Sistema de Coleta e Gerenciamento de Informações via SNMP: Interface Web de Gestão Integrada

André Luís Rodrigues Moura, Larissa Pavarini da Luz

Curso de Tecnologia em Análise e Desenvolvimento de Sistemas – Faculdade de Tecnologia Dep. Júlio Julinho Marcondes de Moura - FATEC Garça   
Garça – SP– Brasil

andre@centermaq.com, larissa.luz01@fatec.sp.gov.br

**Abstract**

**Introduction**: The recent expansion of Brazilian companies, leveraged by the favorable economic moment that the country has gone through, has fostered a follow-up of service rendering, known as Outsourcing or simply Outsourcing. Delivering productivity tools that help companies achieve better levels of efficiency, cost rationalization is a challenge to be achieved.

**Objective**: To deliver an integrated software tool called Printerconnect with hardware information collection functions (counters, supply levels among others), features for management of attendances and equipment.

**Methodology**: It consists of three parts, the first one is a bibliographic survey aiming theoretical basis, second part the modeling and finally the development of a modular software contemplated a webservice that will be the nucleus of the application and will receive data and provide services for all the rest of the application. A module for mobile devices that will be responsible for providing mobility for technicians and integrating GPS (Global Positioning System) positioning functionality.

**Results**: After all the application has been submitted to real tests of use, it can be affirmed that the expected results, so, the necessary functional requirements, whether in the matter of hardware information collection or in the matter of the service management are shown Efficient management that qualifies to be used in a production environment by managing service provider companies with regard to service management and management information collection.

**Conclusions**: After analyzing the results obtained, it is correct to conclude that the system has achieved its objectives both in terms of its interaction with the user or the functionality that he is able to provide.

**Keywords**: Outsourcing, Outsourcing, Collection of Accountants.

**RESUMO**

**Introdução:** A recente expansão de empresas brasileiras, alavancada pelo momento econômico favorável que o país tem atravessado, tem fomentado um seguimento da prestação de serviço, conhecido como *Outsourcing* ou simplesmente Terceirização. Entregar ferramentas de produtividade que auxiliam as empresas a alcançarem melhores níveis de eficiência, racionalização dos custos é um desafio a ser alcançado.

**Objetivo:** Entregar uma ferramenta integrada de software denominada *Printerconnect* com funcionalidades de coleta de informações de *hardware* (contadores, níveis de suprimentos entre outros), funcionalidades para gerência de atendimentos e equipamentos.

**Metodologia:** Constituída em três partes, sendo a primeira um levantamento bibliográfico visando embasamento teórico, segunda parte a modelagem e finalmente o desenvolvimento de um software modular contemplado um *webservice* que será o núcleo da aplicação e receberá dados e fornecerá serviços para todo o restante da aplicação, um módulo para dispositivos móveis que será responsável por fornecer a mobilidade para os técnicos e integrar funcionalidades de posicionamento GPS (Sistema de Posicionamento Global).

**Resultados:** Após toda a aplicação ter sido submetida a testes reais de uso, pode-se afirmar que os resultados esperados, ou seja, os requisitos funcionais necessários, seja na questão de coletas de informações de hardware ou seja ainda na questão da gestão de atendimento se mostram eficientes o que qualifica a ser usada em ambiente de produção gerenciando empresas provedoras de serviços no que se refere a gestão de atendimento e coleta de informações gerenciais.

**Conclusões:** Após análise dos resultados obtidos é correto concluir que o sistema atingiu seus objetivos tanto no que tange a sua interação com o usuário ou as funcionalidades que ele é capaz de fornecer.

**Palavras Chave**: Terceirização, *Outsourcing*, Coleta de Contadores.

# **1. INTRODUÇÃO**

Terceirização não é algo recente, segundo Bergamaschi (2004) isso vem desde o século XVIII, onde ingleses e franceses terceirizavam algumas de suas atividades especializadas, entre as empresas que partem para o caminho da terceirização diversos são os motivos ou as necessidades que as levam a terceirizarem parte de suas atividades. Segundo Gorla (2011), terceirização é a pratica de transferir atividades comerciais das empresas para fornecedores especializados. Ainda segundo Bergamaschi (2004), geralmente atividades especificas que nada tem a ver com os objetivos comerciais das empresas são passadas para empresas ou pessoas que possuam melhor experiência em executá-las.

Em um contexto geral a produção de documentos pelas empresas, ou seja, a utilização do documento impresso nada tem a ver com sua atividade fim, com seu objetivo comercial, trata-se somente de formalidades, questões jurídicas, fiscais ou ainda questões culturais ainda presentes nas empresas.

Dada a crescente adoção de tecnologia por parte das empresas a tendência mais lógica seria uma diminuição na necessidade do papel impresso. Segundo Turban *et. al* (2004), a necessidade de se atingir novos níveis de eficiência e agilidade exigida pelas empresas pode colaborar para que os indivíduos convertam suas informações impressas em informações digitais. Entretanto, diversos obstáculos principalmente no que se refere a burocracia legal faz com que o uso do papel impresso mesmo que em menor número ainda seja necessário nas empresas.

Para que uma empresa consiga produzir seus documentos impressos uma série de recursos tecnológicos, financeiros e humanos são necessários, isso exige esforços de gestão e que muitas vezes não é o foco das empresas. De uma maneira geral os setores de Tecnologia da Informação das empresas são quem geralmente são os responsáveis por manter estes serviços disponíveis, entretanto os recursos humanos destes departamentos que são de maneira geral altamente especializados e caros ficam em muitas vezes empenhados em uma atividade que não pertence em teoria as suas atribuições.

Com o aumento da competitividade das empresas, os departamentos de Tecnologia da Informação das empresas passaram a ter uma visão mais estratégica nas empresas, e para que elas possam entregar o máximo de serviços que levem as organizações a atingirem maiores níveis de eficiência e consequentemente aumento de sua capacidade competitiva é necessário que serviços como gestão de impressoras, suprimentos, *scanners* possamter suas responsabilidades transferidas para fornecedores especializados deixando os setores de TI livres para funções estratégicas nas empresas.

Partindo desta nova necessidade de mercado que surgem as empresas especializadas em entregar serviços de impressão terceirizada. Este tipo de serviço vai da simples locação de equipamentos a chamada terceirização de equipamentos ou o que é chamado de *Outsourcing* de Impressão ou BPO (*Business Process Outsourcing*), que é quando a empresa passa toda a responsabilidade pela produção e gerenciamento da produção e controle de documentos a cargo da empresa contratada, nesta modalidade pode estar incluso desde a locação dos equipamentos, gestão de produção até o emprego de funcionários para gerenciar a operação.

Para Napoleão Filho *et al.* (2013), o outsourcing de impressão consiste na locação e gerenciamento da produção de cópias e impressão com objetivo principal de reduzir o trabalho e gerenciar os custos através de tarifas por página. Normalmente, o cliente fica livre da responsabilidade da depreciação dos equipamentos e investimentos de insumos, deixando estes sob responsabilidade da contratada, pagando, por isso, somente as páginas copiadas ou impressas.

É correto afirmar que a modalidade de *outsourcing* no tocante a produção de documentos é muito vantajosa para as empresas que contratam e também para as que fornecerem este tipo de serviço, entretanto, para as empresas que fornecem este tipo de serviço especializado herdam todos os problemas que antes eram de responsabilidade da contratante, gerenciar insumos, depreciação de hardware, gestão da manutenção dos equipamentos, promover uma diminuição do custo da empresa contratante sem abrir não do lucro necessário para manutenção do seu negócio.

Controlar o envio de suprimentos, controlar as manutenções além de buscar uma forma de colher os contadores (numeradores) de páginas necessários para o faturamento são preocupações que todas as empresas deste seguimento possuem. Como promover uma redução de custos para quem contrata sem abrir mão do lucro de quem é contratado, este é o problema que este projeto visa resolver.

**2. Objetivo**

Construir uma solução de software modular denominada Printerconnet capaz de entregar às empresas do setor *outsourcing* de impressão, ferramentas de gestão de atendimento, coleta de contadores de páginas, níveis de suprimentos e alertas de erros dos equipamentos.

**3. Metodologia**

**3.1 Referencial Teórico**

O sucesso ou fracasso de um software está intrinsicamente ligado a metodologia, as tecnologias e ao processo de concepção em que ele é submetido. Segundo Pressman (2002), após a crise de software, inúmeras teorias foram elaboradas forçando com que o desenvolvimento de software seguisse metodologias mais profissionais, diversas tecnologias e métodos foram concebidos de forma a garantir melhores formas de acompanhamento e gerenciamento do processo de construção de software onde a qualidade é o ponto a ser alcançado.

Segundo Xispe (2004), todo o projeto de software deve obrigatoriamente começar pela análise de requisitos seguindo fielmente as necessidades do cliente. Despois da análise segue para a fase de modelagem de acordo com as necessidades do cliente e pôr fim a codificação e os testes.

Seguindo o pressuposto que diz que o desenvolvimento de software é algo mutável, onde os requisitos podem sofrer alterações durante o desenvolvimento do projeto, a adoção de metodologias de desenvolvimento que possibilitam o melhor acompanhamento do processo e que possam permitir que correções sejam possíveis é algo que deve ser adotado em qualquer que seja o tamanho do projeto. Ainda segundo Xispe (2004), tecnologias surgiram para circundar o problema de mudança de requisitos durante o curso do desenvolvimento de software, estas tecnologias são chamadas de Métodos Ágeis.

Segundo a definição de métodos ágeis, software como qualquer projeto, não pode ser visto como um escopo fechado livre de interferências ou de mudanças, ele deve ser desenvolvido por partes seguindo um ciclo definido onde as entregas surgem de acordo com que os requisitos são entendidos.

Neste contexto surge na década de 90, as chamadas metodologias ágeis para o desenvolvimento de software (Manifesto Ágil, 2010). Segundo Prikladnicki *et al.* (2010), em geral, as metodologias ágeis não trazem conceitos novos para o desenvolvimento de software. O que as diferencia de outras metodologias são o enfoque e os valores. A ideia das metodologias ágeis é priorizar o enfoque nas pessoas e não em processos. Além disso, uma característica das metodologias ágeis é que elas são adaptáveis ao invés de serem prescritivas. Com isso, elas se adaptam a novos fatores decorrentes do desenvolvimento do projeto, ao invés de procurar analisar previamente tudo o que pode acontecer no futuro.

**3.1.1 Metodologia Scrum**

Scrum é uma metodologia para planejamento e gestão de projetos de software, ela é fortemente fundamentada no conceito da ênfase maior estar nas pessoas e não nos processos que estão ligados ao processo de desenvolvimento de software. De acordo com Schwaber & Beedle (2002), o Scrum tem como objetivo definir um processo para projeto que seja focado nas pessoas e não em processos.

As funcionalidades a serem implementadas em um projeto onde é usada a metodologia Scrum são mantidas em uma lista que é conhecida como [*Product* *Backlog*](http://www.desenvolvimentoagil.com.br/scrum/product_backlog). No início de cada *Sprint*, faz-se um [*Sprint* *Planning Meeting*](http://www.desenvolvimentoagil.com.br/scrum/sprint_planning_meeting), ou seja, uma reunião de planejamento na qual o [*Product Owner*](http://www.desenvolvimentoagil.com.br/scrum/product_owner) prioriza os itens do [*Product Backlog*](http://www.desenvolvimentoagil.com.br/scrum/product_backlog) e a equipe seleciona as atividades que ela será capaz de implementar durante o *Sprint* que se inicia. As tarefas alocadas em um *Sprint* são transferidas do [*Product Backlog*](http://www.desenvolvimentoagil.com.br/scrum/product_backlog) para o [*Sprint Backlog*](http://www.desenvolvimentoagil.com.br/scrum/sprint_backlog)*.*

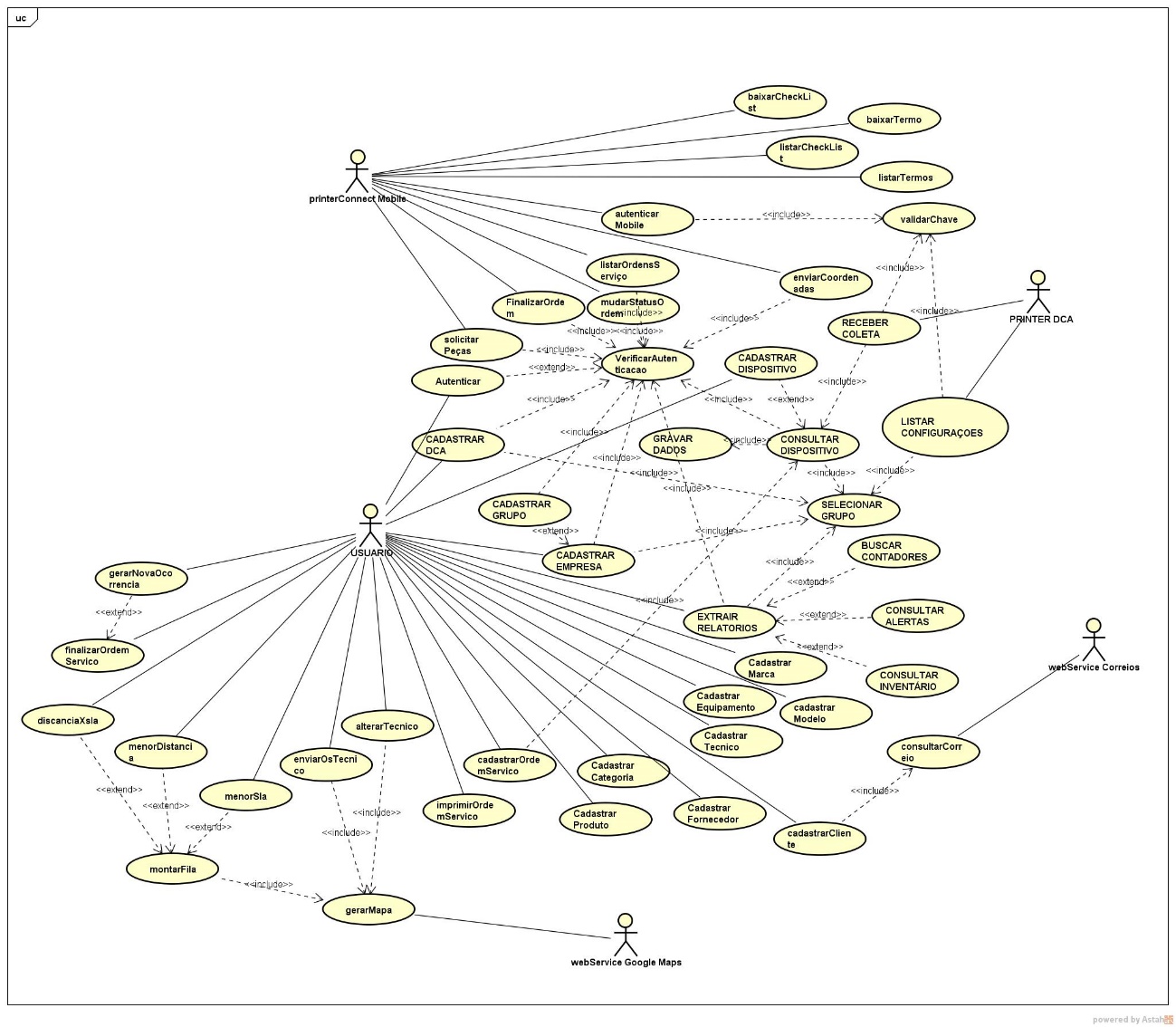
**3.2 Modelagem da Aplicação**

Após ter conhecer a fundo o problema a ser solucionado, entender todas as necessidades do clientes e transforma-las em requisitos funcionais da aplicação, um modelo lógico deve ser criado de forma a ilustrar como tudo deverá ser construído, e segundo as melhores práticas da engenharia de software o uso da UML (*Unified Model Language*) denota o uso do diagrama de caso de uso como sendo o mais indicado para este fim.

**3.2.1 Diagrama de Caso de Uso**

**3.2.1.1 Módulo Web**

O módulo web da aplicação é o que irá conter a maior parte da regra de negócio além de ser ela que irá gerenciar todos os dados envolvidos nos diversos módulos que a solução contemplará. Por isso a elaboração do diagrama de caso de uso, como apresentado na Figura 1, deste módulo consumiu maior parte de tempo e necessitou de maior atenção em sua concepção.

****

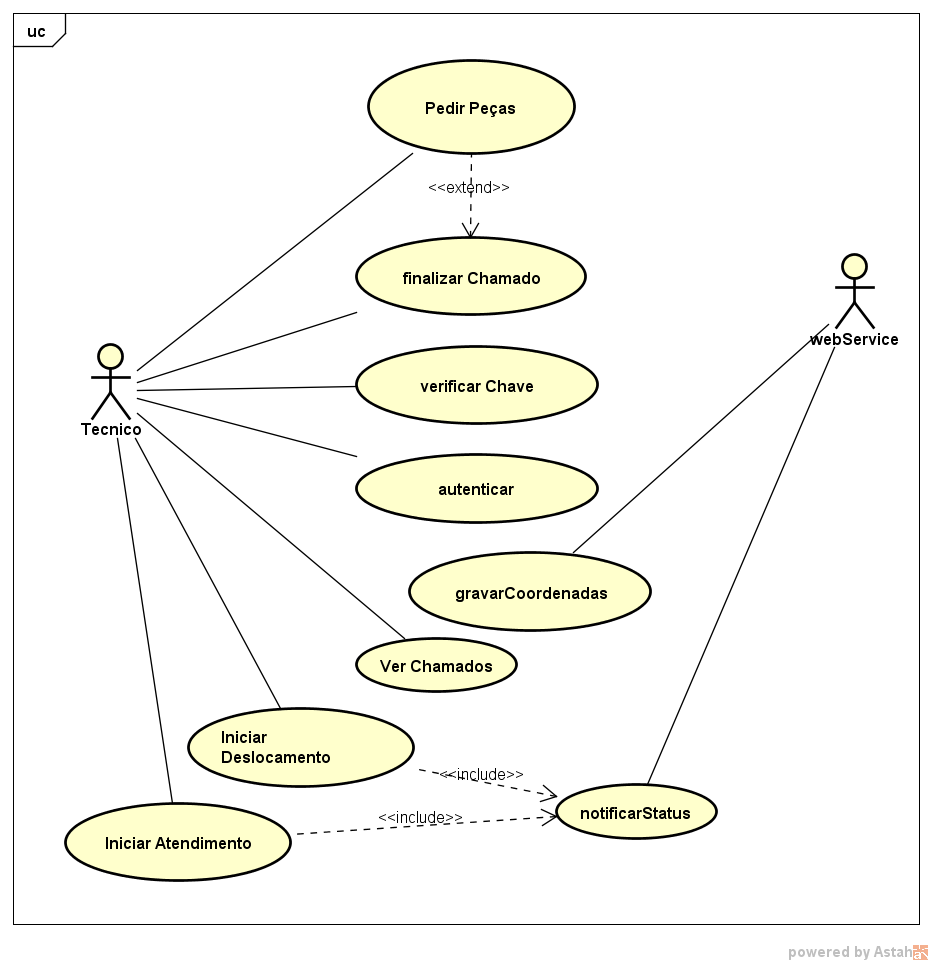
**Figura 1: Diagrama de Caso de Uso Módulo Web**

**Fonte: Própria do Autor.**

O objetivo deste diagrama é representar desde funções básicas do sistema como as entradas de dados através de formulários até a disponibilização de conexões externas através de webservice restFull. Ele tem como atores principais: Usuários, WebService dos Correios, Módulo DCA (*Automatic Data Colector*), Google Maps e Módulo Mobile.

**3.2.1.2 Módulo Mobile**

Tendo sua área de utilização limitada a questões de gestão de atendimento, instalações de equipamentos e captura de posições GPS, o módulo Mobile tem seu diagrama de caso de uso menos extenso, como apresentado na Figura 2, e de menor complexidade se comparado com o modulo web. Entretanto diversos recursos da aplicação dependem deste módulo como é o caso de visões de mapa e cálculo de distância.

****

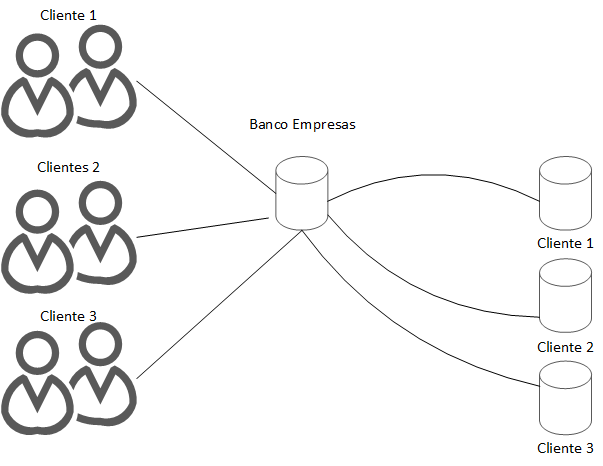
**Figura 2: Caso de Uso Módulo Mobile**

**Fonte: Própria do Autor.**

O objetivo deste diagrama é descrever as funcionalidades principais do módulo além de descrever os atores principais do contexto em que ele está inserido. Este módulo por questões de segurança dos dados ele não conta com bancos de dados internos, todas as informações trabalhadas pelo módulo são enviadas através do *webservice* e gravados no módulo principal. Atores: usuário e *webservice*.

**3.2.2 Modelo Relacional**

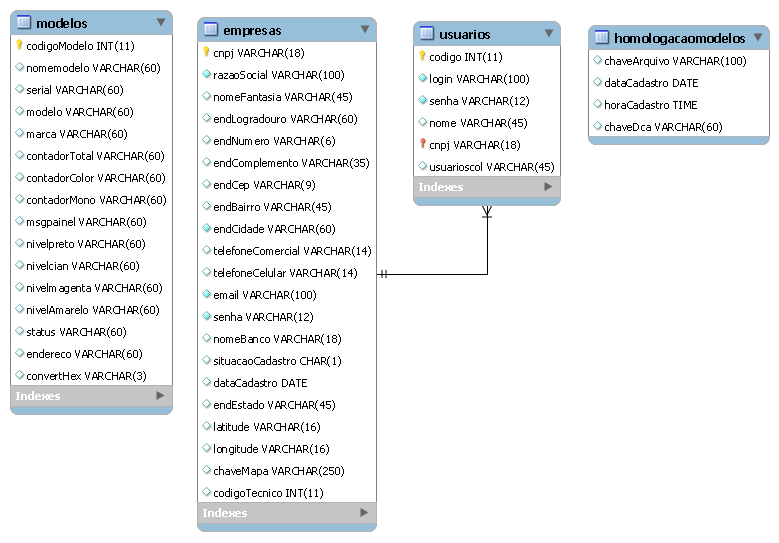
Usando o SGBD MySql como repositório de dados, por motivos de performance, portabilidade de plataforma entre outros pontos positivos, optou-se por seguir a seguinte logica funcional: O sistema contará com dois repositórios de dados distintos, um que será responsável pela autenticação das empresas, armazenamento das chaves de coleta *Object Identification* (OID) usados pelo módulo DCA e demais dados gerencias e de controle da aplicação, já o outro banco de dados será exclusivo para cada empresa, ou seja, cada empresa que se cadastrar para uso da ferramenta receberá um banco de dados exclusivo onde somente ela terá acesso, como apresentado na Figura 3.



**Figura 3: Esquema de Banco de Dados**

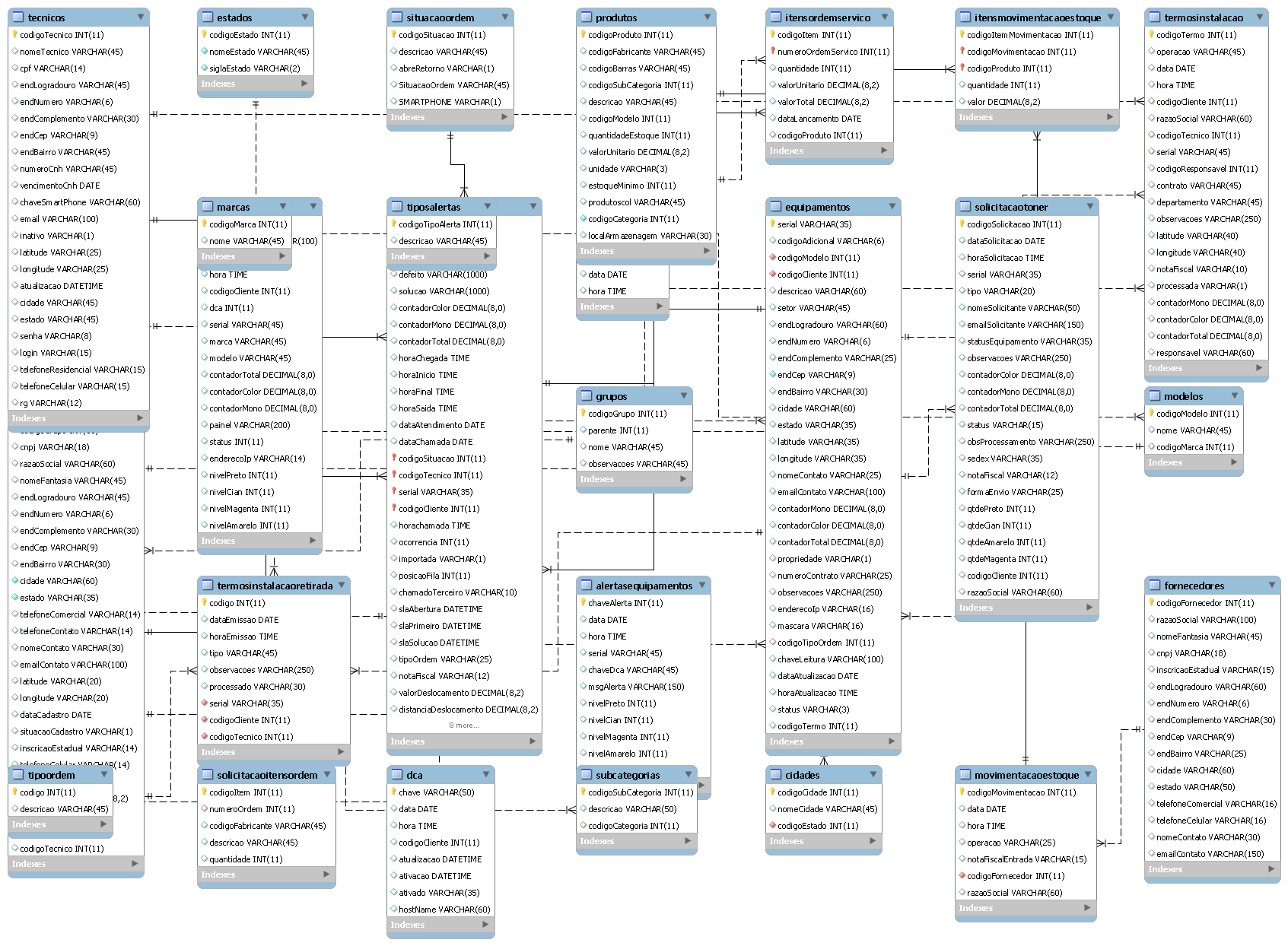
**Fonte: Própria do Autor.**

Este conceito adotado visa além da privacidade dos dados dos usuários, prover uma melhor utilização dos recursos, ou seja, uma vez contratado o serviço a empresa irá receber acesso a aplicação, quando o usuário faz a autenticação na *interface* automaticamente ele é direcionado a um repositório de dados único onde somente ele possui acesso, este repositório de dados seguira a seguinte nomenclatura: “DB + CNPJ da empresa” exemplo: DB0000000000.



**Figura 4: Modelo relacional banco empresas**

**Fonte: Própria do Autor.**



**Figura 5: Modelo Relacional banco privado**

**Fonte: Própria do Autor.**

**3.2.3 Frameworks**

Buscando uma maior agilidade no processo de desenvolvimento de software surgiram os *frameworks* de desenvolvimento, eles têm por finalidade tirar parte da carga de trabalho do desenvolvedor deixando a ele somente o que se refere as regras de negócio e a lógica necessária para a aplicação. Segundo Graciano Neto (2012), um Framework de Aplicação (FA) é uma aplicação semi-completa, construída como uma coleção organizada de componentes de software reusáveis para facilitar a implementação de aplicações de software customizadas.

Entretanto o uso excessivo de *frameworks* no desenvolvimento de aplicações deixa em muitos casos o desenvolvedor refém dos conceitos do *framework*, por isso buscou-se um *framework* que provesse a tecnologia e benefícios esperados deixando o programador livre para explorar recursos não implementados por ele. O framework CodeIgniter (<https://codeigniter.com/>) é um framework MVC (*model view controller*) e foi a solução escolhida por possuir uma curva de aprendizado relativamente rápida, ser robusto e flexível.

**3.2.4 API**

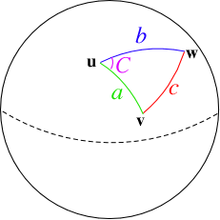
O recurso conhecido como API (*Application Programing Interface*) é uma técnica que permite que aplicações disponibilizem recursos para outros desenvolvedores de forma controlada.

Um exemplo de recursos disponibilizado através de API é a visão de mapas disponibilizado pela Google, através de códigos padronizados é possível criar aplicações complexas utilizando recursos de cálculo de distâncias, rotas entre outros.

Outra API utilizada neste projeto é a API responsável pela pesquisa de CEP nos correios, onde através do envio do CEP é retornado os dados de endereçamento, recurso muito utilizado por lojas virtuais calcularem valores de frete.

**3.2.5 FÓRMULA DE HAVERSINE**

Funções que calculem distâncias entre pontos usando coordenadas geográficas, é algo complexo que em alguns casos exigem muito recurso computacional, segundo Haversine (2011), uma alternativa é a fórmula de Haversine proposta por ele, baseada na ortodromia, que assume a Terra como esférica. Ela garante uma boa taxa de precisão e é menos complexa e computacionalmente menos dispendiosa.



**Figura 6: Visão gráfica de Haversine**

**Fonte: LONGETUDESTORE (2017).**

A equação de Haversine, Haversine (2011), pode ser verificada a seguir:

Onde:

D é a distância entre dois pontos situados na superfície de uma esfera;

R é o raio da esfera (em se tratando da Terra, corresponde a 6.378,137 km);

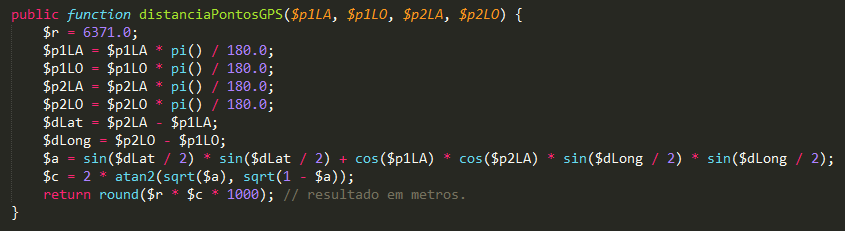
Lat1 é a latitude da origem em graus;

Lat2 é a latitude de destino em graus;

Lon1 é a longitude de origem em graus;

Lon2 é a longitude de destino em graus.

Este algoritmo não é preciso no contexto de criação de rotas uma vez que é considerado apenas distâncias de deslocamento como sendo em linha reta, não desconsiderando mudanças de direção durante o deslocamento. Porem como a funcionalidade somente exige distâncias aproximadas, tal implementação atendente todos os requisitos. A Figura 7 apresenta...

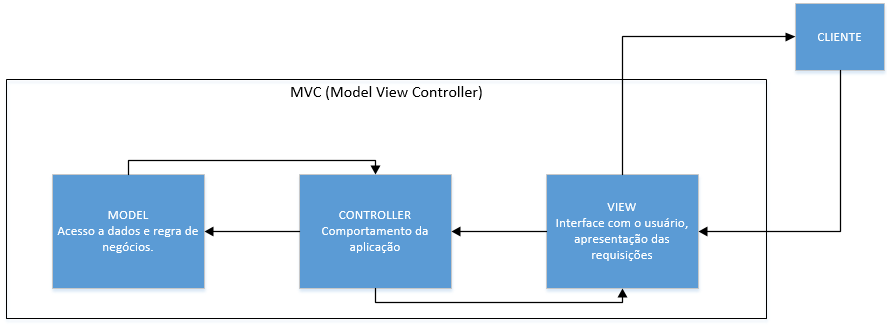
****

**Figura 7: Algoritmo em PHP para Cálculo de Distância (proximidade)**

**Fonte: Própria do Autor.**

**3.2.6 MODELO MVC**

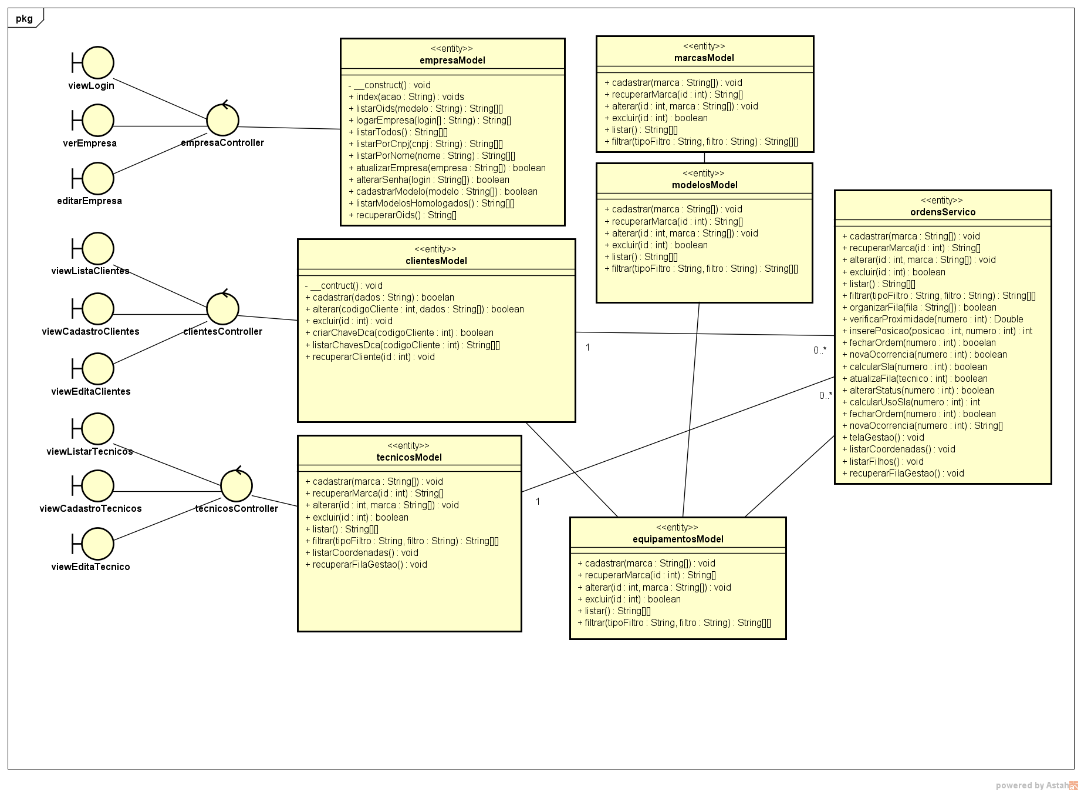
O principal motivo da adoção do modelo MVC (*Model View Controller*), é a organização de trechos de códigos em camadas de acordo com o seu contexto, ou seja, trechos de códigos responsáveis pela persistência de dados, não deveria estar implementado juntamente com a regra de negócio da mesma forma que não poderia estar acoplado com a visão do usuário. Segundo Fragmental (2006), a fragmentação em camadas é a chave para independência entre componentes, objetivando desta forma agrupar os componentes pelas suas responsabilidades em comum.



**Figura 8: Representação do Modelo MVC**

**Fonte: Própria do Autor.**

Já segundo Macarotti (2006), a fundamentação de dividir as funcionalidades de um sistema em camadas de acordo com sua relevância surgiu como alternativa para solucionar problemas onde códigos extremamente extensos e onde lógica de negócio e persistência de dados compartilhavam os mesmos arquivos o que dificulta a manutenção.



**Figura 9: Parte do Diagrama de Classes – Implementação MVC**

**Fonte: Própria do Autor.**

A Figura 9 demostra a implementação das classes *model* (modelo) utilizadas no contexto da ordem de serviço, pode-se observar que grande parte das regras de negócio estão implementadas nestes modelos, inclusive os cálculos de SLA, distâncias entre outros.

**3.2.7 Front End**

O sucesso ou fracasso de uma aplicação está diretamente ligada à como a interface com o usuário foi desenvolvida, buscar conceitos de usabilidade, acessibilidade e experiência do usuário é um assunto que tem ganhado espaço e sobretudo técnicas e conceitos para sua concepção. A IHC (Interface Humano Computador) é uma área da engenharia de software que estuda as boas práticas de desenvolvimento no que apresentação dos dados para os usuários.

Segundo Caetano (2012), a interface é um elemento imprescindível para a aceitação de um sistema por parte do usuário sua eficiência é influenciada pela Interface, existem sistemas onde os riscos fatais estão associados à sua interface.

Visando entregar aos usuários um ambiente interativo e visualmente agradável optou-se por utilizar um conjunto de dois *frameworks* de *layout*, o Bootstrap criado pela Twitter e o Materialize que é baseado no Material *Designer* da Google. Esta escolha se deu pelo fato do vasto número de aplicativos que utilizam estes frameworks e também para criar uma *interface* que assemelha as interfaces nativas da plataforma Android.

**4 Resultados**

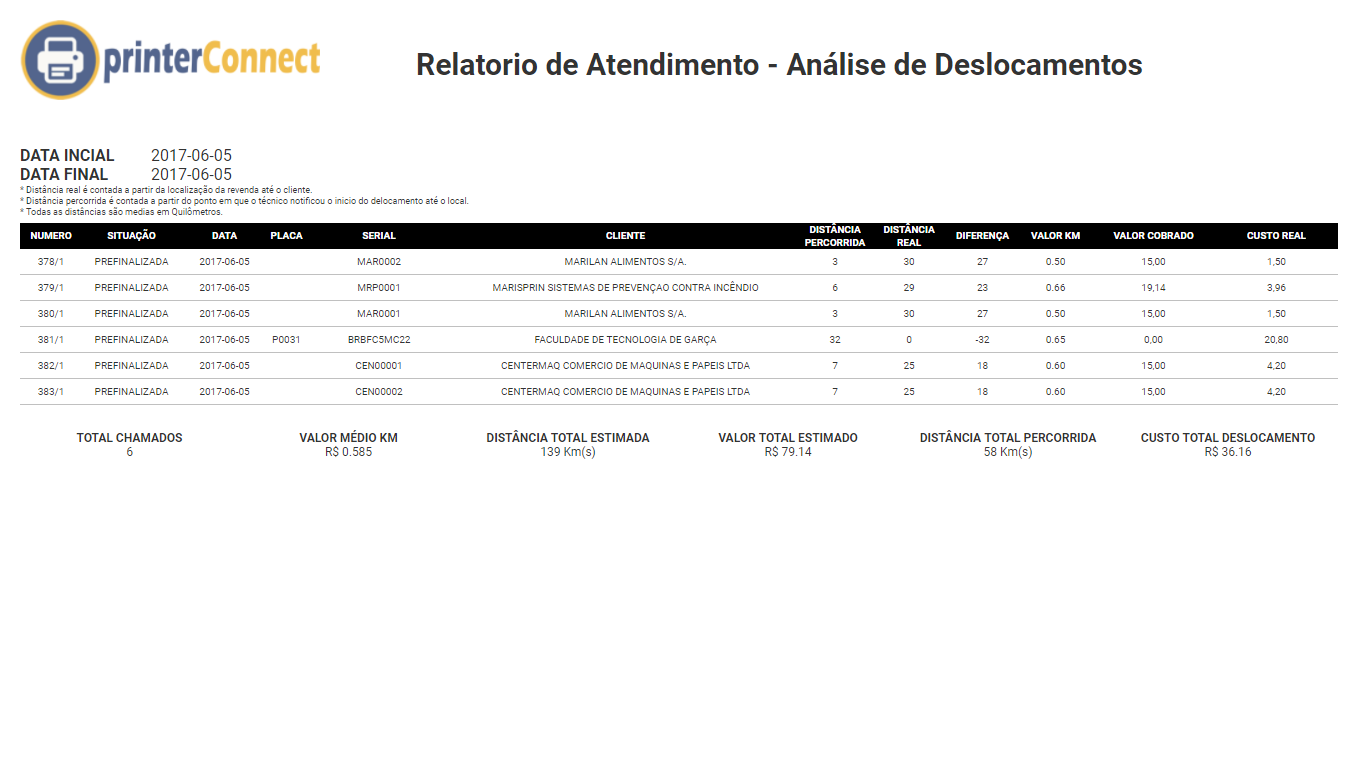
De forma geral a implementação dos módulos que compõe o sistema Printerconnect, após os testes realizados em ambiente real, mostrou satisfazer todos os requisitos funcionais propostos.

Com relação aos dados coletados pelo módulo DCA, nota-se que existe uma taxa de precisão de aproximadamente 98%, esta taxa se deve ao fato de que toda a homologação (parâmetros que devem ser buscados) ser feita de forma global e disponível para todas as licenças da aplicação, cada parâmetro de pesquisa (OID) foi retirado da documentação original do fabricante do equipamento o que garante a acurácia da informação.

Já no que diz respeito a gestão dos atendimentos é correto dizer que o módulo garante uma precisão média de 80 metros quando é solicitado ao sistema a visualização dos pontos de atendimento e a proximidade dos técnicos, esta precisão varia de acordo com a configuração do smartphone utilizado pelo módulo Mobile.

Ainda analisando os módulos de gestão de atendimento, podemos dizer que a funcionalidade de geração de filas (rotas) de atendimento atende as expectativas tendo uma taxa de precisão de 85% no que diz respeito a sugestão eficiente de rotas.

No tocante a relatórios, foi desenvolvido basicamente 3 relatórios para cada funcionalidade básica do sistema, onde estes mostraram bons índices de eficiência na visualização e na exatidão dos dados.



**Figura 10: Relatório de Análise de Deslocamentos**

**Fonte: Própria do Autor.**

**5 Conclusões**

Após análise dos resultados obtidos é correto concluir que o sistema atingiu seus objetivos tanto no que tange a sua interação com o usuário ou as funcionalidade que ele é capaz de fornecer.

As seguintes seções apresentam as telas e seu desenvolvimento, além de seu funcionamento.

**5.1 Interface Web**

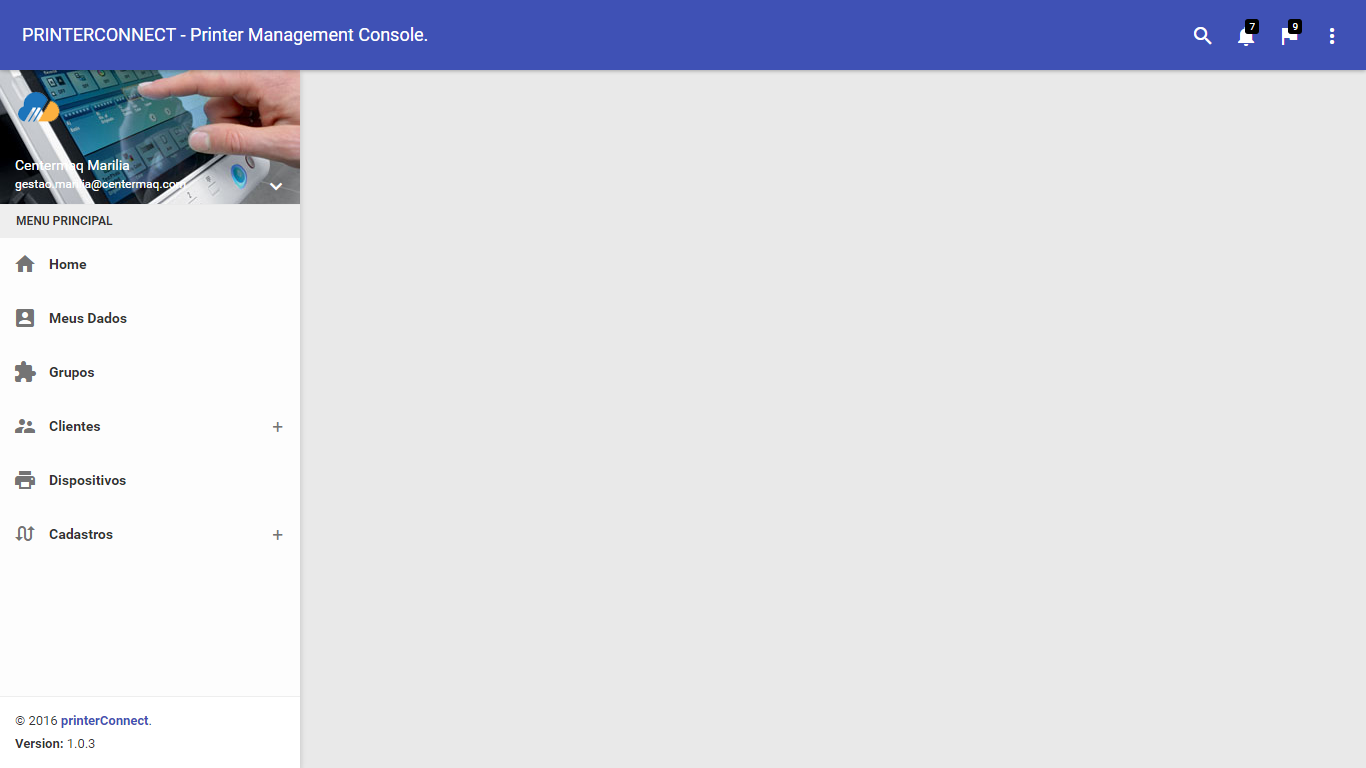


**Figura 11 – Tela inicial do sistema**

**Fonte : Imagem do Autor, 2017**

Todo o padrão de cores do sistema é baseado nas cores do logotipo que foi escolhido, as imagens que estão na interface de abertura do sistema fazem referência aos equipamentos que são gerenciados pelo sistema.

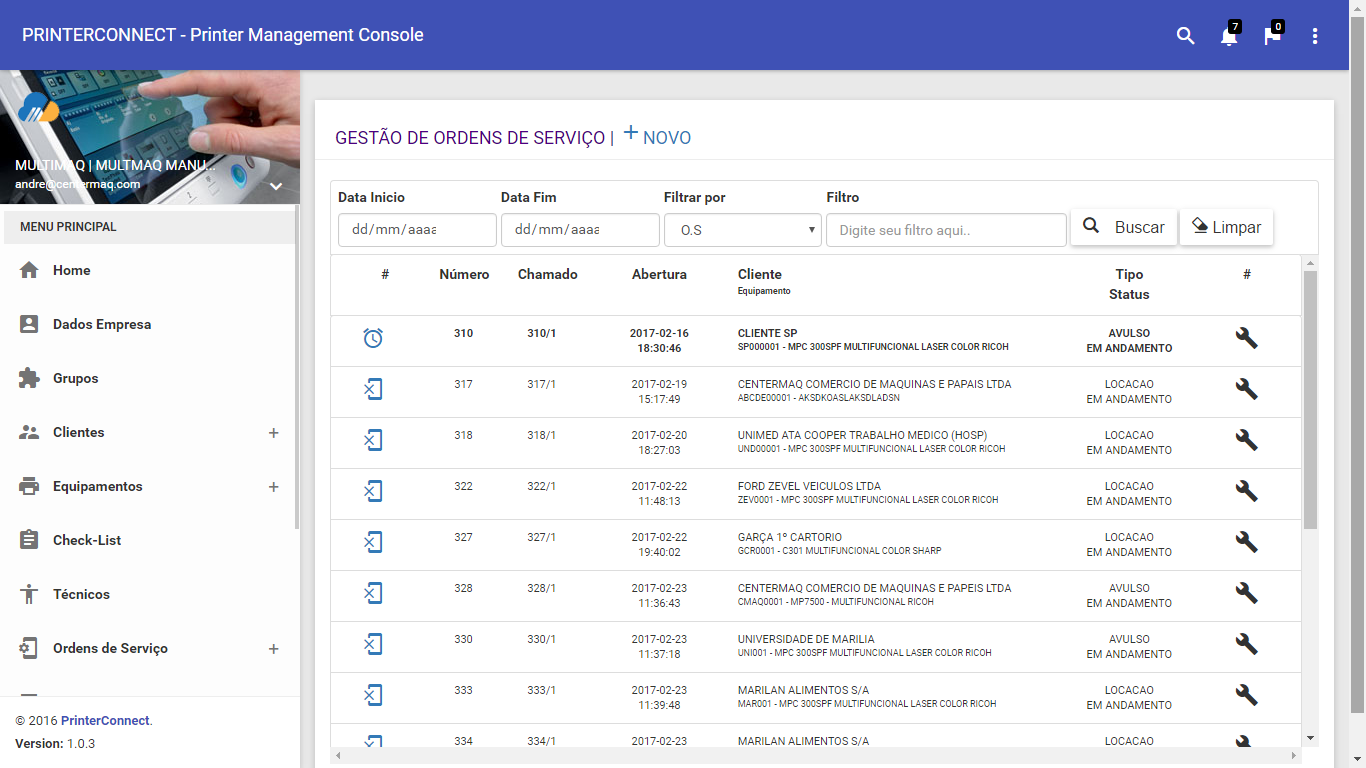
Para deixar a navegação mais intuitiva e agradável, o *login* de acesso ao sistema utiliza técnicas de comunicação assíncrona, através do *framework* *JQuery,* mesmo sendo uma aplicação voltada para a utilização em desktops, a interface foi desenhada utilizando o conceito de *Layout* Responsivo para garantir que caso a aplicação seja aberta em outras plataformas a navegação possa acontecer de forma agradável. A Figura 12 apresenta ....



**Figura 12: Tela Principal e Menu**

**Fonte : Imagem do Autor, 2017.**

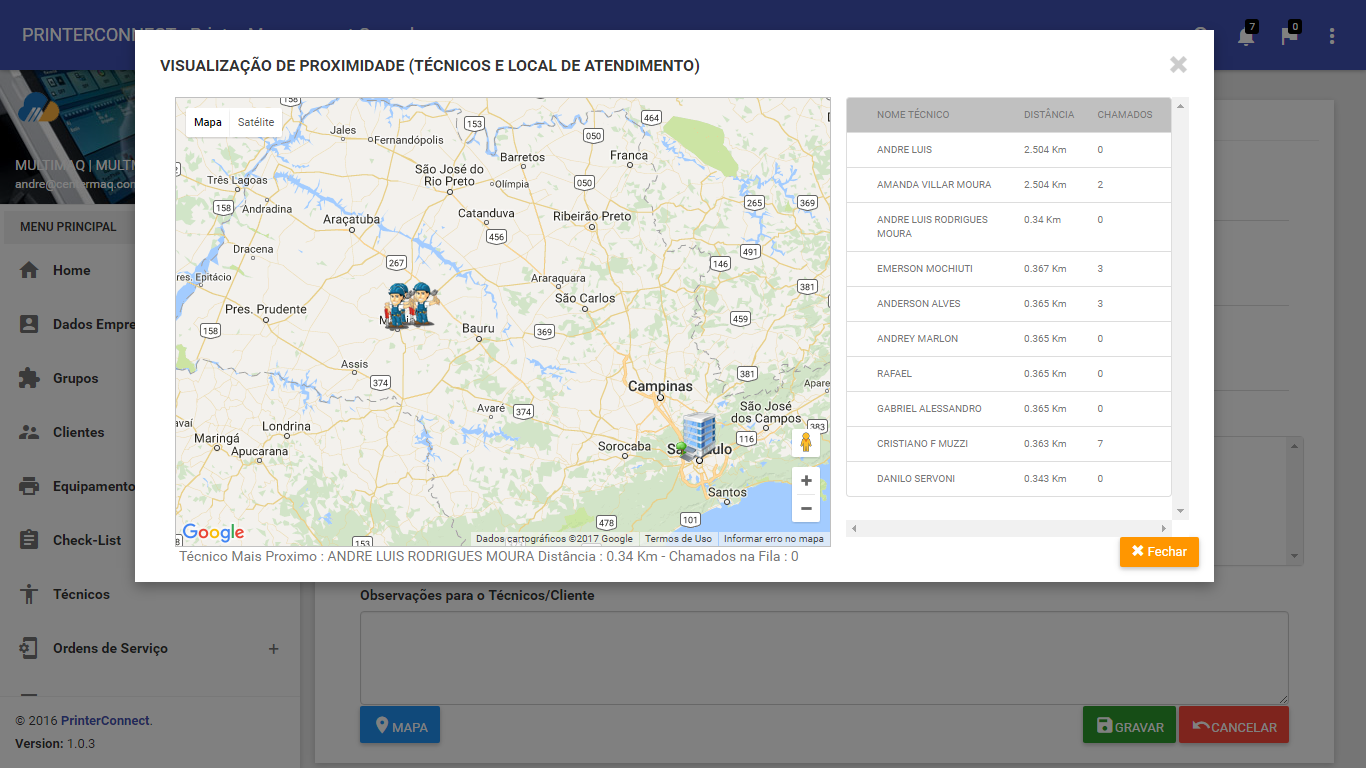
O sistema conta com um menu fixo na lateral esquerda, nenhuma janela ou formulário irá se sobrepor a ela, de forma que o acesso ao menu nunca seja impedido, como mostra a Figura 13



**Figura 13: Lista de Ordens de Serviço**

**Fonte : Imagem do Autor.**

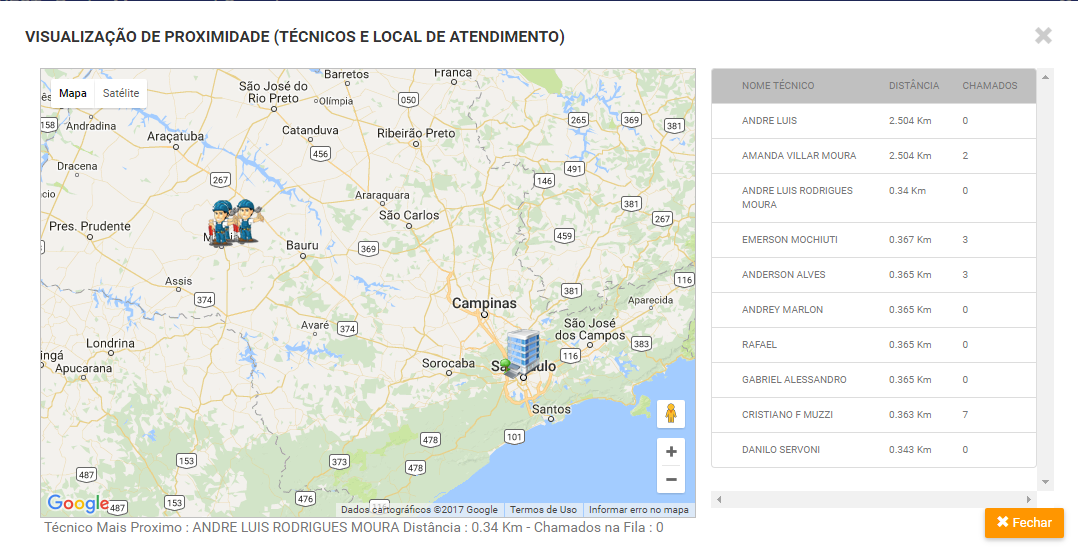
A tela de triagem de ordem de serviço (Figura 7), mostra uma lista com todas as ordens de serviço abertas (por padrão somente são mostradas as situações : Em Andamento ), do lado direito de cada Chamado aparece um ícone que mostra se ela já foi processada (enviada ao técnico) ou não, já do lado direito aparece um menu “drop-down” com opções como : Ver Ordem de Serviço, Imprimir Ordem de Serviço, Incluir Peças, Finalizar.



**Figura 14 : Triagem de Ordem de Serviço**

**Fonte : Imagem do Autor.**

Antes de mostrar a tela de triagem do chamado (figura 8), é mostrado um mapa constando a posição de cada técnico e o chamado aberto, uma lista com o nome, distância do técnico em relação ao cliente, e número de chamado que já estão na fila do técnico, além de uma dica de qual é o técnico mais próximo do local.



**Figura 15 : Atualizando dados da Empresa**

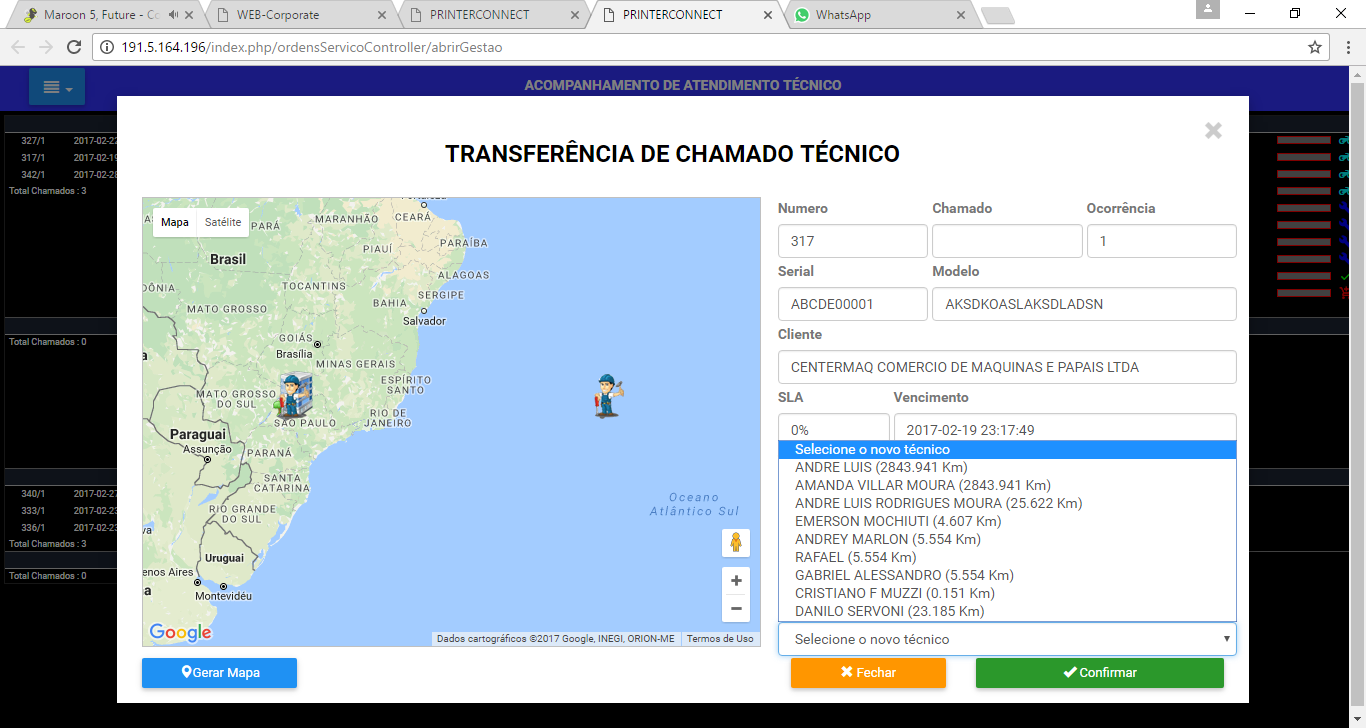
**Fonte : Imagem do Autor.**



**Figura 16 : Acompanhamento de Ordem de Serviço**

**Fonte : Imagem do Autor.**

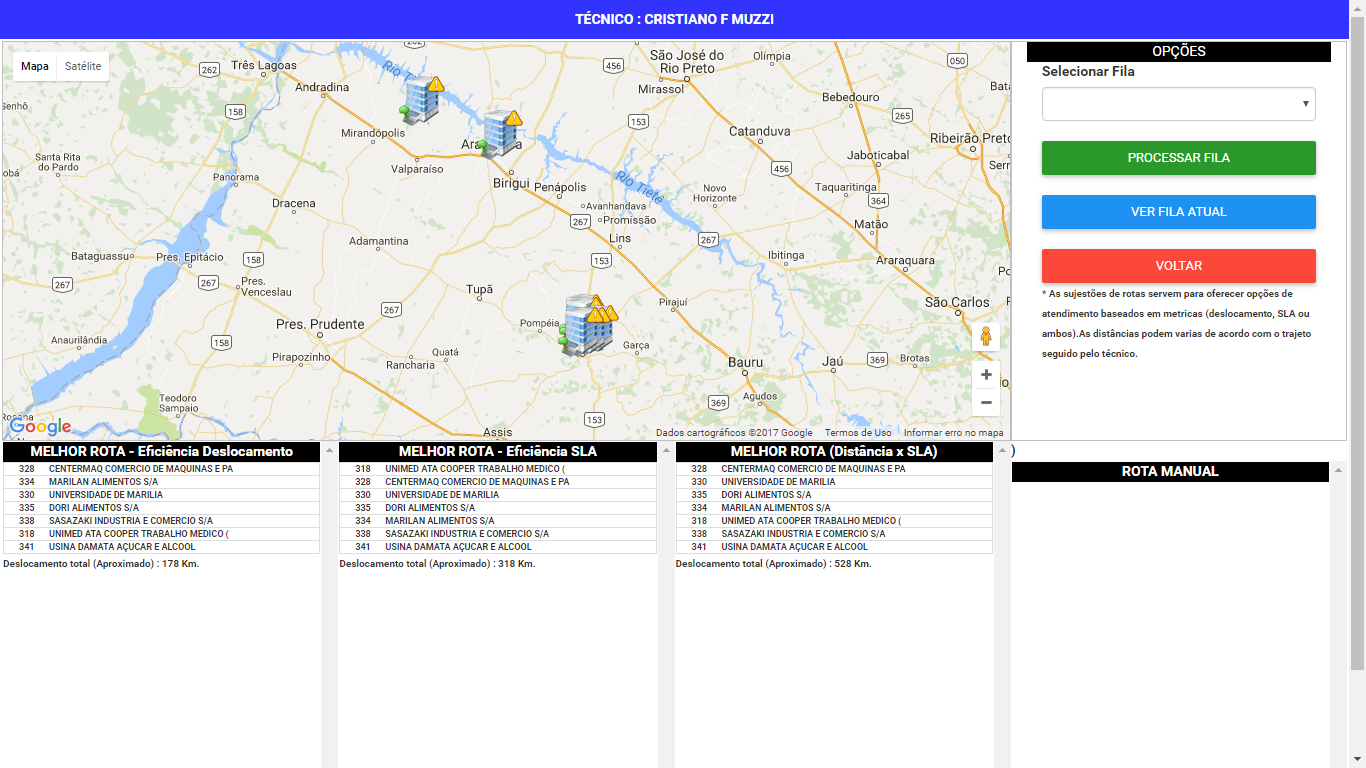
Esta tela é responsável por promover uma visão ampla e completa de todas as filas (técnicos), mostrando os tempos de SLA (*Service Level Agreement*) com uma barra de progresso recursiva, ícones e cores que mostram o status de cada chamado, ao clicar no ícone de um chamado que está em andamento é aberto uma tela para a transferência de chamado entre os técnicos conforme Figura 12.



**Figura 17 : Transferência de Chamado entre Técnicos**

**Fonte : Imagem do Autor.**

Ao transferir o chamado aparece os dados do chamado, vencimento do SLA, lista com técnicos disponíveis e suas distancias em relação ao Cliente.

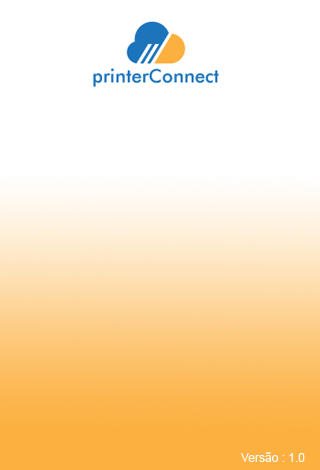


**Figura 18 : Gerando Opções de Fila**

**Fonte : Imagem do Autor.**

# **5.2 Módulo Mobile**

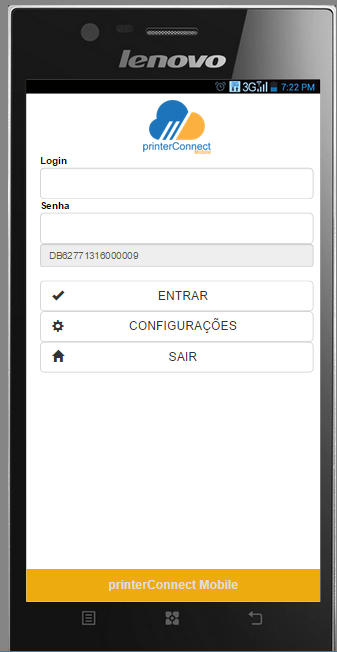
Concentrando as funcionalidades de GPS ao módulo mobile mostrou-se eficiente com uma taxa média de precisão de 80 metros quando o aparelho (smartphone) esteve configurado com precisão alta.

****

**Figura 19 : Tela de Splash**

**Fonte : Imagem do Autor.**

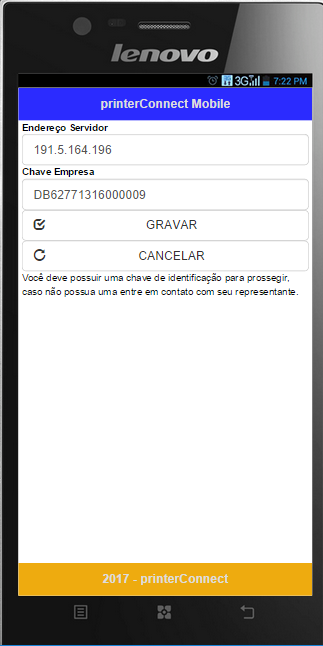
Bastante simples a tela de abertura (*Splash*), contém o logotipo sobre um degrade nas cores padrão do sistema, além da informação da versão atual.

****

**Figura 20 : Tela de Login**

**Fonte : Imagem do Autor.**

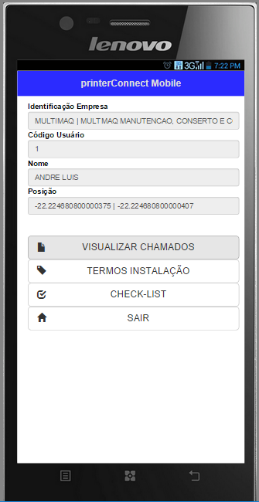
A tela de *login* do sistema tem um layout limpo, com campos de usuário, senha, chave de acesso e os botões : ENTRAR, CONFIGURAÇÕES e SAIR.

****

**Figura 21 : Atualizando dados da Empresa**

**Fonte : Imagem do Autor.**

A tela de configuração é onde o usuário irá informar o endereço do servidor printerConnect além de sua chave de acesso.

****

**Figura 22 : Menu de Opções**

**Fonte : Imagem do Autor.**

Após logar na interface o sistema irá mostrar uma tela com os dados do técnico (Codigo/Nome), coordenadas e botões para acesso as funções.

****

**Figura 23 : Lista de Chamados Técnicos**

**Fonte : Imagem do Autor.**

Clicando em alguma linha automaticamente é mostrado o formulário com os detalhes do chamado: Número, Intervenção, Data e Hora, Cliente, Equipamento, Defeito Relatado, Reclamante, Situação do equipamento. Também é mostrado uma barra de ferramentas com as possíveis ações do chamado: A Caminho, Atendendo, Endereço, Peças, Anotações, Finalizar, Voltar.

**5.3 Infraestrutura**

Um ponto a ser analisado é como o sistema se comporta quando todos os recursos oferecidos são utilizados, ou seja, quando todos os módulos estiverem em operação, qual será o comportamento do servidor

Para analisar tal questão a estrutura foi submetida a testes de desempenho que visam encontrar o ponto em que será necessário um aumento na capacidade do serviço para atender as demandas.

O sistema foi implantando em servidor virtualizado com sistema operacional Windows 2008 Server R2 dotado de um processador virtual de 2.4Ghz, com 4 Gb de memória RAM e duas placas Ethernet 10/100. Como infraestrutura de aplicação foi adotado o *Internet Information Service* (IIS7) da Microsoft por ser uma plataforma de fácil operação, contudo em ambiente de produção será migrado para plataforma Linux e rodando Apache com infraestrutura de aplicação.

Com base em medições e desempenho de resposta a requisições de serviços (*webservice Rest*), o sistema mostrou-se eficiente, os recursos do servidor da aplicação atingiram picos aceitáveis.

**Gráfico 1 – Medição dos Recursos do Servidor de Aplicação**

Ainda relacionado a medições de desempenho no que diz respeito a performance do Sistema Gerenciador de Banco de Dados (SGBD), nota-se que para uma consulta em tabelas com aproximadamente 64 mil registros o tempo de reposta foi inferior a 0.4 segundos o que indica uma performance dentro dos parâmetros esperados.

Com isso conclui-se que o sistema está apto a ser implantando em ambiente de produção entretanto é recomendado a execução de mais testes incluindo testes em provedores diferente com o objetivo de avalizar o comportamento nas mais diversas plataformas que o sistema pode ser implantado.

# **6. REFERÊNCIAS**

BERGAMASCHI, S. Modelos de gestão da terceirização de Tecnologia da Informação: um estudo exploratório. Tese (Doutorado em Administração) - Universidade de São Paulo, São Paulo, 2004.

GORLA, Narasimhaiah; CHIRAVURI, Ananth. 2011. Information systems outsourcing success: a review. In: 2010 International Conference on E-business, Management and Economics. 2011. p. 170-174.

TURBAN, E.; MCLEAN, E; WETHERBE, J. Tecnologia da Informação para Gestão: Transformando os Negócios na Economia Digital. Porto Alegre: Bookman, 2004.

NAPOLEÃO FILHO, Jair; ALVES, Roberto Carlos; PEREIRA, Fernando Lauro; BARRETO, Dalton; NUNES, Paula Martins; PACHECO, Andressa Sasaki Vasques. Eficiência na Administração Pública: o modelo de outsourcing de impressão na Universidade Federal de Santa Catarina. Universidade Federal de Santa Catarina. XIII Coloquio de Gestión Universitaria en Américas. Buenos Aires, Argentina, 27 a 29 de novembro de 2013.

Pressman, R. S., (2002) “Engenharia de Software”, 5ª Ed.,Makron Books.

Manifesto Ágil

<http://www.manifestoagil.com.br/> , Acessado em : 04 jun 2017.

PRIKLADNICKI; LIDDY; MAGALHÃES, <http://www.lbd.dcc.ufmg.br/colecoes/wamps/2010/006.pdf> , Acessado em : 04 jun 2017.

SCHWABER, K.; BEEDLE, M. Agile Software Development with SCRUM. Prentice Hall, 2002.

Valdemar Vicente Graciano Neto <http://www.inf.ufg.br/mestrado/sites/www.inf.ufg.br.mestrado/files/uploads/Dissertacoes/dissertacao\_valdemar\_.pdf> Acessado em : 04 Jun 2017.

MACORATTI, J. Carlos – Artigos de Tecnologia da Informação www.macoratti.net Acessado em 03 fev 2017.

FRAGMENTAL Tecnologia – Artigo MVC e Camadas, http://fragmental.com.br/index.php?title=MVC\_e\_Camadas Acessado em 04 fev 2017.

CAETANO, CINTIA. "Interação Humano-Computador Introdução."

< http://www2.ic.uff.br/~ccaetano/aulas/IHC\_Aula\_1\_Introducao.pdf >. Acessado em 05 jun 2017.

PADOVANI, Stephania, **Avaliação Ergonômica de Sistemas de Navegação em Hipertextos Fechados**. In: MORAES, Anamaria de. Design e Avaliação de Interface. Rio de Janeiro, IusEr, 2002. 27-58.

SHIMIDT, Kevin J, **SNMP ESSENCIAL**. Brasil: ELSEVIER EDITORA, 2001.

SOBRE ADMINISTRAÇÃO < <http://www.sobreadministracao.com/outsourcing-o-que-e-e-como-funciona/> >. Acessado em: 20 out 2016.

PUCRS <http://www.inf.pucrs.br/~gustavo/disciplinas/sd/material/Artigo\_WebServices\_Conceitual.pdf >. Acessado em: 20 out 2016.

UNIVERSITY OF CALIFORNIA <https://www.ics.uci.edu/~fielding/pubs/dissertation/fielding\_dissertation.pdf >. Acessado em: 20 out 2016.

POLISOFTWARE <<https://polisoftware.wordpress.com/2012/11/05/aplicacoes-web-conceitos-ii/>>. Acessado em : 10 set 2016.

BRASIL ESCOLA <<http://meuartigo.brasilescola.uol.com.br/informatica/programacao-orientada-objetos.htm>l>. Acessado em : 11 set 2016.

UNICAMP <http://www.dca.fee.unicamp.br/cursos/PooJava/objetos/conceito.html>. Acessado em: 12 set 2016.

GOOGLE DEVELOPERS < https://developers.google.com/?hl=pt-br>. Acessado em : 02 jan 2017.