

# WEBClicker - Progetto HCI

Marco Loschiavo  
Università degli studi di Firenze  
marco.loschiavo1@stud.unifi.it

## Sommario

*L'uso sempre piu' frequente di compuer e L<sub>smartphone</sub> ha reso necessario che le applicazioni siano facilmente utilizzabili dall'utente finale. La Human Computer Interaction fornisce importanti metodologie per la loro progettazione finalizzata ad un'efficiente interazione. In questo elaborato è stata realizzata una web application che fornisce un servizio in ambito scolastico. Il lavoro è stato suddiviso in 3 fasi principali: nella prima fase è stato effettuato uno studio preliminare per capire quali fossero i requisiti e le criticità del servizio, poi e' stata realizzata l'app ed infine sono stati eseguiti dei test per avere una stima della performance e dell'usabilità.*

*Il codice del progetto è disponibile su: [Github](#).*

## Introduzione

WEBClicker è una applicazione web che fornisce un servizio aggiuntivo in ambito scolastico a professori e studenti. L' app offre ai professori, la possibilità di poter creare e proiettare delle domande riguardanti argomenti del corso spiegati durante le lezioni agli studenti, i quali attraverso la rete, utilizzando il proprio smartphone, tablet o pc possano dare una risposta. Alla fine della votazione il professore può mostrare alla classe i risultati, i quali vengono riportati su un grafico. Questo consente agli studenti di verificare di aver ben compreso la lezione e allo stesso tempo mostra al professore il livello di apprendimento della classe.

Nelle sezioni successive verranno discusse le fasi realizzative dell'applicazione, partendo da un primo processo di Needfinding, dalla crea-

zione di un Mock-up, e per finire alla effettiva implementazione dell'applicazione.

## Processo di needfinding

Nella prima fase di sviluppo della nostra applicazione è stato seguito un processo di *Needfinding*. Il needfinding in generale è il processo in cui si osservano le persone e le loro abitudini per scoprire e capire quali sono le loro necessità ed abitudini. Da questo processo vengono estratte delle *Personas*, cioè degli attori che rappresentano gli utenti del nostro applicativo, con i quali vengono costruiti degli *Scenari*. Quest'ultimi sono contesti ideali in cui le personas utilizzano l'applicazione per soddisfare quelle che sono le loro necessità. Infine grazie alle descrizioni degli scenari sono stati ricavati i *Requisiti* dell'applicazione cioè quello che il prototipo deve includere per soddisfare le necessità degli utenti.

## Needfinding

Il primo passo è stato di ricercare articoli ed applicazioni simili a WEBClicker per capire se la sua realizzazione fosse veramente necessaria. Dalle ricerche sono state trovate alcune applicazioni simili, molte di esse a pagamento e altre con funzionalità che si discostano sostanzialmente dalle nostre idee. Ad esempio [Kahoot!](#) offre delle funzionalità simili a quelle da noi proposte, ma è stato implementato più come forma di gioco risultando quindi molto infantile e poco professionale. Un ulteriore app di questo genere è [iClicker](#), che offre una vasta gamma di servizi tra cui anche quelli di nostro interesse ma a pagamento.

Dopo la prima fase di ricerca, il passo successivo è stato quello di cercare le esigenze degli utenti. Per questa fase è stato deciso di effettuare brevi interviste a studenti e docenti al fine di valutare:

- l'interesse di un applicativo di questo tipo
- il contesto nel quale questo può essere utilizzato
- la frequenza con cui un'applicazione di questo tipo è utilizzata
- caratteristiche di un sistema di questo tipo ideale

Dalle interviste svolte è stato possibile vedere come sia necessario che il sistema risulti semplice ed intuitivo, e preferibilmente privo di installazioni.

Mediamente i professori sono più propensi all'utilizzo di PC portatili, in quanto molti di loro sono soliti spiegare proiettando delle slide. Gli studenti invece sono spesso sprovvisti di pc durante la lezione, ma nessuno tra gli intervistati ha dichiarato di non avere lo smartphone con sé. Inoltre è emerso essere necessario un sistema rapido, in quanto dati i tempi ristretti per svolgere tutto il programma didattico, i docenti non possono permettersi lunghi tempi di attesa. Questo ci ha portato a dover effettuare alcune scelte, e ci ha permesso di ricavare gli obiettivi principali:

- Realizzare un'applicazione web ibrida in modo da poter essere accessi sia da Smartphone che da Tablet che da PC.
- Limitare il processo di autenticazione al sito, rendendolo necessario solo per i docenti. Gli studenti non avranno bisogno di registrarsi per rispondere alle domande. Questo consentirà di ridurre i tempi di attesa durante le lezioni.
- Realizzare un'applicazione che sia facile ed intuitiva in modo da poter essere utilizzabile anche da utenti poco esperti.

## Personas

Dalle interviste condotte sono stati ricavati due principali tipi di *personas* per il lato docente:

- **Prof. Marini** : un docente di Analisi Matematica di 60 anni con una competenza limitata nell'ambito informatico, ma interessato e aperto alle innovazioni. Tiene particolarmente all'apprendimento degli studenti ed è solito chiedere alla classe se durante la spiegazione siano sortiti dei dubbi o se ci siano state cose spiegate in modo poco chiaro. Molto spesso però non riceve risposta anche a causa dell'imbarazzo dei singoli studenti.
- **Prof. Casetti** : un docente di Fisica di 42 anni alle prime armi che si trova spesso in difficoltà ad organizzare i giorni per effettuare i ricevimenti con gli studenti e i giorni in cui fissare gli appelli degli esami.

Per quanto riguarda il lato studente è stata estratta una singola *persona*:

- **Marco**: uno studente universitario frequentante che occasionalmente non comprende alcuni argomenti spiegati dal professore. Spesso si chiede se è il solo a non capire, o se la cosa è estesa a buona parte degli studenti presenti in aula.

## Scenari

- **Prof. Marini** , durante una lezione in classe, vuole fare delle domande agli studenti per verificare se essi abbiano o meno compreso quanto da lui spiegato. Sapendo quasi per certo di non ottenere risposta, a causa dell'imbarazzo dei singoli studenti, e venuto a conoscenza dell'esistenza della nostra applicazione, decide di provarla ad utilizzare.

- **Prof. Casetti**, vuole capire quando la maggior parte degli studenti sono liberi da altri corsi o impegni scolastici per decidere in quali giorni, tra i suoi disponibili, fissare i ricevimenti. Decide così di usare la nostra app per creare un sondaggio.

Gli scenari sopra riportati sono degli scenari d'uso, mentre il successivo sarà uno scenario di adozione.

- **Marco** non contento dell'insegnamento ricevuto e con il consenso degli altri studenti dei quali buona parte lamentano gli stessi problemi, decide di chiedere al professore di poter iniziare ad utilizzare la nostra applicazione, in modo che possa avere un feedback sulla lezione tenuta.

## Requisiti

Il processo di needfinding getta le basi ai requisiti da raggiungere al fine di realizzare un applicativo accessibile e funzionale che si adatti al meglio agli utenti di nostro interesse.

In dettaglio sono stati estratti i requisiti principali, i quali sono stati riportati nella lista seguente:

- Interfaccia di autenticazione per i docenti.
- Interfaccia personale del docente nella quale potrà inserire e gestire una serie di domande. Inoltre dovrà essere possibile mostrare alla classe la domanda con possibili risposte.
- Possibilità di poter inserire le domande direttamente nelle slide, senza dover utilizzare l'applicazione
- Interfaccia di risposta alla domanda per gli studenti

## Mockup e Diagramma di Flusso

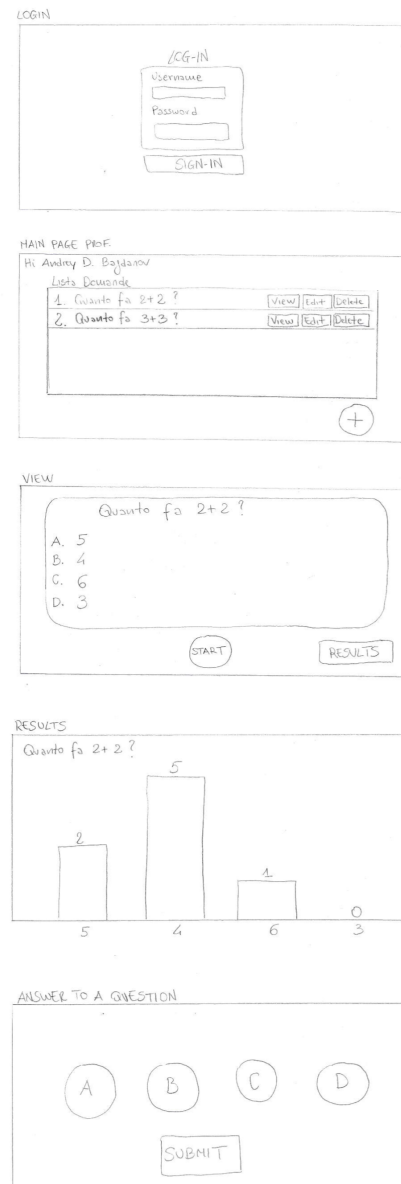


Figura 1: Mockup

Nella fase che ha preceduto l'implementazione sono stati realizzati dei mockup che descrivono le viste dell'applicazione, in modo da avere una riproduzione dell'interfaccia finale. Un mockup è una rappresentazione statica del pro-

dotto, realizzabile utilizzando diverse tecniche. Nel nostro caso è stata utilizzata la paper prototyping. In Figura 1 sono riportati i mockup dell'applicazione.

Inoltre, si è reso opportuno un diagramma di flusso in modo da chiarire la sequenza del programma(vedi Figura 2).

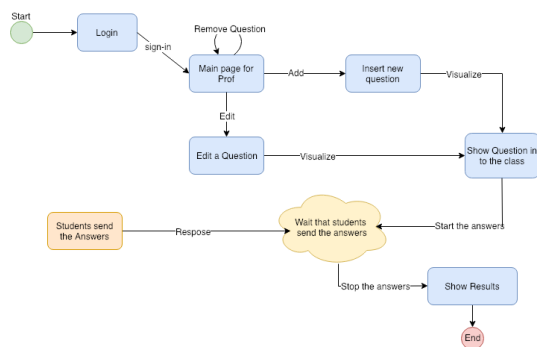


Figura 2: Diagramma di Flusso

## Applicazione

Una volta estratti i requisiti dalla fase di needfinding, il lavoro si è concentrato sulla fase realizzativa dell'app.

Sono stati reliazzati sia il lato client, utilizzando tecnologie come HTML5, CSS, e JavaScript, oltre ai framework Bootstrap[1] e jQuery[3], sia il lato server adoperando il linguaggio PHP[4] per manipolare i dati e per interrogare il nostro Database.

## Implementazione

Il principale strumento utilizzato per lo sviluppo dell'applicazione è stato PHPStorm, un ambiente di sviluppo integrato (IDE) sviluppato dalla JetBrains, che fornisce numerosi strumenti per una vasta gamma di linguaggi di programmazione. L'implementazione è strutturata secondo il pattern architetturale MVC (Model View Controller) così da rispettare il principio della *separation of concern*. Que-

sto permette di suddividere un'applicazione interattiva in tre componenti:

- **Model:** contiene i dati principali e le relative funzionalità.
- **View:** è come i dati sono rappresentati all'interno dell'app.
- **Controller:** gestisce l'input e fa da intermediario tra vista e modello.

Possiamo pensare alle interfacce grafiche (HTML, CSS) come le viste della nostra applicazione, mentre come modello tutta la parte inerente la gestione dei dati e del database. Il controller invece è la parte della nostra app che gestisce le interazioni dell'utente, ed in base a queste va a modificare il modello. Un'esempio di ciò potrebbe essere un utente che preme il pulsante di Reset (la vista permette di visualizzare all'utente tale bottone). A questo punto il compito viene passato al controller, nel nostro caso implementato utilizzando codice Javascript, il quale mediante tecnologia AJAX[2], va ad effettuare operazioni sul Database coinvolgendo il modello.

## Interfaccia e Funzionalità

Recandoci alla pagina iniziale dell'applicazione, viene mostrata la sezione di Login (Figura 3), nella quale un docente, immettendo i propri dati personali, viene reindirizzato nella propria pagina personale (Figura 4).

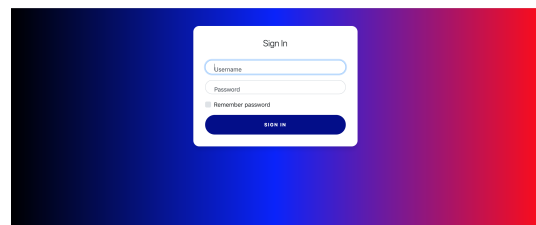


Figura 3: Interfaccia di Login

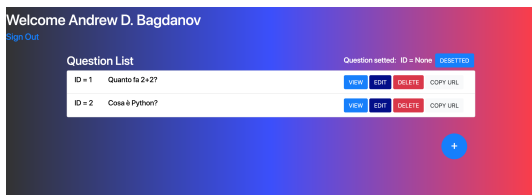


Figura 4: Interfaccia principale del docente

Adesso il professore cliccando sul bottone  $+$  potrà inserire una nuova domanda che andrà ad aggiungersi alla lista di quelle create precedentemente. Sarà consentito, tramite i due bottoni  $+$  e  $-$ , inserire domande contenenti un minimo di 2 ed un massimo di 5 possibili risposte. L'interfaccia che consente di creare una nuova domanda è mostrata in Figura 5.

L'applicazione offre anche la possibilità di rimuovere e modificare delle domande presenti nella lista, mediante i bottoni *delete* e *edit*. L'interfaccia che permette di modificare un domanda è la stessa di quella in Figura 5.

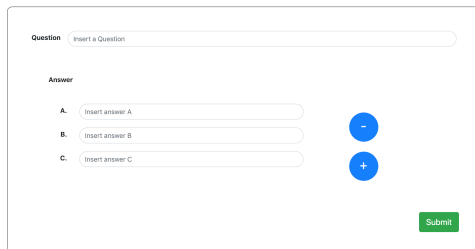


Figura 5: Interfaccia di inserimento nuova domanda

Premendo il pulsante *view* relativo alla domanda di nostro interesse, essa, con le relative risposte, viene impaginata in modo (facendo uso di un proiettore) da poter essere mostrata alla classe (Figura 6). Da qui sarà possibile, attraverso i bottoni di *start* e di *stop* far iniziare e finire il sondaggio, e attraverso il pulsante *results* mostrare i risultati.



Figura 6: Interfaccia di presentazione domanda

Il docente dovrà fornire l'URL che consentirà agli studenti di poter rispondere a tale domanda. L'URL sarà sempre la stessa, quindi non sarà necessario fornirla ogni volta.

Gli studenti tramite smartphone o pc potranno a questo punto rispondere alla domanda posta. Essi visualizzeranno un'interfaccia come quella riportata in Figura 7.



Figura 7: Interfaccia di risposta alla domanda

Non appena tutti gli studenti hanno risposto al quesito, il docente può mostrare alla classe i risultati (Figura 8)

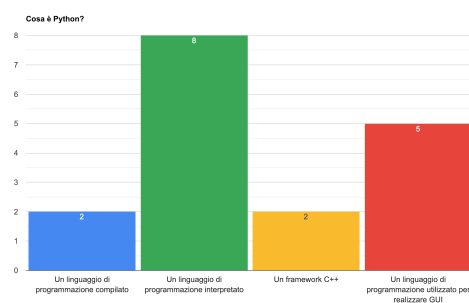


Figura 8: Grafico dei risultati mostrati alla classe

## Usability Test

Durante la fase finale del progetto è stato svolto un usability test, per capire quanto il

prototipo costruito fosse user-friendly per gli utenti del servizio. Dunque l'obiettivo del test era quello di riuscire a valutare alcuni aspetti dell'applicazione come la sua intuitività, la sua facilità d'uso e il suo aspetto. Nel nostro caso sono stati fatti due usability test, uno dedicato agli studenti ed uno ai docenti.

### Caso studente

I candidati sono stati acquisiti all'interno dell'istituto Santa Marta di Firenze e tutti risultano essere degli studenti. In totale le persone partecipanti al test sono stati 9. Una volta selezionati i candidati sono stati portati in un aula del plesso e disposti in ordine casuale. Successivamente gli sono state spiegate le caratteristiche e le funzionalità dell'applicazione senza entrare nel dettaglio, in modo da non falsare il test. Una volta spiegato il contesto nel quale si trovano, facendo riferimento agli scenari descritti precedentemente, gli sono state mostrate 3 domande alle quali è stato chiesto di rispondere. Alla fine di ogni votazione sono stati mostrati i risultati. In sintesi i task al quale sono stati sottoposti i candidati sono stati:

- Visualizzare la domanda proposta dal docente
- Rispondere a tale domanda collegandosi all'indirizzo fornito
- Visualizzare i risultati al test

### Caso docente

Anche per quanto riguarda il lato docente del test, per motivi di praticità, sono stati presi dei candidati di tipo studente. In particolare sono stati presi 7 candidati diversi dai precedenti, in modo che anche per loro sia la prima volta in cui provano l'applicazione. In una prima fase, come nel caso precedente, è stata fatta una breve introduzione all'app, e spiegato il caso d'uso.

Successivamente per ogni candidato sono stati assegnati dei task da compiere:

- Accedere alla propria pagina personale inserendo le credenziali fornite
- Creare una serie di domande (almeno tre)
- Modificare una domanda precedentemente creata
- Cancellare una domanda precedentemente creata
- Far visualizzare alla classe due domande e consentire agli studenti di poter rispondere
- Mostrare i risultati

Alla risposta delle domande hanno partecipato 6 dei 9 candidati che hanno svolto il precedente test.

### Questionari

Alla fine del test, ad ogni candidato è stato chiesto di compilare un questionario. Sostanzialmente sono stati creati due diversi questionari, uno da far compilare al docente ed uno allo studente.

Le domande sottoposte sono di tipo SEQ con una scala che va da 1 a 7 dove 1 rappresenta fortemente disaccordo e 7 fortemente d'accordo; per richiamare l'attenzione del candidato alcune domande sono poste in modo inverso. Complessivamente, tra test e questionario, ogni utente ha impiegato circa 15 minuti. In tabella 1 e 2 è riportato un quadro generale dei risultati ottenuti dal test.

N.	Domanda	Val. Media	$\sigma$
1	L'interfaccia mostrata dopo il login, è risultata essere chiara	6.14	0,63
2	Inoltre è risultata gradevole	5.29	1.03
3	È stato difficile e poco intuitivo inserire una domanda	1.29	0.45
4	La gestione delle domande (modifiche, cancellazioni) è risultata efficiente	6.43	0.49
5	È stato intuibile far visualizzare agli studenti la domanda, permettendo loro di rispondere	6.29	0.70
6	È stato difficile mostrare i risultati ottenuti dal sondaggio agli studenti	1.14	0.35
7	Trova chiara la navigabilità dell'applicazione	4.57	1.05
8	Il servizio fornito risulta esauriente	6.29	0.45
9	In generale sono soddisfatto dell'app	6.22	0.42
10	Pensi che l' applicazione possa risultare utile	6.57	0.49

Tabella 1: Domande poste ai docenti, con le rispettive valutazioni. Ogni risposta va in una scala da 1 (fortemente disaccordo) a 7 (fortemente d'accordo).  $\sigma$  è la deviazione standard.

N.	Domanda	Val. Media	$\sigma$
1	L' interfaccia inerente la proiezione di domande risulta essere efficace alla loro comprensione	6	0.66
2	Avresti preferito che ci fossero state riportate anche le possibili risposte	6	1.05
3	Avresti preferito ci fosse stata riportata anche la domanda	5.22	1.68
4	Inoltre è risultata gradevole	5.33	0.66
5	È stato semplice rispondere alla domanda	6.44	0.68
6	Trovi difficoltoso dover inserire l'indirizzo fornito per rispondere alla domanda	2.22	0.63
7	Il grafico dei risultati risulta essere comprensibile	6.44	0.68
8	Il servizio fornito risulta esauriente	6.67	0.47
9	In generale sono soddisfatto dell'app	6.44	0.50
10	Pensi che l' applicazione possa risultare utile	6.22	0.42

Tabella 2: Domande poste agli studenti, con le rispettive valutazioni. Ogni risposta va in una scala da 1 (fortemente disaccordo) a 7 (fortemente d'accordo).  $\sigma$  è la deviazione standard.

## Analisi

In generale, come possiamo vedere dalle domande 8,9,10 dei due questionari, gli utenti sono rimasti soddisfatti dall'applicazione, reputandola utile. Inoltre è stato riscontrato un buon funzionamento medio dell'applicazione, anche se sono state rilevate alcune criticità, le quali verranno discusse nelle sottosezioni successive suddivise per studenti e docenti.

## Studenti

Per quanto riguarda gli studenti, il nostro interesse era focalizzato sulla praticità e sulla semplicità nel rispondere alle domande ed alle preferenze dei singoli sulla interfaccia di risposta alla domanda. Per quanto riguarda il primo obiettivo, i risultati ottenuti dalle domande 5 e 6, sono più che soddisfacenti. Durante il test però, anche se i candidati erano un numero molto ridotto rispetto a quello reale degli studenti presenti alla lezione, ci siamo accorti che la comunicazione dell'indirizzo potrebbe essere un problema con l'aumentare del numero di persone. Dopo un confronto, avendo tenuto presente anche che l'indirizzo è univoco ed invariante, i candidati hanno proposto di inserire l'indirizzo all'interno dell'interfaccia che riporta la domanda con le possibili risposte. Infatti anche se l'indirizzo resta sempre lo stesso, questo non è altrettanto vero per gli studenti, i quali potrebbero seguire le lezioni saltuariamente o iniziare a frequentare il corso, con quest'ultimo già in svolgimento.

Per quanto riguarda l'interfaccia di risposta alla domanda, possiamo vedere come la grande maggioranza degli studenti avrebbe preferito che ci fossero state riportate sia le possibili opzioni di risposta che la domanda. Ciò inizialmente non fu stato fatto, in quanto ritenuto dopo una prima analisi essere ridondante con le opzioni di risposta proiettate. Secondo i candidati però l'aggiunta di queste caratteri-

stiche renderebbe più comprensibile la fase di risposta.

## Docenti

Per quanto riguarda la parte dell'app dedicata al docente il nostro interesse era quello di valutare quanto fosse user-friendly, valutando anche l'aspetto implementativo. Come si può vedere dalle domande 1,3,4,5 e 6 l'applicazione è risultata essere molto intuitiva e facile da usare. L'unica criticità riscontrata è stata per quanto riguarda la navigabilità dell'applicazione, risultata non intuitiva, in quanto non presenti pulsanti di back e di ritorno alla home page. Questo problema è stato successivamente risolto, aggiungendo i signifier mancanti.

## Suggerimenti

Tra i suggerimenti rilasciati dai candidati sono stati trovati interessanti i seguenti:

- Effettuare alcune migliorie per quanto riguarda la parte di Design
- Includere una funzionalità che consenta di poter inserire anche un'immagine a fianco della domanda.
- Estendere le funzionalità dell'app permettendo di poter rispondere alle domande anche se non presenti alla lezione (ovviamente a domande non riguardanti la spiegazione. Vedere scenario del Prof. Casini).

## Conclusioni

I suggerimenti presentati nella sezione precedente gettano le basi per gli sviluppi futuri, andando così a migliorare ed ad arricchire le funzionalità.

In conclusione, anche se durante la fase di usability test, sono emerse alcune criticità, queste non hanno gravato sul prodotto finale, il



quale è stato apprezzato e reputato utile sia per gli studenti che per i docenti. Studenti e professori avranno quindi un ulteriore strumento per poter svolgere al meglio il proprio lavoro.

## Riferimenti bibliografici

- [1] Bootstrap. *open source toolkit for developing with HTML, CSS, and JS*. <https://getbootstrap.com/>.
- [2] Jesse James Garrett. *A New Approach to Web Applications*. <https://www.adaptivepath.org/ideas/ajax-new-approach-web-applications/>.
- [3] jQuery. *A JavaScript library*. <https://jquery.com/>.
- [4] Jon A. Phillips Michele E. Davis. *Programming in PHP & MySQL*.