



UNIVERSITÀ DEGLI STUDI DI SALERNO

Ingegneria del Software

PROBLEM STATEMENT



ANNO ACCADEMICO 2015/2016

Versione 1.0

Top Manager:

Nome
Prof. De Lucia Andrea

Project Manager:

Nome	Matricola
De Chiara Davide	0222500088
Longo Alessandro	0222500121

Partecipanti:

Nome	Matricola
Castellano Dario	0512102552
D'Avanzo Antonio Luca	0512102502
De Blasio Christian	0512102268
De Rosa Federico	0512101904
Di Domenico Carlo	0512102316
Esposito Fabio	0512102426
Korniychuk Alina	0512102256
Martiniello Pasquale	0512102616
Pecorelli Fabiano	0512102390
Shevchenko Sergiy	0512102368
Tufano Giuseppina	0512102588
Zanin Elvira	0512102396

Revision History:

Data	Versione	Descrizione	Autore
18/10/2015	1.0	Stesura del documento	Membri del Team

Indice

1. Dominio del problema	4
2. Scenari	5
3. Requisiti Funzionali	11
4. Requisiti non Funzionali	11
4.1 Usabilità.....	11
4.2 Affidabilità.....	11
4.3 Performance.....	11
4.4 Manutenibilità	12
4.5 Implementazione	12
4.6 Legal	12
4.7 Sicurezza.....	12
5. Ambiente di Destinazione	13
6. Consegne e Scadenze.....	13

1. Dominio del problema

Nel dipartimento d'Informatica dell'Università degli Studi di Salerno, diversi sono i docenti che per valutare i propri studenti ricorrono all'ausilio di prove scritte con domande a risposta multipla.

Risulta quindi considerevole la mole di lavoro effettuata dai docenti per ogni sessione nel preparare tali prove scritte e soprattutto nel correggerle, visti i numeri sempre crescenti di studenti iscritti. Altro fattore da considerare è la quantità di fogli utilizzati per le stampe di tali prove d'esame. Alcuni docenti inoltre utilizzano tale modalità anche per esercitazioni o prove intercorso, e, per conto proprio, tengono traccia dei diversi risultati ottenuti dagli studenti. L'obiettivo è quindi un miglioramento ed efficientamento delle modalità d'esame sia per gli studenti che per i docenti; il tutto volto a minori tempi per la valutazione degli studenti, di realizzazione delle prove d'esame e ad un risparmio anche economico per il dipartimento.

Si propone di apportare al Dipartimento d'Informatica un beneficio dal punto di vista dei costi (risparmiando sull'acquisto di carta) oltre che un notevole vantaggio dal punto di vista dell'efficienza dei docenti che si troveranno ad utilizzare un sistema che consente di creare e somministrare agli studenti, in modo automatico, dei test sia a carattere valutativo che esercitativo. Un ulteriore vantaggio sarà la facilità dell'esecuzione dei test da parte dello studente il quale avrà la possibilità di valutare al meglio il proprio studio ed eventualmente venire incontro alle proprie carenze attraverso le esercitazioni proposte dal docente.

In particolare il docente avrà la possibilità di creare facilmente dei test, sia a carattere esercitativo che valutativo, inserendo sia domande a risposta multipla che domande a risposta aperta. Il sistema, nel caso di test con domande a risposta multipla, sarà in grado di valutare la prova sulla base delle risposte corrette segnalate dal docente. In questo modo il docente e gli stessi studenti hanno la possibilità di conoscere quasi in tempo reale il risultato dei test.

Il sistema sarà capace di gestire più corsi contemporaneamente. La gestione dei corsi comprenderà anche la possibilità di condurre esercitazioni ed esami relativi a quest'ultimi. Ogni studente avrà a disposizione una serie di esercitazioni che aiuteranno nella comprensione degli argomenti e l'accompagneranno di volta in volta verso il conseguimento dell'esame. Questo sarà un punto fondamentale del sistema: semplificare ed incentivare lo studente.

Il sistema in oltre sarà in grado di creare delle statistiche basate sull'andamento delle esercitazioni e degli esami sostenuti, permettendo al docente di capire sia l'andamento dello studio e della comprensione degli studenti che seguono il corso di quest'ultimo, che l'andamento degli esami nel corso degli anni.

2. Scenari

Si riportano di seguito alcuni scenari significativi per il sistema e quindi per gli attori studente, docente e admin:

Nome Scenario	loginStudente
Istanze di Attori Partecipanti	Giusy: Studente
Flusso Eventi	<ol style="list-style-type: none">1. Giusy accede alla HomePage di SimplEx.2. Avendo già registrato il proprio account, inserisce il proprio ID e Password3. Giusy conferma i dati inseriti cliccando sull'apposito pulsante e attende di essere reindirizzata alla sua pagina personale.

Nome Scenario	esecuzioneEsame
Istanze di Attori Partecipanti	Giusy: Studente
Flusso di Eventi	<ol style="list-style-type: none">1. Giusy deve sostenere un esame.2. Giusy effettua lo scenario loginStudente.3. Giusy visualizza una lista contenente tutti i corsi alla quale è iscritta e seleziona il corso per cui intende effettuare l'esame.4. Giusy visualizza una schermata dalla quale è possibile avviare l'esame5. Giusy seleziona l'esame proposto dal docente.6. Giusy visualizza una schermata contenente tutte le domande dell'esame inserite dal docente, ogni domanda è costituita da un testo e da un numero di risposte.7. Giusy completa le domande visualizzate.8. Giusy termina l'esame confermando le sue risposte entro il tempo messo a disposizione dal docente.

Nome Scenario	esecuzioneEsercitazione
Istanze di Attori Partecipanti	Giusy: Studente
Flusso di Eventi	<ol style="list-style-type: none"> 1. Giusy decide di esercitarsi nello studio del corso di Ingegneria del Software. 2. Giusy effettua lo scenario loginStudente. 3. Giusy visualizza nome, docente e CFU dei corsi alla quale è iscritta e seleziona il corso per cui intende effettuare l'esercitazione. 4. Giusy visualizza l'elenco delle esercitazioni disponibili e per ognuna l'argomento. 5. Giusy visualizza a video le domande inserite dal docente e sotto ognuna le relative risposte, selezionabili con un segno di spunta, per la sessione di esercitazioni sull'argomento. 6. Giusy si accinge a completare l'esercitazione. 7. Quando Giusy ha terminato l'esercitazione, preme sul pulsante di conferma per confermare le sue risposte.

Nome Scenario	visualizzazioneRisultati
Istanze di Attori Partecipanti	Giusy: Studente
Flusso di Eventi	<ol style="list-style-type: none"> 1. Giusy effettua lo scenario loginStudente. 2. Giusy visualizza nome, docente e CFU dei corsi alla quale è iscritta e seleziona il corso per cui intende visualizzare i risultati. 3. Giusy accede alla pagina relativa al corso contenente test ed esercitazioni. 4. Giusy sceglie da un menù se visualizzare i risultati delle esercitazioni o quelli relativi agli esami, premendo su un pulsante poi conferma la scelta. 5. Giusy seleziona l'esame/esercitazione del quale vedere i risultati. 6. Viene mostrato da Simplex il punteggio ottenuto per lo specifico test in una tabella, contenente domanda e relativo punteggio.

Nome Scenario	creazioneTestValutativo
Istanze di Attori Partecipanti	Andrea: Docente
Flusso di Eventi	<ol style="list-style-type: none"> 1. Andrea effettua lo scenario del loginDocente 2. Simplex visualizza l'elenco dei corsi elencati per nome. 3. Andrea seleziona il corso per il quale desidera creare un test di valutazione. 4. Clicca sull'apposito pulsante per la creazione del test valutativo e attende di essere reindirizzato alla pagina relativa. 5. Decide di sottoporre agli studenti un certo numero di domande che inserisce nell'apposito campo. 6. Conferma i dati inseriti cliccando sull'apposito pulsante. 7. Simplex assegna un ID al test appena creato e reindirizza alla pagina per la scelta degli argomenti. 8. Andrea decide gli argomenti sui quali verterà il test, e sceglie tutti quelli relativi al corso in questione. 9. Inoltre per ogni argomento viene visualizzata la percentuale di risposte corrette. 10. Conferma cliccando sull'apposito pulsante e attende di essere reindirizzato alla pagina relativa alle domande. 11. Sceglie manualmente le domande che comporranno il test, selezionandole singolarmente dall'elenco che gli viene mostrato (il docente visualizza la percentuale di risposte corrette per ogni domanda). 12. Conferma le scelte inserite cliccando sull'apposito pulsante e attende che il test venga salvato.

Nome Scenario	avvioTest
Istanze di Attori Partecipanti	Andrea: Docente
Flusso di Eventi	<ol style="list-style-type: none"> 1. Andrea effettua lo scenario del loginDocente 2. Simplex visualizza l'elenco dei corsi elencati per nome. 3. Andrea seleziona il corso per il quale desidera avviare un test creato in precedenza. 4. Quindi seleziona un test non ancora avviato tra quelli disponibili al corso. 5. Decide di rendere disponibile il test di esercitazione per l'intero corso a partire da una certa data. 6. Inserisce la data di termine del test. 7. Decide, tramite un apposito bottone, se mostrare allo studente l'esito finale (o parziale, nel caso in cui siano presenti domande a risposta aperta) della prova. 8. Seleziona gli studenti che potranno effettuare il test. 9. Clicca sul pulsante di conferma e attende che la scadenza venga salvata.

Nome Scenario	inserimentoDomandaARispostaMultipla
Istanze di Attori Partecipanti	Andrea:Docente
Flusso di Eventi	<ol style="list-style-type: none"> 1. Andrea effettua lo scenario loginDocente 2. Viene reindirizzato alla propria Homepage. 3. Simplex mostra tutti i corsi al quale Andrea è associato. 4. Andrea seleziona il corso per il quale vuole inserire una domanda a risposta multipla. 5. Seleziona un argomento tra la lista di argomenti correlati al corso e preme sul pulsante "Visualizza Domande" 6. Viene caricata una pagina in cui sono presenti tutte le domande relative all'argomento. 7. Seleziona l'operazione "Inserisci domanda multipla". 8. Viene restituita una pagina in cui Andrea deve inserire il testo della domanda, il numero di possibili risposte e il testo di ogni risposta. 9. Andrea inserisce il testo della domanda 10. Sceglie il numero delle possibili alternative. 11. Inserisce per ogni alternativa una possibile risposta. 12. Indica la risposta esatta, premendo nel riquadro laterale alla risposta. 13. Conferma i dati inseriti, premendo sul relativo pulsante, e attende mentre Simplex effettua il salvataggio della nuova domanda, che viene notificato con una schermata/finestra.

Nome Scenario	statisticheTest
Istanze di Attori Partecipanti	Andrea: Docente
Flusso di Eventi	<ol style="list-style-type: none"> 1. Andrea effettua lo scenario del loginDocente. 2. Simplex visualizza l'elenco dei corsi elencati per nome. 3. Andrea seleziona il corso di interesse. 4. Simplex visualizza tutti i test che sono stati effettuati e tutti gli argomenti relativi per quel corso. 5. Andrea clicca sul pulsante "Statistica". 6. Simplex restituisce una pagina in cui sono contenute le statistiche di tutti i test effettuati per questo corso, mostrando la percentuale di risposte date correttamente. 7. Andrea seleziona un test per cui vuole conoscere le statistiche nel dettaglio. 8. Simplex apre una finestra in cui è presente la lista di tutte le domande e risposte presenti in quel test. Per le domande visualizza una percentuale di risposte corrette. Per le singole risposte mostra una percentuale riferita al numero di volte che quella risposta è stata data al test.

Nome Scenario	inserimentoDocente
Istanze di Attori Partecipanti	Davide:Admin, Sergio:Docente
Flusso di Eventi	<ol style="list-style-type: none"> 1. Sergio ha bisogno di iscriversi al sistema visto che intende far esercitare i propri studenti per la prossima prova in itinere. 2. Davide esegue lo scenario loginAdmin. 3. Si dirige sulla pagina relativa alla gestione dei docenti e seleziona l'opzione "crea nuovo docente". 4. A questo punto Davide inserisce Nome, Cognome, Email, Numero di Telefono ed eventuali corsi da lui tenuti. 5. Conferma i dati salvando il profilo del docente nel sistema facendo click sul pulsante "conferma". 6. Davide effettua lo scenario logoutAdmin.

Nome Scenario	creaCorsoDiLaurea
Istanze di Attori Partecipanti	Davide: Admin, Alessandro: Docente
Flusso di Eventi	<ol style="list-style-type: none"> 1. Davide, riceve una richiesta per la creazione del corso di di Informatica. 2. Effettua lo scenario loginAdmin. 3. Davide si dirige alla pagina di gestione dei corsi di laurea(tra le varie operazioni disponibili, sceglie quella che gli consenta di gestire i corsi di laurea.) 4. Davide crea il corso di laurea inserendo il nome, la tipologia(triennale, magistrale). 5. Davide conferma l'inserimento dei dati. 6. Appena riceve un messaggio di conferma operazione a buon fine. 7. Davide effettua lo scenario logoutAdmin.

Nome Scenario	creaCorso
Istanze di Attori Partecipanti	Davide: Admin, Andrea: Docente
Flusso di Eventi	<ol style="list-style-type: none"> 1. Davide riceve una richiesta da parte di Andrea per la creazione di un nuovo corso per il nuovo ordinamento del corso di laurea. 2. Davide esegue lo scenario loginAdmin. 3. Seleziona la pagina di gestione dei corsi. 4. Davide tra le opzioni sceglie quella che gli permette di creare un nuovo corso. 5. Davide inserisce a quale corso di laurea appartiene, il nome, numero di CFU. (Se il CdL non esiste, esegue prima lo scenario creaCorsoDiLaurea.) 6. Conferma la creazione e attende la risposta del sistema. 7. Davide, appena ricevuta la conferma che le operazioni siano andate a buon fine. 8. Davide effettua lo scenario logoutAdmin.

Nome Scenario	associazioneDocenteCorso
Istanze di Attori Partecipanti	Davide:Admin, Andrea:Docente
Flusso di Eventi	<ol style="list-style-type: none"> 1. Davide riceve una chiamata da parte di Andrea per poter essere associato ad un particolare corso che terrà nel presente anno accademico. <ol style="list-style-type: none"> a. Se il corso non esiste, esegue lo scenario creaCorso. 2. Davide, effettua lo scenario login. 3. Davide raggiunge la pagina per gestire i corsi di laurea. 4. Effettua lo scenario ricercaCorso oppure scorre la lista e seleziona quello interessato. 5. Tra le operazioni disponibili, clicca sul pulsante "modifica corso". 6. Tra le opzioni disponibili sceglie "associa docente".

	7. Immette la matricola del docente e conferma. 8. Attende il riscontro positivo delle operazioni ed effettua. 9. Davide effettua lo scenario logoutAdmin.
--	--

3. Requisiti Funzionali

Di seguito si riportano i requisiti ad alto livello che il sistema implementerà:

- Esecuzione automatica dei test;
- Valutazione automatica delle prove;
- Gestione argomenti/domande;
- Gestione utenti studenti;
- Gestione utenti docenti;
- Gestione test;
- Gestione statistiche relative ai test;
- Gestione corsi di laurea e relativi insegnamenti;

4. Requisiti non Funzionali

4.1 Usabilità

Sebbene il sistema sia rivolto agli studenti e docenti universitari, dai quali ci si aspetta certa dimestichezza con sistemi software, l'interazione con il sistema non deve presentare un elevato grado di complessità. Deve, piuttosto, essere utilizzabile da un pubblico più eterogeneo possibile. Gli utenti che interagiscono con il sistema devono poter effettuare le operazioni con semplicità e rapidità, senza l'utilizzo di una lettura di un manuale. Queste, tuttavia, devono essere presenti e accessibili per chiunque le desideri.

4.2 Affidabilità

Il sistema è basato sull'architettura client-server quindi entrambi componenti devono essere affidabili ed essere in grado di poter mantenere i propri dati anche in caso di guasti (problemi di fornitura elettrica, usura dell'hardware del server, attacchi informatici, problemi legati al browser, interruzione della connettività alla rete ecc.). Il sistema deve gestire regolarmente un disservizio, quindi ripristinare lo stato precedente al guasto.

Deve essere possibile pianificare dei backup periodici del database la cui cadenza sarà definita più avanti nel progetto. In caso di guasto, un riavvio del sistema è tollerabile, data la natura non-mission critical del sistema.

4.3 Performance

Il sistema deve essere in grado di gestire picchi di 1500 utenze contemporanei per due ore senza problemi. Tale valore dipende dalla probabilità molto alta di avere più corsi che stanno facendo prove in contemporaneo.

Il carico delle esercitazioni invece, non ha un impatto molto evidente sulla performance del sistema, dato che gli studenti possono esercitarsi in qualsiasi orario, il carico viene spalmato in modo più o meno equo nell'arco delle 24 ore.

Essendo sotto carico, il sistema deve avere i tempi di risposta all'incirca di 1 secondo per qualsiasi operazione effettuata dall'utenza.

Per garantire un corretto funzionamento del sistema sotto un carico molto elevato, quest'ultimo deve essere composto da più nodi sia per le istanze apache sia per i database.

4.4 Manutenibilità

Il sistema per essere facilmente manutenibile ed espandibile nel futuro, verrà implementato seguendo un architettura three-tier quindi sarà facilmente manutenibile. Si realizzerà inoltre una matrice di tracciabilità dei requisiti che consentirà di capire l'impatto di una eventuale modifica al sistema.

Inoltre la documentazione del sistema, (rad, sdd, odd) farà sì che il sistema sia progettato e costruito seguendo i principi dell'ingegneria del software garantendone la manutenibilità.

4.5 Implementazione

La parte back-end del sistema, verrà implementata in linguaggio PHP, eseguito sulla piattaforma Apache2. Come persistenza verrà utilizzato database relazione MySQL. Il front-end verrà realizzato con il supporto del framework Bootstrap 3, HTML5, CSS3 e jQuery.

4.6 Legal

SimplEx è realizzato in maniera tale da garantire il rispetto delle norme vigenti sulla privacy, facendo riferimento alla Legge 31 dicembre 1996, n. 675 del Parlamento Italiano, relativa alla tutela delle persone e di altri soggetti rispetto al trattamento dei dati personali.

4.7 Sicurezza

Il sistema prevede un modulo di autenticazione, in modo da negare l'accesso da parte degli utenti non autorizzati. In più il sistema per evitare gli attacchi di tipo "brute force", dopo più tentativi di accesso dalla parte della singola utenza, deve, ad ogni successivo tentativo effettuare verifica "anti-bot".

5. Ambiente di Destinazione

Viste le esigenze di performance elevate è possibile che il sistema risiederà su una piattaforma cloud di tipo SaaS o PaaS implementando così quei requisiti non funzionali di scalabilità computazionale e di banda.

Nel caso di PaaS si adopererà un S.O. Windows con un server Web Apache e database MySql.

Essendo un'applicazione di tipo Web essa potrà essere accessibile da ogni tipologia di dispositivo: personal computer, smartphones e tablet collegati alla rete.

Per garantire le caratteristiche responsive dell'interfaccia si adotteranno appositi framework.

6. Consegne e Scadenze

Documenti	Consegne
Requirements Analysis Document	10 Novembre 2015
System Design Document	30 Novembre 2015
Object Design Document	12 Gennaio 2015
Test Plan	16 Dicembre 2015
Test Case Specification	12 Gennaio 2015
Test Execution Report	12 Gennaio 2015