



Zeus Code - Progetto "P2PCS"

## Studio di Fattibilità

<b>Versione</b>	1.0.0
<b>Approvazione</b>	Marco Dalla B�
<b>Redazione</b>	Andrea Pigatto Riccardo Basso Riccardo Dario
<b>Verifica</b>	Irina Hornoiu Diba Meysami
<b>Stato</b>	Approvato
<b>Uso</b>	Interno
<b>Destinato a</b>	Zeus Code Prof. Tullio Vardanega Prof. Riccardo Cardin

### **Descrizione**

*Studio di Fattibilit  dei capitolati proposti.*

zeuscode17@gmail.com

## Diario delle modifiche

Versione	Data	Nominativo	Ruolo	Descrizione
1.0.0	2019-03-10	Marco Dalla B�	<i>Responsabile di Progetto</i>	Approvazione del documento.
0.0.9	2019-03-09	Irina Horno�, Diba Meysami	<i>Verificatrici</i>	Revisione degli studi di fattibilit� dei capitolati C2 e C3.
0.0.8	2019-03-08	Riccardo Dario	<i>Analista</i>	Stesura dello studio di fattibilit� del capitolato C2.
0.0.7	2019-03-08	Andrea Pigatto	<i>Analista</i>	Stesura dello studio di fattibilit� del capitolato C3.
0.0.6	2019-03-07	Irina Horno�, Diba Meysami	<i>Verificatrici</i>	Revisione degli studi di fattibilit� dei capitolati C1, C4 e C6.
0.0.5	2019-03-06	Riccardo Basso, Irina Horno�	<i>Analista, Verificatrice</i>	Stesura e revisione dello studio di fattibilit� del capitolato C5.
0.0.4	2019-03-06	Andrea Pigatto	<i>Analista</i>	Stesura dello studio di fattibilit� del capitolato C4.
0.0.3	2019-03-05	Riccardo Dario	<i>Analista</i>	Stesura dello studio di fattibilit� del capitolato C1.
0.0.2	2019-03-05	Riccardo Basso	<i>Analista</i>	Stesura dello studio di fattibilit� del capitolato C6.
0.0.1	2019-03-05	Andrea Pigatto	<i>Analista</i>	Creato lo scheletro del documento L�T�X e creato il frontespizio.

## Indice

<b>1</b>	<b>Introduzione</b>	<b>4</b>
1.1	Scopo del Documento	4
1.2	Glossario	4
1.3	Riferimenti	4
1.3.1	Normativi	4
1.3.2	Informativi	4
<b>2</b>	<b>Capitolato Scelto C5 - P2PCS</b>	<b>5</b>
2.1	Informazioni generali	5
2.2	Descrizione	5
2.3	Finalità del progetto	5
2.4	Tecnologie interessate	5
2.5	Aspetti positivi	5
2.6	Criticità e fattori di rischio	6
2.7	Conclusioni	6
<b>3</b>	<b>Valutazioni sugli altri capitolati</b>	<b>7</b>
3.1	Capitolato C1 - Butterfly	7
3.1.1	Informazioni generali	7
3.1.2	Descrizione	7
3.1.3	Finalità del progetto	7
3.1.4	Tecnologie interessate	7
3.1.5	Aspetti positivi	8
3.1.6	Criticità e fattori di rischio	8
3.1.7	Conclusioni	8
3.2	Capitolato C2 - Colletta	9
3.2.1	Informazioni Generali	9
3.2.2	Descrizione	9
3.2.3	Finalità del progetto	9
3.2.4	Tecnologie interessate	9
3.2.5	Aspetti positivi	9
3.2.6	Criticità e fattori di rischio	10
3.2.7	Conclusioni	10
3.3	Capitolato C3 - G&B	11
3.3.1	Informazioni generali	11
3.3.2	Descrizione	11
3.3.3	Finalità del progetto	11
3.3.4	Tecnologie interessate	11
3.3.5	Aspetti positivi	11
3.3.6	Criticità e fattori di rischio	11
3.3.7	Conclusioni	12
3.4	Capitolato C4 - MegAlexa	13
3.4.1	Informazioni generali	13
3.4.2	Descrizione	13
3.4.3	Finalità del progetto	13
3.4.4	Tecnologie interessate	13

3.4.5	Aspetti positivi . . . . .	13
3.4.6	Criticità e fattori di rischio . . . . .	14
3.4.7	Conclusioni . . . . .	14
3.5	Capitolato C6 - Soldino . . . . .	15
3.5.1	Informazioni generali . . . . .	15
3.5.2	Descrizione . . . . .	15
3.5.3	Finalità del progetto . . . . .	15
3.5.4	Tecnologie interessate . . . . .	15
3.5.5	Aspetti positivi . . . . .	16
3.5.6	Criticità e fattori di rischio . . . . .	16
3.5.7	Conclusioni . . . . .	16

# 1 Introduzione

## 1.1 Scopo del Documento

Il seguente documento ha l'obiettivo di descrivere le motivazioni che hanno spinto il gruppo alla scelta del capitolato<sub>G</sub> C5, ovvero *P2PCS*, con la conseguente esclusione degli altri progetti proposti.

## 1.2 Glossario

Al fine di evitare possibili ambiguità relative al linguaggio utilizzato nei documenti formali, viene fornito il *Glossario v1.0.0*. In questo documento vengono definiti e descritti tutti i termini con un significato particolare. Per facilitare la lettura, i termini saranno contrassegnati da una 'G' a pedice.

## 1.3 Riferimenti

### 1.3.1 Normativi

- **Norme di Progetto:** *Norme di Progetto v1.0.0*.

### 1.3.2 Informativi

- **Capitolato<sub>G</sub> d'appalto C1 -Butterfly, monitor per processi CI/CD:**  
<https://www.math.unipd.it/~tullio/IS-1/2018/Progetto/C1.pdf>;
- **Capitolato<sub>G</sub> d'appalto C2 - Colletta, piattaforma raccolta dati di analisi di testo:**  
<https://www.math.unipd.it/~tullio/IS-1/2018/Progetto/C2.pdf>;
- **Capitolato<sub>G</sub> d'appalto C3 - G&B, monitoraggio intelligente di processi DevOps<sub>G</sub>:**  
<https://www.math.unipd.it/~tullio/IS-1/2018/Progetto/C3.pdf>;
- **Capitolato<sub>G</sub> d'appalto C4 - MegAlexa, arricchitore di skill di Amazon Alexa:**  
<https://www.math.unipd.it/~tullio/IS-1/2018/Progetto/C4.pdf>;
- **Capitolato<sub>G</sub> d'appalto C5 - P2PCS, piattaforma di peer-to-peer car sharing:**  
<https://www.math.unipd.it/~tullio/IS-1/2018/Progetto/C5.pdf>;
- **Capitolato<sub>G</sub> d'appalto C6: - Soldino, piattaforma Ethereum<sub>G</sub> per pagamenti IVA:**  
<https://www.math.unipd.it/~tullio/IS-1/2018/Progetto/C6.pdf>;

## 2 Capitolato Scelto C5 - P2PCS

### 2.1 Informazioni generali

- **Nome:** P2PCS: piattaforma di peer-to-peer<sub>G</sub> car sharing;
- **Proponente:** GaiaGo S.r.l;
- **Committente:** Prof. Tullio Vardanega e Prof. Riccardo Cardin.

### 2.2 Descrizione

Questo capitolato propone di integrare un'applicazione Android con l'obiettivo di fornire agli utenti la possibilità di condividere il proprio veicolo con altri utenti (car sharing) e ad invogliarne l'utilizzo attraverso delle strategie di gamification. I due vantaggi principali sono i seguenti:

- **Proprietario del veicolo:** avrà un vantaggio economico in quanto farà fruttare il proprio mezzo quando non deve utilizzarlo, prestandolo ad altri utenti;
- **Usufrutente:** avrà il vantaggio di avere a disposizione un veicolo per potersi spostare pagando solamente le ore effettive di utilizzo.

### 2.3 Finalità del progetto

L'applicazione si baserà su un calendario nel quale il proprietario di un veicolo potrà indicare in che giorni e orari il suo mezzo sarà disponibile. Ogni utente potrà cercare un mezzo libero nella propria zona, prenotarlo e ritirarne le chiavi.

### 2.4 Tecnologie interessate

- **Node.js<sub>G</sub>:** framework<sub>G</sub> open-source lato server basato su JavaScript con un modello asincrono di I/O guidato da eventi;
- **Google Cloud:** per la gestione del database;
- **Octalysis:** framework<sub>G</sub> per integrare una strategia di gamification<sub>G</sub> volta a rendere più accattivante e aggiornata l'applicazione;
- **Movens:** piattaforma open-source<sub>G</sub> che fornisce funzionalità di gestione di servizi nelle smart cities. Copre diversi livelli tecnologici come: dispositivi fisici, connettività e servizi applicativi;
- **Android Studio:** framework<sub>G</sub> per lo sviluppo di applicazioni Android.
- **Java, Kotlin<sub>G</sub>:** linguaggi utilizzati per lo sviluppo di applicazioni Android.

### 2.5 Aspetti positivi

- Acquisizione di nuove competenze nel campo della programmazione mobile e nello sviluppo di un'architettura peer-to-peer<sub>G</sub>;
- Apprendimento di nuovi linguaggi e piattaforme, come Node.js, Kotlin e Movens.
- Comprendere la teoria della gamification<sub>G</sub> e applicarla all'interno di un'applicazione.

## 2.6 Criticità e fattori di rischio

- La concorrenza propone già delle valide alternative e far emergere l'applicazione nel mercato e renderla un prodotto superiore richiederà molto impegno;
- La cessione del proprio veicolo ad utenti terzi può portare diffidenza verso il servizio proposto dall'applicazione.

## 2.7 Conclusioni

Il gruppo ha espresso un giudizio principalmente positivo verso questo capitolato<sub>G</sub>, in quanto convinto di poter proporre un prodotto superiore alla concorrenza e in grado di soddisfare le richieste di mercato e del committente.

## 3 Valutazioni sugli altri capitoli

### 3.1 Capitolo C1 - Butterfly

#### 3.1.1 Informazioni generali

- **Nome:** Butterfly: monitor per processi CI/CD;
- **Proponente:** Imola Informatica;
- **Committente:** Prof. Tullio Vardanega e Prof. Riccardo Cardin.

#### 3.1.2 Descrizione

Il progetto Butterfly propone lo sviluppo di una piattaforma di notifica che raccolga le segnalazioni provenienti dai vari applicativi utilizzati dall'azienda e le riporti nella forma desiderata dall'utilizzatore finale.

#### 3.1.3 Finalità del progetto

Il prodotto finale utilizza un pattern Producer-Consumer che raccoglie le varie segnalazioni mandate dalle applicazioni e le indirizza attraverso i canali scelti dall'utilizzatore. L'azienda propone una soluzione a quattro componenti, così strutturate:

- **Producers:** raccolgono le segnalazioni provenienti dalle varie applicazioni e le pubblicano, sotto forma di messaggio, all'interno dello specifico topic;
- **Broker:** strumento che istanzia e gestisce i topic;
- **Consumers:** componenti che hanno il compito di abbonarsi ai topic adeguati, recuperarne i messaggi ed inviarli verso i destinatari finali. I componenti richiesti hanno come finalità l'invio di segnalazioni attraverso Telegram, Slack e Email;
- **Componente custom specifico:** funzione che permette, attraverso dei metadati relativi agli utenti, di inviare le informazioni solo a chi interessato.

#### 3.1.4 Tecnologie interessate

Per lo sviluppo dei componenti applicativi, l'azienda proponente consiglia:

- **Java, Python<sub>G</sub>, Node.js<sub>G</sub>:** alternative di linguaggi per lo sviluppo dell'applicativo suggerite dal proponente;
- **Apache Kafka<sub>G</sub>:** software open-source per la gestione delle operazioni tra i vari client, da utilizzare come Broker;
- **Docker<sub>G</sub>:** per creare i container relativi alle diverse componenti;
- **API<sub>G</sub> Redmine<sub>G</sub>, GitLab<sub>G</sub>, SonarQube<sub>G</sub>, Telegram<sub>G</sub>, Slack<sub>G</sub>:** utilizzate per potersi interfacciare con omonime applicazioni.

Ulteriori richieste del proponente:

- Rispettare i 12 fattori esposti dal documento "The Twelve-Factor App";
- Fornire API REST per tutte le componenti utilizzate;
- Utilizzo di test unitari e d'integrazione, test di sistema sull'intero sistema.



### 3.1.5 Aspetti positivi

- Le tecnologie proposte hanno larga diffusione nel mondo lavorativo ed approfondire la conoscenza su di esse è un aspetto apprezzato dal gruppo;
- Java è materia di studio nel nostro corso di laurea, per cui il capitolato<sub>G</sub> offre la possibilità di migliorare la padronanza di questo linguaggio.

### 3.1.6 Criticità e fattori di rischio

- Lo sviluppo del componente Producer permetterebbe solamente l'apprendimento di aspetti marginali delle tecnologie coinvolte;
- Il lavoro per la raccolta dati appare ripetitivo e le API<sub>G</sub> da utilizzare sembrano altamente specifiche per il progetto. Probabilmente queste conoscenze acquisite saranno poco spendibili nel futuro, specie se comparate alle offerte di altri capitoli;
- L'interesse da parte del gruppo di lavoro per questo capitolato<sub>G</sub> si è dimostrato scarso.

### 3.1.7 Conclusioni

Lo scopo del capitolato<sub>G</sub> non è risultato molto stimolante, in quanto lo sviluppo di alcune componenti sembra caratterizzato da attività ripetitive. Inoltre, il dover apprendere tecnologie per le quali è richiesta solamente l'integrazione di un sottoinsieme di funzionalità, ha demotivato il gruppo nella scelta di questo progetto.

## 3.2 Capitolato C2 - Colletta

### 3.2.1 Informazioni Generali

- **Nome:** Colletta: piattaforma raccolta dati di analisi di testo;
- **Proponente:** Mivoq S.r.l.;
- **Commitente:** Tullio Vardanega, Riccardo Cardin.

### 3.2.2 Descrizione

L'obiettivo del progetto Colletta è la creazione di una piattaforma collaborativa di raccolta dati in cui gli utenti possono svolgere esercizi grammaticali in diverse lingue. Tali dati devono essere resi disponibili e facilmente consultabili per gli sviluppatori e ricercatori, i quali hanno il fine di insegnare ad un elaboratore a svolgere i medesimi esercizi attraverso tecniche di autoapprendimento.

### 3.2.3 Finalità del progetto

Il risultato finale sarà un'applicazione con tre attori principali ai quali verranno fornite funzionalità diverse:

- **Insegnanti:** Dovranno poter inserire gli esercizi in modo rapido e intuitivo. Gli esercizi inseriti verranno risolti in modo automatico dall'applicazione, fornendo un risultato immediato all'insegnante, il quale dovrà poi validare e modificare, se necessario, le risposte;
- **Allievi:** Dovranno poter eseguire il test in modo pratico e ricevere subito una valutazione. I test possono essere scelti tra quelli già presenti oppure creati al momento (in questo caso verrà utilizzata la soluzione automatica per valutare). Ogni studente verrà ricompensato con un sistema a punti per aver svolto un esercizio e potrà visualizzare lo storico degli esercizi svolti;
- **Sviluppatori:** Dovranno poter consultare i dati prodotti al fine di utilizzarli per l'apprendimento automatico. Allo sviluppatore dovranno essere fornite più di una versione delle risposte fornite, con lo storico delle modifiche effettuate.

### 3.2.4 Tecnologie interessate

- **Hunpos e Freeling:** Entrambi software per il Part of Speech (PoS) tagging, cioè l'etichettatura delle varie parti di una frase;
- **Firebase Storage:** Suggerito per l'immagazzinamento dei dati prodotti dall'applicazione;
- **Web/Mobile programming:** Da utilizzare in modo esclusivo per la realizzazione dell'applicazione. Nessuna tecnologia specifica preposta dall'azienda sotto questo punto di vista.

### 3.2.5 Aspetti positivi

- Nessun vincolo da parte dell'azienda sulle tecnologie da utilizzare, quindi grande libertà;
- Requisiti obbligatori in numero molto ridotto, si ha così una maggiore flessibilità;

- Utilizzo di Firebase, sistema utilizzato da molte aziende in ambito professionale;
- Implementazione dell'applicazione su Web oppure Mobile, campi non trattati (se non in minima parte) nel nostro corso di studi.

#### **3.2.6 Criticità e fattori di rischio**

- Le troppe tecnologie nuove da utilizzare potrebbero portare ad un pesante lavoro di studio di esse prima di poterle utilizzare;
- La non conoscenza della grammatica delle varie lingue potrebbe portare ad una elevata difficoltà implementativa della soluzione.

#### **3.2.7 Conclusioni**

Questo capitolato è stato trovato molto interessante da tutto il gruppo ma al momento della valutazione gli slots disponibili erano già esauriti.

### 3.3 Capitolato C3 - G&B

#### 3.3.1 Informazioni generali

- **Nome:** G&B: monitoraggio intelligente di processi DevOps<sub>G</sub>;
- **Proponente:** Zucchetti;
- **Committente:** Prof. Tullio Vardanega e Prof. Riccardo Cardin.

#### 3.3.2 Descrizione

Il capitolato<sub>G</sub> prevede la realizzazione di un plug-in<sub>G</sub> per monitorare, tramite l'utilizzo di Grafana<sub>G</sub>, un sistema DevOps<sub>G</sub>, cioè un sistema in cui chi produce il software e chi lo usa collaborano strettamente. Perché la collaborazione sia efficace è necessario che si applichi, a tale sistema di monitoraggio, reti Bayesiane al flusso dei dati ricevuti per allarmi o segnalazioni tra gli operatori del servizio Cloud e la linea di produzione del software il tutto visualizzato tramite grafici che permetteranno di analizzare e controllare tali notifiche.

#### 3.3.3 Finalità del progetto

La struttura del plug-in<sub>G</sub> verrà scritta in linguaggio JavaScript che leggerà da un file JSON<sub>G</sub> la definizione della rete Bayesiana e permetterà di associare dei nodi della rete, con informazioni di probabilità, ad un flusso di dati presente in nel sistema di monitoraggio. La rete riceverà il flusso, ad intervalli predefiniti o con continuità, e verranno eseguiti dei calcoli modificando così le probabilità dei nodi. Sia il flusso di dati che la rete verranno monitorati tramite un'apposita dashboard<sub>G</sub> visualizzando il tutto attraverso dei grafici. Opzionalmente il capitolato consiglia la possibilità di un'eventuale generazione di allarmi/notifiche che valutano l'andamento dei dati visualizzati in quel momento.

#### 3.3.4 Tecnologie interessate

- **Grafana:** software Open Source<sub>G</sub> per il monitoraggio di sistemi che, ricevuti dati, consente di raccogliarli in grafici che si possono visualizzare, analizzare, misurare e controllare;
- **JavaScript:** linguaggio di programmazione richiesto per costruire il plug-in<sub>G</sub> di Grafana e per definire la rete di Bayes<sub>G</sub> in formato JSON<sub>G</sub>;
- **Rete di Bayes:** rete di nodi che contengono informazioni di probabilità. Quando si verifica un evento significativo le probabilità dei nodi si aggiornano di conseguenza.

#### 3.3.5 Aspetti positivi

- Il proponente si presenta come la prima software house italiana e quindi crea interesse nel gruppo;
- Il documento fornito per la spiegazione del capitolato è chiaro e i requisiti sono ben definiti.

#### 3.3.6 Criticità e fattori di rischio

- Non vi sono molte tecnologie da apprendere se non l'utilizzo del sistema di monitoraggio tramite Grafana;
- Il solo apprendimento di quest'ultimo software non ha scusitato motivo di interesse al gruppo di lavoro.

### **3.3.7 Conclusioni**

Il capitolato presenta dei punti di sviluppo molto interessanti, tuttavia i posti disponibili per l'appalto sono stati esauriti dal primo lotto.

## 3.4 Capitolato C4 - MegAlexa

### 3.4.1 Informazioni generali

- **Nome:** MegAlexa, arricchitore di skill di Amazon Alexa;
- **Proponente:** ZERO12;
- **Committente:** Prof. Tullio Vardanega e Prof. Riccardo Cardin.

### 3.4.2 Descrizione

La sfida lanciata dall'azienda proponente consiste di progettare una skill di Alexa, l'assistente virtuale prodotto da Amazon, in cui gli utenti tramite un applicativo Web o Mobile (Android o iOS) siano in grado di avviare Workflow.

### 3.4.3 Finalità del progetto

Far sì che un qualsiasi utente che possiede Amazon Alexa possa, attraverso l'applicativo realizzato tramite micro-funzioni già fornite, crearsi una routine con le informazioni che vuole tramite un comando vocale personalizzato.

### 3.4.4 Tecnologie interessate

- **Amazon Alexa:** l'assistente digitale di Amazon;
- **Lambda (AWS):** servizio di elaborazione serverless per l'esecuzione del proprio codice;
- **API Gateway (AWS):** servizio API per la comunicazione con Lambda;
- **Aurora Serverless (AWS):** offre capacità di database;
- **Node.js:** piattaforma per esecuzione di codice JavaScript;
- **HTML5, CSS3 e JavaScript:** linguaggi da utilizzare per l'implementazione dell'interfaccia web;
- **Bootstrap:** framework front end, consigliato dal proponente;
- **Android e iOS:** linguaggi come *Kotlin* (Android) o *Swift* (iOS) sono consigliati dal proponente.

### 3.4.5 Aspetti positivi

- Il proponente offre delle lezioni al fine di introdurre il gruppo alle nuove tecnologie da utilizzare nello sviluppo del progetto e lasciando poi piena libertà di sviluppo dei servizi Google per la realizzazione dell'assistente virtuale;
- Grande presenza nel web di documentazione dettagliata che rende più semplice l'apprendimento di tali tecnologie. In particolare, Amazon fornisce Alexa Skills Kit (raccolta di API, strumenti, documentazioni ed esempi di codice).

#### 3.4.6 Criticità e fattori di rischio

- È obbligatorio che le shortcuts<sub>G</sub> siano multilingua. Echo al momento supporta le lingue: inglese, francese, tedesco, italiano, giapponese e spagnolo. Tuttavia, possiamo realizzare in modo esaustivo solamente la versione italiana ed inglese, viste le nostre limitate conoscenze linguistiche;
- Sono già presenti, nel web, tecnologie per la realizzazione di skills<sub>G</sub> in grado di avviare dei workflow<sub>G</sub> personalizzati, anche se in modo piuttosto grezzo. Infatti, la stessa applicazione di Alexa permette di creare sequenze di azioni precedentemente selezionate.

#### 3.4.7 Conclusioni

Nonostante tale capitolato<sub>G</sub> sia interessante dal punto di vista delle nuove tecnologie che stanno prendendo piede in questo momento e per le competenze curricolari che potrebbe comportare, il gruppo si è mostrato più stimolato verso un altro progetto non meno allettante.

## 3.5 Capitolato C6 - Soldino

### 3.5.1 Informazioni generali

- **Nome:** Soldino: piattaforma Ethereum per pagamenti IVA;
- **Proponente:** Red Babel;
- **Committente:** Prof. Tullio Vardanega e Prof. Riccardo Cardin.

### 3.5.2 Descrizione

Il capitolato C6 richiede di sviluppare un sistema, gestito dal Governo, volto alla gestione dell'IVA tramite la blockchain Ethereum. I proprietari di partita IVA registrati potranno acquistare/vendere beni e servizi. Il Governo è in grado di coniare e distribuire la moneta utilizzata nelle transazioni. I cittadini potranno fare acquisti tramite la moneta coniata dal governo.

### 3.5.3 Finalità del progetto

Lo scopo ultimo di *Soldino* è quello di fornire, tramite un sito web, un insieme di DApps che lavorano su EVM (Ethereum Virtual Machine). Il Governo e le aziende possono eseguire le solite azioni di contabilizzazione legate all'IVA (gestione pagamenti, tassi di cambio...).

### 3.5.4 Tecnologie interessate

- **Ethereum:** blockchain che serve per approvare le transazioni effettuate sulla piattaforma e ad archivarle su un sistema distribuito.
- **DApps:** applicazione decentralizzata che utilizza la blockchain di Ethereum, è composta da più parti possibilmente separate ed ogni sua parte è in grado di eseguire il proprio lavoro indipendentemente;
- **Ethereum Virtual Machine (EVM):** macchina virtuale che permette di verificare ed eseguire il codice sulla blockchain assicurando che venga eseguito nello stesso modo su qualsiasi macchina;
- **Smart Contracts:** dove risiede il codice vero e proprio utilizzato dalle DApp;
- **Solidity:** linguaggio che permette la scrittura di Smart Contracts su EVM;
- **MetaMask:** add-on del browser che permette la gestione dei propri account su rete Ethereum. Serve inoltre a verificare l'identità degli utenti e validare le transazioni;
- **Web3:** API utilizzata per effettuare chiamate ad un nodo remoto di Ethereum;
- **Ropsten:** rete di test che utilizza lo stesso insieme di protocolli di Ethereum, utile a testare le DApp;
- **Truffle:** ambiente di sviluppo che permette la scrittura di Smart Contracts e implementa automaticamente i relativi test;
- **ESlint:** utilizzato per l'analisi sintattica del codice, utilizzato soprattutto per trovare pattern problematici o codice che non aderisce ad una linea guida;
- **JavaScript, HTML, Redux, SCSS, React:** insieme di framework e linguaggi utilizzati per creare il front end.



### 3.5.5 Aspetti positivi

- l'impiego di tecnologie quali React, Redux e SCSS permetterebbe al gruppo di acquisire conoscenze molto utili soprattutto in un futuro ambito lavorativo;
- l'idea di base ha piacevolmente colpito il gruppo, un eventuale riutilizzo del valore aggiunto, tramite una blockchain, ci è sembrata un'idea allettante;
- il gruppo era inoltre molto interessato anche al solo trattamento della blockchain e di ciò che ne fa parte (criptovaluta, EVM...) senza l'aspetto riguardante il trattamento dell'IVA.

### 3.5.6 Criticità e fattori di rischio

- l'impiego di un consistente numero di nuove tecnologie prevede un carico di studio non indifferente, vista anche la scarsità di documentazione presente sul web;
- la distanza fisica della sede di *Red Babel* potrebbe influire in modo negativo sulla comunicazione tra gruppo e proponente/riferente;
- a differenza dello scorso anno, l'interesse verso le criptovalute è molto diminuito e di conseguenza anche il gruppo ha deciso di spostarsi verso una realtà più solida.

### 3.5.7 Conclusioni

Nonostante ci sia stato un forte interesse iniziale verso il capitolato, in quanto utilizzava nuove tecnologie molto interessanti, a seguito di una analisi più oggettiva riguardante appunto queste ultime, si è scelto di spostarsi verso una realtà più concreta e che si avvicini ai nostri interessi più che alla nostra curiosità.