Francesca Santucci  
Matricola: 278531

|  |
| --- |
| TOP ON CONCERT |
| toponconcert.com |
|  |
| Università degli studi di Perugia  Dipartimento di Matematica e Informatica  Progetto di Sistemi Aperti e Distribuiti |
|  |

TOP ON CONCERT

toponconcert.com

# INTERVISTA AL COMMITTENTE

• • •

# Intervista al committente

Ci viene richiesto di realizzare un Web Service che ci consenta di gestire la prenotazione (vendita e acquisto) di biglietti per dei concerti.

Partendo quindi da un’analisi dei requisiti andiamo a distinguere tre tipi principali di utenti del servizio:

* **Cliente:** è l’utente finale, colui che è interessato a visualizzare i concerti ed acquistare il relativo biglietto.  
  La sua interfaccia sarà costituita da una tabella all’interno della quale saranno presenti tutte le informazioni relative ai concerti disponibili nonché le funzionalità che gli consento di acquistare i biglietti degli stessi.
* **Organizzatore:** è l’organizzatore del concerto.

All’organizzatore vengono messe a disposizione una serie di funzionalità che gli consento di gestire (creare, modificare, eliminare) i vari concerti e il relativo numero di biglietti disponibili.

* **Admin: è** l’amministratore dell’intero servizio.

La sua interfaccia è costituita da una serie di tabelle all’interno della quale sono salvate le informazioni relative agli utenti registrati (organizzatori e clienti) e i dettagli dei concerti.

Ogni Cliente e Organizzatore prima di poter usufruire delle varie funzionalità deve registrarsi al Web Service e successivamente autenticarsi.

È importante che i vari utenti abbiano interfacce distinte ed opportunamente sicure.

Ci viene richiesto inoltre che il Web Service sia disponibile solo in lingua italiana.

# ANALISI DEL TEAM DI SVILUPPO

• • •

TOP ON CONCERT

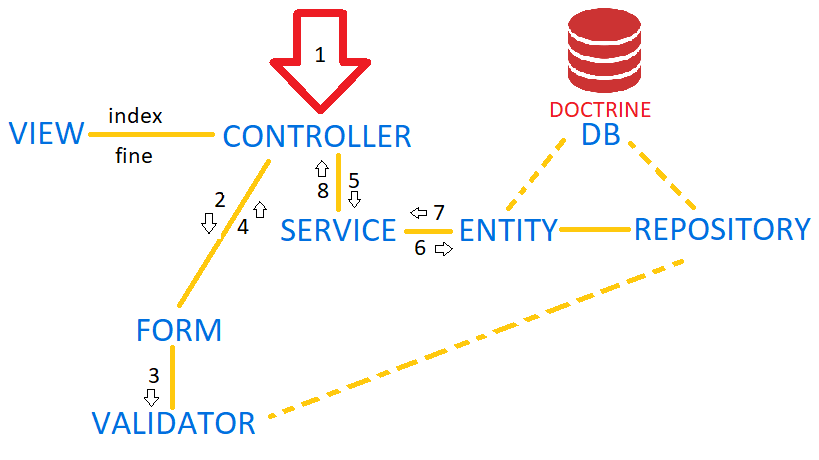
toponconcert.com

# Analisi del team di sviluppo: back-end

Per realizzare il back-end del Web Service richiesto dal committente abbiamo deciso di utilizzare Zend Framework 3, un’architettura logica di supporto per lo sviluppo di applicazioni web scritta in linguaggi PHP.

Zend Framework si basa su una serie di design-pattern tra cui troviamo MVC (Model-View-Controller)tanto che la struttura del nostro progetto rifletterà proprio quest’ultimo.

Di seguito mostriamo come sono interconnesse tutte le varie componenti del nostro progetto e a seguire ne diamo una breve descrizione:



TOP ON CONCERT

toponconcert.com

# ANALISI DEL TEAM DI SVILUPPO

• • •

1. View: visualizza i dati contenuti nel Model e si occupa dell’interazione con gli utenti.
2. Controller : riceve i comandi dell'utente (in genere attraverso la view) e li esegue richiamando (ed eventualmente modificando) le altre componenti.
3. Form: usato per collezionare i dati inseriti dall’utente.
4. Validator: sono usati per verificare la correttezza dei dati inseriti.
5. Service: incapsulano alcune funzionalità logiche della nostra applicazione. Nel nostro caso abbiamo userManager e concertManager.
6. Entity: è una delle componenti fondamentali di Doctrine.

È destinata alla memorizzazione di alcuni dati.

Nel nostro progetto le entità saranno: cunsomer, organizer, concert e ticket.

1. Repository: è una componente di Doctrine. Responsabile del salvataggio e recupero delle entità.
2. Database: è l’archivio dei nostri dati.

In altre parole supponiamo che l’utente debba registrarsi.  
Attraverso la View, che contiene una serie di elementi interattivi, l’utente richiamerà il Controller che gli mostrerà il Form da compilare per la registrazione. Una volta fatto, l’utente invierà tutti i dati inseriti di nuovo al Controller che, tramite la configurazione del Form e dei vari Validator, verificherà la correttezza dei campi inseriti. Se questi risultano essere validi allora il Controller li assegnerà allo userManager (Service) che genererà un’ Entity e tramite le funzioni persist() e flush(), le salverà all’interno del Dababase. Nel caso in cui invece l’utente ha bisogno di visualizzare la lista dei biglietti da lui acquistati entrerà in gioco anche la Repository che ci consente di recuperare una o più entità dal Database.

È importante notare che, nel nostro caso, abbiamo deciso di sviluppare il Database tramite Mysql e usufruiremo inoltre di Doctrine che ci semplificherà di molto l’esecuzione delle varie query alle tabelle del Database. Doctrine infatti è un insieme di librerie [PHP](https://it.wikipedia.org/wiki/PHP)  focalizzato sul fornire servizi di persistenza e funzionalità correlate. Il suo progetto principale è un livello di astrazione del database su cui è costruito.

TOP ON CONCERT

toponconcert.com

# ANALISI DEL TEAM DI SVILUPPO

• • •

# Analisi del team di sviluppo: front-end

Per quanto riguarda il front-end abbiamo deciso di svilupparlo sfruttando Ionic Framework.   
Ionic Framework è uno strumento Open Source per lo sviluppo di applicazioni mobili ibride basato su Cordova e Angular.

Per quanto riguarda le comunicazioni con il Database, queste avverranno tramite l’utilizzo di REST.

TOP ON CONCERT

toponconcert.com

# STRUTTURA DEL DATABASE

• • •

# Struttura del nostro database:

TOP ON CONCERT

toponconcert.com

# BACK-END

• • •

# Elementi importanti del back-end

TOP ON CONCERT

toponconcert.com

# INTERFACCIA UTENTE

• • •

# Interfaccia utente

TOP ON CONCERT

toponconcert.com

# REST

• • •

# Chiamata REST: