

404NotFound

Premi: better than Prezi.



Specifica Tecnica

Versione	1.0
Redazione	Vegro Federico Cossu Mattia Camborata Marco Manuto Monica Rettore Andrea Gobbo Ismaele De Lazzari Enrico
Verifica	Manuto Monica Rettore Andrea
Responsabile	Gobbo Ismaele
Uso	Esterno
Stato	Formale
Ultima modifica	XXX
Lista di distribuzione	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

1	Introduzione	5
1.1	Scopo del documento	5
1.2	Scopo del prodotto	5
1.3	Glossario	5
2	Definizione del Prodotto	6
2.1	Metodo e formalismo di specifica	6
2.2	Presentazione dell'architettura generale del sistema	6
3	Diagrammi dei Package	6
4	Descrizione dei singoli componenti	7
4.1	Premi	7
4.1.1	Premi.	7
4.1.2	Premi.Utility	7
4.1.3	Premi.View	8
4.1.4	Premi.PremiCtrl	8
4.2	Premi.UserManager	8
4.2.1	Premi.UserManager.User	8
4.2.2	Premi.UserManager.View	8
4.2.3	Premi.UserManager.UserManagerCtrl	9
4.2.4	Premi.UserManager.LoginView	9
4.2.5	Premi.UserManager.LoginController	9
4.2.6	Premi.UserManager.RegistrationView	9
4.2.7	Premi.UserManager.RegistrationController	10
4.2.8	Premi.UserManager.ChangePasswordView	10
4.2.9	Premi.UserManager.ChangePasswordController	10
4.3	Premi.Viewer	10
4.3.1	Premi.Viewer.View	10
4.3.2	Premi.ViewerCtrl	11
4.4	Premi.Presentation	11
4.4.1	Premi.Presentation.GraphicObject	11
4.4.2	Premi.Presentation.GoContent	11
4.4.3	Premi.Presentation.Text	11
4.4.4	Premi.Presentation.Image	12
4.4.5	Premi.Presentation.Shape	12
4.4.6	Premi.Presentation.Frame	12
4.4.7	Premi.Presentation.Trail	12
4.4.8	Premi.Presentation.Presentation	13
4.5	Premi.PresentationManager	13
4.5.1	Premi.PresentationManager.Export	13
4.5.2	Premi.PresentationManager.Portable	13
4.5.3	Premi.PresentationManager.View	13
4.5.4	Premi.PresentationManager.PresentationManagerCtrl	14
4.5.5	Premi.PresentationManager.RemoveView	14
4.5.6	Premi.PresentationManager.RemoveController	14

4.5.7	Premi.PresentationManager.PublicView	14
4.5.8	Premi.PresentationManager.PublicController	15
4.5.9	Premi.PresentationManager.EditTitleDescrView	15
4.5.10	Premi.PresentationManager.EditTitleDescrController	15
4.5.11	Premi.PresentationManager.NewView	15
4.5.12	Premi.PresentationManager.NewController	16
4.5.13	Premi.PresentationManager.ListPresView	16
4.5.14	Premi.PresentationManager.ListPresController	16
4.6	Premi.Editor	16
4.6.1	Premi.Editor.View	16
4.6.2	Premi.Editor.EditorCtrl	17
4.6.3	Premi.Editor.TextView	17
4.6.4	Premi.Editor.TextController	17
4.6.5	Premi.Editor.ShapeView	17
4.6.6	Premi.Editor.ShapeController	18
4.6.7	Premi.Editor.ImageView	18
4.6.8	Premi.Editor.ImageController	18
4.6.9	Premi.Editor.StyleView	18
4.6.10	Premi.Editor.StyleController	18
4.6.11	Premi.Editor.FrameListView	19
4.6.12	Premi.Editor.FrameListController	19
4.7	Premi.Editor.FrameEditor	19
4.7.1	Premi.Editor.FrameEditor.View	19
4.7.2	Premi.Editor.FrameEditor.FrameEditorCtrl	19
4.7.3	Premi.	20
4.7.4	Premi.	20
4.7.5	Premi.	20
5	Diagrammi delle attività	20
6	Stime di fattibilità e di bisogno di risorse	20
7	Tracciamento della relazione componenti - requisiti	20
8	Descrizione delle tecnologie e dei design pattern utilizzati	20
9	Mockup Interfaccia [FACOLTATIVO]	20

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_G utilizzati, e un mockup_G 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_G, sviluppato in tecnologia HTML5_G 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_G: verranno prima descritti i package_G e le loro dipendenze e successivamente le singole classi contenute al loro interno. I diagrammi delle classi e dei package_G seguono il formalismo UML_G2.0 e la struttura dei package segue una prassi (best practice_G) di AngularJS_G che propone una suddivisione dei componenti per funzionalità dell'applicazione in alternativa alla classica suddivisione Model-View-Controller_G, più difficile da mantenere per applicazioni di medie o grandi dimensioni. Per ulteriori approfondimenti consultare la guida al sito 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_G.

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_G. Questo aumenta la chiarezza espositiva dei diagrammi, evita la creazione di package_G 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_G che di tipo Model-View-ViewModel_G. 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_G principale chiamato **Premi**; al suo interno sono presenti:

- **Premi** racchiude tutti i package_G e la view e il controller principali;
- **Premi.UserManager** è il package_G 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

—dipendenze di presentation.jpg CITARE L'IMMAGINE DELLE DIPENDENZE CON DI PRESENTATION Completo-Dettagliato.jpg

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`, identifica l'appartenenza di una presentazione ad un determinato `Premi.UserManager.User` e si collega alla base di dati tramite la classe statica `Premi.Utility`

4.5 Premi.PresentationManager

——-userManager-classe.jpg

4.5.1 Premi.PresentationManager.Export

- **Nome:** Export
- **Tipo:** class
- **Package:** Premi.PresentationManager
- **Descrizione:** permette di esportare la presentazione in formato poster
- **Relazioni con altri componenti:** preleva le informazioni da `Premi.Presentation.Presentation` per costruire il poster

4.5.2 Premi.PresentationManager.Portable

- **Nome:** Portable
- **Tipo:** class
- **Package:** Premi.PresentationManager
- **Descrizione:** permette di rendere portabile una presentazione, e quindi di essere indipendente dal software Premi e di essere caricata come pagina HTML da un browser
- **Relazioni con altri componenti:**

4.5.3 Premi.PresentationManager.View

- **Nome:** View
- **Tipo:** class
- **Package:** Premi.PresentationController
- **Descrizione:** è la view generale delle operazioni che l'utente può effettuare sul

4.5.4 Premi.PresentationManager.PresentationManagerCtrl

- **Nome:** PresentationManagerCtrl
- **Tipo:** class
- **Package:** Premi.PresentationManager
- **Descrizione:** fornisce alla view gli strumenti necessari alla gestione delle presentazioni
- **Relazioni con altri componenti:** è il controller dedicato di `Premi.PresentationManager.View`

4.5.5 Premi.PresentationManager.RemoveView

- **Nome:** RemoveView
- **Tipo:** class
- **Package:** Premi.PresentationManager
- **Descrizione:** mostra una finestra di conferma eliminazione della presentazione selezionata

4.5.6 Premi.PresentationManager.RemoveController

- **Nome:** RemoveController
- **Tipo:** class
- **Package:** Premi.PresentationManager
- **Descrizione:** fornisce alla view gli strumenti per rimuovere la presentazione o annullare l'azione
- **Relazioni con altri componenti:** è il controller dedicato di `Premi.PresentationManager.RemoveView` e sfrutta le funzionalità di `Premi.Presentation.Presentation` per l'eliminazione della presentazione

4.5.7 Premi.PresentationManager.PublicView

- **Nome:** PublicView
- **Tipo:** class
- **Package:** Premi.PresentationManager
- **Descrizione:** mostra una finestra di conferma per rendere pubblica o privata una presentazione

4.5.8 Premi.PresentationManager.PublicController

- **Nome:** PublicController
- **Tipo:** class
- **Package:** Premi.PresentationManager
- **Descrizione:** fornisce alla view gli strumenti per rendere pubblica o privata una presentazione
- **Relazioni con altri componenti:** è il controller dedicato di `Premi.PresentationManager.PublicController` e sfrutta le funzionalità di `Premi.Presentation.Presentation` per accedere alla base di dati

4.5.9 Premi.PresentationManager.EditTitleDescrView

- **Nome:** EditTitleDescrView
- **Tipo:** class
- **Package:** Premi.PresentationManager
- **Descrizione:** mostra una finestra per la modifica del titolo e della descrizione della presentazione

4.5.10 Premi.PresentationManager.EditTitleDescrController

- **Nome:** EditTitleDescrController
- **Tipo:** class
- **Package:** Premi.PresentationManager
- **Descrizione:** fornisce alla view la possibilità di modificare titolo e descrizione della presentazione
- **Relazioni con altri componenti:** è il controller dedicato di `Premi.PresentationManager.EditTitleDescrView` e sfrutta le funzionalità di `Premi.Presentation.Presentation` per accedere alla base di dati

4.5.11 Premi.PresentationManager.NewView

- **Nome:** NewView
- **Tipo:** class
- **Package:** Premi.PresentationManager
- **Descrizione:** mostra una finestra per la creazione di una nuova presentazione

4.5.12 Premi.PresentationManager.NewController

- **Nome:** NewController
- **Tipo:** class
- **Package:** Premi.PresentationManager
- **Descrizione:** fornisce alla view gli strumenti per creare una nuova presentazione
- **Relazioni con altri componenti:** è il controller dedicato di `Premi.PresentationManager.New` e sfrutta le funzionalità di `Premi.Presentation.Presentation` per accedere alla base di dati

4.5.13 Premi.PresentationManager.ListPresView

- **Nome:** ListPresView
- **Tipo:** class
- **Package:** Premi.PresentationManager
- **Descrizione:** mostra all'utente la lista delle sue presentazioni, e bottoni aggiuntivi per accedere alle altre funzionalità del package

4.5.14 Premi.PresentationManager.ListPresController

- **Nome:** ListPresController
- **Tipo:** class
- **Package:** Premi.PresentationManager
- **Descrizione:** fornisce alla view una lista preformata delle presentazioni dell'utente
- **Relazioni con altri componenti:** è il controller dedicato di `Premi.PresentationManager.List` e sfrutta le funzionalità di `Premi.Presentation.Presentation` per recuperare la lista delle presentazioni

4.6 Premi.Editor

4.6.1 Premi.Editor.View

- **Nome:** View
- **Tipo:** class
- **Package:** Premi.Editor
- **Descrizione:** view generica dell'editor che funge da contenitore

4.6.2 Premi.Editor.EditorCtrl

- **Nome:** EditorCtrl
- **Tipo:** class
- **Package:** Premi.Editor
- **Descrizione:** fornisce alla view le funzionalità per la modifica di una presentazione
- **Relazioni con altri componenti:** è il controller dedicato di `Premi.Editor.EditorView` e sfrutta `Premi.Presentation.Presentation` per recuperare la presentazione dalla base di dati

4.6.3 Premi.Editor.TextView

- **Nome:** TextView
- **Tipo:** class
- **Package:** Premi.Editor
- **Descrizione:** mostra le opzioni per l'aggiunta di un'area di testo

4.6.4 Premi.Editor.TextController

- **Nome:** TextController
- **Tipo:** class
- **Package:** Premi.Editor
- **Descrizione:** fornisce alla view le funzionalità per creare e modificare un'area di testo
- **Relazioni con altri componenti:** è il controller dedicato di `Premi.Editor.TextView`

4.6.5 Premi.Editor.ShapeView

- **Nome:** ShapeView
- **Tipo:** class
- **Package:** Premi.Editor
- **Descrizione:** mostra le opzioni per l'aggiunta di uno shape

4.6.6 Premi.Editor.ShapeController

- **Nome:** ShapeController
- **Tipo:** class
- **Package:** Premi.Editor
- **Descrizione:** fornisce alla view le funzionalità per creare e modificare uno shape
- **Relazioni con altri componenti:** è il controller dedicato di Premi.Editor.ShapeView

4.6.7 Premi.Editor.ImageView

- **Nome:** ImageView
- **Tipo:** class
- **Package:** Premi.Editor
- **Descrizione:** mostra le opzioni per l'aggiunta di un'immagine

4.6.8 Premi.Editor.ImageController

- **Nome:** ImageController
- **Tipo:** class
- **Package:** Premi.Editor
- **Descrizione:** fornisce alla view le funzionalità per aggiungere un'immagine
- **Relazioni con altri componenti:** è il controller dedicato di Premi.Editor.ImageView

4.6.9 Premi.Editor.StyleView

- **Nome:** StyleView
- **Tipo:** class
- **Package:** Premi.Editor
- **Descrizione:** permette la modifica dello stile di un Frame oppure dello sfondo dell'infografica

4.6.10 Premi.Editor.StyleController

- **Nome:** StyleController
- **Tipo:** class
- **Package:** Premi.Editor
- **Descrizione:** fornisce alla view le funzionalità per la modifica dello stile
- **Relazioni con altri componenti:** è il controller dedicato di Premi.Editor.StyleView

4.6.11 Premi.Editor.FrameListView

- **Nome:** FrameListView
- **Tipo:** class
- **Package:** Premi.Editor
- **Descrizione:** view di base per la rappresentazione di una lista dei Frame contenuti all'interno della presentazione

4.6.12 Premi.Editor.FrameListController

- **Nome:** FrameListController
- **Tipo:** class
- **Package:** Premi.Editor
- **Descrizione:** fornisce alla view le funzionalità per rappresentazione della lista dei Frame
- **Relazioni con altri componenti:** è il controller dedicato di Premi.Editor.FrameListView

4.7 Premi.Editor.FrameEditor

4.7.1 Premi.Editor.FrameEditor.View

- **Nome:** View
- **Tipo:** class
- **Package:** Premi.Editor.FrameEditor.View
- **Descrizione:** è la view generale della fase di modifica dei Frame

4.7.2 Premi.Editor.FrameEditor.FrameEditorCtrl

- **Nome:** FrameEditorCtrl
- **Tipo:** class
- **Package:** Premi.Editor.FrameEditor.FrameEditorCtrl
- **Descrizione:** fornisce alla view le funzionalità per la gestione delle altre view contenute al suo interno
- **Relazioni con altri componenti:** è il controller dedicato di Premi.Editor.FrameEditor.FrameEditorView

4.7.3 Premi.

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

4.7.4 Premi.

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

4.7.5 Premi.

- **Nome:**
- **Tipo:** class
- **Package:** Premi.
- **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]