

UNIVERSIDADE FEDERAL DE SANTA MARIA
CENTRO DE TECNOLOGIA
CURSO DE ENGENHARIA DE CONTROLE E AUTOMAÇÃO

Ana Elisa Marchesan Ramos

**DESENVOLVIMENTO DE SISTEMA COMPLETO PARA
AGENDAMENTO MÉDICO**

Santa Maria, RS
2016

Ana Elisa Marchesan Ramos

**DESENVOLVIMENTO DE SISTEMA COMPLETO PARA
AGENDAMENTO MÉDICO**

Trabalho de Conclusão de Curso
apresentado ao Curso de Graduação em
Engenharia de Controle e Automação, da
Universidade Federal de Santa Maria
(UFSM, RS), como requisito para obtenção
do grau de **Engenheira de Controle e
Automação**

Orientador: Prof. Dr. Eng. Frederico Menine Schaf

Santa Maria, RS
2016

Ana Elisa Marchesan Ramos

**DESENVOLVIMENTO DE SISTEMA COMPLETO PARA
AGENDAMENTO MÉDICO**

Trabalho de Conclusão de Curso
apresentado ao Curso de Graduação em
Engenharia de Controle e Automação, da
Universidade Federal de Santa Maria
(UFSM, RS), como requisito para obtenção
do grau de **Engenheira de Controle e
Automação**

Aprovado em 22 de julho de 2016:

Frederico Menine Schaf. Prof. Dr. Eng. (UFSM)
(Presidente/Orientador)

Sérgio Mergen. Prof. (UFSM)

José Eduardo Baggio. Prof. Dr. Eng. (UFSM)

Santa Maria, RS
2016

DEDICATÓRIA

*Dedico este trabalho à meus pais João Paulo e Nilda,
e ao meu namorado Gilberto.*

AGRADECIMENTOS

Primeiramente agradeço a Deus por tudo que fez por mim nesses cinco anos de graduação, agradeço principalmente por ter me capacitado para fazer este trabalho e que apesar de todos os obstáculos me manteve de pé até aqui.

Em segundo lugar agradeço meus pais por sempre me apoiarem e serem fundamentais para eu ter chegado aonde cheguei. Jamais conseguirei expressar o quanto sou grata por ser filha deles, mas aqui agradeço em especial por eles terem participado com a alma e o coração de cada momento que passei ao longo do curso, graças a eles tive todo apoio necessário para ter forças até o fim.

Um agradecimento especial devo ao meu namorado por ter sido meu colega, meu professor e meu melhor amigo ao longo de todos estes anos, mas, principalmente por ter me auxiliado com muita paciência neste trabalho. Foi quem me puxou as orelhas para me concentrar, mas foi também quem limpou minhas lágrimas todas as vezes que meu computador estragava e minhas forças se esgotavam. Foi quem me ensinou tantas vezes que eu deveria salvar a última versão em algum lugar que não fosse o próprio computador. E eu, obviamente, só aprendi depois da última vez que tropecei no fio do carregador e o notebook não ligou mais. Mesmo assim, me consolou, consertou meu computador e ficou do meu lado. Por isso e muito mais eu agradeço a ele.

Agradeço ao meu Vô por indiretamente sempre me apoiar quando me pedia para fazer um robô pra ele. Agradeço ao meu Dindo, Dindas, Primas, Primo e Afilhados por sempre procurarem entender o que eu fazia no curso, por estarem sempre preocupados comigo e me apoiarem em todas minhas decisões. Agradeço também a família do meu namorado, mas em especial à minha sogra Leila Schneider que sempre tentava me animar dizendo que eu ia tirar boas notas nas provas mesmo quando percebia minha decepção por ter ido mal.

Não posso deixar de agradecer minhas amigas-irmãs Acsa Menezes, Aicha Yasin, Ane Diel, Gabriela Pozzobon, Lidiany Lima, Renata da Rosa, Sarah Menezes e Stephanie Almeida que apesar de serem muitas, são únicas. Elas comemoravam quando eu comemorava, me ouviam quando eu reclamava, entendiam quando eu precisava diminuir a frequência nos encontros pra me dedicar ao curso, choravam comigo quando eu chorava, e nestes cinco anos sempre estiveram ao meu lado compartilhando dos meus sentimentos e falando as palavras que eu precisava ouvir para renovar as forças. Também às minhas manas do GP agradeço por todas as orações, risadas de dar cãibra nas bochechas e ouvidos atentos neste último ano.

Agradeço também ao meu orientador, Dr. Frederico Menine Schaf, por ter sido professor, amigo, conselheiro, por ter confiado em mim e por ter me orientado durante o último ano, sempre com compreensão, paciência e sabedoria, me dando oportunidades de aprender e evoluir na vida acadêmica.

Aos colegas de curso, Kevim Iochims, Alexandre Piussi, Pedro Carlesso, Matheus Schone, Marcos Rabaioli, Charles Haab, Franthisecolly Vieira, Audren Righi e Germano Henz, obrigada pelas explicações antes das provas, pelo compartilhamento de informações, pela ajuda durante os trabalhos e pela companhia diária nas muitas horas de aulas compartilhadas.

E por fim, gostaria de agradecer aos professores da Engenharia de Controle e Automação, por todo o conhecimento transmitido e à Universidade Federal de Santa Maria pela contribuição na minha formação.

O navio está em segurança no porto, mas não é para isso que servem os navios.

(C.S Lewis)

RESUMO

DESENVOLVIMENTO DE UM SISTEMA COMPLETO PARA AGENDAMENTO MÉDICO

AUTORA: Ana Elisa Marchesan Ramos
ORIENTADOR: Frederico Menine Schaf

Este projeto propõe um sistema completo para agendamento e organização de consultas médicas. Ele é composto de Aplicativo para Médicos e Pacientes desenvolvido para dispositivos móveis, como celulares ou tablets, e uma interface Web para uso do médico e/ou seus funcionários, desta forma compondo um sistema capaz de substituir os atuais sistemas usados nas clínicas que são caros e muitas vezes ineficientes. O dispositivo móvel que possui o aplicativo permite que o paciente faça seu cadastro e agendamento de consultas, além disso, é possível acompanhar através do aplicativo funcionalidades que facilitam o relacionamento entre médico-paciente em relação aos métodos atuais. A interface Web disponível no sistema possibilita ao médico ou seus funcionários acessarem o agendamento de consultas, permitindo assim a gestão dos horários, através de edição, confirmação e correção de consultas. A base de dados utilizada pelo sistema é localizada na nuvem e consiste em um conjunto de tabelas que podem ser combinadas para a extração de uma informação mais completa ou sofrerem buscas individuais para pesquisas mais simples. O serviço de banco de dados utilizado para este sistema é o *Google Fusion Tables*, que é acessado via Interface de Programação de Aplicações (API) do Google para PHP, no caso da interface Web, e API Android, no caso do dispositivo móvel. Todas as informações que circulam entre o banco de dados e as aplicações estão protegidas por criptografia e autenticação.

Palavras-chave: Sistema de Agendamento Eletrônico, Banco de Dados, Interface de Programação de Aplicações, Aplicações para Sistemas Móveis.

ABSTRACT

DEVELOPMENT OF A COMPLETE SYSTEM FOR SCHEDULING MEDICAL

AUTHOR: ANA ELISA MARCHESAN RAMOS
ADVISOR: FREDERICO MENINE SCHAF

This project proposes a complete system for scheduling and organizing medical appointments. It consists of Application for Doctors and Patients developed for mobile devices such as mobile phones or tablets, and a Web interface for medical use and/or its employees, thus composing a system to replace the current systems used in clinics that are expensive and in many cases inefficient. The mobile device that runs the application allows the patient to make user registration and scheduling appointments, in addition, he/she can track through the application features that facilitate the relationship between doctor-patient compared to current methods. The Web interface available in the system enables the doctor or related employees to access appointment scheduling, allowing the management of schedules, through editing, confirmation and consultations correction. The database used by the system is located in the cloud and consists of a set of tables that can be combined for the extraction of a more complete or individual searches for simple research information. The database used for this system is the *Google Fusion Tables*, which is accessed via Application Programming Interface (API) for PHP and API for Android, in the case of mobile devices. All information circulating between the database and applications are protected by encryption and authentication.

Keywords: Electronic Scheduling System, Database Systems, Application Programming Interface, Applications for Mobile Systems.

LISTA DE TABELAS

Tabela 1 - Comparativo entre as tecnologias de agendamento	16
Tabela 2 - Tabela de Valores Médios Praticados por Freelancers de TI.....	51
Tabela 3 - Cálculo de Horas de Programação	52

LISTA DE IMAGENS

Figura 1 - Distribuição do mercado mundial de Sistema Operacional para Smartphones. Fonte: Site da IDC (http://www.idc.com/prodserv/smartphone-os-market-share.jsp).....	17
Figura 2 - Interconexão entre os componentes do sistema e o banco de dados.	24
Figura 3 - Diagrama de Classe do Modelo de Domínio do Banco de Dados.....	25
Figura 4 - Tabela de Cadastro Medico	27
Figura 5 - Tabela de Cadastro de Pacientes.....	27
Figura 6 - Tabela de Horários Médicos	28
Figura 7 - Projeto do Google, onde é gerada API Key, chave P12 e conta de email que será associada ao aplicativo criado na plataforma App Inventor	29
Figura 8 - Alguns campos que devem ser preenchidos com os dados retirados do Projeto do Google	29
Figura 9 - Conexão entre Aplicativos Móveis e Banco de Dados Fusion Tables	30
Figura 10- Diagrama de sequência para conexão entre banco de dados e aplicativo.....	30
Figura 11 - Diagrama de sequência para conexão entre servidor de aplicação e banco de dados	31
Figura 12 - Modo Designer do App Inventor	33
Figura 13 – Modo de Blocos do App Inventor	33
Figura 14 - Geração de QR Code pela plataforma App Inventor	34
Figura 15 - Fluxograma do Aplicativo Doctor	40
Figura 16 - Alerta de Erros no Cadastro.....	41
Figura 17 - Erro de Usuário/Senha Inválido.....	42
Figura 18 - Listas da Tela de Filtragem de Médicos	43
Figura 19 - Listas e Avisos da Tela de Lista de Médicos.....	44
Figura 20 - Fluxograma do Aplicativo Doctor for Doctor	45
Figura 21 - Escolha do tempo de atraso	46
Figura 22 - Tela Inicial da Página Web.....	47
Figura 23 - Agenda de um Médico cadastrado.....	47
Figura 24 - Marcação de horário através da busca por CPF.....	48
Figura 25 - Marcação de horário para usuário localizado por CPF.....	48
Figura 26 - Cadastro de paciente novo no Doctor	49
Figura 27 - Possibilidade de cancelamento de consulta pela interface Web.....	50
Figura 28 - Dispositivos que compõe o Sistema de Agendamento Médico.....	50

SUMÁRIO

1	INTRODUÇÃO	14
1.1	SISTEMAS DE AGENDAMENTO E ATRASOS NAS CONSULTAS	14
1.2	PROPOSTA.....	15
1.3	IMPLEMENTAÇÃO DA PROPOSTA	16
2	FUNDAMENTAÇÃO TEÓRICA.....	19
2.1	COMÉRCIO ELETRÔNICO	19
2.2	APLICATIVOS NA ATUALIDADE	20
2.3	TEORIA DE FILAS	21
3	DESENVOLVIMENTO.....	23
3.1	TOPOLOGIA DO SISTEMA	23
3.2	BANCO DE DADOS	24
3.2.1	Google Fusion Tables	26
3.2.2	Integração com Aplicativo Móvel	28
3.2.3	Integração com Servidor PHP.....	30
3.3	APLICATIVO MÓVEL.....	32
3.3.1	App Inventor	32
3.4	INTERFACE WEB	35
3.4.1	Visualização do usuário.....	35
3.4.2	Servidor de aplicação	37
4	RESULTADOS	39
4.1	DOCTOR E DOCTOR FOR DOCTOR.....	39
4.1.1	Doctor	39
4.1.2	Doctor for Doctor.....	44
4.2	PÁGINA WEB DOCTOR.....	46
4.3	APRESENTAÇÃO DOS PROTÓTIPOS	50

4.4	ESTIMATIVA DE CUSTOS	51
5	CONCLUSÃO.....	53
	REFERÊNCIAS	54
	APÊNDICE A – COMUNICAÇÃO VIA GOOGLE FUSION TABLES.....	56

1 INTRODUÇÃO

Neste trabalho foi desenvolvido um Sistema de Agendamento Eletrônico para consultas médicas a fim de facilitar a interação entre médicos, seus auxiliares e pacientes, inserindo mais essa categoria em uma crescente classe de *e-commerce* que é a da prestação de serviços. O sistema de agendamento é composto por dois aplicativos e uma interface *Web*, este conjunto possibilita a melhora do problema atual a cerca da dificuldade para agendar um horário médico e da espera por consultas que atrasam.

1.1 SISTEMAS DE AGENDAMENTO E ATRASOS NAS CONSULTAS

Hoje é possível comprar todo o tipo de produto e serviço através de dispositivos móveis. Porém, a linha de serviços em geral é uma área onde a situação é bem precária. Um clássico exemplo é o de não existir agendamento *online* para consultas médicas e a necessidade de fazê-las pelo método clássico, via ligação telefônica.

Atualmente nos consultórios a única função disponível para facilitar os agendamentos esta baseada na secretária do médico atender ligações de pacientes e marcar as consultas dos mesmos em uma agenda local, que pode ser manual ou computadorizada.

A dificuldade para marcar horário e a demora em ser atendido podem ser grandes causadores de atrito e descontentamento de pacientes. É necessária um busca manual por médicos do plano médico do paciente (se este o tiver), após este procedimento, outra busca por horários é feita através de ligações para os consultórios dos médicos. Depois de estar marcada a consulta, o paciente ainda está sujeito a chegar ao consultório e ter de esperar horas para ser atendido devido à falta de um bom planejamento.

É uma cena rotineira encontrar um paciente que chegou pontualmente para sua consulta folheando revistas na sala de espera, aguardando o médico. Até nos especialistas mais caros e em países como os EUA para ser atendido é pré-requisito esperar.

Nos EUA, porém, o tema tem ganhado destaque com a reação de pacientes que estão mandando a conta da espera para o médico. Um dos casos foi divulgado na revista Isto É onde a consultora Cherie Kerr aguardou 45 minutos e cobrou US\$ 150 do seu oftalmologista (COSTA, 2011). Alguns médicos americanos têm se proposto a pagar ou conceder brindes a

quem é prejudicado pela demora. A média de atraso nos EUA para uma consulta é de 24 minutos.

No Brasil não há dados sobre o tempo de espera, mas os atrasos são rotineiros. “A situação brasileira e a americana são similares”, fala Hugo Campos, membro da Sociedade de Medicina Participativa, órgão internacional de orientação a pacientes.

Da parte dos médicos, existem algumas iniciativas no Brasil semelhantes às dos EUA, um exemplo é o cirurgião plástico Alan Landecker, em São Paulo. Por dia, passam pelo seu consultório de 20 a 30 pacientes, o que faz com que atrasos sejam rotineiros. A solução do médico foi criar um método para minimizar os aborrecimentos. “Se eu percebo que os pacientes terão que esperar mais de 30 minutos, eles são avisados por telefone e podem remarcar para outra data”, explica Alan. Além disso, quem já está no consultório ganha um lanche, mas quando o atraso supera uma hora, o valor da consulta não é cobrado (COSTA, 2011).

1.2 PROPOSTA

Melhor que tentar abafar o problema com brindes é resolvê-lo de forma a evitar estes transtornos. O sistema de agendamento deste projeto foi desenvolvido com a intenção de diminuir estas esperas nos consultórios médicos e facilitar o relacionamento médico-paciente. Como previamente citado, este sistema possibilita ao paciente marcar sua consulta e ainda possui a função de visualizar se será atendido com pontualidade.

O aplicativo visa descomplicar o processo de agendamento para o paciente que terá acesso a todos os médicos em uma única lista e rapidamente acessará qual deles tem horário para uma data mais próxima, facilitando também a clínica do médico. Assim, deixando a secretária com mais tempo livre para atender aos pacientes que estão na sala de espera e tendo um padrão de agendamento que torna o processo mais organizado e justo.

Além de todos os fatores citados anteriormente, sistemas de agendamento que necessitam de interação com pessoas acabam sofrendo com erros comuns como má compreensão das informações, período de disponibilidade geralmente limitado e alto custo operacional. A Tabela 1 mostra um comparativo entre as tecnologias de agendamento.

Tabela 1 - Comparativo entre as tecnologias de agendamento

		Agendamento em arquivo físico	Agendamento via aplicativo local	Agendamento via aplicativo <i>online</i>
Busca de Informações	Usuário	Difícil	Difícil	Fácil
	Operador	Difícil	Fácil	Fácil
Disponibilidade do Sistema		Limitada a ação Humana	Limitada a ação Humana	Alta (24/7)
Custo Operacional	Humano	Alto	Alto	Baixo
	Armazenamento de Dados	Aumenta muito com o volume de dados	Aumenta com o volume de dados	Aumenta com o volume de dados
Geração de relatórios		Difícil	Fácil	Fácil
Escalabilidade do Sistema		Baixa	Média	Alta

Na Tabela 1 podemos visualizar que a opção que mais traz benefícios é a de agendamento *online*. Quando avaliamos as opções de arquivo físico e até mesmo em agendamento computacional local temos uma alta dependência de recursos humanos, limitando o sistema em velocidade da interação, paralelização das tarefas e volume de atendimento. Ainda existem limitações de infraestrutura, como espaço físico ou armazenamento de dados do computador. Os fatores anteriormente citados impactam na busca de informação pelo usuário bem como pelos operadores do sistema.

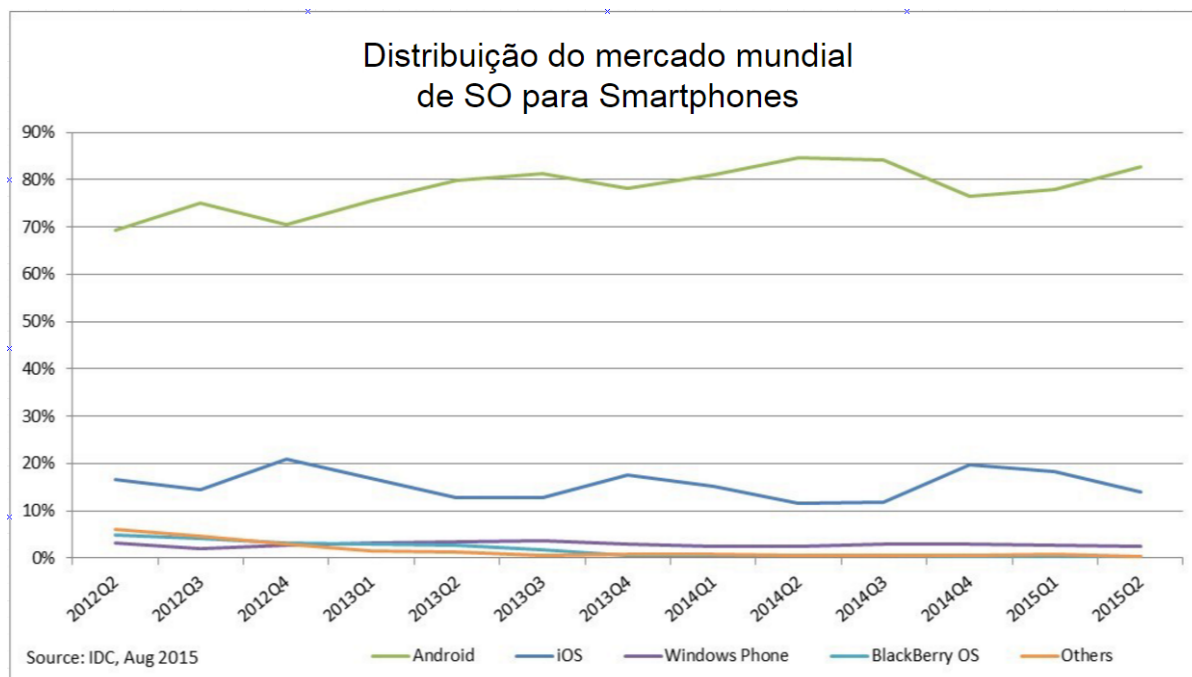
1.3 IMPLEMENTAÇÃO DA PROPOSTA

Visando possibilitar a implementação do sistema em consultórios de todos os portes o sistema foi dividido em duas partes: aplicações móveis, que englobam o aplicativo do médico e do paciente, fazendo todo o processamento de regras no próprio dispositivo do usuário e consultas diretas a um banco de dados *online*; e interface *Web*, onde a aplicação é concentrada na nuvem, de forma que o banco de dados e a geração da interface *Web* são feitas em um servidor remoto não onerando nenhum custo extra de instalação no consultório médico e conferindo escalabilidade ao sistema.

A escolha do sistema operacional (SO) para os aplicativos móveis foi realizada considerando a parcela ocupada por cada SO no mercado de dispositivos móveis (IDC, 2015),

(vide Figura 1), possuir código aberto e plataformas de desenvolvimento para aplicativos serem gratuitas. Desta maneira a escolha natural foi o Android.

Figura 1 - Distribuição do mercado mundial de Sistema Operacional para Smartphones. Fonte: Site da IDC (<http://www.idc.com/prodserv/smartphone-os-market-share.jsp>)



O banco de dados utilizado no trabalho foi o *Google Fusion Tables* (STROPARO, 2009) devido às suas interfaces de programação de aplicação (do inglês *Application Programming Interface* - API) e capacidade de fácil interação com o Android, já que os produtos são mantidos pela mesma empresa, além de ser facilmente escalável que é uma característica de banco de dados na nuvem. O banco de dados do Google ainda possibilita que as tabelas do sistema sejam combinadas à tabelas de domínio público para complementar as procuras, função que pode ser utilizada no futuro para incrementar o sistema com sugestões ou na análise de dados por parte do médico.

A interface *Web* que será utilizada no consultório médico consiste em um servidor de aplicação remoto rodando um aplicativo em PHP (*Hypertext Preprocessor*) que fornece as lógicas do programa e um *site* desenvolvido utilizando HTML 5. A segurança dos dados é feita com o acesso do servidor de aplicação ao banco de dados sendo protegido por autenticação OAUTH 2.0.

O trabalho inicialmente mostrará uma fundamentação teórica e depois será apresentado o desenvolvimento, no qual está a topologia que apresenta as interconexões entre

aplicativos, interface *Web*, banco de dados e as tecnologias necessárias para o funcionamento do sistema. Após as devidas explicações dos tópicos citados, os protótipos e suas respectivas estimativas de custo, seguido da conclusão encerrarão o conteúdo escrito deste trabalho.

2 FUNDAMENTAÇÃO TEÓRICA

Esta seção visa justificar a necessidade do desenvolvimento de um sistema de agendamento médico. Desta forma, serão abordados os fatores que levam a esta necessidade, os motivos que afetam o relacionamento médico-paciente e brevemente será citado como o projeto tende a melhorar o mesmo.

2.1 COMÉRCIO ELETRÔNICO

A Internet transformou a relação das empresas com seus clientes, parceiros e fornecedores, inaugurando uma nova era no mundo dos negócios, alterando processos, funções, relações e formas de realizar negócios através do comércio eletrônico (CE). Não há mais necessidade de elementos clássicos para que ocorra uma venda ou agendamento: endereço físico, secretária, dinheiro. Pode-se comprar sem sair de casa, fazer contato sem fazer chamada telefônica e pagar sem usar papel. Tais avanços se devem principalmente pelo fato da Internet ser acessada por novos meios, tais como telefones celulares e *tablets* (TESTA, 2006).

A capacidade e o potencial dos mercados eletrônicos dependem de quão bem, se consegue alavancar o poder da Internet para criar vantagens competitivas sustentáveis. Algumas das capacidades do mercado eletrônico, são:

- Comunicação instantânea: Ajuda na comunicação instantânea entre os vários participantes dos sistemas de negócios. Também ajuda na redução da "hora de mercado" para novos produtos;
- Acesso Global: Os produtos/serviços oferecidos através de mercados eletrônicos têm um alcance global e proporcionam maior acesso aos mercados;
- Personalização: Quando se tem capacidade para oferecer produtos e serviços em tempo real, a capacidade de personalizar mercadorias para as necessidades são maiores;
- Aumento da disponibilidade: O e-commerce oferece uma maior disponibilidade de produtos da empresa estando disponíveis 24 horas por dia, 7 dias da semana nos 365 dias do ano;

- De-Intermediação: Ajuda na eliminação de intermediários ineficientes, oferecendo uma simplificada distribuição eletrônica e diferenciação dos produtos baseados na escolha do cliente;
- Consolidação e a Convergência: Ajudam a uma maior consolidação e convergência, o que facilita as economias de escala, facilitando a implantação de ordens e consequentemente o seu cumprimento. Isso permite a revitalização dos produtos que agregam valor através de profundidade, contexto ou digitalização;
- Colaboração: Facilita a automatização de transações eletrônicas entre empresas, suporte em tempo real de troca de informações permitindo assim um processo colaborativo.

Outro termo bem forte que se relaciona com o Comércio Eletrônico é a sigla CRM (*Customer Relationship Management*), traduzida GRC (Gestão de Relacionamento com o Cliente). Este termo fala sobre softwares e serviços orientados para ajudar as empresas a gerir mais eficazmente suas relações com os clientes, por todos os tipos de canais, diretos ou indiretos. Os aplicativos de CRM podem ajudar na eficaz gestão de relações com os clientes (GREENBERG, 2001). Dado que o serviço médico é totalmente em função de seus pacientes, surge a ideia de facilitar esta relação através do comércio eletrônico.

2.2 APLICATIVOS NA ATUALIDADE

Um aplicativo móvel, conhecido normalmente por seu nome abreviado app, é um *software* desenvolvido para ser instalado em um dispositivo eletrônico móvel, como um telefone celular, um *smartphone* ou um *tablet*. Este aplicativo pode ser baixado pelo usuário através de uma loja *online*, tal como a *Google Play*.

A termo "app" é uma abreviatura de "aplicação de *software*". Os *apps* popularizaram-se em 2008 e são normalmente assegurados e fornecidos pelos administradores do sistema operacional móvel. No caso de plataformas abertas como a do Android não existe um procedimento de assegurar o funcionamento dos aplicativos, podendo qualquer desenvolvedor fornecer seu *app* para o sistema operacional.

Originalmente as aplicações móveis foram criadas e classificadas como ferramentas de suporte à produtividade e à recuperação de informação generalizada, incluindo correio

eletrônico, calendário, contatos, mercado de ações, informações meteorológicas entre outras do gênero. No entanto, a crescente procura, a disponibilidade facilitada e a evolução dos *apps*, conduziram à rápida expansão para outras categorias, como jogos, GPS, serviços de acompanhamento de pedidos, compra de bilhetes, confirmações de presenças, conexões nas redes sociais, aplicações nas mais diversas áreas, como saúde, mercados de ações, etc., tudo isto para a generalidade dos dispositivos móveis. A disseminação no número e variedade de aplicações serviu de fonte de estímulo à criação de inúmeros *apps* para atender as necessidades mais diversas da maioria dos utilitários.

Os *apps* tem o propósito de facilitar o dia a dia ao seu utilizador, fornecendo-lhe as mais diversas funcionalidades com infinitas possibilidades.

2.3 TEORIA DE FILAS

A teoria das filas é um estudo matemático da melhor solução para otimizar o processo das filas, este estudo proporciona uma maneira de definir o ambiente de um sistema de filas prevendo respostas prováveis e tempos de espera.

A teoria de filas tenta através de análises matemáticas detalhadas encontrar um ponto de equilíbrio que satisfaça o cliente e seja viável economicamente para o provedor do serviço. De forma geral o sistema de filas é um processo de chegada de clientes a um sistema de atendimento para receber um ou mais serviços, executados por certa quantidade de servidores.

O termo cliente é usado de maneira geral e não implica necessariamente num cliente humano, como por exemplo, um processo esperando para utilizar a CPU. No caso deste projeto o cliente é o paciente, o médico é o servidor e a consulta é o serviço oferecido.

A teoria das filas foi desenvolvida para prover modelos que retratem previamente o comportamento de um sistema fornecedor de serviços que possuam demandas que aumentam aleatoriamente (GROSS, 1974). As formações de filas ocorrem porque a procura pelo serviço é maior do que a capacidade do sistema de atender a esta procura ou por fatores externos como o atraso dos outros clientes.

Como forma de aferir o comportamento do sistema de filas, associam-se medidas de desempenho como: tempo médio de espera dos clientes na fila, tempo médio de chegada de clientes, probabilidade de encontrar o sistema lotado, entre outras. Da mesma forma, para o desenvolvimento deste projeto foi observado o comportamento do processo de agendamento para estimar a melhor solução e de que forma essa solução seria implementada.

Existem muitas aplicações respeitáveis da teoria, a maioria das quais tem sido documentadas na literatura de probabilidade, pesquisa operacional e engenharia industrial. Alguns exemplos são fluxo de tráfego (veículos, aeronaves, pessoas, comunicações), escalonamento (pacientes em hospitais, *jobs* em máquinas, programas em computadores) e projetos de atendimentos à serviços (bancos, correios, parques de diversão, restaurantes *fast-food*).

O sistema de agendamento médico é um novo exemplo dessa aplicação, e se relaciona com a teoria de filas no momento em que a motivação de ambos se dá pela melhor utilização dos serviços disponíveis, no menor tempo de espera com a maior rapidez no atendimento.

3 DESENVOLVIMENTO

Nesta seção será explicada a topologia escolhida no trabalho e as ferramentas utilizadas para a implementação do mesmo, abordando o que foi feito ao longo do trabalho e explicando os tipos de comunicação. Será descrito também os aplicativos desenvolvidos, o *Website* e as plataformas de desenvolvimento utilizadas para isto.

3.1 TOPOLOGIA DO SISTEMA

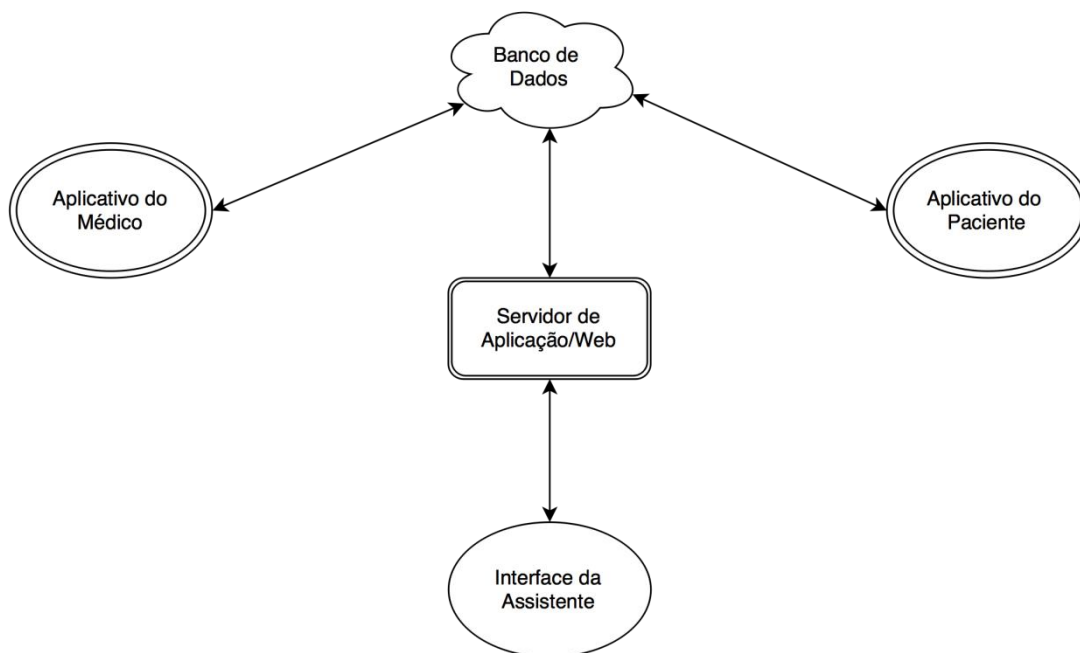
O projeto tanto de um *site* quanto de um aplicativo necessita das definições dos requisitos a serem atendidos, pois assim torna-se possível saber o que se espera e qual a solução mais adaptada a cada situação.

Para isso, definiram-se três tipos de usuário: o paciente, o médico e a secretária. Cada um destes possui particularidades e necessidades diferentes. O paciente, por exemplo, necessita da facilidade de um aplicativo móvel que possa ser acessado do seu *smartphone* para o agendamento de consultas através da filtragem dos médicos por convênio, especialidade e disponibilidade. O médico precisa ter acesso aos seus horários e a possibilidade de avisar seus pacientes do dia de algum possível atraso. Já a secretária necessita do acesso à agenda do médico para a gestão dos horários, através de edição, confirmação e correção de consultas, além da possibilidade de criação de cadastro para pacientes não cadastrados.

A partir deste ponto é possível observar a necessidade da criação de um aplicativo móvel para o paciente, outro diferente destinado ao médico e ainda, uma interface *Web* para a secretária acessá-la de seu computador.

Além disso, é necessário um banco de dados onde as informações serão adicionadas, ou extraídas dependendo do que se deseja. Os aplicativos se conectam diretamente com o banco de dados escolhido através do uso de API's designadas para Android. Já a comunicação entre o *site* e o banco de dados é feita através do servidor de aplicação/*Web*. Estas conexões são mostradas na Figura 2 e serão melhor explicadas na próxima seção.

Figura 2 - Interconexão entre os componentes do sistema e o banco de dados.



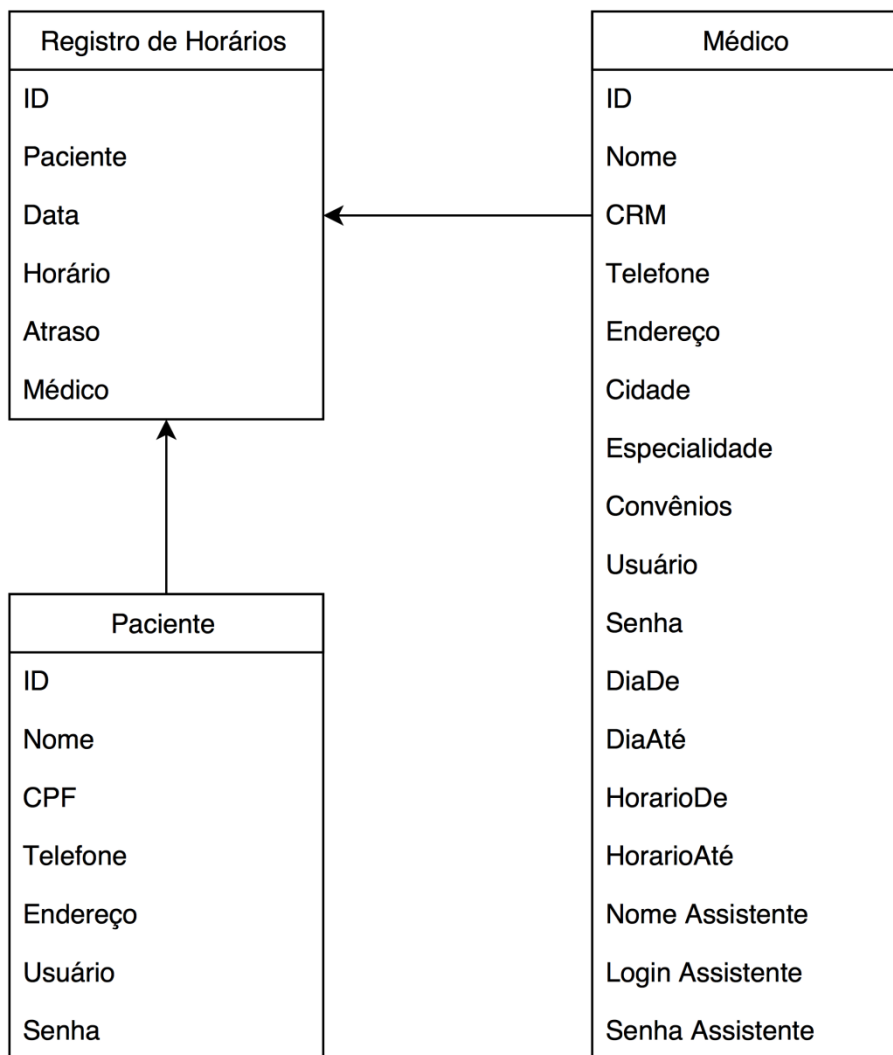
Para o uso dos aplicativos, apenas é necessário um dispositivo móvel que possua Android onde o usuário possa designar tarefas ao sistema. E, para o uso do *site* apenas é necessário um dispositivo que tenha um navegador *Web* para acessá-lo. Naturalmente que em todos os casos é necessário o provimento de conexão com a Internet.

O fluxo de informações recebidas e enviadas pelo equipamento com Android e pelo computador do consultório são feitos por Internet, é dessa forma que todos os dados são conectados ao servidor.

3.2 BANCO DE DADOS

Pelos aplicativos e pelo *site* circulam informações como: Nome, CRM, CPF, Usuário, Senha, Endereço, Telefone, Convênio, etc. Para o armazenamento destes elementos é necessário um banco de dados, e este banco de dados é composto por três tabelas diferentes: a tabela de cadastro médico, a tabela de cadastro de pacientes e a tabela dos horários marcados. O modelo de domínio destas tabelas é apresentado na Figura 3.

Figura 3 - Diagrama de Classe do Modelo de Domínio do Banco de Dados



Pode-se observar que a Tabela de Registro de Horários Médicos é totalmente dependente das outras duas. Esta tabela recebe do aplicativo do Paciente a data e o horário da consulta e para a efetivação do agendamento é necessário que exista um usuário conectado, o qual especifica para quem é o horário, e um usuário de médico para definir com quem será a consulta. Estes dois dados são obtido da tabela de pacientes e da tabela de médicos respectivamente, caso registros destas duas sejam excluídos, a tabela de horários se torna inconsistente.

Na próxima subseção será apresentado o banco de dados utilizado em todo o trabalho e as formas de conexão com este.

3.2.1 Google Fusion Tables

O *Google Fusion Tables* é uma nova opção para utilizar um banco de dados na nuvem computacional do Google com a possibilidade de escalabilidade da solução. Aprofundando um pouco mais, o *Google Fusion Tables* é banco de dados *online* com nova tecnologia que uniformiza diversos tipos de dados e promete economia às empresas.

O Google lançou este novo banco de dados *online* com o objetivo de revolucionar o gerenciamento de dados. A ideia é driblar as limitações dos bancos de dados tradicionais e simplificar as operações de relacionamento de informações (STROPARO, 2009).

O *Fusion Tables* também oferece uma tecnologia de espaço de dados, conceito que existe desde os anos 90 e o Google, percebendo seu potencial, o desenvolve desde a compra da Transformic, em 2005, que é uma pioneira da tecnologia. O esquema de “espaço de dados” tenta resolver o problema de vários tipos e formatos de dados nas empresas, que gastam muito em dinheiro e esforços para torná-los uniformes, com o objetivo de armazená-los e analisá-los em bases de dados convencionais.

Os “espaços de dados” preveem um sistema que cria um índice para oferecer acesso a dados de vários tipos e formatos, resolvendo o problema que o Google chama de “Torre de Babel”. A tecnologia permite que o Google inclua, nas tabelas bidimensionais tradicionais de base de dados, uma terceira coordenada com elementos como *reviews* de produtos, *posts* e mensagens do Twitter, além de uma quarta “dimensão” de atualizações em tempo real. “Agora temos um espaço com quatro dimensões onde podemos incluir novas perguntas para criar novos produtos e oportunidades de marketing”, diz o anúncio (STROPARO, 2009).

O *Fusion Tables* atende os requisitos de robustez, permitindo ainda em versão gratuita para testes tabelas de até 100 MB. Outros tipos de banco de dados também atendem o requisito de robustez, porém são de mais complexa integração com o sistema. No caso do *Fusion Tables* existe uma API bem preparada para aplicativos em Android e com recursos de autenticação simples de serem configurados. Além disso, foi crucial para a escolha deste banco de dados que ele fosse compatível com o App Inventor, este é a interface gráfica usada para a criação dos aplicativos que será elucidada na próxima seção.

Como já mencionado, o sistema de agendamento precisou de três tabelas do *Google Fusion Tables* que podem ser vistas na Figura 4, Figura 5 e Figura 6.

Figura 4 - Tabela de Cadastro Medico

Cadastro Medico anaem@gmail.com

Edited at 23:15 Share

File Edit Tools Help Rows 1 Cards 1 Map of Location

Filter - No filters applied

1-6 of 6

Nome	CRM	Telefone	Endereco	Cidade	Especialidade	Convenios	Usuario	Senha	HorarioDe	HorarioAte	DiasDe	DiasAte	Nome Secretaria	Login Secretaria	SenhaSecretaria
Marcia Lisboa	040500	5581378850	Av Presidente Vargas	Santa Maria	Ginecologista	IFE	marcia	lisboa	8	18	2	5	Ana Ramos	anaramos	ramos123
João Carvalho	55417	5599949466	Av Presidente Vargas	Santa Maria	Oftalmologista	IFE	joao	joao	8	18	1	7	Catarina Flores	catflores	flores321
Maria Elisa		5581349655	Av Presidente Vargas	Santa Maria	Oftalmologista	Unimed	maria	elisa	8	18	2	7	Alexandra Ceolin	ale.ceolin	c.2301
Carlos Barbosa	098526	4797780992	Av Waldemar Grubba, 3010	Jaraguá do Sul	Cardiologista	Particular	carlosb	barbo...	8	18	1	7	Luana Teixeira	luana.t	lu.ana
Carina Flores	876543	4797706555	Rua Leopoldo João Grubba 37	Jaraguá do Sul	Pediatra	Unimed	carina.f	flores...	8	18	2	7	Raquel Gomes	raquel	gomes76
Renata Lima	763937	5599562332	Av Av Rio Branco	Santa Maria	Psiquiatra	IFE	renata	lima93	8	18	1	6	Sarah Cardoso	sarah	cardoso89

Figura 5 - Tabela de Cadastro de Pacientes

Cadastro Paciente anaem@gmail.com

Add Attribution - Edited on 2015 July 3 Share

File Edit Tools Help Cadastro de Usuário Cards 1 Map of Location

Filter - No filters applied

1-14 of 14

Nome	CPF	Telefone	Endereco	Usuario	Senha
Ana Elisa Marchesan Ramos	03162074037	5581376479	Rua André Marques 402/102, Centro	anaemr	elisa212
Aicha Manara Yasin	03289954013	5591890001	Rua Antônio Figueira 245, Camobi	aichayasin	aichayasin22
Thayane Lawrence Brasil Kessner	02786452082	5596417407	Rua André Marques 402/102, Centro	thykessner	1458201
Sarah Menezes dos Santos	03383978095	4198257526	Avenida Silva Jardim 3749, Água Verde, Curitiba	menezesarah	menezesarah
Amina Manara Yasin	03679819048	5591735555	Rua Antônio Figueira 245	aminamanara	300996
Lidiany Oliveira de Lima	03650513064	5581067555	Rua Anauelina Coden Portela 366/105, São João	lidilima	lidiany123
Tamara Manara Yasin	03434731008	5591643553	Rua Antônio Figueira 245	tamarayasin	123456
Acsa Christina Borges Menezes	03041087073	5581129326	Travessa Maria Calcagno 02, Camobi	acsamenezes	090712
Ana Paula Rover Mello	09061944437	5581025555	Rua General Neto 551	anamello	060693
Vitória Raissa Diss	03744616002	05596691249	Rua José Crivelaro 215/203, Km 3	vitoriadiss	vi1234
Giuliane Cardoso	01555948006	5591580181	Rua Felipe de Oliveira 155/201, Centro	giulianecardoso	43215
Carina Rezende Flores	03349120083	5533175464	Rua Floriano Peixoto 106/202	carinaflores	strogonoff
Stephanie de Almeida Machado	02489548069	5599742061	Av Machado Soares 400	fane	090793
Ana	03162074037	5581376479	Rua André	anae	elisa

Figura 6 - Tabela de Horários Médicos

Horarios Medicos

Edited at 20:20

File Edit Tools Help Rows 1 Cards 1 Map of Location

Filter No filters applied. Sorted by Data

Usuario	Data	Horario	Paciente	Atraso
marcia	29/06/2016	14	thykessner	
joao	28/11/2015	15	anaemr	
joao	28/06/2016	15	anaemr	
marcia	27/06/2016	13	anaemr	
marcia	27/06/2016	12	anaemr	
marcia	26/06/2016	13	aichayasin	
maria	26/06/2015	10	aichayasin	
joao	23/07/2016	12	anaemr	
joao	23/07/2016	12	anaemr	
joao	23/07/2016	12	anaemr	
marcia	23/01/2016	12	anaemr	
marcia	23/01/2016	10	anamello	
marcia	20/11/2015	15	anaemr	
anaemr	13/06/2016	14	anaemr	
anaemr	13/06/2016	12	anaemr	
anaemr	13/06/2016	10	anaemr	
joao	11/10/2015	16	anamr	
maria	10/10/2015	10	anaemr	
joao	10/10/2015	17	anaemr	
maria	10/10/2015	11	anaemr	
joao	10/10/2015	16	anaemr	

Nestas tabelas estão armazenadas todas as informações envolvidas no sistema de agendamento desenvolvido. Graças às funções do *Google Fusion Tables* é possível editar e extrair estas informações para os aplicativos e para o *site*. A integração desse banco de dados com estes elementos será esclarecida a seguir.

3.2.2 Integração com Aplicativo Móvel

Para sincronizar as tabelas com o aplicativo é necessário que o banco de dados esteja autenticado a aplicação. Esta autenticação é feita da seguinte maneira:

1. Uma conta de serviço garante acesso a um conjunto de API's do Google
2. Associado à esta conta de serviço está a API do *Fusion Tables* (API key) (Figura 7)
3. Utilizando os itens 1 e 2 é gerado um arquivo chave (chave P12)
4. Este arquivo é inserido na aplicação conferindo-a acesso ao banco de dados.

Após estes procedimentos, na plataforma de programação existem campos onde cada um destes itens deve ser alocado na plataforma de desenvolvimento App Inventor (Figura 8).

Figura 7 - Projeto do Google, onde é gerada API Key, chave P12 e conta de email que será associada ao aplicativo criado na plataforma App Inventor

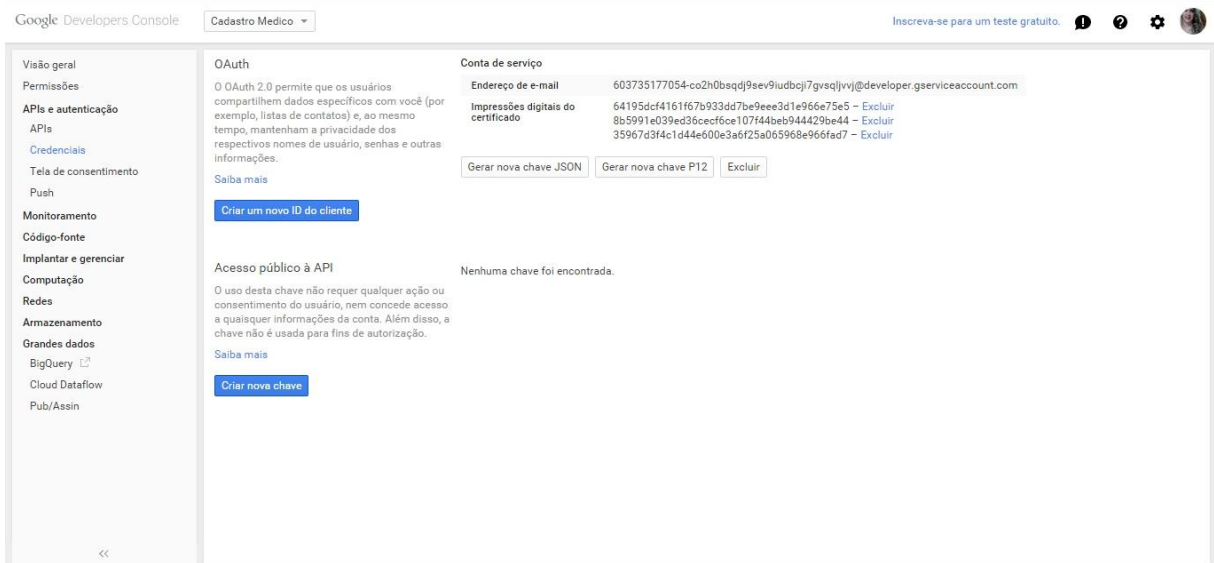
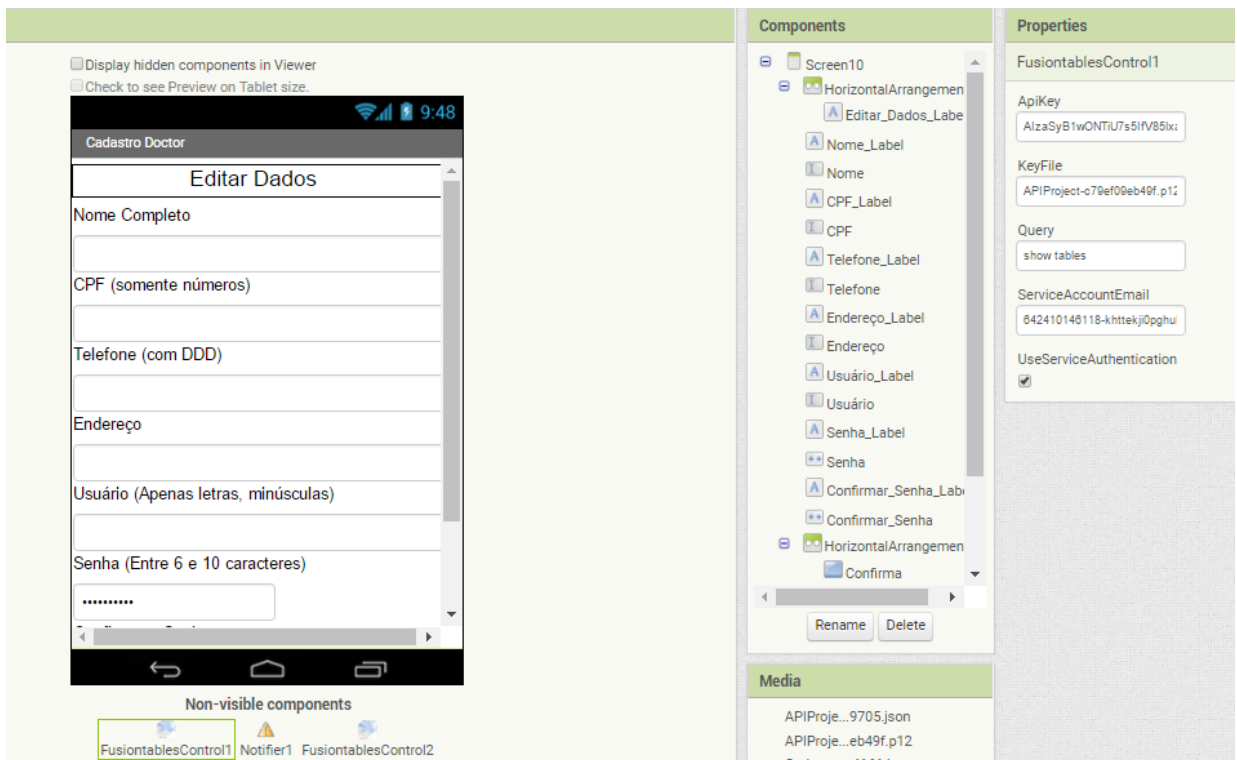
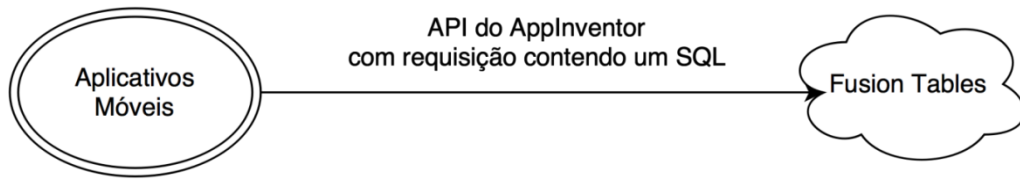


Figura 8 - Alguns campos que devem ser preenchidos com os dados retirados do Projeto do Google



As requisições entre o aplicativo e o banco de dados são feitas por meio de *queries* SQL (Figura 9).

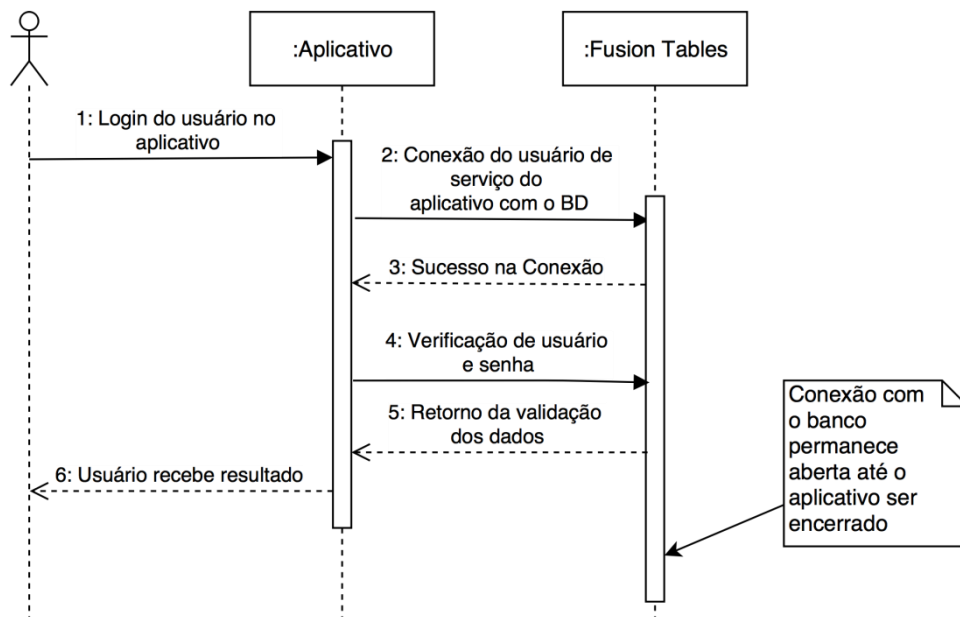
Figura 9 - Conexão entre Aplicativos Móveis e Banco de Dados *Fusion Tables*



A conexão com o banco de dados é iniciada após a primeira consulta feita pelo aplicativo. Esta consulta ocorre no momento em que o usuário faz o *login*.

O diagrama de sequência apresentado na Figura 10 mostra o procedimento anteriormente explicado. Ainda, é possível perceber que a conexão com o banco de dados permanece aberta após o *login*, evitando assim a necessidade de reconexão a cada consulta.

Figura 10- Diagrama de sequência para conexão entre banco de dados e aplicativo



3.2.3 Integração com Servidor PHP

Para a integração com o servidor PHP foi necessário o uso de uma biblioteca específica desenvolvida pelo Google para interconexão de vários de seus serviços com outras linguagens de programação. Esta biblioteca é chamada de Google API.

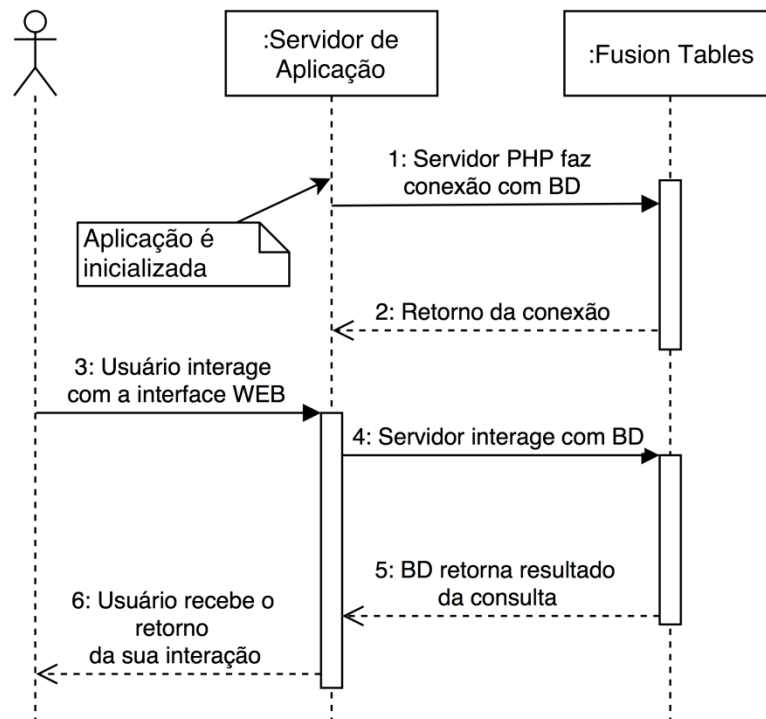
O Google API é construído sobre HTTP (*Hypertext Transfer Protocol*) e JSON (*JavaScript Object Notation*), portanto, qualquer conexão padrão da *Web* pode enviar pedidos e analisar as respostas.

No entanto, em vez de criar solicitações HTTP e analisar respostas manualmente, a API do Google proporciona uma melhor integração de linguagem, melhorias de segurança e suporte para fazer chamadas que requerem autorização do usuário (Autenticação OAUTH2.0).

A conexão com o banco de dados é feita após a aplicação ser inicializada pelo servidor de aplicação. Diferentemente do aplicativo para celular, onde cada aplicativo possui uma conexão com o banco de dados, neste caso existe apenas uma conexão entre o servidor de aplicação e o *Fusion Tables*. Todas as sessões da *Web* compartilham esta conexão.

O diagrama de sequência apresentado na Figura 11 ajuda a descrever o comportamento citado anteriormente.

Figura 11 - Diagrama de sequência para conexão entre servidor de aplicação e banco de dados



3.3 APLICATIVO MÓVEL

O sistema de agendamento é composto por dois aplicativos: o Doctor, para os pacientes, e o Doctor for Doctor, para os médicos. A aplicação desenvolvida para o paciente tem como principais funções que o cliente através de um cadastro possa acessar sua conta e agendar consultas com a facilidade de ter os médicos filtrados por cidade, convênio e especialidade. Outra função disponível no Doctor é a possibilidade de saber se sua consulta vai atrasar, evitando que o paciente perca horas em uma sala de espera de consultório.

O aplicativo desenvolvido para o médico possui a mesma interface para cadastro que a do paciente, com o acréscimo de alguns dados. Além disso, o médico pode visualizar sua agenda e avisar seus pacientes caso as consultas estejam atrasadas.

A seguir será especificada que plataforma foi utilizada para desenvolver ambos aplicativos.

3.3.1 App Inventor

O App Inventor é uma plataforma para desenvolvimento de aplicativos para Android desenvolvido pelo Google e atualmente mantido pelo Instituto de Tecnologia de Massachusetts (MIT). O programa designa um modo de *designer* (Figura 12) para a construção das telas e uma parte de edição de blocos (Figura 13) onde é feita uma programação gráfica via blocos.

O aplicativo criado nesta plataforma roda diretamente no navegador *Web* e armazena os dados deste na nuvem, diretamente em um servidor do Google.

Figura 12 - Modo Designer do App Inventor

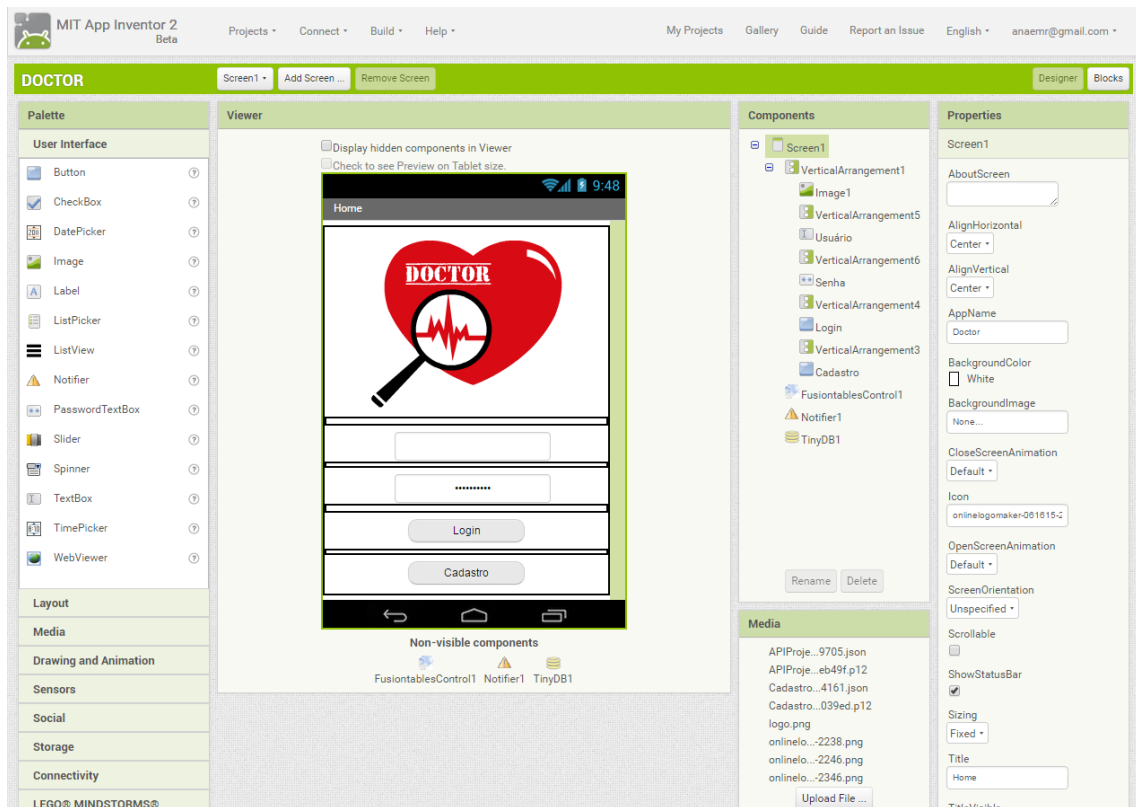
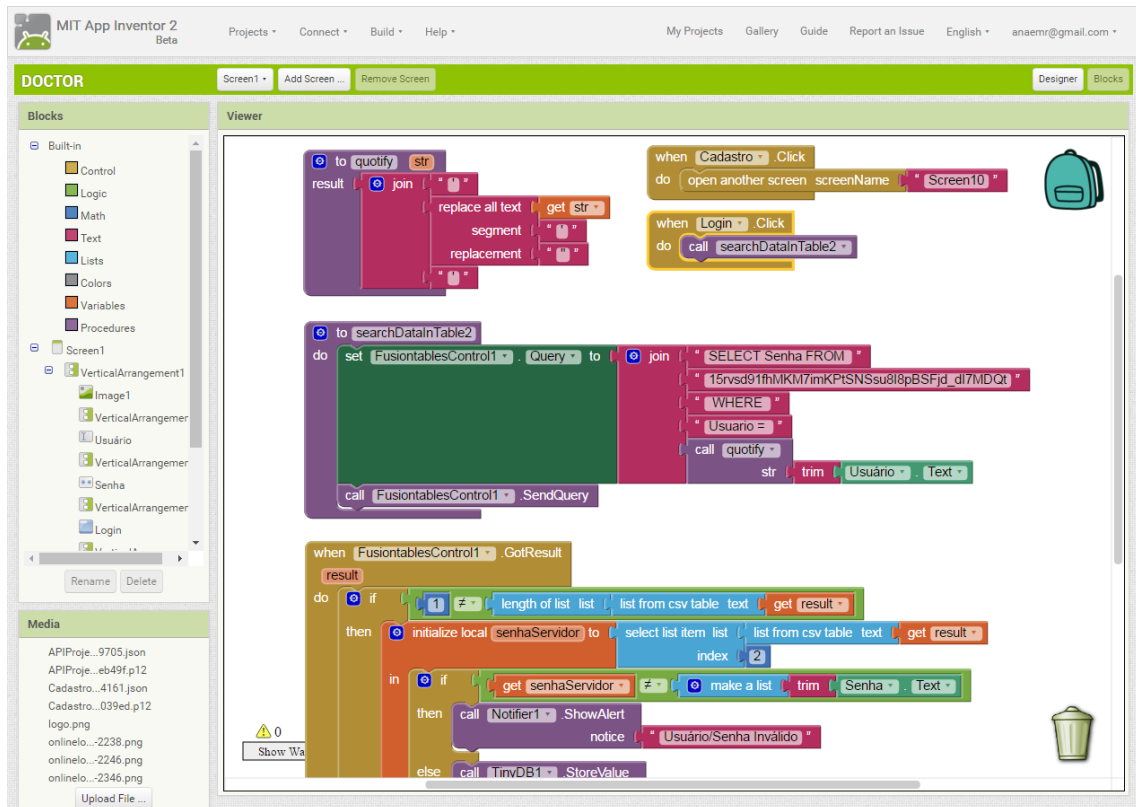


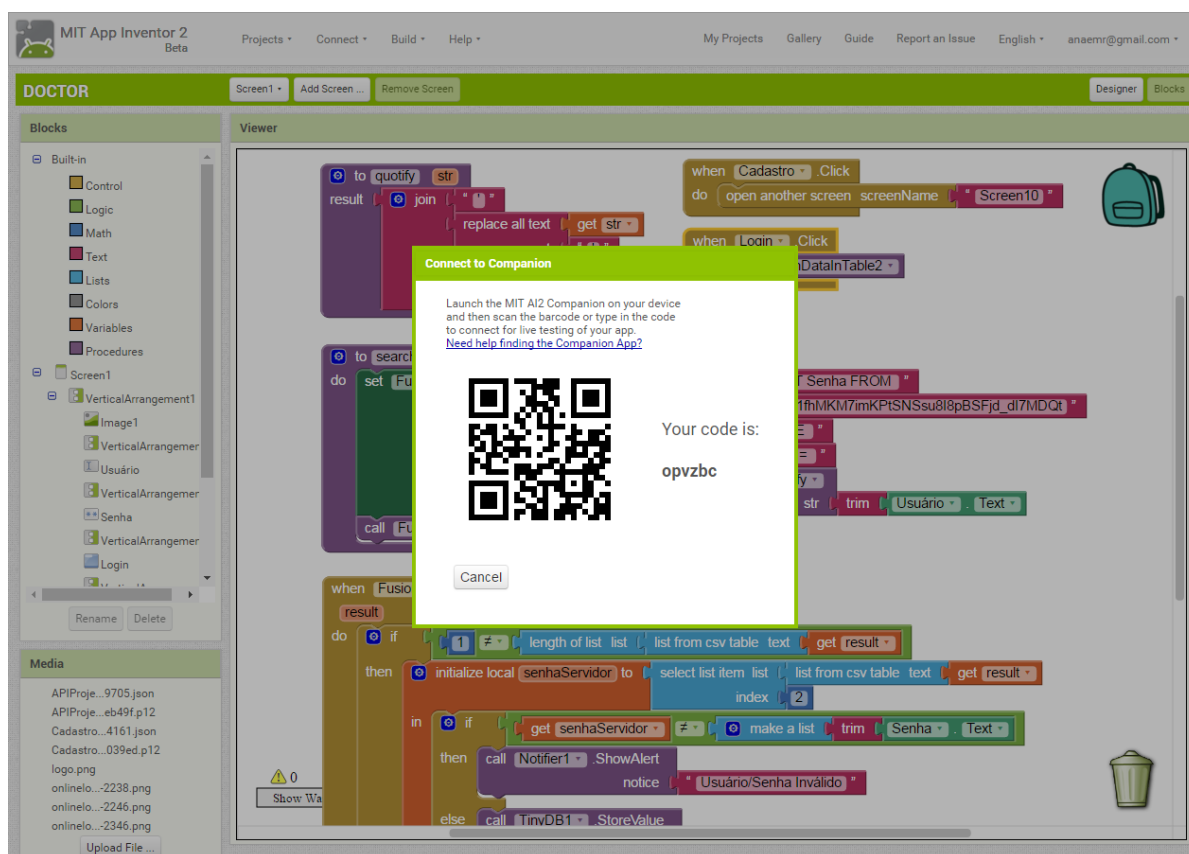
Figura 13 – Modo de Blocos do App Inventor



Nesta plataforma é possível criar as telas com os botões, caixas de texto, listas, *checkbox*, campo de senha, imagens, etc., de acordo com as necessidades do aplicativo e com isto partir para a programação do mesmo. Além disso, pode-se instalar o Aplicativo MIT AI2 Companion em um dispositivo Android, para ler o *QR Code* gerado pelo computador (Figura 14) e a partir disso é possível observar em tempo real o aplicativo que está sendo criado. Desta forma podem-se fazer os testes e ajustes necessários mais rapidamente.

Depois de finalizado, o aplicativo pode ser compilado e instalado como qualquer outro programa em um dispositivo com Android ou ainda pode-se colocar o aplicativo na Google Play, loja *online* que disponibiliza aplicativos para o sistema operacional da empresa.

Figura 14 - Geração de QR Code pela plataforma App Inventor



O App Inventor foi desenvolvido no sentido de permitir que todas as ferramentas disponíveis em dispositivos com Android estivessem ao alcance dos programadores, sendo possível completa abstração de *hardware*, ou seja, é possível gerar um programa para rodar em diferentes aparelhos sem nos preocuparmos com o modelo específico do equipamento.

3.4 INTERFACE WEB

Para completar o sistema de agendamento sentiu-se a necessidade da criação de uma interface *Web* para que a secretária do médico tivesse acesso à agenda do mesmo, permitindo assim que ela faça a gestão dos horários, através da edição, confirmação, correção de consultas e cadastro de novos pacientes.

3.4.1 Visualização do usuário

O *front-end*, ou seja, a forma que o usuário visualiza o *site*, foi construído usando diferentes tecnologias. Entre elas pode-se citar: HTML5 para os elementos da página, CSS para os estilos e JavaScript para lógicas de validação. Estas tecnologias e a forma que o *front-end* interage com o PHP serão descritas a seguir.

3.4.1.1 HTML5

HTML5 (*Hypertext Markup Language*, versão 5) é uma linguagem para estruturação e apresentação de conteúdo para a *World Wide Web* e é uma tecnologia chave da Internet. É a quinta versão da linguagem HTML. Esta nova versão traz consigo importantes mudanças quanto ao papel do HTML no mundo da *Web*, através de novas funcionalidades como semântica e acessibilidade. Um dos seus principais objetivos é facilitar a manipulação dos elementos, possibilitando o desenvolvedor modificar as características dos objetos de forma não intrusiva, fazendo com que isso fique transparente para o usuário final. Para se ter uma ideia disso, diferente das versões anteriores, o HTML5 fornece ferramentas para o CSS e o Javascript fazerem seu trabalho da melhor possível de forma que um *Website* ou aplicação continue leve e funcional (FEITOSA, 2012).

No desenvolvimento da página *Web* o HTML5 foi muito utilizado para a estruturação do site, como: títulos, subtítulos, botões, tabelas, etc. O HTML5 está disponível em várias plataformas diferentes e funciona com a mesma eficiência em PC's, *smartphones* e *tablets*, isto possibilita ao usuário a facilidade de acessar a página do dispositivo que preferir.

3.4.1.2 CSS

O *Cascading Style Sheets* (CSS) é uma "folha de estilo" composta por "camadas" e utilizada para definir a apresentação (aparência) em páginas da Internet que adotam para o seu desenvolvimento linguagens de marcação (como XML, HTML e XHTML). O CSS define como serão exibidos os elementos contidos no código de uma página da Internet e sua maior vantagem é efetuar a separação entre o formato e o conteúdo de um documento (PEREIRA, 2009).

Esta ferramenta foi utilizada para estilizar elementos da página como: cores dos títulos, fonte e tamanho dos textos, posicionamentos de todo o aspecto visual do *site*, entre outros. Em resumo, usou-se este instrumento para o usuário ter acesso à uma interface simples e amigável.

3.4.1.3 JavaScript

JavaScript é uma linguagem de programação que foi originalmente implementada como parte dos navegadores *Web* para que *scripts* pudessem ser executados do lado do cliente e interagissem com o usuário sem a necessidade deste *script* passar pelo servidor, controlando o navegador, realizando comunicação assíncrona e alterando o conteúdo do documento exibido (CORREA, 2015).

É atualmente a principal linguagem para programação *client-side* em navegadores *Web*. Em conjunto com o HTML e o CSS, o JavaScript - bem como suas bibliotecas, dialetos e tecnologias relacionadas - é um dos alicerces fundamentais da *Web*.

Foi concebida para ser uma linguagem *script* com orientação a objetos baseada em protótipos, tipagem fraca e dinâmica e funções de primeira classe. O uso primário de JavaScript é escrever funções que são embarcadas ou incluídas em páginas HTML e que interagem com o Modelo de Objeto de Documentos (DOM) da página (CORREA, 2015).

Um exemplo deste uso na página desenvolvida é a validação dos valores do formulário para garantir que são aceitáveis antes de serem enviados ao servidor. Ou seja, o JavaScript é responsável pelas validações feitas no navegador *Web* (*Webbrowser*), tais como:

- No *login* verifica se existe algum campo em branco e se o usuário é válido dentro dos padrões estabelecidos na criação do mesmo;

- No agendamento de consultas examina se o formato da data é válido;
- No cadastro, verifica se existe algum campo não preenchido e se campos numéricos possuem letras ou caracteres inválidos e vice-versa.

Caso algum destes testes não passar na validação o usuário recebe um aviso na tela que lhe informa o que aconteceu de errado. Pelo fato do código JavaScript rodar localmente no navegador do usuário, e não em um servidor remoto, o navegador pode responder a tais ações rapidamente, fazendo uma aplicação mais responsiva.

3.4.1.4 Interação com PHP

O *front-end* envia os dados para serem processados pelo PHP através de requisições HTTP, mais especificamente POSTs. O PHP aplica as regras que necessitam de validação com consultas no banco de dados. O *front-end* recebe como resposta para suas requisições HTTP um novo HTML que irá exibir as informações compatíveis com a consulta feita pelo usuário.

Um exemplo prático do funcionamento explicado anteriormente pode ser dado pelo uso do botão de marcar uma consulta, onde o sistema pega todas as informações preenchidas no formulário de marcar consulta e envia para o servidor PHP. O servidor verifica com o banco se o horário permanece disponível, caso isso aconteça o servidor PHP salva a consulta e retorna uma nova página de horários já exibindo este último salvo. Caso ocorra um erro é retornada uma página diferente que exibe o alerta de erro no marcar a consulta.

3.4.2 Servidor de aplicação

Um servidor de aplicações disponibiliza um ambiente para a instalação e execução de certas aplicações, centralizando e dispensando a instalação nos computadores clientes. Os servidores de aplicação também são conhecidos por *middleware* (RAYAN, 2014).

O objetivo do servidor de aplicações é disponibilizar uma plataforma que separe do desenvolvedor de *software* algumas das complexidades de um sistema computacional. No desenvolvimento desta aplicação de agendamento médico, por exemplo, o foco dos desenvolvedores deve ser a resolução de problemas relacionados aos agendamentos, e não de

questões de infraestrutura da aplicação. O servidor de aplicações responde a algumas questões comuns a todas as aplicações, como segurança, garantia de disponibilidade, balanceamento de carga e tratamento de exceções.

Existem diversas regras de validação, tais como: horários disponíveis, possibilidade de agendamento de consulta, pesquisa de paciente por CPF, criação de usuário, entre outras. Todas as regras de validação da página *Web* estão rodando no servidor, desta forma todas as consultas passam obrigatoriamente pelo servidor. Este processo evita que o usuário tenha acesso direto às informações do banco de dados e que possa desenvolver suas próprias requisições (*queries*), mantendo assim a base mais segura contra pessoas mal intencionadas.

4 RESULTADOS

Esta seção apresentará os aplicativos, um manual de operação do mesmo, a página da *Web* desenvolvida, o sistema em funcionamento e uma estimativa de custos deste.

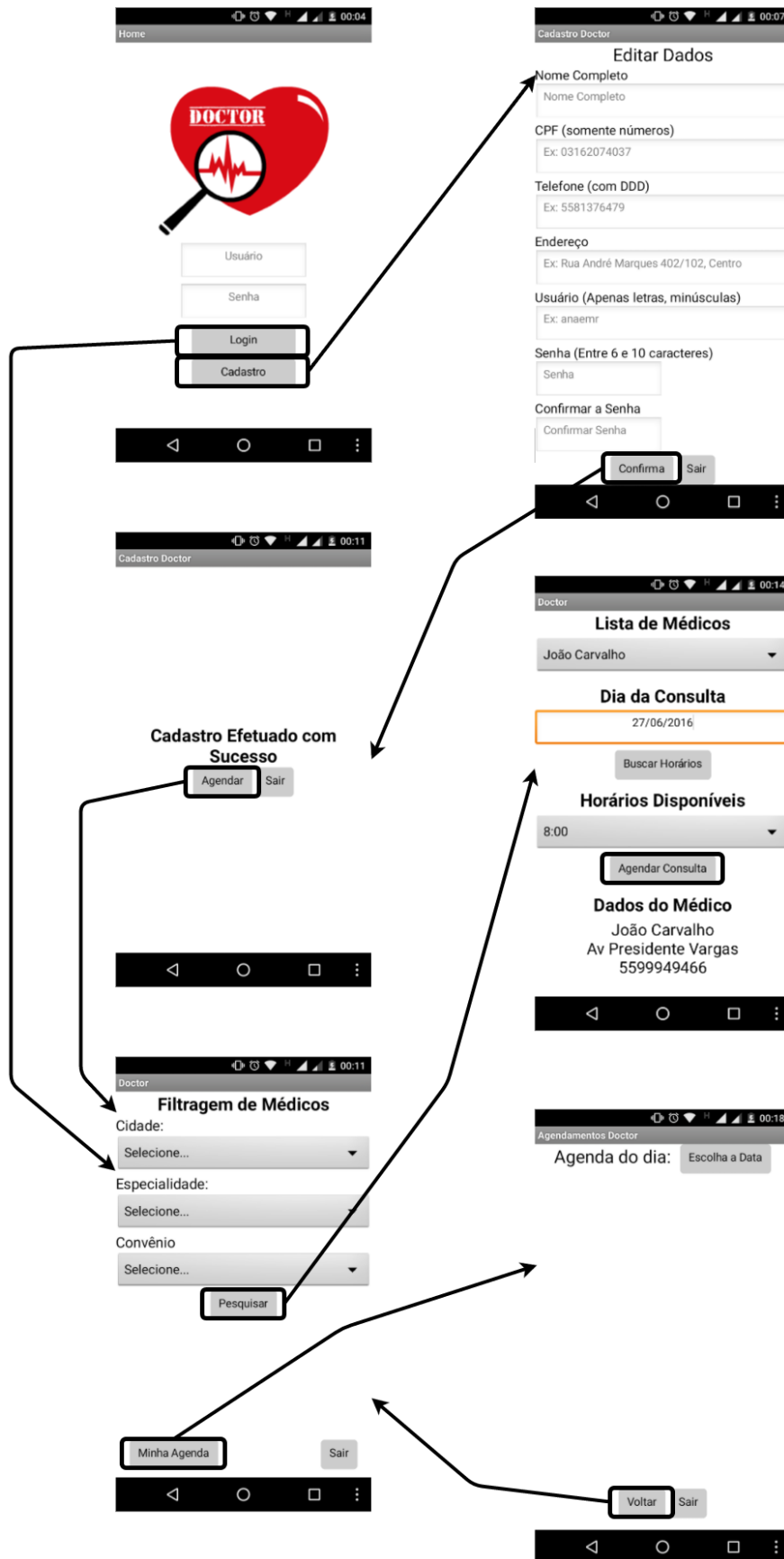
4.1 DOCTOR E DOCTOR FOR DOCTOR

Os programas desenvolvidos para o projeto foram chamados de Doctor (aplicativo para o paciente) e Doctor for Doctor (aplicativo para o médico). Ele faz uso de banco de dados disponíveis na plataforma Android.

4.1.1 Doctor

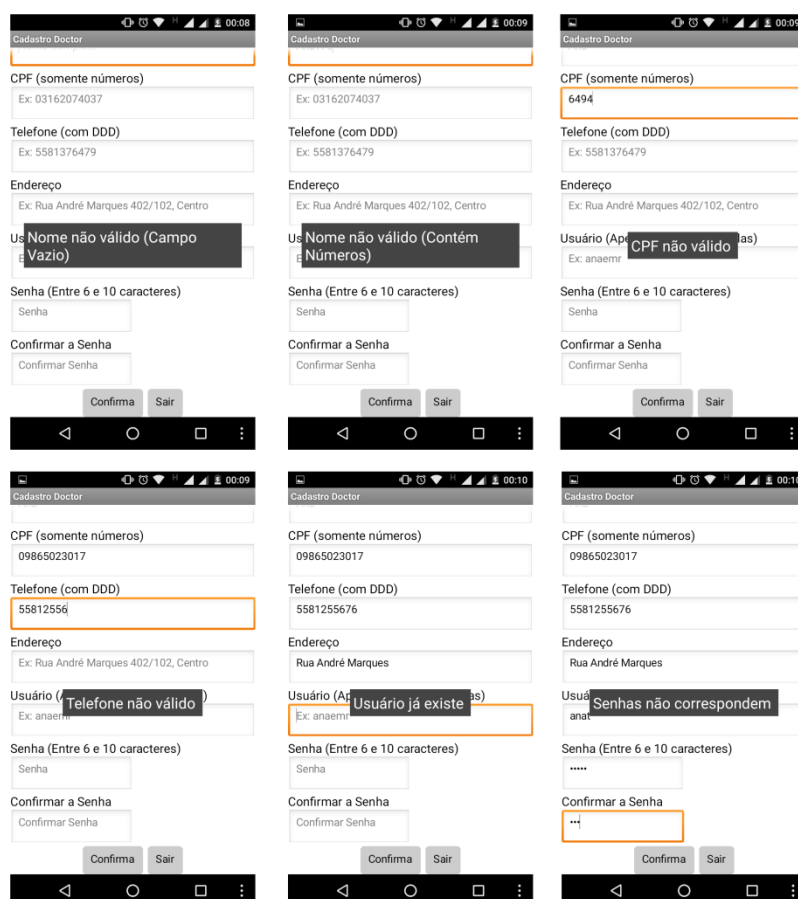
O aplicativo Doctor é composto por 6 (seis) telas, sendo elas: i. Tela Inicial; ii. Tela de Cadastro; iii. Tela de Cadastro Efetuado com Sucesso; iv. Tela de Filtragem de Médicos; v. Tela de Lista de Médicos; e vi. Tela de Minha Agenda. Todas estas telas e seus vínculos estão ilustrados na Figura 15.

Figura 15 - Fluxograma do Aplicativo Doctor



Caso um paciente não possua cadastro, ao clicar em “Cadastro” ele será redirecionado até uma tela onde insere todos os seus dados (Nome Completo, CPF, Telefone, Endereço, Usuário e Senha) e ao confirmar o cadastro o aplicativo faz uma série de testes para verificação dos dados. Se o usuário já existir no servidor, o paciente receberá uma mensagem o alertando disto e ele poderá alterar o usuário para prosseguir com o cadastro. Outros erros que podem acontecer e que serão alertados ao cliente são: campo vazio, números no nome, CPF inválido, número de dígitos diferente de 11 no CPF, número de dígitos menor que 10 no telefone e senha não correspondente. Estes erros são mostrados a seguir na Figura 16.

Figura 16 - Alerta de Erros no Cadastro



Se esta série de testes prosseguir sem erros, os dados são salvos no servidor. O servidor é a tabela do *Google Fusion Tables* que armazena todos estes dados *online*, como já foi explicado anteriormente. Para o caso do cadastro ter ocorrido com sucesso o paciente é redirecionado para a “Tela de Cadastro Efetuado com Sucesso”, e a partir deste momento ele

pode acessar o aplicativo, saindo desta tela e voltando para a Tela Inicial ou clicando no botão “Agendar” e partindo para a “Tela de Filtragem de Médicos”.

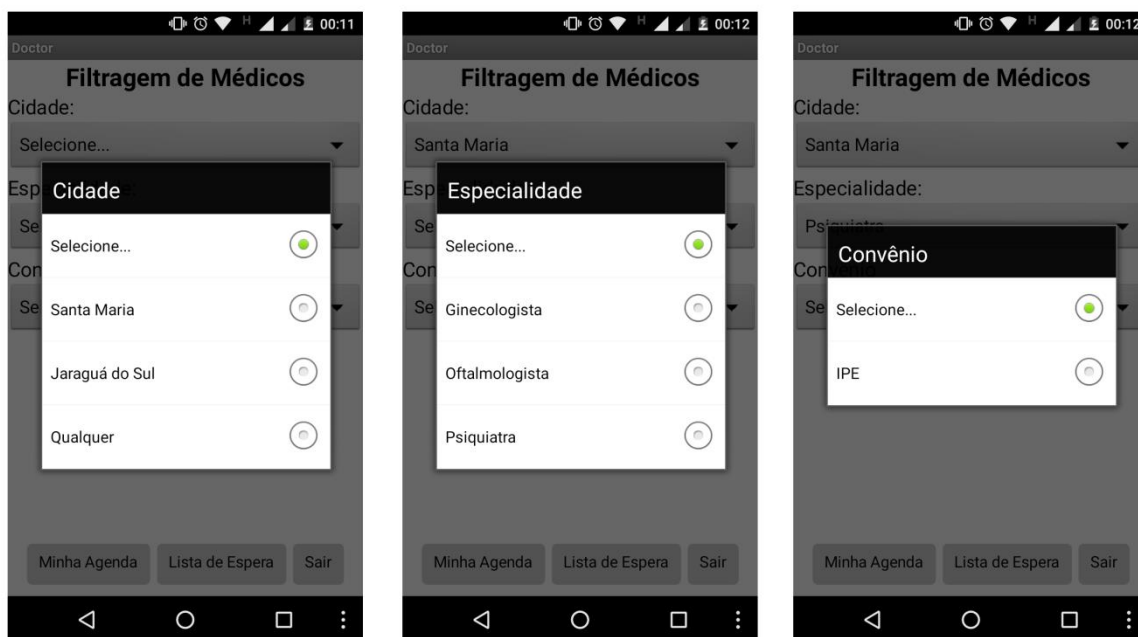
Se o usuário já está cadastrado, na tela inicial ele digitará seu *login* e senha. Caso o usuário insira um nome de usuário e uma senha não correspondente aos encontrados na tabela de usuários do *Google Fusion Tables* ou ocorra uma inconsistência na verificação usuário versus senha o aplicativo entrará em um fluxo alternativo de erro que retorna uma mensagem indicando a falha na autenticação (vide Figura 17).

Figura 17 - Erro de Usuário/Senha Inválido



Mas, se o *login* foi efetuado com sucesso, o paciente será redirecionado para a “Tela de Filtragem de Médicos”. Nesta tela, a partir dos médicos cadastrados no servidor, o usuário poderá escolher a sua cidade em uma lista de cidades, dessa escolha, uma busca das especialidades cadastradas nesta cidade será conduzida e mostrará em outra lista para o usuário que também optará pela de sua preferência. Após isso, o servidor encontrará quais os convênios dos médicos cadastrados nesta especialidade e o paciente poderá escolher o seu e clicar no botão “Pesquisar”. Estas listas da Tela de Filtragem de Médicos são apresentadas na Figura 18.

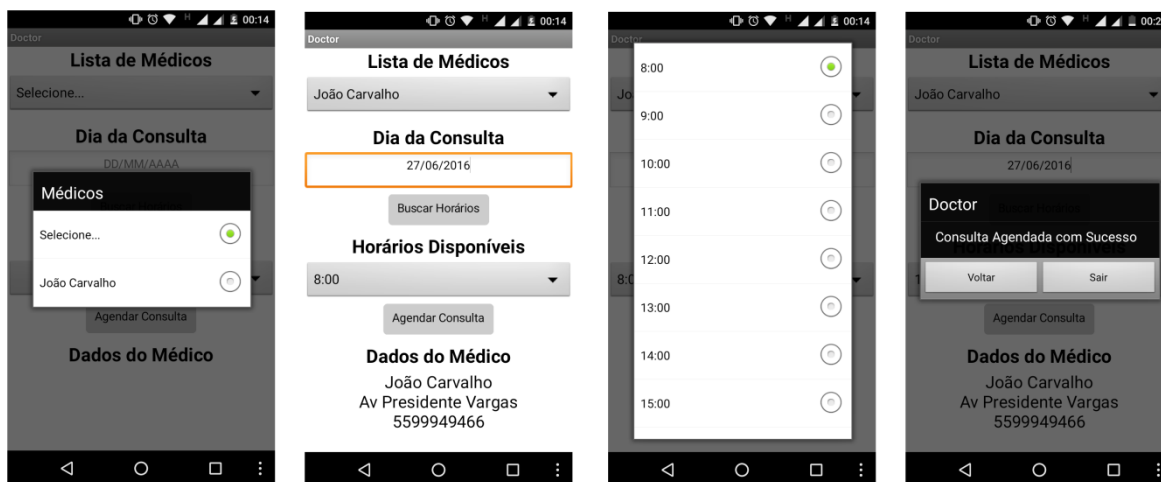
Figura 18 - Listas da Tela de Filtragem de Médicos



Ainda na “Tela de Filtragem de Médicos” o usuário pode escolher clicar no botão “Sair” e ser direcionado para a “Tela inicial” ou clicar no botão “Minha Agenda” e ser direcionado para a tela que pode visualizar suas consultas agendadas.

E por fim, após clicar no botão “Pesquisar” da “Tela de Filtragem”, o usuário será conduzido até a “Tela de Lista de Médicos” onde poderá escolher um médico na lista de médicos disponíveis. Assim que escolher o médico, as informações dele aparecerão automaticamente na tela. Logo após, o usuário pode digitar uma data de sua preferência e clicar no botão “Buscar Horários”. Se o paciente encontrar algum horário da lista que esteja de acordo com o que procura, basta clicar no horário e no botão “Agendar Consulta” e um aviso de que a Consulta foi agendada será mostrado na tela. Estas listas e avisos são ilustrados na Figura 19.

Figura 19 - Listas e Avisos da Tela de Lista de Médicos

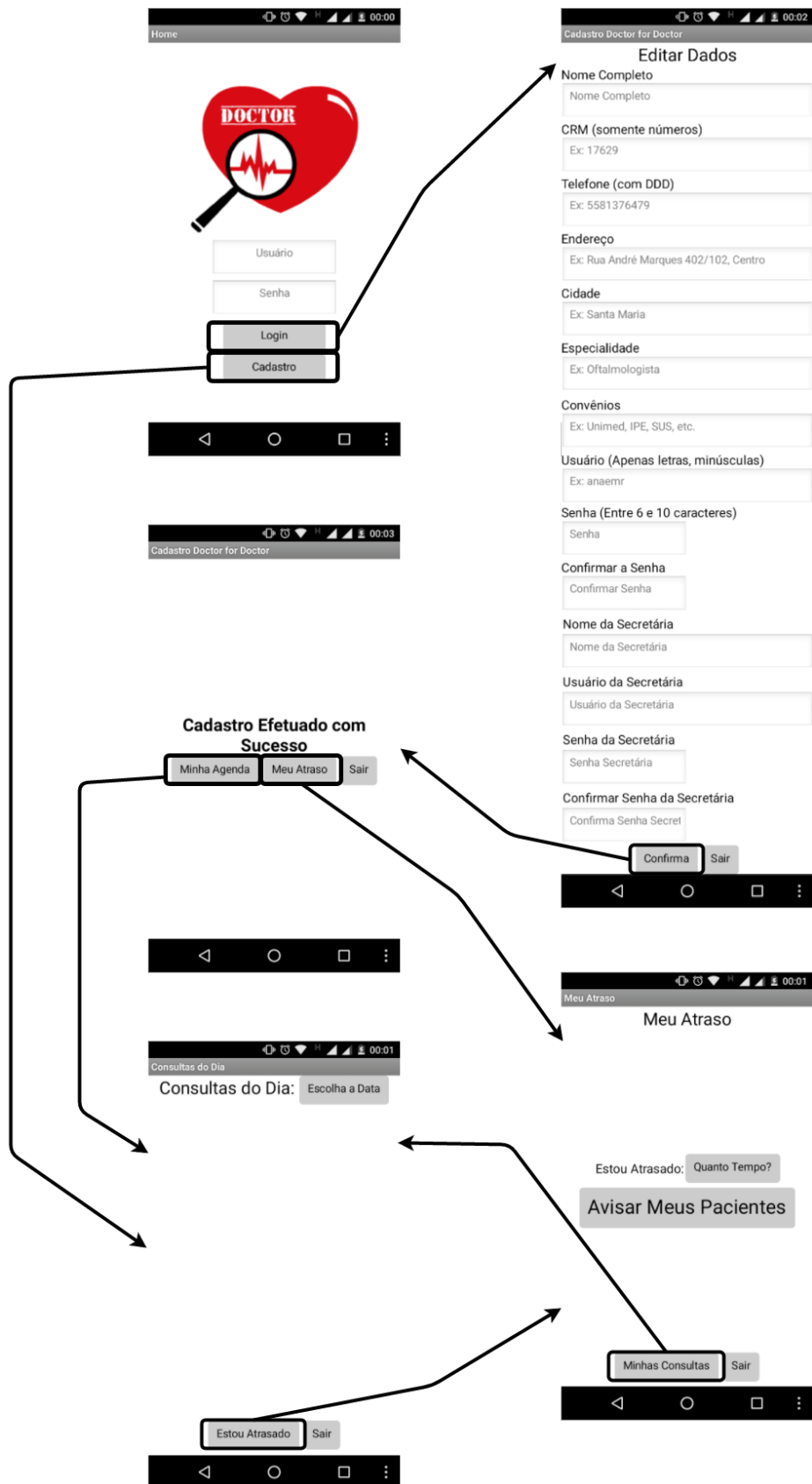


No caso de ser selecionada a opção “Sair” em qualquer uma das telas que possui essa opção, o usuário é redirecionado para a “Página Inicial”. Para sair do aplicativo basta clicar no botão “Voltar” do *smartphone* do cliente.

4.1.2 Doctor for Doctor

A aplicação Doctor for Doctor é composta por 5 (cinco) telas, sendo elas: i. Tela Inicial; ii. Tela de Cadastro; iii. Tela de Cadastro Efetuado com Sucesso; iv. Tela de Consultas do Dia; e v. Tela Meu Atraso. Todas estas telas estão esquematizadas no fluxograma da Figura 20 que apresenta a navegação nas telas e as funcionalidades do sistema.

Figura 20 - Fluxograma do Aplicativo Doctor for Doctor



Este aplicativo é muito similar ao anterior, apenas diferencia-se por:

- Compreender 8 (oito) campos a mais na Tela de Cadastro (CRM, Cidade, Especialidade, Convênios, Nome da Secretária, Usuário da Secretária, Senha da Secretária e Confirmação da Senha da Secretária)
- Apresentar duas telas a menos (Tela de Filtragem de Médicos, Tela de Lista de Médicos), e uma tela a mais (Tela de Meu Atraso).

A Tela Meu Atraso possibilita que o médico avise seus pacientes caso as consultas estejam atrasadas. Clicando no botão “Quanto tempo?” ele escolhe qual o seu atraso (vide Figura 21) e após clicar em “Avisar meu pacientes” os pacientes que têm consulta naquele dia recebem um aviso se que sua consulta vai atrasar, evitando que estes precisem ficar esperando no consultório.

Figura 21 - Escolha do tempo de atraso

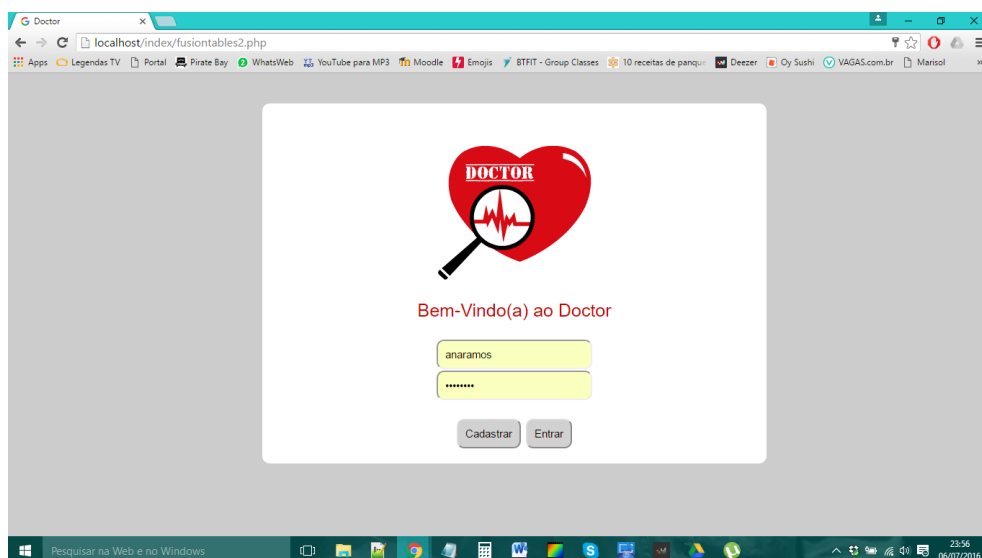


4.2 PÁGINA WEB DOCTOR

A página Doctor é um *site* que pode ser acessado pela assistente do médico de qualquer dispositivo que possua navegador *Web*. Na página inicial do *site* pode-se acessar a

agenda do médico através de um *login* e senha da secretária (vide Figura 22). Estes são associados ao *login* do médico, já que é ele que cria o cadastro de ambos como já visto no cadastro do *app* Doctor for Doctor.

Figura 22 - Tela Inicial da Página Web



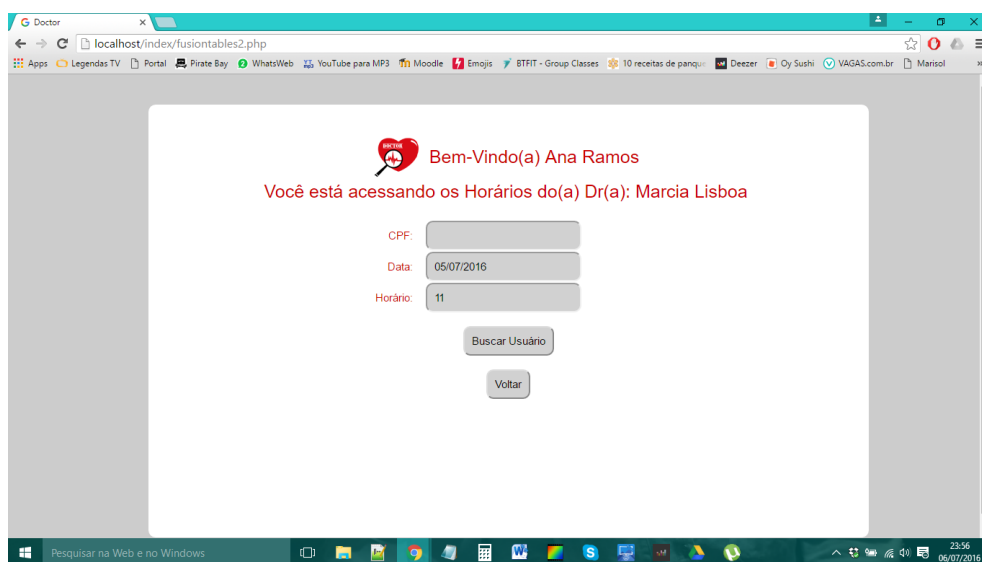
Como mencionado anteriormente, para completar o sistema de agendamento foi criada esta interface *Web* para que a secretária do médico tivesse acesso à agenda do mesmo, permitindo assim que ela faça a gestão dos horários, através da edição, confirmação, correção de consultas e cadastro de novos pacientes. Podemos visualizar esta interface na Figura 23.

Figura 23 - Agenda de um Médico cadastrado



Caso a secretária precise marcar algum horário pelos métodos convencionais (ligação telefônica ou pessoalmente) basta clicar em no *link* “Horário Livre” da preferência de data e hora do paciente, e a página será direcionada para uma busca de pacientes por CPF (Figura 24).

Figura 24 - Marcação de horário através da busca por CPF



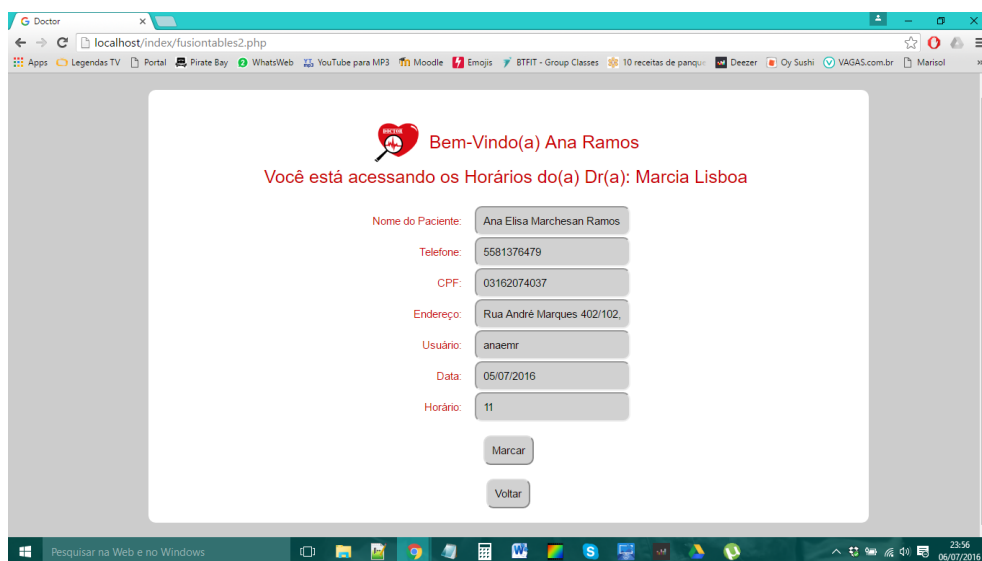
The screenshot shows a web browser window with the URL `localhost/index/fusiontables2.php`. The page displays a welcome message: "Bem-Vindo(a) Ana Ramos" and "Você está acessando os Horários do(a) Dr(a): Marcia Lisboa". Below this, there is a search form with the following fields and values:

Field	Value
CPF:	<input type="text"/>
Data:	05/07/2016
Horário:	11

Below the form are two buttons: "Buscar Usuário" and "Voltar".

Caso o paciente possua cadastro no Doctor, seus dados vão ser exibidos para o usuário do *site* e este poderá marcar a consulta clicando no botão “Marcar” (vide Figura 25).

Figura 25 - Marcação de horário para usuário localizado por CPF



The screenshot shows a web browser window with the URL `localhost/index/fusiontables2.php`. The page displays a welcome message: "Bem-Vindo(a) Ana Ramos" and "Você está acessando os Horários do(a) Dr(a): Marcia Lisboa". Below this, there is a search form with the following fields and values:

Field	Value
Nome do Paciente:	Ana Elisa Marchesan Ramos
Telefone:	5581376479
CPF:	03162074037
Endereço:	Rua André Marques 402/102,
Usuário:	anaemr
Data:	05/07/2016
Horário:	11

Below the form are two buttons: "Marcar" and "Voltar".

Caso o paciente não possua cadastro, a secretária terá a opção de ser direcionada para uma página afim de que o faça (vide Figura 26) para depois prosseguir com o agendamento da forma descrita previamente.

Figura 26 - Cadastro de paciente novo no Doctor

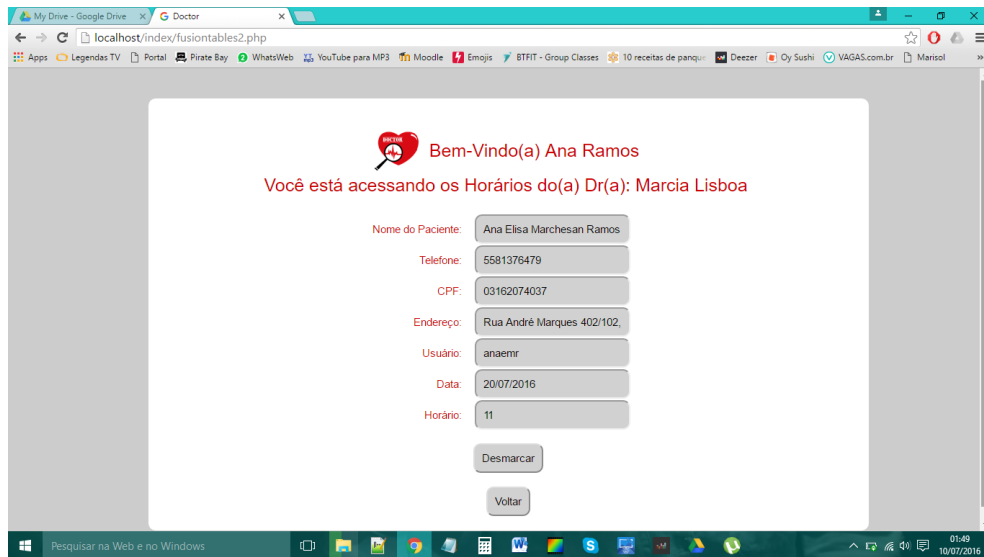


The screenshot shows a web browser window with the following content:

- Browser title: Doctor
- Address bar: localhost/index/fusiontables2.php
- Page content:
 - Logo: A red heart with a white cross and the text "MEDI" above it.
 - Greeting: Bem-Vindo(a) Ana Ramos
 - Message: Você está acessando os Horários do(a) Dr(a): Marcia Lisboa
 - Form fields:
 - Nome do Paciente:
 - Telefone:
 - CPF:
 - Endereço:
 - Usuário:
 - Senha:
 - Confirme a senha:
 - Buttons: Cadastrar, Voltar

Se por algum motivo for necessário o acesso aos dados do paciente, ou ainda, que se desmarque alguma consulta, a secretária tem acesso a este tipo de informação apenas clicando no *link* do nome do paciente na agenda. Assim, o *site* será direcionado para uma página com os dados do paciente, e se quiser cancelar sua consulta basta clicar no botão “Desmarcar” (vide Figura 27).

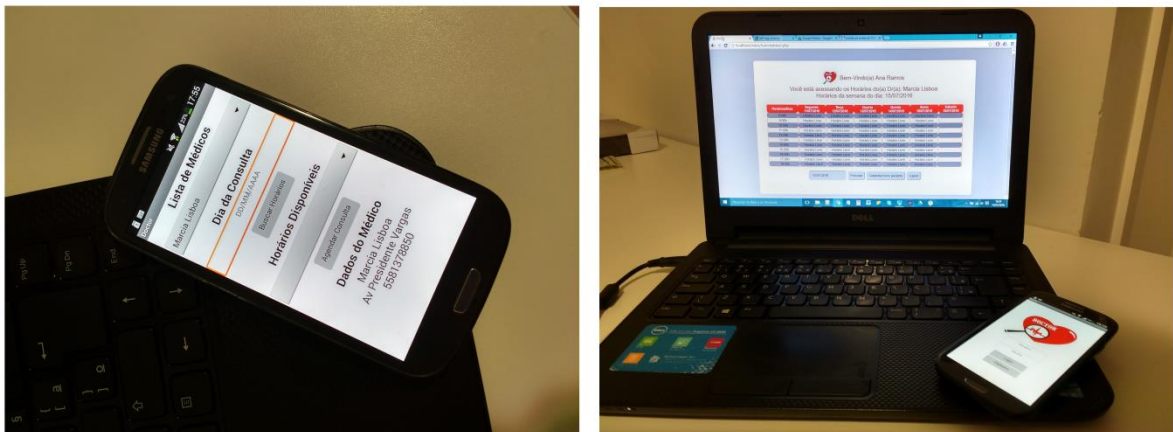
Figura 27 - Possibilidade de cancelamento de consulta pela interface Web



4.3 APRESENTAÇÃO DOS PROTÓTIPOS

Os protótipos deste projeto são os próprios aplicativos e a página *Web* em pleno funcionamento, que foram apresentados detalhadamente na seção anterior. A página *Web* será futuramente acessada através do *link*, e os aplicativos serão, eventualmente disponibilizados para *download* na *Play Store*, mas, já estão disponíveis os arquivos “.apk” de ambos para serem instalados em dispositivos com sistema operacional Android, como podemos ver na Figura 28.

Figura 28 - Dispositivos que compõe o Sistema de Agendamento Médico



4.4 ESTIMATIVA DE CUSTOS

Para comprovar a viabilidade econômica do trabalho é necessário que se faça uma análise de custos do projeto. Como o projeto só necessitou de um computador pessoal para a programação dos aplicativos e da página, e de um *smartphone* para os testes, o custo total do sistema será principalmente composto pelas horas de programação.

De acordo com a Tabela 2, o valor Hora/Homem de Programação é de R\$ 50,00. E, de acordo com a Tabela 3 foram trabalhadas 312 horas de programação, gerando um custo total de R\$ 15.600,00. É importante lembrar que os valores da Tabela 2 são baseados em pesquisa realizada em BH (Minas Gerais) em Março de 2012 (FONSECA, 2012) e devem ser utilizados apenas como referência.

Tabela 2 - Tabela de Valores Médios Praticados por Freelancers de TI

Serviço	Mín. de Horas	Valor (R\$)	Descrição
Reinstalação de Servidor LINUX/Windows	2	90	Instalação em até 02 horas, acima disso valor adicional hora/homem suporte, proporcional ao número de horas utilizadas
Instalação em Servidor LINUX	1	130	Instalação em até 02 horas, acima disso valor adicional hora/homem suporte, proporcional ao número de horas utilizadas
Valor Hora/Homem Suporte	1	45	Valor hora/homem para suporte
Atendimento Suporte fora de horário	1	65	Valor hora/homem – atendimento excepcional for a do horário comercial. Fins de semana e feriados: R\$90.00 h/h.
Manutenção em Base de Dados	1	50	Serviços de limpeza de registros, corrupção ao banco de dados, rotinas de SQL, scripts para atualização de dados em lote, importação de cadastros
Treinamento	2	90	Treinamento para uso de sistemas e ferramentas
Valor Hora/Homem Programação	1	50	Valor hora/homem para programação/desenvolvimento e

			integrações
Valor Hora/Homem Programação II	2	80	Valor hora/homem para desenvolvimento de relatórios, layout de NF e formulários específicos

Tabela 3 - Cálculo de Horas de Programação

Semanas Trabalhadas	Dias Trabalhados				
	Segunda	Terça	Quarta	Quinta	Sexta
06/04/15 a 26/06/15	1 hora	2 horas	1 horas	1 horas	2 horas
07/09/15 a 18/12/15	-	2 horas	2 horas	-	3 horas
04/01/16 a 19/02/16	1 hora	-	2 horas	1 hora	-
22/02/16 a 01/07/16	-	1 hora	1 hora	1 hora	2 horas
Total: 53 Semanas	Média: 6 Horas/Semana			Total: 312 Horas	

A fim de pagar o valor das horas de programação, é estimado que será cobrado pela Google Play R\$ 50,00 para instalação dos aplicativos dos médicos. Desta forma, quando 312 médicos tiverem instalado o aplicativo (Doctor for Doctor), este terá seus custos de desenvolvimento pagos. O aplicativo do usuário é sempre gratuito. Para utilizar o sistema o médico deverá pagar uma taxa de manutenção do sistema de maneira mensal, isso também dará acesso a interface *Web*.

Como evolução do modelo de negócio é necessário avaliar também o uso de propaganda voltada a área de saúde dentro do aplicativo.

5 CONCLUSÃO

A evolução da tecnologia vem mudando a maneira como interagimos entre nós, com o aumento do uso de recursos computacionais em todas as áreas da sociedade. Esse avanço está mostrando novas formas de fazermos tarefas cotidianas de uma maneira mais simples e eficiente.

É evidente que alguns setores da sociedade ainda não estão completamente integrados com o mundo digital, ou “conectados”, porém a criação de novos sistemas mais amigáveis e de fácil adesão são os aceleradores necessários para completar inclusão.

Durante o trabalho foi proposto um sistema para auxiliar profissionais da área da saúde e seus usuários a terem uma interação mais rápida no agendamento de consultas. O sistema mostrou como pode ser utilizado um dispositivo cotidiano, como o celular, para encontrarmos dentro de uma infinidade de médicos aquele que preenche de maneira mais assertiva nossos requisitos de disponibilidade, especialidade, localização, plano de saúde, entre outros.

Uma das vantagens do sistema mostrado no trabalho é permitir que mais de uma pessoa consulte ao mesmo tempo a agenda de um médico e que estas possam fazer o agendamento de maneira quase paralela. Este comportamento seria impossível em um consultório atual que utiliza apenas o recurso de uma assistente fazendo apontamentos em uma agenda.

Visando garantir que o médico possa testar o sistema sem grande investimento inicial toda a aplicação foi concebida para rodar a partir de um servidor de aplicação remoto juntamente com um banco de dados na nuvem. Além de reduzir drasticamente o investimento isso permite com que o médico tenha uma solução escalável com uma infinidade de pacientes podendo ser cadastrados e tudo isso a um custo fixo. Do ponto de vista do paciente o sistema também foi pensado para ter fácil adesão sendo que pode ser utilizado de forma gratuita em seu celular.

Os conceitos aqui abordados podem ser extrapolados para outras aplicações do mesmo gênero, pode-se por exemplo ser utilizado em um *petshop* ou salão de beleza para marcar horários. Com o trabalho é possível concluir que um sistema com banco de dados *online*, onde todas as pessoas podem acessar informações úteis de maneira atualizada, é uma evolução natural na forma de desenvolver aplicações voltadas a um grande número de usuários descentralizados e com baixo investimento inicial.

REFERÊNCIAS

- ABITEBOUL, S. Querying Semi-Structured Data. ICDT, 1997.
- CARVALHO, José Crespo de; ENCANTADO, Laura - Logística e negócio electrónico [Em linha]. Porto, SPI - Sociedade Portuguesa de Inovação, Consultadoria Empresarial e Fomento da Inovação, S.A., 2006.
- CORREA, Gunnar. Definição de: JavaScript. SatellaSoft, 2015. Disponível em: <http://satellasoft.com/?categoria=javascript>. Acesso em: 29 mai. 2016.
- COSTA, Rachel. Um basta ao atraso do médico: Reação às prolongadas esperas nos consultórios obriga os médicos a adotar medidas de compensação pelo tempo perdido. [Medicina & Bem Estar]. Isto É, Edição: 2182, Set, 2011. Disponível em: <http://goo.gl/9gsvhA>. Acesso em: 16 jul. 2015.
- IDC. Smartphone OS Market Share, 2015 Q2. Disponível em: <http://www.idc.com/prodserv/smartphone-os-market-share.jsp>. Acesso em: 4 jul. 2016.
- FEITOSA, Eduardo. O que é HTML 5. Devmedia, 2012. Disponível em: <http://goo.gl/oxWuky>. Acesso em: 02 jun. 2016.
- FONSECA, Gabriela. Tabela de Valores Médios Praticados por Freelancers de TI [Carreira]. Profissionais TI, Abr, 2012.
- FRAGA, Rene. Google Fusion Tables: Compartilhe e Visualize Grandes Conjuntos de Dados. São Paulo: Discovery, 2009.
- FURMAN, J; KARLSSON, J. S; LEON, J.-M; LLOYD, A; NEWMAN, S; ZEYLIGER, P. Megastore: A Scalable Data System for User Facing Applications. SIGMOD, 2008.
- GONZALEZ, H; HAVELY, A; JENSEN, C; Google Fusion Tables: Web-Centered Data Management and Collaboration. SIGMOD, 2010.
- GREENBERG, Paul. CRM, Customer Relationship Management na Velocidade da Luz: conquista e lealdade de clientes em tempo real na *Internet*. Rio de Janeiro: Campus, 2001. Tradução de CRM at the Speed of Light, McGraw-Hill Companies, Inc, 2001.
- GROSS, D., Harris. C. M. Fundamentals of Queueing Theory. Willey series in Probability and Mathematical Statistics, New York 1974.

HAVELY, Alon. Google Fusion Tables. Google Research Blog, Jun, 2009.

JAGADISH, H. V; CHAPMAN, A; ELKISS, A; JAYAPANDIAN, M; LI, Y; NANDI, A; YU, C. Making database systems usable. SIGMOD, 2007.

MADHAYAN, Jayant; SHAPLEY, Rebecca; SHEN, Warren. Google Fusion Tables: Data Management, Integration and Collaboration in the Cloud. SIGMOD, 2012.

PEREIRA, Ana Paula. O que é CSS. Tecmundo, 2009. Disponível em: <http://goo.gl/4Yb89g>. Acesso em: 27 mai. 2016.

RAMAKRISHNAN, R; GEHRKE, J. Database Management Systems. McGraw Hill, 2002.

RAYAN, César. Servidores de Aplicação Web. Prezi, 2014. Disponível em: <https://prezi.com/jyg0voeacblv/servidores-de-aplicacao-web>. Acesso em: 14 mar. 2016.

SOUZA, Laura. ZTE passa RIM e venda de *smartphones* cresce 74% no 2º tri [Tecnologia]. Terra, Ago, 2011.

STROPARO, Elder. Novo banco de dados *online*, Google Fusion Tables. São Paulo, 2009. Disponível em: <http://goo.gl/N4QhbB>. Acesso em: 20 jun. 2015.

TESTA, Mauricio; LUCIANO, Edimara; FREITAS, Henrique. Comércio Eletrônico: Tendências e Necessidades de Pesquisa. Rio de Janeiro: Revista Angrad Volume 7 Número 1, 2006. Disponível em: <https://goo.gl/Zijec7>. Acesso em: 5 mai. 2015.

ZIMMER, Ben; BARRETT, Grant; METCALF, Allan. “App” 2010 Word of the Year, as voted by American Dialect Society. American Dialect Society, Jan, 2011.

APÊNDICE A – COMUNICAÇÃO VIA GOOGLE FUSION TABLES

A comunicação entre os aplicativos se dá via *Google Fusion Tables*, que oferece uma plataforma para visualização de grandes conjunto de dados, uma nova opção para empresas que tentam interesse no compartilhamento de banco de dados na nuvem computacional do Google. Diferente do Google Spreadsheets, que apresenta atualmente suporte a pequenas planilhas, o *Google Fusion Tables* tem capacidade para trabalhar com enormes estruturas. Durante a fase de testes, o serviço está pronto para executar planilhas com limites de 100 MB por tabela e 250 MB por usuário. Além de trazer uma visualização instantânea, os dados são exibidos em um mapa ou gráfico. Colunas com localizações são interpretadas automaticamente e os usuários podem fazer ajustes diretamente no mapa, se necessário. Também é possível usar filtros e agregar ferramentas para visualizações mais seletivas.

Assim como os produtos existentes do Google, o *Google Fusion Tables* oferece uma gama completa de ferramentas para convidar colaboradores para visualização e edição, opção para mesclar dados com outras tabelas de dados e interação *online* entre toda a equipe. Todas as mudanças são registradas e podem ser recuperadas.

O *Google Fusion Tables* ainda adiciona novos recursos que permitem selecionar o conteúdo a ser compartilhado, uma forma de direcionar tarefas relevantes e reduzir a visualização de dados irrelevantes. Outro ponto interessante está na possibilidade de especificar uma atribuição a dados específicos, uma sinalização importante ao mesclar os dados com outras planilhas. Os mapas e gráficos, dependendo do interesse da empresa, podem ser incorporados a *Web sites e blogs*. (FRAGA, 2009)

O *Google Fusion Tables* pode ser considerado um banco de dados SQL já que aceita requisições nesta linguagem, porém a principal diferença entre um banco de dados SQL na nuvem (*cloud SQL*) e o *Fusion Tables* (que é considerado *noSQL*) é a possibilidade de tratar grandes volumes de dados. Isso é feito através de técnicas de divisão de tabelas e posteriormente a fusão delas.

Em um exemplo prático é possível analisar uma busca de médicos por cidade. Supondo que existem muitos médicos em cada cidade o melhor é utilizar uma tabela para cada cidade, assim a busca é feita apenas nos médicos da cidade selecionada. Porém quando é feita uma requisição de busca em múltiplas cidades o procedimento acaba possuindo um custo de processamento muito alto, pois em uma *cloud SQL* a requisição é disparada para um servidor, rodando uma instância de banco de dados com todas as tabelas. Para uma busca sem

mais *scripts* customizados pelo usuário é necessária a fundição das tabelas de todas as cidades, gerando uma tabela com alto volume de dados.

No caso do *Fusion Tables* é possível fazer uma requisição para uma tabela virtual que é gerada pela fusão das tabelas das cidades. Na prática o servidor do *Fusion Tables* recebe uma *query* (solicitação) para uma tabela e dispara a busca para várias instâncias onde estão rodando as tabelas que geram a tabela virtual, com isso o tempo de busca se torna reduzido. Caso seja possível manter o tamanho das tabelas constante o tempo de busca é também constante mesmo com a adição de mais tabelas em “paralelo”.