

Introdução ao Desenvolvimento iPhone

01 - Introdução



Sumário ^[1/2]

- Introdução ao módulo;
- Breve revisão do OS X;
- Introdução à *App Store*:
 - Acesso a aplicações e regras gerais;
 - Contas “Developer”;
 - Recursos Disponíveis;
 - Requisitos para desenvolvimento em iOS;

Sumário [2/2]

- Xcode:
 - Interface;
 - Criação e organização de projectos;
 - Simulação e *debug*;
 - Ferramentas comuns;
- Desenhar APPs:
 - Estados das APPs e *Event Loop*;
 - *Storyboards* e XIBs (NIBs);
 - MVC e *Delegates*, *Actions* e *Outlets*;

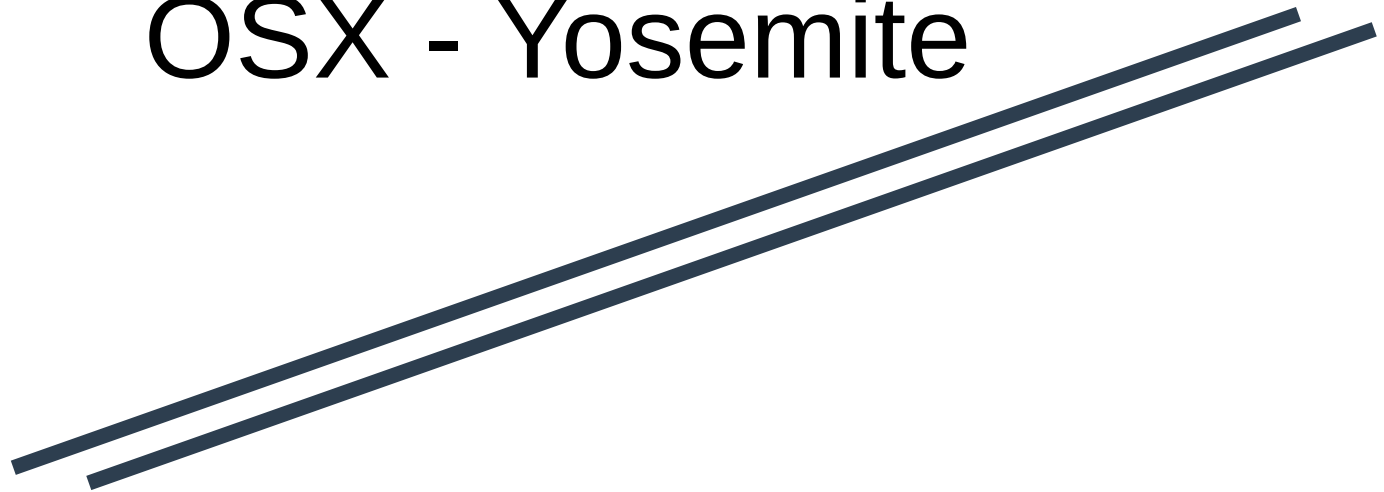
Introdução ao Desenvolvimento iPhone



Introdução ao Desenv. iPhone

- Ferramentas e requisitos;
- Basear em projectos;
- Tecnologias base (Objective-C, UIKit, ...);
- Material (exemplos e PDFs) disponível:
<https://bitbucket.org/Knitter/sok-objectivec>
<https://www.wuala.com/Knitter/sok-objectivec>

OSX - Yosemite



OS X (e iOS)

- Design, integração, controlo:
 - Guias de desenvolvimento e de usabilidade (mais que em Windows);
 - OS e Hardware controlado pela Apple;
 - Distribuição de software controlada (maioria);

Interface Gráfica

- Menu da aplicação sempre no topo;
- Fechar, minimizar, "maximizar horizontal"/fullscreen;
- Finder (explorador de ficheiros) sempre “aberto”;
 - "Enter" editar nome de ficheiro;
 - espaço pré-visualização de ficheiro;

Aplicações

- Pastas com terminação “.app”
- Presentes na pasta “/Aplicações” (/Applications), ou na correspondente do utilizador;
- Arrastar para (des)instalar;

APP Store



Plataforma

- Único método de distribuição de APPs;
- Integração com iOS, OSX e iTunes (Windows);
- Criação (e controlo) de mercado;
- Divisão de receitas entre criador de aplicações e fornecedor de infra-estrutura;
- Conteúdo além de APPs;

Acesso

- Utilização requer “AppleID”; funcionalidades diferentes por tipo de serviços subscritos;
- Registo de conta em <https://appleid.apple.com/account>;
- Gratuito para aceder acesso a Developer Center;
- Requer subscrição de programa pago para publicar aplicações ou testar nos dispositivos;

Contas

- Gratuita (Apple ID apenas), acesso como programador;
- Individual:
 - Criar e publicar aplicações;
 - Até 100 dispositivos de teste;
- Empresa:
 - Distribuir APPs internamente (exclusivo para funcionários da empresa);

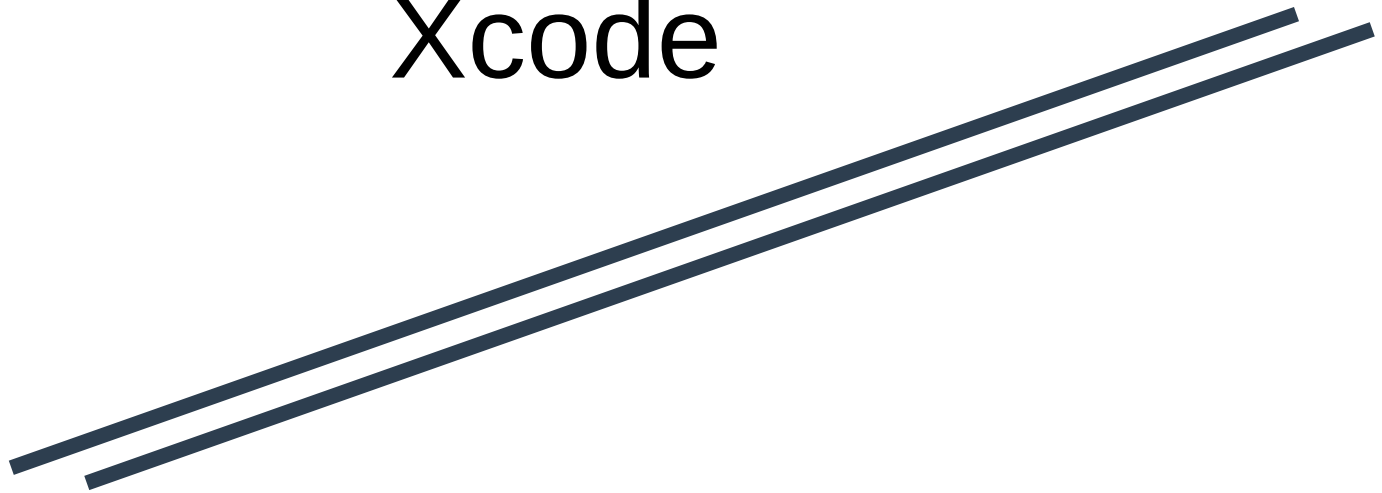
Recursos

- Xcode e ferramentas de desenvolvimento;
- Developer center para documentação (todos os vídeos dos eventos oficiais), suporte e fóruns;
- Certificados, APP IDs e Perfis;
- iTunes Connect para publicação de aplicações;

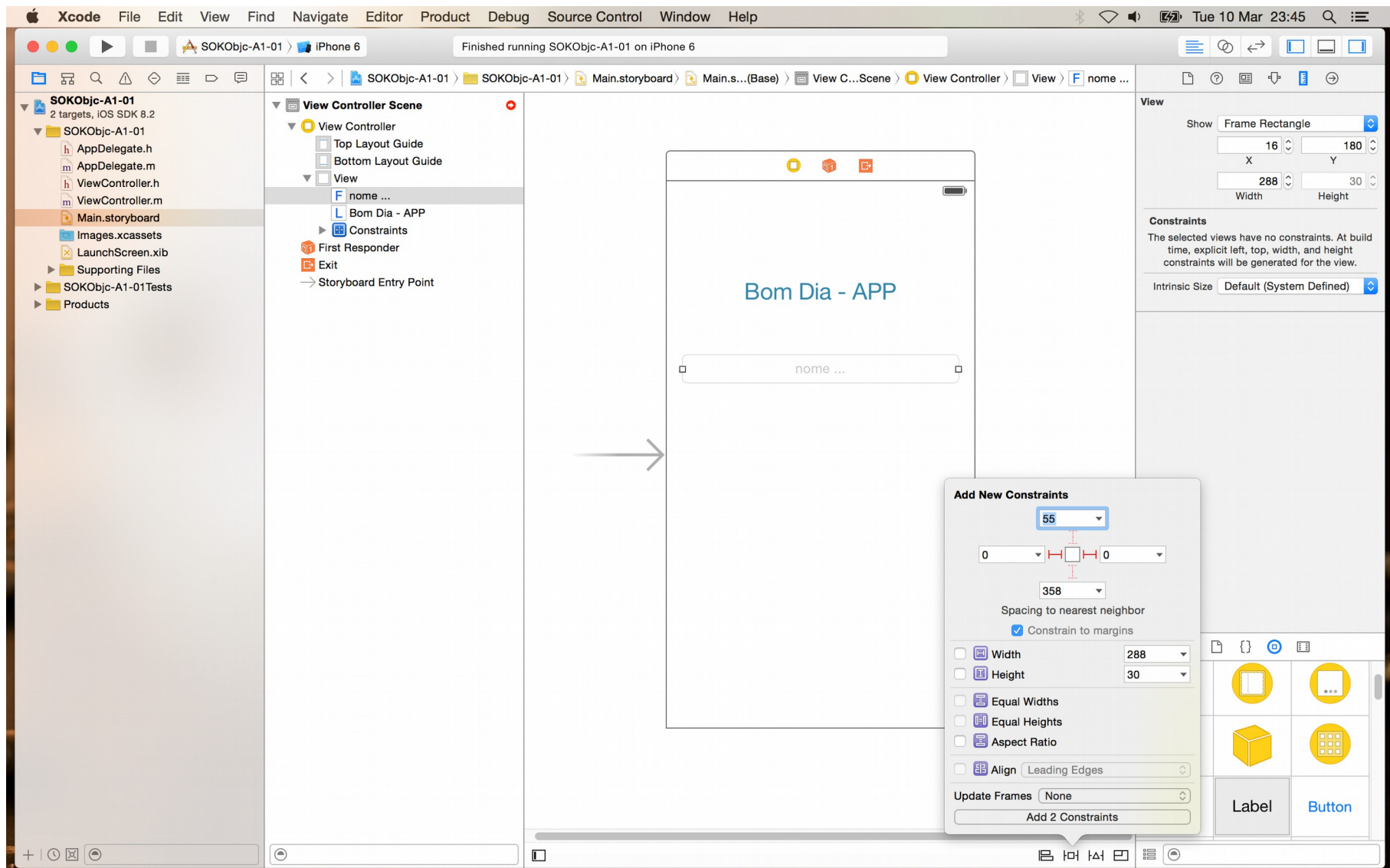
Requisitos (relembrar)

- Computar Apple com OS X recente;
- iPhone, iPod Touch, iPad, Apple Watch;
- Conta de developer activa;
- Configuração de cada APP em desenvolvimento para permitir testes;

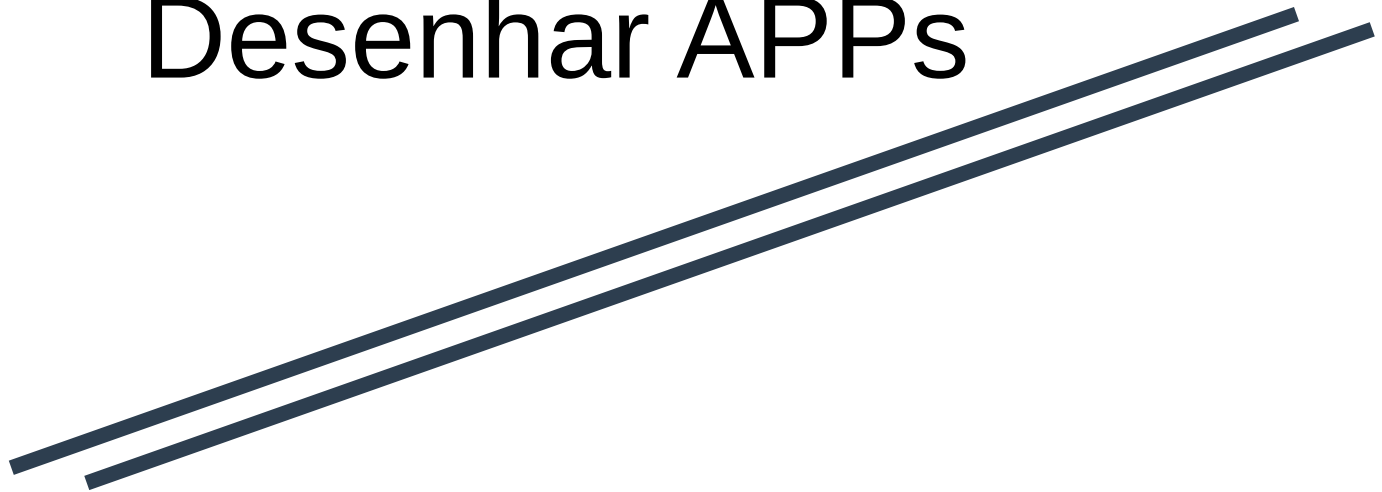
Xcode



Xcode



Desenhar APPs



- Design plano, cores a definir a interface;
- Uso de todo o ecrã com elementos translúcidos;
- Uso de tipos de letra do sistema, botões sem linhas (borders);
- Preferir disposição de conteúdo em camadas;

- Aplicações devem iniciar “imediatamente”, reduzir ecrãs de carregamento ou “splash screens”;
- Reduzir pedidos de informação e preferir logins e registos na última etapa;
- Manter estado entre execuções; recuperar o estado da aplicação;

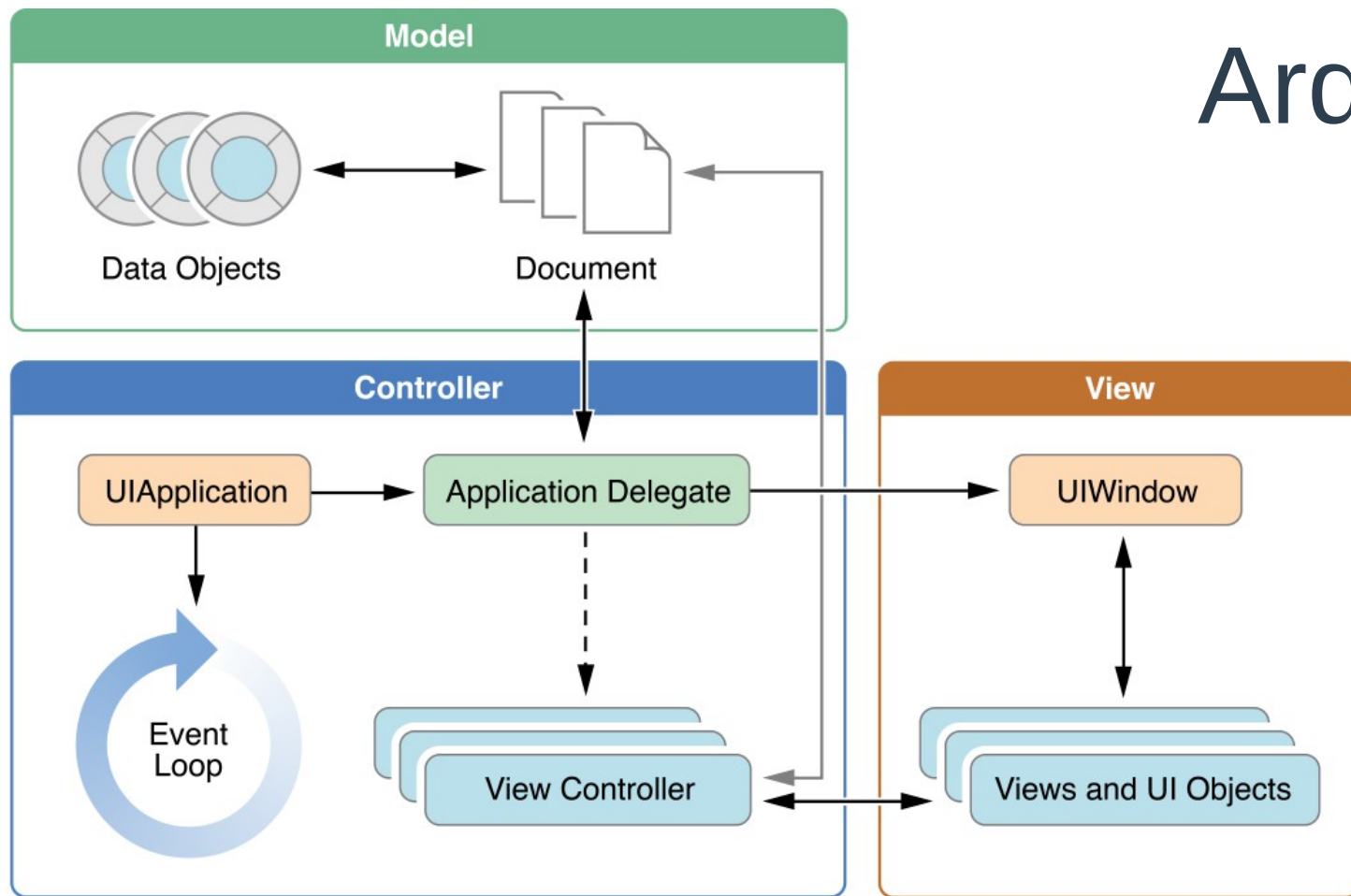
- Dispor elementos e funcionalidades por importância: mais importantes no canto superior esquerdo;
- Tamanho da letra que permita boa leitura mas que permita ajustes automáticos (controlados pelo utilizador)
- Execução de código da aplicação;
- Gestão de memória;
- Fim (início);

- Situar o utilizador, garantido um único caminho para cada ecrã;
- Respeitar as escolhas dos utilizadores (notificações, alertas, etc.);
- Usar os gestos esperados/conhecidos pelos utilizadores;

- Consultar:
<https://developer.apple.com/library/ios/documentation/UserExperience/Conceptual/MobileHIG/>
- Não cumprir as regras poderá implicar rejeição da APP;
- Entender as regras e o seu contexto permite-nos quebrá-las se necessário para uma boa experiência do utilizador;

- iOS 7 quebra a interface anterior (iOS4/5/6);
- Pouca ou nenhuma retro-compatibilidade de IU;
- Se for necessário manter versões anteriores, é mais simples criar duas versões (se possível);

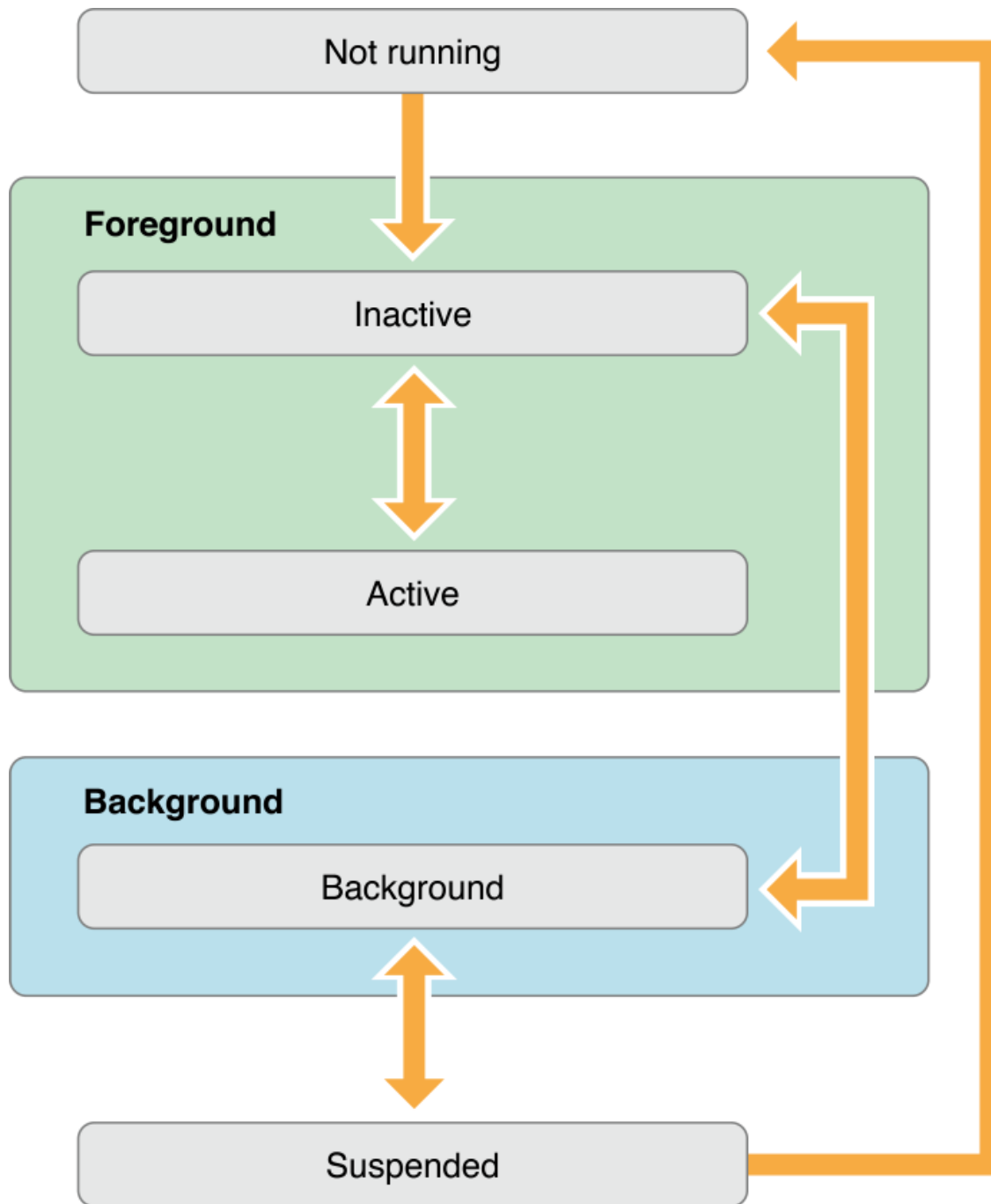
Arq. Geral



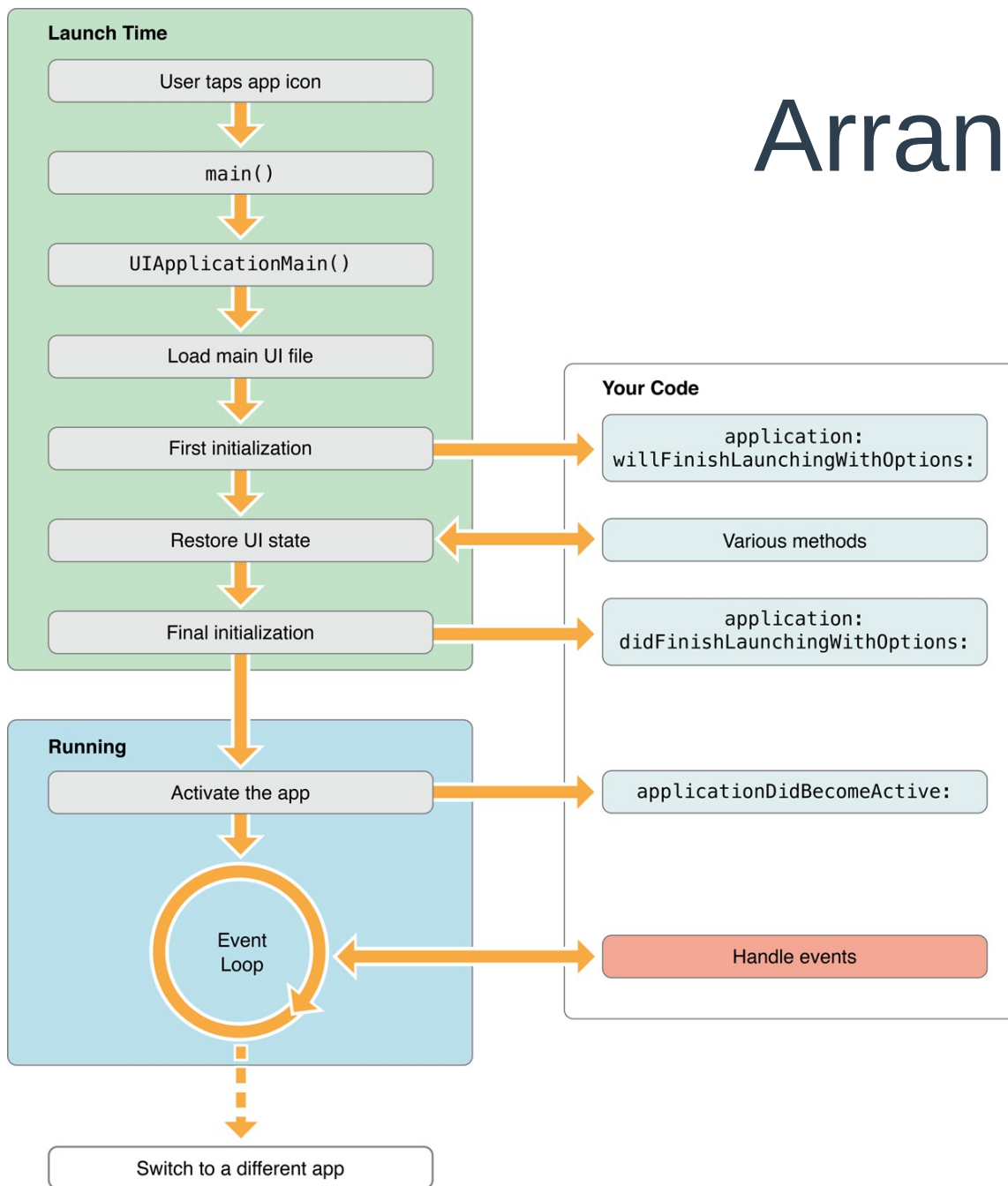
Estados de Execução [1/2]

- Parada: Nunca iniciou ou foi terminada pelo sistema;
- Inactiva: Não recebe eventos mas pode executar código; tipicamente um estado de transição;
- Activa: Estado normal de utilização;
- Background: Continua a executar código; poder manter-se neste estado durante algum tempo;
- Suspensa: Não executa qualquer código; não há aviso de suspensão; mantém-se em memória até que o sistema decida;

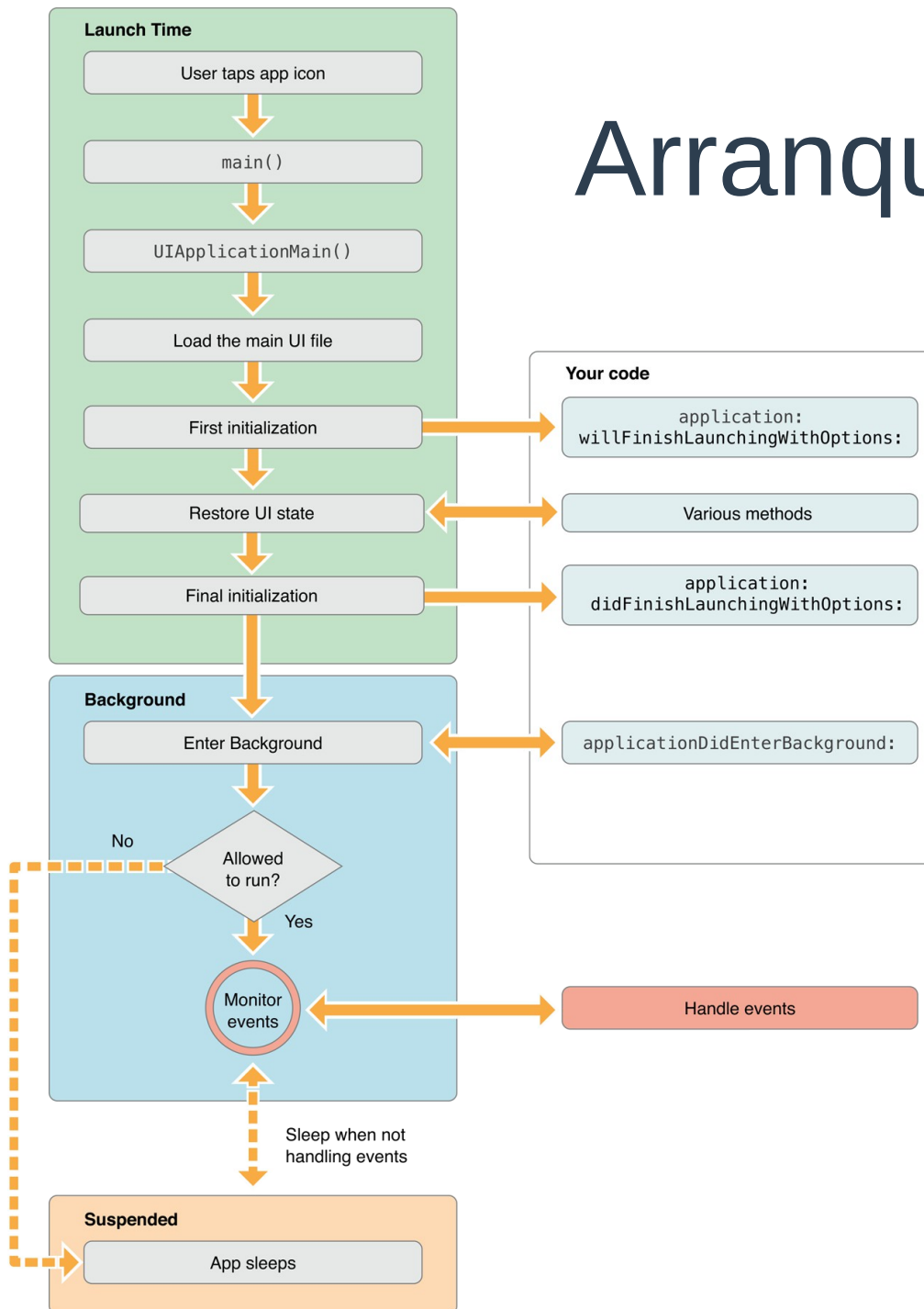
Est. de Ex. [2/2]



Arranque Normal



Arranque p/ Background



Event Loop [2/2]

- Início do ciclo;
- Tratamento de eventos (toques, notificações, etc.)
- Execução de código da aplicação;
- Gestão de memória;
- Fim (início);

Não Controlamos

- O ciclo de vida ou o Event Loop;
- O tempo em que a aplicação se mantém activa;
- Alterações entre estado activo e suspenso (ou qualquer outro estado);
- Só controlamos o código que executa enquanto a aplicação está activa (quase)!

Storyboard

- Introduzido no iOS5;
- Desenhar interface gráfica de forma visual com noção de fluxo de interacção;
- Permite combinar método antigo (XIBs);
- Elementos visuais são instâncias serializadas;

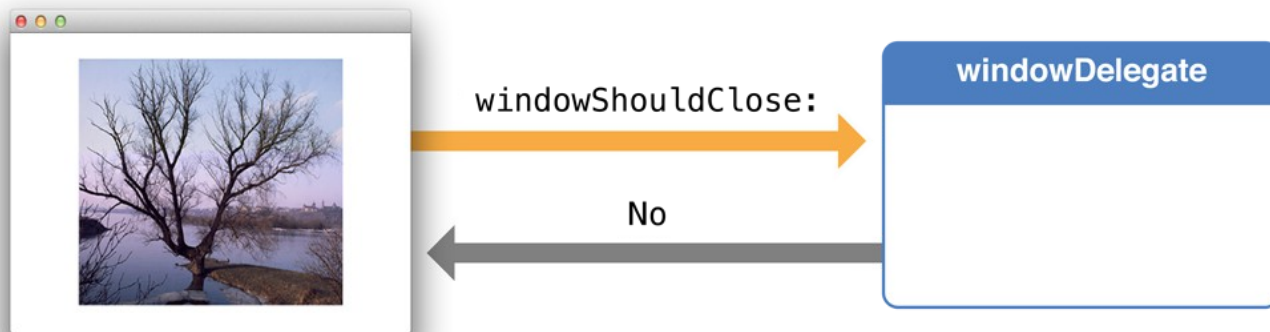
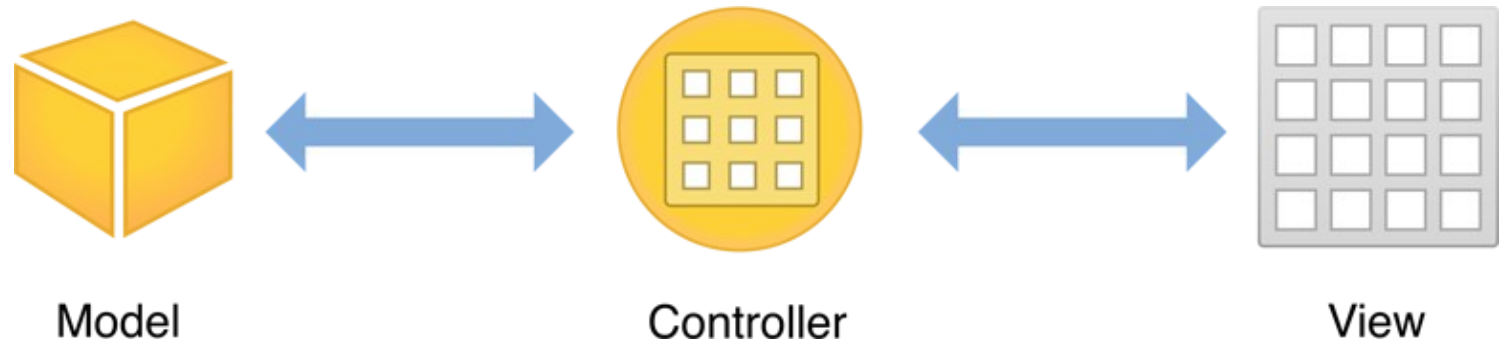
Storyboard - Exemplo

- *View controllers* pré-configurados;
- Componentes gráficos;
 - Configuração de componentes;

XIBs (NIBs)

- Única opção até lançamento do iOS5;
- Cada ficheiro corresponde a um ecrã ou elemento gráfico;
- Carregados automaticamente com base em convenções de nomenclatura;
- Recomendado para:
 - Divisão de ecrãs em aplicações com fluxo muito complexo;
 - Pequenos componentes reutilizáveis;

MVC e *Delegates* [1/2]



MVC e *Delegates* [2/2]

- Storyboard e XIBs agrupam controlador e vistas;
- Controladores registam-se como delegates das vistas;
- Vistas perguntam sempre aos delegates o que fazer a cada interacção do utilizador;

Actions e Outlets [1/3]

- Como referenciar componentes do Storyboard, de *design time*, definidos num ficheiro XML com o código que responde às acções?
- Como modificar, em *runtime*, as características dos componentes definidos no Storyboard?

Actions e Outlets [2/3]

- *Actions* ligam componentes ao método que responde ao evento;
- Definidas visualmente em *design time*;
- Direcção: Storyboard → Código;

Actions e Outlets [3/3]

- *Outlets* permitem ao código referenciar/aceder aos componentes definidos em *design time*;
- Definidas visualmente, em *design time*;
- Permitem modificar características da interface gráfica;
- Direcção: Código → Storyboard;

iOS – Objective-C

Módulo 1

