

Universidade Federal de Ouro Preto



Engenharia de Software II

Sistema de *Empréstimo de Jogos*

Grupo: *caféina++*;

Alunos:	Caio Soares Costa Cibele Oliveira Ferreira Eduardo Matosinhos Florinda Gabriel Caetano Araújo
Professor:	Johnatan Oliveira
Horário:	Seg e Qua - 08:20 - 10:00

Ouro Preto, 29 de março de 2021

Conteúdo

1	Histórico de Revisões	1
2	Processo e Software	1
3	Cronograma	1
4	Levantamento de Requisitos	2
4.1	Diagrama de Caso de Uso	2
4.2	Descrição de Caso de Uso	2
5	Especificação de Requisitos	3
5.1	Requisitos Funcionais	3
5.2	Requisitos Não Funcionais	5
6	Plano de VVT	5
6.1	Requisitos a serem testados	5
6.2	Estratégias e ferramentas de teste	6
6.3	Equipe e infra-estrutura	6
6.4	Execução do Plano de Teste	6
7	Medição e Qualidade de Software	6
8	Observações	6
9	Referências	7

1 Histórico de Revisões

Data	Versão	Descrição	Autor
10/03/2021	0.0	Justificativa do processo de software	Eduardo
10/03/2021	0.0	Levantamento de requisitos	Caio, Cíbele e Gabriel
20/03/2021	0.0	Especificação de requisitos	Caio, Cíbele e Gabriel

Tabela 1: Revisões do Documento

2 Processo e Software

A equipe “caféina++;”, desenvolvedora do software “Game Stonks”, adotou o processo de desenvolvimento incremental. Esta escolha levou em consideração o fato deste modelo permitir obter feedback rápido do cliente a partir de pequenas entregas iterativas. Desta forma, o custo de acomodar mudanças nos requisitos e o risco de falha global do projeto é reduzido. Ademais, é possível validar o sistema parcial a cada incremento, sendo uma vantagem já que o cliente não precisa esperar a entrega final para usá-lo. Portanto, os serviços de mais alta prioridade são entregues primeiros.

3 Cronograma

Nome	Tarefa	Prazo
Eduardo	Definir tema do projeto	01/03
Gabriel	Criar repositório no GitHub	01/03
Eduardo	Escolha e justificativa do Processo de Desenvolvimento de Software	13/03
Caio, Cíbele e Gabriel	Levantamento dos requisitos do software	13/03
Caio, Cíbele e Gabriel	Especificação de requisitos	20/03
Gabriel	Definição da tecnologia, arquitetura e classes	20/03
Caio, Cíbele, Eduardo e Gabriel	Definir backlog	24/03

Tabela 2: Cronograma

4 Levantamento de Requisitos

4.1 Diagrama de Caso de Uso

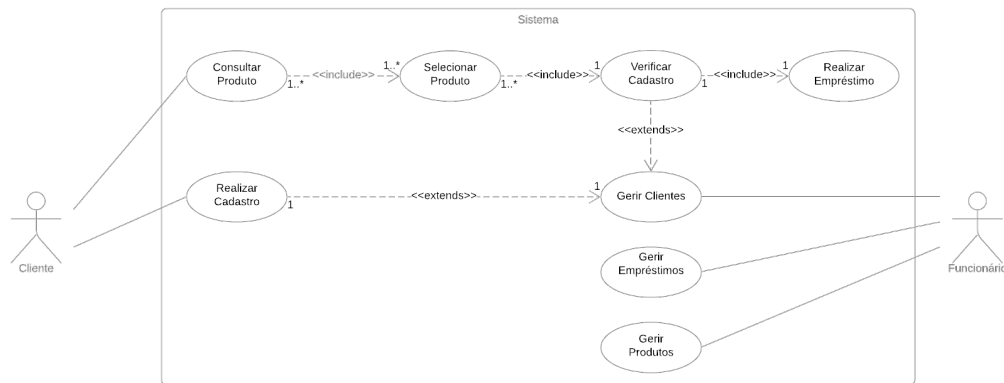


Figura 1: Diagrama de Casos de Uso

4.2 Descrição de Caso de Uso

Código	Serviço	Descrição
RF01	Realizar cadastro	Permite o cliente se cadastrar no sistema.
RF02	Consultar produtos	Permite o cliente consultar um jogo no sistema.
RF03	Selecionar produto	Permite o cliente selecionar um jogo para empréstimo.
RF04	Verificar cadastro	Possibilita a verificação do cadastro do cliente no sistema.
RF05	Realizar empréstimo	Efetiva o empréstimo do produto para o cliente que o selecionou.
RF06	Gerir clientes	Gerencia os clientes que estão cadastrados no software.
RF07	Gerir empréstimos	Gerencia os empréstimos realizados.
RF08	Gerir produtos	Gerencia os produtos (jogos) do software.

Tabela 3: Descrição de Casos de Uso

5 Especificação de Requisitos

5.1 Requisitos Funcionais

RF01– Realizar cadastro: Este serviço realiza todas as tarefas relacionadas à autenticação.

- **Criar login:** Campo de texto para login.
- **Criar senha:** Campo de texto para senha.
- **Recuperar senha:** Permite recuperar a senha.
- **Alterar senha:** Permite alterar a senha.
- **Autenticar login e senha:** Valida o login e senha.
- **Sair do sistema:** Realiza o logout.

RF02– Consultar Produto: Este serviço possibilita que o usuário consulte se o jogo que ele irá pegar emprestado está disponível.

- **Filtrar Jogos:** Possibilita que o usuário filtre os jogos de acordo com tema, nome e empresa de desenvolvedora.
- **Consultar Disponibilidade:** Possibilita que o usuário veja se um jogo estará disponível para empréstimo.

RF03– Selecionar Produto: Este serviço permite ao usuário adicionar um jogo ao carrinho para o empréstimo.

- **Adicionar Jogo:** Permite adicionar jogos ao carrinho.
- **Remover Jogos:** Permite remover jogos do carrinho.
- **Atualizar valor de jogos no carrinho:** Atualiza o valor total dos jogos para empréstimo.

RF04– Verificar cadastro: Este serviço faz a verificação do cadastro do usuário e validação das informações

- **Autenticação de login:** Verifica se o usuário está cadastrado no sistema antes de prosseguir com o empréstimo.
- **Autenticação de informações:** Verifica se as informações inseridas pelo usuário estão corretas.

- **Verificar pendências:** Verifica se o usuário possui algum débito registrado no sistema.

RF05– Realizar Empréstimos: Este serviço possibilita que o usuário efetue o empréstimo do jogo que deseja.

- **Validar dados do cartão:** Valida os dados de cartão de crédito do usuário.
- **Efetua pagamento:** Utiliza uma API para realizar o pagamento do empréstimo solicitado pelo usuário.
- **Libera Empréstimo:** Disponibiliza para o usuário os jogos que ele escolheu por empréstimo.

RF06– Gerir clientes: Este serviço realiza todas as tarefas relacionadas ao cadastro de usuários do sistema.

- **Cadastrar usuário:** Permite cadastrar um usuário no sistema.
- **Consultar usuário:** Permite visualizar todos os dados de um usuário registrado.
- **Buscar usuário:** Permite fazer uma busca de um usuário no sistema.
- **Alterar dados do usuário:** Permite atualizar todos os dados relacionados a um usuário.
- **Remover usuário:** Permite excluir o usuário do sistema.

RF07– Gerir empréstimos: Este serviço realiza todas as tarefas relacionadas à contabilidade de gastos e lucros do Game Stonks.

- **Consultar empréstimos:** Permite ao administrador visualizar os empréstimos realizados.
- **Consultar lucro por jogo:** Permite ao administrador visualizar lucro por empréstimos de jogos.
- **Consultar resumo financeiro:** Permite ao administrador visualizar o resumo financeiro.
- **Exportar resumo financeiro:** Exporta relatório em formato CSV.
- **Registrar gastos com compra de jogos:** Permite que o administrador adicione os gastos com a compra de jogos.

- **Registrar lucro por jogo:** Permite que o administrador adicione os lucros por empréstimos de jogos.

RF08– Gerir produtos: Este serviço realiza todas as tarefas relacionadas ao gerenciamento dos jogos.

- **Registrar jogos:** Permite que o administrador adicione as informações pertinentes aos jogos.
- **Consultar jogos:** Permite ao administrador visualizar os jogos cadastrados no sistema.
- **Editar jogos:** Permite ao administrador editar as informações dos jogos cadastrados no sistema.
- **Excluir jogos:** Permite ao administrador excluir jogos existentes no sistema.

5.2 Requisitos Não Funcionais

Requisitos Não Funcionais são as propriedades que as funções devem ter, tais como desempenho e usabilidade. Não se detenha ao seu nome pouco apropriado (nós o usamos porque é a maneira mais comum de se referir a estes tipos de requisitos)—estes requisitos são tão importantes quanto as exigências funcionais, para o sucesso do produto.

RNF01. A efetivação da reserva do pacote, só deve ser liberada após o cliente estar logado no sistema. **Informações:** usuário e senha. **Regras:** o cliente terá acesso para comprar, consultar e alterar.

6 Plano de VVT

Asseguram que o software cumpra com suas especificações e atenda às necessidades dos usuários. Você deve apresentar um plano de testes, ferramentas que serão utilizadas e coisas do tipo.

Veja um exemplo no link: https://www.cin.ufpe.br/~gta/rup-vc/extend.formal_resources/guidances/examples/resources/test_plan_v1.htm

6.1 Requisitos a serem testados

Esta seção descreve em linhas gerais o conjunto de requisitos a serem testados no projeto a ser desenvolvido, comunicando o que deve ser verifi-

cado. Exemplos de requisitos a serem testados são: desempenho, segurança, interface de usuário, controle de acesso, funcionalidades.

6.2 Estratégias e ferramentas de teste

Apresenta um conjunto de tipos de testes a serem realizados, respectivas técnicas empregadas e critério de finalização de teste. Além disso, é listado o conjunto de ferramentas utilizadas.

6.3 Equipe e infra-estrutura

Contém descrição da equipe e da infra-estrutura utilizada para o desenvolvimento das atividades de testes, incluindo: pessoal, equipamentos, software de apoio, materiais, dentre outros. Isto visa garantir uma estrutura adequada para a execução das atividades de testes previstas no plano.

6.4 Execução do Plano de Teste

7 Medição e Qualidade de Software

Apresente aqui o formato da Medição e qualidade de software. Você deve mostrar os meios que irá avaliar a qualidade do seu software. Apresente o plano e os resultados a partir da prática de ferramentas de detecção de code smells, por exemplo. Em Java, temos uma ferramenta chamada JDEODORANT. Você pode avaliar as métricas de qualidade também, por exemplo, em Java, temos CKMetrics¹

8 Observações

Apresente aqui as dificuldades na disciplina, trabalho prático e coisas do tipo.

¹<https://github.com/mauricioaniche/ck>

Funcionalidade: Envelope	Tempo Despendido(h): 1h
Contador: 02	Criticidade: Baixa

Objeto de Teste:		
<ul style="list-style-type: none"> ➤ Validar uso do assistente para criar documento no modelo Envelope. ➤ Verificar impressão usando um envelope. 		
Descrição do Caso de Teste:		
<ul style="list-style-type: none"> ➤ O aplicativo deverá atender corretamente a configuração fornecida pelo usuário (nas 3 abas: Envelope, Formato e Impressora). ➤ A impressão no envelope deverá ser realizada com sucesso. 		
Pré-Condição:		
<ul style="list-style-type: none"> ➤ O usuário deverá informar dados do Destinatário e Remetente. ➤ Existir uma impressora compatível para impressão no envelope. 		
Dados de Entrada:		
ID	Passo	Procedimento
1	P1	Executar o aplicativo LibreOffice 4.2 (Opção texto)
	P2	Selecionar a opção Texto
	P3	Acessar o menu a opção Envelope (Inserir>>Envelope)
	V1	O aplicativo deverá exibir a tela do assistente para Envelope
	P3	Preencher os dados do campo Destinatário e Remetente
	P4	Clicar no botão Inserir
	P5	Se necessário ajustar a largura da caixa de texto do Destinatário e Remetente
2	P7	Clicar no ícone da Impressora
	V2	Verificar se o envelope foi impresso corretamente.
Resultado Esperado: As operações deverão funcionar corretamente cumprindo todas as regras acima citadas.		

Figura 2: Exemplo

9 Referências

- [1] Sommerville, Ian – Software Engineering, 8th Edition;