

# - Dokumentacja techniczna projektu -

## ***Crypto Wallet & Exchange***

Autorzy: *Eryk Szymański, Jarosław Uzarowicz*

---

Projekt to strona internetowa marketu kryptowalutowego, pozwalająca użytkownikowi na założenie własnego portfela, który daje możliwość zakupu, sprzedaży i transferu kryptowalut z użyciem fikcyjnego salda gotówkowego, uzupełnianego przez użytkownika. W aplikacji wszystkie dane zapisywane są do bazy MySQL. Ponieważ projekt zakłada implementację warstwy wierzchniej typowego marketu kryptowalutowego, czyli element strony internetowej, system nie posiada integracji z API blockchainów, które posiadałby w rzeczywistości.

Aplikacja została stworzona z wykorzystaniem frameworku **ASP .NET Core MVC 6.0.402**.

W systemie możemy wyróżnić **cztery główne modele**:

- ***Kryptowaluta***
  - Jest to obiekt posiadający pola:
    - nazwa,
    - opis,
    - wartość w złotych,
    - logo
  - Strona główna aplikacji, czyli miejsce w którym znajdzie się użytkownik po wejściu, jest listą dostępnych kryptowalut
  - Użytkownik przed zalogowaniem może wyświetlać listę, a także sprawdzać detale pojedynczych obiektów
  - Po zalogowaniu, klient aplikacji posiadający już portfel z saldem gotówkowym ma możliwość zakupu kryptowalut, sprzedaży ich oraz transferu do innych użytkowników systemu
  - Dodawanie, edycja oraz usuwanie obiektów typu kryptowaluta, jest wyłączną domeną Administratora systemu

- **Portfel**

- Obiekt ten posiada pola:
  - nazwa,
  - saldo gotówkowe,
  - lista kryptowalut (saldo kryptowalutowe),
  - właściciel
- Moduł aplikacji zawierający portfele jest dostępny tylko dla użytkowników zalogowanych
- Swoimi portfelami może zarządzać właściciel konta, czyli dodawać, edytować oraz usuwać
- Użytkownik ma możliwość dodania fikcyjnego salda gotówkowego, które następnie pozwala na zakup kryptowalut
- Lista portfeli wyświetla informacje o całkowitym saldzie kryptowalutowym, natomiast po przekierowaniu do detali danego portfela, możemy zobaczyć poszczególne kryptowaluty w nim zawarte

- **Transfer (Transakcja)**

- Jest to obiekt posiadający pola:
  - ilość wysyłanej kryptowaluty,
  - wiadomość,
  - użytkownik wysyłający,
  - portfela wysyłającego,
  - użytkownik odbierający,
  - portfel odbiorcy,
  - przesyłana kryptowaluta
- Transfer może być utworzony przez zalogowanego użytkownika, podczas tworzenia go, wysyłający ma możliwość wybrania z list: wysyłanej kryptowaluty, swojego portfela, z którego wyjdzie transfer oraz portfela odbiorcy. Podczas tworzenia transakcji, odwołanie do konta użytkownika odbiorcy jest automatycznie pobierane na bazie informacji o właścicielu portfela odbierającego.

- Żaden typ użytkownika nie może edytować transferu, natomiast Administrator ma możliwość usunięcia transferu z historii, jednak nie odwróci to efektów transferu, taka operacja musiałaby być wykonana ręcznie.
- **Użytkownik**
  - Podstawowy model użytkownika dostępny w rozwiązaniu Microsoft Identity Framework
  - Role użytkowników, które można wyróżnić w systemie to *User*, czyli klient aplikacji oraz *Admin*, czyli administrator

System posiada zabezpieczenia, przed podstawowymi błędami takimi jak podanie ujemnej czy zerowej wartości lub ilości kryptowaluty podczas zakupu, sprzedaży, transferu i dodawania do bazy danych. Większość zabezpieczeń oraz walidacji akcji i obiektów odbywa się w odpowiednich kontrolerach, oraz jest zawarta w modelach. Elementy walidacji treści dostępnych tylko dla Administratora lub dla użytkowników zalogowanych, odbywa się w kontrolerach oraz plikach szablonów stron \*.cshtml na bazie modułu autoryzacji oraz ról użytkowników.