



UNIVERSITÀ DEGLI STUDI DELL'AQUILA

Dipartimento di Ingegneria e Scienze dell'Informazione e
Matematica

Documentazione progetto programmazione per il web

A cura di

Lorenzo D'amico, Giordana Di Fabio e Andrea Esposito

Anno Accademico 2024/2025

Indice

1	Introduzione	1
1.1	Come scaricare e accedere	1
1.2	Credenziali	2
2	Modello Realizzazione	3
2.1	Funzionalità	3
2.2	Contributi	5
3	Altri documenti	6

Capitolo 1

Introduzione

Questa documentazione descrive il sistema sviluppato nell'ambito del corso di Programmazione per il web e si riferisce al progetto selezionato: **Slope**, una piattaforma online dedicata all'acquisto di skipass.

Il sistema è progettato per utenti individuali e famiglie, offrendo funzionalità guidate e vincoli specifici per garantire semplicità, sicurezza e flessibilità nella gestione degli accessi alle stazioni sciistiche.

1.1 Come scaricare e accedere

Per l'installazione di Slope in locale è richiesto un server che ospiti l'applicazione e un database *SQL*. Si consiglia di utilizzare *XAMPP*, che fornisce un server *Apache* e *MySQL*.

1. Scaricare la cartella in formato zip da GitHub.
2. Spostare la repository nella cartella *htdocs* e rinominarla Slope.
3. Per essere sicuri che l'applicazione Slope carichi correttamente il database, aprire il file di configurazione *my.ini* del server MySQL (è possibile farlo direttamente dal pannello di controllo di XAMPP cliccando su Config accanto a MySQL, oppure accedendo al percorso di installazione di XAMPP nella cartella *mysql*).

Nella sezione `[mysqld]`, cercare la riga *max_allowed_packet* (una variabile di configurazione che definisce la dimensione massima dei pacchetti di rete o comunicazione che MySQL può ricevere o inviare). Modificare il valore in 64M o superiore. Salvare e riavviare il server MySQL.

4. Avviare i server Apache e MySQL dal pannello di controllo di XAMPP.
5. Aprire un browser e digitare *localhost/phpmyadmin*.
Fare clic su Importa, poi su Scegli file e selezionare il file *slope.sql* che si trova nella repository scaricata.

Se nella configurazione di phpMyAdmin è stato modificato username o password, aprire il file `config.php` nella cartella `config` della repository e aggiornare le voci username e password con i valori corretti.

6. Prima di procedere con il lancio dell'applicazione, assicurarsi che la riscrittura degli URL sia attiva sul server Apache.

Per controllare, aprire il file `httpd.conf` (disponibile dal pannello di controllo XAMPP) e cercare la seguente riga:

```
# LoadModule rewrite_module modules/mod_rewrite.so
```

Se è presente il simbolo `#` davanti alla riga, rimuoverlo. Successivamente, cercare la sezione relativa alla directory `htdocs` e assicurarsi che all'interno del blocco `<Directory "C:/xampp/htdocs">` sia presente la riga:

```
AllowOverride All
```

Questo è necessario per abilitare i file `.htaccess`. Salvare e riavviare il server Apache.

7. A questo punto è possibile accedere alla piattaforma. Assicurarsi che Apache e MySQL siano attivi, quindi aprire un browser e digitare nella barra degli indirizzi:

`localhost/slope`

1.2 Credenziali

Per accedere alla parte utente bisogna registrarsi dalla home page del sito cliccando il bottone "registrati", una volta effettuata la registrazione è possibile utilizzare a pieno la piattaforma.

Per accedere alla parte di admin bisogna inserire nella barra di ricerca:

`https://localhost/Slope/Admin/login`

Credenziali admin:

- username: admin
- password: admin

Capitolo 2

Modello Realizzazione

La piattaforma Slope è stata sviluppata seguendo il paradigma MVC (Model-View-Controller), un'architettura che consente di suddividere in modo chiaro le responsabilità tra i vari componenti del sistema. Questo approccio ha facilitato l'individuazione e la correzione degli errori riscontrati durante la fase di testing, rendendo il processo di debug più efficiente.

Lo sviluppo è iniziato con la definizione delle classi appartenenti al livello Entity, per poi proseguire con la creazione del livello Foundation, all'interno del quale è stata implementata una classe fondamentale:

- *PersistentManager*, per la gestione della persistenza degli oggetti.

Le classi sono state inizialmente dotate dei metodi CRUD, ai quali sono stati successivamente aggiunti altri metodi durante l'implementazione del livello Controller.

Per la gestione dei dati è stato utilizzato un database MySQL, amministrato tramite phpMyAdmin incluso in XAMPP.

La logica dei Controller si basa su una classe principale, CFrontController, incaricata di ricevere le richieste da index.php e smistarle verso le rispettive classi C.

Per quanto riguarda la View, è stato utilizzato il motore di template Smarty.

2.1 Funzionalità

- **Utente non registrato**
 - Visualizzazione prezzi per ogni impianto
 - Visualizzazione mappa impianti sciistici
 - Registrazione
- **Utente registrato**
 - Accesso tramite Login
 - Modifica password tramite "Password dimenticata"
 - Visualizzazione informazioni preliminari per ogni impianto

- Visualizzazioni informazioni specifiche per ogni impianto
- Visualizzazione form di prenotazione
- Visualizzazione form inserimento metodo di pagamento
- Visualizzazione prenotazioni effettuate
- Possibilità di modificare ed eliminare le prenotazioni
- Visualizzazione informazioni relative al profilo, metodo di pagamento e abbonamento
- Possibilità di modifica dati profilo relativi all'utente
- Possibilità di modificare e rimuovere il metodo di pagamento
- Possibilità di acquistare un abbonamento

- **Utente amministratore**

- Visualizzare statistiche riguardo le prenotazioni per ogni impianto in un determinato mese
- Visualizzare il numero di utenti iscritti e il numero di utenti che hanno acquistato un abbonamento
- Aggiungere, modificare o eliminare dati pista
- Aggiungere, modificare o eliminare dati impianto
- Aggiungere, modificare o eliminare dati impianto di risalita
- Aggiungere, modificare o eliminare template skipass relativo ad un template per rappresentare un oggetto generico rappresentato da descrizione, durata e tipo. Esempio ("Giornaliero", 1, "intero")
- Aggiungere, modificare o eliminare template assicurazione relativo ad un template per rappresentare un oggetto generico rappresentato da tipo e valore. Esempio ("intero", 10)
- Aggiungere, modificare o eliminare template abbonamento relativo ad un template per rappresentare un oggetto generico rappresentato da descrizione, valore e sconto. Esempio ("Abbonamento", 200, 5)
- Aggiungere, modificare o eliminare oggetto skipass dal quale poi si potrà acquistare uno skipass
- Modificare immagini per ogni impianto
- Modificare le prenotazioni dei singoli utenti
- Modificare le immagini della pagina iniziale

2.2 Contributi

Attività	Contributi
Step 1-6	Giordana e Andrea e Lorenzo
Entity	Giordana e Andrea
Foundation	Andrea e Giordana e Lorenzo
Controller	Lorenzo
View	Giordana
Script del db	Andrea e Lorenzo
Templates	Giordana
Gestione Email	Lorenzo e Andrea

Tabella 2.1: Tabella per i contributi al progetto

Capitolo 3

Altri documenti

Per non appesantire ulteriormente la documentazione non sono inclusi tutti gli step comprendenti:

- Idea di progetto
- Casi D'uso
- Bozza delle pagine
- Diagramma E/R

Possono però essere visualizzati nel percorso Slope/documentazione.