
Problem definition

In this experiment, we are expected to develop a hospital management system. We are expected to implement HMS with the given rule.

With the help of this experiment, we will get familiar with the concept of inheritance, relationships among classes by using object references, control of multiple instances of classes in Java.

Method & Algorithm

Before taking input from input.txt file, my program loops into admission.txt and patient.txt in order to add all admissions and patients to two different array named admissions and patients. With done this, program starts to read input.txt line by line . In case of creating admission and assigning operations to patient, I used Decorator Design Pattern which allows to add new functionality to an existing object without altering its structure. Program loops inside examinations and add examinations intertwined in constructor.

My program includes two DAO classes (Data Access Object) for the functions I stated below. Moreover, these DAO classes hold a string named output which stores informations about transactions and adds to output.txt file.

After reading input.txt and making transactions, my program loops into admissions and patients arrays and writes all admissions and patients to txt files with the hierarchy which is stated in pdf.

Functions

createAdmission(string, Admission) ➡ This function creates an admission and add it to admissions array. When InputManager sees this order, it loops in input.txt createadmission line, addexamination lines) and sends examinations to decorator pattern.

patientDao.add() ➡ This function adds patient to patients array.

createWritingString () ➡ This function loops in admissions or patients arraylist and converts all admissions and patients with properties to string.

writeToFile(String) ➡ Takes a string from createWritingString() function and writes this string to admissions.txt file or patients.txt file.

getById(ID) ➡ Returns an admission or a patient by looping the arrays (admissions, patients) with specific id.

deleteById(ID) ➡ Deletes an admission or a patient by looping the arrays (admissions, patients) with specific id.

printOperations() ➡ Prints operations which given to decorator pattern for an admission.

cost() ➡ Prints cost which calculated from decorator pattern for an admission.

UML CLASS DIAGRAM

