

Object Design

1. Introduzione

1.1 Object Design Trade-offs

Interfaccia vs usabilità - L'interfaccia è stata realizzata sfruttando elementi di facile comprensione dove un utente medio dovrebbe riuscire a intuire le azioni che compie.

Comprensibilità vs tempo - Per rendere comprensibile il codice anche a chi non lo ha implementato direttamente, sono stati aggiunti molti commenti alle linee di codice per renderne immediata la comprensione, questo ha richiesto un lasso di tempo pari ad un quinto del tempo impiegato a scrivere il resto del codice.

Sicurezza vs efficienza - La sicurezza è uno degli aspetti più importanti per quanto riguarda un sistema client-server considerata la moltitudine e la varietà di attacchi che gli possono essere portati. Tutte le informazioni dunque saranno immagazzinate all'interno di un DB in modo criptato. Si è inoltre optato di implementare una semplice sicurezza che si basa su una combinazione di utente e password per interagire col sistema. Tali controlli sono anche efficienti avendo un ritardo praticamente nullo.

1.2 Linee Guida per la Documentazione delle Interfacce

E' stato scelto di sviluppare il sistema sfruttando le potenzialità del linguaggio HTML per quanto riguarda le parti statiche delle pagine lato client, e il linguaggio PHP per tutte le funzioni dinamiche lato server.

Per quanto riguarda la parte di memorizzazione delle informazioni è stato utilizzato il software "PhpMyAdmin", un DBMS facilmente adattabile a tutti i server attualmente utilizzati.

Le classi PHP sono state strutturate in modo che ognuna rappresenti un'entità da manipolare, implementata tramite un solo file che ne descrive chiaramente le funzionalità interne.

Ogni classe è definita nel seguente modo:

- *Nome e descrizione della classe.*
- *Dichiarazione di classe:* Comprende variabili globali, variabili di istanza, funzioni, costruttori ed eventuali commenti locali.

1.3 Definizioni, acronimi e abbreviazioni

- **DB** = Database.
- **DBMS** = Database Management System, sistema che si occupa di gestire i dati presenti nel DB.
- **Root** = Indica il punto iniziale del file system. Il nome deriva dal riferimento alla sua organizzazione ad albero (quindi alla sua radice) comune a molti file system.

1.4 Riferimenti

Bernd Bruegge e Allen H. Dutoit - Object-Oriented Software Engineering– Prentice Hall.

Sommerville, Software Engineering – Addison Wesley.

Documentazione di progetto

2. Packages

I file del sistema sono organizzati in un package principale, che contiene a sua volta tre package: `php`, `img`, `style` (area utente), `ccs` (index). L'interazione col sistema parte dal file `"index.php"`, modulo indice visualizzato da qualsiasi utente che si interfaccia per la prima volta con il sistema.

- Il package `"php"`, contiene le gli script che si occupano della logica del sistema.
- Il package `"img"`, contiene le immagini del sito web
- Il package `"style/css"` contiene i fogli di stile CSS (package `style` per l'index e package `css` per l'area utente)

3. Class Interfaces

Le classi sono state realizzate grazie al software *notepad++*. Il codice viene colorato per renderlo più leggibile, identificare meglio le parti di implementazione e differire i commenti del codice. Di seguito vi è un esempio del codice PHP della classe *Utenti*:

```
/**
 * Interface: InterfaceAdmin
 * Description: Interfaccia che descrive i metodi di Admin
 */
interface InterfaceAdmin{

    /*******METODI DI ACCESSO*****//

    /**
    Function: getId
    Description: restituisce l'id Admin
    */
    function getId();
    /**
    Function: getNome
    Description: restituisce il nome Admin
    */
    function getNome();
    /**
    Function: getCognome
    Description: restituisce il cognome Admin
    */
    function getCognome();
    /**
    Function: getUsername
    Description: restituisce l'username Admin
    */
    function getUsername();
    /**
    Function: getPassword
    Description: restituisce la password Admin
    */
    function getPassword();
```

```

//*****METODI MODIFICATORI*****//

/**
Function: setId
Description: setta l'id Admin
*/
function setId($id);

/**
Function: getNome
Description: setta il nome Admin
*/
function setNome($nome);

/**
Function: getCognome
Description: restituisce il cognome Admin
*/
function setCognome($cognome);

/**
Function: setUsername
Description: setta l'username Admin
*/
function setUsername($username);

/**
Function: setPassword
Description: restituisce la password Admin
*/
function setPassword($password);
}

/**
* Interface: InterfaceAdminManager
* Description: Interfaccia che descrive i metodi di AdminManager
*/
interface InterfaceAdminManager {

/**
Function: selectByUsername
Description: restituisce admin prendendo in input $username
*/
function selectByUsername($username);

/**
Function: selectById
Description: restituisce admin prendendo in input $id
*/
function selectById($id);

/**
Function: modificaAdmin
Description: modifica admin prendendo in input $dati e $id
*/
function modificaAdmin($dati, $id);

}

```

```

/**
 * Interface: InterfaceCorso
 * Description: Interfaccia che descrive i metodi di Corso
 */
    interface InterfaceCorso {

//*****METODI DI ACCESSO*****//

        /**
        Function: getId
        Description: restituisce l'id Corso
        */
        function getId();

        /**
        Function: getNomeCorso
        Description: restituisce il nome del Corso
        */
        function getNomeCorso();

        /**
        Function: getDescrizione
        Description: restituisce la descrizione del Corso
        */
        function getDescrizione();

        /**
        Function: getPrezzo
        Description: restituisce il prezzo del Corso
        */
        function getPrezzo();

        /**
        Function: getIdTrainer
        Description: restituisce l'id del trainer del Corso
        */
        function getIdTrainer();

//*****METODI MODIFICATORI*****//

        /**
        Function: setId
        Description: setta l'id Corso
        */
        function setId($id);

        /**
        Function: setNomeCorso
        Description: setta il nome del Corso
        */
        function setNomeCorso($nome_corso);

        /**
        Function: setDescrizione
        Description: setta la descrizione del Corso
        */
        function setDescrizione($descrizione);

        /**
        Function: setPrezzo
        Description: setta il prezzo del Corso
        */
        function setPrezzo($prezzo);

        /**
        Function: setIdTrainer
        Description: setta l'id del trainer del Corso

```

```

*/
function setIdTrainer($id_trainer);

}

/**
 * Interface: InterfaceCorsoManager
 * Description: Interfaccia che descrive i metodi di CorsoManager
 */
interface InterfaceCorsoManager {

    /**
    Function: modificaCorso
    Description: modifica corso prendendo in input $dati e $id
    */
    function modificaCorso($dati,$id);

    /**
    Function: inserisciCorso
    Description: inserisce corso prendendo in input $dati
    */
    function inserisciCorso($dati);

    /**
    Function: selectByUtente
    Description: restituisce corsi prendendo in input $id utente
    */
    function selectByUtente($id);

    /**
    Function: selectByTrainer
    Description: restituisce corsi prendendo in input $id trainer
    */
    function selectByTrainer($id);

    /**
    Function: selectById
    Description: restituisce corso prendendo in input $id corso
    */
    function selectById($id);

    /**
    Function: selectCorsiDisponibili
    Description: restituisce corsi prendendo in input $id utente
    */
    function selectCorsiDisponibili($id);

    /**
    Function: select
    Description: restituisce corsi prendendo in input una stringa
    */
    function select($nome);

    /**
    Function: selectByNome
    Description: restituisce corso prendendo in input $nome corso
    */
    function selectByNome($nome);

    /**
    Function: eliminaCorso
    Description: restituisce corso prendendo in input $id corso
    */
    function eliminaCorso($id);

}

```

```

/**
 * Interface: InterfaceIscrizione
 * Description: Interfaccia che descrive i metodi di Iscrizione
 */
    interface InterfaceIscrizione {

//*****METODI DI ACCESSO*****//

        /**
        Function: getIdCorso
        Description: restituisce l'id Corso
        */
        function getIdCorso();

        /**
        Function: getIdUtente
        Description: restituisce l'id Utente
        */
        function getIdUtente();

        /**
        Function: getDataIscrizione
        Description: restituisce data_iscrizione
        */
        function getDataIscrizione();

//*****METODI MODIFICATORI*****//

        /**
        Function: setIdCorso
        Description: setta l'id Corso
        */
        function setIdCorso($id_corso);

        /**
        Function: setIdUtente
        Description: setta l'id Utente
        */
        function setIdUtente($id_utente);

        /**
        Function: setDataIscrizione
        Description: setta data_iscrizione
        */
        function setDataIscrizione($data_iscrizione);

```

```

/**
 * Interface: InterfaceIscrizioneManager
 * Description: Interfaccia che descrive i metodi di IscrizioneManager
 */
interface InterfaceIscrizioneManager {

    /**
    Function: iscriviti
    Description: iscrive un utente ad un corso
    */
    function iscriviti($utente, $corso, $data_iscrizione);

    /**
    Function: selectByUtente
    Description: restituisce l'iscrizione prendendo in input $id utente
    */
    function selectByUtente($id);

    /**
    Function: selectByCorso
    Description: restituisce l'iscrizione prendendo in input $id corso
    */
    function selectByCorso($id);

    /**
    Function: eliminaIscrizioneByUtente
    Description: elimina l'iscrizione prendendo in input $id utente
    */
    function eliminaIscrizioneByUtente($id);

    /**
    Function: eliminaIscrizioneByCorso
    Description: elimina l'iscrizione prendendo in input $id corso
    */
    function eliminaIscrizioneByCorso($id);

    /**
    Function: eliminaIscrizione
    Description: elimina l'iscrizione prendendo in input $id utente e $id
corso
    */
    function eliminaIscrizione($utente, $corso);
}

```



```

/**
 * Interface: InterfaceScheda
 * Description: Interfaccia che descrive i metodi di Scheda
 */
    interface InterfaceScheda {

//*****METODI DI ACCESSO*****//

        /**
        Function: getId
        Description: restituisce l'id Scheda
        */
        function getId();

        /**
        Function: getIdUtente
        Description: restituisce l'id Utente
        */
        function getIdUtente();

        /**
        Function: getDescrizione
        Description: restituisce la descrizione
        */
        function getDescrizione();

        /**
        Function: getIdTrainer
        Description: restituisce l'id Trainer
        */
        function getIdTrainer();

//*****METODI MODIFICATORI*****//

        /**
        Function: setId
        Description: setta l'id Scheda
        */
        function setId($id);

        /**
        Function: setIdUtente
        Description: setta l'id Utente
        */
        function setIdUtente($id_utente);

        /**
        Function: setDescription
        Description: setta la descrizione
        */
        function setDescription($descrizione);

        /**
        Function: setIdTrainer
        Description: setta l'id Trainer
        */
        function setIdTrainer($id_trainer);

    }

```

```

/**
 * Interface: InterfaceSchedaManager
 * Description: Interfaccia che descrive i metodi di SchedaManager
 */
interface InterfaceSchedaManager {

    /**
     Function: modificaScheda
     Description: modifica la scheda prendendo come input $descrizione,
     $id_trainer, $id_utente
    */
    function modificaScheda($descrizione, $id_trainer, $id_utente);

    /**
     Function: inserisciScheda
     Description: inserisce una scheda prendendo come input $id_utente,
     $descrizione, $id_trainer
    */
    function inserisciScheda($id_utente,$descrizione,$id_trainer);

    /**
     Function: selectByUtente
     Description: restituisce una scheda prendendo come input $id Utente
    */
    function selectByUtente($id);

    /**
     Function: eliminaScheda
     Description: elimina una scheda prendendo come input $id Scheda
    */
    function eliminaScheda($id);

}

```

```

/**
 * interface: InterfaceTrainer
 * Description: Classe che implementa un account Trainer
 */
    interface InterfaceTrainer {

//*****METODI DI ACCESSO*****//

    /**
    Function: getId
    Description: restituisce l'id Trainer
    */
    function getId();

    /**
    Function: getNome
    Description: restituisce il nome Trainer
    */
    function getNome();

    /**
    Function: getCognome
    Description: restituisce il cognome Trainer
    */
    function getCognome();

    /**
    Function: getUsername
    Description: restituisce l'username Trainer
    */
    function getUsername();

    /**
    Function: getPassword
    Description: restituisce la password Trainer
    */
    function getPassword();

    /**
    Function: getEmail
    Description: restituisce l'email Trainer
    */
    function getEmail();

    /**
    Function: getDataNascita
    Description: restituisce la data di nascita Trainer
    */
    function getDataNascita();

    /**
    Function: getComuneNascita
    Description: restituisce il comune di nascita Trainer
    */
    function getComuneNascita();

    /**
    Function: getIndirizzo
    Description: restituisce l'indirizzo Trainer
    */
    function getIndirizzo();

    /**
    Function: getSesso
    Description: restituisce il sesso del Trainer
    */
    function getSesso();

```

```

//*****METODI MODIFICATORI*****//

/**
Function: setId
Description: setta l'id Trainer
*/
function setId($id);

/**
Function: setNome
Description: setta il nome Trainer
*/
function setNome($nome);

/**
Function: setCognome
Description: setta il cognome Trainer
*/
function setCognome($cognome);

/**
Function: setUsername
Description: setta l'username Trainer
*/
function setUsername($username);

/**
Function: setPassword
Description: setta la password Trainer
*/
function setPassword($password);

/**
Function: setEmail
Description: setta l'email Trainer
*/
function setEmail($email);

/**
Function: setDataNascita
Description: setta la data di nascita Trainer
*/
function setDataNascita($data_nascita);

/**
Function: setComuneNascita
Description: setta il comune di nascita Trainer
*/
function setComuneNascita($comune_nascita);

/**
Function: setIndirizzo
Description: setta l'indirizzo Trainer
*/
function setIndirizzo($indirizzo);

/**
Function: setSesso
Description: setta il sesso del Trainer
*/
function setSesso($sesso);

}

```

```

/**
 * Interface: InterfaceTrainerManager
 * Description: Interfaccia che descrive i metodi di TrainerManager
 */
interface InterfaceTrainerManager {

    /**
    Function: modificaTrainer
    Description: modifica il trainer prendendo come input $dati e $id
    */
    function modificaTrainer($dati, $id);

    /**
    Function: selectByUsername
    Description: restituisce un trainer prendendo come input $username
    */
    function selectByUsername($username);

    /**
    Function: selectById
    Description: restituisce un trainer prendendo come input $id
    */
    function selectById($id);

    /**
    Function: select
    Description: restituisce un trainer prendendo come input una stringa
    */
    function select($nome);

    /**
    Function: inserisciTrainer
    Description: inserisce un trainer prendendo come input $dati
    */
    function inserisciTrainer($dati);

    /**
    Function: eliminaTrainer
    Description: elimina un trainer prendendo come input $id
    */
    function eliminaTrainer($id);
}

```

```

/**
 * interface: InterfaceUtente
 * Description: Classe che implementa un account Utente
 */
    interface InterfaceUtente {

//*****METODI DI ACCESSO*****//

    /**
    Function: getId
    Description: restituisce l'id Utente
    */
    function getId();

    /**
    Function: getNome
    Description: restituisce il nome Utente
    */
    function getNome();

    /**
    Function: getCognome
    Description: restituisce il cognome Utente
    */
    function getCognome();

    /**
    Function: getUsername
    Description: restituisce l'username Utente
    */
    function getUsername();

    /**
    Function: getPassword
    Description: restituisce la password Utente
    */
    function getPassword();

    /**
    Function: getEmail
    Description: restituisce l'email Utente
    */
    function getEmail();

    /**
    Function: getDataNascita
    Description: restituisce la data di nascita Utente
    */
    function getDataNascita();

    /**
    Function: getComuneNascita
    Description: restituisce il comune di nascita Utente
    */
    function getComuneNascita();

    /**
    Function: getIndirizzo
    Description: restituisce l'indirizzo Utente
    */
    function getIndirizzo();

    /**
    Function: getSesso
    Description: restituisce il sesso dell'Utente
    */
    function getSesso();

```

```

//*****METODI MODIFICATORI*****//

/**
Function: setId
Description: setta l'id Utente
*/
function setId($id);

/**
Function: setName
Description: setta il nome Utente
*/
function setName($nome);

/**
Function: setCognome
Description: setta il cognome Utente
*/
function setCognome($cognome);

/**
Function: setUsername
Description: setta l'username Utente
*/
function setUsername($username);

/**
Function: setPassword
Description: setta la password Utente
*/
function setPassword($password);

/**
Function: setEmail
Description: setta l'email Utente
*/
function setEmail($email);

/**
Function: setDataNascita
Description: setta la data di nascita Utente
*/
function setDataNascita($data_nascita);

/**
Function: setComuneNascita
Description: setta il comune di nascita Utente
*/
function setComuneNascita($comune_nascita);

/**
Function: setIndirizzo
Description: setta l'indirizzo Utente
*/
function setIndirizzo($indirizzo);

/**
Function: setSesso
Description: setta il sesso dell'Utente
*/
function setSesso($sesso);
}

```

```

/**
 * Interface: InterfaceUtenteManager
 * Description: Interfaccia che descrive i metodi di UtenteManager
 */
interface InterfaceUtenteManager {

    /**
     Function: modificaUtente
     Description: modifica un utente prendendo come input $dati e $id
     */
     function modificaUtente($dati, $id);

    /**
     Function: selectByUsername
     Description: restituisce un utente prendendo come input $username
     */
     function selectByUsername($username);

    /**
     Function: selectById
     Description: restituisce un utente prendendo come input $id
     */
     function selectById($id);

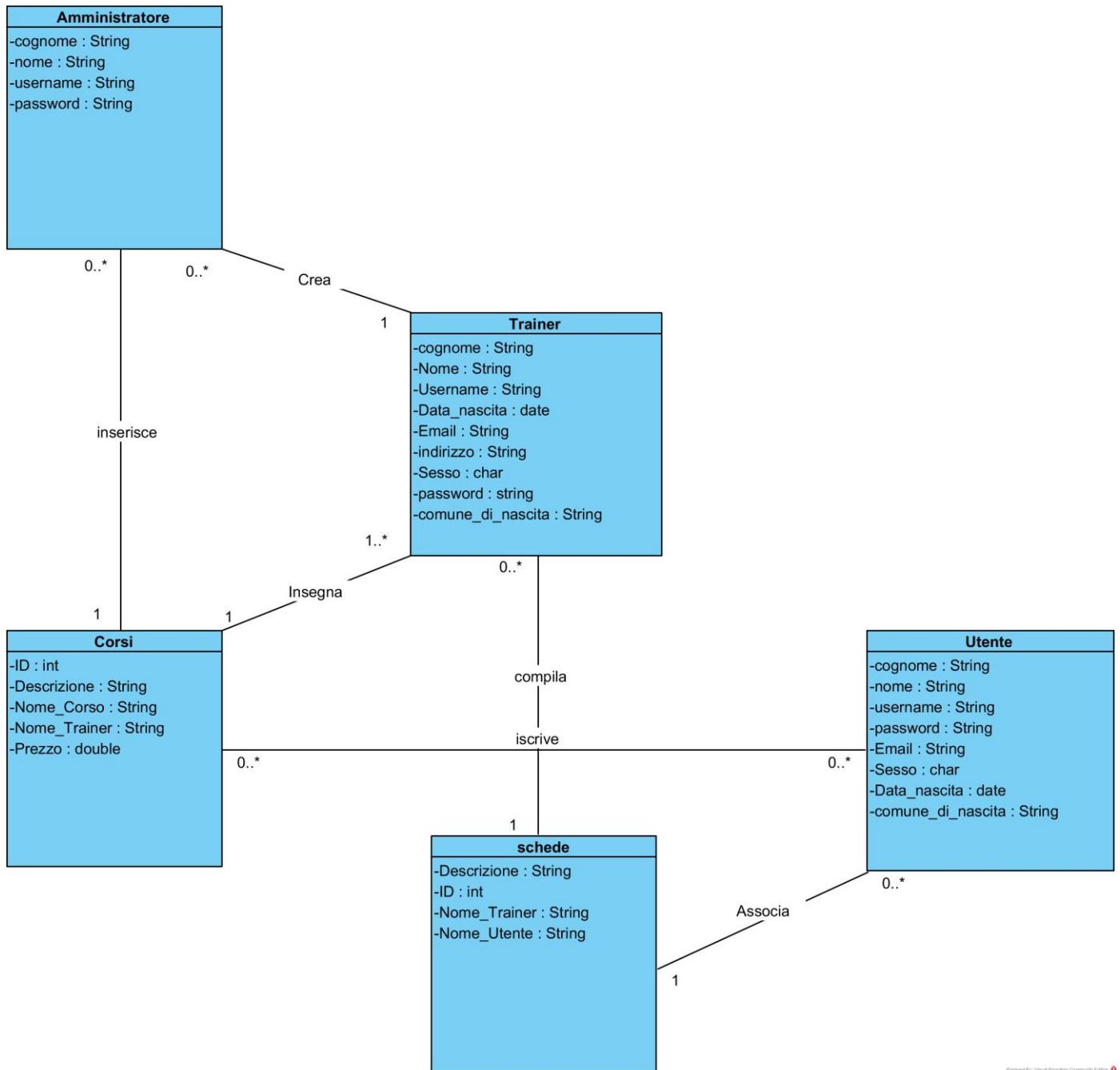
    /**
     Function: select
     Description: restituisce un utente prendendo come input una stringa
     */
     function select($nome);

    /**
     Function: inserisciUtente
     Description: inserisce un utente prendendo come input $dati
     */
     function inserisciUtente($dati);

    /**
     Function: eliminaTrainer
     Description: elimina un utente prendendo come input $id
     */
     function eliminaUtente($id);
}

```


Class Diagram:



Powered By: Visual Paradigm Community Edition

4. Class Diagram

