

# Ferramentas para Coleta Móvel de Dados \*

Alvaro H. Mamani-Aliaga, Bruno G. dos Santos, Jimmy V. Sánchez,  
Tiago Barabasz, Matheus F. Mendonça e Arlindo F. da Conceição<sup>1</sup>

<sup>1</sup> Instituto de Ciência e Tecnologia (ICT)  
Universidade Federal de São Paulo (UNIFESP)  
Rua Talim, 330. São José dos Campos - SP

alvaroma@ime.usp.br,

{bruno.gsantos89, sirghost, tbarabasz, mts.seven}@gmail.com e

arlindo.conceicao@unifesp.br

**Abstract.** *The usage of mobile devices is growing rapidly. Actually, it is increasingly common to use digital forms for mobile data acquisition using mobile devices (smartphones and tablets). This paper presents the project Maritaca, an open solution for create mobile applications to data gathering using Android mobile devices. By using the platform, no programming skill is needed to create the mobile applications. Moreover, the platform facilitates the sharing of the collected data in the cloud, thus allowing the formation of social networks for gathering and information sharing. The project was designed to support a heavy volume of users and data acquisition. It is based on free software and can be accessed at: `maritaca.unifesp.br`.*

**Resumo.** *O uso de dispositivos móveis vem crescendo rapidamente. No cenário atual é cada vez mais comum o uso de formulários eletrônicos móveis para a aquisição de dados através de dispositivos móveis (smartphones e tablets). Este trabalho apresenta o projeto Maritaca, uma solução aberta para a geração de aplicações para coleta de dados a partir de dispositivos móveis Android. O sistema permite a criação automática de aplicações para a coleta móvel de dados, mesmo por usuários sem conhecimentos técnicos de programação. Além disso, a plataforma facilita o compartilhamento dos dados coletados na nuvem, permitindo, assim, a formação de microrredes sociais de coleta e compartilhamento de informações. O projeto foi desenhado para comportar um grande volume de usuários e de dados. Baseado em software livre, o sistema pode ser acessado no endereço: `maritaca.unifesp.br`.*

## 1. Introdução

Nos últimos anos, o mercado de comunicação pessoal móvel evoluiu rapidamente. Essa evolução foi marcada principalmente por três fatores: a queda dos preços, o lançamento de dispositivos móveis com alta capacidade de processamento e o surgimento de novas tecnologias para o desenvolvimento de Aplicações Móveis (Apps). Estes fatores, combinados, criaram condições apropriadas para o surgimento de uma nova categoria de aplicações: a **Coleta Móvel de Dados (CMD)** [Rezende et al. 2010].

---

\*Pesquisa apoiada pela FINEP, processo 04.11.0077.00, referência 1488/10.

Em sistemas de informação, as aplicações de CMD são usadas em muitos cenários. Em geral, elas substituem coletas tradicionais de dados, usando lápis e papel, por coletas realizadas a partir de dispositivos móveis.

Entretanto, apesar do avanço das tecnologias de *hardware* e *software*, a criação de aplicações para coleta móvel de dados continua exigindo o trabalho de programadores e de profissionais de Tecnologia da Informação (TI). Em outras palavras, apesar dos avanços, ainda é preciso programar os questionários eletrônicos. Atualmente, essa é a principal limitação para a ampla utilização de CMDs, pois nem toda empresa possui recursos, quer humanos, quer financeiros, para elaborar aplicações móveis [de Andrade et al. 2012]. Este é o caso, por exemplo, da maioria das pequenas e micro empresas no país. Pode-se chegar a essa conclusão analisando-se o número de sistemas de informações móveis que foram descontinuados por falta de recursos para manutenção de software.

Para enfrentar ou reduzir estas limitações e, desse modo, contribuir para a ampla utilização de ferramentas móveis em sistemas de informação, foi desenvolvido o **Projeto Maritaca**, uma infraestrutura para a geração de aplicações de Coleta Móvel de Dados (CMD). O nome Maritaca vem do acrônimo *MARitaca Is a Tool to creAte Cellular phone Applications*. Desse modo, buscou-se permitir o desenvolvimento de aplicações móveis para coleta de dados sem a necessidade de intervenção de profissionais de TI para a criação e manutenção das aplicações móveis. Além disso, o projeto é de código aberto e altamente escalável.

O restante deste trabalho está organizado da seguinte maneira: a Seção 2 define o escopo dessa pesquisa; a Seção 3 apresenta os trabalhos relacionados; a Seção 4 descreve uma visão geral da arquitetura; a Seção 5 descreve as principais componentes da interface e os serviços presentes na plataforma; a Seção 6 descreve algumas aplicações diretas e indiretas da plataforma. Por fim, são apresentados os trabalhos futuros, as considerações finais e os agradecimentos.

## 2. Objeto do trabalho

A meta inicial do Projeto Maritaca era simplificar o desenvolvimento de aplicações de CMD. No estágio atual da pesquisa, o objetivo principal consiste em prover uma infraestrutura completa para CMD, o que a inclui: a criação simplificada das Apps para coleta, o armazenamento em nuvem dos dados coletados e a disponibilização de ferramentas que auxiliem o compartilhamento dos dados. Buscou-se, sobretudo, permitir o desenvolvimento de aplicações móveis para coleta de dados sem a necessidade de intervenção de profissionais de TI para a criação e manutenção das aplicações móveis.

## 3. Trabalhos Relacionados

Existem algumas ferramentas com propósitos similares aos do Maritaca. A ferramenta *App Inventor*<sup>1</sup> permite construir visualmente aplicativos para a plataforma Android. Concentra-se no desenho passo-a-passo de elementos de interface, conectando-os aos respectivos eventos. O diferencial positivo do projeto Maritaca em relação ao *App Inventor* está na simplicidade de utilização. O Maritaca provê uma forma mais simples e intuitiva para o desenvolvimento de interfaces voltadas para CMD, para fazer isso, oferece

---

<sup>1</sup><http://appinventor.mit.edu>

interfaces gráficas e controles arrastáveis que permitem a geração de aplicações sem a necessidade de profissionais de TI.

O *Nokia Data Gathering*<sup>2</sup> é um sistema para criação de questionários móveis usando dispositivos *Nokia*. Os dados são colocados em um servidor na Internet e podem ser acessados pelos dispositivos móveis com acesso a rede. Os dados são coletados e armazenados nos celulares e podem ser transmitidos para o servidor. No entanto, trata-se de uma solução para dispositivos específicos de um fabricante.

Outro trabalho a ser citado é o Open Data Kit (ODK)<sup>3</sup>, que consiste em um conjunto de ferramentas de código-fonte aberto que auxiliam a criação e o gerenciamento de coleta móvel de dados. Ele é composto por três ferramentas: *Build*, *Collect* e *Aggregate*. A ferramenta *Build* é utilizada para a modelagem dos formulários. A *Collect* é utilizada para a realização da coleta dos dados nos dispositivos móveis, que possuem o sistema operacional *Android*, bem como também para o envio destes ao servidor. Por fim, a ferramenta *Aggregate* reúne os dados coletados existentes no servidor e os converte para formatos mais usuais. Cabe salientar que a ferramenta móvel *Collect* gera as interfaces do aplicativo móvel a partir de um arquivo em formato *XForm*, um padrão de formatação de formulários especificado na linguagem XML. Entretanto, o ODK não oferece infraestrutura para a construção de formulários, mas sim define padrões e APIs.

O projeto *DoForms*<sup>4</sup> é um sistema para criação de questionários móveis multiplataforma, cuja proposta é semelhante ao Projeto Maritaca, entretanto, de código fechado. No *DoForms* há uma cota livre para usuários de até 200 MB, ultrapassado esse limite é cobrada uma taxa mensal por dispositivo móvel utilizado.

#### 4. Arquitetura do Sistema Maritaca

O projeto Maritaca foi desenvolvido como uma aplicação em nuvem [Armbrust et al. 2010, Cook et al. 2011, Nurmi et al. 2009], prevendo o uso do modelo de “software como serviço”.

Os dados coletados a partir de dispositivos móveis são armazenados em nuvem e são visualizados através de um navegador web. Nesta seção será apresentada a arquitetura do projeto, suas componentes e o modelo de integração entre as componentes.

##### 4.1. Componentes da Plataforma Maritaca

A Figura 1 apresenta as principais componentes da arquitetura e ilustra a relação entre essas componentes.

Os elementos apresentados na Figura 1 são:

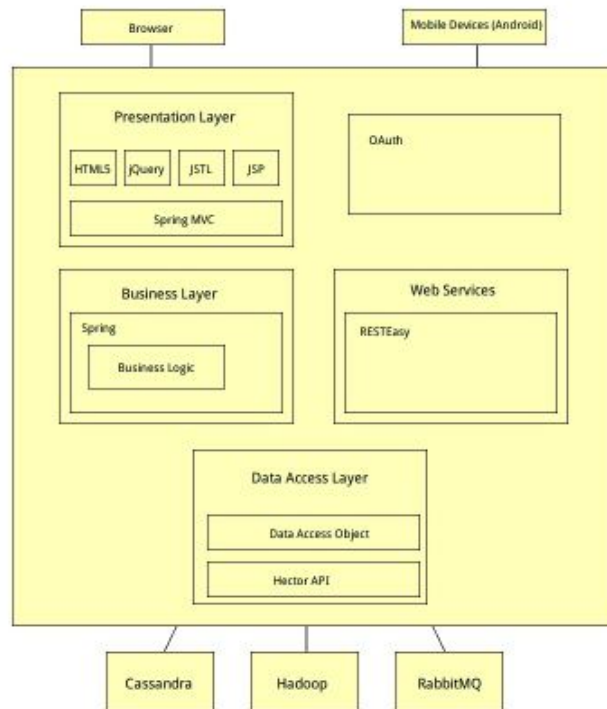
- **Servidor de Aplicações:** a parte servidora do projeto utiliza o servidor de aplicações *JBoss* [Davis and Marrs 2005], onde são hospedados os serviços e componentes Web. Toda a parte servidora, exceto alguns *scripts* de manutenção, foram implementados em Java utilizando o arcabouço *Spring* [Tate and Gehlert 2005]. Todos os *webservices* implementados utilizam a abordagem de serviços *RESTful* [Richardson and Ruby 2007]. Apesar de estar

---

<sup>2</sup><http://projects.developer.nokia.com/ndg/wiki>

<sup>3</sup><http://opendatakit.org/>

<sup>4</sup><http://www.doforms.com/>



**Figura 1. Componentes da solução em nuvem.**

embutido na plataforma Web, o Editor de Formulários é uma aplicação Web independente, escrita em HTML5 e Ajax, permitindo o desenvolvimento rápido e intuitivo dos questionários. Esta componente gera como resultado um **descritor de questionário**, que é gravado em formato XML.

- **Componente móvel:** é uma aplicação Android que interpreta o arquivo XML (descritor da coleta de dados) e renderiza as interfaces automaticamente. Trata-se de uma *Engine* de interpretação, baseada no padrão de projeto *Interpreter* [Gamma et al. 1994].
- **Servidor de Dados Cassandra:** componente para armazenamento escalável baseado no paradigma *NO-SQL*. Utilizamos o Cassandra para persistir, basicamente, os dados coletados e estruturados em formato XML.
- **Sistema de arquivos Hadoop** [Shvachko et al. 2010]: um sistema de arquivos distribuído e escalável. É utilizado no projeto para armazenar dados não estruturados, principalmente arquivos, tais como Apps, imagens, áudio e vídeo.

#### 4.2. Servidor de Aplicações

No módulo servidor encontram-se implementados os serviços e funcionalidades Web do projeto. Dentre as principais pode-se citar: armazenamento de dados estruturados e não estruturados (conteúdo multimídia e Apps), os serviços RESTful utilizados para a comunicação entre o servidor e o dispositivo móvel, o editor de formulários, entre outros serviços.

A interação entre o usuário e a plataforma Web, via um navegador, é feita através da camada Web do sistema, que utiliza principalmente Spring MVC, JQuery e HTML5.

A interação entre dispositivos móveis e a plataforma é realizada sempre utilizando-se de serviços RESTful. A camada web e os serviços RESTful interagem com a camada de lógica do negócio utilizando o arcabouço *Spring*.

### 4.3. Módulo Móvel

O usuário pode editar os questionários a partir de um navegador padrão e, uma vez editado, o questionário pode ser preenchido a partir do dispositivo móvel, onde os dados ficam temporariamente armazenados até serem transferidos para o servidor.

O Módulo Móvel consiste em uma *engine* que traduz o descritor do questionário (representado em formato XML) em uma hierarquia de objetos instanciados responsáveis pela renderização das interfaces e validação dos dados. O modelo computacional utilizado para representar os questionários é a parte mais sofisticada do componente móvel e foi o desenvolvimento tecnológico que viabilizou a solução.

A técnica de mapeamento do XML em uma lista de objetos está inspirada no padrão de projeto *Interpreter* [Gamma et al. 1994]. A aplicação móvel é uma controladora de contexto, que sempre faz referência para um objeto em uso, no caso do Maritaca o objeto em questão trata-se de um objeto do tipo *Question* [Durham et al. 2003]. O objeto é instanciado conforme o tipo da pergunta e cada tipo de objeto implementa as suas respectivas políticas, por exemplo, de validação de campo ou de exibição do controle.

#### 4.3.1. Autenticação

Para ser utilizada a partir do dispositivo móvel, a aplicação móvel requer que o usuário autentique-se no servidor; deste modo, garante-se que o usuário tenha autorização para coletar dados para aquele formulário. Esse processo é feito utilizando o protocolo de autorização OAuth [Tassanaviboon and Gong 2011] sobre o protocolo HTTPS. A Figura 2 ilustra os passos para a autenticação segundo o padrão OAuth: (i) primeiro, o usuário faz uma requisição de autorização para o servidor, com isso, o servidor redireciona o usuário para uma tela de autenticação; (ii) se o usuário é autenticado com sucesso, gera-se um código, esse passo é chamado de “confirmação da autorização”; (iii) com este código, o usuário pode gerar um *token*, que precisa ser renovado após um determinado tempo de validade; (iv) por fim, de posse de um *token* válido para o formulário, o usuário pode coletar, salvar e visualizar os dados. Essa transferência de dados é feita utilizando-se de serviços RESTful, seja nos formatos JSON ou XML.



Figura 2. Modelo de autenticação entre o componente Móvel e o Servidor utilizando OAuth.

### 4.3.2. Formato do arquivo XML de integração

O principal método de integração entre as componentes é o formato *Extensible Markup Language* (XML) gerado pelo Editor de Formulários. O arquivo contém as questões do formulário representadas como *tags*.

Cada *tag* possui os seguintes atributos básicos: *id*, *next*, *previous*, *required*, *label*, *help* e *type*. Os campos *id*, *next* e *previous* são do tipo numérico, sendo que *id* identifica o número da pergunta atual, *next* é o *id* da próxima pergunta e *previous* é o *id* da pergunta anterior a atual. O campo *required* define se a questão é obrigatória, sendo seu valor definido como: *true* ou *false*. O campo *label* deve conter a pergunta a ser mostrada no questionário. O atributo *help* é não obrigatório e contém um texto de ajuda ou de esclarecimento sobre a pergunta. O atributo *type* define o tipo de resposta, que pode assumir, por exemplo, os seguintes valores: *text*, *number*, *radiobutton*, *combobox*, *multimedia*, *gps* etc.

Além disso, alguns tipos de *tag* podem possuir estruturas condicionais, utilizadas para definir a navegação condicional entre as perguntas. A navegação entre as perguntas é definida no rótulo *condition*, onde a resposta da pergunta atual é utilizada para determinar a próxima pergunta a ser exibida. Por exemplo, considere a seguinte questão: “Qual a sua idade?” Se a resposta for um valor inferior a 18 anos, a próxima questão poderia ser: “Qual o nome do seu responsável?” Caso contrário, essa pergunta poderia ser omitida.

### 4.4. Construção automática de aplicações

Sempre que um formulário é salvo, o sistema gera um novo aplicativo Android (arquivo em formato APK) e o armazena no sistema de arquivos distribuído *Hadoop*. Esse processo de compilação e montagem leva alguns segundos, mas, como é realizado em *background*, não afeta negativamente a usabilidade do sistema.

Essa não foi a primeira abordagem adotada. Inicialmente planejava-se criar uma única aplicação Android, na qual descritores XML seriam carregados para cada aplicação de coleta. Entretanto, notou-se que esse método era menos seguro, dificultava o processo de divulgação de aplicações e a manutenção das versões das Apps.

### 4.5. Metodologia de Desenvolvimento

O processo de desenvolvimento do projeto foi baseado na metodologia ágil conhecida como Programação eXtrema (XP) [Beck and Andres 2004]. A prática dessa metodologia foi fundamental no processo de desenvolvimento. Do lado dos desenvolvedores, todos estão envolvidos e têm conhecimento de todas as camadas do código, principalmente devido à programação em par. Outro princípio aplicado foi o desenvolvimento orientado a testes, onde para cada funcionalidade desenvolvida são desenvolvidos também testes unitários e de integração.

## 5. Funcionalidades do Sistema

Esta seção descreve as funcionalidades oferecidas pela Plataforma Maritaca. O sistema permite que um usuário construa aplicações para coleta de dados, que podem ser instaladas em quaisquer dispositivos móveis compatíveis com Android, versão 2.2 ou superior.

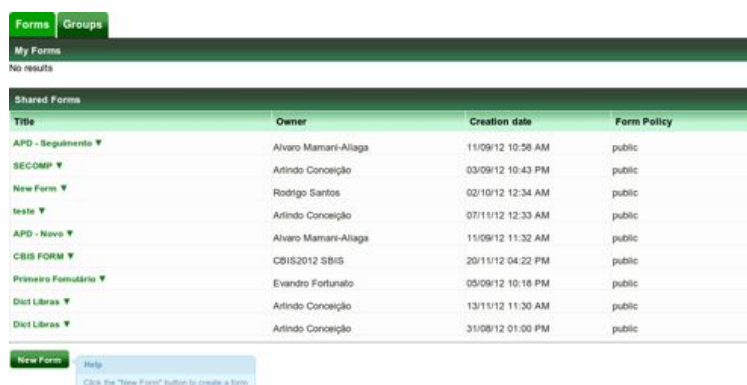
No dispositivo móvel, a App permite a coleta de dados utilizando interfaces amigáveis. Os dados são armazenados no dispositivo móvel até serem transferidos para o servidor. Para realizar a coleta não é necessário que o usuário esteja conectado à Internet.

A seguir são apresentados os principais passos para a utilização da plataforma.

### 5.1. Passo-a-passo para utilização da solução

Os passos a seguir exemplificam a utilização da plataforma:

1. O sistema está disponível em <http://maritaca.unifesp.br>. Para utilizá-lo, o usuário necessita autenticar-se na tela inicial do sistema, o que pode ser realizado de duas maneiras: (i) utilizando usuário e senha registrados no sistema (ii) ou com uso de OpenID [Recordon and Reed 2006].
2. Após a autenticação, é apresentada ao usuário a tela principal do sistema, destinada ao gerenciamento de questionários. A lista de questionários está organizada entre formulários criados pelo próprio usuário (parte superior) e formulários com ele compartilhados (parte inferior). Há ainda um menu com abas na parte superior para o gerenciamento de grupos de usuários. A Figura 3 apresenta a interface para gerenciamento de formulários.



Title	Owner	Creation date	Form Policy
APD - Seguinte	Alvaro Mamani-Alaga	11/09/12 10:58 AM	public
SECOMP	Artindo Conceição	03/09/12 10:43 PM	public
New Form	Rodrigo Santos	02/10/12 12:34 AM	public
teste	Artindo Conceição	07/11/12 12:33 AM	public
APD - Novo	Alvaro Mamani-Alaga	11/09/12 11:32 AM	public
CBIS FORM	CBIS2012 SBIS	20/11/12 04:22 PM	public
Primeiro Formulário	Evandro Fortunato	05/09/12 10:18 PM	public
Dict Libras	Artindo Conceição	13/11/12 11:30 AM	public
Dict Libras	Artindo Conceição	31/08/12 01:00 PM	public

Figura 3. Tela inicial de gerenciamento de formulários.

3. Para criar um novo formulário, basta pressionar o botão *New Form*, no canto inferior esquerdo da tela. A Figura 4 ilustra a interface para criação e edição de formulários. A interface utiliza controles HTML5 arrastáveis (*Drag and Drop*), que permitem ao usuário “arrastar” as componentes a serem utilizadas no questionário. O usuário pode escolher um tipo de pergunta ou dado que queira coletar dentre o conjunto de componentes disponíveis na caixa de ferramentas localizada no lado esquerdo da interface. Na figura apresentada como exemplo, o formulário que está sendo criado possui três perguntas.
4. Uma vez criado o formulário, o usuário pode descarregá-lo para um dispositivo móvel com sistema operacional Android. A instalação do aplicativo no dispositivo móvel é trivial. A Figura 5(a) mostra a tela de autenticação na aplicação móvel. A Figura 5(b) mostra o menu principal da aplicação, a partir da qual inicia-se a coleta clicando-se no botão *Collect*.
5. A Figura 6 mostra a interface (especificada na Figura 4) renderizada automaticamente no dispositivo móvel. Cada pergunta é mostrada em uma tela e cada tela é

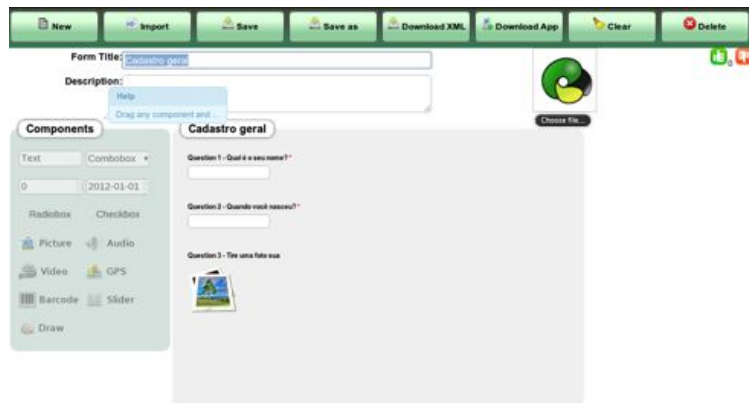


Figura 4. Editor de Formulários, preenchido.



(a) Tela de autenticação

(b) Tela inicial

Figura 5. Aplicação móvel

organizada em duas partes principais: acima, com o título do formulário, botões auxiliares (*ajuda* e *cancelar*), e barra de progresso que indica quantas perguntas do questionário já foram respondidas; e abaixo, ocupando o restante da tela, a questão em si. Os controles de navegação (*retornar* e *avançar*) podem ser executados através de comandos *touch*<sup>5</sup> ou pressionando o botão de menu do celular.

6. A coleta de dados pode ser realizada inúmeras vezes. Cabe reforçar que para coletar os dados não é necessário que o dispositivo móvel esteja conectado à Internet. Uma vez coletados os dados, o usuário pode conectar-se à Internet e enviar as coletas realizadas para o servidor. Uma vez armazenados no servidor, os dados podem ser visualizados acessando-se a plataforma.

## 5.2. Os tipos de pergunta

Um questionário é composto por uma ou mais perguntas. Cada pergunta pode ser de diferentes tipos, tais como: texto, numérico, data, múltipla escolha (*radio button*), seleção

<sup>5</sup>*Touch Screen*: Tecnologia de tela sensível ao toque, em que o usuário interage com a interface do aplicativo pressionando a ponta do dedo na tela do aparelho móvel, podendo fazer gestos predeterminados para executar alguma tarefa.





(a) Primeira questão, do tipo texto. (b) Segunda questão, do tipo data. (c) Terceira questão, do tipo foto.

**Figura 6. Perguntas renderizadas no dispositivo móvel.**

múltipla (*combo box* e *check bok*), foto, áudio, vídeo, localização, código de barras, porcentagem (controle deslizante ou *slider*) e desenho. Pretende-se implementar ainda outros tipos, pois uma das características mais positivas da plataforma, graças ao seu desenho modular, é a facilidade para se criar e integrar ao sistema novas componentes.

Além disso, o usuário pode personalizar valores padrão (*default*) para alguns tipos de campos a serem coletados, configurar validações para os valores inseridos e definir a ordem de navegação entre as perguntas em função dos valores inseridos.

### 5.3. Compartilhamento de dados

Atualmente, o sistema permite criar 3 classes de formulários: privados, públicos e compartilhados. Como os nomes indicam:

- O formulário **privado**: pode ser utilizado apenas pelo seu criador;
- O formulário **público**: pode ser visto por qualquer usuário da plataforma;
- Os formulários **compartilhados** podem ser de dois subtipos: **hierárquico** e **social**;
  - o formulário **compartilhado hierárquico**: o seu criador pode convidar, por exemplo, os usuários A e B, mas os dados coletados por A não são visíveis por B e vice-versa;
  - por sua vez, no formulário **compartilhado social**: os usuários A e B podem ver os dados coletados um pelo outro.

De forma similar aos formulários, as respostas (informações coletadas) de cada formulário são afetadas de acordo a política de compartilhamento adotada. A Tabela 1 resume as políticas implementadas no projeto e como cada política influencia tanto no formulário quanto nas coletas geradas.

		<b>Privado</b>	<b>Comp. Hierárquico</b>	<b>Comp. Social</b>	<b>Público</b>
<b>Questionário</b>	Visualização	Criador	Criador e Lista	Criador e Lista	Todos
	Modificação	Criador	Criador	Criador	Criador
<b>Respostas</b>	Visualização	Criador	Criador	Criador e Lista	Todos
	Coleta	Criador	Criador e Lista	Criador e Lista	Todos

**Tabela 1. Políticas de compartilhamento de dados no Maritaca.**

## 6. Cenários de Aplicação

O projeto tem aplicação em diversos campos de ação direta. Por exemplo, a plataforma pode ser usada em diferentes domínios de aplicações sociais. Um deles é o uso da plataforma como ferramenta de apoio à coleta de dados em programas de atenção básica à saúde que envolvam visitas domiciliares. Com isso, os dados coletados por esses programas podem ser revisados e avaliados sem perda de informação no processo de redigitação e transcrição de dados.

Outro exemplo de aplicação é o uso da plataforma para construir uma base de dados sobre a linguagem brasileira de sinais (LIBRAS) [Digiampietri et al. 2012].

Por fim, planeja-se o uso da plataforma para o acompanhamento nutricional de grupos de pacientes que são convidados a utilizar as *Apps* para registrar fotograficamente a sua alimentação cotidiana.

O número de aplicações possíveis do Maritaca é muito amplo. Estes exemplos são uma pequena amostra de situações relevantes onde o projeto pode ser aplicado. Abaixo enumeramos algumas aplicações potenciais menos óbvias ou indiretas.

### 6.1. Aplicações: Maritaca como ferramenta para formação de Redes Sociais

Uma rede social é uma estrutura de relacionamentos entre atores com propósitos comuns (pessoais, acadêmicos, profissionais etc.) [Gomes 2009]. Observe que o Maritaca pode ser utilizado para a formação do que chamamos de micro redes sociais. Redes locais criadas, para fins específicos, que envolvam coleta de dados.

Utilizando-se a plataforma pode-se formar pequenas redes sociais, onde o usuário pode criar e compartilhar o formulário com os seus contatos, pode convidar novos participantes, pode gerar coletas e baixar as coletas disponíveis, entre outras funcionalidades.

### 6.2. Aplicações: Maritaca como ferramenta para geração de Inteligência Coletiva

Segundo [Lévy 1994], Inteligência Coletiva pode ser definida da seguinte maneira:

“É uma inteligência distribuída por toda parte, incessantemente valorizada, coordenada em tempo real, que resulta em uma mobilização efetiva das competências. Acrescentemos à nossa definição com este complemento indispensável: a base e o objetivo da inteligência coletiva são o reconhecimento e o enriquecimento mútuo das pessoas, e não o culto de comunidades fetichizadas ou hipostasiadas.”

A plataforma Maritaca pode permitir a criação e extração de Inteligência Coletiva a partir das redes complexas formadas por usuários de aplicações de CMD. Os dispositivos móveis são muito comuns na sociedade atual, se utilizados de forma coordenada para captura de dados podem fornecer dados com alto valor agregado.

Atualmente trabalha-se para oferecer a funcionalidade de construção de Inteligência Coletiva. Essas alterações visam permitir a criação simples, flexível e gratuita de aplicações colaborativas capazes de oferecer informações globais, construídas por centenas ou milhares de usuários.

Algumas das formas de aproveitar a inteligência coletiva no projeto são:

- Busca (*Search*): o sistema mostrará os resultados de pesquisa mais relevantes usando o perfil de um usuário.
- Avaliações, opiniões e mecanismos de recomendação: a informação coletada dos usuários influenciará outros usuários.
- Votação e *tagging*: os usuários podem qualificar os formulários e, possivelmente, as respostas, com isso pode-se extrair informações com valor agregado sobre os usuários e sua rede.

## 7. Conclusões e Trabalhos Futuros

O Projeto Maritaca é uma solução aberta para a Coleta Móvel de Dados. O projeto foi iniciado a mais de quatro anos e evoluiu de um gerador automático de aplicações para uma solução completa que permite a coleta, o armazenamento e o compartilhamento de dados tanto estruturados quanto não estruturados. Acreditamos que o sistema resolve um problema prático: a redução de custos de produção de sistemas móveis.

A aplicabilidade do projeto é ampla. Atualmente, pretende-se utilizar o projeto no “Programas Saúde da Família”, onde, mensalmente, são coletados dados sociais e de saúde das famílias acompanhadas pelo programa. São coletados dados tais como o número de nascimentos, gestantes, hospitalizações, óbitos etc. Nosso objetivo é utilizar o Maritaca para essa coleta de dados.

Além disso, atualmente estamos trabalhando no desenvolvimento de um editor de relatórios. O objetivo é construir uma ferramenta adicional que permita configurar, sem programar, relatórios consolidados de dados. Por exemplo, além de construir uma aplicação capaz de coletar os preços de um produto e os locais de venda, deseja-se consultar onde o produto é vendido com o menor preço.

## Referências

- Armbrust, M., Fox, A., Griffith, R., Joseph, A., Katz, R., Konwinski, A., Lee, G., Patterson, D., Rabkin, A., Stoica, I., et al. (2010). A view of cloud computing. *Communications of the ACM*, 53(4):50–58.
- Beck, K. and Andres, C. (2004). *Extreme programming explained: Embrace change*. Addison Wesley.
- Cook, N., Milojicic, D., and Talwar, V. (2011). Cloud management. *Journal of Internet Services and Applications*, pages 1–9.
- Davis, S. and Marrs, T. (2005). *JBoss at work: A practical guide*.
- de Andrade, S. C., Tait, T. F. C., de Oliveira, F. M., and Ribeiro, M. B. (2012). Caracterização do gerenciamento de projetos de software para m-business. In *SBSI-Simpósio Brasileiro de Sistemas de Informação*.
- Digiampietri, L., Teodoro, B., Santiago, C., Oliveira, G., and Araújo, J. (2012). Um sistema de informação extensível para o reconhecimento automático de libras. In *SBSI-Simpósio Brasileiro de Sistemas de Informação*.
- Durham, A., Sussumu, E., and da Conceição, A. (2003). A framework for building language interpreters. In *Companion of the 18th annual ACM SIGPLAN conference on Object-oriented programming, systems, languages, and applications (OOPSLA). Educators Symposium*, page 196. ACM.

- Gamma, E., Helm, R., Johnson, R., and Vlissides, J. (1994). *Design Patterns*. Addison-Wesley Professional.
- Gomes, E. A. (2009). Segurança em aplicações p2p através de reputação inferida de redes sociais.
- Lévy, P. (1994). *A Inteligência Coletiva por uma antropologia do ciberespaço*. La Découverte.
- Nurmi, D., Wolski, R., Grzegorzczak, C., Obertelli, G., Soman, S., Youseff, L., and Zagorodnov, D. (2009). The eucalyptus open-source cloud-computing system. In *Proceedings of the 2009 9th IEEE/ACM International Symposium on Cluster Computing and the Grid*, pages 124–131. IEEE Computer Society.
- Recordon, D. and Reed, D. (2006). OpenID 2.0: a platform for user-centric identity management. In *Proceedings of the second ACM workshop on Digital identity management*, pages 11–16, New York, NY, USA. ACM.
- Rezende, J. V. P., Silva, B. N. M., and da Conceição, A. F. (2010). Plataforma para desenvolvimento simples e flexível de questionários para Coleta Móvel de Dados (CMD). In *I Workshop de Pesquisa e Desenvolvimento em Software Livre (WPeDSL)*, Natal-RN.
- Richardson, L. and Ruby, S. (2007). *RESTful web services*. O’Reilly Media, Incorporated.
- Shvachko, K., Kuang, H., Radia, S., and Chansler, R. (2010). The hadoop distributed file system. In *Proceedings of the 2010 IEEE 26th Symposium on Mass Storage Systems and Technologies (MSST)*, pages 1–10, Washington, DC, USA. IEEE Computer Society.
- Tassanaviboon, A. and Gong, G. (2011). OAuth and ABE based authorization in semi-trusted cloud computing: aauth. In *Proceedings of the second international workshop on Data intensive computing in the clouds*, DataCloud-SC ’11, pages 41–50, New York, NY, USA. ACM.
- Tate, B. and Gehtland, J. (2005). *Spring: a developer’s notebook*. O’Reilly Media, Incorporated.