**INTRODUZIONE** 

# Progetto ingegneria del software M.E.D (Medical Environment Database)



- Gotti Daniele, matricola 1079011
- Bolis Filippo Antonio, matricola 1079493
- Mazzoleni Gabriele, matricola 1079514
- Masinari Gabriele, matricola 1079692





L'obiettivo del progetto è di sviluppare un software alternativo finalizzato alla gestione automatica dei dati relativi al percorso clinico del paziente, compresi raccolta, scrittura e consultazione.

Questa soluzione mira a raccogliere informazioni sui parametri vitali al fine di migliorare l'efficienza ospedaliera e la gestione dei dati. Il software è progettato con un'interfaccia semplice e intuitiva per facilitare l'utilizzo durante tutto il processo, gestendo il percorso clinico del paziente dal suo arrivo fino alla sua dimissione.

DIFFICOLTÀ INCONTRATE

- Utilizzo di Github (almeno inizialmente)
- Realizzazione del diagramma delle classi
- Capire il funzionamento di jooq
- Gestire le numerose classi di collegamento tra logica e gui
- Rispettare la variante del modello MVC
- Gestione delle tabelle
- Collocazione ottimale dei componenti grafici
- Creare un tema unico per rendere omogenei colori e font dell'app con il logo creato



Programmazione: Programmazione ad oggetti Modellazione: Diagrammi UML



Java, Markdown, SQL



Eclipse IDE, StarUML, Github/Github Desktop, draw.io, Window Builder, Db browser, Jautodoc, Structure101 Studio

PARADIGMI E TOOLS



GitHub



Issues, branches, pull request

SOFTWARE CONFIGURATION

SOFTWARE LIFE CYCLE

## **METODO SCRUM**

- Analisi requisiti: esigenze e specifiche software
- Progettazione: creazione di una soluzione per il soddisfacimento requisiti
- Sviluppo: implementazione e scrittura del codice
- **Testing:** test delle parti principali del codice

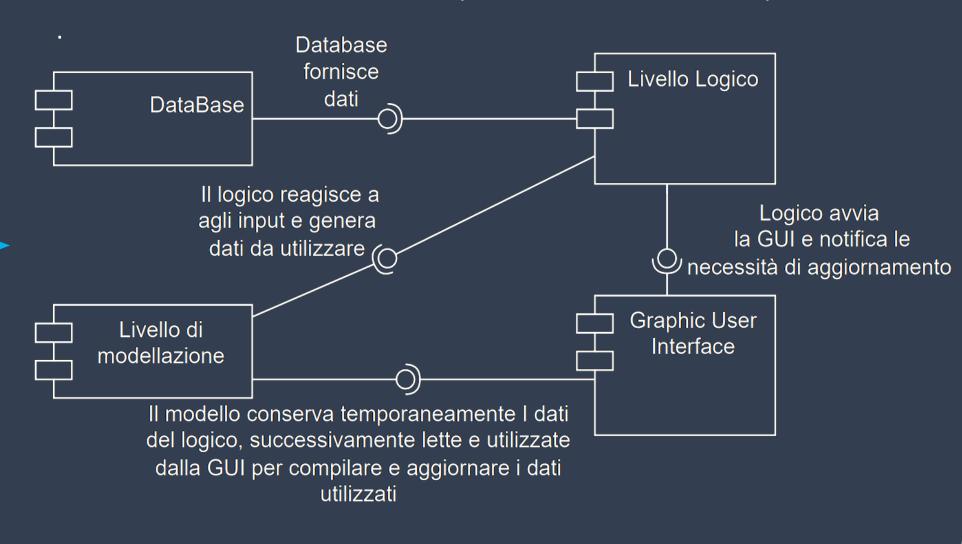


**REQUISITI** 

I requisiti sono stati elicitati tramite intervista diretta con un possibile cliente, un operatore in campo ospedaliero, e attraverso l'analisi delle funzionalità del software ospedaliero attualmente utilizzato in alcune strutture del territorio.

Tutti i dettagli relativi ai requisiti sono stati dettagliatamente descritti nel documento denominato "Documentazione del progetto.pdf".

# Variante Stile MVC (Model-View-Controller)

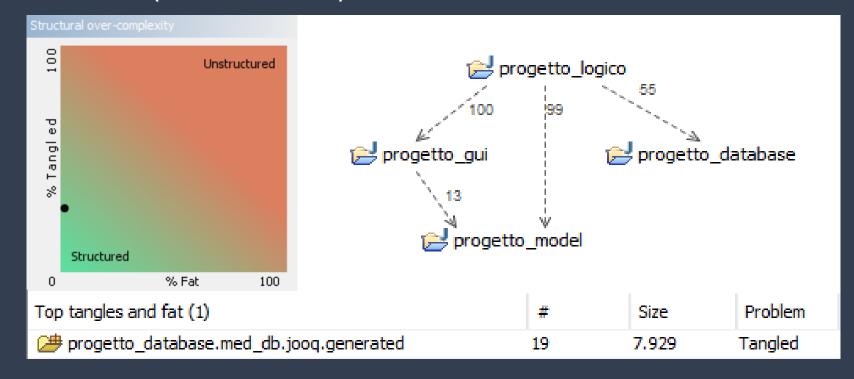


**ARCHITETTURA** 

### Pattern Utilizzati:

- Singleton
- Astrazione Occorenza

# Metriche(Structure 101):



DESIGN PATTERN

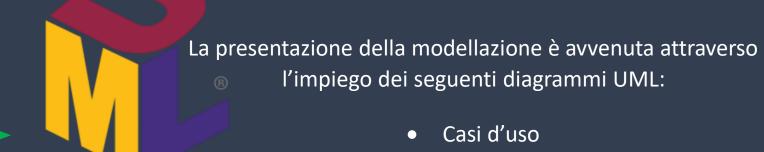


L'implementazione è completa, il sistema è completamente funzionante ed è eseguibile tramite il file main.java, presente nel progetto logica, all'interno della cartella "eseguibile".

Abbiamo utilizzato Eclipse sia per la creazione del database che per la realizzazione del codice in linguaggio java, inoltre è stato utilizzato il tool Window Builder per la creazione della GUI.

**IMPLEMENTAZIONE** 

Procediamo con una breve dimostrazione del funzionamento dell'applicazione creata.



**MODELLAZIONE** 

• Attività

Classe

Sequenza

• Macchina a stati (Evidenziando gli stati del paziente)

Componenti

• Diagramma ER

Package



**TESTING** 

Per collaudare il programma abbiamo sia utilizzato i test Junit di Eclipse sia eseguito test diretti sull'eseguibile per verificarne il corretto funzionamento.

Questi test ci hanno permesso di identificare alcuni difetti di programmazione, oltre che a definire possibili migliorie al programma per quanto riguarda la comodità d'uso (per esempio, l'implementazione di un meccanismo di log out).