|  |  |
| --- | --- |
|  |  |



Data: 25/05/2011

**Especialização – Padrões de Projetos**

Modelagem do Photomap – Um sistema para geotagging de fotos

Acesso ao Lec (LOGIN: eti2007 ; SENHA: etijava )

**Exercício: Modele o sistema abaixo no Jude usando ao menos 5 padrões de projetos do GoF para facilitar sua evolução**

Descrição geral: Um sistema deve permitir aos usuários tirar fotos e visualizá-las em um mapa com sua posição atual.

Outras funcionalidades: Auxílio à Navegação

Observações: Alguns códigos anteriores serão reusados

Descrição da modelagem inicial

0) Interface Visual

Contém um Menu com as opções:

1)Ver Mapa -> Mostra o mapa e se atualiza com a mudança de coordenadas GPS

2) Ver Meu Endereço -> Mostra o endereço atual

3) Tirar Foto -> Tira Foto

4) Clima -> Mostra a temperatura atual

5)Buscar POIs –> Mostra o nome, a descrição e a distância

I) Acesso ao GPS

* Uma classe, GPSLocationSensor, deve ser responsável por encapsular o acesso ao GPS do dispositivo.
* Só pode haver uma instância dessa classe para evitar sobrecarga do GPS.
* Várias partes do sistema podem estar interessadas na informação e desejam ser notificadas caso haja uma mudança nas coordenadas (em geral, restritas a mais de 0,002 graus de diferença)

2) Gerando o endereço

* Para gerar o endereço atual do usuário, vários serviços podem ser consultados: Google Places, Geonames, Yahoo!, ...
* A entrada destes serviços é uma coordenada GPS e a saída pode ser padronizada em um Endereço simples
* Alguns desses serviços saem do ar facilmente, é importante esconder essa falha do resto da aplicação. Caso um serviço falhe, outro deve ser instanciado sem que o cliente da classe perceba.
* Apenas se todos falharem, o resto do sistema deve ser informado.

3) Mostrando um mapa para navegação

* Para mostrar o mapa do usuário, duas técnicas estão disponíveis. Uma usa o serviço do Yahoo e outra do Google Maps
* A versão do mapa do Yahoo deve ser usada quando o usuário estiver fora de Fortaleza e o do Google Maps dentro da cidade. Isso pode ser checado quando o usuário clicar no botão da interface principal: “Abrir Mapa” e deve estar transparente para a classe que chama a geração do mapa.
* A entrada destes serviços é uma coordenada GPS (lat,log) e a saída pode ser padronizada para uma imagem simples do tamanho da tela do celular
* Independente de qual serviço esteja sendo utilizado. O mapa deve ser regerado automaticamente quando existir uma mudança de mais de 0,002 das coordenadas do GPS.

4) Pontos de Interesse

* A cada dez atualizações do GPS, uma lista de Pontos de Interesse (POI) é gerada e posta em memória, para consultas e sugestões de acesso
* Um ponto de interesse possui um nome, uma descrição e pode possuir também uma lista de pontos de interesse internos (exemplo, o North Shopping é um POI, sua Praça de Alimentação e o MacDonalds da praça também são).
* Um POI contém uma distância em relação às coordenadas do usuário. A distância de um POI composto é a média das distâncias de seus POIs.

5) Mostrando o Weather

* Uma classe para mostrar o tempo (ViewWeather) usa um serviço da interface Weather Service para receber o clima atual. O método utilizado é o getWeatherFromLocation(lat, lng):Weather
* Não se deseja mudar esta classe, pois foi desenvolvida por terceiros.
* Devemos integrar ao sistema uma classe WeatherUnderground que foi desenvolvida por outra equipe da nossa empresa e possuí o método getWeatherFromCoords(lng, lat):Weather.