

YouOnAlert: Um Sistema para Alertar Cidadãos Comuns Acerca de Problemas do Cotidiano das Cidades

André L. C. Rocha¹, Vaninha Vieira^{1,3}, Cássio V. S. Prazeres¹, Celso A. S. Santos²

¹Departamento de Ciência da Computação – Universidade Federal da Bahia (UFBA)
Av. Adhemar de Barros, s/n, Ondina – CEP 40.170-110 – Salvador – BA – Brasil

²Departamento de Informática – Universidade Federal do Espírito Santo (UFES)
Av. Fernando Ferrari, s/n, Goiabeiras – CEP 29.060-970 – Vitória – ES – Brasil

³Centro de Projeto Fraunhofer para Engenharia de Software e Sistemas
Parque Tecnológico, Salvador – BA – Brasil

{andreluiz, vaninha, prazeres}@dcc.ufba.br, saibel@inf.ufes.br

Abstract. *The increasing use of smartphones facilitates the development of applications and services that connect communities of users based on their geolocations. Services as travelling and navigation are well established. However, there are few services targeted to personal safety and emergency support. This work aims to create a tool to support ordinary people identifying problem situations that perhaps may cause them some risk. In tests, the prototype identified 100% of the annotated problems and 67% of participants said they would use an application with these goals in their daily lives.*

Resumo. *O crescente uso de smartphones facilita o desenvolvimento de aplicações e serviços que conectam comunidades de usuários baseadas em suas geolocalizações. Serviços como viagens e navegação estão bem estabelecidos. No entanto, ainda existem poucos serviços direcionados para a segurança pessoal e apoio a emergências. Esse trabalho tem o objetivo de criar uma ferramenta para apoiar o cidadão comum na identificação de situações problema que, porventura, possam lhe causar algum risco. Nos testes realizados, o protótipo identificou 100% dos problemas anotados e, 67% dos participantes declararam que usariam uma aplicação com esses objetivos no seu dia a dia.*

1. Introdução

Brabham (2008) define *crowdsourcing* como um modelo de produção para resolver problemas com base na inteligência e conhecimento coletivos. Atualmente, existem diversos sistemas Web que utilizam esse modelo para identificação de elementos geolocalizados utilizando mapas [Goodchild 2007]. Alguns sistemas como OSM [Haklay e Weber 2008], visam o mapeamento voluntário de informações geográficas (VGI – *Volunteered Geographic Information*) de lugares imóveis e com um ciclo de vida longo (como ruas, pontos turísticos, cidades e construções). Outros sistemas visam geração de anotações colaborativas para mapeamento de incidentes com ciclo de vida curto, como crimes [Furtado et al. 2010], crises [Ushahidi 2013] e saúde [Qureshi et al. 2011].

Como os sistemas de computação estão se tornando mais intimamente incorporados em contextos físicos e sociais, *crowdsourcing* vem assumindo novas formas, promovendo a computação ubíqua. Cada vez mais, as multidões estão empenhadas – através de dispositivos móveis – em capturar e compartilhar grandes quantidades de informação [Vukovic et al. 2010]. Nesse cenário, a computação ubíqua e sensível ao contexto, que se caracteriza pela presença de dispositivos portáteis, aparece como peça de fundamental importância. Atualmente, tais dispositivos possuem uma considerável capacidade de processamento. Eles se popularizaram principalmente como *handhelds*, PDAs, *tablets*, e, ultimamente, *smartphones*. Além das funcionalidades originais, *smartphones* possuem diversos recursos e interfaces adicionais como GPS, WiFi e câmeras fotográficas de alta resolução [Vieira et al. 2010] [Loureiro et al. 2009].

Para Ovelgonne et al. (2010), o crescimento no uso de *smartphones* facilita a criação de aplicações e serviços que conectam comunidades de usuários baseadas nas geolocalizações desses indivíduos. Portanto, serviços baseados em localização (tais como viagens e navegação), que fornecem informações aos usuários sobre os seus arredores, estão bem estabelecidos até o momento. No entanto, serviços direcionados para a segurança pessoal e apoio a emergências são escassos. Por exemplo, existem poucas aplicações voltadas para o relato de problemas comuns do cotidiano de grandes áreas urbanas, tais como problemas de segurança, trânsito, infraestrutura, entre outros. Esses problemas podem oferecer riscos diretos à vida e à integridade física dos cidadãos. Diante de ocorrências provenientes de tais problemas, o poder público é acionado para apoiar as vítimas. Essa intervenção pública gera gastos à cidade, que poderiam ser minimizados se os cidadãos pudessem identificar com antecedência – com o auxílio de sistemas informatizados – a existência desses problemas nos seus arredores.

Considerando o contexto de Cidades Inteligentes [Erickson 2010], a ascensão do modelo *crowdsourcing* e as características da computação ubíqua e sensível ao contexto [Castillo et al. 2011], esse trabalho tem o objetivo de criar uma ferramenta para apoiar o cidadão comum na identificação dinâmica, e em tempo real, de situações problema que, porventura, possam lhe causar riscos. Dessa forma, pretende-se ajudar na diminuição de acidentes com vítimas e, conseqüentemente, na redução do gasto público no apoio a essas vítimas. Para tanto, foi desenvolvido um protótipo de anotações sociais em mapas que simula o suporte a um modelo de produção *crowdsourcing* através de uma interface móvel. Esse artigo contribui para a área de pesquisa com: (i) uma nova arquitetura, que alia o uso de informações, produzidas por meio de *crowdsourcing*, a aplicações voltadas para alertas de emergências e (ii) a construção e avaliação experimental de um aplicativo ubíquo desenvolvido com base nesta arquitetura. A experimentação do aplicativo em situações reais de uso evidenciou problemas que podem ocorrer em ambientes ubíquos, bem como, possíveis melhorias na usabilidade e interfaces com usuários em movimento.

A realização desse trabalho teve como passos preliminares: o estudo de propostas focadas no modelo *crowdsourcing* para anunciar problemas por meio de anotações em mapas. Além disso, um Modelo de Contexto [Dockhorn and Rizzi 2012] foi idealizado para modelar algumas das principais situações envolvendo esse ambiente. A maior parte da avaliação do protótipo foi realizada no campus da Universidade Federal da Bahia – UFBA. Anotações foram criadas em pontos estratégicos, representando problemas comuns encontrados em grandes centros urbanos. O restante do artigo é distribuído da seguinte forma: na Seção 2, revisamos a literatura, mostrando alguns

trabalhos relacionados; na Seção 3, descrevemos a proposta; na Seção 4, descrevemos a avaliação e apresentamos os resultados obtidos; na Seção 5 discutimos as considerações finais e os trabalhos futuros.

2. Revisão da literatura

Nessa seção são mostrados alguns estudos que possuem relação com esse trabalho. A pesquisa foi concentrada nas áreas de *crowdsourcing*, informação geográfica voluntária (VGI) e serviços de alerta. De acordo com Brabham (2008), o termo *crowdsourcing* foi cunhado por Jeff Howe e Mark Robinson em junho de 2006. Ele descreve um novo modelo de negócios baseado na Web, que utiliza as soluções criativas de uma rede distribuída de indivíduos. Shah et al. (2011) esclarecem que com o advento dos dispositivos inteligentes, muitas aplicações móveis usam *crowdsourcing* para reportar problemas da comunidade. Shah et al. (2011) propõem CROWDSAFE. A ferramenta habilita, em tempo real, localização, busca e relatório de incidentes criminosos. O sistema é voltado para usuários que tem interesse em informações acerca de crimes.

Sistemas de Informação Geográfica Voluntária são ferramentas para criar e disseminar dados geográficos fornecidos voluntariamente por indivíduos [Coleman et al. 2009]. Alguns trabalhos associam o uso de técnicas de *crowdsourcing* a sistemas VGI, em aplicações voltadas para o informe de crimes e problemas nas grandes áreas urbanas. Segundo Goodchild (2007), soluções como OSM [Haklay e Weber 2008] estão capacitando os cidadãos para criar uma miscelânea global de informações geográficas. Nesse sentido, Furtado et al. (2010) descrevem WikiCrimes, uma aplicação Web 2.0 que oferece um ambiente colaborativo baseado no uso e manipulação direta de mapas, a fim de registrar e pesquisar ocorrências de crimes.

O projeto Spaces 4D [Gonzales et al. 2013] é um projeto acadêmico desenvolvido em colaboração pelas instituições UFBA, UFSC e UNICAMP. O projeto tem o objetivo de aproximar a gestão pública dos cidadãos ao permitir que qualquer pessoa informe problemas através de uma interface Web de mapas. Uma anotação de problema gera um chamado para a autoridade pública. Dessa forma, o gestor público pode tomar conhecimento detalhado dos problemas em tempo curto, e, assim, acionar agentes públicos para investigar e resolver tal problema com mais eficácia.

As anotações combinadas a serviços de alertas podem criar soluções mais inteligentes e pró-ativas, que apoiem os cidadãos em situações cotidianas. Ovelgonne et al. (2010) propõem um serviço de alerta de emergência social, denominado SEAS - *Social Emergency Alert Service*, que faz uso da disponibilidade de *smartphones* e ativa contatos próximos em casos de emergência. O serviço SEAS informa aos membros próximos (família, conhecidos) sobre uma pessoa em necessidade. Para enviar o alerta, basta a vítima clicar em um simples botão. Ovelgonne et al. (2010) também salientam que aplicações móveis direcionadas para a segurança pessoal e apoio a emergências ainda são escassas. Com esse trabalho, os autores chegaram à conclusão que a emissão de alertas para situações de perigo pode ajudar cidadãos em momentos de crise.

O estudo desses trabalhos foi importante para entender melhor o real significado do modelo *crowdsourcing* e como tal modelo pode ser empregado em soluções de VGI com mecanismos de alerta, no apoio a cidadãos em situações de emergência. Esses estudos evidenciaram a escassez de soluções voltadas para esse domínio. Em virtude

dessas questões, esse trabalho identificou a relevância de construir uma aplicação ubíqua com mecanismo de alerta, sensível ao contexto, para notificar cidadãos acerca de problemas na localização que o cerca, à medida que ele se desloca em um centro urbano. A solução proposta é apresentada na seção a seguir.

3. A proposta YouOnAlert

YouOnAlert (UOA) foi desenvolvida pensando na concepção de uma ferramenta que unisse tanto o poder da colaboração voluntária do modelo *crowdsourcing* e o atual crescimento de VGI, quanto as características ubíquas das plataformas móveis e a variabilidade contextual dos ambientes que cercam os cidadãos nos centros urbanos.

A ferramenta é composta por quatro módulos principais (Figura 1): Módulo de Crowdsourcing (MC), Módulo de Aquisição de Contexto (MAC), Módulo de Processamento de Contexto (MPC), e Módulo de Alerta (MA). Três bases de dados são utilizadas: Base Externa (BE), Base de Anotações Local (BAL) e Base de Conhecimento Contextual (BCC). Esses elementos estão distribuídos em três camadas: Camada de Aquisição e Anotação de Problemas (CAAP), Camada de Processamento de Contexto (CPC) e Camada de Interfaces e Alertas (CIA).

A BE é um conjunto de bases colaborativas de anotações de terceiros. Essas bases podem ser integradas, por exemplo, por meio do modelo *Linked Data* [Berners-Lee 2006]. Na BE são armazenados perfis, preferências e anotações de usuários do sistema. O MC consulta essa base para recuperar as anotações de problemas e os dados dos usuários, além de alimentá-la com criações e edições dessas informações.

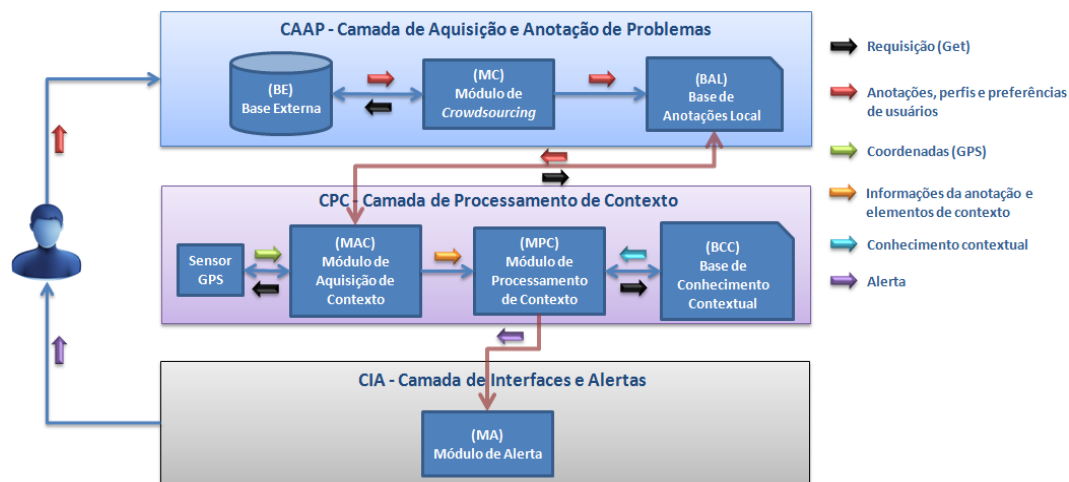


Figura 1. Arquitetura YouOnAlert.

Depois de recuperadas, o MC envia tais informações para a BAL (arquivo no formato JSON), onde são armazenadas para consulta pelo MAC. O MAC recebe constantemente coordenadas oriundas do sensor GPS do dispositivo móvel do usuário. Para cada ponto geográfico recebido por esse módulo, uma consulta é realizada na BAL (comunicação entre CAAP e CPC), verificando se a localização atual do dispositivo está dentro do raio de ação de uma anotação (o raio de ação de uma anotação é definido a partir de informações de contexto. Por exemplo, velocidade de aproximação do ponto anotado). Ou seja, verifica se, por exemplo, uma anotação contida na BAL está a menos que 40 metros do ponto atual recebido por meio do sensor GPS.

Se o dispositivo estiver dentro do raio de ação de uma anotação, isso significa que existe a possibilidade de emissão de um alerta acerca do problema descrito por essa anotação. Para determinar se o alerta é emitido, o MAC envia as informações da anotação (tipo, ponto geográfico, entre outros) e elementos de contexto (proximidade, velocidade de deslocamento) para o MPC. O MPC analisa essas informações de acordo com as situações que foram mapeadas na BCC (a BCC contém o modelo de contexto descrito da Seção 3.2). Por exemplo, se o tipo da anotação for Buraco na Pista e o usuário estiver a bordo de um veículo (velocidade de deslocamento alta), um alerta é emitido. Por outro lado, se para essa mesma anotação, o usuário estiver caminhando na calçada (velocidade de deslocamento baixa), nenhum alerta é emitido.

Se for determinada a necessidade de emissão de um alerta, o MPC envia essa solicitação para o MA (comunicação entre CPC e CIA). Consequentemente, o MA emite um alerta para o usuário na forma de um alarme sonoro, exibindo uma tela de mapa indicando o problema por meio de uma marca. O usuário pode interagir com o MC para criar ou editar registros, alimentando a BE com informações de perfil, preferências e anotações de problemas. Por exemplo, considerando um usuário que costuma transitar, mais frequentemente, em uma determinada região da cidade, essa informação pode ser usada para alimentar o perfil desse indivíduo. Assim, no futuro, podem-se priorizar os alertas nessa região para este usuário em especial.

3.1. Modelo de Contexto

Modelos de contexto são representações abstratas das situações e condições que são relevantes para o universo de discurso de uma aplicação [Dockhorn and Rizzi 2012].

Um exemplo de situação considerada é a seguinte: existe uma anotação de problema em uma via – um buraco na rua por onde os veículos trafegam. Para motoristas, é interessante receber o alerta indicando a presença do buraco à frente. Por outro lado, para pedestres, que estão caminhando pela calçada, não há a necessidade de emitir um alerta para um buraco que está na pista de rolamento dos automóveis. O contexto é utilizado para apoiar a tarefa de emissão de alertas. A depender do contexto em que se encontra o cidadão, o alerta pode, ou não, ser emitido.

O contexto é referente ao cidadão, usuário de um smartphone, pois deseja-se saber o contexto em que ele se encontra, como por exemplo, se ele está caminhando (é um pedestre) ou se está a bordo de um veículo (é um motorista ou passageiro). A Tabela 1 apresenta a identificação das entidades contextuais e elementos de contexto considerados no modelo.

Tabela 1. Modelo de contexto YouOnAlert

Entidade	Elemento de Contexto	Fonte de Contexto	Característica de Aquisição
Cidadão	Pedestre ou Motorista	sensor GPS do <i>smartphone</i>	Dinâmico, implícito. Calculado a partir da velocidade de deslocamento atual do dispositivo
	Tipo do dispositivo	Perfil	Estática, explícita. Indica o tipo do dispositivo usado pelo cidadão.
	Atividade atual	Inferência	Dinâmico, implícito. Indica a atividade atual do cidadão, que pode impactar em emitir ou não o alerta
Dispositivo	Velocidade de deslocamento	sensor GPS do <i>smartphone</i>	Dinâmico, implícito. Interpretado recuperando sucessivamente latitude e longitude dos pontos, e calculado através de operações matemáticas
	Localização	sensor GPS do <i>smartphone</i>	Dinâmico, implícito. Interpretado recuperando sucessivamente latitude e longitude do ponto atual.
	Proximidade de ponto anotado	Base de Anotações dos problemas	Dinâmico, implícito. Interpretado a partir da Localização atual do cidadão em relação à Base de Anotações Local (BAL), calculado através da Distância Euclidiana entre dois pontos.

A adaptação ao contexto ocorre quando é identificada uma proximidade adequada entre o cidadão e um ponto que representa um dado problema anotado (e.g. buraco na pista). As informações de contexto monitoradas em tempo real determinarão se o alerta será emitido ou não. Portanto, considerando o problema exemplificado acima, para indicar se o alerta será emitido, o sistema observará a velocidade de deslocamento do dispositivo (desconsiderando os estados *parado* e *iniciando movimento*); se essa velocidade estiver acima de um determinado valor (e.g. 20 Km/h), então o alerta será emitido, pois é provável que o cidadão esteja a bordo de um veículo (motorista). Caso contrário, o alerta não será emitido, pois provavelmente o cidadão está caminhando na calçada (pedestre) e não está interessado em receber alertas sobre buracos na pista.

3.2. Implementação do Protótipo

A plataforma escolhida para o desenvolvimento do protótipo foi a plataforma *Android*. A escolha dessa plataforma deve-se à sua popularidade e disponibilidade em dispositivos diversos (o que facilitaria a avaliação da proposta por usuários diversos). Outra vantagem é que a linguagem de programação utilizada por sua IDE de desenvolvimento é Java, uma das mais usadas e disseminadas, e que possui uma vasta documentação.

No restante dessa seção, vamos apresentar os recursos do protótipo e seu funcionamento¹. Para isso, mostraremos as principais telas da ferramenta, explicando a funcionalidade e os objetivos de cada parte do sistema.

Na tela de abertura (Figuras 2a e 2b), o sistema verifica se o sensor GPS do dispositivo está ativado. Caso contrário, mostra um aviso e questiona o usuário se deseja ligá-lo. Existe a possibilidade de, mesmo com o GPS ativado, o dispositivo não conseguir determinar sua posição geográfica. Isso acontece, por exemplo, quando o dispositivo se encontra sem acesso à Internet e/ou *indoor*. Nesse caso, o sistema avisa da impossibilidade de funcionamento da ferramenta, já que o fator localização é essencial.

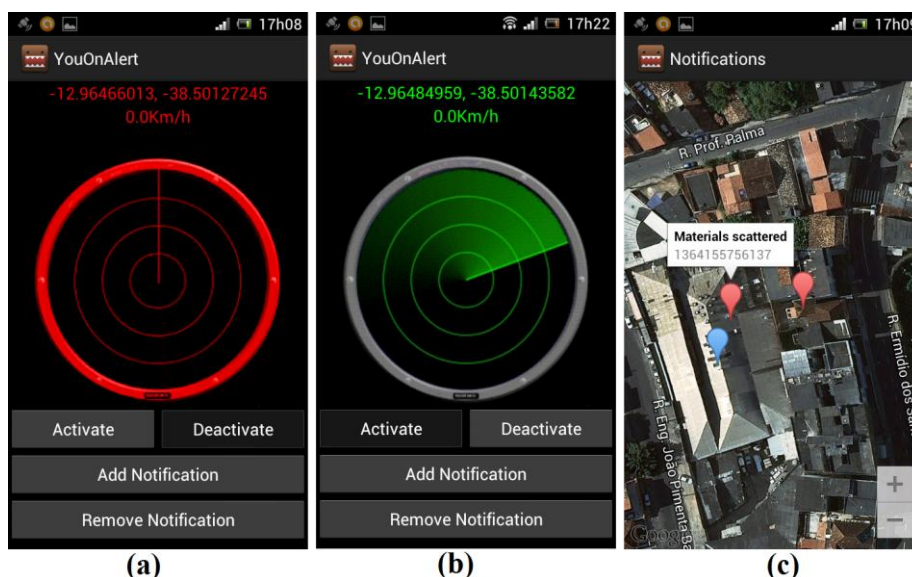


Figura 2. Tela de abertura (2a e 2b) e tela do mapa mostrando problemas (2c).

¹ Disponível para *download* e testes em <https://www.dropbox.com/s/idrjhdtqp4yebpv/YouOnAlert.apk> e <https://docs.google.com/file/d/0B6EboaQTIHGTSTRuTDZIZVpYakU/edit>

Com o GPS funcionando corretamente, as coordenadas geográficas e a velocidade de deslocamento são mostradas na parte superior da tela de abertura. Quando a ferramenta está em modo de alerta (Figura 2b), a interface mostra uma figura animada e a cor das informações na tela muda para verde. Isso indica que a ferramenta está buscando por pontos anotados na região (os problemas informados via *crowdsourcing*).

Quando o sistema identifica que a localização atual está dentro do raio de ação de uma anotação, ele verifica, segundo as condições mapeadas no modelo de contexto, se um alerta deve ser emitido. Caso positivo, um aviso sonoro é emitido pelo dispositivo e uma tela com um mapa é exibida (Figura 2c). Esse mapa pode conter várias marcas, a depender da quantidade de problemas identificados nos arredores do cidadão. Uma marca representa a posição atual do dispositivo, as outras, representam problemas. Cada marca possui um balão de informação com a descrição do tipo de problema (buraco na rua, área com assaltos, entre outros). A tela do mapa é dinâmica, ou seja, à medida que o usuário se desloca, a marca representando a localização atual muda de posição e as marcas representando problemas vão sumindo e aparecendo de acordo com a localização das anotações.

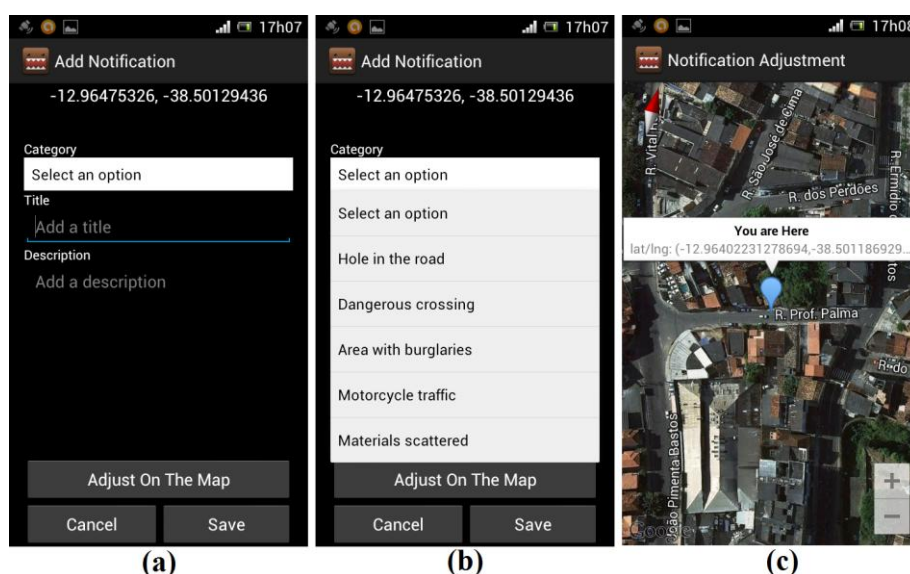


Figura 3. Telas de criação de anotações.

O usuário pode anotar pontos e descrever os problemas associados a eles (Figura 3a). Para isso, o usuário deve fornecer ao sistema categoria, título e descrição do problema. Na Figura 3b, podemos ver algumas das categorias definidas pelo sistema (buraco na rua, travessia perigosa, e outras). O usuário pode ajustar a localização da anotação através de um mapa interativo, onde é possível arrastar a marca para a posição desejada (Figura 3c). Isso permite uma maior precisão na localização dos problemas.

No protótipo atual do aplicativo, o Módulo de Crowdsourcing e a Base Externa não foram desenvolvidos, tendo sido simulado o seu funcionamento. Um único arquivo simula a BE, o MC e a comunicação entre eles. As informações de anotações usadas nos testes foram estaticamente armazenadas nesse componente dentro do dispositivo. Portanto, o Módulo de Aquisição de Contexto recebe as anotações desse componente. Da mesma forma, usuários interagem com esse componente quando criam anotações (as novas anotações são armazenadas localmente, no próprio dispositivo).

4. Avaliação do protótipo

A avaliação do protótipo foi realizada entre os dias 11/03 e 12/04, no período do dia compreendido entre 11 horas da manhã e 14 horas da tarde, no campus de Ondina da Universidade Federal da Bahia – UFBA e nas imediações do Núcleo de Pesquisa em Redes de Computadores – NUPERC/UNIFACS, localizados no bairro de Ondina em Salvador. Essa tarefa contou com a participação de 12 alunos de graduação e pós-graduação, de cursos variados. Os participantes estão entre 18 e 25 anos de idade, possuem dispositivo *Android*, versão 3.2 ou superior, e estão acostumados a utilizar aplicações baseadas em serviços de localização.

Para a avaliação, anotações de problemas foram criadas e posicionadas em locais estratégicos, nas regiões onde o experimento foi realizado (loais onde os participantes costumam trafegar). Essas anotações simularam o informe colaborativo de problemas como travessia perigosa, tráfego intenso de motocicletas, entre outros. Durante a passagem nesses pontos, o protótipo alertou a existência de tais problemas. Os participantes observaram o funcionamento da aplicação, bem como, sua própria reação diante da descoberta de problemas nos seus arredores.

4.1. Pontos positivos

O protótipo identificou 100% dos problemas anotados, emitindo os respectivos alertas, de acordo com o contexto. Os alertas foram emitidos com antecedência suficiente, de modo que o participante foi capaz de tomar conhecimento dos detalhes antes de chegar ao problema. O protótipo identificou múltiplos problemas simultaneamente (quando o participante estava localizado na intersecção de raios de ação de anotações). O protótipo mostrou no mapa a localização dos problemas, bem como, suas descrições.

Aproximadamente, 67% dos participantes declararam que usariam uma aplicação com o objetivo de YouOnAlert no seu dia a dia. Quando perguntados “Em que você espera que uma aplicação como essa possa lhe ajudar?”, aproximadamente 58% dos participantes declararam esperar que a aplicação ajude-os a evitar acidentes.

4.2. Pontos negativos

Foi observado que nas áreas limite do raio de ação de anotações (áreas localizadas na proximidade da borda da circunferência que representa o raio), o protótipo emitiu o alarme sonoro para o mesmo problema várias vezes. Isso ocorreu devido à imprecisão do sensor GPS, que, nesse local, acaba determinando a coordenada do dispositivo às vezes dentro e às vezes fora do raio de ação, comutando entre emissão e não emissão do alerta. Esse problema ocasionou outro inconveniente: por vezes o protótipo soou o alarme e não mostrou o problema na tela do mapa. Isso aconteceu porque o alarme foi emitido quando o GPS apontou a posição atual dentro do raio, e quando a tela do mapa foi exibida, o GPS apontava a posição atual fora do raio de ação da anotação.

Participantes mostraram insatisfação com o fato de o protótipo alertar o mesmo problema tanto na ida quanto na volta do percurso (não reconhecer que um problema já foi alertado para determinado usuário). Outro ponto negativo foi o impacto de desempenho no sistema, causado pelo uso do protótipo em *background*. Participantes relataram lentidão. Alguns participantes declararam que a aplicação é inconveniente por emitir muito alarmes.

4.3. Soluções encontradas

Para solucionar o problema da imprecisão do GPS, um temporizador foi adicionado ao Módulo de Alerta. Agora, o alarme só é emitido se faz mais de sete segundos que a marca saiu do raio de ação (tempo determinado através de testes exaustivos). Quando um alerta entra e sai do raio de ação, em virtude da imprecisão do GPS, isso acontece muito rapidamente (geralmente menos que sete segundos), portanto, considerando esse tempo, o Módulo de Alerta não soa o alarme sucessivamente.

A questão de não reconhecer quando um problema já foi alertado para determinado usuário, depende de aspectos de configuração. Por exemplo, uma maneira de resolver esse problema seria oferecer uma área de configuração, onde o usuário possa dizer se deseja receber apenas alertas para problemas inéditos. Isso também resolveria, em parte, a questão da emissão de muitos alarmes. Com relação à lentidão no sistema, percebida pelos participantes, uma forma de solucionar essa questão é rever a implementação do recurso de *background*.

5. Considerações finais e Trabalhos Futuros

Sistemas de apoio ao cidadão são de considerável importância. A estratégia de *crowdsourcing* possibilita a informação de problemas onde sensores e outros dispositivos não são comuns, ou não podem atuar. A emissão de alertas identificando riscos à população como catástrofes climáticas é uma realidade. Através desses alertas a população obtém conhecimento de eventos graves que podem lhes causar riscos.

Diante disso, um sistema que pode alertar cidadãos acerca de problemas do cotidiano das cidades é de considerável utilidade. Com o YouOnAlert (UOA) qualquer cidadão portando um *smartphone* pode receber alertas que indicam problemas das cidades, relatados por outros cidadãos. Dessa forma, o UOA pode ajudar os cidadãos a evitar incidentes que, ocasionalmente, podem ocorrer com indivíduos desavisados.

Como próximos passos da pesquisa pretende-se investigar a integração do aplicativo com bases colaborativas de anotações – por exemplo, a base do Projeto Spaces 4D – e proceder à criação do módulo de *crowdsourcing*. Essa parte do sistema será responsável pela produção, edição e consumo de anotações fornecidas pelos cidadãos. Com esses componentes, será possível realizar testes mais robustos e, conseqüentemente, obter resultados mais relevantes. Por exemplo, pode-se considerar o perfil do usuário como uma entrada do módulo de contexto. Isso permite contextualizar melhor as preferências do usuário na frequência e forma de emissão dos alertas.

O registro automático de anotações também será explorado em trabalhos futuros. Por exemplo, ao invés de registrar uma anotação de “buraco na pista” somente por meio da interação do usuário com o Módulo de Crowdsourcing, pode-se criar esse registro levando em conta outros recursos do dispositivo. Por exemplo, pode-se usar o acelerômetro do dispositivo para detectar a existência de um buraco na pista e efetivamente anotar esse problema de forma automática.

Referências

Berners-Lee, T. (2006). “Design Issues: Linked Data.”, Disponível em <http://www.w3.org/DesignIssues/LinkedData.html/> (last viewed Mai 28, 2013).

- Brabham, C. (2008). "Crowdsourcing as a Model for Problem Solving. An Introduction and Cases". In *Int. Journal of Research into New Media Technologies*, 14(1):75-90.
- Castillo, A., Buzeto, F., Castanho, C and Jacobi, R. (2011). "Um modelo para o gerenciamento de ontologias em ambientes ubíquos.", In *Proc. of III Simpósio Brasileiro de Computação Ubíqua e Pervasiva (SBCUP 2011)*.
- Coleman, D., Georgiadou, Y. and Labonte, J (2009). "Volunteered Geographic Information: the nature and motivation of producers". In *Proc. of the International Journal of Spatial Data Infrastructures Research*, Vol. 4, 332-358.
- Dockhorn, P. and Rizzi, C. (2012). "Uma Plataforma de Testes para Aplicações Sensíveis a Contexto." In *Proc. of IV Simpósio Brasileiro de Computação Ubíqua e Pervasiva (SBCUP 2012)*, Curitiba. Anais de eventos satélite CSBC, 2012.
- Erickson, T. (2010). "Geocentric Crowdsourcing and Smarter Cities: Enabling Urban Intelligence in Cities and Regions". In *Proc. of 1st International workshop on ubiquitous crowdsourcing*.
- Furtado, V. et al. (2010). "Collective intelligence in law enforcement – The WikiCrimes system". In *Proc. of Elsevier Information Sciences* 180(1):4-17.
- Gonzales, A., Izidoro, D., Willrich, R. and Santos, C. (2013). "Representação Aberta e Semântica de Anotações de Incidentes em Mapas Web". In *Proc. of IX Simpósio Brasileiro de Sistemas de Informação*. v. 1. p. 1-12.
- Goodchild, F. (2007). "Citizens as Sensors: The World of Volunteered Geography". In *Proc. of GeoJournal*, 69(2007), 211-221.
- Haklay, M. and Weber, P. (2008). "OpenStreetMap: User-Generated Street Maps". In *Proc. of IEEE Pervasive Computing* 7 (4): 12-18.
- Loureiro, F., Oliveira, R. and Silva, T. (2009). "Computação Ubíqua Ciente de Contexto: Desafios e Tendências". In *Proc. of 27º Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos*.
- Ovelgonne, M., Sonnenbichler, C. and Geyer-Schulz, A. (2010). "Social Emergency Alert Service - A Location-Based Privacy-Aware Personal Safety Service,". In *Proc. of Next Generation Mobile Applications, Services and Technologies (NGMAST)*.
- Qureshi, H., Keyani, S., Babar, Q. and Mumtaz, A. (2011). "Monitoring Disease Outbreak through Geographical Representation in Rural Areas". In *Proc. of Developments in E-systems Engineering (DeSE)*, pp. 30-35.
- Shah, S., Bao, F., Lu, C. and Chen, I. (2011). "Crowdsafe: crowd sourcing of crime incidents and safe routing on mobile devices", In *Proc. of the 19th ACM SIGSPATIAL Int. Conf. in Geographic Information Systems*, pp. 521–524.
- Ushahidi. (2013). Disponível em <http://www.ushahidi.com/> (last viewed Mai 28, 2013).
- Vieira, R., Martinello M., Ramos, R. and Marcondes, C. (2010). "Privacidade de Localização: Uma abordagem baseada em médias aleatórias", In *Proc. of II Simpósio Brasileiro de Computação Ubíqua e Pervasiva (SBCUP 2010)*.
- Vukovic, M., Kumara, S. and Greenshpan, O. (2010). "Ubiquitous crowdsourcing". In *Proc. of the 12th Ubiquitous computing - Adjunct (UbiComp '10 Adjunct)*. ACM.