

科目名	モバイルアプリ開発2						年度	2026	
英語科目名	Mobile Application Development 2						学期	後期	
学科・学年	ITスペシャリスト科 AI・システム専攻 3年次	必/選	必	時間数	60	単位数	2	種別※	実習
担当教員	山本	教員の実務経験		有	実務経験の職種		システムエンジニア		
【科目の目的】 「モバイルアプリ開発2」では、iOSプラットフォームを対象に、Swift言語を使用したモバイルアプリの開発技術を実践的に学びます。学生は、より複雑な機能を持つアプリケーションの設計・実装を行い、実務的なスキルを磨きます。									
【科目の概要】 個人の生活に定着し、欠かせないと感じることが多いスマートデバイスを、業務でも積極的に活用しようとする企業が増えてきている。本授業ではスマートデバイスを企業で導入するに当たり、知っておきたい基本的な用語やモバイルシステムの構成要素、モバイルアプリケーション開発の概要について学習する。この科目を受講する学生は、モバイルアプリケーション開発の基礎を理解できる。前期までにAndroidアプリケーション開発の基礎を習得しており、それらを応用してiOSアプリケーション開発に取り組む。開発環境のXcodeなど初めて触れる技術や知識が多くなるため、書籍やマニュアルを調べたり、自己解決力が求められる。									
【到達目標】 A. iOSアプリ開発の基本を理解し、基礎的なアプリを作成できる。 B. SwiftとSwiftUIを活用し、ユーザーインターフェースを設計・実装できる。 C. アプリの基本機能を実装し、デバッグ・テストが行える。 D. アプリの収益化や公開手順を理解し、実行できる。 E. 自分のアイデアをアプリとして具現化し、最終発表で効果的に伝えられる。									
【授業の注意点】 授業中の私語や受講態度には厳しく対応する 理由のない遅刻・欠席は認めない 授業時数の4分の3以上出席しない者は定期試験を受験することができない									
評価基準＝ルーブリック									
ルーブリック 評価	レベル5 優れている	レベル4 よい	レベル3 ふつう	レベル2 あと少し	レベル1 要努力				
到達目標 A	基本を完全に理解し、基礎的なアプリを独立して作成できる。	基本的な知識があり、少しのサポートで基礎的なアプリが作成できる。	基本を理解しているが、複数のサポートを必要とする。	基本を部分的に理解しているが、ほぼ全ての作業にサポートが必要である。	開発の基本を理解していない。				
到達目標 B	UI設計・実装を独自のアイデアで行い、ユーザビリティを考慮できる。	基本的なUI設計・実装ができ、多少の修正が可能である。	簡単なUI設計ができ、標準的なコンポーネントが使用できる。	UI設計の基礎を理解するが、サポートが必要である。	UI設計・実装ができない。				
到達目標 C	基本機能を適切に実装し、独立してデバッグ・テストができる。	基本機能を実装し、サポートを受けてデバッグ・テストを行える。	基本機能を理解しているが、複数のサポートを必要とする。	基本機能の理解に不安があり、デバッグやテストができない。	基本機能を実装できない。				
到達目標 D	収益化と公開手順を完全に理解し、実行できる。	基本的な収益化と公開手順を理解し、指導のもとで実行できる。	収益化の手順の一部を理解しているが、複数の指導が必要である。	収益化と公開手順について断片的にしか理解できていない。	収益化や公開手順の理解ができていない。				
到達目標 E	アイデアを独自のアプリに具現化し、明確で説得力のある発表ができる。	アイデアをアプリに具現化し、サポートを受けながら発表できる。	アイデアを一部具現化でき、基本的な発表ができる。	アイデアの具現化が難しく、発表に多くのサポートが必要である。	アイデアを具現化できず、発表も難しい。				
【教科書】 絶対挫折しない iPhoneアプリ開発「超」入門									
【参考資料】									
【成績の評価方法・評価基準】 授業時に行う小テストや確認問題、提出物などを総合的に評価します。									
※種別は講義、実習、演習のいずれかを記入。									

科目名		モバイルアプリ開発 2			年度	2026
英語表記		Mobile Application Development 2			学期	後期
回数	授業テーマ	各授業の目的	授業内容	到達目標＝修得するスキル	評価方法	自己評価
1	iPhoneアプリ開発の基礎知識	iOS開発の基本、Xcode概要、iOSアプリの構造理解	1 iPhoneアプリ開発の概要	iOS開発の基本概念や流れを理解する	2	
			2 アプリ開発に必要なツール	XcodeやSwiftについての基本知識を得る		
			3 iOSエコシステムの理解	App StoreやAppleのエコシステムについて理解する		
2	Xcodeとシミュレータの基本操作	Xcodeの使用方法、シミュレータ操作方法	1 Xcodeの基本操作習得	Xcodeの主要なインターフェースを操作できる	2	
			2 シミュレータの使用方法	シミュレータを使ってアプリをテストするスキルを身につける		
			3 デバッグツールの使用	Xcodeで基本的なデバッグができるようになる		
3	SwiftUIの導入(1)	SwiftUIの概要と基本構文	1 SwiftUIの基本構造理解	SwiftUIのレイアウト構造を理解する	2	
			2 変数と定数の使い方を習得	Swiftでの変数や定数を使えるようになる		
			3 ビューの作成方法学習	簡単なビューをSwiftUIで構築できる		
4	SwiftUIの導入(2)	ビュー構築とレイアウト設定	1 基本的なレイアウト作成	簡単なレイアウトをSwiftUIで実装する	2	
			2 テキストと画像の表示方法	テキストと画像をアプリに追加できる		
			3 インタラクティブな要素	ボタンやスライダーなどのインタラクティブ要素を追加できる		
5	iPhoneアプリ開発の初歩	簡単なアプリの設計、インターフェースの組み立て	1 プロジェクトの作成方法	新規アプリプロジェクトを作成できる	2	
			2 アプリのライフサイクル	アプリの起動から終了までの流れを理解する		
			3 ナビゲーションの設定方法	基本的な画面遷移を実装できる		
6	アイコン画像の使い方	画像のインポート、プロパティとイニシャライザの使用	1 アイコン画像の設定方法	アプリのアイコンを設定できる	2	
			2 コンピューテッドプロパティ	Swiftでのプロパティを活用できる		
			3 イニシャライザの使用	イニシャライザの役割を理解し、活用できる		
7	カウンターアプリ開発	カウンターの作成、クロージャの使用方法	1 クロージャの基礎学習	基本的なクロージャの書き方を理解する	2	
			2 カウンター機能の実装	カウンターアプリの動作を実装できる		
			3 UIとのクロージャ連携	クロージャでUIを操作できる		
8	地図アプリの開発とプロトコル	地図表示機能、プロトコルの活用	1 プロトコルの基礎理解	プロトコルの役割と使用方法を理解する	2	
			2 地図表示の実装	地図表示をアプリ内で行える		
			3 マップとのインタラクション	ユーザーと地図が連動する機能を追加できる		
9	英単語カードアプリとオプションの活用	カードアプリの開発、オプションの利用方法	1 オptional型の理解	Optional型の扱い方を理解する	2	
			2 英単語カードの実装	英単語カードアプリの基礎機能を実装できる		
			3 データ保存と取得方法の活用	データの保存と取得を行える		
10	少し高度なアプリ開発	複数の機能を組み合わせたアプリ開発演習	1 複雑なレイアウト実装	高度なUIレイアウトを作成できる	2	
			2 データ管理方法学習	データの管理と操作を行える		
			3 ユーザー体験の向上施策	UXを意識した設計ができる		
11	アプリ開発の仕上げ	エラー処理、コードの最適化、UIの調整	1 完成度の向上施策	細かいバグ修正ができる	2	
			2 テスト方法の学習	効果的なテスト手法を学び、実践する		
			3 改善ポイントの把握	改善点を見つけ、対策を考えられる		
12	アプリの収益化方法と公開手順	アプリの収益化、App Storeへの公開手順、アプリ更新方法	1 収益化モデルの理解	広告や課金の仕組みを理解する	2	
			2 App Storeへの公開準備	アプリ公開に必要な準備ができる		
			3 リリース後のメンテナンス	公開後の管理方法を学ぶ		
13	実践プロジェクト(1)	個人またはグループでのアプリ制作	1 プロジェクト計画立案	自分のアイデアをプロジェクトとして計画する	2	
			2 アプリ開発開始	基本機能の実装を開始できる		
			3 実装とテスト	テストを通じて問題点を見つけ、改善できる		
14	実践プロジェクト(2)	個人またはグループでのアプリ制作の続き、プレゼン準備	1 プロジェクトの完成	機能を完成させ、デバッグできる	2	
			2 プレゼンテーション準備	アプリを効果的に紹介する準備ができる		
			3 最終調整	アプリの品質を高めるための調整を行う		
15	最終発表とフィードバック	発表と相互評価、フィードバックの共有	1 アプリのプレゼンテーション	開発したアプリを発表し、説明できる	2	
			2 フィードバック受け取り	他者のフィードバックを理解し、今後活かす		
			3 学習成果の振り返り	開発プロセスを振り返り、次への学びを考える		

評価方法：1. 小テスト、2. パフォーマンス評価、3. その他

自己評価：S：とてもよくできた、A：よくできた、B：できた、C：少しできなかった、D：まったくできなかった

備考 等