



Titolo Capitolato

Titolo Documento

Versione x.x in data xx-xx-xxxx

Gruppo 353

Informazioni sul documento

Responsabili	Elena Mattiazzo Valentina Marcon Mirco Cailotto
Redazione	Riccardo E. Giorato Gianluca Marraffa
Verifica	Parwinder Singh Davide Stocco
Stato	In corso
Uso	Esterno
Destinato a	Tullio Vardanega
Email di contatto	353swe@gmail.com

Diario delle modifiche

Versione	Data	Descrizione	Autore	Ruolo
0.0.3	02/12/17	Correzioni alle stesure dei casi d'uso	Gianluca Marraffa	Analista
0.0.2	01/12/17	Aggiunte tipologie d'attori, stesura casi d'uso principali	Mirco Cai-lotto	Analista
0.0.1	01/12/17	Stesura introduzione	Mirco Cai-lotto	Analista

Indice

1	Introduzione	1
1.1	Scopo del documento	1
1.2	Scopo del prodotto	1
1.3	Riferimenti	1
1.3.1	Riferimenti normativi	1
1.3.2	Riferimenti informativi	1
2	Descrizione generale	2
2.1	Obbiettivi del prodotto	2
2.2	Funzioni del prodotto	2
2.3	Caratteristiche degli utenti	2
2.4	Obblighi di progettazione	2
2.4.1	Requisiti minimi	2
2.4.2	Requisiti opzionali	2
3	Casi d'uso	3
3.1	Attori dei casi d'uso	3
3.1.1	Attori primari	3
3.1.2	Attori secondari	4
3.2	Elenco dei casi d'uso	5
3.2.1	UC1 - Breve guida	5
3.2.2	UC2 - Login	6
3.2.3	UC2.1 - Login automatico	7
3.2.4	UC2.2 - Visualizzazione messaggio di errore riguardo a chiave assente	8
3.2.5	UC2.3 - Visualizzazione messaggio di errore riguardo a chiave malformata	8
3.2.6	UC2.4 - Visualizzazione messaggio di errore riguardo a chiave non registrata	9
3.2.7	UC3 - Registrazione	9
3.2.8	UC4 - Logout	9

3.2.9	UC5 - Amministrazione	9
3.2.10	UC6 - Gestione aspetti relativi ai corsi	9
3.2.11	UC7 - Gestione aspetti relativi allo studente	9
4	Requisiti	10
4.1	Classificazione dei requisiti	10
4.1.1	Requisiti funzionali	10
4.1.2	Requisiti di qualità	13
4.1.3	Requisiti di vincolo	14
4.1.4	Requisiti prestazionali	15
4.2	Tracciamento	16
4.2.1	Tracciamento fonti-requisiti	16
4.2.2	Tracciamento requisiti-fonti	16
4.3	Riepilogo	16

1. Introduzione

1.1 Scopo del documento

1.2 Scopo del prodotto

1.3 Riferimenti

1.3.1 Riferimenti normativi

1.3.2 Riferimenti informativi

2. Descrizione generale

2.1 Obbiettivi del prodotto

2.2 Funzioni del prodotto

2.3 Caratteristiche degli utenti

2.4 Obblighi di progettazione

2.4.1 Requisiti minimi

2.4.2 Requisiti opzionali

3. Casi d'uso

3.1 Attori dei casi d'uso

3.1.1 Attori primari

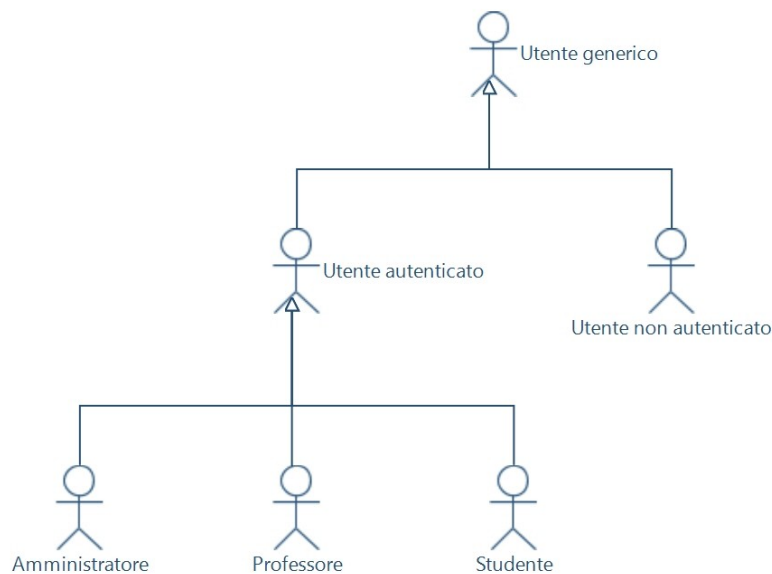


Figura 3.1: Gerarchia attori primari

1. **Utente generico**

Si riferisce ad un utente generico che accede al sito

2. **Utente non autenticato**

Ci si riferisce ad un utente generico con non ha ancora effettuato il login.

3. Utente autenticato

Ci si riferisce ad un utente generico con chiave valida ed autenticato nel sistema tramite la procedura di login.

4. Amministratore

Ci si riferisce ad un utente autenticato nel sistema nel ruolo di amministratore

5. Professore

Ci si riferisce ad un utente autenticato nel sistema nel ruolo di professore.

6. Studente

Ci si riferisce ad un utente autenticato nel sistema nel ruolo di studente

3.1.2 Attori secondari**1. MetaMask**

Plugin del browser MetaMask per interfacciarsi ad una rete Ethereum.

2. Ufficio Universitario

Entità fisica che consente l'immatricolazione e la registrazione dei professori.

3.2 Elenco dei casi d'uso

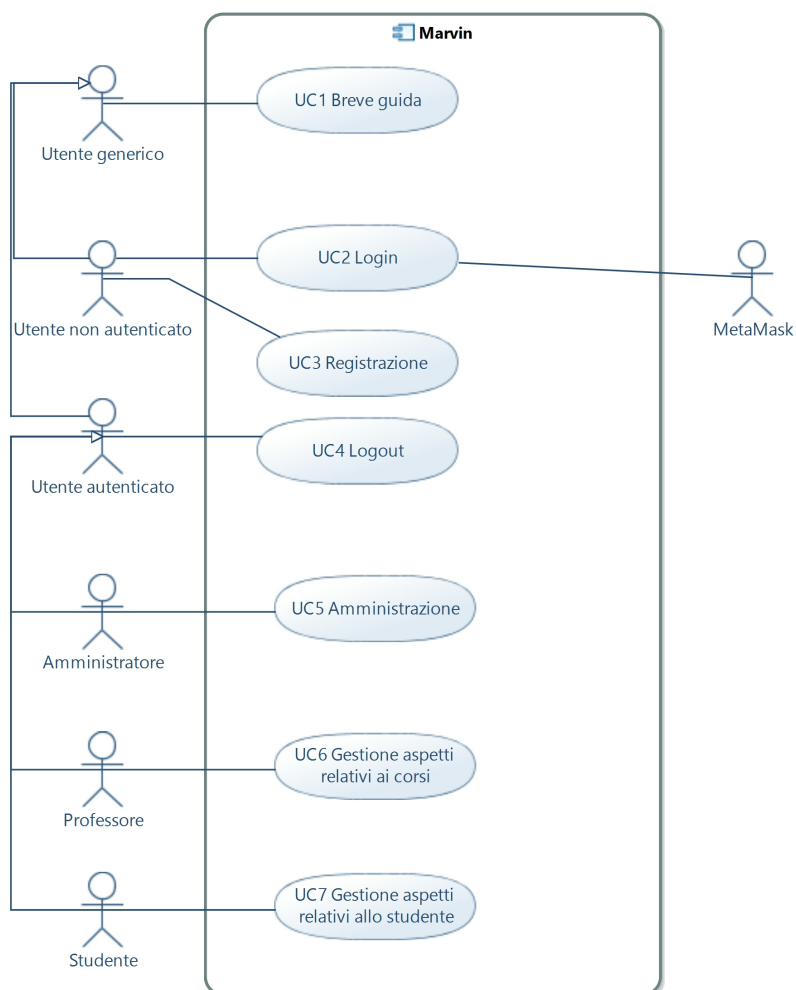


Figura 3.2: Casi d'uso basilari

3.2.1 UC1 - Breve guida

- **Attori primari:** Utente generico;
- **Scopo e descrizione:** L'utente visualizza una breve guida di introduzione su come installare il plugin MetaMask e su come gestire le chiavi in modo da istruirlo sulle modalità di accesso al sistema;

- **Scenario principale:** L'utente accede alla guida;
- **Precondizione:** Il sistema è raggiungibile e funzionante e l'utente desidera aprire la guida;
- **Postcondizione:** L'utente ha avuto delle nozioni riguardanti l'accesso al sistema.

3.2.2 UC2 - Login

- **Attori primari** Utente non autenticato;
- **Scopo e descrizione:** L'utente richiede il login al sistema attraverso il plugin MetaMask;
- **Scenario principale:** L'utente non ancora riconosciuto dal sistema effettua il login;
- **Precondizione:** L'utente non è stato riconosciuto dal sistema;
- **Postcondizione:** L'utente viene riconosciuto da parte del sistema.

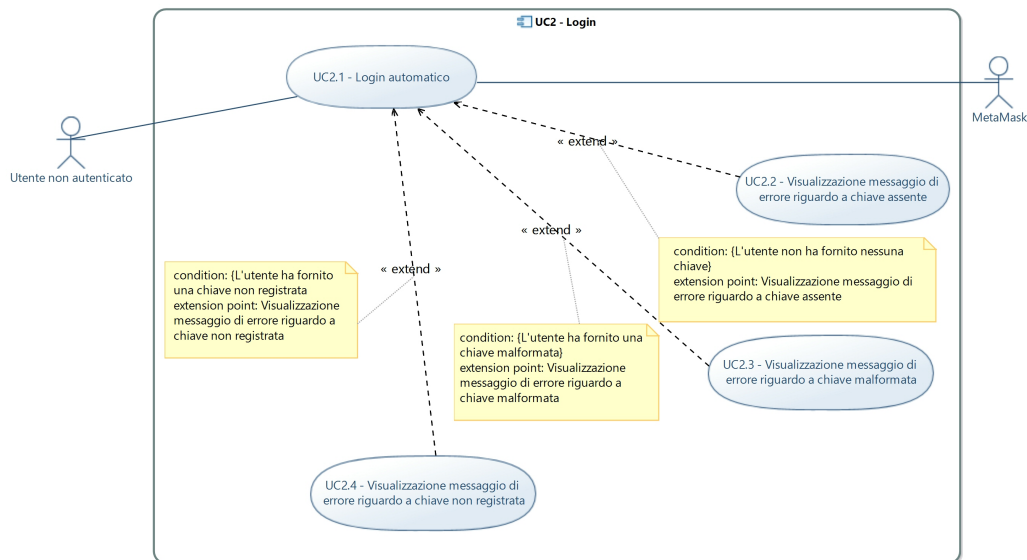


Figura 3.3: UC2 - Login

3.2.3 UC2.1 - Login automatico

- **Attori primari** Utente non autenticato;
- **Attori secondari:** MetaMask;
- **Scopo e descrizione:** L'utente attende il login da parte del sistema senza effettuare nessuna operazione aggiuntiva;
- **Scenario principale:** L'utente non ancora riconosciuto dal sistema richiede il login;
- **Estensioni:**
 1. Se l'utente non ha a disposizione una chiave viene avvisato con un errore a riguardo [UC2.2];
 2. Se l'utente ha a disposizione una chiave malformata viene avvisato con un errore a riguardo [UC2.3];

3. Se l'utente ha a disposizione una chiave non registrata viene avvisato con un errore a riguardo [UC2.4];

- **Precondizione:** L'utente ha richiesto al sistema di venire riconosciuto;
- **Postcondizione:** L'utente viene riconosciuto da parte del sistema.

3.2.4 UC2.2 - Visualizzazione messaggio di errore riguardo a chiave assente

- **Attori primari:** Utente non autenticato;
- **Scopo e descrizione:** L'utente viene avvisato del fatto che non ha fornito nessuna chiave al sistema;
- **Scenario principale:** L'utente visualizza l'errore relativo all'assenza di una chiave per accedere al sistema;
- **Precondizione:** L'utente richiede il login senza fornire una chiave;
- **Postcondizione:** L'utente è consapevole di dover fornire una chiave.

3.2.5 UC2.3 - Visualizzazione messaggio di errore riguardo a chiave malformata

- **Attori primari:** Utente non autenticato;
- **Scopo e descrizione:** L'utente viene avvisato del fatto che ha fornito una chiave malformata, quindi con lunghezza invalida o contenente caratteri non permessi;
- **Scenario principale:** L'utente visualizza il messaggio d'errore relativo alla chiave malformata;

- **Precondizione:** L'utente richiede il login utilizzando una chiave malformata;
- **Postcondizione:** L'utente è consapevole di cambiare la chiave o sistemare quella fornita.

3.2.6 UC2.4 - Visualizzazione messaggio di errore riguardo a chiave non registrata

- **Attori primari:** Utente non autenticato;
- **Scopo e descrizione:** L'utente viene avvisato del fatto che ha fornito una chiave non registrata nel sistema;
- **Scenario principale:** L'utente viene informato che la sua chiave non risulta registrata, suggerendone la registrazione;
- **Precondizione:** L'utente richiede il login utilizzando una chiave non registrata;
- **Postcondizione:** L'utente è consapevole di dover effettuare la registrazione per accedere al sistema.

3.2.7 UC3 - Registrazione

3.2.8 UC4 - Logout

3.2.9 UC5 - Amministrazione

3.2.10 UC6 - Gestione aspetti relativi ai corsi

3.2.11 UC7 - Gestione aspetti relativi allo studente

4. Requisiti

4.1 Classificazione dei requisiti

I requisiti vengono classificati ed assegnati loro un identificativo univoco secondo quanto definito nel documento *Norme di progetto v1.0.0*.

4.1.1 Requisiti funzionali

Identificatore	Importanza	Descrizione	Fonti
R0F1	Obbligatorio	L'amministratore può gestire gli studenti	Interno
R0F1.1	Obbligatorio	L'amministratore può ottenere una lista di tutti gli studenti	Interno
R0F1.2	Obbligatorio	L'amministratore può confermare la registrazione di uno studente	Interno
R0F1.2.1	Obbligatorio	L'amministratore può assegnare un corso di laurea ad uno studente	Interno
R0F2	Obbligatorio	L'amministratore può gestire i professori	Interno
R0F2.1	Obbligatorio	L'amministratore può ottenere una lista di tutti i professori	Interno
R0F2.2	Obbligatorio	L'amministratore può confermare la registrazione di un professore	Interno

Identificatore	Importanza	Descrizione	Fonti
R0F3	Obbligatorio	L'amministratore può gestire i corsi dei vari corsi accademici per ogni anno accademico	Capitolato
R0F3.1	Obbligatorio	L'amministratore può aggiungere un anno accademico	Capitolato
R0F3.2	Obbligatorio	L'amministratore può ottenere una lista di gli anni accademici	Interno
R0F3.3	Obbligatorio	L'amministratore può aggiungere dei corsi di laurea ad ogni anno accademico	Capitolato
R0F3.3.1	Obbligatorio	L'amministratore può assegnare un numero di crediti da raggiungere per un determinato corso di laurea	Interno
R0F3.4	Obbligatorio	L'amministratore può ottenere una lista di tutti i corsi di laurea dato un anno accademico	Interno
R0F3.5	Obbligatorio	L'amministratore può aggiungere degli esami ad ogni corso di laurea	Capitolato
R0F3.5.1	Obbligatorio	L'amministratore può scegliere l'obbligatorietà di un esame	Interno
R0F3.5.2	Obbligatorio	L'amministratore può indicare il numero di crediti di un esame	Interno
R0F3.6	Obbligatorio	L'amministratore può ottenere una lista di tutti gli esami dato il corso di laurea	Interno
R0F3.7	Obbligatorio	L'amministratore può assegnare un professore ad ogni esame	———

Identificatore	Importanza	Descrizione	Fonti
R0F3.8	Obbligatorio	L'amministratore può ottenere l'informazione di quale professore è assegnato a un determinato esame	Interno
R0F4	Obbligatorio	Un professore può gestire gli esami a lui assegnati	Capitolato
R0F4.1	Obbligatorio	Un professore può ottenere la lista di tutti gli esami a lui assegnati	Capitolato
R0F4.2	Obbligatorio	Un professore può ottenere la lista di tutti gli studenti associati a un determinato esame	Capitolato
R0F4.3	Obbligatorio	Un professore può registrare un esito ad un dato esame ad un dato studente registrato a quell'esame	Capitolato
R0F5	Obbligatorio	Uno studente può gestire l'iscrizione agli esami	Capitolato
R0F5.1	Obbligatorio	Uno studente può vedere l'elenco degli esami del suo corso	Capitolato
R0F5.1.1	Obbligatorio	Uno studente può ottenere il numero di crediti assegnati ad un esame	Interno
R0F5.1.2	Obbligatorio	Uno studente può ottenere l'obbligatorietà di un esame	Interno
R0F5.1.3	Obbligatorio	Uno studente può ottenere lo stato di superamento di un dato esame	Interno
R0F5.1.4	Obbligatorio	Uno studente può ottenere il totale dei suoi crediti	Interno
R0F5.1.5	Obbligatorio	Uno studente può ottenere il numero di crediti da raggiungere per la laurea	Interno
R0F5.1	Obbligatorio	Uno studente può iscriversi ad un esame opzionale	Capitolato
R0F6	Obbligatorio	L'utente può effettuare il login	Interno

Identificatore	Importanza	Descrizione	Fonti
R0F7	Obbligatorio	L'utente può effettuare il logout	Interno

Tabella 4.1: Tabella dei requisiti funzionali

4.1.2 Requisiti di qualità

Identificatore	Importanza	Descrizione	Fonti
R0Q1	Obbligatorio	La progettazione e il codice devono seguire le norme e le metriche riportate nei documenti allegati X e Y	Interno
R0Q2	Obbligatorio	L'approccio di scrittura di JavaScript deve essere promise Centric Approach	Capitolato
R0Q2.1	Obbligatorio	L'applicativo non deve fare uso di callback	VE171122
R0Q3	Obbligatorio	Il codice deve attenersi al Airbnb Javascript style guide	Capitolato
R0Q4	Obbligatorio	Lo sviluppo deve essere supportato dall'utilizzo del tool ESLint	Capitolato
R0Q5	Obbligatorio	Dovrà essere fornito un manuale utente in lingua inglese che tratterà l'uso da parte di studenti e professori	VE171122
R0Q6	Obbligatorio	Dovrà essere fornito un manuale di deploy e di utilizzo da parte degli amministratori in lingua inglese	VE171122
R0Q7	Obbligatorio	Il codice sorgente deve essere pubblicato sulla piattaforma GitHub, BitBucket o GitLab	Capitolato

Tabella 4.2: Tabella dei requisiti di qualità

4.1.3 Requisiti di vincolo

Identificatore	Importanza	Descrizione	Fonti
R0V1	Obbligatorio	L'applicativo dovrà essere sviluppato attraverso l'uso di tecnologie web	Capitolato
R0V1.1	Obbligatorio	L'applicativo dovrà essere sviluppato con Node.js	Capitolato
R0V1.2	Obbligatorio	L'applicativo dovrà essere sviluppato con JavaScript 8 (ES8)	Capitolato
R0V1.3	Obbligatorio	L'applicativo dovrà essere sviluppato con il boilerplate Redux Minimal	Capitolato
R0V1.4	Obbligatorio	L'applicativo dovrà essere sviluppato utilizzando React	Capitolato
R0V1.5	Obbligatorio	L'applicativo dovrà essere sviluppato utilizzando Redux	Capitolato
R0V1.6	Obbligatorio	Il deploy del sito andrà eseguito utilizzando Surge.sh	Capitolato
R0V1.7	Desiderabile	È desiderabile l'utilizzo di SCSS in sostituzione a CSS	Capitolato
R0V2	Obbligatorio	Gli smart contract dovranno essere scritti in linguaggio Solidity	Capitolato
R0V3	Obbligatorio	La connessione alla rete Ethereum deve avvenire tramite MetaMask	Capitolato
R0V3.1	Obbligatorio	I test riguardanti gli smart contract dovranno essere eseguiti in una rete locale ed almeno in una rete pubblica	Capitolato
R0V3.2	Obbligatorio	Il deploy degli smart contract dovrà avvenire su rete locale testrpc e rete di test Ropsten	Capitolato

Identificatore	Importanza	Descrizione	Fonti
R2V3.4	Opzionale	È apprezzabile un deploy finale sulla rete principale di Ethereum	Capitolato
R0V4	Obbligatorio	Lo sviluppo degli smart contract dovrà avvenire utilizzando il framework Truffle	Capitolato
R0V5	Obbligatorio	L'applicativo deve essere accessibile ed utilizzabile dal browser Mozilla Firefox a partire dalla versione X	Interno
R0V6	Obbligatorio	L'applicativo deve essere accessibile ed utilizzabile dal browser Google Chrome a partire dalla versione X	Interno
R0V7	Obbligatorio	L'applicativo deve essere accessibile ed utilizzabile dal browser Microsoft Internet Explorer a partire dalla versione X e dal browser Microsoft EDGE	Interno
R1V8	Desiderabile	L'applicativo deve essere accessibile ed utilizzabile da un browser mobile, per le versioni supportate fare riferimento alle controparti PC di Firefox e Chrome	Capitolato
R1V9	Desiderabile	All'utente senza MetaMask deve essere fornito un tutorial base di accesso al sito	Interno
R0V10	Obbligatorio	Un utente non deve poter compiere azioni sul sistema senza aver fatto l'accesso ad esso	Capitolato

Tabella 4.3: Tabella dei requisiti di vincolo

4.1.4 Requisiti prestazionali

Identificatore	Importanza	Descrizione	Fonti

Tabella 4.4: Tabella dei requisiti prestazionali

4.2 Tracciamento

4.2.1 Tracciamento fonti-requisiti

Fonte	Requisiti

Tabella 4.5: Tabella di tracciamento fonti-requisiti

4.2.2 Tracciamento requisiti-fonti

Requisito	Fonti

Tabella 4.6: Tabella di tracciamento requisiti-fonti

4.3 Riepilogo