

POLITECNICO DI MILANO

Facoltà di Ingegneria dell'Informazione
Dipartimento di Elettronica e Informazione



**MINERVA: ANALISI E PROGETTAZIONE
DEL MUSEO VIRTUALE DELL'ISOLA COMACINA**

Relatore: Ing. Francesco AMIGONI
Correlatori: Prof. Stefano DELLA TORRE
Dott.ssa Viola SCHIAFFONATI

Tesina di Laurea di:
Riccardo ANDRETTA matr. 644748
Gabriele ERBA matr. 642030

Anno Accademico 2004-05

A Chiara

Sommario

Il lavoro della presente tesina si inserisce nell'ambito del Progetto Minerva del Politecnico di Milano, mirato allo sviluppo di un software per la creazione guidata di un museo virtuale ad hoc per l'utente. Motivo del lavoro qui svolto è stata la necessità di fornire al Comune di Ossuccio un applicativo per il museo che verrà ivi realizzato. Attraverso l'ausilio del software l'utente potrà creare un museo virtuale in base a propri interessi e curiosità e potrà fruire di un'ampia quantità di reperti rinvenuti presso l'isola Comacina sul Lario. Questi sarebbero altrimenti destinati a restare in gran parte sconosciuti per mancanza di spazio espositivo.

Lo scopo di questa tesina è l'analisi dei requisiti del museo virtuale dell'isola Comacina e la progettazione di un software in grado di soddisfare le richieste della committenza. A tal fine si sono tenute alcune riunioni con il rappresentante della committenza, Prof. Della Torre, docente del Politecnico di Milano, nel corso delle quali si sono definite le problematiche e discusse le strade percorribili, e numerosi incontri con il nostro relatore Ing. Francesco Amigoni, il correlatore Dott.ssa Viola Schiaffonati e l'Arch. Andrea Bonavita allo scopo di approfondire nel dettaglio quanto emerso dai colloqui con Della Torre per individuare le soluzioni appropriate. L'attività di implementazione è stata svolta parallelamente da Calcaterra e Franzosi nel loro lavoro di tesina, grazie anche al nostro aiuto.

Il software così prodotto è stato sottoposto a Della Torre il quale ha confermato il pieno soddisfacimento dei requisiti. Margini di sviluppo restano in particolare per la perfetta gestione dell'accesso concorrente, caratteristica necessaria per l'accesso al sistema via web, e per il raffinamento dell'interfaccia utente eventualmente con elementi di tridimensionalità.

Ringraziamenti

Ringrazio i miei genitori e mia sorella, per il supporto datomi in questi anni

Ringrazio tutti i miei compagni di studio, per aver vissuto una “vita” insieme al Poli, in particolare chi ha condiviso con me lo sviluppo del progetto Minerva ovvero, in rigoroso ordine alfabetico, Gianluigi, Massimo e Riccardo

Indice

Titolo.....	1
Sommario.....	3
Ringraziamenti.....	5
Indice.....	7
1 Introduzione.....	10
2 Stato dell'arte.....	16
2.1 Minerva05.....	16
2.2 Musei virtuali con interfaccia web.....	19
2.2.1 Catalogo.....	20
2.2.2 Tour virtuale.....	23
2.2.3 Museo virtuale.....	26
2.3 Musei virtuali creati dall'utente.....	27
3 Analisi dei requisiti.....	29
3.1 Studio del museo di Ossuccio.....	29
3.2 Ricerca e scelta delle tecnologie appropriate.....	31
3.3 Reingegnerizzazione del software Minerva.....	36
4 Progetto logico e soluzione del problema.....	41
4.1 Definizione di item, temi e criteri.....	41
4.2 Definizione di stanze.....	44
4.3 Interazione dell'utente e navigazione all'interno del museo.....	45
4.4 Pannello di amministrazione.....	46
4.5 Soluzioni individuate nella fase di testing.....	47
4.5.1 Definizione di teche.....	48
4.5.2 Definizione di risorse esterne.....	49
4.5.3 Raffinamento dell'interfaccia utente.....	50
4.5.4 Criteri di raggruppamento.....	54
5 Architettura.....	55
5.1 Architettura generale del sistema.....	55
5.2 Obiettivi tesina Calcaterra-Franzosi.....	59

6 Valutazione.....	62
6.1 Test Bonavita.....	63
6.2 Test Della Torre.....	64
6.3 Prove di funzionamento.....	67
7 Sviluppi futuri e conclusioni.....	73
7.1 Ambiente 3D di navigazione.....	73
7.2 Allestitore.....	75
7.3 Allocazione di stanze e teche.....	76
7.4 JSP, Servlet e modello MVC.....	78
7.4.1 Il modello Model-View-Controller.....	78
7.5 Accesso concorrente via web.....	80
7.6 Sistema centralizzato e distribuito.....	80
7.7 Generalizzazione del progetto.....	82
7.8 Compatibilità delle linee di sviluppo.....	83
7.9 Conclusioni.....	84
Bibliografia.....	85

Capitolo 1

Introduzione

Il Progetto Minerva si inserisce nell'ambito degli strumenti informatici messi a disposizione del settore museale, con particolare riferimento ad un genere di prodotto definito museo virtuale, ossia sistemi software in grado di riprodurre su computer un'esposizione di oggetti collocati in un ambiente puramente immaginario oppure che rispecchi le caratteristiche di un luogo realmente esistente. Grazie allo sviluppo di Internet ed alla ormai ampia diffusione di tecnologie di collegamento a banda larga, i musei virtuali trovano nella rete globale il mezzo principe per la propria presentazione al visitatore, avendo a disposizione un enorme bacino d'utenza e potendo contare su standard di visualizzazione dei contenuti via web sempre più elevati.

La versione di Minerva illustrata in questa tesina rappresenta un'evoluzione ed un riadattamento di quanto realizzato a partire dal 1995 all'interno del Politecnico di Milano, anno in cui è stata concepita l'idea di un progetto sul museo virtuale. Occasione di ulteriore sviluppo e modifica rispetto a quanto fatto finora è stata la necessità di fornire al Comune di Ossuccio un applicativo che desse all'utente la facoltà di creare un museo virtuale secondo i propri gusti artistici ed interessi storici. Tale strumento permette la fruizione di un'ampia quantità di reperti rinvenuti presso l'isola Comacina sul Lario, che non avrebbero trovato spazio sufficiente nel futuro museo archeologico di Ossuccio per l'esposizione al pubblico e che, quindi, sarebbero stati destinati a rimanere in gran parte irrilevanti.

Lo scopo di questa tesina è descrivere l'individuazione dei requisiti specifici della richiesta della committenza e la loro trasposizione in un progetto sufficientemente dettagliato e pronto per l'implementazione, operazione quest'ultima

illustrata nella tesina dei nostri colleghi Calcaterra e Franzosi. A tal fine si sono tenuti alcuni incontri con il Prof. Della Torre del Politecnico di Milano, in veste di rappresentante della

committenza, nel corso dei quali si sono discussi gli obiettivi da perseguire analizzando insieme le richieste e valutando le soluzioni che il nostro software avrebbe potuto offrire. Nei periodi tra gli incontri con Della Torre hanno avuto luogo numerose riunioni con il nostro relatore Ing. Francesco Amigoni ed il correlatore Dott.ssa Viola Schiaffonati per approfondire i temi trattati, riunioni a cui in seguito ha partecipato, da circa metà della fase di sviluppo, l'Arch. Andrea Bonavita, collaboratore di Della Torre, in funzione di indispensabile supporto per i lati più squisitamente contenutistici.

Parallelamente all'analisi dei requisiti ed alla progettazione, è stato portato avanti lo sviluppo di un prototipo secondo il modello evolutivo a spirale del ciclo di vita del software, permettendo in questo modo di ripercorrere le singole fasi nell'arco dell'intero processo. Grazie ai frequenti incontri con relatore e correlatore, con Bonavita e, più sporadicamente, con Della Torre si sono potute delineare in maniera abbastanza dettagliata le esigenze della committenza e giungere infine ad un prodotto valutato positivamente dallo stesso Della Torre.

In tema di musei virtuali quanto viene tipicamente offerto su Internet è un approccio unicamente consultativo all'esposizione, in cui l'utente non ha alcun ruolo nella creazione vera e propria della collezione di opere, ma si configura solamente come visitatore di un'esposizione prodotta secondo i canoni artistici di qualcun'altro. Molto più rari sono invece i casi in cui all'utente è offerta la possibilità di creare una propria collezione in base ad interessi personali.

In questo ambito, ma con una consistente innovatività, si inserisce il lavoro di

tesi svolto da Fabio Ghidotti [1] denominato Progetto Minerva05. Si tratta di un software che si rivolge esclusivamente ai curatori dei musei offrendo loro un supporto nella creazione di allestimenti museali nel settore archeologico, in grado di produrre come risultato un museo virtuale tridimensionale liberamente navigabile dall'utente. Questo

progetto ha rappresentato il punto di partenza per la presente versione di Minerva, per la quale sono state apportate sostanziose modifiche, anche strutturali, e introdotti vari aspetti innovativi per la realizzazione di un prodotto che fosse in linea con le esigenze specifiche del Comune di Ossuccio.

È stata inizialmente svolta un'attenta fase di analisi dei requisiti, concretizzata nei primi incontri con Della Torre nel corso dei quali ci sono state comunicate le idee preliminari su quello che doveva venirsi a configurare come il museo virtuale dell'isola Comacina. In base a quanto appreso in tali riunioni si è proceduto ad un lavoro di affinamento delle specifiche, con la collaborazione di Amigoni e Schiaffonati, che ha permesso di definire con maggiore chiarezza alcuni aspetti degli obiettivi da perseguire. È stata al contempo intrapresa un'analisi del software Minerva05, per comprenderne l'architettura e i principi di funzionamento e capire come si sarebbe dovuto procedere per adattarlo alle esigenze del committente. Com'era nel lavoro di Ghidotti, è stato mantenuto cardine del nuovo progetto l'ambiente ad agenti software JADE [9]. Gli agenti però hanno subito un generale processo di reingegnerizzazione determinato dai loro mutati scopi, dal ridimensionamento delle funzioni ad essi richieste e da una necessità di ottimizzazione generale delle prestazioni e dell'utilizzo delle risorse hardware dovuta alla volontà di creare un prodotto destinato al pubblico e che, pertanto, deve presentare tempi di attesa molto contenuti. Inoltre la struttura ha visto l'introduzione

di un nuovo componente, il server web Jakarta Tomcat [10], destinato a soddisfare le richieste degli utenti interagenti tramite un comune Internet browser, e ad interfacciarsi con JADE grazie alla creazione al suo interno di un agente client, creato ex-novo a tal fine. Altrettanto importante è da considerarsi la migrazione dal database Access [11] a MySql [12], più adatto ad essere utilizzato in ambito web grazie ad una solida gestione della concorrenza degli accessi, una riconosciuta affidabilità e robustezza, nonché la possibilità d'impiego di una licenza Open Source [13] anche per i software a fini commerciali.

Redatto un progetto di massima e svolte queste attività di reingegnerizzazione si è passati all'implementazione di un primo prototipo, che è stato posto alla valutazione di Della Torre per la verifica dell'attività fino a quel punto svolta e della compatibilità delle soluzioni individuate con i requisiti precedentemente indicati. Confermata la correttezza della strada intrapresa e stabiliti con maggior rigore gli scopi del progetto, sono continuati gli incontri periodici, con frequenza circa settimanale, con Amigoni e Schiaffonati e con una nuova figura rappresentata dall'Arch. Andrea Bonavita che ha fornito il materiale per il popolamento del database e ha dato preziosi consigli sull'interfaccia per un design funzionale ed un prodotto facile da utilizzare. Va a questo punto sottolineato che non rientrava nei nostri compiti occuparci in maniera estensiva del lato grafico del progetto, aspetto che rientra nell'ambito degli sviluppi futuri. L'aspetto maggiormente enfatizzato dalla committenza era che il prodotto offrisse al visitatore la possibilità di creare un proprio percorso di visita, secondo interessi personali, e che questo si svolgesse attraverso un modello ciclico in cui partendo da un allestimento o da un singolo reperto fosse possibile creare una nuova collezione secondo diversi criteri e parametri di selezione, il tutto senza soluzione di continuità. Inoltre punti di contatto tra i vari reperti

dovevano essere dei temi, intesi come tematiche culturali condivise da diverse opere, che avrebbero permesso all'utente di approfondire una serie di argomenti correlati agli oggetti in un'ottica decisamente innovativa rispetto a quanto finora offerto dai musei virtuali. Di tutti questi requisiti si è tenuto conto nella formulazione del progetto e nella trasformazione del primo prototipo nel prodotto finale che è stato presentato a Della Torre nell'ultimo incontro svoltosi in novembre, durante il quale è stato positivamente valutato il lavoro compiuto confermando il raggiungimento degli obiettivi prefissati.

Nel corso d'ulteriori sviluppi del lavoro qui presentato andrà approfondita la tematica della concorrenza degli accessi, in particolare per la tracciabilità delle singole richieste nella comunicazione tra server web e server Minerva (quest'ultimo collocato sotto forma di agente in ambito JADE) e per permettere una migliore scalabilità del sistema nel suo complesso. Interessanti prospettive sono infine lasciate aperte in riferimento alla veste grafica (ad esempio con l'implementazione di una visita tridimensionale) ed a fasi di allestimento e allocazione che tengano conto di parametri fisici reali nel caso si voglia impiegare il software in ambienti virtuali che riproducano ambienti reali.

Nel Capitolo 2 viene descritto lo stato dell'arte a partire da un approfondimento di Minerva05 per mostrare in seguito esempi di musei virtuali con interfaccia web e realizzabili dall'utente.

Nel Capitolo 3 si tratta l'analisi dei requisiti del museo virtuale dell'isola Comacina, la scelta delle tecnologie appropriate e la reingegnerizzazione di Minerva05.

Nel Capitolo 4 si illustra il progetto logico e le soluzioni individuate ai

problemi del Capitolo 3. Queste si riferiscono sia ad una fase iniziale del progetto, sia a momenti successivi durante lo sviluppo del prototipo.

Nel Capitolo 5 viene descritta l'architettura generale del sistema e sono brevemente presentati gli obiettivi ed i risultati del lavoro svolto da Calcaterra e Franzosi nella loro tesina.

Nel Capitolo 6 vengono trattati i processi di valutazione ad opera di Della Torre e Bonavita, e si mostrano alcune prove di esecuzione del sistema implementato.

Nel Capitolo 7 sono illustrati i possibili sviluppi futuri e le conclusioni del lavoro svolto in questa tesina.

Capitolo 2

Stato dell'arte

“Factum abiit, monumenta manent”

Ovidio

Questo capitolo ha lo scopo di illustrare quelle che sono le tecnologie oggi disponibili nell'ambito della fruizione di contenuti museali attraverso mezzi informatici. Questa non pretende certo di essere una rassegna esaustiva sull'argomento, che richiederebbe uno spazio eccessivo per le dimensioni della presente tesina, ma vuole piuttosto proporsi come una panoramica sulle principali implementazioni di tecnologie informatiche oggi messe a disposizione da parte dei musei.

Questo progetto nasce dal software *Minerva05* prodotto da Fabio Ghidotti [1]: andremo pertanto ad analizzare le principali caratteristiche di tale versione di Minerva. Inoltre i prodotti qui presentati vengono suddivisi nelle due categorie comprendenti i software indirizzati unicamente alla fruizione dei contenuti espositivi sotto forma digitale con interfaccia web, e quelli invece che permettono all'utente un'interazione più coinvolgente dando la possibilità di creare il museo virtuale in base a decisioni proprie.

2.1 Minerva05

L'impiego di tecnologie informatiche in ambito museale è stato finora per lo più accolto con un certo scetticismo da parte degli addetti del settore. I motivi di questo atteggiamento vanno probabilmente ricercati nella scarsa propensione

all'innovazione tecnologica di tali persone, da una parte, e, dall'altra, nelle limitate funzionalità dei prodotti proposti dagli sviluppatori di applicazioni.

Il Progetto Minerva del Politecnico di Milano si inserisce in questo ambito come un progetto innovativo rispetto a quanto finora reso disponibile, fornendo un sistema in grado di sfruttare tecniche dell'intelligenza artificiale per la realizzazione automatica di allestimenti museali in ambienti virtuali.

Le prime versioni del sistema non si sono avvalse di un supporto continuo da parte di esperti del settore museale, ma sono invece state guidate da scopi accademici, trascurando quelle che potevano rappresentare concrete esigenze e problematiche reali tipiche di questa attività. Il progetto originario, nato nel 1995 da un'idea del Prof. Marco Somalvico, è cresciuto sotto la cura dei suoi sviluppatori non essendo a contatto continuo con esperti del settore museale. In tal modo, si è inconsapevolmente allontanato dalle aspettative degli esperti del settore, non riuscendo a produrre risultati soddisfacenti nella sua missione di allestimento automatizzato, pur presentando aspetti molto interessanti sotto il profilo delle tecniche e delle tecnologie impiegate, e quindi delle potenzialità.

Tutto ciò è servito da insegnamento al lavoro svolto da Ghidotti, il quale si è avvalso del supporto di personale esperto del settore allo scopo di realizzare uno strumento finalmente efficace nel suo intento di realizzazione automatizzata di allestimenti.

Il progetto Minerva05 è stato infatti concepito con l'obiettivo di superare uno dei maggiori vincoli della prima versione del software Minerva (d'ora in poi *Minerva98*), ossia la scarsa configurabilità del software per quanto riguardava la fase di allestimento. Minerva98 inglobava principi di allestimento automatizzato non

configurabili, assumendo di poter impiegare regole generali e universali nella delicata operazione di disposizione degli oggetti all'interno dello spazio virtuale. Questo rappresentava un ulteriore motivo di titubanza verso l'impiego del software da parte degli operatori del settore, in quanto, una volta scelto il set di opere da esporre, impediva l'applicazione di criteri personali, frutto della propria esperienza maturata negli anni e del proprio gusto artistico ed anzi rendendo totalmente oscura l'operazione, cioè fornendo unicamente il risultato ad operazione conclusa. Minerva05 si inserisce a questo punto con una versatilità del tutto nuova: il software propone un'interattività continua con l'operatore, al quale viene fornita ampia scelta di regole e criteri di allestimento. Il processo perciò non si configura più come totalmente automatizzato ma anzi si avvale del fondamentale know-how e gusto artistico dell'utente.

Un'ulteriore novità, non meno importante, è stata l'introduzione del paradigma ad agenti in sostituzione di quello ad oggetti. Per quanto riguarda l'implementazione vera e propria del software, che ha permesso uno sviluppo più omogeneo del progetto nel suo insieme e un approccio più consono al modo di operare di Minerva, apportando inoltre indiscutibili vantaggi per quanto riguarda la scalabilità, la flessibilità e la possibilità di ulteriore sviluppo del progetto. Un agente software [33] è un'entità autonoma, in grado di agire in collaborazione con altri agenti o con esseri umani e percependo informazioni dall'ambiente, in modo tale da operare secondo la propria base di conoscenze per raggiungere i propri obiettivi. Gli agenti coinvolti si occupano di una vasta serie di aspetti durante l'utilizzo del software, andando da funzioni di supporto per la creazione delle esposizioni e per la gestione del sistema in rete, alle fasi direttamente concernenti l'allocazione e l'allestimento automatici. Questi agenti operano, come già detto, in modo autonomo ma collaborando

attivamente fra loro, mediante uno scambio di messaggi strutturati secondo lo standard FIPA [14]. Questa prima implementazione del paradigma ad agenti ha permesso di raggiungere risultati allestitivi senz'altro soddisfacenti, mostrando come siano stati ottenuti anche un buon livello di robustezza e stabilità del sistema, pur soffrendo d'altro canto di prestazioni non sempre eccellenti e di una certa lentezza nei tempi di esecuzione, aspetti però questi ultimi realisticamente migliorabili in sviluppi futuri del sistema.

Infine la presentazione dei risultati dell'operazione di allestimento si avvale di un'accattivante nonché funzionale veste tridimensionale, sia per quanto riguarda l'ambiente espositivo, sia per le opere, realizzati con modelli a tre dimensioni fedeli alla realtà.

2.2 Musei virtuali con interfaccia web

Il concetto di museo virtuale va via via sempre maggiormente diffondendosi nella rete globale, divenendo uno strumento ormai abbastanza diffuso all'interno dei siti web dei musei. Questi siti affiancano ad una tradizionale funzione informativa riguardo al museo stesso (costi, orari, informazioni sulle mostre, ecc.) una sezione dedicata all'esplorazione del museo e delle opere esposte che permette al visitatore virtuale di prendere contatto con le opere stesse, visualizzandone immagini e schede informative più o meno dettagliate. Tipicamente viene offerta una "visita virtuale del museo" che, sotto vesti grafiche variabili dalla pura catalogazione delle opere alla riproduzione fedele dell'ambiente museale, propone un accesso al museo che può considerarsi complementare alla visita reale, volto ad aumentare la visibilità del museo e l'interesse del pubblico.

I fattori che differenziano l'esperienza vissuta dal visitatore virtuale possono essere essenzialmente raccolti in due categorie: il grado di interattività offerta e la veste grafica proposta. Nei prossimi paragrafi sono brevemente esposti i diversi approcci adottati nel fornire l'“esperienza virtuale” distinti in base ai valori di questi due parametri ed ordinati secondo l'importanza crescente attribuita ad essi, che possono essere raggruppati nelle seguenti categorie:

- Catalogo,
- Tour virtuale,
- Museo virtuale,
- Museo virtuale creato dall'utente.

2.2.1 Catalogo

Un approccio classico fornito ai visitatori dei siti web di molti dei musei più importanti del mondo è quello rappresentato da una catalogazione piuttosto semplice delle opere esposte tramite schede contenenti immagini ed informazioni su ciascuna opera. Vediamo alcuni esempi di questo tipo di soluzione.

Sul sito del Guggenheim Museum [2] tramite il link *Collection Online* si accede ad una sezione in cui è possibile consultare le opere esposte secondo un ordinamento alfabetico in base a vari criteri: artista, movimento, titolo, mezzo, data, concetto. Nella parte sinistra dello schermo sono raggruppati i vari ordinamenti possibili ed un motore di ricerca, mentre nella parte centrale si accede ai risultati dell'ordinamento o della ricerca come mostrato in Figura 2.1. Successivamente, selezionando uno dei link proposti, si accede alla lista delle opere relative, mostrate ciascuna con una piccola foto, titolo e data (Figura 2.2). Ciascuna foto funziona da collegamento alla pagina dedicata all'opera.

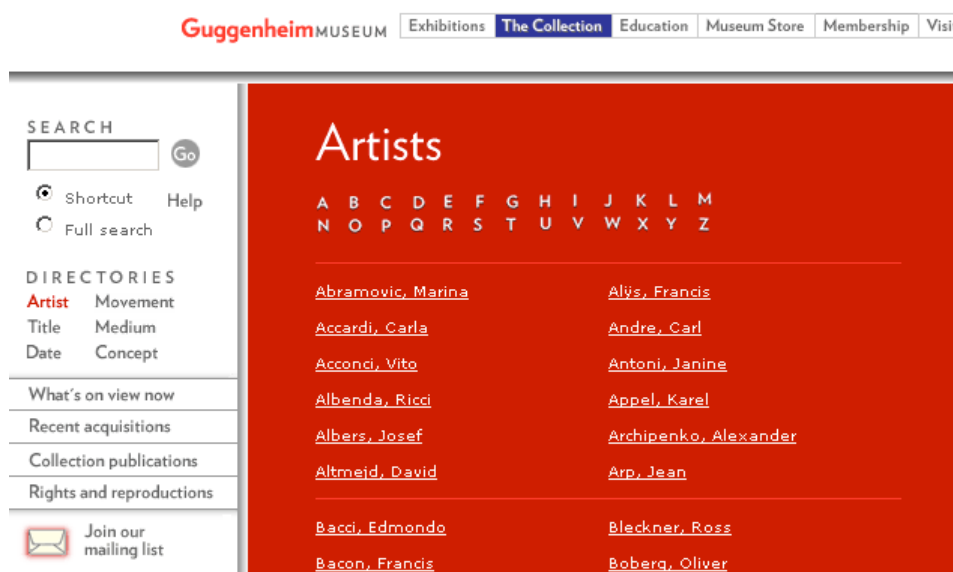


Figura 2.1: pagina di accesso ai vari ordinamenti delle opere del sito del Guggenheim Museum

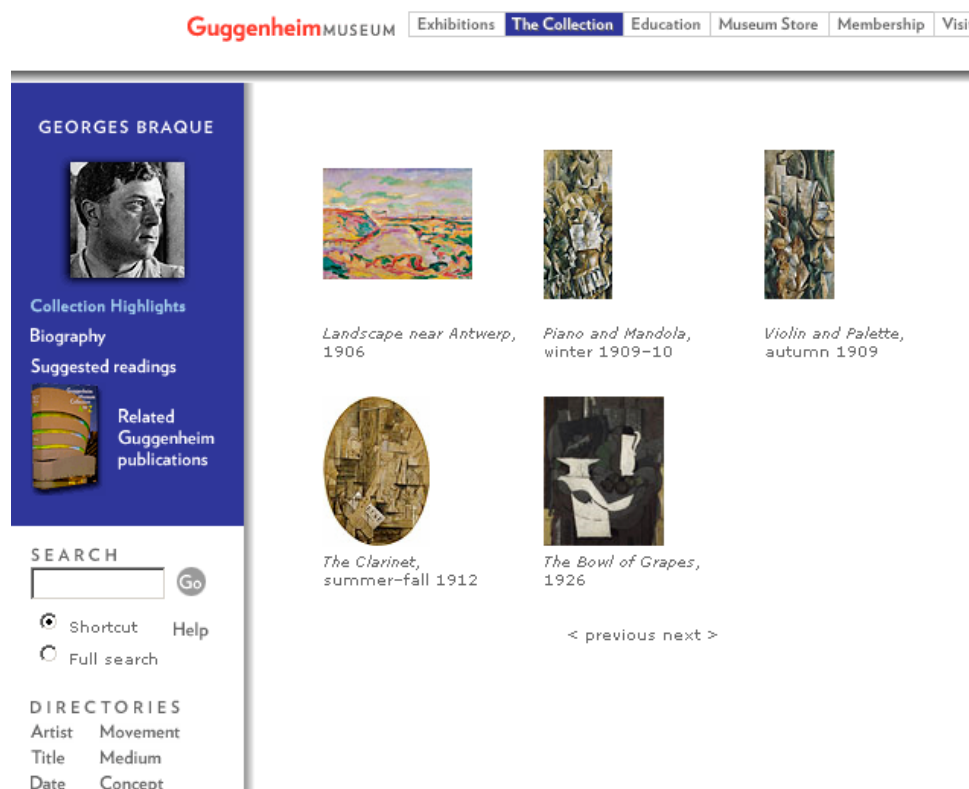


Figura 2.2: pagina dedicata alle opere di un artista; le immagini in piccolo sono link ai dettagli dell'opera

Un altro esempio di catalogazione delle opere ci viene fornito dal sito della National Gallery [3]. In questo caso cliccando prima su *Collection* quindi su *Explore the collection online* si accede alla pagina mostrata in Figura 2.3 in cui gli

ordinamenti proposti sono meno schematici rispetto a quanto visto nel sito del Guggenheim Museum, ma rimane sostanzialmente inalterato il concetto di catalogazione delle opere. In Figura 2.4 è mostrata la pagina dedicata alle biografie degli autori delle opere più importanti esposte dal museo.

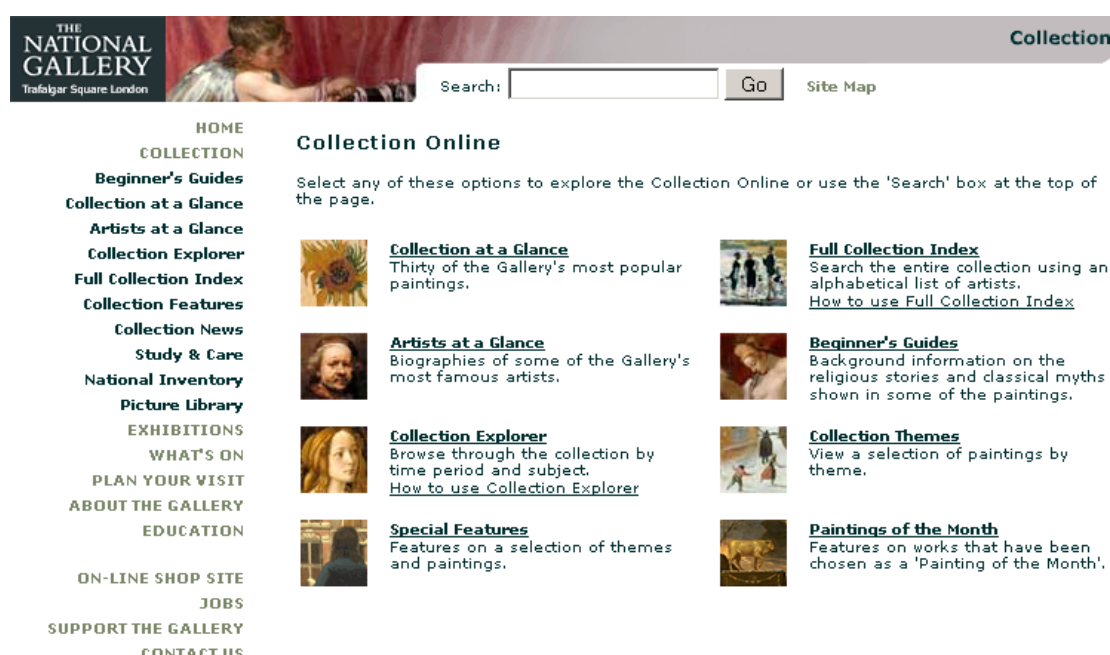


Figura 2.3: pagina di accesso alla consultazione online delle opere della National Gallery

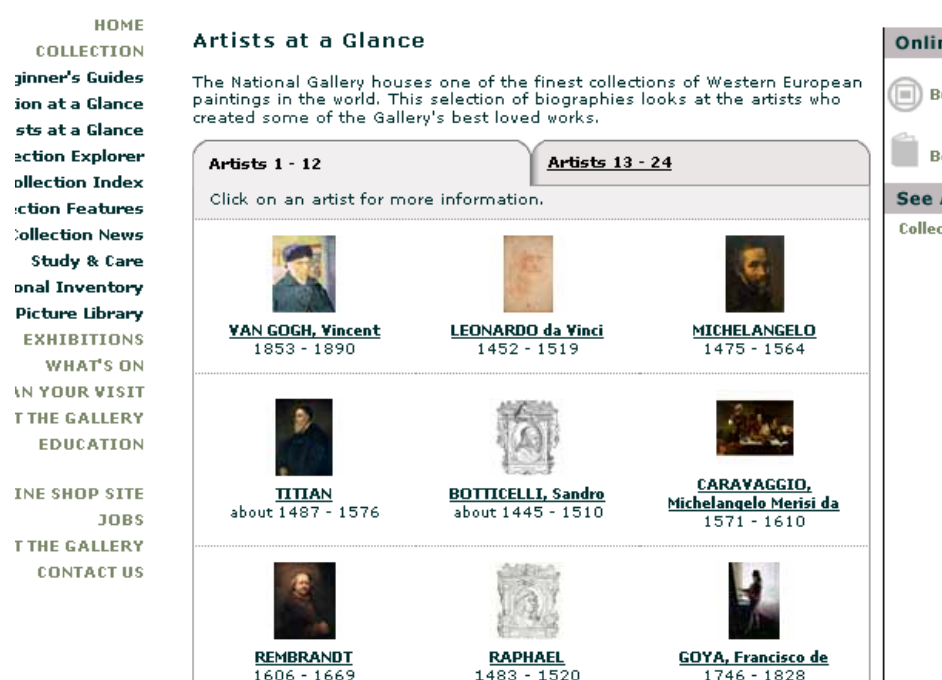


Figura 2.4: collezione degli artisti presenti nella National Gallery

2.2.2 *Tour virtuale*

Altri siti web di importanti musei propongono un'interfaccia più dettagliata e coinvolgente rispetto alla semplice catalogazione navigabile delle opere, pur rimanendo valido nella sostanza il concetto di catalogo comprendente schede di ciascuna opera. Ciò che differenzia in questo caso l'esperienza dell'utente presso il sito è il metodo di accesso alle singole schede e la veste grafica che assume la visita. Tipicamente gli sforzi dei realizzatori di questi siti web si concentrano nel rendere più realistico il percorso di raggiungimento delle pagine dedicate ai singoli oggetti. In questo caso si potranno ad esempio trovare piantine interattive anche tridimensionali, dalle quali accedere ai cataloghi delle opere contenute nelle varie stanze. In questo modo si riesce a familiarizzare anche con l'ambiente e con l'edificio del museo che spesso rappresentano più di un semplice contenitore per il materiale espositivo ma qualcosa che ha valore intrinseco. Un esempio di questo approccio ci viene offerto dal sito web dello State Hermitage Museum [4] in cui cliccando su *Virtual visit* si accede alla lista dei piani del museo. Successivamente abbiamo la possibilità di consultare la piantina di ogni piano del museo con relativa legenda delle stanze in riferimento alle opere contenute. In Figura 2.5 vediamo un'immagine della piantina del pianterreno.

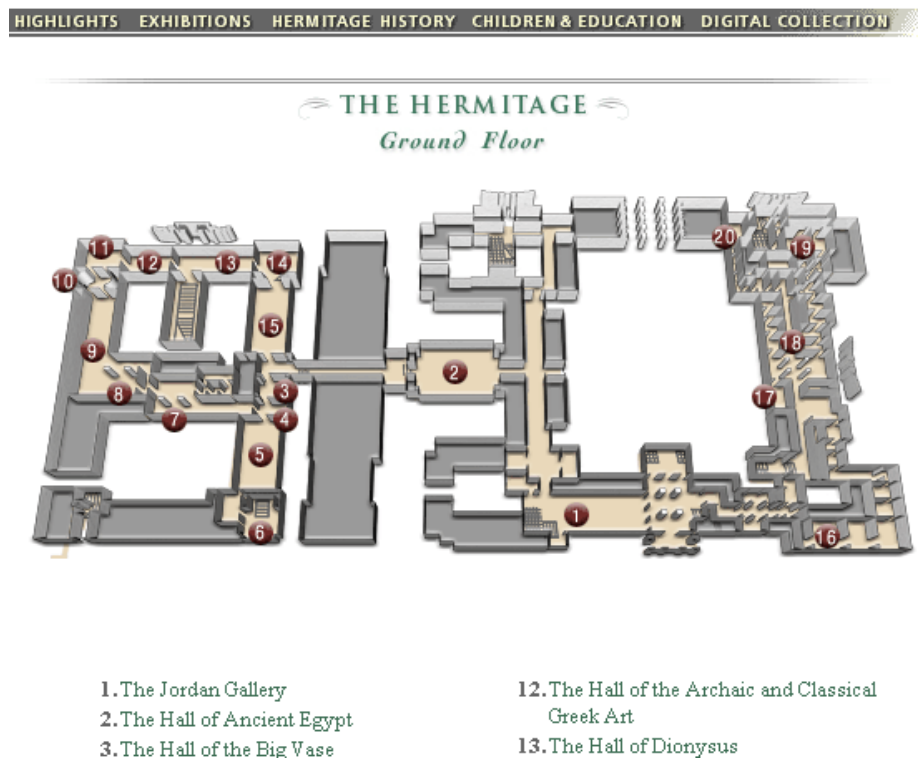


Figura 2.5: sito dello State Hermitage Museum, piantina del pianterreno

Un altro metodo per coinvolgere maggiormente il visitatore consiste nell'impiego di webcam che offrono immagini in tempo reale riprese dall'interno o dall'esterno del museo. Questo rappresenta un ulteriore tentativo di aumentare il realismo dell'esperienza vissuta dall'utente davanti al proprio computer, in quanto permette ad esso di vedere gli ambienti in cui si tengono le esposizioni. Un esempio dell'implementazione di questa tecnologia lo si trova sul sito del Centre Pompidou [5], nella sezione dedicata esclusivamente alle webcam. In Figura 2.6 troviamo un'immagine di questa parte del sito e la visuale offerta da una webcam.

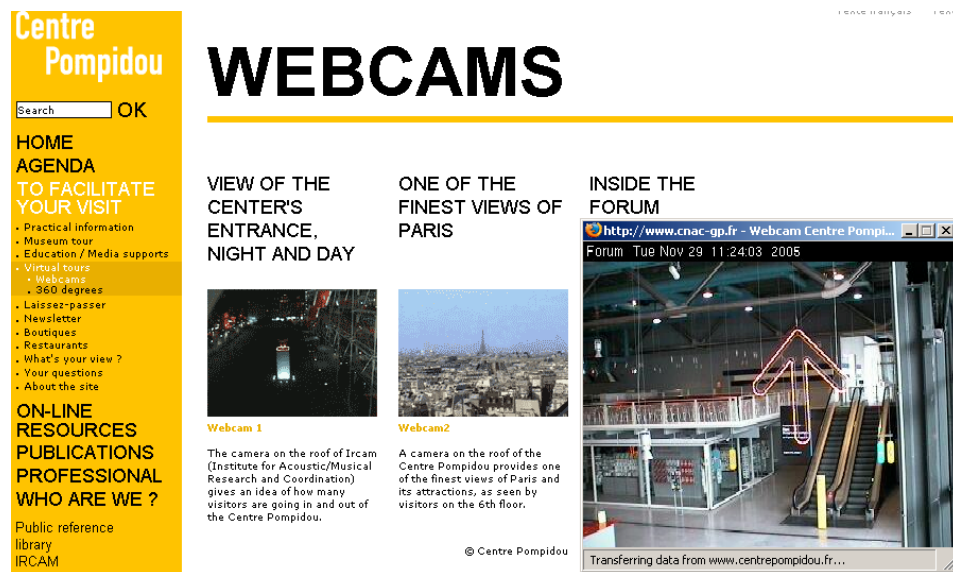
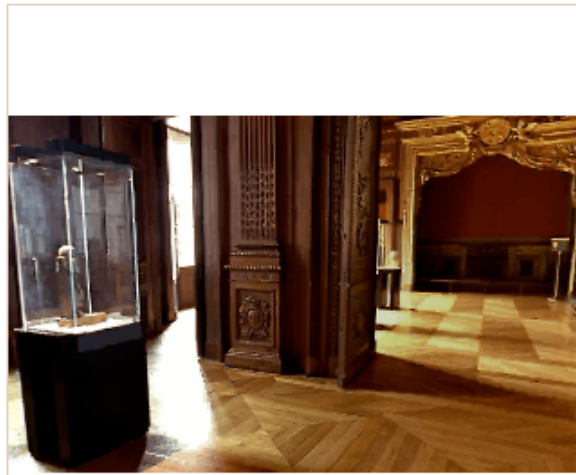


Figura 2.6: immagini in tempo reale dalle webcam del Centre Pompidou

Un'altra interessante tecnologia è quella proposta da Apple denominata Quick Time Virtual Reality (QTVR) [15] in cui la visuale dell'utente è posta al centro di una stanza ed è possibile ruotarla a 360° consultando in pratica un'immagine panoramica dell'ambiente. Tipicamente è possibile anche simulare l'avvicinamento alle pareti della stanza tramite una funzione di zoom ed avere una minima escursione verticale della visuale. Nonostante risultino limitate la qualità delle immagini proposte e la capacità di spostamento, questa tecnologia riesce indubbiamente nell'intento di immergere l'utente nell'ambiente reale di esposizione. Un esempio concreto di impiego di tale tecnologia lo si trova presso il sito web del museo del Louvre [6] di cui vediamo due immagini della stessa stanza vista da angolazioni diverse in Figura 2.7.

Virtual Tours: Egyptian Antiquities



Virtual Tours: Egyptian Antiquities

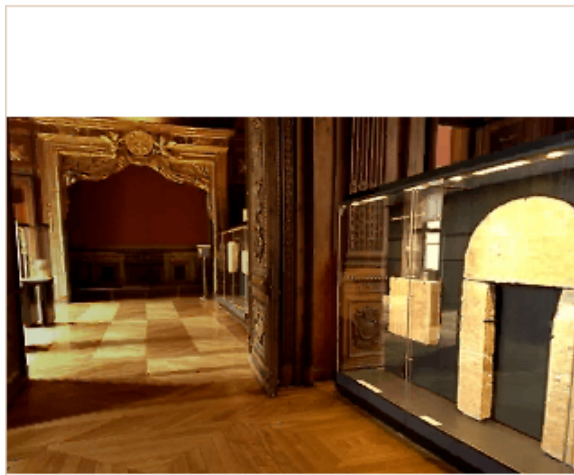


Figura 2.7: tramite la tecnologia QTVR è possibile ruotare di 360° la visuale attorno ad una posizione centrale

2.2.3 Museo virtuale

Considerando i paragrafi precedenti in sequenza ascendente da un punto di vista di coinvolgimento dell'utente e di realismo dell'esperienza offerta, un ulteriore passo avanti in questa direzione è quello offerto dai *musei virtuali*. In questo caso il visitatore viene immerso in un ambiente virtuale tridimensionale che può o riprodurre fedelmente il luogo fisico reale in cui sono collocati gli oggetti esposti, o proporre un qualsiasi altro, realmente esistente o di fantasia. Non solo gli ambienti ma anche le

opere vengono ricostruite tridimensionalmente; il linguaggio più diffuso per questo tipo di applicazioni è Virtual Reality Modeling Language (VRML) [16]. È questo un linguaggio ideato per poter essere utilizzato in rete, pertanto i musei virtuali che ne fanno uso sono accessibili semplicemente tramite browser, senza necessità di installazione di particolari applicativi lato client.

Un esempio di museo virtuale lo si trova nel Virtual Cardiac Museum [7], che in aggiunta mostra anche come l'applicazione del concetto di museo virtuale possa andare oltre la classica concezione di museo archeologico o di opere d'arte. In questo caso vengono mostrati cuori di diversi animali, ed avvicinandosi alla teca di un organo ne viene automaticamente visualizzata nella parte inferiore dello schermo una descrizione. In Figura 2.8 vediamo un'immagine presa da questo museo virtuale.



Figura 2.8: una stanza del Virtual Cardiac Museum

2.3 Musei virtuali creati dall'utente

L'interazione spinta ai massimi livelli tra museo e utente porta infine ad un ribaltamento dei ruoli: il visitatore infatti non è più un soggetto passivo che riceve le

informazioni come gli vengono proposte, pur potendo operare scelte più o meno dettagliate riguardo a ciò che gli interessa, ma diventa egli stesso protagonista nell'operazione di creazione del museo scegliendo le opere che vuole vedere esposte. Come abbiamo già avuto modo di notare, rientra in questa categoria anche l'oggetto della presente tesina.

Un esempio di questo approccio ci viene offerto dal Virtual Museum of Canada [8], in cui l'utente può aggiungere opere ad una galleria personale scegliendole da una collezione di immagini provenienti da diversi musei del Canada. In un secondo momento accedendo alla galleria personale così formata inoltre potrà aggiungere anche commenti a ciascuna opera in modo tale da renderne più completa la presentazione. Tuttavia non è prevista la tridimensionalità, ma viene invece utilizzato il funzionale seppur semplice approccio a catalogo già visto in esempi precedenti. In Figura 2.9 vediamo un esempio di collezione personale.

■ Click on an image to see the description you added.



Woodland Waterfall

Thomson, Tom
For © contact McMichael
Canadian Art Collection. All rights
reserved.



Untitled

Thomson, Tom
For © contact The Winnipeg Art
Gallery. All rights reserved.



Wildflowers

Thomson, Tom
For © contact McMichael
Canadian Art Collection. All rights
reserved.

Figura 2.9: collezione personale creata presso il sito del Virtual Museum of Canada

Capitolo 3

Analisi dei requisiti

*“Cato dixit litterarum radices amaras
esse, fructus iucundiores”*

Catone

3.1 Studio del museo di Ossuccio

Durante la fase di analisi dei requisiti abbiamo avuto più incontri con il Prof. Della Torre e con i progettisti della struttura che accoglierà il museo di Ossuccio. Il dato saliente emerso dalle riunioni è che, nonostante la grande quantità di reperti disponibili, solo una piccola frazione di questi verrà esposta. Inoltre, a parte alcuni “pezzi pregiati”, la collezione è composta principalmente da piccoli reperti come frammenti di marmo o chiodi di ferro di varie epoche, ossia elementi aventi un’importanza specifica relativamente modesta. Lo scopo principale del museo virtuale è dunque permettere all'utenza la fruizione dei reperti non esposti e di fornire al visitatore informazioni aggiuntive che vadano oltre la specificità del singolo reperto, dato che il museo vero e proprio per sua natura avrebbe potuto essere monotono e poco interessante per i non addetti ai lavori. È inoltre stato deciso che la visita del museo virtuale dovrà comunque partire dai reperti fisicamente esposti, rappresentanti gli elementi più importanti della collezione, per poi svilupparsi in direzioni scelte dall'utente in maniera interattiva.

I principali fruitori del software, sempre da quanto emerso dalle riunioni con Della Torre e con i progettisti, possono essere suddivisi in due categorie. La prima categoria è rappresentata dalle scolaresche, che utilizzeranno il software tramite la mediazione di un operatore. Sarà dunque necessaria la presenza di un proiettore per la visione collettiva del museo virtuale. La seconda categoria è composta da gruppi

turistici. Questi ultimi invece utilizzeranno individualmente o in piccolissimi gruppi il nostro software in postazioni differenti dislocate all'interno del museo.

A seguito dell'analisi dei profili di utenza abbiamo implementato il software tenendo conto della semplicità di utilizzo, della responsività nella navigazione del museo e dell'essenzialità delle informazioni visualizzate. Questi requisiti vengono analizzati nei tre punti seguenti.

- Per quanto concerne la semplicità di utilizzo la committenza ha richiesto che il software producesse un risultato “utile” all'utente in non più di due passaggi. Un percorso più lungo potrebbe scoraggiare l'utente nell'utilizzo del software e potrebbe portarlo a perdere l'orientamento tra i passaggi necessari all'ottenimento del risultato. Va infatti tenuto conto che nella maggior parte dei casi il visitatore si troverà di fronte al programma Minerva per la prima volta e dunque avrà naturali difficoltà di familiarizzazione. Da tutto ciò è derivata la necessità di rendere il più possibile breve la procedura di raggiungimento del risultato visivo rappresentato dalle opere inserite nel museo virtuale. Analoga importanza andava riposta inoltre nel dare a ciascun utente la facoltà di sviluppare un percorso di visita personalizzato in base a propri interessi o alle proprie curiosità facendo sì che questo percorso potesse essere costruito in maniera chiara e lineare, rendendo quindi la visita nel suo complesso fluida e senza soluzione di continuità.
- Altro aspetto importante è rappresentato dalla responsività del sistema: è necessario infatti che il tempo di risposta ai comandi dell'utente non sia superiore a qualche secondo al massimo. Questo è dovuto al fatto che il sistema sarà per lo più utilizzato dal pubblico, pertanto l'utente non esperto potrebbe venire facilmente scoraggiato da attese “lunghe”, ovvero da risultati non immediati o poco meno che immediati.

Inoltre ricordiamo che il progetto è orientato al web, quindi anche in prospettiva di una probabile futura pubblicazione su Internet è necessario che agli usuali tempi d'attesa dovuti alle infrastrutture di rete non si sommino eccessivi ritardi di elaborazione sul server, rendendo il tutto troppo lento alla percezione del visitatore.

- Infine le informazioni presentate devono essere concise ma complete, dando maggior risalto all'aspetto visivo legato alle immagini delle opere che non a quello testuale con descrizioni eccessivamente dettagliate e lunghe presentate insieme alle immagini. Altrettanta attenzione andava riposta alla chiarezza del nesso logico tra le scelte effettuate dall'utente e i risultati generati dal programma, in modo tale che risulti sempre evidente cosa rappresentano i risultati di una determinata ricerca quando questi vengono proposti al visitatore

3.2 Ricerca e scelta delle tecnologie appropriate

Gli elementi che caratterizzano il progetto Minerva sono la possibilità di accedere via web al museo e la semplicità d'uso del sistema. Allo stesso modo è necessario che il sistema sia stabile, scalabile e basato su tecnologie consolidate e compatibili con le versioni precedenti di Minerva. Come già detto Minerva è scritto in Java e quindi almeno per la parte centrale del sistema la scelta è stata obbligata: abbiamo quindi deciso di adottare l'ultima versione di Java ovvero la 5. Minerva era già un sistema ad agenti basato su JADE e questo è rimasto immutato.

Per la memorizzazione dei dati nelle versioni precedenti veniva utilizzato il database Access [11]: questa era una soluzione pratica ma poco scalabile, poco performante e limitata nella semantica sql [17]. Inoltre non garantiva la compatibilità con il mondo Unix limitando di fatto i vantaggi in termini di portabilità forniti dall'impiego di Java. Per la memorizzazione è stato quindi deciso di passare all'utilizzo

di un database relazionale indipendente. Sono state valutate varie alternative tenendo conto dei requisiti e del budget. La scelta è ricaduta su MySQL [12], un database di una software house svedese. MySQL è un ottimo prodotto, completo, stabile, performante, scalabile e, cosa non secondaria, disponibile sotto licenza Open Source [13]. È quindi fruibile in forma gratuita anche per scopi commerciali. MySQL supporta le ultime specifiche sql ed offre un'ottima gestione della concorrenza e della consistenza dei database. Per interfacciarlo a Minerva è sufficiente uno specifico driver JDBC anch'esso disponibile sotto licenza Open Source. MySQL è disponibile per molte piattaforme tra cui Windows [18], Linux [19] e Mac OS X [20].

Sebbene il progetto sia completamente multiplatforma e sia stato sviluppato su diversi sistemi, abbiamo eletto Mac OS X come ambiente di riferimento in virtù delle ottime prestazioni di questo sistema con applicazioni Java dovute a una particolare ottimizzazione della Java Virtual Machine.

Per quanto concerne l'interazione con l'utente e la fruibilità su web la ricerca è stata subito indirizzata verso prodotti sviluppati in Java in modo tale da poter comunicare facilmente con il Minerva server basato su JADE, a sua volta scritto in Java. Questo era dettato dall'esigenza di impiego di JADE su cui poggia la struttura ad agenti di Minerva. La necessità pertanto era quella di avere un'applicazione in grado di svolgere la funzione di web server per la comunicazione con l'utente, e che “parlasse” Java per potersi interfacciare senza problemi con Minerva. Nella comunicazione con il server di Minerva il web server assume perciò il ruolo di client, poichè da esso partono le richieste fatte dall'utente. La scelta per questo componente è caduta su Jakarta Tomcat [10], web server Open Source, prodotto dalla Apache Foundation [22], completamente scritto in Java e pensato per servire pagine web dinamiche avvalendosi della tecnologia servlet.

Le servlet sono moduli software scritti in Java che vengono eseguiti lato server per soddisfare le richieste dei client, che in questo caso sono rappresentati dai web browser degli utenti. Esse non sono legate ad un particolare protocollo per la comunicazione tra client e server, vengono eseguite all'interno di Tomcat e creano le singole pagine in modo dinamico in base alle richieste fatte dai client. Nel nostro caso le servlet, tramite una libreria che utilizza JADE, si mettono in contatto con il Minerva server, richiedono le informazioni per la creazione delle pagine del museo e poi le processano fornendo come output codice html leggibile da qualunque browser. In Figura 3.1 viene riportato un diagramma UML [23] *use case* che descrive il sistema e la sua relazione con l'ambiente e gli utilizzatori. In esso vediamo rappresentato l'utente che accede al sistema tramite il web browser collegandosi al server web (indicato in figura con "Servlet Web Server"); qui viene creato un agente "client" in grado di comunicare con il server di Minerva (l'agente *Server Agent*) in cui avviene lo scambio di informazioni tra i vari agenti, il recupero dati dal database e la comunicazione della risposta al web server che si occuperà di fornire la pagina html formattata al browser dell'utente.

Nella scelta di Tomcat un ruolo importante lo hanno giocato la chiara fama del prodotto e le sue riconosciute doti di stabilità e affidabilità nonché la buona scalabilità dello stesso. Tomcat è facilmente installabile, non necessita di configurazioni sofisticate e rende possibile fare interagire le richieste dei client con veri programmi Java senza le limitazioni di molte piattaforme web. Inoltre Jakarta Tomcat essendo scritto in Java non è legato a nessuna piattaforma specifica e ciò permette un più semplice deploy del progetto Minerva. Infine, come già detto, Tomcat è rilasciato sotto licenza Open Source e quindi non prevede il pagamento di nessuna licenza.

Accanto a questi strumenti abbiamo individuato Eclipse [24] come ambiente di sviluppo fortemente personalizzabile grazie a una struttura a plug-in efficiente e flessibile. In questo caso la preferenza è stata motivata dal fatto che si tratta di un ambiente full Java ed è dotato di un ottimo plug-in in grado di interagire in modo proficuo con Tomcat. Inoltre Eclipse fornisce validi strumenti per il debugging di serie, una sofisticata gestione del progetto e la gestione automatizzata della compilazione e della creazione della documentazione grazie alla tecnologia Ant [25]. Per Eclipse è inoltre disponibile il plug-in di Subversion (*svn*) [26], tramite il quale è possibile gestire progetti che vengono sviluppati in parallelo da più persone, come nel caso di Minerva. Auspichiamo che anche per il futuro sia mantenuta stabile la piattaforma di sviluppo.

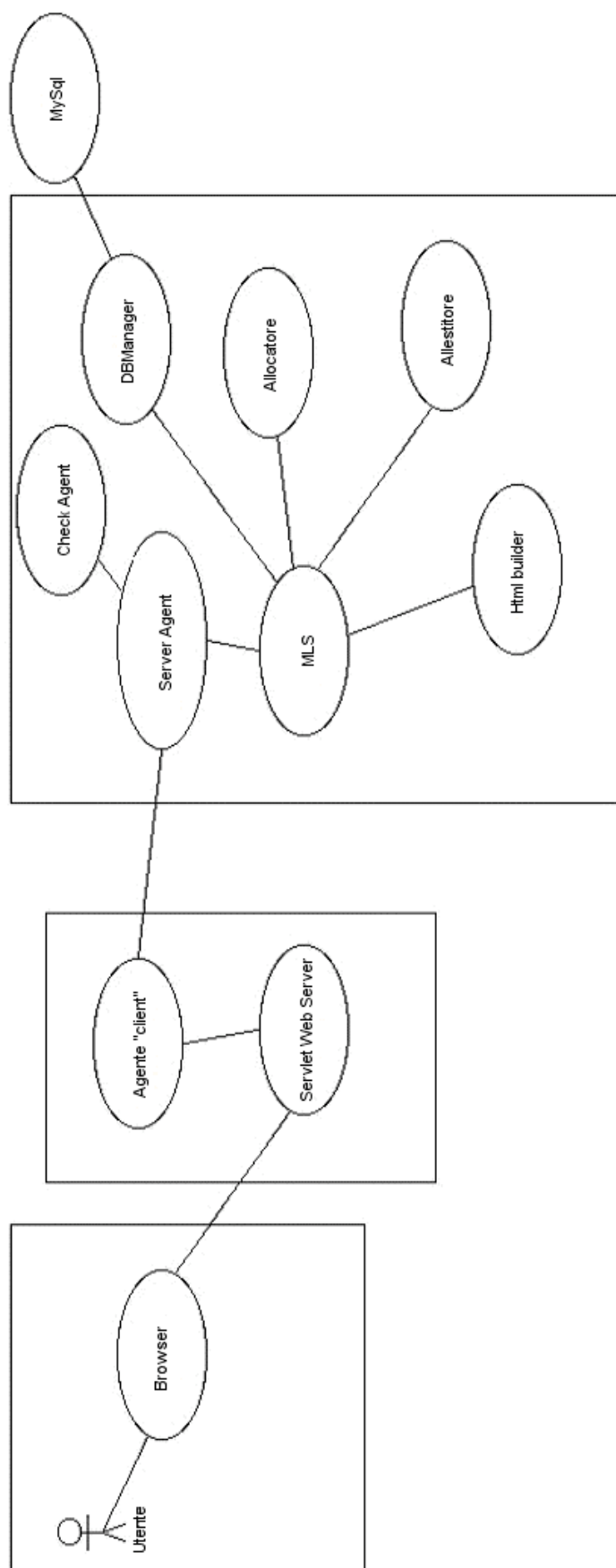


Figura 3.1: diagramma UML use case della struttura di Minerva

3.3 Reingegnerizzazione del software Minerva

Nello sviluppo della presente versione di Minerva si è resa necessaria la reingegnerizzazione di alcuni moduli software di Minerva05, in modo tale che si adattassero meglio alle esigenze del museo virtuale di Ossuccio.

Tra le prime modifiche è stata effettuata quella dell'agente che si occupa dello smistamento e dell'instradamento dei messaggi (denominato agente *MLS: MyLittleServer*) da e per gli altri agenti, che è stato reso più efficiente e maggiormente versatile. In quest'ottica si è scorporato il *comportamento* dell'agente dal resto del suo codice, in modo da poter gestire simultaneamente più comportamenti differenti senza intervenire nel suo codice di gestione. Va infatti ricordato che quanto deve fare un agente viene espresso all'interno del suo comportamento che viene implementato come un oggetto di una classe Java che estende `jade.core.behaviours.Behaviour`, separatamente dal restante codice per la creazione e la gestione dell'agente stesso.

Durante le fasi iniziali di sviluppo è stato notato che l'applicativo utilizzava la CPU dell'elaboratore al 100% per tutto il tempo della sua esecuzione. Tale comportamento è del tutto anomalo visto che così cospicuo impiego di potenza di calcolo è di norma associato ai soli momenti di elaborazione dei risultati in seguito a richieste da parte dell'utente. Una volta completata l'elaborazione, quindi quando si è in fase di attesa, un simile carico sul processore poteva essere giustificato solo da qualche implementazione non efficiente dei cicli software. Ricordiamo inoltre che nell'utilizzo tipico di Minerva i tempi di elaborazione dei risultati rispetto a quelli di fruizione dei risultati stessi da parte degli utenti sono nettamente inferiori, perciò lo spreco di tempo di CPU risulta ancora più evidente. All'atto pratico considerando un tempo di funzionamento del sistema nel suo luogo di impiego finale di diverse ore al

giorno, se non addirittura uno stato di attivazione continuo anche durante la notte per evitare lo spegnimento ed il riavvio di tutte le postazioni ogni giorno, tale impiego ingiustificato del processore oltre ad incidere negativamente sulle performance globali dell'applicazione comporterebbe un riscaldamento eccessivo dell'hardware. Ciò, in ultima analisi, porterebbe ad aumento del rumore legato ai sistemi di raffreddamento attivo (ventole delle cpu e dei case dei computer) e ad una diminuzione del tempo di vita dei processori. Lo scenario di attivazione 24 ore al giorno del sistema diventa poi inevitabile nel momento in cui il museo venisse reso pubblicamente accessibile su Internet, situazione in cui la CPU del server deve gestire anche accessi simultanei, rendendo totalmente inaccettabile uno spreco di risorse.

Passando in rassegna le possibili cause di tale comportamento anomalo si è rivelato responsabile del fenomeno il comportamento ciclico non bloccante della fase di ricezione di messaggi da parte del succitato agente MLS. Come ogni programma server, l'agente MLS deve stare continuamente in ascolto e tramite un comportamento ciclico ad ogni esecuzione del suo metodo `action()` deve controllare se è arrivato un messaggio ed eventualmente processarlo. L'implementazione precedente del comportamento di MLS non prevedeva una chiamata bloccante al metodo di ricezione dei messaggi, con la conseguenza che questo comportamento veniva continuamente eseguito ciclicamente tenendo la CPU sempre al 100% del carico. Per risolvere questo inconveniente è possibile eseguire il metodo `action()` del comportamento di MLS solo quando arriva un nuovo messaggio, tramite la ricezione bloccante. Il metodo `block()` (messo a disposizione dalle librerie di JADE per la creazione e la gestione degli agenti, come il metodo `action()`) contrassegna il comportamento come bloccato in modo che l'agente non lo scheduli per l'esecuzione fintanto che un nuovo messaggio non viene inserito nella sua coda messaggi sbloccando il comportamento e

dando all'agente la possibilità di processare il messaggio. Implementando questa modalità di ricezione è stato pertanto possibile eliminare il difetto di carico anomalo sul processore.

La versione precedente del server di Minerva implementava funzionalità per controllare se gli agenti fossero attivi e, nel caso non lo fossero, li riattivava. Concettualmente è infatti molto importante assicurarsi che tutti gli agenti siano attivi e nel caso non lo siano è importante implementare delle strategie per la gestione degli errori affinché il sistema risulti essere fault-tolerant.

Il principale difetto della vecchia versione del sistema era la mancata separazione e centralizzazione della funzionalità. Infatti il controllo veniva eseguito ogni volta che doveva essere spedito un messaggio, attivando un numero esagerato di controlli in caso di intenso traffico e invece generando controlli insufficienti o nulli in caso di basso traffico. Inoltre in caso di malfunzionamento di MLS il sistema andava in stallo perché non era in grado di accettare richieste dall'esterno. La soluzione da noi scelta è stata la creazione di un nuovo agente, denominato *Check Agent*, che si occupa di verificare e gestire in modo automatico a scadenze regolari la situazione degli agenti e, in caso di anomalie, ripristina la situazione ottimale. Questa soluzione è risultata essere inoltre anche molto più performante in termini di risorse occupate sgravando gli agenti di una funzione logicamente separata e onerosa in termini di computazione.

Attenzione è stata riposta anche al sistema di *logging* delle informazioni riguardanti il funzionamento del sistema. In Minerva05 le informazioni destinate allo sviluppatore erano visualizzate attraverso il terminale di sistema in modo

disorganizzato e senza uniformità. Questo impediva un facile controllo delle operazioni ed in caso di malfunzionamento era difficile individuare l'errore e l'agente che l'aveva causato.

Durante la fase di reingegnerizzazione del software abbiamo quindi aggiunto un sistema di registrazione degli eventi che permettesse di tenere traccia dei messaggi di notifica e di debug. Inoltre si è ritenuto opportuno sviluppare il nuovo sistema in modo da visualizzare le informazioni attraverso la console ma anche da registrarle in un file. Oltre al messaggio inviato dall'agente allo sviluppatore il meccanismo di notifica inserisce automaticamente le informazioni temporali ed il nome dell'agente mittente.

Il nuovo metodo di notifica fornisce ad ogni agente la possibilità di inviare due diversi tipi di messaggi: il primo permette agli agenti di comunicare informazioni di monitoraggio in modo da consentire all'amministratore del server Minerva di controllare il corretto funzionamento. Il secondo tipo di messaggio ha la funzione di inviare messaggi di debug utilizzabili dallo sviluppatore per individuare malfunzionamenti durante la fase di implementazione di nuove caratteristiche. Entrambi i tipi di notifica possono essere abilitati a discrezione sia dell'amministratore sia del progettista in qualsiasi momento, fornendo ad essi un sistema flessibile per il controllo di Minerva.

Infine è stato necessario apportare sostanziali modifiche all'agente che si occupa delle richieste al server del database (agente *DBManager*). Questo è stato causato da due fattori:

- il passaggio dalla tecnologia Access a MySql, con conseguente adattamento delle fasi di inizializzazione e collegamento al database sfruttando le classi messe a disposizione dallo specifico driver JDBC;
- la creazione ex novo di un database apposito per il museo virtuale di Ossuccio, che ha comportato la scrittura di query specifiche per le nuove tabelle per tutte le fasi di inserimento, modifica ed eliminazione dei dati.

Capitolo 4

Progetto logico e soluzione del problema

“Mater certa est, pater numquam”

Massima latina

La fase di progettazione del software riguarda tutte quelle attività che permettono di passare dalla raccolta ed elaborazione dei requisiti di un sistema software alla sua effettiva realizzazione. In quanto tale, la progettazione si pone come un ponte tra la fase di specifica e la fase di codifica. A partire da una descrizione funzionale più o meno formale del sistema, il risultato della fase in oggetto, ossia il progetto, costituisce un punto di partenza al fine di giungere direttamente alla scrittura del codice nel linguaggio di programmazione. Pertanto, è stato in questa fase che abbiamo deciso le modalità di passaggio da “che cosa” doveva essere realizzato a “come” la realizzazione doveva avere luogo. Nei prossimi paragrafi saranno indicati i vari passaggi di progettazione, che non si sono limitati unicamente alla fase immediatamente successiva all’analisi dei requisiti ma che, come spesso accade nel ciclo di vita di un prodotto software, hanno avuto luogo anche in momenti successivi man mano che nuove opportunità o esigenze si rivelavano a noi o alla committenza.

4.1 Definizione di item, temi e criteri

Nella fase iniziale di progettazione del software per la gestione del museo virtuale dell’isola Comacina è stato definito il concetto di *item*, il quale rappresenta l’astrazione di ogni oggetto presente all’interno di Minerva. Nel nostro caso l’idea di item si concretizza nell’opera archeologica, quale può essere un capitello, un’urna, un

frammento architettonico, ecc., ma altrettanto efficace potrebbe essere l'utilizzo di item per indicare, ad esempio, edifici, automobili, alberi o qualunque altro oggetto si voglia prendere in considerazione per sviluppare un museo virtuale concettualmente simile a questo.

Si è ritenuto opportuno memorizzare ogni item nel database attraverso una tabella apposita in cui, oltre ad un codice automaticamente assegnato dal sistema che tabella apposita in cui, oltre ad un codice automaticamente assegnato dal sistema che permette di identificare univocamente l'item, sono presenti i seguenti campi che danno all'amministratore la possibilità di caratterizzare in maniera dettagliata ogni elemento della collezione e di distinguerlo da tutti gli altri:

- *nome*,
- *descrizione*,
- *icona*,
- *priorità*.

Le proprietà nome e descrizione sono abbastanza autoesplicative, laddove il nome sarà utilizzato come titolo associato all'oggetto nella presentazione mentre nella descrizione potranno essere inserite informazioni testuali specifiche più o meno dettagliate ad esso riferite. L'attributo icona memorizza l'icona associata all'item, ossia una piccola immagine che verrà utilizzata per rappresentare in modo visivo l'oggetto inserito nel museo virtuale insieme ad altri e poterlo quindi identificare in modo più immediato. Infine la priorità è un valore numerico che ci permette di fornire un ulteriore parametro di ordinamento agli oggetti in modo tale, ad esempio, da visualizzare per primi quelli con priorità maggiore, oppure magari di nascondere temporaneamente quelli con priorità sotto una certa soglia. In questo senso la priorità rappresenta l'importanza che si vuole attribuire all'oggetto. Come per le proprietà

precedenti la priorità viene definita durante la fase di inserimento di un nuovo item.

Terminata la definizione dell'item si è reso necessario associare ad esso dei contenuti informativi che legassero i diversi item presenti nel database permettendo in questo modo la creazione di insiemi aventi in comune lo stesso contenuto informativo. A questo scopo è stato quindi definito il concetto di *tema* come un argomento specifico che può venire associato a più item all'interno di Minerva. Ad esempio si può considerare il tema dell'impiego di un certo tipo di materiale, al quale saranno associati tutti gli item prodotti con quel materiale. Questo permette di presentare all'utente l'opera definita non esclusivamente dalle sue proprietà di nome, descrizione, ecc., e da sue ulteriori specifiche caratteristiche definite dai valori dei criteri (di cui tratteremo tra poco) ma anche tutta una serie di temi associati. Sfruttando l'interattività messa a disposizione da parte del software l'utilizzatore potrà scegliere quali tra questi argomenti, o temi, approfondire e conseguentemente quali opere includere nel proprio museo virtuale. Lo scopo che si vuole raggiungere con il concetto di tema è di natura principalmente didattica poiché deve guidare l'utente non specialista nella visita del museo. Deve anche "raccontare" all'utente perché certi reperti sono importanti e a cosa si ricollegano, ad esempio supponiamo di parlare del marmo del Duomo di Milano, la descrizione del tema metterà in luce che esso proviene da Candoglia in Piemonte e che veniva trasferito a Milano grazie ai Navigli. Cioè deve trasmettere ciò che c'è intorno al reperto e che forse l'utente già conosce ma non avrebbe mai messo in relazione.

Nella fase di progettazione si è poi reso necessario definire dei valori attraverso i quali poter suddividere e raggruppare gli oggetti visualizzati, sono stati

perciò associati ad ogni item dei valori per alcuni *criteri*. I criteri sono caratteristiche possedute da tutti gli item e in base al valore dei quali è quindi possibile individuare dei sottoinsiemi di oggetti, i cui elementi potranno poi essere visualizzati in ambiti comuni, ossia nella stessa stanza o, in casi particolari, nella stessa teca (i concetti di stanza e teca verranno illustrati nei prossimi paragrafi). Esempio classico di criterio può essere considerato la data di produzione, oppure nel caso di reperti archeologici, come nel nostro caso, il periodo storico a cui l'oggetto risale.

I criteri che sono stati individuati per il museo virtuale dell'isola Comacina sono i seguenti:

- *cronologico*: indica il periodo a cui risale il reperto;
- *topografico*: indica il luogo in cui è stato rinvenuto il reperto;
- *tipologia di reperto*: indica la categoria del reperto (ad es. epigrafe, ceramica, ecc.);
- *modalità di rinvenimento*: indica in quale campagna di scavi è stato rinvenuto il reperto.

La selezione dei temi ci permetterà di individuare l'insieme di oggetti che faranno parte del museo virtuale (ossia tutti gli oggetti collegati ai temi selezionati), all'interno del quale saranno suddivisi in gruppi in base al valore del criterio di ordinamento scelto.

4.2 Definizione di stanze

Nella fase di progettazione dell'interfaccia utente di Minerva è risultato evidente come sarebbe stata poco pratica la fruizione del museo virtuale se all'utente fossero state presentate tutte le opere selezionate dal software attraverso una semplice lista o in una disposizione spaziale casuale. Si è quindi giunti alla conclusione che per

migliorare la fruizione da parte dagli utilizzatori, anche in funzione di una maggiore coerenza con la realtà, la soluzione migliore fosse raggruppare le opere in *stanze*: in esse si trovano gli item accomunati dallo stesso valore del criterio selezionato. Questa suddivisione permette una più facile interpretazione dei risultati ottenuti offrendo la possibilità di focalizzare l'interesse attorno ad un gruppo piccolo di opere. In testa a ciascuna stanza viene indicato il valore del criterio di ordinamento che accomuna i reperti contenuti al suo interno. Naturalmente il numero di stanze presenti nel museo virtuale corrisponde al numero di valori del criterio di ordinamento scelto delle opere collegate ai temi selezionati. Maggiore è il numero di temi scelti, più elevato sarà il numero di reperti mostrati ed allo stesso tempo più numerose probabilmente saranno le stanze create. Più avanti in questo capitolo vedremo che all'interno delle stanze le opere verranno collocate in altri contenitori più piccoli denominati *teche*.

4.3 Interazione dell'utente e navigazione all'interno del museo virtuale

Presso il comune di Ossuccio i visitatori verranno accolti in un centro espositivo dove saranno presenti le opere principali rinvenute sull'isola Comacina. Nel museo saranno posizionate alcune postazioni multimediali con le quali sarà possibile consultare l'ampio archivio di oggetti rinvenuti che purtroppo non sarà possibile esporre integralmente in loco.

L'utente, interagendo con Minerva, vedrà visualizzata la planimetria dell'edificio in cui si trova e nella quale risulteranno evidenziati gli oggetti fisicamente esposti partendo dai quali si potrà creare un museo virtuale personalizzato. Una volta selezionato il reperto d'interesse dell'utente verranno mostrate maggiori informazioni a riguardo: foto dettagliate e un'ampia descrizione.

Nella parte sottostante dello schermo saranno poi elencati i temi correlati all'oggetto ed i possibili criteri di ordinamento. Durante l'esplorazione del museo virtuale l'utente avrà la possibilità di selezionare uno o più temi. Quest'ultimi determineranno quali oggetti saranno visualizzati come risultato del processo di creazione: infatti rientreranno nella collezione tutti gli oggetti correlati ai temi selezionati. Oltre ai temi l'utente dovrà scegliere uno tra i quattro criteri di ordinamento individuati, in modo tale da suddividere i reperti della collezione in stanze diverse in base al valore del criterio selezionato.

Una volta creato il museo virtuale tutti gli oggetti che verranno presentati al suo interno avranno un collegamento alla propria scheda, dove l'utente potrà approfondire la sua conoscenza sul reperto e, allo stesso tempo, iniziare una nuova esplorazione: saranno infatti presenti i temi correlati ad esso e i criteri d'ordinamento.

4.4 Pannello di amministrazione

Durante lo sviluppo delle funzionalità di Minerva si è riscontrata la necessità di fornire all'amministratore un'interfaccia semplice ed agevole per operare sul database. Si è quindi provveduto allo sviluppo di un pannello di amministrazione attraverso il quale è possibile effettuare tutte le operazioni necessarie a mantenere il museo virtuale sempre aggiornato.

Attraverso un'interfaccia amichevole, che non richiede alcuna conoscenza di programmazione o delle tecnologie di database, è possibile amministrare le opere che andranno a comporre il museo virtuale.

Per ognuna di esse sarà possibile specificare:

- *nome*
- *descrizione*
- *fotografie*
- *temi associati*
- *criteri*
- *priorità*

Sempre tramite pannello di gestione l'amministratore potrà inserire i temi e le relative descrizioni che permetteranno all'utente di avere maggiori informazioni. Di notevole importanza è l'assistenza offerta dall'applicativo durante l'inserimento del testo relativo ai temi. Attraverso semplici operazione è possibile includere in esso i collegamenti ipertestuali alle risorse esterne (argomento che verrà trattato in seguito nel capitolo) anch'esse presenti nel database. In questo modo si solleva l'amministratore dall'esigenza di conoscere i dettagli del linguaggio html con cui vengono creati i link nelle pagine web. È possibile utilizzare gli stessi link esterni in più temi migliorando quindi le interconnessioni tra i temi.

Oltre alla fase di inserimento il pannello di controllo permette naturalmente anche la modifica o la cancellazione di tutti i dati inseriti precedentemente.

4.5 Soluzioni individuate nella fase di testing

Durante la fase di testing, che ha accompagnato tutto lo sviluppo del progetto, sono state individuate varie esigenze riguardanti diversi aspetti del sistema che non erano emerse durante la fase preliminare di analisi dei requisiti o nel corso dell'implementazione del primo prototipo. Nei prossimi paragrafi andremo ad analizzare questi aspetti che rappresentano una parte fondamentale della versione

finale del nostro software. La fase di testing è stata inizialmente svolta confrontandoci con l'Ing. Francesco Amigoni e con la Dott.ssa Viola Schiaffonati; in un secondo tempo, dopo la realizzazione del primo prototipo, un ruolo importante nelle fasi di supervisione e testing lo ha ricoperto l'Arch. Andrea Bonavita che parallelamente stava lavorando alla creazione di un buon database per il museo virtuale.

4.5.1 Definizione di teche

Confrontandoci con Andrea Bonavita abbiamo constatato che gli oggetti presenti in ogni stanza erano in numero elevato e spesso presentavano caratteristiche molto simili e in alcuni casi anche lo stesso nome, pur trattandosi di item differenti presenti nel database. Questo problema si è presentato per la natura intrinseca del museo, infatti a parte alcuni “pezzi pregiati” la collezione è composta principalmente da piccoli reperti come frammenti di marmo (due esempi sono mostrati in Figura 4.1). In effetti è anche questo uno dei motivi per cui si è pensato di realizzare un museo virtuale con informazioni aggiuntive, dato che il museo vero e proprio avrebbe potuto essere monotono e poco interessante per i non addetti ai lavori.



Figura 4.1: esempi di reperto: a sinistra una pila termale, a destra un frammento lapideo

Si è quindi presa in considerazione l'idea di creare all'interno delle stanze

delle *teche* che contenessero tutti gli elementi caratterizzati dallo stesso nome (ad esempio “embrice” o “frammento lapideo con decorazione”). In questo modo il museo virtuale assume una conformazione più raccolta permettendo all'utente di identificare più facilmente gli oggetti simili. Questi vengono mostrati all'interno delle teche tramite l'icona associata a ciascuno di essi, la quale, trattandosi di un collegamento, permette di accedere direttamente alla pagina dove vengono mostrate tutte le informazioni relative al reperto.

Nel caso, infine, si volesse dare maggiore risalto a qualcuno di questi oggetti è possibile assegnare ad esso un valore di priorità differente dagli altri “omonimi”, in modo tale da non farlo rientrare nella stessa teca e dedicargli uno spazio separato all'interno della stanza.

4.5.2 Definizione di risorse esterne

Nello sviluppo di Minerva è stata notata la necessità ma soprattutto la possibilità offerta dal mezzo di poter approfondire i temi correlati al museo attraverso risorse non presenti direttamente nel database ma residenti su siti Internet esterni. Questa possibilità era stata intravista fin dall'inizio ma si è configurata come un'esigenza primaria della committenza solo successivamente alla prima fase di analisi dei requisiti, poiché ragionando unicamente sulla carta non era stato possibile individuare in modo dettagliato tutte le caratteristiche che avrebbe dovuto possedere il prodotto finale. Si era perciò convenuto di rimandare la definizione di questo aspetto a dopo la presentazione del primo prototipo in modo tale che potesse essere maggiormente coerente con il sistema fino ad allora implementato. I link esterni sono una delle principali novità di questa versione del progetto Minerva perché aprono un sistema chiuso al mondo esterno, svelando una fonte di conoscenza virtualmente

infinita quale Internet appare oggi, e permettono un notevole accrescimento dell'esperienza didattica dell'utente indirizzandolo verso contenuti avanzati e diversificati ed allo stesso tempo selezionati da un esperto. In futuro non è escluso che questa funzione possa essere potenziata attraverso ricerche intelligenti e personalizzabili via web. Per questa versione del progetto è stata valutata e sviluppata l'idea di aggiungere all'interno delle descrizioni dei temi dei collegamenti verso le "risorse esterne", ossia verso pagine in cui viene approfondito l'oggetto collegato e riguardo al quale vengono presentati un breve testo, una fotografia ed i relativi link alle risorse presenti in Internet. Si è ritenuto necessario non abbandonare completamente l'utente nella visita dei siti esterni e si è quindi deciso di inserire nella parte superiore dello schermo una barra, sempre presente, che permettesse un agevole spostamento tra i siti collegati alla risorsa e allo stesso tempo un immediato e veloce ritorno al museo virtuale precedentemente creato.

Tornando all'esempio del tema del Duomo di Milano un link esterno potrebbe essere relativo alle cave di Candoglia, mentre un altro potrebbe raccontare la storia dei navigli e la loro importanza nel commercio e nella vita di Milano fino alla fine del diciannovesimo secolo. A quest'ultimo potrebbero essere associati una serie di siti web sulla costruzione dei navigli, sulla loro successiva copertura e sul progetto di una nuova valorizzazione degli stessi.

4.5.3 Raffinamento dell'interfaccia utente

Al termine del primo prototipo, la pagina iniziale di Minerva presentava solamente la lista dei tre oggetti maggiormente significativi collocati all'interno del museo dell'isola Comacina che ammonteranno a dieci quando sarà completato il popolamento del database. Si è quindi deciso di sostituire la semplice lista con una

piantina dell'edificio dove sono collocati i reperti in modo da offrire all'utente un immediato riscontro con l'area che li circonda, dando davvero l'idea di Minerva come di uno spazio virtuale che parte dal reale per estenderlo.

Sempre nel primo prototipo, la pagina di presentazione del museo virtuale visualizzava esclusivamente le stanze e gli oggetti corrispondenti ai temi selezionati, come si può vedere in Figura 4.2. Questa modalità non ricordava all'utente da quale oggetto fosse iniziata la realizzazione del museo e si è quindi ritenuto opportuno introdurre anche in questa pagina le informazioni relative all'opera da cui si è partiti. Sono dunque stati inseriti il nome e la fotografia all'inizio della pagina con la descrizione visualizzabile tramite una comoda ed elegante interfaccia a scomparsa. Sempre grazie alla suddetta interfaccia è possibile consultare anche la lista di tutti i temi correlati all'oggetto ed i criteri di ordinamento. In questo modo l'utente può creare un nuovo museo virtuale partendo dall'opera precedentemente individuata con tutte le informazioni ad essa correlate.

Museo virtuale dell'isola Comacina



Figura 4.2: il museo virtuale nel primo prototipo

Dopo aver portato a termine queste prime modifiche all'interfaccia utente di Minerva si è notata la necessità di aggiungere una descrizione ai temi presenti in modo da fornire all'utilizzatore maggiori dettagli e notizie più approfondite. Questa è stata un'esplicita richiesta della committenza perché in questo modo è possibile migliorare le potenzialità didattiche e allo stesso tempo valorizzare i “pezzi pregiati” del museo da cui si dipanano le descrizioni dei temi.

Ciò ha portato ad una nuova modifica importante per quanto riguarda il software: nel database è stata aggiunta la possibilità di inserire un testo per ogni tema nel quale erano presenti anche dei collegamenti alle risorse esterne. Si è quindi affrontato il problema di trovare una collocazione spaziale appropriata delle descrizioni dei temi all'interno della pagina che mostrava il museo virtuale e l'oggetto da cui aveva avuto inizio la costruzione dello stesso. La soluzione ritenuta più adeguata è stata quella di collocare i temi e le relative descrizioni selezionate dall'utilizzatore in una colonna laterale che affiancasse le aree destinate alla visualizzazione dell'opera di partenza con la sua descrizione e quella sottostante più ampia dedicata al museo virtuale vero e proprio. Per non appesantire eccessivamente la presentazione si è adottata la soluzione di fornire all'utente una breve anteprima del testo e la successiva possibilità di visualizzarlo completamente sempre sfruttando un'interfaccia a scomparsa.

Al termine di questa fase di sviluppo e durante la fase di test è apparso chiaramente che la pagina in cui prima era presente solo il museo virtuale era diventata un punto di riferimento importante per l'utente permettendogli di svolgere tutte le operazioni possibili. In essa erano visualizzati i reperti selezionati da Minerva in base ai temi scelti e ordinati in base al criterio, era possibile accedere alle

descrizioni dei temi ed agli eventuali collegamenti esterni ed infine si poteva creare un nuovo museo virtuale. Questa abbondante presenza di informazioni in un'unica pagina poteva disorientare l'utente poiché le informazioni non erano separate graficamente in modo adeguato. Si è quindi deciso di creare tre aree distinte: il museo virtuale vero e proprio, i temi correlati e i riferimenti all'opera di partenza. Per identificare le aree si è deciso di utilizzare colori diversificati: verde per gli elementi correlati all'oggetto dal quale si era partiti per la creazione del museo virtuale, arancione per evidenziare i temi e la loro descrizione ed infine giallo per far risaltare il museo virtuale contenente le stanze e le teche. I colori sono stati scelti in modo tale che sia immediato distinguere le diverse zone fin dal primo sguardo superficiale, indirizzando l'attenzione dell'utente verso le informazioni di maggior interesse. Allo stesso tempo la loro tonalità non doveva ostacolare la fruizione del museo né rendere faticosa la lettura dei contenuti presenti. In Figura 4.3 si può vedere una schermata del prodotto finale raffigurante lo stesso museo virtuale presentato in Figura 4.2.



Figura 4.3: il museo virtuale nella versione finale

4.5.4 Criteri di raggruppamento

Nel primo prototipo realizzato l'utente aveva la possibilità di combinare diversi criteri per la creazione del museo virtuale. Dopo i primi test con dati reali inseriti nel database però si è riscontrato che una scelta multipla dei criteri produceva un ordinamento non coerente con gli scopi didattici ed informativi per cui il sistema era pensato. Con una successiva analisi si è deciso di utilizzare un solo criterio per l'ordinamento permettendo a Minerva di fornire all'utilizzatore informazioni con una maggiore valenza informativa. In ultimo è stato ritenuto opportuno visualizzare il criterio di ordinamento scelto nell'intestazione del museo virtuale in modo da poter aver sempre presente la scelta effettuata in precedenza.

Capitolo 5

Architettura

*“Est aliquid valida scepra tenere
manu”*

Ovidio

L'argomento di questo capitolo è l'architettura del sistema Minerva da un punto di vista generale, senza che si introducano dettagli implementativi. Infatti la realizzazione pratica rientra negli obiettivi della tesina curata da Calcaterra e Franzosi a cui si rimanda il lettore per l'approfondimento, mentre, come già detto, questa tesina si sofferma maggiormente sull'analisi dei requisiti e sulla definizione e risoluzione logica dei problemi che si sono incontrati durante la realizzazione del progetto.

5.1 Architettura generale del sistema

Il sistema è costituito essenzialmente da un client, un server ed un database, secondo la classica struttura a tre livelli. A tutto questo poi bisogna aggiungere ovviamente il browser dell'utente finale per la visualizzazione delle pagine (in Figura 5.1 una raffigurazione dell'architettura).

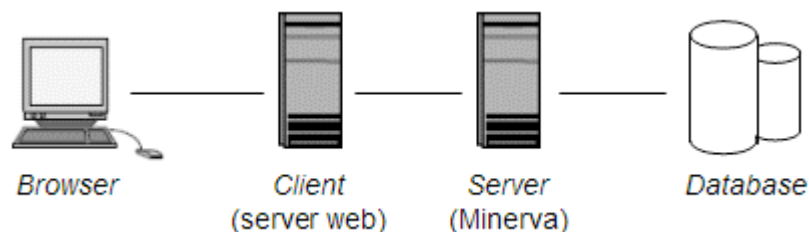


Figura 5.1: architettura del sistema

Il client, implementato appositamente per questo progetto, è costituito da tre componenti situati presso il server web Tomcat: una servlet, un agente lato client e MinervaLog.

- La servlet è responsabile della diretta interazione con l'utente finale, ricevendo ed elaborando le richieste effettuate e organizzando i risultati ottenuti attraverso la creazione di pagine html dinamiche.
- L'agente lato client è invece adibito alla comunicazione con il server tramite messaggi. Le comunicazioni fra l'agente client e l'agente server sfruttano lo standard FIPA [14] per la costruzione dei messaggi e vengono gestite in base alla richiesta specifica: il client mette a disposizione della servlet un repertorio di funzioni per il reperimento delle informazioni richieste al server, attivate a seconda dell'esigenza. Il protocollo con cui vengono effettuate le richieste da parte della servlet al server Minerva utilizza lo standard http e permette di avere robustezza e scalabilità nell'applicativo in caso di intenso traffico tra gli eventuali server.
- Il terzo componente, MinervaLog, è incaricato di tenere traccia dei messaggi scambiati tra gli agenti memorizzandoli in un file esterno, consultabile dallo sviluppatore per il monitoraggio delle attività da essi svolte.

Per quanto riguarda il server siamo partiti dalla struttura ad agenti di Minerva05 basata su JADE [9], aggiungendo soltanto il nuovo agente Check Agent con il preciso compito di controllare che gli altri agenti siano presenti e attivi nel sistema e nel caso non lo fossero di riattivarli. Tale funzionalità è un requisito fondamentale perché il sistema sia fault-tolerant ed era già implementata nella precedente versione di Minerva ma con un approccio non centralizzato che ne

comportava l'esecuzione in occasione della trasmissione di ogni messaggio. In questo modo il controllo dello stato degli agenti veniva eseguito in maniera non omogenea, essendo totalmente assente in mancanza di messaggi scambiati ed al contrario gravando significativamente sulle prestazioni in situazioni di intensi scambi comunicativi. L'introduzione di un agente adibito a questo scopo ha portato sia ad una maggiore robustezza del sistema sia ad un miglioramento delle performance, poichè ora il traffico dei messaggi è ottimizzato. Per una più esaustiva trattazione dell'argomento si rimanda al paragrafo 3.3.

Un'altra novità per il server è rappresentata dal file di configurazione che è stato introdotto, che permette di definire in modo immediato quali agenti si intende attivare, possibilità che si è rivelata molto utile a noi stessi durante la fase di testing. Questa agevolerà anche il compito di eventuali altri sviluppatori futuri del progetto Minerva in quanto li solleverà dalla necessità di intervenire direttamente sul codice, riducendo drasticamente i tempi di sviluppo.

Per quanto riguarda invece le funzionalità primarie abbiamo reputato non necessaria la creazione di nuovi agenti, osservando in fase di analisi dei requisiti che sarebbe stato possibile implementare tutte le funzionalità del nuovo software sfruttando gli agenti esistenti, attraverso opportune modifiche ed aggiornamenti. In pratica, abbiamo aggiunto all'agente MLS tutte le funzioni necessarie per la comunicazione e l'inoltro dei nuovi messaggi sia in fase di richiesta sia in fase di risposta ed all'agente DBManager tutte le interrogazioni da effettuare al database specifiche per questo progetto. In questo modo è stata garantita la retrocompatibilità del server con il client del software Minerva05. Infine è stato fatto un notevole lavoro di ottimizzazione degli agenti, in particolare sono stati corretti vari piccoli errori che creavano situazioni di stallo in attesa di messaggi in cui il processore era impegnato al

100% anche se in realtà non svolgeva nessuna operazione utile (vedere paragrafo 3.3 a riguardo). È poi stato ristrutturato il codice di Server Agent e MLS in modo da renderlo più leggibile anche in vista del passaggio di consegne a nuovi sviluppatori.

Per quanto riguarda il server sql è stato creato appositamente il nuovo database “Ossuccio” e affiancato a quelli già esistenti ereditati dal progetto precedente. I motivi di questa scelta sono stati vari: innanzitutto la separazione dei database permette in fase di installazione di ignorare i database non necessari riducendo i tempi di caricamento. In secondo luogo contribuisce ad una maggiore chiarezza nella struttura e, trattandosi di un nuovo database costruito ad hoc per questo progetto, è stato possibile raggiungere un maggior livello di ottimizzazione.

Separatamente, ma di pari passo rispetto al sistema vero e proprio, è stato implementato Minerva Admin, una seconda servlet adibita alla gestione e alla manutenzione del database. Grazie ad essa è possibile effettuare delle operazioni di inserimento, modifica e cancellazione dei dati senza avere conoscenza precisa dello specifico schema adottato per il database né di alcun linguaggio di programmazione. Minerva Admin permette quindi ad un qualsiasi amministratore di interagire facilmente con il database per modifiche ed aggiornamenti futuri dei dati e delle opere del museo. Questo strumento risulterebbe molto utile anche nel caso in cui si procedesse ad una generalizzazione del progetto Minerva e dunque all'applicazione di questo software per altri musei, in quanto faciliterebbe la fase di popolamento del nuovo database.

5.2 Obiettivi tesina Calcaterra-Franzosi

Come già detto nell'introduzione della tesina il lavoro di sviluppo di questa versione di Minerva è stato gestito da quattro persone, infatti oltre agli autori di questa tesina il gruppo era formato da Gianluigi Calcaterra e Massimo Franzosi.

Nell'intero ciclo di vita di Minerva, dall'analisi dei requisiti ai ritocchi finali prima della consegna del prodotto alla committenza, la divisione dei ruoli non è stata eccessivamente rigida e ci si è spesso suddivisi i compiti in base alle esigenze o alle competenze specifiche di ciascun membro del gruppo. Le fasi del processo non sono state nettamente separate soprattutto dal punto di vista temporale poiché il modello di sviluppo adottato è stato quello a spirale, consentendo ad analisi ed implementazione di intrecciarsi più volte anche ad opera di individui diversi.

I principali obiettivi della tesina di Calcaterra e Franzosi, che riguarda l'implementazione del sistema, erano la stabilità, la scalabilità e le performance del sistema. Inoltre era necessario realizzare un'interfaccia grafica accattivante ma non eccessivamente elaborata, in quanto non rientrava nei nostri scopi accademici concentrarci su questo aspetto, che potrà eventualmente essere sviluppato ulteriormente da una terza parte in seguito. Ciò che ci era richiesto era che l'applicazione si presentasse facile nell'impiego ed amichevole all'uso dell'utente non esperto. Doveva infine essere approntata un'applicazione in grado di gestire l'inserimento e la manutenzione dei dati relativi al museo.

I risultati pratici del lavoro sono stati in linea con le aspettative. In particolare sono migliorate le performance del server di Minerva senza in alcun modo intaccare la sua stabilità. Un punto inizialmente dubbio era costituito dai tempi di risposta del sistema nel suo complesso, non stimabili in modo rigoroso a causa della compresenza di più applicativi in reciproca interazione. Anche alcune esperienze passate in gruppi

di ricerca legati a Minerva avevano avuto esiti non positivi. I tempi di risposta invece ora sono molto buoni, in particolare su architetture Apple. Nella tesina dei nostri colleghi vengono esposti chiaramente i particolari della valutazione delle prestazioni sulle diverse architetture. Ricordiamo infatti che questo progetto è stato sviluppato noi sviluppato sui tre ambienti Mac, Linux e Windows, inducendoci da un lato a massimizzarne la portabilità adottando tecnologie disponibili per ciascuna piattaforma (in questo modo sfruttando e valorizzando una delle caratteristiche principali del linguaggio di programmazione impiegato, ossia Java, che per l'appunto nella portabilità ha uno dei suoi più importanti punti di forza), dall'altro a ricercarne il miglioramento delle performance su ciascuno dei sistemi adottati. Alla luce dei fatti si è constatato chiaramente come la piattaforma Apple sia stata quella in assoluto più performante, fornendo tempi di attesa molto ridotti e sempre accettabili. Le ragioni di ciò vanno a nostro avviso principalmente imputate all'implementazione specifica della Java Virtual Machine adottata da Mac OS X, ottimizzata per questo sistema operativo. Il sistema Windows si colloca al secondo posto nella "classifica" legata alle prestazioni nell'esecuzione del software Minerva, mentre piuttosto deludenti, nonché inaccettabili per un sistema proposto al pubblico, ci sono apparsi i tempi di risposta del sistema basato sulla distribuzione Ubuntu di Linux. In merito a queste riflessioni sulle performance è doveroso ricordare che non era nostro compito eseguire un test comparativo approfondito delle tre piattaforme, e che le nostre valutazioni si limitano a quanto emerso dallo sviluppo e dall'utilizzo di Minerva su sistemi hardware paragonabili dal punto di vista dei dati di targa.

L'interfaccia di navigazione del museo è efficace e per molti versi è andata oltre le aspettative iniziali. Infatti il rigore che la contraddistingue è coniugato a un aspetto piacevole e vivace. In pratica è utilizzabile come interfaccia definitiva a meno

che non si desiderino effetti “scenici” come l’esplorazione di un ambiente tridimensionale. Questo, ribadiamo, non era un risultato in nessun modo preventivabile.

Per quanto concerne l’applicativo di gestione delle informazioni del database è stata realizzata una servlet che nasconde al gestore la logica interna del sistema e permette una semplice manutenzione del museo senza la necessità di nessuna competenza informatica particolare. L’utilità di questa servlet, denominata Minerva Admin, è fondamentale perché permette a un utente specializzato nelle tematiche relative al museo di operare per la modifica dello stesso senza l’assistenza di nessun programmatore.

Il lavoro della tesina della coppia Calcaterra-Franzosi è completato dalla documentazione del codice creata grazie allo strumento JavaDoc. È quindi facile riutilizzare parti del codice da noi scritto senza preoccuparsi dell’implementazione ma facendo riferimento alle interfacce documentate in JavaDoc. L’importanza di una buona documentazione è di primaria importanza. Senza di essa il passaggio di consegne tipico del progetto Minerva diventa difficile, perché è necessario spendere molto tempo per l’apprendimento del sistema prima di poter iniziare a lavorare sulle nuove funzionalità dello stesso.

Capitolo 6

Valutazione

“Facta non verba”

Massima latina

Un problema essenziale con cui ci siamo confrontati è stato la necessità di sviluppare un software che fosse al contempo ingegneristicamente rigoroso e in grado di soddisfare le richieste della committenza.

Dopo una riunione preliminare con il Prof. Della Torre e con i progettisti del museo vero e proprio, con i quali si è cominciato a discutere su come potesse essere dimensionata ed organizzata la parte all'interno del museo dedicata alle postazioni per l'accesso al software Minerva, abbiamo iniziato a lavorare in proprio per lo sviluppo del primo prototipo senza alcun dato reale ma solo con dati fittizi utili nella progettazione dell'architettura del sistema ed a verificare le potenzialità del progetto.

In questa prima fase ci siamo concentrati principalmente sulla stabilità del sistema nelle sue funzioni primarie (operatività del sistema ad agenti; creazione dell'agente client all'interno del web server tramite servlet; comunicazioni tra i diversi server; accessibilità del nuovo database) e sulle caratteristiche di intelligenza artificiale in senso classico. Terminata questa prima fase di sviluppo del prototipo, che si è svolta in continuo contatto con il nostro relatore Ing. Amigoni e il correlatore Dott.ssa Schiaffonati per mostrare loro i progressi ed individuare insieme a loro gli obiettivi immediati da raggiungere oltre che definire le modifiche necessarie a quanto già prodotto, ci è stato affiancato l'Arch. Andrea Bonavita con il compito specifico di supervisionare il lavoro dal punto di vista dei contenuti fornendoci materiale adatto

per il popolamento del database nella fase avanzata di testing del sistema, nonché per indirizzarci verso un'interfaccia funzionale, amichevole per l'utente e al tempo stesso significativa sotto il profilo didattico e nozionistico. La collaborazione è stata molto proficua perché ha permesso di mettere a contatto il sistema già dalle fasi iniziali di sviluppo con la figura di Bonavita svelando in questo modo le esigenze di praticità e facilità d'uso dell'utente non esperto dal punto di vista tecnico, ed allo stesso tempo ponendo sotto il suo giudizio critico la modalità di presentazione dei contenuti. Il progetto si è arricchito quindi dalla collaborazione di persone con profili culturali eterogenei aventi un obiettivo comune. La cadenza delle riunioni era circa settimanale nella fase di maggior lavoro assestandosi sulle due volte al mese circa negli altri periodi.

Nelle riunioni si faceva il punto della situazione e poi si valutavano i cambiamenti da fare per la volta successiva innescando un processo di sviluppo a spirale tipico dell'ingegneria del software.

6.1 Test Bonavita

Dopo la presentazione del primo prototipo, il Prof. Della Torre ci ha messi in contatto con un suo collaboratore, l'Arch. Bonavita, il quale ci ha di fatto affiancati nelle successive fasi di sviluppo del progetto.

La figura di Bonavita ci è apparsa da subito preziosa grazie alla sua esperienza sugli aspetti prettamente legati ai contenuti del museo virtuale, quindi in riferimento alla scelta delle opere da inserire nel database, al reperimento delle informazioni sui reperti stessi, alla compilazione delle descrizioni dei temi, all'individuazione delle possibili risorse esterne che sarebbe stato interessante approfondire tramite l'area appositamente dedicata a questo scopo. In pratica il suo supporto ha integrato il

sistema in tutto ciò che non rientrava nelle nostre competenze specifiche e negli obiettivi accademici di una tesina di ingegneria informatica.

Allo stesso tempo Bonavita rappresentava un potenziale “utente non esperto”, ossia il target d’utenza a cui si rivolgerà principalmente il museo virtuale. L’espressione “non esperto” va riferito ai lati tecnologici, pertanto riguarda tutte le questioni di programmazione, di strutturazione del database, di scelta, configurazione e messa in funzione dei componenti software, dal web server alla piattaforma ad agenti, da MySql ad Eclipse.

Nel corso delle riunioni tenute con l’Ing. Amigoni, Bonavita è stato il principale fruitore delle versioni aggiornate del sistema, apportando utili suggerimenti soprattutto in merito alle modalità di presentazione del museo virtuale ed alle caratteristiche dell’interfaccia. Con lui si è discussa l’ipotesi di affiancare, nella pagina dedicata al museo virtuale vero e proprio creato dall’utente, la descrizione dei temi scelti dall’utente alle opere individuate dal software, permettendo così al visitatore di approfondire le proprie conoscenze su argomenti comuni a vari reperti e spesso collegati a interessanti risorse esterne contestualmente alla visione dell’insieme di opere. In diverse occasioni inoltre ha dato consigli pratici sulle caratteristiche dell’interfaccia, contribuendo a raggiungere un risultato di leggerezza, immediatezza e facilità di fruizione.

6.2 Test Della Torre

Nel corso dello sviluppo di Minerva sono avvenuti alcuni incontri con il Prof. Della Torre, il quale ha rappresentato la figura di contatto con la committenza. In seguito ad una riunione di presentazione in cui ci ha mostrato gli obiettivi generali e le sue idee riguardo al progetto “Museo virtuale dell’Isola Comacina”, gli incontri,

tenutisi con frequenza circa mensile, hanno avuto inizialmente la funzione di definire le linee guida e di focalizzare i requisiti, dai quali poi abbiamo potuto intraprendere la fase di progettazione.

Com'è naturale nello sviluppo di progetti software, in principio le idee sono venute a galla poco alla volta, senza interconnessioni forti tra di loro e con un alto grado di astrattezza. Diverse riunioni con l'Ing. Amigoni, la Dott.ssa Schiaffonati ed in seguito l'Arch. Bonavita, hanno fatto sì che queste idee prendessero via via una forma più definita permettendoci infine di individuare dei requisiti con un buon grado di precisione. Esempio lampante di ciò è rappresentato dalla prima ipotesi sulle modalità di ricerca che il software avrebbe dovuto offrire al visitatore per la creazione del museo virtuale. In seguito al primo incontro con il Prof. Della Torre si era infatti meditato di condurre una selezione delle opere in base a più temi comuni ed a diversi criteri di raggruppamento. Durante la fase di progettazione però è emerso chiaramente che l'approccio a più criteri non sarebbe stato coerente con il tipo di risultato cercato e si è perciò deciso, in modo concorde, che il numero di criteri selezionabili sarebbe stato limitato ad uno.

Una volta stabilito un progetto di massima abbiamo implementato il primo prototipo che, dopo ulteriori miglioramenti studiati e valutati nel corso di alcuni incontri con il relatore, è stato mostrato a Della Torre. L'incontro è stato occasione per verificare se la strada fino ad allora intrapresa rispecchiasse le idee che ci erano state precedentemente comunicate. Nonostante la relativa rudimentalità di questa versione, poter lavorare su un software funzionante da analizzare e mettere alla prova è stato di concreto aiuto nell'avvicinare gli interlocutori e come punto di partenza per il successivo raffinamento degli obiettivi. Con una certa soddisfazione generale si è constatato che quanto espresso dal prototipo non si discostava in maniera sensibile

dagli interessi della committenza divenendo così un buon punto di partenza per la realizzazione del futuro.

Confermata la correttezza generale del lavoro fino ad allora svolto abbiamo portato avanti lo sviluppo del progetto concentrandoci su aspetti quali i collegamenti alle risorse esterne, la disposizione degli oggetti all'interno della pagina del museo virtuale, la modalità di presentazione e quindi le caratteristiche dell'interfaccia ed il pannello di amministrazione per la gestione del database del museo virtuale. Quest'ultimo aspetto in particolare, per quanto non se ne fosse parlato in maniera approfondita con il Prof. Della Torre e non influisca in nessun modo sulla validità del prodotto agli occhi del visitatore, ha rappresentato a nostro avviso un lato irrinunciabile per la completezza generale del progetto, soprattutto in relazione alla reale applicazione di questa versione di Minerva all'interno del museo di Ossuccio dedicato all'isola Comacina in cui una figura probabilmente non esperta di programmazione e/o di tecnologie di database dovrà occuparsi della manutenzione dei contenuti.

Raggiunta una ragionevole maturità del prodotto abbiamo concordato un incontro dimostrativo con Della Torre, nel corso del quale gli abbiamo esposto in maniera approfondita tutti gli aspetti del nostro software mostrandoglielo all'opera e permettendogli di interagire liberamente. Il professore si è dimostrato soddisfatto del risultato raggiunto, apprezzando le soluzioni da noi impiegate per gli obiettivi prefissati e anche per quegli aspetti di cui magari si era discusso meno approfonditamente, quali l'organizzazione spaziale delle pagine del museo virtuale e il pannello di amministrazione.

Da non sottovalutare nel corso di queste riunioni sono stati gli interventi di Bonavita il quale, grazie alla sua area di pertinenza affine a quella di Della Torre, ha spesso dimostrato di saper bene interpretare il suo pensiero aiutandoci e, in un certo senso ponendosi da intermediario, a comprendere meglio le sue valutazioni e le sue idee.

6.3 Prove di funzionamento

In questo paragrafo vengono mostrate alcune prove di funzionamento del sistema nella sua versione finale. La Figura 6.1 mostra la pagina dedicata ad uno specifico reperto, in questo caso il Capitello Bizantino. Dall'alto verso il basso troviamo:

- il titolo “Museo virtuale dell'isola Comacina”, ricorrente in tutte le sezioni del programma;
- il titolo del reperto ed una sua fotografia, cliccabile per visualizzarlo in formato ingrandito ed apprezzarne i dettagli estetici;
- la descrizione del reperto, con dettagli riguardanti il periodo a cui risale, il materiale impiegato per la sua realizzazione ed in alcuni casi la storia che ha vissuto (trasferimenti, rifacimenti, ecc.);
- l'area dedicata alla creazione del museo virtuale a partire dal reperto correntemente visualizzato. Quest'area è logicamente suddivisa in due sottoaree di selezione:
 - la prima è quella dedicata ai temi: qui vengono visualizzati i temi correlati al reperto che possono essere selezionati, in numero variabile, per l'individuazione dei reperti che andranno a far parte del museo virtuale. Solo

- i reperti che condividono i temi selezionati saranno inseriti nelle stanze del museo virtuale e mostrati di conseguenza all'utente;
- la seconda è invece dedicata ai criteri: in questo caso i criteri non dipendono dal particolare oggetto preso in considerazione e la scelta possibile è singola. I criteri tra cui si può attualmente scegliere sono “cronologico”, “topografico”, “tipologia di reperto” e “modalità di rinvenimento”;
 - infine sono mostrati i due bottoni “Crea museo”, nel caso si decida di creare il museo a partire dai temi e dal criterio scelti, e “Torna indietro”, nel caso si voglia invece tornare indietro al museo virtuale precedentemente realizzato.

Museo virtuale dell'isola Comacina

Capitello Bizantino



Capitello di derivazione corinzia, realizzato in marmo di Musso e databile tra il VI e il VII secolo, è l'unico rinvenuto nella cripta della chiesa di Sant'Eufemia sull'Isola Comacina. Si compone di un calato svasato fasciato da quattro foglie d'acanto spinoso che si piegano sotto i risalti angolari della parte superiore; su ciascuna delle facce del capitello fuoriescono dalla corolla due caulicoli che si biforcano in due volute contrapposte, le maggiori avvolte sullo sporto angolare, le minori sullo sporto mediano, che risulta piuttosto accentuato e presenta al centro una linguetta puntuta.

Temi Correlati

- ☒ Architettura romanica e colore: tracce preziose e nascoste
- ☐ Il marmo di Musso
- ☐ Il mito dei magistri comacini
- ☒ Il sito archeologico di S. Eufemia
- ☐ L'Isola Comacina come ultimo baluardo dei Bizantini
- ☒ L'utilizzo di materiale di reimpiego in costruzioni più tarde
- ☐ Modelli e caratteri formali dei capitelli bizantini

Criteri

- ☐ cronologico
- ☐ topografico
- ☒ tipologia di reperto
- ☐ modalità di rinvenimento

Torna indietro

Crea museo

Figura 6.1: Museo virtuale dell'isola Comacina, scheda del reperto “Capitello Bizantino”

Passando al museo virtuale vero e proprio invece se ne può apprezzare una schermata in Figura 6.2, in cui si vede la struttura e l'aspetto della pagina ad esso

dedicata. Come descritto nel paragrafo 4.5.3 sono state individuate tre aree: in verde quella dedicata al reperto da cui è stato realizzato il museo virtuale; in arancione le descrizioni dei temi selezionati; in giallo le stanze del museo virtuale create in base al criterio indicato in testa alla sezione. In particolare è possibile vedere come sia stata implementata un'interfaccia a scomparsa per la descrizione dell'oggetto, per la parte dedicata alla creazione di un nuovo museo e per le descrizioni dei temi, in modo da non mostrare all'utente troppo materiale contemporaneamente e permettergli di espandere le aree di suo interesse.



Figura 6.2: Museo virtuale dell'isola Comacina, la pagina del museo virtuale

Nella zona in giallo le stanze (stilizzate) sono rappresentate dai riquadri più scuri in cui come titolo è mostrato il valore del criterio di raggruppamento a cui corrispondono gli oggetti contenuti. I riquadri grigi interni alle stanze rappresentano le teche, che potranno contenere uno o più oggetti a seconda dell'esito della ricerca dei reperti. Infine si notano i collegamenti (testo evidenziato in blu) presenti nella descrizione del terzo tema, mostrato espanso, che permettono di accedere alle relative pagine delle risorse esterne, di cui viene proposta una schermata in Figura 6.3.



Figura 6.3: Museo virtuale dell'isola Comacina, pagina delle risorse esterne

In questa figura si nota come la pagina sia divisa orizzontalmente in due parti in cui quella superiore contiene la descrizione della risorsa esterna inserita nel database di Minerva, con una foto ed i link ai siti Internet esterni, mentre in quella

sottostante è possibile consultare direttamente uno dei siti esterni scegliendo tra quelli proposti. Tramite il collegamento “Nascondi descrizione” posto accanto al titolo è possibile minimizzare le dimensioni della barra superiore in modo tale da avere ampio spazio sullo schermo per consultare il sito esterno (Figura 6.4).



Figura 6.4: Museo virtuale dell'isola Comacina, pagina delle risorse esterne, descrizione nascosta

Infine in Figura 6.5 viene mostrata una schermata del pannello di amministrazione, in particolare si tratta della pagina di modifica dei dati relativi ad un item che permette di cambiarne il nome, la descrizione, la priorità, i temi associati, i valori dei criteri di raggruppamento e le foto collegate. In particolare per i criteri è possibile inserire un valore scegliendolo tra quelli già presenti nel database tramite un comodo menu a tendina oppure introdurre un nuovo valore.

Modifica item

Nome dell'item
Capitello Bizantino

Descrizione item
Capitello di derivazione corinzia, realizzato in marmo di Musso e databile tra il VI e il VII secolo, è l'unico rinvenuto nella cripta della chiesa di Sant'Eufemia sull'Isola Comacina. Si compone di un calato svasato fasciato da quattro foglie d'acanto spinoso che si piegano sotto i risalti angolari della parte superiore; su ciascuna delle facce del capitello fuoriescono dalla corolla due

Priorita' 0

Scegli i temi relativi all'item

- ☒ Architettura romanica e colore: tracce preziose e nascoste
- ☒ Il marmo di Musso
- ☒ Il mito dei magistri comacini
- ☒ Il sito archeologico di S. Eufemia
- ☒ L'Isola Comacina come ultimo baluardo dei Bizantini
- ☒ L'utilizzo di materiale di reimpiego in costruzioni più tarde
- ☐ Le testimonianze romane
- ☒ Modelli e caratteri formali dei capitelli bizantini

Scegli i criteri relativi all'item

- cronologico**
 - ☒ Scegli un valore esistente VI-VII secolo d.C.
 - ☐ Inserisci un nuovo valore
- topografico**
 - ☐ Scegli un valore esistente Isola Comacina
 - ☒ Inserisci un nuovo valore
- tipologia di reperto**
 - ☒ Scegli un valore esistente Elemento architettonico
 - ☐ Inserisci un nuovo valore
- modalità di rinvenimento**
 - ☒ Scegli un valore esistente Rilevamento visivo in loco
 - ☐ Inserisci un nuovo valore

Foto collegate all'item

☐ elimina foto

Aggiungi foto
Seleziona il numero di foto che vuoi inserire: 1
 Sfoglia...

[Modifica item](#)

Figura 6.5: Museo virtuale dell'isola Comacina, pannello di amministrazione, pagina di modifica dati item

Capitolo 7

Sviluppi futuri e conclusioni

“Audaces fortuna iuvat”

Tibullo

In questo capitolo l’attenzione sarà concentrata sulle possibilità di sviluppi ulteriori di questa versione di Minerva e sulle conclusioni finali. L’obiettivo che ci ha guidati fin dall’inizio è stato il soddisfacimento delle richieste della committenza, pertanto in alcuni casi non è stato necessario approfondire aspetti tecnologici oltre le esigenze pratiche di progetto, aspetti che invece potranno essere oggetto di studio da parte di lavori futuri.

Nei prossimi paragrafi passiamo in rassegna i possibili sviluppi nei vari ambiti che ci sembrano interessanti per possibili sviluppi. Sono stati individuati i seguenti argomenti: ambiente 3D di navigazione; allestitore; allocazione di stanze e teche; modello MVC; accesso concorrente via web; sistema distribuito; generalizzazione del progetto.

L’ultimo paragrafo viene quindi dedicato alle conclusioni della presente tesina.

7.1 Ambiente 3D di navigazione

Lo sviluppo del progetto Minerva nell’ambito di una tesina di intelligenza artificiale si è concentrato principalmente sulla logica di gestione del museo e sui criteri per realizzare un museo a misura d’utente. A questo è stato quindi principalmente dedicato il nostro impegno il cui frutto viene presentato tramite una veste grafica leggera e funzionale, basata su tradizionali pagine web generate

dinamicamente, che lascia pertanto margini di sviluppo nel caso si ricercasse qualcosa a maggiore impatto visivo o che impieghi particolari tecnologie di presentazione. Nel nostro lavoro, come richiesto dalla committenza, l'accento è stato maggiormente posto sull'ampiezza e sulla qualità di scelta delle informazioni che il sistema fornisce all'utente rispetto alla spettacolarizzazione dell'"esperienza virtuale", focalizzando l'attenzione soprattutto sui contenuti e sulla coerenza del sistema. Questo non implica che in futuro non possa essere sviluppato un sistema grafico sovrapposto che, partendo dalle informazioni contenute nelle pagine web, fornisca un museo virtuale tridimensionale a forte impatto visivo.

Per ottenere questo risultato abbiamo individuato principalmente due strade percorribili: l'uso del VRML, come già in precedenza è stato fatto nel progetto Minerva, o l'impiego delle librerie Java 3D [27]. La prima scelta è a nostro avviso poco percorribile perchè introdurrebbe un notevole ritardo nella navigazione del museo, dovuto alla frequente rigenerazione della ricostruzione tridimensionale legata al nostro progetto. Infatti una differenza sostanziale rispetto al progetto Minerva originale è che quest'ultimo creava musei statici e non modificabili in tempo reale dall'utente. Poiché sfruttava un tradizionale approccio lineare in cui da un punto iniziale, con la selezione di opere e criteri, si giungeva ad un risultato finale rappresentato dal museo virtuale tridimensionale all'interno del quale il visitatore poteva muoversi e vedere le opere esposte. Il processo non prevedeva ulteriori step, mentre lasciava naturalmente la possibilità di ricominciare dall'inizio per creare musei virtuali differenti. Come già visto in precedenza invece la presente versione di Minerva mette a disposizione un approccio differente in cui il visitatore può continuamente creare nuove collezioni a partire da quelle precedenti e in cui i momenti di generazione tridimensionali sarebbero molto più ricorrenti, incidendo in

maniera non trascurabile sui tempi d'attesa.

L'alternativa al VRML potrebbe essere l'impiego delle librerie grafiche Java 3D. Ciò è interessante da un punto di vista ingegneristico dato che si integrerebbe perfettamente nel progetto interamente sviluppato in Java, rendendo quindi il tutto più omogeneo. Tuttavia il problema legato alle prestazioni individuato nel caso di VRML potrebbe rimanere sostanzialmente immutato, a causa dell'elevata frequenza delle fasi di rendering.

Infine è ipotizzabile un'eventuale terza via in cui l'effetto 3D venga simulato utilizzando il linguaggio Macromedia Flash [28]. Questa soluzione potrebbe forse rivelarsi come quella dal miglior rapporto prezzo/prestazioni anche se non è sicuramente la più rigorosa dal punto di vista ingegneristico. Nel momento in cui si decidesse di sviluppare una veste grafica tridimensionale andrebbero senz'altro studiati attentamente i requisiti di qualità e performance del risultato e le caratteristiche delle soluzioni precedentemente indicate, individuando quindi la tecnologia più indicata al caso specifico.

7.2 Allestitore

Durante la reingegnerizzazione di Minerva alcuni agenti non sono stati più utilizzati perché, come già detto, la committenza ha avanzato richieste diverse da quelle dei precedenti progetti. Questo però non toglie che in futuro sia possibile reintrodurre alcuni agenti per migliorare e sviluppare il progetto. L'agente allestitore potrebbe avere un nuovo ruolo nel momento in cui si voglia applicare Minerva a un progetto più ampio che preveda una più raffinata operazione di raggruppamento delle opere per la quale potrebbe essere necessario ricorrere alla generazione di un albero tassonomico, ossia una struttura dati adatta a rappresentare chiaramente il risultato del

processo di allestimento. In questo modo sarebbe possibile suddividere l'intera collezione in vari gruppi in base a diversi criteri ed inoltre tramite l'ordine dei gruppi all'interno dell'albero tassonomico verrebbe rappresentato anche l'ordine di inserimento delle opere all'interno del museo virtuale.

L'albero tassonomico tuttavia, pur rappresentando il mezzo ideale per presentare all'utente l'allestimento, risulta complesso da memorizzare all'interno di un database relazionale, a causa delle proprietà strutturali dell'albero. L'agente allestitore interverrebbe a questo punto per tradurre l'albero tassonomico in una struttura dati detta *teoria decidibile*, rappresentata mediante una base di conoscenza implementata in linguaggio CLIPS [29]. Per l'interfacciamento tra Java e CLIPS sarebbe quindi possibile ricorrere al motore JESS [30].

Va comunque tenuto conto che queste fasi aggiuntive richiederebbero un continuo scambio di dati con relativi ritardi di comunicazione. Bisogna pertanto valutare attentamente l'esigenza di ricorrere a questo approccio nel caso non si tratti effettivamente di compiere operazioni di allestimento molto complesse.

7.3 Allocazione di stanze e teche

Attualmente le dimensioni delle stanze e delle teche generate da Minerva dipendono solo dal numero di oggetti presenti al loro interno e non dalle effettive dimensioni delle opere in esse contenute, né da particolari vincoli spaziali. Ciò è coerente con il modello di fruizione richiesto dalla committenza che non richiedeva alcuna correlazione delle stanze virtuali con ambienti reali, in quanto il museo rimarrà unicamente virtuale. Nel caso sorgesse la necessità di rispettare limiti concreti di ingombro andranno necessariamente introdotte modifiche che prevedano uno studio accurato della fase di allocazione.

Sarebbe innanzitutto necessario pensare ad una logica di gestione degli spazi, intendendo con questo quali debbano essere i vincoli di disposizione delle opere all'interno delle stanze in relazione alla presenza di porte, finestre, punti luce e quant'altro. Con altrettanta attenzione andrebbe poi affrontato il problema del rispetto dei parametri fisici di ingombro delle superfici e dei volumi delle stanze. Per fare ciò però diverrebbe anche necessario apportare consistenti modifiche al database in modo tale che possano essere memorizzati i dati volumetrici di ambienti e oggetti, oltre ai dati concettuali sui quali abbiamo lavorato per questa versione di Minerva.

Nell'ottica di realizzazione di questo tipo di modifica riteniamo che tornerebbe utile il valore di priorità che è stato associato ad ogni oggetto inserito nel database, in quanto un problema ricorrente nei musei è quello dell'impossibilità, per mancanza di spazio, di esporre l'intera collezione posseduta, rendendo inevitabile una scelta tra i pezzi da esporre. Attraverso la priorità si potrebbe andare a regolare in modo fine l'importanza da attribuire a ciascun oggetto, allo scopo di disporre di un ulteriore criterio per decidere se un oggetto meriti o meno di occupare parte delle limitate risorse spaziali del museo.

Per produrre risultati davvero soddisfacenti inoltre il processo di allocazione dovrebbe tenere necessariamente conto del risultato del processo di allestimento, in quanto oltre all'ottimizzazione dell'ingombro degli spazi la disposizione degli oggetti va fatta in base ad una certa logica artistica. In questa fase rientrerebbero in gioco l'esperienza ed il gusto artistico del curatore del museo, il quale potrà dare le direttive al sistema su decisioni altrimenti difficili da gestire in maniera totalmente automatizzata, se non a scapito della versatilità e della configurabilità dell'allocazione.

Questi argomenti sono stati studiati attentamente da Ghidotti nel suo lavoro di

tesi [1] (il quale peraltro si è avvalso dei risultati ottenuti nelle precedenti versioni di Minerva anche in riferimento a questi aspetti), pertanto Minerva05 rappresenterebbe senz'altro il punto di riferimento per l'implementazione di questa funzionalità.

7.4 JSP, Servlet e il modello MVC

Parlando del web server Tomcat abbiamo visto come all'interno di esso siano dei particolari componenti, le servlet, ad occuparsi della comunicazione con il Minerva server e a gestire la costruzione dinamica delle pagine html destinate al browser del visitatore. Questo approccio, apparentemente semplice e funzionale, incorpora un difetto di forma, ossia la mancanza di separazione tra presentazione e contenuti. Per realizzare una ben definita separazione tra presentazione e contenuti è necessario ricorrere all'utilizzo delle Java Server Page (jsp), le quali possono essere facilmente scritte e gestite attraverso tool html e xml [31] convenzionali e ampiamente diffusi da chi si occupa fundamentalmente del lato grafico (la presentazione), ossia i *designer*, ai quali non viene quindi chiesto di operare con i lati più complessi dell'applicazione, le servlet, che vengono invece tipicamente gestiti dagli sviluppatori del lato server che possono vantare nozioni di programmazione molto più approfondite. D'altra parte jsp e servlet non rappresentano mondi separati in quanto le prime sono astrazioni di alto livello delle seconde.

7.4.1 Il modello Model-View-Controller

Il modello MVC, mostrato in Figura 7.1, è un approccio ibrido per servire contenuti dinamici, in quanto fa ricorso contemporaneamente all'utilizzo di jsp, servlet e di un terzo componente chiamato JavaBean.

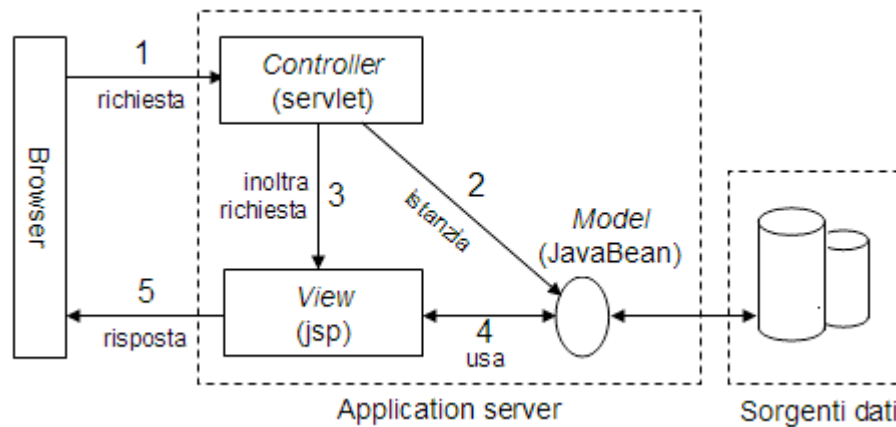


Figura 7.1: il modello Model-View-Controller

In questo caso si sfruttano i punti di forza di entrambe le tecnologie, jsp e servlet, utilizzando jsp per generare lo strato di presentazione e servlet per eseguire compiti intensivi di processo. La servlet agisce da *controller*, ed è incaricata di processare le richieste degli utenti. Di volta in volta, in base alla richiesta, crea un'istanza di un bean (il *model*, un componente riutilizzabile, quale potrebbe essere nel nostro caso un item) che verrà poi utilizzato dalla jsp (la *view*) a cui la servlet decide di inoltrare la richiesta per recuperare i dati elaborati dagli altri server (in questo caso Minerva e MySQL). Bisogna notare che non c'è logica di processo all'interno della pagina jsp; ad essa viene unicamente affidato il compito di recuperare gli oggetti o i bean precedentemente creati dalla servlet, e di estrarne il contenuto dinamico per inserirlo in un template statico. In questo modo viene realizzata una totale separazione tra presentazione e contenuti, con una chiara definizione dei ruoli e delle responsabilità di designer e programmatori che potranno lavorare con una certa indipendenza. Inoltre, maggiore sarà la complessità dell'applicazione, maggiori dovrebbero risultare i benefici dell'implementazione di questo modello.

7.5 Accesso concorrente via web

L'attuale Minerva è pensato per essere eseguito da utenti singoli o al più da un piccolo numero di utenti. Il vero limite per un accesso condiviso di massa è la gestione della servlet che non implementa in modo impeccabile la gestione della concorrenza. Nella fase di comunicazione tra il server web e Minerva va ancora perfezionata la gestione delle sessioni, in modo tale da riuscire a tenere traccia delle diverse destinazioni delle informazioni durante l'intero percorso che va dal web server a Minerva e da questo indietro attraverso Tomcat fino all'utente. Per ovviare a questo inconveniente attualmente la servlet effettua operazioni bloccanti, per tenere separati i flussi informativi, e quindi, in caso di intenso traffico di utenti, rende i tempi di risposta non accettabili.

Per quanto riguarda il server Minerva invece la concorrenza è gestita in modo sicuro e permette anche un elevato traffico a patto di avere delle sufficienti risorse hardware. Altrettanto affidabili da questo punto di vista sono i server MySQL e Tomcat, già ampiamente utilizzati in ambito web e compatibili con un elevato numero di accessi contemporanei.

7.6 Sistema centralizzato e distribuito

L'implementazione attuale di Minerva prevede che ogni postazione sia autosufficiente nel compito di fornire l'accesso al museo virtuale ai visitatori. Ciò significa che su ogni postazione devono essere presenti tutti i componenti software di funzionamento del sistema, ossia il web server Tomcat con l'agente client, gli agenti lato server all'interno dell'ambiente JADE ed il database server MySQL, oltre ovviamente al browser.

Un'implementazione più efficiente del sistema comporterebbe la

centralizzazione di questi componenti server, lasciando alle varie postazioni unicamente il ruolo di punto di accesso al sistema e riducendone di conseguenza le risorse hardware necessarie, poichè su di esse sarebbe necessario lasciare unicamente il web browser. Un altro vantaggio non secondario sarebbe la manutenzione ampiamente semplificata del sistema, in quanto eventuali modifiche (riguardanti ad esempio l'interfaccia grafica, ma si può anche pensare a inserimenti di nuovi oggetti nel database o a variazioni dei parametri di quelli già inseriti) andrebbero effettuate unicamente su una macchina e non ripetute per tutte le postazioni.

Naturalmente questa soluzione richiederebbe il perfezionamento della gestione dell'accesso concorrente di cui si è trattato nel paragrafo precedente, in quanto molti utenti potrebbero trovarsi ad accedere contemporaneamente a pochi sistemi server. Una volta apportate le adeguate modifiche alla gestione della concorrenza non vi sarebbero ulteriori difficoltà nell'implementazione della centralizzazione del sistema, per la quale è prevedibile una struttura distribuita su tre server:

- sul primo verrebbe posto il web server con le relative servlet, in grado di gestire le richieste di accesso provenienti sia dal museo stesso che da qualsiasi altro host su Internet;
- il secondo sarebbe invece dedicato al sistema ad agenti di Minerva, con l'agente server in ascolto delle richieste provenienti dal web server;
- infine il terzo sarebbe il database server, incaricato di recuperare i dati sugli oggetti memorizzati localmente.

I server verrebbero messi in comunicazione tra di loro tramite una rete LAN Ethernet di classe Gigabit, tecnologia ormai ampiamente diffusa e che grazie ad elevati valori di throughput eliminerebbe il rischio di colli di bottiglia legati alle comunicazioni tra i vari computer. Per motivi di sicurezza le postazioni all'interno

del museo verrebbero collegate tramite un'altra rete locale al web server ed infine quest'ultimo verrebbe collegato a Internet per consentire l'accesso agli utenti remoti.

Per limitare il numero di computer utilizzati è anche possibile far convivere su di un solo elaboratore sia Minerva sia MySQL, mentre è bene che Tomcat risieda da solo sulla macchina esposta su Internet per minimizzare i possibili danni dovuti ad attacchi esterni. In caso il numero di accessi dall'esterno diventi considerevole è inoltre possibile pensare a un sistema più complesso in cui vengano replicati i singoli server, gestito da un'applicazione aggiuntiva per bilanciare il carico di lavoro tra di essi.

È da ricordare infine, nel caso di sistema distribuito, la possibilità di impiegare computer con sistemi operativi diversi in base alle migliori performance, ad esempio il server Minerva potrebbe utilizzare Mac Os X e invece MySQL potrebbe essere eseguito su architettura Linux.

7.7 Generalizzazione del progetto

Minerva è stato concepito come un sistema flessibile e generale fin dal principio. È infatti possibile applicare il software a vari musei modificando solo un numero relativamente modesto di linee di codice. Una possibile modifica evolutiva potrebbe essere quella di espandere la servlet MinervaAdmin in modo tale che possa modificare anche la pagina iniziale del museo, tutte le etichette di testo ed anche la grafica attraverso la modifica del *foglio di stile* CSS [32]. Sarebbe poi necessario modificare anche gli agenti DBManager e client in modo tale che queste informazioni vengano acquisite in modo dinamico. In tale maniera sarebbe possibile cambiare il museo virtuale semplicemente sostituendo il database senza fare nessuna modifica al codice sorgente.

La caratteristica innovativa di questa versione di Minerva rispetto alle versioni precedenti è legata all'essere divenuto uno strumento non solo per la creazione di musei virtuali ma anche un vero e proprio “gestore di conoscenza”.

Con il termine gestore di conoscenza intendiamo un sistema in grado di organizzare una serie d'informazioni, selezionate e catalogate da personale esperto, allo scopo di poter essere esplorate da un utente non esperto in modo proficuo attraverso percorsi creati dall'utente stesso in base ai propri interessi, in modo dinamico ed in tempo reale.

Questa caratteristica di Minerva deriva dall'astrazione con cui il sistema è progettato fin dal database. Infatti il concetto base utilizzato è quello di “item”, per sua natura astratto, e lo stesso possiamo dire per il concetto di “tema” che non è altro che un insieme di item aventi una particolare caratteristica in comune.

Un esempio pratico potrebbe essere la realizzazione di un sistema a scopo didattico per la divulgazione di informazioni riguardo varie patologie legate a errati stili di vita. In questo caso gli stili di vita rappresenterebbero i temi e le patologie gli item.

7.8 Compatibilità delle linee di sviluppo

Non tutte le linee di sviluppo che abbiamo qui delineato sono compatibili tra di loro, ad esempio se si introduce un ambiente tridimensionale la generalità del sistema viene meno. Ciò però non è un problema perchè, come insegna l'ingegneria del software, un software si può evolvere in modi diversi e può anche cambiare direzione di sviluppo in base alle necessità. È importante mantenere per quanto possibile tutte le vie aperte in modo tale da non dover reinventare ogni volta la ruota che è in fondo la vera lezione della moderna progettazione del software.

7.9 Conclusioni

L'attività di questa tesina era finalizzata all'analisi del precedente software Minerva e alla definizione delle modifiche da apportare ad esso per la realizzazione del museo virtuale dell'isola Comacina. In particolare erano da prendere in considerazione le esigenze della committenza, in parte differenti da quelle emerse nei precedenti sviluppi del progetto stesso. La metodologia seguita nell'analisi e nello sviluppo del software è stata ciclica, cadenzata da una serie di riunioni con l'Ing Amigoni, il Prof. Della Torre, la Dott.ssa Schiaffonati e l'Arch. Bonavita.

I nuovi concetti introdotti sono stati la definizione di tema, di risorsa esterna ed il ripensamento del concetto di stanza e teca. Lo scopo era quello di permettere all'utente un vero e proprio percorso conoscitivo interamente interattivo a partire dai reperti esposti nel museo reale. Era importante espandere i confini delle informazioni fruibili rispetto a quelle contenute nei reperti del museo, creando quindi un'esperienza più ampia possibile.

È stata anche in parte rivista l'architettura del server Minerva a partire dal database, per il quale è stato utilizzato MySQL, per arrivare fino alla gestione degli agenti ed al controllo della loro disponibilità tramite l'introduzione di un nuovo agente denominato Check Agent.

Infine è stata elaborata l'architettura Client basata sul server Tomcat e sulla tecnologia servlet che consente di utilizzare un semplice browser per l'esplorazione del museo. I requisiti e le soluzioni trovate sono poi state implementate nella tesina Calcaterra-Franzosi.

Il risultato delle due tesine è stato una versione funzionante e molto soddisfacente del museo virtuale dell'isola Comacina, come sottolineato dalla committenza.

Capitolo 8

Bibliografia

Tesi di laurea

- [1] F. Ghidotti, *Minerva: un sistema multiagente per l'allestimento dei musei archeologici*, tesi di laurea A.A. 2003-2004, Politecnico di Milano.

Musei

- [2] Guggenheim Museum, sede di New York, US, ultimo accesso Dicembre 2005,
http://www.guggenheim.org/new_york_index.shtml
- [3] The National Gallery, London, UK, ultimo accesso Dicembre 2005,
<http://www.nationalgallery.org.uk>
- [4] The State Hermitage Museum, San Pietroburgo, Russia, ultimo accesso
Dicembre 2005, <http://www.hermitagemuseum.org>
- [5] Centre Pompidou, Paris, France, ultimo accesso Dicembre 2005,
<http://www.cnac-gp.fr/Pompidou/Accueil.nsf>
- [6] Musée du Louvre, Paris, France, ultimo accesso Dicembre 2005,
<http://www.louvre.fr/louvrea.htm>

Musei virtuali

- [7] Cardiac Virtual Museum, ultimo accesso Dicembre 2005,
<http://arrhythmia.hofstra.edu/vrml/museumn/museumn.html>
- [8] Virtual Museum of Canada, ultimo accesso Dicembre 2005,
http://www.virtualmuseum.ca/English/index_flash.html

Tecnologie

- [9] Java Agent DEvelopment Framework, ultimo accesso Dicembre 2005,
<http://jade.tilab.com>
- [10] Jakarta Tomcat, ultimo accesso Dicembre 2005,
<http://tomcat.apache.org>
- [11] Microsoft Access, ultimo accesso Dicembre 2005,
<http://office.microsoft.com>
- [12] MySql AB, ultimo accesso Dicembre 2005,
<http://www.mysql.com>
- [13] Open Source OSI, ultimo accesso Dicembre 2005,
<http://www.opensource.org>
- [14] FIPA, “The Foundation for Intelligent Physical Agents”, ultimo accesso Dicembre 2005, <http://www.fipa.org>
- [15] QTVR, “Quick Time Virtual Reality”, ultimo accesso Dicembre 2005,
<http://www.apple.com/quicktime/overview/qtvr.html>
- [16] VRML, “Virtual Reality Modeling Language”, ultimo accesso Dicembre 2005,
<http://www.web3d.org>
- [17] Structured query language, ultimo accesso Dicembre 2005,
<http://www.sql.org>
- [19] Microsoft Windows, ultimo accesso Dicembre 2005,
<http://www.microsoft.com>
- [20] Linux, ultimo accesso Dicembre 2005,
<http://www.kernel.org>
- [21] Mac OS X, ultimo accesso Dicembre 2005,
<http://www.apple.com/macosx>

-
- [22] Apache Foundation, ultimo accesso Dicembre 2005,
<http://www.apache.org>
- [23] Unified Modeling Language, ultimo accesso Dicembre 2005,
<http://www.uml.org>
- [24] Eclipse project, ultimo accesso Dicembre 2005,
<http://www.eclipse.org>
- [25] Apache Ant, ultimo accesso Dicembre 2005,
<http://ant.apache.org>
- [26] Subversion, ultimo accesso Dicembre 2005,
<http://subversion.tigris.org>
- [27] Java 3D, ultimo accesso Dicembre 2005,
<http://java.sun.com/products/java-media/3D>
- [28] Macromedia Flash, ultimo accesso Dicembre 2005,
<http://www.macromedia.com>
- [29] CLIPS, ultimo accesso Dicembre 2005,
<http://www.siliconvalleyone.com/clips.htm>
- [30] JESS, ultimo accesso Dicembre 2005,
<http://herzberg.ca.sandia.gov/jess>
- [31] Extensible Markup Language, ultimo accesso Dicembre 2005,
<http://www.w3.org/XML>
- [32] Cascading Style Sheets, ultimo accesso Dicembre 2005,
<http://www.w3.org/Style/CSS>

Testi

- [33] M. J. Wooldridge, N. R. Jennings (eds), "Intelligent Agents: Theory and Practice", *The Knowledge Engineering Review*, 10 (2), pp. 115-152, 1995.