



B.smart - Progetto *Etherless*

Analisi dei requisiti

Informazioni essenziali

Nome documento	Analisi dei requisiti
Versione	1.0.0
Stato	Approvato
Redazione	Denis Benato Erik Nucibella Maurizio Andres Baggio
Verifica	Andrea Didonè Maria Morra
Approvazione	Luigi Sassaro
Uso	Esterno
Distribuzione	B.smart
Destinato a	Red Babel B.smart Prof. Tullio Vardanega Prof. Riccardo Cardin
Riferimento email	b.smart.swe@gmail.com



Registro delle modifiche

Versione	Data	Nominativo	Ruolo	Descrizione
1.0.0	2020-04-12	Luigi Sassaro	Responsabile	Approvazione documento
0.5.0	2020-04-12	Maria Morra	Verificatore	Verifica §4, §4.2, §4.3, §4.4, §4.5
0.4.5	2020-04-11	Erik Nucibella	Analista	Stesura §4.5
0.4.4	2020-04-11	Maurizio Andres Baggio	Analista	Stesura §4.4
0.4.3	2020-04-11	Denis Benato	Analista	Modifica requisiti funzionali §4
0.4.2	2020-04-11	Erik Nucibella	Analista	Stesura requisiti di vincolo §4.3
0.4.1	2020-04-11	Erik Nucibella	Analista	Stesura requisiti di qualità §4.2
0.4.0	2020-04-11	Andrea Didonè	Verificatore	Verifica §2, UC2, UC3, UC4, UC7, UC8, UC9, diagrammi UML, §4,
0.3.8	2020-04-10	Erik Nucibella	Analista	Stesura requisiti funzionali §4
0.3.7	2020-04-10	Maurizio Andres Baggio	Analista	Inizio stesura requisiti funzionali §4
0.3.6	2020-04-08	Erik Nucibella	Analista	Creazione tabelle requisiti §4
0.3.5	2020-04-08	Maurizio Andres Baggio	Analista	Stesura §4
0.3.4	2020-04-08	Denis Benato	Analista	Modifica diagrammi UML
0.3.3	2020-04-08	Erik Nucibella	Analista	Inserimento diagrammi UML



0.3.2	2020-04-08	Erik Nucibella	Analista	Modifica §2 e UC2
0.3.1	2020-04-08	Denis Benato	Analista	Modifica UC3, UC4, UC7, UC8 e UC9
0.3.0	2020-04-08	Maria Morra	Verificatore	Verifica UC1, UC2, UC3, UC4, UC5, UC6, UC7, UC8, UC10, UC11, UC12, UC13
0.2.9	2020-04-07	Maurizio Andres Baggio	Analista	Modifica UC6, UC10 e UC11
0.2.8	2020-04-07	Denis Benato	Analista	Modifica UC7 e UC8
0.2.7	2020-04-07	Maurizio Andres Baggio	Analista	Stesura UC13
0.2.6	2020-04-07	Erik Nucibella	Analista	Modifica UC2
0.2.5	2020-04-07	Erik Nucibella	Analista	Modifica UC3, UC4 e UC5
0.2.4	2020-04-07	Denis Benato	Analista	Modifica UC7.2, UC1, UC3 e UC6
0.2.3	2020-04-07	Maurizio Andres Baggio	Analista	Stesura UC12
0.2.2	2020-04-06	Erik Nucibella	Analista	Stesura UC1 e UC2
0.2.1	2020-04-05	Erik Nucibella	Analista	Modifica delle sezioni §1 e §2
0.2.0	2020-04-05	Andrea Didonè	Verificatore	Verifica UC3, UC4, UC5, UC6, UC7, UC8 e UC9
0.1.8	2020-04-05	Maurizio Andres Baggio	Analista	Stesura UC6, UC10 e UC11
0.1.7	2020-04-05	Denis Benato	Analista	Modifica UC3, UC4, UC7, UC8 e UC9
0.1.6	2020-04-05	Erik Nucibella	Analista	Aggiornata la struttura L ^A T _E X per la modellazione dei casi base e revisione UC5



0.1.5	2020-04-05	Erik Nucibella	Analista	Stesura UC5
0.1.4	2020-04-03	Erik Nucibella	Analista	Aggiornamento template di base e modifica UC7
0.1.3	2020-04-01	Denis Benato	Analista	Stesura UC7, UC8 e UC9
0.1.2	2020-03-30	Denis Benato	Analista	Stesura UC4, UC6
0.1.1	2020-03-30	Maurizio Andres Baggio	Analista	Modifica in §2
0.1.0	2020-03-30	Maria Morra	Verificatore	Verifica §1 §2
0.0.3	2020-03-24	Denis Benato	Analista	Stesura Descrizione Generale §2
0.0.2	2020-03-23	Denis Benato	Analista	Stesura Introduzione §1
0.0.1	2020-03-23	Erik Nucibella	Analista	Crea la struttura del documento in L ^A T _E X



Indice

1	Introduzione	8
1.1	Scopo del documento	8
1.2	Scopo del prodotto	8
1.3	Glossario	8
1.4	Riferimenti	8
1.4.1	Normativi	8
1.4.2	Informativi	8
2	Descrizione generale	9
2.1	Prospettive sul prodotto	9
2.2	Funzioni del prodotto	9
2.3	Caratteristiche degli utenti	9
2.3.1	Tipologie di utenti	9
2.3.2	Autenticazione	10
2.3.2.1	Registrazione	10
2.3.2.2	Login	10
2.3.2.3	Login persistente	11
2.4	Caratteristiche tecniche	11
2.4.1	Architettura dell'applicazione	11
2.4.2	etherless-cli : Breve manuale d'uso della <i>CLIG</i>	11
2.4.3	Conservazione ed archiviazione dei dati	12
2.4.4	Transazioni <i>Ethereum_G</i>	12
2.5	Tecnologie utilizzate	13
2.6	Vincoli Generali	13
3	Casi d'uso	14
3.1	Attori dei casi d'uso	14
3.1.1	Attori primari	14
3.1.2	Attori secondari	14
3.2	Elenco dei casi d'uso	14
3.2.1	UC1 - Visualizzazione guida dell'applicazione	14
3.2.2	UC2 - Inizializzazione ed autenticazione	15
3.2.2.1	UC2.1 - Rimozione dei dati salvati	16
3.2.2.2	UC2.2 - Registrazione	17
3.2.2.3	UC2.3 - Autenticazione	17
3.2.2.4	UC2.3.1 - Autenticazione: Inserimento address	18
3.2.2.5	UC2.3.2 - Autenticazione: Inserimento private key	18
3.2.2.6	UC2.3.3 - Autenticazione: Accesso ad Ethereum	19
3.2.2.7	UC2.3.4 - Autenticazione: Accesso persistente	19
3.2.2.8	UC2.3.4.1 - Accesso persistente: Attivazione	20
3.2.2.9	UC2.3.4.2 - Accesso persistente: Disattivazione	20
3.2.3	UC2.3.5 - Autenticazione: Visualizzazione errore "Credenziali errate"	20
3.2.4	UC3 - Verifica wallet in uso dall'applicazione	21
3.2.4.1	UC3.1 - Verifica <i>wallet_G</i> in utilizzo dall'applicazione	21
3.2.4.2	UC3.2 - Notifica di mancata autenticazione	22
3.2.5	UC4 - Elenco dei prodotti software disponibili in <i>Etherless_G</i>	22
3.2.6	UC5 - Verifica disponibilità di un prodotto software	23
3.2.6.1	UC5.1 - Inserimento nome prodotto software	23



3.2.6.2	UC5.2 - Visualizzazione errore "Prodotto software non trovato" . . .	24
3.2.7	UC6 - Richiesta di pubblicazione del proprio prodotto software	24
3.2.7.1	UC6.1 - Inserimento percorso file	25
3.2.7.2	UC6.2 - Visualizzazione errore "Percorso file errato"	26
3.2.7.3	UC6.3 - Inserimento nome prodotto software	26
3.2.7.4	UC6.4 - Visualizzazione errore "Nome già esistente"	26
3.2.8	UC7 - Richiesta di esecuzione di un prodotto software	27
3.2.8.1	UC7.1 - Inserimento nome prodotto software	28
3.2.8.2	UC7.2 - Visualizzazione errore "Prodotto software non disponibile"	28
3.2.8.3	UC7.3 - Inserimento dei parametri	29
3.2.8.4	UC7.4 - Visualizzazione errore di esecuzione del prodotto software	29
3.2.9	UC8 - Richiesta elenco delle operazioni richieste alla piattaforma	29
3.2.10	UC9 - Richiesta del risultato di una operazione	30
3.2.10.1	UC9.1 - Inserimento di un identificativo	31
3.2.10.2	UC9.2 - Visualizzazione errore "Identificativo non valido"	31
3.2.11	UC10 - Modifica del proprio prodotto software	32
3.2.11.1	UC10.1 - Inserimento nome prodotto software	33
3.2.11.2	UC10.2 - Visualizzazione errore "Prodotto software non esistente"	34
3.2.11.3	UC10.3 - Visualizzazione errore "Permessi di modifica"	34
3.2.11.4	UC10.4 - Inserimento percorso file	34
3.2.11.5	UC10.5 - Visualizzazione errore "Percorso file errato"	35
3.2.12	UC11 - Eliminazione del proprio prodotto software	35
3.2.12.1	UC11.1 - Inserimento nome prodotto software	36
3.2.12.2	UC11.2 - Visualizzazione errore "Prodotto software non esistente"	36
3.2.12.3	UC11.3 - Visualizzazione errore "Permessi di modifica"	37
3.2.13	UC12 - Visualizzazione errore "Utente non autenticato"	37
3.2.14	UC13 - Pagamento	38
3.2.14.1	UC13.1 - Visualizzazione errore "Fondi insufficienti per l'esecuzione"	39
4	Requisiti	40
4.1	Requisiti funzionali	41
4.2	Requisiti di qualità	48
4.3	Requisiti di vincolo	49
4.4	Tracciamento	50
4.4.1	Fonte - Requisito	50
4.4.2	Requisito - Fonte	53
4.5	Evoluzione attesa del sistema	56
4.6	Considerazioni	56



Elenco delle tabelle

4.1.1	Riassuntiva Requisiti Funzionali	41
4.2.1	Riassuntiva Requisiti di Qualità	48
4.3.1	Riassuntiva Requisiti di Vincolo	49
4.4.1	Riassuntiva Fonte - Requisiti	50
4.4.2	Riassuntiva Requisito - Fonti	53



Elenco delle figure

3.2.1	Diagramma UC2 - Inizializzazione ed autenticazione (init)	15
3.2.2	Diagramma UC2.3 - Autenticazione (init)	15
3.2.3	Diagramma UC3 - Verifica wallet in uso (who)	21
3.2.4	Diagramma UC5 - Verifica disponibilità di un prodotto software (find)	23
3.2.5	Diagramma UC6 - Richiesta di pubblicazione del proprio prodotto software (deploy)	24
3.2.6	Diagramma UC6 - Richiesta di pubblicazione del proprio prodotto software (deploy)	24
3.2.7	Diagramma UC7 - Richiesta di esecuzione di un prodotto software (run)	27
3.2.8	Diagramma UC7 - (Dettaglio) Richiesta di esecuzione di un prodotto software (run)	27
3.2.9	Diagramma UC9 - Richiesta del risultato di un operazione (log)	30
3.2.10	Diagramma UC10 - Modifica del proprio prodotto software (edit)	32
3.2.11	Diagramma UC10 - (Dettaglio) Modifica del proprio prodotto software (edit)	32
3.2.12	Diagramma UC11 - Eliminazione del proprio prodotto software (delete)	35
3.2.13	Diagramma UC11 - (Dettaglio) Eliminazione del proprio prodotto software (edit) .	35
3.2.14	Diagramma UC12 - Visualizzazione errore: utente non autenticato (edit)	37
3.2.15	Diagramma UC13 - Pagamento	38



1 Introduzione

1.1 Scopo del documento

Il presente documento ha lo scopo di descrivere in maniera dettagliata i requisiti individuati per il prodotto. Tali caratteristiche sono state identificate attraverso un'approfondita analisi del *capitolato_G* C2 ed i successivi incontri con il proponente, *Red Babel*.

1.2 Scopo del prodotto

Etherless_G è una *DApp_G* accessibile attraverso una *CLI_G*, che ha lo scopo di consentire agli utenti della piattaforma la pubblicazione del proprio software in ambiente *serverless_G*, permettendo a tutti gli utilizzatori del servizio di eseguirlo pagando un corrispettivo attraverso la rete *Ethereum_G*.

1.3 Glossario

Al fine di evitare possibili ambiguità relative al linguaggio utilizzato nei documenti formali, viene fornito il *Glossario 1.0.0*. In questo documento vengono definiti e descritti tutti i termini tecnici e/o specifici che necessitano di un possibile approfondimento. Per facilitare la lettura, i termini saranno contrassegnati in corsivo e da una *G* a pedice.

1.4 Riferimenti

1.4.1 Normativi

- **Norme di Progetto:** *Norme di Progetto 1.0.0*;
- **Capitolato d'appalto C2 - Etherless:**
<https://www.math.unipd.it/~tullio/IS-1/2019/Progetto/C2.pdf>.

1.4.2 Informativi

- **Studio di Fattibilità:** *Studio di Fattibilità 1.0.0*;
- **Capitolato d'appalto C2 - Etherless:**
<https://www.math.unipd.it/~tullio/IS-1/2019/Progetto/C2.pdf>;
- **Software Engineering - Ian Sommerville - 10th Edition 2014:**
 - Chapter 4: Requirements engineering;
- **Sito ufficiale della blockchain Ethereum:** <https://www.ethereum.org/>.
Il sito ufficiale di *Ethereum_G*, che contiene la presentazione dei concetti alla base dell'applicativo come la *blockchain_G*, gli *smart contract_G* ed il meccanismo di *escrow_G*;
- **Sito ufficiale di AWS Lambda:** <https://aws.amazon.com/it/lambda/>.
Descrizione del funzionamento, dei costi e dell'infrastruttura di *AWS Lambda_G*, facente parte della suite *AWS_G*.



2 Descrizione generale

2.1 Prospettive sul prodotto

Il progetto *Etherless* si pone come obiettivo quello di semplificare la **distribuzione** e l'**esecuzione** di un prodotto software accessibile tramite la realizzazione di una piattaforma *cloud_G* di tipo *FaaS_G*.

- **Distribuzione:** Obiettivo principale della piattaforma è quello di permettere ai propri utenti di pubblicare, e quindi distribuire il proprio software ad altri utenti, che possono visualizzare ed eseguire qualsiasi applicazione presente in esso, pagando un corrispettivo al proprietario del software oltre che alla piattaforma stessa;
- **Esecuzione:** Un qualsiasi utente della piattaforma, può liberamente visualizzare ed utilizzare tutto il software pubblicato da altri utenti della piattaforma, in modo rapido e funzionale. L'esecuzione comporta il pagamento di un corrispettivo alla persona (fisica o giuridica) che si rapporta con *AWS Lambda_G*, la quale a sua volta provvederà a pagare chi ha pubblicato il prodotto software, fornendolo in licenza d'uso alla piattaforma.

2.2 Funzioni del prodotto

L'applicativo deve fornire agli *utenti_G* un mezzo per distribuire i loro prodotti software, consentendo agli stessi un modo di trovare ed eseguire software specifici, senza la necessità di interagire direttamente con l'infrastruttura cloud (o *back-end_G* tradizionale) che quest'ultimi richiederebbero.

Quest'obiettivo viene raggiunto grazie all'utilizzo della rete *Ethereum_G* che regola il flusso di denaro e richieste tra la piattaforma e gli *utenti_G*.

Ogni utente può accedere ad *Etherless_G* attraverso il proprio account *Ethereum_G*, identificato attraverso un *address_G* e una *private key_G*, avendo anche la possibilità di creare rapidamente un account nel caso ne fosse sprovvisto.

2.3 Caratteristiche degli utenti

Etherless_G è un'applicativo pensato particolarmente per figure specifiche dell'ambito IT. L'utente tipo dell'applicazione, è infatti sottointeso essere uno sviluppatore o comunque una figura con conoscenze di base di programmazione e sviluppo software.

Per questo motivo, non è prevista alcuna distinzione tra l'account di un **utente utilizzatore** e di un **utente pubblicatore**, in quanto ogni utente della piattaforma è potenzialmente uno sviluppatore ed ha quindi la possibilità di eseguire le stesse operazioni di pubblicazione ed esecuzione delle funzioni.

2.3.1 Tipologie di utenti

È possibile individuare tre tipologie di utenti, suddivisi nel seguente modo:

- **Utente non autenticato:** ovvero che non è ancora autenticato ad *Etherless_G* in quanto non ha ancora immesso le credenziali del proprio *wallet_G* *Ethereum_G* nel sistema;
- **Utente autenticato:** ovvero che è autenticato ad *Etherless_G* avendo fornito e verificato le proprie credenziali *Ethereum_G* all'interno dell'applicazione;
- **Utente generico:** ovvero un utente il cui stato di autenticazione non è rilevante, rappresentabile anche come l'unione dei due casi precedenti **utente autenticato** e **utente non autenticato**.



L'applicazione richiede l'accesso alla rete *Ethereum_G*, e quindi obbligatoriamente la presenza di una connessione *Internet_G* per consentirne l'utilizzo anche di base.

Le funzionalità di un **utente non autenticato** sono quindi limitate, fino al momento dell'autenticazione alla rete *Ethereum_G*; esso può infatti:

- Visualizzare la guida dell'applicazione;
- Essere notificato della mancata autenticazione;
- Registrarsi alla rete *Ethereum_G*, richiedendo quindi un nuovo *wallet_G*;
- Accedere alla piattaforma attraverso le credenziali del proprio *wallet_G* *Ethereum_G*.

L' **utente autenticato** invece, dispone di tutti i requisiti fondamentali per l'utilizzo completo dell'applicazione, e può quindi:

- Visualizzare la guida dell'applicazione;
- Visualizzare l'**address_G**, identificativo del *wallet_G* con cui è loggato al sistema;
- Richiedere l'elenco di tutte le funzioni disponibili nella piattaforma;
- Eseguire una specifica funzione presente nel sistema;
- Pubblicare una propria funzione nella piattaforma;
- Modificare una funzione pubblicata di sua proprietà (nel caso ne fosse presente almeno una);
- Eliminare una funzione pubblicata di sua proprietà (nel caso ne fosse presente almeno una);
- Visualizzare la cronologia dei comandi eseguiti nella sessione corrente;
- Visualizzare i log (risultati) delle funzioni eseguite all'interno della piattaforma;
- Rimuovere i propri dati dall'applicazione, effettuandone il logout.

2.3.2 Autenticazione

L'autenticazione è un'azione fondamentale ed altrimenti bloccante nell'uso dell'applicazione.

La motivazione alla base di questo è data banalmente dalla necessità di dover disporre di un **wallet_G**, che è in grado di garantire la proprietà delle funzioni pubblicate, oltre a contenere l'**address_G** univoco necessario ad eseguire transazioni all'interno della *blockchain_G* *Ethereum_G*.

Etherless_G, dispone di diversi modi per accedere alla rete *Ethereum_G*, tra i quali la creazione di un *wallet_G* direttamente all'interno della piattaforma.

2.3.2.1 Registrazione

Nel caso non si disponesse di un *wallet_G*, la piattaforma fornisce un semplice comando che richiede alla rete *Ethereum_G* la creazione di quest'ultimo.

2.3.2.2 Login

È possibile accedere all'applicazione utilizzando un *wallet_G* esistente semplicemente fornendo le credenziali richieste all'avvio della stessa.



2.3.2.3 Login persistente

Quando l'utente crea un nuovo *wallet_G* o esegue l'accesso, la piattaforma chiede se attivare il login persistente o meno. Il sistema può infatti salvare i dati del *wallet_G* nel *filesystem locale_G* per consentire un *login automatico*, che può sempre essere rimosso tramite il comando di inizializzazione della applicazione.

Il login persistente non è obbligatorio, ma fortemente consigliato in quanto consente di poter utilizzare i comandi dell'applicazione **senza dover ripetere la procedura di autenticazione ad ogni nuova sessione**.

2.4 Caratteristiche tecniche

2.4.1 Architettura dell'applicazione

Come richiesto dal proponente, l'applicazione si basa su tre componenti fondamentali ed indipendenti:

- **etherless-cli**: è il modulo che permette l'interfacciamento dell'utente con la piattaforma *Etherless_G*. Fornisce le funzionalità descritte nella sezione precedente, partendo dalla creazione di un account *Ethereum_G*, arrivando alla visualizzazione del risultato della funzione richiesta;
- **etherless-smart**: componente che gestisce l' *escrow_G* e le transizioni sulla rete *Ethereum_G*, tramite l'utilizzo di *smart contract_G*;
- **etherless-server**: è l'applicativo server che si interfaccia direttamente con *AWS Lambda_G*, riceve i comandi di *etherless-cli_G* ed arbitra l'esecuzione del software in base ad **etherless-smart**.

2.4.2 etherless-cli: Breve manuale d'uso della *CLI_G*

Di seguito l'elenco dei comandi messi a disposizione dall' **etherless-cli** di *Etherless_G* utilizzabili dagli utenti:

- **etherless**: visualizza la guida dell'applicazione, la licenza d'uso e il nome degli autori;
- **etherless help**: visualizza la guida dell'applicazione;
- **etherless init**: reset/avvio dell'applicativo che richiede l'autenticazione tramite il *wallet_G*;
- **etherless list**: visualizza l'elenco delle funzioni disponibili ed utilizzabili in *Etherless_G*;
- **etherless who**: visualizza l'identificativo **address** relativo al *wallet_G* con il quale si ha eseguito l'accesso;
- **etherless find funName**: richiede la visualizzazione dettagliata della funzione **funName**. Ha successo solamente se la funzione esiste in *Etherless_G*;
- **etherless deploy fun.js funName**: pubblica la funzione **fun.js** presente nel *filesystem locale_G* dell'utente nella piattaforma *Etherless_G*, utilizzando come identificativo il nome **funName**;
- **etherless run funName [params]**: richiede l'esecuzione della funzione **funName** utilizzando i parametri specifici della funzione (quindi non sempre obbligatori) **params**;
- **etherless log [id]**: ritorna il dettaglio dell'esecuzione della funzione caratterizzata da **[id]**, codice che caratterizza una funzione eseguita reperibile tramite il comando **history**;
- **etherless history**: visualizza una lista contenente le ultime funzioni eseguite in realzione al *wallet_G* collegato, includendo un **id** univoco caratterizzante ognuna di essa;



- **etherless remove funName**: richiede la rimozione della funzione di propria proprietà **funName**, può avere successo solamente se l'utente detiene la proprietà della funzione specificata;
- **etherless edit funName fun.js**: richiede la modifica della funzione di propria proprietà **funName**, sostituendola con quella contenuta nel file **fun.js**. Può avere successo solamente se l'utente detiene la proprietà della funzione specificata.

2.4.3 Conservazione ed archiviazione dei dati

L'applicazione non memorizza i dati del *wallet_G* dell'utente, se non in locale se e solo se specificato dall'utente. Per garantire il proprio funzionamento il sistema necessita però di memorizzare dati come il codice della funzione pubblicata.

Per fare questo, *Etherless_G* non si appoggia a database esterni ma utilizza il *ledger_G* di *Ethereum_G*, considerato il cuore pulsante della *blockchain_G* stessa.

Questo tipo di approccio garantisce privacy ed integrità dei dati, consentendo di associare l'**address** del *wallet_G* dell'utente con le funzioni da lui pubblicate, grazie all'utilizzo e alla modellazione degli *smart contract_G*.

Nel caso ci fosse la necessità di archiviare grandi quantità e complessi tipi di dati, si valuterà l'utilizzo di un servizio di database come *AWS S3_G*.

2.4.4 Transazioni *Ethereum_G*

Per garantire il suo funzionamento *Etherless_G* si basa sul meccanismo di *escrow_G* utilizzabile tramite gli *smart contract_G*, che permettono un pagamento sicuro in *ETH_G* da parte degli utenti verso la piattaforma, che si occuperà di eseguire la funzione selezionata in ambiente *serverless_G*.

Le transazioni sono un punto focale dell'applicazione, permettono di pagare chi pubblica le funzioni oltre a coprire i costi di mantenimento dati da AWS Lambda.

Lo sviluppatore, si impegna anche a coprire il costo *una tantum* della pubblicazione o aggiornamento del suo prodotto software, dovuto alla necessità di scrittura nella blockchain della rete *Ethereum_G*, che ha un costo ogni qual volta viene effettuata una scrittura.



2.5 Tecnologie utilizzate

Di seguito gli strumenti e le tecnologie richieste e scelte per la realizzazione del progetto:

- **Servizi**
 - **AWS Lambda**: servizio *serverless_G* che consente l'esecuzione di codice sulla piattaforma scalabile di AWS;
 - **Serverless Framework**: strumento per la creazione ed il rilascio di applicazioni in ambienti *serverless_G*;
- **Linguaggi e pacchetti**
 - **ESLint**: strumento per l'analisi statica del codice Javascript/Typescript;
 - **Node.js**: ambiente opensource utilizzato solitamente in accoppiata con **express** per l'esecuzione di codice Javascript lato server;
 - **Solidity**: linguaggio *object-oriented* e *strong-typed* utilizzato per la creazione di *smart contract_G*;
 - **Typescript 3.8**: linguaggio di programmazione che estende javascript aggiungendo la *tipizzazione_G* obbligatoria delle variabili;
 - **Web3.js**: libreria js che permette l'interazione tramite protocolli *HTTP_G* e *IPC_G* con un nodo *Ethereum_G* remoto;
- **Blockchain e ambienti di test**
 - **Ethereum**: *blockchain_G* globale e decentralizzata basata sulla criptovaluta *ETH_G*, utilizzata per la creazione di *DApp_G* basate su *blockchain_G*;
 - **Smart Contract**: protocollo informatico che verifica, esegue e fa rispettare un contratto, ovvero del codice;
 - **Ganache**: *blockchain_G* di test utilizzata per lo sviluppo dell'applicazione. Verrà qui testata pubblicamente l'applicazione in staging;
 - **Truffle**: framework nato per facilitare lo sviluppo di *smart contract_G* e *DApp_G*. Utilizzato inoltre come ambiente di test dell'applicativo in locale.

2.6 Vincoli Generali

- **Utente:**
 - Necessaria una **connessione ad internet**;
 - Avere installati nel sistema **Node.js** ed il gestore di pacchetti **NPM** (Solitamente presente nell'installazione di **Node.js**);
 - Il download dell'applicativo **etherless-cli** attraverso **NPM**;
- **Gestore della piattaforma:**
 - La creazione e la corretta configurazione di un account su **AWS Lambda**.



3 Casi d'uso

3.1 Attori dei casi d'uso

3.1.1 Attori primari

- **Utente autenticato:** un utente che è già autenticato alla piattaforma *Etherless_G* attraverso il proprio *wallet_G*;
- **Utente non autenticato:** un utente che non è ancora autenticato alla piattaforma *Etherless_G*, quindi dotato di utilizzo estremamente limitato;
- **Utente generico:** l'unione di utente autenticato e utente non autenticato;

3.1.2 Attori secondari

- **Ethereum:** piattaforma per applicazioni decentralizzate; dispone di una propria criptovaluta nativa chiamata Ether (*ETH_G*);
- **AWS Lambda:** ambiente *serverless_G* nel quale verranno eseguiti i prodotti software messi a disposizione dagli *utenti_G*;
- **Filesystem locale:** indica lo spazio di archiviazione locale (tipicamente la macchina dell'utente) su cui è installata *etherless-cli*;
- **Gestore della piattaforma:** la persona, fisica o giuridica, che si interfaccia con *AWS Lambda_G*. Il beneficiario dei pagamenti per l'esecuzione del codice sulla piattaforma.

3.2 Elenco dei casi d'uso

In questa sezione sono elencati tutti i casi d'uso individuati. Ogni caso d'uso rappresenta uno scenario per uno o più attori, ovviamente applicabile anche ad eventuali attori derivati. Ogni caso d'uso, inoltre, viene descritto tramite diagrammi dei casi d'uso e possiede una preconditione seguita da una postcondizione.

Prerequisito comune: tutti i casi d'uso descritti di seguito, hanno come preconditione comune quella di rispettare i vincoli generali della sezione 2.6.

3.2.1 UC1 - Visualizzazione guida dell'applicazione

- **Attori Primari:** utente generico;
- **Descrizione:** l'utente visualizza la guida introduttiva, contenente l'elenco dei comandi con la descrizione del relativo utilizzo;
- **Scenario principale:** l'utente esegue il comando *etherless* oppure *etherless help*;
- **Precondizione:**
 - l'utente ha eseguito il comando *etherless* oppure *etherless help*;
 - sono rispettati i vincoli generali indicati nella sezione 2.6
- **Postcondizione:** l'utente visualizza la guida introduttiva mediante interfaccia *etherless-cli*;

3.2.2 UC2 - Inizializzazione ed autenticazione

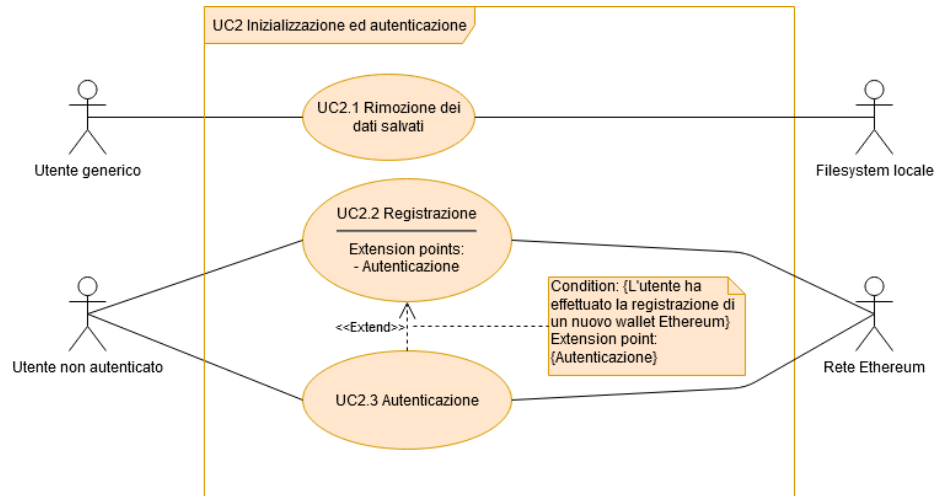


Figura 3.2.1: Diagramma UC2 - Inizializzazione ed autenticazione (init)

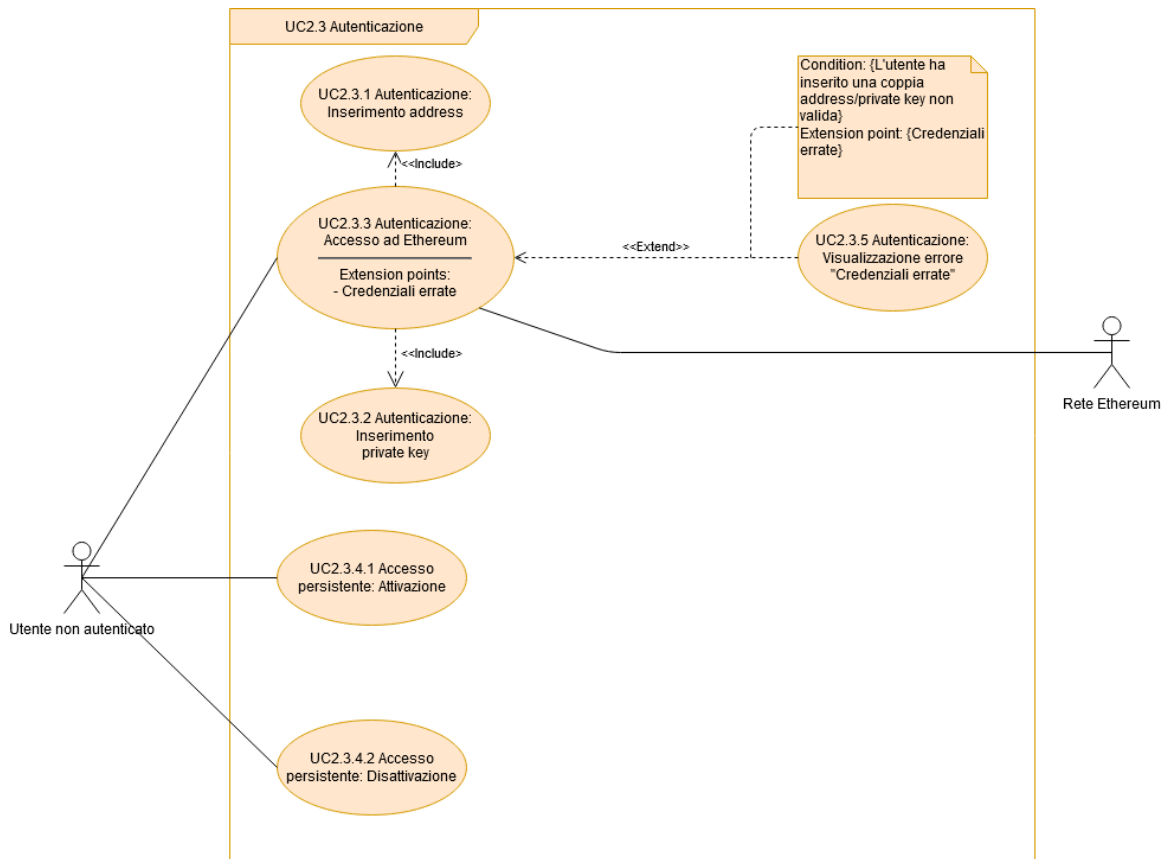


Figura 3.2.2: Diagramma UC2.3 - Autenticazione (init)



- **Attori Primari:** utente generico;
- **Attori Secondari:** *filesystem locale_G*;
- **Descrizione:** l'utente vuole inizializzare l'applicazione, necessitando di accedere ad *Etherless_G* attraverso un nuovo *wallet_G*;
- **Scenario principale:**
 - l'utente, inizializza l'applicazione tramite il comando **init**, collegando all'applicazione un *wallet_G Ethereum_G*;
 - il sistema rimuove eventuali dati salvati nel *filesystem locale_G* relativi ad altri *wallet_G*;
- **Precondizione:**
 - l'utente ha eseguito il comando **init**;
 - sono rispettati i vincoli generali indicati nella sezione 2.6;
- **Postcondizione:** viene collegato un *wallet_G Ethereum_G* all'applicazione e l'utente risulta autenticato.

3.2.2.1 UC2.1 - Rimozione dei dati salvati

- **Attori Primari:** utente generico;
- **Attori Secondari:** *filesystem locale_G*;
- **Descrizione:** l'utente avvia la procedura di inizializzazione ed autenticazione;
- **Scenario principale:** viene rimosso dal *filesystem locale_G* ogni riferimento, se presente, al *wallet_G* collegato in precedenza all'applicazione;
- **Precondizione:** sono rispettati i vincoli generali indicati nella sezione 2.6;
- **Postcondizione:** l'utente risulta non autenticato ed ogni riferimento a precedenti autenticazioni è stato rimosso;
- **Estensioni:**
 - **UC2.2 - Registrazione:** l'utente richiede un nuovo *wallet_G Ethereum_G*;
 - **UC2.3 - Autenticazione:** l'utente richiede l'utilizzo di un *wallet_G Ethereum_G* già esistente.



3.2.2.2 UC2.2 - Registrazione

- **Attori Primari:** utente non autenticato;
- **Attori Secondari:** rete Ethereum;
- **Descrizione:** l'utente avvia la procedura automatica di registrazione, e quindi di creazione di un nuovo *wallet_G Ethereum_G*;
- **Scenario principale:** l'utente, che non dispone di un *wallet_G* necessario all' utilizzo di *Etherless_G*, ne richiede la creazione;
- **Precondizione:**
 - l'utente, in seguito al comando `init`, ha richiesto la creazione di un nuovo account rispondendo **S** oppure **s** alla domanda *Vuoi creare un nuovo wallet Ethereum?* stampata a video nella *CLI_G*;
 - sono rispettati i vincoli generali indicati nella sezione 2.6;
- **Postcondizione:** il sistema richiede alla rete *Ethereum_G* la creazione di un nuovo *wallet_G*, e ne stampa a video le credenziali **address** e **private key**;
- **Estensioni:**
 - **UC2.3 - Autenticazione:** l'utente esegue l'autenticazione con il *wallet_G Ethereum_G* appena creato.

3.2.2.3 UC2.3 - Autenticazione

- **Attori Primari:** utente non autenticato;
- **Attori Secondari:** rete Ethereum;
- **Descrizione:** l'utente tenta l'autenticazione al sistema tramite le credenziali del proprio *wallet_G*
- **Scenario principale:** l'utente, già in possesso di un *wallet_G*, si vuole autenticare alla piattaforma *Etherless_G*;
- **Precondizione:**
 - l'utente, in seguito al comando `init`, non ha richiesto la creazione di un nuovo account rispondendo **N** oppure **n** alla domanda *Vuoi creare un nuovo wallet Ethereum?* stampata a video nella *CLI_G*, oppure ha già concluso la procedura di creazione di un account, attivata rispondendo **S** oppure **s** alla stessa domanda;
 - sono rispettati i vincoli generali indicati nella sezione 2.6;
- **Postcondizione:** il sistema richiede l'immissione in sequenza delle credenziali identificanti il *wallet_G* dell'utente, in particolare **address** e **private key**.



3.2.2.4 UC2.3.1 - Autenticazione: Inserimento address

- **Attori Primari:** utente non autenticato;
- **Descrizione:** l'utente, sta eseguendo l'autenticazione alla piattaforma *Etherless_G*, attraverso il proprio *wallet_G*;
- **Scenario principale:**
 - l'utente, in seguito al comando **init**, non ha richiesto la creazione di un nuovo account rispondendo **N** oppure **n** alla domanda *Vuoi creare un nuovo wallet Ethereum?* stampata a video nella *CLI_G*, oppure ha già concluso la procedura di creazione di un account, attivata rispondendo **S** oppure **s** alla stessa domanda;
 - la *CLI_G* richiede all'utente l'immissione del proprio **address**;
- **Precondizione:** l'utente, in seguito al comando **init**, non ha richiesto la creazione di un nuovo account rispondendo **N** oppure **n** alla domanda *Vuoi creare un nuovo wallet Ethereum?*;
- **Postcondizione:** l'utente ha inserito con successo l' **address** relativo al *wallet_G* che vuole utilizzare con *Etherless_G*.

3.2.2.5 UC2.3.2 - Autenticazione: Inserimento private key

- **Attori Primari:** utente non autenticato;
- **Descrizione:** l'utente sta eseguendo l'autenticazione alla piattaforma *Etherless_G*, attraverso il proprio *wallet_G*;
- **Scenario principale:**
 - l'utente, in seguito al comando **init**, non ha richiesto la creazione di un nuovo account rispondendo **N** oppure **n** alla domanda *Vuoi creare un nuovo wallet Ethereum?* stampata a video nella *CLI_G*, oppure ha già concluso la procedura di creazione di un account, attivata rispondendo **S** oppure **s** alla stessa domanda;
 - l'utente ha inserito con successo il proprio **address**;
 - la *CLI_G* richiede all'utente l'immissione della propria **private key**;
- **Precondizione:** l'utente ha inserito l' **address** richiesto nello step precedente;
- **Postcondizione:** dopo l'inserimento della **private key**, il sistema verifica la correttezza dei dati forniti.



3.2.2.6 UC2.3.3 - Autenticazione: Accesso ad Ethereum

- **Attori Primari:** utente non autenticato;
- **Attori secondari:** rete *Ethereum_G*;
- **Descrizione:** il sistema verifica le credenziali **address** e **private key** fornite dall'utente, accedendo alla rete *Ethereum_G*;
- **Scenario principale:**
 - l'utente, in seguito al comando **init**, non ha richiesto la creazione di un nuovo account rispondendo **N** oppure **n** alla domanda *Vuoi creare un nuovo wallet Ethereum?* stampata a video nella *CLI_G*;
 - l'utente ha inserito con successo il proprio **address** e la propria **private key**;
 - il sistema verifica la correttezza dei dati;
 - sono rispettati i vincoli generali indicati nella sezione 2.6;
- **Precondizione:** l'utente ha inserito con successo i dati necessari all'autenticazione ad *Etherless_G*;
- **Postcondizione:** il sistema verifica le credenziali fornite dall'utente, dando il benvenuto all'utente caratterizzato da **address** nel caso il processo vada a buon fine, o riportando un messaggio di credenziali errate nel caso ci fossero problemi durante l'accesso ad *Ethereum_G*;
- **Estensioni:**
 - **UC2.3.5 - Autenticazione: Visualizzazione errore "Credenziali errate":** l'utente ha inserito una coppia di **private key** e **address** non valida;
- **Inclusioni:**
 - **UC2.3.1 - Autenticazione: Inserimento address:** l'utente inserisce l'**address** di un *wallet_G Ethereum_G*;
 - **UC2.3.2 - Autenticazione: Inserimento private key:** l'utente inserisce la **private key** di un *wallet_G Ethereum_G*.

3.2.2.7 UC2.3.4 - Autenticazione: Accesso persistente

- **Attori Primari:** utente autenticato;
- **Descrizione:** il sistema propone all'utente la possibilità di salvare i dati con cui ha appena effettuato l'accesso, consentendo di attivare un'autenticazione persistente, ovvero automatica;
- **Scenario principale:**
 - l'utente ha effettuato correttamente l'accesso ad *Etherless_G*;
 - il sistema propone all'utente di attivare la procedura di autenticazione persistente;
- **Precondizione:** l'utente ha appena effettuato con successo l'accesso ad *Etherless_G* tramite il comando **init**
- **Postcondizione:** Il sistema richiede all'utente tramite *CLI_G* se vuole attivare la procedura di autenticazione automatica, attendendo una risposta da parte di quest'ultimo.



3.2.2.8 UC2.3.4.1 - Accesso persistente: Attivazione

- **Attori Primari:** utente autenticato;
- **Attori secondari:** *filesystem locale_G*;
- **Descrizione:** l'utente attiva il processo di autenticazione persistente (e quindi automatica);
- **Scenario principale:**
 - l'utente ha effettuato correttamente l'accesso ad *Etherless_G*;
 - in seguito alla richiesta da parte della *CLI_G* di attivazione dell'autenticazione persistente, l'utente ha risposto **s** oppure **S**;
 - l'autenticazione automatica viene attivata, l'utente accede ad *Etherless_G* visualizzando la guida introduttiva;
- **Precondizione:** l'utente, in seguito all'autenticazione, ha risposto **s** oppure **S** alla richiesta di attivazione dell'accesso automatico;
- **Postcondizione:** Il sistema attiva l'autenticazione persistente, scrivendo sul *filesystem locale_G* le credenziali di accesso che utilizzerà fino al prossimo **init**. L'utente accede ad *Etherless_G* visualizzando la guida introduttiva.

3.2.2.9 UC2.3.4.2 - Accesso persistente: Disattivazione

- **Attori Primari:** utente autenticato;
- **Descrizione:** l'utente ha scelto di non attivare la procedura di autenticazione persistente, scegliendo di autenticarsi ad ogni nuova sessione di *Etherless_G*;
- **Scenario principale:**
 - l'utente ha effettuato correttamente l'accesso ad *Etherless_G*;
 - in seguito alla richiesta da parte della *CLI_G* di attivazione dell'autenticazione persistente, l'utente ha risposto **n** oppure **N**;
 - l'autenticazione persistente non viene attivata, l'utente accede ad *Etherless_G* visualizzando la guida introduttiva;
- **Precondizione:** l'utente, in seguito all'autenticazione, ha risposto **n** oppure **N** alla richiesta di attivazione dell'accesso persistente;
- **Postcondizione:** il sistema non attiva l'autenticazione automatica, l'utente accede alla piattaforma visualizzando la guida introduttiva.

3.2.3 UC2.3.5 - Autenticazione: Visualizzazione errore “Credenziali errate”

- **Attori Primari:** utente non autenticato;
- **Descrizione:** all'applicazione informa l'utente che ha inserito delle credenziali non valide per l'accesso alla rete *Ethereum_G*;
- **Scenario principale:** l'utente tenta di accedere alla rete *Ethereum_G* con delle credenziali non valide;
- **Precondizione:** l'utente ha inserito delle credenziali non valide durante la procedura di autenticazione;
- **Postcondizione:** visualizza un messaggio d'errore riguardante la non validità dei dati specificati dall'utente.

3.2.4 UC3 - Verifica wallet in uso dall'applicazione

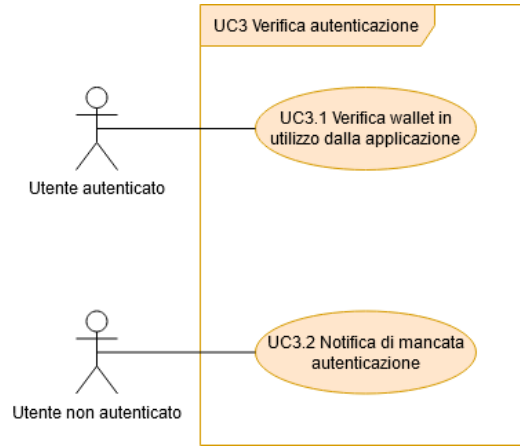


Figura 3.2.3: Diagramma UC3 - Verifica wallet in uso (who)

- **Attori Primari:** utente generico;
- **Descrizione:** viene informato l'utente del suo stato all'interno dell'applicazione, in particolare del *wallet_G* al quale è stato fatto accesso o la mancata autenticazione;
- **Scenario principale:** l'utente, autenticato o meno, esegue il comando **who**, che permette la visualizzazione del proprio *address Ethereum_G* qualora sia possibile;
- **Precondizione:**
 - l'utente ha eseguito il comando **who** nella *CLI_G*;
 - sono rispettati i vincoli generali indicati nella sezione 2.6;
- **Postcondizione:** il sistema mostra all'utente tramite *CLI_G*, l'*address* che caratterizza il *wallet_G* con cui esso è autenticato in *Etherless_G*, altrimenti stampa il messaggio indicante la mancata autenticazione da parte di esso.

3.2.4.1 UC3.1 - Verifica *wallet_G* in utilizzo dall'applicazione

- **Attori Primari:** utente autenticato;
- **Descrizione:** l'applicazione stampa a video la credenziale *address* caratterizzante il *wallet_G* che *Etherless_G* sta attualmente utilizzando;
- **Scenario principale:** l'utente autenticato esegue il comando **who**, che permette la visualizzazione del proprio *address Ethereum_G*;
- **Precondizione:**
 - l'utente ha eseguito il comando **who** nella *CLI_G*;
 - sono rispettati i vincoli generali indicati nella sezione 2.6;
- **Postcondizione:** il sistema mostra all'utente tramite *CLI_G*, l'*address* che caratterizza il *wallet_G* con cui esso è autenticato in *Etherless_G*.



3.2.4.2 UC3.2 - Notifica di mancata autenticazione

- **Attori Primari:** utente non autenticato;
- **Descrizione:** l'applicazione stampa il messaggio indicante la mancata autenticazione da parte dell'utente (in rif. a UC2);
- **Scenario principale:** l'utente non autenticato esegue il comando `who` e viene avvisato della mancata autenticazione;
- **Precondizione:**
 - l'utente ha eseguito il comando `who` nella *CLIG*;
 - sono rispettati i vincoli generali indicati nella sezione 2.6;
- **Postcondizione:** il sistema avverte l'utente della mancata autenticazione da parte di esso tramite apposito messaggio.

3.2.5 UC4 - Elenco dei prodotti software disponibili in *Etherless_G*

- **Attori Primari:** utente autenticato;
- **Descrizione:** l'applicazione elenca i prodotti software disponibili nella piattaforma;
- **Scenario principale:** l'utente richiede la lista di prodotti software pubblicati sulla piattaforma attraverso il comando `list`;
- **Precondizione:**
 - l'utente ha eseguito il comando `list` nella *CLIG*;
 - sono rispettati i vincoli generali indicati nella sezione 2.6;
- **Postcondizione:** l'applicazione fornisce all'utente la lista dei prodotti software disponibili.

3.2.6 UC5 - Verifica disponibilità di un prodotto software

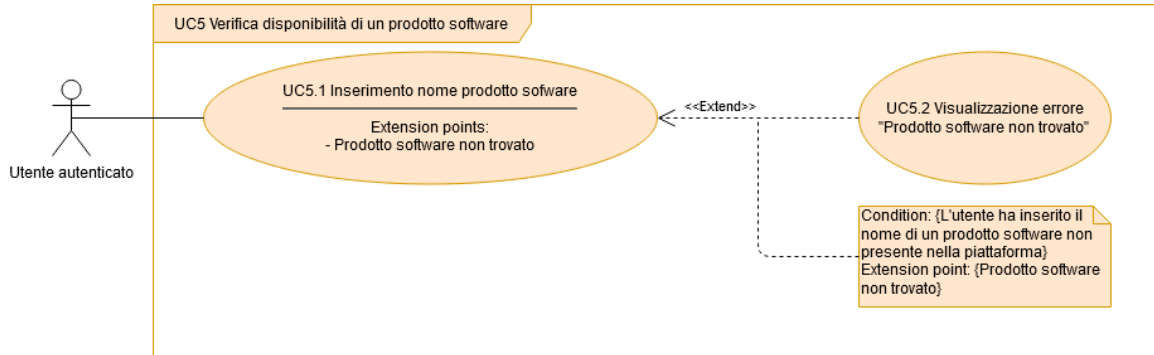


Figura 3.2.4: Diagramma UC5 - Verifica disponibilità di un prodotto software (**find**)

- **Attori Primari:** utente autenticato;
- **Descrizione:** l'applicazione conferma o smentisce la disponibilità di un prodotto software dal nome specificato;
- **Scenario principale:**
 - l'utente esegue il comando **find** per controllare la presenza nella piattaforma di prodotto software identificato dal nome specificato;
 - la *CLIG* visualizza l'esito dell'operazione di pubblicazione;
- **Precondizione:**
 - l'utente ha eseguito il comando **find** nella *CLIG*;
 - sono rispettati i vincoli generali indicati nella sezione 2.6;
- **Postcondizione:** l'applicazione conferma la disponibilità del prodotto software dal nome specificato e stampa alcune informazioni su di esso;

3.2.6.1 UC5.1 - Inserimento nome prodotto software

- **Attori Primari:** utente autenticato;
- **Descrizione:** l'utente può inserire dalla *CLIG* il nome del prodotto software da trovare;
- **Scenario principale:** l'utente inserisce successivamente al comando **find** il nome del prodotto software da trovare;
- **Precondizione:** l'utente ha inserito tramite *CLIG* il comando **find**;
- **Postcondizione:** l'utente ha inserito successivamente al comando **find** il nome del prodotto software da trovare;
- **Estensioni:**
 - **UC5.2:** la piattaforma non dispone di un prodotto software identificato tramite il nome specificato.

3.2.6.2 UC5.2 - Visualizzazione errore “Prodotto software non trovato”

- **Attori Primari:** utente autenticato;
- **Descrizione:** l'applicazione informa l'utente della mancata disponibilità di un prodotto software dal nome specificato;
- **Scenario principale:** l'utente controlla l'esistenza di prodotto software recante il nome specificato e tale prodotto non è in quel momento disponibile;
- **Precondizione:** l'utente ha inserito successivamente al comando **find** il nome del prodotto software da trovare;
- **Postcondizione:** visualizza un messaggio d'errore riguardante la mancata disponibilità del prodotto software specificato dall'utente.

3.2.7 UC6 - Richiesta di pubblicazione del proprio prodotto software

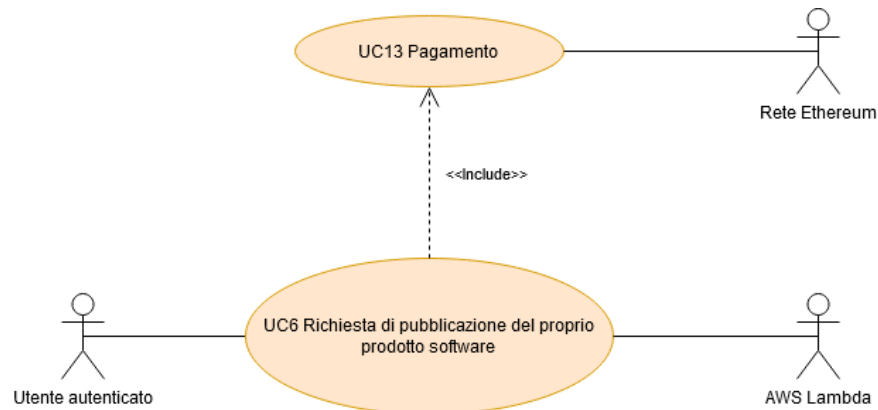


Figura 3.2.5: Diagramma UC6 - Richiesta di pubblicazione del proprio prodotto software (deploy)

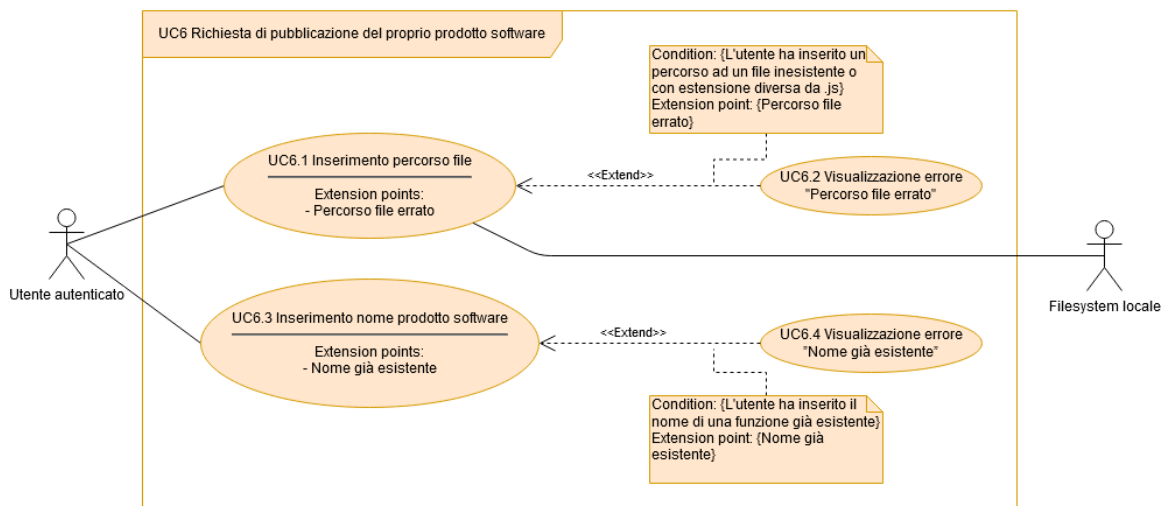


Figura 3.2.6: Diagramma UC6 - Richiesta di pubblicazione del proprio prodotto software (deploy)



- **Attori Primari:** utente autenticato;
- **Attori Secondari:** *AWS Lambda_G*;
- **Descrizione:** l'utente richiede il *deploy_G* del proprio prodotto software sulla piattaforma per renderlo eseguibile da tutti gli utenti;
- **Scenario principale:**
 - l'utente esegue il comando **deploy** per pubblicare il proprio prodotto software su *Etherless_G*;
 - la *CLI_G* visualizza l'esito dell'operazione di pubblicazione;
- **Precondizione:**
 - l'utente possiede un prodotto software in locale;
 - sono rispettati i vincoli generali indicati nella sezione 2.6;
- **Postcondizione:** il sistema pubblica il prodotto software su *Etherless_G*, successivamente eseguibile con il nome specificato dall'utente; L'applicazione informa l'utente dell'esito dell'operazione;
- **Inclusioni:**
 - **UC13:** la pubblicazione di un prodotto software comporta un pagamento ad *Ethereum_G*.

3.2.7.1 UC6.1 - Inserimento percorso file

- **Attori Primari:** utente autenticato;
- **Attori Secondari:** *filesystem locale_G*;
- **Descrizione:** l'utente può inserire dalla *CLI_G* il percorso del file nel *filesystem locale_G* contenente il prodotto software da pubblicare;
- **Scenario principale:** l'utente inserisce il percorso del file, all'interno del *filesystem locale_G*, contenente il prodotto software;
- **Precondizione:** l'utente ha inserito tramite *CLI_G* il comando **deploy**;
- **Postcondizione:** l'utente ha inserito successivamente al comando **deploy** il percorso del file contenente il prodotto software da pubblicare;
- **Estensioni:**
 - **UC6.2:** l'utente ha inserito un percorso errato o che non corrisponde ad un file con estensione **.js**.



3.2.7.2 UC6.2 - Visualizzazione errore “Percorso file errato”

- **Attori Primari:** utente autenticato;
- **Descrizione:** la *CLI_G* restituisce un messaggio di errore dovuto all'invio di un comando **deploy** contenente un percorso di un file errato, in quanto il file non esiste oppure non ha come estensione `.js`;
- **Scenario principale:** la *CLI_G* segnala all'utente l'errore con un messaggio **Percorso del file errato**;
- **Precondizione:** l'utente ha inviato il comando **deploy** seguito da un percorso del file errato;
- **Postcondizione:** la *CLI_G* segnala l'errore dell'utente nell'inserimento del percorso del file con un messaggio.

3.2.7.3 UC6.3 - Inserimento nome prodotto software

- **Attori Primari:** utente autenticato;
- **Descrizione:** l'utente può inserire dalla *CLI_G* il nome che verrà assegnato al proprio prodotto software una volta pubblicato su *Etherless_G* per poter essere successivamente eseguito;
- **Scenario principale:** l'utente inserisce il nome del prodotto software che desidera pubblicare su *Etherless_G*;
- **Precondizione:** l'utente ha inserito tramite *CLI_G* il comando **deploy** seguito dal percorso del file contenente il prodotto software;
- **Postcondizione:** l'utente ha inserito successivamente al comando **deploy** e il percorso del file il nome del prodotto software che desidera pubblicare su *Etherless_G*;
- **Estensioni:**
 - UC6.4: l'utente ha inserito il nome di un prodotto software già esistente su *Etherless_G*.

3.2.7.4 UC6.4 - Visualizzazione errore “Nome già esistente”

- **Attori Primari:** utente autenticato;
- **Descrizione:** la *CLI_G* restituisce un messaggio di errore dovuto all'invio di un comando **deploy** contenente un nome già esistente in *Etherless_G*;
- **Scenario principale:** la *CLI_G* segnala all'utente l'errore con un messaggio **Nome già esistente**;
- **Precondizione:** l'utente ha inviato il comando **deploy** seguito dal percorso del file contenente il prodotto software e il nome da assegnare su *Etherless_G*;
- **Postcondizione:** la *CLI_G* segnala l'errore dell'utente nell'inserimento di un nome per il proprio prodotto software già esistente su *Etherless_G*.

3.2.8 UC7 - Richiesta di esecuzione di un prodotto software

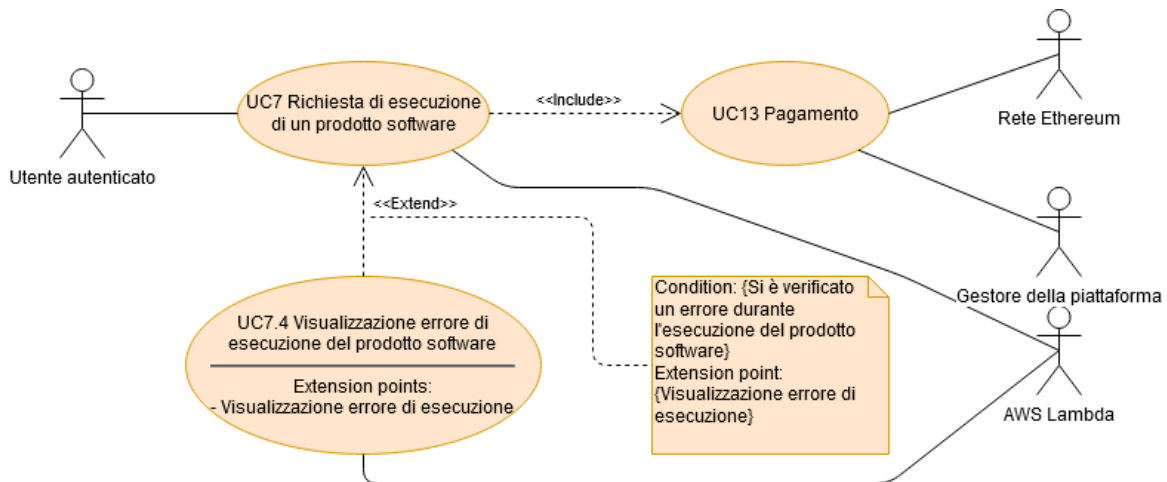


Figura 3.2.7: Diagramma UC7 - Richiesta di esecuzione di un prodotto software (run)

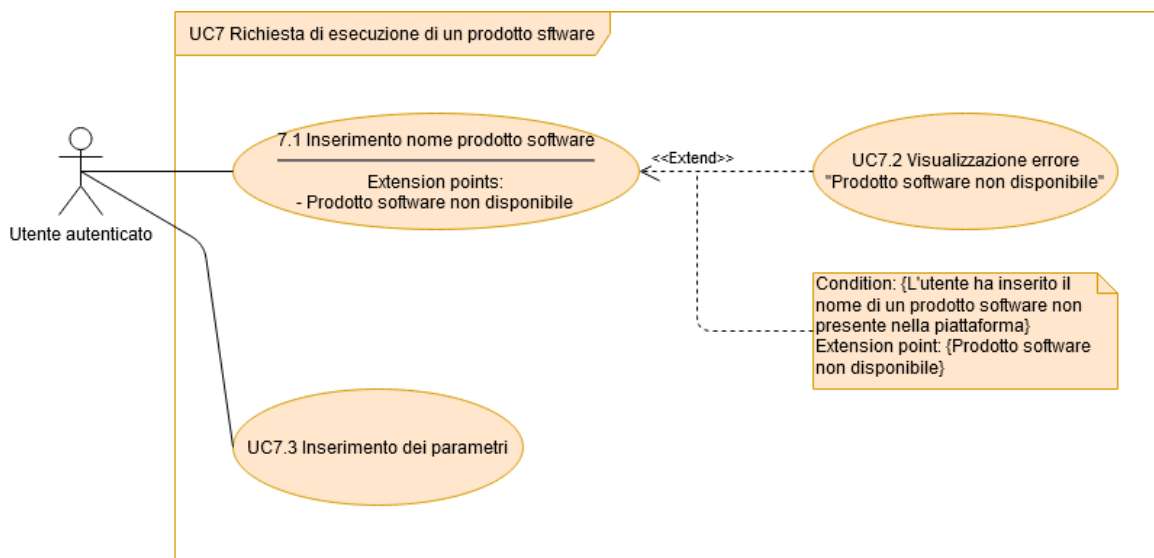


Figura 3.2.8: Diagramma UC7 - (Dettaglio) Richiesta di esecuzione di un prodotto software (run)

- **Attori Primari:** utente autenticato;
- **Attori Secondari:**
 - gestore della piattaforma_G;
 - AWS Lambda_G;
- **Descrizione:** l'utente richiede l'esecuzione di un prodotto software sulla piattaforma;



- **Scenario principale:**

- l'utente esegue il comando **run** per eseguire in *AWS Lambda_G* un prodotto software pubblicato nella piattaforma con il nome specificato al fine di elaborare i dati passati, pagando l'esecuzione al *gestore della piattaforma_G*;
- La *CLI_G* visualizza l'esito dell'operazione di esecuzione;

- **Precondizione:**

- l'utente ha la necessità di eseguire un prodotto software pubblicato sulla piattaforma;
- sono rispettati i vincoli generali indicati nella sezione 2.6;

- **Postcondizione:** l'applicazione informa l'utente dell'esito dell'operazione, ed in caso di successo nell'esecuzione del prodotto software richiesto il risultato di questa; in caso di insuccesso l'errore dal quale è scaturito;

- **Estensioni:**

- **UC7.4:** l'esecuzione del prodotto software incontra un errore;

- **Inclusioni:**

- **UC13:** l'esecuzione di un prodotto software comporta un pagamento ad *Ethereum_G* ed *AWS Lambda_G*.

3.2.8.1 UC7.1 - Inserimento nome prodotto software

- **Attori Primari:** utente autenticato;

- **Descrizione:** l'utente può inserire dalla *CLI_G* il nome del prodotto software del quale intende pagare l'esecuzione;

- **Scenario principale:** l'utente inserisce il nome del prodotto software che desidera eseguire su *Etherless_G*;

- **Precondizione:** l'utente ha inserito tramite *CLI_G* il comando **run**;

- **Postcondizione:** l'utente ha inserito il comando **run** il nome del prodotto software che desidera eseguire su *Etherless_G*;

- **Estensioni:**

- **UC7.2:** l'utente ha inserito il nome di un prodotto software non disponibile su *Etherless_G*.

3.2.8.2 UC7.2 - Visualizzazione errore “Prodotto software non disponibile”

- **Attori Primari:** utente autenticato;

- **Descrizione:** la *CLI_G* restituisce un messaggio di errore dovuto all'invio di un comando **run** contenente un nome non esistente in *Etherless_G*;

- **Scenario principale:** la *CLI_G* segnala all'utente l'errore con un messaggio Nome non esistente;

- **Precondizione:** l'utente ha inviato il comando **run** seguito dal nome del prodotto software su *Etherless_G*;

- **Postcondizione:** la *CLI_G* segnala l'indisponibilità di un prodotto software con il nome specificato nella piattaforma *Etherless_G*.



3.2.8.3 UC7.3 - Inserimento dei parametri

- **Attori Primari:** utente autenticato;
- **Descrizione:** l'utente può inserire dalla *CLI_G* uno o più parametri validi nell'esecuzione del prodotto software del quale intende pagare l'esecuzione;
- **Scenario principale:** l'utente inserisce una lista di parametri da passare all'esecuzione del prodotto software;
- **Precondizione:** l'utente ha inserito tramite *CLI_G* il comando **run** seguito dal nome del prodotto software da eseguire;
- **Postcondizione:** l'utente ha inserito successivamente al comando **run** ed al nome del prodotto software che desidera eseguire su *Etherless_G* anche la lista di parametri.

3.2.8.4 UC7.4 - Visualizzazione errore di esecuzione del prodotto software

- **Attori Primari:** utente autenticato;
- **Attori Secondari:**
 - *AWS Lambda_G*
- **Descrizione:** si è riscontrato un errore nell'esecuzione del prodotto software richiesto dall'utente in *AWS Lambda_G*;
- **Scenario principale:** l'esecuzione di un prodotto software indicato da un utente si arresta in modo anomalo senza produrre i risultati attesi;
- **Precondizione:** l'esecuzione del prodotto software è stata pagata dall'utente ed iniziata da *AWS Lambda_G*;
- **Postcondizione:** la *CLI_G* informa l'utente della ragione dell'arresto anomalo dell'esecuzione del prodotto software indicato dall'utente.

3.2.9 UC8 - Richiesta elenco delle operazioni richieste alla piattaforma

- **Attori Primari:** utente autenticato;
- **Descrizione:** l'utente richiede la lista di operazioni richieste alla piattaforma e pagate attraverso il *wallet_G* collegato all'applicazione;
- **Scenario principale:**
 - l'utente esegue il comando **history** per ottenere la lista dei prodotti software eseguiti;
 - la *CLI_G* visualizza la lista di esecuzioni finanziate;
- **Precondizione:**
 - l'utente ha la necessità di verificare le esecuzioni finanziate sulla piattaforma, o è interessato all'identificativo univoco di una di queste, anche in rif. a UC9;
 - sono rispettati i vincoli generali indicati nella sezione 2.6;
- **Postcondizione:** l'applicazione elenca all'utente le esecuzioni di prodotti software richieste in precedenza e pagate (in rif. ad UC13), ogni richiesta di esecuzione evidenziata viene correlata da un identificativo univoco.

3.2.10 UC9 - Richiesta del risultato di una operazione

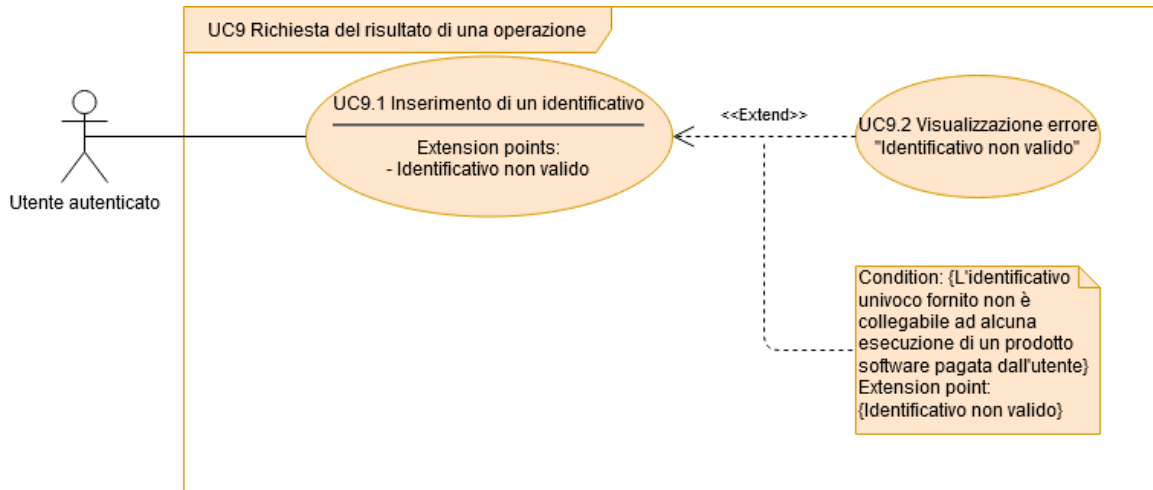


Figura 3.2.9: Diagramma UC9 - Richiesta del risultato di un operazione (log)

- **Attori Primari:** utente autenticato;
- **Descrizione:** l'utente richiede l'esito di una operazione precedentemente richiesta alla piattaforma (in riferimento ad UC7) attraverso un identificativo univoco di tale operazione (in riferimento ad UC8);
- **Scenario principale:**
 - l'utente esegue il comando `log` per ottenere la lista dei prodotti software eseguiti;
 - la `CLIG` visualizza lo stato ed eventualmente il risultato dell'esecuzione selezionata;
- **Precondizione:**
 - l'utente ha a disposizione l'identificativo univoco di una operazione precedentemente eseguita;
 - sono rispettati i vincoli generali indicati nella sezione 2.6;
- **Postcondizione:** l'applicazione segnala all'utente dello stato di tale operazione ed in caso l'esecuzione sia terminata, informa l'utente dell'esito di tale esecuzione.



3.2.10.1 UC9.1 - Inserimento di un identificativo

- **Attori Primari:** utente autenticato;
- **Descrizione:** l'utente può inserire dalla *CLI_G* il nome del prodotto software del quale intende pagare l'esecuzione;
- **Scenario principale:** l'utente inserisce il nome del prodotto software che desidera eseguire su *Etherless_G*;
- **Precondizione:** l'utente ha inserito tramite *CLI_G* il comando **run** seguito dal percorso del file contenente il prodotto software;
- **Postcondizione:** l'utente ha inserito successivamente al comando **run** il nome del prodotto software che desidera eseguire su *Etherless_G*;
- **Estensioni:**
 - **UC9.2:** l'utente ha inserito il nome di un prodotto software esistente su *Etherless_G*.

3.2.10.2 UC9.2 - Visualizzazione errore “Identificativo non valido”

- **Attori Primari:** utente autenticato;
- **Descrizione:** la *CLI_G* restituisce un messaggio di errore dovuto all'invalidità dell'identificativo univoco fornito al comando **log**;
- **Scenario principale:** la *CLI_G* segnala all'utente l'errore con un messaggio **identificativo non valido**;
- **Precondizione:** l'utente ha inviato il comando **log** seguito da un identificativo univoco di esecuzione di un prodotto software su *Etherless_G*;
- **Postcondizione:** la *CLI_G* segnala l'invalidità dell'identificativo univoco specificato dall'utente.

3.2.11 UC10 - Modifica del proprio prodotto software

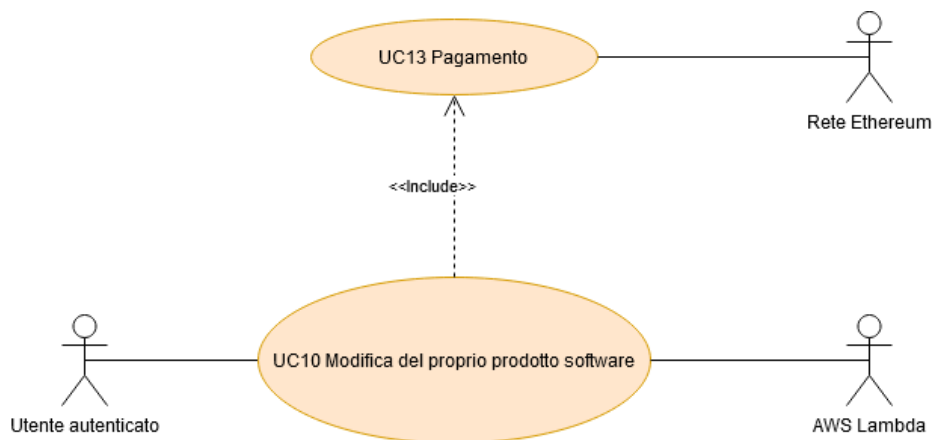


Figura 3.2.10: Diagramma UC10 - Modifica del proprio prodotto software (edit)

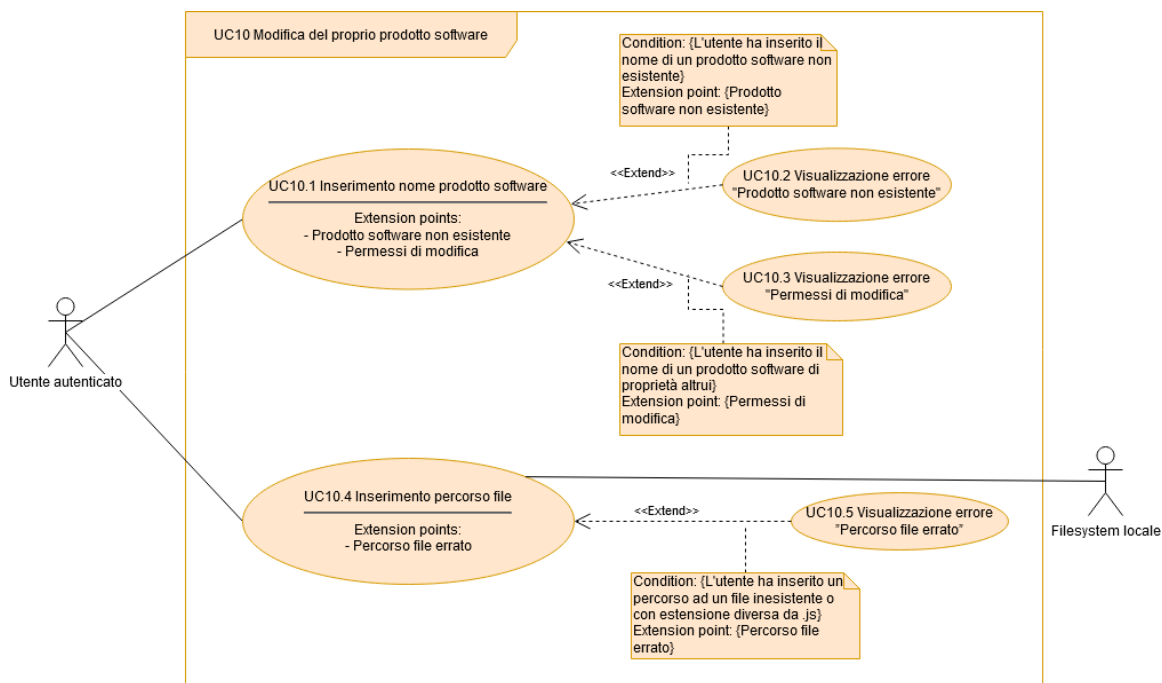


Figura 3.2.11: Diagramma UC10 - (Dettaglio) Modifica del proprio prodotto software (edit)



- **Attori Primari:** utente autenticato;
- **Attori Secondari:** *AWS Lambda_G*;
- **Descrizione:** l'utente richiede la modifica del codice di un determinato prodotto software già pubblicato sulla piattaforma *Etherless_G*;
- **Scenario principale:**
 - l'utente esegue il comando **edit** per modificare il codice del proprio prodotto software su *Etherless_G*;
 - la *CLI_G* visualizza l'esito dell'operazione di modifica;
- **Precondizione:**
 - l'utente dispone di una nuova versione del proprio prodotto software in locale e desidera aggiornarlo sulla piattaforma *Etherless_G*;
 - sono rispettati i vincoli generali indicati nella sezione 2.6;
- **Postcondizione:** l'applicazione pubblica la versione aggiornata del prodotto software su *Etherless_G*. L'applicazione informa l'utente dell'esito dell'operazione;
- **Inclusioni:**
 - **UC13:** la modifica di un prodotto software comporta un pagamento ad *Ethereum_G*.

3.2.11.1 UC10.1 - Inserimento nome prodotto software

- **Attori Primari:** utente autenticato;
- **Descrizione:** l'utente può inserire dalla *CLI_G* il nome del prodotto software da aggiornare, già pubblicato su *Etherless_G*;
- **Scenario principale:** l'utente inserisce il nome del prodotto software che desidera aggiornare su *Etherless_G*;
- **Precondizione:** l'utente ha inserito tramite *CLI_G* il comando **edit**;
- **Postcondizione:** l'utente ha inserito successivamente al comando **edit** il nome del prodotto software che desidera aggiornare su *Etherless_G*;
- **Estensioni:**
 - **UC10.2:** l'utente ha inserito il nome di un prodotto software inesistente su *Etherless_G*;
 - **UC10.3:** l'utente ha inserito il nome di un prodotto software esistente su *Etherless_G* che non è di sua proprietà.



3.2.11.2 UC10.2 - Visualizzazione errore “Prodotto software non esistente”

- **Attori Primari:** utente autenticato;
- **Descrizione:** la *CLIG* restituisce un messaggio di errore dovuto all'invio di un comando `edit` contenente il nome di un prodotto software inesistente su *Etherless_G*;
- **Scenario principale:** la *CLIG* segnala all'utente l'errore con un messaggio Nessun prodotto software corrispondente al nome specificato;
- **Precondizione:** l'utente ha inviato il comando `edit` seguito dal nome del prodotto software da aggiornare su *Etherless_G*;
- **Postcondizione:** la *CLIG* segnala l'errore dell'utente nell'inserimento di un nome per il prodotto software non esistente su *Etherless_G*.

3.2.11.3 UC10.3 - Visualizzazione errore “Permessi di modifica”

- **Attori Primari:** utente autenticato;
- **Descrizione:** la *CLIG* restituisce un messaggio di errore dovuto all'invio di un comando `edit` contenente il nome di un prodotto software su *Etherless_G* di proprietà altrui;
- **Scenario principale:** la *CLIG* segnala all'utente l'errore con un messaggio Non si dispone dei permessi necessari alla modifica del prodotto software specificato;
- **Precondizione:** l'utente ha inviato il comando `edit` seguito dal nome del prodotto software da aggiornare su *Etherless_G*;
- **Postcondizione:** la *CLIG* segnala l'errore dell'utente nell'inserimento del nome di un prodotto software su *Etherless_G* di proprietà altrui.

3.2.11.4 UC10.4 - Inserimento percorso file

- **Attori Primari:** utente autenticato;
- **Attori Secondari:** *filesystem locale_G*;
- **Descrizione:** l'utente può inserire dalla *CLIG* il percorso del file nel *filesystem locale_G* contenente il prodotto software da aggiornare;
- **Scenario principale:** l'utente inserisce il percorso del file, all'interno del *filesystem locale_G*, contenente il prodotto software;
- **Precondizione:** l'utente ha inserito tramite *CLIG* il comando `edit` seguito dal nome del prodotto software su *Etherless_G* da aggiornare;
- **Postcondizione:** l'utente ha inserito successivamente al comando `edit` e il nome del prodotto software il percorso del file contenente il prodotto software da aggiornare;
- **Estensioni:**
 - **UC10.5:** l'utente ha inserito un percorso errato o che non corrisponde ad un file con estensione `.js`.

3.2.11.5 UC10.5 - Visualizzazione errore “Percorso file errato”

- **Attori Primari:** utente autenticato;
- **Descrizione:** la *CLIG* restituisce un messaggio di errore dovuto all'invio di un comando `edit` contenente un percorso di un file errato, in quanto il file deve avere come estensione `.js`;
- **Scenario principale:** la *CLIG* segnala all'utente l'errore con un messaggio **Percorso del file errato**;
- **Precondizione:** l'utente ha inviato il comando `edit` seguito dal nome del prodotto software da aggiornare su *EtherlessG* e il percorso del file contenente il prodotto software;
- **Postcondizione:** la *CLIG* segnala l'errore dell'utente nell'inserimento del percorso del file con un messaggio.

3.2.12 UC11 - Eliminazione del proprio prodotto software

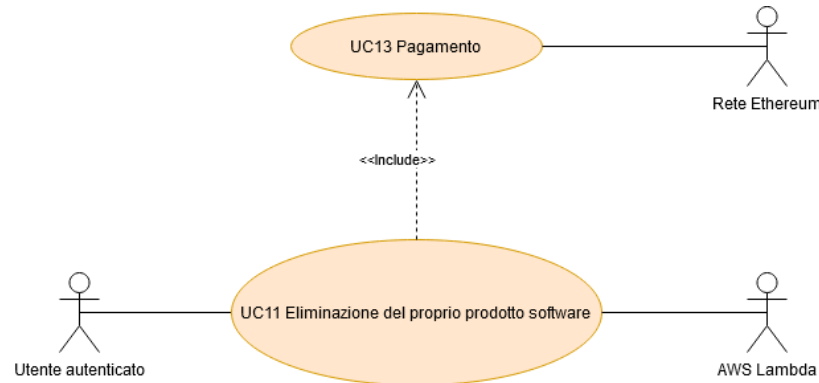


Figura 3.2.12: Diagramma UC11 - Eliminazione del proprio prodotto software (delete)

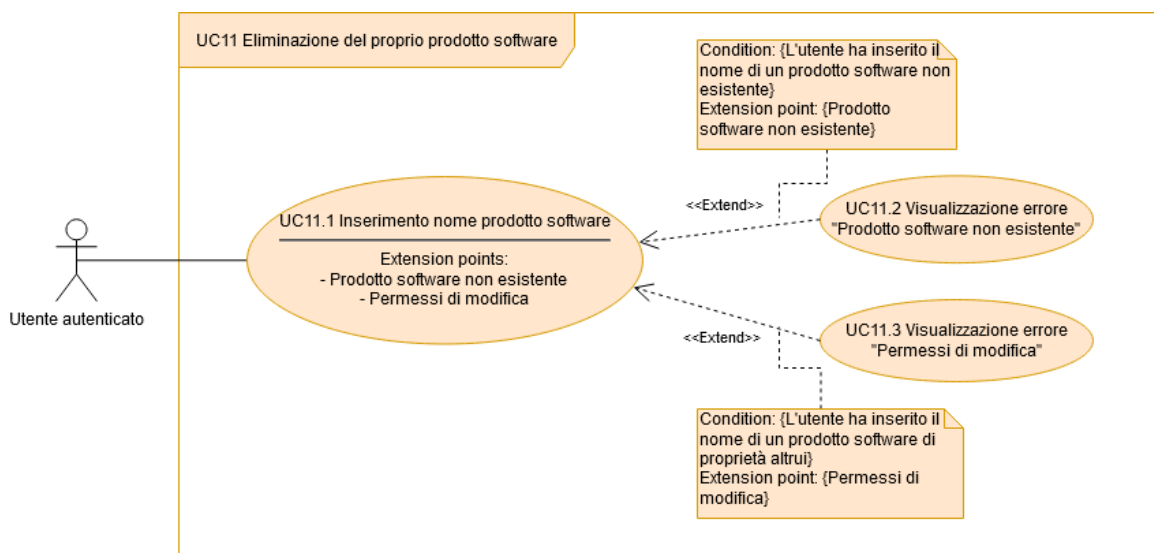


Figura 3.2.13: Diagramma UC11 - (Dettaglio) Eliminazione del proprio prodotto software (edit)



- **Attori Primari:** utente autenticato;
- **Attori Secondari:** *AWS Lambda_G*;
- **Descrizione:** l'utente richiede l'eliminazione di un proprio prodotto software pubblicato sulla piattaforma *Etherless_G*;
- **Scenario principale:**
 - l'utente esegue il comando **delete** per eliminare il prodotto software su *Etherless_G*;
 - la *CLI_G* visualizza l'esito dell'operazione di eliminazione;
- **Precondizione:**
 - l'utente desidera eliminare uno dei suoi prodotti software pubblicati sulla piattaforma *Etherless_G*;
 - sono rispettati i vincoli generali indicati nella sezione 2.6;
- **Postcondizione:** il prodotto software che l'utente desidera eliminare viene rimosso da *Etherless_G*. L'applicazione informa l'utente dell'esito dell'operazione;
- **Inclusioni:**
 - **UC13:** l'eliminazione di un prodotto software comporta un pagamento ad *Ethereum_G*.

3.2.12.1 UC11.1 - Inserimento nome prodotto software

- **Attori Primari:** utente autenticato;
- **Descrizione:** l'utente può inserire dalla *CLI_G* il nome del prodotto software presente su *Etherless_G* da eliminare;
- **Scenario principale:** l'utente inserisce il nome del prodotto software che desidera eliminare da *Etherless_G*;
- **Precondizione:** l'utente ha inserito tramite *CLI_G* il comando **delete**;
- **Postcondizione:** l'utente ha inserito successivamente al comando **delete** il nome del prodotto software che desidera eliminare da *Etherless_G*;
- **Estensioni:**
 - **UC11.2:** l'utente ha inserito il nome di un prodotto software inesistente su *Etherless_G*;
 - **UC11.3:** l'utente ha inserito il nome di un prodotto software esistente su *Etherless_G* che non è di sua proprietà.

3.2.12.2 UC11.2 - Visualizzazione errore “Prodotto software non esistente”

- **Attori Primari:** utente autenticato;
- **Descrizione:** la *CLI_G* restituisce un messaggio di errore dovuto all'invio di un comando **delete** contenente il nome di un prodotto software inesistente su *Etherless_G*;
- **Scenario principale:** la *CLI_G* segnala all'utente l'errore con un messaggio **Nessun prodotto software corrispondente al nome specificato**;
- **Precondizione:** l'utente ha inviato il comando **delete** seguito dal nome del prodotto software da eliminare da *Etherless_G*;
- **Postcondizione:** la *CLI_G* segnala l'errore dell'utente nell'inserimento di un nome per il prodotto software non esistente su *Etherless_G*.

3.2.12.3 UC11.3 - Visualizzazione errore "Permessi di modifica"

- **Attori Primari:** utente autenticato;
- **Descrizione:** la *CLI_G* restituisce un messaggio di errore dovuto all'invio di un comando `delete` contenente il nome di un prodotto software su *Etherless_G* di proprietà altrui;
- **Scenario principale:** la *CLI_G* segnala all'utente l'errore con un messaggio `Non si dispone dei permessi necessari all'eliminazione del prodotto software specificato`;
- **Precondizione:** l'utente ha inviato il comando `delete` seguito dal nome del prodotto software da eliminare da *Etherless_G*;
- **Postcondizione:** la *CLI_G* segnala l'errore dell'utente nell'inserimento del nome di un prodotto software su *Etherless_G* di proprietà altrui.

3.2.13 UC12 - Visualizzazione errore "Utente non autenticato"

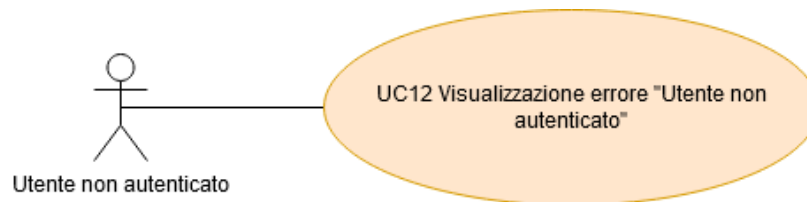


Figura 3.2.14: Diagramma UC12 - Visualizzazione errore: utente non autenticato (edit)

- **Attori Primari:** utente non autenticato;
- **Descrizione:** la *CLI_G* restituisce un messaggio di errore dovuto al tentativo di utilizzo di un comando disponibile solo ad utenti autenticati;
- **Scenario principale:** la *CLI_G* segnala all'utente l'errore con un messaggio `Comando non disponibile, utente non autenticato`;
- **Precondizione:** l'utente ha inviato un comando utilizzabile solo da utenti autenticati, ovvero:
 - `list`;
 - `find`;
 - `deploy`;
 - `run`;
 - `history`;
 - `log`;
 - `edit`;
 - `delete`;
- **Postcondizione:** la *CLI_G* segnala l'errore nell'invio di un comando non utilizzabile all'utente in quanto non autenticato.

3.2.14 UC13 - Pagamento

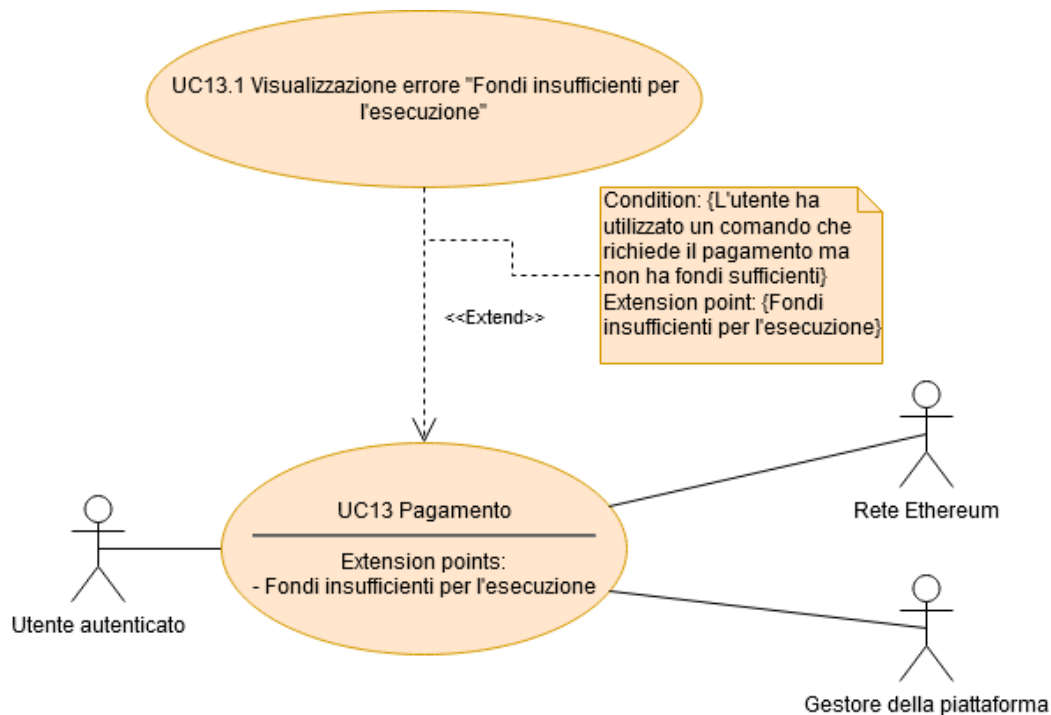


Figura 3.2.15: Diagramma UC13 - Pagamento

- **Attori Primari:** utente autenticato;
- **Attori Secondari:**
 - rete *Ethereum_G*;
 - gestore della piattaforma_G;
- **Descrizione:** l'utente effettua un pagamento ad *Ethereum_G* ed al gestore della piattaforma_G, quest'ultimo solo se necessario;
- **Scenario principale:** l'utente effettua un pagamento destinato alla rete *Ethereum_G* e, se necessario per l'esecuzione del comando, ad *AWS Lambda_G*;
- **Precondizione:** l'utente ha inviato un comando che richiede un pagamento, ovvero:
 - deploy;
 - run;
 - edit;
 - delete;
- **Postcondizione:** l'utente ha effettuato il pagamento richiesto per l'esecuzione del comando dato.
- **Estensioni:**
 - **UC13.1:** l'utente non ha fondi sufficienti per effettuare il pagamento;



3.2.14.1 UC13.1 - Visualizzazione errore “Fondi insufficienti per l'esecuzione”

- **Attori Primari:** utente autenticato;
- **Descrizione:** la CLI_G restituisce un messaggio di errore dovuto alla mancanza di fondi per l'esecuzione del comando;
- **Scenario principale:** la CLI_G segnala all'utente l'errore con un messaggio Fondi insufficienti per l'esecuzione del comando;
- **Precondizione:** l'utente ha inviato un comando correttamente. Il comando invocato dall'utente comporta dei costi;
- **Postcondizione:** la CLI_G segnala la mancanza di fondi nel $wallet_G$ dell'utente.



4 Requisiti

Ogni requisito del progetto *Etherless_G* è composto dalla seguente struttura:

- **codice identificativo:** ogni codice identificativo è univoco e conforme alla seguente codifica:

R[Importanza][Tipologia][Codice]

Il significato delle cui voci è:

- **Importanza:** ogni requisito può assumere uno dei seguenti valori:
 - * *1*: requisito obbligatorio: irrinunciabili per qualcuno degli *stakeholders_G*;
 - * *2*: requisito desiderabile: non strettamente necessari ma a valore aggiunto riconoscibile;
 - * *3*: requisito opzionale: relativamente utili oppure contrattabili più avanti nel progetto;
- **Tipologia:** ogni requisito può assumere uno dei seguenti valori:
 - * *F*: funzionale;
 - * *Q*: qualitativo;
 - * *V*: vincolo;
- **Codice:** è un identificatore univoco del requisito in forma gerarchica padre/figlio;
- **classificazione:** viene riportata l'importanza del requisito. Sebbene questa informazione sia ridondante facilita la lettura della tabella;
- **descrizione:** descrizione breve ma completa del requisito, il meno ambigua possibile;
- **fonti:** ogni requisito può derivare da una o più tra le seguenti opzioni:
 - *capitolato_G*: si tratta di un requisito individuato dalla lettura del *capitolato_G*;
 - *interno*: si tratta di un requisito che gli analisti hanno ritenuto opportuno aggiungere;



4.1 Requisiti funzionali

Tabella 4.1.1: Riassuntiva Requisiti Funzionali

Requisito	Classificazione	Descrizione	Fonti
R2F1	Desiderabile	L'utente può leggere una guida introduttiva contenente l'elenco dei comandi disponibili e una loro descrizione all'avvio dell'applicazione.	Interna UC1
R2F2	Desiderabile	L'utente può leggere una guida introduttiva contenente l'elenco dei comandi disponibili utilizzando il comando <code>help</code> .	Interna UC1
R1F3	Obbligatorio	L'utente può effettuare l'inizializzazione e l'autenticazione ad <i>Etherless_G</i> attraverso il comando <code>init</code> , rimuovendo eventuali dati salvati in precedenza.	Interno UC2, UC2.1
R1F3.1	Obbligatorio	L'applicazione chiede all'utente se desidera richiedere la creazione di un nuovo <i>wallet_G</i> attraverso la rete <i>Ethereum_G</i> .	Interna UC2
R1F3.2	Obbligatorio	L'utente richiede la creazione del nuovo <i>wallet_G</i> rispondendo S oppure s alla richiesta dell'applicazione comparsa in <i>CLI_G</i> <i>Desideri creare un nuovo account Ethereum?</i> .	Interna UC2.2
R1F3.2.1	Obbligatorio	L'applicazione richiede la creazione di un nuovo <i>wallet_G</i> attraverso la rete <i>Ethereum_G</i> .	Interna UC2.2
R1F3.2.2	Obbligatorio	L'applicazione conferma la creazione del nuovo <i>wallet_G</i> , stampando a schermo l' address associato e la relativa private key , autenticando l'utente ad <i>Etherless_G</i> .	Interna UC2.3
R1F3.3	Obbligatorio	L'utente rifiuta la creazione di un nuovo account rispondendo N oppure n alla richiesta dell'applicazione comparsa in <i>CLI_G</i> <i>Desideri creare un nuovo account Ethereum?</i> , oppure ha completato la creazione di un nuovo <i>wallet_G</i> <i>Ethereum_G</i> rispondendo S oppure s alla precedente domanda.	Interna UC2.2



Tabella 4.1.1: Riassuntiva Requisiti Funzionali

Requisito	Classificazione	Descrizione	Fonti
R1F3.3.1	Obbligatorio	L'applicazione richiede le credenziali necessarie all'autenticazione alla rete <i>Ethereum_G</i> .	Interna UC2.3
R1F3.3.2	Obbligatorio	L'utente inserisce l' address relativo al proprio <i>wallet_G Ethereum_G</i> .	Interna UC2.3.1
R1F3.3.3	Obbligatorio	L'utente inserisce la private key relativa al proprio <i>wallet_G Ethereum_G</i> .	Interna UC2.3.2
R1F3.3.4	Obbligatorio	L'applicazione verifica l'autenticità dei dati forniti dall'utente, verificando se corrispondono ad un <i>wallet_G Ethereum_G</i> esistente.	Interna UC2.3.3
R1F3.3.4.1	Obbligatorio	Se i dati forniti sono corretti, l'applicazione autentica l'utente utilizzando il <i>wallet_G</i> corrispondente.	Interna UC2.3.3
R1F3.3.4.2	Obbligatorio	L'applicazione restituisce un messaggio di errore nel caso non possa trovare un wallet con le credenziali indicate dall'utente.	Interna 2.2.5
R2F3.4	Desiderabile	A seguito della corretta autenticazione, l'applicazione richiederà tramite <i>CLI_G</i> l'attivazione dell'autenticazione persistente tramite la domanda <i>Desideri attivare il login persistente?</i> .	Interna UC2.3.4
R2F3.4.1	Desiderabile	Se l'utente risponde s oppure S , l'applicazione attiva l'autenticazione persistente, effettuando il salvataggio dei dati relativi al <i>wallet_G</i> utente nel <i>filesystem_G</i> locale.	Interna UC2.3.4.1
R2F3.4.2	Desiderabile	Se l'utente risponde N oppure n , l'applicazione non attiva l'autenticazione persistente, essa richiederà quindi le credenziali utente ad ogni riapertura.	Interna UC2.3.4.2
R2F3.5	Desiderabile	Un utente autenticato ha la possibilità di effettuare il logout dall'applicazione rimuovendo eventuali dati esistenti attraverso etherless init .	Interna UC2.1



Tabella 4.1.1: Riassuntiva Requisiti Funzionali

Requisito	Classificazione	Descrizione	Fonti
R2F4	Desiderabile	Viene segnalato un errore nel caso in cui un utente non autenticato provasse ad utilizzare un comando che necessita di autenticazione.	Interna UC12
R2F5	Desiderabile	L'utente può verificare il suo stato all'interno dell'applicazione, in particolare del <i>wallet_G</i> a cui è attualmente connesso utilizzando il comando <i>who</i> .	Interna UC3
R2F5.1	Desiderabile	L'applicazione visualizza tramite <i>CLI_G</i> l' <i>address</i> che caratterizza il <i>wallet_G</i> attualmente in uso dall'applicazione.	Interna UC3.1
R2F5.2	Desiderabile	Viene notificata la mancata autenticazione da parte dell'utente a seguito dell'utilizzo del comando <i>who</i> senza essersi autenticato.	Interna UC3.2
R1F6	Obbligatorio	L'utente autenticato può visualizzare l'elenco di tutti i prodotti software pubblicati su <i>Etherless_G</i> utilizzando il comando <i>list</i> .	Capitolato
R1F6.1	Obbligatorio	L'applicazione visualizza l'elenco dei prodotti software pubblicati su <i>Etherless_G</i> nella <i>CLI_G</i> .	Interna UC4
R3F7	Opzionale	L'utente può cercare l'esistenza di un prodotto software specifico su <i>Etherless_G</i> utilizzando il comando <i>find</i> .	Interna UC5
R3F7.1	Opzionale	L'applicazione visualizza il risultato della ricerca dopo l'utilizzo del comando <i>find</i> seguito dal nome del prodotto software che si desidera cercare.	Interna UC5.1
R3F7.2	Opzionale	Viene segnalato un errore nel caso in cui il prodotto software ricercato dall'utente non fosse disponibile su <i>Etherless_G</i> .	Interna UC5.2
R1F8	Obbligatorio	L'utente autenticato può pubblicare il proprio prodotto software su <i>Etherless_G</i> , utilizzando il comando <i>deploy_G</i> , per renderlo disponibile all'esecuzione.	Capitolato



Tabella 4.1.1: Riassuntiva Requisiti Funzionali

Requisito	Classificazione	Descrizione	Fonti
R1F8.1	Obbligatorio	L'utente può eseguire il comando deploy indicando il percorso del file contenente il prodotto software da pubblicare su <i>Etherless_G</i> .	Interna UC6.1
R1F8.1.1	Obbligatorio	Viene segnalato un errore nel caso in cui il percorso del prodotto software da pubblicare sia errato.	Interna UC6.2
R1F8.2	Obbligatorio	L'utente può eseguire il comando deploy indicando il nome che desidera che il suo prodotto software abbia una volta pubblicato su <i>Etherless_G</i> .	Interna UC6.3
R1F8.2.1	Obbligatorio	Viene segnalato un errore nel caso in cui il nome scelto per il prodotto software sia già in uso su <i>Etherless_G</i> .	Interna UC6.4
R1F8.3	Obbligatorio	L'applicazione notifica all'utente autenticato, in mancanza di errori, l'avvenuta pubblicazione del prodotto software su <i>Etherless_G</i> .	Interna UC6
R1F8.4	Obbligatorio	Viene segnalato un errore nel caso in cui, nel tentativo di pubblicare un prodotto software su <i>Etherless_G</i> , l'utente non disponesse di fondi sufficienti nel proprio <i>wallet_G</i> .	Interna UC13.1
R1F9	Obbligatorio	L'utente autenticato può eseguire un prodotto software precedentemente pubblicato su <i>Etherless_G</i> , attraverso il comando run , specificandone nome e parametri. Questa azione prevede il pagamento del servizio al gestore della piattaforma.	Capitolato
R1F9.1	Obbligatorio	L'utente autenticato può specificare dopo il comando run il nome del prodotto software da eseguire.	Interna UC7.1
R1F9.2	Obbligatorio	L'utente autenticato può specificare, dopo il nome del prodotto software, dei parametri che verranno passati al prodotto software da eseguire.	Interna UC7.3



Tabella 4.1.1: Riassuntiva Requisiti Funzionali

Requisito	Classificazione	Descrizione	Fonti
R1F9.3	Obbligatorio	L'esecuzione di un prodotto software comporta l'attesa da parte dell'utente che il programma fornisca all'utente i risultati dell'esecuzione qualora questa venga conclusa con successo.	Capitolato
R1F9.4	Obbligatorio	L'utente viene informato qualora l'esecuzione di un prodotto software venisse interrotta a causa di un errore.	Interna UC7.4
R1F9.5	Obbligatorio	Viene segnalato un errore nel caso in cui il nome del prodotto software da eseguire in <i>Etherless_G</i> non corrisponda a nessun prodotto software disponibile.	Interna UC7.2
R1F9.6	Obbligatorio	L'applicazione notifica all'utente autenticato, in mancanza di errori, il risultato dell'esecuzione del prodotto software.	Interna UC7
R1F9.7	Obbligatorio	Viene segnalato un errore nel caso in cui, nel tentativo di eseguire un prodotto software su <i>Etherless_G</i> , l'utente non disponesse di fondi sufficienti nel proprio <i>wallet_G</i> .	Interna UC13.1
R3F10	Opzionale	L'utente può richiedere di visualizzare le esecuzioni di prodotti software precedentemente pagate.	Interna UC8
R3F11	Opzionale	L'utente può richiedere lo stato, e se disponibile il risultato della esecuzione di un prodotto software precedentemente pagata.	Interna UC9
R3F12	Opzionale	L'utente autenticato specifica l'identificativo univoco della esecuzione di un prodotto software da visualizzare.	Interna UC9.1
R3F13	Opzionale	Viene segnalato un errore qualora l'utente specificasse una esecuzione mai avvenuta o richiesta da un altro utente.	Interna UC9.2
R3F14	Opzionale	L'utente autenticato può richiedere, utilizzando il comando <code>edit</code> , la modifica di un prodotto software da lui pubblicato in <i>Etherless_G</i> .	Interna UC10



Tabella 4.1.1: Riassuntiva Requisiti Funzionali

Requisito	Classificazione	Descrizione	Fonti
R3F14.1	Opzionale	L'utente autenticato specifica il nome del prodotto software da modificare inserendo il nome di quest'ultimo dopo il comando <code>edit</code> .	Interna UC10.1
R3F14.2	Opzionale	Viene segnalata la mancanza del prodotto software recante il nome specificato dall'utente dopo il comando <code>edit</code> .	Interna UC10.2
R3F14.3	Opzionale	Qualora l'utente non fosse il proprietario del prodotto software dal nome specificato verrà segnalato all'utente autenticato un errore relativo ai permessi di modifica del prodotto software.	Interna UC10.3
R3F14.4	Opzionale	L'utente autenticato può specificare, dopo il nome del software specificato a seguito del comando <code>run</code> , il percorso nel <i>filesystem locale_G</i> del file con il quale sostituire il prodotto software dal nome specificato.	Interna UC10.4
R3F14.5	Opzionale	Viene segnalata l'inesistenza nel <i>filesystem locale_G</i> del file specificato.	Interna UC10.5
R3F14.6	Opzionale	Viene segnalato un errore nel caso in cui, nel tentativo di modificare un prodotto software su <i>Etherless_G</i> , l'utente non disponesse di fondi sufficienti nel proprio <i>wallet_G</i> .	Interna UC13.1
R1F15	Obbligatoria	L'utente può revocare i diritti d'uso del suo prodotto software, rimuovendolo dalla piattaforma <i>Etherless_G</i> attraverso il comando <code>delete</code> .	Capitolato
R1F15.1	Obbligatoria	L'utente autenticato può specificare il nome del prodotto software da rimuovere dalla piattaforma dopo il comando <code>delete</code> .	Interna UC11.1
R1F15.2	Obbligatoria	Viene segnalato un errore nel caso in cui il nome del prodotto software da rimuovere da <i>Etherless_G</i> non corrisponda a nessun prodotto software disponibile.	Interna UC11.2



Tabella 4.1.1: Riassuntiva Requisiti Funzionali

Requisito	Classificazione	Descrizione	Fonti
R1F15.3	Obbligatoria	Viene segnalato un errore nel caso in cui il prodotto software da rimuovere dalla piattaforma non sia di proprietà dell'utente che ne richiede l'eliminazione.	Interna UC11.3
R1F15.4	Obbligatorio	L'applicazione notifica all'utente autenticato, in mancanza di errori, l'avvenuta eliminazione del prodotto software da <i>Etherless_G</i> .	Interna UC11
R1F15.5	Obbligatorio	Viene segnalato un errore nel caso in cui, nel tentativo di rimuovere un prodotto software da <i>Etherless_G</i> , l'utente non disponesse di fondi sufficienti nel proprio <i>wallet_G</i> .	Interna UC13.1
R1F16	Obbligatorio	Tutti i pagamenti devono avvenire all'interno della rete <i>Ethereum_G</i> .	Capitolato, Interna UC13



4.2 Requisiti di qualità

Tabella 4.2.1: Riassuntiva Requisiti di Qualità

Requisito	Classificazione	Descrizione	Fonti
R1Q1	Obbligatorio	<i>Etherless_G</i> deve essere sviluppato utilizzando un approccio <i>asincrono_G</i> incentrato su promise/async-await_G .	Capitolato
R1Q2	Obbligatorio	Durante tutta la fase di sviluppo, typescript-eslint_G deve essere utilizzato e validato staticamente attraverso lo strumento ESLint .	Capitolato
R1Q3	Obbligatorio	Gli <i>smart contract_G</i> devono essere aggiornabili.	Capitolato
R1Q4	Obbligatorio	Il codice sorgente di Etherless deve essere pubblicato e versionato attraverso GitLab o Github .	Capitolato
R1Q5	Obbligatorio	La documentazione necessaria all'utilizzo di <i>Etherless_G</i> dovrà essere pubblicata attraverso GitHub o GitLab .	Capitolato
R1Q6	Obbligatorio	Il sistema <i>Etherless_G</i> , sarà distribuito sotto <i>Licenza MIT_G</i> .	Capitolato
R1Q7	Obbligatorio	La documentazione esterna, come il manuale d'uso, deve essere scritta interamente in inglese, così come tutto il codice del progetto ed i relativi commenti.	Capitolato
R1Q8	Obbligatorio	Devono essere rispettate per intero le norme descritte all'interno delle <i>Norme di Progetto 1.0.0</i> .	Capitolato
R1Q9	Obbligatorio	Devono essere rispettate per intero le norme descritte all'interno delle <i>Piano di Qualifica 1.0.0</i> .	Capitolato



4.3 Requisiti di vincolo

Tabella 4.3.1: Riassuntiva Requisiti di Vincolo

Requisito	Classificazione	Descrizione	Fonti
R1V1	Obbligatorio	<i>Etherless_G</i> deve essere sviluppato utilizzando un approccio <i>asincrono_G</i> incentrato su promise/async-await_G .	Capitolato
R1V2	Obbligatorio	Durante tutta la fase di sviluppo, typescript-eslint_G deve essere utilizzato e validato staticamente attraverso lo strumento ESLint .	Capitolato
R1V3	Obbligatorio	Gli <i>smart contract_G</i> devono essere aggiornabili.	Capitolato
R1V4	Obbligatorio	Il codice sorgente di Etherless deve essere pubblicato e versionato attraverso GitLab o Github .	Capitolato
R1V5	Obbligatorio	La documentazione necessaria all'utilizzo di <i>Etherless_G</i> dovrà essere pubblicata attraverso GitHub o GitLab .	Capitolato
R1V6	Obbligatorio	Il sistema <i>Etherless_G</i> sarà distribuito sotto <i>Licenza MIT_G</i> .	Capitolato
R1V7	Obbligatorio	Il progetto deve essere testato e sviluppato su tutti i seguenti ambienti Locale , Test , Staging .	Capitolato
R1V7.1	Obbligatorio	Per l'ambiente locale sarà usato il framework testrpc fornito da Truffle .	Capitolato, Interna
R1V7.2	Obbligatorio	L'ambiente di test sarà lo stesso ambiente utilizzato per lo sviluppo in locale.	Capitolato
R1V7.3	Obbligatorio	L'ambiente di staging, che deve essere accessibile pubblicamente, deve utilizzare la rete <i>Ethereum_G</i> Ropsten .	Capitolato
R1V8	Obbligatorio	RedBabel dovrà essere citato nei crediti all'interno del file readme.md .	Capitolato
R1V9	Obbligatorio	Lo sviluppatore sarà citato nei crediti di copyright.	Capitolato



Tabella 4.3.1: Riassuntiva Requisiti di Vincolo

Requisito	Classificazione	Descrizione	Fonti
R1V10	Obbligatorio	etherless-server dovrà essere sviluppato utilizzando il Serverless framework .	Capitolato
R1V11	Obbligatorio	<i>Etherless_G</i> deve utilizzare l'infrastruttura <i>AWS_G</i> .	Capitolato
R1V12	Obbligatorio	L'infrastruttura <i>Serverless_G</i> sarà realizzata attraverso l'uso di <i>AWS Lambda_G</i> .	Capitolato

4.4 Tracciamento

4.4.1 Fonte - Requisito

Tabella 4.4.1: Riassuntiva Fonte - Requisiti

Fonte	Requisiti
Capitolato	R1F3, R1F6, R1F8 R1F9 R1F9.3 R1F15 R1F16 R1Q1 R1Q2 R1Q3 R1Q4 R1Q5 R1Q6 R1Q7 R1Q8 R1Q9 R1V1 R1V2 R1V3 R1V4 R1V5 R1V6 R1V7 R1V7.1 R1V7.2 R1V7.3 R1V8 R1V9 R1V10 R1V11 R1V12
Interna	R2F1 R2F2 R1F3.1 R1F3.2 R1F3.2.1 R1F3.2.2 R1F3.3 R1F3.3.1 R1F3.3.2 R1F3.3.3 R1F3.3.4 R1F3.3.4.1 R1F3.3.4.2 R2F3.4 R2F3.4.1 R2F3.4.2 R2F3.5 R2F4 R2F5 R2F5.1 R2F5.2 R1F6.1 R3F7 R3F7.1 R3F7.2 R1F8.1 R1F8.1.1 R1F8.2 R1F8.2.1 R1F8.3 R1F8.4 R1F9.1 R1F9.2 R1F9.4 R1F9.5 R1F9.6 R1F9.7 R3F10 R3F11 R3F12 R3F13 R3F14 R3F14.1 R3F14.2 R3F14.3 R3F14.4 R3F14.5 R3F14.6 R1F15.1 R1F15.2 R1F15.3 R1F15.4 R1F15.5 R1F16 R1V7.1
UC1	R2F1 R2F2
UC2	R1F3.1
UC2.1	R2F3.5
UC2.2	R1F3.2 R1F3.2.1 R1F3.3
UC2.3	R1F3.2.2 R1F3.3.1
UC2.3.1	R1F3.3.2



Tabella 4.4.1: Riassuntiva Fonte - Requisiti

Fonte	Requisiti
UC2.3.2	R1F3.3.3
UC2.3.3	R1F3.3.4 R1F3.3.4.1
UC2.3.5	R1F3.3.4.2
UC2.3.4	R2F3.4
UC2.3.4.1	R2F3.4.1
UC2.3.4.3	R2F3.4.2
UC2.3.4.2	Requisito
UC2.3.5	Requisito
UC3	R2F5
UC3.1	R2F5.1
UC3.2	R2F5.2
UC4	R1F6.1
UC5	R3F7
UC5.1	R3F7.1
UC5.2	R3F7.2
UC6	R1F8.3
UC6.1	R1F8.1
UC6.2	R1F8.1.1
UC6.3	R1F8.2
UC6.4	R1F8.2.1
UC7	R1F9.6
UC7.1	R1F9.1



Tabella 4.4.1: Riassuntiva Fonte - Requisiti

Fonte	Requisiti
UC7.2	R1F9.5
UC7.3	R1F9.2
UC7.4	R1F9.4
UC8	R3F10
UC9	R3F11
UC9.1	R3F12
UC9.2	R3F13
UC10	R3F14
UC10.1	R3F14.1
UC10.2	R3F14.2
UC10.3	R3F14.3
UC10.4	R3F14.4
UC10.5	R3F14.5
UC11	R1F15.4
UC11.1	R1F15.1
UC11.2	R1F15.2
UC11.3	R1F15.3
UC12	R2F4
UC13	R1F16
UC13.1	R1F8.4 R1F9.7 R3F14.6 R1F15.5



4.4.2 Requisito - Fonte

Tabella 4.4.2: Riassuntiva Requisito - Fonti

Requisito	Fonti
R2F1	Interna UC1
R2F2	Interna UC1
R1F3	Capitolato
R1F3.1	Interna UC2
R1F3.2	Interna UC2.2
R1F3.2.1	Interna UC2.2
R1F3.2.2	Interna UC2.3
R1F3.3	Interna UC2.2
R1F3.3.1	Interna UC2.3
R1F3.3.2	Interna UC2.3.1
R1F3.3.3	Interna UC2.3.2
R1F3.3.4	Interna UC2.3.3
R1F3.3.4.1	Interna UC2.3.3
R1F3.3.4.2	Interna UC2.3.5
R2F3.4	Interna UC2.3.4
R2F3.4.1	Interna UC2.3.4.1
R2F3.4.2	Interna UC2.3.4.2
R2F3.5	Interna UC2.1
R2F4	Interna UC12
R2F5	Interna UC3
R2F5.1	Interna UC3.1



Tabella 4.4.2: Riassuntiva Requisito - Fonti

Requisito	Fonti
R2F5.2	Interna UC3.2
R1F6	Capitolato
R1F6.1	Interna UC4
R3F7	Interna UC5
R3F7.1	Interna UC5.1
R3F7.2	Interna UC5.2
R1F8	Capitolato
R1F8.1	Interna UC6.1
R1F8.1.1	Interna UC6.2
R1F8.2	Interna UC6.3
R1F8.2.1	Interna UC6.4
R1F8.3	Interna UC6
R1F8.4	Interna UC13.1
R1F9	Capitolato
R1F9.1	Interna UC7.1
R1F9.2	Interna UC7.3
R1F9.3	Capitolato
R1F9.5	Interna UC7.2
R1F9.6	Interna UC7
R1F9.7	Interna UC13.1
R3F10	Interna UC8
R3F11	Interna UC9



Tabella 4.4.2: Riassuntiva Requisito - Fonti

Requisito	Fonti
R3F12	Interna UC9
R3F12	Interna UC9.1
R3F13	Interna UC9.2
R3F14	Interna UC10
R3F14.1	Interna UC10.1
R3F14.2	Interna UC10.2
R3F14.3	Interna UC10.3
R3F14.4	Interna UC10.4
R3F14.5	Interna UC10.5
R3F14.6	Interna UC13.1
R1F15	Capitolato
R1F15.1	Interna UC11.1
R1F15.2	Interna UC11.2
R1F15.3	Interna UC11.3
R1F15.4	Interna UC11
R1F15.5	Interna UC13.1
R1F16	Capitolato, Interna UC13
R1Q1	Capitolato
R1Q2	Capitolato
R1Q3	Capitolato
R1Q4	Capitolato
R1Q5	Capitolato



Tabella 4.4.2: Riassuntiva Requisito - Fonti

Requisito	Fonti
R1Q6	Capitolato
R1Q7	Capitolato
R1Q8	Capitolato
R1Q9	Capitolato

4.5 Evoluzione attesa del sistema

Si prevede il sistema possa evolversi per rendere più accessibili e fruibili i contenuti pubblicati nella piattaforma, principalmente:

- permettere di allegare ad un prodotto software una descrizione per rendere i prodotti software più facilmente cercabili tra tutti quelli disponibili, anche attraverso uno strumento esterno ad **etherless-cli** (ad esempio un sito web) con lo scopo di attrarre una utenza quanto più ampia possibile;
- permettere di fornire una rappresentazione dei parametri accettati dal prodotto software, questo anche in previsione di fornire una interfaccia grafica opzionale per l'esecuzione dei prodotti software, rendendo questi ultimi più fruibili ad un pubblico più ampio delle sole figure nell'ambito IT.

4.6 Considerazioni

I requisiti potranno subire delle variazioni in futuro, per apportare degli aggiornamenti alle voci presenti o delle migliorie. Dunque eventuali modifiche e/o espansioni sono lasciate a momenti futuri.