

PROBLEM STATEMENT

Sommario

1. Dominio del problema.....	2
1.1 Problema	2
1.2 Scopo del sistema.....	2
1.3 Ambito del sistema	2
1.4 Definizioni, acronimi e abbreviazioni.....	2
2. Scenari	3
2.1 SC_Inserisci corso d'insegnamento	3
2.2 SC_Elimina corso d'insegnamento.....	3
2.3 SC_Invio domanda	4
2.4 SC_Visualizza Risposte	5
3. Requisiti Funzionali.....	5
3.1 Gestione Corsi d'insegnamento	5
➤ RF_GCI_1 – Inserisci insegnamento.	5
➤ RF_GCI_2 – Elimina insegnamento.	5
3.2 Gestione Domande.....	6
➤ RF_GD_1 – Invio domanda.....	6
➤ RF_GD_2 – Visualizza risposte	6
4. Requisiti non funzionali.....	6
4.1 RNF_1: Usabilità.	6
4.2 RNF_2: Affidabilità	6
4.3 RNF_3: Performance.....	6
4.4 RNF_4: Supportabilità.	7
4.5 RNF_5: Manutenibilità.....	7
4.6 RNF_7: Packaging	7

1. Dominio del problema

1.1 Problema

Durante il servizio di formazione offerto dall'Università degli Studi di Salerno, i docenti insegnano i vari argomenti del corso agli studenti, dato che la mole di studenti che frequentano i corsi è molto elevata, per il docente risulta difficile riuscire a rispondere ad ogni domanda che gli viene posta durante e dopo la lezione. Ogni docente ha degli orari di ricevimento che mette a disposizione nel caso gli studenti non abbiano capito gli argomenti spiegati a lezione. Succede però, chi per esigenze di trasporto, chi per esigenze lavorative, gli studenti non riescano ad usufruire di questi ricevimenti.

1.2 Scopo del sistema

Con l'obiettivo di ottimizzare il servizio di formazione offerto dall'Università degli Studi di Salerno, si vuole sviluppare un progetto che si collochi nell'ambito dei corsi di studio della Facoltà di Informatica. Lo scopo principale è quello di fornire uno strumento di interazione con gli studenti, utile per lo scambio di domande e valutazioni relative ai vari argomenti spiegati a lezione in modo tale da migliorare sempre di più la didattica all'interno dei vari insegnamenti.

1.3 Ambito del sistema

Il contesto di utilizzo di tale sistema sarà negli insegnamenti presenti all'interno dei vari corsi di laurea del dipartimento d'informatica dell'Università degli Studi di Salerno. Tutto il corpo docente potrà utilizzare tale strumento per potersi interfacciare al meglio con i propri studenti in modo tale da instaurare un dialogo più aperto ma soprattutto produttivo. Tale contesto ha la necessità di un sistema come quello descritto nelle righe precedenti, per migliorare la didattica all'interno dei vari corsi di laurea del dipartimento d'Informatica.

1.4 Definizioni, acronimi e abbreviazioni

Acronimi	Descrizione
RF	Requisito Funzionale
RNF	Requisito non Funzionale
GCI	Gestione Corso d'insegnamento
GD	Gestione domande
SC	Scenario

2. Scenari

2.1 SC_Inserisci corso d'insegnamento

Nome Scenario	Inserisci corso d'insegnamento
Partecipanti	Salvatore: Amministratore
Flusso degli Eventi	<p>Salvatore è un amministratore della piattaforma UniQuestions, viene contattato dal docente De Lucia via e-mail chiedendogli di inserire il proprio corso d'insegnamento nel sistema.</p> <ol style="list-style-type: none">Salvatore entra sulla piattaforma UniQuestions, effettua il login inserendo i propri dati nei campi appositi:<ul style="list-style-type: none">➤ USERNAME: <u>admin@unisa.it</u>➤ PASSWORD: <u>admin</u>Una volta entrato nella piattaforma, Salvatore clicca sul bottone “Crea Corso D’Insegnamento” per accedere all’area per la creazione di un nuovo corso.Salvatore compila i vari campi con le informazioni fornitegli dal docente, quali:<ul style="list-style-type: none">➤ Nome Corso: <u>Ingegneria del Software</u>➤ Corso di laurea: <u>Triennale</u>➤ Anno di Studio: <u>2019/2020</u>➤ Semestre: <u>Primo Semestre</u>➤ Docente: <u>Andrea De Lucia</u>Una volta inserito tutti i dati del corso, Salvatore clicca sul bottone “Conferma” per creare il nuovo corso d'insegnamento.

2.2 SC_Elimina corso d'insegnamento

Nome Scenario	Elimina corso d'insegnamento
Partecipanti	Salvatore: Amministratore
Flusso degli Eventi	Salvatore è un amministratore della piattaforma UniQuestions, decide di eliminare un corso d'insegnamento, poiché si rende conto di aver

	<p>sbagliato ad inserire i dati, oppure il seguente corso non è più presente nel piano di studi.</p> <ol style="list-style-type: none"> Salvatore entra sulla piattaforma UniQuestions, effettua il login inserendo i propri dati nei campi appositi: <ul style="list-style-type: none"> ➤ USERNAME: <u>admin@unisa.it</u> ➤ PASSWORD: <u>admin</u> Una volta entrato nella piattaforma, Salvatore clicca sul bottone “Visualizza Elenco Corsi” per visualizzare la lista dei corsi d’insegnamento presenti sulla piattaforma. Seleziona il corso desiderato per poi cliccare sul bottone “Elimina Corso”, dopo di che clicca su “Conferma” e il corso viene eliminato dalla piattaforma.
--	---

2.3 SC_Invio domanda

Nome Scenario	Invio domanda.
Partecipanti	Fabio: Studente
Flusso degli Eventi	<p>Fabio è uno studente iscritto alla facoltà d’informatica ed è iscritto alla piattaforma UniQuestions, dopo aver seguito la lezione di Ingegneria del Software, si rende conto di non aver capito un argomento spiegato a lezione e quindi decide di contattare il docente per avere delle delucidazioni.</p> <ol style="list-style-type: none"> Fabio entra sulla piattaforma UniQuestions, esegue il login inserendo i propri dati nei campi appositi: <ul style="list-style-type: none"> ➤ USERNAME: <u>fabio@studenti.unisa.it</u> ➤ PASSWORD: <u>fabio123</u> Una volta entrato nella piattaforma, Fabio clicca sul bottone “Visualizza Corsi” per visualizzare la lista dei corsi d’insegnamento in cui è iscritto. Il sistema fa apparire a schermo tutta la lista dei corsi in cui è iscritto Fabio, Fabio seleziona il corso di Ingegneria del Software e clicca sul bottone “Invia Domanda”. Fabio inserisce nella casella Domanda, il quesito che vuole porre al docente,

	<p>➤ DOMANDA: <u>Cosa sono i requisiti funzionali?</u></p> <p>5. Una volta inserito la domanda, Fabio clicca sul bottone “Invia” e il sistema invia la domanda al docente.</p>
--	---

2.4 SC_Visualizza Risposte

Nome Scenario	Visualizza risposte.
Partecipanti	Fabio: Studente
Flusso degli Eventi	<p>Fabio è uno studente iscritto alla facoltà d'informatica ed è iscritto alla piattaforma UniQuestions, dopo aver inviato una domanda al docente, vuole visualizzare se ha ricevuto risposta al quesito.</p> <ol style="list-style-type: none"> 1. Fabio entra sulla piattaforma UniQuestions, esegue il login inserendo i propri dati nei campi appositi: <ul style="list-style-type: none"> ➤ USERNAME: <u>fabio@studenti.unisa.it</u> ➤ PASSWORD: <u>fabio123</u> 2. Fabio per visualizzare le risposte ai quesiti che ha posto ai docenti, clicca sul bottone “Visualizza Risposte”. 3. Il sistema mostra a Fabio la lista con tutte le domande risposte dal docente.

3. Requisiti Funzionali

3.1 Gestione Corsi d'insegnamento

Gestione Corsi d'insegnamento		
Codice	Descrizione	Priorità
RF_GCI_1	Inserisci corso d'insegnamento	ALTA
RF_GCI_2	Elimina corso d'insegnamento	ALTA

➤ **RF_GCI_1 – Inserisci insegnamento.**

Permette all'amministratore di inserire un nuovo corso d'insegnamento all'interno di un corso di laurea.

➤ **RF_GCI_2 – Elimina insegnamento.**

Permette all'amministratore di eliminare un corso d'insegnamento all'interno di un corso di laurea.

3.2 Gestione Domande

Tabella 4 Gestione Domande		
Codice	Descrizione	Priorità
RF_GD_1	Invio domanda	ALTA
RF_GD_2	Visualizza risposte	ALTA

➤ **RF_GD_1 – Invio domanda.**

Permette allo studente di inviare una domanda al docente.

➤ **RF_GD_2 – Visualizza risposte**

Permette allo studente di visualizzare la risposta data dal docente.

4. Requisiti non funzionali

4.1 RNF_1: Usabilità.

Il sistema per garantire una facilità di utilizzo garantirà una interfaccia user-friendly. Essa deve essere curata nei minimi dettagli. Gli utenti che si interfacciano con il sistema devono riuscire ad accedere in maniera molto semplice alle funzionalità del sistema. Il sistema sarà usato da studenti che hanno già conoscenze di sistemi simili al nostro, un esempio tra tutti è la piattaforma E-learning. Quindi, il nostro sistema sarà progettato tenendo conto di quella che è l'user-experience degli studenti e non sarà necessario l'uso di manuale utente per compiere le azioni. Inoltre, l'utente non avrà bisogno di impegnarsi per imparare come compiere le azioni, perché sarà tutto molto semplice e intuitivo.

4.2 RNF_2: Affidabilità.

L'utente che utilizza il sistema difficilmente si scontrerà con errori. Qualora si verificassero, il sistema dovrà essere in grado di gestirli e risolverli, notificando tutti gli errori in maniera adeguata all'utente. Il sistema utilizza un database relazionale. Esso deve essere affidabile e poter mantenere i propri dati coerenti anche in caso di guasti. Il sistema dovrà garantire una connessione verso il database per la manipolazione dei dati e per le query, indipendentemente dal numero di utenti collegati nello stesso istante.

4.3 RNF_3: Performance.

Il sistema deve essere in grado di interagire, contemporaneamente, con circa mille utenti, mantenendo tempi di risposta brevi, quindi dovrà essere sviluppato e implementato garantendo la velocità del servizio e l'assenza di caricamento lunghi per l'accesso ai dati e utilizzo delle funzionalità del sistema. Inoltre, per mantenere alte le performance saranno richiesti server in grado di gestire numerose richieste garantendo ottime prestazioni.

4.4 RNF_4: Supportabilità.

Il sistema deve essere compatibile, correttamente funzionante e deve essere accessibile comodamente anche da dispositivi mobile e tablet grazie ad un'interfaccia responsive.

4.5 RNF_5: Manutenibilità

Il sistema deve essere facilmente manutenibile ed estendibile per supportare lo sviluppo di aggiornamenti e nuove funzionalità. Nelle fasi di progettazione e implementazione si cercherà di ridurre l'accoppiamento e le dipendenze tra le componenti del sistema. Inoltre, si cercherà di rispettare alcune linee guida per la scrittura del codice in modo da facilitare l'aggiunta di nuove funzionalità.

4.6 RNF_7: Packaging

Il sistema sarà rilasciato con più packaging, dove all'interno di ogni pacchetto saranno presenti i diversi sottoinsiemi.