

CENTRO TECNOLÓGICO POSITIVO
CURSO SUPERIOR DE TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO DE
SISTEMAS

GIULIANO HENRIQUE COSTA

HMA – SISTEMA DE GESTÃO E CONTROLE DE BARES E RESTAURANTES
Projeto de intervenção tecnológica

CURITIBA 2015

GIULIANO HENRIQUE COSTA

HMA – SISTEMA PARA GESTÃO E CONTROLE DE BARES E RESTAURANTES

Projeto de Intervenção Tecnológica

Trabalho de Conclusão de Curso apresentado
ao Programa de Aplicação Profissional (PAP),
para obtenção parcial das competências ao título
de Tecnólogo em Análise e Desenvolvimento de Sistemas
pelo Centro Tecnológico Positivo – Unidade Hauer

Orientador: Profº. Allson Wagner Siviero Martins.

RESUMO

Este trabalho busca apresentar um projeto de intervenção, como parte principal do PAP 2015 (Programa de Aplicação Profissional), do Centro Tecnológico Positivo. Com objetivo principal de realizar o desenvolvimento de um sistema para controle do Bar e Restaurante Hermes Bar, que até o momento não possuía nenhum tipo de automação ou ferramenta que auxiliasse na gestão do estabelecimento. Após o mapeamento do processo da organização cliente, foi identificado possíveis melhorias, que poderiam agilizar o atendimento aos clientes e dar mais segurança na gestão e controle de caixa. Com isso, foi definido o escopo do projeto, toda a arquitetura da aplicação e planejamento para desenvolvimento de um produto com qualidade. Para o desenvolvimento do produto foram utilizadas inúmeras ferramentas como Microsoft Visual Studio e Sql Server 2012, entre outras. Foram utilizadas ferramentas para gestão do projeto como: KanbanFlow para controle e acompanhamento das tarefas, Git e GitHub como repositório de dados, meios de comunicação como WhatsApp e e-mail.

Palavras-chave: Desenvolvimento. Sistema. Software. Gestão. Eficiência.

ABSTRACT

This study aims to present an intervention project, as the main part of the PAP 2015 (Professional Application Program), Positive Technology Center. With the main objective to carry out the development of a system to control the Restaurant and Bar Hermes Bar, which has so far not had any automation or tool that would help in the establishment of management. After mapping the client organization's process was identified possible improvements that could expedite service to customers and provide more security in the management and cash control. Thus, it was defined project scope, the entire architecture of the application and planning for the development of a quality product. For product development have been used numerous tools like Microsoft Visual Studio and SQL Server 2012, among others. To project management tools were used as KanbanFlow for control and monitoring tasks, Git and GitHub as a data repository, the media as WhatsApp and e-mail.

Keywords: Development. System. Software. Management. Efficiency.

SUMÁRIO

1 INTRODUÇÃO	6
2 FUNDAMENTAÇÃO TEÓRICA.....	7
2.1 Projeto.....	7
2.1.1 Ciclo de Vida do Projeto.....	7
2.2 Análise do Projeto	8
2.2.1 BPM (Business Process Management).....	9
2.3 Diagramas.....	9
2.3.1 Diagramas de caso de uso	10
2.3.2 Diagrama de Sequência.....	10
2.3.3 Diagrama de Processos	11
2.3.4 Star UML	11
2.3.5 Bizagi Process Modeler	12
2.4 Ferramentas de Desenvolvimento	12
2.4.1 JavaScript	12
2.4.2 JQuery	12
2.4.3 IIS (Internet Information Service)	13
2.4.4 Git	13
2.4.5 GitHub.....	13
2.4.6 C# (C Sharp).....	14
2.4.7 Microsoft Sql Server.....	14
2.4.8 ASP NET	15
2.4.9 WCF (Windows Communication Foundation)	15
2.4.10 CSS (Cascading Style Sheets)	15
2.5 Outros Conceitos	16
2.5.1 DLL (Dynamic-Link Library)	16
2.5.2 Cinco Gerenciamentos	16
2.5.3 Computação em nuvem.....	17
2.5.4 Desenvolvimento Multicamadas	18
2.5.5 EAP (Estrutura Analítica do Projeto).....	18
2.5.6 Prototipagem	18
3 ORGANIZAÇÃO CLIENTE.....	19
3.1 Histórico	19
3.2 Definição de Negócio	19

3.3 Necessidades de Melhorias	19
3.4 Cenários Futuros.....	20
3.5 Visão	20
3.6 Missão.....	20
3.7 Infraestrutura e Recursos Tecnológicos.....	20
4 DIAGNÓSTICO DO AMBIENTE.....	21
4.1 Sistemas Concorrentes.....	24
5 OBJETIVOS	25
6 DESENVOLVIMENTO	26
6.1 Gerenciamento de Projeto	28
6.1.1 Gerenciamento de Comunicação.....	29
6.1.2 Gerenciamento de Requisitos.....	30
6.1.3 Gerenciamento de Configuração	30
6.1.4 Gerenciamento de Qualidade	32
6.2 Cronograma	32
6.3 Desenvolvimento Iterativo e Incremental	33
6.3.1 Concepção.....	33
6.3.2 Elaboração.....	34
6.3.3 Construção	34
6.3.4 Transição	34
6.4 Modelagem e Arquitetura.....	34
6.4.1 Diagramas de Caso de Uso	34
6.4.2 Desenvolvimento Multicamadas	35
6.4.3 Visão de casos de uso significativos.....	36
6.4.4 Diagrama de Classes de Domínio	36
6.4.5 Realização de casos de uso	37
6.4.6 Documentação dos casos de uso Significativos	41
6.5 Modelo Físico de Dados	50
6.6 Desenvolvimento.....	52
6.6.1 DAO	52
6.6.2 ENTITY	52
6.6.3 BLL	52
6.6.4 SERVICE	52
6.6.5 MODEL	53
6.6.6 WEB.....	53

7 CONSIDERAÇÕES FINAIS	53
------------------------------	----

1 INTRODUÇÃO

Este trabalho demonstra de forma prática um projeto de intervenção, que busca acima de tudo o desenvolvimento de um produto que auxilie na gestão da organização cliente.

A organização cliente é o Hermes Bar e Restaurante, empresa do ramo de bares e restaurantes, com sua fundação em 1961, é um dos primeiros bares da cidade de Curitiba-PR.

Atualmente, a organização cliente não possui nenhum sistema de gestão e controle, sendo todas as tarefas realizadas manualmente, somente com auxílio de cadernetas e comandas em papel. Com esse baixo controle dos processos, o atendimento aos clientes, que, por se tratar de um comércio é um dos principais pontos para fidelização, ficava em segundo plano, visto que grande demanda da atenção dos responsáveis ficava voltado ao caixa e acompanhamento *visual* dos produtos consumidos.

Tendo como impacto primordial a melhora na coordenação do bar, a pergunta causadora deste projeto é: **Como poderíamos melhorar a capacidade de gestão e atendimento aos clientes do estabelecimento?**

A proposta definida segue a partir da resposta: O desenvolvimento de uma aplicação web, responsiva, que utilize tecnologias como: C#, ASP .NET MVC, Sql Server e WCF, um sistema que atenda a todos os requisitos básicos de gestão, tais como: controle financeiro, controle administrativo, acompanhamento contábil e satisfação dos clientes. Que possua interação indireta com o cliente e de baixo custo para aquisição de hardware quanto de manutenção.

Com o objetivo formalizado, foi realizado uma pesquisa sobre sistemas candidatos, onde foi constatado que atualmente o mercado de softwares não possui um sistema que atenda às necessidades da organização cliente. Com isso, conseguindo um valor mercadológico do projeto de software a ser desenvolvido.

A aplicação será desenvolvida utilizando tecnologias como C# sendo a linguagem de programação principal, um WCF deverá ser desenvolvido para comunicação da interface de usuário com as regras de negócio, mantendo o desacoplamento total da aplicação. Todas as informações serão armazenadas em um banco de dados Sql Server. O front-end será desenvolvido utilizando Bootstrap como framework CSS, afim de garantir uma aplicação cem por cento responsiva, podendo ser utilizada em qualquer meio digital.

Este projeto está dividido em sete capítulos, iniciando pela demonstração da proposta do sistema, sendo seguido pela apresentação da fundamentação teórica e o desenvolvimento da aplicação. Como terceira parte é fundamentada a organização cliente, e após, no quarto capítulo é realizado o diagnóstico do ambiente. No quinto capítulo, os objetivos do software são evidenciados, subsequentemente no sexto capítulo é formulado o desenvolvimento do sistema, e por último, as considerações finais corroboram para a visão futura do projeto e a aplicação.

2 FUNDAMENTAÇÃO TEÓRICA

As pequenas e microempresas do ramo da alimentação e entretenimento noturno constituem 70% do mercado, o restante divide-se em grandes estabelecimentos e/ou grupos que possuem dois ou mais comércios ligados a um mesmo capital. E este número tende ser maior, visto que grande parte dos comércios estão na informalidade, devido principalmente a grande carga tributária e a dificuldade e tramite na elaboração e emissão de documentação necessária para abertura de uma empresa do setor.

Nos dias de hoje, os bares e restaurantes são conhecidos pelo processo de interação tanto com meios internos, quanto externos, sempre com o foco na obtenção de lucro e retorno do capital investido. Como meio interno tem-se o próprio estabelecimento, com a realização de atendimento direto ao público. Como quesito externo, temos a realização de eventos multiculturais, com a presença de bares e restaurantes como apoiadores e/ou patrocinadores, onde é divulgado a marca e particularidades do estabelecimento.

Em grande parte, os bares não possuem um método que automatize funções gerenciais. Devido ao alto custo de aquisição, implantação e manutenção de sistemas de informação disponíveis atualmente no mercado, a maioria dos bares prefere manter um controle manual de seus processos.

2.1 Projeto

Um projeto é um esforço temporário, formado por um grupo de atividades que são coordenadas e controladas com datas de início, meio e fim, buscando um resultado pré-definido. Um projeto de software consiste em basicamente cinco fases; iniciação, planejamento, execução, controle e finalização. Todas com seus papéis bem definidos onde são aplicados conhecimentos teóricos e técnicos, constituindo habilidades especiais para garantir o sucesso.

2.1.1 Ciclo de Vida do Projeto

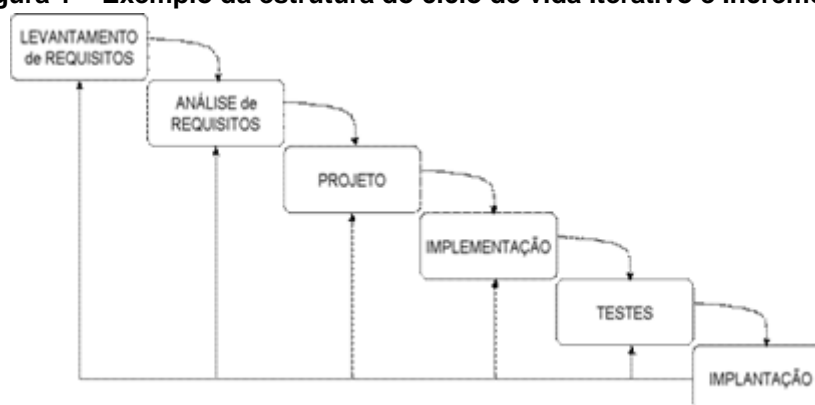
O ciclo de vida de um projeto consiste basicamente na ordenação sequencial das fases de execução do mesmo. Uma fase de um projeto é um conjunto de atividades, relacionadas de forma lógica, e que possui uma conclusão definida pela entrega de um ou mais “entregáveis”. Embora possa observar uma sequencialidade, as fases são executadas em sua maioria em paralelo, alterando somente a intensidade durante o projeto.

Um entregável é qualquer serviço ou produto, verificável e inteligível, que pode completar um projeto ou parte dele.

Existem inúmeros ciclos de vida de projeto, cada um com sua peculiaridade, porém os mais comuns são: Espiral, Cascata e Iterativo e Incremental. Este último foi escolhido para a realização deste projeto devido a sua agilidade e participação ativa da organização cliente no decorrer do projeto.

O Iterativo e Incremental busca a cada iteração, efetuar uma entrega de uma parte do software, executável, e que atenda aos requisitos básicos do cliente.

Figura 1 – Exemplo da estrutura do ciclo de vida Iterativo e Incremental



Este ciclo possui 4 fases básicas para a perfeita aplicação da metodologia no desenvolvimento de um software, são elas: Concepção, elaboração, construção e transição.

A concepção é a fase onde define-se principalmente o escopo do projeto e estima-se os prazos de cada iteração e conclusão do projeto. Na elaboração acontece a modelagem da arquitetura do sistema e levantamento dos requisitos. Construção, como o próprio nome remete, é o desenvolvimento do entregável, e a transição é o momento de entrega final do projeto.

2.2 Análise do Projeto

Inúmeros projetos tendem a fracassar devido à falta de gestão e ou falha no levantamento de certos recursos. Cenário desfavorável, objetivos mal definidos, incompreensão da complexidade do projeto, insuficiência de dados, falta de liderança, podem ser alguns dos fatores que podem acarretar na falência de um projeto.

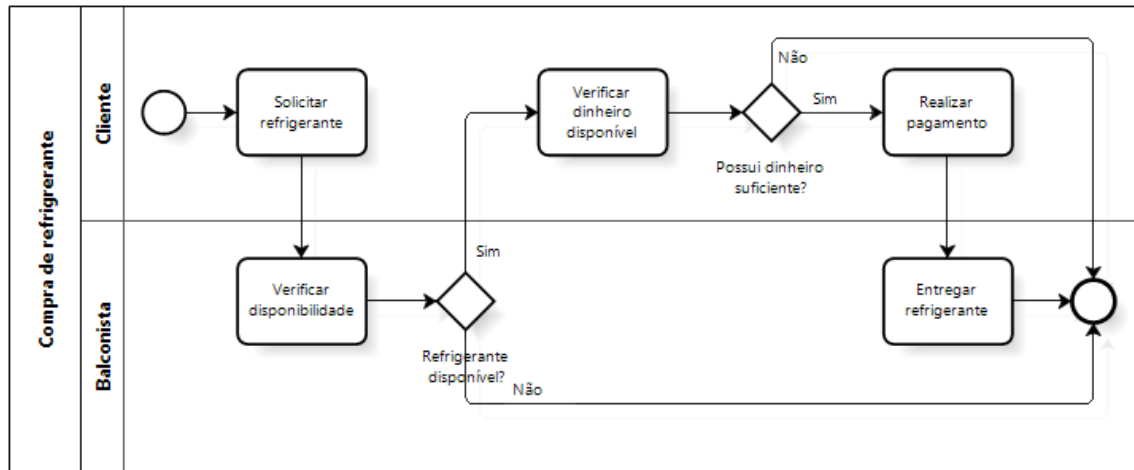
Para evitar qualquer tipo de problema com relação ao andamento do projeto, a Análise é um dos pontos primordiais. Durante esta fase é possível descobrir pontos que necessitam de uma atenção maior.

Para este projeto, iniciamos através do BPM (Business Process Management), onde foi possível compreender, mapear e monitorar todos os processos do cliente piloto.

2.2.1 BPM (Business Process Management)

O BPM tem como objetivo principal modelar fluxos de processos organizacionais. Esta metodologia permite analisar, definir, executar, monitorar e administrar processos, sempre relacionando o processo com a interação de pessoas e/ou sistemas.

FIGURA 2 – Exemplo BPMN



FONTE: IPROCESS (2015)

No diagrama acima é possível analisar o fluxo da compra de refrigerantes onde inicia-se na solicitação do mesmo até o ponto onde o refrigerante é entregue pela balconista. É possível verificar pontos de decisão no fluxo (identificados por um losango), onde é realizada a verificação de um certo elemento.

2.3 Diagramas

Para a construção de um software a utilização de diagramas torna-se obrigatório, visto que o embasamento para vários pontos do desenvolvimento tomam como base os diagramas criados em fases anteriores. A utilização de diagramas é uma forma de padronizar o entendimento de um software ou parte dele. Com isso temos a UML (Unified Modeling Language). Uma linguagem de modelagem que permite representar um sistema de forma padronizada. A UML tem como objetivo: especificação, documentação e estruturação. Podemos dividir a UML em elementos, estes elementos são agrupados em diagramas, com objetivos específicos. Estes diagramas são: Diagrama de Caso de Uso, Diagrama de Classe, Diagrama de Objetos, Diagrama de Estado, Diagrama de Sequência, Diagrama de Colaboração, Diagrama de Atividade e Diagrama de Componente.

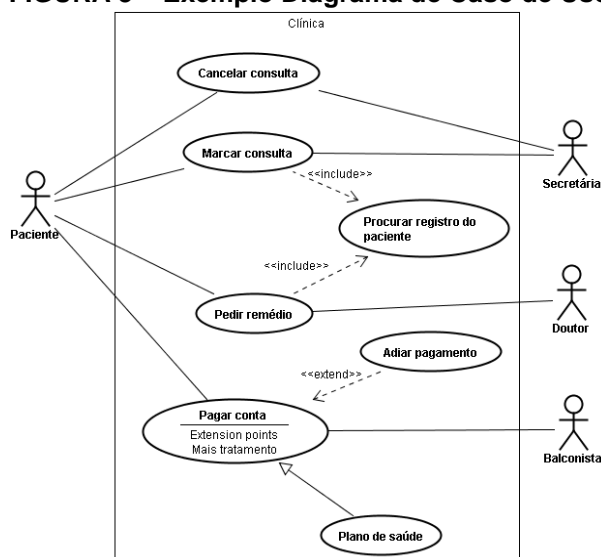
Estes diagramas são divididos em dois grandes grupos, Diagramas Estáticos, onde demonstra-se de forma sucinta, associações, dependências e refinamentos, ou seja, sua estrutura formal. Temos também os Diagramas Dinâmicos, que possuem um contexto diferenciado, sendo utilizado para demonstrar um comportamento do sistema.

2.3.1 Diagramas de caso de uso

Um caso de uso representa uma funcionalidade do sistema, subsistema ou classe, sendo executada por seus atores. Os atores são os elementos ou usuários que interagem com o sistema.

A UML dispõe de um diagrama de casos de uso, uma notação gráfica e simplificada de uma funcionalidade ou de seus conjuntos.

FIGURA 3 – Exemplo Diagrama de Caso de Uso

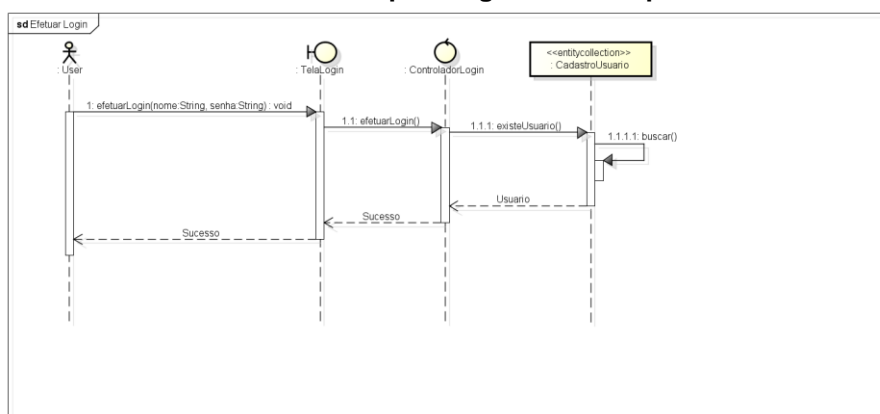


No diagrama acima é possível verificar todas as ações necessárias para realização de um agendamento de consulta. Partindo do momento da solicitação do agendamento, até o pagamento do valor correspondente.

2.3.2 Diagrama de Sequência

O Diagrama de Sequencia ou Diagrama de Mensagens representa de uma forma simplificada e lógica uma sequência de processos lógicos afim de demonstrar a interação entre objetos de um mesmo cenário.

FIGURA 4 – Exemplo Diagrama de Sequência

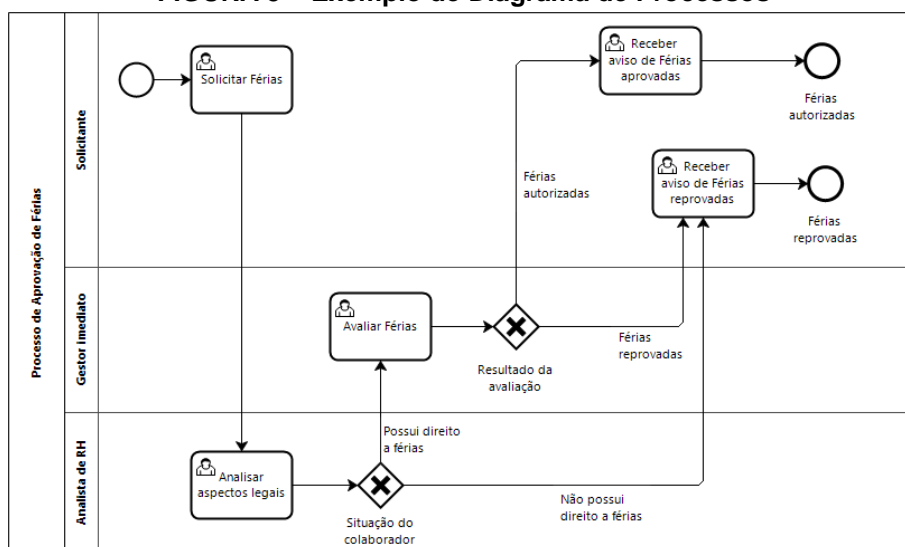


Neste exemplo acima é possível ver os passos necessários pelo software até a conclusão do caso de uso. Iniciando pelo ator e sua solicitação, seguindo o processo até o acesso a base de dados e retornando sucesso caso o seja um usuário válido.

2.3.3 Diagrama de Processos

Um Diagrama de Processos representa o processo em detalhes. Um fluxo funcional que apresenta sequencias, atividades, sendo convertidas em entradas e saídas.

FIGURA 5 – Exemplo de Diagrama de Processos



Neste diagrama de processo acima é possível analisar todo o processo de aprovação de férias, iniciado na solicitação e concluindo na autorização ou reprovação das férias.

2.3.4 Star UML

Star UML, ferramenta para criação de diagramas UML, com a possibilidade de importação e exportação de código fonte em C#. Esta ferramenta foi desenvolvida visando o sistema operacional Microsoft Windows (Versões 7,8 e 10). Por ser uma aplicação completa, ágil e de licença gratuita, foi a escolhida para ser a ferramenta de diagramação deste projeto.

FIGURA 6 – Logotipo StartUML



2.3.5 Bizagi Process Modeler

O Bizagi Process Modeler, é uma ferramenta para criação de modelagem de processos. Um de seus diferenciais é a possibilidade de testar os processos nele descritos, evitando assim duplicidade e falha nas conexões entre tarefas.

FIGURA 7 – Logotipo Bizagi Modeler



2.4 Ferramentas de Desenvolvimento

2.4.1 JavaScript

JavaScript é uma linguagem de programação, executada a partir de navegadores web. JavaScript é amplamente usada para comunicação do client-side com servidores através de requisições AJAX. Também pode-se manipular eventos e alterações dinâmicas das páginas HTML.

FIGURA 8 – Logotipo JavaScript



2.4.2 JQuery

Jquery é uma biblioteca JavaScript. Por ser um framework cross-browser, de fácil utilização, com uma sintaxe simplificada, é a biblioteca JavaScript mais utilizada. Por possuir código aberto, inúmeros plug-ins surgem a cada dia, fazendo com que a Web se torne cada vez mais dinâmica.

FIGURA 9 – Logotipo jQuery



2.4.3 IIS (Internet Information Service)

O IIS é um servidor web, fornecido pela Microsoft. Uma de suas principais características é a possibilidade de renderizar páginas HTML dinâmicas através do ASP (Active Server Pages). O IIS é capaz de rodar aplicações de inúmeras tecnologias, sendo necessário somente a instalação de plug-ins específicos.

FIGURA 10 – Logotipo Microsoft IIS 7



2.4.4 Git

O versionamento de código é indispensável no desenvolvimento de software. Controlar histórico, trabalhar em equipe, criar ramificações sem alterar o projeto principal é necessário para a construção de um sistema.

Com esse propósito, o Git é um sistema para controle de versões. Por ser um dos com melhor performance, gratuito e manter um histórico completo de todas as alterações, foi o versionador escolhido para este projeto.

FIGURA 11 – Logotipo GIT



2.4.5 GitHub

GitHub é um serviço online, para projetos que usam o Git como versionador. Este serviço é uma plataforma para hospedagem de código-fonte ou arquivos, sendo totalmente controlado pelo Git. O GitHub possui licenças comerciais, e gratuitas para projetos de código aberto.

FIGURA 12 – Logotipo GitHub



2.4.6 C# (C Sharp)

C# é uma linguagem de programação, orientada a objetos desenvolvida pela Microsoft com lançamento no ano 2000. C# foi baseado na sintaxe do C++, mas inclui referências de Pascal e Java. Para desenvolver uma aplicação com esta linguagem é necessário a máquina virtual CLR (Common Language Runtime), responsável pela transformação do código fonte em linguagem de máquina.

FIGURA 13 – Logotipo Microsoft C#



2.4.7 Microsoft Sql Server

O SQL Server é o SGBD (Sistema Gerenciador de Banco de Dados) da Microsoft. Por ser um produto de grande porte, capaz de suportar até 2000 requisições simultâneas, mesmo na sua versão Express (Versão básica do SGBD), e pela facilidade de integração com a linguagem C#, foi o sistema de banco de dados escolhido para armazenar os dados e regras de negócio da aplicação.

FIGURA 14 – Logotipo Microsoft Sql Server



2.4.8 ASP NET

É uma plataforma da Microsoft, suportada pela máquina virtual CLR, com intuito de construção de páginas Web dinâmicas. Como o ASP .NET não é uma linguagem de programação, pode ser utilizado através de linguagens suportadas pelo CLR. Uma das vantagens da utilização desta plataforma é a compilação antes da execução, trazendo grandes ganhos de desempenho.

Para a sua utilização, é necessário que a aplicação seja hospedada em servidores IIS.

FIGURA 15 – Logotipo Microsoft Asp .NET



2.4.9 WCF (Windows Communication Foundation)

O WCF é uma plataforma proprietária da Microsoft, onde busca-se a construção de aplicações distribuídas. Normalmente servido como camada de acesso a aplicações web. O WCF possui inúmeras bibliotecas permitindo o desenvolvimento de aplicações que rodem sob um servidor Microsoft Windows Server.

FIGURA 16 – Logotipo Microsoft .NET



2.4.10 CSS (Cascading Style Sheets)

CSS é uma linguagem de marcação, utilizada para definir formatações de interface para uma página web. Com uma sintaxe simples e de rápida utilização, mantém um conjunto de regras que pode ser aplicado em uma ou mais páginas.

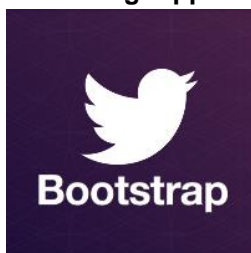
FIGURA 17 – Logotipo CSS3



2.4.10 Bootstrap

Bootstrap é um framework CSS, gratuito, e de fácil utilização. Sua principal utilidade é a rapidez na criação de front-ends responsivos, sem a necessidade de criar nenhum código css para a aplicação.

FIGURA 18 – Logotipo Bootstrap



2.5 Outros Conceitos

2.5.1 DLL (Dynamic-Link Library)

Formato desenvolvimento pela Microsoft, com o conceito de bibliotecas. É uma forma de acoplar código-fonte, fornecendo métodos a uma ou mais aplicações.

2.5.2 Cinco Gerenciamentos

Durante um projeto de software gerências são aplicadas. Essas gerências podem ser agrupadas em 5 grandes grupos. Estes grupos são:

- Gerenciamento de Comunicação;
- Gerenciamento de Configuração;
- Gerenciamento de Projeto;
- Gerenciamento de Requisitos;
- Gerenciamento de Qualidade

2.5.2.1 Gerenciamento de Comunicação

Nesta gerência são definidas as tecnologias e formato para a comunicação tanto da equipe de projeto como a comunicação com terceiros e/ou stakeholders.

2.5.2.2 Gerenciamento de Configuração

As escolhas de como controlar os arquivos de software ou documentação gerada durante todo o projeto é de responsabilidade do Gerenciamento de Configuração. Nesta gerência também temos a atribuição de definir o controle de versão do software a ser desenvolvido.

2.5.2.3 Gerenciamento de Projeto

O gerenciamento de projeto é responsável por assegurar que o projeto atinja os objetivos propostos dentro dos parâmetros definidos. Controlando prazos e cronograma.

2.5.2.4 Gerenciamento de Requisitos

Esta gerência possui o encargo de documentar e organizar os inúmeros requisitos de um sistema, todo o levantamento, e rastreabilidade de toda a documentação gerada.

2.5.2.5 Gerenciamento de Qualidade

Responsável por manter a qualidade em todos os aspectos de um projeto, inspecionando, controlando e garantindo que o desenvolvimento mantenha um padrão de qualidade estabelecido.

2.5.3 Computação em nuvem

Na computação em nuvem é aplicado o conceito de provedor e usuário, ou seja, existe uma empresa que está provendo os serviços (Amazon, por exemplo) e o usuário que acessa esses serviços através de um portal (site). Ou seja, o usuário não possui nada fisicamente em sua casa, trabalho, etc. Quem possui isso fisicamente é a empresa que provê os serviços enquanto que o usuário apenas acessa os mesmos pela internet. É importante lembrar que na computação em nuvem também é aplicado o conceito de “pay as you go” (pague pelo que usar), ou seja, você apenas está pagando pelo que utilizar dos recursos que a empresa provedora fornece.

2.5.4 Desenvolvimento Multicamadas

O desenvolvimento multicamadas é um modelo base em um projeto de software. Facilitando a manutenção, acoplamento, escalabilidade e migração.

É comum encontrar softwares definidos em 3 camadas, sendo a primeira camada responsável pela interface com o usuário. Uma camada intermediária, responsável por conter regra de negócio e uma camada de acesso a dados (comunicação com banco de dados).

2.5.5 EAP (Estrutura Analítica do Projeto)

Em resumo, as atividades a serem executadas por grupos, com objetivo específico de identificar todos os elementos presentes em um projeto. A EAP pode ser considerada um componente de visão macro do projeto como um todo.

2.5.6 Prototipagem

Prototipagem é uma forma de gerar modelos conceituais para demonstrar ao cliente o design de um projeto de software ou produto. Para a equipe de desenvolvimento torna-se uma ferramenta útil para aprovação e análise da localização exata dos atributos existentes.

3 ORGANIZAÇÃO CLIENTE

Para este projeto utilizaremos como Cliente Piloto o bar Hermes Bar e Restaurante LTDA, localizado na Av. Iguaçu, 2504 no bairro Água Verde. Tendo como principal responsável Letícia Sanches Dutra, gerente e sócia do estabelecimento.

FIGURA 19 – Logotipo Hermes Bar



3.1 Histórico

O Hermes Bar é um dos mais antigos bares da cidade de Curitiba. O estabelecimento foi fundado em 1961, e desde então mantém sua identidade histórica. O bar preserva em suas paredes, quadros com fotografias da época de sua inauguração, com as ruas ainda sem pavimentação.

Atualmente gerenciado por Letícia Sanches Dutra, uma das sócias do estabelecimento e terceira proprietária do Bar.

3.2 Definição de Negócio

Por definição, o bar é um polo de encontros para consumo de bebida alcoólica e petiscos, normalmente servidos por garçons ou *barmen*. Em suas inúmeras variações, pode-se encontrar música ao vivo ou mecânica.

Segundo o SEBRAE, é um mercado de concorrência crescente, onde em funcionamento são catalogados mais de 750 mil em todo o Brasil. Grande maioria dos estabelecimentos são pequenos negócios, normalmente classificadas como microempresas individuais ou pequenas empresas.

Algumas legislações brasileiras devem ser levadas em consideração neste trabalho. No Brasil, após a instituição da lei 8.069 de 1990, foi proibido a entrada de menores de idade em bares. Em 2008 a criação e bares em faixas de domínio da rodovia federal foram proibidas através da lei número 11.705.

3.3 Necessidades de Melhorias

Após um levantamento dos processos gerenciais do estabelecimento, foi notado que o mesmo não possui nenhum controle sobre gastos e rendimentos, estoque, funcionários e clientes, assim perdendo informações primordiais para uma boa gestão

e concorrência em um mercado tão acirrado. Com estas informações, fica claro a necessidade de implantação de um sistema que favoreça a gestão entregando informações úteis e mantendo um histórico de acontecimentos do bar.

3.4 Cenários Futuros

Por acreditar no potencial que um sistema pode prover ao estabelecimento, o proprietário nos forneceu acesso irrestrito ao estabelecimento, para que possamos em conjunto desenvolver um produto de qualidade e que possa satisfazer não só ao Hermes Bar, mas atender a necessidade de todos os envolvidos neste processo.

Acreditamos que após a informatização do estabelecimento, conseguiremos aumentar a receita em 60%, visto que, por não possuir controle atualmente, erros são constantes, tanto no fechamento das comandas dos clientes, como no processo de controle de estoque.

3.5 Visão

O cliente piloto é um ícone no ramo de bares. Durante toda sua história manteve o atendimento aos clientes como ponto primordial em seu funcionamento.

3.6 Missão

A missão do Hermes Bar e Restaurante é: “Atender com agilidade e qualidade, garantindo a fidelização do cliente”.

3.7 Infraestrutura e Recursos Tecnológicos

Atualmente o estabelecimento não possui infraestrutura suficiente para atender as solicitações mínimas que o sistema exige. O bar não possui computadores e nem smartphones e/ou tablets para realização de pedidos.

O acesso à internet é realizado através de dois pontos WiFi. O primeiro localizado no piso superior e o segundo no piso inferior.

Para que possa usufruir do sistema em toda sua plenitude, é necessária contratação de uma rede de pelo menos 30 gigabytes através de fibra ótica, podendo ser disponibilizados nos mesmos roteadores presentes hoje. A rede deverá suportar o computador do caixa, computador da cozinha, computador do bar e mais 6 smartphones e/ou tablets para realização de pedidos, somando ainda a entrega de rede para clientes.

4 DIAGNÓSTICO DO AMBIENTE

Mesmo com a facilidade de aquisição e a grande disponibilidade de aplicações para gestão de bares e restaurantes, o Hermes Bar sempre teve uma aversão a tecnologia, acreditando que tiraria a identidade histórica do estabelecimento.

Atualmente o estabelecimento mantém toda a sua gestão em cadernetas e *rascunhos*. O controle das comandas com o consumo dos clientes é através de papel, com valores fixos (foto abaixo), e numeradas sequencialmente. O bar não possui controle dos funcionários, nem do estoque. A contagem dos produtos disponíveis precisa ser realizada diariamente e visualmente. O caixa não possui um controle de abertura, fechamento, sangria e reforço. Todos os dias (de funcionamento), é retirado todo o valor, pago a diária dos funcionários, fornecedores e bandas contratadas, não havendo controle sobre o lucro total, gastos diários, impossibilidade assim uma previsão financeira.

Mas com o crescimento da quantidade clientes, a busca pela qualidade no atendimento e rapidez na tomada de decisão, a inclusão no mundo digital é obrigatória.

As informações acima citadas, foram adquiridas através da realização do BPM. Após o mapeamento de todos os processos do cliente piloto foi possível notar falhas nos processos gerenciais, e controles financeiros.

Figura 20 – BPMN Entrada do cliente sem automação

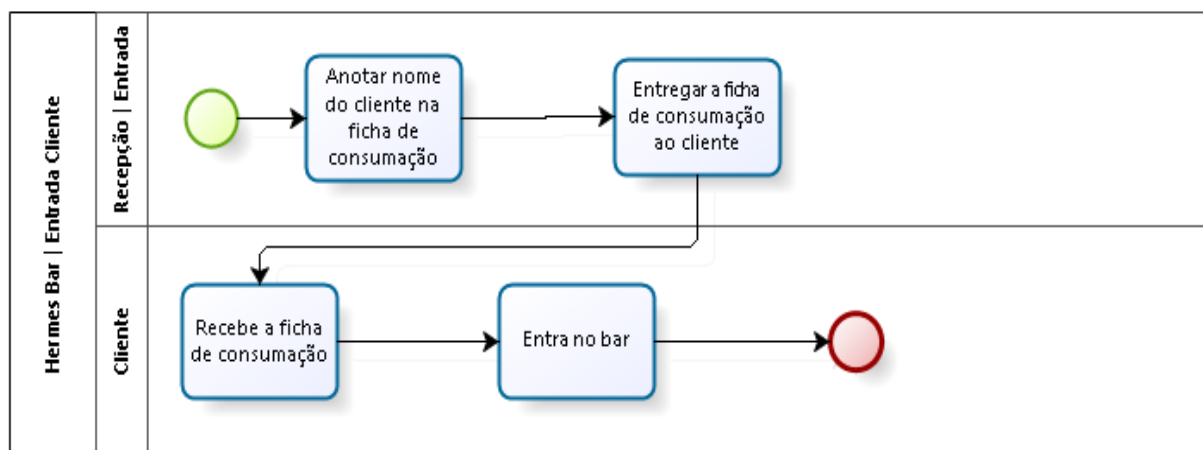


FIGURA 21 – BPMN Solicitação de Pedido sem automação

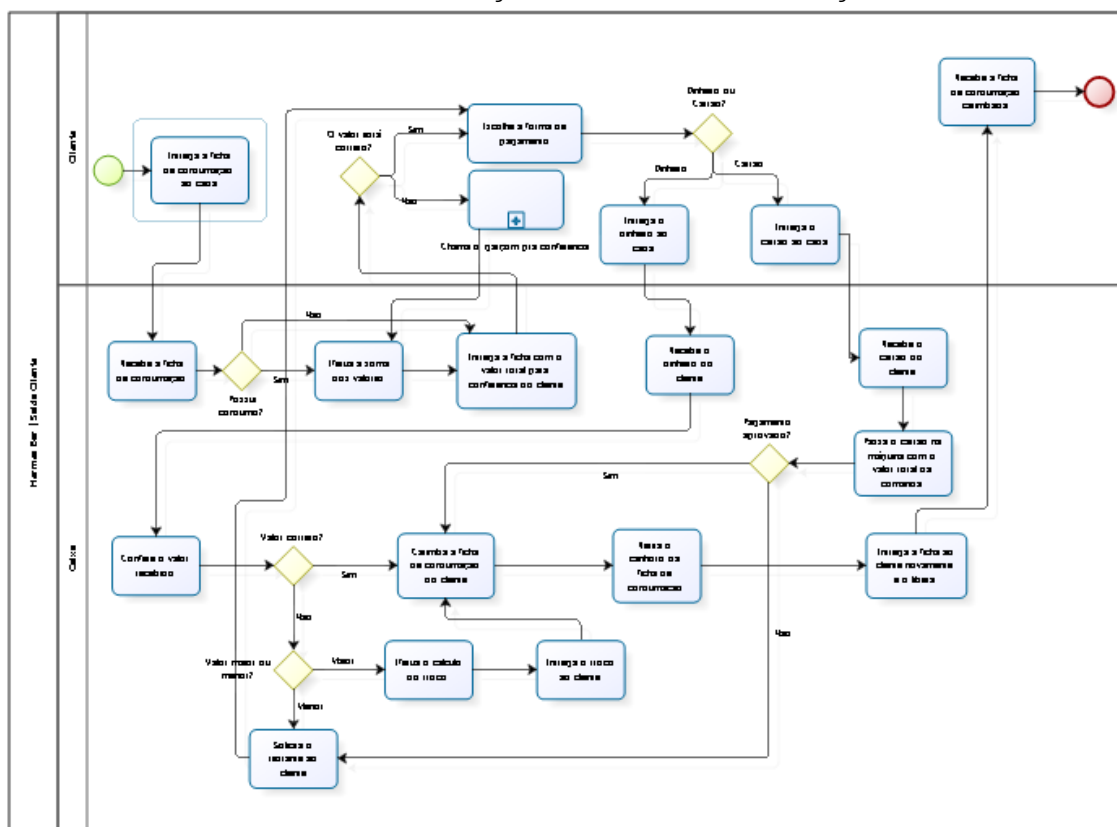


FIGURA 22 – BPMN Pedido a cozinha sem automação

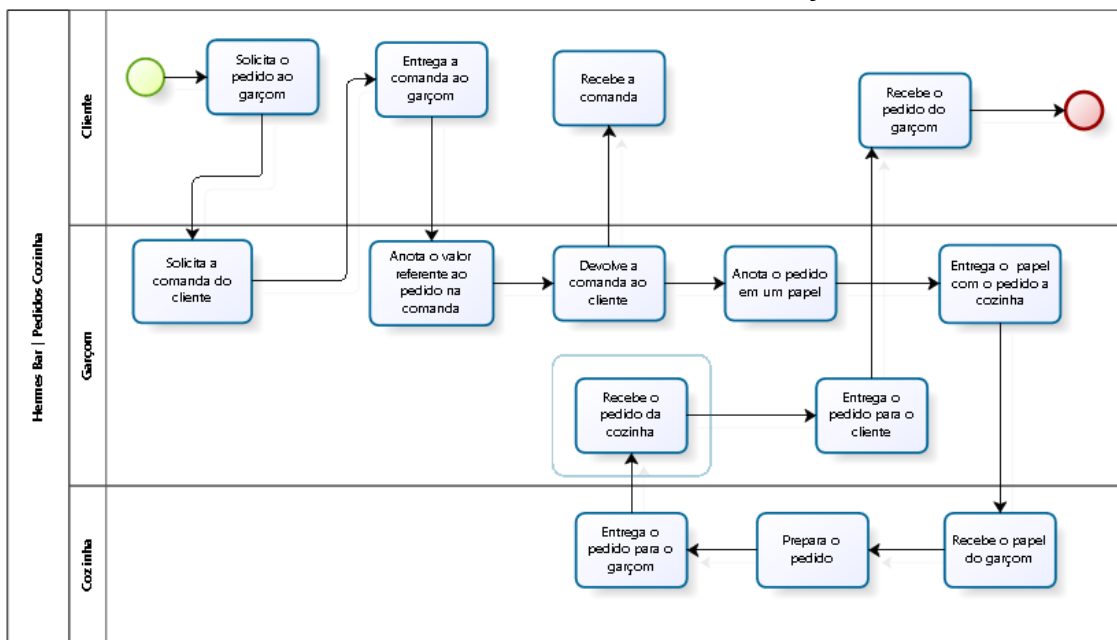
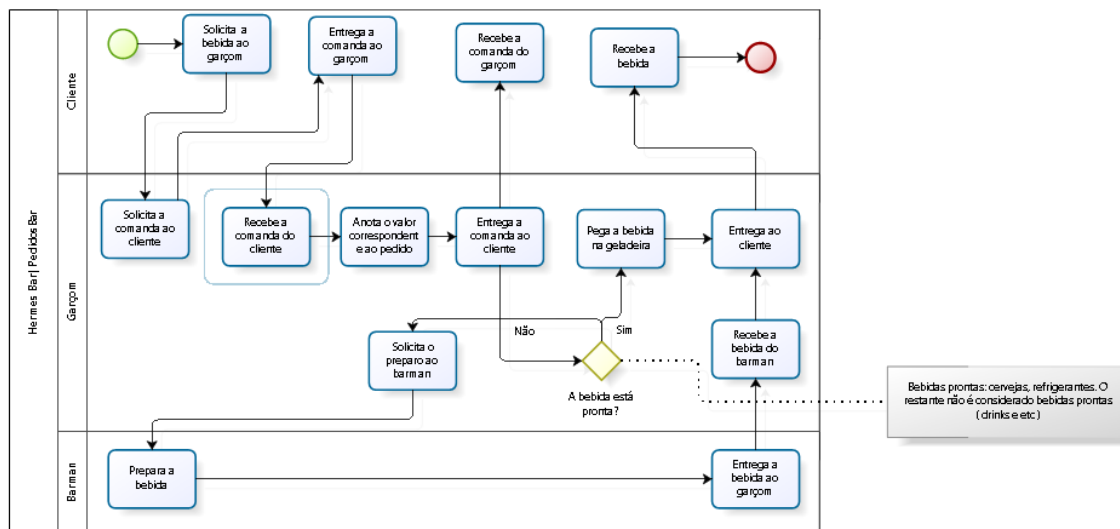


FIGURA 23 – BPMN Saída Cliente sem automação



Após a finalização dos diagramas de processos, conseguimos partir para a verificação da área de atuação, utilizamos a metodologia 5W1H, onde foi possível responder as seguintes perguntas, divididas em grupos de problemáticas, ou seja, para cada fase de levantamento, foi aplicado a metodologia afim de organizar e identificar os problemas e futuras ações corretivas.

WHAT? (O que?) – Possibilidade de gerenciar o estabelecimento com segurança e agilidade

WHO? (Quem?) – Gestores do Hermes Bar

WHERE (Onde?) – No Bar e Restaurante Hermes Bar.

WHY? (Por quê?) - Para proporcionar controle sobre o consumo, financeiro e clientes do estabelecimento, favorecendo ainda mais a qualidade do atendimento

WHEN? (Quando?) – Sempre que a gerencia achar necessário a iteração com informações gerenciais.

HOW? (Como?) – Mantendo dados importantes para posterior análise e transformação em informações.

Com as informações levantadas fica visível a necessidade de implantar um sistema de informação que possa gerenciar o estabelecimento como um todo. Na realização deste projeto, a automatização tende a dar vida nova ao estabelecimento, com crescimento de receita, visto que o controle será maior. Atendimento diferenciado, sendo que boa parte do tempo gasto com anotações ou fechamento manual das comandas pode ser investido em melhorar o atendimento aos clientes, conseguindo assim fidelização dos mesmos.

4.1 Sistemas Concorrentes

Com o ramo de bares e restaurantes em alta, inúmeros sistemas de gestão surgem a cada dia, todos com suas peculiaridades e premissas. Para que o projeto possuísse um diferencial, foi analisado os sistemas concorrentes encontrados na Web.

Foram encontrados vários softwares potenciais, com qualidade, mas grande maioria são softwares voltados a desktop, onde não utilizam tecnologia Web, ou são aplicações muito robustas, que não atendem a grade financeira do cliente piloto.

Alguns encontrados foram:

- Altecซิส: Possui uma grande gama de produtos, mas é necessário a aquisição em módulos separados para conseguir gerir o estabelecimento como um todo;
- eComanda: Plataforma Web, mas com um custo mensal elevado;
- GRFood: Sistema Desktop, com uma interface não amigável, dificultando a utilização.
- Cheff Solutions: Dividido por módulos, a solução disponível pela empresa possui uma grande quantidade de funções, porem com um preço não acessível ao cliente piloto.

Após análise de inúmeros sistemas concorrentes, foi constatado que o mercado carece de uma aplicação de baixo custo, acessível de qualquer lugar e que possa, de maneira confiável, auxiliar na administração do estabelecimento.

Essas informações fortalecem a ideia de que o desenvolvimento do HMA pode ser um diferencial para o Hermes Bar

5 OBJETIVOS

A partir dos requisitos adquiridos com o cliente piloto, este projeto tem como objetivo o desenvolvimento de uma aplicação Web, utilizando tecnologias Microsoft como: C#, Asp NET, Sql Server como armazenamento de dados, Visual Studio como ferramenta para desenvolvimento, WCF para comunicação entre plataformas. Estas tecnologias aplicadas em conjunto forneceriam aos gestores e de mais *stakeholders*, um sistema de gestão, onde seria possível a emissão de pedidos, abertura e fechamento de comandas, controle de caixa e estoque.

Como objetivo gerencial, o sistema tem a premissa de manter uma base de dados, que após passar um por tratamento, possam gerar informações necessárias.

Como objetivos gerais, tem-se:

- Desenvolvimento de um sistema de gestão que proporcione aos gestores conferir e armazenar informações pertinentes;
- Auxiliar o bar no controle de clientes;
- Auxiliar o bar no controle de reservas;
- Auxiliar o bar no controle de estoque;
- Auxiliar o bar no controle de fornecedores;
- Contribuir para a satisfação do cliente.

Como objetivos específicos desse projeto, tem-se:

- Cadastrar clientes;
- Cadastrar funcionários;
- Realizar a entrada e saída de clientes;
- Realizar o controle de estoque;
- Cadastrar fornecedores;
- Cadastrar Produtos;
- Realizar abertura e fechamento de caixa;
- Realizar sangria de caixa;
- Realizar reforço de caixa;
- Emitir relatórios de fluxo de caixa diários;
- Imprimir cupom *não* fiscal (*versão acadêmica*);
- Realizar consultas de todos os casos de cadastro;
- Prover segurança dos dados armazenados na nuvem;
- Criar metodologia de armazenamento local em caso de falha de conexão com a internet;
- Desenvolver aplicação responsiva para utilização em tablets e aparelhos móveis;

6 DESENVOLVIMENTO

O início do projeto teve no mês de julho de 2015, após reunião com gestores do Hermes Bar e Restaurante. Logo em seguida foi realizada uma pesquisa com intuito de descobrir possíveis sistemas que pudessem atender os requisitos de negócio, web e com um baixo custo de contratação. Como não foram encontrados softwares de mercado com as características necessárias, foi decidido iniciar um projeto de software para atender ao cliente piloto.

O primeiro processo a ser realizado, foi a construção do Documento de Visão. Este documento tem como objetivo capturar as perspectivas necessárias, pessoas interessadas no projeto e a identificação de possíveis problemas. Para este documento é necessário o reconhecimento dos *stakeholders*, conforme demonstrado tabela a seguir:

Tabela 1 - Stakeholders

Hermes Bar e Restaurante	
Nome	Função
Leticia Sanches Dutra	Sócia Gerente
Silvana Brainta	Sócia
Giuliano Costa	Gerente de Projeto Desenvolvedor Analista de Teste Analista de Requisitos
Allston Wagner Siviero Martins	Prof. Orientador

FONTE: O autor (2015)

Como a equipe de desenvolvimento do projeto é composta de somente uma pessoa, foram definidos seis papéis afim de manter a organização. Estes papéis possuem definições específicas, conforme demonstrado na tabela a seguir:

Tabela 2 – Definição das funções

Desenvolvimento	
Papel	Função
Gerente de Projetos	Manter a integração entre as partes interessadas, solucionar problemas gerenciais e manter o andamento do projeto no período especificado. Esta função percorre todo o projeto.
Desenvolvedor	Codificar o sistema com base nas especificações levantadas pelo Analista de Requisitos. Preparar e desenvolver a base de dados para a aplicação.
Analista de Requisitos	Identificar todos os requisitos necessários para a construção do software.

Analista de Testes	Desenvolver e aplicar casos de testes em todas as instâncias do processo de criação do software.
Arquiteto de Software	Definir e auxiliar na construção da arquitetura de software, definindo camadas e tecnologias a serem utilizadas
Engenheiro de Software	Neste projeto atua como um mediador entre a gerencia do projeto e as demais funções, auxiliando na tomada de decisões.

FONTE: O autor (2015)

Após o levantamento dos *stakeholders*, uma segunda fase importante para a construção do Documento de Visão é a obtenção do cenário atual. Este cenário demonstra de forma conceitual todas as referências presentes no estabelecimento, conforme demonstrado no apêndice deste trabalho.

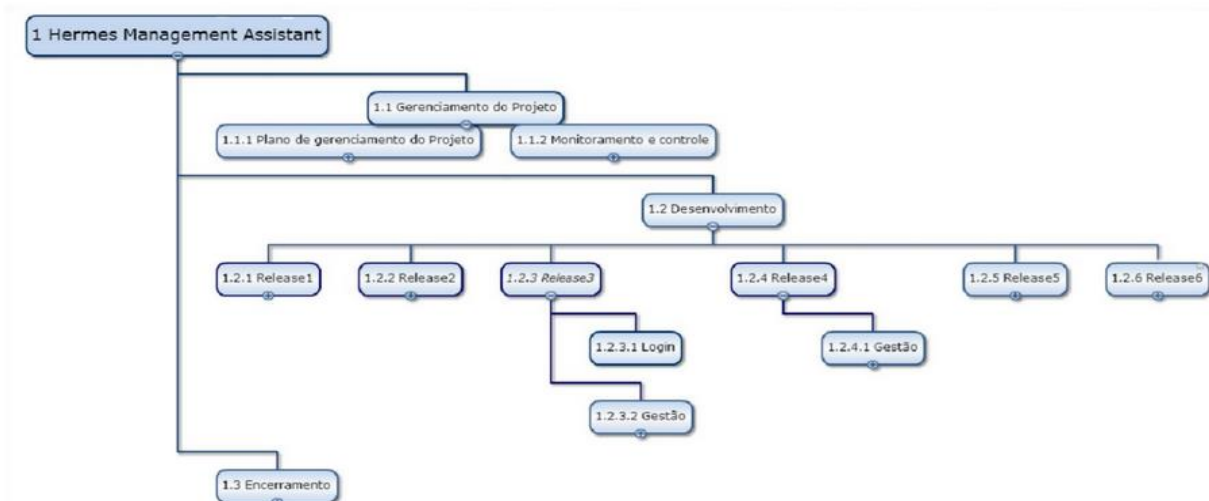
A descrição do projeto vem logo em seguida, mapeando todos os processos necessários para a obtenção da visão macro do projeto a ser desenvolvido. Juntamente com a descrição do projeto é criada a abrangência, ou seja, quais áreas serão impactadas com o desenvolvimento.

No Documento de Visão é necessário a instanciação das restrições e premissas. Entende-se como restrição toda e qualquer negativa referente a todo o processo, por exemplo: *“O projeto não terá integração com sistema de terceiros”*. Como premissa é utilizada uma afirmação, normalmente baseada na conclusão, como por exemplo: *“O sistema será desenvolvido somente para plataforma Microsoft Windows”*.

Com o documento de visão em mãos e com uma expectativa de entrega realizada, foi possível definir os papéis de cada função dentro das cinco gerências escolhidas.

6.1 Gerenciamento de Projeto

Com intuito de manter a organização no andamento do projeto, foi desenvolvido a EAP (Estrutura Analítica do Projeto), para identificação das atividades.



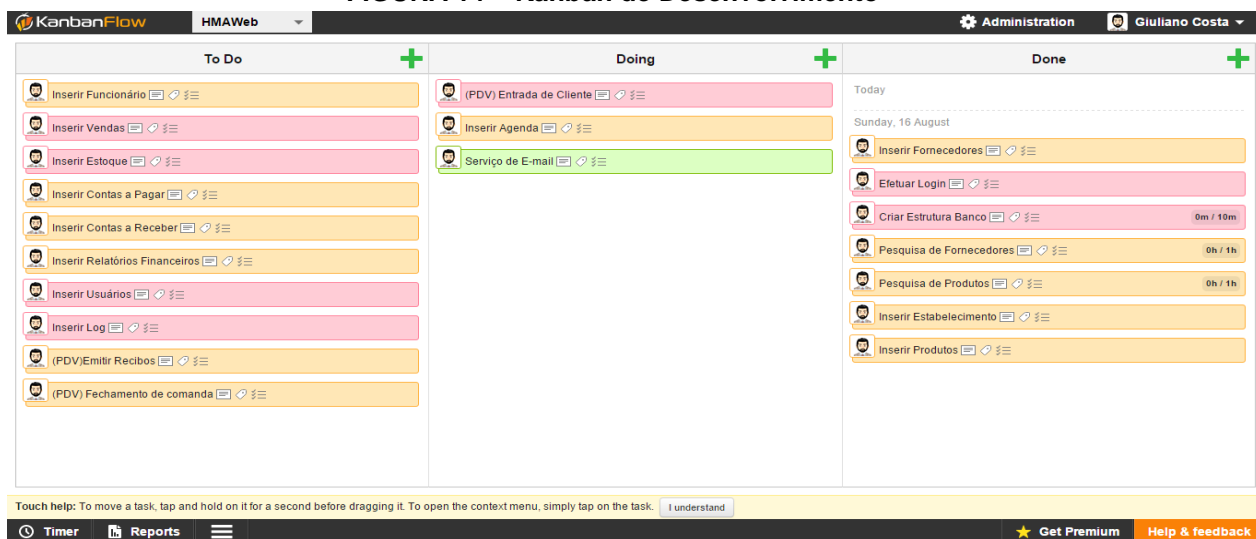
O Kanban foi utilizado para gerenciar cada tarefa referentes a cada grupo de atividade identificados na EAP. Todas as atividades de desenvolvimento foram divididas em dois grandes grupos, estes grupos são: Documentação e Desenvolvimento. Esta divisão foi necessária para manter uma organização nas tarefas e priorização de atividades dentro do projeto.

Foi utilizado o Kanbanflow como ferramenta de controle das atividades e dos boards. No board referente a documentação, todas as atividades de levantamento de requisitos, documentação do projeto, documentação do projeto acadêmico, criação dos protótipos e mapeando das funções estavam presentes. No board de desenvolvimento todas as tarefas referentes a codificação do sistema estavam desenhadas, por exemplo: criação do projeto de regra de negócio, configuração do banco de dados, criação de procedures de consulta.

Em ambos os quadros as atividades foram divididas em 4 sub-grupos, estes eram:

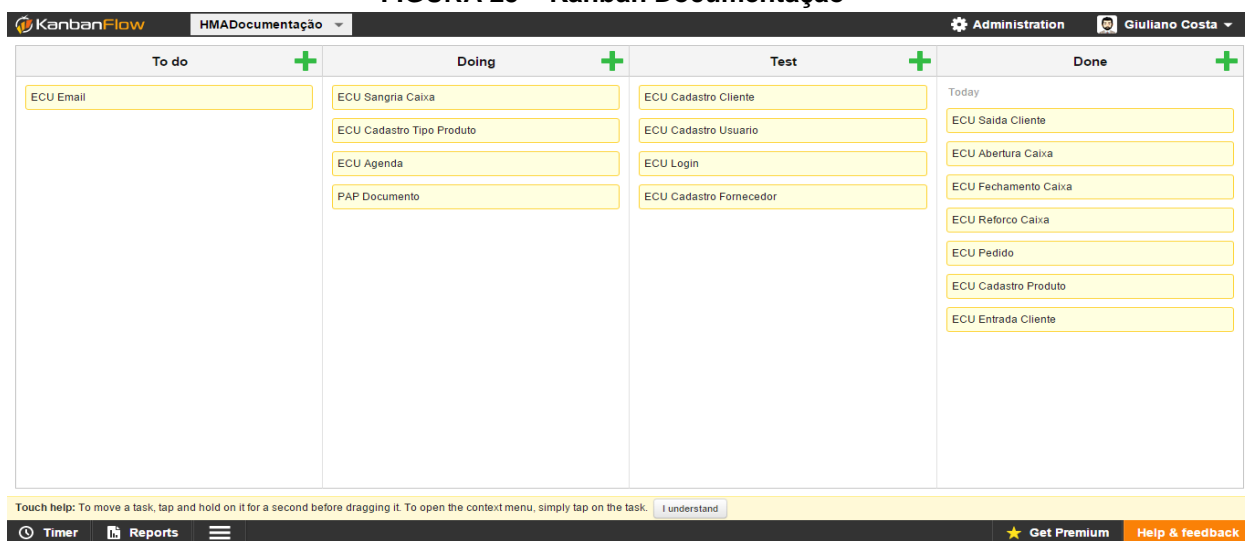
- To Do – Atividades a serem realizadas ou aguardado especificação.
- Doing – Atividades em desenvolvimento.
- Test – Teste ou revisão das atividades concluídas.
- Done – Atividades concluídas, testadas ou revisadas.

FIGURA 14 – Kanban de Desenvolvimento



FONTE: O autor (2015)

FIGURA 25 – Kanban Documentação



FONTE: O autor (2015)

6.1.1 Gerenciamento de Comunicação

O Gerenciamento de Comunicação é a gerencia responsável por determinar como será o processo de comunicação entre as partes interessadas.

Para a comunicação, foram utilizadas duas tecnologias distintas:

- E-mail – O correio eletrônico foi utilizado como ferramenta principal de comunicação. Esse formato foi escolhido devido a facilidade de armazenar e organizar informações e por ser possível a integração de anexos.

- WhatsApp: O aplicativo de mensagem instantânea foi utilizado com menor frequência, somente para aquisição de informações rápidas e agendamento de reuniões com a orientação deste projeto.

Foram realizadas também reuniões presenciais. Com os responsáveis pelo Hermes Bar as reuniões eram realizadas durante horário de funcionamento do mesmo, normalmente aos sábados, da 00:00 até 01:30. Com a orientação deste projeto foram realizadas cinco reuniões presenciais, todas em horário acadêmico.

6.1.2 Gerenciamento de Requisitos

O gerenciamento de requisitos é uma atividade de grande valor ao processo de construção do software. Responsável pelo levantamento, armazenamento e manutenção dos requisitos necessários para a implementação do software e validação dos testes.

Para esta gestão as seguintes ferramentas foram utilizadas:

- Star UML: Com o Star UML foi possível a criação dos diagramas referentes ao desenvolvimento do software. Diagramas de casos de uso, Diagramas de Sequência, Diagramas de Pacote foram alguns dos documentos armazenados nesta ferramenta. A ferramenta ainda disponibiliza a exportação dos diagramas desenvolvidos em código fonte, mais especificamente C#.
- Bizagi Process Modeler: Esta ferramenta foi utilizada para o mapeamento dos processos da organização cliente, conseguindo assim um entendimento de como as tarefas são realizadas no nosso cliente piloto.

Com os diagramas e documentos gerados nesta sessão, foi possível administrar os requisitos necessários para a construção do software e obtenção de sucesso na qualidade do mesmo.

6.1.3 Gerenciamento de Configuração

O gerenciamento de configuração é um grupo de atividades a fim de manter o controle de alterações, versão e desenvolvimento do software.

Para esta gerencia, foi utilizado o GIT como ferramenta de versionamento afim de controlar tanto o código-fonte do sistema, quando a documentação.


```

MINGW32: c:/Users/Giuliano/Desktop/HermesBar
delete mode 100644 HermesBar/HermesBarAPP/HermesBarAPP/www/js/app.js
delete mode 100644 HermesBar/HermesBarAPP/HermesBarAPP/www/js/init-app.js
delete mode 100644 HermesBar/HermesBarAPP/HermesBarAPP/www/lib/jquery.min.js
delete mode 100644 HermesBar/HermesBarAPP/HermesBarAPP/www/marginal/marginal-position.min.js
delete mode 100644 HermesBar/HermesBarAPP/HermesBarAPP/www/xdk/init-dev.js

Giuliano@GIULIANOCOSTA ~/Desktop/HermesBar (develop)
$ git push origin develop
Username for 'https://github.com': giulianocosta@outlook.com
Password for 'https://giulianocosta@outlook.com@github.com':
Counting objects: 13, done.
Delta compression using up to 4 threads.
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 349 bytes | 0 bytes/s, done.
Total 3 (delta 1), reused 0 (delta 0)
To https://github.com/hermesbar/HermesBar.git
   0fc54a6..c3df74a  develop -> develop

Giuliano@GIULIANOCOSTA ~/Desktop/HermesBar (develop)
$ git checkout documentacao
Switched to branch 'documentacao'

Giuliano@GIULIANOCOSTA ~/Desktop/HermesBar (documentacao)
$

```

FONTE: O Autor (2015)

Esta ferramenta possibilitou a criação de dois *branchs* de trabalho. O primeiro *branch* foi nomeado de *Documentação*, neste ambiente foram armazenados todos os documentos referente tanto ao projeto de software quanto a documentação acadêmica. No segundo *branch*, chamado de *Develop*, foi utilizado como repositório de código-fonte e *script's* para geração de banco de dados. Todos os projetos e sub-projetos foram armazenados nos servidores do GitHub.

Para o controle da versão do software foram definidos três grupos:

- Beta: Possuem uma formação completa, mas ainda não disponível para comercialização, utilização em ambiente de produção ou entrega ao cliente.
- RC (Release Candidate): Pronto para utilização em produção, mas não estável, ou seja, ainda pode passar por pequenas modificações.
- Stable: Software ou parte de concluído de fato. Ou seja, o sistema não passará por mudanças ou atualizações.

Para identificação dos grupos de versões e suas subdivisões, foi adotado uma nomenclatura numérica seguindo a regra: 0.00.00.000 Onde:

Tabela 3 – Especificação de modelo de versão

0.	00.	00.	000
Versão do grupo Stable	Versão do grupo Release Candidate	Versão do grupo beta	Versão de correção.

6.1.4 Gerenciamento de Qualidade

Para uma gestão completa da qualidade, é necessário aplicar metodologias, tanto no processo de gerenciamento do projeto quanto no produto a ser desenvolvido.

Para garantir a satisfação do cliente piloto e o bom funcionamento da aplicação, foram adotadas duas metodologias de teste de software.

- Testes unitários: Foram realizados testes unitários em todos os casos de uso significativos, a fim de garantir que o fluxo principal da aplicação estivesse dentro da expectativa do cliente piloto;
- Testes de regressão: Após conclusão de módulos ou atividades de desenvolvimento eram realizados testes de regressão na aplicação, visando garantir que não houve nenhuma alteração nos processos já desenvolvidos e entregues anteriormente.

Os testes unitários foram realizados através do Visual Studio 2013 Ultimate, que prove uma gama de ferramentas e plug-ins para realização de testes. Os testes de regressão eram realizados manualmente, pela equipe de desenvolvimento do sistema.

6.2 Cronograma

Com o conhecimento do ambiente do cliente piloto, o documento de visão concluído e especificado, a definição das gerencias a serem utilizadas, foi possível estabelecer e validar um cronograma do projeto.

O cronograma foi baseado principalmente no calendário acadêmico, visto que datas de entregas e conclusão do curso são fixas. Com isso, foi adotado que o projeto deveria ser entregue em sua primeira fase no dia 04 de dezembro de 2015.

Tabela 04 – Cronograma do Projeto

ETAPA	RESPONSÁVEL	DE	ATÉ
Definição do tema	GIULIANO COSTA	JUN(1)	JUN(5)
Elaboração do Documento de Visão	GIULIANO COSTA	JUN(6)	JUN(12)
Protocolo Ficha de Inscrição	GIULIANO COSTA	JUN(9)	JUN(10)
Pré – Projeto	GIULIANO COSTA	JUN(13)	AGO(1)
Desenvolvimento Software	GIULIANO COSTA	JUN(11)	SET(6)
Projeto Parcial	GIULIANO COSTA	SET(5)	SET(6)
Pré Banca	GIULIANO COSTA	OUT(15)	OUT(30)

	ALLSTON WAGNER		
Avaliação	GIULIANO COSTA ALLSTON WAGNER	DEZ(4)	DEZ(5)

Fonte: O autor (2015)

6.3 Desenvolvimento Iterativo e Incremental

O desenvolvimento Iterativo e Incremental foi adotado como metodologia para o desenvolvimento deste projeto. Optou-se por esta, por ser uma metodologia flexível e com entregas parciais, foi fundamental para avaliação do andamento e qualidade do sistema a ser desenvolvido.

Foram realizados seis ciclos para cada iteração, estes ciclos foram:

- Modelagem: Ciclo responsável por realizar o entendimento das regras de negócio do cliente.
- Requisitos: Ciclo responsável por realizar a criação e edição dos requisitos do sistema.
- Analise: Ciclo responsável por realizar o estudo dos requisitos e definição das necessidades tecnológicas.
- Implementação: Codificação do sistema com base nos requisitos e analise levantados.
- Testes: Neste ciclo eram realizadas as validações referentes a cada caso de uso.
- Implantação: Último ciclo de cada iteração, responsável pela entrega de cada versão release candidate.

6.3.1 Concepção

A concepção ou início, foi na conclusão do Documento de Visão e na reunião para entrega do mesmo documento, onde foi assinado por todos as partes interessadas do projeto e fixado uma data macro para conclusão do projeto.

6.3.2 Elaboração

O marco para a fase de elaboração do projeto foi definido como o levantamento dos casos de uso significativos, com isso obtendo uma visão melhorada do software e suas áreas críticas.

Os casos de uso significativos foram de extrema importância nesta fase, visto que após a criação e análise dos mesmos foi possível determinar uma arquitetura de software satisfatória e que atendesse ao projeto como um todo.

6.3.3 Construção

A construção inicia-se no desenvolvimento da aplicação, ou seja, na codificação do mesmo. Nesta fase o foco era em agilizar o processo de desenvolvimento, utilizando ferramentas e bibliotecas para que o processo tivesse qualidade, segurança e velocidade em sua utilização.

Nesta fase também são criados todos os testes unitários, e aplicados a ao processo de desenvolvimento.

6.3.4 Transição

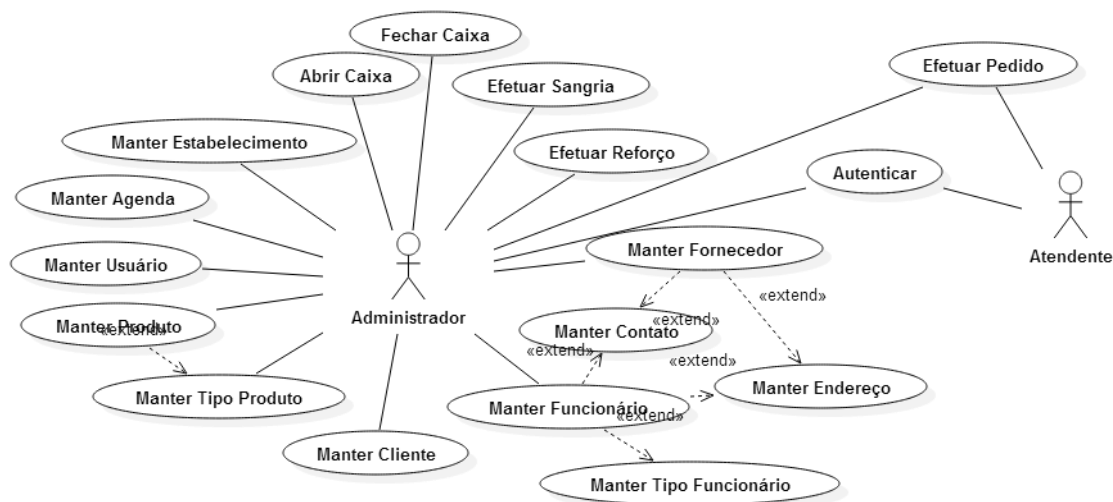
Para este projeto a transição acontece no momento pós apresentação à banca avaliadora acadêmica. Onde é realizado a apresentação final do software e entregue a fase um ao cliente piloto.

6.4 Modelagem e Arquitetura

6.4.1 Diagramas de Caso de Uso

Um diagrama de caso de uso é um documento que descreve uma sequência de eventos através de um ator inicial. Esse ator pode ser tanto um usuário da aplicação quanto uma outra parte do sistema que dispara a utilização de um método corrente.

Por ser um diagrama de fácil visualização e entendimento do fluxo, a utilização do mesmo, e de grande importância no processo de documentação de um sistema.

FIGURA 25 – Diagrama de Caso de Uso HMA

FONTE: O autor (2015)

Neste diagrama é possível visualizar a interação dos atores com as suas funcionalidades respectivas. O usuário Administrador, possui acesso a todos casos de uso de administração, enquanto o ator Garçom possui acesso somente ao módulo de pedidos. Por esta divisão entre atores, e por adotarem tecnologias de interface diferentes, foi aplicado o desenvolvimento multicamadas.

6.4.2 Desenvolvimento Multicamadas

O desenvolvimento multicamadas fornece um desacoplamento, e facilita uma construção genérica da aplicação. Como o sistema passara por inúmeras fases de desenvolvimento, a utilização desta arquitetura fornece uma facilidade na criação e utilização de componentes individuais.

As camadas utilizadas para construção desde software foram:

- DAO – Camada responsável por acesso a banco de dados;
- BLL – Camada responsável por conter regras de negócio;
- ENTITY – Camada responsável por conter objetos de banco de dados;
- MODEL – Camada responsável por conter modelos de view;
- SERVICE – Serviço responsável por efetuar a comunicação da aplicação web com a camada de regra de negócio;
- VIEW – Camada responsável por administrar e exibir as views ao cliente.

A seguir será demonstrado como a arquitetura multicamadas foi aplicada, através das visões de caso de uso significativos, logica e visão de implantação é possível analisar como a estrutura é adaptável a qualquer tecnologia e fornece uma agilidade no desenvolvimento de novas funcionalidades.

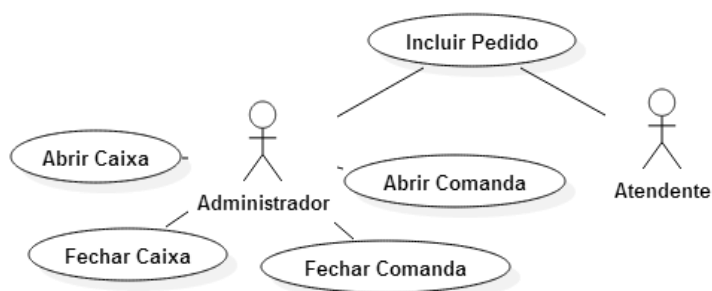
6.4.3 Visão de casos de uso significativos

Esta seção apresenta os casos de uso mais significativos para o software e devido a isso, são utilizados para validação da arquitetura proposta.

- Incluir Pedido: Este caso de uso descreve como inserir um pedido no sistema, o pedido deverá ser realizado pelo ator Garçom;
- Abrir Caixa: Este caso de uso descreve o processo de abertura do caixa na aplicação;
- Fechar Caixa: Este caso de uso descreve o processo de encerramento de caixa;
- Fechar Comanda: Este caso de uso descreve o processo de pagamento e encerramento de comanda do cliente;

Na imagem abaixo está exposta a estrutura dos casos de uso significativos.

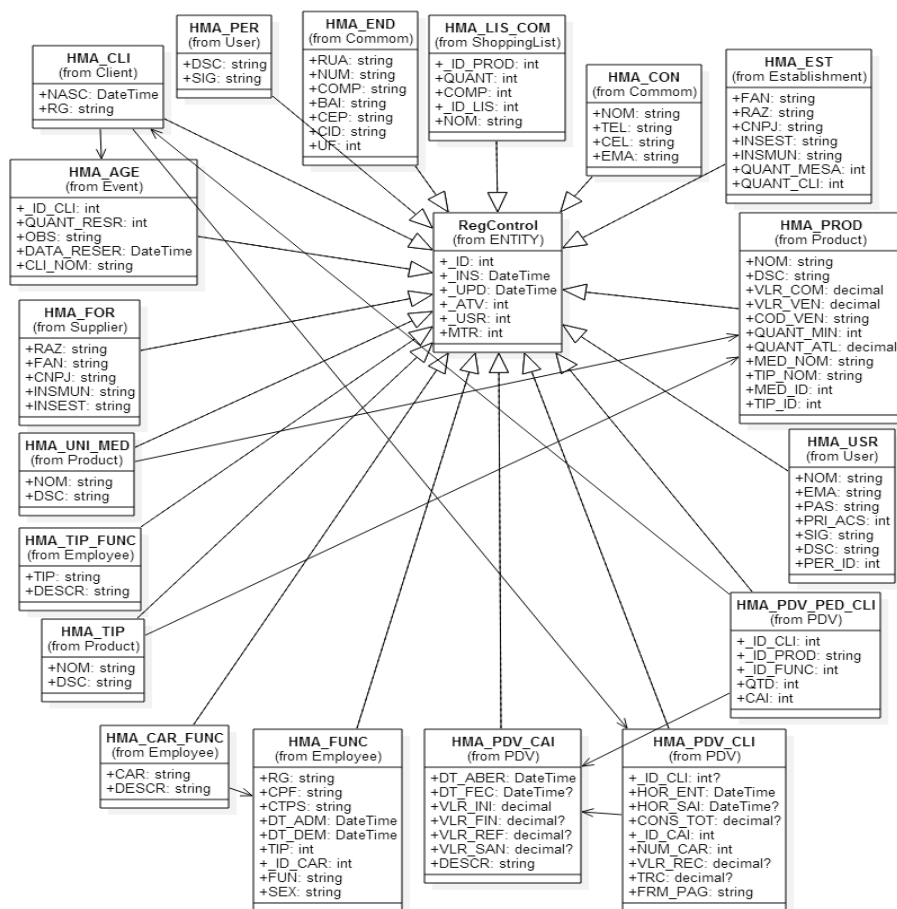
FIGURA 26 – Diagrama de Casos de Uso Significativos



6.4.4 Diagrama de Classes de Domínio

O diagrama de classe fornece uma visão estática e macro do sistema, este diagrama deve conter todas as classes utilizadas para o desenvolvimento e seus atributos com tipos respectivos.

FIGURA 27 – Diagrama de Classe HMA



6.4.5 Realização de casos de uso

Este diagrama é responsável por demonstrar o fluxo de cada caso de uso significativo.

FIGURA 28 – Realização Caso de Uso Abrir Caixa

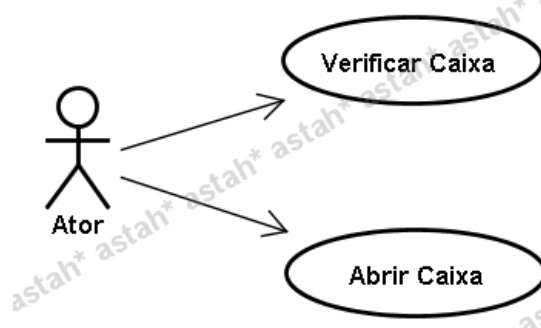


FIGURA 29 – Diagrama de Sequência Abrir Caixa

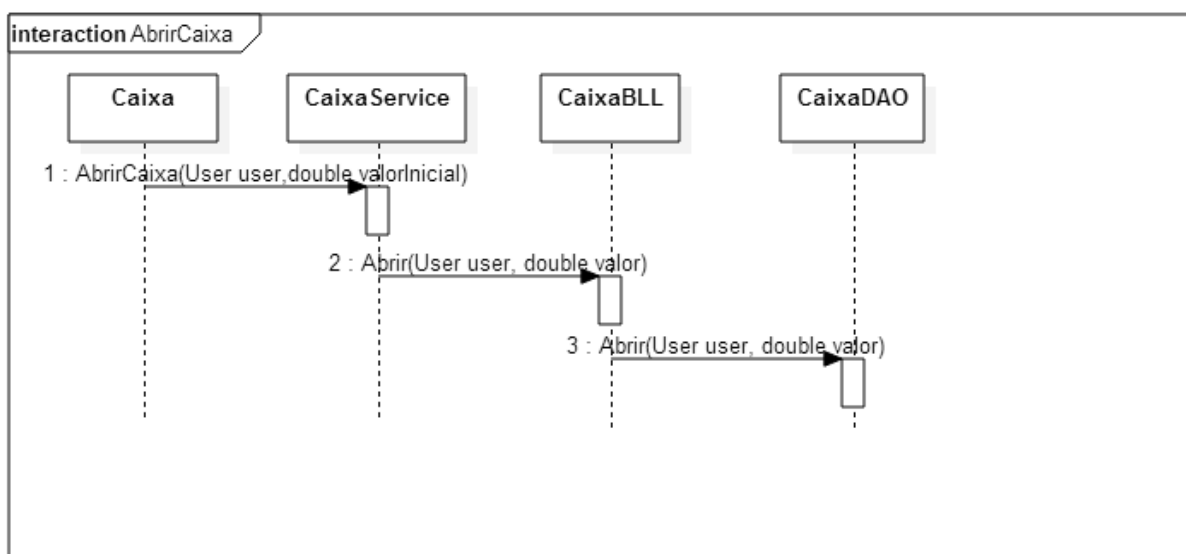


FIGURA 30 – Realização de Caso de Uso – Fechar Caixa

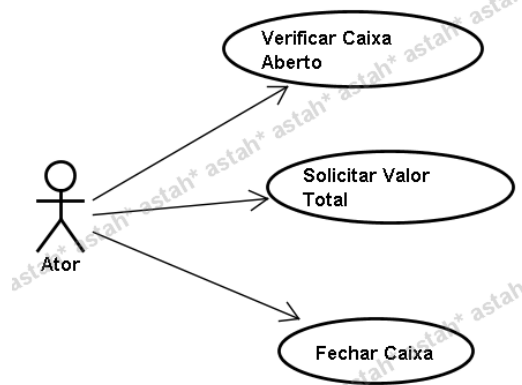


FIGURA 31 – Diagrama de Sequencia – Fechar Caixa

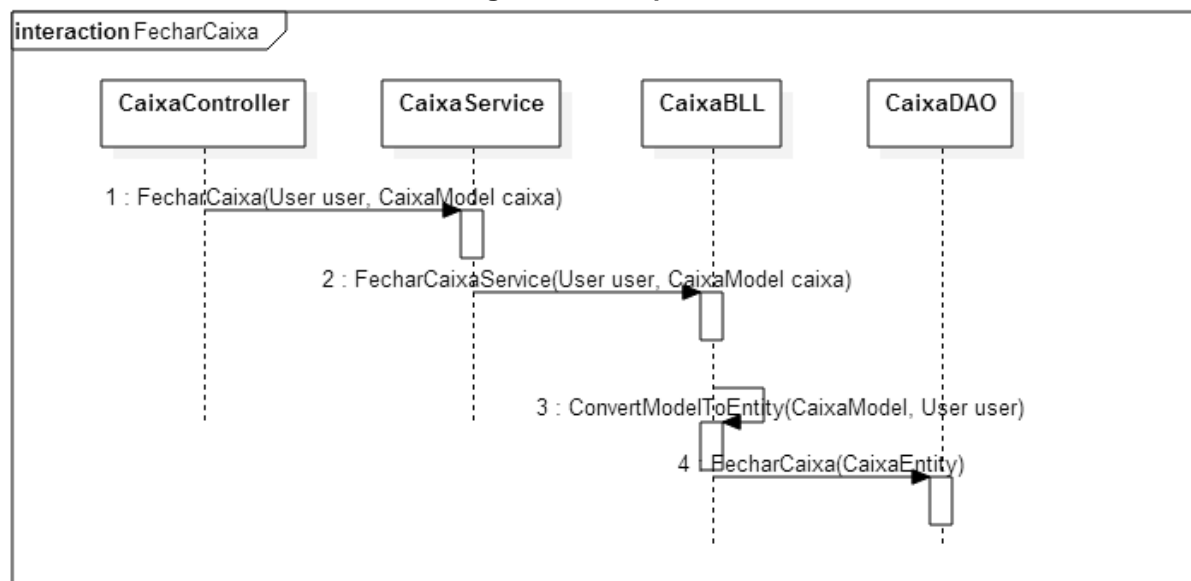


FIGURA 32 – Realização de Caso de Uso – Abrir Comanda

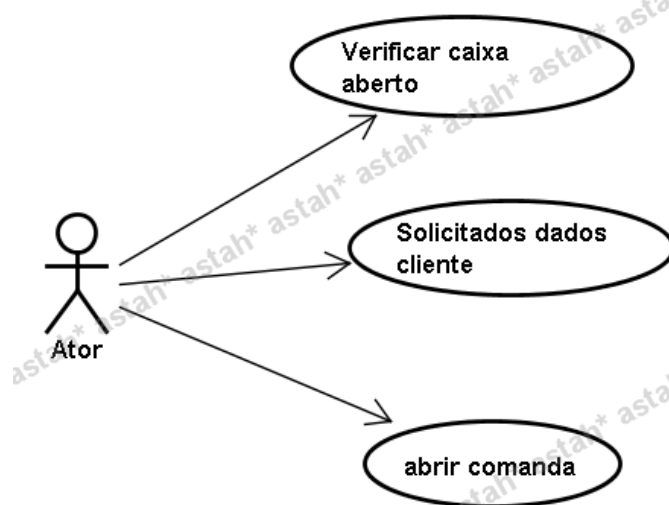


FIGURA 33 – Diagrama de Sequencia – Abrir Comanda

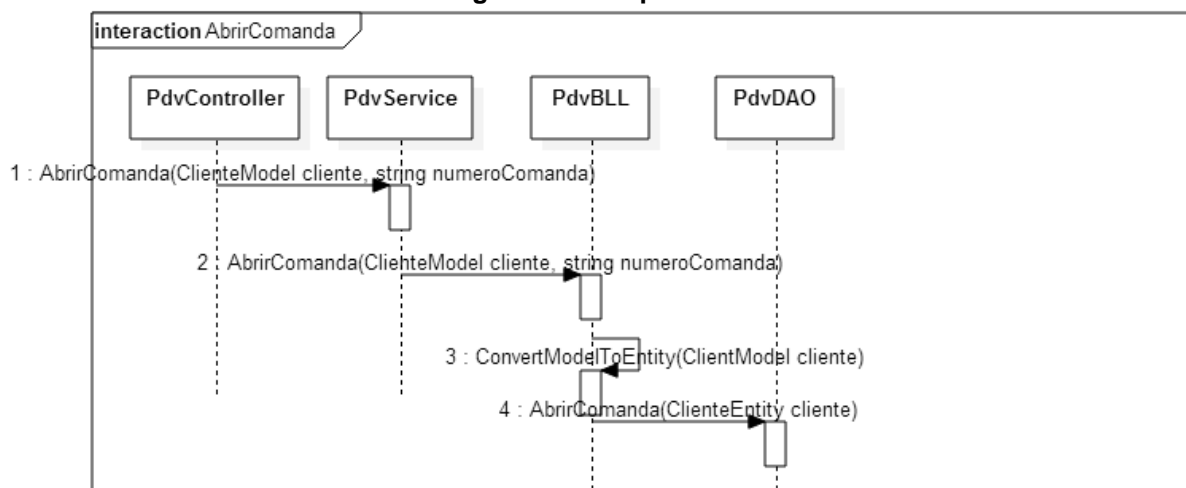


FIGURA 34 – Realização de Caso de Uso – Fechar Comanda

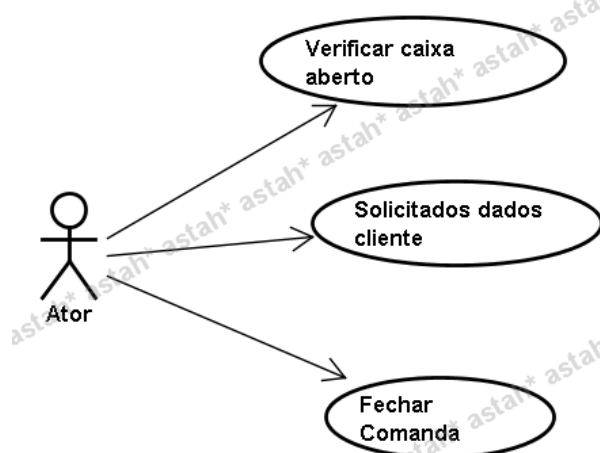
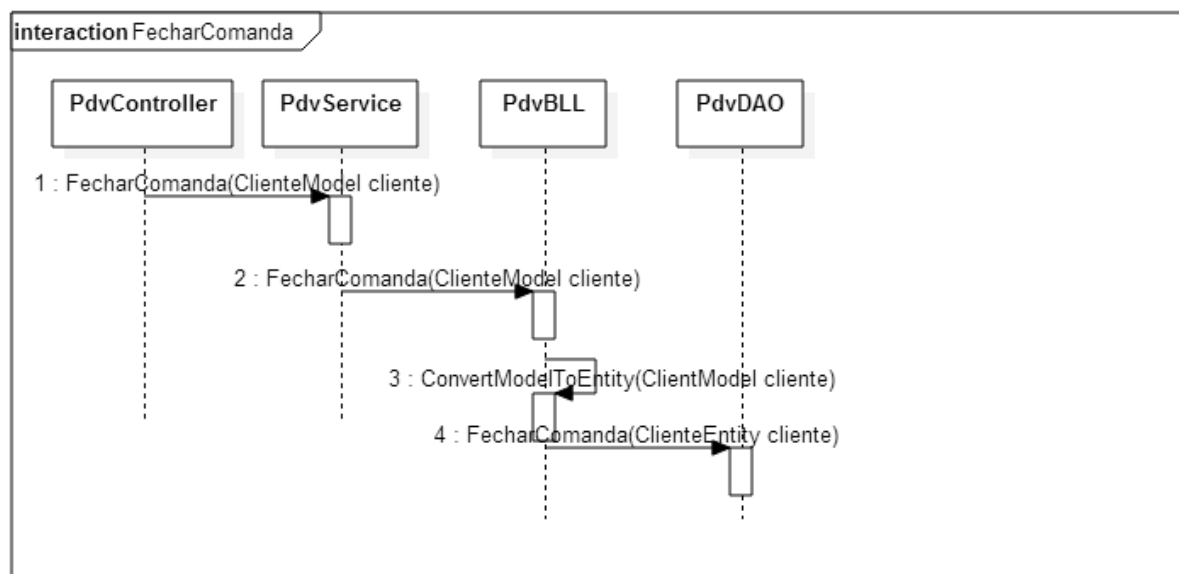
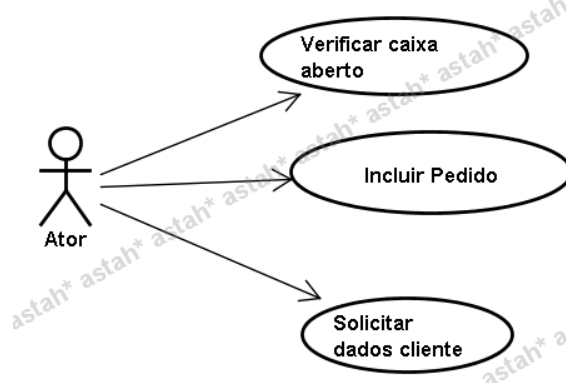
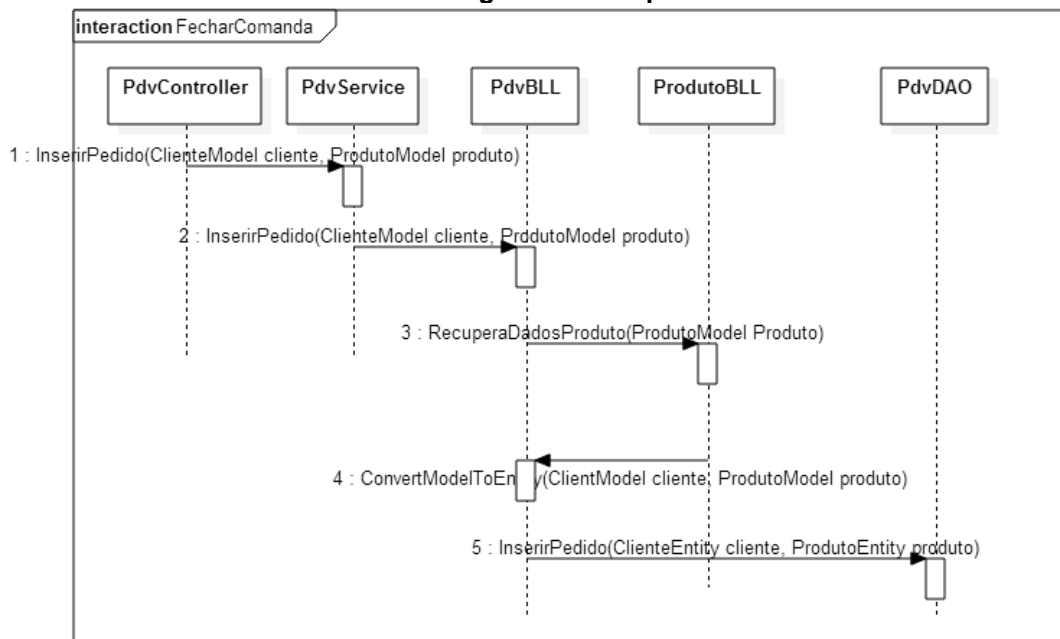


FIGURA 35 – Diagrama de Sequência – Fechar Comanda**FIGURA 36 – Realização de Caso de Uso – Incluir Pedido****FIGURA 37 – Diagrama de Sequência – Incluir Pedido**

6.4.6 Documentação dos casos de uso Significativos

Para a documentação dos Casos de Uso, foi criado um documento que mantém todas as características do caso de uso anexadas em um único arquivo. Protótipo, validações, mensagens, regras de negócios gerais, fluxos alternativos e de exceção estão presentes em um mesmo documento, facilitando a leitura e entendimento, e centralizando as informações necessárias para desenvolvimento.

Caso de Uso Abrir Caixa

Identificação do Produto / Visão Geral / Finalidade

- Tela de abertura de caixa.

Localização no Sistema

- Esta rotina deverá estar disponível em PDV\Caixa\Abrir.
- Também deverá conter um botão de acesso rápido no home do PDV.

Dados para Implementação

O protótipo da tela 'Abrir Caixa' apresenta uma interface com uma barra de título contendo o texto 'Abrir Caixa'. Abaixo da barra, há uma barra de navegação com o link 'Home | Pdv | Abrir Caixa'. O corpo principal da tela contém o rótulo 'Valor Inicial' seguido por um campo de entrada de texto. Na base da interface, há um botão rotulado 'Abrir Caixa'.

- O título da tela deverá ser “HMA – Abrir Caixa”.

Regra de Apresentação dos Atributos

NOME CAMPO	TIPO REGRA	DESCRIÇÃO
Valor Inicial	Ajuda	“Insira o valor inicial do caixa”
	Conteúdo Inicial	Vazio
	Descrição	Valor Inicial
	Disponibilidade	Escrita
	Formato	Numérico
	Tamanho (caracteres)	10
	Máscara	Monetária
	Obrigatoriedade	SIM

Regras de Negócio

- A1 – Botão Abrir Caixa: Deverá efetuar a inserção do valor na base de dados conforme fluxo principal descrito abaixo.

Fluxos

Pré-Condições

- O usuário deverá estar autenticado
- O usuário deverá estar no módulo PDV

Fluxo Principal

1. Ator clica no botão Abrir Caixa
2. O sistema exibe a tela de abertura de caixa
3. O ator informa o valor inicial do caixa
4. O ator confirma a abertura de caixa
5. O sistema exibe mensagem de confirmação para o usuário
6. O ator confirma a mensagem exibida (MSG 0.00.001)
7. O sistema valida a informação inserida (RN 0.00.001)
8. O sistema insere a informação na base de dados
9. O sistema exibe a mensagem de confirmação (MSG 0.00.002)

Fluxo Alternativo

- Não se aplica

Fluxo de Exceção

- Não se aplica

Caso de Uso Fechar Caixa

Identificação do Produto / Visão Geral / Finalidade

- Tela de fechamento de caixa

Localização no Sistema

- Esta rotina deverá estar disponível em PDV\Caixa\Fechar.
- Deverá conter um botão de acesso rápido no home do PDV.

Dados para Implementação

The screenshot shows a web application window titled "Fechar Caixa". At the top, there is a navigation bar with links "Home | Pdv | Fechar Caixa". Below the navigation bar, there is a main content area. In the center of this area, there are two input fields: "Valor Final" (empty) and "Data" (containing "05/10/2015"). At the bottom of the main content area, there is a button labeled "Fechar Caixa".

Regra de Apresentação dos Atributos

NOME CAMPO	TIPO REGRA	DESCRIÇÃO
Valor Final	Ajuda	"Insira o valor total em caixa"
	Conteúdo Inicial	Vazio
	Descrição	Valor Final
	Disponibilidade	Escrita
	Formato	Numérico
	Tamanho (caracteres)	10
	Máscara	Monetária
	Obrigatoriedade	SIM
Data	Ajuda	"Data da solicitação"
	Conteúdo Inicial	Data atual do servidor
	Descrição	Data
	Formato	Data
	Tamanho (caracteres)	10
	Máscara	Data
	Obrigatoriedade	SIM

Regras de Negócio

- A1 – Botão Fechar Caixa: Deverá efetuar a inserção do valor na base de dados conforme fluxo principal descrito abaixo.

Fluxos

Pré-Condições

- O usuário deverá estar autenticado
- O usuário deverá estar no módulo PDV

Fluxo Principal

- 1 Ator clica no botão Fechar Caixa
- 2 O sistema exibe a tela de fechamento de caixa
- 3 O ator informa o valor total em caixa
- 4 O sistema soma todos os valores de entrada no período do caixa
- 5 O sistema retorna à somatória dos valores acima
- 6 O sistema compara a somatória com o valor inserido pelo ator.
- 7 O sistema exibe uma mensagem de confirmação
- 8 O ator confirma a mensagem
- 9 O sistema insere os dados de fechamento na base de dados
- 10 O sistema retorna para o menu principal do PDV.

Fluxo Alternativo

A1 – Este fluxo inicia-se no momento em que o valor inserido pelo ator não coincida com o valor da somatória realizado pelo sistema

1. O sistema exibe uma mensagem de erro de combinação de valores
2. O ator insere novamente o valor para fechamento
3. O sistema verifica se os valores coincidem
4. O sistema retorna para o passo 7 do fluxo principal.

A2 – Este fluxo iniciasse no momento em que o valor inserido pelo ator não coincida com o valor da somatória realizada pelo sistema (A1)

1. O sistema exibe uma mensagem de erro de combinação
2. O sistema envia um e-mail ao usuário responsável pelo sistema
3. O sistema bloqueia a tela para edição
4. O sistema solicita dados de usuário e senha do administrador para desbloquear
5. O ator insere os dados de administrador
6. Inicia-se o fluxo principal

Caso de Uso Entrada de Cliente

Identificação do Produto / Visão Geral / Finalidade

- Tela de entrada de cliente

Localização no Sistema

- Esta rotina deverá estar disponível em PDV\Cliente\Entrada.
- Também deverá conter um botão de acesso rápido no home do PDV.

Dados para Implementação

Entrada Cliente

[Home](#) | [Pdv](#) | [Entrada Cliente](#)

RG

Nome

Tel

DT Nasc

Cartão

Regra de Apresentação dos Atributos

NOME CAMPO	TIPO REGRA	DESCRIÇÃO
RG	Ajuda	"Insira o RG"
	Conteúdo Inicial	Vazio
	Descrição	RG
	Disponibilidade	Escrita
	Formato	Alfanumérico
	Tamanho (caracteres)	20
	Máscara	N/A
	Obrigatoriedade	SIM
Nome	Ajuda	"Insira o nome"
	Conteúdo Inicial	VAZIO
	Descrição	Nome
	Formato	Alfanumérico
	Tamanho (caracteres)	50
	Máscara	N/A
	Obrigatoriedade	SIM
Telefone	Ajuda	"Insira o Telefone"
	Conteúdo Inicial	VAZIO
	Descrição	Telefone

	Formato	Alfanumérico
	Tamanho (caracteres)	12
	Máscara	Telefone
	Obrigatoriedade	SIM
Data de Nascimento	Ajuda	“Insira a data de nascimento”
	Conteúdo Inicial	VAZIO
	Descrição	Data de nascimento
	Formato	Data
	Tamanho (caracteres)	12
	Máscara	Data
	Obrigatoriedade	SIM
Cartão	Ajuda	“Insira o número do próximo cartão”
	Conteúdo Inicial	VAZIO
	Descrição	Número Cartão
	Formato	Numérico
	Tamanho (caracteres)	100
	Máscara	Numérico
	Obrigatoriedade	SIM

Regras de Negócio

- **A1 – Botão Realizar Entrada:** Deverá efetuar a inserção do valor na base de dados conforme fluxo principal descrito abaixo.

Fluxos

Pré-Condições

- O usuário deverá estar autenticado
- O usuário deverá estar no módulo PDV

Fluxo Principal

10. Ator clica no botão Entrada Cliente
11. O sistema exibe a tela de registro de entrada de clientes
12. O usuário insere o número do rg do cliente
13. O sistema verifica se o cliente já possui cadastro
14. O usuário insere o número do cartão disponível para acesso
15. O usuário clica em realizar entrada
16. O sistema insere os dados na base de dados
17. O sistema exibe a mensagem de confirmação
18. O usuário clica na mensagem de confirmação
19. O sistema atualiza a tela de entrada de clientes (limpar todos os registros)

Fluxo Alternativo

A1 – Este fluxo inicia-se no momento na verificação do rg. Caso o cliente não possua cadastro no estabelecimento.

5. O sistema libera os campos (Nome, telefone e data de nascimento)
6. O usuário preenche os campos liberados com as informações fornecidas pelo cliente
7. O sistema efetuar a inserção do cliente no cadastro de clientes (INATIVO)
8. Retorna para o passo 6 do fluxo principal

Fluxo de Exceção

- Não se aplica

Caso de Uso Saída de Cliente

Identificação do Produto / Visão Geral / Finalidade

- Tela de saída do cliente

Localização no Sistema

- Esta rotina deverá estar disponível em PDV\Cliente\Saída.
- Também deverá conter um botão de acesso rápido no home do PDV.

Dados para Implementação

Saída Cliente

[Home](#) | [Pdv](#) | [SaídaCliente](#)

Cartão	<input type="text"/>
Total	<input type="text"/>
Valor Recebido	<input type="text"/>
Troco	<input type="text"/>

Forma Pagamento

▼ Produto	▼ Quantidade	▼ Valor
Água Mineral	1	R\$ 10.00
Entrada Masculina	1	R\$ 20.00
Cerveja	2	R\$ 14.00

Regra de Apresentação dos Atributos

NOME CAMPO	TIPO REGRA	DESCRIÇÃO
Cartão	Ajuda	“Insira o número do cartão”
	Conteúdo Inicial	Vazio
	Descrição	Cartão de entrada
	Disponibilidade	Escrita
	Formato	Numérico
	Tamanho (caracteres)	100
	Máscara	N/A
	Obrigatoriedade	SIM
Total	Ajuda	“Valor total”
	Conteúdo Inicial	VAZIO
	Descrição	Total
	Formato	Numérico Leitura

	Tamanho (caracteres)	10
	Máscara	Monetário
	Obrigatoriedade	N/A
Valor Recebido	Ajuda	“Insira o valor recebido pelo cliente”
	Conteúdo Inicial	VAZIO
	Descrição	Valor Recebido
	Formato	Alfanumérico
	Tamanho (caracteres)	10
	Máscara	Monetário
	Obrigatoriedade	SIM
Troco	Ajuda	“Troco a ser devolvido”
	Conteúdo Inicial	VAZIO
	Descrição	Troco
	Formato	Numérico
	Tamanho (caracteres)	10
	Máscara	Monetário
	Obrigatoriedade	SIM
Forma de Pagamento	Ajuda	“Selecione a forma de pagamento”
	Conteúdo Inicial	VAZIO
	Descrição	Forma de pagamento
	Formato	Checkbox ou Button
	Tamanho (caracteres)	N/A
	Máscara	N/A
	Obrigatoriedade	SIM

Regras de Negócio

- **A1 – Botão Realizar Saída:** Deverá efetuar a inserção do valor na base de dados conforme fluxo principal descrito abaixo.

Fluxos

Pré-Condições

- O usuário deverá estar autenticado
- O usuário deverá estar no módulo PDV

Fluxo Principal

1. Ator clica no botão Saída Cliente
2. O sistema exibe a tela de registro de saída de cliente
3. O ator insere o número do cartão
4. O sistema busca todo o consumo do cliente

5. O sistema exibe o consumo na grid disponível na tela
6. O sistema exibe o valor total de consumo
7. O ator insere o valor recebido pelo cliente
8. O sistema calcula o troco
9. O ator seleciona a forma de pagamento
10. O ator clica em registrar saída
11. O sistema insere os dados na base de dados
12. O sistema libera o cartão para uso

Fluxo Alternativo

- Não se aplica

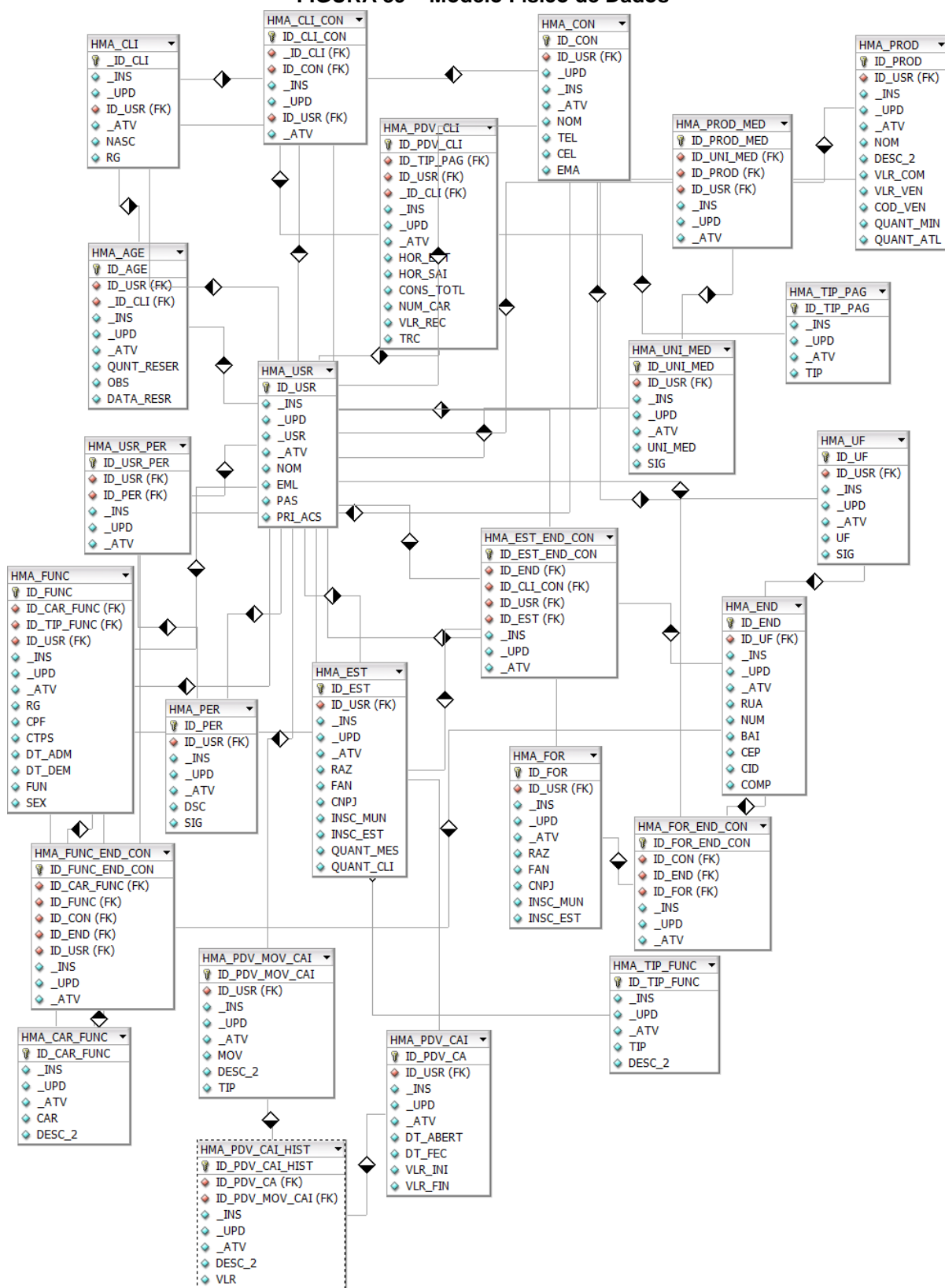
Fluxo de Exceção

- Não se aplica

6.5 Modelo Físico de Dados

Este modelo demonstra como os dados são armazenados no banco de dados. O software utiliza o banco de dados Microsoft Sql Server 2012, por ser um SGBD (Sistema Gerenciador de Banco de Dados) de grande porte, suportando até 2000 requisições simultâneas mesmo em sua versão gratuita, a facilidade na integração com a linguagem C# também foi um ponto a favor da escolha deste banco de dados. Como a aplicação mantém informações de regra de negócio em banco de dados, facilitando assim a manutenção da aplicação, mesmo com o sistema em produção, não sendo necessário compilação e paralização da aplicação.

FIGURA 39 – Modelo Físico de Dados



6.6 Desenvolvimento

O desenvolvimento do código-fonte do HMA, se deu através da criação das camadas. Todo o projeto foi dividido em 6 sub-projetos, visando um desacoplamento total da aplicação. Todos os projetos presentes da *solution*, exceto o projeto WEB e o projeto WCF, foram criados em formato *Class Library*, para que fossem criadas DLL's, separadas para cada projeto. Esta arquitetura foi definida para facilitar e agilizar futuras manutenções ou até mesmo migrações de tecnologias, visto que, por exemplo, para retirar as regras de negócio da aplicação é necessário somente remover a referência do projeto WCF, não afetando nenhuma outra parte da aplicação.

As regras de negócio de importância alta, ou seja, as regras que definem os casos de uso significativos foram criadas em stored procedures localizadas no banco de dados. Essa definição favorece manutenções, não sendo necessário a compilação e publicação da aplicação toda vez que for necessário alterar uma regra de negócio.

6.6.1 DAO

A DAO (Data Access Object), é a camada responsável por criar, encerrar e gerenciar todas as requisições da aplicação ao banco de dados. Esta camada não possui nenhum tipo de especificação de regra de negócio, e trabalha com seus objetos próprios, chamados ENTITYS.

6.6.2 ENTITY

Esta camada é responsável por conter os objetos de banco de dados. Toda solicitação ou envio de informações ao SGBD é realizado através destes objetos, que são cópias das tabelas definidas em banco. Foi utilizado este formato, visando a agilidade na instância de objetos dinamicamente.

6.6.3 BLL

Business Logic Layer ou camada de regra de negócio, é a área responsável por manter as regras de negócio *fixas* da aplicação. Tendo em vista que as principais regras de negócio da aplicação estão localizadas em *stored procedures* no banco de dados, esta camada armazena apenas validações de suporte a regra de negócio principal.

6.6.4 SERVICE

Esta camada é responsável por realizar a comunicação da interface do usuário com as regras de negócio da aplicação. Servindo como um desacoplador da aplicação. Esta camada favorece futuras alterações, podendo ser uma migração de tecnologias, ou inserção de novas camadas de front-end.

6.6.5 MODEL

Este sub-projeto é responsável por conter os objetos de interface. Por ser uma aplicação baseada em Asp .NET MVC, todas as telas possuem um objeto do tipo model para manter um contrato, ou seja, as telas possuem atributos definidos através de uma classe, que são responsáveis pela criação da view do usuário.

6.6.6 WEB

Esta camada é a aplicação em ASP .NET MVC, que tem como propósito principal entregar a interface ao usuário. A definição ou escolha para este formato foi visando a agilidade de uma aplicação em ASP .NET atuando em servidores IIS, a segurança que uma aplicação baseada neste modelo fornece e a agilidade no desenvolvimento.

7 CONSIDERAÇÕES FINAIS

Conhecimento técnico, prazos curtos e custos fixos são algumas das problemáticas a serem gerenciadas para que o projeto ocorra com sucesso. Porém, processo de desenvolvimento de um sistema possui inúmeros outros desafios, é necessário adquirir conhecimento em áreas distintas, entendimento de negócio e até, por que não, uma visão empreendedora, para que possa auxiliar o cliente em seus processos atuais.

Mas apesar das dificuldades possíveis, a dimensão e o aprendizado em realizar um projeto deste porte é imensurável.

Após a realização deste projeto, foi possível analisar os conhecimentos adquiridos. Muitas falhas ocorreram durante este projeto, que serviu como uma grande experiência para projetos futuros.

Um dos pontos mais difíceis para a realização deste projeto, foi o levantamento dos requisitos, onde foi necessário realizar uma abstração técnica e entender como é o funcionamento do cliente piloto, seus processos atuais e a forma de lidar com a gestão. A necessidade de abstração técnica foi um ponto que precisou ser trabalhado para que os levantamentos dos requisitos necessários para o desenvolvimento fossem adquiridos com sucesso.

A comunicação com o cliente piloto foi também um ponto que teve que ser bem gerenciado, como o estabelecimento não funciona em horário comercial padrão, foi necessária uma adequação de horários para que as visitas, levantamentos, trocas de informações e entregas parciais do sistema fossem realizadas durante o horário de funcionamento do bar, normalmente nas madrugadas de sábado para domingo.

Após estes pontos que precisaram de uma atenção em especial, um ponto positivo que merece destaque foi a conduta de nosso professor orientador. Sempre

disposto a ajudar e com um enorme conhecimento e bagagem profissional. Sem dúvida, sem a presença de um profissional deste porte, este projeto não teria ocorrido com sucesso.

O apoio do cliente piloto, que demonstrou confiança e interesse mútuo para o desenvolvimento deste sistema, nos concedeu acesso irrestrito ao estabelecimento, para que pudéssemos analisar e levantar todos os requisitos necessários. A importância desta parte foi grande o suficiente para o estabelecimento alterar seu fluxo atual de processo após a realização do BPMN. Onde foi analisado falhas no atendimento e solicitação de pedidos realizados pelos clientes.

Este projeto teve uma visão diferenciada, não sendo somente para atender a organização cliente, mas sim para o desenvolvimento de um produto comercial. O nosso cliente piloto, entrando como parceiro estratégico, nos forneceu inúmeros conhecimentos da área, fazendo com que obtivéssemos um diferencial para implementar o produto.

Nesta primeira fase, chamada de fase acadêmica, foi entregue os módulos básicos para funcionamento da aplicação em um bar e ou restaurante. Entrada e saída de clientes, abertura e fechamento de caixa, controle de estoque e pedidos são os principais casos de uso para um pdv.

Este projeto já possui uma segunda fase que está em levantamento, onde será desenvolvido módulos fiscais e controle das informações financeiras. Este produto deverá atender as necessidades básicas de todos os bares e restaurantes, mas tendo como premissa a utilização das mais modernas tecnologias, conseguindo assim aplicar agilidade, segurança e confiabilidade aos futuros clientes.

REFERÊNCIAS

CODEPLEX Welcome to Windows Presentation Foundation! Disponível em: <http://wpf.codeplex.com/> Acesso em: 13/03/2015

CODEPLEX WPF Toolkit – January 2009 Release Disponível em: <http://wpf.codeplex.com/releases/view/22567>

Microsoft Sql Server Disponível em: <https://msdn.microsoft.com/pt-br/library/cc564903.aspx> Acesso em: 10/03/2015.

Git Primeiros Passos - Disponível em: <http://git-scm.com/book/pt-br/v1/Primeiros-passos-No%C3%A7%C3%B5es-B%C3%A1sicas-de-Git> Acessado em: 10/03/2015

Kanban Gestão Disponível em: <http://www.knoow.net/cienceconempr/gestao/kanban.htm>
<http://www.significados.com.br/kanban/> Acesso em 10/03/2015

Visual Studio Disponível em: <https://msdn.microsoft.com/pt-br/vstudio/ff431702> Acesso 13/03/2015

Iterativo e Incremental – Disponível em: <http://www.linhadecodigo.com.br/artigo/2108/evolucao-da-metodologia-do-desenvolvimento-de-sistemas.aspx>, Acesso em : 06/10/2015

Gestão de projetos ABNT NBR ISO 10006 Gestão da qualidade – Diretrizes para a qualidade no gerenciamento de Projetos ABNT, Dez 2000

PMI Um Guia do Conjunto de Conhecimentos em Gerenciamento de Projetos Terceira edição (Guia PMBOK),2004.

MSDN Introdução ao Visual C# Disponível em: <http://msdn.microsoft.com/pt-br/vcsharp/dd919145.aspx> Acesso em: 13/03/2015

Mvc Disponível em: <http://www.devmedia.com.br/introducao-ao-padrao-mvc/29308> Acesso em 13/03/2015

Orm Disponível em: <http://www.devmedia.com.br/analise-dos-melhores-orm-object-relational-mapping-para-plataforma-net/5548> Acesso em 13/03/2015

.NET FRAMEWORK Disponível em: [https://msdn.microsoft.com/pt-br/library/hh425099\(v=vs.110\).aspx](https://msdn.microsoft.com/pt-br/library/hh425099(v=vs.110).aspx) Acesso 04/05/2015

JAVAFREE.ORG Design Pattern Disponível em: <http://javafree.uol.com.br/wiki/Design%20Pattern>. Acesso em :04/05/2014

Design Patterns, Marcon Daniel Stefani, Seewald Lucas Adams, Berni Mauricio, Lerina Pablo, Kehl Thiago Nunes, **PDF** páginas 2 e 3.

Bizagi Software. Business Process Management (BPM) Software. Disponível em <http://www.bizagi.com/en/bpm-suite/bpm-products/modeler>. Acesso em: 18 mar. 2015.

GESTÃO DE RESTAURANTES, Pesquisa sobre" automação em restaurantes. Disponível em <https://gestaoderestaurantes.wordpress.com/2008/05/04/pesquisa-sobre-automao-em-restaurantes/>>Acesso em 20 de fevereiro de 2015.

MAGAZINE. **SQL Magazine. Conceitos fundamentais de bancos de dados.** Disponível em: http://www.sqlmagazine.com.br/Colunistas/RicardoRezende/02_ConceitosBD.asp#Korth. Acesso em 24 mar. 2015

SEBRAE. **Sebrae -SP: Comece Certo: Restaurante.** Disponível em: http://www.sebraesp.com.br/arquivos_site/biblioteca/ComeceCerto/Restaurante.pdf . Acesso em: 16 mar. 2015.

APÊNDICE

DOCUMENTO DE VISÃO

Histórico de Versões

Data	Versão	Descrição	Autor	Revisor
16/07/2015	1.0	Primeira versão do documento de visão	Giuliano Costa	

Partes Interessadas

Hermes Bar Curitiba – Bar e Restaurante

Silvana Brainta
Gerente | Proprietária

Letícia Sanches Dutra
Gerente | Proprietária
41 – 9873-0658

Equipe de Desenvolvimento

Giuliano Henrique Costa
Gerente de Projeto | Desenvolvedor
giulianocosta@outlook.com

Allston Wagner Siviero Martins
Professor Orientador

Objetivo

A finalidade deste documento é coletar e definir as necessidades de alto-nível e características do projeto de software **HMA – Hermes Management Assistant** focando nas funcionalidades requeridas pelas partes interessadas e usuários, bem como estes requisitos serão abordados no projeto de software.

A visão do projeto documenta o ambiente geral de processos a ser desenvolvido para o sistema durante o projeto, fornecendo a todos os envolvidos uma descrição compreensível deste e de suas funcionalidades.

Este Documento de Visão documenta apenas as necessidades e funcionalidades do sistema que serão atendidas no projeto de software.

Cenário Atual

Atualmente o estabelecimento não consta com sistema. O controle de caixa é realizado através das somas dos valores preenchidos na comanda, a mesma não possui produtos descritos, somente uma tabela de valores. O controle de estoque não é realizado atualmente, sendo apenas uma contagem visual dos produtos. Os pedidos a cozinha são realizados através de comandas que são coladas em um quadro.

Descrição do Projeto

As funcionalidades aplicadas ao HMA visam aumentar a eficácia e segurança no atendimento ao cliente e proporcionar um maior controle na gestão do estabelecimento.

O projeto tem como foco principal o desenvolvimento de um software que atenda às necessidades do cliente, como: controle de caixa, controle de estoque utilizando inteligência artificial, solicitação de pedidos a cozinha e bar, emissão de relatórios e recibos. O sistema será dividido em módulos, estes descritos abaixo:

- Módulo Caixa
- Módulo Pedidos
- Módulo Estoque
- Módulo Gestão

Cada módulo terá sua responsabilidade bem definida, mas visando a integração de todos.

Envolvimento

Abrangência

O sistema irá abranger todas as áreas funcionais do estabelecimento como: Controle de estoque; Setor administrativo e financeiro; Solicitação de pedidos; Entrada e saída de clientes; Fechamento de caixa; Manutenção de distribuidoras. O HMA não terá atribuições Inter sistemas, ou seja, ele não terá integração com outros sistemas.

Papel das Partes Interessadas

- Gerente (Hermes Bar)

Descrição	Tomada de decisões estratégicas e acompanhamento administrativo do projeto.
Papel no desenvolvimento	Contribuir para o desenvolvimento fornecendo informações sempre que solicitado; Realizar a aprovação dos módulos;
Insumos ao projeto	Acompanhamento do projeto através de relatórios, aprovação dos módulos e homologação do sistema
Representante	Leticia Sanches Dutra

- **Gerente de Projeto**

Descrição	Tomada de decisões estratégicas e acompanhamento do projeto.
Papel no desenvolvimento	Responsabilidade na comunicação com o cliente; Auxiliar a equipe de desenvolvimento em questões de regras de negócio; Inspeção e acompanhamento das tarefas; Auxiliar no direcionamento do projeto; Emissão de relatórios de produtividade
Insumos ao projeto	Gerenciamento do documento de gestão de projeto.

- **Equipe de teste**

Descrição	Testes do sistema e banco de dados
Papel no desenvolvimento	Realizar testes de nível de usuário no sistema e auxiliar a equipe de desenvolvimento
Insumos ao projeto	Emissão de relatórios e controle de bugs

- **Analista de Requisitos**

Descrição	Levantamento dos requisitos necessários para o desenvolvimento do sistema
Papel no desenvolvimento	Efetuar o levantamento dos requisitos do sistema juntamente com o cliente e repassar a equipe de desenvolvimento; Manter atualizado os documentos de requisitos;
Insumos ao projeto	Criação e manutenção do documento de requisitos

- **Equipe de Desenvolvimento**

Descrição	Desenvolvimento do sistema a nível de técnico
Papel no desenvolvimento	Desenvolver o sistema e banco de dados utilizando tecnologias especificadas; Efetuar o levantamento e a decisão da arquitetura do software
Insumos ao projeto	Código fonte do sistema; Diagramas referentes ao desenvolvimento técnico;

Papel dos Atores

- **Administrador do Sistema**

Descrição	Administração geral do sistema
Papel	Administração e manutenção dos módulos do sistema
Insumos ao sistema	Administração geral de todos os módulos do sistema; Criação e manutenção de usuários; Emissão de logs; Controle de configurações gerais
Representante	Leticia Sanches Dutra Silvana Brainta Giuliano Costa

- **Pedidos**

Descrição	Manutenção parcial dos pedidos dos clientes
Papel	Gerir os pedidos dos clientes, tanto para o bar quanto para cozinha
Insumos ao sistema	Gerar pedidos no sistema, consultar pedidos realizados pelo cliente e cancelar pedidos
Representante:	Letícia Sanches Dutra Garçon

NECESSIDADES E FUNCIONALIDADES

CONTROLE DE CAIXA

Fornecer funcionalidades para uma boa gestão e controle do caixa	Crítico

F1.1	Fechamento de Caixa (Diário)
	Ator: Administrador
F1.2	Liberação de comanda (saída)
	Ator: Administrador
F1.3	Cadastro de entrada de clientes
	Ator: Administrador

PEDIDOS

Fornecer aos funcionários e clientes um controle maior na solicitação de pedidos à cozinha e ao bar.		Crítico
F2.1	Emitir pedidos ao bar	
	Ator: Pedidos	
F2.2	Emitir pedidos a cozinha	
	Ator: Pedidos	
F2.3	Visualização de pedidos realizados	
	Ator: Pedidos	

Observação: Todas as funcionalidades acima citadas poderão ser controladas também pelos administradores do sistema

ESTOQUE

Fornecer ao módulo gestão uma forma automatizada de controle de estoque		Crítico
F3.1	Controlar entrada de produtos	
	Ator: Sistema	
F3.2	Controlar saída de produtos	
	Ator: Sistema	
F3.3	Informar ao módulo gestão a baixa de produtos	
	Ator: Sistema	

Observação: Todas as funcionalidades acima citadas poderão ser controladas também pelos administradores do sistema

GESTÃO

Fornecer ao estabelecimento uma central de controle administrativo e financeiro	Crítico
--	----------------

F4.1	Manutenção de funcionários
	Ator: Administrador
F4.2	Emissão de recibo de pagamento dos funcionários
	Ator: Administrador
F4.3	Manutenção de Atrações
	Ator: Administrador
F4.4	Manutenção de fornecedores
	Ator: Administrador
F4.5	Emissão de pedido de compra aos fornecedores
	Ator: Administrador
F4.6	Controle de contas a pagar
	Ator: Administrador
F4.7	Controle de conta a receber
	Ator: Administrador
F4.8	Aviso de vencimentos por e-mail
	Ator: Administrador
F4.9	Importação de NFe
	Ator: Administrador
F4.10	Manutenção de produtos
	Ator: Administrador

Restrições | Premissas

Restrições

- O projeto não terá integração com outros sistemas.
- O projeto somente imprimirá os cupons em impressora não fiscal

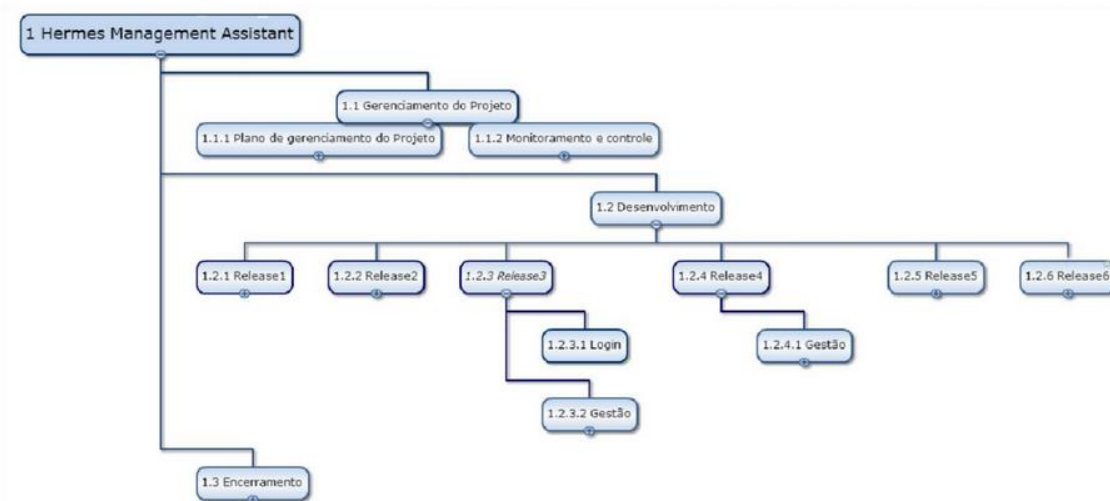
Premissas

- Não existe custo para desenvolvimento e implantação do sistema.
- Os equipamentos necessários (hardwares) serão disponibilizados pelo cliente.
- O sistema será desenvolvido somente para plataforma Microsoft Windows.
- Será necessário uma base de dados atualizada para testes.

Expectativa de Entrega do Produto

Trabalhamos com a expectativa de entregar o produto ao cliente com o término do cronograma definido pelo PAP 2015/2 no dia 12 de dezembro.

EAP Projeto Hermes Bar



ESPECIFICAÇÃO CASO DE USO

Identificação do Produto / Visão Geral / Finalidade

- Criar a tela de cadastro de fornecedor.

Localização no Sistema

- Esta rotina deverá estar disponível em MENU/Gestão/Fornecedor.

Dados para Implementação

[Pesquisa de Fornecedores]

Pesquisa de Fornecedores

Código

Razão Social

▼ Código	▼ Razão Social
8776	Nestlé Brasil Ltda
0685	Ambev

- O título da tela deverá ser “Pesquisa de Fornecedores”.
- Abaixo deverá ter os campos de pesquisa “Código” e “Razão Social”.
- Deverá conter os botões “Pesquisar” e “Novo”.

Regra de Apresentação dos Atributos

NOME CAMPO	TIPO REGRA	DESCRIÇÃO
Razão Social	Ajuda	“Insira o nome da razão social que deseja pesquisar”
	Conteúdo Inicial	Vazio
	Descrição	Nome da razão social
	Disponibilidade	Escrita
	Formato	Alfanumérico
	Tamanho (caracteres)	50
	Máscara	N/A
	Obrigatoriedade	NÃO
Código	Ajuda	“Insira o código que deseja pesquisar”
	Conteúdo Inicial	Vazio
	Descrição	Código da razão social
	Disponibilidade	Escrita
	Formato	Numérico
	Tamanho (caracteres)	100
	Máscara	N/A
	Obrigatoriedade	NÃO

Pesquisar	Ajuda	“Clique para pesquisar uma atração”
	Conteúdo Inicial	N/A
	Descrição	Botão para pesquisar
	Disponibilidade	Leitura/Ação
	Formato	Botão
	Tamanho (caracteres)	N/A
	Máscara	N/A
	Obrigatoriedade	NÃO
Novo	Ajuda	“Clique para cadastrar uma nova atração”
	Conteúdo Inicial	N/A
	Descrição	Botão de acesso ao sistema
	Disponibilidade	Leitura/Ação
	Formato	Botão
	Tamanho (caracteres)	N/A
	Máscara	N/A
	Obrigatoriedade	NÃO

RESULTADO DA CONSULTA

Nome Campo	Tipo/Tamanho	Regras Associadas
Código	Numérico	Campo obrigatório
Razão Social	Alfanumérico	Campo obrigatório
Cnpj	Numérico	Campo obrigatório

Regras de Negócio

- **F1 – Botão Novo:** Deverá chamar a tela de Cadastro de Fornecedor.
- **F2 – Botão Pesquisar:** Deverá exibir as informações de acordo com os filtros preenchidos. Se não houver filtros preenchidos, o sistema irá exibir todos os cadastros.
- **F3 – Alteração:** Para alterar o cadastro, realizar um duplo clique no grid da alteração desejada

Fluxos

Pré-Condições

- O usuário deverá estar no módulo de fornecedor
- O usuário deverá ter perfil de Administrador e/ou Gerente

Fluxo Principal

- O usuário preenche os filtros de busca
- O usuário clica em Pesquisar
- O sistema efetua a pesquisa com base nos filtros
- O sistema retorna as informações e apresenta na grid

Fluxo Alternativo [Cadastrar]

- O usuário clica no botão Novo
- O sistema inicia o [F4 – Cadastrar Fornecedor](#)

Fluxo Alternativo [Editar]

- O usuário realiza duplo clique em uma atração exibida no grid
- O sistema inicia o [F3 –Alteração](#)

Fluxo Alternativo [Excluir]

- O usuário realiza duplo clique em uma atração exibida no grid
- O sistema inicia o caso de uso referente [F8 – Botão Excluir](#)

Fluxo de Exceção

- Não se aplica

[Cadastro de Fornecedores]

Esta tela deverá ser dividida em abas, “Dados Gerais”, “Endereço” e “Contato”.

A aba **Dados Gerais** deverá conter os campos:

Regra de Apresentação dos Atributos

NOME CAMPO	TIPO REGRA	DESCRIÇÃO
Razão Social	Ajuda	“Insira o nome da razão social do fornecedor”
	Conteúdo Inicial	Vazio
	Descrição	Nome da razão social
	Disponibilidade	Escrita
	Formato	Alfanumérico
	Tamanho (caracteres)	50
	Máscara	N/A
	Obrigatoriedade	SIM
Cnpj	Ajuda	“Insira o cnpj do fornecedor”
	Conteúdo Inicial	Vazio
	Descrição	Número do cnpj
	Disponibilidade	Escrita
	Formato	Numérico
	Tamanho (caracteres)	14

Insc. Estadual	Máscara	00.000.000/0000-00
	Obrigatoriedade	NÃO
	Ajuda	“Insira a inscrição estadual do fornecedor”
	Conteúdo Inicial	Vazio
	Descrição	Inscrição estadual do fornecedor
	Disponibilidade	Escrita
	Formato	Alfanumérico
	Tamanho (caracteres)	20
	Máscara	N/A
	Obrigatoriedade	NÃO

Regras de Negócio

- **F5 – Campos Obrigatórios:** Não permitir salvar caso campos obrigatórios estejam em branco. Exibir mensagem:
 - [Mensagem – “*Campos obrigatórios não preenchidos*” - nome dos campos não preenchidos -]

Fluxos

Pré-Condições

- O usuário deverá ter perfil de Administrador e/ou Gerente
- O usuário deverá estar no módulo de fornecedor

Fluxo Principal

- O usuário preenche os campos
- O usuário clica na aba [endereço](#) para continuar o cadastro

Fluxo Alternativo

Não se aplica

Fluxo de Exceção

Não se aplica

MANTER USUÁRIO

Identificação do Produto / Visão Geral / Finalidade

Criar a tela de cadastro de funcionários.

Localização no Sistema

Esta rotina deverá estar disponível em MENU/Gestão/Funcionários.

Dados para Implementação

[Pesquisa de Funcionários]

▼ Código	▼ Nome	▼ Cpf	▼ Rg	▼ Data de Admissão
1	Ana Paula Queiroz	987.678.380-08	21.086.858-9	08/07/2010 12:00:00 AM
2	Giuliano Costa	567.986.409-09	21.086.858-9	08/07/2010 12:00:00 AM
3	Phillip Silva	987.678.380-08	21.086.858-9	08/07/2010 12:00:00 AM
4	Franciele Antqueves	987.678.380-08	21.086.858-9	08/07/2010 12:00:00 AM

- O título da tela deverá ser Cadastro de Funcionários.
- Abaixo deverá ter um “Filtro de pesquisa”.
- Deverá conter os botões “Pesquisar” e “Novo”.

Regra de Apresentação dos Atributos

NOME CAMPO	TIPO REGRA	DESCRIÇÃO
Código	Ajuda	“Insira o código que deseja pesquisar”
	Conteúdo Inicial	Vazio

	Descrição	Código do funcionário
	Disponibilidade	Escrita
	Formato	Numérico
	Tamanho (caracteres)	10
	Máscara	N/A
	Obrigatoriedade	NÃO
Nome	Ajuda	"Insira o nome que deseja pesquisar"
	Conteúdo Inicial	Vazio
	Descrição	Código da razão social
	Disponibilidade	Escrita
	Formato	Texto
	Tamanho (caracteres)	40
	Máscara	N/A
	Obrigatoriedade	NÃO
Pesquisar	Ajuda	"Clique para pesquisar um funcionário"
	Conteúdo Inicial	N/A
	Descrição	Botão para pesquisar
	Disponibilidade	Leitura/Ação
	Formato	Botão
	Tamanho (caracteres)	N/A
	Máscara	N/A
	Obrigatoriedade	NÃO
Novo	Ajuda	"Clique para cadastrar um novo funcionário"
	Conteúdo Inicial	N/A
	Descrição	Botão de acesso ao sistema
	Disponibilidade	Leitura/Ação
	Formato	Botão
	Tamanho (caracteres)	N/A
	Máscara	N/A
	Obrigatoriedade	NÃO

RESULTADO DA CONSULTA

Nome Campo	Tipo/Tamanho	Regras Associadas
Código	Numérico	Campo obrigatório
Nome	Texto	Campo obrigatório
Cpf	Numérico	Campo obrigatório
Rg	Numérico	Campo obrigatório
Data de Admissão	Numérico	Campo obrigatório

Regras de Negócio

- **Fn1 – Botão Novo:** Deverá chamar a tela de Cadastro de Funcionários.
- **Fn2 – Botão Pesquisar:** Deverá exibir as informações de acordo com os filtros preenchidos. Se não houver filtros preenchidos, o sistema irá exibir todos os cadastros.
- **Fn3 – Alteração:** Para alterar o cadastro, realizar um duplo clique no grid da alteração desejada

Fluxos

Pré-Condições

- O usuário deverá estar no módulo de funcionários
- O usuário deverá ter perfil de Administrador e/ou Gerente

Fluxo Principal

- O usuário preenche os filtros de busca
- O usuário clica em Pesquisar
- O sistema efetua a pesquisa com base nos filtros
- O sistema retorna as informações e apresenta na grid

Fluxo Alternativo [Cadastrar]

- O usuário clica no botão Novo
- O sistema inicia o [Fn4 – Cadastrar Funcionário](#)

Fluxo Alternativo [Editar]

- O usuário realiza duplo clique em uma atração exibida no grid
- O sistema inicia o [Fn3 –Alteração](#)

Fluxo Alternativo [Excluir]

- O usuário realiza duplo clique em uma atração exibida no grid
- O sistema inicia o caso de uso referente [Fn9 – Botão Excluir](#)

Fluxo de Exceção

- Não se aplica

[Cadastro de Funcionários]

A interface de usuário para o Cadastro de Funcionários é apresentada em uma janela com o título "Cadastro de Funcionários". No topo, há uma barra de navegação com três abas: "Geral", "Endereço" e "Contato". A aba "Geral" está atualmente selecionada. Abaixo das abas, há um formulário com um campo de texto rotulado "Nome".

- Esta tela deverá ser dividida em abas, “Geral”, “Endereço” e “Contato”.

A aba **Geral** deverá conter os campos:

Regra de Apresentação dos Atributos

NOME CAMPO	TIPO REGRA	DESCRIÇÃO
Nome	Ajuda	“Insira o nome do funcionário”
	Conteúdo Inicial	Vazio
	Descrição	Nome do funcionário
	Disponibilidade	Escrita
	Formato	Texto
	Tamanho (caracteres)	40
	Máscara	N/A
	Obrigatoriedade	SIM
Cpf	Ajuda	“Insira o cpf do funcionário”
	Conteúdo Inicial	Vazio
	Descrição	Número do cpf do funcionário
	Disponibilidade	Escrita
	Formato	Numérico
	Tamanho (caracteres)	11
	Máscara	00.000.000-00
	Obrigatoriedade	SIM

Rg	Ajuda	“Insira o rg do funcionário”
	Conteúdo Inicial	Vazio
	Descrição	Número do rg do funcionário
	Disponibilidade	Escrita
	Formato	Numérico
	Tamanho (caracteres)	15
	Máscara	N/A
	Obrigatoriedade	NÃO
Cart. Trabalho	Ajuda	“Insira o número da cart. de trabalho do funcionário”
	Conteúdo Inicial	Vazio
	Descrição	Número do da cart. de trabalho do funcionário
	Disponibilidade	Escrita
	Formato	Numérico
	Tamanho (caracteres)	20
	Máscara	N/A
	Obrigatoriedade	NÃO
Série	Ajuda	“Insira a série da cart. de trabalho do funcionário”
	Conteúdo Inicial	Vazio
	Descrição	Número de série da cart. de trabalho do funcionário
	Disponibilidade	Escrita
	Formato	Numérico
	Tamanho (caracteres)	20
	Máscara	N/A
	Obrigatoriedade	NÃO
Tipo	Ajuda	“Selecione o tipo de função do funcionário”
	Conteúdo Inicial	Vazio
	Descrição	Tipo de função
	Disponibilidade	Leitura
	Formato	ComboBox
	Tamanho (caracteres)	N/A
	Máscara	N/A
	Obrigatoriedade	SIM
Rg	Ajuda	“Insira o rg do funcionário”
	Conteúdo Inicial	Vazio
	Descrição	Número do rg do funcionário
	Disponibilidade	Escrita

	Formato	Numérico
	Tamanho (caracteres)	15
	Máscara	N/A
	Obrigatoriedade	NÃO
Data Nasc.	Ajuda	“Selecione uma data”
	Conteúdo Inicial	Vazio
	Descrição	Data de nasc. do funcionário
	Disponibilidade	Leitura
	Formato	Numérico
	Tamanho (caracteres)	N/A
	Máscara	N/A
	Obrigatoriedade	SIM
Data Admissão	Ajuda	“Selecione uma data”
	Conteúdo Inicial	Vazio
	Descrição	Data de admissão do funcionário
	Disponibilidade	Leitura
	Formato	Numérico
	Tamanho (caracteres)	N/A
	Máscara	N/A
	Obrigatoriedade	SIM
Salvar	Ajuda	“Clique para salvar o cadastro”
	Conteúdo Inicial	Vazio
	Descrição	Botão para salvar o novo cadastro de funcionário
	Disponibilidade	Leitura/Ação
	Formato	Botão
	Tamanho (caracteres)	N/A
	Máscara	N/A
	Obrigatoriedade	NÃO

Regras de Negócio

- Conforme regra [Fn5 – Botão Salvar](#), [Fn8 – Campos Obrigatórios](#) e [Fn5 – Validação](#)

Fluxos

Pré-Condições

- O usuário deverá ter perfil de Administrador e/ou Gerente
- O usuário deve estar no módulo de funcionários

Fluxo Principal

- O usuário preenche os campos
- O usuário clica no botão salvar
- O sistema efetua a validação dos campos obrigatórios
- O sistema exibe a mensagem na tela [Fn5 – Botão Salvar](#)
- O usuário clica na aba [endereço](#) para continuar o cadastro

Fluxo Alternativo

Não se aplica

Fluxo de Exceção

- O usuário não preenche todos os campos obrigatórios
- O sistema verifica quais campos não foram preenchidos
- O sistema apresenta na tela a mensagem [Fn8 – Campos obrigatórios](#)

A aba **Endereço** deverá conter os campos:

A imagem mostra uma janela de software intitulada "Cadastro de Funcionário". No topo, há três abas: "Dados Gerais", "Endereço" (destacada) e "Contato". A aba "Endereço" contém os seguintes campos de entrada: "Cep", "Rua", "Número", "Bairro", "Complemento" e "Cidade". Cada campo possui um botão "Consultar" ao lado. Além disso, há um campo "UF" com uma seta para baixo. Um botão "Salvar" está localizado no canto inferior esquerdo da aba.

Regra de Apresentação dos Atributos

NOME CAMPO	TIPO REGRA	DESCRIÇÃO
Cep	Ajuda	"Insira o cep do funcionário"
	Conteúdo Inicial	Vazio
	Descrição	Cep do funcionário

	Disponibilidade	Escrita
	Formato	Numérico
	Tamanho (caracteres)	8
	Máscara	N/A
	Obrigatoriedade	SIM
Consultar	Ajuda	“Clique para pesquisar o endereço pelo cep”
	Conteúdo Inicial	N/A
	Descrição	Botão para pesquisar o endereço pelo Cep
	Disponibilidade	Leitura/Ação
	Formato	Botão
	Tamanho (caracteres)	N/A
	Máscara	N/A
	Obrigatoriedade	NÃO
Rua	Ajuda	“Insira a rua do funcionário”
	Conteúdo Inicial	Vazio
	Descrição	Número da rua do funcionário
	Disponibilidade	Leitura/Escrita
	Formato	Escrita
	Tamanho (caracteres)	100
	Máscara	N/A
	Obrigatoriedade	NÃO
Número	Ajuda	“Insira o número do funcionário”
	Conteúdo Inicial	Vazio
	Descrição	Número do funcionário
	Disponibilidade	Escrita
	Formato	Numérico
	Tamanho (caracteres)	20
	Máscara	N/A
	Obrigatoriedade	NÃO
Complemento	Ajuda	“Insira o complemento do funcionário”
	Conteúdo Inicial	Vazio
	Descrição	Complemento do endereço do funcionário
	Disponibilidade	Escrita
	Formato	Alfanumérico
	Tamanho (caracteres)	30
	Máscara	N/A
	Obrigatoriedade	NÃO

Bairro	Ajuda	“Insira o bairro do funcionário”
	Conteúdo Inicial	Vazio
	Descrição	Bairro do funcionário
	Disponibilidade	Escrita/Leitura
	Formato	Escrita
	Tamanho (caracteres)	40
	Máscara	N/A
	Obrigatoriedade	NÃO
Cidade	Ajuda	“Insira a cidade do funcionário”
	Conteúdo Inicial	Vazio
	Descrição	Cidade brasileira do funcionário
	Disponibilidade	Leitura/Escrita
	Formato	Escrita
	Tamanho (caracteres)	40
	Máscara	N/A
	Obrigatoriedade	NÃO
UF	Ajuda	“Selecione o estado brasileiro do funcionário”
	Conteúdo Inicial	Vazio
	Descrição	Estado brasileiro do funcionário
	Disponibilidade	Leitura
	Formato	Combobox
	Tamanho (caracteres)	N/A
	Máscara	N/A
	Obrigatoriedade	NÃO
Salvar	Ajuda	“Clique para salvar os dados do funcionário”
	Conteúdo Inicial	N/A
	Descrição	Botão para salvar o fornecedor
	Disponibilidade	Leitura/Ação
	Formato	Botão
	Tamanho (caracteres)	N/A
	Máscara	N/A
	Obrigatoriedade	NÃO

Regras de Negócio

- **Fn6 – Botão Consultar Cep:** deverá efetuar a busca pelo CEP nos Correios, validar o resultado e preencher os campos na tela.
 - Caso o usuário tenha conexão ativa com a internet os campos: Rua, Bairro, Cidade e Estado deverão vir bloqueados no cadastro.

- Caso o usuário não tenha conexão ativa com a internet todos os campos deverão vir desbloqueados e o botão Pesquisa Cep deverá ser escondido.
- Caso o botão Consultar seja pressionado sem que o campo CEP esteja preenchido. Exibir mensagem:
[Mensagem – “*Insira um Cep valido!*”]
- **Fn7 – Botão Salvar:** Quando acionado, deverá realizar a validação conforme [Fn5 – Validação](#) e logo após salvar o cadastro. Exibir mensagem:
 - [Mensagem – “*Funcionário cadastrado com sucesso*” -]
- **Fn8 – Campos Obrigatórios:** Não permitir salvar caso campos obrigatórios estejam em branco. Exibir mensagem:
 - [Mensagem – “*Campos obrigatórios não preenchidos*” - nome dos campos não preenchidos -]

Fluxos

Pré-Condições

- O usuário deverá ter perfil de Administrador e/ou Gerente
- O usuário deverá estar no módulo de funcionário

Fluxo Principal

- O usuário preenche os campos
- O usuário clica no botão salvar
- O sistema efetua a validação dos campos obrigatórios
- O sistema exibe a mensagem na tela [Fn5 – Botão Salvar](#)
- O usuário clica na aba [contato](#) para finalizar o cadastro

Fluxo Alternativo

Não se aplica

Fluxo de Exceção

- O usuário não preenche todos os campos obrigatórios
- O sistema verifica quais campos não foram preenchidos
- O sistema apresenta na tela a mensagem [Fn8 – Campos Obrigatórios](#)

A aba **Contato** deverá conter os campos:

The image shows a web browser window with the title 'Cadastro de Funcionário'. Inside the window, there are three tabs: 'Dados Gerais', 'Endereço', and 'Contato'. The 'Contato' tab is selected. Below the tabs, there are three input fields labeled 'Telefone', 'Celular', and 'Email'. At the bottom left of the form area, there is a button labeled 'Salvar'.

Regras de Negócio

- **Fn5 – Validação Idade:** Enviar mensagem de alerta caso o funcionário seja menor de 18 anos. Exibir a mensagem:
[Mensagem – “*Funcionário com menos de 18 anos!*
Continuar? Sim, Não”]

Fluxos

Pré-Condições

- O usuário deverá ter perfil de Administrador e/ou Gerente
- O usuário deverá estar no módulo de funcionário

Fluxo Principal

- O usuário preenche os campos
- O usuário clica no botão salvar
- O sistema efetua a validação dos campos obrigatórios
- O sistema exibe a mensagem na tela [Fn7 – Botão Salvar](#)

Fluxo Alternativo

Não se aplica

Fluxo de Exceção [Campos obrigatórios]

- O usuário não preenche todos os campos obrigatórios
- O sistema verifica quais campos não foram preenchidos
- O sistema apresenta na tela a mensagem [Fn8 – Campos Obrigatórios](#)

Fluxo de Exceção [Menor idade]

- O usuário insere a idade do funcionário
- O sistema valida as informações
- O sistema apresenta na tela a mensagem [Fn5 – Validação Idade](#)

[DESENV] – Edição de Funcionários

Dados para Implementação

[Edição de Funcionários]

A interface de usuário para a edição de funcionários, intitulada "Cadastro de Funcionário". No topo, há uma barra de navegação com três abas: "Dados Gerais", "Endereço" e "Contato", sendo a última selecionada. Abaixo das abas, há três campos de entrada de texto rotulados "Telefone", "Celular" e "Email". Na base da interface, há dois botões: "Salvar" (cinza) e "Excluir" (vermelho).

Regras de Negócio

- **Fn9 – Botão Excluir:** Excluir o cadastro. Exibir a mensagem:
 - [Mensagem – “Confirma exclusão da atração? Sim, Não”].

Exibir mensagem:

- [Mensagem – “*Atração excluída com Sucesso*” -]
- **A8 – Edição:** Quando acionado o botão Salvar em uma edição. Exibir mensagem:
 - [Mensagem – “*Atração Editada com Sucesso*” -]

Fluxos

Pré-Condições

- O usuário deverá ter permissão de Administrador e/ou Gerente
- O usuário deverá estar no modelo de funcionário

Fluxo Principal

- O usuário clica no botão Excluir
- O sistema solicita a confirmação da exclusão
- O usuário confirma a exclusão
- O sistema efetua a exclusão da atração
- O sistema efetua a exclusão do funcionário
- O sistema exibe a mensagem de confirmação – [Fn9 – Botão Excluir](#)
- O usuário confirma a mensagem
- O sistema retorna para a tela de [pesquisa funcionários](#)

Fluxo Alternativo [Excluir]

- O usuário não confirma a exclusão
- O sistema interrompe o processo
- O sistema retorna para [tela de exclusão](#)

Fluxo de Exceção

Não se aplica

AUTENTICAR

Identificação do Produto / Visão Geral / Finalidade

Criar a tela de login do sistema.

Localização no Sistema

Esta rotina deverá estar disponível quando a aplicação for executada.

Dados para Implementação

[Tela de Login]

The image shows a mockup of a login window. The window has a title bar with the text 'Login' and standard window control buttons (minimize, maximize, close). Inside the window, there are two text input fields. The first field is labeled 'Login' and the second is labeled 'Senha'. Below the 'Senha' field, there is a blue hyperlink that says 'Esqueceu sua senha?'. At the bottom of the window, there is a button labeled 'Entrar'.

- O título da tela deverá ser “HMA – Login”.
- Abaixo deverá ter os campos “Login” e “Senha”.
- Deverá conter o link “Esqueceu sua senha?” para recuperar a mesma.
- Deverá conter o botão “Enter” para acessar o sistema.

Regra de Apresentação dos Atributos

NOME CAMPO	TIPO REGRA	DESCRIÇÃO
Login	Ajuda	“Informe seu login de acesso”
	Conteúdo Inicial	Vazio
	Descrição	Login do usuário
	Disponibilidade	Escrita
	Formato	Alfanumérico
	Tamanho (caracteres)	50
	Máscara	N/A
	Obrigatoriedade	SIM

Senha	Ajuda	“Informe sua senha de acesso”
	Conteúdo Inicial	Vazio
	Descrição	Senha do usuário
	Disponibilidade	Escrita
	Formato	Alfanumérico
	Tamanho (caracteres)	
	Máscara	Password
	Obrigatoriedade	NÃO
Esqueceu sua Senha?	Ajuda	“Uma nova senha será enviada a seu e-mail”
	Conteúdo Inicial	N/A
	Descrição	Utilitário de recuperação de senha do usuário
	Disponibilidade	Leitura/Ação
	Formato	Link
	Tamanho (caracteres)	N/A
	Máscara	N/A
	Obrigatoriedade	NÃO
Entrar	Ajuda	“Clique para acessar o sistema”
	Conteúdo Inicial	N/A
	Descrição	Botão de acesso ao sistema
	Disponibilidade	Leitura/Ação
	Formato	Botão
	Tamanho (caracteres)	N/A
	Máscara	N/A
	Obrigatoriedade	NÃO

Regras de Negócio

- **L1 – Botão Entrar:** Quando acionado o botão Entrar, deverá validar o login e a senha e caso as informações estejam corretas, acessar o sistema.
- **L2 - Link Esqueceu Sua Senha:** Ao clicar no link, o sistema deverá gerar uma nova senha, criptografar e gravar no banco de dados, enviar a senha para o e-mail do usuário.

Fluxos

Pré-Condições

- O usuário deverá estar devidamente cadastrado no sistema
- Deverá possuir conexão ativa com a internet

Fluxo Principal

- O usuário inicia o sistema
- O usuário insere o Login
- O usuário insere a Senha
- O usuário clica em Entrar
- O sistema carrega a sessão com as informações do usuário
- O sistema valida as permissões do usuário

Fluxo Alternativo

- O usuário clica no Link Esqueceu Sua Senha
- O sistema busca o e-mail do usuário
- O sistema gera uma nova senha
- O sistema criptografa a nova senha gerada
- O sistema salva no banco a nova senha
- O sistema envia pro e-mail do usuário a nova senha

Fluxo de Exceção

- O usuário clica em Entrar sem preencher nenhum campo obrigatório
- O sistema exibe na tela a mensagem (*'Login e/ou E-mail inválido'*)

Fluxo de Exceção

- O usuário clica no link esqueceu sua senha sem preencher o campo Login
- O sistema exibe na tela a mensagem (*'Obrigatório o preenchimento do campo Login'*)

Fluxo de Exceção

- O usuário preenche com informações incorretas
- O sistema exibe na tela a mensagem (*'Login e/ou E-mail inválido'*)

MANTER TIPO PRODUTO

Identificação do Produto / Visão Geral / Finalidade

Criar a tela de cadastro de produtos.

Localização no Sistema

Esta rotina deverá estar disponível em MENU/Gestão/Produtos/Produtos.

Dados para Implementação

[Pesquisa de Produtos]

Pesquisa de Produtos

Código

Nome

Tipo de Produto

Pesquisar

Novo

▼ Cód.	▼ Nome	▼ Marca	▼ Uni.	▼ Quant. Estoque	▼ Vlr. Venda	▼ Tipo
0001	Coca-Cola Lata 350ML	Coca-Colca	Lata	120	3.00	Refrigerante 350ML
0002	Skol Garrafa 600ML	Skol	Garrafa	120	4.00	Cerveja 600ML
003	Água Mineral 200ML	Maseratti	Garrafa	150	1.50	Água 200ML

- O título da tela deverá ser “Pesquisa de Produtos”.
- Abaixo deverá ter o painel “Filtro de pesquisa”.
-

Filtros de Pesquisa

- **Código:** Campo com tamanho máximo de 10 caracteres.
- **Nome:** Campo com tamanho máximo de 100 caracteres
- **Tipo de Produto:** ComboBox contendo os tipos de produtos cadastrados.

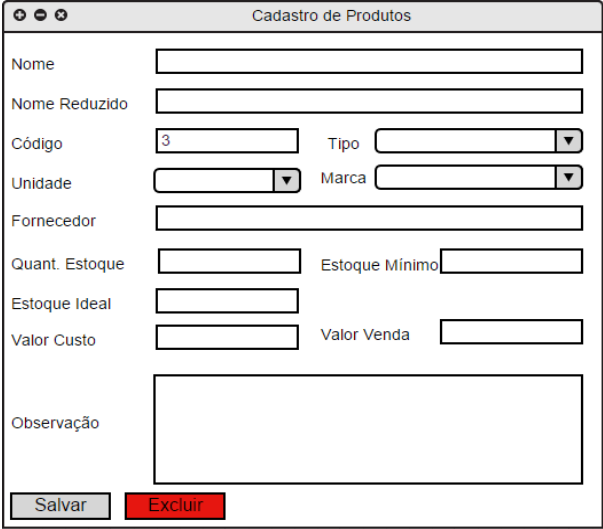
Grid

- **Código:** Exibir o Id do produto.
- **Nome:** Exibir o Nome do produto.
- **Marca:** Exibir a Marca do produto.
- **Unidade:** Exibir a unidade do produto.
- **Quantidade Estoque:** Exibir a quantidade do produto no estoque.
- **Valor da Venda:** Exibir o valor da venda do produto.
- **Tipo:** Exibir o tipo do produto.

Botões:

- **Botão Pesquisar:** Deverá exibir as informações de acordo com os filtros preenchidos. Se não houver filtros preenchidos, o sistema exibirá todas as informações do tipo selecionado.
- **Novo:** Deverá chamar a tela de Cadastro de Produtos

[Cadastro de Produtos]



Formulário de Cadastro de Produtos:

- Nome: Campo de texto.
- Nome Reduzido: Campo de texto.
- Código: Campo numérico (contém o valor 3).
- Tipo: ComboBox.
- Unidade: ComboBox.
- Marca: ComboBox.
- Fornecedor: Campo de texto.
- Quant. Estoque: Campo numérico.
- Estoque Mínimo: Campo numérico.
- Estoque Ideal: Campo numérico.
- Valor Custo: Campo numérico.
- Valor Venda: Campo numérico.
- Observação: Área de texto.
- Botões: Salvar (cinza), Excluir (vermelho).

- **Nome:** [Campo obrigatório]. Campo texto com tamanho máximo de 100 caracteres.
- **Nome Reduzido:** [Campo obrigatório]. Campo texto com tamanho máximo de 50 caracteres.
- **Código:** [Campo obrigatório]. Campo numérico com tamanho máximo de 100 caracteres sugerido pelo sistema.
- **Tipo:** [Campo Obrigatório]. ComboBox contendo todos os tipos de produtos.
- **Unidade:** [Campo Obrigatório]. ComboBox contendo as medidas de unidade.
- **Marca:** [Campo Obrigatório]. ComboBox contendo todas as marcas dos produtos.
- **Fornecedor:** [Campo Obrigatório]. Combobox contendo todos os fornecedores cadastrados no sistema.
- **Quantidade Estoque:**[Somente Leitura] Campo numérico com tamanho máximo de 3 dígitos.
- **Estoque Mínimo:** [Somente Leitura] Campo numérico com tamanho máximo de 3 dígitos.
- **Estoque Ideal:**[Somente Leitura] Campo numérico com tamanho máximo de 3 dígitos.
- **Valor Custo:** [Aplicar Máscara]. Campo numérico com tamanho máximo de 5 caracteres.
- **Valor Venda:** [Aplicar Máscara]. Campo numérico com tamanho máximo de 5 caracteres.
- **Observação:** Área de texto, alfanumérico, com tamanho máximo de 100 caracteres.

Regras:

- **Botão Salvar:** Quando acionado, deverá salvar o cadastro. Exibir a mensagem:
 - [Mensagem – “Produto cadastrado com sucesso!”].

Não permitir salvar caso campos obrigatórios estejam em branco. Exibir mensagem:

- [Mensagem – “*Campos obrigatórios não preenchidos*” - nome dos campos não preenchidos -].

Caso exista erro no preenchimento dos campos. Exibir mensagem:

- [Mensagem – “*Erro ao cadastrar produto!* -].

- **Botão Excluir:** Excluir o cadastro. Exibir a mensagem:
 - [Mensagem – “*Confirma exclusão do produto? Sim, Não*”].
- **Quantidade Estoque:** Deverá ser preenchido pelo sistema, apenas com exibição em tela.
- **Estoque Mínimo:** Deverá ser preenchido pelo sistema, apenas com exibição em tela.
- **Estoque Ideal:** Deverá ser preenchido pelo sistema, apenas com exibição em tela.

[DESENV] – Tipo de Produtos

Localização no Sistema

Esta rotina deverá estar disponível em MENU/Gestão/Produtos/Tipo de Produtos.

Dados para Implementação

[Pesquisa de Tipo de Produtos]

Pesquisa de Tipo de Produtos

Tipo

Pesquisar

Novo

▼ Código	▼ Tipo	▼ Descrição
0001	Água	Água Mineral Maseratti 200ML
0002	Cerveja	Cerveja Skol Garrafa 600ML
0003	Refrigerante	Coca-Cola Lata 350ML

- **Tipo:** Campo alfanumérico com tamanho máximo de 100 caracteres.

Grid

- **Código:** Exibir o Id do produto.
- **Tipo:** Exibir o Tipo do produto.
- **Descrição:** Exibir a Descrição do produto.

Botões:

- **Botão Pesquisar:** Deverá exibir as informações de acordo com os filtros preenchidos. Se não houver filtros preenchidos, o sistema exibirá todas as informações do tipo selecionado.
- **Novo:** Deverá chamar a tela de Cadastro de tipo de produto.

[Cadastro de Tipo de Produto]

O diagrama mostra uma janela de software com o título "Cadastro de Tipo de Produto". Dentro da janela, há dois campos de texto rotulados "Tipo" e "Descrição". Abaixo desses campos, há um botão com o texto "Salvar".

- **Tipo:** Campo alfanumérico com tamanho máximo de 100 caracteres.
- **Descrição:** Campo alfanumérico com tamanho máximo de 100 caracteres.

Regras:

- **Botão Salvar:** Quando acionado, deverá salvar o cadastro. Exibir a mensagem:
 - [Mensagem – “Tipo de produto cadastrado com sucesso!”].
 Não permitir salvar caso campos obrigatórios estejam em branco. Exibir mensagem:
 - [Mensagem – “Campos obrigatórios não preenchidos” - nome dos campos não preenchidos -].

DOCUMENTO DE ARQUITETURA DE SOFTWARE

Data	Autor	Descrição	Versão
30/08/2015	Giuliano Costa	Criação do documento	0.1

OBJETIVO

O Documento de Arquitetura de Software provê uma visão geral da arquitetura através de diferentes tipos de visões para descrever os diferentes aspectos do sistema.

Representação Arquitetural

A Arquitetura do HMA (Hermes Management Assistant), será representada por 4 visões:

- Visão de Casos de Uso Significativos: Esta seção apresenta alguns casos de uso ou cenários (obtidos dos diagramas de caso de uso), arquiteturalmente mais significativos para o software final e que, portanto, são utilizados para validação da arquitetura proposta;
- Visão Lógica: Esta seção apresenta a visão lógica da arquitetura. Tem por objetivo dar uma visão estática e dinâmica do sistema. Para se conseguir isto esta visão será dividida em:
 - Elementos do Modelo Arquiteturalmente Significativos: Aqui conterão os diagramas que representam graficamente as camadas, subsistemas e pacotes do projeto.
 - Realização dos Casos de Uso: Neste pacote, serão armazenadas as informações das realizações de casos de uso relacionadas aos aspectos dinâmicos e classes específicas (visão estática) utilizados nos Casos de Uso.
- Visão de Componente: Esta seção descreve a estrutura geral do modelo de implementação, a divisão do software em camadas e subsistemas no modelo de implementação e todos os componentes significativos do ponto de vista da arquitetura.
- Visão de Implantação: Esta visão compreende o diagrama de implantação (visão topológica da aplicação) e os elementos que o compõem: processos, dispositivos, conexões, componentes de cada processo, configurações usadas para desenvolvimento, teste e produção.

Observação

Este documento pode ser alterado a qualquer momento, sendo necessário a inclusão no cabeçalho o motivo, data e o integrante responsável pela modificação.

Objetivos e Restrições Arquiteturais

A arquitetura proposta tem como objetivo disponibilizar um sistema acessível à todos os funcionários do estabelecimento, utilizando camadas separadas logicamente e aderentes à arquitetura especificada expondo funcionalidades de negócios e/ou aplicação.

- **Módulo Gestão**

Módulo responsável pela gestão do estabelecimento e do sistema: Agenda de eventos, Produtos, Fornecedores, Funcionários, Gestão de Usuários, Abertura e Fechamento de caixa, Abertura e Fechamento de comandas, Sangria e reforço do caixa;

- **Módulo Pedidos**

Módulo responsável por controlar os pedidos dos clientes;

Requisitos Não Funcionais

Entende-se por requisitos não funcionais termos relacionados ao desempenho, usabilidade, confiabilidade e segurança da aplicação;

- **Portabilidade**

- O sistema deverá ser hospedado em servidor Windows Server, preferencialmente versão 2012;

- **Implementação**

- O sistema deverá ser desenvolvido utilizando .NET principalmente C# (Cê Sharp), por ser a linguagem em que a equipe possui maior controle;

- **Interoperabilidade**

- O sistema deverá comunicar-se com Sql Server, pela facilidade de integração com a linguagem acima definida;
- O sistema deverá ser multi-thread, visando a performance de requisições;
- Toda requisição ao servidor deverá ser assíncrona.

- **Éticos**

- O sistema não deverá mostrar dados financeiros a usuários não autorizados (garçom, segurança, cozinha);

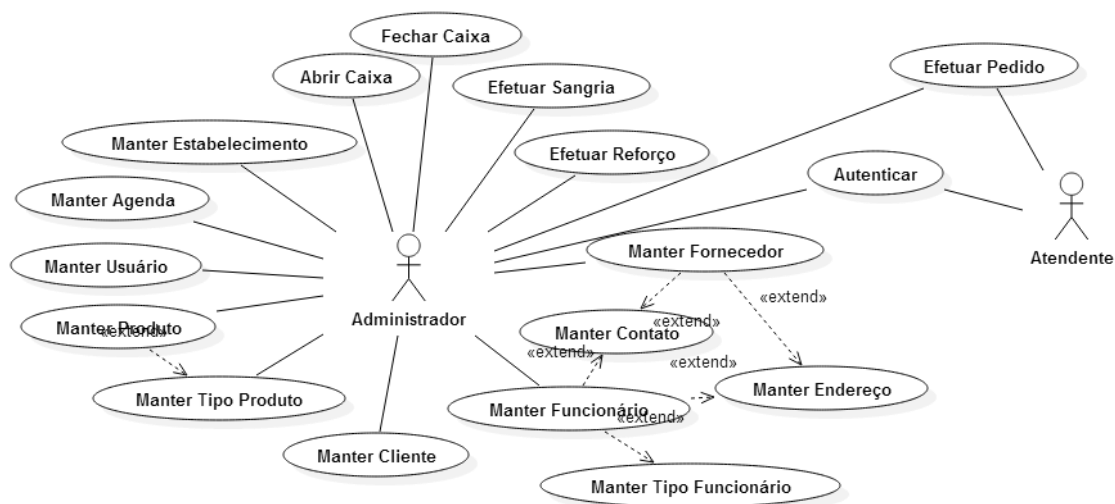
Requisitos Funcionais

Entende-se por requisitos funcionais funções que definem um software ou parte dele;

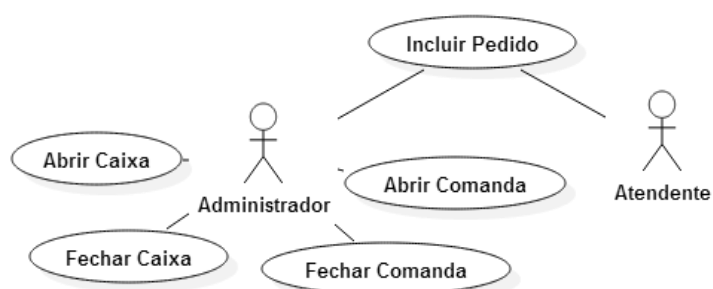
- O sistema deverá ser capaz de trabalhar com qualquer impressora não fiscal disponíveis atualmente no mercado (Somente impressoras com produção e suporte técnico atual);
- O sistema deverá rodar nas plataformas: computadores, smartphones e tablets.

As restrições arquiteturais ficam por conta do conhecimento técnico sobre implementação e o prazo estipulado para o desenvolvimento. O sistema não efetuará backup automaticamente;

Visão de Casos de Uso



Visão de Casos de Uso Significativos



Os casos de uso significativos foram escolhidos utilizando por base os processos de maior impacto para o estabelecimento, não foi efetuado um levantamento de risco dos casos de uso pela não necessidade, visto que a implementação dos mesmos segue o mesmo padrão de complexidade dos demais uc's.

Breve descrição das funcionalidades dos Casos de Uso Significativos:

- **Abrir Caixa:**

Neste caso de uso é realizada a abertura do caixa nos dias de abertura do estabelecimento, somente após a conclusão do fluxo principal deste caso de uso, é possível a realização dos demais casos de uso significativos.

- **Fechar Caixa:**

Neste caso de uso é realizado o fechamento diário do estabelecimento, com demonstração de lucro e efetivo financeiro.

- **Abrir Comanda:**

Neste caso de uso é realizado a abertura da comanda (entrada do cliente) no estabelecimento. Somente após conclusão deste caso de uso é possível iniciar o caso de uso Incluir pedidos.

- **Fechar Comanda:**

Neste caso de uso é realizado o fechamento da conta do cliente (encerramento da comanda). Após este caso de uso, é possível iniciar novamente o caso de uso Abrir Comanda.

- **Incluir Pedidos:**

Este caso de uso poderá ser realizado tanto pelo usuário(s) gestão como o usuário garçom. UC responsável por armazenar e gerir os pedidos realizados pelos clientes do estabelecimento;

Visão Lógica

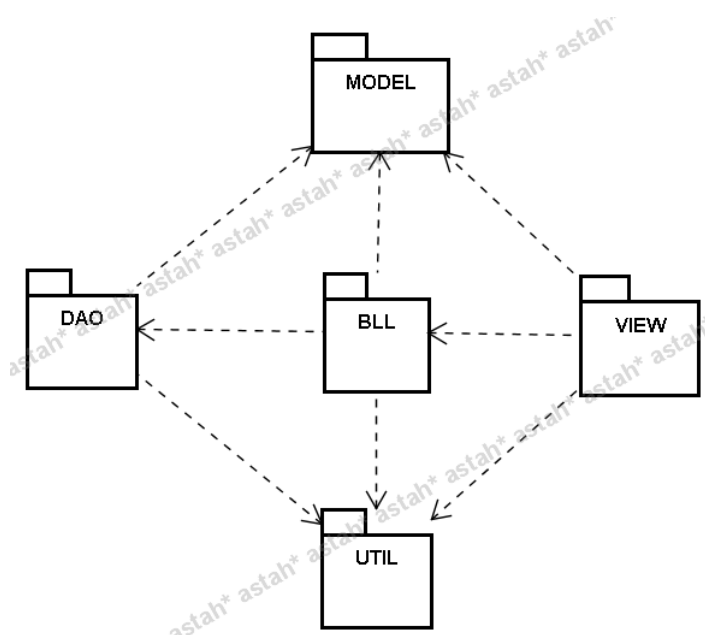
- **Descrição**

O sistema será desenvolvido utilizando a plataforma .NET, mais especificamente C# para controlar regras de negócio (Core) da aplicação e ASP MVC 4, HTML 5, JavaScript e CSS 3 responsáveis pela interface (FrontEnd). Toda comunicação com o CORE da aplicação dará através de uma camada externa (WCF), que terá como principal função o encapsulamento dos métodos.

O armazenamento das informações será através de banco de dados Microsoft Sql Server 2012 Express.

Pacotes e Subsistema

Visão responsável por demonstrar de forma macro todos os projetos de sistema;



- **Model:** Responsável por conter todos os objetos (modelos) que irão percorrer o sistema como um todo (exceto a DAO);
- **Entity:** Responsável por conter todos os objetos (modelos) de banco, os objetos aqui presentes deverão ser cópias exatas da estrutura do banco de dados;
- **DAO (Data Access Object):** Responsável pelo acesso a banco de dados (consultas, inserções, edições e exclusões).
- **BLL (Business Logic Layer):** Camada responsável por conter todas as regras de negócio;
- **WCF :** Camada responsável pela comunicação das aplicações de comunicação com usuário e a camada de regra de negócio.
- **VIEW:** Camada responsável pela interface com usuário, e transporte das informações a camada de abstração (WCF);
- **UTIL:** Camada responsável por conter objetos e funções úteis a todas as camadas do sistema, validações, verificações, conversores e etc;

Realização dos Casos de Uso Significativos

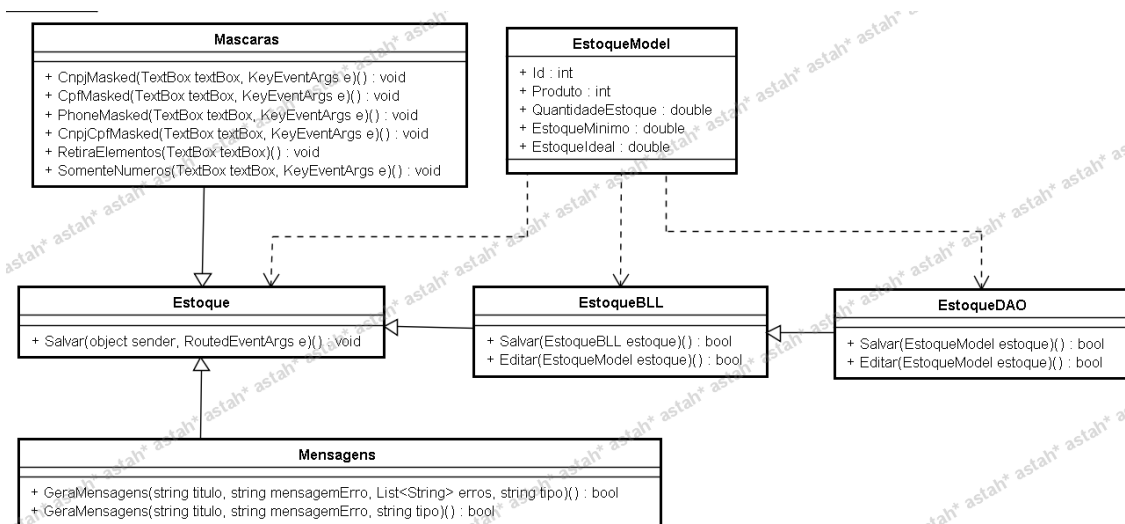
As realizações de caso de uso aqui descritos somente serão demonstrados os fluxos principais, sendo responsabilidade dos documentos de especificação de caso de uso a demonstração dos demais fluxos (alternativos e exceção).

Todos os casos de uso deverão ser estruturados conforme pastas referenciadas no arquivo *HMA_Project.juth* (em anexo), sendo desnecessário a diagramação de pacotes nesta fase documento.

[Manter Estoque]

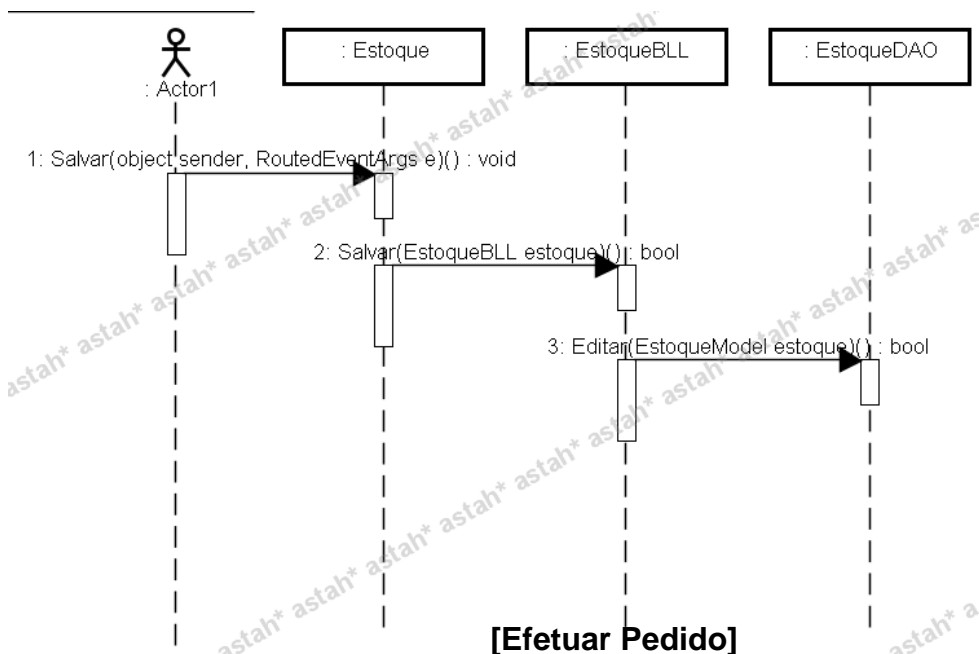
Visão Estática

A visão estática é dada através dos diagramas de classe a seguir



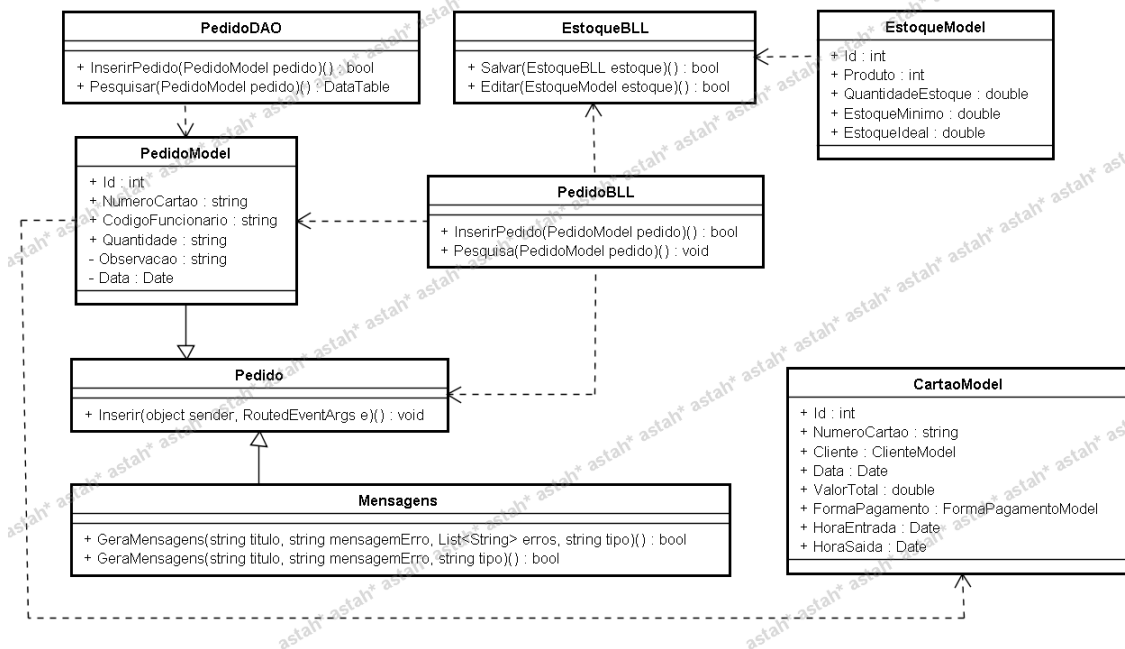
Visão Dinâmica

A visão dinâmica é dada através do diagrama de sequência a seguir (Fluxo Principal)



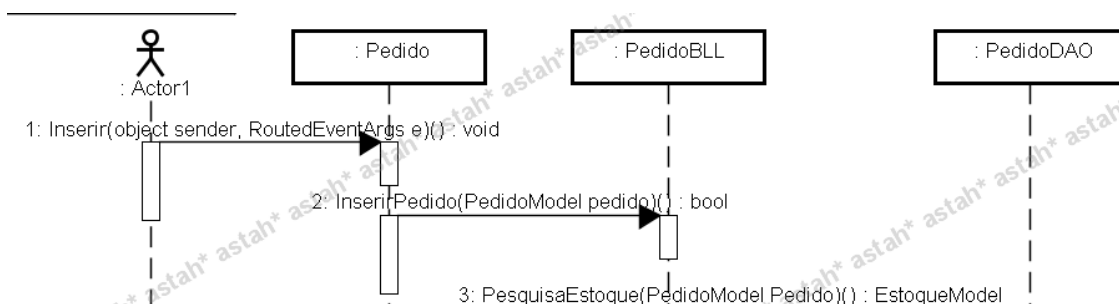
Visão Estática

A visão estática é dada através dos diagramas de classe a seguir



Visão Dinâmica

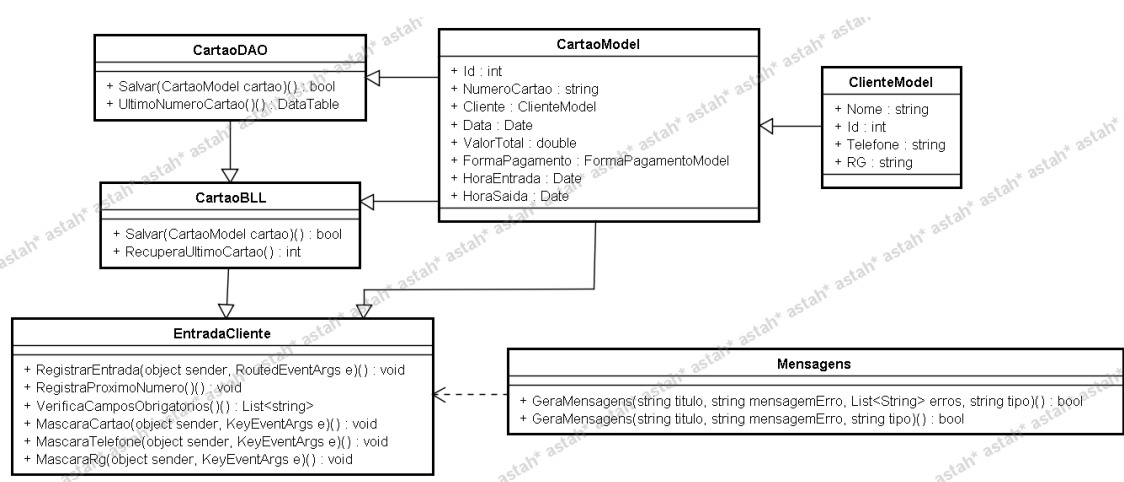
A visão dinâmica é dada através do diagrama de sequência a seguir (Fluxo Principal)



[Manter Cartão]

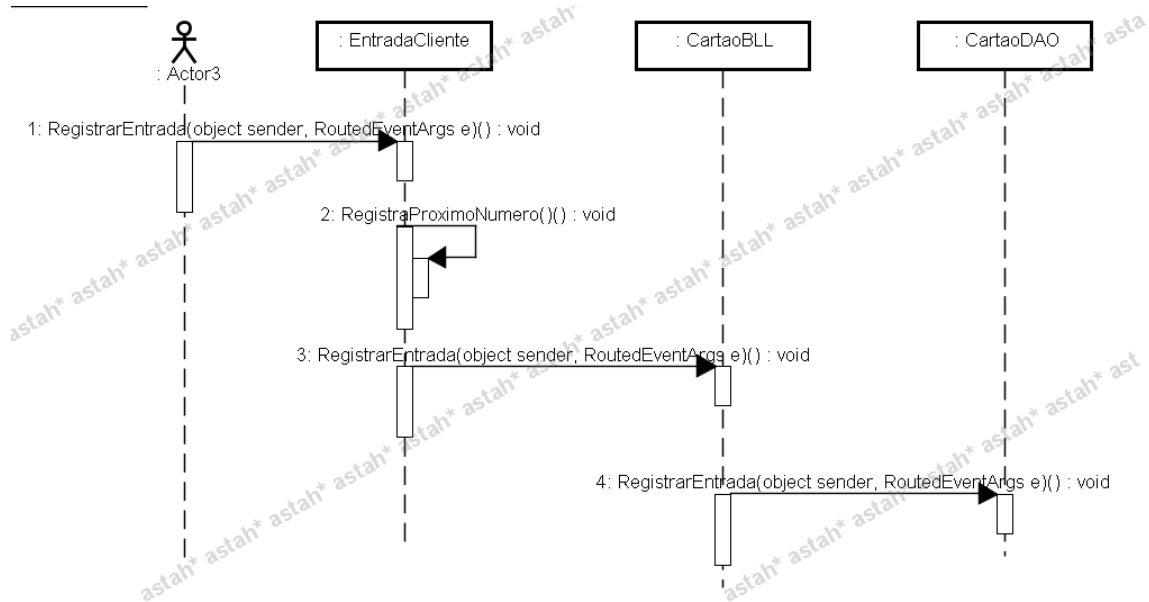
Visão Estática

A visão estática é dada através dos diagramas de classe a seguir



Visão Dinâmica

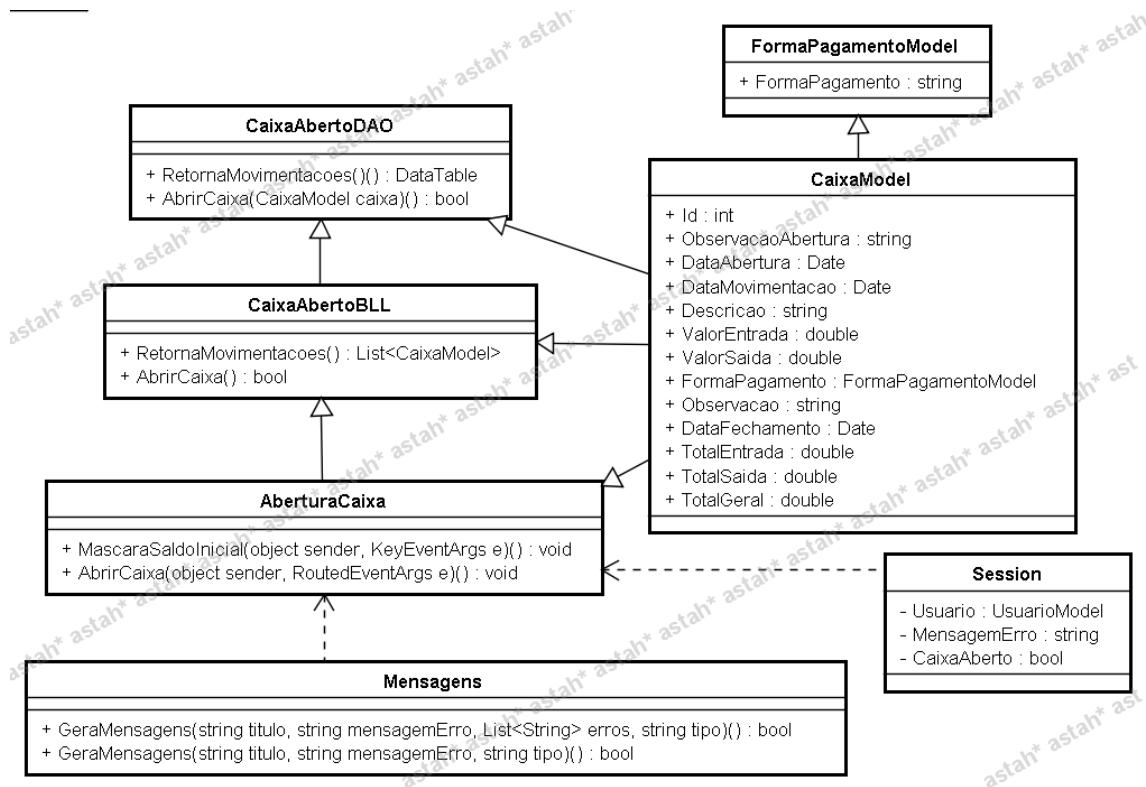
A visão dinâmica é dada através do diagrama de sequência a seguir (Fluxo Principal)



[Abrir Caixa]

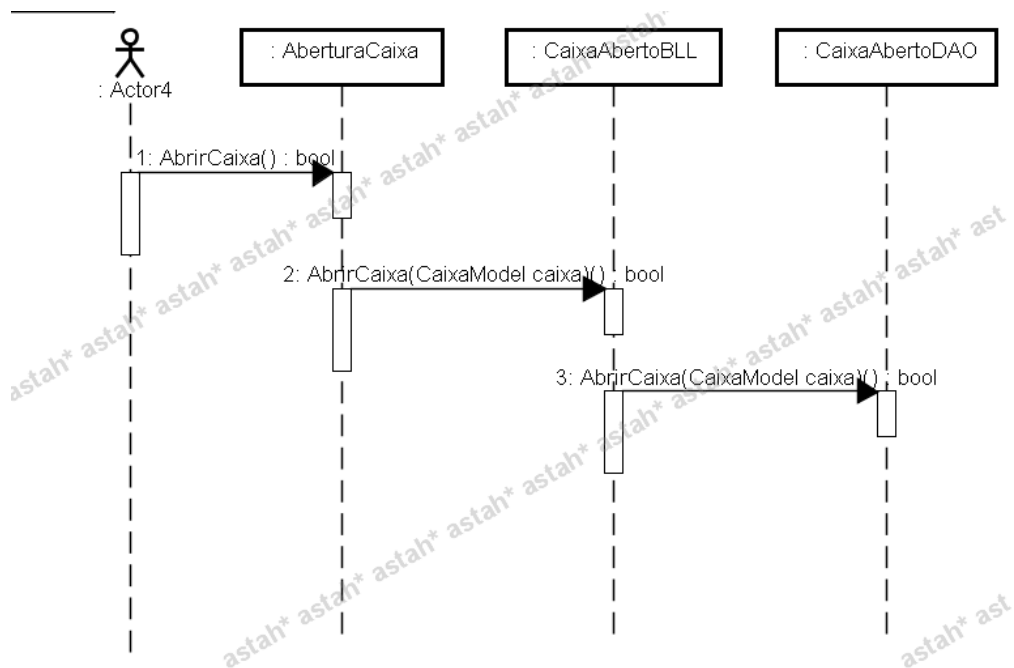
Visão Estática

A visão estática é dada através dos diagramas de classe a seguir



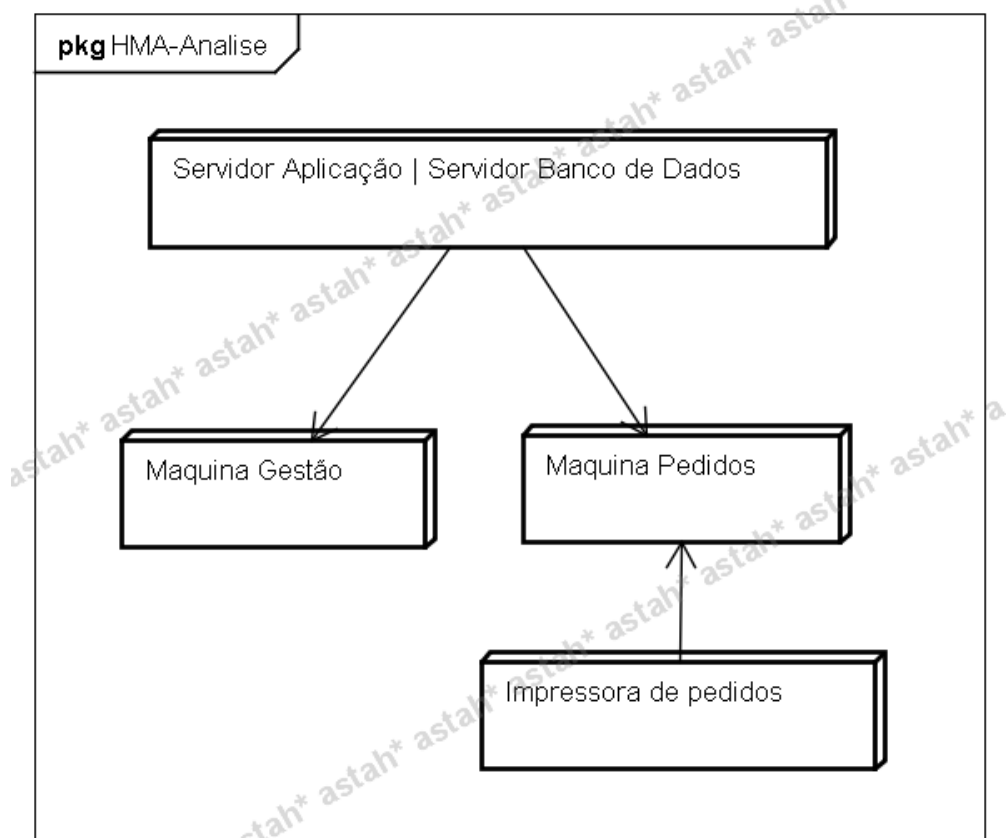
Visão Dinâmica

A visão dinâmica é dada através do diagrama de sequência a seguir (Fluxo Principal)



Visão de implantação

A implantação do HMA, deverá ser realizado conforme diagrama abaixo.



Entendemos que a separação dos servidores de aplicação e banco de dados seria de grande valia, tanto pela segurança quanto pela manutenção dos mesmos. Porém, por se tratar de uma aplicação com baixa quantidade de requisições simultâneas, e pelo estabelecimento não possuir estrutura suficiente para suportar tal divisão, acreditamos que a utilização de um único servidor multi-core, poderá suprir as necessidades iniciais de nosso cliente.

Visão de Dados

Por utilizarmos a camada ENTITY, todas as classes são cópias dos objetos de banco. Por este motivo, torna-se desnecessário a diagramação de tabelas. No arquivo *HMA_Project.juth* (em anexo), a camada *Model* está disponível para consulta.

Tamanho e Performance

O sistema em sua totalidade deverá suportar 10 acessos simultâneos, sendo que em média 4 acessos serão realizados. Por não haver armazenamento de arquivos no banco de dados, a quantidade de informações (em sua grande maioria VARCHAR) armazenadas diariamente não possui relevância em relação a utilização do BD.

Qualidade

- **Características Positivas**

- A instanciação de regras de negócio criadas diretamente em procedures favorece a manutenção da aplicação, não havendo necessidade de parar a utilização para realizar correções e/ou pequenas atualizações;
- O desenvolvimento de ferramentas próprias também favorece caso futuramente alguma aplicação de terceiro deixe de prestar manutenção e/ou suporte;
- O desenvolvimento de um WCF favorece o desenvolvimento para novas plataformas, não ficando restrito a apenas uma tecnologia;

- **Características Negativas**

- A criação de regra de negócio em banco de dados possui um ponto negativo. A necessidade de profissionais específicos, faz com que o custo de manutenção, ou seja, superior de uma aplicação com toda regra em código;
- Como os arquivos de NFe e emissão de recibos serão salvas no próprio servidor, caso haja qualquer problema no hardware de armazenamento, o risco de perda de arquivos é alto. Sendo necessário backup manual dos mesmos.

