

IIS JEAN MONNET

ELABORATO PER L'ESAME DI STATO  
ANNO 2020/2021

CAMPO SCOUT

ELIA COLOMBO

## Contents

TRACCIA ESAME DI STATO .....	3
INTRODUZIONE.....	3
SISTEMI E RETI: .....	5
Architettura di rete: .....	5
Disegno di rete:.....	7
Gestione sicurezza: .....	8
INFORMATICA:.....	9
Modello concettuale: .....	9
Obiettivo:.....	9
Ipotesi:.....	9
Entità: .....	10
Associazioni: .....	12
Descrizione associazioni: .....	12
Molteplicità: .....	13
Vincoli di integrità: .....	13
Schema ER .....	15
Modello logico: .....	16
Mapping: .....	16
Regole di lettura:.....	18
Disegno concettuale:.....	18
Esempio popolamento: .....	19
QUERY: .....	21
CODICE: .....	22
codice login amministratore:.....	22
Codice inserimento attività: .....	23
Codice iscrizione: .....	25
Visualizzazione delle diverse celle delle tabelle: .....	27
Codice Visualizzazione utenti: .....	28

## TRACCIA ESAME DI STATO

Noi Scout Il gruppo scout vuole realizzare una piattaforma per la gestione dei propri iscritti, delle proprie attività, degli eventi ed i campi estivi. Il candidato, servendosi anche delle proprie esperienze di stage, di PCTO e personali, analizzi la realtà di riferimento e individui una soluzione che a suo motivato giudizio sia la più idonea a sviluppare i seguenti punti:

1. il progetto, anche mediante rappresentazioni grafiche, dell'infrastruttura tecnologica ed informatica necessaria a gestire il servizio nel suo complesso, dettagliando:
  - a) l'architettura della rete e le caratteristiche del o dei sistemi server;
  - b) le modalità di comunicazione tra server e dispositivi utilizzati, descrivendo protocolli e servizi software da implementare;
2. L'analisi della base di dati per la gestione del servizio sopra descritto, in particolare
  - a. il modello concettuale;
  - b. il modello logico;
  - c. le interrogazioni significative in linguaggio SQL;
  - d. il progetto di massima della struttura funzionale del sistema, realizzando, con appropriati linguaggi a scelta sia lato client che lato server, un segmento significativo dell'app.

## INTRODUZIONE

L'associazione italiana "scout Assoraid" vuole realizzare una piattaforma per la gestione dei propri iscritti, delle proprie attività del campo estivo 2021.

Il progetto consiste in un interfaccia web divisa in:

- **ISCRIZIONE** → Consiste in una pagina accessibile senza necessità di fare il login, dove una qualsiasi persona iscritta all'associazione si potrà iscrivere al campo inserendo La sede di Appartenenza, la branca di cui fa parte, Il nome, il cognome, l'età, un'e-mail, il codice fiscale e in fine anche il codice persona dato dalla propria sede scout al momento della propria iscrizione. Una volta inseriti tutti i dati bisognerà premere un bottone per completare la registrazione.  
Un sistema esterno controllerà se la persona esiste veramente.  
Una volta eseguiti tutti i controlli verrà inviata un'e-mail sull'indirizzo dato per la conferma della registrazione.
- **LOGIN** → Consiste in una pagina che permettere agli amministratori del sito di poter fare il login inserendo l'e-mail e la password per accedere a funzioni nascoste dagli utenti standard.
- **INSERIMENTO ATTIVITA'** → Consiste in una pagina accessibile solo agli amministratori del sito dove potranno aggiungere un'attività all'interno del calendario specificando il nome

dell'attività, il giorno della settimana, l'ora in cui si andrà a fare quella determinata attività, in quale branca si farà l'attività in questione e infine una breve descrizione dell'attività.

- **VISUALIZZAZIONE ATTIVITA' →** Consiste in 3 pagine separate (una per branca) accessibili a tutti dove si potranno vedere tutte le attività inserite dagli amministratori per la campo scout estivo. Per gli amministratori all'interno di questa pagina apparirà un pulsante per poter accedere all'inserimento delle attività.
- **PAGINA DESCRITTIVA DEL CAMPO →** All'interno del sito ci sarà una pagina dedicata alla descrizione dettagliata del campo dove ci sarà inserito il luogo, il tema del campo, come sarà gestita la Cambusa (dispensa di cibo), ... tutte le caratteristiche del campo. Alla fine della pagina ci sarà una mini descrizioni di cosa sono gli scout e come sono nati.
- **2 PAGINE DI INTRODUZIONE →** nelle pagine di introduzione ci sarà: la descrizione generale del campo e delle foto degli scorsi campi.  
La differenza tra le due pagine di introduzione è la seguente:
  - Nella prima pagina home c'è un immagine che riporta al tema dello scoutista.
  - Nella seconda pagina home c'è un video che riporta ad un vecchio campo scout per far vedere come sarà la giornata al campo.
- **PAGINA CONTATTI →** In questa pagina ci sarà la possibilità di riuscire a contattare l'azienda con semplicità. Inserendo il nome, il proprio indirizzo di posta elettronica e il messaggio che si vorrà inviare poi si potrà premere un pulsante con scritto "send message" e sarà automaticamente inviato per e-mail all'associazione. In questa pagina si potrà osservare anche il luogo dove si trova l'associazione e gli orari di lavoro.
- **VISUALIZZAZIONE UTENTI →** l'amministratore dopo aver fatto il login visualizzerà una pagina nascosta al pubblico dove potrà inserire diversi filtri per la ricerca dell'utente. Così da poter monitorare la situazione delle iscrizioni. Sempre nella stessa pagina ci sarà il numero degli iscritti fino a quel momento.

## SISTEMI E RETI:

### Architettura di rete:

Per il corretto funzionamento del sito bisognerà realizzare uno schema di rete.

Per realizzarlo servirà:

- Una rete interna aziendale che è composta da una connessione internet in fibra ottica che consente di avere una connessione a banda ultra-larga con fibra ottica FTTH. Questa soluzione risulta la più costosa ma ha delle prestazioni perfette per ogni evenienza. Questa tecnologia di rete permette di avere una velocità stimata di 1 gb/s in download e 300 mb/s in upload. I dispositivi all'interno dell'azienda saranno connessi ad internet tramite dei cavi ethernet collegati ad uno switch e a loro volta saranno collegati ad un router.
- Quando parliamo di schema di rete non ci dobbiamo dimenticare la gestione del server web e del server di backup. Per questo motivo si possono tenere in conto due opzioni:
  - Un Cloud Hosting esterno (Es. Aruba)
  - Server interno.

Per capire quale possiamo utilizzare dobbiamo analizzare la situazione nell'uso di entrambi i servizi:

#### ➤ Se dovessi utilizzare un Cloud Hosting ↓

Tra i vantaggi possiamo inserire:

1. Nessun investimento iniziale;
2. Meno costi annui (per quanto riguarda l'abbonamento annuale del Cloud Hosting);
3. Possibilità di migliorare le performance semplicemente richiedendo al provider un nuovo servizio;
4. Risparmio energetico e di spazi, in quanto non bisognerà inserire all'interno della propria azienda un server che sicuramente occuperà spazio e consumerà molta elettricità;

Tra gli svantaggi possiamo inserire:

1. Lentezza nel caso di guasti;
2. Possibile perdita di dati legati a problemi del provider;

#### ➤ Se dovessi utilizzare un server interno ↓

Tra i vantaggi possiamo inserire:

1. Pieno controllo sui propri dati;
2. Utilizzo di regole private per quanto riguarda la sicurezza e la privacy dei dati;
3. In caso di guasto si potrà intervenire immediatamente per riuscire a risolvere il problema;

Tra gli svantaggi possiamo inserire:

1. Costo enorme iniziale per l'acquisto e la configurazione del server;
2. Bisogno di personale per la gestione e la manutenzione del server interno;
3. Costi elevati per migliorare il servizio
4. Costi elevati per la manutenzione hardware e per garantire la sicurezza dei dati.

Analizzando le due proposte per la gestione del sito è stato scelto di utilizzare un servizio di Cloud Hosting esterno fornito da un provider.

Il Cloud Hosting è un servizio che permette di archiviare i propri dati attraverso dei server cloud che virtualizzano i dati.

Come già detto in precedenza usando un servizio Cloud Hosting andiamo incontro a numerosi vantaggi. Uno dei più grandi vantaggi è proprio il costo ridotto, evitando i costi per l'acquisto e per la configurazione del server.

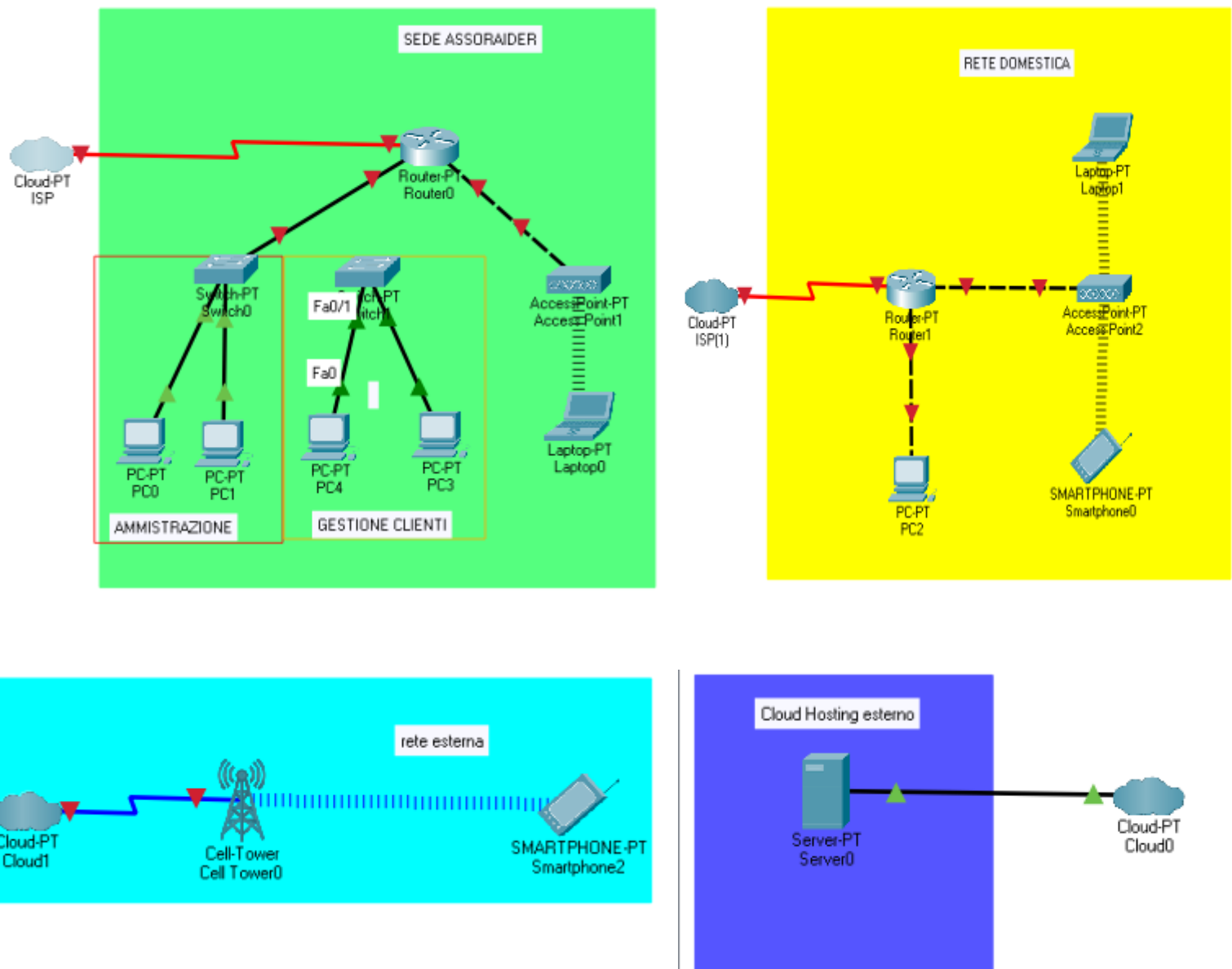
Una caratteristica importante del Cloud Hosting è la sincronizzazione, in modo da riuscire a lavorare su dati/documenti aziendali condivisi con tutti, con la possibilità di lavorare anche in remoto. Il cloud hosting offre anche un servizio di sicurezza, in grado di tutelare le aziende. Offre anche la possibilità di aggiornamenti veloci direttamente gestiti dal provider.

La società Assoraider per la gestione del suo sito fornisce i servizi tramite:

- Protocollo HTTPS
- Database esterno

Per i servizi software si è deciso di sviluppare una piattaforma web dove si utilizzerà Php come linguaggio server e HTML, CSS, JAVASCRIPT per lo sviluppo grafico e funzionale della pagina web.

## Disegno di rete:



Nel disegno di rete possiamo notare:

i diversi tipi di rete (Sede Assoraider, rete domenista e la rete esterna) che comunicano con il cloud esterno per accedere al sito web per l'iscrizione e la visualizzazione delle attività del campo scout. Ovviamente la sede assoraider non accede solo al sito web ma la parte amministrativa potrà visualizzare tutti i dati all'interno del database.

All'interno dell'azienda ci sono due vLan per riuscire a separare la parte amministrativa e la parte di gestione clienti.

VLAN (Virtual Local Area Network) È un tipo di LAN, con l'unica differenza che si tratta di una rete virtuale.

Semplificando molto il concetto, una VLAN è una rete fittizia di dispositivi che appartengono a una o più reti LAN. Una VLAN serve a Suddividere i computer collegati a una rete in base a certe caratteristiche, risparmiando sulle risorse di rete e sui tempi di configurazione e manutenzione.

In questo caso si creano 2 vlan una per la parte amministrativa e una per la parte di gestione clienti le quali saranno connesse ad internet tramite cavo ma saranno viste dal sistema interno come due reti diversi.

## Gestione sicurezza:

Per la protezione dei dati relativi alla registrazione da parte dell'utente al campo scout, alle attività che si andranno ad inserire e tutti i dati interni all'azienda (dati amministratori, ...), è stato abilitato il protocollo HTTPS. HTTPS, è una variante sicura e certificata del protocollo HTTP.

Il Protocollo HTTP lavora con un'architettura client/server, il client esegue una richiesta e Il server restituisce una risposta, usando il protocollo HTTPS garantisce che solamente il Client e il server possano leggere e modificare i dati che vengono scambiati.

Il sito provvederà anche ad un canale diretto di comunicazione tra cliente e azienda. Di fatto il cliente potrà inviare un'e-mail direttamente dal sito web all'azienda per qualsiasi dubbio.

Un altro aspetto da considerare è la protezione di dati sensibili. Il sistema di registrazione al campo è utilizzato da numerosi utenti e per la gestione della sicurezza ci si affida alle competenze dei programmatori, i quali si occuperanno di sviluppare un sistema sicuro che non permetterà di visualizzare i dati privati inseriti dagli utenti.

Inoltre, all'interno del sito ci sarà un sistema che permetterà agli amministratori del sito di visualizzare un bottone nascosto all'utente "normale". Per visualizzare questo bottone basterà effettuare il login nella piattaforma dedicata (inserendo l'e-mail e la password fornita dalla associazione).

Tutte le comunicazioni "Assoraider" con i propri clienti verranno effettuate tramite posta elettronica, grazie al protocollo SMTP. Questo protocollo è soggetto a numerose vulnerabilità, come ad esempio attacchi del tipo MITM (Man-In-The-Middle) o di spoofing.

Per riuscire a minimizzare questi attacchi sarà necessario attuare una serie di precauzioni di Data Loss Prevention (DLP) così da riuscire ad avere una maggiore sicurezza e solidità per quanto riguarda la comunicazione.

Possibili procedure per la sicurezza del messaggio:

- utilizzo di protocolli di crittografia (come S/MIME, TLS)
- utilizzo della PEC (Posta Elettronica Certificata) per comunicazioni ufficiali e legali
- applicazione di protocolli di filtraggio e verifica delle mail in entrata (ad esempio DMARC, SPF, DKIM) sulle caselle dei dipendenti.



## INFORMATICA:

### Modello concettuale:

#### Obiettivo:

L'associazione italiana "scout Assoraider" vuole realizzare una piattaforma per la gestione dei propri iscritti, delle proprie attività del campo estivo 2021.

#### Ipotesi:

- Per poter inserire una nuova attività bisognerà fare il login (login amministratore) così comparirà un pulsante che porterà direttamente alla pagina di aggiunta dell'attività.
- Si ipotizza che all'iscrizione della persona nell'associazione Assoraider in una certa sede, gli verrà dato un codice identificativo che servirà per potersi iscrivere al campo estivo.
- Si ipotizza che ogni sede scout indichi ai propri iscritti il link del sito per riuscire ad iscriversi correttamente al campo.
- Per potersi iscrivere al campo scout bisognerà essere iscritti alla associazione "Assoraider" per questo motivo bisognerà inserire il codice\_persona, dato dalla propria sede, per riuscire a completare l'iscrizione. Per assicurarsi che quel codice persona esista ci si affiderà ad un sistema esterno che controlli l'esistenza effettiva di quella persona.
- Dopo essersi scritto al campo scout si ipotizza che eventuali messaggi da parte dell'azienda verranno fatte tramite e-mail.
- Per qualsiasi dubbio si potrà scrivere per e-mail all'azienda o contattare nei giorni in indicati nel sito.
- Si ipotizza che per riuscire ad iscrivere non servirà effettuare un login.
- Per poter vedere le diverse attività che si faranno durante il campo non servirà di un login, visto che non si tratta di dati sensibili ma solo delle specifiche del campo scout.
- Per accedere al sito non servirà fare un login così da poter far visualizzare a più persone possibili la nostra pagina web. In questo modo le persone estranee potranno interessarsi al mondo dello scoutismo e cercare più informazioni.
- Effettuando il login si potrà accedere ad una pagina nascosta agli utenti standard. In questa pagina ci sarà la possibilità di vedere il numero delle persone che si sono iscritte e anche la possibilità di poter inserire dei filtri per controllare le diverse iscrizioni.

## Entità:

### Iscritto:

- codice\_persona → indica il codice dato alla singola persona una volta iscritta all'associazione "Assoraider";
- nome → indica il nome della persona che si vuole iscrivere;
- cognome → indica il cognome della persona che si vuole iscrivere;
- codice\_fiscale → indica il codice fiscale della persona che si vuole iscrivere;
- patologie → in questo attributo saranno riportate tutte le patologie/allergie della persona iscritta al campo;
- email → in quest'attributo ci sarà l'email della suddetta persona così facendo se ci saranno delle comunicazioni verranno inviate direttamente per email;

### Branca:

- Nome\_Branca → indica il nome del gruppo di appartenenza (questi gruppi sono divisi per età). I lupetti da quando si ha 6 anni fino a 10/11 anni, esploratori da 11/12 anni fino a 14/15 anni e compagnia da 15/16 a 18/19 anni;

### Attività:

- Nome\_attività → indica il nome dell'attività che si andrà a svolgere durante la settimana di campo;
- Giornata\_settimana → indica in quale giorno si andrà a fare questa determinata attività;
- Ore\_attività → indica in che ora si andrà a svolgere l'attività proposta;
- Spiegazione → in quest'attributo ci sarà la spiegazione dell'attività

### Sedi\_scout:

- Sede → indica il nome della sede;
- Provincia → indica in quale provincia si trova la sede;
- Email → indica un email della sede;

### Amministratore:

- Nome → indica il nome dell'amministratore;
- Cognome → indica il cognome dell'amministratore;
- Email → indica l'email dell'amministratore;
- Psw → indica la password dell'amministratore;

**Tabella entità:**

<b>Nome</b>	<b>Descrizione</b>	<b>Attributi</b>	<b>Identificatore</b>
Iscritto	Entità contenente tutte le informazioni relative ad un singolo iscritto al campo	codice_persona, nome, cognome, codice_fiscale, patologie, email	id_iscritti
branca	Entità contenente tutte le Branche dello scoutismo	Nome_branca	Id_branca
attività	Entità contenente tutte le attività di ogni branca che si andranno ad effettuare per tutta la settimana di campo	Nome_attivita, Giorno_settimana,  Ore_attivita, spiegazione	Id_attivita
Sedi_scout	Entità contenente tutte le sedi italiane che fanno parte dell'associazione Assoraiders	Sede, Provincia, email	id
Amministratore	Entità contenente tutti gli utenti amministratori della pagina web	Nome, Cognome, email, psw	Id_Ammministratore

## Associazioni:

Nome	Entità coinvolte	Molteplicità	Attributi
appartiene	Iscritto, branca	N, 1	//
svolge	Branca, attività	1, N	//
È associato	Iscritto, Sedi_scout	N, 1	//
Inserito	Amministratore, Attività	1, N	//

## Descrizione associazioni:

appartiene:

L'associazione indica in quale branca (branco/reparto/compagnia) fa parte l'iscritto al campo scout;

Svolge:

L'associazione indica quale branca durante la settimana farà una determinata attività;

è associato:

L'associazione indica in quale associazione fa parte l'utente che si è iscritto al campo;

inserito:

L'associazione indica quale amministratore ha inserito l'attività;

## Molteplicità:

All'interno del progetto ci sono le seguenti molteplicità:

- Appartiene (Iscritto, Branch): N-1, Ad ogni Iscritto appartiene ad una sola Branch. Una Branch può contenere uno o più iscritti.
- Svolge (Branch, Attività): 1-N, Una singola branch può svolgere una o più attività. Un'attività deve essere svolta da una sola branch.
- Associato (Iscritto, Sede): N-1, Ad ogni iscritto è associato una sola sede. Una sede può avere uno o più iscritti.
- Inserito (Amministratore, Attività): 1-N, Un singolo amministratore può inserire più attività. Un'attività devono essere inserite da un solo amministratore.

## Vincoli di integrità:

1. Vincoli di integrità su chiave primaria su:

- id\_iscritto;
- id\_branch;
- id\_attività;
- id (della sede);
- id\_amministratore;

2. Vincoli d'integrità espliciti:

Entità iscritto:

- L'attributo Codice\_persona (tipo varchar massimo 10 caratteri);
- Gli attributi Nome e Cognome all'interno ci dovranno essere solo lettere (tipo varchar massimo 20 caratteri);
- L'attributo codice fiscale dovrà essere formato da: 3 caratteri alfabetici per il cognome, 3 caratteri alfabetici per il nome, 2 caratteri numerici per l'anno di nascita, 1 carattere alfabetico per il mese di nascita, 2 caratteri numerici per il giorno di nascita ed il sesso, 4 caratteri associati al comune oppure allo stato estero di nascita, 1 carattere alfabetico usato come carattere di controllo (tipo Varchar massimo 16 caratteri);
- L'attributo patologie (tipo Varchar massimo 100 caratteri);
- L'attributo Email dovrà esserci per forza l'uso della @ (tipo varchar massimo 50 caratteri);

Entità Branca:

- Nell'attributo Nome\_branca sarà possibile inserire solamente "Lupetti", "Reparto", "Compagnia";

Entità Attività:

- L'attributo Nome\_attivita (Tipo Varchar massimo 40 caratteri);
- L'attributo Giorno\_settimana potrà contenere soltanto i 7 giorni della settimana (lunedì, martedì, ...);
- L'attributo Ore\_attività potrà solo contenere: "9:00", "11:00", "15:00" e "18:00" (tipo date);
- Spiegazione (tipo text e non ci sono limiti di caratteri)

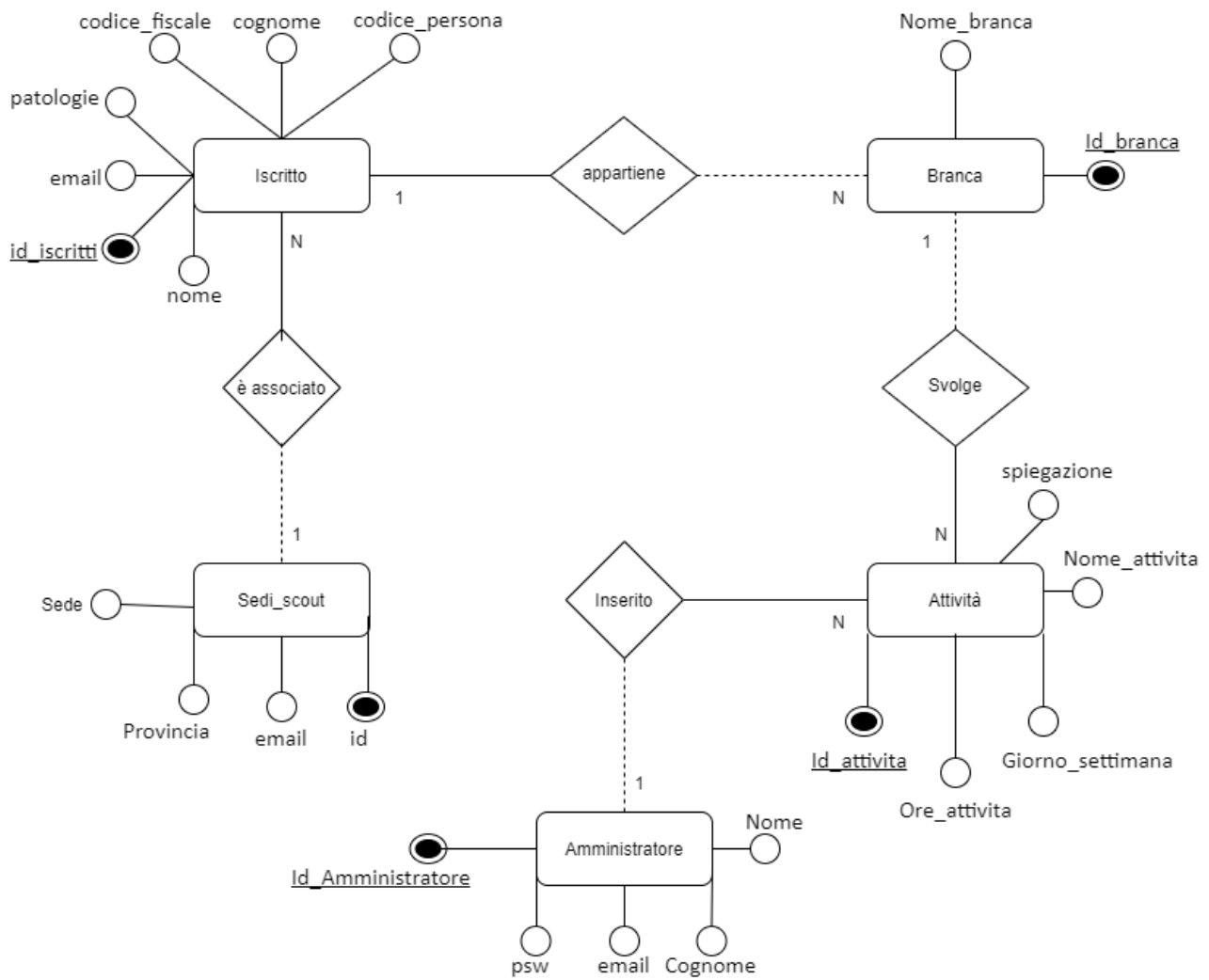
Entità Sedi\_Scout:

- L'attributo Sede (Tipo Varchar Massimo 50 caratteri);
- L'attributo Provincia (Tipo Varchar Massimo 2 caratteri);
- L'attributo Email dovrà esserci per forza l'uso della @ (tipo varchar massimo 50 caratteri);

Entità Amministratore:

- Gli attributi Nome e Cognome all'interno ci dovranno essere solo lettere (tipo varchar massimo 20 caratteri);
- L'attributo Email dovrà esserci per forza l'uso della @ (tipo varchar massimo 50 caratteri);
- L'attributo psw dovrà contenere almeno 1 maiuscola, un carattere speciale e un numero (Tipo Varchar massimo 50 caratteri);

## Schema ER



## Modello logico:

### Mapping:

Mapping dell'associazione appartiene:

Associazione diretta: appartengono (Totale)

Associazione inversa: può appartenere (Parziale)

Associazione binaria "appartengono" tra "iscritto" e "branca" di molteplicità N:1

- Iscritto (id\_iscritti, codice\_persona, sede\_appartenenza, id\_brancaEs, nome, cognome, codice\_fiscale, patologie, email).
- Branca (id\_branca, Nome\_branca)

Con l'attributo "id\_brancaEs" dell'entità "iscritto" risulta essere chiave esterna sull'attributo "id\_branca" dell'entità Branca.

$\text{VRid\_brancaEs(iscritto)} \subseteq \text{VRid\_branca (Branca)}$

Vincolo Referenziale dovuto dalla totalità dell'associazione diretta.

Mapping dell'associazione svolge:

Associazione diretta: svolge (Parziale)

Associazione inversa: essere svolta (Totale)

Associazione binaria "svolge" tra "Branca" e "Attività" di molteplicità 1: N

- Branca (id\_branca, Nome\_branca)
- Attività (id\_attività, Nome\_attività, Giorno\_settimana, Ore\_attività, Id\_brancaEs2, id\_AmmministratoreEs, spiegazione)

Con l'attributo "id\_brancaEs2" dell'entità "Attività" risulta essere chiave esterna sull'attributo "id\_branca" dell'entità Branca.

$\text{VRid\_brancaEs2(Attività)} \subseteq \text{VRid\_branca (Branca)}$

Vincolo Referenziale dovuto dalla totalità dell'associazione inversa.

Mapping dell'associazione Associato:

Associazione diretta: è associato (Totale)

Associazione inversa: ha associato (Parziale)

Associazione binaria "è associato" tra "iscritto" e "Sedi\_scout" di molteplicità N:1



- Iscritto (id\_iscritti, codice\_persona, sede\_appartenenza, id\_brancaEs, nome, cognome, codice\_fiscale, patologie, email).
- Sedi\_scout (Sede, Provincia, email, id)

Con l'attributo "Sede\_appartenenza" dell'entità "iscritto" risulta essere chiave esterna sull'attributo "id\_sede" dell'entità "sedi\_scout";

$\text{VRid\_sede (Sedi\_scout)} \subseteq \text{VRSede\_appartenenza(iscritto)}$

Vincolo Referenziale dovuto dalla totalità dell'associazione diretta.

Mapping dell'associazione Inserito:

Associazione diretta: Inserito (Parziale)

Associazione inversa: è stata inserita (Totale)

Associazione binaria "Inserito" tra "Amministratore" e "attività" di molteplicità 1:N

- Amministratore (id\_amministratore, Nome, Cognome, email, psw)
- Attività (id\_attività, Nome\_attività, Giorno\_settimana, Ore\_attività, Id\_brancaEs2, id\_AmministratoreEs, spiegazione)

Con l'attributo "id\_AmministratoreEs" dell'entità "attività" risulta essere chiave esterna sull'attributo "id\_amministratore" dell'entità "amministratore";

$\text{VRid\_amministratore (Amministratore)} \subseteq \text{VRid\_amministratoreEs (Attività)}$

Vincolo Referenziale dovuto dalla totalità dell'associazione inversa.

## Regole di lettura:

regola lettura Appartiene:

Uno o più iscritti appartengono a una sola branca, Una Branca può avere zero o più iscritti.

Regola lettura Svolge:

Una Branca può svolgere zero o più attività, Un'attività deve essere svolta da una sola branca.

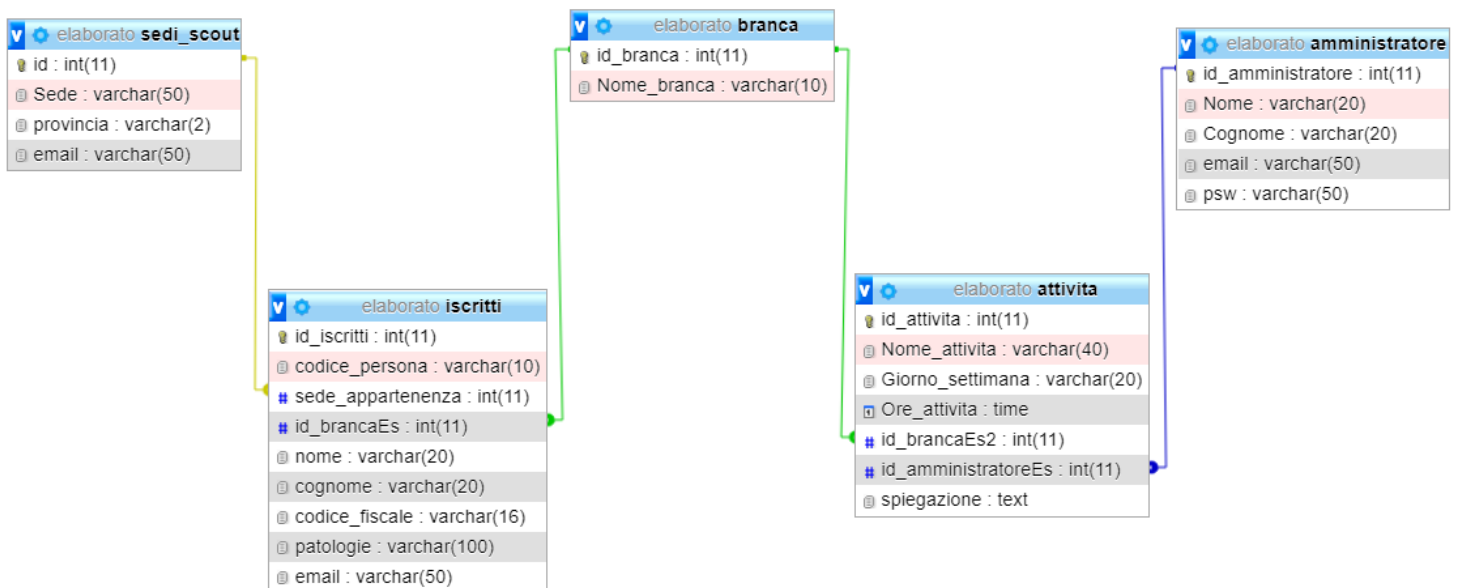
Regola lettura Associato:

zero o più iscritti devono essere associati a una sola sede, Una sede può avere zero o più iscritti.

Regola lettura Inserito:

Un amministratore può inserire una o più attività, Un'attività deve essere stata inserita da un solo amministratore.

## Disegno concettuale:



Esempio popolamento:

**TABELLA ATTIVITA:**

id_attivita	Nome_attivita	Giorno_settimana	Ore_attivita	Id_brancaEs2	Id_amministratoreEs	spiegazione
1	Roverino	lunedì	9:00	2	1	In questo gioco ci saranno due porte e bisognerà cercare di appoggiare un corda arrotolata su se stessa aldilà della porta. ...
2	Scalpo	martedì	9:00	2	1	Bisognerà rubare la promessa all'avversario che sarà attaccata dietro di lui...
3	Bandiera scout	martedì	11:00	1	2	Ogni componente della squadra gli verrà assegnato un numero e quando verrà chiamato dovrà cercare di recuperare la promessa e tornare alla propria base

**TABELLA AMMINISTRATORE:**

id_amministratore	Nome	Cognome	Email	psw
1	Francesco	Bongiovanni	Francesco.bongiovanni@gmail.com	Ciaocomeva.10
2	Giovanni	Colombo	Giovanni.colombo@gmail.com	Supercalifragili.11

**TABELLA BRANCA:**

id_branca	Nome_branca
1	Lupetti
2	Reparto
3	Compagnia

**TABELLA ISCRITTI:**

id_iscritti	Codice_persona	Sede_appartenza	Id_brancaEs	Nome	Cognome	Codice_fiscale	Patologie	email
1	ABCD123456	2	2	Giusep	Ross	GSPRSS	Allergico	Giuspp.r@gmail.com

				pe	i	80A01B 639N	al polline	
2	EFGH123456	2	2	Elisa	Mari ani	LSEMR N80A01 B639U	(vuoto)	Eli.Mari@gmail.com
3	ABCD67890	1	1	Frances ca	Mell i	FRNML L80A01 B639H	Intolleranz a al lattosio	Fra.Melli@gmail.com

**TABELLE SEDI\_SCOUT:**

id	Sede	Provincia	Email
1	Sezione di Andria	BT	andria@assoraider.it
2	Sezione di Bagheria	PA	bagheria@assoraider.it
3	Sezione di senna comasco	CO	como@assoraider.it

**QUERY:**

- `INSERT INTO `amministratore` (`id_amministratore`, `Nome`, `Cognome`, `email`, `password`) VALUES (NULL, 'Francesco', 'Bongiovanni', 'Francesco.bongiovanni@gmail.com ', MD5('Ciaocomeva.10 '));`

Grazie a questa query si potrà inserire all'interno del database nella tabella "amministratore" un amministratore.

- `$sql = "SELECT * FROM amministratore WHERE email ='$email' AND psw = '$psw'";`

Grazie a questa query si controllerà se all'interno della tabella amministratore c'è un email associata ad una password. Se all'interno del database esiste effettuerà il login se no visualizzerà un messaggio di errore.

- `$sql = "INSERT INTO `attivita` (`id_attivita`, `Nome_attivita`, `Giorno_settimana`, `Ore_attivita`, `id_brancaEs2`, `id_amministratoreEs`, `spiegazione`) VALUES (DEFAULT, '$Nome_attivita . "', '$Giorno_settimana . "', '$Ore_attivita . "', '$id_brancaEs2 . "', '$id_amministratoreEs . "', '$spiegazione . "')";`

Grazie a questa query si riuscirà ad inserire nella tabella "attività" un attività con tutti i dati.

- `$sql = "INSERT INTO `iscritti` (`id_iscritti`, `codice_persona`, `sede_appartenenza`, `id_brancaEs`, `nome`, `cognome`, `codice_fiscale`, `patologie`, `email`) VALUES (DEFAULT, '$codice_persona.', '$sede_appartenenza.', '$id_brancaEs.', '$nome.', '$cognome.', '$codice_fiscale.', '$patologie.', '$email.');`

Grazie a questa query si riuscirà ad inserire nella tabella "iscritti" un iscritto con tutti i dati.

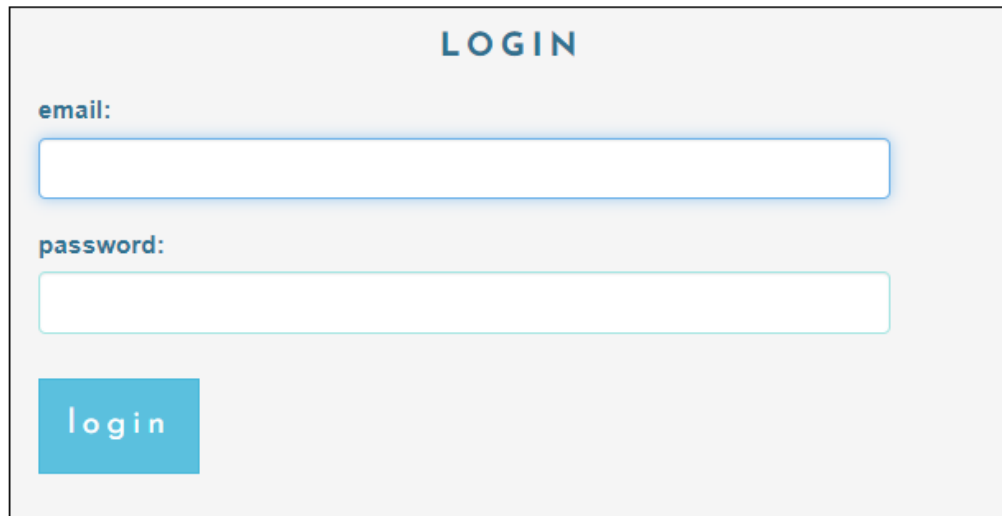
- `$sql_Ore9 = "SELECT Nome_attivita, Giorno_settimana FROM attivita WHERE Ore_attivita = '9:00:00' AND `id_brancaEs2` = 2";`

Grazie a questa query andremo a ricavare il nome dell'attività e il giorno della settimana dalla tabella "attività" dove le ore delle attività è = "9:00:00" e l'id della branca è uguale a 2.

Questo tipo di query si andrà a rifare più volte per le diverse ore delle attività e per le diverse branche. Ma la struttura della query è sempre la stessa.

## CODICE:

codice login amministratore:



**LOGIN**

email:

password:

```

<?php
session_start();
include "connessione.php";

if ($_SERVER["REQUEST_METHOD"] == "POST")
{
    include "connessione.php";

    $email = $_POST["email"];
    $psw = $_POST["psw"];
    $psw = md5($psw);

    echo "email: ".$email;
    echo "<br>";
    echo "psw: ".$psw;

    $sql = "SELECT * FROM amministratore WHERE email = '$email' AND psw = '$psw'";
    $result = $conn->query($sql);
    echo $sql;

    if ($result->num_rows > 0) {
        while ($row = $result->fetch_assoc()) {
            $_SESSION["id_amministratore"] = $row["id_amministratore"];
        }
        header("Location: services.php");
    }else{
        header("Location: login_amministratore.php?insertresult=fail");
    }

    exit();
}
?>

```

Codice inserimento attività:

## INSERIMENTO ATTIVITA'

**Nome della nuova attività:**

**in che branca vuoi inserire l'attività:**

**Giorno attività:**

**Ora:**

**Spiegazione attività:**

inserire

```

1 <?php
2 session_start();
3 include "connessione.php";
4
5
6 if ($_SERVER["REQUEST_METHOD"] == "POST") {
7     include "connessione.php";
8
9     $Nome_attivita = $_POST["Nome_attivita"];
10    $Giorno_settimana = $_POST["Giorno_settimana"];
11    $Ore_attivita = $_POST["Ore_attivita"];
12    $id_brancaEs2 = $_POST["id_brancaEs2"];
13    $spiegazione = $_POST["spiegazione"];
14    $id_amministratoreEs = $_SESSION["id_amministratore"];
15
16    echo "Nome_attivita: " . $Nome_attivita;
17    echo "<br>";
18    echo "Giorno_settimana: " . $Giorno_settimana;
19    echo "<br>";
20    echo "Ore_attivita: " . $Ore_attivita;
21    echo "<br>";
22    echo "id_brancaEs2: " . $id_brancaEs2;
23    echo "<br>";
24    echo "spiegazione: " . $spiegazione;
25    echo "<br>";
26    echo "Amministratore: " . $id_amministratoreEs;
27    echo "<br>";
28    echo "<br>";
29
30    $sql = "INSERT INTO `attivita` (`id_attivita`,`Nome_attivita`,`Giorno_settimana`,`Ore_attivita`,`id_brancaEs2`,`id_amministratoreEs`,`spiegazione`)
31    VALUES (DEFAULT,'" . $Nome_attivita . "','" . $Giorno_settimana . "','" . $Ore_attivita . "','" . $id_brancaEs2 . "','" . $id_amministratoreEs . "','" . $spiegazione . "')";
32
33    //echo $sql;
34
35    if ($conn->query($sql) === TRUE) {
36
37        $sql = "SELECT * FROM `iscritti` WHERE
38        `Nome_attivita` = '" . $Nome_attivita . "' AND
39        `Giorno_settimana` = '" . $Giorno_settimana . "' AND
40        `Ore_attivita` = '" . $Ore_attivita . "' AND
41        `id_brancaEs2` = '" . $id_brancaEs2 . "' AND
42        `id_amministratoreEs` = '" . $id_amministratoreEs . "' AND
43        `spiegazione` = '" . $spiegazione . "'";
44
45        $result = $conn->query($sql);
46
47        if ($result->num_rows > 0) {
48            while ($row = $result->fetch_assoc()) {
49
50                $result = $conn->query($sql);
51
52                if ($result->num_rows > 0) {
53                    while ($row = $result->fetch_assoc()) {
54                        header("Location: aggiungi_attivita.php?insertresult=success&id=" . $row["id_attivita"]);
55                    }
56                } else {
57                    echo "Error: " . $sql . "<br>" . $conn->error;
58                    header("Location: aggiungi_attivita.php?insertresult=fail");
59                }
60            } else {
61                //header("Location: Connection.php?insertresult=fail");
62
63                echo "Error: " . $sql . "<br>" . $conn->error;
64            }
65
66            exit();
67        }
68    }
69    ?>

```



Codice iscrizione:

## Register

**SEDE APPARTENENZA:**

**branca:**

**Codice persona:**

**Nome:**

**Cognome:**

**Codice fiscale:**

**email:**

**Patologie:(inserire all'interno se si hanno allergie, problemi alimentari,...)**

registrarsi

```

1 <?php
2
3 if ($_SERVER["REQUEST_METHOD"] == "POST")
4 {
5     include "connessione.php";
6
7     $codice_persona = $_POST["codice_persona"];
8     $sede_appartenenza = $_POST["sede_appartenenza"];
9     $id_brancaEs = $_POST["id_brancaEs"];
10    $nome = $_POST["nome"];
11    $cognome = $_POST["cognome"];
12    $codice_fiscale = $_POST["codice_fiscale"];
13    $patologie = $_POST["patologie"];
14    $email = $_POST["email"];
15
16    echo "codice_persona: ".$codice_persona;
17    echo "<br>";
18    echo "sede_appartenenza: ".$sede_appartenenza;
19    echo "<br>";
20    echo "id_brancaEs: ".$id_brancaEs;
21    echo "<br>";
22    echo "nome: ".$nome;
23    echo "<br>";
24    echo "cognome: ".$cognome;
25    echo "<br>";
26    echo "codice_fiscale: ".$codice_fiscale;
27    echo "<br>";
28    echo "patologie: ".$patologie;
29    echo "<br>";
30    echo "email: ".$email;
31    echo "<br>";
32    echo "<br>";
33    echo "<br>";
34
35    $sql = "INSERT INTO `iscritti` (`id_iscritti`, `codice_persona`, `sede_appartenenza`, `id_brancaEs`, `nome`, `cognome`, `codice_fiscale`, `patologie`, `email`)
36    VALUES (DEFAULT, '".$codice_persona."', '".$sede_appartenenza."', '".$id_brancaEs."', '".$nome."', '".$cognome."', '".$codice_fiscale."', '".$patologie."', '".$email."')";
37
38    //echo $sql;
39
40    if ($conn->query($sql) === TRUE)
41    {
42
43        $sql = "SELECT * FROM `iscritti` WHERE
44        `codice_persona` = '".$codice_persona.'" AND
45        `sede_appartenenza` = '".$sede_appartenenza.'" AND
46        `id_brancaEs` = '".$id_brancaEs.'" AND
47        `nome` = '".$nome.'" AND
48        `cognome` = '".$cognome.'" AND
49        `codice_fiscale` = '".$codice_fiscale.'" AND
50        `patologie` = '".$patologie.'" AND
51        `email` = '".$email."';";
52
53        $result = $conn->query($sql);
54
55        if ($result->num_rows > 0)
56        {
57            while($row = $result->fetch_assoc())
58            {
59
60                header("Location: registrarsi.php?insertresult=success&id=".$row["id_iscritti"]);
61            }
62        }
63        else
64        {
65            echo "Error: " . $sql . "<br>". $conn->error;
66            header("Location: registrarsi.php?insertresult=fail");
67        }
68    }
69    else
70    {
71        //header("Location: Connection.php?insertresult=fail");
72    }
73    echo "Error: " . $sql . "<br>". $conn->error;
74    }
75
76    exit();
77 }
78 >>

```

## Visualizzazione delle diverse celle delle tabelle:

ORARI	Lunedì	Martedì	Mercoledì	Giovedì	Venerdì	Sabato	Domenica
ORE 8:00	COLAZIONE						
ORE 9:00 (1° attività mattutina)	roverino	ATTIVITA' NON ANCORA INSERITA	hike	ATTIVITA' NON ANCORA INSERITA	ATTIVITA' NON ANCORA INSERITA	ATTIVITA' NON ANCORA INSERITA	ATTIVITA' NON ANCORA INSERITA
ORE 11:00 (2° attività mattutina)	Montaggio tende	ATTIVITA' NON ANCORA INSERITA	hike	lkjhahdf	ATTIVITA' NON ANCORA INSERITA	ATTIVITA' NON ANCORA INSERITA	smontaggio tende
ORE 13:00	PRANZO						
ORE 15:00 (1° attività pomeridiana)	Montaggio tende	ATTIVITA' NON ANCORA INSERITA	incontri con le proprie sezioni	Giochi misti	ATTIVITA' NON ANCORA INSERITA	ATTIVITA' NON ANCORA INSERITA	ATTIVITA' NON ANCORA INSERITA
ORE 17:00	MERENDA						
ORE 18:00 (2° attività pomeridiana)	gioco notturno	ATTIVITA' NON ANCORA INSERITA	ATTIVITA' NON ANCORA INSERITA	Giochi misti	ATTIVITA' NON ANCORA INSERITA	ATTIVITA' NON ANCORA INSERITA	ATTIVITA' NON ANCORA INSERITA
ORE 20:00	CENA						
ORE 21:00	FUOCO (durata 2 ore)						

esempio tabella reparto che sarà all'interno del sito web.

Ovviamente ci saranno anche le altre tabelle per quanto riguarda il branco e la compagnia.

```
<?php
//query per tabella con filtri
$sql_Ore9 = "SELECT Nome_attivita, Giorno_settimana FROM attivita WHERE Ore_attivita = '9:00:00' AND `id_brancaEs2` = 2";
$result_Ore9 = $conn->query($sql_Ore9);
```

```
</td>
<?php
$insertimento = false;
if ($result_Ore9->num_rows > 0) {
    while ($row = $result_Ore9->fetch_assoc()) {
        if ($row["Giorno_settimana"] == "lunedì") {
            echo $row["Nome_attivita"];
            $insertimento = true;
        }
    }
}
if($insertimento == false){
    echo "ATTIVITA' NON ANCORA INSERITA";
}

//reset la testina di lettura sul primo record in quanto precedentemente esaurito dal ciclo while
$result_Ore9->data_seek(0);
?>
```

Codice Visualizzazione utenti:

---

**Filtra Per:**  
**il numero degli iscritti è: 1**  
**in che branca vuoi inserire l'attività:**  
  
**SEDE APPARTENENZA:**  
  
**Nome:**  
  
**Cognome:**  
  
**email:**

```

<?php
session_start();

if ($_SERVER["REQUEST_METHOD"] == "POST")
{
    include "conessione.php";

    $sede_appartenenza = $_POST["sede_appartenenza"];
    $id_brancaEs = $_POST["id_brancaEs"];
    $nome = $_POST["nome"];
    $cognome = $_POST["cognome"];
    $email = $_POST["email"];

    echo "sede_appartenenza: ".$sede_appartenenza;
    echo "<br>";
    echo "id_brancaEs: ".$id_brancaEs;
    echo "<br>";
    echo "nome: ".$nome;
    echo "<br>";
    echo "cognome: ".$cognome;
    echo "<br>";
    echo "email: ".$email;
    echo "<br>";
    echo "<br>";
    echo "<br>";

    $sql = "SELECT * FROM `iscritti`";

    $id_brancaEs = $_POST["id_brancaEs"];
    $sede_appartenenza = $_POST["sede_appartenenza"];
    $nome = $_POST["nome"];
    $cognome = $_POST["cognome"];
    $params = [];

    if(!empty($id_brancaEs))
    {
        $params[] = "id_brancaEs = '". $id_brancaEs. "'";
    }

    if(!empty($sede_appartenenza))
    {
        $params[] = "sede_appartenenza = '". $sede_appartenenza. "'";
    }

    if(!empty($nome))
    {
        $params[] = "nome LIKE '%" . $nome. "%'";
    }
}

```

```

50     }
51
52     if(!empty($cognome))
53     {
54         $params[] = "cognome LIKE '%" . $cognome . "%'";
55     }
56
57
58
59     foreach($params as $i)
60     {
61         echo $i;
62     }
63
64     if (count($params) > 0)
65     {
66         $sql = "SELECT * FROM `iscritti` WHERE ";
67
68         foreach($params as $param)
69         {
70             $sql .= $param . " AND ";
71         }
72
73         $sql = substr($sql,0,-4);
74     }
75     $result = $conn->query($sql);
76
77     if ($result === TRUE) {
78         echo "New record created successfully";
79     } else {
80         echo "Error: " . $sql . "<br>" . $conn->error;
81     }
82
83     $rows = [];
84
85     while($row = $result->fetch_assoc())
86     {
87         $rows[] = $row;
88     }
89
90     $_SESSION["result"] = $rows;
91
92     header("Location: visualizzazioneUtenti.php");
93     exit();
94 }
95 ?>

```