

10

## SQLite Find Update

### Transcrição

[00:00] Estamos com esse problema de que quando clicamos no agendamento que falhou da primeira vez e tentamos reenviar esse agendamento, ele não está sobrescrevendo, ele não está atualizando o registro de agendamento que já existe no banco de dado do SQLite local, ele está criando aqui embaixo, está sempre criando um novo registro, não queremos isso, queremos que ele atualize, queremos que ele faça um update, ao invés de um insert.

[00:31] Só que além disso ele também não está nos informando se deu sucesso ou falha nesse segundo reenvio, quando eu confirmo ele não fala nada, ele simplesmente cria uma nova linha aqui na lista, quando eu rolo a tela aparece uma nova linha, mas ele não avisa que deu sucesso ou falha nesse reenvio.

[00:54] Temos que avisar o usuário, tem que alertar o usuário que teve o sucesso ou a falha do reenvio, para fazer isso eu tenho que assinar mensagens, quando o agendamento é gravado no servidor do Aluracar, vimos que lá no serviço do agendamento, no “AgendamentoService” estamos lançando algumas mensagens, temos a mensagem do sucesso e a mensagem da falha do agendamento, precisamos fazer também nessa página que é “AgendamentosUsuarioView”.

[01:28] Vamos criar aqui uma nova assinatura de mensagem, eu utilizo o serviço de mensagem do Xamarin Forms, que é o “MessagingCenter”, eu faço “Subscribe”, para assinar essa mensagem, passa o tipo que é agendamento e eu vou passar aqui a instância “this” e vou passar o tipo da mensagem, que é “SucessoAgendamento”.

[01:51] Para o “SucessoAgendamento” estamos assinando essa mensagem, “SucessoAgendamento” e aqui vai o quê? Vai um método anônimo, vai uma Action que vai receber a instância do “agendamento” e aqui no corpo dessa Action vamos colocar o quê?

[02:14] Vamos colocar o código para poder exibir a mensagem para o usuário e para exibir a mensagem para usuário temos que utilizar o “DisplayAlert”, “DisplayAlert”, vamos exibir o quê?

[02:25] Primeiro como título eu vou colocar “Reenviar” esse vai ser o título da mensagem, em seguida o corpo da mensagem que vai ser “Reenvio com sucesso” e no final eu coloco a string que vai indicar o texto do botão do “Ok”, eu vou colocar “Ok” mesmo e aqui eu coloco um “await” para ele aguardar o resultado do meu “DisplayAlert” e eu marco aqui esse método anônimo com um “async” porque eu estou utilizando “await” então eu preciso usar também o “async”.

[03:04] Agora se der falha eu tenho que fazer a mesma coisa, eu tenho que assinar uma outra mensagem pelo “MessagingCenter.Subscribe”, eu passa aqui o tipo que é “Agendamento” e eu passo a instância “this” e aqui a string que identifica esse tipo de mensagem, que é “FalhaAgendamento”.

[03:27] Aqui é a mesma coisa, vem o corpo da Action que vai ser acionada quando a mensagem chegar, eu vou ter que passar aqui “Agendamento” como um parâmetro dessa mensagem e eu vou fazer o DisplayAlert de novo para exibir a mensagem para o usuário, “DisplayAlert(“Reenviar”)” e a mensagem de erro que é “Falha ao reenviar”. Aqui o texto botão “Ok” é “Ok” mesmo, coloca aqui um “await” e aqui um “async” porque “await” exige o “async”, além disso tem que fazer o quê?

[04:13] Eu tenho também que cancelar a assinatura dessas mensagens, “MessagingCenter.Unsubscribe”, eu vou passar aqui “this” e o nome da mensagem que é “SucessoAgendamento”, a mesma coisa para falha de agendamento, vou copiar aqui, só vou trocar “FalhaAgendamento”.

[04:38] Agora vamos rodar a aplicação e ver se funciona, vou entrar aqui com o usuário “joao@alura.com.br”, senha “alura123”, vou entrar agora, vou entrar em meus agendamentos, vou clicar no agendamento que falhou, “C3 1.0” falhou, vou clicar nele, confirma o reenvio, “sim”, reenviar, “Reenvio com sucesso!”.

[05:11] Agora vamos modificar o objeto de acesso a dados do agendamento, que é o “AgendamentoDAO” e aqui no método “salvar” vamos ter que tomar uma decisão, se o agendamento já existe no banco de dados vamos ter que atualizar, se ele não existe vamos ter que inserir, como fazemos isso?

[05:31] Primeiro lugar temos que perguntar para essa conexão, para essa instância de conexão se o agendamento já existe, vamos ter que pedir para ela buscar, para ela procurar no banco de dados se já existe esse agendamento e como fazemos para mandar a conexão procurar, para ela buscar?

[05:52] Usamos o método “find” que é em inglês procurar, “find” e aqui eu passo o tipo da tabela, o nome da tabela que no caso é “Agendamento”, então eu passo o tipo do meu modelo que é “Agendamento” e eu passo nesse método um “object pk”, que é chave primária, tem que passar aqui qual é o Id do agendamento que eu estou procurando. Qual é o meu Id?

[06:17] É o “Agendamento.id”, se agendamento.id for encontrado com o método “find”, eu quero atualizar, se não for encontrado eu quero inserir, se ele não encontra o resultado desse método vai ser igual a nulo, se for “== nulo” então ele vai inserir, se ele não encontrar eu quero que ele atualize, vou colocar um “else” aqui, conexão e para atualizar eu tenho que usar um outro comando que é o “update”, então método update passando, da mesma forma eu passo o agendamento que eu quero atualizar.

[07:03] O que ele vai fazer no agendamento aqui quando eu for dar o update é modificar os campos que estão diferentes do meu modelo, eu estou passando o meu modelo e ele vai comparar com o registro que já existe no banco SQLite e vai trocar os campos que são diferentes, com exceção do Id porque ele já encontrou, então não vai ter diferença nenhuma.

[07:28] Agora vamos rodar a comunicação, vamos ver se funciona isso, vamos ver se ele atualiza o mesmo registro que estamos reenviando. Agora eu vou entrar com o usuário “joao@alura.com.br”, senha “alura123”, vou entrar, vou clicar em meus agendamentos, vamos procurar um agendamento que falhou, esse primeiro C3, vou clicar nele, confirma, vamos confirmar, reenviar, “Reenvio com sucesso”.

[07:58] Ele está mostrando a mensagem para mim, está confirmado para mim que deu sucesso e quando eu clico no “Ok”, o C3 ficou preto, ele trocou de cor, porque não foi inserido o novo registro, ele simplesmente atualizou o que já existia.

[06:17] Vamos tentar um outro aqui, esse C3 aqui também, deseja reenviar, sim, Reenvio com sucesso, também ficou preto porque ele trocou o valor da propriedade confirmado, o campo, a coluna Confirmado mudou de False pra True, ele atualizou o confirmado, salvou no banco de dados e não criou novo registro, ele simplesmente atualizou o que já existia.

[08:43] Estamos chegando ao final do nosso curso Xamarin com Visual Studio parte 2, espero que vocês tenham gostado bastante desse curso, espero que vocês façam aplicações fantásticas, multiplataforma com Xamarin. E nos vemos, vamos conversando através do fórum, outros canais, espero que vocês façam os exercícios, que é muito importante, é fundamental que vocês se pratiquem o que vimos nessas aulas, para fixar, para poder colocar em prática o que vocês aprenderam. Tudo de bom para vocês, muito obrigado e até a próxima.