

Università degli Studi di Salerno
Corso di Ingegneria del Software

**EduCat
System Design Document
Versione 1.0**



EduCat

Data: 25/11/2025

Progetto: EduCat	Versione: 1.0
Documento: System Design	Data: 25/11/2025

Coordinatore del progetto:

Nome	Matricola
Isabella Dima	0512119719

Partecipanti:

Nome	Matricola
Isabella Dima	0512119719
Anna Chiara Esposito	0512119143

Scritto da:	Isabella Dima
-------------	---------------

Revision History

Data	Versione	Descrizione	Autore
25/11/2025	1.0	System Design Document	Isabella Dima

Indice

1.	Introduzione.....	4
	<i>1.1 Purpose del Sistema</i>	4
	<i>1.2 Design Goals</i>	4
	<i>1.3 Definizioni, acronimi e abbreviazioni</i>	4
	<i>1.4 References</i>	4
	<i>1.5 Overview</i>	5
2.	Current System	5
3.	Proposed Software Architecture	5
	<i>3.1 Overview</i>	5
	<i>3.2 Subsystem Decomposition</i>	5
	<i>3.3 Hardware/Software Mapping</i>	7
	<i>3.4 Persistent Data Management</i>	8
	<i>3.5 Access Control and Security</i>	9
	<i>3.6 Global Software Control</i>	10
	<i>3.7 Boundary Conditions</i>	10
	3.7.1. <i>Start up</i>	10
	3.7.2 <i>Shut down</i>	10
	3.7.3 <i>Failures</i>	11
4.	Subsystem Services	12
	<i>4.1 Gestione Utenza</i>	12
	<i>4.2 Gestione Lezione</i>	13
	<i>4.3 Recensione</i>	13
	<i>4.4 Segnalazione</i>	13
5.	Glossary	14

1. Introduzione

1.1 Purpose del Sistema

EduCat è una piattaforma web di prenotazione e acquisto di lezioni private rivolta a tutor di diverse discipline, studenti di qualsiasi grado di istruzione e genitori.

L'obiettivo del sistema è offrire a studenti e genitori (in caso di studenti minorenni) un modo semplice per confrontare i tutor in base a preparazione, modalità di insegnamento, fascia di prezzo e disponibilità, così da prenotare lezioni secondo le proprie esigenze e possibilità economiche. I tutor potranno registrarsi sulla piattaforma e offrire ripetizioni a pagamento.

1.2 Design Goals

Performance	Il sistema deve garantire tempi di risposta sotto i 2 secondi per il 90% delle richieste utente al fine di evitare frustrazione da parte dell'utente e l'abbandono del sito.
Affidabilità	Il sistema deve garantire l'integrità dei dati durante le transazioni, in maniera che esse vengano annullate se non sono completate.
Sicurezza	La piattaforma adotta misure di sicurezza quali connessione criptata HTTPS, query parametrizzate, hashing di password e obbligo di password forte.
Portabilità	Il sistema deve essere accessibile su diverse piattaforme utilizzando interfacce e protocolli standardizzati e utilizzando la programmazione modulare.
Usabilità	Il sistema deve essere mobile friendly e offrire un'interfaccia responsive. L'utente medio deve essere in grado di effettuare la registrazione, cercare una lezione e completare l'acquisto in meno di 4 minuti.
Modularità e Manutenibilità	Il sistema adotta pattern MVC per separare la logica di business, la presentazione e l'accesso ai dati persistenti per migliore manutenibilità e maggiore facilità nel debugging.

1.3 Definizioni, acronimi e abbreviazioni

Studenti: utenti che ricercano lezioni private per migliorare la propria preparazione scolastica o universitaria.

Genitori: nel caso di studenti minorenni, i genitori agiscono come referenti per la prenotazione e il pagamento delle lezioni.

Tutor: figure abilitate a offrire lezioni private. Possono registrarsi, gestire il proprio profilo, definire tariffe e disponibilità, e ricevere feedback dagli studenti.

1.4 References

- ProblemStatement_EduCat v1.1
- RAD_EduCat v1.1
- Bernd Bruegge & Allen Dutoit - Object-Oriented Software Engineering: Using UML, Patterns, and Java

1.5 Overview

Il documento è diviso in quattro sezioni:

- **Current System:** illustrazione di architetture di sistemi simili a quello proposto.
- **Proposed Software Architecture:** modello di system design per il nuovo sistema.
- **Subsystem Services:** descrizione di start-up, shutdown e comportamento in caso di errori del sistema.
- **Glossario:** lista di termini utilizzati nel documento con relative definizioni.

2. Current System

Attualmente i sistemi con funzionalità simili a EduCat sono applicazioni web realizzate con pattern MVC e con modello Client/Server.

Caratteristiche comuni sono l'utilizzo di database relazionali, interfacce web responsive e mobile friendly e un sistema di recensioni.

Tuttavia alcuni sistemi mostrano limiti quali scarsa ottimizzazione della ricerca delle lezioni, impossibilità di ricoprire più di un ruolo e mancanza di trasparenza nei processi di segnalazione utenti.

3. Proposed Software Architecture

3.1 Overview

La piattaforma “EduCat” è un’applicazione web basata su architettura Client/Server e design pattern MVC. Così facendo la modularità del sistema permette una maggiore leggibilità e manutenibilità. Il Web Server e Container scelto per il sistema è Apache Tomcat.

Sviluppo del back-end:

- Java23
- Eclipse

Sviluppo del front-end:

- HTML5
- CSS3
- JavaScript
- JSP

Persistenza dei dati:

- JDBC
- MySQL

3.2 Subsystem Decomposition

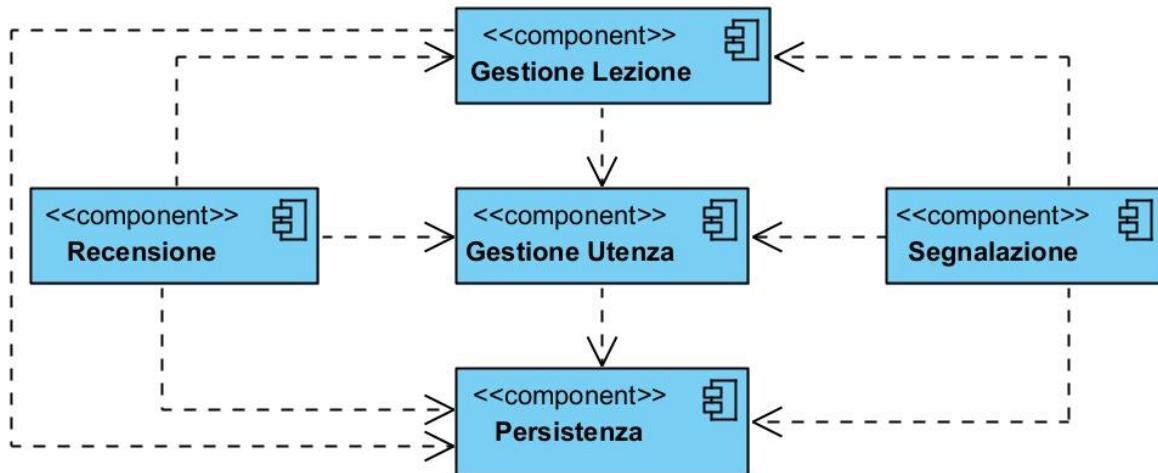
Il sistema viene quindi suddiviso nei seguenti sottosistemi:

- **Gestione Utenza:** si occupa di gestire le funzioni di login, logout e si registrazione per gli utenti di tipo “Studente”, “Genitore” e “Tutor”, visualizzazione, modifica e cancellazione

EduCat	Ingegneria del Software	Pagina 5 di 14
--------	-------------------------	----------------

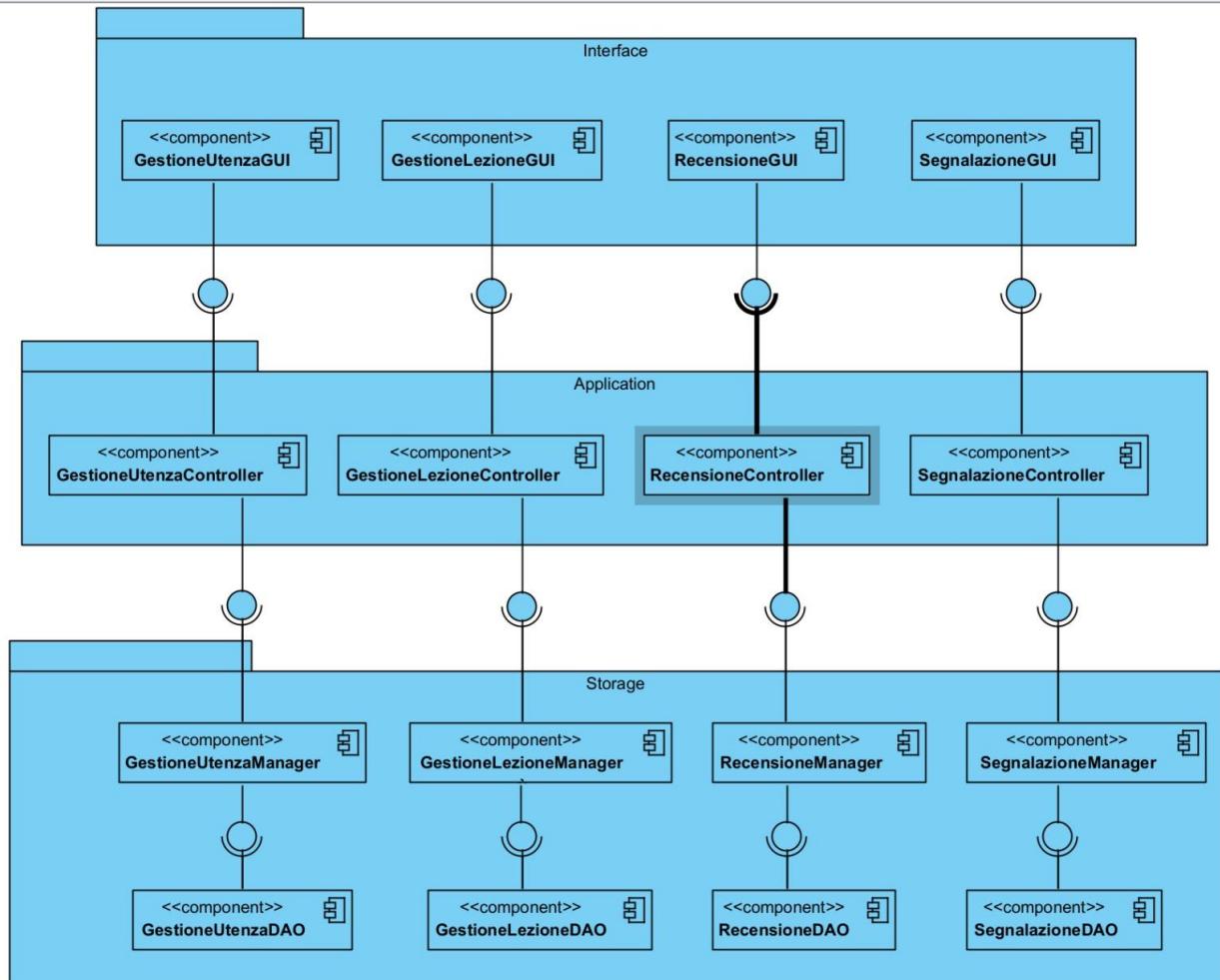
account e cambio ruolo attivo per gli utenti multiruolo. Per gli admin è resa disponibile la funzione di ricerca e cancellazione degli utenti.

- **Gestione Lezioni:** è responsabile della ricerca, visualizzazione delle informazioni, prenotazione e acquisto delle lezioni. Permette anche di visualizzare le lezioni acquistate (sia da parte dello studente/genitore che tutor) ed eventualmente annullare la prenotazione.
- **Gestione Recensioni:** si occupa di inserire e visualizzare le recensioni e calcolare una media delle valutazioni relative al tutor.
- **Gestione Segnalazioni:** è responsabile della lettura e gestione delle segnalazioni e l'eventuale invio di messaggi di avvertimento per il singolo utente.
- **Persistenza:** si occupa di gestire la persistenza dei dati con un database.



Di seguito una vista dettagliata di ciascun sottosistema evidenziando le componenti principali:

- **GUI:** Graphic User Interface, che contiene le varie view che saranno reindirizzate per creare le pagine web da mostrare al cliente.
- **Controller:** si occupa della logica per il controllo del sistema.
- **Manager:** si occupa della logica di business.
- **DAO:** Data Access Object, implementazione dell'omonimo pattern architetturale che si occupa di fornire un accesso astratto ai dati persistenti



3.3 Hardware/Software Mapping

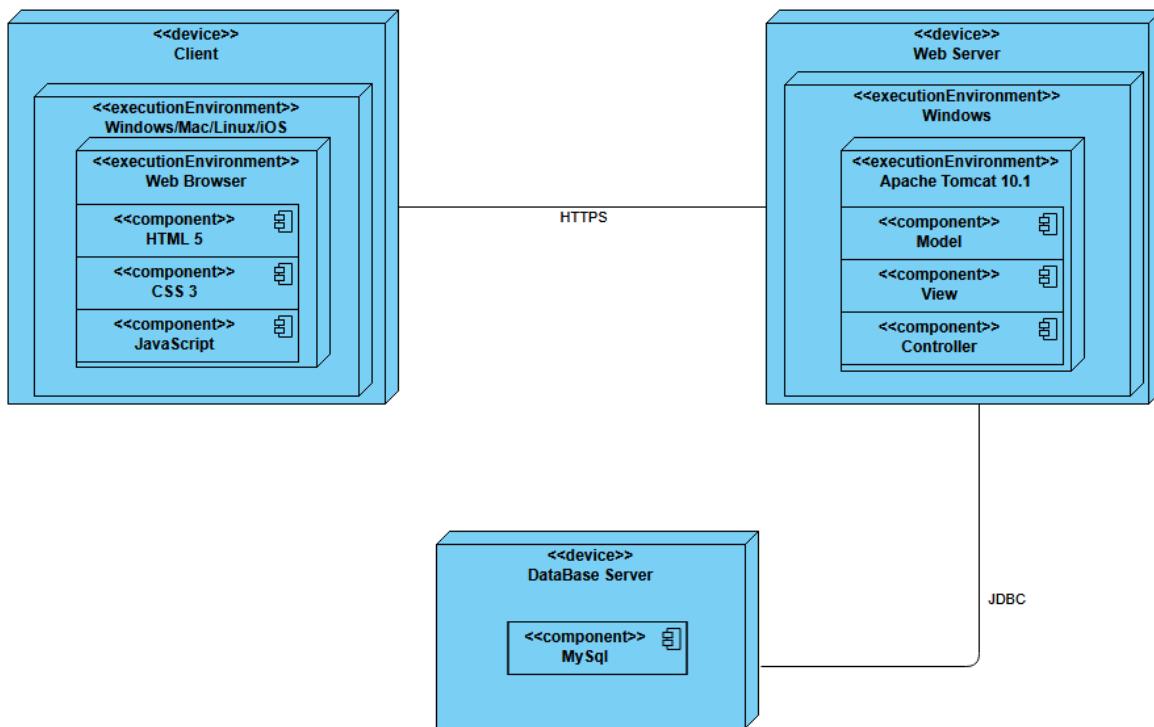
Il sistema si basa su Architettura Client/Server con pattern MVC (Model-View-Controller) formata da un unico server a cui vengono inoltrate le richieste di più client.

Ogni Client connesso a Internet può inviare richieste al Web Server con, per maggiore sicurezza, protocollo HTTPS attraverso il proprio Web Browser.

Il Web Server, eseguito su Windows con Apache Tomcat, risponde alle richieste di ciascun Client.

Il Web Server comunica con il DataBase per interrogarlo, aggiungere o rimuovere dati tramite driver JDBC.

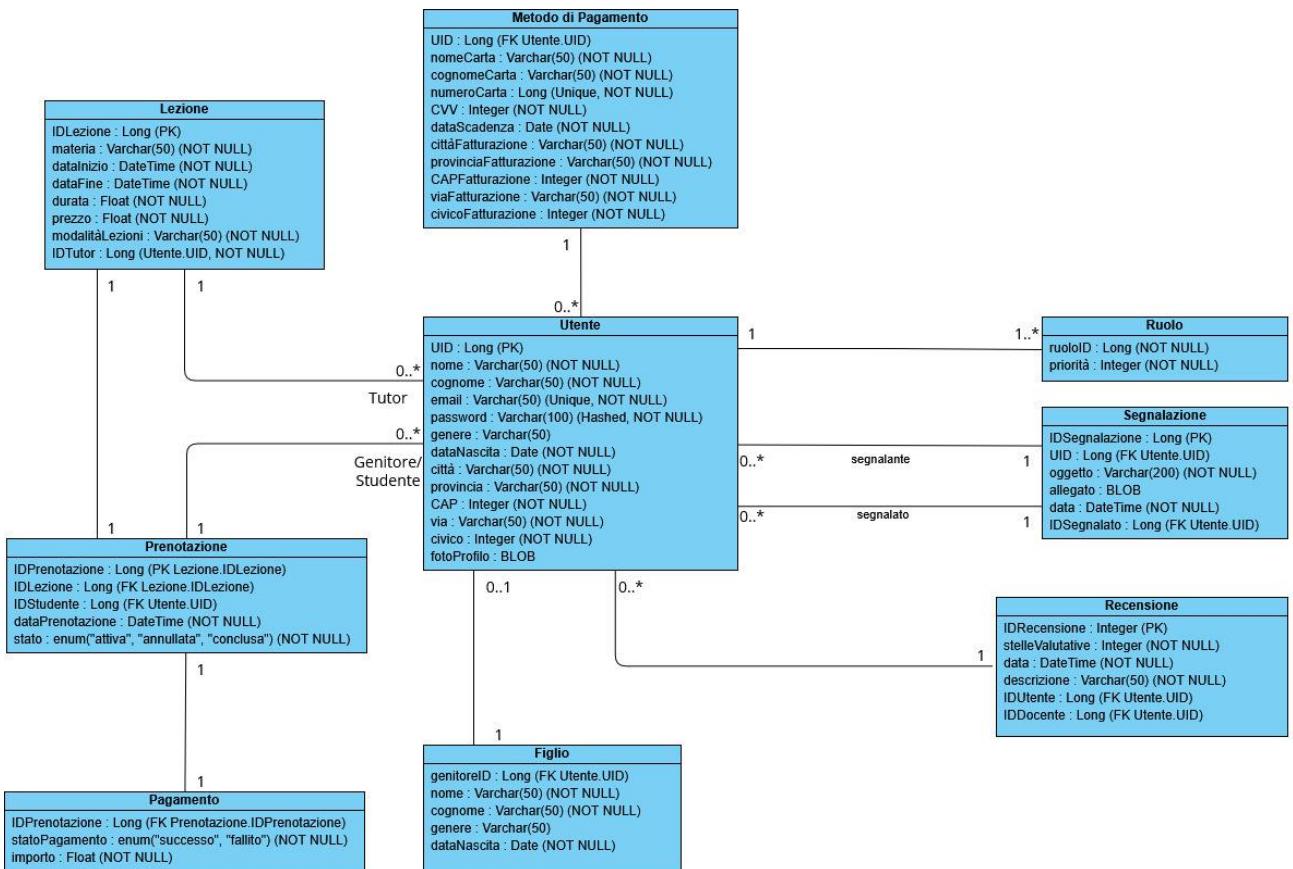
La gestione dei dati persistenti è stata affidata a MySQL per ragioni di sicurezza, affidabilità e performance.



3.4 Persistent Data Management

Per la gestione dei dati persistenti è stato selezionato MySQL, un database relazionale. I motivi di questa scelta sono la facilità di utilizzo, l'integrità e accuratezza dei dati, sicurezza e performance.

Il grafico sottostante rappresenta ciascuna entità all'interno del database con relativi attributi, Primary Key (PK), Foreign Key (FK) e relazioni.



3.5 Access Control and Security

Operazione\Attore	Studente/Genitore	Tutor	Amministratore Utenti
Gestione Utenza	<ul style="list-style-type: none"> • Login • Logout • Iscrizione • Visualizzazione Profilo • Modifica informazioni personali • Modifica metodo di pagamento • Modifica password • Cancellazione account • Cambio ruolo attivo 	<ul style="list-style-type: none"> • Login • Logout • Iscrizione • Visualizzazione Profilo • Modifica informazioni personali • Modifica metodo di pagamento • Modifica password • Cancellazione account • Cambio ruolo attivo 	<ul style="list-style-type: none"> • Login • Logout • Visualizzazione Profilo • Modifica password • Cambio ruolo attivo • Visualizza utenti registrati • Ricerca per nome • Cancella utenti
Gestione Lezioni	<ul style="list-style-type: none"> • Visualizza informazioni lezione • Ricerca lezioni 	<ul style="list-style-type: none"> • Modifica lezioni offerte • Visualizza lezioni • Riscossione 	-

	<ul style="list-style-type: none"> • Acquisto lezioni • Annullamento lezione • Visualizza lezioni acquistate 	<ul style="list-style-type: none"> • pagamento • Cancella lezioni previste 	
Recensioni	<ul style="list-style-type: none"> • Scrivi recensione 	<ul style="list-style-type: none"> • Visualizza tutte le recensioni 	-
Segnalazioni	<ul style="list-style-type: none"> • Segnalazione utenti 	<ul style="list-style-type: none"> • Segnalazione utenti 	<ul style="list-style-type: none"> • Lettura segnalazioni • Risposta segnalazioni • Invio messaggio di avvertimento

3.6 Global Software Control

Il sistema si basa sull’interazione dell’utente con l’interfaccia grafica.

Le richieste vengono avviate attraverso eventi innescati dall’interazione dell’utente con l’interfaccia, che verrà implementata tramite JSP.

La logica di controllo viene delegata alle Servlet che ricevono le richieste, le elaborano e inviano una risposta visibile al client.

3.7 Boundary Conditions

3.7.1. Start up

Il sistema EduCat all’avvio mostrerà una pagina di autenticazione in cui verranno richieste le credenziali di autenticazione(email-password), le quali permetteranno all’utente di poter accedere alla Home Page e a tutte le funzionalità offerte.

UCBC1: Avvio del Sistema

Descrizione: Lo UC descrive che l’Amministratore avvia il sistema.

Attore principale: Amministratore.

Attori secondari: /

Entry condition: L’amministratore accede al server.

Exit condition - On success: Il sistema viene avviato correttamente.

Exit condition - On failure: Il sistema non viene avviato.

Flusso di Eventi:

1. Amministratore: Esegue il comando per inizializzare e istanziare tutte le funzionalità e risorse del sistema.
2. Sistema: Il sistema EduCat avvia tutti i suoi servizi e rende le funzionalità disponibili agli utenti.

3.7.2 Shut down

Il sistema EduCat potrà essere spento solo se i relativi sottosistemi sono inattivi e sono stati salvati i dati persistenti, nel caso in cui questo non dovesse verificarsi il sistema non sarà spento.

UCBC2: Spegnimento del sistema

Descrizione: Lo UC descrive che l'amministratore può spegnere il sistema.

Attore principale: Amministratore.

Attori secondari: /

Entry condition: L'amministratore è loggato.

Exit condition - On success: Il sistema viene spento.

Exit condition - On failure: Il sistema non viene spento.

Flusso di eventi principale:

1. Amministratore: Esegue un comando per l'arresto del sistema.
2. Sistema: Il sistema esegue un controllo per verificare se sono presenti client connessi ed effettua il salvataggio dei dati persistenti. Se non ci sono client il sistema viene spento.

Flusso alternativo a:

Sistema: Notifica all'amministratore la presenza di client connessi e il sistema ritenta lo spegnimento dopo un certo lasso di tempo. Continua a ritentare fino al momento dello spegnimento.

Flusso alternativo b:

Sistema: Disconnette i client ed esegue lo spegnimento.

Flusso alternativo c:

Sistema: Il salvataggio dei dati persistenti non è andato a buon fine. Notifica l'amministratore dell'errore e non procede all'arresto del server.

3.7.3 Failures

Il sistema EduCat potrebbe andare incontro a fallimenti, per minimizzare i rischi di fallimento, il sistema è periodicamente soggetto a dei backup.

UCBC3: Fallimento del Sistema Identificativo

Descrizione: Lo UC descrive il fallimento software del sistema.

Attore principale: Amministratore.

Attori secondari: /

Entry condition: Il sistema inaspettatamente smette di funzionare.

Exit condition - On success: Il sistema viene riavviato con il backup più recente.

Exit condition - On failure: Il sistema non viene riavviato correttamente.

Flusso di eventi principale:

1. Amministratore: Esegue il comando per il ripristino del sistema.
2. Sistema: Include UCBC1 (Avvio del sistema).

Flusso alternativo a:

Sistema: Il sistema non viene avviato correttamente e le funzionalità non sono disponibili agli utenti.

Amministratore: apre un ticket verso il provider

Il sistema EduCat possa avere problemi di accesso ai dati persistenti, oppure che quest'ultimi possono essere corrotti.

UCBC4: Errore di accesso ai dati persistenti

Descrizione: Lo UC descrive il comportamento dell'utente in caso di errore di accesso ai dati persistenti da parte del sistema.

Attore principale: Amministratore

Attori secondari: /

Entry condition: Il sistema non riesce ad accedere ai dati persistenti oppure i dati risultano corrotti.

Exit condition - On success: Il Sistema riprende il normale funzionamento

Exit condition - On failure: Il Sistema non riprende il normale funzionamento

Flusso di eventi principale:

1. Sistema: Notifica all'amministratore problemi di accesso ai dati.
2. Amministratore: Include UCBC2 (Spegnimento del sistema)
3. Amministratore: Esegue un ripristino di accesso ai dati persistenti.
4. Amministratore: Include UCBC1 (Avvio del sistema).

4. Subsystem Services

4.1 Gestione Utenza

<i>Servizio</i>	<i>Descrizione</i>	<i>Interfaccia</i>
Login	Questa funzionalità permette ad un utente registrato di accedere al sistema inserendo le proprie credenziali, verificandone la correttezza.	GestioneUtenzaManager
Registrazione	Questa funzionalità permette ad un utente (Studente, Genitore o Tutor) di creare un nuovo account sulla piattaforma inserendo i dati anagrafici richiesti e le credenziali.	GestioneUtenzaManager
Logout	Questa funzionalità permette all'utente di terminare la propria sessione.	GestioneUtenzaManager
Cancellazione Account	Questa funzionalità permette all'utente di cancellare permanentemente il proprio account dal sistema e i dati associati.	GestioneUtenzaManager
Modifica Profilo	Questa funzionalità permette all'utente di visualizzare e modificare le informazioni presenti sul proprio profilo, inclusi i dati di pagamento e password.	GestioneUtenzaManager
Cambio Ruolo Attivo	Questa funzionalità permette agli utenti con ruoli multipli (es: Genitore e Tutor) di passare da un'interfaccia di ruolo a un'altra senza dover effettuare nuovamente il login.	GestioneUtenzaManager
Gestione Utente (Admin)	Questa funzionalità permette agli admin di visualizzare e ricercare tutti gli utenti iscritti alla piattaforma e cancellare account in caso di violazioni.	GestioneUtenzaManager

4.2 Gestione Lezione

<i>Servizio</i>	<i>Descrizione</i>	<i>Interfaccia</i>
Ricercare Lezioni	Questa funzionalità permette a Studenti e Genitori di filtrare le lezioni disponibili in base a materia, città, prezzo, modalità (online/ in presenza) e disponibilità di orario.	GestioneLezioneManager
Gestione Annunci	Questa funzionalità consente ai Tutor di creare, modificare e rimuovere le informazioni relative agli annunci, tariffe e slot di disponibilità di orario.	GestioneLezioneManager
Prenotazione e Acquisto	Questa funzionalità permette a Studenti e Genitori di prenotare la lezione scelta, specificando lo slot di orario, inserire i dati di pagamento e procedere al checkout finalizzando l'acquisto.	GestioneLezioneManager
Cancellazione Lezione	Questa funzionalità permette a Utenti e Tutor di annullare una lezione prenotata (rispettando i vincoli temporali) e innescando l'eventuale procedura di rimborso.	GestioneLezioneManager
Visualizzazione Storico	Questa funzionalità permette agli Utenti di visualizzare lo storico delle lezioni passate ed eventuali lezioni programmate.	GestioneLezioneManager

4.3 Recensione

<i>Servizio</i>	<i>Descrizione</i>	<i>Interfaccia</i>
Inserimento Recensione	Questa funzionalità permette a Studenti o Genitori di lasciare una valutazione (1-5 stelle) e un eventuale commento testuale per valutare un Tutor una volta terminata la lezione.	RecensioneManager
Visualizzazione Recensioni	Questa funzionalità permette agli Utenti di visualizzare le recensioni relative ad un Tutor per valutare le competenze.	RecensioneManager

4.4 Segnalazione

<i>Servizio</i>	<i>Descrizione</i>	<i>Interfaccia</i>
Invia Segnalazione	Questa funzionalità permette ad un Admin di segnalare un Utente sospetto di comportamenti scorretti, inserendo una descrizione ed eventuali allegati.	SegnalazioneManager
Gestione Segnalazione	Questa funzionalità permette ad un Admin di visualizzare la lista di segnalazioni ricevute da parte di Utenti, leggerne i dettagli e inviare risposte di aggiornamento all'utente segnalante.	SegnalazioneManager

5. Glossary

<i>Termine</i>	<i>Significato</i>
MVC	Model-View-Controller: pattern di architettura software che separa la logica di business, la presentazione e l'accesso ai dati persistenti per migliore manutenibilità e maggiore facilità nel debugging.
DMBS	DataBase Management System: software per la gestione di basi di dati. Consente di interrogare, la base di dati, inserire e modificare i dati al suo interno.