

7-Beats

Plano de Teste

1. Objetivo

Este documento tem o propósito de definir, esclarecer e detalhar os métodos definidos, no Plano de Verificação e Validação, para serem usados na realização da verificação e da validação dos artefatos produzidos durante o desenvolvimento do projeto 7-Beats.

Os resultados referentes a verificação e validação são documentados em relatórios individuais, com identificação dos problemas encontrados e dos interessados a serem informados à respeito.

2. Cronograma e Procedimentos de Avaliação

Nessa sessão é descrita a ordem da realização dos testes e o procedimento de avaliação utilizado em cada teste. Em uma tabela, encontram-se o teste à ser realizado, os responsáveis pela realização deste, a forma de aplicação dos testes e o que este verifica/valida no projeto.

A ordem de realização das avaliações segue a ordem da tabela. Com exceção dos testes de unidade, que são feitos durante o desenvolvimento do código, os outros testes são realizados com a versão executável do código.

Tipo	Nome	Responsável	Aplicação	Verifica/Valida
Teste de Funcionalidade	de Unidade	Desenvolvedor	JUnit	Implementação de métodos de determinadas classes
	de Sistema	Cliente ou representante	Fluxos de Caso de Uso	Verifica execução dos casos de uso do ponto de vista do usuário
Teste Suplementar	de Performance (Teste de Carga)	Testador	Avaliação do sistema para uma quantidade de dados/transações típica	Padrões de tempo de resposta
	de Segurança	Testador	Fluxo de Caso de Uso relacionado aos Login	Integridade; Autenticação; Confidencialidade; Disponibilidade

	de Interface Com Usuário	Usuário final	Caso de Uso especificado	Verifica eficiência e eficácia da interface
Teste de Funcionalidade	de Aceitação (Teste Alfa)	Cliente ou representante	Uso livre do sistema	Valida software quanto aos requisitos (aprovação do cliente)

Tabela 01 - Cronograma de Testes Realizados

3. Caso de Teste

Nessa sessão são definidos os casos de teste utilizados para aplicação de testes. Os casos de teste forma baseados nos casos de uso especificado no documento 7BSW - EOR 2.0 e devem definir a saída esperada, de forma a reduzir a interpretação do critério de sucesso, verificando não somente as condições inválidas de execução, como também as condições válidas.

- Fluxo de Casos de Uso

Caso de Uso: Criar Login

- 1.[IN] O usuário informa um nome de usuário e senha.
- 2.[OUT] O sistema confirma o registro e abre a página inicial.
 - 2.1. Variante: Usuário já existe.

Variante 2.1: Usuário já existe

 - 2.1.1 [OUT] O sistema informa que o login único já existe.
 - 2.1.2 [IN] Usuário informa outro nome para o login único.

Caso de Uso: Pesquisar Música

1. [IN] O usuário informa o nome da música, artista, ou álbum.
2. [OUT] O sistema retorna a música ou o artista ou o álbum pesquisado.
 - 2.1. Variante: Musica, artista ou álbum não encontrado.

Variante 2.1: Musica, artista ou álbum não encontrado

 - 2.1.1 [OUT] O sistema informa que a música, artista ou álbum não existe.
 - 2.1.2 [IN] Usuário informa outro nome de música, artista ou álbum para pesquisa.

Caso de Uso: Controlar Música

1. [IN] O usuário pode: Iniciar música, pausar música ou resumir música.
2. [OUT] O sistema vai tocar a música, ou pausar a música ou retornar para a música.

Caso de Uso: Mudar Música

1. [IN] O usuário pode avançar ou retroceder na música ou pular para a próxima música ou para a anterior.
2. [OUT] O sistema vai avançar ou retroceder dentro da lista de música indo para a próxima música ou para a música anterior.
 - 2.1. Variante: Não ter nenhuma outra música.

- Variante 2.1: Não ter nenhuma outra música
- 2.1.1 [OUT] O sistema vai voltar ao início do ciclo de reprodução

Caso de Uso: Fazer Download

1. [IN] O usuário pode baixar uma música da tela de reprodução.
 2. [OUT] O sistema vai baixar a música selecionada.
 - 2.1. Variante: Música ser sincronizada ou já estar no dispositivo do usuário.
- Variante 2.1: Musica ser sincronizada ou já estar no dispositivo do usuário
- 2.1.1 [OUT] Não vai aparecer o botão de baixar.
- Variante 2.2: O usuário não ser cadastrado
- 2.2.1 [OUT]: Não vai aparecer o botão de baixar.

Caso de Uso: Embaralhar Músicas

1. [IN] O usuário na tela de reprodução pode usar o botão shuffle para deixar a reprodução das músicas aleatória.
2. [OUT] O sistema vai reproduzir a lista de músicas de forma aleatória.

Caso de Uso: Criar Playlist

1. [IN] O usuário pode criar uma lista de músicas personalizada, e pode editar ou deletar essa lista de músicas.
 2. [OUT] O sistema vai criar a lista de música , pode editar essa lista ou deletar ela.
 - 2.1. Variante: Adicionar música.
- Variante 2.1: Adicionar música
- 2.1.1 [OUT] O sistema adiciona o arquivo áudio a playlist especificada.
- 2.2. Variante: Remover música
- Variante 2.2: Remover música
- 2.2.1 [OUT] O sistema deleta o arquivo áudio da playlist especificada.
- 2.3 Variante: Deletar palylist
- Variante 2.3: Deletar Playlist
- 2.3.1 [OUT] O sistema remove a playlist da conta do usuário

- Tabela Casos de Teste para Caso de Uso

Caso de Teste	Objetivo	Caminho	Como Testar	Resultado Esperado
CST1.0 Criar Login	Fluxo principal com variante 2.1	1, 2, 2.1, 2.1.1, 2.1.2	Um novo usuário informa os dados necessário para criação de login único e o sistema cria o cadastro	Cadastro criado
CT2.1 Pesquisar Música (Nome da Música)	Fluxo principal, variante 2.1	1, 2, 2.1, 2.1.1, 2.1.2	O usuário busca pela caixa de pesquisa uma música pelo nome da música	Lista de músicas com nome compatível

CT2.2 Pesquisar Música (Artista)	Fluxo principal, variante 2.1	1, 2, 2.1, 2.1.1, 2.1.2	O usuário busca pela caixa de pesquisa uma música pelo nome do artista	Lista de músicas do artista especificado
CT2.3 Pesquisar Música (Álbum)	Fluxo principal, variante 2.1	1, 2, 2.1, 2.1.1, 2.1.2	O usuário busca pela caixa de pesquisa uma música pelo nome do álbum	Lista de músicas do álbum especificado
CST 3.1 Controlar Música (Iniciar)	Fluxo principal	1, 2	O usuário seleciona o botão “play” na interface gráfica	A música selecionada é executada
CST3.2 Controlar Música (Resumir)	Fluxo principal	1, 2	O usuário reinicia uma música, que já estava em reprodução	O sistema deve iniciar a execução do arqui vídeo a partir do momento que sua execução havia sido parada.
CST3.3 Controlar Música (Pausar)	Fluxo principal com variante 2.1	1, 2, 2.1.1	O usuário seleciona o botão “pause” na interface gráfica	A música em atual execução, tem a execução interrompida
CST4.1 Mudar de Música (Avançar)	Fluxo principal com variante 2.1	1, 2, 2.1, 2.1.1	O usuário seleciona o botão “avançar” na interface gráfica	O sistema altera a música em execução para a próxima na lista de reprodução
CST4.2 Mudar de Música (Retroceder)	Fluxo principal com variante 2.1	1, 2, 2.1, 2.1.1	O usuário seleciona o botão “retroceder” na interface gráfica	O sistema altera a música em execução para a anterior na lista de reprodução
CST5.0 Fazer Download	Fluxo principal com variante 2.1	1, 2, 2.1, 2.1.1	O usuário seleciona o botão “download” na interface gráfica	O sistema transfere os dados do arquivo áudio seleciona para o dispositivo do usuário
CST6.0 Embaralhar Músicas	Fluxo principal	1, 2	O usuário seleciona a opção “shuffle” na interface gráfica	O sistema reproduz a lista de execução com ordem aleatória de músicas reproduzidas
CST7.1 Criar Playlist	Fluxo principal	1, 2	O usuário cadastrado cria uma playlist com especificação de nome	O sistema cria uma playlist com o nome especificado na conta do usuário

CST7.2 Editar Playlist (Adicionar)	Fluxo principal com variante 2.1	1, 2, 2.1, 2.1.1	O usuário adiciona um novo arquivo áudio a uma playlist já criada	O sistema adiciona o arquivo áudio especificado a lista de músicas da playlist selecionada
CST7.3 Editar Playlist (Remover)	Fluxo principal com variante 2.2	1, 2, 2.1, 2.2.1	O usuário remove um arquivo áudio da playlist já criada	O sistema remove o arquivo áudio especificado a lista de músicas da playlist selecionada
CST7.4 Deletar Playlist	Fluxo principal com variante 2.3	1, 2, 2.3, 2.3.1	O usuário deleta uma playlist já existente	O sistema remove a playlist e seus arquivos da conta do usuário

Tabela 02 - Casos de Teste para Casos de Uso

- Caso de Teste Unitário

-

1-Pesquisa

A pesquisa tem três etapas:

1-Fornecimento dos dados de entrada

2-Pesquisa no SGBD

3-Resposta da Pesquisa

Caso de Teste	Condição	Entrada	Resultado Esperado
CTU 1.1	Entrada corresponde a algo no está no SGBD	The art of the peer	Servidor retorna true se pelo menos um album foi encontrado para o caso de teste testSearch1
CTU 1.2	Entrada não corresponde a algo no SGBD	dsfsdsfs	O metodo retorna false se nenhum resultado foi encontrado para o caso de teste testSearch2

2-Play

O play funciona em duas etapas:

1- A função Play recebe uma URL válida

2- A música começa a ser tocada

Caso de Teste	Condição	Entrada	Resultado Esperado
CTU 2.1	O player recebeu uma URL válida.	http://69.28.84.155/public/musicas/kendrick_lamar_sherane_aka_master_splinters_daughter.mp3	O método public boolean play retorna true para TestPlay2
CTU 2.2	O player recebeu uma URL inválida	aUrlStringParam	O método public boolean play retorna false para TestPlay1
CTU 2.3	A lista de Música é válida	List<Song> songList = new ArrayList<>(); songList.add(new Song(3, "", "", "", null));	O método public boolean isValidPlaylist retorna true para TestValidList1
CTU 2.4	A lista de Música é inválida	List<Song> songList = null;	O método public boolean isValidPlaylist retorna false para testValidList2

3- Abertura de Telas

3.1 - Abertura da Tela de Álbum

Caso de Teste	Condição	Entrada	Resultado Esperado
CTU 3.1	A tela executa os métodos onCreate, onCreateView e onStart	Instanciar a classe AlbumFragment	Os três métodos retornam verdadeiro

3.2 - Abertura de Tela de Player

Caso de Teste	Condição	Entrada	Resultado Esperado
CTU 3.2	A tela executa os métodos onCreate e onStart	Instanciar a classe Player	Os dois métodos devem retornar verdadeiro

3.3 - Abertura de Tela de Pesquisa

Caso de Teste	Condição	Entrada	Resultado Esperado
CTU 3.3	A tela executa os métodos onCreate, onCreateView e onStart	Instanciar a classe SearchFragment	Os três métodos retornam verdadeiro

3.4 - Abertura de Tela de Playlist

Caso de Teste	Condição	Entrada	Resultado Esperado
CTU 3.4	A tela executa os métodos onCreate, onCreateView e onStart	Instanciar a classe PlaylistFragment	Os três métodos retornam verdadeiro

4 - Log in

Caso de Teste	Condição	Entrada	Resultado Esperado
CTU 4.1	Senha entre 6 e 45 caracteres.	Login: augustoccesar senha:adminadmin	O metodo isValidPassword é chamado e retorna true para o caso de test testPass1
CTU 4.2	Senha com menos de 6 caracteres	Login: augustoccesar Senha: 12345	O metodo isValidPassword é chamado e retorna false para o caso de test testPass2
CTU 4.3	Senha com mais de 45 caracteres	Login: augustoccesar Senha: abcdefghijklmnopqr	O metodo isValidPassword é chamado e retorna false para o caso de test testPass3

		stuvwxyzabcdefghij klmnopqrstuvwxyz	
CTU 4.4	Senha em branco	Login: augustoccesar Senha:	O metodo isValidPassword é chamado e retorna false para o caso de test testPass3
CTU 4.5	Login entre 6 e 25 caracteres	Login: augustoccesar Senha: 123456	O metodo isValidUsernameLenght é chamado e retorna true para o caso de test testLogin1
CTU 4.6	Login com menos de 6 caracteres	Login: login Senha: 123456	O metodo isValidUsernameLenght é chamado e retorna false para o caso de test testLogin2
CTU 4.7	Login com mais de 25 caracteurs	Login: abcdefghijklmnopqrs tuvwxyz Senha: 123456	O metodo isValidUsernameLenght é chamado e retorna false para o caso de test testLogin3
CTU 4.8	Login em branco	Login: Senha: 123456	O metodo isValidUsernameLenght é chamado e retorna false para o caso de test testLogin4
CTU 4.9	Login com caracteres inválidos	Login: @123456 Senha: 123456	O metodo isValidNameFormat é chamado e retorna false para o caso de test testLogin5
CTU 4.10	Login com caracteres válidos	Login: augustoccesar Senha: adminadmin	O metodo isValidNameFormat é chamado e retorna true para o caso de test testLogin6

5 - Pause

Caso de Teste	Condição	Entrada	Resultado Esperado
CTU 5	Uma música está em execução	Função pause enquanto Money Trees está executando.	Retorna true se a música não estiver tocando para o caso de teste testPause1

6 - Avançar

Caso de Teste	Condição	Entrada	Resultado Esperado
CTU 6.1	Executar uma lista de músicas com mais de uma música. Não deve estar tocando a ultima música.	Função Avançar no meio de uma lista de músicas	Avançar para a próxima música de uma lista de músicas com mais de uma música.
CTU 6.2	Executar a ultima música de uma lista de músicas com mais de uma música.	Função Avançar no final de uma lista de músicas	Avançar para a primeira música em uma lista de músicas com mais de uma música.
CTU 6.3	Executar lista de músicas com apenas uma música.	Função Avançar em uma lista com apenas uma música.	Reiniciar a música

7 - Retroceder

Caso de Teste	Condição	Entrada	Resultado Esperado
CTU 7.1	Executar uma lista de músicas com mais de uma música. Não deve estar tocando a primeira música.	Função Retroceder no meio de uma lista de músicas	Retroceder para a música em uma lista de músicas com mais de uma musica
CTU 7.2	Executar a primeira música de uma lista de músicas com mais de uma música.	Função Retroceder no começo de uma lista de músicas	Retroceder para a ultima música em uma lista de músicas com mais de uma música.

CTU 7.3	Executar lista de músicas com apenas uma música.	Função Retroceder em uma lista com apenas uma música.	Reiniciar a Música
---------	--	---	--------------------

8 - Shuffle

Caso de Teste	Condição	Entrada	Resultado Esperado
CTU 8.1	Executar uma lista de músicas com mais de uma música	Função Shuffle em uma lista com mais de uma música	Reordenar aleatoriamente uma lista de musicas com mais de uma música.
CTU 8.2	Executar uma lista de músicas com apenas uma música.	Função Shuffle em uma lista com apenas uma música.	Executar mudanças na lista