

UNIVERSITÀ DEGLI STUDI DI NAPOLI
PARTHENOPE



DIPARTIMENTO DI INGEGNERIA

**CORSO DI LAUREA IN INGEGNERIA E SCIENZE
INFORMATICHE PER LA CYBERSECURITY**

ANNO ACCADEMICO 2024/2025

ALUNNO

SIMONE D'ANNA, 0334000158

Tema Progetto



**ASSO
HUB**



Traccia d'Esame – Progetto di Sistema Informativo per la Gestione di un'Associazione (AssoHub)

Il progetto *AssoHUB* nasce dall'esigenza reale di digitalizzare e semplificare la gestione interna di un'associazione, sostituendo le procedure manuali e la frammentazione dei dati con un sistema informativo moderno, sicuro e accessibile via web.

L'obiettivo è fornire uno strumento completo che consenta al direttivo di coordinare in modo organizzato iscritti, eventi e movimenti economici, garantendo al contempo una fruizione chiara e intuitiva per i soci e una maggiore trasparenza nelle attività associative.

Struttura del sistema

Al centro dell'applicazione si trova un **database SQL relazionale**, progettato per ospitare tutte le informazioni fondamentali al funzionamento dell'associazione. La base dati è strutturata in modo da garantire integrità, coerenza e tracciabilità, consentendo query efficienti e una gestione scalabile dei dati.

Ogni utente registrato dispone di un **account personale con credenziali univoche**, ed è inquadrato in una delle due categorie principali:

- **Associati:** rappresentano la base sociale dell'associazione. Il sistema gestisce le loro informazioni anagrafiche, i contatti, lo stato della tessera associativa e i versamenti delle quote, includendo importo, data, anno di riferimento e stato del pagamento. Gli associati hanno accesso alla homepage pubblica con gli eventi futuri e possono iscriversi alle attività organizzate, con la possibilità di consultare lo storico delle proprie partecipazioni.
- **Amministratori:** costituiscono il direttivo e dispongono di funzionalità gestionali avanzate. Possono pianificare eventi, registrare e monitorare i movimenti economici, gestire l'elenco degli iscritti e supervisionare l'andamento generale dell'associazione. Ogni movimento economico (entrata o uscita) può essere collegato a un evento specifico o registrato come operazione indipendente, favorendo una contabilità interna chiara e completa.

Funzionalità principali

Il sistema garantisce una serie di servizi centrali:

- **Gestione centralizzata degli utenti** con autenticazione e differenziazione dei privilegi.
- **Amministrazione delle quote associative** con monitoraggio puntuale dei pagamenti, scadenze e rinnovi annuali.
- **Organizzazione e pianificazione degli eventi**, con gestione degli iscritti, tracciamento delle presenze e reportistica delle attività.
- **Contabilità integrata**, che consente la registrazione e la consultazione di entrate e uscite, distinguendo sponsorizzazioni, donazioni, affitti, spese logistiche e altre voci.
- **Dashboard riservata al direttivo**, che fornisce una visione di sintesi dell'andamento economico e organizzativo dell'associazione.

Valore aggiunto del progetto

Il progetto si distingue per la sua **forte adesione a un'esigenza reale**: supportare concretamente la vita quotidiana di un'associazione, migliorando **trasparenza, tracciabilità e organizzazione**.

Rispetto a una gestione tradizionale cartacea o basata su file non centralizzati, *AssoHUB*:

- riduce gli errori umani e la perdita di informazioni,
- aumenta la **trasparenza** verso i soci,
- facilita il lavoro del direttivo,
- offre una piattaforma digitale accessibile anche da remoto.

Inoltre, l'interfaccia web è progettata con un approccio **user-friendly**, per permettere anche a utenti non esperti di accedere facilmente alle informazioni e partecipare attivamente alla vita associativa.

Obiettivi del progetto

1. Analisi e progettazione concettuale

A partire dalla descrizione del contesto associativo, è stato analizzato il dominio della gestione di un'associazione e costruito un modello entità-relazione completo, includendo iscritti, quote associative, eventi, movimenti economici e ruoli utente. Sono state gestite le relazioni di appartenenza e partecipazione, motivando le scelte effettuate per garantire coerenza, integrità e tracciabilità dei dati.

Progettazione Concettuale

L'analisi ha portato all'individuazione delle entità e delle relazioni che costituiscono il cuore del sistema informativo AssoHUB.

Lo schema E/R adottato prevede una struttura centralizzata sull'entità Utente, che rappresenta ogni soggetto registrato all'interno dell'associazione, specializzato in due ruoli distinti e mutuamente esclusivi: Associato e Admin.

Tipo di generalizzazione

Il modello prevede una generalizzazione totale e disgiunta sull'entità Utente:

- **Totale**, poiché ogni utente del sistema deve appartenere obbligatoriamente a una delle due sottoclassi (Associato o Admin).
- **Disgiunta**, poiché un utente può ricoprire un solo ruolo alla volta: non è possibile essere contemporaneamente Associato e Admin.

Utente contiene gli attributi comuni a entrambe le categorie:

ID, nome, cognome, dataNascita, indirizzoResidenza, email, numeroCell, dataIscrizione, idTessera

Descrizione delle entità

1. Utente

Superclasse che raccoglie le informazioni anagrafiche di tutti i soggetti iscritti al sistema.

- Attributi: ID, nome, cognome, dataNascita, indirizzoResidenza, email, numeroCell, dataIscrizione, idTessera.

2. Associato

Sottoclasse di Utente che rappresenta i membri ordinari dell'associazione.

Gli associati possono versare quote, partecipare a eventi e visualizzare le proprie attività personali.

3. Admin

Sottoclasse di Utente che rappresenta i membri del direttivo.

Gli amministratori possono pianificare eventi, registrare movimenti economici e supervisionare l'andamento gestionale.

4. Account

Rappresenta le credenziali di accesso al sistema.

- Attributi: ID, username, password.
- Ogni account è associato **1:1** a un singolo Utente.

5. Quota

Memorizza i versamenti delle quote associative annuali.

- Attributi: ID, tipo, dataVersamento, importo, annoRiferimento, statoPagamento.
- Ogni quota è versata da uno o più associati.

6. Evento

Rappresenta un'attività o iniziativa organizzata dall'associazione.

- Attributi: ID, titolo, descrizione, dataInizio, dataFine, tipoEvento.
- Gli eventi possono essere pianificati dagli admin e aperti alla partecipazione degli associati.

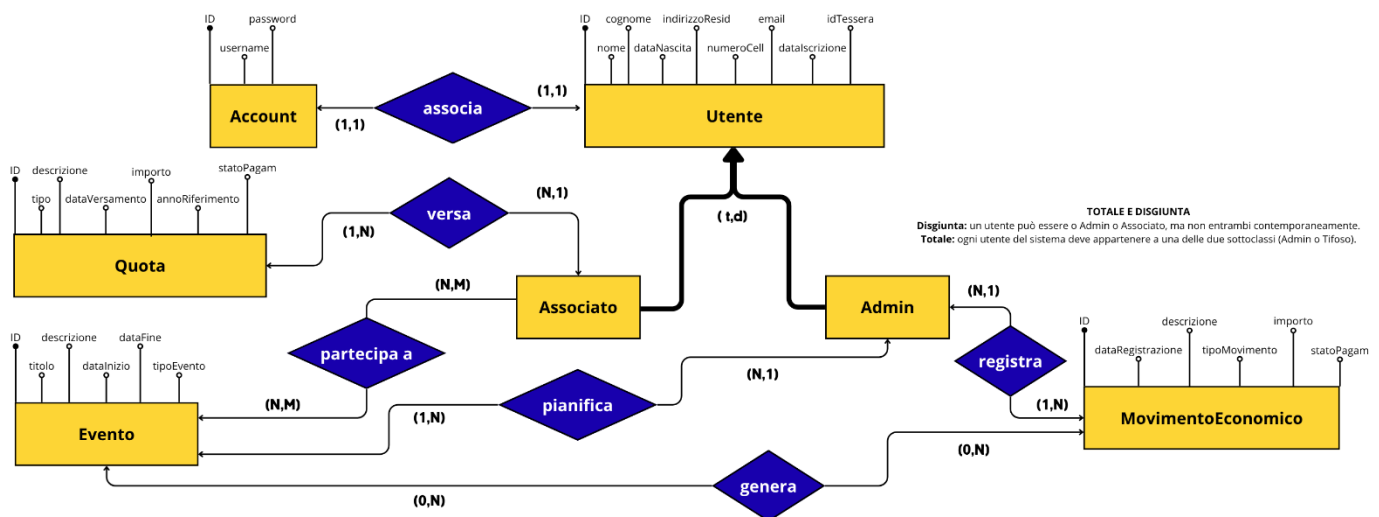
7. MovimentoEconomico

Registra le operazioni di entrata o uscita relative all'associazione.

- Attributi: ID, descrizione, dataRegistrazione, tipoMovimento, importo, statoPagamento.
- Ogni movimento può essere collegato a un evento generato o gestito dagli amministratori.

Progettazione concettuale

Dall'analisi possiamo rappresentare la problematica con un primo schema E/R:



Tipo di generalizzazione

Il progetto *AssoHUB* prevede l'utilizzo di una **generalizzazione totale e disgiunta** per l'entità **Utente**, con lo scopo di modellare in modo rigoroso le diverse figure coinvolte nella gestione delle attività associative.

La generalizzazione dell'entità *Utente* definisce due sottoclassi principali: **Associato** e **Admin**.

Questa struttura eredita gli attributi comuni (come *nome*, *cognome*, *data di nascita*, *indirizzo di residenza*, *email*, *numero di cellulare*, *data di iscrizione*, *id tessera*) dalla superclasse **Utente**, mentre ogni sottoclasse aggiunge comportamenti e ruoli specifici nel contesto del sistema informativo:

- **Associato:** rappresenta il socio ordinario dell'associazione. Può iscriversi agli eventi, partecipare alle attività e versare le proprie quote associative annuali.
- **Admin:** rappresenta un membro del direttivo. Ha accesso a funzionalità gestionali avanzate, come la pianificazione di eventi, la registrazione di movimenti economici e la supervisione della contabilità interna.

La generalizzazione è definita come:

- **Totale:** ogni istanza dell'entità *Utente* deve appartenere obbligatoriamente a una delle due sottoclassi (*Associato* o *Admin*). Non esiste un "utente generico" privo di ruolo.
- **Disgiunta:** un utente può ricoprire **un solo ruolo alla volta**. Non è possibile essere contemporaneamente *Associato* e *Admin*.

Questa scelta modellistica riflette le regole del dominio applicativo: ogni utente registrato nel sistema *AssoHUB* è chiaramente identificato in un ruolo specifico, che determina i permessi di accesso e le operazioni consentite all'interno della piattaforma.

Ciò garantisce coerenza tra i dati, sicurezza nei permessi e chiarezza semantica nelle interazioni previste dal sistema.

Eliminazione della gerarchia: strategie di ristrutturazione

La presenza di una generalizzazione nell'entità *Utente* non è direttamente implementabile nel modello relazionale; perciò, è necessario tradurre la gerarchia in tabelle SQL mediante una delle strategie di ristrutturazione comunemente adottate.

Di seguito vengono analizzate le due principali opzioni, con relativi vantaggi e svantaggi, al fine di motivare la scelta progettuale più adeguata al contesto applicativo di *AssoHUB*.

1. Strategia “Tutto nella superclasse” (un’unica tabella)

Questa soluzione prevede di mantenere un’unica entità *Utente*, che incorpora tutti gli attributi comuni e specifici delle sottoclassi (*Associato*, *Admin*), aggiungendo un attributo discriminante (*ruolo*) per identificare il tipo di utente.

```
Utente(  
    ID,  
    nome,  
    cognome,  
    dataNascita,  
    indirizzoResidenza,  
    email,  
    numeroCell,  
    dataIscrizione,  
    idTessera,  
    ruolo,                -- (Associato / Admin)  
    dataNomina,           -- NULL se non è Admin  
    delegaGestione        -- NULL se non è Admin  
)
```

Vantaggi:

- Struttura centralizzata e semplice da implementare.
- Maggiore rapidità nelle query comuni (autenticazione, elenco utenti).
- Riduzione del numero di tabelle e di join.

Svantaggi:

- Presenza di numerosi valori *NULL* negli attributi specifici delle sottoclassi.
- Difficoltà nel mantenere vincoli di integrità logica (es. obbligo di *delegaGestione* solo per gli Admin).
- Perdita di chiarezza semantica e leggibilità del modello.

Conclusione:

Questa soluzione è adatta a sistemi con differenze minime tra ruoli e poche funzioni specializzate, ma non è ottimale per *AssoHUB*, che prevede ruoli distinti e responsabilità separate.

2. Strategia “Tutto nelle sottoclassi” (tabelle separate)

Questa strategia consiste nell’eliminare la superclasse *Utente* e replicare i suoi attributi comuni in ciascuna delle sottoclassi specializzate.

Ogni ruolo diventa così un’entità autonoma, dotata di tutti gli attributi necessari per descriverlo in modo completo.

```
Associato(  
    ID,  
    nome,  
    cognome,  
    dataNascita,  
    indirizzoResidenza,  
    email,  
    numeroCell,  
    dataIscrizione,  
    idTessera  
)
```

```
Admin(  
    ID,  
    nome,  
    cognome,  
    dataNascita,  
    indirizzoResidenza,  
    email,  
    numeroCell,  
    dataIscrizione,  
    idTessera  
)
```

Vantaggi:

- Nessun valore *NULL*: ogni tabella contiene solo dati pertinenti al proprio ruolo.
- Migliore integrità semantica e logica.
- Possibilità di applicare vincoli specifici e personalizzati per ogni categoria di utente.
- Maggiore chiarezza del modello e semplicità nella manutenzione dei dati.
- Estensibilità futura: facile aggiunta di nuovi ruoli o attributi dedicati.

Svantaggi:

- Ridondanza degli attributi comuni (*nome, email, idTessera, ecc.*).
- Operazioni trasversali (es. login unificato) richiedono unione o controlli separati su più tabelle.
- Maggiore complessità nella gestione delle query globali sugli utenti.

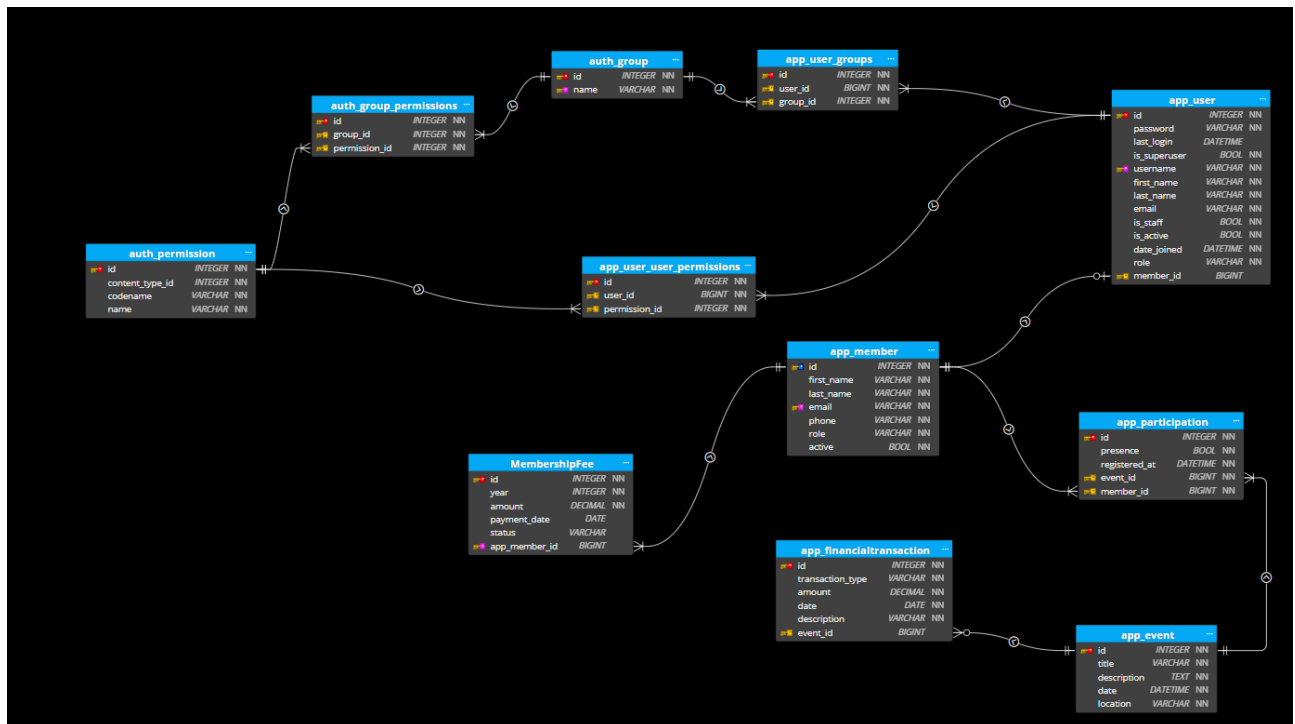
Scelta progettuale: adozione della strategia “Tutto nelle sottoclassi”

Per il progetto *AssoHUB* è stata scelta la **strategia “Tutto nelle sottoclassi”**, in quanto risulta più coerente con la logica del dominio e con la struttura implementata nel database effettivo.

Questa decisione è motivata da:

- Differenze funzionali nette tra *Associato* e *Admin*, ciascuno con compiti e privilegi distinti.
- Necessità di evitare la presenza di attributi *NULL* e garantire **integrità semantica** dei dati.
- Maggiore chiarezza e leggibilità dello schema, soprattutto in fase di analisi e manutenzione.
- Coerenza con la progettazione concettuale a **generalizzazione totale e disgiunta**, dove ogni utente appartiene a una sola categoria.

Pur accettando una certa ridondanza di attributi, questa strategia consente un modello **più pulito, modulare e scalabile**, garantendo una migliore aderenza alle regole del dominio associativo e alle effettive esigenze applicative di *AssoHUB*.



Descrizione delle relazioni tra le entità

Per quanto riguarda le **relazioni**, l'analisi del modello concettuale di *AssoHUB* ha permesso di individuare le seguenti associazioni principali tra le entità del sistema:

1. Relazione tra l'entità *Iscritto* e l'entità *QuotaAssociativa*

Tra le entità **Iscritto** e **QuotaAssociativa** esiste una **relazione uno a molti (1:N)**, in quanto **un iscritto può versare una o più quote associative** nel corso del tempo, mentre **ogni quota è riferita a un solo iscritto**.

Questa relazione consente di monitorare in modo puntuale lo storico dei pagamenti e lo stato della tessera associativa, garantendo la tracciabilità dei versamenti annuali.

2. Relazione tra l'entità Iscritto e l'entità Partecipazione

Tra **Iscritto** e **Partecipazione** esiste una **relazione uno a molti (1:N)**, in quanto **un iscritto può iscriversi a più eventi**, mentre **ogni partecipazione è effettuata da un solo iscritto**.

Tale relazione permette di tenere traccia della presenza dei soci alle attività organizzate e di ricostruire lo storico delle loro partecipazioni.

3. Relazione tra l'entità Evento e l'entità Partecipazione

Tra **Evento** e **Partecipazione** esiste una **relazione uno a molti (1:N)**, in quanto **un evento può accogliere più iscritti**, mentre **ogni partecipazione si riferisce a un solo evento**.

In questo modo è possibile gestire l'elenco dei partecipanti a ciascun evento e, viceversa, mostrare per ogni socio le attività a cui ha aderito.

4. Relazione tra l'entità Evento e l'entità MovimentoEconomico

Tra **Evento** e **MovimentoEconomico** esiste una **relazione uno a molti (1:N)**, poiché **un evento può generare più movimenti economici** (ad esempio entrate da quote o uscite per spese organizzative), mentre **ogni movimento economico è collegato a un solo evento**.

Questa relazione è opzionale, in quanto alcuni movimenti economici possono non essere associati ad alcun evento specifico (ad esempio donazioni o spese generali).

5. Relazione tra l'entità Admin e l'entità Evento

Tra **Admin** e **Evento** esiste una **relazione uno a molti (1:N)**, in quanto **un amministratore può creare o gestire più eventi**, mentre **ogni evento è pianificato da un solo amministratore**.

Questa relazione consente di garantire la tracciabilità delle operazioni di pianificazione e gestione delle attività associative.

6. Relazione tra l'entità Admin e l'entità MovimentoEconomico

Tra **Admin** e **MovimentoEconomico** esiste una **relazione uno a molti (1:N)**, poiché **ogni amministratore può registrare più movimenti economici**, mentre **ogni movimento è stato inserito da un solo admin**.

Ciò permette di mantenere un controllo contabile chiaro e di associare ogni operazione finanziaria all'autore che l'ha registrata.

7. Relazione tra l'entità Utente e l'entità Iscritto

Tra **Utente** e **Iscritto** esiste una **relazione uno a uno (1:1)**, in quanto **ogni utente dispone di un solo profilo anagrafico**, e **ogni iscritto è associato a un unico account utente**.

Questa relazione permette di separare la gestione dei dati anagrafici da quella delle credenziali di accesso, migliorando la sicurezza e l'organizzazione del sistema.

8. Relazione tra l'entità QuotaAssociativa e l'entità MovimentoEconomico (relazione logica)

Si può inoltre individuare una **relazione logica uno a molti (1:N)** tra **QuotaAssociativa** e **MovimentoEconomico**, poiché **ogni quota versata può generare un movimento economico di tipo “Entrata”**, mentre **ogni movimento può riferirsi a una singola quota o a un gruppo di versamenti**.

Questa relazione assicura coerenza tra la gestione dei pagamenti e la contabilità interna.

2. Progettazione logica

Il modello concettuale è stato tradotto in un modello logico relazionale, con la definizione delle tabelle, degli attributi e delle chiavi primarie/esterne.

Particolare attenzione è stata posta nella normalizzazione, così da evitare ridondanze e assicurare una corretta gestione delle relazioni tra soci, pagamenti, eventi e movimenti contabili.

Iscritto(id_iscritto **PK**, nome, cognome, email **UNIQUE**, telefono, ruolo)

QuotaAssociativa(id_quota **PK**, anno, importo, data_pagamento, stato, id_iscritto **FK** → **Iscritto**)

Vincolo unico: (anno, id_iscritto)

Evento(id_evento **PK**, titolo, descrizione, data, luogo)

Partecipazione(id_iscritto **FK** → **Iscritto**, id_evento **FK** → **Evento**, presenza)

Vincolo unico: (id_iscritto, id_evento)

MovimentoEconomico(id_movimento **PK**, tipo, importo, data, descrizione, id_evento **FK** → **Evento** **NULL**)

Utente(id_utente **PK**, username **UNIQUE**, password_hash, ruolo, id_iscritto **FK** → **Iscritto**)

Più nel dettaglio avremo:

Iscritto (Member)

Nome Campo	Descrizione	Tipo Dati	Lunghezza	Vincoli
id	Identificativo univoco dell'iscritto	Int	11	PK, Auto increment, Not null
first_name	Nome dell'iscritto	Varchar	150	Not null
last_name	Cognome dell'iscritto	Varchar	150	Not null
email	Indirizzo di posta elettronica univoco dell'iscritto	Email	254	Unique, Not null
phone	Numero di telefono del socio	Varchar	30	Null ammesso
role	Ruolo del socio (Associato / Amministratore)	Varchar	20	Valori ammessi: "associato", "amministratore", Not null
active	Stato di attivazione del socio	Boolean	1	Default: True

Utente (User)

Nome Campo	Descrizione	Tipo Dati	Lunghezza	Vincoli
id	Identificativo univoco dell'utente	Int	11	PK, Auto increment, Not null
username	Nome utente per l'autenticazione	Varchar	150	Unique, Not null
password	Hash crittografato della password	Varchar	128	Not null
first_name	Nome dell'utente	Varchar	150	Not null
last_name	Cognome dell'utente	Varchar	150	Not null
email	Email associata all'account	Email	254	Unique, Not null
role	Ruolo dell'utente (Associato / Amministratore)	Varchar	20	Valori ammessi: "associato", "amministratore", Default: associato
member_id	Riferimento all'iscritto associato	Int	11	FK → Member(id), Null ammesso
is_staff	Flag di accesso all'area admin Django	Boolean	1	Default: False
is_superuser	Privilegio amministrativo massimo	Boolean	1	Default: False
is_active	Stato dell'account	Boolean	1	Default: True

QuotaAssociativa (MembershipFee)

Nome Campo	Descrizione	Tipo Dati	Lunghezza	Vincoli
id	Identificativo univoco della quota	Int	11	PK, Auto increment, Not null
member_id	Iscritto che ha versato la quota	Int	11	FK → Member(id), Not null
year	Anno di riferimento della quota	Int	4	Default: anno corrente, Not null
amount	Importo della quota associativa	Decimal	(8,2)	Check(amount > 0)* , Not null
payment_date	Data del versamento	Date	-	Null ammesso
status	Stato del pagamento	Varchar	15	Valori ammessi: “ <i>pendente</i> ”, “ <i>pagato</i> ”, Default: <i>pendente</i>
Vincolo Unico	(member_id, year)			Garantisce una sola quota per socio e anno

Evento (Event)

Nome Campo	Descrizione	Tipo Dati	Lunghezza	Vincoli
id	Identificativo univoco dell'evento	Int	11	PK, Auto increment, Not null
title	Titolo dell'evento	Varchar	200	Not null
description	Descrizione testuale dell'evento	Text	-	Null ammesso
date	Data e ora di svolgimento	DateTime	-	Not null
location	Luogo dell'evento	Varchar	200	Not null

Partecipazione (Participation)

Nome Campo	Descrizione	Tipo Dati	Lunghezza	Vincoli
id	Identificativo univoco della partecipazione	Int	11	PK, Auto increment, Not null
member_id	Iscritto che partecipa all'evento	Int	11	FK → Member(id), Not null
event_id	Evento a cui l'iscritto partecipa	Int	11	FK → Event(id), Not null
presence	Stato di presenza del socio (True = presente)	Boolean	1	Default: False
registered_at	Data e ora di iscrizione all'evento	DateTime	-	Default: now()
Vincolo Unico	(member_id, event_id)			Evita iscrizioni duplicate per lo stesso evento

MovimentoEconomico (FinancialTransaction)

Nome Campo	Descrizione	Tipo Dati	Lunghezza	Vincoli
id	Identificativo univoco del movimento economico	Int	11	PK, Auto increment, Not null
transaction_type	Tipo di movimento (Entrata / Uscita)	Varchar	10	Valori ammessi: "entrata", "uscita", Not null
amount	Importo dell'operazione economica	Decimal	(10,2)	Check(amount > 0), Not null
date	Data di registrazione del movimento	Date	-	Default: now(), Not null
description	Descrizione sintetica o causale	Varchar	255	Not null
event_id	Evento collegato al movimento (se presente)	Int	11	FK → Event(id), Null ammesso

Descrizione dei Vincoli

1. Integrità Referenziale

Tipo: Vincolo di Interrelazione

Descrizione:

Le relazioni tra le entità principali del sistema vengono garantite tramite **chiavi esterne (FK)**, che assicurano la coerenza e l'integrità dei collegamenti tra le tabelle.

Ogni chiave esterna obbliga il riferimento a un record esistente, evitando dati orfani o inconsistenze logiche.

Vincoli attivi:

- MembershipFee.member → Member(id)
- Participation.member → Member(id)
- Participation.event → Event(id)
- FinancialTransaction.event → Event(id)
- User.member → Member(id)

Questi vincoli impediscono riferimenti a entità inesistenti e garantiscono la corretta propagazione delle cancellazioni (grazie alle clausole on_delete=models.CASCADE o SET_NULL presenti nei modelli).

2. Integrità delle Chiavi Primarie

Tipo: Vincolo di Chiave

Descrizione:

Ogni tabella del sistema *AssoHUB* possiede una **chiave primaria (PK)** che identifica univocamente ogni record e impedisce la duplicazione dei dati. Le chiavi primarie non possono contenere valori nulli.

Chiavi definite:

- Member(id)
- User(id)
- MembershipFee(id)
- Event(id)
- Participation(id)
- FinancialTransaction(id)

3. Attributi NOT NULL

Tipo: Vincolo di Tupla

Descrizione:

Campi essenziali per il funzionamento del sistema devono essere sempre valorizzati, al fine di garantire la **completezza informativa** e l'affidabilità del database.

Sono pertanto definiti come *NOT NULL* i seguenti attributi:

- Member.first_name, Member.last_name, Member.email, Member.role
- User.username, User.password, User.email, User.role
- MembershipFee.amount, MembershipFee.year, MembershipFee.status
- Event.title, Event.date, Event.location
- FinancialTransaction.transaction_type, FinancialTransaction.amount, FinancialTransaction.date, FinancialTransaction.description

Questa configurazione assicura che nessuna entità venga creata senza i dati fondamentali richiesti per la sua identificazione e operatività.

4. Unicità delle Email

Tipo: Vincolo di Chiave (Unique)

Descrizione:

Sia per la tabella **Member** che per la tabella **User**, l'attributo email è definito come **univoco (UNIQUE)**.

Questo vincolo impedisce la registrazione di più utenti o iscritti con lo stesso indirizzo e-mail, evitando conflitti in fase di autenticazione e garantendo la corrispondenza 1:1 tra indirizzo e identità digitale.

5. Vincolo sulle Quote Associative

Tipo: Vincolo di Interrelazione + Unicità Composta

Descrizione:

Il modello *MembershipFee* impone un vincolo di **unicità composta** (member, year), che garantisce che **ogni socio possa avere al massimo una sola quota per ciascun anno associativo**.

Inoltre, la chiave esterna member assicura che la quota sia sempre associata a un iscritto esistente.

6. Vincolo sul Tipo di Movimento Economico

Tipo: Vincolo di Dominio

Descrizione:

L'attributo `transaction_type` nella tabella *FinancialTransaction* può assumere esclusivamente i valori:

- “**entrata**”
- “**uscita**”

Questo vincolo di dominio assicura la coerenza semantica dei dati contabili e impedisce l'inserimento di movimenti non validi.

7. Vincolo sulle Partecipazioni agli Eventi

Tipo: Vincolo di Interrelazione + Unicità Composta

Descrizione:

Nella tabella *Participation* è presente un vincolo di **unicità composta** (member, event) che assicura che **un socio possa iscriversi una sola volta allo stesso evento**.

Inoltre, le chiavi esterne garantiscono che ogni partecipazione si riferisca a un iscritto e a un evento validi.

Formulazione:

UNIQUE (member_id, event_id)

8. Stato del Socio e Ruoli Utente

Tipo: Vincolo di Dominio

Descrizione:

Gli attributi `role` presenti in *Member* e *User* possono assumere esclusivamente i valori:

- “associato”
- “amministratore”

Questo vincolo assicura una chiara distinzione dei ruoli e impedisce l'inserimento di valori non previsti, garantendo coerenza nei privilegi di accesso e nelle funzionalità.

9. Integrità Logica tra Utente e Iscritto

Tipo: Vincolo di Interrelazione (1:1)

Descrizione:

Tra *User* e *Member* esiste un vincolo di corrispondenza **uno a uno (OneToOneField)**.

Ogni utente del sistema deve essere collegato a un solo iscritto, e viceversa, assicurando la consistenza tra i dati anagrafici (*Member*) e quelli di accesso (*User*).

10. Integrità dei Movimenti Collegati agli Eventi

Tipo: Vincolo di Interrelazione (Opzionale)

Descrizione:

Ogni *FinancialTransaction* può essere collegata a un evento tramite l'attributo `event_id`.

Il vincolo `on_delete=models.SET_NULL` consente di mantenere i dati contabili anche in caso di eliminazione dell'evento associato, preservando la tracciabilità economica.

11. Ordinamento e Completezza Informativa

Tipo: Vincolo di Tupla (Ordering e Meta Constraints)

Descrizione:

Ciascun modello definisce un criterio di ordinamento predefinito (`Meta.ordering`), utile a garantire una visualizzazione coerente e intuitiva dei dati nelle interfacce applicative:

- Member → per cognome e nome
- Event → per data crescente
- MembershipFee → per anno decrescente
- FinancialTransaction → per data decrescente

Questi vincoli non strutturali contribuiscono alla **leggibilità e coerenza funzionale del sistema informativo**.

3. Implementazione applicativa

Realizzare un'applicazione web **Django** che gestisca i dati modellati (iscritti, eventi, quote, movimenti economici) e offra **almeno quattro funzionalità** chiave per il direttivo e per gli associati, rispettando i vincoli:

- **Backend:** Django
- **View/UI:** Django Template System
- **Stile:** Bootstrap CSS
- **JavaScript:** solo dove strettamente indispensabile (es. conferme modal, datepicker, validazioni minime)

Funzionalità implementate (selezione minima garantita: 4)

1. Registrazione e autenticazione utenti

- Registrazione associati (signup) attraverso il portale admin, login/logout, recupero password.

Nome	Email	Telefono	Ruolo	Stato
Giampaolo De Stefano	g.destee@yourmail.com	3384195620	Associato	Attivo
Antonio Ongardi	a.ongardi@yourmail.com	3856558928	Associato	Attivo

- Ruoli: **Associato** (base) e **Amministratore** (direttivo).

Stato quote associative
Pagato
Pendente

Ultimi eventi
Irpina Mood Food Festival
DJ Set - 35 Millimetri
Sagra della Castagna e della Nocciola
Festa della Fejola

- Pagina profilo con dati anagrafici e stato tessera.

Nome	Email	Telefono	Ruolo	Stato
Giampaolo De Stefano	g.destee@yourmail.com	3384195620	Associato	Attivo
Antonio Ongardi	a.ongardi@yourmail.com	3856558928	Associato	Inattivo

- **Access control** con login_required e PermissionRequiredMixin.

AssoHUB

Tema scuroLogin

Accedi

Nome utente

Nome utente

Password

Password

Entra

Gestione eventi e partecipazioni

- CRUD eventi (titolo, descrizione, luogo, data/ora).

AssoHUB

EventiQuoteProfiloDashboardIscrittiMovimenti

Tema scuroCiao Simone D'AnnaLogout

Eventi

Nuovo evento

Prossimi eventi

Sagra della Castagna e della Nocciola

19/10/2025 15:00 - Avella (AV)

Festa "long-run" con prodotti dolci e salati, promozione dei frutti locali

Solo per iscrittiModifica

DJ Set - 35 Millimetri

30/10/2025 23:38 - MoviePlex Avellino

"THE NIGHT IS CALLING AGAIN" è segnata come evento in programma.

Solo per iscrittiModifica

Irpinia Mood Food Festival

28/08/2026 21:00 - Avellino

Festival enogastronomico su scala regionale con stand, show cooking, musica

Solo per iscrittiModifica

Eventi passati

Festa della Fejola - 10/10/2025

Modifica

- Iscrizione/cancellazione associato; tracciamento presenze.

AssoHUB

EventiQuoteProfilo

Tema scuroCiao g.deste g.desteLogout

Iscrizione all'evento registrata.

Eventi

Prossimi eventi

Sagra della Castagna e della Nocciola

19/10/2025 15:00 - Avella (AV)

Festa "long-run" con prodotti dolci e salati, promozione dei frutti locali

Iscritto

DJ Set - 35 Millimetri

30/10/2025 23:38 - MoviePlex Avellino

"THE NIGHT IS CALLING AGAIN" è segnata come evento in programma.

Iscriviti

- **Elenco pubblico degli eventi futuri (homepage) + dettagli evento.**

AssoHUB

Eventi

Quote

Profilo

Tema scuro

Ciao g.deste g.deste

Logout

Eventi

Prossimi eventi

Sagra della Castagna e della Nocciola
19/10/2025 15:00 - Avella (AV)
Festa "long-run" con prodotti dolci e salati, promozione dei frutti locali
[Iscriviti](#)

DJ Set - 35 Millimetri
30/10/2025 23:38 - MoviePlex Avellino
"THE NIGHT IS CALLING AGAIN" è segnata come evento in programma.
[Iscriviti](#)

Irpinia Mood Food Festival
28/08/2026 21:00 - Avellino
Festival enogastronomico su scala regionale con stand, show cooking, musica
[Iscriviti](#)

Eventi passati

Festa della Fejola - 10/10/2025

- (Opz.) esportazione CSV degli iscritti all'evento.

2. Gestione quote associative (pagamenti)

- Registrazione versamenti (importo, data, anno, stato).

AssoHUB Eventi Quote Profilo Dashboard Iscritti Movimenti					Tema scuro	Ciao Simone D'Anna	Logout
Quote associative					Registra quota		
Socio	Anno	Importo	Stato	Data pagamento			
Giampaolo De Stefano	2025	€ 50,00	Pendente				
Antonio Ongardi	2025	€ 50,00	Pagato	10/09/2025			
Giampaolo De Stefano	2024	€ 50,00	Pagato	25/09/2024			

- Storico quote per associato; filtro per anno/stato (pagato, pending, scaduto).

AssoHUB

[Eventi](#)

[Quote](#)

[Profilo](#)

[Dashboard](#)

[Iscritti](#)

[Movimenti](#)

Tema scuro

Ciao Simone D'Anna

Logout

Quote di Giampaolo De Stefano

Anno	Importo	Stato	Data pagamento
2025	€ 50,00	Pendente	
2024	€ 50,00	Pagato	25/09/2024

- Riepilogo di cassa delle quote incassate per periodo.

3. Movimenti economici interni (entrate/uscite)

- CRUD movimenti (categoria, importo, data, descrizione), con **opzionale** collegamento a un evento.
- **Dashboard direttivo**: totale entrate/uscite, saldo, classifica per categoria (tabelle/indicatori).

AssoHUB

Eventi

Quote

Profilo

Dashboard

Iscritti

Movimenti

Tema scuro

Ciao Simone D'Anna

Logout

Movimenti economici

Nuovo movimento

Entrate: € 750,00

Uscite: € 117,00

Saldo: € 633,00

Data	Tipo	Importo	Descrizione	Evento
25/10/2025	Uscita	€ 85,00	Bolletta LUCE Enel N°568970	None
19/10/2025	Entrata	€ 250,00	Pagamento Sagra	Sagra della Castagna e della Nocciola
15/10/2025	Uscita	€ 32,00	Pagamento Internet ISP	None
11/09/2025	Entrata	€ 500,00	Donazione APSUC	None

- Filtri per intervallo date e categorie (donazioni, sponsorizzazioni, affitti, spese) (opz).

4. Area Associato (personale) (opzionale se già coperte le 4 precedenti)

- Panoramica: prossimi eventi, mie iscrizioni, stato quote, dati profilo.
- (Opz.) preferenze notifiche / newsletter.

Le 4 funzionalità minime consigliate: **(1) Autenticazione, (2) Eventi, (3) Quote, (4) Movimenti economici**. L'**Area Associato** può essere il quinto modulo opzionale.