

DisplayAlert

Transcrição

[00:00] Oi, pessoal, tudo bem com vocês? Eu vou Marcelo Oliveira, bem-vindos de volta ao nosso curso do Xamarin com Visual Studio. E hoje, a gente vai dar uma melhorada na nossa tabela de agendamento de test drive. Então, olha só, imagina que o nosso usuário. Ele está aqui na tela do agendamento...

[00:18] Ele começa a preencher os dados dele. Então, ele coloca lá: “fulano de tal”, o nome, e aí, ele coloca o telefone: “1234-5678”, e ele vai colocar lá no email o email dele. Agora, imagina que no momento em que ele vai escolher a data, que ele vai clicar aqui para mudar a data e aí agendar...

[00:45] Imagina que ele, sem querer, ele toque aqui no botão agendar. Então, ele tocou aqui, esbarrou o dedo no agendar e pronto, agendou. Então, imagina, ele fez o agendamento incorreto e o que acontece? Então, para a experiência do usuário ser melhor nesse caso, é bom a gente confirmar, né?

[01:11] Já que ele está enviando dados importantes para o servidor. Então é bom pedir para ele confirmar: “Você tem certeza que você está salvando mesmo esse agendamento?”, “Você quer cancelar?”, “Você tem certeza?”. Então, a gente poderia dar essa opção para ele desfazer, cancelar esse agendamento...

[01:30] Porque aí, ele se sentiria mais seguro de utilizar o nosso aplicativo. Então, como a gente faz isso? Seria legal se a gente pudesse exibir para ele uma pergunta do tipo: “Deseja mesmo enviar o agendamento?”, “Deseja enviar o agendamento?”. E aqui, com um ponto de interrogação, claro.

[01:57] E aí, com essa pergunta, a gente poderia decidir se vai realmente enviar ou não dependendo da resposta dele. Então, a gente pode dar duas opções para ele, sim ou não. Como a gente faria isso? Bom, o Xamarin Forms, ele tem esse método aqui, o DisplayAlert, que ele é muito versátil.

[02:19] Então, ao mesmo tempo em que ele permite que a gente exiba uma mensagem para o usuário, uma informação: “Olha, aconteceu tal coisa”, “Olha, parabéns, você conseguiu finalizar o seu cadastro”, sabe? Informações para ele, que não exigem nenhuma resposta, nenhum tipo de decisão, só com o botão de ok.

[02:47] A gente também pode utilizar DisplayAlert para perguntar, para fazer uma pergunta e ele poder escolher entre duas opções. Então, a gente vai fazer isso agora. A gente vai colocar DisplayAlert, e aí, aqui dentro a gente tem o quê? A gente tem o título que vai aparecer nessa caixa de mensagem, nesse diálogo.

[03:08] Então a gente vai colocar aqui no título: “Salvar Agendamento”. E aí, na mensagem é exatamente esse texto que a gente colocou aqui embaixo: “Deseja mesmo enviar o agendamento?”. E aí, eu vou ter que colocar ainda duas strings que vão ser exibidas para o botão de ok ou cancelar; ou sim ou não.

[03:34] No caso, eu vou perguntar para ele: “Sim ou não?”, “Deseja mesmo enviar o agendamento? Sim ou não?”. Então, eu coloco simplesmente uma opção “sim” e outra opção “não”. Agora, olha só, o DisplayAlert, ele é uma task, ele retorna uma task “booleana”.

[03:58] Então, como ele retorna uma task “booleana”, o que acontece? Se eu fizer a execução desse código, ele vai simplesmente emitir esse alerta, aliás, ele vai fazer essa pergunta e não vai esperar que o usuário escolha o sim ou o não, porque ele já jogou o ponto de execução aqui para baixo.

[04:22] Então, ele vai continuar rodando o aplicativo e para a gente evitar isso, a gente tem que fazer com que ele espere o usuário escolher uma opção. Então, a gente tem que utilizar o `await`. Então, com o `await`, ele vai esperar a resposta do usuário. Agora, além disso, não basta simplesmente eu mostrar essa pergunta para o usuário...

[04:48] Eu tenho que armazenar a resposta do usuário em algum lugar. Então, para isso, eu vou criar aqui uma variável, eu vou chamar de `confirma`. Então, `confirma`, se for verdadeiro, o usuário confirmou o agendamento, ele quer realmente salvar.

[05:05] Se o `confirmar` for falso, então ele não vai fazer nada, ele simplesmente vai ignorar aquela mensagem para o usuário. Então, agora eu tenho que utilizar esse “`confirma`” e fazer uma condição aqui. Então, se o `confirma` for verdadeiro, `if (confirma)`.

[05:25] Então ele vai executar esse código aqui embaixo, que é código que vai mostrar para ele o resultado do agendamento falso, aquele agendamento de mentirinha que a gente está fazendo até agora. Então, a gente vai acertar esse código aqui, isso aqui para cima.

[05:46] Bom, o Visual Studio está reclamando, porque como a gente está usando a `await`, então a gente tem que utilizar também aqui o `async`, nesse `OnAppearing`, a gente tem que utilizar aqui o modificador `async`. Então, aqui... Bom, continua dando a mensagem aqui.

[06:04] É, na verdade, esse `async` é aqui em baixo, o `async` é aqui nesse `lambda`, então o `lambda` tem que ser `async`. Isso, agora muito bem, agora está certinho. Então, o que a gente vai fazer é executar o código e a gente vai ver essa pergunta de confirmação. Então vamos lá.

[06:20] Então, agora a gente vai rodar a aplicação e ver se a mensagem, se a pergunta de confirmação aparece para o usuário corretamente. Então vamos lá, rodando. Agora vai rodar o aplicativo. Vou escolher aqui um veículo, vou clicar no próximo, vou colocar: “fulano de tal”, telefone: “1234-5678”.

[06:56] E aí, eu coloco o email: “fulano@gmail.com”. Ok. E aí, na hora de colocar o agendamento, na hora de colocar a data e a hora, sem querer, esbarra o dedo no agendar, só que dessa vez, olha só: “Deseja mesmo enviar o agendamento?”, não, né? Porque eu toquei sem querer no agendamento.

[07:18] Então, eu vou colocar “não”. Então, ele não fez nada, ele simplesmente ignorou que eu toquei nesse botão agendar. Agora, eu toco de novo, aí, eu arrumo aqui o agendamento, coloco dia 15 e a hora, eu coloco quatro da tarde, por exemplo, dou o ok. Agora sim, eu quero agendar.

[07:38] Então, eu clico em agendar: “Deseja mesmo enviar o agendamento?”, “Sim”, eu confirmo. Então, agora, ele enviou a mensagem porque... Aliás, ele enviou o agendamento porque eu confirmei que eu quero mesmo fazer isso. Então, com isso, a gente conseguiu fazer um ciclo aqui...

[07:59] Que protege o usuário de fazer uma besteira mais séria, porque ele sem querer pode tocar num botão, ele pode fazer alguma coisa que é importante para o sistema, mas que ele tem que ter ciência, ele tem que ter uma noção do que ele está fazendo, para ele não fazer besteira.

[08:17] Então é isso daí. Agora, a gente vai passar para a próxima parte que é como fazer o preenchimento correto do formulário, para enviar para um serviço real, que a gente vai enviar os dados do agendamento para o test drive.