

**CoinVerter**  
**System Design Document**  
**Versione 2.0**



Data: 25/11/2024

Progetto: CoinVerter	Versione: 2.0
Documento: System Design Document	Data: 25/11/2024

**Coordinatore del progetto:**

Nome	Matricola

**Partecipanti:**

Nome	Matricola
Pastore Alfredo	0512108925
Perna Alessandro	0512118963

Scritto da:	Pastore Alfredo
-------------	-----------------

**Revision History**

Data	Versione	Descrizione	Autore
14/11/2024	1.0	Creato scheletro documento	Alfredo Pastore
25/11/2024	2.0	Completato documento	Alfredo Pastore

# Indice

1.	INTRODUZIONE .....	4
1.1.	Scopo del sistema .....	4
1.2.	Design goals .....	4
1.2.1.	Criteri di performance .....	4
1.2.2.	Criteri di affidabilità .....	4
1.2.2.1	DG 1.2.2.1 .....	4
1.2.3.	Criteri di costo .....	<b>Errore. Il segnalibro non è definito.</b>
1.2.4.	Criteri di mantenimento .....	4
1.2.5.	Criteri di End user .....	4
1.3.	Definizioni, acronimi e abbreviazioni .....	5
1.4.	Riferimenti .....	5
2.	ARCHITETTURA SOFTWARE CORRENTE .....	5
3.	ARCHITETTURA SOFTWARE PROPOSTA .....	5
3.1.	Panoramica .....	5
3.2.	Decomposizione in sottosistemi .....	6
3.3.	Hardware/software mapping .....	6
3.4.	Gestione dei dati persistenti .....	7
3.4.1.	Carrello .....	7
3.4.2.	Account .....	7
3.4.3.	Portafoglio .....	7
3.4.4.	Prodotti .....	7
3.5.	Controllo degli accessi e sicurezza .....	8
3.6.	Controllo software generale .....	<b>Errore. Il segnalibro non è definito.</b>
3.7.	Boundary Conditions .....	8
3.7.1.	Avvio del sistema .....	8
3.7.2.	Spegnimento del sistema .....	8
3.7.3.	Failure di sistema .....	8
3.7.3.1	Failure grave .....	8
3.7.3.2	Failure non grave .....	8
4.	GLOSSARIO DEI SERVIZI DEI SOTTOSISTEMI .....	9

# 1. INTRODUZIONE

## *1.1.Scopo del sistema*

Il sistema mira a creare una piattaforma digitale unica dove poter comprare e gestire valute digitali e non eliminando quindi la necessità di avere account diversi in siti diversi con possibili accordi che possono influire sulle valute del cliente in modo imprevedibile

## *1.2.Design goals*

### **1.2.1. Criteri di performance**

#### **1.2.1.1 Tempi di Risposta**

Assicurare che il sistema risponda rapidamente alle richieste degli utenti.

#### **1.2.1.2 Ottimizzazione delle Risorse**

Minimizzare l'uso delle risorse di sistema (CPU, memoria, larghezza di banda) per garantire prestazioni elevate.

### **1.2.2. Criteri di affidabilità**

#### **1.2.2.1 Sicurezza dei dati**

Le password degli account devono essere preservate sul database utilizzando la cifratura "SHA-256".

### **1.2.3. Criteri di mantenimento**

#### **1.2.3.1 Architettura Scalabile**

Progettare il sistema in modo che possa gestire un aumento del carico di lavoro senza compromettere le prestazioni.

#### **1.2.3.2 Codice Chiaro e Documentato**

Scrivere codice ben strutturato e documentato per facilitare la manutenzione e gli aggiornamenti futuri.

### **1.2.4. Criteri di End user**

#### **1.2.4.1 Interazione agevole con il sistema**

Il web client deve prevedere un accesso, semplice, non ambiguo, con uno schema di colori adatto alla maggioranza degli schermi alle funzionalità offerte ai clienti, permettendo l'interazione anche agli utenti non esperti

### ***1.3. Definizioni, acronimi e abbreviazioni***

Termine	Significato
Cliente	Utente che ha un account e che utilizza la piattaforma per fare acquisti
Gestore Catalogo	Persona incaricata di mantenere lo shop digitale
Sito	Rappresenta il web client con cui interagisce l'utente
Catalogo	Elenco di prodotti disponibili all'acquisto
Prodotto	Oggetto disponibile per l'acquisto dal cliente
Account	Insieme di informazioni che il sistema registra sul database per riconoscere l'utente

### ***1.4. Riferimenti***

- <https://capex.com/it>
- <https://www.etoro.com>

## **2. ARCHITETTURA SOFTWARE CORRENTE**

Essendo questo un progetto di greenfield engineering non è presente un'architettura precedente che si mira a migliorare o espandere

## **3. ARCHITETTURA SOFTWARE PROPOSTA**

### ***3.1. Panoramica***

Il sistema sarà composto da 3 componenti principali:

Il web client;

Il server;

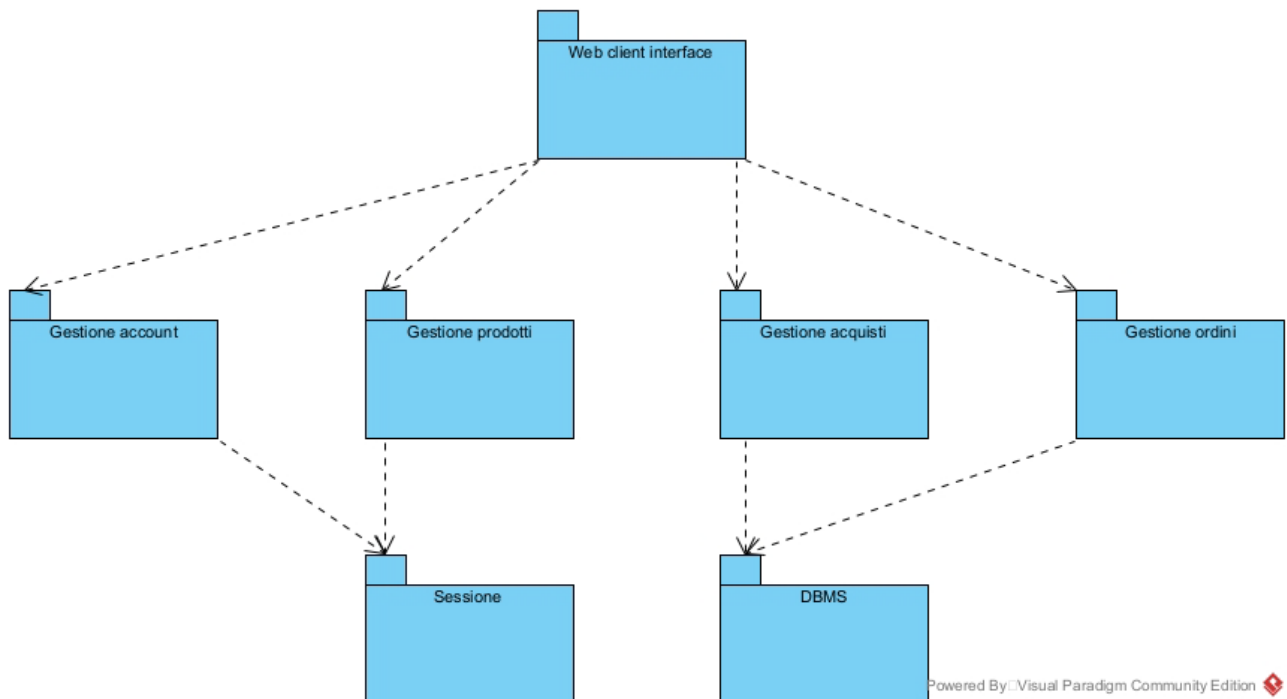
Il DBMS.

Il primo è una pagina web html con elementi css responsive per aumentare l'interattività e l'adattabilità della pagina e le cui funzionalità saranno implementate utilizzando javascript, json e comunicando con il server,

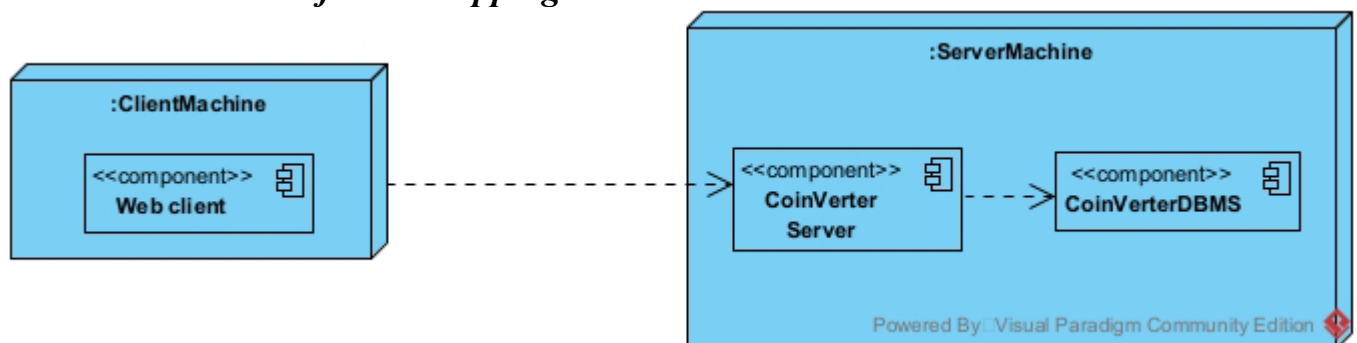
Il server sarà scritto in Java, utilizzando servlet dedicate per offrire i vari servizi e le necessarie computazioni lato server dando priorità a sicurezza e a velocità di risposta

Il DBMS verrà implementato in SQL e gestito tramite la sezione dedicata agli admin del portale web e verrà creato un account amministratore per la manutenzione esterna al sito

### 3.2. Decomposizione in sottosistemi



### 3.3. Hardware/software mapping



- Web Client interface  
Mapping: Software  
Tecnologie usate: HTML,CSS,Javascript  
Lo scopo di questo sottosistema è l'interazione con l'utente.  
Per come è strutturato il progetto l'unica componente necessaria al funzionamento di questo sottosistema è un web browser.
- Gestione account  
Mapping: Software  
Tecnologie usate: Java,Javascript  
In questo sottosistema sono racchiuse tutte le funzionalità di controllo degli account sia utente che di gestione. I controlli sugli account vengono fatti sia dal frontend client tramite javascript che sul server con apposite classi java.
- Gestione prodotti  
Mapping: Software

Tecnologie usate: HTML,CSS,Java,Javascript

Rappresenta la gestione del catalogo dei prodotti offerti, le funzionalità per il gestore catalogo sono offerti tramite specifiche pagine web e i vari controlli vengono effettuati anche in questo caso tramite java e javascript.

- **Gestione acquisti**

Mapping: Hardware,Software

Tecnologie usate: Java,MYSQL

Le responsabilità di questo sottosistema comprendono la gestione delle informazioni della carta del cliente oltre che alle query SQL necessarie per la registrazione dell'acquisto nella base dati.

Vista la criticità di questo sottosistema, in tutte le sue componenti vanno implementate misure di ridondanza e sicurezza

- **Gestione ordini**

Mapping: Software

Tecnologie usate: Java

- **Sessione**

Mapping: Software

Tecnologie usate: Java

La sessione è legata alla macchina dell'utente e non potendo avere certezze sullo spazio disponibile o sulle prestazioni la priorità va sull'efficienza

- **DBMS**

Mapping: Hardware,Software

Tecnologie usate: SSD,MYSQL

In questo sottosistema è compresa la componente hardware in cui verranno salvati i dati necessari al corretto funzionamento del progetto, pertanto una macchina con un uno spazio adeguato va riservata. L'importanza delle informazioni relative agli ordini va gestita con lo spostamento degli ordini antecedenti ad un anno in un archivio dedicato.

### ***3.4. Gestione dei dati persistenti***

#### **3.4.1. Carrello**

Il carrello viene gestito localmente tramite il web browser; gli elementi aggiunti ad esso vengono preservati salvando il carrello stesso in sessione

#### **3.4.2. Account**

Le informazioni relative agli account vengono salvate in un database relazionale

#### **3.4.3. Portafoglio**

Le informazioni legate al portafoglio di un utente sono legate al suo account e pertanto vengono preservate anch'esse su database

#### **3.4.4. Prodotti**

Le informazioni dei prodotti disponibili vanno mantenute anche alla chiusura di tutte le istanze del client e vanno mantenute ed aggiornate tramite un database

### 3.5. Controllo degli accessi e sicurezza

	Account	Prodotto	Acquisto	Ordine
Ospite	Register()	AddToCart() RemoveCart()	X	X
Cliente	Login() Logout()	AddToCart() RemoveCart()	Checkout()	X
Gestore catalogo	Login() Logout()	AddToCart() RemoveCart() AddProduct() RemoveProduct() ModifyProduct()	Checkout()	getOrder(filter)

### 3.6. Boundary Conditions

#### 3.6.1. Avvio del sistema

Il sistema verrà sviluppato così da poter essere avviato all'installazione e lavorare ininterrottamente fino ad uno spegnimento volontario da parte del cliente o fino ad una condizione eccezionale che ne costringe un momentaneo spegnimento.  
(Per gli avvii successivi a quello di installazione vedi 3.7.3.2)

#### 3.6.2. Spegnimento del sistema

In caso il cliente ritenga necessario spegnere il sistema, il cliente potrà utilizzare la funzione di spegnimento sicuro presente nel programma di gestione

#### 3.6.3. Failure di sistema

##### 3.7.3.1 Failure grave

In caso di una condizione di failure che impedisce completamente al sistema di riprendersi un tecnico verrà inviato per eseguire una diagnostica ed eventualmente riavviare il sistema

##### 3.7.3.2 Failure non grave

In casi in cui il problema sia facilmente risolvibile o in caso di semplice riavvio, verrà fornito al cliente un programma di gestione dal quale effettuare il riavvio del sistema



**4. GLOSSARIO DEI SERVIZI DEI SOTTOSISTEMI**

Gestione account

**Autenticazione**

Gestione prodotti

**Manutenzione catalogo**

Gestione acquisti

**Gestione carrello**

Gestione ordini

**Gestione storico ordini**