

Ens demanen elaborar un CRM de gestió de clients i que tindrà que gestionar a través d'un menú d'opcions les altes d'un client, consulta d'un client o tots els clients, generar pressupost, modificació d'un client i eliminació d'un client.

La classe Client disposarà dels següents atributs: id, nom, 1er cognom, 2on cognom (opcional), direcció, telèfon, e-mail i comentaris. També els mètodes Getters i Setters corresponents.

La classe Pressupost disposarà dels següents atributs: id, concepte, valor i iva. També els mètodes Getters i Setters corresponents.

Com a mètodes, tindrem que elaborar els corresponents Getters i Setters propis de la classe i un mètode anomenat generarPressupost(). Aquest mètode ha de rebre el concepte, valor i IVA, per poder retornar un pressupost com l'exemple:

20 h. de desenvolupament web -> 1000 Euros

Subtotal = 1000 Euros

Iva(21%) = 210 Euros

Total = 1210 Euros

L'opció alta d'un client ha d'anomenar el mètode altaClient(), i aquest alhora haurà de demanar a l'usuari que introdueixi per pantalla les dades mencionades anteriorment (o haurà de rebre aquestes dades per paràmetre), exceptuant el id que s'haurà de autogenerar emprant Math. Un cop hagi demanat totes aquestes dades, s'ha d'instanciar l'objecte de la classe i afegir-lo a un array de clients.

L'opció d'eliminar un client ha d'anomenar el mètode donarDeBaixaClient(), i aquest haurà de demanar a l'usuari el nom o el id del client que es vol donar de baixa i si el client està dintre de l'array, l'eliminareu i mostrarem un missatge per pantalla informant l'usuari que hem eliminat el client. Si no hem trobat el client dintre de l'array, avisarem l'usuari dient que el client no estava dintre de la nostra aplicació.

L'opció de consulta d'un client o de tots els clients, ha d'anomenar el mètode veureClients(param1) a on s'indicarà un paràmetre que permetrà el mostrar un client o tots. En el cas de desitjar visualitzar un únic client, haurà de demanar a l'usuari el nom o id del client que vol que mostrem per pantalla, si el client està a la nostra aplicació, li mostrarem tota la informació incloent els pressupostos vinculats, si no hi és, informarem l'usuari dient que el client que ens ha demanat no està dintre de la nostra aplicació.

En el cas de desitjar visualitzar tots els clients, mostreu tots els atributs de cada client.

L'opció de generar pressupost, ha d'anomenar el mètode crearPressupost(), i aquest haurà de demanar a l'usuari el nom o el id del client per poder comprovar si el client està dintre de l'array. Si el client està a la nostra aplicació, obtindrem tots els atributs del client, es demanarà a l'usuari que indiqui el concepte, valor i iva, per anomenar el mètode generarPressupost() de

la classe Clients, emmagatzemar l'objecte corresponentment vinculat a el client i obtenir una resposta com l'exemple indicat anteriorment. Si no hi és, informarem l'usuari dient que el client que ens ha demanat no està dintre de la nostra aplicació.

L'opció de modificar d'un client, ha d'anomenar el mètode modificarClient(), i aquest haurà de demanar a l'usuari el nom o el id del client que volem modificar. Si el tenim a l'aplicació, li demanarem si vol modificar qualsevol dels atributs, exceptuant el id. Farem la modificació pertinent i avisarem a l'usuari que la modificació s'ha realitzat. En cas contrari, indicarem a l'usuari que no existeix el client dins de l'aplicació.

Un cop creada l'estructura de classes, cal fer en la classe principal que les utilitzi de la següent manera i previ el menú:

1. S'ha de definir un client predefinit amb les diferents dades aportades estàticament i corresponents a cada un dels atributs.
2. Definiu un nou email per l'usuari creat.
3. Genereu un pressupost amb el concepte "170h. Elaboració d'una API", preu 2410 € i amb un iva 21%.
4. Genereu un pressupost amb el concepte "250h. Disseny imatge corporativa", preu 3110 € i amb un iva 21%.
5. Mostreu la info del client que heu creat mitjançant el mètode consulta d'un client.
6. S'ha de definir un altre client amb les diferents dades aportades per l'usuari i corresponents a cada un dels atributs..
7. Mostreu la info de tots els clients existents.
8. Elimineu el últim client creat.

Finalment, si et sobra temps, pots fer les següents tasques:

9. Valideu que no puguin existir dos clients amb els mateixos valors en els atributs.
10. Afegiu l'atribut estatus en la classe Pressupost a on es podrà especificar acceptat, denegat o pagat. En el mètode generarPressupost() afegirem l'estatus.
11. Poder transferir un pressupost d'un client a un altre, sol·licitant a l'usuari qui serà el destinatari.
12. Calcular el total de l'iva acumulat entre totes les factures de la totalitat dels clients enregistrats.