

04

Refactoring

Transcrição

[00:00] Agora vamos começar a mexer no layout da tela de agendamento do usuário, em primeiro lugar não temos aqui uma margem para essa listagem agendamento, não temos margem nenhuma. O que precisamos fazer é parar o programa, entrar aqui em ListView e definir a propriedade “Margin” dele para um valor, eu vou colocar 25 que é o valor que já estamos usando em outras telas.

[00:30] Outra coisa que precisamos fazer é adicionar aqui no template, para cada linha de agendamento eu quero exibir, não só modelo, como estamos fazendo agora, mas também a data e a hora do agendamento, porque isso vai orientar o usuário e ele vai saber se ele já marcou algum TestDrive para aquela data, para aquela hora.

[00:54] Além de informar o modelo eu tenho que informar também a data e a hora, eu vou além do Label de “Modelo” eu vou adicionar um outro Label aqui em cima, que vou colocar o texto dele, vou ter que puxar de onde? Vou ter que puxar da data do agendamento, DataAgendamento.

[01:19] Só que esses dois Labels estão dentro de uma View e ela não pode conter múltiplos elementos, eu preciso colocar um container para esses dois Labels, eu preciso de um controle que permita que eu consiga empilhar esses Labels horizontalmente e o componente para empilhar controles é o StackLayout e eu vou colocar a orientação dele como horizontal, “Orientation=“Horizontal”” e vou colocar lá dentro os dois Labels.

[01:55] Só que entre um Label e o outro, entre a data e o modelo do carro, eu quero colocar um separador, eu vou colocar um Label só para separar esses dois textos, “Label” e o texto dele vai ser um espaço em branco, um traço e o espaço em branco, agora voltamos lá no Code Behind e vai adicionar algumas datas e horas para esses agendamentos.

[02:20] Vamos em AgendamentosUsuariosView.xaml.cs e vamos colocar aqui para cada agendamento, colocamos também a data e hora, mas não temos aqui no construtor de agendamento a data e a hora, então precisamos adicionar essa informação. Eu vou em agendamento, vamos na definição de agendamento, eu vou modificar esse construtor para suportar também a data e a hora.

[02:48] Além do preço eu coloco, “DateTime dataAgendamento” e para o horaAgendamento eu coloco um “TimeSpan horaAgendamento”, agora para “settar” os valores da instância eu faço “this.DataAgendamento =” ao parâmetro que está sendo passado no construtor que é “dataAgendamento”, o “this.HoraAgendamento” vai receber também outro parâmetro que é “horaAgendamento”.

[03:25] Agora voltamos para o Code Behind e adicionamos algumas datas e horas para poder formar uma lista, para o primeiro agendamento eu faço “new DateTime”, aqui eu tenho que colocar o ano, “2017”, vamos colocar o mês aqui, eu vou colocar um mês dois, o dia eu vou colocar “15” e agora vou colocar o horário, “new TimeSpan”, aqui eu coloco horas, eu vou colocar “16” horas, “30” minutos, “0” segundos e vou fazer o mesmo para os outros agendamentos.

[04:03] Vou trocar para esse aqui também, só que esse vai ser dia 16 e aqui vou colocar 05 da tarde, então 17 horas e também fazer para o outro agendamento também, vou colocar dia 17, meio-dia e meia, agora vamos rodar a aplicação, apareceram alguns erros aqui.

[04:32] Como mudamos o construtor, acabou estourando alguns erros porque temos que passar também, aqui na “AgendamentoViewModel” tem que passar também data hora do agendamento, só que pelo que estou olhando aqui não podemos modificar essa chamada desse agendamento para incluir o “dataAgendamento” e “horaAgendamento” porque não temos essas informações ainda, quando essa linha é chamada, quando estou instanciando novo agendamento.

[05:04] O que eu preciso fazer aqui é ter dois construtores e fazer uma sobrecarga de construtor, eu vou entrar lá em agendamento e fazer essa sobrecarga, como eu faço isso? Eu crio um novo construtor e eu vou passar só os parâmetros que eu estou utilizando, que são os parâmetros básicos desse construtor, sem a data sem a hora.

[05:31] Em seguida eu coloco nesse Construtor, eu coloco só atribuição desses parâmetros básicos, dessas propriedades básicas, fazendo isso agora e agora no construtor mais completo eu coloco uma referência, faço uma sobrecarga passando os parâmetros básicos do construtor, que é o construtor mais simples, estou passando aqui o quê?

[05:57] O nome, o fone, o e-mail, modelo e o preço, com isso ele vai chamar o construtor mais simples e em seguida vai também “settar” a data e a hora do agendamento, agora rodamos a aplicação e vemos se funciona.

[06:11] Vou colocar o usuário “joao@alura.com.br”, a senha “alura123”, vou entrar, vou para o menu inicial e vou clicar em agendamentos, agora apareceu aqui a data/hora e o modelo do carro, só que não estamos exibindo ainda a hora corretamente, está tudo zero, porque isso está acontecendo?

[06:41] Porque estamos exibindo só a data, ou melhor, a data não contém a hora porque são duas propriedades separadas, o que eu vou fazer é modificar o nosso agendamento, para poder conter uma data formatada, eu vou parar a aplicação, vou entrar aqui em agendamento e vou criar uma nova propriedade, vou chamar essa propriedade de data formatada.

[07:10] Vai ser um DateTime, eu vou chamar de “DataFormatada”, eu só vou ter o “get” e dentro do “get” eu vou retornar a data no formato que eu, então eu retorno data de agendamento, “DataAgendamento” só que eu adiciono nessa data agendamento o quê?

[07:33] A “HoraAgendamento”, adiciono “HoraAgendamento” e no final eu quero retornar uma string com a data e hora formatadas, eu faço isso com o “ToString” e aqui eu faço o formato “dd/MM/yyyy”, em seguida eu coloco a hora em maiúsculo porque eu quero em formato 24 horas e vem minutos, “m” minúsculo. E vou trocar o DateTime por string, vou formatar o documento.

[08:21] Agora eu pego a data formatada e vou colocar lá no binding do nosso View, aqui no “AgendamentoUsuárioView” eu vou colocar no lugar de DataAgendamento eu coloco DataFormatada.

[08:33] Agora rodamos a aplicação e vê o resultado, eu entro com o usuário “joao@alura.com.br”, a senha ”alura123”, vou entrar agora e lá no menu eu vou clicar em meus agendamentos e eu chego na listagem, só tenho aqui data 15 de fevereiro, a hora 4:30 da tarde, 16 de Fevereiro, 5:30 da tarde, 17 de Fevereiro, meio-dia e meia e cada um mostrando um veículo diferente.

[09:06] Com isso conseguimos mostrar de uma forma apresentável a tela de meus agendamentos, agora falta formatar para deixar um pouco mais visível porque está um pouco apagado aqui esse texto, eu quero aumentar a fonte da data e da hora e também deixar em negrito o campo do veículo, a coluna de veículos.

[09:29] Eu vou modificar o FontSize do “DataFormatada”, “FontSize” eu vou colocar como 20 e aqui no modelo eu vou modificar o tipo da fonte para colocar ele como negrito, como eu faço para deixar em negrito? Eu modifio os atributos da fonte, “FontAttributes” e coloco Bold.

[09:54] Agora eu quero alinhar esses três Labels para ficarem centralizados verticalmente dentro de cada linha do nosso ListView. Eu vou lá em VerticalOptions e coloco “Center” e faço isso para cada uma das três Labels.

[10:11] Agora vamos rodar a aplicação e ver como isso é exibido na tela, vou entrar com o usuário “joao@alura.com.br”, a senha “alura123”, vou entrar, vou clicar lá no menu, agora vou clicar em agendamentos, agora ficou certinho, a exibição está perfeita do agendamento do usuário.

