

Ferramenta Android baseada em realidade aumentada e serviços baseados em localização usando notificações

Acadêmico – Ronaldo Rampelotti

Orientador – Dalton Solano dos Reis



Roteiro

- Introdução
- Fundamentação teórica
- Desenvolvimento
- Resultados e discussão
- Conclusão/Extensões
- Demonstração



Introdução

- **Objetivos do trabalho**
 - disponibilizar uma ferramenta para plataforma Android de apoio ao universitário
 - permitir ao usuário visualizar eventos da FURB pelo dispositivo
 - fazer com que o usuário seja avisado da proximidade de um evento da FURB
 - dispor ao usuário a localização do evento por realidade aumentada
 - fazer com que o usuário possa se comunicar via e-mail com membros de uma disciplina em comum



Fundamentação teórica

- REST
 - estilo arquitetural de sistemas de informação distribuídos
 - representação do estado do recurso
 - baseado em padrões de trocas de mensagem como o protocolo SOAP
 - interação entre solicitante e fornecedor é feita com recurso ao protocolo HTTP



Fundamentação teórica

- JSON
 - formato das mensagens
 - menos espaço consumido



Fundamentação teórica

- Notificações (*Push*)

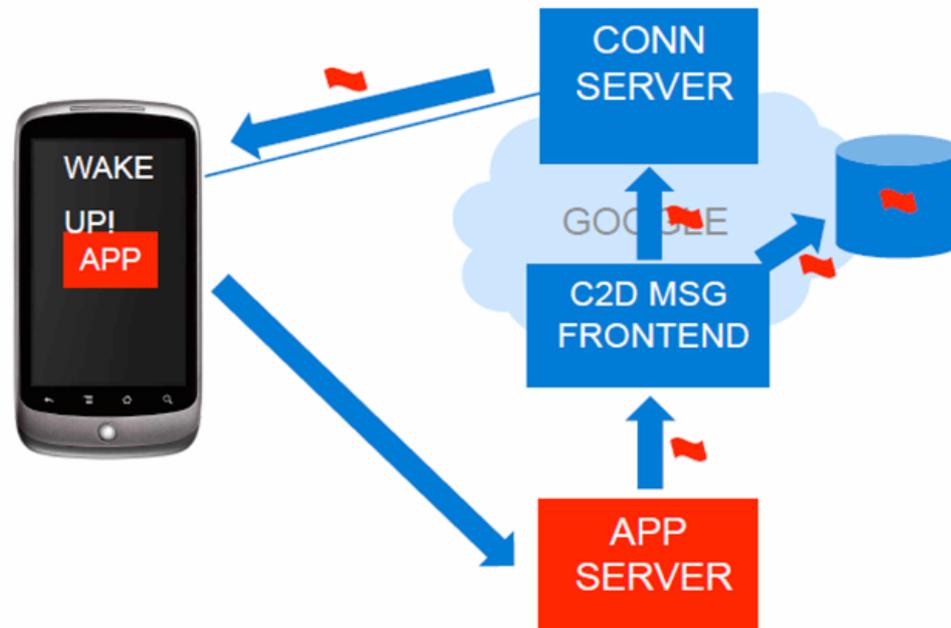
- *utilizado para avisar sobre alguma alteração no meio*
- *sempre com o intuito reativo*
- *economia de bateria*



Fundamentação teórica

- Cloud To Device Message (C2DM)

Life of a Message



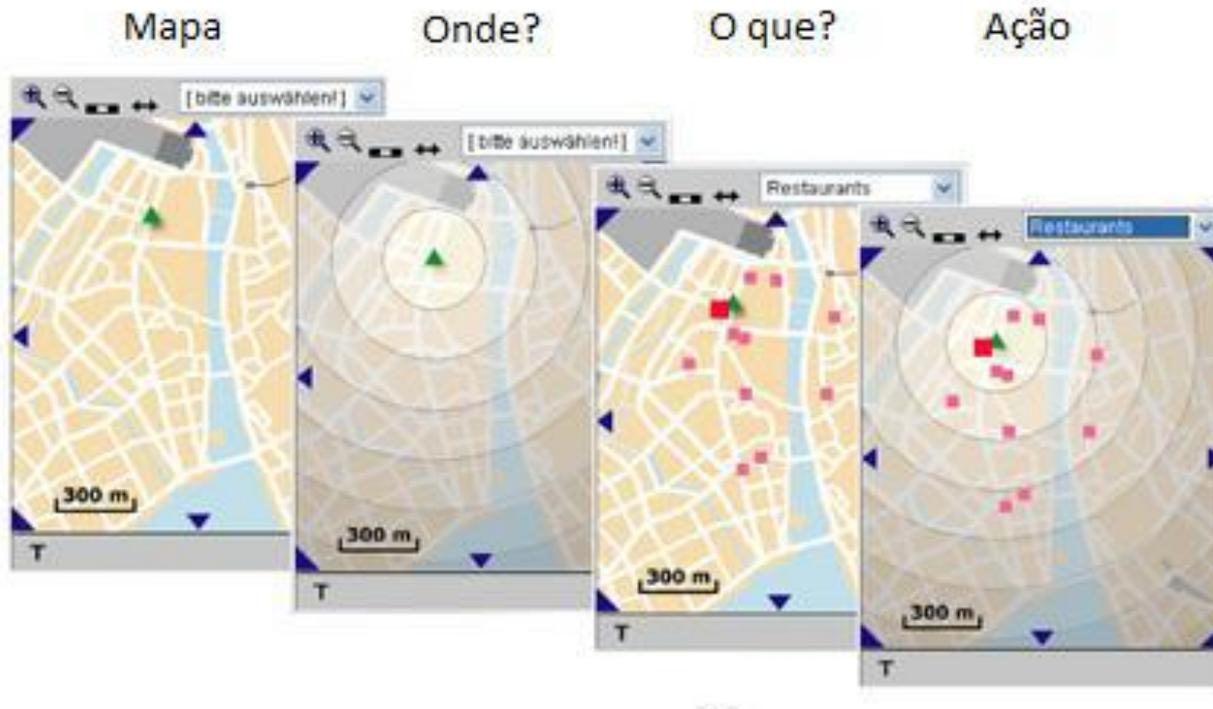
Fundamentação teórica

- Realidade Aumentada



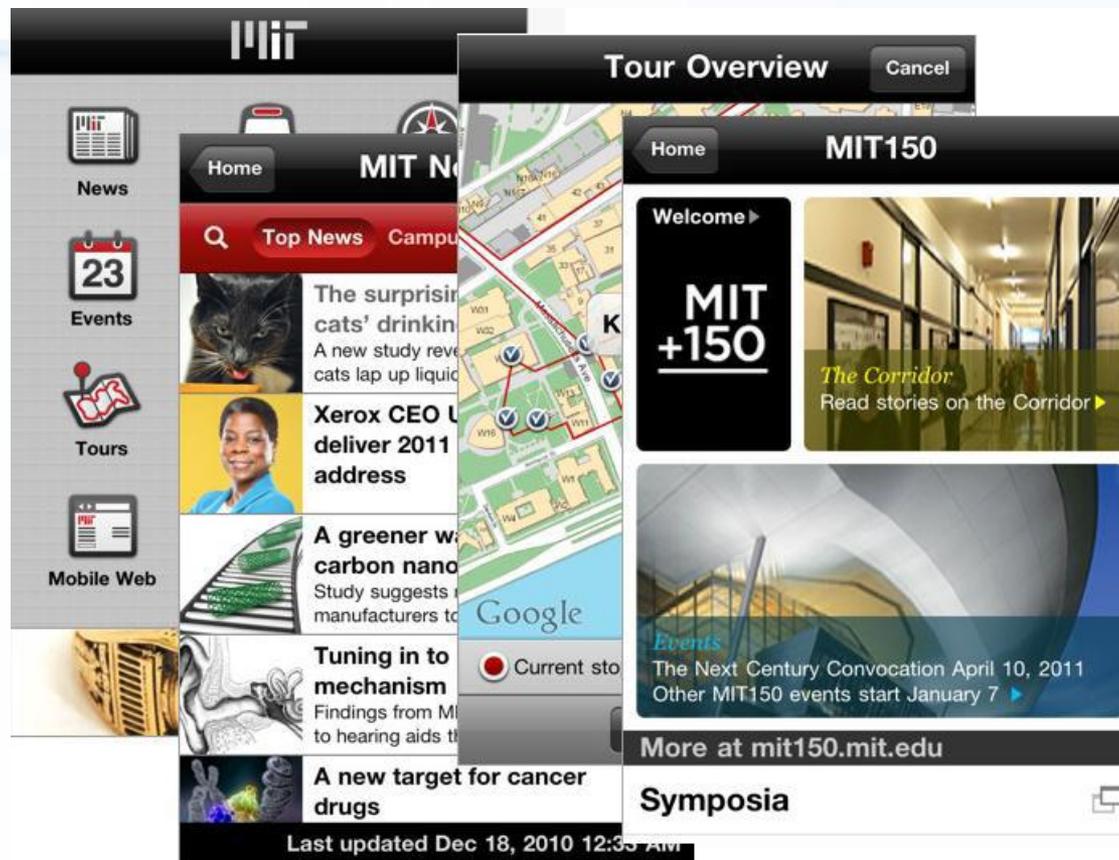
Fundamentação teórica

- Serviços baseados em localização



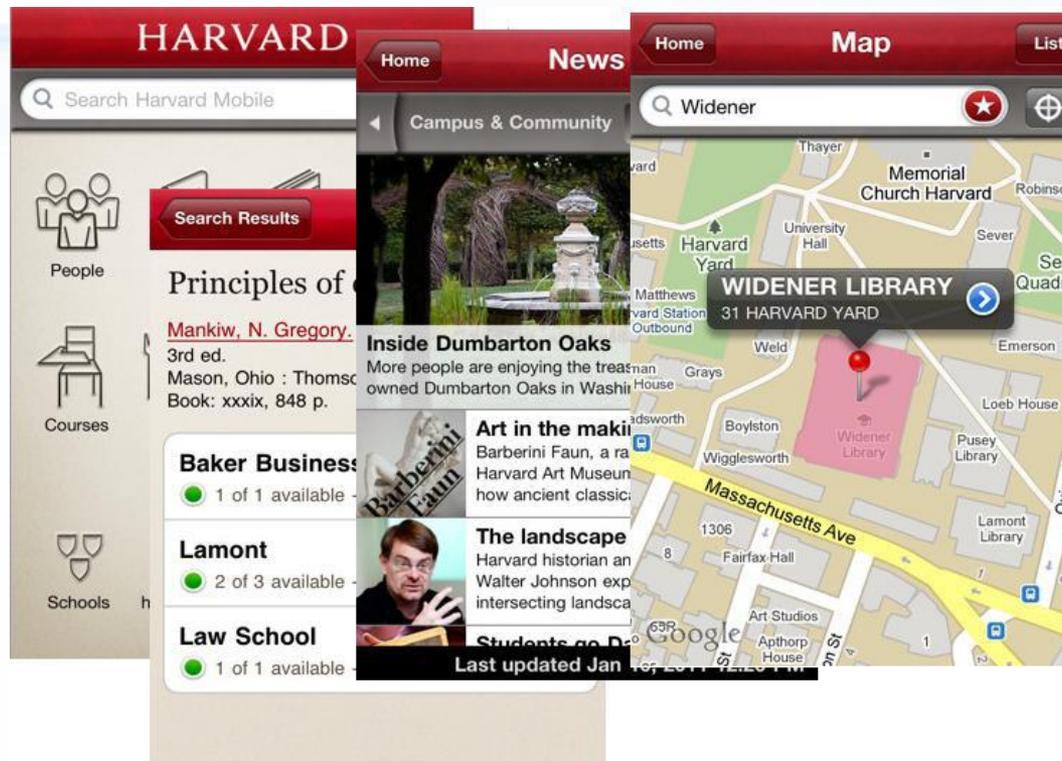
Fundamentação teórica

- Trabalhos correlatos
 - MIT Mobile



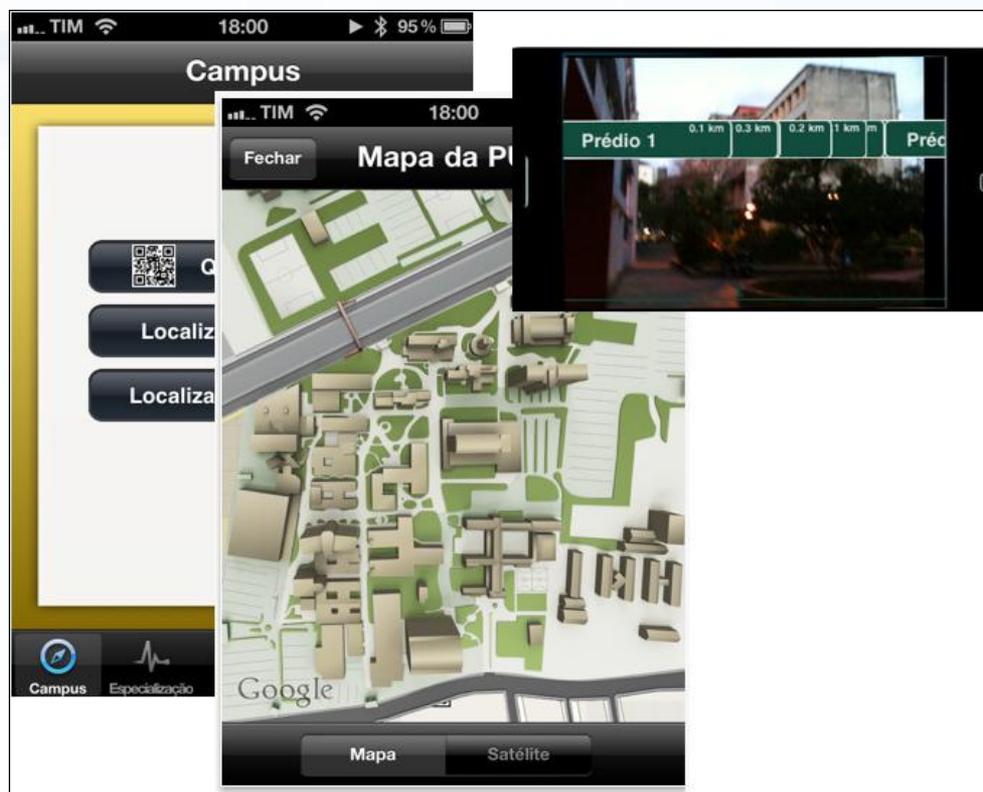
Fundamentação teórica

- Trabalhos correlatos
 - Harvard Mobile



Fundamentação teórica

- Trabalhos correlatos
 - PUCRS Mobi



Desenvolvimento

- Requisitos
 - permitir ao usuário localizar disciplinas
 - permitir ao usuário localizar membros das disciplinas
 - permitir ao usuário localizar algum evento
 - permitir ao usuário localizar ponto de interesse utilizando RA
 - permitir ao usuário se inscrever em eventos
 - permitir comunicação entre membros da disciplina via *e-mail*
 - permitir ao usuário consultar frequências



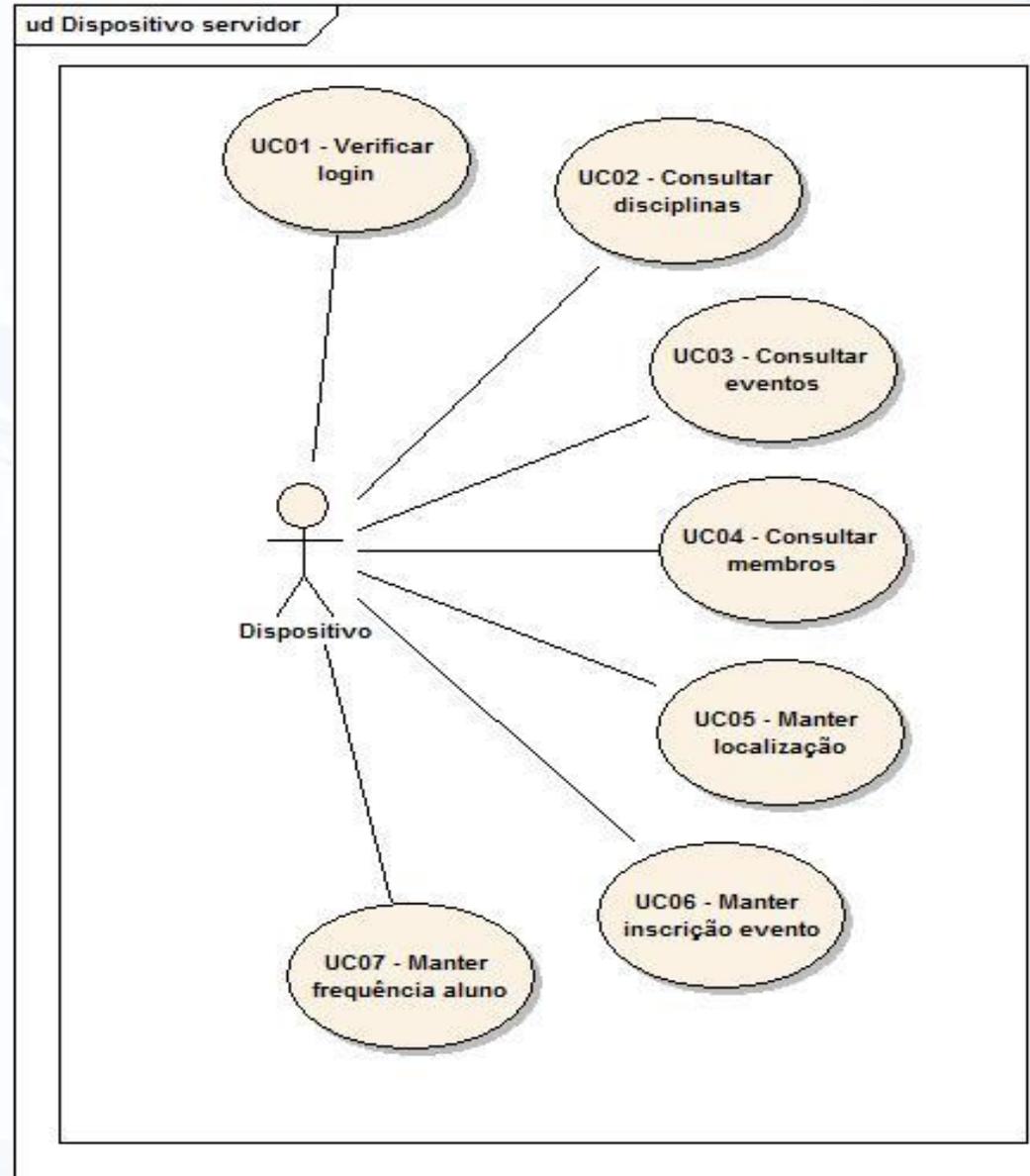
Desenvolvimento

- Requisitos
 - permitir cadastro de eventos através de um *site* por um professor
 - permitir ao professor postar notas
 - permitir notificações para o dispositivo sobre alertas de eventos
 - permitir ao professor postar frequências



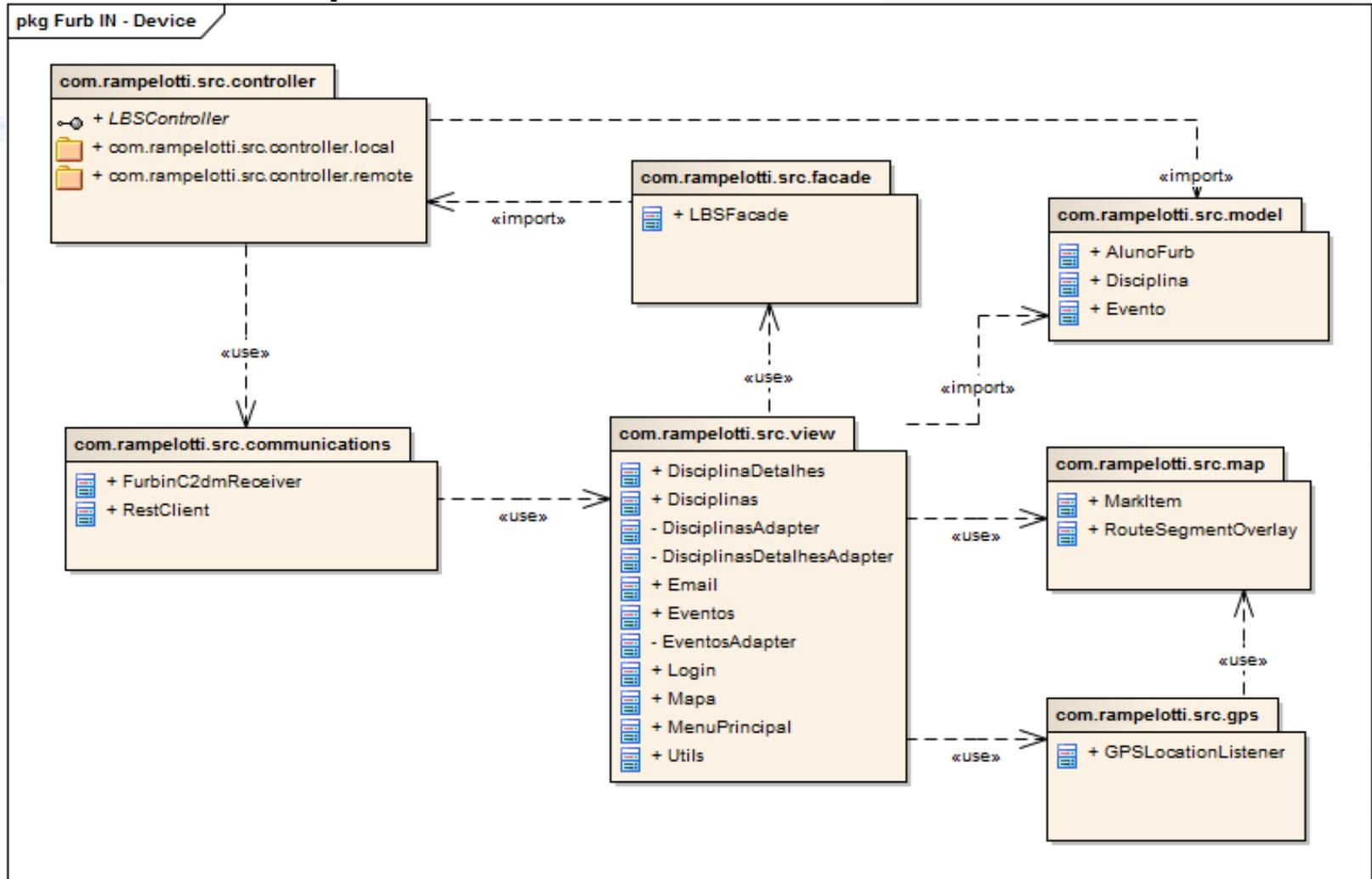
Desenvolvimento

- Especificação
 - Enterprise Architect



Desenvolvimento

- Classes dispositivo



Desenvolvimento

- Técnicas e ferramentas utilizadas
 - Eclipse
 - Android Development Tools (ADT)
 - Arquitetura SOA
 - JSF
 - Motorola Defy



Desenvolvimento

- Servidor SOA
 - geração de serviços REST

```
@Path("/eventos")
public class EventoResource extends AcessoBanco

    private String queryEventos = "SELECT CODEVI

    private String queryInserirEvento = "INSERT

    private String queryEventoUpdate = "UPDATE I

    private String queryEventoDrop = "DELETE FROM

    @GET
    @Produces("application/json; charset=utf-8")
    public List<Evento> getTodosEventos() {
        // Zera mapa de disciplinas
        ArrayList<Evento> eventos = new ArrayLis
        ...
    }
```



Desenvolvimento

- Servidor SOA
 - geração JSON

```
import javax.xml.bind.annotation.XmlRootElement;

@XmlRootElement
public class Evento {

    private int codEvento;

    private String descricao;

    private String dataEvento;
```



Desenvolvimento

- Servidor SOA

- gerando notificações

```
URL url = new URL("https://www.google.com/accounts/ClientLogin");
request = (HttpsURLConnection) url.openConnection();
request.setDoOutput(true);
request.setDoInput(true);

buf.append("accountType").append("=")
    .append((URLEncoder.encode("GOOGLE", "UTF-8")));
buf.append("&Email").append("=")
    .append((URLEncoder.encode(mail, "UTF-8")));
buf.append("&Passwd").append("=")
    .append((URLEncoder.encode(pass, "UTF-8")));
buf.append("&service").append("=")
    .append((URLEncoder.encode("ac2dm", "UTF-8")));
buf.append("&source").append("=")
    .append((URLEncoder.encode("myco-pushapp-1.0", "UTF-8")));

request.setRequestMethod("POST");
request.setRequestProperty("Content-Type",
    "application/x-www-form-urlencoded");
request.setRequestProperty("Content-Length", buf.toString()
    .getBytes().length + "");
```

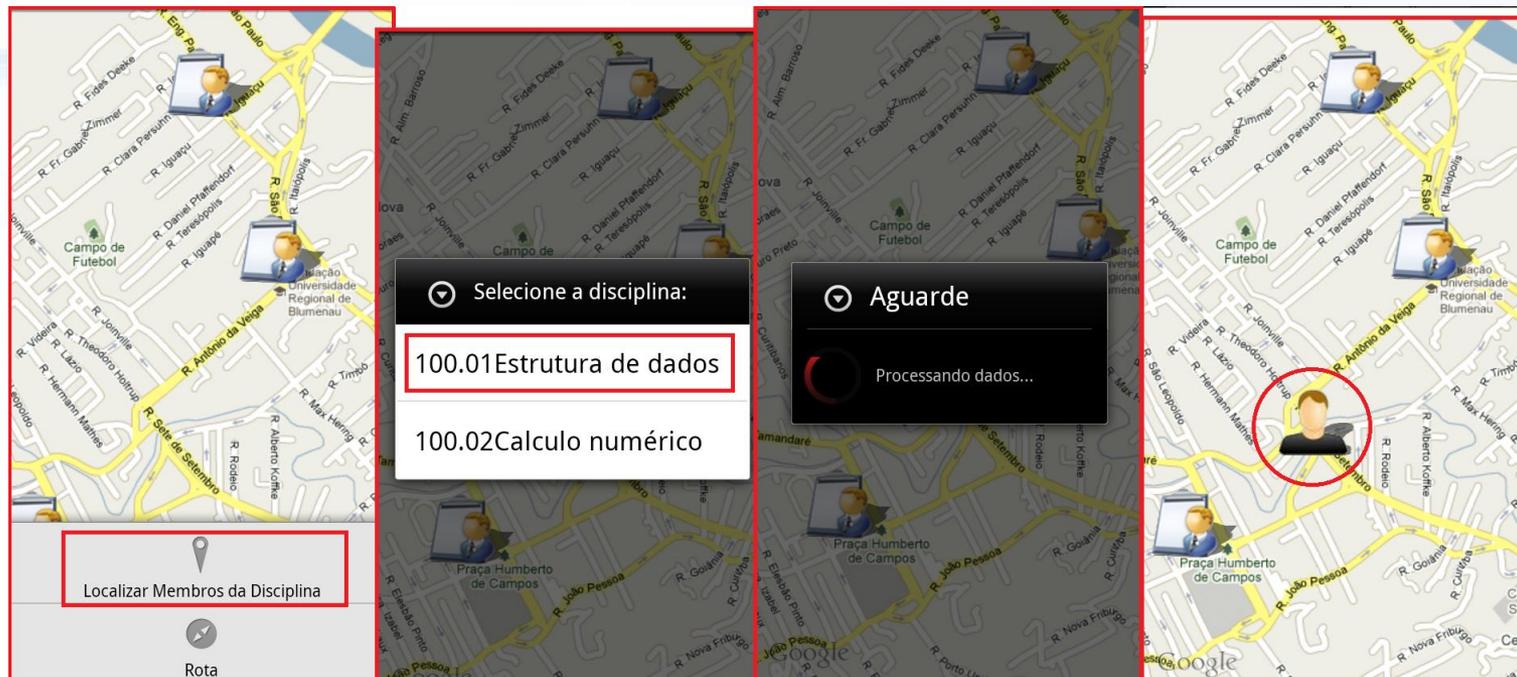
Desenvolvimento

- Dispositivo
 - ouvindo Notificações

```
/**
 * Método de criação da notificação.
 *
 * @param ctx
 */
private void criaNotificacao(Context ctx) {
    // Recupera o serviço do NotificationManager
    NotificationManager notificationManager = (NotificationManager) ctx.getSystemService( Context.NOTIFICATION_SERVICE);
    Notification notificacao = new Notification(R.drawable.icon, "Furbin", System.currentTimeMillis());
    notificacao.sound = Uri.withAppendedPath(Audio.Media.INTERNAL_CONTENT_URI, "6");
    // Flag que vibra e emite um sinal sonoro até o usuário clicar na notificação
    notificacao.flags |= Notification.FLAG_INSISTENT;
    // Flag utilizada para remover a notificação da toolbar quando usuário tiver clicado nela.
    notificacao.flags |= Notification.FLAG_AUTO_CANCEL;
    // PendingIntent para executar a Activity se o usuário selecionar a notificação
    PendingIntent p = PendingIntent.getActivity(ctx, 0, new Intent(ctx.getApplicationContext(), MenuPrincipal.class), 0);
    // Informações
    notificacao.setLatestEventInfo(ctx, "Evento", "Você está próximo ao evento", p);
    // Espera 100ms e vibra por 1000ms, depois espera por 1000 ms e vibra por 1000ms.
    notificacao.vibrate = new long[] { 100, 1000, 1000, 1000 };
    // Id que identifica esta notificação
    notificationManager.notify(R.string.app_name, notificacao);
}
```

Desenvolvimento

- Operacionalidade da aplicação



Resultados e discussão

- Notificações
 - Serviço C2DM fora
- REST
 - WCF sem conhecimento
- Servidor SOA



Conclusão

- Ferramenta
- Estudo de serviços REST
- JSON
- Solução integrada
- Semelhança com os correlatos



Extensões

- adicionar funcionalidade para publicação de mídia georeferenciada
- adicionar funcionalidade para captura de posição com triangulação de rede Wi-Fi obtendo a posição do aluno dentro de um ambiente fechado
- adicionar funcionalidade de leitura de QRCode para acesso a informações em ambiente fechado
- explorar o uso de mapa alternativo, outra fonte de mapas como um desenho da própria universidade em cima do mapa Google



Demonstração

Demonstração vídeo



Obrigada



iPhone APN

