



Universidade do Minho

Conselho de Cursos de Engenharia

Licenciatura em Engenharia Informática

3ºAno

Disciplina de Laboratórios de Informática IV

Ano Lectivo de 2009/2010

Newzic – Relatório de Projecto

**André Carvalho, Bruno Amorim,
Christophe Gonçalves, Ricardo Alves,
Vitor Gonçalves**

Maio 2010

Data de Recepção	
Responsável	
Avaliação	
Observações	

Newzic

André Carvalho (46193),
Bruno Amorim (46194),
Christophe Gonçalves (44245),
Ricardo Alves (47033),
Vitor Gonçalves (44259)

Maio 2010

Resumo

Este relatório serve para dar a conhecer o “Newzic” (pronuncia-se ‘nyoōzik) aos clientes. É o resultado do trabalho de pesquisa e desenvolvimento da equipa do “Newzic”.

O projecto consiste num modelo noticioso online orientado ao paradigma *Web 2.0* onde as notícias são criadas e geridas pela comunidade.

Inicialmente, na fase de fundamentação, procedemos ao levantamento de requisitos por parte do cliente, ou seja, descrevemos o que vai ser feito no nosso projecto.

Depois de recolhidos todos os requisitos do sistema, avançamos para a fase de especificação, onde modelamos todo o sistema. Basicamente, descrevemos como é que todas as funcionalidades foram implementadas, de acordo com a fundamentação inicialmente estabelecida.

Por fim, na fase de implementação, pusemos em prática de todos os aspectos que foram fundamentados e especificados anteriormente, com vista a obter um projecto de software sólido e coerente, desde o início até ao fim.

Área de Aplicação: Notícias sobre música.

Palavras-Chave: Notícia, Comunidade, Jornalista.

Índice

Resumo	i
Índice	ii
Índice de Figuras	iv
Índice de Tabelas	v
1. Introdução	1
1.1. Contextualização	1
1.2. Apresentação do Caso de Estudo	1
1.3. Motivação e Objectivos	1
1.4. Estrutura do Relatório	3
2. Cliente	4
3. Fundamentação	5
3.1. Website	5
3.1.1 Jornalista	6
3.1.2 Moderador	9
3.1.3 Administrador	9
3.1.4 Leitor	10
3.1.5 Ranking	10
3.2. Web Service	10
4. Especificação	12
4.1. WebSite	12
4.1.1 Página Inicial	12
4.1.2 Ranking	13
4.1.3 Pesquisa	13
4.1.4 Ver/Editar Perfil	13
4.1.5 Fazer Queixa	14
4.1.6 Consultar notícia	14
4.1.7 Publicar/Editar Notícia	16
4.1.8 Moderar	16
4.1.9 Administrar	19
4.2. Web Service	20
4.2.1 Leitor	20
4.2.2 Jornalista	21

4.3. Diagrama de Classes	22
4.3.1 Apresentação	23
4.3.2 Negócio	24
4.3.3 Dados	25
4.4. Base de dados	25
4.4.1 Diagrama Conceptual	26
4.4.2 Diagrama Lógico	27
5. Implementação	28
5.1. Camada de Dados	28
5.2. Camada de Negócio	29
5.3. Camada de Apresentação	29
5.4. Screenshots do Website	32
6. Mapa do Projecto	34
6.1. Modelo de desenvolvimento	34
6.2. Planificação	35
6.3. Ferramentas e Tecnologias utilizadas	36
7. Conclusão e Trabalho Futuro	37
Bibliografia	39
Referências WWW	40
Lista de Siglas e Acrónimos	41
 Anexos	
I. Blog	43
II. Web 2.0	44
III. Web Service	45
IV. CAPTCHA	47
V. Modelo de Domínio	48
VI. Diagramas de Use Cases	49
VII. Diagramas de Sequência	50
VIII. Diagrama de Classes	51
IX. Modelos E-R	52

Índice de Figuras

Ilustração 1 - Componentes do Sistema	5
Ilustração 2 - Use Case "Geral"	12
Ilustração 3 - Use Case "Gerir Notícias"	15
Ilustração 4 – Use Case “Moderar”	17
Ilustração 5 - Use Case "Gerir Moderadores"	19
Ilustração 6 - Diagrama de Classes "Camada de Apresentação"	23
Ilustração 7 - Diagrama de Classes "Camada de Negócio"	24
Ilustração 8 - Diagrama de Classes "Camada de Dados"	25
Ilustração 9 - Diagrama Conceptual da Base de Dados	26
Ilustração 10 - Diagrama Lógico da Base de Dados	27
Ilustração 11 - Arquitectura do Sistema	31
Ilustração 12 - Página Principal	32
Ilustração 13 - Criar notícia	32
Ilustração 14 - Criar Notícia	33
Ilustração 15 - Ver Notícia	33

Índice de Tabelas

Tabela 1 - Planificação do projecto

35

1. Introdução

1.1. Contextualização

Nos sites noticiosos tradicionais, o conteúdo jornalístico é fornecido por profissionais da área. Uma agência noticiosa emprega pessoal responsável pela recolha de informação ou subcontrata outras agências para o efeito. Esta informação é a base de um artigo jornalístico. Posteriormente, os artigos são publicados com a aprovação de um ou mais editores.

Actualmente estamos a assistir a uma mudança na forma como a informação é distribuída e consumida. Cada vez mais as pessoas procuram notícias/artigos em *blogs* e fóruns onde (tradicionalmente) o conteúdo é gerado pela comunidade e não por jornalistas. No entanto, essas notícias não estão centralizadas. Um leitor tem que visitar diferentes *blogs* e fóruns para ler diferentes notícias/artigos escritos por diferentes pessoas.

1.2. Apresentação do Caso de Estudo

Desenvolvemos uma plataforma noticiosa que apresenta aos leitores várias notícias sobre o mundo da música escritas por pessoas como eles. Desta forma, os utilizadores têm uma fonte de notícias centralizada onde podem publicar as notícias que acharem importantes e ver o seu trabalho comentado e avaliado pela comunidade.

1.3. Motivação e Objectivos

Quando as notícias são dadas por grandes agências noticiosas, é impossível ter a certeza que não haja interesses políticos, económicos, etc. a influenciar a integridade das mesmas. Para além disso, um site noticioso tradicional tem custos associados com pessoal que escreve, aprova e publica artigos.

As pessoas que já procuram as suas notícias em fóruns e *blogs*, deparam-se com outro problema – não existe um local centralizado onde encontrar as notícias. Um *blog* apenas aloja

notícias de uma pessoa e regra geral, este não é dedicado só a estas, ou seja, o utilizador tem que procurar as notícias no meio da informação toda do *blog*.

No “Newzic”, um jornalista pode publicar a sua notícia/artigo num local dedicado para o efeito, onde tem boa visibilidade e tem o aspecto de uma notícia/artigo escrita/o por um profissional num site noticioso normal. A notícia pode ser lida e avaliada por toda a comunidade com os mesmos interesses que o escritor. Não precisa de colocar uma notícia esporádica no seu *blog*, onde apenas costuma escrever sobre o seu dia-a-dia e onde a notícia teria pouca visibilidade. Até mesmo um jornalista profissional pode publicar uma notícia que o editor do seu jornal tenha recusado.

1.4. Estrutura do Relatório

Secção 2 – Cliente - descrevemos brevemente o cliente para o qual o “Newzic” está a ser desenvolvido.

Secção 3 – Fundamentação – fala de algumas decisões tomadas por nós ao nível da abordagem ao desenvolvimento do projecto.

Secção 4 – Especificação - apresentamos um documento de especificação que indica como as várias funcionalidades do “Newzic” serão implementadas.

Secção 5 – Implementação - apresentamos alguma informação sobre a construção e codificação do projecto.

Secção 6 – Mapa do projecto - descrevemos o processo de desenvolvimento que seguimos para construir o “Newzic”.

Secção 7 – Conclusão e Trabalho futuro – tecemos algumas opiniões críticas sobre o nosso projecto e o trabalho desenvolvido.

2. Cliente

A empresa “MusicALL” pretende criar um Website noticioso gerido pela comunidade de forma envolver os seus leitores no processo noticioso e na esperança de aumentar a sua popularidade. A aposta vira-se para a Web 2.0. Para esse reconhecimento, esta empresa recrutou a nossa equipa de projecto para desenvolver o “Newzic” – website de notícias sobre o panorama musical mundial onde toda a informação é disponibilizada, avaliada e comentada pela comunidade. Este facto trás bastantes vantagens para a empresa, visto que não ter de incorrer em custos de pagamentos salariais, gestão do website, etc.

3. Fundamentação

O nosso sistema é constituído por dois componentes: um *Website* e um *Web Service*.

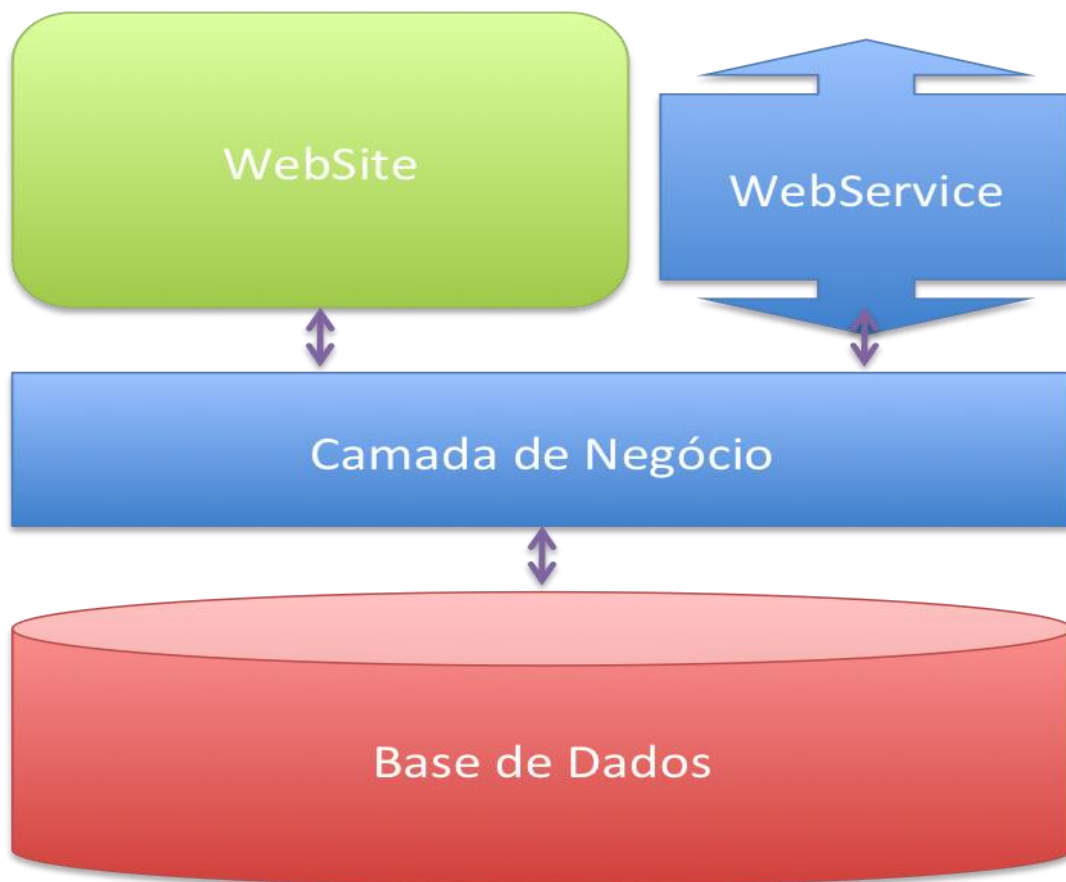


Ilustração 1 - Componentes do Sistema

3.1. Website

A agência noticiosa fornece um *website* aos seus utilizadores em que a página principal mostra os títulos das notícias mais recentes e com melhor classificação (o sistema de classificação de uma notícia é descrito em detalhe na secção *ranking*)

Como haverá utilizadores registados e não registados, e como o *website* disponibiliza diferentes funcionalidades para diferentes tipos de utilizador, na página principal são facultadas a opção de *login/logout* e registo de um utilizador.

Na página principal é possível a pesquisa de notícias, onde o utilizador poderá escolher se quer pesquisar por conteúdo, título, *tags* ou autor, e como quer ordenar essa pesquisa. O resultado será uma lista de notícias com a seguinte estrutura:

- Título da notícia
- *Ranking*
- Data de inserção
- Pequeno excerto

Uma notícia tem no seu cabeçalho o título e data de inserção, podendo ter também um mapa com a localização ou localizações da mesma, sempre que o jornalista que a inseriu assim o pretenda. O corpo é constituído por texto e *links*, seguido de imagens e/ou vídeos. Para facilitar a pesquisa de informação relacionada com a notícia, ao lado de cada uma é fornecida uma lista com as notícias relacionadas ordenada por ranking. O rodapé da notícia contém o *ranking* desta, as *tags* relacionadas com a notícia e ainda uma área para comentários dos utilizadores registados (jornalistas/moderadores/administradores).

No *website* também é possível que um utilizador se registe. Na página de registo é pedido ao utilizador para introduzir os seus dados pessoais tais como o nome, palavra-chave e *e-mail* seguidos de uma validação por parte do utilizador de forma a verificar a sua legitimidade. Essa validação será feita através da introdução de um CAPTCHA.

Para ajudar os utilizadores a se manterem actualizados, o *website* incorpora um sistema de RSS com as últimas notícias inseridas. Desta forma, os utilizadores podem ter conhecimento das últimas notícias sem a necessidade de abrirem a página *web*.

Devido à popularidade das plataformas móveis e ao facto dos utilizadores as usarem cada vez mais para navegarem na *Internet*, o website possui uma opção de visualização específica para estas plataformas.

3.1.1 Jornalista

As notícias chegam ao “Newzic” através da intervenção de um utilizador registado, o qual recebe o título de Jornalista. Pode ser qualquer um de nós, que após ter adquirido novas informações as possa publicar no site. Essas notícias vão desde uma aparição pública de uma

banda, um mapa a descrever as várias cidades por onde vai passar um *tour* ou até mesmo o novo videoclip da sua banda preferida que está no *youtube*.

O Jornalista é, obviamente, responsável pelo conteúdo noticioso que publica, pois está sujeito a regras de publicação e à avaliação das suas notícias por parte dos leitores, que, em determinadas circunstâncias, podem levar uma notícia a desaparecer.

O Jornalista actuará no “Newzic” com as seguintes tarefas:

- Publicação
- Edição
- Comentários
- Votação

Quando um utilizador se autentica como Jornalista, o site disponibiliza-lhe opções para publicar uma nova notícia ou editar uma notícia existente.

Quando o Jornalista decide publicar uma notícia, o “Newzic” fornece-lhe ferramentas muito simples de utilizar. O Jornalista é “conduzido” para uma página onde lhe é apresentado um conjunto de caixas de texto onde será permitido criar a sua notícia. O site não permite a publicação de uma notícia que não contemple os seguintes requisitos:

- Título
- Conteúdo no corpo da notícia
- *Tags* de pesquisa

Uma notícia pode ser composta por vários itens. Estes seguem uma ordem e locais pré-definidos pela *interface* do “Newzic”.

1. Texto
2. Imagem
3. Vídeo
4. Mapa

1. Uma caixa de texto é responsável por receber o corpo da notícia. É obrigatório que a notícia contenha texto nesta secção, seja um artigo completo ou uma simples frase que complemente um vídeo, imagem ou mapa.
2. Também é possível introduzir imagens para compor a notícia. No entanto, estas não ficam junto do corpo da notícia aquando da visualização da mesma. Se a notícia tiver imagens, existe uma área reservada para a apresentação destas. Nessa área é

disponibilizada uma caixa onde será possível visualizar as várias imagens que podem pertencer à notícia. Para adicionar imagens à notícia, o Jornalista tem de fazer o *upload* das mesmas. As imagens que enviar são armazenadas e associadas à notícia, tendo um limite de tamanho por imagem, assim como um limite total de imagens definido pelos administradores do “Newzic”.

3. Existem outros tipos de média que o Jornalista pode publicar: vídeos embebidos provenientes de sites dedicados a esse efeito (*youtube*, *yahoo* vídeo, *metacafe*, etc.). O Jornalista não pode efectuar *uploads* de vídeos. Para essa publicação, existe uma lista de hiperligações de vídeos que pode gerir. Em semelhança com as imagens, os vídeos também não ficam juntos ao corpo da notícia.
4. Se o jornalista o desejar, também pode associar um mapa à sua notícia. Neste caso, ao adicionar um mapa, é-lhe pedido o tipo do mesmo. Este pode ser:
 - Ponto:
 - Concerto
 - Aparição Pública
 - Outros
 - *Tour*: Uma *tour* consiste num conjunto de pontos que representam os vários locais onde uma banda irá actuar num determinado ano.

O Jornalista deverá também incluir *tags* para facilitar a indexação e pesquisa da sua notícia. Também é disponibilizada a opção de pré-visualizar a notícia tal como ela será apresentada no site depois de ser publicada. Após confirmar que tudo está correcto, o Jornalista pode então tornar a sua notícia pública.

O Jornalista pode editar qualquer notícia que tenha publicado. O acesso à notícia pode ser feito através de uma pesquisa ou da sua página de perfil onde lhe será apresentada uma lista de todas as notícias que publicou. Após seleccionar a notícia que deseja editar, o Jornalista é novamente “conduzido” à página onde criou a notícia, mas com todos os itens previamente preenchidos, tal como o Jornalista o tinha feito. Após efectuar as modificações que achar necessárias, pode pré-visualizar e/ou publicar a notícia com as alterações introduzidas.

Um utilizador registado como Jornalista tem o poder (e a responsabilidade) de comentar e votar nas notícias publicadas por outros Jornalistas. Sempre que um Jornalista estiver a ler uma notícia, tem a opção de deixar um comentário numa caixa de texto própria para o efeito.

Além disso, cada notícia tem um sistema de *ranking* associado. Se um Jornalista entender que uma notícia está bem construída, que é verdadeira ou interessante, pode votar “para cima” nessa notícia, aumentando a classificação da mesma ou, alternativamente, pode votar “para baixo”, reduzindo a classificação.

3.1.2 Moderador

Um Moderador tem a responsabilidade de moderar todo o conteúdo do *website*.

O Moderador tem as funções dum Jornalista, com a responsabilidade acrescida de controlar as notícias publicadas pelos utilizadores com o cargo de Jornalistas, ou seja, pode remover uma notícia caso considere que esta infrinja o código de conduta do *website*, ou avisar o Jornalista que a publicou para reescrever a mesma, marcando-a como má. Se o Moderador considerar o assunto da notícia abusivo, pode, em último caso, bloquear a conta do Jornalista.

O Moderador não pode, no entanto, modificar o conteúdo de uma notícia. Isto pretende evitar que o teor das notícias não possa ser manipulado.

O Moderador também tem como função analisar os comentários publicados pelos leitores às notícias. Se os considerar abusivos, pode removê-los e em casos extremos banir os jornalistas que os escreveram.

Um Moderador não pode no entanto banir ou apagar as notícias ou comentários de outros Moderadores. Esse poder é reservado ao Administrador.

Os Jornalistas também podem reportar se uma notícia for abusiva. Caso existam mais de 5 queixas, o Moderador é informado pelo sistema e, conseqüentemente, decide se essa notícia é ou não válida. O Moderador também pode censurar um comentário caso este seja abusivo.

3.1.3 Administrador

O Administrador de um *Web site* é o “agente” responsável por toda a gestão do mesmo.

No nosso website em particular, o Administrador tem a possibilidade de fazer tudo o que o Leitor, Jornalista e o Moderador fazem e tem ainda a cargo adicional, a gestão dos Moderadores, promovendo Jornalistas a Moderadores ou despromovendo Moderadores caso estes tenham praticado comportamentos abusivos ou de algum modo não concordantes com as funções/obrigações do cargo. Em casos extremos, o Administrador reserva o poder de banir Moderadores.

Um Jornalista (ou Moderador) pode reportar uma notícia (doutro) Moderador. Se este receber cinco ou mais queixas, o Administrador será notificado. O Administrador também está

encarregue de ler queixas escritas por qualquer Jornalista ou Moderador feitas a outro utilizador ou ao site em si.

3.1.4 Leitor

Este tipo de utilizador apenas pode pesquisar ou ler as notícias que são publicadas. Não precisa de estar registado para o efeito. É o utilizador com menos privilégios no nosso *website*. Qualquer utilizador que não pertença a nenhum dos grupos acima, pertence portanto (por defeito) ao grupo de leitores.

3.1.5 Ranking

Todas as notícias podem ser votadas pelos utilizadores com privilégios para tal (Jornalistas/Moderadores/Administradores). Estes têm de estar autenticados para puderem votar. Deste modo, cada utilizador não pode votar mais do que uma vez.

O sistema de votação baseia-se no sistema simples de *upvote/downvote*.

Uma notícia começa com uma pontuação igual a zero e os utilizadores podem fazer um voto positivo (soma uma unidade à pontuação actual da notícia) ou um voto negativo (subtrai uma unidade à pontuação actual da notícia). Sendo assim, a notícia pode ter um valor positivo, nulo ou negativo consoante a soma dos votos positivos e negativos.

Também existe um sistema de classificação para os Jornalistas. Este é calculado automaticamente pelo ranking de todas as notícias de cada Jornalista e varia de um a dez, pelo que, os jornalistas que tiverem melhores pontuações nas suas notícias, ficam no “*top 10*”.

3.2. Web Service

A plataforma disponibiliza uma API de modo a que aplicações externas possam interagir com o “Newzic”.

Embora seja necessário usar o website para usufruir de todas as capacidades do “Newzic”, o *Web Service* é robusto o suficiente para proporcionar um nível de extensibilidade à plataforma, que permite aos utilizadores moldarem a sua interacção com o “Newzic” de acordo com as suas necessidades (seja através de uma *applet*, extensão a um *browser* ou um simples *desktop widget*.)

Os métodos do “Newzic” que podem ser invocados remotamente variam consoante as credenciais do utilizador que as invoca. O *Web Service* suporta dois níveis de utilizador e

disponibiliza diferentes métodos para os dois níveis. Os níveis de utilizador suportados são os de Jornalista e Leitor, com privilégios associados idênticos aos privilégios de Jornalista e Leitor do Website, respectivamente.

O utilizador com privilégios de Leitor apenas pode invocar métodos de pesquisa e consulta de notícias. A pesquisa comporta-se de forma idêntica à do *website*, ou seja, recebe os mesmos parâmetros que uma pesquisa no *website* e devolve o mesmo conjunto de resultados. O método de consulta devolve todos os elementos que constituem uma notícia (texto, vídeo e imagem), não havendo portanto nenhuma diferença no conteúdo de uma notícia que seja consultada pelo *Web Service* em relação ao *Website*. Os comentários associados a essa notícia não são no entanto acessíveis através do *Web Service*.

O utilizador com privilégios de Jornalista pode, além dos métodos de pesquisa e consulta, invocar métodos para publicar, editar e votar numa notícia. Qualquer um destes métodos provoca resultados idênticos aos resultados de executar estas acções no *website*, ou seja, não existe nenhuma diferença em publicar, editar ou votar numa notícia pelo *Web Service* ou pelo *website*.

Como os métodos disponibilizados dependem no nível de utilizador e privilégios associados, é portanto obrigatório (para além dos métodos referenciados) a existência de métodos responsáveis pela autenticação.

4. Especificação

4.1. WebSite

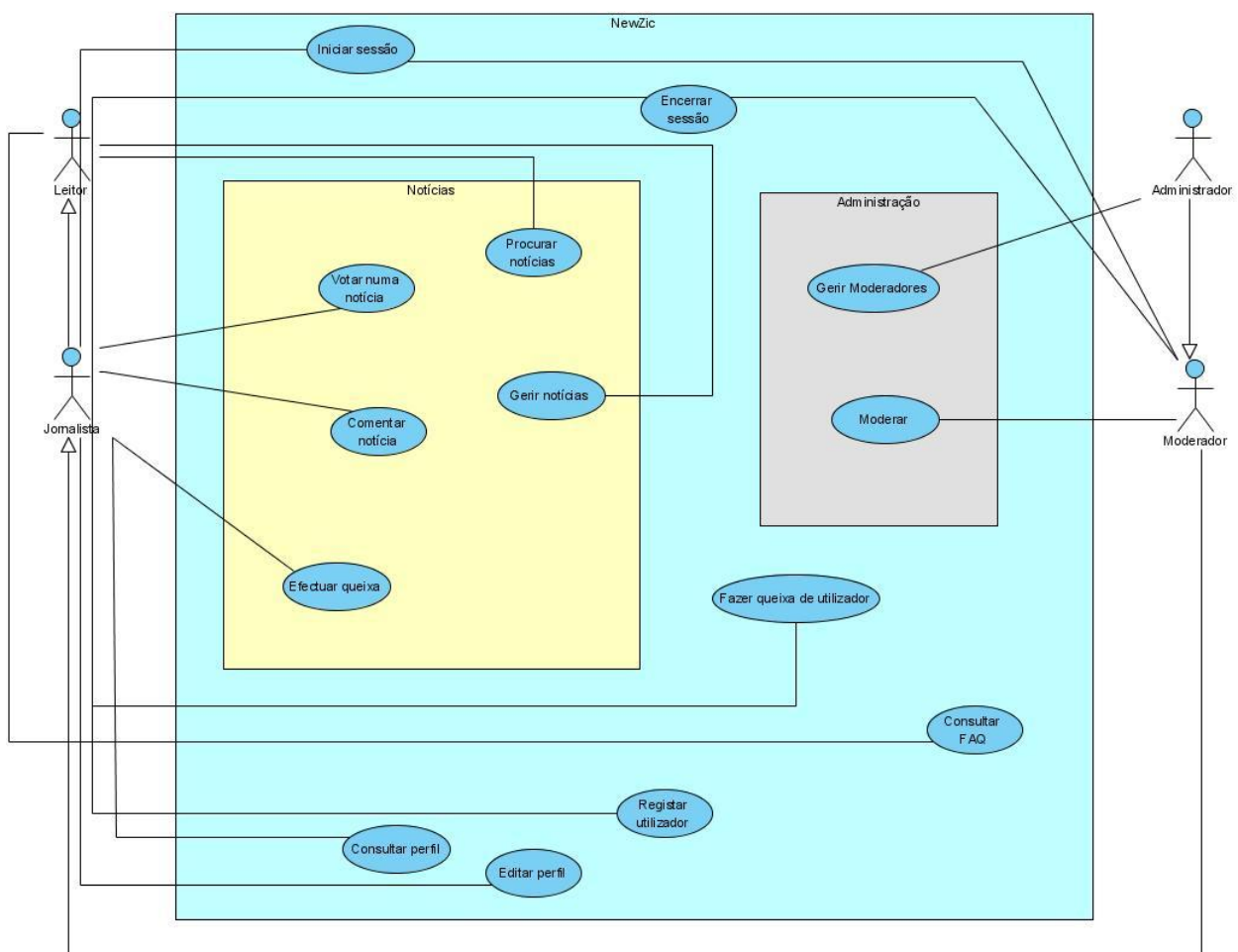


Ilustração 2 - Use Case "Geral"

4.1.1 Página Inicial

Quando um utilizador chega à página principal do “Newzic”, tem acesso a um conjunto das oito melhores notícias posicionadas no ranking, a uma funcionalidade de pesquisa, uma página

com perguntas frequentes (FAQ), uma opção para aceder ao *feed* RSS e ainda a opção de efectuar o login.

Nesta página existe uma funcionalidade de autenticação que, dadas as credenciais de acesso, autentica o utilizador ou mostra uma mensagem de autenticação inválida. Caso o utilizador não ser um Jornalista, pode ainda escolher a opção de registo. Esta opção leva o utilizador para a página onde lhe são pedidos o email, nome e palavra chave. Após a introdução destes dados, também é pedida uma validação através de um CAPTHA.

Um utilizador autenticado tem ainda à sua disposição uma opção que o leva para a sua página de perfil, onde poderá visualizá-lo e editá-lo.

4.1.2 Ranking

O Ranking é um valor invisível ao utilizador que o “Newzic” usa para classificar as notícias em termos de importância (quanto maior o *rank*, maior a sua relevância).

A fórmula que calcula o ranking usa a data de inserção da notícia e a pontuação da mesma para determinar o *rank* da notícia. A fórmula é uma função definida por dois ramos:

- Se a notícia tiver sido publicada em menos de um dia, o *rank* é calculado somando dez à pontuação actual. Desta forma, as novas notícias têm uma oportunidade de destaque em relação às notícias mais antigas.
- Se a notícia tiver sido publicada há mais de um dia, o *rank* é calculado dividindo a pontuação da notícia pela sua idade (data actual – data inserção). Desta forma, com o passar do tempo, as notícias vão perdendo destaque, mesmo que tenham uma boa pontuação.

4.1.3 Pesquisa

Na pesquisa é possível pesquisar notícias pelo seu conteúdo, *tags*, título ou autor, sendo ainda possível ordenar as notícias pesquisadas por ranking, por título, por jornalista e por data de submissão (ascendente ou descendente). Após o utilizador ter escolhido a sua pesquisa e a ter executado, é conduzido para a página de resultados de pesquisa. No topo desta página é mostrada uma descrição da pesquisa efectuada, seguida de todas as opções de pesquisa para o caso de ser necessário pesquisar novamente. A parte central é deixada para a lista de notícias encontradas que são mostradas ao utilizador com o título, classificação, data de inserção e ainda um pequeno excerto do seu texto.

4.1.4 Ver/Editar Perfil

A página de perfil do utilizador é constituída por diversos campos que são criados por este aquando do registo no website do “Newzic”. Posteriormente, esses campos podem ser consultados ou alterados se o utilizador assim o entender – consulta do perfil e edição do mesmo, respectivamente.

Quando o utilizador opta por consultar um perfil, são-lhe apresentados todos os campos respeitantes. O utilizador registado também pode alterar o seu perfil. Para que isto aconteça, o utilizador tem de aceder ao seu perfil e posteriormente seleccionar a opção “Editar perfil”. De seguida, são-lhe apresentados todos os campos, mas com a possibilidade de serem editáveis. O utilizador altera os campos que desejar nas respectivas caixas de texto. Depois de ter editado todos os campos que pretendia, o utilizador tem duas opções: Editar e Voltar.

Se o utilizador seleccionar a opção “Editar”, todas as alterações feitas no seu perfil vão ser actualizadas na base de dados e o *website* apresenta-lhe uma mensagem de sucesso relativamente à edição do seu perfil e mostra no ecrã o seu novo perfil.

Se, por outro lado, e depois de alterados todos os campos que pretendia, o utilizador decidir que não quer editar o seu perfil, pode escolher a opção “Voltar” e os campos ficam com a informação que tinham antes de serem alterados e portanto não é feita nenhuma alteração na base de dados.

4.1.5 Fazer Queixa

Esta página serve para um utilizador poder fazer uma queixa mais formal sobre qualquer assunto, nomeadamente sobre uma notícia, um utilizador ou outro assunto. A estrutura da página consiste num campo para o assunto e um campo para o corpo da mensagem. O utilizador tem que estar autenticado para poder fazer a queixa.

4.1.6 Consultar notícia

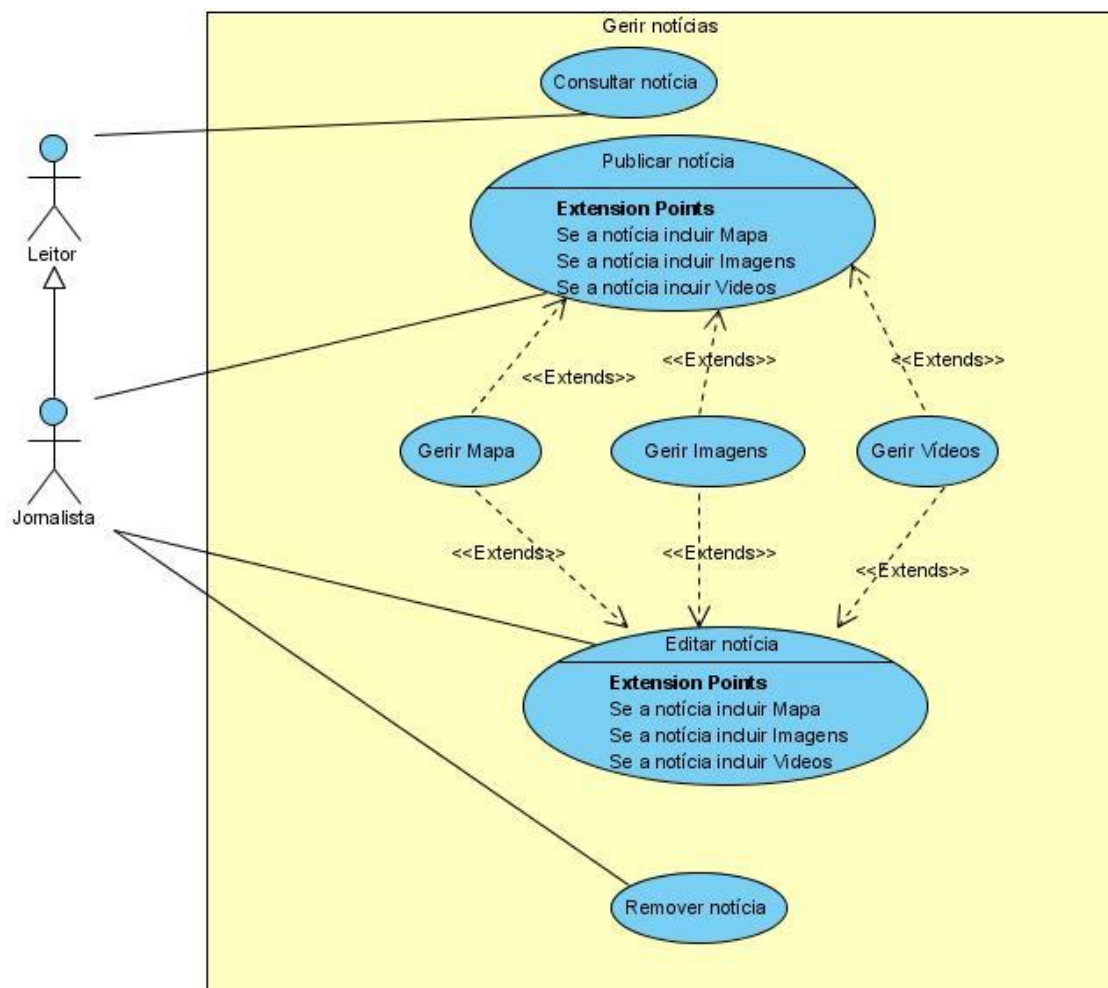


Ilustração 3 - Use Case "Gerir Notícias"

Quando um Leitor selecciona uma notícia é levado para a página da mesma.

No topo da página aparece o título, seguido do texto e as tags. Junto do texto aparece também o autor da mesma e a data de inserção sendo seguida pelas tags. Depois aparecem as imagens relativas à notícia, as quais são apresentadas uma de cada vez com o auxílio de botões para passar à imagem seguinte ou anterior. A seguir às imagens vem uma lista de vídeos, seguida do mapa. No fundo da página existe uma área dedicada ao "Rank", onde é possível votar a notícia, e aos comentários onde é possível aos Jornalistas acrescentarem o seu próprio comentário, utilizando uma caixa de texto para o efeito. Caso um Jornalista deseja denunciar uma notícia, existe, no topo da página, um botão para o fazer.

No caso de o utilizador a visualizar a notícia ser o autor da mesma, no topo da pagina aparecem as opções para "Editar" ou "Apagar". De notar ainda que no caso de ser um Administrador ou Moderador a visualizar uma notícia, estes também tem a opção de "Apagar" disponível.

4.1.7 Publicar/Editar Notícia

Para um Jornalista publicar ou editar uma notícia, existe uma página dedicada a esse efeito. Nesta página aparece um formulário para preencher toda a informação necessária para a publicação. No caso de ser uma notícia já publicada, os campos já se encontram preenchidos. O formulário é constituído pelo seguinte:

- Uma caixa para escrever o título
- Uma caixa para escrever o texto
- Um lugar para gerir as imagens
- Um lugar para gerir as hiperligações dos vídeos
- Um mapa onde é possível adicionar pontos ou *tours*
- Um lugar para gerir as *tags*

A gestão das imagens da notícia é feita por um pequeno controlo de onde é possível visualizar as imagens uma a uma, inserir novas ou remover existentes.

No caso dos vídeos e *tags*, é fornecida uma lista e as funcionalidades necessárias para a gestão destes.

A gestão de pontos e *tours* é feita seleccionando os pontos no mapa e escolhendo a opção pretendida. As opções disponíveis são a adição de um ponto, remoção de um ponto e adição de uma *tour* que nada mais é que um conjunto de pontos.

Após o preenchimento de todos estes campos, o Jornalista pode passar então à publicação, seleccionando a opção para esse efeito. No fundo da página há três opções: uma para fazer a publicação, outra para limpar todos os dados já inseridos e por fim outra que apaga a notícia no caso de já ter sido publicada. Esta última opção só aparece activa se a notícia já tiver sido publicada.

4.1.8 Moderar

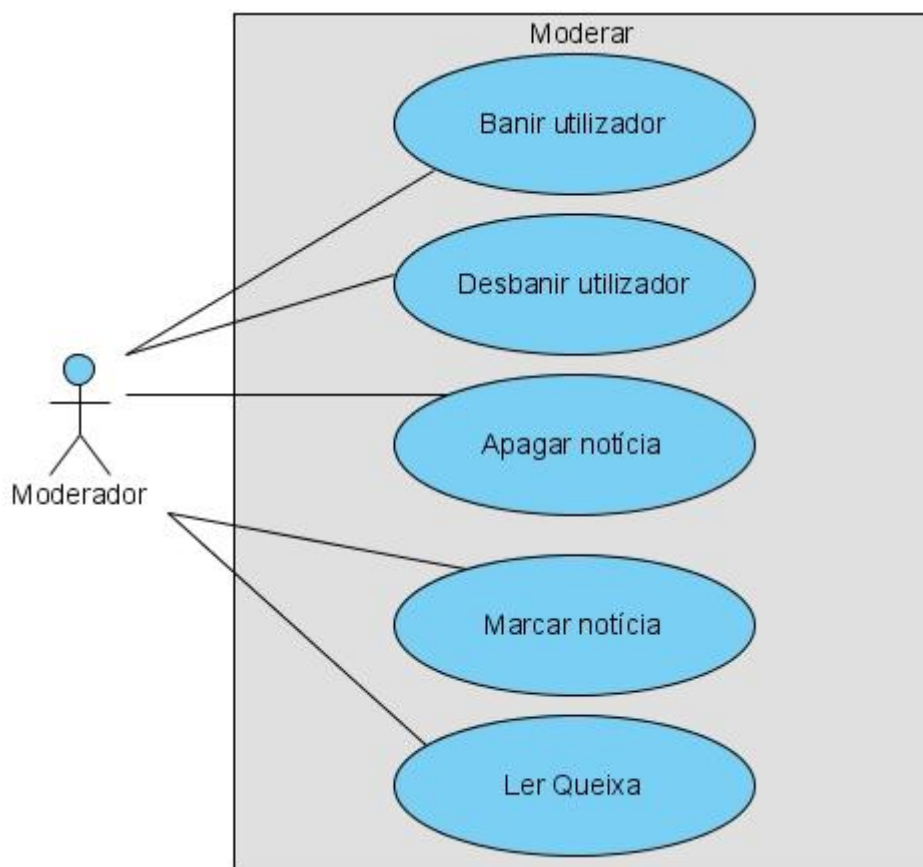


Ilustração 4 – Use Case “Moderar”

Na página de moderação, o Moderador tem acesso às seguintes opções: Gerir Jornalistas e Gerir Notícias reportadas.

Na opção de Gerir Jornalistas, existem as opções de Banir, levantar Ban (caso ele esteja banido) e ver o perfil de um Jornalista.

- Ver Perfil: Ao consultar o perfil de um jornalista, o Moderador é confrontado com uma página que apresenta todos os dados disponíveis fornecidos pelo jornalista quando este se regista:
 - Nome de utilizador
 - Email
 - Notícias publicadas pelo utilizador
 - Status do utilizador

Esta página de perfil é apresentada ao Moderador da mesma forma que é apresentada a outro utilizador. No entanto, este tem acesso a mais opções de moderação tais como banir utilizador. Ao clicar nesta opção, é pedido o tempo de banimento. De seguida o utilizador passa a constar na lista de utilizadores banidos da base de dados.

Após seleccionar uma notícia, o Moderador visualiza-a como um utilizador normal, no entanto tem a possibilidade de removê-la ou marcá-la. Neste último caso, quando uma notícia for editada, passa a não estar marcada após a edição.

- Banir: Quando um Moderador escolhe a opção de “Banir”, o mesmo é levado para a página de Banir Jornalista onde é apresentado o formulário para esse efeito.
- Levantar Ban: Esta opção retira o estado de banido de um Jornalista.

Na opção “Gerir Noticias Reportadas”, o Moderador tem acesso às notícias reportadas. Quando entra nesta página, são apresentadas todas as noticias reportadas não resolvidas nas quais o Moderador pode efectuar as seguintes operações:

- Consultar Noticia: Esta opção permite ao Moderador consultar a notícia. No caso de a notícia não pertencer a um Administrador ou Moderador na página de consulta da notícia é-lhe mostrado a opção adicional “Apagar Noticia”
- Apagar: Permite apagar a notícia, no entanto se essa notícia pertencer a outro Moderador ele não a pode apagar
- Resolvido: Permite ao Moderador marcar a notícia como resolvida desaparecendo assim da lista de notícias reportadas.

4.1.9 Administrar

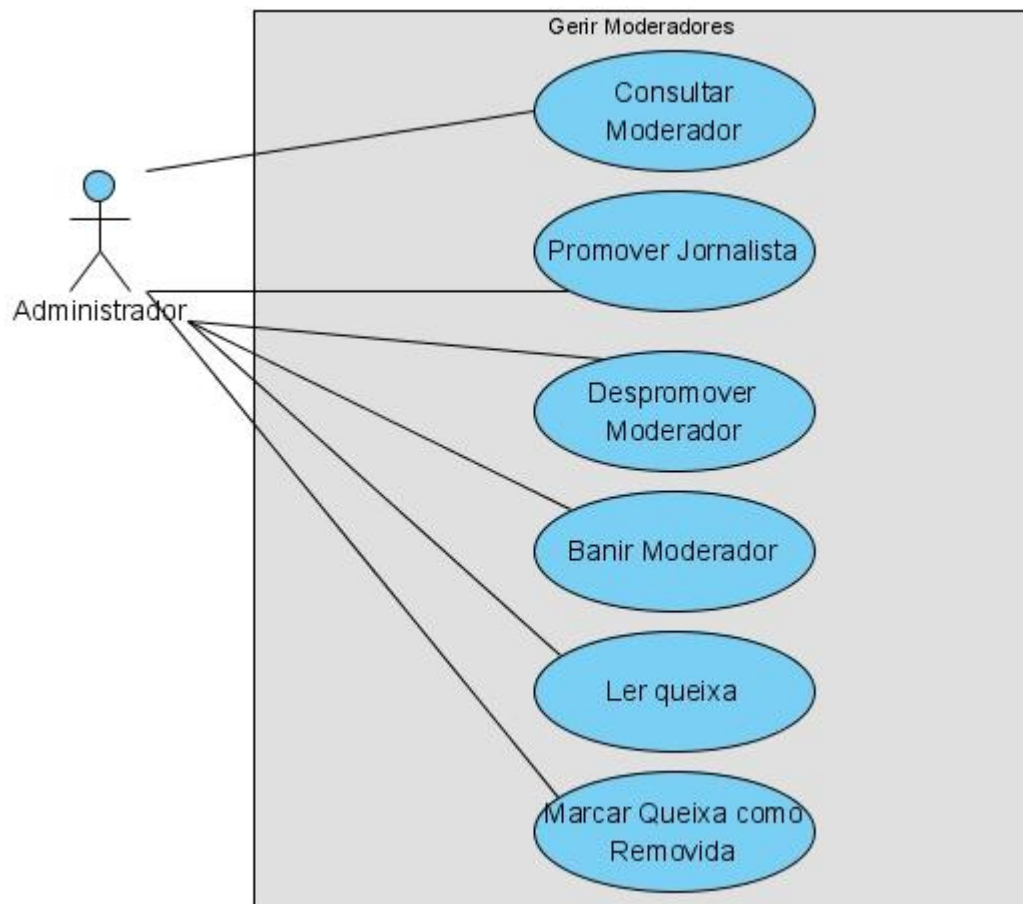


Ilustração 5 - Use Case "Gerir Moderadores"

Administrador, tal como qualquer outro utilizador registado, e após efectuar o login, terá acesso a uma zona específica do site para desempenhar as suas funções. Está secção do "Newzic" da pelo nome de "Administração".

Quando o Administrador entra nesta secção, é apresentado com duas opções: aceder à "Gerir Queixas dos Utilizadores" ou "Gerir Moderadores e Promover Jornalistas".

Ao seleccionar a opção de "Gerir Queixas dos Utilizadores", o Administrador passa a visualizar um género de caixa de mensagens, onde lhe é apresentada uma lista com as queixas não resolvidas que existem no sistema. O Administrador pode então seleccionar uma das queixas para a ler e/ou marcar como resolvida.

Se o Administrador escolher a opção de "Gerir Moderadores e Promover Jornalistas", é mostrada uma lista que por defeito encontra-se preenchida com os Moderadores do "Newzic",

mas existe a possibilidade de alternar para uma lista que apresenta apenas os Jornalistas presentes no sistema. Também será possível ao Administrador procurar na lista apresentada por um Moderador/Jornalista pelo seu nome de utilizador através de uma caixa de texto e um botão de pesquisa.

Se o Administrador estiver a visualizar a lista de Moderadores as opções disponíveis são as seguintes:

- Ver perfil: Esta opção permite ao Administrador consultar o perfil de um Moderador seleccionado.
- Despromover ou banir um Moderador: Ambas as opções estão disponíveis quando o Administrador está a visualizar o perfil de um Moderador. Caso seja seleccionada a opção de banir, é mostrada uma caixa com o tempo de expulsão que deverá ser obrigatoriamente preenchida. Alternativamente, o Administrador pode optar por uma expulsão permanente
- Banir: Esta opção permite ao Administrador banir um Moderador
- Levantar Ban: Esta opção permite ao Administrador retirar o estado de banido de um Moderador

No caso da lista seleccionada ser a de Jornalistas as opções são as seguintes:

- Promover: Esta opção fica disponível quando um Jornalista se encontra seleccionado e permite ao Administrador atribuir o cargo de Moderador a um Jornalista.
- Ver perfil: Esta opção permite ao Administrador consultar o perfil de um Jornalista seleccionado.

4.2. Web Service

O Web Service disponibiliza alguns métodos que podem ser invocados remotamente por outras aplicações. Os métodos disponibilizados são distintos para diferentes níveis de utilizador.

De seguida são apresentados os diferentes métodos de acordo com o nível de utilizador.

4.2.1 Leitor

-List<NoticiaWrap>::searchNoticia(String::query,Int32::type)

Permite procurar notícias.

Recebe uma palavra (ou conjunto de palavras), o tipo de procura (título, *tag* ou autor) e devolve uma lista com os números identificadores das notícias resultantes da procura. Devolve uma lista vazia se não encontrar nenhum resultado.

-NoticiaWrap::getNoticia(Guid::idNoticia)

Permite obter uma notícia através do seu número identificador.

Recebe um número identificador duma notícia e devolve todo o seu conteúdo. Lança uma excepção caso o identificador não corresponda a nenhuma notícia no sistema.

-List<ImagemWrap>::getImagensOfNoticia(Guid::idNoticia)

Permite obter imagens de uma notícia através do seu número identificador.

Recebe um número identificador de uma notícia e devolve as imagens nela presentes. Lança uma excepção caso o identificador não corresponda a nenhuma notícia no sistema.

-List<MapaWrap>::getMapOfNoticia(Guid::idNoticia)

Permite obter o mapa de uma notícia através do seu número identificador.

Recebe um número identificador de uma notícia e devolve o mapa nela presente. Lança uma excepção caso o identificador não corresponda a nenhuma notícia no sistema.

-List<VideoWrap>::getVideosOfNoticia(Guid::idNoticia)

Permite obter os vídeos de uma notícia através do seu número identificador.

Recebe um número identificador de uma notícia e devolve os vídeos nela existentes. Lança uma excepção caso o identificador não corresponda a nenhuma notícia no sistema.

4.2.2 Jornalista

-Void::publicar(NoticiaWrap::noticia, List<ImagemWrap>::imagens, List<VideoWrap>::videos, MapaWrap::mapa, String::token)

Permite a um Jornalista publicar uma notícia.

Recebe uma notícia e um token (gerado aquando o login). Lança uma excepção caso o utilizador não tenha efectuado login ou caso os campos da notícia estejam erroneamente preenchidos.

-Void::editar(NoticiaWrap::noticia, List<ImagemWrap>::imagens, List<VideoWrap>::videos, MapaWrap::mapa, String::token)

Permite a um Jornalista editar uma notícia.

Recebe uma notícia e um token (gerado aquando o login). Lança uma excepção caso o utilizador não tenha efectuado login, caso os campos da notícia estejam erroneamente preenchidos ou a notícia não exista no sistema.

-Void::votar(Guid::idNoticia, String::token)

Permite a um jornalista votar numa notícia.

Recebe um número identificador de notícia e um token (gerado aquando o login). Lança uma excepção caso o utilizador não tenha efectuado login, caso a notícia não exista ou caso o jornalista já tenha votado nessa notícia.

-String::login(String::email,String::password)

Permite que um Jornalista inicie uma sessão.

Recebe os dados necessários para fazer uma autenticação e devolve um token único que identificará o utilizador durante a sessão.

Lança uma excepção se as credenciais não forem válidas.

-Void::logout(String::token)

Permite a um Jornalista terminar a sessão actual.

Recebe o token gerado aquando o login que identifica a sessão. Lança uma excepção caso o token não identifique nenhuma sessão actual.

Uma sessão termina se o Jornalista invocar o método de logout ou automaticamente ao fim de 24horas após o inicio da sessão.

4.3. Diagrama de Classes

Embora os diagramas de sequência sejam bastante úteis para ajudar a planear as várias classes e módulos do projecto, à medida que se vão estudando as funcionalidades, não são muito bons quando se deseja obter uma visão geral de toda a estrutura do sistema. Para tal, existem um outro conjunto de diagramas em UML – os Diagramas de Classe – que são bastante mais adequados. Apresentamos de seguida os Diagramas de Classes do “Newzic”, separados por camadas: Apresentação, Negócio e Dados.

4.3.1 Apresentação

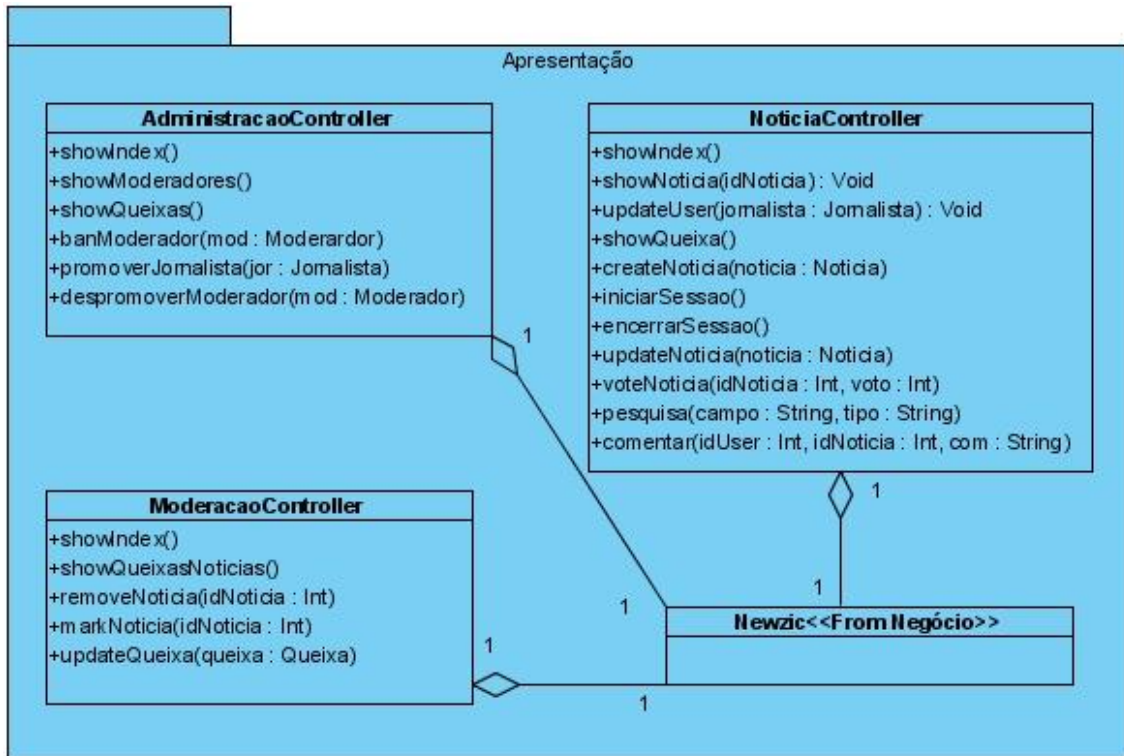


Ilustração 6 - Diagrama de Classes "Camada de Apresentação"

4.3.2 Negócio

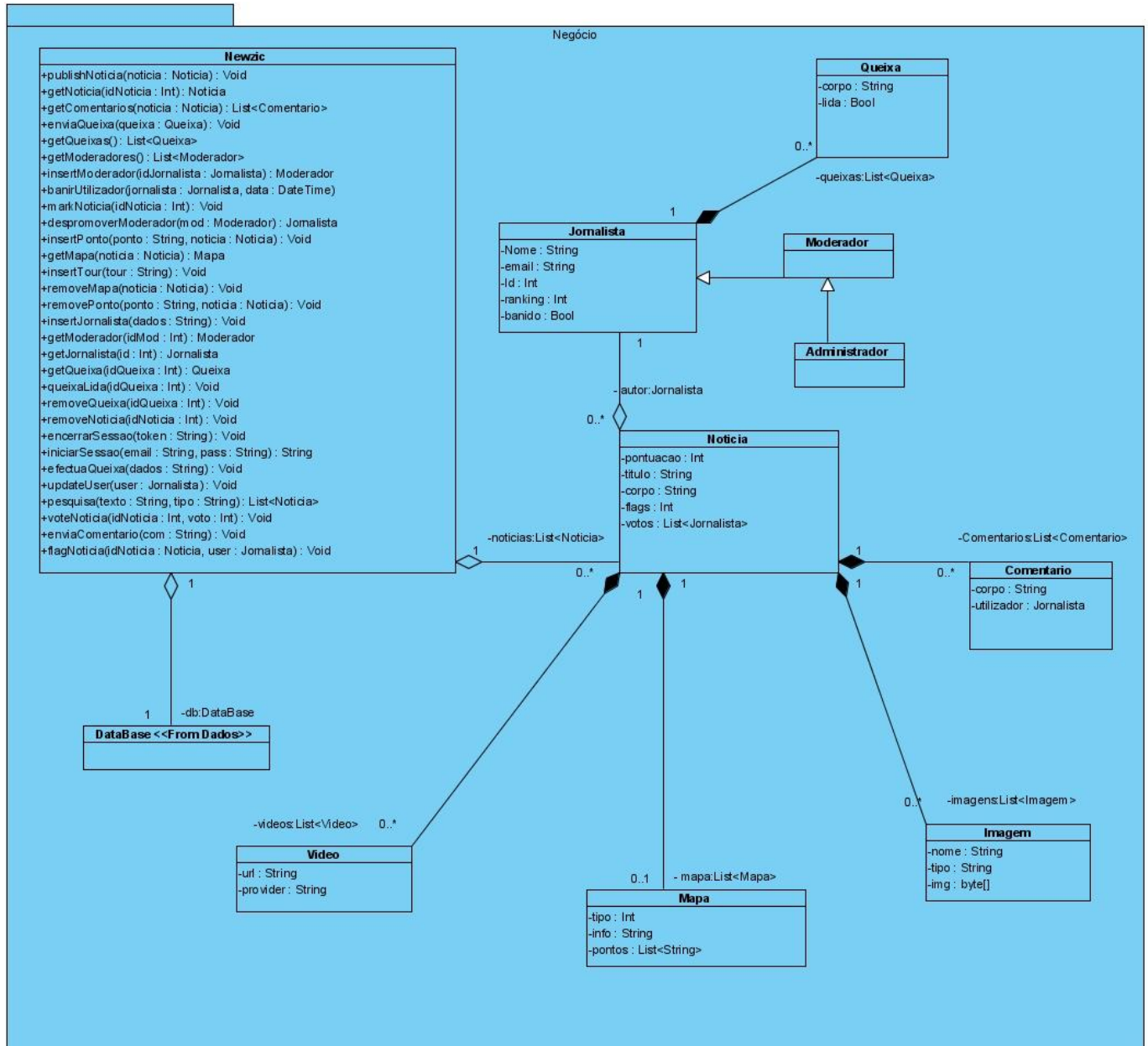


Ilustração 7 - Diagrama de Classes "Camada de Negócio"

4.3.3 Dados



Ilustração 8 - Diagrama de Classes "Camada de Dados"

4.4. Base de dados

Para armazenar toda a informação relacionada com o "Newzic" foi criada uma base de dados relacional capaz de responder às suas necessidades. Para tal foram criadas dez tabelas.

De seguida apresentamos os diagramas conceptual e lógico da base de dados do "Newzic".

4.4.1 Diagrama Conceptual

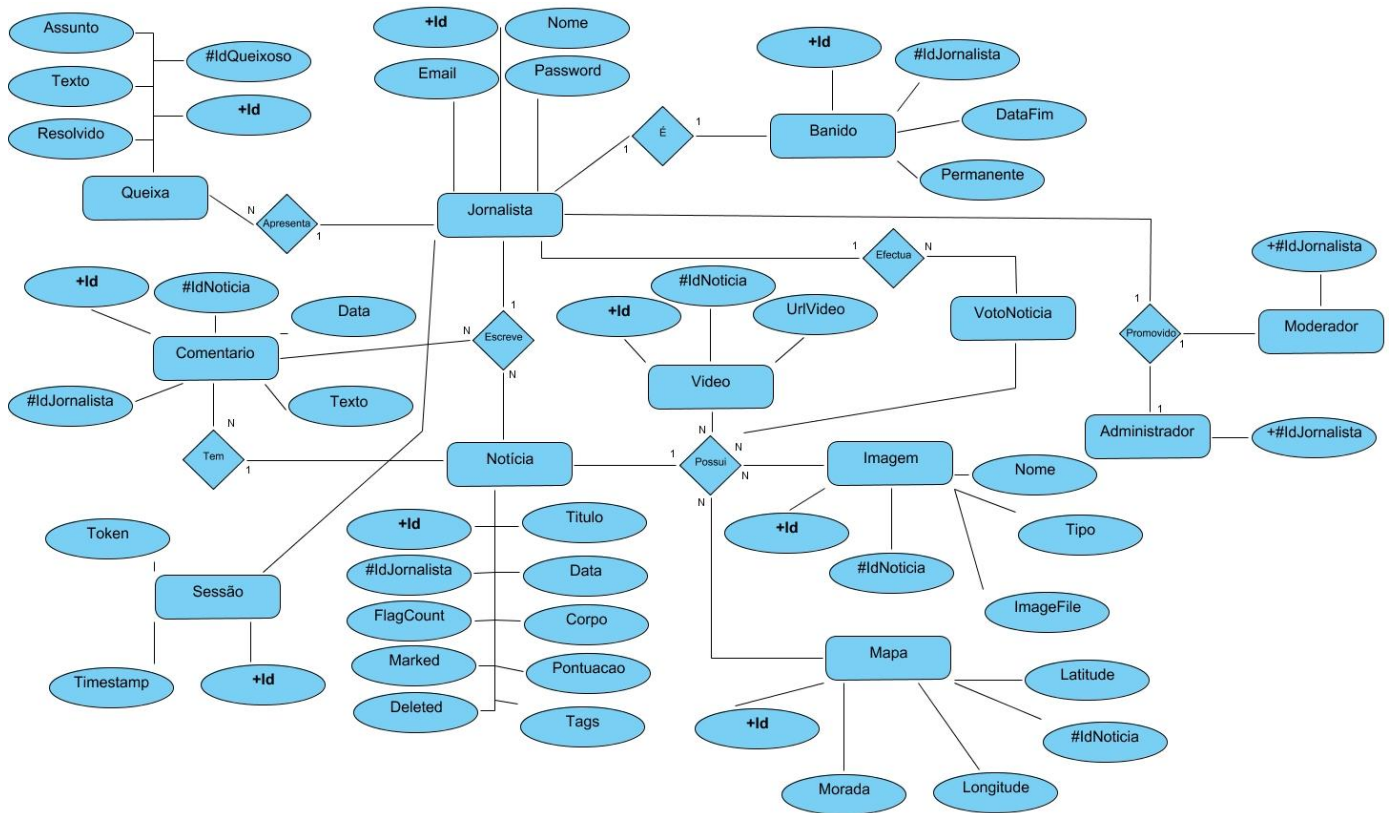


Ilustração 9 - Diagrama Conceptual da Base de Dados

4.4.2 Diagrama Lógico

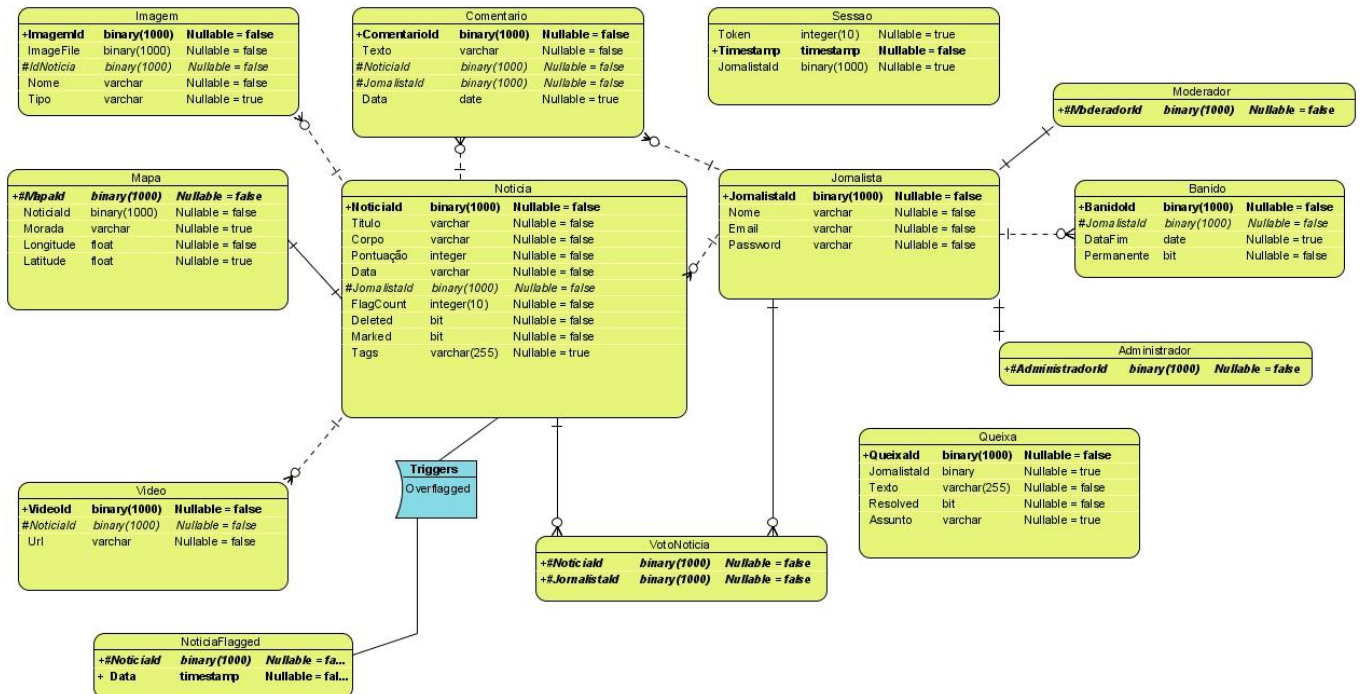


Ilustração 10 - Diagrama Lógico da Base de Dados

Para além destas tabelas foram ainda criados *stored procedures*, *triggers* e *views*.

O *trigger* “Overflagged” insere um novo registo na tabela “FlaggedNoticia” sempre que o campo “Flagged” da tabela “Noticia” é actualizado para um valor igual a cinco, repondo também este valor a zero. Esse novo registo contém o “Id” da notícia e a data de invocação do *trigger*.

No final de cada dia é executado o *stored procedure* “checkBan” para verificar se existe algum utilizador a retirar da tabela de utilizadores banidos. Este verifica se a data no campo “DataFim” é superior à data actual retirando da tabela “Banido” os utilizadores que verificarem esta condição.

5. Implementação

Terminada a especificação do projecto, passa-se à próxima fase do processo de desenvolvimento – a implementação. É neste ponto que, a partir da especificação, se vai construir uma aplicação que corresponde aos requisitos elaborados.

O “Newzic” foi implementado sobre a tecnologia .NET da Microsoft. Além disso utilizamos Google Maps para gerar os mapas das notícias.

5.1. Camada de Dados

Nesta subsecção vamos falar de aspectos relacionados com a camada de dados do “Newzic”. Por um lado, falaremos da configuração do motor de base de dados e da criação da BD nesse motor. Por outro, falaremos do uso do *Framework* LINQ assim como da interface “IDataCRUD” que foi especificada como a única classe que comunica com a BD, embora seja genérica para funcionar com as várias entidades presentes no nosso projecto.

Em relação à criação da Base de Dados, utilizamos o motor de Base de Dados SQL Server 2008 da Microsoft. É natural que assim aconteça e faz todo o sentido visto estarmos a trabalhar com tecnologias .NET e o SQL Server possuir excelente integração com projectos desenvolvidos nessas tecnologias. Com isto, o trabalho de configuração para ligar a base de dados ao nosso projecto foi mínimo. Apenas foi necessário adicionar uma nova fonte de dados ao projecto e o Visual Studio encarregou-se do resto.

O LINQ é um componente do Microsoft .NET Framework que adiciona capacidades para colocar questões sobre dados em linguagens .NET. Isto é muito útil pois proporciona a capacidade de escrever expressões de forma mais clara e natural para seres humanos.

O interface “IDataCRUD” é parametrizado com um objecto do tipo de uma entidade presente nas classes acima mencionadas e a sua principal responsabilidade é facultar a porta de acesso da camada de negócio com a BD. O seu nome deriva das funcionalidades genéricas que disponibiliza a qualquer entidade, estas contemplam a Criação (*Create*), Leitura (*Read*), Actualização (*Update*), Remoção (*Delete*) de dados representativos desta entidade na base de dados, assim como uma operação para registar a persistência dos dados criados, alterados ou removidos. É de notar que todas estas alterações são feitas transicionalmente, isto é, as modificações são executadas de forma atómica, sem riscos de ficarem “a meio”.

5.2. Camada de Negócio

Os esqueletos das classes que representam as nossas entidades de domínio foram criadas com a ajuda da funcionalidade do "LINQ to SQL" do visual studio. Esta funcionalidade permite (com uma simples operação de "drag-n-drop") criar classes que representam os dados armazenados na base de dados.

De seguida implementamos as regras de negócio estendendo cada classe respectivamente de acordo com o especificado. Esta camada não se revelou muito complicada de implementar visto que todos os elementos do grupo de desenvolvimento estavam bastante à vontade com a linguagem e o paradigma "object oriented".

5.3. Camada de Apresentação

A implementação da camada de apresentação do nosso website pode ser dividida em duas partes distintas. Uma em que usamos uma arquitectura MVC em ASPX.NET para a implementação de todo o *Website* e outra parte para a comunicação com a API do *GoogleMaps* assente em Javascript que posteriormente foi integrada com o *Website*.

Website

Para fazer a construção do Website foram desenhadas várias *Views*, *Models* e *Controllers* de forma a separar a lógica de negócio da lógica de apresentação. Enquanto a *View* serve para apresentar um modelo de forma adequada ao utilizador, o *Controller* é responsável pela entrada de dados, por responder ao utilizador com objectos de modelo e ainda pela validação e filtragem da entrada de dados. O *Model* é usado para gerir o domínio de informação e notificar os utilizadores na eventualidade dos dados serem alterados. De notar ainda que é uma representação da informação da aplicação.

Para ligar a nossa aplicação ao serviço de mapas do Google foi implementado algum código em Java script. É absolutamente necessário para a implementação das operações desejadas sobre os mapas. Estas operações são a criação de um ponto e *tours*.

Embora a construção do website não tenha trazido problemas de maior, o mesmo não se verificou para a inclusão dos mapas. Tal deveu-se à dificuldade de programar em Javascript.

O nosso *Website* permite uma visualização através de um comum computador pessoal. No entanto, também é possível visualizá-lo através de equipamentos móveis, isto é, dispositivos como SmartPhones ou outros dispositivos móveis com *browsers* capazes de consumir conteúdo HTML. Para fazer esta implementação foram criados dois ficheiros CSS que fornecem os estilos necessários para uma correcta visualização do *WebSite*.

Web Service

Para estender as funcionalidades do Website a aplicações externas foi criado um *Web Service*. Para a implementação dos seus *WebMethods* foi usada a plataforma NET que automaticamente trata de todo o protocolo **SOAP** para a troca de informação.

Este protocolo é responsável por fornecer determinadas funcionalidades tais como:

- Login
- Ler notícia
- Procurar notícia
- Publicar notícia
- etc...

Desta forma, qualquer pessoa capaz e com vontade de experimentar, pode integrar funcionalidades do “Newzic” com outra aplicação. Um exemplo dessa integração pode ser uma aplicação ou um *Widget* para *SmartPhones* capaz de fornecer ao utilizador as ultima notícias do “Newzic”.

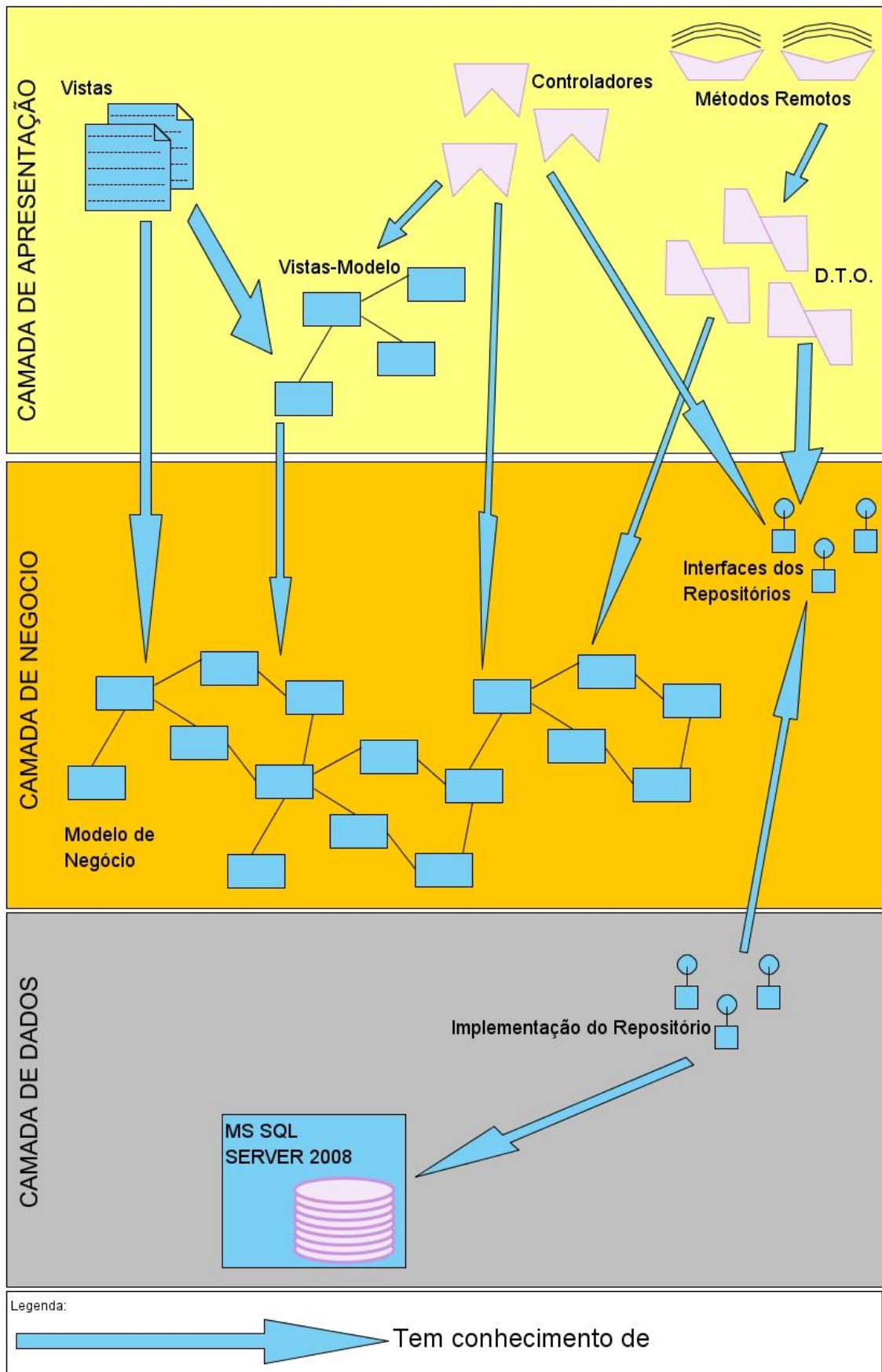


Ilustração 11 - Arquitectura do Sistema

5.4. Screenshots do Website



Ilustração 12 - Página Principal

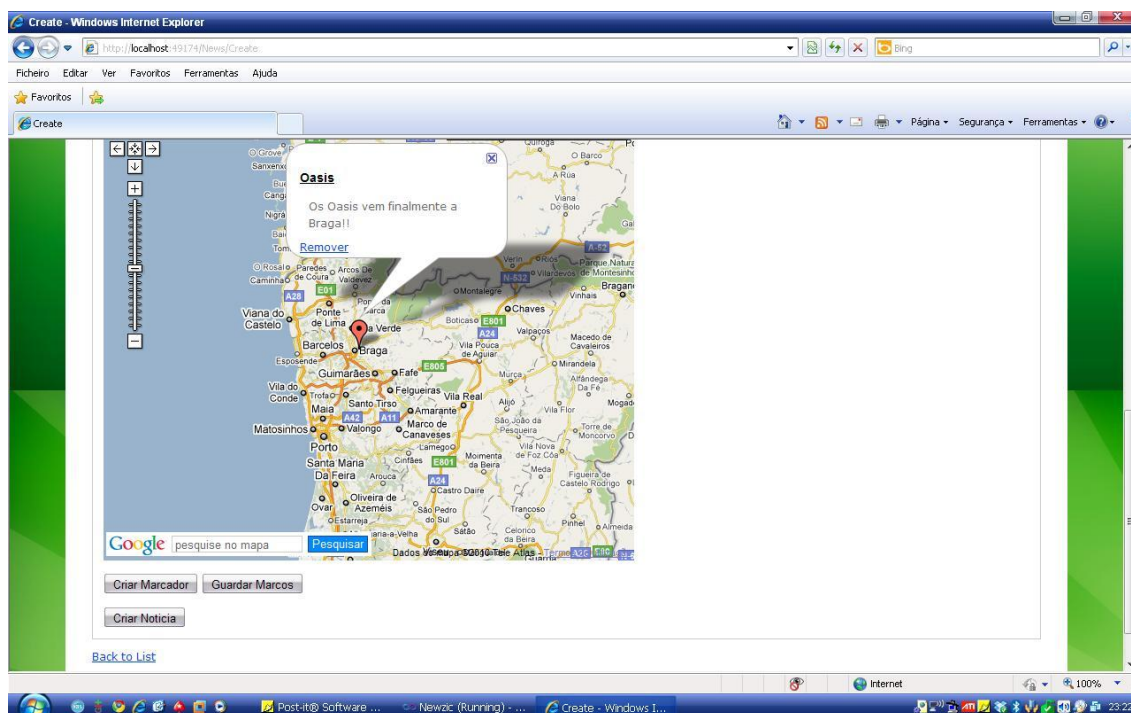


Ilustração 13 - Criar notícia

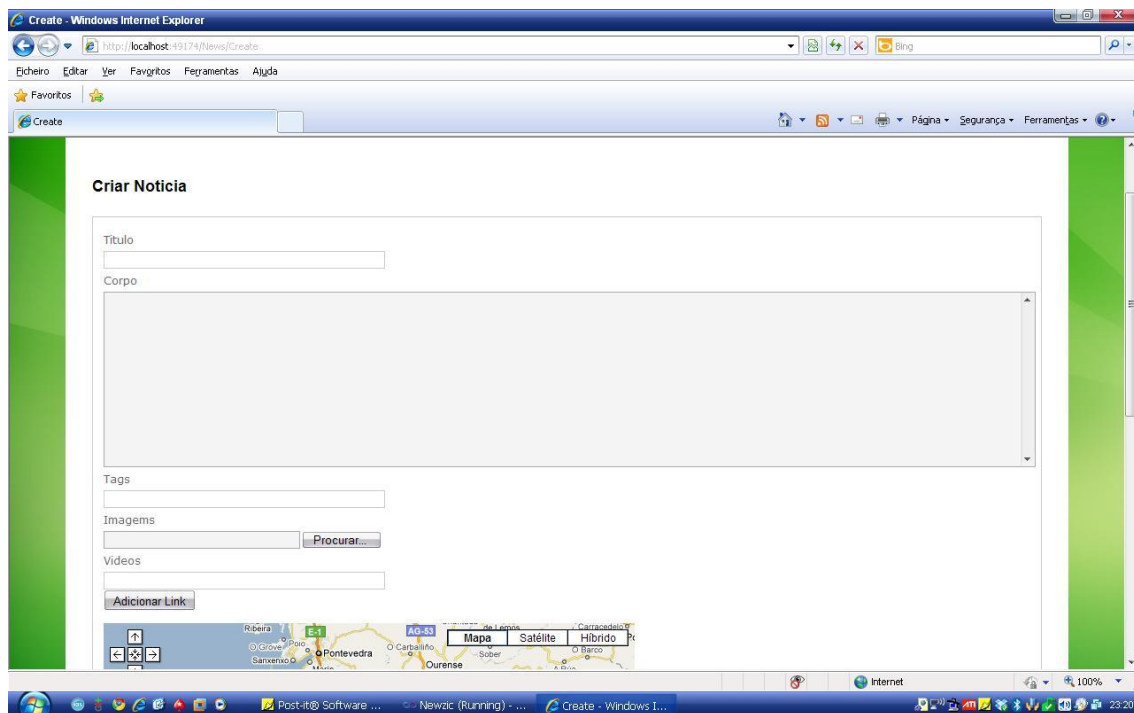


Ilustração 14 - Criar Notícia

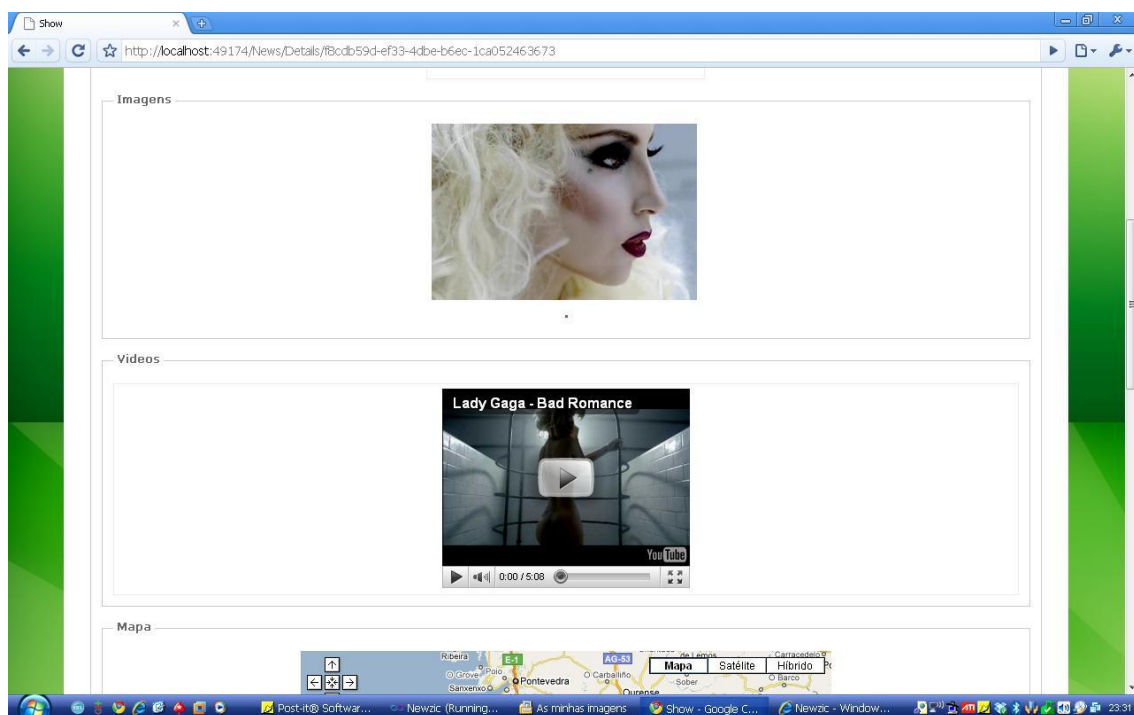


Ilustração 15 - Ver Notícia

6. Mapa do Projecto

Esta secção é um pequeno mapa de desenvolvimento para o “Newzic”. Serve primariamente para a equipa de desenvolvimento se guiar. No entanto, o cliente poderá usá-lo para seguir a evolução do processo de desenvolvimento.

6.1. Modelo de desenvolvimento

O “Newzic” foi desenvolvido de acordo com o Modelo de Desenvolvimento de Software em Cascata com Retrocesso e também com Reutilização de Componentes. Significa isto que por um lado, o “Newzic” foi desenvolvido de forma sequencial. Depois de uma fase estar concluída, iniciou-se a seguinte e assim por diante. A primeira fase foi aprovada, embora com alguns pontos fracos que tiveram de ser posteriormente revistos e corrigidos na segunda fase, pela nossa equipa do projecto. Por vezes foi necessário recuar mais do que uma fase. Essas situações são extremamente indesejáveis quando se tem de alterar demasiada informação, mas tal não foi o nosso caso. Conseguimos perder o menor tempo possível na alteração de fases anteriores.

Reutilização de componentes significa, muito simplesmente, que aproveitamos material já existente que se tornou útil para objectivos do “Newzic”. Trata-se duma prática corrente no desenvolvimento de software.

6.2. Planificação

Nesta secção listamos as várias fases do processo de desenvolvimento e respectivos prazos. A fase actual encontra-se a negrito.

<i>Fase</i>	<i>Descrição</i>	<i>Data de Entrega</i>
1ª	Fundamentação	
	Levantamento de requisitos	22 de Março
	Conclusão da 1ª fase	<u>29 de Março</u>
2ª	Especificação	
	Diagramas de Use Cases	7 de Abril
	Validação da 1ª fase	8 de Abril
	Diagramas de sequência	16 Abril
	Diagramas de Classes e Modelo E-R da Base Dados	22 de Abril
	Conclusão da 2ª fase	<u>26 de Abril</u>
-> 3ª (Fase actual)	Implementação	
	Camada de Negócio concluída	10 de Maio
	Todas as funcionalidades implementadas	24 de Maio
	Conclusão do projecto	<u>28 de Maio</u>
4ª	Apresentação final	Data a definir

Tabela 1 - Planificação do projecto

6.3. Ferramentas e Tecnologias utilizadas

Durante as várias etapas do Projecto utilizámos várias ferramentas e tecnologias. Numa primeira fase, utilizamos papel e lápis para esboçar alguns diagramas. Depois, na segunda fase, utilizámos o *Visual Paradigm* para desenhar esses diagramas em formato digital. Para desenhar o esquema da base de dados utilizamos mais uma vez papel e lápis para fazer esboços. A versão final foi elaborada no *SQL Server*.

As ferramentas e tecnologias utilizadas nesta última fase de projecto foram a plataforma *.NET* (linguagem *C#* e *ASP.NET*) e o *Google Maps* (duas tecnologias requeridas pelo projecto). Para além dessas, utilizamos também o *Visual Studio* como ambiente de desenvolvimento, visto ser o IDE *standard de facto* para *.NET*.

Para controlo de versões, utilizamos o *Microsoft Visual SourceSafe*.

7. Conclusão e Trabalho Futuro

Na primeira fase, tínhamos como objectivo principal definir e aprovar os requisitos para o “Newzic”. Isso foi conseguido com algum sucesso.

Na segunda, tivemos de fazer toda a especificação da aplicação final a desenvolver.

Com efeito, uma boa especificação e planeamento são essenciais para o sucesso dum projecto de software. O “Newzic” não é excepção. Para garantir que o produto final é da maior qualidade possível, tomamos bastante cuidado ao longo de todo o processo de especificação. Para especificar os vários módulos da aplicação, seguimos uma metodologia que já antes deu provas de conseguir bons resultados. Esta metodologia encaixou perfeitamente nos nossos requisitos que já haviam sido elaborados de forma orientada à funcionalidade. Assim, foi apenas uma questão de transcrever as várias funcionalidades para os UCs. A partir daí, refinámos os Diagramas de Sequência a partir dos UCs. Por fim, combinamos a informação de todos os DS no Diagrama de Classes que servirá como guia final para a estrutura final da aplicação.

Em relação à Base de Dados, esta foi a nossa primeira experiência com modelação de BDs, mas achamos o processo bastante simples. A notação E-R é bastante simples e a ferramenta que utilizamos, apesar de extremamente poderosa, é de utilização fácil. Por isso, todo o processo correu de forma suave e livre de problemas.

O Projecto foi finalmente concluído. Foi um processo longo e trabalhoso e embora tenha dado algumas dores de cabeça, estamos muito satisfeitos com o resultado. Foi sem dúvida o nosso maior projecto até à data e permitiu aprender muito sobre os problemas de desenvolver projectos com um grupo de trabalho de várias pessoas. A ferramenta de controlo de versões, que nunca foi vista como muito essencial num projecto com um grupo de 2 pessoas (e que no início deste projecto foi fonte de alguns atrasos nos prazos internos ao grupo) revelou-se fundamental na conclusão do projecto. Outro aspecto que, a nosso ver, não era importante e que a partir deste projecto fará com certeza parte de todos os nossos projectos futuros é um calendário com datas e horas de trabalho específicas e um *roadmap* de projecto detalhado com *deadlines* específicas para diferentes partes do projecto.

A Implementação não foi tão trabalhosa como a Especificação, mas em contrapartida foi a que causou mais dores de cabeça, nomeadamente, a implementação de funcionalidades que exigiram o uso de *Javascript* como o “*image browser*” e os mapas das notícias.

Outro aspecto que importa ser mencionado é o facto de que (embora tenhamos implementado todas as funcionalidades especificadas) termos feito alguns ajustes à Especificação, seja por necessidade (por falta de um total conhecimento sobre as capacidades de uma ferramenta) ou simplesmente por estética. Não foram mudanças radicais como retirar funcionalidades ou substituição de uma funcionalidade por outra completamente diferente. As alterações efectuadas foram coisas mínimas como alteração do nome de um botão ou a mudança do local onde uma imagem de uma notícia aparece em relação ao seu texto. No entanto, acreditamos que em projectos futuros, com mais experiência, a necessidade de alterar a especificação será cada vez menor.

Em suma, estamos bastantes satisfeitos com o resultado final. Implementamos todas as funcionalidades especificadas e produzimos um código bom de manter, com as camadas de apresentação, negócio e dados devidamente divididas, não só logicamente como fisicamente, visto que diferentes camadas foram desenvolvidas em projectos do Visual Studio diferentes.

Esperamos sinceramente que os clientes concordem connosco e que a implementação do nosso projecto seja aprovada com o mesmo grau de sucesso que a fase anterior.

Bibliografia

[B1] Sommerville, I., Software Engineering, 6th Edition, Pearson Education, 2001

[B2] Connolly, T., Begg, C., Database Systems, A Practical Approach to Design, Implementation, and Management , Addison-Wesley, 4^a Edição, 2004

[B#] Evans, Eric, Domain-Driven Design: Tackling Complexity in the Heart of Software, Addison-Wesley, 2003

Referências WWW

- [01] www.wikipedia.org
Enciclopédia online livre e colaborativa. É escrita por várias pessoas de todo o mundo em várias línguas.
- [02] <http://blitz.aeiou.pt/>
Nome de um jornal português, especializado em música e cultura popular. Durante muitos anos foi o único jornal dedicado à música publicado semanalmente em Portugal.
- [03] <http://www.allmusic.com/>
Extensa base de dados em metadata sobre música, parte do grupo All Media Guide. Considerada o padrão de facto pela metadata musical, a base de dados Allmusic é licenciada e usada em centenas de milhares de lojas de música e páginas de compra online.
- [04] <http://www.reddit.com/>
Site de publicações sociais no qual os utilizadores podem postar *links* para conteúdo na Web. Outros utilizadores podem então votar positivamente ou negativamente nos *links* postados, fazendo com que apareçam de uma forma mais ou menos destacada na sua página inicial

Lista de Siglas e Acrónimos

API	Aplication Programing Interface
RSS	Really Simple Syndication
WWW	World Wide Web
XML	eXtensible Markup Language
HTTP	Hypertext Transfer Protocol
HTML	HyperText Markup Language
URI	Uniform Resource Identifier
SOAP	Simple Object Access Protocol
EAI	Enterprise Application Integration
CAPTCHA	Completely Automated Public Turing test to tell Computers and Humans Apart
SQL	Structured Query Language
IDE	Integrated Development Environment
RUP	Rational Unified Process
UML	Unified Modeling Language
UC	Use Case
DS	Diagrama de Sequência
BD	Base de Dados
DC	Diagrama de Classes
ER	Entity – Relationship
SGDB	Sistema Gestor de Base de Dados
MVC	Model View Controller
CSS	Cascading Style Sheets
LINQ	Language INtegrated Query

Anexos

I. Blog

Um *blog* (contração do termo *Web log*) é um site cuja estrutura permite a actualização rápida a partir de acréscimos dos chamados artigos, ou *posts*. Estes são, em geral, organizados de forma cronológica inversa, tendo como foco a temática proposta do *blog*, podendo ser escritos por um número variável de pessoas, de acordo com a política do *blog*.

Muitos *blogs* fornecem comentários ou notícias sobre um assunto em particular; outros funcionam mais como diários *online*. Um *blog* típico combina texto, imagens e *links* para outros *blogs*, páginas da web. A capacidade de leitores deixarem comentários de forma a interagir com o autor e outros leitores é uma parte importante de muitos *blogs*.

Alguns sistemas de criação e edição de *blogs* são muito atractivos pelas facilidades que oferecem, disponibilizando ferramentas próprias que dispensam o conhecimento de HTML. A maioria dos *blogs* são primariamente textuais, embora uma parte seja focada em temas exclusivos como arte, fotografia, vídeos, música ou áudio, formando uma ampla rede de médias sociais. Outro formato é o *microblogging*, que consiste em *blogs* com textos curtos.

Em Dezembro de 2007, o motor de busca de *blogs Technorati* rastreou a existência de mais de 112 milhões de *blogs*. Com o advento do *videoblog*, a palavra *blog* assumiu um significado ainda mais amplo, implicando qualquer tipo de média onde um indivíduo expressa a sua opinião ou simplesmente discorda de um assunto qualquer.

Os *blogs* começaram como um diário *online* e, hoje, são ferramentas indispensáveis como fonte de informação e entretenimento. O que era visto com certa desconfiança pelos meios de comunicação virou até referência para sugestões de reportagem.

A linguagem utilizada pelos utilizadores dos *blogs* foge da rigidez praticada nos meios de comunicação e deixa o leitor mais próximo do assunto, além da possibilidade do diálogo entre comunicador e audiência. Grandes portais de notícias veiculam com frequência informações de *blogs* e dão crédito ao jornalista.

Muitos sites oferecem gratuitamente serviço de hospedagem de *blog* com ferramentas que ajudam na configuração da página na web.

Excerto de texto retirado da Wikipédia e ligeiramente modificado.

II. Web 2.0

Web 2.0 é um termo criado em 2004 pela empresa *O'Reilly Media* para designar uma segunda geração de comunidades e serviços, tendo como conceito a "*Web* como plataforma", envolvendo wikis, aplicações baseadas em folksonomia, redes sociais e tecnologia da informação. Embora o termo tenha uma conotação de uma nova versão para a *Web*, não se refere à actualização nas suas especificações técnicas, mas a uma mudança na forma como ela é encarada por utilizadores e equipas de desenvolvimento, ou seja, o ambiente de interacção que hoje engloba inúmeras linguagens e motivações.

Alguns especialistas em tecnologia, como *Tim Berners-Lee*, o inventor da *World Wide Web* (WWW), alegam que o termo carece de sentido, pois a *Web 2.0* utiliza muitos componentes tecnológicos criados antes mesmo do surgimento da *Web*. Alguns críticos do termo afirmam também que este é apenas uma jogada de marketing.

Excerto de texto retirado da Wikipédia e ligeiramente modificado.

III. Web Service

Web service é uma solução utilizada na integração de sistemas e na comunicação entre diferentes aplicações . Com esta tecnologia é possível que novas aplicações possam interagir com aquelas que já existem e que sistemas desenvolvidos em plataformas diferentes sejam compatíveis. Os *Web services* são componentes que permitem às aplicações enviar e receber dados em formato XML. Cada aplicação pode ter a sua própria "linguagem", que é traduzida para uma linguagem universal, o formato XML.

Para as empresas, os *Web services* podem trazer agilidade para os processos e eficiência na comunicação entre cadeias de produção ou de logística. Toda e qualquer comunicação entre sistemas passa a ser dinâmica e principalmente segura, pois não há intervenção humana.

Essencialmente, o *Web Service* faz com que os recursos da aplicação do software estejam disponíveis sobre a rede de uma forma normalizada. Outras tecnologias fazem a mesma coisa. A título de exemplo, os *browsers* da Internet acedem às páginas Web disponíveis usando por norma as tecnologias da Internet HTTP e HTML. No entanto, estas tecnologias não são bem sucedidas na comunicação e integração de aplicações. Existe uma grande motivação sobre a tecnologia *Web Service* pois possibilita que diferentes aplicações comuniquem entre si e utilizem diferentes recursos.

Utilizando a tecnologia *Web Service*, uma aplicação pode invocar outra para efectuar tarefas simples ou complexas mesmo que as duas aplicações estejam em diferentes sistemas e escritas em linguagens diferentes. Por outras palavras, os *Web Services* fazem com que os seus recursos estejam disponíveis para que qualquer aplicação cliente possa operar e extrair os recursos fornecidos por eles.

Os *Web Services* são identificados por um URI e descritos e definidos usando XML. Um dos motivos que os tornam atractivos é o facto deste modelo ser baseado em tecnologias standards, em particular XML e HTTP. São utilizados para disponibilizar serviços interactivos na Web, podendo ser acedidos por outras aplicações usando, por exemplo, o protocolo SOAP . O objectivo dos *Web Services* é a comunicação "aplicação para aplicação" através da Internet. Esta comunicação é realizada com intuito de facilitar a EAI, que significa a integração das aplicações de uma empresa, ou seja, interoperabilidade entre a informação que circula numa organização nas diferentes aplicações como, por exemplo, o comércio electrónico com os seus clientes e os seus fornecedores. Esta interacção constitui o sistema de informação de uma empresa.

Excerto de texto retirado da Wikipédia e ligeiramente modificado.

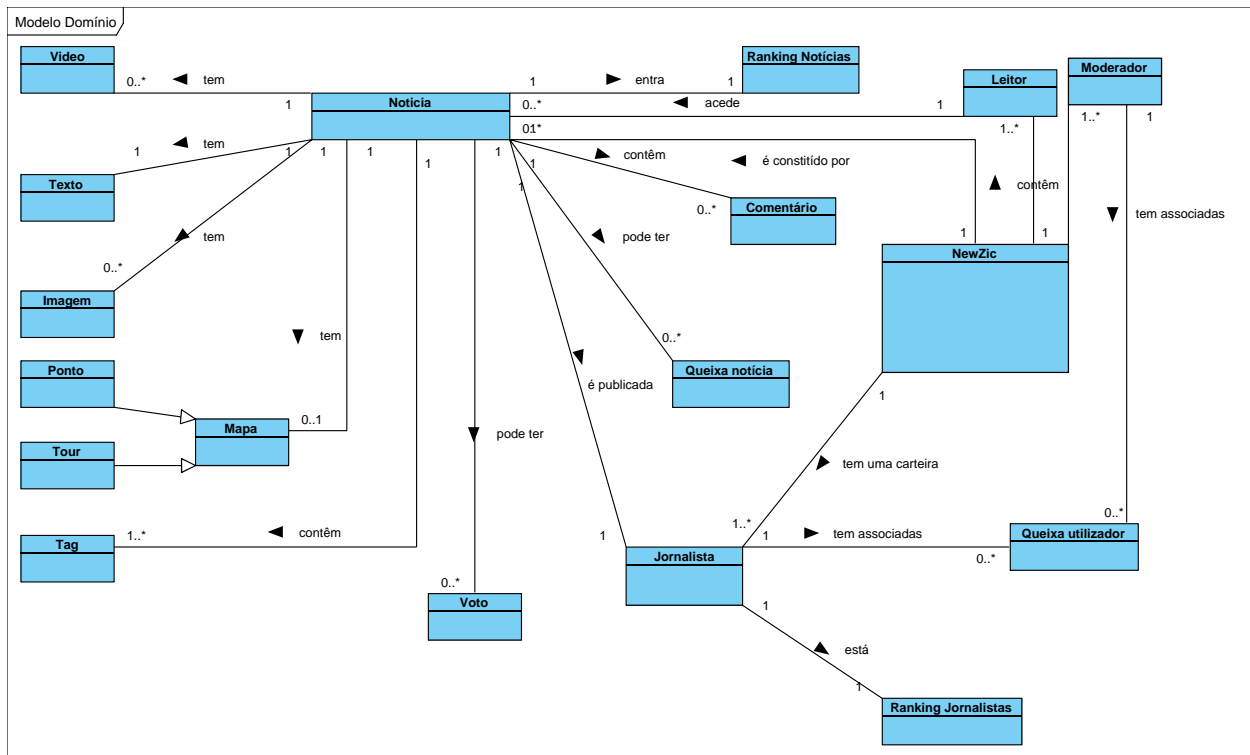
IV. CAPTCHA

CAPTCHA é um acrónimo da expressão "*Completely Automated Public Turing test to tell Computers and Humans Apart*" (teste de *Turing* público completamente automatizado para diferenciação entre computadores e humanos): um teste de desafio cognitivo, utilizado como ferramenta anti-spam, desenvolvido pioneiramente na universidade de *Carnegie-Mellon*. Como o teste é administrado por um computador, em contraste ao teste de *Turing* padrão que é administrado por um ser humano, este teste é na realidade correctamente descrito como um teste de *Turing* reverso.

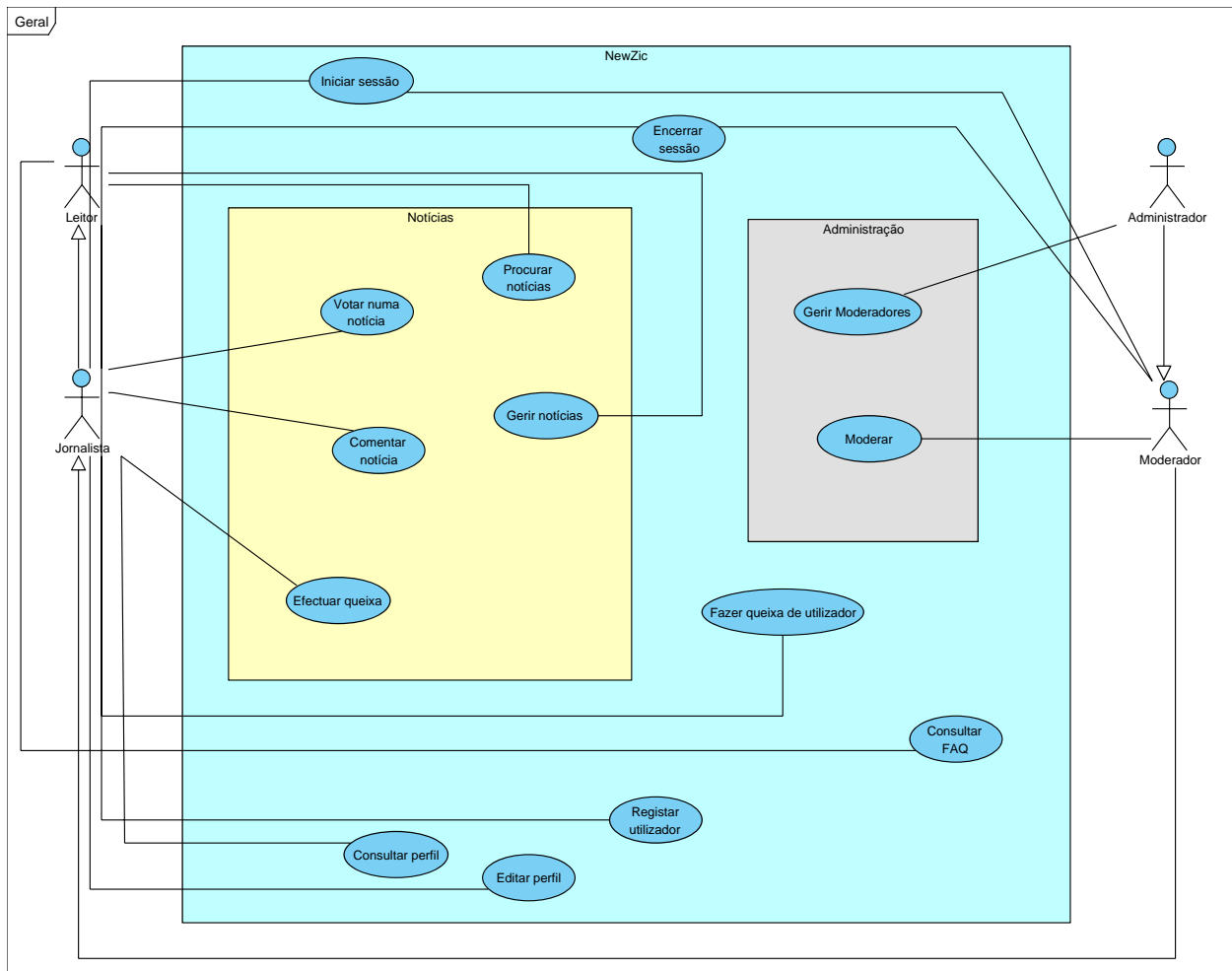
Um CAPTCHA usual envolve um computador (um servidor) que pede que um utilizador termine um teste. Como os computadores são incapazes de resolver o CAPTCHA, todo o utilizador que incorpora uma solução correcta é presumidamente humano. O termo foi inventado em 2000 por *Luis von Ahn*, por *Manuel Blum*, *Nicholas J. Hopper* (todos da universidade do *Carnegie-Mellon*), e por *John Langford* (da IBM).

Excerto retirado da Wikipédia e ligeiramente modificado.

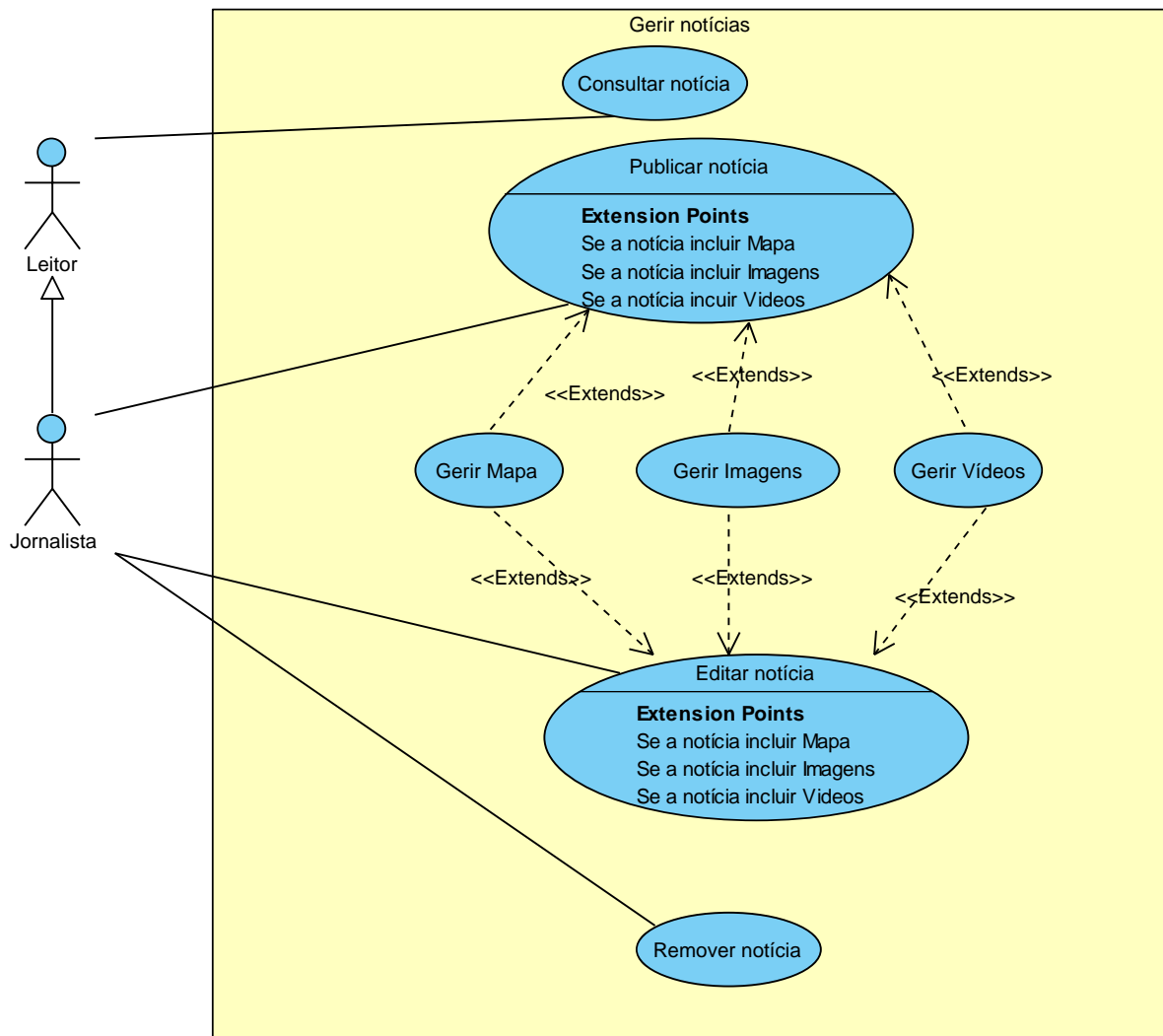
V. Modelo de Domínio



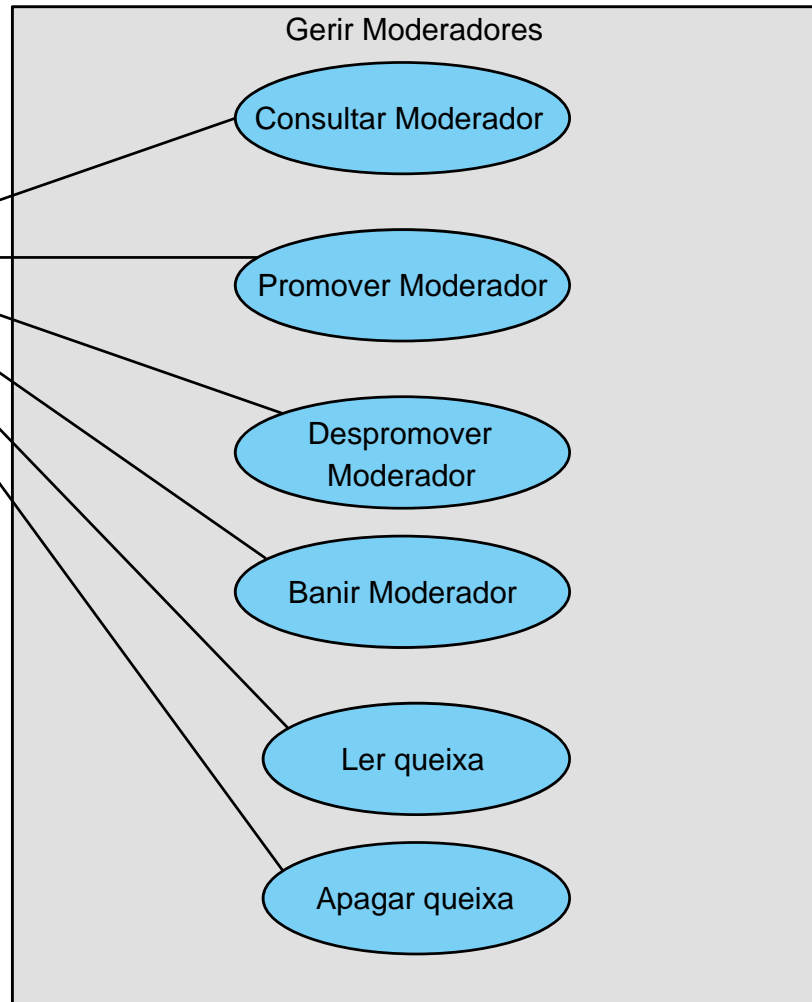
VI. Diagramas de Use Cases



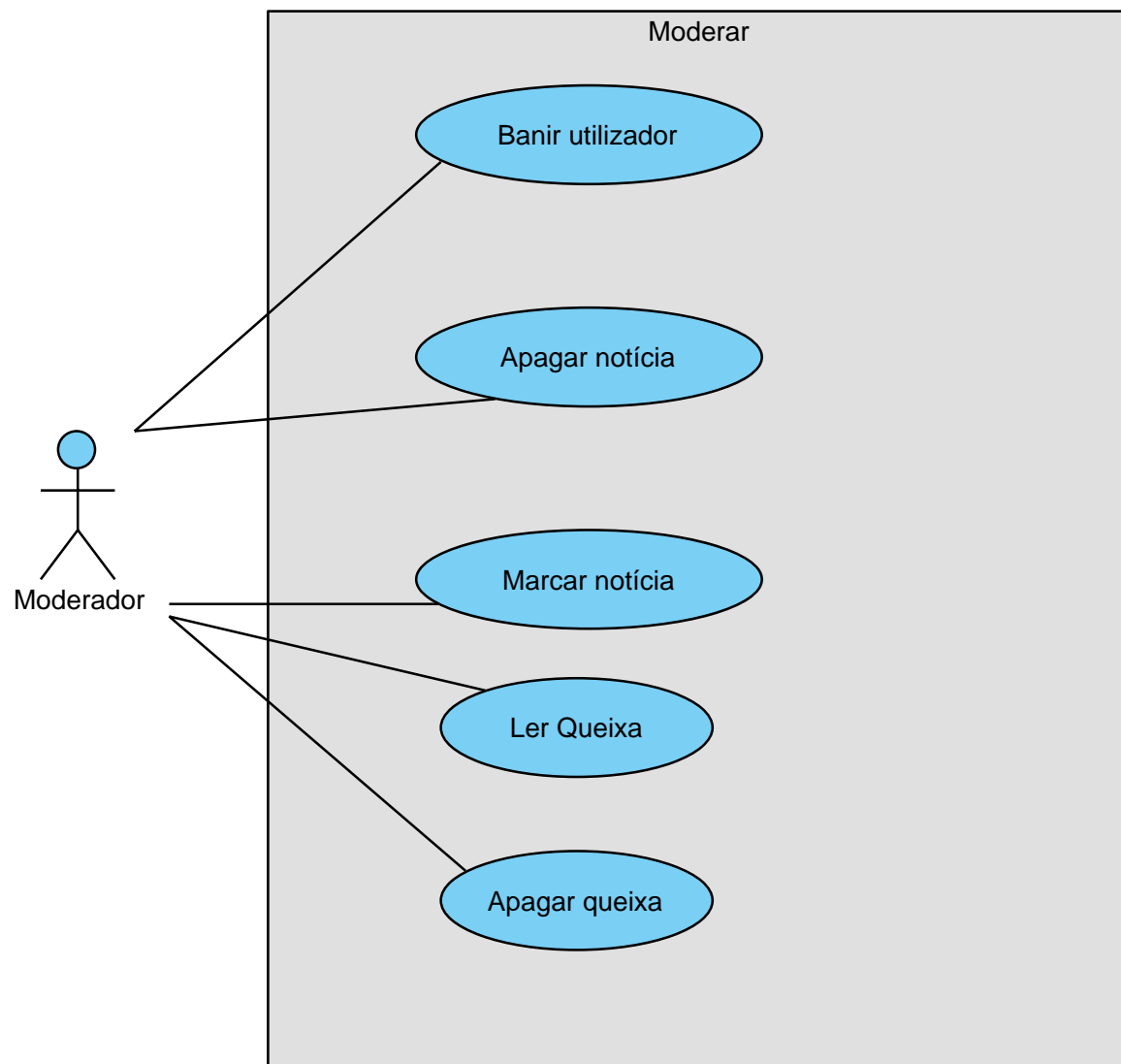
Gerir notícias



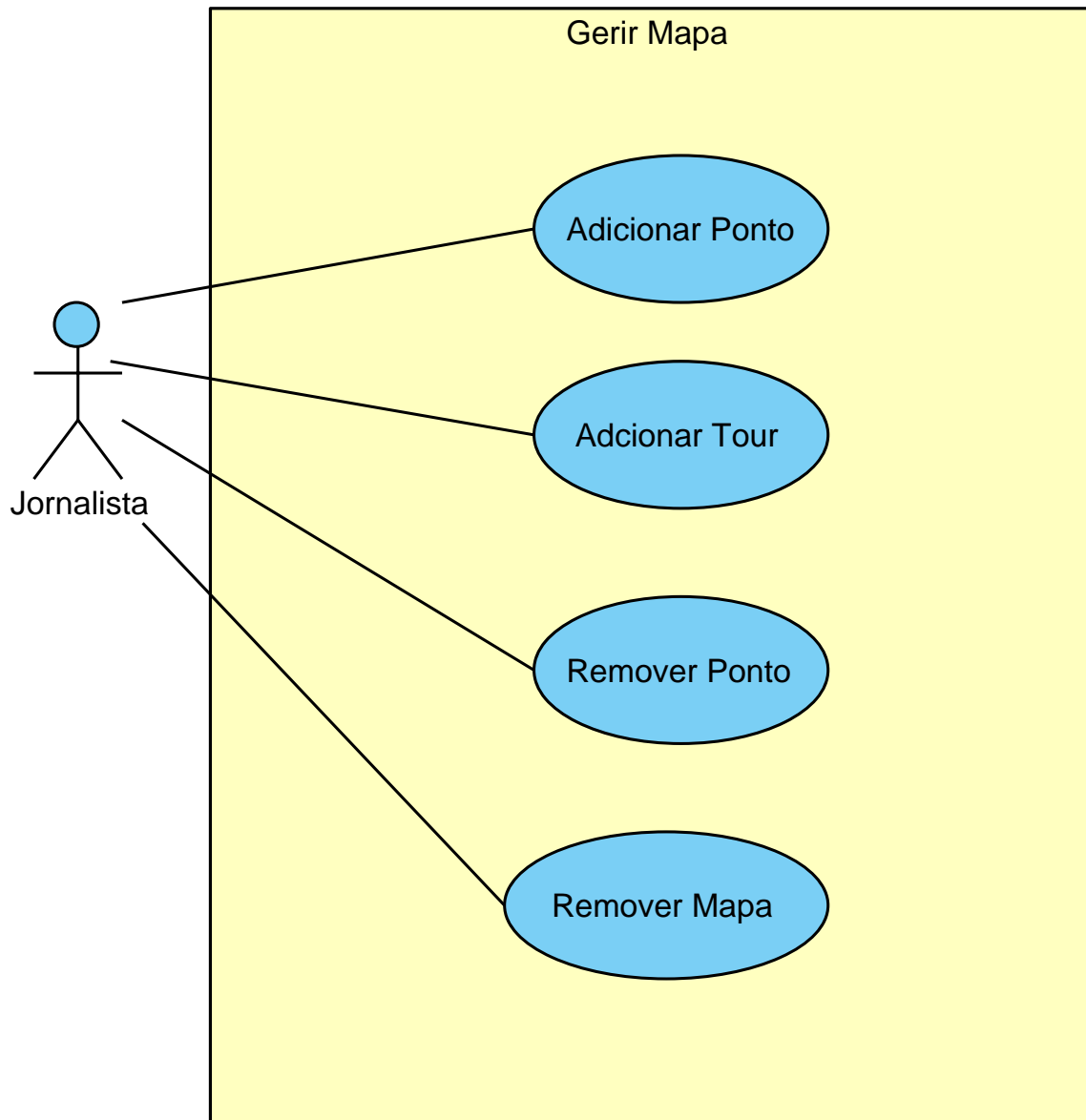
Gerir Moderadores



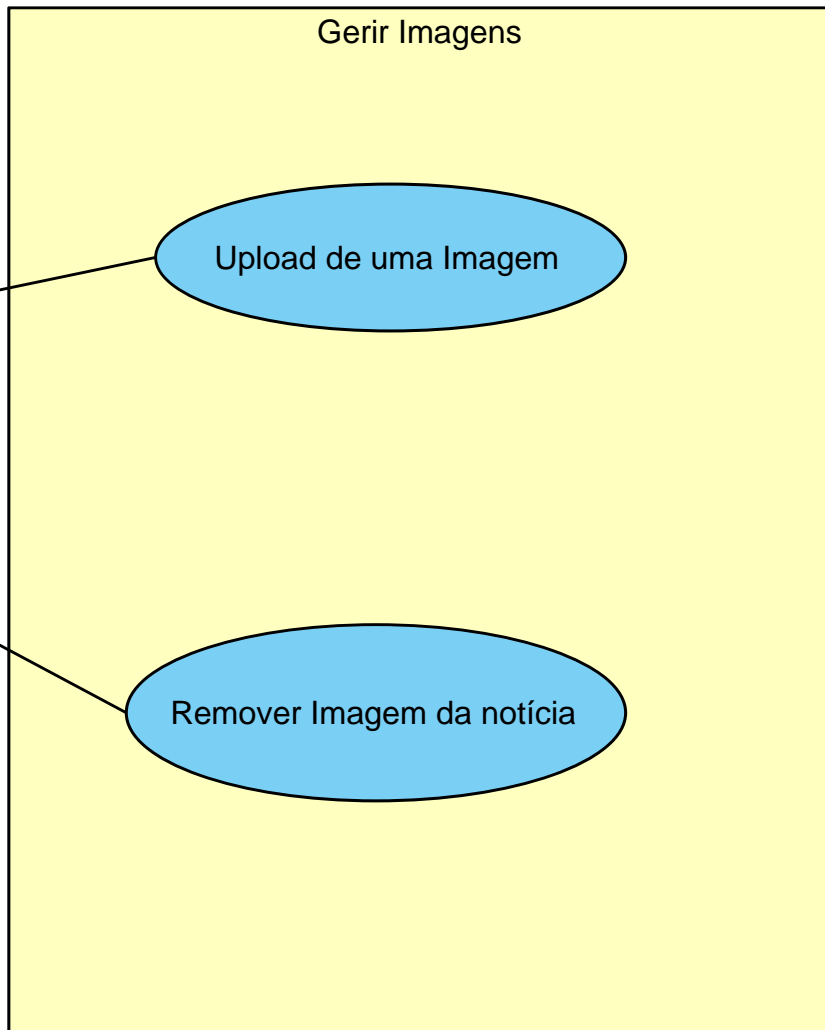
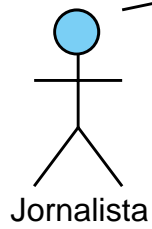
Moderar



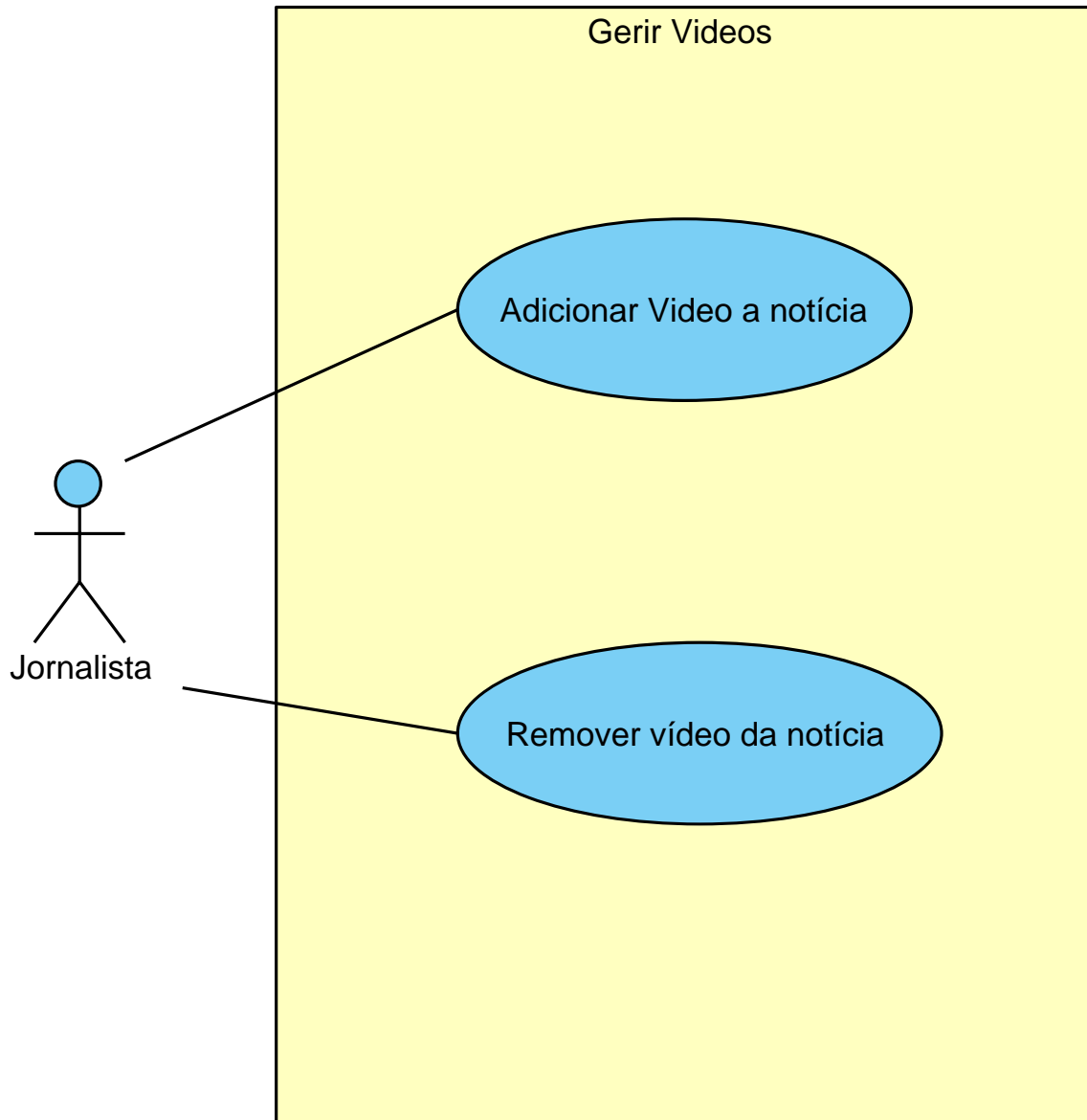
Gerir Mapa



Gerir Imagens



Gerir Videos



Main - Efectuar queixa

Super Use Case		
Author		
Date		
Brief Description	Permite ao Jornalista fazer queixa de um artigo ou notícia	
Preconditions	Utilizador autenticado com notícia seleccionada	
Post-conditions		
Flow of Events		Actor Input
	1	Reporta Notícia
	2	
Aletrnative 2a - Utilizador já apresentou queixa desta notícia		System Response
		Regista Queixa
	1	
		Actor Input
	1	
		System Response
	1	Ignora queixa

Main - Procurar notícias

Super Use Case		
Author		
Date		
Brief Description	Permite ao utilizador procurar notícias	
Preconditions		
Post-conditions		
Flow of Events		Actor Input
	1	
	2	
	3	Selecciona tipo de pesquisa
	4	Insere texto a pesquisar
	5	
	6	
		System Response
		Mostra campo para inserção de texto
		Mostra lista de tipos de pesquisa
		Faz a pesquisa
		Mostra resultados

Main - Consultar Moderador

Super Use Case	Gerir Moderadores	
Author		
Date		
Brief Description	Permite ao Administrador consultar detalhes de um Moderador	
Preconditions	Utilizador autenticado como Administrador; Moderador seleccionado	
Post-conditions		
Flow of Events		Actor Input
	1	Mostra perfil do Moderador Seleccionado

Main - Promover Moderador

Super Use Case	Gerir Moderadores	
Author		
Date		
Brief Description	Permite ao Administrador promover um Jornalista a Moderador ao Sistema	
Preconditions	Utilizador autenticado como Administrador, Jornalista Seleccionado	
Post-conditions	Mais um Moderador no Sistema	
Flow of Events		Actor Input
	1	
	2	Confirma promoção
	3	
	4	
		System Response
		Pede confirmação
		Promove Jornalista a Moderador
		Apresenta mensagem de sucesso

Main - Despromover Moderador

Super Use Case	Gerir Moderadores	
Author		
Date		
Brief Description	Premite ao Administrador despromover um Moderador a Jornalista	
Preconditions	Utilizador autenticado como Adminisrador; Moderador seleccionado	
Post-conditions	Sistema fica com menos um Moderador e mais um Jornalista	
Flow of Events		Actor Input
	1	
	2	Confirma Selecção
	3	
Alternative 1 2a - Cancela selecção		System Response
		Pede Confirmação
		Muda cargo do Moderador para Jornalista
		Actor Input
	1	
		System Response
		Volta ao passo 1

Main - Banir Moderador

Super Use Case	Gerir Moderadores	
Author		
Date		
Brief Description	Permite ao Administrador banir um Moderador	
Preconditions	Utilizador autenticado como Administrador,; Moderador seleccionado	
Post-conditions	Sistema fica com mais um Moderador banido	
Flow of Events		Actor Input
	1	
	2	Insere duraccão
	3	
	4	Confirma selecção
	5	
		System Response
		Pede tempo de duração da expulsão
		Pede confirmação
		Moderador marcado como expulso
Alternative		Actor Input
4a - Cancela selecção	1	
		System Response
		volta ao passo 1
Alternative		Actor Input
4a - Duração da expulsão inválida	1	
	2	
		System Response
		Mostra mensagem de erro
		Volta ao passo 3

Main - Ler queixa

Super Use Case	Gerir Moderadores		
Author			
Date			
Brief Description	Permite ao Administrador ler queixas de Moderadores		
Preconditions	Utilizador autenticado como Administrador		
Post-conditions	Sistema fica com menos uma queixa por ler		
Flow of Events		Actor Input	System Response
	1		Mostra lista de Queixas
	2	Seleciona uma queixa	
	3		Mostra a queixa
	4		Marca queixa como lida
Exception		Actor Input	System Response
1a - Não há queixas	1		Mostra lista vazia

Main - Apagar queixa

Super Use Case	Gerir Moderadores	
Author		
Date		
Brief Description	Permite ao Administrador apagar uma queixa	
Preconditions	Utilizador autenticado como Administrador	
Post-conditions	O Sistema fica com menos uma queixa	
Flow of Events		Actor Input
	1	
	2	Selecciona queixa
	3	
	4	Confirma selecção
	5	
		System Response
		Mostra lista de queixas
		Pede confirmação
		Apaga queixa seleccionada
Alternative		Actor Input
4a - Cancela	1	
Seleccção		
		System Response
		Volta ao passo 1
Exception		Actor Input
1a - Não há	1	
queixas no		
sistema		System Response
		Mostra lista vazia

Main - Iniciar sessão

Super Use Case		
Author		
Date		
Brief Description	Permite a um Jornalista, Moderador ou Administrador autenticar-se	
Preconditions	Utilizador não autenticado	
Post-conditions	Utilizador autenticado com um determinado papel	
Flow of Events		Actor Input
		System Response
	1	Insere email e password
	2	Autentica utilizador
Alternative 2a - autenticação falhada	3	Informa sucesso
		Actor Input
		System Response
	1	Informa insucesso
	2	Volta ao passo 1

Description1 - Encerrar sessão

Super Use Case		
Author		
Date		
Brief Description	Permite a um Jornalista, Moderador ou Administrador autenticado fazer logout	
Preconditions	Jornalista, Moderador ou Administrador autenticado	
Post-conditions	Jornalista, Moderador ou Administrador não autenticado	
Flow of Events		Actor Input
		System Response
	1	Pede logout
	2	Faz logout do utilizador
	3	Informa sucesso

Description1 - Banir utilizador

Super Use Case	Moderar	
Author		
Date		
Brief Description	Permite banir um utilizador por um tempo especificado	
Preconditions	Utilizador existe; Moderador autenticado	
Post-conditions	Utilizador banido	
Flow of Events		Actor Input
	1	Selecciona utilizador
	2	
	3	
	4	Insere tempo de duração
	5	
	6	Confirma remoção
	7	
	8	
Alternative 6a - Não confirma remoção		Actor Input
	1	

Description1 - Consultar status utilizador

Super Use Case	Moderar	
Author		
Date		
Brief Description	Permite consultar os status do utilizador	
Preconditions	Utilizador existe; Moderador autenticado	
Post-conditions		
Flow of Events		Actor Input
		System Response
	1	Selecciona um utilizador
	2	Mostra status do utilizador

Description1 - Apagar notícia

Super Use Case	Moderar	
Author		
Date		
Brief Description	Permite a um Moderador apagar uma notícia	
Preconditions	Notícia existe; Moderador autenticado	
Post-conditions	Notícia apagada	
Flow of Events		Actor Input
	1	Selecciona notícia
	2	
	3	
	4	Confirma remoção
	5	
	6	
Alternative 6a - Não confirma remoção		System Response
	1	

Description1 - Marcar notícia

Super Use Case	Moderar	
Author		
Date		
Brief Description	Permite a um moderador marcar uma notícia	
Preconditions	Notícia existe; Moderador autenticado	
Post-conditions		
Flow of Events		Actor Input
	1	Selecciona a notícia
	2	
	3	Marca notícia
	4	
	5	Confirma marcação
	6	
Alternative 5a - Não confirma marcação		System Response
	1	
		Mostra dados da notícia
		Pede confirmação da marcação
		Regista marca informa sucesso
		System Response
		volta ao passo 2

Main - Comentar notícia

Super Use Case			
Author			
Date			
Brief Description	Jornalista comenta uma notícia publicada		
Preconditions	Jornalista autenticado; Tem de existir pelo menos uma notícia		
Post-conditions	Número de comentários à respectiva notícia + 1		
Flow of Events		Actor Input	System Response
	1		Apresenta a lista de notícias publicadas
	2	Selecciona a notícia que quer comentar	
	3		Selecciona a caixa de texto e escreve o seu comentário à respectiva notícia
	4	Selecciona a opção "Enviar comentário"	
	5		Confirma envio do comentário
	6		
	7		
Alternative		Actor Input	System Response
4a - Jornalista decide não enviar o comentário para o site	1	Selecciona a opção "Cancelar"	
	2		Confirma cancelamento do comentário
	3		Volta ao passo 1

Main - Consultar perfil

Super Use Case		
Author		
Date		
Brief Description	Jornalista consulta um perfil dentro do "NewZic"	
Preconditions	Jornalista autenticado	
Post-conditions		
Flow of Events		Actor Input
	1	System Response Apresenta o perfil ao Jornalista

Main - Editar perfil

Super Use Case			
Author			
Date			
Brief Description	Jornalista edita o seu perfil dentro do "NewZic"		
Preconditions	Jornalista autenticado, Perfil pessoal seleccionado		
Post-conditions	Jornalista edita o seu perfil com sucesso		
Flow of Events		Actor Input	System Response
	1		Apresenta o perfil ao jornalista
	2	Edita os campos que desejar nas respectivas caixas de texto	
	3	Selecciona a opção "Editar"	
	4		Confirma edição do perfil
Alternative 1		Actor Input	System Response
5 - Jornalista decide que não quer editar o seu perfil	1	Selecciona a opção "Cancelar"	
	2		Confirma cancelamento da edição do perfil
	3		Volta ao passo 1

Description1 - Encerrar sessão

Super Use Case		
Author		
Date		
Brief Description	Permite a um Jornalista, Moderador ou Administrador autenticado fazer logout	
Preconditions	Jornalista, Moderador ou Administrador autenticado	
Post-conditions	Jornalista, Moderador ou Administrador não autenticado	
Flow of Events		Actor Input
		System Response
	1	Pede logout
	2	Faz logout do utilizador
	3	Informa sucesso

Main - Iniciar sessão

Super Use Case		
Author		
Date		
Brief Description	Permite a um Jornalista, Moderador ou Administrador autenticar-se	
Preconditions	Utilizador não autenticado	
Post-conditions	Utilizador autenticado com um determinado papel	
Flow of Events		Actor Input
		System Response
	1	Insere email e password
	2	Autentica utilizador
Alternative 2a - autenticação falhada	3	Informa sucesso
		Actor Input
		System Response
	1	Informa insucesso
	2	Volta ao passo 1

Main - Procurar notícias

Super Use Case		
Author		
Date		
Brief Description	Permite ao utilizador procurar notícias	
Preconditions		
Post-conditions		
Flow of Events		Actor Input
	1	
	2	
	3	Selecciona tipo de pesquisa
	4	Insere texto a pesquisar
	5	
	6	
		System Response
		Mostra campo para inserção de texto
		Mostra lista de tipos de pesquisa
		Faz a pesquisa
		Mostra resultados

Main - Efectuar queixa

Super Use Case		
Author		
Date		
Brief Description	Permite ao Jornalista fazer queixa de um artigo ou notícia	
Preconditions	Utilizador autenticado com notícia seleccionada	
Post-conditions		
Flow of Events		Actor Input
	1	Reporta Notícia
	2	
Aletrnative 2a - Utilizador já apresentou queixa desta notícia		System Response
		Regista Queixa
	1	
		Actor Input
	1	
		System Response
	1	Ignora queixa

Main - Consultar perfil

Super Use Case		
Author		
Date		
Brief Description	Jornalista consulta um perfil dentro do "NewZic"	
Preconditions	Jornalista autenticado	
Post-conditions		
Flow of Events		Actor Input
	1	System Response Apresenta o perfil ao Jornalista

Main - Editar perfil

Super Use Case		
Author		
Date		
Brief Description	Jornalista edita o seu perfil dentro do "NewZic"	
Preconditions	Jornalista autenticado, Perfil pessoal seleccionado	
Post-conditions	Jornalista edita o seu perfil com sucesso	
Flow of Events		Actor Input
	1	
	2	Edita os campos que desejar nas respectivas caixas de texto
	3	Selecciona a opção "Editar"
	4	
		System Response
		Apresenta o perfil ao jornalista
		Confirma edição do perfil
Alternative 1		Actor Input
5 - Jornalista decide que não quer editar o seu perfil	1	Selecciona a opção "Cancelar"
	2	
	3	
		System Response
		Confirma cancelamento da edição do perfil
		Volta ao passo 1

Consultar Noticia - Consultar notícia

Super Use Case	Gerir notícias	
Author		
Date		
Brief Description	Descreve o comportamento da interacção do Actor com o Sistema durante o processo de consulta de um notícia	
Preconditions	Notícia seleccionada pelo Actor existe na base de dados.	
Post-conditions	A notícia seleccionada é apresentada ao Actor.	
Flow of Events		Actor Input
	1	Selecciona notícia
	2	
	3	
	4	
	5	
Alternative 3a - Actor não autenticado		System Response
		Pede notícia seleccionada à base de dados
		Verifica que o Actor é utilizador autenticado
		Pede comentários da notícia seleccionada à base de dados
		Apresenta notícia e comentários ao Actor
		Actor Input
	1	
	2	
		System Response
		Verifica que o Actor não é utilizador autenticado
		Apresenta notícia ao Actor

Main - Remover notícia

Super Use Case	Gerir notícias		
Author			
Date			
Brief Description	Descreve o comportamento da interacção do Jornalista com o Sistema quando o Jornalista pretende remover uma notícia da sua autoria.		
Preconditions	O Actor está autenticado como Jornalista; A notícia seleccionada existe e é da autoria do Actor;		
Post-conditions	A notícia seleccionada desaparece do Newzic		
Flow of Events		Actor Input	System Response
	1	Selecciona notícia	
	2		Pede confirmação
	3	Confirma remoção	
	4		Marca notícia seleccionada como removida na base de dados
	5		Apresenta mensagem de sucesso.
Alternative 3a - Cancela Remoção		Actor Input	System Response
	1	Não confirma remoção	
	2		Volta ao passo 1

Main - Publicar notícia

Super Use Case	Gerir notícias	
Author		
Date		
Brief Description	Descreve o comportamento da interação entre um Jornalista e o Sistema quando o Jornalista pretende publicar uma notícia.	
Preconditions	O Actor está autenticado como Jornalista;	
Post-conditions	A notícia publicada existe no Newzic	
Flow of Events		System Response
	1 Preenche dados da notícia	
	2	<<extends>> Gerir Imagens
	3	<<extends>> Gerir Mapa
	4	<<extends>> Gerir Vídeos
	5 Adiciona Tags a notícia	
	6 Pré-visualiza notícia	
	7	Apresenta a notícia tal como irá ficar após publicada
	8 Publica notícia	
	9	Apresenta mensagem de sucesso
Alternative		System Response
6a - Não faz	1 Não faz pré-visualização	
Pré-Visualização	2	salta para o passo 8

Main - Editar notícia

Super Use Case	Gerir notícias		
Author			
Date			
Brief Description	Descreve o comportamento da interacção do Jornalista com o Sistema quando o Jornalista pretende editar uma notícia da sua autoria		
Preconditions	Actor autenticado como Jornalista; notícia seleccionada e é da autoria do Jornalista;		
Post-conditions	A notícia seleccionada foi alterada		
Flow of Events		Actor Input	System Response
	1		Carrega os dados da notícia a partir da base de dados
	2		Apresenta formulário com os campos preenchidos
	3		<<extends>> Gerir Mapa
	4		<<extends>> Gerir Imagens
	5		<<extends>> Gerir Vídeos
	6	Efectua alterações nos campos da notícia	
	7	Pré-visualiza a notícia	
	8		Apresenta a notícia tal como irá ficar após publicada
	9	Publica notícia	
	10		Apresenta mensagem de sucesso
Alternative		Actor Input	System Response
7a - Não faz pré-visualização	1	Não faz pré-visualização	
	2		Salta para o passo 9

Main - Adicionar Ponto

Super Use Case	Gerir Mapa		
Author			
Date			
Brief Description	Descreve o comportamento da interação do Jornalista com o Sistema quando o Jornalista pretende adicionar um ponto no mapa		
Preconditions	O mapa existe e está associado a uma notícia		
Post-conditions	O mapa passa a ter mais um ponto		
Flow of Events		Actor Input	System Response
	1		Apresenta mapa
	2	Selecciona localização	
	3		Guarda coodenadas da selecção
	4		Pede título para o ponto
	5	Insere título	
	6		pede confirmação
	7	confirma	
	8		Associa mapa à notícia
	9		Apresenta mapa com o novo ponto
Alternative		Actor Input	System Response
7a) Não confirma	1		volta ao passo 1

Main - Remover Ponto

Super Use Case	Gerir Mapa	
Author		
Date		
Brief Description	Descreve o comportamento da interacção do Jornalista com o Sistema quando o Jornalista pretende remover um ponto do mapa	
Preconditions	Mapa seleccionado	
Post-conditions	O mapa seleccionado passa a ter menos um ponto	
Flow of Events		Actor Input
	1	Selecciona ponto no mapa
	2	
	3	Confirma remoção
	4	
	5	
	6	
		System Response
		Pede confirmação
		Remove ponto do mapa
		Marca ponto como removido
		Apresenta mensagem de sucesso
Alternative		Actor Input
3a - Não	1	Não confirma remoção
confirma remoção	2	
		System Response
		volta ao passo 1

Main - Remover Mapa

Super Use Case	Gerir Mapa		
Author			
Date			
Brief Description	Descreve o comportamento da interacção do Jornalista com o Sistema quando o Jornalista pretende remover um mapa associado a uma notícia		
Preconditions	Mapa seleccionado		
Post-conditions	O mapa seleccionado é removido da notícia		
Flow of Events		Actor Input	System Response
	1		Pede Confirmação
	2	Confirma Remoção	
	3		Marca pontos do mapa como removidos
	4		Remove mapa da notícia
	5		Apresenta mensagem de sucesso
Exception 2a - Não confirma remoção		Actor Input	System Response
	1	Não confirma remoção	
	2		Apresenta mensagem de remoção cancelada

Main - Adicionar Tour

Super Use Case	Gerir Mapa		
Author			
Date			
Brief Description	Descreve o comportamento da interacção do Jornalista com o Sistema quando o Jornalista pretende adicionar um conjunto de pontos para representar uma Tour		
Preconditions	O mapa existe e está associado a uma notícia		
Post-conditions	O mapa passa a ter mais uma tour		
Flow of Events		Actor Input	System Response
	1		Pede ano e banda
	2	Insere ano e banda	
	3		Apresenta mapa
	4	Selecciona um conjunto de localizações	
	5		Guarda coordenadas da selecção
	6		Pede título para cada ponto
	7	Insere títulos	
	8		pede confirmação
	9	confirma	
	10		Associa mapa a notícia
	11		Apresenta mapa com a nova tour
Alternative		Actor Input	System Response
9a) Nao confirma	1		Volta ao passo 1

Main - Upload de uma Imagem

Super Use Case	Gerir Imagens	
Author		
Date		
Brief Description	Descreve o comportamento da interacção do Jornalista com o Sistema quando o Jornalista pretende fazer upload de uma imagem para a sua conta	
Preconditions	O limite máximo de imagens armazenadas na conta do Actor não foi excedido	
Post-conditions	Existe mais uma imagem na conta do Actor	
Flow of Events		Actor Input
	1	
	2	Selecciona imagem para enviar
	3	
	4	
	5	Efectua upload
	6	
Alternative 4a - Não existe espaço disponível para armazenar a imagem		System Response
	1	
	2	
	3	

Main - Remover Imagem da notícia

Super Use Case	Gerir Imagens		
Author			
Date			
Brief Description	Descreve o comportamento da interacção do Jornalista com o Sistema quando o Jornalista pretende remover uma imagem que esta associada a uma notícia		
Preconditions	Imagem seleccionada		
Post-conditions	A Lista de imagens da notícia passa a ter menos uma imagem		
Flow of Events		Actor Input	System Response
	1		Pede confirmação
	2	Confirma remoção	
	3		Remove a imagem seleccionada da lista de imagens da notícia
	4		Apresenta mensagem de sucesso
Exception 4a - Não confirma remoção		Actor Input	System Response
	1	Não confirma remoção	
	2		Apresenta mensagem de remoção cancelada

Main - Adicionar Video a notícia

Super Use Case	Gerir Videos		
Author			
Date			
Brief Description	Descreve o comportamento da interacção do Jornalista com o Sistema quando o Jornalista pretende associar um vídeo a uma notícia		
Preconditions			
Post-conditions	A noticia passa a conter mais um vídeo associado		
Flow of Events		Actor Input	System Response
	1		Pede URL do vídeo
	2	Insere URL do vídeo	
	3		Associa o URL à notícia
	4		Apresenta mensagem de Sucesso

Main - Remover vídeo da notícia

Super Use Case	Gerir Vídeos	
Author		
Date		
Brief Description	Descreve o comportamento da interacção do Jornalista com o Sistema quando o Jornalista pretende remover um vídeo associado a uma notícia	
Preconditions	Vídeo seleccionado	
Post-conditions	A notícia passa a ter menos um vídeo associado	
Flow of Events		Actor Input
	1	
	2	Confirma remoção
	3	
	4	
Exception 2a - Não confirma remoção		System Response
	1	Não confirma remoção
	2	
		Apresenta mensagem de remoção cancelada

Main - Ler Queixa

Super Use Case	Moderar	
Author		
Date		
Brief Description	Permite ao Moderador ler queixas de Jornalistas	
Preconditions	Utilizador autenticado como Moderador	
Post-conditions	Sistema fica com menos uma queixa por ler	
Flow of Events		Actor Input
	1	
	2	Selecciona uma queixa
	3	
	4	
Exception 1a - Nao há queixas		System Response
		Mostra lista de queixas
		Mostra a queixa
		Marca queixa como lida
		Actor Input
	1	
		System Response
		Mostra lista vazia

Main - Apagar queixa

Super Use Case	Moderar	
Author		
Date		
Brief Description	Permite ao Moderador apagar uma queixa	
Preconditions	Utilizador autenticado como Moderador	
Post-conditions	O Sistema fica com menos uma queixa	
Flow of Events		Actor Input
	1	
	2	Selecciona queixa
	3	
	4	Confirma selecção
	5	
		System Response
		Mostra lista de queixas
		Pede confirmação
		Apaga queixa seleccionada
Alternative		Actor Input
4a - Cancela	1	
Seleccção		
		System Response
		Volta ao passo 1
Exception		Actor Input
1a - Nao há	1	
queixas no		
sistema		System Response
		Mostra lista vazia

Main - Votar numa notícia

Super Use Case		
Author		
Date		
Brief Description	Permite ao jornalista votar positivamente ou negativamente numa notícia	
Preconditions	Utilizador autenticado com notícia seleccionada	
Post-conditions	Número de votos à respectiva notícia + 1	
Flow of Events		Actor Input
	1	efectua votação
	2	Sistema actualiza o número de votos

Main - Consultar FAQ

Super Use Case		
Author		
Date		
Brief Description	Permite ao utilizador consultar ajuda sobre o funcionamento do sistema	
Preconditions		
Post-conditions		
Flow of Events		Actor Input
	1	System Response Mostra tópicos de ajuda

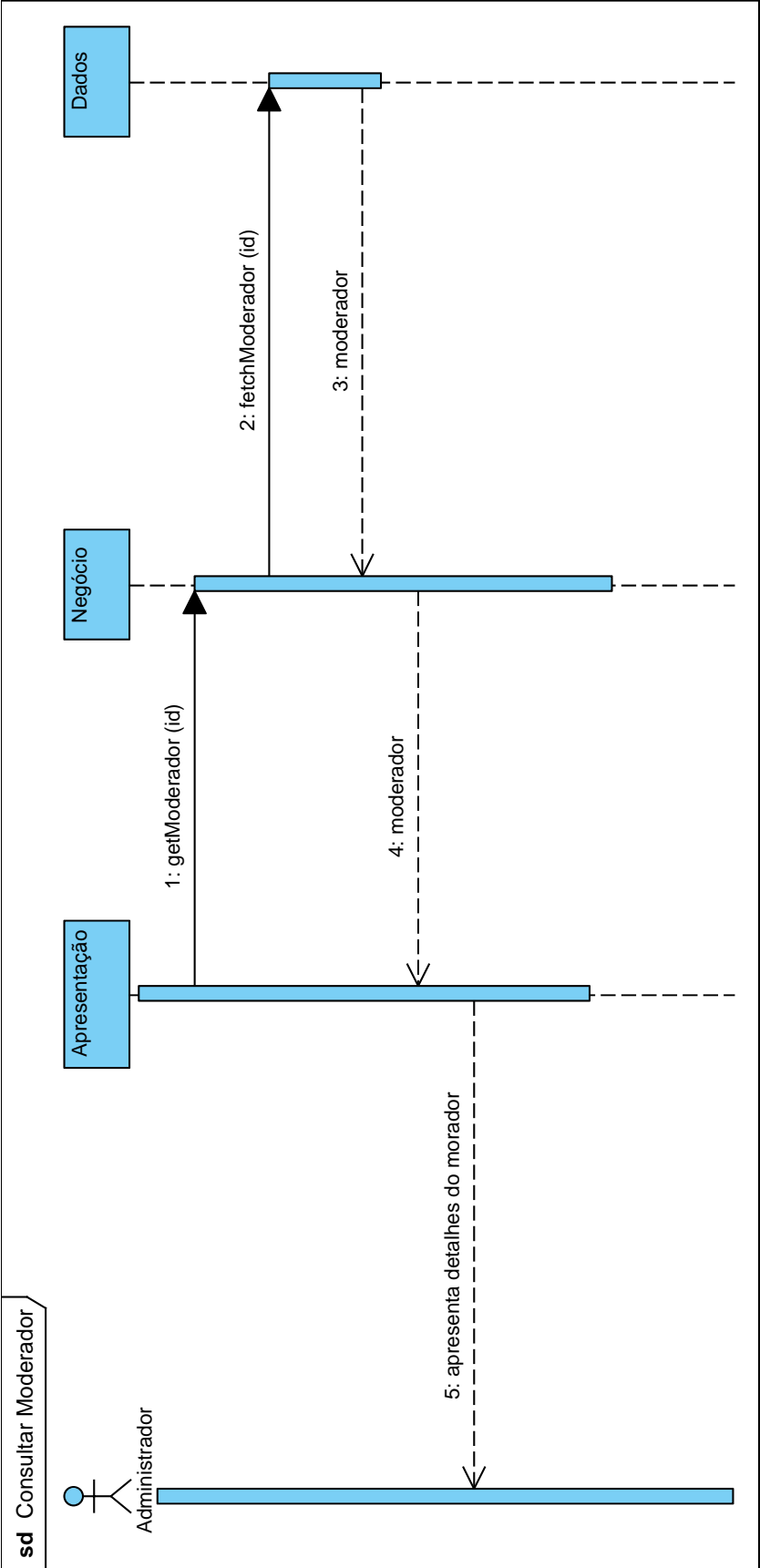
Main - Fazer queixa de utilizador

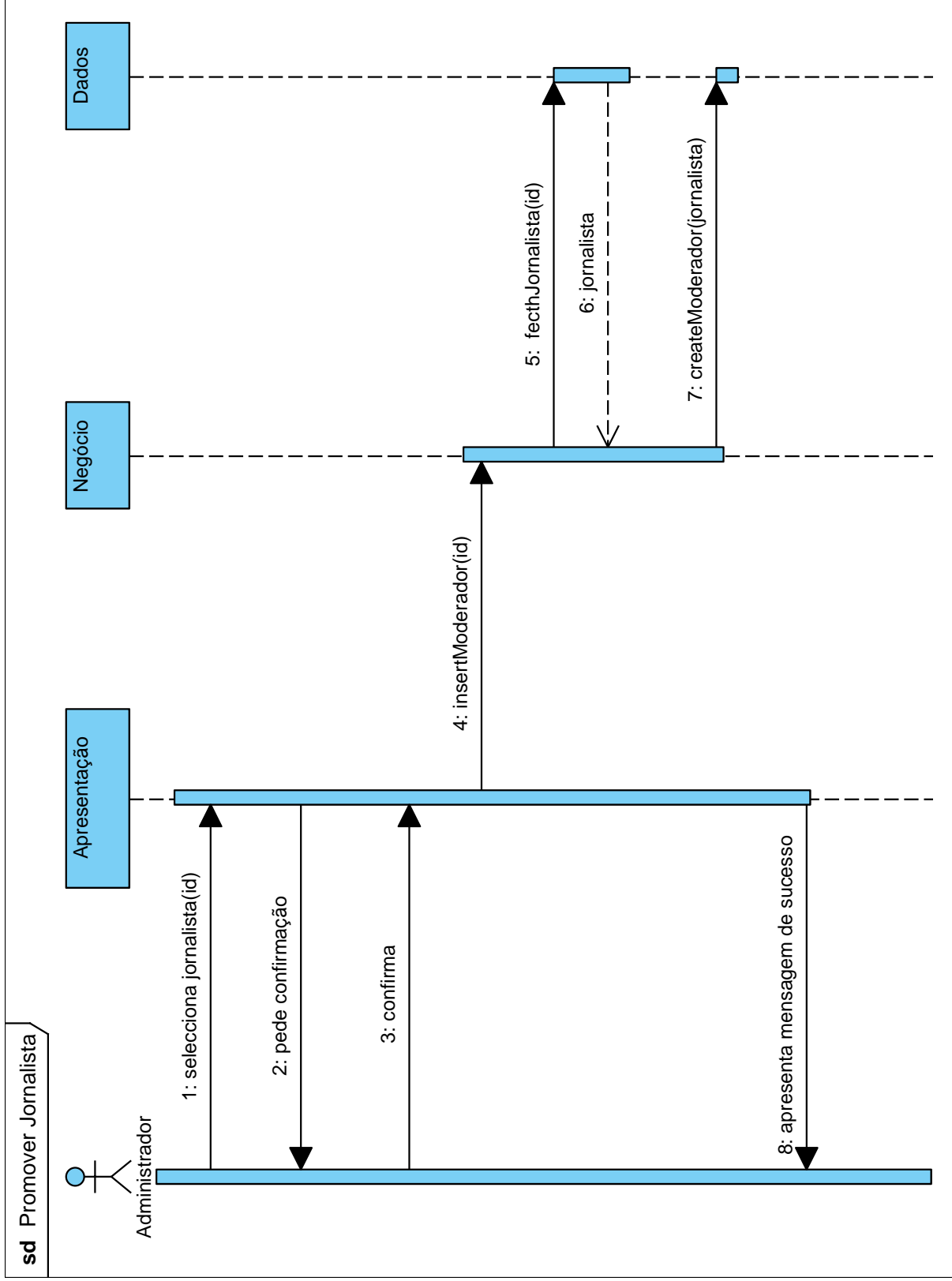
Super Use Case		
Author		
Date		
Brief Description	Fazer queixa de um utilizador	
Preconditions	Utilizador Autenticado como Jornalista ou Moderador	
Post-conditions	Queixa guardada no sistema	
Flow of Events		Actor Input
	1	
	2	Insere os dados pedidos e clica em enviar
	3	
Alternative 2a - Cancelar queixa		System Response
		Mostra o formulário de preenchimento
		Sistema recebe queixa e guarda
		Actor Input
	1	
		System Response
		Volta ao passo 1

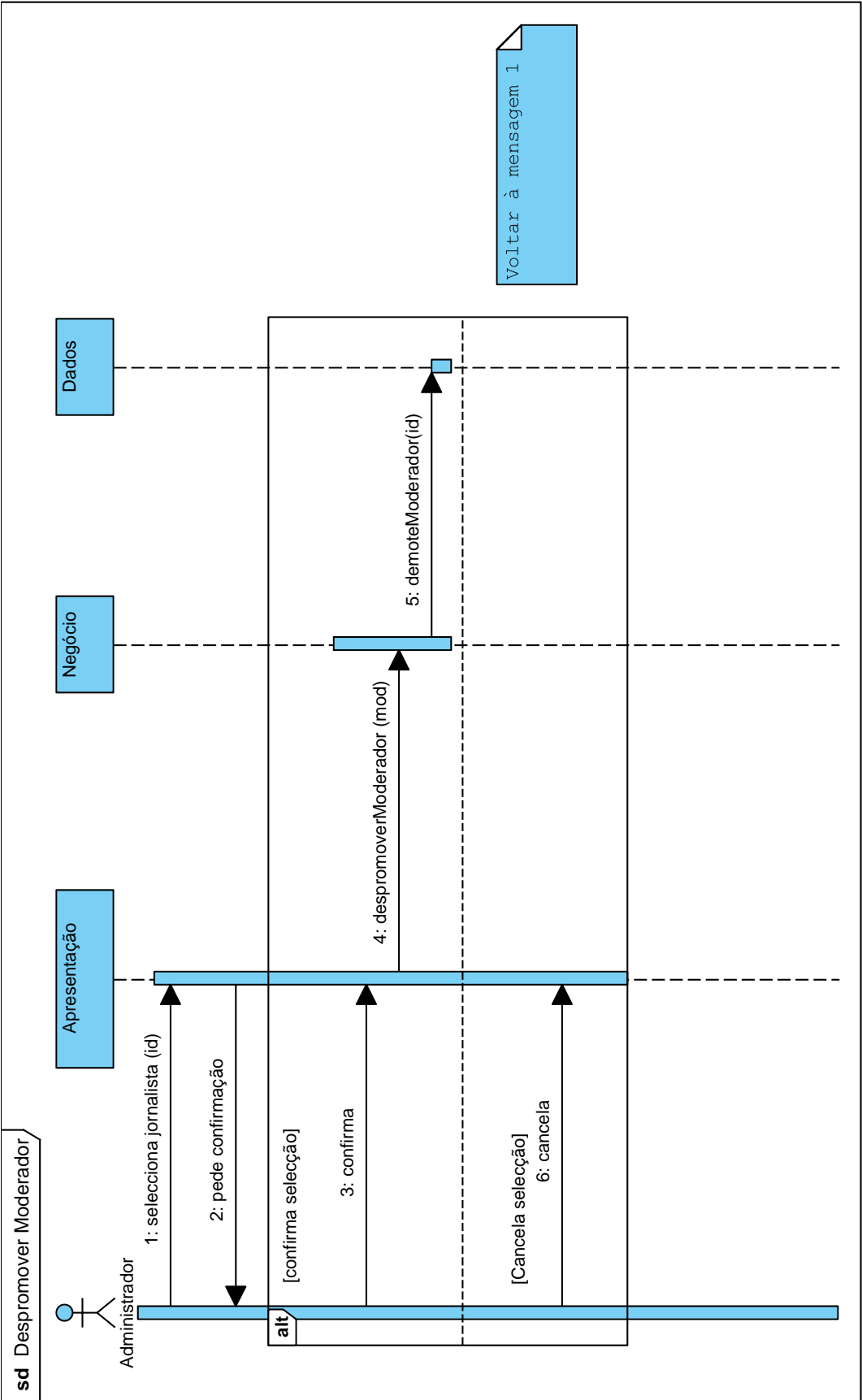
Main - Registar utilizador

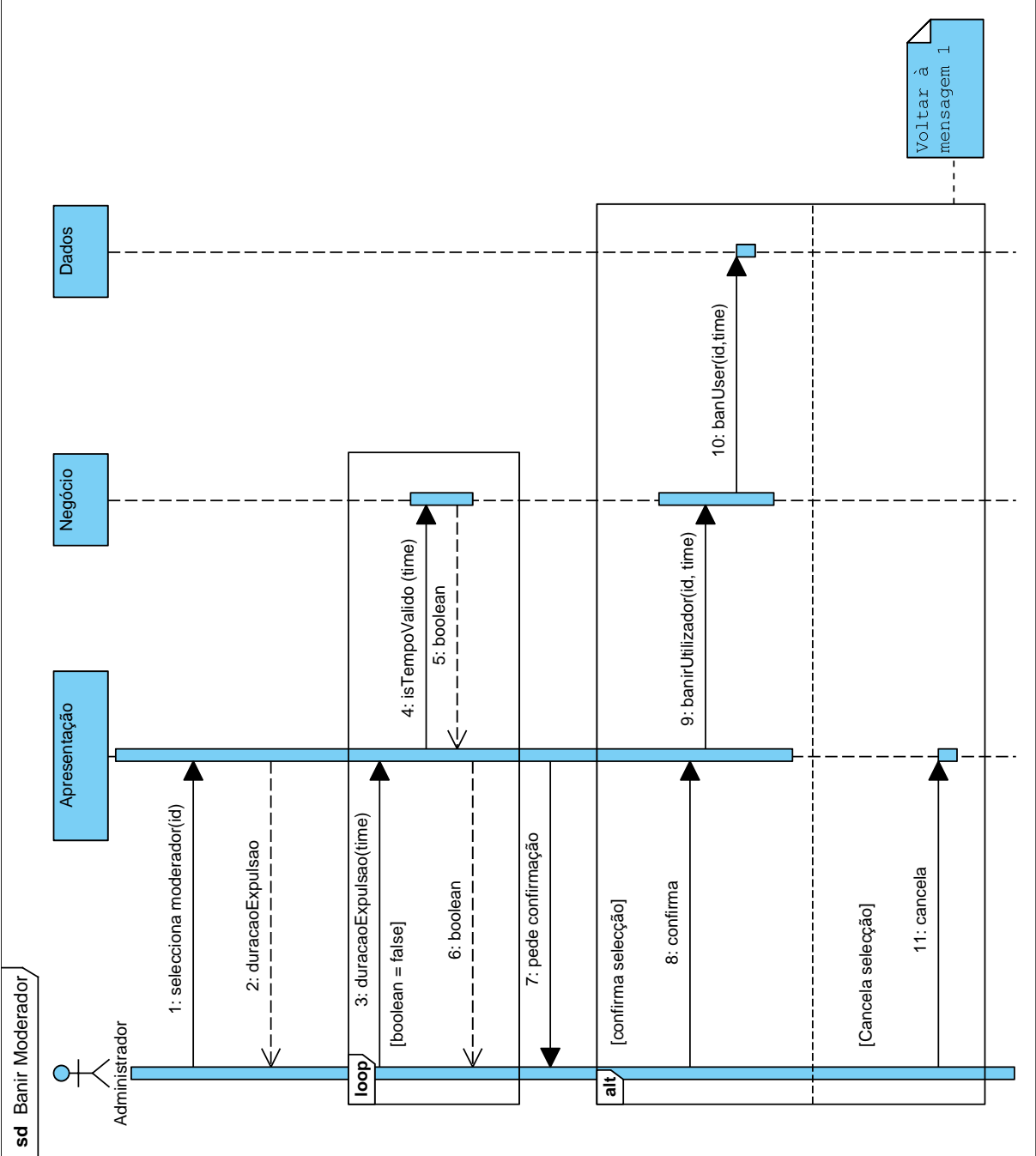
Super Use Case			
Author			
Date			
Brief Description	Permite a um utilizador efectuar o registo no sistema		
Preconditions			
Post-conditions	Existe mais um utilizador registado		
Flow of Events		Actor Input	System Response
	1		Apresenta formulário
	2	Preenche formulário	
	3		Verifica dados
	4		Pede confirmação
	5	Confirma	
	6		registra utilizador
	7		apresenta mensagem de sucesso
Alternative		Actor Input	System Response
5a) Não confirma	1		Volta ao passo 1
Alternative		Actor Input	System Response
3a) Dados inválidos	1		Verifica dados inválidos
	2		Apresenta mensagem de dados inválidos
	3		Volta ao passo 1

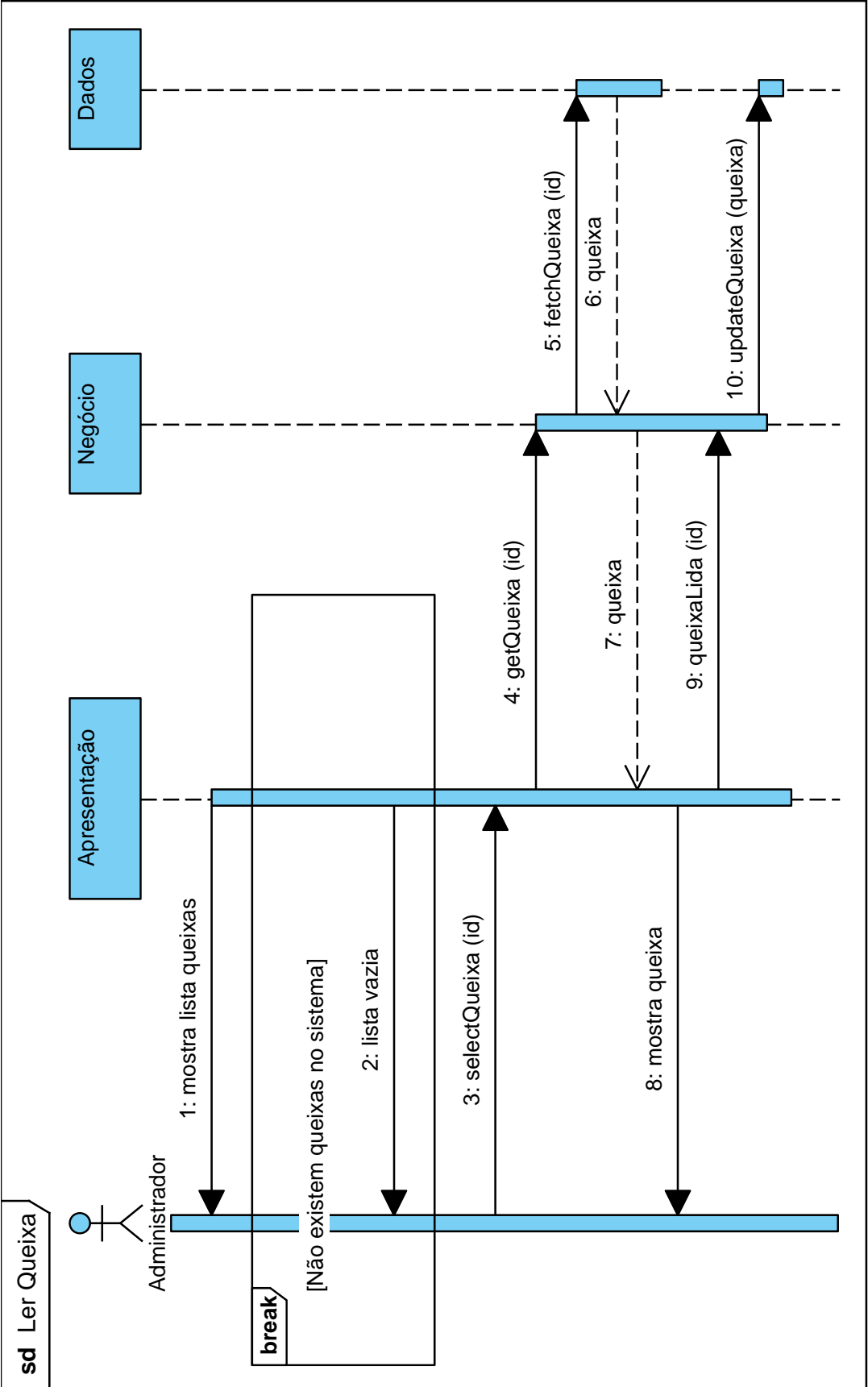
VII. Diagramas de Sequência

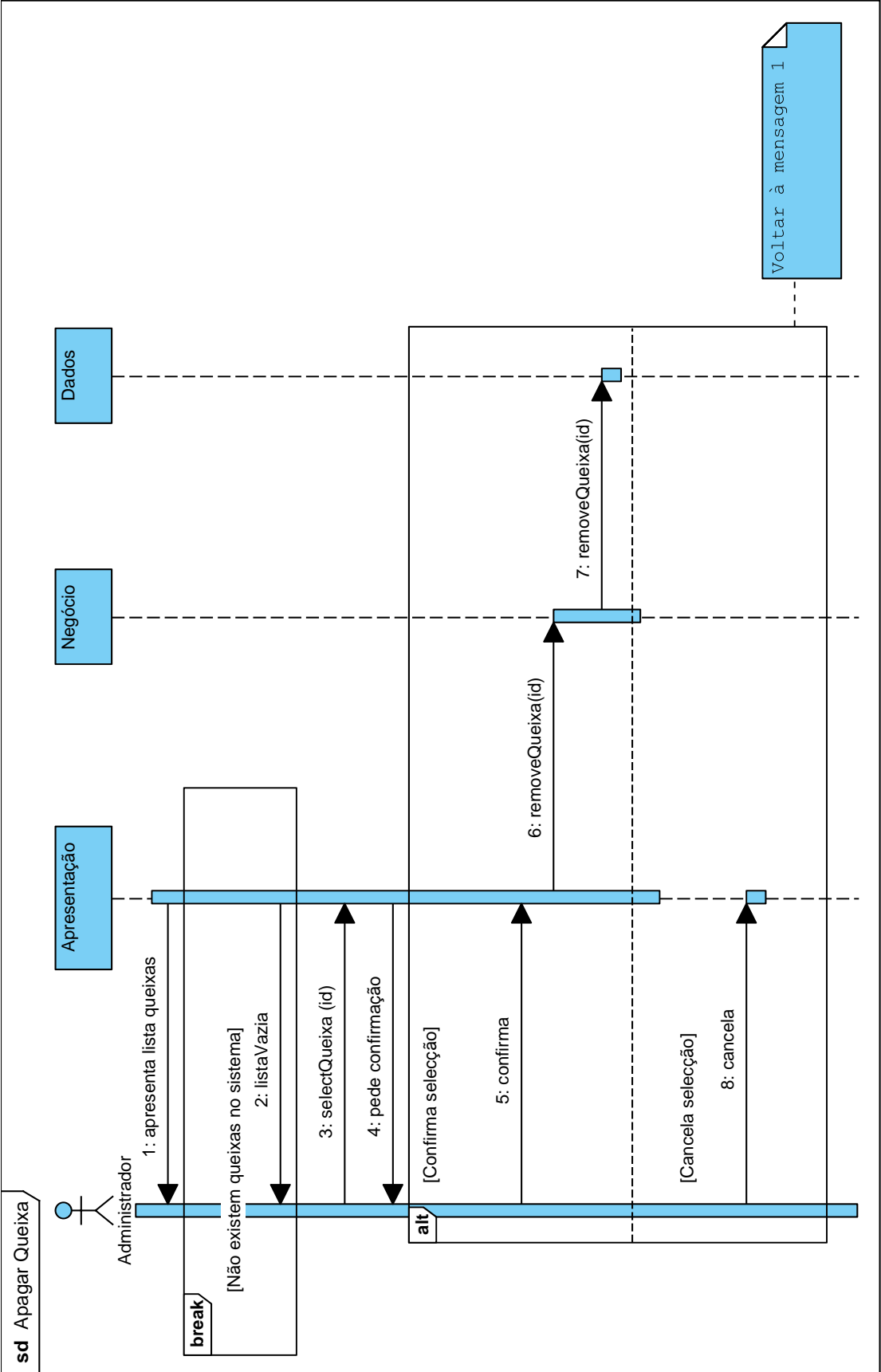


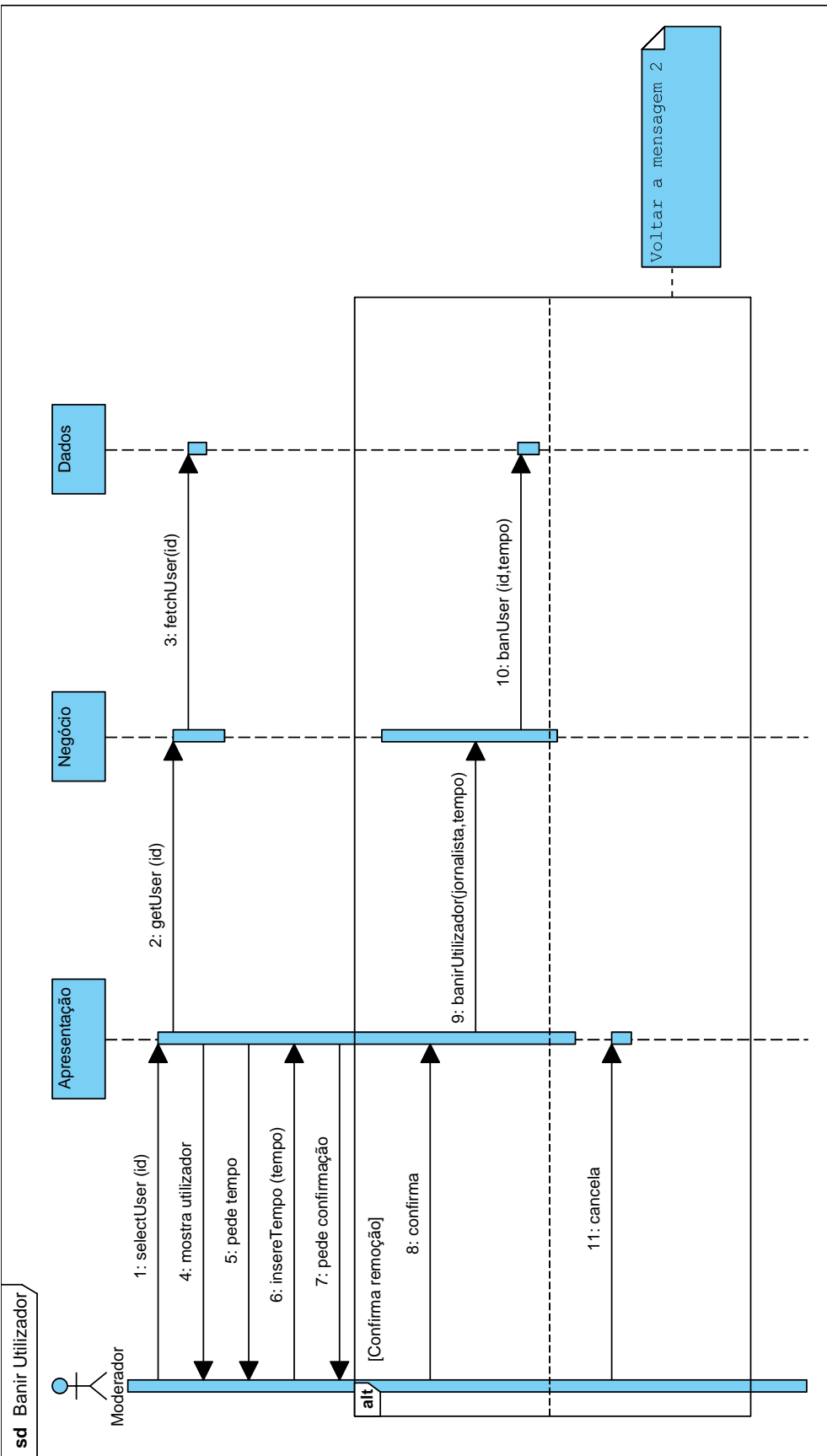


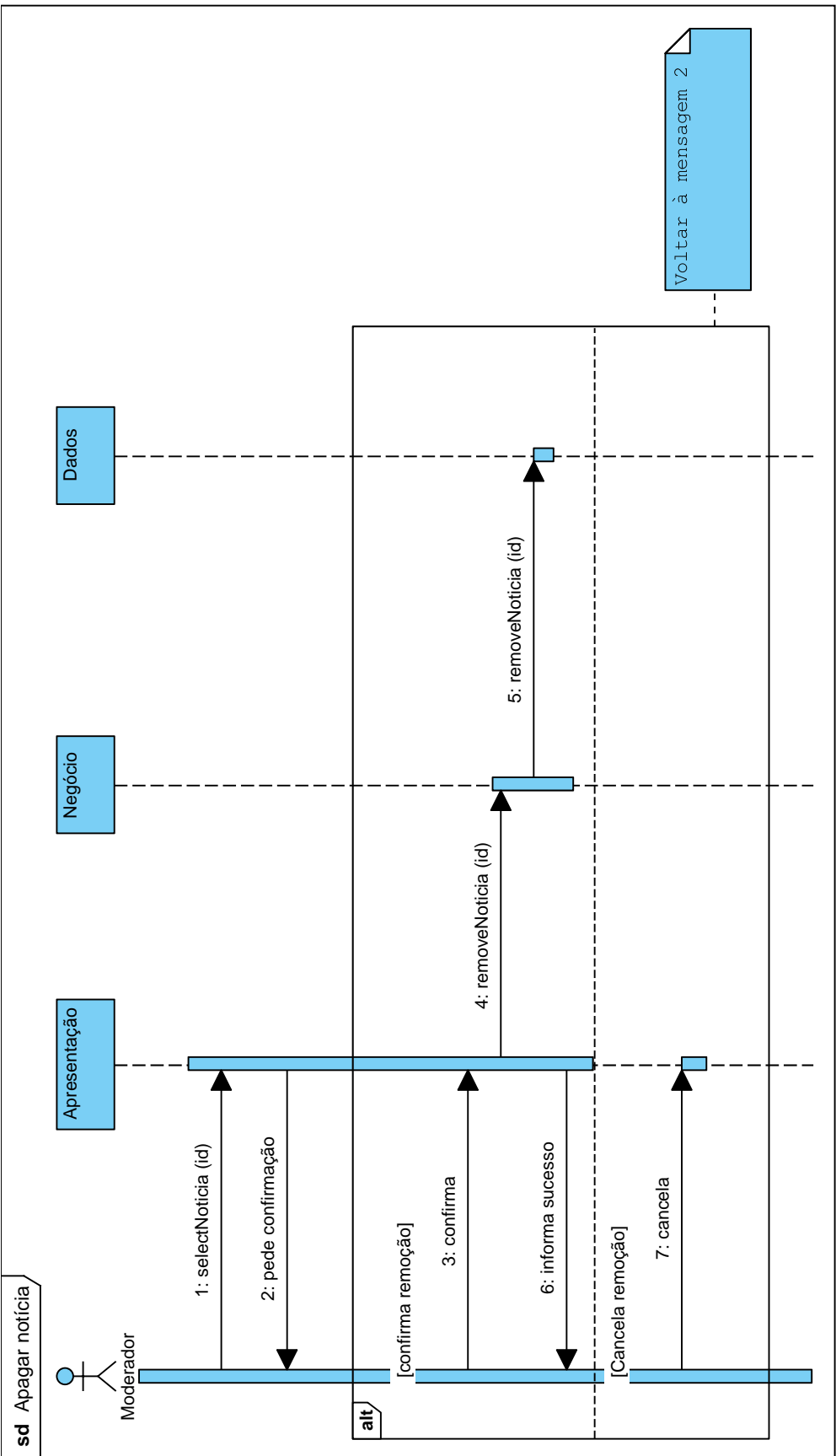


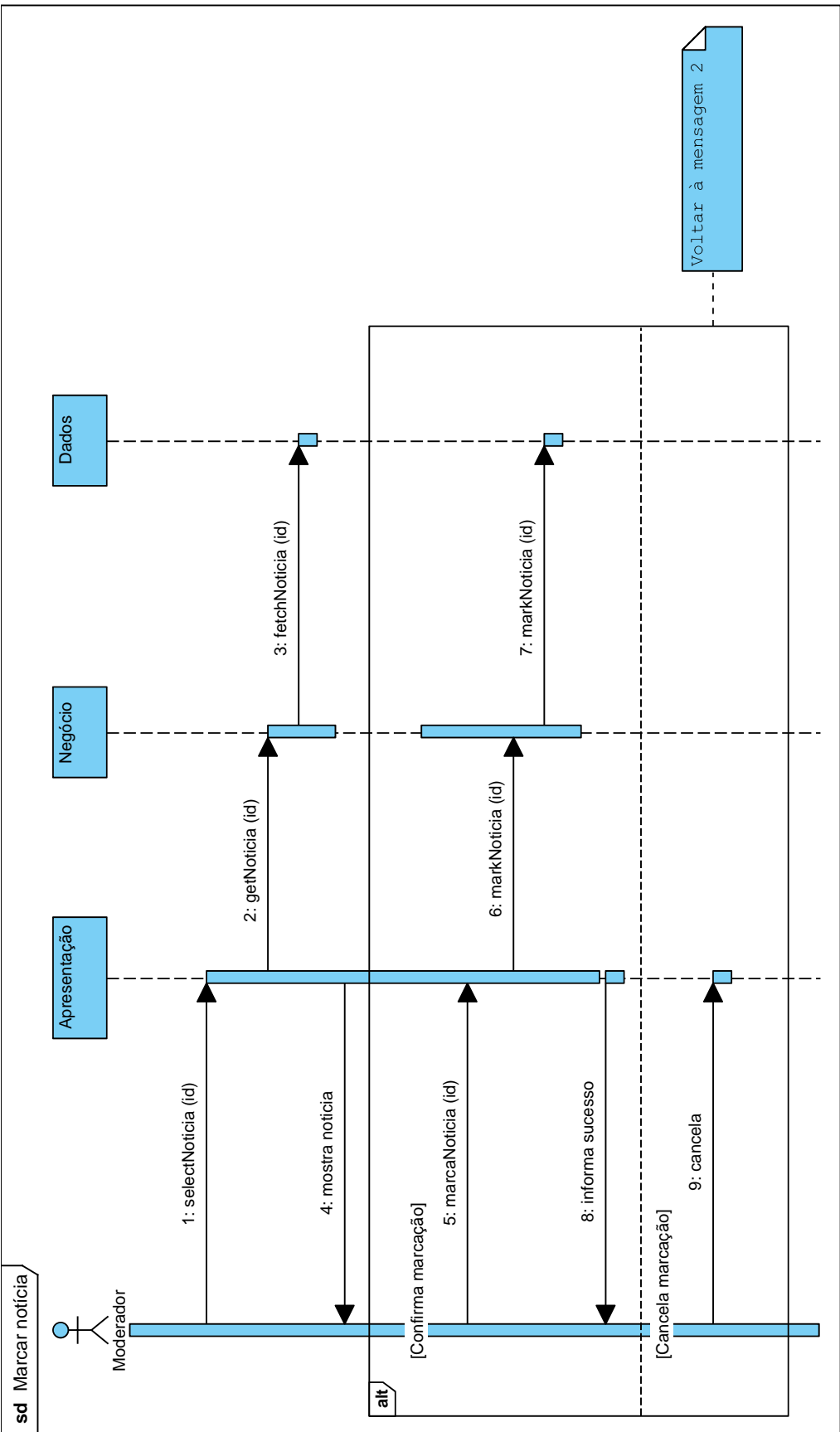


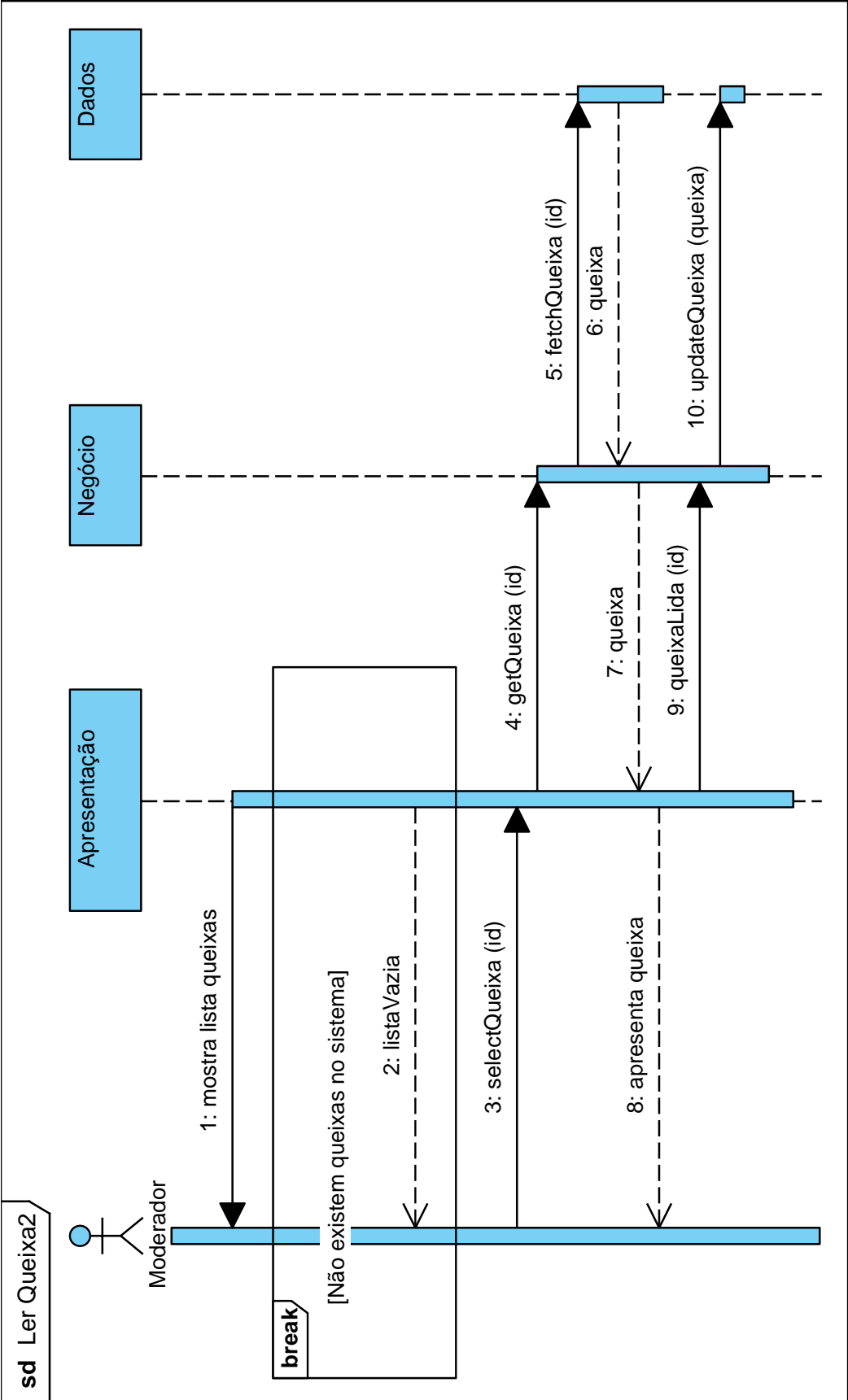


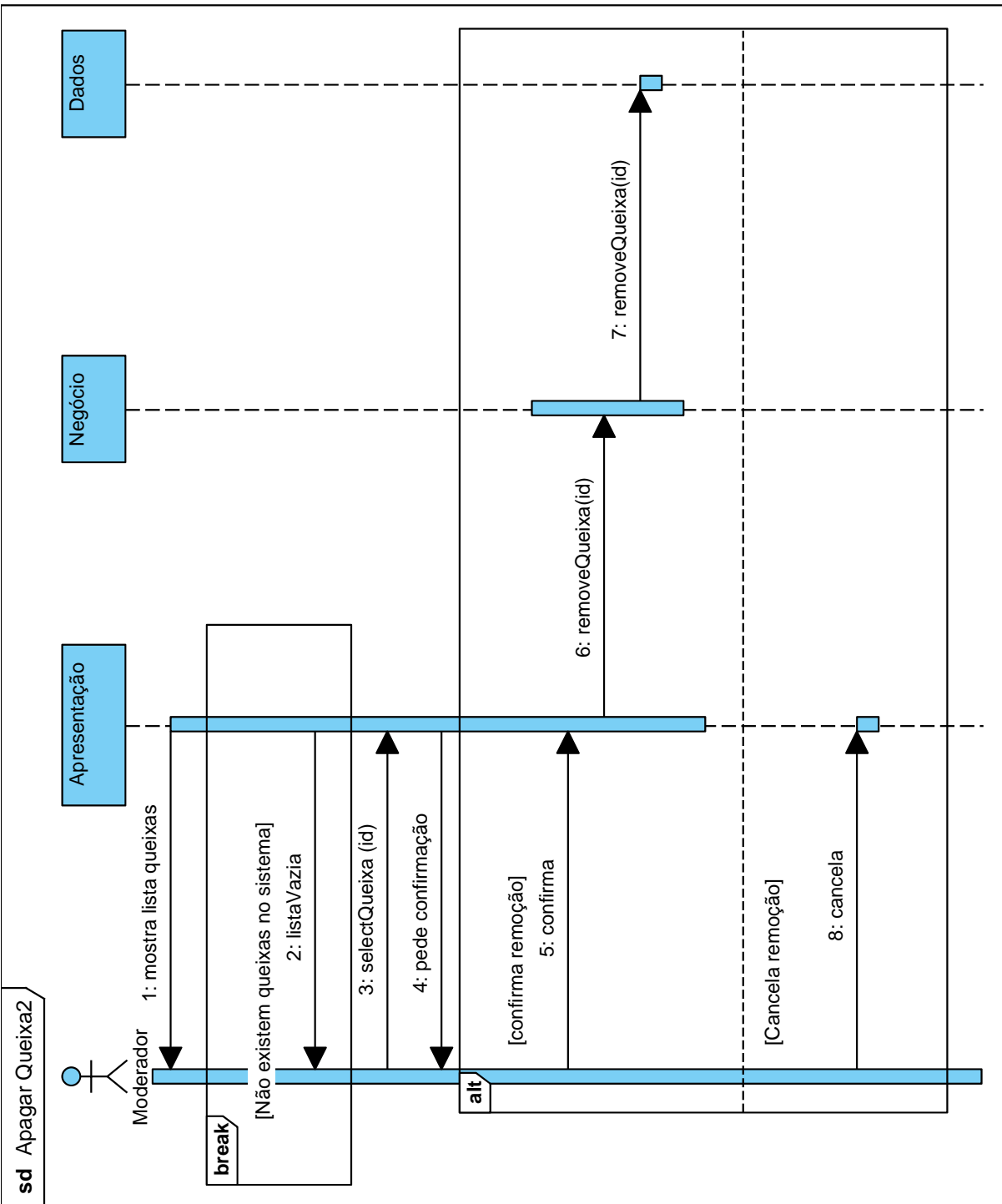


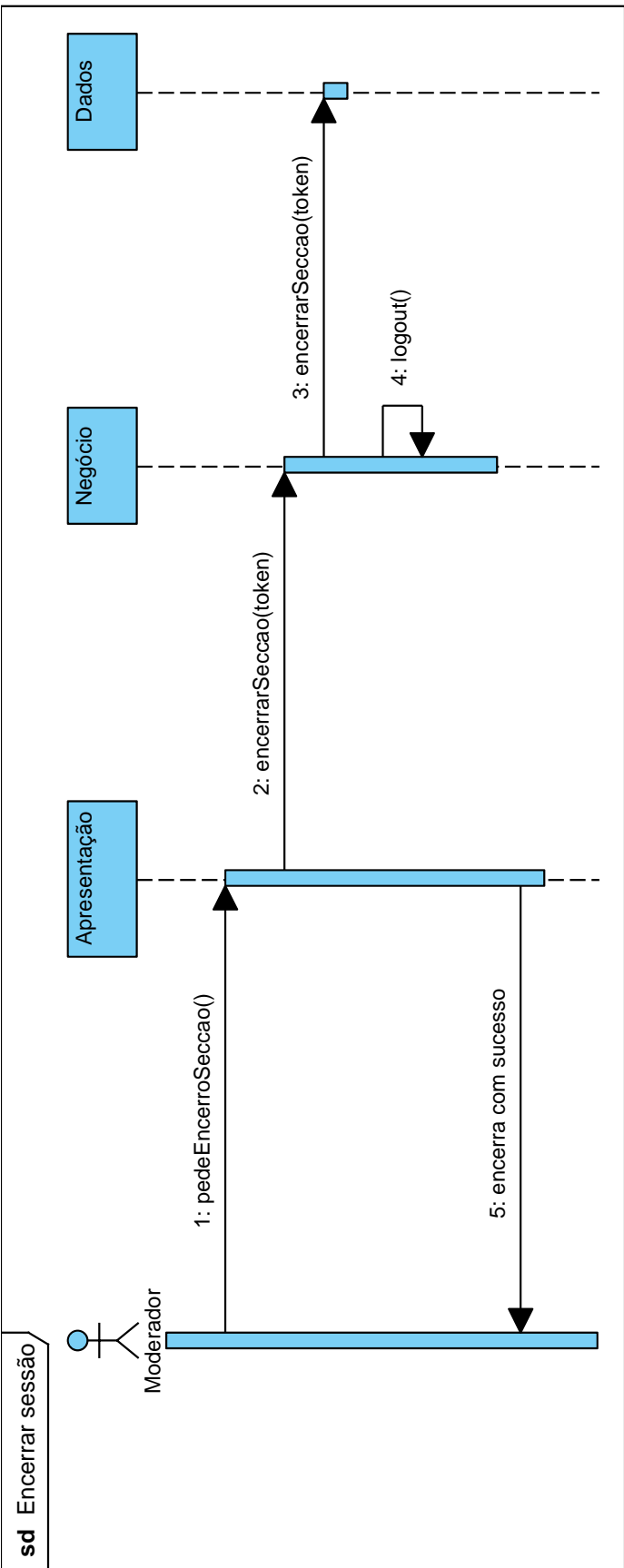


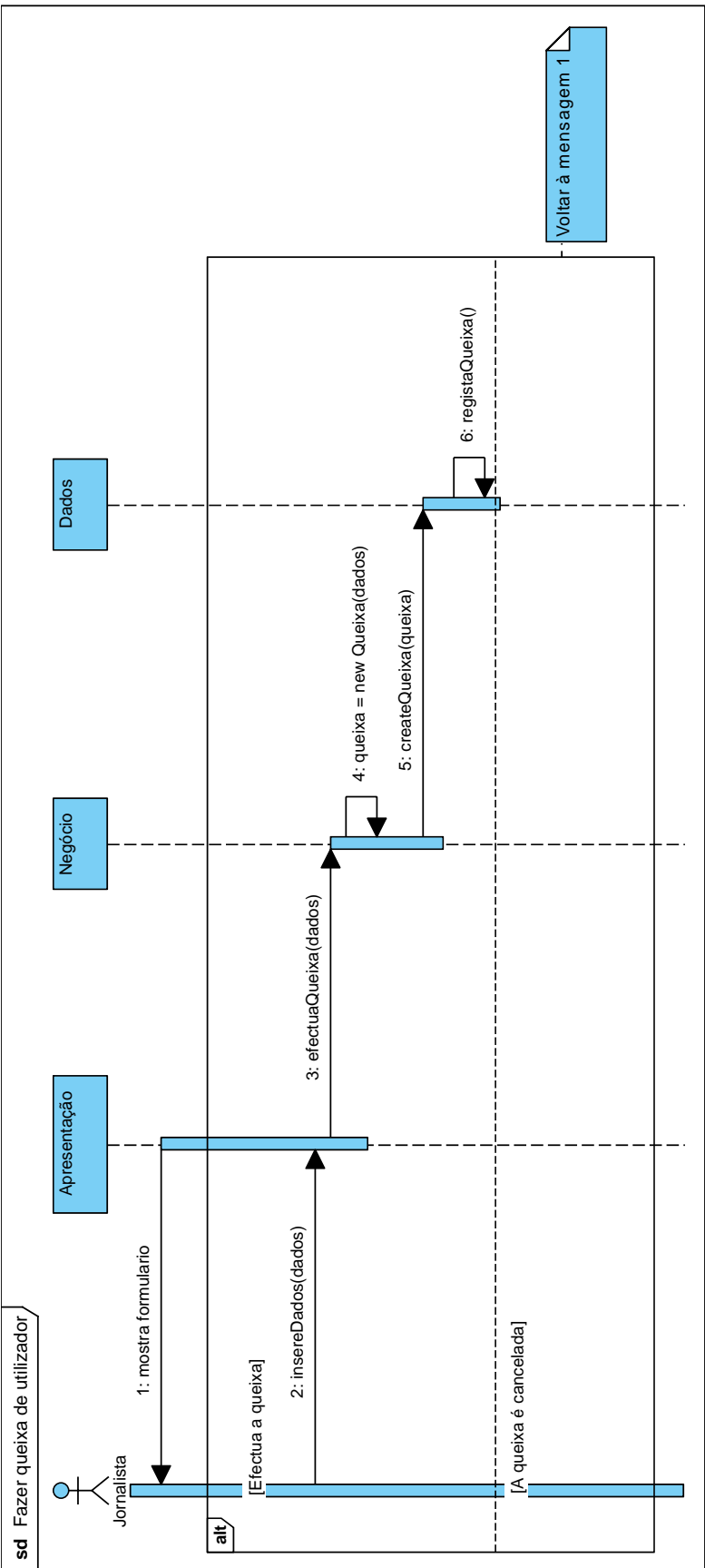


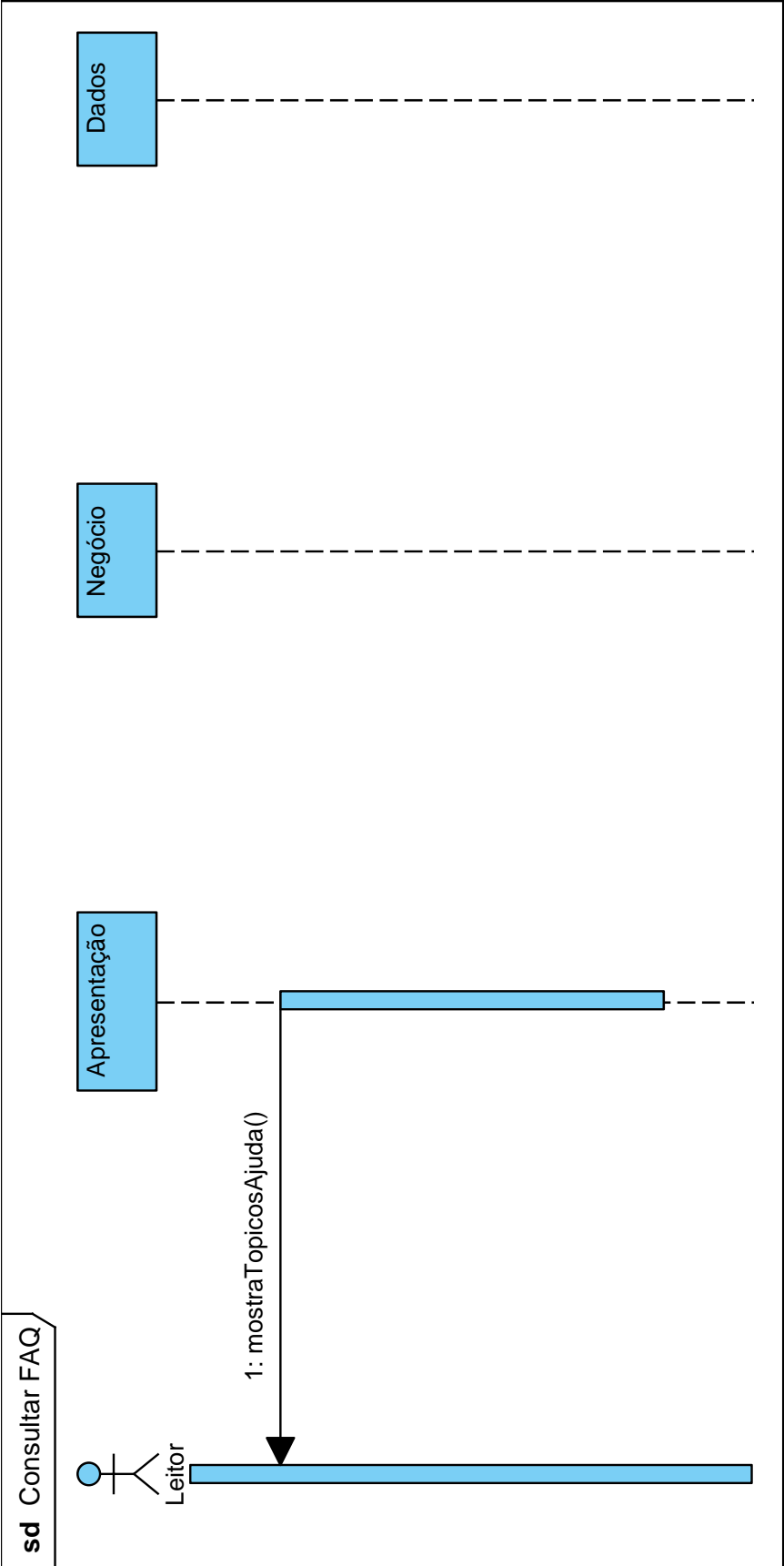


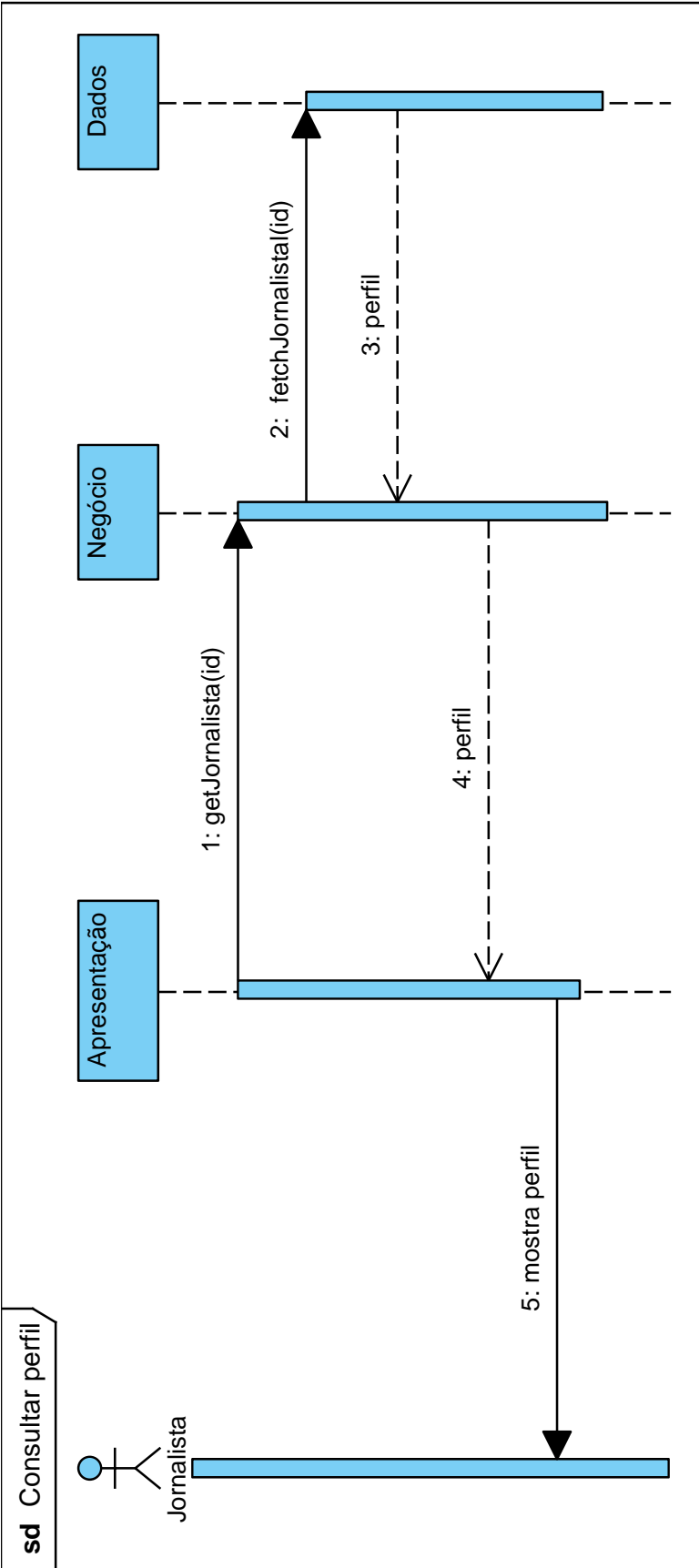


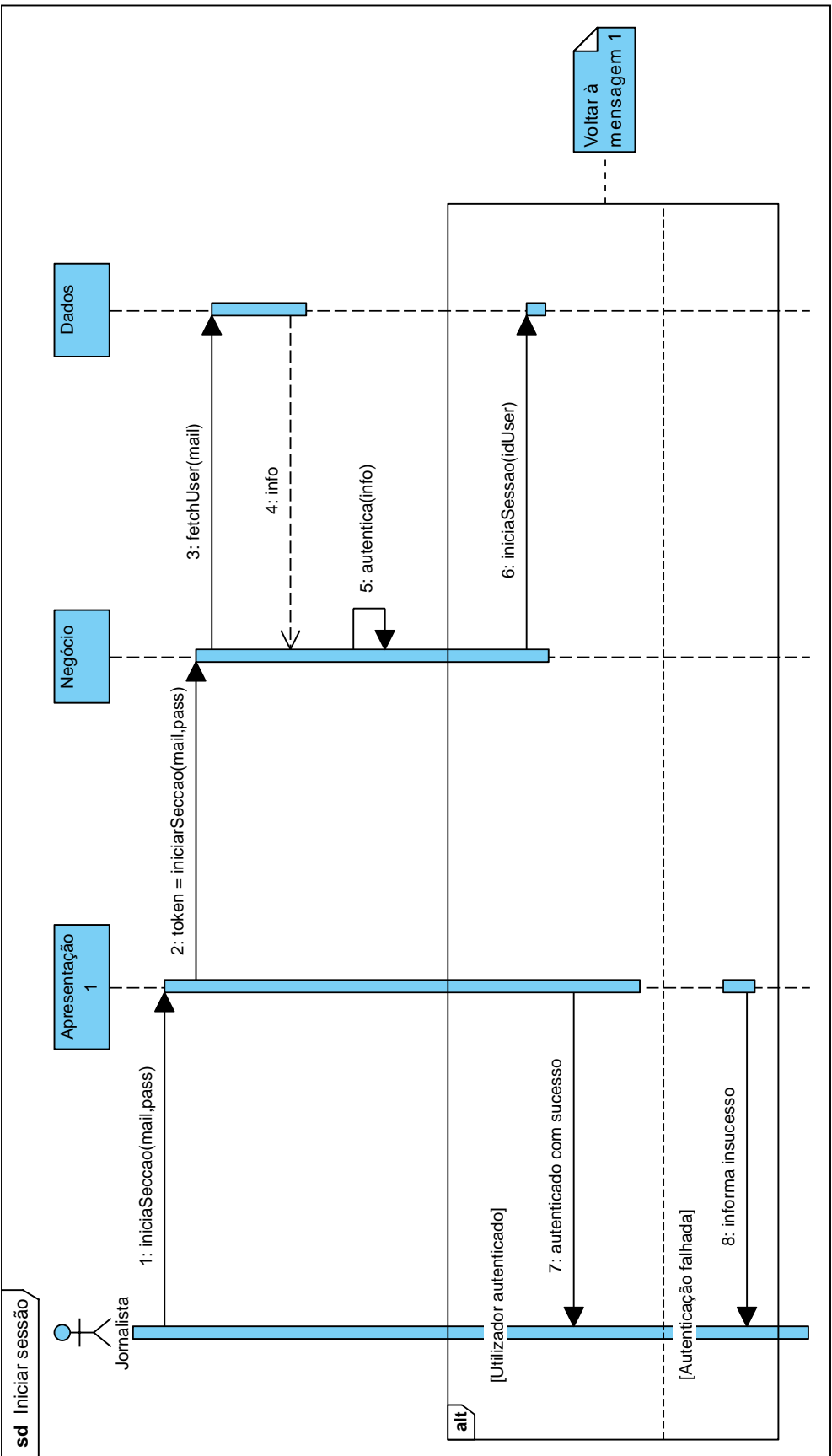


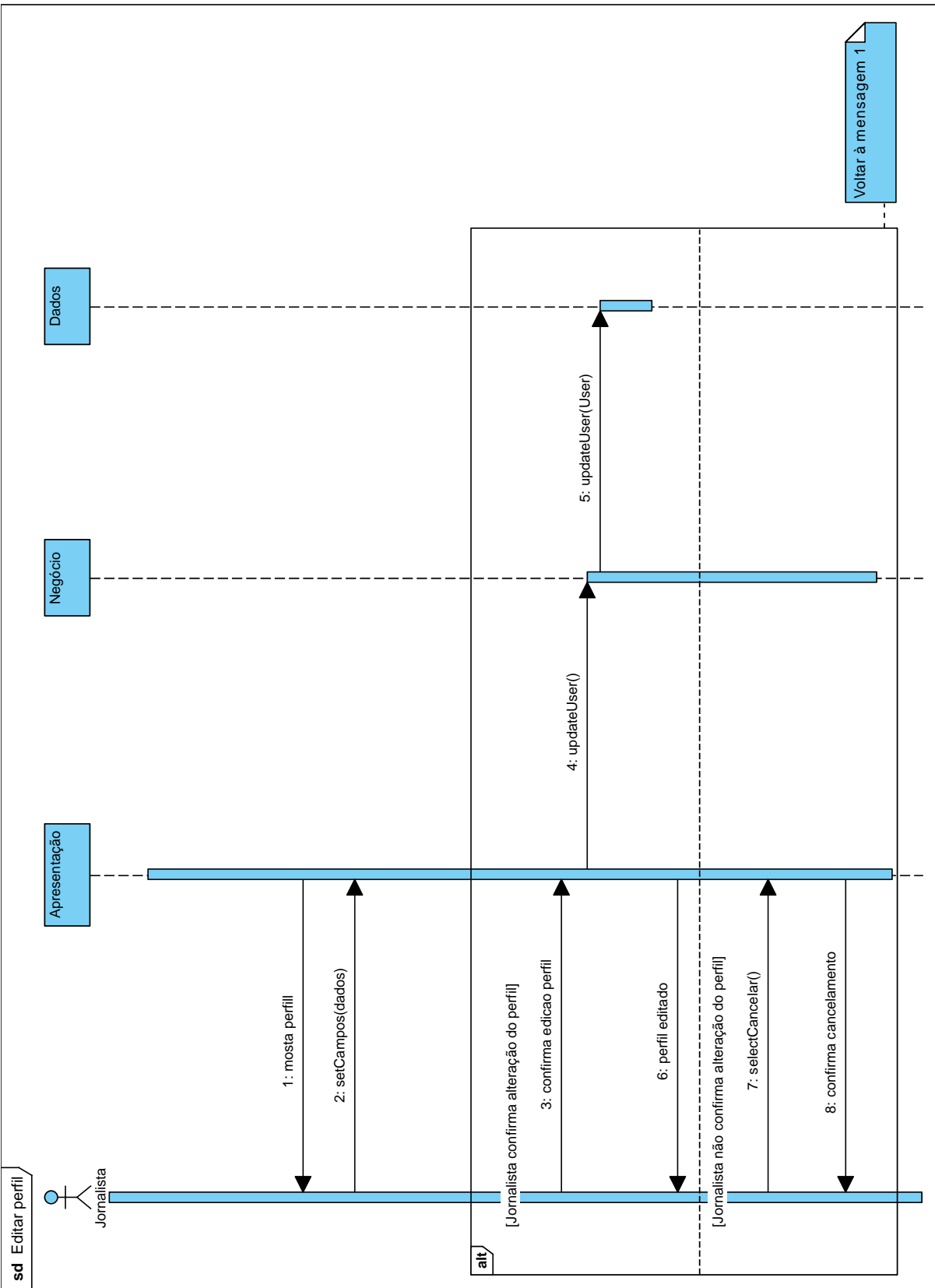


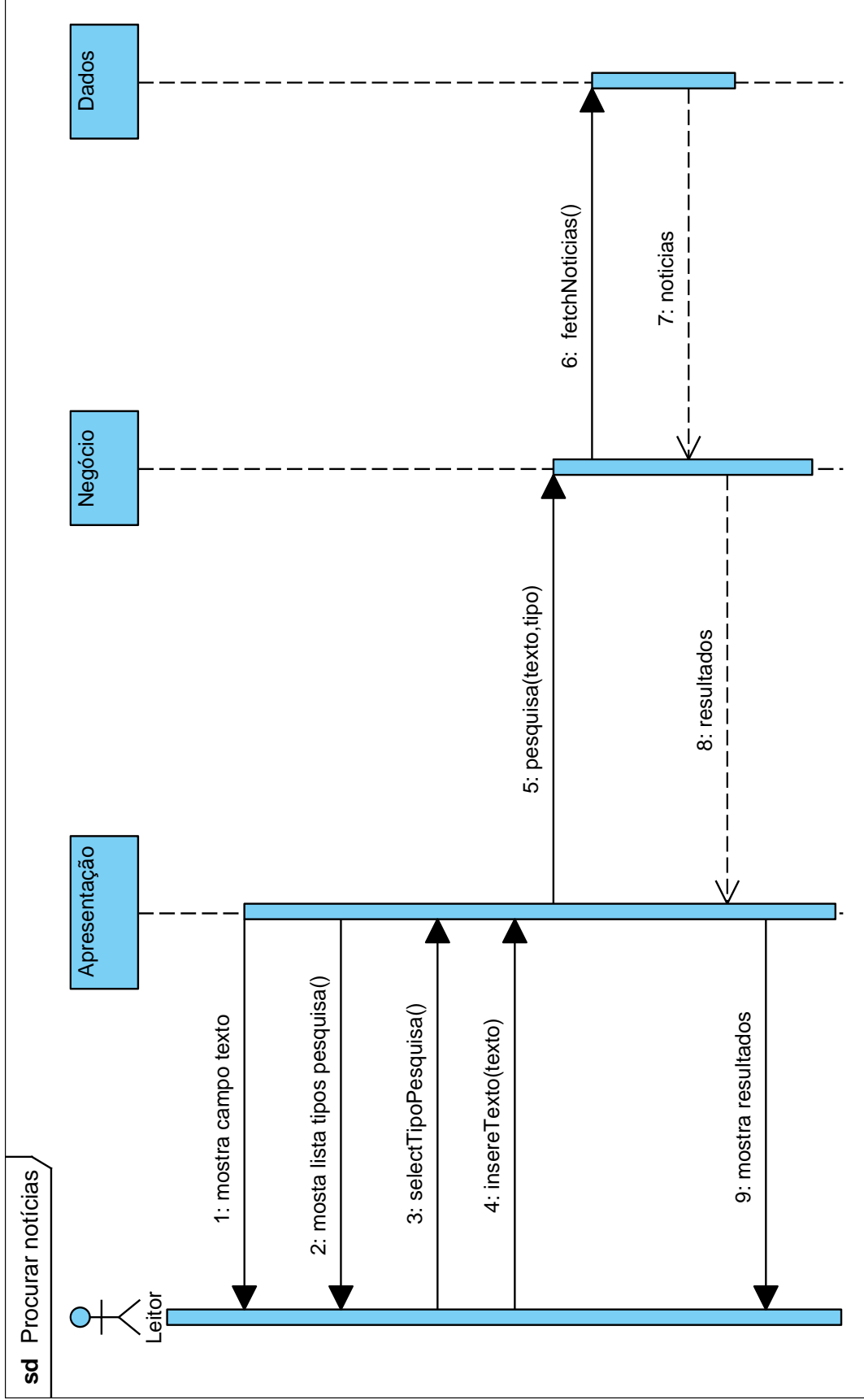


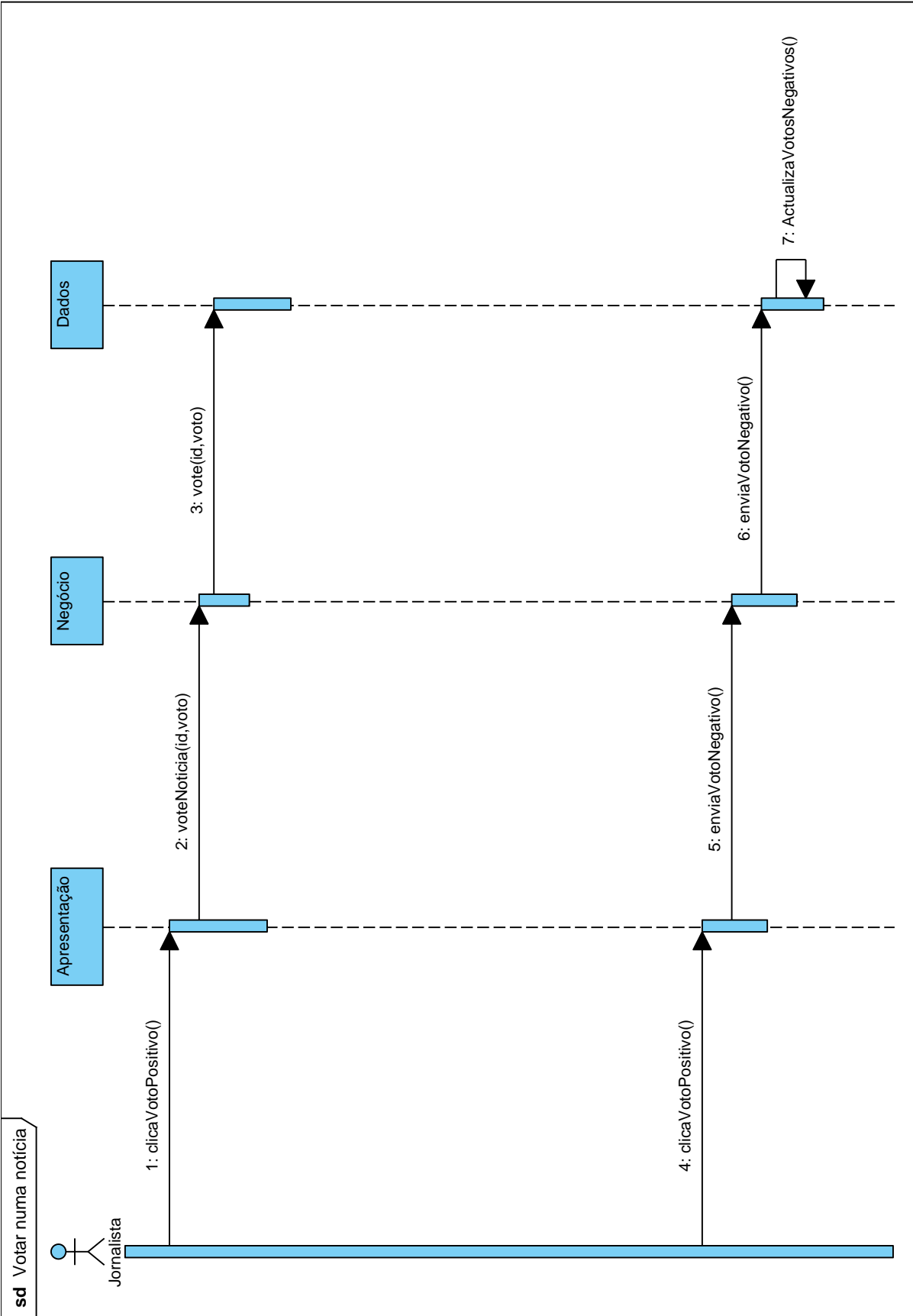


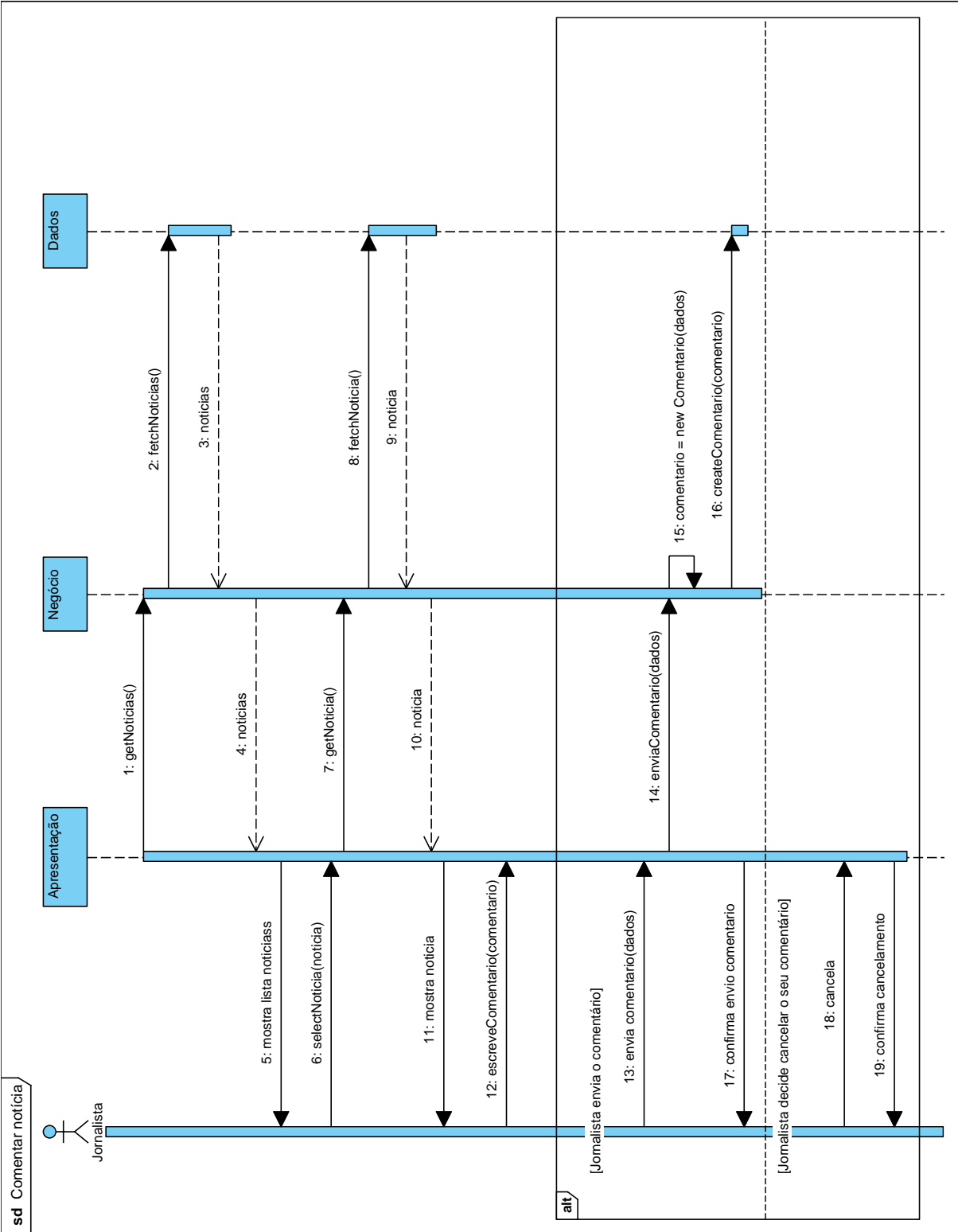


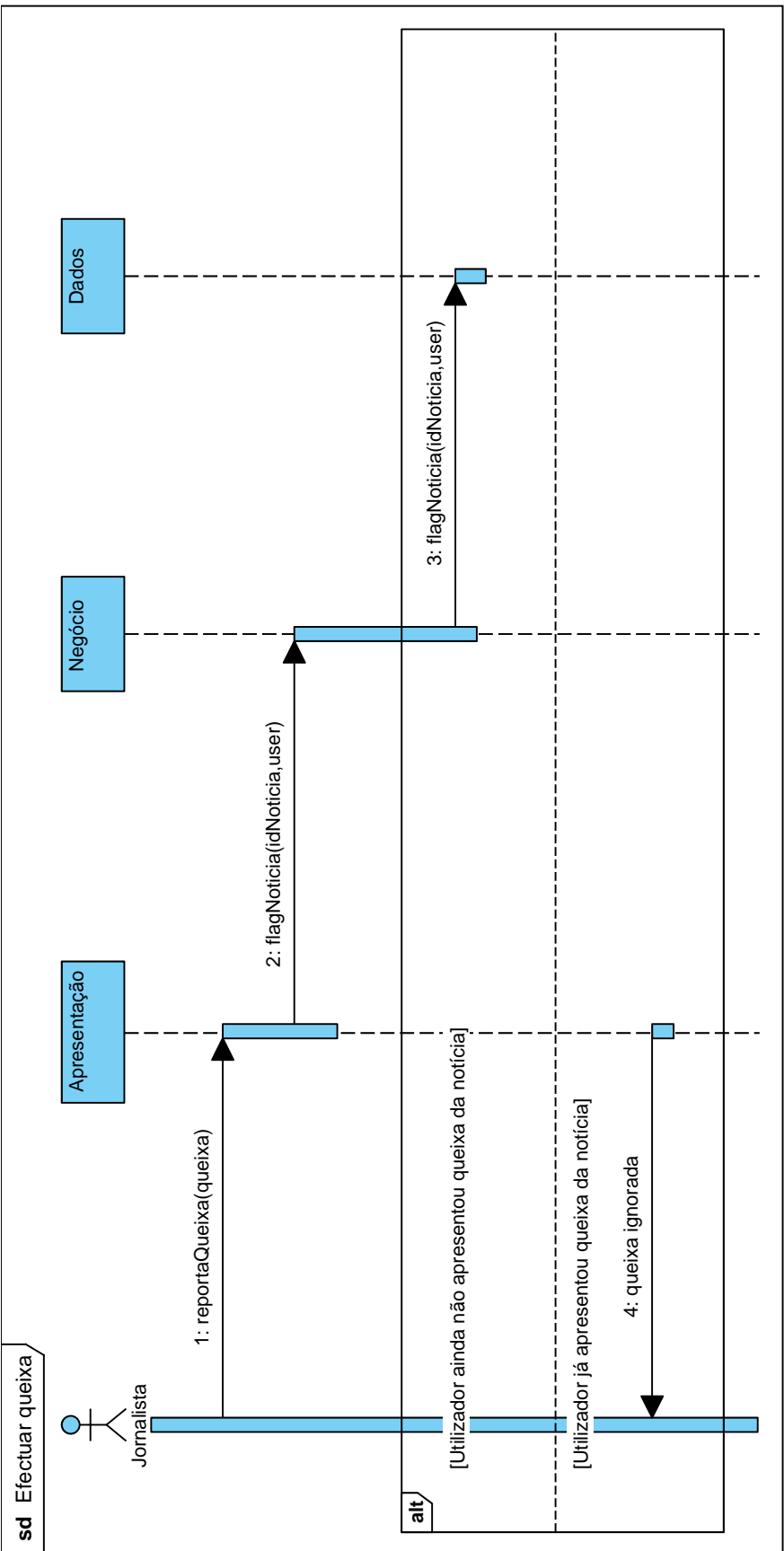


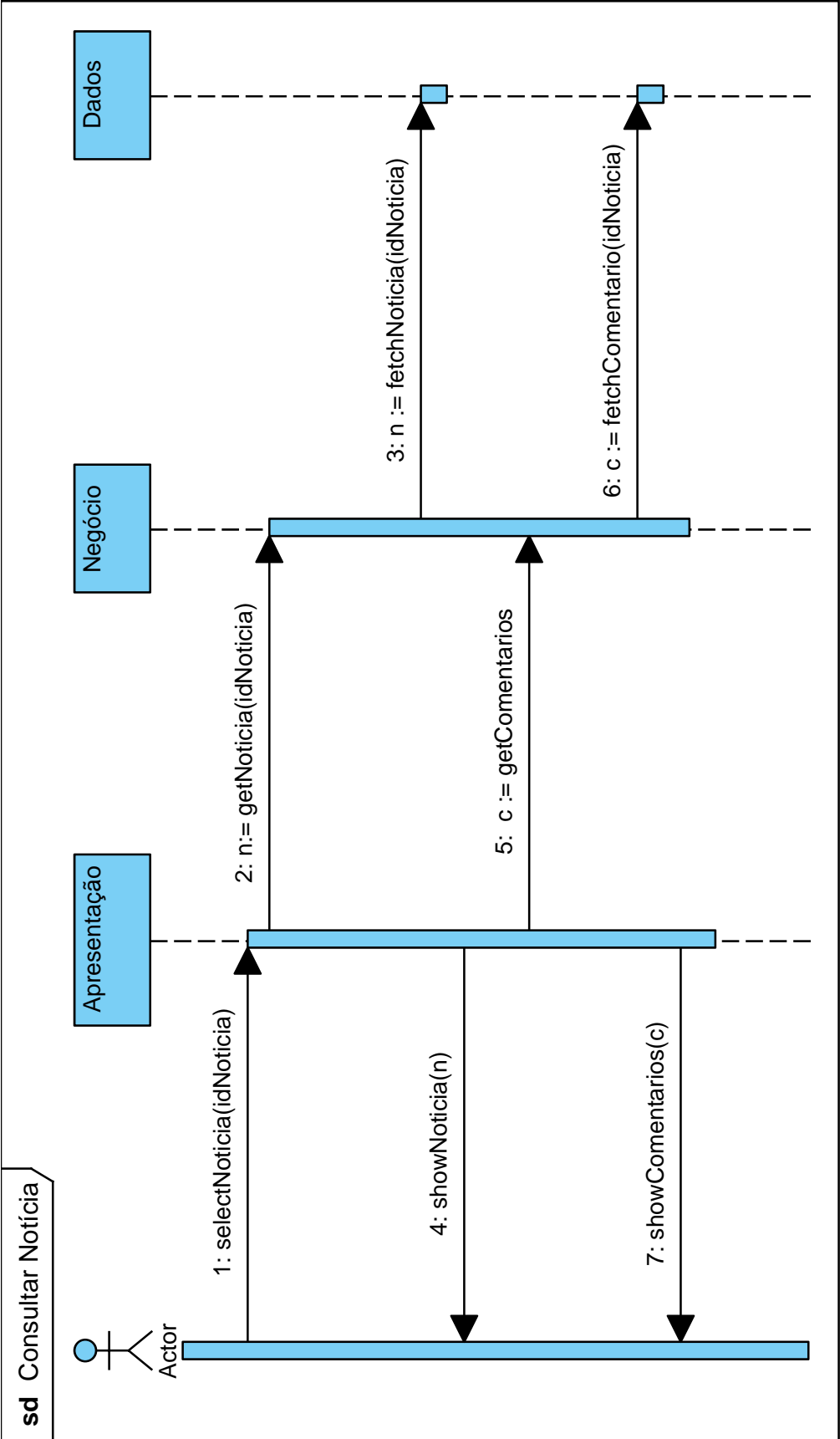


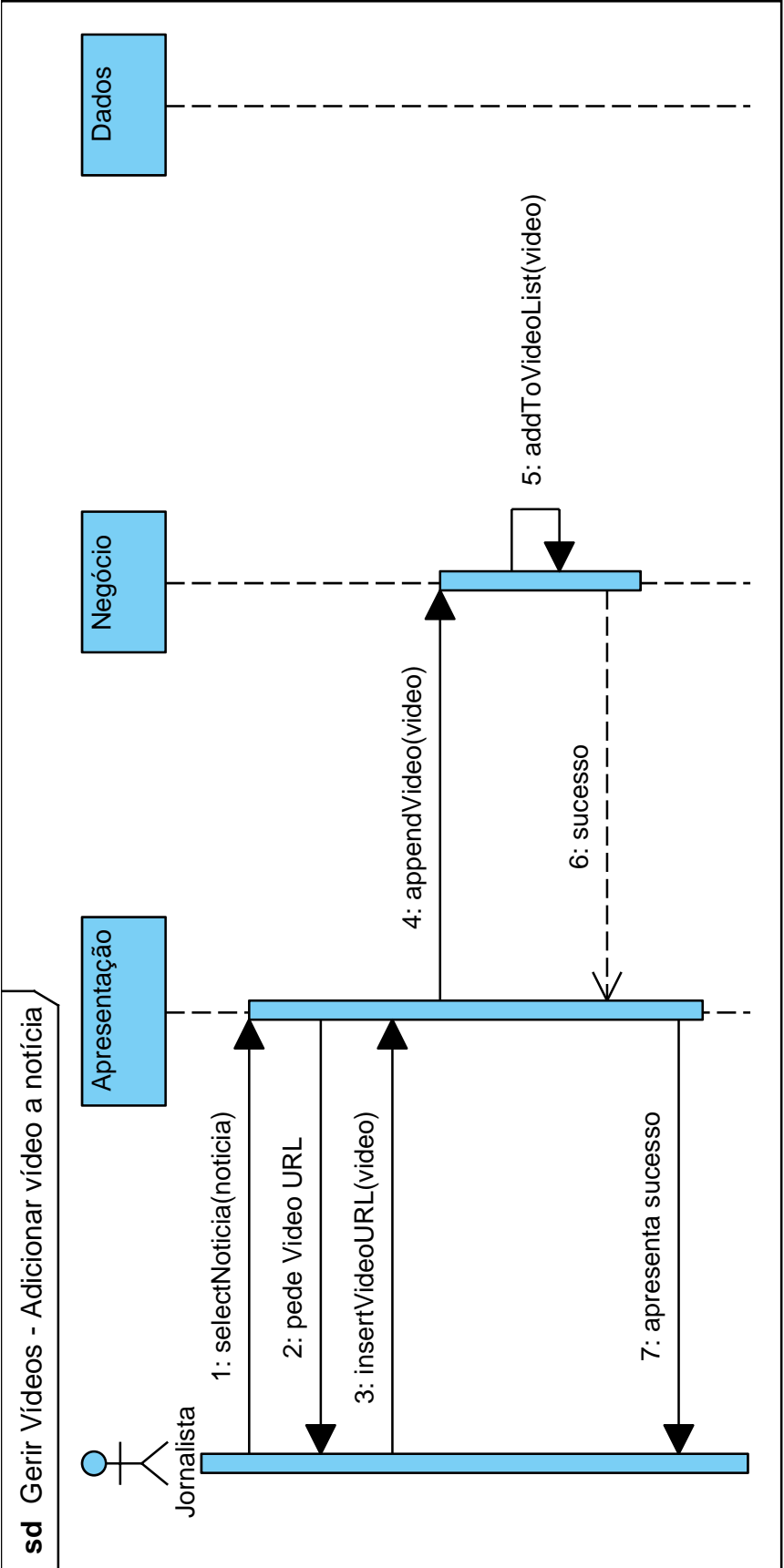


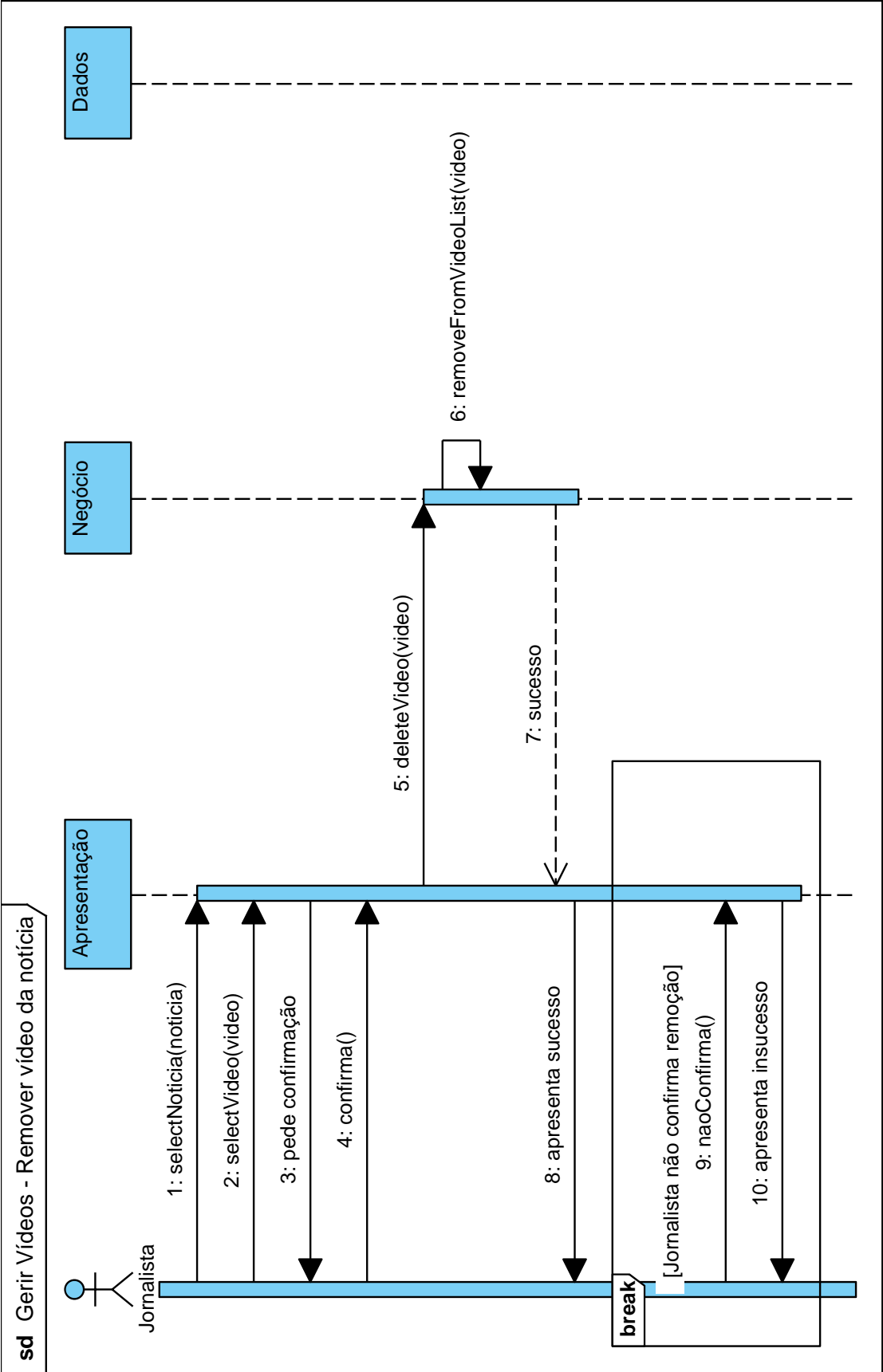


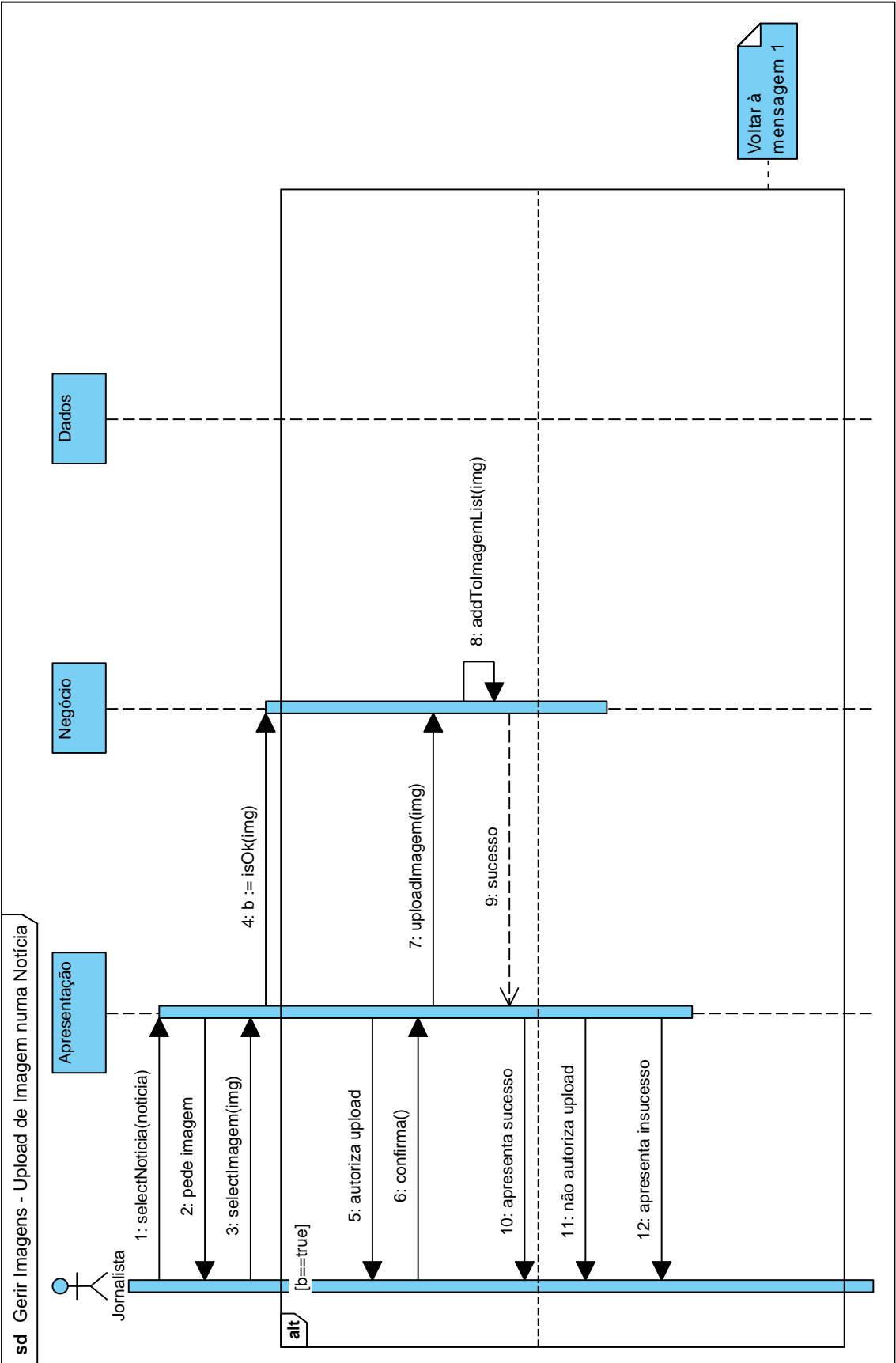


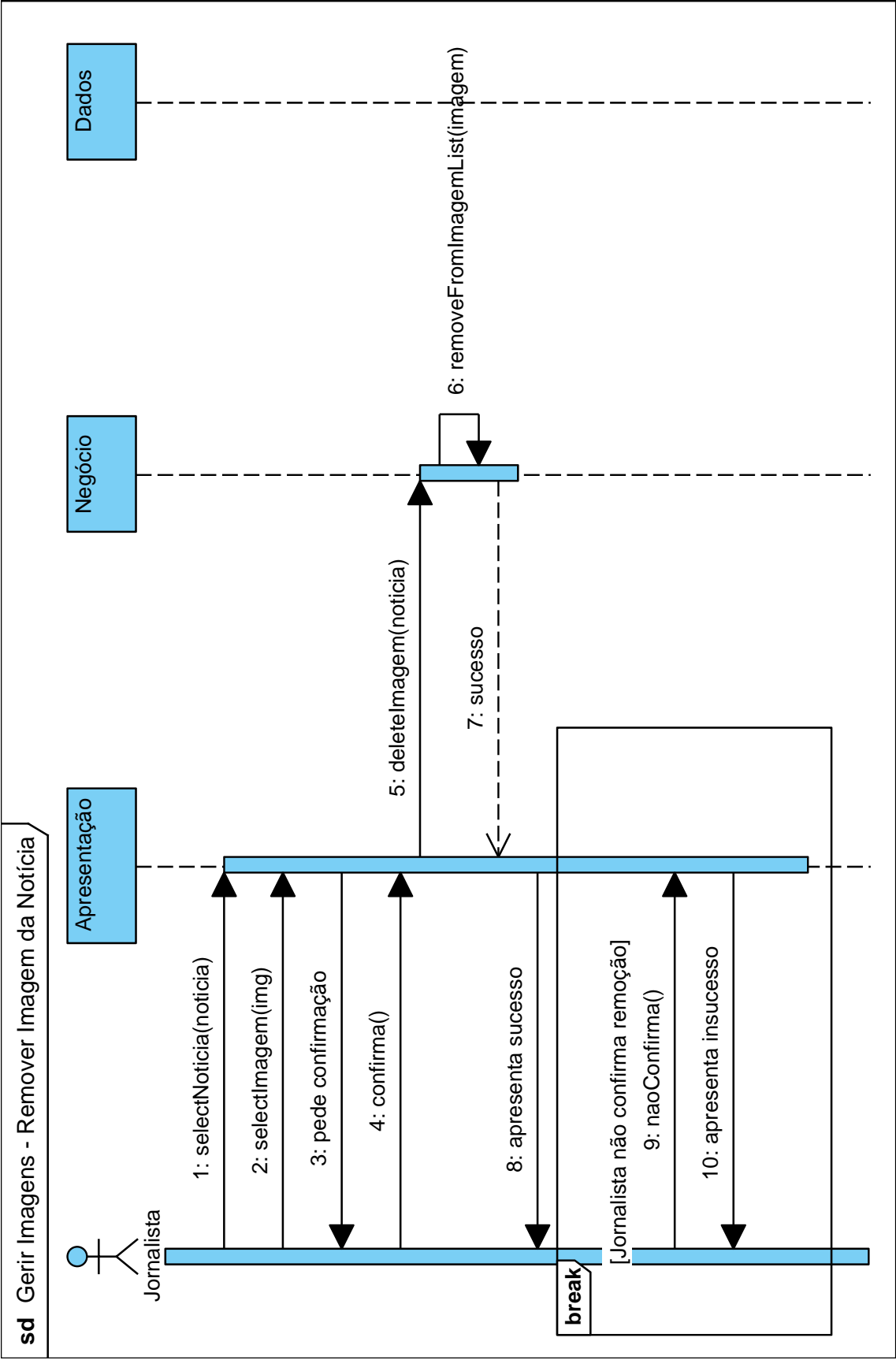


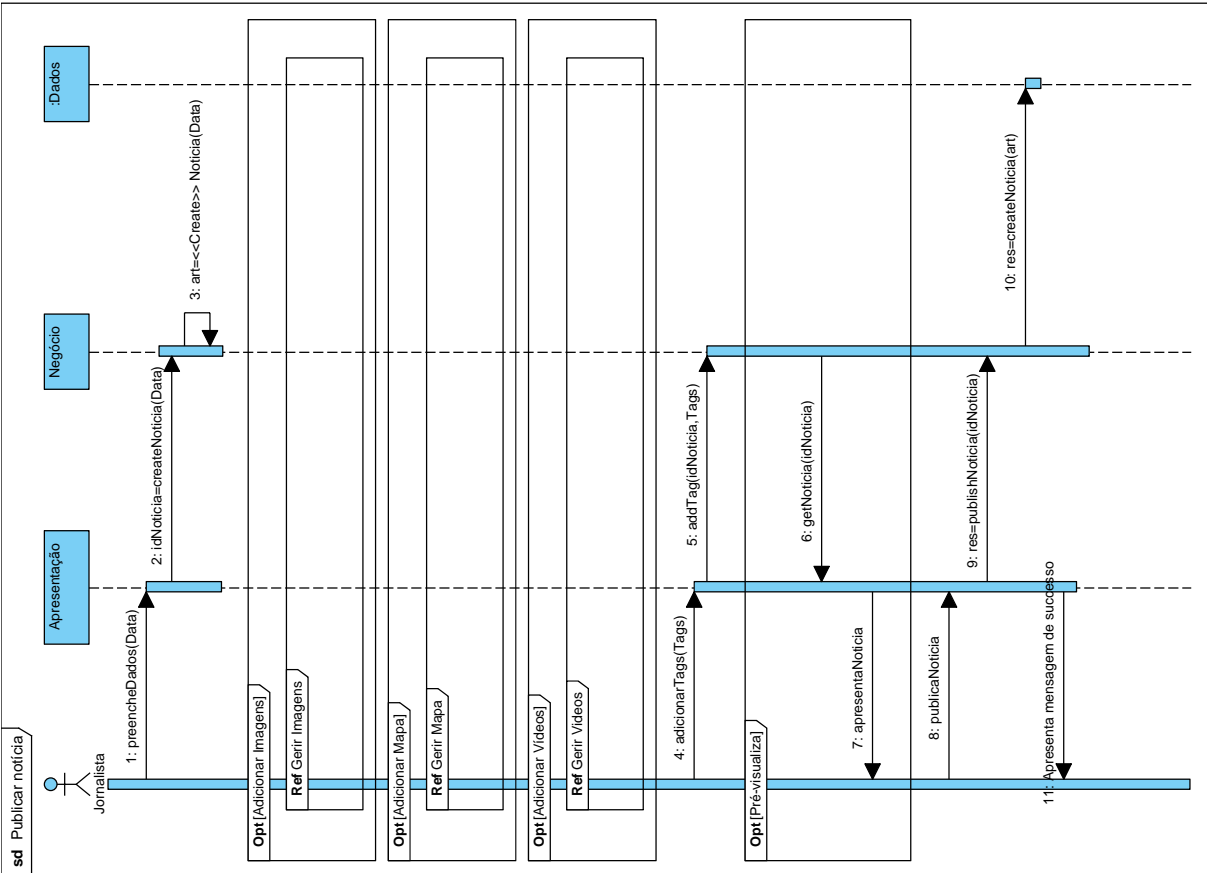


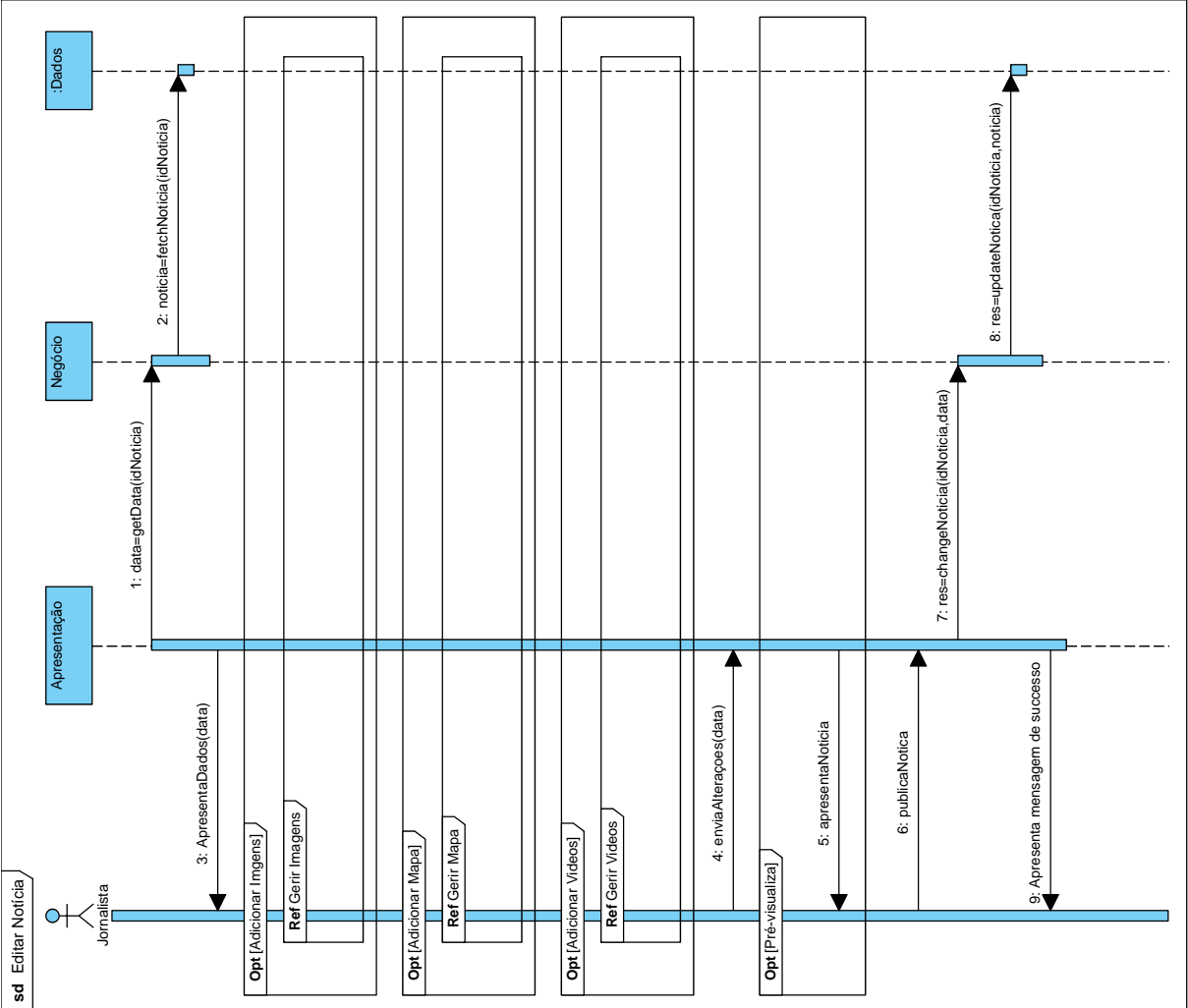


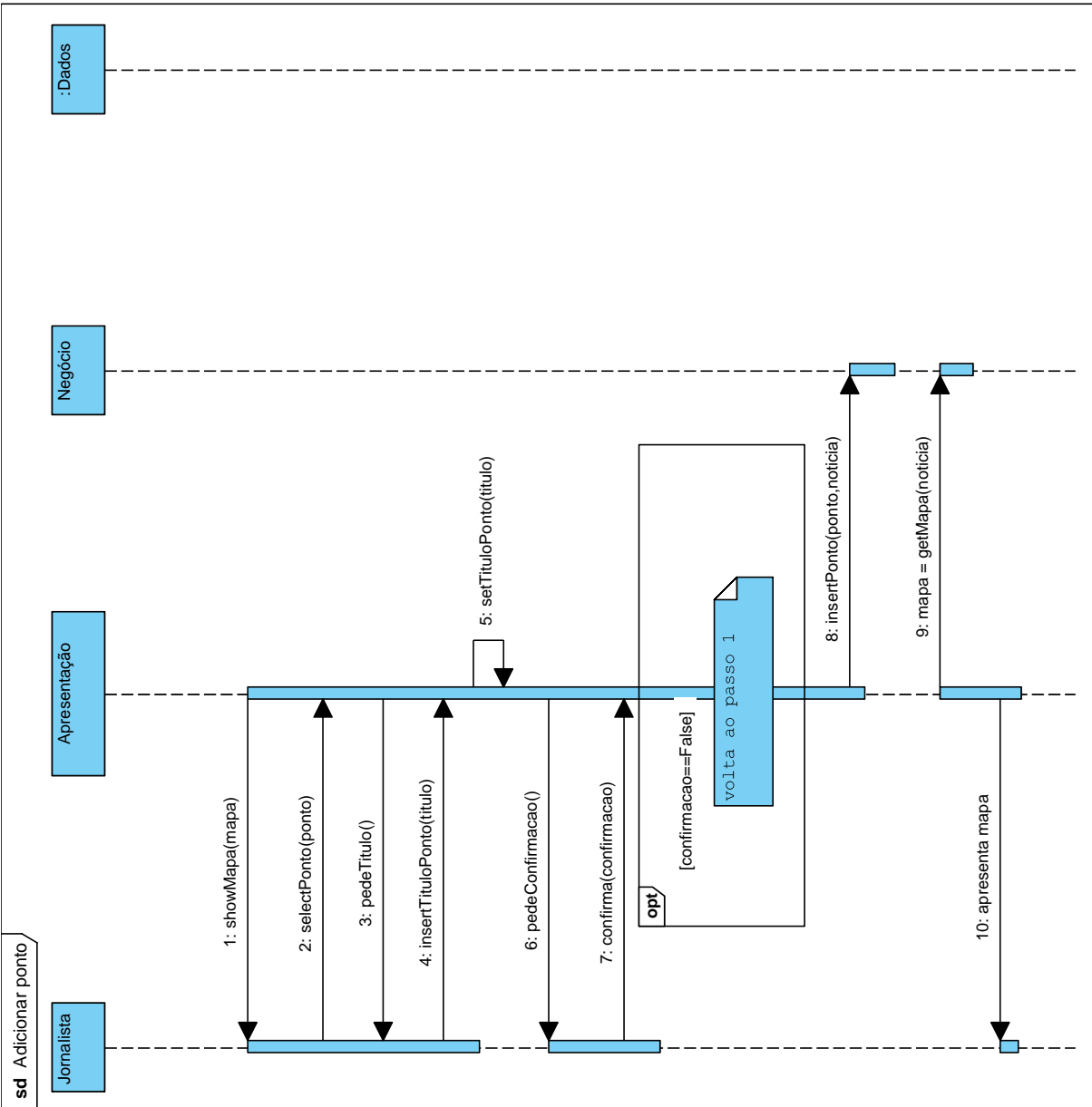


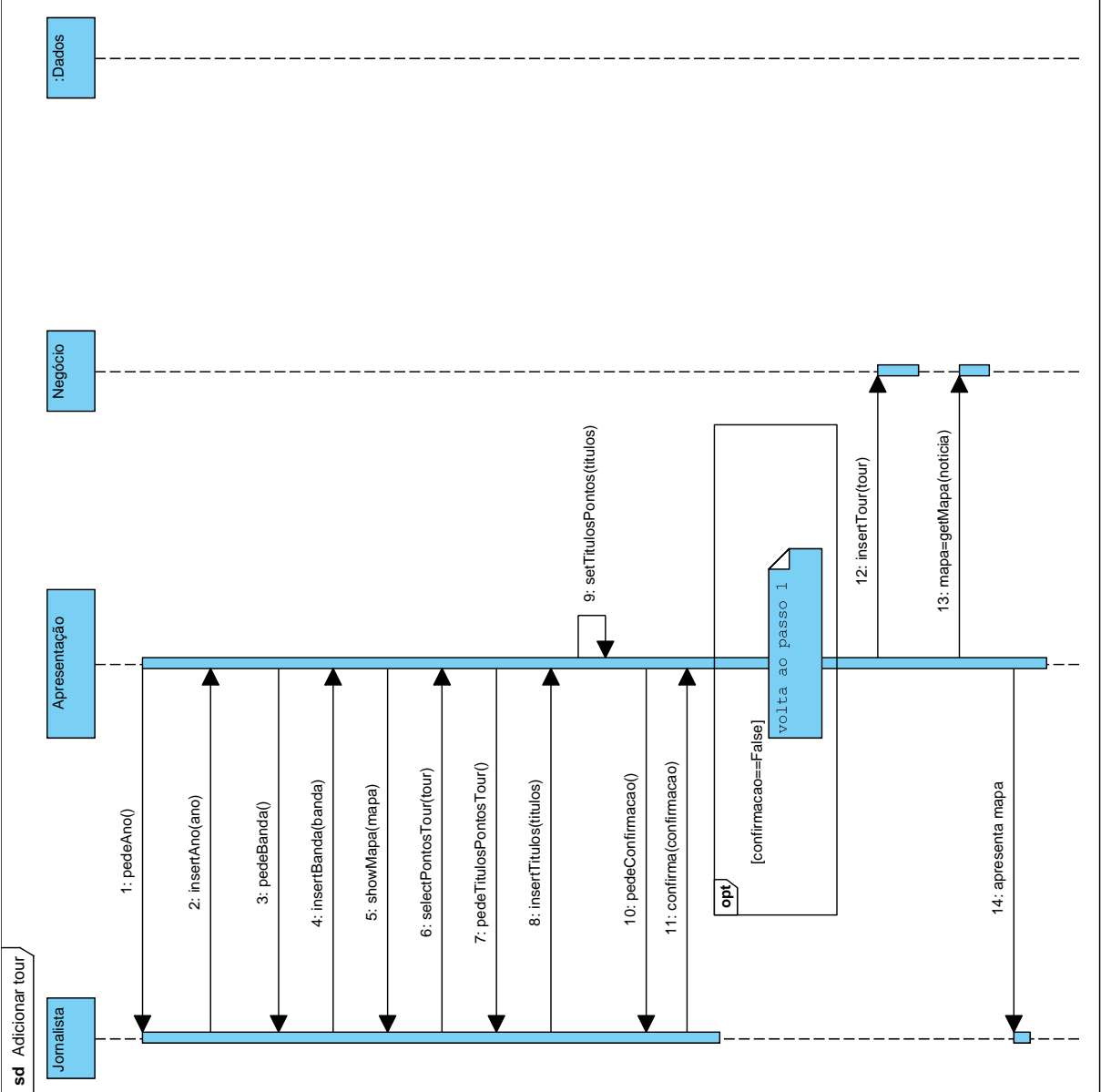


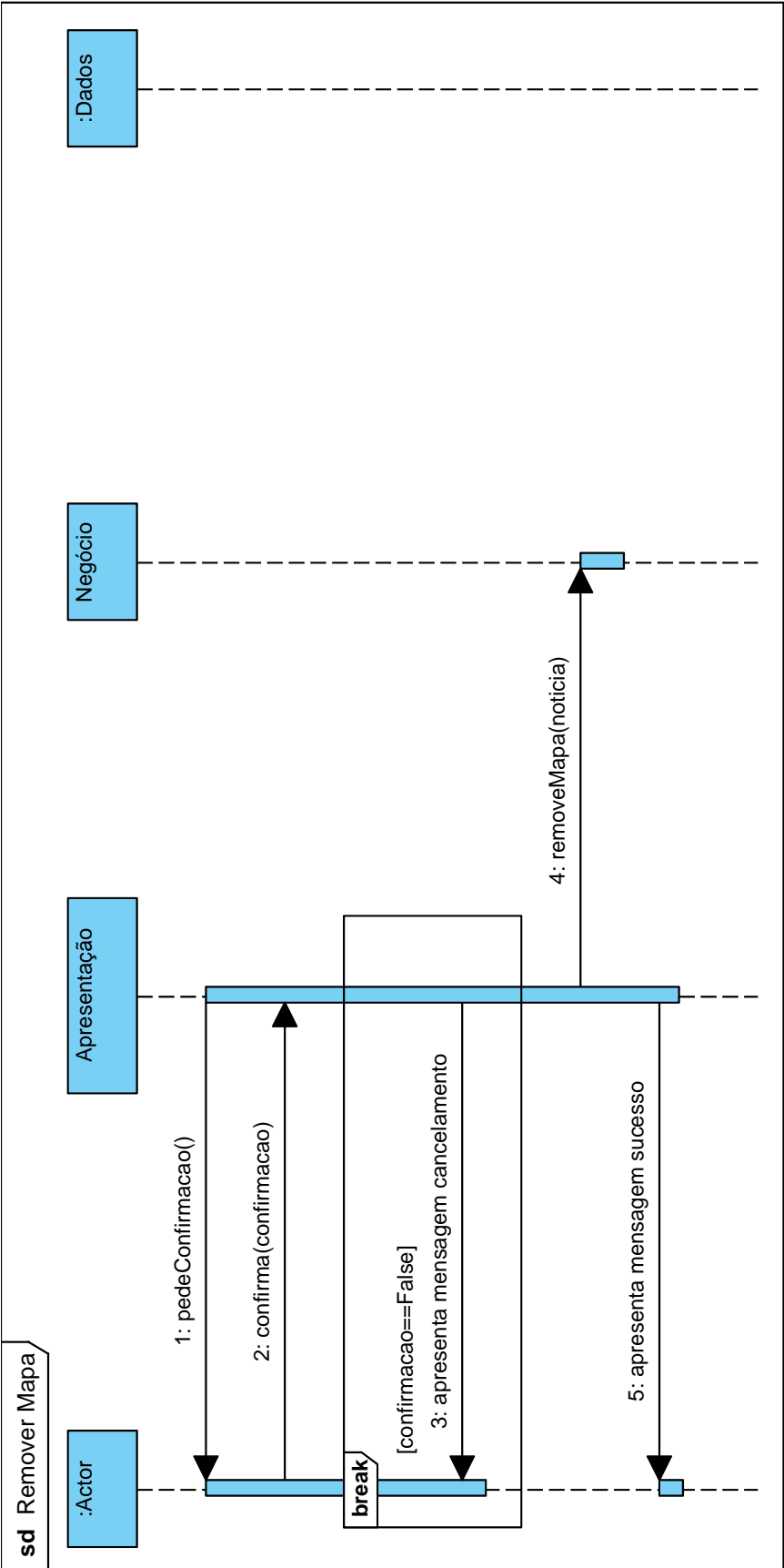


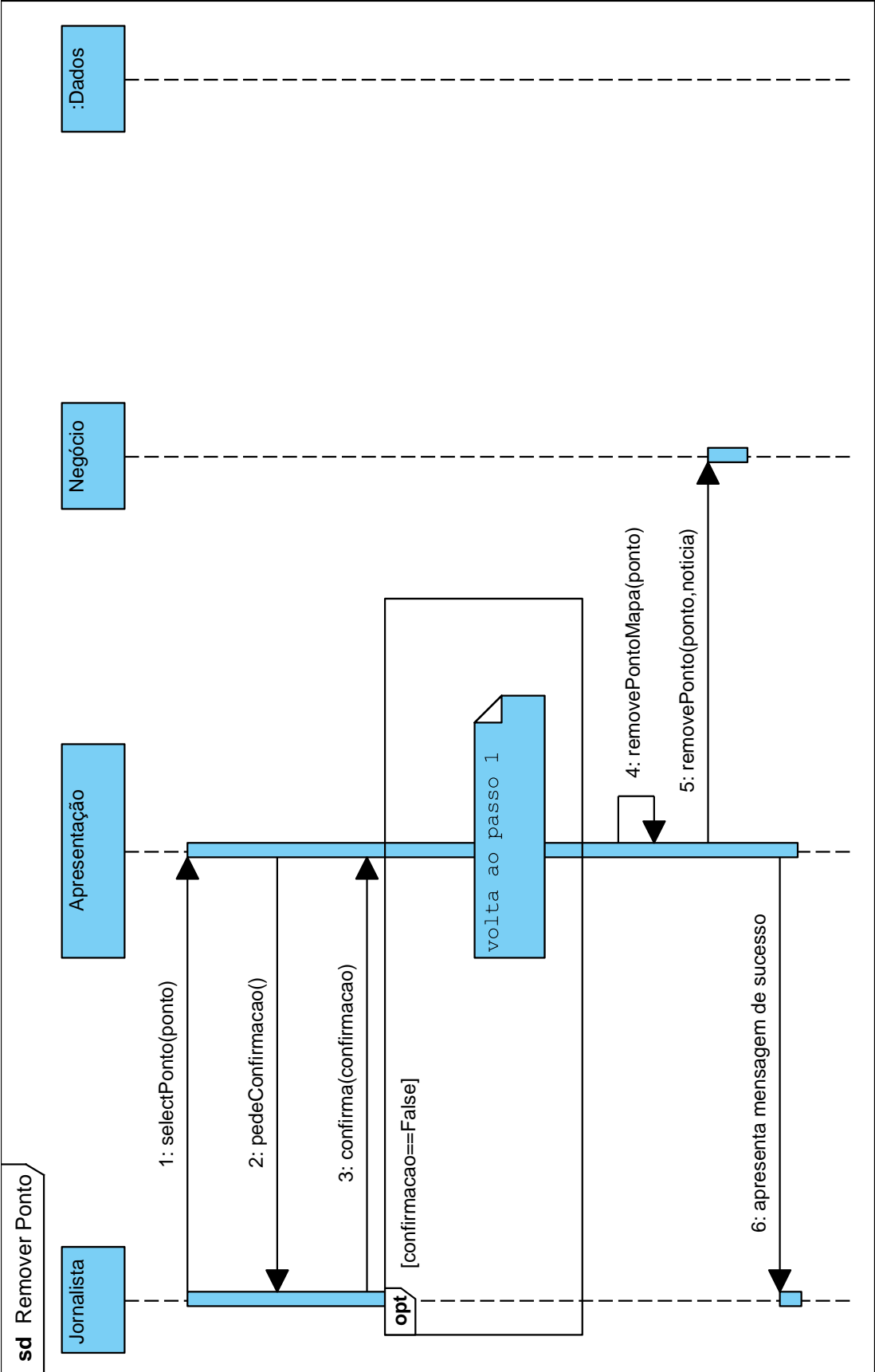


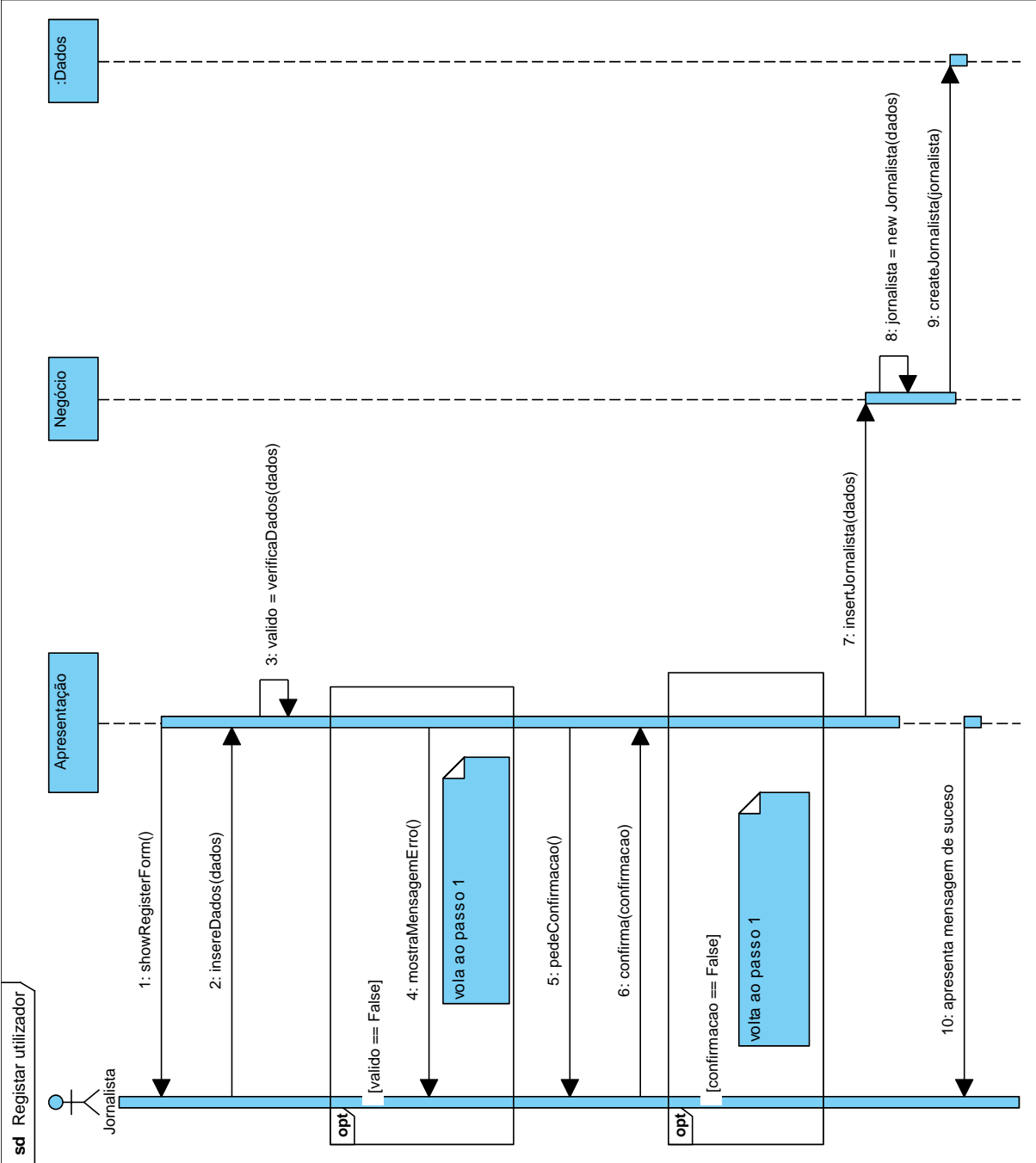


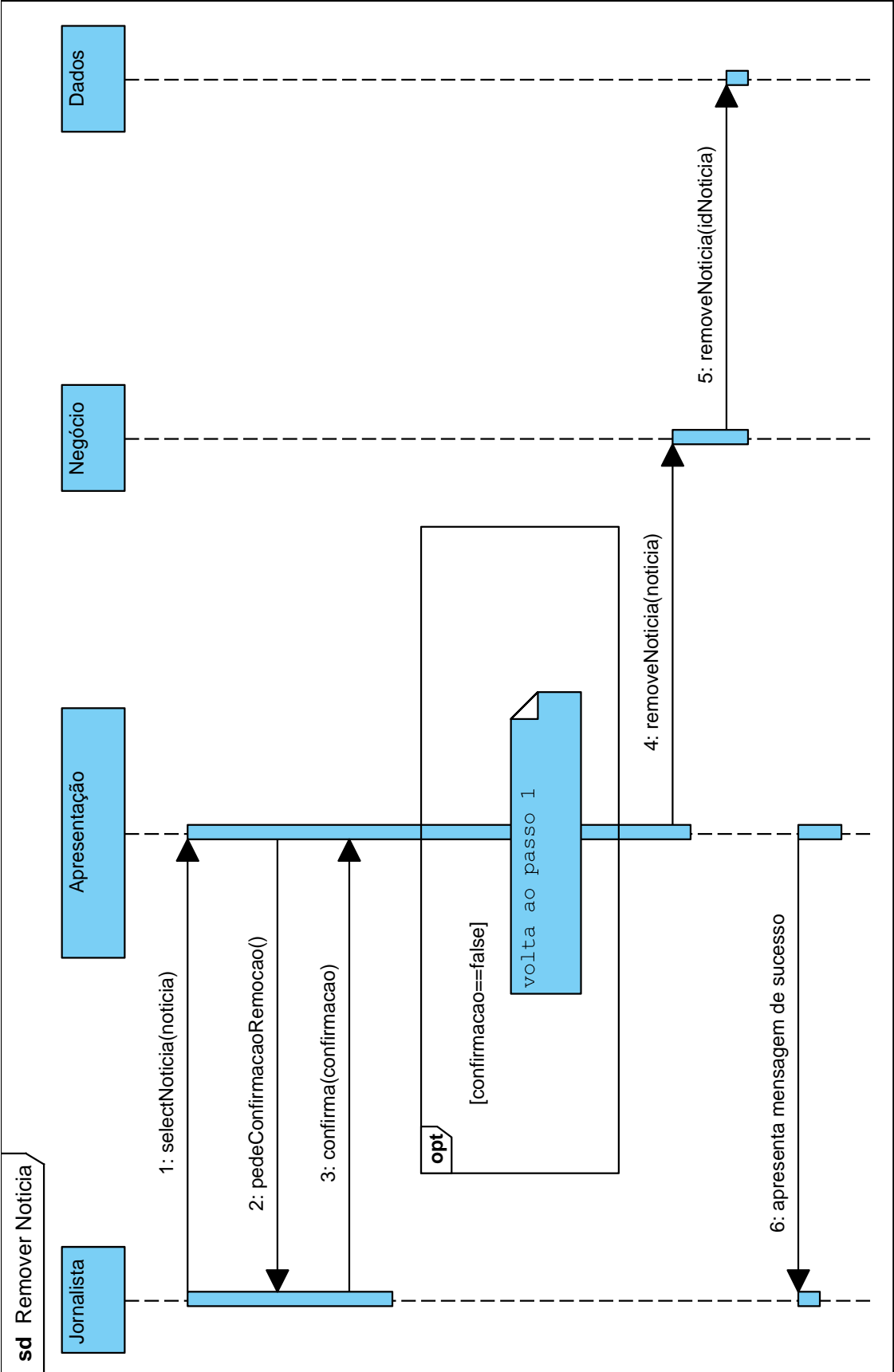




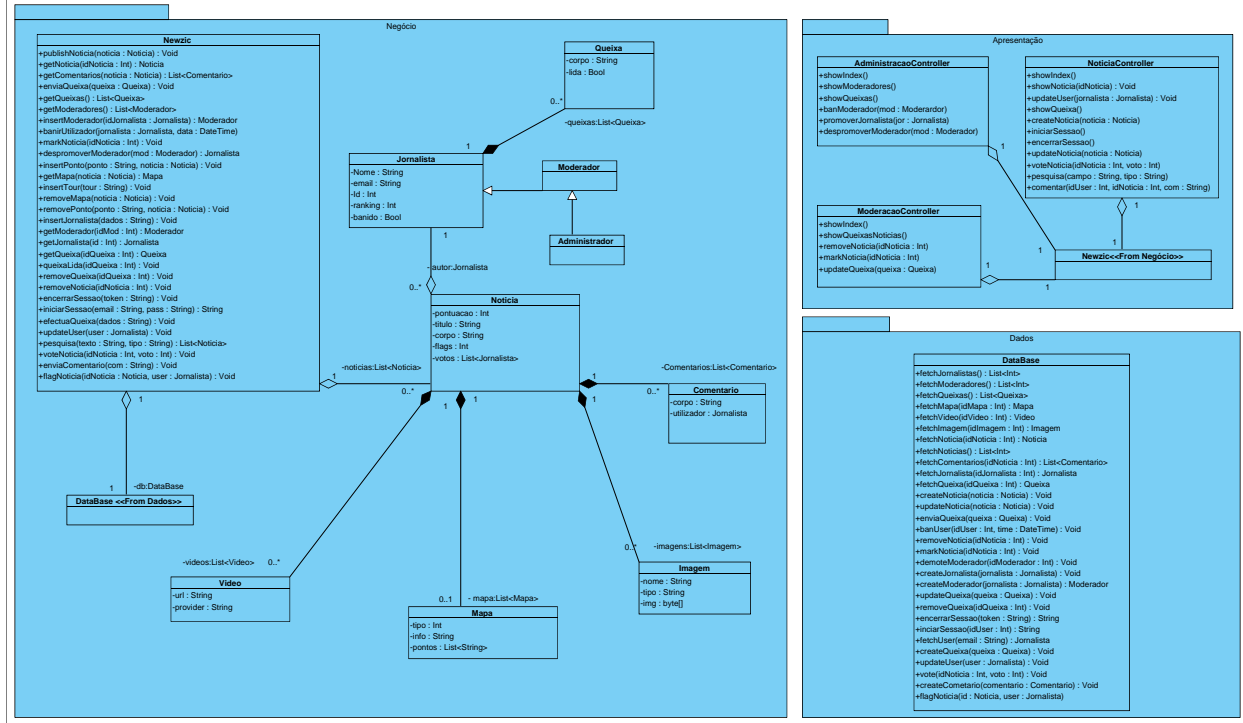


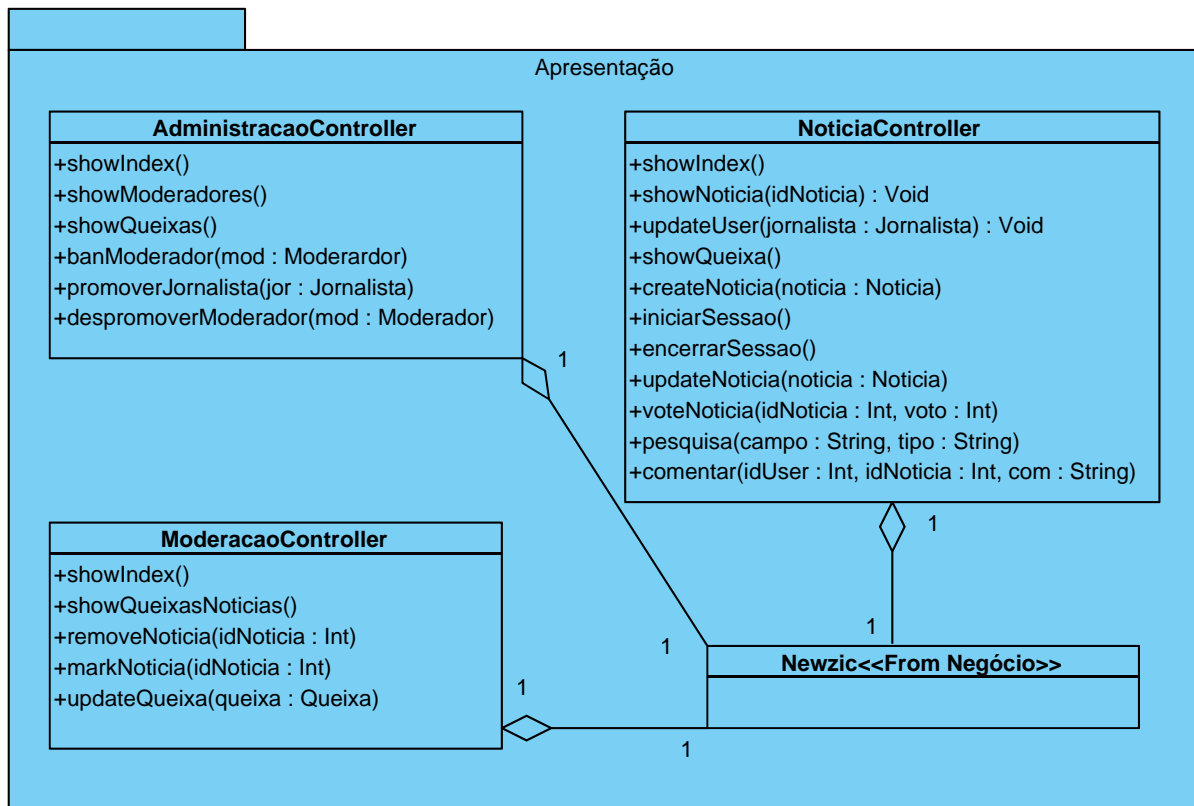


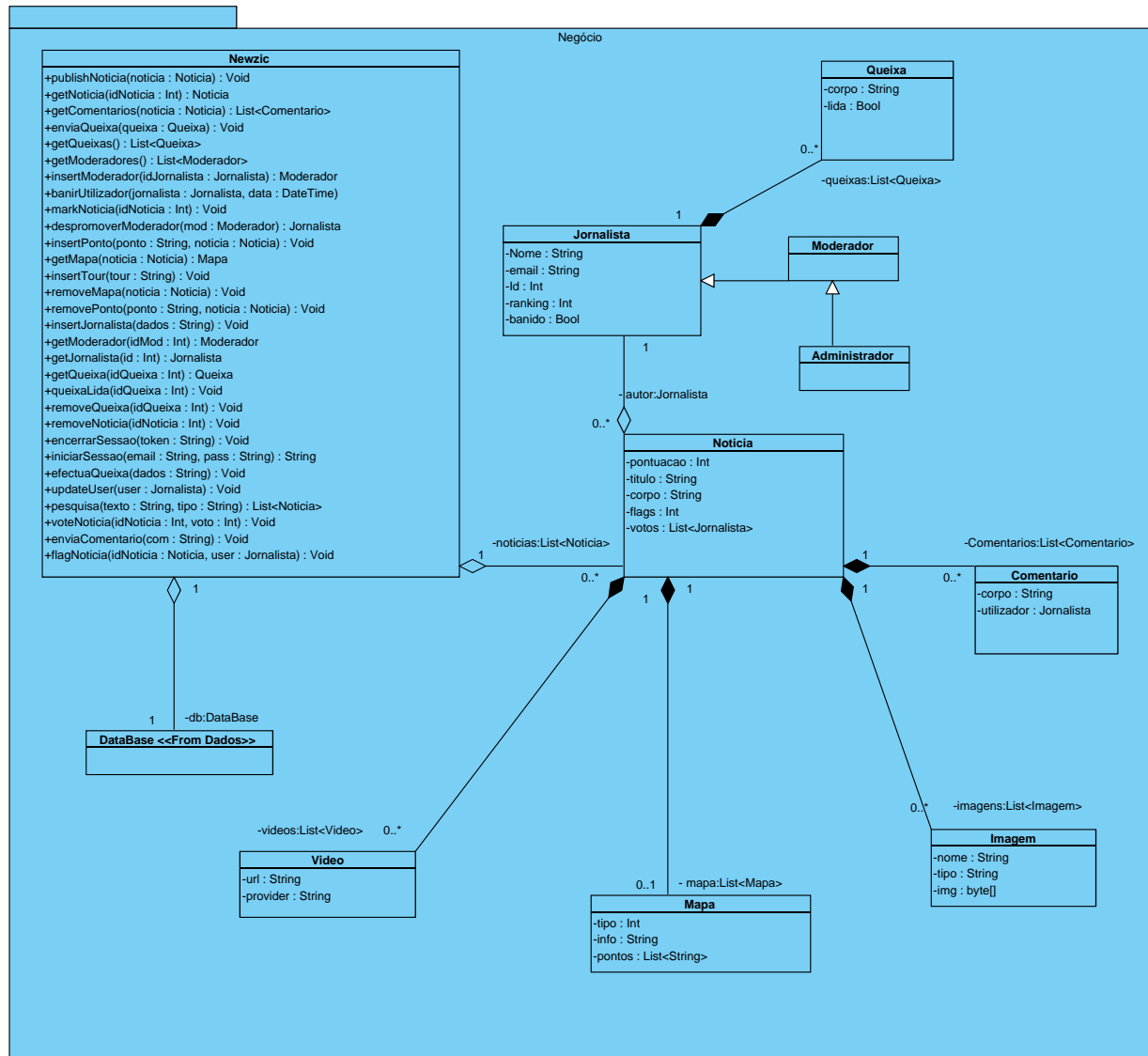




VIII. Diagrama de Classes







Dados

DataBase

```
+fetchJornalistas() : List<Int>
+fetchModeradores() : List<Int>
+fetchQueixas() : List<Queixa>
+fetchMapa(idMapa : Int) : Mapa
+fetchVideo(idVideo : Int) : Video
+fetchImagem(idImagem : Int) : Imagem
+fetchNoticia(idNoticia : Int) : Noticia
+fetchNoticias() : List<Int>
+fetchComentarios(idNoticia : Int) : List<Comentario>
+fetchJornalista(idJornalista : Int) : Jornalista
+fetchQueixa(idQueixa : Int) : Queixa
+createNoticia(noticia : Noticia) : Void
+updateNoticia(noticia : Noticia) : Void
+enviaQueixa(queixa : Queixa) : Void
+banUser(idUser : Int, time : DateTime) : Void
+removeNoticia(idNoticia : Int) : Void
+markNoticia(idNoticia : Int) : Void
+demoteModerador(idModerador : Int) : Void
+createJornalista(jornalista : Jornalista) : Void
+createModerador(jornalista : Jornalista) : Moderador
+updateQueixa(queixa : Queixa) : Void
+removeQueixa(idQueixa : Int) : Void
+encerrarSessao(token : String) : String
+inciarSessao(idUser : Int) : String
+fetchUser(email : String) : Jornalista
+createQueixa(queixa : Queixa) : Void
+updateUser(user : Jornalista) : Void
+vote(idNoticia : Int, voto : Int) : Void
+createComentario(comentario : Comentario) : Void
+flagNoticia(id : Noticia, user : Jornalista)
```

IX. Modelos E-R

