

**CENTRO UNIVERSITÁRIO FEI**

ANTONIO GUSTAVO MUNIZ DA SILVA  
GABRIEL BUENO VILA REAL DE OLIVEIRA  
HENRIQUE VITAL CARVALHO

**RELATÓRIO FINAL:**  
**MODELAGEM DE SOFTWARE ORIENTADO A OBJETOS - CC6522**

São Bernardo do Campo

2º sem / 2021

## SUMÁRIO

<b>DIAGRAMAS</b>	<b>4</b>
<b>Lista de Diagramas</b>	<b>4</b>
<b>Dificuldades</b>	<b>4</b>

## **DIAGRAMAS**

### **Lista de Diagramas**

Todos os seis diagramas solicitados (Diagramas de Casos de Uso, Classes, Sequência, Estados, Atividades e Componentes) estão disponíveis no Github, a documentação está hospedada online podendo ser acessada no link indicado pelo repositório. Já os diagramas podem ser abertos por meio do StarUml com o arquivo “Projeto.mdj”.

Link do repositório: <https://github.com/henriquevital00/oil-recycling/tree/main>

### **Dificuldades**

Ao longo do desenvolvimento dos diagramas esbarramos com uma série de dificuldades. Inicialmente, tivemos problemas na construção do diagrama de casos de uso, pois, partindo da prerrogativa de que o sistema era auto suficiente, tornava-se complexo visualizar como os atores (Operador e Gestor) entrariam nesse fluxo. Além disso, como ambos os atores compartilham algumas ações, percebemos que poderíamos simplificar o diagrama que havia sido criado inicialmente.

No diagrama de classes tivemos menos dificuldades, pois como já havíamos mapeado os casos de uso, ficou mais simples a construção das classes. O único percalço que tivemos foi na conexão entre as classes, pois o sistema é complexo e algumas classes, como o tanque, se repetiam ao longo do processo de sintetização de biodiesel.

O diagrama de sequência foi onde mais tivemos dificuldade, pois além de ser difícil de construir foi o que mais tivemos que refazer. Ao final, percebemos que ao invés de um único diagrama extremamente complexo poderíamos construir diagramas menores para cada classe que se interligavam formando o fluxo completo, tornando mais simples a visualização e o entendimento do sistema.

Os diagramas de estados, atividades e componentes também sofreram muitas alterações. O de estado foi mais simples de construir depois que entendemos o erro inicial que tivemos na construção, o de componentes seguiu a mesma premissa. Foram diagramas mais fáceis de construir depois que compreendemos o que havíamos errado.

Um ponto que tornou muito demorado algumas etapas do desenvolvimento foi o próprio StarUML, ele é um programa bem limitado em certos pontos e que travava constantemente, sendo um problema constante que dificultava a continuidade do projeto.