



# **EMPLOYEE**

## System Design Document

*Programma Lorenzo  
Bonura Davide  
Campisi Lucio  
Lo Valvo Dario*

## **INDICE**

- Obiettivo del sistema
- Architettura software attuale
- Architettura software proposta
  - Panoramica
  - Requisiti minimi per l'utilizzo del software
  - Decomposizione in sottosistemi
  - Suddivisione degli oggetti all'interno dei sottosistemi e delle componenti
  - Mappatura Hardware/Software
- Gestione dei dati persistenti
  - Modello E-R
  - Modello relazionale
  - Struttura delle tabelle

## **Obiettivo del sistema**

L'obiettivo del sistema è quello di gestire le informazioni degli impiegati che lavorano in un'azienda di servizi al cittadino che offre quattro servizi in maniera gerarchica. Inoltre, il sistema deve gestire le turnazioni, tenendo conto delle varie astensioni concedibili, assegnando a ogni impiegato il proprio turno garantendo che il servizio a priorità più alta resti attivo per più tempo rispetto agli altri servizi.

## **Architettura software attuale**

Supponiamo che non esista nessuna soluzione software usata attualmente per svolgere le funzioni che il sistema proposto può effettuare.

## **Architettura software proposta**

### **Panoramica**

Per realizzare il software è stato scelto di utilizzare l'architettura di tipo Repository. Questo tipo di architettura è caratterizzata dalla suddivisione del software in sottosistemi fra loro indipendenti in quanto interagiscono con una singola struttura dati ovvero la repository che rappresenta il sottosistema di storage. Ogni sottosistema è formato da un'interfaccia utente che comunica con il controllore sottostante, che contiene la logica del programma e che si occupa di gestire le comunicazioni con il DBMSBoundary che a sua volta comunica al DBMS che rappresenta la repository.

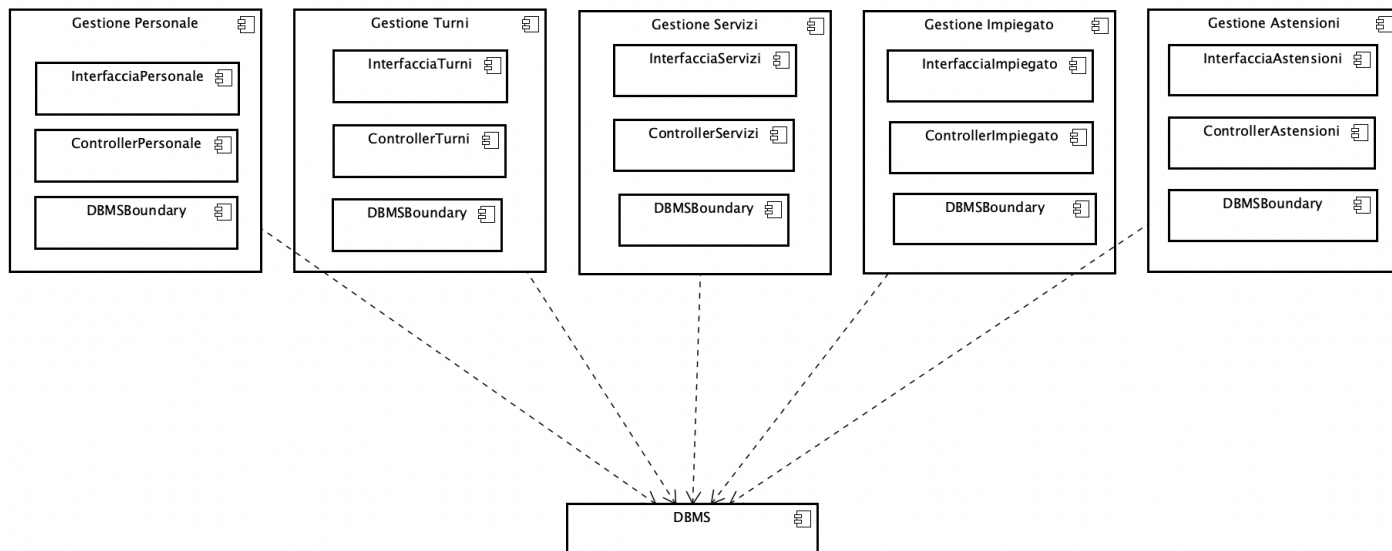
## **Requisiti minimi per l'utilizzo del software**

Per funzionare in modo corretto il software ha bisogno di:

- Connessione internet stabile per permettere di comunicare con il DBMS e mandare mail senza problemi.
- Di essere sempre attivo per garantire che il servizio a priorità più alta sia svolto per più tempo rispetto agli altri servizi e per controllare anche quando chiudere il servizio a priorità più bassa se è necessario.

## Decomposizione in sottosistemi

La decomposizione del sistema in sottosistemi viene a seguire delineata:



Il sottosistema Gestione Personale consente al personale di effettuare le azioni più fondamentali come poter effettuare il login, rilevare la presenza/uscita, permette all'amministratore di registrare gli impiegati oppure effettuare la ricerca per poterne visualizzare/modificare i dati oppure eliminare l'impiegato una volta licenziato. Inoltre, consente al sistema di calcolare lo stipendio degli impiegati ogni mese.

Il sottosistema Gestione Turni permette all'utente generico, che sia l'impiegato o l'amministratore, di visualizzare i turni in base alla data inserita dall'utente. Inoltre, consente al sistema di generare i turni ogni 3 mesi.

Il sottosistema Gestione Servizi permette all'utente generico di visualizzare i servizi con l'annesso stato di attività e il numero di dipendenti che è al momento della visualizzazione in turno. Inoltre, consente al sistema di controllare che ogni giorno il servizio a priorità più alta abbia più dipendenti in turno rispetto agli altri servizi e controlla, nel caso di mancanza di personale, di chiudere il servizio a priorità più bassa.

Il sottosistema Gestione Impiegato consente agli impiegati di svolgere le funzionalità principali riguardanti essi, ovvero comunicare il ritardo, richiedere un permesso, visualizzare i propri dati o la propria situazione lavorativa.

Il sottosistema Gestione Astensioni consente agli impiegati di comunicare o richiedere le astensioni, quindi comunicare il congedo parentale, la malattia, l'assenza, richiedere le ferie e lo sciopero. Inoltre, consente al sistema di gestire le richieste di scioperi.

## Suddivisione degli oggetti all'interno dei sottosistemi e delle componenti

### Gestione Turni:

#### Interfaccia

<<boundary>>  
MenuVisualizzaTurni

<<boundary>>  
ModuloVisualizzaTurni

<<boundary>>  
BoundarySistema

<<boundary>>  
PortaleImpiegato

<<boundary>>  
PortaleAmministratore

#### Controller

<<control>>  
GestoreVisualizzaTurni

<<control>>  
GestoreSistema

## Gestione Servizi:

### Interfaccia

<<boundary>>  
VisualizzaServizi

<<boundary>>  
PortaleAmministratore

<<boundary>>  
PortaleImpiegato

<<boundary>>  
BoundarySistema

### Controller

<<control>>  
GestoreVisualizzaServizi

<<control>>  
GestoreSistema

## Gestione Impiegato:

### Interfaccia

<<boundary>>  
MenuSituazioneLavorativa

<<boundary>>  
MenuVisualizzaDati

<<boundary>>  
PortaleImpiegato

<<boundary>>  
ModuloRichiestaPermesso

<<boundary>>  
ModuloComunicaRitardo

### Controller

<<control>>  
GestoreComunicaRitardo

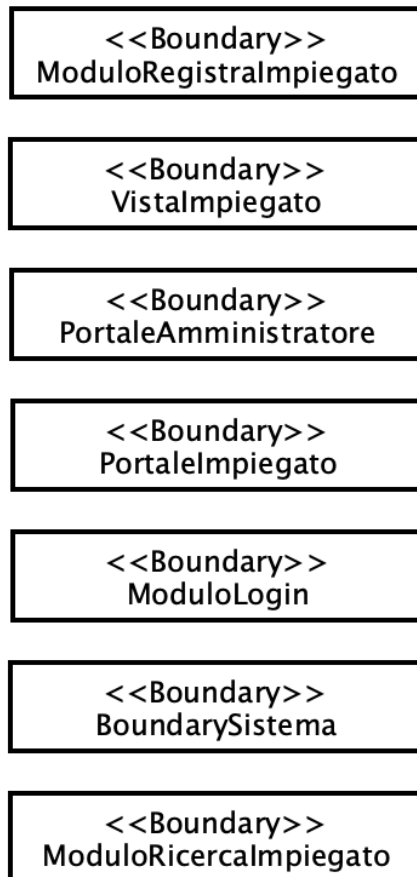
<<control>>  
GestoreRichiestaPermesso

<<control>>  
GestoreSituazioneLavorativa

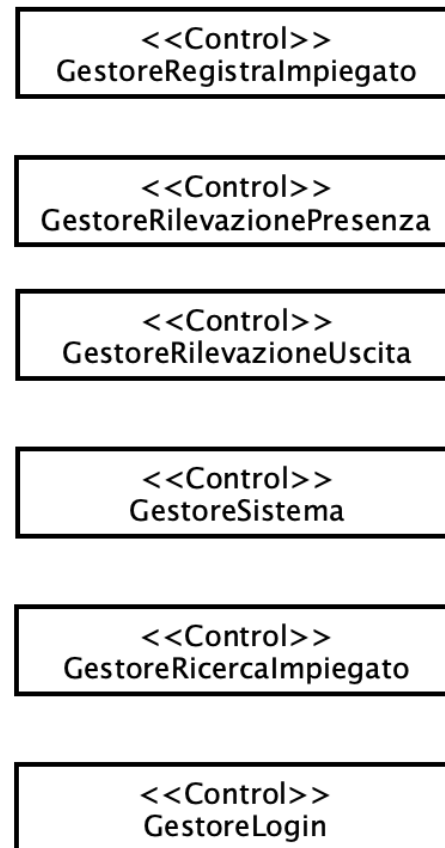
<<control>>  
GestoreVisualizzaDati

## Gestione Personale:

### Interfaccia

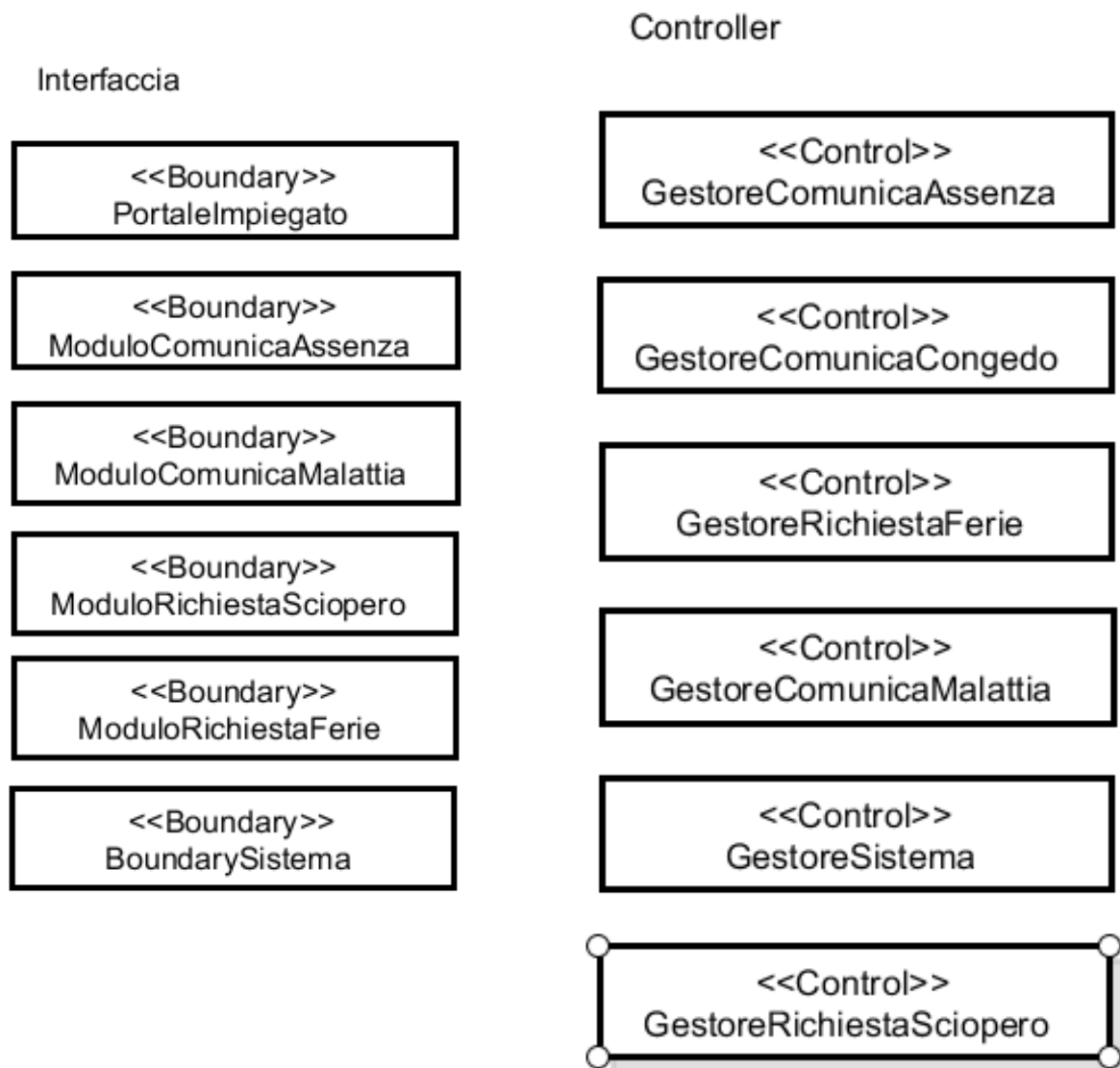


### Controller

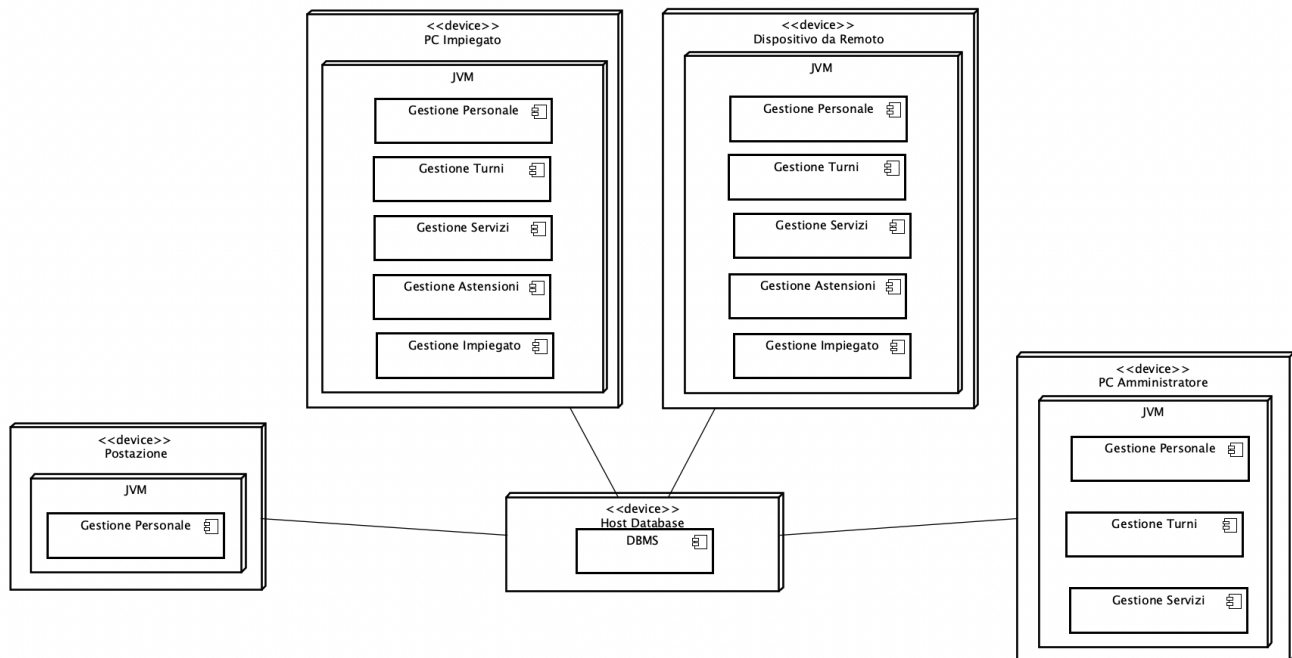




## Gestione Astensioni



## Mappatura Hardware/Software



La mappatura è stata effettuata sulla base del modello architetturale Repository.

La Postazione rappresenta il dispositivo che viene posto all'ingresso dell'azienda per effettuare la rilevazione della presenza o dell'uscita.

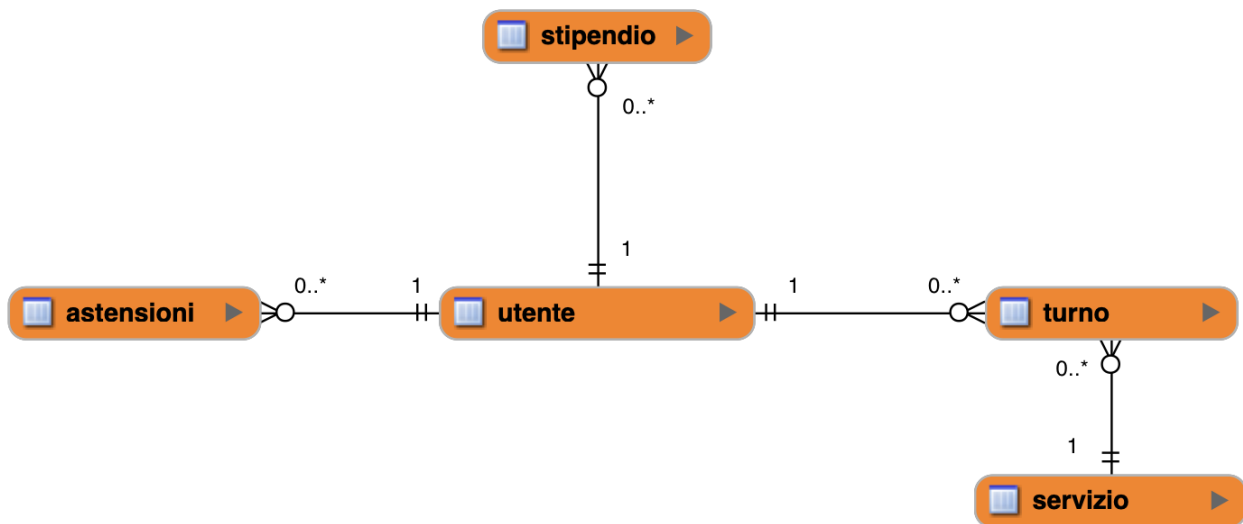
Il PC impiegato rappresenta il dispositivo posto nelle postazioni relative agli impiegati e permette di svolgere tutte le funzioni del sistema relative all'impiegato.

Il PC Amministratore rappresenta il dispositivo posto nella postazione relativa all'amministratore e permette di svolgere tutte le funzioni del sistema relative all'amministratore.

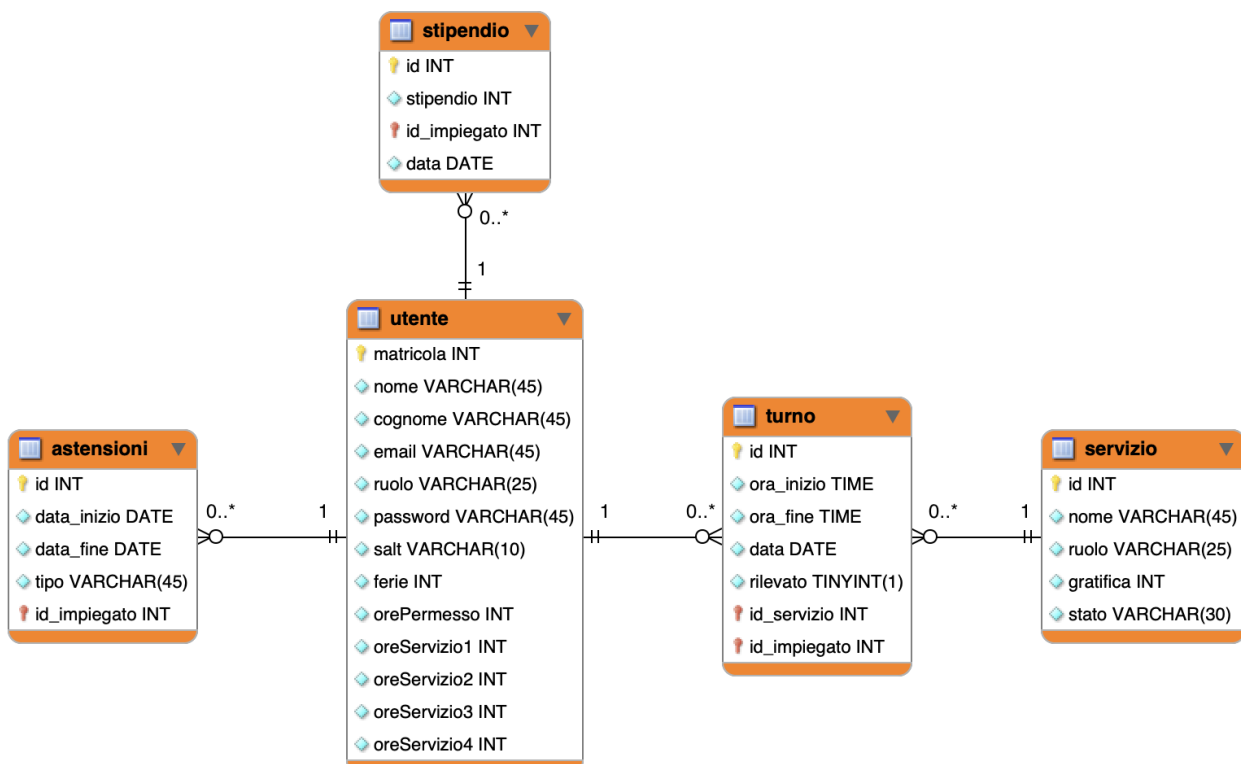
Il Dispositivo da remoto permette di eseguire le stesse funzioni del PC impiegato ma da remoto in modo da consentire all'impiegato di comunicare l'assenza o altre funzionalità.

## Gestione dei dati persistenti

### Modello E-R



### Modello Relazionale



I simboli rappresentanti delle chiavi gialle rappresentano le primary key della tabella mentre i simboli delle chiavi rosse rappresentano le foreign key.

I vincoli di tupla presenti sono i seguenti:

- stipendio:
  - **id\_impiegato** referencia **matricola** nella tabella utente.
- astensioni:
  - **id\_impiegato** referencia **matricola** nella tabella utente.
- turno:
  - **id\_impiegato** referencia **matricola** nella tabella utente.
  - **id\_servizio** referencia **id** nella tabella servizio.

## Struttura delle Tabelle

### Astensioni:

Nome	Tipo	Vincoli	Descrizione
id	INTEGER	PK, AUTO INCREMENT	Codice univoco dell'astensione
data_inizio	DATE		Data dell'inizio dell'astensione
data_fine	DATE		Data della fine dell'astensione
tipo	VARCHAR(45)		Tipo di astensione
id_impiegato	INTEGER	FK	Matricola dell'impiegato

### Stipendio:

Nome	Tipo	Vincoli	Descrizione
id	INTEGER	PK, AUTO INCREMENT	Codice univoco dello stipendio
stipendio	INTEGER		Valore dello stipendio
id_impiegato	INTEGER	FK	Matricola dell'impiegato
data	DATE		Data dell'ultima erogazione

**Utente:**

Nome	Tipo	Vincoli	Descrizione
matricola	INTEGER	PK, AUTO INCREMENT	Matricola dell'impiegato
nome	VARCHAR(45)		Nome dell'impiegato
cognome	VARCHAR(45)		Cognome dell'impiegato
email	VARCHAR(45)		Email dell'impiegato
ruolo	VARCHAR(25)		Ruolo dell'impiegato
password	VARCHAR(45)		Password dell'impiegato
salt	VARCHAR(10)		Attributo da aggiungere alla password prima di crittografarla
ferie	INTEGER	DEFAULT 28	Giorni di ferie rimanenti
orePermesso	INTEGER	DEFAULT 80	Ore di permesso rimanenti
oreServizio1	INTEGER	DEFAULT 0	Ore di servizio svolte nel servizio1
oreServizio2	INTEGER	DEFAULT 0	Ore di servizio svolte nel servizio2
oreServizio3	INTEGER	DEFAULT 0	Ore di servizio svolte nel servizio3
oreServizio4	INTEGER	DEFAULT 0	Ore di servizio svolte nel servizio4

**Turno:**

Nome	Tipo	Vincoli	Descrizione
id	INTEGER	PK, AUTO INCREMENT,	Codice univoco del turno
ora_inizio	TIME		Ora dell'inizio del turno
ora_fine	TIME		Ora della fine del turno
data	DATE		Data del turno
rilevato	BOOLEAN	DEFAULT FALSE	Flag che indica se l'impiegato ha effettuato la rilevazione della presenza o meno
id_servizio	INTEGER	FK	Codice univoco del servizio
id_impiegato	INTEGER	FK	Codice univoco dell'impiegato, cioè la matricola

**Servizio:**

Nome	Tipo	Vincoli	Descrizione
id	INTEGER	PK, AUTO INCREMENT,	Codice univoco del servizio
nome	VARCHAR(45)		Nome del servizio
ruolo	VARCHAR(25)		Grado (ruolo) dell'impiegato che svolge quel servizio
gratifica	INTEGER		Gratifica percepita dagli impiegati che lavorano in quel servizio
stato	VARCHAR(30)	DEFAULT ATTIVO	Flag che indica se il servizio è attivo o chiuso