



# *Progetto di Architetture Software e Sicurezza Informatica*

**Come impostare  
la relazione finale  
(valevole come tesi di laurea triennale)**



# Struttura della Relazione

- Capitolo 1 - Raccolta dei requisiti, divisa in:
  - descrizione sintetica della realtà di interesse (nell'ordine delle poche pagine); questa è la descrizione dell'idea di progetto così come ideata dal gruppo.
  - una specifica completa e dettagliata in linguaggio naturale, che metta in evidenza quali sono i dati che il sistema deve gestire, e quali sono le principali funzionalità, includendo per ciascuna funzione individuata una breve descrizione. Devono essere incluse tutte le user stories ed i mock-up dell'applicazione ideata



- Capitolo 2 - Tecnologie e Metodologie Utilizzate
  - Descrivere (dividendo il capitolo in sezioni) le tecnologie utilizzate nel progetto, soffermandosi in particolare su quelle innovative (ovvero non presentate in modo approfondito nei corsi)
  - Esempi positivi (di tecnologie da descrivere)
    - Autenticazione con servizi esterni (Facebook)
    - Uso di API particolari (Google Maps, Wikipedia, etc.)
    - API Bluetooth per smart watch / fitness bracelet
    - Java su set-top-box
    - Specifico framework Ruby
    - ...
  - Esempi negativi (di tecnologie da citare brevemente senza soffermarsi troppo)
    - Java servlet/JSP
    - Tomcat
    - Ruby e Rails
    - ...



- Metodologia utilizzata di conduzione del progetto, ed organizzazione del lavoro
  - Quanti componenti del gruppo, divisione del lavoro, modalità di integrazione (revisioni periodiche ovvero continua), ...
  - Eventuali tool utilizzati (IDE, GIT, directory condivisa su Google Drive, email, ecc.)
  - Considerazioni sulla qualità del processo di sviluppo (da scrivere a posteriori, ovvero a progetto ultimato)
  - Informazioni quantitative (stimare l'effort in termini di tempo e di codice prodotto da parte di ogni membro del gruppo)
    - Google spreadsheet condiviso, da mettere poi come appendice al progetto



- Capitolo 3 - Analisi Concettuale del Sistema
  - Schema ER, con discussione delle scelte critiche effettuate nella progettazione.
  - Glossario delle entità e delle relazioni. Per ogni entità/relazione bisogna specificare: il significato, gli attributi con relativo dominio, gli identificatori. Per le relazioni, occorre inoltre motivare la cardinalità della partecipazione delle entità coinvolte.
  - Elenco di tutti i vincoli non esprimibili mediante lo schema ER, con attenzione a specificare con precisione ed in dettaglio le "regole aziendali" (*business rule*) più complesse



- Capitolo 4 - Progettazione del Sistema
  - Architettura del sistema
    - Tecnologica - usare diagrammi UML e testo, evidenziare aspetti di networking e distribuzione su più tier (se presenti)
    - Logica - usare diagrammi UML e testo
      - Almeno package diagram per descrivere l'organizzazione dei moduli e le classi in ognuno dei package
  - Progettazione Logica del Data Layer
    - Come realizzato in Rails
    - Specifica di come i vincoli sullo schema logico vengono realizzati, e per ogni vincolo, definizione della politica da seguire nelle operazioni critiche.
    - Elenco delle eventuali utenze che devono essere definite, con relative autorizzazioni



- Capitolo 5 - Realizzazione del sistema
  - Data layer
  - Application layer
    - Elencazione di tutte le interfacce (servizi remoti / REST route) offerti al presentation layer, e per ognuna la segnatura delle operazioni e breve descrizione della stessa operazione
      - Diagramma UML
      - Tabella per ogni interfaccia
  - Presentation layer
    - Organizzazione in moduli ed elementi grafici
    - Per ogni user story, sequence diagram che mostra come esso si concretizza nella chiamata ai vari moduli dell'applicazione (su tutti e tre i livelli)
    - Eventuale testo che mostri aspetti particolare (anche in relazione a quanto mostrato nel capitolo 2)



- Capitolo 6 - Dispiegamento e Validazione

- Indicazioni su come distribuire/installare l'applicazione, in tutte le sue componenti (app mobile, database, ecc.)
- Indicazioni su come eseguire un dimostratore dell'applicazione, a partire dalla macchina virtuale (o dalle macchine virtuali) da consegnare
  - Tutto il progetto va consegnato su una o più macchine virtuali ( cf. VirtualBox - <https://www.virtualbox.org/> ) in cui devono essere contenuti
    - i codici sorgenti di tutti i moduli (inclusa la parte di presentation layer)
    - il DBMS e database
    - ...
- Descrivere come è stata effettuata la validazione e test dell'applicazione (numero di utenti, bug riscontrati, lunghezza del periodo di test, ecc.)