



Progetto: *Nome Progetto*

[codebusterswe@gmail.com](mailto:codebusterswe@gmail.com)

## Studio Di Fattibilità

### Informazioni sul documento

<b>Versione</b>	1.0.0
<b>Approvatori</b>	Sassaro Giacomo
<b>Redattori</b>	Scialpi Paolo Zenere Marco Rago Alessandro Safdari Houssaine
<b>Verificatori</b>	Baldisseri Michele Pirolo Alessandro
<b>Uso</b>	Interno
<b>Distribuzione</b>	Prof. Vardanega Tullio Prof. Cardin Riccardo Gruppo <i>CodeBusters</i>

### Descrizione

Questo documento si occupa di descrivere l'analisi dei capitolati d'appalto realizzata dal gruppo al fine di valutarne la fattibilità

## Registro delle modifiche

Versione	Data	Nominativo	Ruolo	Descrizione
0.1.2	01-12-2020	Zenere Marco	Analista	Correzione studio di fattibilità C4
0.1.1	30-11-2020	Zenere Marco	Analista	Redazione studio di fattibilità C4
0.1.0	26-11-2020	Sassaro Giacomo, Baldisseri Michele	Verificatore	Creazione documento Latex e adattamento bozza Latex
0.0.4	19-11-2020	Zenere Marco, Scialpi Paolo	Analista	Redazione studio di fattibilità C2
0.0.3	19-11-2020	Zenere Marco, Scialpi Paolo	Analista	Redazione studio di fattibilità C1
0.0.2	18-11-2020	Pirolo Alessandro	Analista	Redazione studio di fattibilità C3
0.0.1	16-11-2020	<i>CodeBusters</i>	Analisti	Creazione bozza documento, introduzione e paragrafi.

## Indice

<b>1</b>	<b>Introduzione</b>	<b>4</b>
1.1	Scopo del Documento . . . . .	4
1.2	Glossario . . . . .	4
<b>2</b>	<b>Riferimenti</b>	<b>4</b>
2.1	Normativi . . . . .	4
2.2	Informativi . . . . .	4
<b>3</b>	<b>C1 - BlockCOVID: supporto digitale al contrasto della pandemia</b>	<b>5</b>
3.1	Descrizione del capitolo . . . . .	5
3.2	Tecnologie coinvolte . . . . .	5
3.3	Vincoli . . . . .	5
3.4	Aspetti positivi . . . . .	5
3.5	Aspetti critici . . . . .	5
3.6	Conclusioni . . . . .	6
<b>4</b>	<b>C2 - EmporioLambda: piattaforma di e-commerce in stile Serverless</b>	<b>7</b>
4.1	Descrizione del capitolo . . . . .	7
4.2	Tecnologie coinvolte . . . . .	7
4.3	Vincoli . . . . .	7
4.4	Aspetti positivi . . . . .	8
4.5	Aspetti critici . . . . .	8
4.6	Conclusioni . . . . .	8
<b>5</b>	<b>C3 - GDP - Gathering Detection Platform</b>	<b>9</b>
5.1	Descrizione del capitolo . . . . .	9
5.2	Tecnologie coinvolte . . . . .	9
5.3	Vincoli . . . . .	9
5.4	Aspetti positivi . . . . .	10
5.5	Aspetti critici . . . . .	10
5.6	Conclusioni . . . . .	10
<b>6</b>	<b>C4 - HD Viz: visualizzazione di dati multidimensionali</b>	<b>11</b>
6.1	Descrizione del capitolo . . . . .	11
6.2	Tecnologie coinvolte . . . . .	11
6.3	Vincoli . . . . .	11
6.4	Aspetti positivi . . . . .	11
6.5	Aspetti critici . . . . .	12
6.6	Conclusioni . . . . .	12
<b>7</b>	<b>C5 - PORTACS - Point Of Interest Oriented Real-Time Anti Collision System</b>	<b>13</b>
7.1	Descrizione del capitolo . . . . .	13
7.2	Tecnologie coinvolte . . . . .	14
7.3	Vincoli . . . . .	14
7.4	Aspetti positivi . . . . .	14
7.5	Aspetti critici . . . . .	14
7.6	Conclusioni . . . . .	14

<b>8</b>	<b>C6 - RGP: Realtime Gaming Platform</b>	<b>15</b>
8.1	Descrizione del capitolo . . . . .	15
8.2	Tecnologie coinvolte . . . . .	15
8.3	Vincoli . . . . .	15
8.4	Aspetti positivi . . . . .	16
8.5	Aspetti critici . . . . .	16
8.6	Conclusioni . . . . .	16
<b>9</b>	<b>C7 - SSD: soluzioni di sincronizzazione desktop</b>	<b>17</b>
9.1	Descrizione del capitolo . . . . .	17
9.2	Tecnologie coinvolte . . . . .	17
9.3	Vincoli . . . . .	17
9.4	Aspetti positivi . . . . .	18
9.5	Aspetti critici . . . . .	18
9.6	Conclusioni . . . . .	18

# 1 Introduzione

## 1.1 Scopo del Documento

Questo documento contiene la stesura dello studio di fattibilità riguardante i sette capitolati proposti, per ciascuno di essi vengono evidenziati i seguenti aspetti:

- Descrizione generale;
- Prerequisiti e tecnologie coinvolte;
- Vincoli;
- Aspetti positivi;
- Aspetti critici.

Infine, per ogni capitolato vengono espone le motivazioni e le ragioni per cui il gruppo ha scelto come progetto il capitolato CX *Nome Progetto* a discapito degli altri sei proposti.

I committenti di tutti i capitolati proposti sono Prof. Vardanega Tullio e Prof. Cardin Riccardo

## 1.2 Glossario

Il gruppo *CodeBusters* ha redatto un documento denominato *Glossario 1.0.0* così da evitare ambiguità fra i termini, e per avere chiare fra tutti gli stakeholder le terminologie utilizzate per la realizzazione del presente documento. In tale documento, sono presenti tutti i termini tecnici, ambigui, specifici del progetto e scelti dai membri del gruppo con le loro relative definizioni. Un termine presente nel *Glossario 1.0.0* e utilizzato in questo documento viene indicato con un apice <sup>G</sup> alla fine della parola.

# 2 Riferimenti

## 2.1 Normativi

- *Studio di Fattibilità 0.0.0.*

## 2.2 Informativi

- **Capitolato d'appalto C1 - BLOCKCOVID:**  
<https://www.math.unipd.it/~tullio/IS-1/2020/Progetto/C1.pdf>
- **Capitolato d'appalto C2 - EmporioLambda:**  
<https://www.math.unipd.it/~tullio/IS-1/2020/Progetto/C2.pdf>
- **Capitolato d'appalto C3 - GDP:**  
<https://www.math.unipd.it/~tullio/IS-1/2020/Progetto/C3.pdf>
- **Capitolato d'appalto C4 - HD Viz:**  
<https://www.math.unipd.it/~tullio/IS-1/2020/Progetto/C4.pdf>
- **Capitolato d'appalto C5 - PORTACS:**  
<https://www.math.unipd.it/~tullio/IS-1/2020/Progetto/C5.pdf>
- **Capitolato d'appalto C6 - RGP:**  
<https://www.math.unipd.it/~tullio/IS-1/2020/Progetto/C6.pdf>
- **Capitolato d'appalto C7 - SSD:**  
<https://www.math.unipd.it/~tullio/IS-1/2020/Progetto/C7.pdf>

## 3 C1 - BlockCOVID: supporto digitale al contrasto della pandemia

### 3.1 Descrizione del capitolo

Il capitolato uno propone di sviluppare un'applicazione per registrare le presenze in uffici in modo da permettere la pulizia delle postazioni ed evitare che la stessa sia utilizzata da più persone. L'obiettivo finale è la creazione di un'applicazione mobile con due interfacce. La prima destinata agli amministratori per la gestione degli uffici e delle postazioni mentre la seconda per due categorie di utenti: il dipendente che registra la sua presenza e l'addetto alle pulizie, il quale riceve la lista delle postazioni da disinfettare.

L'azienda proponente è *imola informatica*

### 3.2 Tecnologie coinvolte

L'azienda non impone tecnologie specifiche per lo sviluppo, ma preferisce consigliare l'uso di:

- Java (versione 8 o superiori), Python o nodejs per lo sviluppo del server back-end;
- protocolli asincroni per le comunicazioni app mobile-server;
- un sistema blockchain per salvare con opponibilità a terzi i dati di sanificazione;
- IAAS Kubernetes o di un PAAS, Openshift o Rancher, per il rilascio delle componenti del server e la gestione della scalabilità orizzontale.

### 3.3 Vincoli

Gli obiettivi sono:

- Realizzare un server, e una applicazione mobile, dove è prevista la presenza di due macro-tipologie:
  - Amministratore di sistema, con i permessi di gestire e monitorare postazioni, stanze, utenti;
  - Utente, che può essere il dipendente che scansiona il tag presente alla sua postazione e segnala la sua presenza (con la possibilità di prenotarla), o un addetto alle pulizie che può gestire e monitorare le stanze da igienizzare;
- E' richiesto il report dei test effettuati;
- E' richiesta una documentazione sulle scelte, le loro relative motivazioni ed eventuali problemi incontrati con relative soluzioni.

### 3.4 Aspetti positivi

- Progetto con fine attuale ed utile alla società;
- Stimola molto interesse essendo che richiede l'uso di molti servizi;
- L'azienda risulta essere molto disponibile per seguire il gruppo nello sviluppo del progetto.

### 3.5 Aspetti critici

- Per questo capitolato è presente molta concorrenza;
- Richiede l'apprendimento di molti servizi e linguaggi che i membri del gruppo non conoscono, andando ad incidere molto sul tempo che andrebbe dedicato per l'apprendimento di questi ultimi.

### 3.6 Conclusioni

Visto il contesto attuale di pandemia un'applicazione su questo campo è quantomai utile e dunque stimola interesse. La valutazione è positiva anche per quanto riguarda le tecnologie che si dovranno esplorare per portare a termine questo capitolato. Il lato negativo risiede nella complessità dello sviluppo dell'applicazione mobile nonché del lato server.

## 4 C2 - EmporioLambda: piattaforma di e-commerce in stile Serverless

### 4.1 Descrizione del capitolo

Nel capitolo due viene proposto di creare una piattaforma di E-Commerce, ovvero la compravendita on-line di prodotti. Lo scopo finale del capitolo è avere una piattaforma dove i clienti possano registrarsi, ricercare i prodotti, aggiungerli al carrello (ovvero una pagina che permette di rivedere e considerare i prodotti e l'ammontare complessivo dei prezzi) e l'acquisto. La piattaforma dovrà offrire un'interfaccia per i venditori dove possono aggiungere e modificare i prodotti destinati alla vendita.

L'azienda proponente è *Red Babel*

### 4.2 Tecnologie coinvolte

Il progetto prevede l'adozione di una bozza iniziale dell'architettura dettata dal proponente e di alcune scelte tecnologiche obbligatorie, il fornitore è libero e altamente incoraggiato ad esplorare diverse scelte. Questa applicazione deve essere costruita con la tecnologia serverless con una API basata su HTTP:

- Si utilizzano AWS Lambda e altre componenti di supporto (AWS API Gateway, AWS DynamoDB, AWS S3);
- CloudFormation per la gestione delle risorse sopraelencate;
- Serverless Framework per l'implementazione di applicazioni serverless facilitandone alcune difficili strutture (come permessi, sottoscrizioni o accesso);
- Il BFF (Back end for Front end) sarà implementato in Next.js e richiede l'uso di Typescript come linguaggio principale;
- Il codice sorgente dovrebbe essere pubblicato e aggiornato utilizzando GitHub o GitLab.

### 4.3 Vincoli

E' richiesto l'uso di un numero minimo di ambienti di lavoro: uno locale di ogni sviluppatore, uno per i test accessibile a tutti gli sviluppatori, e un ambiente pubblico accessibile dagli utenti. E' richiesta l'implementazione di tutti i moduli di alto livello:

- EmporioLambda-frontend (EML-FE): modulo che serve le pagine web richieste dal cliente;
- EmporioLambda-backend (EML-BE): modulo che espone i servizi dell'applicazione;
- EmporioLambda-integration (EML-I): rappresenta tutti i servizi di terze parti;
- EmporioLambda-monitoring (EML-MON): è il set di strumenti usati per monitorare lo stato dell'applicazione.

E' richiesta la presenza di determinate minime, (ognuna con le proprie funzioni principali):

- Homepage
- Product Listing Page
- Product Detail Pages
- Shopping Cart
- Account



- Checkout
- Merchant dashboard

Il sito deve implementare i seguenti ruoli utilizzando AWS Cognito Identity:

- Amministratore
- Venditore
- Cliente

L'unica integrazione obbligatoria richiesta è il provider di pagamento, Stripe.

#### 4.4 Aspetti positivi

- E' richiesto l'uso di numerosi servizi forniti da AWS, sia comuni che meno, i quali possono risultare utili al nostro futuro;
- Nel capitolato viene richiesto il pagamento elettronico, argomento attualmente di nicchia;
- Linguaggi come TypeScript e JavaScript sono molto utili anche al di fuori di questo argomento;
- Nel 2020 ormai gli e-commerce sono alla portata di tutti e fanno parte della vita quotidiana delle persone, perciò è un argomento familiare ai membri del gruppo.

#### 4.5 Aspetti critici

- La presentazione del capitolato è approssimativa visto che non è stato fatto alcun seminario di approfondimento.

#### 4.6 Conclusioni

Per il capitolato due si utilizzano tecnologie non troppo complesse e, prendendo in considerazione le altre piattaforme di e-commerce esistenti (per esempio Amazon), è relativamente facile progettarela visto che l'e-commerce è ormai la quotidianità. Proprio perché ci sono già moltissime piattaforme di e-commerce esistenti rende questo capitolato poco appetibile.

## 5 C3 - GDP - Gathering Detection Platform

### 5.1 Descrizione del capitolo

Il capitolato presentato si pone come obiettivo finale la realizzazione di una piattaforma in grado di dare all'utilizzatore finale una rappresentazione grafica delle informazioni sulla probabilità di assembramento nelle zone potenzialmente a rischio. Questo viene fatto sulla base di alcuni dati che vengono individuati, raccolti ed elaborati. La piattaforma deve sfruttare tecniche di machine learning e/o deeplearnig in modo da predire le zone potenzialmente a rischio.

L'azienda proponente è *SyncLab*

### 5.2 Tecnologie coinvolte

Sebbene l'azienda non impone tecnologie specifiche per lo sviluppo del server o della UI, vi sono comunque delle scelte preferenziali da considerare nello svolgimento del progetto:

- utilizzo di Java<sup>G</sup> e di Angular<sup>G</sup> per lo sviluppo della parte di Back-end e di Front-end della componente Web Application del sistema;
- per la gestione delle mappe (heatmap ecc.) il framework Leaflet<sup>G</sup> (<https://leafletjs.com>);
- utilizzo di protocolli asincroni per le comunicazioni tra le diverse componenti;
- utilizzo del pattern Publisher/Subscriber, e adozione del protocollo MQTT ('MQ Telemetry Transport or Message Queue Telemetry Transport'), caratterizzato per essere open, di facile implementazione e ampia diffusione in applicazioni M2M (MachineToMachine) e IoT (InternetOfThings).

Per la parte di Machine Learning, l'azienda da ampia libertà, sebbene ci siano delle tecnologie consigliate:

- Python<sup>G</sup> come linguaggio di programmazione;
- TensorFlow<sup>G</sup>, Pytorch<sup>G</sup>, Keras<sup>G</sup> e Scikit-learn<sup>G</sup> come libreria per l'apprendimento automatico;

Infine, per facilitare la comprensione delle librerie consigliate, l'azienda propone delle guide come cartacee come:

- Hands-on Machine Learning with Stick-Learn, Keras & TensorFlow;
- Phyton Machine Learning,

e qualche altra risorsa online come Kaggle, Aurelien Geron's Github e Google scholar.

### 5.3 Vincoli

La proponente suggerisce una architettura in grado di implementare quello che si definisce essere un Sistema Reattivo, in grado cioè di soddisfare le seguenti caratteristiche:

- responsive: la richiesta di un servizio deve sempre avere una risposta, anche quando si verifica un guasto;
- resilient: i servizi devono poter essere ripristinati a seguito di guasti;
- elastic: i servizi devono poter essere scalati in base alla effettiva domanda;
- message-driven: i servizi devono rispondere al mondo, non tentare di controllare ciò che fa.

Ed è richiesto che tutte le componenti applicative siano correlate da test unitari e d'integrazione. Inoltre, è richiesto che il sistema venga testato nella sua interezza tramite test *end-to-end*.

## 5.4 Aspetti positivi

- Il capitolato ha come obiettivo la creazione di una piattaforma di grande aiuto a fronte della situazione che stiamo vivendo;
- Nel capitolato si tratta l'argomento del machine learning<sup>G</sup>, un argomento molto interessante e che potrebbe risultare molto utile in altri ambiti e in futuro;
- L'azienda sembra essere molto disponibile per seguire il gruppo nel percorso di sviluppo del progetto.

## 5.5 Aspetti critici

- Il capitolato richiede l'apprendimento di molti servizi, sconosciuti dai membri del gruppo, che si basano su argomenti non conosciuti (come il machine learning<sup>G</sup>) e non affrontati nel corso di laurea triennale. Quindi gran parte del tempo andrebbe dedicato allo studio di questi concetti e strumenti, ed infine di applicare queste conoscenze;
- Oltre al tempo impiegato per l'apprendimento di questi strumenti c'è da tenere in considerazione il tempo per addestrare la parte di machine learning<sup>G</sup>.

## 5.6 Conclusioni

Nonostante tale capitolato abbia destato particolare interesse all'interno del gruppo, specialmente per la possibilità di utilizzare tecnologie innovative, il team ha valutato la complessità di tale progetto come molto elevata e ha preferito orientarsi verso un'altra alternativa.

## 6 C4 - HD Viz: visualizzazione di dati multidimensionali

### 6.1 Descrizione del capitolo

Il capitolato C4 ha come obiettivo quello di creare un'applicazione di visualizzazione di grandi quantità di dati con numerose dimensioni in un formato comprensibile dall'occhio umano. A questo scopo è necessario sviluppare algoritmi di intelligenza artificiale che, agendo sulla distanza dei vari punti del grafico, riescano a sviluppare un modello semplificato che ne evidenzia i cluster. L'applicazione dovrà inoltre agire su questi grafici creati evidenziando i dati ottenuti.

L'azienda proponente è *Zucchetti*

### 6.2 Tecnologie coinvolte

L'applicazione "HD Viz" dovrà utilizzare le seguenti tecnologie:

- HTML/CSS/Javascript con libreria D3.js per la parte client;
- La parte server e un database (SQL o NOSQL) potranno essere sviluppati in Java con server Tomcat o in Javascript con server Node.js.

### 6.3 Vincoli

I vincoli del capitolato C4 riguardano, in prima istanza, il numero di dimensioni dei dati da visualizzare. L'applicazione deve riuscire a visualizzare dati con almeno 15 dimensioni. Il formato dei dati da fornire a HD Viz deve essere una query ad un database o anche un file in formato CSV preparato in precedenza. HD Viz dovrà presentare le seguenti visualizzazioni:

- Scatter plot Matrix (fino ad un massimo di 5 dimensioni): è la presentazione a riquadri disposti a matrice di tutte le combinazioni di scatter plot;
- Force Field: converte dati in più dimensioni in un grafico a due/tre dimensioni rappresentandoli sulla base della loro distanza ;
- Heat Map: rappresenta la distanza tra i punti con colori più o meno intensi. In questa visualizzazione è necessario ordinare i dati per evidenziarli;
- Proiezione Lineare Multi Asse: si rappresentano a due dimensioni dati multidimensionali permettendo di spostare gli assi per rendere la visualizzazione più comprensibile.

Il proponente lascia ampia libertà di esplorazione dell'argomento e accetta altri requisiti opzionali proposti dal fornitore.

### 6.4 Aspetti positivi

- L'analisi dei Big Data è una problematica più che mai attuale e che potrebbe risultare molto utile in ambito professionale;
- Nel capitolato si tratta l'intelligenza artificiale, altro argomento molto interessante in un futuro ambito lavorativo;
- Gli strumenti HTML/CSS e Javascript sono già stati visti nel corso di Tecnologie Web;
- La libreria D3 è ben documentata e largamente fornita di esempi sulle visualizzazioni richieste;
- L'azienda ha dato disponibilità per ulteriori chiarimenti in corso d'opera.

## **6.5 Aspetti critici**

- Il capitolato potrebbe risultare difficoltoso nella realizzazione degli algoritmi e nel testarli;
- Altri capitolati hanno per argomento oggetti o situazioni più reali in confronto all'analisi ed elaborazione dei dati.

## **6.6 Conclusioni**

Il capitolato C4 è tra i preferiti dal gruppo che infine ha deciso di (...)

## 7 C5 - PORTACS - Point Of Interest Oriented Real-Time Anti Collision System

### 7.1 Descrizione del capitolo

Il capitolato si pone come obiettivo la realizzazione di una piattaforma in grado di fornire un'interfaccia grafica consenta all'utente utilizzatore il monitoraggio delle attività delle seguenti realtà come:

- Il Robot cameriere: percorre percorsi predefiniti, ritiro e consegna dell'ordine;
- Geolocalizzazione del magazzino: ogni mezzo e persona si spostano all'interno del magazzino ed è georeferenziato; i mezzi devono rispettare il senso di percorrenza della corsia per evitare possibili collisioni; ogni persona a piedi o su mezzo deve raggiungere varie ubicazioni per poter preparare ordini ;
- Geolocalizzazione nei trasporti: ogni mezzo ha un punto di partenza con una lista di consegne deve effettuare; velocità media e deve evitare collisioni; il responsabile è in grado di monitorare lo stato di ogni mezzo in tempo reale;
- Auto a guida automatica: bisogna definire la destinazione dell'auto; ogni auto può controllare la propria velocità così da evitare possibili collisioni.

Queste realtà hanno dei punti in comune: mappatura della posizione in tempo reale e tenere conto dei punti di partenza e destinazione, possibilità di monitorare la velocità ottimale e velocità massima, evitare gli ostacoli/collisioni.

Ogni unità (può rappresentare un robot, un muletto, un'auto) ha punti di partenza nella griglia e una lista di POI "*Point Of Interest*" da raggiungere, con una velocità massimale deve inviare al sistema:

- la propria posizione costantemente;
- la direzione e velocità, in modo tale che il sistema centrale possa coordinare e pilotare tutte le unità per evitare incidenti e ingorghi.

La User Interface conterrà ogni singola unità che dovrà provvedere delle 4 frecce direzionali (si "*accenderà*" quella suggerita dal sistema), il pulsante start/stop e l'indicatore della velocità attuale (limite massimo è la velocità anagrafica).

Il sistema è corredato da un display in real-time della mappa e relativa posizione di ogni singola unità e inoltre il sistema indicherà ad ogni unità la prossima mossa da compiere, in funzione di:

- prossimo POI da raggiungere ;
- posizione delle altre unità evitando collisioni e vanno rispettati vincoli dimensionali (limite delle corsie);
- Plus: posizioni dei pedoni per evitare collisioni.

Il software dovrà essere in grado di accettare i seguenti input:

- Schacchiera oppure mappa con percorsi predefiniti e relativi vincoli (sensi unici, numero massimo di unità contemporaneamente);
- definizione di POI (aree di carico e scarico, sosta).

L'azienda proponente è *Sanmarco Informatica*

## 7.2 Tecnologie coinvolte

Il proponente non impone specifiche tecnologie per lo sviluppo della piattaforma ma invita all'utilizzo di quelle più funzionali e moderne. Tra queste si segnala:

- utilizzo di Nodejs per la parte di Back-end e Reactjs/next.js per lo sviluppo di Front-end che permettono di creare un SPA *"Single Page Application"* ;
- Socket: è una libreria che consente di sviluppare applicazioni real-time e risulta facile da integrare con il Nodejs;

## 7.3 Vincoli

L'azienda proponente non pone vincoli per la scelta degli strumenti e tecnologie di sviluppo ma vi sono alcuni vincoli legati al funzionamento della piattaforma:

- il sistema dovrà avere una visualizzazione in real-time la mappa e la posizione delle singole unità;
- ogni unità dovrà inviare al sistema centrale costantemente la propria posizione, direzione e velocità;
- il sistema deve accettare come input: una Scacchiera o mappa con percorsi.

## 7.4 Aspetti positivi

L'obiettivo del capitolato è quello di creare una piattaforma real-time che facilita il monitoraggio dello stato delle attività di ogni singola unità (robot, muletto, automobile); inoltre il progetto proposto consente all'utente sviluppatore di apprendere le nuove tecnologie che verranno utilizzate per implementare le varie funzionalità, in particolare l'implementazione del meccanismo real-time.

Un altro aspetto positivo è dato dal fatto che l'azienda proponente ha dimostrato disponibilità per ulteriori chiarimenti e/o informazioni.

## 7.5 Aspetti critici

L'azienda proponente non segnala tecnologie per lo sviluppo della piattaforma conosciute e apprese in ambito universitario; ciò potrebbe comportare una dilatazione dei tempi per la scelta delle tecnologie più adeguate e per la formazione e l'autoapprendimento da parte dei membri del gruppo.

## 7.6 Conclusioni

Ciò nonostante, tale capitolato ha attirato l'attenzione e stimolato l'interesse dei membri del gruppo in particolar modo per la possibilità di apprendere nuove tecnologie moderne.

Il capitolato in questione presenta un livello di difficoltà elevato e richiede impegno; alla luce di questo il gruppo si rende disponibile a sviluppare quanto richiesto dal proponente.

## 8 C6 - RGP: Realtime Gaming Platform

### 8.1 Descrizione del capitolo

Il capitolato presentato si pone come obiettivo finale la realizzazione di una piattaforma in grado di dare all'utilizzatore finale una rappresentazione grafica delle informazioni sulla probabilità di assemblamento nelle zone potenzialmente a rischio. Questo viene fatto sulla base di alcuni dati che vengono individuati, raccolti ed elaborati. La piattaforma deve sfruttare tecniche di machine learning e/o deeplearnig in modo da predire le zone potenzialmente a rischio.

L'azienda proponente è *Zero 12*

### 8.2 Tecnologie coinvolte

Sebbene l'azienda non impone tecnologie specifiche per lo sviluppo del server o della UI, vi sono comunque delle scelte preferenziali da considerare nello svolgimento del progetto:

- utilizzo di Java<sup>G</sup> e di Angular<sup>G</sup> per lo sviluppo della parte di Back-end e di Front-end della componente Web Application del sistema;
- per la gestione delle mappe (heatmap ecc.) il framework Leaflet<sup>G</sup> (<https://leafletjs.com>);
- utilizzo di protocolli asincroni per le comunicazioni tra le diverse componenti;
- utilizzo del pattern Publisher/Subscriber, e adozione del protocollo MQTT ('MQ Telemetry Transport or Message Queue Telemetry Transport'), caratterizzato per essere open, di facile implementazione e ampia diffusione in applicazioni M2M (MachineToMachine) e IoT (InternetOfThings).

Per la parte di Machine Learning, l'azienda da ampia libertà, sebbene ci siano delle tecnologie consigliate:

- Python<sup>G</sup> come linguaggio di programmazione;
- TensorFlow<sup>G</sup>, Pytorch<sup>G</sup>, Keras<sup>G</sup> e Scikit-learn<sup>G</sup> come libreria per l'apprendimento automatico;

Infine, per facilitare la comprensione delle librerie consigliate, l'azienda propone delle guide come cartacee come:

- Hands-on Machine Learning with Stick-Learn, Keras & TensorFlow;
- Phyton Machine Learning,

e qualche altra risorsa online come Kaggle, Aurelien Geron's Github e Google scholar.

### 8.3 Vincoli

La proponente suggerisce una architettura in grado di implementare quello che si definisce essere un Sistema Reattivo, in grado cioè di soddisfare le seguenti caratteristiche:

- responsive: la richiesta di un servizio deve sempre avere una risposta, anche quando si verifica un guasto;
- resilient: i servizi devono poter essere ripristinati a seguito di guasti;
- elastic: i servizi devono poter essere scalati in base alla effettiva domanda;
- message-driven: i servizi devono rispondere al mondo, non tentare di controllare ciò che fa.

Ed è richiesto che tutte le componenti applicative siano correlate da test unitari e d'integrazione. Inoltre, è richiesto che il sistema venga testato nella sua interezza tramite test *end-to-end*.



## 8.4 Aspetti positivi

- Il capitolato ha come obiettivo la creazione di una piattaforma di grande aiuto a fronte della situazione che stiamo vivendo;
- Nel capitolato si tratta l'argomento del machine learning<sup>G</sup>, un argomento molto interessante e che potrebbe risultare molto utile in altri ambiti e in futuro;
- L'azienda sembra essere molto disponibile per seguire il gruppo nel percorso di sviluppo del progetto.

## 8.5 Aspetti critici

- Il capitolato richiede l'apprendimento di molti servizi, sconosciuti dai membri del gruppo, che si basano su argomenti non conosciuti (come il machine learning<sup>G</sup>) e non affrontati nel corso di laurea triennale. Quindi gran parte del tempo andrebbe dedicato allo studio di questi concetti e strumenti, ed infine di applicare queste conoscenze;
- Oltre al tempo impiegato per l'apprendimento di questi strumenti c'è da tenere in considerazione il tempo per addestrare la parte di machine learning<sup>G</sup>.

## 8.6 Conclusioni

Nonostante tale capitolato abbia destato particolare interesse all'interno del gruppo, specialmente per la possibilità di utilizzare tecnologie innovative, il team ha valutato la complessità di tale progetto come molto elevata e ha preferito orientarsi verso un'altra alternativa.

## 9 C7 - SSD: soluzioni di sincronizzazione desktop

### 9.1 Descrizione del capitolo

Il capitolato presentato si pone come obiettivo finale la realizzazione di una piattaforma in grado di dare all'utilizzatore finale una rappresentazione grafica delle informazioni sulla probabilità di assemblamento nelle zone potenzialmente a rischio. Questo viene fatto sulla base di alcuni dati che vengono individuati, raccolti ed elaborati. La piattaforma deve sfruttare tecniche di machine learning e/o deeplearnig in modo da predire le zone potenzialmente a rischio. L'azienda proponente è *Zextris*

### 9.2 Tecnologie coinvolte

Sebbene l'azienda non impone tecnologie specifiche per lo sviluppo del server o della UI, vi sono comunque delle scelte preferenziali da considerare nello svolgimento del progetto:

- utilizzo di Java<sup>G</sup> e di Angular<sup>G</sup> per lo sviluppo della parte di Back-end e di Front-end della componente Web Application del sistema;
- per la gestione delle mappe (heatmap ecc.) il framework Leaflet<sup>G</sup> (<https://leafletjs.com>);
- utilizzo di protocolli asincroni per le comunicazioni tra le diverse componenti;
- utilizzo del pattern Publisher/Subscriber, e adozione del protocollo MQTT ('MQ Telemetry Transport or Message Queue Telemetry Transport'), caratterizzato per essere open, di facile implementazione e ampia diffusione in applicazioni M2M (MachineToMachine) e IoT (InternetOfThings).

Per la parte di Machine Learning, l'azienda da ampia libertà, sebbene ci siano delle tecnologie consigliate:

- Python<sup>G</sup> come linguaggio di programmazione;
- TensorFlow<sup>G</sup>, Pytorch<sup>G</sup>, Keras<sup>G</sup> e Scikit-learn<sup>G</sup> come libreria per l'apprendimento automatico;

Infine, per facilitare la comprensione delle librerie consigliate, l'azienda propone delle guide come cartacee come:

- Hands-on Machine Learning with Stick-Learn, Keras & TensorFlow;
- Phyton Machine Learning,

e qualche altra risorsa online come Kaggle, Aurelien Geron's Github e Google scholar.

### 9.3 Vincoli

La proponente suggerisce una architettura in grado di implementare quello che si definisce essere un Sistema Reattivo, in grado cioè di soddisfare le seguenti caratteristiche:

- responsive: la richiesta di un servizio deve sempre avere una risposta, anche quando si verifica un guasto;
- resilient: i servizi devono poter essere ripristinati a seguito di guasti;
- elastic: i servizi devono poter essere scalati in base alla effettiva domanda;
- message-driven: i servizi devono rispondere al mondo, non tentare di controllare ciò che fa.

Ed è richiesto che tutte le componenti applicative siano correlate da test unitari e d'integrazione. Inoltre, è richiesto che il sistema venga testato nella sua interezza tramite test *end-to-end*.

## 9.4 Aspetti positivi

- Il capitolato ha come obiettivo la creazione di una piattaforma di grande aiuto a fronte della situazione che stiamo vivendo;
- Nel capitolato si tratta l'argomento del machine learning<sup>G</sup>, un argomento molto interessante e che potrebbe risultare molto utile in altri ambiti e in futuro;
- L'azienda sembra essere molto disponibile per seguire il gruppo nel percorso di sviluppo del progetto.

## 9.5 Aspetti critici

- Il capitolato richiede l'apprendimento di molti servizi, sconosciuti dai membri del gruppo, che si basano su argomenti non conosciuti (come il machine learning<sup>G</sup>) e non affrontati nel corso di laurea triennale. Quindi gran parte del tempo andrebbe dedicato allo studio di questi concetti e strumenti, ed infine di applicare queste conoscenze;
- Oltre al tempo impiegato per l'apprendimento di questi strumenti c'è da tenere in considerazione il tempo per addestrare la parte di machine learning<sup>G</sup>.

## 9.6 Conclusioni

Nonostante tale capitolato abbia destato particolare interesse all'interno del gruppo, specialmente per la possibilità di utilizzare tecnologie innovative, il team ha valutato la complessità di tale progetto come molto elevata e ha preferito orientarsi verso un'altra alternativa.