

01

AgendamentosUsuario Detail

Transcrição

[00:00] Bem vindos de volta ao nosso curso de Xamarin com Visual Studio, agora vamos dar uma olhada nas funcionalidades que nosso aplicativo fornece para o usuário.

[00:11] O nosso usuário está conseguindo utilizar o aplicativo de TestDrive para agendar um veículo, escolher os acessórios e depois marcar uma data uma hora para agendamento, eu vou fazer isso agora, vou marcar o dia 6 de fevereiro e aqui no horário vou colocar meio-dia e vou agendar, quando eu confirmo o agendamento ele envia as informações lá para o servidor da Aluracar e eu volto de novo pra ter Inicial, só que em nenhum momento eu sei que data e horário e qual veículo eu já agendei, o que precisamos fazer?

[00:53] Temos que exibir para o usuário quais agendamentos estão gravados no banco local do SQLite, para isso vamos criar uma nova tela de agendamento do usuário para exibir todos os agendamentos que já foram realizados por ele, nessa tela vamos exibir o quê?

[01:09] A data, a hora e o veículo do agendamento, para chegar nessa tela vamos ter dois caminhos, o primeiro vai ser através de um botão que vamos criar na tela de perfil, logo abaixo do botão perfil vamos ter meus agendamentos, clicando aqui vamos levar o nosso usuário lá para os agendamentos que ele já realizou.

[01:35] A outra forma, o segundo caminho vai ser, logo após salvar um agendamento vamos voltar e vamos exibir para ele quais foram os agendamentos realizados.

[01:45] O primeiro passo aqui é criar esse botão meus agendamentos, vamos ter que entrar nessa View, que é a MasterView que é o arquivo MasterView.xaml, vamos entrar aqui e vamos procurar onde colocar esse novo botão, aqui logo após o botão perfil criamos um novo botão então “Button”, vamos dar o texto aqui de “Meus Agendamentos”, agora vamos até a ViewModel da MasterView, que é a MasterViewModel e vamos implementar o comando desse botão.

[02:28] Aqui embaixo colocamos uma nova propriedade chamada de “MeusAgendamentosCommand”, eu vou colocar o private set porque eu não quero que essa propriedade seja alterada de fora da classe, agora está privada. Agora está certo, MeusAgendamentosCommand, o que eu preciso fazer agora é criar uma Instância nova do MeusAgendamentosCommand, quando entrarmos no Construtor do MasterViewModel.

[03:05] Aqui dentro de temos que definir comandos, chamamos o método definir comandos, que vai definir vários comandos, inclusive o novo comando que é o meu MeusAgendamentosCommand, “MeusAgendamentosCommand =” ao novo comando e eu vou passar no Construtor dessa classe.

[03:28] Uma Action que vai ser receber, que vai ser executada quando o usuário clicar no botão, dentro eu vou colocar essa Action e o que eu quero fazer aqui é trocar a View, eu quero trocar a visualização para exibir no lugar da listagem de veículos disponíveis para agendamento, eu quero exibir aquilo que já foi agendado.

[03:52] Estamos em qual contexto? Estamos na tela, na MasterView, que é filha, a MasterView ela é a Master da MasterDetail, é a MasterDetailView que vai poder trocar qual é a nossa página de detalhe, que no momento é somente a listagem View, vamos ter que trocar lá na MasterDetailView qual é o detalhe, não vai ser mais ListagemView, vai ter que ser a nova página que é MeusAgendamentos.

[04:27] Precisamos avisar lá em cima a MasterDetailView que houve essa mudança, que o usuário clicou no botão e que ele quer trocar o detalhe, a página de detalhe que vai ser a página de meus agendamentos.

[04:42] Para gente avisar a página de MasterDetail a partir da página MasterView, temos que fazer o quê? Temos que lançar aqui uma nova mensagem avisando a MasterDetailView, na MasterViewModel eu vou lançar uma mensagem nova e como eu lanço uma mensagem dentro do Xamarin Forms?

[05:04] Utilizando o serviço de mensagem que é o MessagingCenter, eu vou enviar mensagem comando “send” que é em inglês “Enviar”, aqui eu vou passar um tipo que vai ser o objeto que vai estar anexo nessa mensagem, que nesse caso eu vou passar o usuário, então “Usuario” e aqui dentro eu faço o que?

[05:28] Eu passo a instância do usuário, “usuario” e aqui o nome que identifica esse tipo de mensagem, que eu vou chamar “MeusAgendamentos”, já estamos com esse código enviando lá para algum lugar do sistema, estamos avisando, o usuário está clicando em meus agendamentos, agora precisamos fazer o quê?

[05:53] Interceptar essa mensagem lá na MasterDetailView para ela poder trocar o detalhe, poder trocar a página que está exibindo o detalhe que atualmente é a ListagemView, vamos trocar, vamos colocar no lugar dela a nossa nova página de meus agendamentos.

[06:11] Agora aproveitando que estamos aqui vamos refatorar esse código, tem um código aqui que não faz parte desse método “DefinirComandos”, esse é um método, é um código que está assinando uma mensagem, eu vou remover daqui, vou colocar aqui no construtor esse método e eu vou remover novamente e vou extrair um novo método chamado assinar mensagens.

[06:36] Eu clico com o direito, Quick Actions and Refactorings, extrair método e agora sim “AssinarMensagens”, agora refatorando ficou melhor o código. Vamos agora realmente assinar essa mensagem “MeusAgendamentos” lá no MasterDetailView, “MasterDetailView.xaml” vamos abrir aqui o Code Behind do MasterDetailView.xaml.cs e vamos assinar como?

[07:13] Vamos assinar no método “OnAppearing”, vamos procurar o “override OnAppearing”, já apareceu aqui, vamos fazer a assinatura da mensagem, qual é a mensagem? “MeusAgendamentos”, para assinar a mensagem eu tenho que utilizar o serviço de mensagens Xamarin Forms e o serviço de mensagens é o “MessagingCenter” e o método para assinar que é “Subscribe”, que em inglês Subscribe é assinar.

[07:48] Aqui vou passar qual é o tipo do dado que está sendo passado na mensagem, que é o usuário, o tipo “usuário” e aqui eu passo o quê? Em primeiro lugar nessa lista de parâmetros eu passei a instância que está assinando, que é a própria classe, que é o próprio caso do Code Behind, que é a Instância “this” e em seguida eu passo qual é a string que identifica a mensagem.

[08:13] Essa mensagem é do tipo “MeusAgendamentos”, em seguida eu passo a Action, eu passo o método anônimo que vai ser executado quando o usuário clicar no botão e recebermos a mensagem de MeusAgendamentos, eu passo aqui uma expressão lambda que vai receber o quê?

[08:36] Uma Instância de usuário, “usuario” e recebendo o usuário eu vou lá dentro colocar o código para efetivamente trocar de página, eu vou ter que trocar qual é a página de detalhe que está sendo exibido nesse MasterDetail, ele tem dois componentes, que a Master e o Detail, que é o mestre e o detalhe, no detalhe vamos trocar esse detalhe pela nova página de meus agendamentos, como eu troco o Detail?

[09:08] Eu acesso “this.Detail”, o detalhe vai ser o quê? Vai ser uma nova instância da página que vamos criar, a página de agendamento, “new” e aqui vou colocar a página de agendamento do usuário, que ainda não criamos, vamos fazer isso agora.

[09:27] Vamos criar aqui dentro de Views, vamos criar o novo item que vai ser uma página muito simples, vai ser uma página content page, eu selecionei aqui Forms Xaml Page e aqui dentro eu coloco no lugar do nome, eu coloco

“AgendamentosUsuario”, que vai exibir os meus agendamentos, eu clico no adicionar e automaticamente dentro da pasta Views ele está criando aqui um Content Page, vai criar uma página simples, que só contém um Label, eu vou instanciar aqui.

[10:08] No “this.Detail” eu vou atribuir uma nova instância de “AgendamentosUsuario”, eu acabei de criar, mas eu vi que tá fora do padrão, eu não coloquei o sufixo “View”, então eu vou trocar, “AgendamentosUsuario” vou colocar aqui “View”, eu preciso trocar também aqui dentro do xaml, o “x:Class” eu preciso trocar para “AgendamentosUsuarioView”, aqui no Code Behind vamos dar uma olhada, aqui também “public partial class AgendamentosUsuarioView” e também trocar aqui qual é o nome do método do construtor.

[10:51] Troquei aqui, agora voltamos lá para o MasterDetailView e vamos criar uma nova instancia do AgendamentoUsuárioView e aqui já definimos isso como detalhe, quando o usuário clicar no botão nós vamos esperar que a nova página de detalhe seja trocada, seja substituída por uma nova instância da página nova de agendamento de usuário.

[11:17] Agora falta mais uma coisa, faltou fazermos o Binding do comando lá no botão de MeusAgendamentos lá na MasterView, vamos entrar em MasterView.xaml e vamos colocar aqui no botão de MeusAgendamentos, vamos colocar o comando e fazer aqui o Binding para o comando que é “MeusAgendamentosCommand”.

[11:47] Agora vamos rodar aplicação e vamos ver se a nova página aparece na tela, agora vou entrar com o usuário “joao@alura.com.br”, senha “alura123”, vou entrar, agora eu clico no menu e vou lá para o botão Meus Agendamentos, quando eu clico aqui eu espero que eu consiga ver a nova página de MeusAgendamentos, clicando agora, não aconteceu nada, porque será que não apareceu a página?

[12:19] Lembramos que a página MasterDetail tem dois componentes, ela tem a página Master e tem a página Detail, o que estamos vendo aqui na tela é só a página Master, a página Detail está meio escondido aqui atrás, se eu clico nessa área cinza eu vou fazer o quê?

[12:39] Eu vou fechar a página Master e exibir a página Detail, se eu clico aqui, essa é a página de meus agendamentos, conseguimos trocar a página Detail, a página de detalhe pela nova página de agendamentos, só que ainda tem um inconveniente, porque eu tive que clicar naquela área cinza, eu não queria isso, eu queria clicar no botão e já aparecesse na tela, aberto para mim a página de Detalhe, como podemos fazer isso?

[13:13] Lá no MasterDetailView eu tenho aqui o comando que vai trocar a tela, eu tenho o “this.Detail” e “settando” o detalhe para a nova página, para uma nova instância da página de agendamento de usuário, eu tenho que dizer para minha MasterDetail, se a minha página Master, que é a página-mestre, é apresentada ou não, como que eu faço isso? Eu vou ter que acessar uma propriedade chamada “isPresented” significa página-mestre, é apresentada ou não.

[13:52] Quando o usuário quer ir para página de meus agendamentos, ele quer que a página mestre, que é a página de menu, não seja apresentada, “this.isPresented = false”, vamos rodar agora em ver qual é o comportamento da nossa aplicação. Vou colocar o usuário “joao@alura.com.br”, senha “alura123” e vou entrar, clicando no menu, vou clicar em meus agendamentos, com isso conseguimos navegar corretamente para nossa página de Meus Atendimento.