



# Analisi dei Requisiti

## Jawa Druids

<b>Versione</b>	v1.0.0
<b>Data approvazione</b>	10-01-2021
<b>Responsabile</b>	Andrea Cecchin
<b>Redattori</b>	Andrea Cecchin Igli Mezini
<b>Verificatori</b>	Alfredo Graziano Mattia Cocco
<b>Stato</b>	Approvato
<b>Lista distribuzione</b>	<i>Jawa Druids</i> <i>Prof. Tullio Vardanega</i> <i>Prof. Riccardo Cardin</i> <i>Sync Lab</i>
<b>Uso</b>	Esterno

## Sommario

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



## Registro delle modifiche

Modifica	Autore	Ruolo	Data	Versione
<i>Approvazione del documento per RR.</i>	Andrea Cecchin	<i>Responsabile</i>	10-01-2021	v1.0.0
<i>Verifica intero documento.</i>	Alfredo Graziano	<i>Verificatore</i>	09-01-2021	v0.2.0
<i>Aggiunta sezione § 4.5.</i>	Igli Mezini	<i>Analista</i>	07-01-2021	v0.1.5
<i>Aggiunte sezioni § 4.4.1, § 4.4.2.</i>	Igli Mezini	<i>Analista</i>	06-01-2021	v0.1.4
<i>Aggiunte sezioni § 2.3, § 4.2, § ??, § 4.3 .</i>	Igli Mezini	<i>Analista</i>	05-01-2021	v0.1.3
<i>Modifica sezione § 4.1 .</i>	Andrea Cecchin	<i>Analista</i>	05-01-2021	v0.1.2
<i>Inizio stesura § 4.</i>	Andrea Cecchin	<i>Analista</i>	04-01-2021	v0.1.1
<i>Verifica § 1, § 2.</i>	Mattia Cocco	<i>Verificatore</i>	28-12-2020	v0.1.0
<i>Stesura § 3.</i>	Andrea Cecchin	<i>Analista</i>	28-12-2020	v0.0.4
<i>Aggiunte sezioni § 3.1, § 3.2.</i>	Andrea Cecchin	<i>Analista</i>	10-12-2020	v0.0.3
<i>Aggiunte sezioni § 2.1, § 2.2.</i>	Andrea Cecchin	<i>Analista</i>	10-12-2020	v0.0.2
<i>Inizio stesura documento § 1.</i>	Andrea Cecchin	<i>Analista</i>	07-12-2020	v0.0.1



# Indice

<b>1</b>	<b>Introduzione</b>	<b>4</b>
1.1	Scopo del documento . . . . .	4
1.2	Scopo del prodotto . . . . .	4
1.3	Glossario . . . . .	4
1.4	Riferimenti . . . . .	5
1.4.1	Riferimenti normativi . . . . .	5
1.4.2	Riferimenti informativi . . . . .	5
<b>2</b>	<b>Descrizione generale</b>	<b>6</b>
2.1	Caratteristiche del prodotto . . . . .	6
2.2	Funzionalità generali . . . . .	6
2.3	Caratteristiche utente . . . . .	6
<b>3</b>	<b>Fasi del progetto</b>	<b>7</b>
3.1	FC1: Acquisizione dati . . . . .	7
3.1.1	FC1.1: Acquisizione con Java . . . . .	7
3.1.2	FC1.2: Database . . . . .	7
3.1.3	FC1.3: Apache Kafka <sub>G</sub> . . . . .	8
3.2	FC2: Elaborazione Dati . . . . .	8
3.2.1	FC2.1: Esplorazione Dati . . . . .	8
3.2.2	FC2.2: Preprocessing . . . . .	8
3.2.3	FC2.3: Caso predizione . . . . .	9
3.2.3.1	FC2.4: Valutazioni e validazione . . . . .	9
3.3	FC3: Visualizzazione dati . . . . .	9
3.3.1	FC3.1: Front-end <sub>G</sub> . . . . .	9
3.3.2	FC3.2: Back-end <sub>G</sub> . . . . .	10
<b>4</b>	<b>Requisiti</b>	<b>11</b>
4.1	Requisiti funzionali . . . . .	11
4.2	Requisiti prestazionali . . . . .	12
4.3	Requisiti di vincolo . . . . .	14
4.4	Tracciamento dei requisiti . . . . .	15
4.4.1	Requisito - fonte . . . . .	15
4.4.2	Fonte - requisito . . . . .	17
4.5	Considerazioni . . . . .	18



## Elenco delle tabelle

4.1	Requisiti funzionali . . . . .	12
4.2	Requisiti di qualità . . . . .	13
4.3	Requisiti di vincolo . . . . .	14
4.4	Tabella tracciamento requisito-fonte . . . . .	16
4.5	Tabella tracciamento fonte-requisito . . . . .	18



# 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<sub>G</sub> e dagli incontri con l'azienda proponente<sub>G</sub> *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<sub>G</sub> 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**. Il prodotto finale è rivolto specificatamente agli organi amministrativi delle singole città, cosicché possano gestire al meglio i punti sensibili di affollamento, come piazze o siti turistici. Lo scopo che il software intende raggiungere non è solo quello della rappresentazione grafica real-time ma anche di poter riuscire a prevedere assembramenti in intervalli futuri di tempo.

A 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 v1.0.0;*
- *Verbale Esterno 17-12-2020;*
- *Capitolato d'appalto C3:*  
<https://www.math.unipd.it/~tullio/IS-1/2020/Progetto/C3.pdf>

### 1.4.2 Riferimenti informativi

- *Presentazione del capitolato:*  
<https://www.math.unipd.it/~tullio/IS-1/2020/Progetto/C3.pdf>
- *Materiale didattico relativo all'Analisi dei Requisiti del corso di Ingegneria del Software:*  
<https://www.math.unipd.it/~tullio/IS-1/2020/Dispense/L07.pdf>
- *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<sub>G</sub> *GDP* - *Gathering Detection Platform* è di creare una piattaforma che riesca a rappresentare mediante visualizzazione grafica zone potenzialmente a rischio di assembramento con l'intento 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*<sub>G</sub>.

### 2.2 Funzionalità generali

Il capitolato<sub>G</sub> *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*<sub>G</sub>.

### 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.

Le conoscenze dell'utente per l'utilizzo del software sono:

- Conoscenza base nell'utilizzo del motore di ricerca;
- Padronanza nella lettura della *heat map*<sub>G</sub>.



## Fasi del progetto

In questo capitolo verranno illustrate le fasi del progetto identificate dal capitolato<sub>c</sub> d'appalto *GDP-Gathering Detection Platform*. Il capitolo viene diviso nelle tre fasi generali del progetto: acquisizione, elaborazione e visualizzazione dei dati. Secondo lo **IEEE Standard 830-1998** in questo capitolo verranno spiegati tutti i punti da sviluppare e nel capitolo successivo i requisiti da implementare per la creazione del prodotto richiesto da *Sync Lab*. La descrizione delle fasi è stata inserita in quanto ritenuta necessaria per il chiarimento della necessità dei requisiti individuati.

### 3.1 FC1: Acquisizione dati

In questa sezione vengono descritte le fasi di acquisizione dei dati.

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

- **Descrizione:** attraverso il linguaggio Java<sub>G</sub> si creerà un programma che preleva informazioni da sorgenti esterne e le invia al server.
- **Linguaggio di programmazione:** Java<sub>G</sub>.
- **Input:** i dati forniti saranno prelevati da siti con live-feed<sub>G</sub> di webcam di varie città 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: Database

- **Descrizione:** creazione del database e archiviazione dei dati in esso per visualizzazione future e mantenimento dei dati;
- **Linguaggio:** mySQL.





### 3.1.3 FC1.3: Apache Kafka<sub>G</sub>

- **Descrizione:** impostazione di una piattaforma di data streaming<sub>G</sub> che consente di gestire e trasferire grandi volumi di dati in tempo reale, abbassando notevolmente i tempi di latenza;
- **Input:** flussi di dati dall'acquisizione con Java<sub>G</sub>;
- **Output:** il flusso di dati rimane immutato.

## 3.2 FC2: Elaborazione Dati

Completata la fase precedente i dati verranno elaborati attraverso librerie di Scikit-learn e TensorFlow con il linguaggio di programmazione Python<sub>G</sub>. Di seguito vengono individuate le fasi da seguire per l'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.
- **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<sub>G</sub>, livello di rumore dei dati e correlazione dei dati.

### 3.2.2 FC2.2: Preprocessing

- **Descrizione:** preparazione dei dati grezzi per renderli adatti ad un modello di Machine Learning<sub>G</sub>.
- **Input:** i dati controllati.
- **Output:** dati pronti per l'elaborazione nel modello Machine Learning<sub>G</sub>.
- **Processo:**
  1. Cleaning: eliminazione o correzione di dati con valori invalidi o corrotti.
  2. Trasformazione dei dati: i dati vengono normalizzati, discretizzati, aggregati, si calcolano nuove variabili etc.



3. Feature extraction: si ricavano, attraverso i dati trasformati, i valori derivati, i quali sono più informativi e non ridondanti, facilitano le fasi successive di apprendimento e generalizzazione.
4. Filtraggio dei dati: eliminazione di dati ridondanti e irrilevanti al training del modello attraverso l'applicazione di appositi filtri.
5. 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 per il training.
- **Output:** modello di Machine Learning<sub>G</sub> allenato sui dati di input.
- **Tipi di algoritmi:** si dividono per classificazione e regressione.

#### 3.2.3.1 FC2.4: Valutazioni e validazione

- **Descrizione:** attraverso varie metriche si valuta quanto valido è il modello nella predizione dei casi.
- **Input:** risposta del modello Machine Learning<sub>G</sub> dai dati di test, dati effettivi ricavati dalle sorgenti esterne.
- **Output:** dati che superano la validazione.

## 3.3 FC3: Visualizzazione dati

In questa sezione verranno illustrate le fasi di sviluppo della parte visiva della web-app.

### 3.3.1 FC3.1: Front-end<sub>G</sub>

- **Descrizione:** sviluppo di una pagina web semplice ed intuitiva.
- **Strumenti:** si utilizzerà Angular<sub>G</sub> e Spring<sub>G</sub>, due librerie per framework<sub>G</sub> di JavaScript<sub>G</sub>.
- **Vincolo:** la web app dovrà essere costruita sia desktop che mobile friendly.
- **Struttura:** la pagina sarà principalmente rivolta alla visione della mappa per la visualizzazione di aree a rischio assembramenti.



### 3.3.2 FC3.2: Back-end<sub>G</sub>

- **Descrizione:** sviluppo della parte di comunicazione di informazioni tra server/database e front-end<sub>G</sub>.
- **Strumenti:** si utilizzerà Java<sub>G</sub>.



## Requisiti

In questa sezione vengono illustrati attraverso una tabella tutti i requisiti<sub>g</sub> individuati dal proponente<sub>g</sub> e dal gruppo Jawa Druids. Ogni requisito viene individuato da un codice identificativo, una sua descrizione, la tipologia di requisito e la fonte di riferimento, la spiegazione di ogni parte è descritta nel documento *Norme del Progetto v1.0.0*. Nella sezione successiva viene illustrato attraverso una tabella il tracciamento dei requisiti alla loro fonte e viceversa.

### 4.1 Requisiti funzionali

Codice RS	Descrizione	Tipo di requisito	Fonte
RSFO1	Realizzazione di motori software 'contapersone'	Obbligatorio	Capitolato <sub>g</sub> V. esterno 17-12-2020 FC1.1
RSFF2	Realizzazione di simulatori di altre sorgenti dati sia dei dati storici/in monitoraggio che dati previsionali	Facoltativo	Capitolato <sub>g</sub> FC1.1
RSFO3	Il sistema deve visualizzare un messaggio d'errore se il flusso di dati esterno viene a mancare	Obbligatorio	Interno FC1.1
RSFO4.1	Archiviazione di tutti i dati acquisiti nel database	Obbligatorio	Capitolato <sub>g</sub> FC1.2
RSFO4.2	Archiviazione di tutti i dati elaborati nel database	Obbligatorio	Capitolato <sub>g</sub> FC1.2
RSFO5	Elaborazione in tempo reale dei dati acquisiti da flussi esterni	Obbligatorio	Capitolato <sub>g</sub>



RSFO5.1	Identificazione di eventi che portano alla variazione del flusso di utenti	Obbligatorio	Capitolato <sub>g</sub>
RSFD6	Previsione dell'insorgenza futura di variazioni significative di flussi di persone	Desiderabile	Capitolato <sub>g</sub> FC2
RSFO7	Visualizzazione dei dati elaborati attraverso heat map <sub>g</sub>	Obbligatorio	Capitolato <sub>g</sub> FC3.1
RSFO8	Apache Kafka <sub>g</sub> deve poter comunicare con il database, l'applicazione web e il modello di Machine Learning <sub>g</sub>	Obbligatorio	Interno FC1.3

Tabella 4.1: Requisiti funzionali

## 4.2 Requisiti prestazionali

Codice RS	Descrizione	Tipo di requisito	Fonte
RSPO1	Capacità di acquisizione continuativa nel tempo dei dati da flussi esterni	Obbligatorio	Capitolato <sub>g</sub>
RSPO2	Modalità a bassa latenza dell'acquisizione di informazioni	Obbligatorio	Capitolato <sub>g</sub>



RSQO1	La progettazione e la codifica dei requisiti devono rispettare le norme e le metriche definite nel documento <i>Norme di Progetto v1.0.0</i>	Obbligatorio	Interno
RSQF2	Il codice sorgente del software deve essere disponibile in una repository <sub>G</sub> pubblica su Github <sub>G</sub>	Facoltativo	Interno
RSQF3	Deve essere sviluppato e fornito un documento con lo schema della base di dati relazionale	Facoltativo	Interno FC1.2
RSQF4	Deve essere realizzato un documento contenente tutti gli errori risolti durante la realizzazione del software	Facoltativo	Interno
RSQO5	Test che dimostrino il corretto funzionamento dei servizi e delle funzionalità previste	Obbligatorio	Capitolato <sub>G</sub>

Tabella 4.2: Requisiti di qualità



## 4.3 Requisiti di vincolo

Codice RS	Descrizione	Tipo di requisito	Fonte
RSVO1	I dati acquisiti da telecamere in tempo reale devono avere data di riferimento associato	Obbligatorio	Interno
RSVO1.1	I dati acquisiti da telecamere in tempo reale devono avere un orario di riferimento associato	Obbligatorio	Interno
RSVO1.2	I dati acquisiti da telecamere in tempo reale devono avere un luogo di riferimento associato	Obbligatorio	Interno
RSVO2	Il front-end <sub>g</sub> del prodotto viene sviluppato utilizzando tecnologie web	Obbligatorio	Capitolato <sub>g</sub> FC3.1
RSVF2.1	Utilizzo di leaflet.js <sub>g</sub> per la creazione di heat map <sub>g</sub>	Facoltativo	Capitolato <sub>g</sub> FC3.1
RSVO2.2	Utilizzo di angular.js <sub>g</sub> per la creazione della wep-app <sub>g</sub>	Obbligatorio	Capitolato <sub>g</sub> FC3.1
RSVO3	Il sistema deve far uso dell'ecosistema Apache Kafka <sub>g</sub>	Obbligatorio	Capitolato FC1.3
RSVO4	Il back end <sub>g</sub> del prodotto viene sviluppato utilizzando il linguaggio Java <sub>g</sub>	Obbligatorio	Capitolato <sub>g</sub> FC3.2

Tabella 4.3: Requisiti di vincolo



## 4.4 Tracciamento dei requisiti

### 4.4.1 Requisito - fonte

Codice RS	Fonte
RSFO1	Capitolato <sub>g</sub> V. esterno 17-12-2020 FC1.1
RSFF2	Capitolato <sub>g</sub> FC1.1
RSFO3	Interno FC1.1
RSFO4.1	Capitolato <sub>g</sub> FC1.2
RSFO4.2	Capitolato <sub>g</sub> FC1.2
RSFO5	Capitolato <sub>g</sub>
RSFO5.1	Capitolato <sub>g</sub>
RSFD6	Capitolato <sub>g</sub> FC2
RSFO7	Capitolato <sub>g</sub> FC3.1
RSFO8	Interno FC1.3
RSPO1	Capitolato <sub>g</sub>
RSPO2	Capitolato <sub>g</sub>
RSQO1	Interno
RSQF2	Interno
RSQF3	Interno FC1.2
RSQF4	Interno





RSQO5	Capitolato <sub>g</sub>
RSVO1	Interno
RSVO1.1	Interno
RSVO1.2	Interno
RSVO2	Capitolato <sub>g</sub> FC3.1
RSVF2.1	Capitolato <sub>g</sub> FC3.1
RSVO2.2	Capitolato <sub>g</sub> FC3.1
RSVO3	Capitolato <sub>g</sub> FC1.3
RSVO4	Capitolato <sub>g</sub> FC3.2

Tabella 4.4: Tabella tracciamento requisito-fonte



#### 4.4.2 Fonte - requisito

Fonte	Codice RS
Capitolato <sub>c</sub>	RSFO1
	RSFF2
	RSFO4.1
	RSFO4.2
	RSFO5
	RSFO5.1
	RSFD6
	RSFO7
	RSPO1
	RSPO2
	RSQO5
	RSVO2
	RSVF2.1
	RSVO2.2
	RSVO3
FC1.1	RSFO1 RSFF2 RSFO3
FC1.2	RSFO4.1 RSFO4.2 RSQF3
FC1.3	RSFO8 RSVO3
FC2	RSFD6
FC3.1	RSFO7 RSVO2 RSVF2.1 RSVO2.2
FC3.2	RSVO4



Interno	RSFO3 RSFO8 RSQO1 RSQF2 RSQF3 RSQF4 RSVO1 RSVO1.1 RSVO1.2
Verbale esterno 17-12-2020	RSFO1

Tabella 4.5: Tabella tracciamento fonte-requisito

## 4.5 Considerazioni

I requisiti potranno subire delle variazioni in futuro, in modo tale da apportare degli aggiornamenti alle voci presenti o delle migliorie. Nel caso in cui le attività pianificate terminassero prima del previsto e dovessero avanzare delle ore di lavoro, potranno essere presi in carico nuovi requisiti per aggiungere del valore al prodotto. Pertanto, qualsiasi espansione è riservata solo per il futuro.