### Aula 3 - Revisão e exercício Calculadora

|  |  |
| --- | --- |
| **Curso: Desenvolvimento Mobile iOS** | **Unidade e Módulo: iOS Core/Controles** |
| **Matéria da aula: Revisão** | **Duração da aula: 3 horas e 30 minutos  (180 minutos líquidos)** |
| **Link:** <https://drive.google.com/open?id=0B5MhKet-B_aUcXpfaVVjUmZnaXM> | |

|  |
| --- |
| **OBJETIVOS** |
| * Os alunos devem aprender a usar TextFields e entender as diferentes características que permitem modificá-los. * O aluno deve entender a função dos delegates nos diferentes componentes, especialmente com o TextField. * Fazer um exercício de “integração” onde todos os conceitos vistos até agora sejam aplicados, criando o primeiro aplicativo realmente útil: uma calculadora. * O aluno deve começar a se familiarizar com as técnicas de depuração e as ferramentas disponíveis para isso. |

|  |
| --- |
| **CONTEÚDOS** |
| 1. UITextField 2. Delegates 3. Depuração |

|  |  |
| --- | --- |
| **Início da aula** |  |
| **Como os objetivos serão apresentados:**  Os objetivos serão apresentados de maneira explícita. Enfatizar que os alunos já estão em condições de implementar aplicativos interativos de tela única. Para isso, serão revistos os conceitos importantes e novas matérias serão introduzidas.  É esperado que os alunos entendam o uso do UITextField e possam fazer o exercício da calculadora para que possam começar a entender sobre depuração. | **15 minutos** |
| **Atividade ou pergunta motivadora:**  Apresentar a calculadora de iOS e comentar com os alunos que eles construirão um aplicativo igual ao final da aula. Antes disso, fazer um intervalo: fazer algum dos exercícios da aula anterior no qual o usuário insere um valor usando um UISlider. Buscamos melhorias e, inevitavelmente, os alunos vão propor o uso de um campo de texto. |

|  |  |
| --- | --- |
| **Desenvolvimento da aula** | **Duração** |
| **Introdução a conteúdo novo** | |
| **Matéria vista na aula anterior:**  Introdução a projetos iOS. Outlets e Actions. | **45 minutos** |
| **Matéria:**  UITextfield. |
| **Resumo:**  Depois de algumas aulas, os alunos já conhecem vários componentes e podem gerenciá-los usando outlets e actions. Criar uma tela com um textfield, um botão e um label. Quando botão é pressionado, o label dá um cumprimento ao usuário, que já preencheu seu nome previamente.  Explicar as diferentes formas de configuração, em especial, tudo que for relacionado ao teclado (preditivo, tipo de teclado, entrada segura).  Enfatizar que o teclado não desaparece sozinho e a visualização que está atrás é perdida temporariamente. Mais à frente será mostrado como resolver este problema.  Em seguida, retirar o botão e fazer com que a saudação seja dinâmica. Portanto, é necessário introduzir o conceito de DELEGATE, lembrando o conceito de observer visto nas aulas anteriores. Ele mostra como atribuir o viewcontroller como delegate (a partir do storyboard ou por código), enfatizando que o viewcontroller deve implementar o protocolo (lembrar a aula de protocolos) e explicar os diferentes métodos disponíveis no UITextFieldDelegate.  Finalizar comentando que todos os componentes vistos são “filhos” de UIView, e se disponibiliza um [arquivo curto](https://drive.google.com/open?id=1FWxTX85XduAR5LSxrE7C6fmyg7OYkBvDxAtedVddGo4) com as principais propriedades da aula, que, portanto, é possível acessar e modificar todos os componentes. |
| **Prática longa** | |
| **Resumo:**  Apresentar a calculadora de iOS novamente. Discutir como criar o projeto, os diferentes componentes a serem usados e as características de configuração de cada um.  Enfatizar que Textfield NÃO deve ser usado, pois foi visto recentemente.  Enfatizar também que o projeto NÃO deve ser perfeito. A ideia é começar com algo semelhante e funcional. Depois, se houver tempo, pode-se tentar aproximá-lo do projeto final; por exemplo, fazer os botões redondos (o conceito de layer deve ser apresentado ou usar botões quadrados como a versão do iOS 10) e um botão que muda de cor ao ser tocado (o conceito de estados para os botões deve ser apresentado).  **Aos 25 minutos**: checkpoint de projeto. Disponibilizar um projeto já criado aos alunos que não conseguiram terminar.  Depois de criar o projeto, os comportamentos devem ser adicionados. Discutir as diferenças entre comportamento de botões com números e botões com operações. Começar com os números, como fazê-los aparecer no rótulo e nas diferentes alternativas: uma ação para cada botão ou uma ação para todos os botões do mesmo tipo.  **20 minutos depois:** checkpoint de números aparecendo no label.  **Dar 15 minutos** para que terminem a interação de números e pensar como apresentar as operações.  **Sobram 30 minutos para encerrar o exercício:**  Provavelmente muitos alunos tenham dificuldade com a maneira de propor a separação em termos e acabem usando muitos if-else para reconhecer a operação.  Eles devem ser orientados a pensar de uma maneira orientada a objetos, definindo uma hierarquia de “operações”, de modo que quando o usuário toca em uma operação o objeto correspondente é criado e então a operação correspondente é simplesmente executada. | **90 minutos** |
| **Reflexão sobre o que foi trabalhado na prática:**  Discutir sobre as diferentes soluções apresentadas.  Buscamos melhorias para o projeto: não poder inserir “0001”, lidar com resultados com vírgula, não mostrar resultados com vírgula se o usuário não puder usá-los, etc. |

|  |  |
| --- | --- |
| **Encerramento da aula** | **Duração** |
| **Atividade de encerramento:**  Fazer uma revisão de todos os componentes vistos e quando convém usar cada um. Enfatizar os problemas que tenham surgido durante as aulas do módulo (botões redondos, gerenciamento do teclado, etc.) | **15 minutos** |