Traccia

Un centro di pronto soccorso registra i nominativi dei pazienti che necessitano di un intervento classificando i casi come “Rossi” (molto urgenti), “gialli” (urgenti), Bianchi (poco urgenti).

Ovviamente tutti i casi Rossi dovranno essere gestiti prima dei casi Gialli e i Gialli prima di quelli Bianchi.

Progettare, mediante un diagramma delle classi UML e implementare in Java una classe ProntoSoccorso che consenta di eseguire le seguenti operazioni, gestendo in modo adeguato le relative eccezioni:

- aggiunta di un paziente alla lista d’attesa in modo da rispettare l’ordine di priorità;

- recupero del nominativo ed eliminazione del paziente col massimo grado di urgenza dalla lista;

- determinazione del numero di pazienti “Rossi”, “Gialli” e “Bianchi” in attesa;

- eliminazione dalla lista di un paziente specificato.

Tutti i pazienti che ricevono le prestazioni del pronto soccorso, sono memorizzati in un file che tiene

traccia, oltre del nominativo e della data, anche del tipo di prestazione e del ticket pagato, tenendo

conto del fatto che alcuni pazienti non usufruiscono dell’esenzione.

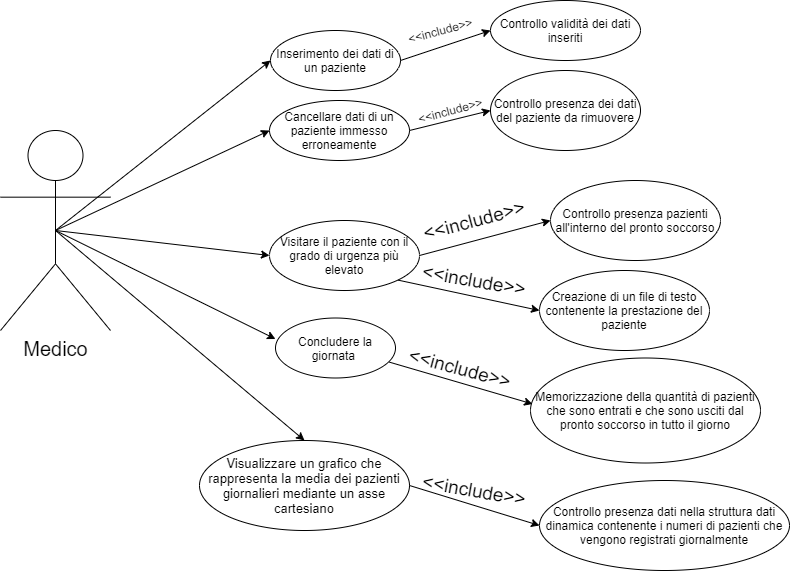
Si preveda, dunque, la possibilità di visualizzare l’elenco dei pazienti paganti, l’elenco dei pazienti

minorenni e la media giornaliera dei pazienti visitati.

Analisi

Il problema richiede di creare un programma in grado di gestire i nominativi dei pazienti che necessitano di un intervento. I pazienti vengono classificati in base alla gravità del caso che può essere identificata tramite 3 colori. Rosso in caso di necessità molto urgente, Giallo in caso di necessità urgente e Bianco in caso di necessità poco urgente. Ovviamente un caso più urgente avrà precedenza sugli altri casi. La registrazione di un paziente avviene riportando i dati personali all’interno di un file chiamato Lista Pazienti. Inoltre ci saranno altri 4 file dove saranno riportati in ognuno di essi i casi Rossi, Gialli e Bianchi e la cronologia dei pazienti registrati. Nel momento in cui il medico andrà a visitare il paziente con il grado di urgenza più elevato il programma aprirà una nuova finestra che richiederà l’inserimento dei dati quali: l’orario di uscita e la prestazione del paziente. Subito dopo aver inserito questi dati, il programma dovrà creare un file che sarebbe la prestazione del paziente. Infine, il programma, permetterà di poter eliminare un paziente dalla lista di attesa, determinare il numero di pazienti in lista d’attesa e visualizzare l’elenco dei pazienti paganti, minorenni e la media giornaliera dei pazienti visitati.

Casi d’uso



Diagrammi delle classi

|  |
| --- |
| CreaFile |
| -NOME\_FILE : string  fileWriter : FileWriter |
| +CreaFile(String NOME\_FILE)  +apri() : void  +chiudi() : void |

|  |
| --- |
| CreaPrestazione |
| -NOME\_FILE : string  inputTastiera : BufferedReader  printer : PrintWriter  input : BufferedWriter  fileWriter : FileWriter |
| +CreaPrestazione(String prestazione, String NOME\_FILE)  +apri : void  +chiudi : void  +scrivi : void |

|  |
| --- |
| CreaTesto |
| -NOME\_FILE : String  -nome : String  -cognome : String  -colore\_caso : String  -data: String  -esenzione: String  -sintomi : String  -ora\_entrata : String  -maggiorenne : String |
| +Apri() : void  +chiudi() : void  +scrivi(DatiPaziente dl) : void  +acquisici() : DatiPaziente |

|  |
| --- |
| CreaTestoFine |
| -NOME\_FILE : String  -nome : String  -cognome : String  -colore\_caso : String  -ora\_entrata : String  -ora\_uscita : String  -magg : String  -inputTastiera : BufferedReader  -printer : PrintWriter  -input : BufferedWriter  -fileWriter : fileWriter |
| +CreaTestoFine(String nome,String c, String colore\_caso,String ora\_entrata, String ora\_uscita, String magg, String NOME\_FILE)  +apri() : void  +chiudi() : void  +acquisisci() : DatiFine  +scrivi(DatiFine dl) : void |

|  |
| --- |
| DatiFine |
| -nome : String  -cognome : String  -colore\_caso : String  -ora\_entrata : String  -ora\_uscita : String  -magg: String |
| +DatiFine()  +DatiFine(String linea)  +toRecord()  +getNome() : String  +getMagg() : String  +setMagg(String magg) :void  +getOra\_entrata() : String  +getOra\_uscita(): String  +setOra\_uscita(String ora\_uscita) : void  +setOra\_entrata(String ora\_entrata) : void  +setNome(String nome) : void  +getCognome() : String  +setCognome(String cognome) : void  +getColore\_caso() : String  +setColore\_caso(String colore\_caso) : void |

|  |
| --- |
| DatiPaziente |
| -nome : String  -cognome: String  -colore\_caso : String  -data: String  -esenzione : String  -sintomi: String  -ora\_entrata : String  -maggiorenne : String |
| +DatiPaziente(String nome,String cognome, String colore\_caso , String data , String esenzione, String sintomi, String ora\_entrata, String maggiorenne)  +DatiPaziente()  +DatiPaziente(String linea)  +toRecord()  +getNome() : String  +getOra\_entrata() : String  +getSintomi() : String  +setSintomi(String sintomi) : void  +setOra\_entrata(String ora\_entrata) : void  +setNome(String nome) : void  +getCognome() : String  +setCognome(String cognome) : void  +getColore\_caso() : String  +setColore\_caso(String colore\_caso) : void  +getData() : String  +setData(String data) : void  +getEsenzione() :String  +setEsenzione(String esenzione) : void  +getMaggiorenne() : String |

|  |
| --- |
| Paziente |
| -nome : SimpleStringProperty  -cognome : SimpleStringProperty  -colore\_caso : SimpleStringProperty  -data : SimpleStringProperty  -esenzione : SimpleStringProperty  -sintomi : SimpleStringProperty  -ora\_entrata : SimpleStringProperty  -maggiorenne : SimpleStringProperty |
| +Paziente(String nome,String cognome,String colore\_caso,String data,String esenzione, String sintomi, String ora\_entrata , String maggiorenne)  +getEsenzione() : String  +getMaggiorenne() : String  +getOra\_entrata() : String  +getSintomi() : String  +getNome() : String  +getCognome() : String  +getColore\_caso() : String  +getData() : String  +setNome(SimpleStringProperty nome)  +setEsenzione(SimpleStringProperty esenzione)  +setSintomi(SimpleStringProperty sintomi)  +setMaggiorenne(SimpleStringProperty maggiorenne)  +setOra\_entrata(SimpleStringProperty ora\_entrata)  +setCognome(SimpleStringProperty cognome)  +setColore\_caso(SimpleStringProperty colore\_caso)  +setData(SimpleStringProperty data) |

DFD ( Data Flow Diagram)