

# Simple Sport Manager

Progetto Basi di Dati – Anno accademico 2020/2021

## 1 – Abstract

“Simple Sport Manager” è una piattaforma online nata nell’estate del 2021 in seguito all’uscita globale dalla pandemia da Covid 19. Dopo un forte richiamo alla ripresa delle attività sportive, questa piattaforma si propone come supporto, ad associazioni e a privati possessori di campi sportivi, nella gestione dei propri tesserati, le relative prenotazioni e non solo.

Il servizio offerto dalla piattaforma consiste, dopo previa registrazione, nell’uso di un portale dedicato online, al quale i vari dipendenti potranno connettersi per registrare nuovi tesserati, catalogandoli come atleti o come arbitri, osservare la disponibilità dei campi e gestire i pagamenti relativi sia al noleggio dei campi, sia quelli relativi al compenso degli arbitri e ai pagamenti dei fornitori per l’acquisto delle nuove attrezzature.

Allo stesso tempo offre la possibilità ai tesserati di connettersi alla piattaforma, con apposite credenziali fornite dall’associazione, avendo la possibilità di prenotare un campo direttamente da casa, filtrando per le caratteristiche preferite. Simple Sport Manager, infatti offre la possibilità di registrare molti dati relativi ai campi di cui si ha a disposizione come relative misure, tipo di terreno, tipo di sport, presenza di attrezzature.

## 2 – Analisi dei requisiti

### 2.1 Descrizione testuale dei requisiti

L’obiettivo è quello di realizzare una base di dati per l’organizzazione e il supporto ad associazioni che gestiscono l’affitto di campi sportivi di diverso tipo. Poiché il sistema si basa sul multi-tenant, lo scopo è quello di fornire un portale privato per ogni **Associazione**, e che quest’ultima lo veda come univocamente a lei dedicato. Ovviamente anche la base di dati dovrà essere adatta a una struttura di questo tipo e permettere quindi ad ogni associazione di vedere solo i propri dati, e nascondere quindi quelli relativi alle altre associazioni presenti. Ognuna nella base di dati sarà quindi rappresentata tramite:

- Ragione sociale
- Sito web
- E-mail

Ogni associazione avrà dei tesserati, che dopo essersi registrati sul portale dell’associazione potranno effettuare la prenotazione dei campi disponibili, sia online (tramite una sezione dedicata alle prenotazioni) sia presso la segreteria. Per i tesserati è noto:

- codice fiscale
- nome
- cognome
- data di nascita
- e-mail
- password (insieme alla e-mail forma le credenziali di accesso)
- cellulare

Ogni tesserato inoltre può essere classificato in quanto “atleta” o “arbitro”, visto che l’associazione prevede la possibilità di poter richiedere un arbitro per eventuali partite importanti.

Ogni associazione possiede una o più **Sedi** dislocate sul territorio, ognuna rappresentata tramite:

- indirizzo (via, CAP, città)
- telefono (supponiamo che sia unico per ogni sede e si identifichi quello della segreteria che gestisce le prenotazioni e il pagamento della tariffa)
- nome (ad esempio “Polisportiva Romana”)

In ogni sede sono presenti i vari **Campi** messi a disposizione, ognuno ha la sua **Tipologia** e specifiche (derivanti dal tipo di sport) gestite direttamente dai dipendenti delle associazioni, di cui sono memorizzati:

- larghezza
- lunghezza
- sport di riferimento
- terreno
- attrezzatura (indica se nel campo sono presenti le attrezzature, ad esempio le reti)

Come citato sopra, è possibile gestire anche i **Dipendenti** dell’associazione, che si dividono in una gerarchia abbastanza elementare ma che rispecchia nella sua essenzialità i compiti principali che devono essere svolti in un sistema di questo tipo. La gerarchia è composta da **Segreteria**, **Responsabile** e **Amministrazione**. La segreteria si occuperà di gestire tutti pagamenti che riguardano i tesserati, sia quelli relativi alle tariffe dei campi sia quelli relativi al compenso degli arbitri. Il responsabile avrà un’area apposita nel portale in cui potrà registrare e aggiornare lo stato dei campi (ad esempio dopo l’arrivo della nuova attrezzatura) e si occupa di gestire i pagamenti verso i fornitori. L’amministrazione invece si occupa di monitorare l’operato dei soggetti sottostanti, e tramite il portale svolge i compiti delle risorse umane (assunzione, inserimento nel sistema e aggiornamento dei dipendenti e del loro stato) e si occupa di registrare l’erogazione degli stipendi.

Ogni dipendente lavora presso un’unica sede dell’associazione, e sono noti i seguenti dati:

- codice fiscale
- nome
- cognome
- e-mail
- password (insieme all’e-mail compongono le sue credenziali di accesso al portale)
- data di nascita
- telefono

Poiché ogni associazione ha bisogno di ammodernare la propria attrezzatura, è possibile mantenere un registro dei **Fornitori**, con i quali si mantiene un rapporto lavorativo stabile, caratterizzati da:

- partita iva
- nome
- e-mail
- telefono

Viste le diverse attività finanziarie presenti, il portale vuole poter offrire alle proprie associazioni la possibilità di visualizzare un estratto conto e di tenere in questo modo, traccia di tutte le spese in uscita e in entrata della propria realtà. Per una maggiore sicurezza le varie registrazioni delle transazioni sono tracciate dall’identificativo del dipendente incaricato. Per ogni **Pagamento** è noto:

- importo
- data

## 2.2 Glossario dei termini

Termine	Descrizione	Collegamenti
Associazione	È il privato o l'ente che vuole mettere a disposizione i campi in suo possesso e si registra sul portale.	Tesserato, Fornitore, Sede, Dipendente
Tesserato	Atleta o Arbitro che vengono tesserati dall'associazione e possono prenotare i campi sportivi.	Entità padre di Atleta e Arbitro, collegamento con: Associazione, Campo, Fattura, Segreteria
Atleta	È un tesserato. Dopo essere stato registrato può prenotare i campi sportivi.	
Arbitro	È un tesserato e può essere chiamato per svolgere il ruolo di arbitro, e inoltre in quanto tesserato può anche prenotare i campi sportivi.	Fattura, Segreteria
Sede	Sede fisica in cui si trovano i campi sportivi dell'associazione.	Associazione, Campo, Dipendente
Campo	Campo sportivo che i tesserati delle associazioni possono prenotare.	Sede, Tesserato
Pagamento	Registro dei vari pagamenti.	Entità padre di Stipendio, Esborso, Fattura; collegamento con Dipendente
Fattura	Registro di tutti i pagamenti correlati ai tesserati, che siano compensi degli arbitri o pagamenti di prenotazioni in generale.	Arbitro, Tesserato, Segreteria
Esborso	Registro dei pagamenti effettuati ai fornitori.	Fornitore, Responsabile
Stipendio	Registro degli stipendi erogati ai dipendenti.	Dipendente, Amministrazione
Fornitore	Azienda, società o altro ente che si occupa della vendita di attrezzature sportive e con il quale una o più associazioni hanno stretto un accordo.	Esborso, Associazione, Responsabile
Dipendente	Impiegato generico di un'associazione.	Pagamento, Stipendio, Amministrazione, Sede, Associazione
Amministrazione	Impiegato al vertice della gerarchia che si occupa di assumere impiegati, di registrarli e gestire il loro operato.	Stipendio, Dipendente
Responsabile	Impiegato che si occupa della manutenzione dei campi sportivi dentro alle varie sedi e registra i pagamenti verso i fornitori.	Esborso, Fornitore
Segreteria	Impiegato che si occupa di registrare i pagamenti dei tesserati relativi alle prenotazioni dei campi.	Fattura, Arbitro, Tesserato

## 3 – Progettazione concettuale

### 3.1 Lista delle Entità

Tutti gli attributi sono “NOT NULL” a parte quelli specificati.

Associazione		
<u>Codice</u>	varchar(20)	Codice identificativo univoco.
Ragione Sociale	varchar(80)	Nome dell'associazione.
Sito Web	varchar(150)	Sito web dell'associazione.
E-mail	varchar(80)	E-mail di riferimento dell'associazione (per utilizzare la piattaforma i dipendenti dovranno usare la loro e-mail aziendale e non la suddetta).

Tesserato – Generalizzazione Totale		
<u>Codice Fiscale (CF)</u>	char(16)	Codice fiscale del tesserato.
Nome	varchar(80)	Nome del tesserato.
Cognome	varchar(80)	Cognome del tesserato.
Data di Nascita	date	Data di nascita del tesserato.
E-mail	varchar(80)	E-mail del tesserato (utilizzata al momento della registrazione alla piattaforma).
Password	varchar(50)	Password associata al tesserato, fornitagli al momento della registrazione via E-mail.
Telefono	varchar(12)	Numero di telefono del tesserato.
<b>ENTITA' FIGLIE</b>	<ul style="list-style-type: none"> <li>• <b>Atleta</b> → può effettuare la prenotazione di campi.</li> <li>• <b>Arbitro</b> → Oltre a poter effettuare la prenotazione di campi, può arbitrare le partite e quindi ricevere un compenso da parte dell'associazione.</li> </ul>	

Sede		
<u>Codice</u>	int	Codice univoco della sede.
Indirizzo	varchar(150)	Indirizzo della sede. È un attributo <b>composto</b> che identifica Via e Civico.
Città	varchar(150)	Città in cui la sede è situata. È un attributo <b>composto</b> che identifica CAP, nome della città, provincia e regione in cui si trova.
Nome	varchar(150)	Nome della sede.
Telefono	varchar(12)	Numero telefonico di riferimento della sede (Può essere <u>NULL</u> ).

Fornitore		
<u>P.IVA</u>	char(11)	Partita IVA del fornitore (univoca).
Ragione Sociale	varchar(150)	Ragione Sociale del fornitore.
E-mail	varchar(80)	E-mail di contatto del fornitore.
Telefono	varchar(12)	Numero di telefono del fornitore (Può essere <u>NULL</u> ).

Campo		
<u>ID</u>	int	Numero identificativo progressivo del campo.
Attrezzatura	bool	Se uguale a TRUE significa che il campo è dotato di attrezzatura.
Tipologia	int	È un attributo <b>composto</b> che descrive il campo da gioco. Comprende i sotto-attributi "Lunghezza" e "Larghezza", "Sport" (ossia lo sport al quale il campo è adibito) e "Terreno".

Dipendente – Generalizzazione Totale		
<u>Codice Fiscale (CF)</u>	char(16)	Codice fiscale del dipendente.
Nome	varchar(80)	Nome del dipendente.
Cognome	varchar(80)	Cognome del dipendente.

Data di Nascita	date	Data di nascita del dipendente.
E-mail	varchar(80)	E-mail associata al dipendente (dovrà utilizzarla al momento del login sulla piattaforma).
Password	varchar(50)	Password del dipendente, utilizzata assieme alla E-mail durante il login.
Telefono	varchar(12)	Numero di telefono del dipendente (Può essere NULL).
<b>ENTITA' FIGLIE</b>	<ul style="list-style-type: none"> <li>• <b>Amministrazione</b> → Gestisce gli stipendi di tutti i dipendenti.</li> <li>• <b>Responsabile</b> → Si occupa dei pagamenti ai fornitori.</li> <li>• <b>Segreteria</b> → Gestisce i pagamenti delle prenotazioni da parte dei tesserati e i compensi degli arbitri.</li> </ul>	

Pagamento – Generalizzazione Totale		
Data	int	Data in cui il pagamento è stato effettuato.
Importo	money	Importo del pagamento; se è positivo allora sarà un'entrata (dal punto di vista dell'associazione), viceversa se è negativo.
<b>ENTITA' FIGLIE</b>	<ul style="list-style-type: none"> <li>• <b>Stipendio</b> → È un pagamento (da parte dell'associazione) verso i suoi dipendenti, gestito dall'amministrazione.</li> <li>• <b>Esborso</b> → È un pagamento (da parte dell'associazione) verso i suoi fornitori, gestito dai responsabili.</li> <li>• <b>Fattura</b> → È un pagamento che può essere un'entrata derivante dalle prenotazioni dei tesserati, o un'uscita verso gli arbitri (compenso).</li> </ul>	

### 3.2 Lista delle Relazioni e cardinalità

#### Associazione - Tesserato: Iscrizione

- Un'associazione può registrare più tesserati (anche nessuno) → (0, N)
- Un tesserato è iscritto a una sola associazione (\*) → (1, 1)

(\*) L'entità "Tesserato" è stata considerata come se rappresentasse un tesserino, quindi una persona può iscriversi a più associazioni ottenendo più "tesserini". In questo modo viene rispettato il modello multi-tenant, ed è facilitato un futuro inserimento delle anagrafiche, infatti le insert non saranno vincolate da una tabella con tutte le persone.

#### Tesserato - Campo: Prenotazione

- Un tesserato può prenotare più campi (anche nessuno) → (0, N)
- Un campo può essere prenotato da più tesserati (anche nessuno) → (0, N)

#### Tesserato - Campo: Storico Prenotazioni (Prenotazioni antecedenti la data attuale)

- Un tesserato può aver prenotato più campi (anche nessuno) → (0, N)
- Un campo può essere stato prenotato in passato da più tesserati (anche nessuno) → (0, N)

#### Campo - Sede: Posizione

- Un campo si può trovare solamente in una sede → (1, 1)
- Una sede può gestire più campi (anche nessuno) → (0, N)

#### Sede - Associazione: Locazione

- Una sede corrisponde solamente ad un'associazione → (1, 1)
- Un'associazione può amministrare più sedi (anche nessuna) → (0, N)

#### Sede - Dipendente: Postazione

- In una sede lavorano più dipendenti (anche nessuno) → (0, N)
- Un dipendente lavora presso un'unica sede → (1, 1)

#### Associazione - Dipendente: Assunzione

- Un'associazione può assumere più dipendenti (anche nessuno) → (0, N)
- Un dipendente può lavorare solamente in una sede → (1, 1)

### Associazione - Fornitore: Contratto

- Un'associazione può firmare contratti con più fornitori (anche nessuno)  $\rightarrow (0, N)$
- Lo stesso fornitore può lavorare con più associazioni (almeno una)  $\rightarrow (1, N)$

### Dipendente - Pagamento: Gestione

- Un dipendente può gestire più pagamenti (anche nessuno)  $\rightarrow (0, N)$
- Un pagamento è gestito da uno e un solo dipendente  $\rightarrow (1, 1)$

### Arbitro - Fattura - Segreteria: Compenso

- Un arbitro può ricevere più pagamenti (anche nessuno)  $\rightarrow (0, N)$
- Una fattura individua solamente un arbitro (anche nessuno)  $\rightarrow (0, 1)$
- La segreteria può gestire più compensi (anche nessuno)  $\rightarrow (0, N)$

### Tesserato - Fattura - Segreteria: Tariffa

- Un tesserato può pagare più prenotazioni (anche nessuna)  $\rightarrow (0, N)$
- Una fattura individua solamente un tesserato (anche nessuno)  $\rightarrow (0, 1)$
- La segreteria può gestire più tariffe (anche nessuna)  $\rightarrow (0, N)$

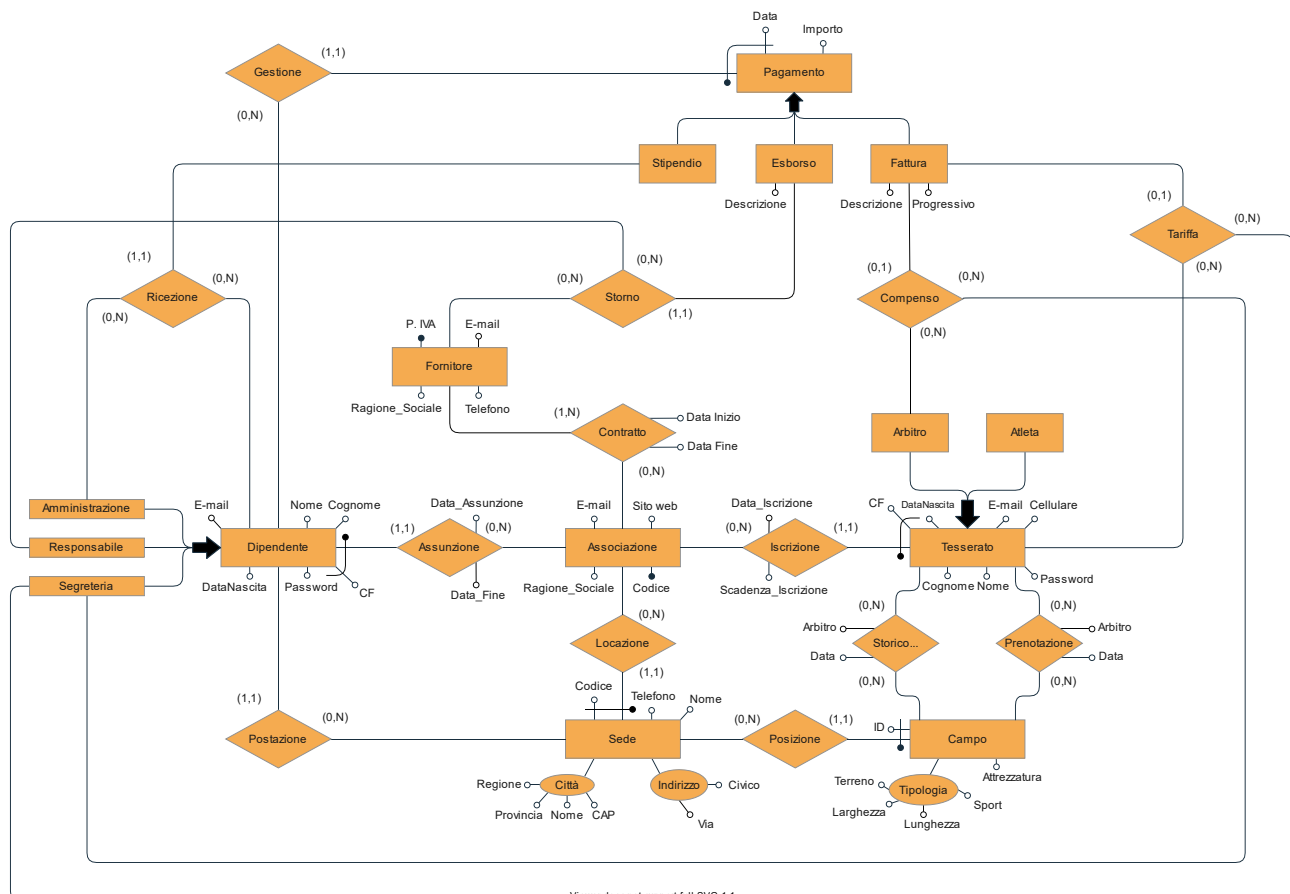
### Fornitore - Esborso - Responsabile: Storno

- A un fornitore si possono registrare più esborsi (anche nessuno)  $\rightarrow (0, N)$
- Un esborso registrato individua solamente un fornitore  $\rightarrow (1, 1)$
- I responsabili possono gestire più esborsi (anche nessuno)  $\rightarrow (0, N)$

### Dipendente - Stipendio - Amministrazione: Ricezione

- Un dipendente può ricevere più stipendi (anche nessuno)  $\rightarrow (0, N)$
- Uno stipendio registrato individua solamente un dipendente  $\rightarrow (1, 1)$
- La segreteria può gestire più stipendi (anche nessuno)  $\rightarrow (0, N)$

## 3.3 Diagramma del modello ER



## 4 – Progettazione logica

### 4.1 Ristrutturazione dello schema

#### 4.1.1 Analisi delle ridondanze

##### Tipologia Campo

Ad ogni campo sportivo vengono associati delle specifiche relative a sport di riferimento, lunghezza, larghezza e tipo di terreno. Tuttavia, i campi sportivi, nella maggior parte dei casi, rispettano le misure indicate dalle federazioni competenti (FIBA, FIPAV, ecc), per lo svolgimento di attività sportive agonistiche e dilettantistiche. Quindi indicare questi attributi all'interno dell'entità "**Campo**" porterebbe sicuramente a grandi ridondanze dovute alla ripetizione delle misure (in molti casi identiche) e alla ripetizione dell'attributo sport per ogni campo registrato.

Nei regolamenti stessi, si fa riferimento alla possibilità per un campo di avere dimensioni ridotte da quelle dichiarate, dovute a vincoli fisici e non aggirabili.

Vista la presenza di casi di questo tipo si è pensato di risolvere lasciando completa libertà alle associazioni, saranno quindi quest'ultime a registrare le misure dei campi a loro disposizione, indicando lo sport di riferimento e il tipo di terreno che presentano. Verrà quindi creata comunque un'ulteriore entità "**Tipologia Campo**", in relazione con "Associazioni".

##### Sede – Città

La "Sede" svolge un ruolo abbastanza rilevante nella struttura della base di dati e in particolare nel contesto in cui verrà utilizzata. Infatti, la previsione è che verrà molto sfruttata dagli utenti per il filtraggio e l'individuazione delle sedi di interesse. La soluzione scelta, quindi, è quella di gestirla in modo più robusto creando l'entità "**Città**" in cui sono registrati tutti i principali comuni italiani.

In questo modo, dato che in una "città" potranno esserci più "sedi" (anche della stessa "associazione") evitiamo di ripetere molte volte diversi **attributi** che altrimenti avrebbero creato ridondanza all'interno dell'entità sede (ogni volta, ad esempio, si ripeterebbero gli attributi CAP, nome della città, provincia e regione).

##### Dipendente – Pagamenti

Il vincolo relativo ai compiti svolti da ogni dipendente di un'associazione e di cui è incaricato, non può essere controllato dalla base di dati, sarà gestito quindi dal lato software tramite una corretta impostazione dei privilegi e l'uso dell'entità grado (vedi sezione 4.1.3).

Per questo motivo, le relazioni **storno**, **ricevere**, **tariffa** e **compenso** non hanno più motivo di essere ternarie, visto che il tracciamento del dipendente che ha svolto l'operazione è ottenibile tramite la relazione "**Gestione**".

#### 4.1.2 Eliminazione delle generalizzazioni

Generalizzazioni	Risoluzione
Tesserato ← Arbitro, Atleta	Le due entità arbitro e atleta vengono <b>accorpate</b> nel padre, visto che gli accessi sono contestuali e possono essere distinti esclusivamente dalla presenza di un campo che indica se un tesserato è un arbitro oppure no.
Dipendente ← Amministrazione, Responsabile, Segreteria	Le tre entità vengono <b>accorpate</b> nel padre "Dipendente", tramite il campo "grado". Ad ogni ruolo verrà assegnato un valore corrispondente tramite l'entità omonima "Grado".

Pagamento ← Stipendio, Esborso, Fattura	<p>Le tre entità figlie vengono <b>collegate</b> al padre (sostituzione delle generalizzazioni con relative relazioni), quindi avremo quattro entità. Questo permette di facilitare il riconoscimento dell'entità del <b>soggetto del pagamento</b> (tesserato, dipendente, fornitore). Dentro l'entità "Fatture" vengono accorpati i pagamenti di tutti i tesserati, sia relativi al compenso degli arbitri, sia quelli delle prenotazioni dei campi (o altre spese generiche, come ad esempio una futura possibilità di acquisto di gadget da parte di un tesserato), visto che individuano sempre come soggetto un tesserato. Per poter tenere traccia dei motivi dovuti alle diverse transazioni abbiamo aggiunto un campo "descrizione".</p> <p>Lo stesso campo è stato utilizzato anche per l'"Esborso", in cui è possibile indicare l'attrezzatura acquistata da un responsabile. L'importo di ogni pagamento rimane fedele al principio di "entrata-uscita" che si contraddistingue dal segno del campo. In questo modo vengono distinti i compensi relativi agli arbitri e tutte le entrate provenienti dai tesserati.</p> <p>Le nuove relazioni che si creano tra "Pagamento" e i tre (ex) figli sono "<b>P-S</b>" ("Stipendio"), "<b>P-E</b>" ("Esborso") e "<b>P-F</b>" ("Fattura"). Per ognuna di queste, le cardinalità sono le seguenti:</p> <ul style="list-style-type: none"> <li>• Un pagamento può contenere un P-* (anche nessuno) → (0, 1)</li> <li>• Un P-* identifica uno e un solo pagamento → (1, 1)</li> </ul>
---	--

#### 4.1.3 Partizionamento/Accorpamento di Entità e Relazioni

##### Città e Indirizzo

Come abbiamo analizzato precedentemente, è conveniente **scomporre** gli attributi composti in entità a sé stanti, in modo tale da poter ridurre al minimo la ridondanza.

Per quanto riguarda l'entità "Sede", l'attributo composto "**città**" verrà trasformato appunto in **entità**, ereditando gli attributi "CAP", "nome", "provincia" e "regione". La relazione che collegherà le entità "Città" e "Sede" sarà quindi "**Luogo**":

- In una città potranno esserci più sedi di una stessa associazione (anche nessuna) → (0, N)
- Una sede si trova in una sola città → (1, 1)

L'attributo composto "**indirizzo**" invece verrà semplicemente scomposto nei sotto-attributi "via" e "numero civico", dato che le ridondanze sono minime e verrà quindi accorpati in "Sede".

##### Tipologia Campo

Nell'entità "Campo", l'attributo composto "**tipologia**" verrà trasformato in **entità**.

Erederà gli attributi "terreno", "sport", "lunghezza" e "larghezza". La nuova relazione che collegherà l'entità "Campo" con "Tipologia" sarà "**Appartenere**":

- Un campo può appartenere ad una e una sola tipologia → (1, 1)
- Una tipologia può categorizzare più campi (anche nessuno) → (0, N)

Dato però che ogni "associazione" dovrà essere in grado di gestire le proprie "tipologie" personalizzate (e non dovrà accedere a quelle delle altre associazioni) è necessaria una relazione che colleghi le due entità. Questa relazione si chiama "**Registrazione**":

- Un'associazione potrà registrare più tipologie (anche nessuna) → (0, N) (\*)
- Una tipologia campo verrà associata a una sola associazione → (1, 1)



(\*) la scelta della cardinalità (0,N) per questa relazione, è stata la stessa adottata anche per le altre entità collegate ad “associazione”, per raffigurare il caso in cui un’associazione decida di registrarsi per ottenere il portale ma non abbia ancora registrato né le proprie sedi e nemmeno le tipologie di campi possedute. Successivamente l’associazione dovrà registrare prima quest’ultime per poi inserire i campi sportivi.

## Grado

Per specificare il ruolo di un dipendente si è optato per la scelta di utilizzare una nuova entità, con un relativo grado indicato da un valore intero, che permetta una gestione più flessibile dei privilegi sul portale usato dai dipendenti (ad esempio un dipendente non potrà apportare modifiche al profilo di un suo superiore o suo pari).

Inoltre, è una soluzione scalabile e robusta rispetto a nuovi ruoli, infatti basterà indicare la descrizione e il relativo grado (tramite un valore intero) nella gerarchia.

Tra “Dipendente” e “Grado” si crea la relazione “**Ruolo**”:

- Un dipendente ha un solo ruolo/grado  $\rightarrow (1, 1)$
- Un grado individua più dipendenti (anche nessuno)  $\rightarrow (0, N)$

## Tesserato

L’entità “tesserato” (che rappresenta l’iscrizione di una persona ad una associazione) nel modello logico ingloba gli attributi della relazione “Iscrizione” presenti nel modello ER (ossia “data iscrizione” e “data scadenza”). Questo perché la previsione è che una persona non si iscriverà a **molte** associazioni (al massimo 2-3, in rari casi): ciò porta ad evitare l’utilizzo sia della cardinalità “molti a molti”, sia di una conseguente tabella adibita esclusivamente alle iscrizioni. In questo modo, con un singolo accesso si può accedere sia ai dati dell’iscrizione sia ai dati anagrafici, visto che questo comporta solo l’aggiunta di tre campi all’interno di “Tesserato”.

Ipotizzando di creare una tabella “Iscrizione”, collegando un tesserato a un’associazione, ad ogni inserimento bisognerebbe controllare che la persona non sia presente nella tabella “Tesserato” (cosa che in un ipotetico utilizzo reale non avrebbe senso, visto che porterebbe a una query in più in una tabella non banale nelle dimensioni: un nuovo iscritto viene inserito **senza effettuare controlli**) e nel caso contrario si sarebbe dovuto prima inserire una tupla in “Tesserato” e successivamente collegarlo a un’associazione tramite la tabella “Iscrizione”.

Dando peso a questa implicazione, la soluzione è stata quella di accorpare le date in “Tesserato” senza utilizzare la tabella “Iscrizione”.

## Dipendente

Per quanto riguarda la relazione “**Assunzione**” tra “Dipendente” e “Associazione” il ragionamento è analogo a quello appena descritto per “Tesserato”: in questo caso è ancora più improbabile il fatto che un dipendente si licenzi da un’associazione per poi venire assunto in un’altra; viene considerato, ma dato che la probabilità è molto bassa l’impatto sulle ridondanze è minimo. Per questo anche qui si è deciso di accorpare “data assunzione” e “data licenziamento” in “Dipendente”. La tupla relativa a un dipendente verrà comunque identificata univocamente dalla relazione con “Associazione”.

## Contratto

La gestione dei “fornitori” è stata trattata diversamente rispetto alle relazioni tra “associazione” e le altre entità. Questo perché il numero di fornitori (per le attrezzature dei campi da gioco) di un’associazione sarà sensibilmente inferiore al numero di istanze delle altre entità (ad esempio “tesserini” o “dipendenti”). Inoltre, le cardinalità differiscono, infatti la relazione “**Contratto**” prevede che:

- Un fornitore sia accessibile da più associazioni  $\rightarrow (1, N)$
- Un’associazione firmi contratti con più fornitori (anche nessuno)  $\rightarrow (0, N)$



## 5 – Query

### Query 1

La query mostra l'estratto conto annuale (anno 2020) e quello dell'anno precedente di tutte le associazioni. Inoltre, indica se è stato un anno positivo o negativo, e calcola la percentuale di crescita.

```
DROP VIEW IF EXISTS saldo_annuale;
CREATE VIEW saldo_annuale AS
  SELECT codass, sum(importo) as saldo
  FROM pagamento P
  WHERE extract(year from P.data) = (extract(year from CURRENT_DATE)-1)
  GROUP BY codass;

DROP VIEW IF EXISTS saldo_anno_prec;
CREATE VIEW saldo_anno_prec AS
  SELECT codass, sum(importo) as saldo
  FROM pagamento P
  WHERE extract(year from P.data) = (extract(year from CURRENT_DATE)-2)
  GROUP BY codass;

SELECT A.codice, A.ragsoc, SA.saldo as "Saldo Anno Corrente", SP.saldo as "Saldo Anno Precedente",
CASE
  WHEN SA.saldo > SP.saldo THEN 'POSITIVO'
  WHEN SA.saldo = SP.saldo THEN 'PARI'
  WHEN SA.saldo IS NULL OR SP.saldo IS NULL THEN 'non disponibile'
  ELSE 'NEGATIVO'
END AS Stato,
CASE
  WHEN SA.saldo > SP.saldo AND SP.saldo > 0::money THEN CONCAT(' ', ROUND((((SA.saldo-SP.saldo)/SP.saldo)*100)::numeric, 2))
  WHEN SA.saldo > SP.saldo AND SP.saldo < 0::money THEN CONCAT(' ', ROUND((((SA.saldo-SP.saldo)/SP.saldo)*100)::numeric, 2)*-1)
  WHEN SP.saldo > SA.saldo AND SA.saldo > 0::money THEN CONCAT(' ', ROUND((((SA.saldo-SP.saldo)/SP.saldo)*100)::numeric, 2))
  WHEN SA.saldo IS NULL OR SP.saldo IS NULL THEN 'non calcolabile'
  ELSE CONCAT(' ', ROUND((((SA.saldo-SP.saldo)/SP.saldo)*100)::numeric, 2))
END AS Percentuale
FROM associazione A
LEFT JOIN saldo_annuale as SA ON SA.codass = A.codice
LEFT JOIN saldo_anno_prec as SP ON SP.codass = A.codice
GROUP BY A.codice, A.ragsoc, SA.saldo, SP.saldo
```

	codice character varying (20)	ragsoc character varying (80)	Saldo Anno Corrente money	Saldo Anno Precedente money	stato text	percentuale text
1	CAME	Calciatori Mestrini	380,00 €	335,00 €	POSITIVO	+13.43
2	JSDB	Jesolo San Donà Basket	105,00 €	[null]	non disponibile	non calcolabile
3	POLRM	Polisportiva Romana	390,00 €	440,00 €	NEGATIVO	-11.36
4	TCPG	Tennis Club Portogruaro	-3.166,00 €	-1.880,05 €	NEGATIVO	-68.40

### Query 2

La query mostra per ogni associazione, le sedi che hanno registrato il maggior numero di prenotazioni lo scorso anno e ne indica la media delle prenotazioni mensili.

Questa query risulta utile a comprendere il carico che la base di dati dovrà supportare in un anno (ad esempio prevedendo come caso limite, che tutte le sedi facciano gli stessi inserimenti della principale).

```
DROP VIEW IF EXISTS prenotazioni_per_sede;
CREATE VIEW prenotazioni_per_sede AS
  SELECT P.codass, P.sede as cod_sede, count(P.sede) as num, ROUND(count(P.sede)/12.0,2) as prenotazioni_mensili
  FROM prenotazioni P
  JOIN Sede S ON P.sede = S.codice AND P.codass = S.codass
  JOIN Associazione A ON A.codice = S.codass
  JOIN Citta C ON C.istat = S.cod_citta
  WHERE extract(year from P.data) = (extract(year from CURRENT_DATE)-1)
  GROUP BY P.codass, P.sede;

SELECT A.ragsoc as associazione, S.nome as nome_sede, C.nome as citta, S.via, M.max as prenotazioni_totali, prenotazioni_mensili
FROM prenotazioni_per_sede P
JOIN (SELECT codass, max(num) as max
      FROM prenotazioni_per_sede
      group by codass) M ON P.codass = M.codass AND num = max
JOIN Associazione A ON A.codice = P.codass
JOIN Sede S ON S.codass = P.codass AND S.codice = cod_sede
JOIN Citta C ON C.istat = S.cod_citta;
```

	associazione character varying (80)	nome_sede character varying (150)	citta character varying (100)	via character varying (150)	prenotazioni_totali bigint	prenotazioni_mensili numeric
1	Calcianti Mestrini	Sede Pulcini Calcio Mestre	Venezia	Via E.Ponti	11	0.92
2	Jesolo San Donà Basket	Jesolo San Donà Basket	San Donà di Piave	Via Iseo	10	0.83
3	Polisportiva Romana	Polisportiva Roma (Tivoli)	Tivoli	Viale Tevere	95	7.92
4	Tennis Club Portogruaro	Tennis Club Portogruaro	Portogruaro	Via Alberti	7	0.58
5	Tennis Club Portogruaro	Tennis Club Portogruaro (Port...	Portogruaro	Via Verdi	7	0.58

### Query 3

La Polisportiva Romana (codice POLRM) vuole organizzare un evento calcistico per i suoi tesserati. Deve decidere quale sede, campo e fascia oraria siano i più adatti per organizzare l'evento.

Nella query si indica il nome della sede, il numero del campo, il relativo terreno e il numero di prenotazioni totali in cui compare nelle due fasce orarie mattino (dalle 8 alle 12) e pomeriggio (dalle 13 alle 21).

```

DROP VIEW IF EXISTS utilizzo_campi_pomeriggio;
CREATE VIEW utilizzo_campi_pomeriggio AS
SELECT p.codass, p.sede, p.id_campo, count(*) as tot_p_pomeriggio
FROM prenotazioni p
JOIN campo c ON c.codass = p.codass AND c.id = p.id_campo
WHERE date_part('hour', p.data) between 13 AND 21
GROUP BY p.codass, p.id_campo, p.sede
ORDER BY p.codass, p.sede;

DROP VIEW IF EXISTS utilizzo_campi_mattino;
CREATE VIEW utilizzo_campi_mattino AS
SELECT p.codass, p.sede, p.id_campo, count(*) as tot_p_mattino
FROM prenotazioni p
JOIN campo c ON c.codass = p.codass AND c.id = p.id_campo
WHERE date_part('hour', p.data) between 8 AND 12
GROUP BY p.codass, p.id_campo, p.sede
ORDER BY p.codass, p.sede;

SELECT s.nome as nome_sede, c.id as num_campo, t.sport, t.terreno, tot_p_mattino, tot_p_pomeriggio
FROM campo c
LEFT JOIN tipologia_campo t
ON t.codass = c.codass AND t.id = c.tipologia
LEFT JOIN sede s
ON s.codass = c.codass AND s.codice = c.cod_sede
LEFT JOIN utilizzo_campi_pomeriggio ucp
ON ucp.codass = c.codass AND ucp.sede = c.cod_sede AND ucp.id_campo = c.id
LEFT JOIN utilizzo_campi_mattino ucm
ON ucm.codass = c.codass AND ucm.sede = c.cod_sede AND ucm.id_campo = c.id
WHERE c.codass = 'POLRM' AND c.attrezzatura AND t.sport like '_alcio%'
AND (tot_p_mattino IS NOT NULL OR tot_p_pomeriggio IS NOT NULL)

```

	nome_sede character varying (150)	num_campo integer	sport character varying (50)	terreno character varying (50)	tot_p_mattino bigint	tot_p_pomeriggio bigint
1	Polisportiva Roma (Tivoli)		1 Calcio	erba	3	14
2	Polisportiva Roma (Tivoli)		2 Calcio 5	gomma	1	4
3	Polisportiva Roma (Tivoli)		3 Calcio 5	erba sintetica	2	20
4	Polisportiva Roma (Tivoli)		4 Calcio 7	erba sintetica	1	31
5	Polisportiva Roma (Tivoli)		5 Calcio 8	erba	1	9

### Query 4

La query mostra il saldo delle sedi dell'associazione dei Calcianti Mestrini (codice CAME), dipendenti attualmente attivi e il totale delle prenotazioni relative a quella sede nell'anno precedente.

```

SELECT s.nome as nome_sede, sum(importo) as saldo, attivi as dipendenti_attivi, prenotazioni_anno
FROM sede s
LEFT JOIN dipendente d ON d.codass = s.codass AND d.cod_sede = s.codice
LEFT JOIN pagamento p ON p.codass = s.codass AND p.id_dipendente = d.cf
LEFT JOIN (SELECT codass, cod_sede, count(*) as attivi
            FROM dipendente d
            WHERE data_fine IS NULL
            GROUP BY codass, cod_sede
            ORDER BY codass, cod_sede) as ta ON ta.codass = s.codass AND ta.cod_sede = s.codice
LEFT JOIN (SELECT codass, sede, count(*) as prenotazioni_anno
            FROM prenotazioni
            WHERE extract(year from data) = extract(year from CURRENT_DATE)-1
            GROUP BY codass, sede) as pr ON pr.codass = s.codass AND pr.sede = s.codice
WHERE
s.codass = 'CAME' AND
(extract(year from p.data) = extract(year from CURRENT_DATE)-1 or importo is null)
GROUP BY s.codice, s.nome, s.codass, attivi, prenotazioni_anno

```

	nome_sede	saldo	dipendenti_attivi	prenotazioni_anno
	character varying (150)	money	bigint	bigint
1	Sede Pulcini Calcio Mestre	285,00 €	4	11
2	Sede Calcio Mestre	95,00 €	4	3
3	Sede Calcio Mestre (Nuova)	[null]	[null]	[null]

## Query 5

Un tesserato della Polisportiva Romana vuole prenotare un campo in data 20/05/2020. La query, quindi, mostra i campi disponibili presso tutte le sedi della Polisportiva Romana (codice POLRM) in data 20/05/2020 filtrato per la fascia oraria dalle 13:30 alle 21:30 e nel caso ci fossero prenotazioni pendenti su quel campo indicare quando è occupato.

```

SELECT s.nome as nome_sede, s.via, s.cod_civico, up.id as num_campo, t.sport, t.terreno,
CASE
    WHEN up.da IS NULL THEN 'DISPONIBILE'
    ELSE 'OCCUPATO'
END as stato,
to_char(up.da, 'HH24:MI:SS') as da, to_char(up.a, 'HH24:MI:SS') as a
FROM sede s
JOIN associazione a
ON a.codice = s.codass
JOIN ((SELECT c.codass, c.cod_sede, c.id, c.tipologia, NULL as da, NULL as a
        FROM campo c
        LEFT JOIN prenotazioni p ON p.codass = c.codass AND p.id_campo = c.id AND p.sede = c.cod_sede
        WHERE (c.codass, c.id) NOT IN (
            SELECT DISTINCT c.codass, id
            FROM campo c
            LEFT JOIN prenotazioni p ON p.codass = c.codass AND p.sede = c.cod_sede AND p.id_campo = c.id
            WHERE data between '2020-5-20 13:30' AND '2020-05-20 21:30'
        )
        GROUP BY c.cod_sede, c.id, c.codass, c.tipologia)
union
(SELECT c.codass, c.cod_sede, c.id, c.tipologia, data as da, data + (ore * INTERVAL '1 hour') as a
FROM campo c
JOIN prenotazioni p ON p.codass = c.codass AND p.sede = c.cod_sede AND p.id_campo = c.id
WHERE data between '2020-5-20 13:30' AND '2020-05-20 21:30'
GROUP BY c.codass, c.id, c.cod_sede, c.tipologia, data, ore)) as up
ON up.codass = s.codass AND up.cod_sede = s.codice
JOIN tipologia_campo t
ON t.codass = a.codice AND t.id = up.tipologia
WHERE a.codice = 'POLRM'
ORDER BY a.codice, s.codice, s.nome, up.id, up.da

```

	nome_sede character varying (150)	via character varying (150)	cod_civico integer	num_campo integer	sport character varying (50)	terreno character varying (50)	stato text	da text	a text
1	Polisportiva Roma (Tivoli)	Viale Tevere	96	1	Calcio	erba	OCCUPATO	18:00:00	20:00:00
2	Polisportiva Roma (Tivoli)	Viale Tevere	96	2	Calcio 5	gomma	DISPONIBILE	[null]	[null]
3	Polisportiva Roma (Tivoli)	Viale Tevere	96	3	Calcio 5	erba sintetica	OCCUPATO	16:30:00	18:30:00
4	Polisportiva Roma (Tivoli)	Viale Tevere	96	3	Calcio 5	erba sintetica	OCCUPATO	18:30:00	19:30:00
5	Polisportiva Roma (Tivoli)	Viale Tevere	96	4	Calcio 7	erba sintetica	OCCUPATO	14:30:00	16:30:00
6	Polisportiva Roma (Tivoli)	Viale Tevere	96	4	Calcio 7	erba sintetica	OCCUPATO	16:30:00	18:00:00
7	Polisportiva Roma (Tivoli)	Viale Tevere	96	4	Calcio 7	erba sintetica	OCCUPATO	18:30:00	20:30:00
8	Polisportiva Roma (Tivoli)	Viale Tevere	96	5	Calcio 8	erba	DISPONIBILE	[null]	[null]
9	Polisportiva Roma (Tivoli)	Viale Tevere	96	6	Basket	parquet	DISPONIBILE	[null]	[null]
10	Polisportiva Roma (Tivoli)	Viale Tevere	96	7	Calcio 8	erba	DISPONIBILE	[null]	[null]
11	Polisportiva Roma (Tivoli)	Viale Tevere	96	8	Basket	parquet	DISPONIBILE	[null]	[null]

## Query 6

La query mostra i tesserati della Polisportiva Romana che hanno fatto almeno 2 prenotazioni nel 2020 e indica il campo più prenotato e il relativo numero di prenotazioni fatte dal tesserato.

```
DROP VIEW IF EXISTS prenotazioni_tesserato_1campo;
CREATE VIEW prenotazioni_tesserato_1campo AS
SELECT codass, id_tesserato, max(num) as max
FROM (
    SELECT codass, id_tesserato, id_campo, count(*) as num
    FROM prenotazioni
    WHERE extract(YEAR from data) = extract(year from CURRENT_DATE)-1
    GROUP BY codass, id_tesserato, id_campo
) as conteggio
GROUP BY codass, id_tesserato;

SELECT
p.id_tesserato as "Codice Fiscale", T.cognome, T.nome, s.nome as "Nome della sede",
p.id_campo as "Numero Campo Preferito", tc.sport, pmax.max as "N° Prenotazioni"
FROM prenotazioni p
JOIN prenotazioni_tesserato_1campo pmax ON pmax.codass = p.codass AND pmax.id_tesserato = p.id_tesserato
JOIN tesserato T ON p.codass = T.codass AND p.id_tesserato = T.cf
JOIN campo c ON c.codass = p.codass AND p.id_campo = c.id
JOIN tipologia_campo tc ON tc.codass = p.codass AND tc.id = c.tipologia
JOIN sede s ON s.codass = p.codass AND s.codice = p.sede
WHERE p.codass = 'POLRM' AND extract(YEAR from P.data) = extract(year from CURRENT_DATE)-1
GROUP BY p.id_tesserato, p.id_campo, pmax.max, T.cognome, T.nome, p.sede, s.nome, tc.sport
HAVING count(*) = pmax.max AND count(*) > 2
ORDER BY pmax.max DESC, cognome, nome;
```

	Codice Fiscale character (16)	cognome character varying (80)	nome character varying (80)	Nome della sede character varying (150)	Numero Campo Preferito integer	sport character varying (50)	N° Prenotazioni bigint
1	FRRZRA72E42E530Z	Farronato	Zaira	Polisportiva Roma (Tivoli)		4 Calcio 7	11
2	NTNNGR56D55D66...	Antoniceilo	Annagrazia	Polisportiva Roma (Tivoli)		4 Calcio 7	10
3	BRNNMO12M71E8...	Barone	Noemi	Polisportiva Roma (Tivoli)		4 Calcio 7	10
4	PPPCLS73T25L810...	Peppe	Celso	Polisportiva Roma (Tivoli)		1 Calcio	10
5	PNTRND84D30H10...	Raimondo	Pantelli	Polisportiva Roma (Tivoli)		3 Calcio 5	10
6	TSSGSI73H28G1900	Tessaroli	Giosuè	Polisportiva Roma (Tivoli)		3 Calcio 5	10

## 6 - Indici

### 6.1 Indice sulle città

Un filtro molto importante per il portale online sarà quello relativo alla città in cui si trovano le varie sedi. La tabella contenente i comuni italiani non subirà né modifiche né inserimenti nel tempo (se non dovuti a casi eccezionali), si presta quindi ottimale per l'utilizzo di un indice. Quindi per ottimizzare la ricerca, l'implementazione di quest'ultimo rende la selezione sulla tabella molto più efficiente.

```
DROP INDEX IF EXISTS idx_istat_citta;  
CREATE INDEX idx_istat_citta ON citta (istat);
```

### 6.2 Indice sui dipendenti

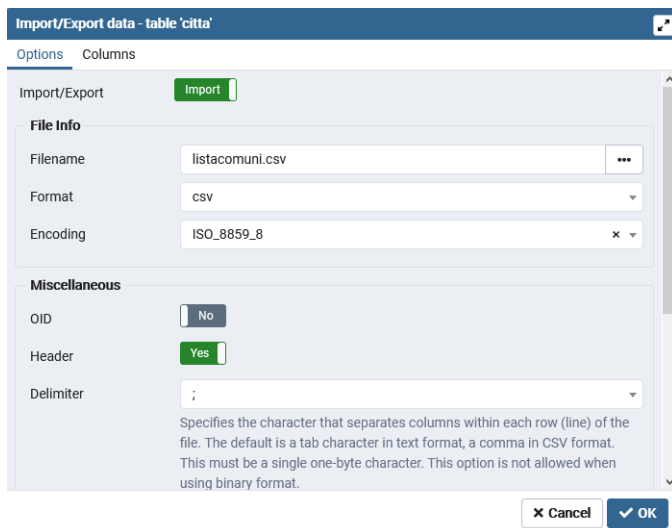
Un'altra entità in cui potrebbe essere utile l'uso di un indice è quella dei dipendenti. A differenza di quella relativa ai tesserati, nella quale si prevede un inserimento e un aggiornamento dei dati abbastanza frequente (ad esempio nel caso relativo all'apertura di una nuova sede), la lista dei dipendenti non subirà eccessive modifiche nel tempo.

In questo caso, visto che la base di dati verrà utilizzata per un sistema multi-tenant, dovremmo creare un indice composito.

```
DROP INDEX IF EXISTS idx_dipendenti;  
CREATE INDEX idx_dipendenti ON dipendente ( codass , cf );
```

## 7 – Istruzioni per il database e per il programma

### 7.1 Istruzioni per il database



Nella cartella di consegna è presente il file **"codice.sql"** che contiene il codice per la creazione, il popolamento e le 6 query richieste per la base di dati.

Prima di eseguire il codice relativo alle insert (righe 201-875), bisogna importare l'elenco dei comuni italiani contenuti nel file **"listacomuni.csv"** nella tabella **"citta"**. Le voci da impostare per l'importo dei dati sono rappresentate nello screenshot qui a fianco.

### 7.2 Istruzioni per il programma

Per utilizzare il codice C++, basta modificare i parametri di configurazione di connessione al database, presenti all'inizio del programma come costanti (**"PGuser"**, **"PGdb"**, **"PGpass"**). Per compilare usare il comando:

```
g++ main.cpp -o main -I /usr/include/postgresql -lpq
```

Il file **"query.h"** contiene le query di cui avrà bisogno il main (la stringa **#include** è già presente nel codice).