

Università degli Studi di Messina



Dipartimento di Scienze matematiche e informatiche,
scienze fisiche e scienze della terra

Corso di Laurea Triennale in Informatica

Progetto di

PROGRAMMAZIONE 3

‘ Web app di gestione di un negozio di abbigliamento’

Realizzato da

GIOVANNI DOMENICO TRIPODI

Anno Accademico

2020/2021

1. INTRODUZIONE

Si vuole progettare un'applicazione web gestionale dedicata ad un negozio di abbigliamento. L'uso di tale sistema sarà destinato al direttore, agli addetti vendita, ai magazzinieri e ai fornitori, ognuno di essi andrà ad interagire in maniera diversa con l'applicazione, in base alle mansioni che derivano dal ruolo che essi occupano all'interno del negozio.

1.1 Tecnologie utilizzate

Le tecnologie utilizzate per lo sviluppo del sito web sono le seguenti:

1. Lato client
 - HTML
 - CSS
 - JavaScript
 - AJAX (tramite jQuery)
2. Lato server
 - PHP

2. SPECIFICHE DELLE OPERAZIONI

Il sistema permette le seguenti operazioni:

- Operazioni CRUD sul direttore (amministratore) applicate sul sistema;
- Operazioni di read e update da parte del personale applicate al magazzino e agli ordini;
- Operazione di read e update sugli Addetti Vendita e Magazzinieri con relativi reparti (addetti vendita) e orari lavorativi (addetti vendita e magazzinieri);

3. LISTA DEGLI ATTORI

Gli utenti che potranno accedere al sistema sono:

- Il Direttore del negozio (o Amministratore);
- Addetti vendita;
- Magazzinieri;
- I fornitori;

Direttore

Il Direttore (o Amministratore) è l'utente che ha la possibilità di:

- Aggiornare le informazioni sul negozio;
- Inserire e modificare gli orari lavorativi dei dipendenti;
- Visualizzare le informazioni;
- Aggiungere gli oggetti presenti nel negozio;
- Modificare gli oggetti presenti nel negozio;
- Eliminare gli oggetti presenti nel negozio;
- Effettuare gli ordini;
- Impostare il reparto di appartenenza per ogni addetto vendita;

Addetto vendita

L'addetto vendita è l'utente che ha la possibilità di:

- Verificare gli articoli presenti nel magazzino;
- Visualizzare le informazioni degli articoli;
- Registrare gli articoli sistemandoli nei reparti;

- Aggiungere gli oggetti presenti nel negozio;
- Modificare gli oggetti presenti nel negozio;
- Eliminare gli oggetti presenti nel negozio;
- Creare ordini;
- Visualizzare il proprio orario lavorativo;

Magazziniere

Il Magazziniere è l'utente che ha la possibilità di:

- Confermare la ricezione o l'invio di un ordine;
- Visualizzare il proprio orario lavorativo;

Fornitore

Il Fornitore è l'utente che ha la possibilità di:

- Prendere in carico o rifiutare gli oggetti da fornire;
- Aggiornare lo stato dell'ordine;
- Verificare se ci sono ordini da prendere in carico;

4. PROGETTAZIONE DEL DATABASE

All'interno del sistema, i dipendenti memorizzeranno gli articoli. Gli articoli saranno divisi in:

- Articoli del negozio:** gli articoli presenti all'interno del negozio. Ogni articolo sarà depositato all'interno di uno scaffale.

- Articoli ordinati:** questi articoli sono relativi agli ordini effettuati dal responsabile del negozio, essi non sono ancora presenti all'interno del negozio, bensì una volta che il fornitore consegnerà l'ordine, gli oggetti saranno spostati al magazzino;

L'articolo generico sarà memorizzato all'interno del sistema secondo un codice che lo identifica univocamente, un nome e un costo.

La funzionalità principale del sistema è quella di creare un inventario per gli articoli del negozio.

L'addetto alla vendita può verificare la presenza di un particolare articolo all'interno del magazzino, e se nel caso non fosse presente, può provvedere a creare un ordine. L'addetto alla vendita, una volta sistemati gli articoli all'interno di uno scaffale, dovrà memorizzarli all'interno del sistema indicando il reparto di appartenenza. Quindi, per quanto riguarda l'addetto alla vendita, le funzioni possibili all'interno del sistema saranno quelle di inserimento di un articolo, eliminazione e segnalazione (di ordine o di reso).

Un'altra funzionalità del sistema è la possibilità di creare ordini da parte degli addetti vendita e del direttore.

I fornitori saranno abilitati ad accedere al sistema in maniera tale da verificare se ci sono delle richieste di ordine per poi prenderle in carico o rifiutarle nel caso non fosse possibile soddisfare le richieste..

I fornitori inoltre, dovranno aggiornare lo stato di avanzamento dell'ordine in maniera tale da informare l'intero punto vendita.

I magazzinieri, ai fini del sistema, avranno il compito di poter confermare la

ricezione di un ordine.

Il direttore potrà impostare e gestire i vari turni di lavoro di ogni dipendente.

ELENCO ENTITA'

NEGOZIO

Negozio è l'entità che contiene le informazioni riguardanti il negozio.

Descrizione attributi:

id_negozio: Identifica univocamente il negozio;

nome: Indica il nome del negozio;

indirizzo: Identifica l'indirizzo civico del negozio;

PERSONA

L'entità Persona descrive gli utenti che utilizzano il sistema.

Descrizione attributi:

id_persona: Identifica univocamente la persona;

nome: è il nome della persona;

cognome: è il cognome della persona;

email: la persona utilizzerà la email per registrarsi al sistema;

password: indica la password utilizzata nel momento della registrazione;

REPARTO

L'entità reparto descrive una area del negozio.

Descrizione attributi:

id_reparto: Identifica univocamente un reparto;

nome: è il nome del reparto;

SEZIONE

L'entità Sezione contiene tutti i tipi di reparto.

Descrizione attributi:

id_sezione: Identifica univocamente la sezione;

sezione: è il nome della sezione;

ORARIO LAVORATIVO

L'entità Orario lavorativo descrive i turni del personale.

Descrizione attributi:

id_lavorativo: Identifica univocamente un'orario lavorativo;

ORARIO

L'entità Orario descrive gli orari di un determinato turno.

Descrizione attributi:

id_orario: Identifica univocamente l'orario;

orario: indica l'orario di un turno;

GIORNO

L'entità Giorno indica il giorno di un determinato turno.

Descrizione attributi:

id_orario: Identifica univocamente il giorno;

orario: indica il giorno di un turno;

ARTICOLO

L'entità Articolo descrive l'articolo presente all'interno del sistema.

Descrizione attributi:

id_articolo: Identifica univocamente l'articolo;

nome_articolo: indica il nome di un articolo;

quantita': indica la quantità di un articolo presente nel sistema;

prezzo: indica il prezzo di un articolo;

FORNITORE

L'entità Fornitore contiene le informazioni dei Fornitori del negozio.

Descrizione attributi:

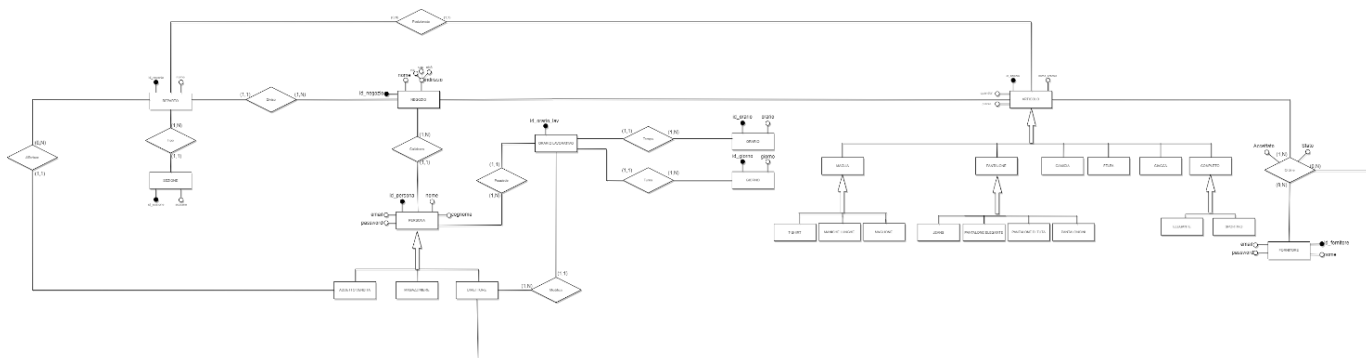
id_fornitore: Identifica univocamente il fornitore;

nome: indica il nome di un fornitore;

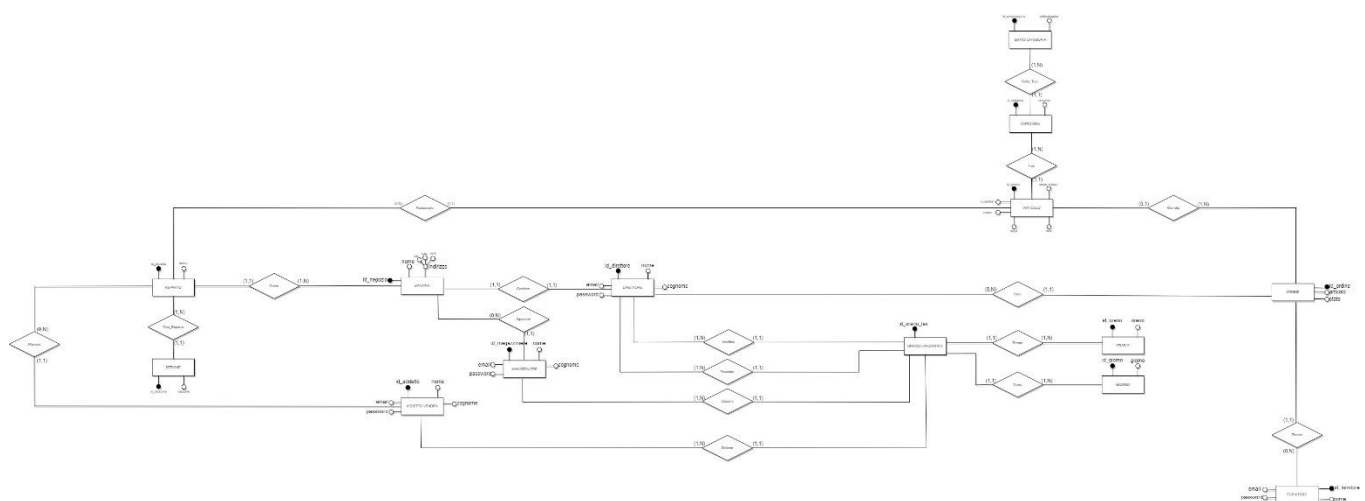
email: email utilizzata nel momento della registrazione al sistema;

password: indica la password utilizzata nel momento della registrazione;

MODELLO E-R



MODELLO E-R RISTRUTTURATO



MODELLO LOGICO

Attributo sottolineato -> PRIMARY KEY

Attributo in grassetto -> FOREIGN KEY

NEGOZIO (id_negozio, id_direttore, nome, via, cap, città)

REPARTO (id_reparto, id_negozio, nome)

ADDETTO VENDITA (id_addetto, id_reparto, nome, cognome, email, password)

MAGAZZINIERE (id_magazziniere, id_negozio, nome, cognome, email, password)

DIRETTORE (id_direttore, id_negozio, nome, cognome, email, password)

ORARIO LAVORATIVO (id_orario_lav, id_direttore, id_magazziniere, id_addetto, id_orario, id_giorno)

ORARIO (id_orario, orario)

GIORNO (id_giorno, giorno)

ORDINE (id_ordine, id_direttore, id_fornitore, articolo, stato)

FORNITORE (id_fornitore, nome, email, password)

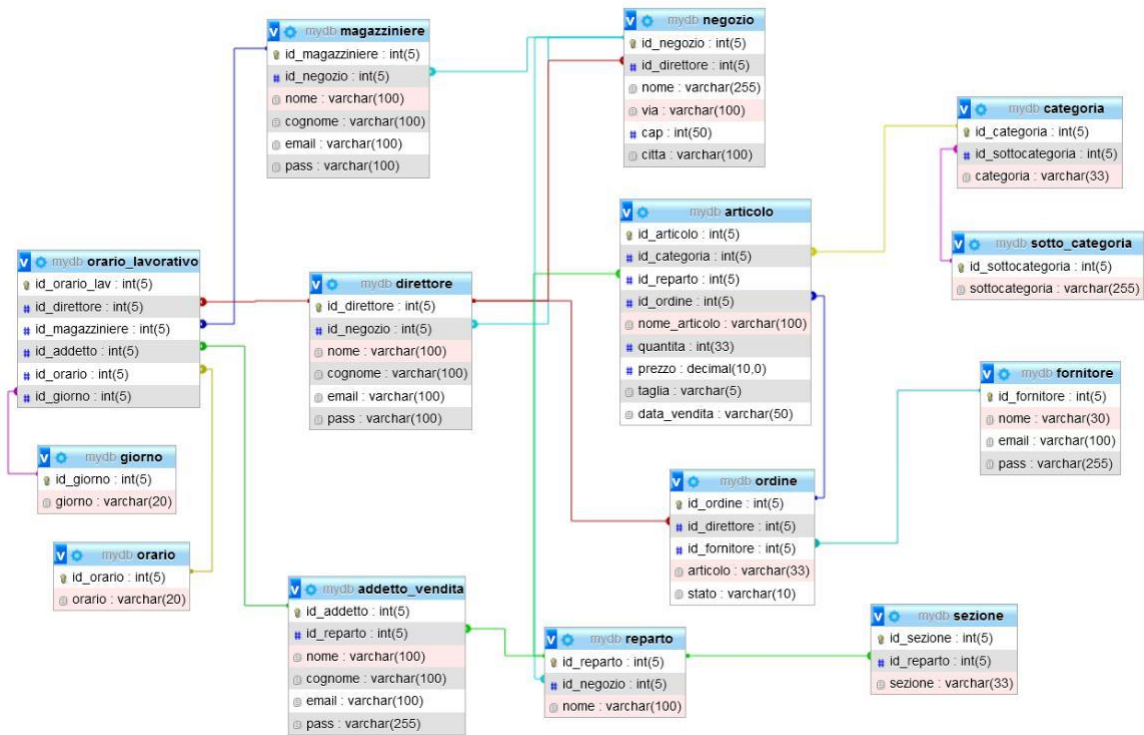
ARTICOLO (id_articolo, id_reparto, id_ordine, nome_articolo, quantita', prezzo, taglia, data)

CATEGORIA (id_categoria, id_articolo, categoria)

SOTTO CATEGORIA (id_sottocategoria, id_categoria, sottocategoria)

SEZIONE (id_sezione, id_reparto, sezione)

DATABASE DESIGN



5. RESTful API

Per andare ad utilizzare una architettura di tipo RESTful, bisogna prima mappare i metodi CRUD (*Create, Read, Update, Delete*).

Ad ogni protocollo HTTP viene associata una determinata operazione:

Metodo HTTP	Op. CRUD	Descrizione
POST	Create	Crea una risorsa
GET	Read	Ottieni una risorsa
PUT	Update	Aggiorna una risorsa
DELETE	Delete	Elimina una risorsa

Tutte le operazioni del sistema sono state mappate seguendo la tabella sovrastante.

AUTENTICAZIONE

Per autenticare un utente e quindi effettuare il login, si utilizza *JSON Web Token* (JWT).

Esso costituisce un codice di accesso (token) e consente lo scambio di dati tra le due parti. Nel caso dello sviluppo di Applicazioni REST, il JWT assicura che l'applicazione sia priva di stato, inviando le informazioni di autenticazione direttamente con la richiesta.

POST Autenticazione

Questa operazione invia al server email e password dell'utente in questione e restituisce un token per autenticarsi all'interno del sistema.

api/auth/login.php

Body Request

Chiave	Obbligatorio	Descrizione
Email	Si	Email dell'utente che vuole accedere al sistema
Password	Si	Indica la password associata all'utente

Body response (application/json)

200 OK

Chiave	Valore
Message	Login riuscito
Jwt	Stringa che identifica l'accesso
Token	Contiene dati utente, ora di generazione del token e tempo di scadenza del token

401 Not authorized

Chiave	Valore
Message	Login fallito

ARTICOLO

POST Crea articolo

Inserisce un nuovo articolo con determinati parametri.

api/article/create.php

Body Request

Chiave	Obbligatorio	Descrizione
Id_sottocategoria	No	Indica l'identificativo della sottocategoria
Id_categoria	Si	Indica l'identificativo della categoria
Id_reparto	Si	Indica l'identificativo del reparto
Nome_articolo	Si	Indica il nome dell'articolo
Quantita	Si	Indica la quantita da inserire
Prezzo	Si	Prezzo dell'articolo
Taglia	Si	Taglia dell'articolo

Body response (application/json)

201 CREATED

Chiave	Valore
Message	L'articolo è stato creato correttamente

503 Service Unavailable

Chiave	Valore
Message	Impossibile creare l'articolo.

400 Bad Request

Chiave	Valore
Message	Impossibile creare l'articolo. I dati sono incompleti.

DELETE Elimina articolo

Permette di eliminare un articolo.

api/article/delete.php

Body Request

Chiave	Obbligatorio	Descrizione
Id_articolo	Sì	Identificativo dell'articolo da eliminare.

Body response (application/json)

200 OK

Chiave	Valore
Message	L'articolo è stato eliminato con successo.

503 Service Unavailable

Chiave	Valore
Message	Impossibile eliminare l'articolo.

GET Trova articoli da sistemare

Trova gli articoli che devono essere sistemati nei reparti.

api/article/read_arrange.php

Body Request

Non abbiamo parametri all'interno della body request, in questo caso.

Body response (application/json)

200 OK

Chiave	Valore
Id_articolo	Identificativo dell'articolo
Nome_articolo	Nome dell'articolo

Taglia	Taglia dell'articolo
--------	----------------------

404 Not found

Chiave	Valore
Message	Nessun articolo trovato.

GET Trova un singolo articolo.

Restituisce tutte le informazioni di un singolo articolo.

```
api/article/read.php?id_articolo='id_inserito'
```

Body Request

Non abbiamo parametri all'interno della body request, in questo caso.

Body response (application/json)

200 OK

Chiave	Valore
Id_articolo	Identificativo dell'articolo
Id_categoria	Identificativo della categoria
Categoria	Nome della categoria
Id_sottocategoria	Identificativo della sottocategoria
Sotto_categoriaii	Nome della sottocategoria
Id_reparto	Identificativo del reparto.
Reparto	Nome del reparto
Nome_articolo	Nome dell'articolo
Prezzo	Prezzo dell'articolo
Taglia	Taglia dell'articolo.

404 Not found

Chiave	Valore
Message	L'articolo non esiste.

GET Trova tutti gli articoli

Restituisce tutti gli articoli presenti.

```
api/article/read.php
```

Body Request

Non abbiamo parametri all'interno della body request, in questo caso.

Body response (application/json)

200 OK

Chiave	Valore
Id_articolo	Identificativo dell'articolo
Nome_articolo	Nome dell'articolo
Prezzo	Prezzo dell'articolo
Taglia	Taglia dell'articolo
Id_reparto	Identificativo del reparto
Reparto	Nome del reparto

404 Not found

Chiave	Valore
Message	Nessun articolo trovato.

GET Cerca articolo per nome

Questa operazione restituisce tutti gli articoli che hanno nel nome i caratteri inseriti all'interno del form di ricerca.

```
api/article/search_php?s='input_inserito'
```

Body Request

Non abbiamo parametri all'interno della body request, in questo caso.

Body response (application/json)

200 OK

Chiave	Valore
Id_articolo	Identificativo dell'articolo
Reparto	Nome del reparto
Nome_articolo	Nome dell'articolo
Prezzo	Prezzo dell'articolo
Taglia	Taglia dell'articolo

404 Not found

Chiave	Valore
Message	Nessun articolo trovato.

PUT Sistema articolo

Questa operazione permette di sistemare gli articoli all'interno dei reparti.

api/article/update_arrange.php

Body Request

Chiave	Obbligatorio	Descrizione
Id_articolo	Si	Indica l'identificativo dell'articolo da eliminare
Id_reparto	Si	Indica l'identificativo del reparto
Id_categoria	Si	Identificativo della categoria
Id_sottocategoria	No	Identificativo della sottocategoria
Prezzo	Si	Prezzo dell'articolo

Body response (application/json)

201 CREATED

Chiave	Valore
Message	L'articolo è stato sistemato con successo

503 Service unavailable

Chiave	Valore
Message	Impossibile sistemare l'articolo

400 Bad request

Chiave	Valore
Message	Impossibile sistemare l'articolo. I dati sono incompleti.

PUT Aggiorna dati di un articolo

Questa operazione permette di aggiornare i dati di un articolo.

api/article/update_arrange.php

Body Request

Chiave	Obbligatorio	Descrizione
Id_articolo	Si	Indica l'identificativo dell'articolo da eliminare
Nome_articolo	Si	Nuovo nome dell'articolo
Prezzo	Si	Nuovo prezzo dell'articolo
Taglia	Si	Nuova taglia dell'articolo

Body response (application/json)

200 OK

Chiave	Valore
Message	L'articolo è stato aggiornato

503 Service unavailable

Chiave	Valore
Message	Impossibile aggiornare l'articolo

SOTTOCATEGORIA

GET Prendi sottocategoria

Restituisce tutte le sottocategorie degli articoli presenti.

```
api/category/read.php
```

Body Request

Non abbiamo parametri all'interno della body request, in questo caso.

Body response (application/json)

200 OK

Chiave	Valore
Id_sottocategoria	Identificativo della sottocategoria
sottocategoria	Nome della sottocategoria

404 Not found

Chiave	Valore
Message	Nessuna sottocategoria trovata.

GET Cerca sottocategorie corrispondenti

Restituisce le sottocategorie in base all'id della categoria selezionata.

```
api/category/search.php?s='id_categoria'
```

Body Request

Non abbiamo parametri all'interno della body request, in questo caso.

Body response (application/json)

200 OK

Chiave	Valore
Id_sottocategoria	Identificativo della sottocategoria
sottocategoria	Nome della sottocategoria

404 Not found

Chiave	Valore
Message	Nessuna sottocategoria trovata.

REPARTO

GET Prendi reparti

Restituisce tutti i reparti presenti.

api/department/read.php

Body Request

Non abbiamo parametri all'interno della body request, in questo caso.

Body response (application/json)

200 OK

Chiave	Valore
Id_reparto	Identificativo del reparto
Nome	Nome del reparto

404 Not found

Chiave	Valore
Message	Nessun reparto trovato.

ORDINI

GET Prendi ordini

Restituisce tutti gli ordini presenti con le relative informazioni.

```
api/orders/check.php
```

Body Request

Non abbiamo parametri all'interno della body request, in questo caso.

Body response (application/json)

200 OK

Chiave	Valore
Id_ordine	Identificativo dell'ordine
Id_fornitore	Identificativo del fornitore (se è presente)
Articolo	Nome dell'articolo
Taglia	Taglia dell'articolo
Stato	Stato di avanzamento dell'ordine

404 Not found

Chiave	Valore
Message	Nessun ordine trovato.

GET Prendi singolo ordine

Restituisce tutte le informazioni di un singolo ordine identificato tramite l'id

```
api/orders/read_one.php?id_ordine='id_inserito'
```

Body Request

Non abbiamo parametri all'interno della body request, in questo caso.

Body response (application/json)

200 OK

Chiave	Valore
Id_ordine	Identificativo dell'ordine
Id_fornitore	Identificativo del fornitore (se è presente)
Articolo	Nome dell'articolo
Taglia	Taglia dell'articolo
Stato	Stato di avanzamento dell'ordine

404 Not found

Chiave	Valore
Message	L'ordine non esiste

POST Crea ordine

Inserisce un nuovo ordine con determinati parametri.

api/orders/create.php

Body Request

Chiave	Obbligatorio	Descrizione
Articolo	Si	Indica il nome dell'articolo da ordinare
Quantita	Si	Quantità dell'articolo da ordinare
Taglia	Si	Taglia dell'articolo

Body response (application/json)

201 CREATED

Chiave	Valore
Message	L'ordine è stato creato correttamente.

503 Service Unavailable

Chiave	Valore
Message	Impossibile creare l'ordine.

400 Bad Request

Chiave	Valore
Message	Impossibile creare l'ordine. I dati sono incompleti.

GET Prendi gli ordini da prendere in carico.

Restituisce tutti gli ordini che devono essere presi in carico dal fornitore.

api/orders/read_task.php

Body Request

Non abbiamo parametri all'interno della body request, in questo caso.

Body response (application/json)

200 OK

Chiave	Valore
Id_ordine	Identificativo dell'ordine
Id_fornitore	Identificativo del fornitore (se è presente)
Articolo	Nome dell'articolo
Taglia	Taglia dell'articolo
Stato	Stato di avanzamento dell'ordine

404 Not found

Chiave	Valore
Message	Nessun ordine trovato.

GET Prendi gli ordini da completare.

Restituisce tutti gli ordini che devono essere completati dal fornitore.

api/orders/read_taskc.php

Body Request

Non abbiamo parametri all'interno della body request, in questo caso.

Body response (application/json)

200 OK

Chiave	Valore
Id_ordine	Identificativo dell'ordine
Id_fornitore	Identificativo del fornitore (se è presente)
Articolo	Nome dell'articolo
Taglia	Taglia dell'articolo
Stato	Stato di avanzamento dell'ordine

404 Not found

Chiave	Valore
Message	Nessun ordine trovato.

PUT Prendi incarico.

Permette al fornitore di prendere in carico un ordine.

api/orders/gettask.php

Body Request

Chiave	Obbligatorio	Descrizione
Id_ordine	Si	Identificativo dell'ordine.
Id_fornitore	Si	Identificativo del fornitore

Body response (application/json)

200 OK

Chiave	Valore
Message	L'incarico è stato preso con successo

503 Service unavailable

Chiave	Valore
--------	--------

Message	Impossibile prendere in carico l'ordine.
---------	--

PUT Completa incarico.

Permette al fornitore di completare l'incarico preso precedentemente.

api/orders/complete_task.php

Body Request

Chiave	Obbligatorio	Descrizione
Id_ordine	Si	Identificativo dell'ordine.

Body response (application/json)

200 OK

Chiave	Valore
Message	L'incarico è stato completato con successo

503 Service unavailable

Chiave	Valore
Message	Impossibile completare l'incarico.

PUT Ricezione ordine.

Permette al magazziniere di comunicare l'avvenuta ricezione dell'ordine.

api/orders/complete.php

Body Request

Chiave	Obbligatorio	Descrizione
Id_ordine	Si	Identificativo dell'ordine.

Body response (application/json)

200 OK

Chiave	Valore
Message	L'ordine è stato ricevuto con successo.

503 Service unavailable

Chiave	Valore
Message	Impossibile ricevere l'ordine.

ADDETTI VENDITA

GET Lista degli addetti vendita

Restituisce tutte le informazioni degli addetti vendita registrati.

api/salesman/read.php

Body Request

Non abbiamo parametri all'interno della body request, in questo caso.

Body response (application/json)

200 OK

Chiave	Valore
Id_addetto	Identificativo dell'addetto
Id_reparto	Identificativo del reparto
Reparto	Nome del reparto
Nome	Nome dell'addetto vendita
Cognome	Cognome dell'addetto vendita
Email	Email dell'addetto vendita

404 Not found

Chiave	Valore
Message	Nessun addetto vendita trovato.

GET Prendi informazioni di un addetto vendita

Restituisce tutte le informazioni di un singolo addetto vendita

```
api/salesman/read_one.php?id_addetto='id_inserito'
```

Body Request

Non abbiamo parametri all'interno della body request, in questo caso.

Body response (application/json)

200 OK

Chiave	Valore
Id_addetto	Identificativo dell'addetto
Id_reparto	Identificativo del reparto
Reparto	Nome del reparto
Nome	Nome dell'addetto vendita
Cognome	Cognome dell'addetto vendita
Email	Email dell'addetto vendita

404 Not found

Chiave	Valore
Message	L'addetto vendita non esiste

PUT Aggiorna reparto addetto vendita.

Permette di aggiornare il reparto di un addetto vendita.

```
api/salesman/update.php
```

Body Request

Chiave	Obbligatorio	Descrizione
Id_addetto	Si	Identificativo dell'addetto
Id_reparto	Si	Identificativo del reparto

Body response (application/json)

200 OK

Chiave	Valore
Message	Il reparto è stato aggiornato.

503 Service unavailable

Chiave	Valore
Message	Impossibile aggiornare il reparto.

ORARIO LAVORATIVO

GET Lista degli orari lavorativi

Restituisce tutti gli orari lavorativi dei dipendenti.

api/shifts/read.php

Body Request

Non abbiamo parametri all'interno della body request, in questo caso.

Body response (application/json)

200 OK

Chiave	Valore
Ruolo	Ruolo che ricopre il dipendente
Id_orario_lav	Identificativo dell'orario lavorativo
Nome	Nome del dipendente
Giorno	Giorno della settimana
Orario	Turno lavorativo

404 Not found

Chiave	Valore
Message	Nessun orario trovato.

GET Prendi orario lavorativo di un addetto vendita

Restituisce l'orario lavorativo di un addetto vendita specifico.

```
api/shifts/reads.php?id_addetto='id_inserito'
```

Body Request

Non abbiamo parametri all'interno della body request, in questo caso.

Body response (application/json)

200 OK

Chiave	Valore
Id_addetto	Identificativo dell'addetto vendita
Giorno	Giorno della settimana
Orario	Turno lavorativo

404 Not found

Chiave	Valore
Message	Nessun orario trovato.

GET Prendi orario lavorativo di un magazziniere

Restituisce l'orario lavorativo di un magazziniere specifico.

```
api/shifts/reads.php?id_magazziniere = 'id_inserito'
```

Body Request

Non abbiamo parametri all'interno della body request, in questo caso.

Body response (application/json)

200 OK

Chiave	Valore
Id_magazziniere	Identificativo del magazziniere
Giorno	Giorno della settimana

Orario	Turno lavorativo
--------	------------------

404 Not found

Chiave	Valore
Message	Nessun orario trovato.

GET Prendi informazioni di un singolo orario lavorativo

Restituisce le informazioni di un singolo orario lavorativo

```
api/shifts/read_one.php?id_orario_lav ='id_inserito'
```

Body Request

Non abbiamo parametri all'interno della body request, in questo caso.

Body response (application/json)

200 OK

Chiave	Valore
Id_orario_lav	Identificativo dell'orario lavorativo
Id_giorno	Identificativo del giorno
Giorno	Giorno della settimana
Id_orario	Identificativo del turno
Orario	Turno lavorativo

404 Not found

Chiave	Valore
Message	L'orario non esiste.

POST Crea orario lavorativo (ADDETTO VENDITA)

Inserisce un nuovo orario lavorativo di un addetto vendita

```
api/shifts/creates.php
```

Body Request

Chiave	Obbligatorio	Descrizione
Id_addetto	Si	Identificativo dell'addetto vendita
Id_giorno	Si	Identificativo del giorno
Id_orario	Si	Identificativo dell'orario

Body response (application/json)

201 CREATED

Chiave	Valore
Message	Il turno è stato creato correttamente

503 Service Unavailable

Chiave	Valore
Message	Impossibile creare il turno.

400 Bad Request

Chiave	Valore
Message	Impossibile inserire il turno. I dati sono incompleti.

POST Crea orario lavorativo (MAGAZZINIERE)

Inserisce un nuovo orario lavorativo di un magazziniere

api/shifts/createw.php

Body Request

Chiave	Obbligatorio	Descrizione
Id_magazziniere	Si	Identificativo del magazziniere
Id_giorno	Si	Identificativo del giorno

Id_orario	Si	Identificativo dell'orario
-----------	----	----------------------------

Body response (application/json)

201 CREATED

Chiave	Valore
Message	Il turno è stato creato correttamente

503 Service Unavailable

Chiave	Valore
Message	Impossibile creare il turno.

400 Bad Request

Chiave	Valore
Message	Impossibile inserire il turno. I dati sono incompleti.

PUT Aggiorna orario lavorativo.

Permette di aggiornare giorno e turno di un singolo orario lavorativo.

api/shifts/update.php

Body Request

Chiave	Obbligatorio	Descrizione
Id_orario_lav	Si	Identificativo dell'orario lavorativo
Id_giorno	Si	Identificativo del giorno
Id_orario	Si	Identificativo del turno

Body response (application/json)

200 OK

Chiave	Valore
Message	L'orario è stato aggiornato

503 Service unavailable

Chiave	Valore
Message	Impossibile aggiornare l'orario.

NEGOZIO

GET Lista dei negozi.

Restituisce tutte le informazioni dei negozi presenti.

api/store/read.php

Body Request

Non abbiamo parametri all'interno della body request, in questo caso.

Body response (application/json)

200 OK

Chiave	Valore
Id_negozio	Identificativo del negozio
Nome	Nome del negozio
Via	Via del negozio
Cap	Cap del negozio
Citta	Citta dove è ubicato il negozio

404 Not found

Chiave	Valore
Message	Nessun negozio trovato

GET Trova informazioni di un singolo negozio.

Restituisce tutte le informazioni di un singolo negozio

api/store/read_one.php

Body Request

Non abbiamo parametri all'interno della body request, in questo caso.

Body response (application/json)

200 OK

Chiave	Valore
Id_negozio	Identificativo del negozio
Nome	Nome del negozio
Via	Via del negozio
Cap	Cap del negozio
Citta	Citta dove è ubicato il negozio

404 Not found

Chiave	Valore
Message	Il negozio non esiste.

PUT Aggiorna info negozio.

Permette di aggiornare nome, via, cap e città di un negozio.

api/store/update.php

Body Request

Chiave	Obbligatorio	Descrizione
Id_negozio	Si	Identificativo del negozio
Nome	Si	Nome del negozio
Via	Si	Via del negozio
Cap	Si	Cap del negozio
Citta	Si	Citta dove è ubicato il negozio

Body response (application/json)

200 OK

Chiave	Valore
Message	Le informazioni sono state aggiornate.

503 Service unavailable

Chiave	Valore
Message	Impossibile aggiornare le informazioni.

CATEGORIE

GET Lista delle categorie.

Restituisce la lista delle categorie presenti.

api/tipology/read.php

Body Request

Non abbiamo parametri all'interno della body request, in questo caso.

Body response (application/json)

200 OK

Chiave	Valore
Id_categoria	Identificativo della categoria
Categoria	Nome della categoria

404 Not found

Chiave	Valore
Message	Nessuna categoria trovata

MAGAZZINIERE

GET Lista dei magazzinieri.

Restituisce la lista e le informazioni dei magazzinieri presenti.

```
api/warehouse/read.php
```

Body Request

Non abbiamo parametri all'interno della body request, in questo caso.

Body response (application/json)

200 OK

Chiave	Valore
Id_magazziniere	Identificativo del magazziniere
Nome	Nome del magazziniere
Cognome	Cognome del magazziniere
Email	Email del magazziniere

404 Not found

Chiave	Valore
Message	Nessun magazziniere trovato.

GIORNI LAVORATIVI

GET Lista dei giorni lavorativi.

Restituisce la lista di tutti i giorni lavorativi.

```
api/wdays/read.php
```

Body Request

Non abbiamo parametri all'interno della body request, in questo caso.

Body response (application/json)

200 OK

Chiave	Valore
Id_giorno	Identificativo del giorno
Giorno	Nome del giorno

404 Not found

Chiave	Valore
Message	Nessun giorno trovato.

TURNI LAVORATIVI

GET Lista dei turni lavorativi.

Restituisce la lista di tutti i turni lavorativi.

api/wtime/read.php

Body Request

Non abbiamo parametri all'interno della body request, in questo caso.

Body response (application/json)

200 OK

Chiave	Valore
Id_orario	Identificativo del turno
Orario	Turno lavorativo

404 Not found

Chiave	Valore
Message	Nessun turno trovato.

