

Progetto Basi Di Dati 2015/16

**Di
Stefano Agresti, Matteo Chesi,
Luca Petrotta**



UNIVERSITÀ DI PISA

1. Analisi delle specifiche

1.1 Introduzione

Nell'affrontare questo progetto, come prima cosa, ci siamo occupati di definire chiaramente gli obiettivi imposti dalla documentazione, riordinando e catalogando le informazioni in più gruppi, così da poter affrontare più facilmente la grande mole di richieste presente nel progetto.

La prima stesura di un glossario dei termini è stata più che altro indicativa, ma ha comunque facilitato notevolmente la creazione dello schema ER, di cui parleremo nel dettaglio nel prossimo capitolo.

Di seguito le sezioni in cui tale glossario è diviso (le stesse in cui è stato poi diviso lo schema ER in fase di costruzione):

1. Caratteristiche e necessità delle piante in generale (paragrafi 2.2.1 - 2.2.5)
2. Interventi in generale e patologie (paragrafi 2.2.6 - 2.2.7)
3. Coltivazione in serra (paragrafo 2.2.8)
4. Area social (paragrafo 2.3.1)
5. Area store (paragrafo 2.3.2)
6. Area manutenzione (paragrafo 2.4)
7. Area garden design (paragrafo 2.5)

L'analisi del paragrafo 2.6 si trova invece alla fine della relazione, insieme al codice effettivo di SQL.

Come si può notare, la divisione in sezioni è stata effettuata col preciso intento di preparare il terreno per una costruzione dello schema ER con strategia "Bottom - Up", anche se di questo si parlerà in maniera più approfondita nel prossimo paragrafo. In generale, abbiamo cercato di mantenere il più possibile vicini concetti legati o simili fra di loro.

1.2 Glossario

1.2.1 Caratteristiche e necessità delle piante in generale

Nome	Breve descrizione
Pianta	<ul style="list-style-type: none">- Caratterizzata da: nome, genere, cultivar, dimensione massima, indice di accrescimento- Caratterizzata da uno o più periodi di fioritura e/o fruttificazione- Caratterizzata da un periodo vegetativo e uno di riposo (alternati durante l'anno)- Caratterizzata da un insieme di esigenze (luce, temperatura, irrigazione e concimazione)- Può essere o non essere sempreverde, dioica o non

	<p>dioica, infestante o non infestante</p> <ul style="list-style-type: none"> - Caratterizzata da un indice di manutenzione - Prese due piante, esiste una distanza minima a cui possono essere messe senza entrare in conflitto
Illuminazione	<ul style="list-style-type: none"> - Caratterizzata da una quantità: “pieno sole”, “mezz’ombra”, “ombra” - Può essere diretta o indiretta - Ogni pianta è caratterizzata da un numero minimo di ore di luce al giorno (diverso fra il periodo vegetativo e di riposo)
Temperatura	<ul style="list-style-type: none"> - Ogni pianta è caratterizzata da un range di temperature all’interno del quale può vivere
Terreno	<ul style="list-style-type: none"> - Caratterizzato da un valore di pH - Presenta più micro/macro elementi in determinate concentrazioni - Caratterizzato da una consistenza e permeabilità, derivate dai materiali di cui è composto
Irrigazione	<ul style="list-style-type: none"> - Ogni pianta è caratterizzata da una periodicità (definita in “volte a settimana”) di irrigazione (diversa fra periodo vegetativo e di riposo) e da una quantità d’acqua necessaria per ogni annaffiatura
Concimazione	<ul style="list-style-type: none"> - Caratterizzata da uno o più elementi (da disciogliere o nebulizzare) - Caratterizzata da un periodo dell’anno in cui è opportuno effettuarla - Caratterizzata da una periodicità; da notare che la concentrazione degli elementi può cambiare fra un intervento e l’altro
Genere ¹	<ul style="list-style-type: none"> - Genere a cui appartiene una pianta - Definisce l’indice di accrescimento (diverso fra parte aerea e radicale)

¹ In realtà, in seguito si è utilizzato il termine “Famiglia”.

1.2.2 Interventi in generale e patologie

Nome	Descrizione
Intervento di manutenzione	<ul style="list-style-type: none">- Intervento necessario per la corretta crescita della pianta- Può essere: rinvaso, piantumazione, potatura, concimazione, trattamento
Potatura	<ul style="list-style-type: none">- Intervento di manutenzione- Effettuata una o più volte all'anno- Può essere di vario tipo- Non tutti i tipi di potatura possono essere effettuati su tutte le piante- Uno stesso tipo, su una stessa pianta, può non essere effettuabile tutto l'anno
Rinvaso	<ul style="list-style-type: none">- Intervento di manutenzione- Per alcune piante, esistono periodi dell'anno in cui non può essere effettuato
Patologia	<ul style="list-style-type: none">- Malattie che possono colpire le piante- Relativamente ad una certa pianta, le varie patologie presentano probabilità ed entità di attacco diverso a seconda del periodo dell'anno- Causata da un agente patogeno- Può essere combattuta mediante lotta chimica o biologica- Caratterizzata da uno o più sintomi (anche condivisi fra varie patologie)
Sintomo	<ul style="list-style-type: none">- Cambiamenti in una pianta che permettono di individuare le patologie- Classificati da un codice- Accompagnati da una descrizione e una o più immagini
Agente Patogeno	<ul style="list-style-type: none">- Agente che causa una patologia
Prodotto	<ul style="list-style-type: none">- Strumento utilizzato per la lotta chimica alle patologie- Basato su uno o più principi attivi (in diverse concentrazioni)- Può essere somministrato in modi diversi- Può essere selettivo o ad ampio spettro- Per ogni agente, esiste un dosaggio consigliato- Per alcuni prodotti, può esistere un periodo dell'anno in cui non è possibile utilizzarli- Se utilizzato su una pianta da frutto, esiste un tempo di attesa

1.2.3 Coltivazioni in serra

Nome	Descrizione
Sede	<ul style="list-style-type: none"> - Contraddistinta da: codice, nome, indirizzo, numero di dipendenti - Composta da serre
Serra	<ul style="list-style-type: none"> - Luogo dove sono coltivate le piante - Contraddistinta da: codice, nome, indirizzo (anche diverso da quello della sede), dimensioni, numero massimo di piante, numero effettivo di piante - Divisa in sezioni
Sezione	<ul style="list-style-type: none"> - Contraddistinta da: codice, nome, numero massimo di piante, numero effettivo di piante - Caratterizzata da un certo livello di illuminazione, umidità e temperatura - Divisa in ripiani
Ripiano	<ul style="list-style-type: none"> - Diviso in contenitori - Contraddistinto da un codice
Pianta Ospitata	<ul style="list-style-type: none"> - Collocata in un contenitore (di dimensione adatta) - Contraddistinta da un codice - Possiede un'anagrafica in cui si possono trovare tutte le informazioni relative - Caratterizzata da una dimensione e un prezzo
Contenitore	<ul style="list-style-type: none"> - Contiene una pianta - Caratterizzato da un codice e una superficie - Caratterizzato da un livello di idratazione e una certa composizione del substrato - Il livello di idratazione deriva dal livello di irrigazione - Un contenitore resta sempre legato alla stessa pianta
Misurazione	<ul style="list-style-type: none"> - Giornaliera - Effettuata per ogni contenitore - Misura esalazioni di gas e concentrazioni degli elementi nel terreno
Report diagnostica di	<ul style="list-style-type: none"> - Giornaliero - Contiene gli esemplari ammalatisi nel corso della giornata - Contiene le patologie possibili, ordinate per attinenza - Contiene le varie misurazioni
Diagnosi	<ul style="list-style-type: none"> - Effettuata da un esperto che decide anche il trattamento da effettuare

Quarantena	- Contiene le piante malate
Trattamento	<ul style="list-style-type: none"> - Effettuato su piante malate - Caratterizzato da un codice trattamento, un codice piante, un prodotto somministrato (con la dose), data

1.2.4 Area social

Nome	Descrizione
Account	<ul style="list-style-type: none"> - Contiene le informazioni di anagrafica di ogni cliente - Contiene: nome, cognome, nickname, password, email, domanda/risposta sicurezza, città - Ha una certa credibilità
Profilo	<ul style="list-style-type: none"> - Collegato ad un account - Contiene le piante di cui è più appassionato - Collegato a delle schede
Scheda	<ul style="list-style-type: none"> - Relativa ad un profilo e ad un esemplare (acquistato dall'utente) - Contraddistinta da un codice - Contiene: nome della pianta, dimensione al momento dell'acquisto, data di acquisto, collocazione dell'esemplare - Se collocata in vaso sono presenti le dimensioni di quest'ultimo
Forum	<ul style="list-style-type: none"> - Luogo dove gli utenti discutono - Gli utenti ci pubblicano post
Post	<ul style="list-style-type: none"> - Contraddistinto da: nickname dell'utente che lo ha pubblicato, un testo, un timestamp, eventuali URL presenti - Se di risposta, ricevono un giudizio da 1 a 5 dagli altri utenti

1.2.5 Area store

Nome	Descrizione
Preferenze	<ul style="list-style-type: none"> - Caratteristiche per ogni utente - Contiene le piante preferite - Contiene le preferenze riguardanti esigenze, indice manutenzione, periodi vari, costo, dimensioni - Alcuni criteri possono essere più importanti di altri - Alcuni criteri possono non essere utilizzati
Ordine	- Caratterizzato da un codice ordine, un codice cliente, un codice pianta, istante di effettuazione

	<ul style="list-style-type: none"> - Ha uno stato (valori: in processazione, in preparazione, spedito, evaso e pendente) - Quando un ordine è evaso, si crea la relativa scheda - Se una pianta non è presente al momento dell'acquisto, l'ordine diventa pendente. Quando un esemplare di quella pianta arriva in azienda, l'ordine cambia stato
--	--

1.2.6 Area manutenzione

Nome	Descrizione
Manutenzione	<ul style="list-style-type: none"> - Può essere: su richiesta, programmato, automatico - È di un certo tipo - Più interventi possono essere aggregati per massimizzare le ore lavorative - Ogni intervento è relativo ad una pianta acquistata da un cliente - Ogni intervento ha una certa entità - Un intervento può essere effettuato dall'utente stesso - Ogni intervento ha un costo
Manutenzione su richiesta	<ul style="list-style-type: none"> - Segnalata da un account relativamente ad un esemplare acquistato - È associata ad una scadenza
Manutenzione programmata	<ul style="list-style-type: none"> - Per ogni pianta acquistata, l'utente può scegliere questa opzione - Basata sull'indice di manutenzione - Se gli addetti non sono sufficienti per gestire questi interventi, deve essere segnalato da un report
Manutenzione automatica	<ul style="list-style-type: none"> - Per ogni pianta acquistata, l'utente può scegliere questa opzione - In base all'indice di manutenzione, gli utenti ricevono notifiche riguardo gli interventi da effettuare - Se gli interventi non sono effettuati, l'entità cresce col passare del tempo

1.2.7 Area garden design

Nome	Descrizione
Giardino	<ul style="list-style-type: none">- Luogo in cui l'utente mette le piante- Diviso in settori- Può essere richiesto un preventivo
Settore	<ul style="list-style-type: none">- Può essere di piena terra o pavimentato- Può contenere vasi- Ha la forma di un poligono- Ha un certo orientamento (punto cardinale) e una certa esposizione alla luce; l'esposizione alla luce può cambiare con l'inserimento di nuove piante- Ogni settore ha un codice- Quando si costruisce un settore, viene suggerita una lista di piante adatte
Vaso	<ul style="list-style-type: none">- Di un certo materiale e dimensione

2. Progettazione Concettuale

2.1 Introduzione

Come già accennato nel capitolo precedente, per costruire lo schema ER abbiamo preferito utilizzare la strategia di progettazione “Bottom - up”, sia per valorizzare il lavoro di gruppo (usando questa strategia è stato possibile lavorare parallelamente sulle diverse sezioni) sia perché, secondo noi, questo era il metodo migliore per poter affrontare un progetto di questa mole, in cui riuscire a mantenere una visione di insieme può diventare piuttosto difficile.

2.2 Costruzione dello schema ER

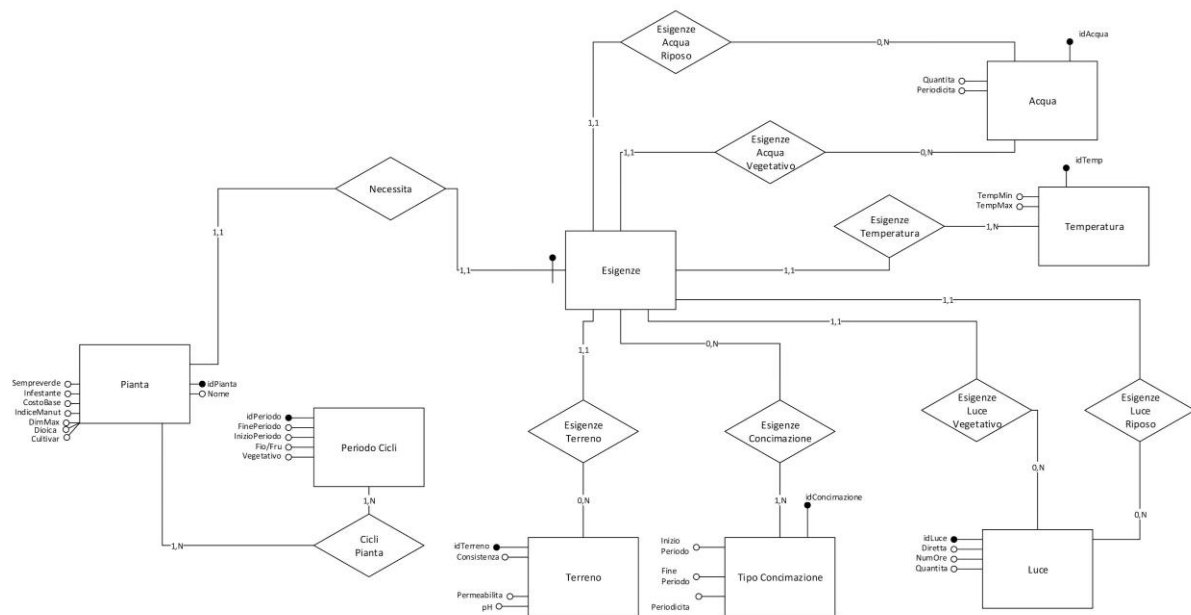
Nel seguire la “Bottom - up”, siamo partiti dalla costruzione delle varie porzioni di schema seguendo all'incirca la stessa separazione utilizzata nel glossario dei termini.

Di seguito, le varie parti dello schema accompagnate dai vincoli di integrità e dalle business rules individuati (lo schema finale completo è allegato alla documentazione). I dettagli sui valori possibili per i vari attributi vengono invece spiegati nel capitolo dedicato al modello relazionale e alla progettazione logica.

Come si può vedere dallo schema, si è deciso di introdurre vari id (valori auto incrementali) nelle varie entità, così da rendere più semplici le operazioni di join e ricerca.

2.2.1 Presentazione delle singole parti

2.2.1.1 Porzione sulle necessità e caratteristiche delle piante

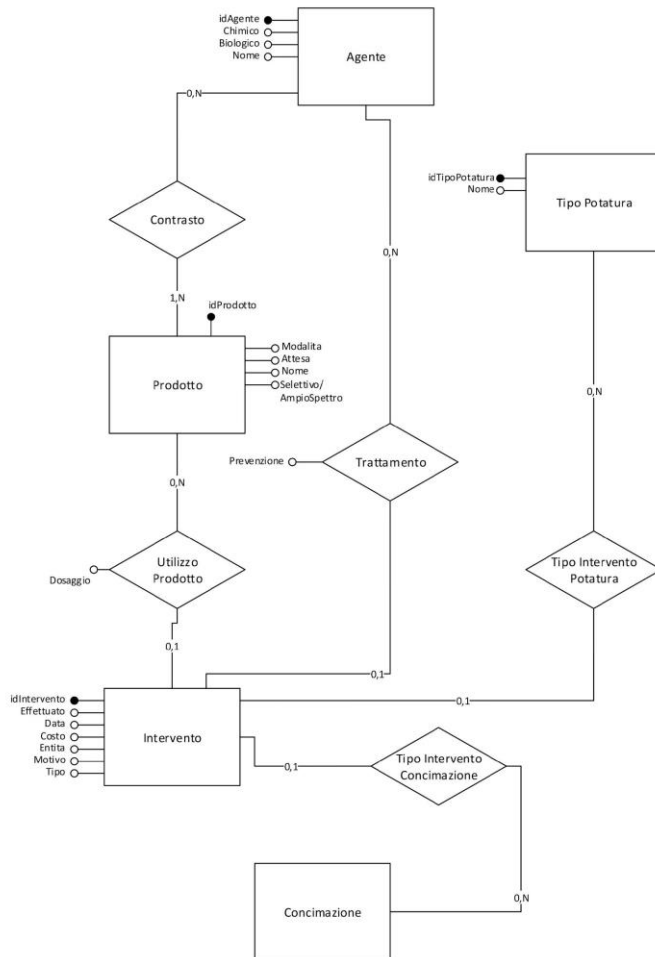


Di seguito, i vincoli non esprimibili tramite lo schema ER.

2.2.1.1.1 Vincoli generici e business rules

- 1) Un periodo di fioritura e/o fruttificazione deve anche essere vegetativo
- 2) Gli elementi che non partecipa alla relazione "UtilizzoElemento", non può partecipare alla relazione "Quantita"
- 3) I periodi vegetativi e di riposo non possono sovrapporsi e devono coprire l'intero anno
- 4) La spiegazione dell'attributo "IndiceManut" e della crescita di una pianta in base al suo indice di accrescimento sono spiegati in dettaglio nel paragrafo "Spiegazione Formule" alla fine del capitolo.

2.2.1.2 Porzione sugli interventi in generale e sulle patologie

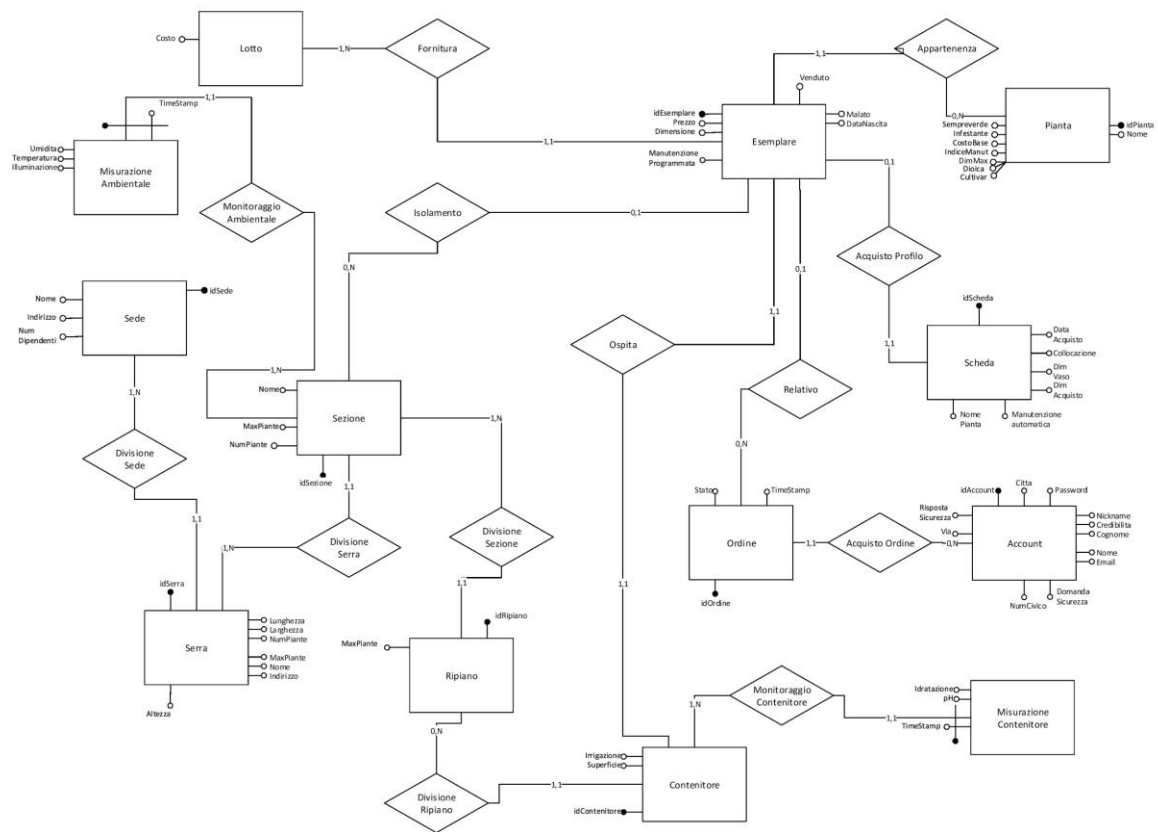


Di seguito, i vincoli non esprimibili tramite lo schema ER.

2.2.1.2.1 Vincoli generici e business rules

- 1) Non si può effettuare un intervento in una certa data se la pianta che lo subisce non tollera tale intervento in tale data
- 2) Se per un agente si ha "Chimico = false", allora tale agente non partecipa alla relazione "Contrasto".

2.2.1.3 Porzione sulla coltivazione in serra

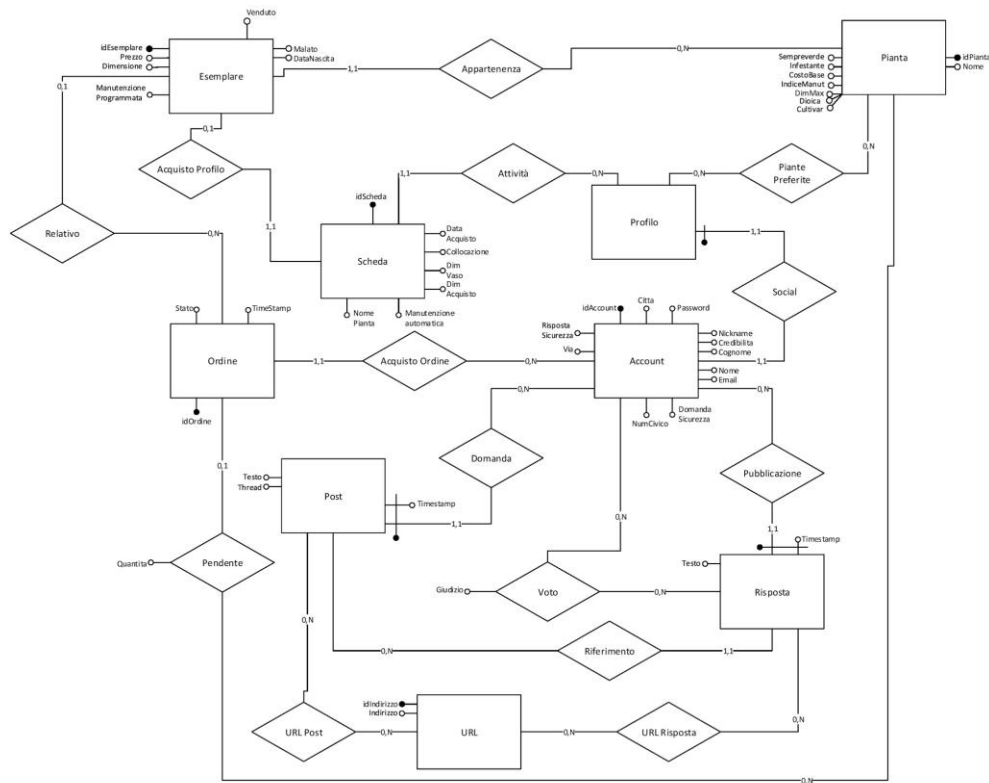


Di seguito, i vincoli non esprimibili tramite lo schema ER.

2.2.1.3.1 Vincoli generici e business rules

- 1) I valori di “NumPiante” e “MaxPiante” in una serra si ricavano per addizione degli attributi omonimi delle sezioni che partecipano a “DivisioneSerra” con quella serra
- 2) Il valore di “NumPiante” in “Sezione” si ricava contando gli esemplari che si trovano in quella sezione
- 3) Un esemplare si trova in isolamento se è malato.

2.2.1.4 Area social e store²



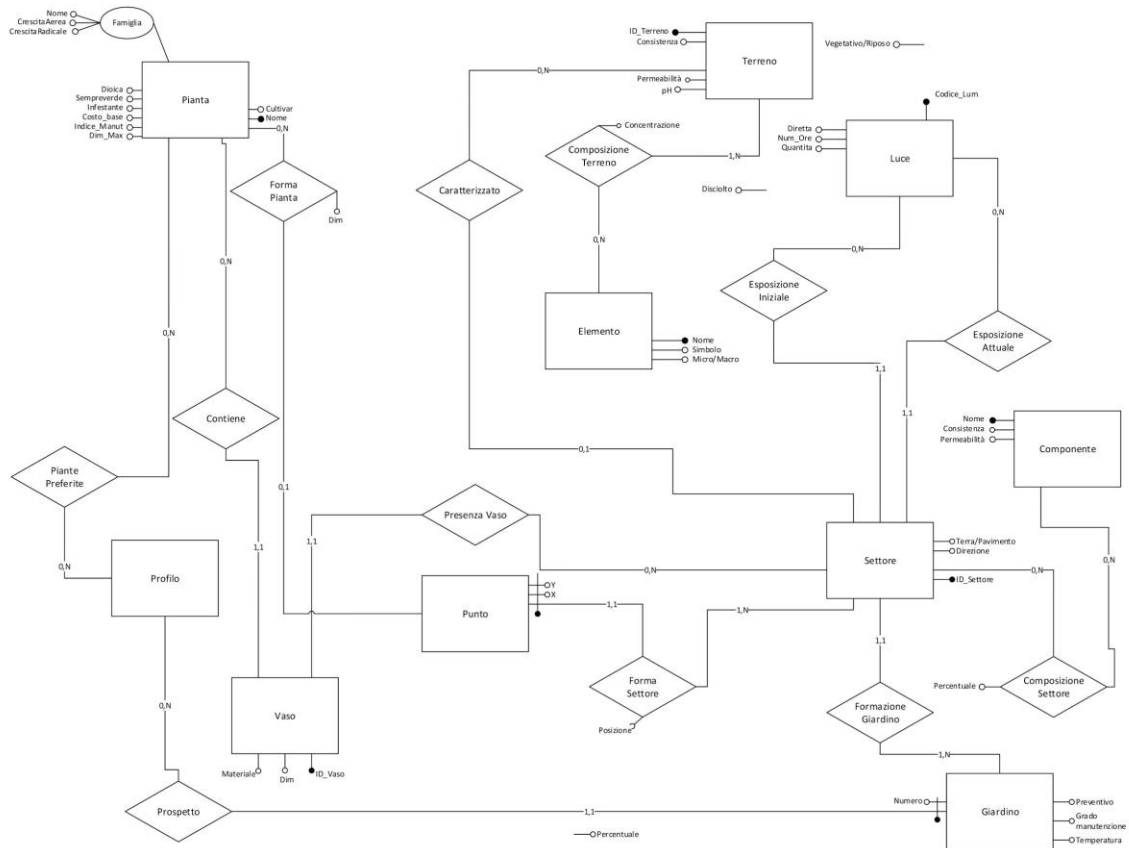
Di seguito, i vincoli non esprimibili tramite lo schema ER.

2.2.1.4.1 Vincoli generici e business rules

- 1) Un account non può votare un suo post o un post che ha già votato precedentemente
- 2) “NomePianta” in “Scheda” si riferisce al nome della specie dell’esemplare relativo a quella scheda
- 3) Il calcolo dell’attributo “Credibilita” in “Account” è spiegato nel dettaglio alla fine del capitolo.

² Viste le dimensioni contenute della porzione relativa allo store, abbiamo deciso di presentare le due sezioni insieme all’interno di questo capitolo

2.2.1.6 Area garden design



Di seguito, i vincoli non esprimibili tramite lo schema ER.

2.2.1.6.1 Vincoli generici e business rules

- 1) L'attributo "Preventivo" in "Giardino" viene calcolato sommando il costo delle piante presenti in quel giardino
- 2) Una pianta può partecipare a "FormaPianta" solo se il settore relativo presenta almeno tre occorrenze in "FormaSettore".

2.2.2 Spiegazione delle associazioni e delle cardinalità

Associazione Relativo

- Cardinalità (1,1) con EsemplareVenduto, in quanto ogni esemplare viene venduto in relazione ad un ordine
- Cardinalità (0,N) con Ordine, in quanto ogni ordine può avere più esemplari ad esso relativi (ma non ne ha nessuno se è pendente)

Associazione Acquisto_Profilo

- Cardinalità (1,1) con EsemplareVenduto, in quanto, nel momento in cui un esemplare viene venduto, viene creata una scheda ad esso relativa
- Cardinalità (1,1) con Scheda, in quanto ogni scheda è associata ad un solo profilo dell'esemplare acquistato

Associazione Attività

- Cardinalità (1,1) con Scheda, in quanto ogni scheda è di un solo profilo/account
- Cardinalità (0,N) con Profilo, in quanto ogni profilo può avere diverse schede (o non averne)

Associazione Acquisto_Ordine

- Cardinalità (1,1) con Ordine, in quanto ogni ordine può essere stato effettuato da un solo account
- Cardinalità (0,N) con Account, in quanto ogni account può effettuare diversi ordini (o non effettuarne)

Associazione Social

- Cardinalità (1,1) con Profilo, in quanto ogni profilo è relativo ad un solo account
- Cardinalità (1,1) con Account, in quanto ogni account è relativo ad un solo profilo

Associazione Piante_Preferite

- Cardinalità (0,N) con Pianta, in quanto ogni pianta può essere indicata come preferita in più profili (oppure nessuno)
- Cardinalità (0,N) con Profilo, in quanto ogni profilo può scegliere diverse piante come preferite (oppure nessuna)

Associazione Domanda

- Cardinalità (1,1) con Post, in quanto ogni post è una domanda fatta da un solo account
- Cardinalità (0,N) con Account, in quanto ogni account può fare diverse domande (o non farne)

Associazione Voto

- Cardinalità (0,N) con Account, in quanto ogni account può votare diverse risposte (oppure nessuna)
- Cardinalità (0,N) con Risposta, in quanto ogni risposta può essere votata da diversi account (oppure da nessuno)

Associazione Pubblicazione

- Cardinalità (1,1) con Risposta, in quanto ogni risposta è pubblicata da un solo account
- Cardinalità (0,N) con Account, in quanto ogni account può pubblicare diverse risposte (o non pubblicarne affatto)

Associazione Riferimento

- Cardinalità (1,1) con Risposta, in quanto ogni risposta è riferita ad un solo post
- Cardinalità (0,N) con Post, in quanto ogni post può ricevere diverse risposte (oppure nessuna)

Attributo multivalore URL

- Cardinalità (0,N) con Risposta/Post, in quanto in ogni risposta/post possono esserci più URL (oppure nessuno)

Associazione Pendente

- Cardinalità (0,1) con Ordine, in quanto ogni ordine può essere pendente con una pianta
- Cardinalità (0,N) con Pianta, in quanto ogni pianta può essere pendente in diversi ordini (oppure nessuno)

Associazione SuggerimentoAcquisti

- Cardinalità (1,N) con Pianta, in quanto per ogni pianta viene suggerita una quantità da acquistare per ogni mese
- Cardinalità (1,1) con ReportConsigliAcquisti, in quanto ogni report è relativo ad una specifica pianta in una specifica data

Associazione Prospetto

- Cardinalità (1,1) con Giardino, in quanto ogni giardino è associato ad un solo profilo
- Cardinalità (0,N) con Profilo, in quanto ogni profilo può fare diversi prospetti di giardino (oppure nessuno)

Associazione Contiene

- Cardinalità (1,1) con Vaso, in quanto ogni vaso contiene una sola pianta
- Cardinalità (0,N) con Pianta, in quanto ogni pianta può essere messa in diversi vasi (oppure in nessuno)

Associazione FormaPianta

- Cardinalità (0,N) con Pianta, in quanto ogni pianta si può trovare in più giardini (oppure in nessuno)
- Cardinalità (0,1) con Punto, in quanto ogni punto può essere associato o ad una pianta o al vertice di un settore

Associazione PresenzaVaso

- Cardinalità (1,1) con Vaso, in quanto ogni vaso è contenuto in un solo settore
- Cardinalità (0,N) con Settore, in quanto ogni settore può contenere diversi vasi (oppure nessuno)

Associazione FormaSettore

- Cardinalità (1,N) con Settore, in quanto ogni settore è formato da diversi punti (ma deve sempre essercene qualcuno affinché sia una forma geometrica)
- Cardinalità (1,1) con Punto, in quanto ogni punto è associato ad un solo settore

Associazione FormazioneGiardino

- Cardinalità (1,1) con Settore, in quanto ogni settore fa parte di un solo giardino
- Cardinalità (1,N) con Giardino, in quanto ogni giardino è composto da diversi settori (ma deve essercene almeno uno)

Associazione ComposizioneSettore

- Cardinalità (0,N) con Settore, in quanto in ogni settore possono trovarsi diversi componenti del terreno (oppure nessuno se è pavimentato)
- Cardinalità (0,N) con Componente, in quanto ogni componente può trovarsi in diversi settori (oppure nessuno)

Associazione EsposizioneAttuale

- Cardinalità (1,1) con Settore, in quanto ogni settore ha una sola tipologia di esposizione di luce (e, ovviamente, non può non averne)
- Cardinalità (0,N) con Luce, in quanto una stessa tipologia di luce può essere ritrovata in diversi settori (oppure in nessuno)

Associazione EsposizioneIniziale

- Cardinalità (1,1) con Settore, in quanto ogni Settore ha una sola tipologia di esposizione iniziale di luce (e, come prima, non può non averne)
- Cardinalità (0,N) con Luce, in quanto una stessa tipologia di luce può essere ritrovata in diversi settori (oppure in nessuno)

Associazione Caratterizzato

- Cardinalità (0,1) con Settore, in quanto ogni settore può essere caratterizzato da un solo terreno (ma non ne ha nessuno se è pavimentato)
- Cardinalità (0,N) con Terreno, in quanto uno stesso terreno può trovarsi in diversi settori

Associazione TerrenoContentitore

- Cardinalità (1,1) con Contentitore, in quanto in ogni contenitore si trova un solo tipo di terreno
- Cardinalità (0,N) con Terreno, in quanto uno stesso tipo di terreno può essere messo in diversi contenitori (oppure in nessuno)

Associazione DivisioneSede

- (1,N) con Sede, in quanto ogni sede ha più serre ma ne ha almeno una
- (1,1) con Serra, in quanto ogni serra è relativa a una sola sede

Associazione DivisioneSerra

- (1,N) con Serra, in quanto ogni serra ha più sezioni, ma ne ha almeno una
- (1,1) con Sezione, in quanto ogni sezione è relativa a una sola serra

Associazione DivisioneSezione

- (1,N) con Sezione, in quanto ogni sezione ha più ripiani, ma ne ha almeno uno
- (1,1) con Ripiano, in quanto ogni ripiano è relativo a una sola sezione

Associazione DivisioneRipiano

- (0,N) con Ripiano, in quanto ogni ripiano contiene più contenitore e può non contenerne
- (1,1) con Contentitore, in quanto ogni contenitore si trova in un solo ripiano

Associazione Sufficienza

- (0,N) con Serra, in quanto per ogni serra esistono più report, ma possono non essercene
- (1,1) con Report, in quanto ogni report è relativo a una sola serra

Associazione Isolamento

- (0,N) con Sezione, in quanto in ogni sezione possono esserci più esemplari in isolamento o non esserci
- (0,1) con EsemplareAzienda, in quanto un esemplare può essere in isolamento in una sezione oppure no

Associazione MonitoraggioAmbientale

- (1,N) con Sezione, in quanto per ogni sezione esistono più misurazioni, ma deve essercene almeno una
- (1,1) con MisurazioneAmbientale, in quanto ogni misurazione è relativa a una sola sezione

Associazione Ospita

- (1,1) con Esemplare, in quanto ogni esemplare è ospitato in un solo contenitore
- (1,1) con Contenitore, in quanto ogni contenitore ospita un solo esemplare

Associazione Appartenenza

- (1,1) con Esemplare, in quanto ogni esemplare appartiene a una sola specie di pianta
- (0,N) con Pianta, in quanto possono esserci più esemplari per una stessa pianta (oppure non essercene)

Associazione MonitoraggioContenitore

- (1,N) con Contenitore, in quanto per ogni contenitore esistono più misurazioni, ma deve essercene almeno una
- (1,1) con MisurazioneContenitore, in quanto ogni misurazione è relativa a un solo contenitore

Associazione InfoContenitore

- (0,1) con MisurazioneContenitore, in quanto ogni misurazione può essere utilizzata in al più un report
- (1,1) con ReportDiagnostica, in quanto ogni report contiene una sola misurazione del contenitore

Associazione Coinvolgimento

- (1,1) con ReportDiagnostica, in quanto ogni report si riferisce ad un solo esemplare
- (0,N) con EsemplareAzienda, in quanto uno stesso esemplare può essere coinvolto in più report oppure in nessuno

Associazione Fornitura

- (1,N) con Lotto, in quanto in uno stesso lotto sono presenti più esemplari, ma almeno uno
- (1,1) con Esemplare, in quanto ogni esemplare si trova in un solo lotto

Associazione Esecuzione

- (0,N) con Esemplare, in quanto per uno stesso esemplare possono esserci più interventi oppure nessuno
- (1,1) con Intervento, in quanto ogni intervento è relativo a un solo esemplare

Associazione Notifica

- (0,1) con Intervento, in quanto non tutti gli interventi vengono notificati e quelli che lo sono vengono notificati una sola volta
- (0,N) con Scheda, in quanto ad una scheda possono essere stati notificati più interventi oppure nessuno

Associazione ComposizioneContenitore

- (1,N) con Contenitore, in quanto in ogni contenitore possono essere presenti più componenti, ma deve essercene almeno uno
- (0,N) con Componente, in quanto ogni componente può trovarsi in più contenitori oppure nessuno

Associazione PresenzaElemento

- (1,N) con MisurazioneContenitore, in quanto ogni misurazione controlla più elementi
- (0,N) con Elemento, in quanto ogni elemento può essere misurato diverse volte o anche nessuna

Associazione InfoSintomi

- (1,N) con ReportDiagnostica, in quanto in ogni report possono essere stati individuati più sintomi (ma deve esserne stato individuato almeno uno)
- (0,N) con Sintomo, in quanto ogni sintomo può essere stato individuato in più report oppure in nessuno

Associazione DiagnosiPossibili

- (1,N) con Report, in quanto ogni report indica più patologie possibili (ma ne indica almeno una)
- (0,N) con Agente, in quanto ogni agente può essere citato in più report oppure in nessuno

Associazione DiagnosiEsperto

- (1,N) con Report, in quanto per ogni report l'esperto formula una diagnosi (ma ne può formulare altre se la prima si rivela sbagliata)
- (0,N) con Agente, in quanto ogni agente può essere citato in più diagnosi oppure in nessuno

Associazione Decisione

- (1,N) con Report, in quanto per ogni report l'esperto decide un trattamento (ma ne può decidere altri se il primo si rivela inefficace)
- (0,1) con InterventoTrattamento, in quanto un trattamento, se è effettuato in relazione ad un report, è relativo a quell'unico report

Associazione TipoInterventoConcimazione

- (1,1) con InterventoConcimazione, in quanto ogni intervento di concimazione è caratterizzato da un unico tipo
- (0,N) con TipoConcimazione, in quanto per ogni tipo di concimazione può esistere più di un intervento che lo riguarda (ma potrebbero anche non esserle)

Associazione TipoInterventoPotatura

- (1,1) con InterventoPotatura, in quanto ogni intervento di potatura è di un tipo specifico
- (0,N) con TipoPotatura, in quanto per ogni tipo di potatura può esistere più di un intervento che lo riguarda (ma potrebbero anche non esserle)

Associazione Trattamento

- (1,1) con InterventoTrattamento, in quanto ogni trattamento viene effettuato contro uno specifico agente
- (0,N) con Agente, in quanto per ogni agente può essere stato effettuato più di un trattamento oppure nessuno

Associazione UtilizzoProdotto

- (0,1) con InterventoTrattamento, in quanto in ogni trattamento si usa un solo prodotto chimico oppure nessuno (se il trattamento è biologico)
- (0,N) con Prodotto, in quanto ogni prodotto può essere stato usato più volte oppure nessuna

Associazione Contrasto

- (1,N) con Prodotto, in quanto ogni prodotto può colpire più agenti, ma deve colpirne almeno uno
- (0,N) con Agente, in quanto ogni agente può essere sensibile a più prodotti oppure a nessuno

Associazione Basato

- (1,N) con Prodotto, in quanto ogni prodotto si basa su più principi attivi, ma almeno uno
- (1,N) con Principio, in quanto ogni principio può essere presente in più prodotti, ma almeno uno

Attributo multivalore PeriodoNonUtilizzo

- (0,N) con Prodotto, perché ogni prodotto può avere più periodi in cui non è utilizzabile oppure nessuno

Associazione Sintomatologia

- (1,N) con Agente, in quanto ogni agente può essere caratterizzato da più sintomi (ma si suppone ne abbia almeno uno)
- (1,N) con Sintomo, in quanto ogni sintomo può essere comune a più patologie (ma si suppone sia caratteristico di almeno una)

Associazione Visivo

- (1,N) con Sintomo, in quanto per ogni sintomo possono essere memorizzate più immagini (ma deve esserne almeno una)
- (1,1) con Immagine, in quanto ogni immagine è relativa ad un solo sintomo

Associazione CondizioniFavorevoliPerAgente

- (0,N) con Agente, in quanto per ogni agente possono essere state rilevate più condizioni favorevoli oppure nessuna
- (1,1) con CondizioniFavorevoli, in quanto ogni set di condizioni favorevoli è memorizzato relativamente ad un solo agente

Associazione ElementiCoinvolti

- (0,N) con Elemento, in quanto uno stesso elemento può favorire più agenti (oppure nessuno)
- (0,N) con CondizioniFavorevoli, in quanto per un agente possono esistere più elementi che lo favoriscono (oppure nessuno)

Associazione Attacco

- (1,N) con Agente, in quanto ci sono possono essere più periodi favorevoli per uno stesso agente
- (1,1) con PeriodoAttacchi, in quanto ogni periodo viene memorizzato in relazione ad un singolo agente

Associazione Vulnerabilità

- (1,1) con PeriodoAttacchi, in quanto ogni periodo viene memorizzato in relazione ad una singola pianta
- (1,N) con Pianta, in quanto ci sono possono essere più periodi di vulnerabilità per una stessa pianta

Associazione PossibilitaPotatura

- (0,N) con Pianta, perché una particolare pianta può tollerare operazioni di potatura in diversi periodi dell'anno (oppure non tollerarne affatto);
- (1,1) con PeriodoPotatura, perché ogni periodo qui si riferisce ad una specifica pianta.

Associazione NecessitaPotatura

- (0,N) con Pianta, perché una particolare pianta può necessitare di più operazioni di potatura in diversi periodi dell'anno (oppure non necessitarne affatto);
- (1,N) con TipoPotatura, perché più piante possono necessitare uno stesso tipo di potatura (ma si suppone che ce ne sia almeno una)

Associazione Effettuabile

- (1,N) con TipoPotatura, perché uno stesso tipo di potatura può essere effettuato in più periodi diversi (ma in almeno uno);
- (1,1) con PeriodoPotatura, perché ogni periodo qui si riferisce ad uno specifico tipo di potatura.

Associazione TolleraRinvaso

- (0,N) con Pianta, perché una particolare pianta può tollerare operazioni di rinvaso in diversi periodi dell'anno (oppure non tollerarne affatto);
- (1,1) con Periodo rinvasi, perché ogni periodo qui si riferisce ad una specifica pianta.

Associazione Necessita

- (1,1) con Pianta, perché ogni pianta ha uno specifico set di esigenze;
- (1,1) con Esigenze, perché, come prima, ogni set di esigenze fa riferimento ad una sola pianta.

Associazione EsigenzeAcqua

- (1,N) con Esigenze, perché le esigenze riguardanti l'acqua per una pianta possono cambiare a seconda del periodo in cui si trova una pianta (ma comunque esiste almeno un tipo di necessità per l'acqua);
- (1,N) con Acqua, perché, per come è stato memorizzato il fabbisogno di acqua, uno stesso set può far riferimento a piante diverse (ma si suppone che nuove occorrenze di acqua vengano inserite solo in relazione all'inserimento di una nuova pianta).

Associazione EsigenzeTemperatura

- (1,1) con Esigenze, perché una pianta ha bisogno di uno specifico range di temperature per poter vivere;
- (1,N) con Temperatura, perché, come è stato detto per "Acqua", uno stesso range di temperature può far riferimento a più piante.

Associazione EsigenzeLuce

- (1,N) con Esigenze, perché dipendentemente dal periodo di una pianta, le sue esigenze riguardanti la luce possono cambiare (ma comunque esiste almeno un tipo di necessità per la luce);
- (0,N) con Luce, perché stessi set di luce possono far riferimento a diverse piante; inoltre, un set di luce può essere inserito in relazione ad un "Settore", ma non riferirsi ad alcuna pianta.

Associazione EsigenzeConcimazione

- (0,N) con Esigenze, perché una pianta può necessitare più concimazioni (oppure nessuna);
- (1,N) con Tipo Concimazione, perché uno stesso tipo di concimazione può essere necessario a più piante (ma si suppone non vi siano tipi di concimazioni non necessari da alcuna pianta).

Associazione UtilizzoElemento

- (1,N) con Tipo Concimazione, perché una concimazione può contenere diversi elementi (ma ne contiene almeno uno);
- (0,N) con Elemento, perché un determinato elemento può essere usato in diverse concimazioni (oppure in nessuna).

Associazione Singolo

- (1,N) con Tipo Concimazione, perché un tipo di concimazione può richiedere più interventi separati (ma ne richiede almeno uno);
- (1,1) con Concimazione, perché una concimazione può essere di un solo tipo.

Associazione Quantita

- (1,N) con Concimazione, perché una concimazione può usare diversi elementi (ma ne usa almeno uno);
- (0,N) con Elemento, perché un elemento può essere usato in diverse concimazioni (oppure in nessuna).

Associazione EsigenzeTerreno

- (1,1) con Esigenze, perché ogni pianta predilige uno specifico tipo di terreno;
- (0,N) con Terreno, perché uno stesso terreno può essere adatto a più piante.

Associazione ComposizioneTerreno

- (1,N) con Terreno, perché in ogni terreno possono essere presenti più elementi (ma deve essercene almeno uno);
- (0,N) con Elemento, perché uno stesso elemento può trovarsi in più terreni (oppure in nessuno).

Associazione Cicli Pianta

- (1,N) con Pianta, perché una pianta può avere un solo periodo vegetativo/di riposo o averne diversi durante l'anno;
- (1,N) con Periodo Cicli, perché piante diverse possono essere caratterizzate da stessi periodi vegetativi/di riposo (ma non vengono inserite nuove occorrenze di "periodo cicli" se non esistono piante a cui riferirli).

Associazione Preferenze Periodi

- (0,N) con Periodo Cicli, perché un utente può esprimere nessuna o più preferenze riguardanti i periodi dei cicli di una pianta;
- (0,N) con Preferenze, perché, per un dato periodo, potrebbero essere arrivate nessuna o più preferenze da parte degli utenti.

Associazione Preferenza Utente

- (0,N) con Profilo, perché un utente può non esprimere preferenze o esprimerne diverse;
- (1,1) con Preferenze, perché ogni set di preferenze è relativo ad un solo profilo.

Associazione Preferenza Terreno

- (0,1) con Preferenze, perché in un set di preferenze un utente può impostare al più un tipo di terreno;
- (0,N) con Terreno, perché un determinato tipo di terreno può non ricevere alcuna preferenza da parte degli utenti, come potrebbe riceverne più di una.

2.2.3 Spiegazione delle formule

Di seguito è riportata la spiegazione di alcune delle formule utilizzate all'interno del database per gestire le varie richieste del progetto.

2.2.3.1 Credibilità di un utente

Ogni utente parte da un punteggio base di 100. Per ogni post aperto, guadagna 2 punti, mentre per ogni risposta data ne guadagna 5. In più, ogni volta che riceve un voto, la sua credibilità cambia come segue:

- $\text{CredibilitàGuadagnata} = 1 + i * (-2/-1/+1/+3/+5)$

Dove il moltiplicatore di i viene assegnato in base al giudizio (da 1 a 5 stelle), mentre i viene calcolato in questa maniera:

- $i = [\text{PunteggioDelVotante} / 10000] + 1$

La variabile i può dunque essere intesa come il livello di un utente, il quale aumenta ogni 10000 punti guadagnati. Di seguito due stime sull'andamento del punteggio di un utente medio e di un "troll":

1)Punteggio/mese per un utente mediamente attivo

- 15 post aperti - +30 punti
- 60 risposte - +300 punti
- Voto medio di 3,5 (con una media di 10 voti ricevuti) per ogni risposta - +1200 punti
- Totale: +1530 punti

Quindi un utente medio passa al livello successivo ogni 6/7 mesi.

2)Punteggio/mese per un "troll"

- 30 post aperti - +60 punti
- 150 risposte - +750 punti
- Voto medio di 1 (con una media di 10 voti ricevuti) per ogni risposta - -3000 punti
- Totale: -2190

Un troll viene invece rapidamente individuato dagli utenti, così da poter essere eventualmente ammonito o espulso dal forum.

2.2.3.2 Entità/Costo di un intervento

Per calcolare l'entità e il costo (indicativi) di un intervento si usa il seguente procedimento. Ad ogni tipo di intervento si associa un entità base E_B come segue:

- $E_B = 1$ per le concimazioni
- $E_B = 3$ per i rinvasi, i trattamenti e le piantumazioni
- $E_B = 5$ per le potature

Poi, in base alla dimensione D dell'esemplare, l'entità totale E_{tot} di un intervento diventa:

- $E_{tot} = (E_B * D)/2$

Il costo C viene calcolato in proporzione all'entità:

- $C = E_{tot} * 5$

Questi valori sono comunque solo indicativi, infatti la valutazione finale è a carico degli addetti, i quali si preoccuperanno di fare un sopralluogo per confermare o correggere le stime.

2.2.3.3 Andamento della crescita in funzione dell'indice di accrescimento

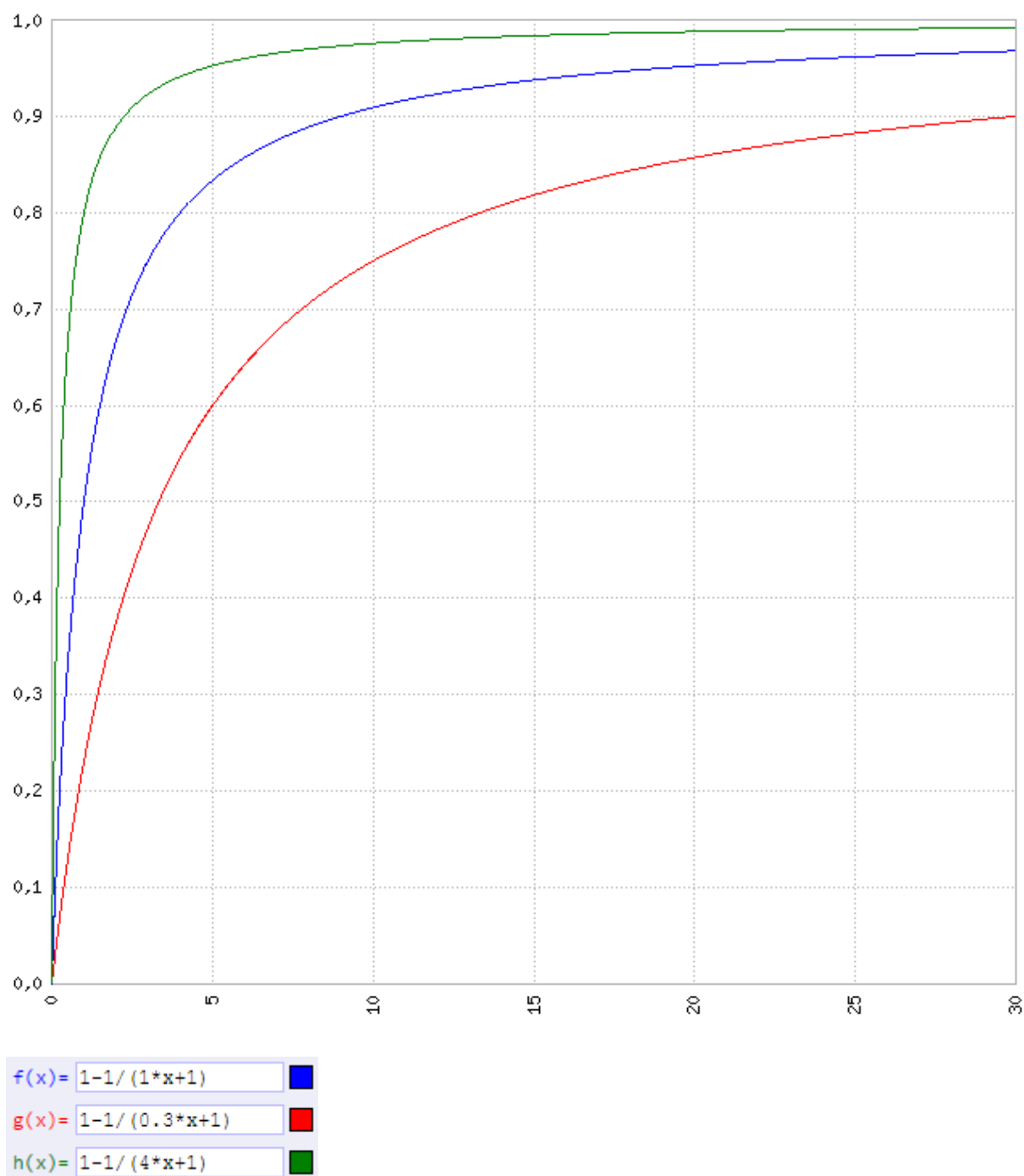
Presa una pianta, a partire dal suo indice di accrescimento, dalla sua dimensione attuale e dai mesi di vita della pianta, siamo in grado di calcolare la sua dimensione attuale grazie alla seguente formula:

- $f(x) = 1 - 1/(ax + 1)$

dove x è il numero di mesi di vita, a l'indice di accrescimento e $f(x)$ il rapporto fra dimensione attuale e dimensione massima di tale pianta. Da notare che 1 è un valore asintotico per $f(x)$, cosa volutamente ottenuta per esprimere il fatto che la crescita di una pianta (se non ostacolata) non si ferma mai, ma rallenta semplicemente all'aumentare dell'età. Comunque, si considera raggiunta la maturità per

$$f(x) \geq \frac{12}{13}$$

questo risultato si raggiunge dopo un anno per una pianta con indice di accrescimento pari a 1, quindi tale valore viene considerato come valore di riferimento.



2.2.3.4 Calcolo di un terreno a partire dalle sue componenti

In alcuni casi, può essere utile calcolare qual è la consistenza e la permeabilità complessiva di un terreno a partire dalle componenti di cui sappiamo essere composto. Nello specifico, all'interno del nostro database, abbiamo previsto questa possibilità nel caso dei settori di un giardino (così non è necessario per un utente sapere le caratteristiche esatte del proprio terreno) e nel caso dei contenitori (per accelerarne l'inserimento).

Il procedimento che abbiamo usato è il seguente:

- Calcoliamo:

$$V = \Sigma (P \times I)$$

dove P è la percentuale con cui la componente è presente nel terreno e I è un indice ottenuto secondo la seguente tabella a seconda della consistenza e della permeabilità:

Consistenza	Permeabilità
Solida → +2	Alta → +2
SemiSolida → +1	Media → +1
Plastica → -1	Bassa → 0
Liquida → -2	MoltoBassa → -1
	Impermeabile → -2

- Ottenuto V si associano le caratteristiche finali al terreno nella seguente maniera:

Per la consistenza:

V	Valore finale
Fra 100 e 200	Solida
Fra 0 e 100	SemiSolida
Fra -100 e 0	Plastica
Fra -200 e -100	Liquida

Per la permeabilità:

V	Valore finale
Fra 120 e 200	Alta
Fra 40 e 120	Media
Fra -40 e 40	Bassa
Fra -120 e -40	MoltoBassa
Fra -200 e -120	Impermeabile

Da notare che, qualunque sia il caso, il pH va inserito a parte.

2.2.3.5 Calcolo indice di manutenzione

Data una pianta, il suo indice di manutenzione viene calcolato tenendo in considerazione la probabilità di subire attacchi, di avere carenze di elementi o acqua, di essere esposta a temperature non tollerate, nonché in base al numero di potature e concimazioni già previste dall'azienda.

Il calcolo effettivo, detto *I* l'indice di manutenzione, avviene nel seguente modo:

1. Si pone $I = P_C + P_E + P_A + P_T$, dove:

- $P_C = N_C / 10$
dove N_C è il numero di concimazioni necessarie
- $P_E = N_E / 10$
dove N_E è il numero degli elementi necessari
- $P_A = P / 10$
dove P è la periodicità delle annaffiature nel periodo vegetativo (calcolata su 10 giorni)
- $P_T = 10 / (T_{MAX} - T_{MIN})$
dove T_{MAX} e T_{MIN} sono la temperatura massima e la minima

2. Si somma ad *I* il numero di potature e il numero di trattamenti necessari in un anno (i trattamenti necessari sono quelli preventivi da fare nei periodi in cui la probabilità di un attacco supera il 50%)

3. Si divide *I* per l'indice di accrescimento (in generale le piante che crescono più lentamente sono più complesse e necessitano cure più particolari).

Considerando per una pianta media i seguenti valori:

- $P_{AT} = 0.2$ (2 concimazioni necessarie)
- $P_E = 0.4$ (4 elementi necessari)
- $P_A = 0.3$ (3 annaffiature ogni 10 giorni ~ 2 volte/settimana)
- $P_T = 0.3$ (30° di range per le temperature)
- Un intervento di potatura all'anno
- Due trattamenti necessari
- Indice di accrescimento pari a 1

Si ottiene un indice di manutenzione pari a 4.2, che verrà quindi considerato il valore di riferimento.

2.2.3.6 Luminosità di un settore

All'inserimento (o all'aggiornamento o alla cancellazione) di una pianta in un settore, bisogna ricalcolare la luce attuale del settore stesso. Per farlo, usiamo la seguente formula:

- Si prende la luce iniziale del settore (passata dall'utente)
- Se, con la somma delle aree delle piante (approssimate a cerchi), si supera il 50% dell'area totale, la luce passa automaticamente a indiretta e la sua quantità diventa 'Media'

- Se la somma supera l'80%, la quantità diventa 'Bassa'.

2.2.3.7 Distanza minima fra due piante

Per calcolare la dimensione minima fra due piante utilizziamo il seguente procedimento:

$$d = (D \times i) + (D' \times i') + D_m \times \Sigma(Q_m) / 10$$

dove:

- D e D' sono le dimensioni delle due piante
- i e i' sono due indici che valgono rispettivamente:
1, se la pianta associata non è infestante;
5, altrimenti
- D_m è la dimensione della pianta più grande
- Q_m è, preso un elemento necessario a entrambe le piante, la quantità maggiore con cui questo elemento è richiesto dalle due piante (le cifre sono scelte così che, per due piante nella media, la distanza aumenta di circa 10 centimetri per ogni elemento).

3. Ristrutturazione del diagramma E-R

3.1 Introduzione

Dopo aver completato lo schema ER, abbiamo effettuato la ristrutturazione in modo tale da eliminare i costrutti non traducibili nel modello relazionale, ovvero attributi multivalori, composti e generalizzazioni. In più, abbiamo effettuato qualche altra piccola modifica.

Di seguito è riportata la lista, mentre in allegato c'è lo schema ER ristrutturato.

3.2 Modifiche effettuate

1. Abbiamo trasformato l'attributo composto multivalore "Periodo non utilizzo" in un'entità omonima a parte, legata a "Prodotto" dalla relazione "Inutilizzo"
2. Abbiamo trasformato l'attributo composto "Famiglia" in un'entità omonima a parte, legata a "Pianta" dalla relazione "Famiglia Pianta"
3. Abbiamo trasformato gli attributi multivalore "URL" in un'entità omonima a parte legata a "Post" e "Risposta" dalle relazioni "URL Post" e "URL Risposta" con cardinalità (0,N) (perché un URL può essere relativo o a un post o a una risposta)
4. Abbiamo separato le relazioni "EsigenzeAcqua" e "EsigenzeLuce" in "EsigenzeAcquaVegetativo", "EsigenzeAcquaRiposo", "EsigenzeLuceVegetativo", "EsigenzeLuceRiposo" tutte con cardinalità (1,1) con "Esigenze" e (0,N) con "Acqua" e "Luce".

3.3 Generalizzazioni

3.3.1 Esempare

Poiché ci interessano per lo più le informazioni relative ad "Esemplare" in sé, e la presenza di valori NULL e dell'attributo "Venduto" non rappresenta un peso in memoria eccessivo, abbiamo deciso di accorpare i figli sul padre;

In seguito a questa decisione, tutte le relazioni collegate a "EsemplareAzienda" o "EsemplareVenduto" sono state collegate a "Esemplare" e la loro cardinalità minima, qualora non lo fosse già, è stata cambiata in 0. Per distinguere i diversi esemplari è stato introdotto un nuovo attributo booleano "venduto", il cui peso in memoria è pressoché trascurabile. Gli attributi "malato" e "manutenzione programmata" sono stati trasferiti sull'entità padre. Anche qui, il peso in memoria (dovuto alla presenza di valori NULL) è pressoché trascurabile (circa 1 MB).

3.3.2 Intervento

Poiché anche qui le informazioni più utilizzate ed interessanti erano quelle relative ad “Intervento” in sé, e vista la scarsa occupazione di memoria dovuta alla presenza dell’attributo “Tipo”, abbiamo deciso di accorpare i figli sul padre.

Come sopra, tutte le relazioni collegate alle entità figlie di “Intervento” sono state collegate all’entità padre e la loro cardinalità minima è stata portata a 0. Per distinguere i vari tipi di intervento si usa una stringa, più pesante degli attributi booleani di “Esemplare”, ma comunque trascurabile ai fini del database (circa 25MB, vista una media di 10 byte ad attributo; inoltre a questi vanno tolti i 10MB che comunque avremmo dovuto usare per memorizzare le chiavi dei singoli tipi di intervento, ottenendo quindi un valore utilizzabile).

4. Analisi delle operazioni sui dati

4.1 Introduzione

Prima di procedere all'analisi delle ridondanze, abbiamo effettuato alcune valutazioni sui dati. Nello specifico, abbiamo prima costruito una tavola dei volumi con le nostre stime sulla quantità delle informazioni passanti per il database nel corso di un anno. Una descrizione delle stime che abbiamo fatto in partenza e dei procedimenti usati per valutare i restanti dati in funzione di tali stime è allegata alla documentazione.

Terminata questa parte, abbiamo individuato le quindici operazioni secondo noi più costose e interessanti per l'analisi delle prestazioni del sistema. L'efficienza di tali operazioni è stata valutata in funzione della tavola dei volumi a cui abbiamo accennato prima.

Di seguito la lista delle operazioni accompagnate da una descrizione. La tavola degli accessi, le porzioni di schema ER interessate e le stime sulle frequenze si trovano in allegato alla documentazione.

4.2 Descrizione delle operazioni frequenti

4.2.1 Calcolo degli esemplari venduti di un tipo di pianta

Data in ingresso una pianta, si vuole sapere quanti esemplari sono stati venduti di un determinato tipo di pianta.

4.2.2 Creazione di un nuovo ordine (non pendente)

Dato un esemplare, si vuole memorizzare le informazioni relative alla vendita.

4.2.3 Calcolo della città in cui effettuare un intervento

Dato l'id di un intervento, si vuole recuperare la città in cui si trova (per effettuare le operazioni di ottimizzazione).

4.2.4 Recupero delle informazioni riguardanti gli elementi necessari ad una pianta

Dato il nome di una pianta, si vuole sapere di quali elementi necessita e in quali concentrazioni (per poter confrontare i dati con le misurazioni e per la valutazione dei terreni).

4.2.5 Invio delle notifiche

Dato l'id di un intervento, si vogliono ricavare le informazioni (l'email) necessarie per poter inviare una notifica all'utente interessato.

4.2.6 Inserimento di un nuovo intervento

Date le informazioni su un intervento, lo si vuole inserire nell'entità omonima.

4.2.7 Valutazione del costo/ dell'indice di manutenzione di un giardino

Dato l'id di un giardino, si vuole saperne il costo e l'indice di manutenzione.

4.2.8 Inserimento di una pianta in un giardino

Dato il nome di una pianta e l'id di un settore, si vuole inserire tale pianta in quel settore.

4.2.9 Lettura delle misurazioni relative ad un report di diagnostica

Dato un report, si vogliono leggere le informazioni relative al contenitore e alla sezione interessate per poter effettuare delle analisi sulle condizioni favorevoli alle malattie.

4.2.10 Creazione di un report di diagnostica

Dato l'id di un esemplare che si è ammalato, si vuole inserire nel report di diagnostica relativo le informazioni sul contenitore e sulla sezione in cui l'esemplare si trovava al momento della diagnosi.

4.2.11 Valutazione sul riempimento di una serra

Dato l'id di una serra, si vuole sapere il numero di piante in essa presenti.

4.2.12 Valutazione sul riempimento di una sezione

Dato l'id di una sezione, vogliamo sapere il numero di piante in essa presenti.

4.2.13 Aggiornamento del numero di piante

Dato l'id di una sezione, vogliamo aggiornare il numero di piante presenti (sia in sezione che in serra).

4.2.14 Creazione di una nuova scheda

Dato un esemplare, creare la scheda relativa; nello specifico, si vuole sapere a quale specie appartiene l'esemplare in questione per inserire l'informazione all'interno della scheda.

4.2.15 Accesso alle informazioni di una scheda

Data una scheda, risalire alla specie cui appartiene l'esemplare relativo a tale scheda.

5. Analisi delle ridondanze

5.1 Introduzione

Definite le operazioni principali da effettuare sul database, ci siamo occupati delle ridondanze, sia di quelle già presenti nello schema ER, sia di quelle da noi volutamente inserite per migliorare le prestazioni del database.

Sia le ridondanze già presenti, sia quelle inserite successivamente si trovano di seguito.

In allegato alla documentazione si possono trovare le valutazioni dettagliate riguardanti le prestazioni del sistema in funzione delle ridondanze.

5.2 Lista delle ridondanze

5.2.1 Attributo “Venduto” in “Esemplare

Quest’attributo, con cui distinguiamo gli esemplari venduti da quelli in magazzino, rappresenta una ridondanza poiché questa stessa informazione può essere recuperata controllando se l’esemplare in questione partecipa alla relazione “Relativo”.

Come si può vedere nella documentazione in allegato, dall’analisi del costo delle operazioni 1 e 2 e dell’occupazione della memoria si è deciso di tenere questo attributo.

All’interno del database, la ridondanza è gestita dalle procedure “ProcedureCreazioneNuovoOrdine” e “ProcedureAggiornamentoOrdinePendente” (vedere il capitolo 9 per i dettagli).

5.2.2 Attributo “Città” in “Intervento”

Quest’attributo, non presente nello schema ER, è stato pensato da noi al fine di accelerare le operazioni di ottimizzazione in fase di gestione degli interventi.

Rappresenta una ridondanza poiché questa stessa informazione può essere recuperata tramite accedendo alle informazioni dell’account che ha acquistato l’esemplare oggetto dell’intervento.

Come si può vedere nella documentazione allegata, dall’analisi del costo delle operazioni 3 e 6 e dell’occupazione della memoria si è deciso di non inserire quest’attributo.

5.2.3 Relazione “EsigenzeElemento”

Questa relazione, non presente nello schema ER, è stata pensata da noi al fine di accelerare il recupero di informazioni sulle esigenze di una pianta.

Rappresenta una ridondanza poiché questa stessa informazione può essere recuperata passando per “Terreno” e controllando quali elementi partecipano a “ComposizioneTerreno”.

Come si può vedere nella documentazione allegata, dall’analisi del costo dell’operazione 4 e dell’occupazione della memoria si è deciso di introdurre questa ridondanza³.

La nuova associazione ha le seguenti cardinalità:

- Cardinalità (1,N) con “Pianta”, in quanto ogni pianta può aver bisogno di diversi elementi (ma si suppone che ne necessiti almeno uno)

³ Da notare che questa ridondanza non necessita quasi mai di aggiornamenti, per cui è pressoché priva di svantaggi.

- Cardinalità (0,N) con “Elemento”, in quanto uno stesso elemento può essere richiesto da più piante (oppure da nessuna)

All'interno del database la ridondanza è gestita da “EventControlloValiditaEsigenzeElemento”, che periodicamente controlla la coerenza delle informazioni, e dal “TriggerAggiornaEsigenzeTerreno”.

5.2.4 Relazione “Notifica”

Questa relazione fra “Intervento” e “Scheda” era stata pensata per accelerare l'invio di notifiche riguardanti interventi programmati e automatici.

Rappresenta una ridondanza poiché, per raggiungere le informazioni sull'utente, si può accedere alle informazioni presenti in “Esemplare”.

Come si può vedere nella documentazione allegata, dall'analisi del costo delle operazioni 5 e 6 e dell'occupazione della memoria si è deciso di non tenere questa ridondanza.

5.2.5 Attributo “Preventivo” in “Giardino”

Quest'attributo in “Giardino” era stato pensato per accelerare l'accesso alle informazioni sul costo di un giardino.

Rappresenta una ridondanza in quanto può essere ottenuto sommando i costi delle piante presenti all'interno del giardino stesso.

Come si può vedere nella documentazione allegata, dall'analisi del costo delle operazioni 7 e 8 e dell'occupazione della memoria si è deciso di eliminare questa ridondanza.

5.2.6 Relazione “InfoContenitore”

Questa relazione fra “ReportDiagnostica” e “MisurazioneContenitore” ci permette di accedere velocemente alle informazioni sulle misurazioni riguardanti i contenitori degli esemplari oggetto dei report.

Rappresenta una ridondanza in quanto, passando per “Esemplare” e “Contenitore”, si possono recuperare le stesse informazioni.

Come si può vedere nella documentazione allegata, dall'analisi del costo delle operazioni 9 e 10 è risultato che la ridondanza non portava né benefici né svantaggi dal punto di vista prestazionale, perciò si è deciso di eliminarla per via dell'occupazione, a questo punto inutile, di memoria.

5.2.7 Relazione “InfoAmbientale”

Questa relazione, fra “ReportDiagnostica” e “MisurazioneSezione”, non è presente all'interno dello schema. In maniera simile alla precedente ridondanza, si era pensato di introdurla allo scopo di accelerare il recupero di informazioni sulle misurazioni riguardanti esemplari oggetto di report.

Rappresenta una ridondanza poiché, passando per “Esemplare”, “Contenitore” e “Ripiano”, si possono recuperare le stesse informazioni.

Come si può vedere nella documentazione allegata, dall'analisi del costo delle operazioni 9 e 10 e dell'occupazione della memoria si è deciso di introdurre questa ridondanza.

La nuova associazione ha le seguenti cardinalità:

- Cardinalità (0,N) con “MisurazioneAmbientale”, in quanto ogni può essere utilizzata in più report (oppure in nessuno)
- Cardinalità (1,1) con “ReportDiagnostica”, in quanto in ogni report viene usata una sola misurazione ambientale

All'interno del database, questa ridondanza è gestita dal "TriggerControlloValiditaReportDiagnostica".

5.2.8 Attributo "NumPiante" in "Serra"

Quest'attributo in "Serra" era stato pensato per permettere di distribuire facilmente i vari esemplari in maniera omogenea attraverso le varie serre.

Rappresenta una ridondanza poiché la stessa informazione può essere recuperata contando gli esemplari presenti nelle serre.

Come si può vedere nella documentazione allegata, dall'analisi del costo delle operazioni 11 e 13 e dell'occupazione della memoria si è deciso di mantenere questa ridondanza.

All'interno del database, questa ridondanza, così come la successiva è controllata dai vari trigger "TriggerAggiornaNumeroPiante".

5.2.9 Attributo "NumPiante" in "Sezione"

Come prima, quest'attributo era stato pensato per permettere di distribuire gli esemplari in maniera uniforme fra le varie sezioni.

Rappresenta una ridondanza poiché la stessa informazione può essere recuperata contando gli esemplari presenti nelle sezioni.

Come si può vedere nella documentazione allegata, dall'analisi del costo delle operazioni 12 e 13 e dell'occupazione della memoria si è deciso di mantenere questa ridondanza.

Come detto prima, anche questa ridondanza è gestita dai vari trigger "TriggerAggiornaNumeroPiante".

5.2.10 Attributo "NomePianta" in "Scheda"

Quest'attributo in "Scheda" è stato inserito per rispondere ad una richiesta specifica del progetto, ovvero la presenza, all'interno di una scheda, del nome della specie dell'esemplare a cui la scheda stessa fa riferimento.

Rappresenta una ridondanza in quanto la stessa informazione può essere recuperata passando per "Esemplare" e per la relazione "Appartenenza".

Come si può vedere nella documentazione allegata, dall'analisi del costo delle operazioni 14 e 15 e dell'occupazione della memoria si è deciso di mantenere questa ridondanza.

All'interno del sistema questa ridondanza è garantita dal trigger "TriggerCreazioneNuovaScheda".

5.2.11 Attributo "DataAcquisto" in "Scheda"

Anche quest'attributo in "Scheda" è stato inserito per rispondere ad una richiesta specifica del progetto.

Rappresenta una ridondanza in quanto la stessa informazione può essere recuperata passando per "Esemplare" e per "Ordine".

Come si può vedere nella documentazione allegata, dall'analisi del costo delle operazioni 13 e 14 e dell'occupazione della memoria si è deciso di mantenere questa ridondanza.

Come prima, questa ridondanza è gestita dal trigger "TriggerCreazioneNuovaScheda".

6. Progettazione logica

6.1 Traduzione dello schema ER in modello relazionale

6.1.1 Introduzione

Dopo aver completato la ristrutturazione, siamo passati alla progettazione logica, trasformando entità e associazioni dello schema ER in strutture del modello relazionale. In accompagnamento alla lista delle tabelle, abbiamo allegato una spiegazione degli attributi esterni e dei domini dei vari valori.

6.1.2 Lista delle tabelle

- **PIANTA**(idPianta, Nome, Cultivar, Dioica, DimMax, IndiceManut, CostoBase, Infestante, Sempreverde, Famiglia)

References:

Famiglia → idFamiglia (Famiglia)

Attributi:

idPianta → auto_increment

Nome → varchar

Dioica → bool

IndiceManut → decimal

Infestante → bool

Sempreverde → bool

DimMax → double

- **FAMIGLIA**(idFamiglia, Nome, CrescitaAerea, CrescitaRadicale)

Attributi:

idFamiglia → auto_increment

Nome → varchar

CrescitaAerea → double

CrescitaRadicale → double

- **REPORTCONSIGLIACQUISTO**(Pianta, Data, Note, Quantita)

References:

Pianta → idPianta (Pianta)

Attributi:

Pianta → int

Data → date

Note → varchar

Quantita → int

Note:

1. Ci si scrive se una pianta ha avuto una forte (de)crescita nello scorso mese; nel report di dicembre si segnalano anche le piante che risultano più vulnerabili alle malattie

- **PERIODOPOTATURA**(Pianta, InizioPeriodo, FinePeriodo, TipoPotatura)

References:

Pianta → Nome (Pianta)

TipoPotatura → Nome (TipoPotatura)

Attributi:

Pianta → int

InizioPeriodo → int

FinePeriodo → int

TipoPotatura → int

Note:

1. InizioPeriodo ≤ FinePeriodo
2. Se un periodo è per esempio ott-feb diventa ott-dic + gen-feb
3. I periodi non possono sovrapporsi

- **TIPOPOTATURA**(idTipoPotatura, Nome)

Attributi:

idTipoPotatura → auto_increment

Nome → varchar

- **NECESSITA POTATURA**(Pianta, TipoPotatura, Quantita)

References:

Pianta → idPianta (Pianta)

TipoPotatura → idTipoPotatura (TipoPotatura)

Attributi:

TipoPotatura → int

Pianta → int

Quantita → int

Note:

1. Deve esistere una occorrenza di Pianta, TipoPotatura in PeriodoPotatura
2. Quantita indica il numero di potature di un determinato tipo che una pianta richiede annualmente

- **PERIODOATTACCHI**(Pianta, InizioPeriodo, FinePeriodo, Agente, Entita, Probabilita)

References:

Pianta → idPianta (Pianta)

Agente → idAgente (Agente)

Attributi:

Pianta → int

InizioPeriodo → int

FinePeriodo → int

Agente → int

Entita → int

Probabilita → int

Note:

1. InizioPeriodo ≤ FinePeriodo
2. Se un periodo è per esempio ott-feb diventa ott-dic + gen-feb
3. I periodi non possono sovrapporsi

- **PERIODO**RINVASI(Pianta, InizioPeriodo, FinePeriodo)

References:

Pianta → idPianta (Pianta)

Attributi:

Pianta → int

InizioPeriodo → int

FinePeriodo → int

Note:

1. InizioPeriodo ≤ FinePeriodo
2. Se un periodo è per esempio ott-feb diventa ott-dic + gen-feb
3. I periodi non possono sovrapporsi

- **CICLI**PIANTA(Pianta, Periodo)

References:

Pianta → idPianta (Pianta)

Periodo → idPeriodo (Periodo)

Attributi:

Pianta → int

Periodo → int

- **PERIODO**CICLI(idPeriodo, InizioPeriodo, FinePeriodo, Fio_Fru, Vegetativo)

Attributi:

idPeriodo → auto_increment

InizioPeriodo → int

FinePeriodo → int

Fio_Fru → varchar

Vegetativo → bool

Note:

1. InizioPeriodo <= FinePeriodo
2. Se un periodo è per esempio ott-feb diventa ott-dic + gen-feb
3. I periodi non possono sovrapporsi
4. Se il periodo è di riposo Fio_Fru = "Nessuno"
5. Fio_Fru - "Floritura", "Fruttificazione", "Entrambi", "Nessuno"

- **ESIGENZE**(Pianta, Terreno, Temperatura, AcquaVegetativo, AcquaRiposo, LuceVegetativo, LuceRiposo)

References:

Pianta - idPianta (Pianta)
Terreno - idTerreno (Terreno)
Temperatura - idTemp (Temperatura)
AcquaVegetativo - idAcqua (Acqua)
AcquaRiposo - idAcqua (Acqua)
LuceVegetativo - idLuce (Luce)
LuceRiposo - idLuce (Luce)

Attributi:

Pianta - int
Terreno - int
Temperatura - int
AcquaVegetativo - int
AcquaRiposo - int
LuceVegetativo - int
LuceRiposo - int

- **ACQUA**(idAcqua, Quantita, Periodicita)

Attributi:

idAcqua - auto_increment
Quantita - varchar
Periodicita - int

Note:

1. Quantita - "Basso", "Medio", "Alto"

2. Periodicità → Indica quante volte bisogna dare un determinato quantitativo di acqua ad una pianta settimanalmente

- **TEMPERATURA**(idTemp, TempMin, TempMax)

Attributi:

idTemp → auto_increment

TempMin → int

TempMax → int

Note:

1. I valori di TempMin e di TempMax sono da considerarsi in Gradi Celsius
2. TempMin < TempMax

- **LUCE**(idLuce, Diretta, NumOre, Quantita)

Attributi:

idLuce → auto_increment

Diretta → bool

NumOre → int

Quantita → varchar

Note:

1. NumOre → Numero minimo di ore di luce che una pianta richiede giornalmente
2. Quantita → “Bassa”, “Media”, “Alta”

- **ESIGENZECONCIMAZIONE**(Pianta, TipoConcimazione)

References:

Pianta → idPianta (Pianta)

TipoConcimazione → idConcimazione (Concimazione)

Attributi:

Pianta → int

TipoConcimazione → int

- **TERRENO**(idTerreno, Consistenza, Permeabilita, pH)

Attributi:

idTerreno → auto_increment
Consistenza → varchar
Permeabilita → varchar
pH → varchar

Note:

1. Consistenza → "Liquida", "Plastica", "SemiSolida", "solida"
2. pH → "Fortemente acido" (<5,5), "Acido" (5,5 - 6,0), "SubAcido" (6,0 - 6,8), "Neutro" (6,8 - 7,3), "SubBasico" (7,3 - 8,0), "Basico" (8,0 - 8,5), "Alcalino" (>8,5)
3. Permeabilita → "Alta" ($>10^{-3}$), "Media" (10^{-3} - 10^{-5}), "Bassa" (10^{-5} - 10^{-7}), "Molto bassa" (10^{-7} - 10^{-9}), "Impermeabile" ($<10^{-9}$)

- **COMPOSIZIONE TERRENO**(Terreno, Elemento, Concentrazione)

References:

Terreno → idTerreno (Terreno)
Elemento → idElemento (Elemento)

Attributi:

Terreno → int
Elemento → int
Concentrazione → int

Note:

1. Concentrazione → La concentrazione di un elemento è espressa in mg/m³

- **ELEMENTO**(idElemento, Nome, Simbolo, Dimensione)

Attributi:

idElemento → auto_increment
Nome → varchar
Simbolo → varchar

Dimensione → varchar

Note:

1. Qui ci sono: gli elementi presenti nei terreni; gli elementi presenti nelle concimazioni; i prodotti delle malattie nelle piante
2. Dimensione → “Micro”, “Macro”, “Meso”

- **ESIGENZEELEMENTO**(Pianta, Elemento, Concentrazione)

References:

Elemento → idElemento (Elemento)

Pianta → idPianta (Pianta)

Attributi:

Pianta → int

Elemento → int

Concentrazione → int

Note:

1. Concentrazione → La concentrazione di un elemento è espressa in mg/m³

- **TIPOCONCIMAZIONE**(idConcimazione, InizioPeriodo, FinePeriodo, Periodicita)

Attributi:

idConcimazione → auto_increment

InizioPeriodo → int

FinePeriodo → int

Periodicita → int

Note:

1. InizioPeriodo < FinePeriodo
2. Se un periodo è per esempio ott-feb diventa ott-dic + gen-feb
3. Periodicità → La periodicità è espressa in settimane fra un singolo intervento e l'altro

- **UTILIZZOELEMENTO**(TipoConcimazione, Elemento, Modalita)

References:

TipoConcimazione → idConcimazione (Concimazione)

Elemento → idElemento (Elemento)

Attributi:

TipoConcimazione → int

Elemento → int

Modalita → varchar

Note:

1. Modalita → “Disciolto”, “Nebulizzato”
2. Elemento → deve essere in almeno un’occorrenza di Elemento in “Quantita”

- **CONCIMAZIONE**(TipoConcimazione, NumeroConcimazione)

References:

TipoConcimazione → idConcimazione (TipoConcimazione)

Attributi:

TipoConcimazione → int

NumeroConcimazione → int

- **QUANTITAELEMENTOPERCONCIMAZIONE**(Concimazione, NumeroConcimazione, Elemento, Quantita)

References:

Concimazione → TipoConcimazione (Concimazione)

NumeroConcimazione → NumeroConcimazione (Concimazione)

Elemento → idElemento (Elemento)

Attributi:

Concimazione → int

NumeroConcimazione → int

Elemento → int

Quantita → int

Note:

1. Quantita → La quantità è espressa in grammi
2. Elemento → Ci deve essere un'occorrenza di Elemento in UtilizzoElemento

- **ESEMPLARE**(idEsemplare, Pianta, Lotto, Prezzo, Dimensione, ManutenzioneProgrammata, Malato, Venduto)

References:

Pianta → idPianta (pianta)

Lotto → idLotto (Lotto)

Attributi:

idEsemplare → auto_increment

Pianta → int

Lotto → int

Prezzo → decimal

Dimensione → decimal

ManutenzioneProgrammata → bool

Malato → bool

Venduto → int

Note:

1. Dimensione < DimMax(Pianta)
2. Venduto = true implica Malato=false
3. Venduto=false implica ManutenzioneProgrammata=NULL

- **LOTTO**(idLotto, Fornitore, Costo)

Attributi:

idLotto → auto_increment

Fornitore → varchar

Costo → int

- **SCHEDA**(idScheda, Account, Esemplare, DataAcquisto, Collocazione, DimVaso, DimAcquisto, ManutenzioneAutomatica, NomePianta)

References:

Account → idAccount (Account)
 NomePianta → Nome (Pianta) (ridondanza)
 Esemplare → idEsemplare (Esemplare)

Attributi:

idScheda → auto_increment
 Account → int
 Esemplare → int
 DataAcquisto → date
 Collocazione → varchar
 DimVaso → decimal
 DimAcquisto → decimal
 ManutenzioneAutomatica → bool
 NomePianta → varchar

Note:

1. La scheda viene creata quando l'esemplare viene venduto
2. Collocazione → "PienaTerra", "Vaso"

- **ACCOUNT**(idAccount, Nickname, Via, NumCivico, Citta, Password, Credibilita, Cognome, Nome, Email, DomandaSicurezza, RispostaSicurezza)

Attributi:

idAccount → auto_increment
 Nickname → varchar
 Via → varchar
 NumCivico → int
 Citta → varchar
 Password → varchar
 Credibilita → int
 Cognome → varchar
 Nome → varchar
 Email → varchar

DomandaSicurezza → varchar
RispostaSicurezza → varchar

- **PIANTEPREFERITE**(Account, Pianta)

References:

Account → idAccount (Account)
Pianta → idPianta (Pianta)

Attributi:

Account → int
Pianta → int

- **ORDINE**(idOrdine, Account, Stato, TimeStamp)

References:

Account → idAccount (Account)

Attributi:

idOrdine → auto_increment
Account → int
Stato → varchar
TimeStamp → TimeStamp

Note:

1. Stato → "Pendente", "In processazione", "In preparazione", "Spedito", "Evaso"

- **RELATIVO**(Ordine, Esemplare)⁴

References:

Ordine → idOrdine (Ordine)
Esemplare → idEsemplare (Esemplare)

⁴ Questa relazione sarebbe potuta essere accorpata ad "Esemplare" creando un attributo "Ordine", ma, poiché abbiamo pensato potesse risultare utile poter analizzare velocemente gli ordini e per non appesantire eccessivamente la tabella con numerosi valori NULL, abbiamo deciso di lasciarla separata.

Attributi:

Ordine → int

Esemplare → int

Note:

1. Questa occorrenza viene creata dopo che viene creato un nuovo ordine o arriva la fornitura di un esemplare per cui c'era pendenza

- PENDENTE(Ordine, Pianta, Quantita)

References:

Ordine → idOrdine (Ordine)

Pianta → idPianta (Pianta)

Attributi:

Ordine → int

Pianta → int

Quantita → int

Note:

1. Un'istanza di pendente viene cancellata dopo che il rispettivo ordine è stato evaso
2. Quantita → Il valore di "Quantita" viene decrementato quando le piante in pendenza disponibili (relative ad un determinato ordine) vengono spedite.

- POST(Account, Timestamp, Testo, Thread)

References:

Account → idAccount (Account)

Attributi:

Account → int

TimeStamp → TimeStamp

Testo → text

Thread → text

- RISPOSTA(Account, Timestamp, AccountPost, TimestampPost, Testo)

References:

Account → idAccount (Account)
AccountPost → Account (Post)
TimestampPost → Timestamp (Post)

Attributi:

Account → int
TimeStamp → TimeStamp
AccountPost → int
TimeStampPost → TimeStamp
Testo → text

- VOTO (AccountVotante, AccountRisposta, TimeStampRisposta, Giudizio)

References:

AccountVotante → idAccount (Account)
AccountRisposta → Account (Risposta)
TimeStampRisposta → TimeStamp (Risposta)

Attributi:

AccountVotante → int
AccountRisposta → int
TimeStampRisposta → TimeStamp
Giudizio → int

Note:

1. Giudizio → 1-5

- URL(idIndirizzo, Indirizzo)

Attributi:

idIndirizzo → int
Indirizzo → varchar

- URLRISPOSTA(Url, AccountRisposta, TimestampRisposta)

References:

TimestampRisposta → Timestamp (Risposta)

AccountRisposta → Account (Risposta)

Url → idIndirizzo (Url)

Attributi:

Url → int

AccountRisposta → int

TimeStampRisposta → TimeStamp

- URLPOST(Url, AccountPost, TimestampPost)

References:

TimestampPost → Timestamp (Post)

AccountPost → Account (Post)

Url → idIndirizzo (Url)

Attributi:

Url → int

AccountPost → int

TimeStampPost → TimeStamp

- SEDE(idSede, Nome, Indirizzo, NumDipendenti)

Attributi:

idSede → int

Nome → varchar

Indirizzo → varchar

NumDipendenti → int

- SERRA(idSerra, Sede, Nome, Indirizzo, Lunghezza, Larghezza, Altezza, NumPiante, MaxPiante)

References:

Sede → idSede (Sede)

Attributi:

idSerra → int
Sede → int
Nome → varchar
Indirizzo → varchar
Lunghezza → int
Larghezza → int
Altezza → int
NumPiante → int
MaxPiante → int

- SEZIONE(idSezione, Serra, Nome, NumPiante, MaxPiante)

References:

Serra → idSerra (Serra)

Attributi:

idSezione → auto_increment
Serra → int
Nome → varchar
NumPiante → int
MaxPiante → int

Note:

1. NumPiante ≤ MaxPiante

- ISOLAMENTO(Sezione, Esemplare)

References:

Sezione → idSezione (Sezione)
Esemplare → idEsemplare (Esemplare)

Attributi:

Sezione → int
Esemplare → int

- RIPIANO(idRipiano, Sezione, MaxPiante)

References:

Sezione → idSezione (Sezione)

Attributi:

idRipiano → auto_increment

Sezione → int

MaxPiante → int

- CONTENITORE(idContenitore, Ripiano, Esemplare, Terreno, Irrigazione, Superficie)

References:

Ripiano → idRipiano (Ripiano)

Esemplare → idEsemplare (Esemplare)

Terreno → idTerreno (Terreno)

Attributi:

idContenitore → auto_increment

Ripiano → int

Esemplare → int

Terreno → int

Irrigazione → varchar

Superficie → int

Note:

1. Irrigazione → “Bassa”, “Media”, “Alta”
2. Superficie → Il valore in “Superficie” corrisponde al diametro dei contenitori

- MISURAZIONECONTENITORE(Contenitore, TimeStamp, pH, Idratazione)

References:

Contenitore → idContenitore (Contenitore)

Attributi:

Contenitore → int
TimeStamp → TimeStamp
pH → varchar
Idratazione → varchar

Note:

1. pH → “Fortemente acido” (<5,5), “Acido” (5,5 - 6,0), “SubAcido” (6,0 - 6,8), “Neutro” (6,8 - 7,3), “SubBasico” (7,3 - 8,0), “Basico” (8,0 - 8,5), “Alcalino” (>8,5)
2. Idratazione → “Bassa”, “Media”, “Alta”

- PRESENZAELEMENTO(Contenitore, TimestampMisurazione, Elemento, Quantita)

References:

Contenitore → idContenitore (Contenitore)
TimestampMisurazione → Timestamp (MisurazioneContenitore)
Elemento → idElemento (Elemento)

Attributi:

Contenitore → int
TimestampMisurazione → TimeStamp
Elemento → int
Quantita → int

Note:

1. Quantita → La Quantità è espressa in mg/m³

- COMPOSIZIONECONTENITORE(Contenitore, Componente, Percentuale)

References:

Contenitore → idContenitore (Contenitore)
Componente → idComponente (Componente)

Attributi:

Contenitore → int
Componente → int

Percentuale → int

- MISURAZIONE AMBIENTALE(Sezione, TimeStamp, Umidita, Temperatura, Illuminazione)

References:

Sezione → idSezione (Sezione)

Attributi:

Sezione → int

TimeStamp → TimeStamp

Umidita → int

Temperatura → int

Illuminazione → varchar

Note:

1. Umidita → L' umidità è espressa come percentuale
2. Temperatura → La temperatura è espressa in gradi celsius
3. Illuminazione → "Bassa", "Media", "Alta"

- VASO(idVaso, Pianta, PosizionePianta, Settore, Dimensione, Materiale)

References:

Pianta → idPianta (Pianta)

Settore → idSettore (Settore)

Attributi:

idVaso → auto_increment

Pianta → int

PosizionePianta → int

Settore → int

Dimensione → int

Materiale → varchar

Note:

1. Dimensione → Il valore in "Dimensione" corrisponde al diametro del vaso

- GIARDINO(Account, Numero, Clima, IndiceManut)

References:

Account → idAccount (Account)

Attributi:

Account → int

Numero → auto_increment

Clima → varchar

IndiceManut → varchar

Note:

1. Numero → “Numero” è un valore incrementale che corrisponde al numero del giardino creato da un utente (se è il terzo giardino creato da un determinato utente, quel giardino avrà numero 3, e così via); per ogni utente “Numero” parte da 1
2. Clima → “MoltoCaldo” (>14), “Caldo” (12-14), “Medio” (10-12), “Freddo” (7-10) “MoltoFreddo” (<7) (i valori sono espressi in gradi Celsius)
3. IndiceManut → “Basso” (< 3), “Medio” (3-5), “Alto” (> 5)

- PUNTO(X, Y, Settore, OrdPosizione)

References:

Settore → idSettore (Settore)

Attributi:

X → int

Y → int

Settore → int

OrdPosizione → int

Note:

1. Devono esistere almeno 3 punti per ogni settore

- **SETTORE**(idSettore, Account, NumeroGiardino, DirCardinale, Base, Terreno, LuceIniziale, LuceAttuale, Area)

References:

Account → Account (Giardino)
 NumeroGiardino → Numero (Giardino)
 Terreno → idTerreno (Terreno)
 LuceIniziale → idLuce (Luce)
 LuceAttuale → idLuce (Luce)

Attributi:

idSettore → auto_increment
 Account → int
 NumeroGiardino → int
 DirCardinale → varchar
 Base → varchar
 Terreno → int
 LuceIniziale → int
 LuceAttuale → int
 Area → int

Note:

1. Base → "PienaTerra", "Pavimento"
2. DirCardinale → "Nord", "Sud", "Est", "Ovest"

- **COMPONENTE**(idComponente, Nome, Consistenza, Permeabilita)

Attributi:

idComponente → auto_increment
 Nome → varchar
 Consistenza → varchar
 Permeabilita → varchar

Note:

1. Consistenza → "Liquida", "Plastica", "SemiSolida", "solida"
2. Permeabilita → "Alto" ($>10^{-3}$), "Medio" ($10^{-3} - 10^{-5}$), "Basso" ($10^{-5} - 10^{-7}$), "Molto basso" ($10^{-7} - 10^{-9}$), "Impermeabile" ($<10^{-9}$)

- **COMPOSIZIONESETTORE**(Settore, Componente, Percentuale)

References:

Settore → idSettore (Settore)

Componente → idComponente (Componente)

Attributi:

Settore → int

Componente → int

Percentuale → int

Note:

1. Percentuale → Indica in percentuale la quantità di un determinato elemento in un determinato settore

- **FORMAPIANTA**(Pianta, X, Y, Settore, Dim, PosPianta)

References:

Pianta → idPianta (Pianta)

X → X (Punto)

Y → Y (Punto)

Settore → Settore (Punto)

Attributi:

Pianta → int

X → int

Y → int

Settore → int

Dim → int

PosPianta → int

- **INTERVENTO**(idIntervento, Esemplare, Effettuato, Data, Costo, Entita, Motivo, Tipo)

References:

Esemplare → idEsemplare

Attributi:

idIntervento → auto_increment
Esemplare → int
Effettuato → bool
Data → date
Costo → decimal
Entita → int
Motivo → varchar
Tipo → varchar

Note:

1. Data → Se “Effettuato” = FALSE, “Data” rappresenta la data in cui l’intervento verrà eseguito, in caso contrario, rappresenterà la data in cui l’intervento è stato effettuato
2. Motivo → “Richiesta”, “Programmato”
3. Tipo → “Trattamento”, “Potatura”, “Concimazione”, “Piantumazione”, “Rinvaso”

● AGENTE(idAgente, Nome, Chimico, Biologico)

Attributi:

idAgente → auto_increment
Nome → varchar
Chimico → bool
Biologico → bool

Note:

1. Chimico/Biologico → Rappresentano il tipo di trattamento a cui un agente è vulnerabile; un agente può essere vulnerabile ad entrambi i tipi di trattamento, o solo ad un tipo, o a nessuno dei due

● TRATTAMENTO(Intervento, Agente, Prevenzione)

References:

Agente → idAgente (Agente)
Intervento → idIntervento (Intervento)

Attributi:

Intervento → int
Agente → int
Prevenzione → bool

- **PRODOTTO**(idProdotto, Nome, Modalita, Attesa, Tipo)

Attributi:

idProdotto → auto_increment
Nome → varchar
Modalita → varchar
Attesa → int
Tipo → varchar

Note:

1. Modalita → “Irrigazione”, “Nebulizzazione”, “Entrambe”
2. Attesa → L’ attesa è il periodo di tempo che bisogna far passare prima di poter mangiare i frutti della pianta, in caso ne abbia; è espresso in settimane
3. Tipo → “Selettivo”, “AmpioSpettro”

- **CONTRASTO**(Prodotto, Agente, DosaggioConsigliato)

References:

Prodotto → idProdotto (Prodotto)
Agente → idAgente (Agente)

Attributi:

Prodotto → int
Agente → int
DosaggioConsigliato → int

Note:

1. DosaggioConsigliato → Il dosaggio consigliato è espresso in grammi

- **PERIODONONUTILIZZO**(Prodotto, InizioPeriodo, FinePeriodo)

References:

Prodotto → idProdotto (Prodotto)

Attributi:

Prodotto → int

InizioPeriodo → int

FinePeriodo → int

Note:

1. InizioPeriodo ≤ Fine
2. Se un periodo è per esempio ott-feb diventa ott-dic + gen-feb

- **PRINCIPIO**(idPrincipio, Nome)

Attributi:

idPrincipio → int

Nome → varchar

- **BASATO**(Prodotto, Principio, Concentrazione)

References:

Prodotto → idProdotto (Prodotto)

Principio → idPrincipio (Principio)

Attributi:

Prodotto → int

Principio → int

Concentrazione → int

Note:

1. Concentrazione → La concentrazione di un principio in un prodotto è espressa in percentuale

- **UTILIZZOPRODOTTO**(Prodotto, Intervento, Dosaggio)

References:

Prodotto → idProdotto (Prodotto)

Intervento → idIntervento (Intervento)

Attributi:

Prodotto → int

Intervento → int

Dosaggio → int

Note:

1. Dosaggio → Il dosaggio è espresso in grammi

- **SINTOMO**(idSintomo, Descrizione)

Attributi:

idSintomo → int

Descrizione → varchar

- **IMMAGINE**(idImmagine, Sintomo, NomeFile)

References:

Sintomo → idSintomo (Sintomo)

Attributi:

idImmagine → auto_increment

Sintomo → int

NomeFile → varchar

- **TIPOINTERVENTO****POTATURA**(Intervento, TipoPotatura)

References:

Intervento → idIntervento (Intervento)

TipoPotatura → idTipoPotatura (TipoPotatura)

Attributi:

Intervento → int

TipoPotatura → int

Note:

1. Intervento → Un intervento può essere inserito in questa entità solo se il suo campo "Tipo"="Potatura"

- **TIPOINTERVENTOCONCIMAZIONE**(Intervento, TipoConcimazione, NumeroConcimazione)

References:

Intervento → idIntervento (Intervento)

TipoConcimazione → TipoConcimazione (Concimazione)

NumeroConcimazione → NumeroConcimazione (Concimazione)

Attributi:

Intervento → int

TipoConcimazione → int

NumeroConcimazione → int

Note:

1. Intervento → Un intervento può essere inserito in questa entità solo se il suo campo "Tipo"="Concimazione"

- **PREFERENZE**(idPreferenze, Account, Dimensione, ImpDimensione, Dioica, ImpDioica, Infestante, ImpInfestante, Temp, ImpTemp, Sempreverde, ImpSempreverde, Costo, ImpCosto, Luce, ImpLuce, Acqua, ImpAcqua, Terreno, ImpTerreno, IndiceManut, ImpIndiceManut)

References:

Account → idAccount (Account)

Terreno → idTerreno (Terreno)

Attributi:

idPreferenze → auto_increment
Account → int
Dimensione → varchar
ImpDimensione → int
Dioica → bool
ImpDioica → int
Infestante → bool
ImpInfestante → int
Temp → varchar
ImpTemp → int
Sempreverde → bool
ImpSempreverde → int
Costo → varchar
ImpCosto → int
Luce → varchar
ImpLuce → int
Acqua → varchar
ImpAcqua → int
Terreno → int
ImpTerreno → int
IndiceManut → varchar
ImpIndiceManut → int

Note:

1. Nei vari attributi verranno inseriti valori generici, non codici specifici, quindi i references non sono necessari
2. Dimensione → "MoltoPiccola" (< 10 cm), "Piccola" (10-30 cm), "Media" (30-70 cm), "Grande" (70-120 cm), "MoltoGrande" (> 120 cm) (diametro)
3. Imp- → Le importanze sono valori compresi tra 1 e 10
4. Temperatura → "MoltoCaldo" (>14), "Caldo" (12-14), "Medio" (10-12), "Freddo" (7-10) "MoltoFreddo" (<7) (i valori sono espressi in gradi Celsius)
5. Luce, Acqua → "Basso", "Medio", "Alto"
6. IndiceManut → "Basso" (< 3), "Medio" (3-5), "Alto" (> 5)

7. Costo → “Basso” (< 15), “MedioBasso” (15-30), “Medio” (30-60),
“MedioAlto” (60-100), “Alto” (> 100)

- **PREFERENZE**PERIODI(Preferenze, Periodo, Importanza)

References:

Preferenze → idPreferenza (Preferenze)

Periodo → idPeriodo (PeriodoCicli)

Attributi:

Preferenze → int

Periodo → int

Importanza → int

Note:

1. Importanza → L'importanza è un valore compreso tra 1 e 10

- **REPORT**DIAGNOSTICA(Data, Esemplare, TimeStampMisurazione, Sezione)

References:

Esemplare → idEsemplare (Esemplare)

Sezione → idSezione (MisurazioneAmbientale)

TimeStamp → TimeStamp (MisurazioneAmbientale)

Attributi:

Data → date

Esemplare → int

TimeStampMisurazione → TimeStamp

Sezione → int

- **DECISIONE**(DataReport, Esemplare, Intervento, Esperto)

References:

DataReport → Data (ReportDiagnostica)

Esemplare → Esemplare (ReportDiagnostica)

Intervento → idIntervento (Intervento)

Attributi:

DataReport → date

Esemplare → int

Intervento → int

Esperto → varchar

- **DIAGNOSI POSSIBILI**(DataReport, Esemplare, Agente, Attinenza)

References:

DataReport → Data (ReportDiagnostica)

Esemplare → Esemplare (ReportDiagnostica)

Agente → idAgente (Agente)

Attributi:

DataReport → date

Esemplare → int

Agente → int

Attinenza → int

Note:

1. Attinenza → Ordina le patologie a partire dalla più probabile alla meno probabile

- **DIAGNOSI ESPERTO**(DataReport, Esemplare, Agente, Numero, Esperto)

References:

DataReport → Data (ReportDiagnostica)

Esemplare → Esemplare (ReportDiagnostica)

Agente → idAgente (Agente)

Attributi:

DataReport → date

Esemplare → int

Agente → int

Esperto → varchar

Numero → int

Note:

1. Numero → Rappresenta il numero della diagnosi effettuata sull'esemplare

- **INFO**SINTOMI(DataReport, Esemplare, Sintomo)

References:

DataReport → Data (ReportDiagnostica)
 Esemplare → Esemplare (ReportDiagnostica)
 Sintomo → idSintomo (Sintomo)

Attributi:

DataReport → date
 Esemplare → int
 Sintomo → int

- **CONDIZIONI**FAVOREVOLI(idCondizioniFavorevoli, Agente, Idratazione, pH, Umidita, Temperatura, Illuminazione, Mese)

References:

Agente → idAgente (Agente)

Attributi:

idCondizioniFavorevoli → auto_increment
 Agente → int
 Idratazione → varchar
 pH → varchar
 Umidita → varchar
 Temperatura → varchar
 Illuminazione → varchar
 Mese → int

Note:

1. Se un periodo è per esempio ott-feb diventa ott-dic + gen-feb
2. pH → “Fortemente acido” (< 5,5), “Acido” (5,5 - 6,0), “SubAcido” (6,0 - 6,8), “Neutro” (6,8 - 7,3), “SubBasico” (7,3 - 8,0), “Basico” (8,0 - 8,5), “Alcalino” (> 8,5)
3. Temperatura → “MoltoCaldo” (> 14), “Caldo” (12-14), “Medio” (10-12), “Freddo” (7-10), “MoltoFreddo” (< 7)
4. Umidita → L’umidità è espressa in percentuale
5. Illuminazione → “Bassa” , “Media” , “Alta”

6. Idratazione → “Bassa”, “Media”, “Alta”

- ELEMENTICOINVOLTI(Elemento, CondizioniFavorevoli,)

References:

Elemento → idElemento (Elemento)

CondizioniFavorevoli → idCondizioniFavorevoli (CondizioniFavorevoli)

Attributi:

Elemento → int

CondizioniFavorevoli → int

7. Analisi delle dipendenze funzionali

7.1 Introduzione

Dopo aver completato la traduzione nel modello relazionale, siamo passati alla fase del controllo sulle dipendenze funzionali.

Nel prossimo paragrafo, la lista completa divisa per tabelle.

7.2 Lista delle dipendenze

- Tabella PIANTA

PIANTA(idPianta, Nome, Cultivar, Dioica, DimMax, IndiceManut, CostoBase, Infestante, Sempreverde, Famiglia)

- idPianta → Nome, Cultivar, Dioica, DimMax, IndiceManut, CostoBase, Infestante, Sempreverde, Famiglia

Poiché “idPianta” è superchiave, la tabella è BCNF

- Tabella FAMIGLIA

FAMIGLIA(idFamiglia, Nome, CrescitaAerea, CrescitaRadicale)

- idFamiglia → Nome, CrescitaAerea, CrescitaRadicale

Poiché “idFamiglia” è superchiave, la tabella è BCNF

- Tabella REPORTCONSIGLIACQUISTO

REPORTCONSIGLIACQUISTO(Pianta, Data, Note, Quantita)

- Pianta, Data → Note, Quantita

Poiché “Pianta”, “Data” è superchiave, la tabella è BCNF

- Tabella PERIODOPOTATURA

PERIODOPOTATURA(Pianta, InizioPeriodo, FinePeriodo, TipoPotatura)

Poiché tutti gli attributi fanno parte della chiave, la tabella è BCNF

- Tabella TIPOPOTATURA

TIPOPOTATURA(idTipoPotatura, Nome)

- idTipoPotatura → Nome

Poiché “idPotatura” è superchiave, la tabella è BCNF

- Tabella NECESSITAPOTATURA

NECESSITAPOTATURA(Pianta, TipoPotatura, Quantita)

- Pianta, TipoPotatura → Quantita

Poiché “Pianta”, “TipoPotatura” è superchiave, la tabella è BCNF

- Tabella PERIODOATTACCHI

PERIODOATTACCHI(Pianta, InizioPeriodo, FinePeriodo, Agente, Entita, Probabilita)

- Pianta, InizioPeriodo, FinePeriodo, Agente → Entita

Poiché “Pianta”, “InizioPeriodo”, “FinePeriodo”, “Agente” è superchiave, la tabella è BCNF

- Tabella PERIODORINVASI

PERIODORINVASI(Pianta, InizioPeriodo, FinePeriodo)

Poiché tutti gli attributi fanno parte della chiave, la tabella è BCNF

- Tabella CICLIPIANTA

CICLIPIANTA(Pianta, Periodo)

Poiché tutti gli attributi fanno parte della chiave, la tabella è BCNF

- Tabella PERIODOCICLI

PERIODOCICLI(idPeriodo, InizioPeriodo, FinePeriodo, Fio_Fru, Vegetativo)

- idPeriodo → InizioPeriodo, FinePeriodo, Fio_Fru, Vegetativo

Poiché “idPeriodo” è superchiave, la tabella è BCNF

- Tabella ESIGENZE

ESIGENZE(Pianta, Terreno, Temperatura, AcquaVegetativo, AcquaRiposo, LuceVegetativo, LuceRiposo)

- Pianta → Terreno, Temperatura, AcquaVegetativo, AcquaRiposo, LuceVegetativo

Poiché “Pianta” è superchiave, la tabella è BCNF

- Tabella ACQUA

ACQUA(idAcqua, Quantita, Periodicita)

- idAcqua → Quantita, Periodicita

Poiché “idAcqua” è superchiave, la tabella è BCNF

- Tabella TEMPERATURA

TEMPERATURA(idTemp, TempMin, TempMax)

- idTemp → TempMin, TempMax

Poiché “idTemp” è superchiave, la tabella è BCNF

- Tabella LUCE

LUCE(idLuce, Diretta, NumOre, Quantita)

- idLuce → Diretta, NumOre, Quantita

Poiché “idLuce ” è superchiave, la tabella è BCNF

- Tabella ESIGENZECONCIMAZIONE

ESIGENZECONCIMAZIONE(Pianta, TipoConcimazione)

Poiché tutti gli attributi fanno parte della chiave, la tabella è BCNF

- Tabella TERRENO

TERRENO(idTerreno, Consistenza, Permeabilita, pH)

- idTerreno → Consistenza, Permeabilita, pH

Poiché “idTerreno ” è superchiave, la tabella è BCNF

- Tabella COMPOSIZIONETERRENO

COMPOSIZIONE TERRENO(Terreno, Elemento, Concentrazione)

- Terreno, Elemento → Concentrazione

Poiché “Terreno” e “Elemento” sono superchiave, la tabella è BCNF

- Tabella ELEMENTO

ELEMENTO(idElemento, Nome, Simbolo, Dimensione)

- idElemento → Nome, Simbolo, Dimensione → varchar

Poiché “idElemento ” è superchiave, la tabella è BCNF

- Tabella TIPOCONCIMAZIONE

TIPOCONCIMAZIONE(idConcimazione, InizioPeriodo, FinePeriodo, Periodicità)

- idConcimazione → InizioPeriodo, FinePeriodo, Periodicità

Poiché “idConcimazione ” è superchiave, la tabella è BCNF

- Tabella UTILIZZOELEMENTO

UTILIZZOELEMENTO(TipoConcimazione, Elemento, Modalità)

- TipoConcimazione, Elemento → Modalità

Poiché “TipoConcimazione” e “Elemento” sono superchiave, la tabella è BCNF

- Tabella CONCIMAZIONE

CONCIMAZIONE(TipoConcimazione, NumeroConcimazione)

Poiché tutti gli attributi fanno parte della chiave, la tabella è BCNF

- Tabella QUANTITAELEMENTOPERCONCIMAZIONE
QUANTITAELEMENTOPERCONCIMAZIONE(Concimazione, NumeroConcimazione, Elemento, Quantita)

- Concimazione , NumeroConcimazione, Elemento → Quantita

Poiché “Concimazione ” , “NumeroConcimazione” e “Elemento” sono superchiave, la tabella è BCNF

- Tabella ESEMPLARE
ESEMPLARE(idEsemplare, Pianta, Lotto, Prezzo, Dimensione, ManutenzioneProgrammata, Malato, Venduto)

- idEsemplare → Pianta, Lotto, Prezzo, Dimensione, ManutenzioneProgrammata, Malato, Venduto

Poiché “idEsemplare ” è superchiave, la tabella è BCNF

- Tabella LOTTO
LOTTO(idLotto, Fornitore, Costo)

- idLotto → Fornitore, Costo

Poiché “idEsemplare ” è superchiave, la tabella è BCNF

- Tabella SCHEDA
SCHEDA(idScheda, Account, Esemplare, DataAcquisto, Collocazione, DimVaso, DimAcquisto, ManutenzioneAutomatica, NomePianta)

- idScheda → Account, Esemplare, DataAcquisto, Collocazione, DimVaso, DimAcquisto, ManutenzioneAutomatica, NomePianta
- Esemplare → idScheda, Account, DataAcquisto, Collocazione, DimVaso, DimAcquisto, ManutenzioneAutomatica, NomePianta

Poiché sia “idScheda” sia “Esemplare” sono superchiavi, la tabella è BCNF

- Tabella ACCOUNT

ACCOUNT(idAccount, Nickname, Via, NumCivico, Citta, Password, Credibilita, Cognome, Nome, Email, DomandaSicurezza, RispostaSicurezza)

- idAccount → Nickname, Via, NumCivico, Citta, Password, Credibilita, Cognome, Nome, Email, DomandaSicurezza, RispostaSicurezza
- Nickname → idAccount, Via, NumCivico, Citta, Password, Credibilita, Cognome, Nome, Email, DomandaSicurezza, RispostaSicurezza
- Email → Nickname, idAccount, Via, NumCivico, Citta, Password, Credibilita, Cognome, Nome, DomandaSicurezza, RispostaSicurezza

Poiché sia “idAccount ” sia “Nickname” sia “Email” sono superchiavi, la tabella è BCNF

- Tabella PIANTEPREFERITE

PIANTEPREFERITE(Account, Pianta)

Poiché tutti gli attributi fanno parte della chiave, la tabella è BCNF

- Tabella ORDINE

ORDINE(idOrdine, Account, Stato, TimeStamp)

- idOrdine → Account, Stato, TimeStamp

Poiché “idOrdine ” è superchiave, la tabella è BCNF

- Tabella RELATIVO

RELATIVO(Ordine, Esemplare)

Poiché tutti gli attributi fanno parte della chiave, la tabella è BCNF

- Tabella PENDENTE

PENDENTE(Ordine, Pianta, Quantita)

- Ordine, Pianta → Quantita

Poiché “Ordine ” e “Pianta ” è superchiave, la tabella è BCNF

- Tabella POST

POST(Account, Timestamp, Testo, Thread)

- Account, TimeStamp → Testo, Thread

Poiché “Account ” e “TimeStamp ” è superchiave, la tabella è BCNF

- Tabella RISPOSTA

RISPOSTA(Account, Timestamp, AccountPost, TimestampPost, Testo)

- Account, TimeStamp → AccountPost, TimestampPost, Testo

Poiché “Account ” e “TimeStamp ” è superchiave, la tabella è BCNF

- Tabella VOTO

VOTO (AccountVotante, AccountRisposta, TimeStampRisposta, Giudizio)

- AccountVotante, AccountRisposta, TimeStampRisposta → Giudizio

Poiché “AccountVotante”, “AccountRisposta” e “TimeStampRisposta” è superchiave, la tabella è BCNF

- Tabella URL

URL(idIndirizzo, Indirizzo)

- idIndirizzo → Indirizzo

Poiché “idIndirizzo” è superchiave, la tabella è BCNF

- Tabella URLRISPOSTA

URLRISPOSTA(Url, AccountRisposta, TimestampRisposta)

Poiché tutti gli attributi fanno parte della chiave, la tabella è BCNF

- Tabella URLPOST

URLPOST(Url, AccountPost, TimestampPost)

Poiché tutti gli attributi fanno parte della chiave, la tabella è BCNF

- Tabella SEDE

SEDE(idSede, Nome, Indirizzo, NumDipendenti)

- idSede → Nome, Indirizzo, NumDipendenti

Poiché "idSede" è superchiave, la tabella è BCNF

- Tabella SERRA

SERRA(idSerra, Sede, Nome, Indirizzo, Lunghezza, Larghezza, Altezza, NumPiante, MaxPiante)

- idSerra → Sede, Nome, Indirizzo, Lunghezza, Larghezza, Altezza, NumPiante, MaxPiante

Poiché "idSerra" è superchiave, la tabella è BCNF

- Tabella SEZIONE

SEZIONE(idSezione, Serra, Nome, NumPiante, MaxPiante)

- idSezione → Serra, Nome, NumPiante, MaxPiante

Poiché "idSezione" è superchiave, la tabella è BCNF

- Tabella ISOLAMENTO

ISOLAMENTO(Sezione, Esemplare)

Poiché tutti gli attributi fanno parte della chiave, la tabella è BCNF

- Tabella RIPIANO

RIPIANO(idRipiano, Sezione, MaxPiante)

- idRipiano → Sezione, MaxPianta

Poiché "idRipiano" è superchiave, la tabella è BCNF

- Tabella CONTENITORE

CONTENITORE(idContenitore, Ripiano, Esemplare, Terreno, Irrigazione, Superficie)

- idContenitore → Ripiano, Esemplare, Terreno, Irrigazione, Superficie

Poiché "idContenitore" è superchiave, la tabella è BCNF.

- Tabella MISURAZIONECONTENITORE

MISURAZIONECONTENITORE(Contenitore, TimeStamp, pH, Idratazione)

- Contenitore, TimeStamp → pH, Idratazione

Poiché “Contenitore”, “TimeStamp” è superchiave, la tabella è BCNF.

- Tabella PRESENZAELEMENTO

PRESENZAELEMENTO(Contenitore, TimestampMisurazione, Elemento, Quantita)

- Contenitore, TimeStampMisurazione, Elemento → Quantita

Poiché “Contenitore”, “TimeStampMisurazione”, “Elemento” è superchiave, la tabella è BCNF.

- Tabella COMPOSIZIONECONTENITORE

COMPOSIZIONECONTENITORE(Contenitore, Componente, Percentuale)

- Contenitore, Componente → Percentuale

Poiché “Contenitore”, “Componente” è superchiave, la tabella è BCNF.

- Tabella MISURAZIONEAMBIENTALE

MISURAZIONEAMBIENTALE(Sezione, TimeStamp, Umidita, Temperatura, Illuminazione)

- Sezione, TimeStamp, Umidita → Temperatura, Illuminazione

Poiché “Sezione”, “TimeStamp”, “Umidita” è superchiave, la tabella è BCNF

- Tabella VOTO

VASO(idVaso, Pianta, PosizionePianta, Settore, Dimensione, Materiale)

- idVaso → Pianta, PosizionePianta, Settore, Dimensione, Materiale

Poiché “idVaso” è superchiave, la tabella è BCNF

- Tabella GIARDINO

GIARDINO(Account, Numero, Clima, IndiceManut)

- Account, Numero → Clima, IndiceManut

Poiché “Account”, “Numero” è superchiave, la tabella è BCNF

- Tabella PUNTO

PUNTO(X, Y, Settore, OrdPosizione)

- X, Y, Settore → OrdPosizione

Poiché “X”, “Y”, “Settore” è superchiave, la tabella è BCNF

- Tabella SETTORE

SETTORE(idSettore, Account, NumeroGiardino, DirCardinale, Base, Terreno, LuceIniziale, LuceAttuale, Area)

- idSettore → Account, NumeroGiardino, DirCardinale, Base, Terreno, LuceIniziale, LuceAttuale, Area

Poiché “idSettore” è superchiave, la tabella è BCNF

- Tabella COMPONENTE

COMPONENTE(idComponente, Nome, Consistenza, Permeabilita)

- idComponente → Nome, Consistenza, Permeabilita

Poiché “idComponente” è superchiave, la tabella è BCNF

- Tabella COMPOSIZIONESETTORE

COMPOSIZIONESETTORE(Settore, Componente, Percentuale)

- Settore, Componente → Percentuale

Poiché “Settore”, “Componente” è superchiave, la tabella è BCNF

- Tabella FORMAPIANTA

FORMAPIANTA(Pianta, X, Y, Settore, Dim, PosPianta)

- Pianta, X, Y, Settore → Dim, PosPianta

Poiché “Pianta”, “X”, “Y”, “Settore” è superchiave, la tabella è BCNF

- Tabella INTERVENTO

INTERVENTO(idIntervento, Esemplare, Effettuato, Data, Costo, Entita, Motivo, Tipo)

- idIntervento → Esemplare, Effettuato, Data, Costo, Entita, Motivo, Tipo

Poiché “idIntervento” è superchiave, la tabella è BCNF

- Tabella AGENTE

AGENTE(idAgente, Nome, Chimico, Biologico)

- idAgente → Nome, Chimico, Biologico

Poiché “idAgente” è superchiave, la tabella è BCNF

- Tabella TRATTAMENTO

TRATTAMENTO(Intervento, Agente, Prevenzione)

- Intervento, Agente → Prevenzione

Poiché “Intervento”, “Agente” è superchiave, la tabella è BCNF

- Tabella PRODOTTO

PRODOTTO(idProdotto, Nome, Modalita, Attesa, Tipo)

- idProdotto → Nome, Modalita, Attesa, Tipo

Poiché “idProdotto” è superchiave, la tabella è BCNF

- Tabella CONTRASTO

CONTRASTO(Prodotto, Agente, DosaggioConsigliato)

- Prodotto, Agente → DosaggioConsigliato

Poiché “Prodotto”, “Agente” è superchiave, la tabella è BCNF

- Tabella PERIODONONUTILIZZO

PERIODONONUTILIZZO(Prodotto, InizioPeriodo, FinePeriodo)

- Prodotto, InizioPeriodo → FinePeriodo

Poiché “Prodotto”, “InizioPeriodo” è superchiave, la tabella è BCNF

- Tabella PRINCIPIO

PRINCIPIO(idPrincipio, Nome)

- idPrincipio → Nome

Poiché “idPrincipio” è superchiave, la tabella è BCNF

- Tabella BASATO

BASATO(Prodotto, Principio, Concentrazione)

- Prodotto, Principio → Concentrazione

Poiché “Prodotto”, “InizioPeriodo” è superchiave, la tabella è BCNF

- Tabella UTILIZZOPRODOTTO

UTILIZZOPRODOTTO(Prodotto, Intervento, Dosaggio)

- Prodotto, Intervento → Dosaggio

Poiché “Prodotto”, “Intervento” è superchiave, la tabella è BCNF

- Tabella SINTOMO

SINTOMO(idSintomo, Descrizione)

- idSintomo → Descrizione

Poiché “Sintomo” è superchiave, la tabella è BCNF

- Tabella IMMAGINE

IMMAGINE(idImmagine, Sintomo, NomeFile)

- idImmagine → Sintomo, NomeFile

Poiché “idImmagine” è superchiave, la tabella è BCNF

- Tabella TIPOINTERVENTOPOTATURA

TIPOINTERVENTOPOTATURA(Intervento, TipoPotatura)

Poiché tutti gli attributi appartengono alla chiave, la tabella è BCNF

- Tabella TIPOINTERVENTOCONCIMAZIONE

TIPOINTERVENTOCONCIMAZIONE(Intervento, TipoConcimazione, NumeroConcimazione)

Poiché tutti gli attributi appartengono alla chiave, la tabella è BCNF

- Tabella PREFERENZE

PREFERENZE(idPreferenze, Account, Dimensione, ImpDimensione, Dioica, ImpDioica, Infestante, ImpInfestante, Temp, ImpTemp, Sempreverde, ImpSempreverde, Costo, ImpCosto, Luce, ImpLuce, Acqua, ImpAcqua, Terreno, ImpTerreno, IndiceManut, ImpIndiceManut)

- idPreferenze - Account, Dimensione, ImpDimensione, Dioica, ImpDioica, Infestante, ImpInfestante, Temp, ImpTemp, Sempreverde, ImpSempreverde, Costo, ImpCosto, Luce, ImpLuce, Acqua, ImpAcqua, Terreno, ImpTerreno, IndiceManut, ImpIndiceManut

Poiché “idPreferenze” è superchiave, la tabella è BCNF

- Tabella PREFERENZEPERIODI

PREFERENZEPERIODI(Preferenze, Periodo, Importanza)

Poiché tutti gli attributi appartengono alla chiave, la tabella è BCNF

- Tabella REPORTDIAGNOSTICA

REPORTDIAGNOSTICA(Data, Esemplare, TimeStampMisurazione, Sezione)

- Data, Esemplare - TimeStampMisurazione, Sezione

Poiché “Data”, “Esemplare” è superchiave, la tabella è BCNF

- Tabella DECISIONE

DECISIONE(DataReport, Esemplare, Intervento, Esperto)

- DataReport, Esemplare, Intervento - Esperto

Poiché “DataReport”, “Esemplare”, “Intervento” è superchiave, la tabella è BCNF

- Tabella DIAGNOSIPOSSIBILI

DIAGNOSIPOSSIBILI(DataReport, Esemplare, Agente, Attinenza)

- DataReport, Esemplare, Agente → Attinenza

Poiché “DataReport”, “Esemplare”, “Agente” è superchiave, la tabella è BCNF

- Tabella DIAGNOSIESPERTO

DIAGNOSIESPERTO(DataReport, Esemplare, Agente, Numero, Esperto)

- DataReport, Esemplare, Agente, Numero → Esperto

Poiché “DataReport”, “Esemplare”, “Agente”, “Numero” è superchiave, la tabella è BCNF

- Tabella INFOSINTOMI

INFOSINTOMI(DataReport, Esemplare, Sintomo)

Poiché tutti gli attributi appartengono alla chiave, la tabella è BCNF

- Tabella CONDIZIONIFAVOREVOLI

CONDIZIONIFAVOREVOLI(idCondizioniFavorevoli, Agente, Idratazione, pH, Umidità, Temperatura, Illuminazione, Mese)

- idCondizioniFavorevoli → Agente, Idratazione, pH, Umidità, Temperatura, Illuminazione, Mese

Poiché “idCondizioniFavorevoli” è superchiave, la tabella è BCNF

- Tabella ELEMENTICOINVOLTI

ELEMENTICOINVOLTI(Elemento, CondizioniFavorevoli,)

Poiché tutti gli attributi appartengono alla chiave, la tabella è BCNF

9. Implementazione su DBMS

Oracle MySQL

9.1 Introduzione

Dopo aver terminato la ristrutturazione del diagramma E-R abbiamo iniziato la fase di implementazione effettiva su Oracle MySQL.

Per prima cosa, utilizzando l'ambiente di sviluppo MySQL Workbench 5.7, abbiamo costruito lo script di creazione dello schema.

Fatto questo, abbiamo parallelamente popolato il database e costruito i vari trigger/procedure/event/function per rispondere alle richieste del progetto.

Di seguito, le fasi dello sviluppo spiegate in dettaglio.

9.2 Implementazione Schema

La prima operazione effettuata è stata la creazione di tutte le tabelle precedentemente progettate nello schema ER. Per farlo, abbiamo dato all'IDE le specifiche delle varie tabelle, dopodiché MySQL ha creato il codice base. Dopo aver fatto questo, abbiamo corretto alcuni particolari, come le azioni da effettuare ON UPDATE e ON DELETE.

Di seguito, un esempio del codice di creazione.

```
-- -----  
-- Table `progettouni`.`Pianta`  
-- -----  
DROP TABLE IF EXISTS `progettouni`.`Pianta` ;  
  
CREATE TABLE IF NOT EXISTS `progettouni`.`Pianta` (  
  `idPianta` INT(11) NOT NULL AUTO_INCREMENT,  
  `Nome` VARCHAR(45) NOT NULL,  
  `Cultivar` VARCHAR(45) NULL DEFAULT NULL,  
  `Dioica` TINYINT(1) NULL DEFAULT NULL,  
  `DimMax` DOUBLE UNSIGNED NULL DEFAULT NULL,  
  `IndiceManut` DECIMAL(5,2) UNSIGNED NULL DEFAULT '4.20',  
  `CostoBase` DECIMAL(6,2) UNSIGNED NULL DEFAULT NULL,  
  `Infestante` TINYINT(1) NULL DEFAULT NULL,  
  `Sempreverde` TINYINT(1) NULL DEFAULT NULL,  
  `Famiglia` INT(11) NULL DEFAULT NULL,  
  PRIMARY KEY (`idPianta`),  
  CONSTRAINT `fk_pianta_famiglia`  
    FOREIGN KEY (`Famiglia`)  
    REFERENCES `progettouni`.`Famiglia` (`idFamiglia`)  
    ON DELETE SET NULL  
    ON UPDATE CASCADE)  
ENGINE = InnoDB  
AUTO_INCREMENT = 601  
DEFAULT CHARACTER SET = latin1;  
  
CREATE INDEX `fk_pianta_famiglia_idx` ON `progettouni`.`Pianta` (`Famiglia` ASC);
```

9.3 Popolamento

Per fare il popolamento, specialmente per le tabelle più corpose, abbiamo costruito dei generatori di valori casuali in C++, così da poter effettuare inserimenti massicci in breve tempo. Da notare che, comunque, i valori sono stati generati in maniera tale da risultare coerenti e realistici, così da poter avere una vera simulazione del funzionamento del sistema a regime.

Di seguito, una parte del codice di popolamento della tabella “Pianta”.

```
INSERT INTO `progettouni`.`Pianta` (`Nome`, `CostoBase`, `IndiceManut`, `DimMax`, `Famiglia`) VALUES ('Calceolaria', '26', '3.88', '0.79', '4');
INSERT INTO `progettouni`.`Pianta` (`Nome`, `CostoBase`, `IndiceManut`, `DimMax`, `Famiglia`) VALUES ('Calla', '58', '8.96', '0.79', '4');
INSERT INTO `progettouni`.`Pianta` (`Nome`, `CostoBase`, `IndiceManut`, `DimMax`, `Famiglia`) VALUES ('Callisia', '146', '3.52', '2.43', '4');
INSERT INTO `progettouni`.`Pianta` (`Nome`, `CostoBase`, `IndiceManut`, `DimMax`, `Famiglia`) VALUES ('Callistemone', '111', '5.39', '1.97', '4');
INSERT INTO `progettouni`.`Pianta` (`Nome`, `CostoBase`, `IndiceManut`, `DimMax`, `Famiglia`) VALUES ('Camarodea', '16', '3.22', '2.44', '4');
INSERT INTO `progettouni`.`Pianta` (`Nome`, `CostoBase`, `IndiceManut`, `DimMax`, `Famiglia`) VALUES ('Cambria', '116', '6.50', '2.28', '4');
INSERT INTO `progettouni`.`Pianta` (`Nome`, `CostoBase`, `IndiceManut`, `DimMax`, `Famiglia`) VALUES ('Camelia', '115', '5.44', '2.48', '4');
INSERT INTO `progettouni`.`Pianta` (`Nome`, `CostoBase`, `IndiceManut`, `DimMax`, `Famiglia`) VALUES ('Campsis', '97', '0.77', '2.9', '4');
INSERT INTO `progettouni`.`Pianta` (`Nome`, `CostoBase`, `IndiceManut`, `DimMax`, `Famiglia`) VALUES ('Capelvenere', '145', '3.70', '0.8', '4');
INSERT INTO `progettouni`.`Pianta` (`Nome`, `CostoBase`, `IndiceManut`, `DimMax`, `Famiglia`) VALUES ('Caprifoglio', '5', '3.43', '3.80', '4');
INSERT INTO `progettouni`.`Pianta` (`Nome`, `CostoBase`, `IndiceManut`, `DimMax`, `Famiglia`) VALUES ('Cicas', '110', '4.18', '0.42', '4');
INSERT INTO `progettouni`.`Pianta` (`Nome`, `CostoBase`, `IndiceManut`, `DimMax`, `Famiglia`) VALUES ('Ciclamino', '51', '9.32', '3.36', '4');
INSERT INTO `progettouni`.`Pianta` (`Nome`, `CostoBase`, `IndiceManut`, `DimMax`, `Famiglia`) VALUES ('Cimbidio', '64', '0.91', '1.5', '4');
INSERT INTO `progettouni`.`Pianta` (`Nome`, `CostoBase`, `IndiceManut`, `DimMax`, `Famiglia`) VALUES ('Cineraria', '97', '8.2', '4.90', '4');
INSERT INTO `progettouni`.`Pianta` (`Nome`, `CostoBase`, `IndiceManut`, `DimMax`, `Famiglia`) VALUES ('Cipero', '6', '9.12', '2.16', '4');
INSERT INTO `progettouni`.`Pianta` (`Nome`, `CostoBase`, `IndiceManut`, `DimMax`, `Famiglia`) VALUES ('Cirtomio', '121', '4.14', '2.57', '4');
INSERT INTO `progettouni`.`Pianta` (`Nome`, `CostoBase`, `IndiceManut`, `DimMax`, `Famiglia`) VALUES ('Cisso', '78', '0.29', '3.86', '4');
INSERT INTO `progettouni`.`Pianta` (`Nome`, `CostoBase`, `IndiceManut`, `DimMax`, `Famiglia`) VALUES ('Cisto', '19', '0.30', '4.41', '4');
INSERT INTO `progettouni`.`Pianta` (`Nome`, `CostoBase`, `IndiceManut`, `DimMax`, `Famiglia`) VALUES ('Citiso', '0', '2.53', '1.22', '4');
INSERT INTO `progettouni`.`Pianta` (`Nome`, `CostoBase`, `IndiceManut`, `DimMax`, `Famiglia`) VALUES ('Clematide', '32', '5.1', '1.77', '4');
INSERT INTO `progettouni`.`Pianta` (`Nome`, `CostoBase`, `IndiceManut`, `DimMax`, `Famiglia`) VALUES ('Clerodendro', '59', '4.59', '2.45', '4');
INSERT INTO `progettouni`.`Pianta` (`Nome`, `CostoBase`, `IndiceManut`, `DimMax`, `Famiglia`) VALUES ('Clivia', '143', '5.19', '3.13', '4');
INSERT INTO `progettouni`.`Pianta` (`Nome`, `CostoBase`, `IndiceManut`, `DimMax`, `Famiglia`) VALUES ('Chlorophytum', '38', '1.6', '1.28', '4');
INSERT INTO `progettouni`.`Pianta` (`Nome`, `CostoBase`, `IndiceManut`, `DimMax`, `Famiglia`) VALUES ('Clusia', '35', '3.1', '4.96', '4');
INSERT INTO `progettouni`.`Pianta` (`Nome`, `CostoBase`, `IndiceManut`, `DimMax`, `Famiglia`) VALUES ('Cocco', '64', '7.38', '1.96', '4');
INSERT INTO `progettouni`.`Pianta` (`Nome`, `CostoBase`, `IndiceManut`, `DimMax`, `Famiglia`) VALUES ('Cocos', '61', '0.80', '1.9', '4');
INSERT INTO `progettouni`.`Pianta` (`Nome`, `CostoBase`, `IndiceManut`, `DimMax`, `Famiglia`) VALUES ('Codonanthe', '52', '2.92', '0.3', '4');
INSERT INTO `progettouni`.`Pianta` (`Nome`, `CostoBase`, `IndiceManut`, `DimMax`, `Famiglia`) VALUES ('Coleus', '60', '8.82', '1.20', '4');
INSERT INTO `progettouni`.`Pianta` (`Nome`, `CostoBase`, `IndiceManut`, `DimMax`, `Famiglia`) VALUES ('Columnnea', '86', '5.13', '1.13', '4');
INSERT INTO `progettouni`.`Pianta` (`Nome`, `CostoBase`, `IndiceManut`, `DimMax`, `Famiglia`) VALUES ('Cordiline', '62', '0.20', '2.43', '4');
INSERT INTO `progettouni`.`Pianta` (`Nome`, `CostoBase`, `IndiceManut`, `DimMax`, `Famiglia`) VALUES ('Corna di cervo o d'alce', '52', '2.8', '4.31', '4');
INSERT INTO `progettouni`.`Pianta` (`Nome`, `CostoBase`, `IndiceManut`, `DimMax`, `Famiglia`) VALUES ('Corona di spine', '41', '7.27', '2.13', '4');
INSERT INTO `progettouni`.`Pianta` (`Nome`, `CostoBase`, `IndiceManut`, `DimMax`, `Famiglia`) VALUES ('Criptanto', '135', '3.56', '0.98', '1');
```

9.4 Trigger, procedure e event

9.4.1 Introduzione

Per implementare tutte le richieste all'interno delle specifiche del progetto, abbiamo creato 47 trigger (contando una volta sola i trigger di insert/update/delete), 12 procedure, 9 event e 3 function (oltre a queste, in realtà, nel database sono presenti altre procedure e function, utilizzate, però, solo all'interno di trigger e event). Oltre a questi strumenti, abbiamo costruito una tabella per memorizzare le sezioni dedicate all'isolamento.

Di seguito, un elenco con il nome di ognuno di questi costrutti accompagnato da una breve descrizione. In fondo al paragrafo sono invece spiegate le operazioni più complesse implementate in questa parte.

9.4.2 Trigger

1. Trigger-EventCreazioneMailingList

- Insieme di trigger e event che organizzano una mailing list. I dettagli sul suo funzionamento si trovano alla fine del paragrafo.

2. TriggerControlloValiditaValutazioneCostoEntitaInterventoInsert
 - Trigger che, all'inserimento di un intervento, oltre a verificarne la validità, ne calcola una stima dell'entità e del costo. Vedere la fine del paragrafo per i dettagli.
3. TriggerControlloValiditaUtilizzoElemento (insert/update)
 - Trigger che controlla la validità di un inserimento/aggiornamento nella tabella "UtilizzoElemento".
4. TriggerControlloValiditaPeriodoPotatura
 - Trigger con cui si controlla la validità di un inserimento in "PeriodoPotatura".
5. TriggerControlloValiditaTipoConcimazione
 - In maniera simile al "TriggerControlloValiditaPeriodoPotatura", anche qui facciamo gli stessi controlli sui periodi.
6. TriggerControlloValiditaTerreno (insert/update)
 - Questo trigger controlla la correttezza di un inserimento/aggiornamento nella tabella "Terreno".
7. TriggerControlloValiditaTemperatura (insert/update)
 - Questo trigger controlla la correttezza di un inserimento/aggiornamento nella tabella "Temperatura".
8. TriggerControlloValiditaSettore (insert/update)
 - Questo trigger controlla la correttezza di un inserimento/aggiornamento nella tabella "Settore".
9. TriggerControlloValiditaScheda (insert/update)
 - Trigger che controlla la correttezza di un inserimento/aggiornamento nella tabella "Scheda".
10. TriggerControlloValiditaProdotto (insert/update)
 - Trigger che controlla la correttezza di un inserimento/aggiornamento nella tabella "Prodotto".
11. TriggerControlloValiditaPreferenzePeriodi (insert/update)
 - Trigger che controlla la correttezza di un inserimento/aggiornamento nella tabella "PreferenzePeriodi".
12. TriggerControlloValiditaPeriodoRinvasi (insert/update)
 - In maniera simile al "TriggerControlloValiditaPeriodoPotatura", anche qui si fanno gli stessi controlli sui periodi.
13. TriggerControlloValiditaPeriodoNonUtilizzo (insert/update)
 - In maniera simile al "TriggerControlloValiditaPeriodoPotatura", anche qui si fanno gli stessi controlli sui periodi.
14. TriggerControlloValiditaPeriodoCicli (insert/update)
 - In maniera simile al "TriggerControlloValiditaPeriodoPotatura", anche qui si fanno gli stessi controlli sui periodi; inoltre controlliamo la correttezza dell'attributo "Fio/Fru".

15. TriggerControlloValiditaPeriodoAttacchi (insert/update)
 - In maniera simile al "TriggerControlloValiditaPeriodoPotatura", anche qui si fanno gli stessi controlli sui periodi.
16. ControlloValiditaNecessitaPotatura (insert/update)
 - Trigger che controlla l'esistenza di un periodo in cui è possibile effettuare una potatura su una data pianta nel momento in cui viene definita necessaria.
17. TriggerControlloValiditaMisurazioneContenitore (insert/update)
 - Questo trigger controlla la correttezza di un inserimento/aggiornamento nella tabella "MisurazioneContenitore".
18. TriggerControlloValiditaMisurazioneAmbientale (insert/update)
 - Questo trigger controlla la correttezza di un inserimento/aggiornamento nella tabella "MisurazioneAmbientale"
19. TriggerControlloValiditaLuce (insert/update)
 - Questo trigger controlla la correttezza di un inserimento/aggiornamento nella tabella "Luce"
20. TriggerControlloValiditaGiardino (insert/update)
 - Questo trigger controlla la correttezza di un inserimento/aggiornamento nella tabella "Giardino"; in più, inserisce il valore "Numero" relativamente alla quantità di giardini inseriti dallo stesso utente.
21. TriggerControlloValiditaElemento (insert/update)
 - Trigger che controlla la correttezza di un inserimento/aggiornamento nella tabella "Elemento".
22. TriggerControlloValiditaContenitore (insert/update)
 - Trigger che controlla la correttezza di un inserimento/aggiornamento nella tabella "Contenitore".
23. TriggerControlloValiditaCondizioniFavorevoli (insert/update)
 - Trigger che controlla la correttezza di un inserimento/aggiornamento nella tabella "CondizioniFavorevoli".
24. TriggerControlloValiditaComposizioneSettore (insert/update)
 - Trigger che controlla la correttezza di un inserimento/aggiornamento nella tabella "ComposizioneSettore".
25. TriggerControlloValiditaComposizioneContenitore (insert/update)
 - Trigger che controlla la correttezza di un inserimento/aggiornamento nella tabella "ComposizioneContenitore"
26. TriggerControlloValiditaComponente (insert/update)
 - Questo trigger controlla la correttezza di un inserimento/aggiornamento nella tabella "Componente".
27. TriggerControlloValiditaBasato (insert/update)
 - Trigger che controlla la correttezza di un inserimento/aggiornamento nella tabella "Basato".

28. TriggerControlloValiditaAcqua (insert/update)
 - Questo trigger controlla la correttezza di un inserimento/aggiornamento nella tabella "Acqua".
29. TriggerControlloValiditaQuantitaElementoPerConcimazione (insert/update)
 - Trigger che, all'inserimento/aggiornamento di un'occorrenza di quantità, controlla che l'elemento inserito sia effettivamente utilizzato da quel tipo di concimazione.
30. TriggerControlloValiditaTipoInterventoPotatura
 - Trigger che, all'inserimento di un intervento di potatura, controlla se è possibile effettuarla durante il periodo specificato.
31. TriggerControlloValiditaTipoInterventoConcimazione
 - Trigger che, all'inserimento di un intervento di concimazione, controlla se è possibile effettuarla durante il periodo specificato.
32. TriggerCreazioneNuovaScheda
 - Trigger che, nel momento in cui un ordine diventa 'Evaso', crea una nuova scheda relativa all'utente e agli esemplari interessati.
33. TriggerControlloInserimentoPiantaSettore (insert/update)
 - Con questo trigger si controlla che l'utente, prima di inserire/spostare una pianta in un settore, lo abbia effettivamente completato; poiché si considera accettabile come forma per un settore qualsiasi poligono, consideriamo 3 il numero minimo di punti necessario.
34. ControllaNumeroPiante (insert/update)
 - Trigger che controlla se, all'inserimento/spostamento di un contenitore in una sezione o in una serra, queste ultime sono piene (ed eventualmente blocca l'inserimento).
35. AggiornaNumeroPiante (insert/update/delete)
 - Trigger che, all'inserimento/spostamento/eliminazione di un contenitore, aggiorna il numero di piante nelle serre e sezioni relative.
36. TriggerAggiornaCredibilitàUtente(CreazionePost/CreazioneRisposta/CreazioneVoto)
 - Trigger che, all'inserimento di un nuovo post/risposta/voto, aggiorna la credibilità dell'utente interessato come spiegato nel paragrafo 2.2.3
37. TriggerCalcolaDimensioneDataNascitaCosto
 - Trigger che, all'inserimento di un esemplare ne calcola dimensione e costo e, eventualmente, ne inizializza la data di nascita con la data corrente.
38. TriggerDistribuzioneEsemplare
 - Trigger che, all'inserimento di un nuovo esemplare, crea una nuova occorrenza di contenitore in cui inserisce il terreno e l'irrigazione più adatti (eventualmente possono poi essere modificati dall'azienda); da notare che è comunque compito degli addetti indicare la composizione effettiva del terreno.

39. TriggerAggiornaNumeroMassimoPianteSerra (insert/update/delete)
 - Trigger che, all'inserimento/aggiornamento/eliminazione di una sezione, aggiorna il numero massimo di piante nella serra corrispondente.
40. TriggerAggiornaNumeroMassimoPianteSezione (insert/update/delete)
 - Trigger che, all'inserimento/aggiornamento/eliminazione di un ripiano, aggiorna il numero massimo di piante nella sezione corrispondente.
41. TriggerAggiornaNumeroPianteOrdine
 - Trigger che, nel momento in cui si crea un' associazione fra un esemplare e un ordine, aggiorna il numero di piante nella serra, nella sezione e nel ripiano corrispondente.
42. TriggerAggiornaTerrenoSettore/Contenitore (insert/update)
 - Trigger che calcola un terreno a partire dalle sue componenti, come spiegato nel paragrafo 2.2.3.
43. TriggerAggiornaLuceSettore (insert/update/delete)
 - Trigger che, all'inserimento di una pianta in un settore, calcola la luce attuale del settore stesso, come spiegato nel paragrafo 2.2.3.
44. TriggerAggiornaEsigenzeTerreno
 - Trigger che, all'inserimento di un'occorrenza di "EsigenzeElemento", si assicura di mantenere coerente l'informazione con "EsigenzeTerreno".
45. TriggerControlloMisurazioni(Ambientali/Contenitore/Elementi)Esigenze
 - Trigger che, all'inserimento di una misurazione, controlla che questa sia in linea con le esigenze delle piante.
46. TriggerControlloValiditaContrasto
 - Trigger che, all'inserimento di una nuova occorrenza in "Contrasto", lo blocca se "Chimico" = "FALSE" per l'agente corrispondente.
47. TriggerControlloValiditaVoto (insert/update)
 - Trigger che controlla la validità di un inserimento/aggiornamento nella tabella "Voto".

9.4.3 Procedure

1. Procedure-TriggerGestioneManutenzioneAutomatica
 - Insieme di procedure e trigger che calcola gli interventi necessari ad un esemplare se per esso è stata scelta l'opzione di manutenzione automatica/programmata. Vedere la fine del paragrafo per i dettagli.
2. ProcedureCalcoloDimensioneEsemplare
 - Questa procedure calcola, a partire da un esemplare e dalla sua data di nascita, la sua dimensione aerea attuale basandosi sul suo indice di accrescimento, come descritto nel capitolo 2.
3. ProcedureCalcoloDimensioneRadicalePianta
 - Simile alla procedure precedente, calcola la dimensione radicale attuale di un esemplare.
4. ProcedureAggiornaDuplicati

- Procedure che controlla se esistono due o più terreni duplicati; nel caso, li inserisce nella materialized view “MV_TerreniDuplicati”. Tale procedure è chiamata dall’event “EventAggiornaTerreniDuplicati”.
5. ProcedureCreazioneNuovoOrdine
 - Procedure che crea un nuovo ordine. Per farlo esegue le seguenti azioni: crea una nuova occorrenza della tabella “Ordine”; trova gli esemplari disponibili per la specie di pianta che ci interessa (eliminando quindi sia quelli già venduti, sia quelli malati); per ogni esemplare trovato, crea una nuova occorrenza di “Relativo” (almeno finché non si soddisfa la richiesta); se gli esemplari disponibili non sono sufficienti, crea una nuova occorrenza di “Pendente”.
 6. ProcedureAggiuntaPiantaSettore
 - Procedure che permette all'utente di aggiungere in maniera semplice una nuova pianta in un dato settore, eventualmente associata ad un vaso.
 7. ProcedureAvanzaStatoOrdine
 - Questa procedure permette di far avanzare lo stato di un ordine (con l’eccezione degli ordini pendenti) senza costringere l’utente a effettuare update personalmente.
 8. ProcedureListaInterventi
 - Procedure che restituisce come result set gli interventi da effettuare o già effettuati su un dato esemplare.
 9. ProcedureListaMisurazioni
 - Procedure con cui si mostrano a video le misurazioni relative ad un dato esemplare, in una specifica data
 10. ProcedureValutazionePreferenze
 - Procedure che, a partire da un dato set di preferenze, individua le piante che più ci si avvicinano. Vedere la fine del paragrafo per i dettagli.
 11. ProcedureSuggerimentoSettore
 - Procedure che, dato un settore, individua le piante da consigliare in base alla compatibilità con tale settore. Vedere la fine del paragrafo per i dettagli.
 12. ProcedureCalcoloTerrenoSettore(Contenitore)
 - Procedure che, dato un settore (contenitore), ne calcola le caratteristiche a partire dalle componenti come spiegato nel paragrafo 2.2.3

9.4.3 Event

1. EventCalcoloInterventiGiornalieri
 - Event che, ogni giorno, popola la materialized view “MV_InterventiGiornata” con gli interventi da effettuare in quella giornata. Vedere la fine del paragrafo per i dettagli sul funzionamento.
2. EventControlloContenitoreEsemplare
 - Event con cui si verifica che a ogni esemplare in azienda sia associato un contenitore.
3. EventControlloCoperturaCicliPianta
 - Event con cui si controlla che, per ogni pianta, i cicli di quella pianta coprano tutto l'anno; se ciò non vale per qualche pianta, manda un messaggio di errore.
4. EventCalcoloCondizioniFavorevoli

- Event che, ogni anno, analizza i dati sulle malattie per individuare le condizioni più favorevoli per ogni agente. Vedere la fine del paragrafo per i dettagli.
5. EventPopolamentoReportAssunzioni
 - Event che, ogni mese, crea un report dove prevede eventuali carenze di personale nei sei mesi successivi. Vedere la fine del paragrafo per i dettagli.
 6. EventGestioneMalattia
 - Event che, alla fine di ogni giornata, valuta tutti i report di diagnostica. Per ogni report, stila una classifica (con metodo di “dense ranking”) delle malattie più probabili per i sintomi individuati. In più, sposta gli esemplari malati in sezioni di isolamento apposite.
 7. EventAggiornaNecessita
 - Event che, ogni anno, analizza i dati sugli interventi aggiustando le necessità di ogni pianta. Vedere la fine del paragrafo per i dettagli.
 8. EventPopolamentoReportConsigliAcquisto
 - Event che, ogni mese, popola un report in cui si consiglia all'azienda una quantità indicativa di esemplari da acquistare per ogni specie. In più, vengono anche segnalate piante che hanno avuto incrementi o decrementi significativi nei trend di vendita. Vedere la fine del paragrafo per i dettagli funzionamento.
 9. EventSegnalazionePianteDeboli
 - Event che, ogni anno, aggiorna il “ReportConsigliAcquisti” di dicembre segnalando le piante più vulnerabili. Vedere la fine del paragrafo per i dettagli.

9.4.4 Function

1. FunctionCalcoloPreventivoGiardino
 - Function che, dato l'account e il numero di un giardino, ne restituisce il preventivo.
2. FunctionCalcoloIndiceManutenzione
 - Function che, data una pianta, ne restituisce l'indice di manutenzione, calcolato come spiegato nel paragrafo 2.2.3.
3. FunctionCalcoloDistanzaMinima
 - Function che, date due piante, calcola la distanza minima a cui metterle per evitare che entrino in conflitto, secondo la modellizzazione spiegata al paragrafo 2.2.3.

9.4.5 Altro

1. CreazioneTabellaisolamento

- File SQL dove si crea una materialized view in cui vengono registrate tutte le sezioni dedicate alla quarantena di esemplari malati. Per sceglierle, si prende, per ogni serra, la sezione più piccola fra quelle che possono ospitare almeno 1/80 degli esemplari totali (stima sulla percentuale di esemplari malati).

9.4.6 Spiegazione delle operazioni più complesse

9.4.6.1 Creazione mailing list

Ogni giorno, vari trigger ed event si occupano di popolare la materialized view “MV_MailingList”. All’interno di tale mailing list vengono inseriti messaggi personalizzati per gli utenti.

I messaggi vengono inviati tramite email e vengono spediti nei seguenti casi:

- Notifica di un intervento da effettuare a breve (viene spedita due giorni prima della data prevista)
- Avviso di un cambio di prezzo per una pianta presente fra le piante preferite di un utente
- Per gli utenti che non hanno ancora inserito delle piante preferite, li invitiamo a farlo
- Email di benvenuto per gli utenti appena iscritti

Di seguito il format delle varie email:

- Notifica intervento:

“Ciao, (nome, se specificato, altrimenti nickname), ci risulta che il tuo esemplare di (nome della pianta) debba a breve effettuare un intervento di (tipo di intervento). Se hai scelto di usufruire del nostro servizio di manutenzione a domicilio, presto dei nostri addetti verranno ad occuparsene. Buon giardinaggio!”

- Avviso di cambio di prezzo:

“Ciao, (nome, se specificato, altrimenti nickname), sappiamo che sei interessato alla pianta (nome della pianta). Lo sai che si è appena abbassata di prezzo? Corri subito da noi a comprarla!”

- Invito a mettere delle piante preferite:

“Ciao, (nome, se specificato, altrimenti nickname), a quanto pare non ci hai ancora detto quali sono le tue piante preferite. Vieni subito a dircelo, così sappiamo se informarti quando una pianta scende di prezzo!”

- Email di benvenuto:

“Ciao, (nome, se specificato, altrimenti nickname), benvenuto nella nostra comunità! Nel nostro store troverai le migliori offerte di piante sul web, mentre nel nostro forum troverai consigli, notizie e molto altro dal mondo del giardinaggio! Vieni subito a scoprire tutti gli altri servizi offerti dalla nostra azienda, ti aspettiamo con ansia!”

```
#Qui si crea una mail per avvertire gli utenti che è necessario effettuare un intervento su una data pianta
create event EventAggiornaMailingListNotificaIntervento
on schedule every 1 day
starts '2016-06-12 00:00:01'
do
begin
set @Mail = 'Ciao ';
set @MailSecondaParte = ', ci risulta che il tuo esemplare di ';
set @MailTerzaParte = ' debba a breve effettuare un intervento di ';
set @MailQuartaParte = '. Se hai scelto di usufruire del nostro servizio di manutenzione a domicilio, presto dei nostri addetti verranno ad occuparsene. Buon giardinaggio!';
insert into MV_MailingList ('IndirizzoMail', 'Mail')
select A.Email,
       if(A.Nome is not null,
          concat(@Mail,
                 concat(A.Nome,
                        concat(@MailSecondaParte,
                               concat(S.NomePianta,
                                      concat(@MailTerzaParte,
                                             concat(I.Tipo, @MailQuartaParte)))))),
          concat(@Mail,
                 concat(A.Nickname,
                        concat(@MailSecondaParte,
                               concat(S.NomePianta,
                                      concat(@MailTerzaParte,
                                             concat(I.Tipo, @MailQuartaParte)))))),
       from Intervento I inner join Scheda S
       on I.Esemplare = S.Esemplare
       inner join Account A
       on S.Account = A.idAccount
       where I.Data = current_date() + interval 2 day and
              I.Effettuato = false and
              (I.Motivo = 'Programmato' or
               I.Motivo = 'Automatico');
```

9.4.6.2. Calcolo interventi necessari ad un esemplare

Dato un esemplare, la pianificazione degli interventi da effettuare su di esso si basa sul seguente procedimento:

- Per ogni anno, vengono previsti gli interventi presenti in “NecessitaPotatura” e “EsigenzeConcimazione” per la specie a cui appartiene l’esemplare
- A questi viene aggiunto un trattamento preventivo in tutti quei casi in cui c’è una probabilità superiore al 50% di subire attacchi da parte di qualche agente
- A questo punto vengono inseriti degli interventi aggiuntivi per i primi due anni di vita (in cui la pianta è più delicata) secondo la seguente politica:

Indice di Manutenzione	1° Anno	2° Anno
Basso	Una concimazione o un trattamento	NA
Medio	Una concimazione e un trattamento	Una concimazione o un trattamento
Alto	Due concimazioni e due trattamenti	Una concimazione e un trattamento

- Per scegliere se effettuare una concimazione o un trattamento, si ragiona in questo modo: per ogni elemento che la pianta necessita, si conta +10; se la probabilità più

alta di subire un attacco (escluse quelle che superano il 50%) è superiore al numero trovato prima, si sceglie il trattamento, altrimenti si sceglie la concimazione.

- Per scegliere il trattamento da effettuare, basta prendere l'agente con più alta probabilità di attacco (esclusi quelli che superano il 50%)
- Per scegliere la concimazione, si sceglie quella che mette a disposizione più elementi fra quelli necessari alla pianta

9.4.6.3 Valutazione delle preferenze

La valutazione delle piante che rispondono ad un dato set di preferenze si effettua nel seguente modo:

- Prima si eliminano le piante che non rispettano le condizioni “Dioica”, “Infestante”, “Sempreverde”
- Si eliminano poi le piante che non rientrano nei range di “Dimensione”, “Costo”, “IndiceManut”; questi range sono tanto più ampi, tanto meno importanti sono per l'utente.
- Si eliminano poi le piante che non rientrano nei range di “Luce”, “Acqua” e “Temperatura” e quelle che non possono vivere nel terreno specificato; anche qui si valuta per range.
- Infine si tolgono le piante che non rispettano almeno il 60% delle condizioni sui periodi
- In generale, i valori NULL vengono sostituiti da range il più ampi possibili

Vedere il codice della ‘ProcedureValutazioniPreferenze’ per maggiori dettagli su come funzionano i range e come questi variano in funzione dell'importanza che l'utente attribuisce all'elemento in questione.

9.4.6.4 Calcolo piante suggerite in un settore

Per valutare le piante più adatte ad un settore, si segue questo procedimento:

- In base all'indice di manutenzione e al clima del giardino, si fa una prima selezione delle piante (come per la valutazione delle preferenze)
- Il terreno (se il settore non è pavimentato), fornisce un ulteriore elemento di selezione
- Il costo viene valutato in base al costo medio delle piante acquistate dall'utente negli ultimi 6 mesi (di default è medio)
- Infine si dà maggior priorità alle piante che coprono un periodo di fioritura il più ampio possibile fra i periodi non ancora coperti dalle piante del settore.

9.4.6.5 Aggiornamento delle necessità di una pianta

Ogni anno, analizzando gli interventi effettuati nel corso dell'anno, si aggiungono nuove necessità alle piante che ne hanno bisogno.

Per le potature e le concimazioni utilizziamo il seguente sistema:

- Se più del 10% degli utenti che ha acquistato una data pianta ha richiesto un particolare intervento, questo viene inserito fra le necessità
- Se è stato richiesto fra il 5 e il 10% viene inserito solo se l'intervento rappresenta più della metà delle richieste totali

La gestione dei trattamenti preventivi funziona invece in maniera diversa:

- in caso sia stato richiesto un intervento contro un particolare agente per più del 10% degli esemplari acquistati di una data pianta, la percentuale di richieste sul totale viene aggiunta alle probabilità di attacco di quell'agente su quella pianta (non oltre il 51%, per evitare crescite esponenziali incontrollate)

All'aggiornamento delle necessità segue poi un aggiornamento dell'indice di manutenzione per tutte le piante, tramite il sistema spiegato al paragrafo 2.2.3.

9.4.6.6 Calcolo delle condizioni favorevoli ad un agente

Una volta all'anno, ricalcoliamo quali sono le condizioni favorevoli per gli agenti basandoci sui dati delle misurazioni nelle serre e nei contenitori nel periodo precedente all'insorgenza di una malattia.

Il procedimento seguito è questo:

- Si prende la situazione di ogni condizione (illuminazione, idratazione, ecc) nella settimana precedente l'insorgenza della patologia (per il database, dal momento in cui è stata individuata con un report di diagnostica)
- Si controlla se, per ogni particolare condizione, l'occorrenza di malattie a essa associata è superiore del 10% alla media generale

Le condizioni che superano tale selezioni vengono inserite nella tabella "CondizioniFavorevoli".

9.4.6.7 Gestione degli interventi quotidiani e report sulle assunzioni

La politica di gestione degli interventi si basa sulla stima di 250 dipendenti divisi in 50 sezioni sul territorio (rappresentate nel database dai dati nell'attributo "Citta" di "Account").

La tecnica di gestione funziona nel seguente modo:

- Si considerano solo gli interventi da effettuare entro al più 15 giorni.
- Si assegnano 15 punti agli interventi che devono essere fatti entro oggi; per gli altri si toglie 1 punto per ogni giorno in avanti, mentre se ne aggiungono 3 per ogni giorno indietro
- Gli interventi su richiesta guadagnano 5 punti
- Se in questo lasso di tempo ci sono due o più interventi da effettuare presso uno stesso utente, questi guadagnano tutti il punteggio del più alto, più 10 punti ulteriori (così si evita di doversi recare da uno stesso utente in brevi lassi di tempo)
- Si stila una classifica per città, aggiungendo interventi finché l'entità totale per quella città non raggiunge 100 (si stima che un addetto possa fare in un giorno al più interventi per un'entità totale di 20)
- Se in una città restano più di 20 punti, questi vengono assegnati alla città che si è vista escludere più interventi nell'assegnamento

Per individuare se sarà necessario assumere personale in un dato periodo, è sufficiente verificare che la somma delle entità degli interventi da effettuare nel mese successivo non sia superiore all'80% di:

$$N \times 20 \times 22$$

Dove N è il numero degli addetti che si occupano degli interventi, 20 l'entità totale che un addetto può gestire in un giorno e 22 il numero medio di giorni lavorativi in un mese. Si prende l'80% così da avere margine in caso di non ottimizzazione delle ore lavorative e per coprire gli interventi che verranno aggiunti nel corso del tempo (per richieste da parte degli utenti o semplicemente per nuovi acquisti).

9.4.6.8 Spiegazione popolamento report consigli acquisti

Ogni mese, il database compila un report per ogni pianta, in cui consiglia il numero di esemplari da acquistare basandosi sul numero di ordini pendenti, sul trend di vendita nei mesi precedenti e sull'eventuale corrispondenza con l'inizio di periodi di fioritura e/o fruttificazione. Il metodo che abbiamo usato per organizzare la compilazione è descritto qui di seguito.

Cominciamo col definire le variabili da utilizzare (si fa qui riferimento ad una specifica pianta):

- X rappresenta il numero di acquisti consigliati
- V rappresenta il numero di esemplari venduti durante il mese appena finito
- VP e VPP rappresentano rispettivamente gli esemplari venduti nel mese precedente a quello appena finito e quelli venduti nel mese ancora precedente
- P rappresenta il numero di ordini in pendenza
- M rappresenta il numero di esemplari in magazzino

Per prima cosa, poniamo:

$$X = V + P - M.$$

Se si ha:

$$M < 20\% \times V$$

allora si aumenta X del 5% per sicurezza.

Si va poi ad analizzare le vendite nel mese precedente. Definita C la crescita percentuale fra VP e V , e CM la crescita percentuale fra VPP e VP , si ha:

- se $C < 5\%$, non si fa nulla (sono considerate normali variazioni di mercato)
- se $5\% < C < 10\%$ e $CM > 0$, allora si aumenta X di una percentuale C
- se $C > 10\%$ e $CM > -10\%$, allora si aumenta X della percentuale C ; se $CM < -10\%$, si aumenta X del 10%, qualunque sia il valore di C (così da non essere ingannati da crescite fittizie, ovvero ottenute a causa di una forte decrescita precedente)

Infine, se il mese attuale corrisponde all'inizio di un periodo di fioritura e/o fruttificazione, si aumenta X di un ulteriore 10% (perché le piante con fiori e/o frutti sono più attraenti).

Da notare che, se c'è stata una (de)crescita superiore al 10%, questa viene comunque segnalata tramite l'attributo 'Note'.

Nel report di dicembre, inoltre, si segnalano quelle piante che, nel corso dell'anno, hanno mostrato di essere particolarmente vulnerabili alle malattie. Queste piante sono individuate se il rapporto fra gli esemplari che si sono ammalati e quelli totali supera del 20% la media generale.

10. Progettazione Fisica

10.1 Introduzione

Dopo aver finito di implementare il nostro database su Mysql, ci siamo occupati della progettazione fisica.

Abbiamo quindi individuato una query da accelerare tramite l'introduzione di un indice e ne abbiamo analizzato le prestazioni prima e dopo l'introduzione di tale indice.

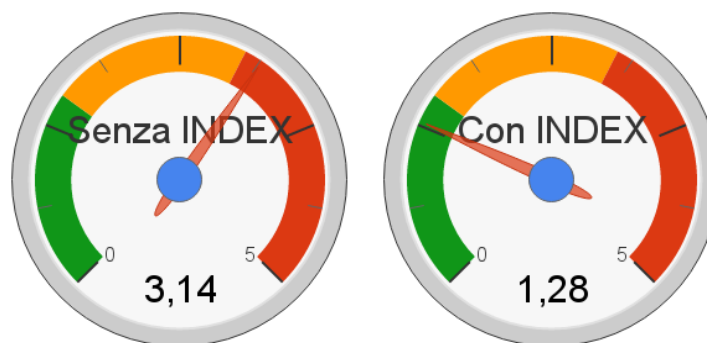
10.2 Analisi delle prestazioni

L'operazione che abbiamo ritenuto più adatta è stata la selezione di interventi effettuati in un intervallo di tempo: tale operazione può risultare interessante all'azienda e, allo stesso tempo, si presta bene all'applicazione di indici. Di seguito la query:

```
SELECT *  
from progettouni.intervento2 I  
where Effettuato = 1 and  
      Data > '2007-12-15' and  
      Data <= '2007-12-15';
```

Abbiamo scelto, per accelerare questa query, di inserire un indice su "Data".

Dopo aver effettuato vari test, la media dei risultati, in secondi, è stata la seguente:



Da notare che, per ottenere margini così significativi, abbiamo dovuto popolare le tabelle interessate con una grande quantità di dati, dopodiché abbiamo sovraccaricato il server effettuando, contemporaneamente alla query, un popolamento molto pesante.