

# 404NotFound

Premi: better than Prezi.



## Specifica Tecnica

|                               |  |
|-------------------------------|--|
| <b>Versione</b>               | 1.0  |
| <b>Redazione</b>              | Vegro Federico<br>Cossu Mattia<br>Camborata Marco<br>Manuto Monica<br>Rettore Andrea<br>Gobbo Ismaele<br>De Lazzari Enrico |
| <b>Verifica</b>               | Manuto Monica<br>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<br>prof. Tullio Vardanega<br>prof. Riccardo Cardin<br>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. . . . .  | 6         |
| 4.1.2    | Premi.Utility . . . . .   | 6         |
| 4.1.3    | Premi.View . . . . .  | 7         |
| 4.1.4    | Premi.PremiCtrl . . . . .   | 7         |
| 4.2      | Premi.UserManager . . . . .   | 7         |
| 4.2.1    | Premi.UserManager.User . . . . .                                    | 7         |
| 4.2.2    | Premi.UserManager.View . . . . .                                    | 7         |
| 4.2.3    | Premi.UserManager.UserManagerCtrl . . . . .                         | 8         |
| 4.2.4    | Premi.UserManager.LoginView . . . . .                               | 8         |
| 4.2.5    | Premi.UserManager.LoginController . . . . .                         | 8         |
| 4.2.6    | Premi.UserManager.RegistrationView . . . . .                        | 8         |
| 4.2.7    | Premi.UserManager.RegistrationController . . . . .                  | 9         |
| 4.2.8    | Premi.UserManager.ChangePasswordView . . . . .                      | 9         |
| 4.2.9    | Premi.UserManager.ChangePasswordController . . . . .                | 9         |
| 4.3      | Premi.Viewer . . . . .  | 9         |
| 4.3.1    | Premi.Viewer.View . . . . .   | 9         |
| 4.3.2    | Premi.ViewerCtrl . . . . .  | 10        |
| 4.4      | Premi.Presentation . . . . .  | 10        |
| 4.4.1    | Premi.Presentation.GraphicObject . . . . .                          | 10        |
| 4.4.2    | Premi.Presentation.GoContent . . . . .                              | 10        |
| 4.4.3    | Premi.Presentation.Text . . . . .                                   | 10        |
| 4.4.4    | Premi.Presentation.Image . . . . .                                  | 11        |
| 4.4.5    | Premi.Presentation.Shape . . . . .                                  | 11        |
| 4.4.6    | Premi.Presentation.Frame . . . . .                                  | 11        |
| 4.4.7    | Premi.Presentation.Trail . . . . .                                  | 11        |
| 4.4.8    | Premi.Presentation.Presentation . . . . .                           | 12        |
| <b>5</b> | <b>Diagrammi delle attività</b>                                     | <b>12</b> |
| <b>6</b> | <b>Stime di fattibilità e di bisogno di risorse</b>                 | <b>12</b> |
| <b>7</b> | <b>Tracciamento della relazione componenti - requisiti</b>          | <b>12</b> |
| <b>8</b> | <b>Descrizione delle tecnologie e dei design pattern utilizzati</b> | <b>12</b> |



## 9 Mockup Interfaccia [FACOLTATIVO]

12

## 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). 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 `packageG` dedicato alla modifica interna delle presentazioni; possiede al suo interno tre ulteriori `packageG`:
  - `Premi.Editor.FrameEditor` si occupa di creare, modificare o cancellare i `FrameG` contenuti nella presentazione;
  - `Premi.Editor.InfographicEditor` posiziona `FrameG` 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  
—template—

#### 4.1.1 Premi.

- **Nome:**
- **Tipo:** class
- **Package:** Premi.
- **Descrizione:**
- **Relazioni con altri componenti:**

—/template—

#### 4.1.2 Premi.Utility

**Nome:** Utility

**Tipo:** class

**Package:** Premi

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



#### 4.1.3 Premi.View

- **Nome:** View
- **Tipo:** class
- **Package:** Premi
- **Descrizione:** view generale che funge da sfondo per le altre view

#### 4.1.4 Premi.PremiCtrl

- **Nome:** PremiCtrl
- **Tipo:** class
- **Package:** Premi
- **Descrizione:** controller generale che modifica la view principale in base alle scelte dell'utente
- **Relazioni con altri componenti:** la classe altera l'aspetto di `Premi.View` caricando le view di cui l'utente ha bisogno

### 4.2 Premi.UserManager

—————premi-class.jpg

#### 4.2.1 Premi.UserManager.User

—————userManager-class.jpg (mostra le relazioni che possiede ESTER-NE al package)

- **Nome:** User
- **Tipo:** class
- **Package:** Premi.UserManger
- **Descrizione:** classe che definisce un utente e fornisce le funzionalità necessarie alla sua creazione, al suo login e ad un eventuale cambio di password
- **Relazioni con altri componenti:** si collega a `Premi.Utility` per le interazioni con la base di dati

#### 4.2.2 Premi.UserManager.View

- **Nome:** View
- **Tipo:** class
- **Package:** Premi.UserManger
- **Descrizione:** vista generale del package `UserManager`. Può contenere le view abbinate alle funzionalità di `User`

#### 4.2.3 Premi.UserManager.UserManagerCtrl

- **Nome:** UserManagerCtrl
- **Tipo:** class
- **Package:** Premi.UserManager
- **Descrizione:** controller generale del package `UserManager` dedicato a fornire alla view strumenti per l'interazione con l'utente
- **Relazioni con altri componenti:** si collega a `Premi.UserManager.View` per mostrare i dati dell'utente che va a prelevare tramite `Premi.UserManager.User`

#### 4.2.4 Premi.UserManager.LoginView

- **Nome:** LoginView
- **Tipo:** class
- **Package:** Premi.UserManager
- **Descrizione:** view che permette all'utente di effettuare la login

#### 4.2.5 Premi.UserManager.LoginController

- **Nome:** LoginController
- **Tipo:** class
- **Package:** Premi.UserManager
- **Descrizione:** fornisce gli strumenti necessari alla view per effettuare la login
- **Relazioni con altri componenti:** con `Premi.UserManager.LoginView` e con `Premi.UserManager.User` per trasmettere le informazioni relative al login

#### 4.2.6 Premi.UserManager.RegistrationView

- **Nome:** RegistrationView
- **Tipo:** class
- **Package:** Premi.UserManager
- **Descrizione:** view che permette all'utente di registrarsi nel sistema

#### 4.2.7 Premi.UserManager.RegistrationController

- **Nome:** RegistrationController
- **Tipo:** class
- **Package:** Premi.UserManager
- **Descrizione:** fornisce gli strumenti necessari alla view per registrare l'utente
- **Relazioni con altri componenti:** con Premi.UserManager.RegistrationView e con Premi.UserManager.User per trasmettere le informazioni relative alla registrazione

#### 4.2.8 Premi.UserManager.ChangePasswordView

- **Nome:** ChangePasswordView
- **Tipo:** class
- **Package:** Premi.UserManager
- **Descrizione:** view che permette all'utente di cambiare la propria password

#### 4.2.9 Premi.UserManager.ChangePasswordController

- **Nome:** ChangePassword
- **Tipo:** class
- **Package:** Premi.UserManager
- **Descrizione:** fornisce gli strumenti necessari alla view per cambiare la password dell'utente
- **Relazioni con altri componenti:** con Premi.UserManager.ChangePasswordView e con Premi.UserManager.User per trasmettere le informazioni relative al cambio password

### 4.3 Premi.Viewer

————viewer-class.jpg

#### 4.3.1 Premi.Viewer.View

- **Nome:** View
- **Tipo:** class
- **Package:** Premi.Viewer
- **Descrizione:** view che mostra la presentazione all'utente, offrendo funzionalità dipendenti dalla visibilità(pubblica o privata) o dal contesto in cui si trova(presentazione live)

#### 4.3.2 Premi.ViewerCtrl

- **Nome:** ViewerCtrl
- **Tipo:** class
- **Package:** Premi.Viewer
- **Descrizione:** abilita funzionalità nella view in base all'utente
- **Relazioni con altri componenti:** con `Premi.Viewer.View`, con `Premi.UserManager.User` per verificare se l'utente è il proprietario della presentazione, e con la libreria esterna `ImpressJs` per aggiungere animazioni alla presentazione

### 4.4 Premi.Presentation

—————-presentation-class.jpg

#### 4.4.1 Premi.Presentation.GraphicObject

- **Nome:** GraphicObject
- **Tipo:** *abstract class*
- **Package:** Premi.Presentation
- **Descrizione:** rappresenta gli oggetti grafici nella presentazione

#### 4.4.2 Premi.Presentation.GoContent

- **Nome:** GoContent
- **Tipo:** *abstract class*
- **Package:** Premi.Presentation
- **Descrizione:** rappresenta gli oggetti grafici con contenuto di una presentazione
- **Relazioni con altri componenti:** estende `Premi.Presentation.GraphicObject`

#### 4.4.3 Premi.Presentation.Text

- **Nome:** Text
- **Tipo:** class
- **Package:** Premi.Presentation
- **Descrizione:** rappresenta un'area di testo nella presentazione
- **Relazioni con altri componenti:** estende `Premi.Presentation.GoContent`

#### 4.4.4 Premi.Presentation.Image

- **Nome:** Image
- **Tipo:** class
- **Package:** Premi.Presentation
- **Descrizione:** rappresenta un'immagine nella presentazione
- **Relazioni con altri componenti:** estende `Premi.Presentation.GoContent`

#### 4.4.5 Premi.Presentation.Shape

- **Nome:** Shape
- **Tipo:** class
- **Package:** Premi.Presentation
- **Descrizione:** rappresenta una figura nella presentazione
- **Relazioni con altri componenti:** estende `Premi.Presentation.GoContent`

#### 4.4.6 Premi.Presentation.Frame

- **Nome:** Frame
- **Tipo:** class
- **Package:** Premi.Presentation
- **Descrizione:** rappresenta un frame nella presentazione
- **Relazioni con altri componenti:** estende `Premi.Presentation.GraphicObject` e possiede un insieme di oggetti `Premi.Presentation.GoContent`

#### 4.4.7 Premi.Presentation.Trail

- **Nome:** Trail
- **Tipo:** class
- **Package:** Premi.Presentation
- **Descrizione:** rappresenta un percorso nella presentazione
- **Relazioni con altri componenti:** possiede una lista di riferimenti ad oggetti di tipo `Premi.Presentation.Frame`, ma non è in possesso degli oggetti stessi

#### 4.4.8 Premi.Presentation.Presentation

- **Nome:** Presentation
- **Tipo:** class
- **Package:** Premi.Presentation
- **Descrizione:** rappresenta una presentazione
- **Relazioni con altri componenti:** possiede una lista di oggetti `premi.Presentation.Frame`

### 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]