

Mattia Cerantola

MindSlide Mobile

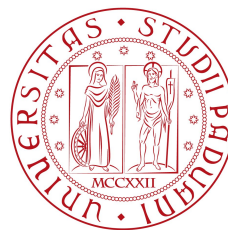
Specifica tecnica

Sommario

Il presente documento riporta l'insieme delle scelte progettuali che si intendono adottare durante lo sviluppo del software **MindSlide Mobile**.

Informazioni documento

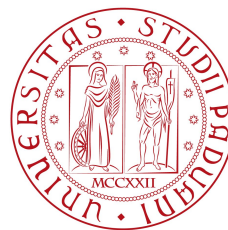
Nome	specifica_tecnica_1.0.pdf
Versione	1.0
Data creazione	04/05/2011
Data ultima modifica	09/05/2011
Stato del Documento	Formale ad uso interno
Distribuzione	Mattia Cerantola Dott. Stefano Boldrin Dott.sa Ombretta Gaggi Dott. Gregorio Piccoli



Registro delle modifiche:

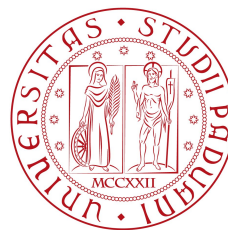
Versione	Data	Modifiche effettuate
1.0	09/05/2011	aggiunto i capitoli Linguaggi (2.2) e Realizzazione di MindSlide Mobile (4) , aggiunte tutte le immagini.
0.3	06/05/2011	aggiunto il capitolo Strumenti e risorse (5) .
0.2	05/05/2011	aggiunto il capitolo MindSlide su touch screen (3) .
0.1	04/05/2011	prima stesura del documento: inseriti i capitoli da Introduzione (1) a MindSlide (2) incluso.

Tabella 1: registro delle modifiche



Indice

1	Introduzione	5
1.1	Scopo del documento	5
1.2	Scopo del prodotto	5
1.3	Glossario	5
1.4	Riferimenti	5
1.4.1	Normativi	5
1.4.2	Informativi	5
2	MindSlide	6
2.1	Architettura	6
2.2	Linguaggi	7
2.2.1	HTML5	7
2.2.2	CSS3	7
2.2.3	JavaScript	7
2.2.4	JSON	7
2.2.5	JQuery	7
3	MindSlide su touch screen	8
3.1	Homepage	9
3.2	MindMap View	11
3.3	Slide View	14
3.4	Presentation View	16
4	Realizzazione di MindSlide Mobile	18
4.1	Simulazione dell'hover	18
4.2	CSS	18
4.3	Homepage Mobile	18
4.4	MindMap View Mobile	19
4.5	Slide View Mobile	19
4.6	Presentation View Mobile	19
5	Strumenti e risorse	21
5.1	Hardware	21
5.2	Software	21
5.2.1	Linguaggi	21
5.2.2	Librerie	21
5.2.3	Editor	21

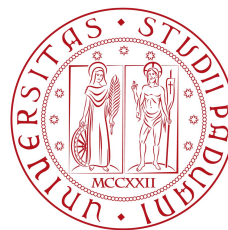


Elenco delle tabelle

1	registro delle modifiche	2
---	--	---

Elenco delle figure

1	diagramma UML del pattern MVC	6
2	Homepage di MindSlide (orientamento portrait)	9
3	Homepage di MindSlide (orientamento landscape)	10
4	MindMap View (orientamento portrait)	11
5	MindMap View (orientamento landscape)	12
6	Slide sibling della slide 4 non raggiungibile tramite drag&drop	13
7	Slide View (orientamento portrait)	14
8	Slide View (orientamento landscape)	15
9	Presentation View (orientamento portrait)	16
10	Presentation View (orientamento landscape)	17



1 Introduzione

1.1 Scopo del documento

Lo scopo di questo documento è quello di fornire una visione della progettazione che si intende attuare nello sviluppo del software *MindSlide Mobile*. Verranno quindi analizzati in questo documento:

1. la struttura attuale di *MindSlide* (2);
2. il comportamento attuale di *MindSlide su touch screen* (3);
3. le implementazioni/modifiche che si intendono attuare per la *Realizzazione di MindSlide Mobile* (4);
4. quali *Strumenti e risorse* (5) si andranno ad utilizzare.

1.2 Scopo del prodotto

Il prodotto è un'evoluzione ed implementazione di *MindSlide*: esso fornisce un applicativo utilizzabile da **browser** per la creazione, lo sviluppo, la visualizzazione ed il salvataggio locale di presentazioni di **slide**. Tale software fa uso del linguaggio **HTML5** ed è in grado di funzionare anche in mancanza di una connessione attiva ad Internet a condizione che la pagina web sia stata visitata almeno una volta in modalità online.

Le nuove funzionalità renderanno l'applicazione utilizzabile anche su dispositivo **mobile**, come telefoni e tablet, massimizzando l'esperienza **mobile** tramite l'utilizzo di funzionalità dedicate, come il **multitouch**, le **gesture** e l'uso dell'**accelerometro**.

NOTA: come già detto nel documento *Analisi_dei_requisiti_1.0* per dispositivi **mobile** si intendono *iPad 1.0* o successivi e *iPhone 3.0* o successivi; si veda il documento sopra citato per maggiori informazioni.

1.3 Glossario

Al fine di eliminare ambiguità e incomprensioni, tutti i termini tecnici e le sigle utilizzati nel presente documento sono riportati nel Glossario, fornito come allegato.

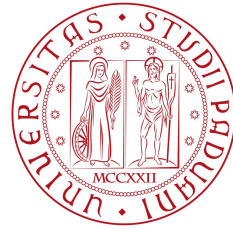
1.4 Riferimenti

1.4.1 Normativi

- Documento *analisi_requisiti_1.0*.

1.4.2 Informativi

- *MindSlide*:
<http://impslide.sourceforge.net>;
- Documentazione jQueryMobile:
<http://jquerymobile.com>;
- Documentazione Scripty2:
<http://scripty2.com/doc>.



2 MindSlide

In questo capitolo analizzeremo la struttura del software **MindSlide**, dato che esso è la base da cui è iniziato il progetto di stage. In particolare verranno analizzati:

1. descrizione dell'*Architettura* (2.1);
2. i *Linguaggi* (2.2) utilizzati;

2.1 Architettura

MindSlide utilizza il pattern architetturale MVC. Tale pattern prevede l'individuazione di tre componenti principali: Model, Controller e View, messe in comunicazione tra loro. Ogni componente interna può essere scomposta a sua volta in più sottocomponenti, divise per funzionalità fornite o per scopo.

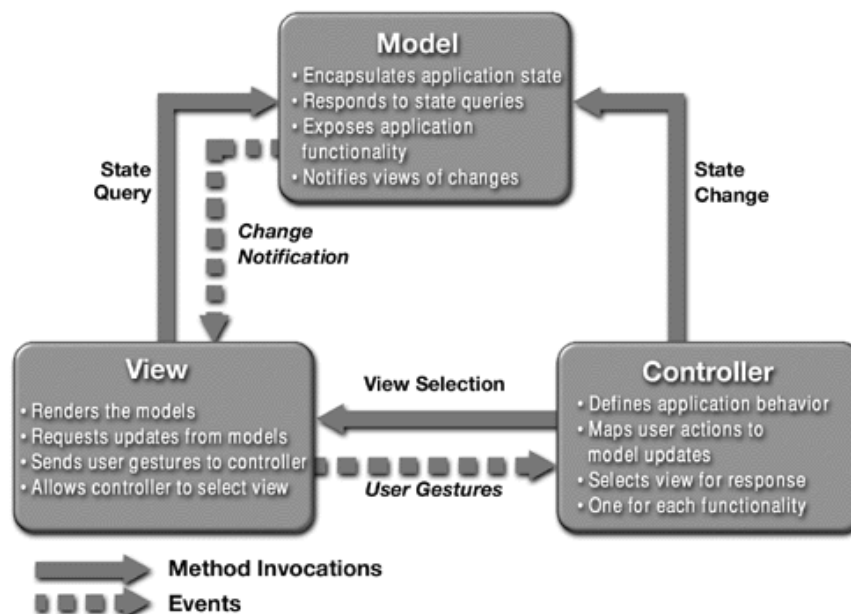
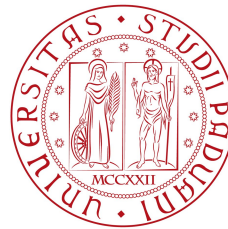


Figura 1: diagramma UML del pattern MVC

La componente Model racchiude al suo interno tutti i dati che l'utente immette nel sistema, oltre ad oggetti di configurazione specifici dell'applicazione. Gli oggetti contenuti nella componente Model rappresentano quindi lo stato globale del progetto. Per offrire le proprie funzionalità la componente sfrutta un modulo di persistenza che si occupa di salvare e caricare gli oggetti in file JSON, strutturati secondo le esigenze di rappresentazione del progetto.

La componente Controller opera in comunicazione sia con la componente Model, sia con la componente View. Le connessioni con Model riguardano principalmente la gestione degli oggetti di stato, realizzate attraverso appositi gestori in grado di modificare lo stato del Model. Dal lato opposto vi è la comunicazione con la componente View, che riguarda principalmente l'invio di messaggi all'utente e la gestione delle richieste effettuate all'interfaccia.

La componente View si occupa di presentare all'utente l'interfaccia con cui interagire. Per fornire queste funzionalità essa raccoglie al suo interno più sottosistemi, tra cui una libreria



grafica in grado di rappresentare efficacemente la struttura della **MindMap** ed un modulo in grado di notificare messaggi all'utente.

Le dipendenze fra le tre componenti sono descritte nel [diagramma UML del pattern MVC](#) (1).

2.2 Linguaggi

In questo paragrafo analizzeremo i principali linguaggi su cui si basa **MindSlide** (per la definizione dei linguaggi si rimanda al documento *glossario_2-0*).

2.2.1 HTML5

HTML5 è il linguaggio principale utilizzato nella realizzazione di **MindSlide**. Essendo il software un'applicazione online esso è interamente basato su questa tecnologia, che ne permette l'esecuzione su tutti i **browser** compatibili con la versione 5.

HTML5 introduce e consente di utilizzare diverse funzionalità tra cui l'offline storage e il supporto alla grafica **SVG**, ampiamente utilizzati da **MindSlide**.

2.2.2 CSS3

MindSlide fa un ampio uso dei fogli di stile e grazie alle nuove funzionalità della versione 3 (come ad esempio i bordi arrotondati, gli spostamenti dinamici degli oggetti e la gestione dell'opacità) offre una grafica semplice e funzionale in tutte le sue interfacce.

2.2.3 JavaScript

JavaScript è il motore principale di **MindSlide**: intercetta e gestisce le azioni di input, fornisce le interfacce ed i comandi all'utente e consente di essere eseguito dinamicamente sui **browser** senza dover compilare nulla od installare componenti aggiuntive.

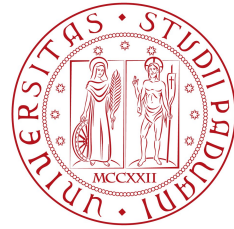
2.2.4 JSON

MindSlide si occupa del salvataggio e del caricamento dei dati su **local storage**; i dati salvati tramite questo sistema devono essere necessariamente in formato stringa.

JSON si occupa della trasformazione dei dati in stringa quando essi devono essere salvati nel **local storage** e della loro ritrasformazione quando essi devono essere caricati e letti da **MindSlide**.

2.2.5 JQuery

JQuery viene utilizzato all'interno di **MindSlide** per cercare di raggiungere una standardizzazione ed una portabilità massima: esso infatti si occupa, tra le altre cose, della gestione degli eventi su diversi **browser**.

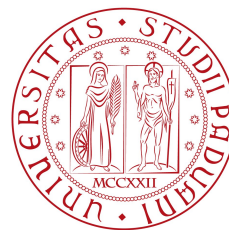


3 MindSlide su touch screen

Attualmente **MindSlide** presenta già una parziale compatibilità con i dispositivi **touch** screen nonostante sia stato progettato e realizzato per dispositivi come PC o MAC. In questo capitolo andremo ad analizzare il comportamento attuale di **MindSlide** eseguito su dispositivi dotati di schermo tattile, distinguendo i comportamenti rilevati in base alle diverse interfacce utilizzate:

1. [Homepage \(3.1\)](#)
2. [MindMap View \(3.2\)](#);
3. [Slide View \(3.3\)](#);
4. [Presentation View \(3.4\)](#).

É importante far notare, come già accennato nel documento *analisi_dei_requisiti_1_0*, che una pagina web non-touch eseguita su dispositivo **touch** risulta essere "adattata" per funzionare su questo tipo di device, cercando di interpretare (spesso erroneamente) il **touch** come il click del mouse. Questa interpretazione ambigua genera spesso confusione, soprattutto quando si cercano di interpretare funzioni di **hover**, **focus**, **out**, **etc**, praticamente non riproducibili (se non generando comportamenti imprevedibili) su dispositivo **touch**.



3.1 Homepage



Figura 2: Homepage di MindSlide (orientamento portrait)

Specifica tecnica

Versione: 1.0

Mail: mattiaglove@gmail.com

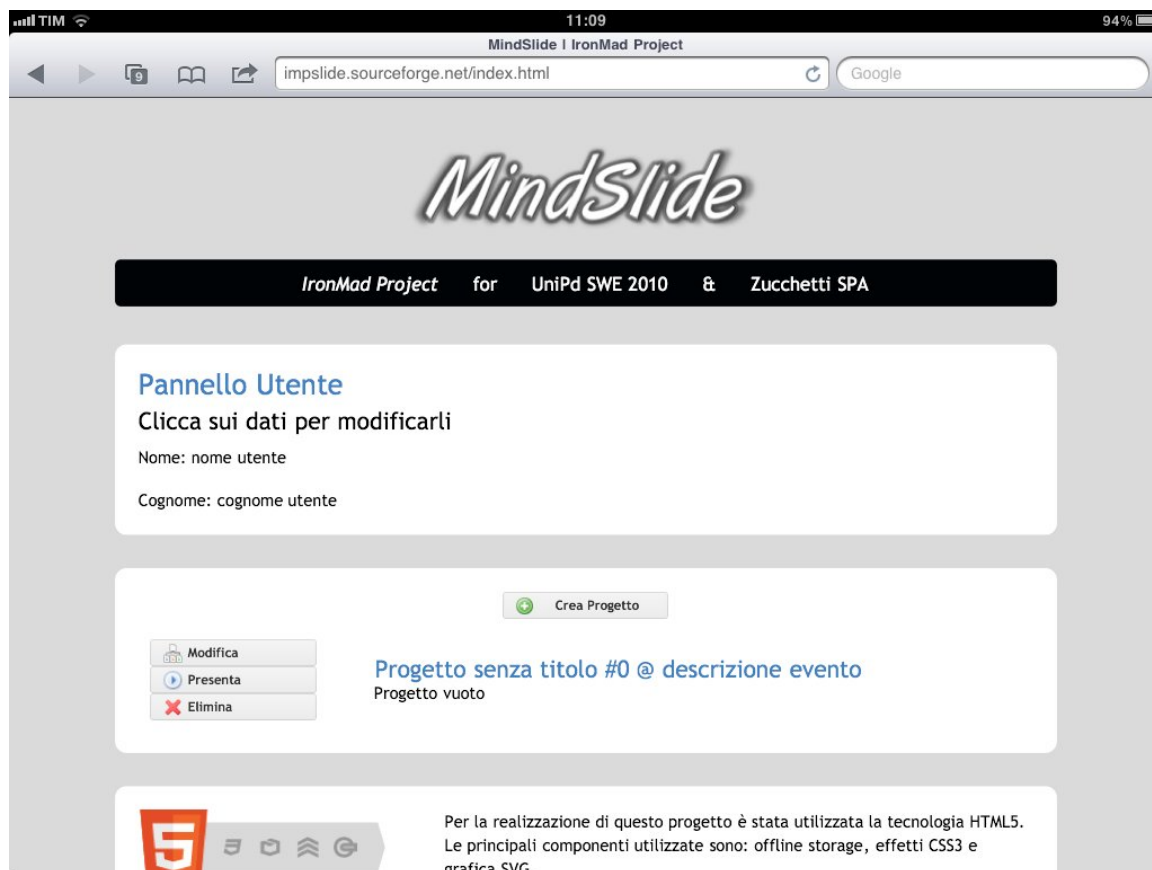
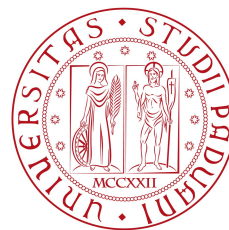


Figura 3: Homepage di MindSlide (orientamento landscape)

Come si può vedere dalle figure [Homepage di MindSlide \(orientamento portrait\)](#) (2) e [Homepage di MindSlide \(orientamento landscape\)](#) (3) la homepage viene correttamente visualizzata: tutti gli elementi sono disposti e visualizzati come se il software venisse eseguito su un normale PC da tavolo e le varie funzioni di creazione, visualizzazione, modifica ed eliminazione delle presentazioni funzionano correttamente; tuttavia i pulsanti per le funzioni di creazione/visualizzazione/modifica/eliminazione risentono del famoso effetto "simulazione dell'hover" che porta ad avere il seguente effetto: il primo **touch** evidenzia il pulsante (equivalente ad **hover** su un computer dotato di mouse), il secondo esegue effettivamente il **touch** (equivalente al click su un computer dotato di mouse).

Bisogna inoltre notare che lo spazio non è sfruttato bene, non esistono fogli di stile dedicati e la visualizzazione **landscape/portrait** viene gestita secondo le regole del dispositivo e non del CSS. Questo problema riguarda tutte le interfacce di **mindslide**, in quanto non è stato ideato nessun foglio di stile dedicato a dispositivi di piccole dimensioni e/o con schermo orientabile; si eviterà quindi di citare nuovamente questo problema nell'analisi del software di partenza; verranno invece ampiamente discusse soluzioni dedicate per ogni interfaccia di **MindSlide Mobile** nel capitolo [Realizzazione di MindSlide Mobile](#) (4) e relativi sottocapitoli.

Modifiche ed implementazioni da apportare a questa interfaccia verranno affrontate nel paragrafo [Homepage Mobile](#) (4.3).

3.2 MindMap View

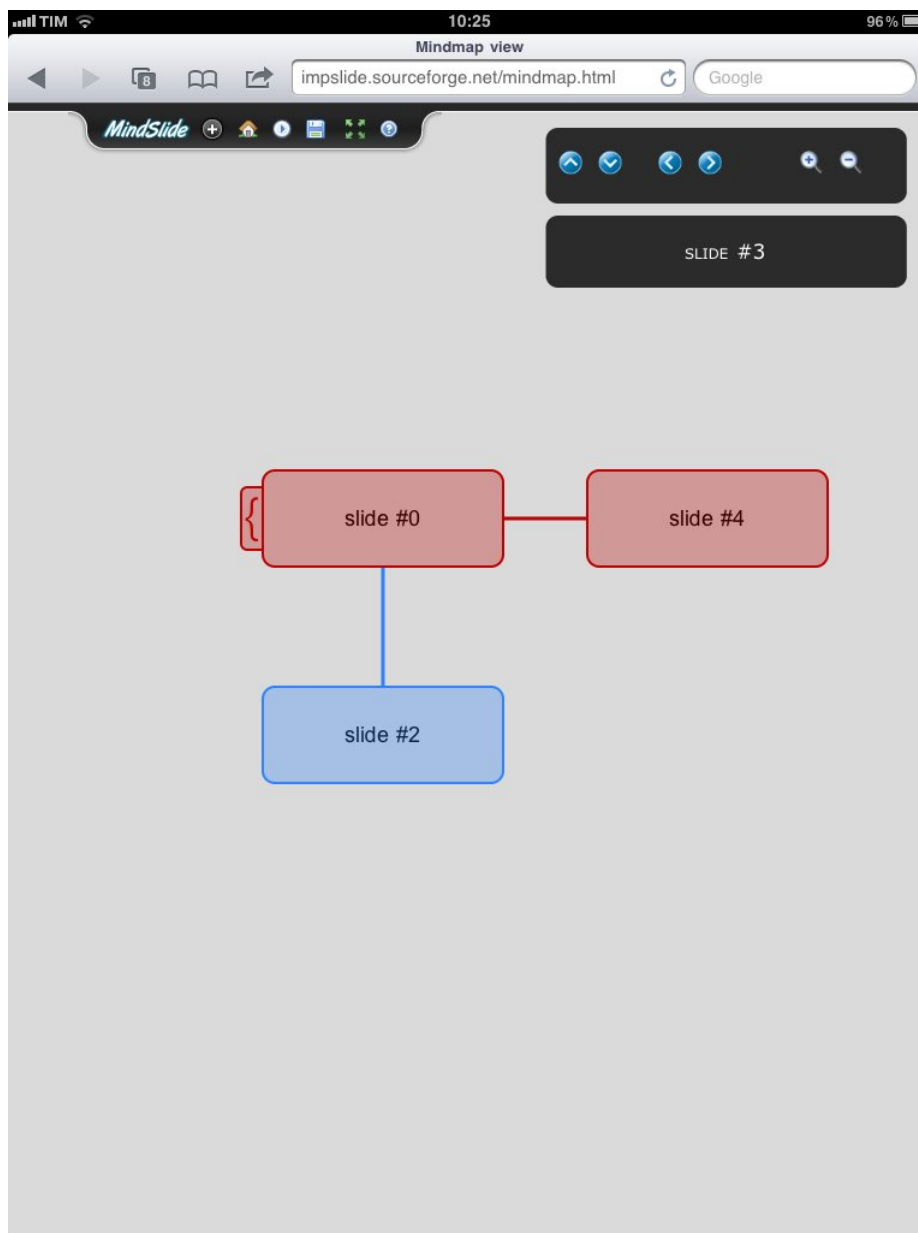


Figura 4: MindMap View (orientamento portrait)

Specifica tecnica

Versione: 1.0

Mail: mattiaglove@gmail.com

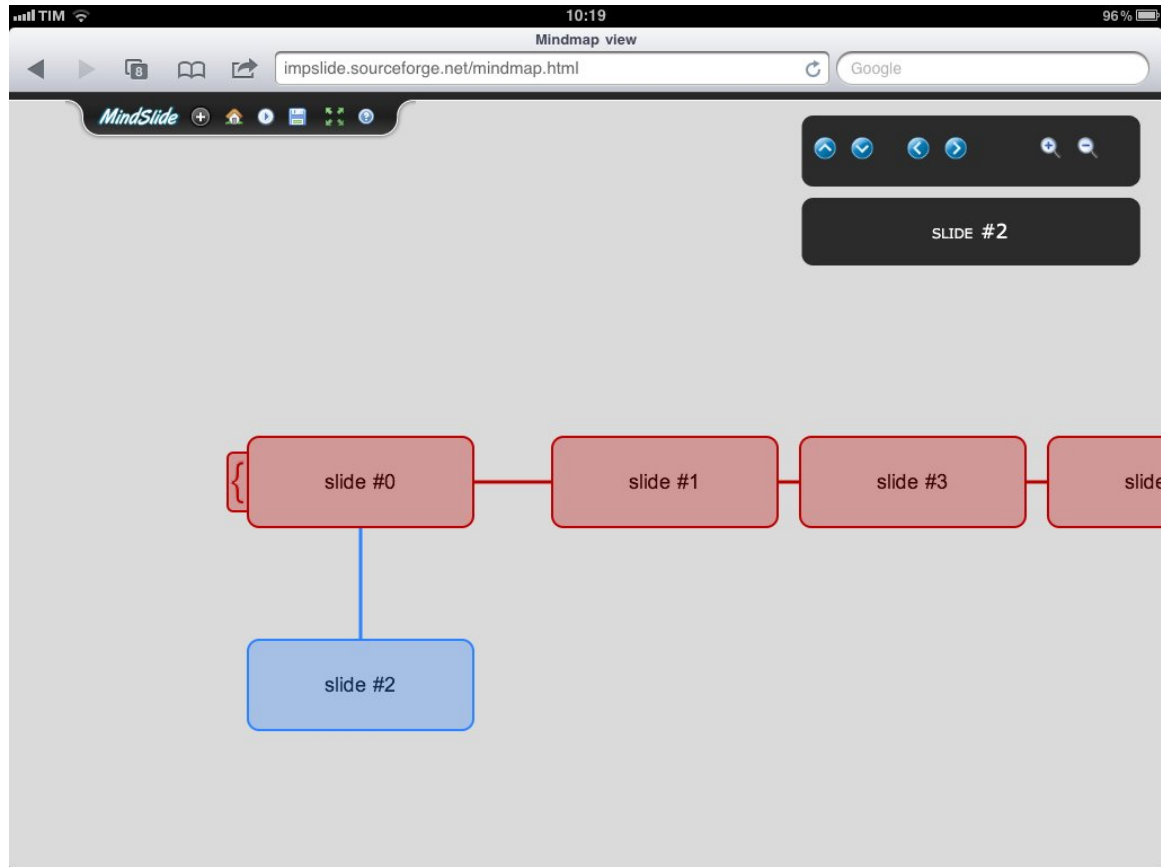


Figura 5: MindMap View (orientamento landscape)

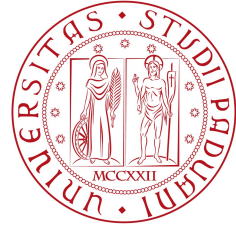
Anche questa interfaccia viene visualizzata su dispositivo **mobile** senza particolari problemi, come possiamo vedere dalle figure [MindMap View \(orientamento portrait\)](#) (4) e [MindMap View \(orientamento landscape\)](#) (5).

Tutti gli elementi vengono caricati correttamente: la barra di gestione progetto (sia aperta che chiusa), il gritter di navigazione (in alto a destra), la **mindmap** e le **slide**.

Questa interfaccia, esattamente come i bottoni della homepage, risente dell'effetto "simulazione dell'**hover**", di cui abbiamo già parlato sopra. In questa modalità esso risulta essere particolarmente fastidioso, in quanto l'**hover** genera una serie di eventi: il **touch** su una **slide** genera l'apertura di un gritter di anteprima (esattamente come succede con l'azione di **hover** del mouse su PC) e l'evidenziazione con un colore più marcato della serie di **slide** di cui fa parte la **slide** su cui è stato eseguito il **touch**.

Ciò non è sicuramente la miglior azione da eseguire su un dispositivo **touch**, dato che il tocco a schermo indica un'interesse ad eseguire un'operazione su quella precisa **slide**, contrariamente a quanto succede con il mouse, dove un passaggio su una **slide** può indicare una semplice transizione e non un'interesse a quella specifica **slide**.

Le azioni di **drag&drop** e le **gesture** hanno un comportamento molto diverso rispetto all'esecuzione su terminale fisso: il **drag&drop** eseguito sulla **mindmap** non causa lo spostamento isolato



di quest'ultima, lasciando correttamente ancorati menù di progetto e gritter vari, ma viene interpretato dal dispositivo **mobile** come spostamento dell'intera pagina, causando l'impossibilità di raggiungere **slide** attualmente non visualizzate, come si vede dall'immagine [Slide sibling della slide 4 non raggiungibile tramite drag&drop \(6\)](#).

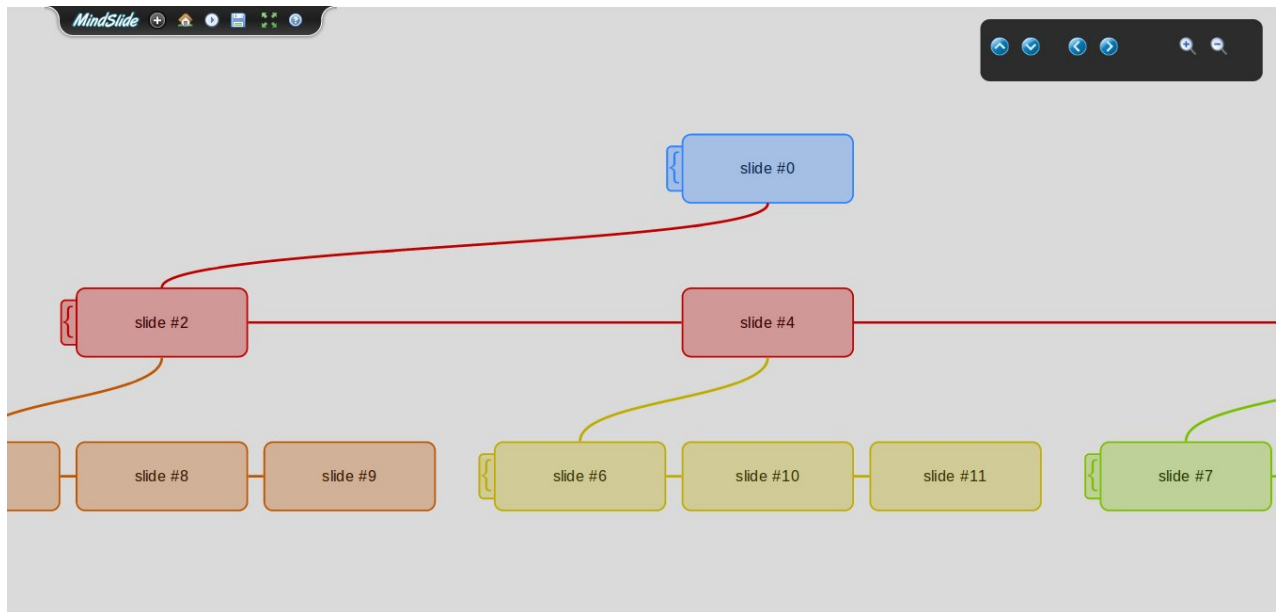
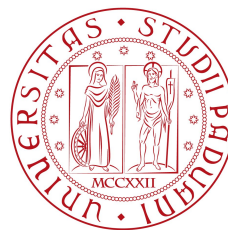


Figura 6: Slide sibling della slide 4 non raggiungibile tramite drag&drop

In realtà le **slide** esterne allo schermo sono comunque raggiungibili, grazie al gritter di navigazione che mette a disposizione delle frecce di navigazione che spostano solo la **mindmap**. Tuttavia questa soluzione non è accettabile in un software mirato a sfruttare al meglio le risorse di un dispositivo **touch**. Si parlerà ampiamente di come implementare questa funzione nel capitolo [MindMap View Mobile \(4.4\)](#).

Anche le **gesture** di **pinch** e **reverse-pinch** non si comportano come ci si aspetterebbe: esattamente come il **drag&drop** esse interagiscono con l'intera pagina e non in maniera isolata con la **mindmap**. Anche queste azioni possono comunque essere attivate grazie al gritter di navigazione, toccando le lenti di ingrandimento/rimpicciolimento; come detto prima queste soluzioni non possono però essere lo strumento principale, ma bensì un'alternativa a chi non disponga di dispositivi compatibili con le **gesture**. Nel capitolo [MindMap View Mobile \(4.4\)](#) si evidenzieranno le implementazioni da attuare al software per aggiungere questa funzionalità.



3.3 Slide View

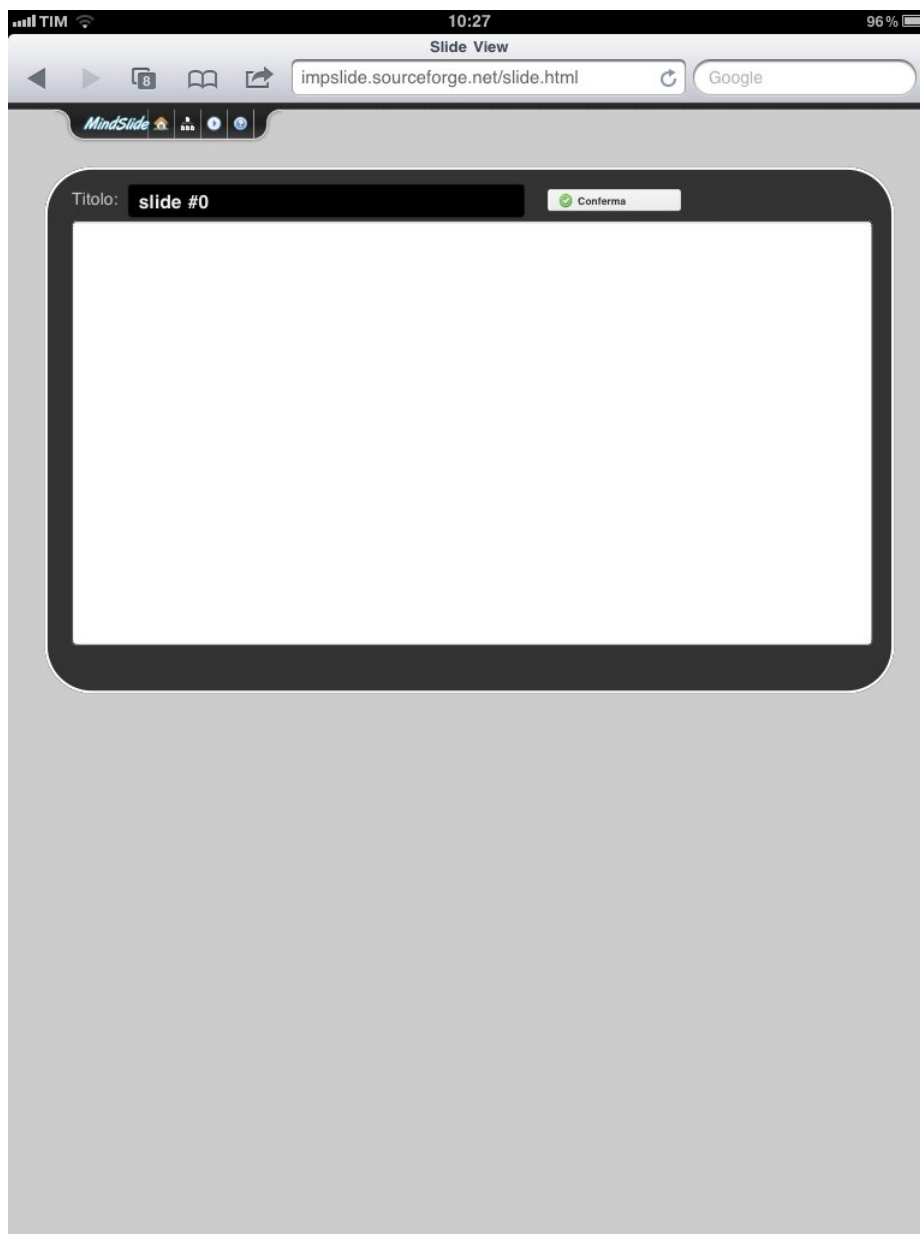


Figura 7: Slide View (orientamento portrait)

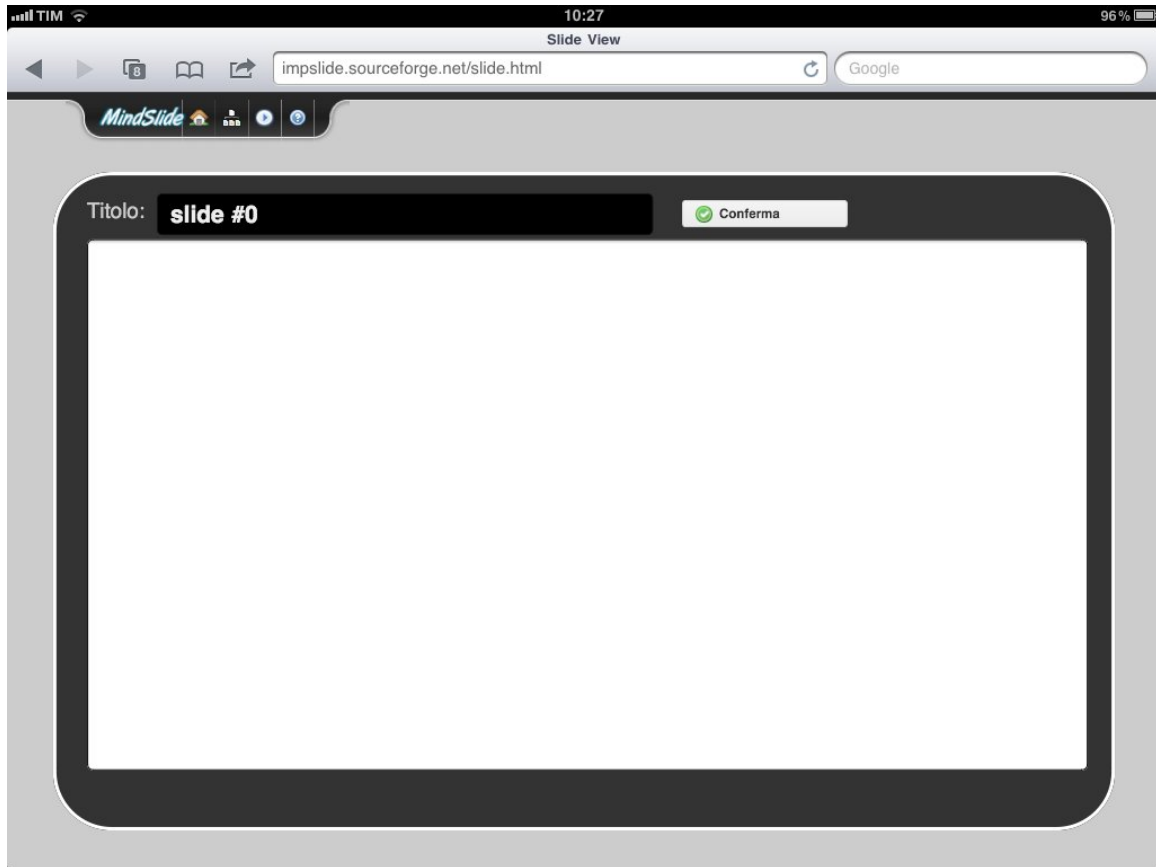
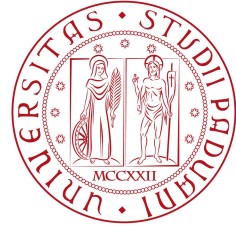
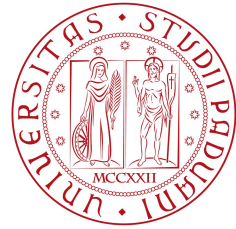


Figura 8: Slide View (orientamento landscape)

In modalità **slide view** la compatibilità inizia a venir meno: nonostante la pagina sia decisamente meno complessa della **mindmap view** essa include un **Rich Text Editor**, *TinyMCE*. Il problema risulta essere la mancanza di compatibilità da parte dei dispositivi **mobile** verso questo tipo di editor: ciò causa errori **JavaScript** durante l'esecuzione su **mobile device** ed il **Rich Text Editor** non viene caricato, come si può notare dalle figure [Slide View \(orientamento portrait\)](#) (7) e [Slide View \(orientamento landscape\)](#) (8).

A parte il solito problema della "simulazione dell'**hover**" sul pulsante di conferma tutte le altre funzionalità sono correttamente supportate e l'inserimento di titolo e testo avvengono tramite tastiera virtuale del dispositivo **mobile**. Come affrontare il problema **Rich Text Editor** sarà il contenuto del capitolo [Slide View Mobile](#) (4.5).



3.4 Presentation View

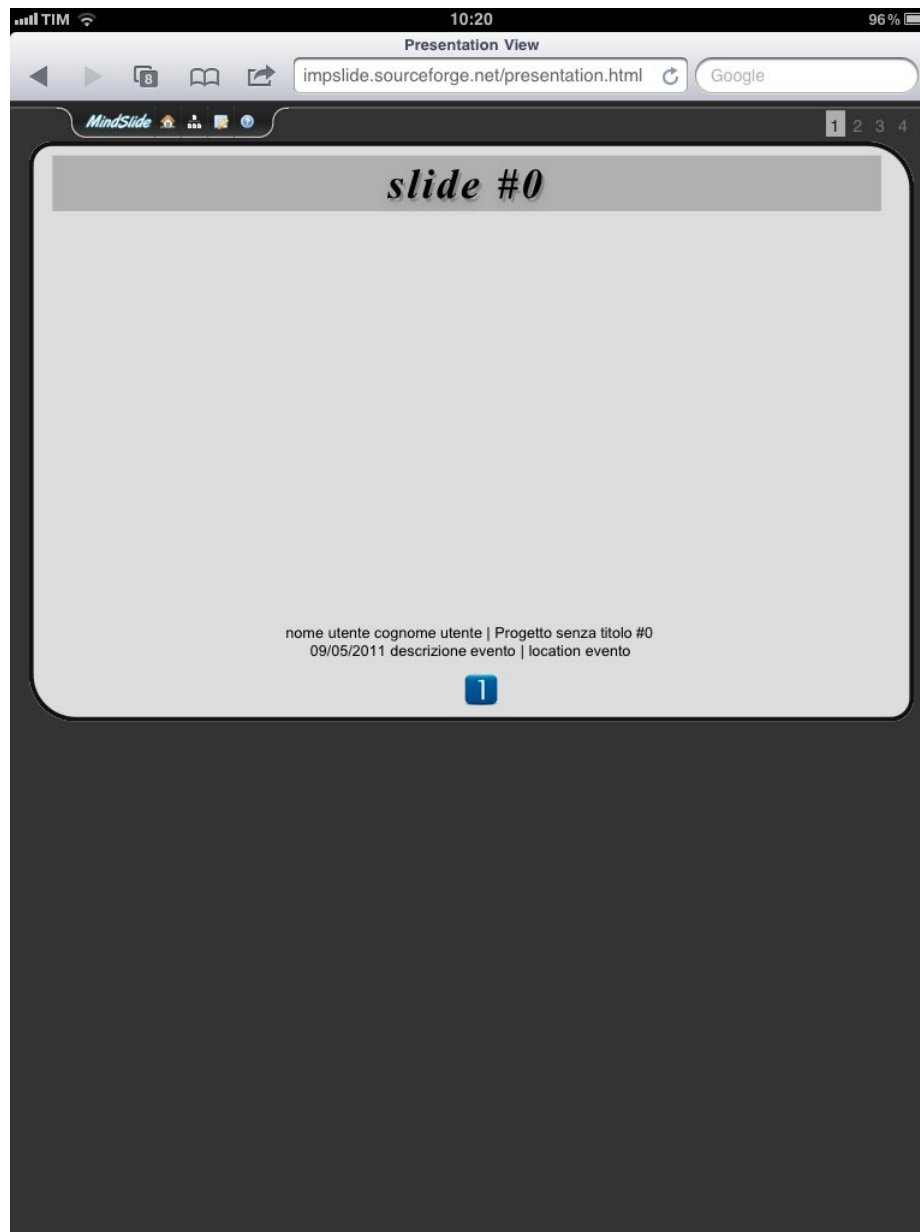


Figura 9: Presentation View (orientamento portrait)

Specifica tecnica

Versione: 1.0

Mail: mattiaglove@gmail.com

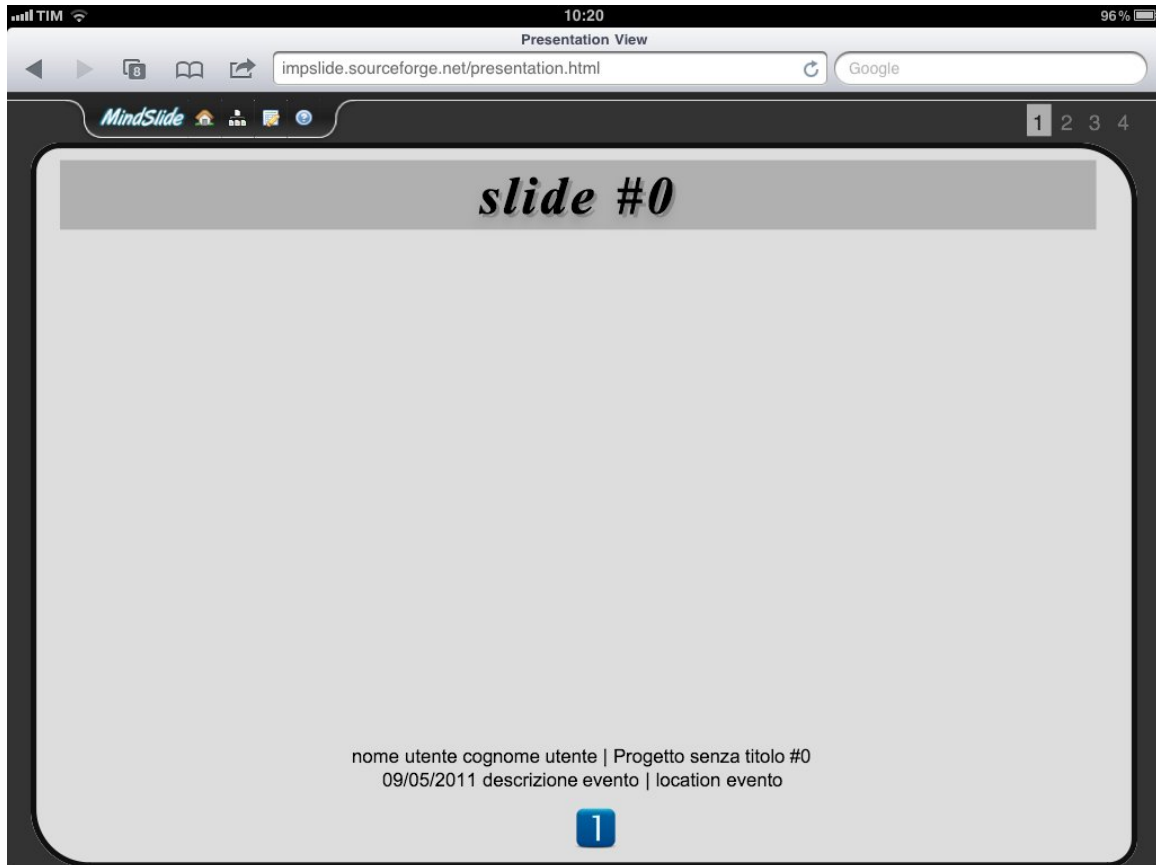
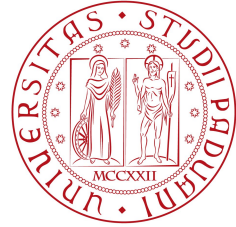
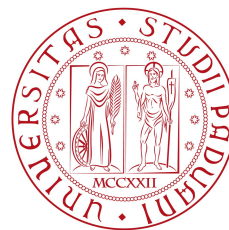


Figura 10: Presentation View (orientamento landscape)

La `presentation view` viene caricata e visualizzata correttamente anche su dispositivo mobile (come da figure [Presentation View \(orientamento portrait\)](#) (9) e [Presentation View \(orientamento landscape\)](#) (10)): si riesce correttamente a leggere le `slide`, a passare a quelle successive/precedenti ed a raggiungere `slide child/slide padre`. La possibilità di scorrere la presentazione è quindi correttamente funzionante.

Mancano tuttavia alcune funzionalità che richiedono l'utilizzo della tastiera, come ad esempio la possibilità di scorrere le `slide` velocemente utilizzando scorciatoie (e non cliccando con il mouse sul menù di navigazione), la possibilità di cambiare temi, etc. Si tratterà in maniera approfondita delle implementazioni da apportare alla `presentation view` nel capitolo [Presentation View Mobile](#) (4.6).



4 Realizzazione di MindSlide Mobile

In questo capitolo verranno evidenziate le modifiche e le implementazioni che verranno applicate all'attuale **MindSlide**, evolvendolo così nella versione **mobile**.

Prima di passare all'analisi approfondita delle funzionalità da implementare per ogni interfaccia di **MindSlide** è importante far notare che, come detto nei paragrafi [MindSlide su touch screen](#) (3) e [Homepage](#) (3.1) attualmente nessuna interfaccia di **MindSlide** è dotata di **CSS** dedicato ai dispositivi **mobile** (compreso l'orientamento **portrait/landscape**) e di **JavaScript** che distingua il **touch** dal click del mouse (ergo ogni interfaccia risente del bug "simulazione dell'**hover**").

Il problema della [Simulazione dell'**hover**](#) (4.1) verrà discusso di seguito in generale e non nelle singole sezioni dedicate alle interfacce in quanto esso affligge l'intero sistema e parlarne in ogni sezione sarebbe ridondante.

Il problema del [CSS](#) (4.2) verrà discusso di seguito in generale e approfondito in ogni sezione con descrizioni specifiche delle implementazioni che si intendono attuare.

4.1 Simulazione dell'**hover**

Come detto in precedenza, questo effetto è causato dall'interpretazione erranea del dispositivo **touch** degli eventi puramente dedicati al mouse, come **hover**, **focus**, **out**, etc.

Per ovviare a questo problema basta semplicemente gestire via **JavaScript** il riconoscimento dei dispositivi di input che il device mette a disposizione ed abilitare determinati comportamenti a secondi di quale periferica è disponibile (es: disabilitare le funzioni **hover** in assenza del mouse). In questo modo non vi è più alcuna possibilità di interpretazioni ed il dispositivo esegue sempre l'azione corretta a seconda del device su cui viene eseguito, massimizzando le prestazioni ed eseguendo sempre ciò che l'utente si aspetta.

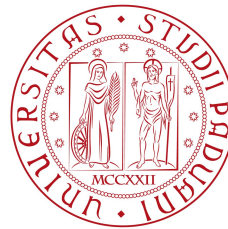
4.2 CSS

Attualmente nessuna interfaccia del software dispone di fogli di stile pensati per i dispositivi **mobile**: tutte le visualizzazioni sono gestite dal dispositivo stesso, scelta non sempre ottimale. La scelta che si intende attuare è, in generale, quella di fornire un **resize** della pagina a seconda dell'inclinazione del dispositivo e di massimizzare la visibilità degli elementi.

4.3 Homepage Mobile

La homepage non presenta particolari funzionalità o operazioni che possono essere massimizzate con l'uso del **touch**. Verrà fornito uno stile apposito per i dispositivi **mobile**: le descrizioni saranno ridotte al minimo in modo tale da lasciare più spazio ai pulsanti ed ai link ed i pulsanti saranno ingranditi in modo da facilitarne il **touch**. Per quanto riguarda l'orientamento del **device** le modifiche che si andranno ad implementare riguardano la disposizione degli elementi a schermo sono le seguenti:

- *Orientamento **portrait*** - la disposizione degli elementi verrà lasciata "a pila": l'orientamento verticale del dispositivo permette di visualizzare tutti gli elementi uno sopra l'altro senza perderne la visibilità. La pagina stretta e lunga si presenta in maniera gradevole e il suo utilizzo è semplice ed intuitivo. Il layout sarà quindi lasciato su un'unica "colonna";
- *Orientamento **landscape*** - la pagina verrà suddivisa in due "colonne": gli elementi relativi alle presentazioni esistenti saranno posti sulla sinistra in uno spazio che occuperà circa l'80% della pagina, mentre i link informativi saranno posti nel restante 20%. In questo modo non sarà più obbligatorio scorrere la pagina verso il basso per avere le informazioni ricercate.



4.4 MindMap View Mobile

Nella modalità **MindMap View** verranno eseguite diverse implementazioni che permetteranno di massimizzare l'esperienza **touch** nell'uso dell'applicazione:

- **touch** - il **touch** su una **slide** scatenerà 3 eventi in simultanea:
 1. comparsa del gritter di anteprima della **slide** toccata;
 2. evidenziazione (con un colore più scuro) dell'intera **slide list** di cui fa parte la **slide** toccata;
 3. apertura del PieMenu sulla **slide** toccata.

Questa serie di operazioni rende visibile il contenuto in anteprima della **slide** toccata, ne evidenzia la sequenza di appartenenza e consente di iniziare subito ad eseguire operazioni sulla **slide** toccata, massimizzando le prestazioni dell'intero sistema con un singolo tocco;

- **drag&drop** - la **mindmap** sarà liberamente spostabile all'interno dell'area di lavoro tramite **drag&drop** ed il suo spostamento non causerà movimenti agli altri elementi (project label, gritter, etc.). Questo permetterà di poter spostarsi lungo la **mindmap** senza perdere di vista elementi essenziali, come ad esempio i link alle altre interfacce;
- **pinch e reverse-pinch** - la **mindmap** sarà liberamente ingrandibile/riducibile all'interno dell'area di lavoro tramite **pinch e reverse-pinch** ed il suo ridimensionamento non causerà effetti sugli altri elementi (project label, gritter, etc.). Questo permetterà di poter ridimensionare la **mindmap** senza perdere di vista elementi essenziali, come ad esempio i link alle altre interfacce.

La **mindmap** si sviluppa sia in verticale che in orizzontale; non verranno perciò fatte particolari distinzioni sulla disposizione degli elementi tra orientamento **landscape** e **portrait** del dispositivo e verrà mantenuto il comportamento di default del **device**.

4.5 Slide View Mobile

L'interfaccia **Slide View** non presenta particolari funzionalità o operazioni che possono essere massimizzate con l'uso del **touch**. Verrà effettuato un controllo **JavaScript** che disabiliterà le chiamate alla libreria *TinyMCE* durante l'uso dell'applicazione su dispositivo **mobile**, evitando così gli errori generati dall'incompatibilità tra dispositivo ed editor. Gli inserimenti di titolo e contenuto della **slide** saranno quindi di testo puro e saranno inseriti tramite tastiera virtuale del **device mobile**.

Il comportamento di default del dispositivo al cambio di orientamento verrà disabilitato: invece di subire un **resize** la **slide** editabile occuperà sempre tutta l'altezza disponibile della pagina, sia in **landscape** (visualizzazione consigliata) che in **portrait**, evitando così il poco gradevole spazio vuoto lasciato di default (si veda l'immagine [Slide View \(orientamento portrait\)](#) (7)).

4.6 Presentation View Mobile

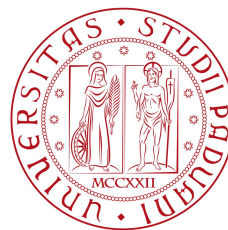
Nella modalità **Presentation View** verranno eseguite diverse implementazioni che permetteranno di massimizzare l'uso dell'**accelerometro** (nei dispositivi compatibili):

- **orientamento** - tutti i comportamenti di default attuati dal dispositivo al cambio di orientamento verranno disabilitati; la visualizzazione standard sarà quella in **landscape** poiché essa richiama la forma di una **slide** classica (rettangolo con lati orizzontali più lunghi di quelli verticali) e non sarà disponibile un orientamento **portrait**;
- **passaggio alla slide successiva** - l'orientamento del dispositivo verso destra innescherà il passaggio alla **slide** successiva (se presente, altrimenti non causerà nessun effetto);
- **passaggio alla slide precedente** - l'orientamento del dispositivo verso sinistra innescherà il passaggio alla **slide** precedente (se presente, altrimenti non causerà nessun effetto);

Specifica tecnica

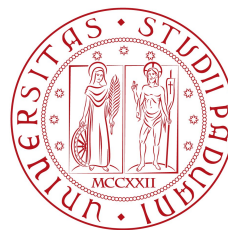
Versione: 1.0

Mail: mattiaglove@gmail.com



- *passaggio alla slide padre* - l'orientamento del dispositivo verso l'alto innescherà il passaggio alla **slide** padre (se presente, altrimenti non causerà nessun effetto).

Data la possibilità della presenza di più di una **slide** figlia è stata scartata l'idea di un passaggio ad una di queste tramite inclinazione del dispositivo, poiché sarebbe stato difficile capire quale **slide child** l'utente intenda raggiungere compiendo questo gesto.



5 Strumenti e risorse

In questo capitolo verranno elencate le risorse [Hardware \(5.1\)](#) e [Software \(5.2\)](#) che si intendono utilizzare nella realizzazione di MindSlide Mobile.

5.1 Hardware

- Personal Computer con S.O. *Microsoft Windows 7*;
- *Apple iPad 1.0*;
- *Apple iPhone 3.0*.

5.2 Software

5.2.1 Linguaggi

- HTML5;
- CSS;
- JavaScript.

5.2.2 Librerie

- *Scripty2*:
<http://scripty2.com>
- *jQueryMobile*:
<http://jquerymobile.com>

5.2.3 Editor

- *Eclipse 3.5.2* (per la realizzazione della documentazione in \LaTeX):
<http://www.eclipse.org>
- *Notepad++ 5.9* (per la realizzazione del codice Javascript, HTML5 e CSS):
<http://notepad-plus-plus.org>.