

INSTITUTO FEDERAL DE MINAS GERAIS
CAMPUS SÃO JOÃO EVANGELISTA
PÓS GRADUAÇÃO LATO SENSU EM MEIO AMBIENTE

JEAN MONTEIRO LIMA

DRPN: SOFTWARE DE DIAGNÓSTICO DO RISCO DE POLUIÇÃO DE
NASCENTES

SÃO JOÃO EVANGELISTA

2017

JEAN MONTEIRO LIMA

**DRPN: SOFTWARE DE DIAGNÓSTICO DO RISCO DE POLUIÇÃO DE
NASCENTES**

Trabalho de conclusão de curso apresentado ao Instituto Federal de Minas Gerais – Campus São João Evangelista como exigência para obtenção do título do Título de Especialista em meio Ambiente.

Orientadora: Profa. Dra Graziele Wolff de Almeida Carvalho

SÃO JOÃO EVANGELISTA

2017

FICHA CATALOGRÁFICA

L732d Lima, Jean Monteiro.
2017

DRPN: Software de Diagnóstico do Risco de Poluição de Nascentes.
/ Jean Monteiro Lima. – 2017.

85f.

Trabalho de Conclusão de Curso (Pós-graduação Lato Sensu em
Meio Ambiente) – Instituto Federal de Educação, Ciência e Tecnologia
de Minas Gerais – Campus São João Evangelista, 2017.

Orientadora: Grazielle Wolff de Almeida Carvalho.

1. Software. 2. Águas. 3. Preservação. I. Lima, Jean Monteiro.
II. Instituto Federal de Educação, Ciência e Tecnologia de Minas
Gerais – Campus São João Evangelista. III Título.

CDD 004.2

Elaborada pela Biblioteca Professor Pedro Valério

Instituto Federal de Educação, Ciência e Tecnologia de Minas Gerais
Campus São João Evangelista

Bibliotecária Responsável: Rejane Valéria Santos – CRB-6/2907

JEAN MONTEIRO LIMA

**DRPN: SOFTWARE DE DIAGNÓSTICO DO RISCO DE POLUIÇÃO DE
NASCENTES**

Trabalho de conclusão de curso apresentado ao
Instituto Federal de Minas Gerais - Campus São
João Evangelista como exigência para obtenção
do Título de Especialista em meio Ambiente

Aprovada em 02 / 08 / 2017

BANCA EXAMINADORA

Orientadora Profa. Dra Grazielle Wolff de Almeida Carvalho
IFMG - SJE

Prof. Dr. Giuslan Carvalho Pereira
IFMG - SJE

Prof. Me. Fábio Rodrigues Martins
IFMG - SJE

:

RESUMO

Neste trabalho foi desenvolvido um sistema que facilita o trabalho do consultor ambiental na aplicação do índice de avaliação diagnóstica das águas das nascentes, conforme literatura do Índice de Risco à Poluição das Águas de Nascentes. Seguindo as fases de levantamento de requisitos, de análise e projeto, de implementação, de testes e implantação, e a de validação de software, foi construído o Aplicativo DRPN – Diagnostico do Risco de Poluição de Nascentes para uso em celular, que se mostrou como uma excelente ferramenta de auxílio ao trabalho do consultor quando utilizado na avaliação das nascentes do Instituto Federal de Minas Gerais, campus de São João Evangelista.

Palavras chave: *Software, Águas, Preservação.*

ABSTRACT

In this work a system was developed that facilitates the work of the environmental consultant in the application of the index of diagnostic evaluation of the waters of the springs, according to the literature of the Risk Index to the Water Pollution of the Spring. Following the requirements survey, analysis and design, implementation, testing and deployment, and software validation phases, the DRPN Application - Spring Pollution Risk Diagnosis was developed for use in mobile phones, which was shown as an excellent tool to assist the consultant's work when used in the evaluation of the sources of the Federal Institute of Minas Gerais, São João Evangelista campus.

Key words: Software, Water, Preservation.

LISTA DE FIGURAS

Figura 1 – Diagrama de Caso de Uso do DRPN -----	20
Figura 2 – Interface Principal -----	22
Figura 3 - Interface de Avaliação -----	22
Figura 4 – Diagrama de Classes do Banco de Dados – DAO -----	23
Figura 5 – Diagrama de Seqüência do Pacote do Banco de Dados -----	24
Figura 6 - Diagrama de Classe da pasta Principal do <i>Software</i> -----	24
Figura 7 - Projeto da tabela do Banco de Dados do <i>Software</i> -----	25
Figura 8 – Interface Principal do DRPN - 00 -----	28
Figura 9 – Interface de Avaliação da Nascente do DRPN- 00 -----	29
Figura 10 – Interface de Exclusão e Correção das avaliações de Nascentes do DRPN- 00---	30
Figura 11 – Área do Campus IFMG – São João Evangelista -----	31
Figura 12 – Nascente com Fluxo de Água no Campus -----	31
Figura 13 - Foto Nascente Paineira -----	32
Figura 14 - Foto Nascente Ypê Amarelo 1 -----	33
Figura 15 - Foto Nascente Braúna 1 -----	34
Figura 16 - Foto Nascente Peroba -----	35
Figura 17 - Foto Nascente Embaúba -----	36
Figura 18 - Foto Nascente Jacarandá -----	37
Figura 19 – Interface Principal do DRPN -00 -----	38
Figura 20 – Nascente Ypê Amarelo 2 -----	39
Figura 21 – Nascente Ypê Amarelo 3 -----	40
Figura 22 – Nascente Braúna 2 -----	40
Figura 23 – Nascente Braúna 3 -----	41
Figura 24 – Interface Principal do DRPN após Reavaliação das nascentes Braúna e Ype ---	42

LISTA DE TABELAS

Tabela 1- Parâmetros e Pesos do IIAN -----	26
Tabela 2- Parâmetros e Pesos do IVIAP -----	26
Tabela 3 – Classe de Pressões e Impactos Antropicos do IIAN -----	27
Tabela 4 – Classificação quanto vulnerabilidade intrínseca do IVIAP -----	27
Tabela 5 – Matriz de Classificação do Índice de Risco à Poluição das Águas de Nascentes-	28
Tabela 6 – Resumo das Avaliações das Nascentes do IFMG -----	38
Tabela 7 – Comparativo das Reavaliações das Nascentes Ype e Braúnas -----	41

LISTA DE QUADROS

Quadro 1 – Lista de Fluxos de Eventos do Software DRPN -----	21
--	----

LISTA DE ABREVEATURAS

CAR - Cadastro Ambiental Rural

DAO – Data Access Object .

DRPN – Diagnostico de Risco de Poluição de Nascentes

IDE - Ambiente de Desenvolvimento Integrado

IFMG – Instituto Federal de Minas Gerais

IIAN - Índice de Impacto Ambiental em Nascentes

IRPAN - Índice de Risco à Poluição das Águas de Nascentes

IVIAP - Índice de Vulnerabilidade Intrínseca dos Aquíferos à Poluição

MMA - Ministério do Meio Ambiente

SENAR- Serviço Nacional de Aprendizagem Rural

SQL - Structured Query Language

XML - eXtensible Markup Language

SUMÁRIO

1 - INTRODUÇÃO	9
2 - JUSTIFICATIVA	11
3 – OBJETIVOS	12
4 – REVISÃO DA LITERATURA	13
5 – METODOLOGIA	16
6 – DESENVOLVIMENTO	19
7 - AVALIAÇÃO DAS NASCENTE	31
8 - CONSIDERAÇÕES	43
REFERÊNCIAS BIBLIOGRÁFICAS	44
APÊNDICE A - CÓDIGO DAS CLASSE DO DRPN	46
APÊNDICE B - TELAS DO ANDROID STUDIO	83

1 - INTRODUÇÃO

Hoje há uma gama de conhecimentos científicos que demandam um ardo esforço para colocá-los em prática. Porém foram desenvolvidas ferramentas que visam auxiliar a humanidade na sintetização destes conhecimentos de forma que eles sejam utilizados para o bem comum da sociedade (PRETTO, 1999).

Estas ferramentas são os vários tipos de softwares que auxiliam as pessoas nos mais variados processos, fazendo-as interagir mais com o celular ou computador, tanto em casa, quanto no trabalho, resolvendo problemas comuns e diários.

Desta forma, estes avanços tecnológicos propiciam a inserção de inovações nos recursos metodológicos para a preservação do meio ambiente. Hoje em dia os *softwares* ambientais são uma ferramenta bastante usada para facilitar o trabalho dos consultores ou ambientalistas. Pois hoje se torna impossível ignorar os impactos antrópicos elucidados pelas ciências ambientais sobre a qualidade da vida no planeta e em especial da realidade ambiental no século XXI e a questão dos recursos hídricos e a necessidade de garantia desses para as gerações futuras.

Assim, Felipe (2009) faz um apontamento importante neste âmbito da resignificação do valor da água, corroborando com a função estrutural e determinante da mesma na localização espacial do homem e suas atividades:

(...) mais do que a compreensão da água enquanto elemento natural dotado de características físicas e químicas próprias, anseia-se, na vanguarda da epistemologia geográfica, conceber a água como um elemento ambiental. Suas funções social, econômica e natural devem ser compreendidas de forma conjunta e com o mesmo peso. (FELIPPE, 2009, p18).

Existe atualmente, uma grande oferta destes *softwares* no mercado, cabendo ao consultor ambiental analisar e decidir qual têm maior potencialidade de contribuir com o processo de análise ambiental de gerenciamento da água e que auxiliem na tomada de decisões sobre os problemas relacionados à escassez, e também à qualidade e bom uso da água, de modo que sejam analisados conjuntamente, de forma a fornecer soluções que atendam o contexto de oferta e demanda hídrica de determinado local (MAGALHÃES Jr, 2012).

Para uma gestão eficiente é necessário possuir dados concretos que respaldem as futuras ações a serem tomadas, por isso os indicadores de qualidade ambiental vêm ganhando espaço no cenário mundial.

Em termos conceituais, indicadores estão aqui sendo compreendidos como informações quantitativas, mas que não por isso perdem necessariamente eventuais características qualitativas, que possibilitem a descrição de um sistema, ou de um componente deste (UNESCO, 1984). Um índice, por sua vez, é considerado como um agrupamento de dois ou mais indicadores, relacionando valores de diferentes indicadores a um padrão observado em determinado sistema (UNESCO, 1984).

Pensando em informações que auxiliam na tomada de decisões o Ministério do Meio Ambiente (MMA) através do Programa Plantadores de Rios, lançou um aplicativo acoplado ao Cadastro Ambiental Rural (CAR) que permite a participação popular de pessoas e instituições que queiram investir na preservação e proteção de rios e nascentes.

Contudo, cabe ao consultor ambiental ainda escolher, planejar e construir um *software* ambiental, contextualizando-o a sua realidade ambiental e social. Desta forma o presente projeto pretende contribuir com o processo de recuperação e preservação de nascentes, atendendo às necessidades instrumentais dos consultores, ambientalista ou ainda empresas que atuam na área de meio ambiente.

Neste sentido foi desenvolvido de um *software* do índice de risco de poluição das águas das nascentes sintetizado por Filizzola (2013) a partir das críticas e adaptações de Felipe e Magalhães Jr. (2012).

2 - JUSTIFICATIVA

No presente trabalho, o conceito de nascentes utilizado foi proposto por Felipe e Magalhães Jr. (2009). Estes autores definem uma nascente como “um sistema ambiental marcado por uma feição geomorfológica ou estrutura geológica em que ocorre a exfiltração da água de modo temporário ou perene, formando canais de drenagem a jusante”.

Tendo em vista a delicadeza da questão das nascentes, principalmente em meio urbano, a realização de uma análise amparada por indicadores ambientais representa uma ferramenta valiosa para a gestão de nascentes e, por conseguinte, de recursos hídricos.

Os indicadores ambientais podem apresentar limitações, principalmente no que se trata de oferecer valores claros para processos que sejam permeados por aspectos subjetivos integrantes da realidade e relacionados à dimensão cognitiva (MAGALHÃES Jr, 2012). Apesar disso, a utilização dos indicadores como ferramenta é extremamente útil e passível de ser integrada às ações práticas que visem à gestão de recursos hídricos.

No que toca o processo de exfiltração, Davis (1966) elenca três variáveis principais determinantes, sendo elas: a permeabilidade do aquífero, a área de recarga do aquífero e a quantidade de recarga. Além desses aspectos naturais, as interações do homem (usos e atividades) com as nascentes geram as pressões e impactos no sistema, representando um ponto fundamental para a compreensão real da situação das nascentes, principalmente quando se encontram em ambientes com elevada concentração humana ou com usos e atividades de elevado potencial de degradação.

Diante da problemática exposta, esse trabalho buscou vincular a capacidade diagnóstica dos índices e indicadores aplicados a recursos hídricos determinados a partir índice de risco de poluição das águas das nascentes sintetizado por Filizzola (2013), com o recorte espacial do sistema ambiental das nascentes com o uso do *software* desenvolvido com o intuito de facilitar o trabalho do consultor ambiental.

3 - OBJETIVOS

3.1 - Objetivo Geral

O objetivo principal deste trabalho consiste em desenvolver um *software* do índice de risco de poluição das águas das nascentes sintetizado por Filizzola (2013) e fazer o uso do mesmo na avaliação das nascente do Campus de São João Evangelista.

3.2 - Objetivos específicos

- Modelar e implementar o *software* do índice de risco de poluição das águas das nascentes;
- Avaliar e corrigir erros o *software* do índice de risco de poluição das águas das nascentes;
- Avaliar as nascentes do Campus de São João Evangelista.

4- REVISÃO DA LITERATURA

4.1 - Poluição de Meio Ambiente

Primeiramente, a base para construir qualquer índice que reflita condições acerca da poluição é justamente uma conceituação sólida acerca desse termo. A legislação brasileira, na forma da Lei nº 6.938, de 31 de agosto de 1981, em disposição sobre a Política Nacional do Meio Ambiente, define poluição como:

(...) a degradação da qualidade ambiental resultante de atividades que direta ou indiretamente:

- a) Prejudiquem a saúde, a segurança e o bem-estar da população;
- b) Criem condições adversas às atividades sociais e econômicas;
- c) Afetem desfavoravelmente a biota;
- d) Afetem as condições estéticas ou sanitárias do meio ambiente;
- e) Lancem matérias ou energia em desacordo com os padrões ambientais estabelecidos (BRASIL, 1981. Lei nº 6.938).

Neste contexto “a alteração adversa das características do meio ambiente”. Fica evidente assim a amplitude que a legislação implica ao termo poluição. Nos moldes propostos ela pode ocorrer em qualquer parte do meio ambiente. Apesar dessa grande abrangência conceitual, que muitas vezes pode levar a perda de precisão quando trabalhando em casos pontuais.

Von Sperling (2005) aponta que poluição das águas pode ser considerada como “a adição de substâncias ou de formas de energia que, direta ou indiretamente, alterem a natureza do corpo d’água de uma maneira tal que prejudique os legítimos usos que dele são feitos”. Essa proposta possui um caráter prático relevante e nesse sentido, tendo em vista que o presente trabalho tem uma proposta de software que busca analisar o estado das nascentes, e em última instância levar a tomada de decisões para reverter quaisquer danos avaliados, a definição apontada por Von Sperling (2005) para poluição das águas foi considerada para fins desse trabalho.

4.2- Conceito de Nascente

Davis (1966), dentro da linha de pensamento da hidrogeologia, não pretende abordar o tema com muita ênfase, considerando como uma nascente “qualquer descarga natural de água

que seja grande o suficiente para fluir sob a forma de um pequeno canal”. A amplitude dessa conceituação impõe entraves para distinguir qualquer diferença entre nascentes.

Valente e Gomes (2005), conceitua uma nascente como manifestação do lençol freático, vinculada necessariamente com um curso d’água. O fato dos autores relacionarem de forma restritiva a ocorrência de nascentes com a presença de cursos d’água, revela um descuido com a observação prática e também certo teor do conceito popular de nascentes (FELIPPE, 2009).

Buscando iniciar a inversão desse quadro de deficiência conceitual das nascentes, ante a importância e abrangência dessas feições no ciclo hidrológico e hidrossocial, Felipe e Magalhães Jr. (2009), como já foi exposto aqui, definem uma nascente como “um sistema ambiental marcado por uma feição geomorfológica ou estrutura geológica em que ocorre a exfiltração da água de modo temporário ou perene, formando canais de drenagem a jusante”.

4.3 - Conceitos Sobre os Índices de Avaliação de Nascentes

a) Índice de Impacto Ambiental em Nascentes (IIAN) – O IIAN foi primeiramente proposto por Gomes et al. (2005), como uma ferramenta de avaliação qualitativa do grau de impacto antrópico sobre nascentes. No entanto, Felipe e Magalhães Jr. (2012) propuseram uma releitura crítica do IIAN original, e adaptaram o método de forma que o resultado foi um índice qualitativo do grau de proteção de nascentes e também fizeram mudanças que possibilitassem o cruzamento do IIAN com o IVIAP, resultando no Índice de Risco de Poluição das Águas da Nascente - IRPAN. Porém, a essência de atribuir de forma qualitativa o grau de proteção das nascentes continua a mesma.

b) Índice de Vulnerabilidade Intrínseca dos Aquíferos à Poluição (IVIAP) - Foi realizado uma adaptação do IVI de Nobre et al. (2008), de forma que os parâmetros foram adequados mais diretamente à dinâmica de nascentes. Os seis parâmetros originais foram reduzidos para três, que englobam de forma geral as principais características naturais dos meios hidrogeológico e geomorfológico que são preponderantes na análise da vulnerabilidade à poluição dos sistemas aquíferos que respondem pelas águas das nascentes.

c) Índice de Risco à Poluição das Águas de Nascentes (IRPAN) - O IRPAN como descrito anteriormente, resultou do cruzamento das pontuações do IIAN e do IVIAP, na forma índice de risco de contaminação das águas superficiais e subsuperficiais das nascentes.

Onde o resultado da classificação encontra respaldo na interpretação de cada parâmetro do IIAN e do IVIAP, e na consideração dos cenários possíveis resultantes da interação dos parâmetros dos dois subíndices.

4.4 - Desenvolvimento de Software

Segundo Sommerville (2003), o desenvolvimento de *software* apresenta estágios iniciais de especificação do sistema até a sua manutenção, bem como depois que *software* entrar em operação. O ramo da ciência que gerencia todos estes aspectos é a engenharia de *software* que se ocupa de todos os aspectos da produção e comercialização. Foram elaborados vários modelos de processos desenvolvimento de *software* e nenhum pode ser considerado o ideal, devido às suas divergências.

Entretanto, para Sommerville (2003), todos compartilham de atividades fundamentais como especificação, projeto e implementação, validação e evolução. Segundo Pressman (1995), “*software* não se desgasta”. Exatamente por não se tratar de peça física, como o hardware, o *software* não se desgasta. Não existem peças de reposição de um *software*. Apesar da não existência de desgaste, existe a depreciação, onde o *software* sofre alterações em relação à mudança do seu escopo ou mesmo correções de programação que acabam comprometendo a sua vida.

5 - METODOLOGIA

A proposta do projeto de desenvolvimento de *software* já trouxe consigo o tema “diagnóstico de risco de poluição da nascente”. Também já apresentou inicialmente alguns requisitos básicos que o *software* dispõe como:

- permitir o cadastro das nascentes;
- permitir o acesso ao banco de dados;
- formular as perguntas sobre a qualidade ambiental da nascente;
- possuir interface gráfica para escolha da resposta;
- Emitir registros e avaliação sobre a qualidade ambiental da nascente avaliada, bem como possibilidade de imprimir relatórios;

Porém foram realizadas muitas outras etapas para o desenvolvimento do *software* ambiental que atenderam os objetivos de proteção ambiental. Segundo Bezerra (2002) é necessário que se prossigam algumas atividades que podem ser assim classificadas: levantamento de requisitos; análise e projeto; implementação; testes e implantação; avaliação; validação. Cada uma destas etapas encontra-se detalhada a seguir:

5.1 - Fase de levantamento de requisitos

Nesta fase foi determinados quais os requisitos que o sistema deve ter. Para tanto, discute-se as funcionalidades do sistema a partir de estudo das necessidades dos usuários. Dentre as técnicas utilizadas, destacou-se o estudo teórico acerca da temática, entrevistas com especialistas na área de domínio do *software*, e comparação com outros sistemas já existentes sobre o mesmo assunto/domínio de abrangência.

Ainda segundo Bezerra (2002), o produto desta etapa é um documento, escrito em notação informal (linguagem natural), abordando:

a) Requisitos funcionais, que delineiam as funcionalidades do sistema (exemplo: o sistema deve permitir acesso aos usuários através de cadastro, mediante *login* e senha e outras funcionalidades que o sistema deve ou não permitir tanto à nível de usuário, quanto de administrador).

b) Requisitos não-funcionais, que declaram as características do sistema em termos de confiabilidade (tolerância à falhas), desempenho (tempo de resposta esperado para as

funcionalidades do sistema), portabilidade (restrições sobre plataformas de *hardware* e *software*, facilidade de transportar sistema para outras plataformas).

c) **Restrições** em relação a custos, prazos, aspectos legais, plataforma tecnológica ou aquisição de novos *softwares*/equipamentos.

O levantamento dos requisitos necessários para a definição do *software* proposto foi realizado a partir do estudo dos tutoriais ambientais, do conceito de *software* ambiental a partir da análise das teorias dos índices de qualidade ambiental. O *software* foi desenvolvido para rodar no sistema operacional ANDROID, uma vez que o será utilizado pelo consultor ambiental no seu celular no trabalho de campo.

5.2 - Fase de análise e projeto

Durante esta etapa os problemas na construção do *software*, foram identificados e analisados conforme suas fontes e também foram determinadas as possíveis soluções. Esta etapa envolveu a definição de como alcançar os objetivos determinados durante a análise e o projetar o *software*.

5.3 - Fase de Implementação

Esta etapa teve como suporte as fases de análise e de projeto. O objetivo foi gerar materiais (telas) que auxiliaram o desenvolvimento do tutorial das funcionalidades do diagnóstico de risco de poluição da nascente, bem como todo o seu *design*. Esta fase refere-se ao efetivo desenvolvimento do *software* por meio da aplicação dos recursos tecnológicos identificados nas fases anteriores. Foi feita uma modelagem básica seguindo as especificações dos requisitos levantados, em seguida foi elaborado um diagrama de classes usando a *IDE Astah Community* e codificação com a *IDE Android Studio*. No desenvolvimento do aplicativo, utilizou-se dos recursos da linguagem de programação XML, JAVA, SQL e Padrão DAO – *Data Access Object*.

5.4 - Fase de testes e implantação

Essa etapa caracteriza-se pelo processo de testes que o *software* deve passar para entrar em funcionamento. A implantação consistiu em instalar o *software* no celular para que seja iniciada sua utilização, testando-o e fazendo-o funcionar adequadamente, segundo os

parâmetros estabelecidos no projeto. Bezerra (2002) ao se reportar a etapa de testes no desenvolvimento de um *software*, destaca que apesar da existência de uma etapa específica para testagem, ela deve ocorrer ao longo do processo e, sobretudo, quando houver alterações no *software*. Nesta fase, corrigiram-se os erros e falhas do *software*.

5.5 - Fase de validação

Durante a validação buscou-se medir a eficiência do *software*, porém as coletas dos dados ocorrem através da prática com a utilização do mesmo. Para isso, utilizou a entrevista não estruturada, validada por colegas do curso de Especialização em Meio Ambiente do IFMG de São João Evangelista que utilizaram o *software*. Foi verificado o grau de interesse dos mesmos no momento em que eles interagiram com o *software* e as impressões dos colegas sobre as vantagens proporcionadas. Com as informações colhidas foram gerados relatórios para análise, também foi realizada a avaliação de todas as nascentes do Campus de São João Evangelista de modo a auxiliar no diagnóstico ambiental das mesmas.

6 - DESENVOLVIMENTO DA PROPOSTA

6.1 – Identificação dos Sujeitos

Os *softwares* são desenvolvidos para auxiliar na solução das necessidades humanas. Assim através da experiência como consultor ambiental e depoimentos de demais profissionais que atuam na área foi possível concluir que é muito subjetiva a avaliação dos impactos ambientais, cujos pesos variam muito de consultor para consultor.

Uma maneira de minimizar esta variação desenvolveu-se índices que exigem dos profissionais um conhecimento bastante teórico que o mesmo deve levar à campo no momento de suas análises.

Desta forma, um *software* abordando o **Índice de Risco à Poluição das Águas de Nascentes** é de extrema importância para o processo de avaliação e preservação ambiental, devido a praticidade de sua utilização em campo.

6.2 - Compreensão do Processo para o Desenvolvimento

Para atender a necessidades dos consultores ambientais quanto a avaliação das nascentes é necessário o desenvolvimento de um sistema que possua funcionalidades que atendam os objetivos pelos os quais foi idealizado.

Estas funcionalidades são descritas através de requisitos funcionais e não funcionais, onde os requisitos funcionais estão relacionados às funcionalidades executadas pelo sistema, o seu comportamento diante das entradas e saídas dos dados. Já os requisitos não funcionais dizem respeito às restrições para que essas funcionalidades sejam executadas dentro dos padrões de qualidade esperados.

6.2.1 - Levantamento de Requisitos

Partindo de entrevista dialogada com os colegas do curso e consultores identificou-se os Requisitos Funcionais descritos na lista:

- O sistema deve dispor de uma interface de apresentação sobre suas funcionalidades;
- O sistema deve permitir o acesso público a qualquer usuário
- O sistema deve criar um banco de dados e tabelas através de Sistema de Gerenciamento de Banco de Dados;

- O sistema deve emitir relatório da avaliação da nascente quanto ao risco a poluição;
- O sistema deve permitir a correção da avaliação feita na nascente e salvará o resultado no banco de dados;

- O sistema emitirá automaticamente ao final da avaliação o resultado da nascente para que o consultor tenha conhecimento do risco de poluição das águas;

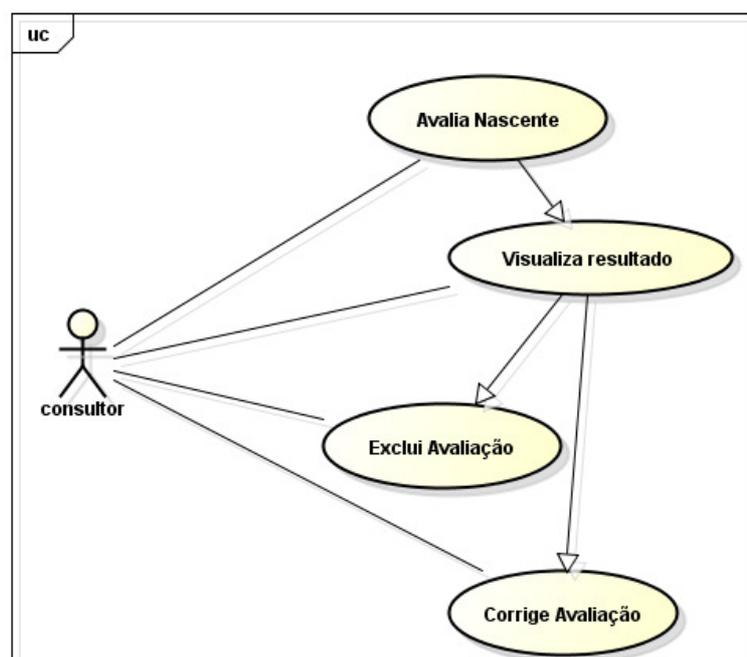
Em relação aos Requisitos não Funcionais temos como características de confiabilidade ótima uma vez que trabalha com servidor no próprio celular *SQLite*, de bom desempenho, pois é executado em qualquer celular, *tablet* que tenha o sistema operacional *Android* instalado, de portabilidade e segurança compatível com nível de segurança determinada pelo usuário.

Já em relação as Restrições do sistema deve-se verificar a atualização do *Android* e do *SQLite* pois o sistema foi desenvolvido para ter banco de dados local.

6.2.2 - Análise do Projeto

Depois de identificada a viabilidade para prosseguir com o desenvolvimento do software foi elaborado o Diagrama de Caso de Uso, Lista de Fluxos de Eventos, *Interface*, que apoiaram na compreensão do contexto no qual os processos de funcionamento do *software* e de sua interação com o usuário (consultor).

Figura 1 – Diagrama de Caso de Uso do DRPN



Fonte: Astah Community

Na Figura 1, temos o diagrama de Caso de Uso do *Software* desenvolvido, onde através de elementos gráficos e textuais podemos entender as relações do atores (consultor) com sistema, bem como as principais funcionalidades do sistema.

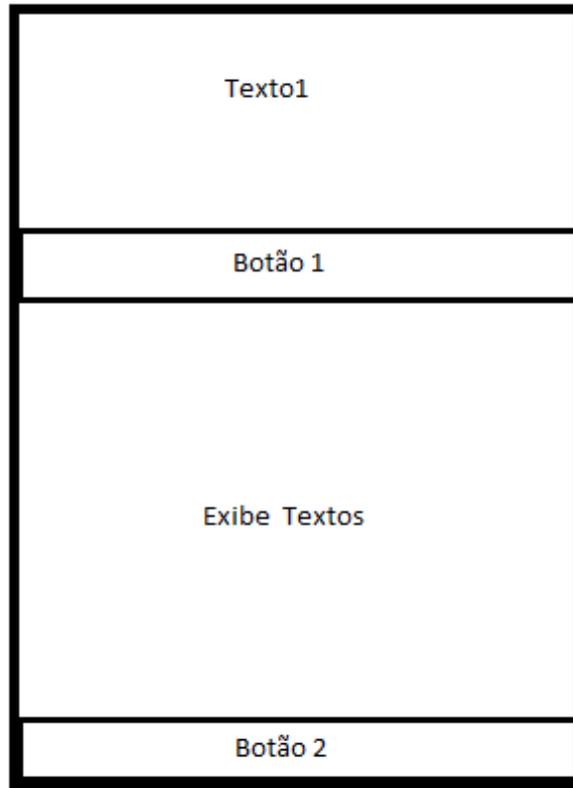
Para um melhor entendimento de como o sistema irá funcionar foi confeccionado Lista de Fluxos de Eventos (Quadro 1) para uma descrição mais detalhada do Caso de Uso.

Quadro 1 – Lista de Fluxos de Eventos do Software DRPN

<u>Lista de Fluxos de Eventos</u>
1 - Qualquer usuário (consultor) que tiver instalado o aplicativo pode iniciar o sistema;
2 - O sistema inicia sua interface com a tela de apresentação e com informações sobre o que é o Software, com todas as nascentes avaliadas que estão salvas no banco de dados e dois botões “Avaliar Nova Nascente” e “ Sair”;
3 - O usuário (consultor) clica no botão “Avaliar Nova Nascente” e tem acesso a interface de avaliação com um questionário a ser preenchido;
4 - O usuário (consultor) insere as informações solicitadas no formulário de questionário e clica em “Avaliar Nascente”;
5 - O sistema verifica, cria o objeto com o nome da nascente no banco de dados “Nascente”, armazena as informações do objeto dentro da tabela e exibe o resultado da avaliação na interface inicial do aplicativo;
6 - O usuário (consultor) escolhe na interface inicial, uma das nascentes já avaliadas clicando sobre ela;
7 - O sistema exibe uma tela de mensagem com duas opções de escolha, “Editar”, que permite acesso novamente ao questionário de avaliação da nascente para correção de mesmo e alteração do resultado da avaliação, e” Excluir” , que apaga a nascente do banco de dados.
8 - O usuário clica no botão “Sair” o aplicativo fecha salvando todas as nascentes avaliadas no banco de dados interno no celular.

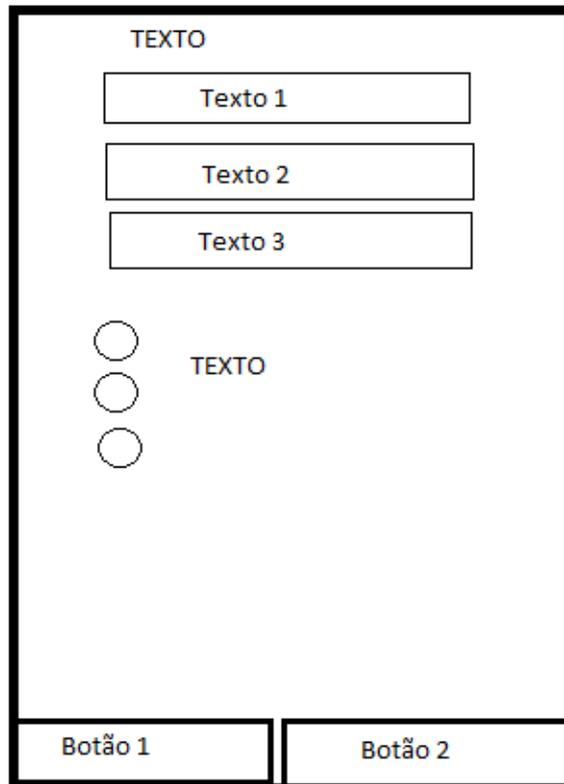
De forma a levar o entendimento de todo o funcionamento do *software* foi elaborado uma seqüência de protótipos de *interfaces* com o objetivo de ilustrar este funcionamento. Assim, através das alternativas as interfaces podem ser avaliadas e, ao mesmo tempo, os usuários passam a ter uma idéia do funcionamento do sistema.

Figura 2 – Interface Principal



Fonte: Astah Community

Figura 3 - Interface de Avaliação

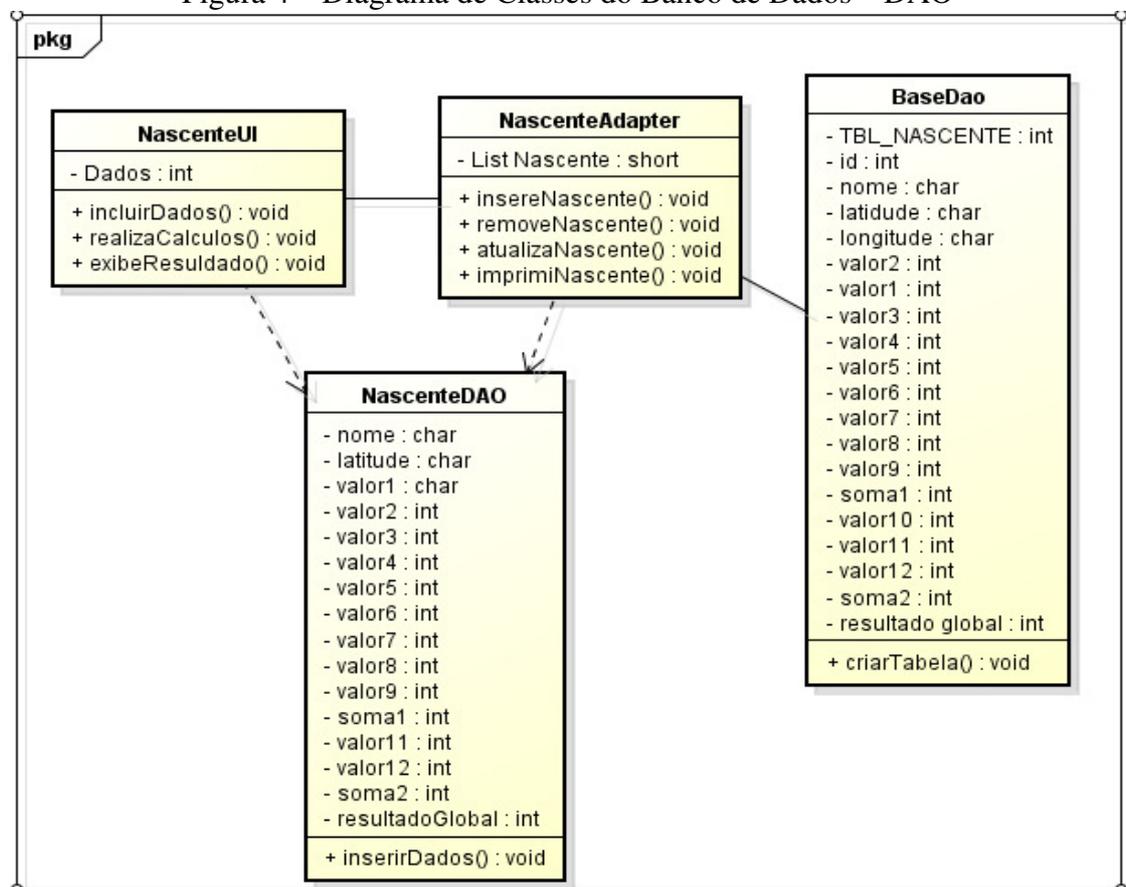


Fonte: Astah Community

6.2.3 – Projeto do Software

Nesta etapa foi desenvolvido artefatos que representam modelos do *software* que tem o foco nos aspectos da arquitetura, *interface* humano-computador, banco de dados e classes que compõem o sistema de forma a auxiliar na definição das tecnologias utilizadas na criação do sistema. Como resultado, desta etapa antecipou-se a identificação de possíveis defeitos no *software*, além de ter reduzido os custos de implementação e de garantir que determinados requisitos (funcionais e não funcionais) fossem atendidos. Devido a quantidade de classe que compõem o *software* esta forma representada em pastas(pacotes) conforme a finalidade.

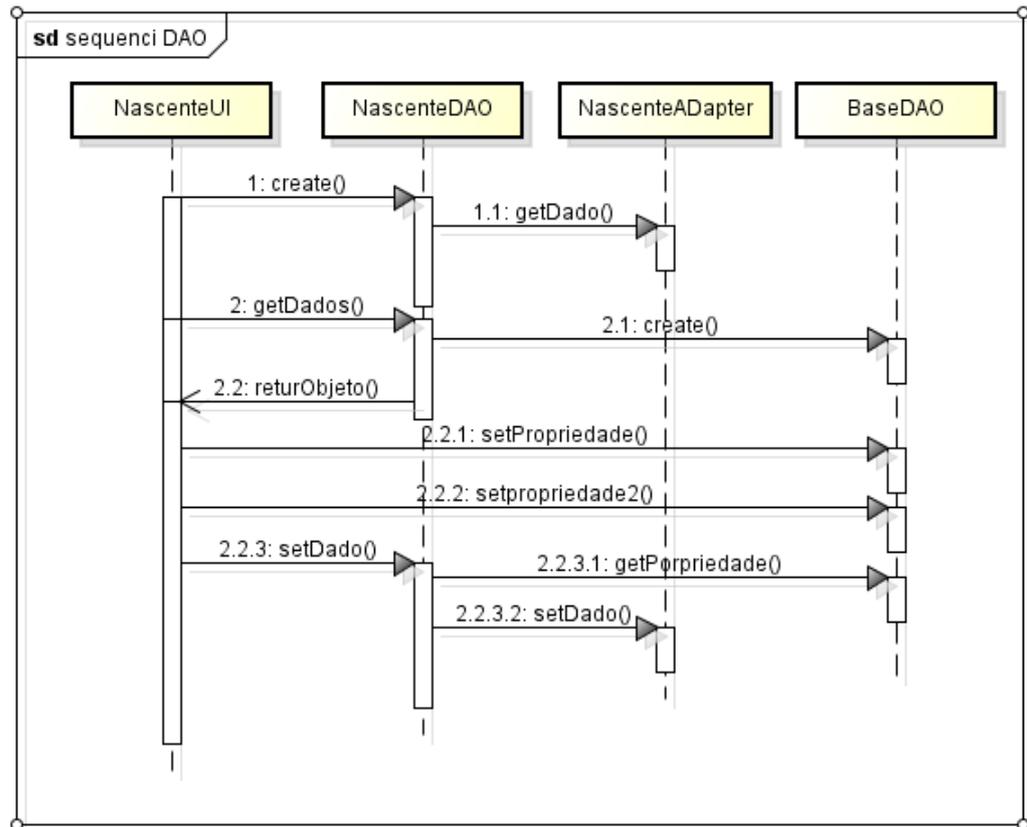
Figura 4 – Diagrama de Classes do Banco de Dados – DAO



Fonte: Astah Community

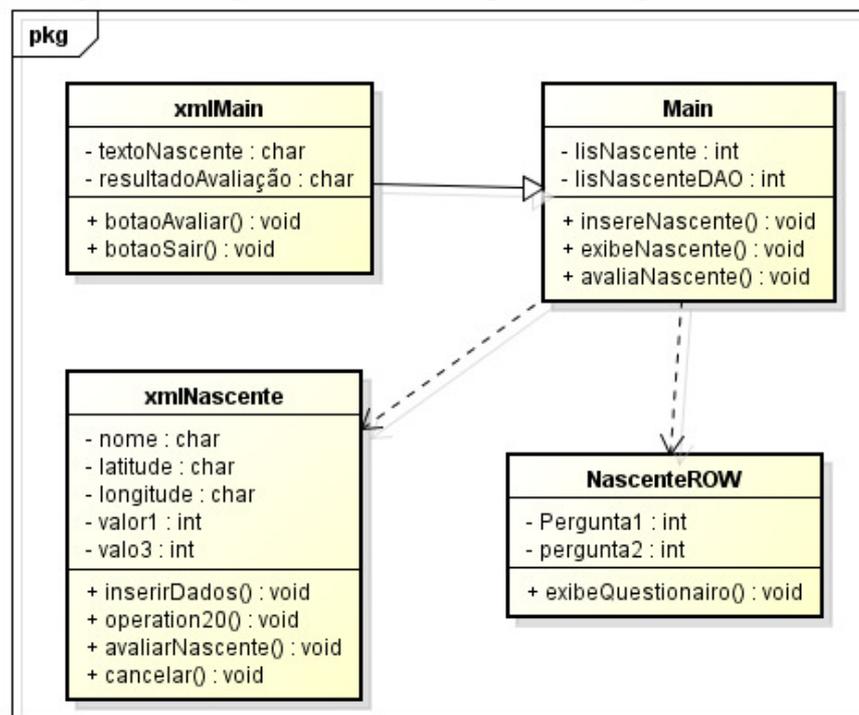
Na Figura 4, verifica-se que a Classe **NascenteAdapter** encapsula o acesso aos dados e contém os mapeamentos destes. A Classe **BaseDAO** é a origem dos dados: BD, XML. A Classe **NascenteUI** contém a lógica de negócio e usa o objeto **NascenteDAO**.

Figura 5 – Diagrama de Seqüência do Pacote do Banco de Dados



Fonte: Astah Community

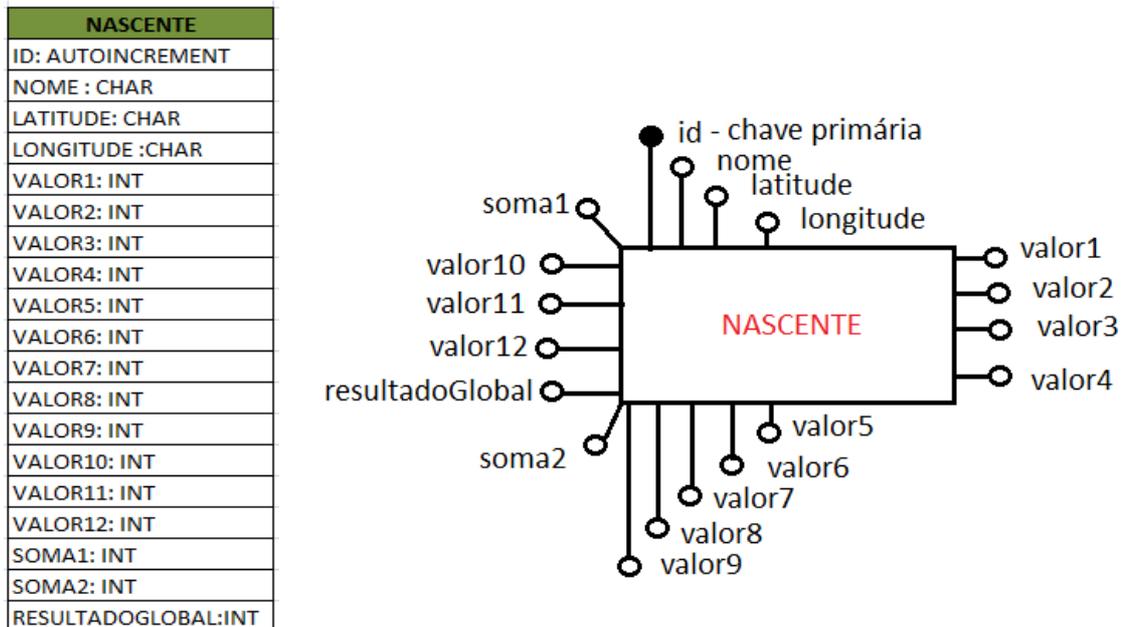
Figura 6 - Diagrama de Classe da pasta Principal do Software



Fonte: Astah Community

O projeto de banco de dados é composto apenas por uma tabela que será criada pelo *software*. Essa tabela será nomeada com a palavra “nascente” onde será armazenado todas as informações da avaliação da nascente realizada pelo consultor ambiental.

Figura 7 - Projeto da tabela do Banco de Dados do Software



Fonte: Astah Community

6.3 – Implementação do Projeto

Iniciou-se a implementação instalando todas as ferramentas necessárias para o desenvolvimento como: *IDE SQLite* que traz junto servidor local para gerenciar o banco de dados, e as IDEs *Astah Community* e *ANDROID STUDIO*.

O Índice de Risco de Poluição das Águas das Nascentes (IRPAN) como apresentado anteriormente e composto de parâmetros de dois índices, do qual 9 (nove) são do Índice de Impacto Ambiental em Nascentes (IIAN) que demonstram o quanto as ações antrópicas são propícias a causar danos a nascente e 3 (três) do Índice de Vulnerabilidade Intrínseca dos Aquíferos à Poluição (IVIAP) demonstram o quanto as características físicas (geológica, geomorfológica e pedológicas) facilitam ou não a poluição das águas da nascente.

Assim foi implementada três variáveis para cada um dos Parâmetros Macroscópicos do Índice de Impacto Ambiental da Nascente (IIAN) e Índice de Vulnerabilidade Intrínseca

dos Aquíferos à Poluição (IVIAP) com os seus respectivos valores de peso demonstrado nas tabelas a seguir:

Tabela 1- Parâmetros e Pesos do IIAN de Felipe e Magalhães Jr. (2012)

Parâmetros Macroscópicos com pesos	Qualificação		
	Ruim (1)	Médio (2)	Bom (3)
Odor da água (2)	forte	com odor	não há
Lixo ao redor da nascente (3)	muito	Pouco	não há
Materiais flutuantes (lixo na água) (5)	muito	Pouco	não há
Óleos e Graxas (5)	muito	Pouco	não há
Esgoto na nascente (5)	visível	provável	não há
Vegetação (4)	ausente	secundária (alterada, espécies invasoras)	primária (nativa, não alterada)
Uso da nascente (3)	dessedentação de animais	captação humana	não há
Acesso (1)	fácil	difícil	sem acesso
Equipamentos urbanos (3)	a menos de 50 metros	entre 50 e 100 metros	> 100 metros

Fonte: Adaptado de Fizzola(2013).

Tabela 2- Parâmetros e Pesos do IVIAP de Felipe e Magalhães Jr. (2012)

Parâmetros com pesos	Qualificação da vulnerabilidade		
	Alta (1)	Média (2)	Baixa (3)
Profundidade do Nível Freático (metros) (3)	0 - 10	10 – 30	>30
Infiltração e Recarga (4)	superfícies permeáveis, com topografia plana ou suavizada (0 a 6%) e cobertura vegetal cuja densidade seja favorável à infiltração	superfícies permeáveis em áreas com declividades moderadas a elevadas (6 a 12%) e cobertura vegetal densa ou moderadamente densa	superfícies pouco permeáveis, com declividades elevadas (>12%) e cobertura vegetal ausente ou rarefeita
Aquífero (permeabilidade) (5)	rochas permeáveis: aquíferos com permeabilidade relativamente elevada derivada da associação de variáveis como proporção de vazios primários, estruturas geológicas e/ou cavidades cársticas	rochas com permeabilidade relativamente intermediária derivada da associação entre as variáveis: proporção de vazios primários, estruturas geológicas e/ou cavidades cársticas em situação intermediária entre as categorias alta e baixa	rochas com permeabilidade relativamente baixa derivada da associação entre a proporção de vazios primários, estruturas geológicas e/ou cavidades cársticas.

Fonte: Adaptado de Fizzola(2013).

O sistema processa os valores das variáveis de entrada com três somas, onde primeiramente realiza a soma das variáveis dos parâmetros do IIAN e o resultado pode ser comparado com a classe de pressão e impactos que refletem, indiretamente, o grau de atenção necessário para a proteção de cada nascente na Tabela 3. Essa classificação se deu,

primeiramente, a partir do cálculo do valor máximo possível (87) da soma e do valor mínimo (31). Os valores máximos e mínimos possíveis foram divididos em quatro classes pelo método de intervalos iguais – A, B, C e D – sendo a classe E composta pelo valor mínimo.

Tabela 3 – Classe de Pressões e Impactos Antrópicos do IIAN de Felipe e Magalhães Jr. (2012)

Classe	Classes de Pressões e Impactos	Pontuação
A	Muito baixos	74 - 87
B	Baixos	60 - 73
C	Medianos	46 - 59
D	Elevados	32 - 45
E	Muito elevados	31

Fonte: Adaptado de Fizzola(2013).

Como o procedimento anterior o resultado da segunda soma foi enquadrado em 5 classes – A, B, C, D, E – de modo que a diferença entre o maior valor possível (36) e o menor (12) foi dividida por 4 classes – A, B, C, D – ficando a classe E composta pelo menor valor (12). Essas classes (intervalos iguais) fornecem o grau de vulnerabilidade intrínseca de aquíferos, conforme representado na Tabela 4.

Tabela 4 – Classificação quanto vulnerabilidade intrínseca do IVIAP de Felipe e Magalhães Jr. (2012)

Classe	Vulnerabilidade	Pontuação
A	Muito baixa	31 - 36
B	Baixa	25 - 30
C	Mediana	19 - 24
D	Elevada	13 - 18
E	Muito elevada	12

Fonte: Adaptado de Nobre et al. (2008).

Já a terceira e última soma que o sistema realiza fornece o Diagnóstico de Risco à Poluição de Nascentes (DRPN) que é resultado do cruzamento das pontuações do IIAN e do IVIAP, na forma de classes. Assim, tem 25 (vinte e cinco) possibilidades de cruzamentos das classes que foram enquadradas em uma matriz 5x5 resultando, por sua vez, em 7 (sete) classes que indicam o índice de risco de contaminação das águas, superficiais e subsuperficiais, das nascentes. Com esta análise é possível obter, por exemplo, uma classe de risco A (Muito

Baixo) quando tivermos uma nascente com classe A no IIAN, situada em aquífero de classe B quanto à vulnerabilidade, conforme exposto na Tabela 5.

Tabela 5 – Matriz de Classificação do Índice de Risco à Poluição das Águas de Nascentes

IVIAP/IIAN	A	B	C	D	E
A	Muito Baixo	Baixo	Médio-Baixo	Médio-Baixo	Médio
B	Baixo	Baixo	Médio-Baixo	Médio	Médio-Alto
C	Médio-Baixo	Médio-Baixo	Médio	Médio-Alto	Médio-Alto
D	Médio-Baixo	Médio	Médio-Alto	Alto	Alto
E	Médio	Médio-Alto	Médio-Alto	Alto	Muito Alto

Fonte: Adaptado de Fizzola(2013).

Todas as informações processadas pelo sistema são armazenadas no banco de dados de forma individual por nascente e que são exibidas no momento que o aplicativo é iniciado no celular, facilitando assim um análise e comparação mais rápida dos resultados.

6.4 – Implantação e Teste

Com o protótipo desenvolvido foi realizado vários testes de verificação do funcionamento correto das *interfaces*, botões e textos no *software*. Também foi verificado se os resultados das somas e classificação dos índices foram processados corretamente.

Figura 8 – Interface Principal do DRPN- 00



Fonte: Tela celular

Após verificar o funcionamento correto das funcionalidades do Software DRPN – 00 foi realizada a distribuição do mesmo no email da turma ano 2016 para os alunos do curso de especialização de meio ambiente do Instituto Federal de Minas Gerais – Campus São João Evangelista.

Depois que os alunos fizeram o teste do aplicativo foram questionados se gostaram de fazer uma avaliação de nascente daquela maneira. Responderam que acharam bastante moderno, uma vez que nunca haviam feito este procedimento usando o celular, pois o resultado sai na hora e é possível fazer uma observação dos principais parâmetros de maneira rápida e segura, sem ficar recorrendo a literatura.

Figura 9 – Interface de Avaliação da Nascente do IRPAN- 00

Identificação da Nascente

Nome

Latitude

Longitude

Avaliação dos Impactos Antrópicos

1 - Odor da água (cheiro)

- Odor forte
- Odor suportável
- Não ha odor

2 - Presença de Lixo ao redor da nascente

- Muito lixo
- Pouco lixo
- Nenhum lixo

3 - Materiais Flutuantes na água (lixo na água)

- Muito material
- Pouco material
- Nenhum material

4 - Presença de Óleos e Grachas

- Muito

Superfícies permeáveis em áreas com declividades moderadas a elevadas (6 a 12%) e cobertura vegetal densa ou moderadamente densa

Superfícies pouco permeáveis, com declividades elevadas (>12%) e cobertura vegetal ausente ou rarefeita

12 - Geologia do Aquífero (permeabilidade)

Rochas permeáveis: aquíferos com permeabilidade relativamente elevada derivada da associação de variáveis como proporção de vazios primários, estruturas geológicas e/ou cavidades cársticas

Rochas com permeabilidade relativamente intermediária derivada da associação entre as variáveis: proporção de vazios primários, estruturas geológicas e/ou cavidades cársticas em situação intermediária entre as categorias alta e baixa

Rochas com permeabilidade relativamente baixa derivada da associação entre a proporção de vazios primários, estruturas geológicas e/ou cavidades cársticas

AVALIAR NASCENTE **CANCELAR**

Fonte: Tela celular

Na etapa final dos testes foram realizadas experimentações diversas sobre as funcionalidade de exclusão, correção e atualização das avaliações de nascentes que o sistema disponibiliza.

Figura 10 – Interface de Exclusão e Correção das avaliações de Nascentes do DRPN- 00

Índice de Risco de Poluição das Águas da Nascente
 - Resulta da relação entre os impactos antrópicos sobre as nascentes e a vulnerabilidade natural à poluição dos sistemas aquíferos que mantêm as nascentes.

AVALIAR NOVA NASCENTE

Nascentes que foram Avaliadas

Nome: Teste Impactos Antrópicos: Muito Baixo
 Latitude: 33333
 Longitude: 33333

Selecionar:

Editar
 Excluir

SAIR

Identificação da Nascente

Nome: Teste4
 Latitude: 33333
 Longitude: 33333

Avaliação dos Impactos Antrópicos

1 - Odor da água (cheiro)
 Odor forte
 Odor suportável
 Não ha odor

2 - Presença de Lixo ao redor da nascente
 Muito lixo
 Pouco lixo
 Nenhum lixo

3 - Materiais Flutuantes na água (lixo na água)
 Muito material
 Pouco material
 Nenhum material

4 - Presença de Óleos e Grachas
 Muito

Fonte: Tela celular

7 – AVALIAÇÃO DAS NASCENTES DO IFMG – SÃO JOÃO EVANGELISTA

Com o auxílio do consultor ambiental e cursista da Especialização em Meio Ambiente do IFMG – São João Evangelista Sr. Álisson César Rodrigues Pereira, realizamos o trabalho de campo com o objetivo de avaliar as nascentes existentes na área da instituição de ensino.

Figura 11 – Área do Campus IFMG – São João Evangelista



Fonte: Google Earth Pro - 2017

O campus apresenta uma área de 301,3702 ha, compreendendo uma área de aproximadamente 133,9700 ha de vegetação pertencente ao domínio de Mata Atlântica. Foram identificadas treze nascentes, onde destas, 6 (seis) apresentam um pequeno fluxo de água superficial.

Figura 12 – Nascentes com Fluxo de Água no Campus



Fonte: Google Earth Pro - 2017

De acordo com Gomes (2004), o diagnóstico ambiental é uma ferramenta indispensável em um trabalho de pesquisa de campo, características relacionadas à vegetação, ao solo, ao ar e à água podem ser avaliadas “em tempo real” e segundo Bosquilia (2014), a preocupação com os recursos hídricos vem aumentando, isso se deve pelo fato de a água ser o bem mais importante na terra, o mesmo ressalta que, tem sido visto um maior reconhecimento de que são insuficientes os cuidados destinados aos recursos hídricos. Desta forma iniciou-se a diagnóstico do risco de poluição das águas das nascentes com o uso do DRPN – 00.

7.1 – Avaliação da Nascente Paineira

A primeira nascente visitada (figura 13) encontra-se próximo à área de cultivo de sorgo, e apresenta as seguintes coordenadas ($18^{\circ} 32'.845''$ S $42^{\circ}45'.028''$ W). Observou-se a presença de vegetação nativa ao entorno da mesma, embora a faixa de mata ciliar encontra-se entorno de 15 m.

Figura 13 - Foto Nascente Paineira



Fonte: PEREIRA (2016)

Na avaliação realizada pelo software DRPN-00, a nascente foi classificada com potencial “Médio Baixo” a poluição de suas águas. Também com relação aos impactos antrópicos a nascente foi classificada como “Baixo”. Já em relação a vulnerabilidade natural ela foi classificada como “Média”.

7.2 – Avaliação da Nascente Ypê

A segunda nascente avaliada (figura 14) encontra-se próxima a residência 03 e apresenta as seguintes coordenadas ($18^{\circ} 32' 921''\text{S } 42^{\circ} 45'. 121''\text{W}$), no entorno da mesma verificou-se a presença de planta invasora Capim Colonião (*Panicum maximum*), e também vegetação nativa. Na área a direita da nascente, localizada acima de uma estrada, existe uma área com declividade acentuada e pastagem degradada.

Figura 14 - Foto Nascente Ypê Amarelo



Fonte: PEREIRA (2016)

Na avaliação realizada pelo software DRPN-00, a nascente foi classificada com potencial “Médio Baixo” a poluição de suas águas. Também com relação aos impactos antrópicos a nascente foi classificada como “Baixo”. Já em relação a vulnerabilidade natural ela foi classificada como “Média”.

7.3 – Avaliação da Nascente Braúna

A terceira nascente avaliada (figura 15) está localizada nas proximidades do abatedouro. A área apresenta as seguintes coordenadas ($18^{\circ} 33' 092'' \text{ S } 42^{\circ} 45'.205'' \text{ W}$).

Notou-se a prevalência de capim napiê (*Pennisetum purpureum*) em torno da nascente, assim como escoamento de água pluvial com indícios de assoreamento na área da nascente. Pastagem de braquiária (*Brachiaria decumbens*) ao redor da nascente. Ausência de mata ciliar, com poucos remanescentes de mata, foram características da área identificadas.

Figura 15 - Foto Nascente Braúna



Fonte: PEREIRA (2016)

Na avaliação realizada pelo software DRPN-00, a nascente foi classificada com potencial “Médio Alto” a poluição de suas águas. Também com relação aos impactos antrópicos a nascente foi classificada como “Elevado”. Já em relação a vulnerabilidade natural ela foi classificada como “Média”.

7.4 – Avaliação da Nascente Peroba

A quarta nascente avaliada (figura 16) localiza-se próxima ao abatedouro e apresenta as seguintes coordenadas (18° 33'.068” S 42° 45'.240” W). Verificou-se a esquerda da nascente é possível observar vasta área de mata nativa. Porém, à direita, observa-se presença de faixa restrita de mata ciliar com aproximadamente 5 metros, além de pastagem com Braquiária (*Brachiaria decumbens*).

Figura 16 - Foto Nascente Peroba



Fonte: PEREIRA (2016)

Na avaliação realizada pelo software DRPN-00, a nascente foi classificada com potencial “Média” a poluição de suas águas. Também com relação aos impactos antrópicos a nascente foi classificada como “Média”. Já em relação a vulnerabilidade natural ela foi classificada como “Média”.

7.5 – Avaliação da Nascente Embaúba

A quinta nascente avaliada (figura 17) apresenta as seguintes coordenadas ($18^{\circ} 33' 316''$ S, $42^{\circ} 44' 988''$ W). Está situada nas proximidades do plantio de milho e sorgo. Verificou-se a presença de pastagem com braquiária (*Brachiaria decumbens*) e presença de taboa (*Typha domingensis*) em área de preservação permanente. Em torno da nascente a faixa de mata ciliar está em desacordo com o código florestal, constando de aproximadamente dez metros de vegetação. Ao lado esquerdo da nascente encontra-se área de cultivo de milho anual.

Figura 17 - Foto Nascente Embaúba



Fonte: PEREIRA (2016)

Na avaliação realizada pelo software DRPN-00, a nascente foi classificada com potencial “Médio Baixo” a poluição de suas águas. Também com relação aos impactos antrópicos a nascente foi classificada como “Baixo”. Já em relação a vulnerabilidade natural ela foi classificada como “Média”.

7.6 – Avaliação da Nascente Jacarandá

A sexta nascente avaliada (figura 18) apresenta as seguintes coordenadas ($18^{\circ} 33' 025''$ S $42^{\circ} 45' 820''$ W), localizada próxima à portaria do campus. A situação da nascente encontra-se ativa, porém foi necessária uma limpeza no olho d'água devido a presença de taboa (*Typha domingensis*) e de outras plantas invasoras ao redor do curso d'água. A área à esquerda da nascente se trata de uma área de topo de morro, a mesma encontra-se em estágio de regeneração. À direita da nascente encontra-se também uma área de topo de morro, que por vários anos foi realizado o cultivo de café e atualmente está sendo efetuado novo plantio de café.

Figura 18 - Foto Nascente Jacarandá



Fonte: PEREIRA (2016)

Na avaliação realizada pelo software DRPN-00, a nascente foi classificada com potencial “Média” a poluição de suas águas. Também com relação aos impactos antrópicos a nascente foi classificada como “Média”. Já em relação a vulnerabilidade natural ela foi classificada como “Média”.

7.7 – Comparação dos Resultados das Avaliações das Nascentes

Pelo fato de todas as nascentes se localizarem dentro da mesma micro bacia hidrogeológica, foi escolhido as mesmas variáveis para os parâmetros de vulnerabilidade natural. Já as variáveis dos parâmetros de impactos antrópicos foram escolhidas conforme o critério adotado pelo consultor no momento da avaliação.

Na Tabela 6 é possível verificar os resultados quanto a vulnerabilidade Natural onde todas as nascentes avaliadas ficaram com o resultado “Médio” o que demonstra que naturalmente a formação geológica de escudo cristalino bastante intemperizado, coberto com pacotes de sedimentos e Latossolo bastante permeável, apresenta uma facilidade de se contaminar com poluentes as águas subterrâneas do aquífero. Já em relação aos impactos antrópicos observa-se três nascente (Paineira, Ypê e Embaúba) encontram-se mais protegida

da poluição e que as nascente (Peroba e Jacarandá) devido a sua localização próximo a equipamentos urbanos possuem um médio potencial de serem poluídas e que a nascente Braúna é a que necessita de uma maior atenção em relação a poluição de suas águas.

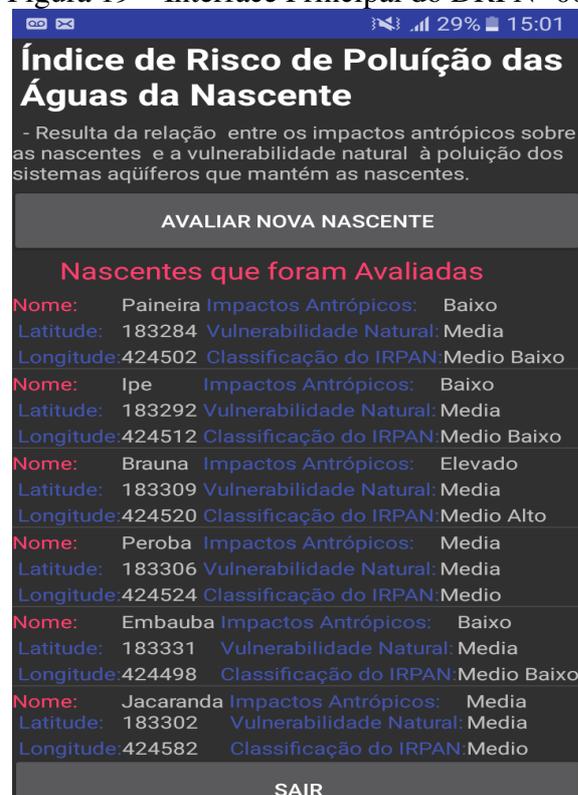
Da mesma forma estas informações podem ser analisadas conforme o DRPN-00 onde verifica-se que a nascente Braúna apresenta Médio Alto potencial de ter suas águas poluídas. Como pode ser observado na tabela 6.

Tabela 6 – Resumo das Avaliações das Nascentes do IFMG

Nome	Impactos Antrópicos	Vulner. Natural	DRPN
Paineira	Baixo	Médio	Médio Baixo
Ypê	Baixo	Médio	Médio Baixo
Braúna	Elevado	Médio	Médio Alto
Peroba	Medio	Médio	Médio
Embaúba	Baixo	Médio	Médio Baixo
Jacarandá	Medio	Médio	Médio

Na Figura 19 pode se verificar na interface principal do software o resultado das avaliações de todas as nascentes do Campus do IFMG-SJE

Figura 19 – Interface Principal do DRPN -00



Fonte: Tela celular

7.7.1 – Medidas de Recuperação de Nascentes Adotadas

Após o diagnóstico de avaliação das nascentes com o uso do Software DRPN, foi ministrado um mini-curso de proteção de nascentes no campus do IFMG - São João Evangelista, através de financiamento do Serviço Nacional de Aprendizagem Rural (SENAR), onde foi adotado medidas de preservação das nascentes Ypê e Braúnas por meio de aula pratica.

Como medidas de recuperação nascente Ypê. Limparam as margens e o canal principal da nascente até a bacia de acumulo da água de modo a facilitar o escoamento das águas (Figura 21). Depois instalaram canos de PVC de modo a auxiliar a exfiltração das águas das águas e protegeram a região do olho d'água com argila de modo a evitar o pisoteio ou até mesmo o assoreamento desta por sedimento provenientes dos barrancos trazidos pela enxurrada, devido a sua proximidade da estrada vicinal (Figura 20).

Figura 20 – Nascente Ypê Amarelo 2



Figura 21 – Nascente Ypê Amarelo 3



Na nascente Braúna realizaram a limpeza do canal principal, das margens e do olho d'água como forma de facilitar a exfiltração e o escoamento das águas. Também retiram todo o material que poluía a água (Fig. 22 e 23).

Figura 22 – Nascente Braúna 2



Figura 23 – Nascente Braúna 3



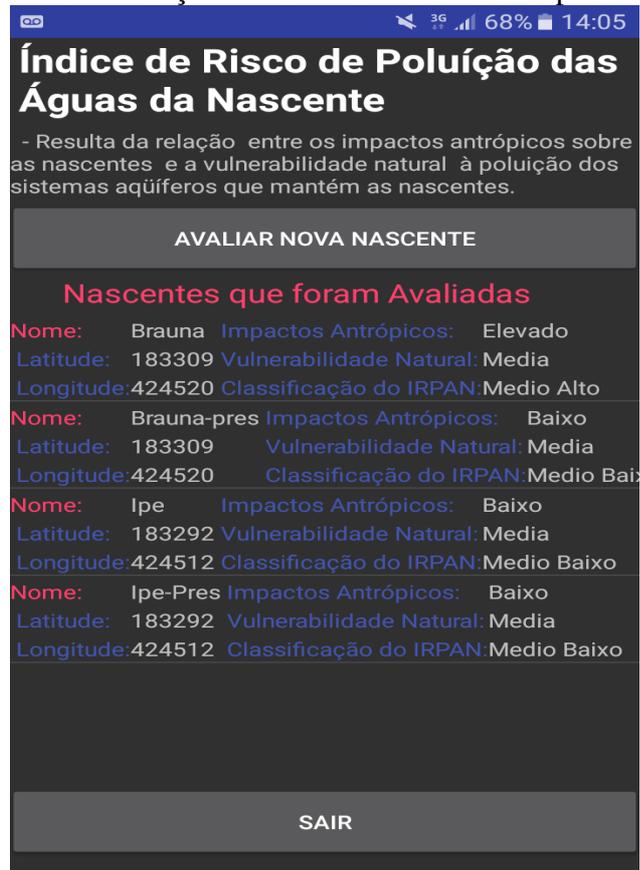
A partir da nova realidade ambiental reavaliou-se novamente as nascente Braúna e Ype com o aplicativo desenvolvido DRPN, onde verificou que as medidas de preservação adotadas foram eficientes uma vez que o índice da nascente Braúna que era “Médio Alto” foi melhorado para “Médio Baixo”. Já o índice da nascente Ype manteve a mesma classificação em “Médio Baixo”, uma vez que a mesma já apresentava um menor risco á poluição. Como pode ser observado na tabela 7.

Tabela 7 – Comparativo das Reavaliações das Nascentes Braúnas e Ype

Nome	Impactos Antrópicos	Vulner. Natural	DRPN
Braúna	Elevado	Médio	Médio Alto
Braúna-Pres	Baixo	Médio	Médio Baixo
Ypê	Baixo	Médio	Médio Baixo
Ypê-Pres	Baixo	Médio	Médio Baixo

Na Figura 24 pode se verificar na interface principal do software o resultado das reavaliações das nascentes Braúna e Ype do Campus do IFMG-SJE .

Figura 24 – Interface Principal do DRPN-00 após Reavaliação das nascentes Braúna e Ype



Fonte: Tela celular

Conclui-se que as medidas foram satisfatórias para melhorar a qualidade e quantidade água das duas nascentes. Porém ainda existe três nascentes que necessitam urgentemente de ações como estas para continuarem existindo, pois nos estudos preliminares haviam 13 (treze) nascente no Campus e só foram avaliadas 6(seis) que ainda apresentam um escoamento superficial

8 – CONSIDERAÇÕES

Os testes com o Software de Diagnóstico do Risco de Poluição de Nascentes demonstraram que ele é uma excelente ferramenta que visa a facilitar o trabalho dos consultores ambientais que atuam na área.

Também o Software demonstrou-se eficiente na avaliação das nascentes que existem no Campus do IFMG – São João Evangelista mesmo que estas se localizam dentro de uma propriedade rural com grande quantidade de vegetação nativa e com aplicação de práticas ecológicas de recuperação e preservação. No entanto, sua utilização é mais eficiente para avaliar nascentes que estejam sofrendo interferência da urbanização onde é demonstrado classificações do DRPN mais preocupantes.

Neste contexto, o desenvolvimento de software como esse criam variantes metodológicas aos modelos convencionais de análise ambiental, aumentando o interesse dos consultores e, conseqüentemente, facilitando práticas preservacionista e de trabalho.

É importante também que o consultor tenha em mente que o uso de software como recurso de trabalho deve ter um enfoque voltado para motivar novas formas de preservar a água. Acredita-se assim que a utilização de software dessa natureza exerçam valiosa contribuição na transposição de metodologias de trabalho antigas para as mais modernas.

Sabe-se que tudo evolui e que o DRPN - 00 está pronto mas nunca estará acabado, uma vez, que existem várias possibilidades de evolução a serem inclusas no mesmo como: novas funções, novas abordagens e até mesmo novas questões para avaliar o estágio de conservação das nascentes e facilitando assim a vida do consultor ambiental.

9 - REFERÊNCIAS BIBLIOGRÁFICAS

BEZERRA, Eduardo. **Princípios de análise e projeto de sistemas com UML**. Rio de Janeiro: Campus, 2002.

BOSQUILIA, R. W. D. **Geotecnologias aplicadas ao mapeamento de drenagens e nascentes**. Dissertação (Mestrado) Escola superior de Agricultura “Luiz Queiroz “ , 2014.

BRASIL – **Lei nº 6.938. 31 de agosto de 1981. Dispõe sobre a Política Nacional do Meio Ambiente, seus fins e mecanismos de formulação e aplicação, e dá outras providências**. Disponível em: <http://www.planalto.gov.br/ccivil_03/leis/16938.htm>. Acesso em: 18/10/2016.

DAVIS, S. N. **Hidrogeology**. New York: 1966.

FELIPPE, M. F. **Caracterização e tipologia de nascentes em unidades de conservação de Belo Horizonte com base em variáveis geomorfológicas, hidrológicas e ambientais**. Dissertação (mestrado) – Universidade Federal de Minas Gerais, Instituto de Geociências, 2009.

FELIPPE, M. F.; MAGALHÃES Jr. A. P. **Análise da variabilidade da vazão das nascentes no Parque das Mangabeiras (Belo Horizonte-MG) em relação aos seus condicionantes ambientais**. In Anais do XIII Simpósio Brasileiro de Geografia Física Aplicada, Viçosa- MG, 2009.

FELIPPE, M. F.; MAGALHÃES Jr. A. P. **Impactos ambientais macroscópicos e qualidade das águas em nascentes de parques municipais em Belo Horizonte-MG**. Geografias (UFMG), v. 8, n. 2. pp. 8 – 22. Jul-Dez, 2012.

FILIZZOLA, B.R. **PROPOSTA METODOLÓGICA PARA DESENVOLVIMENTO DO ÍNDICE DE RISCO À POLUIÇÃO DAS ÁGUAS DE NASCENTES**, Monografia Bacharelado Geografia - UFMG, 2013

GOMES, M. A. F. **A importância das águas superficiais e subterrâneas no diagnóstico ambiental**. In: HAMMES, V. S. **Ver-Percepção do Diagnóstico Ambiental**. São Paulo: Globo S.A., 2004. p. 30-33.

MAGALHÃES Jr. A. P. **Indicadores ambientais e recursos hídricos – realidade e perspectivas para o Brasil a partir da experiência francesa**. 4a ed. – Rio de Janeiro: Bertrand Brasil, 2012.

NOBRE, R. C. M.; NOBRE, M. M. M.; ROTUNNO, O. C. F.; MANSUR, W. J. **Mapeamento do Índice de Perigo ao Consumo de Águas Subterrâneas Vulneráveis à Contaminação**. Revista Brasileira de Recursos Hídricos, v. 13, n. 2. p.101 – 111. Abr- Jun, 2008.

PRETTO, Nelson de Luca (org.). **Globalização & Organização: mercado de trabalho, tecnologias de comunicação, educação a distancia e sociedade planetária**. Ijuí: Ed. Unijuí, 1999.

PRESSMAN, R. S. (1995). **Engenharia de Software**. Makron Books.

SOMMERVILLE, I. (2003). **Engenharia de software**. Addison Wesley.

TOMINAGA, L. K. **Análise e mapeamento de risco**. In: TOMINAGA, L. K.; SANTORO, J.; AMARAL, R.(Orgs.) **Desastres Naturais: conhecer para prevenir**. Instituto Geológico. São Paulo, 2009. p. 147-160. cap. 9.

VALENTE, O. F.; GOMES, M. A. **Conservação de nascentes: hidrologia e manejo de bacias hidrográficas de cabeceiras**. Viçosa: Aprenda Fácil, 2005.

VON SPERLING, M. **Introdução à qualidade das águas e ao tratamento de esgotos**. 3. ed. Belo Horizonte: UFMG/Departamento de Engenharia Sanitária, 2005. v.1, p. 47.

APÊNDICE A - CODIGO DAS CLASSE DO DRPN

CÓDIGOS DAS INTERFACES - XML

Classe main

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    android:orientation="vertical" >
    <TextView
        android:text="Índice de Risco de Poluição das Águas da Nascente"
        android:layout_width="366dp"
        android:layout_height="wrap_content"
        android:id="@+id/textView2"
        android:textColor="@android:color/background_light"
        android:textSize="24dp"
        android:textStyle="bold" />
    <TextView
        android:text="- Resulta da relação entre os impactos antrópicos
sobre as nascentes e a vulnerabilidade natural à poluição dos sistemas
aquiíferos que mantém as nascentes."
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:id="@+id/textView7" />
    <Button
        android:id="@+id/add"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Avaliar Nova Nascente"
        android:onClick="onClick" />
    <TextView
        android:text="Nascentes que foram Avaliadas"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="10dp"
        android:textSize="19dp"
        android:textColor="@android:color/background_light"
        android:id="@+id/textView" />
    <ListView
        android:id="@android:id/list"
        android:layout_width="fill_parent"
        android:layout_height="269dp"
        />
</LinearLayout>

```

Classe nascente

```

<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="@android:color/darker_gray">
    <ScrollView
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:background="@android:color/darker_gray">
        <LinearLayout
            android:orientation="vertical"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_marginLeft="20dp">
            <TextView
                android:text="Identificação da Nascente"
                android:layout_marginTop="10dp"
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:textSize="20dp"
                android:layout_marginLeft="15dp"
                android:textColor="@color/colorPrimaryDark"
                android:textStyle="bold"
                android:id="@+id/textView18" />
            <TextView
                android:text="Nome"
                android:layout_marginLeft="30dp"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:id="@+id/textView19"
                android:textSize="16dp"
                android:textColor="@color/colorPrimaryDark"
                android:layout_weight="1" />
            <EditText
                android:layout_height="wrap_content"
                android:inputType="textPersonName"
                android:text=""
                android:ems="10"
                android:textColor="@color/colorPrimary"
                android:id="@+id/editText"
                android:layout_width="200dp"
                android:layout_marginLeft="50dp"
                android:background="@android:color/background_light"/>
            <TextView
                android:text="Latitude"
                android:layout_marginLeft="30dp"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:id="@+id/textView20"
                android:textSize="16dp"
                android:textColor="@color/colorPrimaryDark"
                android:layout_weight="1" />
            <EditText
                android:layout_height="wrap_content"
                android:inputType="textPersonName"
                android:text=""
                android:ems="10"
                android:textColor="@color/colorPrimaryDark"
                android:id="@+id/editText1"

```

```

        android:layout_width="200dp"
        android:layout_marginLeft="50dp"
        android:background="@android:color/background_light"
        android:backgroundTint="@android:color/background_light"/>
<TextView
    android:text="Longitude"
    android:layout_marginLeft="30dp"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:id="@+id/textView21"
    android:textSize="16dp"
    android:textColor="@color/colorPrimaryDark"
    android:layout_weight="1" />
<EditText
    android:layout_height="wrap_content"
    android:inputType="textPersonName"
    android:text=""
    android:ems="10"
    android:textColor="@color/colorPrimaryDark"
    android:id="@+id/editText2"
    android:layout_width="200dp"
    android:layout_marginLeft="50dp"
    android:background="@android:color/background_light"/>
<TextView
    android:text="Avaliação dos Impactos Antrópicos"
    android:layout_marginTop="10dp"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:textSize="20dp"
    android:layout_marginLeft="15dp"
    android:textColor="@color/colorPrimaryDark"
    android:textStyle="bold"
    android:id="@+id/textView4" />
<RadioGroup
    android:id = "@+id/valor1"
    android:layout_width="match_parent"
    android:layout_height="match_parent" >
    <TextView
        android:text="1 - Odor da água (cheiro)"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:textColor="@color/colorPrimaryDark"
        android:textSize="16dp"
        android:layout_marginTop="10dp"
        android:id="@+id/textView5" />
    <RadioButton
        android:text="Odor forte"
        android:layout_marginLeft="30dp"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/var1op1"
        android:textSize="16dp"
        android:textColor="@color/colorPrimaryDark"
        android:layout_weight="1" />
    <RadioButton
        android:text="Odor suportável"
        android:layout_marginLeft="30dp"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:textSize="16dp"
        android:id="@+id/var1op2"

```

```

        android:textColor="@color/colorPrimaryDark"
        android:layout_weight="1" />
<RadioButton
    android:text="Não ha odor"
    android:layout_marginLeft="30dp"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:id="@+id/var1op3"
    android:textSize="16dp"
    android:textColor="@color/colorPrimaryDark"
    android:layout_weight="1" />
</RadioGroup>
<RadioGroup
    android:id = "@+id/valor2"
    android:layout_width="match_parent"
    android:layout_height="match_parent" >
    <TextView
        android:text="2 - Presença de Lixo ao redor da nascente"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:textColor="@color/colorPrimaryDark"
        android:textSize="16dp"
        android:id="@+id/textView6" />
    <RadioButton
        android:text="Muito lixo"
        android:layout_marginLeft="30dp"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:id="@+id/var2op1"
        android:textColor="@color/colorPrimaryDark"
        android:textSize="16dp"
        android:layout_weight="1" />
    <RadioButton
        android:text="Pouco lixo"
        android:layout_marginLeft="30dp"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/var2op2"
        android:textColor="@color/colorPrimaryDark"
        android:textSize="16dp"
        android:layout_weight="1" />
    <RadioButton
        android:text="Nenhum lixo"
        android:layout_marginLeft="30dp"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/var2op3"
        android:textColor="@color/colorPrimaryDark"
        android:textSize="16dp"
        android:layout_weight="1" />
</RadioGroup>
<RadioGroup
    android:id = "@+id/valor3"
    android:layout_width="match_parent"
    android:layout_height="match_parent">
    <TextView
        android:text="3 - Materiais Flutuantes na água (lixo na água)"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:textColor="@color/colorPrimaryDark"
        android:textSize="16dp"

```

```

        android:id="@+id/textView7" />
<RadioButton
    android:text="Muito material"
    android:layout_marginLeft="30dp"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:id="@+id/var3op1"
    android:textColor="@color/colorPrimaryDark"
    android:textSize="16dp"
    android:layout_weight="1" />
<RadioButton
    android:text="Pouco material"
    android:layout_marginLeft="30dp"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:id="@+id/var3op2"
    android:textColor="@color/colorPrimaryDark"
    android:textSize="16dp"
    android:layout_weight="1" />
<RadioButton
    android:text="Nenhum material"
    android:layout_marginLeft="30dp"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:id="@+id/var3op3"
    android:textColor="@color/colorPrimaryDark"
    android:textSize="16dp"
    android:layout_weight="1" />
</RadioGroup>
<RadioGroup
    android:id = "@+id/valor4"
    android:layout_width="match_parent"
    android:layout_height="match_parent">
<TextView
    android:text="4 - Presença de Óleos e Grachas"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:textColor="@color/colorPrimaryDark"
    android:textSize="16dp"
    android:id="@+id/textView8" />
<RadioButton
    android:text="Muito "
    android:layout_marginLeft="30dp"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:id="@+id/var4op1"
    android:textColor="@color/colorPrimaryDark"
    android:textSize="16dp"
    android:layout_weight="1" />
<RadioButton
    android:text="Pouco "
    android:layout_marginLeft="30dp"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:id="@+id/var4op2"
    android:textColor="@color/colorPrimaryDark"
    android:textSize="16dp"
    android:layout_weight="1" />
<RadioButton
    android:text="Nenhum "
    android:layout_marginLeft="30dp"

```

```

        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/var4op3"
        android:textColor="@color/colorPrimaryDark"
        android:textSize="16dp"
        android:layout_weight="1" />
</RadioGroup>
<RadioGroup
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:id="@+id/valor5" >
    <TextView
        android:text="5 - Lançamento de Esgoto na nascente"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:id="@+id/textView12"
        android:textColor="@color/colorPrimaryDark"
        android:textSize="16dp"
        android:layout_weight="1" />
    <RadioButton
        android:text="Visível"
        android:layout_marginLeft="30dp"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/var5op1"
        android:textColor="@color/colorPrimaryDark"
        android:textSize="16dp"
        android:layout_weight="1" />
    <RadioButton
        android:text="Provável"
        android:layout_marginLeft="30dp"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/var5op2"
        android:textColor="@color/colorPrimaryDark"
        android:textSize="16dp"
        android:layout_weight="1" />
    <RadioButton
        android:text="Inexistente"
        android:layout_marginLeft="30dp"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/var5op3"
        android:textColor="@color/colorPrimaryDark"
        android:textSize="16dp"
        android:layout_weight="1" />
</RadioGroup>
<RadioGroup
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:id="@+id/valor6" >
    <TextView
        android:text="6 - Vegetação no entorno da nascente"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/textView11"
        android:textColor="@color/colorPrimaryDark"
        android:textSize="16dp"
        android:layout_weight="1" />

    <RadioButton

```

```

        android:text="Ausente"
        android:layout_marginLeft="30dp"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/var6op1"
        android:textColor="@color/colorPrimaryDark"
        android:textSize="16dp"
        android:layout_weight="1" />
    <RadioButton
    android:text="Secundária (alterada com espécies invasoras)"
    android:layout_width="wrap_content"
    android:layout_marginLeft="30dp"
    android:layout_height="wrap_content"
    android:id="@+id/var6op2"
    android:textColor="@color/colorPrimaryDark"
    android:textSize="16dp"
    android:layout_weight="1" />
    <RadioButton
    android:text="Primária (nativa não alterada)"
    android:layout_width="wrap_content"
    android:layout_marginLeft="30dp"
    android:layout_height="wrap_content"
    android:id="@+id/var6op3"
    android:textColor="@color/colorPrimaryDark"
    android:textSize="16dp"
    android:layout_weight="1" />
</RadioGroup>
<RadioGroup
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:id="@+id/valor7" >
    <TextView
        android:text="7 - Uso das águas da nascente"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/textView10"
        android:textColor="@color/colorPrimaryDark"
        android:textSize="16dp"
        android:layout_weight="1" />
    <RadioButton
        android:text="Dessedentação de animais"
        android:layout_width="wrap_content"
        android:layout_marginLeft="30dp"
        android:layout_height="wrap_content"
        android:id="@+id/var7op1"
        android:textColor="@color/colorPrimaryDark"
        android:textSize="16dp"
        android:layout_weight="1" />
    <RadioButton
        android:text="Captação humana"
        android:layout_marginLeft="30dp"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/var7op2"
        android:textColor="@color/colorPrimaryDark"
        android:textSize="16dp"
        android:layout_weight="1" />
    <RadioButton
        android:text="Nenhum uso"
        android:layout_marginLeft="30dp"
        android:layout_width="wrap_content"

```

```

        android:layout_height="wrap_content"
        android:id="@+id/var7op3"
        android:textColor="@color/colorPrimaryDark"
        android:textSize="16dp"
        android:layout_weight="1" />
</RadioGroup>
<RadioGroup
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:id="@+id/valor8" >
    <TextView
        android:text="8 - Acesso ao local da nascente"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/textView9"
        android:textColor="@color/colorPrimaryDark"
        android:textSize="16dp"
        android:layout_weight="1" />
    <RadioButton
        android:text="Fácil acesso"
        android:layout_marginLeft="30dp"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/var8op1"
        android:textColor="@color/colorPrimaryDark"
        android:textSize="16dp"
        android:layout_weight="1" />
    <RadioButton
        android:text="Difícil acesso"
        android:layout_marginLeft="30dp"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/var8op2"
        android:textColor="@color/colorPrimaryDark"
        android:textSize="16dp"
        android:layout_weight="1" />
    <RadioButton
        android:text="Nenhum acesso"
        android:layout_marginLeft="30dp"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/var8op3"
        android:textColor="@color/colorPrimaryDark"
        android:textSize="16dp"
        android:layout_weight="1" />
</RadioGroup>
<RadioGroup
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:id="@+id/valor9" >
    <TextView
        android:text="9 - Distância dos equipamentos urbanos"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/textView3"
        android:textColor="@color/colorPrimaryDark"
        android:textSize="16dp"
        android:layout_weight="1" />
    <RadioButton
        android:text="Menos de 50 metros"
        android:layout_marginLeft="30dp"

```

```

        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/var9op1"
        android:textColor="@color/colorPrimaryDark"
        android:textSize="16dp"
        android:layout_weight="1" />
    <RadioButton
        android:text="Entre 50 e 100 metros"
        android:layout_marginLeft="30dp"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/var9op2"
        android:textColor="@color/colorPrimaryDark"
        android:textSize="16dp"
        android:layout_weight="1" />
    <RadioButton
        android:text="Maior que 100 metros"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/var9op3"
        android:textColor="@color/colorPrimaryDark"
        android:textSize="16dp"
        android:layout_marginLeft="30dp"
        android:layout_weight="1" />
</RadioGroup>
<TextView
    android:text="Vulnerabilidade Natural à Poluição"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginTop="10dp"
    android:layout_marginLeft="15dp"
    android:textColor="@color/colorPrimaryDark"
    android:textSize="20dp"
    android:textStyle="bold"
    android:id="@+id/textView13" />
<RadioGroup
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:id="@+id/valor10">
    <TextView
        android:text=" 10 - Estimativa da Profundidade do Nível
Freático (metros)"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/textView14"
        android:textColor="@color/colorPrimaryDark"
        android:textSize="16dp"
        android:layout_marginTop="10dp"
        android:layout_weight="1" />
    <RadioButton
        android:text="Áte 10 metros"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/var10op1"
        android:textColor="@color/colorPrimaryDark"
        android:textSize="16dp"
        android:layout_marginLeft="30dp"
        android:layout_weight="1" />
    <RadioButton
        android:text=" 10 à 30 metros"
        android:layout_width="wrap_content"

```

```

        android:layout_height="wrap_content"
        android:id="@+id/var10op2"
        android:textColor="@color/colorPrimaryDark"
        android:textSize="16dp"
        android:layout_marginLeft="30dp"
        android:layout_weight="1" />
<RadioButton
    android:text="Maior de 30 metros"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:id="@+id/var10op3"
    android:textColor="@color/colorPrimaryDark"
    android:textSize="16dp"
    android:layout_marginLeft="30dp"
    android:layout_weight="1" />
<TextView
    android:text="OBS: É mensurada a partir da profundidade
média do nível da água em poços escavados no aquífero"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:id="@+id/textView15"
    android:textColor="@color/colorPrimaryDark"
    android:textSize="14dp"
    android:layout_marginTop="5dp"
    android:layout_weight="1" />
</RadioGroup>
<RadioGroup
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:id="@+id/valor11">
    <TextView
        android:text=" 11 - Capacidade de Infiltração e Recarga"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/textView17"
        android:textColor="@color/colorPrimaryDark"
        android:textSize="16dp"
        android:layout_marginTop="10dp"
        android:layout_weight="1" />
    <RadioButton
        android:text="Superfícies permeáveis, com topografia
plana ou suavizada (0 a 6%) e cobertura vegetal cuja densidade seja
favorável à infiltração "
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:id="@+id/var11op1"
        android:textColor="@color/colorPrimaryDark"
        android:textSize="16dp"
        android:layout_marginTop="10dp"
        android:layout_marginLeft="30dp"
        android:layout_weight="1"
        android:layout_marginRight="10dp"
        android:layout_marginEnd="10dp" />
    <RadioButton
        android:text=" Superfícies permeáveis em áreas com
declividades moderadas a elevadas (6 a 12%) e cobertura vegetal densa ou
moderadamente densa "
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/var11op2"
        android:textColor="@color/colorPrimaryDark"

```

```

        android:textSize="16dp"
        android:layout_marginTop="10dp"
        android:layout_marginLeft="30dp"
        android:layout_marginEnd="10dp"
        android:layout_marginRight="10dp"
        android:layout_weight="1" />
<RadioButton
    android:text="Superfícies pouco permeáveis, com
declividades elevadas (>12%) e cobertura vegetal ausente ou rarefeita"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:id="@+id/var11op3"
    android:layout_marginTop="10dp"
    android:textColor="@color/colorPrimaryDark"
    android:textSize="16dp"
    android:layout_marginLeft="30dp"
    android:layout_marginRight="10dp"
    android:layout_marginEnd="10dp"
    android:layout_weight="1" />
</RadioGroup>
<RadioGroup
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:id="@+id/valor12">
    <TextView
        android:text=" 12 - Geologia do Aquífero (permeabilidade)"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/textView16"
        android:textColor="@color/colorPrimaryDark"
        android:textSize="16dp"
        android:layout_marginTop="10dp"
        android:layout_weight="1" />
    <RadioButton
        android:text="Rochas permeáveis: aquíferos com
permeabilidade relativamente elevada derivada da associação de variáveis
como proporção de vazios primários, estruturas geológicas e/ou cavidades
cársticas "
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/var12op1"
        android:textColor="@color/colorPrimaryDark"
        android:textSize="16dp"
        android:layout_marginTop="10dp"
        android:layout_marginLeft="30dp"
        android:layout_weight="1"
        android:layout_marginRight="10dp"
        android:layout_marginEnd="10dp"/>
    <RadioButton
        android:text="Rochas com permeabilidade relativamente
intermediária derivada da associação entre as variáveis: proporção de
vazios primários, estruturas geológicas e/ou cavidades cársticas em
situação intermediária entre as categorias alta e baixa "
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/var12op2"
        android:layout_marginTop="10dp"
        android:textColor="@color/colorPrimaryDark"
        android:textSize="16dp"
        android:layout_marginLeft="30dp"
        android:layout_weight="1"

```

```

        android:layout_marginRight="10dp"
        android:layout_marginEnd="10dp"/>
<RadioButton
    android:text="Rochas com permeabilidade relativamente
baixa derivada da associação entre a proporção de vazios primários,
estruturas geológicas e/ou cavidades cársticas"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:id="@+id/var12op3"
    android:textColor="@color/colorPrimaryDark"
    android:textSize="16dp"
    android:gravity="fill_horizontal"
    android:layout_marginLeft="30dp"
    android:layout_marginTop="10dp"
    android:layout_weight="1"
    android:layout_marginRight="10dp"
    android:layout_marginEnd="10dp"/>
</RadioGroup>
<LinearLayout
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:layout_below="@+id/edtTelefone"
    android:gravity="center_vertical">
    <Button
        android:text="Avaliar Nascente"
        android:layout_width="0dip"
        android:layout_weight="1.0"
        android:layout_height="wrap_content"
        android:id="@+id/btnConfirmar"
        android:layout_below="@+id/edtTelefone"
        android:onClick="btnConfirmar_click"
        android:layout_alignParentLeft="true">
    </Button>

    <Button
        android:text="Cancelar"
        android:layout_height="wrap_content"
        android:layout_width="0dip"
        android:layout_weight="1.0"
        android:id="@+id/btnCancelar"
        android:layout_below="@+id/edtTelefone"
        android:onClick="btnCancelar_click"
        android:layout_alignParentRight="true">
    </Button>
</LinearLayout>
</LinearLayout>
</ScrollView>
</RelativeLayout>

```

Classe nascente row

```

<?xml version="1.0" encoding="utf-8" ?>
<LinearLayout
    android:id="@+id/LinearLayout01"
    android:layout_width="fill_parent"
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_height="wrap_content"
    android:orientation="vertical">
    <TableLayout
        android:id="@+id/TableLayout"
        android:layout_height="wrap_content"
        android:layout_width="fill_parent">
        <TableRow
            android:layout_width="match_parent"
            android:layout_height="match_parent" >
            <TextView
                android:text="Nome: "
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:layout_column="0"
                android:id="@+id/textNome" />
            <TextView
                android:text=""
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:id="@+id/txtValorNome" />
            <TextView
                android:text="Impactos Antrópicos: "
                android:layout_height="wrap_content"
                android:id="@+id/textImpAntrop"
                android:layout_column="1"
                android:layout_width="match_parent" />
            <TextView
                android:text=""
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:id="@+id/txtValorImpacAntrop" />
        </TableRow>
        <TableRow
            android:layout_width="match_parent"
            android:layout_height="match_parent" >
            <TextView
                android:text="Latitude: "
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:layout_column="0"
                android:id="@+id/textLatitude" />
            <TextView
                android:text=""
                android:layout_height="wrap_content"
                android:layout_width="fill_parent"
                android:id="@+id/txtValorLatitude"/>
            <TextView
                android:text="Vulnerabilidade Natural: "
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:layout_column="1"
                android:id="@+id/textVulnerabilidadeNatural" />
            <TextView
                android:text=""

```

```

        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/txtValorVulnNatural"
        android:layout_below="@+id/textLongitude">
    </TextView>
</TableRow>
<TableRow
    android:layout_width="match_parent"
    android:layout_height="match_parent" >
    <TextView
        android:layout_height="wrap_content"
        android:layout_width="fill_parent"
        android:id="@+id/textLongitude"
        android:text="Longitude: "
        android:layout_column="0"
        android:layout_below="@+id/txtValorLongitude">
    </TextView>
    <TextView
        android:text=""
        android:layout_height="wrap_content"
        android:layout_width="fill_parent"
        android:id="@+id/txtValorLongitude">
    </TextView>
    <TextView
        android:text="Classificação do IRPAN:"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_column="1"
        android:id="@+id/textClassificacao" />
    <TextView
        android:text=""
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/txtValorClassificacao" />
    </TableRow>
</TableLayout>
</LinearLayout>

```

Classe arrays

```

<?xml version="1.0" encoding="utf-8"?>
<resources>
    <string-array name="menu">
        <item>Editar</item>
        <item>Excluir</item>
    </string-array>
</resources>

```

CODIGOS DAS CLASSES DO JAVA

Classe main

```

ackage com.example.jean.irpan_00;

/**
 * Created by JEAN on 22/06/2017.
 */
import android.os.Bundle;
import java.util.List;
import android.app.Activity;
import android.app.ListActivity;
import android.content.Intent;
import android.view.ContextMenu;
import android.view.Menu;
import android.view.MenuItem;
import android.view.View;
import android.view.ContextMenu.ContextMenuInfo;
import android.widget.AdapterView;
import android.widget.AdapterView;
import android.widget.Toast;
public class main extends ListActivity{
    private static final int INCLUIR = 0;
    private static final int ALTERAR = 1;
    private NascenteDAO lNascenteDAO; //instância responsável pela
    persistência dos dados
    List<Nascentes> lstNascentes; //lista de contatos cadastrados no BD
    NascenteAdapter adapter; //Adapter responsável por apresentar os
    contatos na tela
    boolean blnShort = false;
    int Posicao = 0; //determinar a posição do contato dentro da lista
    lstNascentes
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.main);
        lNascenteDAO = new NascenteDAO(this);
        lNascenteDAO.open();
        lstNascentes = lNascenteDAO.Consultar();
        adapter = new NascenteAdapter(this, lstNascentes);
        setListAdapter(adapter);
        registerForContextMenu(getListView());
    }
    // Este evento será chamado pelo atributo onClick
    // que está definido no botão criado no arquivo main.xml
    public void onClick(View view) {
        switch (view.getId()) {
            case R.id.add:
                InserirNascente();
                break;
        }
    }
    //Rotina executada quando finalizar a Activity ContatoUI
    @Override
    protected void onActivityResult(int requestCode, int resultCode, Intent
    data) {
        Nascentes lNascente = null;
        try {
            super.onActivityResult(requestCode, resultCode, data);
            if (resultCode == Activity.RESULT_OK)

```

```

        {
            //obtem dados inseridos/alterados na Activity ContatoUI
            lNascente = (Nascentes)
data.getExtras().getSerializable("nascente");
            //o valor do requestCode foi definido na função
startActivityForResult
            if (requestCode == INCLUIR)
            {
                //verifica se digitou algo no nome do contato
                if (!lNascente.getNome().equals(""))
                {
                    //necessário abrir novamente o BD pois ele foi
                    fechado no método onPause()
                    lNascenteDAO.open();
                    //insere o contato no Banco de Dados SQLite
                    lNascenteDAO.Inserir(lNascente);
                    //insere o contato na lista de contatos em memória
                    lstNascentes.add(lNascente);
                }
            } else if (requestCode == ALTERAR) {
                lNascenteDAO.open();
                //atualiza o contato no Banco de Dados SQLite
                lNascenteDAO.Alterar(lNascente);
                //atualiza o contato na lista de contatos em memória
                lstNascentes.set(Posicao, lNascente);
            }

            //método responsável pela atualiza da lista de dados na tela
            adapter.notifyDataSetChanged();
        }
    }
    catch (Exception e) {
        trace("Erro : " + e.getMessage());
    }
}

private void InserirNascente() {
    try
    {
        //a variável "tipo" tem a função de definir o comportamento da Activity
        //ContatoUI, agora a variável tipo está definida com o valor "0" para
        //informar que será uma inclusão da nascente
        Intent it = new Intent(this, NascenteUI.class);
        it.putExtra("tipo", INCLUIR);
        startActivityForResult(it, INCLUIR); //chama a tela e inclusão
    }
    catch (Exception e) {
        trace("Erro : " + e.getMessage());
    }
}

@Override
protected void onResume() {
    //quando a Activity main receber o foco novamente abre-se novamente
    a conexão
    lNascenteDAO.open();
    super.onResume();
}

@Override
protected void onPause() {
    //toda vez que o programa peder o foco fecha-se a conexão com o BD
    lNascenteDAO.close();
}

```

```

        super.onPause();
    }

    public void toast (String msg)
    {
        Toast.makeText (getApplicationContext(), msg,
        Toast.LENGTH_SHORT).show ();
    }
    private void trace (String msg)
    {
        toast (msg);
    }

    @Override
    public void onCreateContextMenu(ContextMenu menu, View v,
    ContextMenuInfo menuInfo) {
        try
        {
            //Criação do popup menu com as opções que termos sobre
            //nossas nascentes
            AdapterView.AdapterContextMenuInfo info =
            (AdapterView.AdapterContextMenuInfo)menuInfo;
            if (!blnShort)
            {
                Posicao = info.position;
            }
            blnShort = false;
            menu.setHeaderTitle("Selecione:");
            //a origem dos dados do menu está definido no arquivo arrays.xml
            String[] menuItems =
            getResources().getStringArray (R.array.menu);
            for (int i = 0; i<menuItems.length; i++) {
                menu.add(Menu.NONE, i, i, menuItems[i]);
            }
        } catch (Exception e) {
            trace("Erro : " + e.getMessage());
        }
    }
    //Este método é disparado quando o usuário clicar em um item do
    ContextMenu
    @Override
    public boolean onContextItemSelected(MenuItem item) {
        Nascentes lNascente = null;
        try
        {
            int menuItemIndex = item.getItemId();
            //Carregar a instância POJO com a posição selecionada na tela
            lNascente = (Nascentes) getListAdapter().getItem(Posicao);
            if (menuItemIndex == 0){
                //Carregar a Activity NascenteUI com o registro selecionado na tela
                Intent it = new Intent(this, NascenteUI.class);
                it.putExtra("tipo", ALTERAR);
                it.putExtra("nascente", lNascente);
                startActivityForResult(it, ALTERAR); //chama a tela de
                alteração
            } else if (menuItemIndex == 1){
                //Excluir do Banco de Dados e da tela o registro selecionado

                lNascenteDAO.Excluir(lNascente);
                lstNascentes.remove(lNascente);
                adapter.notifyDataSetChanged(); //atualiza a tela
            }
        }
    }

```

```
    }  
    } catch (Exception e) {  
        trace("Erro : " + e.getMessage());  
    }  
    return true;  
}  
@Override  
protected void onItemClick(ListView l, View v, int position, long  
id) {  
    super.onItemClick(l, v, position, id);  
    //por padrão o ContextMenu, só é executado através de LongClick, mas  
    //nesse caso toda vez que executar um ShortClick, abriremos o menu  
    //e também guardaremos qual a posição do itm selecionado  
    Posicao = position;  
    blnShort = true;  
    this.openContextMenu(l);  
}  
}
```

Classe BaseDAO – banco de dados

```

package com.example.jean.irpan_00;
/**
 * Created by JEAN on 22/06/2017.
 */
import android.content.Context;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteOpenHelper;
//Classe responsável pela criação do Banco de Dados e tabelas
public class BaseDAO extends SQLiteOpenHelper {
    public static final String TBL_NASCENTE = "nascente";
    public static final String NASCENTE_ID = "_id";
    public static final String NASCENTE_NOME = "nome";
    public static final String NASCENTE_LATITUDE = "latitude";
    public static final String NASCENTE_LONGITUDE = "longitude";
    public static final String NASCENTE_VALOR1 = "valor1";
    public static final String NASCENTE_VALOR2 = "valor2";
    public static final String NASCENTE_VALOR3 = "valor3";
    public static final String NASCENTE_VALOR4 = "valor4";
    public static final String NASCENTE_VALOR5 = "valor5";
    public static final String NASCENTE_VALOR6 = "valor6";
    public static final String NASCENTE_VALOR7 = "valor7";
    public static final String NASCENTE_VALOR8 = "valor8";
    public static final String NASCENTE_VALOR9 = "valor9";
    public static final String NASCENTE_SOMA1 = "soma1";
    public static final String NASCENTE_VALOR10 = "valor10";
    public static final String NASCENTE_VALOR11 = "valor11";
    public static final String NASCENTE_VALOR12 = "valor12";
    public static final String NASCENTE_SOMA2 = "soma2";
    public static final String NASCENTE_RESULTADOGLOBAL =
"resultadoGlobal";
    private static final String DATABASE_NAME = "nascente.db";
    private static final int DATABASE_VERSION = 1;
    //Estrutura da tabela Nascente (sql statement)
    private static final String CREATE_NASCENTE = "create table " +
        TBL_NASCENTE + " ( " + NASCENTE_ID + " integer primary key
autoincrement, " +
        NASCENTE_NOME + " text not null, " +
        NASCENTE_LATITUDE + " text not null, " +
        NASCENTE_LONGITUDE + " text not null, " +
        NASCENTE_VALOR1 + " text not null, " +
        NASCENTE_VALOR2 + " text not null, " +
        NASCENTE_VALOR3 + " text not null, " +
        NASCENTE_VALOR4 + " text not null, " +
        NASCENTE_VALOR5 + " text not null, " +
        NASCENTE_VALOR6 + " text not null, " +
        NASCENTE_VALOR7 + " text not null, " +
        NASCENTE_VALOR8 + " text not null, " +
        NASCENTE_VALOR9 + " text not null, " +
        NASCENTE_SOMA1 + " text not null, " +
        NASCENTE_VALOR10 + " text not null, " +
        NASCENTE_VALOR11 + " text not null, " +
        NASCENTE_VALOR12 + " text not null, " +
        NASCENTE_SOMA2 + " text not null, " +
        NASCENTE_RESULTADOGLOBAL + " text not null);";
    public BaseDAO(Context context) {
        super(context, DATABASE_NAME, null, DATABASE_VERSION);
    }

    @Override

```

```
public void onCreate(SQLiteDatabase database) {  
    //Criação da tabela  
    database.execSQL(CREATE_NASCENTE);  
}  
  
@Override  
public void onUpgrade(SQLiteDatabase db, int oldVersion, int  
newVersion) {  
    //Caso seja necessário mudar a estrutura da tabela  
    //deverá primeiro excluir a tabela e depois recriá-la  
    db.execSQL("DROP TABLE IF EXISTS " + TBL_NASCENTE);  
    onCreate(db);  
}  
}
```

Classe Nascente Adapter

```

package com.example.jean.irpan_00;
/**
 * Created by JEAN on 22/06/2017.
 */
import java.util.List;
import android.content.Context;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.BaseAdapter;
import android.widget.TextView;
import android.widget.Toast;
public class NascenteAdapter extends BaseAdapter {
    private Context context;
    private List<Nascentes> lstNascente;
    private LayoutInflater inflater;
    public NascenteAdapter(Context context, List<Nascentes> listNascente) {
        this.context = context;
        this.lstNascente = listNascente;
        inflater = (LayoutInflater)
context.getSystemService(Context.LAYOUT_INFLATER_SERVICE);
    }
    //Atualiza ListView de acordo com o lstNascente
    @Override
    public void notifyDataSetChanged() {
        try{
            super.notifyDataSetChanged();
        }catch (Exception e) {
            trace("Erro : " + e.getMessage());
        }
    }
    public int getCount() {
        return lstNascente.size();
    }
    //Remover item da lista
    public void remove(final Nascentes item) {
        this.lstNascente.remove(item);
    }
    //Adicionar item na lista
    public void add(final Nascentes item) {
        this.lstNascente.add(item);
    }
    public Object getItem(int position) {
        return lstNascente.get(position);
    }
    public long getItemId(int position) {
        return position;
    }
    public View getView(int position, View convertView, ViewGroup
viewGroup) {
        try
        {
            Nascentes nascente = lstNascente.get(position);
            //O ViewHolder irá guardar a instâncias dos objetos do estado_row
            ViewHolder holder;
            //Quando o objeto convertView não for nulo nós não precisaremos inflar
            //os objetos do XML, ele será nulo quando for a primeira vez que for
            carregado
            if (convertView == null) {

```

```

        convertView = inflater.inflate(R.layout.nascente_row, null);
        //Cria o ViewHolder e guarda a instância dos objetos (layout.nascente_row,
        null)
        holder = new ViewHolder();
        holder.tvNome = (TextView) convertView.findViewById(
R.id.txtValorNome);
        holder.tvLatitude = (TextView)
convertView.findViewById(R.id.txtValorLatitude);
        holder.tvLongitude = (TextView)
convertView.findViewById(R.id.txtValorLongitude);
        holder.tvImpactosAntropicos = (TextView)
convertView.findViewById(R.id.txtValorImpacAntrop);
        holder.tvVunelrabilidadeNatural = (TextView)
convertView.findViewById(R.id.txtValorVulnNatural);
        holder.tvClassificacao = (TextView)
convertView.findViewById(R.id.txtValorClassificacao);
        convertView.setTag(holder);
    } else {
        //pega o ViewHolder para ter um acesso rápido aos objetos do XML
        //ele sempre passará por aqui quando, por exemplo, for efetuado uma
        rolagem na tela
        holder = (ViewHolder) convertView.getTag();
    }
    holder.tvNome.setText(nascente.getNome());
    holder.tvLatitude.setText(nascente.getLatitude());
    holder.tvLongitude.setText(nascente.getLongitude());
    holder.tvImpactosAntropicos.setText(nascente.getSoma1());
    holder.tvVunelrabilidadeNatural.setText(nascente.getSoma2());
    holder.tvClassificacao.setText(nascente.getResultadoGlobal());
    return convertView;
} catch (Exception e) {
    trace("Erro : " + e.getMessage());
}
return convertView;
}
}
public void toast (String msg)
{
    Toast.makeText (context, msg, Toast.LENGTH_SHORT).show ();
}
private void trace (String msg)
{
    toast (msg);
}
//Criada esta classe estática para guardar a referência dos objetos
abaixo
static class ViewHolder {
    public TextView tvNome;
    public TextView tvLatitude;
    public TextView tvLongitude;
    public TextView tvImpactosAntropicos;
    public TextView tvVunelrabilidadeNatural;
    public TextView tvClassificacao;
}
}

```

Classe NascenteDAO

```

package com.example.jean.irpan_00;
/**
 * Created by JEAN on 22/06/2017.
 */
import java.util.ArrayList;
import java.util.List;
import android.content.ContentValues;
import android.content.Context;
import android.database.Cursor;
import android.database.SQLException;
import android.database.sqlite.SQLiteDatabase;
public class NascenteDAO {
    private SQLiteDatabase database;
    private BaseDAO dbHelper; //Campos da tabela nascente
    private String[] colunas = {BaseDAO.NASCENTE_ID,
        BaseDAO.NASCENTE_NOME,
        BaseDAO.NASCENTE_LATITUDE,
        BaseDAO.NASCENTE_LONGITUDE,
        BaseDAO.NASCENTE_VALOR1,
        BaseDAO.NASCENTE_VALOR2,
        BaseDAO.NASCENTE_VALOR3,
        BaseDAO.NASCENTE_VALOR4,
        BaseDAO.NASCENTE_VALOR5,
        BaseDAO.NASCENTE_VALOR6,
        BaseDAO.NASCENTE_VALOR7,
        BaseDAO.NASCENTE_VALOR8,
        BaseDAO.NASCENTE_VALOR9,
        BaseDAO.NASCENTE_SOMA1,
        BaseDAO.NASCENTE_VALOR10,
        BaseDAO.NASCENTE_VALOR11,
        BaseDAO.NASCENTE_VALOR12,
        BaseDAO.NASCENTE_SOMA2,
        BaseDAO.NASCENTE_RESULTADOGLOBAL };
    public NascenteDAO(Context context) {
        dbHelper = new BaseDAO(context);
    }
    public void open() throws SQLException {
        database = dbHelper.getWritableDatabase();
    }
    public void close() {
        dbHelper.close();
    }
    public long Inserir(Nascentes pValue) {
        ContentValues values = new ContentValues();
        //Carregar os valores nos campos do nascente que será incluído
        values.put(BaseDAO.NASCENTE_NOME, pValue.getNome());
        values.put(BaseDAO.NASCENTE_LATITUDE, pValue.getLatitude());
        values.put(BaseDAO.NASCENTE_LONGITUDE, pValue.getLongitude());
        values.put(BaseDAO.NASCENTE_VALOR1, pValue.getValor1());
        values.put(BaseDAO.NASCENTE_VALOR2, pValue.getValor2());
        values.put(BaseDAO.NASCENTE_VALOR3, pValue.getValor3());
        values.put(BaseDAO.NASCENTE_VALOR4, pValue.getValor4());
        values.put(BaseDAO.NASCENTE_VALOR5, pValue.getValor5());
        values.put(BaseDAO.NASCENTE_VALOR6, pValue.getValor6());
        values.put(BaseDAO.NASCENTE_VALOR7, pValue.getValor7());
        values.put(BaseDAO.NASCENTE_VALOR8, pValue.getValor8());
        values.put(BaseDAO.NASCENTE_VALOR9, pValue.getValor9());
        values.put(BaseDAO.NASCENTE_SOMA1, pValue.getSoma1());
        values.put(BaseDAO.NASCENTE_VALOR10, pValue.getValor10());
    }
}

```

```

        values.put(BaseDAO.NASCENTE_VALOR11, pValue.getValor11());
        values.put(BaseDAO.NASCENTE_VALOR12, pValue.getValor12());
        values.put(BaseDAO.NASCENTE_SOMA2, pValue.getSoma2());
        values.put(BaseDAO.NASCENTE_RESULTADOGLOBAL,
pValue.getResultadoGlobal());
        return database.insert(BaseDAO.TBL_NASCENTE, null, values);
    }

    public int Alterar(Nascentes pValue) {
        long id = pValue.getId();
        ContentValues values = new ContentValues();
        //Carregar os novos valores nos campos que serão alterados
        values.put(BaseDAO.NASCENTE_NOME, pValue.getNome());
        values.put(BaseDAO.NASCENTE_LATITUDE, pValue.getLatitude());
        values.put(BaseDAO.NASCENTE_LONGITUDE, pValue.getLongitude());
        values.put(BaseDAO.NASCENTE_VALOR1, pValue.getValor1());
        values.put(BaseDAO.NASCENTE_VALOR2, pValue.getValor2());
        values.put(BaseDAO.NASCENTE_VALOR3, pValue.getValor3());
        values.put(BaseDAO.NASCENTE_VALOR4, pValue.getValor4());
        values.put(BaseDAO.NASCENTE_VALOR5, pValue.getValor5());
        values.put(BaseDAO.NASCENTE_VALOR6, pValue.getValor6());
        values.put(BaseDAO.NASCENTE_VALOR7, pValue.getValor7());
        values.put(BaseDAO.NASCENTE_VALOR8, pValue.getValor8());
        values.put(BaseDAO.NASCENTE_VALOR9, pValue.getValor9());
        values.put(BaseDAO.NASCENTE_SOMA1, pValue.getSoma1());
        values.put(BaseDAO.NASCENTE_VALOR10, pValue.getValor10());
        values.put(BaseDAO.NASCENTE_VALOR11, pValue.getValor11());
        values.put(BaseDAO.NASCENTE_VALOR12, pValue.getValor12());
        values.put(BaseDAO.NASCENTE_SOMA2, pValue.getSoma2());
        values.put(BaseDAO.NASCENTE_RESULTADOGLOBAL,
pValue.getResultadoGlobal());
        //Alterar o registro com base no ID
        return database.update(BaseDAO.TBL_NASCENTE, values,
BaseDAO.NASCENTE_ID + " = " + id, null);
    }

    public void Excluir(Nascentes pValue) {
        long id = pValue.getId();
        //Exclui o registro com base no ID
        database.delete(BaseDAO.TBL_NASCENTE, BaseDAO.NASCENTE_ID + " = " +
id, null);
    }

    public List<Nascentes> Consultar() {
        List<Nascentes> lstNascente = new ArrayList<Nascentes>();
        //Consulta para trazer todos os dados da tabela Nascentes ordenados
pela coluna Nome
        Cursor cursor = database.query(BaseDAO.TBL_NASCENTE, colunas,
null, null, null, null, BaseDAO.NASCENTE_NOME);
        cursor.moveToFirst();
        while (!cursor.isAfterLast()) {
            Nascente lNascente = cursorToContato(cursor);
            lstNascente.add(lNascente);
            cursor.moveToNext();
        }
        //Tenha certeza que você fechou o cursor
        cursor.close();
        return lstNascente;
    }

    //Converter o Cursor de dados no objeto Nascente
    private Nascentes cursorToContato(Cursor cursor) {
        Nascentes lNascente = new Nascentes();
        lNascente.setId(cursor.getLong(0));
        lNascente.setNome(cursor.getString(1));
    }

```

```
lNascente.setLatitude(cursor.getString(2));
lNascente.setLongitude(cursor.getString(3));
lNascente.setValor1(cursor.getString(4));
lNascente.setValor2(cursor.getString(5));
lNascente.setValor3(cursor.getString(6));
lNascente.setValor4(cursor.getString(7));
lNascente.setValor5(cursor.getString(8));
lNascente.setValor6(cursor.getString(9));
lNascente.setValor7(cursor.getString(10));
lNascente.setValor8(cursor.getString(11));
lNascente.setValor9(cursor.getString(12));
lNascente.setSomal(cursor.getString(13));
lNascente.setValor10(cursor.getString(14));
lNascente.setValor11(cursor.getString(15));
lNascente.setValor12(cursor.getString(16));
lNascente.setSoma2(cursor.getString(17));
lNascente.setResultadoGlobal(cursor.getString(18));
return lNascente;
    }
}
```

Classe Nascentes

```

package com.example.jean.irpan_00;
/**
 * Created by JEAN on 22/06/2017.
 */
import java.io.Serializable;
public class Nascentes implements Serializable {
    private static final long serialVersionUID = 1L;
    public Nascentes() {
        nome = "";
        longitude = "";
        latitude = "";
        valor1 = "";
        valor2 = "";
        valor3 = "";
        valor4 = "";
        valor5 = "";
        valor6 = "";
        valor7 = "";
        valor8 = "";
        valor9 = "";
        valor10 = "";
        valor11 = "";
        valor12 = "";
        soma1 = "";
        soma2 = "";
        resultadoGlobal = "";
    }
    private long id;
    private String nome;
    private String longitude;
    private String latitude;
    private String valor1;
    private String valor2;
    private String valor3;
    private String valor4;
    private String valor5;
    private String valor6;
    private String valor7;
    private String valor8;
    private String valor9;
    private String valor10;
    private String valor11;
    private String valor12;
    private String soma1;
    private String soma2;
    private String resultadoGlobal;
    public static long getSerialVersionUID() {
        return serialVersionUID;
    }
    public long getId() {
        return id;
    }
    public void setId(long id) {
        this.id = id;
    }
    public String getNome() {
        return nome;
    }
    public void setNome(String nome) {

```

```
        this.nome = nome;
    }
    public String getLongitude() {
        return longitude;
    }
    public void setLongitude(String longitude) {
        this.longitude = longitude;
    }
    public String getLatitude() {
        return latitude;
    }
    public void setLatitude(String latitude) {
        this.latitude = latitude;
    }
    public String getValor1() {
        return valor1;
    }
    public void setValor1(String valor1) {
        this.valor1 = valor1;
    }
    public String getValor2() {
        return valor2;
    }
    public void setValor2(String valor2) {
        this.valor2 = valor2;
    }
    public String getValor3() {
        return valor3;
    }
    public void setValor3(String valor3) {
        this.valor3 = valor3;
    }
    public String getValor4() {
        return valor4;
    }
    public void setValor4(String valor4) {
        this.valor4 = valor4;
    }
    public String getValor5() {
        return valor5;
    }
    public void setValor5(String valor5) {
        this.valor5 = valor5;
    }
    public String getValor6() {
        return valor6;
    }
    public void setValor6(String valor6) {
        this.valor6 = valor6;
    }
    public String getValor7() {
        return valor7;
    }
    public void setValor7(String valor7) {
        this.valor7 = valor7;
    }
    public String getValor8() {
        return valor8;
    }
    public void setValor8(String valor8) {
        this.valor8 = valor8;    }
}
```

```
public String getValor9() {
    return valor9;
}
public void setValor9(String valor9) {
    this.valor9 = valor9;
}
public String getValor10() {
    return valor10;
}
public void setValor10(String valor10) {
    this.valor10 = valor10;
}
public String getValor11() {
    return valor11;
}
public void setValor11(String valor11) {
    this.valor11 = valor11;
}
public String getValor12() {
    return valor12;
}
public void setValor12(String valor12) {
    this.valor12 = valor12;
}
public String getSoma1() {
    return soma1;
}
public void setSoma1(String soma1) {
    this.soma1 = soma1;
}
public String getSoma2() {
    return soma2;
}
public void setSoma2(String soma2) {
    this.soma2 = soma2;
}
public String getResultadoGlobal() {
    return resultadoGlobal;
}
public void setResultadoGlobal(String resultadoGlobal) {
    this.resultadoGlobal = resultadoGlobal;
}
public String toString(){
    return nome + latitude + longitude;
}
}
```

Classe NascenteUI

```

package com.example.jean.irpan_00;
/**
 * Created by JEAN on 22/06/2017.
 */
import android.app.Activity;
import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.EditText;
import android.widget.RadioButton;
import android.widget.Toast;
public class NascenteUI extends Activity {
    private static final int INCLUIR = 0;
    //private static final int ALTERAR = 1;
    Nascentes lNascentes;
    RadioButton v1;
    RadioButton v2;
    RadioButton v3;
    RadioButton v4;
    RadioButton v5;
    RadioButton v6;
    RadioButton v7;
    RadioButton v8;
    RadioButton v9;
    RadioButton v10;
    RadioButton v11;
    RadioButton v12;
    RadioButton v13;
    RadioButton v14;
    RadioButton v15;
    RadioButton v16;
    RadioButton v17;
    RadioButton v18;
    RadioButton v19;
    RadioButton v20;
    RadioButton v21;
    RadioButton v22;
    RadioButton v23;
    RadioButton v24;
    RadioButton v25;
    RadioButton v26;
    RadioButton v27;
    RadioButton v28;
    RadioButton v29;
    RadioButton v30;
    RadioButton v31;
    RadioButton v32;
    RadioButton v33;
    RadioButton v34;
    RadioButton v35;
    RadioButton v36;
    EditText txtNome;
    EditText txtLatitude;
    EditText txtLongitude;
    int valor1;
    int valor2;
    int valor3;
    int valor4;
    int valor5;

```

```

int valor6;
int valor7;
int valor8;
int valor9;
int valor10;
int valor11;
int valor12;
int soma1;
int soma2;
String resultado;
String antropica;
String natural;
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.nascente);
    try
    {
        final Bundle data = (Bundle) getIntent().getExtras();
        int lint = data.getInt("tipo");
        if (lint == INCLUIR)
        {
            //quando for incluir um contato criamos uma nova instância
            lNascentes = new Nascentes();
        }else{
            //quando for alterar o contato carregamos a classe que veio por Bundle
            lNascentes = (Nascentes) data.getSerializable("nascente");
        }
        //Criação dos objetos da Activity
        v1 = (RadioButton) findViewById(R.id.var1op1);
        v2 = (RadioButton) findViewById(R.id.var1op2);
        v3 = (RadioButton) findViewById(R.id.var1op3);
        v4 = (RadioButton) findViewById(R.id.var2op1);
        v5 = (RadioButton) findViewById(R.id.var2op2);
        v6 = (RadioButton) findViewById(R.id.var2op3);
        v7 = (RadioButton) findViewById(R.id.var3op1);
        v8 = (RadioButton) findViewById(R.id.var3op2);
        v9 = (RadioButton) findViewById(R.id.var3op3);
        v10 = (RadioButton) findViewById(R.id.var4op1);
        v11 = (RadioButton) findViewById(R.id.var4op2);
        v12 = (RadioButton) findViewById(R.id.var4op3);
        v13 = (RadioButton) findViewById(R.id.var5op1);
        v14 = (RadioButton) findViewById(R.id.var5op2);
        v15 = (RadioButton) findViewById(R.id.var5op3);
        v16 = (RadioButton) findViewById(R.id.var6op1);
        v17 = (RadioButton) findViewById(R.id.var6op2);
        v18 = (RadioButton) findViewById(R.id.var6op3);
        v19 = (RadioButton) findViewById(R.id.var7op1);
        v20 = (RadioButton) findViewById(R.id.var7op2);
        v21 = (RadioButton) findViewById(R.id.var7op3);
        v22 = (RadioButton) findViewById(R.id.var8op1);
        v23 = (RadioButton) findViewById(R.id.var8op2);
        v24 = (RadioButton) findViewById(R.id.var8op3);
        v25 = (RadioButton) findViewById(R.id.var9op1);
        v26 = (RadioButton) findViewById(R.id.var9op2);
        v27 = (RadioButton) findViewById(R.id.var9op3);
        v28 = (RadioButton) findViewById(R.id.var10op1);
        v29 = (RadioButton) findViewById(R.id.var10op2);
        v30 = (RadioButton) findViewById(R.id.var10op3);
        v31 = (RadioButton) findViewById(R.id.var11op1);
        v32 = (RadioButton) findViewById(R.id.var11op2);
    }
}

```

```

v33 = (RadioButton) findViewById(R.id.var11op3);
v34 = (RadioButton) findViewById(R.id.var12op1);
v35 = (RadioButton) findViewById(R.id.var12op2);
v36 = (RadioButton) findViewById(R.id.var12op3);
txtNome = (EditText) findViewById(R.id.editText);
txtLatitude = (EditText) findViewById(R.id.editText1);
txtLongitude = (EditText) findViewById(R.id.editText2);
txtNome.setText(lNascentes.getNome());
txtLatitude.setText(lNascentes.getLatitude());
txtLongitude.setText(lNascentes.getLongitude());
valor1 = Integer.parseInt(lNascentes.getValor1());
valor2 = Integer.parseInt(lNascentes.getValor2());
valor3 = Integer.parseInt(lNascentes.getValor3());
valor4 = Integer.parseInt(lNascentes.getValor4());
valor5 = Integer.parseInt(lNascentes.getValor5());
valor6 = Integer.parseInt(lNascentes.getValor6());
valor7 = Integer.parseInt(lNascentes.getValor7());
valor8 = Integer.parseInt(lNascentes.getValor8());
valor9 = Integer.parseInt(lNascentes.getValor9());
valor10 = Integer.parseInt(lNascentes.getValor10());
valor11 = Integer.parseInt(lNascentes.getValor11());
valor12 = Integer.parseInt(lNascentes.getValor12());
if(valor1 == 2){
    v1.setChecked(true);
} else if(valor1 == 4){
    v2.setChecked(true);
} else{
    v3.setChecked(true);
}
if(valor2 == 3){
    v4.setChecked(true);
} else if(valor2 == 6){
    v5.setChecked(true);
} else{
    v6.setChecked(true);
}
if(valor3 == 5){
    v7.setChecked(true);
} else if(valor3 == 10){
    v8.setChecked(true);
} else{
    v9.setChecked(true);
}
if(valor4 == 5){
    v10.setChecked(true);
} else if(valor4 == 10){
    v11.setChecked(true);
} else{
    v12.setChecked(true);
}
if(valor5 == 5){
    v13.setChecked(true);
} else if(valor5 == 10){
    v14.setChecked(true);
} else{
    v15.setChecked(true);
}
if(valor6 == 4){
    v16.setChecked(true);
} else if(valor6 == 8){
    v17.setChecked(true);
}

```

```

    }else{
        v18.setChecked(true);
    }
    if(valor7 == 3){
        v19.setChecked(true);
    }else if(valor7 == 6){
        v20.setChecked(true);
    }else{
        v21.setChecked(true);
    }
    if(valor8 == 1){
        v22.setChecked(true);
    }else if(valor8 == 2){
        v23.setChecked(true);
    }else{
        v24.setChecked(true);
    }if(valor9 == 3){
        v25.setChecked(true);
    }else if(valor9 == 6){
        v26.setChecked(true);
    }else{
        v27.setChecked(true);
    }
    if(valor10 == 3){
        v28.setChecked(true);
    }else if(valor10 == 6){
        v29.setChecked(true);
    }else{
        v30.setChecked(true);
    }
    if(valor11 == 4){
        v31.setChecked(true);
    }else if(valor11 == 8){
        v32.setChecked(true);
    }else{
        v33.setChecked(true);
    }
    if(valor12 == 5){
        v34.setChecked(true);
    }else if(valor12 == 10){
        v35.setChecked(true);
    }else{
        v36.setChecked(true);
    }
    //Carregando os objetos com os dados do nascente
    //caso seja uma inclusão ele virá carregado com os atributos text
    //definido no arquivo MainActivity.xml
    }catch (Exception e) {
        trace("Preencha todos os campos!");
        //"Aqui tem um problema na conversão de string para inteiro. O
        programa começa com os checkbox sem preenchimento ai o valor fica zero"
    }
}
public void btnConfirmar_click(View view)
{
    try
    {
        //Quando confirmar a inclusão ou alteração deve-se devolver
        //o registro com os dados preenchidos na tela e informar
        //o RESULT_OK e em seguida finalizar a Activity
        if (v1.isChecked()){

```

```
        valor1=2;
    }
    if (v2.isChecked()) {
        valor1=4;
    }
    if (v3.isChecked()) {
        valor1=6;
    }
    if (v4.isChecked()) {
        valor2=3;
    }
    if (v5.isChecked()) {
        valor2=6;
    }
    if (v6.isChecked()) {
        valor2=9;
    }
    if (v7.isChecked()) {
        valor3=5;
    }
    if (v8.isChecked()) {
        valor3=10;
    }
    if (v9.isChecked()) {
        valor3=15;
    }
    if (v10.isChecked()) {
        valor4=5;
    }
    if (v11.isChecked()) {
        valor4=10;
    }
    if (v12.isChecked()) {
        valor4=15;
    }
    if (v13.isChecked()) {
        valor5=5;
    }
    if (v14.isChecked()) {
        valor5=10;
    }
    if (v15.isChecked()) {
        valor5=15;
    }
    if (v16.isChecked()) {
        valor6=4;
    }
    if (v17.isChecked()) {
        valor6=8;
    }
    if (v18.isChecked()) {
        valor6=12;
    }
    if (v19.isChecked()) {
        valor7=3;
    }
    if (v20.isChecked()) {
        valor7=6;
    }
    if (v21.isChecked()) {
        valor7=9;
    }
}
```

```

if (v22.isChecked()) {
    valor8=1;
}
if (v23.isChecked()) {
    valor8=2;
}
if (v24.isChecked()) {
    valor8=3;
}
if (v25.isChecked()) {
    valor9=3;
}
if (v26.isChecked()) {
    valor9=6;
}
if (v27.isChecked()) {
    valor9=9;
}
if (v28.isChecked()) {
    valor10=3;
}
if (v29.isChecked()) {
    valor10=6;
}
if (v30.isChecked()) {
    valor10=9;
}
if (v31.isChecked()) {
    valor11=4;
}
if (v32.isChecked()) {
    valor11=8;
}
if (v33.isChecked()) {
    valor11=12;
}
if (v34.isChecked()) {
    valor12=5;
}
if (v35.isChecked()) {
    valor12=10;
}
if (v36.isChecked()) {
    valor12=15;
}
soma1 =
valor1+valor2+valor3+valor4+valor5+valor6+valor7+valor8+valor9;
soma2 = valor10+valor11+valor12;
Intent data = new Intent();
lNascentes.setName(txtNome.getText().toString());
lNascentes.setLatitude(txtLatitude.getText().toString());
lNascentes.setLongitude(txtLongitude.getText().toString());
lNascentes.setValor1(String.valueOf(valor1));
lNascentes.setValor2(String.valueOf(valor2));
lNascentes.setValor3(String.valueOf(valor3));
lNascentes.setValor4(String.valueOf(valor4));
lNascentes.setValor5(String.valueOf(valor5));
lNascentes.setValor6(String.valueOf(valor6));
lNascentes.setValor7(String.valueOf(valor7));
lNascentes.setValor8(String.valueOf(valor8));
lNascentes.setValor9(String.valueOf(valor9));
lNascentes.setValor10(String.valueOf(valor10));

```

```

lNascentes.setValor11(String.valueOf(valor11));
lNascentes.setValor12(String.valueOf(valor12));
if(soma2 >= 31) {
    if(somal>=74) {
        antropica = "Muito Baixo";
        natural = "Muito Baixo";
        resultado = "Muito Baixo";
    }else if(somal>=60 && somal<73) {
        antropica = "Baixo";
        natural = "Muito Baixo";
        resultado = "Baixo";
    }else if(somal>=46 && somal<59) {
        antropica = "Media";
        natural = "Muito Baixo";
        resultado = "Medio Baixo";
    }else if(somal>=32 && somal<45) {
        antropica = "Elevado";
        natural = "Muito Baixo";
        resultado = "Medio Baixo";
    }else {
        antropica = "Muito Elevado";
        natural = "Muito Baixo";
        resultado = "Medio";
    }
}
}else if(soma2 >= 25 && soma2 <31) {
    if(somal>=74) {
        antropica = "Muito Baixo";
        natural = "Baixo";
        resultado = "Baixo";
    }else if(somal>=60 && somal<73) {
        antropica = "Baixo";
        natural = "Baixo";
        resultado = "Baixo";
    }else if(somal>=46 && somal<59) {
        antropica = "Media";
        natural = "Baixo";
        resultado = "Medio Baixo";
    }else if(somal>=32 && somal<45) {
        antropica = "Elevado";
        natural = "Baixo";
        resultado = "Medio";
    }else {
        antropica = "Muito Elevado";
        natural = "Baixo";
        resultado = "Medio Alto";
    }
}
}else if(soma2 >= 19 && soma2 <25) {
    if(somal>=74) {
        antropica = "Muito Baixo";
        natural = "Media";
        resultado = "Medio Baixo";
    }else if(somal>=60 && somal<73) {
        antropica = "Baixo";
        natural = "Media";
        resultado = "Medio Baixo";
    }else if(somal>=46 && somal<59) {
        antropica = "Media";
        natural = "Media";
        resultado = "Medio";
    }else if(somal>=32 && somal<45) {
        antropica = "Elevado";
    }
}
}

```

```

        natural = "Media";
        resultado = "Medio Alto";
    }else{
        antropica = "Muito Elevado";
        natural = "Media";
        resultado = "Medio Alto";
    }
}else if(soma2 >= 13 && soma2 <18){
    if(somal>=74){
        antropica = "Muito Baixo";
        natural = "Elevado";
        resultado = "Medio Baixo";
    }else if(somal>=60 && somal<73){
        antropica = "Baixo";
        natural = "Elevado";
        resultado = "Medio";
    }else if(somal>=46 && somal<59){
        antropica = "Media";
        natural = "Elevado";
        resultado = "Medio Alto";
    }else if(somal>=32 && somal<45){
        antropica = "Elevado";
        natural = "Elevado";
        resultado = "Alto";
    }else{
        antropica = "Muito Elevado";
        natural = "Elevado";
        resultado = "Alto";
    }
}
}else{
    if(somal>=74){
        antropica = "Muito Baixo";
        natural = "Muito Elevado";
        resultado = "Medio";
    }else if(somal>=60 && somal<73){
        antropica = "Baixo";
        natural = "Muito Elevado";
        resultado = "Medio Alto";
    }else if(somal>=46 && somal<59){
        antropica = "Media";
        natural = "Muito Elevado";
        resultado = "Medio Alto";
    }else if(somal>=32 && somal<45){
        antropica = "Elevado";
        natural = "Muito Elevado";
        resultado = "Alto";
    }else{
        antropica = "Muito Elevado";
        natural = "Muito Elevado";
        resultado = "Muito Alto";
    }
}
lNascentes.setSoma1(antropica);
lNascentes.setSoma2(natural);
lNascentes.setResultGlobal(resultado);
data.putExtra("nascente", lNascentes);
setResult(Activity.RESULT_OK, data);
finish();
}catch (Exception e) {
    trace("Erro : " + e.getMessage());
}
}

```

```

}
public void btnCancelar_click(View view)
{
    try
    {
        //Quando for simplesmente cancelar a operação de inclusão
        //ou alteração deve-se apenas informar o RESULT_CANCELED
        //e em seguida finalizar a Activity
        setResult(Activity.RESULT_CANCELED);
        finish();
    } catch (Exception e) {
        trace("Erro : " + e.getMessage());
    }
}
public void toast (String msg)
{
    Toast.makeText (getApplicationContext(), msg,
Toast.LENGTH_SHORT).show ();
}
private void trace (String msg)
{
    toast (msg);
}
}

```

CODIGO DO PACOTE AndroidManifest

Classe AndroidManifest.xml

```

<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
package="com.example.jean.irpan_00">

<application

    android:allowBackup="true"
    android:icon="@mipmap/ic_launcher"
    android:label="@string/app_name"
    android:supportsRtl="true"
    android:theme="@style/AppTheme">
    <activity android:name=".main">
        <intent-filter>
            <action android:name="android.intent.action.MAIN" />

            <category android:name="android.intent.category.LAUNCHER" />
        </intent-filter>
    </activity>
    <activity android:name="NascenteUI" android:label="IRPAN-00"/>
</application>

</manifest>

```

APÊNDICE B - TELAS DO ANDROID STUDIO

TELAS DO ANDROIDSTUDIO COM TODAS AS CLASSES

