

ERASMUS 2K23

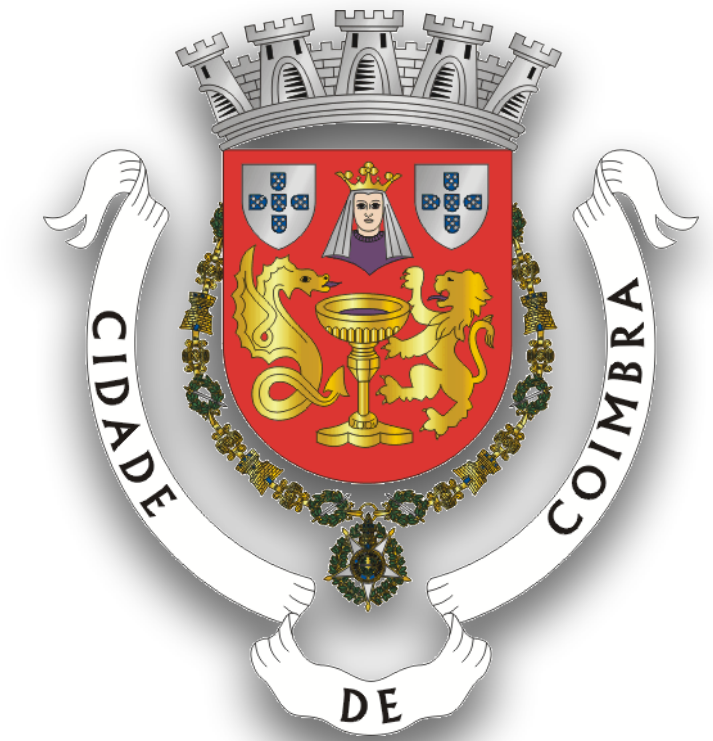
Flusso Coimbra



Una presentazione di Demelas Davide

Il Luogo

- La città dove si è svolto il nostro stage all'estero è **Coimbra**.
- E' un comune di circa 143 mila abitanti situato nel centro del Portogallo ed è il capoluogo del **Distretto di Coimbra**
- E' famosa per la sua **Università** (una delle prime in Europa), che è inoltre il maggior concentrazione universitario di tutto il paese (20.000 studenti fuori sede).



Il Contesto

- Per questa esperienza siamo partiti in **10 compagni** di classe
- La nostra permanenza a Coimbra è durata **1 mese** (dal 20/02 al 20/03)
- Abbiamo lavorato dal **22/02** al **17/03**, dal lunedì al venerdì
- Ci hanno anche accompagnato figure, le quali:
 1. **Group leader**: Pasquale Bucci
 2. **Tutor Esperienza**: Isabel
 3. **Tutor aziendale**: João Claro



Noi e pasquale al check-in

L'Azienda

- L'azienda presso la quale ho lavorato in questa permanenza è **TYBO (Track Your Business Online)**
- Si tratta di un'azienda che si occupa principalmente di **contabilità**
- Il suo **obbiettivo** principale è quello di cercare di **unire il Business alla Contabilità**, che di norma viaggiano su linee separate
- I dipendenti erano circa una **trentina** e svolgevamo tutti il lavoro su delle **scrivanie** al computer



La mia postazione di lavoro

Il Progetto

- Il **progetto**, che la società ci ha assegnato, è stato quello di **creare un sito web** dove data una lista di account, cliccando su di uno è possibile risalire a una tabella (o a un avviso nel caso non ci siano dati), tutto questo sfruttando la **API** della società
- Nella **homepage** verranno visualizzati una serie di **account** numerati in base all'ambito (1001...; 2001...; 3001...; 4001...; 5001...; 6001...), ognuno linkerà alla pagina rispettiva
- Una volta aperto uno degli account verrà visualizzata una **tabella** contenente i dati di quell'account. La tabella viene creata tramite un codice che ricava le informazioni dalla **API** e le inserisce in una **tabella** in codice **HTML**, sfruttando un **ciclo iterativo**
- Nella **homepage** inoltre affianco a ogni **sigla degli account** è visibile il **totale** per quello specifico account e il **totale nell'ultimo mese** (in modo da capire subito se all'interno sono presenti dati o meno)
- Nel caso **non ci siano dati da visualizzare** semplicemente viene visualizzata la **scritta "¡Nada para exhibir para esta conta!"**

Il Progetto: foto illustrative (homepage)

Terreno		
1001	€ 0.00	€ 0.00
1002	€ 0.00	€ 0.00
Mao De Obra Direta		
2001	€ 0.00	€ 0.00
Materiasra		
3001	€ 961.68	€ 0.00
3002	€ 2510.65	€ 0.00
3003	€ 2126.28	€ 0.00
3004	€ 9401.26	€ 0.00
3005	€ 1060.46	€ 0.00
3006	€ 0.00	€ 0.00
3007	€ 0.00	€ 0.00
3008	€ 0.00	€ 0.00
3009	€ 0.00	€ 0.00
3010	€ 0.00	€ 0.00
3011	€ 6124.79	€ 0.00
3012	€ 77057.09	€ 0.00
3013	€ 3803.58	€ 0.00
3014	€ 0.00	€ 0.00
3015	€ 0.00	€ 0.00
3016	€ 0.00	€ 0.00
3017	€ 0.00	€ 0.00
3018	€ 4423.03	€ 0.00
3019	€ 19490.88	€ 0.00
3020	€ 0.00	€ 0.00
3021	€ 13328.55	€ 0.00
3022	€ 143.71	€ 0.00
3023	€ 88.50	€ 0.00
3024	€ 32063.72	€ 0.00
3025	€ 0.00	€ 0.00
3026	€ 0.00	€ 0.00
3027	€ 0.00	€ 0.00
3999	€ 2828.32	€ 0.00
Equipamento		
4001	€ 6056.11	€ 0.00
4002	€ 1528.56	€ 0.00
4003	€ 67.16	€ 0.00
4004	€ 0.00	€ 0.00
Subempreitadas		
5001	€ 3030.00	€ 0.00
5002	€ 64646.00	€ 0.00
5003	€ 0.00	€ 0.00
5004	€ 82191.98	€ 0.00
5005	€ 0.00	€ 0.00
5006	€ 63271.44	€ 0.00
5007	€ 68498.00	€ 0.00

5008	€ 2354.50	€ 0.00
5009	€ 0.00	€ 0.00
5010	€ 0.00	€ 0.00
5011	€ 0.00	€ 0.00
5012	€ 20037.00	€ 0.00
5013	€ 0.00	€ 0.00
5014	€ 0.00	€ 0.00
5015	€ 2580.00	€ 0.00
5016	€ 3360.00	€ 0.00
5017	€ 0.00	€ 0.00
5018	€ 9155.23	€ 0.00
5019	€ 0.00	€ 0.00
5020	€ 0.00	€ 0.00
5021	€ 0.00	€ 0.00
5022	€ 23096.20	€ 0.00
5023	€ 16227.00	€ 0.00
5024	€ 0.00	€ 0.00
5025	€ 0.00	€ 0.00
5027	€ 5000.00	€ 0.00
5029	€ 16171.00	€ 0.00
Prestação Serviços		
6001	€ 0.00	€ 0.00
6002	€ 17927.50	€ 0.00
6003	€ 900.00	€ 0.00
6004	€ 0.00	€ 0.00
6005	€ 1459.63	€ 0.00
6006	€ 75.00	€ 0.00
6007	€ 0.00	€ 0.00
6008	€ 0.00	€ 0.00
6009	€ 0.00	€ 0.00
6010	€ 0.00	€ 0.00
6011	€ 0.00	€ 0.00
6012	€ 0.00	€ 0.00
6013	€ 1882.66	€ 0.00
6014	€ 0.00	€ 0.00
6015	€ 92.81	€ 0.00
6016	€ 1800.00	€ 0.00
6017	€ 4360.00	€ 0.00

Il Progetto: foto illustrative (tabella)

Conta 3011

dataDoc	dataVencimento	nDocEsterno	descrizione	extPocCabID	periodo	diario	nDocInterno	valorDebito (€)	valorCredito (€)	porPagar	saldo
2021-06-18	2021-06-19	FT FVV1/202110675		372AEABD-1810-453A-8D58-9D9AFEDD3258	2021061	10	10.1.384	1278.4	0	0	0
2021-07-12	2021-07-13	FR 3526009189		4588FEB9-2612-4990-B97C-8897A217FBD5	2021071	10	10.1.399	111.2	0	0	0
2021-12-09	2021-12-10	FT 2021A187/4990		59D2419C-D6B1-4CD3-9896-9499116723E9	2021121	10	10.1.524	909.74	0	0	0
2022-03-18	2022-03-19	FT 202200534/007890		3B5F49BF-605C-44A8-9CA2-EE9F364DB585	2022031	10	10.1.613	47.15	0	0	0
2022-06-30	2022-06-30	FRI 2022/645		97A36A5F-9E63-4D12-A76F-7276DE76499C	2022061	56	14	349.57	0	0	0
2022-06-30	2022-06-30	FT V1/202209268		3FEF5F09-F7FF-4560-BF72-DE31C63DDFFC	2022061	56	17	499.33	0	0	0
2022-07-31	2022-07-31	FT R101/00007389		3A86E133-DE0C-4631-8602-46E50F4936E1	2022071	56	29	5.61	0	0	0
2022-07-31	2022-07-31	FT 2022A187/3109	PALEGESSOS, INDÚSTRIA E COMÉRCIO DE PALETES E GESS	5B64F866-5F89-4DD7-9470-51FEB908BAF	2022071	56	39	1534.52	0	0	0
2022-08-31	2022-08-31	FT 2022A139/2762	PALEGESSOS, INDÚSTRIA E COMÉRCIO DE PALETES E GESS	0D94EEC0-8CC5-4DE7-BFC6-04D9A16D65FF	2022081	56	7	496.08	0	0	0
2022-08-31	2022-08-31	FT 2022A187/3331	PALEGESSOS, INDÚSTRIA E COMÉRCIO DE PALETES E GESS	38B3377C-8D00-4F39-B669-591099EC7EB2	2022081	56	10	893.19	0	0	0
								6124.79 €	0.00 €		

[voltar a página inicial](#)

Tasto per tornare alla homepage

Totale calcolati

Il Progetto: foto illustrative (codice)

Codice per impostare i numeri degli account

```
class PagesController < ApplicationController
  def home
    # client = FirstStep::Client.new

    # (3001..3027) is a ruby range: https://ruby-doc.org/core-3.1.0/Range.html
    # .to_a converts the range to an array
    # array << element adds element to the array
    @accounts =
    {
      1000 => [1001, 1002],
      2000 => [2001],
      3000 => (3001..3027).to_a << 3999,
      4000 => (4001..4004).to_a,
      5000 => (5001..5025).to_a << 5027 << 5029,
      6000 => (6001..6017).to_a
    }

    # let's make a request for each account
    # @ext will gather the information from all accounts in a hash
    # @ext = { 3001 => ..., 3002 => ..., 3003 => ...}
    # @ext = {}
    # accounts.each do |account|
    #   response = client.extratatos(account_number: account, from: "2019-01-01", to: "2023-02-28")
    #   @ext[account_number] = response["list"]
    # end

    # you can put it in an array instead
    # @ext = []
    # accounts.each do |account|
    #   response = client.extratatos(account_number: account, from: "2019-01-01", to: "2023-02-28")
    #   @ext << response["list"]
    # end
  end
end
```


Il Progetto: foto illustrative (codice)

Codice che permette di visualizzare sul sito le tabelle

```
class AccountsController < ApplicationController

  def show

    begin
      @account_id = params[:item]
      @project = params[:project]
      @family = params[:family]
      @last_number = params[:last_number]
      from = params[:from] || '2019-01-01' #https://stackoverflow.com/a/1554380
      to = "2023-03-14"

      client = FirstStep::Client.new
      response = client.extratras(project: @project, family: @family, item: @account_id, last_number: @last_number, from: from, to: to)
      @ext = response&.[('list')] # https://stackoverflow.com/a/36813420
      # same as:
      # @ext = response['list'] if response.present?
      #https://ruby-doc.org/core-2.4.0/Enumerable.html#method-i-sum
      @total_credit = @ext.sum { |row| row['valorCredito'].to_f }
      @total_debit = (@ext.sum { |row| row['valorDebito'].to_f }) - @total_credit } // codice che esegue la somma dei
      // valori del credito e del debito
    rescue NoMethodError => ex
    end

  end

end
```

Il Progetto: foto illustrative (codice)

```
<div class="homepage">

<table id="home-table" >
  <tr id="row-title"> <td colspan="3"> Terreno <td> </tr>

  <% @accounts[1000].each do |account| %>
    <tr id="row">
      <td id="account-number"> <%= link_to account, account_path("2002","01",account,"6") %> </td>
      <td id="sum"> € <%= FirstStep::SumDebit.new.debit("2002","01",account,"6") %> </td>
      <td id="sum"> € <%= FirstStep::SumDebit.new.monthly("2002","01",account,"6") %> </td>
    </tr>
  <% end %>
</table>

<table id="home-table" >
  <tr id="row-title"> <td colspan="3"> Mao De Obra Direta <td> <tr>
  <% @accounts[2000].each do |account| %>
    <tr id="row">
      <td id="account-number"> <%= link_to account, account_path("2002","02",account,"6") %> </td>
      <td id="sum"> € <%= FirstStep::SumDebit.new.debit("2002","02",account,"6") %> </td>
      <td id="sum"> € <%= FirstStep::SumDebit.new.monthly("2002","02",account,"6") %> </td>
    </tr>
  <% end %>
</table>

<table id="home-table" >
  <tr id="row-title"> <td colspan="3"> Materiasra <td> <tr>
  <% @accounts[3000].each do |account| %>
    <tr id="row">
      <td id="account-number"> <%= link_to account, account_path("2002","03",account,"3") %> </td>
      <td id="sum"> € <%= FirstStep::SumDebit.new.debit("2002","03",account,"3") %> </td>
      <td id="sum"> € <%= FirstStep::SumDebit.new.monthly("2002","03",account,"3") %> </td>
    </tr>
  <% end %>
</table>
```

```
<table id="home-table" >
  <tr id="row-title"> <td colspan="3"> Subempreitadas <td> <tr>
  <% @accounts[5000].each do |account| %>
    <tr id="row">
      <td id="account-number"> <%= link_to account, account_path("2002","05",account,"6") %> </td>
      <td id="sum"> € <%= FirstStep::SumDebit.new.debit("2002","05",account,"6") %> </td>
      <td id="sum"> € <%= FirstStep::SumDebit.new.monthly("2002","05",account,"6") %> </td>
    </tr>
  <% end %>
</table>

<table id="home-table" >
  <tr id="row-title"> <td colspan="3"> Prestação Serviços <td> <tr>
  <% @accounts[6000].each do |account| %>
    <tr id="row">
      <td id="account-number"> <%= link_to account, account_path("2002","06",account,"6") %> </td>
      <td id="sum"> € <%= FirstStep::SumDebit.new.debit("2002","06",account,"6") %> </td>
      <td id="sum"> € <%= FirstStep::SumDebit.new.monthly("2002","06",account,"6") %> </td>
    </tr>
  <% end %>
</table>
</div>
```

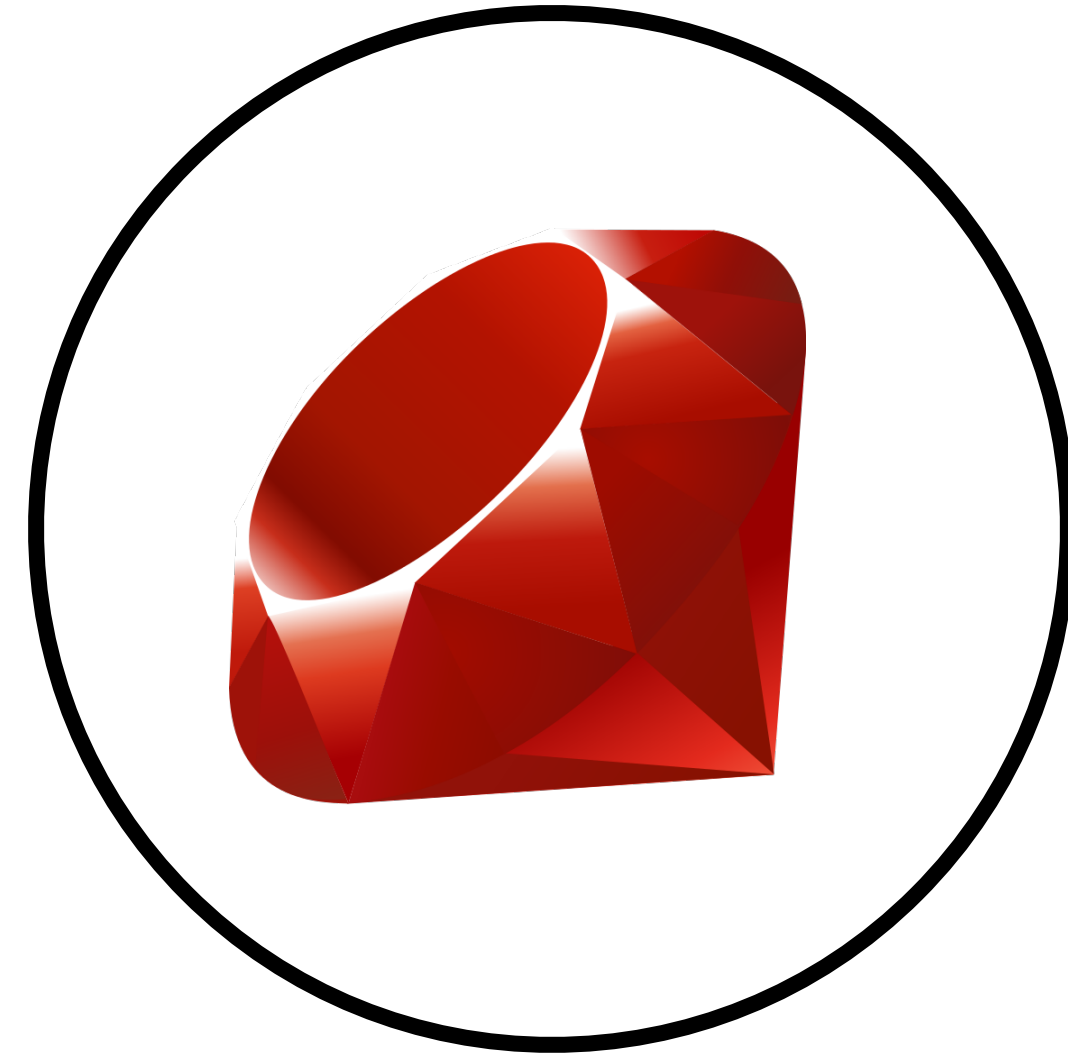
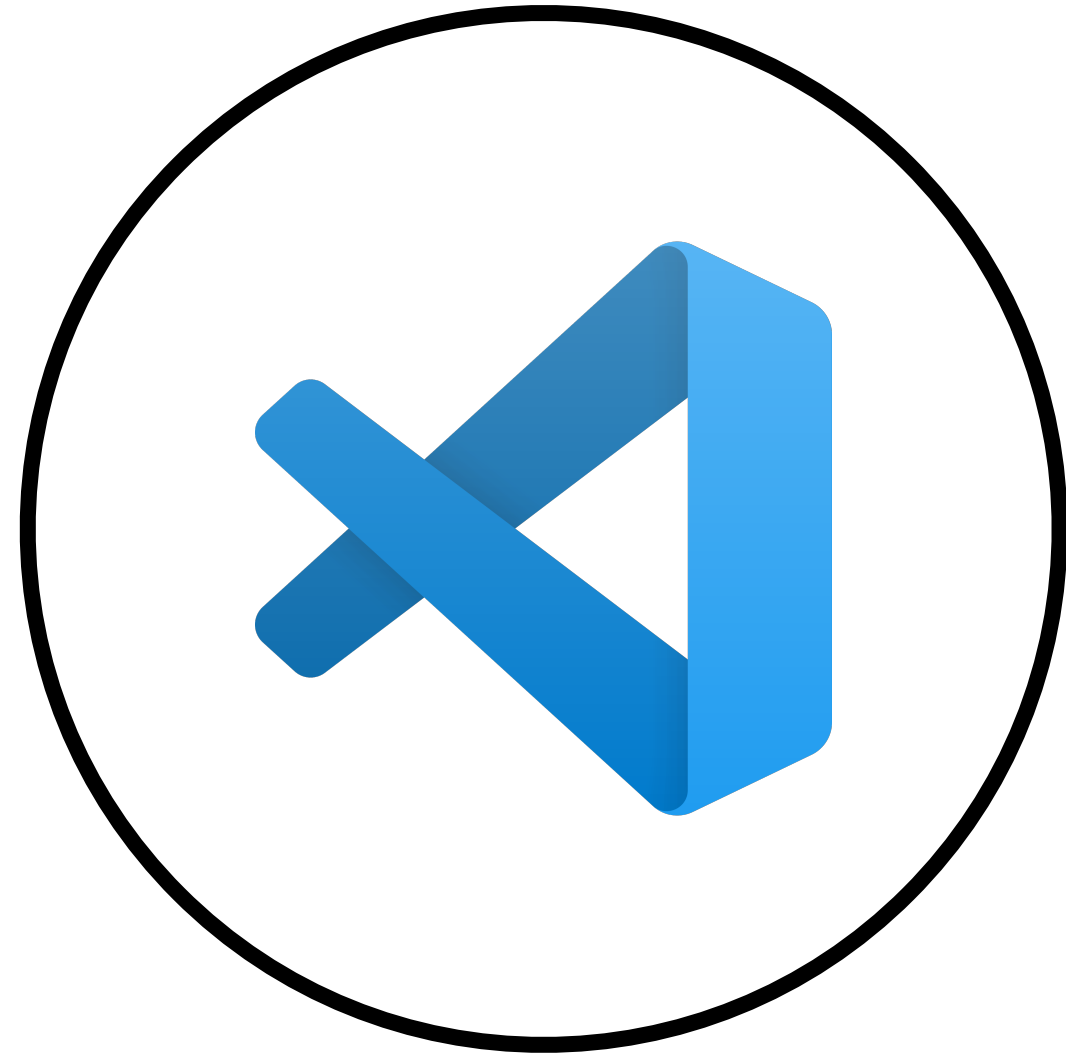
Codice che permette di stampare la serie di account nella homepage

CALENDARIO DELLE MANSIONI

LUN 20/02	MAR 21/02	MER 22/02	GIO 23/02	VEN 24/02
ARRIVO A COIMBRA	CARNEVALE	GIORNO DI RIPOSO	Presentazione al capo e al tutor aziendale strutturato come un vero e proprio colloquio di lavoro	Introduzione alla società e al progetto da sviluppare. Studio di BlockChain , Smart Contract e Ruby on Rails .
LUN 27/02	MAR 28/02	MER 01/03	GIO 02/03	VEN 03/03
Visione dei tutorial ricevuti dal tutor aziendale Creazione di una prima bozza di progetto per imparare a usare il nuovo linguaggio	Miglioramento dal punto di vista stilistico della bozza, tramite linguaggio CSS Conclusione del primo corso (Ruby) e inizio visione del secondo (Ruby on Rails)	Revisione della bozza per continuare ad imparare nuove funzionalità riguardanti Ruby on Rails . Studio di reCAPTCHA e Autenticazione a due fattori . Introduzione al sito della società e all'uso delle API Controllo del tutor sui progressi fatti	Inizio ufficiale del progetto da portare avanti in questo Erasmus Creazione della homepage del progetto cercando di comprendere meglio l' uso delle API Riunione con il tutor per chiarimenti sul linguaggio e su funzionalità nuove per noi (API private, uso di GitLab, ecc...)	Continuo studio sulle API tramite test che sfruttando quelle di tipo " open source " Riunione col tutor per capire come sfruttare le API private sul nostro sito Miglioramento della homepage (dal punto di vista delle funzionalità) dopo i chiarimenti

LUN 06/03	MAR 07/03	MER 08/03	GIO 09/03	VEN 10/03
<p>Recupero informazioni dall'API fornito dalla società</p> <p>Scrittura codice per stampare le informazioni dall'API direttamente su tabelle nel sito</p> <p>Scrittura del codice per il calcolo del credito e del debito</p>	<p>Risoluzione di problemi che impedivano il proseguimento del progetto</p> <p>Miglioramento della tabella cercando di renderla dinamica</p>	<p>Cambio di rotta del progetto, siamo passati a un sito dove dalla homepage si possono raggiungere le varie tabelle</p> <p>Aiuto da remoto da parte del tutor per riuscire a rendere possibile questa idea</p>	<p>Automatizzazione della creazione delle tabelle</p> <p>Migliorie stilistiche come: diminuzione delle colonne (solo quelle necessarie), tasto che riporta direttamente alla homepage, data impostata secondo lo standard "gg/mm/aaaa"</p>	<p>Correzione del codice da parte del tutor che ha rimosso le ripetizioni</p> <p>Miglioramento estetico del sito</p> <p>Gestione caso in cui l'account non abbia nessun dato da mostrare</p>
LUN 13/03	MAR 14/03	MER 15/03	GIO 16/03	VEN 17/03
<p>Estensione del sito a tutti gli account presenti</p> <p>Tentativo di aggiungere la funzionalità che permette di vedere accanto a ogni account il loro totale</p>	<p>Siamo riusciti ad aggiungere la funzionalità che stavamo tentando di inserire il giorno precedente e abbiamo continuato a migliorare il sito dal punto di vista estetico</p>	<p>Richiesta del capo di aggiungere una nuova funzionalità al sito, con la quale ogni account avrebbe avuto un link per scaricare dei file pdf</p> <p>Miglioramento estetico del sito</p>	<p>Termine dell'esperienza lavorativa</p> <p>Alla fine abbiamo rinunciato ad aggiungere la funzionalità per la quale si potevano scaricare i file PDF, ma abbiamo aggiunto una terza colonna nella home dove si possono visualizzare i totali dell'ultimo mese</p>	<p>GIORNO DI LAVORO SALTATO PER SCIOPERO DEI MEZZI PUBBLICI</p>

Gli Strumenti digitali





1. Visual Studio Code

- Visual Studio Code è un editor di codice sorgente di Microsoft.
- E' molto famoso nel mondo del coding perché permette di scrivere in quasi tutti i linguaggi presenti
- E' sfruttato maggiormente per programmare in HTML, Javascript, la famiglia C (C, C++, C#), ecc...
- Inoltre permette di installare estensioni prodotte anche da utenti, che possono: cambiare aspetto del ambiente, aggiungere funzionalità, permettere di programmare in un certo linguaggio.
- E' dotato di una Shell a riga di comando che funziona esattamente come quella presente in tutti i nostri dispositivi



2. Postman

- **PostMan** è una piattaforma **API** che permette di **progettare, costruire, testare e iterare** le proprie **API** o quelle di altri sviluppatori.
- Semplicemente inserendo il **link della API** all'interno del programma è possibile fare tutti i tipi di richiesta
- Nel nostro caso abbiamo eseguito delle **GET Request**, in modo da ricavare dei dati
- Una volta inviata la richiesta, se tutto è andato a buon fine, è possibile visualizzare un **file JSON**, che rappresenta **tutte le informazioni** in maniera gerarchica



2. Postman

The screenshot displays the Postman web interface. At the top, there's a navigation bar with 'Home', 'Workspaces', 'API Network', and 'Explore'. A search bar and utility buttons like 'Invite', 'Upgrade', and 'No Environment' are also present. The main workspace is titled 'My Workspace' and contains a sidebar with navigation options: Collections, APIs, Environments, Mock Servers, Monitors, Flows, and History. The 'Collections' sidebar shows a tree view for 'My first collection' with sub-folders and individual requests. A 'Create a collection for your requests' prompt is visible, with a 'Create Collection' button and a 'Use a Template' link. The main area shows a REST client for a GET request to 'https://www.thecocktaildb.com/api/json/v1/1/search.php?s=margarita'. The 'Params' tab is active, displaying a table of query parameters:

Key	Value	Description
<input checked="" type="checkbox"/> s	margarita	
Key	Value	Description

Below the table, there's a 'Response' section which is currently empty. A cartoon astronaut holding a rocket is shown with the text 'Click Send to get a response'. The bottom status bar includes 'Online', 'Find and Replace', 'Console', 'Cookies', 'Capture requests', 'Runner', 'Trash', and a help icon.



3. Ruby On Rails

- **Ruby on Rails** (o **Rails**) è un **framework open source** che sfrutta il linguaggio di programmazione **Ruby**. E' ispirato al pattern di programmazione **MVC** (**Model View Control**)
- E' stato inventato da **David Heinemeier Hansson** con lo scopo di **rendere più semplice la programmazione**, riuscendo a sviluppare applicazioni con **meno codice possibile**
- I principi dietro questo framework sono "**dont repeat yourself**" e "**convention over configuration**"
- Grazie a questo framework è inoltre possibile **dinamicizzare la scrittura di file HTML**
- Nel nostro caso è stato **utile** soprattutto **per la creazione di tabelle**, infatti con Rails è bastato creare un **unico file, contenente il codice della tabella**, che venisse poi richiamato da altri. Inoltre permette l'implementazione di concetti della programmazione in dei file HTML



4. Git Lab

- GitLab è una **piattaforma web opensource** che permette la **condivisione del codice** da parte degli utenti
- E' stata molto utile durante il nostro stage dato che il nostro tutor per motivi di lavoro non poteva essere sempre presente ad aiutarci, di conseguenza abbiamo creato una "**repository**" dove **pubblicare il codice aggiornato** in modo che potesse correggercelo e rinviarcelo con la spiegazione di quello che aveva modificato
- La cosa molto utile di **GitLab** è il fatto che semplicemente scrivendo nella **Shell a riga di comando** presente in Visual Studio Code il comando **push** è possibile **caricare automaticamente** il codice aggiornato
- Inoltre sul sito è possibile **vedere evidenziati** in verde le **parti del codice che sono cambiate** rispetto alla versione precedente o in rosso **quelle che sono state eliminate definitivamente**

Competenze acquisite

- Grazie a questa esperienza sono riuscito ad **acquisire nuove competenze** e a **migliorare quelle che già possedevo**.
- Il fatto di dovermi interfacciare con persone che non conoscevo affatto, e che oltretutto parlavano un'altra lingua, ritengo che abbia migliorato in me l'aspetto della **comunicazione**, migliorando ancora di più il mio **modo di esprimermi**, riuscendo a essere più disinvolto
- Avere un **progetto** da portare avanti utilizzando **strumenti nuovi** e **non** potendo sempre **contare** sulla **presenza fissa del tutor** mi ha permesso di migliorare la mia **indipendenza produttiva**, di fatto molte delle cose che ho imparato le ho cercate e assimilate per conto mio, e quasi sempre gli errori del codice gli ho corretti sempre io.
- Inoltre vivere **un mese senza i genitori** ha sviluppato in me una sorta di **miglioramento nell'autogestione**, riuscendo a fare una giusta spesa, a organizzarmi il tempo che avevo in base a le faccende domestiche, allo svago e al lavoro e anche a non cercare sempre una figura di supporto in caso di bisogno, ma arrangiandomi (salvo casi particolari).

Pro e Contro

PRO

- Uscita dalla **confort zone**
- **Indipendenza**
- Vivere **da soli**
- Vedere come funziona **il mondo del lavoro**
- Interfacciarsi con **nuove culture**
- Provare **nuove cose**

CONTRO

- **Primo approccio con una lingua nuova** non totalmente positivo
- Per il resto non mi vengono in mente