

proposte di progetto

a.a. 2003-2004

interpretare le proposte

- vengono date delle tracce su cui discutere
- è compito dei gruppi raffinare la proposta o modificarla durante la fase di inception
 - un buon progetto dovrebbe essere
 - innovativo (rispetto a ciò che già c'è in giro)
 - interessante (per le problematiche di analisi/progetto che pone)
 - il dominio non è eccessivamente ampio (altrimenti l'analisi prende troppo tempo)
 - almeno uno dei componenti del gruppo (compreso il manager) ha un interesse specifico nello sviluppo (e fa anche da "customer")

resac: contabilità per un gruppo di ricerca

Un gruppo di ricerca deve gestire i fondi di ricerca e le spese effettuate con tali fondi. Spesso i fondi di ricerca sono associati ad un progetto che richiede la rendicontazione delle spese. Il sistema dove gestire fondi, impegni, contratti, rendicontazioni, ecc. con una certa flessibilità rispetto a situazioni incongrue tra procedure amministrative e procedure interne al gruppo.

Dominio tecnico e da chiarire. Aspetti architetturali standard, probabilmente distribuito. Usufruibile via web? portabilità, dati critici. testing problematico perché sistema via web. Esperienza in uno dei domini più importanti nelle applicazioni informatiche.

exam: gestione compiti d'esame

Un gruppo di docenti cooperano alla redazione dei compiti per gli esami di un certo corso. Il testo dei compiti è composto da vari esercizi. La redazione di ciascun esercizio può essere da zero o come modifica di un esercizio precedentemente assegnato. Poiché i compiti d'esame non devono differire tra di loro è importante tenere traccia della "storia" delle modifiche effettuate a ciascun esercizio. E quali varianti sono state assegnate ed in quali appelli. Il sistema non deve imporre vincoli sul formato in cui gli esercizi sono redatti (tipicamente word o latex ma potrebbero cambiare in futuro) e dovrebbe permettere di effettuare ricerche di esercizi in base al testo dell'esercizio.

Dominio abbastanza chiaro ma potrebbe avere risvolti interessanti. Requisiti non funzionali: semplicità d'uso, compatibilità word e latex, portabilità dei dati, sicurezza contro flessibilità di utilizzo da più utenti. Aspetti architetturali: standard e compatibilità tra formati

seherazade: vendita di racconti on-line

Si considera il seguente business on-line riguardante la vendita di racconti. Il cliente puo' leggere l'inizio di una storia ma per leggere il finale deve acquistarlo. Per ciascun racconto parte del ricavato va all'autore del racconto. Il comitato di redazione ha il compito di scegliere i racconti inviati dagli utenti e di deciderne o meno la pubblicazione per la vendita. Inoltre puo' inserire tra le proposte racconti difficili da trovare su cui non gravano piu' i diritti d'autore. Ciascun cliente ha un conto aperto e paga i base ai racconti scelti. Il sistema informatico dovrebbe gestire tutti gli aspetti sia quelli di distribuzione dei racconti e dei finali, quelli di gestione del comitato editoriale sia quelli di billing. Poiche' il modello di business potrebbe cambiare il sistema dovrebbe avere un progetto flessibile rispetto a tutte le caratteristiche di questo per poter essere adattato velocemente.

Dominio interessante: front office vs. back office. requisiti non funzionali: facilmente modificabile al variare del modello di business, usabilità lato utente. Aspetti architetturali: standard

palmgroup: un workgroup basato su palmari

In un gruppo di lavoro un certo numero di informazioni devono essere condivise e decisioni devono essere prese collettivamente ma non sempre i membri del gruppo possono essere fisicamente presenti o raggiungibili. Mediante l'uso di palmari ed eventualmente di un server centrale si vuole agevolare la procedura per stabilire meeting, condividere informazioni, modificare documenti di lavoro in genere. Una persona puo' appartenere a vari gruppi di lavoro e interagire con il sistema in vari modi, palmare, web, email. Il sistema dovrebbe essere non essere legato a tecnologie specifiche. Gli aspetti di software integration potrebbero essere non banali.

Dominio vasto e generale. Requisiti non funzionali: portabilità lato client. Aspetti architetturali: sviluppo su palmare. Richiede un paio di palmari.

goodom: parsing di documenti xml errati

Il DOM e' un utile modello supportato dalla W3C per rappresentare e manipolare i contenuti di documenti XML "ben-formati" (well-formed). Purtroppo molte pagine html disponibili sul web non sono direttamente rappresentabili secondo tale modello perche' non sono ben-formate. Si vuole sviluppare un'applicazione (o una libreria?) per creare rappresentazioni DOM di un documento html non ben formato minimizzando il numero di modifiche sul documento necessarie per renderlo ben-formato.

Dominio tecnico, il concetto di rappresentazione di un documento non ben formato deve essere raffinato.

Aspetti architetturali: standard. applicazioni possono essere pretty printer o wrappers.

automail: classificatore automatico di email

Molte persone ricevono una gran quantita' di email per ogni giorno. Alcune di queste vengono classificate come junk. Cio' puo' essere fatto automaticamente da mailer avanzati come mozilla. Tale classificazione, basata su classificatori bayesiani, sembra essere molto efficace. Sfruttando questa tecnologia si deve ideare un sistema per la classificazione automatica di email in cui l'utente interagisce con la classificazione fatta in automatico per correggerla. Il classificatore quindi apprende e modifica il suo comportamento. La variante rispetto alla tecnologia attuale e' la possibilita' di avere varie categorie. Una email puo' appartenere a varie categorie. La classificazione di alcune email inoltre potrebbe venir forzata in base a regole sintattiche deterministiche che possono rafforzare automaticamente l'apprendimento delle informazioni relative a certe categorie. Il classificatore dovrebbe essere indipendente dalla piattaforma e dal sistema. Si dovrebbe fornire un adattatore per mozilla e/o per altri mailer (outlook?). <http://www.mozilla.org/>

Dominio abbastanza chiaro. Aspetti architetturali: reverse engineering, anche parziale, di mozilla. Non è chiaro a che livello possa avvenire l'integrazione.

move: gestione flessibile spostamento oggetti

Una persona possiede un numero abbastanza elevato di oggetti (per esempio, articoli scientifici o libri o videocassette, ecc.) e deve, per ragioni di spazio, conservarli nella propria casa in molti posti diversi (alcuni in un armadio, alcuni su uno scaffale, ecc.). Poiche' questa persona desidera reperire ciascun oggetto in modo il piu' veloce possibile, si vorrebbe ottenere un sistema in grado di determinare, in ogni momento, la posizione di un oggetto, "tracciandone" gli eventuali spostamenti da un armadio ad un altro. Idealmente, l'utente dovrebbe richiedere un particolare oggetto (o elemento che ne faccia parte - per esempio, un film anziche' un'intera videocassetta -) ed il sistema dovrebbe essere in grado di rispondere con l'attuale locazione in cui trovare l'oggetto.

Dominio estendibile con molti dettagli. Aspetti architetturali: standard + uso di tecnologie di autorilevamento. Testing potenzialmente limitato da mancata disponibilita' di strumenti di autorilevamento (magari in emulazione)

ecart: carrello della spesa interattivo

Una delle difficolta' che piu' frequentemente un consumatore si trova ad affrontare in un supermarket e' quella di tenere traccia del costo complessivo dei prodotti che fino a quel momento ha depositato nel proprio carrello. Un sistema molto flessibile, da questo punto di vista, dovrebbe consentire un aggiornamento dinamico (ed immediato) di un costo totale per ogni prodotto che venga inserito nel/rimosso dal carrello. Per esempio, si potrebbe pensare di dotare ogni carrello di un piccolo dispositivo (palmare?) che registri i codici a barre dei prodotti candidati all'acquisto. Addirittura, l'idea potrebbe essere estesa agli scaffali, pensando che ciascuno di essi "sappia" qualcosa su se' stesso (per esempio, quali prodotti sono terminati) e lo comunichi ad un sistema centrale.

Dominio chiaro, ma ulteriormente circoscrivibile. Aspetti architetturali possono coinvolgere l'uso di palmari e tecnologia wireless (magari in emulazione)