IT用語２（カリキュラム３章）

■ 開発

▼ウォーターフォール開発

開発手順を1つずつ確認しながら工程を進めていく手法。開発を各工程に分けて進め、次のフェーズに進んでしまうと後戻りができない手法でもあります。

例）各機能それぞれ同時並行で企画・設計・実装・テストと進めていく

▼アジャイル開発

クライアントの要望に応えるシステムをできる限り素早くリリースしようという考えに基づいている開発手法です。搭載する機能を最低限の状態にする方法が取られるケースが多いです。開発メンバーがシステムにおける優先度に順位をつけ、短い期間での納品を目指して動きます。チーム内でスムーズに連携をとる「スクラム」という手法が主に取られます。

例）各機能それぞれ個別に企画・設計・実装・テストと進めていく

▼マイグレーション

ソフトウェアやシステム、データなどを別の環境に移転したり、新しい環境に切り替えたりすること

▼デグレーション

新しいバージョンのソフトウェアの品質が、以前より悪くなること。

また、以前修正した不具合やバグが再発・復活すること。新しいファイルなどを古い内容で上書きしてしまい、更新内容が失われることを指す場合もある。

▼デプロイ

開発したソフトウェアを実際の運用環境に配置・展開して実用に供することを指す場合が多い。

▼コンフリクト

複数の装置やプログラムなどが同じ資源を同時に利用しようとして競合状態になってしまうことを意味する場合が多い。「競合」「衝突」の訳語が当てられることが多い。

▼環境

ある主体にとって、その周囲を取り囲み影響を与える世界やその状態、状況のこと。

▼開発環境（ローカル）

機器やソフトウェア、システムを開発するために必要な機材やソフトウェアなどの組み合わせ。また、それらを組み合わせて構成された、開発者の作業環境。

▼検証環境（ステージング）

開発された新システムの導入や公開の直前に、本番と同じ状態で実際に稼働させてみて問題が起きないか確認・検証するために構築される。検証が済んだシステムは本番環境（プロダクション環境）に導入され利用に供される。

▼本番環境（プロダクション環境）

情報システムやソフトウェアの開発で、できあがった製品を実際に稼働させて利用に供するための施設や機材、

ソフトウェアなどを組み合わせた運用環境のこと。

▼設計書／仕様書

工業製品やソフトウェア、情報システム、サービスなどを開発する際に、備えるべき機能やその性能、特性、満たすべき要件などを図表や文章などで記述した文書。

▼要件定義

システムやソフトウェアの開発において、実装すべき機能や満たすべき性能などを明確にしていく作業のこと。いわゆる上流工程の一部で、実際の開発・実装作業を始める前に行う作業の一つである。

▼基本設計書

製品の基本的な設計。また、製品開発において基本的な設計を策定する工程。構成や仕様、機能などの概要をまとめたものを意味する場合と、中枢や基盤の設計を意味する場合がある。

▼詳細設計書

詳細設計とは、ソフトウェアや情報システムの開発工程の一つで、前工程で定義された要素の仕様や動作の詳細を定義する工程。全体の工程の分割の仕方により具体的な作業内容が異なる。

▼単体テスト（UT）

単体テストとは、ソフトウェアやシステムのテスト手法の一つで、単一の部品（モジュール）を対象に行うテスト。

▼結合テスト（CT／IT）

結合テストとは、開発中のソフトウェアのテスト手法の一つで、複数のモジュール（部品）を組み合わせて行うテスト。個々のモジュールの単体テスト後に行う。

▼総合テスト（ST）

システムテストとは、開発中のソフトウェアや情報システムのテスト手法の一つで、開発の最終段階にシステム全体を対象に行われるテスト。

▼エビデンス／証跡

システム開発のテスト工程などで、システムが正しく動作している、あるいは不具合が生じたことを示す証拠となる書類やデータなどをこのように呼ぶことが多い。

▼データパッチ

ソフトウェアを構成するプログラムやデータを更新・修正するために、新しい版との相違点（差分）を抜き出したデータのことをパッチという。ファイルの形にまとめることが多いため「パッチファイル」とも呼ばれる。パッチを反映させて最新の状態に更新することを「パッチを当てる」「パッチを適用する」などという。

▼Access

機器やソフトウェア、システム、情報などを利用（操作、取得、閲覧、読み書きなど）可能な状態にすることをアクセスという。特に、コンピュータのストレージ装置（外部記憶装置）内に保存されたデータを読み出したり、上書きや削除などを行うことをアクセスということが多い。

▼マクロ

アプリケーションソフトの操作を自動化する機能の一つにマクロがある。利用者が複数の操作手順を一連の手続きとして記録したもので、簡単な操作で繰り返し実行することができる。

▼VBA

Microsoft Officeに含まれるアプリケーションソフトの拡張機能で、利用者が簡易なプログラムを記述して実行することで複雑な処理の自動化などを行なうことができるもの。また、そのために用意されたプログラミング言語。

※▼GAS（Google Apps Script）

Google Apps Script（GAS）とは、Googleが開発・提供しているプログラミング言語です。JavaScriptというWebブラウザ上で動作するプログラミング言語をベースに作成されています。

▼API

あるコンピュータプログラム（ソフトウェア）の機能や管理するデータなどを、外部の他のプログラムから呼び出して利用するための手順やデータ形式などを定めた規約のこと。

▼WEB API

コンピュータプログラムの提供する機能を外部の別のプログラムから呼び出して利用するための手順・規約（API：Application Programming Interface）の類型の一つで、HTTPなどWebの技術を用いて構築されたもの。

▼REST API

Webシステムを外部から利用するためのプログラムの呼び出し規約（API）の種類の一つで、「REST」（レスト）と呼ばれる設計原則に従って策定されたもの。RESTそのものは適用範囲の広い抽象的なモデルだが、一般的にはRESTの考え方をWeb APIに適用したものをRESTful APIと呼んでいる。

▼SOAP

異なるコンピュータ上で動作するプログラム同士がネットワークを通じてメッセージを伝え合い、連携して動作するための通信プロトコル（規約）の一つ。メッセージの記述にXMLを、データ伝送に主にHTTPを用い、Webサービスの提供や利用に適しているとされる。

▼Ajax

Webページを表示した状態のまま、別のページや再読込などを伴わずにWebサーバ側と通信を行い、動的に表示内容を変更する手法。ページ上でプログラムを実行できるプログラミング言語JavaScriptの拡張機能を用いる。

▼XML

XMLとは、文書やデータの意味や構造を記述するためのマークアップ言語の一つ。マークアップ言語とは、「タグ」と呼ばれる特定の文字列で地の文に情報の意味や構造、装飾などを埋め込んでいく言語のことで、XMLはユーザが独自のタグを指定できることから、マークアップ言語を作成するためのメタ言語とも言われる。

▼JSON

JSONとは、JavaScriptにおけるオブジェクトの表記法を応用したテキスト（文字）ベースのデータ形式。多数の要素が複雑な構造で組み合わせられたデータを簡潔な表記で書き表すことができる。JavaScriptプログラム上ではコードとして実行するだけで読み込みが完了する。

▼ステータスコード

システムが処理結果や現在の状態を外部に知らせるために発する数字や短い符号。問題発生時にその種類や状態を知らせるものは「エラーコード」とも呼ばれる。

▼クロスオリジン（Cross-Origin）／CORS（Cross-Origin Resource Sharing）

CORSとは、Webサーバに特殊なHTTPヘッダ項目を追加することで、送り出したWebページ上のスクリプトがWebブラウザから別のサーバへアクセスできるようにすること。

▼バリデーション

検証、実証、認可、妥当性確認などの意味を持つ英単語。ITの分野では、対象がその仕様や文法などに照らして適切に記述・構築されているか否かを検証するという意味で用いられることが多い。

※▼データ連携

データ連携とは、企業内外のさまざまなシステムやアプリケーションのデータをつないで、有効活用するための仕組みやプロセスのことです。

▼Daemon

UNIX系OSでメインメモリ上に常駐して特定の機能を提供するプログラム。利用者の操作とは無関係に処理を行うバックグラウンドプロセスの一種で、別のプログラムや機器、通信相手など外部からの要求を受け付けて処理を実行したり、あらかじめ設定されたタイミングで自律的に特定の処理を実行したりする。

▼CRON

多くのUNIX系OSで標準的に利用される常駐プログラム（デーモン）の一種で、利用者の設定したスケジュールに従って指定されたプログラムを定期的に起動してくれるもの。

▼バッチ

複数のプログラムからなる作業において、あらかじめ一連の手順を登録しておき、まとめて連続的に実行する方式。または、一定期間や一定量ごとにデータをまとめて一括して処理する方式。

▼ログ

ある機器やソフトウェア、システムについて、その起動や停止、エラーや障害の発生、利用者による操作や設定の変更、外部との通信など、稼働中に起こった出来事の内容を日時などとともに時系列に記録したものをログという。

▼CSV

テキスト（文字）データの形式の一つで、項目をカンマ「,」で区切って列挙したもの。複数の項目をレコードとしてまとめる場合は、改行でレコードの区切りを表す。標準のファイル拡張子は「.csv」。

▼TSV

複数の項目で構成されるデータを複数件列挙して表現することができる汎用的なデータ形式の一つで、項目間の区切り文字にタブ文字を用いるもの。

▼プルリクエスト（マージリクエスト）

分散バージョン管理システム（VCS）の機能の一つで、コードなどを追加・修正した際、本体への反映を他の開発者に依頼する機能。「変更を本人以外がレビューしてから反映させる」という手順を容易に実現することができる。

■ チケット管理ツール

▼asana

Asanaは、ワークマネジメントツールである。 共にFacebook社員の生産性向上に関わった、共同創業者のDustin Moskovitzと元エンジニアのJustin Rosensteinが2008年に創設した。

▼Backlog

バックログとは、残務、積み残し、受注残、未処理分、在庫分、予備にとっておく、未処理のまま溜まる、などの意味を持つ英単語。実施すべきとされながら、未処理、未着手のまま放置されている作業や業務、案件などのこと。今後着手すべき作業などを期限や優先度などとともに列挙にしたものを指すこともある。

▼Jooto

Jootoは、株式会社PR TIMESが提供している、クラウド型のプロジェクト・タスク管理ツール。日本語と英語両方に対応しており、ドラッグ・アンド・ドロップで操作できるシンプルなプロジェクト管理を実現。

▼Readmine

Redmineとは、オープンソースのプロジェクト管理ソフトの一つ。複数人の共同作業における進捗管理や情報共有などができるシステムで、基本的な操作や管理はWebブラウザを通じて行なうことができる。

▼Trello

「Trello」とは、カードを動かしながらタスクを視覚的に管理できるカンバン方式のツールです。壁面に付箋紙を貼るような感覚で、ドラック＆ドロップで直感的に操作が可能。わかり易いインターフェースと、多数の外部ツール（Slack、Github、Salesforceなど）と連携したプラットフォームの構築により、サービスが充実していること。また、無料での利用範囲の広さも魅力的

■ バージョン管理

バージョン管理とは、データを作成・更新などする際に変更履歴を保存し、後からそのデータの任意の時点の版を参照できるようにする仕組み。コンピュータプログラムの開発・修正を円滑に進めるためによく利用される手法で、文書管理など他の分野でも利用されている。

▼集中型

複数の作業者が同じファイル群を扱うものとします．

ある1つのサーバで中央集中型システムは稼働し，だれでもそのサーバからファイルのコピーを取得して作業ができます．変更した内容はサーバにある内容に反映(コミット)できます．

また，他の利用者が更新操作を行ったことにより，サーバ内のファイルが変更されると，その内容も自分の環境に同期できます．ここで，サーバ内にあるファイル群のことをリポジトリと呼びます．

▼分散型

分散型システムは中央集中型と違って，主となるサーバは持っていません．その代わりに，だれもが独立したリポジトリを用意して，他のリポジトリと同期することができます．中央集中型ではリポジトリを用意できるのは1つのサーバだけでした．それに対し，分散型は複数のリポジトリを持つことができます．

なので，中央集中型ができないことを分散型では可能になります．リポジトリを複数存在することから「分散型」と呼ばれます．

▼Github

GitHubとは、バージョン管理システム「Git」（ギット）で管理するデータをインターネット上で共有・公開することができるネットサービスの一つ。米マイクロソフト（Microsoft）社傘下の同名企業（GitHub Inc.）が運営している。

▼Backlog

Backlogとは、株式会社ヌーラボが提供するプロジェクト管理ツール。株式会社ヌーラボがASPとしてサービス提供しているほか、パッケージ版の販売も行われている。

▼SourceTree

SourceTreeとは、Atlassian社が提供するGitの分散管理システムツール操作を効率的に扱うGUIで、GUIとはユーザー画面上で視覚的に操作ができる機能です。 ... 誰がいつ、どのようなプログラムを書いたのか、修正や追加をした場合も視覚的に確認できます。

▼Tortoise SVN

TortoiseSVN は、バージョン管理システムであるSubversionのクライアントフロントエンド。GNU General Public Licenseのもとに配布されているフリーソフトウェアである。

▼Git Flow

git-flowとは、プラグイン(ツール)のことです。。

Vincent Driessen氏がブログに書いた"A successful Git branching model" というブランチモデルの導入を簡単にする git プラグインである。

▼master

master ブランチは、Gitでリポジトリを新規作成するとデフォルトで作成されるブランチですね。初心者の中には、masterブランチに直接コミットしている方も多いでしょう。それはそれで正しい使い方なのですが、git-flowでは master ブランチに直接コミットすることはなく、マージを行うだけのブランチになります。誤って直接コミットしてしまわないように注意しましょう。

▼release

release ブランチは、その名の通り製品をリリースするために使うブランチです。製品のリリース時には、関連する作業が必要になる場合が多いでしょう。そういった作業は、develop ブランチから release ブランチを切って、そのブランチでリリース作業を行います。リリース作業が完了したら、master ブランチと develop ブランチにマージして、master ブランチのマージコミットにリリースタグ（バージョンなど）をうちましょう。その後、release ブランチは削除します。

▼develop

develop ブランチは、開発の中心となるブランチです。開発中は develop ブランチからブランチを切って、作業完了後に再びマージするという作業を繰り返すことになります。master ブランチ同様、直接このブランチにコミットすることはないので注意しましょう。リポジトリを新規作成したときに、master ブランチから develop ブランチを切っておきます。

▼feature

feature ブランチは、機能の追加や変更、バグフィックスを行うブランチです。ひとつの変更に対してひとつの feature ブランチを切ることになるため、開発中で最も使われるブランチになるでしょう。ブランチの名前は、変更の内容がすぐに分かるような名称にしなければなりません。develop ブランチから派生させ、作業完了後に再び develop ブランチにマージします。そして、マージ完了後に削除するというのが一連の流れです。

■ FW／ライブラリ

▼FW

ソフトウェア開発の分野では、ある領域のソフトウェアに必要とされる汎用的な機能や基本的な制御構造をまとめた半完成品のことを「ソフトウェアフレームワーク」「アプリケーションフレームワーク」などと呼び、これを略してフレームワークと呼ぶことが多い。

▼Spring Framework

Java言語によるソフトウェア開発を支援するフレームワークの一つ。様々な開発プロジェクトに適用できる汎用性の高さとモジュール化された豊富な機能に定評がある。

▼Struts

Javaを用いてWebアプリケーションを開発するためのフレームワークの一つ。Apacheソフトウェア財団（ASF）が開発を主催し、オープンソースソフトウェアとして公開している。

▼JSF（Java Server Faces）

JSFとは、Java言語を用いてWebアプリケーションを開発するためのフレームワークの一つ。Javaの開発元であるOracle社が開発・提供しているもので、その仕様はJava EE（Java Platform, Enterprise Edition）の一部として取り込まれている。

▼ライブラリ

ITの分野では、ある特定の機能を持ったコンピュータプログラムを他のプログラムから呼び出して利用できるように部品化し、そのようなプログラム部品を複数集めて一つのファイルに収納したものをライブラリという。

※▼Bootstrap

ブートとは、コンピュータを起動すること。また、人間がコンピュータに電源を投入（あるいはリセット）してから、OSが操作可能な状態になるまでに自動的に行われる一連の処理。

※▼Flyway

▼Jackson

Jackson(ジャクソン)は、JSON（※）形式のデータを処理するための高速かつ軽量なJavaライブラリです。「Jackson Java JSON-processor」とも呼ばれています。

基本データ形式。作業しやすいようにするためのもの

▼jQuery

Webブラウザ上で動作するJavaScriptライブラリの一つ。ブラウザの種類に依らず簡潔な記述で豊富な機能を活用することができる。オープンソースとして公開・配布されている。

▼Junit

JUnitとは、Java言語で開発されたプログラムの単体テスト（ユニットテスト）を行なうためのソフトウェア。また、そこで用いられるテストコードの記述体系を含むテストフレームワーク。

▼Log4j

Apache log4jは、Javaのロギングユーティリティ。元々はCeki Gülcüにより開発されていたが、現在はApacheソフトウェア財団のプロジェクトの一つとなっている。主にデバッグ用ツールとして使用されている。log4jは、Java logging frameworkの一つである。

▼Lombok

Lombok は、 Java言語におけるボイラープレートコードをソースコードから排除するために使用するライブラリである。 ボイラープレートコードとは、言語仕様上省く事ができない定型的なコードの事である。 ボイラープレートコードは本質的なロジックでないため、アプリケーションを実装する上で冗長なコードとなる。

▼ORM

O/Rマッピングとは、オブジェクト指向プログラミング言語におけるオブジェクトとリレーショナルデータベース（RDB）の間でデータ形式の相互変換を行うこと。そのための機能やソフトウェアを「O/Rマッパー」（O/R mapper）という。

■ 連絡ツール

▼ChatWork

Chatworkとは、Chatwork株式会社が提供するクラウド型ビジネスチャットツールである。メッセージのやりとりだけでなく、タスク管理やファイル共有、ビデオ通話などが可能である。

▼Slack

Slackは、スチュワート・バターフィールドによって開発されたチームコミュニケーションツール。 Slackという名称は"Searchable Log of All Conversation and Knowledge"のアクロニムである。

▼IP Messenger

IP Messenger は白水啓章さんが作られた LAN 用メッセンジャーソフトです。

同じネットワーク内のメンバーを自動認識し、簡単にクライアント同士がリアルタイムに情報交換できる優れたソフトウェアです。

■ Java

▼Javadoc

Javadocとは、Java言語のソースコードに一定の書式でコメントを記述することで、クラスの説明やメソッドの呼び出し方などの仕様をHTMLなどの文書形式でまとめてくれる仕組み。また、そのためのコメントの書式。

▼アノテーション

プログラミングでは、ソースコード中に登場する要素（クラスやメソッドなど）に対して、処理系に伝達したい付加的な情報（メタデータ）を注記する仕組みのことをアノテーションという。

▼メモリ管理

メモリー管理とは、使用していないオブジェクトをヒープから削除して、新しいオブジェクトで使用できるようにするプロセスです。 この章では、基本的なメモリー管理の概念を示し、オブジェクト割当ておよびガベージ・コレクションがOracle JRockit JVMでどのように機能するかを説明します。

▼ヒープ

Javaヒープは、Javaプロセス内に存在するJavaオブジェクトを格納するための領域です。 Javaヒープは、New世代領域、Old世代領域およびPermanent世代領域に大別され、Java VMが管理・制御します。

▼GC

Java言語では、Javaオブジェクトに対するメモリ領域の割り当てや解放をJVM（Java仮想マシン）が自動的に行います。 この自動解放メカニズムを「ガベージ・コレクション」と呼びます。

▼GoF

GoF（ゴフ）とは”The Gang Of Four（4人の奴等）”の略であり、エリック・ガンマ、リチャード・ヘルム、ラルフ・ジョンソン、ジョン・ブリシディースの4人を指しています。 ... 邦題は「オブジェクト指向における再利用のためのデザインパターン（ソフトバンクパブリッシング刊）」です。

▼デザインパターン

デザインパターンとは、JavaやRubyなどのオブジェクト指向の言語で使われる設計パターンのことです。 ... 「生成」や「構造」、「振る舞い」の3つに分類された、23のパターンがあります。 デザインパターンは体系だっているので、ビギナークラスのエンジニアがオブジェクト指向を学ぶ題材としても役に立つことでしょう。

■ その他

▼オブジェクト指向

オブジェクト指向とは、コンピュータプログラムの設計や実装についての考え方の一つで、互いに密接に関連するデータと手続き（処理手順）をオブジェクト（object）と呼ばれる一つのまとまりとして定義し、様々なオブジェクトを組み合わせて関連性や相互作用を記述していくことによりシステム全体を構築していく手法。

▼バッチ処理

バッチ処理とは、複数のプログラムからなる作業において、あらかじめ一連の手順を登録しておき、まとめて連続的に実行する方式。または、一定期間や一定量ごとにデータをまとめて一括して処理する方式。

▼コンパイル

コンパイルとは、プログラミング言語で書かれたコンピュータプログラム（ソースコード）を解析し、コンピュータが直接実行可能な形式のプログラム（オブジェクトコード）に変換すること。そのためのソフトウェアをコンパイラ（compiler）という。

▼カバレッジ

ソフトウェア開発において、出来上がったプログラムのテストをする際に、どの程度をテスト対象とする（ことができる）かをカバレッジ（テストカバレッジ）という。範囲の定義の仕方によりいくつかの種類があり、命令全体のうちテストできるものの比率を「ステートメントカバレッジ」（命令網羅率）、コード内の分岐のうちテストできるものの比率を「ブランチカバレッジ」（分岐網羅率）、コード内に記述された条件のうちテストできるものの比率を「コンディションカバレッジ」（条件網羅率）、とそれぞれ呼ぶ。

▼プラットホーム

プラットフォームとは、台、壇、台地、高台、舞台、演壇、乗降場、綱領などの意味を持つ英単語。ITの分野では、ある機器やソフトウェアを動作させるのに必要な、基盤となる装置やソフトウェア、サービス、あるいはそれらの組み合わせ（動作環境）のことをプラットフォームという。