Allegato Tecnico

Jawa Druids

Versione	1.0.0
Data approvazione	??-??-????
Responsabile	Nome Cognome
Redattori	Nome Cognome Nome Cognome
Verificatori	Nome Cognome Nome Cognome Nome Cognome
Stato	Approvato
Lista distribuzione	Jawa Druids Prof. Tullio Vardanega Prof. Riccardo Cardin
Uso	Esterno

Sommario

Il presente documento contiene le scelte architetturali che il gruppo *Jawa Druids* ha effettuato ai fini realizzativi del progetto. Contiene i design pattern e i diagrammi di attività, sequenza, classi e package.

Indice

Introduzione

1.1 Scopo del documento

Lo scopo del documento è quello di elencare e motivare le scelte architetturali fatte dal gruppo Jawa Druids, per quanto riguarda il progetto GDP: Gathering Detection Platform.

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 quella di poter riuscire a prevedere assembramenti in intervalli futuri di tempo.
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.

Architettura del prodotto

2.1 Descrizione generale

In fase di progettazione, il gruppo *Jawa Druids* ha deciso di suddividere la modellazione architetturale di *Gathering-Detection-Platform* in tre distinti moduli, tutti indipendenti tra loro. Il primo modulo si occupa solamente di leggere, tramite file JSON$_G$, tutte le webcam disponibili per poi effettuare il riconoscimento persone tramite i frame scaricati. Successivamente i dati estrapolati verranno invitati al database. Il secondo modulo, il machine-learning$_G$, si occupa di recuperare questi dati dal database per lavorarli producendo predizioni per le ore future. Infine il terzo modulo, la web-app$_G$ vera e propria, si occuperà di rappresentare graficamente i dati all'interno del database mediante una heat-map$_G$ e farli visualizzare all'utente.

2.2 Architettura Acquisition

L'architettura riguardante il modulo di acquisizione, ovvero il primo modulo del software, è basata sul fatto che è creata sul paradigma della codifica procedurale. Inoltre non presenta alcuna classe in quanto non crea oggetti, crea esclusivamente un array con i dati che estrapola dai frame e dalle informazioni del tempo.

2.3 Architettura Prediction

L'architettura del modulo del machine-learning si può semplificare ad un modulo unico con all'interno i metodi necessari per prelevare dati dal database per poi reinviarli da lavorati. Non necessita classi interne in quanto svolge esclusivamente operazioni funzionali

2.4 Architettura Web-App

Per il modulo relativo al front-end$_G$, si è deciso di utilizzare il pattern *Model-View-Controller*(MVC). Questa scelta è dovuta al fatto che, essendo la web-app sviluppata con spring, il pattern è quello che più si adatta alla tipologia sia di modellazione sia di scopo.