

Università degli Studi di Salerno

Corso di Ingegneria del Software



**UStrike!**

SDD

Versione 2.0

Data: 19/01/2026

# Informazioni sul Progetto

**Progetto:** UStrike!

**Documento:** SDD

**Versione:** 2.0

**Data:** 19/01/2026

Coordinatore del progetto:

Nome Matricola

Partecipanti:

Nome	Matricola
Russo Serena	0512119098
Stefanile Andrea	0512119557
Valito Marcello	0512119014

Scritto da:

Russo Serena, Stefanile Andrea, Valito Marcello

## Revision History

Data	Versione	Descrizione	Autore
24/11/2025	0.1	Stesura del documento SDD	Russo Serena, Stefanile Andrea, Valito Marcello
25/11/2025	1.0	Rifinimento del SDD e consegna	Russo Serena, Stefanile Andrea, Valito Marcello
10/01/2026	1.1	Correzione della decomposizione in sottosistemi	Valito Marcello
11/01/2026	1.2	Rifinimento Access control list	Valito Marcello, Russo Serena
13/01/2026	1.3	Completamento Servizi	Serena Russo, Stefanile Andrea
15/01/2026	1.4	Revisione del documento	Russo Serena, Stefanile Andrea, Valito Marcello
19/01/2026	2.0	Correzioni finali	Valito Marcello

## Sommario

<b>1.0-Introduzione .....</b>	4
<b>1.1-Scopo del sistema .....</b>	4
<b>1.2-Obiettivi del sistema.....</b>	4
<b>1.3-Definizione, acronimi e abbreviazioni .....</b>	4
<b>1.4-Riferimenti .....</b>	4
<b>1.5-Panoramica.....</b>	4
<b>2.0-Architettura Software Attuale .....</b>	5
<b>3.0-Architettura Software Proposta .....</b>	5
<b>3.1-Panoramica .....</b>	5
<b>3.2-Decomposizione in sottosistemi.....</b>	5
<b>3.3-Mappatura Hardware/Software .....</b>	5
<b>3.4-Gestione dei dati persistenti.....</b>	6
<b>3.5-Controllo Accesso e Sicurezza.....</b>	6
<b>3.6-Controllo software globale.....</b>	9
<b>3.7-Condizioni di boundary.....</b>	9
<b>4.0-Servizi dei sottosistemi .....</b>	9
<b>5.0-Glossario .....</b>	10

## **1.0-Introduzione**

### **1.1-Scopo del sistema**

Nel settore dell'intrattenimento, strutture come sale da bowling e arcade si affidano ancora a metodi di gestione tradizionali, quali prenotazioni telefoniche e agende cartacee. Questo approccio manuale presenta diverse criticità: è soggetto a errori umani (prenotazioni sovrapposte o registrate in maniera errata), risulta inefficiente durante le ore di punta e offre un'esperienza al cliente poco moderna e frammentata. Manca una piattaforma digitale unificata che possa semplificare e automatizzare la gestione delle prenotazioni per i vari servizi offerti (bowling, biliardo, go-kart).

L'obiettivo del progetto UStrike! è colmare questa lacuna, creando un sistema centralizzato che migliori l'efficienza operativa dello staff e l'esperienza complessiva del cliente, riducendo gli errori e ottimizzando l'interazione tra l'arcade e il suo pubblico.

### **1.2-Obiettivi del sistema**

Il sistema “UStrike!” è stato progettato considerando i seguenti obiettivi di design:

#### **1.2.1-Criteri di performance**

- **OP1.0:** Il sistema deve essere fruibile tramite Web da qualunque dispositivo (desktop o mobile) con connettività a Internet.
- **OP2.0:** il sistema deve garantire tempi di risposta brevi in condizioni operative ordinarie; in presenza di picchi di connessioni simultanee è ammesso un aumento contenuto dei tempi di risposta, senza degradare la fruibilità del servizio.

#### **1.2.2-Criteri di affidabilità**

- **OA1.0:** Il sistema deve essere in grado di continuare a funzionare anche in caso di problemi con un componente o durante picchi moderati di traffico.
- **OA2.0:** Le password degli utenti non vengono mai memorizzate in chiaro; il sistema applica un algoritmo di hashing dedicato alle password e persiste esclusivamente l'hash risultante. I dati sono protetti mediante misure di sicurezza adeguate.

#### **1.2.3-Criteri di manutenzione**

- **OM1.0:** Il sistema deve essere progettato e realizzato in modo da garantire un'elevata manutenibilità, consentendo modifiche e interventi di manutenzione futuri in maniera semplice e con impatto minimo sul resto del sistema.

## **1.3-Definizione, acronimi e abbreviazioni**

- RAD: Requirements Analysis Document
- SDD: System Document Design
- Utente: Termine generico per indicare un qualsiasi individuo che interagisce con la piattaforma.
- Cliente: Utente Registrato a UStrike!
- Staff: Membro del team di UStrike!
- Manager: Capo dello staff.
- Servizio: Uno dei tre servizi principali di UStrike! (Bowling, Biliardo, Go-Kart)

## **1.4-Riferimenti**

Per la stesura del presente documento si fa riferimento alla terminologia e ai concetti definiti nel *Problem Statement* e nel *Requirement Analysis Document* del progetto UStrike!.

## **1.5-Panoramica**

Questo documento illustra le principali decisioni di design e le scelte tecniche adottate per il sistema UStrike. Per una descrizione più generale del contesto e del problema si rimanda al Problem Statement, mentre per l'analisi dei requisiti e delle funzionalità previste si fa riferimento al Requirement Analysis Document.

Inoltre, il documento presenta la scomposizione del sistema in sottosistemi e la relativa mappatura hardware/software; gli aspetti relativi ai dati persistenti sono approfonditi nel documento DatiPersistenti\_UStrike.

## 2.0-Architettura Software Attuale

Poiché UStrike! si propone di creare un'architettura software per l'arcade fisico non è presente alcuna architettura software precedente.

## 3.0-Architettura Software Proposta

### 3.1-Panoramica

Il sistema UStrike! adotta un'architettura MVC, tipica dei sistemi web. I sottosistemi principali previsti dall'architettura MVC sono i seguenti:

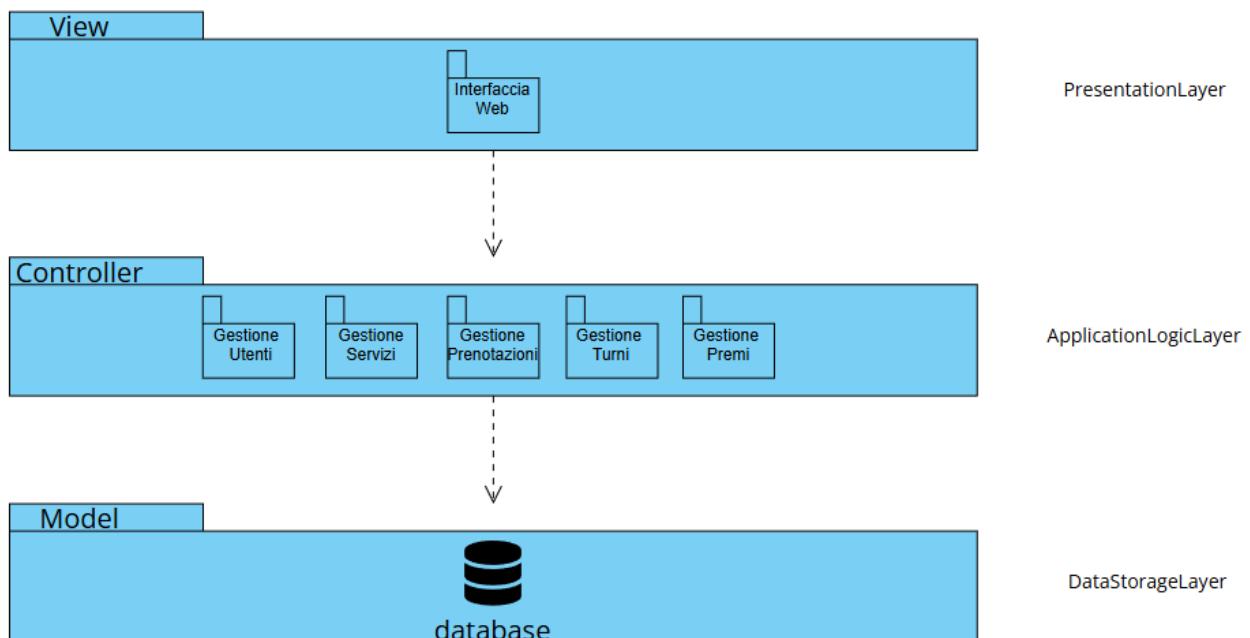
Model: rappresenta il sistema di gestione dei dati e della logica di business.

View: rappresenta il sistema di interazione diretta con l'utente, è composta da tutte le interfacce che l'utente è in grado di vedere e con le quali è in grado di interagire.

Controller: in questo sottosistema sono presenti tutte le componentistiche che elaborano i dati, si occupa anche delle interazioni tra View e Model.

### 3.2-Decomposizione in sottosistemi

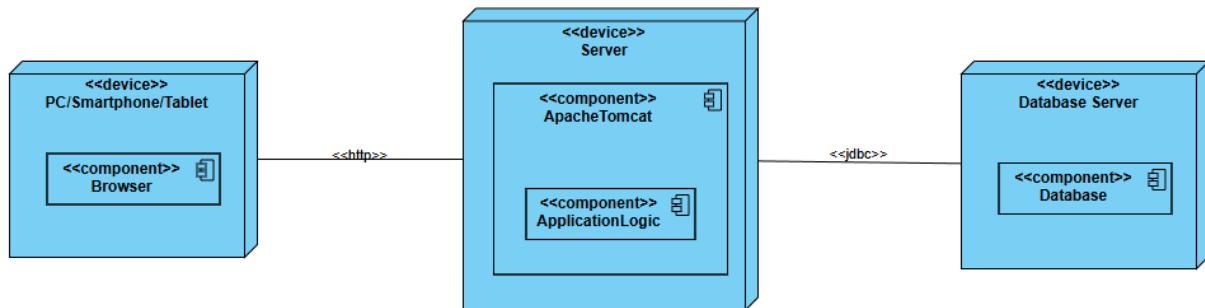
- View → PresentationLayer
- Controller → DataManagementLayer/Application Logic
- Model → DataStorageLayer



### 3.3-Mappatura Hardware/Software

Il sistema utilizza un'architettura Client/Server. Il Web Server è rappresentato da Apache Tomcat 11 ed è situato su una singola macchina, la logica del sistema è composta da Java Servlet mentre l'interfaccia utente è realizzata tramite l'utilizzo di pagine JSP. Il client è rappresentato dal Web Browser utilizzato dall'utente. La comunicazione tra i nodi viene effettuata tramite richieste e risposte HTTP tra client e server mentre tra

server e database verranno effettuate delle query JDBC.



### 3.4-Gestione dei dati persistenti

Per la descrizione dei file/dati persistenti si rimanda al documento “DatiPersistenti\_UStrike!”. Il database del sistema è stato progettato e realizzato in MySQL, mediante la definizione dello schema relazionale tramite istruzioni SQL.

### 3.5-Controllo Accesso e Sicurezza

#### UTENTI

	Guest	Cliente	Staff	Manager
Registrazione	✓	✗	✗	✗
Login	✗	✓	✓	✓
Logout	✗	✓	✓	✓
Visualizza Profilo	✗	✓	✓	✓
Modifica Profilo	✗	✓	✗	✓
Cambio Password	✗	✓	✓	✓

#### PRENOTAZIONE

	Guest	Cliente	Staff	Manager

Prenotazione Bowling	✗	✓	✗	✗
Prenotazione Go-Kart	✗	✓	✗	✗
Annulla Prenotazione	✗	✓	✗	✗
Rifiuta Bowling	✗	✗	✓	✗
Rifiuta Go- Kart	✗	✗	✓	✗
Accetta Bowling	✗	✗	✓	✗
Accetta Go- Kart	✗	✗	✓	✗

#### TURNO CLASS

	Guest	Cliente	Staff	Manager
Crea Turnistica Bowling	✗	✗	✗	✓
Crea Turnistica Go-Kart	✗	✗	✗	✓
Crea Turnistica Biliardo	✗	✗	✗	✓
Modifica Turnistica Bowling	✗	✗	✗	✓

Modifica Turnistica Go-Kart	✗	✗	✗	✓
Modifica Turnistica Biliardo	✗	✗	✗	✓
Visualizza Turnistica Bowling	✗	✗	✓	✓
Visualizza Turnistica Go-Kart	✗	✗	✓	✓
Visualizza Turnistica Biliardo	✗	✗	✓	✓

#### PREMIO

	Guest	Cliente	Staff	Manager
Aggiungi Premio	✗	✗	✓	✗
Rimuovi Premio	✗	✗	✓	✗
Modifica Premio	✗	✗	✓	✗
Catalogo Premi	✓	✓	✓	✓
Visualizza premio			✓	✓

## SERVIZIO

	Guest	Cliente	Staff	Manager
Servizio Bowling	✓	✓	✓	✓
Servizio Go-Kart	✓	✓	✓	✓
Servizio Biliardo	✓	✓	✓	✓
Abilità Servizio	✗	✗	✓	✗
Disabilità Servizio	✗	✗	✓	✗

## 3.6-Controllo software globale

Poiché UStrike! è un'applicazione web organizzata secondo il pattern MVC, le richieste HTTP inviate dal browser vengono ricevute e gestite da Apache Tomcat, che le instrada verso il componente di controllo appropriato in base alle associazioni URL definite. Il Controller (implementato tramite Servlet) coordina l'elaborazione della richiesta, invocando quando necessario il Model (servizi applicativi e componenti di accesso ai dati) per recuperare o aggiornare le informazioni richieste. Completata l'elaborazione, il Controller seleziona la View più opportuna (pagina JSP) e le fornisce i dati da presentare; la JSP viene quindi renderizzata in HTML e la risposta viene restituita al client per la visualizzazione.

## 3.7-Condizioni di boundary

Le condizioni di boundary riguardanti accensione e spegnimento del sistema sia da lato server vengono qui di seguito riportate:

- Avvio del sistema: Con l'accensione del sistema vengono avviati anche web server e database già popolato.
- Spegnimento del sistema: Quando il software viene chiuso si ha la terminazione del sistema con il logout dell'utente. Per quanto riguarda la terminazione del server essa viene delegata al manager. Una volta terminato il server nessun client potrà più accedere e il server dovrà garantire la terminazione di tutte le transazioni in corso (in caso di terminazione imprevista).

## 4.0-Servizi dei sottosistemi

Gestore Utenti	
Login	Permette all'utente di accedere al sistema e svolgere tutte le operazioni a lui consentite a patto che acceda con le giuste credenziali.

Registrazione	Consente all'utente di registrare un account nel sistema.
Password dimenticata	Consente all'utente di accedere all'account con una password provvisoria fornita dal sistema.
Visualizza informazioni personali	Permette all'utente di visualizzare le proprie informazioni personali.
Logout	Permette all'utente di uscire dall'account presente in quel momento sul sistema lato client.
Modifica password	Permette all'utente di modificare la propria password inserita nel sistema.
Modifica profilo	Permette all'utente di modificare i propri dati personali.

Gestione Prenotazioni	
Effettua prenotazione	Permette all'utente di effettuare una prenotazione per un dato servizio ad un dato giorno ed orario.
Accetta prenotazione	Permette all'utente dello staff di accettare una prenotazione effettuata da un utente.
Rifiuta prenotazione	Permette all'utente dello staff di rifiutare una prenotazione effettuata da un utente.
Visualizza catalogo prenotazioni	Permette all'utente di vedere lo storico delle prenotazioni effettuate e in quale stato sono.
Ordina catalogo prenotazioni	Permette all'utente staff di filtrare lo storico delle prenotazioni
Annulla Prenotazione	Permette all'utente di cancellare una prenotazione che è nello stato "in Attesa"

Gestione Turnistica	
Visualizza tabella turni settimanale	Permette di vedere la tabella turni settimanale.
Assegna turno	Permette all'utente manager di assegnare un turno a un user staff.
Modifica turno	Permette all'utente manager di modificare l'assegnazione di un turno a un user staff

Gestione Premi	
Aggiungi premi	Permette di aggiungere un premio nel sistema.
Rimuovi premio	Permette di rimuovere un premio dal sistema.
Visualizza premio	Permette di vedere un singolo premio nel dettaglio.
Modifica premio	Permette di modificare le informazioni relative ad un determinato premio.
Visualizza catalogo premi	Permette all'utente/staff di visualizzare l'elenco di tutti i premi con i loro prezzi in ticket.

Gestione Servizi	
Visualizza servizio	Permette all'utente di visualizzare un servizio.
Disabilita servizio	Permette all'utente staff di rendere lo stato di un servizio "disabilitato".
Abilita servizio	Permette all'utente staff di rendere lo stato di un servizio "abilitato".

## 5.0-Glossario

- MySQL: database service utilizzato da UStrike! per la gestione dei dati.

- DBMS: è un software che permette di creare, gestire, e interrogare database in modo efficiente e sicuro.
- Client: componente che accede ai servizi del server.
- Server: componente che fornisce servizi ai client.
- JSP: tecnologia che permette di creare pagine web dinamiche.
- Servlet: programma java eseguito su un web server per elaborare richieste da parte dei client.