

**INSTITUTO FEDERAL DE
EDUCAÇÃO, CIÊNCIA E TECNOLOGIA**
GOIÁS
Câmpus Jataí

**Coordenação do Curso Superior de Tecnologia em
Análise e Desenvolvimento de Sistemas**

David Trindade Pires

SISTEMA DE PEDIDOS VIA ANDROID

Jataí, Fevereiro de 2018.

SISTEMA DE PEDIDOS VIA ANDROID

Trabalho de Conclusão de Curso apresentado ao Instituto Federal de Goiás - Campus Jataí como um dos pré-requisitos necessários para aprovação no Curso Superior Tecnológico em Análise e Desenvolvimento de Sistemas.

David Trindade Pires

Orientador: Renato Oliveira Abreu

Jataí, Fevereiro de 2018.

PIR/sis	Pires, David Trindade. Sistema de pedidos via Android / David Trindade Pires. -- Jataí: IFG – Campus Jataí/Coordenação dos cursos de Informática – Tecnologia em Análise e Desenvolvimento de Sistemas de Informação, 2018. 84 f.; il. Orientador: Prof^o. : Me. Renato Oliveira Abreu. Bibliografias. 1. Produtos. 2. Pedidos. 3. Clientes. 4. Vendedores. I. Abreu, Renato Oliveira. II. IFG – Campus Jataí. III. Título CDD 005.43
---------	--

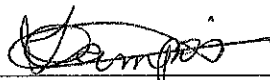
David Trindade Pires

SISTEMA DE VENDA VIA ANDROID

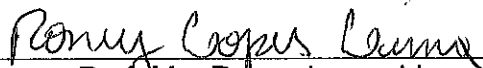
Trabalho de Conclusão de Curso, apresentado ao IFG / Campus Jataí, como requisito parcial para aprovação no Curso Superior Tecnológico em Análise e Desenvolvimento de Sistemas, sendo considerado aprovado pela Banca Examinadora formada pelos professores:



Prof. Me. Renato Oliveira Abreu



Prof. Esp. Fabrício Vieira Campos



Prof. Me. Roney Lopes Lima

Jataí, 26 de Fevereiro de 2018.

Dedico esta monografia primeiramente a Deus,
todos os meus familiares, amigos, meu
orientador e aos meus Pais.

RESUMO

O avanço da tecnologia móvel está presente no cotidiano de muitas pessoas possibilitando a conexão com a internet em quase todos os lugares que esteja. O uso do Smartphone está presente na vida da maior parte da população global. Muitos vendedores de porta a porta utilizam seu Smartphone para ajudar na sua venda para contatar clientes por meio das redes sociais, e também o auxilia com o uso de aplicativos como calculadoras e outros. Ainda não é comum a utilização de aplicativos de registro de pedidos com o intuito de organizar e manter histórico de seus pedidos. Vendedores ainda utilizam blocos de pedidos e catálogo de produtos para cumprir este papel. Este trabalho apresenta um estudo e desenvolvimento de um software para dispositivos Android capaz de substituir o bloco de pedidos e catálogo de produtos, viabilizando a praticidade, confiabilidade e confortabilidade na utilização. Apesar de conter alguns aplicativos de pedidos na loja de aplicativos do Android muitos são complexos e voltado para empresas. No software apresentado o vendedor terá a capacidade de gerenciar cliente, produtos e suas vendas por uma interface simples e prática que garantirá acesso a qualquer momento necessário, dependendo apenas do acesso à internet de um Smartphone com sistema operacional Android.

Palavras-chave: Produtos, Pedidos, Clientes, Vendedores.

ABSTRACT

The advance of mobile technology is present in the daily lives of many people, enabling internet connection in almost every place that it is, as well as the use of Smartphones. Many hawkers use these devices in their jobs, to contact customers through social networks and other applications, such as calculators. The use of order registration applications to organize and maintain the order history is not common among these professionals, they still using paper forms and product catalogs to make this job. This work presents a study and development of software for Android devices capable of replacing use of paper forms and product catalogs, making it practical, reliable and comfortable to use. Although are there many apps for this purpose in the Android store, they are hard to use and focused to private companies use. The application develop here is ables the sellers to manage his customers, products and r sales through a simple and practical interface that ensures the access whenever necessary, depending only on the internet connection and a Smartphone with Android operating system.

Keywords: Products, Orders, Customers, Sellers.

LISTA DE FIGURAS

Figura 1 – Aplicativo “meus pedidos”	15
Figura 2 - Aplicativo “PedidoOK fit”	16
Figura 3 - Ciclo de Vida do UP	17
Figura 4 - Fases do RUP.....	19
Figura 5 - Diagrama Entidade e Relacionamento.....	23
Figura 6 - Diagrama de Caso de Uso.....	25
Figura 7 - Diagrama de Classes.....	26
Figura 8 - Diagrama de Sequência.....	27
Figura 9 - Diagrama de Atividade.....	28
Figura 10 - Ranking das linguagens de Outubro de 2017	32
Figura 11 - Tela menu SPA.....	40
Figura 12 - Tela de pedidos SPA	41
Figura 13 - Diagrama de Entidade e Relacionamento.....	56
Figura 14 - Diagrama de Casos de Uso do Sistema de Pedidos via Android	57
Figura 15 - Diagrama de Classes do Sistema de Pedidos via Android	58
Figura 16 - Diagrama de Sequência Cadastrar Pedido.....	59
Figura 17 - Diagrama de Atividade Cadastrar Pedido no Sistema	60

TABELAS

Tabela 1 - Requisito Funcional – Cadastrar Usuário	42
Tabela 2 - Requisito Funcional – Logar no sistema	42
Tabela 3 - Requisito Funcional – Deslogar do sistema	43
Tabela 4 - Requisito Funcional – Cadastrar Cliente	43
Tabela 5 - Requisito Funcional – Alterar Cliente	43
Tabela 6 - Requisito Funcional – Excluir Cliente	44
Tabela 7 - Requisito Funcional – Consultar Cliente	44
Tabela 8 - Requisito Funcional – Cadastrar Produto.....	44
Tabela 9 - Requisito Funcional – Alterar Produto.....	45
Tabela 10 - Requisito Funcional – Excluir Produto.....	45
Tabela 11 - Requisito Funcional – Consultar Produto	45
Tabela 12 - Requisito Funcional – Cadastrar Pedido	46
Tabela 13 - Requisito Funcional – Alterar Pedido	46
Tabela 14 - Requisito Funcional – Excluir Pedido	47
Tabela 15 - Requisito Funcional – Consultar Pedido	47
Tabela 16 - Casos de Uso do Sistema de Vendas via Android	48
Tabela 17 - Expansão do Caso de Uso Cadastrar Usuário.....	50
Tabela 18 - Expansão do Caso de Uso Logar no Sistema.....	50
Tabela 19 - Expansão do Caso de Uso Deslogar no Sistema	51
Tabela 20 - Expansão do Caso de Uso Cadastrar Cliente.....	51
Tabela 21 - Expansão do Caso de Uso Alterar Cliente	51
Tabela 22 - Expansão do Caso de Uso Excluir Cliente	52
Tabela 23 - Expansão do Caso de Uso Consultar Cliente	52
Tabela 24 - Expansão do Caso de Uso Cadastrar Produto	52
Tabela 25 - Expansão do Caso de Uso Alterar Produto.....	53
Tabela 26 - Expansão do Caso de Uso Excluir Produto	53
Tabela 27 - Expansão do Caso de Uso Consultar Produto.....	53
Tabela 28 - Expansão do Caso de Uso Cadastrar Pedido	54
Tabela 29 - Expansão do Caso de Uso Alterar Pedido	54
Tabela 30 - Expansão do Caso de Uso Excluir Pedido.....	54
Tabela 31 - Expansão do Caso de Uso Consultar Pedido	55

SUMÁRIO

1 Introdução.....	11
1.1 O tema	13
1.2 Justificativa	14
1.3 Metodologia	15
1.3.1 UP - Processo Unificado	16
1.3.1.1 Fases da UP	17
1.3.2 Atividades básicas no processo de desenvolvimento de software	19
1.3.3 UML.....	21
1.3.3.1 Diagrama de entidade e relacionamento – DER.....	22
1.3.3.2 Diagrama de Caso de Uso	24
1.3.3.3 Digrama de Classes	25
1.3.3.4 Digrama de Sequência	26
1.3.3.5 Digrama de Atividade	27
1.3.4 Recursos e Softwares utilizados	29
1.3.4.1 Astah* Community	29
1.3.4.2 Android Studio	29
1.3.4.3 Java	31
1.3.4.4 Firebase.....	33
1.3.4.5 NoSQL.....	33
1.3.5 Requisitos para instalação dos Softwares utilizados.....	35
1.3.5.1 Astah* Community	35
1.3.5.2 Android Studio	36
1.3.5.3 Firebase.....	38
2 Desenvolvimento do sistema.....	39
2.1 Visão geral do sistema.....	39
2.2 Levantamento de Requisitos	41
2.3.1 Requisito Cadastrar usuário.....	42
2.3.2 Requisito Logar no sistema.....	42
2.3.3 Requisito Deslogar do sistema.....	43
2.3.4 Requisito Cadastrar cliente	43
2.3.5 Requisito Alterar cliente	43
2.3.6 Requisito Excluir cliente	44
2.3.7 Requisito Consultar cliente.....	44
2.3.8 Requisito Cadastrar produto	44
2.3.9 Requisito Alterar produto.....	45
2.3.10 Requisito Excluir produto	45
2.3.11 Requisito Consultar produto.....	45
2.3.12 Requisito Cadastrar pedido.....	46
2.3.13 Requisito Alterar pedido	46
2.3.14 Requisito Excluir pedido.....	47
2.3.15 Requisito Consultar pedido	47
2.4 Casos de uso	47
2.4.1 Casos de uso do sistema.....	48
2.4.2 Expansão dos casos de uso do sistema	49
2.4.2.1 Expansão do caso de uso "Cadastrar Usuário"	50

2.4.2.2 Expansão do caso de uso "Logar no Sistema"	50
2.4.2.3 Expansão do caso de uso "Deslogar no Sistema"	51
2.4.2.4 Expansão do caso de uso "Cadastrar Cliente"	51
2.4.2.5 Expansão do caso de uso "Alterar Cliente"	51
2.4.2.6 Expansão do caso de uso "Excluir Cliente"	52
2.4.2.7 Expansão do caso de uso "Consultar Cliente"	52
2.4.2.8 Expansão do caso de uso "Cadastrar Produto"	52
2.4.2.9 Expansão do caso de uso "Alterar Produto"	53
2.4.2.10 Expansão do caso de uso "Excluir Produto"	53
2.4.2.11 Expansão do caso de uso "Consultar Produto"	53
2.4.2.12 Expansão do caso de uso "Cadastrar Pedido"	54
2.4.2.13 Expansão do caso de uso "Alterar Pedido"	54
2.4.2.14 Expansão do caso de uso "Excluir Pedido"	54
2.4.2.15 Expansão do caso de uso "Consultar Pedido"	55
3 Diagramas	56
3.1 Diagrama de entidade e relacionamento do sistema	56
3.2 Diagrama de casos de uso	57
3.3 Diagrama de classes	58
3.4 Diagrama de Sequência	59
3.5 Diagrama de atividade	60
4 Conclusão	61
5 Referências	62
APÊNDICE A – Telas do sistema	64
APÊNDICE B – Diagramas de Sequência	71
APÊNDICE C – Diagramas de atividade	76
APÊNDICE D – Entrevista com vendedora de cosméticos	84

1 INTRODUÇÃO

O projeto de Trabalho de Conclusão de Curso Sistema de Pedidos via Android apresenta uma ferramenta simples e pratica para vendedores de porta a porta com o intuito de facilitar o seu cotidiano de vendas. Proporciona uma estrutura responsiva aos seus aparelhos de Smartphone para garantir melhor usabilidade. O desenvolvimento foi baseado no levantamento dos requisitos de vendedores de grandes empresas e softwares existentes no mercado, ao qual foi oferecido uma abordagem dinâmica e pratica que beneficiará os vendedores do ramo. O software foi desenvolvido voltados para plataforma mobile Android, a qual tem maior representatividade no mercado pois é de fácil acesso a população de vários tipos de classes sociais.

No livro de Marcus Cobra intitulado como Administração de Vendas relata que a atividade de vendas é antiga e já vem datadas em documentos da história da antiga Grécia que relatam a venda como atividade de troca, ou seja, muitos produtos cultivados no campo eram vendidos na cidade, e produtos manufaturados nas cidades eram comercializados no campo, também conhecida como escambo.

Antes da profissão de vendedor que foi concebida na Revolução Industrial na Inglaterra por volta do século XVIII até metade do XIX, existiam mercadores e artesões que já exerciam a atividade de vendedores. Na idade média já surgiram vendedores de “porta a porta” intitulados como vendedores ambulantes. Traziam produtos do campo e vendiam na cidade, e traziam produtos da cidade e vendiam no campo.

A partir dos anos 40 as vendas pessoais se tornaram um oficio mais profissional, pois além de muita pressão e de conversas desnecessárias dos vendedores, os compradores queriam mais informações das vendas. Com isso tornou-se maior os profissionais de vendas até então amadores.

Os compradores atuais buscam dos vendedores informações nas vendas e pós-vendas. Com isso o mercado vem se tornando mais competitivo a cada dia e os vendedores são forçados a buscar diferencias na apresentação de seus produtos e tratamento da venda antes e depois da conclusão.

Muitos vendedores externos ainda usam seus blocos de pedidos e catálogos de produtos para fazer suas vendas, e posteriormente são obrigados a registrar no sistema de CRM da empresa, ou até mesmo perder os registros por

estarem em meios físicos como cadernos de pedidos.

Segundo matéria feita pela Associação Brasileira de Empresas de Vendas Diretas em 2017, as vendas diretas é um sistema de comercialização de bens de consumo e serviços, neste modelo o vendedor não tem um estabelecimento fixo para trabalhar e o contato com os clientes é feito de forma pessoal. Muitas empresas como Natura, Mary Kay, O Boticário, Tappuerware utilizam este modelo e é muito difundido. No Brasil cerca de 4.3 milhões de pessoas são vendedores diretos representantes de marcas de vendas diretas¹. O modelo é bem simples e basta a pessoa fazer um cadastro na empresa para receber o catálogo de produto e o bloco de pedidos para começar a atuar como vendedor dos produtos.

Com o avanço da tecnologia, atualmente os dispositivos móveis contam com sistemas operacionais robustos, como Android da Google e IOS da Apple que suportam um gama de aplicativos instalados em seus sistemas, a maioria deles desenvolvidos por terceiros.

Segundo mencionado na matéria feita pelo G1 em 2017 75% da população mundial possui acesso ao celular, e grande parte são dispostos chamados de “telefones inteligentes” que são os Smartphones². Esses dispositivos vêm facilitando a vida de seus usuários proporcionando-lhes uma gama de aplicativos, acesso à internet, comunicação com outros dispositivos e até mesmo monitoramento de sua saúde.

¹ Disponível em: <http://www.abevd.org.br/venda-direta/> acesso em fev. 2018.

² Disponível em: <http://g1.globo.com/tecnologia/noticia/2012/07/75-da-populacao-mundial-tem-acesso-ao-celular-afirma-pesquisa.html> acesso em fev. 2018.

1.1 O TEMA

O sistema de pedidos via Android é um software simples, prático e funcional que garante facilidade no cotidiano de vendas dos vendedores de porta a porta. Oferece segurança no armazenamento de dados pois o banco de dados está na nuvem e praticidade pois facilmente é baixado na loja de aplicativo do Android. O vendedor poderá leva-lo em seu bolso pois é instalado no Smartphone, e ter acesso no momento em que precisar. O sistema apresenta telas simples de gerenciamento de clientes, produtos e pedidos.

Para o desenvolvimento do software foi utilizado recursos do Google como o Android que é o sistema operacional para Smartphone de várias marcas presente no mercado que é baixo custo para aquisição e oferece uma enorme gama de aplicativos gratuitos e pagos para serem utilizados, e é muito difundido atualmente. O software foi desenvolvido em com a IDE oferecida pelo Google, o Android Studio que oferece um ambiente de desenvolvimento completo a partir da linguagem JAVA. Para integração com banco de dados e sistema de autenticação utilizou-se outra ferramenta gratuita do Google que é o Firebase, esta ferramenta no plano Spark oferece banco de dados de até 1GB de armazenamento, 10GB/mês de download e 100 conexões simultâneas, além de disponibilizar o controle dos usuários do sistema por meio de e-mail, Facebook, conta Google, entre outros módulos³.

O objetivo do software foi atender as requisições dos vendedores de porta a porta nas suas vendas, oferecendo um software que contém informações dos produtos como nome, valor e código de clientes como contato, endereço e informações pessoais e também dos registros de pedidos de produtos dos seus respectivos clientes. Vários vendedores conseguem utilizar o software centralizando todas as informações em um único sistema.

Para melhor entendimento das necessidades foram analisados dois dos mais populares softwares que estão presentes na loja de aplicativos do Android, e foi feita análise levando em consideração suas deficiências e rotinas complexas. Além do estudo do mercado e de softwares existentes foi feita observação com uma vendedora de cosméticos para entender melhor a realidade do vendedor de porta a porta, a entrevista encontra-se no apêndice deste trabalho.

³ Disponível em: <https://firebase.google.com/pricing/?hl=pt-br> acesso em fev. 2018.

1.2 JUSTIFICATIVA

Levando em consideração as condições tecnológicas atuais torna-se viável a utilização de um sistema capaz de absorver o papel que os blocos de pedidos e os catálogos de produtos tem em um aplicativo de celular. Este aplicativo tem como funcionalidade o gerenciamento de clientes, produtos e pedidos em tempo real por meio da internet apenas com um dispositivo que caberá na palma de sua mão, o Smartphone.

Neste aplicativo poderá ver facilmente os produtos disponíveis para o registro de pedidos com seus respectivos valores. Também consultar o cadastro de cliente e fazer os pedidos facilmente utilizando o dispositivo. Também terá um histórico de todas as vendas realizadas.

A proposta de desenvolvimento deste aplicativo facilitará o cotidiano de trabalho do vendedor de porta a porta, provendo funcionalidade praticas para um registro de pedidos ágil com o armazenamento de informações na nuvem que garantirá uma confiabilidade dos dados gerados.

Atualmente os Smartphones com sistema operacional Android está com preço muito baixo, hoje em dia com R\$280,00 qualquer pessoa consegue comprar um aparelho. O acesso à internet móvel está muito barato, pois a planos de internet diária que custam 75 centavos ao dia.

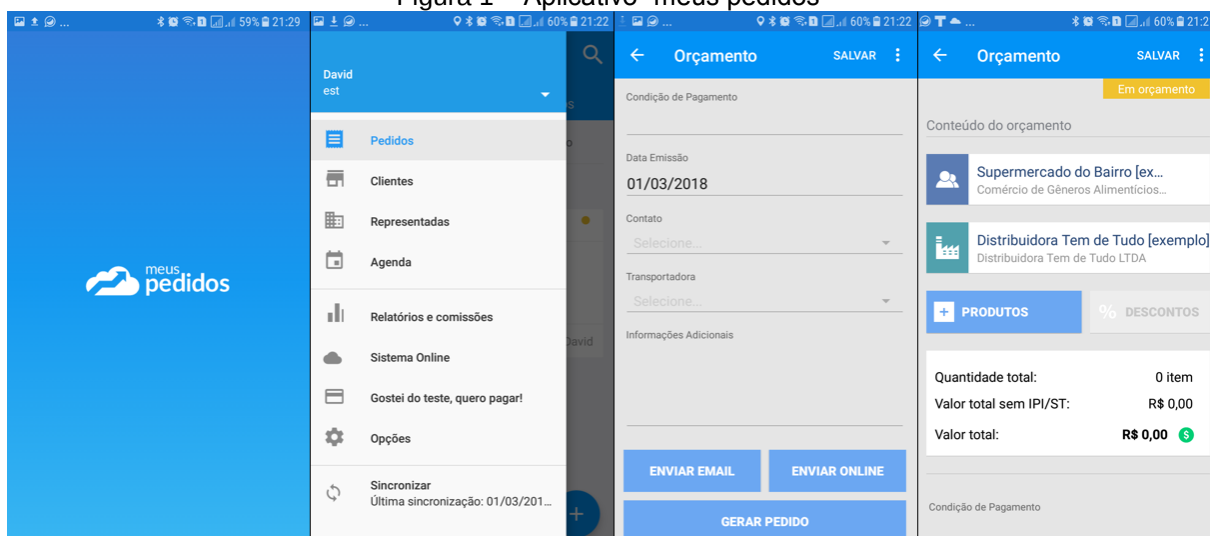
Em suma o software oferece recursos de fácil utilização com segurança e usabilidade para o usuário, já outros softwares de pedidos oferecem de forma complexas e muito poluídos. Também foi levado em consideração a observação do cotidiano dos vendedores de porta a porta.

1.3 METODOLOGIA

A metodologia utilizada foi baseada em recursos disponibilizados da internet e na análise de dois dos mais populares aplicativos de pedidos disponibilizado na loja de aplicativos do Android.

A primeira análise foi feita no aplicativo “meus pedidos” que tem 100 mil downloads na loja de aplicativos do Android e possui recursos como gerenciamento de pedidos, clientes, representante, agenda entre outros. Oferece cinco dias para utilizar o sistema gratuitamente, após o período de teste é necessário a aquisição do aplicativo por meio de mensalidade. O aplicativo possui recursos de pedidos e orçamento e guarda as informações na nuvem. Quem não está acostumado pode se perder nas funcionalidades que apresenta, necessitando de estudo para utilização do mesmo. Segue abaixo imagens do aplicativo.

Figura 1 – Aplicativo “meus pedidos”

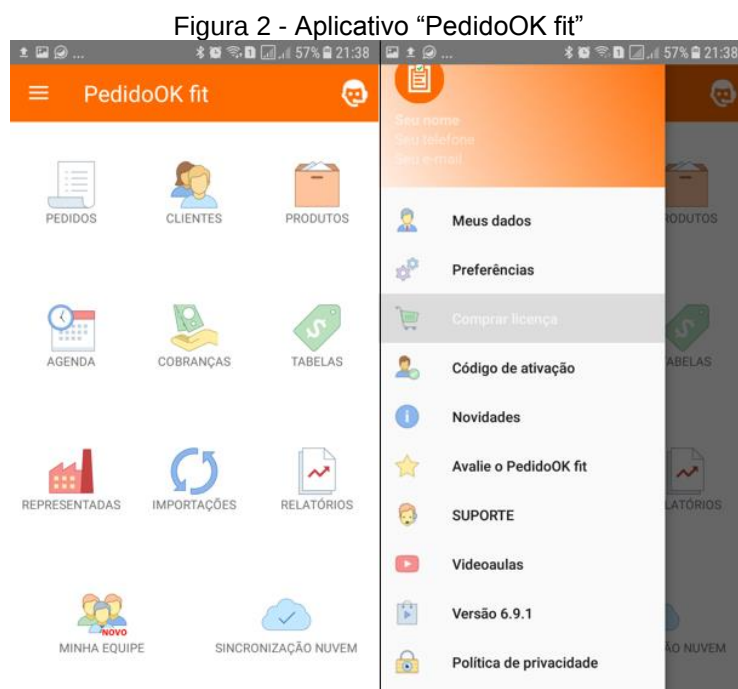


Fonte: print screen do aplicativo meuspedidos⁴

A segunda análise foi feita no aplicativo pago “PedidoOK fit” que tem 100 mil downloads na loja de aplicativos do Android e possui recursos como gerenciamento de pedidos, clientes, produtos, agendas entre outros. A versão do aplicativo armazena os dados dos pedidos no Smartphone em que foi baixado, então se ocorrer qualquer problema acarretará perda dos dados. Para armazenamento na

⁴ Disponível em: https://play.google.com/store/apps/details?id=br.com.meuspedidos&hl=pt_BR acesso em fev. 2018

nuvem é necessário fazer uma conta na plataforma PC e seguir os procedimentos passados pelo tutorial do aplicativo. Para mais informações na utilização é necessário seguir o vídeo aulas disponibilizadas no menu do aplicativo para melhor entendimento da ferramenta. Segue abaixo imagens do aplicativo:



Fonte: print screen do aplicativo PedidosOK fit⁵

Foram feitos questionamentos para vendedores de porta a porta para melhor abordagem do desenvolvimento da aplicação. Foi levantado os pontos do que é necessário para fazer uma venda de porta a porta. No apêndice consta entrevista com vendedora de cosméticos como registro de observação.

O método utilizado para o desenvolvimento do software será o UP (Unified Process, ou em português Processo Unificado). Neste método são utilizados diagramas de UML que é uma linguagem de modelagem unificada de projeto.

1.3.1 UP - Processo Unificado

⁵ Disponível em: https://play.google.com/store/apps/details?id=br.com.releaseti.pedidookfit&hl=pt_BR acesso em fev. 2018.

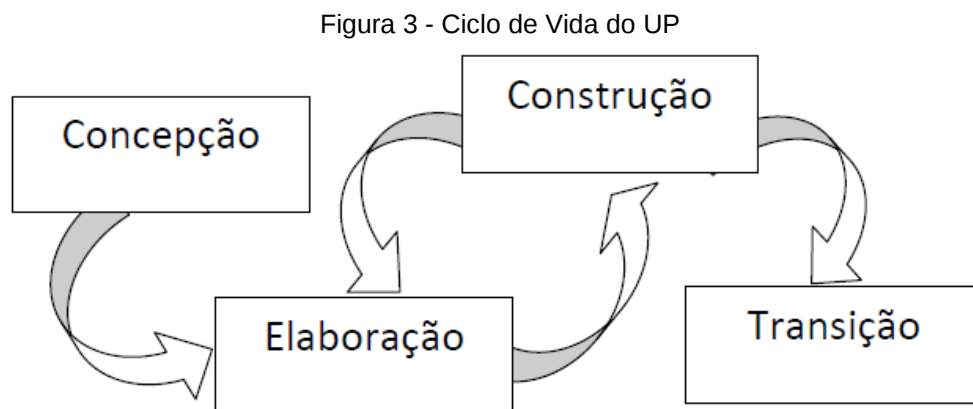
O Processo Unificado - UP (sigla em inglês para Unified Process) é um método de desenvolvimento de sistema que é dividido por quatro fases. A primeira é a Concepção, que é o levantamento da análise de requisitos. A segunda é a Elaboração, que é representada pela análise dos requisitos e o projeto do software. A Construção que é quando o software é implementado e testado. E por último a Transição que representa a implantação do software.

Este método está associado com a notação UML, e foi desenvolvido por três gurus da orientação a objetos Grady Booch, James Rumbaugh e Ivar Jacobson. Eles colocaram no método os seus mais de trinta anos de experiências para desenvolvimento do método.

O UP leva em consideração três valores que o fundamenta. O primeiro valor descreve que é dirigido por casos de uso, onde o planejamento do desenvolvimento é realizado após a identificação dos casos de uso, que serão tratados com priorização dos mais complexos. O segundo é centrado na arquitetura, que irá considerar os requisitos para realização da construção da arquitetura, que será baseada na estrutura de classes produzida por um modelo conceitual. O terceiro é iterativo e incremental que dará para cada ciclo de trabalho realizado características adicionadas a arquitetura do sistema aproximando-a do sistema final.

1.3.1.1 Fases da UP

As quatro fases da UP são representadas por Concepção, Elaboração, Construção e Transição. Conforme mostra a figura 1.



Fonte: Pagina infoescola.⁶

⁶ Disponível em: <https://www.infoescola.com/engenharia-de-software/rup/> acesso em fev. 2018.

A fase de concepção incorpora o estudo de viabilidade, o levantamento dos requisitos e uma parte da sua análise. A fase de elaboração incorpora o detalhamento da análise de requisitos, a modelagem de domínio e o projeto. A fase de construção corresponde à programação e testes, e a fase de transição consiste na instalação do sistema e migração de dados. (WAZLAWICK. Análise e Projeto de Sistemas de Informação Orientado a Objetos, 2010, pag. 5)

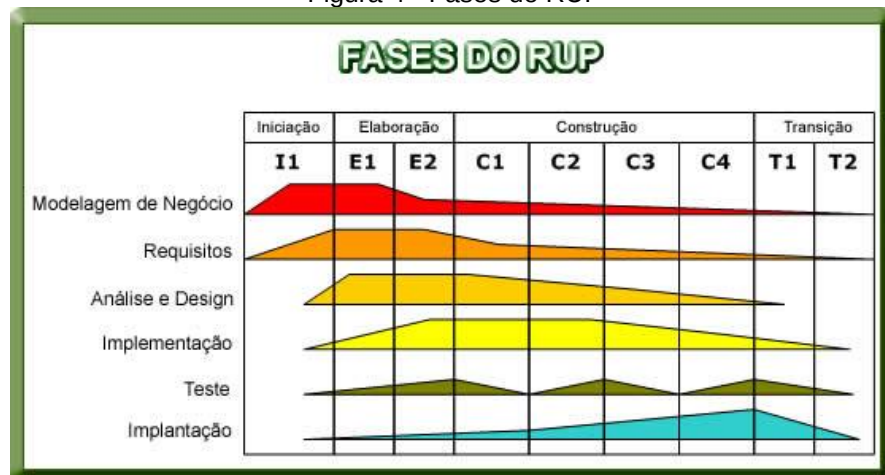
A fase da concepção que é a primeira fase do processo unificado trata do estudo da viabilidade e levantamento de requisitos. Ou seja, nesta fase será definido os requisitos e riscos para desenvolvimento do sistema. O cronograma para o desenvolvimento será definido a partir de casos de uso de alto nível.

A segunda fase é a fase de elaboração e a terceira é a construção do sistema. Na elaboração está presente a maior parte de análise e projeto, e na construção está presente a implementação e testes. Nos ciclos iterativos ocorre uma análise detalhada do sistema viabilizando a modelagem de domínio, adotando padrões de projetos para o projeto do sistema.

A terceira fase é chamada de transição, nesta fase será substituído o sistema até então utilizado antes do projeto, seja ele manual ou computadorizado, pelo sistema desenvolvido. Nesta fase será decidido se serão migrados os dados antigos para o sistema atual ou iniciado do zero, viabilizando melhor organização e atendimento dos requisitos solicitados.

A UP é uma metodologia ágil que busca fornecer de forma prática e rápida e com pouca burocracia o desenvolvimento do software de forma eficiente. E gera uma documentação com o intuito de que cada tarefa atribuída para o desenvolvedor seja com objetivos claros, considerando principalmente à produção de um código capaz de atender em menor tempo os requisitos propostos.

Figura 4 - Fases do RUP



Fonte: Pagina infoescola.⁷

As diferentes atividades de análise e projeto não ocorrem de forma estanque em cada uma das fases do processo unificado, mas podem ocorrer com maior ou menor ênfase nas diferentes fases. A figura 1.2 é a representação clássica da distribuição das atividades de desenvolvimento de sistema e sua ênfase nas diferentes fases da implementação mais conhecida do UP, denominada RUP, ou Rational Unified Process (Kruchten, 2003).

(WAZLAWICK. Análise e Projeto de Sistemas de Informação Orientado a Objetos, 2010, pag. 6)

1.3.2 Atividades básicas no processo de desenvolvimento de software

As atividades do desenvolvimento do software devem ser definidas como atividades organizadas para planejar, desenvolver, testar e manter o software. E para isso as atividades mais importantes são os levantamentos de requisitos, análise de requisitos, projeto, implementação, testes e a implantação.

No levantamento de requisitos é a etapa mais importante da atividade de desenvolvimento do software. É onde será realizado entrevistas para levantamento dos requisitos propostos para o software deixando claro as expectativas esperadas no sistema pelos usuários. Possibilitando ao desenvolvedor uma clara visão do software a ser desenvolvido, evitando problemas de funcionalidade do sistema causada pela má comunicação entre usuário e desenvolvedor. Então nessa fase é

⁷ Disponível em: <https://www.infoescola.com/engenharia-de-software/rup/> acesso em fev. 2018.

essencial gastar o tempo necessário para compreender os processos que serão automatizados no negócio em uma organização com o intuito de atendê-las de forma completa. O restante do processo de desenvolvimento orienta-se da concepção do levantamento dos requisitos passados pelos usuários que vão de acordo com suas reais necessidades.

A análise de requisitos também conhecida como especificação de requisitos detalha os dados levantados no levantamento de requisitos, e são gerados modelos com a finalidade de apresentar o software a ser desenvolvido. Nessa etapa o objetivo é criar uma estratégia de solução antes da estratégia de realização, ou seja, antes de definir como o sistema irá solucionar o requisito o sistema deve definir o que deve solucionar no requisito.

Equipes de desenvolvimento devem definir completamente o problema levantado no requisito antes de partirem para o desenvolvimento da solução do problema em questão, e para isso precisam validar e verificar os modelos construídos.

Na validação o principal objetivo é de garantir que o sistema atenderá as necessidades passadas no levantamento de requisitos do cliente. A verificação usa os modelos construídos na análise para verificação na conformidade passadas no levantamento de requisitos passado pelo cliente.

No projeto deve ser definido como o sistema funcionará internamente a fim de garantir que os requisitos passados pelo cliente sejam atendidos. Nessa fase devem ser definidos qual será a linguagem de programação utilizada, arquitetura do sistema, banco de dados, interface gráfica dentre outros. Gerar um descritivo mencionando qual a tarefa do software de acordo com as análises levantadas na fase de análise de requisitos.

O projeto possui duas atividades básicas: projeto da arquitetura (ou parte de alto nível), e projeto detalhado (ou projeto de baixo nível).

Em um processo de desenvolvimento orientado a objetos, o projeto da arquitetura normalmente é realizado por um arquiteto de software. O projeto da arquitetura visa distribuir as classes de objetos relacionados do sistema em subsistemas e seus componentes, distribuindo também esses componentes pelos recursos de hardware disponíveis.

Já no projeto detalhado, são modeladas as relações de cada módulo com o objetivo de realizar as funcionalidades do módulo. Além de desenvolver o

projeto de interface com o usuário e o projeto de banco de dados.
(Atividades básicas ao processo de desenvolvimento de Software.
DEVMEDIA. Disponível em: <<http://www.devmedia.com.br/atividades-basicas-ao-processo-de-desenvolvimento-de-software/5413>> Acesso em: 19 nov. 2017.)

A implementação é a etapa responsável por gerar o código executável que é compilado a partir da descrição computacional da fase de projeto. Nesse processo o desenvolvimento é feito através da orientação a objetos baseados nas classes de objetos levantados, utilizando linguagens de programação com suporte a orientação a objetos como por exemplo a linguagem Java e C++. É comum a utilização de bibliotecas de classes prontas e ferramentas para agilizar a produção do código.

Na etapa de teste é realizado várias validações no software desenvolvido testando todas as funcionalidades e verificando se estão de acordo com os requisitos levantados inicialmente no projeto, com o intuito de validar e homologar o software utilizado pelo usuário eliminando qualquer defeito ou não comprimento de requisitos passados. Isso é feito por cada modulo desenvolvido no sistema, e ao final os módulos serão integrados formando o produto de software.

A etapa final é a implantação onde será feita a instalação do software no ambiente do usuário final. Nesta etapa inclui os manuais e procedimentos do software, migração e/ou importação de dados do sistema antigo para o novo sistema, treinamento do software com instruções do uso correto.

Essas são algumas etapas básicas do procedimento de desenvolvimento de um software capaz de atender as necessidades levantadas em um prazo determinado com um orçamento que não foge do levantado inicialmente.

1.3.3 UML

A linguagem UML (sigla em inglês para Unified Modeling Language ou Linguagem de Modelagem Unificada) é utilizada internacionalmente de forma padrão na modelagem de software baseados na orientação a objetos. Ela não é uma linguagem de programação ou uma metodologia, é uma linguagem de modelagem que tem como objetivo auxiliar na definição de características de um software.

Deve ficar bem claro, porém, que a UML não é uma linguagem de programação e sim uma linguagem de modelagem, uma notação, cujo o objetivo é auxiliar os engenheiros de software a definirem as características do sistema, tais como seus requisitos, seu comportamento, sua estrutura lógica, a dinâmica de seus processos e até mesmo suas necessidades físicas em relação ao equipamento sobre o qual o sistema deverá ser implantado.

(Guedes, Gileanes.UML2 uma abordagem prática, 2011, pag. 19)

A UML fornece uma estrutura simples com o objetivo de auxiliar na engenharia de software. Ela é utilizada antes mesmo do software começar a ser desenvolvido, e é independente de qualquer tipo de linguagem de programação podendo ser utilizada em qualquer processo no desenvolvimento de algum software. O principal objetivo é de padronizar o processo de desenvolvimento de software.

Na modelagem do software com a UML pode encontrar diversos benefícios que contribuirão para um sistema robusto, estruturado, organizado e adaptável. Isso porque será levado em consideração fatores como documentação do sistema, modularização, mudanças ou melhorias em módulos do sistemas e ajustes necessários. Vários profissionais afirmam que os softwares são "vivos" pois nunca estão definitivamente prontos e a estruturação e modelagem garante um software de baixo custo de desenvolvimento, implantação e principalmente manutenção.

Se um software não for bem planejado o custo da manutenção será muito alto e desgastará a equipe de desenvolvimento devido a complexibilidade que será ao fazer a manutenção.

Um modelo de software captura uma visão de um sistema físico, é uma abstração do sistema com um certo propósito, como descrever aspectos estruturais ou comportamentais do software. Esse propósito determina o que deve ser incluído no modelo e o que é considerado irrelevante. Assim um modelo descreve completamente aqueles aspectos do sistema físico que são relevantes ao propósito do modelo, no nível apropriado de detalhe.

(Guedes, Gileanes.UML2 uma abordagem prática, 2011, pag. 21)

1.3.3.1 Diagrama de entidade e relacionamento – DER

O diagrama de entidade e relacionamento (DER), baseado no modelo de entidade e relacionamento (MER), consiste em um modelo conceitual usado para

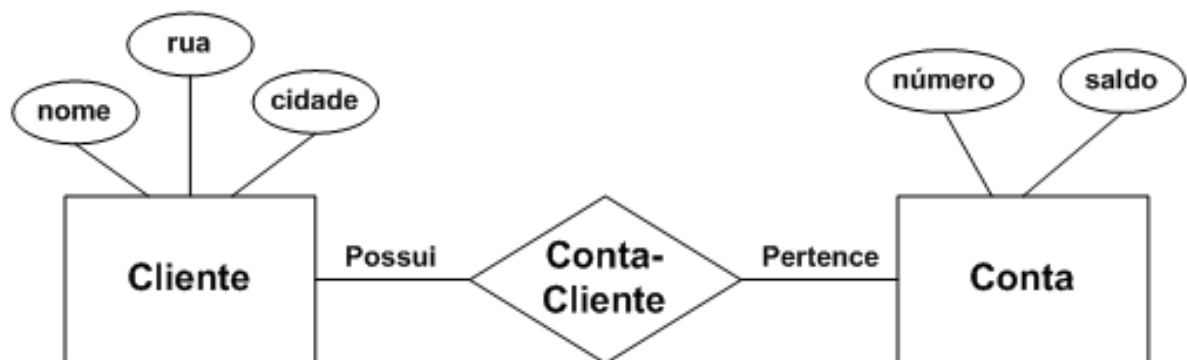
descrever objetos de um software a ser construído relacionando os seus atributos e relacionamentos.

O MER tem como objetivo listar as entidades, atributos e relacionamento trazendo as informações tais como tipos de dados, restrições, descrições de entidade entre outras. E o DER é a representação gráfica associada ao MER ou parte dele.

Os diagramas de entidade e relacionamento são representados por entidades, atributos, relacionamentos, ligação entre atributos a entidades e ligação entre entidades a relacionamentos conforme a figura 3.

- Entidades: São representados por retângulos, descreve algo de importância para um usuário ou organização, como por exemplo: empregados, livros, vendas produtos etc.
- Atributos: São representados por elipses, e descrevem as características da entidade, como por exemplo: fabricante, modelo, placa etc.
- Relacionamentos: São representados por losangos, e descrevem as associações de elementos de uma ou mais entidades.
- Ligação entre atributos a entidades e entidades a relacionamentos: São representados por linhas e representa as ligações entre atributos a entidades e ligação entre entidades a relacionamento.
- Dentro das figuras geométricas são colocados os seus respectivos nomes.

Figura 5 - Diagrama Entidade e Relacionamento



Fonte: Pagina USP⁸

⁸ Disponível em: <https://www.ime.usp.br/~andrers/aulas/bd2005-1/aula6.html> acesso em fev. 2018.

1.3.3.2 Diagrama de Caso de Uso

O diagrama de Caso de Uso é utilizado na UML para auxiliar no processo de levantamento e análise dos requisitos do sistema. É um diagrama mais geral e informal auxiliando a compreensão do comportamento externo do sistema por meio de uma linguagem simples. Ele é utilizado durante todo o processo e ajuda no embasamento de outros diagramas.

Apresenta uma linguagem simples e de fácil compreensão para que os usuários possam ter uma ideia geral de como o sistema irá se comportar. Procura identificar os atores (usuários, outros sistemas ou até mesmo algum hardware especial) que utilizarão de alguma forma o software, bem como os serviços, ou seja, as funcionalidades que o sistema disponibilizará aos atores, conhecidas nesse diagrama como casos de uso.

(Guedes, Gileanes.UML2 uma abordagem prática, 2011, pag. 31)

O diagrama de Caso de Uso é representado por dois itens principais: os atores e casos de uso. E para o relacionamento entre eles é utilizado a associação. A figura 4 ilustra a interação.

- **Atores:** São representados por símbolos de "bonecos magros" e representam diversos papéis dentro do sistema. Pode ser qualquer elemento externo que interaja com o sistema, seja ele um hardware externo ou usuários normais como por exemplo: Gerente, Funcionário, Cliente, Sistema Integrado e etc.
- **Casos de uso:** São representados por elipses que contém em seu interior o texto que descreve sua funcionalidade. Ele é responsável pela captura dos requisitos do sistema como serviços, tarefas e funcionalidades que o software precisa apresentar.
- **Associações:** São representadas por linhas sólidas, que ligam o ator com o caso de uso e vice-versa. Elas representam interações ou relacionamento entre os casos de usos, e atores.

Figura 6 - Diagrama de Caso de Uso



Fonte: Guedes, 2013, p. 58.

1.3.3.3 Diagrama de Classes

O diagrama de classes é um dos diagramas mais importantes e utilizado na UML e serve como base para outros diagramas. Ele define uma estrutura de classes que será implementada pelo sistema compondo-se de atributos e métodos que cada classe tem. Também define como são feitos os relacionamentos entre as classes e como é feita a troca de informações entre elas.

Basicamente, o diagrama de classes é composto por suas classes e pelas associações existentes entre elas, ou seja, os relacionamentos entre as classes. Alguns métodos de desenvolvimento de software, como o Processo Unificado, recomendam que se utilize o diagrama de classes ainda durante a fase de análise, produzindo-se um modelo conceitual a respeito das informações necessárias ao software.

(Guedes, Gileanes.UML2 uma abordagem prática, 2011, pag. 101)

Como dito anteriormente classes tem atributos e métodos (também conhecidos como operações) que armazenam os dados dos objetos das classes. Na representação do diagrama é formado por três divisões, são elas:

- A primeira divisão contém informações sobre a descrição ou nome da classe. Na figura 5 ela está representada por "Aluno" e "Turma".
- A segunda divisão é responsável por guardar os atributos e qual são os tipos de dados respectivos. Na figura 5 o "Aluno" possui o atributo "nome" representado pelo tipo de dados "String" e o atributo "Turma" possui os atributos "código" e "descrição" ambos representados por "String".
- A terceira divisão representa a listagem dos métodos dispostos nas suas respectivas classes. Na figura 5 o "Aluno" tem o método matricular, e Turma

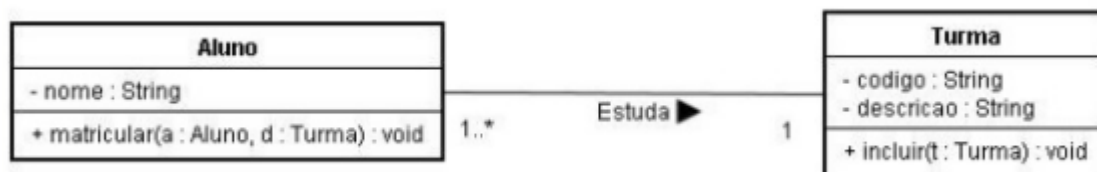
tem o método "incluir". Nele estão destacados os parâmetros esperados e o retorno esperados do método (void significa que o método não terá retorno).

Os símbolos "-" significa que a visibilidade é privada, o "+" significa que a visibilidade é pública e o "#" significa que a visibilidade é protegida dos atributos ou métodos da classe.

E para relacionamento ou associações é feito por uma linha que descreve o vínculo que os objetos de uma classe terão com os outros de uma ou mais classes. Nessa linha contém nome identificando o tipo de vínculo que é estabelecido entre os objetos.

A multiplicidade representada pelos "*", "1" ou "1.." representam os diversos valores de multiplicidade que pode ser utilizado para a associação. Ou seja, no caso da figura 5 o Aluno tem a multiplicidade "1..*" e a Turma 1, então uma turma pode conter um ou mais alunos.

Figura 7 - Diagrama de Classes



Fonte: Guedes, 2013, p. 157.

1.3.3.4 Diagrama de Sequência

O diagrama de sequência é um diagrama representado por uma sequência de processos que será feita no software. Já que na maioria dos projetos pode se ter uma grande quantidade de classes com métodos diferente, fica difícil determinar qual a sequência de um comportamento. O diagrama de sequência trata esta informação de forma simples e lógica.

O diagrama de sequência é um diagrama comportamental que preocupa-se com a ordem temporal em que as mensagens são trocadas entre os objetos envolvidos em um determinado processo. Em geral, baseia-se em um caso de uso definido pelo diagrama de mesmo nome e apoia-se no diagrama de classes para determinar os objetos das classes envolvidas em um processo. (Guedes, Gileanes.UML2 uma abordagem prática, 2011, pag. 33)

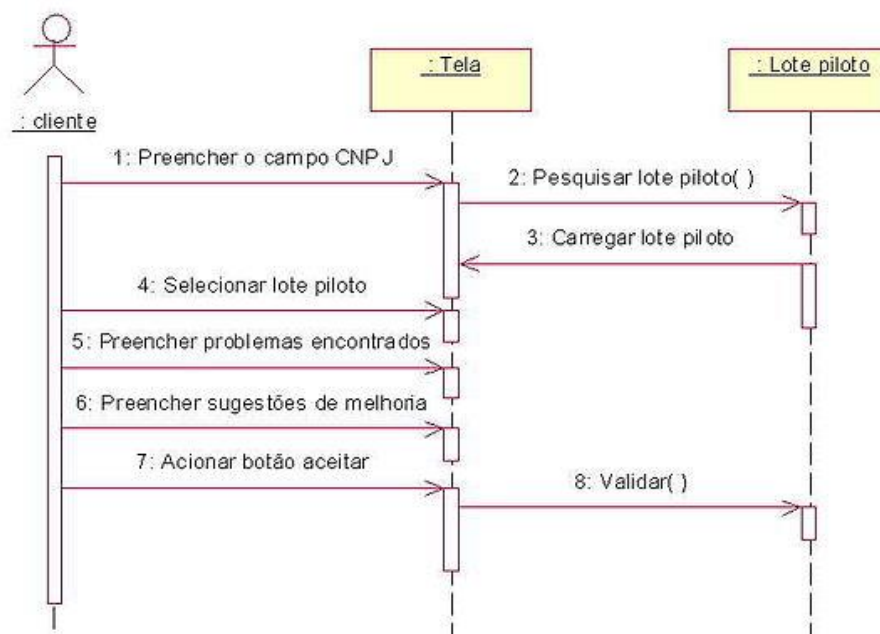
O diagrama de sequência tem como base o diagrama de caso de uso,

cada diagrama de sequência é feito para cada um caso de uso e geralmente um caso de uso é disparado por um ator ao qual tem o intuito de validar o diagrama de classes, pois no processo de criação do diagrama de sequência ajudará a perceber quais são os métodos necessários nas classes.

Os conceitos de criação do diagrama de sequência, são:

- Atores: são representados pelo símbolo de "boneco magro" que representa entidades externas que solicitam serviços, ou seja, ele é responsável pela mensagem inicial.
- Objetos: São representados pelos retângulos que são instancias das classes no processo.
- Linha de vida: ou lifeline são as linhas verticais que representam o tempo de vida de um objeto.

Figura 8 - Diagrama de Sequência



Fonte: Guedes, 2013, p. 205.

1.3.3.5 Diagrama de Atividade

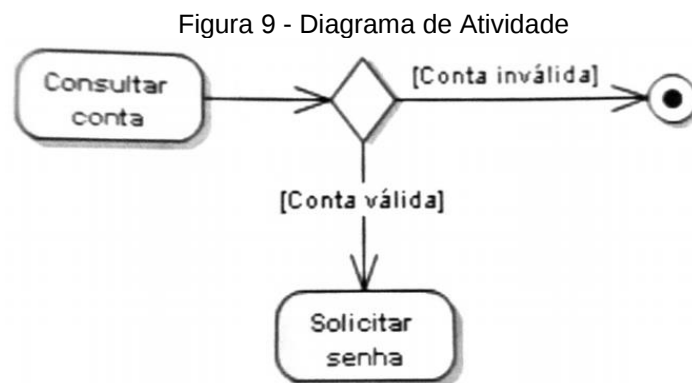
O diagrama de atividade tem como objetivo descrever os passos que serão necessários para a conclusão de uma tarefa. Neste diagrama são listadas todas as atividades passo a passo indicando cada ação a ser tomada dependendo da tomada de decisão até a conclusão da tarefa.

A modelagem de atividade enfatiza a sequência e condições para coordenar comportamentos de baixo nível. Dessa forma, o diagrama de atividade é o diagrama com maior ênfase ao nível de algoritmo da UML e provavelmente um dos mais detalhistas. Este diagrama é utilizado, como o próprio nome diz, para modelar atividades que podem ser um método ou um algoritmo, ou mesmo um processo completo. Atividades podem descrever computação procedural. Neste contexto, elas são os métodos correspondentes às operações sobre classes.

(Guedes, Gileanes.UML2 uma abordagem prática, 2011, pag. 277)

O diagrama de atividade tem como conceito:

- Atividades: representado pelo retângulo grande que representa o comportamento a ser realizado.
- Nó de ação: representado por um pequeno retângulo com as bordas arredondadas que representa um passo que deve ser executado em uma atividade.
- Fluxo de controle: representado por uma linha com seta apontando para o novo nó, e representa o envio de sinais de controle.
- Nó inicial: representado por um círculo preenchido que representa o início do fluxo.
- Nó de Final de atividade: representado por um círculo preenchido dentro de um círculo vazio que representa o fim do fluxo de uma atividade.
- Nó de decisão: representado por um losango e texto entre colchetes para representar escolha entre dois ou mais fluxos possíveis.



Fonte: Guedes, 2013, p. 281.

1.3.4 Recursos e Softwares utilizados

Para auxílio na análise e desenvolvimento do sistema serão utilizados softwares e recursos disponíveis na internet.

1.3.4.1 Astah* Community

O Astah* UML é uma ferramenta completa para desenvolvimento da UML fornecendo vários digramas da UML 2, tais como: diagramas de Classes, Casos de Uso, Sequência, Comunicação, Maquina de Estados, Atividade, Componentes, Implantação e Diagrama de Estrutura Composta, entre outros.

Apesar do Astah* ser uma ferramenta proprietária ela tem a versão Astah* Community que é uma excelente versão grátis para uso não comercial, possibilitando o desenvolvimento dos diagramas⁹. Antigamente conhecida com o JUDE Community, ela é feita na linguagem Java e necessita do Java Runtime Environment (JRE) para funcionar.

Além de ser uma ferramenta de design UML ela também fornece plug-ins denominado validador para comunicação com algumas IDE's de desenvolvimento, que garante que as classes de softwares estejam de acordo com as boas práticas de desenvolvimento estabelecidas pela comunidade científica e profissional. Também tem a possibilidade de exportar códigos Java a partir de modelos predefinidos no software.

Enfim a ferramenta Astah* Community será utilizado para gerar os diagramas que auxiliaram no desenvolvimento deste software fornecendo toda estrutura facilitada para geração dos mesmos.

1.3.4.2 Android Studio

O Android Studio é uma IDE que pode ser instalado em sistemas operacionais Windows da Microsoft, ou Linux, que facilita a programação e fornece ferramentas e recursos para que a tarefa se torne fácil, rápida e ágil¹⁰.

⁹ Disponível em: <http://astah.net/editions/community> acesso em fev. 2018.

¹⁰ Disponível em: <https://developer.android.com/studio/intro/index.html?hl=pt-br> acesso em fev. 2018.

O Android Studio é o ambiente de desenvolvimento integrado (IDE) oficial para o desenvolvimento de aplicativos Android e é baseado no IntelliJ IDEA. Além do editor de código e das ferramentas de desenvolvedor avançados do IntelliJ, o Android Studio oferece ainda mais recursos para aumentar sua produtividade na criação de aplicativos Android, como:

- Um sistema de compilação flexível baseado no Gradle
- Um emulador rápido com inúmeros recursos
- Um ambiente unificado para você poder desenvolver para todos os dispositivos Android
- Instant Run para aplicar alterações a aplicativos em execução sem precisar compilar um novo APK
- Modelos de códigos e integração com GitHub para ajudar a criar recursos comuns dos aplicativos e importar exemplos de código
- Ferramentas e estruturas de teste cheias de possibilidades
- Ferramentas de verificação de código suspeito para detectar problemas de desempenho, usabilidade, compatibilidade com versões e outros
- Compatibilidade com C++ e NDK
- Compatibilidade embutida com o Google Cloud Platform, facilitando a integração do Google Cloud Messaging e do App Engine

(Conheça o Android Studio. Android Studio. Disponível em: <
<https://developer.android.com/studio/intro/index.html?hl=pt-br>> Acesso em: 21 nov.
 2017.)

Apesar do sistema Android ter sido anunciado comercialmente por volta 2008 o Android Studio só foi lançado em 16 de maio de 2013 na conferência Google I/O disponibilizado de forma gratuita pela licença Apache 2.0. Antes da ferramenta ser lançada desenvolvedores Android tinha que ter em seus ambientes de desenvolvimento a ferramenta de IDE Eclipse com o ADT Plugin (sigla em inglês para Android Development Tools) para conseguir desenvolver aplicações para Android. Após o lançamento do Android Studio o desenvolvimento ficou facilitado pois uniu na ferramenta quase todos os recursos necessários para o desenvolvimento.

A linguagem Java é a linguagem utilizada para o desenvolvimento dos aplicativos Android. Porém há algumas particularidades que fazem parte do Android além do que é tratado na linguagem de programação JAVA. Os quatros pilares principais para que ele funcione é uma dessas particularidades formado pela Activity,

Services, Content Providers e Broadcast Receivers.

- Activity: é a representação de uma tela do aplicativo, por exemplo, em um aplicativo de vendas uma Activity é a tela de venda.
- Services: são serviços que operam em segundo plano executando ações sem interromper o aplicativo atual em execução. Por exemplo é usado para buscar algo na internet sem que haja a interação do usuário.
- Content Providers: é o provedor de conteúdo, ele é responsável pela interação de armazenamento de dados seja em banco de dados, na web ou mesmo em arquivos. Ele permite a inclusão, alteração ou exclusão dos dados.
- Broadcast Receivers: são componentes que respondem a estímulos do sistema Android, por exemplo, quando a tela é desligada automaticamente, quando a uma notificação de algum aplicativo, bateria fraca e o envio de estímulo entre aplicativos.

O sistema operacional do Android é baseado em Linux e possibilita o desenvolvimento de uma alta gama de aplicações baseadas em Java principalmente. Também é possível atender outras linguagens como o C++ por meio de adaptações.

1.3.4.3 Java

O JAVA é uma linguagem de orientação a objetos que possui sua própria estrutura, paradigma de programação e regra de sintaxe. Ela é baseada na POO (sigla para programação orientada a objetos), que atualmente é o paradigma mais difundido entre os desenvolvedores. A POO é um padrão e sua estrutura de código é aconselhada pelas boas práticas de desenvolvimento estabelecidas por uma comunidade científica e profissional, que visa principalmente para questões como segurança e reaproveitamento de código.

Desenvolvida pela empresa Sun incorporada pela a Oracle pela compra da Sun em abril de 2009. Segundo pesquisa realizada pelo site tiobe.com em outubro de 2017 a linguagem de programação Java está em primeiro no top 10 de linguagens de programação mais populares, conforme mostra a figura 8. Ela é tão popular por sua praticidade, retro compatibilidade, escalabilidade, performance, confiabilidade e frescor com que a Oracle a mantém. Empresas como Facebook, eBay, Twitter, Spotify, Android e Oracle a utilizam, pois proporciona escalabilidade e performance.

O Java é uma linguagem compilada, ou seja, os códigos fontes são arquivos ".java" que serão compilados e gerados bytecodes com extensões ".class" que serão executados. Os bytecodes são instruções que serão executadas em uma Java Virtual Machine (JVM)¹¹.

Figura 10 - Ranking das linguagens de Outubro de 2017

Oct 2017	Oct 2016	Change	Programming Language	Ratings	Change
1	1		Java	12.431%	-6.37%
2	2		C	8.374%	-1.46%
3	3		C++	5.007%	-0.79%
4	4		C#	3.858%	-0.51%
5	5		Python	3.803%	+0.03%
6	6		JavaScript	3.010%	+0.26%
7	7		PHP	2.790%	+0.05%
8	8		Visual Basic .NET	2.735%	+0.08%
9	11	▲	Assembly language	2.374%	+0.14%
10	13	▲	Ruby	2.324%	+0.32%

Fonte: Pagina TIOBE indes top 10.

A JVM é uma máquina virtual do JAVA que é a plataforma responsável por interpretar os bytecodes do java. Ela é multiplataforma podendo ser instalados em vários sistemas operacionais tais como Windows, Linux e MACOSX, Solaris tornando-a flexível já que um código gerado em um sistema operacional poderá ser utilizado em outro sem problema.

Para desenvolvimento em Java é necessário o kit de desenvolvimento em Java intitulado como JDK (sigla em inglês para Java Development Kit) que fornece ao desenvolvedor além do compilador outras ferramentas, tais como bibliotecas completas de utilitários que auxilia na construção do aplicativo. Para execução dos arquivos compilados é necessário ter o ambiente de execução Java o JRE (sigla em inglês para Java Runtime Environment, também conhecido como tempo de execução Java) que inclui bibliotecas de códigos e componentes que são necessários para executar programas em Java, que como dito está disponível para diversas plataformas.

¹¹ Disponível em: <https://www.ibm.com/developerworks/br/java/tutorials/j-introtojava1/index.html>
Acesso em fev. 2018.

1.3.4.4 Firebase

O Google Firebase é um serviço em nuvem do Google, adquirido em 2014 de um startup fundada por James Tamplin e Andrew Lee em 2011. Ela prove API de integração com sistemas online fornecendo interação em tempo real com os dados armazenados em qualquer parte do mundo ou plataforma. A ideia inicial dos desenvolvedores era de fazer um sistema de bate papo integrado com outros sistemas, porém começaram a perceber que estavam sendo passados outros dados em tempo real para a plataforma, o que não era o objetivo inicial da aplicação, então decidiram separar a ideia de bate papo e criar o Firebase em abril de 2012.

Firebase é um back-end completo para aplicações mobile e web. Traz um visual simples de usar e limpo com suporte a várias linguagens de programação tais como C++, Java, Javascript, Node.js, Objective-C e Swift. Incorporam várias funcionalidades em seu ambiente tais como AdMod, AdWords, Analytics, Autenticação, Relatórios de Erros, Database, Dynamic Links, Hosting, Indexing, Invites, Messaging, Notifications, Offline, Real time, Remote Config, Storage, Synchronization e TestLab.

Dois recursos muito utilizados da plataforma é o FirebaseAuth (serviço de autenticação) e o Database (Serviço de banco de dados). O FirebaseAuth é o serviço que prove um sistema de autenticação de usuários via e-mail, Facebook, GitHub, Google Sign e Twiter. O Database é um banco de dados NoSQL utilizado para armazenar dados no formato JSON.

O site oficial é o firebase.google.com disponibiliza a utilização de forma gratuita da ferramenta com certas restrições¹².

1.3.4.5 NoSQL

NoSQL é um termo usado para descrever bancos de dados não relacionais de alto desempenho. Os bancos de dados NoSQL usam diversos modelos de dados, incluindo documentos, gráficos, chave-valor e colunares. Bancos de dados NoSQL são amplamente reconhecidos pela facilidade de desenvolvimento, desempenho escalável, alta disponibilidade e resiliência.

¹² Disponível em: <https://firebase.google.com/docs/database/?hl=pt-br> acesso em fev. 2018.

(O que é NoSQL. Disponível em: < <https://aws.amazon.com/pt/nosql/>> Acesso em: 21 nov. 2017.)

O NoSql é um banco de dados não relacional que vem se tornando muito usual nos softwares atuais principalmente mobile e web. Google e AWS são empresas que investem e Netflix é uma das grandes empresas que usa o NoSQL. Ele tem uma proposta diferente do banco de dados relacional que traz uma linguagem estruturada de consulta SQL (sigla em inglês para Structured Query Language) para suprir necessidades como alta performance, replicação, escalabilidade, suporte à dados estruturados e sub colunas.

Em banco de dados NoSql as informações estão agrupadas em um mesmo registro e com isso é descartado a estrutura relacional deixando todas as informações em um mesmo registro, fornecendo consulta mais rápidas do que banco de dados relacionais, além disso é tolerante a falhas.

Bancos de dados relacionais escalam, mas quanto maior o tamanho, mais custoso se torna essa escalabilidade, seja pelo custo de novas máquinas, seja pelo aumento de especialistas nos bancos de dados utilizados.

Já os não relacionais, permitem uma escalabilidade mais barata e menos trabalhosa, pois não exigem máquinas extremamente poderosas e sua facilidade de manutenção permite que um número menor de profissionais seja necessário.

(Introdução aos bancos de dados NoSQL. Disponível em: <<http://www.devmedia.com.br/introducao-aos-bancos-de-dados-nosql/26044>>

Acesso em: 21 nov. 2017.)

O NoSQL utiliza o formato JSON (JavaScript Object Notation) que é um modelo para armazenamento e transmissão de informações por arquivo texto e tem como principal benefício a capacidade de estruturar informações de forma compacta do que os outros modelos.

A ideia utilizada pelo JSON para representar informações é tremendamente simples: para cada valor representado, atribui-se um nome (ou rótulo) que descreve o seu significado. Esta sintaxe é derivada da forma utilizada pelo JavaScript para representar informações. Por exemplo, para representar o ano de 2012, utiliza-se a seguinte sintaxe:

- "ano": 2012

(Introdução ao formato JSON. Disponível em: <<http://www.devmedia.com.br/introducao-ao-formato-json/25275>> Acesso em: 21 nov. 2017.)

1.3.5 Requisitos para instalação dos Softwares utilizados

Nesta sessão foi descrito todos os requisitos e procedimentos de instalação dos softwares utilizado para a análise e desenvolvimento do sistema.

1.3.5.1 Astah* Community

Os requisitos para instalação do Astah* são:

Windows Requirements

Windows 10, 8.1 and 7 (32bit, 64bit); 1 GHz or better; 2GB RAM (4GB recommended); 400MB Drive Space; Oracle Java 8 (Update 144) or later

Mac Requirements

Mac OS x 10.12.6 or later; Intel Processor; 2GB RAM (4GB recommended); 400 MB Drive Space; Oracle Java 8 (Update 144) or later

Linux Requirements

Ubuntu 16.04, CentOS 7(1611), Oracle Java 8 (Update 144) or later; Desktop Environment:

(System Requirements. Disponível em: <<http://astah.net/requirements>> Acesso em: 23 nov. 2017.)

No projeto foi utilizado sistema Windows e para instalação foi seguindo os seguintes passos:

1. Criar uma conta no sistema da ChangeVision pelo link (https://members.change-vision.com/members/files/astah_community/6_9_0) de cadastro.
2. Logar no sistema da ChangeVision pelo link (https://members.change-vision.com/members/files/astah_community/6_9_0) de login.
3. Fazer download da última versão do sistema Astah*. No projeto foi utilizada a

versão astah-community-6_9_0-b4c6e9-jre-setup.exe do site
(https://members.change-vision.com/members/files/astah_community/6_9_0).

4. Execute o executável feito o download, e siga o processo de instalação.
5. Após a finalização da instalação o programa já estará pronto para o uso.

A versão Astah* Community é uma versão gratuita para uso não comercial da ferramenta.

1.3.5.2 Android Studio

Os requisitos de instalação do Android Studio, são:

Windows

Microsoft® Windows® 7/8/10 (32 ou 64 bits)

- Mínimo de 3 GB de RAM, 8 GB de RAM recomendados, mais 1 GB para o Android Emulador
- Mínimo de 2 GB de espaço livre em disco,
- 4 GB recomendados (500 MB para o IDE + 1,5 GB para o Android SDK e as imagens do sistema do emulador)
- Resolução de tela mínima de 1.280 x 800
- Para o emulador acelerado: Sistema operacional de 64 bits e processador Intel® compatível com Intel® VT-x, Intel® EM64T (Intel® 64) e a funcionalidade Execute Disable (XD) Bit

Mac

- Mac® OS X® 10.10 (Yosemite) ou posterior, até a versão 10.12 (macOS Sierra)
- Mínimo de 3 GB de RAM, 8 GB de RAM recomendados, mais 1 GB para o Android Emulator
- Mínimo de 2 GB de espaço livre em disco,
- 4 GB recomendados (500 MB para o IDE + 1,5 GB para o Android SDK e as imagens do sistema do emulador)
- Resolução de tela mínima de 1.280 x 800

Linux

- Área de trabalho GNOME ou KDE

- Testado no Ubuntu® 12.04, Precise Pangolin (distribuição de 64 bits capaz de executar aplicativos de 32 bits)
- Distribuição de 64 bits capaz de executar aplicativos de 32 bits
- Biblioteca C do GNU (glibc) 2.19 ou posterior
- Mínimo de 3 GB de RAM, 8 GB de RAM recomendados, mais 1 GB para o Android Emulator
- Mínimo de 2 GB de espaço livre em disco,
- 4 GB recomendados (500 MB para o IDE + 1,5 GB para o Android SDK e as imagens do sistema do emulador)
- Resolução de tela mínima de 1.280 x 800
- Para o emulador acelerado: Processador Intel® compatível com Intel® VT-x, Intel® EM64T (Intel® 64) e a funcionalidade Execute Disable (XD) Bit, ou processador AMD compatível com AMD Virtualization™ (AMD-V™)

(Android Studio O IDE oficial do Android. Disponível em: <
<https://developer.android.com/studio/index.html?hl=pt-br>> Acesso em: 23 nov. 2017.
)

No projeto foi utilizado sistema Windows e para instalação foi seguindo os seguintes passos:

1. Fazer o download da versão 2.1.3 do Android Studio (<https://dl.google.com/dl/android/studio/install/2.1.3.0/android-studio-bundle-143.3101438-windows.exe>) com o SDK incluído que é necessário para execução do Android.
2. Após o download, e executar o executável feito o download e seguir os passos de instalação passado pelo programa.
3. A primeira execução do aplicativo pedirá as configurações, escolha a "Standard", e ele irá atualizar.
4. Após atualizado ele já estará pronto para execução.

Android Studio é uma IDE gratuita oferecida pelo Google.

1.3.5.3 Firebase

No projeto foi utilizado dois recursos do Firebase que são:

- Database: que fornece uma estrutura de banco de dados NoSQL
- Authentication: que fornece uma estrutura de autenticação por e-mail, Facebook, GitHub, Google Sign e Twiter.

Para utilização do Firebase foram seguidos os seguintes passos.

1. Criar uma conta no Google para utilização no Firebase pelo link (<https://accounts.google.com/SignUp?hl=pt-BR>)
2. Fazer login no Firebase usando uma conta do Google pelo link (<https://firebase.google.com/>)
3. Criar um novo projeto no Firebase console pelo link (<https://console.firebase.google.com/u/0/>)
4. Clicar para "Adicionar o Firebase ao seu aplicativo Android"
5. Coloque o "Nome do pacote do Android"
6. Para tirar o "Certificado de assinatura de depuração SHA-1" entre no seu projeto do Android Studio e clique em Gradle (no canto direito da IDE) na guia FirebaseApp, Tasks, android clique em signingReport e ira aparecer no console.
7. Será gerado um arquivo "google-service.json" que terá de ser incorporado ao projeto na pasta "app".

No projeto foi utilizado o plano Spark do Firebase que é um plano gratuito que disponibiliza os recursos necessário para desenvolvimento do Software.

2 DESENVOLVIMENTO DO SISTEMA

A análise do projeto foi baseada na análise e levantamento de requisitos para criar os recursos e atingir o objetivo do desenvolvimento do sistema. Será apresentada o estudo realizado para criação do sistema de Pedidos via Android, que tem por objetivo de facilitar a venda dos vendedores externos.

2.1 Visão geral do sistema

O sistema de pedidos via Android é um facilitador e controlador de informações de pedidos que traz um sistema de pedidos que é disponibilizado para a plataforma de celular móvel com sistema operacional Android. Ela tem recursos como consulta de clientes, consulta de produtos e pedidos via o aplicativo. Com isso o vendedor externo terá a facilidade de livrar-se dos catálogos de produtos e blocos de pedidos, podendo usar somente o dispositivo com o sistema instalado.

A principal funcionalidade do sistema é de disponibilizar acesso online e móvel a plataforma. Pois proporciona consulta do banco de dados via nuvem, que possibilita o acesso de qualquer lugar que disponibilize acesso à internet. E é disponibilizada via plataforma mobile que fornece mobilidade para o sistema podendo ser feita o registro do pedido em qualquer parte de onde esteja.

Pode ser instalado em qualquer dispositivo com sistema operacional Android, ou seja, se for da vontade do usuário utilizar o sistema em seu "SmartPhone" pessoal, ele terá o sistema 24 horas no bolso podendo acessá-lo para consulta ou vendas a qualquer momento.

O sistema disponibiliza cadastros de seus clientes facilitando o trabalho de reunir informações do cliente para fazer o registro de pedidos. Também conta com a lista de produtos que facilitará na consulta de preços. E por fim fornece o registro de pedidos possibilitando vincular um ou vários produtos para o cliente.

O sistema de pedidos via Android é um sistema simples e prático que traz como benefício a facilidade na utilização com menus de fácil acesso e intuitivos. Na figura 11 mostra a facilidade na usabilidade do sistema por acesso a quatro menus principais. O menu “Deslogar” que levará o usuário até a tela de login possibilitando a troca ou criação de usuário sem dificuldades. O menu “Produto” que oferece o gerenciamento dos produtos do sistema como visualizar, cadastrar, alterar e excluir,

possibilitando o usuário ter acesso aos dados como código, nome e valor do produto. O menu “Cliente” abre o gerenciamento de clientes oferece recursos de visualizar, cadastrar, alterar e excluir clientes possibilitando aos usuários acesso a dados como nome, CPF ou CNPJ, idade, dois telefones, e-mail, rua, número, bairro, cidade, estado e país do cliente. No menu de “Pedido” o sistema tem telas com recursos de visualizar, cadastrar, alterar e excluir pedidos.

Figura 11 - Tela menu SPA

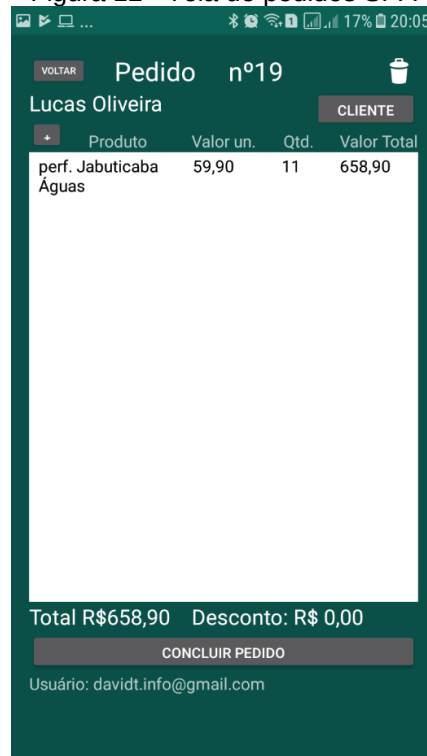


Fonte: Elaborado pelo autor

A tela de pedidos do Sistema de Pedidos via Android concentra todas as informações necessárias que o usuário precisa para fazer um pedido. Nela contém o número do pedido que é de forma incremental no sistema, o vínculo de cliente, os produtos vinculados que pode ser vários, o valor total, o desconto que o usuário concede no valor total, a conclusão do pedido e por fim o usuário que efetuou o registro. Além de possibilitar a alteração ou exclusão de cliente, produtos já vinculados no registro de pedido. Para todas as funcionalidades citadas basta apenas o usuário clicar em cima dos botões ou informações que precisa alterar ou incluir no pedido. O ícone de lixeira permite excluir o pedido por completo. Todas as informações do pedido serão mantidas mesmo se houver exclusão de cliente e produtos, pois o pedido

é vinculado no banco de dados quando é efetuada.

Figura 12 - Tela de pedidos SPA



Fonte: Elaborado pelo autor

O SPA tem uma abordagem de fácil utilização com o intuito de entregar uma experiência intuitiva e agradável ao usuário, oferecendo recursos simples e robustos para ajudar-lhes no seu cotidiano de trabalho.

2.2 Levantamento de Requisitos

Para levantamento de requisitos foi observado o cotidiano do vendedor de porta a porta além de softwares de registros de pedidos, com o intuito de substituir o catálogo de produtos e o bloco de pedidos da rotina de trabalho de forma natural. Foi levantado requisitos funcionais (RF) e requisitos não-funcionais (RNF) a partir desta análise.

2.3.1 Requisito Cadastrar usuário

Tabela 1 - Requisito Funcional – Cadastrar Usuário

RF1 CADASTRAR USUÁRIO			OCULTO ()	
DESCRIÇÃO: O SISTEMA DEVE CADASTRAR USUÁRIOS PARA ACESSO AO SISTEMA				
REQUISITOS NÃO-FUNCIONAIS				
NOME	RESTRIÇÃO	CATEGORIA	DESEJÁVEL	PERMANENTEMENTE
NF1.1 CADASTRO	DEVERÁ PREENCHER OS CAMPOS OBRIGATORIOS	INTERFACE	(X)	(X)
NF1.2 JANELA ÚNICA	AS INFORMAÇÕES DE CADASTRO DEVEM CONSTAR EM UMA JANELA UNIFICADA	INTERFACE	(X)	(X)

2.3.2 Requisito Logar no sistema

Tabela 2 - Requisito Funcional – Logar no sistema

RF2 LOGAR NO SISTEMA			OCULTO ()	
DESCRIÇÃO: O SISTEMA DEVE TER A FUNCIONALIDADE DE ACESSO AO SISTEMA POR USUÁRIO CADASTRADO				
REQUISITOS NÃO-FUNCIONAIS				
NOME	RESTRIÇÃO	CATEGORIA	DESEJÁVEL	PERMANENTEMENTE
NF2.1 ACESSO	O USUÁRIO CADASTRADO TERÁ ACESSO AO SISTEMA	SEGURANÇA	(X)	(X)
NF2.2 LOGAR	DEVERA INFORMAR EMAIL E SENHA OBRIGATORIAMENTE	INTEREFACE	(X)	(X)
NF3.3 ÚNICA JANELA	AS INFORMAÇÕES DE CADASTRO DEVEM CONSTAR EM UMA JANELA UNIFICADA	INTERFACE	(X)	(X)

2.3.3 Requisito Deslogar do sistema

Tabela 3 - Requisito Funcional – Deslogar do sistema

RF3 DESLOGAR DO SISTEMA			OCULTO ()	
DESCRIÇÃO: O SISTEMA DEVE TER A FUNCIONALIDADE DE DESLOGAR USUÁRIO LOGADO				
REQUISITOS NÃO-FUNCIONAIS				
NOME	RESTRIÇÃO	CATEGORIA	DESEJÁVEL	PERMANENTEMENTE
NF3.1 DESLOGAR	O USUÁRIO DEVE ESTAR LOGADO PARA DESLOGAR DO SISTEMA	INTERFACE	(X)	(X)

2.3.4 Requisito Cadastrar cliente

Tabela 4 - Requisito Funcional – Cadastrar Cliente

RF4 CADASTRAR CLIENTE			OCULTO ()	
DESCRIÇÃO: O SISTEMA DEVE CADASTRAR NOVO CLIENTE				
REQUISITOS NÃO-FUNCIONAIS				
NOME	RESTRIÇÃO	CATEGORIA	DESEJÁVEL	PERMANENTEMENTE
NF4.1 CADASTRO	DEVE PREENCHER OS CAMPOS OBRIGATORIOS	INTERFACE	(X)	(X)
NF4.2 DUPLICIDADE	NÃO DEVE CONTER CLIENTE EM DUPLICIDADE CADASTRADO NO SISTEMA	SEGURANÇA	(X)	(X)
NF4.3 ÚNICA JANELA	AS INFORMAÇÕES DE CADASTRO DEVEM CONSTAR EM UMA JANELA UNIFICADA	INTERFACE	(X)	(X)

2.3.5 Requisito Alterar cliente

Tabela 5 - Requisito Funcional – Alterar Cliente

RF5 ALTERAR CLIENTE			OCULTO ()	
DESCRIÇÃO: O SISTEMA DEVE ALTERAR CLIENTE EXISTENTE				
REQUISITOS NÃO-FUNCIONAIS				
NOME	RESTRIÇÃO	CATEGORIA	DESEJÁVEL	PERMANENTEMENTE
NF5.1 IDENTIFICADOR	DEVERÁ PRESERVAR O IDENTIFICADOR DO CLIENTE	SEGURANÇA	(X)	(X)
NF5.2 ÚNICA JANELA	AS INFORMAÇÕES DE CADASTRO DEVEM CONSTAR EM UMA JANELA UNIFICADA	INTERFACE	(X)	(X)

2.3.6 Requisito Excluir cliente

Tabela 6 - Requisito Funcional – Excluir Cliente

RF6 EXCLUIR CLIENTE			OCULTO ()	
DESCRIÇÃO: O SISTEMA DEVE EXCLUIR CLIENTE EXISTENTE				
REQUISITOS NÃO-FUNCIONAIS				
NOME	RESTRIÇÃO	CATEGORIA	DESEJÁVEL	PERMANENTEMENTE
NF6.1 EXCLUSÃO	A EXCLUSÃO PODE SER FEITA SE O CLIENTE NÃO TIVER VENDA VINCULADA	SEGURANÇA	(X)	(X)
NF6.2 ÚNICA JANELA	AS INFORMAÇÕES DE CADASTRO DEVEM CONSTAR EM UMA JANELA UNIFICADA	INTERFACE	(X)	(X)

2.3.7 Requisito Consultar cliente

Tabela 7 - Requisito Funcional – Consultar Cliente

RF7 CONSULTAR CLIENTE				OCULTO ()	
DESCRIÇÃO: O SISTEME DEVE TER FUNCIONALIDADE DE CONSULTAR CLIENTES CADASTRADOS					
REQUISITOS NÃO-FUNCIONAIS					
NOME	RESTRIÇÃO		CATEGORIA	DESEJÁVEL	PERMANENTEMENTE
NF7.1 CONSULTA	O CLIENTE DEVE SER CONSULTADO PELO SEU NOME		INTERFACE	(X)	(X)
NF7.2 ÚNICA JANELA	AS INFORMAÇÕES DE CADASTRO DEVEM CONSTAR EM UMA JANELA UNIFICADA		INTERFACE	(X)	(X)

2.3.8 Requisito Cadastrar produto

Tabela 8 - Requisito Funcional – Cadastrar Produto

RF8 CADASTRAR PRODUTO			OCULTO ()	
DESCRIÇÃO: O SISTEMA DEVE CADASTRAR NOVO PRODUTO				
REQUISITOS NÃO-FUNCIONAIS				
NOME	RESTRIÇÃO	CATEGORIA	DESEJÁVEL	PERMANENTEMENTE
NF8.1 CADASTRO	DEVERÁ PREENCHER OS CAMPOS OBRIGATORIOS	INTERFACE	(X)	(X)
NF8.4 DUPLICIDADE	NÃO DEVE CONTER PRODUTO EM DUPLICIDADE CADASTRADO NO SISTEMA	SEGURANÇA	(X)	(X)
NF8.3 ÚNICA JANELA	AS INFORMAÇÕES DE CADASTRO DEVEM CONSTAR EM UMA JANELA UNIFICADA	INTERFACE	(X)	(X)

2.3.9 Requisito Alterar produto

Tabela 9 - Requisito Funcional – Alterar Produto

RF9 ALTERAR PRODUTO			OCULTO ()	
DESCRIÇÃO: O SISTEMA DEVE ALTERAR PRODUTO CADASTRADO				
REQUISITOS NÃO-FUNCIONAIS				
NOME	RESTRIÇÃO	CATEGORIA	DESEJÁVEL	PERMANENTEMENTE
NF9.1 IDENTIFICADOR	DEVERÁ PRESERVAR O IDENTIFICADOR DO PRODUTO	SEGURANÇA	(X)	(X)
NF9.2 ÚNICA JANELA	AS INFORMAÇÕES DE CADASTRO DEVEM CONSTAR EM UMA JANELA UNIFICADA	INTERFACE	(X)	(X)

2.3.10 Requisito Excluir produto

Tabela 10 - Requisito Funcional – Excluir Produto

RF10 EXCLUIR PRODUTO			OCULTO ()	
DESCRIÇÃO: O SISTEMA DEVE EXCLUIR PRODUTO CADASTRADO				
REQUISITOS NÃO-FUNCIONAIS				
NOME	RESTRIÇÃO	CATEGORIA	DESEJÁVEL	PERMANENTEMENTE
NF10.1 EXCLUSÃO	A EXCLUSÃO PODE SER FEITA SE O PRODUTO NÃO TIVER VENDA VINCULADA	SEGURANÇA	(X)	(X)
NF10.2 ÚNICA JANELA	AS INFORMAÇÕES DE CADASTRO DEVEM CONSTAR EM UMA JANELA UNIFICADA	INTERFACE	(X)	(X)

2.3.11 Requisito Consultar produto

Tabela 11 - Requisito Funcional – Consultar Produto

RF11 CONSULTAR PRODUTO			OCULTO ()		
DESCRIÇÃO: O SISTEMA DEVE CONSULTAR PRODUTOS CADASTRADOS					
REQUISITOS NÃO-FUNCIONAIS					
NOME	RESTRIÇÃO		CATEGORIA	DESEJÁVEL	PERMANENTEMENTE
NF11.1	O PRODUTO DEVE SER CONSULTADO PELO SEU CÓDIGO		INTERFACE	(X)	(X)
NF11.2 ÚNICA JANELA	AS INFORMAÇÕES DE CADASTRO DEVEM CONSTAR EM UMA JANELA UNIFICADA		INTERFACE	(X)	(X)

2.3.12 Requisito Cadastrar pedido

Tabela 12 - Requisito Funcional – Cadastrar Pedido

RF12 CADASTRAR PEDIDO			OCULTO ()	
DESCRIÇÃO: O SISTEMA DEVE CADASTRAR NOVO PEDIDO				
REQUISITOS NÃO-FUNCIONAIS				
NOME	RESTRIÇÃO	CATEGORIA	DESEJÁVEL	PERMANENTEMENTE
NF12.1 CADASTRO	DEVE PREENCHER OS CAMPOS OBRIGATORIOS	INTERFACE	(X)	(X)
NF12.2 CLIENTE	O PEDIDO DEVE CONTER UM CLIENTE	INTERFACE	(X)	(X)
NF12.3 PRODUTO	O PEDIDO DEVE CONTER UM OU MAIS PRODUTOS	INTERFACE	(X)	(X)
NF12.4 IDENTIFICADOR	DEVE SER GERADO AUTOMATICAMENTE IDENTIFICADOR NUMERAL CO DE FORMA CRESCENTE	INTERFACE	(X)	(X)
NF12.5 ÚNICA JANELA	AS INFORMAÇÕES DE CADASTRO DEVEM CONSTAR EM UMA JANELA UNIFICADA	INTERFACE	(X)	(X)

2.3.13 Requisito Alterar pedido

Tabela 13 - Requisito Funcional – Alterar Pedido

RF13 ALTERAR PEDIDO			OCULTO ()	
DESCRIÇÃO: O SISTEMA DEVE ALTERAR PEDIDO CADASTRADO				
REQUISITOS NÃO-FUNCIONAIS				
NOME	RESTRIÇÃO	CATEGORIA	DESEJÁVEL	PERMANENTEMENTE
NF13.1 IDENTIFICADOR	DEVE PRESERVAR O IDENTIFICADOR DO PEDIDO	SEGURANÇA	(X)	(X)
NF13.2 ÚNICA JANELA	AS INFORMAÇÕES DE CADASTRO DEVEM CONSTAR EM UMA JANELA UNIFICADA	INTERFACE	(X)	(X)

2.3.14 Requisito Excluir pedido

Tabela 14 - Requisito Funcional – Excluir Pedido

RF14 EXCLUIR PEDIDO			OCULTO ()		
DESCRIÇÃO: O SISTEMA DEVE EXCLUIR PEDIDO CADASTRADO					
REQUISITOS NÃO-FUNCIONAIS					
NOME	RESTRIÇÃO	CATEGORIA	DESEJÁVEL	PERMANENTEMENTE	
NF14.2 ÚNICA JANELA	AS INFORMAÇÕES DE CADASTRO DEVEM CONSTAR EM UMA JANELA UNIFICADA	INTERFACE	(X)	(X)	

2.3.15 Requisito Consultar pedido

Tabela 15 - Requisito Funcional – Consultar Pedido

RF15 CONSULTAR PEDIDO				OCULTO ()	
DESCRIÇÃO: O SISTEMA DEVE CONSULTAR PEDIDOS CADASTRADOS					
REQUISITOS NÃO-FUNCIONAIS					
NOME	RESTRIÇÃO		CATEGORIA	DESEJÁVEL	PERMANENTEMENTE
NF15.1	O PEDIDO DEVE SER CONSULTADO PELO SEU NUMERO		INTERFACE	(X)	(X)
NF15.2 ÚNICA JANELA	AS INFORMAÇÕES DE CADASTRO DEVEM CONSTAR EM UMA JANELA UNIFICADA		INTERFACE	(X)	(X)

2.4 Casos de uso

Os casos de usos são importantes para levantamento dos processos que terá no sistema, descrevendo os comportamentos e funcionalidades.

2.4.1 Casos de uso do sistema

Tabela 16 - Casos de Uso do Sistema de Vendas via Android

CASOS DE USO			
NOME	ATORES	DESCRIÇÃO	REF. CRUZADA
CADASTRAR USUÁRIO	USUÁRIO	NA TELA INICIAL DO SISTEMA TERÁ UM LINK PARA O FORMULARIO DE CADASTRO DE NOVO USUARIO.	RF1
LOGAR NO SISTEMA	USUÁRIO	NA TELA INICIAL TERÁ A FUNÇÃO DO USUARIO LOGAR COM USUÁRIO E SENHA CADASTRADO NO SISTEMA	RF2, RF1
DESLOGAR DO SISTEMA	USUÁRIO	O USUÁRIO LOGADO NO SISTEMA TEM A FUNCIONALIDADE DE DESLOGAR DO SISTEMA	RF3, RF2
CADASTRAR CLIENTE	USUÁRIO	APÓS LOGIN FEITO NO SISTEMA O USUÁRIO IRÁ ABRIR O FORMULARIO DE CADASTRO DE CLIENTE E INCLUIR NOVO CLIENTE	RF4
ALTERAR CLIENTE	USUÁRIO	ÁPOS LOGIN FEITO NO SISTEMA O USUÁRIO IRÁ ABRIR FORMULARIO PARA ALTERAÇÃO DE CLIENTE JÁ EXISTENTE	RF5, RF4
EXCLUIR CLIENTE	USUÁRIO	ÁPOS LOGIN FEITO NO SISTEMA O USUÁRIO IRÁ ABRIR FORMULARIO PARA EXCLUSÃO DE CLIENTE JÁ EXISTENTE	RF6, RF4
CONSULTAR CLIENTE	USUÁRIO	ÁPOS LOGIN FEITO NO SISTEMA O USUÁRIO IRÁ ABRIR CONSULTA DE CLIENTES JÁ CADASTRADOS NO SISTEMA	RF7, RF4

CADASTRAR PRODUTO	USUÁRIO	ÁPOS LOGIN FEITO NO SISTEMA O USUÁRIO IRÁ ABRIR O FORMULARIO DE CADASTRO DE PRODUTO E INCLUIR NOVO PRODUTO	RF8
ALTERAR PRODUTO	USUÁRIO	ÁPOS LOGIN FEITO NO SISTEMA O USUÁRIO IRÁ ABRIR FORMULARIO PARA ALTERAÇÃO DE PRODUTO JÁ EXISTENTE	RF9, RF8
EXCLUIR PRODUTO	USUÁRIO	ÁPOS LOGIN FEITO NO SISTEMA O USUÁRIO IRÁ ABRIR FORMULARIO PARA EXCLUSÃO DE PRODUTO JÁ EXISTENTE	RF10, RF8
CONSULTAR PRODUTO	USUÁRIO	ÁPOS LOGIN FEITO NO SISTEMA O USUÁRIO IRÁ ABRIR CONSULTA DE PRODUTOS CADASTRADOS NO SISTEMA	RF11, RF8
CADASTRAR PEDIDO	USUÁRIO	APÓS LOGIN FEITO NO SISTEMA O USUÁRIO IRÁ ABRIR O FORMULARIO DE CADASTRO DE PEDIDO, VINCULAR UM CLIENTE E UM OU VARIOS PRODUTOS, E INCLUIR NOVO PEDIDO	RF12, RF4, RF8
ALTERAR PEDIDO	USUÁRIO	ÁPOS LOGIN FEITO NO SISTEMA O USUÁRIO IRÁ ABRIR FORMULARIO PARA ALTERAÇÃO DE PEDIDO JÁ EXISTENTE	RF13, RF12
EXCLUIR PEDIDO	USUÁRIO	ÁPOS LOGIN FEITO NO SISTEMA O USUÁRIO IRÁ ABRIR FORMULARIO PARA EXCLUSÃO DE PEDIDO JÁ EXISTENTE	RF14, RF12
CONSULTAR PEDIDO	USUÁRIO	ÁPOS LOGIN FEITO NO SISTEMA O USUÁRIO IRÁ ABRIR CONSULTA DE PEDIDOS CADASTRADOS NO SISTEMA	RF15, RF12

Fonte: Elaborado pelo autor

2.4.2 Expansão dos casos de uso do sistema

A expansão de casos de uso tem como objetivo a explicação mais detalhada das diferentes funcionalidades que o sistema terá, permitindo interferir e identificar com mais clareza em outras necessidades.

2.4.2.1 Expansão do caso de uso "Cadastrar Usuário"

Tabela 17 - Expansão do Caso de Uso Cadastrar Usuário

CASO DE USO: CADASTRAR USUÁRIO	
FLUXO PRINCIPAL	TRATAMENTO DE EXCEÇÕES
1- O USUÁRIO DEVE SELECIONAR O LINK DE CADASTRO, NA TELA DE LOGIN 2- DEVE PREENCHER AS INFORMAÇÕES 3- DEVE PREENCHER TODOS OS CAMPOS OBRIGATORIOS 4- DEVE CADASTRAR CLICANDO NO BOTÃO CADASTRAR 5- O LOGIN SERÁ AUTOMATICAMENTE FORNECIDO	1.a – CAMPOS OBRIGATORIOS NÃO PREENCHIDOS, OU INVÁLIDOS 1.a.1 – JANELA INFORMANDO O ERRO AO USUÁRIO 1.a.2 – RETORNA AO PASSO 1
	2.a – USUÁRIO JÁ CADASTRADO 2.a.1 – JANELA INFORMANDO A EXISTENCIA DO USUARIO 2.a.2 – RETORNA AO PASSO 2

2.4.2.2 Expansão do caso de uso "Logar no Sistema"

Tabela 18 - Expansão do Caso de Uso Logar no Sistema

CASO DE USO: LOGAR NO SISTEMA	
FLUXO PRINCIPAL	TRATAMENTO DE EXCEÇÕES
1- O USUÁRIO CADASTRADO INFORMA LOGIN E SENHA 2- CONFIRMA CLICANDO EM LOGAR 3- ACESSA O SISTEMA	1.a – CAMPOS OBRIGATORIOS NÃO PREENCHIDOS, OU INVÁLIDOS 1.a.1 – JANELA INFORMANDO O ERRO AO USUÁRIO 1.a.2 – RETORNA AO PASSO 1
	2.a – USUÁRIO NÃO EXISTE 2.a.1 – JANELA INFORMANDO QUE O USUÁRIO NÃO EXISTE 2.a.2 – RETORNA AO PASSO 2

2.4.2.3 Expansão do caso de uso "Deslogar no Sistema"

Tabela 19 - Expansão do Caso de Uso Deslogar no Sistema

CASO DE USO: DESLOGAR NO SISTEMA
FLUXO PRINCIPAL
1- O USUÁRIO LOGA NO SISTEMA 2- O USUÁRIO DESLOGA NO SISTEMA

2.4.2.4 Expansão do caso de uso "Cadastrar Cliente"

Tabela 20 - Expansão do Caso de Uso Cadastrar Cliente

CASO DE USO: CADASTRAR CLIENTE	
FLUXO PRINCIPAL	TRATAMENTO DE EXCEÇÕES
1- O USUÁRIO LOGA NO SISTEMA 2- SELECIONA O FORMULARIO PARA CADASTRO DO CLIENTE 3- PREENCHE AS INFORMAÇÕES DO CLIENTE 4- CONFIRMA OS DADOS DE CADASTRO NO BOTÃO CADASTRAR	1.a – CAMPOS OBRIGATORIOS NÃO PREENCHIDOS, OU INVÁLIDOS 1.a.1 – SURGE JANELA INFORMANDO O ERRO AO USUÁRIO 1.a.2 – RETORNA AO PASSO 1
	2.a – CLIENTE EM DUPLICIDADE 2.a.1 – SURGE JANELA INFORMANDO A DUPLICIDADE 2.a.2 – RETORNA AO PASSO 2

2.4.2.5 Expansão do caso de uso "Alterar Cliente"

Tabela 21 - Expansão do Caso de Uso Alterar Cliente

CASO DE USO: ALTERAR CLIENTE	
FLUXO PRINCIPAL	TRATAMENTO DE EXCEÇÕES
1- O USUÁRIO LOGA NO SISTEMA 2- SELECIONA O FORMULARIO DE ALTERAÇÃO DE CLIENTE 3- ALTERA AS INFORMAÇÕES DO CLIENTE 4- CONFIRMA OS DADOS ALTERADOS NO BOTÃO ALTERAR	1.a – CAMPOS OBRIGATORIOS NÃO PREENCHIDOS, OU INVÁLIDOS 1.a.1 – SURGE JANELA INFORMANDO O ERRO AO USUÁRIO 1.a.2 – RETORNA AO PASSO 1

2.4.2.6 Expansão do caso de uso "Excluir Cliente"

Tabela 22 - Expansão do Caso de Uso Excluir Cliente

CASO DE USO: EXCLUIR CLIENTE	
FLUXO PRINCIPAL	TRATAMENTO DE EXCEÇÕES
1- O USUÁRIO LOGA NO SISTEMA 2- SELECIONA O CLIENTE PARA EXCLUSÃO 3- CONFIRMA A EXCLUSÃO DO CLIENTE	1.a – CLIENTE VINCULADO EM UMA O MAIS VENDA 1.a.1 – SURGE JANELA INFORMANDO O VINCULO 1.a.2 – RETORNA AO PASSO 1

2.4.2.7 Expansão do caso de uso "Consultar Cliente"

Tabela 23 - Expansão do Caso de Uso Consultar Cliente

CASO DE USO: CONSULTAR CLIENTE	
FLUXO PRINCIPAL	
1- O USUÁRIO LOGA NO SISTEMA 2- SELECIONA CONSULTA DE CLIENTES 3- SELECIONA CLIENTE DESEJADO	

2.4.2.8 Expansão do caso de uso "Cadastrar Produto"

Tabela 24 - Expansão do Caso de Uso Cadastrar Produto

CASO DE USO: CADASTRAR PRODUTO	
FLUXO PRINCIPAL	TRATAMENTO DE EXCEÇÕES
1- O USUÁRIO LOGA NO SISTEMA 2- SELECIONA O FORMULARIO PARA CADASTRO DE PRODUTO 3- PREENCHE AS INFORMAÇÕES DO PRODUTO A SER CADASTRADO 4- CONFIRMA OS DADOS DE CADASTRO NO BOTÃO CADASTRAR	1.a – CAMPOS OBRIGATORIOS NÃO PREENCHIDOS, OU INVÁLIDOS 1.a.1 – SURGE JANELA INFORMANDO O ERRO AO USUÁRIO 1.a.2 – RETORNA AO PASSO 1
	2.a – PRODUTO EM DUPLICIDADE 2.a.1 – SURGE JANELA INFORMANDO A DUPLICIDADE 2.a.2 – RETORNA AO PASSO 2

2.4.2.9 Expansão do caso de uso "Alterar Produto"

Tabela 25 - Expansão do Caso de Uso Alterar Produto

CASO DE USO: ALTERAR PRODUTO	
FLUXO PRINCIPAL	TRATAMENTO DE EXCEÇÕES
1- O USUÁRIO LOGA NO SISTEMA 2- SELECIONA O FORMULARIO DE ALTERAÇÃO DE PRODUTO 3- ALTERA AS INFORMAÇÕES DO PRODUTO JÁ CADASTRADO 4- CONFIRMA OS DADOS ALTERADOS NO BOTÃO ALTERAR	1.a – CAMPOS OBRIGATORIOS NÃO PREENCHIDOS, OU INVÁLIDOS 1.a.1 – SURGE JANELA INFORMANDO O ERRO AO USUÁRIO 1.a.2 – RETORNA AO PASSO 1

2.4.2.10 Expansão do caso de uso "Excluir Produto"

Tabela 26 - Expansão do Caso de Uso Excluir Produto

CASO DE USO: EXCLUIR PRODUTO	
FLUXO PRINCIPAL	TRATAMENTO DE EXCEÇÕES
1- O USUÁRIO LOGA NO SISTEMA 2- SELECIONA O PRODUTO PARA EXCLUSÃO 3- CONFIRMA A EXCLUSÃO DO PRODUTO JÁ CADASTRADO	1.a – PRODUTO VINCULADO EM UMA O MAIS VENDA 1.a.1 – SURGE JANELA INFORMANDO O VINCULO 1.a.2 – RETORNA AO PASSO 1

2.4.2.11 Expansão do caso de uso "Consultar Produto"

Tabela 27 - Expansão do Caso de Uso Consultar Produto

CASO DE USO: CONSULTAR PRODUTO	
FLUXO PRINCIPAL	
1- O USUÁRIO LOGA NO SISTEMA 2- SELECIONA CONSULTA DE PRODUTOS 3- SELECIONA PRODUTO DESEJADO	

2.4.2.12 Expansão do caso de uso "Cadastrar Pedido"

Tabela 28 - Expansão do Caso de Uso Cadastrar Pedido

CASO DE USO: CADASTRAR PEDIDO	
FLUXO PRINCIPAL	TRATAMENTO DE EXCEÇÕES
1- O USUÁRIO LOGA NO SISTEMA 2- SELECIONA O FORMULARIO PARA CADASTRO DO PEDIDO 3- PREENCHE AS INFORMAÇÕES DO PEDIDO 4- VINCULA UM CLIENTE NO PEDIDO 5- VINCULA UM OU MAIS PRODUTOS NO PEDIDO 6- CONFIRMA OS DADOS DE CADASTRO NO BOTÃO CADASTRAR	1.a – CAMPOS OBRIGATORIOS NÃO PREENCHIDOS, OU INVÁLIDOS 1.a.1 – SURGE JANELA INFORMANDO O ERRO AO USUÁRIO 1.a.2 – RETORNA AO PASSO 1 2.a – CLIENTE E PRODUTO NÃO VINCULADO 2.a.1 – SURGE JANELA INFORMANDO O ERRO AO USUÁRIO 2.a.2 – RETORNA AO PASSO 2

2.4.2.13 Expansão do caso de uso "Alterar Pedido"

Tabela 29 - Expansão do Caso de Uso Alterar Pedido

CASO DE USO: ALTERAR PEDIDO	
FLUXO PRINCIPAL	TRATAMENTO DE EXCEÇÕES
1- O USUÁRIO LOGA NO SISTEMA 2- SELECIONA O FORMULARIO DE ALTERAÇÃO DE PEDIDO 3- ALTERA AS INFORMAÇÕES DO PEDIDO 4- CONFIRMA OS DADOS ALTERADOS NO BOTÃO ALTERAR	1.a – CAMPOS OBRIGATORIOS NÃO PREENCHIDOS, OU INVÁLIDOS 1.a.1 – SURGE JANELA INFORMANDO O ERRO AO USUÁRIO 1.a.2 – RETORNA AO PASSO 1 2.a – CLIENTE OU PRODUTO NÃO VINCULADO 2.a.1 – SURGE JANELA INFORMANDO O ERRO AO USUÁRIO 2.a.2 – RETORNA AO PASSO 2

2.4.2.14 Expansão do caso de uso "Excluir Pedido"

Tabela 30 - Expansão do Caso de Uso Excluir Pedido

CASO DE USO: EXCLUIR PEDIDO	
FLUXO PRINCIPAL	
1- O USUÁRIO LOGA NO SISTEMA 2- SELECIONA O PEDIDO PARA EXCLUSÃO 3- CONFIRMA A EXCLUSÃO DO PEDIDO NO BOTÃO EXCLUIR PEDIDO	

2.4.2.15 Expansão do caso de uso "Consultar Pedido"

Tabela 31 - Expansão do Caso de Uso Consultar Pedido

CASO DE USO: CONSULTAR Pedido
FLUXO PRINCIPAL
1- O USUÁRIO LOGA NO SISTEMA 2- SELECIONA CONSULTA DE PEDIDOS 3- SELECIONA O PEDIDO DESEJADO

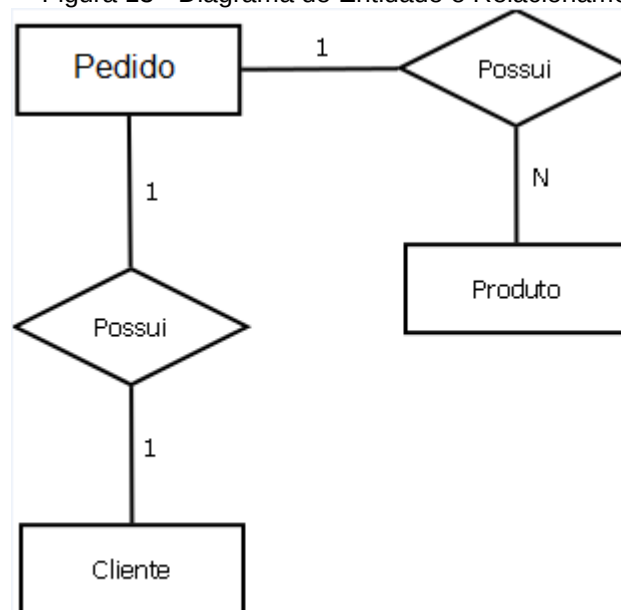
3 DIAGRAMAS

Os diagramas do sistema servem para fornecer múltiplas visões do sistema a ser feito, ajudando na análise sob vários aspectos, proporcionando ferramentas para o desenvolvimento de um sistema completo e sem falhas de projeto. Nessa seção será feito os diagramas que são utilizados como base para o sistema.

3.1 Diagrama de entidade e relacionamento do sistema

O diagrama de entidade e relacionamento do sistema, representa o relacionamento que o pedido terá com produto e cliente. Uma venda poderá conter um cliente e vários produtos, segue na imagem abaixo diagrama.

Figura 13 - Diagrama de Entidade e Relacionamento

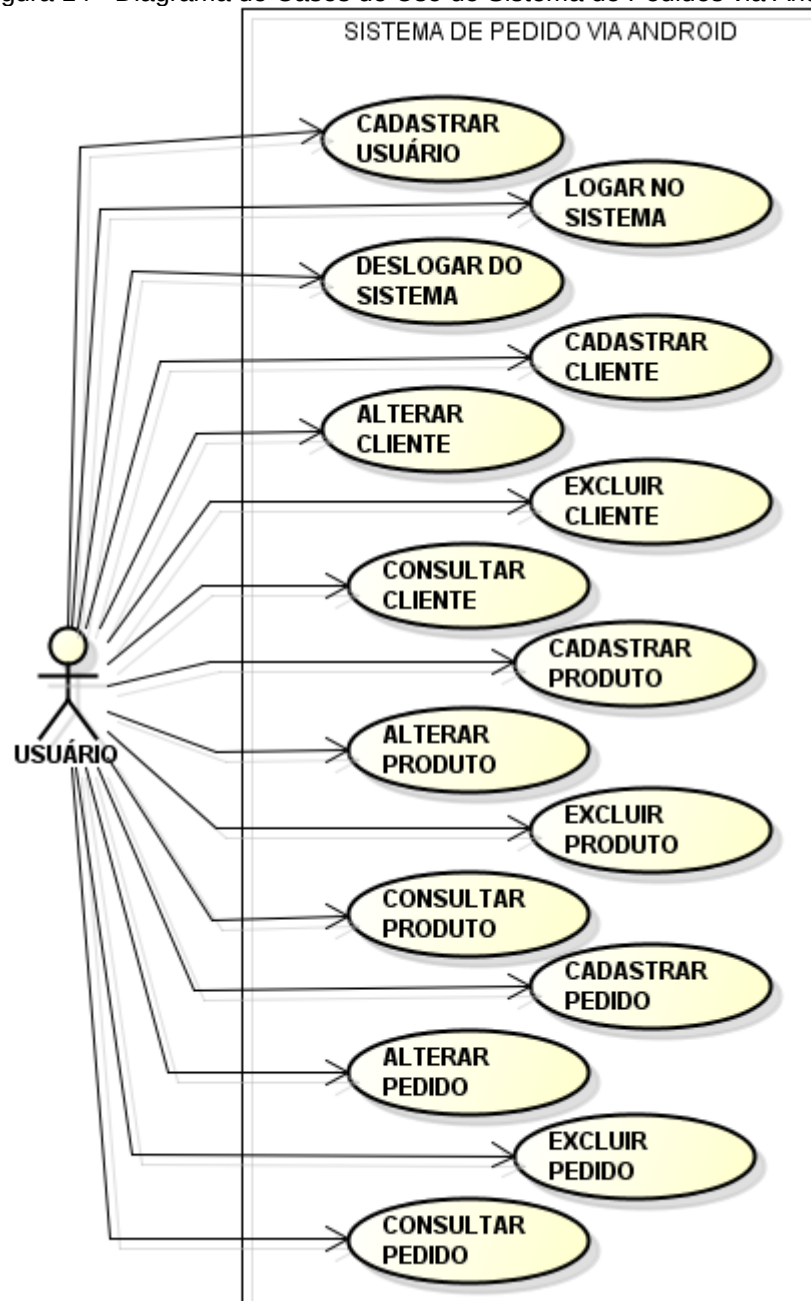


Fonte: Elaborado pelo autor

3.2 Diagrama de casos de uso

Na imagem abaixo mostra o diagrama de casos de uso do sistema, representando todas as informações que o usuário poderá ter interação no software.

Figura 14 - Diagrama de Casos de Uso do Sistema de Pedidos via Android

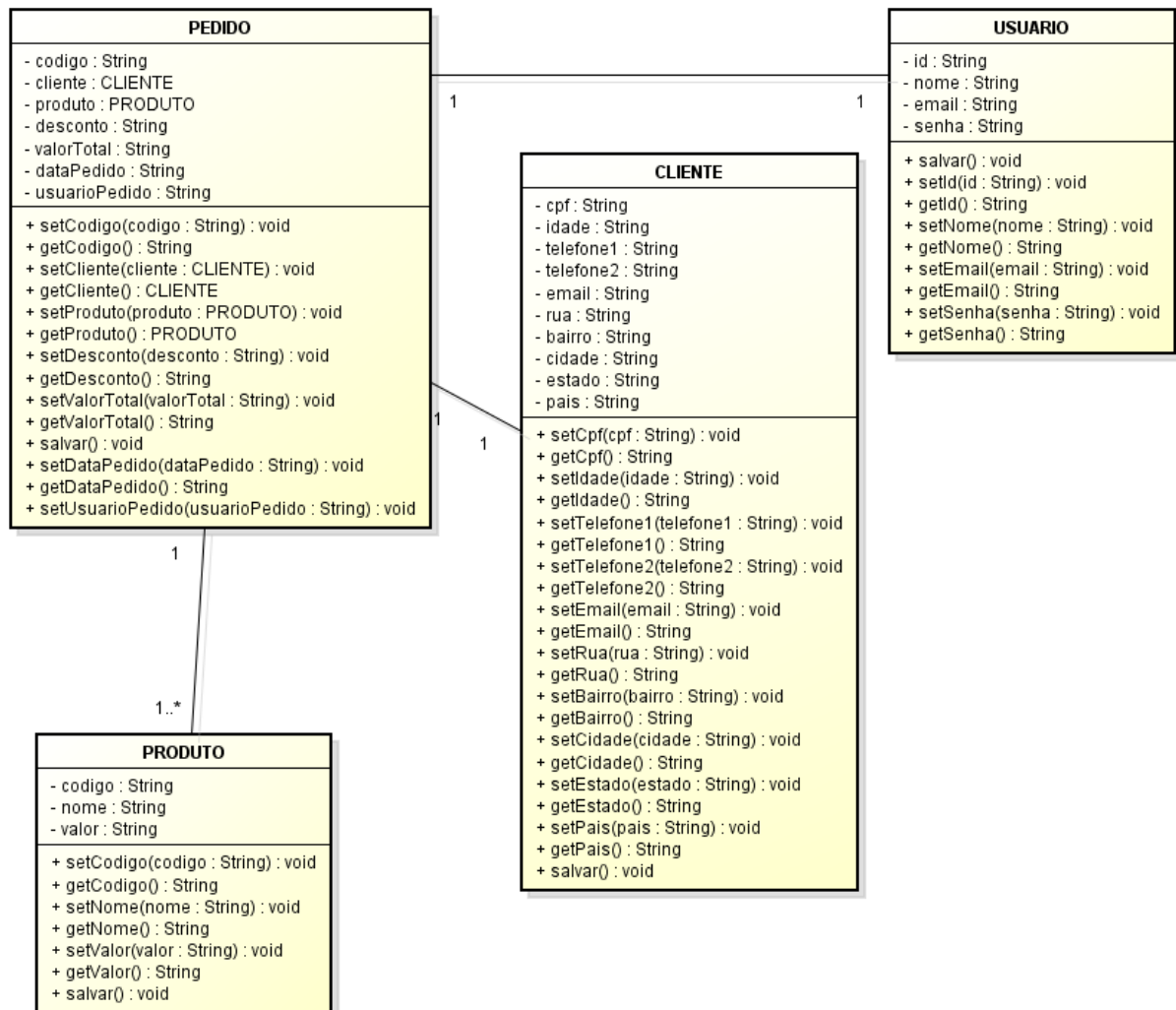


Fonte: Elaborado pelo autor

3.3 Diagrama de classes

O diagrama de classes representa as quatro classes do sistema que são pedido, cliente, produto e usuário, ao quais mostra seu atributos, métodos e relacionamento, o diagrama está representado na imagem abaixo.

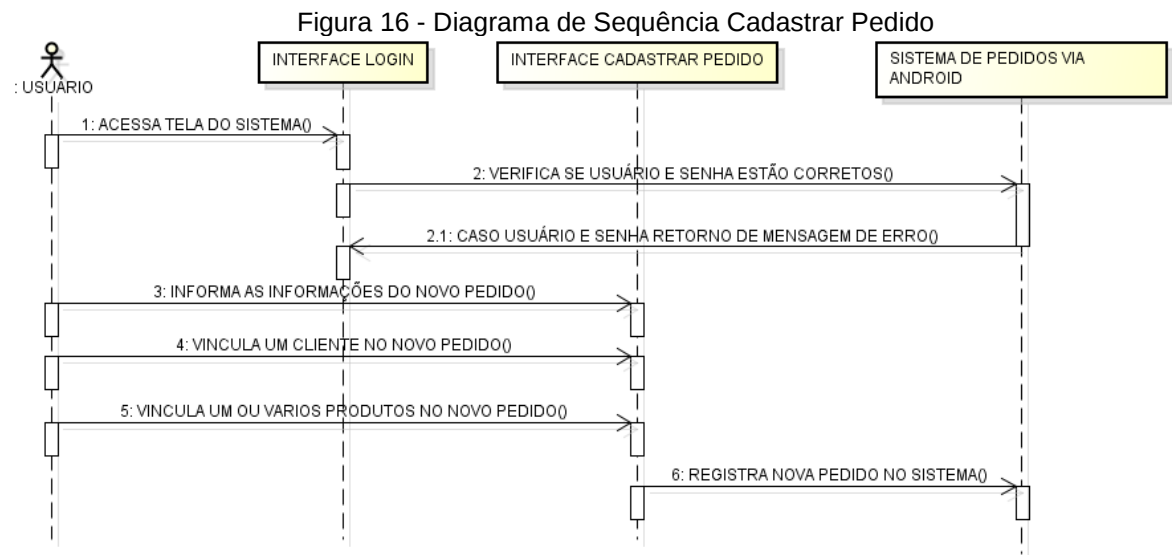
Figura 15 - Diagrama de Classes do Sistema de Pedidos via Android



Fonte: Elaborado pelo autor

3.4 Diagrama de Sequência

Na imagem abaixo é mostrado o diagrama de sequência para cadastro de um novo pedido no sistema. O diagrama mostra as sequencias necessárias que o usuário deverá seguir para concluir a tarefa.

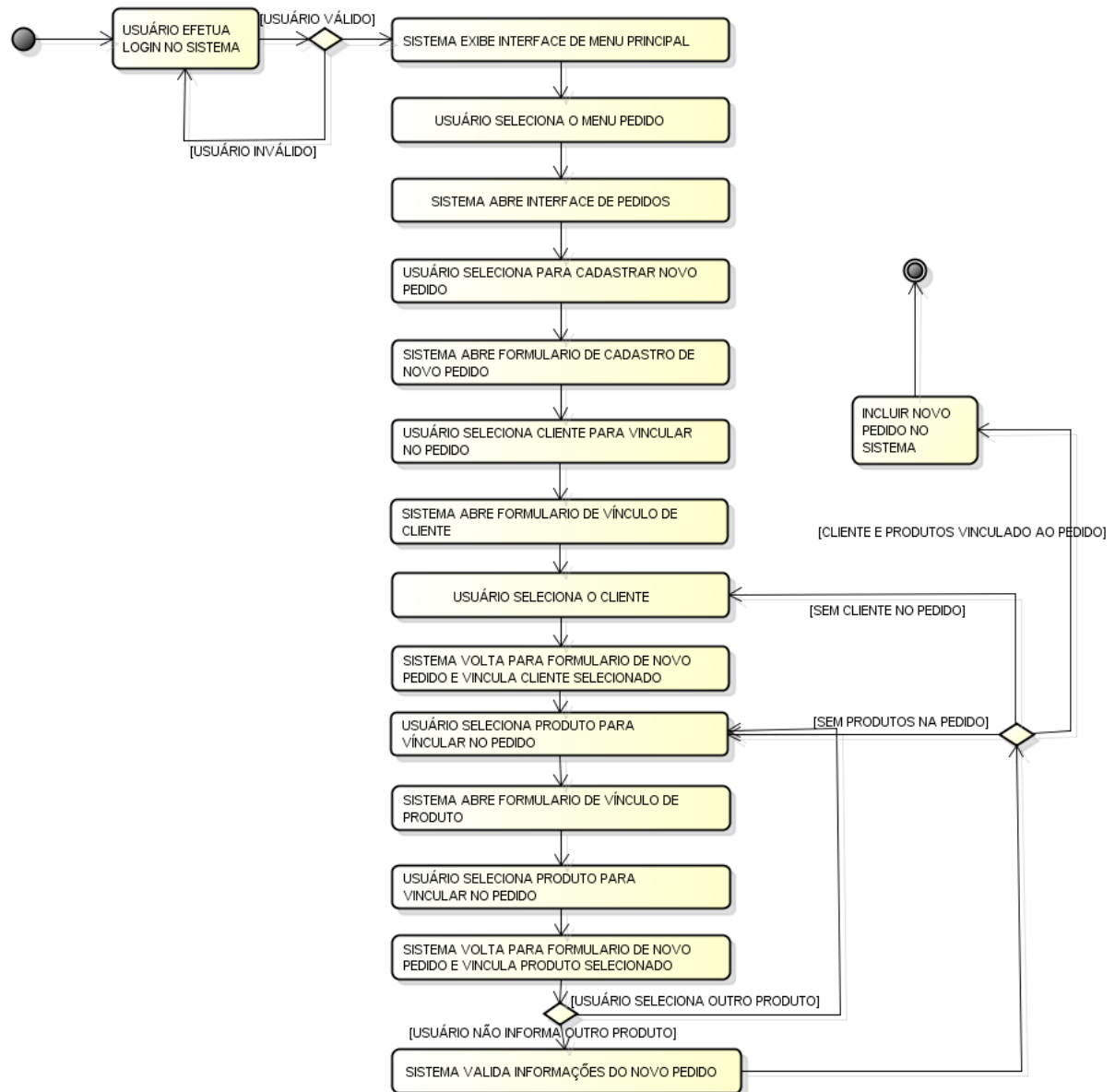


Fonte: Elaborado pelo autor

3.5 DIAGRAMA DE ATIVIDADE

Na imagem abaixo é mostrado o diagrama de atividade para cadastro de um novo pedido no sistema. O diagrama mostra as atividades necessárias que o usuário deverá seguir para concluir a tarefa.

Figura 17 - Diagrama de Atividade Cadastrar Pedido no Sistema



Fonte: Elaborado pelo autor

4 CONCLUSÃO

O objetivo focal deste aplicativo foi de desenvolver um software para aplicativo móvel capaz de dinamizar a tarefa que um vendedor externo tem em seu dia a dia de vendas. Auxiliando com consulta a vendas passadas, consulta de produtos, consultada de informações de seus clientes.

Hoje em dia as tecnologias estão muito avançadas e com custo bem reduzidos, uma pessoa comum pode comprar um SmartPhone com muita facilidade pois o preço é bem baixo. Também pode contratar planos de internet para o dispositivo de forma barata que garante um rápido acesso à internet. Isso é tudo que o aplicativo precisa para funcionar.

O desafio do desenvolvimento do software foi de torná-lo visualmente atrativo e fácil de utilização, viabilizando o uso do vendedor externo, que irá notar as vantagens na utilização do software.

O sistema de vendas via Android pode ser facilmente integrado com outras aplicações ou até mesmo ser expandido, já que utiliza o Firebase do Google, e é um sistema que pode ser feita conexões de várias linguagens de desenvolvimento.

O vendedor externo terá um sistema simples e usual, que estará junto com aplicativos do seu dia a dia, como telefone, agenda, redes sociais, o que torna o aplicativo de venda via Android, de fácil acesso e será difícil do vendedor esquecê-lo, já que é um dispositivo de uso pessoal.

5 REFERÊNCIAS

COBRA, Marcos. ADMINISTRAÇÃO DE VENDAS. Atlas, 5ª Edição, 2014.

GUEDES, Guleanes. UML2 UMA ABORDAGEM PRÁTICA, Novatec, 2ª Edição, 2011.

Wazlawick, Raul. ANÁLISE E PROJETO DE SISTEMAS DE INFORMAÇÃO ORIENTADOS A OBJETOS, 2ª Edição, 2011.

Venda direta. Disponível em: <<http://www.abevd.org.br/venda-direta/>> Acesso em: 27 de fevereiro de 2018.

Evolução da profissão de vendedor. Disponível em: <[Http://www.administradores.com.br/artigos/negocios/evolucao-da-profissao-de-vendedor/50042/](http://www.administradores.com.br/artigos/negocios/evolucao-da-profissao-de-vendedor/50042/)> Acesso em: 15 de outubro de 2017.

Conhecendo a evolução de vendas. Disponível em: <[Https://www.administradores.com.br/artigos/cotidiano/conhecendo-a-evolucao-de-vendas/99218/](https://www.administradores.com.br/artigos/cotidiano/conhecendo-a-evolucao-de-vendas/99218/)> Acesso em: 15 de outubro de 2017.

75% da população mundial tem acesso ao celular, afirma pesquisa. Disponível em: <[Http://g1.globo.com/tecnologia/noticia/2012/07/75-da-populacao-mundial-tem-acesso-ao-celular-afirma-pesquisa.html](http://g1.globo.com/tecnologia/noticia/2012/07/75-da-populacao-mundial-tem-acesso-ao-celular-afirma-pesquisa.html)> Acesso em: 20 de outubro de 2017.

RUP. Disponível em: <[Https://www.infoescola.com/engenharia-de-software/rup](https://www.infoescola.com/engenharia-de-software/rup)> Acesso em: 22 de outubro de 2017.

Atividades básicas ao processo de desenvolvimento de software. Disponível em: <[Http://www.devmedia.com.br/atividades-basicas-ao-processo-de-desenvolvimento-de-software/5413](http://www.devmedia.com.br/atividades-basicas-ao-processo-de-desenvolvimento-de-software/5413)> Acesso em: 23 de outubro de 2017.

Astah: Desenvolvendo Plug-ins para o software. Disponível em: <[Http://www.devmedia.com.br/astah-desenvolvendo-plug-ins-para-o-software/31127](http://www.devmedia.com.br/astah-desenvolvendo-plug-ins-para-o-software/31127)> Acesso raem: 23 de outubro de 2017.

Conheça o Android Studio. Disponível em: <[Https://developer.android.com/studio/intro/index.html?hl=pt-br](https://developer.android.com/studio/intro/index.html?hl=pt-br)> Acesso em: 25 de outubro de 2017.

Recursos do Sistema Android. Disponível em: <[Http://www.androidpro.com.br/recursos-do-sistema-android/](http://www.androidpro.com.br/recursos-do-sistema-android/)> Acesso em: 25 de outubro de 2017.

Fundamentos da linguagem Java. Disponível em: <<https://www.ibm.com/developerworks/br/java/tutorials/j-introtojava1/index.html>> Acesso em: 26 de outubro de 2017.

TIOBE Index for November 2017. Disponível em: <<https://www.tiobe.com/tiobe-index/>> Acesso em: 26 de outubro de 2017.

Google Firebase for dummies: o que é e como funciona a plataforma. Disponível em: <<https://blog.mastertech.tech/tecnologia/google-firebase-for-dummies-o-que-e-e-como-funciona-plataforma/>> Acesso em: 27 de outubro de 2017.

Firebase. Disponível em: <<https://en.wikipedia.org/wiki/Firebase>> Acesso em: 27 de outubro de 2017.

O que é NoSql?. Disponível em: <<https://aws.amazon.com/pt/nosql/>> Acesso em: 28 de outubro de 2017.

NoSql - você realmente sabe do que estamos falando?. Disponível em: <<https://imasters.com.br/artigo/17043/banco-de-dados/nosql-voce-realmente-sabe-do-que-estamos-falando?trace=1519021197&source=single>> Acesso em: 28 de outubro de 2017.

Introdução ao formato JSON. Disponível em: <<http://www.devmedia.com.br/introducao-ao-formato-json/25275>> Acesso em: 30 de outubro de 2017.

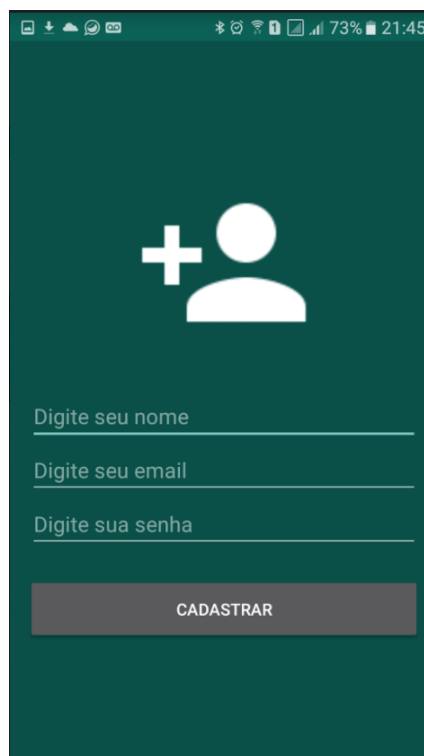
APÊNDICE A – TELAS DO SISTEMA

Tela de login do sistema



A screenshot of the SPA login screen. The background is a dark teal color. At the top, the status bar shows various icons and the time 20:04. The main heading 'SPA' is in large white letters, with the subtitle 'Sistema de Pedidos via Android' below it. There are two input fields: 'Digite seu e-mail' and 'Digite a senha', both with white text and underlined. Below these is a dark grey button labeled 'LOGAR'. At the bottom, there is a link that says 'Não tem conta? Solicite acesso!'.

Tela de cadastro de usuário no sistema



A screenshot of the SPA user registration screen. The background is a dark teal color. At the top, the status bar shows various icons and the time 21:45. The main heading is a white icon of a person with a plus sign. Below this are three input fields: 'Digite seu nome', 'Digite seu email', and 'Digite sua senha', all with white text and underlined. Below these is a dark grey button labeled 'CADASTRAR'.

Tela de menu do sistema



Tela de produtos cadastrados no sistema

Tela de cadastro de produtos no sistema



VOLTAR Cadastro de Produtos

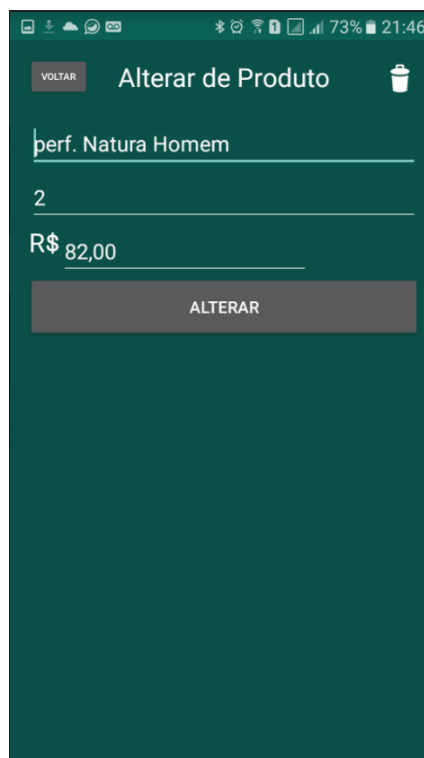
Nome do Produto

Codigo do produto

R\$ Valor do produto

CADASTRAR

Tela de alterar produto no sistema



VOLTAR Alterar de Produto

perf. Natura Homem

2

R\$ 82,00

ALTERAR

Tela de alterar cliente no sistema

VOLTAR Alterar Cliente

Lucas Oliveira

86662025640

30 64992363141

6436214536

luoliver@gmail.com

Rua taparazios n 89

Dona Gercina

Rio Verde

Go

Brasil

ALTERAR

Tela de menu de vendas cadastradas no sistema

← Pedidos +
Lucas Oliveira cod.Venda: 19 Valor Total: 658,90
Kuka Lituano cod.Venda: 18 Valor Total: 100,00
David Trindade cod.Venda: 16 Valor Total: 877,00
Antonio Rodrigues cod.Venda: 15 Valor Total: 340,00
David Trindade cod.Venda: 14 Valor Total: 0,00
Rogério Fernandes cod.Venda: 13 Valor Total: 589,00
Antonio Rodrigues cod.Venda: 11 Valor Total: 77,50
Fernando Ferreira cod.Venda: 10 Valor Total: 154,00
David Trindade cod.Venda: 9 Valor Total: 100,00

Tela de cadastrar venda no sistema

VOLTAR Pedido nº19

Lucas Oliveira CLIENTE

+	Produto	Valor un.	Qtd.	Valor Total
	perf. Jabuticaba Águas	59,90	11	658,90

Total R\$658,90 Desconto: R\$ 0,00

CONCLUIR PEDIDO

Usuário: davidt.info@gmail.com

Tela de vincular cliente na venda

Selecione um Cliente

David Trindade
CPF/CNPJ: 04201843173

Fernando Ferreira
CPF/CNPJ: 04204204242

Antonio Rodrigues
CPF/CNPJ: 09709709779

Rogerio Fernandes
CPF/CNPJ: 32132376123

Lucas Oliveira
CPF/CNPJ: 86662025640

Tela de vincular produto na venda



Selecionar um Produto

Sabonete Mel
código: 18 | valor:R\$45,00

des. roll-on natura homem
código: 16 | valor:R\$16,40

des. roll-on erva doce
código: 15 | valor:R\$14,90

des. roll-on Kaiak fem.
código: 14 | valor:R\$16,40

des. em creme
código: 13 | valor:R\$19,90

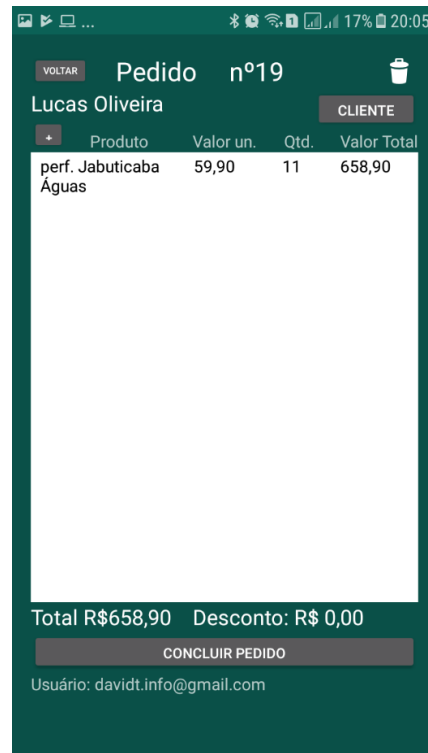
perf. Jabuticaba Águas
código: 12 | valor:R\$59,90

perf. Biografia
código: 11 | valor:R\$114,90

perf. Humor
código: 10 | valor:R\$99,90

perf. Sr. N
código: 9 | valor:R\$89,90

Tela de alterar venda no sistema



VOLTAR **Pedido n°19** 

Lucas Oliveira **CLIENTE**

+	Produto	Valor un.	Qtd.	Valor Total
	perf. Jabuticaba Águas	59,90	11	658,90

Total R\$658,90 Desconto: R\$ 0,00

CONCLUIR PEDIDO

Usuário: davidt.info@gmail.com

APÊNDICE B – Diagramas de Sequência

Diagrama de Sequência Cadastrar Usuário

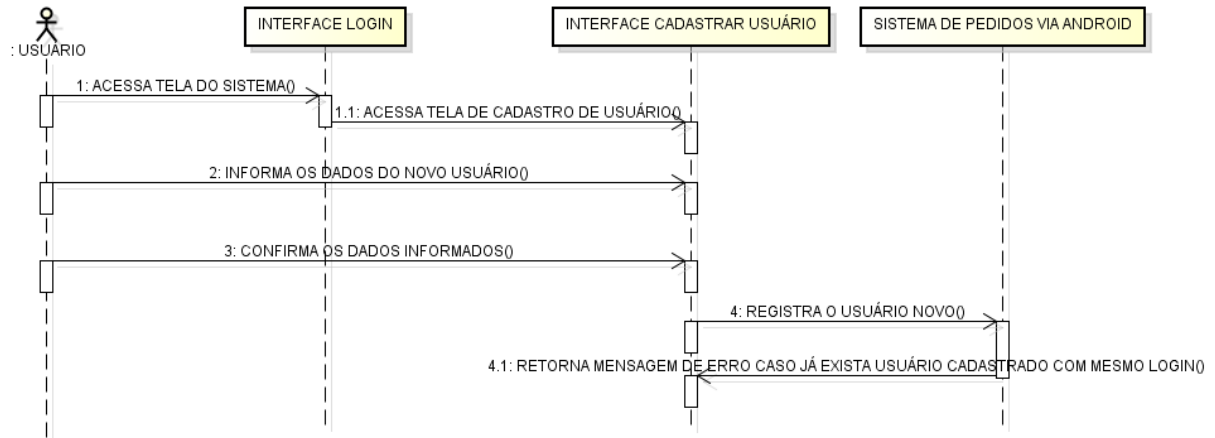


Diagrama de Sequência Logar no Sistema

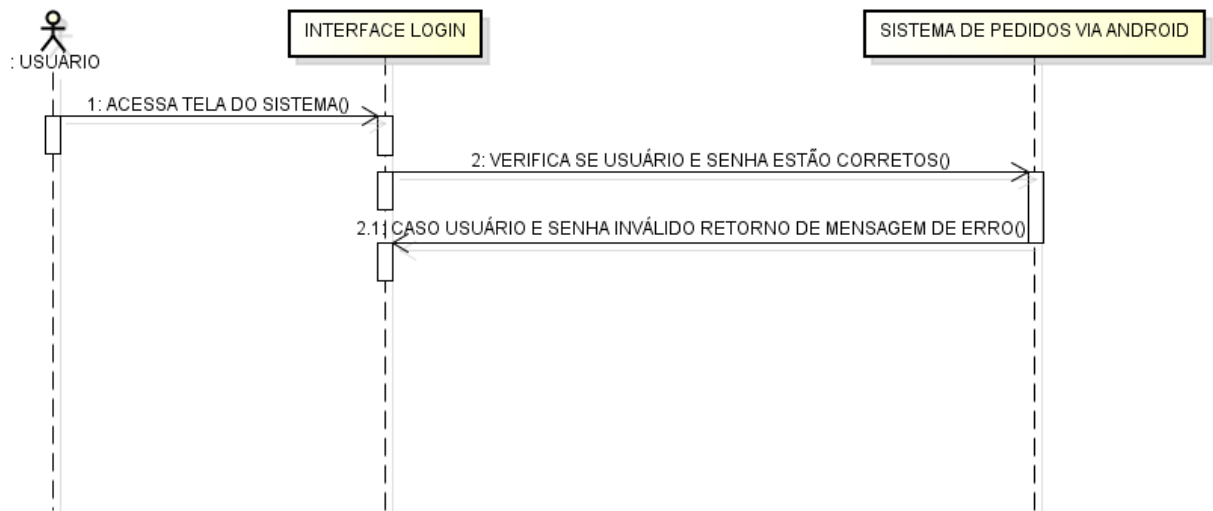


Diagrama de Sequência Deslogar no Sistema

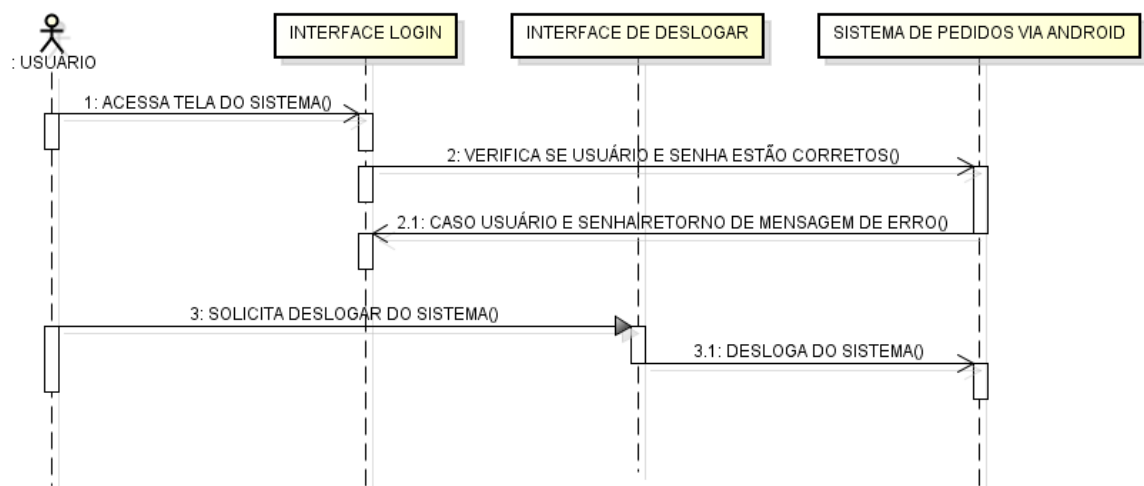


Diagrama de Sequência Cadastrar Cliente

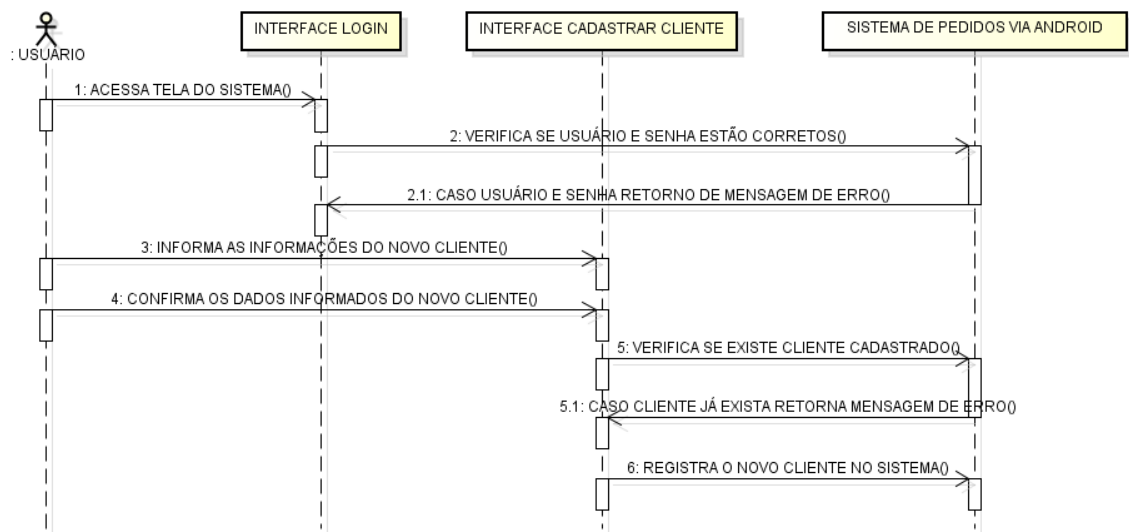


Diagrama de Sequência Alterar Cliente

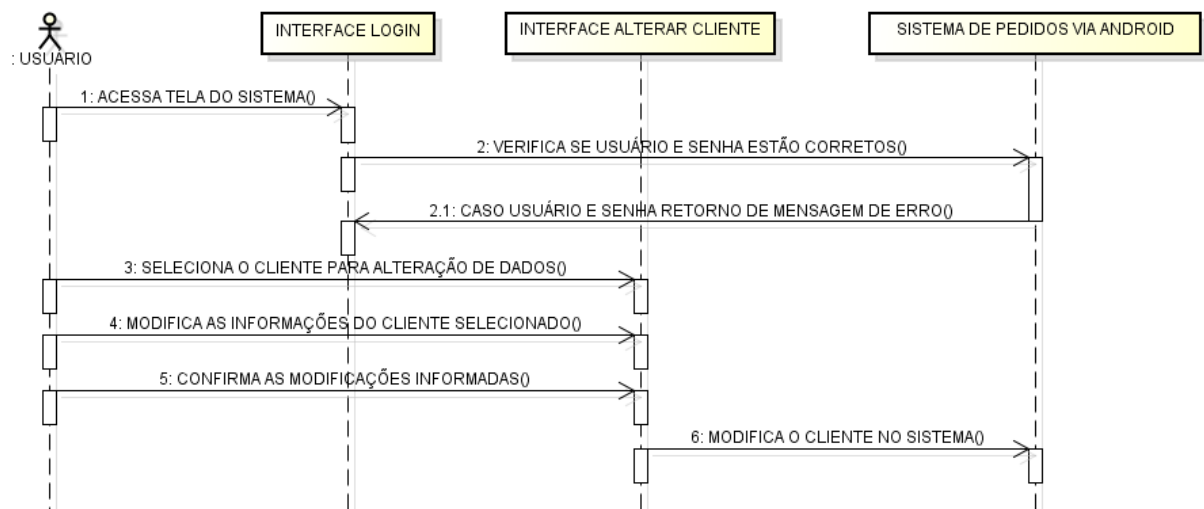


Diagrama de Sequência Excluir Cliente

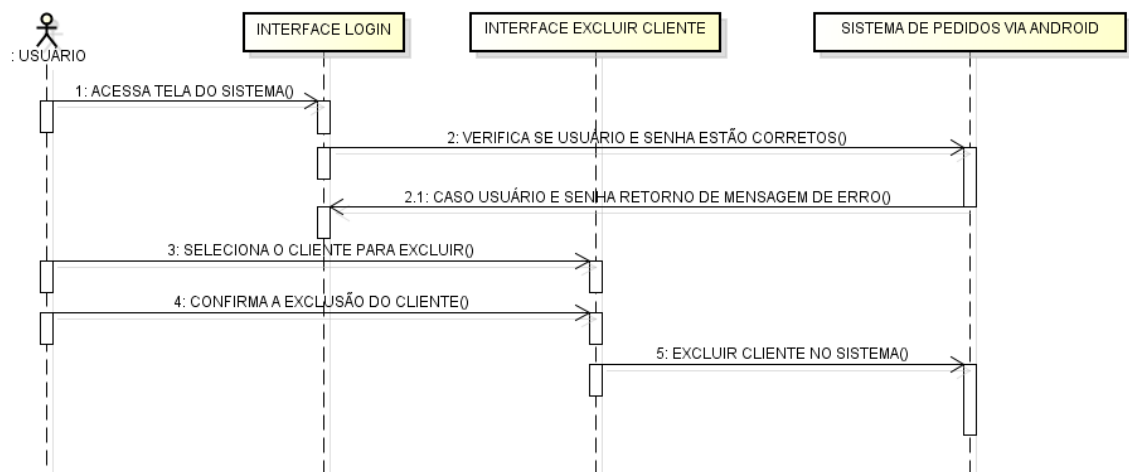


Diagrama de Sequência Consultar Cliente

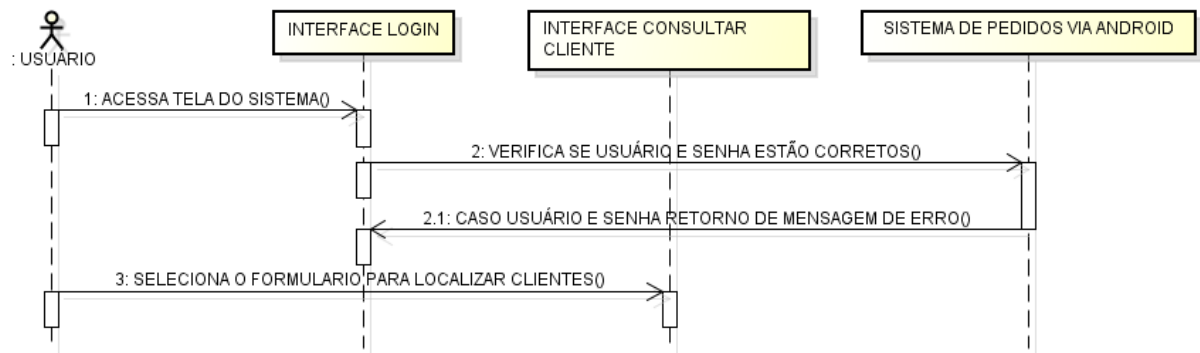


Diagrama de Sequência Consultar Produto

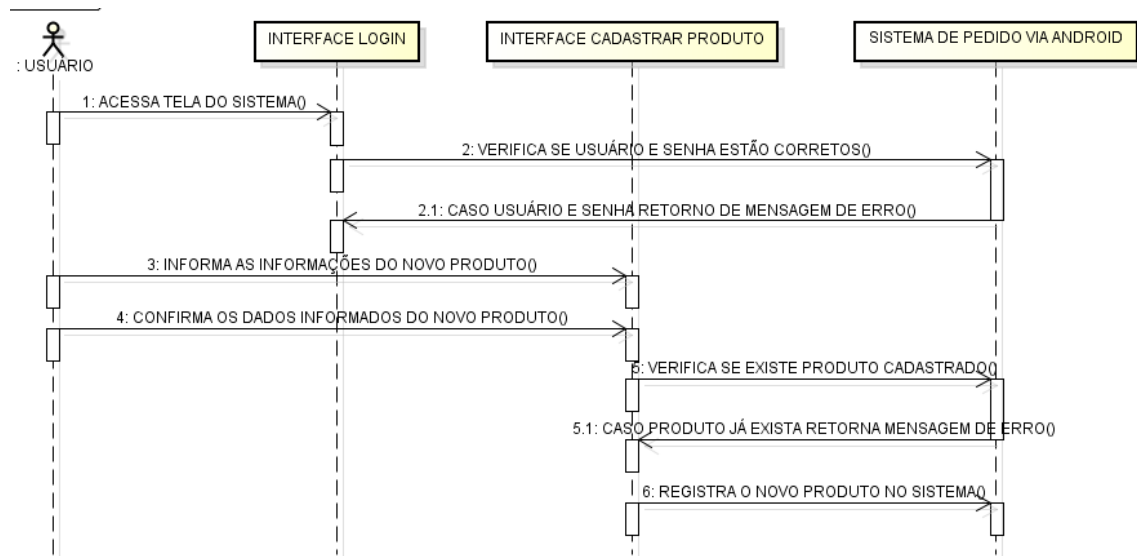


Diagrama de Sequência Alterar Produto

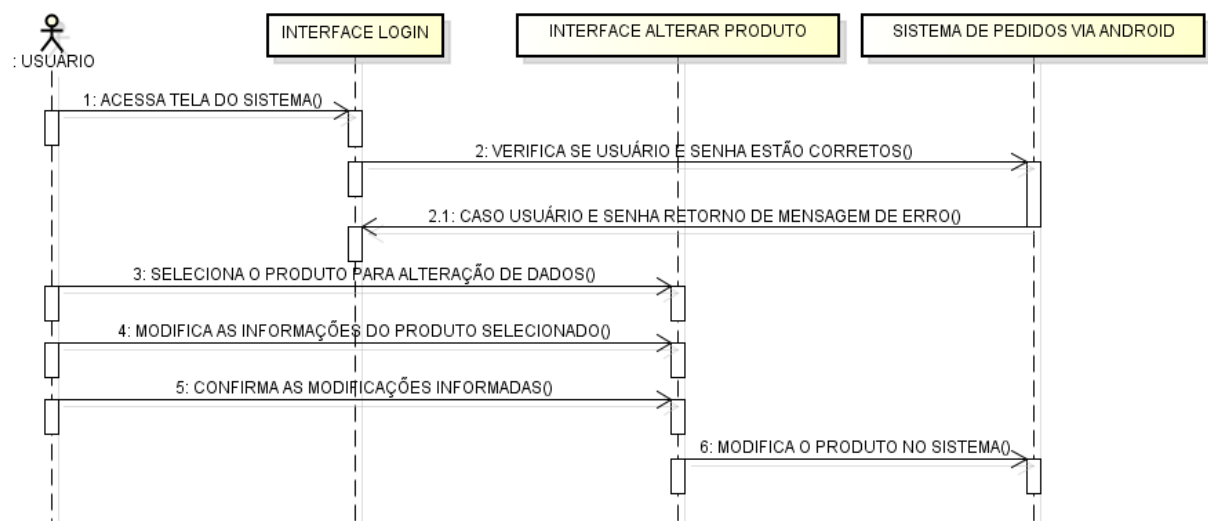


Diagrama de Sequência Excluir Produto

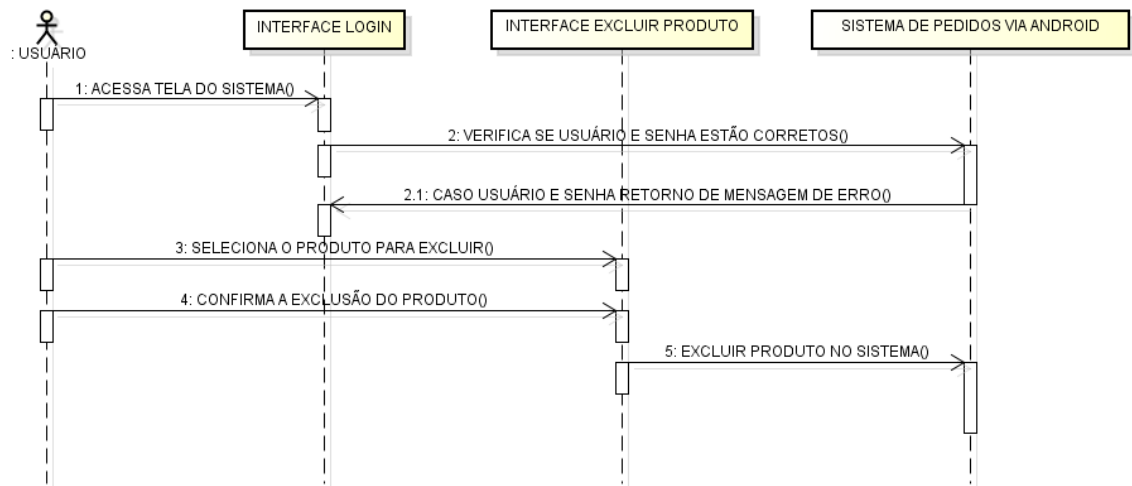


Diagrama de Sequência Consultar Produto

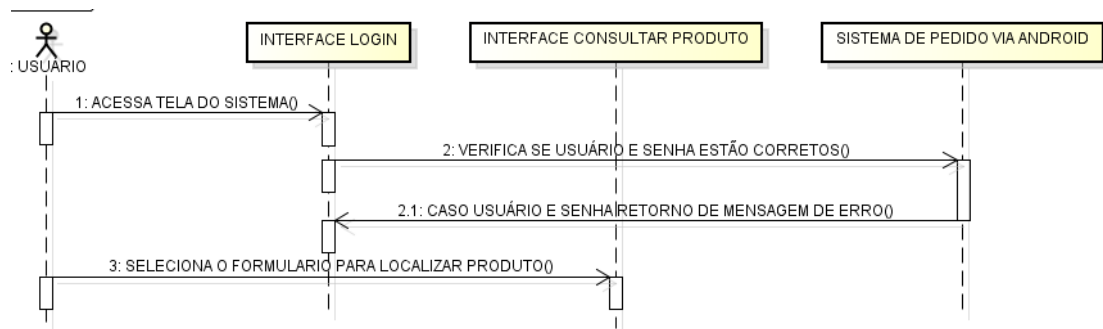


Diagrama de Sequência Cadastrar Pedido

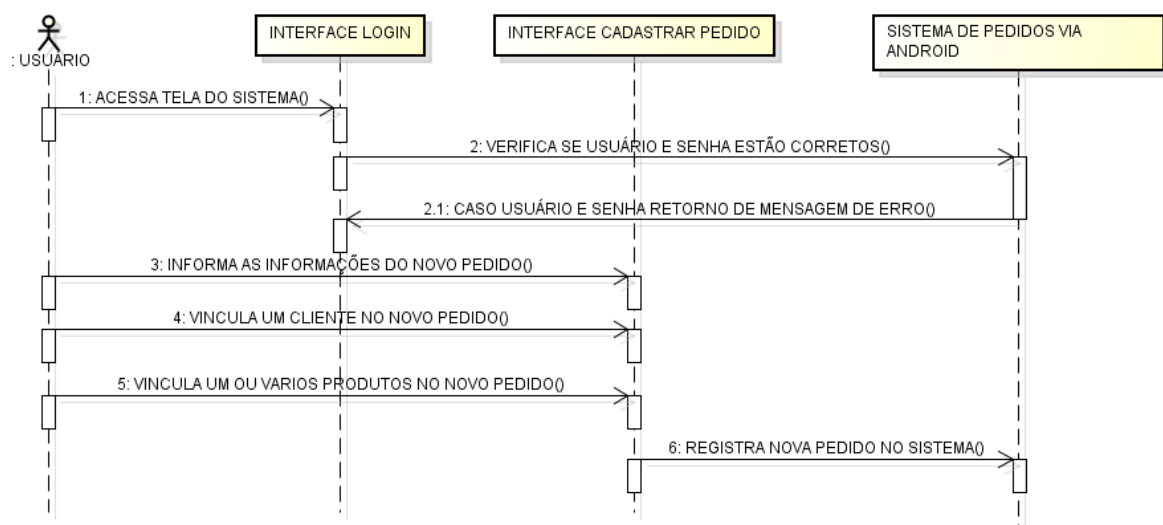


Diagrama de Sequência Alterar Pedido

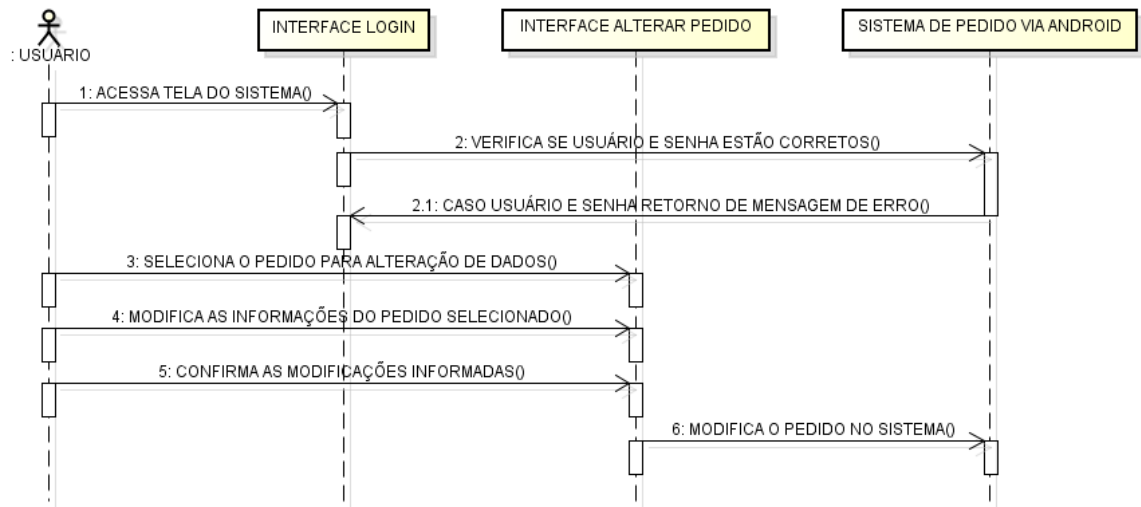


Diagrama de Sequência Excluir Venda

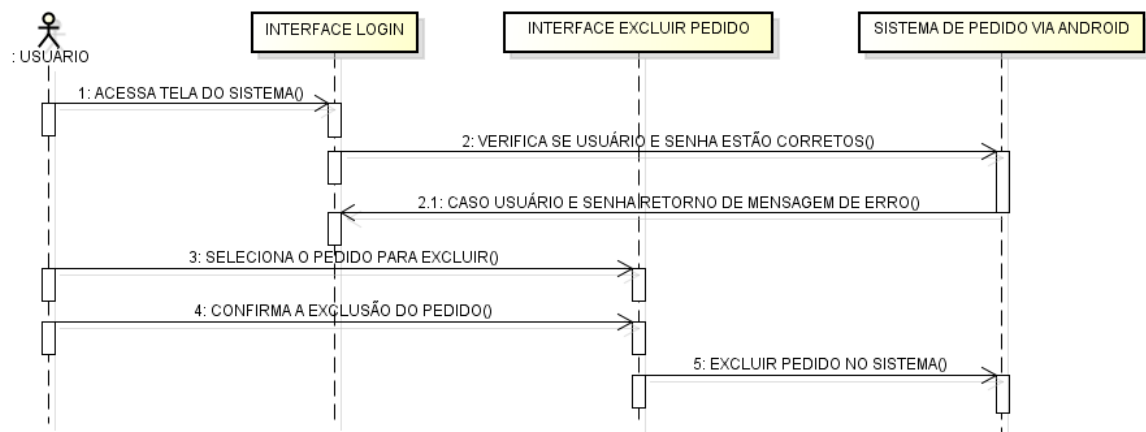
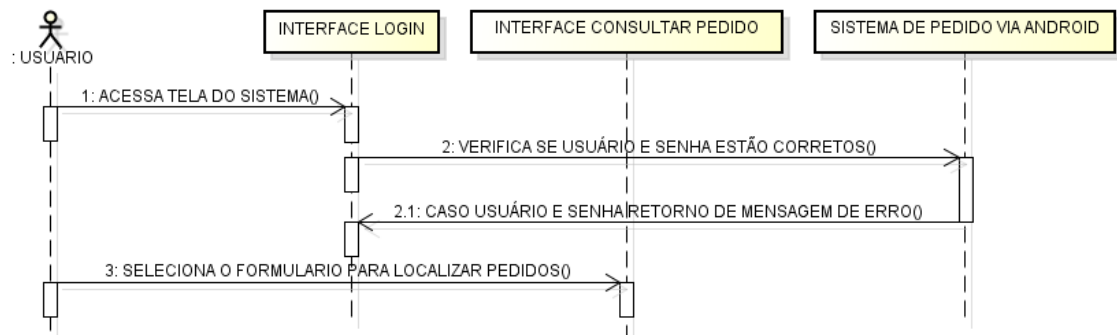


Diagrama de Sequência Consultar Venda



APÊNDICE C – Diagramas de atividade

Diagrama de Atividade Cadastrar Usuário no Sistema



Diagrama de Atividade Logar no Sistema

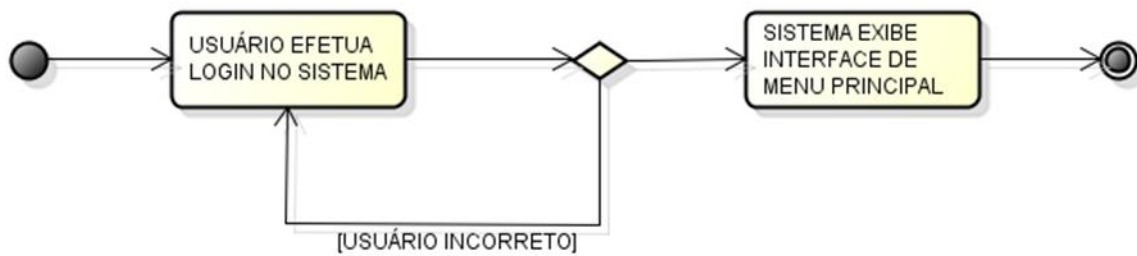


Diagrama de Atividade Deslogar no Sistema

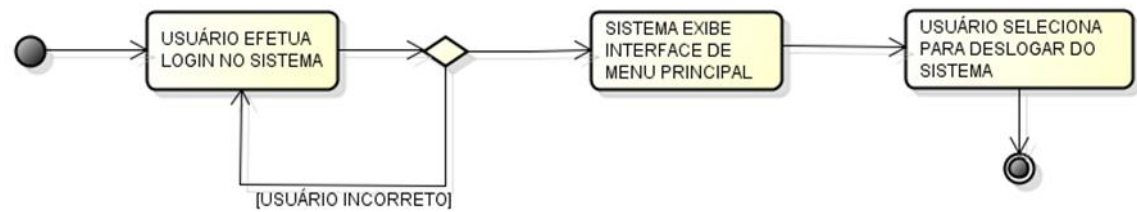


Diagrama de Atividade Cadastrar Cliente no Sistema

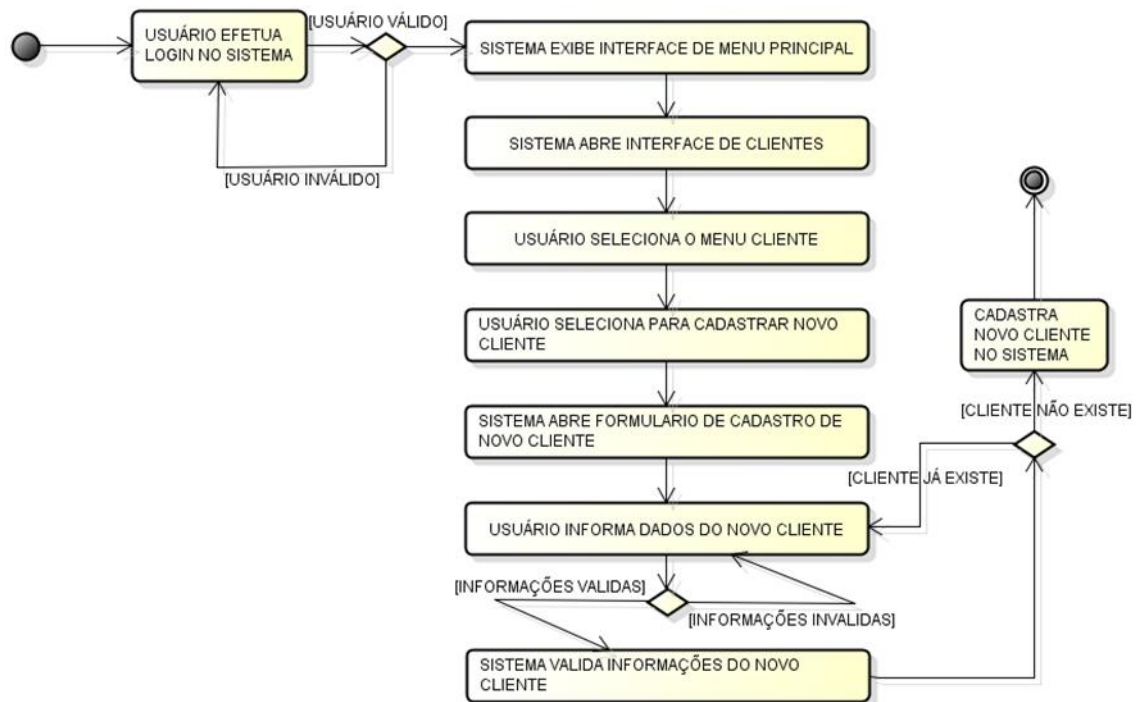


Diagrama de Atividade Alterar Cliente no Sistema

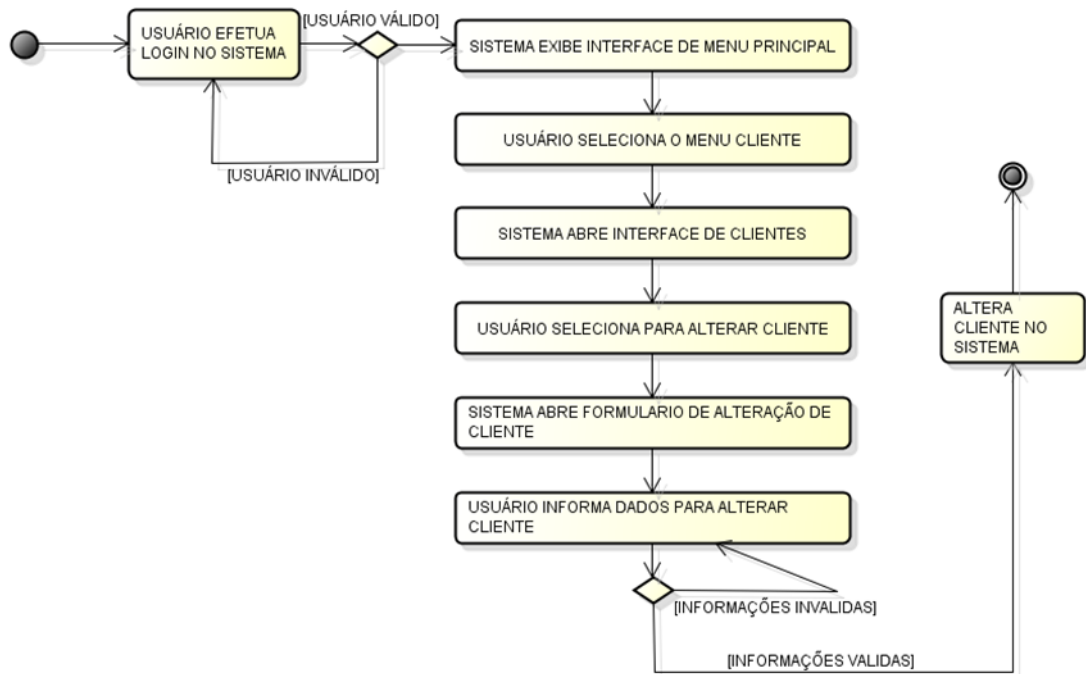


Diagrama de Atividade Excluir Cliente no Sistema

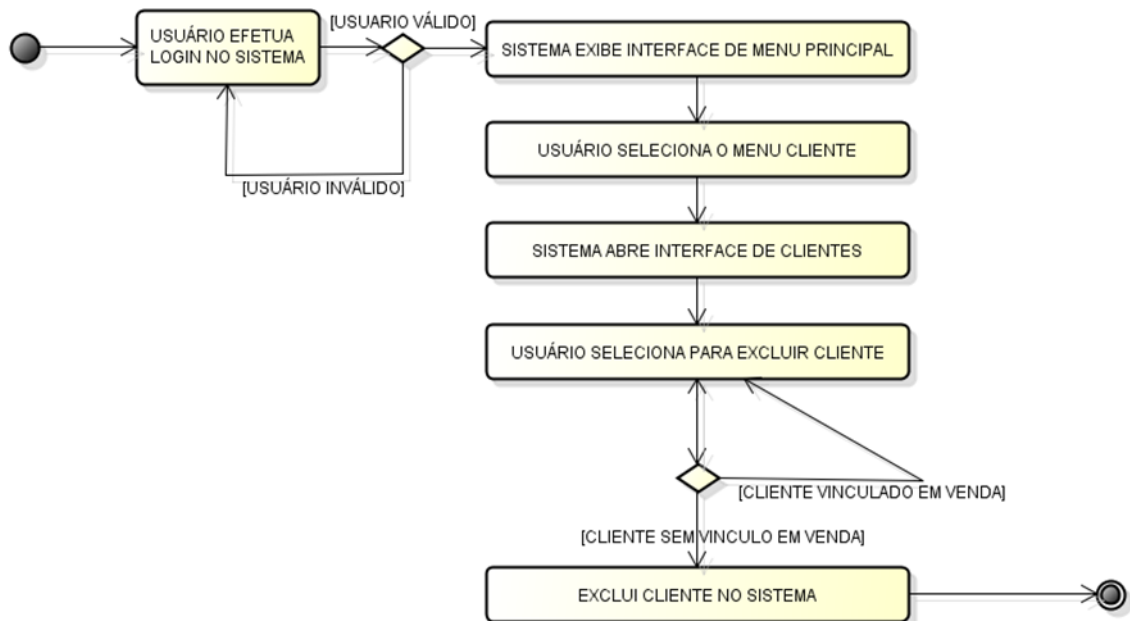


Diagrama de Atividade Consultar Cliente no Sistema

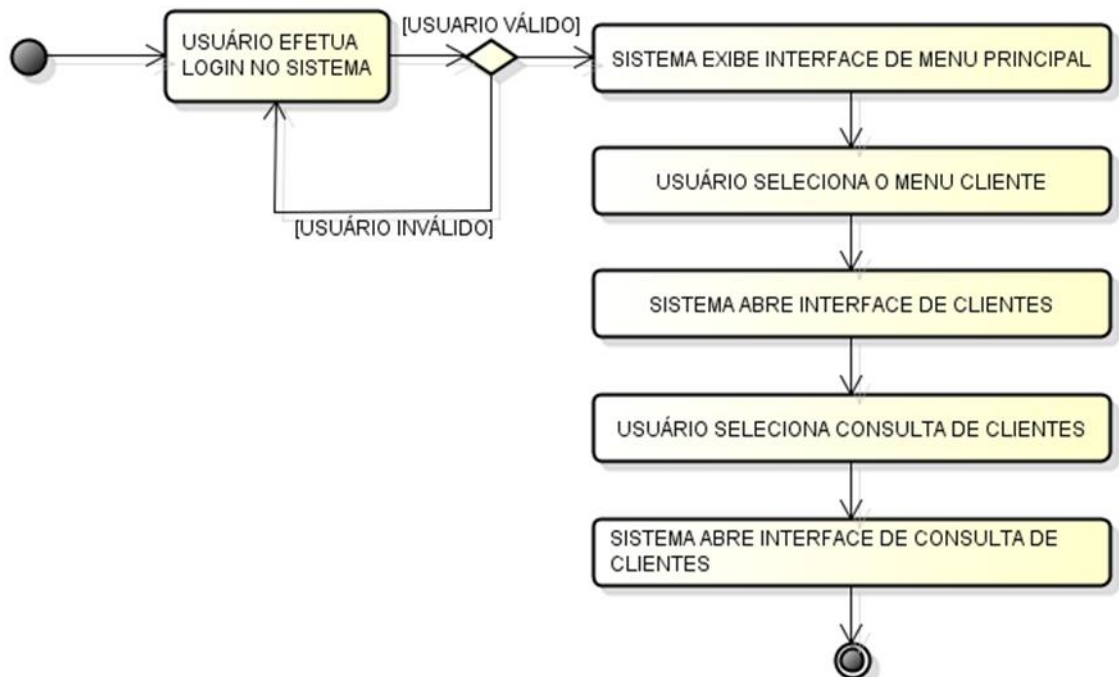


Diagrama de Atividade Cadastrar Produto no Sistema

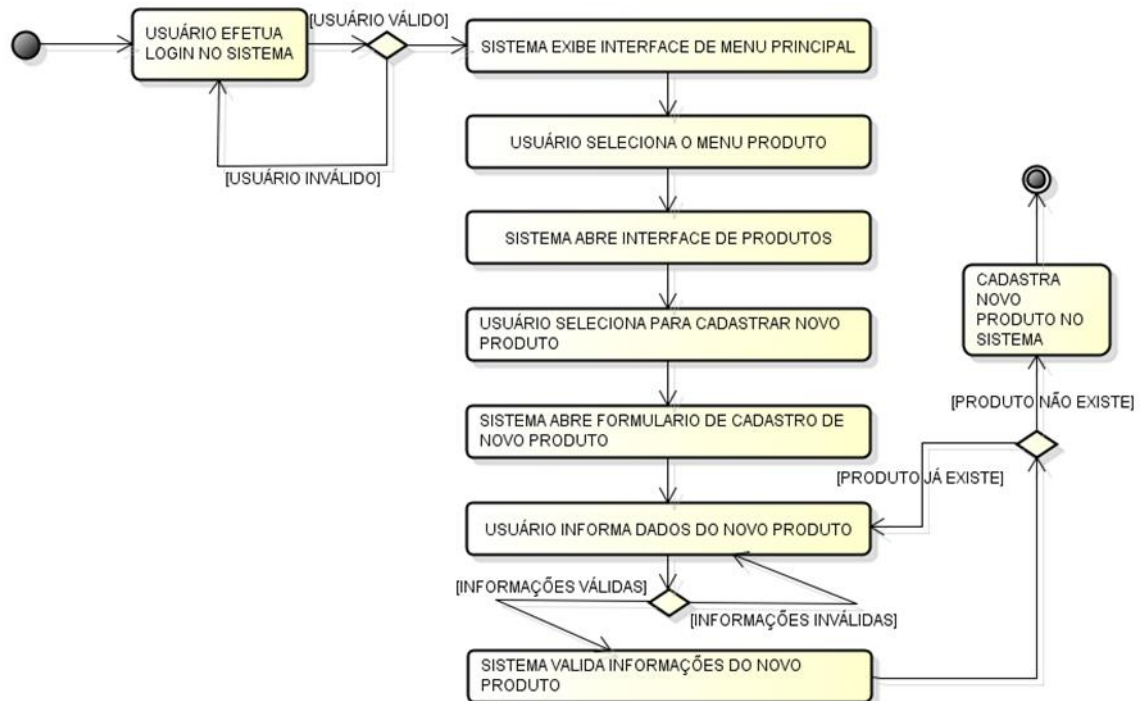


Diagrama de Atividade Alterar Produto no Sistema

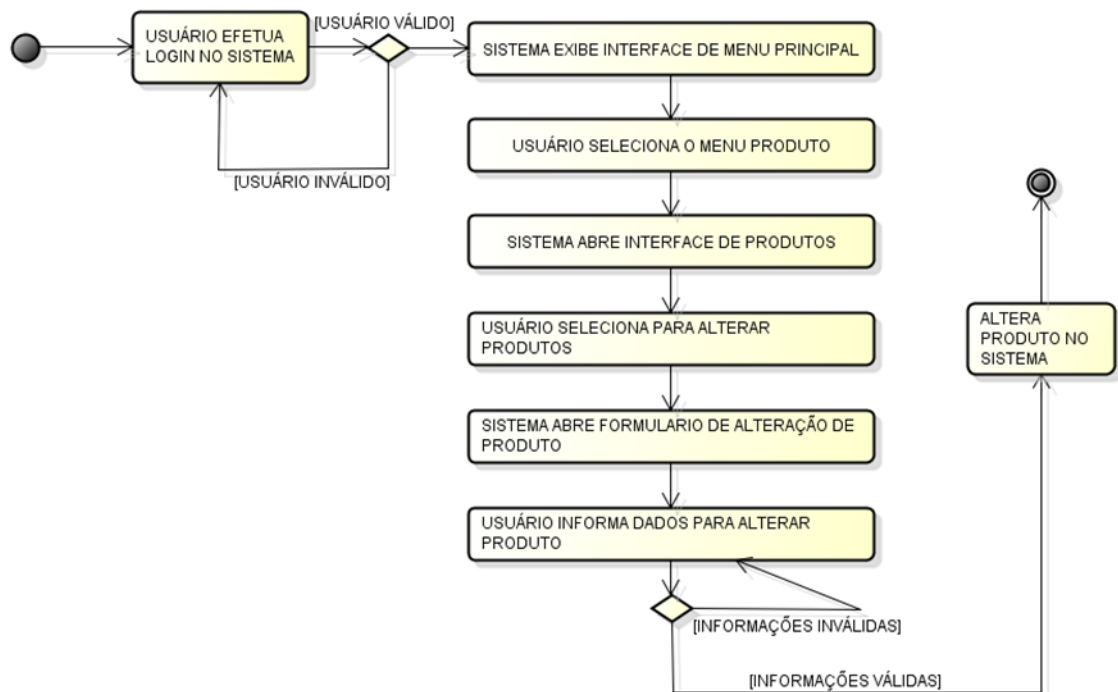


Diagrama de Atividade Excluir Produto no Sistema

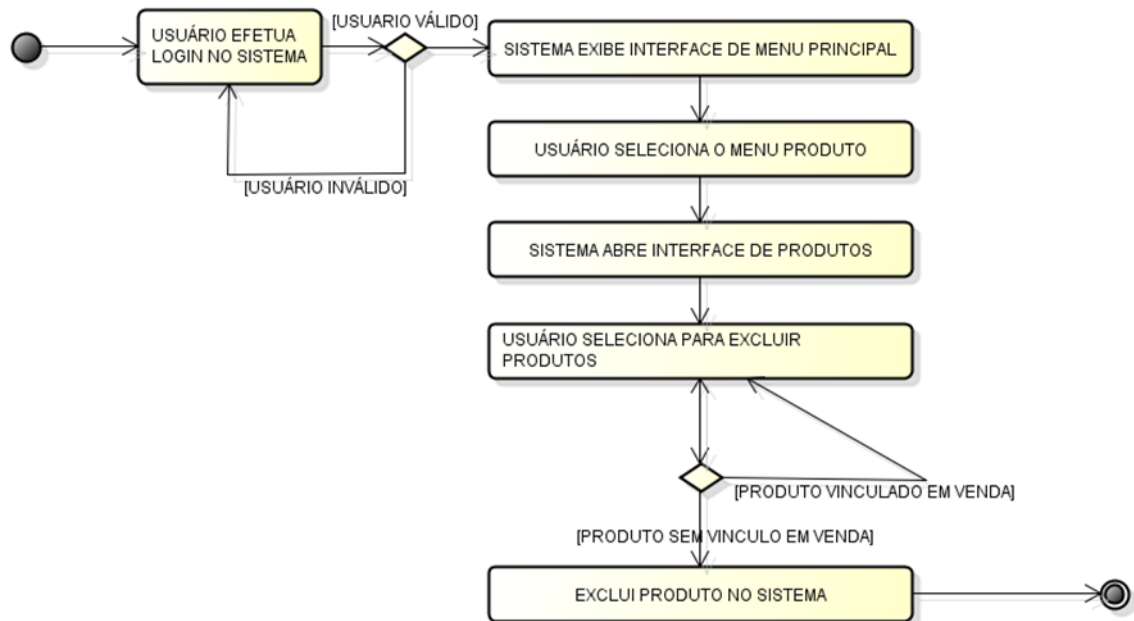


Diagrama de Atividade Consultar Produto no Sistema

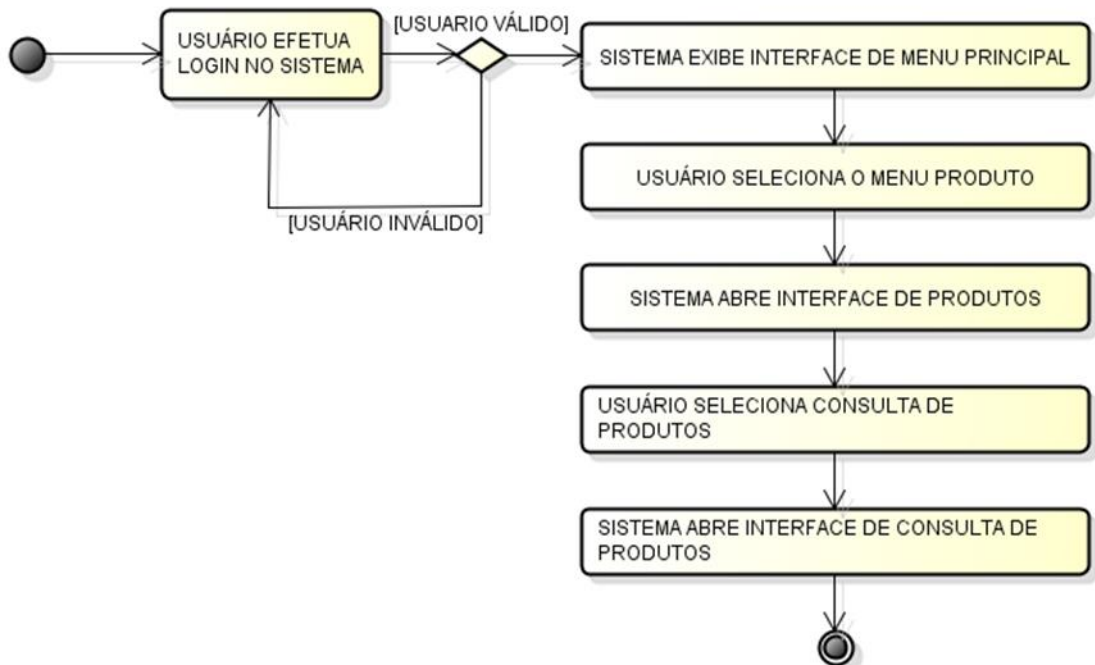


Diagrama de Atividade Cadastrar Pedido no Sistema

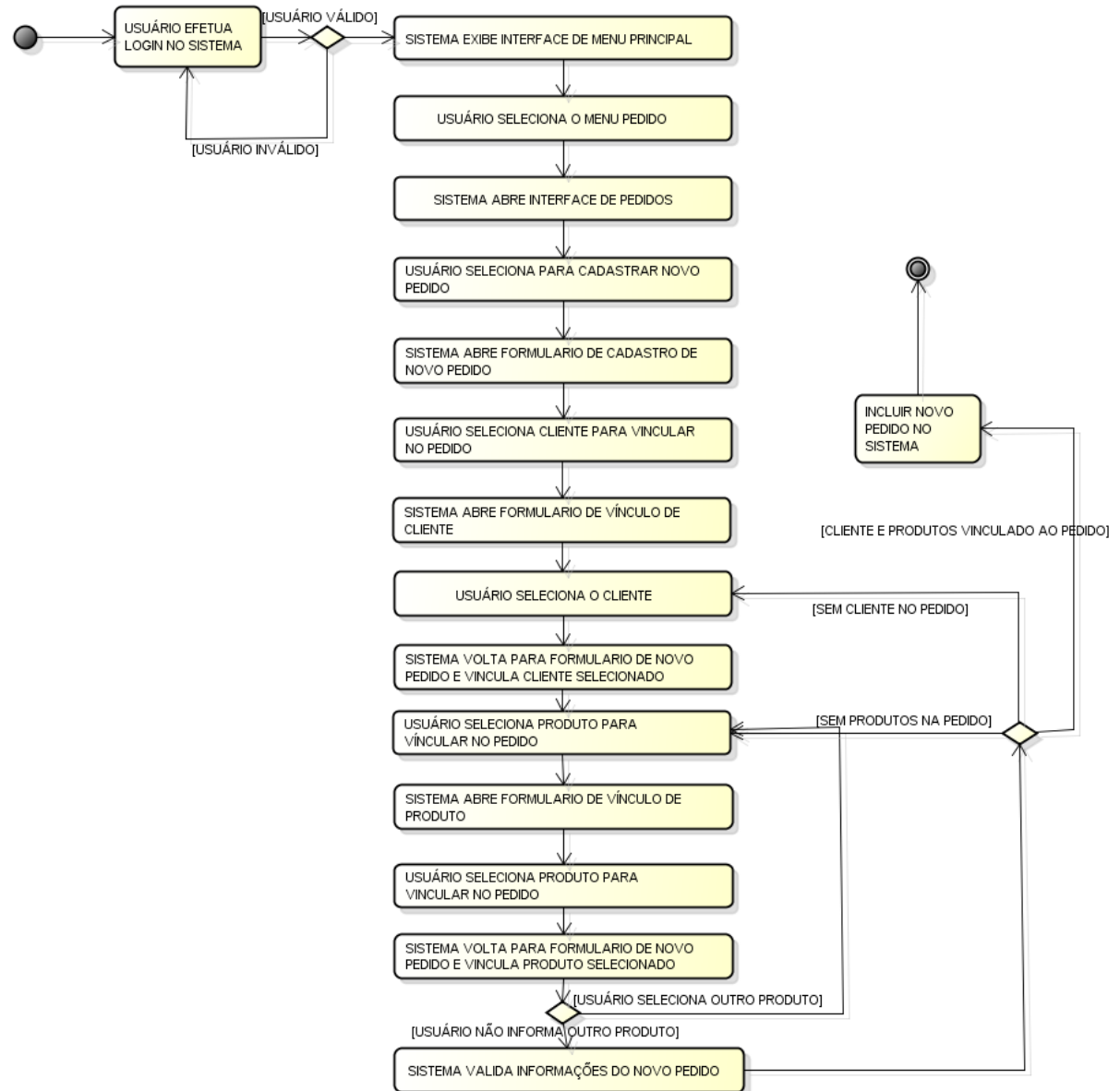


Diagrama de Atividade Alterar Pedido no Sistema

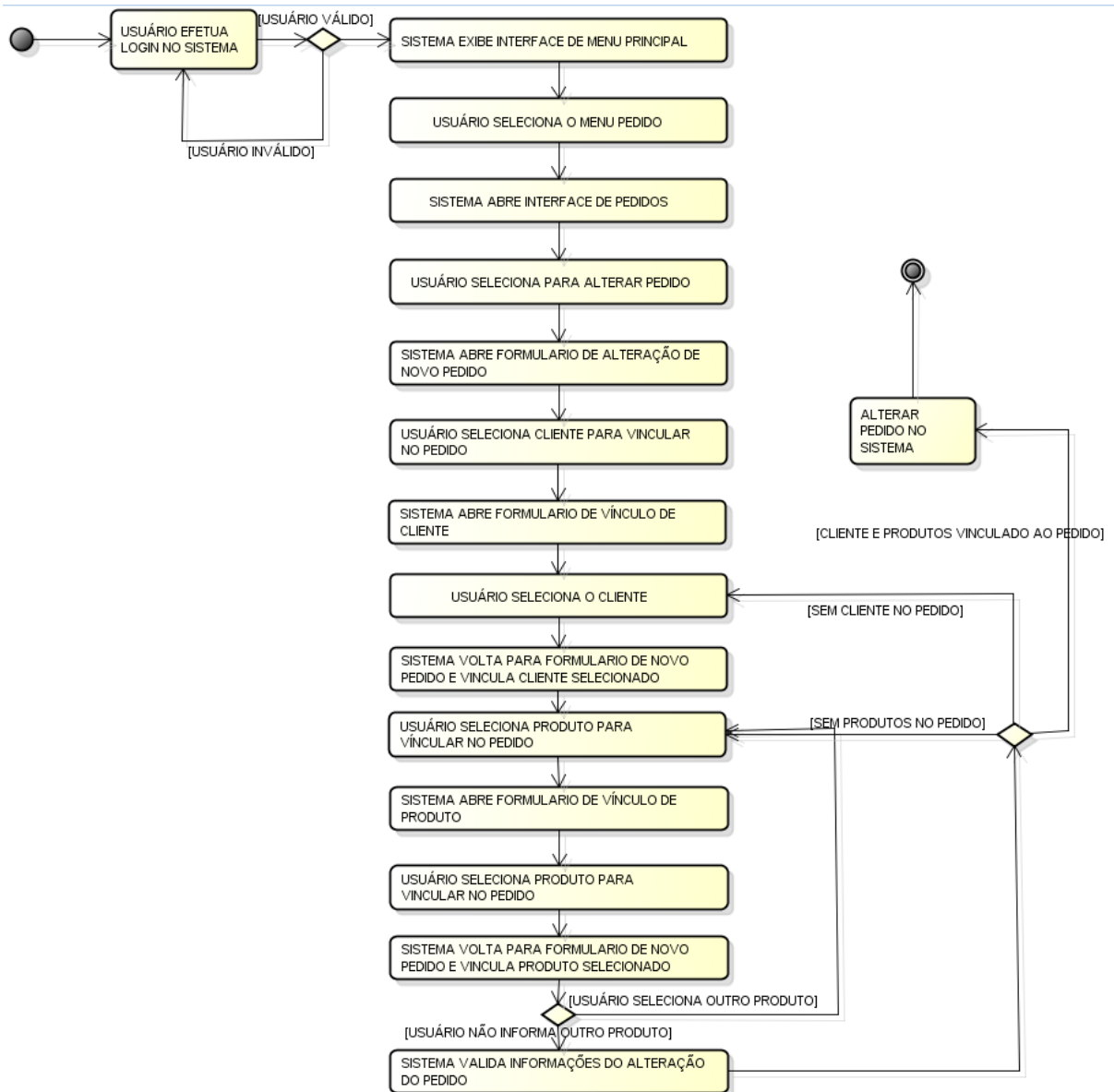


Diagrama de Atividade Excluir Pedido no Sistema

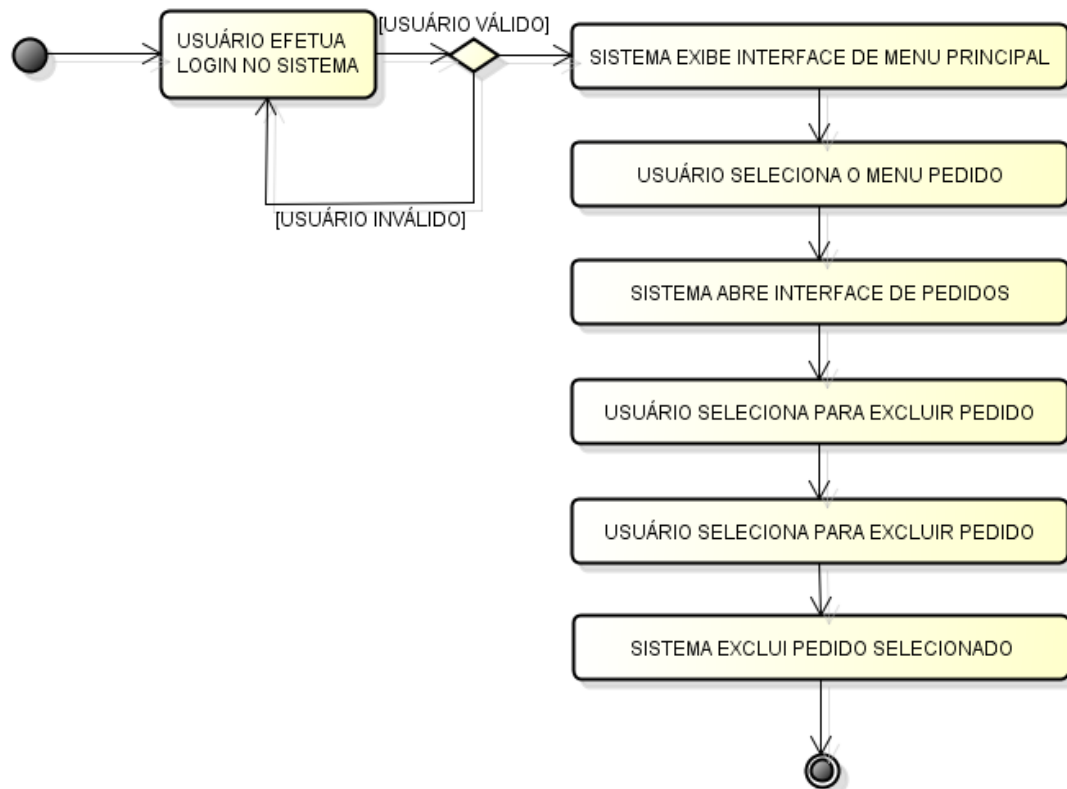
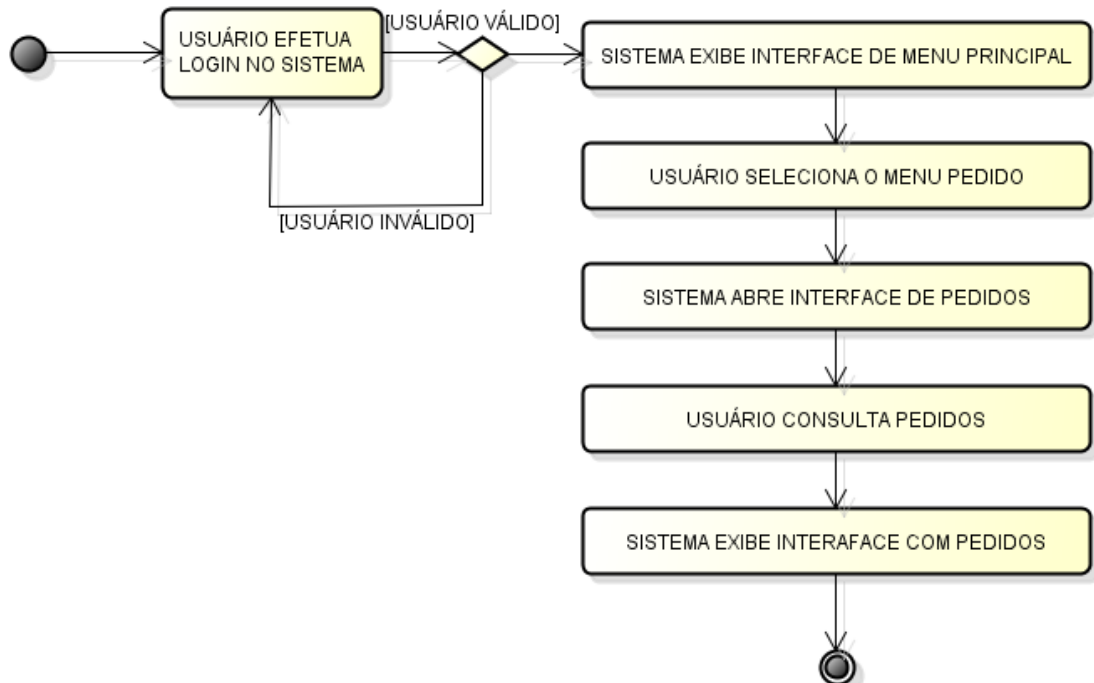


Diagrama de Atividade Consultar Pedido no Sistema



APÊNDICE D – Entrevista com vendedora de cosméticos

Foi realizada entrevista com uma representante de cosméticos de uma marca famosa, que disponibiliza para suas consultoras uma revista com códigos de produtos para pedidos. Esta vendedora dispõe da revista, do catálogo de produtos, caderno de pedidos e uma calculadora.

Foram levantadas suas necessidades para suprir dificuldades do seu dia-a-dia de trabalho externo na entrevista que segue abaixo:

1. Qual sua maior dificuldade no trabalho externo?
 - a. A maior dificuldade é de encontrar uma forma de organizar as vendas, e sempre ter disponíveis em mãos, já que o caderno de pedidos quando é perdido, também se perde todas as vendas registradas.
2. Quais funcionalidades o sistema deve conter?
 - a. Cadastro do cliente, cadastro de produtos e controle da venda.
3. Como o sistema deve funcionar na sua concepção?
 - a. O sistema tem que fazer a venda, onde na venda deve ser vinculado um cliente, vários produtos, mostrando as informações dos produtos, podendo colocar a quantidade desejada. Também deve fornecer o valor total e desconto para venda em questão.
4. Seria de ajuda a exclusão e alteração da venda, cliente e produto?
 - a. Sim, pois há necessidade de excluir algum produto que já saiu de linha, ou um cliente que mudou de cidade, ou até mesmo uma venda que não quero armazenar-la.
5. Acha que um sistema em dispositivo móvel em Android iria beneficiá-la no dia-a-dia?
 - a. Sim, pois poderia dispensar o uso da calculadora, catálogo, e caderno de pedidos, e poderia ter em minhas mãos todos os clientes, vendas e produtos sem precisar ter a revista para saber o preço. Por exemplo, quando estiver na rua sem intenções de vender, e achar algum cliente interessado no valor de algum produto.
6. Você possui dispositivo móvel com sistema operacional Android? Se sim usa internet móvel?
 - a. Sim, possuo um celular com sistema Android, e utilizo a internet móvel, pois tenho várias clientes que me contata por aplicativo de mensagens para fazer novas vendas.