**Università degli Studi di Salerno**

*Corso di Basi di Dati e Sistemi Informativi su Rete*

Requirement Analysis Document

**ArcheoTour**

Partecipanti:

Rosa Maria Gaetano 0510201640

Salvatore Merola 0510201650

# Indice

# Introduzione

# Requisiti funzionali

# Requisiti non funzionali

# Modelli del sistema

## 4.1 Attori

## 4.2 Modello dei casi d'uso (Diagramma e sviluppo dei singoli casi d'uso)

## 4.3 Diagramma delle Classi

## 4.4 Diagrammi di sequenza

# Diagramma Navigazionale e Mocks up delle interfacce

### 2. Requisiti funzionali

### 2.1 Descrizione del sito

Funzionalità che recupera da un database e visualizza una breve descrizione del sito di interesse.

### 2.2 Ricerca alberghi e ristoranti

Funzionalità che si connette ai principali motori di ricerca e restituisce l’elenco degli alberghi e dei ristoranti vicini al sito di interesse

### 2.3 Orari di apertura e tariffe

La funzionalità recupera da un database gli orari di apertura dei siti archeologici e le tariffe di ingresso agli stessi e le visualizza.

### 2.4 News

La funzionalità recupera alcuni feed RSS relativi al sito e li visualizza.

### 2.5 Mappe e GPS

La funzionalità permette all’utente di visualizzare la mappa del sito. Se richiesto,può anche offrire un servizio di geolocalizzazione all’interno del sito.

### 2.6 Punti di interesse

La funzionalità permette di individuare alcuni punti di interesse vicini al sito.

### 2.7 Percorsi

Funzionalità che recupera da un database i relativi percorsi possibili all’interno del sito e li visualizza.

### 2.8 Meteo

Funzionalità che permette di visualizzare il meteo alla località del sito utilizzando le API di Yahoo meteo

### 3. Requisiti non funzionali

L’applicazione dovrà rispettare i seguenti requisiti non funzionali

### 3.1 Usabilità

L’applicazione è destinata ad utenti che non hanno particolari conoscenze informatiche.

Per l’utente finale deve essere estremamente facile accedere al sistema, tramite un’interfaccia grafica intuitiva, ed avere la possibilità di usufruire di tutte le funzionalità che ArcheoTour mette a disposizione.

Deve essere possibile utilizzare il sistema senza la lettura di istruzioni preliminari.

Inoltre, per favorire un utilizzo semplice, deve essere possibile, agli utenti, raggiungere tutte le  funzionalità in meno di 5 passi,

### 3.2 Affidabilità

Il sistema è basato su un server di database relazionali.

Deve essere affidabile e poter mantenere i dati anche in caso di guasti (di natura elettrica, usura hardware, attacchi alla sicurezza).

Deve essere possibile in caso di guasti poter cambiare server in maniera rapida per ridurre i tempi di inutilizzabilità dell’app al minimo.

Deve essere possibile pianificare dei backup periodici del database.

Deve essere inoltre necessario garantire la consistenza dei dati, cioè fare in modo che ogni utente acceda a dati sempre aggiornati.

Inoltre, in caso di errore, questo verrà segnalato all’utente tramite un opportuno messaggio.

### 3.3 Performance (Prestazioni)

Tempo di risposta: Il tempo di caricamento dell’app e i tempi di risposta alle altre richieste dell’utente devono essere sufficientemente bassi.

Throughput: L’utente può effettuare molteplici operazioni in poco tempo.

Disponibilità: L’utente può accedere all’applicazione ed usufruire delle sue funzionalità ventiquattro ore su ventiquattro.

### 3.4 Supportabilità

Il sistema deve essere facilmente aggiornabile in seguito a cambiamenti del dominio di applicazione e a cambiamenti dovuti all’introduzione di nuove tecnologie. Dovrà dunque  essere suddiviso in moduli, in modo da rendere semplice l’aggiunta o l’eliminazione di una funzionalità.

### 3.5 Implementazione

Il sistema verrà implementato tramite una base di dati realizzata con il DBMS Mysql, dove saranno immagazzinate le informazioni riguardanti i siti archeologici.

Il sistema verrà realizzato utilizzando l’SDK messo a disposizione su Google utilizzando il linguaggio di programmazione java.

Per il trasferimento dei dati dal server al client verrà utilizzato JSON

Sarà quindi possibile utilizzare il sistema in tutti i dispositivi dotati di sistema operativo Android con *versione minima 2.3.3*

### 3.6 Packaging (vincoli sulla consegna reale del sistema)

L’applicazione verrà distribuita tramite apk disponibile su internet.

### 3.7 Operatività

L’applicazione ArcheoTour deve essere operativa nel 90% dei casi, salvo problemi legati ad un’improvvisa mancanza dell’ alimentazione al sistema o ad una interruzione dell’accesso ad internet dovuta a cause esterne.

### 3.8 Persistenza

Il sistema dovrà permettere di conservare i dati, finchè lo si ritiene necessario.

### 3.9. Tecnologia

La scelta è orientata ad un'applicazione  Android, è quindi richiesto tale sistema operativo per l’utilizzo della stessa. L'applicazione si appoggerà ad una base di dati residente su un server remoto per la conservazione dei dati relativi i siti archeologici e al server SQLite integrato in Android per la gestione delle impostazioni dell’applicazione.

### 3.10. Sicurezza

Nessuna richiesta particolare di sicurezza in quanto non sono trattati dati sensibili ne è richiesta autenticazione

**4 Modelli del Sistema**

### 4.1 Attori

*Amministratore*: è colui che controlla l’elenco dei siti archeologici. Ha la possibilità di aggiungere, modificare o rimuovere un sito dall’elenco. Può aggiungere definizioni delle mappe, tariffe, orari, percorsi.

*Utente generico*: qualunque utente interessato all’applicazione che può accedervi per usufruire delle varie funzionalità.

### 4.2 Modelli di casi d’uso.

### 4.2.1 Visualizzazione descrizione sito

**Attori**: Utente generico

**Exit condition**: viene lanciata una activity che visualizza una breve descrizione con foto del sito.

**Rilevanza**: Media

**Frequenza stimata**: 5 usi/visita

**Flusso di eventi**

1. Utente: richiede all’app la visualizzazione della descrizione del sito.
2. Sistema: lancia l’activity che visualizza una breve descrizione con foto del sito.

### 4.2.2 Visualizzazione orari e tariffe

**Attori**: Utente generico

**Exit condition**: viene lanciata una activity che visualizza orari di apertura e tariffe.

**Rilevanza**: Media

**Frequenza stimata**: 1 uso/visita

**Flusso di eventi**

1. Utente: richiede all’app la visualizzazione di orari e tariffe
2. Sistema lancia l’activity che visualizza le seguenti informazioni:
   * + - * Orari di apertura
         * Tariffe relative alle varie tipologie di percorso

### 4.2.3 Visualizzazione news

**Attori**: Utente generico

**Exit condition**: viene lanciata una activity che visualizza le news.

**Rilevanza**: Media

**Frequenza stimata**: 5 usi/visita

**Flusso di eventi**

1. Utente: richiede all’app la visualizzazione di news riguardanti il sito di

interesse

1. Sistema lancia l’activity si connette ad un servizio di feed RSS e visualizza le

ultime notizie relative a tutto ciò che interessa il sito.

Nella stessa activity si connette al servizio Yahoo Meteo e visualizza la situazione meteo al sito di interesse.

### 4.2.4 Visualizzazione mappa GPS

**Attori**: Utente generico

**Exit condition**: viene lanciata una activity che visualizza una mappa GPS.

**Rilevanza**: Media

**Frequenza stimata**: 10 usi/visita

**Flusso di eventi**

1. Utente: richiede all’app la visualizzazione della mappa.
2. Sistema lancia l’activity che rileva la posizione attuale tramite l’antenna GPS e utilizza la posizione rilevata per mostrare sulla mappa:
   * I Punti Di Interesse nelle vicinanze.
   * La posizione dell’utente all’interno del sito.

**4 Modelli del Sistema**