

# An Hand For Humanity

## FOR A BETTER WORLD

### Sommario

Introduzione.....	2
Requisiti da soddisfare.....	2
Concetti principali .....	3
Schema Scheletro.....	3
Schema Intermedio.....	4
Schema Finale.....	5
Schema Logico .....	7
Codice SQL .....	8
DBMS alternativo.....	9
Sito	
Web.....	10
.....	10

Progetto Realizzato per l'esame di "Basi di Dati" da:

- 1) Simone De Angelis – 0044837
- 2) Francesco Pennacchia – 0044364 (Nuova Matricola univaq: 247848)
- 3) Riccardo Scarpellino – 0043945

## Introduzione

L'organizzazione umanitaria "An Hand for Humanity" si occupa di portare solidarietà a livello internazionale, ovvero svolgendo compiti come: sfamare i senzatetto, allestire campi medici, prestare primo soccorso e molto altro ancora.

L'organizzazione è così gestita:

Non è presente una sede centrale, bensì sedi distribuite in tutto il mondo.

Ogni sede ha come riferimento la "cassa comune".

La cassa comune è un conto bancario unico per ogni sede dove, ognuna di esse può attingere per pagare gli stipendi del proprio personale oppure effettuare spese di altro tipo.

L'organizzazione ha anche numerosi partner e volontari con cui collabora durante lo svolgimento delle mansioni e quando questi non dovessero bastare, vengono incaricate ditte esterne.

## Requisiti da soddisfare

L'esigenza di creare un database nasce dal bisogno di gestire i seguenti punti:

- 1) il personale operante presso ogni sede e relative trasferte; 2)  
le varie mansioni svolte, il luogo ed i collaboratori.
- 3) il flusso monetario, rappresentato ad esempio da stipendi, donazioni effettuate e/o ricevute, pagamento ditte esterne, spese di vario tipo e soprattutto l'autore di queste transazioni.

Il Database potrà essere utilizzato da chiunque, tramite un apposito sito web, oppure dagli amministratori tramite un pannello di controllo come ad es "phpmyadmin".

# PROGETTAZIONE DEL DATABASE

---

*Nota: per i seguenti diagrammi ER abbiamo scelto di utilizzare le cardinalità come con la convenzione del workbench*

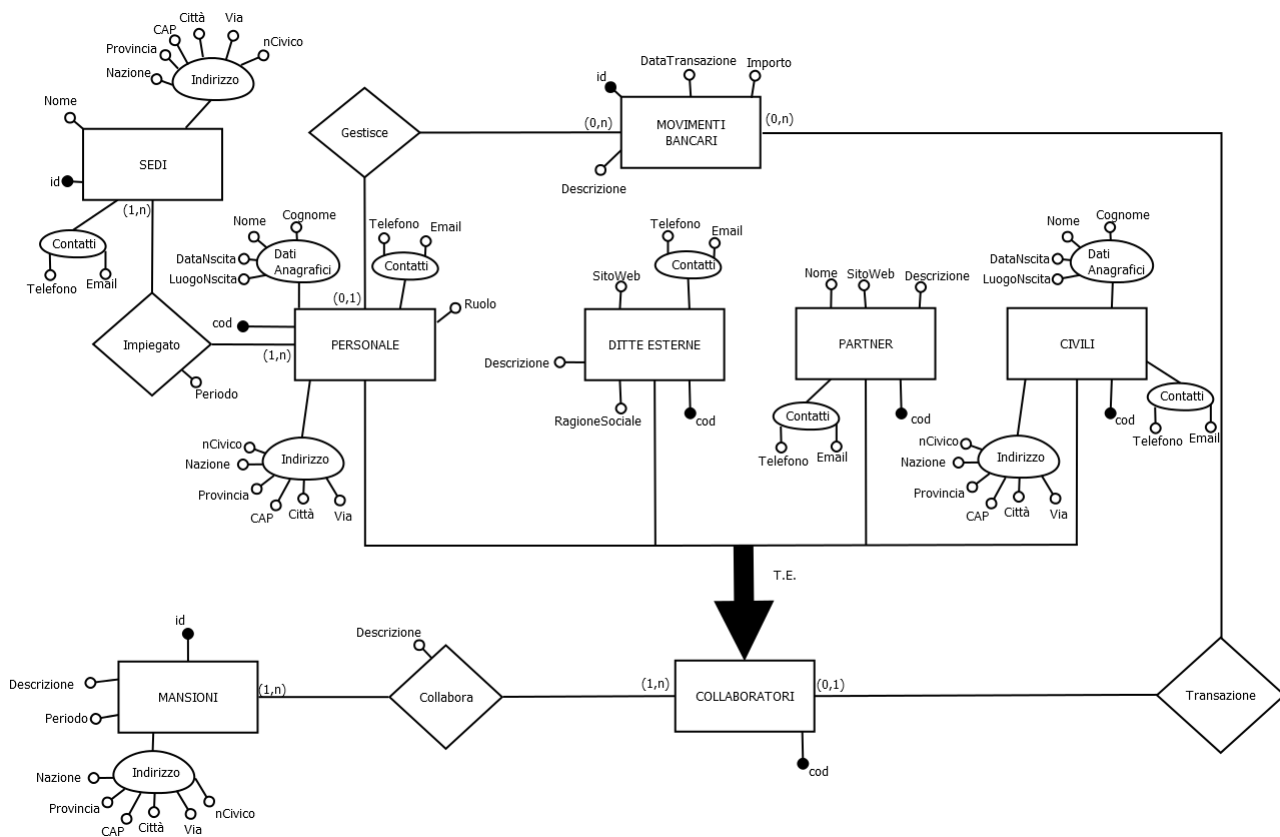
## Concetti principali

Dopo una prima analisi abbiamo dedotto che i concetti principali, al fine di realizzare un database inerente alla nostra richiesta, sono i seguenti:

- Personale
- Sedi
- Mansioni
- Collaboratori
- Movimenti bancari

L'intero database ruoterà intorno a questi concetti.

## Schema Scheletro

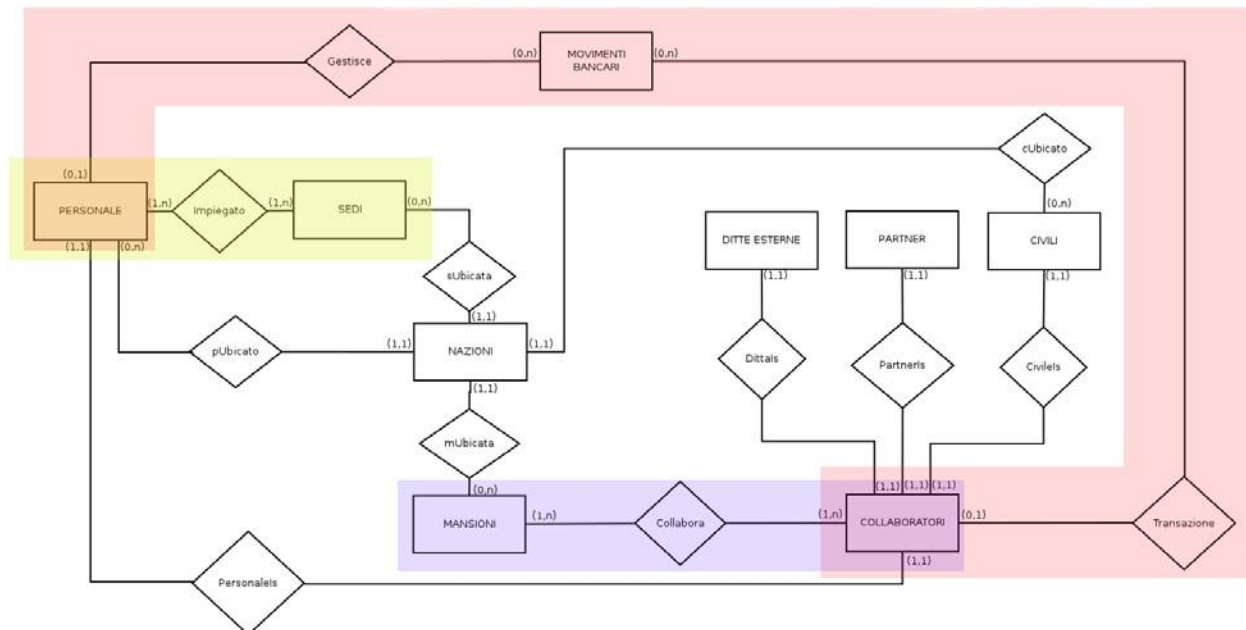


In questo primo schema possiamo vedere una prima idea di quello che poi sarà il database finale.

Abbiamo scelto le entità: “personale”, “ditte esterne”, “partner” e “civili” come generalizzazioni dell’entità “collaboratori”. Si tratta di una generalizzazione totale ed esclusiva.

Usando questa strategia, in futuro, è possibile inserire nuovi tipi di collaboratori senza dover modificare il resto dello schema.

## Schema Intermedio

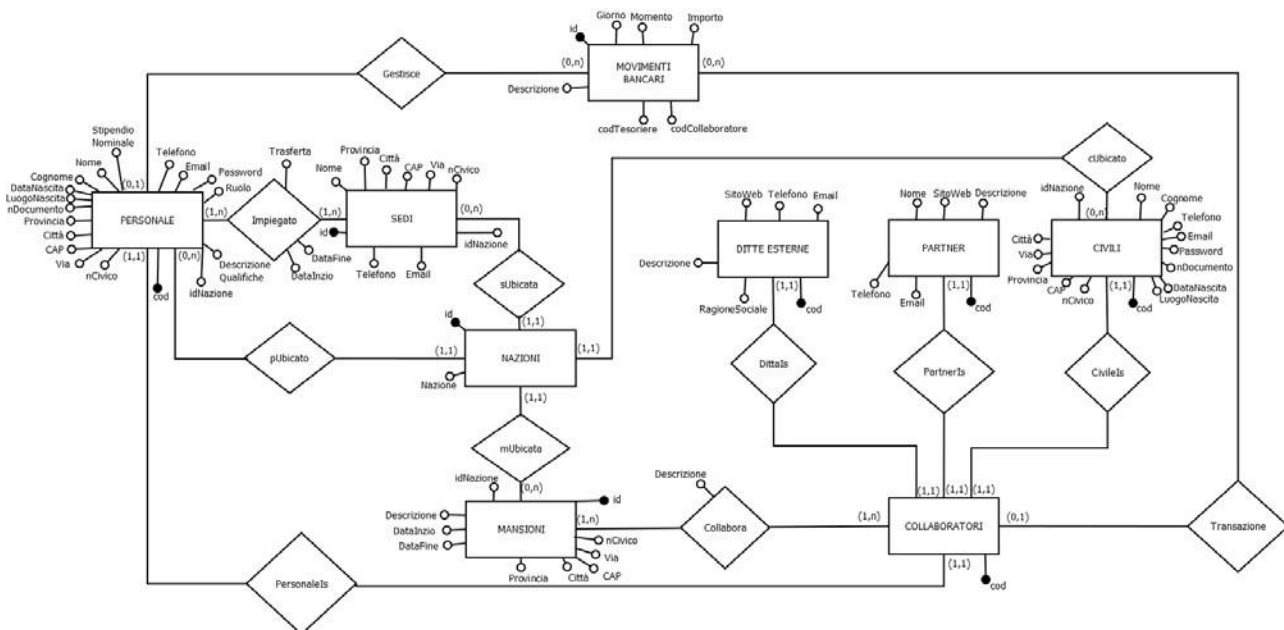


Per semplificare il lavoro, in questa fase, abbiamo adottato una strategia di progettazione “bottom-up”. Abbiamo quindi diviso lo schema in tre parti, vi abbiamo lavorato singolarmente ed infine le abbiamo ricollegate.

- 1) La prima parte (in giallo) riguarda la gestione del personale presso le sedi.  
Da qui possiamo conoscere i dati dei dipendenti, le loro trasferte e tutte le informazioni sulla sede.
- 2) La seconda parte (in rosso) riguarda la gestione del flusso di denaro.  
Da qui è possibile pagare gli stipendi ai dipendenti, ricevere donazioni ed effettuare qualunque altro tipo di spesa. I tesorieri, i quali sono membri del personale, si occupano della gestione di questo flusso.
- 3) La terza ed ultima parte (in viola) riguarda lo svolgimento delle mansioni.  
Da qui è possibile tener traccia di tutte le ditte, partner, civili o personale che hanno lavorato ad una mansione

Questi tre punti ci permettono di gestire in maniera efficiente il database, come dettato da specifiche.

## Schema Finale



Questo è lo schema finale completo di tutti gli attributi.

Nell'entità "Personale" possiamo trovare informazioni sui dipendenti che lavorano per l'organizzazione (quindi che percepiscono uno stipendio) come: dati anagrafici, residenza, ruolo svolto, contatti, qualifica, stipendio nominale ed eventuali dati di accesso al sito web.

Nell'entità "Sedi" abbiamo informazioni riguardo il nome della sede, i contatti e la posizione geografica.

La relazione "Impiegato" collega le entità "Personale" e "Sedi". Questa relazione ci informa sulle trasferte del dipendente tramite un attributo 'trasferta' e gli attributi 'data inizio' e 'data fine'.

Nell'entità "Mansioni" troviamo come informazioni il luogo e la data del compito svolto e una descrizione del lavoro fatto.

Nell'entità "Ditte esterne" troviamo una descrizione delle ditte che sono state ingaggiate per eseguire dei compiti specifici non svolgibili dal personale o dai volontari e di cui teniamo traccia archiviandone il nome ed i contatti.

Nell'entità "Partner" troviamo una descrizione di altre organizzazioni che collaborano con An Hand for Humanity, non a scopo di lucro. Teniamo traccia del loro nome e dei loro contatti.

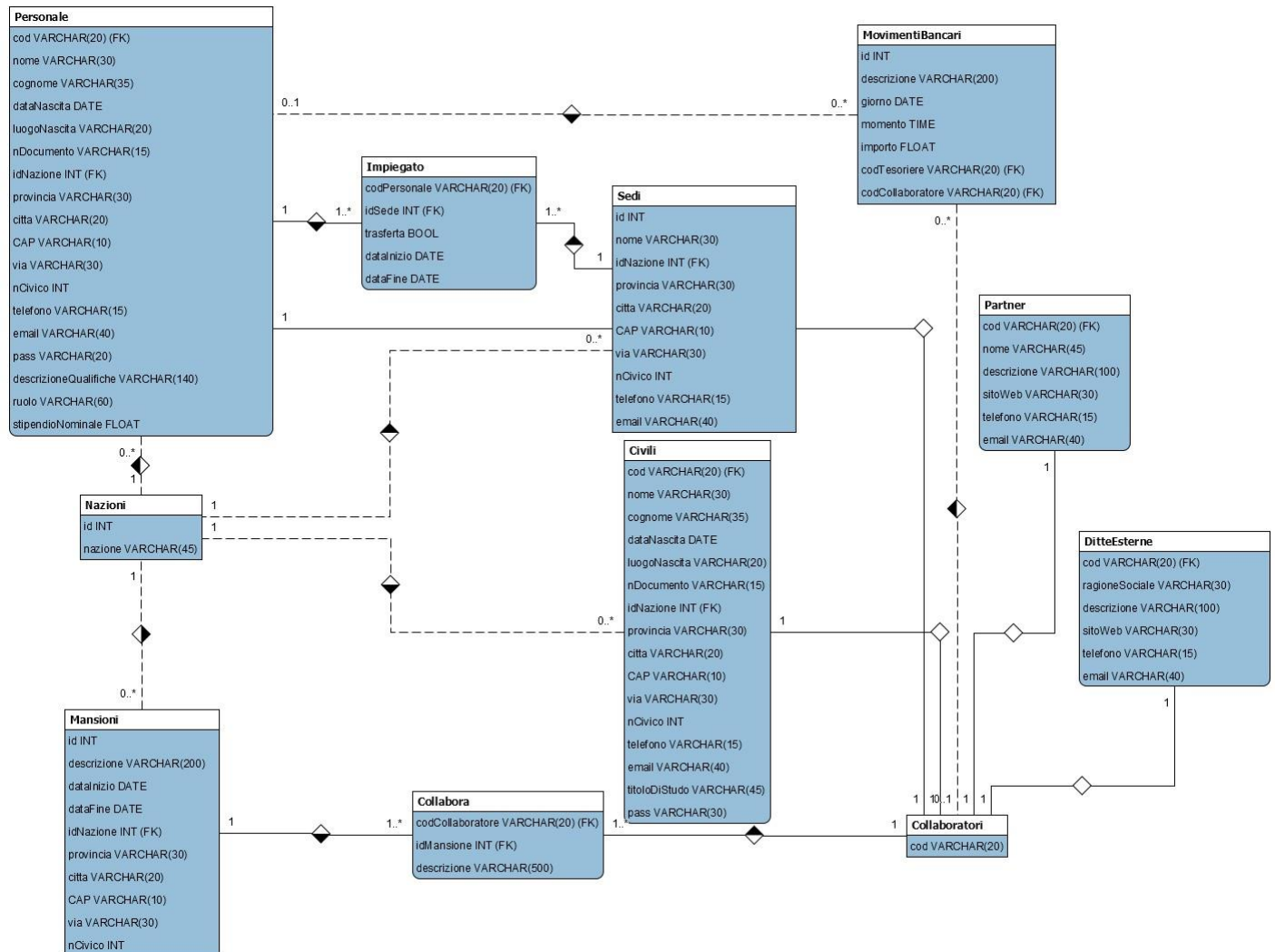
Infine nell'entità "Civili" gestiamo i dati anagrafici, la residenza, la qualifica e il ruolo svolto da persone al di fuori di organizzazioni che si impegnano a dare una mano, senza finalità economiche, allo svolgimento delle mansioni.

Le entità "Personale", "Ditte esterne", "Partner" e "Civili" sono generalizzazioni dell'entità "Collaboratori", pertanto le chiavi primarie delle suddette entità vengono inserite automaticamente, attraverso dei trigger, dentro l'entità collaboratori, per avere una più semplice gestione.

L'entità "Collaboratori" è quindi collegata all'entità "Mansioni" tramite relazione (n:n) perché un collaboratore può svolgere più mansioni e di conseguenza una mansione può essere svolta da uno o più collaboratori.

L'entità "Movimenti Bancari" ha lo scopo di tracciare tutti i movimenti del denaro nella cassa comune. In questa lista di movimenti possiamo vedere, la data, l'importo, la descrizione e, tramite relazioni (1:n) con le entità sopra citate, possiamo tenere traccia di quale tesoriere del personale effettua un eventuale pagamento e verso chi. Oppure se si tratta di una donazione, riusciamo a tenere traccia da chi ci viene effettuata, salvo se il donatore non vuol far conoscere la propria identità.

# Schema logico



Questo è lo schema logico, derivato dal diagramma ER soprastante.

Le chiavi primarie delle entità le abbiamo scelte di due tipi diversi: id INT, cod VARCHAR(20).

Questa scelta deriva dal fatto che, impostando tutte le chiavi di tipo INT all'interno dell'entità "Collaboratori", dove sfociano le entità figlie, si sarebbero creati molti conflitti. Onde evitare questo, abbiamo deciso di utilizzare queste chiavi di tipo VARCHAR(20) in modo che, una volta in "Collaboratori", la loro provenienza fosse facilmente distinguibile.

Da notare anche che, nella relazione "Impiegato" abbiamo impostato tre chiavi primarie: *codPersonale*, *idSede*, *dataInizio*. Questo perché, siccome possiamo avere più trasferte di un impiegato nella stessa sede, abbiamo deciso di impostare insieme alle altre anche la data d'inizio della trasferta (o lavoro). Questa terna ci garantirà l'univocità. In alternativa avremmo potuto trasformare la relazione "Impiegato" in un'entità, magari era la scelta più corretta, ma ad i fini pratici ci sembrava più sconveniente.

Cliccare [QUI](#) per aprire il progetto realizzato sul "workbench mysql".

---

# Codice SQL

Di seguito vi sono i link ai vari file .sql

- 1) [Creazione Utenti](#)
- 2) [Creazione Database & Trigger](#)
- 3) [Inserimento valori di esempio](#)
- 4) [Query utili](#)
- 5) [Procedure](#)

## Commenti al codice

- Per quel che riguarda i campi *pass* nelle tabelle “Personale” e “Civili” sarebbe buona norma, trattandosi di una password da utilizzare per l’account sul sito web, criptare tali campi in modo da renderli illeggibili a terze parti. Per una questione di semplicità nel realizzare il sito web, li abbiamo lasciati come campi leggibili.
- Relativamente alla generalizzazione dei collaboratori, in mysql non è possibile realizzare un campo che ospiti più di una chiave esterna. Per rimediare a questo, abbiamo creato quattro trigger: “cod\_personale”, “cod\_ditta”, “cod\_partner” e “cod\_civile”. Questi trigger fanno in modo che, quando viene inserito un nuovo record in una delle quattro entità figlie, il codice appena immesso venga automaticamente inserito anche nell’entità “Collaboratori”.
- In merito all’inserimento dei dati, si potrebbe presentare un ‘warning’ durante l’esecuzione dello script. Questo warning ci avverte che, nell’inserire valori nella tabella “MovimentiBancari” ci sono alcuni record con gli attributi ‘giorno’ e ‘momento’ nulli, cosa vietata nella creazione delle tabelle. Però avendo creato il trigger “inserimento\_movimento” questo fa in modo che, se venisse inserito un movimento bancario con i suddetti attributi nulli, imposti automaticamente la data e l’ora attuale.
- In mysql non è possibile creare dei ruoli, ma è possibile creare utenti. Abbiamo creato quattro tipologie di utente, più uno già esistente (root), atti a coprire tutti i casi di utenti che potrebbero utilizzare il database.



---

## DBMS alternativo

Come DBMS alternativo abbiamo scelto “MariaDB”.



MariaDB è un fork di MySQL, ovvero è un software scritto partendo dal codice sorgente di MySQL, dallo stesso programmatore di MySQL.

Abbiamo scelto MariaDB perché, come accennato sopra, è una versione ampliata e migliorata di MySQL. Dal punto di vista del codice SQL, le due sintassi sono quasi equivalenti tra loro. MariaDB ha il vantaggio però di avere motori di storage più avanzati e una migliore gestione degli utenti. Esso si appoggia al software “HeidiSQL” che permette l’utilizzo dei database attraverso una comoda ed intuitiva interfaccia grafica.

Cliccando [QUI](#) può trovare il codice per la creazione delle tabelle e dei triggers.

La sintassi per la creazione degli utenti è uguale alla sintassi usata per la creazione degli utenti in MySQL, per questo abbiamo riutilizzato lo stesso file .sql (che si trova in /Risorse/SQL/utenti.sql).

Come si nota facilmente, la sintassi è praticamente identica se non fosse che, nella creazione tabelle va specificato il set di caratteri da utilizzare, tramite l’istruzione “collate”.

Dalle ultime versioni di MariaDB non è più disponibile la gestione dei ruoli, così anche qui abbiamo optato per creare degli utenti.

---

## Sito Web

Per testare il funzionamento del DB abbiamo pensato di abbozzare un sito web.

Essendo l'organizzazione internazionale, abbiamo pensato di realizzare il suddetto sito in lingua inglese.

*Nota: l'ideale sarebbe stato fare anche il DB in inglese, ma non volevamo rischiare, visto che l'esame è incentrato sul DB*

Il sito è ospitato sui server di Altervista ed è accessibile cliccando [QUI](http://anhandforhumanity.altervista.org/anhandforhumanity/index.php).

(<http://anhandforhumanity.altervista.org/anhandforhumanity/index.php>)

I codici sorgente sono visionabili nella cartella “/Risorse/Sito\_Web/anhandforhumanity/”.

Il sito è molto semplice, serve solamente a dare un imprinting di quello che sarà poi un sito realmente strutturato e funzionante.

Nella home page è possibile farsi un'idea di cosa sia “An Hand for Humanity”, vi si trova la descrizione, i nostri compiti principali ed i nostri partner più importanti.

Nella pagina “offices” viene mostrato l'elenco delle nazioni e vengono colorate in rosso quelle in cui sono presenti una o più sedi.

Nella pagina “jobs” tramite un modulo di ricerca è possibile visualizzare tutte le mansioni svolte dall'organizzazione (non specificando niente si visualizzano tutte le mansioni).

Nella pagina “members area” è possibile fare il login come membro del personale oppure come civile.

Eseguendo l'accesso come membro del personale è possibile visualizzare l'accredito del proprio stipendio (nel caso si fosse tesoriere anche le spese effettuate) e tutte le trasferte effettuate.

Nel caso si acceda come civile, è possibile visualizzare le donazioni fatte. Le lasciamo dei riferimenti per provare questa pagina:

- personale:

email: [tonio.cartonio@libero.it](mailto:tonio.cartonio@libero.it)

password: [ciccino](#)

- civile:

email: [chuck.norris@chuck.norris](mailto:chuck.norris@chuck.norris)

password: [chucknorris](#)

Nella pagina “donate” è possibile fare donazioni all'organizzazione come anonimo o come membro registrato

---

(Al momento l'opzione non è disponibile, per un problema dovuto ad altervista con l'utilizzo del trigger per l'inserimento della data nei movimenti bancari).

Il file “/Risorse/Sito\_Web/anhandforhumanity/resources/query.php” è il file che contiene le query utilizzate per realizzare questo breve sito web.

Le query sono sotto forma alcune di funzioni e alcune di stringhe. Nel file mancano solamente le query per effettuare il login che, per comodità, abbiamo lasciato nel file “/Risorse/Sito\_Web/anhandforhumanity/login.php”.

Siccome l'account gratuito del server Altervista non ci permette di creare né trigger né utenti, abbiamo dovuto modificare la versione online del sito, bypassando questi ultimi.

## Informazioni aggiuntive

Potete visualizzare i risultati delle query [QUI](#) ( Delle query assegnate dal professore ).

Potete visualizzare il dump dei file [QUI](#).