

# 404NotFound

Premi: better than Prezi.



## Specifica Tecnica

<b>Versione</b>	1.0
<b>Redazione</b>	Vegro Federico Cossu Mattia Camborata Marco Manuto Monica Rettore Andrea Gobbo Ismaele De Lazzari Enrico
<b>Verifica</b>	Manuto Monica Rettore Andrea
<b>Responsabile</b>	Gobbo Ismaele
<b>Uso</b>	Esterno
<b>Stato</b>	Formale
<b>Ultima modifica</b>	XXX
<b>Lista di distribuzione</b>	404NotFound prof. Tullio Vardanega prof. Riccardo Cardin Zucchetti S.p.a.

## Registro delle modifiche

Versione	Autore	Data	Descrizione
1.0	Autore vers 1.0	12-01-2015	Scrittura versione finale ecc.
0.1	Autore vers 0.1	19-12-2014	Stesura scheletro ecc.

Tabella 1: Storico versioni del documento.

## Indice

<b>1</b>	<b>Introduzione</b>	<b>4</b>
1.1	Scopo del documento . . . . .	4
1.2	Scopo del prodotto . . . . .	4
1.3	Glossario . . . . .	4
<b>2</b>	<b>Definizione del Prodotto</b>	<b>5</b>
2.1	Metodo e formalismo di specifica . . . . .	5
2.2	Presentazione dell'architettura generale del sistema . . . . .	5
<b>3</b>	<b>Diagrammi dei Package</b>	<b>5</b>
<b>4</b>	<b>Descrizione dei singoli componenti</b>	<b>6</b>
4.1	Premi . . . . .	6
4.1.1	Premi.Utility . . . . .	6
<b>5</b>	<b>Diagrammi delle attività</b>	<b>7</b>
<b>6</b>	<b>Stime di fattibilità e di bisogno di risorse</b>	<b>7</b>
<b>7</b>	<b>Tracciamento della relazione componenti - requisiti</b>	<b>7</b>
<b>8</b>	<b>Descrizione delle tecnologie e dei design pattern utilizzati</b>	<b>7</b>
<b>9</b>	<b>Mockup Interfaccia [FACOLTATIVO]</b>	<b>7</b>

## Elenco delle tabelle

1	Storico versioni del documento. . . . .	1
---	---	---

## Elenco delle figure

# 1 Introduzione

## 1.1 Scopo del documento

Questo documento definisce la progettazione ad alto livello di Premi. Viene prima descritta la struttura generale del sistema e successivamente vengono analizzate le varie componenti software in relazione alle loro attività principali. Segue poi la descrizione delle tecnologie e dei Design Pattern<sub>G</sub> utilizzati, e un mockup<sub>G</sub> dell'interfaccia grafica lato utente.

## 1.2 Scopo del prodotto

Lo scopo del progetto è la realizzazione di un software di presentazione di slide non basato sul modello di PowerPoint<sub>G</sub>, sviluppato in tecnologia HTML5<sub>G</sub> e che funzioni sia su desktop che su dispositivo mobile. Il software dovrà permettere la creazione da parte dell'autore e la successiva presentazione del lavoro, fornendo effetti grafici di supporto allo storytelling e alla creazione di mappe mentali.

## 1.3 Glossario

Al fine di evitare ogni ambiguità relativa al linguaggio e ai termini utilizzati nei documenti formali tutti i termini e gli acronimi presenti nel seguente documento che necessitano di definizione saranno seguiti da una "G" in pedice e saranno riportati in un documento esterno denominato Glossario.pdf. Tale documento accompagna e completa il presente e consiste in un listato ordinato di termini e acronimi con le rispettive definizioni e spiegazioni.

## 2 Definizione del Prodotto

### 2.1 Metodo e formalismo di specifica

Verrà qui esposta l'architettura di Premi ad alto livello seguendo un approccio top-down<sub>G</sub>: verranno prima descritti i package<sub>G</sub> e le loro dipendenze e successivamente le singole classi contenute al loro interno. I diagrammi delle classi e dei package<sub>G</sub> seguono il formalismo UML<sub>G</sub>2.0 e la struttura dei package segue una prassi (best practice<sub>G</sub>) di AngularJS<sub>G</sub> che propone una suddivisione dei componenti per funzionalità dell'applicazione in alternativa alla classica suddivisione Model-View-Controller<sub>G</sub>, più difficile da mantenere per applicazioni di medie o grandi dimensioni. Per ulteriori approfondimenti consultare la guida al sito [scotch.io](http://scotch.io) oppure il tutorial di [urigo:angular-meteor](http://urigo.angular-meteor.com). Si illustreranno poi i Design Pattern utilizzati nella fase di progettazione ad alto livello e si descriveranno le interazioni dell'utente con l'applicazione attraverso i diagrammi di attività<sub>G</sub>.

### 2.2 Presentazione dell'architettura generale del sistema

I componenti sono stati suddivisi prima in base al loro contributo a specifiche funzionalità del software e solo successivamente per appartenenza ai ruoli del pattern MVC<sub>G</sub>. Questo aumenta la chiarezza espositiva dei diagrammi, evita la creazione di package<sub>G</sub> contenenti un numero eccessivo di classi e aiuta a compiere verifiche mirate a singoli componenti.

È importante specificare che il framework AngularJS unisce view e controller attraverso una dichiarazione esterna a entrambi, che fa parte del meccanismo detto di *routing* o di reindirizzamento dell'utente; view e controller inoltre non fanno di essere collegati tra loro e comunicano attraverso un oggetto chiamato *\$scope*. Questo rende l'architettura sia di tipo Model-View-Controller<sub>G</sub> che di tipo Model-View-ViewModel<sub>G</sub>. Per motivi di leggibilità *\$scope* e *routing* non verranno rappresentati in modo esplicito nei diagrammi dei package e delle classi di questo documento, ma sono comunque da considerarsi impliciti nelle dipendenze tra i view e controller dei componenti.

## 3 Diagrammi dei Package

Di seguito vengono descritti componenti principali del sistema e le loro dipendenze.

\_\_\_\_\_pkg.jpg

L'applicazione è costituita da un solo package<sub>G</sub> principale chiamato **Premi**; al suo interno sono presenti:

- **Premi** racchiude tutti i package<sub>G</sub> e la view e il controller principali;
- **Premi.UserManager** è il package<sub>G</sub> di gestione dei dati dell'utente;
- **Premi.Viewer** racchiude gli elementi necessari alla visualizzazione della presentazione nei vari contesti previsti;
- **Premi.Presentation** racchiude la struttura generale della presentazione;

- `Premi.PresentationManager` contiene gli elementi necessari alla gestione delle presentazioni da parte dell'utente;
- `Premi.Editor` è il package<sub>G</sub> dedicato alla modifica interna delle presentazioni; possiede al suo interno tre ulteriori package<sub>G</sub>:
  - `Premi.Editor.FrameEditor` si occupa di creare, modificare o cancellare i Frame<sub>G</sub> contenuti nella presentazione;
  - `Premi.Editor.InfographicEditor` posiziona Frame<sub>G</sub> o altri elementi all'interno di un poster;
  - `Premi.Editor.TrailsEditor` ordina i Frame per la creazione di uno o più percorsi di presentazione.

## 4 Descrizione dei singoli componenti

Tipo, obiettivo e funzione del componente

Relazioni d'uso di altre componenti

Interfacce con le relazioni di uso da altre componenti

### 4.1 Premi

—————premi-class.jpg

#### 4.1.1 Premi.Utility

**Nome:** Utility

**Tipo:** class

**Package:** Premi

**Descrizione:** classe statica di utilità che fornisce strumenti per interagire con la base di dati

- **Nome:**
- **Tipo:**
- **Package:**
- **Descrizione:**
- **Relazioni con altri Componenti:**

- 5 Diagrammi delle attività
- 6 Stime di fattibilità e di bisogno di risorse
- 7 Tracciamento della relazione componenti - requisiti
- 8 Descrizione delle tecnologie e dei design pattern utilizzati
- 9 Mockup Interfaccia [FACOLTATIVO]