Projeto Interdisciplinar 2021

pROJETO

Visão DE INTERAÇÃO ENTRE COMPONENTES

01/09/2021

# BLOCO7

# DIAGRAMAS DE SEQUENCIA BLOCO7-WEB

## Caso de Uso Entity create( )

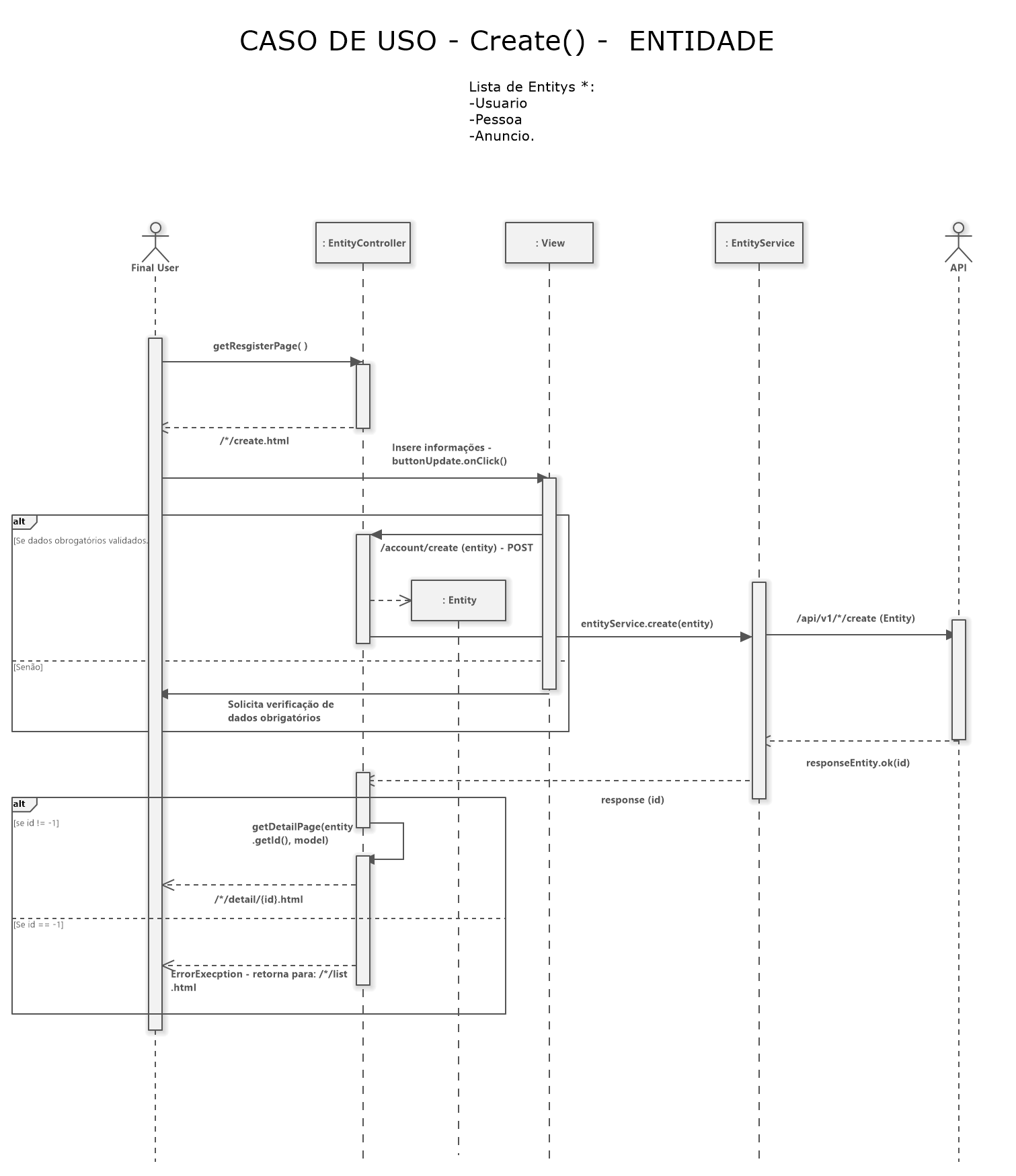


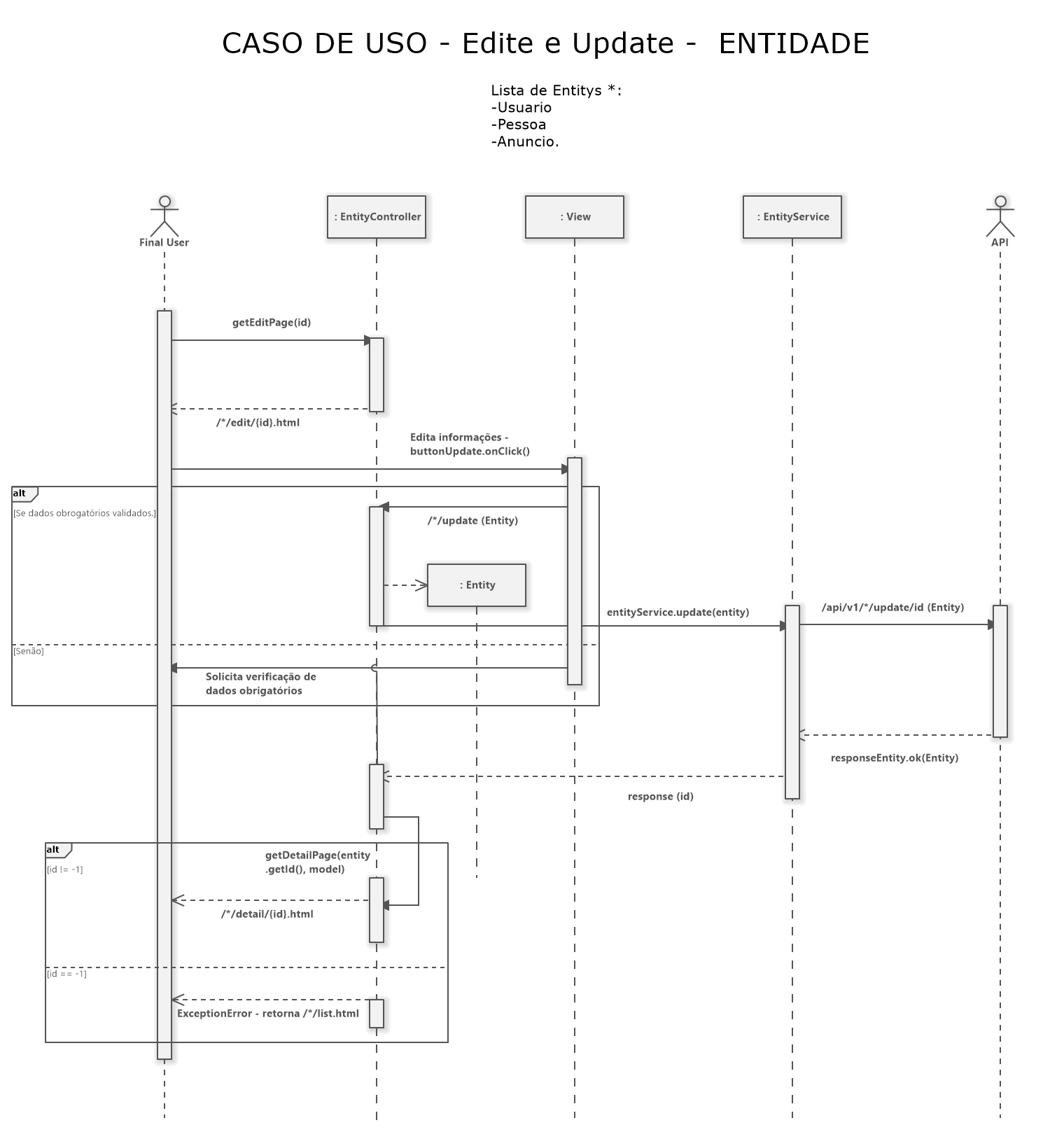
Figura 1 - DSQ BLOCO7-WEB - Create()

Para se realizar o processo de cadastro de uma nova entidade, utiliza-se do processo básico de cadastro representado na imagem a cima (imagem 1). Para este caso, o usuário faz a solicitação da página de cadastro (getRegisterPage()) ao acionar o botão o qual chama a mesma. Esta requisição é encaminhada para o controller que retorna para o usuário a página /\*/create.html (o \* representa o diretório referente a entidade a qual está se realizando o cadastro).

Com a view em mãos, o usuário insere os dados no formulário de cadastro e ao concluir aciona o botão buttonCreate. Neste momento, é realizado uma validação dos dados inseridos. Caso os dados não possuam problemas ou os campos obrigatórios não foram inseridos, é informado para o usuário o erro e solicita ao mesmo a correção para prosseguir. Se as informações estiverem corretas é feita uma requisição para o controller acionando o método entityService.create(entity). Neste momento é instanciado um objeto do tipo da entidade com os dados inseridos pelo o usuário e é passado como parâmetro dentro da função create().

Ao receber a entidade, a classe Service se encarrega de direcionar o endpoint para o Boco7-API. Como retorno, se tem o id gerado pelo projeto Boco7-API. Este id é repassado como retorno do service para o controller e é realizado uma verificação. ‘Se o id for igual a -1, significa que a entidade não foi criada com sucesso e o usuário é direcionado para a pagina /\*/list.html, mas caso seja diferente de -1, a entidade foi criada e o usuário é direcionado para a pagina detalhes referente a esta entidade.

## Caso de Uso Entity update(id)

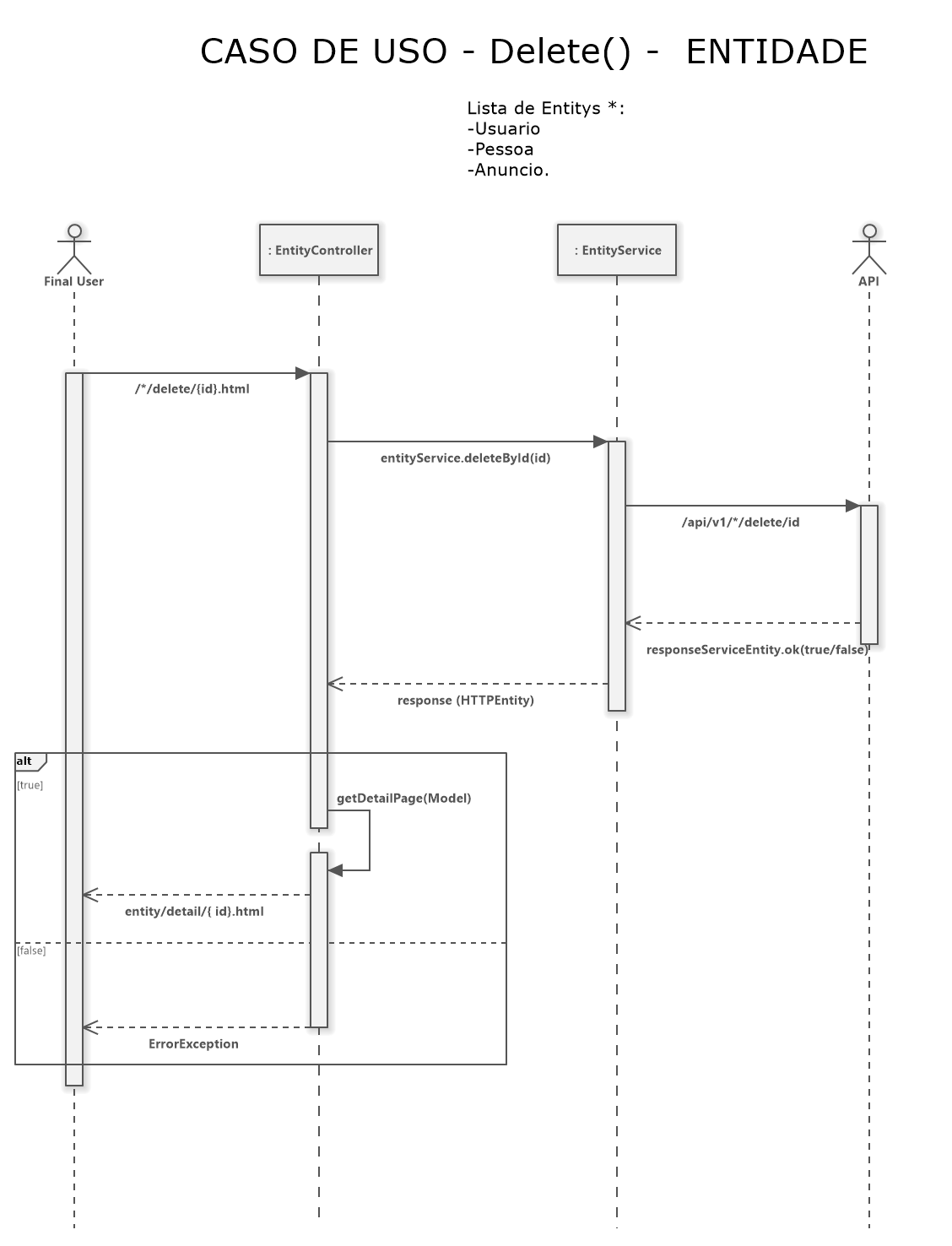


Para realizar a atualização dos dados de uma entidade previamente cadastrada, temos como exemplo o processo representado pela figura 2. Para atualizar, o usuário faz a solicitação da página de editar (getEditPage(id)) ao acionar o botão que que referência o qual objeto será editado. Neste processo, é extraído o id da entidade em questão e o mesmo é passado como parado pela função. Está requisição é encaminhada para o controller que estancia o objeto do tipo entity e o injeta através de um Bean ao montar a página /\*/edit/{id}.html e posteriormente é exibida ao usuário contendo todos os dados da entidade selecionada para edição.

Neste momento, através da view de detalhes, o usuário altera as informações as quais deseja e conclui o processo acionando o buttonUpdate. Neste momento, é realizado uma validação dos dados inseridos. Caso os dados não possuam problemas ou os campos obrigatórios não foram inseridos, é informado para o usuário qual o erro e solicita ao mesmo a correção para prosseguir. Se as informações estiverem corretas é feita uma requisição para o controller acionando o método entityService.update(entity).

Ao receber a entidade, a classe Service se encarrega de direcionar o endpoint para o Boco7-API. O mesmo restorna a resposta o responseEntity e encaminha para o service. Dando sequência, é feito o retorno para o controller com response e o controller, caso o update tenha sido realizado com sucesso, direciona o usuário para pagina de detalhes da entidade.

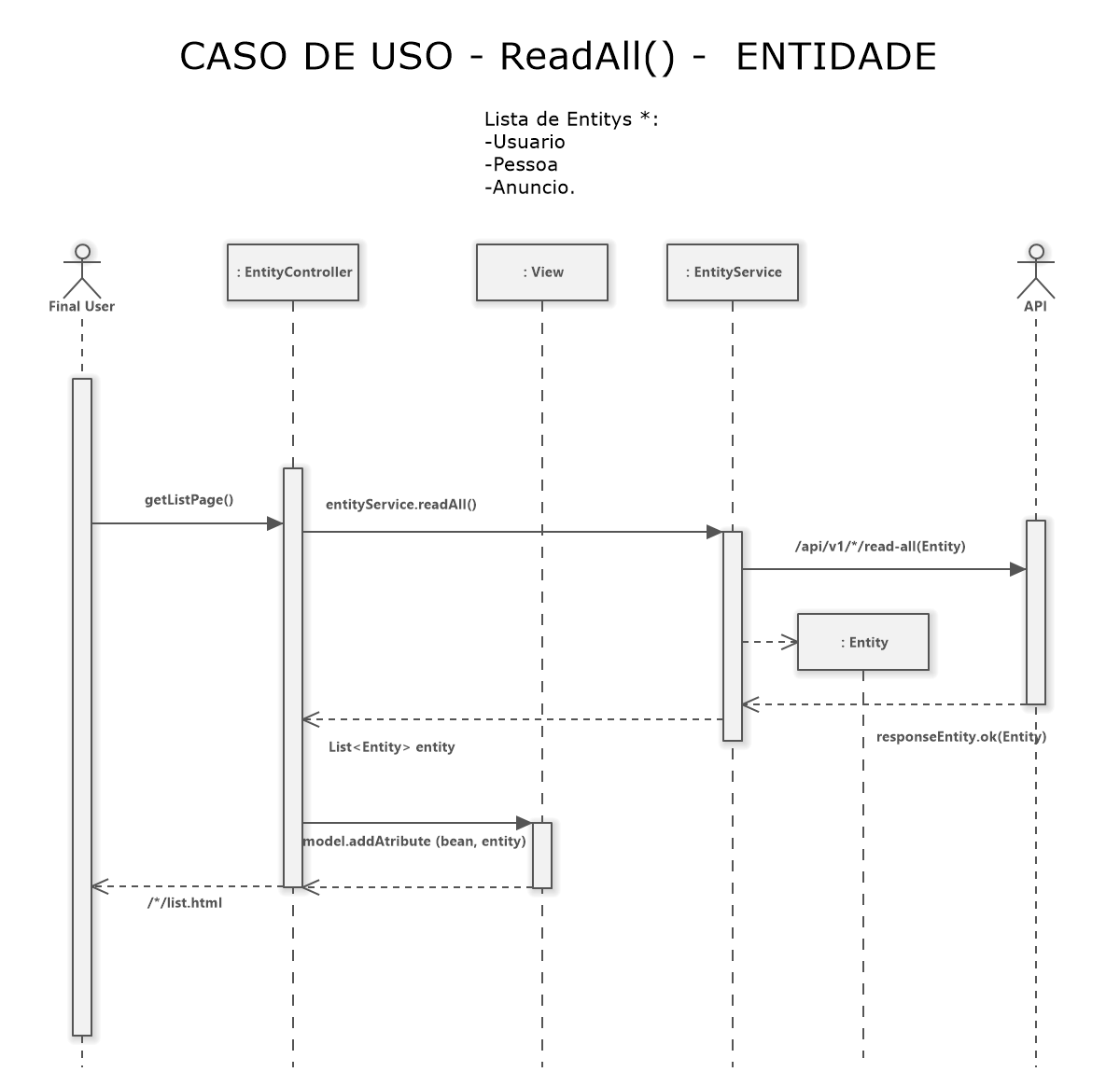
## Caso de Uso Entity delete(id)



Para realizar o processe de deletar uma entidade, o usuário faz a requisição /\*/delete/{id} através do botão de deletar. Neste momento é salvo o id da entidade a ser deletada e é passado como parâmetro para pelo controller através da função entityService.delete(id). A classer service recebe esta requisição e encaminha para o projeto Bloco7-API a requisição de exclusão para o endpoint api/v1/\*/delete/{id}.

Como retorno se tem true ou false. Caso seja falso, é retorno um erro para o usuário impedindo a exclusão. Se caso verdadeiro, a entidade é excluída com sucesso.

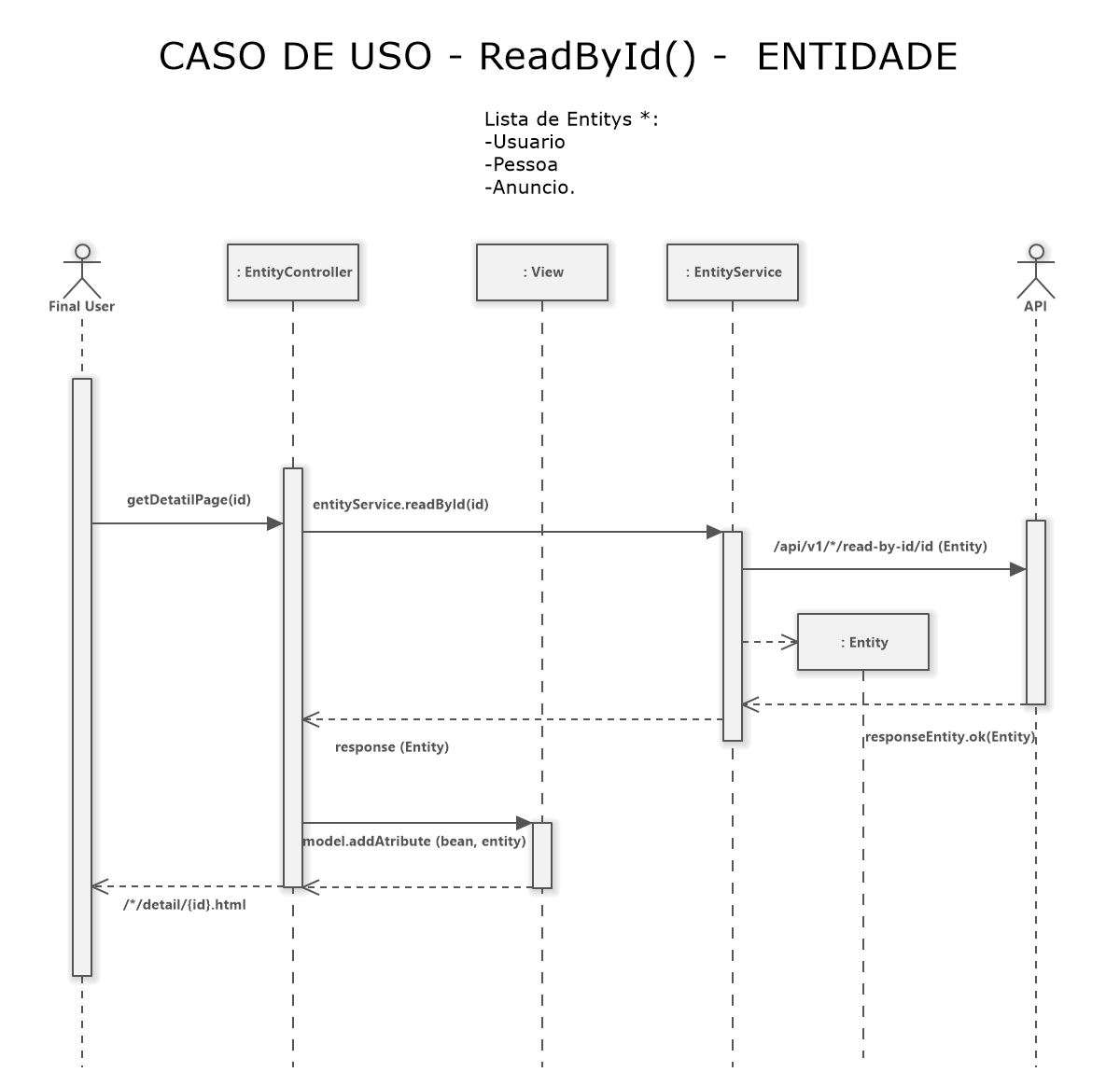
## Caso de Uso Entity readAll()



Para se realizar a listagem de todos os cadastros de uma entidade, o usuário faz a requisição para o controller através da chamada getListPage(). O controller aciona a classe de serviço que direciona o endpoint para Bloco7-API indicado por /api/v1/\*/read-all. Neste momento é criado uma lista instanciado a entidade para que receba a resposta da API.

Ao receber o resultado de responseEntity, o serviço alimenta a lista com as entidades e repassa para o controller que fica encarregado de injetar o model no html e retornar para o usuário a pagina /\*/list.html com a lista de todos os cadastros existentes da entidade em questão.

## Caso de Uso Entity readById(id)



Para a leitura dos detalhes de apenas um cadastro, o usuário faz a requisição através do controller ao selecionar qual cadastro deseja visualizar os dados. Esta requisição é realizada através da função getDetailPage(id). Ao realizar esta chamada a classe de controle é acionada a qual executa a função para listagem de detalhes (entityService.readById(id)) e passa como parâmetro o id.

Esta requisição é enviada para classe entityService que tem como objetivo realizar o direcionamento da requisição para o projeto Bloco7-API. Para criação do objeto, o service instancia uma variável do tipo da entidade que ficara encarregada de encapsular os dados recebidos através do API. Estes dados são validados através do método responseEntity.ok(entity) o qual encaminha para o controller o response contendo todos os dados solicitados do cadastro da entidade.

Após isto, o controller monta a view de detalhes (/\*/detail/{id}) e a retorna para o usuário que poderá visualizar estas informações.

# Aprovação e autoridade para proceder

Aprovamos o projeto como descrito acima e autorizamos a equipe a prosseguir.

|  |  |  |
| --- | --- | --- |
| Nome | Assinatura | Data |
|  |  |  |