**UNIVERSIDADE KATYAVALA BWILA**

**Instituto Superior Politécnico**

**DEPARTAMENTO DE INFORMÁTICA**

**DESENHO DE UMA INTERFACE MÓVEL PARA UM SISTEMA DE NOTIFICAÇÃO EPIDEMIOLÓGICO**

Nº\_\_\_\_/2019

**AUTOR**: Olívio Constantino António

Benguela

2019

Inovação– Investigação–Desenvolvimento

*Região Académica II – Benguela e Kwanza Sul*

**UNIVERSIDADE KATYAVALA BWILA**

**Instituto Superior Politécnico**

Departamento de Informática

Trabalho de Fim de Curso Apresentado Para Obtenção do Grau de Licenciado Em Ciências da Computação

**DESENHO DE UMA INTERFACE MÓVEL PARA UM SISTEMA DE NOTIFICAÇÃO EPIDEMIOLÓGICO**

**AUTOR**: Olívio Constantino António

**ORIENTADOR**: Dr. Lázaro Emílio Makili

**CO-ORIENTADOR:** Me. Adán Montiel Valdez

Benguela

2019

Inovação– Investigação–Desenvolvimento

*Região Académica II – Benguela e Kwanza Sul*

**DEDICATÓRIA**

Dedico este trabalho a minha familía, em especialmente ao meu pai Abrantes António e minha mãe Victória Constantino.

**AGRADECIMENTOS**

O presente trabalho representa os cinco anos de formação e não só, que nada seria possível se não fosse pela ajuda de Deus, inúmeras pessoas e Instituições que desde o inicio da licenciatura até ao presente momento não mediram esforços para me estender a mão, assim sendo aos meus agradecimentos vão:

A Deus todo poderoso, pela vida, misericórdia, saúde, amor, luz e tudo quanto tem operado em minha vida, sem Deus nada seria possível.

A todos professores que depositaram em mim seus conhecimentos, sabedoria e educação desde o início da minha vida académica.

A todos professores do ISP-UKB, em especial ao Dr. Lázaro Emílio Makili e Me. Adan Montiel que incansavelmente orientaram-me na elaboração deste trabalho.

Aos meus pais Abrantes António e Victória Constantino pela educação, amor e apoio constante em toda minha vida.

A minha tia Elizabeth Huvi, pelo suporte, conselho e tudo que tem feito em minha vida.

Ao Instituto Nacional de Bolsas de Estudos Internas (INAGBE) pelo apoio financeiro durante os cinco anos de formação.

Ao Corpo-Directivo e todos os funcionários do ISP-UKB.

Aos meus amigos, colegas e familiares que sempre estiveram dispostos a me ajudar, em especial aos integrantes do grupo ‘HEOAsoft’.

A todos aqueles que directas ou indirectamente contribuíram para que este trabalho se torna-se uma realidade.

A todos o meu muito obrigado.

# **RESUMO**

O processo de notificação de doenças de interesse epidemiológico têm sido feito com muitas dificuldades, isto porque os dados são registados em papéis e existe um atraso no envio e tratamento destas informações, assim sendo torna o processo demorado, não eficaz e com um enorme risco de se cometerem erros, isto porque não existe uma aplicação no qual os hospitais, postos de saúde e clínicas particulares registam as ocorrências diárias das doenças diagnosticadas ou comprovadas em laboratórios, a fim de manter um maior controlo pelas entidades competentes. Se pretende desenhar e desenvolver uma interface que funcione em dispositivos móveis para que os médicos e técnicos de laboratórios possam armazenar, e notificar casos de doenças de notificação epidemiológico. A aplicação desenvolveu-se no Android studio, utilizou-se para tal a linguagem de programação java, usou-se o visual paradgima e brModelo como ferramenta de apoio em análise e desenho do sistema com a metodologia Rational Unified Process(RUP). Usou-se os métodos históricos e dedutivos, a investigação documental e o pesquisa bibliográficas como tecnícas de recolha de informações. Com isto esperasse garantir um melhor controlo, podendo assim adoptar métodos preventivos para diminuir os índices destas doenças, diminuindo também gastos financeiros e o tempo de comunicação das informações.

Palavras-chaves: Epidemiologia, Android, vigilância, saúde pública, notificação.

# **ABSTRACT**

For doctors and technicians from Laboratories to inform the Ministry of Health about diseases of interest of mandatory notification in order to act on this information, have been made with many difficulties, this because the data are registered in papers by these and delivered the persons responsible for the notification epidemiological study. This process is very time-consuming, not done effectively and at a huge risk of making mistakes, because there is no application in which hospitals, clinics and private clinics record the daily occurrences of the diagnosed diseases in order to maintain control by the competent authorities. It is intended to design and develop an interface that works on mobile devices so that doctors and laboratory technicians can store and report cases of diseases of epidemiological interest. To this end, a number of tools and knowledge were used in connection with the new technologies, thus ensuring better control, thus allowing preventive methods to be used to reduce the rates of these diseases, also reducing financial expenses and information communication time.

Keywords: Epidemiology, Android, surveillance, public health, notification.

**ÍNDICE GERAL**

[RESUMO v](#_Toc27216304)

[ABSTRACT vi](#_Toc27216305)

[ÍNDICE DE FIGURAS x](#_Toc27216306)

[ÍNDICE DE TABELAS xi](#_Toc27216307)

[LISTA DE SIGLAS E ABREVIATURAS xii](#_Toc27216308)

[INTRODUÇÃO 13](#_Toc27216309)

[CAPÍTULO I: FUNDAMENTAÇÃO TEÓRICA SOBRE VIGILÂNCIA EPIDEMIOLÓGICA 19](#_Toc27216310)

[1.1. INTRODUÇÃO 20](#_Toc27216311)

[1.2. OBJECTIVOS DA VIGILÂNCIA EPIDEMIOLÓGICA 20](#_Toc27216312)

[1.3. PROCESSOS DA VIGILÂNCIA EPIDEMIOLÓGICA 21](#_Toc27216313)

[1.4. SITUAÇÃO ACTUAL DA VIGILÂNCIA EPIDEMIOLÓGICA EM ANGOLA 23](#_Toc27216314)

[1.5. CONCLUSÃO 24](#_Toc27216315)

[CAPÍTULO II: DESCRIÇÃO DAS FERRAMENTAS E TECNOLOGIAS UTILIZADAS PARA O DESENVOLVIMENTO 25](#_Toc27216316)

[2.1. INTRODUÇÃO 26](#_Toc27216317)

[2.2. METODOLOGIAS DE DESENVOLVIMENTO 26](#_Toc27216318)

[2.3. FERRAMENTAS CASE 27](#_Toc27216319)

[2.3.1. VISUAL PARADIGM 27](#_Toc27216320)

[2.3.2. STARUML 28](#_Toc27216321)

[2.3.3. BRMODELO 28](#_Toc27216322)

[2.4. SISTEMAS OPERATIVOS PARA DISPOSITIVOS MÓVEIS 28](#_Toc27216323)

[2.4.1. ANDROID 29](#_Toc27216324)

[2.4.2. iOS: IPHONE 29](#_Toc27216325)

[2.4.3. WINDOWS PHONE 30](#_Toc27216326)

[2.4.4. BLACKBERRY 30](#_Toc27216327)

[2.4.5. SYMBIAN 31](#_Toc27216328)

[2.5. PORQUÊ A ESCOLHA DO ANDROID? 31](#_Toc27216329)

[2.5.1. CARACTERÍSTICAS DO ANDROID 32](#_Toc27216332)

[2.6. OPEN HANDSET ALLIANCE (OHA) 33](#_Toc27216333)

[2.7. LINGUAGENS DE PROGRAMAÇÃO 33](#_Toc27216334)

[2.7.1. JAVA 34](#_Toc27216335)

[2.7.2. KOTLIN 35](#_Toc27216336)

[2.7.3. C/C++ 35](#_Toc27216337)

[2.7.4. XML 36](#_Toc27216338)

[2.7.5. PHP 36](#_Toc27216339)

[2.7.6. JSON 37](#_Toc27216340)

[2.8. SISTEMAS GESTORES DE BASE DE DADOS 37](#_Toc27216341)

[2.8.1. SQLITE 38](#_Toc27216342)

[2.8.2. MYSQL 38](#_Toc27216343)

[2.8.3. FIREBASE REALTIME DATABASE 38](#_Toc27216344)

[2.9. AMBIENTE DE DESENVOLVIMENTO (IDE) PARA ANDROID 39](#_Toc27216345)

[2.9.1. ANDROID STUDIO 39](#_Toc27216346)

[2.9.2. ECLIPSE 40](#_Toc27216347)

[2.9.3. XAMARIM 40](#_Toc27216348)

[2.10. ANDROID SDK TOOLS 41](#_Toc27216349)

[2.11. WEB SERVICE 41](#_Toc27216350)

[2.12. DISTRIBUIÇÃO DAS APLICAÇÕES 42](#_Toc27216351)

[2.13. CONCLUSÃO 42](#_Toc27216352)

[CAPÍTULO III: DESCRIÇÃO DO PROCESSO DE DESENVOLVIMENTO 43](#_Toc27216353)

[3.1. INTRODUÇÃO 44](#_Toc27216356)

[3.2. DESCRIÇÃO DO NEGÓCIO 44](#_Toc27216357)

[3.3. PROCESSOS DO NEGÓCIO 45](#_Toc27216358)

[3.4. REGRAS DO NEGÓCIO 46](#_Toc27216359)

[3.5. ACTORES DO NEGÓCIO 46](#_Toc27216360)

[3.6. TRABALHADORES DO NEGÓCIO 47](#_Toc27216361)

[3.7. CASOS DE USO DO NEGÓCIO 47](#_Toc27216362)

[3.8. DIAGRAMA DE CASOS DE USO DO NEGÓCIO 48](#_Toc27216363)

[3.9. DESCRIÇÃO TEXTUAL E DIAGRAMA DE ACTIVIDADE DE CASOS DE USO 49](#_Toc27216364)

[3.9.1. CASO DE USO FAZER CONSULTA MÉDICA 49](#_Toc27216365)

[3.9.2. CASO DE USO NOTIFICAR CASO 51](#_Toc27216366)

[3.10. DESCRIÇÃO DO SISTEMA PROPOSTO 52](#_Toc27216367)

[3.11. ANÁLISE DE REQUISITOS 52](#_Toc27216368)

[3.11.1. REQUISITOS FUNCIONAIS 53](#_Toc27216369)

[3.11.2. REQUISITOS NÃO FUNCIONAIS 53](#_Toc27216370)

[3.12. ACTORES DO SISTEMA 54](#_Toc27216371)

[3.13. CASOS DE USO DO SISTEMA 55](#_Toc27216372)

[3.14. DIAGRAMA DE CASOS DE USO DO SISTEMA 56](#_Toc27216373)

[3.15. DESCRIÇÃO TEXTUAL DOS CASOS DE USO DO SISTEMA 57](#_Toc27216374)

[3.16. DIAGRAMAS DE INTERACÇÃO 58](#_Toc27216375)

[3.16.1. DIAGRAMAS DE COLABORAÇÃO 59](#_Toc27216376)

[3.16.2. DIAGRAMAS DE SEQUÊNCIA 60](#_Toc27216377)

[3.17. DIAGRAMA DE CLASSES 61](#_Toc27216378)

[3.18. MODELO CONCEITUAL (DIAGRAMA ENTIDADE E RELAÇÃO) 62](#_Toc27216379)

[3.19. MODELO LÓGICO (MODELO RELACIONAL) 63](#_Toc27216380)

[3.20. DIAGRAMA DE IMPLEMENTAÇÃO 64](#_Toc27216381)

[3.21. DIAGRAMA DE IMPLANTAÇÃO 64](#_Toc27216382)

[3.22. FLUXO DE TESTE 65](#_Toc27216383)

[3.23. CONCLUSÃO 66](#_Toc27216384)

[CONCLUSÕES 67](#_Toc27216385)

[RECOMENDAÇÕES 68](#_Toc27216386)

[BIBLIOGRAFIA 69](#_Toc27216387)

[ANEXOS 71](#_Toc27216388)

[APÊNDICES 74](#_Toc27216389)

**ÍNDICE DE FIGURAS**

[*Figura 1-Principais sistemas operativos para celulares.* 29](#_Toc27216390)

[*Figura 2-Diagrama de caso de uso do negócio* 48](#_Toc27216391)

[*Figura 3-Diagrama de actividades do caso de uso do negócio "Fazer consulta médica"* 50](#_Toc27216392)

[*Figura 4-Diagrama de actividades do caso de uso do negócio "notificar caso"* 51](#_Toc27216393)

[*Figura 5-Diagrama de caso de uso do sistema* 56](#_Toc27216394)

[*Figura 6-Diagrama de colaboração "entrar no sistema"* 59](#_Toc27216395)

[*Figura 7-Diagrama de colaboração "notificar caso"* 59](#_Toc27216396)

[*Figura 8-Diagrama de sequência "entrar no sistema"* 60](#_Toc27216397)

[*Figura 9-Diagrama de sequência "notificar caso"* 60](#_Toc27216398)

[*Figura 10-Diagrama de classes* 61](#_Toc27216399)

[*Figura 11-Modelo conceitual da base de dados* 62](#_Toc27216400)

[*Figura 12-Modelo lógico da base de dados* 63](#_Toc27216401)

[*Figura 13-Diagrama de implementação* 64](#_Toc27216402)

[*Figura 14-Diagrama de implantação* 65](#_Toc27216403)

[*Figura 15-Formulário de investigação de casos de cólera* 71](#_Toc27216404)

[*Figura 16-Formulário de notificação de casos de malária* 72](#_Toc27216405)

[*Figura 17-Lista de algumas doenças e suas prioridades de notificação* 73](#_Toc27216406)

[*Figura 18-Tela de login da aplicação* 74](#_Toc27216407)

[*Figura 19-Tela menu principal da aplicação* 74](#_Toc27216408)

[*Figura 20-Tela dados do paciente da aplicação* 75](#_Toc27216409)

[*Figura 21-Tela formulário notificar caso.* 75](#_Toc27216410)

[*Figura 22-Activity notificações* 76](#_Toc27216411)

ÍNDICE DE TABELAS

[Tabela 1: Comparação dos sistemas operativos móveis 1º trimestre de 2016 à 1º trimestre de 2017 32](#_Toc530842864)

[Tabela 2: Regras do negócio 46](#_Toc530842865)

[Tabela 3: Actores do negócio 46](#_Toc530842866)

[Tabela 4: Trabalhadores do negócio 47](#_Toc530842867)

[Tabela 5: Casos de uso do negócio 48](#_Toc530842868)

[Tabela 6: Descrição do caso de uso do negócio “marcar consulta” 49](#_Toc530842869)

[Tabela 7: Descrição do caso de uso do negócio “notificar caso” 51](#_Toc530842870)

[Tabela 8: Actores do sistema 55](#_Toc530842871)

[Tabela 9: Descrição do caso de uso do sistema “iniciar sessão” 57](#_Toc530842872)

[Tabela 10: Descrição do caso de uso do sistema “notificar caso” 58](#_Toc530842873)

[Tabela 11: Caso de teste “entrar no sistema” 66](#_Toc530842874)

# **LISTA DE SIGLAS E ABREVIATURAS**

**AVD -** *Android Virtual Device* (Dispositivo Virtual Android)

**CASE -***Computer-Aided Software Engineering* (Engenharia de *Software* Auxiliada por Computador)

**CRUD -** *Create, Read, Update and Delete* (Criar, Ler, Atualizar e Excluir)

**DER -** Diagrama entidade relação

**ER -** Entidade relação

**JDK -** *Java Development Kit* (Kit de Desenvolvimento do Java)

**HTTP -** *HyperText Transfer Protocol* (Protocolo de Transferência de Hipertexto)

**IA-** Inteligência Artificial

**IDC**- *International Data Comparation* (Comparação Internacional de Dados)

**IDE-** *Integrated Development Environment* (Ambiente de Desenvolvimento Integrado)

**INSP-** Instituto Nacional de Saúde Pública

**OMS -** Organização Mundial da Saúde

**PHP -** *Hypertext Preprocessor* **(**Pré-Procestador de Hipertexto)

**RAM -** *Random Access Memory* (Memória de Acesso Aleatório)

**RUP -** *Rational Unified Process* (Processo Racional Unificado)

**SDK -** *Android Software Development Kit* (Kit de Desenvolvimento de Software para Android)

**SQL -** *Structured Query Language* (Linguagem de Consulta Estruturado)

**TI -** Tecnologia de Informação

**UML -** *Unified Modeling Language* (Linguagem de Modelagem Unificada

**UP -** *Unified Process* (Processo Unificado)

**XML -** *eXtensible Markup Language* (Linguagem de Marcação extensível)

**XP -** *Extreme Programming* (Programação Extrema)

# **INTRODUÇÃO**

A mobilidade, rapidez, facilidade, baixo custo para alguns casos, eficácia bem como avanço da própria tecnologia, são alguns factores que estão elevando cada vez mais as aplicações móveis ao mais alto nível.

Hoje em dia os dispositivos móveis (telemóveis, tablets, etc…), são utilizados em toda parte do mundo para realizar inúmeras tarefas tendo em conta as suas vantagens, apesar de que em alguns lugares usa-se com maior frequência em relação a outros, isto devido a vários factores. Os dispositivos móveis são hoje utilizados para diferentes actividades diárias do homem.

Uma aplicação móvel capaz de notificar os casos de diagnosticos e exames de laboratório de interesse epidemiológico, mediante envio de informações precisas e a tempo integral, enquadrar-se perfeitamente nas tarefas das apilicações móveis tendo em conta os recursos que os dispositivos móveis nos oferecem.

Na identificação de casos e eventos prioritários, é necessário defenir as fontes de informação sobre eventos de saúde no município, incluido os pontos de contacto da comunidade com os serviços de saúde, destacam-se as unidades sanitárias, agentes comunitários de saúde, proteção civíl, serviços veterinários, laboratórios industriais, entre outros (Ministério da Saúde, 2013, p. 31). As notificações de maior frequência a se registarem no país são as notificações centradas nos resultados dos diagnósticos e laboratórios, motivo pelo qual o nosso trabalho está centrado inicialmente nas notificações com origens nestas fontes.

No presente trabalho apresenta-se possíveis soluções bem como os técnicas utilizadas para alcançá-lo até construir a interface móvel para um sistema de vigilância epidemiológica.

O processo de notificação de casos de interesse epidemiológico é feito de forma manual, o que tem com isto causado muitas dificuldades principalmente na recolha de informações, pois as vezes cometem-se erros, outras vezes as informações chegam tardiamente às entidades competentes facilitando assim a propagação de certas doenças que poderiam ser previamente controladas se o conhecimento das mesmas doenças tivesse existido com maior antecedência. Tendo em conta estes factores, a escolha do tema **justifica-se** pelo facto de existir insuficiência no processo de notificação para casos de interesse epidemiológico como referido acima, a implementação de uma aplicação que funcione em dispositivos móveis, favoreceria a realização do processo com maior antecedência, rapidez e segurança. Desta forma tem-se o seguinte **problema de investigação,** como melhorar o processo de notificação de casos de doenças de interesse epidemiológico no país? Para dar solução a este problema tem-se como **objecto de estudo**, conjunto de ferramentas informáticas para o desenvolvimento de aplicações móveis nos processos de vigilância epidemiológica utilizando linguagens de alto nível e sistema gestor de base de dados. O **campo de acção** será o conjunto de ferramentas informáticas para o desenvolvimento de uma aplicação móvel para um sistema de notificação de doenças de interesse epidemiológico. Com isto pretende-se alcançar o seguinte **objectivo geral**, desenvolver uma interface móvel para o processo de notificação de casos de interesse epidemiológico, utilizando linguagens de alto nível e sistemas gestores de bases de dados. A fim de concretizar este objectivo traçou-se os seguintes **objectivos específicos**:

* Revisar a bibliografia relacionada ao tema,
* Seleccionar as ferramentas para o desenvolvimento da aplicação,
* Elaborar o projecto de Engenharia de *Software*,
* Desenhar e desenvolver a interface móvel,
* Testar a aplicação.

Como qualquer outro *software*, uma aplicação móvel segue um conjunto de etapas para ser desenvolvido, e, é importante primeiro escolher uma metodologia de desenvolvimento e em seguida escolher os métodos e técnicas de investigação para recolha de informações que nos informam das mais básicas as mais complexas regras com que funciona o negócio a modelar.

Por ser uma metodologia híbrida, optou-se pelo Processo Unificado Racional (RUP) como metodologia de desenvolvimento de software.

“Um método é o conjunto das actividades sistemáticas e racionais que, com maior segurança e economia, permite alcançar o objectivo - conhecimentos válidos e verdadeiros, traçando o caminho a ser seguido, detectando erros e auxiliando as decisões do cientista” (Marconi & Lakatos, 2003, p. 83).

Utilizou-se os seguintes métodos:

**Método Dedutivo:** O método dedutivo é um método que parte do geral para o particular, umas das suas características é que não existem graduações intermediárias, pois as premissas sustentam de modo completo a conclusão em correcto ou incorrecto (Marconi & Lakatos, 2003).

Com este método entendeu-se de forma geral como funcionam as aplicações de notificações e com isto adaptar esses mesmos conhecimentos para o desenvolvimento e aprimoramento da nossa aplicação, de forma a padronizar e respeitar as técnicas de elaboração de aplicações relacionados aos processos de notificações.

**Método-Histórico:** "Promovido por Boas. Partindo do princípio de que as actuais formas de vida social, as instituições e os costumes têm origem no passado, é importante pesquisar suas raízes, para compreender sua natureza e função. Assim, o método histórico consiste em investigar acontecimentos, processos e instituições do passado para verificar a sua influência na sociedade de hoje, pois as instituições alcançaram sua forma actual através de alterações de suas partes componentes, ao longo do tempo, influenciadas pelo contexto cultural particular de cada época. Seu estudo, para uma melhor compreensão do papel que actualmente desempenham na sociedade.” (Marconi & Lakatos, 2003, p. 107).

Com este método entendeu-se melhor como tem sido feito o processo de notificação epidemiológica, quais os caminhos que se seguem, como são armazenadas estas informações, em diferentes níveis, podendo assim perspectivar uma aplicação que não fuja daquilo que é o processo até então utilizado e que tenha capacidade de se enquadrar a possíveis mudanças no seu desenrolar histórico.

“Técnica é um conjunto de preceitos ou processos de que se serve uma ciência ou arte; é a habilidade para usar esses preceitos ou normas, a parte prática. Toda ciência utiliza inúmeras técnicas na obtenção de seus propósitos.” (Marconi & Lakatos, 2003, p. 174).

Utilizou-se as seguintes técnicas de pesquisa.

**Investigação Documental: “**A característica da pesquisa documental é que a fonte de colecta de dados está restrita a documentos, escritos ou não, constituindo o que se denomina de fontes primárias. Estas podem ser feitas no momento em que o fa*c*to ou fenómeno ocorre, ou depois.” (Marconi & Lakatos, 2003).

Esta técnica foi importante na revisão bibiográfica, uma vez que o homem está sujeito a esquecer, mas se o que ele esqueceu foi registado então encontra-se com maior facilidade as respostas procuradas. Desta forma com a mesma técnica entendeu-se melhor como tem sido feito o processo de notificação, que e como preencher os formulários ***(ver anexos 1 e 2)***.

**Pesquisa Bibliográfica:** “A pesquisa bibliográfica, ou de fontes secundárias, abrange toda bibliografia já tornada pública em relação ao tema de estudo, desde publicações avulsas, boletins, jornais, revistas, livros, pesquisas, monografias, teses, material cartográfico etc., até meios de comunicação orais: rádio, gravações em fita magnética e audiovisuais: filmes e televisão. Sua finalidade é colocar o pesquisador em contacto directo com tudo o que foi escrito, dito ou filmado sobre determinado assunto, inclusive conferências seguidas de debates que tenham sido transcritos por alguma forma, quer publicadas, quer gravadas.” (Marconi & Lakatos, 2003, p. 183)

O tema em questão é rico em bibliografia por constituir um problema social e de saúde pública, tanto no país ou internacionalmente. Com esta técnica explorou-se mais e melhor a respeito do problema, verificando como funciona o processo e quais as regras estabelecidas pelas organizações internacionais no processo de notificação de casos suspeitos para notificação epidemiológica.

**Troca de ideia ou brainstorming:** O brainstorming **(“tempestade de ideias”, em tradução literal)** foi desenvolvido formalmente em 1957 por Osborn, o qual argumentou que este método técnica e ferramenta para inovação aumenta a qualidade e a quantidade das ideias geradas pelos membros do grupo (Paulus, Larey, Putman, Leggett, & Roland, 1996), **A técnica se baseia em princípios como foco em quantidade, ausência de críticas às ideias e combinação de ideias.** Nas decisões importantes a serem tomadas em relação ao processo de inovação se refere a quais abordagens utilizar ao longo do processo essas abordagens, aqui chamadas de métodos técnicas e ferramentas para inovação, incluem brainstorming, análise morfológica, grupo focal, teste de conceito, entre outros ( Buchele, Teza, Dandolini, & Souza, 2014). Uma das grandes vantagens que tem esta técnica é que na troca de ideia os participantes podem contribuir a melhorá-la ou a produzir uma versão totalmente nova.

A interface móvel desenvolvida consttui um módulo de um sistema de notificação epidemiológico, envolvendo por outras entidades trabalhando em outros módulos com a finalidade de tornar em um sistema completo e interligado, então, a troca de conhecimento com outros envolvidos no processo de investigação e desenvolvimento do tema em questão foi de grande importância.

O trabalho cujo tema é desenho de uma interface móvel para um sistema de alerta epidemiológico, com objectivo de melhorar o sistema de vigilância epidemiológica do país, está divido em três (3) capítulos como se descreve:

* **Capítulo I: Fundamentação teórica sobre vigilância epidemiológica;** neste capítulo apresenta-se análises teóricas e técnicas aplicadas no processo de vigilância epidemiológico podendo assim entender-se melhor o objecto a ser modelado.
* **Capítulo II: Descrição das ferramentas e tecnologias utilizadas para o desenvolvimento**; neste capítulo apresenta-se algumas ferramentas e tecnologias utilizadas no processo de desenho e implementação de aplicações móveis, suas vantagens e desvantagens assim como as escolhas feitas e porque optou-se por estas escolhas.
* **Capítulo III: Descrição do processo de desenvolvimento;** neste capítulo fez-se uma abordagem geral sobre o processo de desenvolvimento com maior destaque na modelação do negócio e do sistema desenvolvido.

Para além dos mencionados capítulos, integra também o trabalho outras secções como conclusões, recomendações, bibliografia, anexos e apêndices.

# **CAPÍTULO I: FUNDAMENTAÇÃO TEÓRICA SOBRE VIGILÂNCIA EPIDEMIOLÓGICA**

* 1. INTRODUÇÃO

“A vigilância é a actual recolha sistemática de dados de saúde, sua análise e interpretação. Ela inclui a divulgação oportuna dos dados obtidos, junto de quem deles precisa para actuar.” (Organização Mundial da Saúde, 2011, p. 5).

“A Região Africana da OMS enfrentou, na última década, epidemias de cólera, meningite meningocócica, febre tifóide e gripe, entre outras e, recentemente, epidemias pelo vírus Ébola. É sabido que os países com sistemas de vigilância frágeis ou sem sistemas de vigilância de base comunitária não conseguem detectar prontamente nem responder em tempo útil aos acontecimentos ou ameaças para a saúde pública Há uma necessidade de fortalecer a vigilância das doenças a todos os níveis, principalmente ao nível das comunidades.” (Organização Mundial da Saúde, 2015, p. 1)

A vigilância epidemiológica é um processo que visa dotar os especialistas de uma orientação técnica a fim de tomar decisões importantíssimas para a não propagação e cura de doenças consideradas ameaças para a saúde pública e não só numa população bem definida.

A vigilância epidemiológica constitui uma ferramenta importante para o planeamento e controlo das doenças para que as medidas interventivas sejam feitas de forma imediata.

* 1. OBJECTIVOS DA VIGILÂNCIA EPIDEMIOLÓGICA

Segundo a OMS (2012), a vigilância é definida como a recolha, análise, interpretação e divulgação de dados sistemáticas e contínuas para a acção da saúde pública. Procura:

* Identificar surtos que constituem riscos para a saúde pública, a fim de determinar uma acção imediata;
* Controlar melhor as doenças e perceber se tendem a melhorar ou piorar;
* Entender a real dimensão dos problemas das doenças de uma determinada região;
* Determinar as prioridades e eficácia no processo de controlo de doenças;
* Garantir a segurança na saúde pública, através da redução de riscos.
  1. PROCESSOS DA VIGILÂNCIA EPIDEMIOLÓGICA

Segundo a OMS (2011), a vigilância epidemiológica de doenças de interesse epidemiológico envolve a recolha, análise, interpretação e divulgação de dados.

* **Recolha de dados;** Em condições aprovadas as amostras são recolhidas para serem analisadas, nela deve incluir a data e hora da recolha, uma breve descrição de amostra, caso a fonte seja humana deve incluir o nome, idade e sexo da pessoa bem como as informações de colector de amostras.
* **Processamento de dados;** Nesta fase os dados recolhidos na fase anterior são compilados e integrados.
* **Análise e interpretação de dados;** Os dados são analisados por especialistas para obterem uma breve conclusão da gravidade da situação e preparação de formulários para a divulgação das mesmas, ou se necessário da notificação do caso.
* **Divulgação de informações;** Esta fase corresponde a publicação das informações para o público geral, interessado ou ligado directamente a este problema.
* **Notificação;** Uma vez que a aplicação desenvolvida tem o seu forte no processo de notificação para os casos de interesse epidemiológico, então esta fase constitui uma das mais importantes no processo de elaboração do projecto. O processo de notificação é entendido como o fase de circulação de informação desde as suas origens até as responsáveis e especialistas em epidemiologia, este processo deve ser feito continuamente e de forma regular.

A notificação em epidemiologia é activada por duas razões, uma quando houver suspeitas clínicas acerca dos sintomas apresentados por um doente e outra quando houver confirmação clínica ou laboratorial.

As doenças ou sintomas apresentado pelo paciente se carecerem de notificação epidemiológica então deve ser feita de acordo com as suas prioridades, para que se tome medidas apropriadas e antecipadas para contornar a situação.

Para os dois tipos de notificações (baseado em diagnósticos e baseado em laboratórios), a notificação procede-se como se descreve abaixo:

1. Uma vez que o paciente apresenta sintomas de interesse epidemiológico, os responsáveis pelo atendimento do mesmo, preenchem um formulário específico destinado a esse fim e notificar o caso, de acordo as prioridades de notificação, nesta devem conter informações como o hospital, laboratório ou posto médico, constam ainda informações do doente, e de quem notifica.
2. Após ter recebido a notificação a Direcção Nacional de Saúde Pública deve organizar uma equipa de especialistas em epidemiologia para que possam recolher os dados, examinar as pessoas expostas e implementar medidas de controlo.

O Ministério da Saúde (2013), afirma que dependendo de vários factores como a organização político-administrativo do país, meios de comunicação, sistema de informação sanitária, entre outros, as informações notificadas têm diversas fontes como:

* Unidades sanitárias
* Laboratórios
* Agentes comunitários
* Protecção civil e bombeiros
* Líderes das aldeias como sobas ou mesmo curandeiros
* Parteiras tradicionais
* Pontos fronteiriços
* Serviços veterinários
* Saúde ambiental
* Terapeutas tradicionais
* Outros

A aplicação desenvolvida inicialmente actua em duas fontes consideradas de maior relevância por constituírem fontes primárias, uma vez que a maioria dos casos de notificação epidemiológica registam-se nestas fontes.

* **Notificação de casos baseados em exame de laboratório**

Para que todo e qualquer sistema de vigilância epidemiológica funcione com alta qualidade, é importante que existam laboratórios em todos os níveis de serviço.

A vigilância de base laboratorial é a recolha, análise, interpretação e divulgação de dados sistemáticas e contínuas com base em infecções confirmadas em laboratório para a acção da saúde pública (Organização Mundial da Saúde, 2011, p. 5)**.**

* **Notificação de casos baseados em diagnóstico médico**

Esta notificação está baseada nas suspeitas dos médicos de acordo os diagnósticos por eles realizados durante uma sessão de consulta com os seus pacientes, apesar de não ser um caso confirmado como as de laboratório, estas notificações dos médicos não deixam de ser uma preocupação para a Direcção Nacional de Saúde Pública.

* 1. SITUAÇÃO ACTUAL DA VIGILÂNCIA EPIDEMIOLÓGICA EM ANGOLA

Apesar da melhoria significativa dos principais indicadores de saúde globais do País, Angola ainda tem uma elevada taxa de mortalidade materna, infantil e infanto-juvenil, alta incidência de doenças infecciosas e parasitárias com destaque para as grandes endemias, doenças respiratórias e doenças diarreicas, um nível de mal nutrição ainda elevado em menores de 5 anos, uma persistência de surtos de Cólera, Raiva e Sarampo, e um aumento exponencial das doenças crónicas não transmissíveis (DCNT), sinistralidade rodoviária e violência. As doenças transmissíveis, ainda são responsáveis por mais de 50% dos óbitos registados na população (Ministério da Saúde, 2012, p. 16).

A população estimada é de cerca de 24,3 milhões de habitantes1, registando-se uma elevada taxa de crescimento anual da população, de cerca de 3,0%. A população está distribuída em 55% em zonas urbanas e periurbanas e 45% em zonas rurais. Regista-se um movimento migratório interno de cerca de 20% das zonas rurais para as áreas periurbanas (OMS, 2016, p. 3).

Actualmente em Angola o processo de notificação de doenças de interesse epidemiológico tem sido feito com uma certa dificuldade, principalmente a relativa so atraso da chegada de informações ao Ministério da Saúde. Mas, o país conta com um sistema muito forte no que diz respeito a prevenção de surtos, bem como está munido de profissionais capazes de lidar com qualquer situação que constitui ameaça à saúde pública.

Segundo a OMS (2016), uma das políticas adoptadas para reforçar a vigilância epidemiológica no país, é a elaboração e distribuição de boletins epidemiológicos (semanais, trimestrais e anuais).

* 1. CONCLUSÃO

Uma notificação eficaz envolve uma circulação de informação atempada, contínua e regular acerca da ocorrência de casos. Neste capítulo apresentou-se os aportes teóricos e práticos envolvidos no processo de vigilância epidemiológica, analisou-se os processos envolvidos, os objectivos e a situação actual do sistema de vigilância em Angola. Constatou-se que existem políticas envolvidas no processos de notificação de casos de interesse epidemiológico, políticas estas que necessitam de ferramentas informáticas no sentido de tornar o sistema mais eficaz e seguro. Pode-se ainda definir as fontes de notificação baseados em exames de laboratório e diagnóstico médico, pois constituem fontes fidedignas para o nivel central.

# **CAPÍTULO II: DESCRIÇÃO DAS FERRAMENTAS E TECNOLOGIAS UTILIZADAS PARA O DESENVOLVIMENTO**

* 1. **INTRODUÇÃO**

No presente capítulo aborda-se sobre as tecnologias e ferramentas possíveis para a programação para dispositivos móveis assim como as ferramentas escolhidas e as justificações da escolha de cada uma das ferramentas utilizadas para o desenho e implementação da interface móvel para um sistema de notificação epidemiológica.

* 1. **METODOLOGIAS DE DESENVOLVIMENTO**

A metodologia é o conjunto de procedimentos e técnicas que são usadas, normalmente já estabelecidas por experiência anterior, pela academia ou pela instituição a qual estamos ligados. (Pereira, 2011, p. 11).

A metodologia de desenvolvimento escolhida foi o RUP, uma vez que se trata de uma metodologia híbrida com processos simples, e resumidas apenas em 4 fazes de desenvolvimento. Além disto desde o início nesta carreira propriamente no processo de Engenharia de *Software*, tem sido o RUP a metodologia de elaboração dos projectos, então isto torna-se vantagem por ser uma metodologia já conhecida.

Abaixo apresenta-se duas metodologias com caminhos totalmente diferentes, mas com o mesmo objectivo.

***Rational Unified Process* (RUP)**, baseado no modelo *Unified Process* (UP) e especificado pela empresa *Rational*, que foi adquirida há algum tempo pela IBM. O RUP é, na realidade, uma base de conhecimento composta de mais de 3.200 modelos de artefactos (planilhas, arquivos de texto etc.) que podem ser adaptados para a organização e para o projecto. Possui pontos de controlo (ou marcos) muito bem definidos, servindo como um guia bastante completo para a equipa e para os gestores de projecto (Pereira, 2011, p. 13).

***Extreme Programming* (XP)**, um dos métodos ditos ágeis, sendo bastante usado actualmente, pois é "leve" com respeito à documentação externa produzida, desonerando a equipe desse trabalho e focando na produção de código que atenda o mais rapidamente aos usuários. Não é exactamente um modelo de processos, mas sim uma colecção de práticas a serem adoptadas pela equipa de desenvolvimento. Por esta razão vem sendo adoptado actualmente em conjunto com modelos de *gestão* de projectos, *como* o SCRUM (Pereira, 2011, p. 14).

* 1. **FERRAMENTAS CASE**

Para a modelagem de sistemas utilizando a UML, usualmente emprega-se Ferramentas CASE (*Computer-Aided Software Engineering* – engenharia de *software*auxiliada por computador), apelidadas simplesmente de CASE (Pereira, 2011, p. 3).

Actualmente existem várias ferramentas CASE disponíveis, sendo cada uma com características particulares, mas com os mesmos objectivos.

A ferramenta CASE escolhida para a modelagem do sistema foi a Visual Paradigm for UML na versão 10.0 por ser uma ferramenta fácil de utilizar e muito dinâmica na construção dos seus diagramas com, uma variedade de diagramas disponíveis. Além disso o brModelo foi também a ferramenta CASE escolhida para o desenho e modelagem da base de dados.

* + 1. **VISUAL PARADIGM**

O Visual Paradigm é um provedor líder e globalmente reconhecido para soluções de *software* negócios e sua transformação. Ele permite que as organizações aprimorem uma agilidade de negócios e de TI e promovam uma inovação por meio de padrões abertos populares (visual paradigm, 2018).

O Visual Paradigm oferece uma simplicidade e recursos do mais alto nível para modelagem de sistemas, motivo pelo qual grandes empresas usam esta ferramenta para a modelagem dos seus sistemas.

### **STARUML**

Para StarUML (2018), o StarUML é um sofisticado modelador de *software* destinado a suportar modelagem ágil e concisa.

Os principais alvos dos usuários são:

* Equipes de desenvolvimento ágeis e pequenas
* Pessoas profissionais
* Institutos educacionais.

O StarUML é compatível com sistemas operativos como *Windows*, *Linux* ou *Mac OS*, além de suportar vários diagramas, exportação e muito mais.

### **BRMODELO**

Em 2005 foi desenvolvida uma ferramenta de código aberto e totalmente gratuita voltada para ensino de modelagem de base de dados relacionais com base na metodologia defendida por Carlos A. Heuser no livro “Projeto de Base de Dados” (sis4, 2017).

A repercurssão que teve o brModelo levou ao desenvolvimento de novas versões com melhorias significativas, e nos últimos tempos o brModelo tem sido utilizado por diversas universidades e centros de formações para modelagem de bases de dados relacionais.

* 1. **SISTEMAS OPERATIVOS PARA DISPOSITIVOS MÓVEIS**

Um sistema operativo é a base de qualquer aplicação, pois é por intermédio dela que estas aplicações se executam.

Existem inúmeros sistemas operativos para dispositivos móveis, mas muitos desde já não fabricam mais seus aplicativos ou novas actualizações do *software* pois a concorrência é muito apertada.

Escolhe-se o Android como sistema padrão na qual a aplicação inicialmente vai rodar ou funcionar, isto por ser o sistema operativo *mobile* que domina o actual mercado.

Desde o surgimento dos telemóveis até à presente data vários são os sistemas operativos para telemóveis lançados por várias empresas apesar de que actualmente o mercado é dominado pelos sistemas operativos Android e Mac OS X, existem outros sistemas como ilustra a imagem a figura abaixo com as respectivas descrições.



*Figura 1-Principais sistemas operativos para celulares.*

### **ANDROID**

O Android é o sistema operativo criado pela Google. É um sistema aberto, acessível a todos. Ele foi baseado nun núcleo Linux e suporta qualquer tipo de conexão sem fio - 3G, EDGE, Wi-Fi e Bluetooth. Ele é compatível com quase tudo tratando-se de multimídia. Com a programação aberta, ele pode ser alterado, adaptar-se e manter um baixo custo. Ele já conta com inúmeros aplicativos para personalizar o celular (Spadari, 2017).

É o sistema operativo móvel mais popular do mundo, funciona em telemóveis, tabletes, televisores, carros, relógios e muito mais. Actualmente na versão 9 (Pie) lançado no segundo semestre de 2018, foi desenvolvido com Inteliêngia Artificial (IA) para tornar o *smartphone* mais inteligente, rápido e personalizado.

### **iOS: IPHONE**

O sistema operativo que permite aos *iPhones* e *iPads* rodarem no *Mac* OS X modificado é chamado de *iPhone*. Este foi o primeiro sistema operativo criado para telefones móveis e oferece suporte para as tecnologias de toque múltiplas adaptadas à multimídia (vídeos, imagens e músicas) (Spadari, 2017).

Acredita-se que as razões que levam o *iPhone* ser menos popular que o Android é a dificuldade de partilha de ficheiros directo com outros aplicativos que não sejam da *Apple*, mas que ao mesmo tempo torna-se vantagem pois evita que seja alvo de vírus, e o custo de aquisição dos aparelhos que suportam este sistema operativo.

### **WINDOWS PHONE**

O *Windows* *Phone* é um sistema operativo para telefones móveis baseados no *Kernel* do *Windows* CE6. Ele foi usado primeiramente nos *Pocket* PCs. Ele usa o mesmo padrão de aplicativos usados na versão para computador, mas requer um *hardware* potente para funcionar correctamente. Este sistema integra e é compatível com todos as aplicações de escritório da versão PC, como *Word, Excel, Power Point, Windows* *Media* *Player*, entre outros (Spadari, 2017).

Nos últimos tempos o *Windows Phone* tem caído bastante, e se continuar assim pode seguir o rumo de outros sistemas operativos que caíram em desuso.

### **BLACKBERRY**

O *BlackBerry* é um sistema operativo desenvolvido pela empresa canadense RIM (*Research In Motion*). Ele integra diversas funções importantes, que foram integradas pela primeira vez nos celulares chamados *smartphones*. Ele tem editor de texto, acesso à *internet*, *e-mail* e a tecnologia IPv6. O que o diferencia dos demais é que ele utiliza um serviço próprio de e-mail RIM, chamado BBM. As mensagens e *e-mails* no envio e recepção chegam até a 200 kbps, utilizando a tecnologia EDGE. Ele suporta todas as funções de escritório necessárias para criar documentos, planilhas, apresentações, etc. (Spadari, 2017).

Tanto a arquitectura como o próprio sistema operativo, é bastante potente, mas que aos poucos também não se tem mantido forte no mercado, num momento em que as principais concorrentes não param de lançar novas versões.

### **SYMBIAN**

O sistema *Symbian*surgiu da parceria de um grupo das fabricantes Nokia, Ericsson, *Motorola e Panasonic*. É um sistema directo, sem muitos detalhes, cujo maior objectivo é a rapidez e a acessibilidade. A Nokia também licencia o *Symbian* para empresas que não pertençam ao grupo.

O sistema *Symbian* é um sistema aberto e de baixo custo, possui recursos para gerir e utilizar pouca bateria e permite a instalação de outros *software*s, ao contrário do iPhone (Spadari, 2017).

A plataforma *Symbian* continuará a evoluir com a Nokia, que começou a disponibilizar o desenvolvimento futuro da plataforma por meio de um modelo alternativo directo e aberto.

Além deste existem outros como o *Firefox* Os, *Palm Webos MeeGo* e o *Brew MP.*

* 1. **PORQUÊ A ESCOLHA DO ANDROID?**

*Android* é uma plataforma completamente livre e de código aberto (*open source)* desenhada para dispositivos móveis desenvolvida pela *Google* e actualmente é mantida pela *Open Handset Alliance* (OHA) (Lacheta, 2016).

Actualmente o *Android* é o sistema operativo móvel mais utilizado no mundo, e está disponível para diversas plataformas, como *smartphones* e *tablets*, TV (*Google* TV), relógios (*Android Wear*), óculos (*Google Glass*), carros (*Android Auto*) (Lacheta, 2016, p. 18).

Ela utiliza a linguagem Java que é uma linguagem bem familiarizada, podendo assim facilitar na programação das aplicações ou desta aplicação em particular.

A *Google* continua reinando soberana no mundo *mobile*, aponta o último relatório da *International Data Comparation* (IDC), Comparação Internacional de Dados em Português, sobre o primeiro trimestre de 2017 (tecmundo, 2017).

A tabela abaixo representa os relatórios das pesquisas feitas pela IDC até ao primeiro trimestre de 2017 na comparação dos sistemas operativos móveis.

*Tabela 1: Comparação dos sistemas operativos móveis 1º trimestre de 2016 à 1º trimestre de 2017*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Período | Android | iOS | Windows phone | Outros |
| 1º Trimestre/2016 | 83,4% | 15,4% | 0,8% | 0,4% |
| 2º Trimestre/2016 | 87,6% | 11,7% | 0,4% | 0,3% |
| 3º Trimestre/2016 | 86,8% | 12,5% | 0,3% | 0,4% |
| 4º Trimestre/2016 | 81,4% | 18,2% | 0,2% | 0,2% |
| 1º Trimestre/2017 | 85,0% | 14,7% | 0,1% | 0,1% |

*Fonte: IDC, lida em tecmundo (2017).*

Como se pode observar, o Android é sem sombra de dúvidas o sistema operativo para dispositivos móveis mais utilizado no mundo, com uma percentagem superior a 80%, isto devido a vários factores como por exemplo a auto adaptação dos mesmos ao bolso de cada um, e a existência de diferentes modelos para diferentes gostos.

1. 1. 1. **CARACTERÍSTICAS DO ANDROID**

Sendo o Android o sistema operativo mais utilizado no mundo para aplicativos móveis, abaixo apresenta-se algumas caracteristicas deste sistema opertaivo

* Interfaces magníficas
* Código fonte aberto e gratuito
* Utiliza linguagem Java ou Kotlin para *activity* e XML para *layout*
* Multitarefa
* Compatibilidade cruzada.
  1. **OPEN HANDSET ALLIANCE (OHA)**

A *Open Handset Alliance* (OHA) é um grupo formado por gigantes do mercado de telefonia móvel liderados pelo *Google*, cujo objetivo do grupo é definir uma plataforma única e aberta para telemóveis além disso, essa aliança permite criar uma plataforma moderna e flexível para o desenvolvimento de aplicações corporativas (Lacheta, 2016).

Destacam-se alguns integrantes deste grupo com nomes consagrados como Intel, HTC, LG, *Motorola, Samsung, Sony Ericsson, Toshiba, Huawei, Sprint Nextel, China Mobile, T-Mobile, ASUS, Acer, Dell, Garmin* e muito mais, tornando assim o *Android* como líder mundial da tecnologia móvel.

Com o avanço da tecnologia móvel a nível mundial, a OHA tem como objectivo principal manter uma plataforma-padrão na qual todas as novas tendências do mercado estejam englobadas em uma única solução (Lacheta, 2016). Este facto transforma-se em grande vantagem para os fabricantes de telemóveis, além disso, por ser uma plataforma de código aberto permite com que os fabricantes possam fazer alterações no código fonte de forma a melhorar a qualidade dos seus produtos sem precisar compartilhar as alterações com ninguém, pois assim garante a licença Apache Software Foundation (ASF).

* 1. **LINGUAGENS DE PROGRAMAÇÃO**

Dependendo do sistema operativo que o dispositivo móvel poderá executar, as linguagens de programação variam em relação aos mesmos.

* Java (Android, Blackberry)
* Kotlin (Android)
* Objective C (iOS)
* C/C++ (Sybiam)
* C# (Windows phone)
* Python (Sybiam)

Uma vez que a aplicação a desenvolver é uma aplicação que funciona em aplicativos com sistema operativo Android, escolheu-se a linguagem Java como a linguagem padrão da programação, pelo domínio próprio, simplicidade e robustez, uma segunda opção seria o *Kotlin* uma vez que a *Google* oficializou a um tempo pra cá que o *Kotlin* é também uma linguagem oficial de desenvolvimento Android, mas pelas suas desvantagens como necessidade de memória, e tempo de compilação escolheu-se a linguagem Java. Além disto utilizou-se também o XML e PHP que foi muito útil para a sincronização com a aplicação Web, para facilitar no processo de notificação quer imediata ou de rotina.

Abaixo detalha-se as principais linguagens para o desenvolvimento *mobile*, incluido a linguagem Java que foi a linguagem escolhida para o desenvolvimento da interface móvel para o sistema de alerta epidemiológica.

* + 1. **JAVA**

Segundo Cordeiro (2017), o Java foi inicialmente desenvolvido pela *Sun Microsystems*em 1995, e é usado para uma ampla gama de *softwares*. O código Java é executado por uma máquina virtual, que é executada em dispositivos Android e interpreta o código. O Java é uma linguagem de programação capaz de criar tanto aplicativos para desktop, aplicações comerciais, softwares robustos, completos e independentes, aplicativos para a Web. Além disso, caracteriza-se por ser muito parecida com C++.

Dentre as inúmeras características do Java destacam-se:

* Simples e familiar;
* Orientada a Objetos;
* Compilada e Interpretada;
* Pronta para Redes;
* Distribuído;
* Multiprocessamento (Multithread);
* Portabilidade;
* Segura.

O Java é uma linguagem de programação orientada a objectos com vários recursos como construtores, execuções não verificadas, execuções verificadas e muito mais. É a linguagem mais suportada pela *Google* e o que a maioria das aplicações da *Play Store* são criadas.

* + 1. **KOTLIN**

Segundo Cordeiro (2017), o Kotlin é uma linguagem estaticamente tipada, desenvolvida pela *JetBrains*, cuja sintaxe é mais expressiva e concisa do que a do Java. Com recursos como expressões *lambda*, sobrecarga de operadores, *templates* de *strings* e muito mais.

Como o Java e o Kotlin são altamente interoperáveis, elas podem ser usadas juntas no mesmo projecto. Inclusive, o Kotlin tem como base, e é executado na Maquina Virtual do próprio Java, mas apesar disto o Kotlin tem algumas desvantagens que devem ser consideradas:

* Necessidade de memória, isto porque o Kotlintem sua própria biblioteca que é adicionada no seu aplicativo.
* O tempo de compilação de códigos Kotlin, é um pouco mais lento usando o *Gradle*, do que em Java.
  + 1. **C/C++**

O Android Studio oferece suporte para C/C++ usando o *Android Native Development Kit.* Com isto não precisamos escrever os códigos que não são executados na *Java Virtual Machine*, mas sim executados nativamente no dispositivo e oferecem mais controlo sobre coisas como memória (Cordeiro, 2017). A grande desvantagem é a complexidade de implementação e o crescente número de erros que pode dar pela importação das bibliotecas escritas em C ou C++.

* + 1. **XML**

A linguagem *eXtensible Markup Language* (XML) foi sendo progressivamente introduzida sobretudo a partir da década de 90 do século passado e é, actualmente uma das linguagens que permite a criação de documentos com bastante flexibilidade ao nível da manipulação de dados, uma vez que não é dependente do *hardware* ou *software* utilizado (Tavares, 2016).

XML armazena dados no formato de texto simples. Isto fornece uma maneira independente de *software* e *hardware* de armazenar, transportar e compartilhar dados. O XML também facilita a expansão ou a actualização para novos sistemas operativos, novos aplicativos ou novos navegadores, sem perda de dados (w3school, 2018), o XML dá a possibilidade de editar o documento em qualquer editor de texto.

* + 1. **PHP**

*Hypertext Preprocessor (*PHP) É uma linguagem de *script*, em grande parte usada no lado do servidor. Actualmente a linguagem PHP é um projecto da *Apache Software Foundation* (Tavares, 2016). A tecnologia PHP incorpora a linguagem PHP que é baseada nas linguagens C, Java e Perl e ainda pode ser vista como uma combinação de linguagem de programação e servidores de aplicações.

O PHP pode interagir com diferentes ferramentas de base de dados, tais como, MySQL, SQLite, PostgreSQL, ORACLE, etc., tornando o conteúdo criado cada vez mais dinâmico.

* + 2. **JSON**

JSON (JavaScript Object Notation) é um formato leve de intercâmbio de dados. JSON é um formato de texto que é completamente independente da linguagem, mas usa convenções familiares aos programadores da família C de linguagens, incluindo C, C ++, C #, Java, JavaScript, Perl, Python e muitos outros. Essas propriedades tornam o JSON uma linguagem ideal de intercâmbio de dados (Crockford, 2000), usa sintaxe JavaScript, mas o formato JSON é apenas texto. O texto pode ser lido e usado como formato de dados por qualquer linguagem de programação (w3school, 2018)

* 1. **SISTEMAS GESTORES DE BASE DE DADOS**

Um servidor é um sistema de computação centralizado que fornece serviços a uma rede de computadores. As redes que utilizam servidores estão baseadas geralmente na arquitectura Cliente/Servidor. Um sistema gestor de base de dados(SGBD)é o *software* que permite criar, manter e manipular bases de dados para diversas aplicações. Fornece uma interface que permite que o utilizador se comunique com os dados armazenados fisicamente na base de dados sem se preocupar com o formato dos dados, como e onde estiverem armazenados (Damas, 2005). Em casos de falhas ou acidentes, o SGBD é o responsável pela recuperação, validação e segurança dos dados. Existem vários sistemas gestores de base de dados, abaixo fez-se menção de alguns.

Por ser uma aplicação que vai trabalhar a par da aplicação Web, escolheu-se o SQLite como servidor de base de dados, que é um servidor muito útil para aplicações que desejam armazenar poucas quantidades de informação e como a aplicação está voltada somente para guardar e enviar os casos confirmados ou suspeitos de epidemia, o SQLite é sem sombra de dúvidas a melhor escolha para a gestão dos dados locais, além disso ela utiliza a linquagem SQL que é a mesma utilizada para a plataforma web.

* + 1. **SQLITE**

Segundo o *site* oficial o SQLite é uma biblioteca auto-suficiente , sem servidor, sem configuração, transacional de base de dados SQL. O código para o SQLite é livre e disponível para o uso de qualquer finalidade. O SQLite é uma das bases de dados mais amplamente implantadas no mundo, com diversos aplicações de diferentes níveis.

SQLite não é uma biblioteca cliente usada para conectar com um grande servidor de base de dados, mas sim o próprio servidor de base de dados. O SQLite é um mecanismo de base de dados SQL incorporado. Ao contrário da maioria dos outros bases de dados SQL, o SQLite não possui um processo de servidor separado. O SQLite lê e grava diretamente em arquivos de disco comuns (SQLite, s.d.).

* + 1. **MYSQL**

É um dos mais populares programas de administração de bases de dados que existe.

O MySQL é um sistema de gestão de base de dados relacionais, suporta SQL, de código aberto e para utilização profissional. É o mais conhecido a nível mundial e suporta multiprocestamento e multiplataforma (Neves & Ruas, 2005).

Uma das características do MySQL, é o suporte prático a qualquer plataforma actual, o que torna útil para os aplicativos móveis, por exigir poucos recursos de *hardware*.

* + 1. **FIREBASE REALTIME DATABASE**

Todos os dados do *Firebase Realtime Database* são armazenados como objectos JSON. Pense no base de dados como uma árvore JSON hospedada na nuvem. Ao contrário de um base de dados SQL, não há tabelas nem registos (Google, 2018).

[Armazenamento de documentos em](https://db-engines.com/en/article/Document+Stores) tempo real hospedado na nuvem . Os clientes iOS, Android e JavaScript compartilham uma instância do Realtime Database e recebem automaticamente actualizações com os dados mais recentes. A base de dados principal é a loja de documentos, isto quer dizer que não é possível o uso de chaves estrangeiras pois não se trata de um SGBD relacional. O *Firebase Realtime Database* é de licença comercial e está baseado em nuvem o que pode se tornar um constrangimento para quem não tem acesso a *internet*.

* 1. **AMBIENTE DE DESENVOLVIMENTO (IDE) PARA ANDROID**

*Integrated Development Environment* (IDE), é uma aplicação informática que proporciona serviços integrais para facilitar ao programador no desenvolvimento de *software*. Nela encontramos um editor de código e um depurador.

Com tantos IDE disponíveis para o desenvolvimento de aplicações *Android*, uma das coisas mais difíceis é escolher em qual trabalhar. Depois de bem analisadas, para o desenvolvimento da aplicação escolhe-se o Android Studio na versão 3.0, porque este nos oferece inúmeros recursos e está dirigido somente ao desenvolvimento de aplicações Android de forma unificada na linguagem Java que é uma linguagem familiar, tem um sistema de compilação mais flexível e *emulador* mais rápido além de suportar C++ e Kotlin.

Como ambiente de desenvolvimento optou-se pelo Android Studio na versão 3.0, é trabalhoso no momento de instalação e configuração, mas é sem sombra de dúvidas o melhor IDE para desenho e desenvolvimento de aplicações móveis.

Abaixo descreve-se alguns IDE possíveis para a programação de aplicações que rodam em sistemas operativos *Android*.

* + 1. **ANDROID STUDIO**

Android Studio é um ambiente de desenvolvimento que utiliza as linguagens Java, Kotlin e C++.

O Android Studio é até então o IDE mais utilizado para desenvolvimento de aplicações móveis (Android). Uma das desvantagens associada ao mesmo é a elevada necessidade de recursos de *hardware*, porém o Android Studio é uma ferramenta muito eficaz, compacta e segura para o desenvolvimento de aplicações móveis. O Android Studio é um ambiente de desenvolvimento integrado para desenvolver para a plataforma Android. Foi anunciado em 16 de Maio de 2013 na conferência Google I/O.

Além deste existem tantos outros que podem ser usados para o desenvolvimento de aplicações compatíveis com o Android como as que se seguem.

* + 1. **ECLIPSE**

O Eclipse é um ambiente de desenvolvimento que utiliza a linguagem Java. Antes do surgimento do Android Studio, o Eclipse era a opção recomendada pela *Microsoft* para o desenvolvimento de aplicações android, mas que ultimamente perdeu peso para o desenvolvimento de aplicações móveis em relação ao android studio isto porque ela não está dirigida somente para desenvolvimento de aplicações móveis, mas para tantas outras plataformas e linguagens o que fornece um risco de se cometerem erros e torna o processo mais lento.

* + 1. **XAMARIM**

Xamarim é um ambiente de desenvolvimento baseado na linguagem C#, construída pela *Microsoft* e incluído no *Visual Studio*. Apesar de não ter uma linguagem propriamente android, a grande vantagem que o Xamarim nos oferece é a possibilidade de a partir do mesmo código podemos obter produto para iOS e *Windows phone.*

Os IDEs acima supracitados são ambientes de desenvolvimento com Java por isto têm que garantir tanto o *Java Development Kit (*JDK) e o *Java Runtime Edition (*JRE) das versões compatíveis estejam instaladas. Além deste existem tantos outros como o AIDE (Android IDE), o Python, etc.

* 1. **ANDROID SDK TOOLS**

*Android Software Development Kit* (Kit de Desenvolvimento de *Software* para *Android*) ou simplesmente SDK é um conjunto de ferramentas de desenvolvimento aplicações, compatíveis com *Windows*, *Linux* e *Mac iOS.*

No pacote do SDK temos o emulador que nos permite executar as aplicações de uma forma simulada nos computadores, independente do sistema operativo que estiver a usar, bastando apenas configurar antes o *Android Virtual Device* (AVD). Mas o processo de execução virtual dos códigos é muito lento, por isto é bem mais eficaz testar as aplicações directamente num dispositivo físico.

Oficialmente a Google indica que, os aplicativos Android podem ser escritos com as linguagens [Kotlin](https://www.androidpro.com.br/blog/kotlin/kotlin/), [Java](https://www.androidpro.com.br/blog/java/programacao-android-java-basico-parte-01/) e C++ usando o [Android SDK](https://www.androidpro.com.br/blog/android-studio/android-sdk/), mas além destas com outras também é possível, podendo por tanto ser mais dispendioso e trabalhoso.

* 1. **WEB SERVICE**

Um *Web service* é um conjunto de métodos acedidos e invocados por outros programas utilizando tecnologias *Web*. Ela é utilizado para transferir dados através de protocolos de comunicação para diferentes aplicações e dispositivos, independentemente das linguagens de programação utilizadas em cada plataforma com suporte *web* (Opensoft, 2016).

Para que exista comunicação entre aplicações desenvolvidas em linguagens diferentes utilizando a tecnologia *Web service,* é necessário uma linguagem intermédia que garanta a comunicação entre a linguagem do *Web service* e o sistema que faz o pedido ao Web service. Para tal, existem protocolos de comunicação como o *Simple Object Access Protocol* (SOAP) e o *Representational State Transfer* (REST). O protocolo SOAP utiliza XML para enviar mensagens e, geralmente, serve-se do protocolo HTTP para transportar os dados, enquanto que o REST baseia-se no protocolo HTTP e permite utilizar vários formatos para representação de dados, como  *JavaScript Object Notation* (JSON) que é um dos mais utilizados, XML, RSS, entre outros. (Opensoft, 2016).

*Web service* foi a tecnologia escolhida para a interacção de informações e sistemas, uma vez que este necessita apenas de tecnologias XML/JSON e protocolo HTTP.

Com isto, garante-se uma maior segurança dos dados, e redução significativa dos custos uma vez que, com a implementação do *Web service*, a aplicação é capaz de funcionar tanto *online* como *offline*. Quando o usuário estiver *online* a comunicação é feita directamente com a aplicação *Web*, e quando este estiver *offline* os dados são armazenados no seu dispositivo móvel, uma vez que o utilizador tenha acesso a *internet*, a sincronização é activada e os dados são enviados.

* 1. **DISTRIBUIÇÃO DAS APLICAÇÕES**

Quando executamos o código Java da aplicação este cria de forma automática um ficheiro com extensão ‘.apk’ este ficheiro corresponde ao *setup* da aplicação pode ser enviado aos dispositivos nos quais se deseja instalar a aplicação ou então colocar a aplicação em lojas virtuais como a *Google Play Store* ou ainda podem ser instalados no momento da fabricação dos aparelhos.

* 1. **CONCLUSÃO**

As ferramentas de desenvolvimento de *software* não são totalmente exploradas do ponto de vista funcional, seja por complexidade do assunto abordado ou pela usabilidade. Neste capítulo fez-se uma abordagem a respeito das tecnologias móveis no que concerne as seus sistemas operativos, às ferramentas necessárias para desenvolvimento de aplicações ligadas as aplicações móveis. Falou-se também da *Web service* queconstitui uma tecnologia importante para o desenvolvimento de sistemas que partilham dados.

# **CAPÍTULO III: DESCRIÇÃO DO PROCESSO DE DESENVOLVIMENTO**



## INTRODUÇÃO

“A engenharia de *software* tem por objectivo apoiar o desenvolvimento profissional de *software*, mais do que a programação individual. Ela inclui técnicas que apoiam especificação, projecto e evolução de programas, que normalmente não são relevantes para o desenvolvimento de *software* pessoal” (Sommerville, 2011, p. 3).

Durante um turno de 12 horas, uma enfermeira é capaz de andar entre seis e oito quilómetros pelo hospital. Com um dispositivo móvel em mãos, suas caminhadas poderiam ser reduzidas drasticamente, e, com isto, ela ficaria mais tempo com os pacientes. Mas o quanto computadores e impressoras portáteis, *tablets* e *scanners*, por exemplo, ajudariam esses profissionais? (Butcher, 2018)

Uma aplicação que funcione em dispositivos móveis com objectivo de melhorar ou tornar digital o sistema de alerta epidemiológica seria muito útil não só para os que actuam directamente na área mas para a população em geral.

## DESCRIÇÃO DO NEGÓCIO

A saúde é um bem importantíssimo para a humanidade. Evitar ao máximo que uma dada doença que constitui uma ameaça à saúde pública, potencial epidémico, erradicação ou mesmo de caracter internacional esteja controlada e previnida é o objectivo dos especialistas em saúde pública em representação do Ministério da Saúde. Para tal existe um grande número de profissionais especializados em vigilância epidemiológica espalhados por todo o país que garantem a não propagação das doenças de interesse epidemiológico e o prévio tratamento das mesmas, evitando as elevatas taxas de mortalidade e garantindo saúde e bem estar às comunidades.

As unidades sanitárias (Hospital, centro médico, clínica ou laboratório) antendem diariamente vários pacientes dependendo da capacidade e procura de cada uma. Uma vez que um paciente sinta-se incomodado, dirige-se a unidade sanitária para fazer consulta médica ou exame de laboratório. Durante a sessão de consulta, se os sintomas e sinais apresentados pelo paciente for suspeita de uma caso de interesse epidemiológico, o médico preenche a ficha de notificação e entrega ao especialista em vigilância epidemiológica para notificar o caso. Além disso, um outro profissional de saúde( técnico de laboratório) é responsável pela notificação de caso confirmado em laboratório, quando o resultado do exame seja positivo para casos de interesse de notificação epidemiológica. O processo acontece da mesma forma diferenciando em alguns aspectos nos formulários de notificação *(ver anexos 1 e 2)*. O profissional de saúde que notifica o caso, deve registar o mesmo caso no livro de resgisto da unidade sanitária em que este se encontra, para que a mesma unidade sanitária, tenha o controlo a nível local.

Para que estas informações cheguem ao Ministério da Saúde passa por diversos níveis, isto é, as informações são levantadas entregues as delegações municipais, estes por sua vez são entregues as direcções provinciais que são os responsáveis em encaminhar para a Direcção Nacional de Saúde Pública. Com estas informações o Ministério da Saúde tem um maior controlo das doenças e pode adoptar métodos preventivos para melhorar a saúde das comunidades.

## PROCESSOS DO NEGÓCIO

Um processo é um conjunto de actividades ou séries de passos que o negócio executa para entregar um produto ou efectuar um serviço.

A seguir apresenta-se os processos extraídos do negócio acima descrito.

* Fazer consulta médica
* Fazer exame de laboratório
* Notificar caso

## REGRAS DO NEGÓCIO

As Regras de Negócios são declarações sobre políticas ou condições que devem ser satisfeitas (Rational , 2001); estas leis é que formam o estilo do funcionamento das empresas.

Na tabela a seguir apresenta-se as principais regras tiradas do negócio descrito acima bem como uma breve justificação das mesmas mesmas.

*Tabela 2: Regras do negócio*

|  |  |  |
| --- | --- | --- |
| Nº | Regra | Justificação |
| 1 | Notificar casos suspeitos | Sempre que um médico ou técnico de laboratório detetar caso suspeito ou confirmado, este deve notificar ao especialista em epidemiologia responsável pela unidade sanitária. |
| 2 | Preencher dados relevantes | Ao notificar um caso tem dados que são relevantes para levar a cabo á investigação do caso, dados como a unidade sanitária, o profissional de saúde e a data de notificação. |

## ACTORES DO NEGÓCIO

Segundo Pereira (2011); O termo actor donegóciose refere ao papel que alguém ou alguma coisa interpreta enquanto interage com o negócio sendo modelado.

Na tabela abaixo apresenta-se os actores extraidos do negócio com as respectivas justificações.

*Tabela 3: Actores do negócio*

|  |  |
| --- | --- |
| Actor | Justificação |
| Paciente | É a pessoa, que se sente incomodada e assim sendo contacta o centro hospitalar para marcar consulta e é a partir deste que os processos são activados. O paciente é praticamente o centro de tudo. |
| Médico | É a pessoa que atende os pacientes em suas consultas e notifica casos suspeitos de acordo com os diagnósticos observados. |
| Técnico de laboratório | É pessoa responsável pelos exames médicos e notificação de casos confirmados a partir dos resultados obtidos. |

## TRABALHADORES DO NEGÓCIO

Trabalhador de negócio é uma classe que representa uma abstração de uma pessoa que actua no negócio (Rational , 2001).

A tabela abaixo apresenta os trabalhadores do negócio e as possíveis justificações.

*Tabela 4: Trabalhadores do negócio*

|  |  |
| --- | --- |
| Trabalhador | Justificação |
| Agente Epidemiológico | Interage com os técnicos de laboratório e médicos no processo de notificação de casos de interesse epidemiológico. |
| Médico | É o responsável por garantir que o caso de uso fazer consultas médicas se execute. |
| Técnico de laboratório | É a pessoa responsável por garantir que o caso de uso fazer exames de laboratório se execute. |
| Recepcionista | Interage e liga os pacientes aos profissionais de saúde, informa e recolhe os dados do paciente quando este desejar fazer consulta ou exame de laboratório. |

## CASOS DE USO DO NEGÓCIO

Os casos de uso já se tornaram uma característica fundamental da linguagem de modelagem unificada (UML — do *inglês unified modeling language*). Um caso de uso identifica os actoresenvolvidos numa interacção e dá nome ao tipo de interacção. Esta é, então, suplementada por informações adicionaisque descrevem a interacção com o sistema. A informação adicional pode ser uma descrição textual ou umou mais modelos gráficos, como diagramas de sequência ou de estados da UML (Sommerville, 2011, p. 74).

Existem muitas coisas em comum entre casos de uso e processos de negócio, tanto que, um processo pode ser um caso de caso, mas um caso de uso não pode ser um processo.

Os casos de uso são analisados em dois pontos distintos que é o negócio e sistema, com os mesmos objectivos, mas que precisa-se de muita atenção para não confundí-los.

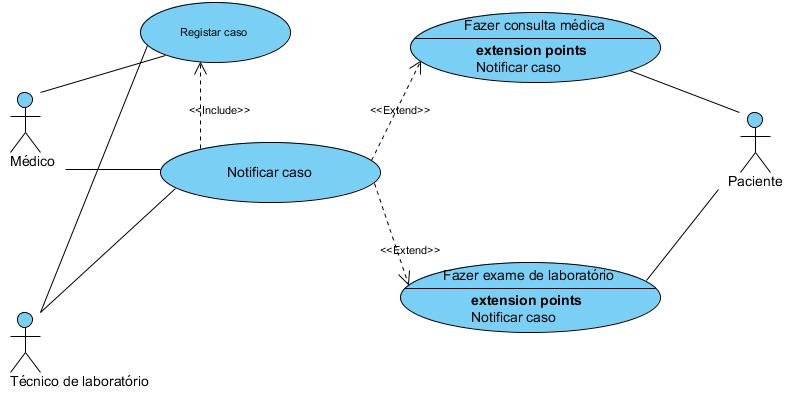
Na tabela abaixo apresenta-se os casos de uso do negócio e a justificação dos mesmos.

*Tabela 5: Casos de uso do negócio*

|  |  |  |
| --- | --- | --- |
| Nº | Caso de Uso | Justificação |
| 1 | Fazer consulta médica | Uma pessoa sente-se encomodado (doente) e dirige-se a uma unidade sanitária para fazer uma consulta com o médico especialista. |
| 2 | Fazer exame de laboratório | O pacientes solicita o exame de laboratório para confirmar o resultado de um caso suspeita recomendado pelo médico ou para confirmar as suas suspeitas de acordo com alguns sinais e sintomas apresntados. |
| 3 | Notificar caso | O profissional de saúde ( médico ou técnico de laboratório) suspeite ou confirme de se tratar de um caso de notificação de acordo com os sinais e sintomas apresentados ou com o resultado do exame realizado é preenchido o formulário para a notificação do caso. |
| 4 | Registar caso | Para manter um controlo local na unidade sanitária, é necessário que os dados notificados estejam registados nos livros de registos, por isso cabe aos médicos e técnicos de laboratório executar este caso de uso. |

## DIAGRAMA DE CASOS DE USO DO NEGÓCIO

A figura abaixo espelha os casos de uso retirados do negócio assim como a interacção do mesmo com os actores envolvidos.



*Figura 2-Diagrama de caso de uso do negócio*

## DESCRIÇÃO TEXTUAL E DIAGRAMA DE ACTIVIDADE DE CASOS DE USO

Neste ponto foca-se na descrição detalhada de alguns caso de uso assim o diagrama de actividade do mesmo para entender como funciona esta relação entre as acções e os actores envolvidos.

* + 1. **CASO DE USO FAZER CONSULTA MÉDICA**

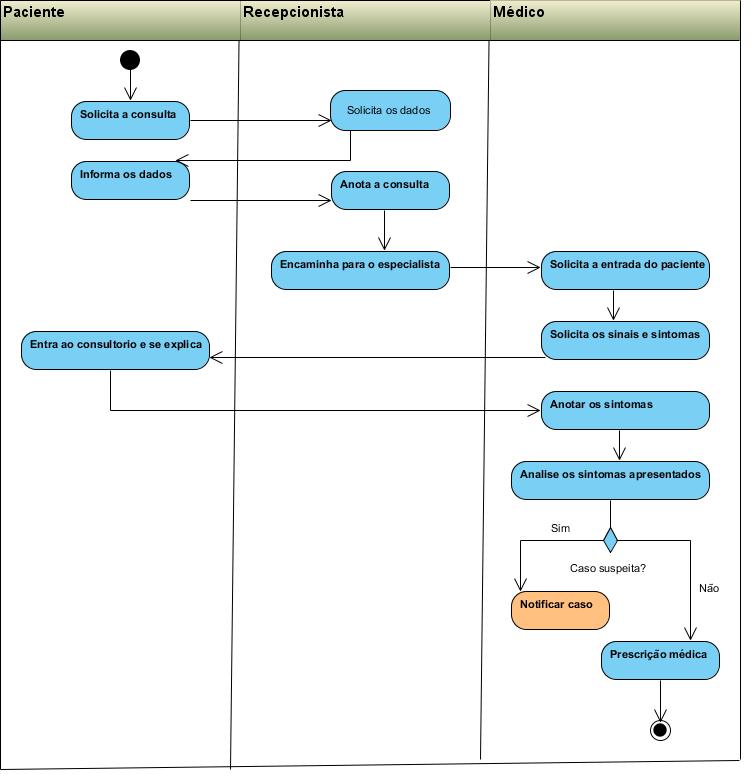
A tabela abaixo apresenta a descrição detalhada do caso de uso do negócio fazer consulta médica.

*Tabela 6: Descrição do caso de uso do negócio “fazer consulta médica”*

|  |  |
| --- | --- |
| Nome do Caso de uso | Fazer consulta médica |
| Resumo | Uma pessoa que se sente encomadada (paciente) dirige-se á unidade sanitária para explicar os sinais e sintomas na esperança de ver a sua situação melhorada. |
| Actor | Paciente |
| Pré-condições | Sentir-se incomodado |
| Pós-condições | ======================================= |
| Fluxo Básico ou normal | |
| Acção do actor | Resposta do negócio |
| 1 O paciente dirige-se a unidade sanitária e solicita uma consulta;  2 O paciente informa os seus dados e a consulta desejada;  3 O paciente entra no consultório e informa os sinais e sintomas; | 1.1 A recepcionista solicita os dados pessoais;  2.1 A recepcionista anota a consulta;  2.2 A recepcionista encaminha o paciente para o médico especialista;  2.3 O médico solicita a presença do paciente;  2.4 O médico solicita os sintomas e sinais ao paciente.  3.1 Registra os sinais e sintomas;  3.2 Analisa os sinais e sintomas apresentados; FA1  3.3 Prescrição médica. |
| Fluxo alterno | |
| FA1 – Caso os sintomas apresentados pelo paciente seja suspeita de notificação obrigatório, caso de uso notificar caso é activo, senão o médico prescreve os devidos medicamentos e cuidados que o paciente deve ter para ter a sua saúde melhorada. | |

O diagrama de actividades descreve as actividades realizadas pelo actor e pelo sistema, o diagrama de actividades está associado com o caso de uso.

A figura abaixo reflecte a interacção entre o paciente e a recepcionista nas actividades envolvidas no caso de uso marcar consulta.



*Figura 3-Diagrama de actividades do caso de uso do negócio "Fazer consulta médica"*

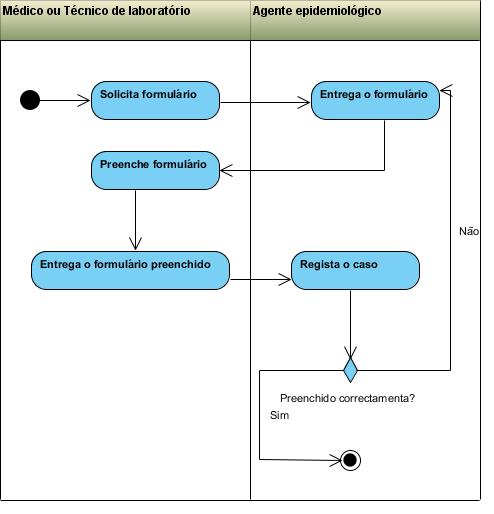
### **CASO DE USO NOTIFICAR CASO**

A tabela abaixo apresenta a descrição do caso de uso do negócio notificar caso.

*Tabela 7: Descrição do caso de uso do negócio “notificar caso”*

|  |  |
| --- | --- |
| Nome do Caso de uso | Notificar caso |
| Resumo | O caso de uso inicia quando o profissional de saúde detecta um caso de epidemia e tem como objectivo notificar o responsável pela epidemiologia da unidade sanitária. |
| Actor | Médico ou técnico de laboratório |
| Pré-condições | Caso suspeite ou caso confirmado |
| Pós-condições | Notificar ao agente epidemiológico |
| Fluxo Básico ou normal | |
| Acção do actor | Resposta do negócio |
| 1 Solicita a o formulário de notificação;  2 Preenche a ficha de notificação;  3 Entrega ao agente epidemiológico. | 1.1 O Agente epidemiologico entrega o formulário de notificação;  3.1 Verifica se os dados foram preenchidos correctamente; FA1 |
| Fluxo alterno | |
| FA1- Caso o profissional de saúde não tenha preenchido correctamente os dados, volta ao passo 1.1 | |

A figura abaixo reflecte a interacção entre os actores e trabalhadores nas actividades envolvidas no caso de uso notificar caso.



*Figura 4-Diagrama de actividades do caso de uso do negócio "notificar caso"*

## DESCRIÇÃO DO SISTEMA PROPOSTO

Deseja-se tornar digital o sistema de alerta epidemiológica mediante uma aplicação móvel, uma das grandes preocupações é o envio de informações a respeito de casos suspeitos ou casos confirmados. A aplicação que funcione em aplicativos como telemóveis, tabletes, etc. com sistema operativo móvel, vai permitir que os médicos e técnicos de laboratórios notifiquem caso no mais rápido de tempo possível e de forma mais segura.

Com o mesmo os utilizadores podem gerir seus pacientes, doenças e notificar casos, além disto como extra poderá ainda o usuário gerir consultas ou exames de laboratório.

O projecto é parte integrante de um sistema epidemiológico cujo principal objectivo é enviar as notificações recolhidas em diversos pontos para o servidor Web do mesmo sistema, por isto a gestão de instituições e de doenças será feita no servidor e a aplicação estará sincronizada no sentido de carregar as doenças por meio deste servidor, e assim sucede no momento de criação de um novo usuário.

Para envio de informação fez-se o uso de permissão de internet e utilizou-se o CRUD em PHP para que as mesmas informações guardadas na aplicação sejam também enviadas ao servidor Web.

Para entrar no sistema o usuário precisa estar autenticado com senha e utilizador válido.

## ANÁLISE DE REQUISITOS

Os requisitos de um sistema são as descrições do que o sistema deve fazer, os serviços que oferece e as restrições ao seu funcionamento. Esses requisitos reflectem as necessidades dos clientes para um sistema que serve a uma finalidade determinada, como controlar um dispositivo, colocar um pedido ou encontrar informações (Sommerville, 2011, p. 57). Abaixo apresenta-se os requisitos necessários na análise que são os funcionais e os não funcionais.

### **REQUISITOS FUNCIONAIS**

Requisitos funcionais. São declarações de serviços que o sistema deve fornecer, de como o sistema deve reagir a entradas específicas e de como o sistema deve comportar-se em determinadas situações. Em alguns casos, os requisitos funcionais também podem explicitar o que o sistema não deve fazer (Sommerville, Engenharia de Software, 2011, p. 59).

O sistema apresenta os seguintes requisitos funcionais:

1. Iniciar sessão
2. Gerir conta de usuário
3. Gerir consultas
4. Gerir análises
5. Notificar casos
6. Gerir notificações
7. Listar doenças
8. Gerir paciente
9. Terminar sessão.

### **REQUISITOS NÃO FUNCIONAIS**

Requisitos não funcionais. São restrições aos serviços ou funções oferecidos pelo sistema. Incluem restrições de *timing*, restrições no processo de desenvolvimento e restrições impostas pelas normas. Ao contrário das características individuais ou serviços do sistema, os requisitos não funcionais, muitas vezes, aplicam-se ao sistema como um todo (Sommerville, 2011, p. 59).

Os requisitos não funcionais identificados são os seguintes:

**Interface**

* Legível
* Dinâmica
* Simples
* Cores bem definidas

**Usabilidade**

* Fácil de usar, mesmo sem experiências
* Disponibilidade de ajuda para os usuários
* Rapidez na pesquisa de informações

**Portabilidade**

Funciona em todos dispositivos móveis com sistema operativo *Android* a partir versão 4.0 (*ice cream sandwich*) por diante.

**Software**

Dispositivos móveis com sistema operativo *Android* versão 4.0 (*ice cream sandwich*) ou superior.

**Hardware**

Funciona em dispositivo com sistema operativo *Android* ou em máquinas com *emuladores* para correr um dispositivo Android com memória RAM de 512 MB, 512 MB de armazenamento interno.

**Segurança**

A segurança do sistema é por nível de acesso, o utilizador precisa de autenticação para entrar no sistema.

## ACTORES DO SISTEMA

Actor é aquele que interage com o sistema enquanto o mesmo está em funcionamento, assim sendo um actor não é necestariamente uma pessoa, podendo também ser um dispositivo (Pereira, 2011).

Na tabela abaixo apresnta-se os actores que interagem com o sistema proposto.

*Tabela 8: Actores do sistema*

|  |  |
| --- | --- |
| Actor | Justificação |
| Aplicação Web | Responsável pela gestão de alguns processos como por exemplo a gestão de instituições que não é competência nem do médico, nem do técnico de laboratório mas que é muito útil para ambos. |
| Médico | Gere os seus pacientes, diagnósticos, exames e notifica casos suspeitos. |
| Técnico de laboratório | Notifica casos confirmados e gere exames de laboratórios. |

## CASOS DE USO DO SISTEMA

Um caso de uso corresponde a um conjunto de acções executadas durante a realização de uma funcionalidade do sistema. Os casos de uso concentram-se nas relações entre as funções do sistema e os usuários que delas participam de alguma forma (Pereira, 2011, p. 31).

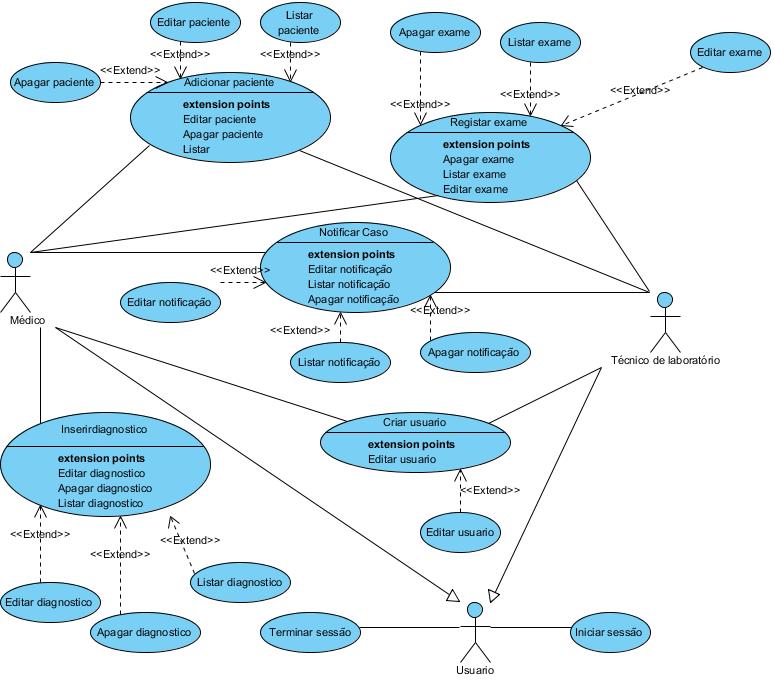
Foram extraídos os seguintes casos de uso do sistema:

1. Iniciar sessão
2. Criar usuário
3. Editar usuário
4. Apagar usuário
5. Registar paciente
6. Editar paciente
7. Listar pacientes
8. Eliminar Paciente
9. Registar diagnostico
10. Editar diagnostico
11. Eliminar diagnostico
12. Listar diagnostico
13. Notificar caso
14. Editar notificação
15. Apagar notificação
16. Listar notificação
17. Registar exame
18. Editar exame
19. Apagar Exame
20. Listar exames
21. Terminar sessão

## DIAGRAMA DE CASOS DE USO DO SISTEMA

Os casos de uso é uma técnica da UML baseada em cenários que identificam os atores em uma interação e que descreve a interação em si.

A figura abaixo espelha os casos de uso do sistema assim como a interacção do mesmo com os actores envolvidos.



*Figura 5-Diagrama de caso de uso do sistema*

## DESCRIÇÃO TEXTUAL DOS CASOS DE USO DO SISTEMA

A tabela abaixo apresenta a descrição detalhada do caso de uso do sistema iniciar sessão, que constitui um requisito obrigatório para usar a aplicação.

*Tabela 9: Descrição do caso de uso do sistema “iniciar sessão”*

|  |  |
| --- | --- |
| Nome do Caso de uso | Iniciar sessão |
| Resumo | O caso de uso inicia quando o usuário solicita ao sistema o formulário de login com propósito de entrar no sistema e termina quando este acede a página principal do sistema ou fecha o aplicação. |
| Actor | Usuário |
| Pré-condições | Ser usuário do sistema |
| Pós-condições | Aceder a página principal ou fechar a aplicação. |
| Fluxo Básico ou normal | |
| Acção do actor | **Resposta do sistema** |
| 1 Clica sobre o ícone da aplicação;  2 Preenche os campos e clica em entrar; | 1.1 Abre o formulário de login;  2.1 Verifica se todos campos foram preenchidos correctamente; FA1  2.2 Faz a validação do usuário; FA2  2.3 Abre a inteface início. |
| Fluxo Alterno | |
| FA1- Se os dados forem inseridos incorrectamente, o sistema emite uma mensagem indicando o campo mal preenchido e volta ao passo 2.  FA2-Se não se comprovar a autenticação do utilizador, voltar ao passo 2. | |
| Estrutura de dados | |
| Usuário e senha | |

A tabela abaixo apresenta a descrição detalhada do caso de uso do sistema notificar caso, que é o principal caso de uso do sistema.

*Tabela 10: Descrição do caso de uso do sistema “notificar caso”*

|  |  |
| --- | --- |
| Nome do Caso de uso | Notificar caso |
| Resumo | O caso de uso tem por objectivo informar um caso suspeito ou confirmado. O caso de uso inicia quando o utilizador detecta caso e preenche o formulário de notificação e termina quando este é guardado e enviado. |
| Actor | Médico ou técnico de laboratório |
| Pré-condições | O usuário deve estar dentro do sistema |
| Pós-condições | ============================ |
| Fluxo Básico ou normal | |
| Acção do actor | **Resposta do sistema** |
| 1 Clica em notificar caso;  2 Preenche os campos e clica em notificar; | 1.1 Abre o formulário de notificação;  2.1 Verifica se todos campos foram preenchidos correctamente; FA1  2.2 Envia e guarda as informações. |
| Fluxo Alterno | |
| FA1- Se os campos não foram inseridos correctamente, o sistema emite uma mensagem de erro indicando o campo mal preenchido e volta ao passo 3. | |
| Estrutura de dados | |
| Informações do usuário, instituição, paciente e diagnóstico ou exame. | |

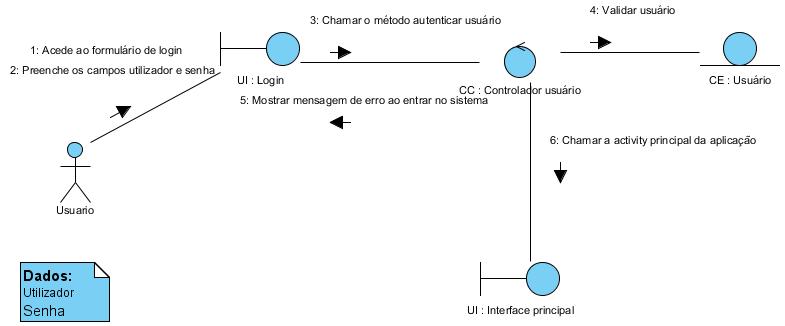
## DIAGRAMAS DE INTERACÇÃO

Como o nome indica são diagramas que demostram a interacção entre o utilizador e o sistema. Neste subtema, apresenta-se dois diagramas de interacção que são os diagramas de colaboração e os diagramas de sequência como se apresenta a seguir.

### **DIAGRAMAS DE COLABORAÇÃO**

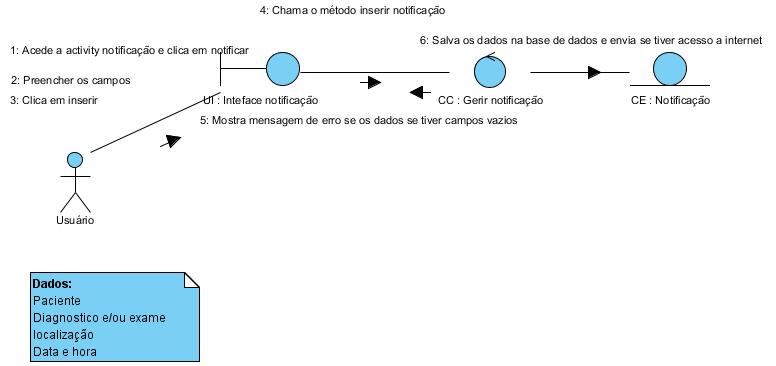
O diagrama de colaboração mostra, de maneira semelhante ao diagrama de sequência, a colaboração dinâmica entre os objectos. O diagrama de colaboração é desenhado como um diagrama de objecto, onde os diversos objectos são mostrados juntamente com os seus relacionamentos, dando à ordenação estrutural em que as mensagens são trocadas entre os objectos de um sistema (Rational , 2001).

A figura abaixo representa o diagrama de colaboração referente ao caso de uso entrar.



*Figura 6-Diagrama de colaboração "entrar no sistema"*

A notificação de casos de doenças de interesse epidemiológico representa uma das tarefas mais importantes na implentação da aplicação, a figura abaixo representa a diagramade colaboração do caso de uso notificar caso.

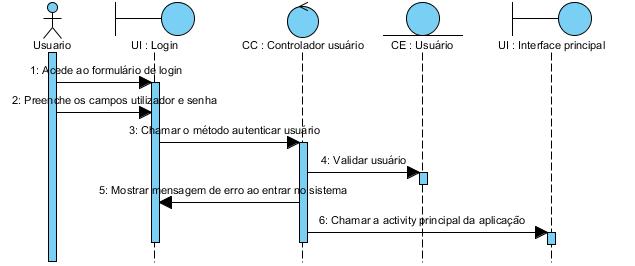


*Figura 7-Diagrama de colaboração "notificar caso"*

* + 1. **DIAGRAMAS DE SEQUÊNCIA**

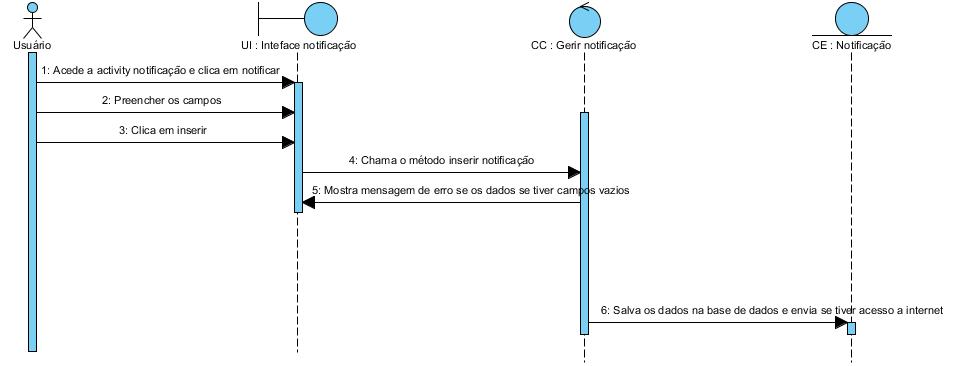
Como o nome indica, um diagrama de sequência mostra a sequência de interacções que ocorrem durante um caso de uso em particular ou numa instância de caso de uso (Sommerville, Engenharia de Software, 2011, p. 87).

Assim sendo a figura abaixo demostra detalhadamente a interacção entre o utilizador e a aplicação quando este disparar o caso de uso entrar no sistema representado pelo diagrama de sequência que é um caso de uso bastante importante pois sem passar por este o utilizador não terá acesso aos requisitos funcionais da aplicação desenhada.



*Figura 8-Diagrama de sequência "entrar no sistema"*

Na figura asseguir apresenta-se o diagrama de sequência notificar caso.

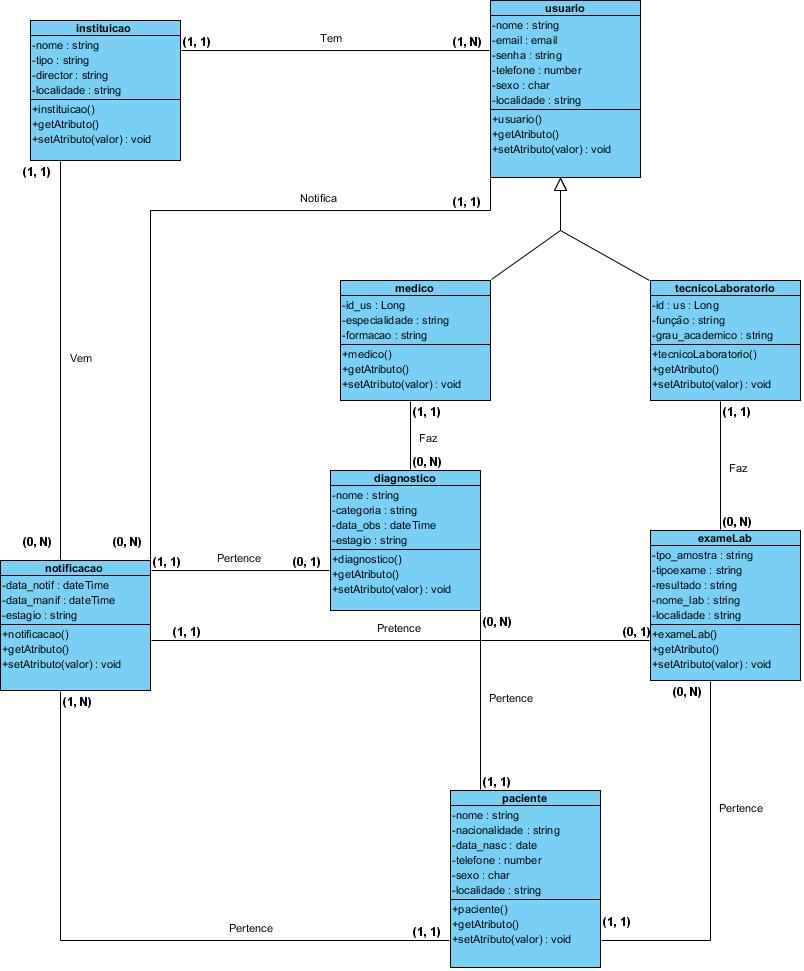


*Figura 9-Diagrama de sequência "notificar caso"*

* 1. **DIAGRAMA DE CLASSES**

Os diagramas de classes são usados no desenvolvimento de um modelo de sistema orientado a objectos para mostrar as classes do sistema e as associações entre estas classes (Sommerville, 2011, p. 90), o diagrama de classes também serve de base para os demais diagramas do modelo do sistema.

A seguir apresenta-se uma figura relativa ao diagrama de classe da aplicação desenvolvida.

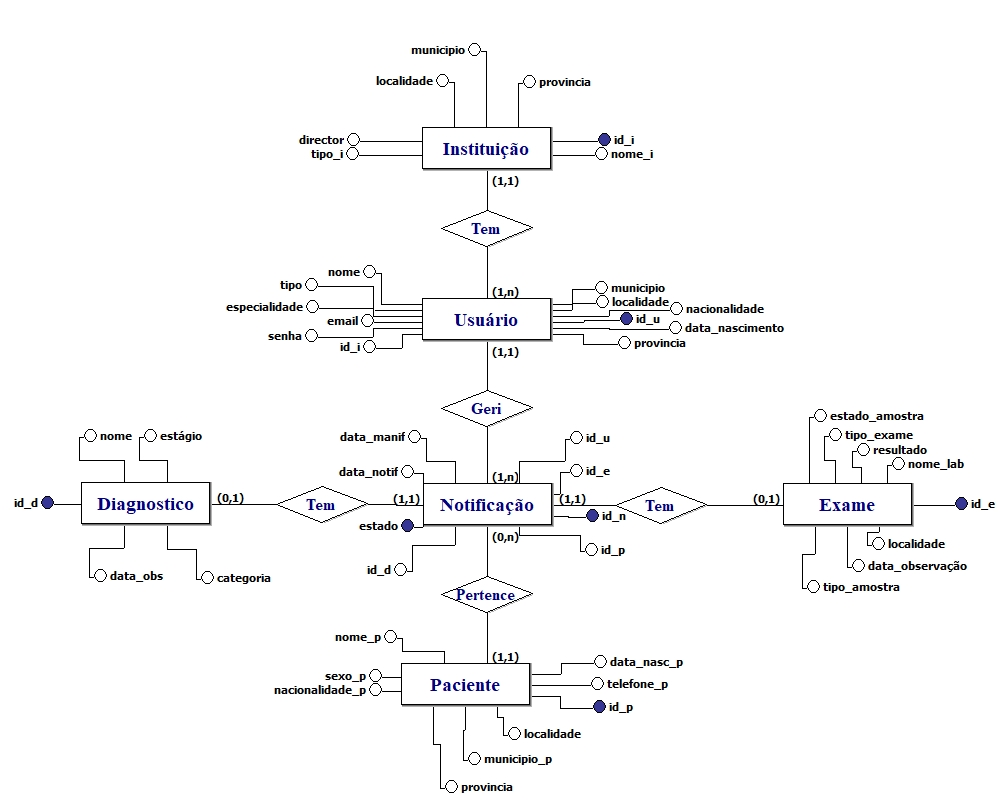


*Figura 10-Diagrama de classes*

## MODELO CONCEITUAL (DIAGRAMA ENTIDADE E RELAÇÃO)

Os modelos de dados conceituais oferecem construções no nível mais próximo do projetista para organização do mundo de interesse. O par entidade/relação é a pedra fundamental do modelo de dados. Esses pares podem ser apresentados graficamente por meio do diagrama entidade/relação (DER). A finalidade principal do DER é representar entidades de dados e seus relacionamentos dos quais as entidades de dados são representadas por um retângulo rotulado, relacionamentos são indicados por uma linha rotulada que conectam entidades.

A seguir apresenta-se a figura que representa o modelo conceitual da aplicação desenvolvida, para fazer o mesmo modelo foi utilizada o ferramenta brModelo que auxilia muito na representação de diagramas entidade relação, podendo a partir deste gerar o modelo lógico da base de dados aplicando as técnicas de normalização.

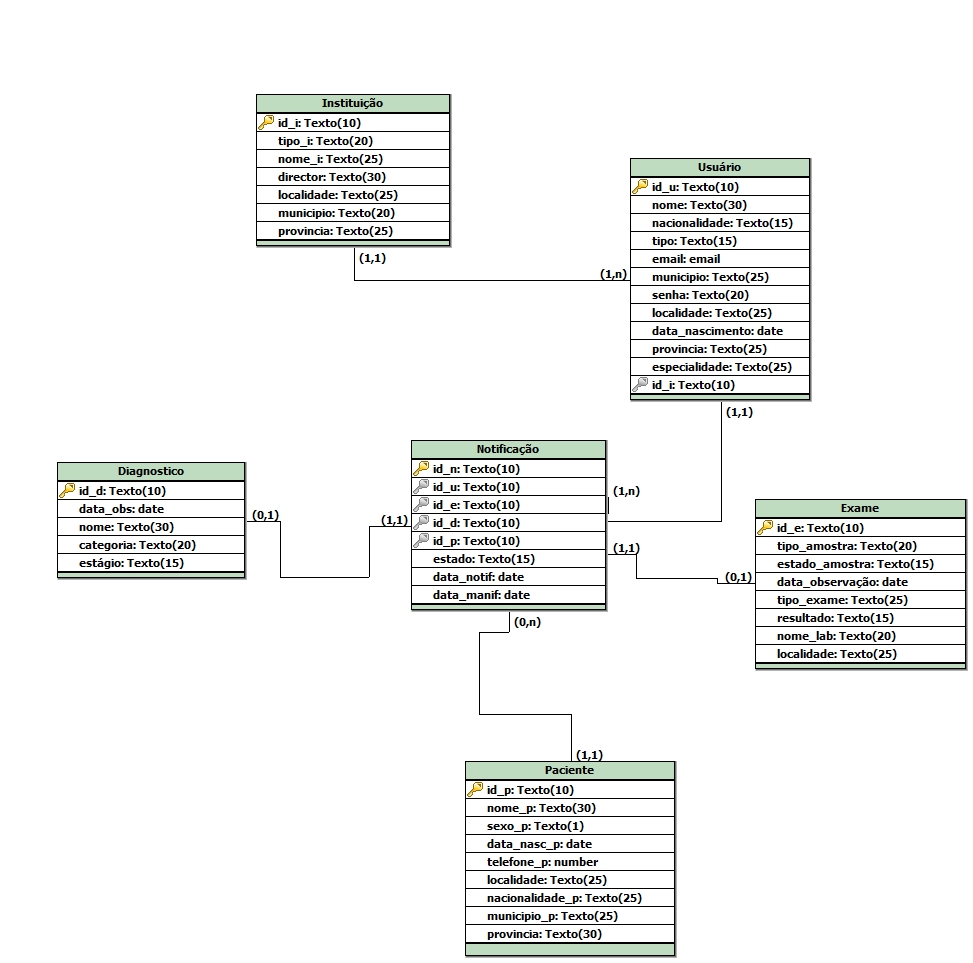


*Figura 11-Modelo conceitual da base de dados*

## MODELO LÓGICO (MODELO RELACIONAL)

Em uma base de dados ou banco de dados, os dados representam uma coleção de dados organizados que se relacionam de forma a criar algum sentido e facilitar em algum estudo ou pesquisa, O modelo relacional é o modelo lógico mais utilizado eis a razão de optar-se por esta escolha tendo em conta também a facilidade no entendimento que a mesma oferece.

Abaixo estão representados as tabelas bem como as relações existentes entre as mesmas com os respectivos atributos. A figura é referente ao modelo lógico da base de dados, gerado do modelo conceitual pelas técnicas de normalização.

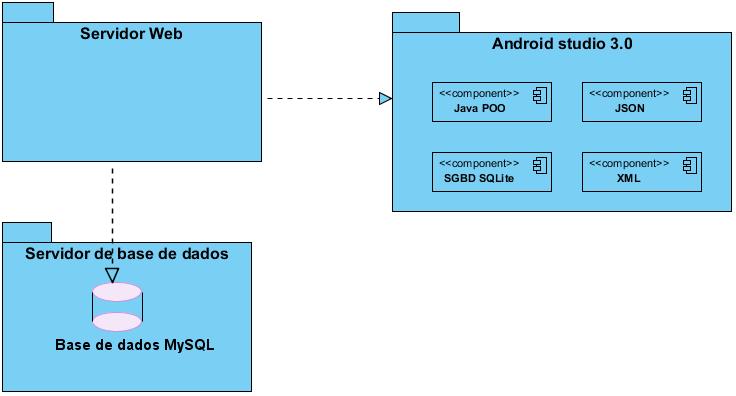
****

*Figura 12-Modelo lógico da base de dados*

## DIAGRAMA DE IMPLEMENTAÇÃO

A implementação pode envolver o desenvolvimento de programas em alto ou baixo nível de linguagens de programação, bem como personalização e adaptação de sistemas genéricos de prateleira, para atender aos requisitos específicos de uma organização (Sommerville, 2013, p. 135). os diagramas de implementação modelam a arquitetura física de um sistema.

A figura abaixo representa o diagrama de implementação do sistema proposto.



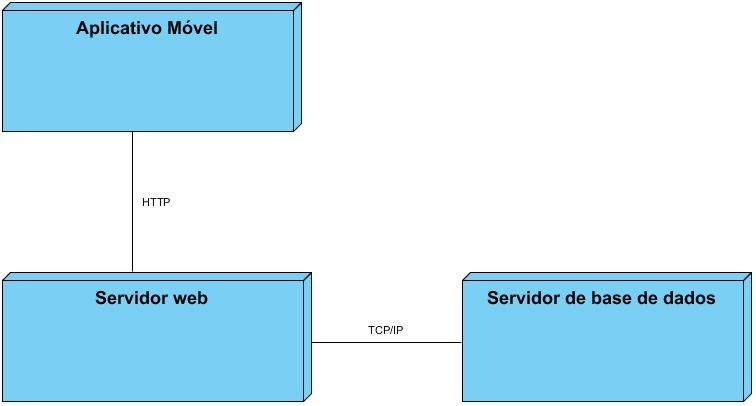
*Figura 13-Diagrama de implementação*

## DIAGRAMA DE IMPLANTAÇÃO

Um diagrama de implantação modela o inter-relacionamento entre recursos de infra-estruturas, redes ou artefactos de sistemas (Medeiros, 2004), representa os elementos de projetos em nível de implantação indicam a funcionalidade e os sistemas do referido software que serão alocados no ambiente computacional físico que vai apoiar o software..

A aplicação desenvolvida utiliza dois servidores de base de dados, o SQLite que funciona embutido na aplicação móvel e MySQL que armazena os dados replicados da aplicação móvel num servidor *Web.* Com isto garante-se que a aplicação funcione tanto *online* como *offline*.

A figura abaixo representa o diagrama de implantação da aplicação desenvolvida.



*Figura 14-Diagrama de implantação*

## FLUXO DE TESTE

Para Rational (2001) os testes servem para:

* Verificar a interacção entre os objectos.
* Verificar a correcta integração de todos os componentes do *software*.
* Verificar se todos os requerimentos foram correctamente implementados.
* Identificar e garantir defeitos abordados antes da implantação do *software*.

A fim de verificar a integridade entre os componentes bem como a interacção entre os objectos do *software* apresenta-se o fluxo de teste para o caso de teste entrar no sistema de forma detalhada.

Na tabela abaixo apresenta-se o caso de teste entrar no sistema que constitui o requisito ensencial para ter acesso ao sistema.

*Tabela 11: Caso de teste “entrar no sistema”*

|  |  |  |
| --- | --- | --- |
| Caso de teste | Entrar no sistema | |
| Teste | Verificar os dados do formulário e analisar a correspondência entre os dados. | |
| Entrada | Usuário e senha | |
| Resultado do teste | Caso exista algum campo vazio, mal preenchido ou não haja uma correspondência entre o nome do utilizador e a senha inserida o sistema mostra uma mensagem indicando o referido erro. | |
| Fluxo Básico ou normal | | |
| Acção do actor | | Resposta do sistema |
| 1🡪 O Usuário preenche o formulário para entrar no sistema e clica em entrar; | | 1.1🡪 O sistema verifica se todos campos foram preenchidos e de forma correcta; **FA1**  1.2🡪 O sistema valida o usuário, e chama a *interface* principal. **FA2** |
| Fluxo alternativo | | |
| FA1 – Caso exista algum erro nos dados ou campos vazio, o sistema informa o erro correspondente.  FA2 – Se o utilizador ou a senha inserida não existem, ou não correspondem, o sistema emite uma mensagem de erro. | | |

## CONCLUSÃO

Tanto a modelação do negócio como a modelação do sistema, constituem bases importantes para implementação e implantação de um *software*. Neste capítulo fez-se o projecto de Análise e Desenho de Sistema, apresentou-se os principais componentes e diagramas para explicar tecnicamente o funcionamento da aplicação. Foram apresentados alguns diagramas em UML feitas utilizando ferramentas específicas tais como visual paradigma, brModelo e servidor de base de dados.. A descrição detalhada do negócio assim como o entendimento da mesma, constituiu a peça fundamental para a proposta do sistema descrito neste capítulo.

# **CONCLUSÕES**

A revisão bibliográfica por meio das técnicas de pesquisas como a investigação documental, pesquisa bibliográfica e *brainstorming* que ajudou bastante em entender o negócio modelado bem como na análise e tomada de decisões das escolhas de ferramentas utilizadas para o desenvolvimento do *software*.

Hoje em dia existem diversas ferramentas informáticas auxiliadas pelo computador disponível e prático para desenvolvimento de aplicações móveis, entre as quais seleccionou-se o *Android Studio* na versão 3.0 como IDE, Java como linguagem de programação, XML que serviu para a elaboração da interface gráfica, JSON serviu como linguagem de intercâmbio de dados entre a aplicação e o servidor *Web*, visual paradigma e brModelo como ferramentas CASE para o desenho dos diagramas, SQLite e MySQL como gestores de base de dados sendo o primeiro embutida na aplicação para a gestão dos dados *offline* e outro para gestão de dados *online*.

Para o projecto de Engenharia de *Software* utilizou-se o RUP como metodologia de desenvolvimento e foi bastante eficaz e prático na modelação do negócio e no funcionamento do sistema proposto.

Na implementação da aplicação utilizou-se a tecnologia *Web service* foi perfeita para a sincronização dos dados entre a aplicação móvel e o servidor *Web*, permitindo assim que o processo de notificação seja feito de forma rápida e segura sempre que tiver acesso a *internet,* com suporte auma *interface* gráfica, simples e sustentável.

Os testes realizados durante e depois da implementação da aplicação funcionaram correctamente de acordo com as necessidades observadas e exigidas, a aplicação funciona em dispositivos com suporte a sistemas operativos *Android* 4.0 (*ice cream sandwich*) por diante.

# **RECOMENDAÇÕES**

A aplicação móvel desenvolvida é parte integrante de um sistema de vigilância epidemiológica, foi desenhada e desenvolvida com objectivo de melhorar o processo de notificação epidemiológica, facilitando as actividades de todos os envolvidos neste processo, por isto recomenda-se ao Ministério da Saúde que abrace o projecto e coloque como co-piloto em ao menos uma província para constatar as vantagens que o mesmo nos oferece.

Recomenda-se ainda que se alargue a aplicação no sentido de fazer com que outros tipos de notificação sejam implementados não ficando apenas nas notificaçoes de exames de laboratórios e diagnósticos médicos, e que seja implementada uma interface móvel que rode em dispositivos com outros sistemas operativos.

# **BIBLIOGRAFIA[[1]](#footnote-1)**

Buchele, G. T., Teza, P., Dandolini, G. A., & Souza, J. A. (2014). Métodos, Ttécnicas e ferramentas para inovação: brainstorming no contexto da inovação. p. 21. Obtido de http://www.dequi.eel.usp.br/~barcza/FerramentasDaQualidadeSEBRAE

Butcher, I. (20 de 02 de 2018). *Estudo aponta benefícios da tecnologia móvel em hospitais*. Obtido em 16 de 11 de 2018, de mobiletime: https://www.mobiletime.com.br/noticias/20/02/2018/estudo-aponta-beneficios-da-tecnologia-movel-em-hospitais/

Cordeiro, F. (2017). *Introdução à Linguagem de Programação Android*. Obtido em 07 de 10 de 2018, de androidpro: https://www.androidpro.com.br/blog/desenvolvimento-android/linguagem-de-programacao-android/

Cordeiro, F. (2017). *Kotlin: Introdução e Primeiros Passos*. Obtido em 07 de 10 de 2018, de androidPro: https://www.androidpro.com.br/blog/kotlin/kotlin/

Damas, L. (2005). *SQL* (13ª ed.). Lisboa, Portugal: FCA.

DEVMEDIA. (2017). *devmedia*. Obtido em 24 de 11 de 2018, de SQLite - O Pequeno Notável - DevMedia: https://www.devmedia.com.br/sqlite-o-pequeno-notavel/7249

Gil, A. C. (2008). *Métodos e Técnicas de Pesquisa Social* (6ª ed.). São Paulo (S.P), Brazil: Atlas (S.A).

Lacheta, R. (2016). *Android essencial* (1ª ed.). São Paulo (SP), Brazil: Novatec.

Marconi, M. d., & Lakatos, E. M. (2003). *Fundamentos da metodologia científica* (5ª ed.). São Paulo(SP), Brazil: Atlas (S.A).

Medeiros, E. (2004). *Desenvolvendo software com UML 2.0: definitivo.* São Paulo, Brasil: Pearson Makron Books.

Ministério da Saúde. (2012). *Plano nacional de desenvolvimento sanitário 2012-2025* (Vol. II). Luanda, Angola.

Ministério da Saúde. (2013). *Manual de Vigilância Epidemiológica Integrada de Doenças e Resposta.* (1ª ed.). Luanda, Angola.

Neves, P., & Ruas, R. (2005). *O guia prático do MySQL* (1ª ed.). Lisboa, Portugal: Centro Atlântico.

OMS. (2016). *Estratéria de cooperação da OMS 2015-2019 com Angola.* Angola: Escritório da OMS em Angola.

Opensoft. (07 de 06 de 2016). *Web service: O que é? Como funciona? Para que serve?* Obtido em 22 de 11 de 2018, de opensoft.pt: https://www.opensoft.pt/web-service/

Organização Mundial da Saúde. (2011). *ESTRATÉGIA INTEGRADA DE VIGILÂNCIA E RESPOSTA ÀS DOENÇAS NA REGIÃO AFRICANA* (2ª ed.). (O. Escritório Regional para a África, Trad.) Brazzaville, Congo.

Organização Mundial da Saúde. (2015). *Vigilância e Resposta Integrada às Doenças na Região Africana.* Brazzaville, Congo.

Organização Mundial da Saúde(O.M.S). (2012). *MANUAL PARA A VIGILÂNCIA INTEGRADA DE DOENÇAS DE ORIGEM ALIMENTAR NA REGIÃO AFRICANA DA OMS.* Brazzaville, Congo.

Paulus, P. B., Larey, T. S., Putman, V. L., Leggett, K. L., & Roland, E. J. (1996). Social Influence Processes in Computer Brainstorming. *Researchgate*, 14. Obtido de https://www.researchgate.net/publication/233328153\_Social\_Influence\_Processing\_in\_Computer\_Brainstorming/

Pereira, L. A. (2011). *Análise e Modelagem de Sistema com a UML: Com dicas e exercícios resolvidos* (1ª ed.). Rio de Janeiro, Brazil.

Rational . (11 de 2001). Rational Unified Process Best Practices for Software Development Teams. Rational Software White Paper.

sis4. (17 de 03 de 2017). *brModelo 3.2*. (Ronaldo, Editor) Obtido em 06 de 05 de 2019, de sis4.com: http://www.sis4.com/brModelo/

Sommerville, I. (2011). *Engenharia de Software* (9ª ed.). (K. Oliveira, & I. Bosnic, Trads.) São Paulo(S.P), Brazil.

Sommerville, I. (2013). *Engenharia de Software* (9ª ed.). (K. Oliveira, & I. Bosnic, Trads.) São Paulo, Brazil: Pearson Education.

Spadari, A. (20 de Julho de 2017). *Sistemas operacionais para celulares e dispositivos móveis*. (P. Muxfeldt, Editor) Obtido em 16 de 11 de 2018, de ccm: https://br.ccm.net/faq/11106-sistemas-operacionais-para-celulares-e-dispositivos

SQLite. (s.d.). *About SQLite*. Obtido em 02 de 05 de 2019, de SQLite: https://www.sqlite.org/about.html

starUML. (4 de 2018). *Documentação StarUML*. Obtido em 24 de 11 de 2018, de staruml: https://docs.staruml.io/

Tavares, F. (2016). *PHP com Programação Orientada a Objectos* (1ª ed.). Lisboa, Portugal: FCA-Editora de Informática, Lda.

tecmundo. (18 de 07 de 2017). *Android cresce, iOS diminui e Windows Phone quase some, aponta relatório*. (D. Ciriaco, Editor) Obtido em 16 de 11 de 2018, de tecmundo: https://www.tecmundo.com.br/dispositivos-moveis/119411-android-ios-windows-phone-mercado.htm

visual paradigm. (2018). *Visual paradigma*. Obtido em 24 de 11 de 2018, de visual-paradigm.com: https://www.visual-paradigm.com/

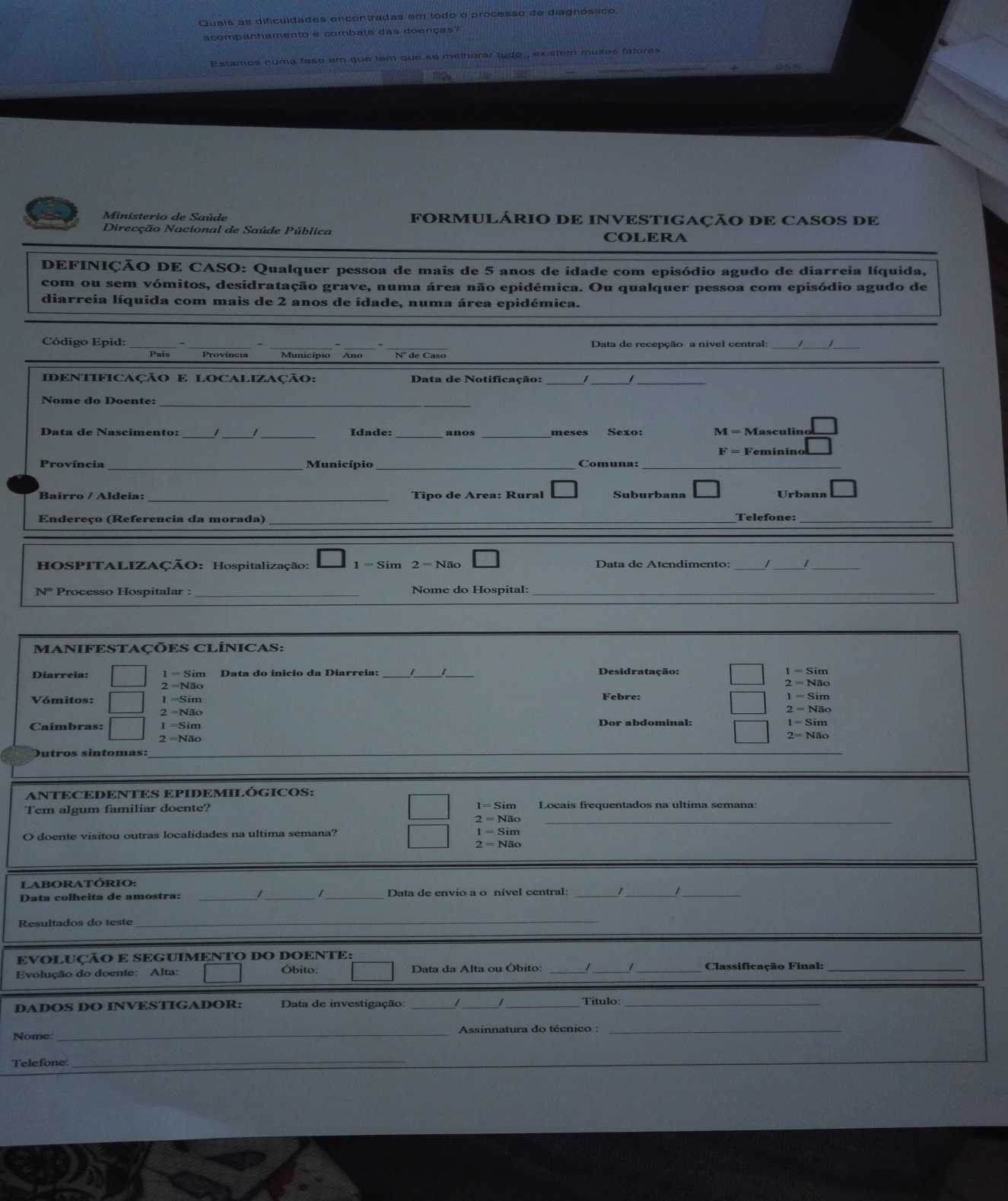
w3school. (2018). *w3c*. Obtido em 24 de 11 de 2018, de Introdução ao XML: https://www.w3.org/XML/

# **ANEXOS**

Abaixo apresenta-se algumas figuras refente as pequisas efectuadas.

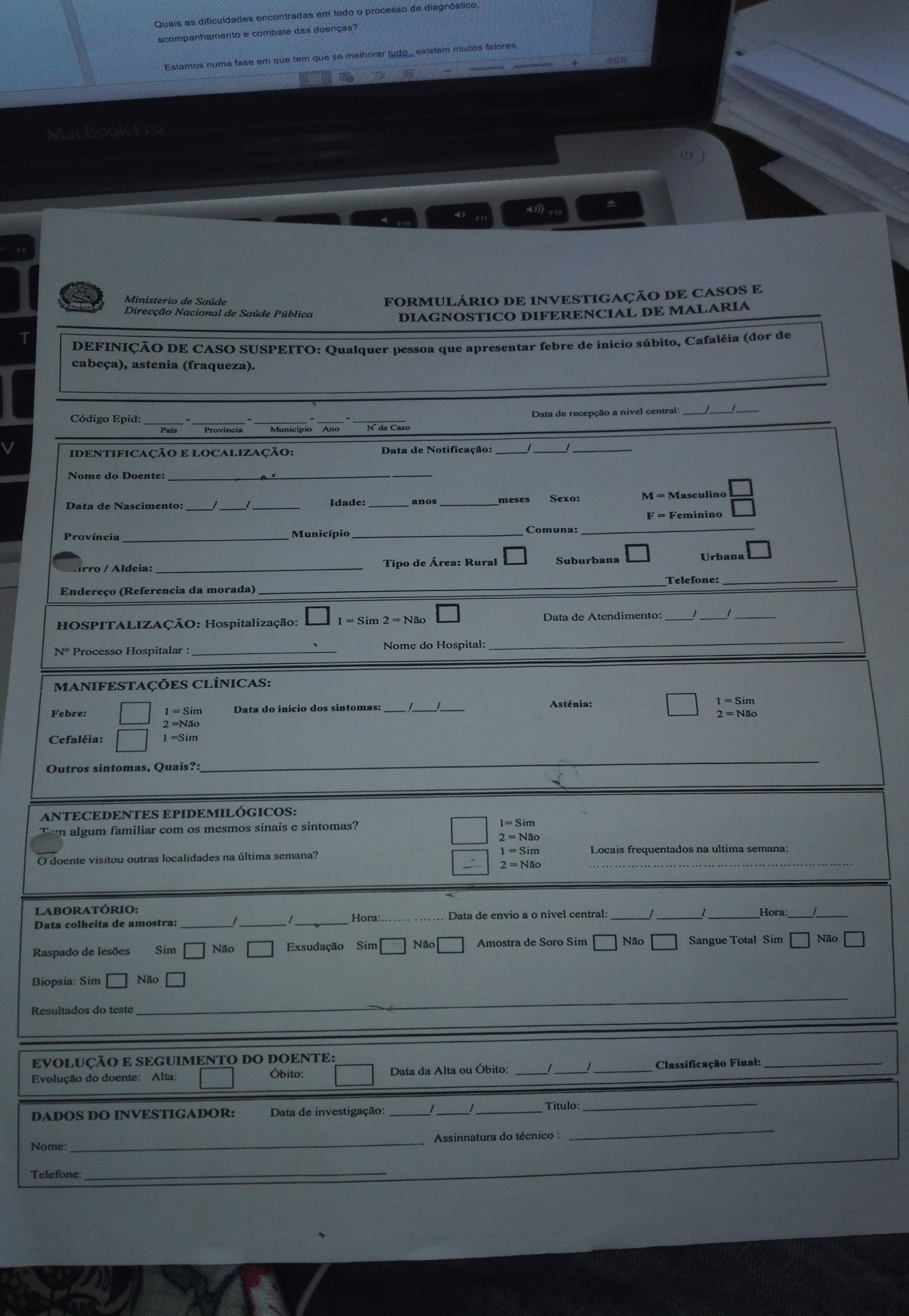
**IMAGENS DE PESQUISAS DE CAMPO**

Anexo 1- Cólera-Formulário de investigação de casos.



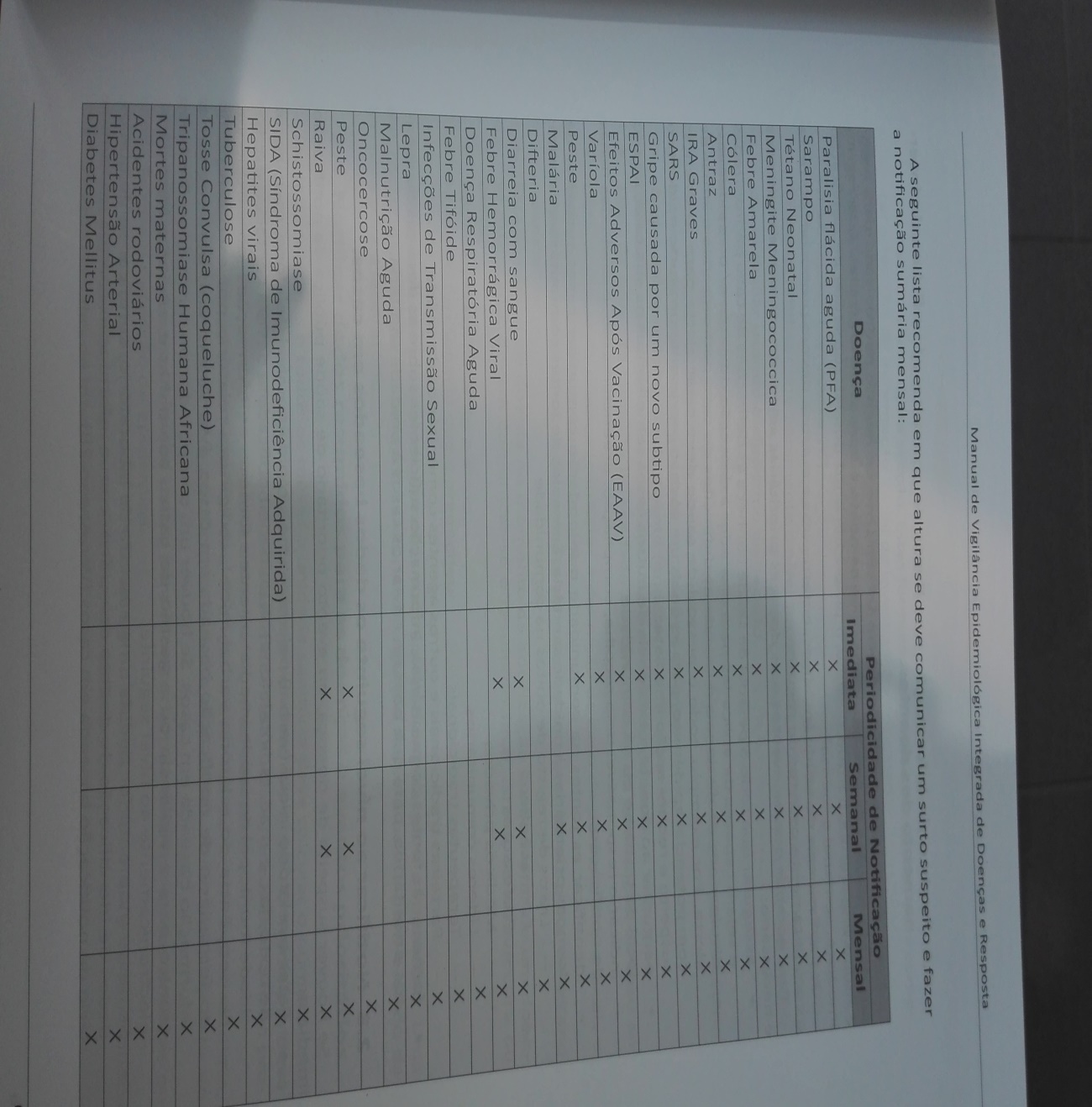
*Figura 15-Formulário de investigação de casos de cólera*

Anexo 2- Malária-Formulário de investigação de casos.



*Figura 16-Formulário de notificação de casos de malária*

Anexo 3- Listas de doenças suspeitas e prioridade de notificação.



*Figura 17-Lista de algumas doenças e suas prioridades de notificação*

# **APÊNDICES**

Abaixo apresenta-se algumas figuras refente a aplicação desenvolvida.

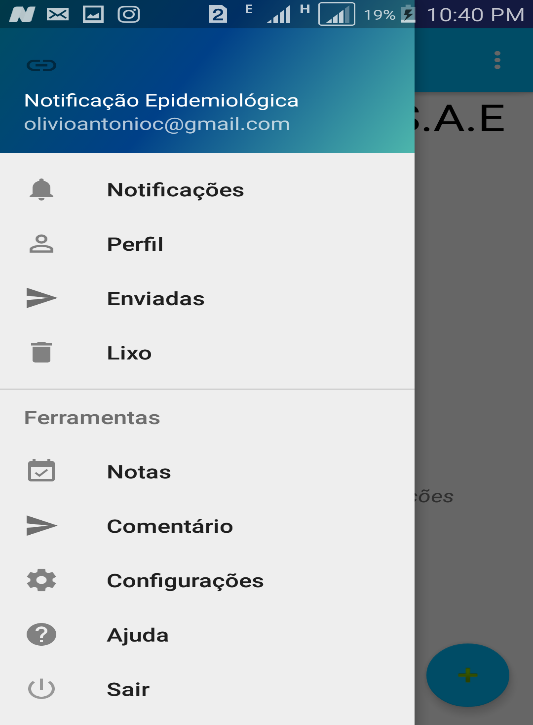
**IMAGENS DA APLICAÇÃO MÓVEL**

Apêndice 1- Tela de login da aplicação.



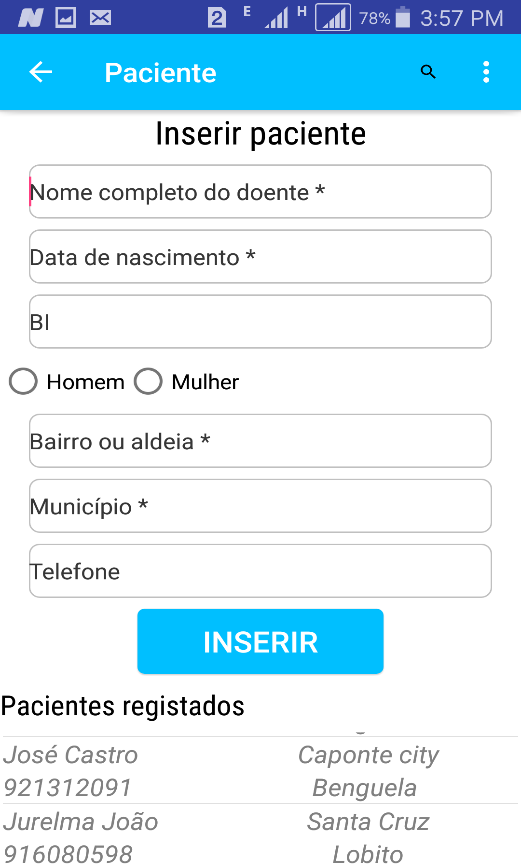
*Figura 18-Tela de login da aplicação*

Apêndice 2- Tela menu principal da aplicação.



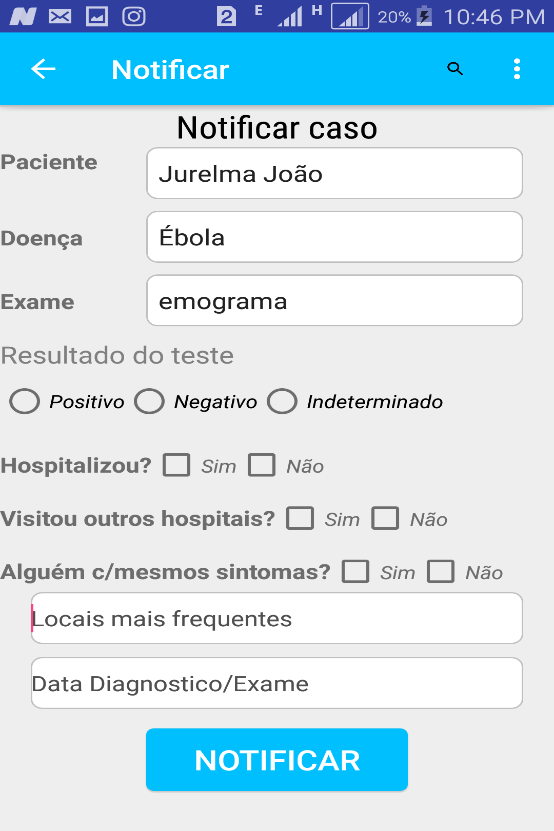
*Figura 19-Tela menu principal da aplicação*

Apêndice 3- Tela formulário “dados do paciente da aplicação”.



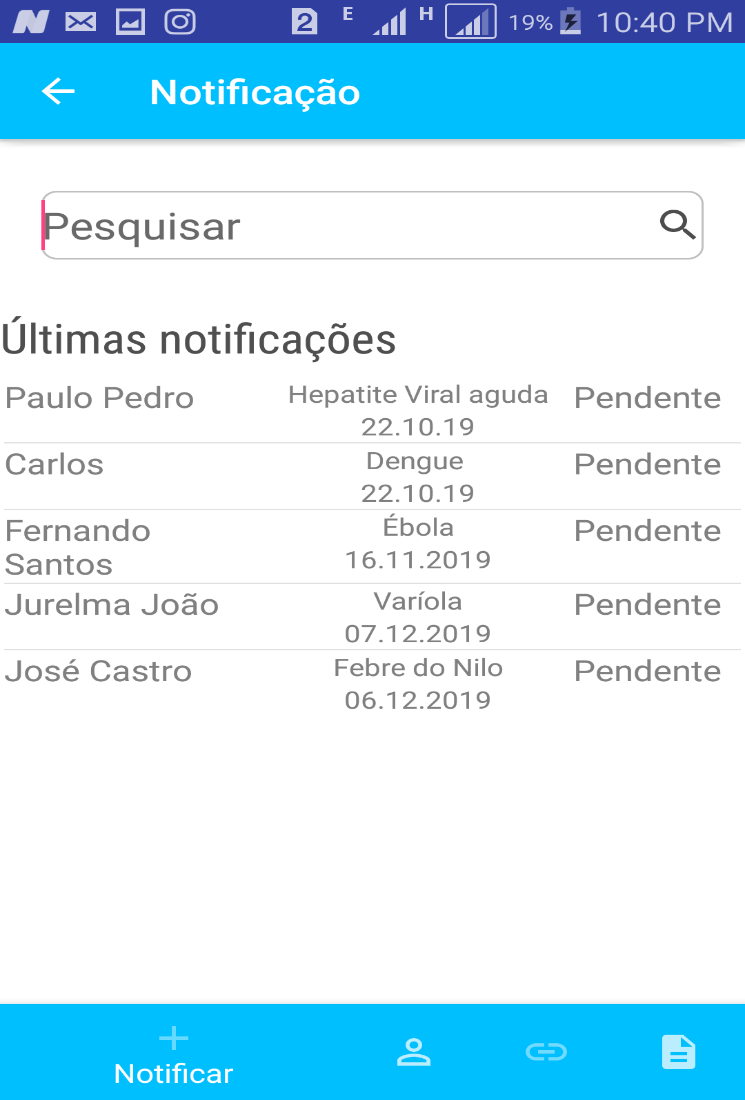
*Figura 20-Tela dados do paciente da aplicação*

Apêndice 4 –Tela formulário final de notificação.



*Figura 21-Tela formulário notificar caso.*

Apêndice 5 –Activity notificações



*Figura 22-Activity notificações*

1. Utilizou-se a norma *American Psychological Association* (APA) 6ª edição. [↑](#footnote-ref-1)