



**UNIVERSITATEA TEHNICĂ**  
DIN CLUJ-NAPOCA

Platforma online pentru gestionarea de valuta virtuala  
(Crypto Virtual Waller)

Name : Hudisteanu Mihai-Andrei  
Group : 30232

# Deliverable 1

## Project Specification

Ideea din spatele din acestei aplicatii este de a crea un serviciu care sa faciliteze vanzarea si cumpararea de asseturi virtuale pentru oricine, prin prezentarea volumului mare de date in grafice si prin crearea de recomandari in functie de variatiile de pret de pe piata.

## Functional Requirements

Login

Register a new account

Buying Crypto

Selling Crypto

Transactions between Users

Vizualizarea variatiilor in pret pe piata

Vizualizarea variatiilor in pret pentru portofelul propriu

Calcularea automata de profit

Vizualizarea istoricului personal de tranzactii

Generare de statistici pe baza istoricului de tranzactii

Trimitere de notificari setate de user in functie de variatia de pret

# Use Case Model

## Use Cases Identification

Use-Case: Login

Level:

Primary Actor: User / Admin

Main success scenario: Tranzitie spre pagina principala a utilizatorului care s a logat cu succes sau a adminului

- Extensions:
  - Verificare User/Parola
  - Display error in caz de dat invalide
  - Generare de JWT si trimiterea lui inapoi la Client odata cu Login Response

Use-Case: Register

Level:

Primary Actor: Literally Anyone

Main success scenario: Salvarea credentialelor criptate in baza de date si redirectionarea catre LoginScreen

- Extensions:
  - Verificare User/Parola/Email
  - Display error in caz de dat invalide / userul exista deja (User/Admin)

Use-Case: View marketplace

Level:

Primary Actor: User

Main success scenario: Decide pagina principal reprezentata de un tabel cu toate posibilitatile de cumparare, preturi si un scurt istoric al prețurilor din ultima luna

Extensions : Display data loading error ...

Use-Case: View Currency

Level:

Primary Actor: User

Main success scenario: Vizualizarea tuturor informatiilor pe care le detin serverul pentru Currency-ul respectiv pentru informarea userului

- Extensions:
  - Loading Error / Server is down
  - Cumpararea de valuta noua
  - Verificarea legitimitatii tranzactiei

Use-Case: View PersonalWallet

Level:

Primary Actor: User

Main success scenario: Vizualizarea datelor despre conturile personale/ tranzactii / preturi

Extensions: Loading Error / Server is down

Use-Case: View History/Statistics

Level:

Primary Actor: User

Main success scenario: Vizualizarea tuturor datelor despre activitatea personală în cadrul aplicației

Extensions: Loading Error / Server is down

Use-Case: Set Annotations

Level:

Primary Actor: User

Main success scenario: Setarea de anotatii pentru a anunta userul despre ce se intampla pe aplicatie(a crescut/scazut pretul la un asset) sau remindere

Extensions: Loading Error / Server is down

Use-Case: Buy more

Level:

Primary Actor: User

Main success scenario: Cumpararea unui volum de valuta pe care userul deja il detine, setare de treshhold la care aplicatia vinde/cumpara un asset automat

Extensions: Loading Error / Server is down / Translation error

Use-Case: Sell

Level:

Primary Actor: User

Main success scenario: vanzarea unui volum de valuta pe care userul deja il detine, setare de treshhold la care aplicatia vinde/cumpara un asset automat

Extensions: Loading Error / Server is down / Translation error

Use-Case: Crud Currency

Level:

Primary Actor: Admin

Main success scenario: Posibilitatea de a updata orice proprietate a unui currency

Extensions: Loading Error / Server is down / Translation error / database error

Includes : Autorizarea userului si verificarea acestuia

Use-Case: Crud User

Level:

Primary Actor: Admin

Main success scenario: Posibilitatea de a updata orice proprietate a unui user

Extensions: Loading Error / Server is down / Translation error / database error

Includes : Autorizarea userului si verificarea acestuia

Use-Case: Crud Transactions

Level:

Primary Actor: Admin

Main success scenario: Posibilitatea de a updata orice proprietate a unei tranzactii

Extensions: Loading Error / Server is down / Translation error / database error

Includes : Autorizarea userului si verificarea acestuia

Use-Case: Define Admin

Level:

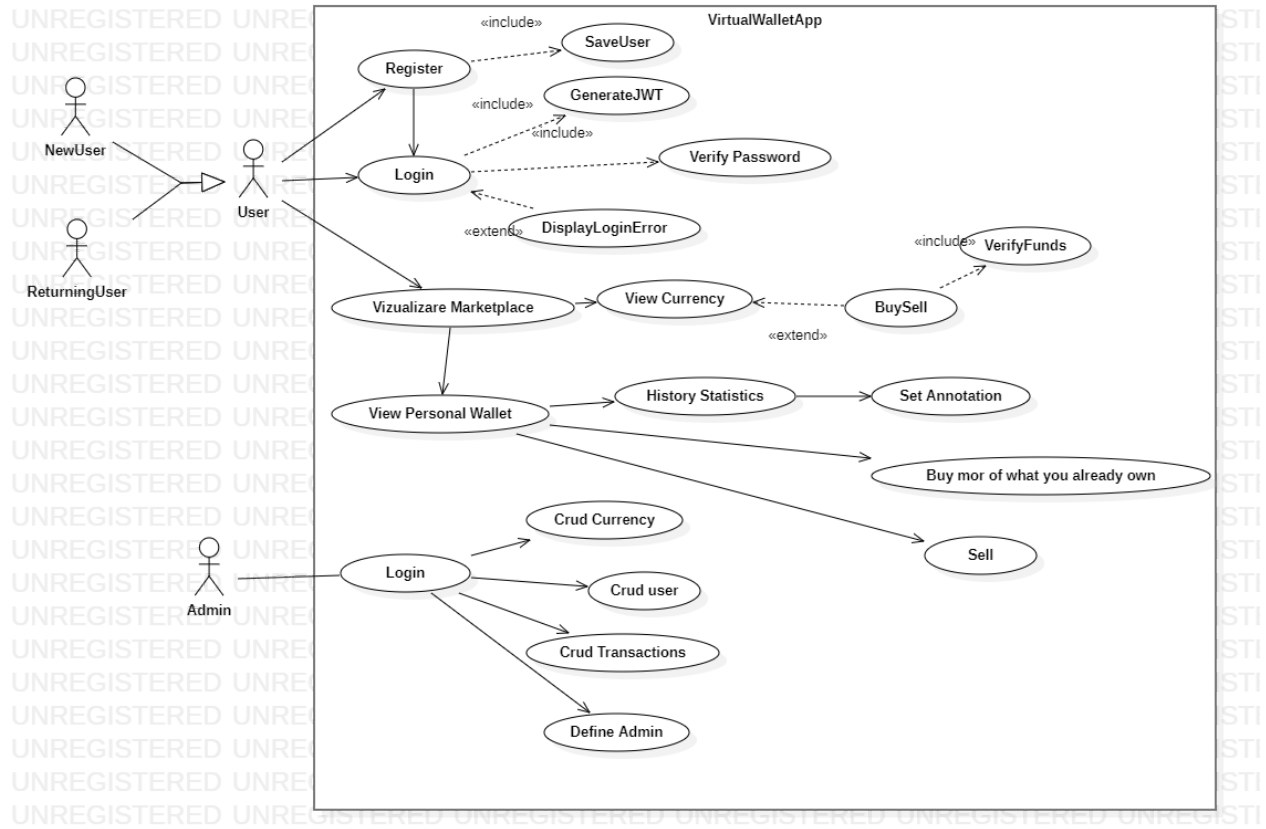
Primary Actor: Admin

Main success scenario: Posibilitatea de a creea mai multi admini

Extensions: Loading Error / Server is down / Translation error / database error

Includes : Autorizarea userului si verificarea acestuia

## UML Use Case Diagrams



# Supplementary Specification

## Non-functional Requirements

[Choose 4 NF for your system, describe them and explain why these NF are suitable for your implementation. ]

Sistemul de Logare va fi securizat prin Spring Security. Criptarea parolelor in baza de date reprezinta o necesitate in aceasta aplicatie, deoarece se ocupa cu tranzactii de valuta. De asemenea identificarea userilor printr-un token generate la fiecare logare de catre server poate face diferenta intre ce poate si ce nu poate un user sa acceseze faza de un administrator.

Avand in vedere volatilitatea pietei aplicatie si numarul mare de asseturi care apar va trebui construita astfel incat modificarile ulterioare sa fie cat mai facile, din acest motiv aplicatia va fi construita pe layere separate care pot fi extinse cu usurinta.

O mare problema pe care alte aplicatii care se ocupa cu cupararea/vanzarea de valuta o prezinta este volumul mare de informatie si pasi pe care userul trebuie sa le faca la fiecare tranzactie. Din acest motiv acest tip de aplicatii devin intimidante pentru un avarage user. Pentru a nu suferi de acest inconvenient am decis sa pastrez UI-ul aplicatie cat de simplu posibil, lasand totodata userul sa acceseze date suplimentare daca acesta decide ca are nevoie de mai multe informatii pentru a face o investitie informata.

Pentru a nu creea componente diferite la nivel de Frontend pentru User si Admin am decis ca ambele tipuri de utilizatori sa le fie prezentat acelasi portal, diferenta intre cei doi fiind facuta de backend, acesta lasand doar userii care sunt admini sa acceseze anumite pagini. (din cate m-am informat e posibil dar nu stiu daca pot sa o duc pana la capat)

O alta mare problema a unei astfel de aplicatii este ca piata de cryptocurrencies functioneaza non stop (fata de 5/7 zile in cazul bursei) din acest motiv preturile si eventualele modificari la aplicatie trebuie facute "on the run". Actualizarea datelor(preturi) ar trebui de asemenea updatata cat mai des atat pe client cat si pe server.

Usability - The principle of least astonishment (cel mai bun UI nu surprinde, semana cu ce a folosit userul inainte)



## **Design Constraints**

[This section needs to indicate any design constraints on the system being built. Design constraints represent design decisions that have been mandated and must be adhered to. Examples include software languages, software process requirements, prescribed use of developmental tools, architectural and design constraints, purchased components, class libraries, and so on.]

Timp

Integration - Numarul de apeluri care pot fi facute la API-urile folosite

## **Glossary**

[Present the noteworthy terms and their definition, format and validation rules if appropriate.]

# **Deliverable 2**

## **Domain Model**

[Define the domain model and create the conceptual class diagrams]

## **Architectural Design**

### **Conceptual Architecture**

[Define the system's conceptual architecture; use an architectural style and pattern - highlight its use and motivate your choice.]

## **Package Design**

[Create a package diagram]

## **Component and Deployment Diagram**

[Create the component and deployment diagrams.]

# **Deliverable 3**

## **Design Model**

### **Dynamic Behavior**

[Create the interaction diagrams (1 sequence, 1 communication diagrams) for 2 relevant scenarios]

### **Class Diagram**

[Create the UML class diagram; apply GoF patterns and motivate your choice]

### **Data Model**

[Create the data model for the system.]

# **System Testing**

[Describe the testing methods and some test cases.]

## **Future Improvements**

[Present some features that apply to the application scope.]

## **Conclusion**

## **Bibliography**