

# roundabout

*Roundabout - Etherless*

## Analisi dei requisiti

<b>Versione</b>	0.0.2
<b>Approvazione</b>	
<b>Redazione</b>	Egon Galvani Antonio Zlatkovski
<b>Verifica</b>	
<b>Stato</b>	Non approvato
<b>Uso</b>	Esterno
<b>Destinato a</b>	Roundabout Prof. Tullio Vardanega Prof. Riccardo Cardin

### **Descrizione**

*Analisi dei requisiti del gruppo Roundabout per la realizzazione del progetto Etherless*

team.roundabout.13@gmail.com

## Diario delle modifiche

Versione	Data	Nominativo	Ruolo	Descrizione
0.0.2	2020-03-18	Antonio Zlatkovski	<i>Analista</i>	Scrittura dell'introduzione del documento
0.0.1	2020-03-14	Egon Galvani	<i>Analista</i>	Creata struttura del documento in L <sup>A</sup> T <sub>E</sub> X

## Indice

<b>1</b>	<b>Introduzione</b>	<b>5</b>
1.1	Scopo del documento . . . . .	5
1.2	Scopo del prodotto . . . . .	5
1.3	Documenti complementari . . . . .	5
1.3.1	Glossario . . . . .	5
1.4	Riferimenti . . . . .	5
1.4.1	Normativi . . . . .	5
1.4.2	Informativi . . . . .	5

## Elenco delle figure

## Elenco delle tabelle

# 1 Introduzione

## 1.1 Scopo del documento

Il presente documento ha l'obiettivo di individuare le funzionalità e i casi d'uso previsti dal progetto Etherless, proposto dall'azienda Red Babel. Le informazioni qui riportate, individuate da un'approfondita analisi del capitolato stesso e dai successivi incontri con il proponente, rappresentano la base di partenza per la successiva fase di progettazione.

## 1.2 Scopo del prodotto

Si vuole sviluppare una piattaforma cloud che consenta agli sviluppatori di fare il deploy di funzioni Javascript e gestisca il pagamento per la loro esecuzione tramite la piattaforma Ethereum. Il prodotto finale prevede quindi l'integrazione di due tecnologie, Serverless e Ethereum.

Il lato Serverless si occupa dell'esecuzione delle funzioni fornite dagli sviluppatori. Tali funzioni vengono salvate ed eseguite in un servizio cloud esterno, quale Amazon Web Services. In particolare si vuole utilizzare AWS Lambda per organizzarle secondo la politica FaaS, ovvero Function as a Service.

La richiesta di utilizzo di una funzione e il successivo pagamento vengono invece gestiti tramite la piattaforma Ethereum sfruttando gli smart contracts. Il pagamento viene effettuato in ETH. Una percentuale significativa di ogni pagamento viene riservata agli amministratori del servizio. Lo sviluppatore e l'utente finale interagiscono con il prodotto tramite una CLI che prevede alcuni comandi intuitivi:

## 1.3 Documenti complementari

### 1.3.1 Glossario

I termini tecnici utilizzati in questo documento sono contrassegnati da una 'G' a pedice ed ulteriormente approfonditi nel Glossario denominato "Glossario Esterno 1.0.0", disponibile in allegato al presente documento.

## 1.4 Riferimenti

### 1.4.1 Normativi

- **Norme di progetto:** *Norme di progetto v1.0.0*;
- **Verbale esterno con il proponente:** *VE\_2020\_03\_18*;
- **Capitolato d'appalto Etherless:**  
<https://www.math.unipd.it/~tullio/IS-1/2019/Progetto/C2.pdf>.

### 1.4.2 Informativi

- **Studio di fattibilità:** *Studio di fattibilità v1.0.0*;
- **Sito ufficiale Ethereum:** <https://ethereum.org/>;
- **Sito del framework Serverless:** <https://serverless.com/>;
- **Amazon Web Services:** <https://aws.amazon.com/>.