



S O L U T I O N S

8Lab Solutions - Progetto "Soldino"

Studio di fattibilità

Versione	0.2.0
Approvazione	
Redazione	
Verifica	
Stato	
Uso	Interno
Destinato a	8Lab Solutions Prof. Tullio Vardanega Prof. Riccardo Cardin

Descrizione

Studio di fattibilità dei capitolati proposti

Indice

1	Changelog	2
2	Capitolato scelto C6 - Soldino	3
2.1	Informazioni generali	3
2.2	Descrizione	3
2.3	Obiettivo finale	3
2.4	Studio del dominio	3
2.4.1	Dominio applicativo	3
2.4.2	Dominio tecnologico	3
2.5	Aspetti positivi	4
2.6	Criticità e fattori di rischio	4
2.7	Conclusioni	4
3	Valutazioni sugli altri capitolati	5
3.1	Capitolato C1 - Butterfly	5
3.1.1	Informazioni generali	5
3.1.2	Descrizione	5
3.1.3	Studio del dominio	5
3.1.4	Aspetti positivi	5
3.1.5	Criticità e fattori di rischio	6
3.1.6	Conclusioni	6
3.2	Capitolato C2 - Colletta	7
3.2.1	Informazioni sul capitolato	7
3.2.2	Descrizione	7
3.2.3	Obiettivo finale	7
3.2.4	Studio del dominio	7
3.2.5	Aspetti positivi	8
3.2.6	Criticità e fattori di rischio	8
3.2.7	Conclusioni	8
3.3	Capitolato C3 - G&B	9
3.3.1	Informazioni sul capitolato	9
3.3.2	Descrizione	9
3.3.3	Studio del dominio	9
3.3.4	Aspetti positivi	9
3.3.5	Criticità e fattori di rischio	9
3.3.6	Conclusioni	9
3.4	Capitolato C4 - MegAlexa	10
3.4.1	Informazioni sul capitolato	10
3.4.2	Descrizione	10
3.4.3	Studio del dominio	10
3.4.4	Aspetti positivi	10
3.4.5	Criticità e fattori di Rischio	10
3.4.6	Conclusioni	11
3.5	Capitolato C5 - P2PCS	12
3.5.1	Descrizione	12
3.5.2	Obiettivo Finale	12
3.5.3	Dominio tecnologico	12
3.5.4	Aspetti Positivi	12
3.5.5	Criticità e fattori di rischio	12
3.5.6	Conclusioni	12

1 Changelog

Versione	Data	Nominativo	Ruolo	Descrizione
0.2.0	06/12/2018	Francesco Donè	<i>Revisore</i>	Revisione studio di fattibilità dei capitoli C3, C4 e C5.
0.1.0	05/12/2018	Sara Feltrin	<i>Revisore</i>	Revisione studio di fattibilità dei capitoli C1 e C2.
0.0.6	05/12/2018	Mattia Bolzonella	<i>Analista</i>	Completato studio del dominio tecnologico del capitolo C6
0.0.5	04/12/2018	Mattia Bolzonella	<i>Analista</i>	Modificati titoli delle sottosezioni 2.4 e 2.5.
0.0.4	03/12/2018	Sara Feltrin, Mattia Bolzonella	<i>Revisore, Analista</i>	Corretta struttura del documento.
0.0.3	02/12/2018	Sara Feltrin, Mattia Bolzonella	<i>Revisore, Analista</i>	Completata stesura studio fattibilità del capitolo C1. Sistemate sezioni del documento.
0.0.2	26/11/2018	Mattia Bolzonella, Sara Feltrin	<i>Analista, Revisore</i>	Cominciata stesura analisi capitolo C1.
0.0.1	25/11/2018	Mattia Bolzonella	<i>Analista</i>	Creato documento latex e creata pagina del titolo.

Tabella 1: Changelog di questo documento

2 Capitolato scelto C6 - Soldino

2.1 Informazioni generali

- **Nome:** Soldino
- **Proponente:** *Red Babel*
- **Committente:** Prof. Tullio Vardanega e Prof. Riccardo Cardin

2.2 Descrizione

Attualmente il sistema di pagamento dell'IVA prevede che siano le aziende a registrare i loro acquisti/vendite, e che con cadenza trimestrale venga calcolato il saldo. Se l'azienda risulta debitrice allora deve versare allo stato il rispettivo ammontare, altrimenti ottiene un credito IVA da poter utilizzare successivamente.

Soldino nasce per automatizzare la gestione di questo processo, proponendo una piattaforma di e-commerce controllata dal governo, nella quale le aziende e i cittadini possano comprare e/o vendere beni e servizi per mezzo di una criptovaluta.

2.3 Obiettivo finale

L'obiettivo di Soldino è la creazione di un insieme di $\mathbb{D}Apps_G$ per la gestione di un e-commerce. Governo ed aziende verranno assistite da Soldino nella gestione dell'IVA sottostante le operazioni di compravendita di beni e servizi tra aziende e clienti. La business logic deve affidarsi al meccanismo degli Smart Contracts $_G$, che arricchisce le transazioni con regole user-defined, garantendone maggiore controllo e sicurezza. I diversi attori potranno accedere alla piattaforma attraverso un'interfaccia web dedicata, previa autenticazione con Metamask $_G$ (disponibile come add-on per i browser come Chrome $_G$, Opera $_G$, Firefox $_G$ e dal browser Brave $_G$). Il mezzo di pagamento sarà un criptovaluta basata sullo standard ECR20 $_G$, denominata "Cubit" $_G$.

2.4 Studio del dominio

2.4.1 Dominio applicativo

Si individuano tre attori:

- **Ente governativo:** È in grado di coniare e distribuire la criptovaluta utilizzata in Soldino. Potrà inoltre accedere e gestire la lista delle aziende iscritte alla piattaforma, e controllare che a cadenza trimestrale esse abbiano pagato in caso di stato di debito.
- **Proprietario di un'azienda:** Deve poter registrare la propria impresa nell'e-commerce, gestire i prodotti/servizi offerti ed acquistare prodotti da altre aziende. Inoltre deve poter gestire tutto ciò che riguarda l'IVA, ovvero pagare l'eventuale saldo a cadenza trimestrale (in caso di debito), creare un documento di resoconto di periodo e gestire le ricevute IVA.
- **Persona fisica $_G$:** Può convertire Euro in Cubit e successivamente acquistare i beni e servizi offerti sulla piattaforma.

2.4.2 Dominio tecnologico

Per lo sviluppo del lato backend $_G$ si individuano le seguenti tecnologie:

- **Blockchain:** è un database distribuito e decentralizzato che utilizza una rete peer to peer.
- **Ethereum:** è un'infrastruttura decentralizzata open source di computing che esegue Smart Contracts $_G$. Utilizza la blockchain $_G$ per sincronizzare e salvare i cambiamenti di stato del sistema.
- **Ethereum Virtual Machine (EVM):** è la macchina virtuale che esegue smart contracts.
- **Solidity:** linguaggio orientato ai contratti. È utilizzato per implementare smart contracts.
- **$\mathbb{D}Apps$:** è una web application che si interfaccia con uno o più smart contract;
- **Truffle:** framework per lo sviluppo di smart contract.
- **Metamask:** plugin browser che permette di autenticarsi alla rete Ethereum $_G$.

Per lo sviluppo del lato front-end:

- **Javascript:** linguaggio di scripting client-side. Il proponente impone di sottostare allo standard esposto nella "Airbnb Javascript style guide", e di utilizzare un tool di analisi statica del codice e syntax checking, "ESlint";
- **React:** libreria open source JavaScript per la creazione di interfacce grafiche e la gestione delle interazioni in ambito web
- **Redux:** libreria open source Javascript per la gestione degli stati di React;
- **SCSS:** estensione del linguaggio CSS, che ne aumenta le funzionalità ed espressività.

2.5 Aspetti positivi

2.6 Criticità e fattori di rischio

2.7 Conclusioni

Il gruppo ha accolto con entusiasmo il capitolato, in quanto tra tutte le proposte è quella che copre più tecnologie innovative, ma che sono già ampiamente sfruttate e che probabilmente avranno un impatto sempre maggiore nel mercato. Con questo progetto avremo l'opportunità di studiare un campo dell'informatica che all'università è poco trattato, potendo aggiungere al nostro bagaglio curricolare delle voci particolarmente interessanti. Inoltre il gruppo ha apprezzato che RedBabel abbia proposto un approccio moderno allo sviluppo delle interfacce web, scegliendo tecnologie quali React e Redux, sempre più diffuse e richieste anche in ambito lavorativo.

L'interesse verso queste nuove tecnologie ha spinto il gruppo 8labsolutions ad optare per questa scelta nonostante la consapevolezza che, essendo tecnologie a noi poco conosciute, il loro studio richiederà un impegno notevole. Siamo inoltre consci del fatto che la documentazione/risorse disponibili potranno essere inferiori a confronto con tecnologie già consolidate.

3 Valutazioni sugli altri capitolati

3.1 Capitolato 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

Butterfly mira allo sviluppo di una piattaforma che permette di accentrare, standardizzare, automatizzare e personalizzare le segnalazioni di diversi strumenti di versionamento, di continuous integration e continuous delivery, così da permettere all'utente di interfacciarsi ad un'unica dashboard.

3.1.3 Studio del dominio

3.1.3.1 Dominio applicativo

Il prodotto finale, integrando al suo interno le segnalazioni delle diverse applicazioni, semplifica e organizza il lavoro dell'utente. L'azienda propone, per la realizzazione di questa soluzione, l'utilizzo di quattro componenti:

- **Producers**, che hanno la funzionalità di recuperare le segnalazioni e mostrarle come messaggi nei rispettivi Topic;
- **Broker**, come strumento per istanziare e gestire i Topic;
- **Consumers**, che hanno il compito di iscriversi a diversi Topic specifici così da reindirizzare i messaggi verso gli utenti finali;
- **Componente custom specifico**, inteso come un componente da implementare per l'azienda che permetta di indirizzare la notifica alla persona più idonea.

3.1.3.2 Dominio tecnologico

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

- **Java**
- **Python**
- **Nodejs**
- **Apache Kafka:** è una piattaforma stream processing di software consigliato per lo sviluppo del componente broker.

Mentre i requisiti obbligatori sono:

- rispettare i 12 fattori presenti in "The Twelve-Factor App" nelle applicazioni sviluppate;
- utilizzare Docker come container per l'istanziamento dei componenti;
- esporre le API Rest dei componenti per l'utilizzo dell'applicazione;
- utilizzare test unitari e d'integrazione per ogni componente realizzato.

3.1.4 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 materiale di studio per cui il capitolato offre la possibilità di migliorare la padronanza di questo linguaggio.

3.1.5 Criticità e fattori di rischio

- L'apprendimento delle tecnologie coinvolte nel lato Producer coprirebbe solo aspetti marginali (le segnalazioni) delle suddette, senza conferire capacità estese;
- Il lavoro per la raccolta dati appare ripetitivo e le API da imparare ad usare sono molto specifiche, quindi sarebbero circoscritte al progetto e non più utilizzate in futuro_G.

3.1.6 Conclusioni

Lo scopo del capitolato non è risultato molto interessante, in quanto lo sviluppo di alcune componenti sembra caratterizzato da attività ripetitive. Inoltre il dover apprendere tecnologie per le quali è solamente richiesto l'integrazione di un sottoinsieme di funzionalità non ha suscitato un forte interesse.

3.2 Capitolato C2 - Colletta

3.2.1 Informazioni sul capitolato

- **Nome:** Colletta: piattaforma per raccolta dati mediante esercizi di grammatica
- **Proponente:** Mivoq s.r.l.
- **Committente:** Prof. Tullio Vardanega e Prof. Riccardo Cardin

3.2.2 Descrizione

Il progetto consiste nella realizzazione di una piattaforma collaborativa di raccolta dati in cui gli utenti possano predisporre e/o svolgere piccoli esercizi di grammatica. Verranno osservati sia le prove proposte dagli insegnanti, sia i loro svolgimenti da parte degli allievi.

3.2.3 Obiettivo finale

Lo scopo del progetto è la raccolta di dati relativi alla classificazione grammaticale di parole nel contesto in cui vengono utilizzate e la possibilità di rendere facilmente disponibili ed esportabili tali informazioni. La raccolta dati non deve avvenire in modo esplicito, piuttosto gli utenti devono trovare un'utilità intrinseca nell'utilizzo della piattaforma. A tal fine il proponente suggerisce l'implementazione di un sistema predisposto alla gestione di esercizi di grammatica, come l'analisi grammaticale.

3.2.4 Studio del dominio

3.2.4.1 Studio del dominio applicativo Vengono delineati tre principali attori: insegnanti, allievi e sviluppatori.

- **Insegnanti:** L'insegnante dovrà poter creare esercizi di analisi grammaticale in modo agevole. Dopo l'inserimento di nuove frasi nel sistema, un tool integrato provvederà automaticamente allo svolgimento dell'esercizio, proponendo una soluzione. L'insegnante dovrà successivamente correggere e/o validare il risultato proposto, al fine di garantire che i propri allievi ricevano del materiale controllato.
- **Allievi:** L'allievo che accede al sistema dovrà poter svolgere gli esercizi proposti dall'insegnante e ricevere una valutazione immediata. Gli verrà data la possibilità di esprimere una preferenza sui propri insegnanti in modo da privilegiare le versioni di un insegnante rispetto ad un altro. La scelta dell'esercizio da svolgere avverrà tramite un elenco di frasi proposte o inserendo autonomamente una frase nel sistema. Verrà fornita la possibilità di selezionare l'insegnante per la valutazione tra quelli che hanno predisposto quel determinato esercizio. Nel caso non ce ne fossero, il sistema automatico provvederà alla correzione con valutazione. E' previsto anche uno storico dei progressi nel tempo e un sistema di ricompensa.
- **Sviluppatori:** Gli sviluppatori sono interessati prevalentemente ad accedere ai dati raccolti dagli utenti al fine di utilizzarli nella fase di addestramento di sistemi di apprendimento automatico. Allo sviluppatore dovrà essere fornita più di una versione dell'annotazione di ogni frase, in modo tale da dedurre quale sia quella più corretta. E' dunque importante che venga loro fornito uno storico dei dati per poter estrarre solo i dati d'interesse ed escludere le correzioni di alcuni utenti.

3.2.4.2 Studio del dominio tecnologico

- **Hunpos/Freeling:** Hunpos [Mivoq(2014-2018)] e FreeLing [TALP Research Center, UPC(2008-2018a)] sono due software specializzati nel Part of Speech (PoS) tagging, che consiste nell'interpretare un testo etichettando ciascuna parola con il relativo significato grammaticale. I due software sfruttano delle tecniche di apprendimento automatico supervisionato per tale classificazione;
- **Firebase:** è una piattaforma per sviluppatori web e mobile offerta da Google tra i quali è presente Firebase(FB) Storage. Quest'ultimo è un servizio per la gestione dei dati che consente in particolare upload e download sicuri anche con una connessione di scarsa qualità. Può salvare immagini, video, audio ed ogni contenuto generato dall'utente. I dati relativi a ciascun individuo sono successivamente sfruttati da altri servizi (e.g. FB Analytics) per offrire una versione personalizzata dell'applicazione;
- **Web/Mobile programming:** il proponente richiede che la piattaforma sia sviluppata sotto forma di pagina web oppure come applicazione mobile. L'azienda non ha imposto l'adozione di nessuna tecnologia specifica per quanto riguarda questa parte del progetto, quindi la scelta spetta agli sviluppatori.

3.2.5 Aspetti positivi

- Il proponente non ha specificato nessuna tecnologia con la quale sviluppare la piattaforma. Quindi viene lasciata agli sviluppatori totale libertà di scelta;
- Nel capitolato i requisiti obbligatori, sia espliciti che impliciti, sono in numero inferiore rispetto agli opzionali. Ciò rende maggiormente flessibile la quantità di requisiti da gestire;
- La piattaforma Google FireBase potrebbe risultare una conoscenza utile da applicare successivamente nel mondo del lavoro;
- L'azienda rimane aperta a proposte differenti dall'analisi grammaticale, purché venga mantenuto l'obiettivo finale.

3.2.6 Criticità e fattori di rischio

- Uno dei requisiti opzionali di maggior interesse da parte del proponente consiste nel multilinguismo della piattaforma. Il tempo necessario da dedicare all'analisi grammaticale di lingue straniere è complesso da quantificare;
- Nel progetto vengono applicati temi già ampiamente studiati nel corso di studi universitario, che non andrebbero ad aumentare il bagaglio di tecnologie conosciute.

3.2.7 Conclusioni

Il gruppo ha apprezzato questa proposta tanto che il capitolato si è piazzato al terzo posto nella classifica delle preferenze. Tuttavia il team ha deciso di orientarsi verso i capitolati rivolti a nuove tecnologie, considerate più interessanti e che potranno arricchire maggiormente il prospetto curricolare.

3.3 Capitolato C3 - G&B

3.3.1 Informazioni sul capitolato

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

3.3.2 Descrizione

Il capitolato prevede la costruzione di un software per monitorare un sistema DevOps, cioè un sistema in cui, a livello aziendale, chi produce il software e chi lo usa collaborano strettamente. Per migliorare ulteriormente il servizio erogato si richiede un secondo software che supporti il primo visualizzando, analizzando, misurando e controllando i dati forniti dal primo.

3.3.3 Studio del dominio

3.3.3.1 Dominio applicativo

La struttura del software da realizzare è conforme ai seguenti punti:

- Un flusso di dati in input viene associato a una rete Bayesiana: questa rete è composta di nodi contenenti informazioni di probabilità;
- La rete riceve il flusso e lo usa per fare dei calcoli, aggiornando quindi le probabilità dei propri nodi;
- Sia il flusso dati che la rete sono monitorati in un cruscotto;
- L'andamento dei dati determina l'eventuale generazione di allarmi e notifiche.

3.3.3.2 Dominio tecnologico

Le tecnologie proposte per lo sviluppo del progetto sono:

- **Grafana:** software *open-source* che, ricevuti i dati in input, consente di raccogliarli in un cruscotto, visualizzarli, analizzarli, misurarli e controllarli;
- **InfluxDB:** database di tipo *Time Series*, generati con continuità temporale e atti a essere letti e monitorati costantemente per misurarne le variazioni;
- **JavaScript:** Linguaggio di programmazione richiesto per costruire i *plug-in* di Grafana e per definire la rete di Bayes in formato .json;
- **Rete di Bayes:** rete di nodi che contengono informazioni di probabilità; quando un evento significativo si verifica, le probabilità dei nodi si aggiornano conseguentemente.

3.3.4 Aspetti positivi

- L'azienda è grande e possiede esperienza utile all'apprendimento di tutti i componenti del gruppo;
- Il dominio del problema è chiaro e circoscritto;
- I requisiti sono ben manifesti e comprensibili nel capitolato;
- Le tecnologie riguardano vari ambiti (database, linguaggi, probabilità, monitoraggio) e sono in numero ragionevole da apprendere.

3.3.5 Criticità e fattori di rischio

- il capitolato è fortemente conteso tra i gruppi appaltatori: la probabilità di aggiudicarselo è inferiore a quella desiderata.

3.3.6 Conclusioni

Il capitolato è stato escluso dalle preferenze. Ci sono dei fattori positivi riguardo questo capitolato: in particolare, la definizione chiara del dominio e dei requisiti del software. Tuttavia, questi non sono sufficienti. Si ritiene che l'insieme delle tecnologie non sia il migliore disponibile: alcune di esse (es. Reti di Bayes) sono circoscritte, altre (Grafana) sono di dubbia diffusione. Inoltre, il capitolato è fortemente conteso tra i gruppi.

3.4 Capitolato C4 - MegAlexa

3.4.1 Informazioni sul capitolato

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

3.4.2 Descrizione

Creare una skill per Alexa di Amazon in grado di avviare dei workflow creati dagli utenti tramite interfaccia web o mobile app per iOS e Android.

3.4.3 Studio del dominio

3.4.3.1 Dominio applicativo

L'obiettivo è di realizzare un applicativo che, forniti dei connettori per diversi applicativi, permetta all'utente di costruirsi le proprie routine. Di conseguenza, un utente registrato avrà a disposizione dei connettori (detti anche micro-funzioni) che potrà inserire all'interno di un workflow eseguito poi tramite controllo vocale.

3.4.3.2 Dominio tecnologico

- **Alexa:** Assistente vocale Amazon basato sul cloud e l'intelligenza che alimenta Amazon Echo;
- **Lambda (AWS):** Servizio di elaborazione serverless che esegue del codice in risposta a determinati eventi;
- **API gateway (AWS):** Servizio API per la comunicazione con Lambda;
- **Aurora Serverless (AWS):** offre capacità di database senza dover allocare e gestire server;
- **NodeJS:** piattaforma event-driven per esecuzione di codice JavaScript server-side;
- **Bootstrap:** Framework specifico per sviluppi di tipo front-end più utilizzato (solamente consigliato).
- **HTML5**
- **CSS3**
- **Javascript**

3.4.4 Aspetti positivi

- Il proponente offre delle lezioni al fine di introdurre al gruppo le nuove tecnologie da utilizzare nello sviluppo del progetto e dirigere lo studio autonomo;
- La massiccia presenza nel web di documentazione dettagliata, esempi e strumenti rende relativamente semplice l'apprendimento di tali tecnologie, in particolare Amazon fornisce Alexa Skills Kit (raccolta di API self-service, strumenti, documentazioni, esempi);
- Amazon ed il mercato in generale sembrano per ora molto interessati agli Assistenti vocali, quindi la conoscenza di tali tecnologie può essere una nota rilevante a livello curriculare.

3.4.5 Criticità e fattori di Rischio

- È obbligatorio che le shortcut siano multilingua. Echo al momento supporta: Inglese, Francese Tedesco, Italiano, Giapponese, Spagnolo; tuttavia le nostre conoscenze in ambito linguistico ci permettono di realizzare in modo esaustivo solamente le versioni italiana e inglese;
- Sono già presenti nel web tecnologie per realizzare, anche se in modo piuttosto grezzo ciò che viene richiesto dal capitolato; la stessa applicazione di Alexa permette di creare sequenze di azioni precedentemente selezionate;
- Il proponente offre l'opportunità a solamente due gruppi di aggiudicarsi il capitolato.

3.4.6 Conclusioni

Nonostante tale capitolato abbia destato particolare interesse all'interno del gruppo, sia a livello tecnologico che di competenze curriculari, l'esiguo numero di posti disponibili e la presenza di capitolati non meno interessanti, ha comportato lo spostamento in secondo piano del suddetto progetto.

3.5 Capitolato C5 - P2PCS

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

3.5.1 Descrizione

Creare un'applicazione per car sharing condominiale che permetta ad utenti che posseggono un'auto di prestarla a vicini che ne fanno richiesta, permettendo così di evitare che il veicolo resti un peso economico per chi la possiede ma non la utilizza spesso.

3.5.2 Studio del dominio

3.5.2.1 Dominio applicativo

Un utente registrato potrà cercare un'auto libera nella zona interessata, prenotarla per quando ne avrà bisogno e andare a ritirare le chiavi. I proprietari invece potranno offrire la propria macchina nei giorni in cui segnaleranno che non è inutilizzata.

3.5.2.2 Dominio tecnologico

- **JavaScript:** per la stesura del codice;
- **Node.js:** framework impiegato per la scrittura di applicazioni JavaScript dal lato server con un modello asincrono di I/O basato su eventi, permettendo un'ottimizzazione di tempi e risorse;
- **Google Cloud:** per la gestione del database;
- **Octalysis:** framework per ottimizzare il sistema più verso la motivazione delle persone che verso la pura efficienza;
- **Henshin:** nello specifico si parla di Movens ovvero una piattaforma software open source, specifica nell'impiego di mobilità, gestione dell'IoT e smart cities. Tra servizi principali offerti ci sono la completa gestione della condivisione dei veicoli, la gestione dell'assicurazione e la gestione della connessione peer-to-peer;
- **Android:** per lo sviluppo di un'app.

3.5.3 Aspetti Positivi

- Possibilità di imparare linguaggi e tecnologie molto utilizzati e richiesti;
- Possibilità di capire come funziona uno standup di una metodologia Agile all'interno di un'azienda;
- Possibilità di comprendere la teoria del Gamification;
- L'azienda fornirebbe parte di ciò che è richiesto per lo sviluppo dell'applicazione.

3.5.4 Criticità e fattori di rischio

- Recentemente simili progetti italiani per car sharing si sono rivelati fallimentari raccogliendo uno scarso numero di utenti.

3.5.5 Conclusioni

Il gruppo ha espresso un giudizio principalmente negativo su questo capitolato soprattutto considerando i fallimenti dei altre compagnie in questo campo.