**Documento de Visão de Projeto**



**Games Fun - ECOMMERCE**

Desenho de um círculo

Descrição gerada automaticamente com confiança média

**Histórico de Versões**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Data** | **Versão** | **Descrição** | **Autor** | **Revisores** |
| 07/09/2021 | 1.0 | Modelagem e desenvolvimento | Eduardo Borges | Eduardo Borges |

|  |  |
| --- | --- |
| **Cliente** | EB Games para Consoles |
| **Documento** | Documento de Visão de Projeto: *Games Fun* |
| **Data** | 07/09/2021 |
| **Autor** | **Eduardo da Silva Franco Borges**  eduardo.borges01@fatec.sp.gov.br |

**Página de Assinaturas**

|  |  |  |
| --- | --- | --- |
| Revisado e Aprovado por: |  |  |
|  |  | 07/09/2021 |

**Sumário**

1.Obejtivo 04

1.1 Escopo 04

1.2 Referências 04

2. Necessidade de Negócio 04

3. Objetivo do Projeto 04

4. Declaração preliminar de Escopo 05

4.1 Descrição 05

4.2 Produtos a serem entregues 05

4.3 Requisitos 05

4.3.1. Requisitos Funcionais 05

4.3.2. Requisitos Não funcionais 05

4.3.3. Regras de negócio 06

5. Premissas 06

6. Influência das partes interessadas 06

7. Representação Arquitetural 07

7.1 Restrições arquiteturais 08

7.2 Objetivos e restrições arquiteturais 08

8. Visão de caso de Uso 09

8.1 Diagrama de Caso de Uso de Manter Dados de Clientes (Visão Administrador) 09

8.2 Diagrama de Caso de Uso de Cadastro de Cliente (Visão do cliente) 10

8.3 Diagrama de Caso de Uso de Edição de Cliente (Visão Cliente) 11

8.4 Descrição dos Casos de Uso Arquiteturalmente Significativos 12

9. Diagramas de sequência 13

9.1 Diagrama de Sequência de cadastrar cliente 13

9.2 Diagrama de Sequência de editar cliente 13

9.3 Diagrama de Sequência Fluxo de condução 14

10. Visão de Lógica 15

10.1 Camada de Apresentação 16

10.2 Camada de Negócio 16

10.3 Camada de Persistência 17

11. Diagramas de Classe 18

11.1 Implementação 19

11.2 Domínio 20

12. Visão de Implantação 21

13. Visão de Implementação 22

14. Visão de Dados 22

15. Tamanho e Performance 26

16. Qualidade 26

17. Cronograma Macro 26

18. Referências 27

Objetivo

Este documento trata principalmente da documentação das necessidades de negócios, da justificativa do projeto, do entendimento atual das necessidades do cliente e descreve resumidamente o novo produto, serviço ou resultado que deve satisfazer esses requisitos.

Tem o objetivo de alinhar as expectativas dos interessados para formalizar o início do projeto. Apresentar uma visão arquitetural dos sistemas da linha **Ecommerce**. O intuito é salientar diferentes aspectos deste produto, obtidos a partir de decisões arquiteturais realizadas no âmbito dos sistemas da linha **Ecommerce**.

* 1. Escopo

O escopo deste documento trata do desenvolvimento de um sistema de vendas online especificamente voltado para jogos de mídia física.

Este documento mostrará as partes significativas do ponto de vista da arquitetura do modelo de design, como sua divisão em subsistemas e pacotes. Além disso, mostra sua divisão em classes e utilitários de classe.

* 1. Referências

Para a construção deste documento foram utilizadas as seguintes referências:

* Reuniões informais entre o grupo EB Games para Consoles.
* Documentos elaborados pela EB Development:

PROPOSTA COMERCIAL ECOMMERCE.docx

Documento de Requisitos (GamesFun).docx

Este documento influencia os seguintes documentos:

* Documento de Requisitos
* Proposta comercial

Necessidades de Negócio

Um sistema ecommerce para vendas é necessário para que a empresa possa dentro dele efetuar uma exposição de seus itens a venda, neste caso produtos da linha jogos em mídia física, a fim de difundir sua marca e ter um retorno de investimento, onde através do sistema terão o controle modularizado de partes do negócio, como controle de estoque, controle de vendas com múltiplas formas de pagamento, cadastramento de novos fornecedores e clientes da loja e afins.

O sistema deverá possibilitar a coleta analítica de movimentações dentro da plataforma através de gráficos, indicando crescimento/baixa de vendas referentes a um determinado produto ou categoria de produto em um determinado período.

Deverá haver separação de dashboards e rotas permitidas em relação à prioridade de usuário, fazendo assim a diferenciação de um usuário padrão comum (clientes da loja) e usuários administradores, que terão encargo de controlar as funções de gerenciamento de produtos, clientes, fornecedores e pedidos recebidos.

Objetivo do Projeto

Desenvolver uma plataforma para soluções web capaz de:

* Armazenar informações em uma base de dados;
* Utilizar o protocolo HTTP;
* Ser executada em qualquer navegador;
* Ter padrão de usabilidade responsiva para diferentes resoluções;

Desenvolver uma solução para ecommerce de vendas de jogos de mídia física, que possibilite:

* Controlar o estoque de produtos presentes no estoque e entrada/saída de novos, e também seus respectivos motivos de entrada/saída, podendo variar desde venda à avaria ou recall;
* Gerenciamento de movimentações dentro do sistema, como vendas, pedidos realizados, alterações cadastrais, cancelamentos, trocas etc.;
* Integração com o sistema utilizado pela instituição de ensino

Dessa forma, maximizando os lucros da empresa e facilitando o intermédio de pagamentos online, além de permitir através dos registros obtidos a cada transação o controle analítico das operações realizadas e uma projeção estatística através de gráficos do lucro obtido na plataforma em um determinado período.

Declaração Preliminar de Escopo

Esta seção descreve, em alto nível, o escopo do projeto. Os requisitos serão mais bem detalhados nos documentos de Requisitos e Proposta Comercial.

* 1. Descrição

O GamesFun é uma plataforma online que funciona como um ecommerce de vendas voltado à Jogos em mídia física, como jogos para consoles, computadores e portáteis e afins. Na plataforma os clientes poderão ver os produtos atuais à venda e fazer pedidos, além de contar com sistema de troca e demais procedimentos feitos exclusivamente online.

* 1. Produtos a serem entregues

Os seguintes itens são considerados produtos do projeto, na sua etapa 1.

* Sistema ecommerce completo, etapa 1, implementado de acordo com a especificação feita na fase de análise. (código objeto e código fonte).
* Documentos de especificação do sistema, concebido na fase de elaboração.
* Hospedagem do sistema em ambiente online, com funcionamento 24 x 7.
  1. Requisitos

É importante salientar que o documento de visão será feito no momento de iniciação do projeto. No qual, ainda não se tem todo o detalhamento. Portanto, os requisitos devem descritos em alto nível e depois detalhados em outro documento, caso o projeto seja aprovado.

* + 1. Requisitos Funcionais

O sistema deve ser capaz de efetuar o cadastro, exclusão, alteração e consulta de clientes da plataforma;

O sistema deve ser capaz de efetuar o controle de vendas;

O sistema deve ser capaz de fazer o controle de estoque completo, gerenciando tanto as movimentações de saída quanto de entrada no estoque;

O sistema deve ser capaz de produzir uma listagem de produtos vendidos na loja, indicando disponibilidade em estoque ou não;

O sistema deve ter controle de autenticação e privilégios de usuário comum/administrador;

O sistema deve possibilitar trocas caso haja problemas no produto ou em quaisquer fases do pedido feito pelo cliente;

O sistema deve permitir a consulta analítica através de gráficos sobre as vendas em um determinado período de tempo para determinado produto ou categoria de produto.

* + 1. Requisitos Não Funcionais

Utilizar linguagem Java;

Utilizar o banco de dados MySQL;

A arquitetura da solução obedecer aos designs patterns MVC e DAO;

Os demais designs patterns a ser seguidos na codificação são: Facade, Strategy, Command e ViewHelper;

O sistema deve rodar nos seguintes browsers:

* IE/Edge
* Google Chrome
* Firefox
  + 1. Regras de Negócio

Todo cliente cadastrado deverá ter endereços de entrega e de cobrança definidos;

Os cartões aceitos deverão possuir alguma das bandeiras dentre as aceitas pela loja.

Deverá haver uma classificação em ranking par ao cliente, medida pelas transações feitas pelo mesmo;

A compra somente será aprovada após retorno da empresa do cartão de crédito, definindo se o pagamento foi aprovado ou não autorizado;

Um produto só poderá ser adicionado ao carrinho se houver disponibilidade em estoque;

Deverá ser garantida a adição de apenas 1 cupom de promoção à compra;

Só será possível pedido de troca para compras que tiverem sido entregues;

O estoque não deverá permitir as entradas e saídas sem uma data definida.

Premissas

* A premissa principal do projeto é trazer para o cliente uma plataforma multifuncional que seja simples e intuitiva ao usuário, e que permita o mesmo ter o controle completo de finanças, estoque de produtos, exposição dos mesmos aos clientes e demais funcionalidades já descritas.

Através do desenvolvimento com utilização de boas práticas e padrões difundidos no mercado, o produto entregue funcionará de maneira ampla e fluida, permitindo que seu uso tenha uma longa vida útil sem que haja grandes preocupações por parte do cliente.

* Embora o objetivo do projeto seja a facilitação no gerenciamento e organização do cliente, com foco no aumento de lucros e difusão da marca, é importante ressaltar que o projeto não tem como premissa a divulgação ou quaisquer vínculos publicitários acerca da marca e sua exposição em plataformas terceiras ou indexação em buscadores.
* O projeto será orientado pelo professor Rodrigo Rocha.

Influência das Partes Interessadas

* Sandra da Silva Franco: Diretora na EB Games para Consoles e patrocinadora do projeto. Interessada em maximizar o lucro obtido com a venda de produtos da empresa, bem como facilitar o processo de compras e ter uma ferramenta de análise dos lucros e baixas do negócio.
* Eduardo Borges: Arquiteto e desenvolvedor do GamesFun: Interessado em entregar como produto do projeto um sistema capaz de fazer a gestão completa de vendas e controle de estoque para a EB Games para Consoles, e que dentro de seu desenvolvimento estejam atreladas boas práticas seguidas pelo mercado de TI, bem como padrões e arquiteturas pré-definidas que possibilitem um funcionamento íntegro, fluído e confiável.

Representação Arquitetural

Os sistemas serão desenvolvidos tendo como base a arquitetura ilustrada na Figura 1. Toda a arquitetura será baseada em alguns dos padrões de projetos tradicionais e padrões arquiteturais do GoF e também nos padrões Java EE sendo executados dentro de um Servidor de Aplicações.

Diagrama

Descrição gerada automaticamente

**Figura 1 - Modelo Arquitetural Genérico**

O Container WEB abrigará os componentes da arquitetura responsáveis pela camada de apresentação. Será utilizada a tecnologia JSP para a visualização, com base na implementação do modelo MVC.

As Classes de Domínio são as classes que representam os objetos de manipulação simples, contendo somente os seus respectivos atributos e os métodos getters/setters.

As Classes de Negócio representam as classes responsáveis por aplicar as regras de negócio do sistema como, por exemplo, Valida CPF. Constarão também nas classes de negócio o relacionamento com os DAOs responsáveis por persistir e recuperar os objetos no banco de dados.

Ainda na camada de Negocio, para manipulação dos Boundaries (como exemplo banco de dados), utiliza-se o padrão Facade (ou Fachada), onde o componente de Fachada assimila-se a um Controller Genérico que se encarrega de encaminhar as operações para o Banco de Dados e retornar o resultado de cada uma delas. Para esta arquitetura de software, será utilizado o padrão de projeto Command, que será a entidade a invocar uma instância da Fachada, sendo acionado por um componente de ViewHelper, que está presente ainda dentro do container Web, com função de redirecionamento e visualização gráfica para o usuário final.

O diagrama a seguir, Figura 2, representa a organização das classes dentro dos pacotes, tanto da arquitetura como da aplicação que fará uso da arquitetura. É importante atentar para a padronização de nomes de pacotes e a separação entre aplicação e arquitetura, bem como para o tipo de aplicação que se pretende desenvolver.

Diagrama

Descrição gerada automaticamente

**Figura 2 - Diagrama em pacotes**

* 1. Restrições Arquiteturais

Foram identificadas algumas orientações / restrições pertinentes ao desenvolvimento deste subsistema:

* Utilização do JDK 1.8 do Java;
* Utilização do servidor TomCat 7.X;
* Utilização do framework Highcharts para geração de gráficos analíticos;
* Utilização do SGBD MySQL.
  1. Objetivos e Restrições Arquiteturais

Alguns requisitos registrados que impactam diretamente a arquitetura do GamesFun são:

* A integração de API de pagamento com cartão de crédito para processo de vendas do sistema;
* A integração de API de cálculo de frete para as entregas;

Considerando premissas definidas para os Sistemas ecommerce pode-se citar as seguintes restrições:

* Utilização da Linguagem Java
* Considerar a utilização de software Livre, quando possível
* O Sistema de Gerenciamento de Banco de Dados a ser considerado em implementações de âmbito corporativo será o SGBD MySQL.

1. Visão de Caso de uso

Esta seção apresenta os Casos de Uso arquiteturalmente significativos, que foram selecionados considerando-se o pacote do Modelo de Casos de Uso que representa o GamesFun.

A classificação dos casos de uso, em termos de significância, foi realizada com base na observação de pelo menos um dos seguintes critérios:

* Casos de uso que estendem outros Casos de Uso
* Casos de Uso que são incluídos em outros Casos de Uso e
* Casos de uso que acessam sistemas externos
  1. Diagrama de Caso de Uso de Manter Dados de Clientes (Visão Administrador)

Diagrama

Descrição gerada automaticamente

Figura 3.1 Diagrama de Caso de Uso de Manter Dados de Clientes (Visão Adminstrador)

* 1. Diagrama de Caso de Uso de Cadastro de Cliente (Visão do cliente)

Diagrama

Descrição gerada automaticamente

Figura 3.2 Diagrama de Caso de Uso de Cadastro de Cliente (Visão Cliente)

* 1. Diagrama de Caso de Uso de Edição de Cliente (Visão Cliente)

Diagrama

Descrição gerada automaticamente

Figura 3.3 Diagrama de Caso de Uso de Edição de Cliente (Visão Cliente)

* 1. Descrição dos Casos de Uso Arquiteturalmente Significativos

**Manter dados de cliente (Visão Administrador)**

Este caso de uso se inicia após a a conclusão da operação de cadastro de cliente, uma vez que todos os dados tenham sido preenchidos corretamente. Através dele um usuário é cadastrado no sistema com as credenciais inseridas pelo administrador no gerenciamento de clientes, na sessão de cadastro.

**Cadastrar Usuário (Visão Cliente)**

Este caso de uso se inicia após a a conclusão da operação de cadastro de cliente, uma vez que todos os dados tenham sido preenchidos corretamente. Através dele um usuário é cadastrado no sistema com as credenciais inseridas pelo cliente no ato do cadastro.

**Editar dados de Cliente (Visão Cliente)**

Este caso de uso se inicia após a a conclusão da operação da edição do cliente, uma vez que todos os dados tenham sido preenchidos corretamente. Através dele o usuário atrelado ao cliente tem os dados atualizados, de acordo com o que foi preenchido pelo cliente.

1. Diagramas de sequência
   1. [Uma imagem contendo Diagrama

      Descrição gerada automaticamente](diagrama_sequencia_cadastrar_usuario.pdf)Diagrama de Sequência de cadastrar cliente

**Clique na imagem para ver em detalhe (Botão direito > abrir link)**

**Clique na imagem para ver em detalhe (Botão direito > abrir link)**

**Clique na imagem para ver em detalhe (Botão direito > abrir link)**

* 1. Diagrama de Sequência de editar cliente

**Clique na imagem para ver em detalhe (Botão direito > abrir link)**

[Uma imagem contendo Gráfico

Descrição gerada automaticamente](diagrama_sequencia_editar_usuario.pdf)

* 1. [Gráfico

     Descrição gerada automaticamente](diagrama_sequencia_fluxo_venda.pdf)Diagrama de Sequência Fluxo de condução

**Clique na imagem para ver em detalhe (Botão direito > abrir link)**

1. Visão de Lógica

Esta visão apresenta elementos de design significativos do ponto de vista da arquitetura, descrevendo a organização do Sistema GamesFun em pacotes, bem como a organização desses pacotes em camadas.

O Diagrama com as camadas do sistema GamesFun é ilustrado na figura 4.1.

Diagrama

Descrição gerada automaticamente

Figura 4.1: Camadas do GamesFun com as dependências de tecnologia

**Apresentação**: Contém classes para as interfaces gráficas com os usuários (GUI). Através destas interfaces os usuários conseguem interagir com o GamesFun, com o intuito de incluir, alterar e excluir produtos, clientes, fazer pedidos etc.

**Negócio**: Contém classes que controlam a execução das funcionalidades do GamesFun.

**Persistência**: Contém classes responsáveis por persistir as entidades de modelo. Por exemplo,

contém as classes que permitem ler e gravar os objetos no banco de dados relacional.

A

Figura 4.2 ilustra o diagrama de camadas com as tecnologias utilizadas no desenvolvimento, já descritas na figura 4.1.

Neste momento, é importante ressaltar que a camada de apresentação envolve componentes que são executados na porção servidora e na porção cliente. Na porção servidora, são executados os componentes que montam as páginas html/jsp de resposta e controlam o fluxo de interação com o usuário. Assim, envolve interpretação de páginas jsp, utilização de helpers para montagem das páginas de resposta, delegação de solicitação para a camada de negócio, obtenção de respostas da camada de negócio, dentre outros. Já na porção cliente, estão os códigos javascript que são executados nos navegadores dos usuários com o intuito de facilitar a interação. Por exemplo, podem fazer uma pré-verificação dos dados digitados (como o preenchimento de campos obrigatórios), solicitação de páginas via ajax, de modo a tornar mais dinâmica a interação com o usuário, dentre outros.

* 1. Camada de Apresentação

Nesta camada, temos o pacote viewHelper, que contém as classes de chamada de camadas de visualização e redirecionamento de requests e a pasta WebContent, que contém todos os arquivos relacionados à exibição de informações, o que engloba páginas JSP e HTML, imagens, javascript, arquivos de estilização, dentre outros. Já o pacote de controle desta camada, contém as rotas mapeadas via Servlet, responsáveis pela comunicação com as classes da camada de negócio. Estes pacotes podem ser vistos na Figura 4.1.

Diagrama

Descrição gerada automaticamente

Figura 4.1: Camada de Apresentação

* 1. Camada de Negócio

Nesta camada, temos o pacote Negocio, onde estão localizadas as strategies do sistema, as strategies são as classes responsáveis por controlar as regras de negócio da aplicação. O pacote domain, contém as classes que representam o modelo, ou seja, aquelas que contém as informações sobre o Sistema GamesFun.

A imagem 4.2 ilustra os pacotes descritos.

Diagrama

Descrição gerada automaticamente

Figura 4.2: Camada de Negócios

* 1. Camada de Persistência

Nesta camada temos o pacote dao que contém as classes e interfaces responsáveis por persistir as informações do GamesFun no BD relacional.

Interface gráfica do usuário

Descrição gerada automaticamente

Figura 4.3: Camada de Persistência.

1. Diagramas de Classe
   1. Implementação

A figura 5.1. ilustra as principais classes de controler, negócio, validações etc., que juntas somam a implementação do sistema como um todo.

Diagrama

Descrição gerada automaticamente

Figura 5.1: Classes de implementação

* 1. Domínio

A figura 5.2. ilustra as principais classes de domínio da aplicação.

Diagrama

Descrição gerada automaticamente

Figura 5.2: Classes de domínio

1. Visão de Implantação

Esta seção descreve as configurações da rede física (hardware) na qual o GamesFun será implantado e executado.

Trata-se de uma visão do Modelo de Implantação que, para a configuração em questão, indica meios de utilização possíveis para o sistema (computadores e dispositivos mobile), que executarão o GamesFun, uma vez que todo o gerenciamento de servidores e pacotes, sistema de deploy e afins serão totalmente realizados com tecnologias aliadas à Web. A figura 6 ilustra o modelo de implantação para o GamesFun.

Diagrama

Descrição gerada automaticamente

Figura 6: Visão de Implantação do GamesFun

Na Figura 6 observa-se os seguintes nós físicos:

* **Web Client Application (WCA):** Aplicativos com interface de usuário via navegador, construídos com base no Framework Bootstrap e nas tecnologias JSP.
* **Servidor DB**: Nó que contém o BD Central do Sistema GamesFun
* **Servidor online hospedado**: Servidor de hospedagem onde é mantido o sistema.

1. Visão de Implementação

Esta visão descreve a estrutura geral de implementação, a decomposição do software em camadas de implementação.

A estrutura geral de implementação para o GamesFun é baseada na estrutura da Visão Lógica, assim, não há necessidade de detalhar os diagramas de camadas e pacotes de implementação, uma vez que são fortemente baseados naqueles desenvolvidos para Visão Lógica.

1. Visão de Dados

O mecanismo de persistência utilizado no sistema GamesFun utiliza-se o banco de dados Relacional MySQL. O controle de transações está sendo feito de maneira a convergir com o padrão DAO e demais padrões do GoF.

A figura 7 apresenta a visão física da base de dados do GamesFun.

Linha do tempo

Descrição gerada automaticamente

Figura 7 – Modelo Físico

A figura 8 apresenta modelo lógico da base de dados do GamesFun.

Diagrama, Esquemático

Descrição gerada automaticamente

Figura 8 – Modelo Lógico

Tabela 2: Mapeamento Objeto-Relacional

|  |  |
| --- | --- |
| **Classe** | **Entidade** |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

1. Tamanho e Performance

O sistema GamesFun será usado para realização de vendas e controle de estoque de produtos (jogos de mídia física) de uma loja especializada, consequentemente terá uma grande base.

As estimativas do número de usuários e de carga de utilização em períodos de pico de utilização, bem como maiores informações sobre questões relacionadas ao tamanho e desempenho do sistema GamesFun podem ser obtidas no documento de requisitos não funcionais.

1. Qualidade

O sistema GamesFun será usado para as vendas de jogo em mídia física, consequentemente tratando de altos volumes financeiros e um grande número de operações de compra/venda, movimentações de estoque e cadastramentos de clientes diariamente.

Eventuais erros e/ou falhas na sua operação podem levar a prejuízos significativos tanto em termos financeiros quanto na imagem da EB Games para Consoles, portanto na fase de design deve-se levar em consideração como fatores prioritários a confiabilidade e robustez do sistema.

Adicionalmente, o sistema GamesFun pode ser alvo de ataques de “hackers” para roubar ou simplesmente corromper informações, possibilidade aumentada pela interface do sistema disponível na Internet, para evitar que tais ataques sejam bem-sucedidos uma infraestrutura de segurança deve ser especificada e projetada.

Maiores informações sobre questões relacionadas aos requisitos de qualidade do sistema GamesFun podem ser obtidas no documento de requisitos não funcionais.

1. Cronograma Macro.

|  |  |
| --- | --- |
| **Resultado** |  |
| Plano Preliminar | Semana 2 |
| Plano Fase 1 | Semana 3 |
| Especificação Fase 1 | Semana 5 |
| Piloto Fase 1 | Semana 11 |
| Solução Testada Fase 1 | Semana 12 |
| Plano Fase 2 | Semana 13 |
| Especificação Fase 2 | Semana 14 |
| Piloto Fase 2 | Semana 20 |
| Solução Testada Fase 2 | Semana 21 |

Obs: Os prazos apresentados são uma estimativa inicial considerando as informações disponíveis nesta etapa do projeto. Um cronograma detalhado será elaborado na fase de planejamento e, eventualmente, estes prazos podem ser modificados.

1. Referências

Unified Modeling Language: <http://www.omg.org/technology/documents/formal/uml.htm>

RUP. Rational Unified Process.