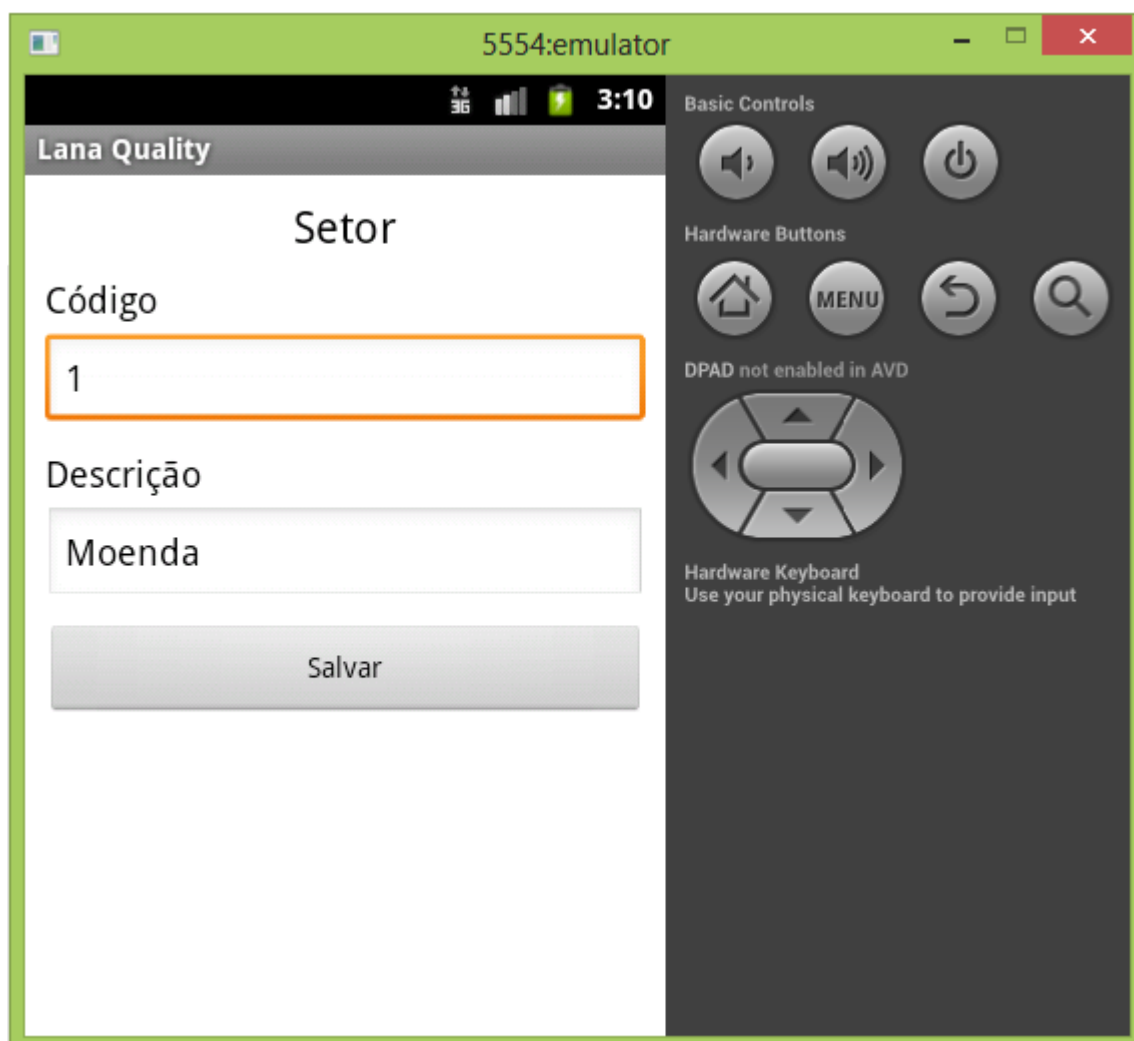


Figura 17 – Tela Setor

Tela de Avaliação 5s contém o código de avaliação, nome do avaliador e a data da avaliação. Nessa tela também é possível digitar as notas, utilizar a câmera e a informações contidas no sistema para ajudar no critério de avaliação.

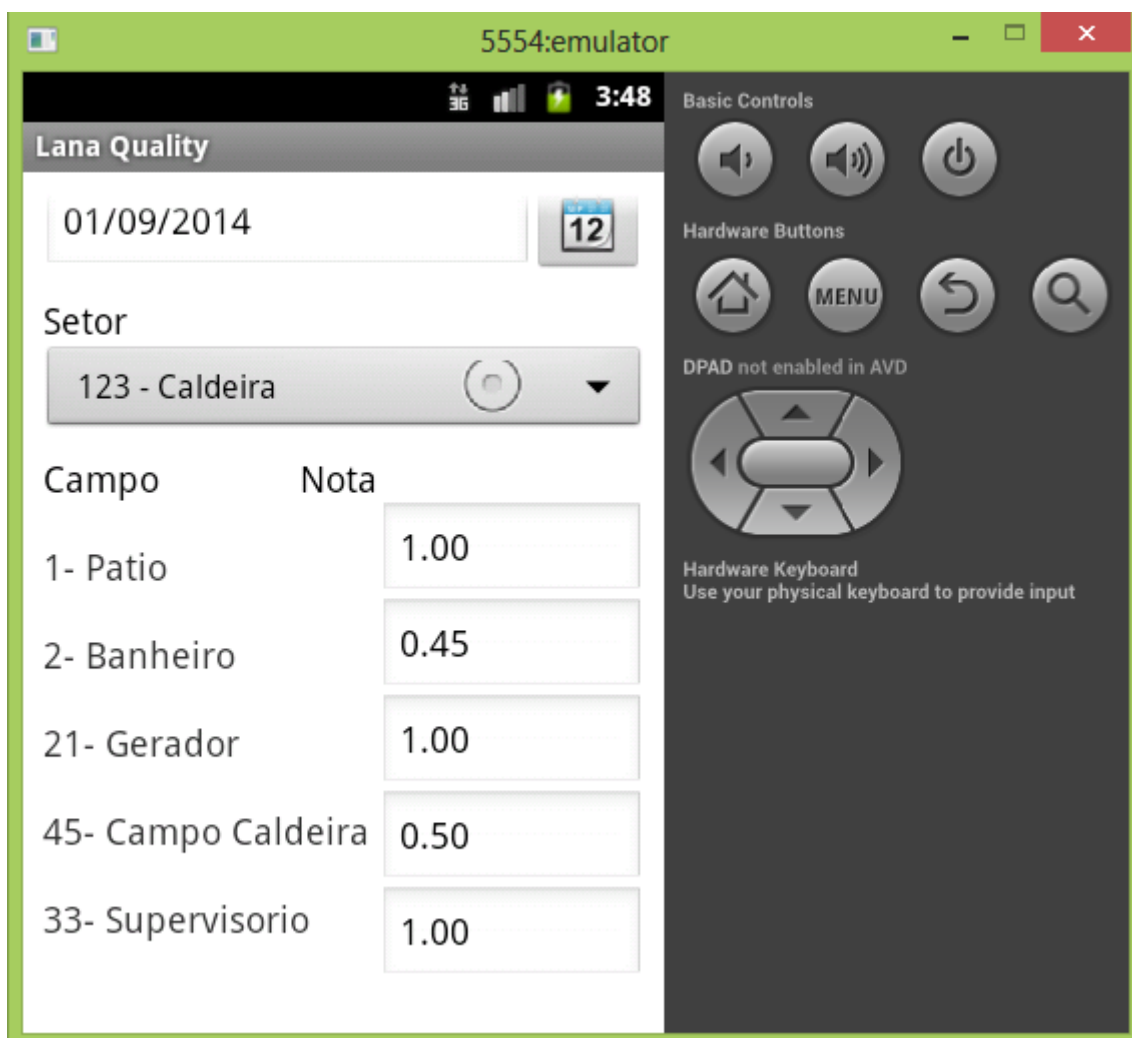


Figura 18 - Tela Avaliação

6. CONCLUSÃO

A qualidade é um segmento em constante ascensão no mercado. Empresas estão cada vez mais dispostas a investir em soluções que visem sua qualidade de produtos e serviços prestados. O uso da tecnologia a favor desse segmento aumenta a cada dia. O aplicativo construído nesse trabalho foi desenvolvido para dar auxílio ao auditor, melhorando a qualidade de avaliação 5s, proporcionando ao auditor maior desempenho e precisão na hora da avaliação. O auditor poderá, por exemplo, tirar fotos no mesmo dispositivo eliminando a câmera convencional e o uso de planilhas de papel que eram usadas anteriormente nas avaliações passadas.

REFERÊNCIAS

ABLESON, W. Frank; Collins, Chalie; Sen, Robi. *Unlocking Android. A Developer's Guide*. 1.ed. GreenWich: Manning, 2009.

CAMPOS, Vicente Falconi, TQC: Controle de Qualidade total (no estilo japonês) – Belo Horizonte: : Fundação Cristiano Ottoni, Escola de Engenharia da UMG, 1992 (Rio de Janeiro: Bloch Ed.)

DEITEL, Paul; DEITEL, Harvey. *JAVA – COMO PROGRAMAR*. São Paulo: Pearson, 2010

FARTO, Guilherme de Cleva. **Abordagem Orientada a Serviços para Implementação de um Aplicativo Google Android**. 2010, 83p. Monografia de Conclusão de Curso – Instituto Municipal de Ensino Superior de Assis, São Paulo, Assis, 2010.

GRADVOHL, André Leon S. *Introdução a Linguagem de Programação JAVA*, Universidade Estadual de Campinas UNICAMP, 2008.

HJELMAS, E. e LOW, B.K.: Face detection: A survey. *Computer Vision and Image Understanding*, 83(3):236–274, September 2001

www.significados.com.br/iso/-Significado de ISO - acessado dia 21/10/2013.

rticonsultoria.com.br/2010/03/09/conceitos-do-programa-5s/Conceitos do Programa 5s - acessado dia 21/10/2013.

LECHETA, Ricardo R.. *Google Android: Aprenda a criar aplicações para dispositivos móveis com o Android SDK*. São Paulo: Novatec, 2009.

LECHETA, Ricardo. *Google Android: Aprenda a criar aplicações para dispositivos móveis com o Android SDK*. 2. ed. São Paulo: Novatec, 2010.

LIMA, Fernando Cesar de. **Casos de Uso** - Introdução. Disponível em: http://200.230.71.24/moodle/file.php/120/Engenharia_01_CasosUsoIntroducao.pdf . Acesso em: 06 Março. 2014.

PEREIRA, Lúcio Camilo Oliva; SILVA, Michel Lourenço da Silva. **Android para Desenvolvedores**. 1. ed. Rio de Janeiro: Brasport Livros e Multimídia Ltda., 2009.

SILVA, João Martins. *5S: O Ambiente da qualidade*, Belo Horizonte: Fundação Cristiano Ottoni, 1994

www.significados.com.br/iso/- Significado de ISO - acessado dia 21/10/2013.

rticonsultoria.com.br/2010/03/09/conceitos-do-programa-5s/ -Conceitos do Programa 5s - acessado dia 21/10/2013

ANEXOS

CRONOGRAMA

Meses	Atividades
Novembro	Levantamento de Requisitos
Dezembro	Análise de Requisitos
Janeiro	Gerenciamento dos Requisitos
Fevereiro	Planejamento do Projeto
Março	Finalização da Análise Orientada á Objetos
Abril	Qualificação
Maior, Junho, Julho, Setembro	Implementação