

ALMA MATER STUDIORUM
UNIVERSITÀ DEGLI STUDI DI BOLOGNA

Facoltà di Ingegneria

Corso di Laurea in INGEGNERIA INFORMATICA

Tesi di Laurea in FONDAMENTI DI INFORMATICA L-B

**Progetto e realizzazione di un
sistema di supporto al docente
per la gestione di prove in
laboratorio**

Candidato:
Silvia Cereda

Relatore:
Chiar.mo Prof. Enrico Denti

Anno Accademico 2004/2005 - Sessione II

PAROLE CHIAVE

Laboratorio

Gestione

Esami

Docente

Inserire qui
la dedica

Indice

Introduzione	1
1 L'applicativo preesistente	3
1.1 Struttura e configurazione di EsamiLIA	3
1.1.1 Installazione	4
1.1.2 Configurazione	4
1.1.3 Principi di funzionamento	5
1.2 Carenze riscontrate nel sistema	6
1.3 Possibili vulnerabilità	7
1.4 Riprogettazione	8
2 Analisi preliminare del progetto	11
2.1 Requisiti	11
2.1.1 Analisi	11
2.1.2 Fattori di qualità	12
2.2 Possibili approcci	13
2.2.1 Aggiunta di una patch	13
2.2.2 Reingegnerizzazione del sistema	13
2.2.3 Possibile soluzione	14
2.3 Requisiti specifici	15
2.3.1 Parte studente	15
2.3.2 Tool di amministrazione per il docente	16
3 Analisi di casi d'uso e scenari	19
3.1 Analisi degli utenti coinvolti	19
3.2 Analisi dei casi d'uso	19

3.2.1	Analisi dei casi d'uso dello studente	20
3.2.2	Analisi dei casi d'uso del docente	20
3.2.3	Analisi dei casi d'uso dell'esaminatore	21
3.3	Analisi degli scenari	23
3.3.1	Scenari dello studente	23
3.3.2	Scenari del docente	24
3.3.3	Scenari dell'esaminatore	25
4	Progetto e collaudo della parte studente	29
4.1	Progetto dell'applicativo	30
4.2	Autenticazione dello studente	31
4.2.1	L'unicità del login	32
4.2.2	Il proxy e l'identificazione dell'indirizzo IP	33
4.3	Sezione contenente le domande	34
4.4	Sezione per la consegna dei file	35
4.4.1	Consegna dei file	36
4.4.2	Prelievo e rimozione dei file	37
4.4.3	Problemi di path con Internet Explorer	38
4.5	Test e collaudo	38
4.5.1	Controllo della visibilità	39
4.5.2	Collaudo con diversi browser e sistemi operativi	39
4.5.3	Controllo dei contenuti	40
4.5.4	Collaudo di funzionalità	40
5	Progetto e implementazione della parte docente	41
5.1	Progetto del database	41
5.2	Registrazione dell'utente	42
5.2.1	Autenticazione tramite password	43
5.3	Il login dell'utente	45
5.4	Il ruolo del docente	46
5.4.1	La delega ad un altro utente	46
5.5	Creazione di un appello d'esame	48
5.5.1	L'abilitazione di IP e sottoreti alla parte studente	49
5.5.2	La suddivisione in gruppi	50
5.6	Stato di un turno d'esame	51

5.7	La ricerca di un esame	53
5.8	Le modifiche possibili ai turni d'esame	54
5.8.1	La modifica del numero totale dei turni	54
5.8.2	L'eliminazione di turni d'esame	56
5.9	L'inserimento della lista e dei testi delle prove	57
5.10	Controlli durante l'esame	59
5.10.1	L'inizio del turno esame	59
5.10.2	Gestione della fase di entrata degli studenti	59
5.10.3	Controllo della situazione in laboratorio	60
5.10.4	Modifica delle impostazioni di uno studente	61
5.11	La chiusura del turno d'esame	62
5.11.1	Cronologia degli esami terminati	62
6	Collaudo dell'applicativo	63
6.1	Annessione dell'applicativo parte studente alla parte docente .	63
6.1.1	Interfacciamento della parte studente	65
6.2	Collaudo del progetto completo	65
6.2.1	Controllo dei contenuti	65
6.2.2	Collaudo di funzionalità	66
6.2.3	Collaudo con diversi browser e sistemi operativi	66
6.2.4	Controllo della visibilità	66
6.3	Pre-test dell'applicativo	66
6.4	Test finale dell'applicativo	67
	Conclusioni	69
A	Manuale docente	71
A.1	Creazione dell'appello d'esame	71
A.1.1	Modifiche ai turni d'esame	72
A.1.2	L'eliminazione di un turno d'esame	72
A.1.3	Inserimento dei file dell'esame	72
A.2	Inizio dell'esame	73
A.2.1	Appello degli studenti	74
A.2.2	Controllo del laboratorio	74
A.2.3	Modifica dei dati degli studenti	75

A.3 Termine dell'esame	75
A.4 Principio di funzionamento	75
Bibliografia	77

Introduzione

In molti degli insegnamenti di informatica tenuti nei Corsi di Laurea attivi nella facoltà di Ingegneria sono previste prove di accertamento da svolgersi in laboratorio, il cui superamento costituisce una parte rilevante della valutazione dell'esame.

Per motivi di praticità e di sicurezza da alcuni anni si cerca di evitare la consegna degli elaborati mediante floppy disk, e per tale motivo è stato predisposto un sistema di consegna via rete. Questo sistema offre una serie di requisiti tra i quali la facilità di utilizzo, la possibilità per il docente di inserire le liste d'esame scaricate dal sistema di ateneo Uniwex, di inserire i testi degli esercizi ed eventualmente delle domande differenziate in base a gruppi predefiniti, e la possibilità per lo studente di consegnare in rete i file prodotti.

Il sistema già esistente si è dimostrato molto utile ed è tuttora usato da molti docenti, tuttavia l'esperienza ha messo in luce anche alcuni limiti.

In particolare sono emersi alcuni aspetti legati alla difficoltà di gestione, poiché il docente deve caricare la lista d'esame, i testi degli esercizi, scaricare gli elaborati ed è obbligato ad adattarsi ad una struttura di file interni che non è sempre di facile comprensione per tutti. L'applicativo esistente manca inoltre di un sistema di monitoraggio in tempo reale dell'evoluzione dell'esame, ed è pertanto non completamente robusto sotto il profilo della sicurezza, intesa come certezza che un utente possa vedere solo il proprio compito, non possa eseguire l'autenticazione al sistema più di una volta, o impersonare differenti utenti, ecc.

Si è quindi ritenuto opportuno procedere a una reingegnerizzazione dell'applicazione esistente per fornire al docente un supporto utile, sicuro e soprattutto

semplice da usare che permetta di gestire in modo globale l'esame a partire dalla sua creazione fino al controllo del suo svolgimento esecuzione offrendo al contempo al docente la possibilità di dirigere e controllare la prova dalla propria postazione, per avere una visione globale e completa della situazione del laboratorio, mentre un assistente esegue il normale controllo di routine in loco.

La tesi è suddivisa in cinque capitoli. Nel primo capitolo si parlerà del sistema utilizzato in precedenza, evidenziandone in particolare limiti e vulnerabilità. Il secondo capitolo illustrerà lo studio di fattibilità e i risultati della raccolta e dell'analisi dei requisiti. Nel terzo capitolo saranno mostrati i casi d'uso individuati e verranno analizzati gli scenari più rilevanti. Nel quarto e nel quinto capitolo invece verranno descritte la progettazione, l'implementazione e il successivo collaudo della sezione riferita allo studente e di quella riferita al docente.

Capitolo 1

L'applicativo preesistente

Le prove ed esercitazioni effettuate in laboratorio hanno spesso grande rilevanza ai fini del superamento di determinati esami dei corsi di Ingegneria. Inizialmente gli elaborati prodotti dagli studenti in queste prove venivano consegnati al docente su floppy disk in duplice copia, onde evitare che la prova risultasse irrimediabilmente compromessa dalla illeggibilità del dischetto.

Il tempo e l'esperienza hanno dimostrato che questo metodo era non solo scomodo, poiché il docente doveva ricevere una grande quantità di dischetti che, una volta corretti, diventavano inutili e risultavano ingombranti, dato che non venivano poi richiesti dagli studenti e dunque restituiti, ma anche potenzialmente poco sicuro in quanto occorreva vigilare affinché gli studenti non potessero scambiarsi i file durante l'esame.

Allo scopo di evitare al professore di doversi fare carico di tutti i floppy disk consegnati, e limitare così anche il pericolo delle copie, il professor Lodolo creò all'inizio del 2001 una Applicazione Web per la gestione degli esami chiamata EsamiLIA.

1.1 Struttura e configurazione di EsamiLIA

EsamiLIA è un'applicazione web scritta in linguaggio PHP, distribuita sotto forma di archivi di ZIP che il docente deve scompattare e configurare tramite la modifica di determinati file di testo.

1.1.1 Installazione

Per installare l'applicativo il docente crea all'interno del web server una directory per l'esame con due sottodirectory, *config* che deve contenere i file di configurazione, e *studenti*, con permessi tali che chiunque potesse scriverci, che deve invece contenere gli elaborati degli studenti.

La directory base contiene quindi alla fine della configurazione una serie di file PHP per permettere il login dello studente, la lettura del testo dell'esercizio, eventuali domande, se previste, e la consegna degli elaborati, assieme alle relative immagini e file di supporto o contenenti le impostazioni grafiche quali il file CSS. [1]

1.1.2 Configurazione

La directory *config* contiene tutti i file per configurare l'applicazione per un determinato esame, quali le pagine HTML con le avvertenze relative alle pagine dell'esercizio, delle domande o della consegna, i testi degli esercizi, la definizione delle domande per i vari gruppi, la lista degli studenti iscritti all'esame scaricata dal sistema di Ateneo Uniwex, e il file di configurazione vero e proprio *appesami.ini*.

Il file *appesami.ini* contiene tutte le informazioni per il funzionamento dell'esame, in particolare definisce la presenza di domande a risposta multipla, il numero dei gruppi per la suddivisione dei compiti, e i gruppi d'esame a cui sono assegnati i diversi indirizzi IP del laboratorio.

I file *infoese.htm*, *infocon.htm* e *infodom.htm* sono tre file HTML che devono mostrare il testo da visualizzare nella sezione delle avvertenze posta nella parte sinistra della pagina, i file *esercizioN.htm* e *domandeN.ini* (con *N* il numero che indica il gruppo) contengono rispettivamente il testo dell'esercizio, e le domande con relative risposte per ogni gruppo.

I file delle domande *domandeN.ini* hanno una struttura di questo tipo:

In questo file, la parte generale definisce le informazioni utili alla lettura del

```
[Generale]
Domande=1
[Domanda1]
Risposte=3

Testo=Quale delle seguenti dichiarazioni è corretta?
Risposta1=Testo della risposta 1
Risposta2=Testo della risposta 2
Risposta3=Testo della risposta 3
```

Figura 1.1: Schema del file *domandeN.ini*

file, in particolare quante volte cercare i tag di tipo *DomandaN*, e all'interno di ognuno di questi appaiono il numero totale delle risposte previste per la domanda attuale, il testo della domanda e i testi delle risposte.

Si noti che non esiste nei file di configurazione delle domande il riferimento alla risposta corretta, quindi il docente durante la correzione vede il numero della risposta scelta dallo studente.

1.1.3 Principi di funzionamento

L'applicazione parte presentando allo studente una schermata di scelta degli esami che possono essere effettuati, che porta alla pagina del login: quando lo studente cerca di accedere al sistema, dopo aver digitato il proprio numero matricola, l'applicativo controlla in *lista.txt* l'effettiva iscrizione dello studente all'esame. Se la matricola non appartiene alla lista d'esame il sistema non mostra alcun messaggio e ritorna automaticamente alla pagina di login; in caso di successo invece viene creata per ogni studente una directory il cui nome segue una struttura del tipo *N-Cognome-Nome* in cui *N* definisce il progressivo di iscrizione alla lista dell'esame e viene salvato un file con un nome simile a *_Pxx-Gyy* in cui *xx* si riferisce alle ultime due cifre dell'indirizzo IP del laboratorio dal quale lo studente accede, mentre *yy* mostra il gruppo di appartenenza, calcolato in base alla mappa dei gruppi contenuta in *appesami.ini*. Questo file serve prevalentemente per verificare sospetti di copiatura consentendo di ricostruire la posizione degli studenti all'interno del laboratorio.

La pagina dell'esercizio contiene solamente il testo dell'esercizio e non permette alcuna operazione: il file del testo d'esame è in sola lettura.

La pagina delle domande contiene un form con le domande e le possibili risposte costruito automaticamente sulla base dei file *domandeN.ini*. Alla conferma da parte dello studente, viene creato nella directory dello studente un file *_domandeN.txt* (in cui N rappresenta sempre il gruppo di appartenenza dello studente) contenente le lettere corrispondenti alle risposte di ogni domanda.

Il pulsante "Azzera" cancella il contenuto del file creato ed annulla di conseguenza la consegna delle domande.

La pagina di consegna permette di inviare nella directory dello studente i file dell'esercizio. All'invio di ogni file viene mostrato nella stessa pagina di consegna la lista di ciò che lo studente ha già consegnato, per consentirgli di scaricare i propri elaborati, o eventualmente di eliminarli.

1.2 Carenze riscontrate nel sistema

Il limite principale incontrato nell'utilizzo di EsamiLIA è soprattutto quello derivante dal fatto che il docente, per definire un nuovo esame, deve avere in primo luogo un account nella macchina remota, dopodiché deve accedere a tale macchina e poi creare le directory, scompattare gli archivi e inserire lista ed esercizi a mano: operazioni queste alquanto laboriose e che potrebbero essere semplificate e automatizzate dall'utilizzo un front-end grafico.

Dopo aver creato nel proprio spazio all'interno del web server la struttura dell'esame, che viene replicata per ogni prova di ogni docente, è necessario modificare un file generale situato nella directory radice di tutte le directory dei docenti, che contiene la lista delle prove che possono essere effettuate dagli studenti, che corrisponde al testo che lo studente vede nella prima schermata dell'applicazione e che gli permette di scegliere quale prova effettuare. Questo file deve essere quindi aggiornato manualmente per l'inserimento di nuove voci relative alle prove e per l'eliminazione delle prove obsolete.

Il docente zelante che volesse alleggerire questo file delle voci degli esami che sono già stati effettuati, dovrebbe creare un nuovo file come copia di backup della lista che ha modificato e aggiornato.

Allo stesso modo, al termine della prova, il docente è obbligato a rientrare nella macchina e salvare in locale le prove degli studenti per poterle correggere e valutare.

Al termine della prova in laboratorio inoltre alcuni docenti, per incentivare gli studenti a non indugiare sulla consegna dei propri elaborati, o per definire un termine di consegne rigoroso, scollegano il server per impedire ulteriori accessi: questo comportamento evidenzia la mancanza di un metodo meno spartano per impedire agli studenti ulteriori accessi e consegne. È palese che questo procedimento oltre che molto macchinoso è complicato e pericoloso.

Oltre a queste piccole carenze, è apparsa chiara l'esigenza di disporre di una parte di controllo e di gestione del laboratorio per permettere al docente di sapere cosa avviene durante la prova, ovvero di conoscere chi è seduto nella determinata postazione, a quale gruppo di esercizio appartiene, quali file ha già consegnato, eventualmente potendo modificare tali dati, ad esempio rendendo possibile lo spostamento di uno studente da una postazione ad un'altra o il cambiamento del gruppo di appartenenza dello studente.

1.3 Possibili vulnerabilità

Le vulnerabilità del sistema riguardano soprattutto non tanto aspetti tecnici del sistema, quando organizzativi.

Essendo infatti EsamiLIA formato da una serie di pacchetti che vengono replicati per ogni esame e per ogni docente, capita che l'applicativo venga considerato come un pacchetto a sé e per tale motivo venga modificato e personalizzato per meglio aderire alle esigenze di un determinato insegnamento. Questa considerazione ha portato al fiorire di più versioni incompatibili l'una con l'altra, con diverse patch e conseguenti piccoli bug specifici della singola versione.

In una di queste versioni in particolare è stata riscontrata la disattivazione

del controllo di un file di flag che determina se uno studente si è autenticato da una macchina ben precisa. Senza questo controllo uno studente potrebbe autenticarsi più volte dalla stessa macchina con il numero di matricola di diversi studenti, e quindi sarebbe possibile in linea di principio per uno studente sottrarre i file di un altro studente consegnandoli poi come propri. Sebbene il docente possa accorgersi della “similitudine” in fase di correzione, è certamente preferibile prevenire questo genere di rischi.

Un'altra potenziale vulnerabilità del sistema riguarda il controllo sull'indirizzo IP della macchina da cui proviene la richiesta di autenticazione per l'esame. All'ingresso nel sistema infatti viene inviato utilizzando il metodo *GET* l'indirizzo IP della macchina da cui si tenta di accedere all'esame. Questo sistema è potenzialmente rischioso poiché i parametri in *GET* possono essere agilmente manipolati e modificati da utenti maliziosi in quanto accodati all'indirizzo della pagina richiesta.

1.4 Riprogettazione

La riprogettazione ha quindi l'obiettivo di superare tali limitazioni e di offrire al contempo al docente un'interfaccia di amministrazione e monitoraggio evolute.

In particolare si richiede di eliminare il file eseguibile *esamilia.exe* che altro non fa che aprire un browser nascondendo la barra degli strumenti e la barra di stato utilizzando, in sua vece un normale browser che mostri sulla barra del titolo i dati dello studente (nome, cognome, numero di matricola dello studente e gruppo di appartenenza) in modo che sia facile riconoscere lo studente autenticato durante il controllo dei documenti in laboratorio.

La parte studente dovrà inoltre garantire un controllo che riconosca un successivo login per lo stesso studente anche dalla stessa postazione, in modo tale che, senza la diretta modifica da parte del docente, lo studente possa autenticarsi una sola volta. Il controllo sul login dovrà essere fatto sulla base della presenza della directory dello studente che funge da flag sul login.

Dovrà essere inoltre creata una consolle amministrativa per il docente che permetta di conoscere quali sono i terminali utilizzati in laboratorio, i dati

dello studente seduto ad una determinata postazione ed eventualmente di riabilitarne l'accesso, cambiarne il gruppo di esercizio, spostarlo, ecc.

Il programma completo, che sarà quindi un'applicazione web, dovrà funzionare completamente sotto protocollo HTTPS per garantire la sicurezza e la certezza dell'autenticazione tra client e server garantendo la sicurezza del collegamento.[2]

Capitolo 2

Analisi preliminare del progetto

2.1 Requisiti

L'impatto previsto sullo studente deve essere minimo: le operazioni da svolgersi non dovrebbero cambiare, quindi chi avesse utilizzato in precedenza EsamiLIA dovrebbe trovare intuitivo utilizzare anche il nuovo sistema.

Dal punto di vista del docente, la creazione delle prove verrà facilitata mediante un front-end grafico che compie automaticamente tutte quelle azioni che attualmente il docente deve fare a mano. Viene inoltre aggiunta tutta una parte di controllo della situazione del laboratorio che al momento non esiste.

2.1.1 Analisi

Dall'analisi dei rischi è emerso che potrebbero verificarsi problemi derivanti dalla presenza di potenziali rischi tecnologici e da rischi organizzativi.

Requisiti tecnologici

I requisiti tecnologici riguardano prevalentemente l'ambiente di sviluppo scelto, e di conseguenza l'esperienza richiesta e soprattutto i tempi di sviluppo. I tempi infatti sono abbastanza stretti, in quanto è necessario terminare velocemente la parte studente e poi cominciare a creare il tool di amministrazione del docente, ed è quindi necessario optare per un ambiente ed un linguaggio di programmazione che possano essere usati senza troppi intoppi e su cui si abbia una buona maneggevolezza. Sarà quindi necessario scegliere un lin-

guaggio già studiato e già utilizzato, che possa essere di facile utilizzo anche da parte di chi in futuro avrà bisogno di modificare l'applicativo per apporre ulteriori funzionalità o per correggere i bug trovati.

Requisiti organizzativi

I requisiti organizzativi riguardano soprattutto l'impatto che il nuovo applicativo potrebbe avere sul docente, il cui iter di creazione e impostazione di un esame cambia di molto dalla situazione attuale. Potrebbe infatti verificarsi una potenziale rigidità dell'ambiente organizzativo nell'accettare i cambiamenti connessi all'adozione di nuove procedure.

Si deve cercare quindi di semplificare al massimo le operazioni del docente e rendere il tool di amministrazione il più intuitivo possibile, nonché fornire all'utente un manuale che mostri velocemente come fare per creare e gestire i propri esami.

I requisiti organizzativi potrebbero toccare anche l'impatto del nuovo programma sullo studente; per evitare il rischio che lo studente non si ritrovi nel nuovo applicativo, si cercherà di mantenere l'ambiente grafico e le impostazioni principali come erano in precedenza, in modo che l'utente finale non noti molto la differenza.

2.1.2 Fattori di qualità

Il sistema deve garantire diversi fattori di qualità, tra cui i più importanti sono la robustezza, l'estensibilità, l'integrità e la facilità d'uso.

La robustezza si traduce nella possibilità per il sistema di comportarsi in maniera accettabile anche in situazioni anomale; l'estensibilità richiede che il codice del sistema sia facilmente modificabile, in modo che sia facile in futuro aggiungere ulteriori funzionalità per soddisfare ulteriori requisiti nati in base alle richieste e alle necessità dei docenti (si richiede quindi che il linguaggio di programmazione utilizzato sia semplice e conosciuto); in base a questo requisito si richiede inoltre che il software non risulti fragile di fronte alle modifiche. Deve essere inoltre garantita l'integrità del software, in quanto capiterà sicuramente che gli studenti cerchino sempre nuovi e più fantasiosi metodi per copiare; il sistema deve di conseguenza essere protetto

da intrusioni o manomissioni di natura volontaria e involontaria, ed è anche necessario che il software sia facile da usare, in modo che l'utente medio possa effettuare le operazioni basilari sia nell'utilizzo della parte studente, sia nell'utilizzo del tool di amministrazione per il docente.

2.2 Possibili approcci

Le possibili soluzioni di progetto riguardano l'opportunità di partire dall'applicazione esistente ed inserire una nuova patch, e da quello continuare anche per la creazione del tool per il docente, oppure reingegnerizzare tutto il sistema creandolo ex-novo con un linguaggio di programmazione che può anche essere differente a quello finora usato.

2.2.1 Aggiunta di una patch

Per inserire una ulteriore patch al sistema già esistente è necessario avere una conoscenza molto buona del PHP per poter analizzare tutto il codice per scoprire i punti deboli del sistema, ed eventualmente scovare debolezze non ancora individuate dai docenti e dagli esaminatore.

Scegliendo questa opportunità i tempi di progettazione della parte studenti possono accorciarsi, ma è necessario seguire gli schemi ben precisi definiti dalla presenza del codice esistente anche per la progettazione del tool per il docente.

2.2.2 Reingegnerizzazione del sistema

La possibilità di creare un nuovo sistema da capo offre un grado di libertà in più in quanto è possibile decidere come organizzarsi e soprattutto è possibile scegliere il linguaggio di programmazione più consono o più semplice da usare.

Questa opportunità ha tempi di programmazione proporzionali alla conoscenza del linguaggio di programmazione

Linguaggio PHP

Il PHP è un linguaggio di scripting general-purpose che è specialmente utilizzato per lo sviluppo Web e può essere incluso in pagine HTML.[3]

È un linguaggio discretamente complesso e poco manutenibile.

Tecnologia .NET

Si tratta del linguaggio di programmazione per applicazioni Web di Microsoft. Costituisce una possibile alternativa all'utilizzo del PHP. Il server però è e sarà su piattaforma Linux, e , dato che ASP e .NET non sono portabili su piattaforme non Windows.

Linguaggio JSP

Le Java Server Pages [5] sono un altro linguaggio lato server in grado di generare pagine dinamica. Questo linguaggio però, utilizzando la sintassi Java[4], è indipendente dal server e dalla piattaforma.

Le JSP e Java sono linguaggi semplici, conosciuti, e soprattutto multipiattaforma: si tratta perciò del linguaggio di programmazione ideale per lo sviluppo di questa applicazione.

2.2.3 Possibile soluzione

Dall'analisi dei requisiti e dei possibili approcci al problema, si è stati portati a optare per la reingegnerizzazione completa dell'applicativo, in modo tale da risolvere i bug che erano già stato trovati e da ottenere una migliore interconnettività con la parte di controllo remoto della situazione del laboratorio durante l'esame.

I requisiti del server portano inoltre ad escludere la possibilità di utilizzo di tecnologia Microsoft, dato che il server su cui è installata l'applicazione attuale e su cui sarà installato il progetto completo è su piattaforma Linux. Il discreto grado di difficoltà di progettazione causato dalla poca conoscenza del linguaggio portano ad escludere l'utilizzo del PHP a favore di altri linguaggi multipiattaforma meglio conosciuti.

È stato pertanto deciso che la tecnologia di riferimento e il linguaggio utilizzato verteranno sulle Java Server Pages, affiancate dai canonici HTML e

JavaScript. Sarà inoltre eventualmente possibile avvalersi di XML e di fogli di stile XSLT.

2.3 Requisiti specifici

2.3.1 Parte studente

Per autenticarsi, lo studente deve scegliere la materia dalla lista degli esami e digitare il proprio numero di matricola completo: può accedere al testo del compito solamente se il suo numero di matricola appare nella lista d'esame; in caso contrario un messaggio di errore notificherà l'impossibilità di affrontare l'esame e il sistema tornerà alla pagina del login. Si presuppone che il controllo tra nome e persona venga fatto in laboratorio richiedendo un documento d'identità dato che è necessario associare univocamente la persona che sta svolgendo la prova d'esame con l'utente che si è autenticato all'esame, in quanto non è possibile controllarlo automaticamente utilizzando i dati forniti dal sistema operativo.

Nel browser dello studente che si è regolarmente autenticato devono essere sempre visualizzati i dati personali del candidato (nome, cognome, numero di matricola) e il numero del compito.

Lo studente che ha effettuato correttamente il login può leggere il testo dell'esame, rispondere ad alcune domande a risposta multipla (se l'esame lo prevede) e consegnare i file che produce per tutta la durata dell'esame. Lo studente può consegnare, prelevare, rimuovere i file prodotti attraverso la pagina di consegna, che mostra anche una lista dei file inviati.

Lo studente che fa logout dalla pagina dell'esame (chiude la pagina del browser, si blocca il PC, ...) non deve più poter accedere all'esame neanche dalla stessa postazione.

Il docente deve poter inserire nel sistema la data del proprio esame e la lista degli studenti iscritti all'esame scaricata da Uniwex. Il docente può decidere il numero di compiti da inserire (ovvero in quanti gruppi deve essere suddiviso il laboratorio) e può scegliere le associazioni tra le macchine e il testo

dell'esame. Può anche inserire delle domande a risposta multipla.

Al momento del login, vengono caricati nella pagina dell'esame il testo dell'esame ed eventuali domande a risposta multipla.

Il testo dell'esercizio e delle domande vengono definiti in base al posizionamento dello studente nel laboratorio.

La posizione dello studente nel laboratorio dipende dal numero della macchina, che rispecchia l'indirizzo IP all'interno della LAN.

2.3.2 Tool di amministrazione per il docente

Si vuole creare un tool di amministrazione che permetta al docente di controllare la situazione real time degli studenti in laboratorio durante un esame.

Il docente può creare differenti appelli d'esami, specificando data, indirizzi IP con il permesso di accedere all'esame, turni d'esame, i gruppi, e la presenza di domande a risposta multipla per ogni turno. Il docente, o l'esaminatore per lui, può modificare, eliminare o aggiungere turni d'esame.

Prima di iniziare l'esame il docente deve inserire il file contenente la lista dell'esame, in modo che l'accesso al sistema sia permesso solo agli utenti iscritti. All'inizio dell'esame il docente deve inserire la lista d'esame, il testo dell'esame sottoforma di pagine HTML ed eventualmente anche informazioni relative al compito, alla consegna o alle domande (se previste) sempre sottoforma di file HTML e con nomi predefiniti. Le domande devono essere inserite attraverso un apposito file di configurazione XML e il file contenente le domande deve avere un nome differente in base al gruppo a cui sottoporre le domande.

All'inizio dell'esame il docente può fare l'appello degli studenti e definire quali di quelli iscritti alla prova possono autenticarsi al sistema. Uno studente definito come assente equivale ad uno studente non iscritto in lista. Deve essere possibile aggiungere altri studenti alla lista d'esame, in modo da poter eventualmente permettere, a discrezione del docente, a chi non è in lista di poter affrontare il compito. La lista degli studenti deve poter essere modificabile in qualunque momento dell'esame per permettere anche agli studenti

in ritardo segnalati come assenti di autenticarsi al sistema.

Il docente autenticato può visualizzare la disposizione degli studenti nel laboratorio mediante un'apposita mappa in cui, per ogni postazione, può vedere alcune informazioni sullo studente e sullo stato del suo esame. In particolare, per ogni postazione (di cui si conosce già l'indirizzo IP), oltre a nome, cognome, e numero di matricola dello studente deve essere visualizzato il numero/lettera del compito e la lista dei file inviati.

Il docente può ri-abilitare l'accesso di uno studente all'esame sulla stessa postazione o un'altra nel caso in cui, una volta che questi avesse concluso la propria sessione di lavoro, non possa più accedere alla pagina dell'esame.

Il docente inoltre può decidere di cambiare il tipo di compito assegnato agli studenti: in questo caso vengono eliminati tutti gli eventuali file già inviati dallo studente.

Il docente può monitorare solamente la situazione del laboratorio del proprio esame: non può visualizzare la situazione dell'esame di un collega.

Al termine dell'esame il docente deve poter impedire ulteriori consegne al sistema. Terminando l'esame non deve più essere possibile per uno studente autenticarsi al sistema, ed eventuali file consegnati dopo il termine dell'esame non devono essere considerati.

Capitolo 3

Analisi di casi d'uso e scenari

I casi d'uso e i relativi scenari permettono di formalizzare i requisiti funzionali, di comprendere meglio il funzionamento del sistema (e quindi di metterne in evidenza eventuali carenze) e di comunicare meglio con il cliente.

Questi sono stati infatti utilizzati per descrivere al docente quali punti dovevano essere affrontati in quanto l'insieme di tutti i casi d'uso costituisce l'immagine del sistema verso l'esterno.

3.1 Analisi degli utenti coinvolti

Dall'analisi precedentemente fatta è emerso che gli utenti saranno principalmente tre: il docente, l'esaminatore e lo studente. Per motivi organizzativi è stato scelto di dividere la figura del docente tra esaminatore e docente vero e proprio poiché risulta più comodo considerare la figura complessiva del docente come formata dalle funzioni relative a creazione, gestione e controllo delle prove, e dalle funzioni relative alla gestione delle utenze che possono “vedere” gli esami del docente in atto.

Nel sistema infatti il docente è una specializzazione dell'utente “esaminatore” dato che ne condivide tutti i casi d'uso e ha in più la possibilità di delegare altri esaminatori per il proprio insegnamento.

3.2 Analisi dei casi d'uso

Per “caso d'uso” si intende la descrizione di una richiesta fatta al sistema da una qualsiasi entità esterna al sistema stesso (attore) e definisce un insieme di scenari legati da un obiettivo comune.

3.2.1 Analisi dei casi d'uso dello studente

Uno studente può semplicemente affrontare l'esame, comprensivo di tutte le sue funzionalità, compreso rispondere alle domande, se la prova lo prevede. Dall'analisi dei requisiti emerge che lo studente può effettuare nel sistema un insieme limitato di azioni, che sono limitate a (può effettuare solamente l'esame).

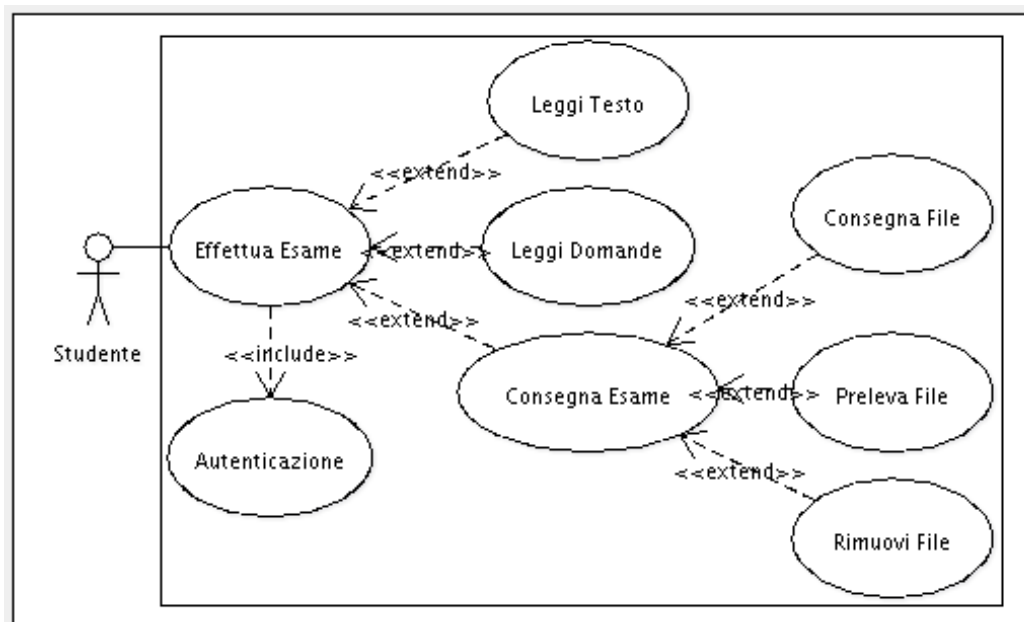


Figura 3.1: Elenco dei casi d'uso dello studente

La figura 3.1 mostra che nessuna operazione può essere compiuta dallo studente senza che questi si sia correttamente autenticato al sistema.

3.2.2 Analisi dei casi d'uso del docente

Come emerso dall'analisi dei requisiti, nella parte docente sono stati individuati due potenziali utenti del sistema: il docente e l'esaminatore.

L'esaminatore può compiere tutte le azioni di creazione, gestione e controllo delle prove, mentre il docente ha la possibilità aggiuntiva di scegliere quali altri esaminatori possono gestire le proprie prove.

La figura 3.2 mostra quali sono le relazioni tra l'utente "docente" e l'utente "esaminatore" mettendo in risalto le possibilità che il docente ha in più ri-

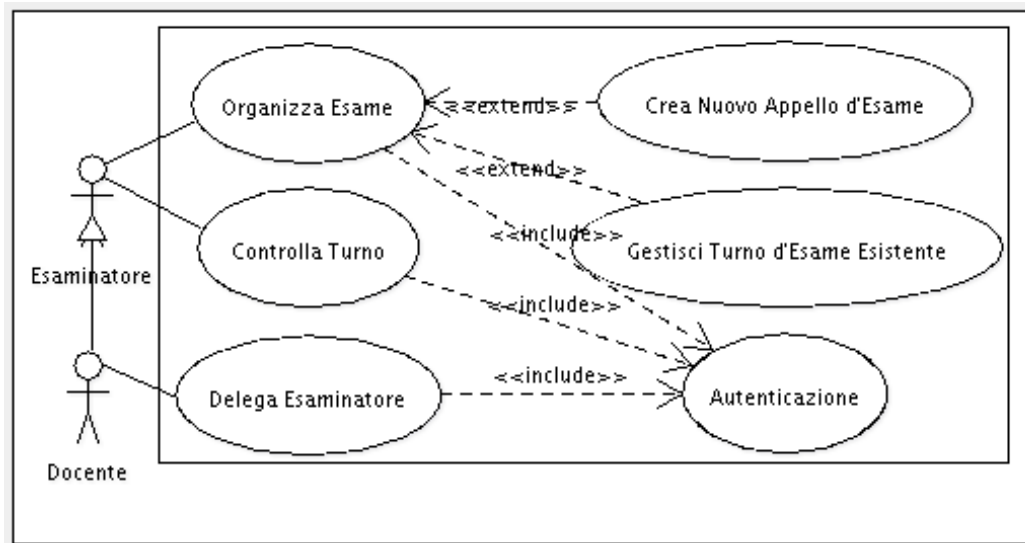


Figura 3.2: Elenco dei casi d'uso di docente ed esaminatore

spetto all'esaminatore, e mostrando una visione ad alto livello dei casi d'uso dell'esaminatore.

3.2.3 Analisi dei casi d'uso dell'esaminatore

I due grandi casi d'uso a cui è collegata la figura dell'esaminatore riguardano l'organizzazione degli appelli d'esame e dei turni ad essi correlati, e il controllo degli stessi.

L'organizzazione degli appelli d'esame è a sua volta suddivisa tra creazione dell'appello e gestione dei turni appartenenti ad un appello già creato.

La creazione di un nuovo appello d'esame prevede che l'utente possa definire in una data a sua scelta un esame avente uno o più turni. In questo caso d'uso l'esaminatore definisce le impostazioni basilari dell'esame, ovvero il numero dei turni totali, la suddivisione in gruppi per ogni turno, la possibilità di porre agli studenti domande a risposta multipla, il laboratorio o gli indirizzi IP dai quali sarà possibile per gli studenti accedere al sistema.

Ogni impostazione definita durante la creazione dell'esame può essere ampiamente modificata.

L'esaminatore ha la possibilità di modificare ognuna delle voci espresse du-

rante la creazione dei turni appartenente all'appello d'esame creato. La modifica può anche implicare una successiva creazione di turni aggiuntivi all'appello d'esame, o l'eliminazione di turni che saranno verificati come eccedenti. E' eventualmente anche possibile per l'esaminatore eliminare determinati turni, arrangiando automaticamente i turni rimasti aggiornando il numero dei turni totali e riordinando il numero dei turni per evitare che ci siano dei "buchi" tra i turni d'esame.

L'esaminatore può inoltre inserire i file relativi al turno che sta per iniziare,

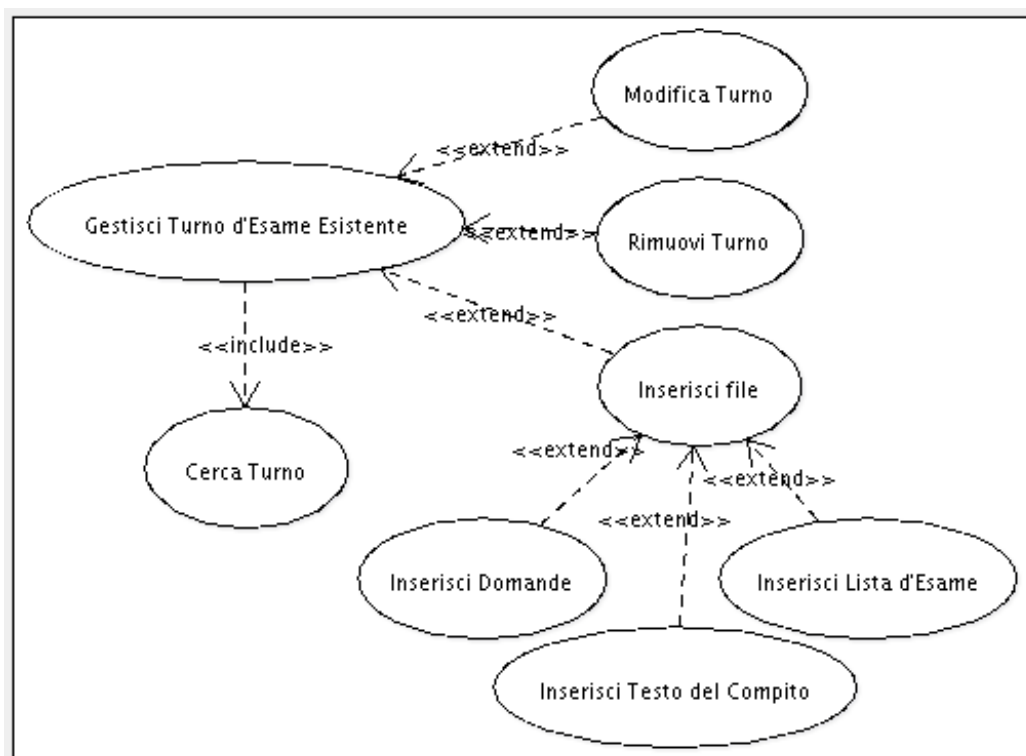


Figura 3.3: Casi d'uso inerenti alla gestione di esami esistenti

ovvero il file comprendente la lista degli studenti iscritti, ed eventualmente, se la prova lo prevede, i file relativi al testo del compito e alle domande a risposta multipla, chiamati in modi differenti in base al gruppo. L'esaminatore può inoltre decidere di inserire file di avvertimento per l'esame, la consegna o le domande.

Il controllo dell'esame estende sempre la ricerca di un turno d'esame esistente. L'esaminatore può iniziare un turno d'esame e fare l'appello degli studenti,

sulla base del file della lista che deve avere inserito in precedenza.

Durante la prova, l'esaminatore può controllare la situazione del laboratorio, per vedere dove si è autenticato uno studente e cosa ha consegnato fino al momento attuale. L'esaminatore può inoltre decidere di cambiare i gruppi di appartenenza degli studenti, o cambiare la postazione in caso di guasti, previa ricerca dello studente all'interno del laboratorio.

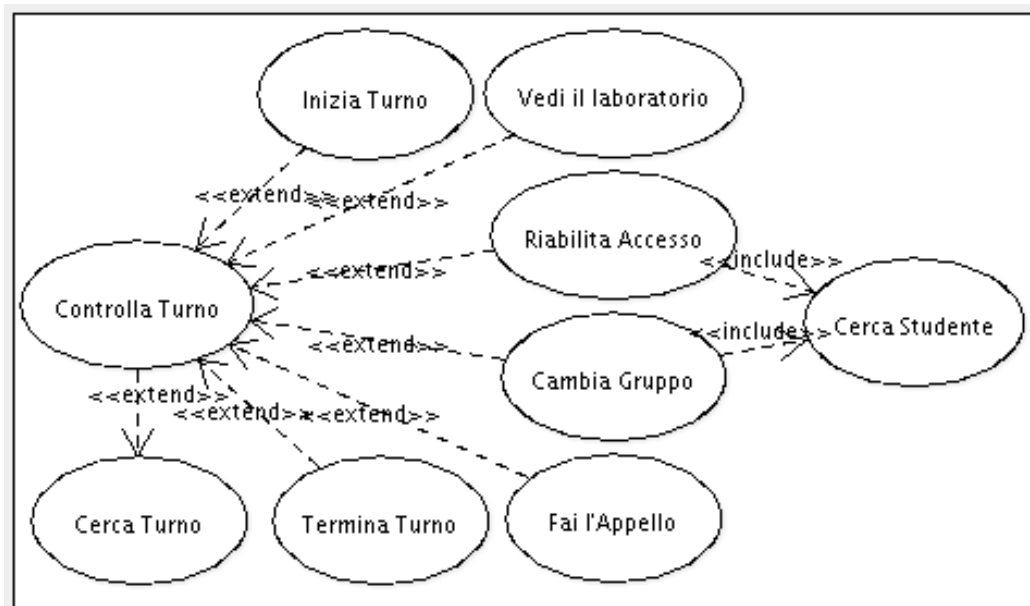


Figura 3.4: Casi d'uso inerenti ai controlli del turno in atto

Al termine dell'esame, per impedire ulteriori consegne, l'esaminatore può terminare il turno in atto. In questo modo non sarà più possibile per gli studenti accedere al sistema.

3.3 Analisi degli scenari

Verranno di seguito descritti gli scenari principali, o degni di nota dei tre attori individuati nel sistema.

3.3.1 Scenari dello studente

Gli scenari dello studente sono molto semplici e non necessitano di grandi spiegazioni. Vengono quindi di seguito mostrati gli scenari relativi ai casi

d'uso di maggiore interesse e complessità.

Effettua esame

Introduzione: Lo studente vuole effettuare l'esame.

Relazioni: Il caso d'uso in questione si può relazionare con i seguenti casi d'uso: *Leggi testo, Consegna file, Leggi domande.*

Attori: studente

Precondizioni: L'utente è iscritto alla lista d'esame.

Flusso principale:

1. L'utente entra nel sistema (*login*)
2. L'utente inizia l'esame
3. L'utente legge il testo ed elabora i suoi file
4. L'utente naviga tra le domande e risponde
5. L'utente consegna i propri file
6. L'utente termina l'esame
7. L'utente esce dal sistema (*logout*)

Consegna file

Introduzione: Lo studente vuole consegnare alcuni file che ha prodotto.

Relazioni: Il caso d'uso in questione si può relazionare con i seguenti casi d'uso: *Preleva file, Rimuovi file.*

Attori: studente

Precondizioni: L'utente è entrato nel sistema.

Flusso principale:

1. L'utente si posiziona nella pagina di consegna
2. L'utente cerca nel proprio disco il file da consegnare e lo seleziona
3. Il sistema carica il file selezionato
4. Il sistema aggiunge alla lista il file che è appena stato inserito
5. Il sistema è pronto ad una nuova operazione

3.3.2 Scenari del docente

L'unico scenario che attualmente contraddistingue il docente dall'esaminatore riguarda la possibilità per il docente di avere una directory propria in cui far confluire i propri esami, e di poter delegare altri esaminatori.

Delega utente

Introduzione: Il docente vuole permettere a determinati utenti di creare e controllare i propri esami, oppure vuole negare tale permesso ad utenti a cui aveva già dato la delega.

Attori: docente

Flusso principale:

1. Il docente seleziona gli utenti che non possono ancora creare i propri esami, o gli utenti che non devono più poter creare esami
2. Il docente conferma la propria scelta
3. Il sistema aggiorna i diritti degli utenti selezionati
4. Il sistema è pronto per una nuova operazione

3.3.3 Scenari dell'esaminatore

Modifica un turno d'esame esistente

Introduzione: l'esaminatore vuole modificare alcuni dati relativi ad un turno di un appello d'esame

Attori: esaminatore

Relazioni: Cerca turno

Precondizioni: L'esame esiste già. L'esaminatore ha trovato il turno da modificare.

Flusso principale:

1. L'esaminatore compila i campi che erano stati lasciati vuoti, o che vuole modificare
2. L'esaminatore conferma le modifiche
3. Il sistema è pronto per una nuova operazione

Flusso alternativo A

1. L'esaminatore ha modificato la data del turno dell'esame
2. Il sistema cerca tutti i turni appartenenti all'appello d'esame del turno selezionato e ne modifica la data

Flusso alternativo B

1. L'esaminatore inserisce un numero di turni totali minore del valore attuale
2. Il sistema elimina i turni appartenenti all'appello d'esame che contiene il turno selezionato che hanno un turno maggiore di quello selezionato
3. Il sistema aggiorna il numero dei turni totali per tutti gli esami appartenenti all'appello d'esame del turno selezionato

- Flusso alternativo C**
1. L'esaminatore inserisce un numero di turni totali maggiore del valore attuale
 2. Il sistema aggiorna il numero dei turni totali per tutti gli esami appartenenti all'appello d'esame del turno selezionato
 3. Il sistema aggiunge i turni eccedenti come copia dell'ultimo turno esistente dell'appello d'esame

Rimuovi un turno d'esame esistente

Introduzione: l'esaminatore vuole eliminare un turno appartenente ad un appello d'esame già programmato

Attori: esaminatore

Relazioni: Cerca turno

Precondizioni: L'esame esiste già. L'esaminatore ha trovato il turno da eliminare.

- Flusso principale:**
1. L'esaminatore seleziona il turno da modificare
 2. Il sistema chiede conferma all'utente
 3. Il sistema elimina il turno e aggiorna i turni dell'esame rimasti
 4. Il sistema è pronto per una nuova operazione

- Flusso alternativo A**
1. Il turno selezionato è l'unico turno dell'appello d'esame
 2. Il sistema elimina ogni riferimento all'appello d'esame

- Flusso alternativo B**
1. Il turno selezionato è l'ultimo tra i turni dell'appello d'esame
 2. Il sistema elimina il turno selezionato
 3. Il sistema decrementa il numero dei turni totali dell'appello d'esame

- Flusso alternativo C**
1. L'appello d'esame con più di due turni e il turno selezionato non è l'ultimo
 2. Il sistema elimina il turno selezionato
 3. Il sistema decrementa il numero del turno per i turni dell'appello maggiori del turno eliminato
 4. Il sistema decrementa il numero dei turni totali dell'appello d'esame

Fai l'appello

Introduzione: Il docente vuole fare l'appello degli studenti per identificarli e per poter definire esattamente quali studenti possono accedere e autenticarsi all'applicazione lato-studente. Il docente può inoltre aggiungere altri studenti che non erano nella lista d'esame in modo che possano accedere al sistema d'esame.

Attori: esaminatore

Relazioni: Cerca turno

Precondizioni: Il turno esiste già. L'esaminatore ha trovato il turno.

Flusso principale:

1. L'esaminatore seleziona il turno
2. Il sistema mostra la lista degli studenti iscritti al turno
3. L'esaminatore chiama ogni studente e definisce se è presente o assente
4. L'esaminatore salva le modifiche
5. Il sistema salva la lista aggiornata

Flusso alternativo

1. L'esaminatore aggiunge i dati di uno studente non iscritto alla lista
2. L'esaminatore conferma la scelta
3. Il sistema aggiunge alla lista i dati del nuovo studente

Capitolo 4

Progetto e collaudo della parte studente

Questo capitolo tratta della progettazione della prima parte dell'applicativo, ovvero il lato riferito allo studente e che verrà utilizzato durante la prova in laboratorio. Il progetto di questa nuova versione di EsamiLIA, progettato in modo diverso e in cui sono saranno aggiunti tutti quei piccoli accorgimenti che il docente desidera, deve essere un progetto indipendente e poter funzionare anche senza la presenza del tool per il docente.

È stato deciso che per questioni di comodità la parte studente avrà la stessa veste grafica di EsamiLIA, utilizzando quindi lo stesso file CSS [1] e cercando di adattare la disposizione degli oggetti nelle pagine come gli studenti erano abituati a vederlo. Lo scopo principale è quindi che “ai morsetti” non si noti la differenza con la vecchia applicazione.

L'applicazione inoltre sarà sotto forma di Web Application in modo da garantire velocità nella progettazione, semplicità di utilizzo, e standardizzazione delle tecniche da utilizzare. Non sarà più necessario nascondere la barra degli indirizzi o far utilizzare allo studente un eseguibile utilizzabile solamente dalle piattaforme Windows che altro non fa che chiamare il server nella pagina di scelta degli esami.

Le tecnologie utilizzate si baseranno su linguaggi server-side (Java Server Pages) e client-side come JavaScript per non appesantire troppo il server nel caso di sottoscrizioni incomplete o scorrette di form, dato che si stima la pos-

sibilità di un'ottantina di connessioni simultanee sul server, corrispondenti al numero di postazioni presenti in laboratorio.

4.1 Progetto dell'applicativo

La parte studente deve rendere disponibile allo studente, una volta autenticatosi, la pagina contenente il testo dell'esercizio, le domande a risposta multipla (se previste) e la pagina per consegnare i propri file e per vedere quelli già consegnati, in modo da poterli riscaricare o eventualmente eliminare dal server. Ognuna di queste pagine segue lo schema di frame che ha EsamiLIA e richiede la presenza di tre pagine HTML chiamate esattamente come nel vecchio applicativo *infoese.html*, *infodom.html* e *infocon.html* per mostrare in ognuna di queste pagine delle avvertenze relative all'esercizio, alle domande, o alla consegna. In questa primo sviluppo della nuova applicazione è stato deciso che i file dovranno chiamarsi esattamente come prescritto: non sarà quindi per ora consentito di utilizzare file con nomi diversi per supplire alle stesse funzioni.

Anche l'autenticazione iniziale seguirà lo schema base di autenticazione già utilizzato, permettendo però di settare la presenza di studenti assenti e che quindi non avranno modo di accedere al sistema.

Una "conversazione" rappresenta una sequenza di pagine di senso compiuto ed è univocamente definita dall'insieme delle pagine che la compongono e dall'insieme delle interfacce di input/output per la comunicazione tra le pagine (detto flusso della conversazione). La sessione quindi, condivisa tra Client e Server, deve essere associata ad una o più conversazioni effettuate da un singolo utente ed appartiene ad esso: una sessione non può mai essere condivisa tra più utenti.

Nel nuovo EsamiLIA sono state individuate due sole conversazioni: una relativa all'autenticazione dello studente, e una relativa all'esecuzione dell'esame. Queste due conversazioni verranno poi mutuamente richiamate, o riciclate con stati differenti, che mostreranno all'utente informazioni diverse.

La "sessione" rappresenta lo stato associato ad una sequenza di pagine visualizzate da un utente: contiene tutte le informazioni necessarie durante

l'esecuzione, che possono essere ad esempio informazioni di sistema (IP di provenienza, lista delle pagine visualizzate, ecc) oppure informazioni di natura applicativa (nome e cognome, username, ecc). La sessione è inoltre l'unico stato necessario per la gestione di una applicazione web. Esistono due tecniche di base per gestire la sessione: attraverso l'utilizzo della struttura dei cookie, oppure attraverso la gestione di uno stato sul server per ogni utente collegato.

Nel primo caso, i cookie fanno parte dello standard HTTP, sono quindi sempre disponibili, mentre per quanto riguarda la seconda opportunità, la gestione dello stato è possibile in alcune architetture specifiche. Si è quindi pertanto di utilizzare dei cookie per gestire la sessione del Client.

4.2 Autenticazione dello studente

All'avvio della prova, lo studente deve dimostrare di essere regolarmente iscritto all'esame avendo utilizzato il servizio di Ateneo Uniwex. Attraverso Uniwex infatti è possibile scaricare la lista degli utenti iscritti all'esame in formato CSV o in formato XML utilizzando in entrambi i casi una struttura immutabile e che mostra alcuni dati degli studenti.

Per compatibilità con l'applicativo preesistente, si è mantenuto il controllo sull'iscrizione alla lista d'esame attraverso il file CSV rinominato come *lista.txt*.

È stato inoltre aggiunta la possibilità per il docente di indicare come assente uno studente precedentemente iscritto. Per semplicità è stato deciso di utilizzare come parametro di default per settare l'assenza dello studente, l'inserimento su Uniwex del voto "CANC" allo studente.

Al login lo studente deve inserire il proprio numero di matricola completo. Al tentativo di accesso, il sistema controlla la lista dell'esame e controlla se lo studente appare iscritto alla lista e non è stato segnalato come assente. Se la matricola non è corretta, o lo studente è assente, il sistema avvisa lo studente dell'errore ritornando alla pagina iniziale.

In caso contrario, il sistema crea una directory del tipo *Cognome_Matricola_IdIscrizione* e all'interno, esattamente come EsamiLIA, un file che funge da flag di login che si chiama *_Pxx-Gyy* con *xx* le ultime cifre dell'indirizzo

IP della macchina da cui lo studente accede e *yy* il gruppo di appartenenza. Questo file contiene una serie di informazioni di comodo e alcune informazioni relative all'esame, quali nome, cognome, numero di matricola, identificativo numerico di iscrizione all'esame, indirizzo IP completo del PC da cui sono acceduti e il timestamp del primo login al sistema.

Dopo aver creato la directory e il file di flag il sistema crea dei cookie con alcuni dati dello studente e ricalcola la pagina iniziale con lo stato di login effettuato. Il sistema quindi mostra a video il nome dello studente e lo invita ad accedere alla pagina dell'esercizio.

4.2.1 L'unicità del login

Per ovvi motivi, è necessario garantire che l'autenticazione al sistema sia effettuata solamente dallo studente stesso e, a meno di esplicito intervento del docente, solo dalla macchina da cui ha compiuto il primo login, in quanto il dato che permette l'autenticazione è il numero di matricola dello studente, dato questo facilmente conoscibile e conseguentemente utilizzabile con scopi maligni.

Ad ogni login il sistema controlla se esiste già una directory con gli stessi parametri dello studente e il file di flag. Per evitare che lo studente possa autenticarsi più di una volta, il sistema controlla prima di tutto se esiste un file che inizia con *_P*, e in tal caso si accorge che è già stato effettuato un login per lo studente e ne impedisce un successivo. Questa soluzione, adottata solo in un primo momento, non permetteva non solo che studenti malintenzionati potessero autenticarsi con la matricola di uno studente che era già acceduto, ma impediva anche allo stesso studente che per sbaglio avesse chiuso la finestra di autenticazione o che per qualunque motivo avesse dovuto ritentare il login, di accedere. È allora stato deciso che il sistema controlli tre momenti differenti:

- Se il sistema non trova alcun file *_P*, capisce che è il primo accesso, e quindi crea il nuovo file di flag e lo studente può fare normalmente il login
- Se il sistema trova un file che ha il nome uguale al nome del file che creerebbe se fosse il primo login dello studente, allora lo studente può

fare un ulteriore login e accedere di nuovo. Si noti che questa impostazione è facilmente modificabile e nel caso in cui venisse scelto che l'utente può accedere al sistema solo una volta, basterebbe cambiare il comando relativo allo stato del sistema nel caso di flag corretto trovato.

- Se viene trovato un altro file di flag con impostazioni differenti da quelle che creerebbe, il sistema mostra a video un messaggio di errore e logga tutti i dati del secondo utente che cerca di effettuare l'accesso.

4.2.2 Il proxy e l'identificazione dell'indirizzo IP

Un problema contro il quale ci si è scontrati è quello di riuscire a definire con chiarezza l'indirizzo IP della macchina dello studente che tenta l'accesso.

Il problema principale deriva soprattutto dalla presenza di un proxy tra la sottorete del laboratorio, in cui tutti gli indirizzi IP sono indirizzi privati, e la suddivisione in gruppi è fatta in base a questi indirizzi privati, e il server che dovrà essere esterno per permettere in futuro all'applicazione docente di essere acceduta anche da PC esterni al laboratorio.

Senza la possibilità di determinare con precisione l'IP locale della macchina risulta inoltre impossibile fare tutti i controlli sui doppi login.

Presupponendo che il proxy sia abbastanza trasparente, dovrebbe essere semplice ricavare l'indirizzo IP del client attraverso la stringa passata attraverso la Request.

```
request.getHeader("HTTP_X_FORWARDED_FOR")
```

Se il proxy però non è abbastanza trasparente, ciò che vedrebbe il server sarebbe solamente l'indirizzo del proxy e tutti gli studenti apparirebbero come autenticati dalla stessa macchina, e verrebbe loro assegnato lo stesso gruppo di compito.

Se non ci fosse un proxy invece basterebbe semplicemente la seguente stringa a mostrare l'indirizzo IP della macchina.

```
request.getRemoteAddr()
```

Una possibile soluzione che scavalchi la figura del server e memorizzi l'indirizzo IP locale della macchina può essere anche attuata lato client mediante uno script JavaScript. [6] Questa soluzione si basa sull'utilizzo di Java, ed è

quindi compatibile solo con browser della famiglia Netscape, compresi Mozilla e Firefox; persino Opera supporta questa funzionalità. Sfortunatamente questa soluzione non può essere adottata in tutte le situazioni, in particolare modo nel caso in cui gli studenti stiano utilizzando Windows in laboratorio; Internet Explorer è infatti il browser di default di questo sistema operativo e non è stata implementata la possibilità di utilizzo di codice Java all'interno degli script JavaScript. Impedire l'accesso all'applicazione solamente ai browser di tipo Internet Explorer sarebbe molto semplice, ma sarebbe anche una scelta piuttosto forte, e quindi non sarà considerata.

4.3 Sezione contenente le domande

La pagina delle domande presuppone l'esistenza di un file XML nominato allo stesso modo dei file *DomandeN.ini* utilizzati in EsamiLIA. A differenza dei file utilizzati nel vecchio sistema però ha una struttura semplificata, che contiene tanti oggetti quante sono le domande da porre agli studenti; e all'interno di questi ci sono le descrizioni delle domande e delle risposte, che verranno quindi stampate a video con una formattazione opportuna e nello stesso ordine in cui sono presentate sul file XML.

La struttura del file XML è simile a quella riportata in figura A.1.

Come si può notare, non c'è riferimento a quale tra le risposte sia quella corretta e, a differenza del file *DomandeN.ini* utilizzato in EsamiLIA, non è più necessario inserire riferimenti al numero totale di domande né al numero totale di risposte per ogni domanda.

Seguendo lo schema dell'applicazione precedente, è stato deciso che le domande saranno a risposta multipla, ovvero un'unica risposta vera tra quelle proposte. Dovrebbe essere possibile in futuro scegliere di impostare le domande affinché per ogni possibile risposta possa essere scelto se l'affermazione è vera, falsa, o se non se ne conosce la veridicità, oppure potrebbe essere una buona idea contemplare eventuali domande aperte, in modo che all'interno del file *_domandeN.txt* venga salvata la risposta che lo studente potrebbe scrivere in un campo di testo.

Al momento attuale la pagina delle domande legge il file XML e attraverso un foglio di stile mostra a video le domande e le relative risposte. Lo studente

```
<Domande>
  <Domanda>
    <Question>
      Se si vuole che un metodo non possa essere ridefinito nelle
      classi derivate come si deve procedere?
    </Question>
    <Response>
      Il metodo deve essere definito come static
    </Response>
    <Response>
      Il metodo deve essere definito come final
    </Response>
    <Response>
      La classe deve essere marcata come final e il metodo deve
      essere marcato come static
    </Response>
  </Domanda>
  <Domanda>...</Domanda>
</Domande>
```

Figura 4.1: Schema del file *DomandeN.xml*

può rispondere alle domande, e premendo il pulsante “Invia” verrà salvato nel file *_domandeN.txt* il numero della domanda e la relativa risposta. Al successivo caricamento della pagina delle domande, lo studente può rivedere le risposte che ha già dato, cambiarle, e, premendo sul pulsante “Azzera”, può annullare tutte le risposte che ha dato eliminando il file dal server.

4.4 Sezione per la consegna dei file

La pagina per la consegna dei file contiene due ambienti: una form per la scelta del file da salvare sul server, e la lista dei file che sono attualmente salvati nella directory del docente, senza però contare il file di flag *_Pxx-Gyy* e l'eventuale file in cui sono salvate le domande *_domandeN.txt*. I file elencati e visibili dall'utente possono essere scaricati dallo studente, o eliminati. Queste opzioni sono state pensate e realizzate per permettere allo studente di fare

copie di backup durante la prova in modo da non perdere il lavoro in caso di guasti improvvisi.

4.4.1 Consegna dei file

Al fine di consegnare un file sul server, le azioni che lo studente deve compiere sono la ricerca del file in locale e la pressione del bottone di conferma. Alla convalida della form, il sistema deve essere in grado di prendere il file di cui lo studente ha lasciato il path e salvarlo nella directory riservata dello studente. Per fare l'upload di un file è necessario una form nella pagina web, e un metodo nella classe Java.

La form della pagina web deve avere un aspetto simile a questo:

```
<form name='consegnaForm' action='insert.jsp?status=add'  
  method='post' enctype='multipart/form-data'>  
  <input type='file' size='60' name='cmuds' />  
  <input type='button' onClick='consegnaForm.submit()'  
    value='Consegna' />  
</form>
```

Figura 4.2: Schema della form per l'upload di un file

Questa form presenta alcune particolarità da tenere in conto nel caso di aggiornamenti futuri.

Nel tag `<form>` è infatti inserito un elemento insolito, `enctype='multipart/form-data'` che è necessario per inviare sul server i dati relativi al file selezionato.

Nel caso in cui all'interno della form, oltre al campo di tipo "file", vengano aggiunti altri tag di tipo `<input>` da inviare tramite la Request, il comportamento della form non è predicibile.

Sono state quindi trovate due possibili alternative per aggirare questo ostacolo affinché il servizio di upload sia corretto e funzionante: si può scegliere di passare tutti i parametri, compreso il nome del file, attraverso il metodo GET, oppure si può utilizzare un metodo "ibrido" che permette di inserire il nome del file normalmente in POST, mentre tutti gli altri campi che sarebbero stati "hidden", attraverso la notazione in GET.

```
action='insert.jsp?status=add'
```

La prima soluzione non è prudente perché i parametri passati in GET sono visibili e facilmente modificabili, mentre la seconda soluzione non è elegante, ma funzionante. È stato per questo motivo deciso di utilizzare la seconda opportunità.

Per l'upload vero e proprio dei file è stato utilizzato un package disponibile per Apache - Tomcat chiamato Commons File Upload [7]. FileUpload elabora le richieste HTTP conforme all'RFC "Form-based File Upload in HTML" [8]. Questo significa che, se una richiesta HTTP è accettata usando il metodo POST, e con un content type di tipo "multipart/form-data", il FileUpload può elaborare la Request, e rendere i risultati disponibili in un modo facilmente utilizzabile dal chiamante.

Ciò che non è ancora stato implementato e che potrebbe essere progettato in futuro è la possibilità per lo studente di inserire più file contemporaneamente.

Nel caso di caricamento di un file avente lo stesso nome di uno di quelli già presenti, il sistema sovrascrive il file esistente con quello nuovo.

4.4.2 Prelievo e rimozione dei file

Per quanto riguarda le operazioni che lo studente può compiere nella sezione che mostra la lista dei file già consegnati, il sistema trasforma il path assoluto del file che ha trovato sul server al path relativo dell'esame e rende disponibile all'utente il collegamento ai file della propria directory affinché possa scaricarli. Il sistema, nel caso di download dei file, non è in grado di riportare sul file di log l'azione in quanto si tratta della semplice operazione di click su un link che porta a un file, senza avvertire il server, e quindi senza dargli la possibilità di tenere traccia, nel caso di possibili bug, di chi ha scaricato i file di uno studente, o di quante volte li abbia scaricati.

Per ogni file esiste anche un bottone che permette all'utente di eliminare il file scelto. Nel caso di eliminazione di un file, il sistema riporta sul file di log l'azione ed elimina completamente dal server ogni riferimento al file: è stato deciso di non mantenere un sistema di versioning nelle directory degli studenti che monitori e salvi i file inseriti inizialmente, poi sovrascritti o eliminati.

4.4.3 Problemi di path con Internet Explorer

Il progetto, completamente compatibile con Mozilla Firefox e Netscape Navigator, ha dato alcuni problemi nell'inserimento e download dei file durante l'uso dell'applicativo con Internet Explorer.

Contrariamente a Firefox che all'upload, indipendentemente dalla piattaforma, riporta solamente il nome del file da caricare, Internet Explorer passa come parametro il path completo del file, a cominciare da *C:*. Si è quindi resa necessaria la modifica al codice inerente all'upload del file in cui, poco prima della creazione del file sul server, viene determinato il nome finale del file, inteso come la parte terminale del path, e viene salvato il file nella directory riservata allo studente definita dal server, con il nome del file. In questo modo viene aggirato il problema del path completo dato da Windows.

Un altro problema riscontrato con Internet Explorer è relativo alla presenza di spazi all'interno del nome del file. Internet Explorer infatti sostituisce gli spazi con la stringa "%20" quindi nel tentativo di download di file contenuti nel nome degli spazi, il sistema avverte l'utente dell'errore, per non aver trovato il file che cercava. In seguito a questa incompatibilità di Internet Explorer si è resa necessaria una ulteriore modifica al sistema: in questo caso si sono andate a modificare le impostazioni di creazione del file sul server, cercando l'esistenza di eventuali caratteri di spazi, convertendoli in "underscore". Procedendo in questo modo si è eliminato alla radice il problema di possibili nomi di file contenenti spazi.

4.5 Test e collaudo

L'usabilità di questa parte dell'applicativo deve essere dimostrata mediante un collaudo che evidenzia lacune e anomalie.

Le sessioni di prova non si sono fermate alla verifica della funzionalità del sito, ma hanno anche incluso le attività necessarie a dimostrarne la navigabilità, l'accessibilità, la compatibilità coi vari browser e le varie versioni degli stessi ed in generale il rispetto delle regole di usability.

Oltre ai test effettuati dal progettista, il collaudo del progetto è stato condotto da persone esterne, ovvero il sistema è stato testato in laboratorio durante

l'appello estivo dell'esame di Fondamenti di Informatica L-A del professor Denti in cui gli studenti erano stati lasciati all'oscuro del cambiamento né era stato detto loro come utilizzarlo. Hanno partecipato al collaudo della parte studente 14 studenti di Ingegneria Informatica che non hanno avuto alcun tipo di problema, e hanno contribuito tutor e docente, a dare feedback sulla parte relativa alla creazione delle directory degli studenti.

L'esperienza ha indicato come opportuno la rielaborazione alcune frasi da inserire nel file di log dato che non è sempre immediato riconoscere le operazioni compiute da un singolo utente, in particolar modo nel caso di anomalie inerenti la consegna dei file; nel caso specifico uno studente non aveva consegnato tutti i file richiesti nella consegna dell'esame, ed alcuni erano file compilati mancanti dei sorgenti, o sorgenti che non compilavano in cui mancava ovviamente il file compilato.

4.5.1 Controllo della visibilità

È stato verificato che la veste grafica non subisce grandi cambiamenti tra l'utilizzo di un browser ed un altro. Non sono presenti neanche bruschi cambiamenti di interfaccia grafica tra una pagina e l'altra, ma si è cercato di mantenere uniformità tra le pagine dell'applicazione, e si è anche cercato di mantenere uniformità con le stesse pagine di EsamiLIA.

L'utilizzo di un CSS per uniformare il layout delle pagine è stato inoltre molto utile e ha permesso di generare velocemente pagine il cui testo appartiene alla stessa famiglia di caratteri, ha lo stesso sfondo, ecc.

4.5.2 Collaudo con diversi browser e sistemi operativi

Il sistema è stato collaudato effettuando i test su sistemi Linux (Gentoo e RedHat) e Windows (Windows XP e Windows NT) cercando di simulare al meglio i PC del laboratorio, utilizzando i browser più probabili per i due sistemi, ovvero Internet Explorer 5.5, Internet Explorer 6.0, Mozilla Firefox 1.0.2, Mozilla 1.7, Netscape Navigator 7.0.

Oltre a qualche imperfezione derivata dalle particolarità di Internet Explorer che è stata prontamente modificata nel corso della progettazione (si notino infatti i problemi derivati dalla impossibilità di recuperare l'indirizzo IP lato client o tutti i problemi derivati dalla gestione dei file, intesi come download

e upload degli elaborati degli studenti) tutti i browser e i sistemi operativi hanno superato il collaudo, non presentando anomalie e disfunzioni.

4.5.3 Controllo dei contenuti

Fortunatamente questa sezione non ha avuto bisogno di grandi controlli dei contenuti, poiché la maggior parte del testo che viene visualizzata proviene dalle pagine del testo dell'esame e delle domande che vengono inserite dai docenti di volta in volta. In ogni caso, per quanto riguarda la gestione dei messaggi d'errore, o dei frammenti di testo dell'applicativo, il controllo è andato a buon fine.

Si è fatta inoltre grande attenzione a inserire in ogni pagina opportuni titoli aggiungendo gli elementi di tipo `<title>Titolo della pagina</title>`, ma non si sono inserite notizie di copyright.

4.5.4 Collaudo di funzionalità

Si è cercato di uniformare il codice HTML affinché soddisfi tutti i requisiti richiesti dallo standard del World Wide Web Consortium [9], in particolare è stato validato con successo il foglio di stile CSS, e si è cercato di aderire il più possibile allo standard di XHTML per le pagine JSP.

Capitolo 5

Progetto e implementazione della parte docente

Il tool di amministrazione per il docente deve basarsi sulla parte già progettata dell'applicativo e pertanto utilizza lo stesso tipo di architettura a multilivello che è stato utilizzato per la creazione del nuovo sistema per gli studenti.

Compito di questo tool è quello di automatizzare tutte quelle operazioni di creazione dell'esame e di reperimento degli elaborati che attualmente il docente è obbligato a fare a mano, entrando nello spazio del server.

Si rende pertanto necessario un sistema di login e di memorizzazione di utenti e dati che a sua volta richiede l'utilizzo di un database.

5.1 Progetto del database

Compito del database è quello di tenere traccia degli utenti del sistema, ovvero docenti ed esaminatori, e degli esami che da essi vengono impostati.

Docenti e esaminatori sono differenziati in base al diverso gruppo di appartenenza. Un docente infatti può creare i propri esami, mentre un esaminatore, a meno di esplicita abilitazione da parte di un docente, non può compiere alcuna azione sugli stessi. Viene dunque tenuta traccia delle abilitazioni degli utenti mediante associazioni tra questi utenti e i docenti per i quali possono creare esami.

Ad ogni esame possono essere associati diversi indirizzi IP o sottoreti dalle quali si può accedere alla parte studente dell'applicativo; ovvero vengono

determinati a priori gli indirizzi delle postazioni per cui è abilitato il login. Questa funzionalità viene realizzata associando l'identificativo dell'esame con l'indirizzo IP da abilitare.

Per i docenti il nome della directory sulla quale verranno memorizzati gli esami e gli elaborati degli studenti è associato al proprio username: è compito di una specifica tabella tenere traccia del nome tale directory.

Per ogni turno d'esame invece, una tabella apposita tiene traccia dei dati per la configurazione degli stessi, assieme ad uno stato che permette di conoscere l'avanzamento di tale esame. Associata alla tabella dei turni d'esame, una ulteriore tabella definisce gli indirizzi IP e le sottoreti che possono accedere all'esame.

La figura 5.1 mostra la struttura conseguente del database.

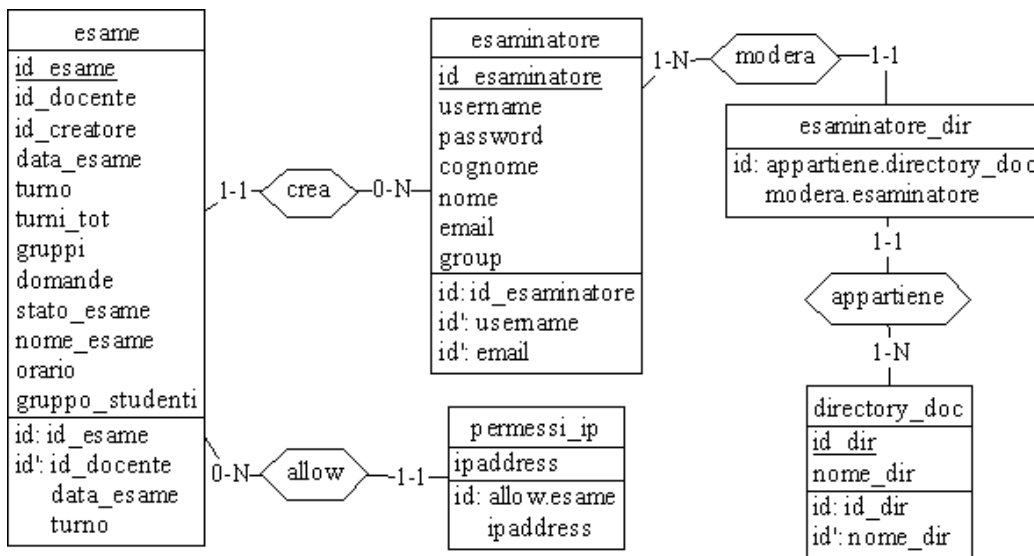


Figura 5.1: Struttura del database

Il database così progettato verrà installato sul DBMS PostgreSQL (versione 7.4.5) [10] in quanto tale è quello residente sul server installato in facoltà.

5.2 Registrazione dell'utente

Al fine di utilizzare il sistema, gli utenti devono autenticarsi: a tal fine devono essersi preventivamente registrati.

Attualmente non viene fatta alcuna verifica sull'identità di un utente che si registra: questa scelta, adottata per il momento per semplicità (ma modificabile in futuro inserendo protocolli di autenticazione più stringenti, ad esempio basati su LDAP) non è comunque causa di problemi in quanto:

1. il sistema è posto su rete interna protetta;
2. gli studenti conoscono i nomi dei loro docenti, quindi un docente "improbabile" verrebbe immediatamente riconosciuto sia da loro sia dagli altri docenti;
3. un utente esaminatore non ha comunque alcun diritto sul sistema se non viene all'uopo delegato da un docente.

Il sistema perciò, pur non imponendo particolari vincoli nella fase di registrazione, garantisce un livello di sicurezza ritenuto sufficiente per gli scopi che si propone, in quanto non permette comunque l'accesso ai dati di esame di altri utenti (docenti o esaminatori).

La differenza sostanziale tra docente ed esaminatore riguarda il fatto che per il primo viene anche creata una directory avente per nome lo username del docente, e ottiene automaticamente i diritti per scrivere sulla propria directory. Un esaminatore, al contrario, può agire solo su specifica disposizione di un docente.

5.2.1 Autenticazione tramite password

Il sistema prevede un meccanismo di autenticazione basato sull'hashing delle password attraverso algoritmi di cifratura. [11]

In fase di registrazione vengono memorizzati sul database tutti i dati dell'utente, compresa la password scelta dall'utente che viene cifrata per mezzo di un algoritmo unidirezionale, ovvero sarà possibile cifrare la password ma non decifrarla e sarà salvata sul database.

Al momento del login il sistema legge la password in chiaro inserita dall'utente, la cifra ottenendo un hash e confronta il contenuto dell'hash ottenuto con quello relativo alla password presente sul database dell'utente che tenta di autenticarsi. Se i due hash coincidono, l'utente è correttamente autenticato

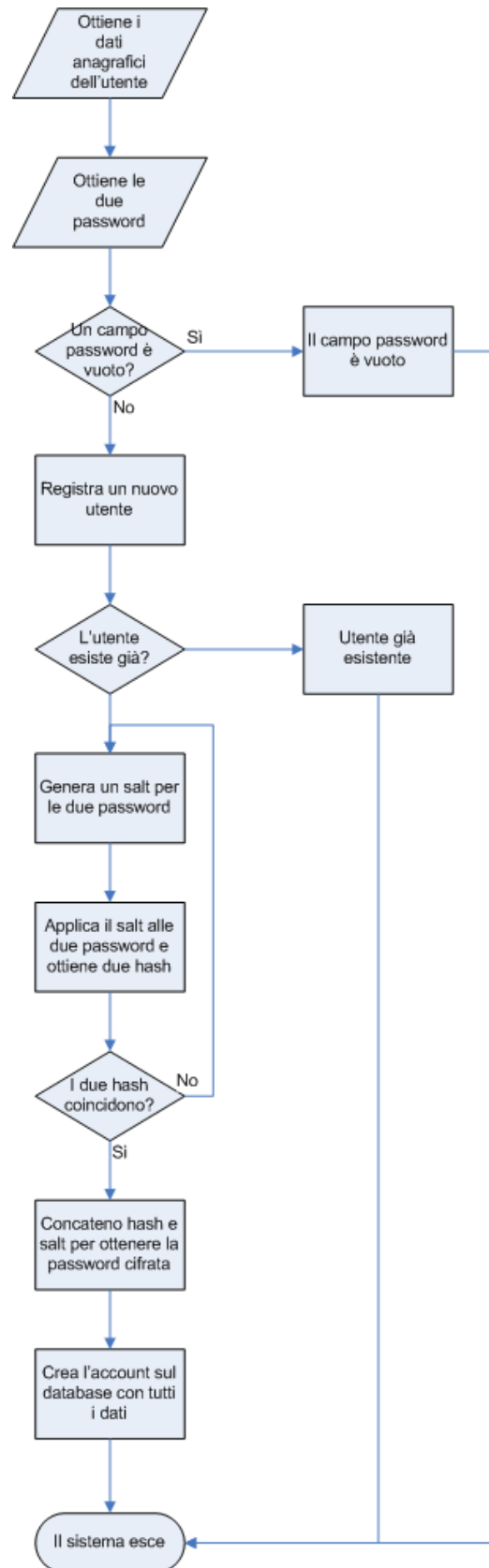


Figura 5.2: Diagramma della cifratura della password all'atto della registrazione di un utente

e può accedere al sistema.

Si prendono le due password inserite in chiaro e si genera un solo salt per tutte e due: un salt è un array di byte generato in modo casuale che occorre per criptare la password in chiaro e ottenere di conseguenza l'hash. Dalle due password in chiaro applicando il salt si ottengono due hash: se sono identici si concatena hash e salt per ottenere la password crittografata. Si crea l'account sul database con tutti i dati.

Viceversa, per controllare la correttezza di una password inserita, il sistema estrae il salt dalla password salvata sul database per cifrare la nuova password, genera un nuovo hash utilizzando il salt e controlla se l'hash ottenuto corrisponde all'hash estratto dal database.

Mediante questo sistema la password in chiaro non viene mai salvata, ma viene utilizzata solo in ingresso all'atto del login.

5.3 Il login dell'utente

Quando l'utente, effettuando il login, invia username e password, il sistema effettua i seguenti passaggi:

- cerca l'utente sul database attraverso lo username
- estrae il record contenente informazioni dell'account dell'utente ma utilizza per il momento solo la password cifrata
- estrae la chiave
- estrae l'hash
- prova a cifrare la password inserita dall'utente; ovvero genera un nuovo hash a partire dalla password in chiaro utilizzando il salt estratto in precedenza.
- controlla se il nuovo hash generato corrisponde a quello estratto dal database.

Se l'autenticazione ha avuto esito positivo si conosce il testo in chiaro della password inserita al momento della login. Il sistema si comporta di conseguenza:

- Genera un nuovo salt

- Prende la password che ha inserito l'utente (che è ovviamente quella giusta e che è quella in chiaro) e genera un nuovo hash utilizzando come chiave il nuovo salt generato
- Concatena hash e salt per ottenere una nuova password crittografata
- Aggiorna il campo sul database con questo valore

5.4 Il ruolo del docente

Come già notato nei diagrammi dei casi d'uso, il docente può effettuare tutte le operazioni di creazione, modifica e gestione dei propri esami, e in più può delegare un altro utente affinché possa gestire gli esami al suo posto.

Il docente deve essere in grado di conoscere con precisione gli utenti che sono stati delegati e modificarne la situazione attuale, in modo da permettere al proprio assistente, che potrebbe cambiare di anno in anno, di gestire i propri esami, o da toglierglieli in caso fosse necessario cambiare esaminatore o gestire i propri esami da sé.

5.4.1 La delega ad un altro utente

A differenza di un esaminatore, il docente, una volta autenticato, ha la possibilità di accedere ad una sezione chiamata *Delega utente* in cui sono evidenziati tutti gli esaminatori delegati. In questa sezione il docente vede la lista di tutti gli utenti registrati al sistema; in particolare sono evidenziati quelli a cui ha già concesso la delega e che quindi possono gestire i suoi esami. Il docente può aggiungere o togliere, selezionando le checkbox, utenti senza delega o utenti già delegati. Una volta confermata l'azione al sistema, viene aggiornata la situazione delle deleghe: il docente può quindi vedere la lista degli utenti che possono ora creare i propri esami. Questa operazione può essere effettuata in ogni momento e garantisce al docente il pieno controllo dei propri esami.

Nella delega ad un utente, il sistema cerca sul database l'identificativo della directory del docente, e mostra i nomi degli utenti che sono stati abilitati alla gestione degli esami; viceversa, gli utenti che senza delega sono quelli che non compaiono nella lista sopra descritta.

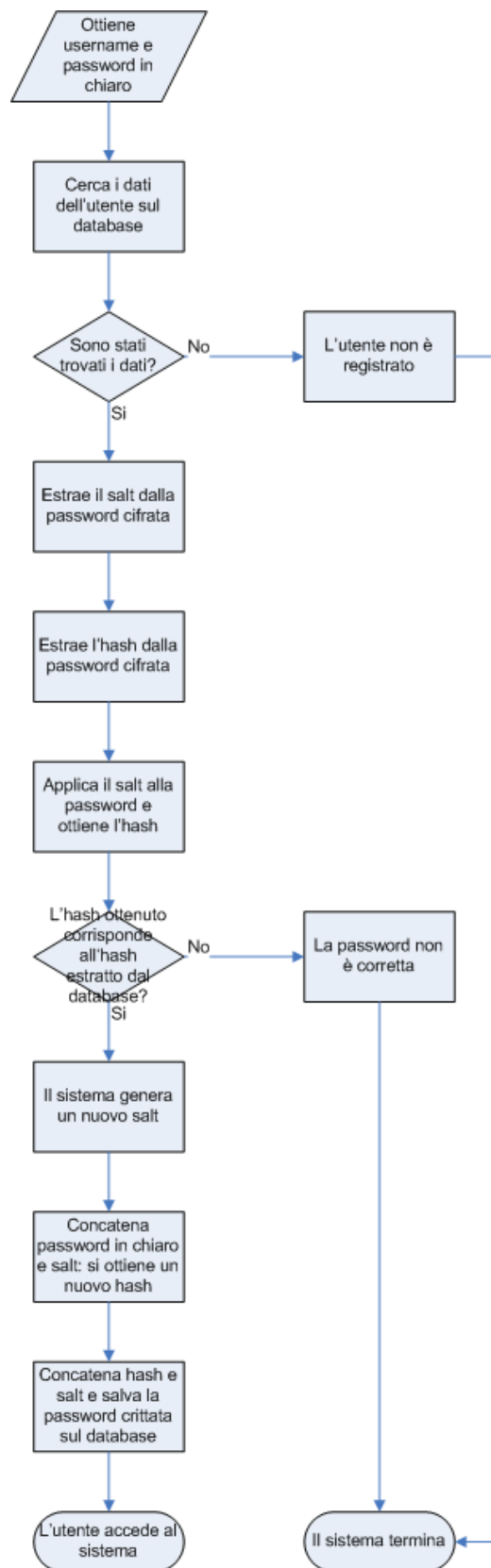


Figura 5.3: Diagramma del controllo della password al login dell'utente

Alla convalida della form da parte del docente, il sistema aggiunge o toglie determinati record nella tabella predisposta sulla base delle scelte fatte dal docente.

5.5 Creazione di un appello d'esame

Alla creazione di un nuovo appello d'esame vengono definiti turni, gruppi, e si indicano se sono previste domande a risposta multipla.

L'utente compila la form inserendo i valori dei campi obbligatori, ovvero il nome del docente proprietario dell'esame, che può essere lui stesso se l'utente è un docente, o può essere un altro docente nel caso di deleghe, la data dell'esame, gli indirizzi IP che possono accedere al sistema e il numero totale di turni. Il sistema non permette all'utente di creare esami la cui data appartenga al passato facendo notare l'errore e chiedendo di modificarla, e lancia un avviso nel caso in cui si voglia creare un esame per il giorno stesso; se l'utente conferma la volontà di creare un esame che si svolga in giornata, il sistema prosegue, altrimenti richiede di modificare la data della prova.

A questo punto il sistema mostra una nuova schermata, che dipende dal numero totale dei gruppi, che domanderà all'utente di definire per ogni turno previsto l'orario, il numero dei gruppi in cui suddividere il laboratorio, e la possibilità o meno di mostrare domande a risposta multipla agli studenti.

Gli indirizzi IP da segnalare corrispondono agli indirizzi IP privati delle sottoreti del laboratorio (nel caso in cui il computer sia nascosto da un proxy o da un router), oppure agli indirizzi IP pubblici nel caso di computer connessi direttamente alla rete, che si vuole possano accedere alla parte studente dell'applicativo durante il corso dell'esame. Il valore dei gruppi serve per determinare, una volta iniziato l'esame, a quale gruppo di esercizio appartiene un computer.

Una volta che l'utente convalida la form per la creazione dell'appello d'esame, vengono inseriti sul database vari record con i dati relativi ad ogni turno d'esame. Non viene salvato alcun altro dato relativo all'appello in sé perché tutte le informazioni necessarie sono già memorizzate all'interno dei record dei turni, e sono questi che verranno utilizzati dal docente per le modifiche, l'inserimento dei file necessari all'esame, l'inizio e il termine dell'esame in

Imposta nuovo esame:

Esame del prof.: (*) edenti

Data dell'appello: (*) 7 Ottobre 2005

Nome dell'esame: _____

Gruppo di studenti: _____ (es: A-K, Informatici, ecc.)

Orario dell'esame: _____

IP Allowed? LAB 2 (IP: 192.168.70.*)

LAB 3 (IP: 192.168.69.*)

altro IP: _____

Numero di turni: (*) 1 Inserire 1 se non presenti

I campi contrassegnati da (*) sono obbligatori

Figura 5.4: Form per la creazione di un appello d'esame

corso.

Per vincoli di chiave nella tabella del database non possono essere definiti più appelli d'esame appartenenti alla stessa data per lo stesso docente, ma è comunque possibile creare un appello d'esame contenente più turni. Non è inoltre possibile inserire un numero di turni totali o di gruppi nullo o negativo.

5.5.1 L'abilitazione di IP e sottoreti alla parte studente

Nella form utilizzata per la creazione di un appello d'esame sono presenti i campi degli indirizzi IP abilitati per l'accesso al sistema in fase d'esame. Vengono proposte le sottoreti del Laboratorio didattico di Informatica (Lab3) e del Laboratorio di Ingegneria Informatica (Lab2), essendo questi i luoghi solitamente predisposti allo svolgimento degli appelli d'esame, e inoltre è possibile inserire altri indirizzi IP per utilizzare il sistema con macchine esterne ai suddetti laboratori o appartenenti ad altre sottoreti. Tali sottoreti sono definite dalla sostituzione dell'ultimo numero della sequenza dell'indirizzo IP con un asterisco che indica l'intervallo di indirizzi ammessi e quindi appartenenti alla sottorete stessa. Questi indirizzi, inseriti in fase di creazione dell'esame, si riferiscono all'appello d'esame completo ma sono facilmente modificabili, eliminabili, o se ne possono aggiungere di nuovi tramite la pagina di modifica del turno d'esame.

All'atto della creazione di un nuovo appello d'esame, il sistema inserisce nella tabella *permessi_ip* tutti gli indirizzi IP che sono stati selezionati associati a ciascuno dei turni creati mediante l'identificativo del turno. Questa soluzione permette di personalizzare i punti d'accesso ai turni d'esame modificando degli indirizzi IP e di effettuare eventualmente da laboratori differenti prove simultanee, appartenenti a turni diversi.

5.5.2 La suddivisione in gruppi

La suddivisione in gruppi si base sull'indispensabile acquisizione dell'indirizzo IP locale della macchina da cui perviene l'accesso al sistema, poiché attraverso questi, e grazie ad una mappa dei laboratori della Facoltà, è stata definita la suddivisione in gruppi. È quindi di fondamentale importanza che il metodo di acquisizione degli indirizzi IP sia funzionante e non sia modificato da esterni.

La suddivisione in gruppi serve per poter presentare a studenti seduti su postazioni contigue esercizi diversi. La scelta del totale dei gruppi riguarda il singolo turno e non l'appello d'esame nel suo complesso; questa scelta agevola il docente nella personalizzazione dell'esame, perché può accadere che i primi turni di un appello siano quelli più pieni, e quindi necessitano una maggiore suddivisione in gruppi, mentre gli ultimi turni siano utilizzati da un numero minore di studenti, e di conseguenza il numero dei gruppi è minore.

Il numero dei gruppi è un dato appartenente al turno selezionato, esattamente come il flag per definire la possibilità di domande a risposta multipla, ed entrambi possono essere velocemente modificati.

Al momento la suddivisione dei gruppi è stata realizzata tenendo traccia dello schema degli indirizzi IP utilizzati per suddividere il laboratorio con l'applicativo preesistente. I raggruppamenti sono stati fatti in base ai totali dei gruppi predefiniti utilizzati dai docenti, ovvero nel caso di uno, due, quattro, o otto gruppi. Nel caso in cui l'utente scegliesse di suddividere il laboratorio utilizzando un valore differente di gruppi, o comunque non adeguatamente configurato, il sistema non saprebbe come effettuare la suddivisione e associerebbe tutte le macchine al gruppo 1; questa scelta deriva dal fatto che sia plausibile che esista sempre almeno un gruppo, ed è obbligata in quanto non esistono associazioni tra gli indirizzi IP delle macchine che accedono e le varie

suddivisioni in gruppi nel caso di valori non contemplati.

Sono inoltre state fatte le suddivisioni in gruppi solamente per il Lab2 e per il Lab3, in quanto sono i laboratori generalmente utilizzati per le prove d'esame; per gli altri indirizzi IP non esiste una numerazione predefinita, siccome non è possibile attuare particolari politiche di assegnamento dei gruppi in quanto l'utilizzo di IP diversi da quelli di Lab2 e Lab3 è considerato un evento eccezionale. In particolare, siccome la sottorete del Lab3 contiene macchine il cui indirizzo IP termina con numeri superiori a 100, nell'eventualità in cui gli indirizzi IP esterni terminassero allo stesso modo, seguirebbero lo stesso assegnamento dei gruppi del Lab3, mentre se il valore fosse minore, il loro gruppo di appartenenza sarebbe il n.1.

5.6 Stato di un turno d'esame

Lo stato è un dato appartenente ad ognuno dei record dei turni d'esame e rappresenta un particolare momento di vita del turno stesso. Questo stato, che definisce la creazione, l'inizio o il termine di una prova, viene aggiornato in base alle azioni eseguite su di esso.

La decisione di suddividere lo stato di un esame in questi tre particolari momenti si è rivelata molto opportuna dato che le differenti operazioni da svolgersi nel sistema possono essere effettuate solamente sui turni che appartengono ad un determinato stato: non deve essere infatti possibile modificare un turno già terminato, o fare l'appello degli studenti iscritti ad un turno in cui non è ancora stata inserita la lista d'esame, ecc.

Al momento della creazione, al nuovo appello d'esame viene assegnato lo stato iniziale: tutti i turni d'esame appartenenti a tale appello sono quindi definiti come "creati". Questo stato viene mantenuto per le modifiche e l'inserimento dei file inerenti al turno selezionato, fino alla decisione di "iniziare un esame". In questo caso infatti lo stato viene aggiornato e il turno passa a far parte degli esami "in corso": da questo punto in avanti non sarà più possibile effettuare modifiche di qualunque tipo alle impostazioni dell'esame, né sarà più consentito inviare i file contenenti la lista dell'esame, i testi degli esercizi o delle domande. Questo stato è necessario nel caso in cui si voglia fare l'appello degli studenti, il controllo del laboratorio, o le modifiche alle impostazioni

degli studenti.

Una volta terminato il turno dell'esame, lo stato viene ancora aggiornato e ottiene lo stato "terminato": non è più possibile effettuare interazioni con questo turno, se non per avere uno storico degli esami terminati e reperire l'archivio contenente gli elaborati prodotti dagli studenti durante tale turno.

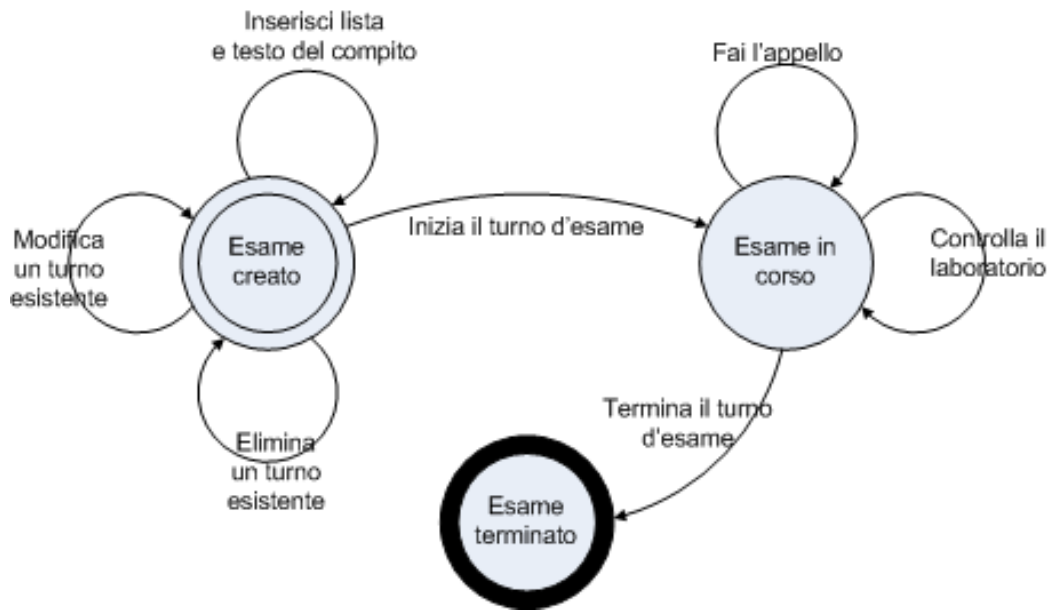


Figura 5.5: Diagramma degli stati di un turno d'esame

Lo stato di un turno d'esame viene utilizzato soprattutto nelle ricerche per "scremare" il risultato in base al target della ricerca stessa: la visualizzazione degli esami creati infatti non deve mostrare record di esami che sono stati definiti "in corso" o "terminati". Allo stesso modo, la history degli esami passati non produce riferimenti a esami che non sono ancora terminati, o ancora i controlli da fare sul laboratorio o sugli studenti che stanno effettuando un esame possono essere compiuti solamente su esami "in corso" in quanto gli esami "creati" producono un output nullo, e gli esami "terminati" mostrano un output non attuale e di scarso interesse per il docente.

5.7 La ricerca di un esame

La ricerca dei turni d'esame dipende dallo stato del turno e dalle deleghe di un utente. In particolare, un esaminatore a cui nessun docente ha conferito la delega non ha la possibilità di vedere alcun esame mentre un docente o un esaminatore che ha ricevuto almeno una delega può vedere tutti gli esami definiti da lui stesso o dal docente che rappresenta.

Nel caso di deleghe multiple, ovvero nel caso in cui un utente possa visualizzare gli esami di almeno due docenti, il risultato di una ricerca viene ordinato per data e per docente.

Qualunque operazione diversa dalla creazione di un nuovo appello d'esame effettuata da un utente richiede che sia stato trovato l'esame di interessa tramite la ricerca. Tale ricerca verte soprattutto sullo stato dell'esame e identifica in modo univoco un turno d'esame in modo che sia possibile effettuare azioni su di esso. Queste azioni possono essere di modifica delle impostazioni, di controllo, di inizio o termine dell'esame, o semplicemente di visualizzazione.

La ricerca individua, a seconda dello scopo da raggiungere, turni d'esame pertinenti. Nello specifico, la ricerca di un turno d'esame "creato" è interessante per le modifiche e le eliminazioni dei turni stessi, nonché per visualizzare un report degli esami futuri che sono già stati creati e per aggiungere i file relativi alla lista degli iscritti e agli esercizi e le domande multiple e per iniziare un esame.

Analogamente, la ricerca di un turno "in corso" è necessaria per tutte quelle operazioni di monitoring dell'esame, dall'appello degli studenti, al controllo del laboratorio e alla modifica delle informazioni di uno studente, intesi come gruppo di appartenenza e postazione di consegna nonché ovviamente per terminare un esame in corso.

La ricerca di turni "terminati" è un'azione che viene effettuata più raramente in quanto è utilizzata solo per visualizzare lo storico degli esami passati, e serve come statistica e reperimento degli archivi contenenti gli elaborati degli studenti durante l'esame selezionato.

All'atto della ricerca di un determinato tipo di esami, il sistema come prima

cosa filtra i dati presenti sul database in base deleghe dell'utente, e determina quindi tutti i turni d'esame associati, e al tempo stesso controlla che siano associati a un determinato stato.

5.8 Le modifiche possibili ai turni d'esame

La modifica di un esame avviene in seguito alla ricerca di un turno d'esame creato e non ancora iniziato, e permette all'utente di modificare tutti i campi che erano stati compilati in fase di creazione di quello specifico appello d'esame.

La modifica di dati semplici quali il numero dei gruppi, la possibilità di porre domande a risposta multipla, il nome della prova e l'orario (dati questi ultimi facoltativi) altro non fa che un aggiornamento del record del turno selezionato aggiornando i campi che sono stati modificati.

La modifica dei dati relativi agli indirizzi IP abilitati riguarda solamente il turno selezionato, in modo da permettere impostazioni differenziate per gli accessi per i diversi turni d'esame. Il sistema mostra gli indirizzi IP che sono stati associati al turno selezionato, e permette all'utente di aggiungere altri due campi in cui inserire nuovi valori da aggiungere a quelli preesistenti. La modifica degli indirizzi IP implica l'aggiornamento dei dati inseriti nell'apposita tabella di associazione tra l'esame e gli IP relativi.

La modifica della data del turno d'esame invece influisce su tutti gli altri turni appartenenti allo stesso appello d'esame, ovvero vengono cercati tutti i turni appartenenti a quel docente e a quella data, e viene a tutti quanti aggiornata la data d'esame.

Siccome nell'applicativo esiste il concetto di numero d'ordine dei turni, per ogni appello d'esame contenente N turni si prevedono N turni consecutivi numerati da 1 a N; pertanto non risulta possibile modificare il numero di un turno con un altro, perché questa modifica produrrebbe inconsistenze all'interno del sistema.

5.8.1 La modifica del numero totale dei turni

Nel caso in cui l'utente decida di modificare il numero dei turni totali dell'esame, il sistema si comporta in modo differente in base al nuovo valore inserito. Il sistema infatti, basandosi sul nuovo numero di turni, elimina o aggiunge

Modifica esame:

16/09/2005 - Prof. Enrico Denti - Turno 1 di 1

Modifica esame:

Data: 16 / 09 / 2005 (GG/MM/AAAA)

Nome:

Orario:

Gruppo:

Turno: 1 di 1

Gruppi:

Domande:

Ip Allowed:

Figura 5.6: Pagina di modifica di un turno d'esame

dei turni all'appello scelto.

Se il numero dei turni totali viene impostato a 0, allora il sistema cancella tutti i turni facenti parte dello stesso appello d'esame. Cancella quindi tutti i riferimenti relativi all'appello d'esame, eliminandolo completamente.

Se la modifica ai turni totali prevede che ci siano un numero maggiore di quelli preesistenti, il sistema aggiunge i turni mancanti inserendo le stesse impostazioni dell'ultimo turno trovato appartenente allo stesso appello d'esame. Questa soluzione è basata sul fatto che generalmente i turni iniziali sono quelli con maggiore affluenza e in cui ci sono più suddivisioni in gruppi, mentre i turni finali sono meno frequentati e richiedono un numero minore di gruppi di esercizi: per questo motivo un turno successivo avrà o le stesse impostazioni, o impostazioni meno stringenti.

Dal punto di vista delle richieste sul database, questo tipo di modifica implica una ricerca degli altri turni appartenenti allo stesso appello d'esame e un aggiornamento iniziale nei turni esistenti del nuovo numero di turni totali. In seguito verranno poi aggiunti gli altri turni mancanti come copia dell'ultimo

turno trovato appartenente allo stesso appello e con permessi per gli indirizzi IP uguali all'ultimo turno.

Se il numero di turni scelto è invece minore del numero di turni preesistente, il sistema trova tutti i turni appartenenti allo stesso appello d'esame, eliminando i record dei turni eccedenti, e aggiorna il numero di turni totali per i record dei turni rimanenti.

5.8.2 L'eliminazione di turni d'esame

La pagina di eliminazione di un turno d'esame permette all'utente di eliminare un determinato turno d'esame, effettuando eventuali compattamenti nel numero dei turni per quelli rimanenti appartenenti allo stesso appello d'esame.

Il caso più semplice è quello in cui venga eliminato l'unico turno di un appello contenente un solo turno: in questo caso viene eliminato il turno selezionato e gli indirizzi IP associati al turno stesso. Essendo l'unico turno dell'appello d'esame, l'appello stesso viene eliminato dal sistema.

Un altro caso semplice riguarda la richiesta di eliminazione dell'ultimo di una serie di turni di un appello d'esame: il sistema elimina il turno selezionato e decrementa il numero dei turni totali di tutti gli altri turni appartenenti allo stesso appello d'esame.

Nel caso in cui venga richiesta l'eliminazione di un turno diverso dall'ultimo per un appello d'esame avente più turni il sistema provvede a eliminare il turno selezionato e gli indirizzi IP associati, poi decrementa il numero d'ordine di tutti i turni il cui valore era superiore a quello del turno eliminato, in modo da eliminare le discontinuità, e aggiorna per tutti i turni il nuovo numero di turni totali, decrementandoli.

In seguito alla modifica o all'eliminazione di un turno d'esame il sistema mostra a video all'utente un report contenente tutti i turni d'esame marcati come "creati" di proprietà dell'utente assieme alle notizie importanti relative al turno, quali data, turno, turni totali, gruppi, possibilità di domande a risposta multipla.

5.9 L'inserimento della lista e dei testi delle prove

L'inserimento dei file è l'unica azione che preveda la scrittura su disco e la creazione delle directory appartenenti al turno, in quanto creazione, modifica ed eliminazione degli esami effettuano solamente inserimenti o aggiornamenti alle tabelle del database.

Le directory vengono create in dipendenza di alcuni parametri del turno selezionato. Vengono create all'interno della directory che porta lo username del docente titolare dell'esame, e il loro nome è della forma di *AAAAMMGG[/-TN]* in cui la prima parte del nome della directory è la data dell'esame a partire da anno, mese giorno, mentre la seconda parte esiste solamente nel caso in cui ci sia più di un turno; il suffisso serve a discriminare il turno *N*. Al momento della creazione della directory, viene creata una ulteriore directory *conf*, figlia della directory dell'esame, in cui vengono salvati i file inseriti e il file dei log che verrà sviluppato durante lo svolgimento dell'esame.

In questa sezione viene richiesto all'utente di inserire una serie di file di configurazione che hanno nome invariabile e indipendente dall'esame, e altri file che dipendono dal numero dei gruppi utilizzati.

lista.txt è uno di quei file che devono essere inseriti indipendentemente dalla configurazione specifica della prova d'esame. È il file CSV scaricato dal sistema di Ateneo Uniwex, a cui viene cambiata l'estensione. Questa impostazione non è stata modificata rispetto alle richieste del sistema preesistente.

infoese.html è il file contenente alcune informazioni sull'esercizio. Nel sistema preesistente aveva estensione **.htm**.

infocon.html è il file contenente le informazioni per la consegna degli elaborati. Nel sistema preesistente aveva estensione **.htm**.

infodom.html se esiste, è il file che contiene le informazioni per le domande a risposta multipla. Nel sistema preesistente aveva estensione **.htm**.

E' necessario che questi file si chiamino esattamente come richiesto, comprese le estensioni complete, e il rispetto di maiuscole e minuscole.

Selezione esame:

16/09/2005 - Prof. Enrico Denti - Turno 1 di 1

E' fondamentale che i file si chiamino **esattamente** come descritto (lettere maiuscole e minuscole comprese).

Inserire qui i file:

- lista.txt
- infocon.html contenente le informazioni per gli studenti sulla consegna (per esempio quali files devono essere consegnati)
- infoese.html contenente le informazioni per gli studenti sull'esame (per esempio il tempo a disposizione)
- esercizio1.html e relativi file (immagini, ecc) se previsti

Figura 5.7: Pagina di inserimento dei file per un turno d'esame

Gli altri file che il docente deve inserire sono quelli degli esercizi e alle domande per ogni gruppo, e devono chiamarsi *esercizioN.html* e *domandeN.xml* in cui N è il valore del gruppo. Questo vincolo sui nomi dei file sono stati ripresi dall'applicativo precedente e si è ritenuto di mantenerli per compatibilità all'indietro, per abitudini dei docenti, per un possibile riutilizzo di file, etc.

Nella pagina dell'inserimento dei file è inserita una form esattamente come quella che usa lo studente per l'inserimento in remoto dei propri elaborati, e utilizza, esattamente come la parte studente, il package del FileUpload [7]. Come visto precedentemente questo tipo di form necessita di una struttura "ibrida" per l'inserimento di ulteriori parametri provenienti da campi di tipo hidden allo scopo di inviare correttamente i dati contenuti nel file .

Per ogni invio, il sistema cerca nella directory dell'esame la presenza del file corretto e mostra un messaggio nel caso in cui il file sia già stato inserito; in caso contrario il file deve essere ancora inviato.

Per rimediare a possibili dimenticanze degli utenti nel nominare le pagine web il sistema controlla l'esistenza del file chiamato correttamente e, in caso

lo trovi, cerca il corrispettivo file con estensione `.htm`. Se neanche in questo caso il sistema trova il file, utilizza alcuni file di default che sono stati salvati sul server apposta per adempiere a queste mancanze. Questa soluzione è stata ideata per consentire a quei docenti che preferiscono consegnare il testo dell'esercizio in cartaceo, o utilizzare altri sistemi per mostrare il testo del compito di non ricevere eccezioni causate dall'assenza del file cercato dal sistema.

Al fine di mantenere le pagine dell'esercizio ben formate, è necessario che il codice dei file di configurazione e dell'esercizio siano scritti in html corretto ma senza alcun tag di `<head>`, né il tag `<body>`. Questa richiesta deriva dal fatto i documenti contenenti le informazioni e gli esercizi vengono inclusi all'interno di un'altra pagina HTML, avente già di per sé quei due tag: ripetizioni dei tag `<head>` e `<body>` possono creare inconsistenze nella visualizzazione delle pagine.

5.10 Controlli durante l'esame

I controlli effettuabili durante la prova d'esame sono la vera parte innovativa di tutto il progetto, in quanto non esistono nel vecchio applicativo e sono stati progettati in base alle richieste dei docenti.

5.10.1 L'inizio del turno esame

Nella pagina dell'inizio della prova l'utente può solamente cercare il turno d'esame che vuole che inizi e confermare la scelta.

Il turno d'esame aggiornerà il proprio stato e la prova sarà a tutti gli effetti "in corso": non sarà più possibile effettuare modifiche o inserimento di file, e allo stesso tempo nella pagina iniziale dell'applicativo parte studente verrà aggiunta l'opzione per effettuare l'esame appena iniziato.

5.10.2 Gestione della fase di entrata degli studenti

L'appello degli studenti all'ingresso del laboratorio è il primo tra i controlli che il docente può effettuare per limitare gli accessi indebiti al sistema.

Tramite questo servizio, il docente potrà vedere a video la lista degli studenti iscritti all'esame, in base al file `lista.txt` inserito. Per ogni studente viene mostrata a video una riga recante gli stessi dati del file della lista, ovvero nome,

cognome, numero di matricola e il progressivo dell'iscrizione dello studente nella lista di Uniwex; due radiobutton permettono di decidere sulla presenza o assenza dello studente all'esame. Questa funzionalità si basa sulla presenza della stringa di comodo "CANC" all'interno del file di lista che, se esiste, definisce l'assenza dello studente alla prova. Il docente può impostare tale parametro in modo automatico semplicemente definendo selezionando assenti gli studenti che non si presentano alla prova.

Una volta accettate le modifiche, il sistema riscrive il file di lista aggiungendo la stringa "CANC" per tutti quegli studenti che sono stati definiti assenti; l'applicativo lato studente impedirà a tutti gli studenti assenti di effettuare l'esame.

Nella pagina dell'appello degli studenti sono disponibili alcune righe vuote in cui è possibile inserire i dati di nuovi studenti non iscritti in lista che si vuole possano effettuare l'esame. Queste righe sono inizialmente disabilitate: l'utente deve impostare il valore del radiobutton a "presente" e può così compilare i campi richiesti per permettere allo studente di affrontare l'esame. Una volta accettate le modifiche, il file *lista.txt* viene riscritto inserendo non solo le modifiche sulla presenza degli studenti che già iscritti, ma anche appendendo in fondo al file i dati dei nuovi studenti iscritti.

5.10.3 Controllo della situazione in laboratorio

Il controllo del laboratorio permette al docente di conoscere la macchina da cui uno studente ha fatto login, i dati dello studente letti dal file della lista d'esame e i file che ha consegnato.

La pagina di controllo si ricarica periodicamente e legge il contenuto delle directory interne al turno d'esame selezionato per mostrare al docente i file inviati, escludendo i file di flag creati in seguito al login ed eventualmente all'invio delle risposte per le domande a risposta multipla.

Il controllo dell'esame mostra graficamente al docente la situazione del Lab3 tramite l'utilizzo della mappa con il posizionamento dei PC, in modo che il docente possa avere la sensazione di trovarsi in laboratorio e possa vedere in modo grafico lo sviluppo dell'esame.

In questa pagina il docente può solamente controllare lo sviluppo della prova

da parte degli studenti e non può effettuare delle modifiche sulle impostazioni degli studenti. Il sistema mostra comunque a video al docente se da una stessa postazione sono stati effettuati i login con due differenti numeri di matricola.

5.10.4 Modifica delle impostazioni di uno studente

Questa sezione permette al docente di modificare alcune delle impostazioni dello studente. Se uno studente, dopo essere acceduto dalla propria postazione, venisse spostato, non potrebbe più accedere al sistema da una postazione diversa, dato che per motivi di sicurezza non è possibile autenticarsi due volte da PC diversi. Utilizzando questo sistema il docente è libero di spostare a suo piacimento uno studente, riabilitandogli l'accesso eliminando il file di flag che determina la postazione da cui è stato fatto il primo login: l'effetto che se ne ottiene da questa azione equivale ad un nuovo "primo login", anche se eventuali file già consegnati non sono toccati dal sistema. Questa operazione però, oltre alla possibilità di un nuovo accesso da una postazione differente, porta all'assegnazione di un nuovo gruppo d'esame basato sulla posizione del PC da cui si accede ora; in questo caso il docente può decidere di modificare il gruppo di appartenenza dello studente e permettergli di proseguire nello svolgimento della prova che aveva iniziato.

È anche possibile per il docente cambiare allo studente il gruppo di appartenenza, inserendone uno nuovo in qualunque momento. Il sistema in questo caso modifica il file di flag riscrivendolo con un nuovo valore del gruppo di appartenenza. Non verrà data comunicazione allo studente da parte dell'applicativo, benché nei dati di intestazione sia aggiornato il gruppo di appartenenza, e il testo dell'esercizio e le domande siano ora differenti da quelli mostrati in precedenza. Neanche in questo caso vengono toccati i file che lo studente può avere già consegnato.

La modifica delle impostazioni dello studente sarà utile quando verrà riabilitato l'unico accesso al programma; ovvero nel caso in cui lo studente, una volta chiuso il browser, tenti un ulteriore login, non riuscirebbe più ad accedere, e dovrebbe chiedere l'intervento del docente che resettì il file di flag e gli permetta un ulteriore login.

Si noti che in ognuno di questi casi, benché venga modificato o cancellato il file di flag, ogni azione è scrupolosamente salvata su file di log ed è quindi possibile controllare a posteriori eventuali accessi indebiti o situazioni poco chiare.

5.11 La chiusura del turno d'esame

Al termine di un esame le operazioni effettuate dal sistema sono quelle del cambiamento dello stato del turno d'esame selezionato, della ridenominazione del file di lista come *lista_old.txt* e della creazione di un nuovo file di lista vuoto, e della creazione di un archivio zip contenente tutta la directory dell'esame compressa.

Il cambiamento dello stato implica per il docente l'impossibilità di effettuare qualunque altra azione di modifica o controllo sull'esame.

Il cambiamento di stato dell'esame impedisce inoltre allo studente di effettuare ulteriori login alla prova, in quanto non è possibile visualizzare e scegliere di effettuare un esame il cui stato risulta "terminato". Siccome può capitare che lo studente selezioni l'esame appena prima della chiusura dell'esame, la modifica di *lista.txt* impedisce ulteriori login degli studenti.

Se lo studente è già autenticato e all'interno del sistema, non è possibile dare avviso del fatto che altri file inseriti sul server non sarebbero presi in considerazione, né è possibile impedire ulteriori invii da parte degli studenti. Il modo più immediato per conoscere la situazione del laboratorio al termine del tempo prestabilito per l'esame è quello di fare una "fotografia" del sistema, e degli elaborati in esso consegnati creando un archivio ZIP messo a disposizione dell'esaminatore e del docente perché possa essere scaricato e corretto.

5.11.1 Cronologia degli esami terminati

Nel caso in cui l'utente non avesse il tempo di scaricare l'archivio contenente gli elaborati, la pagina della history degli esami terminati permette al docente e all'esaminatore di vedere tutti gli esami che sono stati terminati, e mette a disposizione il link per scaricare nuovamente l'archivio ZIP.

Capitolo 6

Collaudo dell'applicativo

6.1 Annessione dell'applicativo parte studente alla parte docente

Il passo finale dell'implementazione progetto riguarda il necessario collegamento tra le due parti del progetto, che fino ad ora sono state mantenute separate. Il sistema deve essere in grado di rispondere correttamente alle azioni di creazione e inizio dell'esame definite da docenti o esaminatori, mostrando nell'applicativo parte studente le voci corrette dei turni d'esame e permettendo l'accesso degli studenti solo agli esami realmente definiti come "in corso". Il sistema deve ora quindi riconoscere l'esame che si desidera sostenere e conseguentemente reperire il file con la lista degli studenti iscritti per permettere loro l'accesso alla determinata prova, riconoscendo l'albero di directory all'interno del quale deve creare le directory, il file di flag per il login e salvare gli elaborati degli studenti.

Per combinare le funzionalità del tool di amministrazione per il docente affinché possa interfacciarsi con il progetto della parte studente è stato definito un albero di directory all'interno del quale saranno salvati gli esami, i file di configurazione degli stessi e gli elaborati degli studenti. A partire dalla directory padre chiamata *studenti* interna alla web application vengono create le directory dei docenti, chiamate con lo stesso nome dello username del docente; all'interno di queste sono create le directory degli esami all'atto del primo invio dei file per l'esame.

La directory dell'esame è chiamata in base alla data dell'esame e ha un suf-

fisso di tipo *_TN* nel caso in cui l'appello d'esame contenga più prove e *N* definisce il valore del turno. Questa directory contiene un'ulteriore directory *conf* al cui interno si trovano la lista degli iscritti all'esame, i testi dei compiti e delle domande, eventualmente le pagine HTML con le avvertenze per la consegna, l'esercizio e le domande; all'interno di questa directory vengono successivamente salvati il file di log dell'esame e l'archivio zip che il docente può scaricare una volta terminata la prova per la correzione degli elaborati. La directory dell'esame contiene inoltre tutte le directory che l'applicativo parte studente creerà per ogni studente e dentro le quali verranno salvati gli elaborati inviati dagli studenti al server.

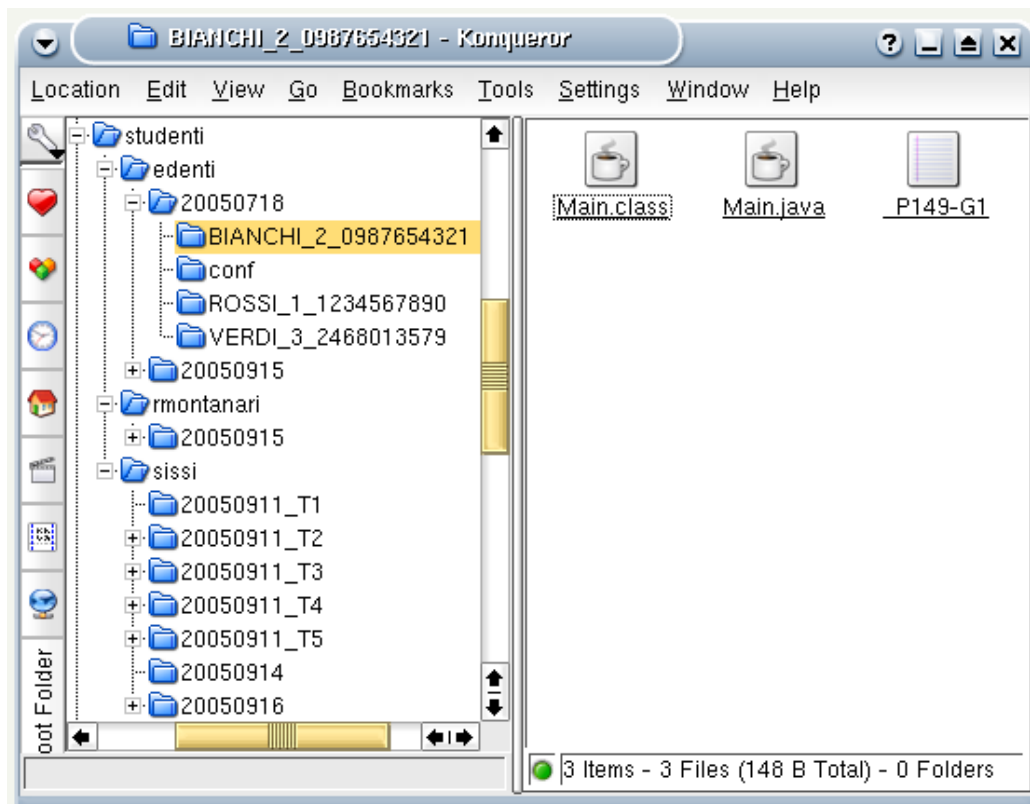


Figura 6.1: Schema dell'albero di directory che contengono i file degli esami

6.1.1 Interfacciamento della parte studente

Una volta configurato l'esame, l'applicativo parte studente deve reperire con i dati della prova. Per questo motivo è stata creata la pagina antecedente all'inizio della prova vera e propria, che, tramite una query sul database, individua gli esami "in corso" e mostra allo studente tanti link quanti sono questi esami per scegliere la prova da affrontare. Questi link portano tutti alla stessa pagina di login; la differenza risiede però in due parametri hidden che riportano l'identificativo dell'esame e il percorso in cui il sistema deve creare la directory dello studente. In seguito al login dello studente il sistema crea dei cookie aggiuntivi a quelli progettati precedentemente contenenti informazioni sull'esame e sulla sua locazione.

6.2 Collaudo del progetto completo

Esattamente come è stato fatto per il collaudo della prima parte dell'applicativo, le sessioni di prova non si sono limitate alla verifica della funzionalità del sito, ma hanno anche incluso tutte quelle attività necessarie a dimostrarne la navigabilità, l'accessibilità, la compatibilità coi vari browser e le varie versioni degli stessi ed in generale il rispetto delle regole di usability.

6.2.1 Controllo dei contenuti

Il controllo finale dei contenuti è stato effettuato dal docente che ha messo in luce alcune ambiguità sull'utilizzo del termine "esame" che spesso è stato utilizzato quale sinonimo di "appello d'esame" mentre altre volte come sinonimo di "turno appartenente ad un appello d'esame": sono stati modificati alcuni dei contenuti della parte docente per ovviare a queste ambiguità.

Sono inoltre stati specificate alcune richieste, ovvero nel caso dell'inserimento dei file dell'esame è stata chiarita meglio la necessità da parte di un docente di inserire i file con lo stesso nome che viene richiesto, rispettando in particolare lettere maiuscole, minuscole e l'estensione dei file.

6.2.2 Collaudo di funzionalità

Si è cercato di uniformare il codice HTML affinché soddisfi i requisiti dello standard del World Wide Web Consortium [9], in particolare è stato validato con successo il foglio di stile CSS, e si è cercato di aderire il più possibile allo standard di XHTML per le pagine JSP.

6.2.3 Collaudo con diversi browser e sistemi operativi

Il sistema è stato collaudato effettuando i test su sistemi Linux (Gentoo e RedHat) e Windows (Windows XP e Windows NT) cercando di simulare al meglio le macchine del laboratorio, utilizzando i browser più probabili per i due sistemi: Internet Explorer 5.5, Internet Explorer 6.0, Mozilla Firefox 1.0.2, Mozilla 1.7, Netscape Navigator 7.0.

6.2.4 Controllo della visibilità

È stato verificato che la veste grafica del sistema non subisce grandi cambiamenti tra l'utilizzo di un browser ed un altro. Non sono presenti bruschi cambiamenti di interfaccia grafica tra una pagina e l'altra benché il layout sia leggermente cambiato tra le pagine appartenenti alla parte studente e quelle del tool di amministrazione del docente, dato che viene comunque utilizzato lo stesso file CSS per tutte le pagine.

6.3 Pre-test dell'applicativo

L'applicativo nella sua veste definitiva è stato collaudato una prima volta in Lab3 durante l'appello d'esame autunnale del corso di Fondamenti di Informatica L-B del professor Denti.

Il tool di amministrazione del docente ha funzionato senza problemi, mentre si sono verificate alcune eccezioni nell'applicazione lato studente.

La stessa eccezione si è verificata in seguito alla corretta autenticazione dello studente al sistema: dopo aver premuto il pulsante per accedere al testo del compito un'eccezione avvertiva dell'impossibilità di salvare il cognome dello studente, in quanto la stringa richiesta attraverso i cookie risultava nulla.

Sapendo che se un cookie viene creato inserendo una stringa contenente spazi,

questo crea una stringa nel cookie compresa tra virgolette e quindi nel reperimento dei file potrebbero verificarsi delle inconsistenze, si è pensato che il problema provenisse dalla presenza di studenti con nomi cognomi doppi (ad esempio invece di essere MARIO#DE ROSSI venivano salvato come "MARIO#DE ROSSI"). Questo bug era stato notato in precedenza, in concomitanza di test intermedi al tool per il docente, ed era già stato sistemato intercettando in tutti i campi di nome e cognome dello studente gli spazi e i caratteri speciali come l'apostrofo sostituendoli con "_".

L'eccezione è tra l'altro capitata a studenti con nomi e cognomi senza un comune denominatore, ovvero gli studenti a cui si è presentato il problema hanno nomi con e senza spazi: la possibilità di un errore derivante da questo genere di situazione è quindi da escludere. Uno tra questi studenti ha inoltre affermato di essere riuscito ad entrare nel sistema ritentando il login in un secondo momento senza avere problemi. È stato quindi supposto che il server non abbia serializzato le richieste fatte dai client, che potrebbero aver tentato l'accesso contemporaneamente, ma il log del sistema smentisce anche questa ipotesi. In ogni caso il login e la successiva creazione dei cookie dello studente sono andati a buon fine, dato che in caso contrario gli studenti non sarebbero neanche potuti accedere, senza contare che i file di flag di questi studenti sono stati correttamente creati e contengono i dati corretti degli studenti.

Per sistemare questo bug il sistema ora controlla che i cookie siano stati creati correttamente prima di caricare la pagina e, nel caso si verificano degli errori sul contenuto dei cookie, il sistema provvede a crearne e inviarne dei nuovi basati sulle informazioni scritte sul file di flag: in questo modo lo studente è in grado di accedere senza problemi al testo del compito d'esame.

6.4 Test finale dell'applicativo

Il test finale è stato effettuato in Lab3 simulando le stesse caratteristiche del test precedente. La parte di tool di amministrazione ha funzionato egregiamente, e l'applicativo lato studente non ha dato ora alcun problema: i cookie sono stati creati correttamente, gli utenti sono potuti accedere correttamente alla pagina dell'esercizio e di consegna degli elaborati, caricando sul server file di diverse dimensioni, provando anche a scaricarli e ad eliminarne qualcuno o tutti.

Il sistema ha risposto egregiamente anche tentando login multipli da postazioni diverse.

Conclusioni

È stato realizzato un servizio che fornisce una valida alternativa al sistema di gestione di esami preesistente. Il sistema è facilmente estendibile: in futuro si potrebbero aggiungere ulteriori funzioni al sistema, quali ad esempio la possibilità per lo studente di rispondere a domande a risposta aperta, oppure la gestione di file XML per la lettura della lista d'esame, o la scrittura dei file di flag del login e delle risposte alle domande.

Con lo sviluppo del sistema di monitoring e gestione per il docente si è voluto sopperire ad una mancanza che i docenti sentivano da tempo e si è cercato di automatizzare alcune operazioni e permettere al docente ed esaminatore di controllare l'esame tramite un'interfaccia web.

Ci sono alcuni aspetti che dovrebbero essere ancora migliorati nel componente: in particolare la grafica del tool, benché semplice e funzionale, può apparire un po' spartana. Un altro aspetto che si potrebbe implementare potrebbe essere l'inserimento di una guida interattiva che accompagni il docente durante il percorso che lo porti ad effettuare e terminare un esame.

Al sistema di amministrazione sono state inoltre aggiunte la pagina, non contemplata all'inizio del progetto, per eseguire l'appello degli studenti all'entrata in laboratorio, e la funzione che crea l'archivio ZIP degli elaborati degli studenti.

Altri miglioramenti potrebbero vertere sulla registrazione di docenti ed esaminatori. Sebbene al momento chiunque abbia la possibilità di iscriversi in qualità di docente o esaminatore, anche se non quella di vedere e modificare gli esami degli altri utenti, sarebbe più sicuro offrire un'autenticazione tramite il backend LDAP, che permetterebbe di associare alla figura del docente

o dell'esaminatore, una persona appartenente allo staff ufficiale interno della Facoltà, eventualmente mantenendo sul database traccia di quali utenti possano realmente autenticarsi in quanto docenti o assistenti, e quali invece non possano farlo.

È stato inoltre ritenuto utile, poter inviare via email al docente titolare dell'esame e all'esaminatore che ha iniziato e monitorato la prova d'esame l'archivio ZIP contenente gli elaborati degli studenti. Gli amministratori di sistema della Facoltà hanno confermato che questa aggiunta è fattibile in quanto esiste già un server SMTP che, sebbene non accetti il relay, permette di inviare email provenienti dall'interno della Facoltà.

Appendice A

Manuale docente

Per utilizzare il sistema di creazione e gestione degli esami è necessario registrarsi compilando la form di iscrizione che si trova cliccando sul link “Registrati” visibile dalla pagina di login del docente. La scelta in questa form di nominarsi docente o esaminatore porta alla creazione di utenti con permessi differenti: per un Docente viene creata una directory al cui interno il sistema inserisce in seguito le sottosezioni per le configurazioni degli esami e gli elaborati degli studenti, mentre un Esaminatore non ha una directory riservata, e non può compiere azioni se non in seguito alla delega di un docente. Un docente può delegare un altro utente andando sulla sezione “Delega altro utente” e selezionando l’utente che si vuole possa accedere alle impostazioni per i propri esami: solo in questo modo quindi un esaminatore può creare e gestire degli esami.

A.1 Creazione dell’appello d’esame

E’ possibile creare nuovi appelli d’esame, possibilmente contenenti più turni d’esame.

Per ogni esame è possibile impostare data, numero di turni, laboratorio in cui effettuare l’esame e, per ogni turno è possibile scegliere il numero di gruppi in cui suddividere gli studenti, ed è possibile indicare se saranno poste domande a risposta multipla.

A.1.1 Modifiche ai turni d'esame

Una volta creato un appello d'esame è possibile modificare le impostazioni per ognuno di questi.

In particolare modificando la data dell'esame, il sistema sposta la data di tutti i turni d'esame appartenenti all'appello d'esame selezionato; modificando i gruppi, le sottoreti abilitate per accedere al sistema, la possibilità di inserire domande a risposta multipla, il sistema aggiorna i dati del turno selezionato. Nel caso di modifiche al numero di turni, il sistema aggiunge i turni mancanti inserendo le stesse impostazioni dell'ultimo turno trovato appartenente allo stesso appello d'esame: questa soluzione è basata sul fatto che solitamente i turni iniziali sono quelli con maggiore affluenza e in cui ci sono più suddivisioni in gruppi, mentre i turni finali sono meno frequentati e richiedono un numero minore di gruppi di esercizi; per questo motivo un turno successivo avrà o le stesse impostazioni, o impostazioni meno stringenti.

A.1.2 L'eliminazione di un turno d'esame

Nel caso in cui venga richiesta l'eliminazione di un turno d'esame, il sistema provvede a eliminarlo assieme agli indirizzi IP associati, decrementando eventualmente il numero d'ordine di tutti i turni di valore superiore a quello del turno eliminato, in modo da eliminare le discontinuità, e aggiorna per tutti i turni rimanenti appartenenti all'appello d'esame il nuovo numero di turni totali.

In seguito alla modifica o all'eliminazione di un turno d'esame il sistema mostra a video un report contenente tutti i turni d'esame creati e non ancora iniziati di proprietà dell'utente assieme ai dati importanti relativi al turno, quali data, turno, turni totali, gruppi, possibilità di domande a risposta multipla.

A.1.3 Inserimento dei file dell'esame

L'inserimento dei file prevede la scrittura su disco e la creazione delle directory appartenenti al turno, il cui nome dipende da alcuni dei parametri del turno selezionato. La directory viene creata all'interno della directory del docente titolare dell'esame, e il suo nome è della forma di *AAAAMMGG[_TN]*

in cui la prima parte è la data dell'esame a partire da anno, mese giorno, mentre la seconda parte esiste solamente nel caso in cui ci sia più di un turno: il suffisso discrimina il turno N .

Il sistema chiede di inserire una serie di file di configurazione che devono chiamarsi sempre allo stesso modo, e altri file che dipendono dal numero dei gruppi utilizzati.

lista.txt è il file CSV scaricato dal sistema di Ateneo Uniwex, cambiato di estensione.

infoese.html contiene alcune informazioni sull'esercizio.

infocon.html contiene le informazione per la consegna degli elaborati

infodom.html se esiste, contiene le informazioni per le domande a risposta multipla.

E' necessario che questi file si chiamino esattamente come richiesto, comprese le estensioni complete, e il rispetto di maiuscole e minuscole.

Gli altri file da inserire sono quelli degli esercizi e alle domande per ogni gruppo, e devono chiamarsi *esercizioN.html* e *DomandeN.xml* in cui N è il valore del gruppo.

Affinché le pagine mostrate nell'applicativo lato studente siano ben formate, è necessario che i file *infoese.html*, *infodom.html*, *infocon.html* e i testi degli esercizi contenuti nei file *esercizioN.html* non contengano al loro interno i tag `<head>` e `<body>` in quanto queste pagine vengono incluse all'interno di altre che contengono già questi tag, e la presenza di tag doppi può portare ad inconsistenze nella visualizzazione della pagina.

Il file contenente le domande a deve seguire lo schema proposto in figura A.1.

A.2 Inizio dell'esame

Affinché un turno esame sia visibile e selezionabile dagli studenti, è necessario confermare l'inizio della prova cercando il turno d'esame e confermando la form per l'inizio dell'esame.

```
<Domande>
  <Domanda>
    <Question> Testo della domanda </Question>
    <Response> Risposta n.1 </Response>
    <Response> Risposta n.2 </Response>
  </Domanda>
  <Domanda>...</Domanda>
</Domande>
```

Figura A.1: Schema del file *DomandeN.xml*

A.2.1 Appello degli studenti

Tramite questo servizio, è possibile vedere la lista degli studenti iscritti all'esame, in base al file *lista.txt* inserito. Per ogni studente viene mostrata una riga recante gli stessi dati del file della lista; due radiobutton permettono di decidere sulla presenza o assenza dello studente all'esame. Gli studenti definiti come "assenti" non potranno accedere all'esame.

Sono inoltre disponibili al termine della lista degli iscritti alcune righe vuote in cui è possibile inserire i dati di studenti non iscritti e che si vuole possano effettuare l'esame. Queste righe sono inizialmente disabilitate: è necessario impostare il valore del radiobutton a "presente" e può così compilare i campi richiesti per permettere allo studente di affrontare l'esame. Una volta accettate le modifiche, lo studente risulta correttamente iscritto e può effettuare l'esame.

A.2.2 Controllo del laboratorio

Il controllo del laboratorio permette al docente di conoscere la macchina da cui uno studente ha fatto login, i dati dello studente letti dal file della lista d'esame e i file che ha consegnato.

La pagina di controllo si ricarica periodicamente e legge il contenuto delle directory interne al turno d'esame selezionato per mostrare al docente i file inviati, escludendo i file di flag creati in seguito al login ed eventualmente all'invio delle risposte per le domande a risposta multipla.

A.2.3 Modifica dei dati degli studenti

Utilizzando questo sistema è possibile spostare uno studente, riabilitandogli conseguentemente l'accesso: l'effetto ottenuto da questa operazione è quello di ad un nuovo "primo login", benché eventuali file già consegnati non vengano toccati dal sistema.

È anche possibile cambiare in qualsiasi momento il gruppo di appartenenza dello studente. Lo studente non darà comunicazione allo studente, benché nei dati di intestazione della pagina web dello studente sia aggiornato il gruppo di appartenenza, e il testo dell'esercizio e le domande siano differenti.

A.3 Termine dell'esame

Il termine di un turno d'esame implica l'impossibilità di effettuare qualunque altra azione di modifica o di controllo sull'esame.

Questa funzione impedisce inoltre allo studente di effettuare ulteriori login, in quanto non è possibile visualizzare un esame "terminato". Se lo studente però si è già autenticato, non è possibile avvisarlo che ulteriori file inseriti non saranno presi in considerazione. Al termine dell'esame infatti, non essendo possibile staccare le connessioni degli studenti, viene creato un archivio ZIP che mostra una "fotografia" della situazione del laboratorio alla fine di tale esame. Questo archivio può essere scaricato quindi in seguito al termine dell'esame, oppure mediante la visualizzazione della history dei turni d'esame terminati.

A.4 Principio di funzionamento

L'applicazione parte dalla pagina di scelta dell'esame: lo studente definisce l'esame che vuole effettuare e può quindi digitare il proprio numero di matricola. Alla convalida della form, il sistema controlla sul file *lista.txt* se la matricola è contenuta nell'elenco degli iscritti: se la matricola non è nell'elenco il sistema ritorna alla pagina di login; se la matricola è valida il sistema crea, se non esiste già, una directory figlia di quella dell'esame il cui nome segue la struttura *Cognome_IdIscrizione_Matricola*, in cui *IdIscrizione* è il numero di iscrizione alla lista di Uniwex.

In base all'indirizzo IP del PC su cui lo studente si trova e alla mappa dei

gruppi memorizzata dal sistema, viene determinato il gruppo di appartenenza, e di conseguenza l'esercizio e le domande.

Nella directory dello studente viene controllata l'esistenza un file con il nome *_Pxx-Gyy* in cui *xx* sono le ultime cifre dell'indirizzo IP e *yy* è il gruppo di appartenenza. Se tale file non esiste, si tratta del primo login, e il sistema crea questo file inserendo all'interno una serie di informazioni, quali il nome, il cognome e numero di matricola dello studente, l'indirizzo IP completo del PC da cui si è acceduto e il timestamp del primo login; se tale file esiste invece significa che viene effettuato un ulteriore login e il sistema acconsente al nuovo accesso al sistema, se invece esiste un altro file di configurazione con parametri differenti, significa che qualcuno, da una postazione differente dal PC dello studente sta tentando l'accesso e il sistema impedisce tale accesso.

La pagina dell'esercizio contiene solamente il testo dell'esercizio e non permette interazioni. La pagina per le domande a risposta multipla contiene una form costruita automaticamente sulla base del file *domandeN.xml*, in cui *N* è il numero del gruppo e mostra le domande e le possibili risposte. Premendo il pulsante di conferma, le risposte alle domande vengono salvate nel file *_domandeN.txt*, contenuto nella directory dello studente. Questo file contiene le associazioni tra il numero della domanda e la relativa risposta. Il pulsante "Azzera" annulla la consegna delle domande cancellando il suddetto file.

La pagina per la consegna consente di inviare nella directory personale dello studente i file relativi all'esercizio. L'elenco dei file consegnati è visualizzato nella pagina ed è possibile anche cancellare prelevare o cancellare un file.

Bibliografia

- [1] Cascading Style Sheets
<http://www.w3.org/Style/CSS/>

- [2] Wikipedia.it - Secure Sockets Layer
<http://it.wikipedia.org/wiki/SSL>

- [3] Apache Foundation Php: Hypertext Preprocessor
www.php.net

- [4] Sun MicroSystem
java.sun.com

- [5] Sun MicroSystem - JavaServer Pages Technology
java.sun.com/products/jsp/

- [6] Ip Grabber
<http://javascript.internet.com/user-details/ip.html>

- [7] Commons FileUpload
<http://jakarta.apache.org/commons/fileupload/index.html>

- [8] Nebel et al. (1995) *RFC 1867 - Form-based File Upload in HTML*
<http://www.ietf.org/rfc/rfc1867.txt>

- [9] World Wide Web Consortium
www.w3.org

- [10] PostgreSQL
<http://www.postgresql.org>

- [11] Rivest et al. (1992) *RFC 1321 - The MD5 Message-Digest Algorithm*
<http://www.ietf.org/rfc/rfc1321.txt>