

# QUIZZIPEDIA



[team404swe@gmail.com](mailto:team404swe@gmail.com)

## Specifica Tecnica 1.0

Informazioni sul documento	
Nome Documento	Specifica Tecnica 1.0
Versione	1.0
Uso	Esterno
Data Creazione	22 aprile 2016
Data Ultima Modifica	9 maggio 2016
Redazione	M. Crivellaro - L. Alessio
Verifica	Alex Beccaro
Approvazione	Martin V. Mbouenda
Committente	Zucchetti SPA
Lista di distribuzione	Prof. Vardanega Tullio TEAM404

## Registro delle modifiche

Versione	Autore	Data	Descrizione
0.0.5	Luca Alessio (Progettista)	26/04/2016	Stesura parziale sezione "Tecnologie e strumenti utilizzati"
0.0.4	Davide Bortot (Progettista)	25/04/2016	Ampliata la sezione "Architettura generale del sistema".
0.0.3	Luca Alessio (Progettista)	24/04/2016	Stesura parziale sezione "Architettura generale del sistema"
0.0.2	Davide Bortot (Progettista)	23/04/2016	Strutturazione iniziale del documento e stesura sezione introduttiva
0.0.1	Marco Crivellaro (Progettista)	22/04/2016	Creazione documento.

## Indice

<b>1</b>	<b>Introduzione</b>	<b>2</b>
1.1	Scopo del documento . . . . .	2
1.2	Scopo del prodotto . . . . .	2
1.3	Glossario . . . . .	2
1.4	Riferimenti . . . . .	2
1.4.1	Normativi . . . . .	2
1.4.2	Informativi . . . . .	2
<b>2</b>	<b>Specifiche del prodotto</b>	<b>3</b>
2.1	Architettura generale del sistema . . . . .	3
2.2	Descrizione del componente Model . . . . .	3
2.3	Descrizione del componente View . . . . .	3
2.4	Descrizione del componente Controller . . . . .	3
<b>3</b>	<b>Tecnologie e strumenti utilizzati</b>	<b>3</b>
3.1	HTML5 . . . . .	3
3.2	CSS3 . . . . .	4
3.3	Javascript . . . . .	4
3.4	Node.js . . . . .	4
3.5	PostgreSQL . . . . .	5
<b>4</b>	<b>Diagrammi dei packages</b>	<b>5</b>
4.1	Package della componente Model . . . . .	5
4.2	Package della componente View . . . . .	5
4.3	Package della componente Controller . . . . .	5
<b>5</b>	<b>Diagrammi delle classi</b>	<b>5</b>
5.1	Diagrammi delle classi del Model . . . . .	5
5.2	Diagrammi delle classi del View . . . . .	5
5.3	Diagrammi delle classi del Controller . . . . .	5
<b>6</b>	<b>Diagrammi di attività</b>	<b>5</b>
<b>7</b>	<b>Diagrammi di sequenza</b>	<b>5</b>
<b>8</b>	<b>Tracciamento</b>	<b>5</b>
<b>9</b>	<b>Analisi di fattibilità</b>	<b>5</b>

**Elenco delle figure**

**Elenco delle tabelle**

## Sommario

# 1 Introduzione

## 1.1 Scopo del documento

## 1.2 Scopo del prodotto

Il progetto **Quizzipedia** ha come obiettivo lo sviluppo di un sistema software basato su tecnologie Web (Javascript<sub>6</sub>, Node.js<sub>6</sub>, HTML5<sub>6</sub>, CSS3<sub>6</sub>) che permetta la creazione, gestione e fruizione di questionari. Il sistema dovrà quindi poter archiviare i questionari suddivisi per argomento, le cui domande dovranno essere raccolte attraverso uno specifico linguaggio di markup (Quiz Markup Language) d'ora in poi denominato QML<sub>6</sub>. In un caso d'uso a titolo esemplificativo, un "esaminatore" dovrà poter costruire il proprio questionario scegliendo tra le domande archiviate, ed il questionario così composto sarà presentato e fruibile all' "esaminando", traducendo l'oggetto QML in una pagina HTML<sub>6</sub>, tramite un'apposita interfaccia web. Il sistema presentato dovrà inoltre poter proporre questionari preconfezionati e valutare le risposte fornite dall'utente finale.

Per un'analisi più precisa ed approfondita del progetto si rimanda al documento "*analisi\_dei\_requisiti\_2.0.pdf*".

## 1.3 Glossario

Viene allegato un glossario nel file "*glossario\_2.0.pdf*" nel quale viene data una definizione a tutti i termini che in questo documento appaiono con il simbolo '₆' a pedice.

## 1.4 Riferimenti

### 1.4.1 Normativi

- Capitolato d'appalto Quizzipedia:  
<http://www.math.unipd.it/~tullio/IS-1/2015/Progetto/C5.pdf>
- Norme di Progetto: "*norme\_di\_progetto\_2.0.pdf*"

### 1.4.2 Informativi

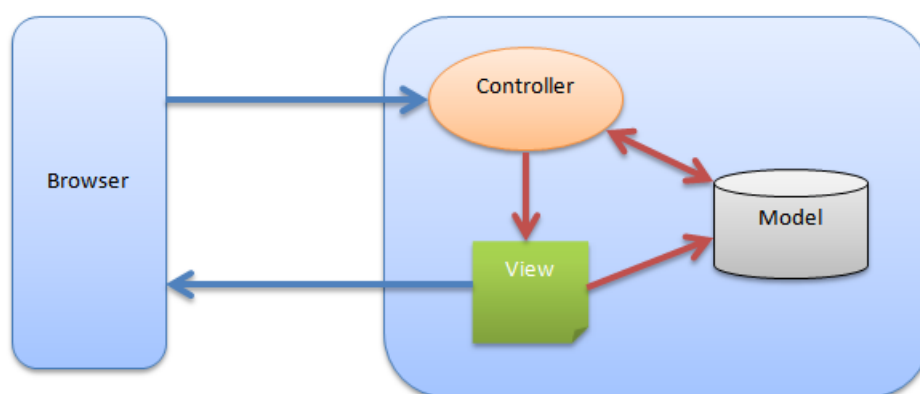
- Corso di Ingegneria del Software anno 2015/2016:  
<http://www.math.unipd.it/~tullio/IS-1/2015/>
- Regole del progetto didattico:  
<http://www.math.unipd.it/~tullio/IS-1/2015/Dispense/PD01.pdf> <http://www.math.unipd.it/~tullio/IS-1/2015/Progetto/>

## 2 Specifiche del prodotto

I contenuti della specifica saranno presentati seguendo l'approccio top-down: dalla descrizione macroscopica del sistema si scenderà sempre più in dettaglio passando alla descrizione delle singole componenti. Verranno inoltre descritti i design pattern utilizzati e come essi sono stati applicati. Per agevolare la comprensione si è scelto di utilizzare i diagrammi dei package, delle classi, di attività e di sequenza, descritti attraverso lo standard UML 2.0.

### 2.1 Architettura generale del sistema

Quizzipedia viene identificato come (server based web-service); l'utente interagisce col sistema tramite interfaccia web su browser mentre i dati vengono organizzati su server esterno. La base di dati raccoglie principalmente quesiti memorizzati in QML, che su richiesta verranno elaborati da un interprete. Il suddetto interprete processa l'input QML, precedentemente validato nella forma da un parser apposito, traducendolo in linguaggio HTML5 visualizzabile da browser. Nella realizzazione del sistema Quizzipedia verrà adottato il design pattern Model View Controller.



- **Model:** definisce l'organizzazione dei dati e ne specifica le modalità di accesso. Nel sistema Quizzipedia è costituito dalla parte server che opera sulla sottostante la base di dati. Fanno parte del Model anche le componenti *Parser* e *Interprete* che operano sui dati del DBMS.
- **View:** rappresenta l'interfaccia grafica presentata all'utilizzatore, la quale visualizza i dati e cattura le interazioni dell'utente.
- **Controller:** inoltra le richieste della View al Model

### 2.2 Descrizione del componente Model

### 2.3 Descrizione del componente View

### 2.4 Descrizione del componente Controller

## 3 Tecnologie e strumenti utilizzati

### 3.1 HTML5

Linguaggio di markup per la progettazione di pagine Web. Richiesto espressamente nel capitolato per la creazione dell'interfaccia utente (mi pare, devo controllare).

- **Utilizzo:**
- **Vantaggi:** favorisce la portabilità su diversi dispositivi (desktop, smartphone, tablet...) e browser. Sono punti a favore anche l'elevata compatibilità con tecnologie quali CSS3 e Javascript.
- **Svantaggi:** essendo HTML5 un linguaggio non ancora standard il rischio nel suo utilizzo è l'instabilità dei tag utilizzati. Un tag oggi accettato potrebbe essere modificato in un futuro prossimo, rendendo la visualizzazione dell'interfaccia utente dipendente dal stabilità dei tag utilizzati.

### 3.2 CSS3

- **Utilizzo:**
- **Vantaggi:**
- **Svantaggi:**

### 3.3 Javascript

E' un linguaggio di scripting debolmente orientato agli oggetti, utilizzato nelle applicazioni Web. Viene interpretato all'interno del browser. Permette di definire funzionalità simili a quelle offerte da C++ e Java, quali cicli e strutture di controllo. Viene solitamente affiancato a pagine statiche HTML per poter gestire i contenuti dinamicamente, offrendo funzionalità che il linguaggio di markup non può offrire.

- **Utilizzo:** nel sistema Quizzipedia Javascript rivestirà un ruolo importante, verrà infatti utilizzato nella realizzazione del parser/interprete/boh andrea dimmi te.
- **Vantaggi:** eseguito lato Client (sul browser) non sovraccarica il Server per l'esecuzione di richieste, anche se lo script è complesso.
- **Svantaggi:** per script sorgenti molto corposi, può risultare oneroso in termini di tempo lo scaricamento dei contenuti. Deve, inoltre, fare affidamento ad un linguaggio che possa fisicamente effettuare transazioni di dati quando lo script esegue operazioni su oggetti remoti (eg: database). E' altresì un linguaggio non tipizzato, quindi occorre porre attenzione ai tipi delle variabili che non sono dichiarati, ma variano dinamicamente.
- **Variabili e oggetti:** le variabili se sono dichiarate all'interno di una funzione sono visibili solo all'interno di essa; se sono invece esterne sono globali. Vengono dichiarate con la keyword `var` o semplicemente assegnando loro un valore. Ogni elemento in JavaScript è un tipo primitivo o un oggetto. Gli oggetti sono entità dotate di unicità (sono uguali solo a sé stessi) e identificabili con vettori associativi, che associano nomi di proprietà a valori.

### 3.4 Node.js

- **Utilizzo:**
- **Vantaggi:**
- **Svantaggi:**



### 3.5 PostGreSQL

- Utilizzo:
- Vantaggi:
- Svantaggi:

## 4 Diagrammi dei packages

### 4.1 Package della componente Model

### 4.2 Package della componente View

### 4.3 Package della componente Controller

## 5 Diagrammi delle classi

### 5.1 Diagrammi delle classi del Model

### 5.2 Diagrammi delle classi del View

### 5.3 Diagrammi delle classi del Controller

## 6 Diagrammi di attività

## 7 Diagrammi di sequenza

## 8 Tracciamento

## 9 Analisi di fattibilità