



Universidade do Minho

Conselho de Cursos de Engenharia

Licenciatura em Engenharia Informática

3ºAno

Disciplina de Laboratórios de Informática IV

Ano Lectivo de 2008/2009

Pérgles – Relatório da 2ª fase de Projecto

**Bruno Amorim, Christophe Gonçalves,
Martinho Fernandes, Ricardo Alves**

Data de Recepção	
Responsável	
Avaliação	

Péricles – Relatório da 2ª fase de Projecto

Bruno Amorim
Christophe Gonçalves
Martinho Fernandes
Ricardo Alves

Abril, 2009

Resumo

Este documento expõe uma alternativa ao método eleitoral tradicional. Ao abordar o problema de forma diferente, esta visa colmatar algumas falhas do sistema eleitoral actual.

Área de Aplicação: Eleições Autárquicas, Legislativas, Presidenciais, Europeias e Referendos Nacionais ou Europeus.

Palavras-Chave: Bases de dados relacionais, .NET Framework, C#, SOAP, WSDL, SQL, Eleições, Votação Online, Webservice, Website, Aplicação de Administração, Péricles, UML, Google Maps, ASP.NET.

Índice

1. Introdução	1
1.1. Contextualização	1
1.2. Apresentação do Caso de Estudo	1
1.3. Motivação e Objectivos	1
2. Fundamentação	3
2.1. Website	4
2.2. Aplicação de administração	5
2.3. Péricles	6
3. Especificação	8
3.1. Webservice	8
3.1.1 Consulta	8
3.1.2 Voto	10
3.1.3 Suporte	11
3.1.4 Administração	12
3.1.5 Sessão	14
3.2. Aplicação de administração	15
3.3. Website	16
3.4. Formato de dados LINX	20
3.5. Tecnologias e Ferramentas utilizadas	21
4. Conclusão	22

Bibliografia	23
Referências WWW	24
Lista de Siglas e Acrônimos	25
 Anexos	
I. Use Cases	27
II. Diagramas de Sequência	55
III. Diagrama de Classes	56
IV. Especificação da Base da Dados	57

Índice de Figuras

Ilustração 1 - Componentes do sistema	3
Ilustração 2 - Aplicações clientes do <i>web service Péricles</i>	6

1. Introdução

1.1. Contextualização

Em democracia o povo toma decisões através de eleições. Tradicionalmente o eleitor dirige-se a uma mesa de voto, escolhe o seu candidato, preenche um boletim de voto, tradicionalmente em papel, e deposita-o numa urna. No final do período de voto os membros da mesa de voto efectuem a contagem manual dos votos.

Actualmente estamos a assistir a uma mudança na direcção do voto electrónico. O processo é bastante semelhante, sendo a única diferença o preenchimento do boletim de voto. Num sistema de voto electrónico o eleitor dirige-se à mesa de voto e efectua o seu voto num terminal especificamente desenhado para esse fim. A contagem é efectuada automaticamente pelo sistema informático.

Apesar das vantagens oferecidas pelo método electrónico, este ainda encontra forte resistência à mudança.

1.2. Apresentação do Caso de Estudo

Desenvolvemos um sistema de voto electrónico que permite que os eleitores efectuem o sufrágio através da Internet, ao qual chamamos Péricles. Desta forma os eleitores podem votar nas mesas de voto usando terminais públicos, ou em casa usando os seus próprios terminais.

1.3. Motivação e Objectivos

Tanto o método de voto tradicional como o método electrónico falham em resolver o problema recorrente da falta de afluência às urnas, a malfadada abstenção. Uma das principais causas desse problema é a falta de vontade dos eleitores para se deslocarem às urnas.

Um sistema de voto que elimine essa necessidade poderia reduzir significativamente as taxas de abstenção, fazendo com que o sistema eleitoral reflecta mais fielmente a vontade do povo. É aqui que entra o *Péricles*.

O sistema também permite que os eleitores votem através do método tradicional sendo esses votos contados manualmente e incorporados no sistema posteriormente.

2. Fundamentação

O nosso sistema é constituído por 4 componentes: um *web service* (Péricles), uma aplicação administrativa, um *website* público e uma base de dados. O Péricles assenta sobre essa base de dados e oferece uma API que é usada pelo website e pela aplicação administrativa.

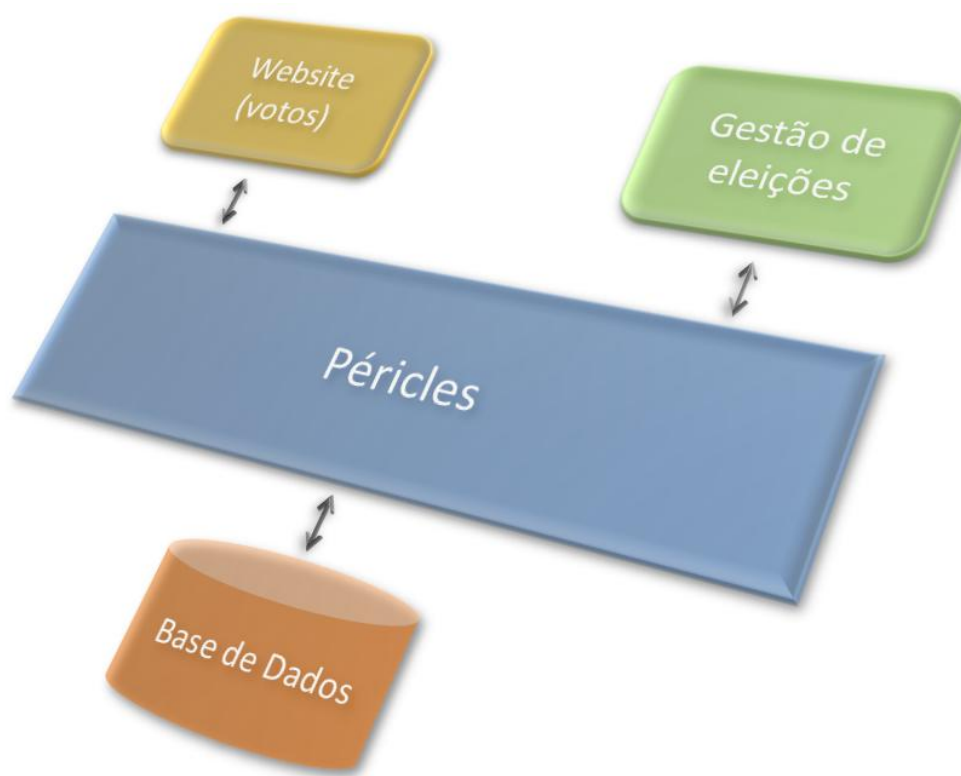


Ilustração 1 - Componentes do sistema

2.1. Website

Para um eleitor exercer o seu direito de voto online, tem à sua disposição um *website* onde o pode fazer.

No entanto, antes de poder utilizar este método de voto, o eleitor tem de obter uma senha de acesso. Essa senha deve ser adquirida anteriormente às eleições num período definido pelo administrador do sistema. Este processo serve para evitar situações fraudulentas, garantindo a integridade dos votos. Sem esta senha o eleitor não poderá efectuar o *login* a partir do seu computador pessoal para exercer o seu direito de voto desta forma.

Depois do eleitor obter a senha e efectuar o *login*, é-lhe apresentada a lista de eleições em que pode votar, no caso de haver mais do que uma eleição no mesmo dia. O eleitor pode optar por votar numa ou mais eleições ao mesmo tempo ou por votar em várias fases (por exemplo, no período da manhã vota numa e no período da tarde vota noutra). Também pode optar por não fazer nenhuma votação, fazendo *logout* no sistema.

Não é permitido alterar o voto, ou seja, se o eleitor votar num candidato/partido, posteriormente não poderá alterar o seu voto.

Os votos são registados de forma a manter o anonimato. Mais tarde os resultados são disponibilizados publicamente no *site*.

O site usa a API do *Google Maps* para apresentar os resultados de uma eleição de uma forma mais agradável, com recurso a mapas interactivos. Nestes mapas é possível distinguir os partidos/candidatos vencedores em cada região. O utilizador tem à sua disposição um conjunto de opções de modo a filtrar resultados com determinados critérios. Os critérios permitidos são os seguintes:

- visualizar apenas as regiões onde venceu um determinado partido;
- apresentar apenas regiões onde a abstenção foi superior a um determinado valor;
- apresentar apenas regiões onde o partido/candidato vencedor mudou;

As regiões podem ser preenchidas com cores ou símbolos dos respectivos partidos/candidatos. Também é possível visualizar os resultados sob a forma de gráficos em cima das regiões.

Seleccionando uma região disponibilizam-se resultados mais detalhados sobre a eleição em questão nessa região. Esses detalhes incluem:

- número total de votos;
- percentagem de abstenção;
- nome e foto (se disponível) de cada candidato, ou cabeça da lista do partido;
- número de votos de cada candidato/partido;
- número de votos nulos e brancos;

Em alternativa à visualização dos resultados directamente nos mapas, também é possível comparar os resultados actuais com eleições anteriores. Esta comparação é disponibilizada sob a forma de um gráfico, que representa vários critérios, número de votos, sejam estes de cada partido/candidato, nulos ou brancos, ou percentagem de abstenção.

Quando ocorre mais do que uma eleição no mesmo dia, o utilizador deve escolher quais os resultados que deseja visualizar, seja no mapa ou na comparação de resultados.

Os resultados estão apenas disponíveis após o fecho das urnas e depois de todas as juntas enviarem os seus resultados para a nossa aplicação.

O nosso sistema permite integrar votos que sejam efectuados pelo método tradicional, ou seja, em mesas de voto com boletins em papel. Deste modo, uma votação não tem de ser exclusivamente física ou electrónica (online).

De modo a permitir que os dados adquiridos nas mesas (número de votos e eleitores que votaram) possam ser integrados no sistema, o *síte* tem um local próprio para registar a contagem de votos da freguesia. Após confirmação, os dados são colocados na base de dados.

2.2. Aplicação de administração

A aplicação administrativa é uma aplicação que permite criar eleições, definir datas para as mesmas, definir os candidatos e adicionar utilizadores.

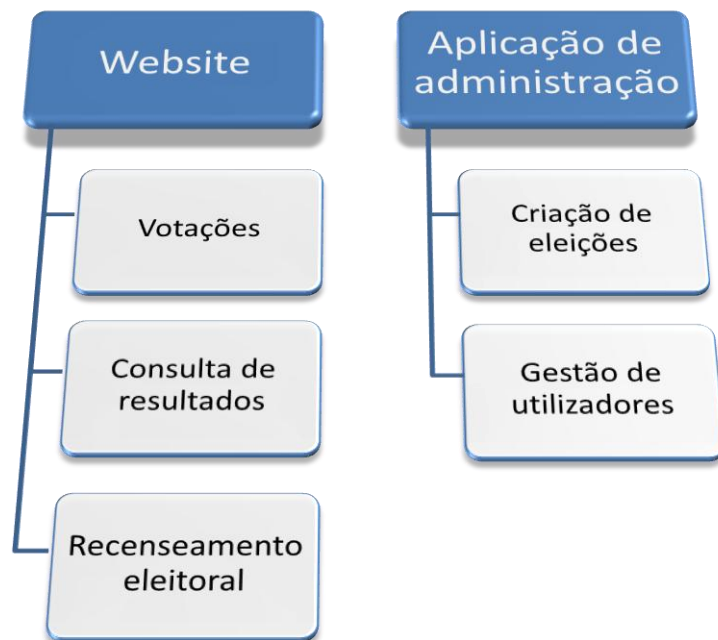


Ilustração 2 - Aplicações clientes do *web service Péricles*

2.3. Péricles

O Péricles é um *web service* que serve de núcleo ao nosso serviço de votação online. Este serviço é usado para a submissão, tratamento e consulta de todos os dados referentes a votações. O serviço permite que decorram várias votações em paralelo podendo os resultados ser consultados separadamente ou em conjunto dependendo da significância destes.

De modo a assegurar a segurança dos dados, o Péricles implementa quatro níveis de utilizador – administração, suporte, voto e consulta – e a comunicação é feita usando um canal de comunicação seguro.

O utilizador com privilégios de administração pode criar novas votações assim como cancelar ou alterar votações já existentes. Pode alterar parâmetros como modificar hora da abertura e fecho da votação, qual a votação e numero de opções, etc. Tem privilégios para adicionar

novas contas com qualquer nível de execução (suporte e eleitor). Para ter acesso a este nível é necessária autenticação prévia.

O utilizador com privilégios de suporte pode adicionar resultados manualmente. Estes podem vir de uma votação a decorrer fisicamente em paralelo com a que decorre no sistema. Cada conta de suporte está associada a uma e só uma votação – para um utilizador poder submeter resultados em mais do que uma votação deverá ter um número de contas de suporte igual ao número de votações a que pretende ter privilégios de adição de votos. Este nível de acesso requer autenticação.

O utilizador com privilégios de eleitor pode efectuar votos. A mesma conta de eleitor pode estar associada a uma ou mais eleições. Para ter acesso a este nível também é necessária autenticação.

O utilizador com privilégios de consulta pode consultar os resultados da votação a qualquer momento. Visto que a consulta é pública, este nível de utilizador é dado automaticamente a qualquer utilizador que não se identifique com nenhum outro nível de utilizador aquando da interacção com o sistema. Este é o único nível que não necessita de autenticação.

Todos os níveis de acesso são mutuamente exclusivos.

Para facilitar a integração do sistema com aplicações de terceiros, é disponibilizada uma API para interacção com o sistema. As outras partes do sistema (o *website* dos votos e a aplicação administrativa) também funcionam sobre esta API.

3. Especificação

3.1. Webservice

O *webservice* é o *core* do projecto. Tanto o *website* como a aplicação de administração assentam sobre o *webservice*. Este disponibiliza uma série de métodos que podem ser invocados remotamente por outras aplicações e variam consoante o nível de utilizador.

De seguida apresentamos os métodos disponíveis em cada nível de utilizador.

3.1.1 Consulta

- List<Election>getElectionsByYear(Int::Ano)

retorna uma lista de eleições de todas as eleições que estão a decorrer (ou decorreram) num dado ano. Se não houver eleições nesse ano é devolvida uma lista vazia.

- List<Election>::getElectionsByDate(DateTime::date)

retorna uma lista de eleições de todas as eleições que estão a decorrer (ou decorreram) numa dada data.

Se não houver eleições nessa data é devolvida uma lista vazia.

- Dictionary <Candidate,Int>::getVotes(Int::Eleição)

retorna um dicionário com o candidato (ou o nome da opção se for um referendo) e o numero de votos de uma dada eleição.

Será lançada uma exceção se o código da eleição estiver errado.

- Dictionary<Candidate,Int>::getVotes(Int::Eleição,Int::região)

retorna um dicionário com o candidato (ou o nome da opção se for um referendo) e o numero de votos de uma dada eleição e região.

Será lançada uma exceção se o código da eleição ou o código da região estiver errado.

- Int::getVotes(Int::Candidato,Int::Eleição)

retorna o numero de votos de um candidato numa dada eleição.

Será lançada uma exceção se o código do candidato ou o código da eleição estiverem errados.

- Int::getVotes(Int::Candidato,Int::Eleição,Int::Regiao)

retorna o numero de votos de um candidato numa dada eleição, numa dada região.

Será lançada uma exceção se o código do candidato ou o código da eleição estiverem errados.

- List <Candidate>::getCandidates(Int::Eleição)

retorna um dicionário com o código e nome do candidato.

Será lançada uma exceção se o código da eleição estiver errado ou a eleição em questão for um referendo.

- List<Party>::getPartys(Int::Eleição)

retorna uma lista com partidos presentes numa dada eleição.

Será lançada uma exceção se o código da eleição estiver errado ou a eleição em questão for um referendo.

- List<Candidate>::getCandidatesByParty(Int::Party)

retorna um dicionário com o código e nome do candidato por partido.

Será lançada uma exceção se o código da eleição estiver errado ou a eleição em questão for um referendo.

- String::getQuestion(Int::Eleição)

retorna a pergunta de um dado referendo.

Será lançada uma exceção se o código da eleição estiver errado ou a eleição em questão não for um referendo.

- List<AdministrativeRegion>::getAdministrativeRegions(Int::Eleição)

retorna uma lista de todas as regiões administrativas de uma dada eleição.

Será lançada uma exceção se o código da eleição estiver errado.

3.1.2 Voto

- Void::vote(Int::Candidato, Int::Eleição, String::Token)

Permite votar num candidato numa dada eleição (enviando o *token* gerado aquando do login). Será lançada uma exceção se houve um erro na votação.

3.1.3 Suporte

- Void::registerVoter(Voter::Eleitor,String::Token)

permite registar um novo eleitor.

A senha para poder votar online será enviado para o e-mail.

- Void::deleteVoter(Voter::Eleitor,String::Token)

Permite remover um eleitor.

Será lançada uma excepção se houve um erro na remoção.

- Int::updateVoter(Voter::Eleitor,String::Token)

permite alterar um eleitor.

Será lançada uma excepção se houve um erro na remoção.

- Voter::getVoter(Int::CodigoEleitor,String::Token)

retorna um eleitor dado o seu código de eleitor.

Será lançada uma excepção se o código de eleitor estiver errado.

- Voter::getVoter(Int::numeroEleitor,Int::freguesia,String::Token)

retorna um eleitor dado o seu número de eleitor e a freguesia.

Será lançada uma excepção se o número de eleitor e/ou a freguesia estiverem errados.

- Void::createCandidate(Candidate::candidato,String::Token)

Adiciona um novo candidato.

Será lançada uma exceção se o candidato já existir

- Void::updateCandidate(Candidate::candidato,String::Token)

Altera um candidato existente.

Será lançada uma exceção se o candidato não existir.

- Void::deleteCandidate(Candidate::candidato,String::Token)

Elimina um candidato existente.

Será lançada uma exceção se o candidato não existir.

3.1.4 Administração

- Void::createElection(Election::eleicao, String::Token)

cria uma nova eleição.

Será lançada uma exceção se a eleição já existir.

- Void::updateElection(Election::eleicao,String::Token)

altera uma eleição.

Será lançada uma exceção se a eleição não existir.

- Void::deleteElection(Election::eleicao, String::Token)

permite eliminar uma eleição.

Será lançada uma exceção se a eleição não existir.

- Election::getElection(Int::codigoEleicao)

devolve uma eleição dado o seu código.

Será lançada uma exceção se a eleição não existir.

- Void::createSupportUser(User::supportUser, String::Token)

cria um novo utilizador de suporte.

Será lançada uma exceção se o utilizador de suporte já existir.

- Void::updateSupportUser(User::supportUser, String::Token)

altera um utilizador de suporte existente.

Será lançada uma exceção se o utilizador de suporte não existir

- Void::deleteSupportUser(Int::supportUser, String::Token)

elimina um utilizador de suporte existente.

Será lançada uma exceção se o utilizador de suporte não existir

- User::getSupportUser(Int::freguesia, String::Token)

devolve o utilizador de suporte associado a uma dada freguesia.

Será lançada uma exceção se a freguesia não existir.

- Void::createParty(Party::partido)

Adiciona um novo partido.

Será lançada uma exceção se o partido já existir

- Void::updateParty(Party::partido)

Altera um partido existente.

Será lançada uma exceção se o partido não existir

- Void::deleteParty(Party::partido)

Elimina um partido existente.

Será lançada uma exceção se o partido não existir

3.1.5 Sessão

- String::loginVoter(Int::NumeroEleitor,Int::freguesia,String::password)

Inicia uma sessão de voto para o eleitor com o número e freguesia dados e devolve um token para utilizar em chamadas futuras.

Será lançada uma exceção se as credenciais não forem válidas.

- String::loginSupport(Int::freguesia,String::password)

Inicia uma sessão de voto para o eleitor com o número e freguesia dados e devolve um token para utilizar em chamadas futuras.

Será lançada uma excepção se as credenciais não forem válidas.

- String::loginAdmin(String::password)

Inicia uma sessão de voto para o eleitor com o número e freguesia dados e devolve um token para utilizar em chamadas futuras.

Será lançada uma excepção se as credenciais não forem válidas.

- Void::logout(String::Token)

Termina a sessão, invalidando o *token* dado. *Tokens* inválidos são ignorados.

Todos os métodos que recebem um *token* como parâmetro usam-no para autorização. Esse *token* é gerado aquando do *login* e dura até ao final da sessão. Há duas formas de terminar a sessão: explicitamente, fazendo *logout*, ou implicitamente, após um determinado período de tempo. Quando um *token* inválido é passado a um desses métodos é lançada uma excepção.

3.2. Aplicação de administração

Para se poder configurar e administrar o sistema existe uma aplicação (à parte do website) para tornar esse processo o mais simples e intuitivo possível – Aplicação de Administração do Péracles.

Assim como o website esta aplicação usa a API do webservice para interagir com este.

Aquando do início do programa o utilizador (administrador) é confrontado com um ecrã de login (este login é obrigatório e deve ser efectuado sempre que se inicia o programa mesmo que não seja a primeira vez que este é iniciado).

Depois do login efectuado o utilizador é apresentado com quatro listas – uma lista de todas as votações (eleição ou referendo), outra com todos os utilizadores com privilégios de suporte registados, outra com todos os candidatos registados e por fim uma de todos os partidos registados.

O administrador pode seleccionar qualquer elemento destas quatro listas para obter mais detalhes sobre elas. Também é possível modificar e apagar qualquer um destes elementos. Estas acções estão atribuídas a dois botões que poderão ser usados se uma votação ou utilizador de suporte estiver seleccionado.

O administrador pode também adicionar novos elementos a qualquer uma destas listas. Para isso apenas tem que seleccionar uma das listas e carregar no botão de Adicionar. Depois será apresentado um formulário com os todos os dados que devem ser preenchidos (o nome, tipo e quantidade de campos a preencher varia de acordo com a lista a que se está a adicionar os elementos). O formulário do botão de modificar é idêntico, apenas obriga a que um elemento de uma lista tenha sido seleccionado.

Sempre que for efectuada uma adição, remoção e actualização, o programa mostra um aviso a pedir para o utilizador confirmar a acção.

Sempre que o programa é encerrado é efectuado um logout automaticamente.

3.3. Website

O Website é um dos pontos fulcrais do Sistema de Votação Online. É aqui que os utilizadores podem exercer o direito de voto, efectuar queixas, consultar resultados, submeter resultados de mesas de voto tradicionais, gerir eleitores e gerar credencias de acesso. O acesso ao Website é feito via um browser, onde o utilizador tem a opção de fazer login. Dependendo do tipo de utilizador e após este ficar devidamente autenticado surgem novas opções.

Os tipos de utilizador são os seguintes: Eleitor, Suporte, Administrador e Consulta. Este último é atribuido a qualquer utilizador que não efectue Login. Quando um utilizador fica autenticado

não pode fazer consultas, pois como foi fundamentado na primeira parte deste projecto, os modos de utilizador são mutuamente exclusivos.

No Site, o Eleitor deve efectuar o Login para aceder as opções de Voto Online. No caso da autenticação falhar, surge uma mensagem de erro na qual é disponibilizado um email para contactar um Assistente de Suporte. No caso de um utilizador se autenticar, a opção para Votar Online fica disponível. Depois do Eleitor escolher esta opção, surge uma lista com as eleições activas que ficam disponíveis para escolha. O Eleitor selecciona a/as eleição/ões onde deseja participar e continua para a página seguinte. O site disponibiliza um boletim de voto e o Eleitor exerce o seu direito seleccionando o Candidato em que pretende votar. O voto é transmitido para a camada de dados de forma totalmente anónima, pois não fica registada qual foi a opção de voto do Eleitor. O voto fica então registado e é incrementado aos votos do Candidato escolhido. Não existe nenhuma ligação entre esse incremento e quem o propocionou. É registado que o Eleitor exerceu o seu direito nessa eleição e de seguida é apresentada uma mensagem de sucesso.

Outro caso possível é, o Eleitor, após estar autenticado, sair do website antes de efectuar qualquer voto. Ele pode voltar a qualquer altura durante a abertura das urnas, e concluir o seu escrutinio eleitoral.

No site, o Eleitor ainda tem outra opção disponível - fazer uma queixa. A mesma não tem que ser feita necessariamente no dia da propria eleição. A queixa é efectuada com o utilizador autenticado através de um formulário onde é necessário preencher a razão da mesma e a/as eleição/ões à/às qual/ais se refere. Esta é apresentada numa lista das últimas eleições. O tempo que um eleitor tem para apresentar uma queixa é definido pelo Administrador do sistema. A queixa é armazenada, juntamente com dados do Eleitor, data e hora, e a que eleições se refere.

O utilizador de Suporte geralmente é representado por um membro da junta de freguesia. Deve apenas existir um por freguesia. No site, após correcta autenticação, este utilizador tem as seguintes opções: registar, editar e listar eleitores, gerar senhas de acesso, gerir candidatos da freguesia e no final da contagem da eleição nas mesas tradicionais, submeter os resultados destas.

Para registar um novo eleitor o site disponibiliza um formulário que deverá ser preenchido com os dados do futuro Eleitor, nome e número do bilhete de identidade. Estes dados são transmitidos para a camada de negócio. Após a confirmação de que estes são suficientes e correctos, o novo Eleitor é criado na camada de dados, excepto se este já existir. Se tal

acontecer, é apresentada uma mensagem de erro, senão, a inserção é concluída, é atribuído um número de eleitor e aparece uma mensagem a confirmar que o novo Eleitor foi inserido com sucesso. O número de eleitor atribuído é o primeiro que não está a ser usado.

Quando autenticado, o utilizador de Suporte pode editar qualquer campo pertencente à tabela dos eleitores. Aquando do falecimento de um Eleitor, este deve ser removido pelo utilizador de Suporte. No sistema isto é implementado com um *soft-delete*, e o número de eleitor fica livre para ser usado por um novo Eleitor.

Outra opção disponível para um utilizador de Suporte autenticado é adicionar um Eleitor que já tenha registo noutra freguesia no caso deste alterar a sua residência. Isto é feito com uma alteração na tabela dos eleitores. Deve ser preenchido um formulário com o nome e número do bilhete de identidade e a freguesia é alterada automaticamente. Ao mesmo tempo é atribuído um número de eleitor.

Num período determinado pelo Administrador, os eleitores dirigem-se às suas respectivas juntas de freguesia para pedir a senha de acesso ao voto online. Esta senha é gerada no Website pelo utilizador de Suporte e é enviada para um email dado pelo eleitor no momento da requisição da sua respectiva senha. O utilizador de Suporte deve autenticar-se para gerar a senha. Este selecciona a opção de gerar credenciais e carrega uma página onde é obrigatório inserir o nome, o número do bilhete de identidade e o email de recepção da senha do eleitor em questão. O email deve ser inserido duas vezes para evitar erros. A necessidade de inserir o nome e o número de bilhete de identidade serve para reduzir os riscos de gerar uma senha para a pessoa errada. O Website comunica com o Webservice e este tenta gerar uma senha. Se os dados forem incorrectos, isto é, se um dos dados não existe, se o número do bilhete de identidade não coincide ou ainda se os emails não são iguais, o Website apresenta uma mensagem de erro e o utilizador de Suporte deve verificar os dados e repetir o processo. Se a senha for gerada com sucesso, é enviada para o endereço de email fornecido.

Os utilizadores de suporte têm a responsabilidade de adicionar, editar e remover os candidatos que participam na eleição da sua freguesia. O utilizador de Suporte deve usar a opção disponível no Website para efectuar essa operação. São inseridos os dados do candidato e a eleição a que eles se candidatam através de um formulário. Em caso de engano é possível editar a informação dos candidatos ou então remover um ou mais candidatos. O sistema regista automaticamente que estes se candidataram na freguesia que o utilizador de Suporte gere.

Por último, um utilizador de Suporte autenticado pode ainda pedir uma listagem dos eleitores da freguesia pela qual é responsável.

O Administrador tem uma aplicação própria para gerir vários parâmetros administrativos e não tem interacção com o Website.

As consultas são efectuadas no Website quando não existe nenhuma sessão iniciada. O utilizador pode visualizar resultados de duas maneiras: via um mapa ou simplesmente numa listagem com gráficos. O primeiro passo para consultar resultados de uma eleição é seleccionar uma eleição. O site disponibiliza uma ferramenta de pesquisa para localizar a eleição que o utilizador pretende consultar. Após seleccionar a eleição, o segundo passo é escolher como é que a deseja visualizar. As opções que o utilizador tem ao seu dispor são: visualizar resultados no mapa, listagem simplificada com gráficos ou comparação com outra eleição.

Na visualização de resultados no mapa, por defeito, é disponibilizado o mapa de Portugal. O utilizador pode focar uma determinada região se desejar visualizar os resultados da mesma. Os resultados que são apresentados por defeito dependem do tipo de eleição, mas representam sempre a opção ou partido vencedor.

O mapa contém as delimitações dos vários distritos. Quando o utilizador selecciona um distrito, surgem mais detalhes sobre a eleição nesse distrito. O utilizador pode optar por focar um concelho ou uma freguesia e obter os resultados detalhados da mesma forma. Nesta visualização também é possível apresentar os resultados da abstenção, colorindo com uma cor, as regiões que têm uma taxa de abstenção superior a um valor escolhido pelo utilizador.

O utilizador pode visualizar as regiões pintadas com a cor do partido vencedor, ou apenas pintar as regiões onde venceu um determinado partido. Por último, também é possível pintar as regiões onde o partido vencedor foi alterado em relação as últimas eleições do mesmo tipo.

A visualização de listagem simplificada com gráficos apresenta uma tabela com os resultados da eleição a nível nacional e um gráfico com as percentagens nacionais de cada partido. É possível visualizar estes resultados e gráficos para cada região. A qualquer momento é possível escolher a opção comparar. Seleccionando essa opção a consulta passa a ficar em modo de comparação. O utilizador escolhe uma eleição anterior do mesmo tipo e passa a visualizar a comparação dos dados entre as eleições.

3.4. Formato de dados LINX

LINX é um formato de dados usado para submeter votos resultantes de eleições tradicionais no sistema *Péricles*. De seguida apresenta-se a gramática EBNF desta linguagem:

```
DocumentoLinx : "LINX" Identificação Eleitores Votos
```

```
Identificação : "Freguesia" IdFreguesia Freguesia
```

```
Eleitores : "Eleitores" ListaNúmerosEleitor
```

```
Votos : "Votos" ListaVotos
```

```
ListaNúmerosEleitor : NúmeroEleitor*
```

```
ListaVotos : Voto*
```

```
Voto : Candidato NúmeroVotos
```

```
Candidato : String  
           | "Brancos"  
           | "Nulos"
```

```
IdFreguesia : Id
```

```
Freguesia : String
```

```
NúmeroEleitor : Número
```

```
NúmeroVotos : Número
```

Os elementos são separados por sequências de *whitespace* de qualquer comprimento.

Aqui temos um pequeno exemplo de um documento LINX:

LINX	
Freguesia	123456 Prozelos
Eleitores	
1	
2	
3	
4	
5	
234	
6	
10	
43	
Votos	
PS	1
PSD	5
Branco	3
Nulos	1

3.5. Tecnologias e Ferramentas utilizadas

Durante esta etapa do projecto utilizámos várias ferramentas e tecnologias. Numa primeira fase utilizamos papel e lápis para esboçar alguns diagramas. Depois utilizámos o *Visio* 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*.

Também decidimos as ferramentas e tecnologias que vão ser usadas na última etapa do projecto. A plataforma *.NET* (linguagem *C#* e *ASP.NET*) e o *Google Maps* são duas tecnologias requeridas pelo projecto e por isso serão utilizadas. Para além destas vamos utilizar o *Visual Studio* como ambiente de desenvolvimento, visto ser o *IDE standard de facto* para *.NET*. Também vamos utilizar o sistema de controlo de versões *Subversion*.

4. Conclusão

Com a segunda fase (Especificação) conseguimos organizar as nossas ideias e objectivos para o projecto, explicados na primeira fase (Fundamentação). Com os diagramas de UML foi-nos possível estruturar e modelar o projecto de uma forma sólida de modo a identificar certos problemas futuros não muito óbvios e corrigi-los atempadamente. Inicialmente, tivemos alguns problemas com certas ferramentas mas achamos que todos eles foram ultrapassados com a troca por outras ferramentas ou com um uso mais prolongado das mesmas. Somos da opinião que fizemos uma boa especificação, o que fará com que a terceira fase (Construção) corra sem surpresas e que tornará o nosso trabalho o mais fiel possível à ideia inicial.

Bibliografia

[B1] Martin Fowler, UML Distilled: A Brief Guide to the Standard Object Modeling Language, 3rd edition, Addison Wesley, 2003

Referências WWW

[01] **pt.wikipedia.org**

Enciclopédia gratuita online. Fonte vasta de informação onde é possível pesquisar vários assuntos e encontrar referências e apontadores adicionais.

[02] **www.msdnaa.net**

Loja virtual onde é possível adquirir software da microsoft a custo zero para alunos da universidade do minho.

[03] **<http://www.di.uminho.pt/~omb/leili4.html>**

Site da disciplina de LI4, onde é possível obter detalhes sobre os projecto da disciplina e informação relativa aos seminarios da cadeira.

Lista de Siglas e Acrónimos

LI4 Laboratórios de Informática IV

UML Unified Modeling Language

BD Base de Dados

SOAP Simple Object Application Protocol

WSDL Web Service Description Language

SQL Structured Query Language

ASP Active Server Pages

EBNF Extended Backus-Naür Form

LINX LINX is Not XML

Anexos

I. Use Cases

Utilizador Autenticado

Iniciar Sessão

Pre-Condições:

User não tem sessão iniciada

Garantias:

A sessão está iniciada com as credenciais fornecidas

Trigger:

No site, User selecciona opção de Login

Cenário de Sucesso Principal:

1. User insere username e password
2. Sistem autentica o User
3. Sistema regista que o User foi autenticado
4. Site informa que o User esta autenticado

Extensões

2a - O Sistema falha a autenticação do User

- .1 O Site informa que o User não foi autenticado
- .2 O User volta a inserir username e password ou desiste

Terminar Sessão

Pre-condição:

A sessão esta iniciada

Garantias:

Não existe nenhuma sessão iniciada

Triggers:

User selecciona opção de Logout

Cenario de Sucesso Principal:

1. Sistema "desliga" User
2. Sistema regista o Logout de User
3. Site informa User que foi desligado

Consulta

Consultar Resultados

Cenário de Sucesso Principal:

1. Utilizador escolhe a eleição na qual pretende visualizar os resultados
2. Sistema mostra ao utilizador os resultados da eleição que escolheu

Eleitor

Votar

Cenário de Sucesso Principal:

1. <<include>> Login
2. Sistema mostra ao utilizador a lista de eleições disponíveis para votar
3. Utilizador escolhe em que eleição pretende votar
4. Utilizador vota na eleição que escolheu
5. O voto é registado na base de dados
6. Sistema confirma sucesso da votação
7. <<include>> Logout

Cenário de Sucesso Alternativo:

3. Utilizador opta por não votar em nenhuma eleição
4. <<include>> Logout

Extensões:

4a - Sistema de voto indisponível

- .1 Sistema informa utilizador sobre a indisponibilidade do sistema de voto
- .2 Termina o Use Case

Apresentar Queixa

Cenário de Sucesso Principal:

- 1. Utilizador apresenta queixa referente ao sistema de votação
- 2. Sistema envia a queixa do utilizador para a base de dados

Administrador

Tratar dados invalidos

Garantias:

O estado do sistema não sofre alterações.

Trigger:

Foram introduzidos dados inválidos durante uma outra operação.

Cenário de sucesso principal:

1. O sistema notifica o utilizador sobre a razão pela qual os dados são inválidos.
2. O utilizador corrige os dados que introduziu.
3. O sistema valida os novos dados.
4. A operação que foi interrompida prossegue.

Cenário de sucesso alternativo:

2. O utilizador desiste da operação que foi interrompida.
3. A operação que foi interrompida é cancelada.

Extensões:

- 3a. Os dados não são válidos.
 - .1 Este processo recomeça.

Gerir Candidatos

Adicionar Candidato

Cenário de Sucesso Principal:

1. User insere dados do candidato
2. Sistema pede confirmação
3. User Confirma adição
4. Sistema Adiciona candidato
5. Sistema Confirma Adição

Cenário de Sucesso Alterativo:

3. User Cancela acção
4. Voltar ao passo 1

Cenário de Sucesso Alterativo1:

4. candidato já existe
5. Volta ao passo 1

Extensões:

4a - Sistema: Falha de comunicação

4b - Sistema: Dados insuficientes/Campos obrigatórios não preenchidos ou mal preenchidos

Visualizar Candidato

Pre Condições:

Listagem de candidatos efectuada

Cenário de Sucesso Principall:

1. User Selecciona um candidato
2. Sistema Mostra informação do candidato

Extensões:

2a - Sistema: Falha de comunicação

Alterar Candidato

Pre Condição:

Listagem de candidatos efectuada

Cenário de Sucesso Principal:

1. User Selecciona candidato
2. User insere novos dados do candidato
3. Sistema pede confirmação
4. User Confirma alteração
5. Sistema Altera dados do candidato
6. Sistema Confirma Alteração

Cenário de Sucesso Alternativo:

4. User Cancela acção
5. Voltar passo 2

Extensões:

5a - Sistema: Falha de comunicação

5b - Sistema: Dados insuficientes/Campos obrigatórios não preenchidos ou mal preenchidos

Remover Candidato

Pre Condição:

Listagem de candidatos efectuada

Cenário de Sucesso Principal:

1. User Selecciona um candidato
2. Sistema pede confirmação
3. User Confirma remoção
4. Sistema Apaga candidato
5. Sistema Confirma remoção

Cenário de Sucesso Alternativo:

3. User: Cancela acção de remoção
4. Voltar passo 1

Extensões:

4a - Sistema: Falha de comunicação

Listar Candidato

Cenário de Sucesso Principal:

1. Sistema: lista todos os candidatos

Extensões:

- 1a - Sistema: Falha de comunicação

Gerir Partidos

Adicionar Partido

Cenário de Sucesso Principal:

1. User insere dados do partido
2. Sistema pede confirmação
3. User Confirma adição
4. Sistema Adiciona partido
5. Sistema Confirma Adição

Cenário de Sucesso Alternativo:

3. User Cancela acção
4. Voltar ao passo 1

Cenário de Sucesso Alternativo1:

4. partido já existe

5. Volta ao passo 1

Extensões:

4a - Sistema: Falha de comunicação

4b - Sistema: Dados insuficientes/Campos obrigatórios não preenchidos ou mal preenchidos

Visualizar Partido

Pre Condições:

Listagem de partidos efectuada

Cenário de Sucesso Principal:

1. User Selecciona um partido
2. Sistema Mostra informação do partido

Extensões:

2a - Sistema: Falha de comunicação

Alterar Partido

Pre Condição:

Listagem de partidos efectuada

Cenário de Sucesso Principal:

1. User Selecciona partido

2. User insere novos dados de partido
3. Sistema pede confirmação
4. User Confirma alteração
5. Sistema Altera dados do partido
6. Sistema Confirma Alteração

Remover Partido

Pre Condição:

Listagem de partidos efectuada

Cenário de Sucesso Principal:

1. User Selecciona um partido
2. Sistema pede confirmação
3. User Confirma remoção
4. Sistema Apaga partido
5. Sistema Confirma remoção

Cenário de Sucesso Alternativo:

3. User: Cancela acção de remoção
4. Voltar passo 1

Extensões:

4a - Sistema: Falha de comunicação

Listar Partidos

Cenário de Sucesso Principal:

1. Sistema: lista todos os partidos

Extensões:

1a - Sistema: Falha de comunicação

Gerir Eleições

Definir Eleição

Pré-condições:

Ter sessão iniciada como administrador.

Garantias:

Existe uma nova eleição no sistema com os dados fornecidos.

Trigger:

O administrador pediu para criar uma eleição.

Cenário de sucesso principal:

1. O sistema pede os dados da nova eleição.

2. O administrador introduz os dados da nova eleição.

3. O sistema valida os dados.

4. O sistema cria a nova eleição.

5. O sistema regista a criação de uma eleição.

6. O sistema confirma a criação de uma eleição.

Extensões:

6a. Os dados não são válidos.

1. Tratar dados inválidos

Alterar Eleição

Pré-condições:

Ter sessão iniciada como administrador.

Garantias:

A eleição escolhida tem novos dados.

Trigger:

O administrador pediu para alterar uma eleição.

Cenário de sucesso principal:

1. "Listar eleições" (o sistema lista apenas aquelas que ainda não decorreram).

2. O administrador escolhe uma eleição.
3. O sistema apresenta os dados da eleição escolhida para modificação.
4. O administrador altera os dados conforme necessário.
5. O sistema valida os dados.
6. O sistema efectua a alteração.
7. O sistema regista a alteração.
8. O sistema confirma a alteração.

Extensões:

5a. Os dados não são válidos.

1. Tratar dados inválidos.

Listar Eleições

Pré-condições:

Ter sessão iniciada como administrador.

Garantias:

O estado do sistema não sofre alterações.

Triggers:

- O administrador pediu para listar eleições.
- Uma outra operação requisitou a listagem de eleições.

Cenário de sucesso principal

1. O administrador escolhe a forma e os critérios de listagem.
2. O sistema lista as eleições segundo a forma e os critérios escolhidos.

Gerir Utilizadores de Suporte

Criar Utilizador de Suporte

Pré-condições:

Ter sessão iniciada como administrador.

Garantias:

Existe mais um utilizador de suporte, com os dados que foram introduzidos.

Trigger:

O administrador pediu para criar um utilizador de suporte.

Cenário de sucesso principal:

1. O sistema pede os dados do novo utilizador.
2. O administrador introduz os dados do novo utilizador.
3. O sistema valida os dados introduzidos.
4. O sistema cria um novo utilizador.
5. O sistema regista a criação do novo utilizador.

6. O sistema confirma a criação do novo utilizador.
7. O sistema pede o meio de envio dos dados de acesso do novo utilizador.
8. O administrador escolhe o meio de envio.
9. O sistema envia os dados de acesso para o novo utilizador, ou imprime-os para envio por meios não electrónicos.
10. O sistema regista o envio dos dados de acesso, ou a sua impressão.

Extensões:

3a. Os dados não são válidos.

1. "Tratar dados inválidos"

Listar Utilizador de Suporte

Pré-condições:

Ter sessão iniciada como administrador.

Garantias:

O estado do sistema não sofre alterações.

Triggers:

- O administrador pediu para listar os utilizadores de suporte.
- Uma outra operação requisitou a listagem dos utilizadores de suporte.

Cenário de sucesso principal:

1. O administrador escolhe a forma e os critérios de listagem.
2. O sistema lista os utilizadores de suporte segundo a forma e os critérios escolhidos.

Modificar Utilizador de Suporte

Pré-condições:

Ter sessão iniciada como administrador.

Garantias:

O utilizador de suporte escolhido tem novos dados.

Trigger:

O administrador pediu para modificar um utilizador de suporte.

Cenário de sucesso principal:

1. "Listar utilizadores de suporte"
2. O administrador escolhe o utilizador que quer modificar.
3. O sistema apresenta os dados do utilizador escolhido para modificação.
4. O administrador modifica os dados do utilizador conforme necessário.
5. O sistema valida as alterações.
6. O sistema efectua as alterações.
7. O sistema regista as alterações.

8. O sistema confirma as alterações ao administrador.

Extensões:

5a. As alterações não são válidas.

1. "Tratar dados inválidos.

Remover Utilizador de Suporte

Pré-condições:

Ter sessão iniciada como administrador.

Garantias:

O utilizador de suporte escolhido não existe.

Trigger:

O administrador pede a remoção de um utilizador de suporte.

Cenário de sucesso principal:

1. "Listar utilizadores de suporte"
2. administrador escolhe o utilizador que quer remover.
3. O sistema apresenta os dados do utilizador escolhido.
4. O administrador confirma a remoção.
5. O sistema remove o utilizador escolhido.

6. O sistema regista a remoção do utilizador escolhido
7. O sistema confirma a remoção do utilizador escolhido.

Suporte

Submeter Votos Tradicionais

Cenário de Sucesso Principal:

1. User Preenche os campos
2. Sistema Pede confirmação
3. User Confirma submissão
4. Sistema Envia os dados
5. Sistema Confirma Submissão

Cenário de Sucesso Alternativo:

3. User Cancela submissão
4. Voltar ao passo 1

Extensões:

- 4a - Sistema Dados insuficientes/Campos obrigatórios não preenchidos ou mal preenchidos
- 4b - Sistema Falha na comunicação
- 5a - Sistema Dados Incorrectos

Gerar Credenciais de Acesso

Normal:

1 - User: Insere nome, numero do BI e email de recepção

2 - Sistema: Envia password para email de recepção

Excepção1:

1

2 - Sistema: nome e/ou numero de leitor incorrectos

Excepção2:

1

2 - Sistema: Falha de comunicação

Gerir Eleitores

Adicionar Eleitor

Cenário de Sucesso Principal:

1. User insere dados de eleitor

2. Sistema pede confirmação

3. User Confirma adição

4. Sistema Adiciona eleitor

5. Sistema Confirma Adição

Cenário de Sucesso Alterativo:

3. User Cancela acção

4. Voltar ao passo 1

Cenário de Sucesso Alternativo1:

4. Utilizador já existe

5. Volta ao passo 1

Extensões:

4a - Sistema: Falha de comunicação

4b - Sistema: Dados insuficientes/Campos obrigatórios não preenchidos ou mal preenchidos

Visualizar Eleitor

Pre Condições:

Listagem de eleitores efectuada

Cenário de Sucesso Principal:

1. User Selecciona um eleitor

2. Sistema Mostra informação do eleitor

Extensões:

2a - Sistema: Falha de comunicação

Alterar Eleitor

Pre Condição:

Listagem de eleitores efectuada

Cenário de Sucesso Principal:

1. User Selecciona utilizador
2. User insere novos dados de eleitor
3. Sistema pede confirmação
4. User Confirma alteração
5. Sistema Altera dados do eleitor
6. Sistema Confirma Alteração

Cenário de Sucesso Alternativo:

4. User Cancela acção
5. Voltar passo 2

Extensões:

5a - Sistema: Falha de comunicação

5b - Sistema: Dados insuficientes/Campos obrigatórios não preenchidos ou mal preenchidos

Remover Eleitor

Pre Condição:

Listagem de eleitores efectuada

Cenário de Sucessi Principal:

1. User Selecciona um eleitor
2. Sistema pede confirmação
3. User Confirma remoção
4. Sistema Apaga eleitor
5. Sistema Confirma remoção

Cenário de Sucesso Alternativo:

3. User: Cancela acção de remoção
4. Voltar passo 1

Extensões:

4a - Sistema: Falha de comunicação

Listar Eleitores

Cenário de Sucesso Principall:

1. Sistema: lista todos os eleitores da freguesia

Extensões:

- 1a - Sistema: Falha de comunicação

Gerir Candidatos

Adicionar Candidato

Cenário de Sucesso Principal:

1. User insere dados do candidato
2. Sistema pede confirmação
3. User Confirma adição
4. Sistema Adiciona candidato
5. Sistema Confirma Adição

Cenário de Sucesso Alterantivo:

3. User Cancela acção
4. Voltar ao passo 1

Cenário de Sucesso Alterantivo1:

4. candidato já existe

5. Volta ao passo 1

Extensões:

4a - Sistema: Falha de comunicação

4b - Sistema: Dados insuficientes/Campos obrigatórios não preenchidos ou mal preenchidos~

Visualizar Candidato

Pre Condições:

Listagem de candidatos efectuada

Cenário de Sucesso Principal:

1. User Selecciona um candidato
2. Sistema Mostra informação do candidato

Extensões:

2a - Sistema: Falha de comunicação

Alterar Candidato

Pre Condição:

Listagem de candidatos efectuada

Cenário de Sucesso Principal:

1. User Selecciona candidato

2. User insere novos dados do candidato
3. Sistema pede confirmação
4. User Confirma alteração
5. Sistema Altera dados do candidato
6. Sistema Confirma Alteração

Cenário de Sucesso Alternativo:

4. User Cancela acção
5. Voltar passo 2

Extensões:

5a - Sistema: Falha de comunicação

5b - Sistema: Dados insuficientes/Campos obrigatórios não preenchidos ou mal preenchidos

Remover Candidato

Pre Condição:

Listagem de candidatos efectuada

Cenário de Sucesso Principal:

1. User Selecciona um candidato
2. Sistema pede confirmação
3. User Confirma remoção

4. Sistema Apaga candidato
5. Sistema Confirma remoção

Cenário de Sucesso Alternativo:

3. User: Cancela acção de remoção
4. Voltar passo 1

Extensões:

4a - Sistema: Falha de comunicação

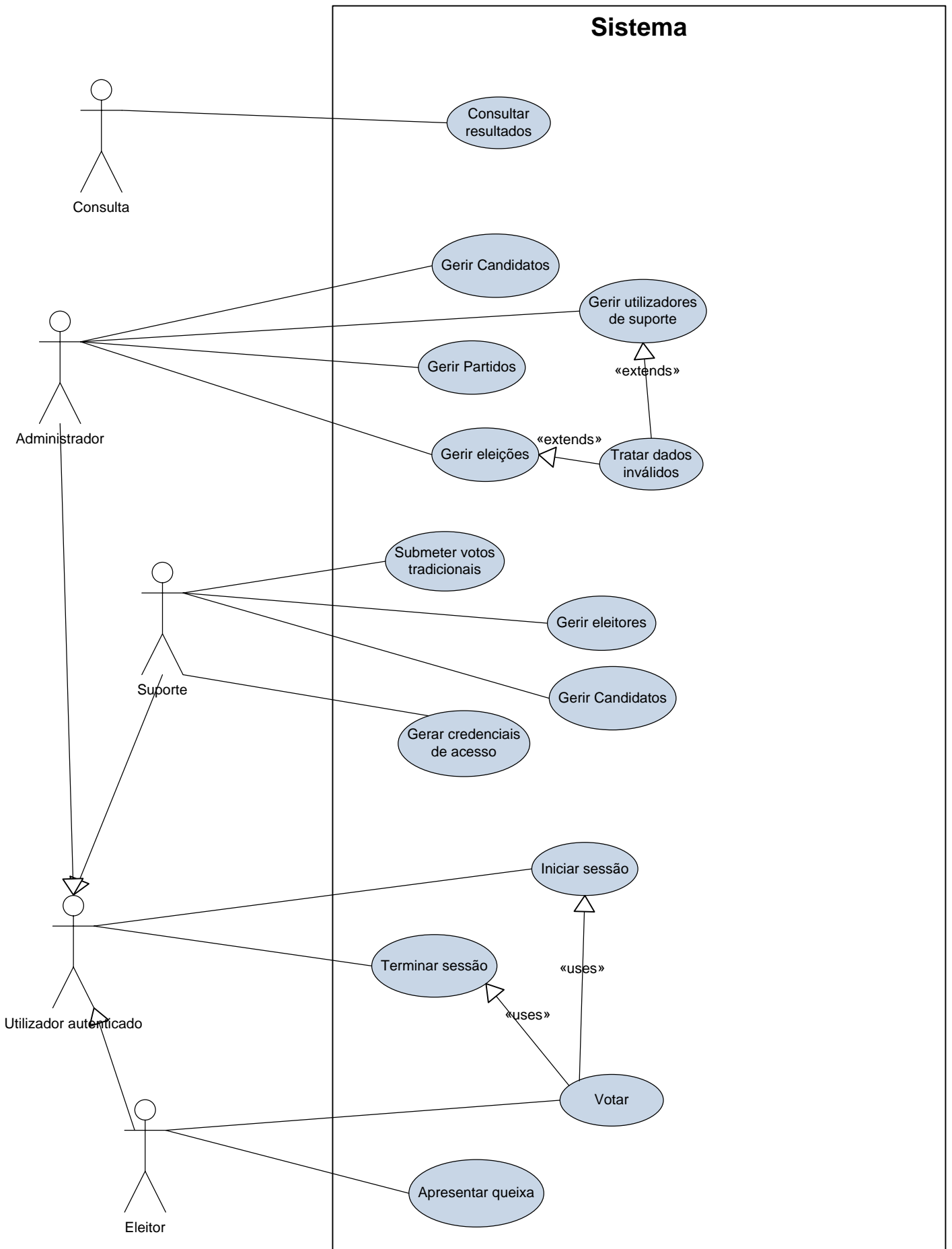
Listar Candidatos

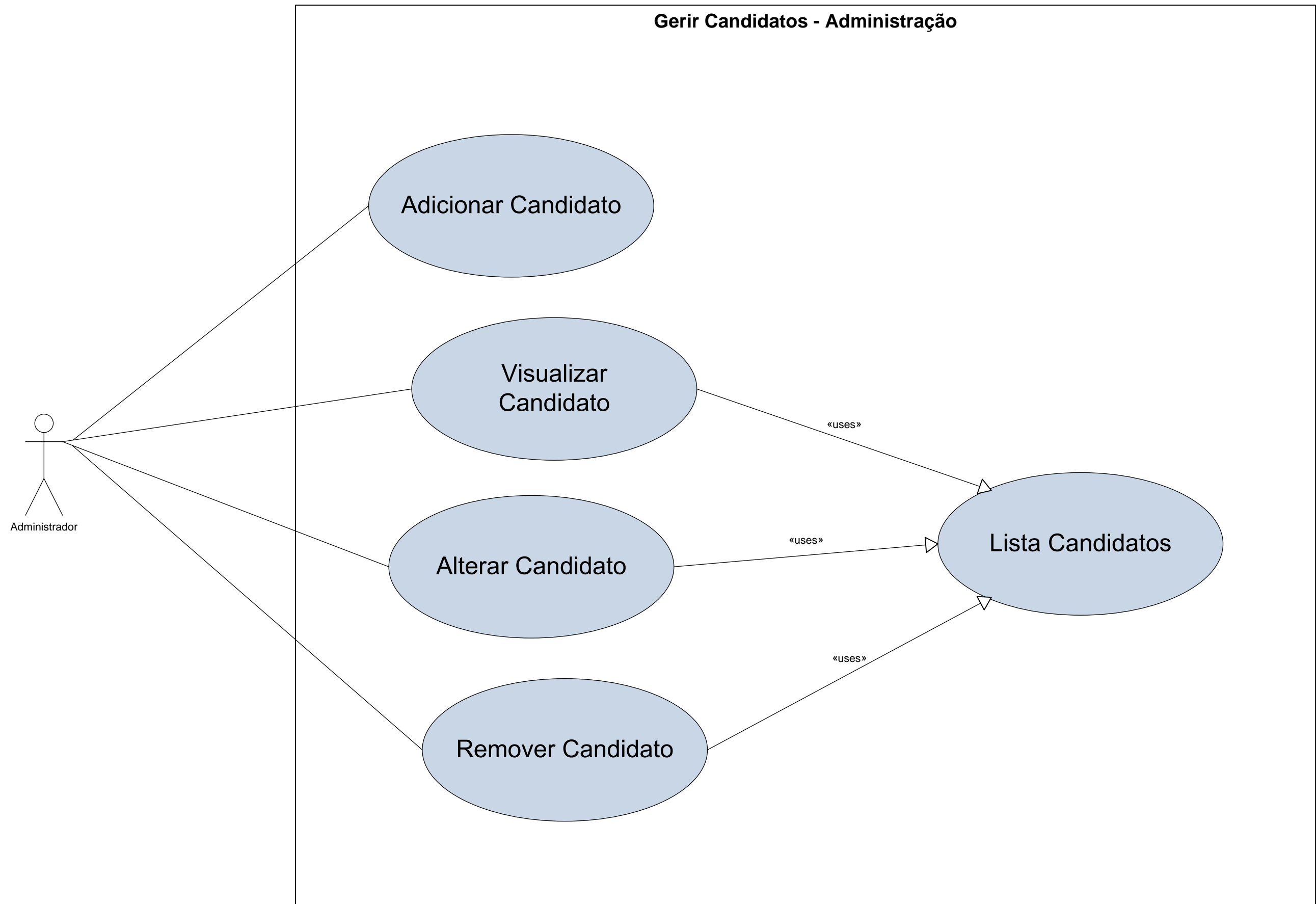
Cenário de Sucesso Principall:

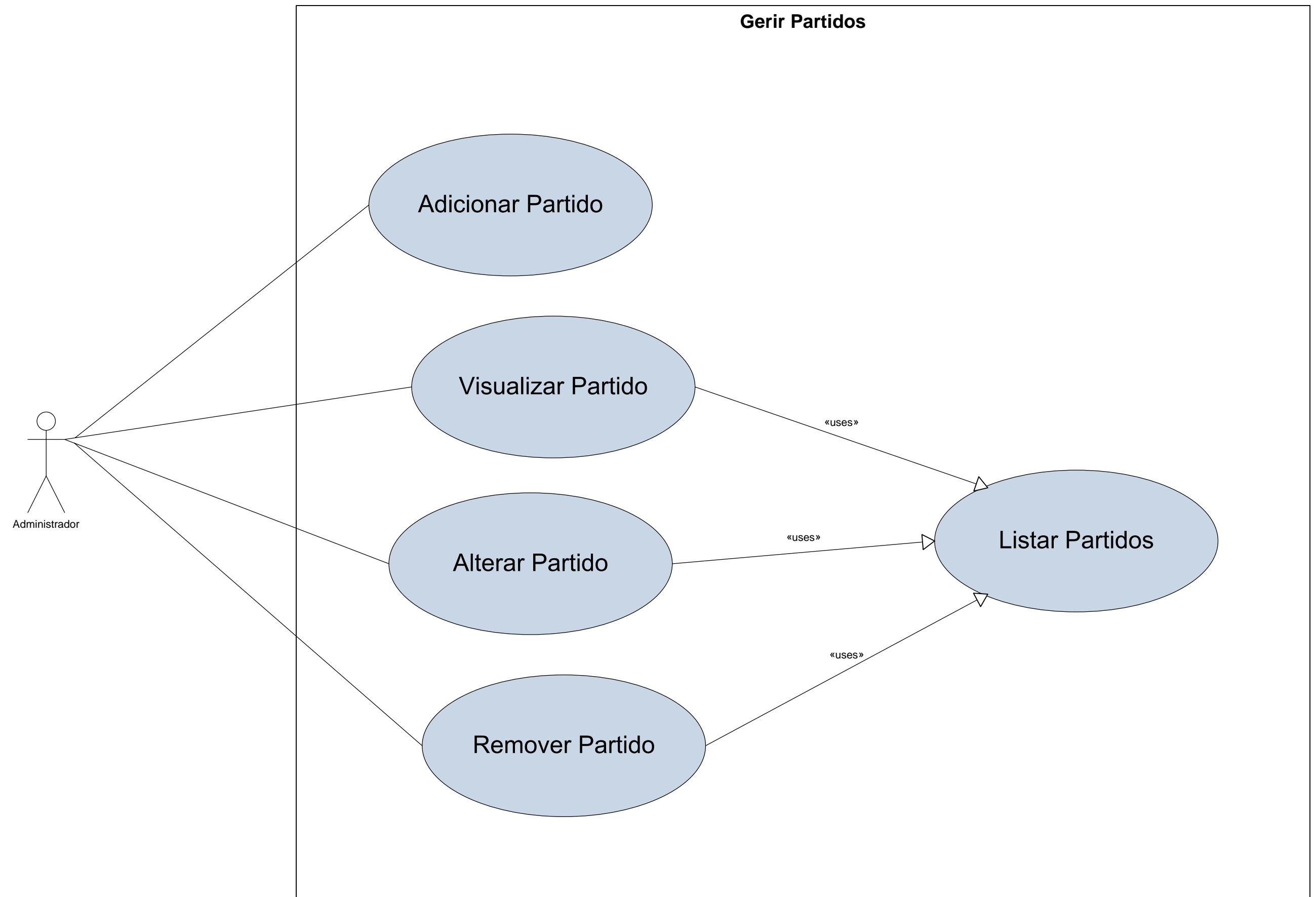
1. Sistema: lista todos os candidatos da freguesia

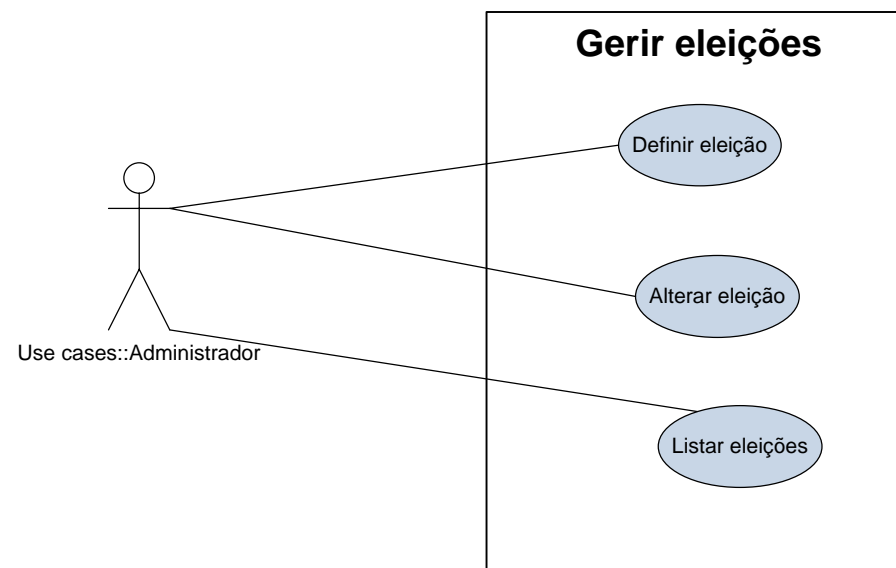
Extensões:

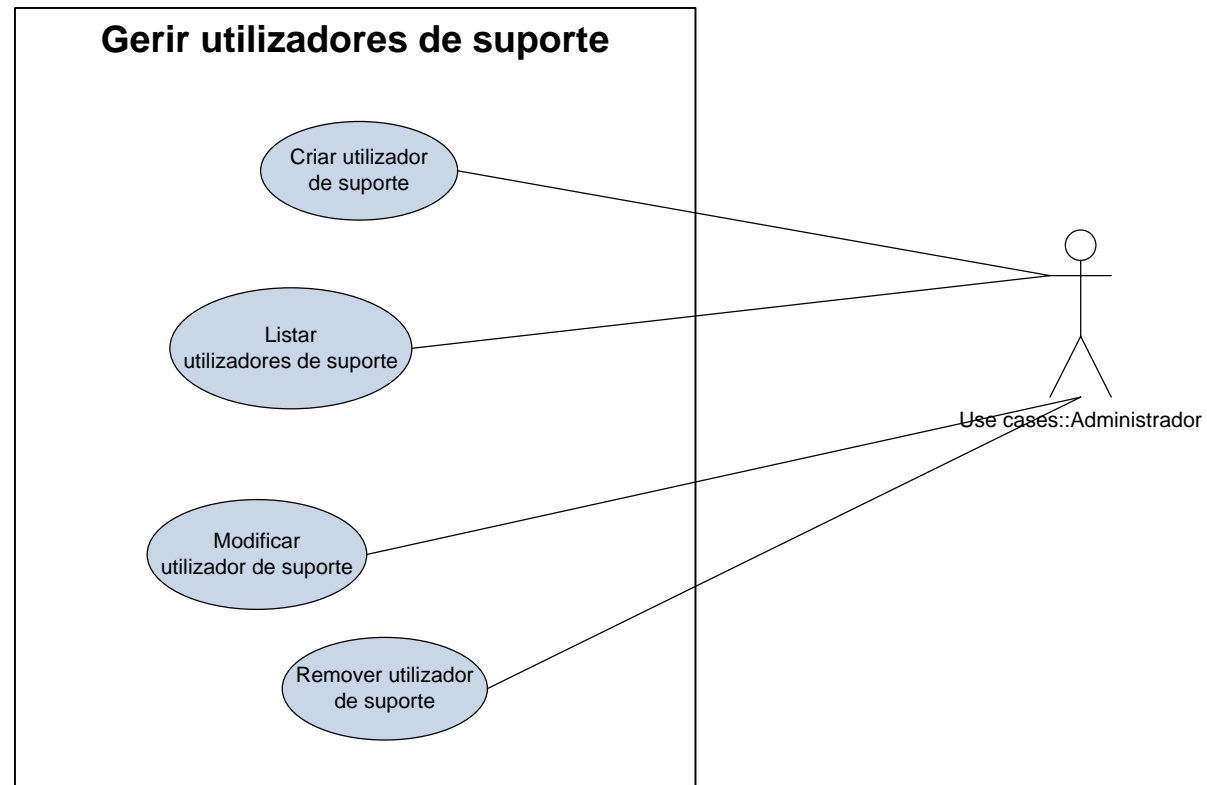
1a - Sistema: Falha de comunicação

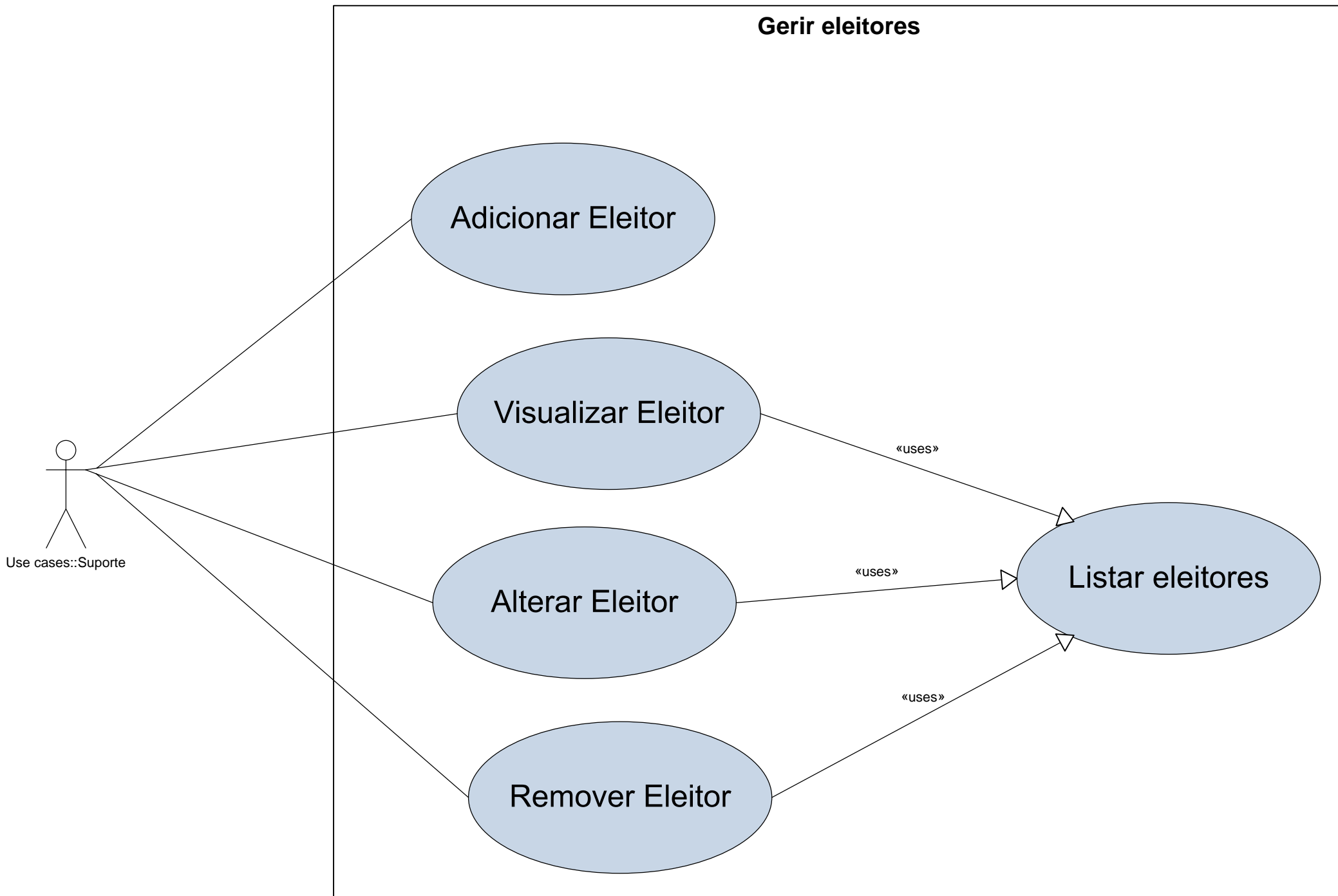


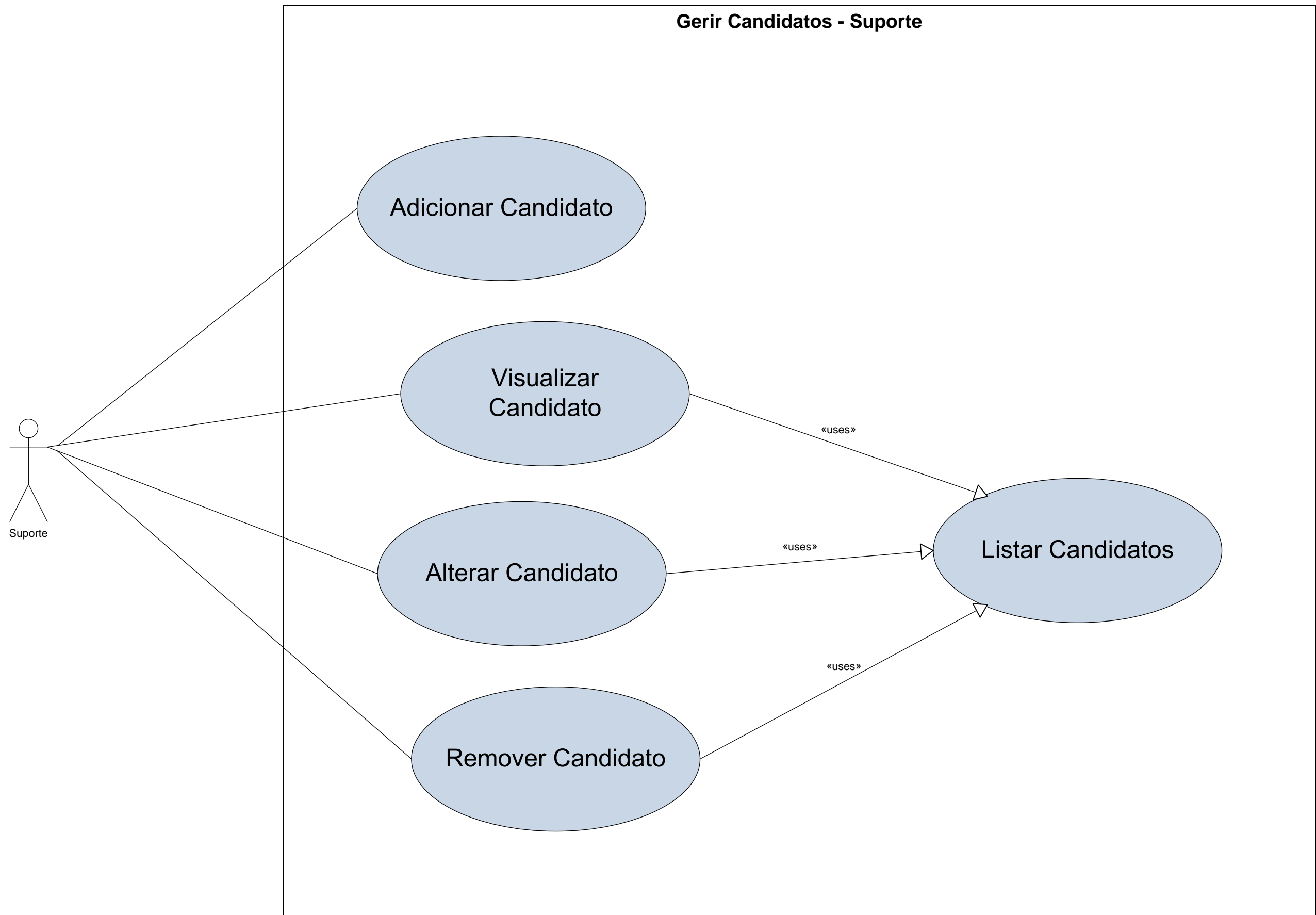






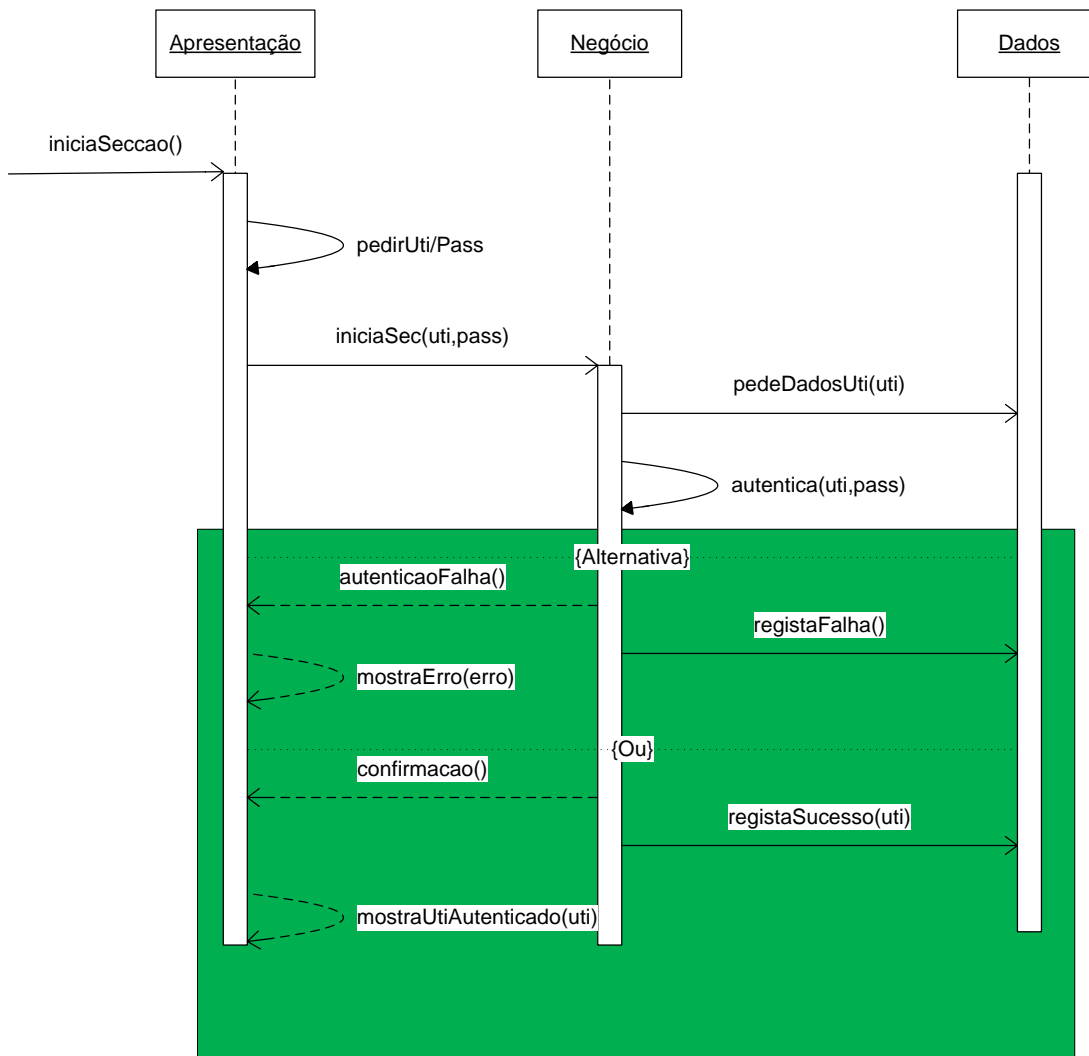




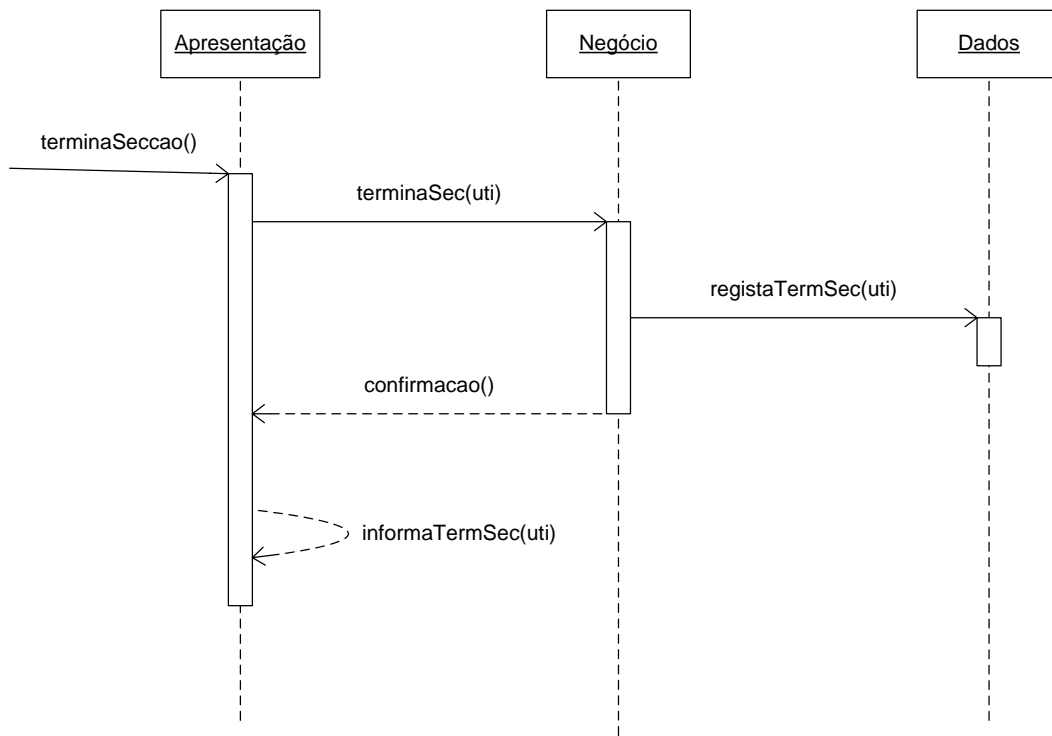


II. Diagramas de Sequência

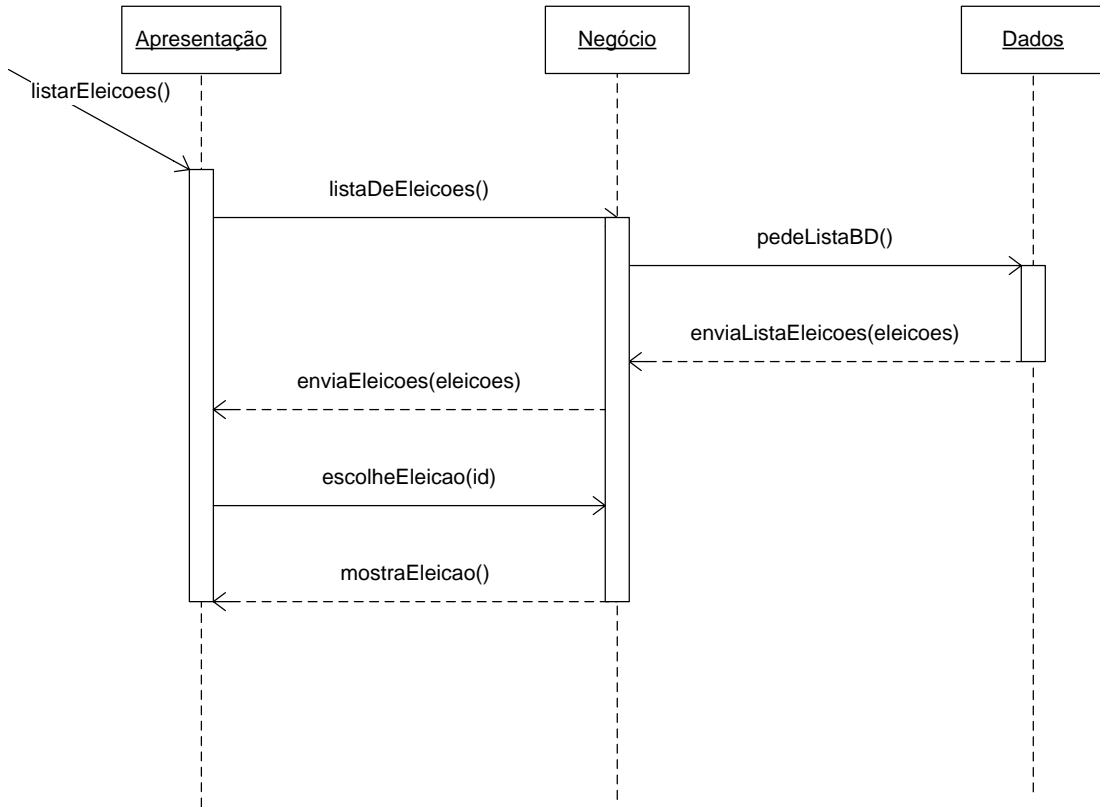
Utilizador Autenticado – Iniciar Sessão



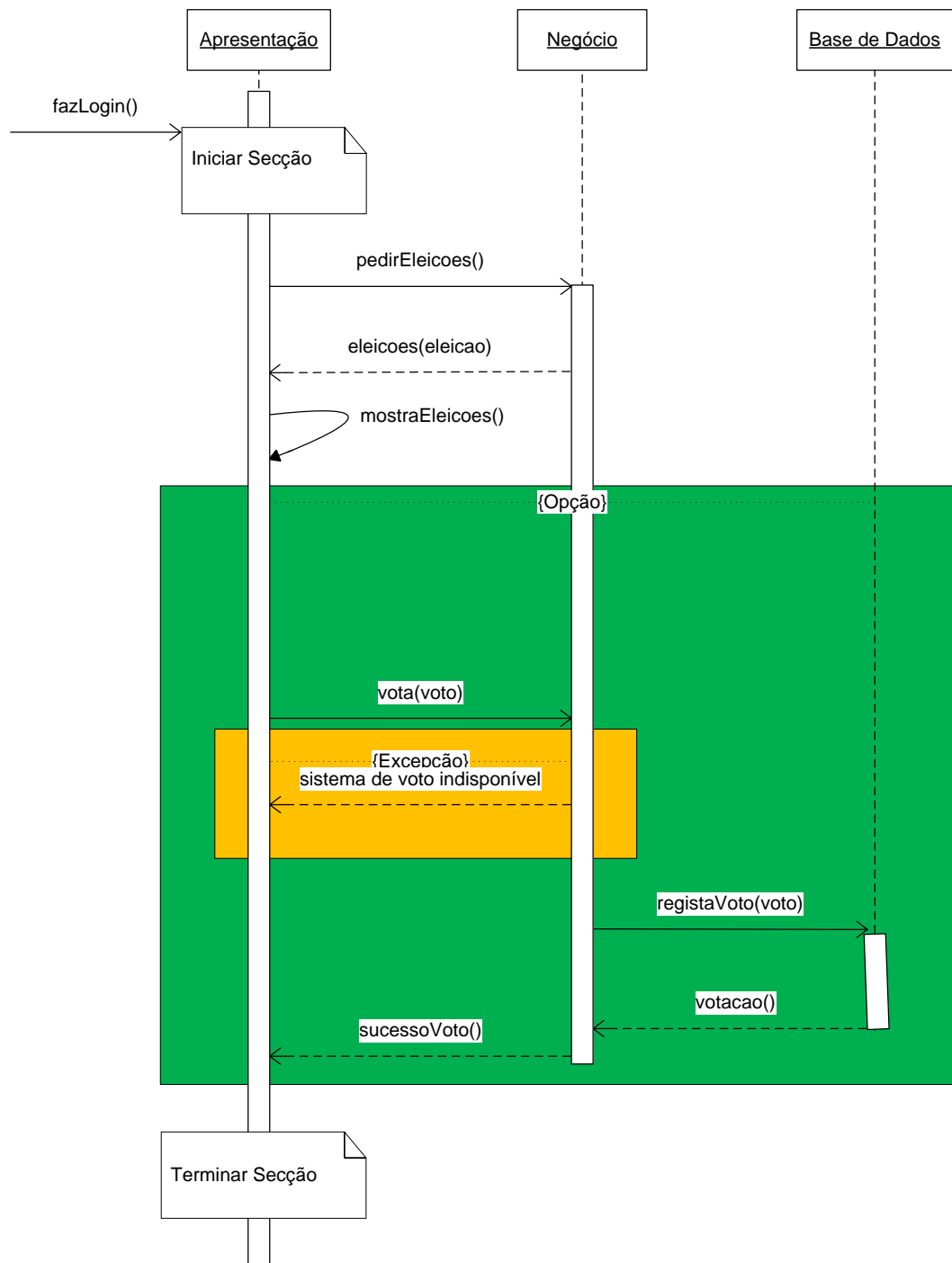
Utilizador Autenticado – Terminar Sessão



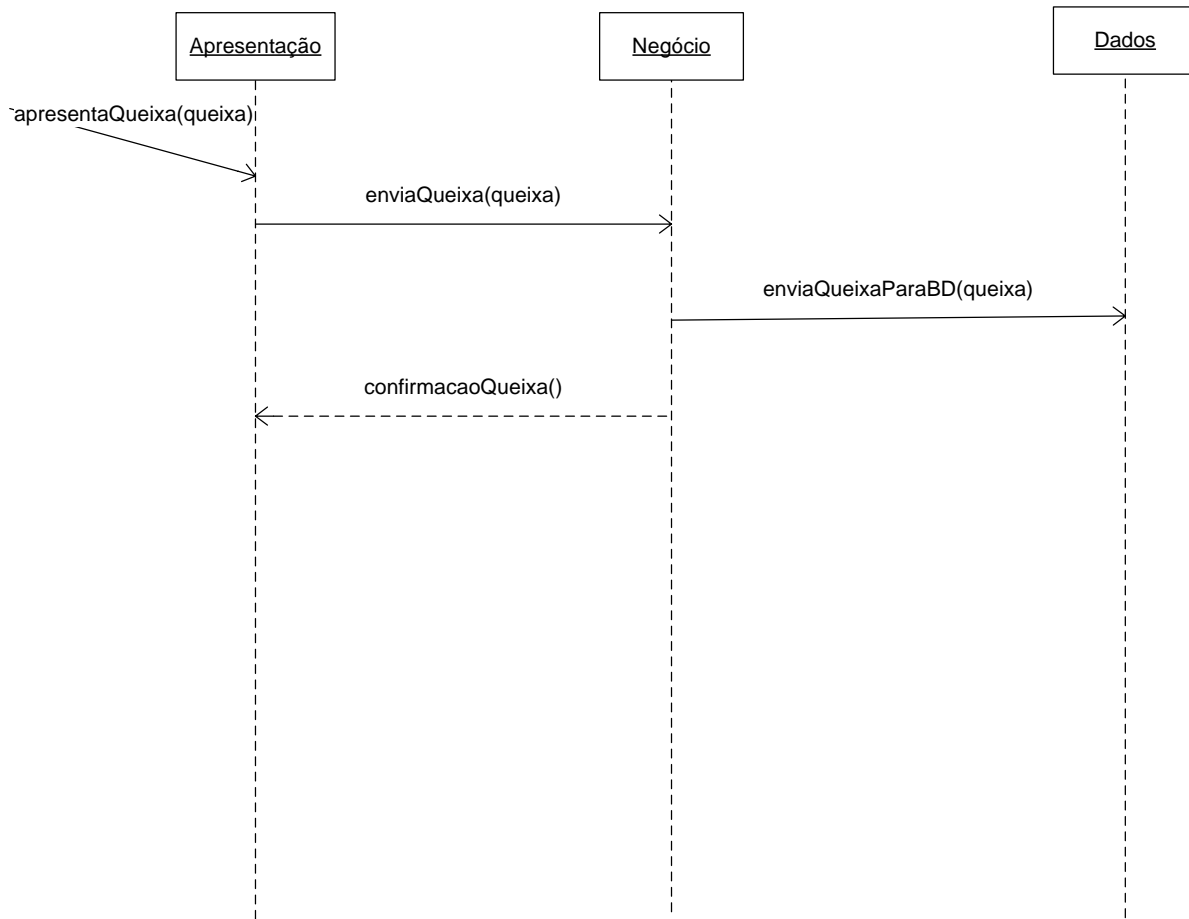
Consulta – Consultar Resultados



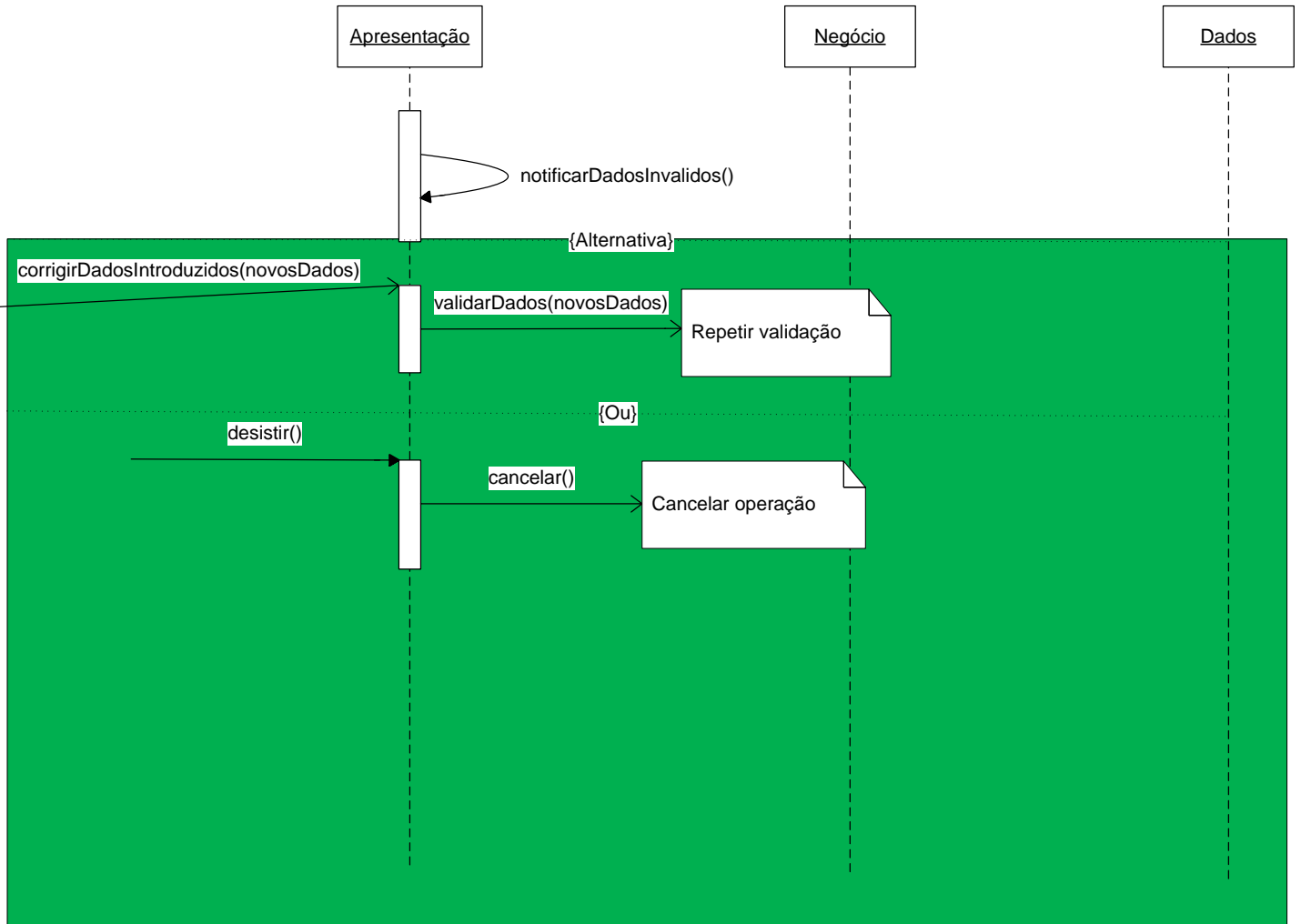
Eleitor - Votar



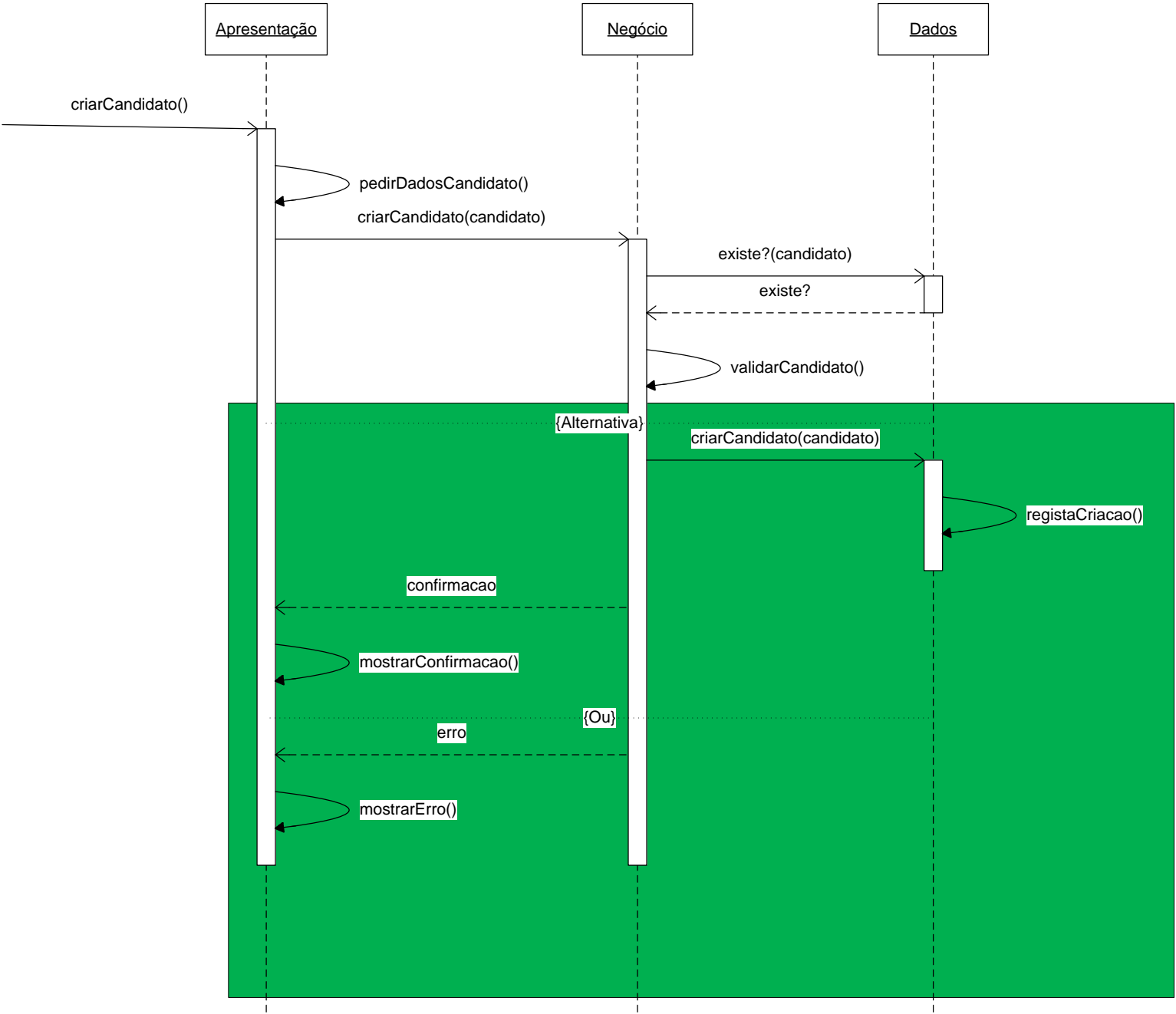
Eleitor – Apresentar Queixa



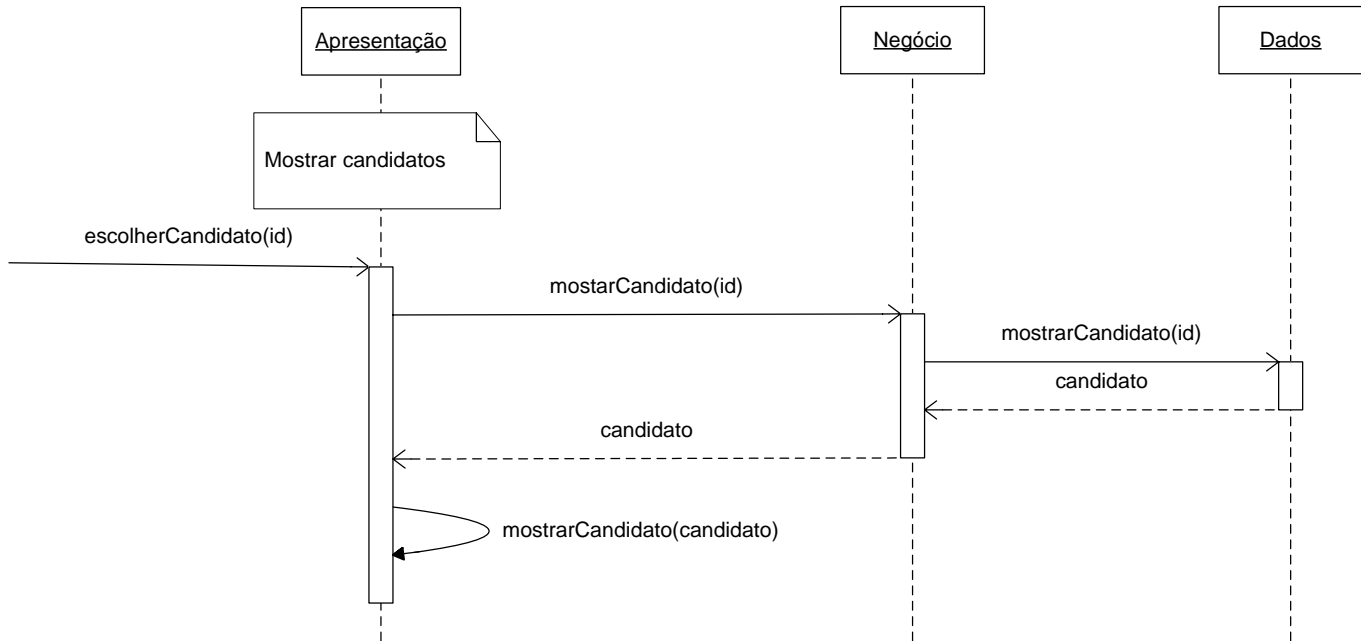
Administrador – Tratar dados inválidos



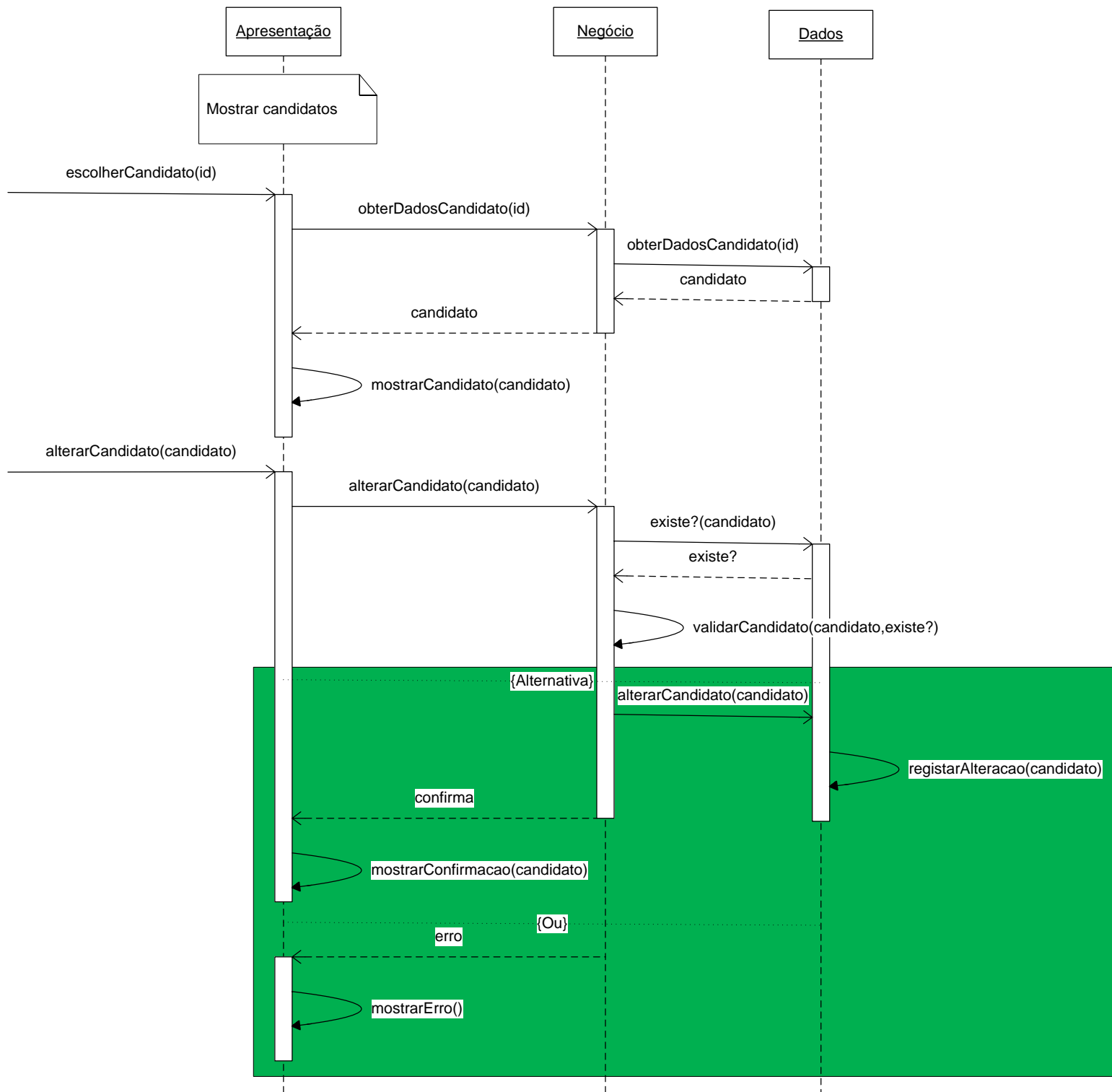
Administrador – Adicionar Candidato



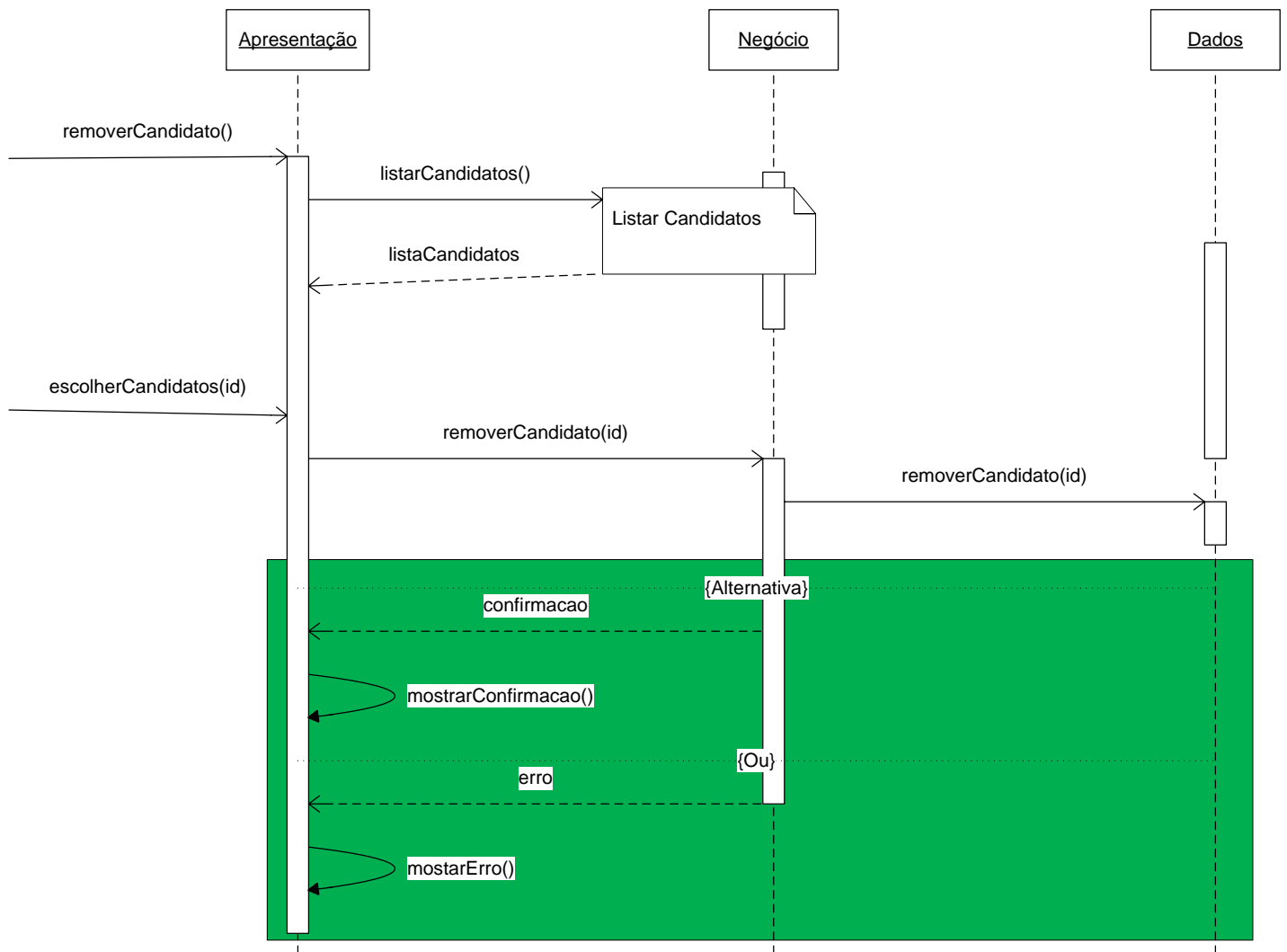
Administrador – Visualizar candidato



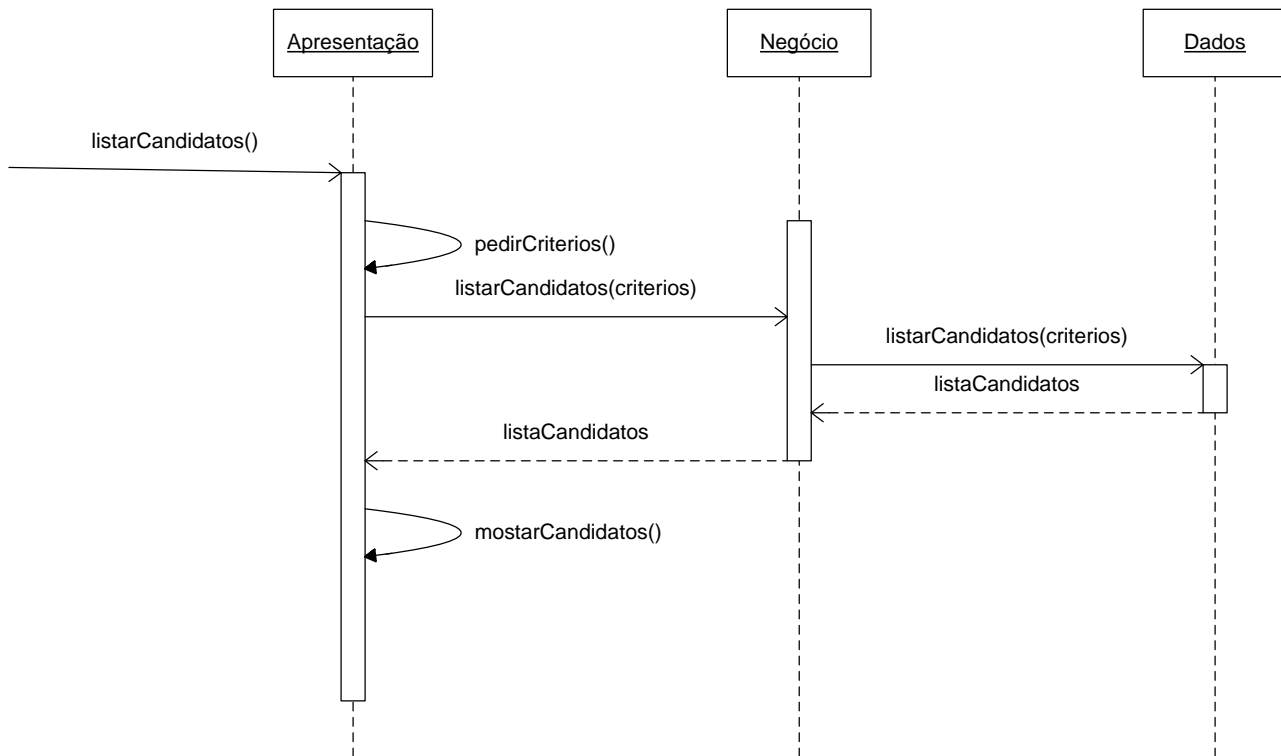
Administrador – Alterar Candidato



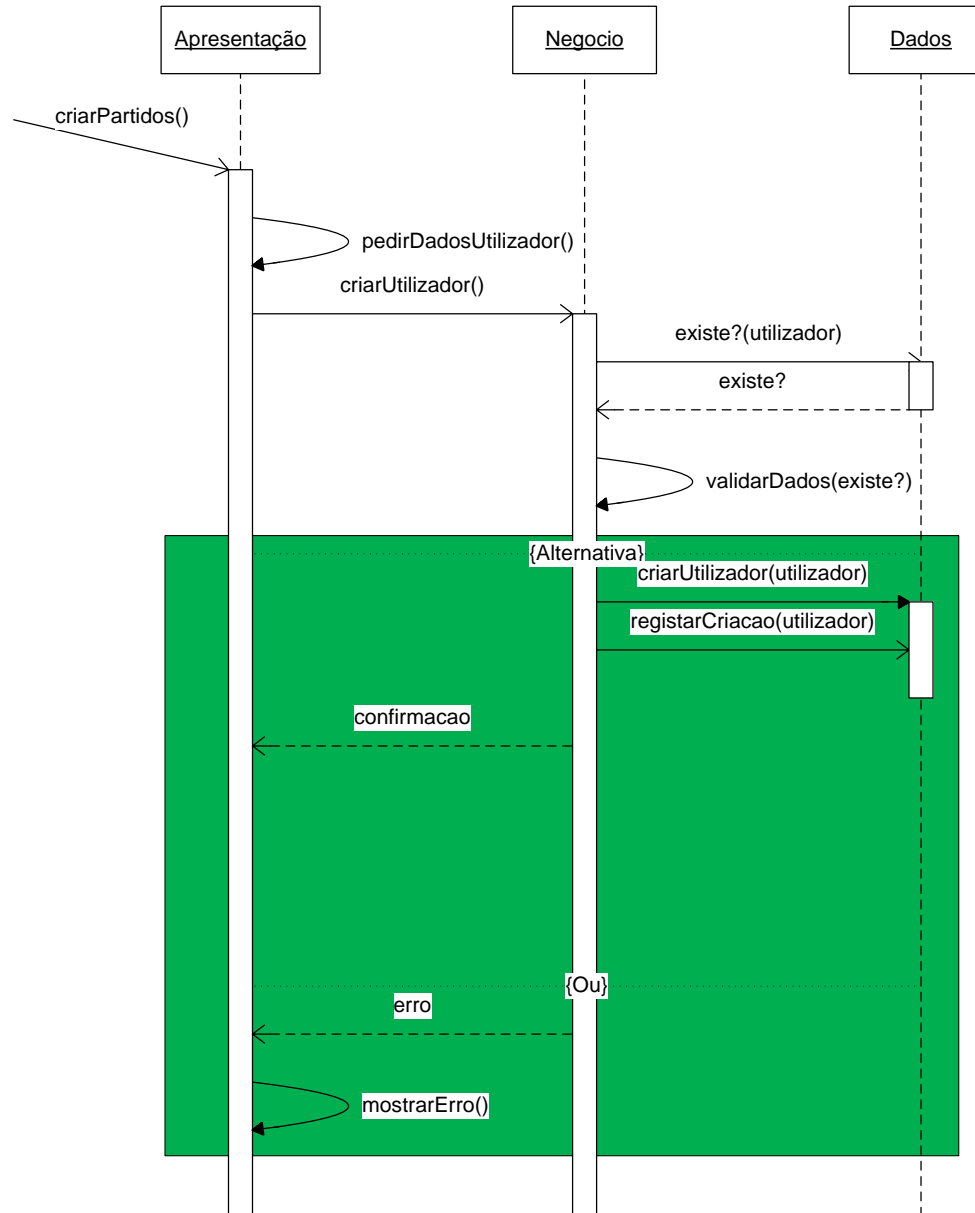
Administrador – Remover Candidato

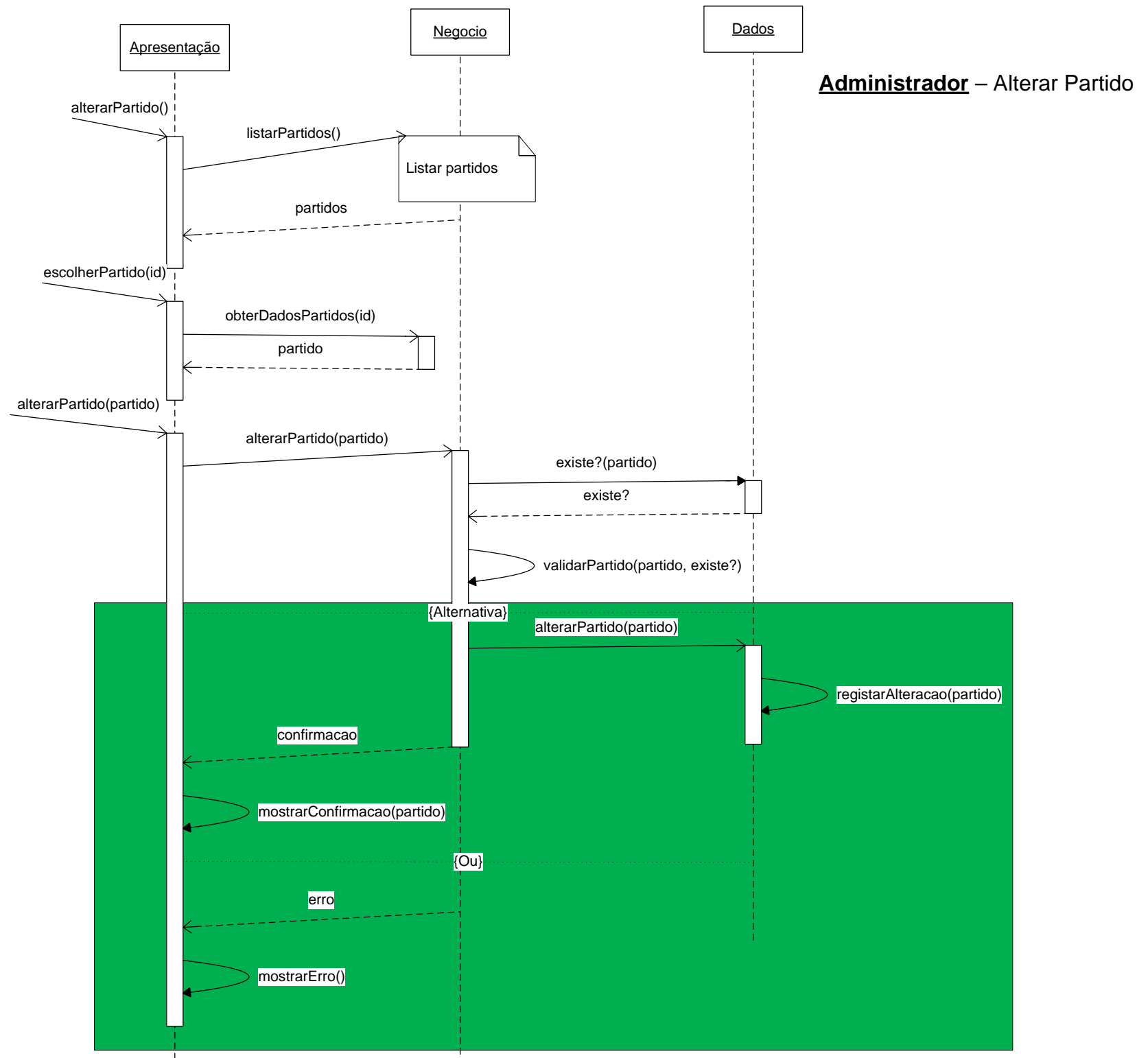


Administrador – Listar Candidatos

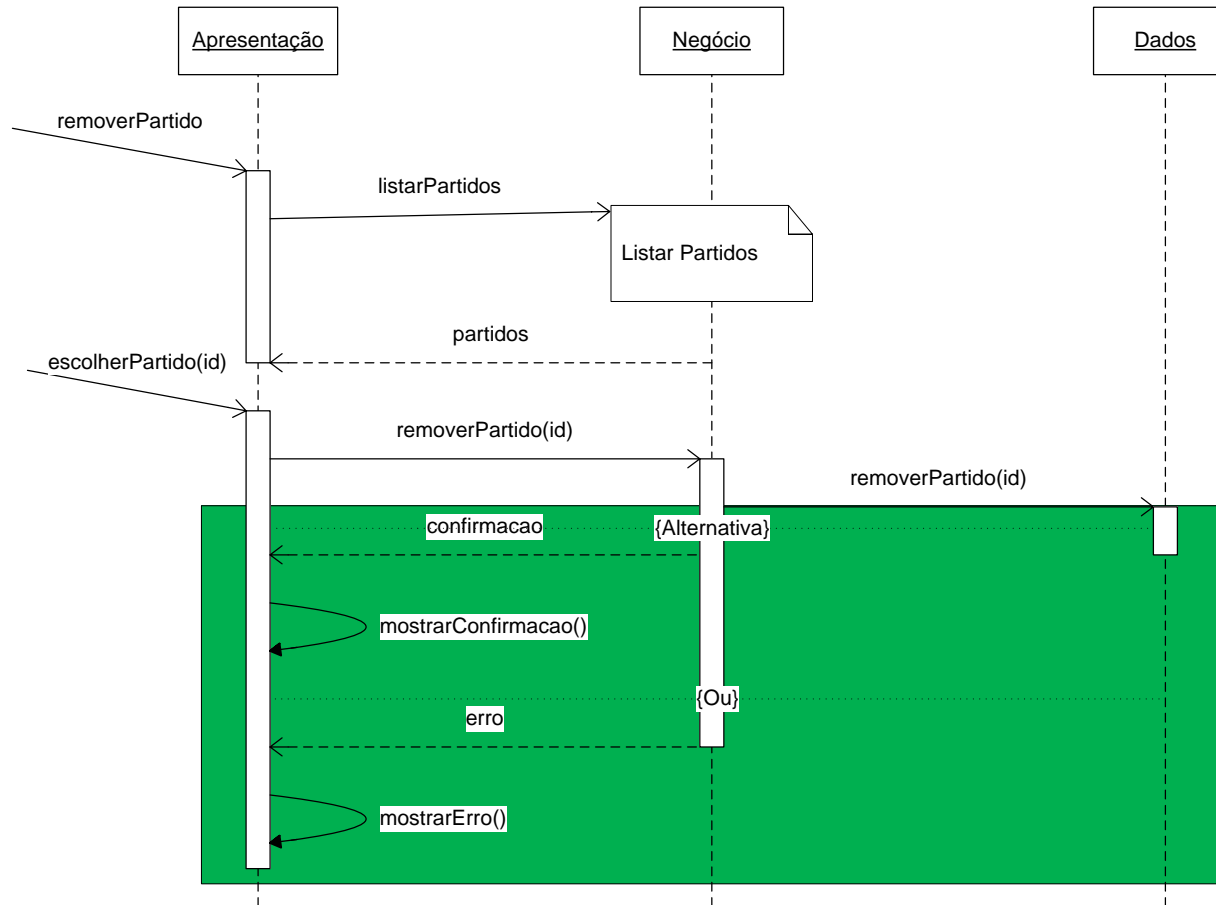


Administrador – Adicionar partidos

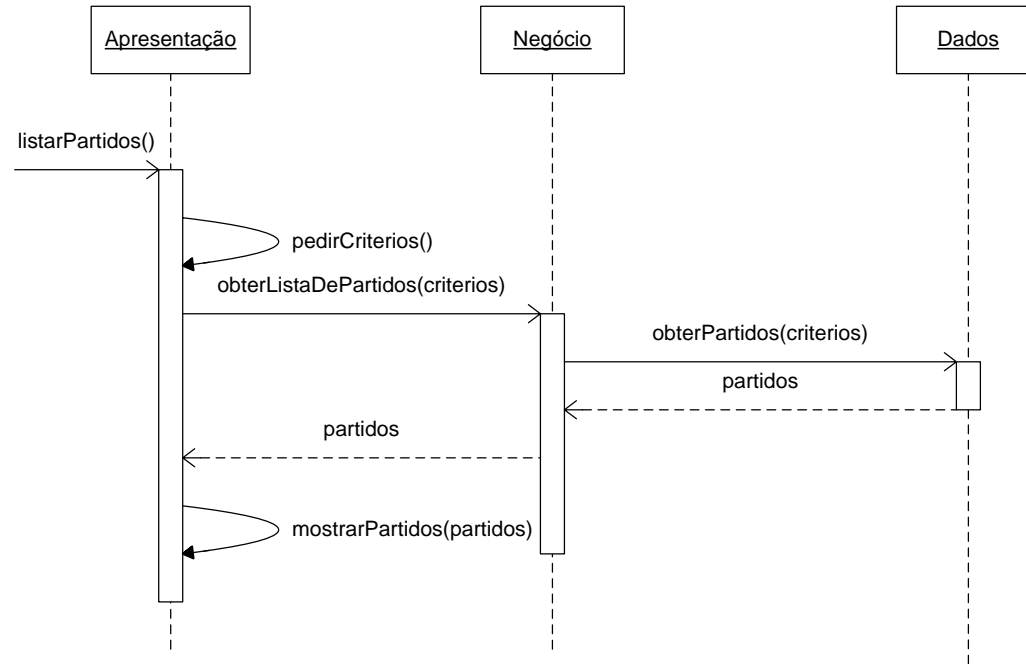




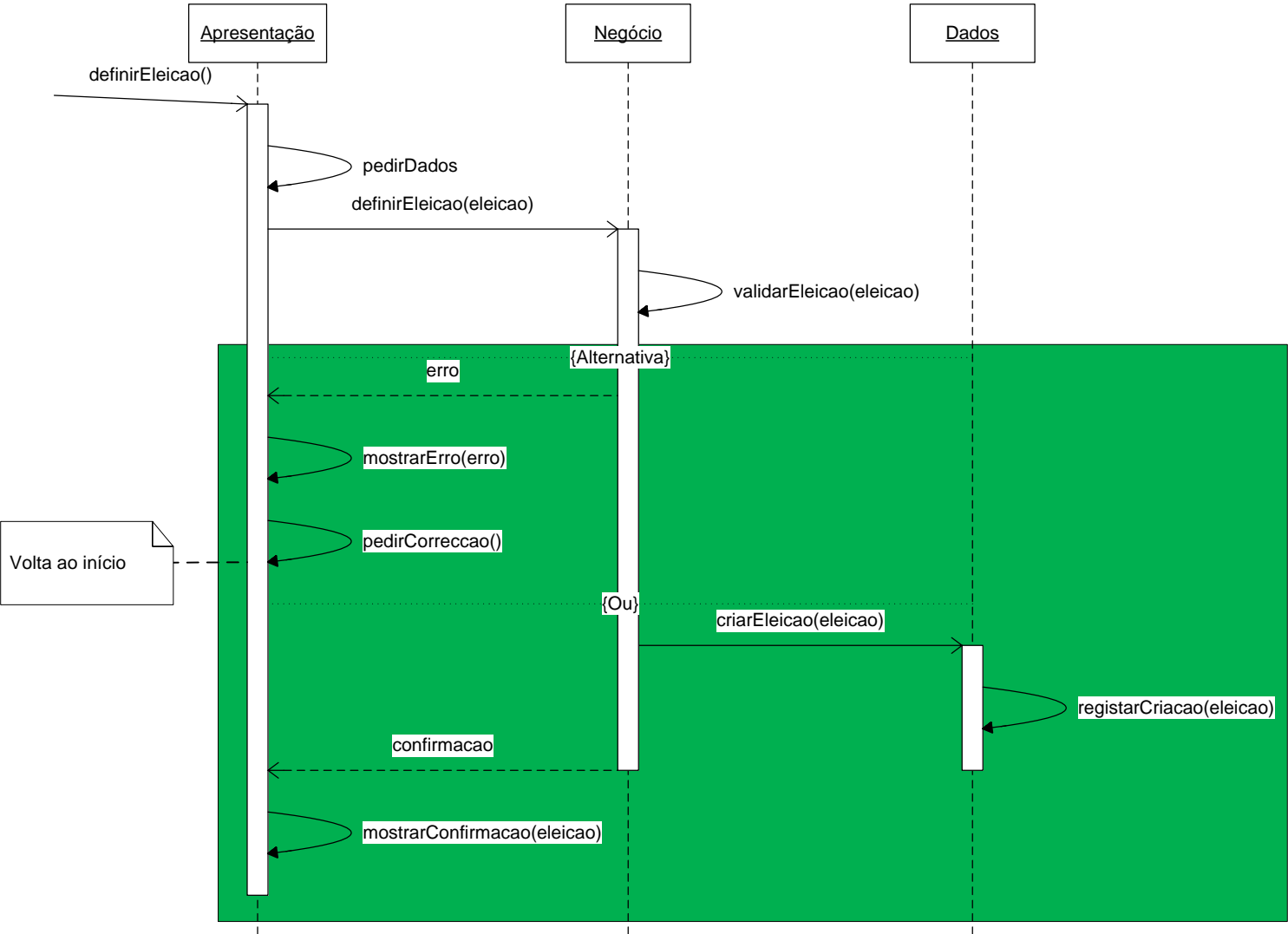
Administrador – Remover partido



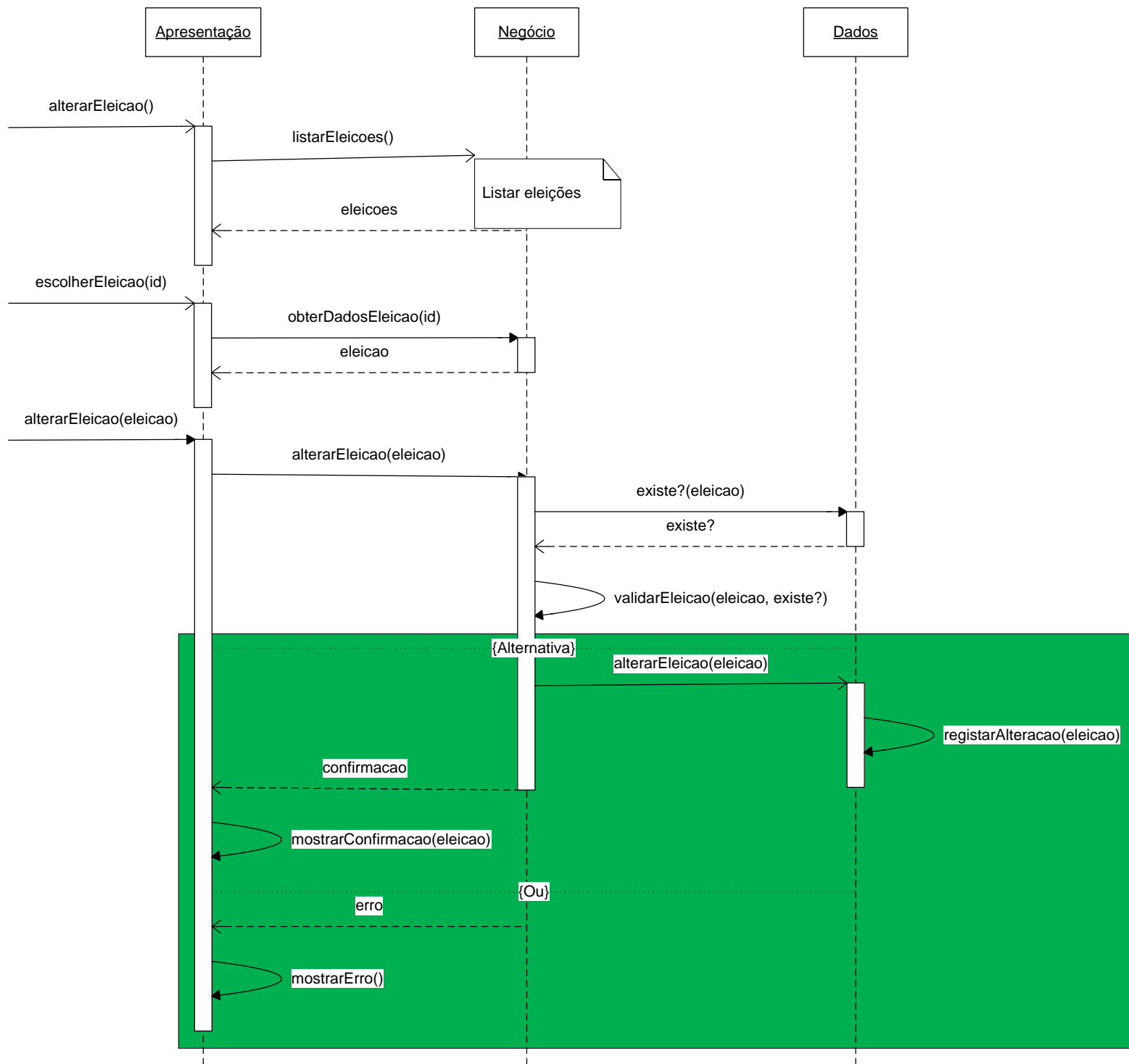
Administrador – Listar Partidos



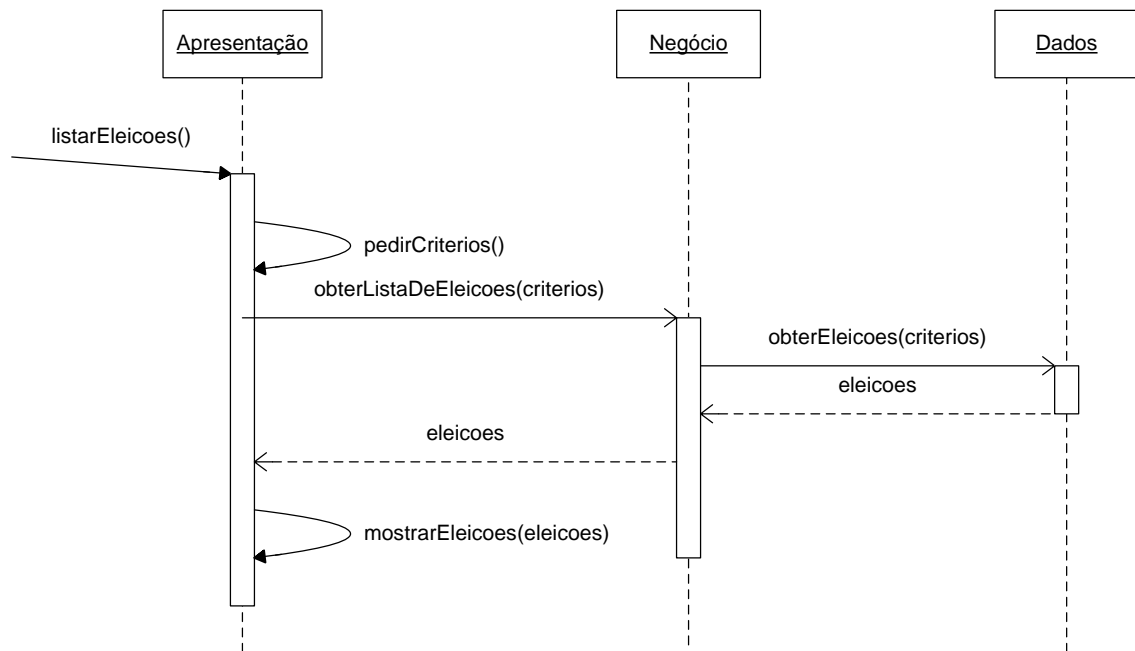
Administrador – Definir eleição



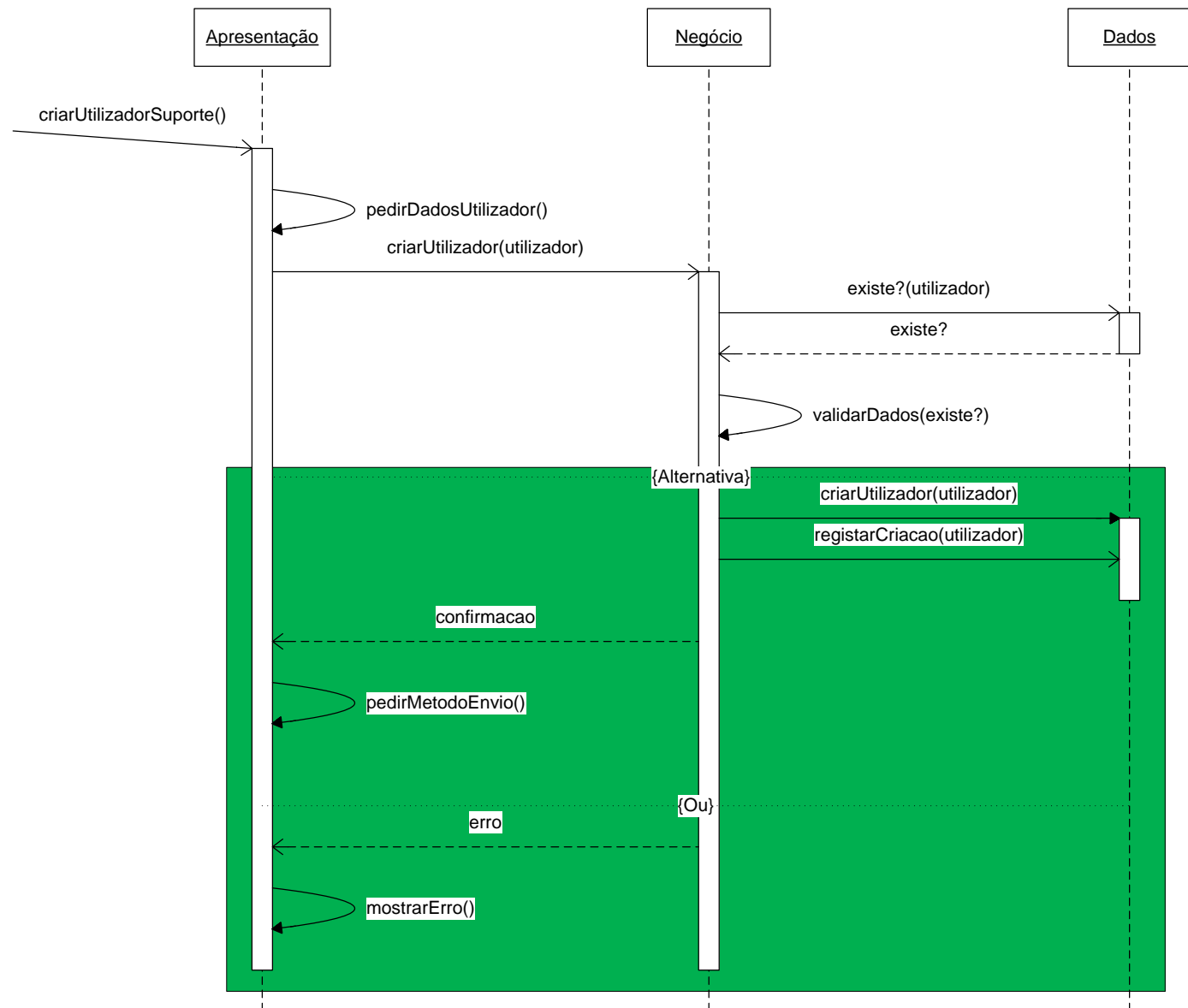
Administrador – Alterar eleição



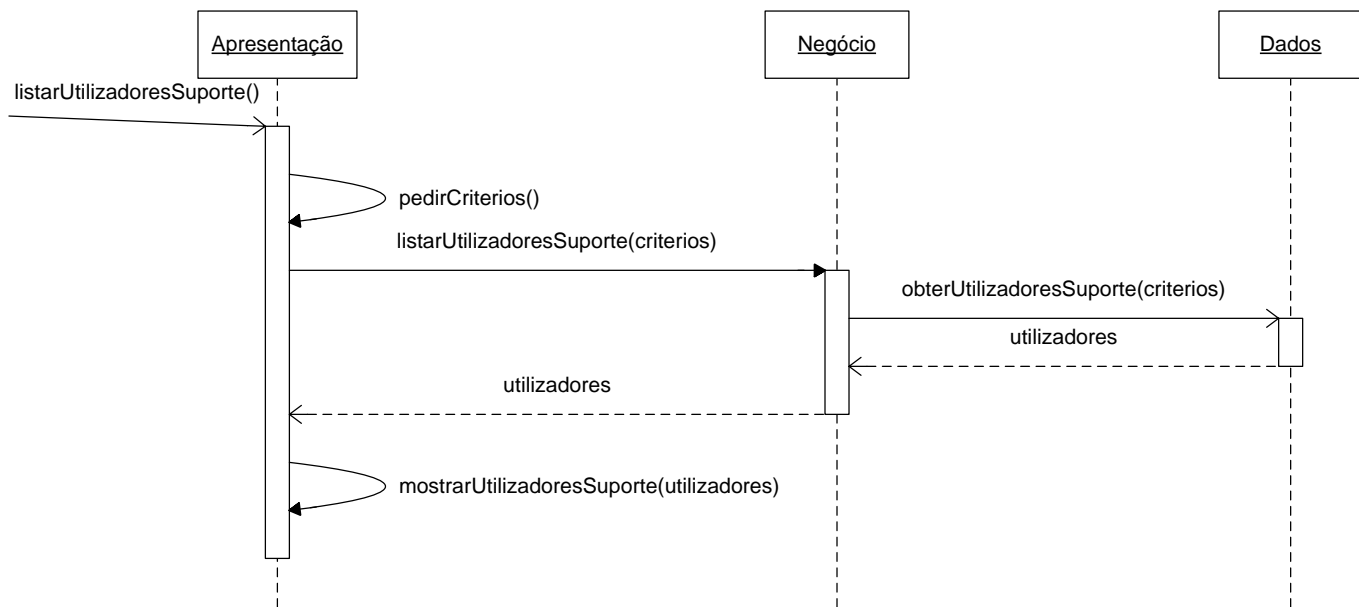
Administrador – Listar eleições



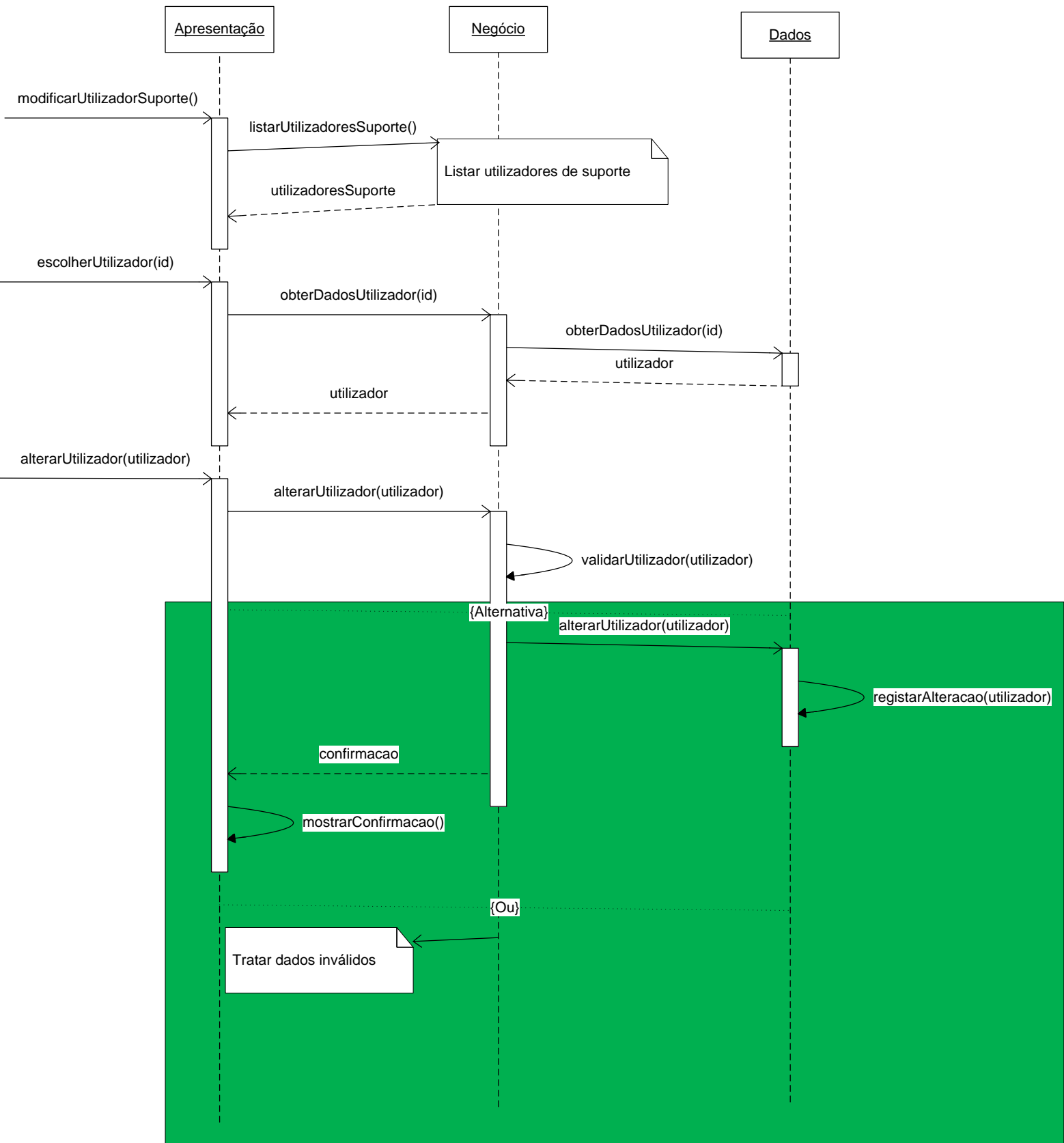
Administrador – Criar utilizador de suporte



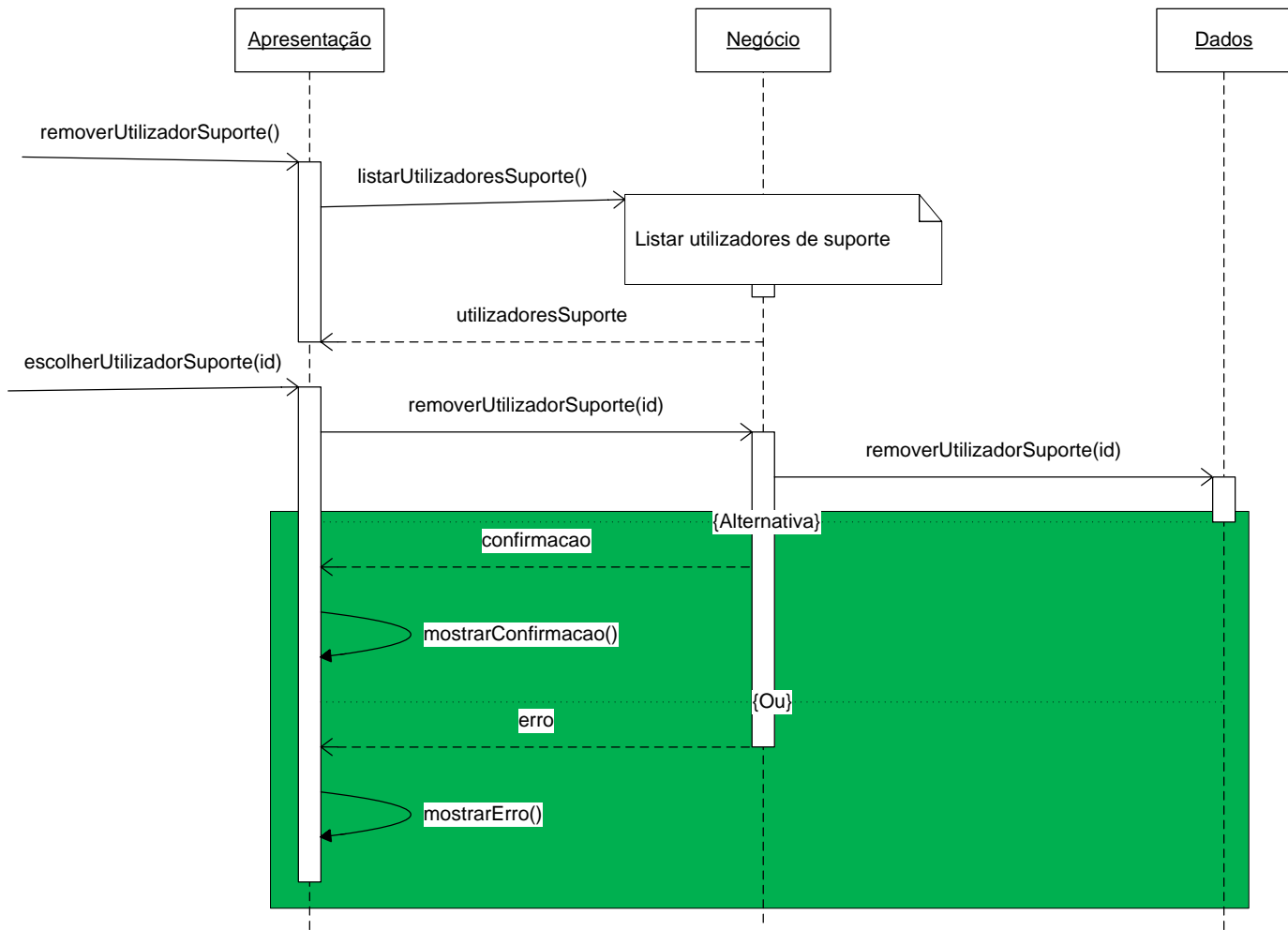
Administrador – Listar utilizador de suporte



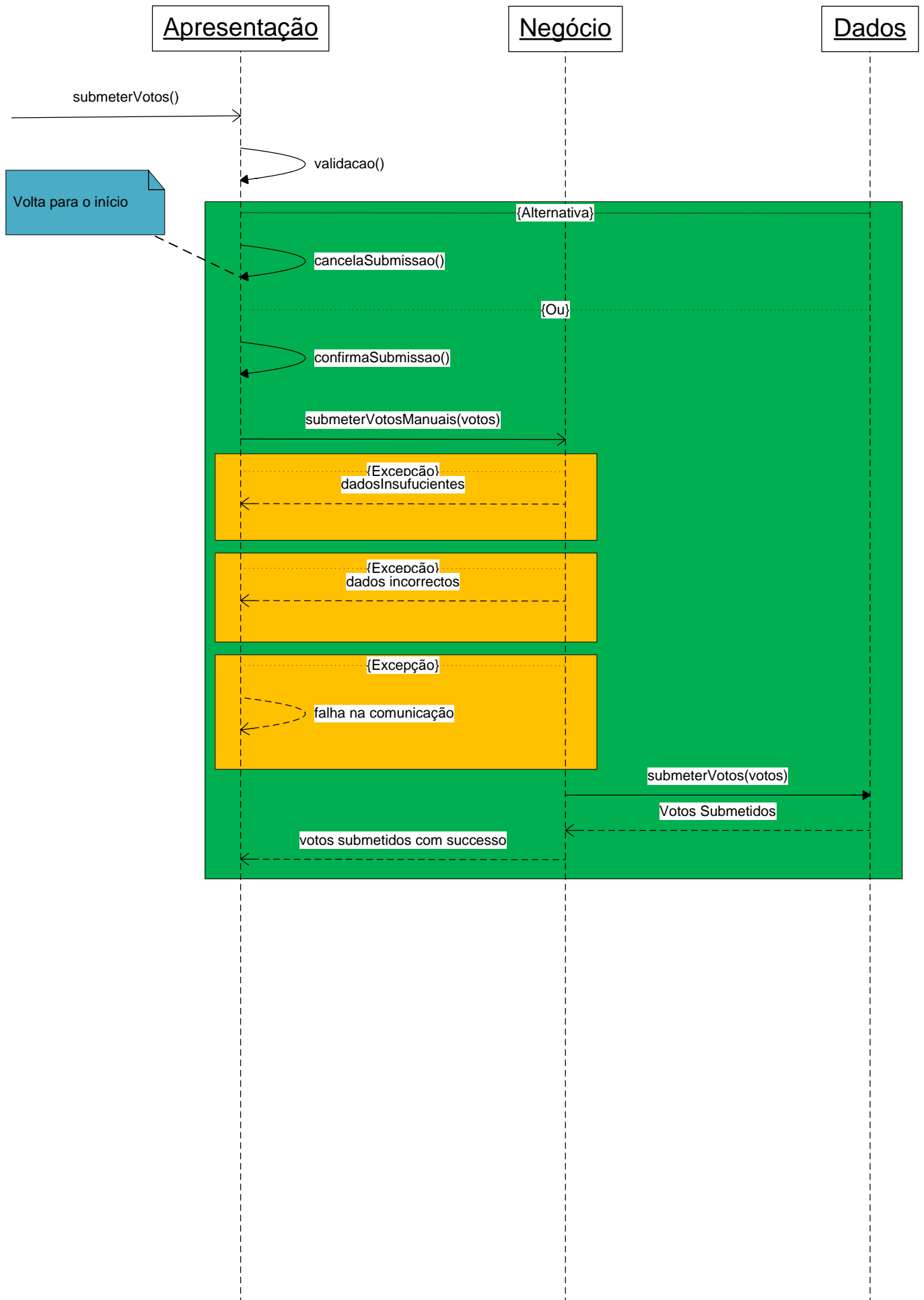
Administrador – Modificar utilizador de suporte



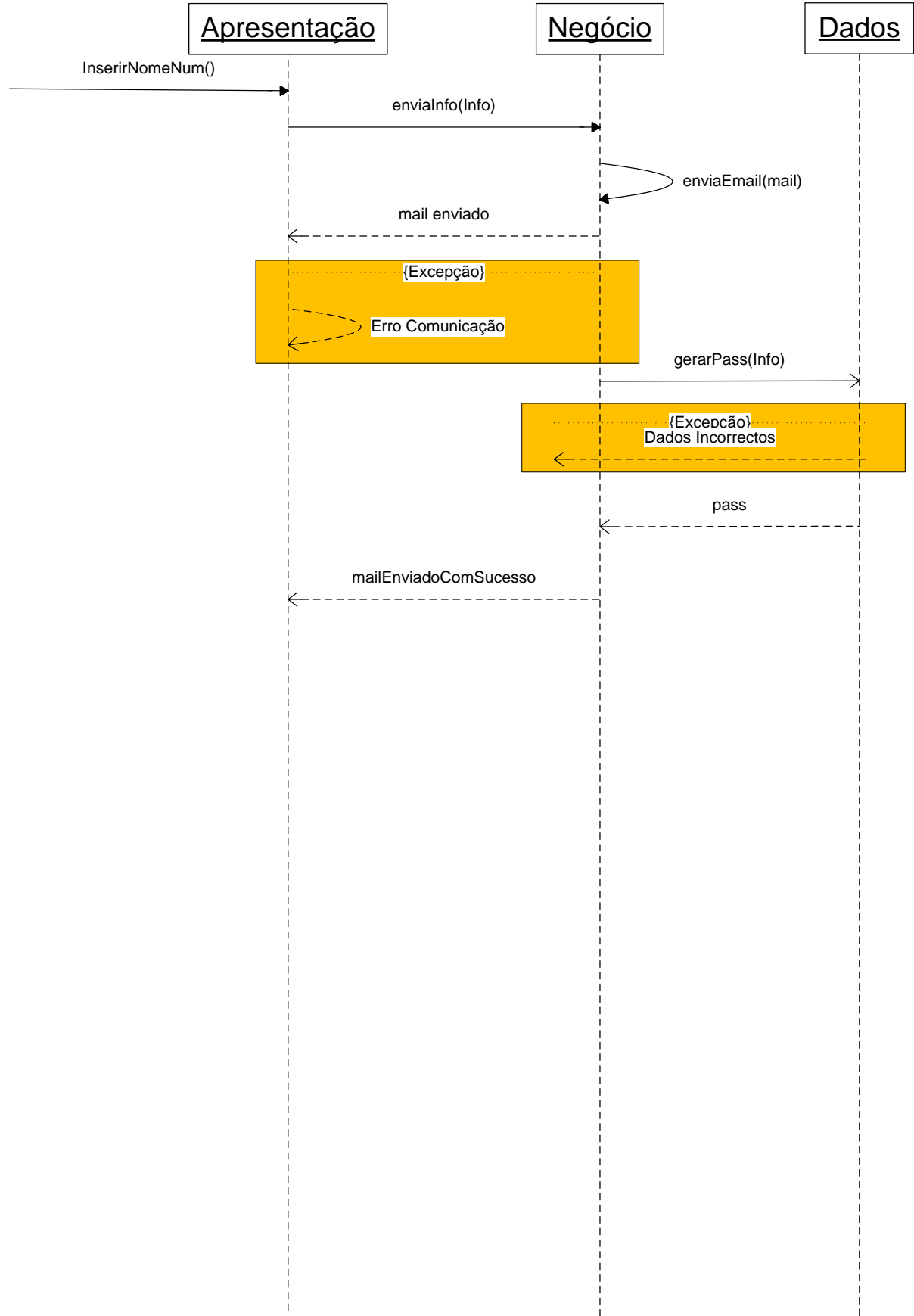
Administrador – Remover utilizador de suporte



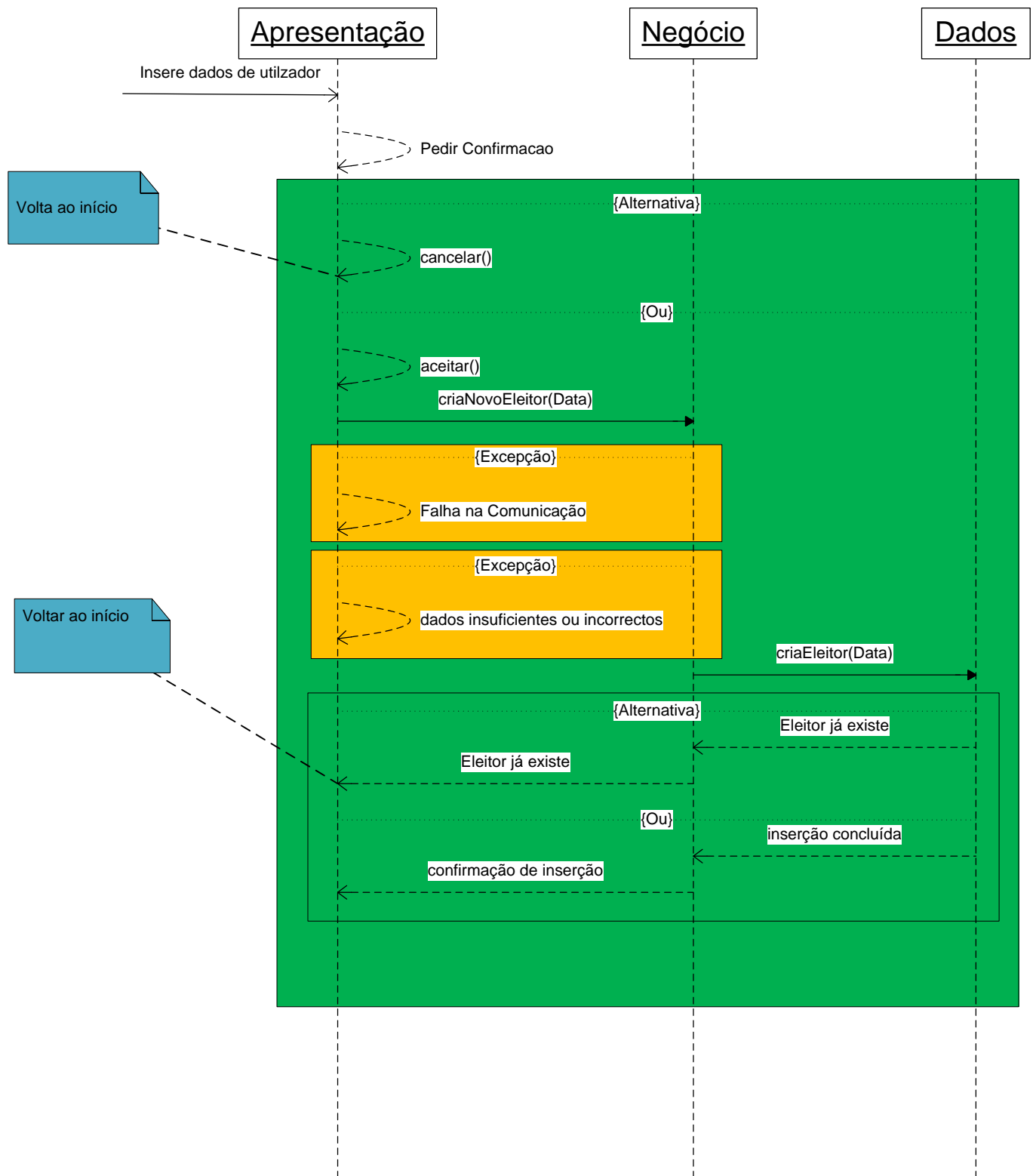
Suporte – Submeter Votos Tradicionais



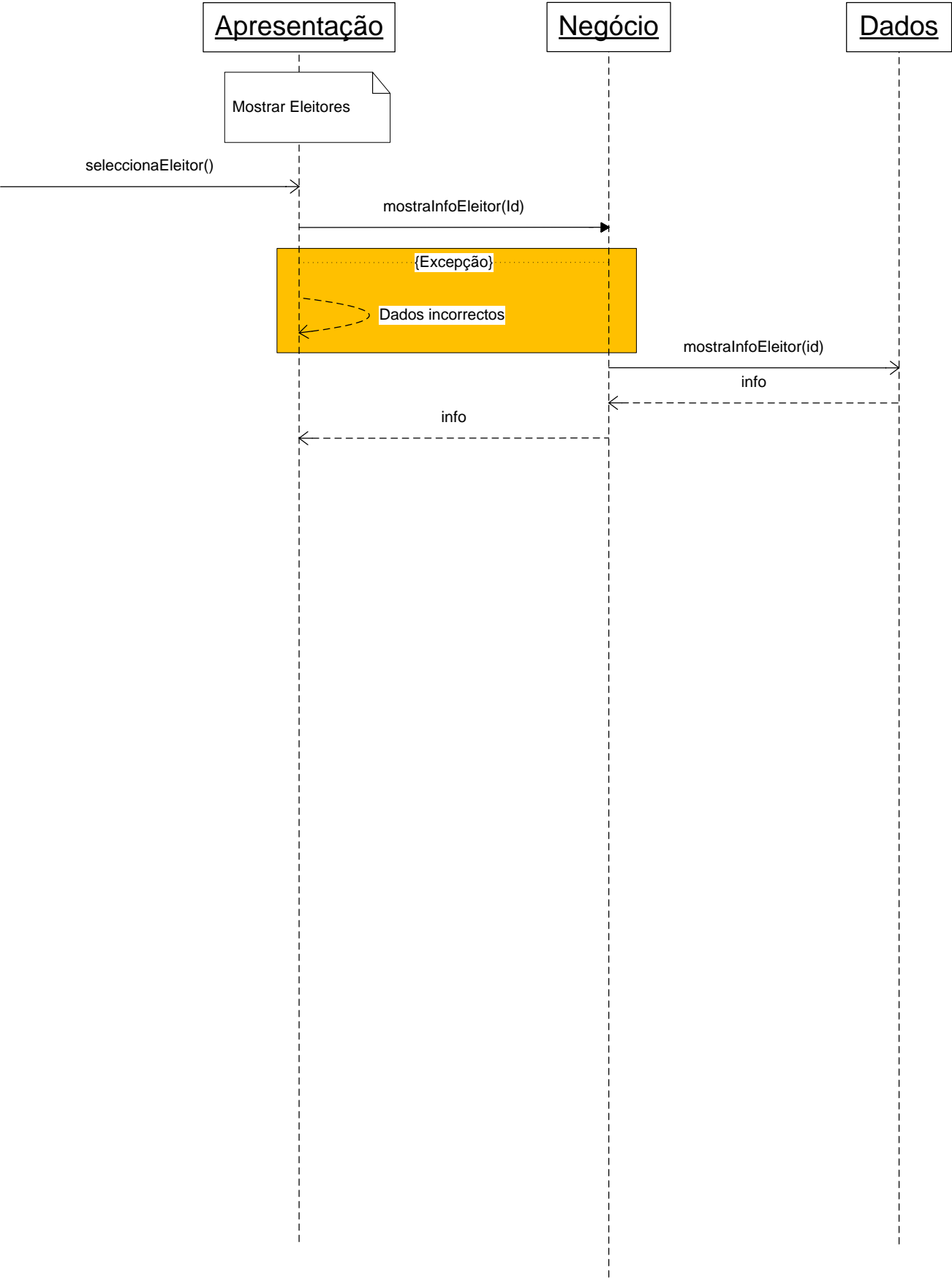
Suporte – Criar Credenciais de acesso



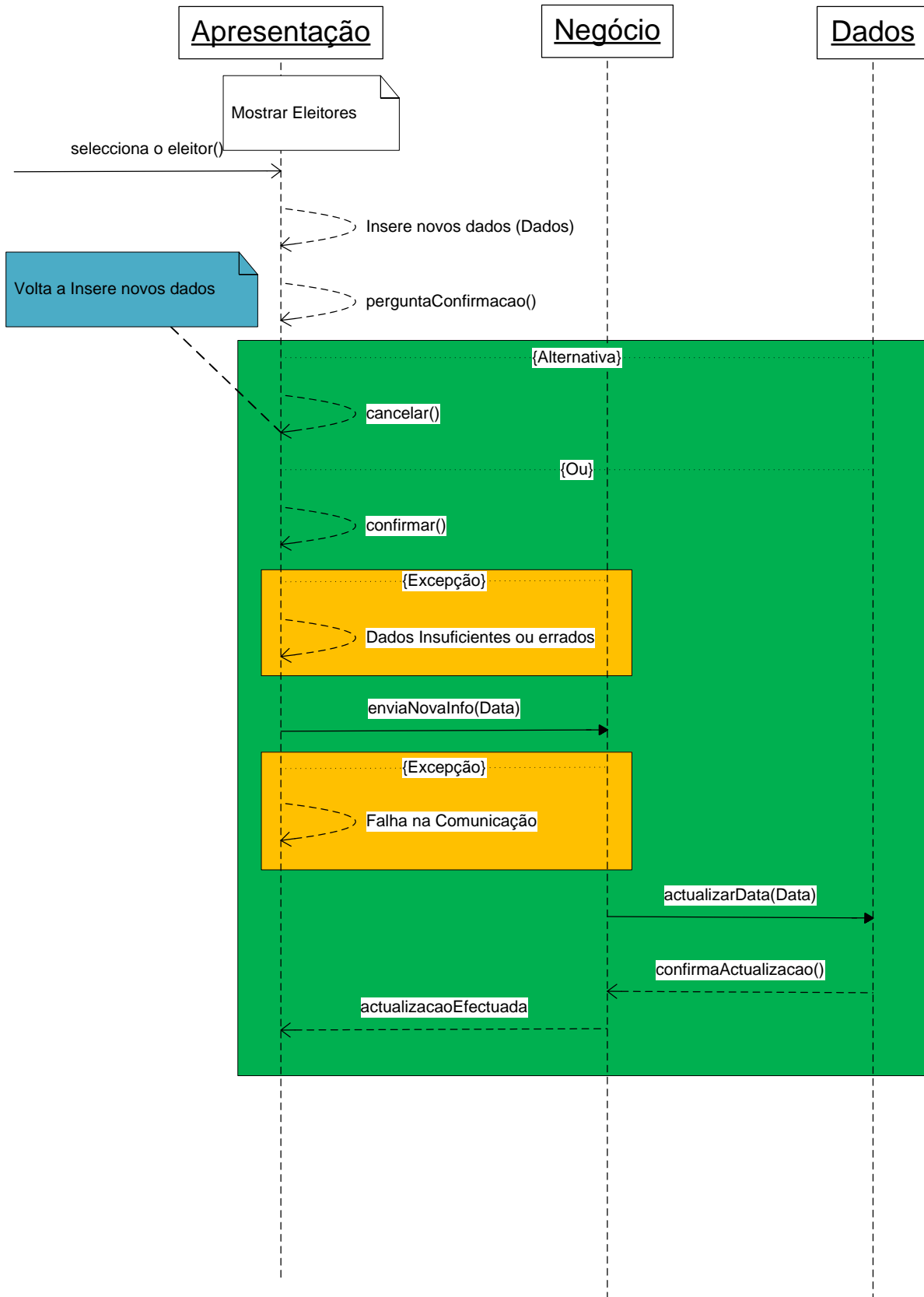
Suporte – Adicionar eleitor



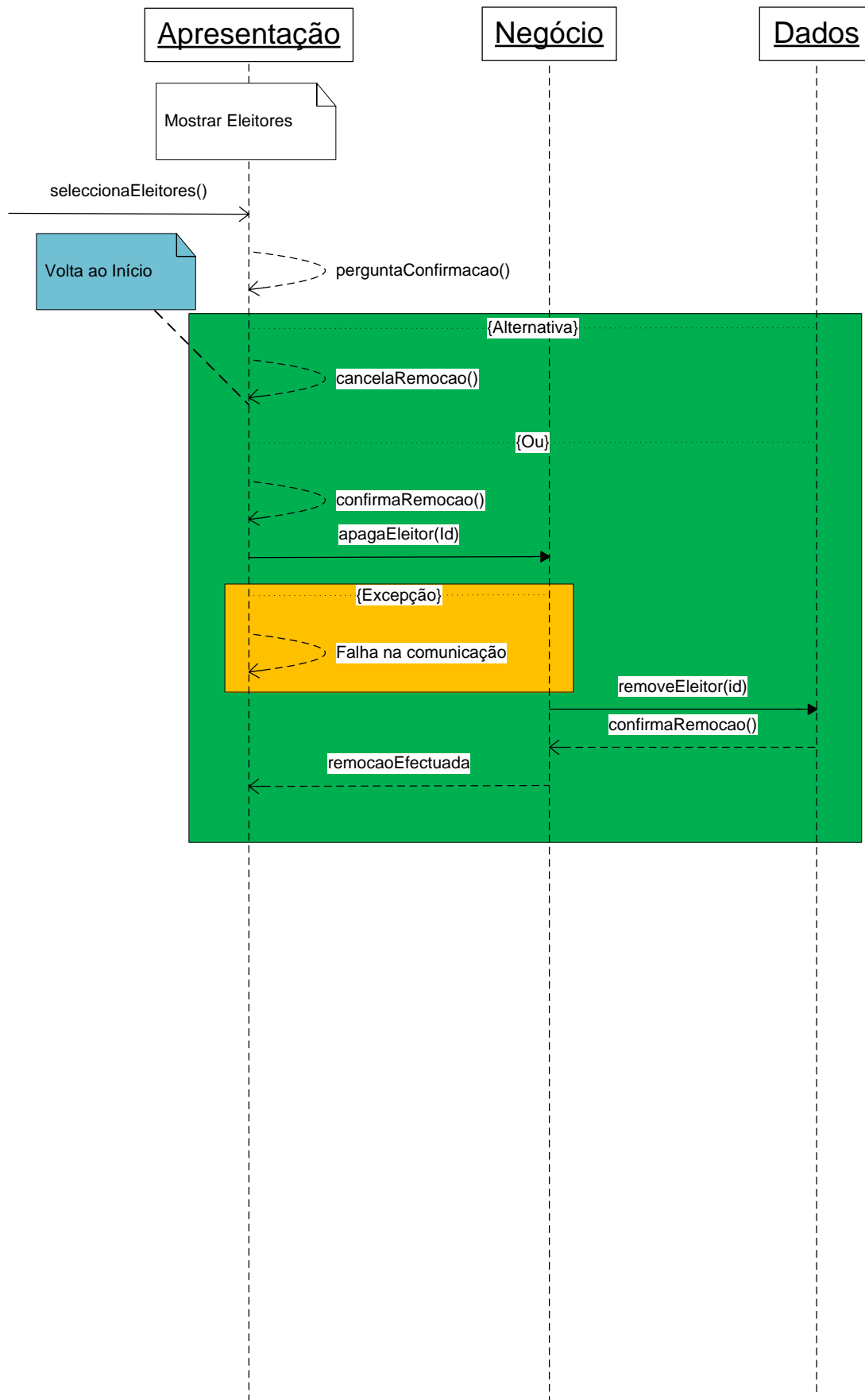
Suporte – Visualizar eleitor



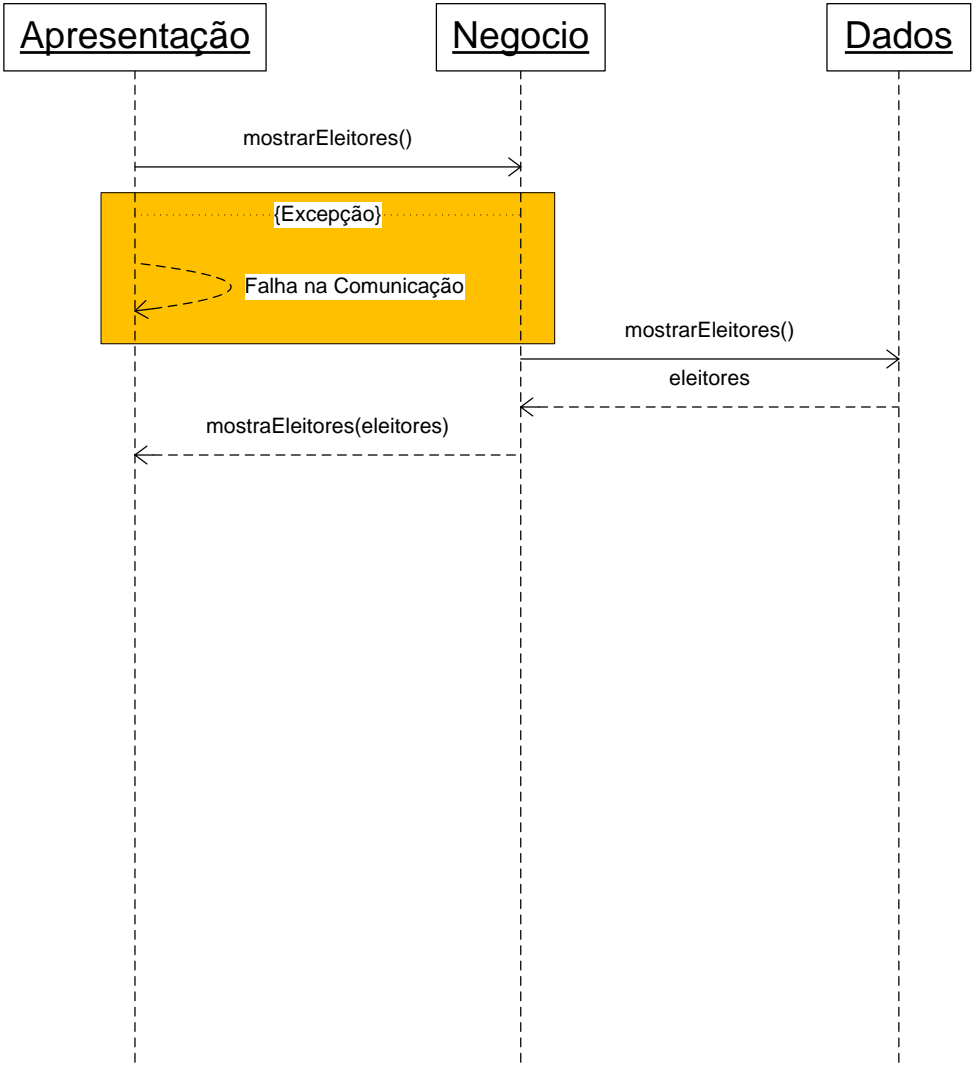
Suporte – Alterar eleitor



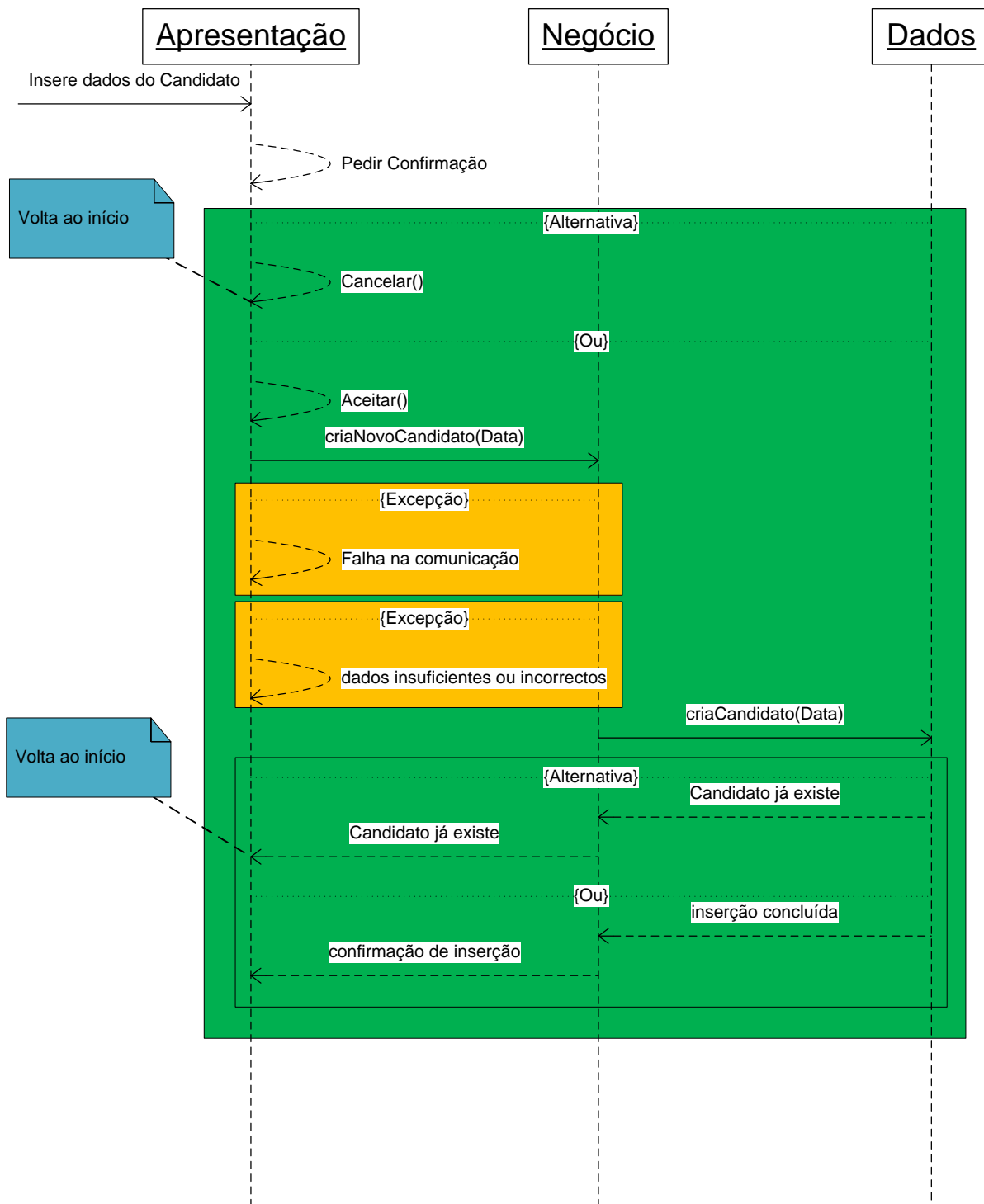
Suporte – Remover eleitor



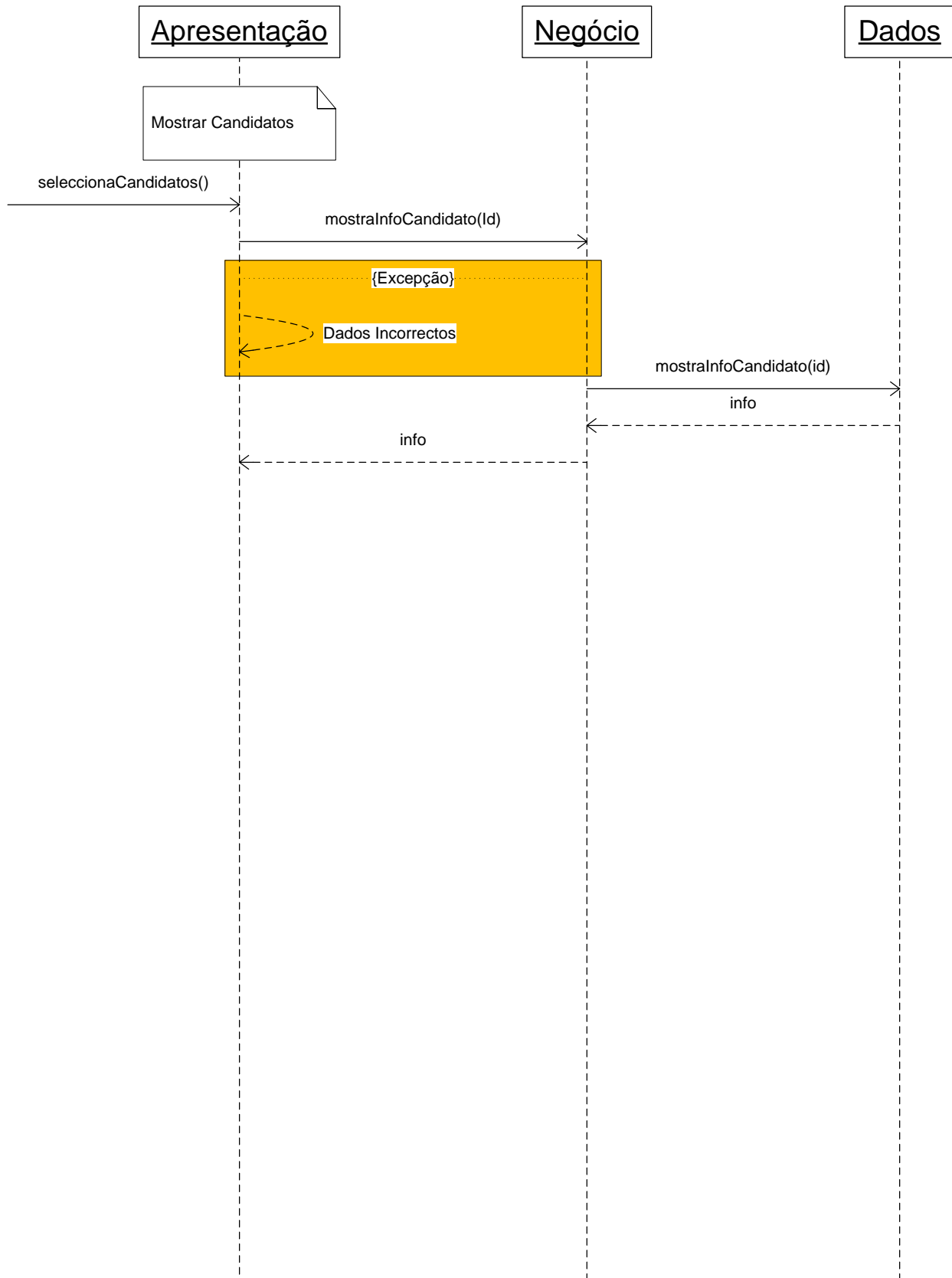
Suporte – Listar eleitores



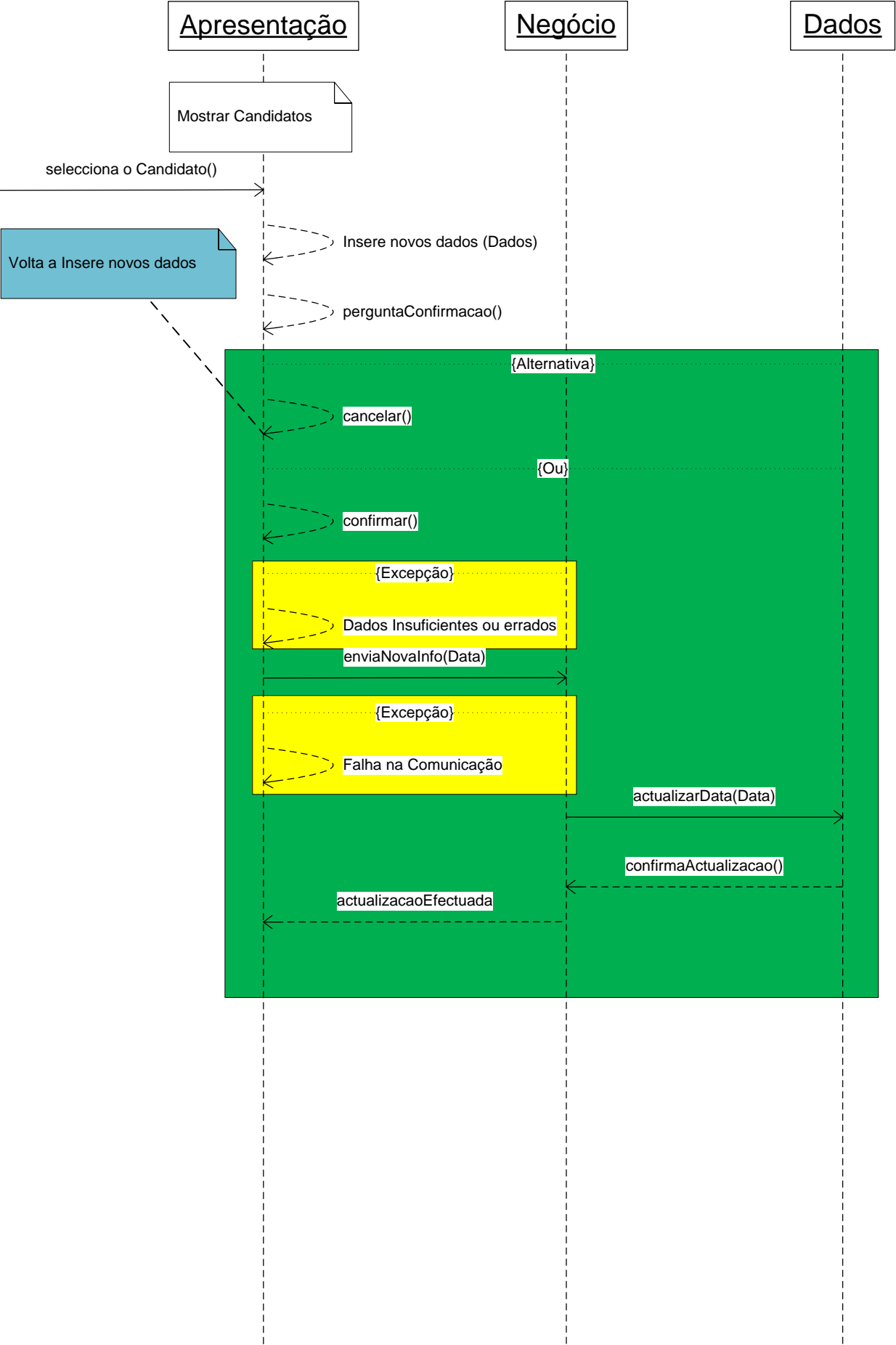
Suporte – Adicionar Candidatos



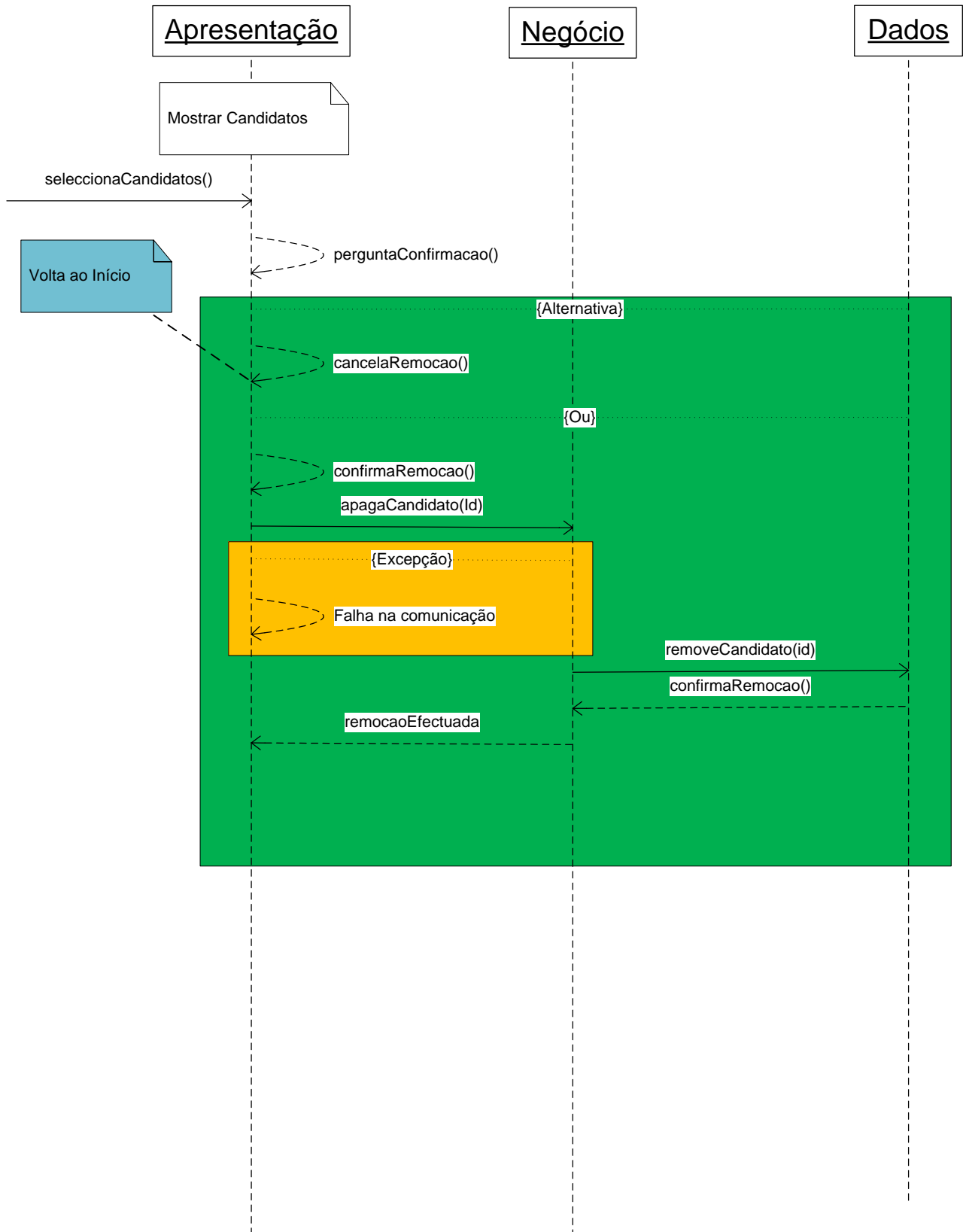
Suporte – Visualizar Candidato



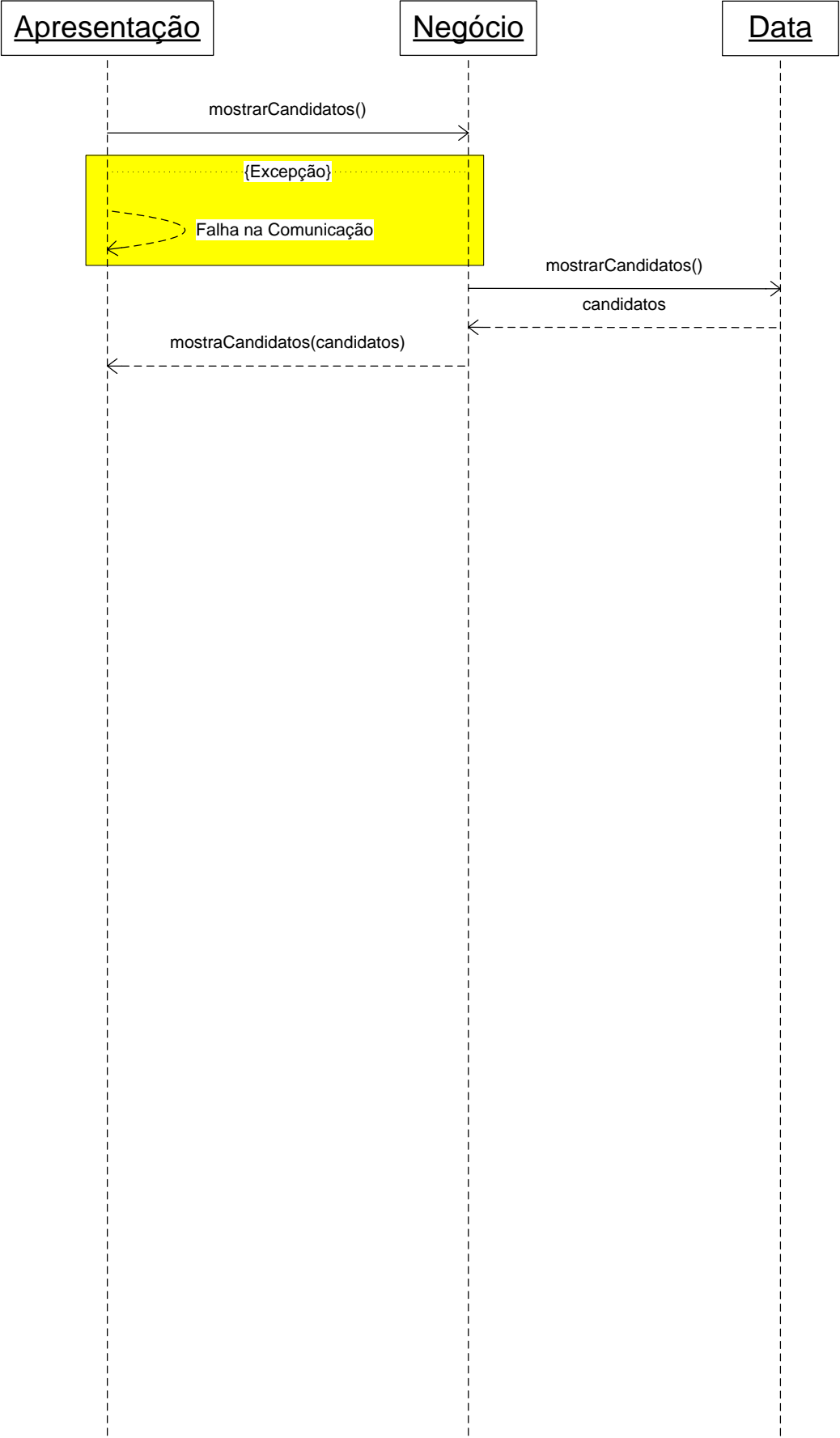
Suporte – Alterar Candidato



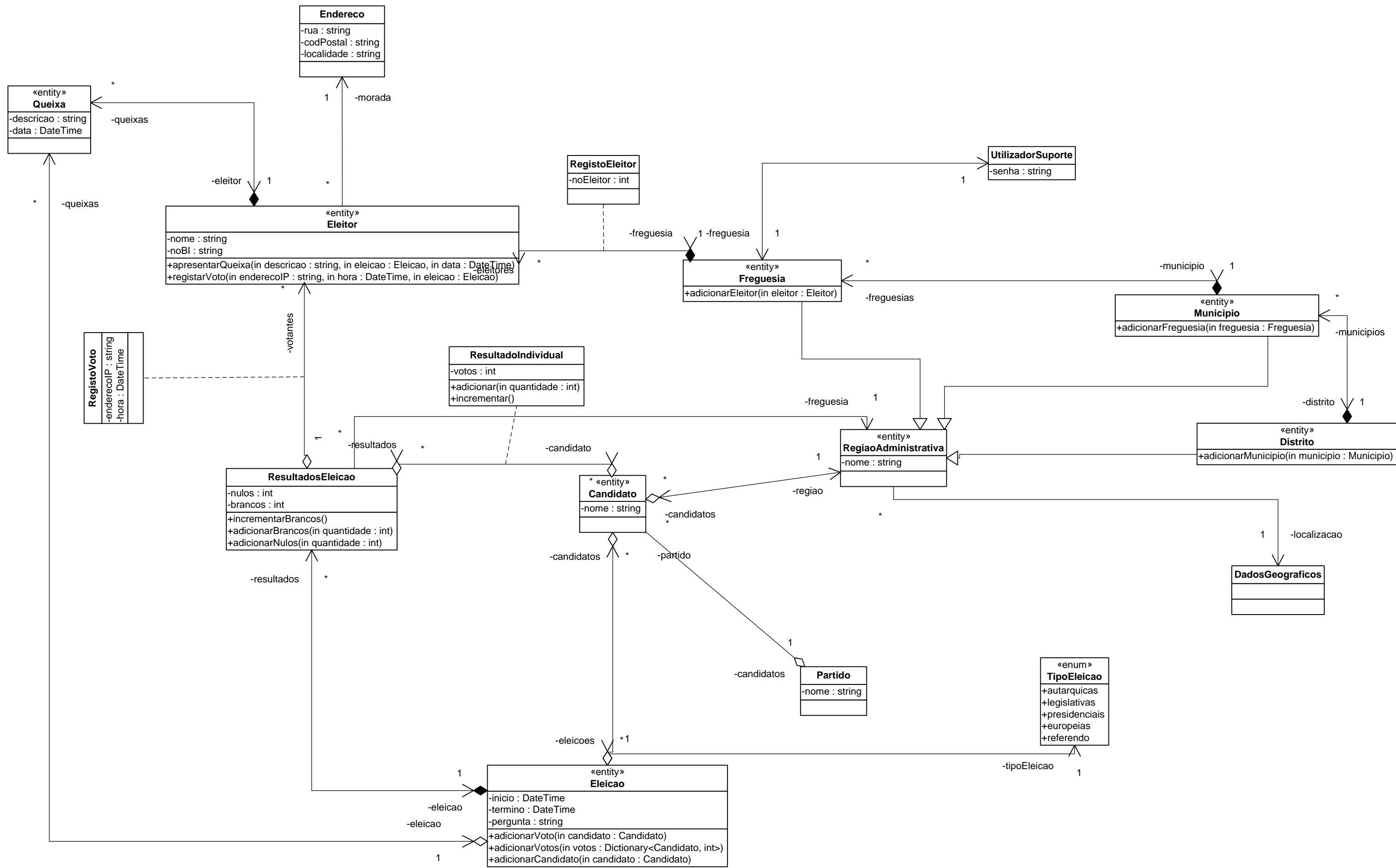
Suporte – Remover Candidato



Suporte – Listar Candidatos



III. Diagrama de Classes



IV. Especificação da Base da Dados

