

# Riunione con IMOLA INFORMATICA

RAMtastic6

03 maggio 2024



email: [ramtastic6@gmail.com](mailto:ramtastic6@gmail.com)

## Informazioni sul documento

Redattori: Filippo T.  
Verificatori: Samuele V.

Verificatori esterni: 

# Indice

<b>1 Partecipanti</b>	<b>3</b>
<b>2 Riassunto dell'incontro</b>	<b>3</b>
2.1 Domande del Gruppo . . . . .	3
2.1.1 Discussione sull'implementazione della Chat . . . . .	3
2.1.2 Progettazione del <i>frontend</i> . . . . .	3
2.1.3 Creazione dei test per il <i>frontend</i> . . . . .	3
2.1.4 Organizzazione delle cartelle del <i>frontend</i> . . . . .	3

# 1 Partecipanti

NOME	DURATA
Alessandro (IMOLA INFORMATICA)	20m
Filippo T.	20m
Leonardo B.	20m
Michele Z.	20m
Riccardo Z.	20m
Samuele V.	20m
Davide B.	20m

## 2 Riassunto dell'incontro

### 2.1 Domande del Gruppo

#### 2.1.1 Discussione sull'implementazione della Chat

L'incontro si è aperto con la richiesta, da parte del Gruppo, di poter rendere la funzione di "Chat" un requisito opzionale. La richiesta nasce dal palese ritardo in cui si ritrova il Gruppo.

La richiesta è stata accettata dal proponente, anche perché era stata già concessa in precedenza per un altro gruppo.

#### 2.1.2 Progettazione del *frontend*

La seconda domanda posta dal Gruppo ha riguardato la progettazione del lato *frontend* del Progetto, che di solito richiede il diagramma delle classi.

La risposta data dal proponente è che dipende dal livello di dettaglio, cosa che è definita dai Professori; una cosa che si può fare è però quella di definire la struttura dell'applicazione, quindi, con NEXT, andare a mostrare l'alberatura dei componenti principali.

E' stato dato anche un consiglio per la progettazione del *backend*, che è quello di andare ad analizzare quali sono le chiamate che vengono fatte, e come interagiscono con *frontend* e *database*.

In sostanza, si tratta di "spiegare il flusso dell'applicazione, [...] in modo che in futuro si possa spiegarne il funzionamento semplicemente guardandone la struttura".

#### 2.1.3 Creazione dei test per il *frontend*

Il Gruppo non sapeva bene come si potessero realizzare degli *unit test* per il frontend.

Il proponente, dopo aver ribadito che comunque è bene, al fine di raggiungere l'80% di copertura richiesta, concentrarsi sui test per il *backend*, più semplici, ha spiegato che una buona pratica per fare *unit testing* del frontend è quella di definire come la UI debba modificarsi nel corso di un'interazione.

E' stato consigliato di concentrarsi principalmente sui componenti principali, di modo che siano ben definiti i vari esiti di un'interazione, e che non appaiono errori non gestiti.

#### 2.1.4 Organizzazione delle cartelle del *frontend*

Come ultima domanda, il Gruppo ha chiesto un consiglio su come gestire le cartelle del progetto, lato *frontend*; specialmente, se fosse il caso di creare una cartella "components", figlia della cartella

”app”, dove inserire i sorgenti dei vari componenti. E’ stata fatta una condivisione schermo per mostrare la struttura usata dal *PoC*.

Il proponente ha confermato che inserire una cartella ”components” dove salvare i componenti condivisi da più pagine è una buona pratica. Per componenti specifici di una singola pagina, invece, come può essere il file *Loading.tsx*, caricato in automatico da NEXT al momento del fetch dei dati per, ad esempio, la visualizzazione delle ordinazioni, una buona idea può anche essere quella di inserire il sorgente direttamente dentro la cartella che contiene la pagina in cui sarà utilizzato.

E’ stato anche proposto di inserire una cartella madre ”src”, all’interno della quale salvare il contenuto di ”app” e la cartella ”components”, per organizzare ancora più in dettaglio la struttura del *frontend*.