

科目名	クラウドコンピューティング 1								年度	2026
英語科目名	Cloud Computing 1								学期	後期
学科・学年	I Tスペシャリスト科	2年次	必/選	必	時間数	60	単位数	2	種別※	実習
担当教員	金井		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】 パブリッククラウド(AWS Azure)に触れ、最新動向やクラウドを取り巻く関連技術について学ぶことでクラウドエンジニアへの業界の需要に応えるスキルを身に着ける第一歩とする。										
【科目の概要】 実際の業務でも使用されるクラウドサービスなど、最先端のI Tサービスを体験します。										
【到達目標】 クラウドサービスの概要、特徴、事例を学び、身に着ける。 各クラウドサービス提供事業社の違いを理解する。 SaaS IaaS PaaSの違いを理解し、サービスを利用する上での簡易的な設定や操作ができるようにする。										
【授業の注意点】 資料は紙およびデジタルデータで配布する。デジタルデータの場合は授業中に指定するサーバからのダウンロードとなる。授業に必要なスマートフォン、ペットボトルは机上に置かないこと。私語を慎み、積極的に実習に参加すること。理由のない遅刻や欠席は認めない。また、授業時数の4分の3以上出席しない者は定期試験を受講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	AWSのオリジナルドキュメントを参照し、自分でAPIの実装、ポリシーの設定をした上でサービスを利用できる	有識者のアドバイスを元にポリシーを設定し、自分でAWSサーバレスサービスを実装することができる	有識者のサポートがあれば自力でAWSサーバレスサービスを実装することができる	有識者サポートの下AWSの各種サービスを活用することができる	有識者によるサポートがあってもAWSサービスを利用することができない					
到達目標 B	需要があり、他との差別化が図れるアプリケーションを企画し、完成させることができる	他に似たようなサービスはあるが、有用なアプリケーションを企画し、完成させることができる	アプリケーションの企画はできるが、自力での実装が難しくサポートがあれば完成させることができる	企画、実装共にサポートがあればアプリケーションを完成させることができる	サポートがあっても企画、実装共にアプリケーションを完成させることが難しい					
到達目標 C	AWSとAzureの違いを理解し、サポートが無くても自分でAzureの各種サービスを使いこなすことができる	有識者のアドバイスを元に自力でAzureの各種サービスを使いこなすことができる	有識者のサポートがあれば自力でAzureの各種サービスを使いこなすことできる	有識者サポートの下Azureの各種サービスを活用することができる	有識者によるサポートがあってもAzureサービスを利用することができない					
到達目標 D										
到達目標 E										
【教科書】 この一冊で全部わかるクラウドの基本第2版 (SBCreative)										
【参考資料】										
【成績の評価方法・評価基準】 定期試験と課題、理解度確認の小テストを行い、総合的に評価する。授業参加度、授業態度も評価に含まれる。										
※種別は講義、実習、演習のいずれかを記入。										

科目名		クラウドコンピューティング 1			年度	2026
英語表記		Cloud Computing 1			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	AWSを使ってWebサーバを立てる	AWSAcademyを利用してクラウドコンピューティングを体験する	1 クラウド技術とは	ITリソースをサービスとして提供し、オンデマンド、セルフサービスなど5つの基本特性について理解する	1, 2	
			2 AWSのグローバルインフラ構成に	リージョン、AZの違いを理解し選択時の考え方を習得する		
			3 デフォルトVPC内にEC2サーバを起動し、WEBサーバにする	インスタンス利用方法、Webサーバ化の手法を身に付ける		
2	AWSを使ってVPCを作成する	ネットワーク、サブネットワークの考え方を復習し、VPCで実践する	1 IPアドレス復習	CIDR表記、サブネットワーキングの復習	1, 2	
			2 サーバへの理解	仮想サーバ (EC2) の機能について理解する		
			3 VPC作成	VPCの中にサブネットワークを3つ作り、うち1つをパブリック化してパブリックサブネット内に立てたEC2へのアクセスを試みる		
3	AWSを使ってロードバランサで負荷分散する	ELBの利点を理解し、第2回授業で作成したVPC内で負荷分散を実践する	1 VPCに付随する機能への理解	SG ACLの設定方法、複数レイヤーでのセキュリティ対策について理解する	2	
			2 スナップショット、自作AMIを使ってミラーリ	第2回で作成しているEC2サーバを別AZのパブリックサブネットへ複製する		
			3 ELBの作成	異なるAZのパブリックサブネットに立っているWEBサーバに負荷分散するELBを作成し、同じELBエンドポイントにアクセスしてそれぞれのサーバへ切り替わることを確認する		
4	AWSのストレージサービスを使用する	ブロックストレージ、オブジェクトストレージの違いを理解し、S3を使用した静的Webホスティングを実践する	1 S3とEBS	S3とEBSの違いを理解し、ユースケースを学んで設計時の選択に活かす	2	
			2 S3とEC2	EC2を使ったWebホスティングとS3を使ったWebホスティングの違いと最適な設計について理解する		
			3 S3で静的ウェブサイトホスティング	バケットポリシーの設定と静的ウェブサイトホスティングができるようになる		
5	AWSのEC2でLAMPサーバを作ってWordPressを導入する	需要の高いCMSであるWordPressが利用できるLAMP環境作成をグループワークで実践する	1 グループ作成	グループング、アイスブレイクグループ作業の意義について理解する	2	
			2 LAMP環境作成	VPCを作成し、パブリックサブネット内にEC2をたてて、Linux Apache MariaDB PHPの環境を構築する		
			3 WordPressの設定	MariaDBと連携し、自作ブログサイトを公開する		
6	AWSLambdaを使ってサーバレスアーキテクチャを実践する	サーバレスアーキテクチャを理解し、Lambdaを使ってWebサイトの表示や、Email通知を実践する	1 サーバレスアーキテクチャ	サーバレスアーキテクチャとは何かを理解する	2	
			2 Lambdaを使ってWebサイトを作成	Lambda関数でWebサイトを表示する		
			3 Lambdaを使って、Email通知を実装	Lambda関数でS3へのファイルアップロードをトリガーにEmail通知を行う機能を実装する		
7	AWS DynamoDBを使ってNoSQLのデータベースを操作する	NoSQLデータベースへの理解を深める	1 AWSのデータベースサービス	RDS Aurora DynamoDB Redshiftなどの特徴を理解し、それぞれにマッチしたユースケースについて理解する	2	
			2 APIを使ってLambdaからDynamoDBにログデータを書き込む	手動でデータを格納してみ、NoSQLのイメージを付けた後、LambdaからDynamoDBAPIを呼び出し、ログデータを書き込む処理を実装する		
			3 WebAPIを自作する	【任意課題】授業内容2まで実施が完了して授業時間に余裕がある場合のみ、APIの自作課題を提示		
8	AWS RDSを使ってデータベースを操作する	EC2にセットアップしたMariaDBとRDSでセットアップしたMariaDBの違いを意識し、理解を深める	1 マネージドサービスとアンマネージドサービスの違い	EC2とRDSのメリット、デメリットについて理解する	2	
			2 次回授業準備	Azure for Studentsを設定する		
9	Microsoft Azureを使ってみよう	AWSとの違いを意識しながらAzureへの理解を深める	1 Azureの強み	AWSと比較し、Azureの強みを理解する	2	
			2 Vnetを作成	Azure Virtual Networkを作成し、仮想ネットワークの作成方法を理解する		
			3 NSGを作成	NSGを作成し、サブネットと関連付けて最低限のセキュリティを確保する		
10	Azure VMを使ってインスタンスを起動する	AWS EC2との違いを意識しながらVMが起動できるようにする	1 VMをパブリックサブネットに配置	VMをパブリックサブネットに配置、起動し、パブリックIPアドレスが割り振られるところまで確認する	2	
			2 Webサーバ化	VMへSSHで接続し、Webサーバ化する		
			3 AGWを使って負荷分散	ELBとAGWの違いを意識しながら、Webサーバを異なるZoneに配置したVMへ負荷分散するAGWを作成する。		
11	Microsoft PowerAppsを使ってローコードでアプリケーションを作成する	アプリ作成のクラウドプラットフォームを使用してアプリを作成する	1 PowerAppsの環境構築	学生配布済みのMicrosoftアカウントを使って環境をセットアップする	2	
			2 スプレッドシートをデータソースとして	ToDoアプリ作成し、ローコードプラットフォームアプリ開発を体験する		
			3 クイズアプリを作成	チュートリアルに従ってクイズアプリを自作する		
12	Microsoft PowerAppsを使ったアプリの企画を考える	自分が作ってみたいアプリケーションを作成するための企画書を作成する	1 企画書の作成	企画書を作成する	2	
13	Microsoft PowerAppsを使ったアプリを自作する	アプリの企画に沿って実装する	1 アプリの実装	実装を進める	2	
14	Microsoft PowerAppsを使ったアプリを自作する	自作したアプリを完成させ、プレゼンする動画を作成する	1 アプリの実装	実装を進め、完成させる	2	
			2 プレゼン動画作成	完成したアプリケーションの作成動機、有用性、挙動、操作性についてプレゼンテーションしている動画を作成する		
15	作品評価	他学生の作品を閲覧し、学生同士で評価する	1 アプリの評価	他学生の作品についてのプレゼンテーションを見て、評価する	2	

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