

**Dream&Fly**  
**System Design Document**  
**Versione 0.4**



Data: 31/01/2024

Progetto: Dream&Fly	Versione: 0.4
Documento: SDD	Data: 31/01/2024

**Coordinatore del progetto:**

Nome	Matricola
Singh Amandeep	0512113476
Paolillo Valentina	0512114820

**Partecipanti:**

Nome	Matricola
Singh Amandeep	0512113476
Paolillo Valentina	0512114820

<b>Scritto da:</b>	Singh Amandeep, Paolillo Valentina
--------------------	------------------------------------

**Revision History**

Data	Versione	Descrizione	Autore
13/12/2023	0.1	Prima stesura	Paolillo Valentina, Singh Amandeep
26/01/2024	0.2	Stesura obiettivi di Design	Paolillo Valentina, Singh Amandeep
30/01/2024	0.3	Continuo stesura sezione 3	Singh Amandeep, Paolillo Valentina
31/01/2024	0.4	Continuo subsystem decomposition	Singh Amandeep, Paolillo Valentina

# Indice

1.	INTRODUCTION .....	4
1.1.	Purpose of the system.....	4
1.2.	Design Goals .....	4
1.3.	Definitions, acronyms, and abbreviations .....	5
1.4.	References .....	5
1.5.	Overview .....	5
2.	CURRENT SOFTWARE ARCHITECTURE .....	5
3.	PROPOSED SOFTWARE ARCHITECTURE .....	5
3.1.	Overview .....	5
3.2.	Subsystem decomposition.....	6
3.3.	Hardware/software mapping .....	7
3.4.	Persistent data management .....	7
3.5.	Access control and security.....	9
3.6.	Global software control.....	10
3.7.	Boundary conditions .....	10
4.	SUBSYSTEM SERVICES .....	11

# 1. INTRODUCTION

## ***1.1. Purpose of the system***

Come meglio illustrato nel capitolo 1 del documento “RAD”, la realizzazione di Dream&Fly ha come obiettivo quello di essere di agevolare la prenotazione di capsule ai viaggiatori. In particolare, con questo progetto, il processo di prenotazione risulterà più rapido, efficiente e intuitivo.

## ***1.2. Design Goals***

Di seguito, vengono descritti gli obiettivi di design che devono essere rispettati nella realizzazione del Sistema:

- **Criteri di performance**

- Throughput

Il sistema deve essere in grado di gestire contemporaneamente un carico massimo di almeno 100 richieste di prenotazione al minuto, senza degradazione delle prestazioni.

- Tempo di risposta

Il sistema deve essere in grado di gestire almeno 500 utenti contemporaneamente senza rallentamenti evidenti. Il tempo di risposta medio per le richieste degli utenti non deve superare gli 2 secondi.

- **Criteri di affidabilità**

- Robustezza

Il sistema deve, in caso di errore, rifiutare i dati in ingresso e rispondere tramite messaggi d'errore.

- Disponibilità

Il sistema deve essere disponibile e funzionante sempre, eventuali periodi di inattività pianificati per manutenzione devono essere comunicati agli utenti in anticipo.

- Sicurezza

Il sistema deve garantire la sicurezza delle password crittografandole prima di salvarle nel database.

- **Criteri di costo**

- Sviluppo

Il costo complessivo del progetto ammonta ad un massimo di 200 ore (max 50 ore per ogni membro del team).

- **Criteri di manutenzione**

- Estendibilità

Sarà possibile aggiungere nuove funzionalità al sistema, in base alle esigenze dell'utenza.

- Portabilità

Il Sistema deve essere fruibile su tutti i dispositivi mobile e desktop in maniera indipendente dal sistema operativo o dall'hardware utilizzato.

- **Criteri dell'utente finale**

- Usabilità

Il Sistema deve avere un'interfaccia semplice e immediata in modo da consentire un'interazione rapida ed efficiente, così da ridurre i tempi di controllo.

### ***1.3. Definitions, acronyms, and abbreviations***

- **RDBMS:** Relational Database Management System (gestore di database basati sul modello relazionale)
- **DB:** DataBase, ovvero "Base di Dati" utilizzata per memorizzare i dati persistenti;
- **SDD:** System Design Document
- **GUI:** Graphical User Interface
- **SQL:** Structured Query Language
- **HTTP:** Hypertext Transfer Protocol
- **JDBC:** Java DataBase Connectivity
- **MySQL:** sistema di gestione di base di dati basato su SQL
- **SQL: Structured Query Language**
- **MySQL: sistema di gestione di base di dati basato su SQL**
- **Tier:** uno dei diversi strati o livelli
- **Interface:** interfaccia
- **Application logic:** logica di applicazione
- **Web Application:** applicazione accessibile via Web per mezzo di una rete Internet
- **Web Browser/Browser:** applicazione software per l'acquisizione, la presentazione e la navigazione di risorse sul web
- **Web Server:** applicazione software che, in esecuzione su un server, è in grado di gestire le richieste di trasferimento di pagine web di un client, tipicamente un Web Browser

### ***1.4. References***

Per stilare la presente documentazione, si è preso come riferimento il libro di testo "Object-Oriented Software Engineering Using UML, Patterns and Java: Third Edition, di Bernd Bruegge ed Allen H. Dutoit".

### ***1.5. Overview***

Dopo questa prima sezione di introduzione del presente Documento, il punto 2 descriverà brevemente l'architettura del Sistema corrente, mentre al punto 3 verrà fornita una dettagliata descrizione dell'architettura del Sistema proposto. In particolare, questa sezione descriverà la decomposizione in sottosistemi, la corrispondenza tra hardware e software, la gestione dei dati persistenti, il controllo degli accessi, la sicurezza, il flusso di controllo e la gestione delle condizioni limite del Sistema proposto. Infine, il punto 4 descriverà i servizi offerti dai sottosistemi individuati.

## **2. CURRENT SOFTWARE ARCHITECTURE**

## **3. PROPOSED SOFTWARE ARCHITECTURE**

### ***3.1. Overview***

Il Sistema proposto è una Web Application che vuole facilitare, velocizzare e rendere più sicura la prenotazione delle capsule aeroportuali. L'architettura software scelta per la realizzazione del Sistema è la Three Tier, la quale, fornendo la possibilità di eseguire ciascun tier sulla propria infrastruttura, offre numerosi vantaggi, tra cui uno sviluppo più veloce e una maggiore scalabilità, affidabilità e sicurezza. Questo pattern architetturale si compone di tre livelli:

1. **Interface:** è il tier di presentazione, dunque delle interfacce utente. Si occupa di visualizzare le informazioni all'utente e di raccogliere informazioni da quest'ultimo;
2. **Application Logic:** è il tier della logica di business dell'applicazione. Si occupa di elaborare le informazioni raccolte nel tier Interface e di aggiungere, rimuovere o modificare i dati nel tier Storage;
3. **Storage:** è il tier dell'accesso ai dati. Si occupa della persistenza delle informazioni e della loro gestione e archiviazione.

Nel Sistema realizzato con la suddetta architettura tutte le comunicazioni passano attraverso l'Application Logic tier, l'Interface tier e lo Storage tier non comunicano mai direttamente tra loro.

### ***3.2. Subsystem decomposition***

I componenti del Sistema sono suddivisi tra i layer Interface, Application Logic e Storage.

L'Interface layer contiene le componenti:

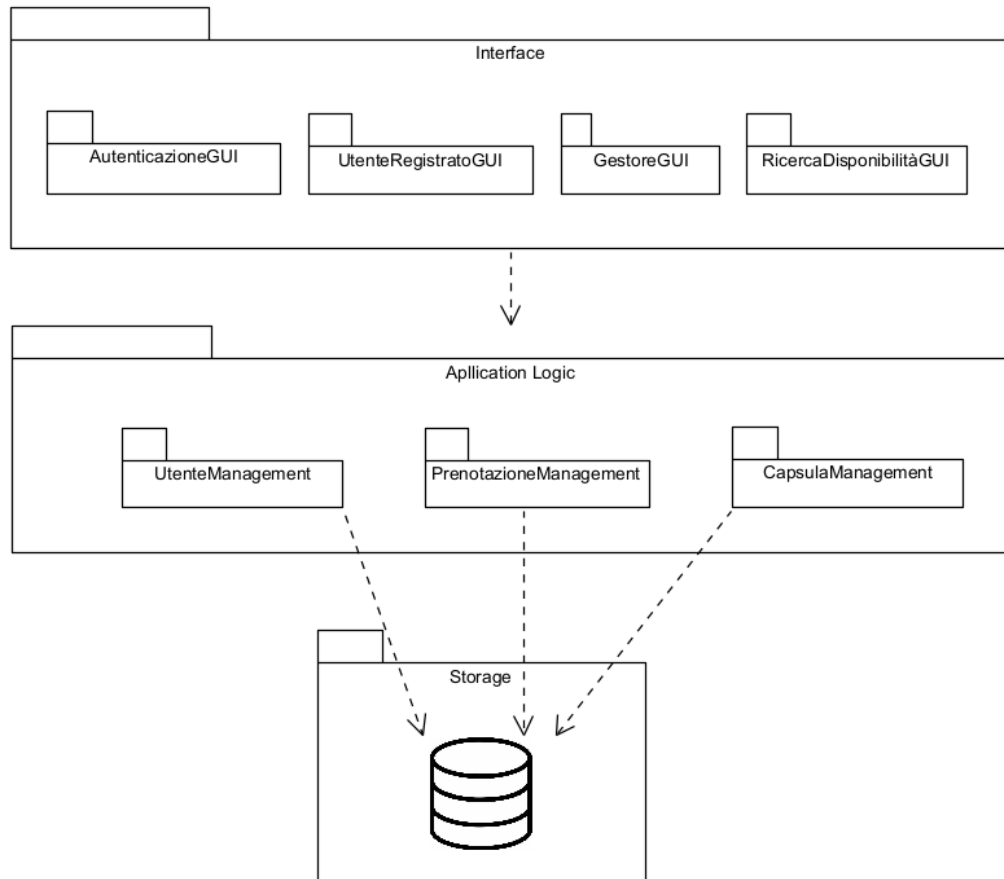
- **AutenticazioneGUI:** consentire agli utenti di autenticarsi o registrarsi.
- **UtenteRegistratoGUI:** rappresenta l'insieme delle pagine web con le quali l'utente registrato (e autenticato) può interagire.
- **GestoreGUI:** rappresenta l'insieme delle pagine web con le quali i gestori del sistema possono interagire a seconda del loro ruolo per poter effettuare varie operazioni.
- **RicercaDisponibilitàGUI:** consente agli utenti di poter cercare le disponibilità anche senza autenticazione.

L'Application Logic layer contiene le componenti:

- **UtenteManagement:** fornisce le funzionalità per l'autenticazione e registrazione, e gestisce in generale la definizione e la gestione dei permessi, garantendo l'accesso alle specifiche aree e funzionalità del sistema in base ai ruoli degli utenti.
- **CapsulaManagement:** fornisce le funzionalità per inserire, modificare o visualizzare i dati di una capsula.
- **PrenotazioneManagement:** fornisce le funzionalità per gestire le prenotazioni delle capsule, inclusa la visualizzazione e la creazione delle prenotazioni.

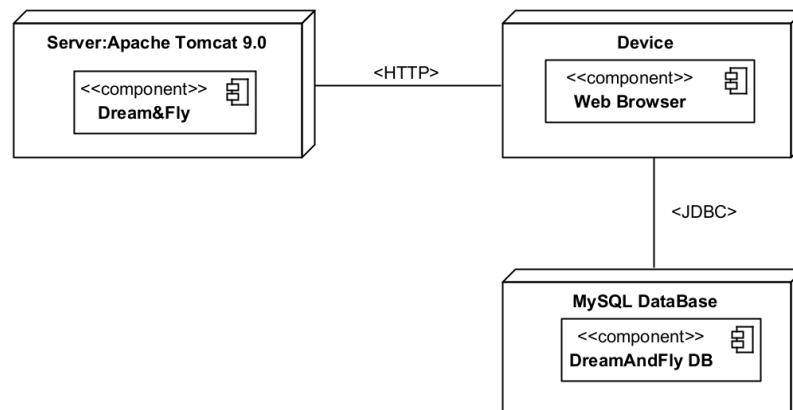
Lo Storage layer è costituito da un unico sottosistema:

- **Accesso ai dati persistenti:** si occupa di leggere e modificare i dati memorizzati sul Database.



### 3.3. Hardware/software mapping

Il sistema Dream&Fly è realizzato come una Web Application. Per poter interagire con il Sistema è necessario accedervi tramite un Web Browser. Il Web Browser comunicherà con Web Server Apache Tomcat attraverso il protocollo HTTP, che si occuperà pertanto di elaborare e rispondere a richieste dal client. La persistenza dei dati è invece mantenuta tramite l'utilizzo di un database MySQL, che sarà contattato dal Server tramite JDBC.



### 3.4. Persistent data management

Per la memorizzazione dei dati persistenti si è deciso di utilizzare un RDBMS (Relational Database Management System) poiché permette di accedere in modo semplice ed efficiente ai dati,

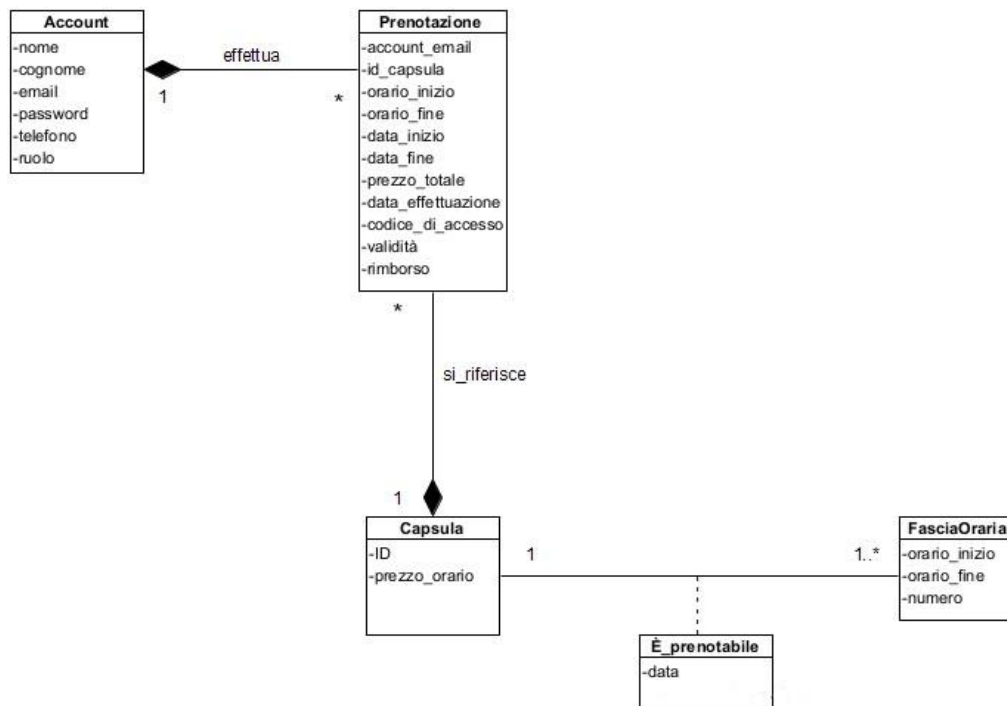
conservandone la consistenza, la privacy e l'affidabilità. Inoltre, possiede un meccanismo di permessi per cui utenti con operazioni diverse possono accedere a sezioni diverse della base di dati in maniera protetta.

### 3.4.1. Dizionario dei dati

Entità	Descrizione	Attributi	Identificatori
<b>Account</b>	Identità di un utente sulla piattaforma	<ul style="list-style-type: none"> <li>Nome</li> <li>Cognome</li> <li>E-mail</li> <li>Password</li> <li>Cellulare</li> <li>Ruolo</li> </ul>	E-mail
<b>Prenotazione</b>	Rappresenta la prenotazione che l'utente effettua in base alle proprie preferenze con tutti i relativi dati.	<ul style="list-style-type: none"> <li>Orario inizio</li> <li>Orario fine</li> <li>Data inizio</li> <li>Data fine</li> <li>Prezzo totale</li> <li>Data effettuazione</li> <li>Codice accesso</li> <li>Validità</li> <li>Rimborso</li> </ul>	Codice accesso
<b>Capsula</b>	Una capsula rappresenta una cabina aeroportuale disponibile per la prenotazione da parte degli utenti	<ul style="list-style-type: none"> <li>Id</li> <li>Prezzo orario</li> </ul>	Id
<b>Fascia oraria</b>	Intervallo di tempo per cui si rende prenotabile una capsula.	<ul style="list-style-type: none"> <li>Orario inizio</li> <li>Orario fine</li> <li>Numero</li> </ul>	Numero

Relazione	Descrizione	Entità coinvolte	Attributi
<b>Effettua</b>	L'account effettua una prenotazione	<ul style="list-style-type: none"> <li>Account (0, N)</li> <li>Prenotazione (1,1)</li> </ul>	
<b>Si riferisce</b>	La prenotazione si riferisce a una capsula	<ul style="list-style-type: none"> <li>Prenotazione (1,1)</li> <li>Capsula (0, N)</li> </ul>	
<b>È prenotabile</b>	La capsula è prenotabile in fasce orarie	<ul style="list-style-type: none"> <li>Capsula (1, N)</li> <li>Fascia oraria (0, N)</li> </ul>	Data





### 3.5. Access control and security

La sicurezza per l'accesso al Sistema è garantita tramite autenticazione con credenziali personali. Ciascun gestore ha delle credenziali che gli sono state assegnate al momento dell'assunzione e che cambierà al suo primo accesso; al contrario, gli utenti dovranno registrarsi in autonomia.

Nella matrice di accesso che segue riportiamo:

- nella prima colonna, gli attori del Sistema;
- nella prima riga, un'istanza delle classi del nostro Sistema;
- nelle celle risultanti dall'incrocio tra riga e colonna, abbiamo le operazioni che l'attore può compiere sull'oggetto in questione.

	Utente	Prenotazione	Capsula
Utente visitatore	<ul style="list-style-type: none"> <li>• Registrazione</li> </ul>		<ul style="list-style-type: none"> <li>• Ricercare capsule disponibili</li> </ul>
Utente registrato	<ul style="list-style-type: none"> <li>• Login</li> <li>• Logout</li> <li>• Visualizza/modifica dati</li> <li>• Cambio password</li> </ul>	<ul style="list-style-type: none"> <li>• Visualizza prenotazioni</li> <li>• Inserire dati pagamento</li> <li>• Effettuare prenotazioni</li> <li>• Eliminare prenotazione</li> </ul>	<ul style="list-style-type: none"> <li>• Ricercare capsule disponibili</li> </ul>
Gestore capsule	<ul style="list-style-type: none"> <li>• Login</li> <li>• Logout</li> <li>• Visualizza/modifica dati</li> <li>• Cambio password</li> </ul>		<ul style="list-style-type: none"> <li>• Registrare una capsula</li> <li>• Visualizzare elenco capsule</li> <li>• Modificare disponibilità</li> </ul>

			capsula <ul style="list-style-type: none"> <li>• Prolungare disponibilità capsula</li> <li>• Modifica prezzo</li> </ul>
Gestore account	<ul style="list-style-type: none"> <li>• Login</li> <li>• Logout</li> <li>• Visualizza/modifica dati</li> <li>• Cambio password</li> <li>• Visualizzare elenco account registrati</li> <li>• Registrare nuovo gestore</li> <li>• Eliminare account</li> </ul>		
Gestore prenotazioni	<ul style="list-style-type: none"> <li>• Login</li> <li>• Logout</li> <li>• Visualizza/modifica dati</li> <li>• Cambio password</li> </ul>	<ul style="list-style-type: none"> <li>• Visualizzare prenotazioni (anche con filtri)</li> </ul>	

### 3.6. Global software control

Il Sistema può essere definito come “event-driven”. Il WebServer attende le richieste dal WebBrowser. Alla ricezione di una richiesta, il WebServer la elabora e la invia alla servlet appropriata o JSP, risultando così un flusso di controllo basato sugli eventi.

### 3.7. Boundary conditions

**Avvio del Sistema.** Al momento dell'avvio, il Sistema necessita di un Web Server che fornisca il servizio di accesso ad un database MySQL per la gestione dei dati persistenti e l'interpretazione ed esecuzione del codice lato Server. Quando un utente accede al Sistema, gli verrà presentata una pagina nel Browser Web con le operazioni che gli è consentito svolgere a seconda del suo ruolo all'interno del sistema.

**Terminazione.** Alla chiusura dell'applicazione, il Sistema termina con un logout automatico, laddove necessario. Il Sistema non si occupa di salvare eventuali dati lasciati in sospeso durante l'utilizzo, sia che l'utente si trovasse nel lato amministrativo, sia che si trovasse nel lato utente. Pertanto, al riavvio del Sistema, quest'ultimo non ripresenterà i dati immessi in precedenza se questi non sono stati opportunamente salvati.

**Fallimento.** Si possono individuare diverse situazioni di fallimento:

- Nel caso di guasti dovuti al sovraccarico del database con successivo fallimento dello stesso, non è prevista alcuna procedura di salvataggio né di ripristino dei dati;
- Nel caso in cui si verifichi un'interruzione inaspettata dell'alimentazione, non sono previste procedure in grado di riportare il Sistema nello stato in cui era prima dello spegnimento;
- Se un utente invia al Sistema informazioni errate oppure l'utente non sottomette delle informazioni per la corretta esecuzione di un'operazione, il Sistema risponderà con un messaggio di errore;

## 4. SUBSYSTEM SERVICES

UtenteManagement	
Servizio	Descrizione
Registrazione	Permette di creare un nuovo account
Login	Permette di effettuare l'autenticazione
Logout	Permette di far disconnettere l'utente autenticato
Dati utente	Permette di visualizzare e modificare i dati personali all'utente
Cambio password	Permette di reimpostare la password
Recupera elenco account registrati	Permette di ottenere l'elenco degli account registrati.
Eliminazione account	Permette di eliminare un account
Recuperare un account	Permette di recuperare un account registrato
Registrare un nuovo gestore	Permette di registrare un nuovo gestore

CapsulaManagement	
Servizio	Descrizione
Registrazione capsula	Permette di inserire una nuova capsula
Visualizzare elenco capsule	Permette di visualizzare l'elenco di tutte le capsule registrate
Modifica disponibilità capsula	Permette di modificare la disponibilità di una capsula in una determinata fascia oraria (disponibile/non disponibile)
Prolunga disponibilità capsula	Permette di prolungare la disponibilità di una capsula
Modifica prezzo	Permette di modificare il prezzo orario di una capsula
Ricerca capsule disponibili	Permette di ricercare le capsule disponibili che un utente può prenotare

PrenotazioneManagement	
Servizio	Descrizione
Visualizzazione prenotazioni	Permette all'utente di visualizzare lo storico delle sue prenotazioni
Visualizzazione prenotazioni gestore	Permette al gestore di visualizzare l'elenco di tutte le prenotazioni
Inserimento dati pagamento	Permette all'utente di inserire i propri dati di pagamento
Effettua prenotazione	Permette all'utente di finalizzare la prenotazione
Elimina prenotazione	Permette all'utente di eliminare una prenotazione