

ISTITUTO TECNICO TECNOLOGICO

“Emanuele Morselli”

Via Pitagora, s.n.c. - 93012 GELA (Caltanissetta)

Cod. Fisc. 82002330858 – Cod. Minister. CLTF020005 - Tel. 0933/930997 – Fax 0933/930888

E-mail: cltf020005@istruzione.it – pec: cltf020005@pec.istruzione.it Sito Web: www.itismorselli.edu.it

Indirizzo: Informatica e Telecomunicazioni

A.S. 2019/2020

Classe 5[^]E

Prof. S. Di Dio Cafiso, Informatica

Prof.ssa G. Muscia, Sistemi e reti

Argomento concernente l’elaborato delle discipline

Informatica – Sistemi e reti

(art. 17, O.M. n.10 del 16 MAG 2020)

IMPORTANTE: L’elaborato dovrà essere consegnato entro il 13/06/2020 tramite il seguente indirizzo e-mail: cltf020005@istruzione.it. Chiedere conferma ricezione

Candidato: **Zuppardo Emanuele**

Considerazioni a carattere generale

Il candidato¹,

considerato il seguente tema d'esame, basandosi sugli argomenti affrontati nel corrente anno scolastico (vedi programma disciplinare qui allegato), avendo fatto un'opportuna ricerca bibliografica e/o sitografica sia sugli aspetti organizzativi dell'attività da svolgere sia sulle congetture di composizione del risultato finale, realizzi un elaborato rispondente al proprio argomento e ne implementi una porzione significativa del codice/software; infine, se ritenuto utile, produca anche un file di presentazione (PowerPoint o altro) dell'attività svolta.

Tema d'esame - Gruppo H -

Uno Studio Commercialista intende migliorare l'organizzazione della propria attività utilizzando una “web application” (e relativo db associato) che gli permetta di archiviare alcune informazioni riguardanti i propri clienti. In particolare vorrebbe ottenere come risultato: a) una catalogazione dei propri clienti per tipologia (privato, azienda, ente pubblico); b) per ogni cliente vorrebbe memorizzare il tipo di attività svolta (dichiarazione redditi, contabilità, statuto societario, fattura elettronica, ecc.); c) indicazioni sugli archivi digitali che ha prodotto la propria attività di consulente, in modo da poterli reperire/individuare all'occorrenza.

A tal fine il titolare dello Studio Commercialista si avvale di un ipotetico consulente il quale, dopo un'intervista immaginaria alla quale segue un'attenta analisi del problema dato, stilerà una raccolta di requisiti (elenco di caratteristiche e funzioni del

¹ L’elaborato assegnato a più candidati non deve essere inteso come lavoro di gruppo, bensì deve essere sviluppato da ogni studente in completa autonomia.

sistema) che comprenderà ciò che è richiesto al software che deve essere realizzato. Poi, il progettista che darà seguito al documento di analisi, dopo aver realizzato la specifica dei requisiti, descriverà i moduli e le interfacce del sistema. Infine, un team di sviluppatori implementerà i moduli e le interfacce secondo le specifiche imposte dal progettista. Il candidato, impersonando il ruolo delle figure professionali prima menzionate, prescindendo dagli aspetti formali della progettazione del software, descriva e produca tutto quello che le proprie conoscenze/competenze gli permettono di esprimere, non tralasciando i seguenti macroargomenti: modello concettuale, modello logico, normalizzazione, realizzazione fisica del DB, query e sottoquery, php e interazione con l'utente.

Lo studio, ha anche la necessità di realizzare la rete LAN interna, oltre ai diversi uffici dove lavorano i dipendenti, lo studio ha un locale per la segreteria ed una sala d'attesa che è servita attraverso una postazione fissa collegata ad Internet ed un collegamento WiFi a disposizione dei soli clienti dello studio. Si tenga in particolare considerazione la natura dei dati sensibili presenti negli archivi aziendali e le necessarie misure di sicurezza, nonché la gestione di tutti i servizi descritti nella prima parte dell'elaborato. Si chiede di realizzare, anche mediante rappresentazioni grafiche, l'infrastruttura tecnologica ed informatica necessaria a gestire il servizio nel suo complesso, in particolare si chiede di descrivere tipologia e architettura della rete con riferimento ai protocolli ISO/OSI utilizzati, tipologia di dispositivi necessari per realizzare la rete, protocolli e strumenti necessari a garantire la sicurezza dei dati sensibili trasmessi e la comunicazione web.

Soluzione elaborato Esame di Stato

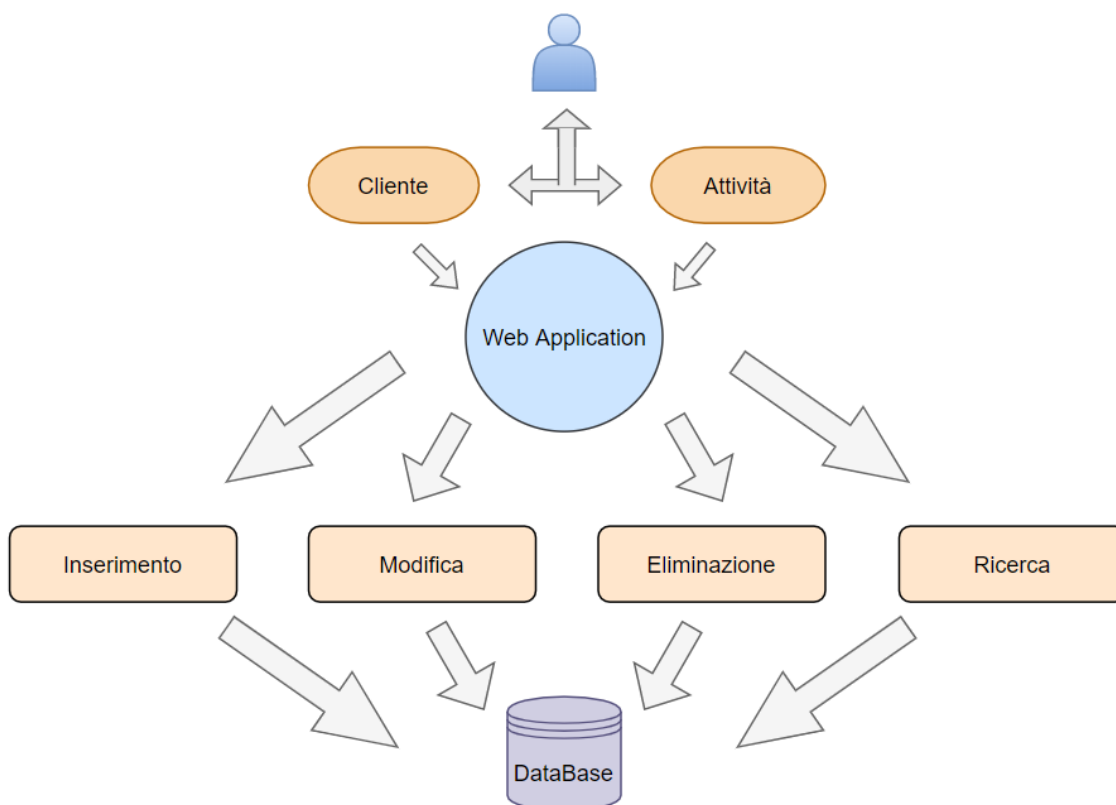
Informatica - Sistemi e Reti

Analisi del testo

Il testo sopra citato chiede un'attenta analisi del problema, prodotta grazie ad un'intervista immaginaria con il titolare dello Studio Commercialista.

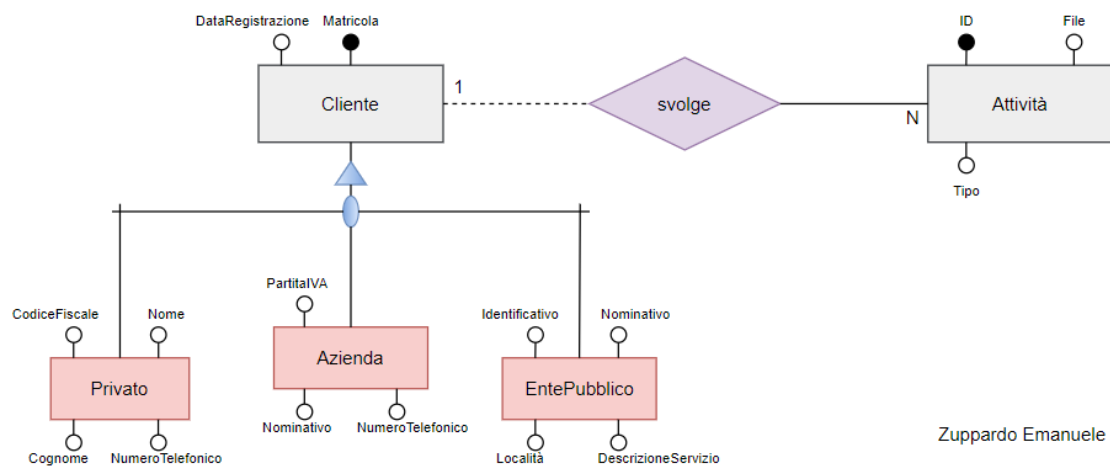
In questa ipotetica intervista supponiamo che lo studio commercialista chieda un sistema in grado di memorizzare i clienti e di suddividerli in categorie (privati, aziende ed enti pubblici) permettendo inoltre di memorizzare le attività svolte dai clienti stessi, magari sotto forma di file pdf, zip o docx da potere caricare, e da successivamente poter visualizzare. Sarà inoltre bene considerare che un'attività potrà essere di differenti tipologie, nonché dichiarazione redditi, contabilità, statuto societario, fattura elettronica. Qualsiasi cliente potrà inoltre essere modificato in modo da rendere il sistema scalabile, versatile per qualunque evenienza, come errori di inserimento dati o richieste di alterazioni di dati.

Il progettista, secondo il rapporto del consulente, elabora le seguenti ipotesi di sviluppo: il sistema sarà dotato di diverse interfacce, quali un'interfaccia di inserimento dati dei clienti per tipologia, un inserimento attività dei clienti specificando l'identificativo del cliente, un'interfaccia che permetta la visualizzazione di tutti i clienti sezionati per tipologia e infine un'interfaccia che permetta la ricerca per ogni singolo cliente, l'eliminazione e la modifica di quest'ultimo.



Modello concettuale

Il progettista quindi realizza un **modello concettuale E-R** (Entity-Relationship), modello che rappresenta i concetti richiesti dall'utente, in questo caso il titolare dello studio commercialista, in modo semplice e diretto.



Attraverso il modello concettuale il progettista, quindi, comunica al team di sviluppatori in che modo dovrà essere schematizzato il sistema di archiviazione dati.

Questo modello è tuttavia dotato di una particolarità: le entità Privato, Azienda ed EntePubblico sono entità figlie dell'entità Cliente, e quindi si parla di **gerarchia di generalizzazione o specializzazione**, utile per permettere una facile suddivisione dei clienti. In particolare si utilizza una **partizione di tipo verticale** in cui le entità figlie sono tra di loro disgiunte e la chiave primaria dell'entità padre diviene la chiave primaria ed esterna delle entità figlie, come si nota successivamente nel modello logico.

Modello logico

Inoltre il progettista fornisce agli sviluppatori il seguente modello logico:

Clienti (Matricola, DataRegistrazione);

Privati (MatricolaPrivato*, CodFiscale, Nome, Cognome, NumeroTelefonico);

Aziende (MatricolaAzienda*, PartitaIVA, NumeroTelefonico);

EntiPubblici (MatricolaEnte*, Identificativo, Nominativo, Località, DescrizioneServizio);

Attività (ID*, MatricolaCliente*, Tipo, File);

Normalizzazione

Secondo il progettista il modello ideato risulta già normalizzato per tutti i livelli della normalizzazione e quindi non è necessario trattare un'ulteriore normalizzazione. Difatti, grazie alla generalizzazione per partizione delle entità si riesce ad ottenere un modello già normalizzato.

Modello fisico

A questo punto, il team di sviluppatori, ha tutti i requisiti per realizzare il database vero e proprio e permettere, tramite pagine HTML e PHP, l'interazione con l'utente. In seguito la realizzazione del database in linguaggio SQL (Structured Query Language):

CREAZIONE TABELLA CLIENTI

```
1 CREATE TABLE clienti (  
2     Matricola int(11) NOT NULL AUTO_INCREMENT,  
3     Data_registrazione timestamp NULL DEFAULT current_timestamp(),  
4     PRIMARY KEY (Matricola)  
5 )
```

In questo caso abbiamo una chiave primaria, "Matricola", di tipo **AUTO_INCREMENT**, permettendo così agli elementi che vengono inseriti di essere diversificati in modo univoco. Inoltre il campo "Data_registrazione" è di tipo **timestamp**, quindi un tipo che riesce a contenere il formato "<AAAA-MM-GG> <hh:mm:ss>", e attraverso la funzione sql **current_timestamp()** si ricava la data e l'ora in cui viene creato l'elemento, in questo caso il cliente.

CREAZIONE TABELLA ATTIVITA'

```
1 CREATE TABLE attività (  
2     ID int(11) NOT NULL AUTO_INCREMENT,  
3     Tipo varchar(30) NOT NULL,  
4     File varchar(200) DEFAULT NULL,  
5     Matricola int(11) DEFAULT NULL,  
6     PRIMARY KEY (ID),  
7     FOREIGN KEY (Matricola) REFERENCES clienti (Matricola) ON DELETE SET NULL ON UPDATE CASCADE  
8 )
```

Anche in questa tabella la chiave primaria, "ID", è di tipo **AUTO_INCREMENT**. Il campo File di **varchar(200)** verrà poi utilizzato per contenere il nome del file caricato e consentire quindi il successivo "**Download**" del file, dopo la ricerca dell'attività (l'argomento verrà trattato più avanti). La chiave esterna Matricola, che fa riferimento alla chiave primaria Matricola in Clienti, è indispensabile per garantire un riferimento con la tabella clienti. In aggiunta vengono inserite le clausole **ON DELETE** e **ON UPDATE** per determinare il comportamento della chiave esterna nel momento in cui viene modificata la chiave di riferimento. È stata inserita la clausola **ON DELETE** impostata a **SET NULL** per questioni di semplicità nell'eliminazione dell'attività (anche questo verrà trattato successivamente).

CREAZIONE TABELLA PRIVATI

```
1 CREATE TABLE privati (  
2     CodFiscale varchar(16) NOT NULL,  
3     Cognome varchar(15) NOT NULL,  
4     Nome varchar(30) NOT NULL,  
5     NumeroTelefonico varchar(10) DEFAULT NULL,  
6     MatricolaPrivato int(11) NOT NULL,  
7     PRIMARY KEY (MatricolaPrivato),  
8     UNIQUE KEY CodFiscale (CodFiscale),  
9     FOREIGN KEY (MatricolaPrivato) REFERENCES clienti (Matricola) ON DELETE CASCADE ON UPDATE CASCADE  
10 )
```

La tabella figlia privati si basa sulla chiave primaria ed esterna MatricolaPrivato. Anche qui vengono utilizzate le clausole ON DELETE e ON UPDATE. Il campo "CodFiscale" non viene pensato come una chiave primaria, ma solo come un campo UNIQUE di privati.

CREAZIONE TABELLA AZIENDE

```
1 CREATE TABLE aziende (  
2     PartitaIVA varchar(11) NOT NULL,  
3     NomeAzienda varchar(30) DEFAULT NULL,  
4     MatricolaAzienda int(11) NOT NULL,  
5     PEC varchar(30) NOT NULL,  
6     PRIMARY KEY (MatricolaAzienda),  
7     UNIQUE KEY PartitaIVA (PartitaIVA),  
8     FOREIGN KEY (MatricolaAzienda) REFERENCES clienti (Matricola) ON DELETE CASCADE ON UPDATE CASCADE  
9 )
```

La tabella aziende possiede, come la tabella privati, la chiave primarie ed esterna MatricolaAzienda che sfrutta le clausole ON DELETE e ON UPDATE per rimanere coerente alla chiave primaria cui fa riferimento.

CREAZIONE TABELLA ENTIPUBBLICI

```
1 CREATE TABLE entipubblici (  
2     Identificativo int(11) NOT NULL,  
3     Nominativo varchar(30) NOT NULL,  
4     DescrizioneServizio varchar(80) DEFAULT NULL,  
5     Località varchar(10) DEFAULT NULL,  
6     MatricolaEnte int(11) NOT NULL,  
7     PRIMARY KEY (MatricolaEnte),  
8     UNIQUE KEY Identificativo (Identificativo),  
9     FOREIGN KEY (MatricolaEnte) REFERENCES clienti (Matricola) ON DELETE CASCADE ON UPDATE CASCADE  
10 )
```

Analogamente alle altre due entità figlie dell'entità clienti abbiamo la tabella entipubblici.

CREAZIONE VISTA ALLCLIENT

```
1 CREATE VIEW allclient AS  
2     SELECT *  
3     FROM clienti AS c LEFT OUTER JOIN privati AS p ON c.Matricola = p.MatricolaPrivato  
4         LEFT OUTER JOIN aziende AS a ON c.Matricola = a.MatricolaAzienda  
5         LEFT OUTER JOIN entipubblici AS e ON c.Matricola = e.MatricolaEnte
```

Grazie a questa vista si ottiene un'unica tabella di tutti i clienti, che risulterà molto utile nelle operazioni di ricerca del cliente tramite matricola.

Query e interazione con l'utente

Il team di sviluppatori inoltre fornisce allo studio commercialista una serie di file PHP e HTML per potere gestire tutte le operazioni che chiede il titolare, in modo tale da consentire un facile utilizzo per tutti gli utenti, anche i più inesperti.

N.B: Nei seguenti screenshot, per questioni di grandezza della schermata stessa molte stringhe sono state appositamente messe a capo, anche se comunque, così facendo, non si verifica nessun tipo di errore.

File di configurazione:

```
1 <?php
2     $host = "localhost";
3     $username = "root";
4     $password = "";
5     $nome_db = "studio_commercialista_db";
6     $conn = mysqli_connect($host, $username, $password);
7     $conn_db = mysqli_select_db($conn, $nome_db);
8 >?
```

Grazie a questo file php, e grazie alla funzione **include** "<file.php>", possiamo evitare le ripetizioni di codici identici nei vari file a venire. La funzione include, difatti, consiste nell'includere il contenuto presente in un file PHP all'interno di un altro file, come una sorta di copia e incolla di codice nella posizione in cui viene chiamata la funzione. Il file connectdb.php permette quindi grazie alle funzioni **mysqli_connect** e **mysqli_select_db** di connettere il database.

Inserimento privati:

```
1 <?php
2     include "connectdb.php";
3
4     //upload privato
5     if(isset($_POST["name"]) && isset($_POST["surname"]) && isset($_POST["cf"]) && isset($_POST["telefonnumber"])){
6         if((($_POST["name"] != "" && ($_POST["surname"] != "" && ($_POST["cf"] != "" && ($_POST["telefonnumber"] != "")))
7             $name = $_POST["name"];
8             $surname = $_POST["surname"];
9             $cf = $_POST["cf"];
10            $telefonnumber = $_POST["telefonnumber"];
11            $sql = "SELECT CodFiscale FROM privati WHERE CodFiscale = '$cf'";
12            $result = mysqli_query($conn, $sql);
13            $row = mysqli_num_rows($result);
14            if($row == 0){
15                $sql = "INSERT INTO clienti (Data_registrazione) VALUES (NOW())";
16                mysqli_query($conn, $sql);
17                $matricola = mysqli_insert_id($conn);
18                $sql = "INSERT INTO privati (Nome, Cognome, CodFiscale, NumeroTelefonico, MatricolaPrivato)
19                    VALUES ('$name', '$surname', '$cf', '$telefonnumber', '$matricola')";
20
21                if(mysqli_query($conn, $sql)){
22                    header("Location: insert.php?uploaded=".$matricola);
23                }
24                else{
25                    header("Location: insert.php?notuploaded");
26                }
27            }else{
28                header("Location: insert.php?errorprivatoid=".$cf);
29            }
30        }else{
31            header("Location: insert.php?fieldrequired");
32        }
33    }
```

Inserimento aziende:

```
35 //upload azienda
36 if(isset($_POST["name"]) && isset($_POST["iva"]) && isset($_POST["pec"])){
37     if(($_POST["name"] != "" && $_POST["iva"] != "" && ($_POST["pec"] != "")){
38         $name = $_POST["name"];
39         $iva = $_POST["iva"];
40         $pec = $_POST["pec"];
41         $sql = "SELECT PartitaIVA FROM aziende WHERE PartitaIVA = '$iva'";
42         $result = mysqli_query($conn, $sql);
43         $row = mysqli_num_rows($result);
44         if($row == 0){
45             $sql = "INSERT INTO clienti (Data_registrazione) VALUES (NOW())";
46             mysqli_query($conn, $sql);
47             $matricola = mysqli_insert_id($conn);
48             $sql = "INSERT INTO aziende (NomeAzienda, PartitaIVA, PEC, MatricolaAzienda)
49                 VALUES ('$name', '$iva', '$pec', '$matricola')";
50             if(mysqli_query($conn, $sql)){
51                 header("Location: insert.php?uploaded=".$matricola);
52             }
53             else{
54                 header("Location: insert.php?notuploaded");
55             }
56         }else{
57             header("Location: insert.php?erroraziendaid=".$iva);
58         }
59     }else{
60         header("Location: insert.php?fieldrequired");
61     }
62 }
```

Inserimento enti pubblici:

```
64 //upload ente pubblico
65 if(isset($_POST["name"]) && isset($_POST["id"]) && isset($_POST["locality"]) && isset($_POST["description"])){
66     if(($_POST["name"] != "" && $_POST["id"] != "" && $_POST["locality"] != "" && $_POST["description"] != ""){
67         $name = $_POST["name"];
68         $id = $_POST["id"];
69         $locality = $_POST["locality"];
70         $description = $_POST["description"];
71         $sql = "SELECT Identificativo FROM entipubblici WHERE Identificativo = '$id'";
72         $result = mysqli_query($conn, $sql);
73         $row = mysqli_num_rows($result);
74         if($row == 0){
75             $sql = "INSERT INTO clienti (Data_registrazione) VALUES (NOW())";
76             mysqli_query($conn, $sql);
77             $matricola = mysqli_insert_id($conn);
78             $sql = "INSERT INTO entipubblici (Nominativo, Identificativo, Località, DescrizioneServizio, MatricolaEnte)
79                 VALUES ('$name', '$id', '$locality', '$description', '$matricola')";
80             if(mysqli_query($conn, $sql)){
81                 header("Location: insert.php?uploaded=".$matricola);
82             }
83             else{
84                 header("Location: insert.php?notuploaded");
85             }
86         }else{
87             header("Location: insert.php?errorenteid=".$id);
88         }
89     }else{
90         header("Location: insert.php?fieldrequired");
91     }
92 }
```

Questi tre spezzoni di codice permettono all'utente di inserire clienti nel database. Tutto può avvenire grazie alle variabili superglobals `$_POST` che vengono passate da un altro file PHP (ad esempio nel caso di inserimento di privati vengono passati i valori name, surname, cf e telefonnumber da un altro file mediante form e textbox). Dopo una verifica della presenza dei valori, possibile con la funzione `isset`, e dopo un'ulteriore verifica per eventuali stringhe vuote (in tal caso la pagina stamperà: "Inserire tutti i campi!"), i valori vengono passati su delle variabili e si procede facendo una verifica di univocità per i vari campi UNIQUE delle tabelle (CodFiscale per privati, PartitaIVA per aziende e Identificativo per enti pubblici) eliminando così qualsiasi tipo di errore.

Attraverso la funzione `mysqli_num_rows(<risultato della query>)` riusciamo a contare le righe che una query ottiene e quindi, nel caso in cui il campo univoco fosse già presente nella tabella, la variabile `$row` avrà valore 1 e non entrerà mai nell'if. Avvenuta questa verifica inizia l'inserimento vero e proprio, notando però che in realtà avviene un doppio inserimento, prima sulla tabella clienti poi sulla tabella privati/aziende/entipubblici. La query per l'inserimento nella tabella clienti ("INSERT INTO clienti (Data_registrazione) VALUES Now()") va ad inserire solo il campo Data_registrazione, dato che il campo Matricola è un campo di tipo AUTO_INCREMENT e, quindi, non è necessario un inserimento manuale, utilizzando la funzione Now() che ci fornisce la data e l'ora attuale. Per ricavare quindi il valore del campo Matricola, per poterlo poi inserire nell'entità figlia, si utilizza la funzione `mysqli_insert_id(<risultato dell query>)`, che, scrivendo il valore che ritorna in una variabile, ci fornisce l'ultimo valore AUTO_INCREMENT inserito. Ottenuto il valore del campo Matricola non resta altro che eseguire la query di INSERT.

Inserimento attività PHP:

```
1  <?php
2      include "connectdb.php";
3      $matricola = $_POST["id"];
4      $type = $_POST["type"];
5      if (isset($_POST['save'])) {
6          //controllo se la matricola del cliente passato tramite POST è presente nella tabella clienti
7          $sql = "SELECT Matricola FROM clienti WHERE Matricola = '$matricola'";
8          $result = mysqli_query($conn, $sql);
9          $row = mysqli_num_rows($result);
10         if($row == 0){
11             header("Location: insert.php?id=" . $matricola);
12         }
13         else{
14             //creazione dell'attività per permettere di ricavare l'id attività da poi aggiungere ad inizio nome
15             $sql = "INSERT INTO attività (Matricola) VALUES (\'$matricola\')";
16             mysqli_query($conn, $sql);
17             $id = mysqli_insert_id($conn);
18             $filename = $_FILES['myfile']['name'];
19             $filename = "$id-" . $filename;
20             //destinazione in cui il file dovrà andare a finire
21             $destination = 'C:\xampp\htdocs\uploads\' . $filename;
22             //ricavare l'estensione del file per un successivo controllo
23             $extension = pathinfo($filename, PATHINFO_EXTENSION);
24             //ricavare il file fisico allocato nella variabile superglobale FILES
25             $file = $_FILES['myfile']['tmp_name'];
26             $size = $_FILES['myfile']['size'];
27             if (!in_array($extension, ['zip', 'pdf', 'docx'])){
28                 $sql = "DELETE FROM attività WHERE ID = $id";
29                 mysqli_query($conn, $sql);
30                 header("Location: insert.php?extension");
31             }else{
32                 if ($_FILES['myfile']['size'] > 10000000) { //limite dimensione file di 10mb
33                     $sql = "DELETE FROM attività WHERE ID = $id";
34                     mysqli_query($conn, $sql);
35                     header("Location: insert.php?larger");
36                 }else{
37                     //mettere il file nella sua destinazione
38                     if (move_uploaded_file($file, $destination)) {
39                         $sql = "UPDATE attività SET File = '$filename', Tipo = '$type' WHERE ID = '$id'";
40                         if (mysqli_query($conn, $sql)) {
41                             //File caricato correttamente
42                             header("Location: insert.php?ok");
43                         }
44                     }else{
45                         //File non caricato
46                         header("Location: insert.php?no");
47                     }
48                 }
49             }
50         }
51     }
52  >?>
```

Inserimento attività HTML:

```
147 <form action="uploadfile.php" method="post" enctype="multipart/form-data" >  
148     Inserisci ID cliente: <br>  
149     <input type="text" name="id" required> <br> <br>  
150     Inserisci un file di estensione <br> .pdf .zip .docx <br> <br>  
151     <input type="file" name="myfile" required> <br>  
152     <p id="send2">  
153         <button id="send2" name="save">  </button>  
154     </p>  
155 </form>
```

Nel file PHP dell'inserimento attività si inizia con l'include del file di configurazione al database, e successivamente si va a verificare se il bottone con name="save" viene passato tramite post così da entrare nell'inserimento. A seguire viene controllato il campo Matricola cliente, verificando quindi se il cliente esiste nella tabella clienti tramite la query di riga 7. Se questa query ritorna una riga allora si procede inserendo alla tabella attività una tupla, utile solo a ricavare l'id da poi aggiungere davanti al nome dell'attività (ad esempio il file "FatturaElettronica56.pdf" diverrà "<ID>-FatturaElettronica56.pdf" quindi un ipotetico "5-FatturaElettronica56.pdf"). Questa scelta degli sviluppatori è dovuta al fatto che nel momento in cui un file viene inserito nella cartella del server è possibile che si presenti un altro file con lo stesso nome, andando a sovrascriverlo. In questo modo si va ad identificare univocamente ogni attività, grazie all'ID posto davanti. Viene quindi ricavata la destinazione in cui andrà il file, la variabile \$destination (in questo caso è una destinazione temporanea utilizzata per test che dovrà poi essere cambiata a seconda di dove vorranno archiviarsi le attività) e l'estensione, con la variabile \$extension, che verrà utilizzata nella riga 27, del codice PHP, e che, attraverso la funzione **in_array(<array>, <valore da cercare>)**, garantirà l'inserimento di soli file con estensioni ".pdf", ".zip" e ".docx". Da considerare è la variabile superglobale \$_FILES, che viene molto usata in questo spezzone di codice e che permette di ricavare dati importanti dei file caricati durante la sessione come ad esempio "\$_FILES['myfile']['name']" permette di ricavare il nome del file caricato o \$_FILES['myfile']['tmp_name'] che permette di ottenere il file vero e proprio. La variabile superglobale \$_FILES quindi immagazzina nei suoi campi tutti i dati riguardanti il/i file precedentemente caricati. Viene anche verificata la dimensione del file caricato e infine viene posizionato il file nella destinazione tramite la funzione **move_uploaded_file(<file><destinazione>)**.

Infine vengono inseriti tutti i campi precedentemente non inseriti.

Ricerca clienti per Matricola:

```
1 <?php
2 include "connectdb.php";
3 session_start();
4 $searchbar = $_POST["searchbar"];
5 if(isset($_POST['searchclient'])){
6     $sql = "SELECT Matricola, Nome, Cognome, CodFiscale, NumeroTelefonico, Data_registrazione
7         FROM allclient WHERE MatricolaPrivato IS NOT NULL";
8     $result = mysqli_query($conn, $sql);
9     $row = mysqli_num_rows($result);
10    if($row >= 1){
11        while($row = mysqli_fetch_assoc($result)){
12            if($row["Matricola"] == $searchbar){
13                $_SESSION["Matricola"] = $row["Matricola"];
14                $_SESSION["Nome"] = $row["Nome"];
15                $_SESSION["Cognome"] = $row["Cognome"];
16                $_SESSION["CodFiscale"] = $row["CodFiscale"];
17                $_SESSION["NumeroTelefonico"] = $row["NumeroTelefonico"];
18                $_SESSION["Data_registrazione"] = $row["Data_registrazione"];
19                header("Location: search.php?values=privato");
20                return;
21            }
22        }
23    }
24
25    $sql = "SELECT Matricola, NomeAzienda, PartitaIVA, PEC, Data_registrazione
26        FROM allclient WHERE MatricolaAzienda IS NOT NULL";
27    $result = mysqli_query($conn, $sql);
28    $row = mysqli_num_rows($result);
29    if($row >= 1){
30        while($row = mysqli_fetch_assoc($result)){
31            if($row["Matricola"] == $searchbar){
32                $_SESSION["Matricola"] = $row["Matricola"];
33                $_SESSION["NomeAzienda"] = $row["NomeAzienda"];
34                $_SESSION["PartitaIVA"] = $row["PartitaIVA"];
35                $_SESSION["PEC"] = $row["PEC"];
36                $_SESSION["Data_registrazione"] = $row["Data_registrazione"];
37                header("Location: search.php?values=azienda");
38                return;
39            }
40        }
41    }
42
43    $sql = "SELECT Matricola, Nominativo, Identificativo, Località, DescrizioneServizio,
44        Data_registrazione FROM allclient WHERE MatricolaEnte IS NOT NULL";
45    $result = mysqli_query($conn, $sql);
46    while($row = mysqli_fetch_assoc($result)){
47        if($row["Matricola"] == $searchbar){
48            $_SESSION["Matricola"] = $row["Matricola"];
49            $_SESSION["Nominativo"] = $row["Nominativo"];
50            $_SESSION["Identificativo"] = $row["Identificativo"];
51            $_SESSION["Località"] = $row["Località"];
52            $_SESSION["DescrizioneServizio"] = $row["DescrizioneServizio"];
53            $_SESSION["Data_registrazione"] = $row["Data_registrazione"];
54            header("Location: search.php?values=ente");
55            return;
56        }
57    }
58    header("Location: search.php?values=iderrorclient");
59    return;
60 }
```

Attraverso questa pagina PHP è possibile ricercare il cliente per Matricola. A differenza delle altre pagine PHP qui si inizializza la sessione nel momento in cui viene aperto il file, dato che viene molto usata la variabile superglobale `$_SESSION` per il passaggio dei valori cercati. Attraverso il costrutto `if` a riga 5 è possibile verificare se il valore 'searchclient', nonché il bottone di invio ricerca, è presente nella variabile `$_POST`. Successivamente tramite una query sulla vista `allclient`, che contiene tutti i clienti di tutte le tipologie, si escludono quei valori che sono uguali a `NULL` nella colonna della chiave primaria della singola entità figlia (ad esempio nel caso di privati si vanno ad escludere i valori che valgono `NULL` nella colonna `MatricolaPrivato`, lasciando quindi solo i campi in cui sono presenti i valori della tabella privati, come avviene nella riga 6), nonché quei valori che non

appartengono alle altre tipologie di clienti. Così facendo si otterrà una tabella che contiene non solo tutti i campi della tabella privati, ma anche il campo Data_registrazione che invece è presente nella tabella clienti. Se si ottiene una tabella con 1 o più righe vale a dire che sono presenti privati. Infine attraverso la funzione **mysqli_fetch_assoc** (<risultato della query>), che recupera i campi della tabella per inserirli in una variabile trasformata in array associativo, riusciamo a passare tutti i valori del cliente con Matricola uguale al campo che abbiamo inserito nella precedente pagina sull'apposito textbox per la ricerca (\$searchbar).

```
45 if($_GET["values"] == "privato") {
46     echo "<table>";
47     echo "<tr>";
48     echo "<td>";
49     echo "<form action=\"deleteclientactivities.php\" method='post'>";
50     echo "<button id='deletebutton' name='deleteclient'";
51         onClick=\"return confirm('Vuoi davvero eliminare questo cliente?";
52         Così facendo eliminerai anche le attività relative ad esso.')\">";
53         <img src='delete_button.png' width='30' height='30' /> </button>";
54     echo "</form>";
55     echo "</td>";
56     echo "<th> Matricola </th>";
57     echo "<th> Nome </th>";
58     echo "<th> Cognome </th>";
59     echo "<th> Codice Fiscale </th>";
60     echo "<th> Numero Telefonico </th>";
61     echo "<th> Data registrazione </th>";
62     echo "<th> Tipo </th>";
63     echo "<th> Attività </th>";
64     echo "</tr>";
65     echo "<tr>";
66     echo "<td>";
67     echo "<form action=\"editclient.php\" method='post'>";
68     echo "<button id='editbutton' name='editprivato'";
69         <img src='edit_button.png' width='30' height='30' /> </button>";
70     echo "</form>";
71     echo "</td>";
72     echo "<td>".$_SESSION['Matricola']. "</td>";
73     echo "<td>".$_SESSION['Nome']. "</td>";
74     echo "<td>".$_SESSION['Cognome']. "</td>";
75     echo "<td>".$_SESSION['CodFiscale']. "</td>";
76     echo "<td>".$_SESSION['NumeroTelefonico']. "</td>";
77     echo "<td>".$_SESSION['Data_registrazione']. "</td>";
78     echo "<td> Privato </td>";
79     echo "<td>";
80     echo "<a href=\"search.php?idprivato=".$_SESSION['Matricola']. "\"> Clicca qui </a>";
81     echo "</td>";
82     echo "</tr>";
83 }
```



Tramite questo codice è quindi possibile ottenere come output una tabella con tutti i valori del cliente richiesto.

Ricerca cliente per ID



Ricerca attività per ID



	Matricola	Nome	Cognome	Codice Fiscale	Numero Telefonico	Data registrazione	Tipo	Attività
	1	Emanuele	Zuppardo	ZPPMNL01T04C351T	3484374705	2020-06-10 20:58:35	Privato	Clicca qui

Inoltre, sono state inseriti due bottoni, con relative immagini, per l'eliminazione e la modifica del cliente ricercato, e una colonna che permette di ottenere tutte le attività svolte da quel cliente.

Eliminazione cliente:

```
1 <?php
2 include "connectdb.php";
3 session_start();
4 $matricola = $_SESSION["Matricola"];
5 $id = $_SESSION["ID"];
6
7 if(isset($_POST["deleteclient"])){
8     $sql = "DELETE FROM clienti WHERE Matricola = $matricola";
9     if(mysqli_query($conn, $sql)){
10         $sql = "SELECT File FROM attività WHERE Matricola IS NULL";
11         $result = mysqli_query($conn, $sql);
12         $row = mysqli_num_rows($result);
13         if($row >= 1){
14             while($row = mysqli_fetch_assoc($result)){
15                 $filePath = "C:\\xampp\\htdocs\\uploads\\" . $row["File"];
16                 if(unlink($filePath)){
17                     $sql = "DELETE FROM attività WHERE Matricola IS NULL";
18                     mysqli_query($conn, $sql);
19                     header("Location: search.php?values=deleted");
20                 }
21             }
22             header("Location: search.php?values=clientdeleted");
23         }else{
24             header("Location: search.php?values=errordelate");
25         }
26     }
```

Con questo codice è possibile eliminare il cliente cliccando sul pulsante di elimina sopra menzionato. Questo codice, inoltre, elimina anche tutte le eventuali attività del cliente. Difatti, nel momento in cui viene eliminato un cliente (privato, aziende, enti pubblici) dalla tabella clienti, grazie alla clausola ON DELETE impostata a SET NULL nella chiave esterna Matricola sull'entità attività, il campo Matricola di attività viene impostato a NULL consentendo quindi alla query di riga 10 di ottenere come risultato una tabella con la/le attività relative al cliente. Inoltre, con la funzione **unlink(<percorso file>)**, come si nota nella riga 16, si va ad eliminare il relativo file conservato nella cartella prestabilita. Infine, con la query a riga 17, si eliminano le attività del cliente dal database.

Modifica cliente privato:

```
61 if(isset($_POST["saveprivato"])){
62     $name = $_POST["name"];
63     $surname = $_POST["surname"];
64     $cf = $_POST["cf"];
65     $telefonnumber = $_POST["telefonnumber"];
66     $sql = "SELECT CodFiscale FROM privati WHERE CodFiscale = '$cf' AND MatricolaPrivato != $matricola";
67     $result = mysqli_query($conn, $sql);
68     $row = mysqli_num_rows($result);
69     if($row == 0){
70         $sql = "UPDATE privati SET Nome = '$name', Cognome = '$surname',
71             CodFiscale = '$cf', NumeroTelefonico = '$telefonnumber'
72             WHERE MatricolaPrivato = $matricola";
73         if($result = mysqli_query($conn, $sql)){
74             header("Location: search.php?updated=ok");
75         }else{
76             header("Location: search.php?updated=no");
77         }
78     }else{
79         header("Location: search.php?updated=no");
80     }
81 }
```

Modifica cliente azienda:

```
107 if(isset($_POST["saveazienda"])){
108     $name = $_POST["name"];
109     $iva = $_POST["iva"];
110     $pec = $_POST["pec"];
111     $sql = "SELECT PartitaIVA FROM aziende WHERE PartitaIVA = '$iva' AND MatricolaAzienda != $matricola";
112     $result = mysqli_query($conn, $sql);
113     $row = mysqli_num_rows($result);
114     if($row == 0){
115         $sql = "UPDATE aziende SET NomeAzienda = '$name', PartitaIVA = '$iva',
116             PEC = '$pec' WHERE MatricolaAzienda = $matricola";
117         if($result = mysqli_query($conn, $sql)){
118             header("Location: search.php?updated=ok");
119         }else{
120             header("Location: search.php?updated=no");
121         }
122     }else{
123         header("Location: search.php?updated=no");
124     }
125 }
```

Modifica cliente ente pubblico:

```
155 if(isset($_POST["saveente"])){
156     $name = $_POST["name"];
157     $id = $_POST["id"];
158     $locality = $_POST["locality"];
159     $description = $_POST["description"];
160     $sql = "SELECT Identificativo FROM entipubblici WHERE Identificativo = $id AND MatricolaEnte != $matricola";
161     $result = mysqli_query($conn, $sql);
162     $row = mysqli_num_rows($result);
163     if($row == 0){
164         $sql = "UPDATE entipubblici SET Nominativo = '$name', Identificativo = '$id',
165             Localita = '$locality', DescrizioneServizio = '$description'
166             WHERE MatricolaEnte = $matricola";
167         if($result = mysqli_query($conn, $sql)){
168             header("Location: search.php?updated=ok");
169         }else{
170             header("Location: search.php?updated=no");
171         }
172     }else{
173         header("Location: search.php?updated=no");
174     }
175 }
```

Le pagine sopra mostrate consentono di modificare i clienti, precedentemente cercati per ID, cliccando sul bottone di modifica sopra menzionato. Oltre al passaggio dei valori del cliente, che appariranno già inseriti sui textbox, viene prima verificato se il nuovo campo UNIQUE (su privati CodFiscale, su aziende PartitaIVA, su entipubblici Identificativo) è uguale ad un campo UNIQUE di un altro cliente dello stesso tipo. Infine viene fatta l'effettiva modifica dei campi inserendo quelli passati.

Ricerca attività per ID:

```
62 if(isset($_POST["searchactivities"])){
63     $sql = "SELECT * FROM attività WHERE ID = $searchbar";
64     $result = mysqli_query($conn, $sql);
65     $row = mysqli_num_rows($result);
66     if($row == 1){
67         while($row = mysqli_fetch_assoc($result)){
68             if($row["ID"] == $searchbar){
69                 $_SESSION["ID"] = $row["ID"];
70                 $_SESSION["Tipo"] = $row["Tipo"];
71                 $_SESSION["File"] = $row["File"];
72                 $_SESSION["Matricola"] = $row["Matricola"];
73                 header("Location: search.php?values=activities");
74                 return;
75             }
76         }
77     }else{
78         header("Location: search.php?values=iderroractivities");
79     }
80 }
```


Output ricerca attività per ID:

```
165 if($_GET["values"] == "activities") {
166     echo "<table>";
167     echo "<tr>";
168     echo "<td>";
169     echo "<form action='deleteclientactivities.php' method='post'>";
170     echo "<button id='deletebutton' name='deleteactivities'";
171     echo "onClick='return confirm(\"Vuoi davvero eliminare questa attività?\")\"";
172     echo "<img src='delete_button.png' width='30' height='30' /> </button>";
173     echo "</form>";
174     echo "</td>";
175     echo "<th> ID </th>";
176     echo "<th> Tipo </th>";
177     echo "<th> File </th>";
178     echo "<th> ID Cliente </th>";
179     echo "</tr>";
180     echo "<tr>";
181     echo "<td> </td>";
182     echo "<td>".$_SESSION['ID']. "</td>";
183     echo "<td>".$_SESSION['Tipo']. "</td>";
184     echo "<td>";
185     echo "<a href='\"/uploads/\"".$_SESSION['File']."'> download> Download File </a>";
186     echo "</td>";
187     echo "<td>".$_SESSION['Matricola']. "</td>";
188     echo "</tr>";
189 }
```

Con gli spezzoni di codice sopra mostrati tramite l'utilizzo della funzione `mysqli_fetch_assoc()` si ricavano, come avviene anche per la ricerca clienti, tutti i valori dell'attività che viene ricercata. Il tutto viene passato tramite la variabile `$_SESSION` che si occuperà di passare i valori da una pagina all'altra. Infine, tramite il secondo spezzone di codice si ottiene una tabella contenente i valori allocati sulla variabile `$_SESSION`, mostrando anche una stringa "Download File" (riga 185), con collegamento al percorso del file dell'attività, che permette di scaricare il file stesso come si nota nella schermata successiva.

Ricerca cliente per ID

Ricerca attività per ID

	ID	Tipo	File	ID Cliente
	2	Dichiarazione_Redditi	Download File	1

Come anche per i clienti, anche le attività vengono eliminate attraverso il medesimo pulsante tramite il seguente codice, eliminando l'attività non solo dal database ma anche dalla cartella in cui è salvata.

```
28 if(isset($_POST["deleteactivities"])){
29     $sql = "SELECT File FROM attività WHERE ID = $id";
30     $result = mysqli_query($conn, $sql);
31     $row = mysqli_fetch_assoc($result);
32     $filePath = "C:\\xampp\\htdocs\\uploads\\" . $row["File"];
33     if(unlink($filePath)){
34         $sql = "DELETE FROM attività WHERE ID = $id";
35         if($result = mysqli_query($conn, $sql)){
36             header("Location: search.php?values=activitiesdeleted");
37         }else{
38             header("Location: search.php?values=errordelete");
39         }
40     }
41 }
42 }
```

Visualizzazione clienti privati:

```
33 <?php
34 include "connectdb.php";
35 if(isset($_POST["send"])){
36     $sql = "SELECT * FROM aziende";
37     $result = mysqli_query($conn, $sql);
38     $row = mysqli_num_rows($result);
39     if($row >= 1) {
40         echo "<table>";
41         echo "<tr>";
42         echo "<th> Matricola </th>";
43         echo "<th> Nome Azienda </th>";
44         echo "<th> Partita IVA </th>";
45         echo "<th> PEC </th>";
46         echo "</tr>";
47         while($row = mysqli_fetch_assoc($result)){
48             echo "<tr>";
49             echo "<td>".$row['MatricolaAzienda']. "</td>";
50             echo "<td>".$row['NomeAzienda']. "</td>";
51             echo "<td>".$row['PartitaIVA']. "</td>";
52             echo "<td>".$row['PEC']. "</td>";
53             echo "</tr>";
54         }
55         echo "</table>";
56     }
57 }
58 ?>
```


Visualizzazione clienti aziende:

```
33 <?php
34     include "connectdb.php";
35     if(isset($_POST["send"])){
36         $sql = "SELECT * FROM privati";
37         $result = mysqli_query($conn, $sql);
38         $row = mysqli_num_rows($result);
39         if($row >= 1) {
40             echo "<table>";
41             echo "<tr>";
42                 echo "<th> Matricola </th>";
43                 echo "<th> Nome </th>";
44                 echo "<th> Cognome </th>";
45                 echo "<th> Codice Fiscale </th>";
46                 echo "<th> Numero Telefonico </th>";
47             echo "</tr>";
48             while($row = mysqli_fetch_assoc($result)){
49                 echo "<tr>";
50                     echo "<td>".$row['MatricolaPrivato']. "</td>";
51                     echo "<td>".$row['Nome']. "</td>";
52                     echo "<td>".$row['Cognome']. "</td>";
53                     echo "<td>".$row['CodFiscale']. "</td>";
54                     echo "<td>".$row['NumeroTelefonico']. "</td>";
55                 echo "</tr>";
56             }
57             echo "</table>";
58         }
59     }
60     ?>
```

Visualizzazione clienti enti pubblici:

```
33 <?php
34     include "connectdb.php";
35     if(isset($_POST["send"])){
36         $sql = "SELECT * FROM entipubblici";
37         $result = mysqli_query($conn, $sql);
38         $row = mysqli_num_rows($result);
39         if($row >= 1) {
40             echo "<table>";
41             echo "<tr>";
42                 echo "<th> Matricola </th>";
43                 echo "<th> Nominativo </th>";
44                 echo "<th> Identificativo </th>";
45                 echo "<th> Località </th>";
46                 echo "<th> Descrizione Servizio </th>";
47             echo "</tr>";
48             while($row = mysqli_fetch_assoc($result)){
49                 echo "<tr>";
50                     echo "<td>".$row['MatricolaEnte']. "</td>";
51                     echo "<td>".$row['Nominativo']. "</td>";
52                     echo "<td>".$row['Identificativo']. "</td>";
53                     echo "<td>".$row['Località']. "</td>";
54                     echo "<td>".$row['DescrizioneServizio']. "</td>";
55                 echo "</tr>";
56             }
57             echo "</table>";
58         }
59     }
60     ?>
```

Mediante questi codici è possibile mostrare in una tabella tutti i clienti divisi per tipologia. Anche qui viene utilizzata la funzione `mysqli_fetch_assoc()`.

Output visualizzazione privati:

☒ Privati ☐ Aziende ☐ Enti pubblici

Matricola	Nome	Cognome	Codice Fiscale	Numero Telefonico
1	Emanuele	Zuppardo	ZPPMNL01T04C351T	3484374705
4	Giuseppe Massimo	Corrao	CRRGPP04C19H194H	3468720595
7	Tiziano Giuseppe	Nicosia	NCSBDS01S09D960W	3425288454
8	Antonio Giulio	Nuzzi	NZZNNG01C16D960F	3471914720
12	Luca	Bianco	BNCLCU80A01H501Q	3256978415

Output visualizzazione aziende:

☐ Privati ☒ Aziende ☐ Enti pubblici

Matricola	Nome Azienda	Partita IVA	PEC
2	PcPerformance S.r.l.	01548963254	pcperformancesrl@gmail.com
5	Amarù S.r.l.	41025698754	amarusrl@gmail.com

Output visualizzazione enti pubblici:

☐ Privati ☐ Aziende ☒ Enti pubblici

Matricola	Nominativo	Identificativo	Località	Descrizione Servizio
3	Polizia Statale	205	Gela	Stabilisce la sicurezza cittadina
6	Vigili del fuoco	545	Gela	Stabilisce la sicurezza cittadina
9	Istat	4562	Italia	Si occupa di censimenti e indagini

N.B: i dati inseriti nelle tabelle sono dati fittizi.

Realizzazione infrastruttura di rete

Il titolare dello studio commercialista chiede di realizzare una LAN interna dotata di uffici dove lavorano i dipendenti, un locale per la segreteria e una sala d'attesa con la possibilità di poter far connettere ad internet i clienti.

Si pensa quindi ad una rete di architettura client-server dove il server, che possiede tutte le pagine sopra mostrate (PHP e HTML), offre servizi ai client, garantendo il funzionamento del sistema. Inoltre è da considerare che il server dovrà possedere un indirizzo IP fisso (IP statico) per poter essere raggiunto da qualsiasi host connesso alla LAN. I client invece saranno dotati di indirizzi IP dinamici forniti mediante DHCP (Dynamic Host Configuration Protocol), nonché un protocollo di configurazione IP dinamica. Mediante il protocollo DHCP, fornito da un semplice router, riusciremo quindi a stabilire ad ogni host che si connette alla rete un IP dinamico. Per fare ciò è necessario quindi escludere quegli indirizzi che necessitano di essere statici, e che quindi non verranno utilizzati durante l'assegnazione dinamica degli indirizzi IP.

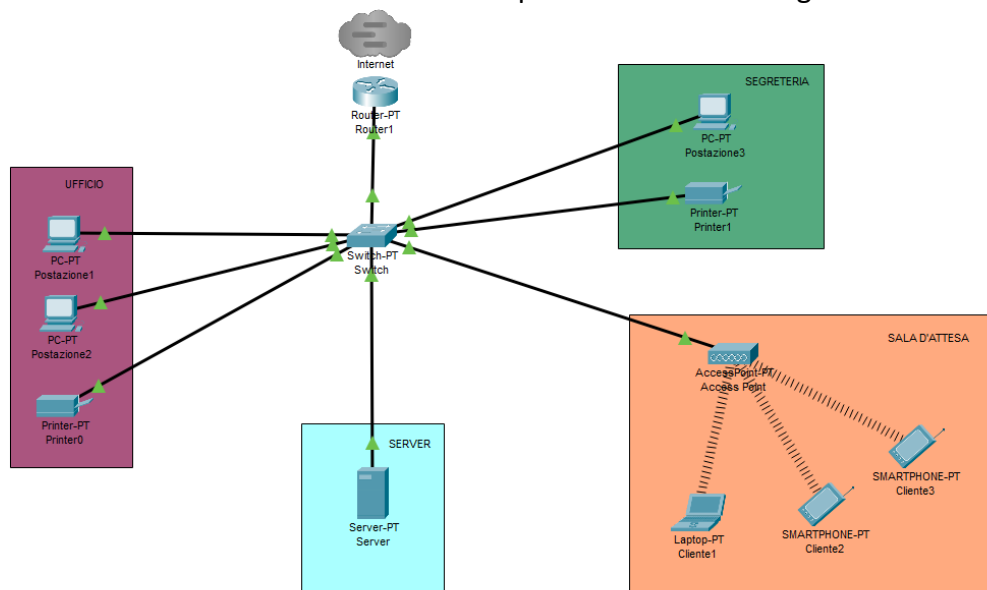
Inoltre, attraverso protocolli come HTTP ed SSH consentiamo agli host di visualizzare pagine Web in modo sicuro.

A questo punto, definita l'architettura di rete e i principali protocolli utilizzati, bisognerà garantire sicurezza al sistema, attraverso, magari, servizi Firewall. Un metodo è quello di implementare sul server meccanismi che siano in grado di bloccare alcune risorse, come ad esempio una ACL (Access Control List), nonché un meccanismo che potremo utilizzare per bloccare, a tutti gli host che cercano di comunicare con il server, tutte le porte eccetto quelle necessarie al funzionamento delle pagine web.

	Action	Protocol	Remote IP	Remote Wild Card	Remote Port	Local Port
1	Allow	TCP	0.0.0.0	255.255.255.255	any	80
2	Allow	TCP	0.0.0.0	255.255.255.255	any	443
3	Deny	TCP	0.0.0.0	255.255.255.255	any	any

Come si nota in questa immagine viene, quindi, consentito l'accesso alle sole porte 80 e 443, rispettivamente HTTP e SSH, bloccando invece qualsiasi altra porta del server e garantendo protezione dai malintenzionati.

Una possibile soluzione di infrastruttura di rete potrebbe essere la seguente:



In questa infrastruttura di rete LAN sono presenti, come si nota, diversi dispositivi connessi ad uno switch, a sua volta connesso ad un router, che come detto prima consente di assegnare gli indirizzi IP dinamici. L'unico dispositivo con indirizzo IP statico è il server centrale. Inoltre, per garantire alla sala d'attesa una connessione internet per i soli clienti si procede configurando un Access Point e impostando una password, alla rete wireless, che verrà poi fornita da un dipendente per tutti i nuovi clienti.