

FIAP - PLANO DE ENSINO

Curso	TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO DE SISTEMAS		
Disciplina	MOBILE DEVELOPMENT AND IoT - ANDROID		
Carga Horária Anual	80h		
Professor	DOUGLAS CABRAL / GUSTAVO CALIXTO	Turno	Manhã/ Noite

Ementa

Desenvolvimento de aplicativos para Android. Ambiente de Desenvolvimento Android Studio. Gerenciamento de Eventos. Componentes de Interface com o Usuário, Activity, Internacionalização, Gerenciadores de Layout, Estilos, Temas, Integração com o backend (JSON e RESTful), Intents, Broadcast Receivers, Serviços, API Google Maps, Persistência de Dados com SQLite. Desenvolvimento híbrido.

Objetivos	Apresentar ao aluno o conhecimento suficiente para a criação de aplicações móveis na plataforma Android.
Competências	Projetar aplicações móveis que sejam não só viáveis tecnicamente mas também ofereçam uma boa interface com o usuário.
Habilidades	<ul style="list-style-type: none"> ▪ Configurar corretamente os ambientes de desenvolvimento das plataformas; ▪ Entender os principais elementos arquiteturais das plataformas; ▪ Utilizar corretamente os elementos de interface com o usuário; ▪ Gerenciar os eventos produzidos pelo usuário e respondê-los corretamente; ▪ Utilizar APIs de Mapas; ▪ Criar camadas de persistência de dados; ▪ Implementar comunicação entre as aplicações nos dispositivos com sistemas <i>back end</i>; ▪ Implementar comunicação entre aplicações dentro do mesmo dispositivo; ▪ Projetar e implementar jogos simples; ▪ Fazer uso de recursos multimídia.

Conteúdo	
1º Semestre	2º Semestre
<p>Apresentação do Professor / Conteúdo</p> <p>Programático.</p> <ul style="list-style-type: none"> • Apresentação da matéria • Diferenças no desenvolvimento Nativo e Híbrido <p>Introdução a programação Java para Android</p> <ul style="list-style-type: none"> • Configuração do Projeto no Android Studio • Criação de emuladores • Configuração do SDK <p>Ambiente de desenvolvimento, Activity, Views</p> <ul style="list-style-type: none"> • Activities e o Ciclo de Vida • Entendendo Views • Back Stack <p>EditText, Button, ImageButton, ImageView e outros componentes</p> <ul style="list-style-type: none"> • Formulários • Imagens • Alerts e Toasts <p>Resources e internacionalização</p> <ul style="list-style-type: none"> • Internacionalização de Strings • Ícones e Mipmaps • Arquivos XML <p>Intents, Spinner, ArrayAdapter</p> <ul style="list-style-type: none"> • Passagem de Parâmetro • Chamando outros Apps • Intent Implícito e Explícito • Combobox de dados • O que são Adapters e quando utilizá-los <p>Layouts</p> <ul style="list-style-type: none"> • Tipos de Layouts • Vantagens / Desvantagens <p>Pickers</p> <ul style="list-style-type: none"> • Trabalhando com seleção de data e hora <p>Dialogs</p> <ul style="list-style-type: none"> • Tipos de Dialogs • Dialogs personalizados 	<p>Persistência</p> <ul style="list-style-type: none"> • Shared Preferences • File Output • File Input • SQLite • Outras alternativas de armazenamento <p>AsyncTask, Requisições HTTP</p> <ul style="list-style-type: none"> • Threads • UI Thread do Android • Tarefas consideradas pesadas para a UI Thread. • Consumo de APIs sem necessidade de bibliotecas <p>WebServices, Retrofit</p> <ul style="list-style-type: none"> • Consumo de APIs RESTful • Bibliotecas e clientes REST para Android • GET de dados com Retrofit • POST de dados com Retrofit • Outras alternativas ao Retrofit <p>Services</p> <ul style="list-style-type: none"> • Started Service • Integração Serviços / Activity • Consumo de APIs utilizando Services <p>Desenvolvimento de Aplicações com Kotlin</p> <ul style="list-style-type: none"> • História do Kotlin • Configuração do Ambiente e do Android Studio • Conversão de Arquivos Java para Kotlin • Comparação de Sintaxe • Extension Functions • Infix Functions • High Order Functions

<p>Toolbar e Navigations</p> <ul style="list-style-type: none"> ● Diferenças entre ActionBar e Toolbar ● Criação de Menus personalizados <p>Fragments</p> <ul style="list-style-type: none"> ● Entendendo o que são Fragments ● Integração de Mapas e Fragments <p>BroadCast Receiver</p> <ul style="list-style-type: none"> ● Interceptando mensagens do sistema ● Criar um leitor de SMS <p>Notifications</p> <ul style="list-style-type: none"> ● Criação de notificações 	<ul style="list-style-type: none"> ● Lambdas ● Anko ● Consumo RESTFul com Kotlin puro
---	--

Metodologia

Aulas Expositivas, Aulas Práticas com Estudos de Caso, Pesquisas, Discussões, Prática Profissional.

Avaliação

A média final para aprovação deve ser maior ou igual a 6,0 pontos. Essa nota será a soma das notas da prova semestral (PS), avaliação multidisciplinar (AM) e a nota de avaliação continuada (NAC) da seguinte forma:

$$PS*0,5 + AM*0,3 + NAC*0,2 \geq 6,0$$

Onde:

PS → Prova Semestral

AM → Avaliação Multidisciplinar

NAC → Nota de Avaliação Contínua

Bibliografia	
Básica	<ol style="list-style-type: none"> 1- LECHETA, Ricardo; Google Android – Aprenda a criar aplicações para dispositivos móveis com Android SDK. 3ª ed. São Paulo: Novatec, 2013. 2- PEREIRA, L.C.O.; Silva, M.L. Android para Desenvolvedores. 2ª ed. São Paulo: Editora Brasport, 2012. 3- DEITEL, P.; DEITEL, H.; DEITEL, A. Android: como programar. 2ª ed. São Paulo: Bookman, 2015.
Complementar	<ol style="list-style-type: none"> 1- LECHETA, Ricardo; Google Android para Tablets – Aprenda a desenvolver aplicações para Android - De smartphones e tablets. São Paulo: Novatec, 2012. 2- LEE, V.; SCHNEIDER, H.; SHELL, R. Aplicações Móveis: arquitetura, projetos e desenvolvimento. 1ª ed. São Paulo: Pearson Education do Brasil, 2005. * 3- BOND, Martin; Haywood, Dan; Law, Debbie; Roxburgh, Longshaw Peter. Aprenda J2EE: com EJB, JSP, Servlets, JNDI, JDBC e XML. 1ª ed. São Paulo: Pearson Education do Brasil, 2003. * 4- ERL, Thomas. SOA: Princípios de design de serviços. 1ª ed. São Paulo: Pearson Education do Brasil, 2009. * 5- DEITEL, P.J.; DEITEL, H. M. Ajax Rich Internet Applications e Desenvolvimento Web para Programadores. 1ª ed. São Paulo: Pearson Education do Brasil, 2008. *

(*) Acervo online