

I.T.I.S. Leonardo Da Vinci
Toscana, Firenze

ESAME DI STATO
2015-2016

**DIGITALIZZAZIONE
INFORMATIZZAZIONE
DI
UN QUESTIONARIO
“ALTERNANZA SCUOLA-LAVORO”**

Classe V INFORMATICA
John Ralph Umandal

Indice

1. Premessa e Introduzione
 - 1.1 Premessa
 - 1.2 Introduzione
2. Progettazione del database
 - 2.1 Analisi
 - 2.2 Modello E/R
 - 2.3 Mapping
 - 2.4 Query
3. Sviluppo delle pagine dinamiche
 - 3.1 Front-end
 - 3.2 Back-end
4. Pubblicazione

1. 1 Premessa

Ho deciso di portare la tesina che ha come tema principale l'informatizzazione, il cui ambito, a mio avviso, è parte essenziale di ogni applicazione informatica fino ad oggi.

La tesina contiene funzioni che restituiscono analisi e statistiche che non sarò in grado di documentare né di presentare appieno per mancanza di persone disponibili per l'utilizzo di tale applicativi ai fini della tesina stessa.

Lo scopo delle parole scritte in corsivo e sottolineate è quello di esporre i nessi logici della tesina ed evidenziano in maniera particolare gli argomenti trattati nelle varie materie

Poichè il progetto ruota attorno allo sviluppo e alla realizzazione di una web-app, ritengo che sia appropriato, se non quasi doveroso, dare una definizione ed una breve introduzione sulla web-app.

The term "Web-Apps" is used to define softwares running on internet. Thanks to the new technologies, we have now the opportunity to both move softwares and datas out from PC into the cloud, which server is connected. One of the big advantages about moving data into the cloud is that there's no need of installations anymore and you can access to programs' functions through the browser. Everything is stored in the cloud and all datas are computed on the server.

1.2 Introduzione

Grazie ad una piattaforma web, accessibile tramite un browser per mezzo di internet, è possibile sviluppare un sistema di informatizzazione volto a digitalizzare un questionario generale per poi salvare le risposte degli utenti.

Il progetto della mia tesina tratta lo sviluppo di pagine web “amichevoli” e dinamiche sia per l’utente grazie all’utilizzo del linguaggio di scripting JavaScript a cui si appoggerà la libreria jQuery, sia per quanto riguarda il server, grazie all’utilizzo del linguaggio PHP per la manipolazione delle informazioni ricevute o da inviare ad una base di dati, cioè ad un database su cui avverrà il salvataggio dei dati in modo permanente.

Detto progetto ruota intorno al documento qui in allegato ovvero un ipotetico questionario di autovalutazione sul progetto “Alternanza Scuola Lavoro”. L’applicazione web da me proposta utilizza una macchina virtuale, cioè un sistema operativo emulato all’interno della macchina, affinché possa essere eseguita su un altro computer indifferente dalle specifiche tecniche hardware.

La macchina virtuale in questo progetto funge da web server, la quale è a sua volta associata ad un’opportuna rete interna che darà accesso al servizio web senza compromettere l’intera infrastruttura.

L’utente che usufruirà del servizio web di autovalutazione dovrà registrarsi tramite un codice univoco che dovrà essere inserito al momento della richiesta di registrazione, generato dalla segreteria scolastica o da un ente di livello gerarchico superiore.

L’applicativo oggetto della tesina provvederà successivamente alla generazione di dati statistici in base alle risposte date nel questionario.

In questo progetto vengono quindi rappresentati gli elementi e le caratteristiche essenziali per lo sviluppo di una web-app.

Con questo progetto, intendo presentare un insieme di tecnologie moderne, e come lavorano insieme per lo sviluppo di una Web Application.

2. Progettazione del database

Nucleo del progetto è il database, attraverso il quale, grazie ad un DBMS installato nel server, verranno gestiti e salvati i dati elaborati collegandoli fra di loro. Poiché all'interno del database vi è una grande quantità di informazioni da gestire, il DBMS è una parte importante per poter evitare inconsistenze delle informazioni stesse e riuscire ad ovviare un ordinamento dei dati in maniera controllata e gestibile da parte del amministratore e/o moderatore .

Per lo sviluppo di tale database si segue quindi un sistema di progettazione preciso:

2.1 Analisi

2.2 Modello E/R (Entity/Relationship), cioè uno schema concettuale astratto

2.3 Mapping, modello che rappresenta le associazioni logiche tra le varie istanze di entità

2.4 Creazione fisica delle tabelle

2.5 Interrogazione dei dati, dette query

2.1 Analisi

Il progetto richiede la gestione delle autovalutazioni dei tirocini effettuati dagli studenti negli enti o nelle aziende. Un tirocinio può essere effettuato da più studenti, anche in diverse date, ai quali verrà assegnato un insegnante come tutore interno della scuola. L'insegnante svolge la funzione di tutore ai soli studenti al quale insegna.

Le aziende danno la propria disponibilità nel dare spazio ai tirocinanti, anche in più sessioni ed in date differenti.

Il questionario di autovalutazione è costituito da domande tipizzate la cui risposta è associata ad una griglia di valutazione.

Gli studenti, inoltre, per accedere al modulo di autovalutazione da compilare, dovranno avvalersi di un account ottenuto attraverso una registrazione.

La scuola, mediante le statistiche ottenute, rileva gli enti e le aziende che hanno ricevuto un feedback positivo numeroso al fine di riuscire a selezionare in maniera più efficiente e dettagliata le potenziali aziende compatibili per i futuri tirocini degli studenti.

Dopo un'attenta analisi del progetto e delle eventuali ipotesi citate precedentemente, si identificano le entità ed associazioni che costituiscono la base di dati.

Entità

Accounting(user, password);
Ranghi(id_rango, nome, scrittura, lettura, modifica, cancellazione);
Accessi(id_accesso, logindate);
Studenti(matricola, cognome, nome);
Insegnanti(matricola, cognome, nome);
Classi(codice_classe, specializzazione);
Specializzazioni(id_specializzazione, descrizione);
EntiAziende(id_enteazienda, nome, indirizzo);
Stage(id_stage, attivitasettore, tutor_esterno);
Domande(id_domanda, descrizione);
TipiValutazione(tipo, descrizione);
Autovalutazioni(id_questionario, datavalutazione);

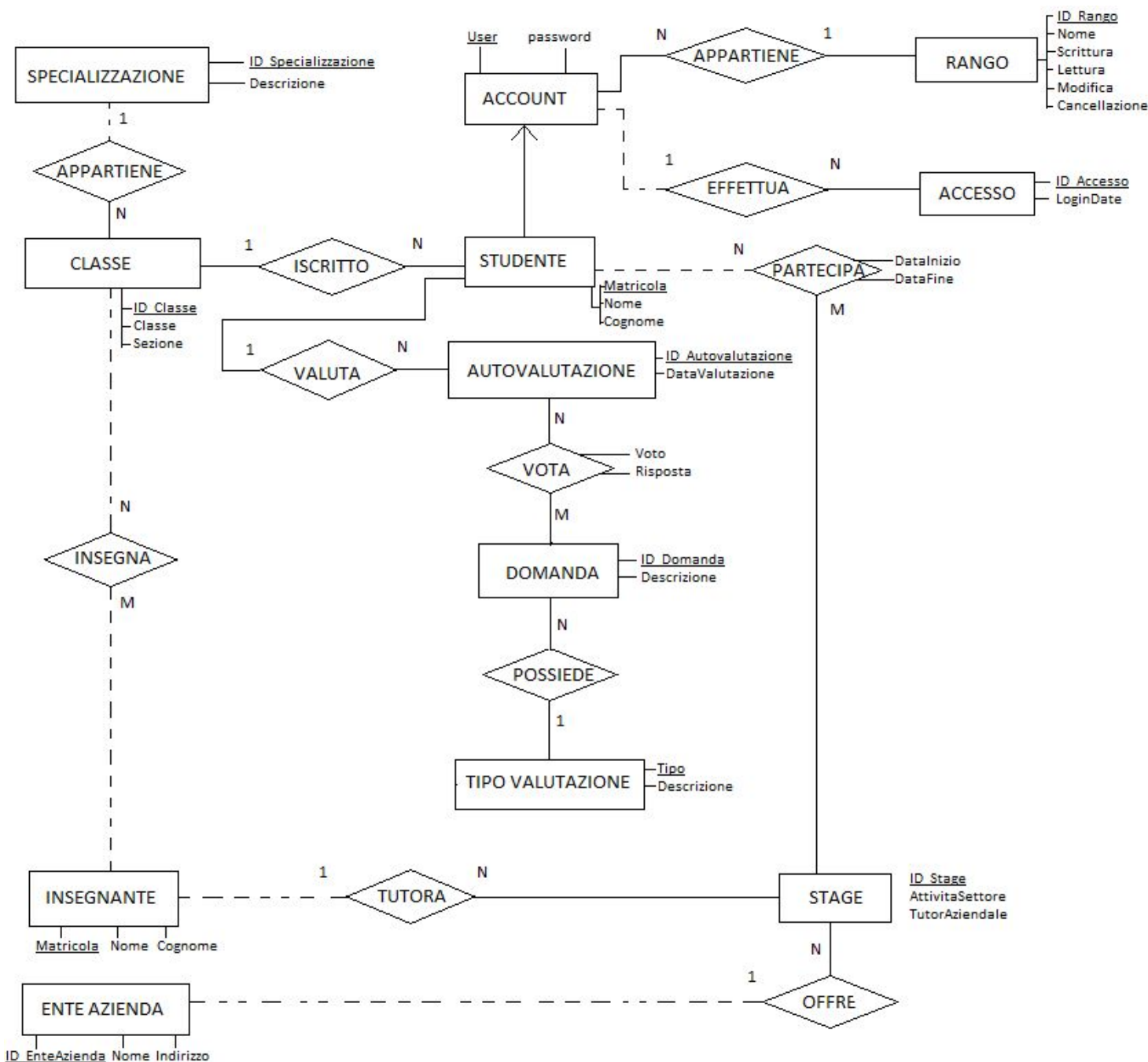
Associazioni

Accounting	<->	Ranghi,	N a 1;
Accounting	<->	Accessi,	1 a N;
Accounting	->	Studenti,	ISA;
Studenti	<->	Classi,	N a 1;
Insegnante	<->	Classi,	N a M;
Classi	<->	Specializzazioni,	N a 1;
Studenti	<->	Stage,	N a M;
Insegnante	<->	Stage,	1 a N;
EntiAziende	<->	Stage,	1 a N;
Studenti	<->	Autovalutazioni,	1 a N;
Stage	<->	Autovalutazioni,	1 a N;
Domande	<->	Autovalutazioni,	N a M;
Domande	<->	TipiValutazione,	N a 1;

Da tale analisi si definisce lo schema fisico concettuale.

2.2 Modello E/R

La relazione tra le varie entità è qui rappresentata dal modello e/r attraverso il diagramma seguente:



2.3 Mapping

Sulla base del modello fisico concettuale viene poi specificato il modello logico, chiamato “*mapping*” che definisce le relazioni logiche tra le istanze delle varie entità.

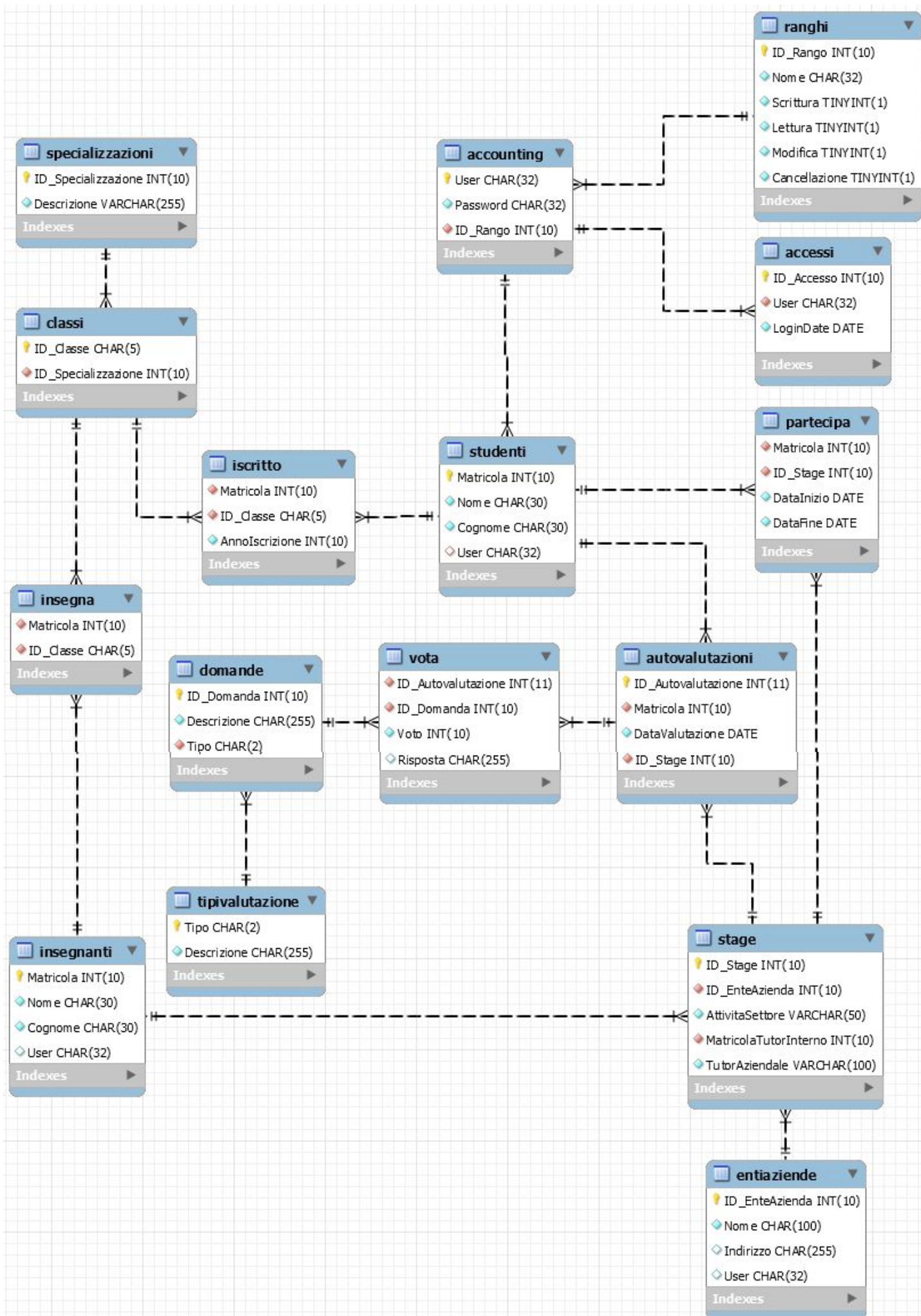
Le chiavi primarie verranno indicate affianco all’attributo sottolineato con [PK], le secondarie con [FK].

Per le associazioni “*molti a molti*” vi è la necessità di creare una tabella contenente le chiavi primarie delle singole tabelle associate.

Il mapping normalizzato alla terza forma normale, quindi, è il seguente:

```
Accounting([PK]user, password, [FK]id_rango);
Ranghi([PK]id_rango, nome, scrittura, lettura, modifica, cancellazione);
Accessi([PK]id_accesso, logindate, [FK]user);
Studenti([PK]matricola, cognome, nome, [FK]user);
Insegnanti([PK]matricola, cognome, nome);
Specializzazioni([PK]id_specializzazione, descrizione);
Classi([PK]codice_classe, specializzazione, [FK]id_specializzazione);
Insegna([FK]matricola, [FK]codice_classe);
EntiAziende([PK]id_enteazienda, nome, indirizzo);
Stage([PK]id_stage, attivitasettore, tutor_esterno, [FK]id_enteazienda,
[FK]matricola);
TipiValutazione([PK]tipo, descrizione);
Domande([PK]id_domanda, descrizione,[FK]tipo,);
Autovalutazioni([PK]id_questionario, datavalutazione, [FK]matricola,
[FK]id_stage );
Vota([FK]id_questionario, [FK]id_domanda, voto, risposta);
```

Ciò viene rappresentato da uno schema:



2.4 Creazione fisica delle tabelle

Per la creazione fisica delle tabelle del database è necessario un DBMS(DataBase Management System), ovvero un software il cui compito è quello di gestire dati e informazioni. Il DBMS utilizzato nel progetto è MySQL.

I comandi per la creazione delle tabelle sono i seguenti:

```
CREATE DATABASE ASL;
USE ASL;
CREATE TABLE Ranghi(
    ID_Rango int unsigned not null auto_increment,
    Nome char(32) not null,
    Scrittura boolean not null default false,
    Lettura boolean not null default false,
    Modifica boolean not null default false,
    Cancellazione boolean not null default false,

    PRIMARY KEY(ID_Rango)
);

CREATE TABLE Accounting(
    User CHAR(32) NOT NULL,
    Password CHAR(32) NOT NULL,
    ID_Rango INT UNSIGNED NOT NULL,

    PRIMARY KEY(User),
    FOREIGN KEY(ID_Rango) REFERENCES Ranghi(ID_Rango)
);

CREATE TABLE Accessi(
    ID_Accesso INT UNSIGNED NOT NULL AUTO_INCREMENT,
    User CHAR(32) NOT NULL,
    LoginDate DATE NOT NULL,

    PRIMARY KEY(ID_Accesso),
    FOREIGN KEY(User) REFERENCES Accounting(User)
);

CREATE TABLE Specializzazioni(
    ID_Specializzazione INT UNSIGNED NOT NULL AUTO_INCREMENT,
    Descrizione VARCHAR(255) NOT NULL,

    PRIMARY KEY(ID_Specializzazione)
);

CREATE TABLE Classi(
    ID_Classe CHAR(5),
    ID_Specializzazione INT UNSIGNED NOT NULL,

    PRIMARY KEY(ID_Classe),
    FOREIGN KEY(ID_Specializzazione) REFERENCES Specializzazioni(ID_Specializzazione)
);

CREATE TABLE Studenti(
    Matricola INT UNSIGNED NOT NULL auto_increment,
    Nome Char(30) NOT NULL,
    Cognome char(30) not null,
    User CHAR(32) DEFAULT NULL,
```

```

        PRIMARY KEY(Matricola),
        FOREIGN KEY(User) REFERENCES Accounting(User)
    );

CREATE TABLE Iscritto(
    Matricola INT UNSIGNED NOT NULL,
    ID_Classe CHAR(5) NOT NULL,
    AnnoIscrizione INT UNSIGNED NOT NULL,

    FOREIGN KEY(Matricola) REFERENCES Studenti(Matricola),
    FOREIGN KEY(ID_Classe) REFERENCES Classi(ID_Classe)
);

CREATE TABLE Insegnanti(
    Matricola INT UNSIGNED NOT NULL auto_increment,
    Nome Char(30) NOT NULL,
    Cognome char(30) not null,
    User CHAR(32) DEFAULT NULL,

    primary key(Matricola)
);

CREATE TABLE Insegna(
    Matricola INT UNSIGNED NOT NULL auto_increment,
    ID_Classe CHAR(5) NOT NULL,

    FOREIGN KEY(Matricola) REFERENCES Insegnanti(Matricola),
    FOREIGN KEY(ID_Classe) REFERENCES Classi(ID_Classe)
);

CREATE TABLE EntiAziende(
    ID_EnteAzienda INT UNSIGNED NOT NULL AUTO_INCREMENT,
    Nome CHAR(100) NOT NULL,
    Indirizzo CHAR(255) DEFAULT NULL,
    User CHAR(32) DEFAULT NULL,

    PRIMARY KEY(ID_EnteAzienda)
);

CREATE TABLE Stage(
    ID_Stage INT UNSIGNED NOT NULL AUTO_INCREMENT,
    ID_EnteAzienda INT UNSIGNED NOT NULL,
    AttivitaSettore VARCHAR(50) NOT NULL,
    MatricolaTutorInterno INT UNSIGNED NOT NULL,
    TutorAziendale VARCHAR(100) NOT NULL,

    PRIMARY KEY(ID_Stage),
    FOREIGN KEY(ID_EnteAzienda) REFERENCES EntiAziende(ID_EnteAzienda),
    FOREIGN KEY(MatricolaTutorInterno) REFERENCES Insegnanti(Matricola)
);

CREATE TABLE Partecipa(
    Matricola INT UNSIGNED NOT NULL,
    ID_Stage INT UNSIGNED NOT NULL,
    DataInizio DATE NOT NULL,
    DataFine DATE NOT NULL,

    FOREIGN KEY(Matricola) REFERENCES Studenti(Matricola),
    FOREIGN KEY(ID_Stage) REFERENCES Stage(ID_Stage)
);

```

```

CREATE TABLE AutoValutazioni(
    ID_Autovalutazione INT NOT NULL AUTO_INCREMENT,
    Matricola INT UNSIGNED NOT NULL,
    DataValutazione DATE NOT NULL,
    ID_Stage INT UNSIGNED NOT NULL,

    PRIMARY KEY(ID_Autovalutazione),
    FOREIGN KEY(Matricola) REFERENCES Studenti(Matricola),
    FOREIGN KEY(ID_Stage) REFERENCES Stage(ID_Stage)
);

CREATE TABLE TipiValutazione(
    Tipo CHAR(2) NOT NULL,
    Descrizione CHAR(255) NOT NULL,

    PRIMARY KEY(Tipo)
);

CREATE TABLE Domande(
    ID_Domanda INT UNSIGNED NOT NULL AUTO_INCREMENT,
    Descrizione CHAR(255) NOT NULL,
    Tipo CHAR(2) NOT NULL,

    PRIMARY KEY(ID_Domanda),
    FOREIGN KEY(Tipo) REFERENCES TipiValutazione(Tipo)
);

CREATE TABLE Vota(
    ID_Autovalutazione INT NOT NULL,
    ID_Domanda INT UNSIGNED NOT NULL,
    Voto INT UNSIGNED NOT NULL DEFAULT 2,
    Risposta CHAR(255) DEFAULT NULL,

    FOREIGN KEY(ID_Autovalutazione) REFERENCES AutoValutazioni(ID_Autovalutazione),
    FOREIGN KEY(ID_Domanda) REFERENCES Domande(ID_Domanda)
);

CREATE OR REPLACE VIEW InformazioniStage AS(
SELECT S.id_stage,
        Stud.nome as stud_nome, Stud.cognome as stud_cognome,
        EA.ID_EnteAzienda as id_azienza, EA.nome as azienda_nome,
        S.attivitasettore as attivita,
        S.tutoraziendale, I.nome as ins_nome,
        I.cognome as ins_cognome,
        P.datainizio, P.datafine
FROM
    Studenti as Stud,
    Partecipa as P,
    Insegnanti as I,
    Stage as S,
    EntiAziende as EA
WHERE
    Stud.matricola = P.matricola AND
    P.id_stage = S.id_stage AND
    S.id_enteazienda = EA.id_enteazienda AND
    S.matricolatutorinterno = I.matricola)

```

2.5 Query

Le query sono comandi sviluppati per estrarre o aggiornare dati all'interno di un database.

Le principali interrogazioni effettuate al database, utilizzate per uno sviluppo di dati staticistici sono:

```
SELECT Studenti.ID_Alunno, CONCAT(Cognome, ' ', Nome) AS Studente,
AVG(Risposte.ID_Valutazione) AS MediaValutazione
FROM Studenti, Questionari, Domande, Risposte
WHERE Studenti.ID_Alunno = Autovalutazioni.ID_Alunno AND
Risposte.ID_Questionario = Autovalutazioni.ID_Questionario AND
Risposte.ID_Domanda = Domande.ID_Domanda AND
Domande.Tipo = TipiValutazione.Tipo AND
TipiValutazione.Descrizione <> 'DA'
GROUP BY Studenti.ID_Alunno
ORDER BY Studente DESC
```

che restituisce la media dei voti di ogni singolo studente,

```
SELECT Domande.id_domanda, Domande.descrizione, Griglia_Valutazioni.descrizione AS voto,
(100/R.tot_questionari)*COUNT(Risposte.id_valutazione) AS percentuale
FROM Risposte, Domande, Griglia_Valutazioni, ( SELECT COUNT(*) as tot_questionari
FROM Questionari) AS R
WHERE Risposte.ID_Domanda = Domande.ID_Domanda AND
Risposte.ID_Valutazione = Griglia_Valutazioni.ID_Valutazione AND
Domande.Tipo = 'C' AND
Domande.id_domanda = ". $row["id_domanda"]."
GROUP BY Domande.id_domanda, Risposte.id_valutazione
```

che restituisce la percentuale di ogni voto per ogni domanda del questionario.

Vi sono inoltre query essenziali che restituiscono dati da elaborare attraverso il linguaggio di scripting PHP come:

il caricamento delle domande nel questionario

```
SELECT * FROM Domande
```

il salvataggio delle risposte di ogni domanda [dove “?” sta per variabile]

```
INSERT INTO Vota(ID_Autovalutazione, ID_Domanda, Voto)
VALUES (?, ?, ?)
```

3. Sviluppo delle pagine dinamiche

Nello sviluppo delle pagine dinamiche invece entra in gioco l'utilizzo del linguaggio di scripting PHP, già conosciuti negli ambiti informatici web.

PHP è un linguaggio che permette di elaborare dati inviati o ricevuti dall'utente garantendo una comunicazione diretta tra il server e il database.

Grazie a PHP è possibile generare pagine differenti dipendentemente dai dati in elaborazione che dovranno essere inviate al browser.

In questo progetto, due sono le interfacce essenziali sviluppate:

3.1 "front-end", interfaccia lato utente;

3.2 "back-end", interfaccia lato server;

Entrambe le interfacce avranno come punto di riferimento il file "question.php" e "load_question.php".

3.1 Front-end, Interfaccia lato utente

Affinché l'utente possa facilmente interagire con le pagine è necessario uno stile grafico definito dal linguaggio di mark-up CSS (Cascading Style Sheets).

Per facilitare lo sviluppo grafico delle pagine web anche a chi non è di competenza,

è utile l'uso di un framework particolare, ovvero Bootstrap.

Esso è incluso in tutte le pagine dell'applicazione e si riferisce alla sintassi

```
<link rel="stylesheet" href="bootstrap-3.3.6-dist/css/bootstrap.css" type="text/css">
```

con la definizione degli stili nell'attributo `class` di ogni tag HTML.

Un esempio chiaro è lo stile formale definito dalla classe "panel".

Per rendere la pagina dinamica più amichevole si utilizza la tecnologia AJAX, grazie alla quale non è necessario il ripetuto refresh della pagina per aggiornarla.

AJAX fa ricorso, oltre al PHP, ad un altro linguaggio di scripting JavaScript al quale si appoggia la libreria jQuery.

Esso ha la funzionalità di controllare che tutti i parametri siano rigorosamente completati prima dell'invio dei dati al server ed una eventuale richiesta delle risorse in rete.

REPORT DELLO STUDENTE

Avevi le conoscenze adeguate per affrontare i compiti che ti sono stati assegnati

☐ Per niente ☐ Poco ☐ Così Così ☐ Sufficiente

I compiti assegnati ti sono sembrati coerenti con il percorso formativo da te intrapreso

☐ Per niente ☐ Poco ☐ Così Così ☐ Sufficiente

Quanto impegno ti sembra di aver dedicato all'attività svolta

☐ Per niente ☐ Poco ☐ Così Così ☐ Sufficiente

Nel contesto in cui sei stato/a inserito/a hai avuto spazi di autonomia e di iniziativa personale))

☐ Per niente ☐ Poco ☐ Così Così ☐ Sufficiente

Sei Soddisfatto dell'immagine di te che hai trasmesso

☐ Per niente ☐ Poco ☐ Così Così ☐ Sufficiente

Ritieni di aver tratto vantaggio dall'esperienza di Alternanza Scuola Lavoro

☐ Per niente ☐ Poco ☐ Così Così ☐ Sufficiente

Ritieni che l'Alternanza sia una modalità di apprendimento efficace

☐ Per niente ☐ Poco ☐ Così Così ☐ Sufficiente

Il tempo trascorso in azienda, rispetto alle finalità generali del progetto ed ai contenuti dell'Alternanza, è stato adeguato

☐ Per niente ☐ Poco ☐ Così Così ☐ Sufficiente

Rispetto alle tue aspettative l'ambiente di lavoro che hai sperimentato ti è sembrato migliore di come lo immaginavi

☐ Per niente ☐ Poco ☐ Così Così ☐ Sufficiente

Da questa esperienza pesi di aver ricavato maggiori sollecitazioni allo studio

☐ Per niente ☐ Poco ☐ Così Così ☐ Sufficiente

Da questa esperienza pesi di aver ricavato utili indicazioni al fine di orientare le tue scelte future

☐ Per niente ☐ Poco ☐ Così Così ☐ Sufficiente

Quali sono stati gli aspetti positivi del Progetto di Alternanza?

Quali sono stati gli aspetti negativi del Progetto di Alternanza?

Che cosa cambieresti per migliorare il Progetto di Alternanza?

Quali materie di studio previste nel percorso curricolare sono state direttamente interessate dall'esperienza lavorativa?

Invia autovalutazione

3.2 Back-end, interfaccia lato server

Il cuore dell'interfaccia lato server è il linguaggio PHP dove esso è computato dalla macchina stessa che fornisce le risorse al pubblico.

Una delle funzioni frequenti e possibili del linguaggio PHP è la connessione al database, definito dal codice “connection.php”:

```
//CONNESSIONE AL DATABASE
    $servername = "localhost";
    $account = "root";
    $password = "admin";
    $db = "asl";

    $conn = new mysqli($servername, $account, $password, $db);

    //se non stabilisce la connessione avverti
    if($conn->connect_error)
        die("<div class=\"alert alert-warning\"><strong>Attenzione!</strong> Errore di
connessione al database.</div>");
```

Il server sfrutta in modo efficiente il PHP: infatti, attraverso quest'ultimo, la macchina è in grado di sviluppare il questionario attraverso l'elaborazione delle domande contenute all'interno del database.

Il codice è definito in “new_question2.php” che richiama un'altra pagina PHP chiamata “load_question.php”:

new_question2.php

```
<form id="" action="/process.php" method="post" role="form">
    <?php require 'load_questions.php';?>
    <div class="form-group">
        <div class="row">
            <div class="col-sm-6 col-sm-offset-3">
                <input type="submit" name="questionario-submit"
id="questionario-submit" class="form-control btn btn-success" value="Invia verdetto!">
            </div>
        </div>
    </div>
</form>
```

“load_question.php” si occuperà di estrarre le domande dal database e di costruire il questionario in una form

```
<?php
//CARICA LE DOMANDE/RECORD DELLA TABELLA Domande.
require 'connection.php';

$query = "SELECT * FROM Domande";
$result = $conn->query($query);

while($row = $result->fetch_assoc()){
    //creo riga contenente domanda | input(radio button o textarea)
    $div_row = "<div class=\"row\">";

    //se la domanda non è DA(domanda aperta)
    if($row["Tipo"]!="DA"){
        $div_row .= "<div class=\"col-sm-6 col-sm-offset-1\">"
        . $row["Descrizione"] . "</div>";
        $div_row .= "<div class=\"col-sm-4\">";
        //crea domanda con i radiobutton
        $radio = "
            <div class=\"form-group\">
                <label class=\"radio-inline\"><input type=\"radio\"
name=\"q\".$row[\"ID_Domanda\"].\"\" value=1 checked>Per niente</label>
                <label class=\"radio-inline\"><input type=\"radio\"
name=\"q\".$row[\"ID_Domanda\"].\"\" value=2>Poco</label>
                <label class=\"radio-inline\"><input type=\"radio\"
name=\"q\".$row[\"ID_Domanda\"].\"\" value=3>Cosi Cosi</label>
                <label class=\"radio-inline\"><input type=\"radio\"
name=\"q\".$row[\"ID_Domanda\"].\"\" value=4>Sufficiente</label>";
        $div_row .= $radio . "</div></div>";
    }else{
        //sennò crea domanda | textarea
        $div_row .= "<div class=\"col-sm-10 col-sm-offset-1\"><div
class=\"panel-group\">";
        $div_row .= "
            <div class=\"form-group\">
                <div class=\"panel panel-info\">
                    <div class=\"panel panel-heading\">
                        <label
for=\"q\".$row[\"ID_Domanda\"].\"\">".$row["Descrizione"]."</label>
                    <div>
                        <textarea class=\"form-control noresize\"
maxlength=\"255\" rows=\"5\" id=\"q\".$row[\"ID_Domanda\"].\"\"
name=\"q\".$row[\"ID_Domanda\"].\"\"></textarea>
                    </div>
                </div>";
        $div_row .= "</div></div></div></div>";
    }

    $div_row .= "<div class=\"col-sm-2\"></div>";
    $div_row .= "</div>";

    //stampa domanda costruita
    echo $div_row;
}
```


“process.php” ha il compito di salvare nel database la valutazione del questionario:

```
<?php
    session_start();
    require 'connection.php';
    //controllo sui bottoni e sulla sessione
    if(!isset($_POST["questionario-submit"])||!isset($_SESSION["login"])||!isset($_SESSION["quest"]
    )){
        header("Location: /home.php");
        exit;
    }
?>

<!DOCTYPE HTML>
<html>
    <head>
        <?php
            require '/essentials/head.php';
        ?>
        <title>Compilazione questionario</title>
    </head>
    <body>
        <div class="container">
            <?php
                $matricola_studente = $_SESSION["login"]["matricola"];
                $id_stage = $_SESSION["quest"]["id_azienza"];
                $date = Date("Y-m-d");

                //INSERIMENTO DEL QUESTIONARIO
                $nuovo_quest = $conn->prepare("INSERT INTO Autovalutazioni(Matricola,
DataValutazione, ID_Stage) VALUES(?,?,?)");
                $nuovo_quest->bind_param("isi", $matricola_studente, $date, $id_stage);
                $nuovo_quest->execute();

                //ESTRAZIONE ID DEL NUOVO QUESTIONARIO CREATO
                $result = $conn->query("SELECT          id_autovalutazione FROM
Autovalutazioni
                                WHERE   matricola =
'$matricola_studente' AND
                                id_stage =
'$id_stage' AND datavalutazione = '$date'");
                $record = $result->fetch_assoc();
                $id_quest = $record["id_autovalutazione"];

                //calcolo del numero delle domande
                $query = "SELECT COUNT(*) AS n_domande FROM Domande";
                $result = $conn->query($query);
                $row = $result->fetch_assoc();
                //prendo il numero delle domande che sono nel database
                $n_domande = $row["n_domande"];

                //controllo sugli input cosicche non ci siano errori spontanei alla
modifica dell'html da parte
                //dell'utente
                for($i=1; $i<=$n_domande; $i++){
                    if(!isset($_POST["q$i"])){
```

```

        die("<div class=\"alert
alert-danger\"><strong>Attenzione!</strong> Errore di compilazione questionario. Attendere
5secondi...</div>");

        header("refresh:3; url=new_question.php");
    }

    //inserimento delle domande una ad una
    for($i=1; $i<=$n_domande; $i++){
        $query = "SELECT ID_Domanda, Tipivalutazione.Tipo,
Tipivalutazione.Descrizione from Tipivalutazione, Domande
                                WHERE Domande.Tipo = Tipivalutazione.Tipo
AND ID_Domanda = $i
                                ORDER BY ID_Domanda";
        $result = $conn->query($query);
        $row = $result->fetch_assoc();

        $risposta = $_POST["q$i"];

        //preparo l'inserimento
        //se domanda chiusa
        $query_insert_chiuso = "INSERT INTO
Vota(ID_Autovalutazione,ID_Domanda,Voto) VALUES(?,?,?)";
        $query_insert_aperto = "INSERT INTO
Vota(ID_Autovalutazione,ID_Domanda,Risposta) VALUES(?,?,?)";
        $stmt_chiuso = $conn->prepare($query_insert_chiuso);
        $stmt_chiuso->bind_param("iii", $id_quest, $i, $risposta);

        $stmt_aperto = $conn->prepare($query_insert_aperto);
        $stmt_aperto->bind_param("iis", $id_quest, $i, $risposta);

        if($row["Tipo"]!="DA"){
            echo $stmt_chiuso->execute();
        }else{
            echo $stmt_aperto->execute();
        }
    }

    unset($_SESSION["quest"]);
    $GLOBALS["msg"] = "<div class=\"alert
alert-success\"><strong>Completato!</strong> Hai completato correttamente il questionario.
Attendere 5secondi...</div>";

    header("Location: home.php");
    ?>
</div>
</body>
</html>

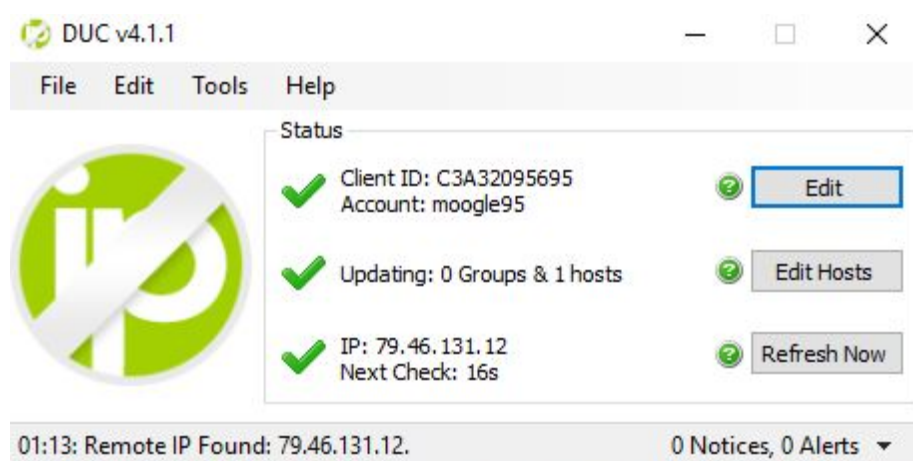
```

4. Pubblicazione dell'applicazione web

Il progetto è accessibile da browser digitando sulla barra dell'indirizzo il seguente dominio: www.ralphumandal.ddns.net.

Attraverso il DNS, protocollo applicativo non orientato alla connessione avviato nell'istante in cui viene richiesta l'applicazione web da browser, è possibile la traduzione del dominio in un indirizzo IP pubblico.

Il provider *no-ip* fornisce quindi un servizio di DNS gratuito, il quale grazie ad un software che fornisce lui stesso, aggiorna il database su cui vi è l'associazione nome<>ip.



Non essendo economicamente predisposto per l'acquisto di un dedicato o di un server per poter fare housing, il sistema che ho sviluppato si appoggia ad una macchina virtualizzata, cioè l'emulazione di sistema operativo all'interno del PC personale, che viene utilizzata come server di hosting.

Per hosting si intende un servizio di rete in cui vengo allocate le pagine web in un web-server e rese accessibili dalla rete Internet.

Per il progetto vi è l'utilizzo del software "VMware" per l'emulazione del sistema operativo "Windows 10".

All'interno della macchina virtuale (server) vi è installato quindi il software di aggiornamento DNS, Apache2.4 per il servizio web HTTP con PHP e il DBMS MySQL.

Configurazione della macchina virtuale

```
C:\Windows\system32\cmd.exe

Server DNS . . . . . : fec0:0:0:ffff::1%1
                   : fec0:0:0:ffff::2%1
                   : fec0:0:0:ffff::3%1
NetBIOS su TCP/IP . . . . . : Attivato

Scheda Ethernet Ethernet0:

Suffisso DNS specifico per connessione:
Descrizione . . . . . : Intel(R) 82574L Gigabit Network Connection
Indirizzo fisico. . . . . : 00-0C-29-3B-60-CA
DHCP abilitato. . . . . : Si
Configurazione automatica abilitata . . . . . : Si
Indirizzo IPv6 locale rispetto al collegamento . : fe80::704e:255d:f889:ba48%11(Preferenziale)
Indirizzo IPv4. . . . . : 192.168.0.201(Preferenziale)
Subnet mask . . . . . : 255.255.255.0
Lease ottenuto. . . . . : venerdì 7 maggio 1880 04:00:59
Scadenza lease . . . . . : giovedì 2 giugno 2016 02:13:25
Gateway predefinito . . . . . : 192.168.0.1
Server DHCP . . . . . : 192.168.0.1
IAID DHCPv6 . . . . . : 50334761
DUID Client DHCPv6. . . . . : 00-01-00-01-1E-CF-F9-F6-00-0C-29-3B-60-CA
Server DNS . . . . . : 192.168.0.1
NetBIOS su TCP/IP . . . . . : Attivato

Scheda Tunnel isatap.{B89F1721-677D-4AD0-B25F-41BEA178E634}:

Stato supporto. . . . . : Supporto disconnesso
Suffisso DNS specifico per connessione:
Descrizione . . . . . : Microsoft ISATAP Adapter
Indirizzo fisico. . . . . : 00-00-00-00-00-00-E0
```

INDIRIZZO IP LOCALE: 192.168.0.201
SUBNET MASK: 255.255.255.0
GATEWAY: 192.168.0.1
Server DNS: 192.168.0.1 (fornito da isp utilizzabile da tutti gli altri dispositivi in rete)

Si noti che il router è impostato affinché la macchina virtualizzata, alla richiesta del DHCP, riceva un indirizzo fisso (*192.168.0.201*).

Impostazione di TCP/IP LAN

Indirizzo IP

Subnet mask IP

Instradamento RIP

Versione RIP

192

168

0

1

255

255

255

0

Entrambi

Disattivata

☒ Usa router come server DHCP

Indirizzo IP iniziale

Indirizzo IP finale

192

168

0

2

192

168

0

254

Prenotazione indirizzi

	#	Indirizzo IP	Nome dispositivo	Indirizzo MAC
<input type="radio"/>	1	192.168.0.9	MOOGLE	C0:EE:FB:33:3B:CF
<input type="radio"/>	2	192.168.0.100	MySky HD	A0:C5:62:E8:7E:3D
<input type="radio"/>	3	192.168.0.200	MEW-PC	1C:87:2C:46:20:F5
<input type="radio"/>	4	192.168.0.50	MTAB	8C:18:D9:46:CA:5E
<input type="radio"/>	5	192.168.0.201	Server Virtuale	00:0C:29:3B:60:CA

Il modemrouter utilizzato è un modello DGN2200 v3 della NETGEAR che fornisce servizi di sicurezza (*firewall*) e il *portforwarding* necessari per il progetto da fornire in pubblico dominio.

In dettaglio vi è l'apertura della porta 80 su cui rimarrà in ascolto la macchina virtuale.

Inoltro/Attivazione delle porte

Applica Annulla

Selezionare il tipo di servizio

☐ Inoltro delle porte

☒ Attivazione delle porte

☐ Disattiva attivazione delle porte

Timeout di attivazione delle porte(in minuti) 9999

Tabella portmap per attivazione delle porte

	#	Attiva	Nome servizio	Tipo servizio	Connessione in entrata	Utente servizio
<input type="radio"/>	1	<input type="checkbox"/>	MySky HD	TCP:60000	TCP/UDP:60000	192.168.0.100
<input type="radio"/>	2	<input checked="" type="checkbox"/>	Server Desktop Remoto	TCP:3389	TCP:3389	192.168.0.201
<input type="radio"/>	3	<input checked="" type="checkbox"/>	VPN	TCP:1723	TCP/UDP:1723	192.168.0.201
<input type="radio"/>	4	<input checked="" type="checkbox"/>	Custom RDP port	TCP:40000	TCP:40000	192.168.0.201
<input type="radio"/>	5	<input checked="" type="checkbox"/>	HTTP WebServer	TCP:80	TCP/UDP:80	192.168.0.201

Gli utenti esterni alla richiesta delle risorse della porta 80, grazie al portforwarding che fornisce NETGEAR, comunicheranno direttamente con il webserver virtuale.

Inoltro/Attivazione delle porte

Selezionare il tipo di servizio

☒ Inoltro delle porte

☐ Attivazione delle porte

Nome servizio FTP(TCP:20,21)

Indirizzo IP server 192 . 168 . 0 . + Aggiungi

	#	Nome servizio	Porta iniziale esterna	Porta finale esterna	Porta iniziale interna	Porta finale interna	Indirizzo IP interno
<input type="radio"/>	1	VPN-PPTP(TCP:1723)	1723	1723	1723	1723	192.168.0.201
<input type="radio"/>	2	Server Desktop Remoto	3389	3389	3389	3389	192.168.0.201
<input type="radio"/>	3	HTTP(TCP:80)	80	80	80	80	192.168.0.201
<input type="radio"/>	4	Custom RDP Port	40000	40000	40000	40000	192.168.0.201