

M070 - ESAME DI STATO DI ISTITUTO TECNICO INDUSTRIALE

Indirizzo: INFORMATICA

Tema di: INFORMATICA GENERALE E APPLICAZIONI TECNICO-SCIENTIFICHE

LA TRACCIA

Un vivaio vuole realizzare una base di dati per gestire le sue attività di vendita di piante e le sue attività esterne.

Si vogliono memorizzare, oltre alle informazioni generali sulle diverse specie di piante, anche quelle relative alle specifiche piante presenti nel vivaio.

Di ogni specie deve essere registrato il nome, un'immagine, una breve descrizione, informazioni relative al modo di coltivazione e alle caratteristiche dell'esposizione ed infine se si tratta di pianta da interno o da esterno.

Si vogliono inoltre registrare i dati relativi alle piante effettivamente presenti nel vivaio, raggruppate per specie, eventualmente suddivise in diversi tipi in base al prezzo di vendita.

Per ogni specie (o per ogni tipo, se la specie è suddivisa in tipi), è necessario registrare il numero di esemplari presenti nel vivaio e il costo di ogni esemplare.

Si vogliono inoltre gestire le informazioni relative al personale che lavora nel vivaio (agronomi, operai, amministrativi) registrando i dati anagrafici, la qualifica e, per gli agronomi, l'anno di assunzione nel vivaio. Ogni singola specie del vivaio è sotto la responsabilità di un agronomo.

Le attività esterne del vivaio, quali ad esempio la potatura o la manutenzione dei giardini, sono svolte dal personale in base alla qualifica (in generale ogni lavoratore è in grado di svolgere più di un'attività) e sono caratterizzate da un codice, da un nome, da un costo orario. Per le attività esterne si vogliono mantenere tutte le informazioni relative ai clienti che richiedono le attività, in particolare se si tratta di privati o di aziende, la data di prenotazione e quella di effettuazione dell'intervento richiesto e se per l'intervento sono necessarie piante del vivaio.

Il candidato consideri la situazione sopra descritta, precisi eventuali ipotesi aggiuntive e realizzi:

- un'analisi della realtà di riferimento che illustri le premesse per i successivi passi della progettazione della base di dati;
- uno schema concettuale della base di dati;
- uno schema logico della base di dati;
- la definizione delle relazioni della base di dati in linguaggio SQL;
- le seguenti interrogazioni espresse in linguaggio SQL:
 1. dato il nome di una pianta, riportare quanti esemplari di quella pianta sono presenti nel vivaio;
 2. dato il nome di una stagione, visualizzare il nome delle piante che fioriscono in quella stagione;
 3. dato il nome di un intervento esterno, tra quelli previsti dal vivaio, riportare il nome e il telefono dei soggetti che hanno richiesto quell'intervento nel corso di un determinato anno solare;
 4. dato il nome di un agronomo, riportare quanti esemplari di piante sono sotto la sua responsabilità;
 5. visualizzare nome, descrizione e quantità di esemplari presenti nel vivaio, della pianta più economica da interno;
 6. riportare nome degli interventi richiesti non ancora evasi con il nome e il telefono del richiedente.

Il candidato sviluppi inoltre, a scelta uno dei seguenti moduli:

1. Si vuole realizzare un sito Internet che presenti al pubblico il vivaio illustrandone i prodotti e le diverse attività.
2. Si vuole consentire la gestione delle attività esterne del vivaio attraverso una prenotazione on-line da parte dei clienti. Illustrare le modalità di realizzazione di questa funzione e gli strumenti tecnici adottabili.

M070 - ESAME DI STATO DI ISTITUTO TECNICO INDUSTRIALE

Indirizzo: INFORMATICA

Tema di: INFORMATICA GENERALE E APPLICAZIONI TECNICO-SCIENTIFICHE

LA SOLUZIONE

[Analisi](#)

[Ipotesi aggiuntive](#)

[Schema concettuale](#)

[Schema logico](#)

[Definizione della relazioni](#)

[Interrogazioni](#)

[Moduli](#)

[Descrizione della piattaforma operativa](#)

ANALISI

La prima entità da considerare è quella che si riferisce alla "specie di piante". La definizione botanica di specie è "*Complesso di individui aventi gli stessi caratteri biologici e morfologici e che riproducendosi danno una prole feconda*" (Zingarelli). Quindi occorrenza di questa entità corrisponde ad una pianta con specifiche caratteristiche omogenee a tutti gli individui. (ad esempio Abete bianco è diverso da Abete rosso ma Geranio è una unica specie indipendentemente dal colore dei fiori.)

L'entità richiesta è una specie di enciclopedia botanica in cui le specie sono elencate in forma non gerarchica (senza tassonomia). Le proprietà di questa entità sono informazioni di carattere generale, non collegate all'azienda ma riferite solo alla specie che descrivono. Si vogliono memorizzare, oltre alle informazioni generali sulle diverse specie di piante, anche quelle relative alle specifiche piante presenti nel vivaio. Non tutte le specie presenti nell'"enciclopedia" sono effettivamente vendute dal vivaio. E' necessario quindi aggiungere informazioni relative alla realtà dell'azienda. Se questi dati fossero riferiti effettivamente ad ogni singola specie si potrebbe pensare di realizzare un "IS-A" inserendo una relazione 1:1 oppure campi facoltativi. Purtroppo le specifiche richiedono che le piante di una stessa specie siano classificate per prezzo di vendita. Il rispetto di questa specifica fa nascere una relazione 1:N tra l'entità "specie" (l'enciclopedia botanica) e l'entità "piante" (il magazzino del vivaio). Una pianta del vivaio è quindi identificata non solo dalla sua specie ma anche dalla sua classe di prezzo. La successiva entità da analizzare è l'anagrafe degli impiegati. Questa entità si presenta come una struttura "IS-A" perchè gli agronomi hanno un dato aggiuntivo rispetto al resto del personale. L'entità "personale" (nella sola parte agronomi) è collegata con una relazione 1:N alla entità piante per definire la responsabilità. Si potrebbe più semplicemente collegare a "specie" (sempre con una relazione 1:N) prevedendo il collegamento incompleto (per le specie non presenti nel vivaio) ma si violerebbe l'ipotesi che l'entità specie non sia in alcun modo collegato all'azienda.

Per gestire le attività esterne si devono prevedere due entità (attività e clienti). L'entità attività è una anagrafe che contiene tutti i tipi di attività svolte con le relative proprietà; l'entità clienti è una anagrafe dei clienti che contiene tutti i clienti già registrati con le proprietà essenziali. L'anagrafe attività è collegata con una relazione N:M all'anagrafe del personale per definire le competenze. Le due entità attività e clienti sono legate tra loro da una relazione N:M che è la registrazione di tutte le effettive attività richieste dai clienti e svolte dal personale. Non è possibile collegare direttamente il personale con i clienti mediante una relazione N:M di attività perchè in questo modo viene a mancare una anagrafe delle attività che quindi possono essere proposte ai clienti solo dopo che un cliente le ha effettivamente richieste creando un circolo vizioso.

IPOTESI AGGIUNTIVE

Entità "specie": La chiave primaria è artificiale ad autoincremento per semplificare il collegamento 1:N con "piante" ma va bene anche la scelta del nome come chiave primaria non essendoci problemi di omonimia. L'immagine un file esterno individuato dal suo percorso nel file system ma si può anche usare un BLOB (OLE in ambiente MS). E' stato aggiunto il campo "stagione" non inserito nelle specifiche iniziali per realizzare la interrogazione 2; in alternativa l'interrogazione 2 può essere realizzata con una ricerca con l'operatore LIKE nel campo "info" supponendo che l'informazione di fioritura sia sempre presente. I campi

esposizione, interno/esterno e stagione sono realizzati con il tipo SET in alternativa potevano anche essere ENUM, INT oppure CHAR. Se la responsabilità dell'agronomo è inserita in questa entità deve essere facoltativa.

Entità "piante": La chiave primaria è costituita dalla combinazione tra la chiave esterna della specie e la classe di prezzo. La classe di prezzo è realizzata con un tipo SET ('unico', 'piccolo', 'medio', 'grande'). Si può anche usare un INT come chiave esterna di una look-up table ma non può essere un prezzo perché una margherita piccola ha comunque un prezzo diverso da una sequoia piccola. Se la responsabilità dell'agronomo è inserita in questa entità deve essere obbligatoria.

Entità "personale": La chiave primaria è artificiale ad autoincremento per evitare i problemi generati dalle omonimie. Sono previste solo le proprietà strettamente necessarie per la risoluzione dei quesiti. L' "IS-A" degli agronomi è realizzato con un campo facoltativo. Un vincolo di integrità deve impedire che siano associate piante ai dipendenti non agronomi.

Entità "attività": La chiave primaria è artificiale ad autoincremento. Sono previste solo le proprietà strettamente necessarie per la risoluzione dei quesiti.

Entità "clienti": La chiave primaria è artificiale ad autoincremento. Sono previste solo le proprietà strettamente necessarie per la risoluzione dei quesiti.

Relazione "dettaglioattività": La relazione contiene oltre alle chiavi esportate le proprietà necessarie per definire una attività effettivamente prenotata e eventualmente svolta. Il riconoscimento se l'attività è stata svolta o no è realizzato con la facoltatività del campo "data effettuazione" (se il campo è NULL l'intervento non è ancora effettuato). Una possibile alternativa è la creazione di un campo booleano di effettuazione che ha il vantaggio di consentire di "programmare" la data di intervento (cioè scriverla in banca dati prima che l'intervento sia effettuato). La necessità di piante del vivaio è realizzata con un campo SET (si,no) e non con un collegamento alla entità piante perché le specifiche non richiedono di individuare quali piante sono necessarie. Inoltre le specifiche non richiedono né di individuare quale dipendente ha effettivamente eseguito l'intervento né la durata dell'intervento da cui ricavare il costo quindi queste relazioni e proprietà seppure molto importanti non sono state realizzate.

INTERROGAZIONI

1. dato il nome di una pianta, riportare quanti esemplari di quella pianta sono presenti nel vivaio. La presenza di più occorrenze della stessa pianta nella entità "piante" (classi di prezzo) rende non banale l'interrogazione. E' necessario infatti sommare i campi quantità di tutti i record che hanno lo stesso idspec. La query è predisposta per l'inserimento in uno script PHP.

SELECT

SUM(t1.qt)

FROM

piante AS t1,

specie AS t2

WHERE

t1.idspec=t2.idspec

AND

t2.nome=' \$nome '

2. dato il nome di una stagione, visualizzare il nome delle piante che fioriscono in quella stagione. L'inserimento della proprietà "stagione" nella entità "specie" rende particolarmente semplice la query.

SELECT

t1.nome

FROM

specie AS t1

WHERE

t1.stagione=' \$stagione '

3. dato il nome di un intervento esterno, tra quelli previsti dal vivaio, riportare il nome e il telefono dei soggetti che hanno richiesto quell'intervento nel corso di un determinato anno solare.

SELECT

```
t1.denom,  
t1.telef  
FROM  
  clienti AS t1,  
  attivita AS t2,  
  dettatt AS t3  
WHERE  
  t1.idcl=t3.idcl  
AND  
  t2.idatt=t3.idatt  
AND  
  t3.datap>='$anno-01-01'  
AND  
  t3.datap<='$anno-12-31'  
AND  
  t2.descriz='$descatt'
```

4. dato il nome di un agronomo, riportare quanti esemplari di piante sono sotto la sua responsabilità. La specifica è aperta a varie interpretazioni. In questa ipotesi il termine "esemplari" viene interpretato come singole pezzi e quindi vengono sommate le quantità ma si potrebbe anche interpretare come conteggio delle specie.

```
SELECT  
  SUM(t1.qt)  
FROM  
  piante AS t1,  
  personale AS t2  
WHERE  
  t1.idpers=t2.idpers  
AND  
  t2.nome=' $nome '  
AND  
  t2.cognome=' $cognome '
```

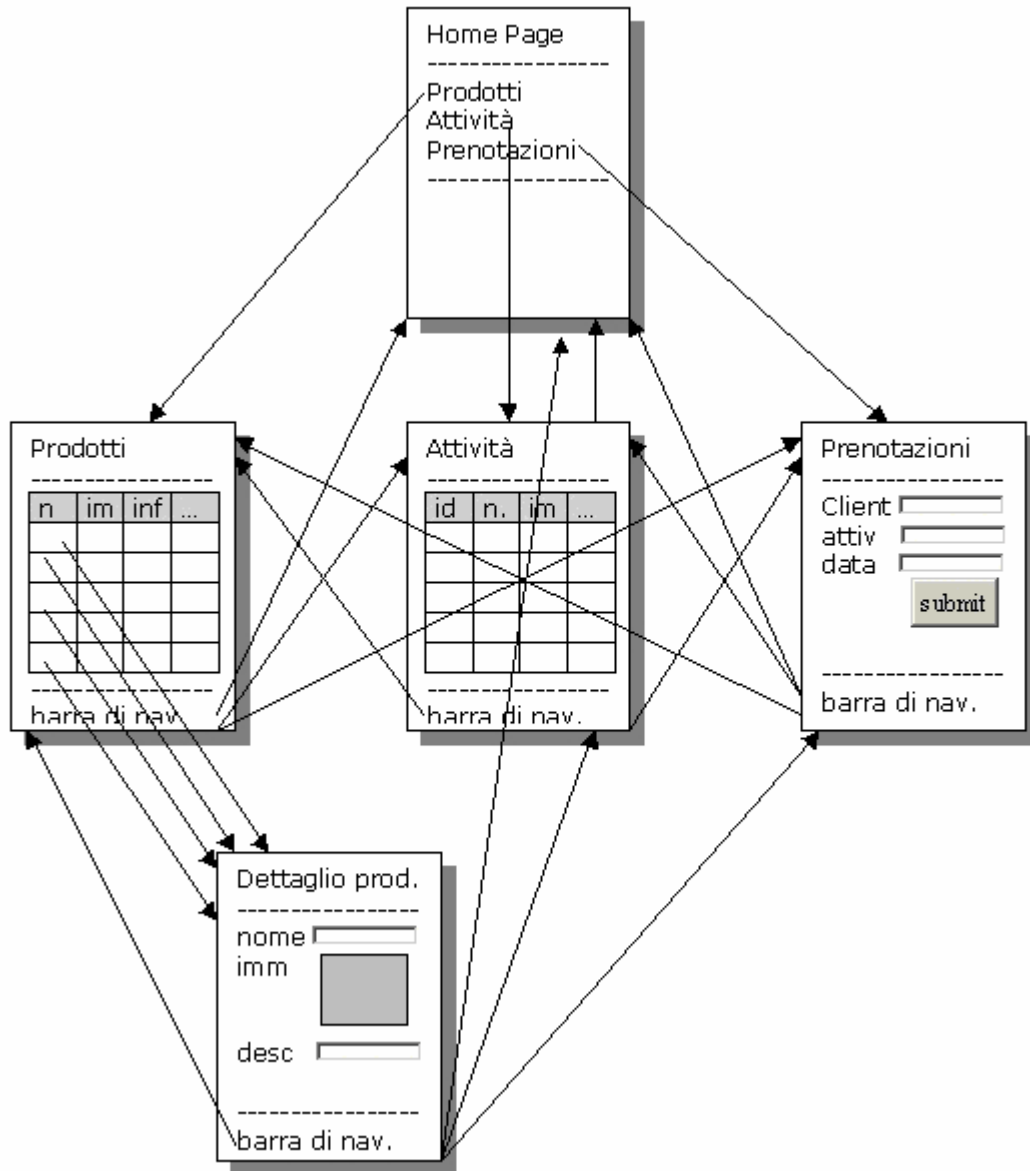
5. visualizzare nome, descrizione e quantità di esemplari presenti nel vivaio, della pianta più economica da interno. La ricerca viene effettuata utilizzando gli operatori ORDER BY e LIMIT 1; in alternativa si potrebbe usare la funzione MIN.

```
SELECT  
  t2.nome,  
  t2.descriz,  
  t1.costo,  
  t1.qt  
FROM  
  piante AS t1,  
  specie AS t2  
WHERE  
  t1.idspec=t2.idspec  
AND  
  t2.intext='interno'  
ORDER BY t1.costo ASC  
LIMIT 1
```

6. riportare nome degli interventi richiesti non ancora evasi con il nome e il telefono del richiedente.

```
SELECT  
  t2.descriz,  
  t1.denom,  
  t1.telefono  
FROM
```

```
clienti AS t1,  
attivit  AS t2,  
dettatt AS t3  
WHERE  
t1.idcl=t3.idcl  
AND  
t2.idatt=t3.idatt  
AND  
ISNULL(t3.datai)
```

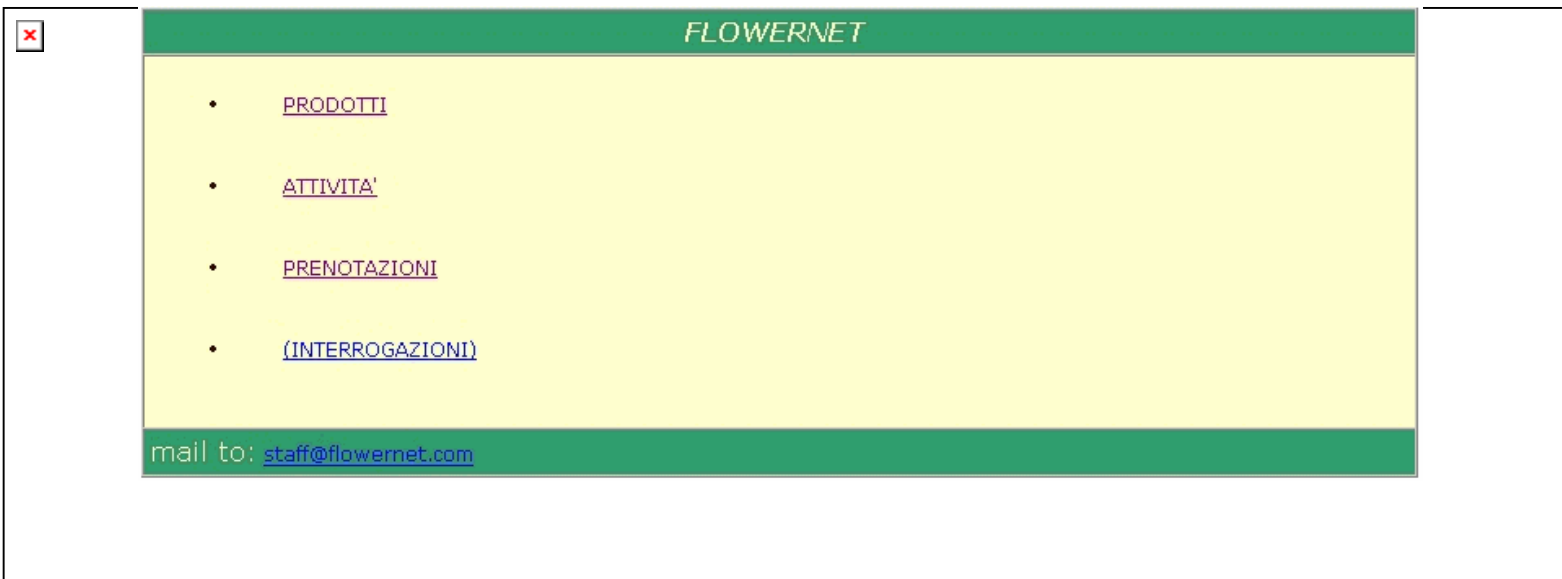


Grafo dei collegamenti

MODULI

1. Si vuole realizzare un sito Internet che presenti al pubblico il vivaio illustrandone i prodotti e le diverse attivit . Il sito Internet dell'azienda deve essere una vetrina che presenti le risorse offerte dall'azienda. In questo caso non   richiesta una particolare interattivit  ma poich  i dati sono variabili   necessario usare tecniche di web attivo. Per i dettagli sulla piattaforma vedere il successivo capitolo di [Descrizione della piattaforma operativa](#). In questo capitolo viene mostrato il grafo dei collegamenti e lo sviluppo delle pagine di presentazione dei prodotti e delle attivit .

Per ogni pagina sviluppata viene mostrata la presentazione, se la pagina è attiva viene anche mostrato lo pseudocodice in forma sintetica, la query in forma dettagliata e le eventuali variabili di ingresso. Se la pagina ha campi modulo viengono mostrate le variabili di uscita.



Home Page La home page non è attiva quindi viene mostrata solo la presentazione

Pagina di presentazione dei prodotti La pagina di presentazione dei prodotti mostra le informazioni principali dei prodotti effettivamente presenti nel vivaio. E' una pagina attiva che si connette alla banca dati esegue la query e presenta i dati sotto forma di tabella. Le voci della tabella sono collegamenti attivi alla pagina di dettaglio.









Pseudocodice

```
<intestazione pagina>
<connessione alla banca dati>
<query>
<riga di intestazione campi>
while(!fine tabella) {
<riga dinamica>
}
<barra di navigazione>
```

Query

```
SELECT
    t1.idspec,
    t1.nome,
    t2.tipo,
    t1.foto,
    t1.espo,
    t1.intext,
    t1.stagione,
    t2.qt,
    t2.costo
FROM
    specie AS t1,
    piante AS t2
WHERE
```

t1.idspec=t2.idspec

FLOWERNET Prodotti						
nome/tipo	foto.	esp.	posiz.	fior.	disp.	prezzo
geranio (piccolo)		sole	esterno	estate	5	1.00
geranio (medio)		sole	esterno	estate	10	2.00
geranio (grande)		sole	esterno	estate	1	3.00
giglio (unico)		penombra	interno	primavera	99	5.00
tiglio (piccolo)		sole	esterno	primavera	1	9.99
tiglio (medio)		sole	esterno	primavera	9	19.99
tiglio (grande)		sole	esterno	primavera	0	99.99
ciclamino (unico)		ombra	interno	inverno	101	0.75

Home Prodotti Attività Prenotazioni

Pagina di dettaglio di un prodotto La pagina di dettaglio di un prodotto mostra tutte le informazioni di singolo prodotto selezionato dalla pagina di presentazione. E' una pagina attiva che si connette alla banca dati esegue la query e presenta i dati sotto forma di tabella.

Parametri di ingresso:

\$ispec = codice della specie di pianta

\$tipo = classe di prezzo della pianta

Pseudocodice

```
<intestazione pagina>  
<connessione alla banca dati>  
<query>  
<estrazione riga>  
<dati dinamici>  
}  
<barra di navigazione>
```

Query

```
SELECT  
    t1.idspec,  
    t1.nome,  
    t2.tipo,  
    t1.foto,  
    t1.descriz,  
    t1.info,  
    t1.espo,  
    t1.intext,  
    t1.stagione,  
    t2.qt,  
    t2.costo,  
    t3.nome,  
    t3.cognome  
FROM  
    specie AS t1,
```

**piante AS t2,
personale AS t3**

WHERE

t1.idspec=t2.idspec

AND

t1.idpers=t3.idpers


AND

t1.idspec=\$idspec

AND

t2.tipo='\$tipo'

FLOWERNET
geranio piccolo

<i>descrizione</i>	Geranio da giardino
<i>foto</i>	
<i>informazioni</i>	Geranio da giardino Il comune geranio da giardino, classificato nel genere Pelargonium, preferisce i climi caldi e secchi, in cui può raggiungere le dimensioni di un cespuglio. Ne esistono diverse specie, dai fiori a lobi singoli o doppi e dalle svariate tonalità, principalmente rosa, rosse e bianche. Fiorisce in estate
<i>esposizione</i>	sole
<i>disposizione</i>	esterno
<i>fioritura</i>	estate
<i>disponibilità</i>	5
<i>prezzo €</i>	1.00
<i>referente</i>	Mario Bianchi

[Home](#) [Prodotti](#) [Attività](#) [Prenotazioni](#)

Pagina di presentazione delle attività La pagina di presentazione dei prodotti mostra le informazioni principali dei delle attività disponibili. E' una pagina attiva che si connette alla banca dati esegue la query e presenta i dati sotto forma di tabella. Le voci della tabella sono collegamenti attivi alla pagina di dettaglio.

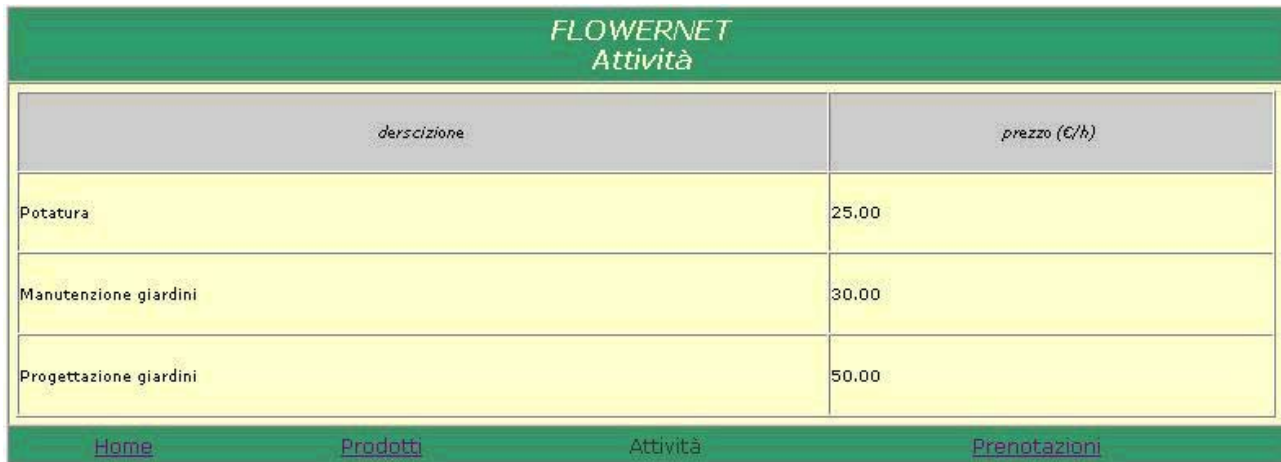
Pseudocodice

```
<intestazione pagina>  
<connessione alla banca dati>  
<query>  
<riga di intestazione campi>  
while(!fine tabella) {  
<riga dinamica>  
}  
<barra di navigazione>
```

Query

```
SELECT  
t1.descriz,
```


t1.costo
FROM attività AS t1



descrizione	prezzo (€/h)
Potatura	25.00
Manutenzione giardini	30.00
Progettazione giardini	50.00

2. Si vuole consentire la gestione delle attività esterne del vivaio attraverso una prenotazione on-line da parte dei clienti. Illustrare le modalità di realizzazione di questa funzione e gli strumenti tecnici adottabili.

Pagina di prenotazione Una soluzione completa per questa pagina attiva è particolarmente complessa perchè diversamente dai casi precedenti in questo caso la comunicazione è bidirezionale. Una soluzione semplice presuppone che le prenotazioni possano essere fatte solo dai clienti già registrati e che la registrazione dei clienti sia fatta per via non telematica oppure per via telematica autenticata. In questo caso la pagina presenta solo due caselle a discesa per selezionare il cliente da una lista e l'attività da una seconda lista, la data di prenotazione viene automaticamente impostata all'ora locale. La pagina è sviluppata sotto forma di un automa a stati finiti. Contiene una form che lancia la pagina stessa con lo stato cambiato. Nello stato "S" (Select) la pagina mostra i risultati di una select sulle tabelle "clienti" e "attività"; nello stato "I" (Insert) la pagina inserisce un nuovo record nella tabella "dettatt".

Pseudocodice

```
<intestazione pagina>  
<connessione alla banca dati>  
<query clienti>  
<modulo select>  
while(!fine tabella) {  
<option dinamica>  
}  
<fine modulo>  
<query attività>  
<modulo select>  
while(!fine tabella) {  
<option dinamica>  
}  
<fine modulo>  
<estrazione data>  
<barra di navigazione>
```

Query

```
INSERT INTO dettatt(idcl,idatt,datap,piante) VALUES($idcl,$idatt,$data,'no')
```

SELECT t1.idcl,t1.denom,t1.telefono,t.tipo FROM clienti AS t1

FLOWERNET Prenotazione	
denominazione cliente	Mario Rossi - 051223344 - privato
descrizione attività	Potatura - 25.00 €/h
data	22/06/2003
conferma	<input type="button" value="Invia"/> <input type="button" value="Reimposta"/>

[Home](#) [Prodotti](#) [Attività](#) [Prenotazioni](#)

SELECT t1.idatt, t1.descriz, t1.costo FROM attivita AS t1

DESCRIZIONE DELLA PIATTAFORMA OPERATIVA

Il principio di funzionamento si basa sull'impiego di protocolli di comunicazione standard. Si parte da un normale browser (Internet Explorer, Netscape ...) in esecuzione su un computer che viene quindi definito "client". Inserendo l'indirizzo di una pagina web (URL) il browser chiede la pagina all'host specificato (RICHIESTA HTTP).

Nell'host deve essere in esecuzione un web server (apache, IIS ...) che è un programma che riceve le richieste dal client e le processa.

Se la richiesta si riferisce ad una pagina statica (un documento in formato html) il web server lo preleva dalla cartella specificata (RICERCA DOCUMENTO) e lo inoltra al client (RISPOSTA HTTP) allegando eventuali files non di puro testo specificati nelle tag html (immagini, applet, suoni ...). E' compito del client trasformare le tag contenute nel documento in presentazioni nella finestra del browser (PRESENTAZIONE) eventualmente elaborando le eventuali tag di script client side (JAVASCRIPT).

Se invece la pagina richiesta è attiva (il web server riconosce il tipo di pagina dall'estensione, nel nostro caso "php" ma esistono anche molte altre estensioni riconosciute come "asp", "jsp" ...) il web server inoltra la pagina (DOCUMENTO DINAMICO) ad un altro programma contemporaneamente in esecuzione sull'host.

Questo programma, genericamente chiamato "script interpreter" ,(nel nostro caso php ma esiste anche asp, jsp ...) legge ed interpreta le tag contenute nel documento, manda in uscita, indietro (OUTPUT HTML) verso il web server, inalterate tutte le tag html mentre esegue le istruzioni contenute all'interno delle tag di script che lo riguardano.

Tutto ciò che sta dentro a queste tag viene interpretato come un programma dall'interpreter che esegue sequenzialmente come una CPU le istruzioni che incontra. Le istruzioni possono fare una ampia gamma di elaborazioni in stile "C" ma soprattutto possono produrre risultati in uscita con opportune istruzioni di output.

Le uscite prodotte dalla elaborazione, che normalmente sono in forma di tag html (OUTPUT HTML) tornano verso il web server che si vede tornare la pagina trasformata da programma a risultato della elaborazione del programma (cioè una normale pagina html).

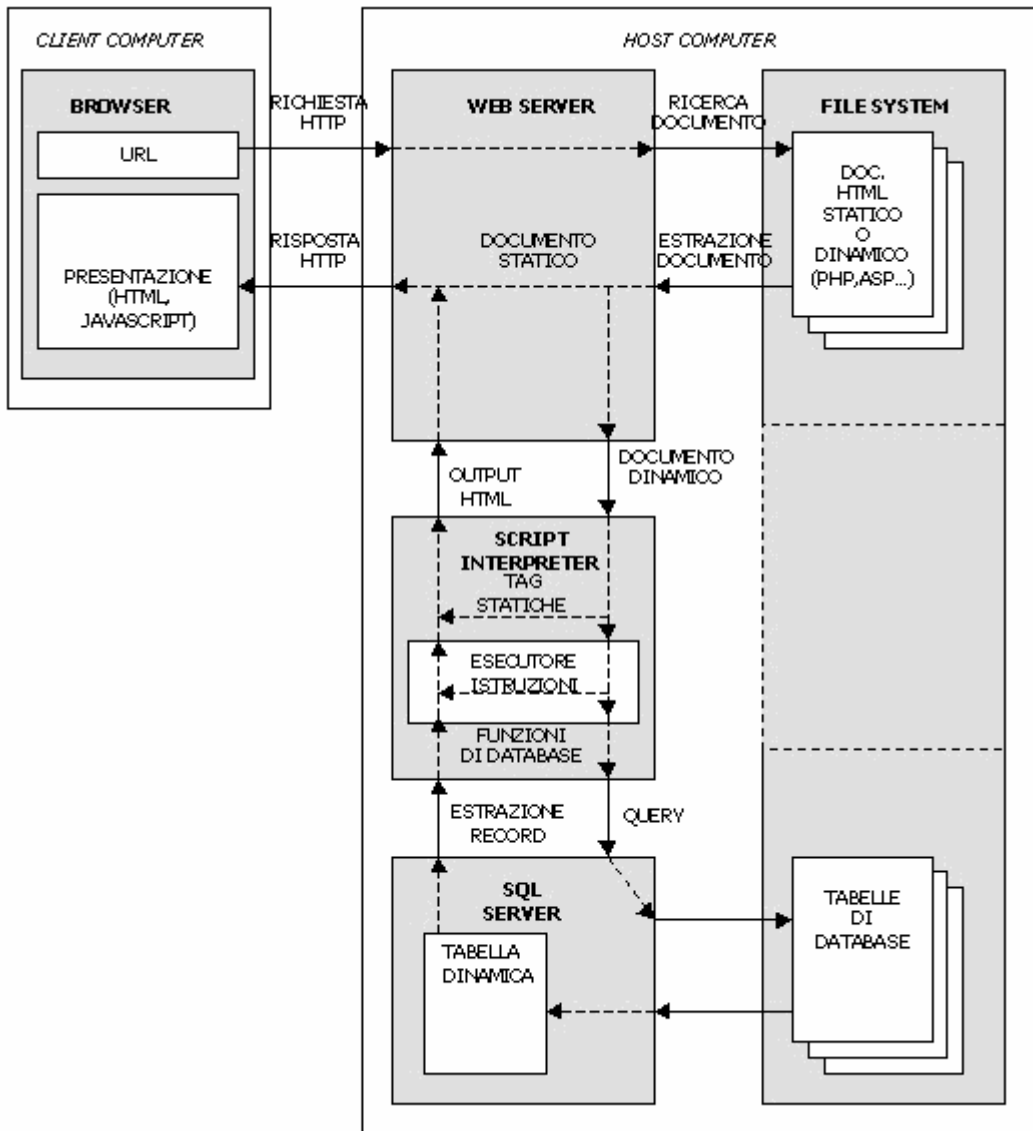
A questo punto il web server inoltra la pagina elaborata al client che la presenta nella finestra del browser (PRESENTAZIONE). Eventuali tag di script client-side (JAVASCRIPT) passano inalterate in questo processo perché devono essere elaborate dal client.

Il motivo principale per cui ha senso usare il web attivo è la necessità di estrarre dati "vivi" (cioè variabili nel tempo) da una banca dati.

Una banca dati è un insieme di archivi organizzati sotto forma di tabelle e gestiti da un programma di interrogazione detto sql server.

Per potere accedere ai dati di una banca dati deve essere in esecuzione sull'host il programma sql server (nel nostro caso mySql). Lo script interpretato dallo script interpreter può contenere l'invocazione funzioni che chiedono dati al sql server (FUNZIONI DI DATABASE). Queste

funzioni mandano una richiesta al programma sql server detta "QUERY"; il sql server risponde estraendo i dati richiesti dalla banca dati residente nel file system secondo i criteri specificati nella query e rendendo disponibile l'accesso ad una tabella generata "al volo" che contiene il risultato della richiesta (TABELLA DINAMICA). Questa tabella è vista dallo script come un file sequenziale in cui ogni record è costituito da una array di stringhe; quindi lo script scorrendo la tabella può estrarre le informazioni richieste (ESTRAZIONE RECORD) e formattarle per l'uscita



verso il web server (OUTPUT HTML).

L'ambiente operativo come si vede dalla descrizione di principio è costituito da tre componenti lato server (oltre al client che però è un browser standard).

- Web server
- Script interpreter
- Sql server

Nell'ipotesi di operare su una piattaforma "Open source" (Software sviluppato in progetti aperti di cui sono gratuitamente disponibili sia gli eseguibili che i sorgenti) i tre componenti possono essere rispettivamente:

- Apache
- Php
- Mysql

Tutti questi componenti operano sia sulla piattaforma linux che sulla piattaforma win32. L'architettura hardware non viene sviluppata perchè non è richiesta dalle specifiche.