PATRÓ STRATEGY

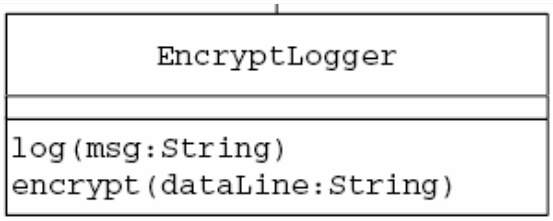
Josep Vila Regàs

Descripció

- Conjunt d'algorismes relacionats.
- Escollir algorismes dinàmicament.
- Es respecte el principi OCP.
  - Cada algorisme implementat a una nova classe.
  - L'objecte client treballa amb la classe interfície.

Classe EncryptLogger

- Es necessita una classe que encripti missatges.
- S'utilitza el mètode *encrypt(dataLine:string)*
- L'algorisme d'encriptació implementat al mètode pot ser de moltes maneres.



Mètodes d'encriptació 1/2

- SimpleEncrption
  - Els caràcters són desplaçats una posició a la dreta.

Example:
Plain text:This is a message
Cipher text:eThis is a messag

- CaesarCypher
  - Els caràcters es subtitueixen pel caràcter 2 posicions a la dreta de l'alfabet.

Example:
Plain text:This is a message
Cipher text:Uijt jt b nfttbhf

Mètodes d'encriptació 2/2

- SubstitutionCypher
  - S'utilitza una taula de substitució de caràcters.

Table 36.2 Sample Letter Substitution Table

A	T	i	B	h	s	a	e	m	X	Y	M	P	C	g	F	Q	w	r	t
s	m	N	o	W	R	T	X	Y	A	B	D	F	I	n	d	i	a	U	S

- CodeBookCypher
  - S'utilitza una taula de substitució de paraules.

Table 36.3 Sample Code-Book Table

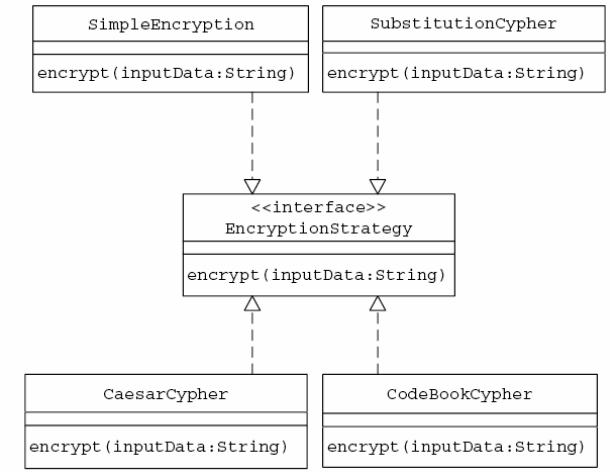
This	Design
Is	Patterns
Book	CD
A	Are

Implementació dels algorismes

- Tots els algorismes dins el mètode *encrypt(String)* utilitzant condicionals.
  - No respecte el principi OCP.
- Aplicar herència on cada subclasse de EncryptLogger implementa un algorisme específic.
  - El client ha de crear una instància nova de la subclasse i reemplaçar l'actual objecte de context.

Implementació: Patró Strategy

- Cada estratègia està encapsulada en una classe diferent.



Classe EncryptLogger

- La classe EncryptLogger:
  - No implementa cap mètode d'encriptació.
  - Punter a l'estratègia actual.
  - El mètode encrypt utilitza el de l'estrègia actual.

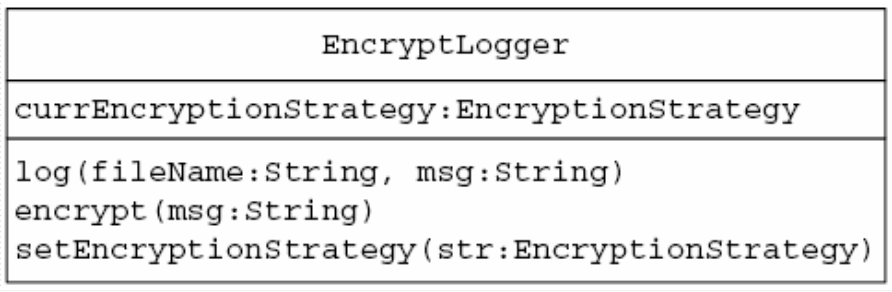
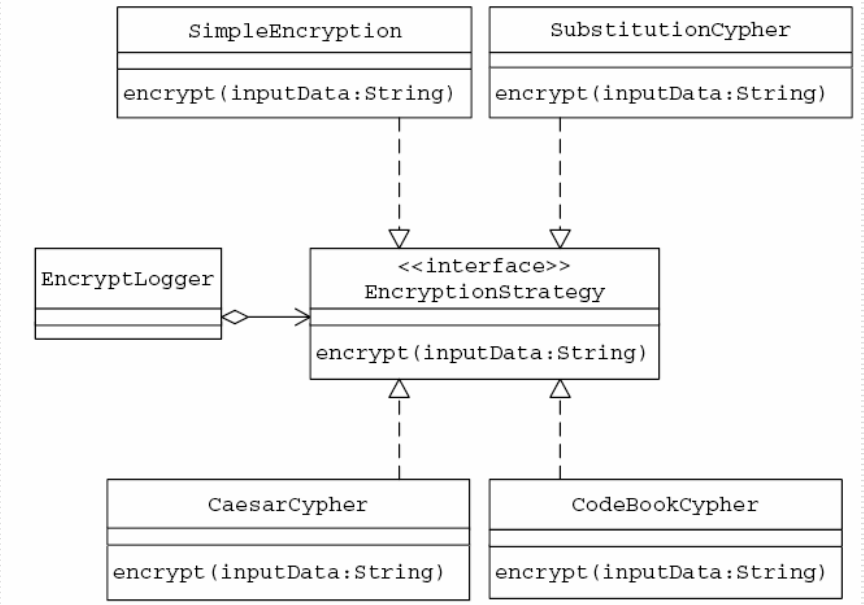


Diagrama de classes.



Implementació del client

```
class LoggerClient {
  public static void main(String[] args) {
    FileLogger logger = new FileLogger();
    EncryptLogger eLogger = new EncryptLogger(logger);
    eLogger.log("log1.txt",
                "this message is to be encrypted & logged");
    EncryptionStrategy strategy = new SubstitutionCypher();

     eLogger.setEncryptionStrategy(strategy);
     eLogger.log("log2.txt",
                 "this message is to be encrypted & logged");
     strategy = new CodeBookCypher();
     eLogger.setEncryptionStrategy(strategy);
     eLogger.log("log3.txt","This is a true statement");
   }
 }//End of class
```