

Design

プロダクトの設計をします

テーマ

「コンサートチケット販売プラットフォーム」をテーマに学習します

モデリング

ユースケース図

ユースケース図は、アクターとアクト、システムという要素からなるモデリング手法です

システムが何をするべきか、どのアクター同士が関わるかを明確にすることで、設計・実装の指針となります

このプラットフォームには以下の要素が存在します

- アクター
 - コン서트チケット販売者
 - コン서트チケット購入者
- システム
 - コン서트チケット販売プラットフォーム
- アクト
 - コンサートを登録する
 - チケットを購入する
 - コンサート情報を保持する
 - コンサート情報を確認する

これを[docs/design/ch01/usecase](#)に図示したので確認してください

シーケンス図

ユースケース図は、アクターとシステムという要素からなるモデリング手法です

ユースケース図で定義したアクト毎に、アクターがどのようなふるまいをするのかを時系列順に整理します

シーケンス内の条件分岐、終了判定を記載するなど、ユースケース図よりもアクトに対しての解像度が高くなります

ここでアクターがシステムに干渉するためのインターフェース（関数など）を定義します

これを[docs/design/ch01/sequence](#)に図示したので確認してください

システム設計

プロダクトにおけるシステムがどのような要件を満たすことで要求を実現するのかを決定します

特にスマートコントラクト開発においては、上記のアクトに対応するエントリーポイントとなる関数、ブロックチェーン上に保存される変数（型や命名）を決定します（これは一般的なアプリケーションとDBの設計をする場合と掃除関係にあります）

ch01の実現に必要なシステム設計をdocs/design/ch01/systemに記載したので確認してください

上記を完了したら[003-test](#)に進んでください