

**UNIVERSITÀ  
DEGLI STUDI  
DI PADOVA**

**CORSO DI LAUREA IN INGEGNERIA INFORMATICA**

RELAZIONE DI TIROCINIO

PRESSO L'INTENER SERVICE PROVIDER WORK.NET s.r.l

“Aggiornamento sito istituzionale e creazione di un servizio di controllo di gestione/budget, fruibile via web, per piccole aziende”

RELATORE: Professore Sergio Congiu

TIROCINANTE: Cristina Zanella, matricola 595903-IF

## Indice Generale

Introduzione.....	3
1. Presentazione.....	4
1.1 L'azienda.....	4
1.2 Il progetto.....	6
1.3 Terminologia.....	7
2. Linguaggi ed applicativi utilizzati.....	9
2.1 Linguaggio SQL.....	9
2.2 Linguaggio php.....	12
2.3 Database MySQL/PostgreSQL.....	14
3. Realizzazione del progetto.....	18
3.1 Sito istituzionale.....	18
3.2 Servizio di controllo di gestione/budget.....	20
3.2.1 Interfacce di inserimento.....	23
3.2.2 Elaborazioni.....	25
3.2.3 Stampe a video.....	26
4. Conclusioni.....	29

# Introduzione

Questa tesi nasce da un'esperienza di stage presso un Internet Service Provider nel periodo che va dal 01/08/2012 al 30/09/2012.

Si è scelto di svolgere uno stage per curiosità verso il mondo del lavoro, ma soprattutto per avere una conoscenza di come i concetti imparati durante i corsi all'università si presentano e si integrano nel mondo del lavoro. Inoltre, ritengo che un'attività di tirocinio sia utile anche per capire come cambiano l'ambiente, i ritmi e la mole di lavoro tra il periodo di studio e quello lavorativo.

L'obiettivo dell'attività di tirocinio è stato principalmente la realizzazione di un servizio in internet che permettesse agli utenti di prevedere i costi dei loro progetti basandosi su dati reali e conosciuti da loro inseriti precedentemente in un database; secondariamente si è dovuto aggiornare anche il sito istituzionale del soggetto ospitante contenente anche un collegamento al servizio di gestione/budget.

Per capire meglio quanto realizzato, nel primo capitolo, oltre alla presentazione dell'azienda, vi sarà anche la spiegazione delle caratteristiche e delle richieste riguardanti il progetto di stage; per favorire la comprensione di questa relazione sul lavoro svolto, il capitolo si conclude con la spiegazione dei termini tecnici presenti nel testo.

Per la realizzazione sono stati utilizzati linguaggi e database diversi: i linguaggi php e SQL ed i database mySQL e postgresQL; le rispettive caratteristiche principali ed i punti in cui sono stati utilizzati saranno presentati nel secondo capitolo della tesi.

Questi due capitoli di presentazione, saranno seguiti dal terzo capitolo interamente dedicato alla descrizione della progettazione e della realizzazione del progetto svolto, anche con l'ausilio di immagini e diagrammi di flusso per rappresentare le parti più elaborate e complesse che portano ai risultati voluti.

Il capitolo conclusivo invece riporterà il risultato finale del progetto e le impressioni finali sull'attività del tirocinio.

Si osserva che, dei concetti riportati, ci si è limitati spesso a fornire una spiegazione generale evidenziandone le caratteristiche salienti e quindi più importanti, mentre per i processi descritti si è scelto di non entrare nel dettaglio poiché in molti casi, come ad esempio nelle parti relative a ciò che nel seguito verrà denominato 'elaborazioni', si sarebbe dovuto affrontare e descrivere i dettagli di un'operazione economica, sconfinando in tematiche che non competono a questa relazione.

# 1. Presentazione

## 1.1 L'azienda

WORK.NET s.r.l. è un Internet Service Provider (ISP) dal 1995 con sede nella città di Belluno. È una piccola società, composta da una manciata di soci, che fornisce una svariata gamma di servizi differenti ai suoi clienti, garantendo sempre grande professionalità e serietà nello svolgimento del lavoro richiesto.

I principali servizi a disposizione sono:

- **connettività a Internet:** fornisce connettività ai Clienti, consigliando sempre soluzioni ottimali tra prezzo e banda usata, in funzione della necessità e delle esigenze anche tenendo conto della dislocazione geografica dell'azienda nel territorio, e sue eventuali filiali operative. I tipi di connessione disponibili sono: Dial-Up PSTN e ISDN – ADSL – HDSL – LINK Satellitare bidirezionale – circuito diretto cdn – atm o altro.
- **connettività satellitare:** connettività bidirezionale che permette l'interconnessione fra sedi centrali e periferiche distribuite capillarmente sul territorio. Supporta tutte le applicazioni accessibili da una rete tcp/ip tradizionale, incluse quelle di tipo transazionale (es. le applicazioni gestionali), e quelle 'real-time' (es. le videoconferenze e la telefonia), oltre alle applicazioni web.
- **consulenza e servizi per la sicurezza su Internet:** servizi di antivirus e di firewalling che offrono una protezione efficace per tutte le aziende che si collegano ad Internet, per i loro dati e la loro intera infrastruttura IT. Il Firewall Linux fornisce un massimo livello di sicurezza, monitorizzando, verificando, analizzando e registrando il data transfer attivo sulla rete. L'azienda, inoltre, aggiorna di continuo il firewall operandosi ad applicare le versioni e gli upgrade del software e delle configurazioni del firewalls.
- **Progettazione, realizzazione server postali per la gestione interna delle e-mail aziendali:** i Mail Server realizzati gestiscono in modo affidabile la ricezione, distribuzione e invio della posta elettronica, centralizzata nel Server Postale all'interno dell'azienda, con la possibilità di creare e gestire le Mailing-List sia interne che esterne. Il Mail Server supporta anche l'amministrazione di domini virtuali, per la personalizzazione della posta, anche con interfaccia web-based che semplifica sia la configurazione che la continua manutenzione del servizio. Estrema scalabilità del Mail Server che permette praticamente un numero illimitato di utenti.
- **Voice over IP – VPN:** servizio che permette su collegamenti xDSL, ISDN e PSTN di effettuare telefonate sia interurbane che intercontinentali attraverso la rete internet abbattendo considerevolmente i costi o di effettuare collegamenti in rete privata virtuale a maggiore garanzia delle informazioni trasmesse attraverso la rete.
- **Progettazione, installazione e management di reti anche wireless:** l'azienda, anche attraverso i suoi collaboratori, muniti di regolare autorizzazione ministeriale, progetta, realizza, installa e certifica reti aziendali di categoria 5 e superiore.
- **Management remoto dei servizi forniti:** configurazioni, aggiornamenti e manutenzioni

gestiti direttamente dall'azienda con collegamenti remoti per tutti i servizi forniti, in modo totalmente trasparente per il Cliente.

- **Fornitura apparati per reti e loro accessori:** l'azienda fornisce tutti gli apparati di rete necessari alla realizzazione delle infrastrutture progettate.
- **Progettazione, realizzazione e gestione del Sito Web, analisi sull'utilizzo- remaking:** caratteristica fondamentale dell'azienda è l'estrema flessibilità aziendale, che permette la realizzazione di qualsiasi progetto web, che va dalle cose più semplici, come la realizzazione delle prime pagine web, ad una estrema realizzazione con regia anche complessa (con la collaborazione ad hoc dei propri partners del settore) di siti complessi, con software di animazione, 3D, integrazione con data-base online e quant'altro possa essere richiesto dal Cliente, compreso il più comune remaking del sito già esistente.
- **E-commerce e Portali informativi:** si realizzano sistemi di e-commerce e portali informativi governati da database anche collegati con il sistema informativo interno del cliente per annullare i disallineamenti temporali dei dati che sono evidenti nei sistemi tradizionali che si basano sul trasferimento periodico dei dati.
- **Interfaccia con sistemi bancari per la vendita online:** interfacciamento del portale – negozio-online del Cliente con la gestione bancaria per la transazione di vendita online con carta di credito, anche per portali non direttamente gestiti.
- **Progettazione – realizzazione – manutenzione – assistenza tecnica software:** grazie alla curiosità e la professionalità dei suoi collaboratori, che le ha permesso di acquisire esperienze sempre più ampie e articolate, ha una notevole esperienza nella progettazione e realizzazione di software gestionali, stand-alone e distribuiti (client-server), su ogni tipologia e problematica aziendale. L'azienda si propone, quindi, come partner al fianco dei propri Clienti per il raggiungimento degli obiettivi aziendali, cooperando con il Cliente senza sostituirsi ad esso.
- **Housing – Hosting Dedicato – Hosting Condiviso:** sono servizi flessibili che permettono alle aziende che delegano in outsourcing le loro attività su Internet di usufruire di tecnologie d'avanguardia senza incorrere in oneri di gestione. Le imprese possono beneficiare di numerosi vantaggi, tra cui adottare le nuove tecnologie in tempi rapidi, con la certezza di aver accesso a Know-how specialistico e ridurre i costi IT; hanno anche la garanzia di qualità e obbligazione di risultati, risparmiando così risorse interne all'azienda, potendo concentrarsi sul proprio core-business e sapendo comunque di aver affidato le soluzioni IT ad un partner affidabile e competente.  
I servizi di Housing, Hosting Dedicato e Hosting Condiviso sono completi di servizi di gestione operativa, servizi di gestione sistemistica in ambiente Microsoft, Unix, etc e servizi di database administration su SQL Server, PHP, ASP.
- **Domini di Secondo Livello top level domains tld:** l'azienda è accreditata come provider-maintainer presso i più autorevoli enti di registrazione domini italiani ed esteri. Risolve qualsiasi problematica in questo campo, compresa la nota dolosa del cambio registrar da enti internazionali, quando il caso lo richieda tramite l'Internet Council americano.

## ***1.2 Il progetto***

Il progetto che è stato richiesto di realizzare è composto dall'unione di due sotto progetti differenti sia per modalità di realizzazione sia per difficoltà:

- sito istituzionale
- servizio di controllo di gestione/budget

Il primo, definibile come il più semplice, consisteva nell'aggiornamento del sito istituzionale dell'azienda, facendo fundamentalmente lavoro di valutazione e di web design.

Si è valutata, innanzitutto, la ripartizione dei contenuti e dei collegamenti, per renderli facilmente consultabili, e la scelta dei colori più adatti per rendere il sito serio per l'attività che deve presentare, ma non troppo noioso od impersonale da invogliare l'utente a chiudere la pagina. Si è anche considerato l'inserimento di immagini, come il logo dell'azienda, e quindi le loro eventuali dimensioni così da non farle risultare troppo protagoniste all'interno delle pagine.

La richiesta principale per questa prima parte è stata quella di differenziare i contenuti in base agli utenti visitatori; per questo si sono definite due categorie di utenti: semplici visitatori e utenti registrati.

La parte più complicata ed anche la principale del progetto consiste nella realizzazione di un servizio di controllo di gestione/budget per piccole aziende fruibile attraverso la rete.

Innanzitutto, si è dovuto capire cosa fosse un sistema di controllo di gestione/budget in ambito aziendale e solo in seguito procedere alla realizzazione in base alle richieste effettuate dal tutore aziendale. Queste sono state inizialmente scarse e poco chiare, ma in seguito a varie discussioni sono state precisate e si sono individuati i seguenti obiettivi:

- realizzazione completa del servizio: database e sito
- possibilità per l'utente di inserire le informazioni necessarie
- possibilità per l'utente di visualizzare ciò che ha inserito
- previsioni future sui costi di: centri di costo, articoli, materiali, etc.
- possibilità per l'utente di visualizzare i risultati delle elaborazioni di previsione

Da queste, quindi, si è potuto capire che si sarebbe dovuto andare a manipolare il database stesso sia utilizzando i dati in esso inseriti dall'utente sia andando a modificare i campi già esistenti.

Infine, durante l'analisi del progetto si sono incontrati molti concetti appartenenti al mondo economico: è stato quindi necessario un loro approfondimento per progettare correttamente le procedure che consentono all'utente di inserire i giusti dati per realizzare le previsioni richieste.

I termini suddetti saranno presentati nel paragrafo seguente con i rispettivi significati così da permettere ai lettori di acquisire un minimo di familiarità con i concetti relativi ad un servizio di gestione/budget, mentre l'esecuzione dettagliata del progetto qui presentato sarà esposta nel penultimo capitolo.

### **1.3 Terminologia**

Di seguito saranno riportati i significati dei termini economici utili a capire la parte principale del progetto. In caso il termine abbia più di un significato verrà riportato quello inerente al progetto considerato.

**Budget:** è un documento amministrativo in cui vengono stabiliti in via preventiva gli atti di gestione che l'azienda intende effettuare in un periodo futuro, al fine di conseguire un determinato risultato. È detto anche bilancio di previsione e rientra tra gli strumenti fondamentali di programmazione e controllo dell'azienda. In base alla previsione delle vendite e tenendo conto delle scorte esistenti e dei costi da sostenere (ore uomo, materie prime, servizi esterni, ecc.) espone i ricavi presunti. Di norma si riferisce all'anno contabile (detto: esercizio) successivo a quello in cui viene elaborato.

**Controllo di gestione:** indica le attività di controllo sulla gestione ed il funzionamento di un soggetto pubblico e/o privato. Esso assume particolare rilevanza nell'organizzazione aziendale poiché è volto a guidare la gestione verso il conseguimento degli obiettivi stabiliti in sede di pianificazione aziendale, rilevando, attraverso la misurazione di appositi indicatori, lo scostamento tra obiettivi pianificati e risultati conseguiti e informando di tali scostamenti gli organi responsabili, affinché possano decidere e attuare le opportune azioni correttive.

**Centro di costo:** è un'entità autonoma e omogenea mirata a due finalità: assorbire tutti i costi inerenti ad una data lavorazione o servizio e proiettare questo costo al prodotto finito in proporzione all'uso richiesto al centro stesso. O, più semplicemente, è un segmento aziendale il cui manager ha il controllo sui costi, ma non ha controllo sul fatturato o sull'uso delle risorse destinate agli investimenti.

**Voce di spesa:** termine scelto dal tutore aziendale per racchiudere tutte le spese che l'azienda deve sostenere e che concorrono nel creare il costo finale di un centro di costo (es. elettricità).

**Prodotto o prodotto finito:** In economia è un insieme di attributi tangibili e intangibili volti a procurare un beneficio a un utilizzatore, ottenuto tipicamente attraverso un processo di produzione o creazione a partire da risorse iniziali e con valore aggiunto finale. In marketing, invece, viene definito come tutto ciò che può essere offerto a un mercato per attenzione, acquisizione, uso o consumo, per soddisfare un desiderio o un bisogno. Un prodotto non è quindi solamente l'oggetto fisico in sé (bene), ma include anche servizi, persone, luoghi, organizzazione e idee.

**Materia prima:** sono considerate materie prime tutti quei materiali che sono alla base per la fabbricazione e produzione di altri beni tramite l'utilizzo di opportune lavorazioni e processi industriali che permettono di ottenere il prodotto finale desiderato. Costituiscono in pratica il grezzo originario.

**Semilavorato:** con il termine semilavorato o prodotto intermedio si intende un particolare prodotto che necessita di ulteriori lavorazioni per essere commercializzato come prodotto finale. Per esempio nel campo dell'edilizia il semilavorato è la piastrella.

**Ciclo di lavoro:** è una successione ordinata e razionale di operazioni necessaria per trasformare un prodotto grezzo o una o più materie prime in un prodotto finito. Il ciclo di lavorazione viene solitamente riportato, per le lavorazioni artigianali e industriali del settore meccanico, sul cosiddetto "foglio di lavorazione" o "cartellino di lavorazione", affinché il ciclo sia ripetibile e controllabile. L'aggettivo "razionale" sta ad indicare che tra tutti i modi per costruire un elemento, dovrà essere scelto quello che, data una certa realtà produttiva, risulti il meno costoso.

**Ore uomo o ore mano d'opera:** si riferiscono al personale dei reparti produttivi di un'azienda, detto personale produttivo. Sono tutte quelle ore che un dipendente ha speso per la realizzazione di un lavoro richiesto.

**Ore macchina:** sono tutte quelle ore che un macchinario del reparto produttivo ha effettuato o deve effettuare per la realizzazione di un determinato prodotto.

**Distinta base:** in inglese BOM (Bill of Materials) è l'elenco di tutti i componenti, semilavorati e materie prime necessari per realizzare un prodotto. Una distinta base è organizzata gerarchicamente, e si rappresenta come un albero, con in testa il prodotto finito, ed a scendere nei vari livelli si trovano i semilavorati e le materie prime. Per convenzione la cima dell'albero, ovvero il prodotto finito, si trova a livello zero; i suoi componenti diretti sono a livello uno, e così via.

**Costo:** in economia, management, e contabilità, è un flusso economico uscente associato ad un evento commerciale o ad una transazione economica. Per un bene indica in particolare quanto denaro è servito per produrre tale bene. La determinazione del costo può avvenire attraverso numerose modalità e tenendo conto di più variabili. Il costo di un bene, inteso come costo di produzione del bene, è quindi distinto dal prezzo che rappresenta invece il valore di mercato del bene. I costi in un'azienda si dividono in costi fissi, ossia quelli di cui si conosce l'entità a priori, e costi diretti, cioè quei costi che variano al variare della produzione.

## 2. Linguaggi ed applicativi utilizzati

In questo capitolo saranno presentati i linguaggi e gli applicativi utilizzati durante il progetto; ossia i linguaggi sql e php e i database mysql e postgresQL.

Alcune definizioni per capire le descrizioni dei linguaggi:

- **linguaggio standardizzato:** le norme tecniche o standard sono documenti tecnici che contengono specifiche tecniche di applicazione volontaria, sono elaborate attraverso il consenso delle parti interessate (produttori, pubblica amministrazione, utenti e consumatori, centri di ricerca e laboratori, collegi e ordini professionali) e sono basate sui risultati dell'esperienza e dello sviluppo tecnologico. Inoltre, esse sono approvate da un organismo regionale, nazionale, sovranazionale o internazionale di normazione riconosciuto e sono disponibili al pubblico.
- **modello relazionale:** è un modello logico di rappresentazione o strutturazione dei dati di un database implementato su *sistemi di gestione di basi di dati* (DBMS), detti perciò *sistemi di gestione di basi di dati relazionali* (RDBMS). Si basa sulla teoria degli insiemi ed è strutturato intorno al concetto matematico di relazione, chiamata più spesso tabella. Il concetto fondamentale di questo modello è che i tutti i dati possono essere rappresentati come relazioni e manipolati con gli operatori dell'algebra relazionale (unione, prodotto cartesiano, etc.)
- **linguaggio di programmazione interpretato:** in informatica e programmazione, un interprete è un programma in grado di eseguire altri programmi a partire direttamente dal relativo codice sorgente. Un interprete ha lo scopo di eseguire un programma in un linguaggio di alto livello, senza la previa compilazione dello stesso, cioè di eseguire le istruzioni nel linguaggio usato, traducendole di volta in volta in istruzioni in linguaggio macchina.
- **wrapper:** in informatica, e in particolare in programmazione, un **wrapper** è un modulo software che ne "riveste" un altro, ovvero che funziona da tramite fra i propri clienti (che usano l'interfaccia del wrapper) e il modulo rivestito (che svolge effettivamente i servizi richiesti, su delega dell'oggetto wrapper).
- **sessione:** in informatica e telecomunicazioni, il termine sessione indica il colloquio tra utente ed elaboratore su un determinato argomento e l'attività svolta tra due S\_entità per trasferire dati in entrambi i sensi per tutta la durata del collegamento.

### 2.1 Linguaggio sql

Lo Structured Query Language, o più semplicemente SQL, è un linguaggio standardizzato per i database basati sul modello relazionale (RDBMS).

La prima versione fu sviluppata da IBM all'inizio degli anni Settanta ed era progettata per manipolare dati memorizzati nel database relazionale ideato e brevettato da IBM; venne standardizzato dall'ANSI nel 1986.

Negli anni successivi vennero rilasciate altre versioni, riconosciute anche dall'ISO (Organizzazione internazionale per la normazione o, in inglese, International Organization for Standardization) ed ad

oggi la maggior parte dei sistemi per la gestione di database implementano questi standard ed aggiungono funzionalità aggiuntive proprietarie.

La maggior parte delle implementazioni dispongono di interfaccia alla riga di comando per l'esecuzione diretta di comandi, in alternativa alla sola interfaccia grafica GUI.

Con SQL si leggono, modificano, cancellano dati, tramite l'utilizzo di particolari costrutti di programmazione chiamati query, e si esercitano funzioni gestionali ed amministrative sul sistema dei database.

Non si tratta, quindi, solo di un semplice query language, includendo anche gli altri linguaggi tipici per la progettazione e gestione/amministrazione del database.

Il linguaggio si divide in:

- **Data Definition Language (DDL):** permette di creare e cancellare database o di modificarne la struttura
- **Data Manipulation Language (DML):** permette di inserire, cancellare, modificare i dati
- **Data Control Language (DCL):** permette di gestire gli utenti e i permessi
- **Query language (QL):** permette di interrogare il database, cioè di leggere i dati.
- **Device Media Control Language (DMCL):** permette di controllare i supporti (memorie di massa) dove vengono memorizzati i dati.

Nel progetto richiesto non sono state utilizzati tutti i linguaggi in cui si divide l'SQL, ma solo quelli di creazione e manipolazione dei dati di un database.

Si riportano alcuni esempi.

*Data Definition Language (DDL):* è stato utilizzato per creare il database e le rispettive tabelle.

```
CREATE DATABASE controllo_gestione;  
USE controllo_gestione;
```

```
CREATE TABLE Distinta_base(  
    artpadre_code varchar(15) NOT NULL,  
    artfiglio_code varchar(15) NOT NULL,  
    qta_dist double(12,4) NOT NULL,  
    distinta_sigla varchar(6) NOT NULL,  
    PRIMARY KEY(artpadre_code,artfiglio_code)  
)ENGINE=InnoDB;
```

La prima immagine riporta i comandi per creare ed utilizzare il database, mentre la seconda è un esempio di realizzazione di tabella con tutti i campi che la compongono e la loro tipologia. VARCHAR indica, per esempio, un campo stringa in un cui è possibile utilizzare anche caratteri diversi dalle lettere, mentre DOUBLE è un campo numerico per numeri con virgola: il 12 indica i numeri prima della virgola, mentre il 4 rappresenta le cifre decimali desiderate.

*Data Manipulation Language (DML)*: di questo linguaggio sono stati utilizzati i comandi di inserimento, modifica e cancellazione dei dati, nonostante ve ne siano molti di più.

Un esempio di inserimento in una tabella (in questo caso denominata Temp5):

```
"INSERT INTO `Temp5` (`temp5_codeart`, `temp5_qta`) VALUES('$articolo','$tot')";
```

Un esempio di modifica di un campo già esistente e con un valore non nullo presente.

```
"UPDATE Anagrafica_CDC SET cdc_manodopera='$totale' WHERE cdc_code='$centro'";
```

La clausola WHERE indica la riga o record, contrassegnata dal campo cdc\_code che ha come valore quello della variabile \$centro, su cui andare a modificare il contenuto del campo cdc\_manodopera. Anagrafica\_CDC è la tabella in cui trovare il campo.

Un esempio di cancellazione di dati da una tabella:

```
"DELETE FROM Temp4 WHERE temp4_sigla='$sig'";
```

In questo caso si eliminano tutti i record della tabella Temp4 dove il campo indicato nella clausola WHERE ha il valore della variabile \$sig.

*Query Language*: di questo linguaggio è stato utilizzato il costrutto classico delle query in tutte le sue varie forme.

```
"SELECT SUM(temp3_temptotmacchina) AS macchina, temp3_cdc  
FROM Temp3  
GROUP BY temp3_cdc";
```

Nella query in figura compaiono la clausola SELECT che indica i campi che la query deve estrarre dalla tabella indicata nella clausola FROM.

La clausola GROUP BY è una opzionale ed è utilizzata per raggruppare i record estratti in base al campo indicato. Al contrario, SUM() è un operatore di cui è dotato SQL e somma tutti i valori presenti nella colonna denominata come il campo indicato tra parentesi.

Nelle query è possibile inserire altre clausole, come WHERE vista precedentemente, oppure indicare più di una tabella con diversi modalità od ancora utilizzare altri operatori.

## 2.2 Linguaggio php



PHP (acronimo ricorsivo di "PHP: Hypertext Preprocessor", originariamente acronimo di "Personal Home Page") è un linguaggio di programmazione interpretato, inizialmente concepito per la programmazione Web, ovvero la realizzazione di pagine web dinamiche.

È nato nel 1994 ad opera del danese Rasmus Lerdorf ed in origine era una raccolta di *script* CGI che permettevano una facile gestione delle pagine personali.

Il pacchetto originario venne in seguito esteso e riscritto dallo stesso Lerdorf in C, aggiungendo funzionalità quali il supporto al database mSQL e prese a chiamarsi **PHP/FI**, dove *FI* sta per *Form Interpreter* (interprete di *form*), prevedendo la possibilità di integrare il codice PHP nel codice HTML in modo da semplificare la realizzazione di pagine dinamiche.

Nel 1998 venne sviluppata la terza versione di PHP (il cui acronimo assunse il significato attuale) riscrivendone il motore. Le caratteristiche chiave della versione erano la straordinaria estensibilità, la connettività ai database e il supporto iniziale per il paradigma a oggetti.

PHP diventò ben presto in grado di competere con i linguaggi a lui simili già esistenti e cominciò ad essere usato su larga scala.

Attualmente è utilizzato principalmente per sviluppare applicazioni web lato server ma può essere usato anche per scrivere script a riga di comando o applicazioni stand-alone con interfaccia grafica. Un esempio di software scritto in PHP è MediaWiki.

PHP riprende per molti versi la sintassi del C e del Perl.

Certi costrutti derivati dal C, come gli operatori fra bit e la gestione di stringhe come array, permettono in alcuni casi di agire a basso livello; tuttavia è fondamentale un linguaggio di alto livello, caratteristica questa rafforzata dall'esistenza delle sue moltissime API.

PHP è in grado di interfacciarsi a innumerevoli database tra cui MySQL, PostgreSQL, Oracle, Firebird, solo per citarne alcuni, e supporta numerose tecnologie, come XML e FTP. Si integra anche con altri linguaggi/piattaforme quali Java e .NET e si può dire che esista un wrapper per ogni libreria esistente. Fornisce un'API specifica per interagire con Apache, nonostante funzioni naturalmente con numerosi altri server web. È ottimamente integrato con il database MySQL, per il quale possiede più di una API.

Per questo motivo esiste un'enorme quantità di script e librerie in PHP, disponibili liberamente su Internet.

Nel progetto in esame il php è stato il linguaggio principale utilizzato venendo allo stesso tempo integrato con il codice HTML, per la grafica, ed il linguaggio SQL presentato nel precedente paragrafo.

Essendo, comunque, un linguaggio di programmazione dotato dei classici costrutti presenti anche negli altri suoi simili non vi è molto da descrivere di particolare. L'unica particolarità non consueta e, per questo, degna di nota all'interno del progetto, è stato l'utilizzo delle sessioni con relativo array.

Le sessioni permettono di avere più utenti contemporaneamente e per questo sono utilizzate nei siti che richiedono agli utenti di essere registrati.

Un esempio di apertura di sessione ed una di chiusura:

```
<?php
session_start();
?>
```

```
<?php
if ($convalida=="logged_out")
{
    session_unset();
    session_destroy();
}
?>
```

I due tag `<?php` e `?>` indicano rispettivamente dove inizia e dove finisce il codice php.

Le sessioni, inoltre, hanno un proprio array costituito da una o più variabili di sessione, chiamato `$_SESSION`, che permette il passaggio di parametri da una pagina all'altra, rimanendo attivo per tutta la durata della sessione stessa.

L'array viene costruito appena l'utente effettua il login al sito.

```
$sig=$_SESSION["sigla"];
```

Il termine tra parentesi indica il nome della variabile di sessione presa in considerazione, mentre il termine `$sig` è la classica definizione delle variabili in php.

Per onore di cronaca, si citano gli altri due array per il passaggio di parametri propri del linguaggio php. Il primo è `$_GET` che viene passato tramite la stringa che compare nella barra di indirizzo del browser, mentre il secondo, `$_POST`, viene passato in background.

## 2.3 Database *mySQL* e *postgreSQL*

### MySQL



Il codice di MySQL venne sviluppato fin dal 1979 dalla ditta TcX ataconsult, poi rinominata MySQL AB, ma è solo dal 1996 che vengono distribuite versioni che supportano la maggior parte della sintassi SQL. MySQL AB è stata rilevata da Sun Microsystems nel 2008, mentre nel 2010 quest'ultima è stata acquisita da Oracle Corporation.

MySQL, definito Oracle MySQL, è un Relational database management system (RDBMS), composto da un client, con interfaccia a riga di comando, e un server entrambi disponibili sia per sistemi Unix che per Windows, anche se prevale un suo utilizzo in ambito Unix.

Possiede delle interfacce per diversi linguaggi, compreso un driver ODBC, due driver Java, un driver per Mono e .NET ed una libreria per python.

MySQL fa parte di pacchetti, come piattaforma LAMP e WAMP, usati per sviluppare in locale siti web.

MySQL, essendo scritto in linguaggio C e C++, è disponibile su molti differenti sistemi operativi tra cui Digital Unix, GNU/Linux, Mac OS X, NetBSD, OpenBSD, OS/2 Warp, Solaris, SunOS, SCO OpenServer, SCO UnixWare, SGI Irix, tutte le versioni di Windows da Windows95 fino a Windows 8 e molti altri. Al contrario, le piattaforme di riferimento sono solo Linux e Solaris. La documentazione di MySQL offre comunque un aiuto per chi volesse provare a compilare il software su qualsiasi sistema operativo discretamente diffuso, con l'utilizzo anche di tool per aumentare la compatibilità.

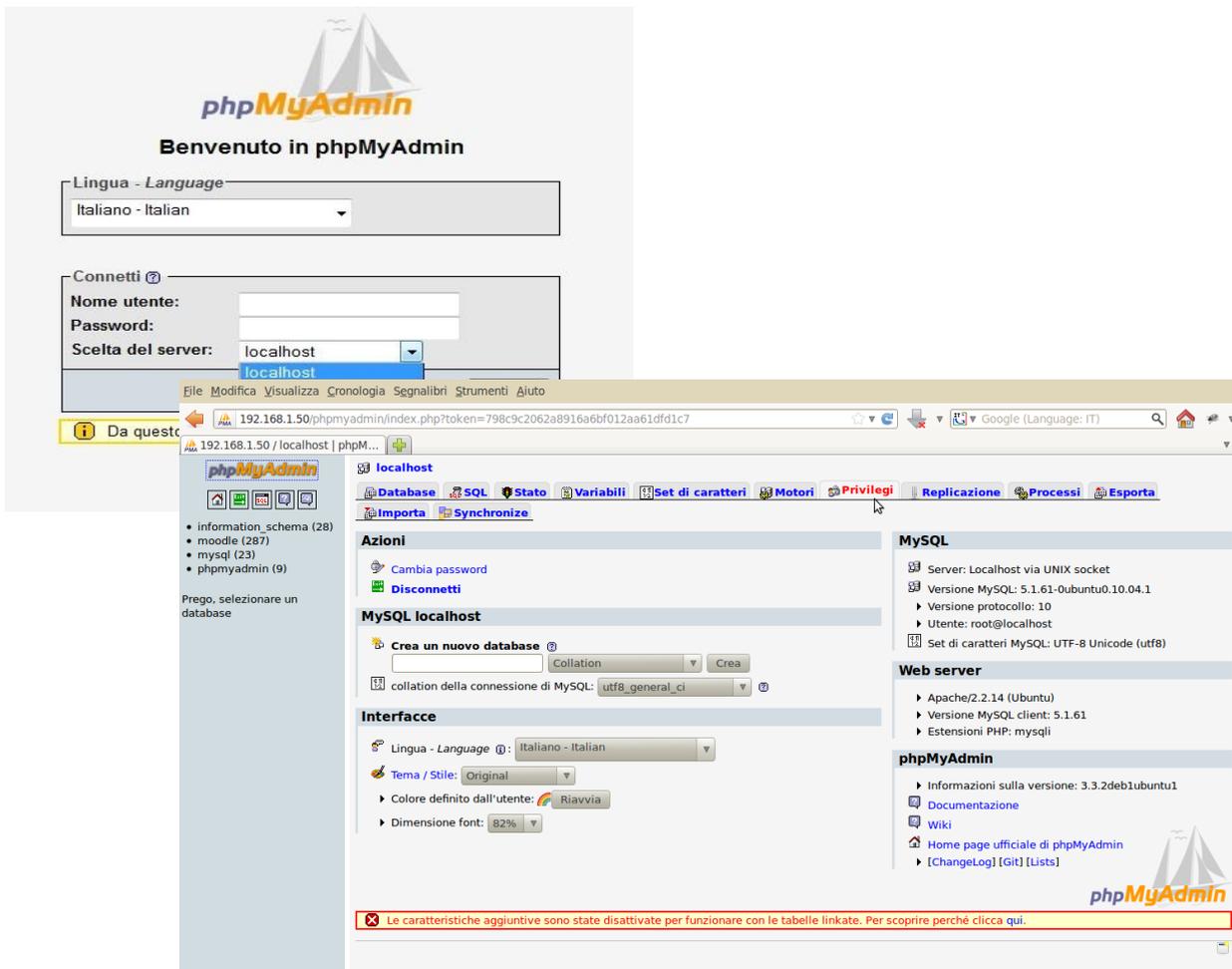
Sono disponibili dei driver per i linguaggi C, C++, C#, Eiffel, Java, Perl, PHP, Python, Ruby, Tcl e per le piattaforme Mono e .Net.

In alternativa alla riga di comando, preferita da utenti/amministratori esperti, esistono diversi gestori grafici per l'amministrazione di MySQL, chiamati MySQL Manager.

Uno dei programmi più popolari per amministrare i database MySQL è phpMyAdmin che richiede un server web come Apache HTTP Server ed il supporto del linguaggio PHP. È utilizzabile facilmente tramite un qualsiasi browser.

Ci sono altri strumenti meno diffusi come per esempio MySQLAdministrator e MySQL Query Browser della MySQLAB. Per il disegno e la modellazione di database MySQL, invece, esiste MySQL Workbench: integra il disegno, la modellazione, la creazione e l'aggiornamento di database in un unico ambiente di lavoro.

Le immagini seguenti sono rispettivamente la schermata di accesso e la homepage, con l'elenco dei database presenti, di phpMyAdmin.



Infine, il linguaggio SQL di MySQL comprende numerose estensioni che sono tipiche di altri DBMS, quali PostgreSQL, Oracle e Sybase; in questo modo le query non standard scritte per altri DBMS in alcuni casi funzioneranno senza problemi.

## PostgreSQL



PostgreSQL è un completo database relazionale ad oggetti rilasciato con licenza libera.

È una reale alternativa sia rispetto ad altri prodotti liberi, come MySQL, che a quelli a codice chiuso, come Oracle, ed offre caratteristiche uniche che lo pongono per alcuni aspetti all'avanguardia nel settore dei database. In più, ha capacità decisamente superiori a quelle dei

concorrenti sia in ambito di grandezza di database supportati sia in ambito di quantità di dati contenuti nelle tabelle.

Nato negli anni Ottanta come progetto dell'università di Berkeley venne poi rielaborato, prendendo il nome di Postgres, per superare i limiti dei concorrenti dell'epoca.

Il nuovo progetto puntava a fornire un supporto completo ai tipi di dati, in particolare la possibilità di definirne di nuovi, ed a dare la possibilità di descrivere la relazione tra le entità (tabelle), che fino ad allora veniva lasciata completamente all'utente. Perciò non solo preservava l'integrità dei dati, ma era in grado di leggere informazioni da tabelle relazionate in modo naturale, seguendo le regole definite dall'utente stesso.

Tuttavia, dopo alcune versioni caratterizzate da varie modifiche e miglioramenti ed anche complete riscritture del sistema di regole, il progetto Postgres venne abbandonato.

Nonostante questo, la licenza libera dava modo agli sviluppatori Open Source di ottenere una copia del software per migliorarlo a loro discrezione. Così nel 1994 due studenti del Berkeley aggiunsero a Postgres un interprete SQL per rimpiazzare il vecchio risalente ai primi anni Ottanta.

Il nuovo software venne quindi rilasciato sul web col nome di Postgres95, ma nel 1996 cambiò nuovamente e, per evidenziare il supporto al linguaggio SQL, venne chiamato PostgreSQL.

Il primo rilascio di PostgreSQL è stata la versione 6.

Da allora, ad occuparsi del progetto è una comunità di sviluppatori volontari provenienti da tutto il mondo che si coordina attraverso Internet. Alla versione 6 ne sono seguite altre, ognuna delle quali ha portato nuovi miglioramenti. Nel gennaio 2005 è stata rilasciata la versione 8, la prima nativa per Windows.

PostgreSQL usa il linguaggio SQL per eseguire delle query sui dati che sono conservati come una serie di tabelle con chiavi esterne che servono a collegare i dati correlati. Tuttavia, il linguaggio SQL rimane inadatto come linguaggio di programmazione, non supportando molti dei principali operatori di base, come le strutture di controllo di ciclo e condizionale; pertanto, ogni venditore ha scritto le sue estensioni al linguaggio SQL per aggiungere queste caratteristiche, ma non è detto che queste operino su diverse piattaforme di database.

Per risolvere il problema, perciò, con PostgreSQL si è data la possibilità ai programmatori di utilizzare un diverso linguaggio di programmazione. I linguaggi disponibili sono svariati e molto differenti tra loro:

- Un linguaggio nativo chiamato **PL/pgSQL** simile al linguaggio procedurale di Oracle **PL/SQL**, che offre particolari vantaggi nelle procedure che fanno uso intensivo di query.
- Wrapper per i più diffusi linguaggi di scripting come **Perl**, **Python**, **Tcl**, e **Ruby** che permettono di utilizzare la loro potenza nella manipolazione delle stringhe e nel link ad estese librerie di funzioni esterne.
- Le procedure che richiedono prestazioni maggiori e logiche di programmazione complesse possono utilizzare il **C** ed il **C++**.
- Inoltre è disponibile anche un interfacciamento all'esoterico linguaggio **R**, ricco di funzioni statistiche e per il calcolo matriciale.

Il programmatore può inserire il codice sul server sotto forma di *funzioni*, che rendono il codice riusabile, in modo che il codice SQL possa richiamare funzioni scritte in altri linguaggi (come il C o

il Perl).

La programmabilità di PostgreSQL è il suo principale punto di forza ed il principale vantaggio verso i suoi concorrenti anche perché rende più semplice costruire applicazioni per il mondo reale, utilizzando i dati prelevati dal database.

In più, contrariamente ai database SQL, che richiedono che sia l'utente a prelevare i dati semplici ed a raggruppare le informazioni correlate tramite query, PostgreSQL, basato sulla programmazione orientata agli oggetti, permette agli utenti di definire nuovi tipi di dato basati su quelli normali di SQL, permettendo al database stesso di comprendere dati complessi. Per esempio, si può definire un indirizzo come un insieme di diverse stringhe di testo per rappresentare il numero civico, la città, ecc. Da qui in poi si possono creare facilmente tabelle che contengono tutti i campi necessari a memorizzare un indirizzo con una sola linea di codice.

PostgreSQL, inoltre, permette l'ereditarietà dei tipi, uno dei principali concetti della programmazione orientata agli oggetti

I punti di forza della programmabilità di PostgreSQL, quindi, sono i seguenti:

- **Incremento delle prestazioni** in quanto la logica viene applicata direttamente dal server di database in una volta, riducendo il passaggio di informazioni tra il client ed il server.
- **Incremento dell'affidabilità** dovuto alla centralizzazione del codice di controllo sul server, non dovendo gestire la sincronizzazione della logica tra molteplici client e i dati memorizzati sul server.
- Inserendo livelli di astrazione dei dati direttamente sul server, il codice del client può essere più snello e semplice.

Questi vantaggi fanno di PostgreSQL, probabilmente, il più avanzato sistema database dal punto di vista della programmabilità.

Nel progetto presentato si è fatto inizialmente uso del database MySQL, utilizzando phpMyAdmin per la sua amministrazione; dopo di che, si è importato il tutto all'interno di PostgreSQL, facendo attenzione a modificare i pochi comandi propri di MySQL in quelli del suo successore.

### 3. Realizzazione del progetto

L'azienda ricercava un modo semplice e veloce per l'analisi dei costi; tuttavia, valutando le opzioni disponibili nella rete si erano trovati programmi di grandi dimensioni che molto spesso operavano calcoli non necessari agli scopi desiderati o le cui licenze avevano un costo troppo elevato. Per questo motivo ed altri propri dell'azienda stessa, si è andati a creare il progetto richiesto, in particolare la seconda parte.

In questo capitolo verrà presentata la realizzazione passo a passo di entrambe le parti di cui esso si compone.

#### 3.1 Sito istituzionale

Come prima parte dell'attività di tirocinio è stato chiesto l'aggiornamento del sito personale dell'azienda; pertanto si sono dapprima analizzati e valutati gli aspetti relativi alla grafica e alla disposizione dei contenuti.

Le caratteristiche che si è scelto di modificare sono state la grandezza delle pagine, i colori, la disposizione di contenuti e collegamenti e le immagini eventuali.

Dopo di che si è lavorato sulla richiesta principale: diversificare i contenuti a seconda degli utenti presenti nel sito.

#### Pagine:

Può capitare, girando per la rete, di incontrare siti in cui le dimensioni delle pagine cambiano a seconda della grandezza dello schermo; ciò rende, a volte, impossibile o difficile osservare la pagina nella sua interezza essendo costretti a muoversi con le barre di scorrimento della finestra.

Per questo motivo, alle pagine si è data una dimensione fissa e non esageratamente grande, così da poterle visualizzare sia su schermi piccoli che grandi.

#### Contenuti e collegamenti:

I contenuti presenti nel sito sono stati forniti dal tutore aziendale e per ognuno di questi si è creato un collegamento diverso; così facendo si è andato a dividere, per esempio, la descrizione dell'azienda da quella dei servizi o delle modalità di contatto.

Le diverse voci, poi, sono state raggruppate nel menù principale al di sotto del logo.

Inoltre, vi è un collegamento a cui si è data maggiore importanza. Si tratta del link che riporta alle offerte del mese, cioè la sezione dedicata ai prodotti che l'azienda mette in vendita.

### Immagini:

Non sono state inserite immagini, poiché non se ne è vista la necessità.

Le uniche eccezioni sono state il logo aziendale, in alto nella prima pagina, e l'eventuale immagine riportante l'oggetto dell'offerta, se presente.

### Colori:

La scelta dei colori per il sito è stata condizionata dal logo dell'azienda, poiché si è scelto di riprenderne i toni principali; si è fatto in modo, però, di non renderli troppo brillanti e, quindi, fastidiosi per gli occhi dell'utente.

Per lo sfondo, invece, si è rimasti sul classico bianco, sempre per non rischiare che le tonalità del logo diventassero delle macchie colorate dalla parvenza casuale.

### Diversificazione degli utenti:

È stato chiesto di diversificare i visitatori del sito; quindi di dividerli in utenti registrati ed utenti occasionali.

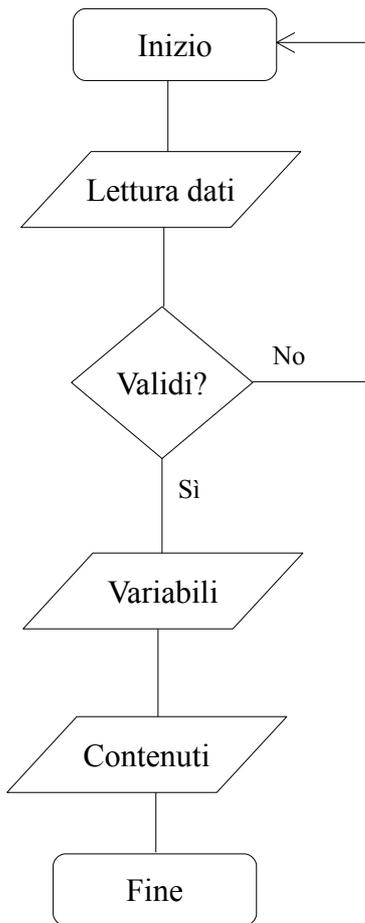
Per fare questo è stata prevista la possibilità agli utenti registrati inseriscano username e password tramite un piccolo form di registrazione.

User  Pass



È stato creato con il linguaggio php poiché, vista la possibilità di avere più utenti contemporaneamente, si sono rese necessarie le sessioni da esso messe a disposizione.

Sono stati inseriti, inoltre, dei controlli sulla validità del dato inviato e, se l'esito è positivo, nelle pagine compariranno anche i contenuti normalmente nascosti.



Leggo i dati dal form

Confronto i dati inviati, se presenti, con i valori nel database. Se non combaciano, torno al form privo di dati.

Se, invece, sono validi memorizzo le variabili di sessione importanti (es. il nome dell'utente che ha effettuato il Login)

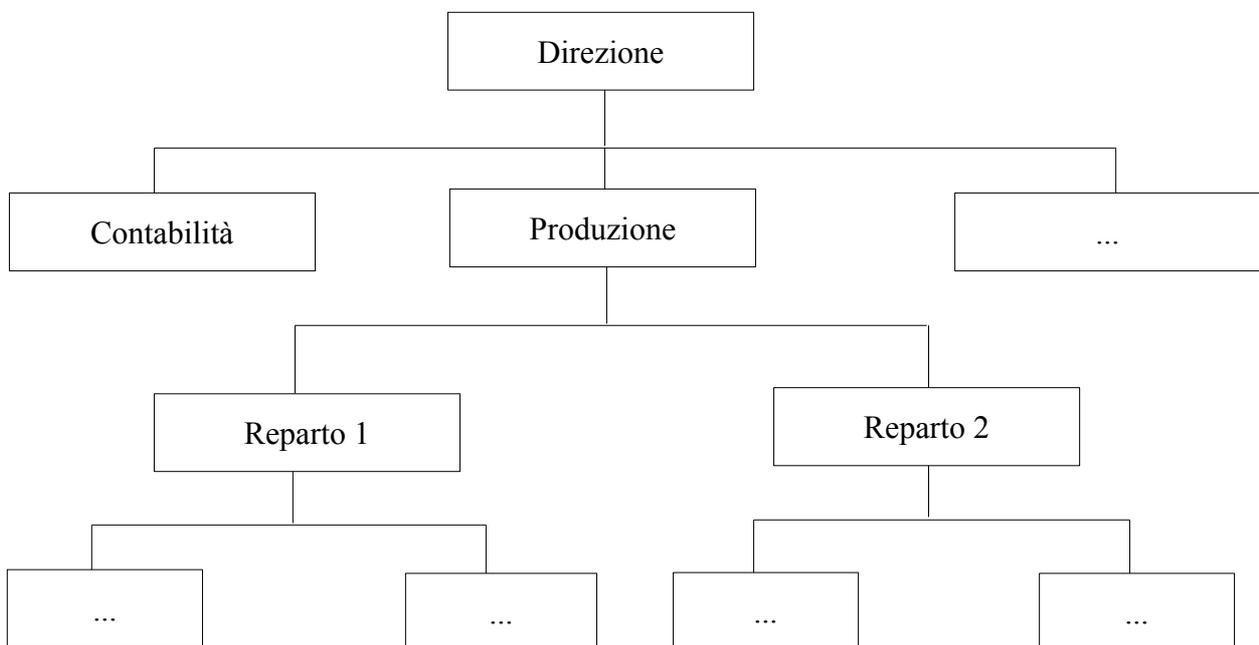
Dopo di che, faccio comparire nelle pagine i contenuti di norma nascosti.

### 3.2 Servizio di controllo di gestione/budget

Prima di affrontare le richieste per la seconda parte del progetto, si è dovuto acquisire la conoscenza di cosa fosse un normale sistema di controllo di gestione/budget e di che dati esso necessiti per essere realizzato.

Basandosi sulla praticità del servizio richiesto, si è potuto capire che è volto alla previsione dei costi futuri di un'azienda così da valutare se rispettano gli obiettivi pianificati o, nel caso li violasse, quali azioni correttive l'azienda dovrebbe andare ad applicare.

Per trovare i dati necessari al calcolo di queste previsioni si è dovuto iniziare dall'azienda stessa; infatti, si è scoperto di doverla suddividere nei suoi centri di costo (cdc) e prendere nota delle relazioni tra questi, ossia disporli in uno schema gerarchico come nell'esempio generico riportato.



Dopo di che, si è dovuto elencare le voci di spesa (vds) presenti e la loro distribuzione nei diversi centri di costo. Per esempio, una voce di spesa potrebbe essere l'elettricità e quindi bisogna andare a valutare quanti chilowatt il centro di costo assorbe.

Sia per cdc sia per vds, però, si è dovuto anche tenere presenti altri dati utili per il calcolo dei costi futuri, come ad esempio le ore uomo e le ore macchina per i reparti di produzione o la quantità di una determinata voce di spesa consumata.

Osservando, poi, che il costo futuro è il risultato di una previsione effettuata su un prodotto si è andati a stilare un elenco di tutti gli articoli che l'azienda produce, distinguendoli tra prodotti finiti (PF), semilavorati (SL) e materie prime (MP). La suddivisione si è resa necessaria, poiché per il servizio richiesto sono fondamentali le distinte base, ossia le relazioni gerarchiche tra i prodotti.

Ciascun tipo di questi, inoltre, è il risultato di una sequenza specifica di lavorazioni e, quindi, si è dovuto tenere conto anche dei cicli di lavoro con i dati salienti utili per il progetto (fase di lavorazione, ore macchine e uomo per fase, etc.). I dati sulle lavorazioni dei prodotti sono utili

anche per previsioni future su altri costi non riguardanti gli articoli.

Infine, osservando i dati necessari e le operazioni future da svolgere, si è dovuto suddividere i costi presenti in azienda tra costi fissi e costi diretti; i primi sono anche un dato utile, per esempio, per le voci di spesa dove si guarda il costo unitario (es. elettricità presa in KW, il costo unitario è quello per 1 KW), mentre i secondi sono quelli derivati dai calcoli eseguiti sui dati conosciuti.

Dopo aver raccolto i dati necessari, si è potuto iniziare a valutare la creazione del database in MySQL.

Durante la sua realizzazione si è dovuto prestare attenzione a quali campi delle tabelle tenere a valore nullo, perché sede di un valore facoltativo o di un futuro costo diretto, e quali invece dovessero comparire all'utente come campi obbligatori da riempire.

Per esempio:

```
CREATE TABLE Vds_Cdc (  
    cdcvds_code varchar(15) NOT NULL,  
    vdscdc_code varchar(15) NOT NULL,  
    vdscdc_qta double(12,4),  
    vdscdc_assorbimento double(12,4),  
    vdscdc_sigla varchar(6) NOT NULL,  
    PRIMARY KEY(cdcvds_code,vdscdc_code)  
)ENGINE=InnoDB;
```

Nella tabella che associa ad ogni centro di costo le sue voci di spesa si nota che i campi della quantità e dell'assorbimento non sono obbligatori. In questo caso sono opzionali poiché per alcuni reparti di un'azienda, nei quali verrà contato l'assorbimento, la quantità di una voce di spesa potrebbe dipendere dai processi in essi effettuati e quindi calcolata successivamente, mentre per altri sarà l'assorbimento a non poter essere calcolato e quindi verrà segnata la quantità fissata.

I campi, invece, con la dicitura 'NOT NULL' sono quelli che devono obbligatoriamente contenere un dato valido.

Riassumendo, dopo la creazione del codice del database, le informazioni necessarie sono risultate essere:

- tabelle
    - articoli
    - prodotti finiti
    - centri di costo
    - voci di spesa
    - distinte base
    - relazioni tra centri di costo
    - relazioni tra centri di costo e voci di spesa
    - cicli di lavoro
- N.B.** La tabella dei prodotti finiti contiene i prodotti destinati alla vendita, mentre la tabella degli articoli contiene anche semilavorati e materie prime

- dati
  - obbligatori
    - codici di riconoscimento per centri di costo, voci di spesa, articoli, prodotti finiti, distinte base e cicli di lavoro
    - modalità di ribaltamento per i centri di costo
    - unità di misura e costo unitario per le voci di spesa
    - tipologia di articolo per la tabella degli articoli
    - il numero finale da vendere per i prodotti finiti
    - fase di lavorazione, ore macchina e uomo a fase per i cicli di lavoro
    - quantità per le distinte base, ossia quanti articoli dello stesso tipo compongono un determinato prodotto
  - opzionali
    - valore di ribaltamento, ore macchina e uomo, totale dei consumi per i centri di costo
    - livello, costo materiali, costo mano d'opera, costo macchina, costo totale per gli articoli
    - quantità per la tabella che relaziona i centri di costo, cioè quanto di ciò che è a disposizione del cdc padre va assegnato al cdc figlio
    - quantità ed assorbimento per la tabella che assegna le voci di spesa ad ogni centro di costo; l'assorbimento è solitamente assegnato ai reparti di produzione, mentre la quantità è per i reparti dirigenziali

Successivamente, avendo constatato che il database fosse visibile in phpMyAdmin senza errori ed avendo inserito dati di prova, si è andati a creare il sito per effettuare le previsioni richieste, interrogare e modificare il database.

Esso si è andato a realizzare in base alle richieste del tutor aziendale, per il servizio richiesto, i cui punti salienti erano:

- **punto 1:** possibilità per l'utente di inserire le informazioni necessarie nel database
- **punto 2:** previsioni future sui costi di: centri di costo, articoli, materiali, etc.
- **punto 3:** possibilità per l'utente di visualizzare ciò che ha inserito ed i risultati richiesti

### 3.2.1 Interfacce di inserimento

Considerando il punto 1, si è capito di dover dare all'utente la possibilità di accedere al database, per inserire i dati conosciuti, tramite il sito. Pertanto, si è vista la necessità di creare un'interfaccia di inserimento di dati.

La maschera di inserimento scelta è il form.

**Relazione tra centro di costo e voce di spesa**  
I campi contrassegnati con un asterisco (\*) sono obbligatori!

Codice centro di costo*	<input type="text"/>
Voce di spesa collegata*	<input type="text"/>
Quantità	<input type="text"/>
Assorbimento	<input type="text"/>

invia

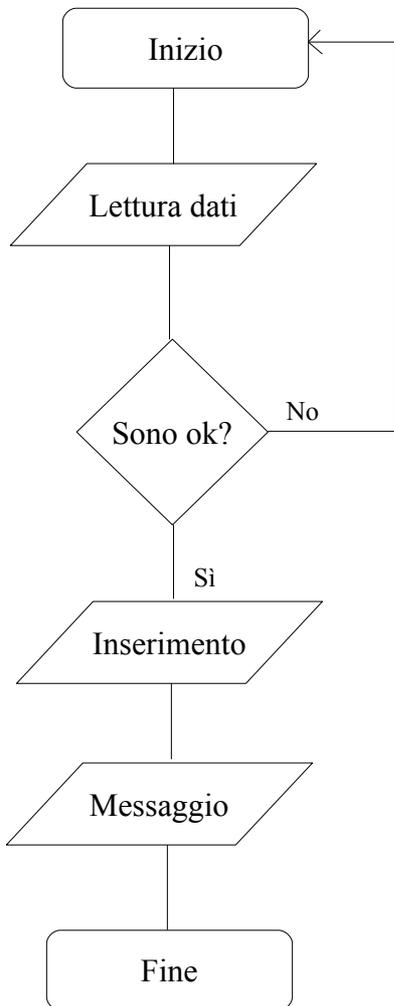
Per la creazione della maschera ci si è basati sulle tabelle del database create in precedenza. Infatti, i dati obbligatori o che potrebbero essere conosciuti, quindi opzionali, corrispondono ad un campo di inserimento del form, come in figura. Contrariamente, i campi che riceveranno dati provenienti dalle elaborazioni future non compaiono nell'interfaccia; questo per non rischiare errori in futuro o sbagli da parte dell'utente.

In più, dove possibile, si è reso più semplice ed intuitivo l'inserimento dei dati con l'ausilio di menù a tendina contenente l'elenco dei nomi dei prodotti inseriti invece dei semplici codici, spesso più difficili da ricordare. Si è deciso di utilizzare i menù per aiutare l'utente a non sbagliare nella digitazione o nella selezione.

Invece, per l'integrità e la coerenza del database sono stati inseriti dei controlli sui dati ricevuti dal form: si è valutato se i campi obbligatori contengono un valore valido e se è già presente nel database un inserimento con lo stesso codice.

Se l'esito dei controlli è positivo i dati saranno inseriti nella tabella apposita e verrà stampata a video la possibilità di inserire un nuovo elemento; al contrario, se il risultato è negativo verrà visualizzato il form stesso senza dati ed un messaggio che riporterà l'errore commesso.

Di seguito è riportato il diagramma di flusso di un inserimento.



Leggo i dati dal form di inserimento

Controllo se ci sono tutti i dati obbligatori e se hanno un valore valido. Se non è così rivisualizzo il form con l'errore commesso.

Se i controlli hanno dato esito positivo, inserisco i dati nella tabella apposita.

Visualizzo a video un messaggio di successo e la possibilità di inserire un altro elemento.

### **3.2.2 Elaborazioni**

Il punto 2 è il fulcro di questa parte del progetto richiesto, poiché descrive lo scopo finale del servizio di controllo di gestione/budget.

Le previsioni sono dei calcoli complessi che intrecciano tra loro dati molto diversi e ricavati da tabelle differenti in modo da raggiungere il risultato finale; le più lunghe e complicate possono richiedere diversi minuti per l'esecuzione. Tuttavia, non si è in grado di svolgerle direttamente come l'inserimento di dati o le stampe a video (descritte successivamente), ma hanno bisogno di svariate operazioni precedenti.

Sia queste ultime sia le previsioni sono state riassunte sotto il nome di elaborazioni, poiché realizzate con le stesse modalità.

Le elaborazioni sono state scritte in modo da essere invisibili all'utente; infatti, necessitano solamente di un link o un tasto, con il nome dell'operazione desiderata, per essere attivate. L'unico segnale che indica all'utente che l'elaborazione è stata eseguita è il messaggio finale di successo.

In queste operazioni, la particolarità principale, è l'uso di tabelle temporanee per il salvataggio dei calcoli intermedi, così da non rischiare di modificare i dati già presenti nelle tabelle e potenzialmente necessari.

Si è scelta la tabella come struttura di supporto ausiliario, al posto per esempio dell'array più flessibile, per la facilità con cui può essere manipolata e per la quantità di dati che può contenere. Inoltre, la struttura e l'interrogazione dell'array diventano sempre più complicate via via che gli si aggiungono dati, mentre la tabella mantiene la sua forma tradizionale e le modalità di richiesta di dati.

Il fatto che la struttura scelta non si modifichi in alcun modo, però, è molto utile per le elaborazioni. Difatti, nonostante siano procedure lineari, cioè che svolgono un'azione solo dopo aver terminato la precedente, in esse si sono dovuti realizzare diversi inserimenti, modifiche, cancellazioni ed interrogazioni, sia su singoli campi sia su intere tabelle, tramite il linguaggio SQL. Queste continue scritture e letture in tabelle sono la causa principale della lentezza delle operazioni.

Anche le strutture di controllo di ciclo, come il while ed il for, sono state usate in grande misura all'interno del codice delle elaborazioni, anche annidati tra loro, per essere certi di arrivare al calcolo finale esatto e per effettuare tutti i controlli necessari; inoltre, sono stati molto utili per lavorare velocemente su una tabella anche di grosse dimensioni.

Per esempio, per il calcolo dei costi di un prodotto si è dovuto ciclare un'intera tabella in modo da esplodere tutta la distinta base dell'articolo desiderato; dopo di che si è dovuto calcolare a ritroso la somma e gli eventuali prodotti dei costi dei singoli componenti del prodotto in esame.

Infine, un'altra proprietà delle elaborazioni è l'ordine: prima vengono svolte quelle che necessitano solo di dati già presenti nel database e, successivamente, saranno effettuate le operazioni che utilizzano anche i risultati provenienti dalle esecuzioni precedenti. Per questo, il sito è anche stato dotato di una piccola guida per l'utente sulla sequenza da seguire.

Con questi processi si possono, quindi, calcolare diversi costi: il costo orario per ogni centro di costo, unito al calcolo delle ore uomo e macchina di ciascuno, oppure il costo in totale del centro stesso. Inoltre, si può prevedere il costo presunto del prodotto finito, contando anche tutte le spese indirette ad esso collegato.

Di seguito viene riportato il diagramma di flusso dell'elaborazione per il calcolo delle ore uomo e

ore macchina di un centro di costo che dà come risultato un dato utile per un'altra elaborazione più complessa; questa è anche una delle più semplici.

### 3.2.3 Stampe a video

Per visualizzare i dati od i risultati richiesti si è scelta la stampa a video ed a seconda di ciò che si è voluto visualizzare si sono distinti due tipi di stampe:

- semplici
- dirette/derivate

Le stampe denominate 'semplici' sono quelle che non contengono calcoli o ne contengono di molto semplici, come somme o prodotti.

Anche queste, a loro volta, si dividono in due filoni:

- quelle che stampano semplicemente il contenuto di una tabella e che non contengono alcun calcolo

Codice Voce	Voce di spesa	Unità di misura	Costo unitario	
<a href="#">VDS1</a>	elettricità	KW	0.1400	X
<a href="#">VDS2</a>	olio lubrificante	l	1.0000	X

da queste si è data la possibilità di modificare gli elementi inseriti, cliccando sul codice, oppure di cancellarlo, cliccando sulla X.

Si è deciso di fornire queste opzioni, poiché vi è sempre la possibilità che l'utente sbagli ad inserire qualche dato o che se lo dimentichi.

- quelle che stampano tutti i dati riguardanti un elemento scelto  
per questo genere di stampa si è usufruito di un piccolo form, composto da un solo campo, da cui si sceglie l'elemento desiderato

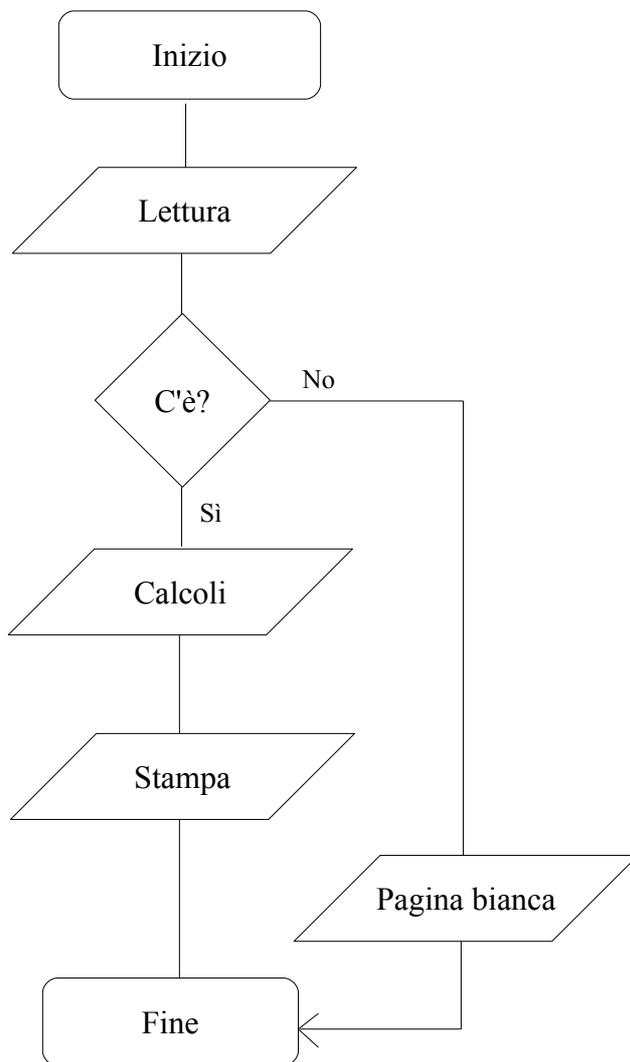
**Riepilogo centri di costo**  
Selezionare la voce e premere invia

Centro di costo

invia

Inviato il dato, se presente, viene estratto quanto desiderato dalle tabelle del database e, con un lavoro di grafica, si visualizza a video uno schema ordinato e leggibile di quanto richiesto.

Viene riportato di seguito il diagramma del funzionamento.



Leggo i dati dal form

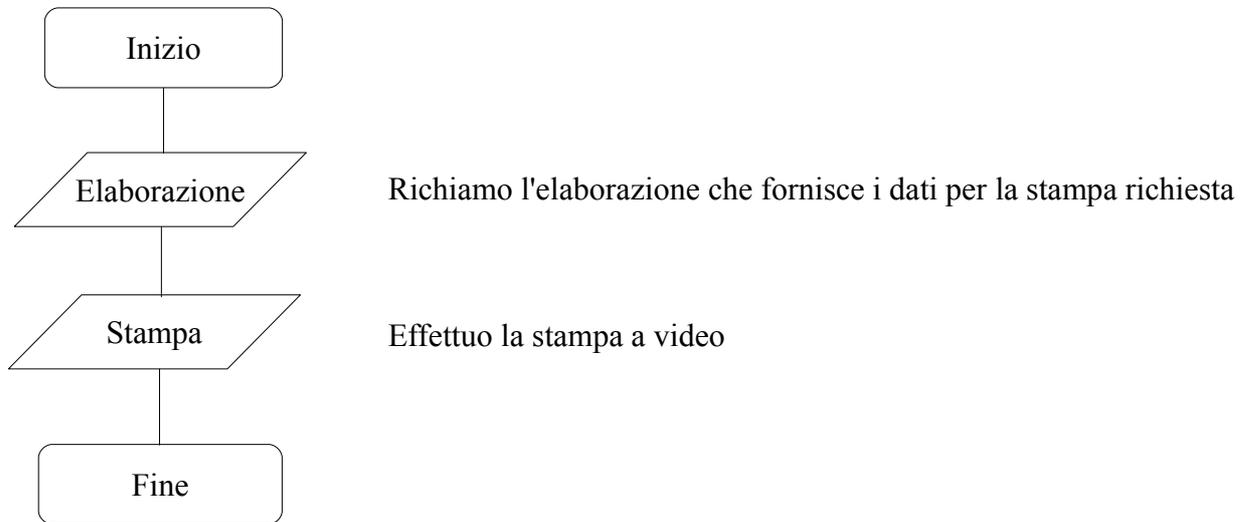
Controllo se il dato è presente

Se il dato è presente svolgo gli eventuali calcoli e mando a video la stampa desiderata.

Se il dato, invece, non viene riscontrato a video comparirà solo una pagina bianca.

Al contrario, le dirette/derivate sono le stampe dei risultati derivati dalle elaborazioni richieste; pertanto, il lavoro più impegnativo è lasciato ad un'altra sezione del servizio di controllo di gestione/budget, mentre la stampa si riduce ad un lavoro di grafica.

Di queste si riporta solamente il diagramma di flusso.



## 4. Conclusioni

Posso sicuramente ritenermi soddisfatta di come si è svolta l'intera attività di tirocinio e dei risultati ottenuti.

Innanzitutto, ho potuto affrontare i problemi pratici che riguardano la realizzazione di un servizio via web, interamente scritto senza l'utilizzo di un ambiente di sviluppo ed, in più, in un linguaggio di cui possedevo solo un'infarinatura basilare.

Ho anche potuto notare il grande utilizzo delle basi di dati in quella parte del mondo del lavoro che opera principalmente via web e ho constatato che sono in alcuni casi il cuore del progetto da realizzare.

Il tutto è stato creato per una precisa realtà aziendale; tuttavia, i concetti presentati possono essere ampliati e sviluppati per altre realtà di interesse decise dall'azienda.