Unitn.solve.it

Documento di progetto

**Obiettivo del progetto**

Il progetto Unitn.solve.it consiste nella realizzazione di una Webapp accesibile a studenti e docenti dell’università di Trento per mettere a disposizione un servizio automatico di esercitazione e di condivisione di materiale didattico. Il focus dell’applicazione è quello di offrire, agli studenti che gradiscono, l’opportunità di esercitarsi con del materiale inerente ai corsi di studio seguiti, nella maniera più efficiente possibile stando allo stesso tempo al passo con le proposte del docente. In particolare, il sistema si occupa di gestire consegne di esercizi e relative soluzioni e di proporle ad intervalli di tempo prestabiliti agli studenti. Il sistema è incentrato nella distribuzione di esercizi di carattere “problem solving”. Gli esercizi sono proposti e dunque inseriti nel sistema Unitn.solve.it dagli insegnanti che desiderano usufruire del servizio. Oltre alla pubblicazione di esercizi “problem solving” i docenti possono pubblicare per ogni corso anche del materiale propedeutico alla comprensione degli argomenti inerenti agli esercizi pubblicati. I docenti, inoltre, stabiliscono a loro discrezione le tempistiche di distribuzione degli esercizi agli studenti iscritti ai corsi. Gli studenti possono iscriversi ai corsi creati dagli insegnanti e ricevere i task proposti nelle tempistiche stabilite //è una ridondanza dei requisiti?.

**Macro-funzionalità del sistema**

* Le esercitazioni pubblicabili su unitn.solve.it sono incentrate sul problem solving, dunque ad esse si può fornire una risposta che è valutabile in termini di successo in maniera ben definita (con un punteggio espresso in centesimi) e automatizzata (la piattaforma deve poter inferire con dei sistemi di valutazione automatica, siano essi delle formule numeriche o dei criteri elencati, il punteggio alla risposta) //questo sarà uno dei requisiti funzionali associati alla prima macro-funzionalità
* Il materiale e le esercitazioni pubblicate su Untn.solve.it sono suddivisi in corsi che i docenti sono tenuti a creare
* Per ogni corso creato i docenti possono caricare del materiale didattico (video, pdf, documenti, ecc.) e pubblicare delle esercitazioni
* Gli studenti che accedono alla piattaforma possono iscriversi a qualunque dei corsi proposti dai docenti e usufruire del materiale caricato e delle esercitazioni pubblicate
* Unitn.solve.it raccoglie delle statistiche sullo svolgimento delle esercitazioni mostrando ai docenti, in maniera anonima, le statistiche di successo associate ad una singola esercitazione, e agli studenti le statistiche a lungo termine sul proprio successo di svolgimento di ogni singola esercitazione
* Se il docente che gestisce un dato corso lo consente, gli studenti iscritti possono usufruire delle esercitazioni passate per svolgere dei test a tempo in preparazione dell’esame

**Requisiti funzionali**

Strutturazione delle esercitazioni

* [**RF1**] Il sistema deve permettere ai docenti di creare un’esercitazione, che abbia un testo (in cui viene descritto il problema da risolvere), una tipologia di risposta adeguata al problema allegato e una spiegazione dettagliata della procedura di risoluzione del problema.
* [**RF2**] Il sistema deve permettere ai docenti, al momento della creazione di un’esercitazione, di scegliere una delle seguenti tipologie di risposta:
  + [**RF2.1**] Risposta numerica
  + [**RF2.2**] Crocette a risposta multipla
  + [**RF2.3**] Crocette a risposta unica
  + [**RF2.4**] Risposta tramite form CMS //da stabilire dopo la consegna
* [**RF3**] Il sistema deve permettere ai docenti di pubblicare le esercitazioni in una certa data, e per ogni esercitazione di specificare la data di pubblicazione della soluzione, (che dovrà essere successiva a quella di pubblicazione)
  + [**RF3.1**] Il docente è vincolato a caricare per ogni esercitazione il file con la soluzione dell’esercizio e la risposta numerica.
* [**RF4**] Il sistema permette al docente di visualizzare per ogni esercitazione le impostazioni e modificarle
  + [**RF34.1**] Una volta pubblicate le esercitazioni possono essere rese non visibili dai docenti.
* [**RF5**] Il sistema valuta in modo automatico le risposte fornite dagli studenti alle esercitazioni, ogni risposta viene valutata in centesimi, gli studenti possono sottomettere un numero arbitrario di soluzioni, ma il sistema terrà poi conto dell’ultima risposta.

Creazione dei corsi

* [**RF6**] Dopo il sign in il sistema deve permettere ai docenti di vedere i propri corsi e di crearne di nuovi.
  + [**RF6.1**] Alla creazione di un corso permette di assegnare ad un corso un titolo ed una descrizione.
* [**RF7**] Deve permettere agli studenti di vedere i corsi disponibili (ricercando per materia o per professore), e a parte i corsi ai quali è già iscritto.

Creazione e pubblicazione materiale didattico

* [**RF8**] Il sistema deve permettere al docente di caricare materiale come video, pdf e documenti all’interno del corso dal proprio dispositivo
  + [**RF8.1**] I file possono includere una data di pubblicazione, se non è inclusa viene pubblicato immediatamente
* [**RF9**] Il sistema deve permettere al docente di caricare esercitazioni, ogni esercitazione deve essere composta da:
  + [**RF9.1**] La consegna dell’esercitazione, sotto forma di un file pdf o da una stringa di testo
  + [**RF9.2**] Una risposta (vedi RF2)
  + [**RF9.3**] soluzione all’esercitazione, sotto forma di un file pdf o da una stringa di testo
  + [**RF9.4**] Una data di pubblicazione dell’esercitazione (opzionale), se non è inclusa viene pubblicata immediatamente
  + [**RF9.5**] Una data di scadenza dell’esercitazione, dopo la quale verrà pubblicata la soluzione, e impedita la possibilità agli studenti di rispondere
* [**RF10**] Il sistema deve permettere ai docenti di registrarsi tramite le propie credenziali unitn
* [**RF11**] Il sistema deve riconoscere in automatico quando un docente si registra alla piattaforma e garantirgli il ruolo di docente
* [**RF12**] Il sistema deve permettere agli utenti con il ruolo di docente (e solo a loro) di creare corsi
* [**RF13**] Il sistema deve fornire ai docenti una schermata per visualizzare tutti i propri corsi
* [**RF14**] Il sistema deve fornire ai docenti, per ogni corso
  + [**RF14.1**] Una schermata principale del corso, contenente file ed esercitazioni pubblicate, e tutte quelle programmate per la pubblicazione (filtri? ordine cronologico? …)
  + [**RF14.2**] Una schermata per caricare tutto il necessario per pubblicare una nuova esercitazione o un nuovo file

Funzionalità degli studenti

* [**RF15**] *Il sistema deve permettere agli studenti di effettuare un sign-in tramite le credenziali dell’università di Trento*
* [**RF16**] *Il sistema, al momento del sign-in di uno studente, deve in automatico verificare qualora lo studente sia già iscritto alla piattaforma e, in caso contrario, riconoscerlo come studente iscritto*
* [**RF17**] *Il sistema, a sign-in dello studente effettuato, deve mostrare allo studente con le modalità adeguate al contenuto (potrebbe essere non funzionale e in tal caso sarebbe da meglio specificare e porre come misurabile):*
  + *Tutti i corsi a cui lo studente è iscritto*
  + *L’elenco di tutti i corsi disponibili a cui lo studente non è iscritto attualmente*
* [**RF18**] *Quando lo studente seleziona dal relativo elenco un corso a cui è iscritto, la piattaforma deve mostrare l’elenco dei materiali didattici in quel corso resi visibili allo studente e l’elenco delle esercitazioni suddivise in:*
  + *esercitazioni in corso*
  + *esercitazioni già terminate*
  + *esercitazioni il cui documento di spiegazione non è ancora stato visualizzato*
* [**RF19**] *Il sistema deve fornire ad ogni studente una schermata per visualizzare tutti i corsi a cui sono iscritti ed iscriversi ad un nuovo corso (tipo tasto in un angolino che fa comparire un pop-up)*
* [**RF20**] *Il sistema deve fornire ad ogni studente una schermata per ogni corso, contenente:*
  + [**RF20.1**] *L’elenco delle esercitazioni (filtri per quelle terminate?)*
  + [**RF20.2**] *L’elenco dei file caricati dal docente*
* [**RF21**] *Quando lo studente seleziona dal relativo elenco un corso a cui non è attualmente iscritto, la piattaforma deve mostrare i dettagli associati al corso (docente, abstract del corso)*
* [**RF22**] *La piattaforma deve permettere agli studenti di effettuare una ricerca dagli elenchi di corsi in base ai seguenti criteri:*
  + [**RF22.1**] *Nome del docente*
  + [**RF22.2**] *Nome del corso*

Statistiche delle esercitazioni

* [**RF23**] *La piattaforma deve raccogliere dati statistici sui risultati degli studenti nello svolgimento delle esercitazioni*
  + *Per una data esercitazione la piattaforma deve raccogliere i dati esclusivamente sull’ultima consegna effettuata da ogni studente*
  + *La piattaforma deve raccogliere i dati al momento della conclusione dell’esercitazione, che coincide con la pubblicazione del testo/file di spiegazione dello svolgimento*
* [**RF24**] *La piattaforma deve permettere ai docenti di visualizzare le statistiche raccolte per una data esercitazione dopo la sua terminazione*
* [**RF25**] *La piattaforma deve permettere agli studenti di visualizzazione per ogni corso le statistiche di svolgimento di tutti gli esercizi già consegnati*

o    *Una schermata con le statistiche delle valutazioni degli esercizi svolti*

* [**RF26**] Il sistema deve permettere al docente di vedere il numero di studenti che hanno consegnato almeno una risposta ad una certa esercitazione, e in forma anonima di visualizzare le ultime soluzioni di ogni singolo studente con i relativi punteggi. il sistema deve permettere inoltre di visualizzare una media dei risultati  ed un istogramma (studenti)/(punteggio in centesimi).

Il sistema permette di visualizzare le risposte consegnate alle esercitazioni passate con i relativi punteggi.

Gestione delle simulazioni d’esame

o    [**RF27**] Ogni esercitazione può essere marcata dal docente come “esercitazione d’esame”

o    [**RF28**] Un’esercitazione d’esame è un’esercitazione che potrebbe comparire in un esame

o    [**RF29**] Per simulare un esame, gli studenti devono scegliere il numero di domande e il tempo che vogliono avere a disposizione

o    [**RF30**] Il numero di domande scelte non deve essere superiore al numero di esercitazioni d’esame pubblicate

o    [**RF31**] Quando una simulazione d’esame inizia, vengono scelte quel numero di domande casualmente tra quelle che possono comparire ad un esame

o    [**RF32**] Allo scadere del tempo, o quando lo studente comunica al sistema di aver finito, le domande vengono corrette, e viene comunicato il punteggio finale

o    [**RF33**] (Il punteggio viene salvato a parte nella schermata delle statistiche dello studente?)

**Requisiti non funzionali**

*Al momento della sottoposizione delle risposte da parte dello studente il sistema deve mettere a disposizione un tempo di 2 secondi per la correzione automatica della risposta, deve inoltre permettere di caricare solo un numero limitato di risposte consecutive e poi bloccare la sottoposizione di risposte per un breve lasso di tempo: dopo 3-4 risposte a distanza inferiore di 10 secondi l’una dall’ altra e necessario aspettare 30 secondi prima di poter sottomettere un altra risposta.*

Ai fini della sicurezza il sistema mette a disposizione un tempo di 2 secondi per correggere la singola risposta, e limita lo studente a caricare 1 risposta ogni 20-30 secondi.

* [RNF1] L’accesso al sistema (meccanismi di sign-in e sign-up) deve avenire in maniera sicura
* Tutte le transazioni tra front-end e back-end devono avvenire in maniera sicura, quindi essere trasmesse tramite protocollo https
* Se un utente front-end non è attivo (ossia non invia richieste al server) la sessione di utilizzo può  20 min
* A sign-in effettuato una sessione di attività del front-end deve essere identificata da un token temporaneo (univoco nel suo dominio di esistenza) riconosciuto dal back-edn, che dunque lo deve conservare durante tutto il suo arco di attivazione
* Tutte le richieste https che il client efettua verso il server devono essere marcate con il token della sessione
* Il back-end può fornire risposte alle richieste https dei client solo se il token fornito nella richiesta è valido (presente tra i token attivi noti al back-end) ed il tempo trascorso dall’ultima richiesta ricevuta con lo stesso token è inferiore al tempo limite stabilito al requisito [RF … ]. se nessuna delle due precedenti condizioni è verificata il back-end informa il front-end che la sessione è scaduta e se il token è ancora in memoria del back-end esso viene rimosso
* Al meccanismo di specifica della risposta di un esercizio da parte di un utente loggato vengono applicati i seguienti vincoli:
  + Non si può effettuare submitting di risposte a meno di 30 secondi unda l
  + La verifica del punteggio della risposta è effettuato a front-end
  + Il submit di una risposta è svincolato dal controllo del punteggio per la medesima
  + Ogni volta che viene effettuato un submit nel database il punteggio viene sorascritto
* Ogni riciesta http inviata al server
* [RNFx] Il sistema deve essere accessibile ventiquattr’ore su ventiquattro a meno di malfunzionamenti o manutenzione, sono previsti 5 giorni l’anno di manutenzione.
* [RNFx+1] Usabilità: l’utente è in grado di usare il sistema senza la necessità di essere formato su come funziona dopo 30 minuti di utilizzo.
* Il numero di elementi presenti a schermo con cui l’utente può interagire è limitato a 8.
* [RNFx+2] Privacy:
  + Non è possibile visualizzare dati di altri account
  + Le statistiche visibili ai docenti sono anonime
  + L’accesso al sistema è delegato alle API di Unitn.
  + L’unica informazione sull’utente a cui il sistema accede dalla login e se è studente o Docente.
* [RNFx+3] Multipiattaforma: il sistema è accessibile da computer smartphone e tablet, purché supportino la versione x e successive di firefox, e probabile , ma non garantito, che sia accessibile da altri browser.