



Fundação Educacional do Município de Assis
IMESA - Instituto Municipal de Ensino Superior de Assis

RAFAELA MARTINS DOS SANTOS

UMA APLICAÇÃO MOBILE PARA AVALIAÇÃO DE DESEMPENHO

Assis
2015

RAFAELA MARTINS DOS SANTOS

UMA APLICAÇÃO MOBILE PARA AVALIAÇÃO DE DESEMPENHO

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.

Orientador: Dr. Luiz Carlos Begosso

Área de Concentração: Ciências Exatas e da Terra

Assis
2015

FICHA CATALOGRÁFICA

SANTOS, Rafaela Martins

Uma aplicação mobile para avaliação de desempenho / Rafaela Martins dos Santos.

Fundação Educacional do Município de Assis – FEMA – ASSIS, 2014.

76p.

Orientador: Profº Dr.Luiz Carlos Begosso

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

1.Google Android 2.Web Service 3. Java 4. Desempenho

CDD: 001.61

Biblioteca da FEMA

UMA APLICAÇÃO MOBILE PARA AVALIAÇÃO DE DESEMPENHO

RAFAELA MARTINS DOS SANTOS

Trabalho de Conclusão de Curso apresentado ao
Instituto Municipal de Ensino Superior de Assis,
como requisito do Curso de Graduação, analisado
pela seguinte comissão examinadora:

Orientador: Dr. Luiz Carlos Begosso

Analisador (1): Me. Felipe Alexandre Cardoso Pazinato

Assis
2015

DEDICATÓRIA

Dedico este trabalho á Deus, família, namorado e todas as pessoas que acreditaram em meus sonhos e me apoiaram.

AGRADECIMENTOS

Primeiramente, agradeço a Deus, pela força, sabedoria e por todas as vezes que fez os momentos impossíveis se tornarem possíveis.

Aos meus pais, pelo apoio e dedicação.

Ao meu namorado Filipe, por todo o seu carinho e atenção.

Ao meu orientador e grande profissional Luiz Carlos Begosso, pela orientação, paciência, atenção e amizade.

“Não é merecedor do favo de mel aquele que evita a colmeia porque as abelhas têm
ferrões”

– William Shakespeare

RESUMO

A proposta deste trabalho foi a de ajudar, ainda que em pequena escala, o preparo do ambiente de uma organização, de forma que consiste no desenvolvimento de uma aplicação mobile para a avaliação de desempenho do colaborador, em que serão medidos seu comprometimento e produtividade na empresa.

Com a finalidade de apor essa aplicação para as organizações, além da aplicação mobile desenvolvida empregando a plataforma Google Android, é usada uma aplicação Web em que é feita uma integração por meio de um Webservice, a fim de ter os dados associados e agrupados para melhor análise da empresa.

Palavras-chave: Desempenho; Google Android; Java; Webservice.

ABSTRACT

The purpose of this study was to further help on a small scale preparation of an organization's environment so that consisti in developing a mobile application for the evaluation of employee performance, which will be measured in their commitment and productivity in the company.

In order to affix this application to organizations other than the mobile application developed using the Google Android platform, a WEB application is used in which the integration is made through a Web Service in order to be grouped and associated data to better company analysis.

Keywords: Performance; Google Android; Java; WebService.

LISTA DE FIGURAS

Figura 1 - Aplicações multicamadas em Java EE	19
Figura 2 - Serviço básico de <i>Web Service</i>	20
Figura 3 - Aplicativos do Android representados pelo Bugdroid.....	25
Figura 4 - Emulador do Android Integrado no Eclipse.....	26
Figura 5 - Camadas da plataforma Android	26
Figura 6 - Exemplo de diagramas UML.....	29
Figura 7 - Mapa Mental Avaliação de Desempenho - Mobile.....	30
Figura 8 - Mapa Mental Avaliação de Desempenho - Web.....	30
Figura 9 - Diagrama de caso de uso	32
Figura 10 - Caso de Uso Efetuar Login	33
Figura 11 - Caso de Uso Manter Funcionário.....	34
Figura 12 - Caso de Uso Atualizar Usuário	36
Figura 13 - Caso de Uso Manter Função	37
Figura 14 - Caso de Uso Avaliar Funcionário.....	39
Figura 15 - Caso de Uso Consultar Avaliação.....	41
Figura 16 - Caso de Uso Realizar Meta	42
Figura 17 - Caso de Uso Consultar Meta	44
Figura 18 - Caso de Uso Enviar Meta	45
Figura 19 - Caso de Uso Enviar Avaliação.....	46
Figura 20 - Caso de Uso Consultar Status.....	48
Figura 21 - Caso de Uso Atualizar Status	50
Figura 22 - Diagrama de Atividade Efetuar Login.....	51
Figura 23 - Diagrama de Atividade Avaliar Funcionário	52
Figura 24 - Diagrama de Atividade Realizar Meta.....	52
Figura 25 - Diagrama de Classe Geral.....	53
Figura 26 - Diagrama Entidade Relacionamento Geral.....	54
Figura 27 – Estrutura Analítica do Projeto - EAP	55
Figura 28 - Telas da Aplicação Mobile	57
Figura 29 - Tela Inicial – Metas	62
Figura 30 - Cadastro Metas.....	63
Figura 31 - Consultar Meta.....	64
Figura 32 - Lista Metas.....	65
Figura 33 - Tela Inicial - Avaliações	66

Figura 34 - Avaliação - Primeira Etapa.....	67
Figura 35 - Avaliação - Segunda Etapa.....	68
Figura 36 - Avaliação - Terceira Etapa.....	68
Figura 37 - Lista Avaliação.....	69
Figura 38 - Interface Web - Metas.....	70
Figura 39 - Interface Web - Avaliações	71

LISTA DE TABELAS

Tabela 1 - Exemplos de recursos e URIs	22
Tabela 2 - Requisições utilizando o protocolo HTTP	22
Tabela 3 - Lista de Eventos.....	31

SUMÁRIO

1. INTRODUÇÃO	15
1.1 OBJETIVO	16
1.2 MOTIVAÇÕES.....	17
1.3 JUSTIFICATIVA.....	17
1.4. ESTRUTURA DO TRABALHO	17
2. METODOLOGIA DE DESENVOLVIMENTO	18
2.1 PLATAFORMA JAVA ENTERPRISE EDITION (JEE).....	18
2.2 WEB SERVICE.....	20
2.2.1 REST - <i>Representational State Transfer</i>	21
2.2.2 URI- <i>Uniform Resource Identifier</i>	21
2.2.3 Protocolo HTTP	22
2.2.4 <i>Web Service</i> em JAVA	23
2.3 GOOGLE ANDROID	24
2.3.1 OHA- <i>Open Handset Alliance</i>	25
2.3.2 Android SDK	25
2.3.3 Arquitetura do Android	26
2.3.4 Recursos do Android	28
2.4 UML- UNIFIED MODELING LANGUAGE	28
3. ANÁLISE E ESPECIFICAÇÃO DO SISTEMA	30
3.1 MAPA MENTAL.....	30
3.2 LISTA DE EVENTOS	31
3.3 DIAGRAMA DE CASO DE USO	32
3.4 ESPECIFICAÇÃO DOS CASOS DE USO	32
3.5 DIAGRAMA DE ATIVIDADE	51
3.6 DIAGRAMA DE CLASSE	53
3.7 DIAGRAMA ENTIDADE RELACIONAMENTO	53
4. PLANEJAMENTO DO PROJETO.....	55
5. DESENVOLVIMENTO DO SISTEMA.....	56
5.1 PROJETO MOBILE COM ANDROID	56

5.2 APLICAÇÃO WEB.....	57
5.3 WEB SERVICE RESTFUL	58
6. CONCLUSÃO	59
APÊNDICE A – INTERFACES ANDROID - METAS.....	62
APÊNDICE B – INTERFACES ANDROID – AVALIAÇÕES	66
APÊNDICE C – INTERFACE WEB – METAS.....	70
APÊNDICE C – INTERFACE WEB – AVALIAÇÕES.....	71
ANEXO A – FORMULÁRIO DA AVALIAÇÃO DE DESEMPENHO	72

1. INTRODUÇÃO

A era da informação vem crescendo cada vez mais em ritmo acelerado, com esse fato, a tecnologia está sendo aprimorada a cada dia, contribuindo assim com diversas áreas, até mesmo no ambiente de trabalho ela vem fazendo grandes efeitos.

A Internet tem sido um ambiente de apoio à renovação das formas de ensino convencionais, onde o seu foco principal é facilidade de acesso rápido as informações e o ensino a distância.

Cada vez mais tem se pensado em formas de aprimorar o ensino nas mais diversas áreas, e com o surgimento do Google Android, o mercado tecnológico sofreu um grande avanço, deixando a sociedade fascinada por essa nova tecnologia, sendo hoje em dia um dos melhores sistemas operacionais móveis. Tais avanços referem-se a sua praticidade e facilidade de uso, onde as pessoas conseguem acesso á um sistema computacional em qualquer lugar e a qualquer momento, precisando apenas de um dispositivo móvel e estar conectado em uma rede.

Preparar e avaliar os colaboradores de uma empresa para o ambiente em que atuam, tem sido um grande desafio. Esse trabalho aborda questões comuns no ambiente de trabalho como: a dificuldade de avaliar o desempenho do colaborador. O bom desempenho entre os funcionários em uma empresa é sem sombra de dúvida, um dos fatores mais importante para o sucesso da mesma. A relação interna é um dos pilares mais altos, e se a empresa conseguir alcançá-lo, já está um longo caminho à frente, ou pelo menos um passo a frente dos concorrentes. Por essa razão, nesse trabalho, foi desenvolvida uma aplicação mobile que utiliza recursos de avaliação do desempenho dos colaboradores de uma organização, independentemente do segmento em que ela atua. Desenvolveu-se um software Web, que gera a integração dos dados armazenados nos *tablets* e *smartphones*, através de um Web Service.

O foco do aplicativo aqui proposto, é que os funcionários da empresa tenham a condição de avaliar seus colegas de trabalho.

A avaliação de desempenho serve para considerar o desempenho de sua equipe, e assim, oferecer resultados positivos, gerando benefícios e melhoras do desempenho

de cada colaborador da empresa. Encontram-se na literatura diversos sistemas correlatos a esta proposta, que já atuam nesse segmento. A seguir, apresentam-se alguns deles.

De acordo com a empresa Software Sob Medida, o Software de Avaliação de Desempenho desenvolvido pela mesma, é uma ferramenta 100% on-line, que pode gerenciar todo o processo de avaliação de desempenho, seja ele 180º ou 360º de maneira simples e eficaz (Software Sob Medida, 2014);

Segundo a empresa Metadados, especializada no fornecimento de soluções tecnológicas e serviços de apoio à gestão do capital humano, o módulo “Avaliação de Desempenho” presente em seu sistema, proporciona á empresa o reconhecimento das pessoas que realmente se destacam, permitindo acompanhar a gestão do capital humano dentro da organização (Metadados, 2014);

O Sistema RH1000[®] é voltado para o resultado, possibilita, além de estruturar e organizar os processos de RH, implantar avaliação de desempenho (Gestão por Competências), gerenciar treinamentos, controlar o plano de desenvolvimento individual e gerenciar seleção interna e externa (Sistema RH1000,2014);

O software de Avaliação de Desempenho da Menvie é um sistema online para a área de Recursos Humanos para avaliar seus colaboradores e a pesquisa de clima organizacional da empresa. (Menvie, 2014);

Enfim a empresa portuguesa Alvo, também acredita que é importante a avaliação de desempenho nas organizações. A empresa Alvo trabalha no mercado empresarial do software de gestão e tecnologias de informação, onde desenvolveu um Software para o processamento de salários, gestão de recrutamento, avaliação e desempenho. (Alvo, 2014).

1.1 OBJETIVO

Esse trabalho tem como objetivo principal implementar uma aplicação mobile e uma aplicação Web que sirva como um recurso de avaliação de desempenho dos colaboradores em uma organização.

1.2 MOTIVAÇÕES

São várias as motivações para desenvolver esse trabalho abordando o tema de avaliação de desempenho. Entre elas, destacam-se o mundo Web e o quão a linguagem Java vem sendo utilizada, não se esquecendo da praticidade que o Google Android proporciona e o quanto o dispositivo móvel vem fazendo parte do dia-a-dia das pessoas.

Outra motivação que foi decisiva para a preparação do trabalho, são as grandes e consideráveis chances que a aplicação dessas tecnologias tendem a proporcionar, como um ganho de qualidade, agilidade e facilidade notável aos serviços prestados por organizações.

1.3 JUSTIFICATIVA

Atrair competência, habilidade e inovações para a organização significam não somente pensar nas operações da mesma no presente, mas principalmente no seu futuro, e isso está relacionado com o desempenho dos colaboradores das empresas, portanto, o aplicativo desenvolvido visa conceber, estrategicamente, a evolução desses colaboradores, de modo que conforme eles sejam avaliados, eles possam obter uma motivação para que haja melhorias tanto da sua parte como da organização como um todo.

1.4. ESTRUTURA DO TRABALHO

O trabalho está estruturado em seis capítulos, iniciando com uma breve introdução a respeito do que será abordado. No segundo capítulo o assunto a ser relatado é a metodologia do desenvolvimento, explicando sobre o conceito da plataforma Java EE, o conceito de Web Service e a sua grande utilidade com relação a integração de dados em plataformas distintas, além de descrever sobre o foco desse trabalho, plataforma Android e o conceito da linguagem UML. No terceiro capítulo é mencionada a análise e especificação do sistema, em que é composto pelo mapa mental, diagramas de caso de uso, diagramas de classe, diagramas de atividade e diagrama de entidade relacionamento. O quarto capítulo apresenta o planejamento do projeto, com a estrutura analítica do projeto e o cronograma. O quinto capítulo é composto pelo desenvolvimento do Sistema, já o sexto capítulo relata a conclusão do trabalho e os trabalhos futuros.

2. METODOLOGIA DE DESENVOLVIMENTO

2.1 PLATAFORMA JAVA ENTERPRISE EDITION (JEE)

A Java EE (Java Platform, Enterprise Edition) é uma plataforma padrão para desenvolver aplicações na linguagem Java. Essa plataforma abrange um conjunto de repletas bibliotecas e funcionalidades bem detalhadas que funcionam como uma receita para implementar software Java distribuído, baseado em componentes modulares que executam em servidores de aplicações e que suportam alguns requisitos, entre eles podemos citar : escalabilidade, segurança e integridade . Além dos requisitos citados, há outros referentes a aplicações corporativas ou de grande porte, que também são de grande relevância (FARIA, 2013).

A Java EE (Java Enterprise Edition) é um grande conjunto de especificações, em que essas especificações quando implementadas, servem como auxílio no desenvolvimento da aplicação, pois não precisa se preocupar com grande parte do código de infraestrutura, o que exigiria muito trabalho (CAELUM, 2015).

Entre esse conjunto de especificações da Java EE, as mais conhecidas são (FARIA, 2013):

- Servlets: são componentes Java executados no servidor para gerar conteúdo dinâmico para a web, como HTML e XML;
- JSP- *JavaServer Pages*: é uma especialização de Servlets que permite que aplicações web desenvolvidas em Java sejam mais fáceis de manter;
- JSF- *JavaServer Faces*: é um framework web baseado em Java que tem como objetivo tornar o desenvolvimento de interfaces de sistemas para a web de uma forma mais simples, através de um modelo de componentes reutilizáveis;
- JPA (Java Persistence API): é uma API padrão do Java para persistência de dados, que usa um conceito de mapeamento objeto-relacional. Essa tecnologia é muito útil para o desenvolvimento de sistemas que necessitam de integração com banco de dados, pois trazem uma alta produtividade. Para finalizar, essa API possibilita que você desenvolva aplicações usando banco de dados sem precisar escrever uma linha de SQL;

- EJB (*Enterprise Java Beans*): este componente roda no Java EE e é um dos *containers* mais conhecidos da arquitetura e possuem como principais objetivos, fornecer facilidade e produtividade no desenvolvimento de componentes distribuídos, transacionados, seguros e portáteis.

Esse conjunto que a plataforma *Java Enterprise Edition* oferece, tem como objetivo facilitar a vida do programador, cabendo a ele se concentrar somente nas regras de negócio da empresa. Essas tecnologias são divididas em um modelo multicamadas em que devem ocorrer interações constantes e diretas entre diversas partes, como ilustra a figura 1 (FILHO,2015).

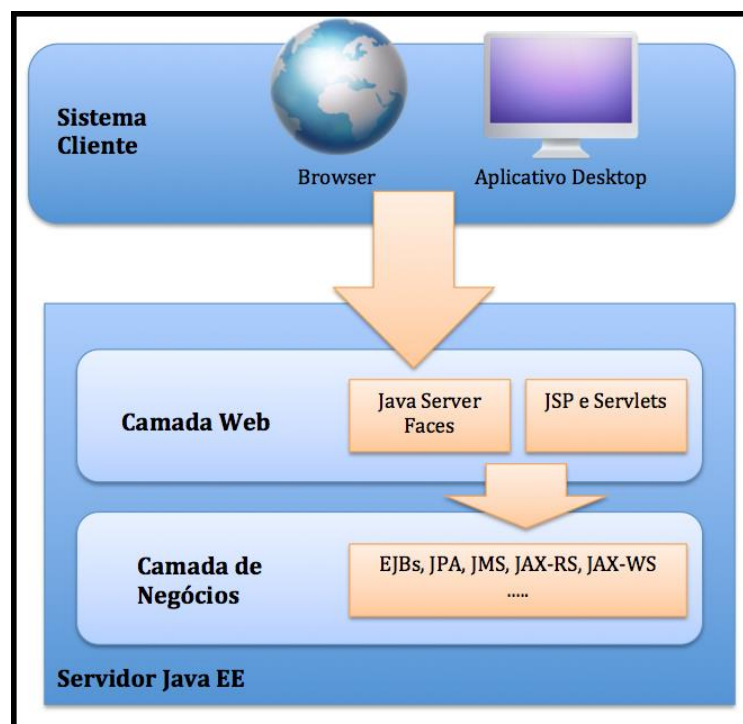


Figura 1 - Aplicações multicamadas em Java EE (In: FILHO,2015)

Uma das camadas deste modelo abrange os componentes Web de uma aplicação Java EE, que podem ser desenvolvidos, por exemplo, com o uso de Servlets, páginas JSP ou JSF que apresentam a parte visual (interface) ao usuário, já as regras de negócio são definidas em outra camada da aplicação JAVA EE, por meio do uso de APIs e frameworks direcionados ao desenvolvimento dos componentes de negócio, implementados em EJBs. A Java EE engloba também diversas outras APIs, como por exemplo: JMS (*Java Message Service*) para troca de mensagens;

JTA (*Java Transaction API*) para controle de transações; e JAX-RS/JAX-WS, para criação de web services (FILHO, 2015).

2.2 WEB SERVICE

Os *Web Services* permitem que novas aplicações possam interagir com eficiência com aquelas que já existem e que sistemas desenvolvidos em plataformas distintas sejam compatíveis (MEDEIROS, 2015).

Os *Web Service* trabalham em um nível de abstração semelhante à Internet, sendo assim são apropriados para superar qualquer plataforma de hardware ou linguagem de programação. A linguagem XML, é o formato padrão utilizado pelos *Web Services*, por ser flexível o suficiente para acomodar qualquer tipo e estrutura de dados e até mesmo criar novos, se necessário. Os *Web Services* são adaptados para a web, utilizando o protocolo de rede padrão HTTP, e o formato XML para fazer a troca de dados através de documentos, em que se enviam requisições através da rede e recebem como resposta um documento XML, porém o documento não é obrigatório, isso deverá ser definido pelo desenvolvedor. Os padrões de *Web Services* definem suas especificações, ou seja, o formato de suas mensagens e como elas serão publicadas e descobertas na rede (NEWCOMER, 2002).

A figura 2 esboça a visão de Cerami (2002) sobre o conceito de *Web Services*, em que ilustra a transferência de dados em plataformas diferentes utilizando o formato XML.

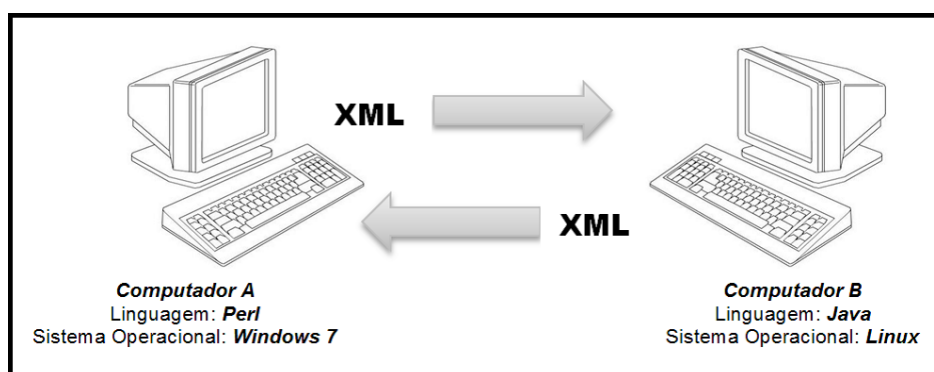


Figura 2 - Serviço básico de *Web Service* (In: CERAMI, 2002, p. 4)

2.2.1 REST - *Representational State Transfer*

A arquitetura REST (*Representational State Transfer*) ou (Transferência de Estado Representativo), é uma maneira de desenvolvimento de *Web Services* que teve origem na tese de doutorado de Roy Fielding. Este, por sua vez, é co-autor de um dos protocolos mais utilizados no mundo, o HTTP (*HyperText Transfer Protocol*). Assim, fica evidente que a arquitetura REST é concebida para uso do protocolo HTTP, além disso, é guiada dentre outros preceitos pelo que seriam as boas práticas de uso deste protocolo (SAUDATE, 2013):

- Uso adequado dos métodos HTTP;
- Uso adequado de URL's;
- Uso de códigos de status padronizados para representação de sucessos ou falhas;
- Uso adequado de cabeçalhos HTTP;
- Interligações entre vários recursos diferentes.

O modelo REST chegou no mercado com o propósito de expor serviços de uma forma mais simples por meio do protocolo HTTP. As aplicações móveis são candidatas a usar serviços REST, justamente pela simplicidade nos dados de retorno (SILVA, 2008).

A arquitetura REST além de ser muito interessante é fácil de utilizar. A arquitetura de comunicação REST é baseada em três princípios básicos (SAMPAIO, 2011):

- Recurso: um recurso pode ser considerado um produto ou um cliente etc;
- Localização: através de uma URI (*Uniform Resource Identifier*) se define a localização de um recurso. Podendo até mesmo incluir sua identidade.
- Verbo: são métodos HTTP, como: GET, PUT, DELETE OU POST.

2.2.2 URI- *Uniform Resource Identifier*

O recurso é o papel fundamental da arquitetura REST. Os recursos na Web são identificados através da URI que como o nome mesmo já diz é um identificador único para um recurso, composto pelo nome e um endereço estruturado indicando onde encontrar determinado recurso. Alguns exemplos de recursos e URIs estão listados na tabela 1 (GONÇALVES, 2010).

Existe uma relação entre recurso e URI, em que essa relação é de um para muitos, isso significa que um recurso pode ser referenciado por muitas URIs, mas uma URI só pode ser associada com um recurso (POLONIA, 2011).

Resource	URI
The catalog of Apress books	http://www.apress.com/book/catalog
The cover of the Java EE 6 book	http://www.apress.com/book/catalog/beginning-javaee6.jpg
Information about jobs at Apress	http://www.apress.com/info/jobs
The weather in Paris for 2008	http://www.weather.com/weather/2008?location=Paris,France
Interesting photos on Flickr for January 1st, 2009	http://www.flickr.com/explore/interesting/2009/01/01
Interesting photos on Flickr for the last 24 hours	http://www.flickr.com/explore/interesting/24hours
The list of adventure movies	http://www.movies.com/categories/adventure

Tabela 1 - Exemplos de recursos e URIs (GONÇALVES, 2010)

2.2.3 Protocolo HTTP

Em 1996, Tim Berners-Lee, Roy Fielding e Henrik Frystyk Nielsen fizeram trabalhos em conjunto, os quais levaram a publicação de uma RFC (*Request for Comments*) em que descreviam o protocolo HTTP (*HyperText Transfer Protocol* - Protocolo de Transferência de Hipertexto) (SAUDATE, 2013).

HTTP é um protocolo para distribuição, além disso, é colaborativo e um sistema de informação hipermídia. O protocolo HTTP levou à criação da *World Wide Web*, juntamente com URIs, HTML, e os primeiros navegadores (GONÇALVES, 2010).

A tabela 2 representa, de forma simples, os formatos de requisições em que o cliente envia uma solicitação para um servidor á espera de uma resposta, além de exemplificar como o método GET foi utilizado para solicitar o conteúdo da URL /cervejaria/clientes (GONÇALVES, 2010) e (SAUDATE, 2013).

DESCRIÇÃO	REQUISIÇÃO
Este protocolo segue o seguinte formato de requisições:	<método> <URL> HTTP/<versão> <Cabeçalhos - Sempre vários, um em cada linha> <corpo da requisição>
Algo parecido enviado para o servidor:	GET /cervejaria/clientes HTTP/1.1 Host: localhost:8080 Accept: text/html
Portanto, o que foi passado nesta requisição foi:	GET /cervejaria/clientes HTTP/1.1

Tabela 2 - Requisições utilizando o protocolo HTTP (SAUDATE, 2013)

2.2.4 Web Service em JAVA

Na plataforma *Java Enterprise Edition* (JEE), os serviços Web podem ser implementados de inúmeras maneiras. Nessa seção, será descrito especificamente a funcionalidade da API (JAX-RS), API para criação de *Web Services RESTful*.

JAX-RS é uma API da plataforma Java EE, projetada para desenvolver aplicações de forma mais fácil que usam a arquitetura REST.

A API *Java API for RESTful Web Services* (JAX-RS) usa anotações linguagem de programação Java para simplificar o desenvolvimento de *Web Services RESTful*. As classes de linguagem de programação Java são usadas por meio de anotações JAX-RS para definir os recursos e as ações que podem ser realizadas sobre os mesmos. Anotações JAX-RS são anotações de tempo de execução, assim gerando as classes auxiliares e componentes para o recurso. Uma aplicação Java EE contendo a API JAX-RS, conseqüentemente terá classes de recursos configurados, as classes auxiliares e componentes gerados, bem como o recurso expostos aos clientes por meio de um servidor Java EE (JENDROCK et al., 2013).

A API JAX-RS faz a conversão entre tipos do Java e os *MIME* do HTTP automaticamente. Um exemplo que pode ser citado é que se você marcar um método de uma classe com a annotation "`@Produces(MediaType.TEXT_PLAIN)`", o JAX-RS converteria o tipo Java para o MIME "text/plain", que representa texto sem formatação, e retornaria o conteúdo dentro do HTTP como resposta para o usuário. Como toda especificação, para utilizar os recursos disponíveis na API JAX-RS, é necessário o uso de alguma implementação da mesma. As *annotations* ou anotações são usadas para identificar recursos que deseja-se disponibilizar e que possam atender requisições HTTP. As principais *annotations* usadas para identificar recursos que se deseja disponibilizar e que possam atender requisições HTTP são (PENCHIKALA, 2015) e (MACEDO, 2015):

- `@Path`: anotação responsável por especificar um caminho relativo para um determinado recurso. Também identifica em qual URI um recurso será disponibilizado para receber requisições;

- @GET: anotação responsável por especificar que um método (recurso) processará apenas requisições do tipo GET. Define o acesso a leitura para determinado recurso;
- @PUT: responsável por atualizar um recurso;
- @POST: adiciona um recurso;
- @DELETE: anotação responsável pela remoção dos recursos;
- @Produces: essa anotação especifica os tipos *MIME* que um método lançará como resposta para o cliente;
- @Consumes: Especifica os tipos *MIME* que um método pode receber do usuário.

2.3 GOOGLE ANDROID

O Android é uma plataforma desenvolvida pela Google, especialmente para dispositivos móveis como aparelhos celulares. É uma plataforma que contém diversos aplicativos principais, como Navegador de Internet, Contatos e o Telefone propriamente dito, além disso, existe o Android SDK que é um conjunto de ferramentas e APIs para o desenvolvimento de aplicativos para a plataforma, utilizando a linguagem Java. O sistema operacional Android é baseado no Linux e teve seu desenvolvimento iniciado em 2003 pela empresa Android Inc. Em 2005, a empresa foi adquirida pelo Google que hoje lidera o desenvolvimento do Android. Em 2007, criou-se uma associação de empresas chamada de *Open Handset Alliance*, que como o nome mesmo já diz é uma aliança/associação de empresas líderes em tecnologia móvel, cuja missão é desenvolver uma plataforma para dispositivos móveis que seja completa, aberta e gratuita. Além de vários benefícios que a plataforma Android proporciona, ela ainda possui código aberto, ou seja, você tem acesso aos códigos-fonte e também pode contribuir com o projeto fazendo alterações e melhorias (MONTEIRO, 2012).

A plataforma Android está presente em dispositivos móveis e disponível para consumidores dos mais diferentes perfis. A logo do Android é representada por um robô verde chamado Bugdroid criado pela ilustradora Irina Blok. Como a própria ilustradora diz em sua página: “A ideia era criar uma logo *open source* (muito

parecido com a plataforma Android)” (HAMANN, 2013). A figura 3 ilustra o Bugdroid e alguns dos seus aplicativos.



Figura 3 - Aplicativos do Android representados pelo Bugdroid, (In: THE MOVIE THEME SONG, 2015)

2.3.1 OHA- *Open Handset Alliance*

A OHA é formada por 84 empresas de tecnologia, que se uniram para acelerar a inovação em consumidores móveis, que juntos desenvolvem Android, a primeira plataforma móvel, completa e livre (*Open Handset Alliance, 2015*).

Como já dito anteriormente a *Open Handset Alliance* é um grupo formado por grandes empresas do mercado de telefonia de celulares liderados pelo Google, algumas delas são: Samsung, ASUS, Dell, Garmin, LG, Motorola e muitas outras. Essa aliança tem por objetivo principal a criação de uma plataforma moderna e flexível para o desenvolvimento de aplicações corporativas, além de definir uma plataforma única e aberta a fim de deixar os consumidores satisfeitos (LECHETA, 2013).

2.3.2 Android SDK

O Android SDK é um software utilizado para desenvolver aplicações no Android, que possui um conjunto de ferramentas, uma API completa para a linguagem Java, um emulador para simular o celular e por fim todas as classes necessárias para o desenvolvimento das aplicações. Embora o SDK contenha um emulador que pode ser executado como um aplicativo comum, é necessário que instale um plug-in para

o Eclipse para assim integrar o ambiente de desenvolvimento Java com o emulador, como ilustra a figura 4 (LECHETA, 2013).

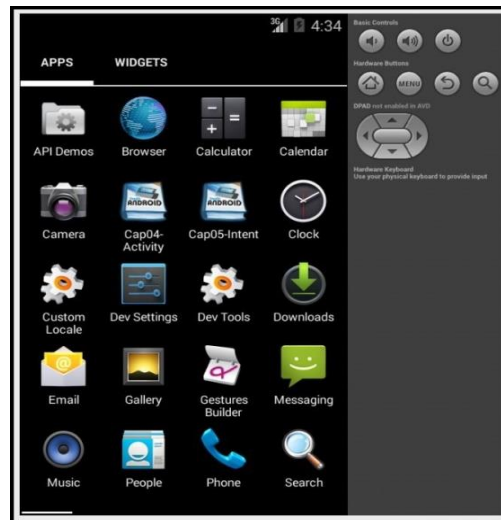


Figura 4 - Emulador do Android Integrado no Eclipse

Com esse plug-in é possível executar o emulador no eclipse. Isso é possível porque ocorre a instalação da aplicação automaticamente e a integração do debug do Eclipse, conseqüentemente ocorrendo a depuração do código-fonte como uma aplicação qualquer da linguagem Java (LECHETA, 2013).

2.3.3 Arquitetura do Android

O Android possui uma arquitetura estruturada e que atualmente é dividido em seis camadas. A figura 5 ilustra as divisões das camadas, sendo elas: aplicativos, framework, bibliotecas, ambiente de execução, Kernel Linux e abstração de hardware- HAL (FARIA, 2008) e (ANDROID, 2015).

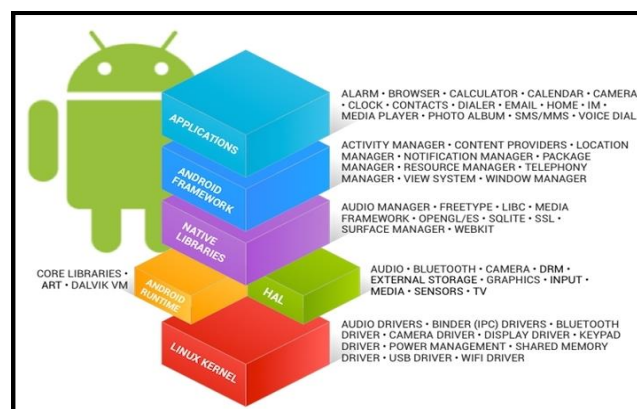


Figura 5 - Camadas da plataforma Android (In: ANDROID, 2015)

Abaixo, se encontram descrições com detalhes de cada camada composta pela arquitetura Android.

- Kernel Linux : A plataforma Google Android é construída sobre um Kernel Linux e sobre uma máquina virtual, ambas as tecnologias são essenciais para o desenvolvimento das aplicações (KING, et al., 2012).

O sistema operacional do Android é responsável por gerenciar a memória, os processos, *threads* e a segurança das aplicações desenvolvidas, não deixando de lado as redes e *drivers* (LECHETA, 2013).

Composta pelo *kernel* 2.6 a camada Kernel Linux é responsável pelos serviços, segurança, gerenciamento de memória e processos, rede e drivers (FARIA, 2008).

- Bibliotecas: A camada bibliotecas fornece um conjunto de bibliotecas C/C++ utilizadas pelo sistema. As bibliotecas C padrão (*Libc*) e também as relacionadas à multimídia, visualização de camadas 2D e 3D, funções para navegadores *Web*, funções para gráficos, funções de aceleração de hardware, renderização 3D, fontes bitmap e vetorizadas e funções de acesso ao banco de dados SQLite, fazem parte deste conjunto (FARIA, 2008).

- Android Runtime: Ambiente de execução. É uma pequena instância da máquina virtual *Dalvik* criada toda vez que uma aplicação em Android é executada (FARIA, 2008).

As aplicações Android implementadas são executadas da máquina virtual Dalvik, em que é baseada em registradores e otimizada para consumir pouca memória (MONTEIRO, 2012).

A Dalvik é otimizada para utilizar o mínimo de consumo de memória, bateria e CPU. Considera-se que a Dalvik seja uma máquina virtual com melhor desempenho, maior integração com a nova geração de hardware e projetada para executar várias máquinas virtuais paralelamente (FARIA, 2008).

- Framework: Na camada Framework se encontram todas as APIs e os recursos empregados nos aplicativos, com classes visuais como botões e *views*, provedor de conteúdo, ou seja, troca de recursos entre aplicativos e enfim, gerenciadores de recursos, de notificação e de pacotes (FARIA, 2008).

- Aplicativos: A camada superior, ou seja, que se encontra acima de todas as outras camadas é a de aplicativos, em que estão presentes todos os aplicativos do Android, como cliente de email, navegador web, contatos e outros. Isso denota que, para desenvolver programas para a plataforma Android, vamos criar os aplicativos em Java na máquina virtual Dalvik (FARIA, 2008).
- HAL- Abstração de Hardware: A camada HAL serve como uma interface padrão que permite ao sistema Android chamar para a camada de driver do dispositivo as implementações de nível mais baixo de seus drivers e hardware (ANDROID, 2015).

2.3.4 Recursos do Android

A plataforma Android oferece vários recursos a fim de facilitar o desenvolvimento da aplicação móvel, dentre eles podemos citar (PEREIRA, SILVA, 2009):

Framework da aplicação: permite a incorporação, a reutilização e a substituição de recursos;

Máquina Virtual Dalvik: otimizada para aplicações móveis;

SQLite: gerenciador de banco de dados, utilizado para armazenamento de dados;

Suporte Multimídia: suporte uma variedade de formatos de sons, imagens e vídeos;

Câmera, GPS, Bússola e Acelerômetro: opção de integração com a câmera, localização via GPS, Bússola e Acelerômetro;

Poderoso Ambiente de Desenvolvimento: possui um emulador que simula um celular, ferramenta de depuração, analisador de memória e performance e um plugin para a IDE Eclipse (ADT) – Android Development Tools.

2.4 UML- UNIFIED MODELING LANGUAGE

A UML - *Unified Modeling Language* é uma linguagem para modelagem de software orientado a objetos. Os esforços para a criação da UML se iniciaram em 1994, em fusão de três métodos Booch, OMT e OOSE. A UML permite a criação de modelos de software, oferecendo praticidade para quem utiliza a ferramenta como modelagem para levantamento de requisitos. A linguagem de modelagem unificada é composta por vários diagramas que representam o modelo do software a ser

desenvolvido. A figura 6 ilustra essa estrutura e, em seguida, apresentam-se as características de cada diagrama (LOBO, 2009) e (BOOCH, etc al.,2005).

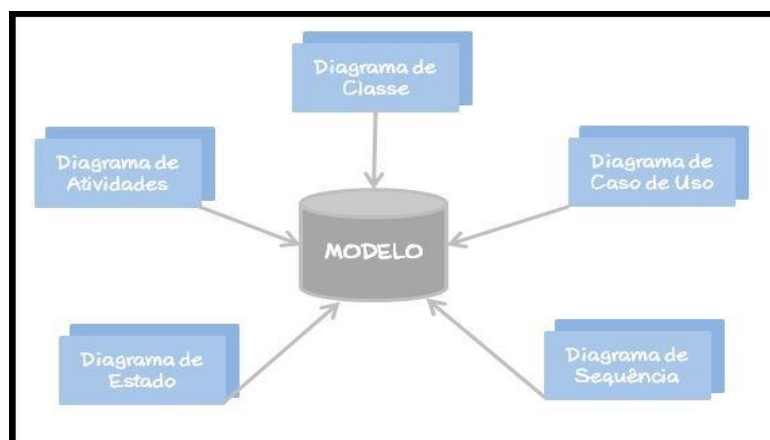


Figura 6 - Exemplo de diagramas UML

- Diagrama de Atividades: esse diagrama tem como objetivo principal modelar o comportamento dinâmico do sistema, em que descreve os passos percorridos para a conclusão de uma atividade (MARTINS, 2010) e (MARTINEZ, 2015).
- Diagrama de Classe: diagrama representado por classes e o relacionamento entre elas. As classes são compostas por métodos e atributos (BOOCH, et al., 2005) e (MARTINEZ, 2015).
- Diagrama de Caso de Uso: representa os requisitos do sistema e suas funcionalidades. A forma concebida é a seguinte: o usuário do sistema é representado pelo “Ator” trocando mensagens que simulam as funções que o sistema oferece (LOBO, 2009).
- Diagrama de Estado: tem como objetivo acompanhar as mudanças de um objeto dentro de um processo (MARTINEZ, 2015).
- Diagrama de Sequência: diagrama de interação. Grande tendência em facilitar a visibilidade do fluxo de sequências, ou seja, mostrar com clareza a ordem temporal das mensagens (LARMAN, 2005).

3. ANÁLISE E ESPECIFICAÇÃO DO SISTEMA

3.1 MAPA MENTAL

O Mapa mental é considerado um padrão mundial para gerenciamento de ideias em que é composto por uma ideia principal e os outros ramos do mapa significa uma decomposição da ideia central em ideias relacionadas.

As figuras 7 e 8 ilustram os Mapas Mentais do Software Avaliação de Desempenho Mobile e Web.

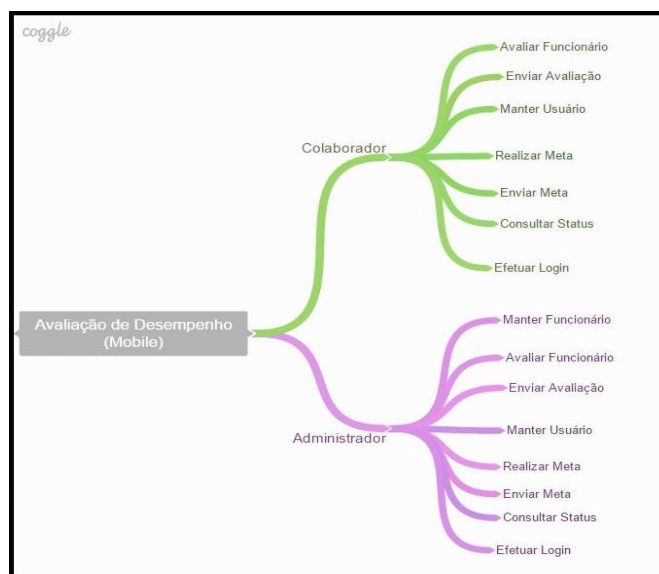


Figura 7 - Mapa Mental Avaliação de Desempenho - Mobile

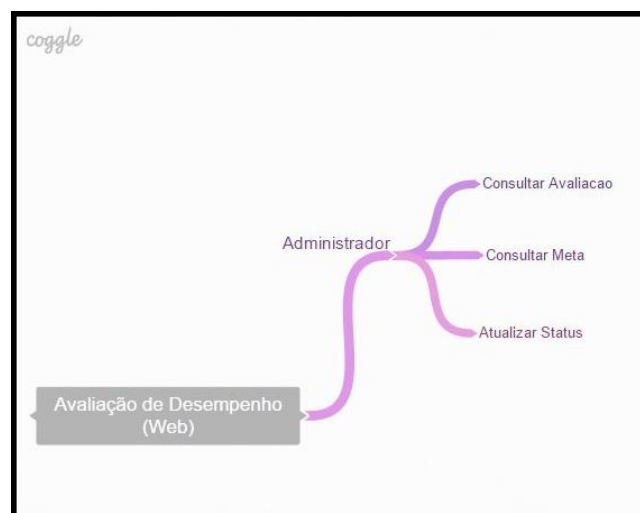


Figura 8 - Mapa Mental Avaliação de Desempenho - Web

3.2 LISTA DE EVENTOS

A tabela 3 apresenta a lista de eventos, ou seja, a lista de funcionalidades do sistema, e os atores que tem autonomia para realizar determinado evento.

Lista de Eventos			
Nº	Evento	Descrição	Ator
1	Efetuar Login	Efetuar login na aplicação	Administrador/Colaborador
2	Manter Funcionário	Efetuar inclusão/ alteração no cadastrado de Funcionário	Administrador
3	Atualizar Usuário	Efetuar alteração no cadastrado de Usuário	Administrador/Colaborador
4	Manter Função	Efetuar inclusão/ alteração no cadastrado de Função	Administrador
5	Avaliar Funcionário	Avaliar o desempenho dos funcionários (colaborador/administrador)	Colaborador/Administrador
6	Consultar Avaliação	Efetuar Consulta da Avaliação do Funcionário	Administrador
7	Consultar Meta	Efetuar Consulta da Meta do Funcionário	Administrador
8	Realizar Meta	Efetuar o cadastro de Meta	Colaborador/Administrador
9	Enviar Meta	Enviar a Meta para a aplicação Web	Colaborador/Administrador
10	Enviar Avaliação	Enviar a Avaliação para a aplicação Web	Colaborador/Administrador
11	Consultar Status	Consultar status da Meta	Colaborador/Administrador
12	Atualizar Status	Atualizar o Status da Meta	Administrador

Tabela 3 - Lista de Eventos

3.3 DIAGRAMA DE CASO DE USO

O diagrama de caso de uso, como mencionado anteriormente, representa claramente as funcionalidades do sistema. A figura 9 ilustra o diagrama de caso de uso do software Avaliação de Desempenho. Como se pode observar, o Administrador herda todas as funções que um Colaborador contém, porém somente o Administrador poderá manter o Colaborador e o Administrador, ou seja, somente ele tem acesso as alterações nos cadastros citados.

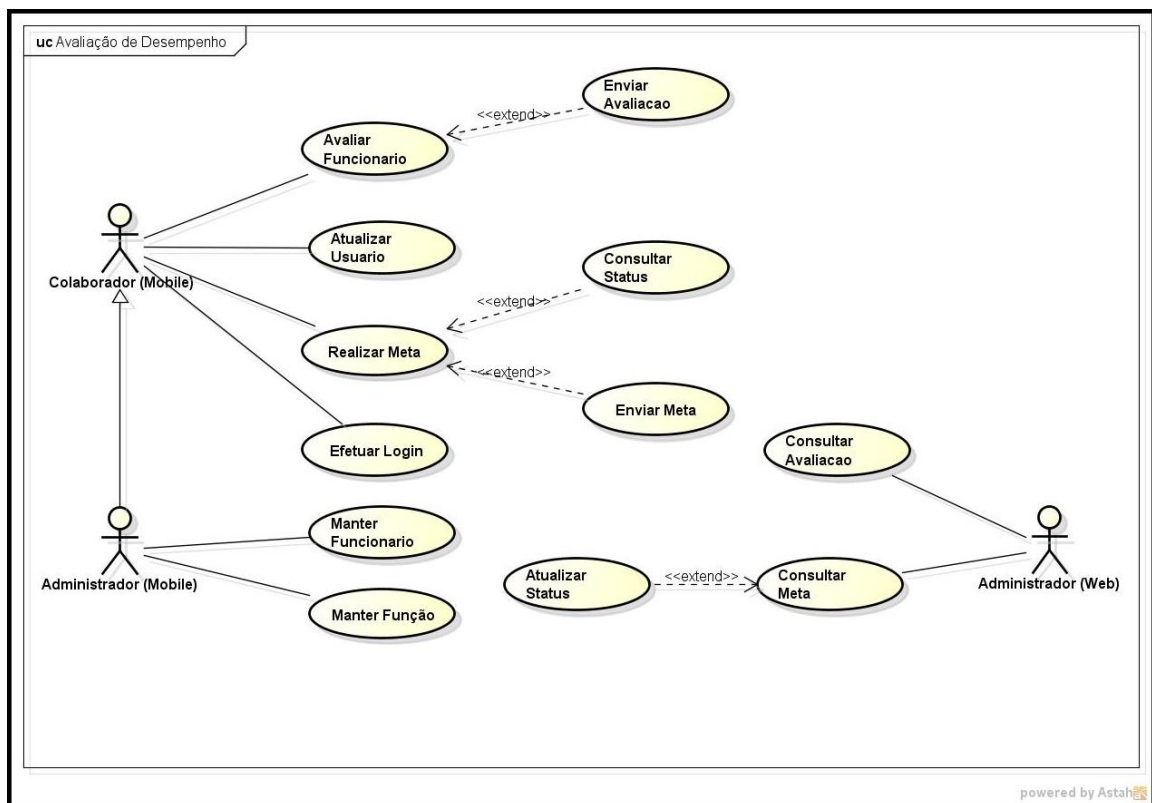


Figura 9 - Diagrama de caso de uso

3.4 ESPECIFICAÇÃO DOS CASOS DE USO

Para esclarecer as funções do sistema, abaixo serão descritos com detalhes cada função e seus fluxos, para isso serão apresentados para cada caso de uso: os objetivos, atores, pré-condições, evento inicial, fluxo principal, fluxos alternativos, fluxos de exceção e pós-condições.

A figura 10 apresenta o Caso de Uso Efetuar Login.

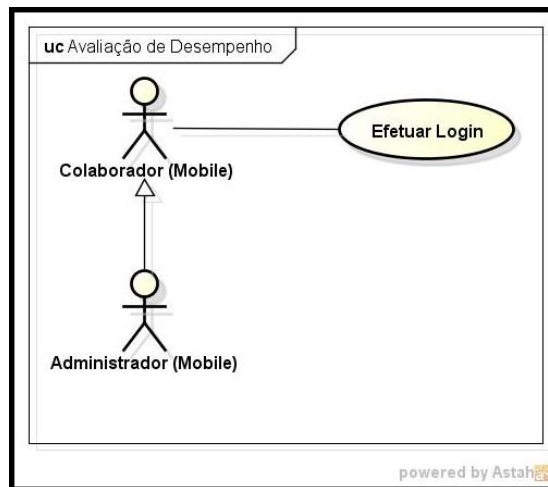


Figura 10 - Caso de Uso Efetuar Login

1. Finalidade/Objetivo

Permite ao Usuário logar no sistema.

2. Atores

Colaborador/Administrador

3. Pré-condições

O Usuário deve ter uma senha, um login e estar cadastrado como funcionário;

4. Evento inicial

O Colaborador/Administrador executa a aplicação.

5. Fluxo principal

a. O sistema verifica a existência do Usuário e Funcionário; (A1)

b. O sistema solicita o preenchimento dos campos: senha, login e funcionário;

c. O Usuário informa os dados;

d. O sistema solicita confirmação; (A2)

e. O caso de uso é encerrado.

6. Fluxos alternativos

A1 – Adicionar novo Usuário

- a. O Sistema solicita que adicione um novo Usuário;
- b. O Usuário seleciona o funcionário em questão.
- c. O Usuário adiciona os dados;
- d. O sistema solicita confirmação. (A2) (A3)

A2 – Cancelar Operação

- a. O Usuário Cancela a Operação;
- b. O sistema retorna ao passo 5.a. do Fluxo Principal.

A3 – Adicionar novo Funcionário

- a. O Sistema solicita que adicione um novo Funcionário;
- b. O Usuário adiciona os dados;
- c. O sistema solicita confirmação. (A1)

A figura 11 ilustra o Caso de Uso: Manter Funcionário

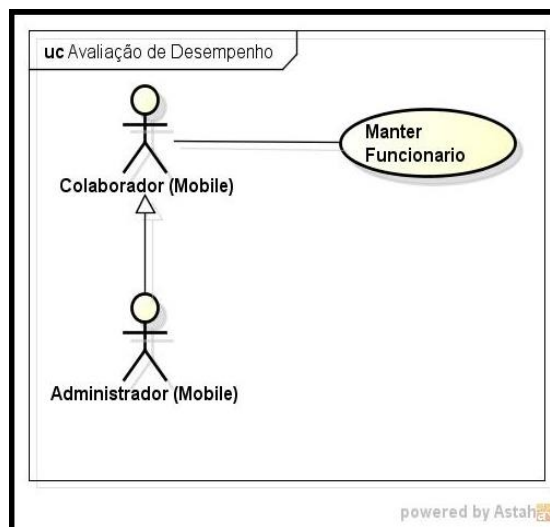


Figura 11 - Caso de Uso Manter Funcionário

1. Finalidade/Objetivo

Permite ao Administrador incluir e alterar os funcionários (colaborador/administrador).

2. Atores

Administrador

3. Pré-condições

O Administrador deve ter efetuado login no sistema.

4. Evento inicial

O Administrador escolhe a opção de Manter Colaborador.

5. Fluxo principal

a. O sistema oferece a interface de Manter Colaborador; (A2)(A3)(A4)

b. O caso de uso é encerrado.

6. Fluxos alternativos

A1 – Cancelar a operação

a. O Administrador cancela a operação;

b. O sistema retorna ao passo 5.a. do Fluxo Principal.

A2 – Incluir Funcionário

a. O Administrador solicita a inclusão de um novo Funcionário;

b. O sistema solicita a inclusão dos dados necessários;

c. O Administrador informa os dados;

d. O sistema solicita confirmação; (A1)

e. O sistema retorna ao passo 5.b. do Fluxo Principal.

A3 – Alterar Funcionário

a. O Administrador solicita a alteração de um Colaborador;

b. O Administrador faz as devidas alterações;

c. O sistema solicita confirmação; (A1)

d. O sistema retorna ao passo 5.a. do Fluxo Principal.

7. Pós-condição

O Administrador fecha o “Manter Funcionário”

A figura 12 ilustra o Caso de Uso: Atualizar Usuário

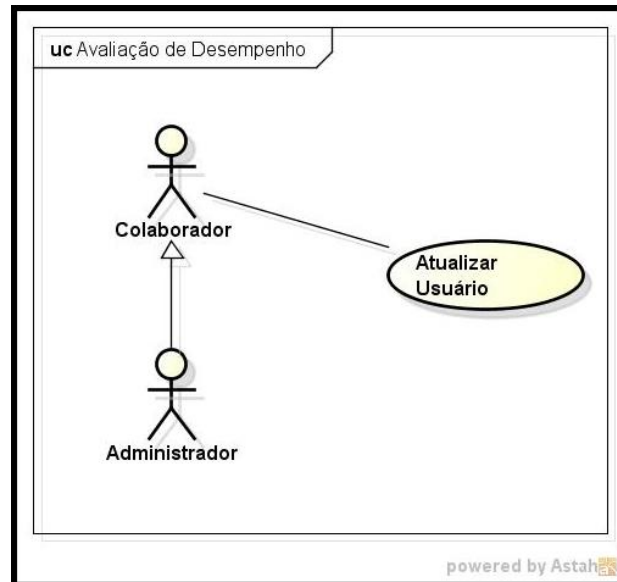


Figura 12 - Caso de Uso Atualizar Usuário

1. Finalidade/Objetivo

Permite ao Administrador/Colaborador atualizar o cadastro de usuário.

2. Atores

Administrador/Colaborador

3. Pré-condições

O Administrador/Colaborador deve ter efetuado login no sistema.

4. Evento inicial

O Administrador/Colaborador escolhe a opção de Atualizar Usuário.

5. Fluxo principal

- a. O sistema oferece a interface de Atualizar Usuário;
- b. O Administrador/Colaborador atualiza os dados; (A1)
- b. O caso de uso é encerrado.

6. Fluxos alternativos

A1 – Cancelar a operação

- a. O Administrador/Colaborador cancela a operação;
- b. O sistema retorna ao passo 5.a. do Fluxo Principal.

7. Pós-condição

O Administrador/Colaborador fecha o “Atualizar Usuário”

A figura 13 ilustra o Caso de Uso: Manter Função

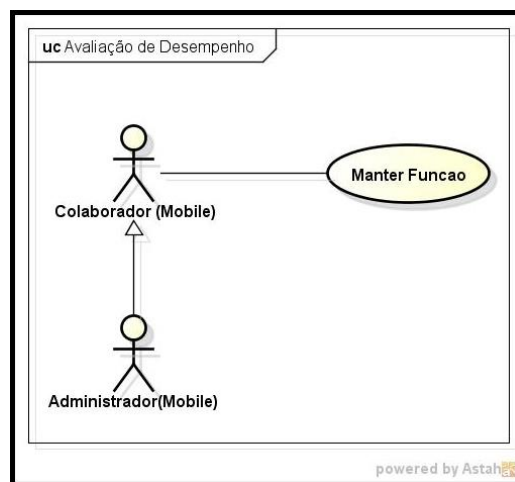


Figura 13 - Caso de Uso Manter Função

1. Finalidade/Objetivo

Permite ao Administrador incluir e alterar as funções.

2. Ator

Administrador

3. Pré-condições

O Administrador deve ter efetuado login no sistema.

4. Evento inicial

O Administrador escolhe a opção de Manter Função.

5. Fluxo principal

- a. O sistema oferece a interface de Manter Função; (A2)(A3)(A4)

b. O caso de uso é encerrado.

6. Fluxos alternativos

A1 – Cancelar a operação

- a. O Administrador cancela a operação;
- b. O sistema retorna ao passo 5.a. do Fluxo Principal.

A2 – Incluir Função

- a. O Administrador solicita a inclusão de um nova Função;
- b. O sistema solicita a inclusão dos dados necessários;
- c. O Administrador informa os dados;
- d. O sistema solicita confirmação; (A1)
- e. O sistema retorna ao passo 5.b. do Fluxo Principal.

A3 – Alterar Função

- a. O Administrador solicita a alteração de uma Função;
- b. O Administrador faz as devidas alterações;
- c. O sistema solicita confirmação; (A1)
- d. O sistema retorna ao passo 5.a. do Fluxo Principal.

7. Pós-condição

O Administrador fecha o “Manter Função”.

A imagem 14 ilustra o Caso de Uso: Avaliar Funcionário

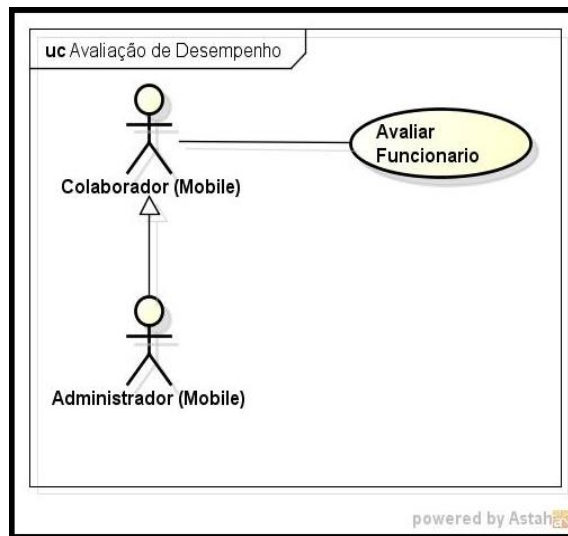


Figura 14 - Caso de Uso Avaliar Funcionário

1. Finalidade/Objetivo

Permite ao Administrador/Colaborador fornecer informações ao sistema a fim de avaliar o desenvolvimento dos funcionários da organização.

2. Atores

Administrador/Colaborador

3. Pré-condições

O Administrador/Colaborador deve ter efetuado login no sistema.

4. Evento inicial

O Administrador/Colaborador escolhe a opção de Realizar Avaliação.

5. Fluxo principal

- a. O sistema consulta o nome do usuário logado;
- b. O sistema apresenta uma lista de funcionários cadastrados no Sistema;
- c. O sistema consulta se o Funcionário já foi avaliado pelo usuário logado; (E1)
- d. O sistema apresenta o status do Funcionário com a descrição "Não Avaliado";
- e. O sistema solicita que o Administrador/Colaborador responda as questões para a avaliação;

- f. O Administrador/Colaborador informa os dados;
- g. O sistema solicita a confirmação; (A1)
- h. O sistema adiciona a avaliação;
- i. O caso de uso é encerrado.

6. Fluxo alternativo

A1 – Cancelar a operação

- a. O Administrador/Colaborador cancela a operação;
- b. O sistema retorna ao passo 5.a. do Fluxo Principal.

7. Fluxo de exceção

E1 – Funcionário já foi avaliado

- a. O sistema apresenta o status do Funcionário com a descrição “Avaliado”;
- b. O sistema retorna ao passo 5.b. do Fluxo Principal.

8. Pós-condições

O Administrador/Colaborador fecha a Avaliação do Funcionário.

A figura 15 ilustra o Caso de Uso: Consultar Avaliação

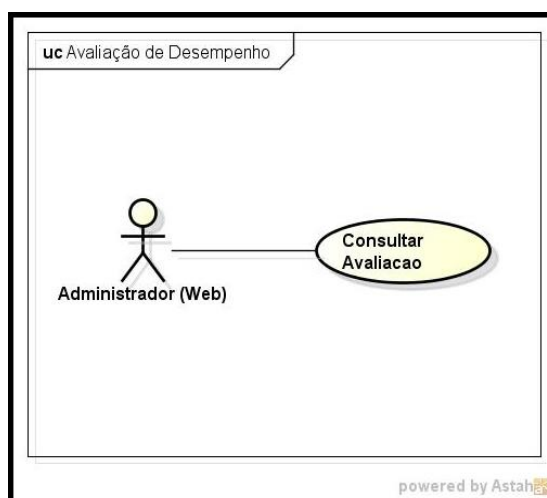


Figura 15 - Caso de Uso Consultar Avaliação

1. Finalidade/Objetivo

Permite ao Administrador consultar as informações sobre as avaliações realizadas.

2. Atores

Administrador

3. Pré-condições

O Administrador deve estar logado no sistema Web.

4. Evento inicial

O Administrador escolhe a opção de Consultar Avaliação.

5. Fluxo principal

a. O sistema oferece

a interface de consulta para avaliações;

b. O Sistema apresenta os dados para a filtragem; (A1)(A2)

c. . O Ator informa os dados;

d. O sistema exibe os dados das avaliações;

e. O caso de uso é encerrado.

6. Fluxo alternativo

A1 – Cancelar a operação

- a. O Administrador cancela a operação;
- b. O sistema retorna ao passo 5.a. do Fluxo Principal.

A2 – Não há avaliação cadastrada

- a. O Sistema não apresenta os dados para a filtragem;

7. Pós-condições

O Administrador fecha a Consulta de Avaliação.

A figura 16 ilustra o Caso de Uso: Realizar Meta.

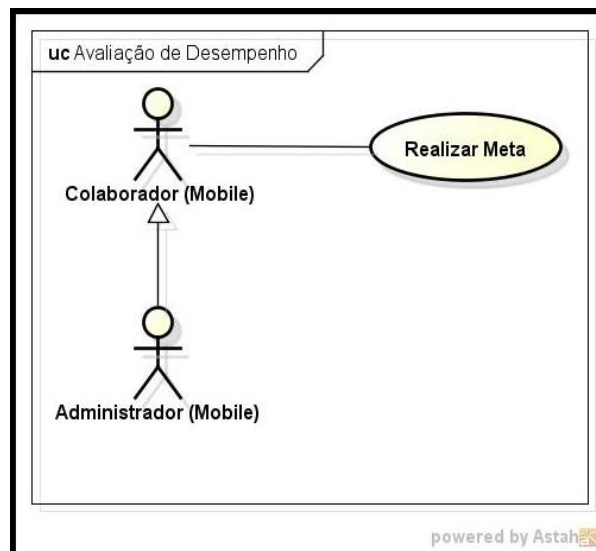


Figura 16 - Caso de Uso Realizar Meta

1. Finalidade/Objetivo

Permite ao Administrador/Colaborador realizar metas, ou seja, objetivos para o aprimoramento e melhor desempenho no trabalho.

2. Atores

Administrador/Colaborador

3. Pré-condições

O Administrador/Colaborador deve ter efetuado login no sistema.

4. Evento inicial

O Administrador/Colaborador escolhe a opção de Realizar Metas.

5. Fluxo principal

- a. O sistema consulta o nome do usuário logado;
- b. O sistema permite ao usuário o cadastro de uma meta;
- c. O usuário informa a meta e o critério da meta;
- d. O sistema solicita confirmação; (A1)
- e. O sistema adiciona a meta;
- f. O caso de uso é encerrado.

6. Fluxo alternativo

A1 – Cancelar a operação

- a. O Administrador/Colaborador cancela a operação;
- b. O sistema retorna ao passo 5.a. do Fluxo Principal.

7. Pós-condições

O Administrador/Colaborador fecha a Cadastro de Metas.

A figura 17 ilustra o Caso de Uso: Consultar Meta.

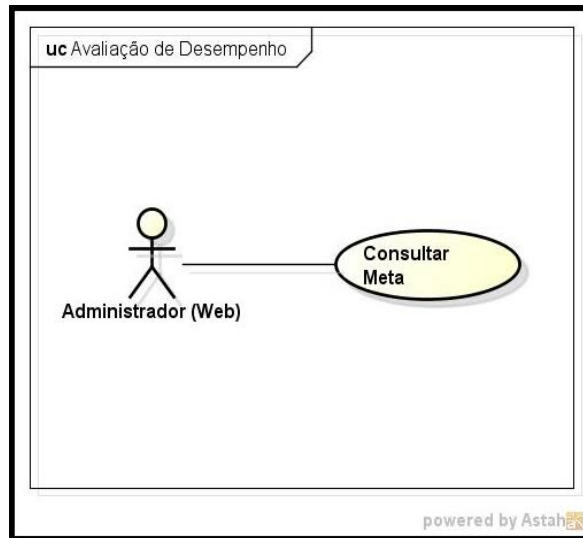


Figura 17 - Caso de Uso Consultar Meta

1. Finalidade/Objetivo

Permite ao Administrador consultar as informações sobre as metas realizadas.

2. Atores

Administrador

3. Pré-condições

O Administrador deve estar logado no sistema Web.

4. Evento inicial

O Administrador escolhe a opção de Consultar Meta.

5. Fluxo principal

- a. O sistema oferece a interface de consulta para meta;
- b. O Sistema apresenta os dados para a filtragem; (A1)(A2)
- c.. O Ator informa os dados;
- d. O sistema exibe os dados das metas;
- e. O caso de uso é encerrado.

6. Fluxo alternativo

A1 – Cancelar a operação

- a. O Administrador cancela a operação;
- b. O sistema retorna ao passo 5.a. do Fluxo Principal.

A2 – Não há meta cadastrada

- a. O Sistema não apresenta os dados para a filtragem;

7. Pós-condições

O Administrador fecha a Consulta de Meta.

A figura 18 ilustra o Caso de Uso: Enviar Meta.

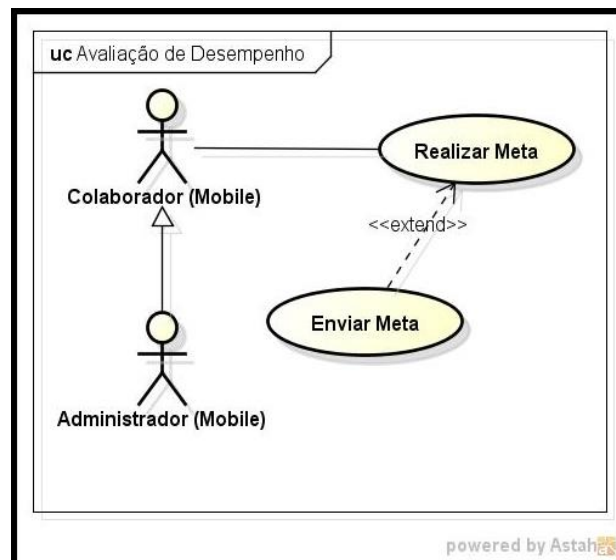


Figura 18 - Caso de Uso Enviar Meta

1. Finalidade/Objetivo

Permite ao Administrador/Colaborador enviar as metas cadastradas na aplicação móvel para a Web.

2. Atores

Colaborador/Administrador

3. Pré-condições

O Administrador/Colaborador deve ter efetuado login no sistema;

O Administrador/Colaborador deve uma meta cadastrada.

4. Evento inicial

O Administrador/Colaborador escolhe a opção Enviar Meta.

5. Fluxo principal

a. O sistema verifica a existência de conexão com a Internet; (A1)

b. O sistema envia a meta cadastrada para a aplicação Web;

c. O caso de uso é encerrado.

6. Fluxos alternativos

A1 – Sem Conexão

a. O Sistema emite uma mensagem informando que não há conexão para o envio dos dados;

b. O sistema retorna á tela Inicial.

A figura 19 ilustra o Caso de Uso: Enviar Avaliação.

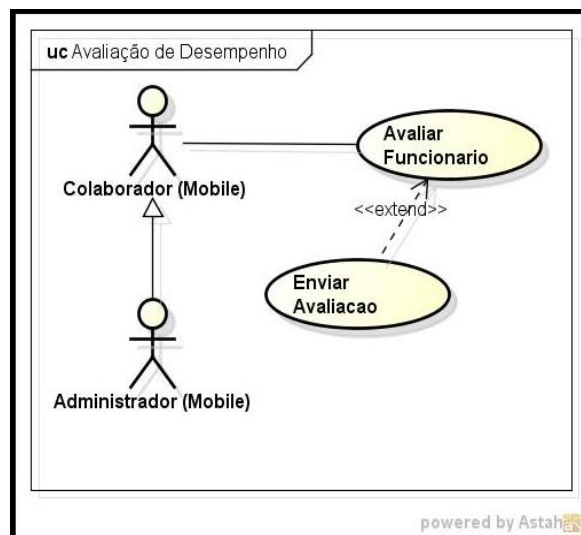


Figura 19 - Caso de Uso Enviar Avaliação

1. Finalidade/Objetivo

Permite ao Administrador/Colaborador enviar a avaliação cadastrada na aplicação móvel para a Web.

2. Atores

Colaborador/Administrador

3. Pré-condições

O Administrador/Colaborador deve ter efetuado login no sistema;

O Administrador/Colaborador deve uma avaliação cadastrada.

4. Evento inicial

O Administrador/Colaborador escolhe a opção Enviar Avaliação.

5. Fluxo principal

a. O sistema verifica a existência de conexão com a Internet; (A1)

b. O sistema envia a avaliação cadastrada para a aplicação Web;

c. O caso de uso é encerrado.

6. Fluxos alternativos

A1 – Sem Conexão

a. O Sistema emite uma mensagem informando que não há conexão para o envio dos dados;

b. O sistema retorna á tela Inicial.

A figura 20 ilustra o Caso de Uso: Consultar Status

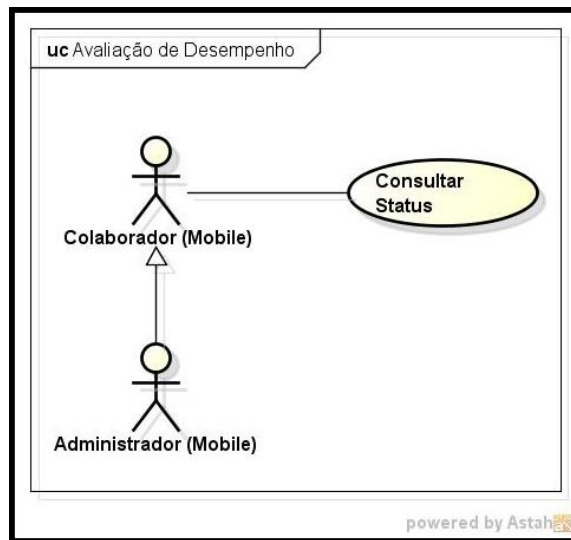


Figura 20 - Caso de Uso Consultar Status

1. Finalidade/Objetivo

Permite ao Administrador/Colaborador consultar o status da meta, ou seja, verificar se a meta foi realizada ou não.

2. Atores

Administrador/Colaborador

3. Pré-condições

O Administrador/Colaborador deve ter efetuado login no sistema.

4. Evento inicial

O Administrador/Colaborador escolhe a opção de Consultar Status.

5. Fluxo principal

a. O sistema verifica a existência de conexão com a Internet; (A2)

b. O sistema exibe o status das meta; (A1)

c. O caso de uso é encerrado.

6. Fluxo alternativo

A1 – Cancelar a operação

a. O Administrador/Colaborador cancela a operação;

b. O sistema retorna ao passo 5.a. do Fluxo Principal.

A2 – Sem Conexão

a. O Sistema emite uma mensagem informando que não há conexão com a Internet para a consulta do Status;

b. O sistema retorna á tela Inicial.

7. Pós-condições

O Administrador/Colaborador fecha a Consulta de Status da Meta.

A figura 21 ilustra o Caso de Uso: Atualizar Status

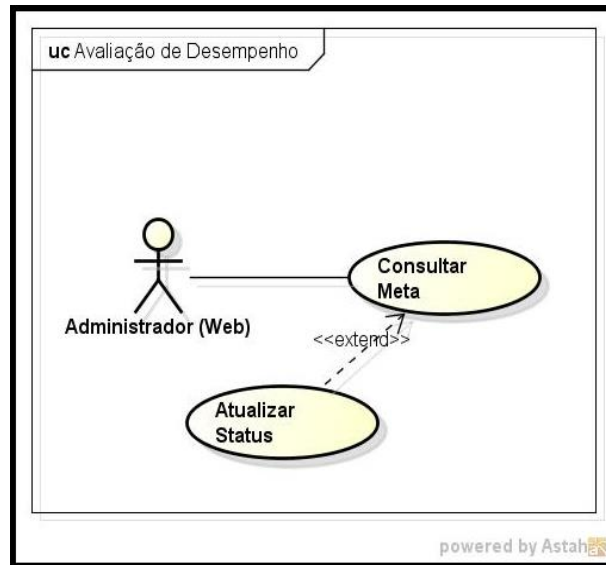


Figura 21 - Caso de Uso Atualizar Status

1. Finalidade/Objetivo

Permite ao Administrador atualizar o status da meta, ou seja, definir se a meta foi realizada ou não.

2. Atores

Administrador

3. Pré-condições

O Administrador deve estar logado no sistema Web.

O Administrador deve ter uma meta cadastrada no sistema Web.

4. Evento inicial

- a. O Administrador escolhe a opção de Atualizar Status.
- b. O Sistema oferece a interface de atualizar o status meta;
- c. O Sistema apresenta a meta; (A1)(A2)
- d. O Ator informa o status da meta;
- e. O caso de uso é encerrado.

6. Fluxo alternativo

A1 – Cancelar a operação

- a. O Administrador cancela a operação;
- b. O sistema retorna ao passo 5.a. do Fluxo Principal.

A2 – Não há meta cadastrada

- a. O Sistema emite uma mensagem informando que não há meta cadastrada;
- b. O sistema retorna á tela Inicial.

7. Pós-condições

O Administrador/Colaborador fecha a tela Atualizar Status.

3.5 DIAGRAMA DE ATIVIDADE

O Diagrama de atividades como já mencionado anteriormente tem como objetivo modelar o comportamento dinâmico do sistema, ou seja, mostrar cada situação/atividade exibindo passo a passo. Portanto, serão expostas as imagens dos diagramas de atividades de alguns casos de uso representados na seção anterior.

A figura 22 ilustra o diagrama de atividade do caso de uso Efetuar Login.

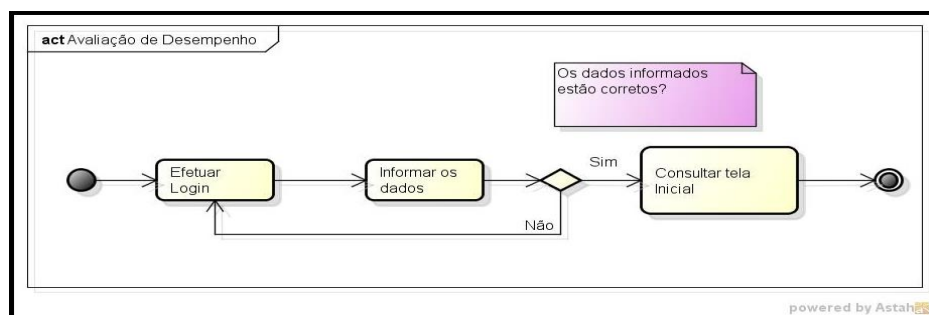


Figura 22 - Diagrama de Atividade Efetuar Login

Após sabermos como o sistema se comporta ao fazer o login no sistema, o próximo diagrama de atividade que informa passo a passo de como funciona as atividades e

especificações, é o de Avaliar Funcionário, que basicamente contém os mesmos passos em busca de uma única conclusão: Avaliar os funcionários da empresa. A figura 23 representa com maior clareza os passos informados.

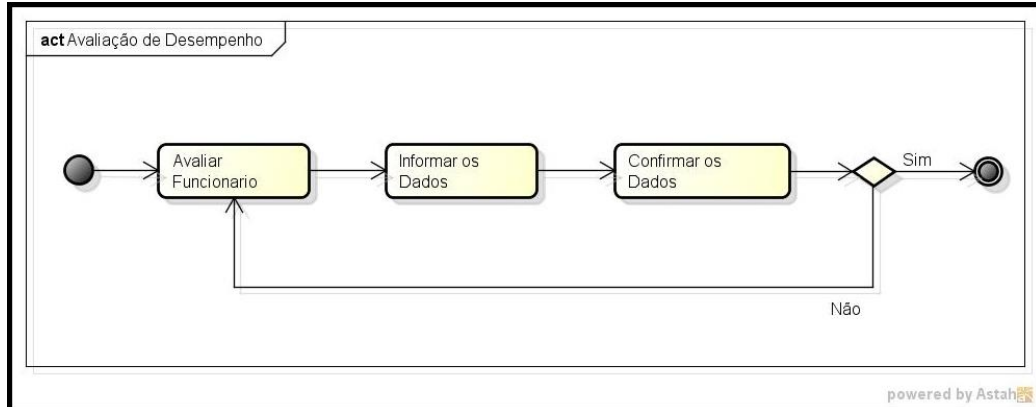


Figura 23 - Diagrama de Atividade Avaliar Funcionário

A figura 24 representa o diagrama de atividade do caso de uso Realizar Meta, em que caso não confirme os dados (cancele a operação) o sistema rompe com o procedimento e retorna ao passo inicial, caso contrário os dados são salvos.

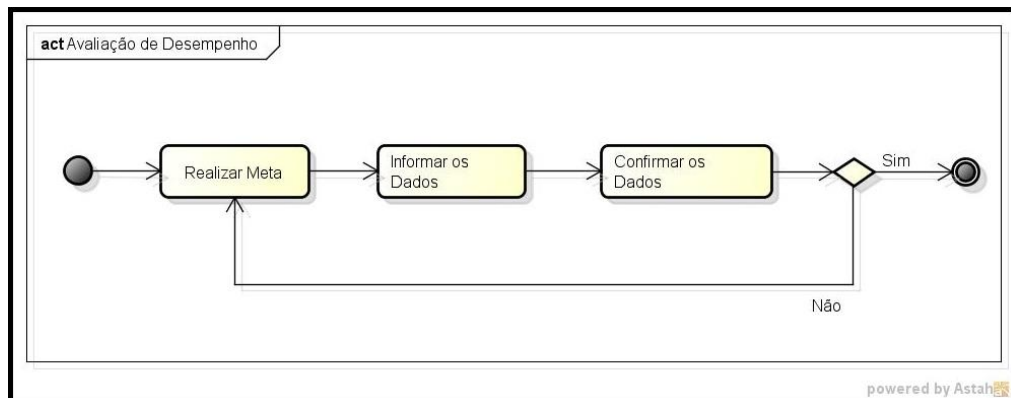


Figura 24 - Diagrama de Atividade Realizar Meta

3.6 DIAGRAMA DE CLASSE

O diagrama de classes, como mencionado anteriormente, representa as classes e a relação entre elas. As classes são compostas por métodos e atributos. A figura 25 ilustra as classes do sistema de Avaliação de Desempenho, sendo a classe ConsultaAvaliação e ConsultaMeta pertencentes á aplicação Web.

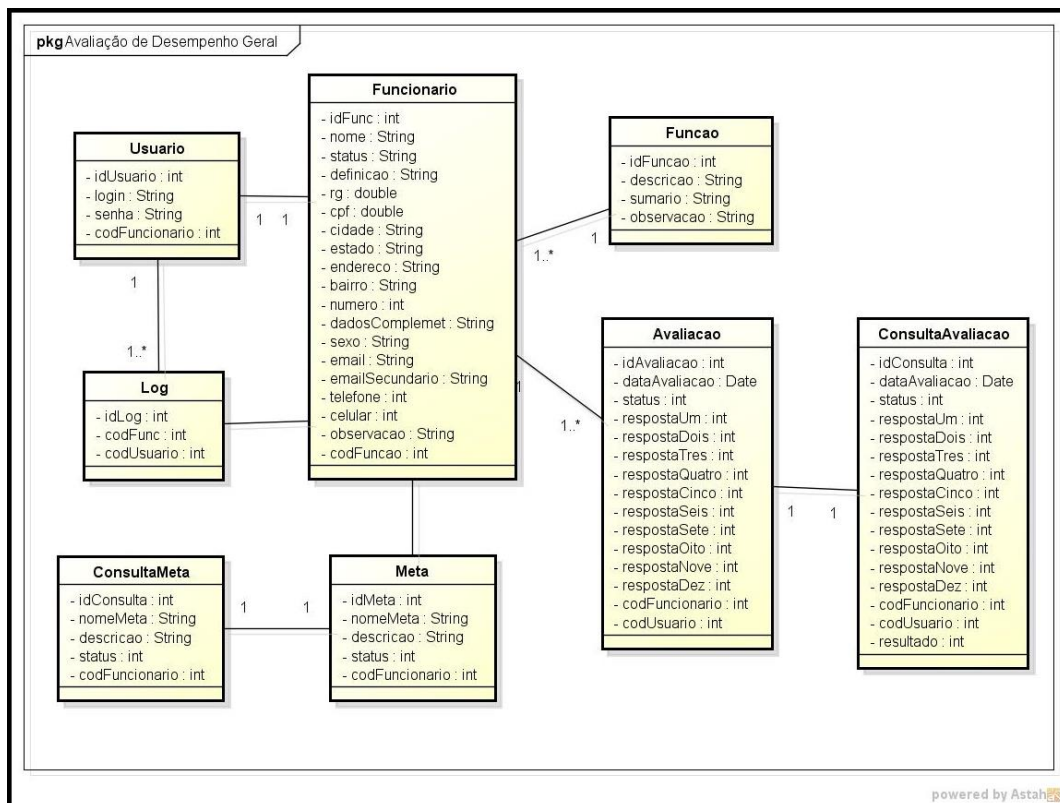


Figura 25 - Diagrama de Classe Geral

3.7 DIAGRAMA ENTIDADE RELACIONAMENTO

O Diagrama Entidade Relacionamento – DER ilustra o modelo de dados de um sistema. A figura 26 ilustra a representação do modelo de dados da aplicação Mobile e Web, sendo que as entidades da aplicação Web são: Consultar_Meta e

Consultar_Avaliação que são alimentadas pelo Sistema, especificamente pelas entidades Avaliação e Meta.

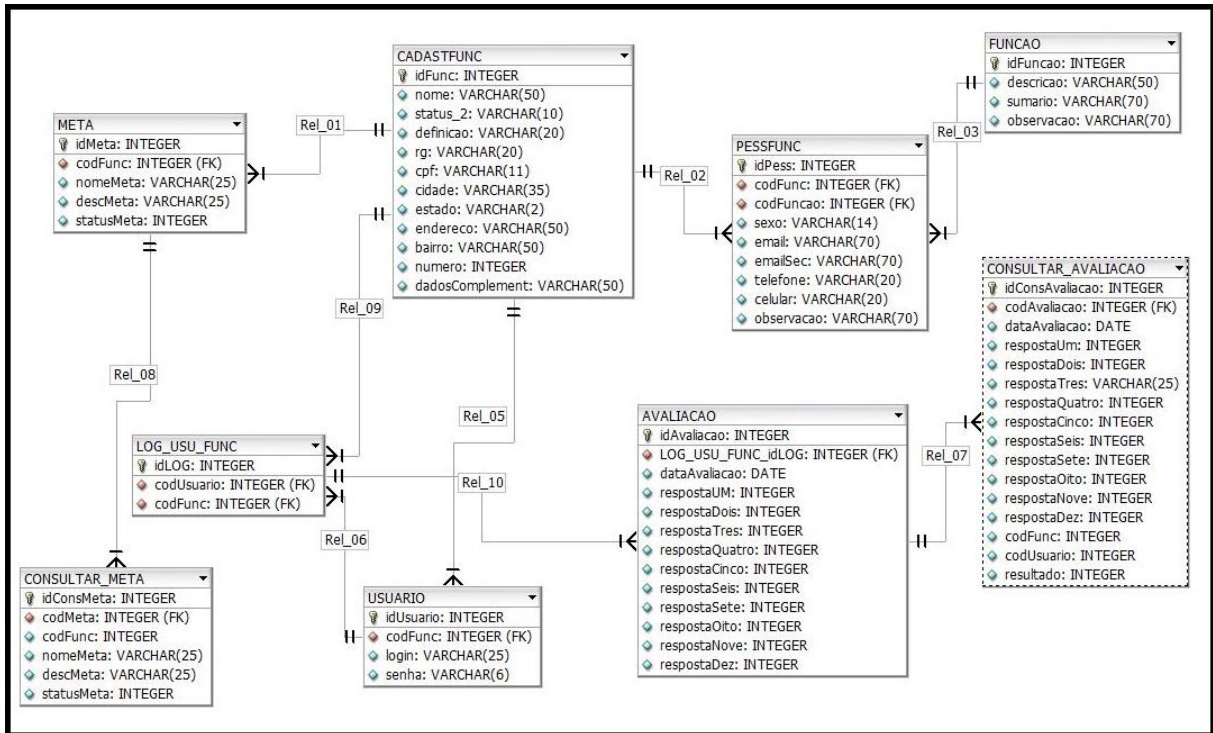


Figura 26 - Diagrama Entidade Relacionamento Geral

4. PLANEJAMENTO DO PROJETO

O projeto é realizado em etapas, ou seja, com início, meio e fim. Primeiramente foi definido o escopo, que é o levantamento dos requisitos para entregar o produto final com o resultado esperado. Após ser definido o escopo, é necessário assegurar que o projeto será realizado no tempo previsto, portanto é indispensável que seja declarado o tempo que levará até o término do trabalho. Outra etapa a ser definida é a da qualidade, que irá garantir que o produto deve satisfazer as necessidades do usuário. As últimas etapas a serem definidas são os riscos e as partes interessadas, em que é possível identificar os riscos do projeto e as organizações que possam utilizar o produto, respectivamente.

Escopo: O escopo é uma etapa principal que pode ser definida pela figura 27, que ilustra os segmentos do projeto através do EAP- Estrutura Analítica de Projetos.

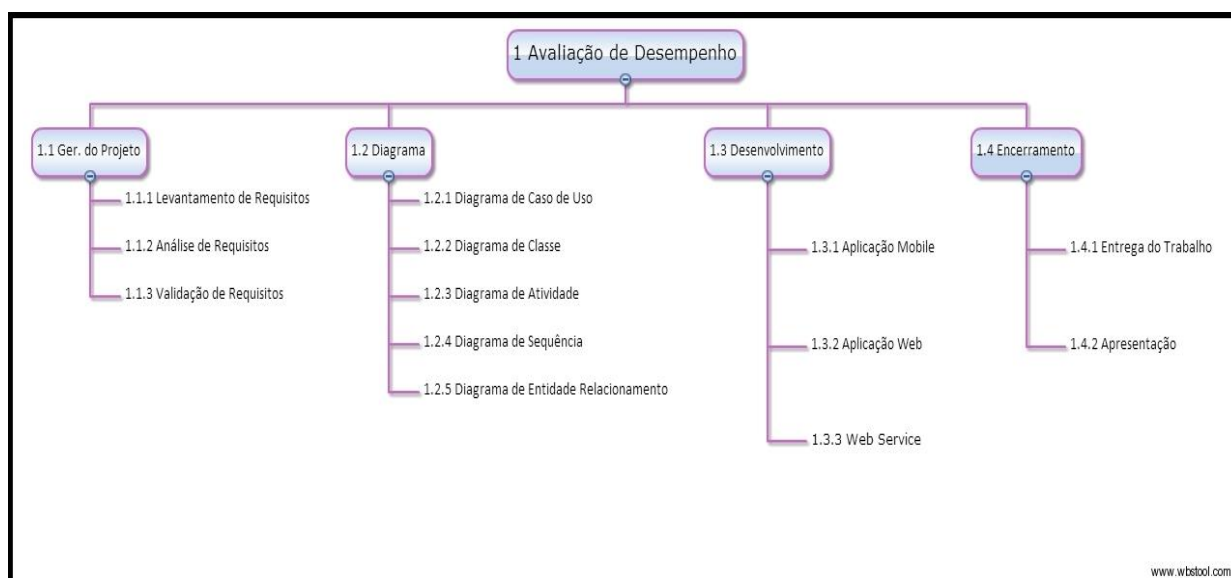


Figura 27 – Estrutura Analítica do Projeto - EAP

Qualidade: Integração da aplicação mobile e a aplicação Web, fornecendo consultas dos dados digitados no aplicativo móvel na Web.

Partes interessadas: As organizações, independente do segmento poderão usar as aplicações, pois a avaliação de desempenho tende a proporcionar que os colaboradores verifiquem como está o seu comportamento no ambiente de trabalho.

5. DESENVOLVIMENTO DO SISTEMA

Neste trabalho foi construída uma aplicação Web e uma aplicação Mobile, em que os dados informados no aplicativo são expostos na Web via Web Service. A sugestão da aplicação Mobile é colher os dados de forma exterior ao ambiente do trabalho e salvá-las em seu banco de dados para posteriormente serem enviadas, via Internet, para o software Web. O Anexo A apresenta o formulário original para a avaliação do desempenho e que foi adaptado para o cenário aqui estabelecido.

5.1 PROJETO MOBILE COM ANDROID

Nesse trabalho foi construído uma aplicação móvel utilizando a plataforma Google Android e o banco de dados SQLite.

A aplicação Mobile é responsável por colher às informações geradas no ambiente externo da organização, sendo assim, os colaboradores poderão fazer a avaliação dos seus colegas de trabalho no horário que estiver disponível, não o comprometendo com um horário.

As informações que estão no aplicativo Android poderão ser enviadas para a aplicação Web através de uma boa conexão com a internet. Os apêndices A e B contêm as interfaces Android das Metas e das Avaliações.

Para finalizar este tópico, vale salientar que as informações do aplicativo interagem com a aplicação Web, utilizando o formato JSON, através das requisições HTTP já mencionadas.

A figura 28 ilustra a interface da aplicação, a tela de login e a sua tela inicial composta pelas opções de acesso ao usuário.

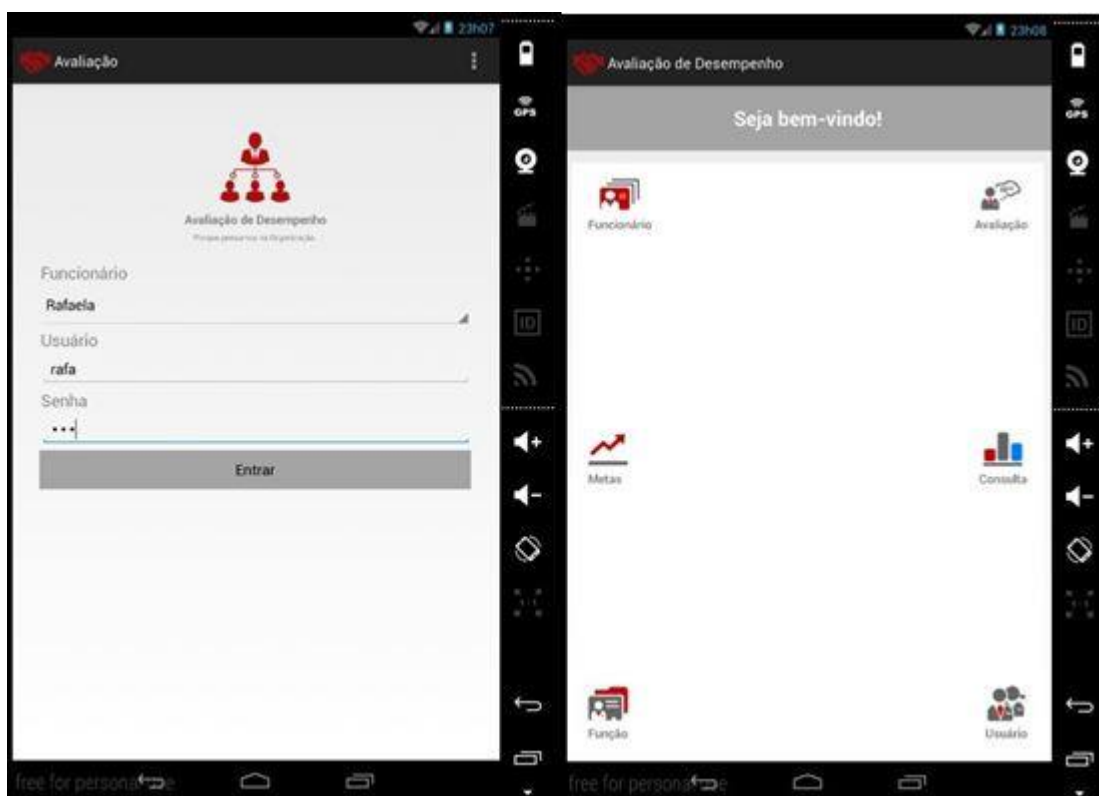


Figura 28 - Telas da Aplicação Mobile

5.2 APLICAÇÃO WEB

Para a construção da aplicação Web foi utilizada a plataforma Java EE com a linguagem Java, pois como relatado anteriormente a plataforma contém vários recursos que facilitam a construção do ambiente Web.

Dentre os vários recursos presentes na plataforma Java EE, foram utilizados os seguintes: JAX-RS para trabalhar com o Web Service, Java Persistence (JPA) para fazer a persistência de dados e o Tomcat que é um servidor de aplicações Java para Web. Para a interface será usada a biblioteca de componentes PrimeFaces.

O banco de dados para armazenar as informações da aplicação é o MySql por ser muito usado em softwares Web.

Portanto, a aplicação Web disponibiliza as interfaces para os usuários a fim de visualizar os dados digitados na aplicação móvel. Os apêndices C e D apresentam as interfaces construídas para a aplicação web sobre as Metas e as Avaliações.

5.3 WEB SERVICE RESTFUL

O Web Service constitui em técnicas dos padrões *Representational State Transfer* (REST) que utilizam o protocolo HTTP como já apresentado anteriormente.

O Web Service tem como principal objetivo integrar as aplicações de diferentes plataformas, nesse caso, uma aplicação móvel com uma aplicação Web, para que assim a aplicação Web permita gravar os dados que foram digitados no aparelho móvel.

6. CONCLUSÃO

A motivação pelo tema escolhido se deve ao fato de que os colaboradores de uma organização são uma das partes fundamentais da mesma como um todo, aos quais contribuem para o seu sucesso e ao fato dos celulares invadirem a sociedade, pela sua alta facilidade e praticidade no dia-a-dia.

O objetivo geral desse trabalho foi desenvolver e implementar uma aplicação *Mobile* e *Web* que sirvam como um recurso para avaliação de desempenho dos colaboradores de uma empresa. Deste modo, acredita-se que os colaboradores, ao realizarem metas e serem avaliados pelos seus colegas de trabalho, tenham uma melhora em seu comportamento, se dedicando ao trabalho e esquematizando metas com o objetivo de realizá-las.

Como trabalhos futuros, pretende-se desenvolver na aplicação *Web*, gráficos para consulta dos dados realizados no dispositivo móvel. Para a aplicação *mobile*, pretende-se colocar a foto do colaborador/administrador no cadastro do Funcionário e ainda neste mesmo cadastro conectar-se ao GPS indicando a localidade através dos mapas.

REFERÊNCIAS

- ANDROID. **The Android Source Code**. Disponível em: <https://source.android.com/source/>. Acesso em: 23 fev. 2015.
- BOOCH, Grady; RUMBAUGH, James; JACOBSEN, Ivar. **UML: Guia do Usuário**. Tradução de Fábio Freitas da Silva e Cristina de Amorim Machado. 2. Ed. Rio de Janeiro: Editora Elsevier, 2005.
- CAELUM. **Java para desenvolvimento Web**. Disponível em: <https://www.caelum.com.br/download/caelum-java-web-fj21.pdf>. Acesso em: 15 fev. 2015.
- CERAMI, Ethan. **Web Services Essentials – Distributed Applications With XM – RPC, SOAP, UDDI e WSDL**. 1. ed. Sebastopol: O’ Reilly Media, 2002.
- FARIA, Alessandro de Oliveira. Programe seu Andróide. **Linux Magazine**, v.1, n.43, 2008, p. 73-77.
- FARIA, Thiago. **Java EE 7 com JSF, PrimeFaces e CDI**. São Paulo: AlgaWorks, 2013.
- FILHO, Luiz Carlos Q. **As novidades da Java EE 7 na prática**. Disponível em: <http://www.devmedia.com.br/as-novidades-da-java-ee-7-na-pratica/30071>. Acesso em: 18 fev. 2015.
- GONÇALVES, Antonio. **Beginning Java EETM 6 Platform With GlassFish™ 3 – From Novice to Professional**. 2. ed. New York: Apress, 2010.
- HAMANN, Renan. **Android: a história do robô-mascote mais famoso do mundo [ilustração]**. Disponível em: <http://www.tecmundo.com.br/android/46241-android-a-historia-do-robô-mascote-mais-famoso-do-mundo-ilustracao-.htm>. Acesso em: 20 fev. 2015.
- KING, Chris; ORTIZ, C. Enrique; SEM, Robi; ABLESON, W. Frank. **Android em Ação**. 3ª ed. Tradução de Manning Publications Co. Rio de Janeiro: Editora Elsevier, 2012.
- LARMAN, Craig. **Utilizando UML e padrões- Uma introdução à análise e ao projeto orientados a objetos e ao desenvolvimento iterativo**. 3 Ed. São Paulo: Editora Bookman, 2005.
- LECHETA, Ricardo R. **Google Android: aprenda a criar aplicações para dispositivos móveis com o Android SDK**. 3. ed. São Paulo: Novatec Editora, 2013.
- LOBO, Edson J. R. **Guia Prático de Engenharia de Software**. São Paulo: Editora Digerati Books, 2009.
- MACEDO, Alexandre. **Criando um Webservice Restful em Java**. Disponível em: <http://www.k19.com.br/artigos/criando-um-webservice-restful-em-java/>. Acesso em: 02 mar. 2015.
- MARTINEZ, Marina. **UML**. Disponível em: <http://www.infoescola.com/engenharia-de-software/uml/>. Acesso em: 04 mar. 2015.
- MARTINS, José Carlos Cordeiro. **Gerenciando projetos de desenvolvimento de software com PMI, RUP e UML**. 5 Ed. Rio de Janeiro: Brasport, 2010.
- MEDEIROS, Higor. **Conhecendo Web Services**. Disponível em: <http://www.devmedia.com.br/conhecendo-web-services/5070>. Acesso em: 18 fev. 2015.
- MONTEIRO, João Bosco. **Google Android- Crie aplicações para celulares e tablets**. São Paulo: Casa do Código, 2012.
- NEWCOMER, Eric. **Understanding Web Services – XML, WSDL, SOAP and**

UDDI. Indianapolis: Pearson Education Corporate Sales Division, 2002.

Open Handset Alliance. **O que seria necessário para construir um celular melhor?** Disponível em: <http://www.openhandsetalliance.com/>. Acesso em: 19 fev. 2015.

PENCHIKALA, Srini. **Java EE 6 – Suporte a REST com Anotações no JAX-RS 1.1.** INFOQ – Traduzido por Renato Marco. Disponível em: <<http://www.infoq.com/br/news/2010/02/javaee6-rest>>. Acesso em: 02 mar. 2015.

PEREIRA, Lúcio Camilo Oliveira; SILVA, Michel Lourenço. **Android para Desenvolvedores.** Rio de Janeiro: Editora Brasport, 2009.

POLÔNIA, Pablo Valério. **Proposta de Arquitetura Orientada a Recursos para Scada na Web.** 2011. 145p. Dissertação (mestrado) – Universidade Federal de Santa Catarina – Centro Tecnológico, Santa Catarina, Florianópolis, 2011.

RICHARDSON, Leonard; RUBY, Sam. **RESTful Web Services**, First Edition.

Gravenstein Highway North: O'Reilly, 2007.

SAUDATE, Alexandre. **REST – Construa API's inteligentes de maneira simples. 2013.** São Paulo: Casa do Código, 2013.

SAMPAIO, Cleuton. **Java Enterprise Edition 6 – Desenvolvendo Aplicações Corporativas.** 1 ed. Rio de Janeiro: Brasport Livros e Multimídia Ltda, 2011.

SENAI. **Avaliação de Desempenho.** Disponível em: <http://campinas.sp.senai.br/>. Acesso em: 18 set. 2015.

SILVA, Edgar. **A era dos Jetsons já começou, e o REST tem papel importante nisso.** Linux Magazine, v.1, n.43, 2008, p. 28.

THE MOVIE THEME SONG. Google – Android – Be Together Not The Same Pictures & Wallpapers. Disponível em: <http://www.themoviethemesong.com/google-android-be-together-not-the-same-tv-advert/>. Acesso em: 20 fev. 2015.

APÊNDICE A – INTERFACES ANDROID - METAS

Esse item é constituído pela interface gráfica das metas no ambiente mobile que são formadas por basicamente quatro telas, sendo elas: tela inicial, cadastro, lista e consulta status, conforme apresentado abaixo pelas figuras 29, 30, 31 e 32.

A tela inicial é representada pelos botões “Cadastro” e “Lista”, conforme a figura 29 ilustra.

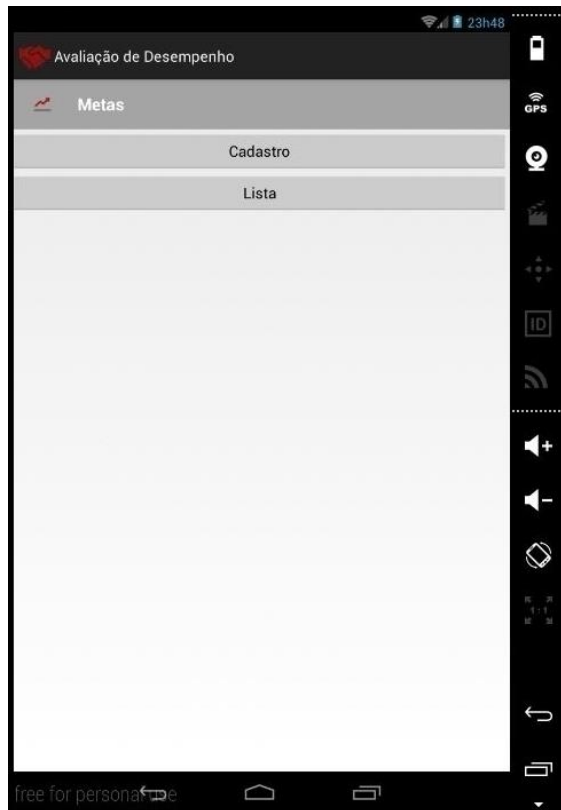


Figura 29 - Tela Inicial – Metas

A tela de cadastro, conforme a figura 30 ilustra é utilizada para os colaboradores e administradores cadastrarem as metas, sendo composta pelos seguintes campos: nome, descrição (deve informar os detalhes da meta nomeada), valores que é definido como o critério da avaliação da meta, ou seja, para o administrador do sistema Web conseguir identificar se a meta foi realizada ou não, e por fim o campo status que é preenchido com o valor default igual a zero, indicando que está aguardando avaliação.

O valor do status só sofrerá alteração quando o administrador avaliar a meta e quando o usuário consultar a mesma.

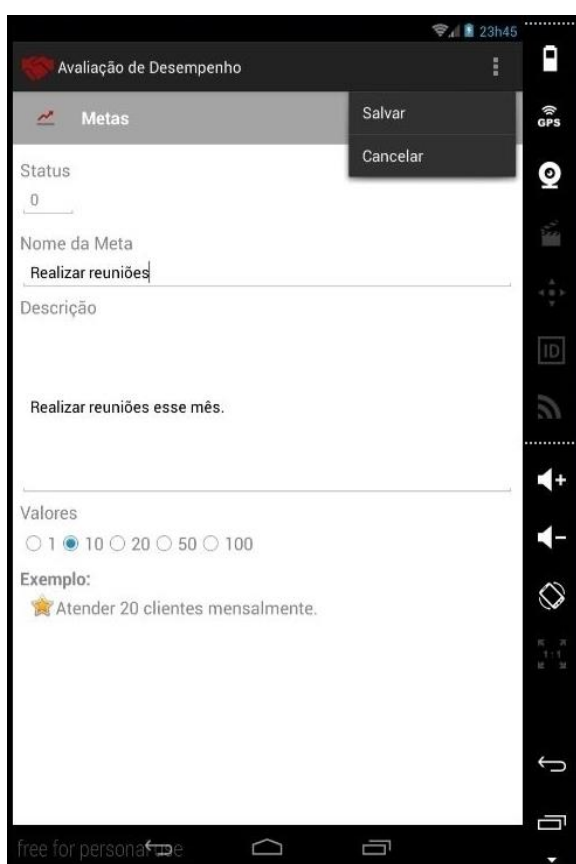


Figura 30 - Cadastro Metas

A tela de consultar status, conforme apresentado abaixo, contém basicamente os mesmos campos que tela de cadastro, porém contém o botão “Consultar Status” e a legenda do status indicando que o valor zero indica que a meta está aguardando avaliação, o valor um indica que a meta foi realizada e por fim, o valor 2 significa que a meta não foi realizada. Lembrando que essa avaliação será realizada pelo administrador do sistema Web.

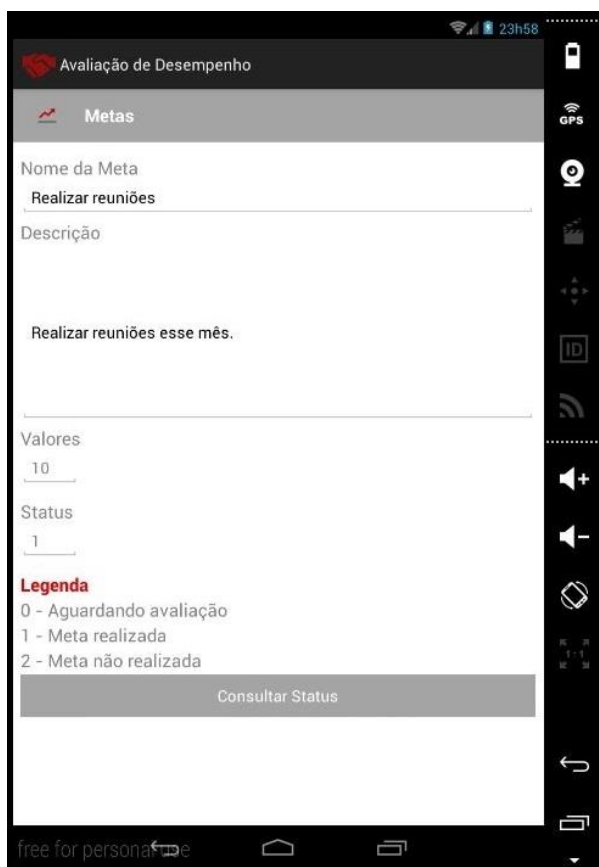


Figura 31 - Consultar Meta

A figura 32 ilustra a lista das metas, como o próprio nome diz, é a lista de todas as metas cadastradas por usuário.



Figura 32 - Lista Metas

APÊNDICE B – INTERFACES ANDROID – AVALIAÇÕES

Esse item é constituído pelas interfaces gráficas das avaliações no ambiente mobile que são formadas por basicamente cinco telas, sendo elas: tela inicial, tela de avaliar funcionário que é composta por 3 telas e por fim a tela da lista das avaliações.

A tela inicial é representada pelos botões “Avaliação”, conforme a figura 33 ilustra.

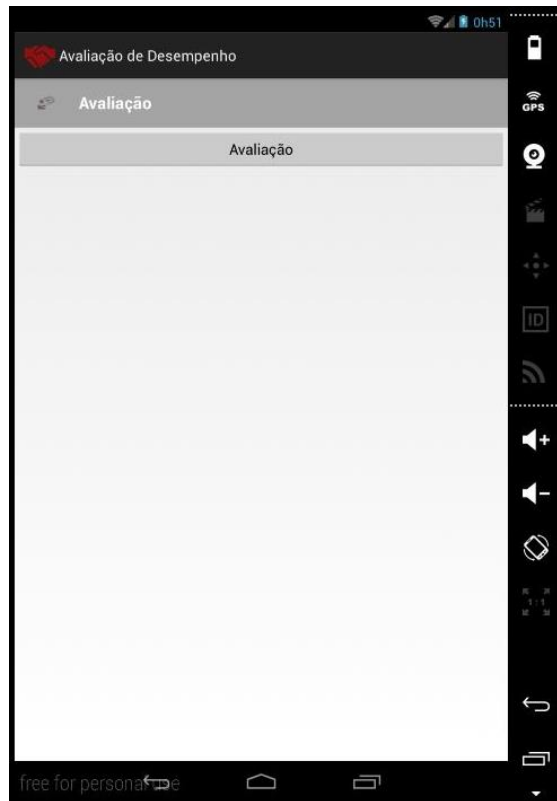


Figura 33 - Tela Inicial - Avaliações

A tela para avaliar o funcionário contém 3 etapas, conforme as figuras 34, 35, 36 lustram, sendo cada uma delas composta por uma tela conforme ilustrado nas figuras . Cada pergunta possui quatro respostas, o usuário poderá selecionar apenas uma, sendo a primeira valendo 100%, a segunda 75%, a terceira 50% e a quarta 25%. Quando clicar no menu “Salvar” o sistema irá fazer a comunicação com o Web Service assim, encaminhando os dados da avaliação para o Sistema Web.

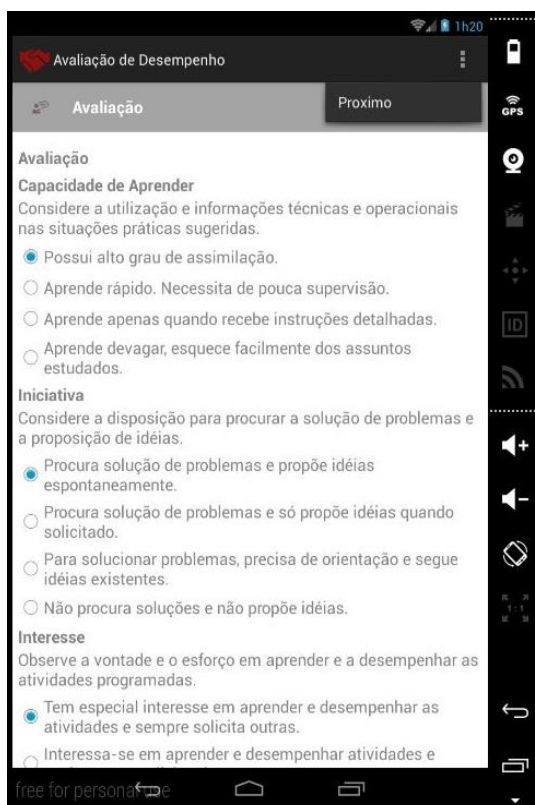


Figura 34 - Avaliação - Primeira Etapa

Avaliação de Desempenho

Avaliação Proximo

Avaliação

Responsabilidade
 Considere a forma como desempenha suas tarefas e a confiança que inspira quando lhe são atribuídas.

- É de inteira confiança, sempre preocupado com as tarefas atribuídas.
- Não precisa ser lembrado sobre as tarefas que deve realizar, demonstrando ser merecedor de confiança.
- Pode-se contar com ele desde que se sinta controlado.
- Não consegue assumir as tarefas e não responde por elas.

Ética
 Considere a maneira como se relaciona com os colegas, supervisores e a integração com a equipe de trabalho.

- Mantém ótimas relações com o pessoal; integra-se com facilidade a equipe.
- Relaciona-se bem, geralmente integra-se com a equipe.
- Apresenta dificuldades em relacionar-se. Demora a integrar-se a equipe.
- Apresenta dificuldades em se relacionar; não se integra a equipe.

Assiduidade
 Observe o cumprimento dos horários preestabelecidos, considerando: atrasos, faltas e solicitações de saída.

- Não atrasa, não falta e não solicita saídas antecipadas
- Algumas vezes atrasa, falta ou solicita saídas antecipadas.

free for personal use

Figura 35 - Avaliação - Segunda Etapa

Avaliação de Desempenho

Avaliação Salvar

Avaliação

Segurança
 Considere a disposição em cumprir os hábitos e normas de segurança.

- Observa e cumpre todas as normas.
- Quando supervisionado, segue as normas.
- Necessita freqüentemente ser alertado sobre as normas.
- Não observa, na maioria das vezes, as normas de segurança.

Indicadores
 Considere a demonstração de atitudes relacionadas com os serviços da empresa e os interesses do cliente.

- Mantém ótimas relações com os clientes e adota atitudes cabíveis com facilidade, mesmo frente a situações complexas.
- Relaciona-se bem com os clientes e geralmente adota atitudes mesmo frente a situações complexas.
- Apresenta dificuldades em relacionar-se com os clientes e demora para tomar atitudes em situações complexas.
- Não se relaciona com os clientes e não toma atitudes em situações complexas.

Conhecimentos
 Avalie, em relação aos conhecimentos teóricos necessários para desempenhar satisfatoriamente as atividades programadas.

- Tem os conhecimentos necessários.

free for personal use

Figura 36 - Avaliação - Terceira Etapa

A lista das avaliações é composta pelo nome dos funcionários cadastrados no sistema e o status em que o usuário conseguirá visualizar qual colaborador/administrador ele já avaliou, conforme a figura 37 ilustra.

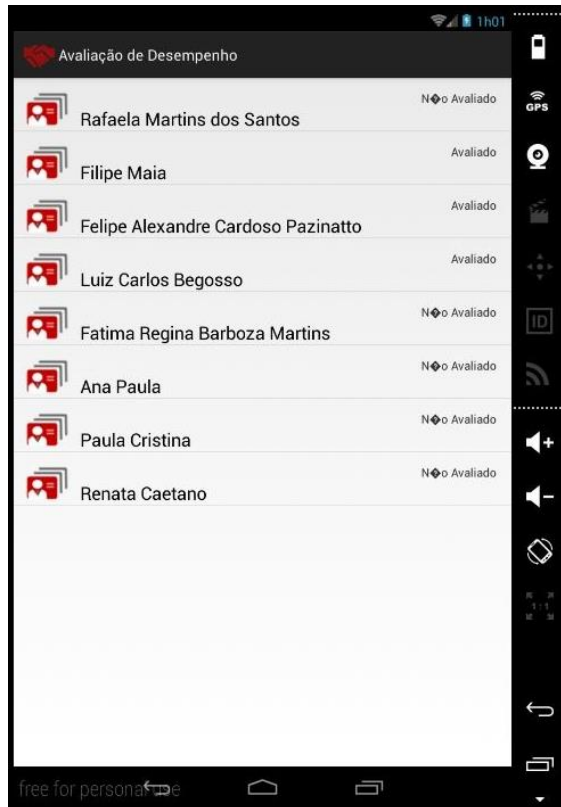


Figura 37 - Lista Avaliação

APÊNDICE C – INTERFACE WEB – METAS

A figura 38 é responsável por demonstrar a interface da lista de metas, que são as metas que foram digitadas no dispositivo móvel. Essa tela tem a opção de alterar o status da meta através de dois botões representados pela cor verde ou vermelho, que significam que a meta foi realizada ou então, não realizada.





















AVALIAÇÃO DE DESEMPENHO							
Avaliação				Meta			
LISTA - META							
IdMeta	Meta	Descrição	Status	Código Funcionário	Funcionário	Critério	Alterar Status
1	Aumentar o desempenho dos funcionarios	Meses: Abril e Setembro	2	1	Rafaela Martins dos Santos	100	 
2	Marcar reuniões	Todas as terças feiras	1	1	Rafaela Martins dos Santos	50	 
3	Vender mais	vender mais nos meses de abril e novembro	2	2	Filipe Maia	100	 
4	Fazer mais programas Web	Fazer mais programas Web esse mes	2	4	Luiz Carlos Begosso	10	 
5	Analisar programas	Analisar mais programas essa semana	1	3	Felipe Alexandre Cardoso Pazinato	10	 
6	atender mais clientes esse mes	atender mais clientes esse mes	1	1	Rafaela Martins dos Santos	20	 
7	Realizar treinamentos	Realizar treinamentos esse mês.	0	1	Rafaela Martins dos Santos	10	 
8	Realizar reuniões	Realizar reuniões esse mês.	1	1	Rafaela Martins dos Santos	10	 
9	Teste	Teste	0	1	Rafaela Martins dos Santos	1	 
10	Teste	Teste2	0	1	Rafaela Martins dos Santos	1	 

Figura 38 - Interface Web - Metas

APÊNDICE D – INTERFACE WEB – AVALIAÇÕES

Esse item é responsável por demonstrar a interface da lista de avaliações, que são as avaliações que foram digitadas no dispositivo móvel pelos administradores/colaboradores, conforme a figura 39 ilustra.

AVALIAÇÃO DE DESEMPENHO														
				Avaliação					Meta					
LISTA - AVALIAÇÃO														
IdAvaliacao	Código Funcionário	Funcionário	Código Usuário	Usuário	Resposta Um	Resposta Dois	Resposta Três	Resposta Quatro	Resposta Cinco	Resposta Seis	Resposta Sete	Resposta Oito	Resposta Nove	Resposta Dez
1	2	Filipe Maia	1	Rafaela Martins dos Santos	100	100	100	100	25	25	75	100	100	100
2	4	Lulz Carlos Begosso	1	Rafaela Martins dos Santos	100	75	50	50	75	100	100	50	50	100
3	3	Felipe Alexandre Cardoso Pazinato	5	Fatima Regina Barboza Martins	100	100	100	100	50	50	100	100	75	100
4	3	Felipe Alexandre Cardoso Pazinato	5	Fatima Regina Barboza Martins	25	50	100	100	50	100	100	25	100	100
5	3	Felipe Alexandre Cardoso Pazinato	1	Rafaela Martins dos Santos	75	75	25	100	75	75	100	50	25	100

Figura 39 - Interface Web - Avaliações

ANEXO A – FORMULÁRIO DA AVALIAÇÃO DE DESEMPENHO

Nesse item é representado o formulário para a avaliação de desempenho do SENAI.



REGULAMENTO DE ESTÁGIO

Escola SENAI “Roberto Mange”
CFP 5.01 - Campinas

FORMULÁRIO PARA AVALIAÇÃO DE DESEMPENHO DO ESTAGIÁRIO / FUNCIONÁRIO - EMPRESA

Questionário preenchido na empresa preferencialmente com o estagiário pelo seu coordenador para que seja registrado o desempenho do aluno / funcionário.

Empresa:	Data: / / 200
Estagiário:	
Avaliador (coordenador do estagiário na empresa):	
Função / Setor:	/

AVALIAÇÃO FEITA PELO COORDENADOR DO ESTAGIÁRIO NA EMPRESA

1. **CAPACIDADE DE APRENDER** – Considere a utilização e informações técnicas e operacionais nas situações práticas sugeridas.

- Possui alto grau de assimilação, não requer auxílio.
- Aprende rápido, relembra e associa a aprendizados anteriores. Necessita de pouca supervisão.
- Aprende apenas quando recebe instruções detalhadas e acompanhamento no desenvolvimento das tarefas.
- Aprende devagar, esquece facilmente dos assuntos estudados e necessita de supervisão contínua.

2. **INICIATIVA** - Considere a disposição para procurar a solução de problemas e a proposição de idéias.

- Procura solução de problemas e propõe idéias espontaneamente.
- Procura solução de problemas e só propõe idéias quando solicitado.
- Não procura soluções e não propõe idéias.
- Para solucionar problemas, precisa de orientação e segue idéias existentes.

3. **INTERESSE** - Observe a vontade e o esforço em aprender e a desempenhar as atividades programadas.

- Faz estritamente o necessário, aprendendo apenas as atividades obrigatórias.

- Interessa-se em aprender e desempenhar atividades e aceita outras adicionais.
- Tem especial interesse em aprender e desempenhar as atividades e sempre solicita outras.
- Não tem interesse em aprender e não demonstra satisfação pelas atividades.

4. **RESPONSABILIDADE** - Considere a forma como desempenha suas tarefas e a confiança que inspira quando lhe são atribuídas.

- Não precisa ser lembrado sobre as tarefas que deve realizar, demonstrando ser merecedor de confiança.
- Não consegue assumir as tarefas e não responde por elas.
- Pode-se contar com ele desde que se sinta controlado.
- É de inteira confiança, sempre preocupado com as tarefas atribuídas.

5. **PARTICIPAÇÃO** - Considere a disposição para colaborar e disponibilidade para atender os colegas e a supervisão.

- Não gosta de colaborar. Demonstra má vontade quando solicitado.
- Colabora e atende ativa e espontaneamente.
- Colabora e atende sempre que solicitado.
- Colabora quando solicitado, mas algumas vezes não o faz de boa vontade.

6. **RELACIONAMENTO HUMANO** - Considere a maneira como se relaciona com os colegas, supervisores e a integração com a equipe de trabalho.

- Relaciona-se bem, geralmente integra-se com a equipe.
- Apresenta dificuldades em relacionar-se. Demora a integrar-se a equipe.
- Mantém ótimas relações com o pessoal; integra-se com facilidade a equipe.
- Apresenta dificuldades em se relacionar; não se integra a equipe.

7. **QUALIDADE DE TRABALHO** - Considere a exatidão buscada e apresentada nos trabalhos realizados.

- Trabalho de excelente qualidade.
- Número aceitável de erros, face ao volume de trabalho realizado.
- Necessita de uma inspeção mais acentuada nas suas tarefas.
- Trabalho de qualidade insatisfatória.

8. **CONHECIMENTOS** - Avalie, em relação aos conhecimentos teóricos necessários para desempenhar satisfatoriamente as atividades programadas.

- Tem os conhecimentos necessários.

- Tem conhecimentos razoáveis.
- Não tem conhecimentos suficientes.

9. **ASSIDUIDADE** - Observe o cumprimento dos horários preestabelecidos, considerando: atrasos, faltas e solicitações de saída.

- Frequentemente atrasa, falta e/ou solicita saídas antecipadas. Não justifica.
- Não atrasa, não falta e não solicita saídas antecipadas.
- Algumas vezes atrasa, falta ou solicita saídas antecipadas. Justifica.
- Algumas vezes atrasa, falta ou solicita saídas antecipadas. Não justifica.

10. **SEGURANÇA** - Considere a disposição em cumprir os hábitos e normas de segurança.

- Observa e cumpre todas as normas.
- Quando supervisionado, segue as normas.
- Necessita frequentemente ser alertado sobre as normas.
- Não observa, na maioria das vezes, as normas de segurança.

11. Que tipo de conhecimentos / assuntos poderiam ser melhor explorados no curso?

12. Em quais cursos oferecidos pela Escola a empresa teria interesse em contratar estagiários?

- | | |
|---|--|
| <input type="checkbox"/> Técnico em Construção de Ferramentas | <input type="checkbox"/> Técnico em Eletroeletrônica |
| <input type="checkbox"/> Técnico em Mecatrônica | <input type="checkbox"/> OUTRO _____ |

Avaliador: _____

Data: ___/___/___

Carimbo

A nota final da Avaliação de Desempenho do aluno na Empresa será dada pela equação:

$$\sum \text{Pontos} \times 2,5$$

Será atribuída uma nota ao estagiário, obedecendo a pontuação da tabela a seguir:

1. Capacidade de aprender	Pontos	6. Relacionamento Humano	Pontos
Primeiro item	4	Primeiro item	3
Segundo item	3	Segundo item	2
Terceiro item	2	Terceiro item	4
Quarto item	1	Quarto item	1

2. Inciativa	Pontos	7. Qualidade de trabalho	Pontos
Primeiro item	4	Primeiro item	4
Segundo item	3	Segundo item	3
Terceiro item	1	Terceiro item	2
Quarto item	2	Quarto item	1

3. Interesse	Pontos	8. Conhecimentos	Pontos
Primeiro item	2	Primeiro item	4
Segundo item	3	Segundo item	3
Terceiro item	4	Terceiro item	2
Quarto item	1		

4. Responsabilidade	Pontos	9. Assiduidade	Pontos
Primeiro item	3	Primeiro item	1
Segundo item	1	Segundo item	4
Terceiro item	2	Terceiro item	3
Quarto item	4	Quarto item	2

5. Participação	Pontos	10. Segurança	Pontos
Primeiro item	1	Primeiro item	4

Segundo item	4	Segundo item	3
Terceiro item	3	Terceiro item	2
Quarto item	2	Quarto item	1