

**Universidade Federal da Paraíba**  
**Centro de Informática**  
**Ciência da Computação**  
**Linguagem de Programação**

**Projeto – Etapa 2**

Arnaldo Gualberto de Andrade e Silva – 10921064

Hugo Neves de Oliveira – 10921073

José Ivan Vilarouca Filho – 10911040

**João Pessoa  
2013**

## **1. Descrição das Classes**

Segue abaixo uma descrição resumida das principais classes do projeto organizadas por pacote.

### **1. Pacote *Controller***

- **SQLiteHelper.java:** faz as principais operações sobre o banco de dados local, como abrir conexão, executar query, etc. Todas as classes que usam o banco de dados local, utilizam essa classe.
- **BabyController.java:** faz as operações CRUD na tabela Baby do banco de dados local com ajuda da classe SQLiteHelper.java.
- **HistoricController.java:** faz as operações CRUD na tabela Baby do banco de dados local com ajuda da classe SQLiteHelper.java.
- **ReminderController.java:** faz as operações CRUD na tabela Baby do banco de dados local com ajuda da classe SQLiteHelper.java.
- **TakenVaccineController.java:** faz as operações CRUD na tabela Baby do banco de dados local com ajuda da classe SQLiteHelper.java.
- **VaccineController.java:** faz as operações CRUD na tabela Baby do banco de dados local com ajuda da classe SQLiteHelper.java.

### **1.2 Pacote Model**

- **Baby.java:** representa um objeto bebê com seus atributos e métodos correspondentes.
- **Book.java:** representa um objeto livro a ser retornado nas dicas de livros.
- **BookAdapter.java:** Base adapter para a lista de livros retornados da consulta ao servidos externo.
- **Historic.java:** possui uma lista que encapsula as operações sobre o histórico da criança.
- **Progress.java:** classe que representa todo o progresso de desenvolvimento do bebê.
- **Reminder.java:** representa os alarmes de uma vacina e sua dose para cada bebê.
- **TakenVaccine.java:** representa as vacinas marcadas como tomadas de um bebê.
- **Tip.java:** usada para representar as dicas de cada categoria do aplicativo na view “Dicas”.
- **Vaccine.java:** representa todas as vacinas do aplicativo. No aplicativo, há dois tipos de vacinas: default e não-default. As defaults, são as que já vem com o aplicativo , não podem ser removidas pelo usuário e só é permitida a edição do

mês que a vacina foi tomada. As não-defaults são as vacinas adicionadas pelo usuário. Todas as operações são permitidas nesse último tipo.

- **WeigthHeigth.java:** representa os atributos de um histórico de um bebê, são eles: peso, altura e mês.

### 1.3 Provider

- **BabyProvider.java:** provê acesso aos dados dos bebês para outras aplicações.

### 1.4 View

Todas as classes desse pacote representam cada activity do aplicativo]

### 1.5 Custom

- **Page.java:** define a interface que representa uma view de paginação, como para as imagens dos gráficos na view GraphicsActivity.java do pacote *view*
- **ImagePage.java:** implementa a interface *Page* para as imagens dos gráficos
- **TipAdapter.java:** base adapter para a lista de dicas

### 1.6 Webservice

- **AddRemoteBaby.java:** classe que executa, em background, a operação de adicionar o bebê remotamente no web service.
- **JSONParser.java:** responsável por fazer o parser do objeto json recebido de uma URL.
- **LoginAuthenticatorBaby.java:** classe que executa, em background, a operação de login do usuário.
- **RemoveRemoteBaby.java:** classe que executa, em background, a operação de adicionar o bebê remotamente no web service.