

# Progetto Sistemi Distribuiti 2020-2021

Matteo Scarpone, Riccardo Berto

June 7, 2021

## 1 Introduzione

Questa traccia è volta alla costruzione di un'applicazione distribuita che coinvolge più di un calcolatore. Per l'implementazione del progetto è possibile scegliere liberamente linguaggi di programmazione, *framework* di sviluppo ed eventuali librerie di supporto. L'implementazione non è quindi legata ai soli strumenti che sono stati trattati durante le lezioni di laboratorio. Il progetto può essere svolto singolarmente o in coppia.

## 2 Progetto Kanban Board

Lo scopo di questo progetto è di sviluppare una *web-application* che consenta di organizzare il lavoro di un *team* di sviluppo tenendo traccia dei progressi svolti. Alcuni esempi di applicazioni di questo genere sono:

- <https://kanbanflow.com/>
- <https://trello.com/it>
- <https://kanboard.org/>

Il progetto è strutturato attraverso una base dati che deve essere memorizzata sul server tramite una delle seguenti possibilità:

- File su memoria secondaria in formato JSON, XML, TOML o YAML
- Un database come ad esempio PostgreSQL, SQLite, Redis, CockroachDB, etcd ...

Per descrivere i dati contenuti nella board è necessario definire la struttura di un *tile* e la struttura di una *colonna*. Un *tile* è la più piccola unità che costituisce la kanban board. È formato da:

- Titolo
- Autore
- Contenuto
- Tipo di messaggio (*organizzativo* o *informativo*)

La board deve consentire di trattare tile con due tipi di contenuto alternativi: *testuale* e *multimediale*. I tile testuali contengono un paragrafo di testo indefinitamente lungo, mentre i tile multimediali contengono un file immagine. Il file immagine deve essere memorizzato in formato ridimensionato: se il file immagine caricato dall'utente in fase di creazione del tile ha una risoluzione superiore a  $900 \times 900$ , il server deve ridimensionare l'immagine. È necessario che i tile risultino visivamente distinti a seconda del tipo di messaggio (organizzativo o informativo). Il tipo di messaggio viene selezionato soltanto in fase di creazione del tile.

Ogni *tile* deve essere contenuto in una colonna. Alla colonna sono associati un titolo e uno stato (*in corso* o *archiviato*). È necessario rappresentare la relazione di appartenenza che associa un insieme di tile ad una specifica colonna. La scelta della gestione della relazione all'interno della base dati è lasciata allo studente.

Si deve quindi implementare un'interfaccia web che permetta di visualizzare le varie colonne con i tile in esse contenuti e uno o più form per la creazione di un nuovo tile o di una nuova colonna. Durante la creazione di un tile deve essere indicata una colonna di appartenenza. Il titolo di ogni colonna deve essere univoco mentre più tile possono condividere lo stesso titolo. L'interfaccia web deve consentire lo spostamento di un tile da una colonna ad un'altra. Non è richiesto che questo avvenga tramite trascinamento: è possibile implementare questa logica attraverso un form di modifica del tile. Un tile può essere aggiornato in una o tutte le sue parti. Una colonna può subire la modifica al titolo, a patto che non esistano altre colonne con lo stesso titolo. L'applicazione deve consentire anche l'eliminazione di tile e colonne con stato *in corso*. L'eliminazione di una colonna provoca l'eliminazione di tutti i tile associati ad essa. Inoltre, in ogni momento, è possibile cambiare lo stato di ogni colonna. Le colonne archiviate sono visualizzate in una pagina diversa da quella in cui sono presenti le colonne il cui stato è *in corso*. Alle colonne archiviate non possono essere aggiunti nuovi tile. Una colonna nello stato *archiviato* non può essere cancellata.

### 3 Modalità di consegna

La *deadline* per la consegna del progetto è fissata per il giorno 1/07/2021 alle ore 13:00 (CEST). Per consegnare l'elaborato si userà la seguente pagina: <https://elearning.unimib.it/mod/assign/view.php?id=742436>. La consegna dovrà essere fornita tramite un file archivio di tipo **zip**. Non sono ammessi altri formati di compressione. All'interno dell'archivio, oltre ai file in cui è presente l'implementazione del progetto, dovranno essere presenti i seguenti file:

1. **README**: in questo file devono essere indicati nome, cognome e matricola di ogni componente del gruppo. Eventualmente è possibile inserire una descrizione o introduzione al lavoro che il gruppo ha svolto.
2. **ISTRUZIONI**: in questo file deve essere indicata una chiara modalità per poter eseguire e testare correttamente l'elaborato consegnato.

È possibile scrivere i file scegliendo un formato tra:

- Plain text
- Markdown
- PDF
- Org

L'assenza di uno dei file sopralistati comporterà l'esclusione dalla correzione del progetto. Ogni consegna sarà controllata sia automaticamente sia manualmente per verificare eventuali plagii. Nel caso in cui venissero riscontrati casi di plagio, i progetti in questione saranno esclusi dalla valutazione.

### 4 Spunti su tecnologie interessanti

Si propone di seguito una lista di tecnologie che è possibile valutare per l'implementazione del progetto. Questa lista non è esaustiva e non rappresenta un vincolo.

- Golang e framework Gin
- Scheme con implementazione Racket
- Rust con framework Rocket o con framework Actix
- Haskell con framework Scotty o con framework Yesod
- Python con framework Flask o con framework Django
- Ruby con framework Ruby On Rails
- Java con framework Spring