

# Report Finale

## Organizzazione del lavoro

Per la realizzazione di questo progetto abbiamo innanzitutto elaborato un'accurata valutazione dei requisiti, ponendo fondamentale importanza nella gestione di tanti casi di utilizzo quanti possibili, in modo da garantire una visione a tutto tondo dell'idea. Importante è stato per noi prevedere qualsiasi necessità dell'Utente finale.

Poi, con gli Use Case e i vari diagrammi del D2, siamo riusciti a valutare più approfonditamente i problemi e le complessità di un'idea così elaborata. Riteniamo tuttavia di aver analizzato adeguatamente tali prospettive.

Con le classi abbiamo cercato prima di tutto di pensare in chiave relazionale. Primaria era la necessità, con tali quantità di dati, di preservare la semplicità e l'ordine dei diagrammi.

Infine, per quanto riguarda lo sviluppo, abbiamo cercato di fare semplicemente il più possibile, spinti da già tanto lavoro fatto per progettare l'applicazione.

Purtroppo, però, nonostante abbiamo implementato una parte corposa e scritto molto codice, speravamo di fare di più.

## Ruoli e attività

Non c'è stata una rigida suddivisione dei ruoli; certamente ognuno aveva delle aree su cui lavorare, ma spesso è capitato a l'uno di mettere mano su campi altrui.

Tale scelta è maturata a causa di due condizioni:

- Nessuno di noi aveva esperienze pregresse, né aveva realizzato progetti di qualsiasi natura
- Irene purtroppo ha abbandonato il progetto appena iniziata la fase di sviluppo, per motivazioni del tutto comprensibili, ovviamente

A grandi linee, le aree a ciascuno assegnate sono state:

Nome	Ruolo	Attività
------	-------	----------

Davide	Project leader / Sviluppatore	Requisiti non funzionali Contest Diagram, OCL, BackEnd APIs, BackEnd logic, Swagger, Autenticazione, Testing Jest, Deployment, Revisione generale di diversi deliverables, codice e documentazione
Giovanni	UX Designer / Sviluppatore	Requisiti funzionali, Use Case e diagrammi D2, Class Diagrams, Database design e Mongoose Schemas, FrontEnd (Vue, html, css, gestione API esterne), Gestione comunicazione e corretto funzionamento di frontend e backend insieme, Documentazione
Irene	UX Designer	Requisiti funzionali, Class Diagram, Documentazione, HTML e CSS di Login, Signup e Privacy Policy

## Carico e distribuzione del lavoro

Di seguito sono indicate le ore di lavoro di ciascuno.

	D1	D2	D3	D4	D5	Totale
Davide	12	20	20	104	1	157
Giovanni	12	25	30	97	1	164
Irene	12	5	10	-	-	

## Criticità

Critiche sono state le seguenti fasi:

- gestione dei requisiti funzionali e tante idee
- realizzazione delle classi (e db)

- Implementazione di backend e frontend con un linguaggio e framework che nessuno di noi aveva mai usato
- Ridefinizione dei ruoli e dei compiti a progetto in corso, causa abbandono da parte di un membro del team (con conseguente studio di tecnologie e teoria associati a ogni ruolo)

## Autovalutazione

Per varie motivazioni, tra cui l'impegno e il tempo dedicati al progetto, per non parlare del fatto che entrambi abbiamo dovuto studiare da capo qualsiasi cosa riguardasse questa materia (non avremmo nemmeno mai visto un diagramma UML, se non fosse stato per Programmazione 2), riteniamo che entrambi, Davide Ciacciolo e Giovanni Francesco Pettinacci, abbiamo diritto alla valutazione 30 con lode. Vorremmo inoltre segnalare il fatto che la gran parte del lavoro è stata svolta soltanto da noi due, dal momento che il terzo componente del gruppo (Irene Trazzi) ha rinunciato al corso di studi a progetto in corso, e che uno di noi due è stato impegnato in un progetto Erasmus nello stesso periodo. Inoltre, essendoci appassionati col tempo a questo progetto, e dal momento che fin dalla prima modellazione dei requisiti questo si rivelava essere parecchio ambizioso (essendo un'intero social network, con caratteristiche di implementazioni particolarmente complesse come la password recovery, l'implementazione della mappa degli eventi, etc.) , crediamo di aver prodotto grandi quantità di codice e di documentazione, delle quali possiamo dire di ritenerci orgogliosi. Per comprendere le dimensioni di questo progetto, approssimativamente il numero di linee di codice del backend è intorno ai 2800 (senza la parte di testing), quelle del frontend ammontano a circa 4000.