

**Universidade Tecnológica Federal do Paraná
Departamento Acadêmico de Informática
Curso de Tecnologia em Desenvolvimento de
Sistemas Distribuídos**

**Alailson Bolzon Alonso
Ketty Akemi Mena de Oliveira**

**APLICAÇÕES COMERCIAIS PARA DISPOSITIVOS
MÓVEIS USANDO QR-CODE**

Trabalho de Conclusão de Curso

**CURITIBA
2012**

Alailson Bolzon Alonso
Ketty Akemi Mena de Oliveira

Aplicações Comerciais para Dispositivos Móveis
Usando QR-CODE

Proposta apresentada à disciplina de Trabalho de Conclusão de Tecnologia em Desenvolvimento de Sistemas Distribuídos, como requisito parcial para obtenção do título de tecnólogo em sistemas distribuídos.

Orientador: Paulo Roberto Bueno

Curitiba
2012

AGRADECIMENTOS

Agradecemos primeiramente ao professor Paulo Roberto Bueno, que foi nosso orientador e nos guiou permitindo que esse trabalho fosse criado. Queremos agradecer também a todos os professores que fizeram parte da nossa formação, nos instruindo e nos moldando, permitindo que nos tornássemos profissionais capazes de realizar esse projeto. Em especial para a professora e coordenadora Wânia Meira Matos Figueiredo, que nos instruiu por diversas dificuldades acadêmicas e sem a sua ajuda não teríamos conseguido chegar até aqui. Aos professores Robson Ribeiro Linhares, Leandro Batista Almeida e Mariângela de Oliveira Gomes Setti que plantaram em nosso ser o gosto pela programação, aos professores Luiz Augusto Pelisson, Fabiano Scriptore Carvalho e Ana Cristina Barreiras K. Vendramim por toda ajuda e apoio, e aos demais educadores pela paciência e orientação.

Por fim agradecemos aos nossos familiares, mães e irmãos que nos deram o carinho e apoio necessário em todo o processo de nossa criação até nessa fase final da nossa formação acadêmica.

Sumário

1	INTRODUÇÃO.....	1
1.1	Apresentação.....	2
1.2	Justificativa.....	3
1.3	Objetivo Geral.....	3
1.4	Objetivos Específicos.....	4
1.5	Estrutura e Organização.....	4
2	LEVANTAMENTO BIBLIOGRÁFICO E ESTADO DA ARTE.....	6
2.1	Descrição do QR-Code.....	6
2.2	Análise Mercadológica Brasileira.....	9
2.3	Uso do QR-Code no Brasil.....	12
3	METODOLOGIA.....	15
3.1	Fundamentos.....	15
3.1.1	ANÁLISE E DOCUMENTAÇÃO.....	15
3.1.2	MODELAGEM DE BANCO DE DADOS.....	16
3.1.3	CODIFICAÇÃO.....	16
3.1.4	TESTES.....	16
3.2	Tecnologias.....	17
4	RECURSOS DE HARDWARE E SOFTWARE.....	19
4.1	Recursos de Hardware.....	19
4.2	Recursos de Software.....	20
4.3	Viabilidade.....	21
4.3.1	VIABILIDADE TÉCNICA.....	21
4.3.2	VIABILIDADE ECONÔMICA.....	21
4.3.3	CONCLUSÃO.....	22
5	PROJETO DESENVOLVIDO E RESULTADOS.....	23
5.1	Sistema Web.....	23
5.2	Aplicativo Móvel Cliente.....	27
5.3	Aplicativo Móvel Cinema.....	29
5.4	Projeto de Software.....	31
5.4.1	LEVANTAMENTO DE REQUISITOS.....	32
5.4.1.1	Requisitos Funcionais.....	32
5.4.1.2	Requisitos Não-funcionais:.....	32
5.4.2	DIAGRAMAS DE CASOS DE USO.....	34
5.4.2.1	Casos de Uso do Sistema Web - Cliente.....	34
5.4.2.2	Casos de Uso do Sistema Web - Administrador.....	36
5.4.2.3	Casos de Uso do Aplicativo Móvel Cliente.....	38
5.4.2.4	Casos de Uso do Aplicativo Móvel Cinema.....	38
5.4.3	DIAGRAMAS DE CLASSES.....	39
5.4.3.1	Diagrama de Classe do Sistema Web.....	39
5.4.3.2	Diagrama de Classe do Aplicativo Móvel Cliente.....	43
5.4.3.3	Diagrama de Classe do Aplicativo Móvel Cinema.....	44
6	ESTIMATIVA POR MEIO DE PONTOS DE CASO DE USO.....	45
6.1	UUCP – Unadjusted Use Case Points	45
6.2	TFC – Technical Complexity Factor.....	47
6.3	EFC – Environmental Complexity Factor.....	48
6.4	PF – Productivity Factor.....	49
6.5	Cálculo Final de Horas Necessárias.....	50

7 CRONOGRAMA.....	51
8 PROCEDIMENTO DE TESTE E VALIDAÇÃO.....	54
9 ANÁLISE DE RISCOS.....	57
9.1 Planos de Contingência.....	59
10 CONCLUSÕES.....	60
10.1 Propostas de Melhorias.....	61
11 REFERÊNCIAS.....	62
APÊNDICE A – Detalhamento dos Casos de Uso.....	64
APÊNDICE B – Detalhamento dos Casos de Uso.....	98
APÊNDICE C – Detalhamento dos Dicionários de Informações.....	100
APÊNDICE D – Diagramas de Sequência.....	129
APÊNDICE E – Detalhamento dos Cronogramas.....	150

Lista de Figuras

Figura 1: QR-Code contendo o endereço virtual da UTFPR.....	1
Figura 2: Código QR utilizado pela Coca-Cola.	7
Figura 3: Comparação de tamanho entre código de barras 1D e 2D.	8
Figura 4: Identificação de microchip através de um micro QR-Code.....	8
Figura 5: Estrutura do código QR.....	9
Figura 6: Crescimento de celulares para uso pessoal de 2009-2011. ...	10
Figura 7: Evolução de acesso SMP de 1998 – 2011.	11
Figura 8: Acessos do SMP por tecnologia.	11
Figura 9: Primeiro anúncio utilizando QR-Code no Brasil.....	13
Figura 10: QR-Code de divulgação da Sprite.	14
Figura 11: Código QR com site da Marvel contendo clipe do filme.	14
Figura 12: Diagrama de Blocos do Projeto.....	23
Figura 13: Tela Inicial do Sistema Web de Compra de Ingressos.....	24
Figura 14: Tela de Auto Cadastro do Sistema Web.....	24
Figura 15: Tela de Gerenciamento de Créditos do Sistema Web.....	25
Figura 16: Tela de Gerenciamento de Ingressos Sistema Web.....	25
Figura 17: Tela de Menu para o Administrador do Sistema Web.....	26
Figura 18: Tela de Gerenciamento de Filmes Sistema Web.....	27
Figura 19: Tela do celular com o ícone do aplicativo móvel cliente.....	27
Figura 20: Telas de login do Aplicativo Móvel Cliente.....	28
Figura 21: Tela da Câmera do Aplicativo Móvel Cliente.....	28
Figura 22: Tela do celular com o ícone do aplicativo móvel do cinema	29
Figura 23: Tela inicial do aplicativo móvel do cinema.....	30
Figura 24: Tela da Câmera do Aplicativo Móvel do Cinema.....	30
Figura 25: Validação do código através de campo texto.....	31
Figura 26: Casos de Uso Geral do Sistema Web pelo cliente	35
Figura 27: Caso de Uso Geral do Sistema Web Acessado Pelo Administrador.....	36
Figura 28: Caso de Uso Aplicativo Móvel Cliente.....	37
Figura 29: Caso de Uso Aplicativo Móvel Cinema.....	38
Figura 30: Diagrama de classe Sistema Web - Relacionamentos.....	39
Figura 31: Classes Servlets.....	39
Figura 32: Classes de Entidade.....	40
Figura 33: Classes de Persistência.....	40
Figura 34: Classes Beans.....	41
Figura 35: Diagrama de Classes do Aplicativo Móvel Cliente.....	42
Figura 36: Diagrama de Classes do Aplicativo Móvel Cinema.....	43

Lista de Tabelas

Tabela 1: Capacidade de armazenamento do QR-Code.....	6
Tabela 2: Capacidade de correção de erros.....	7
Tabela 3: Custo de desenvolvimento.....	22
Tabela 4: Categorias de Caso de Uso(CLEMMONS, 2006).....	45
Tabela 5: Cálculo UUCW do projeto.....	46
Tabela 6: Categoria de Ator(CLEMMONS, 2006).....	46
Tabela 7: Cálculo do UAW do Projeto.....	47
Tabela 8: Cálculo do FatorT para o Projeto.....	48
Tabela 9: Cálculo do ECF.....	49
Tabela 10: Análise Documentação e Modelagem de Dados.....	51
Tabela 11: Programação do Sistema Web.....	52
Tabela 12: Programação dos Sistemas Móveis.....	52
Tabela 13: Etapa Final.....	52
Tabela 14: Análise de Riscos.....	57

LISTA DE ABREVIATURAS E SIGLAS

AMPS: Advanced Mobile Phone System.

Anatel: Agência Nacional de Telecomunicações.

API: Application Programming Interface.

BSD: Berkeley Software Distribution.

CPF: Cadastro de Pessoas Físicas.

DER: Diagrama de Entidade e Relacionamento.

EFC: Environmental Complexity Factor.

GB: Gigabyte.

IBGE: Instituto Brasileiro de Geografia e Estatística.

IDC: International Data Corporation.

IDE: Integrated Development Environment.

JDK: Java Development Kit.

JME: Java Micro Edition.

JSF: Java Server Faces.

JVM: Java Virtual Machine.

MB: Megabyte.

Mhz: Megahertz.

MVC: Model View Controller.

PDI: Primeira Data de Início.

PDT: Primeira Data de Término.

PF: Productivity Factor.

PNG: Portable Network Graphics.

QR: Quick Response.

QR-Code: Quick Response Code.

RAM: Random Access Memory.

SMP: Serviço Móvel Pessoal.

SQL: Structured Query Language.

TDMA: Time Division Multiple Access.

TFC: Technical Complexity Factor.

UAW: Unadjusted Actor Weight.

UDI: Última Data de Início.

UDT: Ultima Data de Término.

UML: Unified Modeling Language.

USB: Universal Serial BUS.

UUCP: Unadjusted Use Case Points.

UUCW: Unadjusted Use Case Weight.

ZXing: Zebra Crossing.

RESUMO

Tendo em vista o avanço tecnológico na área de telefones celulares, com o surgimento dos *smartphones* e com a internet cada vez mais acessível, tornou-se viável a implantação de projetos interligando aplicações web a aplicativos móveis. A ferramenta *QR-Code (Quick Response Code)* é largamente utilizada em países como o Japão, Estados Unidos e Coreia do Sul. Nesses países o QR-Code é utilizado para diversos fins como compra de produtos e controle de estoque. No Brasil, ele é utilizado em na área de marketing realizando o redirecionamento à endereços virtuais para divulgação de produtos. Será apresentado nesse projeto um sistema de cinema interativo, utilizando o QR-Code para realizar compra de ingressos e apresentá-lo como comprovante na entrada do cinema.

Palavras chaves: QR-Code, smartphone, aplicação web, aplicativos móveis.

ABSTRACT

Given the technological advancement in the mobile phones field, with the emergence of the smartphones and the Internet becoming more accessible, it has become practicable to implement a project that linking a web application to a mobile application. The QR-Code (Quick Response Code) tool is widely used in Japan, United States and South Korea. In those countries the QR-Code is used for several purposes such as product purchasing and inventory control. In Brazil, it is used for advertising performing redirection to virtual addresses for promotion of products. In this paper will be presented a system of interactive movie theater, using QR-Code to buy tickets and even present it as a voucher at the entrance of the movie theater.

Keywords: QR-Code, smartphone, web application, mobile application.

1 INTRODUÇÃO



Figura 1: QR-Code contendo o endereço virtual da UTFPR

O *QR-Code* (*Quick Response Code*) é uma imagem bidimensional que permite um amplo armazenamento de caracteres, conhecido popularmente como código de barras 2D. Foi criado em 1994 pela automobilística japonesa Denso-Wave. A Figura 1 demonstra um QR-Code do endereço virtual da UTFPR.

Inicialmente era utilizado para controle de estoque, catalogando diferentes partes na construção de veículos. Atualmente essa ferramenta é largamente utilizado em países como Japão, Estados Unidos e Coréia do Sul, mas ainda não se popularizou no Brasil.

A maioria das empresas brasileiras utilizam os códigos QR para armazenar um link, fazendo a divulgação de um site ou produto, mas a sua utilização pode ir muito além disso, podendo ser utilizado para a compra e venda de produtos.

Pensando em divulgar e demonstrar as possíveis utilizações dessa ferramenta é que decidimos realizar o trabalho de conclusão envolvendo esse tema, desenvolvendo três aplicações usando Java

móvel, plataforma Android e Java web, ambientado no tema cinema interativo.

Apelidado de ICine o sistema realizará quatro operações simples:

- O cliente, portando um celular com câmera, conexão com a internet e um cadastro no site do ICine digitalizará e decodificará uma imagem de *QR-Code* do filme no horário desejado. Essa imagem decodificada passará parâmetro para a aplicação web que retornará um e-mail contendo outro *QR-Code*, este por sua vez, será apresentado como *ticket* na entrada.
- O cliente, através do site do ICine poderá comprar ingressos para os filmes em exibição. Recebendo um *ticket* com o ingresso por e-mail que será validado na entrada do cinema.
- Validação dos ingressos através de um aplicativo móvel instalado no celular do funcionário do cinema.
- O cliente, portando um celular com câmera, conexão com a internet e um cadastro no site do ICine digitalizará e decodificará uma imagem de *QR-Code* promocional recebendo um *QR-Code* de ingresso por e-mail.

1.1 Apresentação

O objetivo principal desse trabalho é demonstrar diferentes utilizações do *QR-Code* em uma aplicação comercial que interaja com os usuários através de aplicativos móveis.

Devido ao crescente interesse dos brasileiros por *smartphones* e com a conectividade cada vez mais acessível, tornou-se comum o uso desses aparelhos no cotidiano. As empresas investem cada vez mais na produção de aplicativos para esse tipo de dispositivo visando conquistar essa nova gama de consumidores. Com o objetivo de atingir esse mercado emergente que o tema *QR-Code* foi escolhido.

A fraca utilização dessa ferramenta no Brasil é o que viabilizou o desenvolvimento desse projeto, que pretende demonstrar diferentes utilizações dessa ferramenta, realizando a integração de aplicativos celulares e aplicações web.

1.2 Justificativa

Com a realização deste projeto espera-se transmitir a viabilidade da utilização do *QR-Code* como mais uma ferramenta comercial.

O projeto traria benefícios tanto à empresa, que possuiria mais uma forma de vender seu produto, quanto ao cliente que possuiria uma forma confortável e prática de realizar sua compra. Visa-se também os benefícios não mensuráveis como possuir um cadastramento de clientes, podendo futuramente fazer um estudo mais detalhado sobre o mercado em si, e a fidelização dos clientes, já que devido ao cadastro, a empresa pode oferecer promoções especiais.

Do ponto de vista acadêmico, esse projeto possibilitou uma visão mais abrangente de desenvolvimento de projetos comerciais, proporcionando-nos a possibilidade de agir ativamente em todas as fases do desenvolvimento, participando em conjunto de sua análise, documentação, desenvolvimento, validações e testes. Este projeto também abriu as portas para o desenvolvimento para smartphones, sendo um incentivo a mais no aprendizado de uma nova forma de programação, com dificuldades diferenciadas e um mercado de trabalho emergente.

1.3 Objetivo Geral

Este projeto tem como objetivo utilizar o *QR-Code* de maneira diferenciada da normalmente utilizada no Brasil, integrando aplicativos móveis com uma aplicação web. Ambientado com o tema de cinema interativo, esse projeto visa aproximar essa nova ferramenta de aplicativos que seriam aceitos comercialmente pela população brasileira.

Visando esse objetivo resolvemos ambientar a aplicação em um sistema de venda de ingressos on-line, em que o cliente poderá realizar compras, cadastros e ganhar descontos utilizando um *smartphone*.

1.4 Objetivos Específicos

Os objetivos específicos deste projeto são:

- a) Investigar o mercado brasileiro que viabilizaria a aceitação da aplicação pela população brasileira;
- b) Desenvolver uma aplicação web que serve de intermediador entre as aplicações móveis e a base de dados;
- c) Desenvolver uma aplicação móvel na plataforma JME (*Java Micro Edition*) para compra de ingressos;
- d) Desenvolver duas aplicações móveis na plataforma Android uma para compra de ingressos e outra que seria utilizado pelos funcionários do cinema que realizaria a validação desses ingressos;

1.5 Estrutura e Organização

Este documento está dividido em onze tópicos:

1. Introdução, onde será apresentado e justificado o tema abrangido no trabalho, também serão demonstrados os objetivos, os produtos e os benefícios esperados;
2. Levantamento bibliográfico e estado da arte, abordará a fundamentação teórica que embasou o projeto e apresentará uma análise do mercado brasileiro que viabilizaria a implementação real do projeto;
3. Metodologia descreve o desenvolvimento do projeto, sendo dividido em dois tópicos: fundamentos e tecnologias;
4. Em Recursos de Hardware e Software é descrito os recursos necessários e a viabilidade para a execução do projeto.
5. No “Projeto a ser Desenvolvido e Resultados Iniciais do Mesmo”, será descrita a solução funcional do projeto, demonstrando através

de diagramas o sistema desenvolvido, aqui também será apresentado toda a documentação do projeto, como diagramas de classes, casos de uso e requisitos;

6. Estimativa por meio de pontos de casos de uso demonstra todos os passos intermediários para levantar as horas necessárias para o desenvolvimento do sistema;
7. Cronograma, apresenta as etapas necessárias para a conclusão do projeto e a quantidade de horas utilizadas para a conclusão do mesmo;
8. Procedimento de Testes e Validação descreve como os módulos do sistema foram testados.
9. Análise de Riscos, apresenta os riscos potenciais do projeto e o impacto dos mesmos no sucesso ou fracasso do projeto;
10. Conclusões, apresenta as conclusões tiradas pelos autores do projeto no desenvolvimento do mesmo. São enumerados possíveis trabalhos futuros devidamente justificados;
11. Referências Bibliográficas, descreve todas as referências utilizadas para o desenvolvimento do projeto.

Neste capítulo foram apresentados os objetivos gerais e específicos desse projeto, visando introduzir o leitor à estruturação e o tema do mesmo. No capítulo 2 serão apresentados os levantamentos bibliográficos e estado da arte abordando o histórico do QR-Code, explicando seu funcionamento e apresentando uma análise mercadológica do mesmo no país.

2 LEVANTAMENTO BIBLIOGRÁFICO E ESTADO DA ARTE

Neste capítulo serão abordados o histórico do código QR, demonstrados algumas utilizações no Brasil e uma breve pesquisa de mercado que possibilitaria a viabilização do projeto no mercado brasileiro.

2.1 Descrição do QR-Code

O Código de barras bidimensional, criado pela Denso Wave possui um amplo armazenamento de dados, como podemos analisar na tabela abaixo.

Tabela 1: Capacidade de armazenamento do QR-Code

Tipo de Caractere	Quantidade Máxima Permitida
Apenas número	7.089
Alfanumérico	4.296
Binário (8-bits)	2.953
Kanji/Kana ¹	1.817

Fonte: Denso Wave, 2012

Fazendo uma breve comparação podemos analisar a diferença de armazenamento entre o código de barras unidimensional e bidimensional, enquanto o código de barras unidimensional armazena cerca de 20 caracteres numéricos o bidimensional pode chegar a 7.089.

Além dessa há outras características que o tornam mais eficiente e uma escolha melhor do que o código de barras unidimensional, como por exemplo a capacidade de correção de erros, demonstrada da tabela a seguir.

1 Algarismos utilizados no alfabeto japonês

Tabela 2: Capacidade de correção de erros

Nível	Capacidade de correção
Nível L	7% das palavras podem ser restauradas
Nível M	15% das palavras podem ser restauradas
Nível Q	25 % das palavras podem ser restauradas
Nível H	30% das palavras podem ser restauradas

Fonte: Denso Wave, 2012

Essa característica permite que mesmo danificado o código ainda possa ser lido. Um máximo de 30% de “codewords²” pode ser restaurado. Em alguns casos, como o *QR-Code* utilizado pela empresa da Coca-Cola na Figura 2, o código é “danificado” propositalmente para criar formas, a imagem a seguir é um exemplo de como um código bidimensional pode ser manipulado e continuar funcional.



Figura 2: Código QR utilizado pela Coca-Cola.

Fonte: *Branding Magazine*, 2012.

Outra vantagem do *QR-Code* é que a área necessária para transcrevê-lo pode ser um décimo menor que de um código de barras

2 Um *codeword* é uma unidade que constrói a área de dados. No *QR-Code* um *codeword* é representado por 8 bits.

tradicional. A Figura 3 mostra a comparação de espaço entre um código de barras unidimensional e um *QR-Code*.



Figura 3: Comparação de tamanho entre código de barras 1D e 2D.

Fonte: Denso Wave, 2012.

Em casos onde é necessário um espaço de impressão ainda menor que o de um código de barras QR, pode-se ainda utilizar o micro QR-code. Um micro QR-code nada mais é que um código QR muito pequeno que se encaixa em aplicações que necessitam de um espaço menor e uma menor quantidade de dados, um exemplo de aplicação de micro QR-code é a impressão de identificadores em componentes eletrônicos como demonstrado na Figura 4.

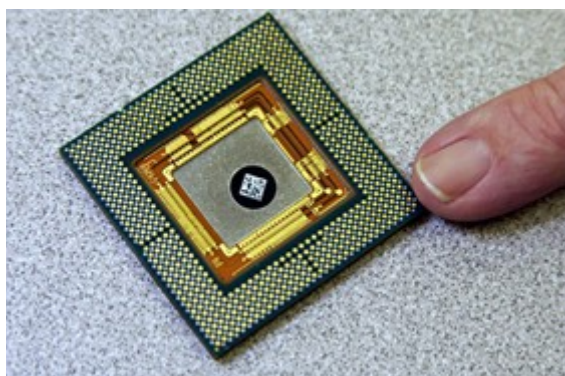


Figura 4: Identificação de microchip através de um micro QR-Code.

Fonte: Denso Wave, 2012.

A estrutura do código QR permite ainda que ele possa ser lido a partir de qualquer direção, pois ele possui um padrão de identificação de posição, a Figura 5 descreve melhor essa funcionalidade.

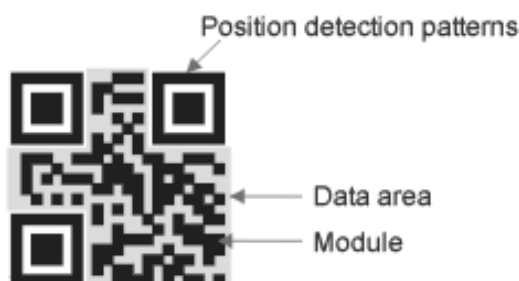


Figura 5: Estrutura do código QR.

Fonte: Denso Wave, 2012.

O código QR, foi utilizado inicialmente para usos industriais, mas com o avanço da tecnologia e com o mercado de *smartphones* em ascensão, ele foi se tornando popular na área de publicidade.

Atualmente existem vários softwares de decodificação para cada sistema operacional de *smarphones*, a seguir serão listados uma relação de alguns desses softwares para cada tipo de plataforma.

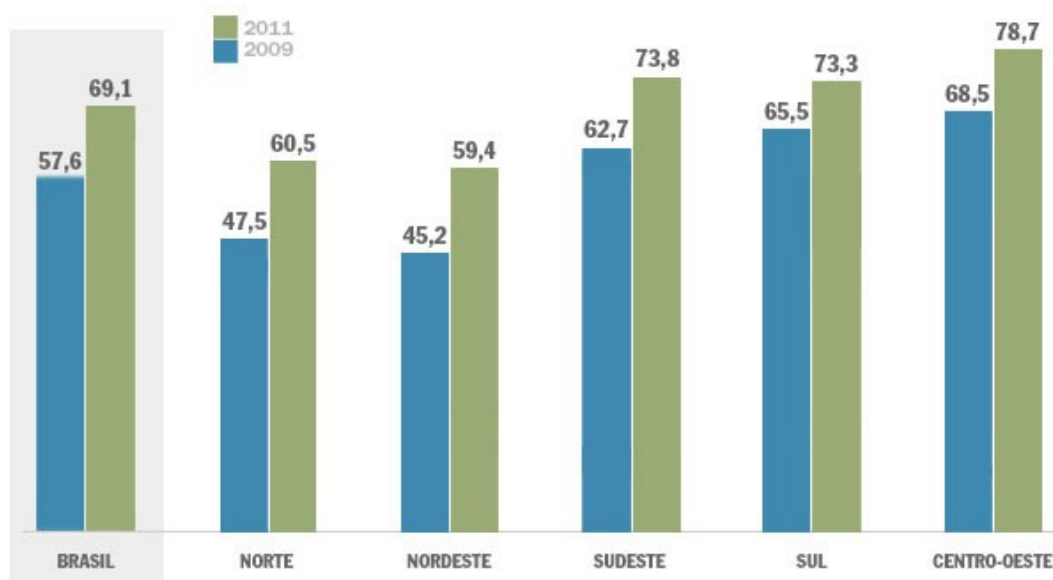
- a) Android: Barcode Scanner, Bee-Tag, I-Nigma
- b) Blackberry: Bee-Tag, I-Nigma
- c) iPhone: Bee-Tag, I-Nigma, NeoReader, QuickMark, Scan.
- d) Symbian: Bee-Tag, I-Nigma.
- e) Windows Mobile: Microsoft Tag, QuickMark, Esponse Qr Reader, QR Code Reader.

2.2 Análise Mercadológica Brasileira

Nos últimos anos o Brasil vem apresentando um crescimento significativo no número de celulares por habitante, é o que mostra o levantamento do IBGE (Instituto Brasileiro de Geografia e Estatística) referente ao ano de 2011 (Pesquisa Nacional por Amostra de Domicilio, 2011). Mais de 69% da população brasileira possui um celular de uso pessoal (pesquisa realizada com brasileiros acima de 10 anos de idade). O celular está entre os 5 bens duráveis com maior crescimento chegando a marca de 26,6% ficando atrás apenas de microcomputadores com e sem

acesso a internet. A Figura 6 ilustra a porcentagem de celulares pela população brasileira, dividindo também o crescimento por região.

Celular para uso pessoal, na população a partir de 10 anos (%)



Fonte: IBGE, Diretoria de Pesquisas, Coordenação de Trabalho e Rendimento, Pesquisa Nacional por Amostra de Domicílios 2009/2011.

Figura 6: Crescimento de celulares para uso pessoal de 2009-2011.

Fonte: Instituto Brasileiro de Geografia e Estatística, 2012.

Em relação ao acesso ao SMP (Serviço Móvel Pessoal) o Brasil teve 242,2 milhões de acesso em 2011 com um crescimento de 19,4% como mostra o levantamento anual da Anatel (Agência Nacional de Telecomunicações, 2012). 81,8% desses acessos foram realizados por celulares pré-pagos e 18,2% por pós-pagos. A Figura 7 mostra a evolução do SMP de 1998-2011 (Agência Nacional de Telecomunicações, 2012).

Uma das principais características do SMP é a rápida evolução das tecnologias associadas à prestação do serviço. Em 2010, houve a extinção do serviço de tecnologia analógica (tecnologia AMPS - *Advanced Mobile Phone System*) ativos na planta das operadoras do Serviço Móvel Pessoal. Em 2011, o mesmo aconteceu com os acessos da tecnologia digital TDMA (*Time Division Multiple Access*), ditos de segunda geração – 2G.

EVOLUÇÃO DOS ACESSOS DO SMP (em milhões)

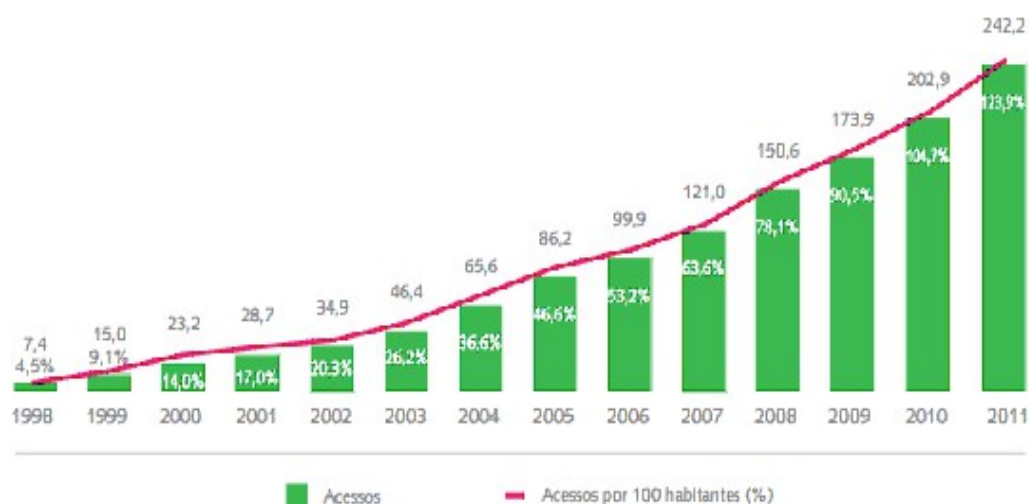


Figura 7: Evolução de acesso SMP de 1998 – 2011.

Fonte: Agência Nacional de Telecomunicações, 2012.

Em 2011 a tecnologia de banda larga móvel vem se mostrando em crescente ascensão é o que demonstra o gráfico da Figura 8.

ACESSOS DO SMP POR TECNOLOGIA (em %)



Figura 8: Acessos do SMP por tecnologia.

Fonte: Agência Nacional de Telecomunicações, 2012.

O estudo do IBGE também levantou o crescente acesso a internet pelos brasileiros, em 2011, o acesso (de qualquer lugar, não apenas em casa) à rede chega a 77,7 milhões de pessoas, que representam 46,5% dos habitantes acima dos 10 anos de idade. São 9,9 milhões de pessoas a mais do que o observado na pesquisa anterior, um crescimento de 4,9 pontos percentuais.

A proporção de usuários de internet cresceu em todas as regiões, e passou da metade da população em três delas: Sudeste (54,2%), Centro-Oeste (53,1%) e Sul (50,1%). E assim como no caso dos celulares, os jovens também são maioria no acesso à web: na faixa dos 15 aos 19 anos, o índice chega a 74,1%. O maior aumento, porém, foi verificado no grupo entre 30 a 39 anos: 9,3 pontos percentuais de 2009 para 2011, passando de 42% para 51,3%.

Segundo a pesquisa do Our Mobile Planet divulgado em maio desse ano, 14% da população brasileira possui um *smartphone* e 43% desses usuários acessam a internet através do celular diariamente (Google, 2012), e esse número tende a crescer, segundo um levantamento da IDC (*International Data Corporation*) foram vendidos 6,8 milhões de *smartphones* de janeiro a julho desse ano, um crescimento de 77% em comparação com o mesmo período do ano de 2011 (Mobilitytime, 2012).

Com o crescimento da venda de *smartphones* e do acesso da internet pelo celular o mercado fica mais favorável para sistemas como o apresentado nesse projeto, onde há um sistema web para clientes e administradores que provê informações aos aplicativos móveis. O usuário utiliza os aplicativos móveis para realizar operações diretamente pelo celular, facilitando e agilizando algumas operações.

2.3 Uso do QR-Code no Brasil

Atualmente o *QR-Code* é largamente utilizado como uma ferramenta de marketing, usado principalmente na promoção e divulgação de produtos e empresas.

Há empresas que apostam na utilização de uma forma diferenciada, como é o caso do Banco do Brasil que disponibilizou um aplicativo móvel que realiza leitura de boletos a partir de uma imagem QR. O aplicativo é gratuito e está disponibilizado para as plataformas Android e iPhone (Banco do Brasil, 2012).



Figura 10: QR-Code de divulgação da Sprite.

Fonte: QR-Code no Brasil, 2012.



Figura 11: Código QR com site da Marvel contendo clipe do filme.

Fonte: QR-Code no Brasil, 2012.

Neste capítulo foram abordados o histórico do QR-Code, explicando seu funcionamento e apresentando uma análise mercadológica do mesmo no país. No próximo capítulo será apresentado os fundamentos e tecnologias utilizadas para o desenvolvimento deste projeto.

3 METODOLOGIA

Abaixo serão apresentados os fundamentos e as tecnologias utilizadas para o desenvolvimento e conclusão deste projeto.

3.1 Fundamentos

Para o melhor andamento e desenvolvimento do projeto foi acordado dividir o projeto em quatro etapas principais: análise e documentação, modelagem de banco de dados, codificação e testes. Essas etapas estão descritas detalhadamente nos tópicos a seguir.

3.1.1 Análise e Documentação

Embora inicialmente tenha sido feito uma breve descrição da funcionalidade do sistema no documento de pré-projeto, foi realizado um levantamento de requisitos para melhor análise e descrição das mesmas, descrevendo os requisitos funcionais e não funcionais dos projetos.

Com o levantamento de requisitos concluído pode-se dar andamento nos diagramas de casos de uso, classes e sequência que documentam os sistemas realizados. Essa etapa foi realizada utilizando a linguagem de modelagem UML (*Unified Modeling Language*) na versão 2.0 devido a sua ampla utilização no mercado de trabalho tanto nacional como internacional. O UML independe de linguagem de programação, facilita compreensão das funcionalidades dos sistemas e documenta de maneira clara e objetiva o sistema a ser criado.

3.1.2 Modelagem de Banco de dados

Tendo a documentação disponível foi dado início a etapa de modelagem de banco de dados.

Com o intuito de evitar redundâncias, dados conflitantes e para garantir um melhor desempenho e manutenção no banco de dados é que foi decidido a criação de um DER (Diagrama de Entidade e Relacionamento), para auxiliar o refinamento e implementação do banco de dados. A implementação do banco de dados foi realizada com a criação de um *script* em linguagem SQL (*Structured Query Language*), facilitando possíveis futuras restaurações.

3.1.3 Codificação

A implementação do banco de dados possibilitou o início da codificação efetiva dos sistemas web e móvel. Os sistemas foram codificados simultaneamente, ficando cada autor responsável por um sistema.

O desenvolvimento do projeto web foi realizado no modelo MVC de padrão de projetos (*Model View Controller*). Essa decisão foi tomada devido a facilidade de manutenção do código.

Para o sistema móvel foi utilizado a plataforma Android 2.2.3 Gingerbread devido sua grande incidência nos dispositivos atualmente em circulação. Foi utilizado o padrão de projeto *singleton* para as classes Cliente e Filme pois garante a existência de apenas uma instância, facilitando assim o acesso as informações contidas nessas classes.

3.1.4 Testes

Os testes foram realizados pelos autores do projeto, visando abranger os mais variados imprevistos, com o intuito de correção de erros ou tratamento de mensagens quando necessário. Os testes menores foram realizados durante a codificação dos sistemas, conforme as funcionalidades foram sendo concluídas, após a finalização dos sistemas foram realizados

testes severos no projeto para analisar a integração do sistema web com os aplicativos móveis criados. Os testes finais foram realizados visualizando possíveis cenários e executando roteiros de uso para a análise dos resultados obtidos, se necessário efetuando correções.

3.2 Tecnologias

Os sistemas foram desenvolvidos na casa dos autores, ambientado em local arejado com energia elétrica e internet banda larga 24 horas mantido para uso pessoal dos autores.

A linguagem de programação escolhida para o desenvolvimento dos sistemas foi o Java, pois é livre e possui boas IDEs (*Integrated Development Environment*) de desenvolvimento também livres como Netbeans e Eclipse, é Orientado a Objetos possuindo reuso de código, possui comunidade de desenvolvedores ativa e possui portabilidade devido a JVM (*Java Virtual Machine*).

Após pesquisarmos as API's (*Application Programming Interface*) de desenvolvimento disponíveis gratuitamente para codificação e decodificação em código QR, foi escolhido a biblioteca ZXing (*Zebra Crossing*), devido a sua comunidade de desenvolvedores, ampla documentação existente, popularidade e facilidade de uso. Essa ferramenta foi utilizada no desenvolvimento de todos os sistemas (móveis e web).

Inicialmente o sistema web foi desenvolvido utilizando o *framework* JSF (*Java Server Faces*), *framework* de mapeamento de objeto-relacional Hibernate, com a suíte de componentes para JSF Primefaces e servidor de aplicação Tomcat. Posteriormente devido a conflitos constantes com o *framework* de mapeamento e componentes do JSF foi decidido a remoção do mapeamento pelo Hibernate e realizado o mapeamento manualmente pelo código.

Os aplicativos móveis foram desenvolvidos utilizando o Android-sdk e o Netbeans IDE com o *plugin* de desenvolvimento Android. O Android-sdk possui as ferramentas necessárias para a criação de dispositivos virtuais

que simulam celulares reais e permitem fácil distribuição e depuração, permite também a integração com um dispositivo real via porta USB(*Universal Serial BUS*) desde que o *driver*³ do fabricante do mesmo esteja devidamente instalado na máquina.

Neste capítulo foram apresentados os fundamentos e tecnologias utilizadas para o desenvolvimento deste projeto. No próximo capítulo serão abordados os recursos necessários para o seu desenvolvimento.

3 Software que realiza a comunicação entre o sistema e o dispositivo.

4 RECURSOS DE HARDWARE E SOFTWARE

Neste capítulo serão apresentados os recursos utilizados para o desenvolvimento deste projeto, sendo separado em três tópicos, recursos de hardware, recursos de software e viabilidade.

4.1 Recursos de Hardware

Para a implementação deste projeto, serão necessários no mínimo:

- a) Uma máquina para desenvolvimento contendo no mínimo um processador 800 Mhz (*Megahertz*) Intel Pentium III ou equivalente; 512 MB (*Megabyte*) de memória RAM (*Random Access Memory*) e 20 GB (*Gigabyte*) de armazenamento.
- b) Uma máquina para ser utilizada como servidor contendo no mínimo um processador Pentium 4 ou equivalente, 2 GB de memória RAM e 80GB de armazenamento.
- c) Um *smartphone* com sistema operacional Android.

Os custos de equipamentos podem variar de acordo com o mercado.

Os recursos efetivamente utilizados para a realização do projeto foram:

- a) Notebook Microboard, processador intel i5, 8GB de memória RAM e 500GB de armazenamento e possuindo Sistema Operacional Windows 7 Home premium 64-bit.
- b) Notebook Microboard, processador interl i5, 16GB de memória RAM e 640GB de armazenamento e possuindo Sistema Operacional Windows 7 Professional.
- c) 2 celulares LG Optimus-pro C660H.

Esses equipamentos foram adquiridos para uso pessoal dos autores não necessitando a realização de aquisições voltadas exclusivamente ao projeto.

4.2 Recursos de Software

Os softwares escolhidos para a execução do projeto são todos livres e possuem custo zero. Os tópicos a seguir descrevem os softwares utilizados.

- a) LibreOffice: ferramenta de escritório livre.
- b) JDK 1.7: *Java Development Kit* distribuído gratuitamente pela Oracle.
- c) PostgreSQL 9.1: Utiliza uma licença de uso BSD (*Berkeley Software Distribution*) e pode livremente ser incorporado por aplicações pessoais e/ou comerciais sem nenhum custo ao desenvolvedor.
- d) PgAdminIII: Ferramenta gratuita de administração para banco de dados PostgreSQL.
- e) Netbeans 7.1.1: IDE de uso e distribuição gratuitas, inclusive no ambiente empresarial.
- f) Plugin NBAndroid: *Plugin* para desenvolvimento de aplicativos Android.
- g) Plugin Netbeans UML: *Plugin* para diagramação UML.
- h) JSF: Framework mantido pela Oracle.
- i) Primefaces 2.2.1: Ferramenta para componentes JSF.
- j) Tomcat: Servidor de aplicação gratuito mantido pelo Apache.
- k) ZXing: Biblioteca *open-source* de codificação e decodificação de códigos QR.

Os Sistemas Operacionais utilizados para o desenvolvimento do projeto foram : Windows 7 Home Premium e Windows 7 Professional, mas não há restrições para a utilização de distribuições Linux como Debian, Ubuntu, Red Hat, Suse, entre outros.

4.3 Viabilidade

Nos tópicos abaixo serão abordados as viabilidades técnicas e viabilidades econômicas da implantação do projeto.

4.3.1 Viabilidade Técnica

A implementação desse projeto automatizará o processo de vendas de ingressos e dispensará a figura do vendedor. O cliente poderá realizar a compra de ingressos de maneira mais cômoda, simplesmente acessando a internet através do celular.

A linguagem Java, proposta e utilizada para o desenvolvimento do projeto, é uma das mais utilizadas atualmente, garantindo mão de obra com conhecimento técnico necessário para o desenvolvimento do projeto no prazo estipulado. A biblioteca ZXing possui comunidade ativa e ampla documentação no próprio site, facilitando assim a sua utilização na implementação do sistema. Os *smartphones* com sistema Android estão cada vez mais populares e tendo cada vez mais aceitação no mercado brasileiro, garantindo possíveis usuários e clientes para o sistema.

4.3.2 Viabilidade Econômica

Considerando a necessidade de compra de todos os itens de hardware mínimos, citados anteriormente, para o desenvolvimento do projeto o custo de aquisição de hardware ficaria em torno de 2300 reais.

A tabela abaixo relata os custos aproximados para o desenvolvimento do sistema, dividida em colunas contendo a descrição da atividade desenvolvida, horas necessárias para a conclusão da atividade e o custo dessa mão de obra.

Tabela 3: Custo de desenvolvimento

Atividade	Horas	Preço/Hora
Análise/Documentação	16	R\$ 35,00
Modelagem de Banco de Dados	16	R\$ 35,00
Codificação Aplicação Web	200	R\$ 25,00
Codificação JME	70	R\$ 25,00
Codificação Android	120	R\$ 25,00
Testes	16	R\$ 15,00
Total	438	R\$ 11.110,00

Não há custos para aquisição de softwares para o desenvolvimento do projeto pois os softwares propostos e utilizados são todos gratuitos.

4.3.3 Conclusão

Analisando as viabilidades técnicas e econômicas, podemos perceber que é viável a implantação deste projeto pois os custos somados não chegam a totalizar 15.000 reais. No mercado de cinema é um investimento baixo, acessível até a empresas de pequeno porte, a implementação do projeto representará em um diferencial para a empresa no mercado brasileiro atual. A tecnologia proposta para a implementação é uma das mais utilizadas atualmente no mercado de software, consolidada no mercado internacional, garantindo mão de obra qualificada e possíveis futuras manutenções no sistema.

Neste capítulo foram abordados os recursos utilizados para o desenvolvimento deste projeto. No próximo capítulo serão apresentados os resultados do projeto desenvolvido.

5 PROJETO DESENVOLVIDO E RESULTADOS

Este projeto está dividido em três módulos principais: Sistema Web, Aplicativo Móvel Cliente e Aplicativo Móvel Cinema. O diagrama de blocos na Figura 12 demonstra de maneira sucinta a integração dos três sistemas.

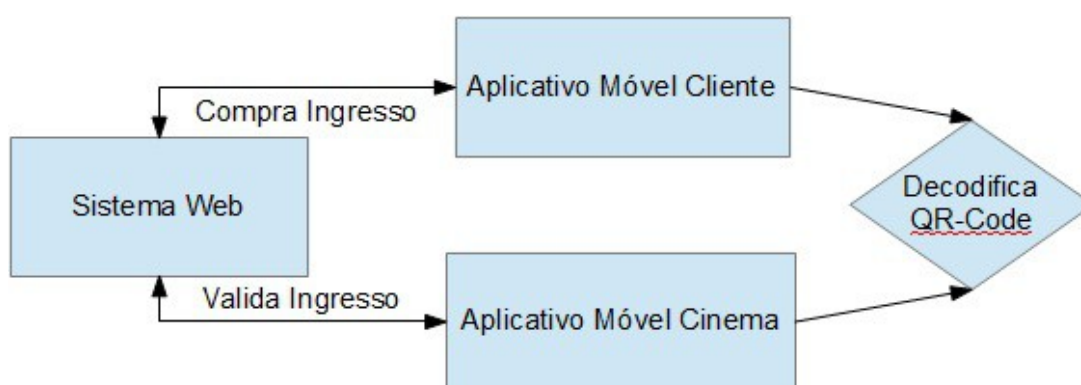


Figura 12: Diagrama de Blocos do Projeto

Os tópicos a seguir explicarão o funcionamento de cada sistema separadamente.

5.1 Sistema Web

O sistema web foi desenvolvido na linguagem Java, em um notebook contendo sistema operacional Windows 7 Professional, com a IDE Netbeans 7.1.1, com servidor de aplicação TomCat.

A Figura 13 mostra a tela inicial do Sistema Web, através dessa tela é possível se logar como Administrador ou Cliente, caso o cliente ainda não possua cadastro ele pode se cadastrar clicando no botão “Cadastre-se” no canto superior esquerdo da tela. O menu Cinema gerencia os ingressos e créditos do cliente quando logado.

A Figura 14 mostra a tela de auto cadastro de cliente, possuindo os campos de preenchimento Nome, CPF (Cadastro de Pessoas Físicas), E-mail, Senha e Confirmação de senha obrigatórios. Inserindo os campos

corretamente e clicando em cadastrar o sistema realizará o login automaticamente, voltará a tela inicial e apresentará a seguinte mensagem “Cadastro efetuado com sucesso”. Após cadastrado, o E-mail do cliente será o login de acesso às demais telas do sistema.

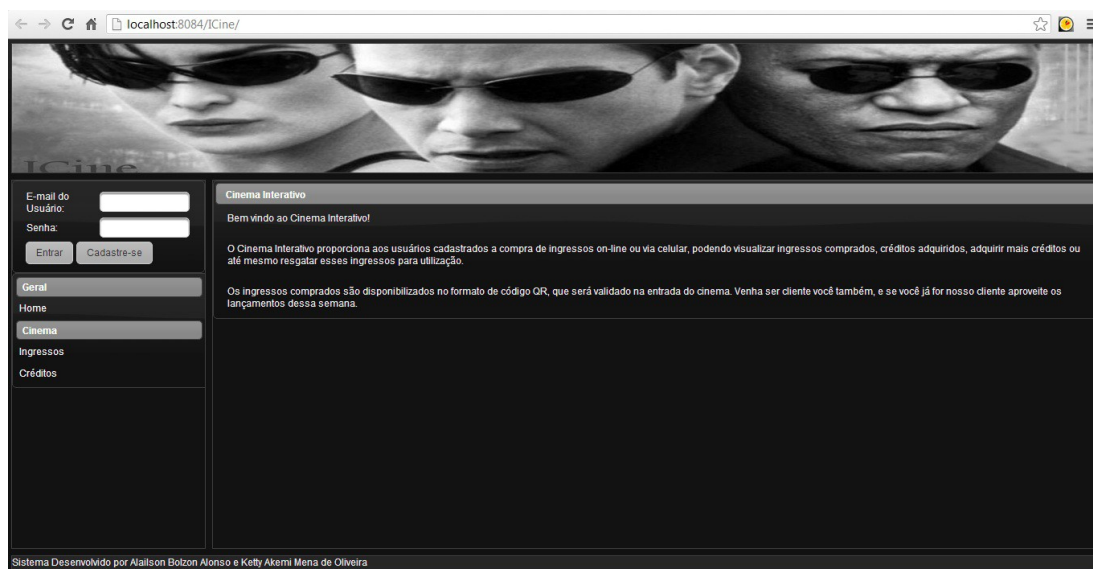


Figura 13: Tela Inicial do Sistema Web de Compra de Ingressos

The image shows a registration form titled 'Cadastro' with a sub-header 'Cliente'. The form contains several input fields: 'Nome*' (required), 'CPF*' (required), 'E-mail*' (required), 'Senha*' (required), 'Confirme a senha*' (required), 'Data de Nascimento' (required), and 'Telefone' (optional). Each field is represented by a white rectangular input box. At the bottom of the form, there is a 'Cadastrar' button.

Figura 14: Tela de Auto Cadastro do Sistema Web

A tela de gerenciamento de créditos, mostrado na Figura 15, contém a mensagem dos créditos possuídos pelo cliente e um campo para digitar o valor do crédito, por não se tratar de um sistema real a tela de compra de créditos não possui função de cobrança, o valor digitado é inserido no banco de dados do cliente apenas para exemplificar a funcionalidade de compra de créditos.

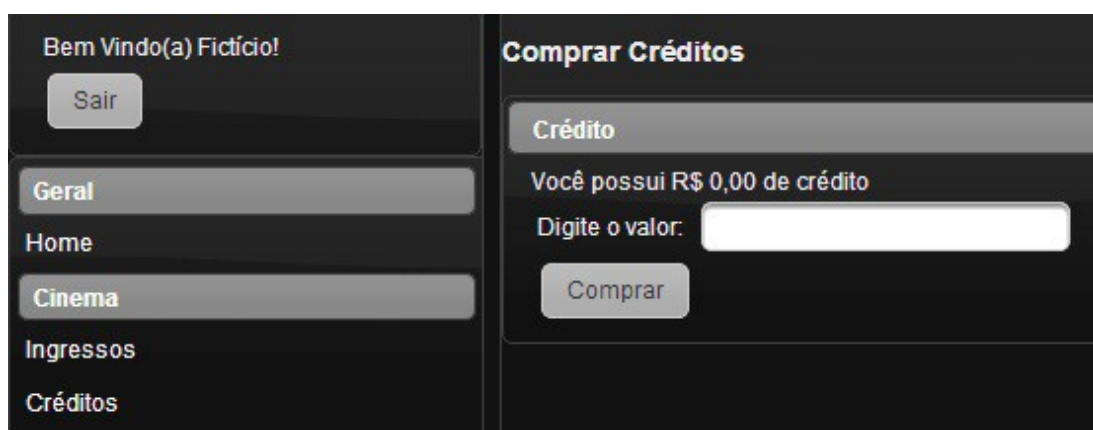


Figura 15: Tela de Gerenciamento de Créditos do Sistema Web

A Figura 16 mostra a tela de gerenciamento de ingressos do cliente "Fictício". Como o cliente acabou de ser cadastrado ele ainda não possui ingressos como mostra a mensagem. Ao clicar no link "Comprar Ingressos" o cliente será redirecionado para a tela de compra de ingressos, e caso possua créditos poderá efetuar a compra.

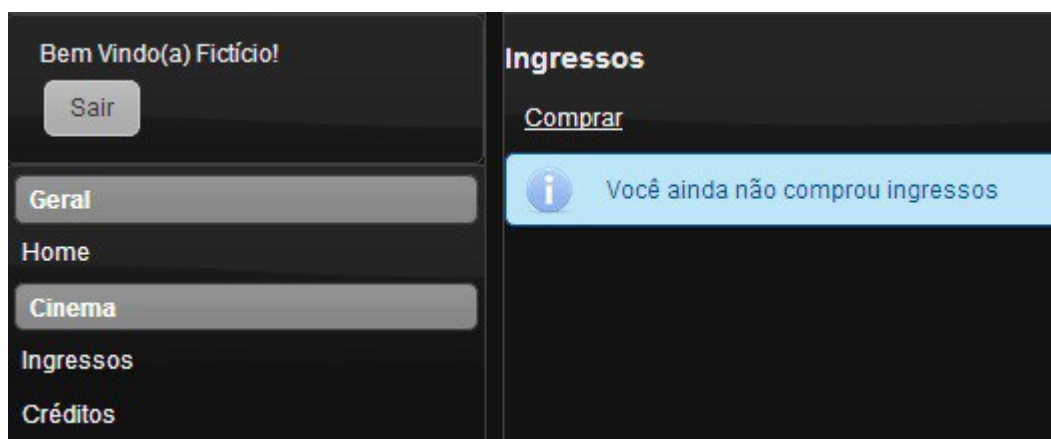


Figura 16: Tela de Gerenciamento de Ingressos Sistema Web

O Sistema Web possui também uma área para administradores do sistema, que realizará o gerenciamento dos cadastros de Cliente, Filme, Gênero, Horário e Sala. Caso seja necessário é possível realizar a validação de um ingresso.

A Figura 17 mostra o menu gerado para o usuário administrador do sistema.

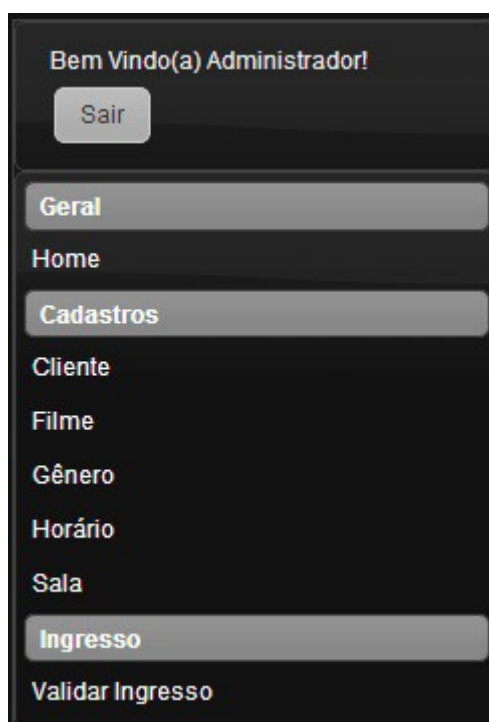
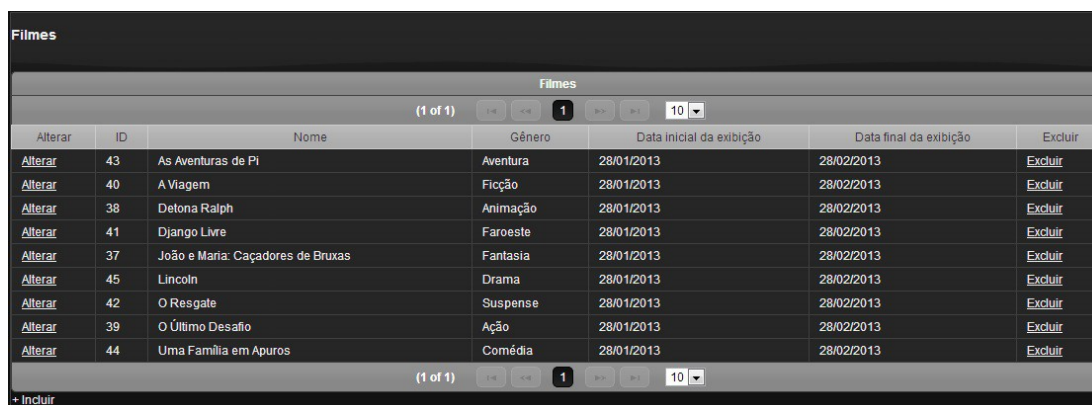


Figura 17: Tela de Menu para o Administrador do Sistema Web

As telas de visualização são muito semelhantes entre si, todas elas possuem uma tabela com as informações dos dados contidas no banco de dados com links para a alteração, inclusão e exclusão dos cadastros. A Figura 18 mostra a tela de visualização de filmes, contendo filmes já previamente cadastrados.



Alterar	ID	Nome	Gênero	Data inicial da exibição	Data final da exibição	Excluir
Alterar	43	As Aventuras de Pi	Aventura	28/01/2013	28/02/2013	Excluir
Alterar	40	A Viagem	Ficção	28/01/2013	28/02/2013	Excluir
Alterar	38	Detona Ralph	Animação	28/01/2013	28/02/2013	Excluir
Alterar	41	Django Livre	Faroeste	28/01/2013	28/02/2013	Excluir
Alterar	37	João e Maria: Caçadores de Bruxas	Fantasia	28/01/2013	28/02/2013	Excluir
Alterar	45	Lincoln	Drama	28/01/2013	28/02/2013	Excluir
Alterar	42	O Resgate	Suspense	28/01/2013	28/02/2013	Excluir
Alterar	39	O Último Desafio	Ação	28/01/2013	28/02/2013	Excluir
Alterar	44	Uma Família em Apuros	Comédia	28/01/2013	28/02/2013	Excluir

Figura 18: Tela de Gerenciamento de Filmes Sistema Web

5.2 Aplicativo Móvel Cliente

O aplicativo móvel utilizado pelo cliente foi desenvolvido em linguagem Java, na IDE Netbeans 7.1.1 contendo *plugin* para desenvolvimento para sistema operacional Android, em um notebook contendo sistema operacional Windows 7 Home Premium. Este aplicativo foi desenvolvido para sistema operacional Android 2.3.3 GingerBread e testado em celulares LG Optimus Pro.

Após instalado, o aplicativo móvel de cliente apresentará um ícone representando um ingresso com o nome ICine que ao ser clicado inicializará o aplicativo. A Figura 19 mostra a tela do celular com o ícone do aplicativo.

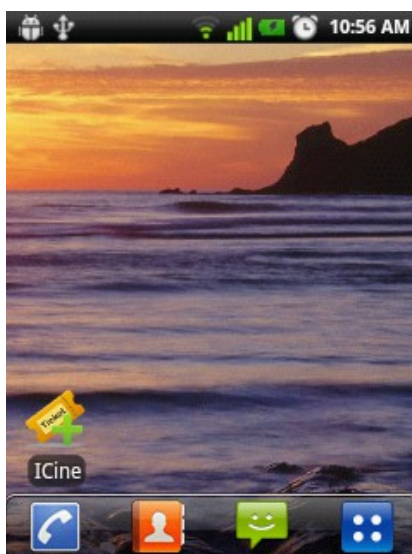


Figura 19: Tela do celular com o ícone do aplicativo móvel cliente

Ao iniciar, o aplicativo apresenta uma tela de login para o cliente, o login e senha a ser digitados são os mesmos do acesso ao Sistema Web. As Figuras 20 demonstra a tela de login vazia e preenchida respectivamente.

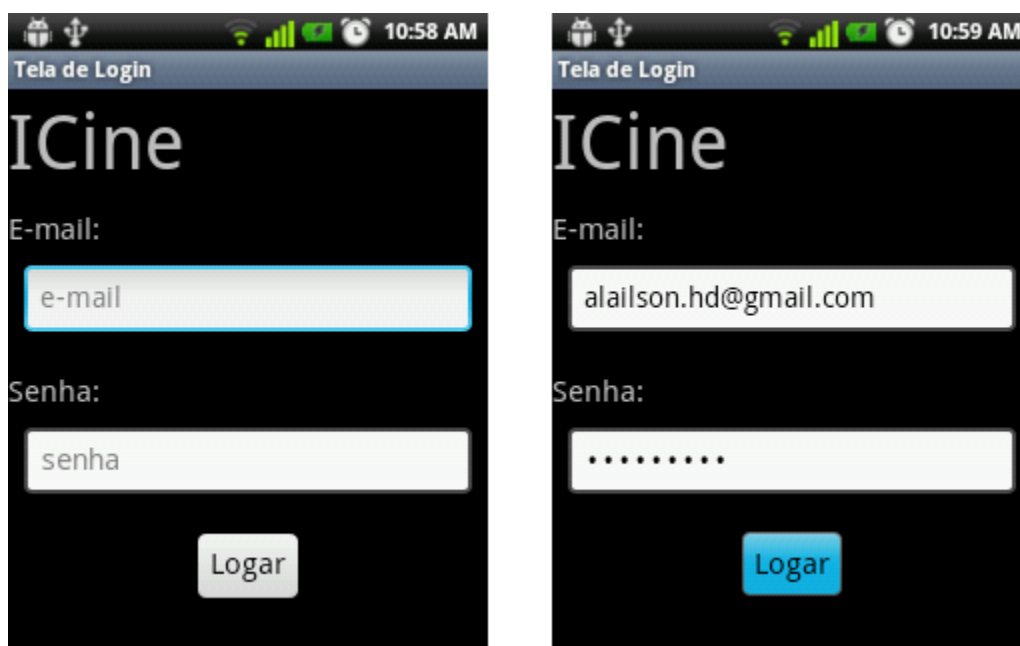


Figura 20: Telas de login do Aplicativo Móvel Cliente

Ao realizar o login corretamente o aplicativo aciona a câmera do celular para fotografar o QR-Code do Filme que o cliente quer assistir, tela representada na Figura 21 .



Figura 21: Tela da Câmera do Aplicativo Móvel Cliente

Ao digitalizar a imagem o celular decodificará o código e enviará as informações para o servidor web, o servidor por sua vez realizará as devidas validações e efetuará a compra do ingresso retornando uma mensagem de sucesso e enviando um e-mail com uma imagem QR referente ao ingresso comprado.

5.3 Aplicativo Móvel Cinema

O aplicativo móvel utilizado para validação de ingressos foi desenvolvido em linguagem Java, na IDE Netbeans 7.1.1 contendo *plugin* para desenvolvimento para sistema operacional Android, em um notebook contendo sistema operacional Windows 7 Home Premium. Este aplicativo foi desenvolvido para sistema operacional Android 2.3.3 GingerBread e testado em celulares LG Optimus Pro.

Após instalado, o aplicativo móvel utilizado para validação de ingressos apresentará um ícone representando um rolo de filme com o nome AdmICine que ao ser clicado inicializará o aplicativo.

A Figura 22 mostra a tela do celular com o ícone do aplicativo.

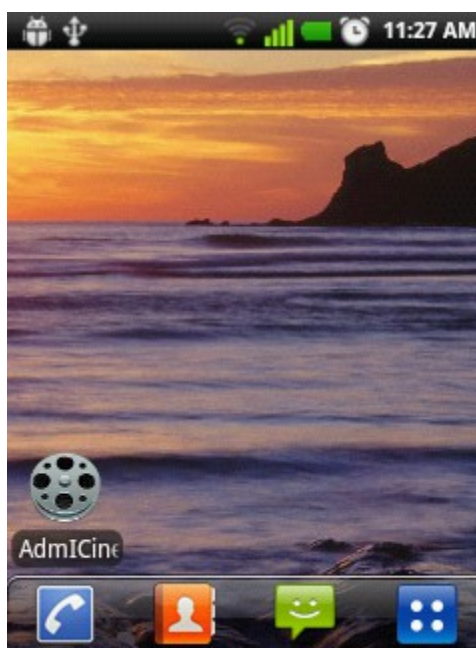


Figura 22: Tela do celular com o ícone do aplicativo móvel do cinema

Ao iniciar o aplicativo aparecerá uma tela com 2 ícones, um simbolizando a lente de uma câmera e outro simbolizando um teclado, a Figura 23 mostra essa tela.

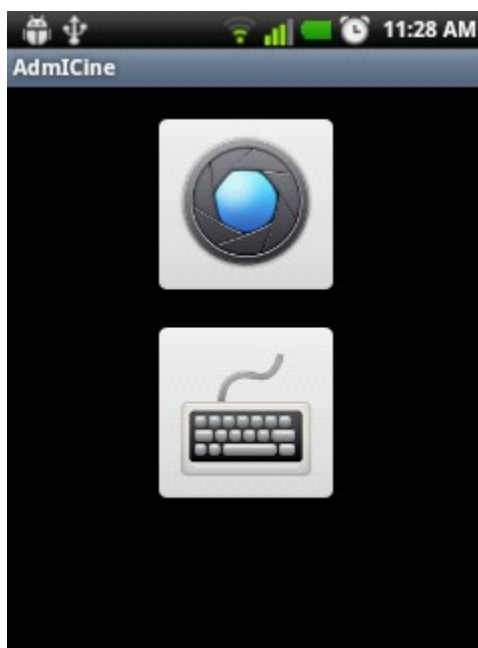


Figura 23: Tela inicial do aplicativo móvel do cinema

A Figura 24 mostra a tela iniciada ao clicar no ícone superior. A câmera do celular é acionada para que o ingresso possa ser decodificado e validado pelo servidor web, o servidor validará o ingresso e retornará uma mensagem que poderá ser de sucesso ou, caso o *QR-Code* seja inválido, de alerta.



Figura 24: Tela da Câmera do Aplicativo Móvel do Cinema

Caso a imagem esteja muito ruim de ser digitalizada ou o cliente não possuir a imagem em mãos o funcionário tem a opção de validar o ingresso digitando o código diretamente em um campo de texto, a Figura 25 demonstra essa funcionalidade que é acessada pelo segundo ícone da tela inicial.

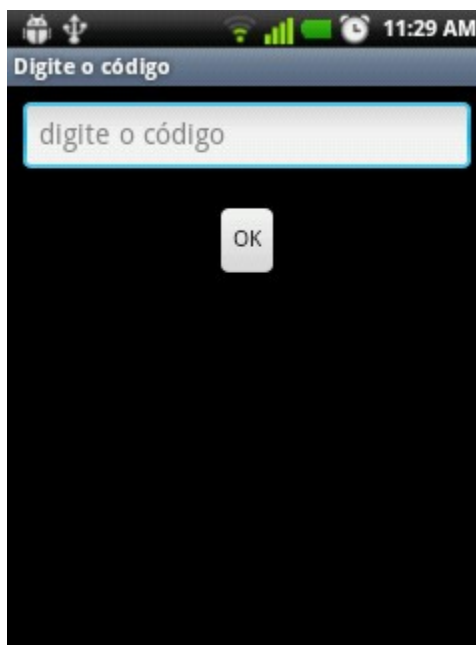


Figura 25: Validação do código através de campo texto

A validação do código ocorre da mesma forma que a demonstrada anteriormente, retornando ao funcionário uma mensagem de sucesso ou alerta.

5.4 Projeto de Software

Nos tópicos a seguir serão apresentados a análise orientada a objetos contendo levantamento de requisitos, diagrama de casos de uso, diagrama de classes e diagrama de sequência que embasaram o desenvolvimento e codificação dos sistemas.

5.4.1 Levantamento de Requisitos

O projeto pretende implantar um novo sistema para permitir que os clientes realizem compras de ingressos utilizando um telefone celular (ou *smartphone*) com acesso à internet. Para uma rápida transferência de informações o sistema se utilizará do *QR-Code* como uma ferramenta.

5.4.1.1 Requisitos Funcionais

Serão necessários 3 programas distintos: um sistema web, uma aplicação móvel para o cliente e uma aplicação móvel do cinema.

O sistema web será dividido em duas áreas de atuação: área do cliente e área do administrador do sistema.

Na área de administrador do cinema os responsáveis pelo cinema poderão gerenciar (incluir, alterar e excluir) as informações de clientes, filmes, salas e horários de exibição. Ainda na área de administrador do sistema os responsáveis poderão validar ingressos.

Para ter acesso a área cliente do sistema, o cliente deve primeiro fazer um cadastro no sistema web e registrar suas informações. Feito isso o cliente pode, realizar a compra de créditos e ingressos.

A aplicação móvel do cliente será responsável por realizar a compra de ingressos através da digitalização de uma imagem *QR-Code*. Realizada a compra uma imagem *QR-Code* de ingresso será enviada para o e-mail do cliente.

A aplicação móvel do cinema realizará as validações dos *QR-Codes* de ingressos apresentados pelos clientes na entrada do cinema. O funcionário do cinema digitalizará o ingresso com a câmera ou digitará o código manualmente para liberar ou não a entrada para a sala do filme.

5.4.1.2 Requisitos Não-funcionais:

O sistema web do cinema será executado em um servidor e estará disponível na internet. O tempo de resposta máximo permitido pelo sistema web é de 30 segundos. Esse sistema manterá um banco de dados onde

será armazenadas as informações dos clientes (dados pessoais e de contato), filmes (duração, gênero, ano, data de exibição), salas (número da sala, número de poltronas), horários (sala, filme, horário de exibição) entre outros. Através desse sistema também será possível realizar as validações de ingressos. Essas informações só poderão ser realizadas se o usuário administrador estiver logado. O sistema web também servirá de servidor para as validações necessitadas pelos aplicativos móveis.

Na área de acesso de cliente do sistema web, será disponibilizadas as funções de compra de créditos e ingressos, que estarão disponíveis somente com a identificação do mesmo.

O sistema web provê as seguintes funcionalidades aos aplicativos móveis:

- a) envio de e-mail com anexo de imagem *QR-Code* com extensão *PNG(Portable Network Graphics)*;
- b) Validações de compra de ingressos
 - Verifica se o filme ainda está em cartaz;
 - Verifica se ainda possuem poltronas disponíveis;
 - Verifica se os créditos do cliente são suficientes;
 - Atualiza o campo de ingressos vendidos na tabela do cliente no banco de dados;
 - Verifica se os ingressos vendidos no banco de dados é múltiplo de 10, se for, disponibiliza um ingresso promocional;
- c) Validação de ingresso:
 - Verifica se a data e horário do filme está de acordo;
 - Verifica se o ingresso já foi utilizado;
 - Atualiza o status do ingresso para utilizado;

Para que os aplicativos móveis funcionem corretamente é necessário que os mesmos sejam instalados em *smartphones* contendo

sistema operacional Android 2.3.3 com câmera fotográfica embutida. O tempo de resposta máximo permitido pelos aplicativos é de 9 segundos.

Com esse aplicativo instalado o cliente poderá realizar a compra de ingressos através da digitalização de uma imagem *QR-Code* que representa a exibição de um filme em um determinado horário. A aplicação móvel se comunicará com o sistema web e enviará as informações da imagem decodificada. O sistema web, por sua vez, utilizando as informações de login verifica se o cliente possui créditos suficientes e, em caso afirmativo, realiza a compra do ingresso para o cliente especificado.

O cliente apresentará a imagem recebida na entrada do cinema ao funcionário, que por sua vez, se utilizando do aplicativo móvel do cinema digitalizará e decodificará a imagem para que a validação seja feita pelo sistema web. O sistema web recebe e analisa essas informações e libera ou não a entrada para a sala do filme.

5.4.2 Diagramas de Casos de Uso

A seguir serão apresentados todos os diagrama de casos do projeto. Os casos de uso serão divididos por sistema e ator, sendo apresentado primeiramente um diagrama geral de todos os requisitos e logo em seguida cada requisito será apresentado num caso de uso próprio.

5.4.2.1 Casos de Uso do Sistema *Web* - Cliente

O diagrama de caso de uso da Figura 26 demonstra todas as requisitos do sistema web que serão utilizados pelo cliente. Este diagrama contém todas as ações que o ator poderá realizar, para mais detalhes sobre os casos de uso vide Apêndice A – Detalhamento dos casos de uso.

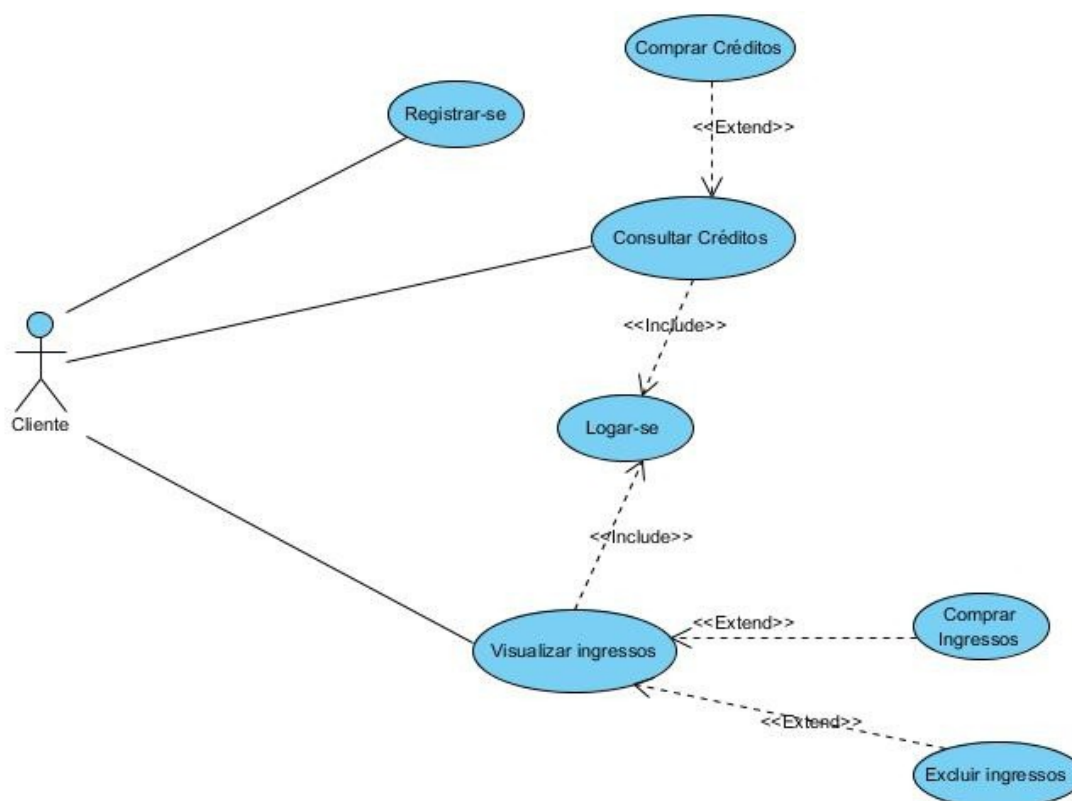


Figura 26: Casos de Uso Geral do Sistema Web pelo cliente

Caso de uso: Caso de uso geral do sistema Web acessado pelo Cliente.

Descrição: Esse caso de uso permite ver todas as ações que um cliente pode realizar. Essas requisições são: registrar-se, logar-se, consultar crédito, comprar crédito, visualizar ingresso, comprar ingresso e excluir ingresso.

5.4.2.2 Casos de Uso do Sistema Web - Administrador

O diagrama de caso de uso da Figura 27 demonstra todas as requisitos do sistema web que serão utilizados pelo administrador do sistema, ou um funcionário do cinema qualificado. Nessa área de atuação o ator poderá gerenciar dados importantes do banco de dados. Este diagrama contém as ações principais que o ator poderá realizar, para mais detalhes vide Apêndice A – Detalhamento dos casos de uso.

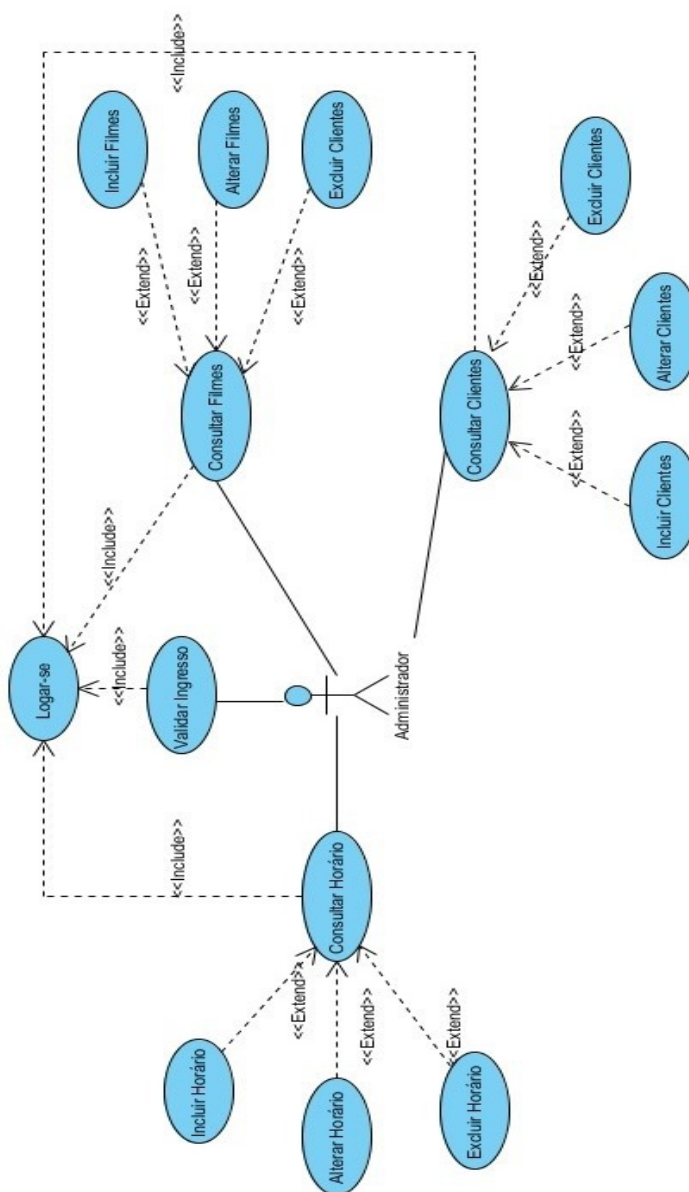


Figura 27: Caso de Uso Geral do Sistema Web Acessado Pelo Administrador

Caso de uso: Caso de uso geral do sistema Web acessado pelo Administrador - Figura 27.

Descrição: Esse caso de uso permite ver todas as ações que o administrador pode realizar. Essas requisições são: logar-se, gerenciar cliente (inserir, alterar e excluir), gerenciar filme (inserir, alterar e excluir), gerenciar horário (inserir, alterar e excluir) e validar ingresso.

5.4.2.3 Casos de Uso do Aplicativo Móvel Cliente

O diagrama de caso de uso da Figura 28 demonstra todos os requisitos do aplicativo celular que serão utilizados pelo cliente do sistema. Este diagrama contém todas as ações que o ator poderá realizar, para mais detalhes vide Apêndice A – Detalhamento dos Casos de Uso.

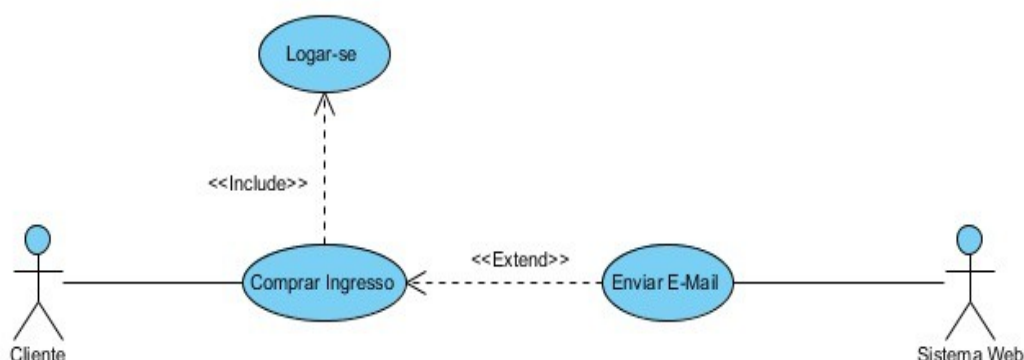


Figura 28: Caso de Uso Aplicativo Móvel Cliente

Caso de uso: Caso de uso geral do Aplicativo Móvel Cliente – Figura 28.

Descrição: Esse caso de uso permite ver todas as ações que o cliente pode realizar. Essas requisições são: logar-se e comprar ingresso.

5.4.2.4 Casos de Uso do Aplicativo Móvel Cinema

O diagrama de caso de uso representado na Figura 29 demonstra todos os requisitos do aplicativo celular que serão utilizados pelo funcionário do cinema. Este diagrama contém todas as ações que o ator poderá realizar. Essas ações serão divididas uma a uma e explicadas mais detalhadamente nos casos de uso seguintes, para maiores detalhes vide Apêndice A – Detalhamento dos Casos de Uso.

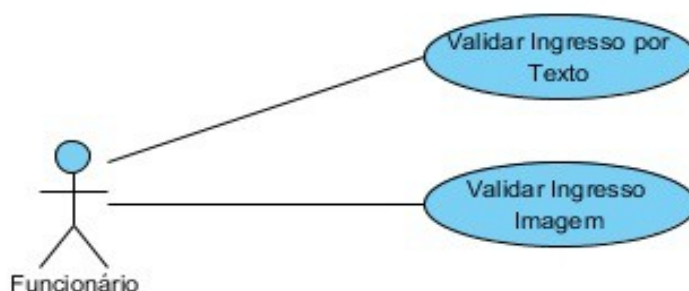


Figura 29: Caso de Uso Aplicativo Móvel Cinema

Caso de uso: Caso de uso geral do Aplicativo Móvel Cinema – Figura 29.

Descrição: Esse caso de uso permite ver todas as ações que o funcionário pode realizar. Essas requisições são: Validar Ingresso por texto ou validar ingresso por imagem.

5.4.3 Diagramas de Classes

A seguir serão apresentados todos os diagrama de classe do projeto. Os diagramas estão separados por sistema, sendo o primeiro referente ao sistema web o segundo do aplicativo móvel cliente e o terceiro do aplicativo móvel cinema.

5.4.3.1 Diagrama de Classe do Sistema Web

Como o diagrama de classes do sistema web é muito muito extenso, ele foi dividido em alguns documentos, o diagrama a da Figura 30 representa as relações do diagrama completo do sistema web.

As classes omitidas podem ser visualizadas no Apêndice B – Detalhamento dos Diagrama de Classes.

A Figura 31 representa as classes de *servlet*, responsáveis por suprir os aplicativos móveis.

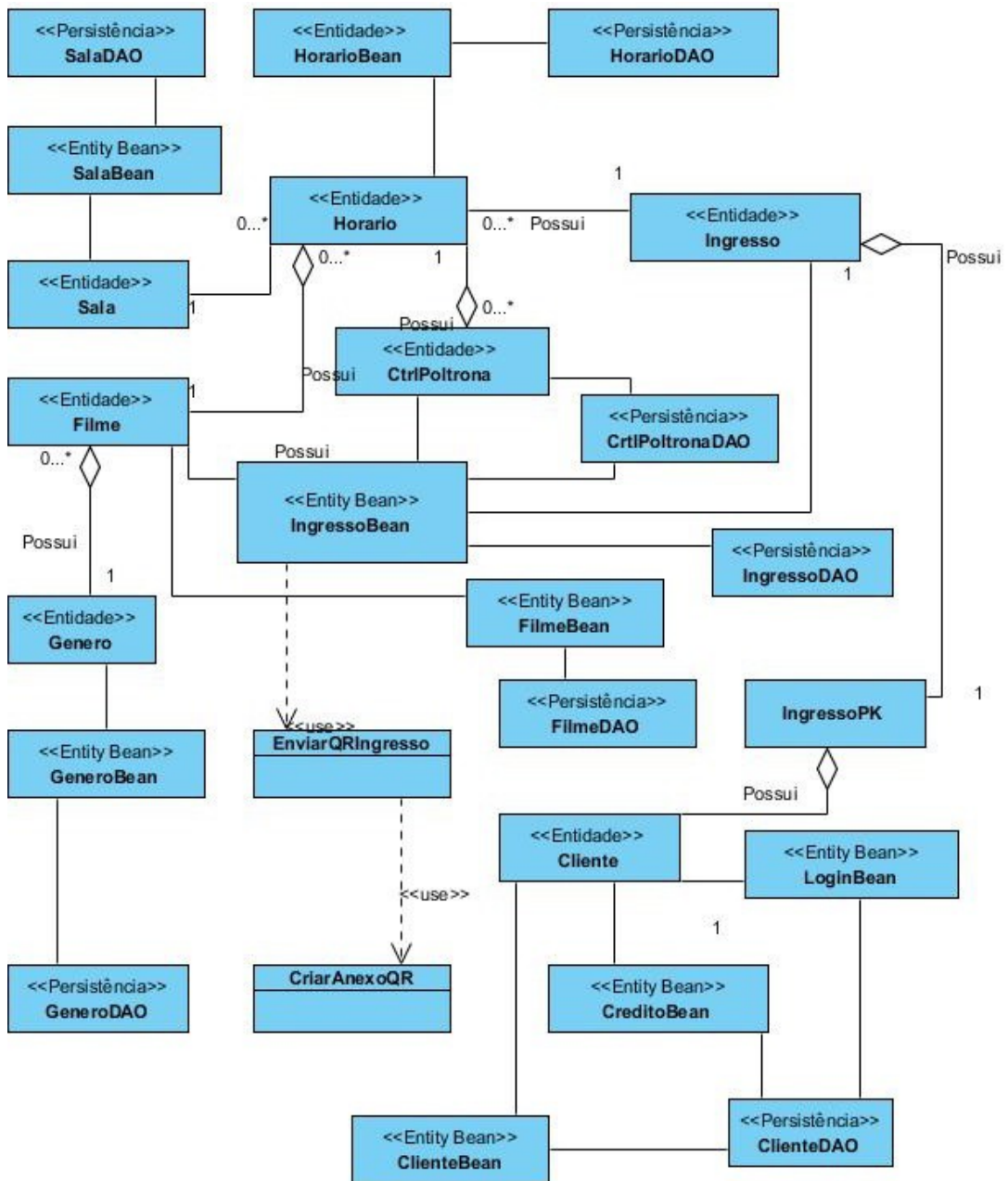


Figura 30: Diagrama de classe Sistema Web - Relacionamentos

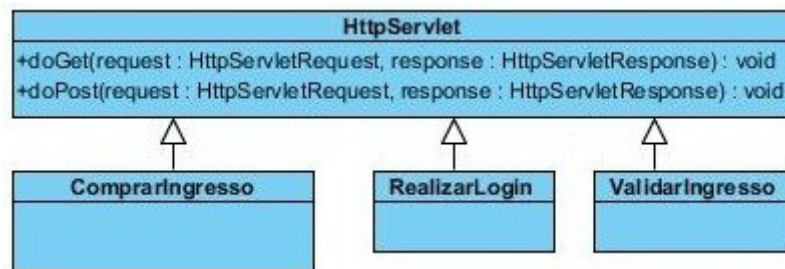


Figura 31: Classes Servlets

A Figura 32 é das classes de entidade principais do sistema.

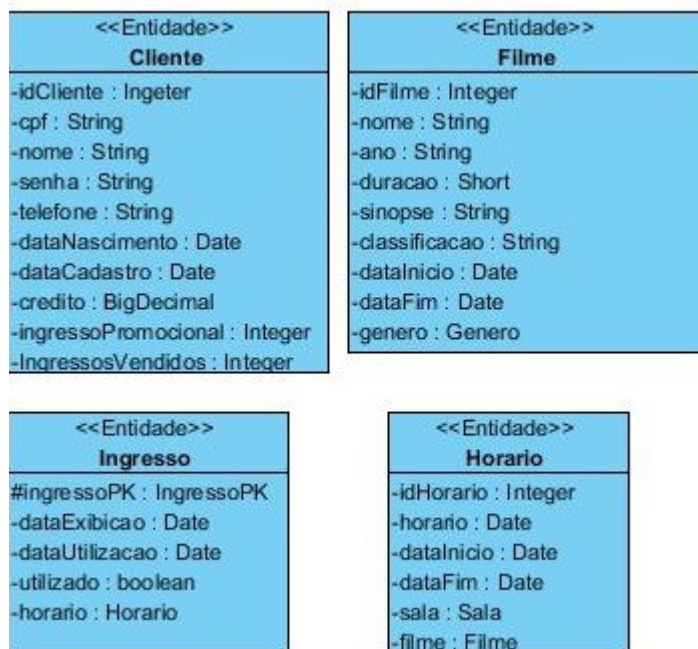


Figura 32: Classes de Entidade

As classes de persistência estão representadas na Figura 33.

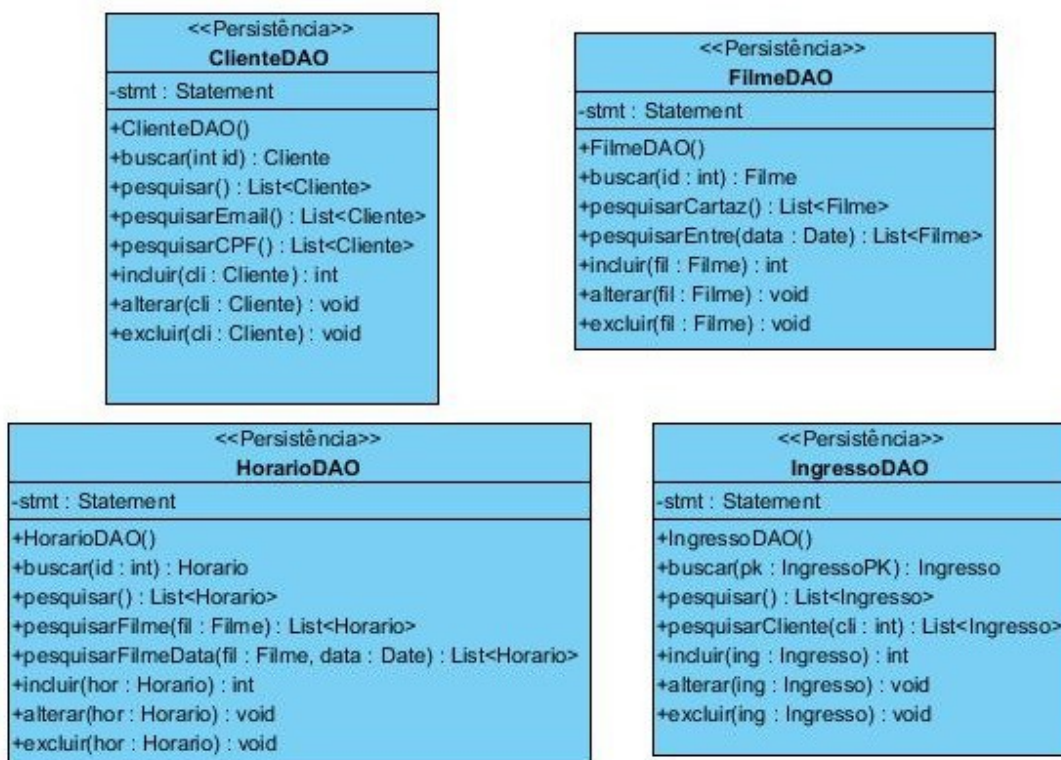


Figura 33: Classes de Persistência

As classes que controlam o mapeamento das páginas do sistema e que realizam operações de compra e validação de ingresso estão representadas na Figura 34.



Figura 34: Classes Beans

Para mais detalhes sobre o dicionário de informações dessas classes, vide Apêndice C – Detalhamento dos Dicionários de Informação.

5.4.3.3 Diagrama de Classe do Aplicativo Móvel Cliente

O diagrama de classes da Figura 58 é referente ao Aplicativo móvel acessado pelo cliente do sistema.

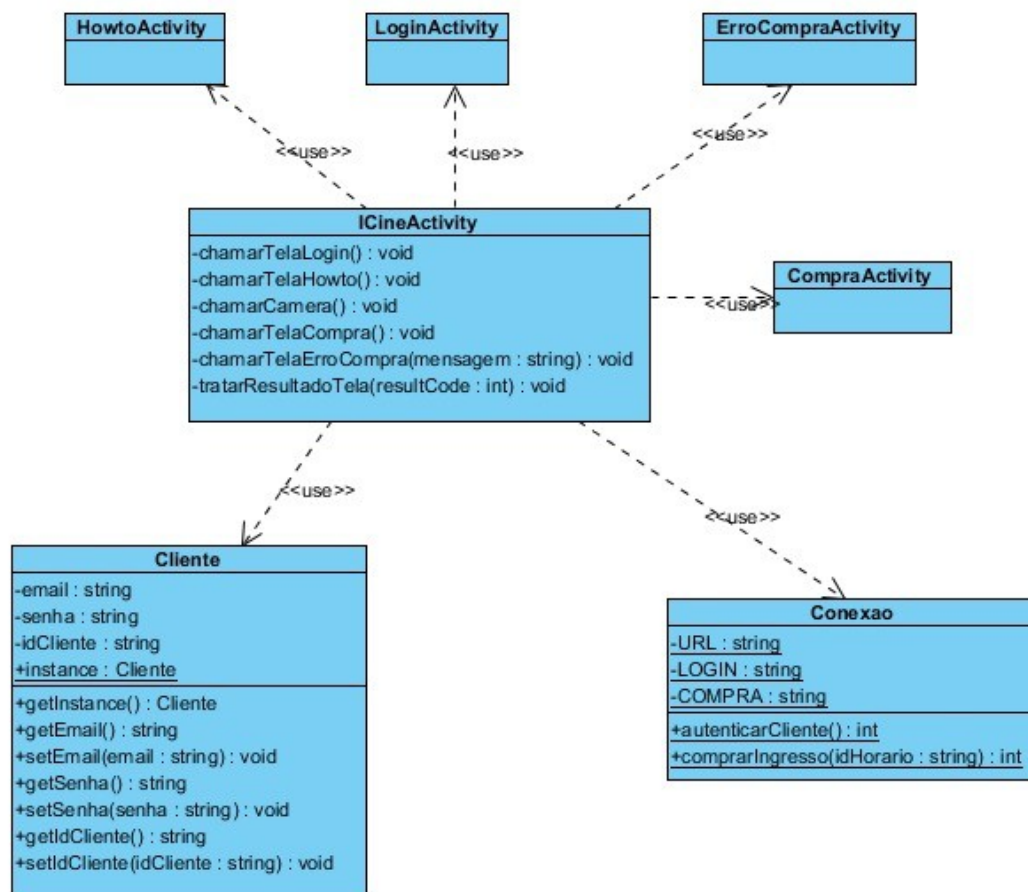


Figura 35: Diagrama de Classes do Aplicativo Móvel Cliente

Para mais detalhes sobre o dicionário de informações dessas classes, vide Apêndice C – Detalhamento dos Dicionários de Informação.

5.4.3.3 Diagrama de Classe do Aplicativo Móvel Cinema

O diagrama de classes representado na Figura 59 é referente ao Aplicativo móvel acessado pelo funcionário do cinema para validar os ingressos.

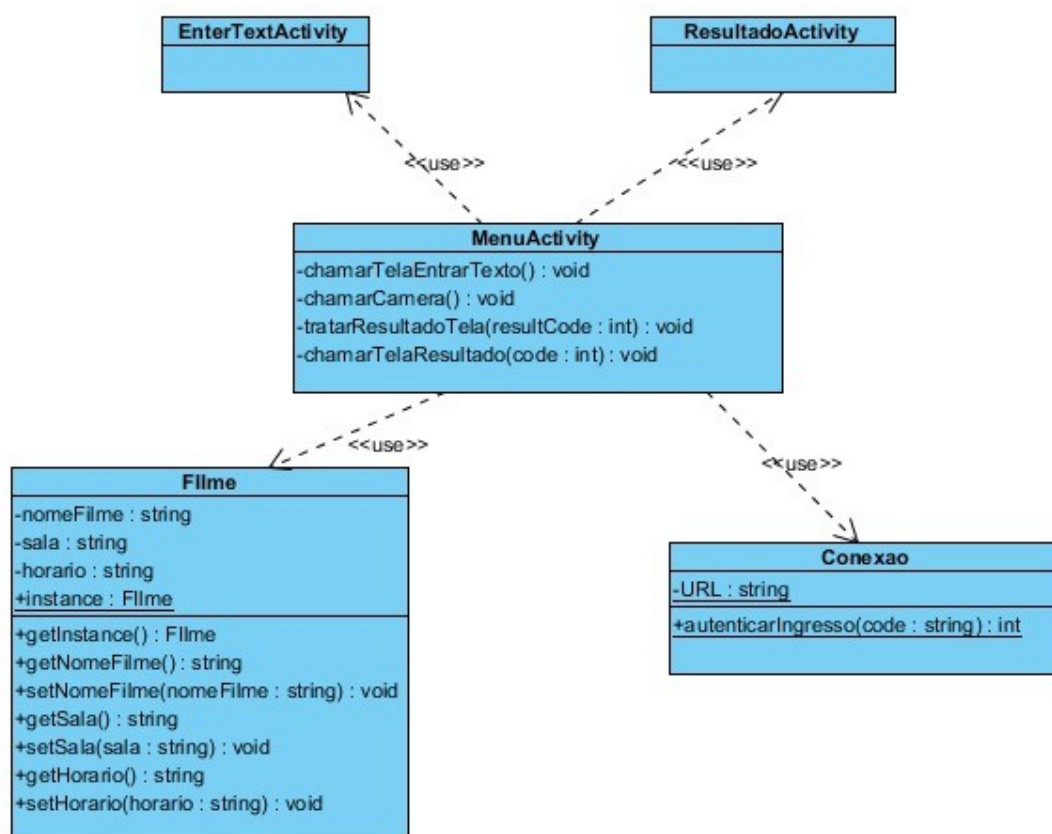


Figura 36: Diagrama de Classes do Aplicativo Móvel Cinema

Para um maior entendimento do sistema consultar o Apêndice D – Diagramas de Sequências que contém um diagrama de sequência para cada caso de uso do sistema.

Neste capítulo foram apresentados os principais diagramas de caso de uso e de classe dos sistemas desenvolvidos. No próximo capítulo será apresentado o cálculo da estimativa de horas utilizadas para o desenvolvimento.

6 ESTIMATIVA POR MEIO DE PONTOS DE CASO DE USO

Nesse capítulo será descrito todos os Pontos de Casos de Uso estimados para desenvolver o sistema e o cálculo da estimativa de horas utilizadas para o desenvolvimento. Os valores utilizados nesse projeto foram retirados dos estudos baseados no trabalho de Gustav Karner (CLEMMONS, 2006).

6.1 UUCP – *Unadjusted Use Case Points*

O Cálculo dos “Pontos de Casos de Uso não Ajustado” é baseado na somatória de dois fatores: UUCW (*Unadjusted Use Case Weight*) e UAW (*Unadjusted Actor Weight*). Nos dois casos é utilizado um valor constante (representado pela coluna Peso) multiplicado pela somatória quantitativa da ocorrência das categorias no projeto.

Neste projeto foi utilizado da tabela 4 para o cálculo do UUCW.

O Cálculo total do UUCW desse projeto está apresentado na tabela 5.

Tabela 4: Categorias de Caso de Uso (CLEMMONS, 2006)

Categoria de Caso de Uso	Descrição	Peso
Simple	Interface de usuário simples. Utiliza apenas uma entidade do banco de dados. O cenário de sucesso possui menos de 3 passos e sua implementação possui menos de 5 classes.	5
Médio	Possui interface gráfica. Utiliza duas ou mais entidades do banco de dados. O cenário de sucesso possui ente 4 e 7 passos e sua implementação envolve entre 5 a 10 classes.	10
Complexo	Possui uma interface ou processo de usuário complexo. Utiliza 3 ou mais entidades do banco de dados. O cenário de secesso possui mais de 7 passos. Sua implementação envolve mais de 10 classes.	15

Tabela 5: Cálculo UUCW do projeto

Categoria	Casos de Uso	Peso	N° casos de uso	Result.
Simple	Consultar Crédito, Visualizar Ingressos, Consultar Clientes, Consultar Filmes, Consultar Gênero, Consultar Horário, Consultar Sala, Excluir Ingressos, Excluir Cliente, Excluir Filmes, Excluir Gênero, Excluir Horário, Excluir Salas, Logar-se (Cliente Web)	5	14	70
Médio	Registrar-se, Comprar Créditos, Incluir Clientes, Alterar Clientes, Incluir Filme, Alterar Filmes, Incluir Gênero, Alterar Gênero, Incluir Horário, Alterar Horário, Incluir Salas, Alterar Salas, Validar Ingresso, Logar-se(Cliente Móvel), Validar Ingresso Texto (Aplicativo Móvel Cinema), Validar Ingresso Imagem (Aplicativo Móvel Cinema).	10	16	160
Complexo	Comprar Ingresso (Sistema Web), Comprar Ingresso (Aplicativo Móvel Cliente)	15	2	30
UUCW Total				260

O Peso dos atores não ajustados (UAW) foi retirado da tabela 6.

Tabela 6: Categoria de Ator(CLEMMONS, 2006)

Categoria de Ator	Descrição	Peso
Simple	Outro Sistema acessado através de uma API de programação.	1
Médio	Outro Sistema interagindo através de um protocolo de comunicação TCP-IP.	2
Complexo	Um usuário interagindo através de uma interface gráfica.	3

A tabela 7 representa o cálculo total do UAW desse projeto.

Tabela 7: Cálculo do UAW do Projeto

Tipo do Ator	Atores do Sistema	Peso	Nº de Atores	Result.
Simple	Nenhum.	1	0	0
Médio	Sistema Web.	2	1	2
Complexo	Cliente (Sist. Web), Administrador (Sist. Web), Cliente (App. Móvel), Funcionário (App. Móvel)	3	4	12
Total de UAW				14

Baseado nos dados acima mencionados o valor do UUCP desse projeto é a soma dos UUCW com os UAW totalizando 274.

6.2 TFC – Technical Complexity Factor

O Cálculo do TFC é realizado pela seguinte fórmula:

- $TCF = 0.6 + (0.01 * \text{FatorT})$

O FatorT é o resultado da avaliação de importância dos fatores técnicos pelos autores do projeto. Cada fator pode ser avaliado com importância entre 0 a 5 e é multiplicado por uma constante. A tabela 8 contém as constantes e o FatorT deste projeto.

Tabela 8: Cálculo do FatorT para o Projeto

Fator Técnico	Descrição	Peso	Importância	Result.
T1	Sistema Distribuído	2	0	0
T2	Tempo de Resposta	1	1	1
T3	Eficiência	1	3	3
T4	Processamento Complexo	1	0	0
T5	Reuso de Código	1	0	0
T6	Facilidade de Instalação	0.5	4	2
T7	Facilidade de Uso	0.5	5	2.5
T8	Portabilidade	2	0	0
T9	Facilidade de mudança	1	0	0
T10	Concorrência	1	0	0
T11	Recursos de Segurança	1	2	2
T12	Acessível por Terceiros	1	0	0
T13	Requer Treinamento Especial.	1	0	0
			Total do FatorT	10.5

Colocando os valores na fórmula o fator de complexidade técnico desse projeto resultou em 0.705.

6.3 EFC – Environmental Complexity Factor

Esse fator mensura a qualidade da equipe de desenvolvimento, baseado em suas experiências e trabalhos anteriores.

O Cálculo do EFC é realizado pela seguinte fórmula:

- $ECF = 1.4 + (- 0.03 * \text{FatorE})$

O FatorE é o resultado da avaliação de experiência dos atores pelos fatores de ambiente. Cada fator pode ser avaliado com valores entre 0 a 5 e

é multiplicado por uma constante. A tabela 9 contém as constantes e o FatorE deste projeto.

Tabela 9: Cálculo do ECF

Fator de Ambiente	Descrição	Peso	Avaliação	Result.
E1	Familiaridade com RUP ou outro método formal	1.5	5	7,5
E2	Experiência com a aplicação desenvolvida.	0.5	5	2,5
E3	Experiência com Orientação a objetos.	1	5	5
E4	Presença de Analista Experiente.	0.5	3	1.5
E5	Motivação	1	5	5
E6	Requisitos estáveis	2	5	10
E7	Desenvolvedores em meio expediente.	-1	3	-3
E8	Dificuldade em programação.	-1	0	0
Total do FatorE				28.5

Colocando os valores na fórmula o fator de complexidade de ambiente desse projeto resultou em 0.545.

6.4 PF – Productivity Factor

O fator de produtividade é o número de horas por ponto de caso de uso do projeto baseado em experiências anteriores. Esse valor geralmente varia entre 15 e 30 horas, como esse é o primeiro projeto dos autores em conjunto o fator de produtividade utilizado foi de 20 horas.

6.5 Cálculo Final de Horas Necessárias

A fórmula que estima quantas horas são necessárias para realizar o projeto é a seguinte:

- $UCP = UUCP * TCF * ECF * PF$

Colocando os valores já previamente calculados na fórmula chegamos ao seguinte resultado:

- $UCP = 274 * 0.705 * 0.545 * 20 = 2105 \text{ horas.}$

Esse valor resulta em 11 meses de trabalho com 2 desenvolvedores com carga horária de 4 horas diárias, 20 horas semanais. Nota-se que este total é apenas uma estimativa. Na prática levou-se menos tempo, devido ao empenho e conhecimento prévio empregado pelos autores do projeto.

Este capítulo apresentou a estimativa por pontos de caso de uso, apresentando os cálculo e a estimativa de horas necessárias para implementar o projeto. No próximo capítulo será apresentado o cronograma de desenvolvimento do projeto.

7 CRONOGRAMA

A seguir estão listadas as principais atividades para realizar a implementação do projeto seus tempos estimados. O tempo total não representa o tempo para a realização de todas as atividades visto que existem atividades omitidas e atividades que podem ser executadas simultaneamente. Mais detalhes consultar o Apêndice E – Detalhamento dos Cronogramas.

O Apêndice E também contém a rede PERT/CPM em relação a todas as atividades do projeto.

A tabela 39 demonstra as atividades e horas estimadas da fase de análise do projeto.

Tabela 10: Análise Documentação e Modelagem de Dados

Atividades	Horas Estimadas
Análise/Documentação e Modelagem Banco de dados	
Levantamento de Requisitos	8 horas
Diagramas de Caso de Uso	12 horas
Diagrama de Classe	8 horas
Diagrama de Sequência	32 horas
Diagrama de Entidade-Relacionamento	8 horas
Implementação do Banco de dados	8 horas

A tabela 40 demonstra as atividades e horas estimadas da fase de programação web do projeto, esta fase pode ser executada em paralelo com a fase de programação para Sistemas Móveis representado na tabela 41.

Tabela 11: Programação do Sistema Web

Atividades	Horas Estimadas
Programação do Sistema Web	
Implementação do CRUD de cliente	100 horas
Implementação do método de Gerar Código QR	16 horas
Implementação do método de enviar e-mail	16 horas
Implementação do CRUD de filme	100 horas
Implementação do CRUD de horário	100 horas
Implementação do CRUD de Ingresso	100 horas
Implementação do servlet de login	16 horas
Implementação do auto cadastro de cliente	16 horas
Implementação do método de comprar ingresso	56 horas
Implementação do servlet de comprar ingresso	16 horas
Implementação do método de validar ingresso	36 horas
Implementação do servlet de validar ingresso	16 horas

Tabela 12: Programação dos Sistemas Móveis

Atividades	Horas Estimadas
Programação dos Sistemas Móveis	
Montar o ambiente de programação Android	16 horas
Adaptação da API de decodificação do QR-Code	16 horas
Implementação do método de login	16 horas
Implementação do método de login através do aplicativo móvel	36 horas
Implementação do método de comprar ingresso pelo aplicativo móvel	36 horas
Implementação do método de validar ingresso pelo celular	36 horas

A fase de testes pode ser visualizada na tabela 42 nomeada como Etapa Final.

Tabela 13: Etapa Final

Atividades	Horas Estimadas
Etapa Final	
Testes	16 horas

Neste capítulo foram apresentados as tabelas com a estimativa em horas para cada etapa de execução do projeto. No próximo capítulo serão descritos os procedimentos de testes e validação utilizados nos sistemas.

8 PROCEDIMENTO DE TESTE E VALIDAÇÃO

O procedimento de teste para esse projeto foi realizado paralelo ao desenvolvimento do mesmo, conforme as atividades eram concluídas eram realizados pequenos testes para verificar a funcionalidade e o desempenho. Cada atividade e seu procedimento de testes serão explicados mais detalhadamente a seguir.

O módulo de análise deste projeto foi realizado pelos autores em conjunto, toda a parte de análise foi discutida e revisada para que estivesse bem fundamentadas para o início da implementação do projeto.

A implementação do banco de dados foi realizada através de um *script* SQL visando facilitar possíveis futuras restaurações. Foi uma escolha acertada pois em uma das etapas do projeto foi necessário realizar a restauração do banco de dados.

A implementação do sistema web foi realizada em módulos, primeiramente sendo implementadas as classes de entidade com suas telas e métodos de busca, inclusão, alteração e exclusão para as tabelas que não possuíam relacionamentos. A ordem exata da implementação foi: cliente, gênero, sala, filme, horário e ingresso. O procedimento de testes para cada módulo foi muito similar, conforme a classe de entidade era codificada eram realizados testes de busca, inclusão alteração e exclusão das mesmas. Mesmo com a realização dos testes a cada etapa do projeto, ao final da codificação foram realizados testes englobando o sistema por inteiro, seguindo os cenários dos casos de uso. Nesta etapa foram detectados conflitos entre as consultas visualizadas na tela do programa com o que efetivamente estava no banco dados. Para a resolução deste problema foram realizadas algumas medidas. Inicialmente o banco de dados foi restaurado, posteriormente o *framework* de mapeamento objeto-relacional Hibernate foi trocado pelo Oracle Toplink, mesmo assim os erros permaneceram. Para solucionar esse problema foi removido o *framework* de

mapeamento objeto-relacional e realizadas as implementações dessas classes manualmente.

Em paralelo a implementação do sistema web foram iniciadas as implementações dos aplicativos celulares. Anteriormente a esta implementação, foram realizadas adaptações na API ZXing para que ela ficasse mais leve e rápida a fim de melhorar seu desempenho nos aplicativos celulares. A cada mudança na API foram realizados testes com *smartphones* da LG (Optimus-Pro), verificando o desempenho e funcionalidade. Pela familiaridade com a API ZXing, ao fim da adaptação da API foi realizado a implementação dos métodos de geração de *QR-Code* e envio de e-mail com a imagem gerada em anexo, para ser inserido no sistema web.

Ao final da implementação das classes de persistência no sistema web, foi iniciada a codificação de login e verificadas as sessões criadas pelos usuários no sistema. Após realizados os testes e verificado o completo funcionamento do método foi implementado o *servlet* de login para responder ao aplicativo celular.

Os casos de uso de comprar e validar ingresso foram implementados utilizando os métodos de criação e envio de e-mail. Ao final de cada etapa foram realizados testes para verificar se as validações implementadas estavam corretas e corrigidas quando necessário. Foi notado uma perda de desempenho no sistema com o método de enviar e-mail, foram realizadas várias tentativas para sanar o problema mas nenhuma delas surtiu o efeito desejado.

Com os métodos de compra e validação funcionando pode ser implementado os *servlets* que respondem as solicitações dos aplicativos celulares. Os testes desta etapa foram realizados pelos autores em conjunto corrigindo falhas em ambos os sistemas.

Com os sistemas prontos foram realizados testes usando como base os casos de uso do projeto. Nessa etapa são corrigidos erros de validação, verificados e tratados mensagens de erro exibidos para o cliente em busca

de tentar forçar situações inusitadas e mesmo assim manter o programa em funcionamento.

Foram realizados testes de caixa-preta para cada tela do sistema, embora os campos da telas estejam todos com máscaras, impedindo a entrada de letras em campos numéricos e vice-versa. Foram realizados testes de validação para datas, verificando entrada de datas de nascimento e impedindo o cadastramento caso seja igual a data atual ou posterior, incluindo validações mais complexas como a inserção da data inicial e final de filmes e horários, neste caso em especial as datas iniciais devem ser maiores ou iguais a data atual e as datas finais devem ser maiores que as iniciais, as datas iniciais e finais do filme devem conter as datas iniciais e finais dos horários.

As validações de e-mail foram realizadas através de “Expressões Regulares”. Para a validação de CPF foi utilizado um método específico.

Todas as validações foram testadas a fim de impedir o cadastramento de informações incorretas.

Neste capítulo foram apresentados os procedimentos usados para os testes dos sistemas. O próximo capítulo expõe uma análise de riscos para o projeto.

9 ANÁLISE DE RISCOS

Os riscos analisados para o desenvolvimento do projeto estão listados na tabela abaixo:

Tabela 14: Análise de Riscos

Risco	Probabilidade	Impacto	Resolução
Banco de dados			
Banco de dados Indisponível	Baixa	4	O Sistema notifica a falha ao usuário.
Perca de dados	Muito Baixa	4	Restauração do Banco de dados.
Equipamentos			
Falha no <i>smartphone</i> utilizado no desenvolvimento.	Baixa	3	Substituição do aparelho utilizado.
Falha no servidor	Muito Baixa	4	Serviço de servidor virtualizado na nuvem.
Falha na infraestrutura de desenvolvimento.	Média	2	Substituir o componente defeituoso.
Software			
Vírus na máquina do desenvolvedor.	Média	2	Uso de software antivírus.
Humana			
Desenvolvedor indisponível	Média	2	Aumento nas horas semanais trabalhadas para cumprimento do cronograma.

O nível de impacto descrito na tabela a cima varia entre 1 a 4 sendo que 1= impacto baixo, 2 = impacto marginal, 3= impacto crítico e 4 impacto catastrófico.

Os riscos mais impactantes para este projeto são a indisponibilidade do banco de dados, perca dos dados inseridos no mesmo ou falha no servidor. Como este é apenas um sistema acadêmico com pouca massa de dados, e os dados inseridos sendo apenas para testes e tendo pouca

importância, não foi realizado um investimento na manutenção e conservação dos mesmos. A probabilidade de incidência desses riscos sendo baixa ou muito baixa, colaborou na despreocupação com a ocorrência desses agravantes, caso que seria analisado mais cuidadosamente em um sistema real.

No caso de falha nos smartphones e falha na infraestrutura, haveria um transtorno maior no desenvolvimento do projeto para os autores. Seria necessário despender recursos para a substituição dos equipamentos.

O capítulo apresentou um levantamento dos possíveis riscos que envolvem o desenvolvimento do projeto apresentado. O próximo capítulo trata dos planos de contingência para os riscos mencionados.

9.1 Planos de Contingência

A seguir serão listados os planos de contingência para cada risco analisado.

- Perca de Dados: Realização periódica de Back-ups na base de dados para possíveis restaurações.
- Falha no *smarphone* utilizado no desenvolvimento: possuir um aparelho do mesmo modelo para rápida substituição.
- Falha no servidor: Caso ocorra a empresa responsável pela virtualização realizaria a restauração automaticamente.
- Falha na infraestrutura de desenvolvimento: Realização de Back-ups periódicos do sistema em desenvolvimento podendo-se usar versionamento facilitaria a recuperação do trabalho. A máquina será substituída pela reserva.
- Desenvolvedor indisponível: Tentar ao máximo adiantar o cronograma a fim de contornar possíveis atrasos.

Neste capítulo foram listados as resoluções para os riscos apresentados na análise de riscos. No próximo capítulo serão apresentadas as conclusões e trabalhos futuros que podem ser desenvolvidos a partir deste trabalho.

10 CONCLUSÕES

A proposta desse projeto foi a implementação de um sistema que integrasse aplicativos móveis com uma aplicação *web* utilizando a ferramenta *QR-Code* como mediadora dessa integração. O objetivo principal era fazer com que o *QR-Code* realizasse funções dentro do sistema, diversificando do modo com que ele é empregado no Brasil atualmente.

Para embasar ainda mais a proposta, foram realizadas pesquisas sobre o *QR-Code* e sua utilização no Brasil, uma análise mercadológica do uso dos celulares *smartphones* e seu crescimento também foi realizada.

O principal desafio do projeto foi a adaptação da API de geração dos códigos QR e sua implementação para o sistema operacional Android.

Com a análise mercadológica completa foi iniciada a fase de Análise e Documentação da proposta. Essa fase foi tranquila e não houveram imprevistos nem atrasos tendo duração de um mês e meio.

Para fundamentar essa proposta foi decidido a implementação de um sistema de compra de ingressos de cinema através da internet.

A fase de codificação do sistema *web* teve início em Janeiro de 2012 e teve duração de 8 meses. Nessa etapa foram encontrados erros de conflitos entre o *framework* de *mapeamento-relacional* e o *framework* JSF, a solução do problema durou em média 120 horas tendo duração de mais ou menos um mês.

A codificação móvel ocorreu em paralelo com a codificação *web*, levando menos tempo e tendo duração de 5 meses. Como a implementação do projeto foi realizada por dois programadores um ficou responsável pelo sistema *web* e o outro pelos aplicativos móveis. O desenvolvedor responsável pela codificação dos aplicativos celulares ficou responsável pelos testes e adaptação da API ZXing desenvolvendo também os métodos de envio de e-mail. Os erros ocorridos na fase de codificação *web* foram solucionados em conjunto, com a experiência dos dois desenvolvedores.

10.1 Propostas de Melhorias

Uma das propostas de melhorias do sistema é o desenvolvimento do aplicativo móvel cliente para o Sistema Operacional IOS. Esta melhoria abrangeria um maior número de clientes incluindo os usuários de iPhone e iPad.

Outra proposta seria codificar o ingresso diretamente pelo aplicativo, evitando que o usuário necessitasse abrir um aplicativo cliente de e-mail. Essa melhoria assegura que o usuário possa usufruir de seus ingressos mesmo em casos de falha do provedor de e-mail.

11 REFERÊNCIAS

AGÊNCIA NACIONAL DE TELECOMUNICAÇÕES. **Relatório Anual 2011**. Disponível em: <<http://www.anatel.gov.br/Portal/verificaDocumentos/documento.asp?numeroPublicacao=278637&assuntoPublicacao=Relat%F3rio%20Anual%202011&caminhoRel=In%EDcio--&filtro=1&documentoPath=278637.pdf>>. Acesso em: 16 nov. 2012.

BANCO DO BRASIL. **QR Code**. Disponível em: <<http://www.bb.com.br/portalbb/page3,101,2183,0,0,1,1.bb>>. Acesso em 02 dez. 2012.

Banda larga móvel dobrou no País em 2011. Disponível em: <<http://www.meioemensagem.com.br/home/marketing/noticias/2012/03/20/Banda-larga-movel-dobrou-no-Pais-em-2011.html>>. Acesso em 02 dez 2012.

Branding Magazine. Disponível em: <<http://www.brandingmagazine.com/wp-content/uploads/2011/11/nook-tablet3.jpg>>. Acesso em 01 dez 2012.

Brasil ultrapassa marca de 200 milhões de telefones celulares, mostra IBGE. Disponível em: <<http://www.piauihoje.com/noticias/brasil-ultrapassa-marca-de-200-milhoes-de-telefones-celulares-mostra-ibge-26590.html>>. Acesso em 03 dez. 2012.

CLEMMONS, Roy K. **Cosstalk: The Journal of Defense Software Engineering**. 2006.

Cresce número de adolescentes e crianças com celular, diz IBGE. Disponível em: <<http://g1.globo.com/mato-grosso-do-sul/noticia/2012/10/cresce-numero-de-adolescentes-e-criancas-com-celular-no-pais-diz-ibge.html>>. Acesso em: 13 dez 2012.

DENSO WAVE. **Micro qr code**. Disponível em: <<http://www.qrcode.com/en/microqr.html>>. Acesso em: 16 nov. 2012

DENSO WAVE. **Qr Code features**. Disponível em: <<http://www.qrcode.com/en/qrfeature.html>>. Acesso em: 15 nov. 2012.

GOOGLE. **Nosso Planeta Mobile: Brasil – como entender o usuário celular**. Disponível em: <http://services.google.com/fh/files/blogs/our_mobile_planet_brazil_pt_BR.pdf>. Acesso em: 17 nov. 2012.

INSTITUTO BRASILEIRO DE GEOGRAFIA E ESTATÍSTICA. **Acesso à Internet e posse de telefone móvel celular para uso pessoal 2008: De 2005 para 2008, acesso à Internet aumenta 75,3%**. Disponível em:

<http://www.ibge.gov.br/home/presidencia/noticias/noticia_impressao.php?id_noticia=1517>. Acesso em 06 dez. 2012.

INSTITUTO BRASILEIRO DE GEOGRAFIA E ESTATÍSTICA. **PNAD 2011: crescimento da renda foi maior nas classes de rendimento mais baixas.** Disponível em: <http://www.ibge.gov.br/home/presidencia/noticias/noticia_visualiza.php?id_noticia=2222&id_pagina=1>. Acesso em: 17 nov. 2012.

LECHETA, Ricardo R. **Google ANDROID: Aprenda a criar aplicações para dispositivos móveis com Android SDK.** 2. ed. São Paulo: Novatec, 2010.

MERIGO, Carlos. **Fast Shop veicula anúncio com QR Code.** Disponível em: <<http://www.brainstorm9.com.br/1564/diversos/fast-shop-veicula-anuncio-com-qr-code/>>. Acesso em: 18 nov. 2012.

MILANI, André. **PostgreSQL: Guia do programador.** São Paulo: Novatec, 2008.

QR-Code no Brasil. Disponível em: <<http://qrcodenobrasil.blogspot.com.br/2012/04/qr-code-e-os-vingadores-avengers-no.html>>. Acesso em 29 nov 2012.

Vendas de smartphones no Brasil subiram 77% no primeiro semestre, diz IDC. Disponível em: <<http://www.mobiletime.com.br/13/09/2012/vendas-de-smartphones-no-brasil-subiram-77-no-primeiro-semester-diz-idc/299299/news.aspx>>. Acesso em: 16 nov. 2012.

APÊNDICE A – Detalhamento dos Casos de Uso

Este apêndice contém os diagramas de casos de uso do projeto detalhados.

Os Diagramas de Caso de Uso a seguir representam todos os casos de uso do sistema web acessados pelo cliente.



Figura 1: Caso de Uso Registrar-se

Caso de Uso: Diagrama de caso de uso Registrar-se .

Descrição: Esse caso de uso permite fazer o cadastro de um novo cliente.

Ator Principal: Cliente.

Pré-condições: Não existe pré-condições para esse caso de uso.

Pós-condições: O sistema cadastra, realiza o login do cliente cadastrado e redireciona para a página inicial.

Fluxo Básico de Eventos

1. Cliente aciona botão “Cadastre-se”.
2. Cliente informa os campos obrigatórios (Nome, CPF, E-mail, Senha e Confirmação de Senha) e se quiser preenche os demais campos.
3. Cliente solicita o cadastro.
4. Sistema realiza cadastro.
5. Sistema realiza login do cliente cadastrado.
6. Sistema redireciona o cliente para a página inicial.
7. Sistema retorna mensagem de sucesso.

Fluxo Alternativo – Campo obrigatório faltante

1. Cliente não preenche todos os campos obrigatórios necessários.

2. Cliente solicita cadastro
3. Sistema retorna mensagem solicitando campo faltante.

Regras de Negócio

- a) Só poderá haver um cadastro por CPF;
- b) Só poderá haver um cadastro por E-mail;
- c) O cliente deve informar um endereço de e-mail válido;
- d) O Cliente deve informar número de CPF válido;

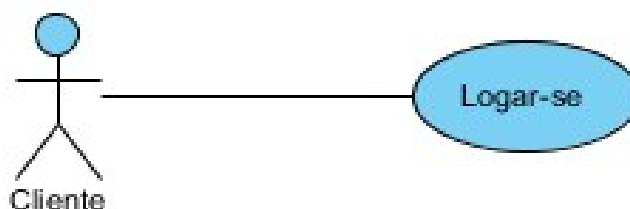


Figura 2: Caso de uso Logar-se - Cliente

Caso de Uso: Diagrama de caso de uso Logar-se - Cliente.

Descrição: Esse caso de uso permite ao usuário fazer o login no sistema.

Ator Principal: Cliente.

Pré-condições: Não existe pré-condições para esse caso de uso.

Pós-condições: O sistema realiza o login do usuário e redireciona para a página inicial do sistema.

Fluxo Básico de Eventos

1. Cliente informa o E-mail e Senha.
2. Cliente solicita o Login.
3. Sistema verifica informações enviadas.
4. Sistema adiciona nome do cliente no menu de login.
5. Sistema realiza login no sistema.
6. Sistema redireciona usuário para página inicial do sistema.

Fluxo Alternativo – Cliente Informa campo errado

1. Cliente informa campo e-mail ou campo senha incorretamente.

2. Cliente solicita login no sistema.
3. Sistema retorna mensagem de campo incorreto.

Regras de Negócio

- a) O Cliente só poderá realizar o login se informado campo login e senha corretamente, respeitando caracteres maiúsculos e minúsculos.

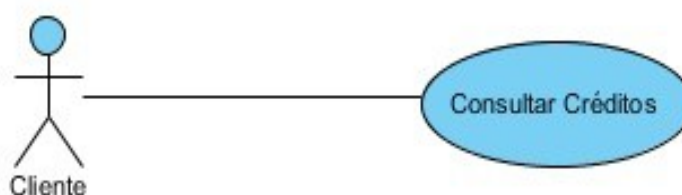


Figura 3: Caso de Uso - Consultar Crédito

Caso de Uso: Diagrama de caso de uso Consultar Crédito.

Descrição: Esse caso de uso permite ao usuário fazer o a consulta de créditos.

Ator Principal: Cliente.

Pré-condições: O cliente deve estar logado no sistema.

Pós-condições: O sistema consulta os créditos que o cliente possui no banco de dados e redireciona para a página de comprar créditos.

Fluxo Básico de Eventos

1. Cliente aciona menu “Créditos”.
2. Sistema redireciona para página de compra de créditos informando saldo do Cliente.

Regras de Negócio

- a) O Cliente só poderá realizar a consulta se estiver logado no sistema.



Figura 4: Caso de uso Comprar Créditos

Caso de Uso: Diagrama de caso de uso Comprar Crédito.

Descrição: Esse caso de uso permite ao usuário comprar créditos.

Ator Principal: Cliente.

Pré-condições: O cliente deve estar logado no sistema.

Pós-condições: O sistema atualiza os créditos do cliente no banco de dados e mostra novo saldo na tela de Créditos.

Fluxo Básico de Eventos

1. Cliente aciona menu “Créditos”.
2. Sistema redireciona para página de compra de créditos informando saldo do Cliente.
3. Cliente informa quantidade de créditos que deseja incluir.
4. Sistema atualiza o campo de créditos da tabela de cliente adicionando o novo valor ao anterior.
5. Sistema atualiza tela de Créditos com o novo Saldo.

Regras de Negócio

- a) O Cliente só poderá realizar a consulta se estiver logado no sistema.



Figura 5: Caso de Uso Visualizar Ingressos

Caso de Uso: Diagrama de caso de uso Visualizar Ingressos

Descrição: Esse caso de uso permite ao usuário visualizar os ingressos vinculados a ele.

Ator Principal: Cliente.

Pré-condições: O cliente deve estar logado no sistema.

Pós-condições: O sistema consulta os ingressos que o cliente possui no banco de dados e redireciona para a página de Ingressos.

Fluxo Básico de Eventos

1. Cliente aciona menu de ingressos.
2. Sistema busca ingressos do cliente no banco de dados, ordenada por data.
3. Sistema redireciona o cliente para a página de Ingressos.

Regras de Negócio

- a) O Cliente só poderá realizar a consulta se estiver logado no sistema.

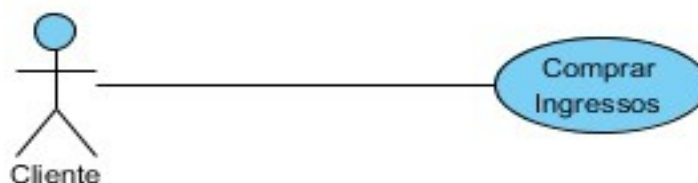


Figura 6: Caso de Uso Comprar Ingressos

Caso de Uso: Comprar Ingressos.

Descrição: Esse caso de uso permite ao usuário realizar a compra de ingressos.

Ator Principal: Cliente.

Pré-condições: O cliente deve estar logado no sistema.

Pós-condições: O sistema realiza a compra de ingressos e redireciona para a página de consulta de ingressos.

Fluxo Básico de Eventos

1. Cliente aciona menu de ingressos.
2. Sistema redireciona o cliente para a página de Ingressos.
3. Cliente solicita redirecionamento para página de compra.
4. Sistema redireciona para página de compra de ingressos.
5. Cliente informa Data da exibição;
6. Sistema preenche lista com os filmes em cartaz na data escolhida;
7. Cliente escolhe o filme;
8. Sistema preenche lista com os horários do filme que ainda possuam poltronas disponíveis;
9. Cliente solicita compra;

10. Sistema verifica créditos promocionais do cliente, se não possuir o sistema verifica crédito do cliente;
11. Sistema debita valor do ingresso do crédito do cliente
12. Sistema incrementa um ingresso ao cadastro do cliente
13. Sistema verifica se ingressos comprados é múltiplo de 10 e se for inclui um ingresso promocional ao cadastro do cliente.
14. Sistema realiza cadastro do ingresso no banco de dados.
15. O Sistema Web envia um e-mail com imagem do QR-Code do ingresso em anexo.
16. Sistema redireciona para página de consulta de ingressos.

Fluxo Alternativo – Cliente cancela compra

1. Cliente aciona o botão cancelar na tela de compra de ingressos
2. Sistema redireciona o Cliente para página de consulta de ingressos.

Fluxo Alternativo – Cliente não possui ingressos promocionais e nem créditos suficientes

1. Cliente solicita a compra de ingresso.
2. Sistema verifica créditos promocionais do cliente, se não possuir o sistema verifica crédito do cliente.
3. Sistema envia mensagem na tela do usuário informando que o mesmo não possui créditos suficientes.

Regras de Negócio

- a) O Cliente só poderá realizar a compra se estiver logado no sistema.
- b) O Cliente só poderá efetuar a compra se possuir ingressos promocionais ou créditos igual ou superior a 10 reais.
- c) A data de exibição informada deverá ser maior que a data atual.



Figura 7: Caso de Uso Excluir Ingresso

Caso de Uso: Excluir Ingresso.

Descrição: Esse caso de uso permite ao usuário excluir ingressos vinculados a ele.

Ator Principal: Cliente.

Pré-condições: O cliente deve estar logado no sistema e possuir ingressos.

Pós-condições: O sistema excluir permanentemente o ingresso selecionado e redireciona para a página de Ingressos.

Fluxo Básico de Eventos

1. Cliente aciona menu de ingressos.
2. Sistema redireciona o cliente para a página de Ingressos.
3. Cliente aciona botão de exclusão de um ingresso
4. Sistema solicita confirmação de exclusão
5. Cliente confirma a exclusão do ingresso.
6. Sistema exclui ingresso do banco de dados.
7. Sistema atualiza a página de ingressos.

Regras de Negócio

- a) O Cliente deve estar logado.
- b) O Cliente deve possuir ingressos.
- c) O Cliente deve confirmar a exclusão para excluir o ingresso.

Os Diagramas de Caso de Uso a seguir representam todos os casos de uso do sistema web acessados pelo Administrador.

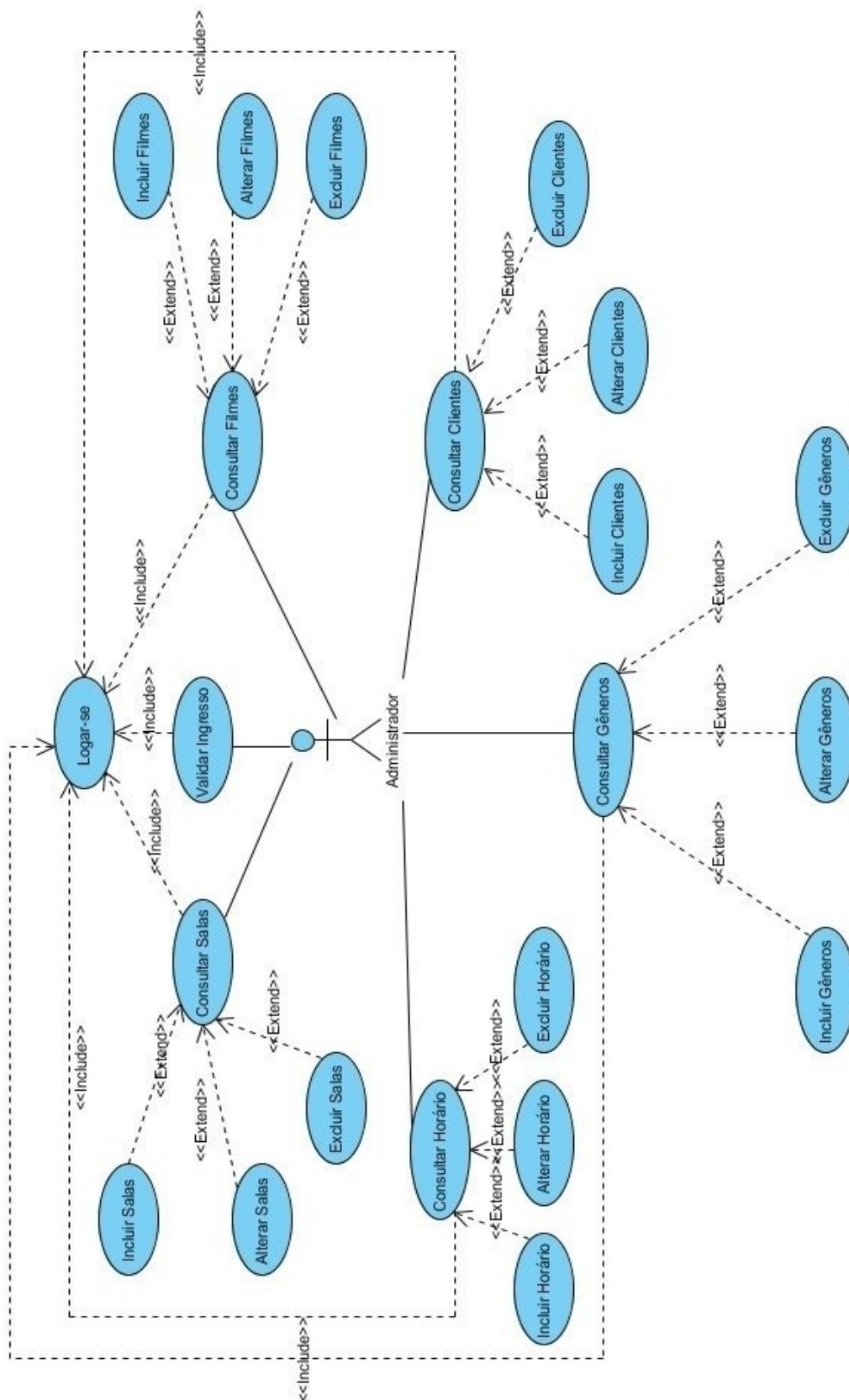


Figura 8: Caso de Uso Geral do Sistema Web Acessado Pelo Administrador

Caso de uso: Caso de uso geral do sistema Web acessado pelo Administrador.

Descrição: Esse caso de uso permite ver todas as ações que o administrador pode realizar. Essas requisições são: logar-se, gerenciar cliente (inserir, alterar e excluir), gerenciar filme (inserir, alterar e excluir), gerenciar gênero (inserir, alterar e excluir), gerenciar horário (inserir, alterar e excluir), gerenciar sala (inserir, alterar e excluir) e validar ingresso.

Ator Principal: Administrador.

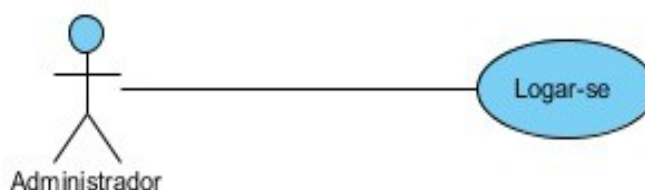


Figura 9: Caso de Uso Logar-se - Administrador

Caso de Uso: Logar-se – Administrador.

Descrição: Esse caso de uso permite fazer login como Administrador.

Ator Principal: Administrador.

Pré-condições: Não existe pré-condições para esse caso de uso.

Pós-condições: O sistema realiza o login do administrador e redireciona para a área de administração do sistema.

Fluxo Básico de Eventos

1. Administrador informa e-mail e senha de administrador.
2. Administrador aciona o botão “Logar”.
3. Sistema realiza login do administrador.
4. Sistema redireciona para área de administração do sistema.

Fluxo Alternativo – Campo e-mail ou senha incorretos

1. Administrador informa campo e-mail e/ou senha incorretos.
2. Administrador solicita login no sistema.
3. Sistema retorna mensagem de campo incorreto.

Regras de Negócio

- a) O Administrador só poderá realizar o login se informado campo login e senha corretamente, respeitando caracteres maiúsculos e minúsculos.

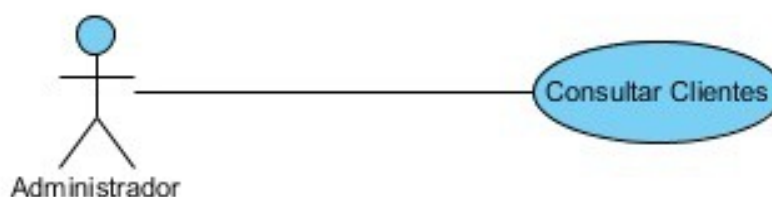


Figura 10: Caso de Uso Consultar Clientes

Caso de Uso: Consultar Clientes.

Descrição: Esse caso de uso permite consultar os clientes cadastrados no sistema.

Ator Principal: Administrador.

Pré-condições: O Administrador precisa estar logado no sistema.

Pós-condições: O sistema redireciona para página de clientes com a lista dos clientes cadastrados no banco de dados.

Fluxo Básico de Eventos

1. Administrador aciona menu de Clientes.
2. Sistema busca clientes no banco de dados.
3. Sistema redireciona o Administrador para a página de clientes com a lista atualizada.

Regras de Negócio

- a) O Administrador deve estar logado no sistema.



Figura 11: Caso de Uso Incluir Clientes

Caso de Uso: Incluir Clientes

Descrição: Esse caso de uso permite cadastrar um cliente.

Ator Principal: Administrador.

Pré-condições: O Administrador deve estar logado no sistema e ter acionado o menu de Clientes.

Pós-condições: O sistema cadastra um novo cliente e redireciona o Administrador para a página de consultar clientes .

Fluxo Básico de Eventos

1. Administrador aciona link de Inserir na página de consultar clientes.
2. Sistema redireciona o administrador para a página de incluir clientes.
3. Administrador informa os campos obrigatórios (Nome, CPF, E-mail, Senha e Confirmação de Senha) e se quiser preenche os demais campos.
4. Administrador solicita cadastro das informações.
5. Sistema cadastra informações no banco de dados.
6. Sistema atualiza lista de clientes
7. Sistema redireciona administrador para a página consultar clientes.

Fluxo Alternativo – Administrador não preenche todos os campos obrigatórios

1. Administrador não preenche todos os campos obrigatórios.
2. Administrador solicita cadastro das informações.
3. Sistema retorna mensagem de campo obrigatório faltante.

Fluxo Alternativo – Administrador solicita cancelamento

1. Administrador aciona botão voltar.
2. Sistema redireciona administrador para página de consultar clientes

Regras de Negócio

- a) Só poderá haver um cadastro por CPF;
- b) Só poderá haver um cadastro por E-mail;
- c) O Administrador deve informar um endereço de e-mail válido;
- d) O Administrador deve informar número de CPF válido;
- e) O Administrador deve estar logado no sistema.



Figura 12: Caso de Uso Alterar Clientes

Caso de Uso: Alterar Clientes

Descrição: Esse caso de uso permite alterar um cliente cadastrado.

Ator Principal: Administrador.

Pré-condições: O Administrador deve estar logado no sistema e ter acionado o menu de Clientes.

Pós-condições: O sistema altera um cliente cadastrado e redireciona o Administrador para a página de consultar clientes .

Fluxo Básico de Eventos

1. Administrador aciona link de Alterar na página de consultar clientes.
2. Sistema redireciona o administrador para a página de alterar clientes.
3. Administrador altera a informação desejada.
4. Administrador aciona botão de alterar informações.
5. Sistema atualiza informações no banco de dados.
6. Sistema atualiza lista de clientes
7. Sistema redireciona administrador para a página consultar clientes.

Fluxo Alternativo – Administrador solicita cancelamento

1. Administrador aciona botão voltar.
2. Sistema redireciona administrador para página de consultar clientes

Regras de Negócio

- a) Só poderá haver um cadastro por CPF;
- b) Só poderá haver um cadastro por E-mail;
- c) O Administrador deve informar um endereço de e-mail válido;
- d) O Administrador deve informar número de CPF válido;
- e) O Administrador deve estar logado no sistema.
- f) O campo idCliente não pode ser alterado

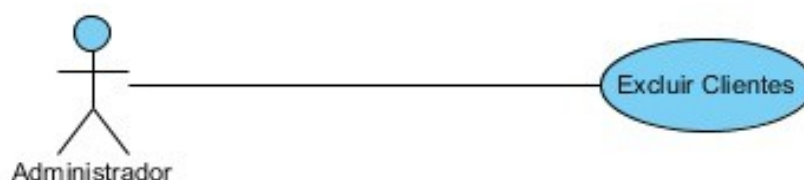


Figura 13: Caso de Uso Excluir Clientes

Caso de Uso: Excluir Clientes

Descrição: Esse caso de uso permite excluir um cliente.

Ator Principal: Administrador.

Pré-condições: O Administrador deve estar logado no sistema e ter acionado o menu de Clientes.

Pós-condições: O sistema exclui um cliente e atualiza tabela de clientes.

Fluxo Básico de Eventos

1. Administrador aciona link de Excluir na página de consultar clientes.
2. Sistema solicita confirmação de exclusão.
3. Administrador confirma exclusão de cliente.
4. Sistema exclui cadastro de cliente do banco de dados.
5. Sistema atualiza lista de clientes cadastrados.
6. Sistema atualiza página de consultar clientes.

Fluxo Alternativo – Administrador não confirma exclusão

1. Administrador aciona link de Excluir na página de consultar clientes.
2. Sistema solicita confirmação de exclusão.

3. Administrador não confirma exclusão de cliente.
4. Sistema fecha solicitação de confirmação.

Regras de Negócio

- a) O Administrador deve estar logado no sistema.



Figura 14: Caso de Uso Consultar Filmes

Caso de Uso: Consultar Filmes

Descrição: Esse caso de uso permite consultar os filmes cadastrados no sistema.

Ator Principal: Administrador.

Pré-condições: O Administrador precisa estar logado no sistema.

Pós-condições: O sistema redireciona para página de filmes com a lista dos filmes cadastrados no banco de dados.

Fluxo Básico de Eventos

1. Administrador aciona menu de Filmes.
2. Sistema busca filmes no banco de dados.
3. Sistema redireciona o Administrador para a página de filmes com a lista atualizada.

Regras de Negócio

- a) O Administrador deve estar logado no sistema.



Figura 15: Caso de Uso Incluir Filmes

Caso de Uso: Incluir Filmes

Descrição: Esse caso de uso permite cadastrar um filme.

Ator Principal: Administrador.

Pré-condições: O Administrador deve estar logado no sistema e ter acionado o menu de filmes.

Pós-condições: O sistema cadastra um novo filme e redireciona o Administrador para a página de consultar filmes .

Fluxo Básico de Eventos

1. Administrador aciona link de Inserir na página de consultar filmes.
2. Sistema redireciona o administrador para a página de incluir filmes.
3. Administrador informa os campos obrigatórios (Nome, Ano, Gênero, Data Início e Data Fim) e se quiser preenche os demais campos.
4. Administrador solicita cadastro das informações.
5. Sistema cadastra informações no banco de dados.
6. Sistema atualiza lista de filmes.
7. Sistema redireciona administrador para a página consultar filmes.

Fluxo Alternativo – Administrador não preenche todos os campos obrigatórios

1. Administrador não preenche todos os campos obrigatórios.
2. Administrador solicita cadastro das informações.
3. Sistema retorna mensagem de campo obrigatório faltante.

Fluxo Alternativo – Administrador solicita cancelamento

1. Administrador aciona botão voltar.

2. Sistema redireciona administrador para página de consultar filmes

Regras de Negócio

- a) Data Início deve ser menor que a Data Fim.
- b) Data Início deve ser maior ou igual a data atual do sistema.
- c) O Administrador deve estar logado no sistema.



Figura 16: Caso de Uso Alterar Filme

Caso de Uso: Alterar Filme.

Descrição: Esse caso de uso permite alterar um filme cadastrado.

Ator Principal: Administrador.

Pré-condições: O Administrador deve estar logado no sistema e ter acionado o menu de Filmes.

Pós-condições: O sistema altera um filme cadastrado e redireciona o Administrado para a página de consultar filmes .

Fluxo Básico de Eventos

1. Administrador aciona link de Alterar na página de consultar filmes.
2. Sistema redireciona o administrador para a página de alterar filmes.
3. Administrador altera a informação desejada.
4. Administrador aciona botão de salvar alterações.
5. Sistema atualiza informações no banco de dados.
6. Sistema atualiza lista de filmes.
7. Sistema redireciona administrador para a página consultar filmes.

Fluxo Alternativo – Administrador solicita cancelamento

1. Administrador aciona botão voltar.
2. Sistema redireciona administrador para página de consultar filmes.

Regras de Negócio

- a) Data Início deve ser menor que a Data Fim.
- b) Data Início deve ser maior ou igual a data atual do sistema.
- c) O Administrador deve estar logado no sistema.

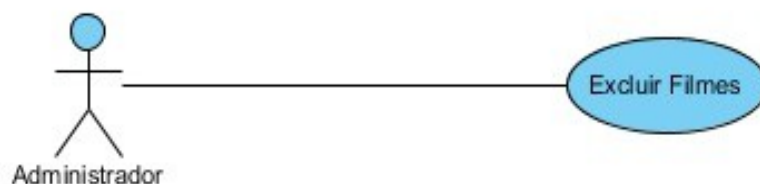


Figura 17: Caso de Uso Excluir Filmes

Caso de Uso: Excluir Filmes

Descrição: Esse caso de uso permite excluir um filme.

Ator Principal: Administrador.

Pré-condições: O Administrador deve estar logado no sistema e ter acionado o menu de Filmes.

Pós-condições: O sistema exclui um filme e atualiza tabela de filmes.

Fluxo Básico de Eventos

1. Administrador aciona link de Excluir na página de consultar filmes.
2. Sistema solicita confirmação de exclusão.
3. Administrador confirma exclusão de filme.
4. Sistema exclui cadastro de filme do banco de dados.
5. Sistema atualiza lista de filmes cadastrados.
6. Sistema atualiza página de consultar filmes.

Fluxo Alternativo – Administrador não confirma exclusão

1. Administrador aciona link de Excluir na página de consultar filmes.
2. Sistema solicita confirmação de exclusão.
3. Administrador não confirma exclusão de filme.
4. Sistema fecha solicitação de confirmação.

Regras de Negócio

- a) O Administrador deve estar logado no sistema.



Figura 18: Caso de Uso Consultar Gêneros

Caso de Uso: Consultar Gêneros.

Descrição: Esse caso de uso permite consultar os gêneros cadastrados no sistema.

Ator Principal: Administrador.

Pré-condições: O Administrador precisa estar logado no sistema.

Pós-condições: O sistema redireciona para página de gênero com a lista dos gêneros cadastrados no banco de dados.

Fluxo Básico de Eventos

1. Administrador aciona menu de Gêneros.
2. Sistema busca gêneros no banco de dados.
3. Sistema redireciona o Administrador para a página de gêneros com a lista atualizada.

Regras de Negócio

- a) O Administrador deve estar logado no sistema.



Figura 19: Caso de Uso Incluir Gêneros

Caso de Uso: Incluir Gêneros

Descrição: Esse caso de uso permite cadastrar um gênero.

Ator Principal: Administrador.

Pré-condições: O Administrador deve estar logado no sistema e ter acionado o menu de gênero.

Pós-condições: O sistema cadastra um novo gênero e redireciona o Administrador para a página de consultar gênero .

Fluxo Básico de Eventos

1. Administrador aciona link de Inserir na página de consultar gênero.
2. Sistema redireciona o administrador para a página de incluir gênero.
3. Administrador informa os campos obrigatórios (Nome e Descrição).
4. Administrador solicita cadastro das informações.
5. Sistema cadastra informações no banco de dados.
6. Sistema atualiza lista de gêneros.
7. Sistema redireciona administrador para a página consultar gêneros.

Fluxo Alternativo – Administrador não preenche todos os campos obrigatórios

1. Administrador não preenche todos os campos obrigatórios.
2. Administrador solicita cadastro das informações.
3. Sistema retorna mensagem de campo obrigatório faltante.

Fluxo Alternativo – Administrador solicita cancelamento

1. Administrador aciona botão voltar.
2. Sistema redireciona administrador para página de consultar gêneros.

Regras de Negócio

- a) O Administrador deve estar logado no sistema.



Figura 20: Caso de Uso Alterar Gêneros

Caso de Uso: Alterar Gêneros

Descrição: Esse caso de uso permite alterar um gênero cadastrado.

Ator Principal: Administrador.

Pré-condições: O Administrador deve estar logado no sistema e ter acionado o menu de Gênero.

Pós-condições: O sistema altera um gênero cadastrado e redireciona o Administrado para a página de consultar gêneros .

Fluxo Básico de Eventos

1. Administrador aciona link de Alterar na página de consultar gêneros.
2. Sistema redireciona o administrador para a página de alterar gêneros.
3. Administrador altera a informação desejada.
4. Administrador aciona botão de salvar alterações.
5. Sistema atualiza informações no banco de dados.
6. Sistema atualiza lista de gêneros.
7. Sistema redireciona administrador para a página consultar gêneros.

Fluxo Alternativo – Administrador solicita cancelamento

1. Administrador aciona botão voltar.
2. Sistema redireciona administrador para página de consultar gêneros.

Regras de Negócio

- a) O Administrador deve estar logado no sistema.



Figura 21: Caso de Uso Excluir Gêneros

Caso de Uso: Excluir Gêneros

Descrição: Esse caso de uso permite excluir um gênero.

Ator Principal: Administrador.

Pré-condições: O Administrador deve estar logado no sistema e ter acionado o menu de Gênero.

Pós-condições: O sistema exclui um gênero e atualiza tabela de gêneros.

Fluxo Básico de Eventos

1. Administrador aciona link de Excluir na página de consultar gêneros.
2. Sistema solicita confirmação de exclusão.
3. Administrador confirma exclusão de gênero.
4. Sistema exclui cadastro de gênero do banco de dados.
5. Sistema atualiza lista de gêneros cadastrados.
6. Sistema atualiza página de consultar gêneros.

Fluxo Alternativo – Administrador não confirma exclusão

1. Administrador aciona link de Excluir na página de consultar gêneros.
2. Sistema solicita confirmação de exclusão.
3. Administrador não confirma exclusão de gênero.
4. Sistema fecha solicitação de confirmação.

Regras de Negócio

- a) O Administrador deve estar logado no sistema.

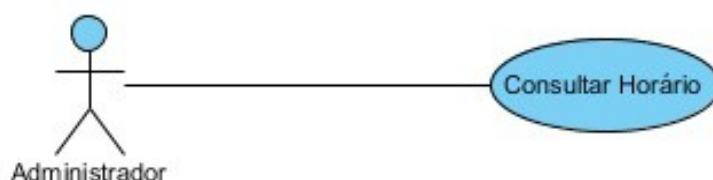


Figura 22: Caso de Uso Consultar Horário

Caso de Uso: Consultar Horário.

Descrição: Esse caso de uso permite consultar os horários cadastrados no sistema.

Ator Principal: Administrador.

Pré-condições: O Administrador precisa estar logado no sistema.

Pós-condições: O sistema redireciona para página de horários com a lista dos horários cadastrados no banco de dados.

Fluxo Básico de Eventos

1. Administrador aciona menu de Horário.

2. Sistema busca horários no banco de dados.
3. Sistema redireciona o Administrador para a página de horários com a lista atualizada.

Regras de Negócio

- a) O Administrador deve estar logado no sistema.



Figura 23: Caso de Uso Incluir Horário

Caso de Uso: Incluir Horário.

Descrição: Esse caso de uso permite cadastrar um horário.

Ator Principal: Administrador.

Pré-condições: O Administrador deve estar logado no sistema e ter acionado o menu de Horário.

Pós-condições: O sistema cadastra um novo horário e redireciona o Administrador para a página de consultar horários .

Fluxo Básico de Eventos

1. Administrador aciona link de Inserir na página de consultar horários.
2. Sistema redireciona o administrador para a página de incluir horários.
3. Administrador informa os campos obrigatórios (Horário, Sala, Filme, Data Início e Data Fim).
4. Administrador solicita cadastro das informações.
5. Sistema cadastra informações no banco de dados.
6. Sistema atualiza lista de horários.
7. Sistema redireciona administrador para a página consultar horários.

Fluxo Alternativo – Administrador não preenche todos os campos obrigatórios

1. Administrador não preenche todos os campos obrigatórios.

2. Administrador solicita cadastro das informações.
3. Sistema retorna mensagem de campo obrigatório faltante.

Fluxo Alternativo – Administrador solicita cancelamento

1. Administrador aciona botão voltar.
2. Sistema redireciona administrador para página de consultar horários

Regras de Negócio

- a) Data Início deve ser menor que a Data Fim.
- b) Data Início deve ser maior ou igual a data atual do sistema.
- c) Data Início e Data Fim devem estar entre a Data Início e Data Fim do Filme
- d) O Administrador deve estar logado no sistema.



Figura 24: Caso de Uso Alterar Horário

Caso de Uso: Alterar Horário.

Descrição: Esse caso de uso permite alterar um horário cadastrado.

Ator Principal: Administrador.

Pré-condições: O Administrador deve estar logado no sistema e ter acionado o menu de Horário.

Pós-condições: O sistema altera um horário cadastrado e redireciona o Administrado para a página de consultar horários .

Fluxo Básico de Eventos

1. Administrador aciona link de Alterar na página de consultar horários.
2. Sistema redireciona o administrador para a página de alterar horários.
3. Administrador altera a informação desejada.
4. Administrador aciona botão de salvar alterações.
5. Sistema atualiza informações no banco de dados.

6. Sistema atualiza lista de horários.
7. Sistema redireciona administrador para a página consultar horários.

Fluxo Alternativo – Administrador solicita cancelamento

1. Administrador aciona botão voltar.
2. Sistema redireciona administrador para página de consultar horários.

Regras de Negócio

- a) Data Início deve ser menor que a Data Fim.
- b) Data Início deve ser maior ou igual a data atual do sistema.
- c) Data Início e Data Fim devem estar entre a Data Início e Data Fim do Filme.
- d) O Administrador deve estar logado no sistema.



Figura 25: Caso de Uso Excluir Horário

Caso de Uso: Excluir Horário

Descrição: Esse caso de uso permite excluir um horário.

Ator Principal: Administrador.

Pré-condições: O Administrador deve estar logado no sistema e ter acionado o menu de Horário.

Pós-condições: O sistema exclui um horário e atualiza tabela de horário.

Fluxo Básico de Eventos

1. Administrador aciona link de Excluir na página de consultar horário.
2. Sistema solicita confirmação de exclusão.
3. Administrador confirma exclusão de horário.
4. Sistema exclui cadastro de horário do banco de dados.
5. Sistema atualiza lista de horários cadastrados.
6. Sistema atualiza página de consultar horários.

Fluxo Alternativo – Administrador não confirma exclusão

1. Administrador aciona link de Excluir na página de consultar horários.
2. Sistema solicita confirmação de exclusão.
3. Administrador não confirma exclusão de horário.
4. Sistema fecha solicitação de confirmação.

Regras de Negócio

- a) O Administrador deve estar logado no sistema.

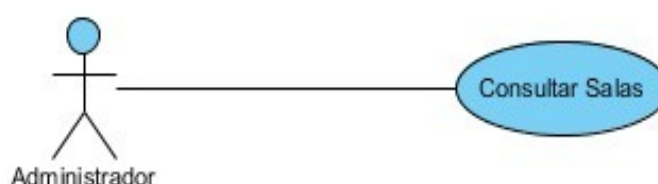


Figura 26: Caso de Uso Consultar Salas

Caso de Uso: Consultar Salas.

Descrição: Esse caso de uso permite consultar os salas cadastradas no sistema.

Ator Principal: Administrador.

Pré-condições: O Administrador precisa estar logado no sistema.

Pós-condições: O sistema redireciona para página de salas com a lista das salas cadastradas no banco de dados.

Fluxo Básico de Eventos

1. Administrador aciona menu de Salas.
2. Sistema busca Salas no banco de dados.
3. Sistema redireciona o Administrador para a página de salas com a lista atualizada.

Regras de Negócio

- a) O Administrador deve estar logado no sistema.



Figura 27: Caso de Uso Incluir Salas

Caso de Uso: Incluir Salas.

Descrição: Esse caso de uso permite cadastrar uma sala.

Ator Principal: Administrador.

Pré-condições: O Administrador deve estar logado no sistema e ter acionado o menu de Sala.

Pós-condições: O sistema cadastra uma nova sala e redireciona o Administrador para a página de consultar salas .

Fluxo Básico de Eventos

1. Administrador aciona link de Inserir na página de consultar salas.
2. Sistema redireciona o administrador para a página de incluir salas.
3. Administrador informa os campos obrigatórios (Número da Sala, Número de Poltronas, Tecnologia 3D, Está em Funcionamento).
4. Administrador solicita cadastro das informações.
5. Sistema cadastra informações no banco de dados.
6. Sistema atualiza lista de salas.
7. Sistema redireciona administrador para a página consultar salas.

Fluxo Alternativo – Administrador não preenche todos os campos obrigatórios

1. Administrador não preenche todos os campos obrigatórios.
2. Administrador solicita cadastro das informações.
3. Sistema retorna mensagem de campo obrigatório faltante.

Fluxo Alternativo – Administrador solicita cancelamento

1. Administrador aciona botão voltar.
2. Sistema redireciona administrador para página de consultar salas.

Regras de Negócio

- a) O Administrador deve estar logado no sistema.



Figura 28: Caso de Uso Alterar Salas

Caso de Uso: Alterar Salas.

Descrição: Esse caso de uso permite alterar uma sala cadastrada.

Ator Principal: Administrador.

Pré-condições: O Administrador deve estar logado no sistema e ter acionado o menu de Sala.

Pós-condições: O sistema altera uma sala cadastrada e redireciona o Administrado para a página de consultar salas .

Fluxo Básico de Eventos

1. Administrador aciona link de Alterar na página de consultar salas.
2. Sistema redireciona o administrador para a página de alterar salas.
3. Administrador altera a informação desejada.
4. Administrador aciona botão de salvar alterações.
5. Sistema atualiza informações no banco de dados.
6. Sistema atualiza lista de salas.
7. Sistema redireciona administrador para a página consultar salas.

Fluxo Alternativo – Administrador solicita cancelamento

1. Administrador aciona botão voltar.
2. Sistema redireciona administrador para página de consultar salas.

Regras de Negócio

- a) O Administrador deve estar logado no sistema.

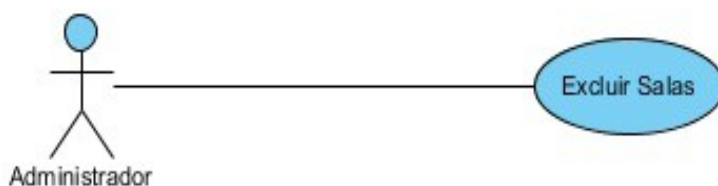


Figura 29: Caso de Uso Excluir Salas

Caso de Uso: Excluir Salas.

Descrição: Esse caso de uso permite excluir uma sala.

Ator Principal: Administrador.

Pré-condições: O Administrador deve estar logado no sistema e ter acionado o menu de Sala.

Pós-condições: O sistema exclui uma sala e atualiza tabela de sala.

Fluxo Básico de Eventos

1. Administrador aciona link de Excluir na página de consultar sala.
2. Sistema solicita confirmação de exclusão.
3. Administrador confirma exclusão de sala.
4. Sistema exclui cadastro de sala do banco de dados.
5. Sistema atualiza lista de salas cadastradas.
6. Sistema atualiza página de consultar salas.

Fluxo Alternativo – Administrador não confirma exclusão

1. Administrador aciona link de Excluir na página de consultar salas.
2. Sistema solicita confirmação de exclusão.
3. Administrador não confirma exclusão de sala.
4. Sistema fecha solicitação de confirmação.

Regras de Negócio

- a) O Administrador deve estar logado no sistema

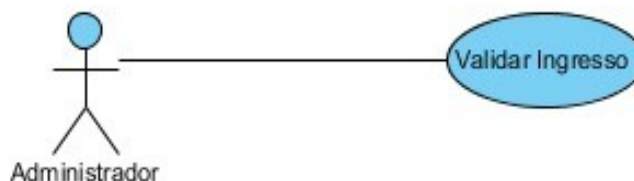


Figura 30: Caso de Uso Validar Ingresso

Caso de Uso: Validar Ingresso.

Descrição: Esse caso de uso permite validar um ingresso comprado pelo site ou pelo aplicativo.

Ator Principal: Administrador.

Pré-condições: O Administrador deve estar logado no sistema e ter acionado o menu de Validar Sala.

Pós-condições: O Sistema marca ingresso como utilizado e envia mensagem de sucesso.

Fluxo Básico de Eventos

1. Administrador digita o código do ingresso a ser validado.
2. Sistema verifica se o código é válido
3. Sistema verifica se a data e horário do ingresso estão corretos.
4. Sistema marca ingresso como utilizado.
5. Sistema manda mensagem de ingresso validado com sucesso.

Fluxo Alternativo – Administrador informa código incorreto

1. Administrador digita o código inválido do ingresso.
2. Sistema verifica se o código é válido
3. Sistema envia mensagem informando que o código é inválido.

Regras de Negócio

- a) O Administrador deve estar logado no sistema

Os Diagramas de Caso de Uso a seguir representam os casos de uso do aplicativo móvel do cliente

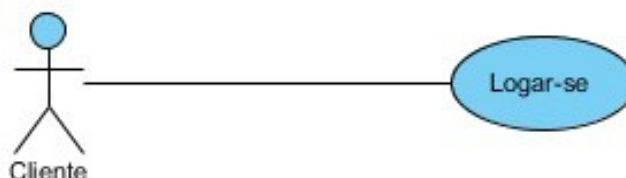


Figura 31: Caso de uso Logar-se - Aplicativo Móvel Cliente

Caso de Uso: Logar-se.

Descrição: Esse caso de uso permite ao usuário fazer o login no sistema.

Ator Principal: Cliente.

Pré-condições: Não existe pré-condições para esse caso de uso.

Pós-condições: O sistema realiza o login do usuário e redireciona para a tela de instruções de uso.

Fluxo Básico de Eventos

1. Cliente informa o E-mail e Senha.
2. Cliente solicita o Login.
3. Sistema verifica informações enviadas.
4. Sistema realiza login no sistema.
5. Sistema redireciona usuário para a tela de instruções de uso.
6. Sistema aciona câmera do celular para digitalização.

Fluxo Alternativo – Cliente Informa campo errado

1. Cliente informa campo e-mail ou campo senha incorretamente.
2. Cliente solicita login no sistema.
3. Sistema retorna mensagem de campo incorreto.

Regras de Negócio

- a) O Cliente só poderá realizar o login se informado campo login e senha corretamente, respeitando caracteres maiúsculos e minúsculos.

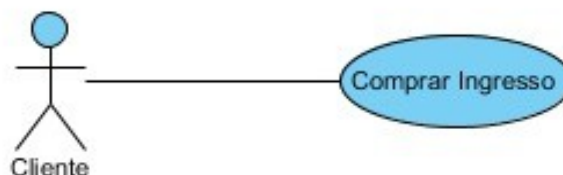


Figura 32: Caso de Uso Comprar Ingresso

Caso de uso: Comprar ingresso.

Descrição: O Cliente usa a Câmera do celular para digitalizar uma imagem QR-Code que representa o horário de exibição de um filme.

Ator Principal: Cliente

Ator de Suporte: Sistema Web

Pré-condições: O cinema deve ter disponibilizado a imagem a ser digitalizada. O cliente deve estar logado no aplicativo móvel.

Pós-condições: Realiza a compra do ingresso no cadastro do cliente para o filme escolhido.

Fluxo Básico:

1. O Cliente digitaliza a imagem com a câmera.
2. O Aplicativo faz a requisição de compra de ingresso para o Sistema Web.
3. Sistema Web verifica créditos promocionais do cliente, se não possuir o sistema verifica crédito do cliente.
4. Sistema Web debita valor do ingresso do crédito do cliente.
5. Sistema Web incrementa um ingresso ao cadastro do cliente.
6. Sistema Web verifica se ingressos comprados é múltiplo de 10 e se for inclui um ingresso promocional ao cadastro do cliente.
7. Sistema Web realiza cadastro do ingresso no banco de dados.
8. O Sistema Web envia um e-mail com imagem do QR-Code do ingresso em anexo.
9. Sistema Web retorna mensagem de sucesso para o aplicativo.
10. O Aplicativo trata a mensagem e exibe a tela de sucesso.

Fluxo Alternativo – imagem errada

- 1 O imagem digitalizada pode não se tratar de um filme.
- 2 O Sistema exibe uma tela explicando o que ocorreu.
- 3 O Sistema Web não envia e-mail.

Regras de Negócio:

- a) O código a ser digitalizado deve ser representar um filme segunda as regras do cinema.

Os Casos de Uso a seguir representam os casos de uso do aplicativo móvel cinema.

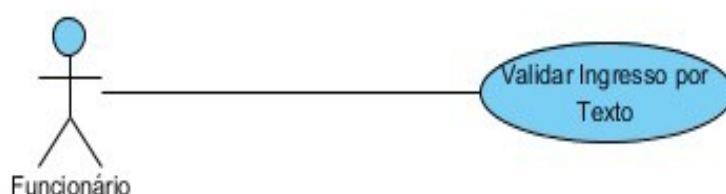


Figura 33: Caso de Uso Validar Ingresso por Texto

Caso de uso: Validar Ingresso por Texto.

Descrição: O Funcionário usa teclado do celular para digitar o código apresentado pelo cliente.

Ator Principal: Funcionário

Pré-condições: Cliente apresenta o código texto na entrada do cinema.

Pós-condições: Valida o ingresso do cliente e libera a entrada.

Fluxo Básico:

1. Funcionário digita o código
2. O Aplicativo envia o código para o Sistema Web.
3. Sistema Web verifica se o código é válido
4. Sistema Web verifica se a data e horário do ingresso estão corretos.
5. Sistema marca ingresso como utilizado.
6. O Sistema manda mensagem de ingresso validado com sucesso para o Aplicativo.

7. O Aplicativo trata a mensagem recebida.
8. O Aplicativo exibe a tela de liberação da sala com informações sobre o filme, sala e horário.

Fluxo Alternativo – Código texto inválido

1. Funcionário digita o código
2. O Aplicativo envia o código para o Sistema Web.
3. Sistema Web verifica que o código é inválido
4. O Sistema Web retorna mensagem de erro ao Aplicativo.
5. Aplicativo trata a mensagem e retorna para a tela do celular.

Regras de Negócio:

- a) O código-ingresso apresentado pelo cliente deve ser um ingresso válido.

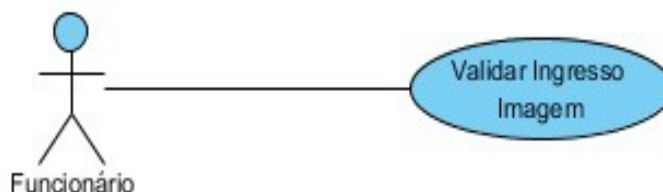


Figura 34: Caso de Uso Validar Ingresso por Imagem

Caso de uso: Validar ingresso por Imagem.

Descrição: O Funcionário usa a Câmera do celular para digitalizar o código imagem apresentado pelo cliente.

Ator Principal: Funcionário

Pré-condições: Cliente apresenta o código imagem na entrada.

Pós-condições: Valida o ingresso do cliente e libera a entrada.

Fluxo Básico:

1. Funcionário digitaliza a imagem com a câmera
2. O Aplicativo decodifica a imagem.
3. O Aplicativo envia o código da imagem para o Sistema Web.
4. Sistema Web verifica se o código é válido

5. Sistema Web verifica se a data e horário do ingresso estão corretos.
6. Sistema marca ingresso como utilizado.
7. O Sistema manda mensagem de ingresso validado com sucesso para o Aplicativo.
8. O Aplicativo trata a mensagem recebida.
9. O Aplicativo exibe a tela de liberação da sala com informações sobre o filme, sala e horário.

Fluxo Alternativo – imagem inválida

1. Funcionário digitaliza a imagem.
2. O Aplicativo decodifica a imagem.
3. O Aplicativo envia a imagem decodificada para o Sistema Web.
4. Sistema Web verifica que o código é inválido
5. O Sistema Web retorna mensagem de erro ao Aplicativo.
6. Aplicativo trata a mensagem e retorna para a tela do celular.

Regras de Negócio:

- a) O código-ingresso apresentado pelo cliente deve ser um ingresso válido.

APÊNDICE B – Detalhamento dos Casos de Uso

Este apêndice contém as classes omitidas no corpo principal do projeto.

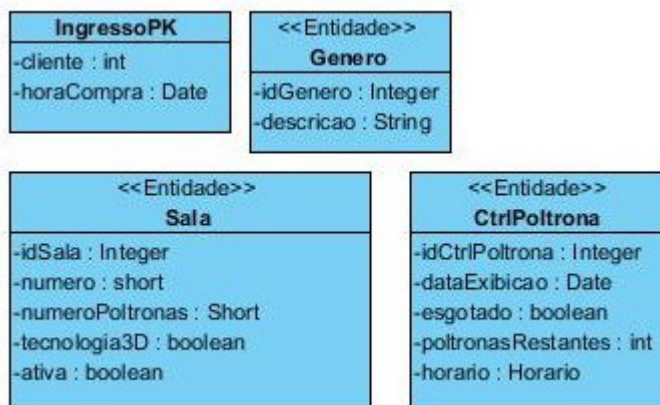


Figura 1: Classes de entidades

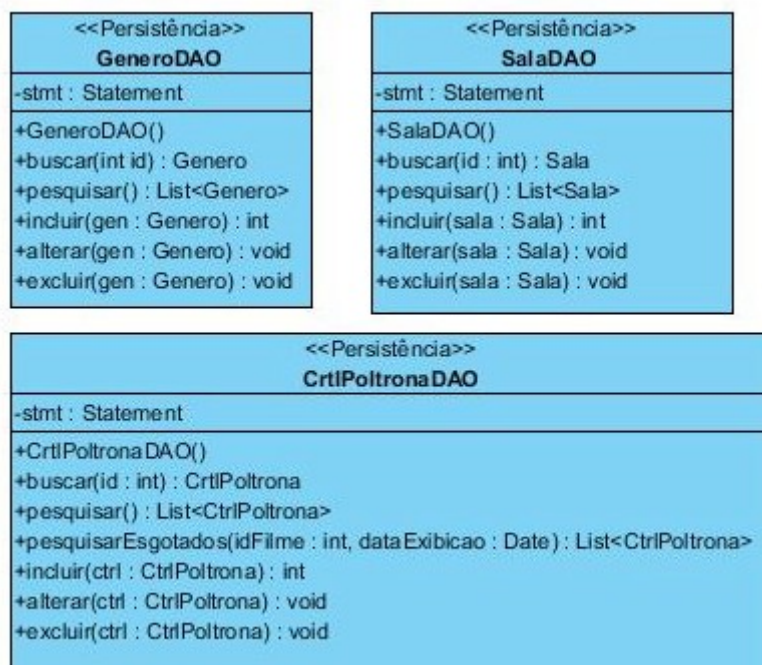


Figura 2: Classes de Persistência

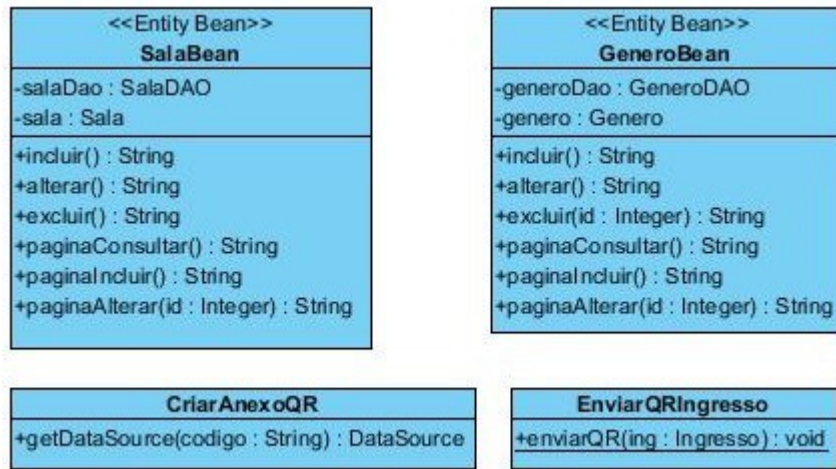


Figura 3: Classes Beans

APÊNDICE C – Detalhamento dos Dicionários de Informações

Este apêndice contém os dicionários de informações do projeto.

As tabelas 1, 2, 3, 4, 5, 6, 7 e 8 são referentes as classes de entidade

Tabela 1: Dicionário da Classe Cliente

Classe	Descrição
Cliente	Classe que representa a tabela cliente do banco de dados.
Atributos	Descrição
idCliente	Código que identifica o cliente no banco de dados.
cpf	CPF do cliente cadastrado.
nome	Nome do cliente cadastrado.
senha	Senha de acesso do cliente aos sistemas.
telefone	Número de telefone do cliente.
dataNascimento	Data de nascimento do cliente.
dataCadastro	Data em que foi efetuado o cadastro do cliente.
credito	Número BigDecimal que representa o saldo do cliente no sistema.
ingressoPromocional	Quantidade de ingressos promocionais que o cliente possui.
ingressosVendidos	Quantidade total de ingressos vendidos para esse cliente.
Métodos	Descrição
<i>Getters e Setters</i>	A Classe possui um método <i>Get</i> e um método <i>Set</i> para cada atributo da classe.

Tabela 2: Dicionário da Classe Filme

Classe	Descrição
Filme	Classe que representa a tabela de filme do banco de dados.
Atributos	Descrição
idFilme	Código que identifica o filme no banco de dados.
nome	Nome do filme cadastrado.
ano	Ano do filme cadastrado.
duracao	Tempo de duração do filme em minutos.
sinopse	Sinopse do filme cadastrado.
classificacao	Classificação etária do filme.
dataInicio	Data que o filme começará a ser exibido no cinema.
dataFim	Data do último dia de exibição do filme.
genero	Instância da classe Gênero que classifica o filme.
Métodos	Descrição
<i>Getters e Setters</i>	A classe possui um método <i>Get</i> e um método <i>Set</i> para cada atributo da classe.

Tabela 3: Dicionário da Classe Horário

Classe	Descrição
Horario	Classe que representa a tabela horário no banco de dados
Atributos	Descrição
idHorario	Código que identifica o horário no banco de dados
horario	Horário que o filme será exibido.
dataInicio	Primeiro dia que o filme será exibido neste horário.
dataFim	Último dia que o filme será exibido neste horário.
sala	Instância da classe Sala que representa a sala do cinema em que o filme será exibido nesse horário.
filme	Instância da classe Filme que representa o filme a ser exibido neste horário.
Métodos	Descrição
<i>Getters e Setters</i>	A classe possui um método <i>Get</i> e um método <i>Set</i> para cada atributo da classe.

Tabela 4: Dicionário da Classe Ingresso

Classe	Descrição
Ingresso	Classe que representa a tabela ingresso do banco de dados
Atributos	Descrição
ingressoPK	Instância da classe IngressoPK que representa as chaves compostas do ingresso no banco de dados
dataExibicao	Data de exibição do ingresso.
dataUtilizacao	Data que o ingresso foi utilizado.
utilizado	<i>Boolean</i> que indica se o ingresso foi ou não utilizado (<i>true</i> = sim/ <i>false</i> = não)
horario	Instância da classe horário que representa o horário de exibição do filme.
Métodos	Descrição
<i>Getters e Setters</i>	A classe possui um método <i>Get</i> e um método <i>Set</i> para cada atributo da classe.

Tabela 5: Dicionário da Classe Gênero

Classe	Descrição
Genero	Classe que representa a tabela gênero no banco de dados
Atributos	Descrição
idGenero	Código que identifica o gênero no banco de dados
descricao	Descrição do gênero de classificação do filme.
Métodos	Descrição
<i>Getter e Setters</i>	A classe possui um método <i>Get</i> e um método <i>Set</i> para cada atributo da classe.

Tabela 6: Dicionário da Classe Sala

Classe	Descrição
Sala	Classe que representa a tabela sala no banco de dados.
Atributos	Descrição
idSala	Código que identifica a sala no banco de dados.
numero	Número da sala do cinema.
numeroPoltronas	Número de poltronas que a sala possui.
tecnologia3D	<i>Boolean</i> que indica se a sala possui ou não tecnologia 3d (<i>true</i> = sim / <i>false</i> = não)
ativa	<i>Boolean</i> que indica se a sala está ativa.
Métodos	Descrição
<i>Getter e Setters</i>	A classe possui um método <i>Get</i> e um método <i>Set</i> para cada atributo da classe.

Tabela 7: Dicionário da Classe IngressoPK

Classe	Descrição
IngressoPK	Classe que representa a chave composta da tabela ingresso no banco de dados.
Atributos	Descrição
cliente	Código do cliente à que pertence o ingresso.
horaCompra	Horário em que foi comprado o ingresso.
Métodos	Descrição
<i>Getters e Setters</i>	A classe possui um método <i>Get</i> e um método <i>Set</i> para cada atributo da classe.

Tabela 8: Dicionario da Classe CtrlPoltrona

Classe	Descrição
CtrlPoltrona	Classe que representa a tabela ctrlPoltrona no banco de dados. Essa tabela foi criada para ter um controle de poltronas do horário do filme exibido, para que não fossem vendidos mais ingressos do que a sala pudesse suportar.
Atributos	Descrição
idCtrlPoltrona	Código que identifica um ctrlPoltrona no banco de dados.
dataExibicao	Data da exibição desse controle de poltronas.
esgotado	<i>Boolean</i> que indica se os ingressos para esse horário desse filme estão esgotados.
poltronasRestantes	Número de poltronas restantes para venda de ingresso.
horario	Instância da classe horário que representa o horário em que será exibido o filme
Métodos	Descrição
<i>Getters e Setters</i>	A classe possui um método <i>Get</i> e um método <i>Set</i> para cada atributo da classe.

As tabelas 9, 10, 11, 12, 13, 14 e 15 são referentes as classes de persistência.

Tabela 9: Dicionário da Classe ClienteDAO

Classe	Descrição
ClienteDAO	Classe de controle de acesso da tabela cliente. Responsável por gerenciar os dados do cliente no banco de dados.
Atributos	Descrição
stmt	Atributo da classe <i>Statement</i> , responsável pela execução das <i>queries</i> no banco de dados.
Métodos	Descrição
ClienteDAO()	Construtor da classe ClienteDAO(), responsável por iniciar o atributo <i>statement</i> .
buscar(id:int)	Método responsável por executar a <i>query</i> de busca de um cliente pelo idCliente.
pesquisar()	Método responsável por executar a <i>query</i> de busca de todos os clientes do banco de dados.
pesquisarEmail(email:String)	Método responsável por executar a <i>query</i> de busca de cliente pelo e-mail.
pesquisarCPF(cpf:String)	Método responsável por executar a <i>query</i> de busca de cliente por CPF.
incluir(cli:Cliente)	Método responsável por executar a <i>query</i> de inclusão de um cliente conforme os parâmetros passados.
alterar(cli:Cliente)	Método responsável por executar a <i>query</i> de atualização de um cliente conforme os parâmetros passados.
excluir(cli:Cliente)	Método responsável por executar a <i>query</i> de exclusão de um cliente conforme os parâmetros passados.

Tabela 10: Dicionário da Classe FilmeDAO

Classe	Descrição
FilmeDAO	Classe de controle de acesso da tabela filme. Responsável por gerenciar os dados do filme no banco de dados.
Atributos	Descrição
stmt	Atributo da classe <i>Statement</i> , responsável pela execução das <i>querys</i> no banco de dados.
Métodos	Descrição
FilmeDAO()	Construtor da classe FilmeDAO(), responsável por iniciar o atributo <i>statement</i> .
buscar(id:int)	Método responsável por executar a <i>query</i> de busca de um filme pelo idFilme.
pesquisar()	Método responsável por executar a <i>query</i> de busca de todos os filmes do banco de dados.
pesquisarCartaz()	Método responsável por executar a <i>query</i> de busca de todos os filmes em exibição do banco de dados.
pesquisarEntre(data:Date)	Método responsável por executar a <i>query</i> de busca de todos os filmes que estejam em exibição na data especificada por parâmetro.
incluir(fil:Filme)	Método responsável por executar a <i>query</i> de inclusão de um filme conforme os parâmetros passados.
alterar(fil:Filme)	Método responsável por executar a <i>query</i> de atualização de um filme conforme os parâmetros passados.
excluir(fil:Filme)	Método responsável por executar a <i>query</i> de exclusão de um filme conforme os parâmetros passados.

Tabela 11: Dicionário da Classe HorarioDAO

Classe	Descrição
HorarioDAO	Classe de controle de acesso da tabela horário. Responsável por gerenciar os dados do horário no banco de dados.
Atributos	Descrição
stmt	Atributo da classe <i>Statement</i> , responsável pela execução das <i>queries</i> no banco de dados.
Métodos	Descrição
HorarioDAO()	Construtor da classe HorarioDAO(), responsável por iniciar o atributo <i>statement</i> .
buscar(id:int)	Método responsável por executar a <i>query</i> de busca de um horário pelo idHorario.
pesquisar()	Método responsável por executar a <i>query</i> de busca de todos os horários do banco de dados.
pesquisarFilme(fil:Filme)	Método responsável por executar a <i>query</i> de busca de todos os horários de um determinado filme.
pesquisarFilmeData(data:Date)	Método responsável por executar a <i>query</i> de busca de todos os horários de um determinado filme em uma certa data.
incluir(hor:Horario)	Método responsável por executar a <i>query</i> de inclusão de um horário conforme os parâmetros passados.
alterar(hor:Horario)	Método responsável por executar a <i>query</i> de atualização de um horário conforme os parâmetros passados.
excluir(hor:Horario)	Método responsável por executar a <i>query</i> de exclusão de um horário conforme os parâmetros passados.

Tabela 12: Dicionário da Classe IngressoDAO

Classe	Descrição
IngressoDAO	Classe de controle de acesso da tabela ingresso. Responsável por gerenciar os dados do ingresso no banco de dados.
Atributos	Descrição
stmt	Atributo da classe <i>Statement</i> , responsável pela execução das <i>querys</i> no banco de dados.
Métodos	Descrição
IngressoDAO()	Construtor da classe IngressoDAO(), responsável por iniciar o atributo <i>statement</i> .
buscar(id:int)	Método responsável por executar a <i>query</i> de busca de um ingresso pelo idIngresso.
pesquisar()	Método responsável por executar a <i>query</i> de busca de todos os ingressos do banco de dados.
pesquisarCliente(cli:Cliente)	Método responsável por executar a <i>query</i> de busca de todos os ingressos de um determinado cliente.
incluir(ing:Ingresso)	Método responsável por executar a <i>query</i> de inclusão de um ingresso conforme os parâmetros passados.
alterar(ing:Ingresso)	Método responsável por executar a <i>query</i> de atualização de um ingresso conforme os parâmetros passados.
excluir(ing:Ingresso)	Método responsável por executar a <i>query</i> de exclusão de um ingresso conforme os parâmetros passados.

Tabela 13: Dicionário da Classe GeneroDAO

Classe	Descrição
GeneroDAO	Classe de controle de acesso da tabela gênero. Responsável por gerenciar os dados do gênero no banco de dados.
Atributos	Descrição
stmt	Atributo da classe <i>Statement</i> , responsável pela execução das <i>queries</i> no banco de dados.
Métodos	Descrição
GeneroDAO()	Construtor da classe GeneroDAO(), responsável por iniciar o atributo <i>statement</i> .
buscar(id:int)	Método responsável por executar a <i>query</i> de busca de um gênero pelo idGenero.
pesquisar()	Método responsável por executar a <i>query</i> de busca de todos os gêneros do banco de dados.
incluir(gen:Genero)	Método responsável por executar a <i>query</i> de inclusão de um gênero conforme os parâmetros passados.
alterar(gen:Genero)	Método responsável por executar a <i>query</i> de atualização de um gênero conforme os parâmetros passados.
excluir(gen:Genero)	Método responsável por executar a <i>query</i> de exclusão de um gênero conforme os parâmetros passados.

Tabela 14: Dicionário da Classe SalaDAO

Classe	Descrição
SalaDAO	Classe de controle de acesso da tabela sala. Responsável por gerenciar os dados da sala no banco de dados.
Atributos	Descrição
stmt	Atributo da classe <i>Statement</i> , responsável pela execução das <i>queries</i> no banco de dados.
Métodos	Descrição
SalaDAO()	Construtor da classe SalaDAO(), responsável por iniciar o atributo <i>statement</i> .
buscar(id:int)	Método responsável por executar a <i>query</i> de busca de uma sala pelo idSala.
pesquisar()	Método responsável por executar a <i>query</i> de busca de todas as salas do banco de dados.
incluir(sala:Sala)	Método responsável por executar a <i>query</i> de inclusão de uma sala conforme os parâmetros passados.
alterar(sala:Sala)	Método responsável por executar a <i>query</i> de atualização de uma sala conforme os parâmetros passados.
excluir(sala:Sala)	Método responsável por executar a <i>query</i> de exclusão de uma sala conforme os parâmetros passados.

Tabela 15: Dicionário da Classe CtrlPoltronaDAO

Classe	Descrição
CtrlPoltronaDAO	Classe de controle de acesso da tabela ctrlPoltrona. Responsável por gerenciar os dados de controle de poltronas no banco de dados.
Atributos	Descrição
stmt	Atributo da classe <i>Statement</i> , responsável pela execução das <i>queries</i> no banco de dados.
Métodos	Descrição
CtrlPoltronaDAO()	Construtor da classe CtrlPoltronaDAO(), responsável por iniciar o atributo <i>statement</i> .
buscar(id:int)	Método responsável por executar a <i>query</i> de busca de um ctrlPoltrona pelo idCtrlPoltrona.
pesquisar()	Método responsável por executar a <i>query</i> de busca de todos os controles de poltronas do banco de dados.
pesquisarEsgotados(idFilme:int, dataExibicao:Date)	Método responsável por executar a <i>query</i> de busca de todos os controles de poltronas esgotados do banco de dados.
incluir(ctrl:CtrlPoltrona)	Método responsável por executar a <i>query</i> de inclusão de um controle de poltrona conforme os parâmetros passados.
alterar(ctrl:CtrlPoltrona)	Método responsável por executar a <i>query</i> de atualização de um controle de poltrona conforme os parâmetros passados.
excluir(ctrl:CtrlPoltrona)	Método responsável por executar a <i>query</i> de exclusão de um controle de poltrona conforme os parâmetros passados.

As tabelas 16, 17 e 18 são referentes as classes servlet.

Tabela 16: Dicionário da Classe ComprarIngresso

Classe	Descrição
ComprarIngresso	Herda da Classe HttpServlet por tratar requisições Http. Esse <i>servlet</i> é responsável por responder à requisição de compra de ingressos pelo aplicativo móvel cliente.
Atributos	Descrição
Nenhum	Nenhum atributo relevante para o projeto do sistema. Apenas os herdados pela classe.
Método	Descrição
doGet(request:HttpServletRequest, response: HttpServletResponse)	Retorna mensagem de passagem de parâmetros incorreta.
doPost(request:HttpServletRequest, response: HttpServletResponse)	Realiza a compra de um ingresso com o idCliente e idHorario passados por parâmetro. Esse método realiza todas as validações necessárias para verificar a possibilidade de compra do ingresso pelo cliente. Retorna mensagem de sucesso ou erro ao aplicativo móvel cliente.

Tabela 17: Dicionário da Classe RealizarLogin

Classe	Descrição
RealizarLogin	Herda da Classe HttpServlet por tratar requisições Http. Esse <i>servlet</i> é responsável por responder à requisição de realizar login pelo aplicativo móvel cliente.
Atributos	Descrição
Nenhum	Nenhum atributo relevante para o projeto do sistema. Apenas os herdados pela classe.
Método	Descrição
doGet(request:HttpServletRequest, response: HttpServletResponse)	Retorna mensagem de passagem de parâmetros incorreta.
doPost(request:HttpServletRequest, response: HttpServletResponse)	Verifica se o e-mail e senha do cliente passados por parâmetros existem e são pertencentes ao mesmo usuário. Retorna mensagem de sucesso ou erro ao aplicativo móvel cliente.

Tabela 18: Dicionário da Classe ValidarIngresso

Classe	Descrição
ValidarIngresso	Herda da Classe HttpServlet por tratar requisições Http. Esse <i>servlet</i> é responsável por responder à requisição de validar ingresso realizado pelo aplicativo móvel cinema.
Atributos	Descrição
Nenhum	Nenhum atributo relevante para o projeto do sistema. Apenas os herdados pela classe.
Método	Descrição
doGet(request:HttpServletRequest, response: HttpServletResponse)	Retorna mensagem de passagem de parâmetros incorreta.
doPost(request:HttpServletRequest, response: HttpServletResponse)	Verifica se o ingresso é um ingresso válido, e se o ingresso pode ser validado. Retorna mensagem de sucesso ou erro ao aplicativo móvel cinema.

As tabelas 19, 20, 21, 22, 23 e 24 descrevem as classes beans do sistema web.

Tabela 19: Dicionário da Classe CreditoBean

Classe	Descrição
CreditoBean	É a classe responsável por incluir créditos no cadastro de um cliente.
Atributos	Descrição
clienteDao	Instância da classe ClienteDAO responsável por acionar os métodos da mesma.
cliente	Instância da classe Cliente responsável por receber e exibir os créditos do cliente na tela do sistema.
Métodos	Descrição
verificaCliente()	Método responsável por pegar os dados do cliente.
comprar()	Realiza validações para os campos preenchidos. Invoca o método alterar da classe ClienteDAO com o novo crédito, preenche mensagem de erro ou sucesso e redireciona o cliente para tela correta.
paginaComprar()	Método de redirecionamento para a página de comprar crédito.

Tabela 20: Dicionário da Classe ClienteBean

Classe	Descrição
ClienteBean	É a classe intermediadora entre as telas do sistema e a classe ClienteDAO que realiza as operações no banco de dados, essa classe também é responsável pelo redirecionamento de telas.
Atributos	Descrição
clienteDao	Instância da classe ClienteDAO responsável por acionar os métodos da mesma.
cliente	Instância da classe Cliente responsável por receber e exibir os dados do cliente na tela do sistema.
Métodos	Descrição
incluir()	Realiza validações para os campos preenchidos. Invoca o método incluir da classe ClienteDAO, preenche mensagem de erro ou sucesso e redireciona o cliente para tela correta.
alterar()	Realiza validações para os campos preenchidos. Invoca o método alterar da classe ClienteDAO, preenche mensagem de erro ou sucesso e redireciona o cliente para tela correta.
excluir(id:Integer)	Invoca o método excluir da classe ClienteDAO com o id como parâmetro, preenche mensagem de erro ou sucesso e redireciona o cliente para tela correta.
paginaConsultar()	Método de redirecionamento para a página consultar cliente.
paginaIncluir()	Método de redirecionamento para a página de incluir cliente.
paginaAlterar(id:Integer)	Método de redirecionamento para a página de alterar cliente com os campos do cliente já preenchidos.

Tabela 21: Dicionário da classe FilmeBean

Classe	Descrição
FilmeBean	É a classe intermediadora entre as telas do sistema e a classe FilmeDAO que realiza as operações no banco de dados, essa classe também é responsável pelo redirecionamento de telas.
Atributos	Descrição
filmeDao	Instância da classe FilmeDAO responsável por acionar os métodos da mesma.
filme	Instância da classe Filme responsável por receber e exibir os dados do filme na tela do sistema.
Métodos	Descrição
incluir()	Realiza validações para os campos preenchidos. Invoca o método incluir da classe FilmeDAO, preenche mensagem de erro ou sucesso e redireciona o cliente para tela correta.
alterar()	Realiza validações para os campos preenchidos. Invoca o método alterar da classe FilmeDAO, preenche mensagem de erro ou sucesso e redireciona o cliente para tela correta.
excluir(id:Integer)	Invoca o método excluir da classe FilmeDAO com o id como parâmetro, preenche mensagem de erro ou sucesso e redireciona o cliente para tela correta.
validaDatas()	Método que faz as validações das datas para serem inseridas no banco de dados do filme.
paginaConsultar()	Método de redirecionamento para a página consultar filme.
paginaIncluir()	Método de redirecionamento para a página de incluir filme.
paginaAlterar(id:Integer)	Método de redirecionamento para a página de alterar filme com os campos do filme já preenchidos.

Tabela 22: Dicionário da Classe HorárioBean

Classe	Descrição
HorarioBean	É a classe intermediadora entre as telas do sistema e a classe HorárioDAO que realiza as operações no banco de dados, essa classe também é responsável pelo redirecionamento de telas.
Atributos	Descrição
horarioDao	Instância da classe HorárioDAO responsável por acionar os métodos da mesma.
horario	Instância da classe Horário responsável por receber e exibir os dados do horário na tela do sistema.
Métodos	Descrição
incluir()	Realiza validações para os campos preenchidos. Invoca o método incluir da classe HorárioDAO, preenche mensagem de erro ou sucesso e redireciona o cliente para tela correta.
alterar()	Realiza validações para os campos preenchidos. Invoca o método alterar da classe HorárioDAO, preenche mensagem de erro ou sucesso e redireciona o cliente para tela correta.
excluir(id:Integer)	Invoca o método excluir da classe HorárioDAO com o id como parâmetro, preenche mensagem de erro ou sucesso e redireciona o cliente para tela correta.
paginaConsultar()	Método de redirecionamento para a página de consultar horário.
paginaIncluir()	Método de redirecionamento para a página de incluir horário.
paginaAlterar(id:Integer)	Método de redirecionamento para a página de alterar horário com os campos do horário já preenchidos.

Tabela 23: Dicionário da Classe IngressoBean

Classe	Descrição
IngressoBean	É a classe intermediadora entre as telas do sistema e a classe IngressoBean que realiza as operações no banco de dados, essa classe também é responsável pelo redirecionamento de telas.
Atributos	Descrição
ingressoDao	Instância da classe IngressoDAO responsável por acionar os métodos da mesma.
ingresso	Instância da classe Ingresso responsável por receber e exibir os dados do ingresso na tela do sistema.
filmeDao	Instância da classe FilmeDAO responsável por acionar os métodos da mesma.
filme	Instância da classe Filme responsável por receber e exibir os dados do ingresso na tela do sistema.
ctrlPoltronaDao	Instância da classe CtrlPoltronaDAO responsável por acionar os métodos da mesma.
ctrlPoltrona	Instância da classe CtrlPoltrona utilizada nos métodos de validação da classe.
Métodos	Descrição
incluir()	Realiza validações para os campos preenchidos. Invoca o método incluir da classe IngressoDAO, preenche mensagem de erro ou sucesso e redireciona o cliente para tela correta.
alterar()	Realiza validações para os campos preenchidos. Invoca o método alterar da classe IngressoDAO, preenche mensagem de erro ou sucesso e redireciona o cliente para tela correta.
excluir(id:Integer)	Invoca o método excluir da classe IngressoDAO com o id como parâmetro, preenche mensagem de erro ou sucesso e redireciona o cliente para tela correta.
validar()	Realiza validações para o código do ingresso, altera o ingresso para utilizado e redireciona o cliente para a tela correta.
validarPoltrona(horario:Horario, data:Date)	Decrementa o número de poltronas disponíveis do ctrlPoltrona.
validaCompra()	Realiza as validações para verificar se a compra pode ser realizada. Retorna <i>true</i> se o cliente puder realizar a compra.
preencheHorario()	Método que realiza o preenchimento da lista de horários para um filme em uma data especificada.
preencheFilme()	Método que realiza o preenchimento da lista de filmes para uma data especificada.
paginaConsultar()	Método de redirecionamento para a página consultar ingressos.
paginaComprar()	Método de redirecionamento para a página de comprar ingresso.
paginaValidar()	Método de redirecionamento para a página de validar ingresso.

Tabela 24: Dicionário da Classe LoginBean

Classe	Descrição
LoginBean	É a classe responsável por logar um cliente no sistema web.
Atributos	Descrição
contexto	Atributo da classe FacesContext responsável por receber a sessão atual do sistema.
session	Atributo da classe HttpSession responsável por manter e atribuir valores a sessão.
email	<i>String</i> responsável por receber e exibir dados da tela do sistema.
senha	<i>String</i> responsável por receber a senha da tela do sistema.
clienteDao	Instância da classe ClienteDAO responsável por acionar os métodos da mesma.
cliente	Instância da classe Cliente responsável por receber e exibir dados do cliente na tela do sistema.
Métodos	Descrição
LoginBean()	Método responsável por iniciar os dados na sessão como deslogado.
efetuarLogin()	Verifica se os dados informados estão corretos para realizar o login do usuário e insere os dados do usuário na sessão do sistema.

As tabelas 25, 26, 27 e 28 descrevem as classes utilizadas pelos aplicativos móveis.

Tabela 25: Dicionário da Classe GeneroBean

Classe	Descrição
GeneroBean	É a classe intermediadora entre as telas do sistema e a classe GeneroDAO que realiza as operações no banco de dados, essa classe também é responsável pelo redirecionamento de telas.
Atributos	Descrição
generoDao	Instância da classe GeneroDAO responsável por acionar os métodos da mesma.
genero	Instância da classe Genero responsável por receber e exibir os dados do gênero na tela do sistema.
Métodos	Descrição
incluir()	Realiza validações para os campos preenchidos. Invoca o método incluir da classe GeneroDAO, preenche mensagem de erro ou sucesso e redireciona o cliente para tela correta.
alterar()	Realiza validações para os campos preenchidos. Invoca o método alterar da classe GeneroDAO, preenche mensagem de erro ou sucesso e redireciona o cliente para tela correta.
excluir(id:Integer)	Invoca o método excluir da classe GeneroDAO com o id como parâmetro, preenche mensagem de erro ou sucesso e redireciona o cliente para tela correta.
paginaConsultar()	Método de redirecionamento para a página de consultar gênero.
paginaIncluir()	Método de redirecionamento para a página de incluir gênero.
paginaAlterar(id:Integer)	Método de redirecionamento para a página de alterar gênero com os campos do gênero já preenchidos.

Tabela 26: Dicionário da Classe SalaBean

Classe	Descrição
SalaBean	É a classe intermediadora entre as telas do sistema e a classe SalaDAO que realiza as operações no banco de dados, essa classe também é responsável pelo redirecionamento de telas.
Atributos	Descrição
salaDao	Instância da classe SalaDAO responsável por acionar os métodos da mesma.
sala	Instância da classe Sala responsável por receber e exibir os dados da sala na tela do sistema.
Métodos	Descrição
incluir()	Realiza validações para os campos preenchidos. Invoca o método incluir da classe SalaDAO, preenche mensagem de erro ou sucesso e redireciona o cliente para tela correta.
alterar()	Realiza validações para os campos preenchidos. Invoca o método alterar da classe SalaDAO, preenche mensagem de erro ou sucesso e redireciona o cliente para tela correta.
excluir(id:Integer)	Invoca o método excluir da classe SalaDAO com o id como parâmetro, preenche mensagem de erro ou sucesso e redireciona o cliente para tela correta.
paginaConsultar()	Método de redirecionamento para a página de consultar sala.
paginaIncluir()	Método de redirecionamento para a página de incluir sala.
paginaAlterar(id:Integer)	Método de redirecionamento para a página de alterar sala com os campos da sala já preenchidos.

Tabela 27: Dicionário da Classe CriarAnexoQR

Classe	Descrição
CriarAnexoQR	É a classe responsável por criar uma imagem <i>QR-Code</i> a partir do código do ingresso e converte a imagem em um <i>DataSource</i> para o anexo.
Atributos	Descrição
Nenhum	A classe não possui nenhum atributo.
Métodos	Descrição
getDataSource(codigo:String)	Método estático responsável por criar uma imagem <i>QR-Code</i> a partir do código do ingresso e converte a imagem em um <i>DataSource</i> para o anexo.

Tabela 28: Dicionário da Classe EnviarQRIngresso

Classe	Descrição
EnviarQRIngresso	É a classe responsável por enviar um e-mail contendo uma imagem de <i>QR-Code</i> de ingresso como anexo.
Atributos	Descrição
Nenhum	A classe não possui nenhum atributo.
Métodos	Descrição
enviarQR(ing:Ingresso)	Método estático responsável por enviar um e-mail com anexo. Se utiliza do método <code>getDataSource(codigo:String)</code> da classe <code>CriarAnexoQR</code> .

A seguir estão listados as tabelas de dicionário de informação para cada classe do aplicativo móvel cliente.

Tabela 29: Dicionário da Classe ICineActivity

Classe	Descrição
ICineActivity	Herda da classe android.app.Activity por tratar eventos de controle de tela (arquitetura Android). Essa <i>Activity</i> é responsável pelo controle de eventos e chamadas das outras telas da aplicação e tratar seus resultados.
Atributos	Descrição
Nenhum	Nenhum atributo relevante para o projeto do sistema. Apenas os herdados pela classe..
Métodos	Descrição
chamarTelaLogin()	Inicia a Tela de Login.
chamarTelaHowto()	Inicia a Tela de Instruções.
chamarCamera()	Chama a tela digitalizar código da biblioteca ZXing.
chamarTelaCompra()	Inicia a Tela de confirmação de compra.
chamarTelaErroCompra(mensagem: String)	Inicia a Tela de Mensagem de erro. A mensagem a ser exibida é passado como parâmetro.
tratarResultadoTela(resultCode:int)	Método responsável por tratar os códigos de retorno das outras telas da aplicação e decidir qual tela chamar na sequência.

Tabela 30: Dicionário da Classe LoginActivity

Classe	Descrição
LoginActivity	Herda da classe android.app.Activity por tratar eventos de controle de tela (arquitetura Android). Essa <i>Activity</i> apenas recebe os dados e os retorna para a ICineActivity.
Atributos	Descrição
Nenhum	Nenhum atributo relevante para o projeto do sistema. Apenas os herdados pela classe e referências aos botões e campos da Tela.
Métodos	Descrição
Nenhum	Nenhum método relevante para o projeto do sistema. Apenas os herdados pela classe que fazem parte do ciclo de vida da <i>Activity</i> e de controle de eventos do botão.

Tabela 31: Dicionário da Classe HowtoActivity

Classe	Descrição
HowtoActivity	Herda da classe android.app.Activity por tratar eventos de controle de tela (arquitetura Android). Essa <i>Activity</i> apenas exibe instruções de uso e retorna para a ICineActivity após 5 segundos.
Atributos	Descrição
Nenhum	Nenhum atributo relevante para o projeto do sistema. Apenas os herdados pela classe.
Métodos	Descrição
Nenhum	Nenhum método relevante para o projeto do sistema. Apenas os herdados pela classe que fazem parte do ciclo de vida da <i>Activity</i>

Tabela 32: Dicionário da Classe ErroCompraActivity

Classe	Descrição
ErroCompraActivity	Herda da classe android.app.Activity por tratar eventos de controle de tela (arquitetura Android). Essa <i>Activity</i> apenas exibe os motivos pelos quais não foi possível realizar a compra.
Atributos	Descrição
Nenhum	Nenhum atributo relevante para o projeto do sistema. Apenas os herdados pela classe.
Métodos	Descrição
Nenhum	Nenhum método relevante para o projeto do sistema. Apenas os herdados pela classe que fazem parte do ciclo de vida da <i>Activity</i> .

Tabela 33: Dicionário da Classe CompraActivity

Classe	Descrição
CompraActivity	Herda da classe android.app.Activity por tratar eventos de controle de tela (arquitetura Android). Essa <i>Activity</i> apenas exibe mensagem de sucesso da compra e informa o usuário que seu ingresso será enviado por e-mail.
Atributos	Descrição
Nenhum	Nenhum atributo relevante para o projeto do sistema. Apenas os herdados pela classe.
Métodos	Descrição
Nenhum	Nenhum método relevante para o projeto do sistema. Apenas os herdados pela classe que fazem parte do ciclo de vida da <i>Activity</i> .

Tabela 34: Dicionário da Classe Cliente

Classe	Descrição
Cliente	Armazena temporariamente dados da conta do cliente para realizar compra de ingressos. Apenas uma única instância dessa classe será usada pelo sistema (<i>Singleton</i>).
Atributos	Descrição
email	Armazena o e-mail digitado pelo cliente para informações de login.
senha	Armazena a senha digitada pelo cliente para informações de login.
idCliente	Armazena o ID do cliente recuperado quando o aplicativo faz requisição de login ao sistema web do cinema.
instance	Uma instância dessa classe que será a única usada pelo sistema (<i>Singleton</i>).
Métodos	Descrição
getInstance()	Recupera a instância única dessa classe para uso do programa. Se não houver instância dessa classe, uma nova será criada.
<i>getters</i> e <i>setters</i>	Métodos de acesso aos campos privados.

Tabela 35: Dicionário da Classe Conexao

Classe	Descrição
Conexao	Essa classe mantém as variáveis usadas para a conexão com o sistema web do cinema.
Atributos	Descrição
URL	Endereço do servidor na internet é usado tanto para realizar o login quanto para a compra.
LOGIN	Caminho de contexto para o <i>servlet</i> de login. Usado juntamente com a URL.
COMPRA	Caminho de contexto para o <i>servlet</i> de compra de ingressos. Usado juntamente com a URL.
Métodos	Descrição
autenticarCliente()	Usa as informações da classe Cliente para realizar o login no sistema web do cinema.
comprarIngresso(String idHorario)	Realiza a compra do ingresso para o cliente em questão ao fazer uma requisição ao sistema web do cinema.

A seguir estão listados os tabelas de dicionário de informação para cada classe do aplicativo móvel cinema.

Tabela 36: Dicionário da Classe MenuActivity

Classe	Descrição
MenuActivity	Herda da classe android.app.Activity por tratar eventos de controle de tela (arquitetura Android). Essa <i>Activity</i> é responsável pelo controle de eventos e chamadas das outras telas da aplicação e tratar seus resultados.
Atributos	Descrição
Nenhum	Nenhum atributo relevante para o projeto do sistema. Apenas os herdados pela classe e referências aos botões da Tela.
Métodos	Descrição
chamarTelaEntrarTexto()	Inicia a Tela de digitação manual do código.
chamarTelaResultado(code:int)	Inicia a Tela de Resultado da validação do ingresso.
chamarCamera()	Chama a tela digitalizar código da biblioteca ZXing.
tratarResultadoTela(resultCode:int)	Método responsável por tratar os códigos de retorno das outras telas da aplicação e decidir qual tela chamar na sequência.

Tabela 37: Dicionário da Classe EnterTextActivity

Classe	Descrição
EnterTextActivity	Herda da classe android.app.Activity por tratar eventos de controle de tela (arquitetura Android). Essa <i>Activity</i> apenas recebe os dados e os retorna para a MenuActivity.
Atributos	Descrição
Nenhum	Nenhum atributo relevante para o projeto do sistema. Apenas os herdados pela classe e referências aos botões e campos da Tela.
Métodos	Descrição
Nenhum	Nenhum método relevante para o projeto do sistema. Apenas os herdados pela classe que fazem parte do ciclo de vida da <i>Activity</i> e de controle de eventos do botão.

Tabela 38: Dicionário da Classe ResultadoActivity

Classe	Descrição
ResultadoActivity	Herda da classe android.app.Activity por tratar eventos de controle de tela (arquitetura Android). Essa <i>Activity</i> apenas exibe mensagem de sucesso da compra e informa ao funcionário se o ingresso foi validado ou não para a liberação da entrada para o cliente.
Atributos	Descrição
Nenhum	Nenhum atributo relevante para o projeto do sistema. Apenas os herdados pela classe.
Métodos	Descrição
Nenhum	Nenhum método relevante para o projeto do sistema. Apenas os herdados pela classe que fazem parte do ciclo de vida da <i>Activity</i> .

Tabela 39: Dicionário da Classe Filme

Classe	Descrição
Filme	Armazena temporariamente dados do filme do ingresso para exibir caso a liberação ocorra. Apenas uma única instância dessa classe será usada pelo sistema (<i>Singleton</i>).
Atributos	Descrição
nomeFilme	Armazena o nome do filme.
sala	Armazena a sala na qual o filme será exibido nesse horário.
horario	Armazena o horário de exibição do filme.
instance	Uma instância dessa classe que será a única usada pelo sistema (<i>Singleton</i>).
Métodos	Descrição
getInstance()	Recupera a instância única dessa classe para uso do programa. Se não houver instância dessa classe, uma nova será criada.
<i>getters e setters</i>	Métodos de acesso aos campos privados.

Tabela 40: Dicionário da Classe Conexao

Classe	Descrição
Conexao	Essa classe mantém as variáveis usadas para a conexão com o sistema <i>web</i> do cinema.
Atributos	Descrição
URL	Endereço do servidor na internet é usado para realizar a validação do ingresso.
Métodos	Descrição
autenticarIngresso(code:String)	Usa as informações recuperadas do ingresso e faz requisição ao sistema <i>web</i> a fim de verificar a autenticidade do ingresso.

APÊNDICE D – Diagramas de Sequência

A seguir serão apresentados os diagramas de sequência do sistema web.

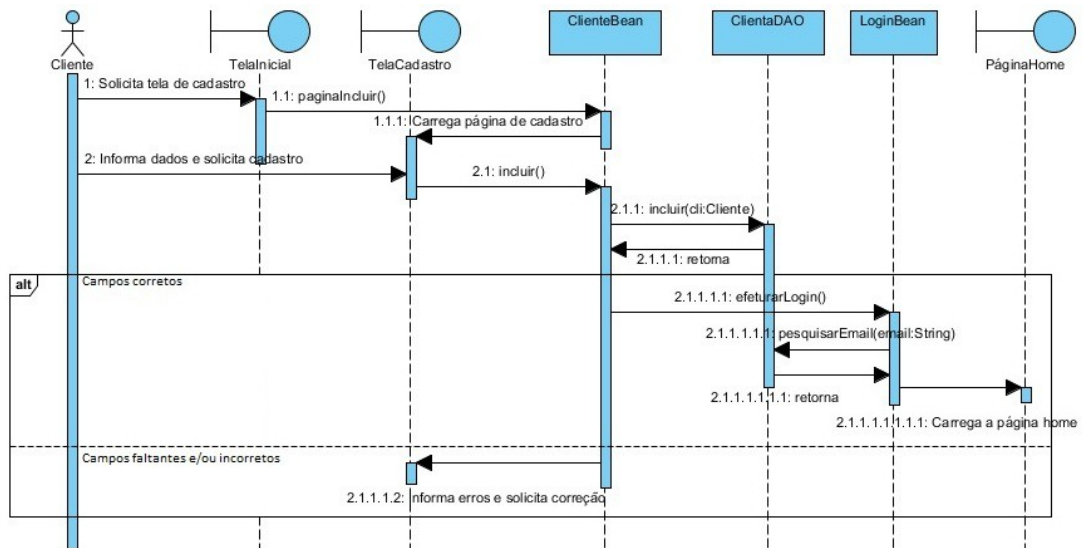


Figura 1: Diagrama de Sequência Registrar-se

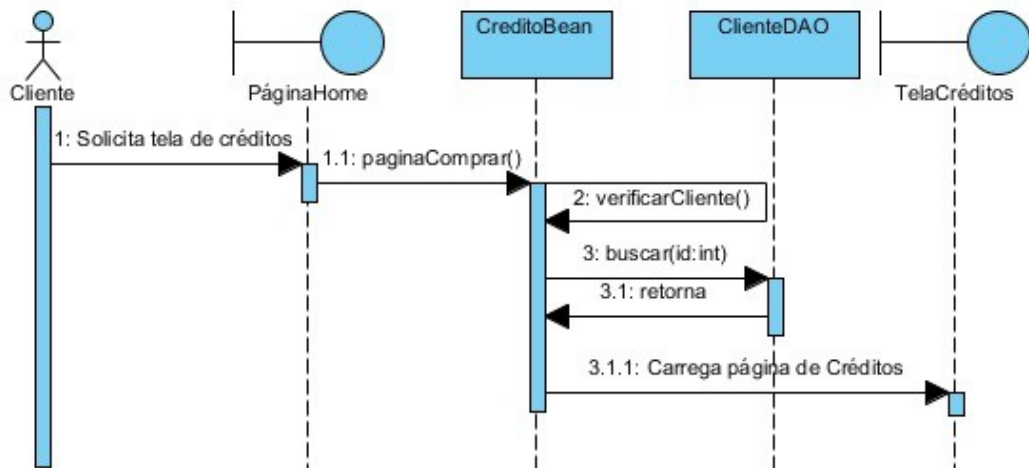


Figura 2: Diagrama de Sequência Visualizar Créditos

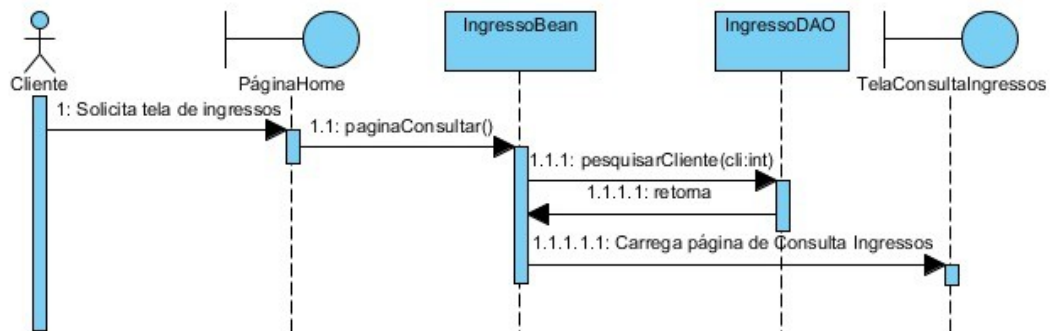


Figura 3: Diagrama de Sequência Visualizar Ingressos

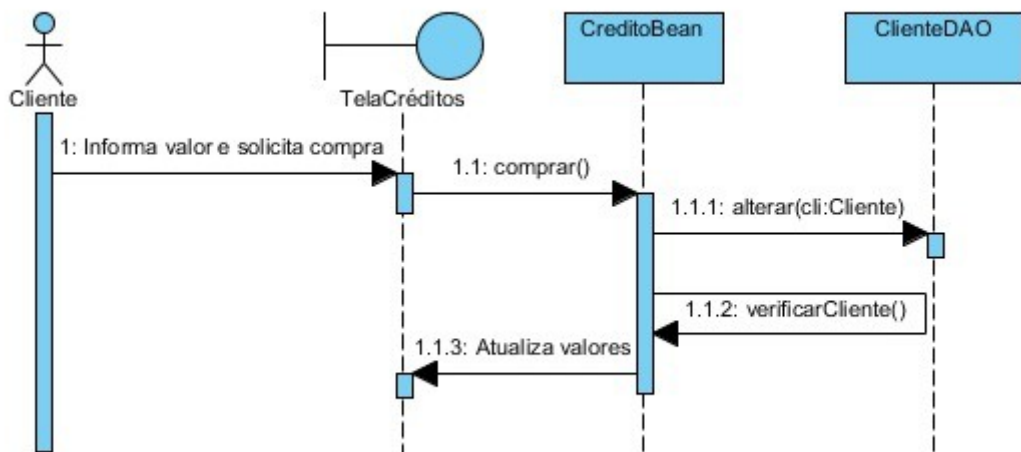


Figura 4: Diagrama de Sequência Comprar Créditos

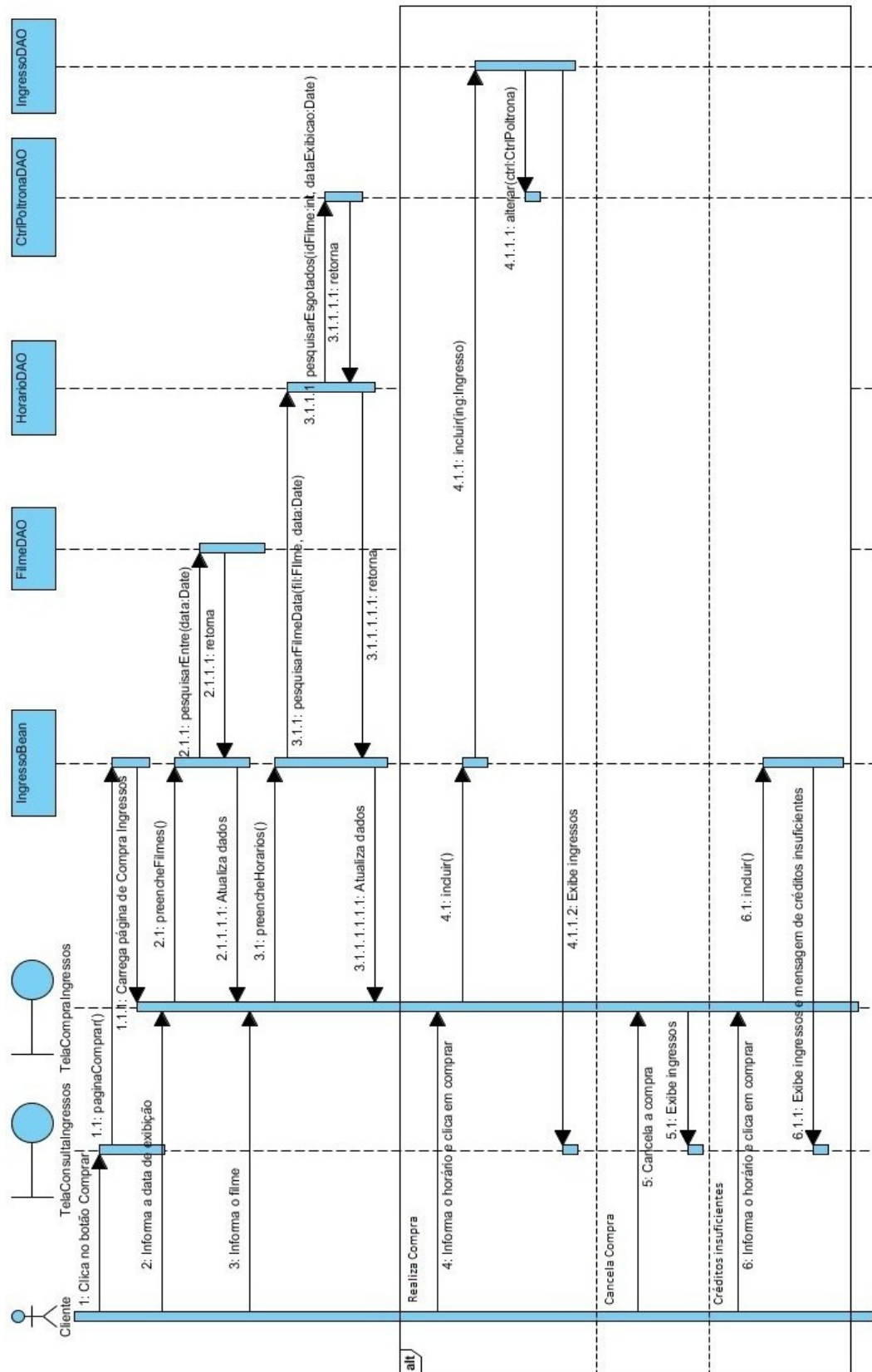


Figura 5: Diagrama de Sequência Comprar Ingresso

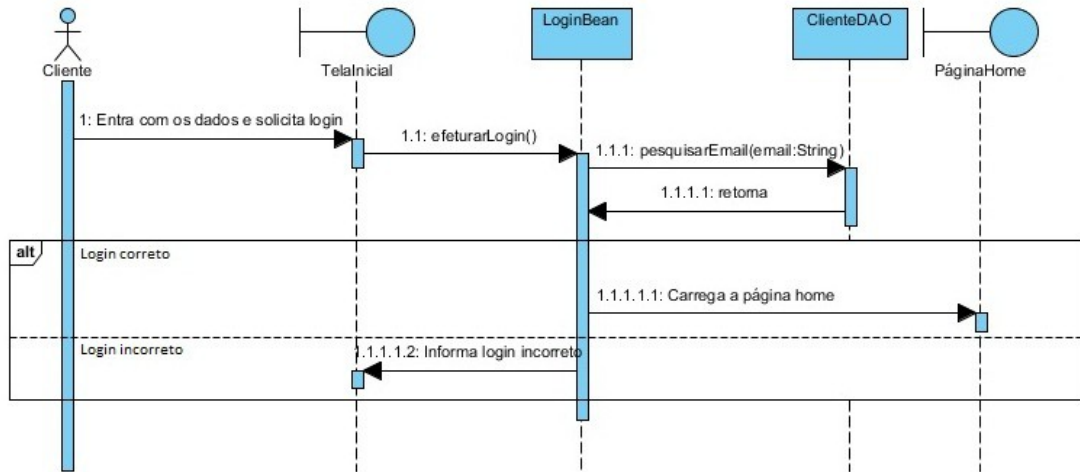


Figura 6: Diagrama de sequência Logar-se Web (Cliente)

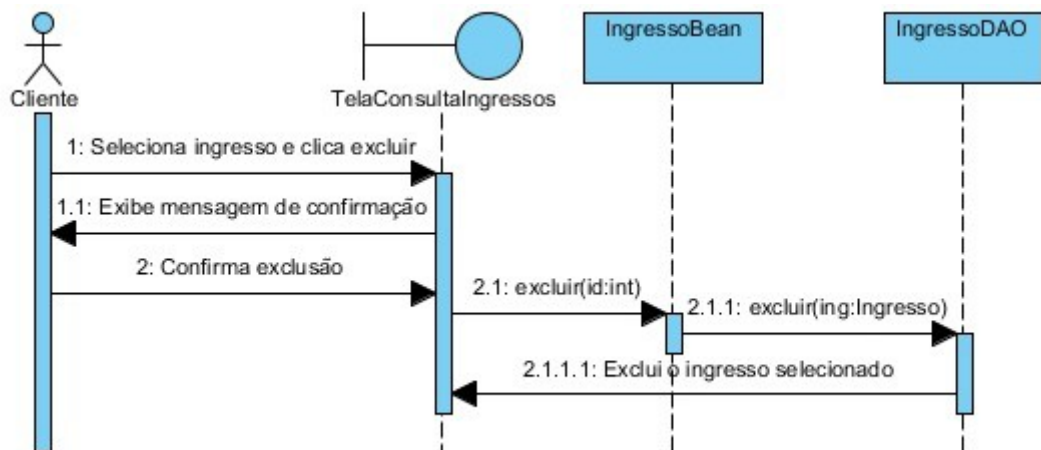


Figura 7: Diagrama de Sequências Excluir Ingresso

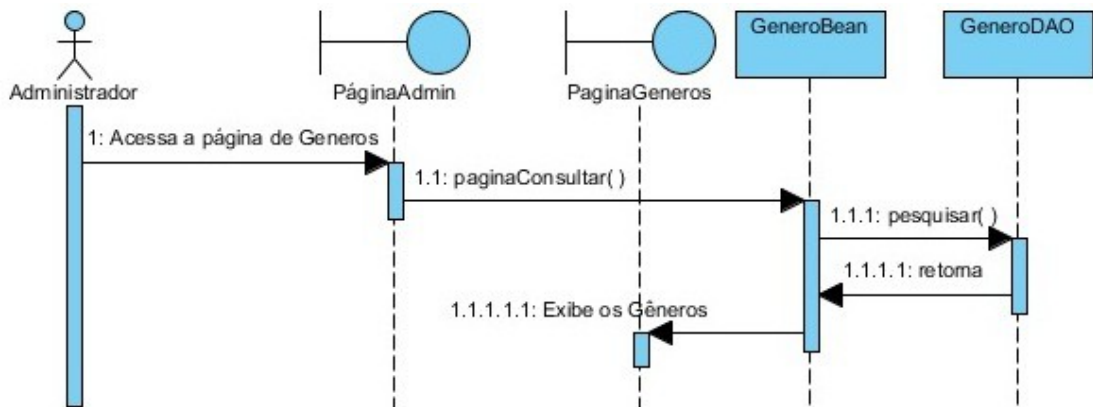


Figura 8: Diagrama de Sequências Consultar Gêneros

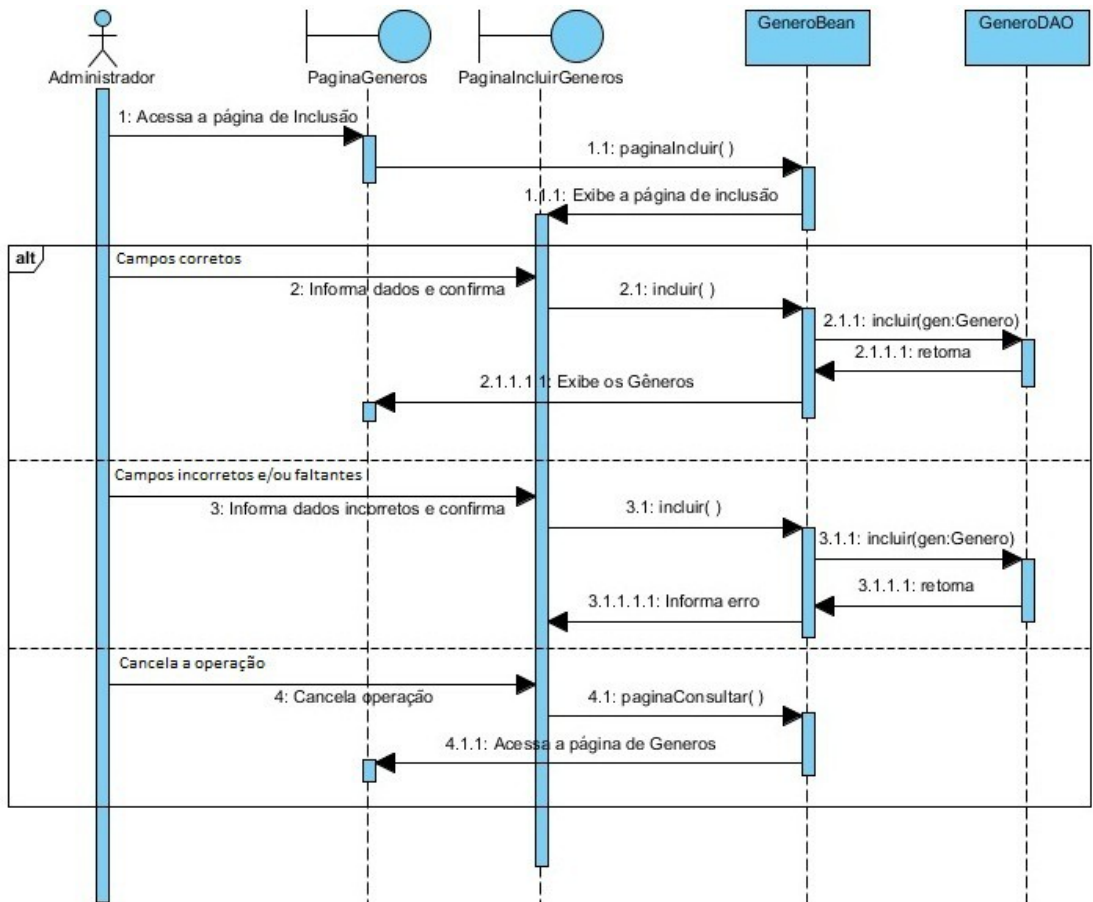


Figura 9: Diagrama de Sequências Incluir Gêneros

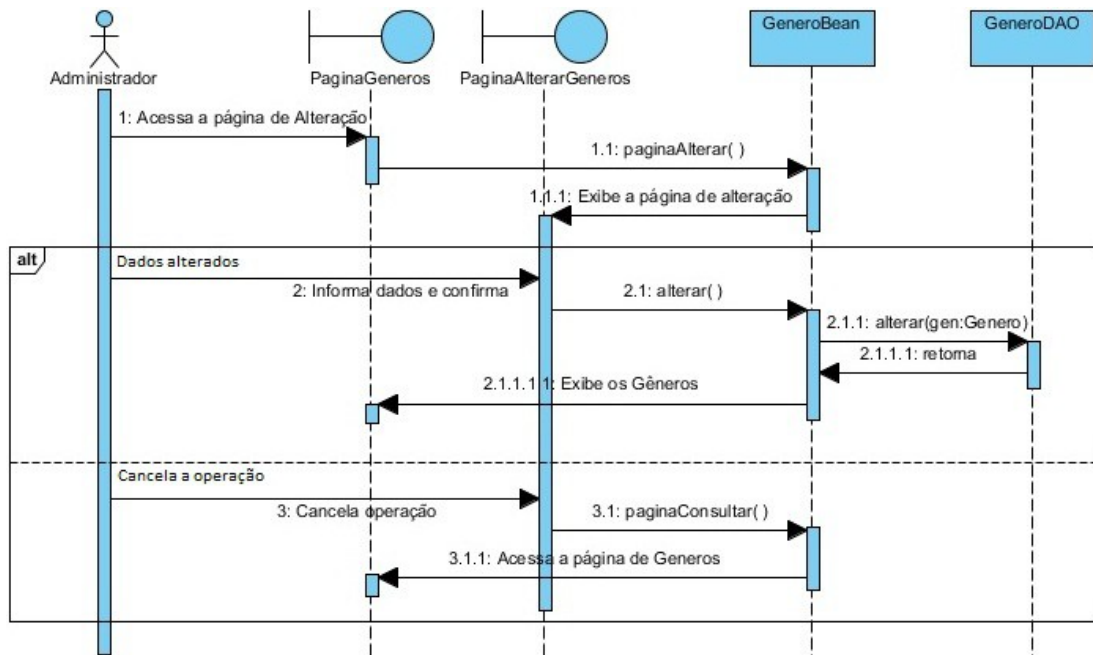


Figura 10: Diagrama de Sequências Alterar Gêneros

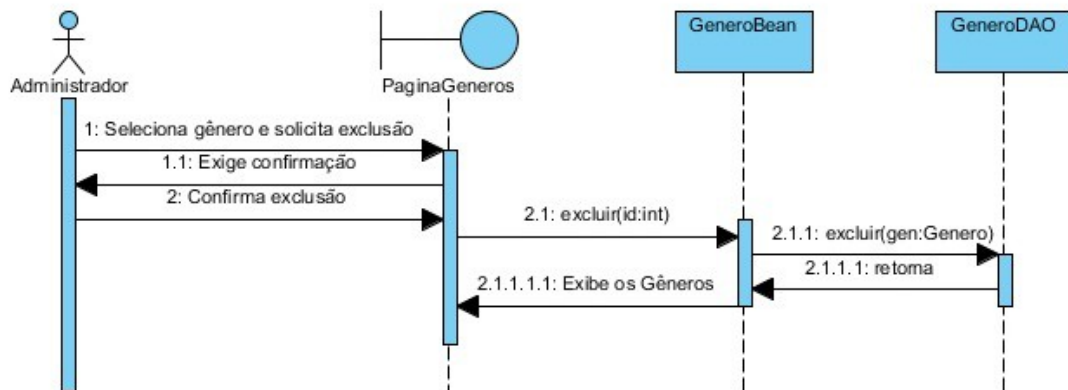


Figura 11: Diagrama de Sequências Excluir Gêneros

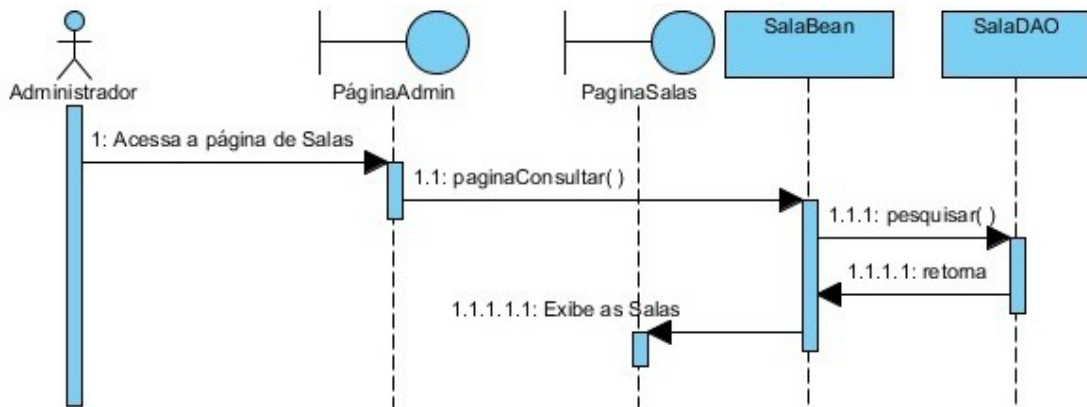


Tabela 12: Diagrama de Sequências Consultar Sala

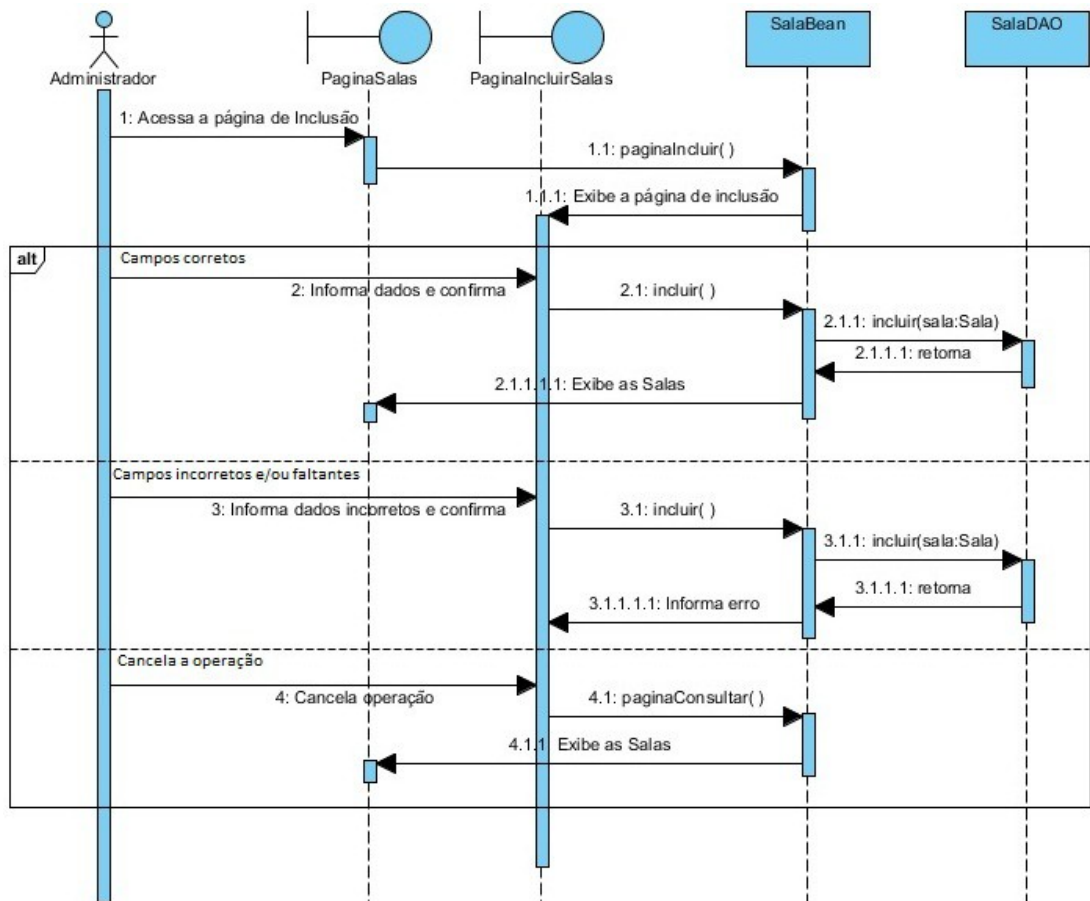


Figura 13: Diagrama de Sequências Incluir Salas

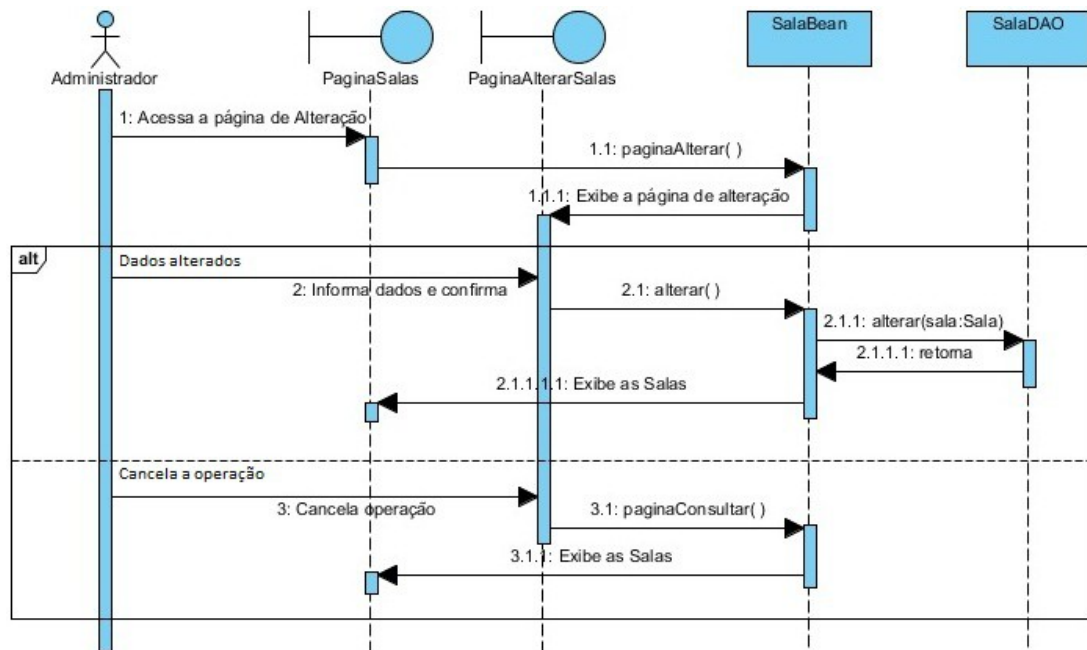


Figura 14: Diagrama de Sequências Alterar Salas

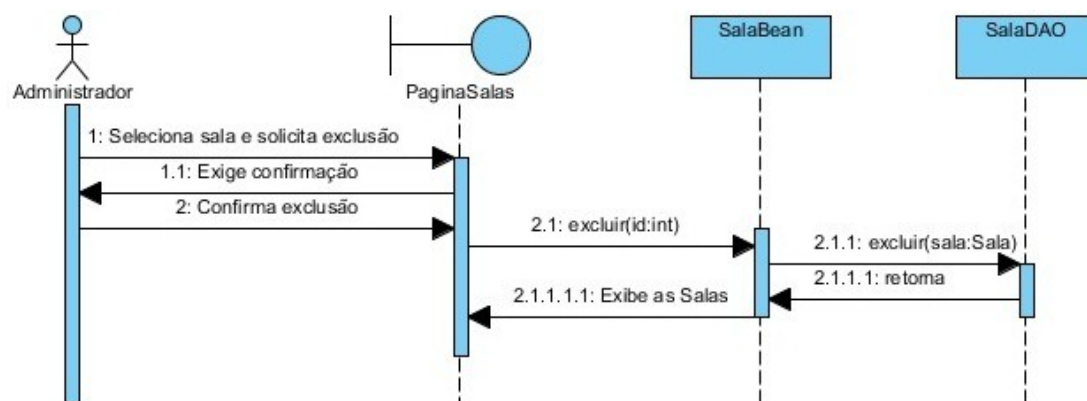


Figura 15: Diagrama de Sequências Excluir Salas

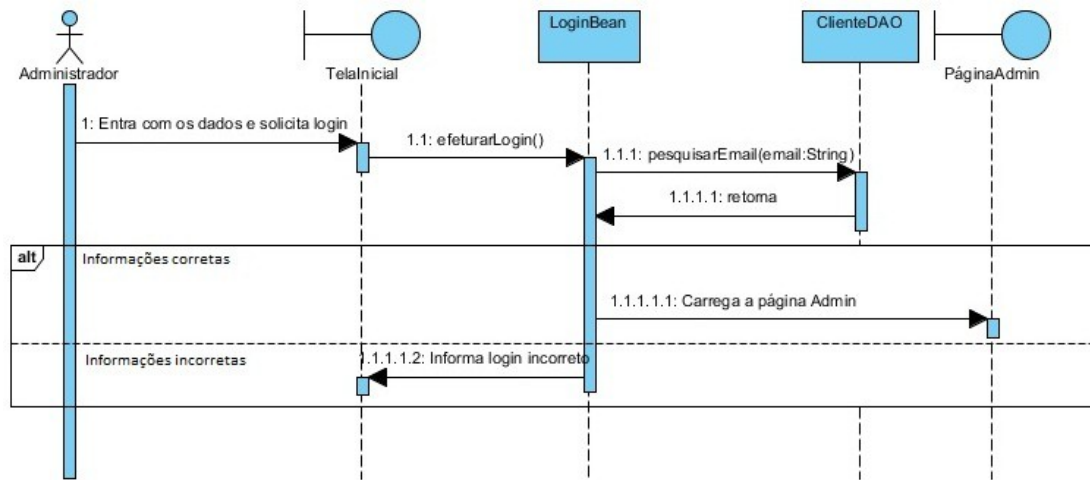


Figura 16: Diagrama de Sequência Logar-se Administrador

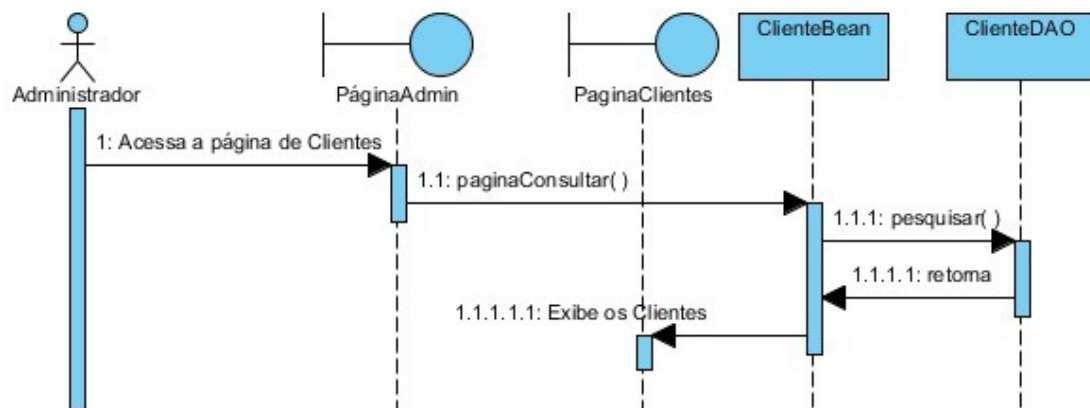


Figura 17: Diagrama de Sequência Consultar Clientes

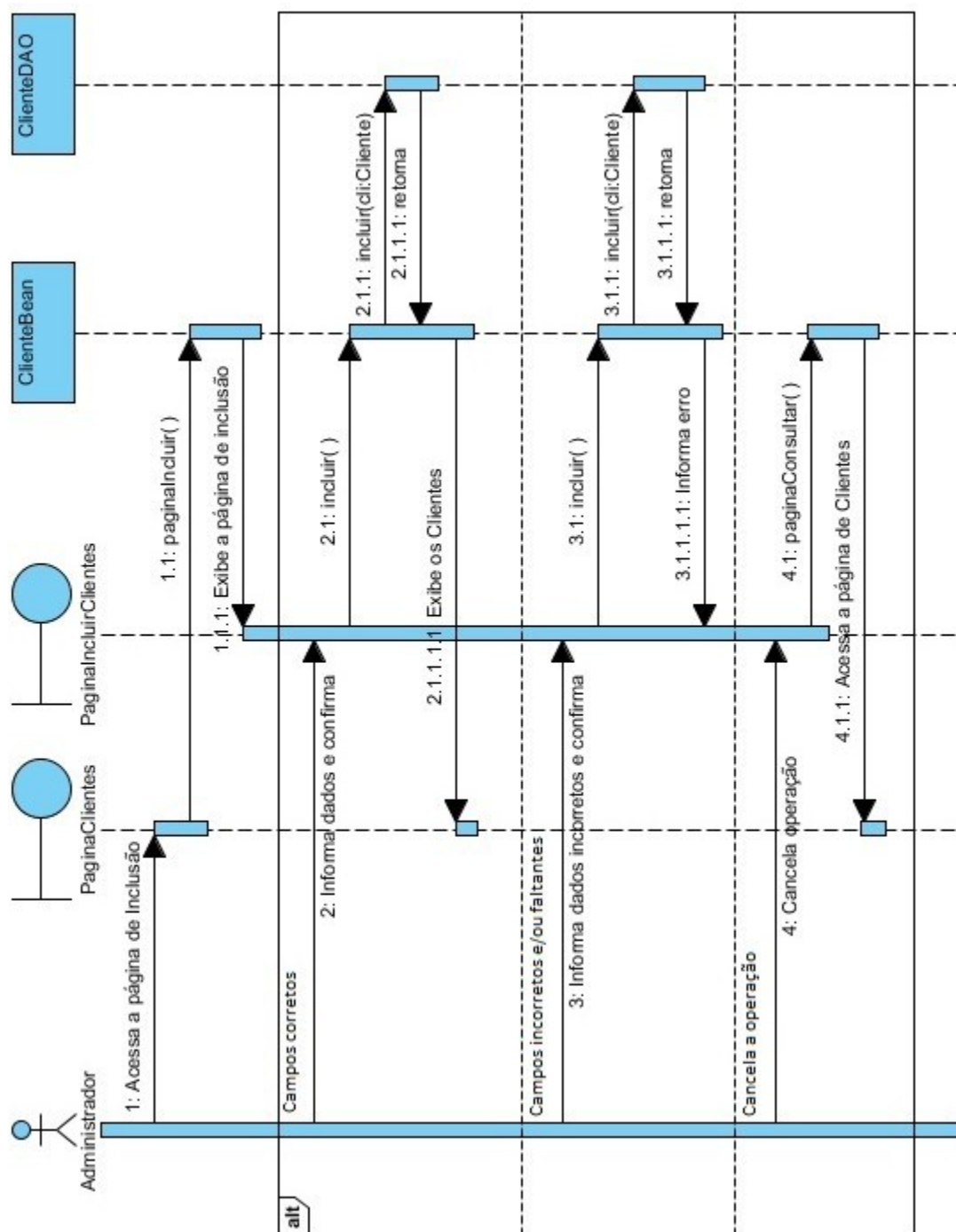


Figura 18: Diagrama de Sequência Incluir Cliente

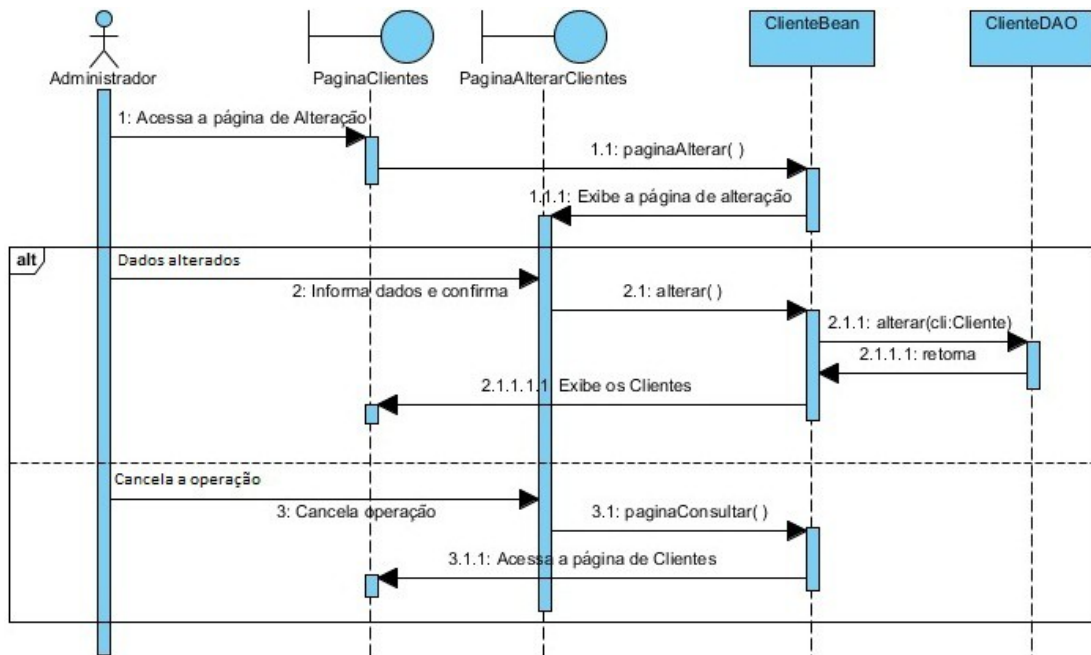


Figura 19: Diagrama de Sequência Alterar Cliente

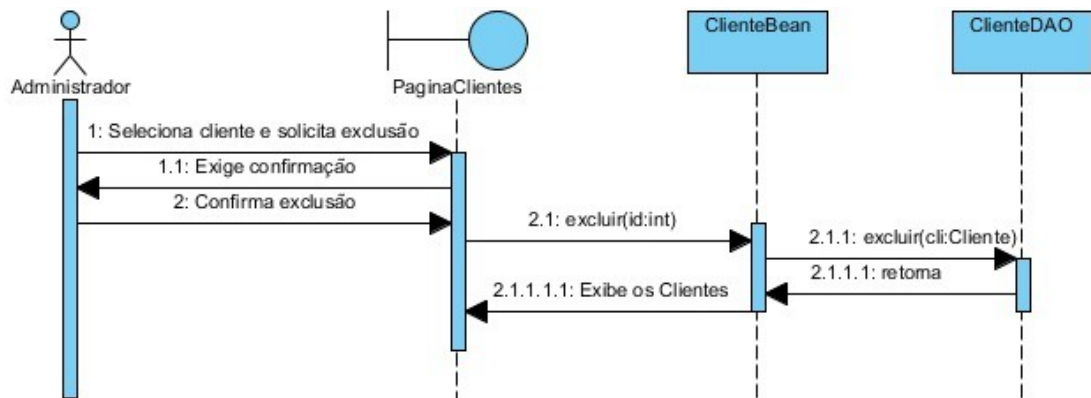


Figura 20: Diagrama de Sequência Excluir Cliente

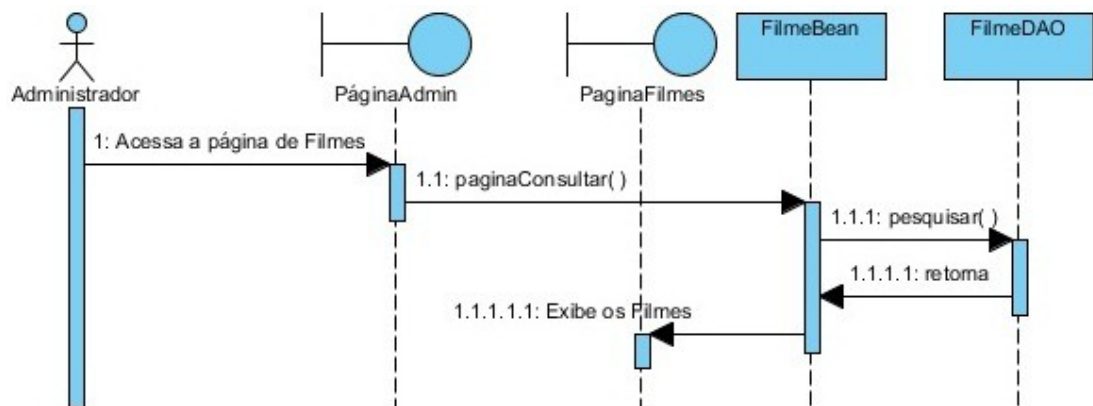


Figura 21: Diagrama de Sequência Consultar Filmes

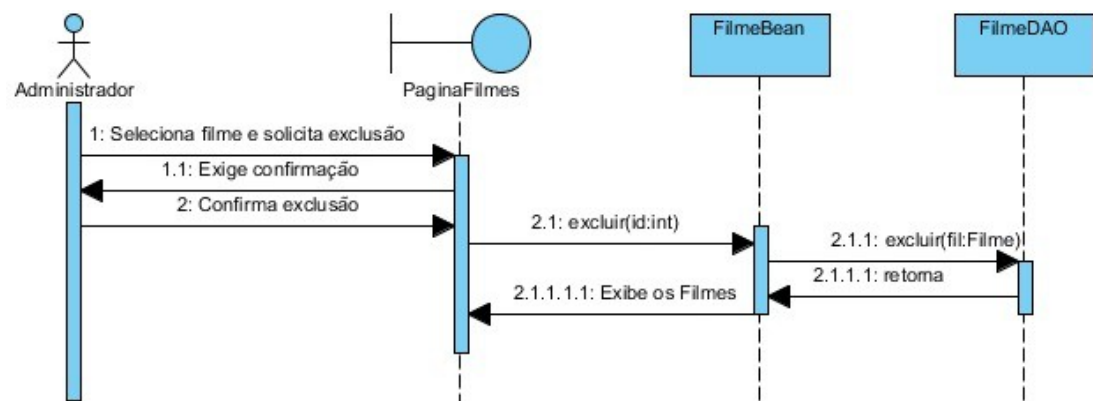


Figura 22: Diagrama de Sequência Excluir Filme

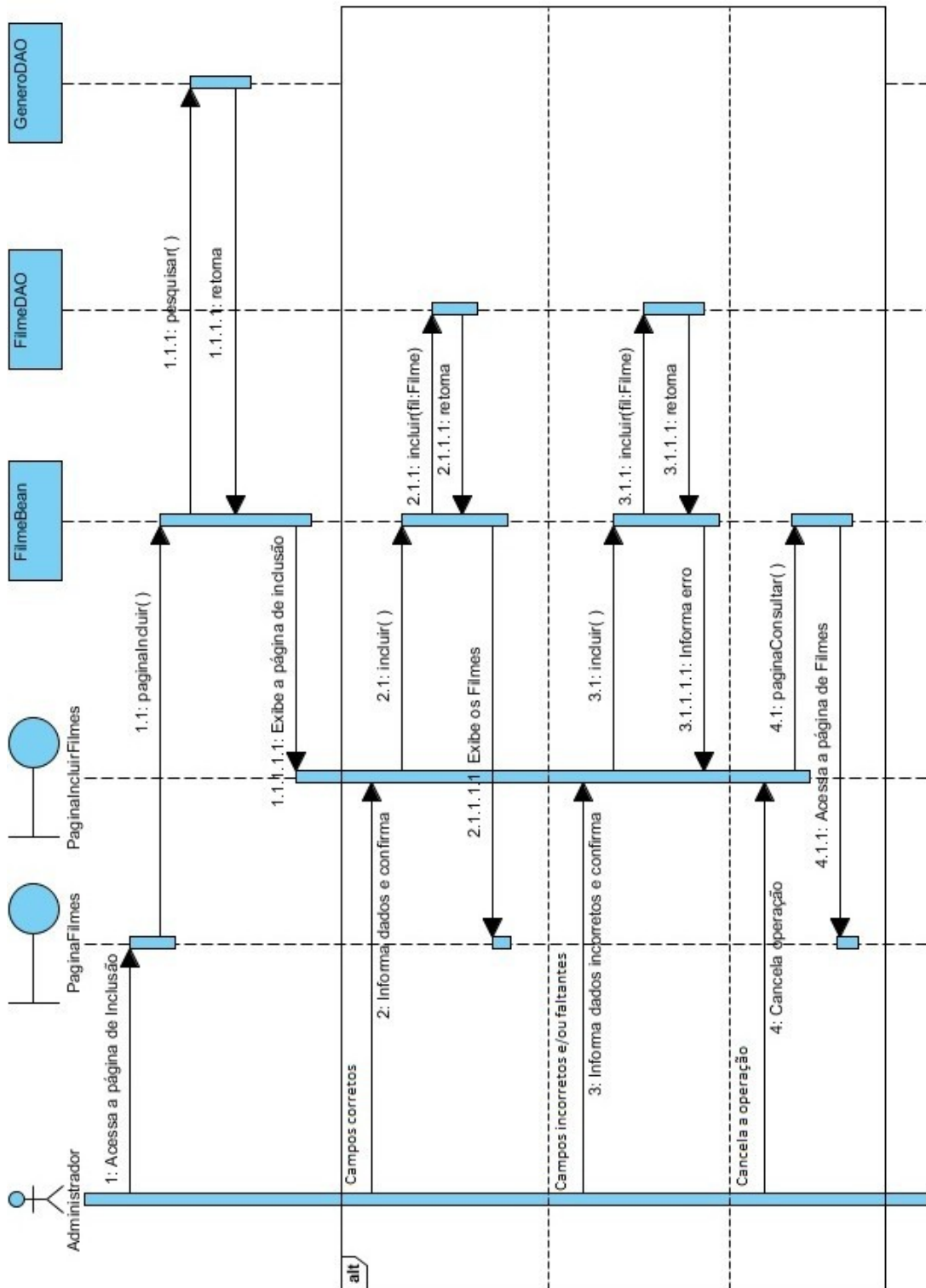


Figura 23: Diagrama de Sequência Incluir Filmes

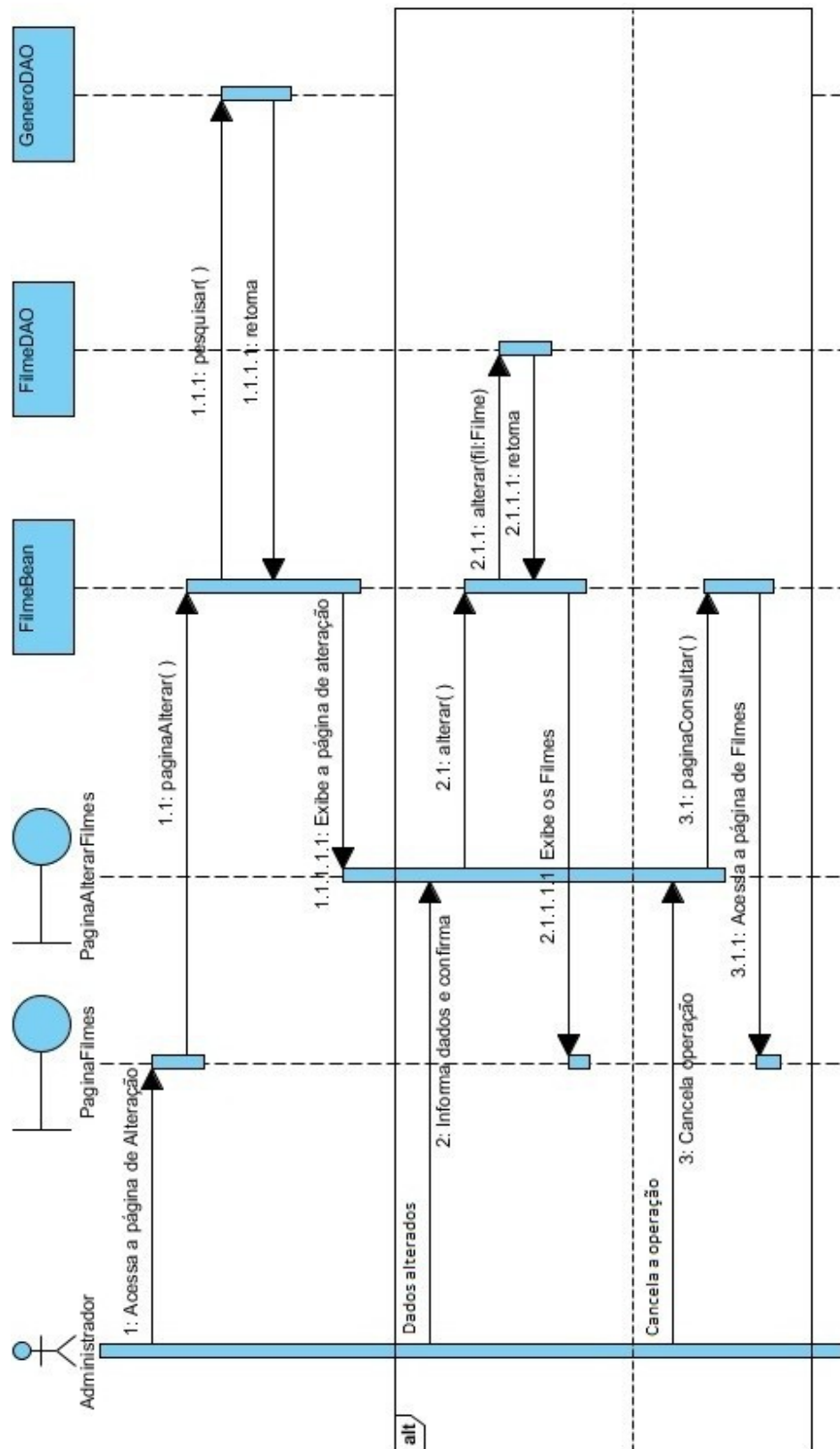


Figura 24: Diagrama de Sequência Alterar Filmes

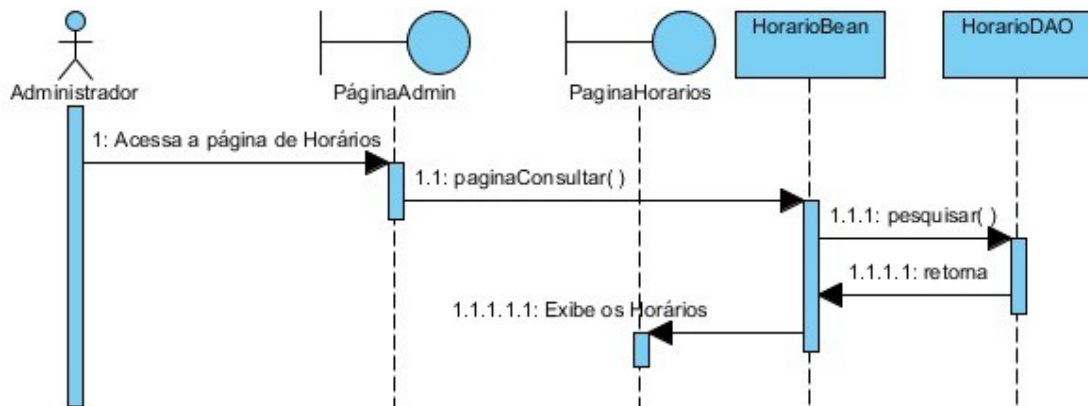


Figura 25: Diagrama de Sequência Consultar Horários

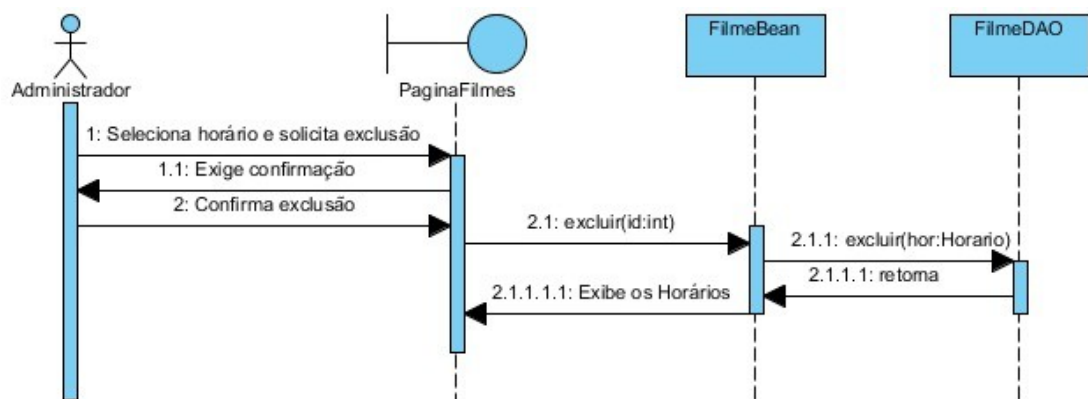


Figura 26: Diagrama de Sequência Excluir Horário

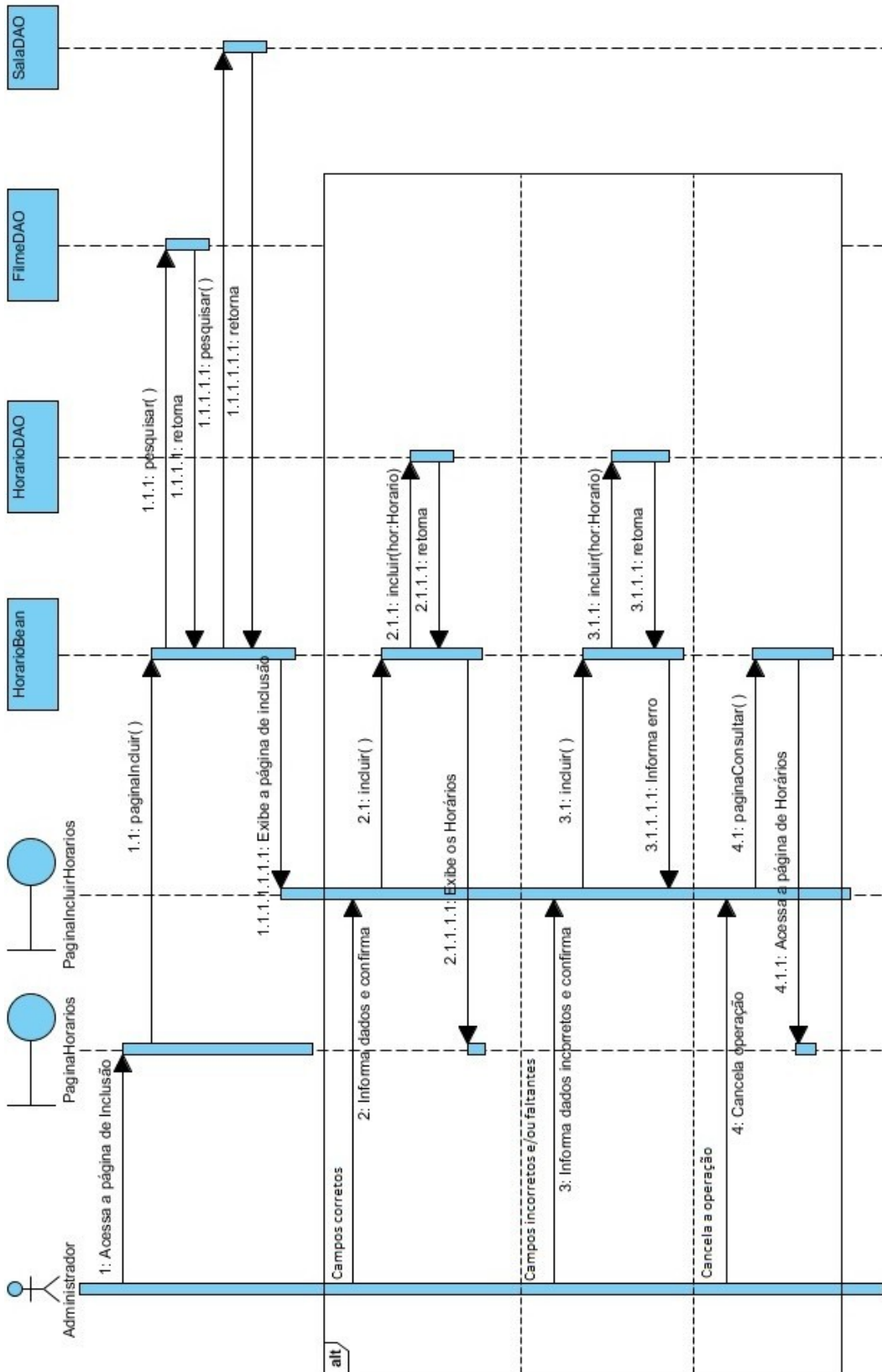


Figura 27: Diagrama de Sequência Incluir Horário

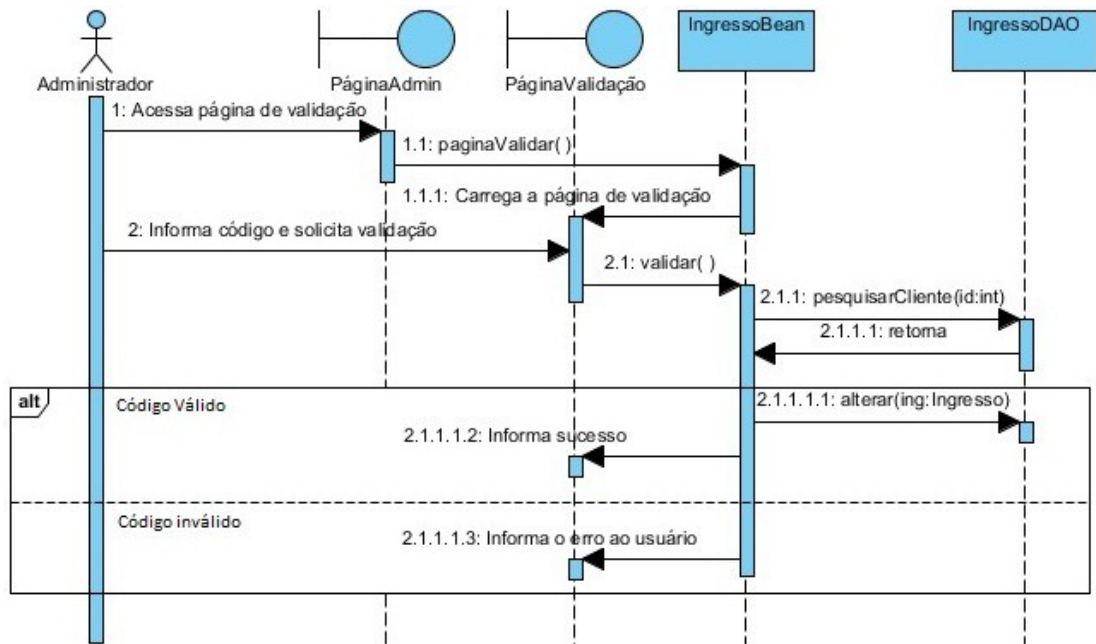


Figura 29: Diagrama de Sequência Validar Ingresso

As seguir serão apresentadas as figuras dos diagramas de sequências para os aplicativos móveis.

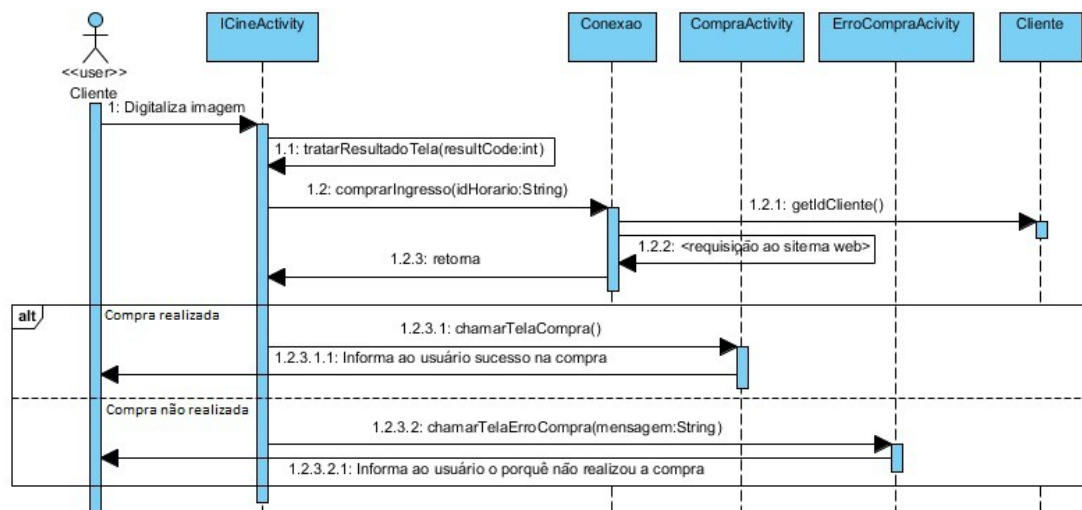


Figura 30: Diagrama de Sequência Comprar Ingresso

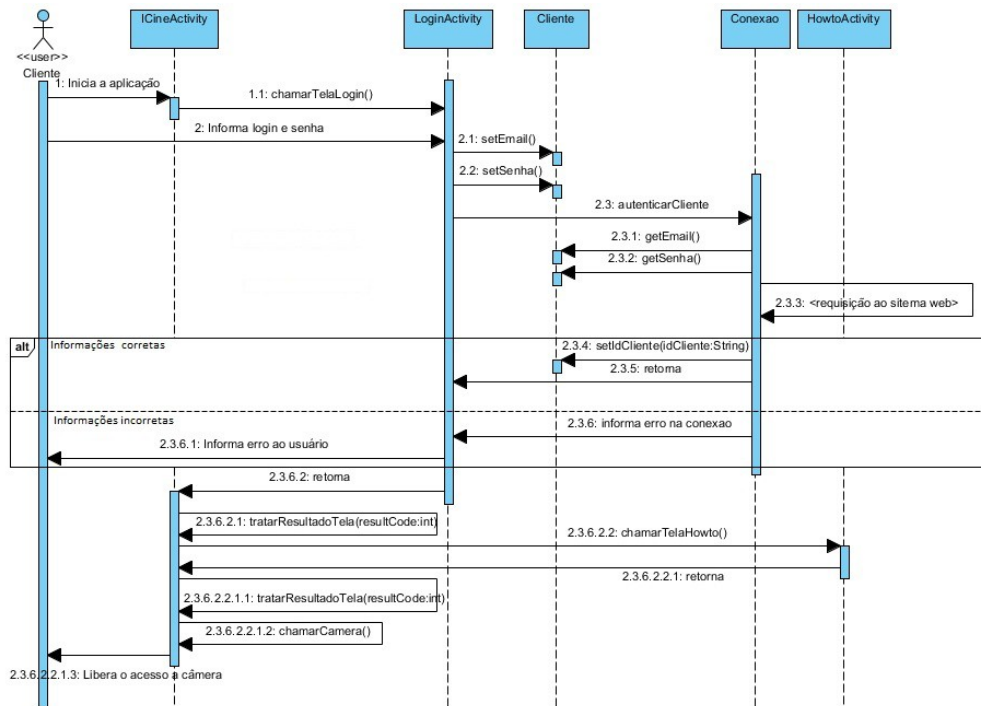


Figura 31: Diagrama de Sequência Logar-se Aplicativo Móvel

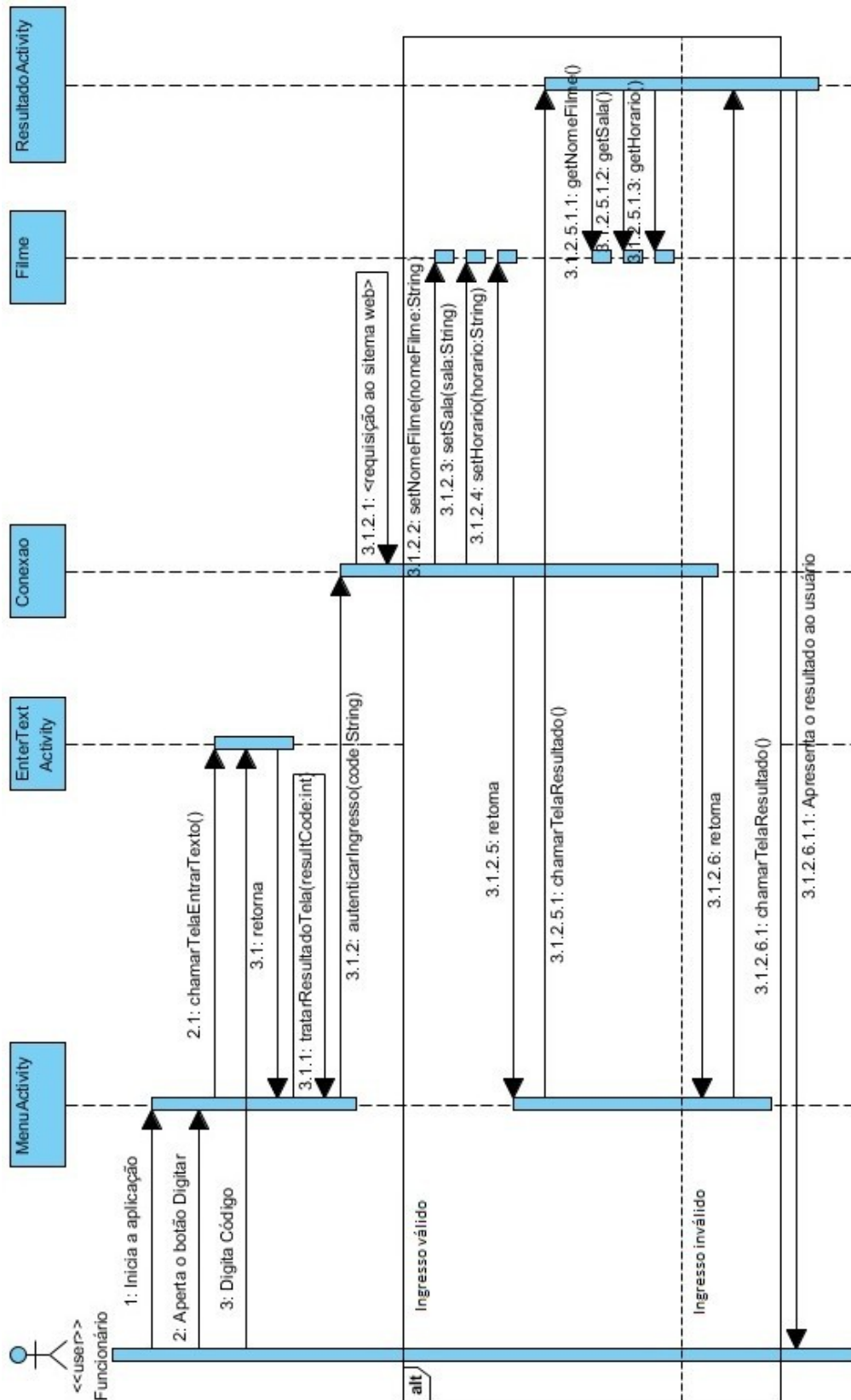


Figura 32: Diagrama de Sequência Validar Ingresso por Texto

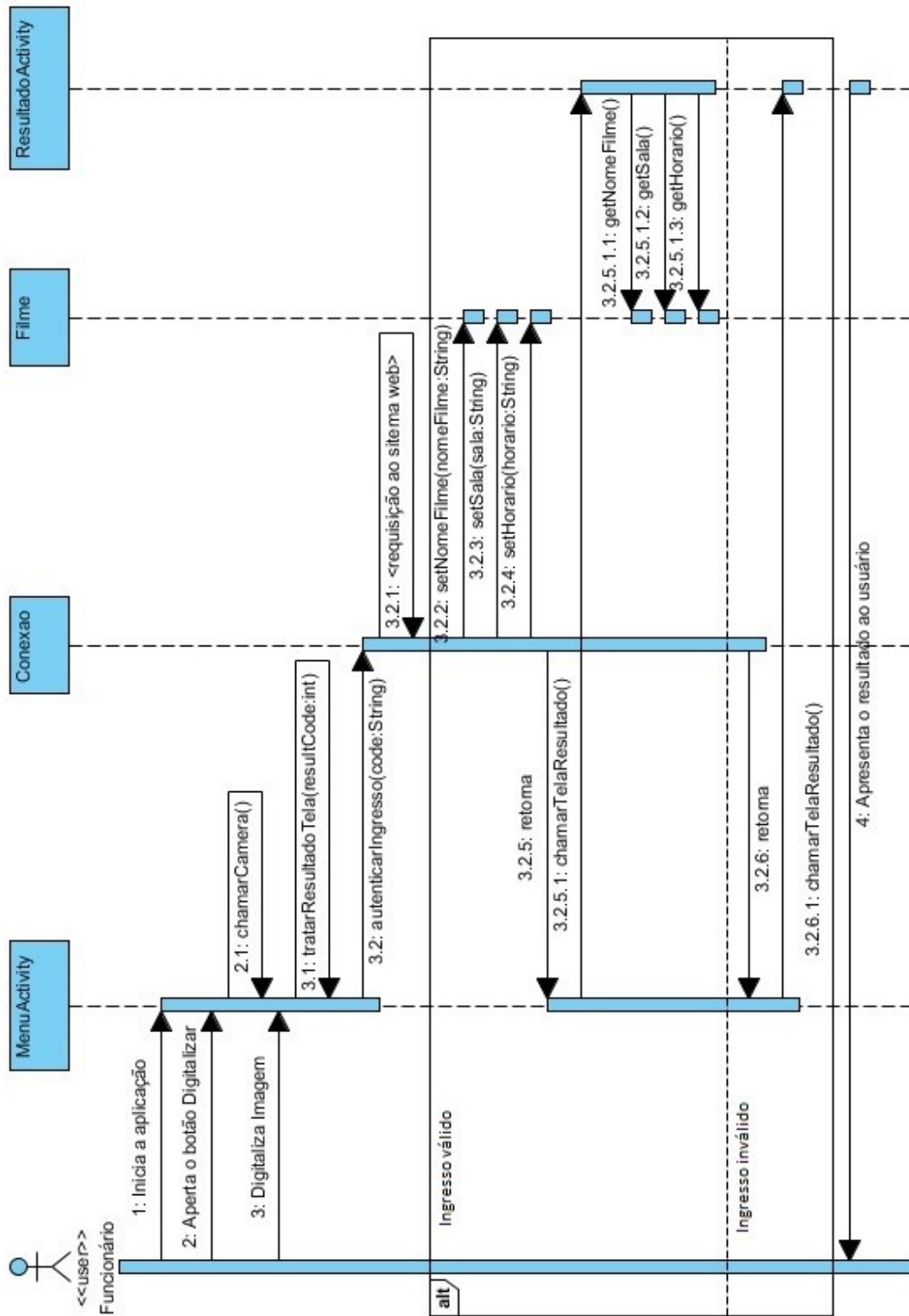


Figura 33: Diagrama de Sequência Validar Ingresso por Imagem

APÊNDICE E – Detalhamento dos Cronogramas

Na tabela 1 estão listadas as atividades utilizadas para fazer a rede PERT/CPM deste projeto e suas dependências.

Tabela 1: Lista de atividades com dependências

Atividades	Dep.
A - Levantamento de Requisitos	-
B - Diagramas de Caso de Uso	A
C - Diagrama de Classe	B
D - Diagrama de Sequência	C
E - Diagrama de Entidade-Relacionamento	D
F - Implementação do Banco de dados	E
G - Montar o ambiente de programação Android	E
H - Implementação do CRUD de cliente	F
I - Adaptação da API de decodificação do QR-Code	G
J - Implementação do CRUD de gênero	H
K - Implementação do método de Gerar Código QR	I
L - Implementação do CRUD de sala	J
M - implementação do método de enviar e-mail	K
N - Implementação do CRUD de filme	L
O - Implementação do CRUD de horário	N
P - Implementação do CRUD de Ingresso e inclusão do CtrlPoltrona	O
Q - Implementação do método de login	P
R - Implementação do servlet de login	Q
S - Implementação do auto cadastro de cliente	R
T - Implementação do método de login através do aplicativo móvel	M, R
U - Implementação do método de comprar ingresso	S
V - Implementação do servlet de comprar ingresso	U
W - Implementação do método de validar ingresso	V
X - Implementação do método de comprar ingresso pelo aplicativo móvel	V,T
Y - Implementação do servlet de validar ingresso	W
Z - Implementação do método de validar ingresso pelo celular.	Y,X
Aa - Testes	Z

As atividades da rede PERT/CPM são:

1. Início do Projeto
2. Levantamento de Requisitos Concluído
3. Diagrama de Caso de Uso Concluído
4. Diagrama de Classes Concluído
5. Diagrama de Sequências concluído
6. Diagrama de Entidade e Relacionamento Concluído
7. Banco de dados Implementado
8. Ambiente Android Funcionando
9. CRUD Cliente Funcionando
10. API do QR-Code Adaptada
11. CRUD Gênero Funcionando
12. Método de Gerar QR-Code Funcionando
13. CRUD Sala Funcionando
14. Método de Enviar E-mail Funcionando
15. CRUD de Filme Funcionando
16. CRUD de Horário Funcionando
17. CRUD de Ingressos Funcionando
18. Método de Login Funcionando
19. Servlet de Login Implementado
20. Auto-Cadastro de Cliente Implementado
21. Implementado o Login de Cliente por App Móvel
22. Comprar Ingresso Web Funcionando
23. Comprar Ingresso Servlet Funcionando
24. Validar Ingresso Web Funcionando
25. Comprar Ingresso por App Web Concluído
26. Servlet de Validação de ingresso Implementado
27. Validar Ingresso por Aplicativo Implementado
28. Codificação terminada
29. Programa Testado

Tabela 2: Tabela de Folga do Projeto

Lista de Ativ.	Duração em horas	Data Início		Data Fim		Folga Total
		PDI	UDI	PDT	UDT	
A	8	0	0	8	8	0
B	12	8	8	20	20	0
C	8	20	20	28	28	0
D	32	28	28	60	60	0
E	8	60	60	68	68	0
F	8	68	68	76	76	0
G	16	68	748	84	764	680
H	100	76	76	176	176	0
I	16	84	764	100	780	680
J	100	176	176	276	276	0
K	16	100	780	116	796	680
L	100	276	276	376	376	0
M	16	116	796	276	812	536
N	100	376	376	476	476	0
O	100	476	476	576	576	0
P	100	576	576	676	676	0
Q	16	676	676	692	692	0
R	16	692	692	708	708	0
S	16	708	708	724	724	0
T	36	708	776	744	812	68
U	56	724	724	780	780	0
V	16	780	780	796	796	0
W	36	796	796	832	832	0
X	36	796	812	832	848	16
Y	16	832	832	848	848	0
Z	36	848	848	884	884	0
Aa	16	884	884	900	900	0

Para a tabela 2 considere PDI (Primeira Data de Início), UDI (Última Data de Início), PDT (Primeira Data de Término) e UDT (Última Data de Término).

O caminho crítico para esse projeto é: A, B, C, D, E, F, H, J, L, N, O, P, Q, R, S, U, V, W, Y, Z, Aa.

Rede PERT/CPM

A rede PERT deste projeto foi dividido em 3 imagens para melhor compreensão.

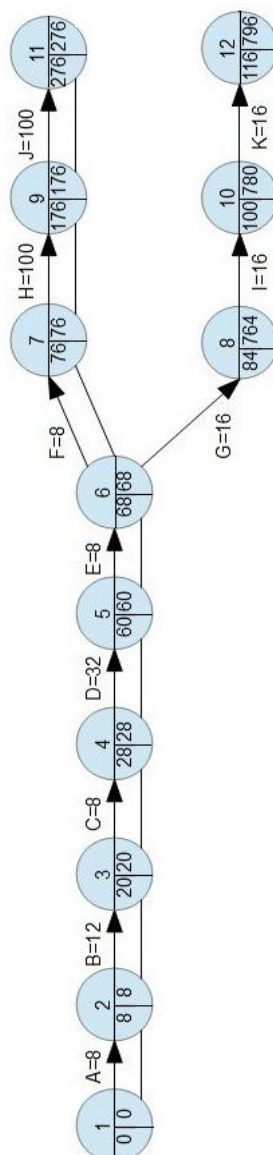


Figura 1: Rede PERT/CPM Primeira Parte

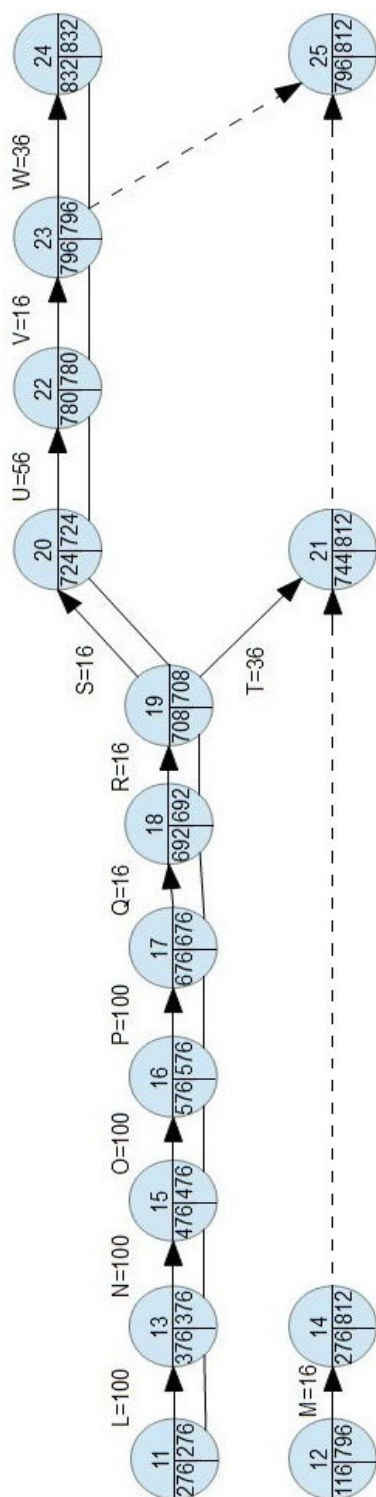


Figura 2: Rede PERT/CMP Segunda Parte

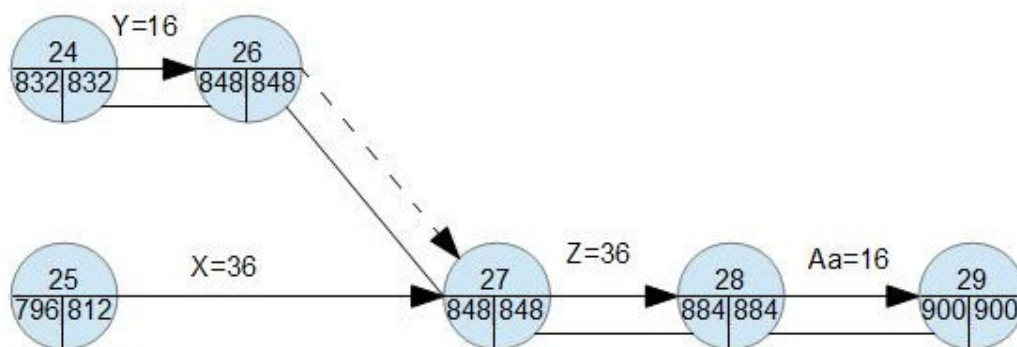


Figura 3: Rede PERT/CMP Terceira Parte