

PROGETTO: IOT FOR EMERGENCY MANAGEMENT

DESCRIZIONE

Questa applicazione si inserisce nell'ambito della gestione delle emergenze (terremoti, incendi). In particolare vuole fornire un aiuto in fase di evacuazione in interno mediante una piattaforma di sensori-attuatori. L'applicazione si compone di due parti in CONDIZIONI DI EMERGENZA:

- in pre-movimento per sollecitare l'evacuazione degli occupanti;
- in movimento per indicare la migliore via da percorrere.

Esso inoltre potrà essere affiancato da un sistema di navigazione in CONDIZIONI ORDINARIE che sia in grado di dirigere l'utente verso una determinata area preselezionata.

L'applicazione deve essere in grado di calcolare il percorso a peso minimo per raggiungere il luogo di raccolta. Il peso minimo di un tratto di strada (arco nel grafo) è determinato da alcuni fattori che dipendono dallo spazio architettonico quali: distanza e presenza di scale. È determinato inoltre da fattori che dipendono dai dati raccolti dai sensori quali: vulnerabilità, presenza di fumo o altri gas tossici, macerie, numero di persone.

REQUISITI NON FUNZIONALI

- R.1. Il sistema è di tipo client-server come riportato in Figura 1
- R.2. La mobile app deve essere sviluppata per piattaforma Android
- R.3. La mobile/web app deve utilizzare l'algoritmo di Dijkstra per calcolare il percorso a peso minimo
- R.4. La mobile app deve essere in grado di funzionare sia in modalità online sia in modalità offline
- R.5. La web app deve essere sviluppata con Zend Framework
- R.6. Il server usa servizi REST
- R.7. Il server usa un DB relazionale per memorizzare la mappa e i dati su posizione persone e parametri per il calcolo dei costi degli archi
- R.8. Il dati inviati e ricevuti dai vari componenti del sistema sono in formato JSON

SCENARI

- S.1. Mobile App: Autenticazione
- S.2. Mobile App: Invio aggiornamento propria posizione sul server
- S.3. Mobile App: Ricezione aggiornamento parametri di costo o allarme
- S.4. Mobile App: Calcolo percorso mediante algoritmo di Dijkstra (online/offline e emergenza/non emergenza)
- S.5. Mobile App: Visualizzazione del percorso (online/offline e emergenza/non emergenza)
- S.6. Web App: Autenticazione di tre classi di utenti (admin, emergency team, user) + guest
- S.7. Web App: Invio aggiornamento parametri di costo o allarme
- S.8. Web App: Ricezione aggiornamento su posizione persone
- S.9. Web App: Calcolo percorso mediante algoritmo di Dijkstra (emergenza/non emergenza)
- S.10. Web App: Visualizzazione del percorso (emergenza/non emergenza)
- S.11. Server: Invio (alla Mobile App) e ricezione (da Web App) aggiornamento parametri e allarme
- S.12. Server: Invio (alla Mobile App) e ricezione (da Web App) aggiornamento posizione persone

ARCHITETTURA

Architettura three-tier: (mobile app o web-app) + server + DBMS

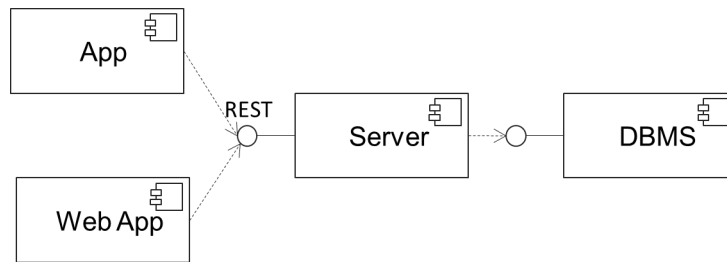


Figura 1 – Architettura del sistema

VALIDITÀ DEL PROGETTO

<i>Corso</i>	<i>Requisiti Non Funzionali</i>	<i>Scenari</i>	<i>Architettura</i>	<i>Appello</i>
<i>IdS</i>	R.1, R.2, R.3, R.4, R.8	S.1, S.2, S.3, S.4, S.5,	App	APRILE '16
<i>IdS+TID</i>	R.1, R.2, R.3, R.4, R.6, R.7, R.8	S.1, S.2, S.3, S.4, S.5, S.11, S.12	App+Server	LUGLIO '16
<i>IdS+PO</i>	R.1, R.2, R.3, R.4, R.6, R.7, R.8	S.1, S.2, S.3, S.4, S.5, S.11, S.12	App+Server	LUGLIO '16
<i>IdS+TWeb</i>	R.1, R.3, R.5, R.6, R.7, R.8	S.6, S.7, S.8, S.9, S.10, S.11, S.12	WebApp+Server	LUGLIO '16
<i>IdS+TID+PO</i>	R.1, R.2, R.3, R.4, R.5, R.6, R.7, R.8	S.1, S.2, S.3, S.4, S.5, S.6, S.7, S.8, S.9, S.10, S.11, S.12	App+WebApp+Server	LUGLIO '16
<i>IdS+TID+TWeb</i>	R.1, R.2, R.3, R.4, R.5, R.6, R.7, R.8	S.1, S.2, S.3, S.4, S.5, S.6, S.7, S.8, S.9, S.10, S.11, S.12	App+WebApp+Server	LUGLIO '16

PIANO E METODO DI LAVORO

<i>Scadenza</i>	<i>Attività</i>
<i>4 Novembre</i>	- Formare i gruppi
<i>11 Novembre</i>	- Intervista con stakeholders per raccogliere i requisiti (user stories)
<i>18 Novembre</i>	- Piano di lavoro
	- Business Plan Game
<i>25 Novembre</i>	- 1 st Iteration Planning Game (Parte A)
<i>Ogni 2/3 settimane</i>	- Nuovo Iteration Planning Game (Parte A)
<i>29 Gennaio</i>	- Fine analisi requisiti (fine planning game parte A)
	- 1 st Iteration Planning Game (Parte B)
<i>Ogni 2/3 settimane</i>	- Nuovo Iteration Planning Game (Parte B)
<i>Aprile</i>	- Release finale per gruppi IdS
<i>Luglio</i>	- Release finale per tutti gli altri gruppi

Parte A

- User Stories
- Task Cards
- CRC
- Presentation, Navigation, Dialogue Model
- Piano dei Test di correttezza e di usabilità
- Diagramma delle classi di analisi
- Diagramma delle sequenze, stato, e attività di analisi

Parte B

- Task Cards
- Diagramma dei componenti e deployment
- Classi di progettazione + schema DB + ORM
- Codice
- Metriche coesione, accoppiamento, complessità
- Esecuzione dei Test

Release

- Codice
- Manuale utente + installazione
- Relazione

Requisiti Non Funzionali

- Usare Bitbucket

Valutazione Progetti

- Verrà usato lo schema Excel che trovate sul sito
- Verranno tolti due punti per ogni mese di ritardo (senza eccezioni)