



16º Congresso Nacional de Iniciação Científica

TÍTULO: SISTEMA WEB GERENCIADOR DE CÉLULAS VEM VÊ

CATEGORIA: CONCLUÍDO

ÁREA: CIÊNCIAS EXATAS E DA TERRA

SUBÁREA: COMPUTAÇÃO E INFORMÁTICA

INSTITUIÇÃO: FACULDADE DE TECNOLOGIA DE CARAPICUIBA

AUTOR(ES): EDUARDA NOGUEIRA TAVARES, ANDERSON DIAS TEIXEIRA, EDSON ARTHUR DE SOUZA PIMENTEL, JONES CASEMIRO DE LIMA, OSVALDIMAR COSTA

ORIENTADOR(ES): ANDREIA CRISTINA GRISOLIO MACHION, RODRIGO BOSSINI TAVARES MOREIRA, SILVIA MARIA FARANI COSTA

Realização:



Apoio:



RESUMO

Vem Vê é um convite feito muitos anos atrás refletido nos dias atuais em igrejas que trabalham no Modelo de Discipulado Apostólico (MDA). Nesse modelo o foco é ajudar pessoas a caminharem com Jesus, seguindo os princípios bíblicos ensinados por Ele. O sistema Vem Vê proposto visa auxiliar na automatização do gerenciamento de células, na coleta e manutenção dos dados, por meio de uma aplicação web, acessível também por dispositivos móveis, funcionalidade que visa facilitar acesso e contato com os visitantes em até 48h.

INTRODUÇÃO

A incessante busca por bem-estar, em todas as áreas da vida, tem norteado as ações da sociedade. Uma delas é a área espiritual, o que justifica o crescimento de instituições religiosas no Brasil, de acordo com o último censo realizado (IBGE, 2010).

Tais instituições vêm se modificando e moldando suas ações para estreitar seu relacionamento com as pessoas que as frequentam. Algumas delas têm como visão de trabalho o modelo de discipulado apostólico (MDA), que proporciona um ambiente de intimidade, para o desenvolvimento da amizade como um dos princípios básicos.

Para que esse ambiente aconteça, a instituição religiosa que atua de acordo com esse modelo, chamada de igreja local, se estrutura em células (grupos menores de 6 a 12 pessoas) que se reúnem uma vez por semana com dia, horário e local marcados. A estrutura hierárquica da visão celular é organizada por quatro redes, sendo elas: verde, vermelha, amarela e azul. Cada rede possui setores compostos por duas ou mais células e cada célula possui um líder e um líder em treinamento, para organiza-la e auxiliar seus membros no crescimento espiritual.

Um dos objetivos dessas reuniões é orientar, auxiliar e acompanhar pessoas em um estilo de vida, seguindo os ensinamentos de Jesus, encontrados na Bíblia Cristã. A ideia resume-se em cuidar bem das pessoas, visando o seu bem-estar espiritual, fazendo com que, elas, por sua vez, disseminem o que aprenderam a outros indivíduos do seu círculo de relacionamento, convidando-os a experimentarem o mesmo conjunto de práticas que encontraram na sua célula. Dessa forma, o grupo de pessoas que se reúnem num lar semanalmente aumentará

o que possibilita a multiplicação das células e assim, conseqüentemente, o crescimento da igreja local.

A fim de convidar as pessoas a participarem dos encontros realizados nas células e reuniões que acontecem no prédio da igreja local, normalmente aos fins de semana, existe uma preocupação com a maneira com que os convidados são recepcionados: em primeiro lugar, é preciso verificar se a reunião foi boa para cada pessoa sobre vários aspectos (local físico, pessoas encontradas, ambiente, como foi recepcionado). Também é importante fazer novo contato em no máximo 48h. Por conta disso, essas reuniões contam com um grupo de recepção, membros de células da igreja local, que procuram recolher algumas informações para contato com cada convidado utilizando um formulário próprio.

Esses dados são encaminhados ao líder de uma célula mais próxima do convidado, que encaminha, por sua vez, para algum membro, ou ele mesmo entra em contato, utilizando os meios de comunicação disponíveis. Porém, muitas vezes não é possível repassar essas informações do formulário preenchido para algum líder em 48h, o que atrasa o contato com o convidado.

Com base nesse cenário, este projeto visa desenvolver um sistema que auxilie não apenas o gerenciamento dos dados coletados de cada pessoa, mas também facilite e automatize a gestão de todos os processos envolvidos, seja no controle de usuários, seja no acompanhamento das atividades relacionadas à igreja. Propõe-se um sistema dividido em dois módulos, o primeiro módulo é uma aplicação web que visa o gerenciamento de membros, células, setores, líderes e visitantes; e o segundo módulo, para dispositivos móveis é composto pelo gerenciamento de visitantes com endereço próximo de alguma célula para que uma visita possa ser agendada e o contato feito dentro de 48h.

OBJETIVO GERAL

Desenvolver e documentar um sistema de gerenciamento de instituições religiosas que trabalham em células, bem como agendamento de visitas entre membros e visitantes da instituição, mantendo informações básicas sobre os membros ativos e possibilitando o contato em 48h com o visitante de uma célula ou de uma reunião de domingo

OBJETIVOS ESPECÍFICOS

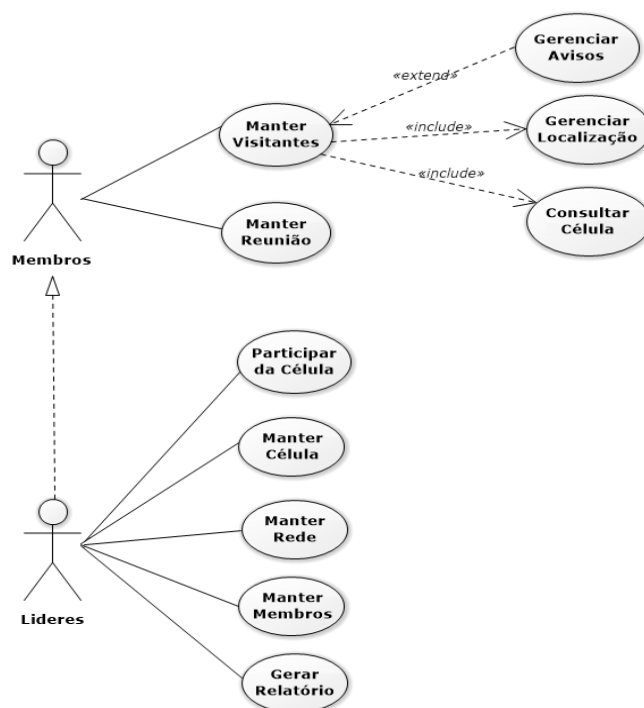
A fim de cumprir o objetivo geral proposto neste projeto, tem-se os seguintes objetivos específicos:

- Oferecer uma interface web para gerenciamento de membros, células, setores, redes, visitantes e líderes;
- Permitir que cada usuário (líderes e membros) tenham acessos específicos para a utilização do sistema;
- Disponibilizar um banco de dados suporte para a aplicação;
- Oferecer uma interface para dispositivos móveis para gerenciamento de visitantes com endereço próximo de alguma célula para que uma visita possa ser agendada e o contato feito dentro de 48h.

PROJETO TÉCNICO

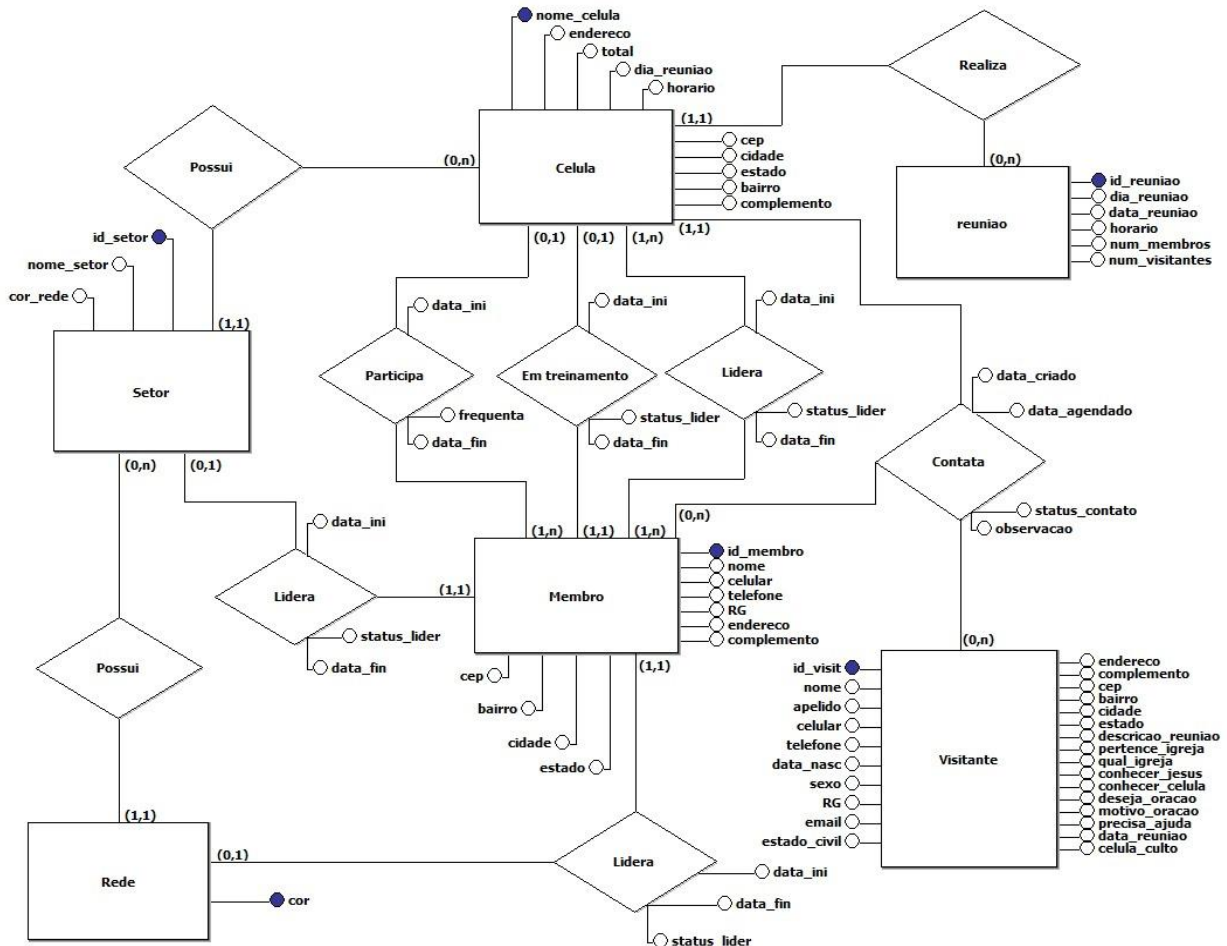
Para este projeto partiu-se da análise do modelo MDA e sua aplicação a uma igreja, bem como sua modelagem para um sistema web. A seguir estão ilustradas suas principais características por meio de um diagrama de casos de uso (Figura 1) (BOOCH, RUMBAUGH e JACOBSON, 2005) e da modelagem dos dados, tanto para o módulo web, quanto para o módulo para dispositivos móveis.

Figura 1 – Diagrama de Caso de Uso



Neste projeto, o modelo conceitual de dados utilizado foi o Modelo Entidade Relacionamento (HEUSER, 1998). A Figura 2 mostra o Diagrama Entidade Relacionamento dos dados.

Figura 2 – Diagrama de Modelo Entidade e Relacionamento



FERRAMENTAS E MÉTODOS

O ambiente de desenvolvimento empregado foi o Eclipse Java EE IDE versão Luna, por apresentar uma grande comunidade de desenvolvedores que o utilizam, com bom suporte, inclusive para diferentes *plug-ins* como o SVN (THE ECLIPSE FOUNDATION, 2012), utilizado para a comunicação com o GitHub, um repositório utilizado para publicar e baixar os códigos alterados por todos os integrantes do projeto.

Para o banco de dados, foi utilizado como Sistema de Gerenciamento de Banco de Dados o MySQL 5.7, por ser *open source* e estar disponível sob a licença

GPL (Licença Pública Geral, designação da licença para *software* livre) e é apoiado por uma comunidade grande e ativa de desenvolvedores de código aberto. A linguagem de programação utilizada, devido sua aceitação pelos mais diversos dispositivos e portabilidade, é o Java 8 (DAVIS, 2015). Como servidor responsável pela execução da aplicação optou-se pelo Apache Tomcat 8 (APACHE SOFTWARE FOUNDATION, 2014), devido à sua capacidade de implementar as tecnologias Java Servlet e JavaServer Pages.

Para a definição de dependências entre as classes Java e o desenvolvimento da aplicação web em módulos, foi utilizada a metodologia Spring Framework 4.1.6 (GENTIL, 2012), também *open source*. Outra ferramenta utilizada foi o Bootstrap 3.3.6 (COSTA, 2014), um *framework* popular para HTML e CSS, utilizado para desenvolvimento *front end*, tornando as páginas *web* responsivas, independente do tamanho da tela do dispositivo.

Para hospedagem da aplicação web desenvolvida, utilizamos o OpenShift desenvolvido pela Red Hat, uma solução baseada na categoria de computação em nuvem chamado *Platform-as-a-Service* (PaaS). Isso permite aos clientes desenvolver, executar e gerenciar aplicativos sem a complexidade de construção e manutenção da infraestrutura. O OpenShift disponibiliza a hospedagem de até 3 aplicações com espaço de armazenamento reduzido, porém de forma escalável e sem custos.

Para realizar a funcionalidade de geolocalização, utilizou-se o serviço gratuito da Google Maps chamado Distance Matrix API, fornece a distância e o tempo de percurso para uma matriz de origens e destinos. As informações retornadas são baseadas na rota entre os pontos de partida e chegada. Essa ferramenta disponibiliza *web services* que podem ser acessados por meio de sua URL, utilizando o método GET do protocolo HTTP. Para o retorno, optou-se pela formatação JSON, um modelo de objeto em JavaScript de fácil utilização e leve para interpretar e gerar código. A partir das informações retornadas de distância e tempo dos endereços traçados, pode-se calcular qual célula estará mais próxima de um visitante.

O Quadro 1 mostra o trecho de código de implementação usada para acessar o web service da Google Maps Distance Matrix API.

Quadro 1 - Implementação API Google Maps

```
/**
 * metodo acessa o web services do google maps e retorna uma String com dados em json
 */
public String calcularDistanciaGoogleApis(String enderecoOrigem, String enderecoDestinos) {

    //exemplo: https://maps.googleapis.com/maps/api/distancematrix/json?
    //origins=Seattle&destinations=San+Francisco|New+York&key=AizaSyCXhk3-X2KZVP5GjpUqdzns4TcDUiKftbo

    StringBuilder builderJson = new StringBuilder();
    try {
        String key = "AIzaSyCXhk3-X2KZVP5GjpUqdzns4TcDUiKftbo"; //chave de acesso gerada no Google Developers Console
        String urlMaps = "https://maps.googleapis.com/maps/api/distancematrix/json?" //url api google + enderecos origem e destinos
            + "origins="+URLEncoder.encode(enderecoOrigem, "utf-8")
            + "&destinations="+URLEncoder.encode(enderecoDestinos, "utf-8")+ "&" +key;

        URL url = new URL(urlMaps);
        HttpURLConnection conexao = (HttpURLConnection) url.openConnection(); //abre conexao com web services google
        conexao.setRequestProperty("Content-Type", "application/json");
        conexao.setRequestMethod("GET");
        conexao.setDoInput( true );
        conexao.connect();

        BufferedReader in = new BufferedReader(new InputStreamReader(conexao.getInputStream())); //ler dados de entrada
        String inputLine;
        while ((inputLine = in.readLine()) != null) {
            builderJson.append(inputLine);
        }
        in.close();
    } catch (IOException e) {
        e.printStackTrace();
    }
    return builderJson.toString(); //retorna String com dados em json
}
```

O sistema foi projetado para acesso em computadores por meio da aplicação web e dispositivos móveis que utilizem o Android como sistema operacional. Para isso foi utilizado o Android Studio (ANDROID STUDIO DEVELOPERS, 2013), como ambiente integrado de desenvolvimento para dispositivos móveis, pois fornece arquivos dentro do diretório SDK, possibilitando maior flexibilidade ao processo de construção.

A ferramenta faz uso do recurso Injection Language, pois permite que *strings* de outras linguagens sejam validadas pela IDE como SQL, XML, HTML entre outros, e possibilita a execução de testes da aplicação em diferentes tipos de dispositivos.

Levando em consideração que a aplicação web tem seu *layout* responsivo, optou-se por apresentar a aplicação em uma aplicação para dispositivos móveis utilizando o recurso WebView (THIENGO, 2013), porque possibilita abrir sites de terceiros sem a apresentação da tela como em um *browser*, mas utilizando os seus recursos, pois trabalha com o *webkit* que dá suporte ao recurso HTML5 e estilos CSS.

A função do sistema Android WebView permite economizar tempo, bateria, além de ter acesso rápido às páginas da Internet por meio de aplicativos de terceiros. Essa função é muito utilizada em redes sociais como o Facebook e Twitter.

RESULTADOS

Foram escolhidas algumas telas para descrever o funcionamento da aplicação web e do app para Android. Na Figura 3 ilustra-se o cadastro de visitantes, no qual são registradas todas as suas informações.

Figura 3 – Tela de Cadastro de Visitantes

Cadastro dos Visitantes

Nome:

Apellido:

RG:

Visitou célula ou reunião?:

Data de Nascimento:

Sexo:

Data da Visita:

Quer conhecer Jesus?: Sim Não

Telefone:

Celular:

Pertence a alguma igreja?: Sim Não

Pertence a qual igreja?:

Estado Civil:

E-mail:

Pertence a alguma célula?: Sim Não

Quer conhecer alguma célula?: Sim Não

Endereço:

Complemento:

Deseja oração?: Sim Não

Qual o motivo da oração?:

Bairro:

Cidade:

Precisa de alguma ajuda específica?:

Estado:

CEP:

O que chamou a atenção na reunião de celebração ou na célula?:

Para realizar uma consulta sobre informações de cada célula, escolhe-se que se pretende visualizar as informações, como mostra a Figura 4.

Figura 4 - Tela para Consulta de Células

Consulta de Células

Escolha a Célula:

Detalhes da Célula:

Célula Apolo

Total de membros: 2
Endereço: Rua da célula de Apolo
Complemento:
Cep: 04462-111
Bairro: Centro
Cidade: São Paulo
Estado: SP

Reunião da Célula
Dia da semana: Terça
Horário: 18:00:00

[Alterar dados](#)

Líderes

1º Líder em Treinamento: João da Silva

[Detalhes dos líderes](#)

Rede / Setor

Rede: Vermelho
Setor: Genérico

	Nome do membro	Celular	Telefone	Endereço	Ações
1	Amanda Croce	11-91111-7777	11-4111-7777	Rua Arthur, 123	Desvincular membro Excluir participação
2	Carla Marquezzi	11-99222-6545	11-4222-6545	Rua Costa, 564	Desvincular membro Excluir participação

As telas ilustradas a seguir foram escolhidas para mostrar o módulo para dispositivos móveis do sistema Vem Vê.

As mostradas nas Figuras 5 e 6 ilustram a principal funcionalidade do sistema: a agenda de visitantes e a busca pela célula mais próxima, respectivamente. Na primeira, é possível escolher cada novo visitante e na segunda, visualizar a rota para percurso entre ele e a célula mais próxima.

Figura 5 – Agenda de Visitas

Vem vê

Agenda das visitas

Escolha a Célula

Todas

Status

Não contactado

Pesquisar

Pesquisar

Dados do Visitante	Célula responsável	Status Agendamento	Ação
João Augusto Endereço: Rua Nina Barueri Telefone: (11)	Agape	Não contactado 02:28:41	Rota Agendar/ Concluir Detalhes

Figura 6 – Definição de Rotas

Endereço do visitante (partida):
Rua Nina Barueri 06416-000 SP

Endereço da célula (chegada):
Rua Carlos Lacerda, 372 Barueri 06415-150 SP

Distância km: 0.7 km

Nome célula: Agape
Líder: Osvaldimar Costa

Nome visitante: João Augusto
Telefone visitante: (11) 9346-5780 /

Mapa Satélite

Prefeitura Municipal

R. Maria Helena

R. José

EM Luiz de Oliveira Andrade

Google

Dados cartográficos ©2016 Google

Termos de Uso

Voltar

O sistema também permite o redirecionamento da visita para uma outra célula (Figura 7).

Figura 7 – Redirecionamento de contato para outra Célula



The screenshot shows a mobile application interface with a dark status bar at the top displaying various icons and the time 22:15. The main content area is white and contains a dialog box with a close button (X) in the top right corner. The dialog title is "Direcionar contato para outra célula" with a blue arrow icon. Below the title, there is a text field containing "Nome do visitante: João Augusto". Underneath, there is a section labeled "Escolha a Célula" with an empty text input field. Below that is a section labeled "Observação" with a larger empty text area. At the bottom of the dialog, there are two buttons: "Direcionar" (blue) and "Cancelar" (white).

CONSIDERAÇÕES FINAIS

O propósito deste projeto foi apresentar uma aplicação web que visa auxiliar no gerenciamento de igrejas que trabalham em células, no que diz respeito a dados dos membros ativos, células, líderes, líderes em treinamento, setores aos quais cada célula pertence, reuniões (reuniões de celebração ou reuniões em célula) e os visitantes, tudo isso considerando a grande dificuldade que igrejas em célula encontram no gerenciamento dos membros ativos em cada célula e no contato 48h com os visitantes.

Entende-se que o uso da aplicação web e do app móvel ajuda, de modo geral, a tornar o gerenciamento das células e reuniões mais ágil e eficaz, permitindo que membros que estão em uma célula, porém, não frequentando com assiduidade a igreja possam ser adicionados como membros ativos.

FONTES CONSULTADAS

ANDROID STUDIO DEVELOPERS. Android Studio Release Notes. Disponível em: <<http://developer.android.com/intl/pt-br/tools/revisions/studio.html>> Publicado em: 24 maio 2013. Acesso em: 30 abr 2016.

INSTITUTO BRASILEIRO DE GEOGRAFIA E ESTATISTICA. Censo Demográfico 2010: Características gerais da população, religião e pessoas com deficiência. Disponível em: <http://www.ibge.gov.br/home/estatistica/populacao/censo2010/caracteristicas_religiao_deficiencia/caracteristicas_religiao_deficiencia_tab_pdf.shtm> Acesso em: 09 maio 2015.

APACHE SOFTWARE FOUNDATION, THE. Apache Tomcat®. Disponível em: <<https://tomcat.apache.org/download-80.cgi>> Publicado em: 22 maio 2014. Acesso em: 30 out 2015.

BOOCH, G; RUMBAUGH, J; JACOBSON, I. UML – Guia do Usuário. 2. ed. Elsevier, 2005.

COSTA, G. O que é Bootstrap?. Disponível em: <<http://www.tutorialwebdesign.com.br/o-que-e-bootstrap/>> Publicado em: 29 abr 2014. Acesso em: 30 abr 2016.

DAVIS, A. What's New in Java 8. Disponível em: <<https://leanpub.com/whatsnewinjava8/read>> Publicado em: 30 jun 2015. Acesso em: 28 dez 2015.

ECLIPSE FOUNDATION, THE. Subversive - SVN Team Provider. Disponível em: <<http://marketplace.eclipse.org/content/subversive-svn-team-provider>> Publicado em: 09 out 2012. Acesso em: 29 out 2015.

GENTIL, E. Introdução ao Spring Framework. Disponível em: <<http://www.devmedia.com.br/introducao-ao-spring-framework/26212>> Publicado em: 26 out 2012. Acesso em: 30 abr 2016.

HEUSER, C. Projeto de Banco de Dados. 4. ed. Sagra L, 1998.

THIENGO, V. WebView no Android, Entendendo e Utilizando. Disponível em: <<http://www.thiengo.com.br/webview-no-android-entendendo-e-utilizando>> Publicado em: 30 dez 2013. Acesso em: 28 abr 2016.