# 📊 Report di Analisi Funzionale e Tecnica

## Sistema Gestione Ordini Clienti - Benza

**Data Analisi:** 24 Dicembre 2024 **Versione Report:** 1.0 - Situazione Iniziale **Analista:** Claude (Senior Software Architect)

---

## Executive Summary

Il progetto è un **sistema CRM/gestione ordini** sviluppato in PHP/MySQL con circa **10+ anni di storia**. L'applicazione gestisce l'intero ciclo di vita degli ordini: dall'acquisizione del cliente, alla gestione delle fasi (stati), fino alla fatturazione.

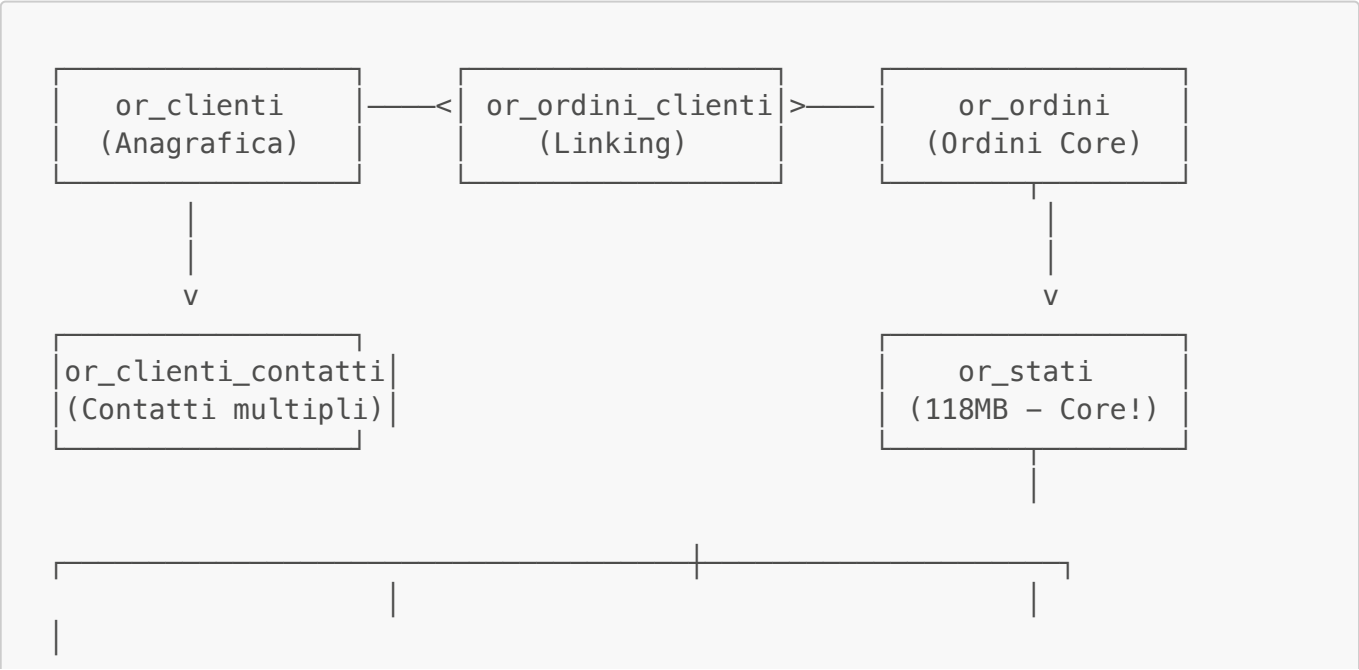
### Numeri Chiave

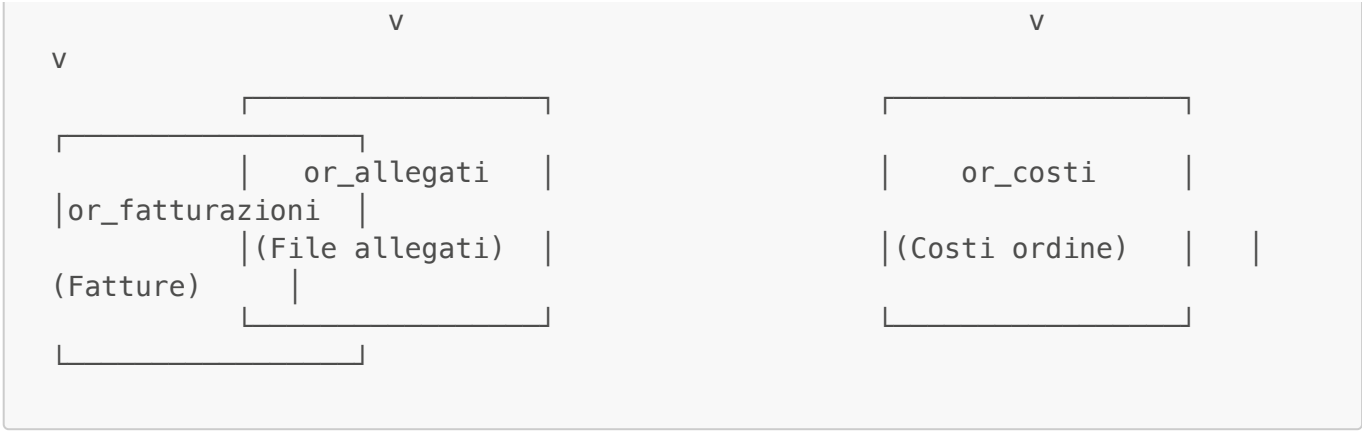
| Metrica | Valore |
|---|---|
| Tabelle DB | 22 |
| File PHP | 117+ |
| Dimensione DB | ~125 MB (di cui 118 MB solo `or_stati`) |
| Charset | Misto Latin1/UTF8 (⚠ problematico) |
| Versione PHP stimata | 5.5+ (usa mysqli) |
| Vulnerabilità critiche | 5+ |

---

## 1. Mappatura del Database (Analisi Critica)

### 1.1 Entità Principali e Relazioni






### 1.2 Catalogo Tabelle

| Tabella | Funzione | Dimensione | Stato |
|---|---|---|---|
| admin | Credenziali admin | 1 record | ⚠ Password plain text |
| or_ordini | Ordini principali | 1.4 MB | ✅ Core |
| or_clienti | Anagrafica clienti | 2.4 MB | ✅ Core |
| or_stati | Cronologia stati ordine | **118 MB** | ✅ Core (più grande!) |
| or_categorie | 11 categorie stati | 11 record | ✅ Lookup |
| or_dipendenti | Anagrafica dipendenti | 18 record | ✅ Lookup |
| or_clienti_contatti | Contatti aggiuntivi | 314 KB | ✅ Usata |
| or_ordini_clienti | Link ordini-clienti | 151 KB | ✅ Linking |
| or_province | Province italiane | 114 record | ✅ Lookup |
| or_pagamenti | Metodi pagamento | 9 record | ✅ Lookup |
| or_costi | Costi per ordine | 258 KB | ✅ Usata |
| or_costi_altri | Costi variabili | 45 record | ✅ Usata |
| or_costi_gen | Costi generali | 42 KB | ✅ Lookup |
| or_fatturazioni | Fatture emesse | 245 KB | ✅ Usata |
| or_allegati | File allegati | 3.3 MB | ✅ Usata |
| log_preventivo | Log accessi | 119 KB | ✅ Analytics |
| or_anni_passati | Archivio storico | 12 anni | ✅ Storico |
| or_progetti_ricevuti | Progetti ricevuti | ? | ⚠ Raramente usata |
| or_progetti_spediti | Progetti spediti | ? | ⚠ Raramente usata |
| or_stati_clienti | Link stati-clienti | minima | ⚠ Solo DELETE |

### 1.3 Analisi del Debito Tecnico DB

### 🔴 Problema Charset (CRITICO)

```
or_fatturazioni → utf8
Tutte le altre   → latin1
```

**Impatto migrazione utf8mb4:**

- 118 MB di dati in `or_stati` da convertire
- Rischio perdita caratteri speciali (€, accenti, emoji)
- Query con collation mista potrebbero fallire
- Necessario backup completo prima della migrazione

### 🔴 Assenza Foreign Keys

Nessun vincolo FK dichiarato nello schema SQL. Le relazioni sono gestite solo a livello applicativo → rischio integrità referenziale.

### 🟡 Colonne "Zombie" (Presenti ma NON Usate nel Codice)

| Tabella | Colonna | Tipo | Ipotesi |
|---|---|---|---|
| `or_stati` | `preventivo` | VARCHAR(128) | Refactoring incompiuto |
| `or_stati` | `inviato` | INT(1) | Feature email disabilitata |
| `or_stati` | `email_spedita` | INT(1) | Feature email disabilitata |
| `or_ordini` | `id_ordine` | INT(11) | E-commerce mai integrato (commento: "e-commerce") |
| `or_fatturazioni` | `numero` | VARCHAR(16) | Sempre vuoto - numerazione non automatica |
| `or_clienti` | `sdi` | VARCHAR(7) | Campo SDI inserito ma mai visualizzato |
| `or_stati` | `foto` | VARCHAR(1024) | Raramente usato |
| `or_stati` | `pdf` | VARCHAR(1024) | Raramente usato |

### 🟡 Nomi Non-Standard

| Tabella | Colonna | Problema | Suggerimento |
|---|---|---|---|
| `or_ordini` | `genap` | Nome incomprensibile | Documentare o rinominare |
| `or_ordini` | `corriere` | Hardcoded "GLS" | Scelta corriere mai implementata |
| `or_clienti` | `s_via, s_cap, s_comune...` | Prefisso "s_" poco chiaro | Usare `shipping_address`, `billing_address` |

| Tabella | Colonna | Problema | Suggerimento |
|---|---|---|---|
| `or_clienti` | `nominativo2` | Ambiguo | Rinominare in `azienda_affiliata` o simile |
| `log_preventivo` | `anno, mese` | INT per date | Usare campo DATE |

# 2. Analisi Funzionale (Cosa fa il Software?)

## 2.1 Panoramica Prodotto

Il software è un **CRM per la gestione del ciclo commerciale** con focus su:

- Tracciamento ordini dalla richiesta alla consegna
- Gestione pipeline vendite (11 fasi/categorie)
- Gestione costi e fatturazione
- Reportistica commerciale

## 2.2 Mappa delle Schermate

```
📱 LOGIN
└── formlogin.php → login.php

📊 DASHBOARD/RICERCA (Home principale)
├── index.php (redirect)
├── ricerca.php (ricerca avanzata)
├── search.php (AJAX quick search)
└── sit_clienti.php (situazione per cliente)

👥 GESTIONE CLIENTI
├── clienti.php (elenco)
├── nuovo_cliente.php → nuovo_cliente_db.php
├── modifica_cliente.php → modifica_cliente_db.php
├── dati_cliente.php (visualizzazione)
└── clienti_contatti.php (contatti multipli)

📦 GESTIONE ORDINI
├── ordine_stati.php ⭐ (CORE - dettaglio ordine)
├── nuovo_ordine.php → nuovo_ordine_db.php
├── modifica_ordine.php
└── ordine_allegati.php

🔄 GESTIONE STATI (Pipeline)
├── nuovo_stato.php → nuovo_stato_db.php
├── modifica_stato.php → modifica_stato_db.php
└── cancella_stato.php

💰 FINANZE
├── da_pagare.php (ordini da saldare)
├── fatturazione.php → fatturazione_db.php
├── riba.php (gestione RiBa)
```

```
├── nuovo_costo.php (costi variabili)
└── stampa_bilancio.php

📈 STATISTICHE
├── stat_generali.php
├── stat_vendite.php
├── stat_dipendenti.php
└── archivio.php (anni passati)
```

### 2.3 Categorie Ordini (Pipeline Commerciale)

| Cod | Categoria | Fase |
|---|---|---|
| 1 | Primo contatto | 🟢 Inizio |
| 2 | Contatto in corso | 🟢 Negoziazione |
| 3 | Richiesta tecnica | 🟡 Analisi |
| 4 | Preventivo inviato | 🟡 Proposta |
| 5 | Ordine | 🔵 Conferma |
| 6 | Produzione | 🔵 Esecuzione |
| 7 | Spedizione | 🔵 Logistica |
| 8 | Venduto | ✅ Chiuso positivo |
| 9 | Problema | 🔴 Issue |
| 10 | Problema risolto | ✅ Issue chiuso |
| 11 | Non venduto | ❌ Chiuso negativo |

### 2.4 Flussi Operativi Principali

**Flusso 1: Acquisizione Cliente**

```
1. Utente cerca cliente esistente (ricerca.php / sit_clienti.php)
2. Se non esiste → nuovo_cliente.php
3. Compila form: ragione sociale, P.IVA, indirizzo, contatti
4. Sistema salva in or_clienti
5. Opzionale: aggiunge contatti multipli (clienti_contatti.php)
```

**Flusso 2: Gestione Ordine (CORE)**

```
1. Nuovo ordine → associa cliente(i) esistente(i)
2. Sistema crea record in or_ordini + or_ordini_clienti
3. Aggiunta stati progressivi:
   - Primo contatto → Preventivo → Ordine → Produzione → Spedizione →
```

```
Venduto
4. Ogni stato: data, descrizione, dipendente responsabile, allegati
5. Tracking completo in ordine_stati.php
```

**Flusso 3: Gestione Costi e Fatturazione**

```
1. Da ordine_stati.php → aggiungi costi
2. Tipi costo: materiali, trasporto, variabili, altri
3. Calcolo automatico IVA per tipo
4. Fatturazione: genera record + allega PDF
5. Gestione RiBa per pagamenti bancari
```

### 2.5 Funzionalità Core vs Accessorie

**⭐ CORE (Usate quotidianamente)**

- Ricerca ordini/clienti
- Gestione stati ordine (nuovo_stato, modifica_stato)
- Visualizzazione cronologia (ordine_stati.php)
- Situazione clienti (sit_clienti.php)

**📌 RILEVANTI (Usate frequentemente)**

- Gestione anagrafica clienti
- Gestione costi ordine
- Fatturazione
- Allegati ordine

**📎 ACCESSORIE (Usate occasionalmente)**

- Statistiche e report
- Log preventivi
- Archivio anni passati
- Gestione RiBa

**⚠ APPARENTEMENTE ABBANDONATE**

- Feature invio email (flag `inviato`, `email_spedita` non usati)
- Integrazione e-commerce (`id_ordine` sempre 0)
- Gestione progetti ricevuti/spediti (tabelle quasi vuote)
- Numerazione fatture automatica (`numero` sempre vuoto)

---

## 3. Architettura e Flusso Dati

### 3.1 Gestione Database

**Tecnologia:** MySQLi Object-Oriented

```
// File: datalog.php
$mysqli = new mysqli($host, $user, $pass, $db_name);
```

**Pattern di connessione:**

- Connessione singleton in `datalog.php`
- Include globale in ogni file
- Nessun connection pooling
- Nessun ORM/Query Builder

**Sicurezza query:**

| Pattern | Presenza | File Esempio |
|---|---|---|
| Prepared Statements | ✅ Parziale | nuovo_stato_db.php |
| Query concatenate | ⚠ MOLTE | ricerca.php, validUser.php |
| Escaping manuale | ⚠ Inconsistente | vari |

### 3.2 Separazione Logica/Presentazione

**Architettura: Spaghetti Code** 🍝

```
❌ NON presente:
- Pattern MVC
- Template engine
- Separazione controller/view
- Routing centralizzato

✅ Presente (parzialmente):
- File _db.php separati per operazioni DB
- header.php / footer.php come include comuni
- funzioni.php per utility condivise
```

**Struttura tipica file:**

```
<?php
include("ok.php");         // Autenticazione
include("header.php");     // HTML header
// ... logica PHP mista a HTML ...
include("footer.php");     // HTML footer
?>
```

### 3.3 Autenticazione e Sessione

**Metodo: Cookie-Based Plain Text** ⚠ INSICURO

```
// login.php
setcookie('authentication', $_POST['user'].'|'.$_POST['pass'], $expires,
'/');

// ok.php (verifica)
if(isset($_COOKIE['authentication'])) {
    $dati = explode('|', $_COOKIE['authentication']);
    validUser($dati[0], $dati[1]);  // Query SQL!
}
```

**Problemi:**

1. Password in plain text nel cookie
2. Nessun hashing (né lato client né server)
3. Cookie con scadenza 2029 (5+ anni)
4. Nessun flag `Secure` o `HttpOnly`
5. Nessun session_start() - tutto via cookie
6. Nessun rate limiting

### 3.4 Analisi JavaScript

**Librerie utilizzate:**

| Libreria | Versione | Stato |
|---|---|---|
| jQuery | 1.9.1 (2013) | ⚠ Obsoleto, vulnerabile |
| jQuery UI | 1.9.2 | ⚠ Obsoleto |
| TinyMCE | 2.4 (2012) | ⚠ Molto obsoleto |
| Select2 | 4.0.4 | ✅ Relativamente recente |
| Shadowbox | ? | ⚠ Abbandonato |

**Logica client-side:**

- Principalmente UI/UX (popup, modal, collapse)
- Validazione form base (validazione.js)
- AJAX per ricerca rapida (header.php → search.php)
- Form submission via POST tradizionale (non SPA)

**Funzionalità JS principali:**

```
// funzioni.js
doPopUp_mod_cliente()      // Apre popup modifica cliente
doPopUp_stampa_ordine()    // Apre popup stampa
```

```
// validazione.js
verifica()                  // Valida campi form spedizione

// Login.js
logInit()                   // Submit login (password in chiaro!)
```

---

## 4. Inventario delle Criticità per il Refactoring

### 4.1 🔴 Vulnerabilità di Sicurezza (CRITICHE)

**SQL Injection**

| File | Linea | Codice Vulnerabile |
|---|---|---|
| `validUser.php` | 16 | `WHERE user = "'.$user_l.'" AND password = "'.$pass_l.'"` |
| `ricerca.php` | 34-56 | 8+ campi LIKE concatenati |
| `funzioni.php` | 178 | `WHERE cod_ordine = ".$cod_ordine` |

**Credenziali Hardcoded**

| File | Contenuto |
|---|---|
| `datalog.php` | User/password DB (root/ap1983, Sql958286/5125o0ijd8) |
| `table_admin.sql` | Admin: info@benza.it / Benza2013! |
| `table_or_dipendenti.sql` | Password dipendenti in chiaro |

**XSS Potenziale**

| File | Rischio |
|---|---|
| `header.php` | `$("#display").html(html)` senza sanitizzazione |
| `ricerca.php` | Output stripslashes senza htmlspecialchars |

**Altre Vulnerabilità**

- **CSRF**: Nessun token nelle form
- **File Upload**: No validazione estensione in allegati
- **Session Fixation**: Cookie senza rigenerazione

### 4.2 🟠 Funzioni/Pattern Deprecati

| Elemento | Stato | Azione |
|---|---|---|
| jQuery 1.9.1 | Obsoleto (2013) | Aggiornare a 3.x |

| Elemento | Stato | Azione |
|---|---|---|
| TinyMCE 2.4 | Obsoleto (2012) | Aggiornare a 6.x |
| Cookie auth | Insicuro | Migrare a session + JWT |
| mysql_ functions | Non presenti ✅ | - |

### 4.3 🟡 Hard-coding e Configurazione

| Tipo | File | Contenuto |
|---|---|---|
| Credenziali DB | datalog.php | Host, user, password |
| Path assoluti | vari | `/Users/paolo/...` (development) |
| Corriere | or_ordini | Sempre "GLS" |
| Email | vari | info@benza.it hardcoded |

### 4.4 📋 Debito Tecnico Complessivo

| Categoria | Severità | Elementi |
|---|---|---|
| Sicurezza | 🔴 CRITICA | SQL Injection, XSS, Auth debole |
| Architettura | 🟠 ALTA | No MVC, spaghetti code |
| Database | 🟠 ALTA | Charset misto, no FK, colonne zombie |
| Dipendenze | 🟡 MEDIA | jQuery/TinyMCE obsoleti |
| Codice | 🟡 MEDIA | Duplicazione, naming inconsistente |

---

## 5. Raccomandazioni per il Refactoring

### Priorità 1: Sicurezza (Immediato)

1. ☐ Implementare Prepared Statements ovunque
2. ☐ Spostare credenziali in file .env
3. ☐ Implementare password hashing (bcrypt)
4. ☐ Migrare a session-based auth
5. ☐ Aggiungere CSRF token

### Priorità 2: Database (Breve termine)

6. ☐ Backup completo
7. ☐ Migrazione charset a utf8mb4
8. ☐ Aggiungere Foreign Keys
9. ☐ Rimuovere/documentare colonne zombie

### Priorità 3: Architettura (Medio termine)

10. ☐ Valutare framework PHP (Laravel/Symfony)
11. ☐ Implementare pattern MVC
12. ☐ Creare layer di accesso dati (Repository)
13. ☐ Aggiornare frontend (Vue/React o Blade/Twig)

### Priorità 4: Modernizzazione (Lungo termine)

14. ☐ API REST per future integrazioni
15. ☐ Test automatizzati
16. ☐ CI/CD pipeline
17. ☐ Containerizzazione (Docker)

---

## 6. Avanzamento Lavori Refactoring

Stato Attuale: 🔴 NON INIZIATO

| Fase | Stato | Data Inizio | Data Fine | Note |
|---|---|---|---|---|
| Analisi iniziale | ✅ Completata | 24/12/2024 | 24/12/2024 | Questo documento |
| Sicurezza critica | ⬜ Da fare | - | - | - |
| Migrazione DB | ⬜ Da fare | - | - | - |
| Refactoring PHP | ⬜ Da fare | - | - | - |
| Modernizzazione UI | ⬜ Da fare | - | - | - |
| Testing | ⬜ Da fare | - | - | - |
| Deploy | ⬜ Da fare | - | - | - |

---

## Appendice A: File Principali del Progetto

```
/vecchio_software/
├── datalog.php          # Connessione DB (credenziali!)
├── ok.php               # Verifica autenticazione
├── validUser.php        # Login validation (SQL Injection!)
├── header.php           # Template header
├── footer.php           # Template footer
├── funzioni.php         # Utility functions
├── ricerca.php          # Ricerca avanzata (SQL Injection!)
├── ordine_stati.php     # ⭐ Core - Dettaglio ordine
├── nuovo_stato_db.php   # Insert stati (usa prepared)
├── modifica_stato_db.php # Update stati
├── sit_clienti.php      # Situazione clienti
├── fatturazione.php     # Gestione fatture
└── /sql_tables/         # Schema DB
    ├── table_admin.sql
    ├── table_or_ordini.sql
    ├── table_or_clienti.sql
```

```
    ├── table_or_stati.sql
    └── ...
```

---

**Documento generato il 24/12/2024 Prossimo aggiornamento: Al termine di ogni fase di refactoring**