# Laboration – Text inserter

Denna laboration skall lösas individuellt och du skall ha förståelse för de olika delarna i din implementation vid inlämning.Det är ej tillåtet att kopiera en annans students lösning för aktuellt projekt.Du kan bli ombed att redovisa din lösning muntligen.

## Bakgrund

Att kunna spara ner meddelanden i text är vitalt för ett bra liv idag. Applikationen vi arbetar med i denna laboration gör just detta.

Men det har visat sig att risken att någon tjuvläser meddelanden är vanligt. Vi har därför bestämt att anlita dig för att utöka funktionaliteten i applikationen.

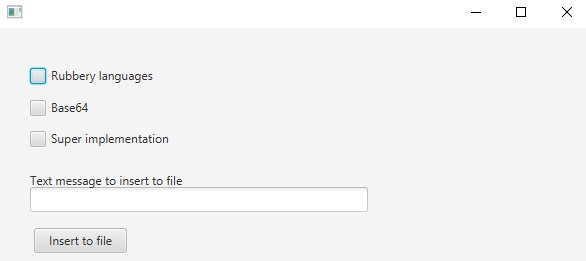
Utbyggnaden av applikationen skall göras med hjälp av designmönstret [**Decorator**](https://sourcemaking.com/design_patterns/decorator)**.**

## Projektet

Kod att utgå ifrån hittas i anslutning till denna fil.

## Att göra

## När du startar applikationen så möts du av följande:



I applikationen är det förnärvarande möjligt att skriva in ett meddelande och trycka på ”Insert to file” det som då sker är att meddelandet sparas i en .txt fil i ditt projekt.

Vi vill nu utöka funktionaliteten så man via checkboxar (redan implementerade i vyn) kan välja om texten skall modifieras (encodas) innan den sparas i text-filen.

Det skall gå att encoda med:   
**Base64**,   
**Rövarspråket** och   
**Super** **Implementation** (*du väljer vad detta skall innebära*).

DU skall även inkludera en kortare förklaring (0.5–1 A4) som beskriver hur din utökning av applikationen förhåller sig till designprincipen OCP (Open Closed Principle).

### Konkret:

* Få önskat resultat i applikationen genom att implementera Decorator pattern.
* Inkludera beskrivning över hur din utökning av applikationen förhåller sig till OCP.

## Förberedelser

* Föreläsningar och laborationer.
* Kursbok, kap 1 och 3.

## Inlämning

Du laddar upp en .zip innehållandes:

1. Koden för en fungerande lösning av applikationen där Decorator pattern är implementerat.

## Betyg

Inlämningen kan ge något av betygen underkänd (U)\* eller godkänd (G).

\*Vid betyget U (underkänd) kan du antingen ges möjlighet att komplettera din lösning upp till betyget G (godkänd) alternativt så behöver du genomföra en helt ny uppgift.

### Betygskriterier

När vi betygsätter inlämningen tittar vi på om:

* Efterfrågat mönster finns implementerat.
* Kvalitet på lösningen.
* Funktionaliteten i programmet.
* Kvalitet på text inlämningen.

## Exempellösningar för rövarspråket och Base64:

Base64.getEncoder().encodeToString(“STRING”.getBytes());

public String rubberLang(String message) {

String vowels = "aeyuoåäöi";

String outString = "";

int length = message.length();

for (int i = 0; i < length; i++) {

if (vowels.contains(message.toLowerCase().charAt(i) + "")) {

char c = message.charAt(i);

outString += c + "o" + c;

} else {

outString += message.charAt(i);

}}

return outString;   
}