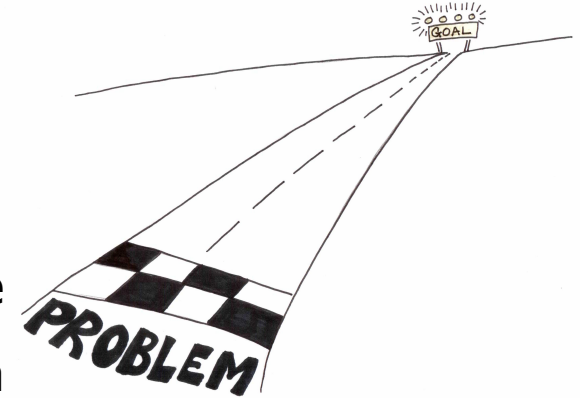


Ingegneria del Software



Informatica – ICD
Università degli Studi di Bari Aldo Moro
Docente: Pasquale Ardimento

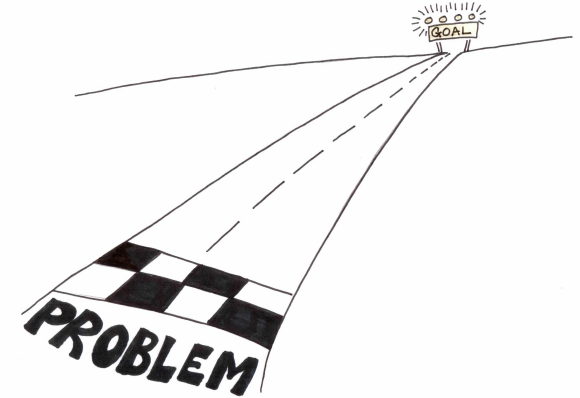
Definizione del problema...



- ⇒ Una società sportiva organizza delle escursioni. Le escursioni hanno durata giornaliera e sono di due tipi: gita in montagna e gita in mongolfiera.
- ⇒ Per ogni escursione sono previsti alcuni tipi di optional acquistabili dall'eventuale partecipante. I tipi di optional sono tre: pranzo, merenda, visita a un sito. I tipi di optional associati ad una escursione possono differire da caso a caso. Ad esempio: la gita in montagna del giorno x può prevedere solo la cena ed il pernottato, la gita in mongolfiera del giorno y non prevede alcun optional, la gita in montagna del giorno z può prevedere tutti gli optional.

... Definizione del problema

- ⇒ Il prezzo degli optional è fisso ed è determinato solo dal tipo. Ogni escursione prevede sia un numero massimo sia un numero minimo di partecipanti.
- ⇒ Il prezzo di un'escursione, i limiti al numero dei partecipanti e tutto ciò che corrisponde alla definizione della stessa escursione è fissato all'atto della sua immissione nel sistema della società sportiva. Tali dati possono essere modificati, in corso di esercizio, dopo che essi sono stati immessi.



Requisiti Funzionali ...

RF₁. Il sistema deve consentire ad un *manager di una escursione* di inserire una escursione in un qualunque momento

RF₂. Il sistema deve permettere al *manager di una escursione* la modifica o l'annullamento della singola escursione. Devono essere avvertiti tutti gli iscritti alla escursione, attraverso la posta elettronica.

RF₃. Il sistema deve permettere ad un *partecipante* di registrarsi al sistema consentendogli di indicare a quali escursioni intende partecipare e la scelta di eventuali optional; il sistema deve calcolare il costo comprensivo degli optional e renderlo noto al partecipante; dopo l'accettazione deve essere avvisato il manager della escursione, attraverso posta elettronica.

... Requisiti Funzionali ...

RF₄. Il sistema deve permettere ad un *partecipante* sia la cancellazione da una escursione sia l'eventuale modifica del numero e del tipo di optional scelti; nel caso della modifica degli optional scelti deve essere calcolato il nuovo costo risultante. In ogni caso deve essere avvertito dei cambiamenti il manager della escursione, attraverso posta elettronica.

RF₅. Il sistema deve permettere ad un *manager di una escursione* sia la cancellazione di un partecipante da una data escursione sia la modifica del numero e del tipo di optional scelti; nel caso della modifica degli optional scelti deve essere calcolato il nuovo costo risultante; in ogni caso deve essere avvertito il partecipante interessato, via posta elettronica, delle modifiche avvenute e delle motivazioni.

Principali Requisiti Informativi...

- ⇒ **RI1.** Ogni escursione è caratterizzata da:
 - data, tipo, numero minimo di partecipanti, numero massimo di partecipanti, tipi di optional possibili, manager della escursione e relativo indirizzo di posta elettronica.
- ⇒ **RI2.** Le escursioni sono di due tipi:
 - gita in montagna e gita in mongolfiera. Costo, numero minimo e numero massimo di partecipanti iscrivibili sono stabiliti all'atto dell'inserimento e possono variare da escursione a escursione. Pure i tipi di optional previsti per ogni escursione sono fissati all'atto dell'inserimento.
- ⇒ **RI3.** I possibili tipi di optional sono:
 - «pranzo», «merenda», «visita ad un sito». I relativi costi sono fissi indipendentemente dalla escursione a cui sono associati.

... Principali Requisiti Informativi

- ⇒ **RI4**. Ogni escursione ha uno ed un solo manager
- ⇒ **RI5** Un manager può gestire più escursioni
- ⇒ **RI6** Un partecipante è caratterizzato da:
 - ▣ codice fiscale, nome, cognome, indirizzo, anno di nascita, sesso, numero tessera sanitaria, certificato di Sana e Robusta Costituzione (SRC), data di rilascio del certificato di SRC, indirizzo di posta elettronica.
 - ▣ Il certificato di SRC è memorizzato nel sistema come un file di testo
- ⇒ **RI7** Un partecipante non può iscriversi ad una escursione se la data di rilascio del certificato SRC è antecedente alla data di svolgimento della escursione di più di un anno.

... Principali Requisiti Informativi ...

- ⇒ **RI8** I dati di un partecipante devono essere registrati alla prima escursione a cui intende prendere parte. Essi devono essere mantenuti dal sistema anche dopo il termina della escursione a cui ha partecipato ed anche nel caso in cui si cancelli all'unica escursione a cui si è iscritto.
- ⇒ **RI9** Non è possibile apportare modifiche a escursioni in corso o già disputate
- ⇒ **RI10** E' possibile apportare modifiche ad una escursione fino a due giorni prima della data di svolgimento della stessa
- ⇒ **RI11** Il sistema non gestisce liste di attesa per le escursioni
 - qualora il numero massimo di partecipanti per una escursione sia incrementato, in seguito ad una modifica apportata dal manager di tale escursione, il sistema non informerà coloro che avrebbero voluto iscriversi e non hanno potuto a causa del già avvenuto raggiungimento del numero massimo di partecipanti

... Principali Requisiti Informativi

- ⇒ **RI12** Qualora il numero di partecipanti ad una escursione sia ridotto, in seguito ad una modifica apportata dal manager di tale escursione, ad esempio ad n , l'eventuale numero eccedente di iscritti deve essere azzerato escludendo i partecipanti che, cronologicamente, si siano iscritti come $n+1$ esimo, $n+2$ esimo, ...
 - gli iscritti esclusi devono essere informati attraverso posta elettronica
- ⇒ **RI13** Qualora una escursione sia annullata è necessario avvisare tutti gli iscritti attraverso posta elettronica; una escursione annullata non deve essere mai cancellata dal sistema.

Requisiti delle interfacce utente

- ⇒ L'uso del sistema avviene attraverso un'interfaccia grafica windows-based che mostra menù, pulsanti e campi di testo, attraverso i quali l'utente può scegliere ed operare tra le diverse operazioni effettuabili
- ⇒ Il sistema monitora le operazioni richieste, in modo tale da
 - **RIU1** controllare le operazioni distruttive (chiedendo sempre conferma all'utente prima di eseguire irreversibilmente una di queste)
 - **RIU2** abilitare solo le funzionalità consentite
 - **RIU3** fornire all'utente solo le informazioni necessarie per eseguire le operazioni abilitate.

Requisiti dell'ambiente di esecuzione

- ⇒ **RAE₁₁** Il sistema deve essere progettato come un'applicazione stand-alone, ovvero potrà essere eseguito in maniera completamente autonoma su un sistema isolato

Requisiti di riservatezza ...

- ⇒ **RR1** L'accesso alle funzionalità del sistema è soggetto ad autenticazione
 - l'autenticazione usa la chiave (username, password) per verificare l'identità dell'utente
 - Le password devono essere crittografate
- ⇒ **RR2** Ogni partecipante è autorizzato ad accedere solo ai propri dati
- ⇒ **RR3** Ogni manager di una escursione può accedere solo ai dati delle competizioni da lui gestite
- ⇒ **RR4** Ogni *manager di una escursione* può accedere ai dati di una escursione fino a quando questa si conclude
- ⇒ **RR5** Il *manager del sistema* può accedere, in sola lettura, a tutte le informazioni riguardanti le escursioni, anche a quelle già concluse.

Requisiti operativi: hardware

⇒ **RHW1** Personal computer che possieda almeno la seguente dotazione

- ❑ Processore 1Ghz
- ❑ RAM: 512 MB
- ❑ HD: 40 GB
- ❑ Scheda Grafica da 32MB
- ❑ Mouse
- ❑ Tastiera
- ❑ Risoluzione minima display: 1024x768

Requisiti operativi

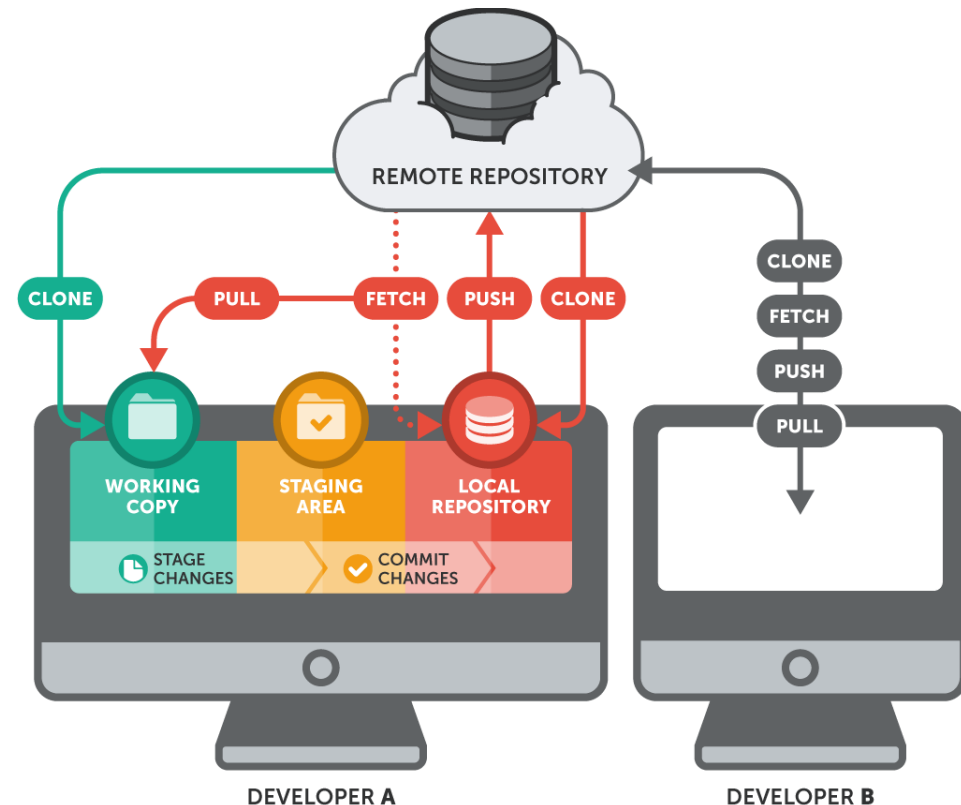
software

Software

- ⇒ Ambiente di modellazione del sistema: Visual Paradigm Community edition versione 14.x
 - <http://www.visual-paradigm.com/download/>
- ⇒ Ambiente di sviluppo: Eclipse IDE MARs (4.5.2) release dipendente dal sistema operativo
 - <https://www.eclipse.org/downloads/>
- ⇒ Linguaggio di programmazione: Java SE Development Kit 8
 - <http://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html>
- ⇒ Database Management System: MySQL Community Server versione 5.7.X
 - <http://dev.mysql.com/downloads/>

GIT

- ⇒ Sistema di controllo della versione distribuito (Version control system o VCS)
- ❑ tiene traccia dei cambiamenti occorsi nei file software
 - ❑ coordina il lavoro su molteplici file tra molteplici persone che possono lavorare su calcolatori differenti
 - ❑ Creato da Linus Torvalds nel 2005 per supportare lo sviluppo del Kernel di Linux



VCS: sistema che registra, nel tempo, i cambiamenti ad un file o ad una serie di file

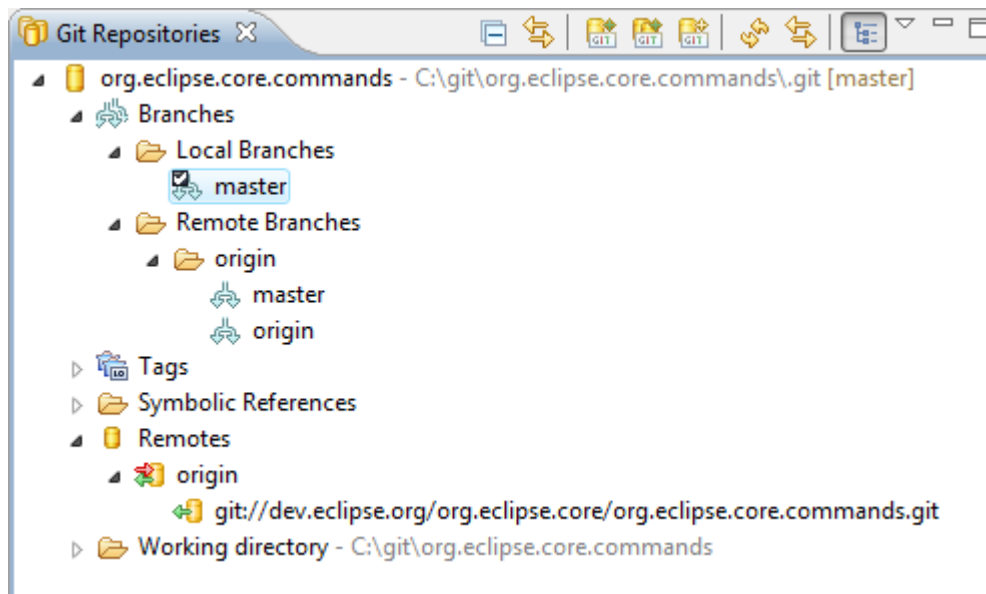
Gitlab (server-side)

- ⇒ servizio di hosting di progetti software basato sull'uso di GIT, strumento per la gestione del versioning di un sistema software;
- ⇒ GitLab permette di:
 - ❑ Gestire repository di progetti software
 - ❑ Gestire ruoli e permessi utente
 - ❑ Gestione delle issue (idee, problemi, bug, ..)
 - ❑ Gestione documentale mediante Wiki
 - ❑ Gestione del versionamento basato su GIT
- ⇒ Versione da usare: community edition
- ⇒ Riferimenti utili
 - ❑ <https://about.gitlab.com/>
 - ❑ <https://docs.gitlab.com/ce/README.html#user-documentation>



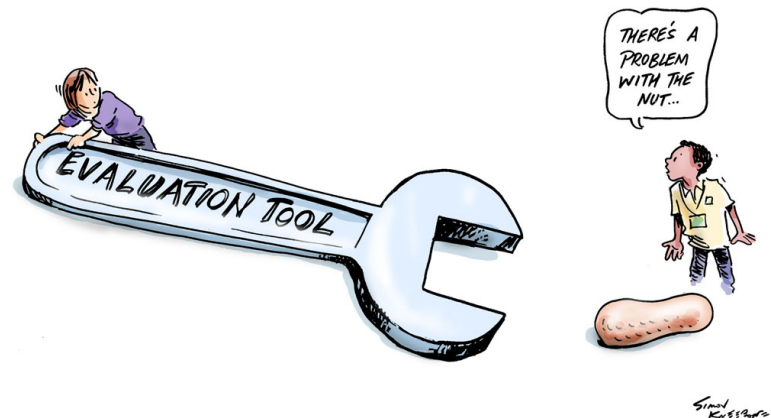
Egit (client-side)

- ⇒ software integrato all'interno di Eclipse per connettersi a repository Git
- ⇒ Riferimento utile
 - ❑ <http://wiki.eclipse.org/EGit>



Stan4J

- ⇒ strumento di analisi della struttura del codice sorgente
 - ❑ Codice scritto in Java in ambiente Eclipse
- ⇒ supporta una serie di metriche quali
 - ❑ "Conteggio" (numero di package, numero di classi per package, ..)
 - ❑ Numero stimato di righe di codice
 - ❑ Complessità ciclomatica di McCabe
 - ❑ Metriche di Chidamber & Kemerer (WMC, DIT, ...)
- ⇒ Per ogni metrica fornisce
 - ❑ feedback sulle misure raccolte
 - ❑ riscontro delle violazioni rispetto ai valori soglia
- ⇒ Riferimento Web
 - ❑ <http://stan4j.com/>



Vincoli: relativi alla fase di analisi e specifica requisiti

- ⇒ Realizzare uno ed uno solo diagramma degli stati
 - ❑ Scegliere e motivare opportunamente l'entità rispetto alla quale realizzare il diagramma degli stati
- ⇒ Uso della tecnica dell'analisi testuale per la realizzazione sia del diagramma delle classi entity sia del diagramma dei casi d'uso
- ⇒ Realizzazione di due diagrammi delle classi: Entity e BCE
 - ❑ Il diagramma BCE è realizzato contestualmente ai diagrammi di sequenza
- ⇒ Realizzazione delle seguenti matrici di tracciabilità
 - ❑ Casi d'uso-Requisiti (funzionali e non funzionali)
 - ❑ Classi-Requisiti (funzionali e non funzionali)
 - ❑ Classi BCE – Requisiti (funzionali e non funzionali)

Vincoli di design

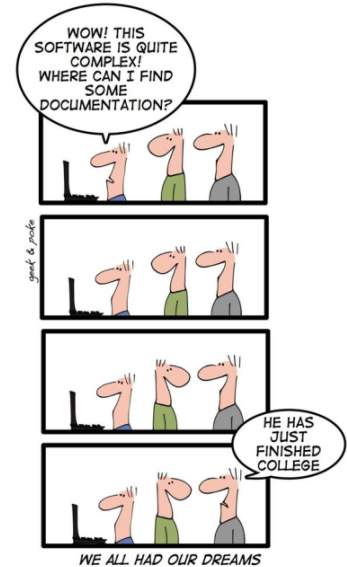
- ⇒ Adozione di uno stile architettuale a propria scelta tra quelli spiegati durante le lezioni (presenti nelle dispense)
- ⇒ Applicazione, opportuna e motivata, di almeno 5 pattern tra quelli spiegati a lezione (presenti nelle dispense)
- ⇒ realizzazione dei diagrammi di sequenza (base ed alternativi) di due funzionalità (a scelta del gruppo)
- ⇒ Realizzazione delle seguenti matrici di tracciabilità
 - ❑ Classi BCE-Diagramma delle componenti
 - ❑ Diagramma delle componenti-Diagramma delle classi

Vincoli di implementazione

- ⇒ Documentazione di tutti i package, classi ed operazioni mediante JavaDoc
- ⇒ Versionamento del codice sorgente mediante Git (GitLab-Egit)

Requisiti operativi: documentazione da produrre

- ⇒ Tutti i manufatti devono essere prodotti conformemente ai template forniti
 - ❑ Documento di specifica dei requisiti software
 - ❑ Documenti di Progettazione
- ⇒ i manufatti richiesti devono essere consegnati su un compact disk riportante, sulla faccia non incisa, le seguenti informazioni generali
 - ❑ "data di consegna", "nome del caso di studio", e per ogni studente informazioni quali nome, cognome, corso di laurea, e-mail



Consegna

- ⇒ **Data di consegna:** per ogni appello la consegna avvenire il giorno indicato sul calendario
- ⇒ **Manufatti da consegnare:**
 - ❑ Documento di Analisi e specifica dei requisiti (SRS)
 - ❑ Documento di Progettazione
 - ❑ Progetto Visual Paradigm (diagrammi UML, analisi testuale, specifiche, ...)
 - ❑ Progetto Eclipse (codice sorgente del prodotto realizzato)
 - ❑ DUMP (SQL) del database
 - ❑ Javadoc, in formato html, del codice sorgente realizzato
 - ❑ Eseguitibile del prodotto realizzato
- ⇒ **Accesso GitLab:** creare un account di sola lettura, fornire le credenziali di tale account al docente (trascritte in un file readme.txt presente nel cd oggetto di consegna)



Note

- ⇒ I software GitLab, EGIT, Stan4J saranno oggetto di specifiche presentazioni
- ⇒ Nelle lezioni relative alla Progettazione del software saranno spiegate le metriche oggetto di rilevazione di Stan4J

