



# Analisi dei Requisiti

Jawa Druids

Versione	x.x.x
Data approvazione	xx-xx-xxxx
Responsabile	Nome Cognome
Redattori	Nome Cognome
Verificatori	Nome Cognome
	Nome Cognome
Stato	Stato
Lista distribuzione	Jawa Druids
	Nome Professori
	Sync Lab
Uso	Uso del documento

## Sommario

L'*Analisi dei Requisiti* individua tutti i requisiti da implementare nel prodotto da sviluppare.



## Registro delle modifiche

Modifica	Autore	Ruolo	Data	Versione
<i>Descrizione delle modifiche</i>	Nome Cognome	<i>Analista</i>	23-11-2020	v0.0.3
<i>Stesura cp 2</i>	Andrea Cecchin	<i>Analista</i>	10-12-2020	v0.0.2
<i>Inizio stesura documento cp 1</i>	Andrea Cecchin	<i>Analista</i>	07-12-2020	v0.0.1



# Indice

<b>1</b>	<b>Introduzione</b>	<b>3</b>
1.1	Scopo del documento . . . . .	3
1.2	Scopo del prodotto . . . . .	3
1.3	Glossario . . . . .	3
1.4	Riferimenti . . . . .	3
1.4.1	Riferimenti normativi . . . . .	3
1.4.2	Riferimenti informativi . . . . .	4
<b>2</b>	<b>Descrizione generale</b>	<b>5</b>
2.1	Caratteristiche del prodotto . . . . .	5
2.2	Funzionalità generali . . . . .	5
2.3	Caratteristiche utente . . . . .	5
<b>3</b>	<b>Requisiti specifici</b>	<b>6</b>
3.1	FC1: Acquisizione dati . . . . .	6
3.1.1	FC1.1: Acquisizione con java . . . . .	6
3.1.2	FC1.2: Creazione database . . . . .	6
3.1.3	FC1.3: Salvataggio dati . . . . .	6
3.2	FC2: Elaborazione Dati . . . . .	7
3.2.1	FC2.1: Esplorazione Dati . . . . .	7
3.2.2	FC2.2: Preprocessing . . . . .	7
3.2.3	FC2.3: Caso predizione . . . . .	8
3.2.4	FC2.4: Valutazioni e validazione . . . . .	8
3.3	FC3: Visualizzazione dati . . . . .	8
3.3.1	Creazione pagina web . . . . .	8



# Introduzione

## 1.1 Scopo del documento

Lo scopo del documento è quello di formalizzare i contenuti e le qualità che il prodotto sviluppato dovrà raggiungere. I requisiti sono stati individuati attraverso lo studio del capitolato e incontri con l'azienda proponente *Sync Lab*. Il documento inoltre è necessario a:

- descrivere accuratamente tutti i requisiti proposti dal proponente;
- comprendere da parte del committente quali sono le richieste del cliente;
- definire il formato e contenuto di ogni requisito specifico del software.

## 1.2 Scopo del prodotto

In seguito alla pandemia del virus COVID-19 è nata l'esigenza di limitare il più possibile i contatti fra le persone, specialmente evitando la formazione di assembramenti. Il progetto *GDP: Gathering Detection Platform* di *Sync Lab* ha pertanto l'obiettivo di **creare una piattaforma in grado di rappresentare graficamente le zone potenzialmente a rischio di assembramento, al fine di prevenirlo.**

Al tal fine il gruppo *Jawa Druids* si prefigge di sviluppare un prototipo software in grado di acquisire, monitorare ed analizzare i molteplici dati provenienti dai diversi sistemi e dispositivi, a scopo di identificare i possibili eventi che concorrono all'insorgere di variazioni di flussi di utenti. Il gruppo prevede inoltre lo sviluppo di un'applicazione web da interporre fra i dati elaborati e l'utente, per favorirne la consultazione.

## 1.3 Glossario

All'interno della documentazione viene fornito un *Glossario*, con l'obiettivo di assistere il lettore specificando il significato e contesto d'utilizzo di alcuni termini strettamente tecnici o ambigui, segnalati con una *G* a pedice.

## 1.4 Riferimenti

### 1.4.1 Riferimenti normativi

- *Norme di Progetto*



### 1.4.2 Riferimenti informativi

- *IEEE Recommended Practice for Software Requirements Specifications:*  
<https://ieeexplore.ieee.org/document/720574>
- *Seminario per approfondimenti tecnici del capitolato C3:*  
<https://www.math.unipd.it/~tullio/IS-1/2020/Progetto/ST1.pdf>



## Descrizione generale

### 2.1 Caratteristiche del prodotto

L'idea del capitolato *GDP* - *Gathering Detection Platform* è di creare una piattaforma che riesca a rappresentare mediante visualizzazione grafica zone potenzialmente a rischio di assembramento e cercare di prevenirle. La piattaforma utilizzerà dati prelevati da sensori (come telecamere, dispositivi contapersone, etc.) o sorgenti dati (come flussi di prenotazioni Uber, le tabelle degli orari di autobus/metro/treno, etc.), i quali mediante la loro elaborazione verranno rappresentati tramite una *heat-map*.

### 2.2 Funzionalità generali

Il capitolato *GDP* individua tre principali funzionalità da sviluppare:

- Acquisizione di dati: l'acquisizione avverrà attraverso sistemi di monitoraggio e motori software "contapersone" applicati ad immagini/stream delle videocamere;
- Elaborazione di dati: i dati verranno elaborati per generare valore aggiunto agli stessi e confrontare flussi diversi di informazioni;
- Rappresentazione di dati: attraverso un sito web i dati elaborati verranno visualizzati a video mediante una *heat-map*

### 2.3 Caratteristiche utente

Il progetto è rivolto principalmente ad utenti di tipo amministrativi, cioè i quali devono visualizzare l'intera mappa di una regione per motivi lavorativi.



## Requisiti specifici

In questo capitolo verranno illustrati le funzionalità identificate dal capitolato d'appalto *GDP-Gathering Detection Platform*.

### 3.1 FC1: Acquisizione dati

L'acquisizione dei dati avverrà in tre fasi descritte di seguito.

#### 3.1.1 FC1.1: Acquisizione con java

- **Descrizione:** attraverso il linguaggio Java si creerà un programma che preleva informazioni da sorgenti esterne e le invia al server.
- **Linguaggio di programmazione:** Java.
- **Input:** i dati forniti saranno prelevati da siti con live-feed di webcam di Roma e simulatori di spostamenti di persone.
- **Output:** i dati resteranno immutati.
- **Risposta ad errori:** nel caso di mancanza di risposta dai siti con live-feed il programma si bloccherà ed invierà un segnale di errore al server.

#### 3.1.2 FC1.2: Creazione database

- **Descrizione:** attraverso lo studio dei tipi di dato acquisiti si svilupperà un database (??? che tipo ???).
- ??? si può aggiungere una bozza dello schema er ???

#### 3.1.3 FC1.3: Salvataggio dati

- **Descrizione:** i dati verranno inseriti all'interno di un database.
- **Tipo di server:** ???non so cosa scrivere???

??? I dati verranno inseriti all'interno di un database, questo sarà sviluppato usando Apache Kafka un sistema distribuito che consiste di server e client i quali comunicano tra loro attraverso un protocollo di rete performante di tipo TCP. ???(non so dove inserire)



## 3.2 FC2: Elaborazione Dati

Ottenuti i dati, essi verranno elaborati attraverso librerie di sci-kit e tensorflow con il linguaggio di programmazione python. Di seguito vengono individuate le fasi dell'elaborazione dei dati.

### 3.2.1 FC2.1: Esplorazione Dati

- **Descrizione:** si discriminano elementi all'interno del dataset che portano a predizioni errate del modello.
- **Input:** i dati vengono prelevati dal database ???(ita o eng).
- **Output:** i dati controllati vengono aggiunti in appositi spazi per individuare la loro correttezza.
- **Processo:** si controlla se c'è presenza di valori mancanti, dataset non bilanciati, outliers, livello di rumore dei dati e correlazione dei dati. ???(significato: Outlier è un termine utilizzato in statistica per definire, in un insieme di osservazioni, un valore anomalo e aberrante. Un valore quindi chiaramente distante dalle altre osservazioni disponibili.) ???

### 3.2.2 FC2.2: Preprocessing

- **Descrizione:** preparazione dei dati grezzi e renderli adatti ad un modello di machine learning.
- **Input:** i dati controllati.
- **Output:** dati pronti per l'elaborazione nel modello machine learning.
- **Processo:**

[leftmargin = 2cm]Cleaning: eliminazione o correzione di dati con valori invalidi o corrotti. Trasformazione dei dati: i dati vengono normalizzati, si calcolano nuove variabili etc ???(non mi piace cosa ho scritto)??? Feature extraction: si ricavano attraverso i dati trasformati valori derivati, i quali sono più informativi e non ridondanti, facilitano le fasi successive di apprendimento e generalizzazione. Filtraggio dei dati: attraverso appositi filtri eliminare i dati ridondanti e irrilevanti al training del modello. Train / Test set splitting: si dividono i dati in due gruppi uno per il training e uno per il testing





### 3.2.3 FC2.3: Caso predizione

- **Descrizione:** in questa fase si effettua una scelta sull'algoritmo più adeguato da utilizzare per il training di dati.
- **Input:** dati controllati nella fase di preprocessing
- **Output:** modello di ML allenato sui dati di input
- **Tipi di algoritmi:** si dividono per classificazione e regressione.???non so se va bene???

### 3.2.4 FC2.4: Valutazioni e validazione

- **Descrizione:** attraverso varie metriche si valuta quanto valido è il modello nella predizione dei casi.
- **Input:** risposta del modello ML dai dati di test

## 3.3 FC3: Visualizzazione dati

### 3.3.1 Creazione pagina web

- **Descrizione:** sviluppo di una pagina web semplice ed intuitiva.
- **Strumenti:** si utilizzerà Angular e Spring, due librerie per framework di javascript.
- **Vincolo:** la web app dovrà essere costruita sia desktop che mobile friendly.
- **Struttura:** la pagina sarà divisa principalmente in 3 sezioni. ???si potrebbe aggiungere l'immagine del template del sito???

La pagina web creata attraverso html, php e javascript permetterà la visualizzazione dei dati elaborati. I dati verranno prelevati attraverso relative query al server utilizzando il linguaggio di programmazione php. La pagina sarà suddivisa principalmente in tre parti:



Logo/title

Search address:

MAP

Link to information page/validation mark like css or html/  
author



La sezione in alto individua un banner per il logo e login per gli utenti admin per effettuare modifiche alla mappa. La sezione centrale visualizza la heat-map, una search bar per cercare il sito di assembramento che si vuole controllare e a lato di essa un menù per impostazioni di parametri aggiuntivi.

la sezione di footer sarà utilizzata per inserimento di informazioni sull'azienda proponente, il gruppo Jawa Druids e link per relativi ai propri siti web.