

Scuola Arti e Mestieri Trevano

Sezione informatica

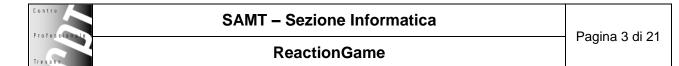
Carthafind

Titolo del progetto: Carthafind Nadir Barlozzo

Classe: I3BB
Anno scolastico: 2017/2018
Docente responsabile: Luca Muggiasca



1 Introduzione	3
1.1 Informazioni sul progetto	
1.2 Abstract	3
1.3 Scopo	
Analisi	
1.4 Analisi del dominio	3
1.5 Analisi dei costi e benefici	
1.6 Analisi e specifica dei requisiti	4
1.7 Pianificazione	
1.7.1 Software	8
1.7.2 Hardware	8
2 Progettazione	8
2.1 Design dei dati e database	8
2.2 Design delle interfacce	g
3 Implementazione	11
3.1 Grafica	
3.2 Database	11
3.3 PHP	12
3.4 Javascript (Jquery)	
4 Test	17
4.1 Mancanze/limitazioni conosciute	18
5 Consuntivo	
6 Conclusioni	20
6.1 Sviluppi futuri	20
6.2 Considerazioni personali	20
7 Bibliografia	20
7.1 Sitografia	20
8 Allegati	20
9 Glossario	



1 Introduzione

1.1 Informazioni sul progetto

Docente Responsabile: Luca Muggiasca Scuola: Scuola d'Arti Mestieri Trevano

Sezione: Informatica Materia: Modulo 306

Data inizio progetto: 16.03.2017 Data consegna progetto: 18.05.2018

1.2 Abstract

As the number of student's developed projects in our school section increases every year, we need a method to keep track of them and provide us long-term availability. I was told to produce a tool to facilitate researches and provide a storage place to these projects.

1.3 Scopo

Lo scopo del progetto consiste nel creare un applicativo utilizzato per la ricerca avanzata tra i progetti già conclusi alla Scuola Arti e Mestieri di Trevano, tramite parole chiave, descrizioni, titoli o docenti responsabili.

Analisi

1.4 Analisi del dominio

Il metodo utilizzato fino ad ora consiste in una banca dati contenente i dati dei progetti e accessibile quindi solamente tramite interrogazioni SQL effettuate da linea di comando o programmi appositi come HeidiSQL o WorkbenchSQL. Questo applicativo online ha l'obbiettivo di rendere la ricerca di informazioni più veloce e user-friendly, dato che non risulta esistere niente di simile in commercio.

1.5 Analisi dei costi e benefici

Categoria	Costo
Spazio web	30 CHF
Personale	76 ore * 1 persona * 40 CHF/h = 3040 CHF
TOT	3070 CHF

Nadir Barlozzo 3 Versione: 21.05.2018



ReactionGame

Pagina 4 di 21

1.6 Analisi e specifica dei requisiti

	ID: REQ-01				
Nome	Gestione registrazione				
Priorità	1				
Versione	1.0				
Note					
Sotto requisiti					
001	Deve esserci un form per la registrazione di un nuovo utente				
002	Deve esserci un campo per il nome di tipo testo				
003	Deve esserci un campo per il cognome di tipo testo				
004	Deve esserci un campo per il nickname di tipo testo				
005	Deve esserci un campo per la password di tipo testo codificato				
006	Deve esserci un campo per l'email di tipo testo e i domini accettati sono solo samtrevano ed edu				
007	Deve essere inviato un link per confermare la registrazione sull'email registrata				

ID: REQ-02						
Nome	Gestione login e logout					
Priorità	1					
Versione	1.0					
Note	Note					
Sotto requisiti						
001	Deve esserci un form per il login di un utente					
002	Deve esserci un campo per l'email di tipo testo e i domini accettati sono solo samtrevano ed edu					
003	Deve esserci un campo per la password di tipo testo codificato					
004	Deve esserci un pulsante per effettuare il logout dell'utente					



Pagina 5 di 21

ID: REQ-03				
Nome	Ricerca dei progetti			
Priorità	1			
Versione	1.0			
Note				
	Sotto requisiti			
001	Deve esserci un campo di tipo testo per inserire la parola da ricercare			
002	Deve esserci una tabella contenente i dati dei progetti			
003	Deve esserci un campo della tabella per il titolo			
004	Deve esserci un campo della tabella per la descrizione			
005	Deve esserci un campo della tabella per nome e cognome del responsabile			
006	Deve esserci un campo della tabella per contenere gli autori			
007	Deve esserci un campo della tabella per la durata			
008	Deve esserci un campo della tabella per il voto finale			
009	Deve esserci un campo della tabella per la percentuale di completamento			
010	Deve esserci un campo della tabella per i commenti aggiuntivi			

	ID: REQ-04				
Nome	Nome Gestione inserimento nuovo progetto				
Priorità	1				
Versione	1.2				
Note					
	Sotto requisiti				
001	Deve esserci un form per la registrazione di nuovi progetti				
002	Deve esserci un campo per il titolo di tipo testo				
003	Deve esserci un campo per la descrizione di tipo testo				
004	Deve esserci un campo per la scelta del responsabile di tipo select				
005	Deve esserci un campo per la scelta dei partecipanti di tipo parallel select				
006	Deve esserci un campo per la durata di tipo intero				



Pagina 6 di 21

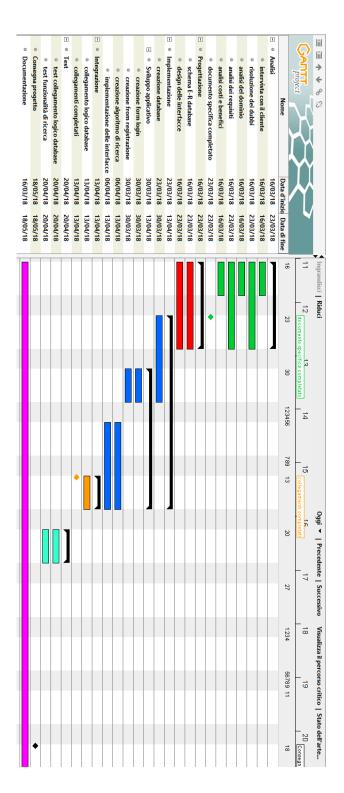
007	Deve esserci un campo per il voto finale di tipo intero (1-6)			
008	Deve esserci un campo per il completamento di tipo intero percentuale (0-100)			
009	Deve esserci un campo per commenti aggiuntivi di tipo testo			
010	Deve esserci un campo per l'inserimento di parole chiave inerenti di tipo CKI			
011	Deve esserci un campo per l'upload del file di documentazione di tipo file			



ReactionGame

Pagina 7 di 21

1.7 Pianificazione





ReactionGame

Pagina 8 di 21

1.7.1 Software

Microsoft Word 2013 Gantt Project 2.8.5 PowerPoint 2013 Notepad++ 7.5 Google Chrome 64.0 TortoiseGit 2.5.0.0 DbSchema 6.0.3 BootstrapStudio 4.1.7

1.7.2 Hardware

PC portatile

2 Progettazione

2.1 Design dei dati e database

Il database è relativamente semplice, composto da 8 tabelle delle quali 4 relazionali (**contains**, **participates**, **teacher** e **author**), e le altre 4 indipendenti (**project**, **keyword**, **user** e **verify**).

- La tabella **project** contiene tutte le informazioni primitive del progetto quali titolo, descrizione, durata, voto finale, progresso, commenti aggiuntivi, responsabile e percorso alla sua documentazione.
- La tabella **keyword** contiene i nomi delle parole chiave esistenti e relazionabili al progetto.
- La tabella **user** contiene le informazioni relative agli utenti registrati a Carthafind quali nome, cognome, grado di privilegi, nome utente, password ed email.
- La tabella **verify** contiene i codici di attivazione per confermare la registrazione al sito, vengono conservati finché l'utente non ne fa uso.

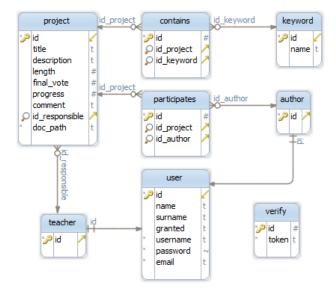


Figura 1 Design database Carthafind, fatto con DbSchema



ReactionGame

Pagina 9 di 21

2.2 Design delle interfacce





Registrati

Figura 2 Interfaccia di login

Registrazione



Esegui il login

Figura 3 Interfaccia di registrazione



Pagina 10 di 21

ReactionGame

Carthafind - Strumento di ricerca per progetti SAMT

Alcuni dei progetti creati nel nostro istituto dopo la loro conclusione vengono dimenticati.

Lo scopo di questo applicativo è schedare e classificare ogni progetto per poi metterli alla portata degli utenti, quali possono essere persone sia interne che esterne alla scuola.

Potete inoltre tramite la barra di ricerca filtrare i prodotti tramite parole chiavi così da avere solo quelli che più vi interessano.



Titolo	Descrizione	Responsabile	Autore/i	Durata	Voto finale	Completamento	Commmenti
Codice Morse	Sistema di traduzione da codice morse a testuale mostrato su display	Massimo Sartori	Davide Paradiso, Patrick Sartori, Federico Agosta, Alessandro Gomes,	96 ore	5	70%	

Figura 4 Interfaccia per utente registrato

Aggiungi un progetto

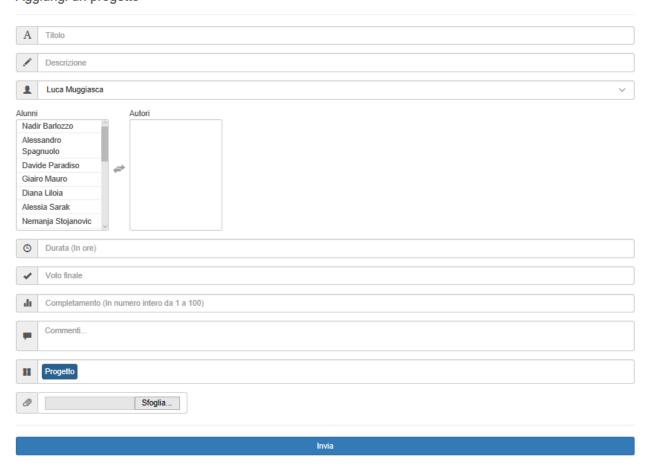


Figura 5 Interfaccia supplementare alla 4 se si ha privilegi da amministratore

Professionale

SAMT - Sezione Informatica

ReactionGame

Pagina 11 di 21

3 Implementazione

3.1 Grafica

L'implementazione del progetto è partita dal creare le interfacce precedentemente sviluppate su Bootstrap Studio, ha richiesto meno del previsto in quanto la semplicità e la vasta documentazione di Bootstrap mi hanno permesso di combinare i componenti necessari senza nessun problema.

Arrivato a questo punto ho appurato che effettivamente le pagine non erano completamente responsive, pertanto mi sono dedicato un altro po' di tempo a capire i funzionamenti di impaginazione questo pacchetto per far sì che Carthafind fosse accessibile da molteplici dispositivi, come mi ero imposto dall'inizio.

Dopo aver analizzato la media queries di Bootstrap ho avuto una visione più ampia di come fare, unendole al sistema di griglie sono riuscito ad ottenere il risultato aspettato.

Media queries

We use the following media queries in our Less files to create the key breakpoints in our grid system.

```
/* Extra small devices (phones, less than 768px) */
/* No media query since this is the default in Bootstrap */

/* Small devices (tablets, 768px and up) */
@media (min-width: @screen-sm-min) { ... }

/* Medium devices (desktops, 992px and up) */
@media (min-width: @screen-md-min) { ... }

/* Large devices (large desktops, 1200px and up) */
@media (min-width: @screen-lg-min) { ... }
```

We occasionally expand on these media queries to include a max-width to limit CSS to a narrower set of devices.

```
@media (max-width: @screen-xs-max) { ... }
@media (min-width: @screen-sm-min) and (max-width: @screen-sm-max) { ... }
@media (min-width: @screen-md-min) and (max-width: @screen-md-max) { ... }
@media (min-width: @screen-lg-min) { ... }
```

Figura 6 Media queries di Bootstrap

Come ultimo punto della grafica vorrei indicare il file *carthafind.css*, il quale contiene regole di stile aggiuntive per la corretta visualizzazione del CKI, della select parallela e di una buona parte del sito.

3.2 Database

Finite le interfacce grafiche mi sono spostato sulla parte pratica, iniziando a creare il database sempre dallo schema che avevo fatto antecedentemente, l'ho dovuto ristrutturare un paio di volte in quanto mi ero accorto che alcune relazioni non avevano senso o erano poco utili da avere.

Il file carthafind.sql se eseguito creerà una copia identica del database usato in locale.

Nadir Barlozzo 11 Versione: 21.05.2018

Centro Profess**ions** le

SAMT - Sezione Informatica

ReactionGame

Pagina 12 di 21

3.3 PHP

La prossima fase è stata creare i file PHP che si occupassero di gestire le interrogazioni al database. La prima cosa che ho voluto fare è stato creare un file di connessione al database in modo da non avere lo stesso identico codice ridondante in ogni modulo, viene definito come segue:

```
define('DBHOST', 'localhost');
3
          define('DBUSER', 'root');
 4
          define('DBPASS', '');
5
          define('DBNAME', 'carthafind');
6
          $conn = new mysqli(DBHOST, DBUSER, DBPASS, DBNAME);
8
9
          if (!$conn) {
10
11
              die("Connessione al database fallita: " . $conn->connect_error);
12
```

Figura 7 Codice - Connessione

In questo script vengono definite quattro variabili globali ognuna delle quali contiene una parte dell'autenticazione per l'accesso al database locale, viene poi creato un oggetto di tipo *mysqli* che stabilisce la connessione ed infine controllato se quest'ultima è valida.

A seguire ho creato lo script di registrazione, incaricato di interrogare il database per l'inserimento di un nuovo utente ed inviare un email di conferma all'email passatagli.
I punti importanti del file sono definiti da:

```
function getToken($length) {
 6
 7
          $token = "";
          $codeAlphabet = "ABCDEFGHIJKLMNOPQRSTUVWXYZ";
 8
 9
          $codeAlphabet.= "abcdefghijklmnopqrstuvwxyz";
10
          $codeAlphabet.= "0123456789";
          $max = strlen($codeAlphabet);
11
12
          for ($i=0; $i < $length; $i++) {
13
14
               $token .= $codeAlphabet[random int(0, $max-1)];
15
16
17
          return $token;
18
```

Figura 8 Codice - Funzione che genera un token casuale di una lunghezza x

```
function sendMail($id, $email, $token) {
21
          $subject = "Conferma registrazione per Carthafind.ch";
22
          $to = Semail:
          $headers = "From: decima@carthafind.ch\r\n";
23
          $headers .= "Content-Type: text/html; charset=UTF-8\r\n";
24
25
          $body="Perfavore vada al seguente <a href='localhost:3600/Carthafind/verify.php?id=".$id."&token=".$token</pre>
26
               "'>link</a> per completare l'attivazione del suo account Carthafind";
27
          mail($to, $subject, $body, $headers);
28
```

Figura 9 Codice - Funzione che invia l'email di conferma con un link generato in HTML tramite il tag <a> il quale punta al file verify.php

Nadir Barlozzo 12 Versione: 21.05.2018

Professionale

SAMT - Sezione Informatica

ReactionGame

Pagina 13 di 21

Figura 10 Codice - Codifica della stringa password in un hash di tipo SHA256

Fatto lo script di registrazione ho scritto quello di login così da testare se le informazioni passate funzionassero come previsto.

Questo modulo si occupa di convalidare i dati inseriti, viene fatto tramite un controllo sull'email esistente o meno e, tramite una comparazione tra l'hash sul database e quello locale, dato che la codifica è one-way. Inoltre, ha il compito di inizializzare una sessione con l'id dell'utente che ha effettuato il login e reindirizzarlo sul file *home.php*.

```
if($count == 1 && $row['password'] == $password) {
    $_SESSION['user'] = $row['id'];
    header("Location: home.php");
} else {
    $_$outMessage = "Credenziali non valide, riprovare.";
}
```

Figura 11 Codice - Convalida dati e creazione della sessione

Insieme al file *login.php* è stato d'obbligo creare un file *logout.php* che d'altra parte si occupa di resettare la sessione, riportando l'utente alla pagina di login.

Completata la parte di accesso a Carthafind sono passato al core dell'intero progetto, *home.php.* Questo file contiene diversi punti salienti come:

```
$\footnote{\text{sextensions}} = \text{array("pdf", "doc", "docx");}

$\footnote{\text{sexplodedExt}} = \text{explode('.', $_FILES['doc_path']['name']);}

$\footnote{\text{sexplodedExt}} = \text{strtolower(end($\text{sexplodedExt}));}

$\footnote{\text{sdoc_path}} = \text{FILES['doc_path']['name'];}

$\footnote{\text{sextensions}} = \text{array("pdf", "doc", "docx");}

$\footnote{\text{sexplodedExt}} = \text{explode('.', $_FILES['doc_path']['name'];}

$\footnote{\text{sexplodedExt}} = \text{strtolower(end($\text{sexplodedExt}));}

$\footnote{\text{soc_path}} = \text{splodedExt}] = \text{splodedExt}

$\footnote{\text{sexplodedExt}} = \text{splodedExt}] = \text{splodedExt}

$\footnote{\text{splodedExt}} = \text{splodedExt}] = \text
```

Figura 12 Codice - Controllo estensione del file caricato dall'utente ed assegnazione

```
$\text{suploaddir = 'C:/xampp/htdocs/Carthafind/documentations/';}

$\text{suserfile_tmp = $_FILES['doc_path']['tmp_name'];}

$\text{suserfile_name = $_FILES['doc_path']['name'];}

$\text{move_uploaded_file($userfile_tmp, $uploaddir . $userfile_name);}
$\text{suserfile_name};
```

Figura 13 Codice - Spostamento del file con nome temporaneo in una cartella designata con nome permanente

Nadir Barlozzo 13 Versione: 21.05.2018



Pagina 14 di 21

```
148
149
                $keywords = explode(",", $keywords);
150
                foreach($keywords as $key) {
                    $key = strtolower($key);
152
                    $query = "SELECT 'id' FROM 'keyword' WHERE 'name'='".$key."'";
154
                    $result = $conn->query($query);
155
156
                    $kId = 0:
157
158
                    if($result->num_rows > 0) {
                        $kId = $result->fetch assoc();
159
                        $kId = $kId['id'];
160
161
162
                    else {
                        $query = "INSERT INTO `keyword` (`name`) VALUES ('".$key."')";
163
164
                        $conn->query($query);
165
                        $query = "SELECT 'id' FROM 'keyword' ORDER BY 'id' DESC LIMIT 1";
166
                        $result2 = $conn->query($query);
167
168
                        $kId = $result2->fetch_assoc();
169
                        $kId = $kId['id'];
170
171
172
                    $query = "INSERT INTO `contains` (`id_project`, `id_keyword`) VALUES (".$projectId.",".$kId.")";
173
                    $conn->query($query);
174
```

Figura 14 Codice - Controllo e referenza sull'esistenza delle parole chiave inserite o inserimento se nuove

```
$authors = explode(",", $authors);
177
178
               foreach($authors as $auth) {
179
                   $authA = explode(" ", $auth);
180
                   if($authA[0] != "") {
181
                       $query = "SELECT 'id' FROM 'user' WHERE 'name'='".$authA[0]."' AND 'surname'='".$authA[1]."";
182
                       $result2 = $conn->query($query);
183
                       $authI = $result2->fetch_assoc();
184
                       $authI = $authI['id'];
185
186
187
                       $query = "INSERT INTO `participates` ('id_project`, 'id_author') VALUES (".$projectId.",".$authI.")";
188
                       $conn->query($query);
189
190
```

Figura 15 Codice - Relazioni degli autori inseriti con il progetto

```
while($row = $result->fetch_assoc()) {
                                                    $query = "SELECT 'id keyword' FROM 'contains' WHERE 'id project'='".$row['id']."'";
306
                                                    $keyIds = $conn->query($query);
308
                                                    Smetadata = "";
                                                    while($key = $keyIds->fetch_assoc()) {
                                                        $query = "SELECT `name`
                                                                                FROM 'keyword' WHERE 'id'="".$key['id_keyword'].""";
                                                        $kname = $conn->query($query);
314
                                                        $kname = $kname->fetch_assoc();
315
                                                        $kname = $kname['name']
316
                                                        $metadata .= $kname.",";
```

Figura 16 Codice - Creazione dei metadata da inserire nel contenitore di ogni progetto per le ricerche future

ReactionGame

3.4 Javascript (Jquery)

Una volta definito il core server-side c'è ancora bisogna di un ultimo intermediario, il quale relaziona i dati provenienti dalle interrogazioni SQL effettuate tramite PHP con la grafica utente, questo viene compiuto da Javascript, più precisamente con una libreria chiamata Jquery.

Il tutto viene messo gestito all'interno di un singolo file carthafind.js che contiene i seguenti blocchi salienti:

```
var activeSystemClass = $('.list-group-item.active');
          $('#system-search').keyup(function() {
              var finder = this;
              var tableBody = $('.table-list-search tbody');
              var tableRowsClass = $('.table-list-search tbody tr');
              tableRowsClass.each(function(i, val) {
11
12
                  var rowText = $(val).text().toLowerCase();
var inputText = $(finder).val().toLowerCase();
13
14
15
                  $ ('.search-query-sf').remove();
if(inputText != '')
16
17
18
19
20
21
22
23
24
25
                      tableBody.prepend('<strong>Stai cercando: "'
                      + $(finder).val()
                      + '"</strong>');
                  rowText += $(val).data("keywords");
                  if(rowText.indexOf(inputText) == -1)
26
27
                      tableRowsClass.eg(i).hide();
28
                      tableRowsClass.eq(i).show();
31
32
33
34
              if(tableRowsClass.children(':visible').length == 0)
                  tableBody.append('Nessun risultato.');
```

Figura 17 Codice - CRT

```
$(".input-tag").bind("keydown", function(kp) {
72
              var tag = $(".input-tag")
73
              .val()
74
              .trim();
75
              $(".tags").removeClass("danger");
76
              if (tag.length > 0) {
78
                  backSpace = 0;
79
                  if (kp.keyCode == 32) {
                      $(".new-tag").before('' + tag + close + "");
80
81
                      (".tagarea").val((.tagarea").val().replace(/\s/g,'')+","+tag);
                      $(this).val("");
82
83
84
                else {
85
                  if (kp.keyCode == 8) {
86
                      $(".new-tag")
87
                      .prev()
88
                      .addClass("danger");
89
                      backSpace++;
90
                      if (backSpace == 2) {
                          $(".new-tag")
                          .prev()
92
93
                          .remove();
94
                          backSpace = 0;
95
96
98
          1);
```

Figura 18 Codice - CKI I



Pagina 16 di 21

```
100
     $(".tag-box").on("click", ".close", function() {
101
                $(this)
102
                .parent()
103
                .remove();
104
               var str = $('.tagarea').val();
105
               var res = str.replace($(this).parent().get(0).textContent + ",", " ");
106
               res = res.replace("," + $(this).parent().get(0).textContent, "");
107
               res = res.replace($(this).parent().get(0).textContent, "");
108
                res = res.replace("," + $(this).parent().get(0).textContent + ",", "");
109
                $(".tagarea").val(res);
110
           });
111
112
           $(".tag-box").click(function() {
113
               $(".input-tag").focus();
114
           });
```

Figura 19 Codice - CKI II

```
116
           $(".tag-box").on("dblclick", ".tags", function(cl) {
117
                var tags = $(this);
118
                var tag = tags.text().trim();
119
                $(".tags").removeClass("edit");
120
                tags.addClass("edit");
121
                tags.html('<input class="input-tag" value="' + tag + '" type="text">');
122
                $(".new-tag").hide();
123
                tags.find(".input-tag").focus();
124
125
                tag = $(this)
126
                .find(".input-tag")
127
                .val();
128
                $(".tags").dblclick(function() {
129
                    tags.html(tag + close);
130
                    $(".tags").removeClass("edit");
131
                    $ (".new-tag").show();
132
                1):
133
                tags.find(".input-tag").bind("keydown", function(edit) {
134
135
                    tag = $(this).val();
136
                    if (edit.keyCode == 70) {
      Ė
137
                        $ (".new-tag").show();
138
                        $(".input-tag").focus();
139
                        $(".tags").removeClass("edit");
140
      ₿
                        if (tag.length > 0) {
141
                            tags.html(tag + close);
142
                            } else {
143
                            tags.remove();
144
145
146
                });
147
            });
```

Figura 20 Codice - CKI III



ReactionGame

Pagina 17 di 21

4 Test

Test Case:	TC-01	Nome:	Controllo registrazione		
Riferimento:	REQ-01		g.cc.		
Descrizione:	Registrazione di	di un nuovo utente.			
Prerequisiti:	Avere accesso ad internet e al dominio samtinfo.ch/lpiarch.				
Procedura:	 Raggiungere la pagina http://samtinfo.ch/lpiarch, premere su <i>login</i> e a seguire su <i>registrati</i>. Compilare i campi richiesti. Premere <i>registrati</i>. 				
Risultati attesi:	Un contenitore appare sopra al campo nome mostrando il risultato della registrazione, di colore verde se avvenuta con successo o di colore rosso in caso contrario. Inoltre deve esserci un'email nella casella postale registrata che chiede la conferma dell'account.				

Test Case:	TC-02	Nome:	Controllo login	
Riferimento:	REQ-02			
Descrizione:	Login di un uten	te registrato.		
Prerequisiti:	Avere effettuato la registrazione con l'utente usato in questo test case.			
Procedura:	 Raggiungere la pagina http://samtinfo.ch/lpiarch, premere su <i>login</i>. Compilare i campi richiesti. 			
	3. Premere <i>login</i> .			
Risultati attesi:	Un contenitore di colore rosso appare sopra al campo email se la convalida ha dato un errore, in caso contrario viene effettuato l'accesso e si viene reindirizzati al file home.php.			

Test Case:	TC-03	Nome:	Controllo logout		
Riferimento:	REQ-02		Some region		
Descrizione:	Logout di un ute	gout di un utente registrato.			
Prerequisiti:	Avere effettuato la registrazione con l'utente usato in questo test case ed avere effettuato il login.				
Procedura:	 Premere il bottone in alto a destra con il vostro <i>username</i>. Premere su <i>esci</i>. 				
Risultati attesi:	Si viene reindiriz	lirizzati al file <i>login.php</i>			



ReactionGame

Pagina 18 di 21

Test Case:	TC-04	Nome:	Controllo ricerca	
Riferimento:	REQ-03		33	
Descrizione:	Ricerca tra progetti.			
Prerequisiti:	Avere effettuato il login ed aver un utente con privilegi di registered.			
Procedura:	Scrivere una parola nel campo di ricerca.			
Risultati attesi:		orogetti contenenti un campo o una parte del metadata uguale alla parola a devono essere evidenziati, tutti gli altri nascosti.		

Test Case:	TC-05	Nome:	Controllo inserimento progetto	
Riferimento:	REQ-04		eenmene meenmene progene	
Descrizione:	Inserimento di un nuovo progetto.			
Prerequisiti:	Avere effettuato il login ed aver un utente con privilegi di administrator.			
	Scrollare fino a raggiungere il form di inserimento per nuovi progetti.			
Procedura:	Compilare i campi richiesti (Commenti è facoltativo), per le parole chiave guardare l'allegato B.			
Risultati attesi:	Il progetto viene aggiunto prontamente alla lista dei progetti esistenti e si viene riportarti all'inizio del form per l'inserimento di un altro record.			

4.1 Mancanze/limitazioni conosciute

Al momento Carthafind permette solamente l'inserimento di nuovi progetti, senza possibilità di modifica o eliminazione di quelli esistenti.

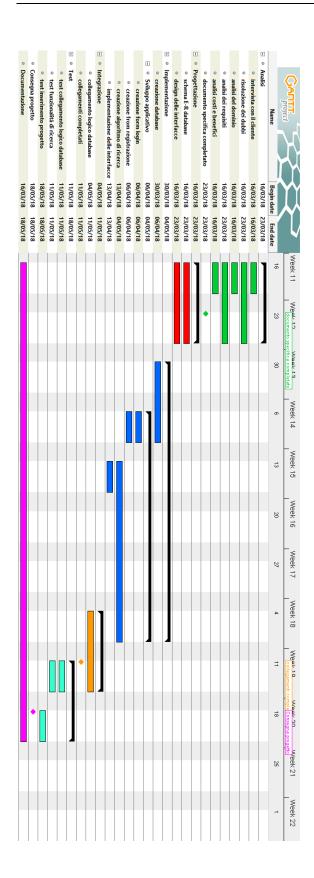
Inoltre i controlli effettuati sui campi del form non sono del tutto corretti e necessitano di un ulteriore revisione e svariati test.



ReactionGame

Pagina 19 di 21

5 Consuntivo





ReactionGame

Pagina 20 di 21

6 Conclusioni

Il prodotto è senz'altro unico nel suo genere ed è stato interessante lavorarci sopra per la durata del tempo che mi è stato concesso, credo che al suo stato corrente sia già utilizzabile senza grossi problemi nell'esperienza dell'utente finale.

6.1 Sviluppi futuri

Maggiori e migliori controlli sui dati inseriti dall'utente, per esempio asserire se le parole chiave scelte dall'utente sono effettivamente esistenti tramite un dizionario o un programma improntato al self-learning. Effettuare una ricerca anche all'interno del file di documentazione passato dall'utente quando viene usato il CRT.

6.2 Considerazioni personali

Ho trovato personalmente il progetto divertente e anche interessante per quanto riguarda la gestione in singolo, infatti avendo lavorato da solo ho potuto decidere tutte le tappe e le durate concordate con le mie preferenze personali.

7 Bibliografia

7.1 Sitografia

https://getbootstrap.com/docs/4.1/getting-started/introduction/, Documentazione Bootstrap.

http://api.jquery.com/, API Jquery

https://goo.gl/cBCHbs, Lettura rilasciata da Microsoft stessa sulla pratica di salvare BLOB nei database.

8 Allegati

Allegato A: I3_Diari_Carthafind.pdf

Allegato B: CKI.pdf



ReactionGame

Pagina 21 di 21

9 Glossario

Parola	Significato		
CKI	Carthafind Keyword Input		
CRT	Carthafind Research Tool		
API	Application Programming Interface		
BLOB	Binary Large OBject		
Bootstrap	Un framework front-end gratuito per uno sviluppo web più facile e veloce		
Framework	Software che fornisce funzionalità generiche le quali possono essere modificate in modo selettivo da un ulteriore codice scritto dall'utente		
PHP	Personal Home Page		