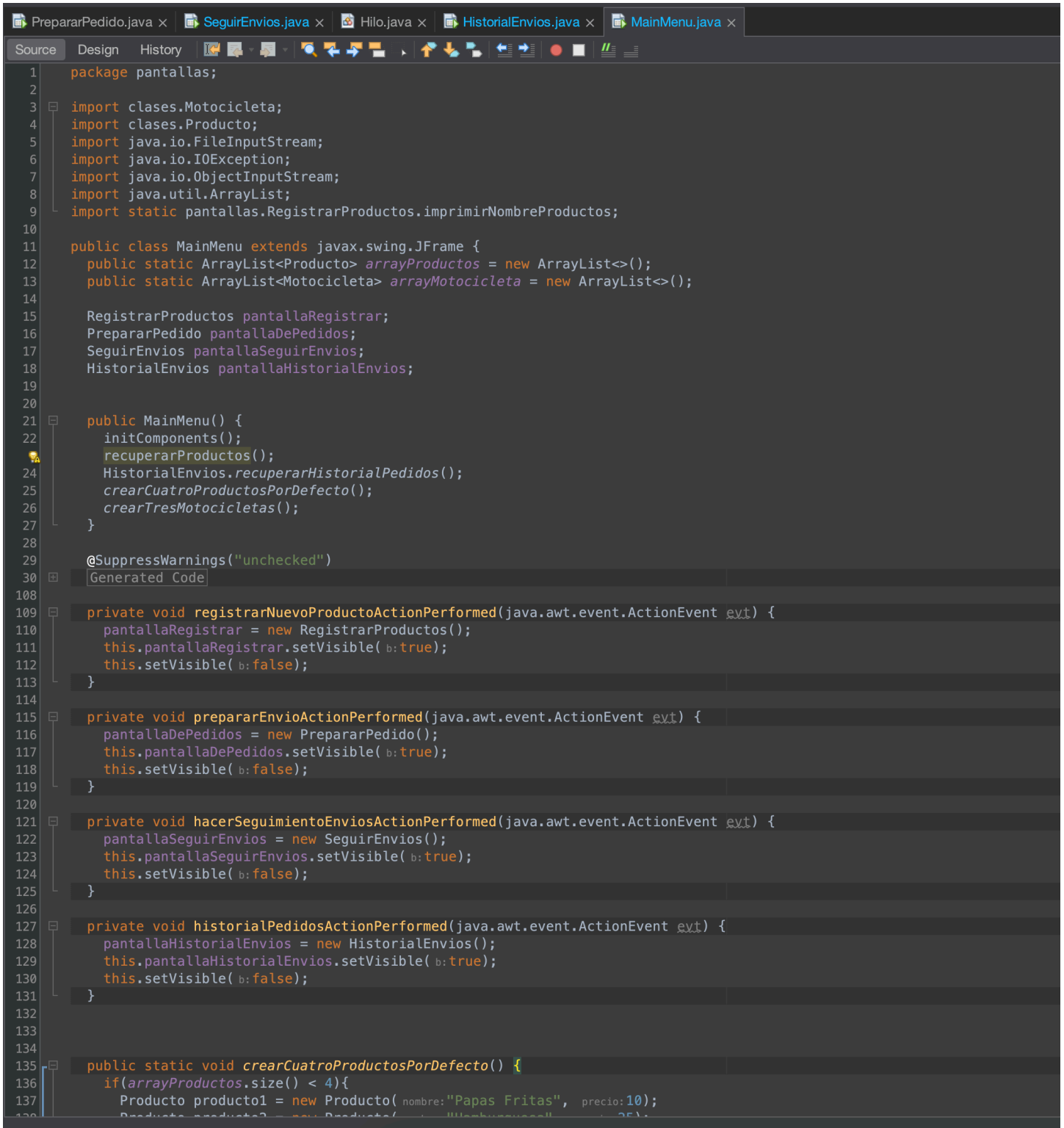


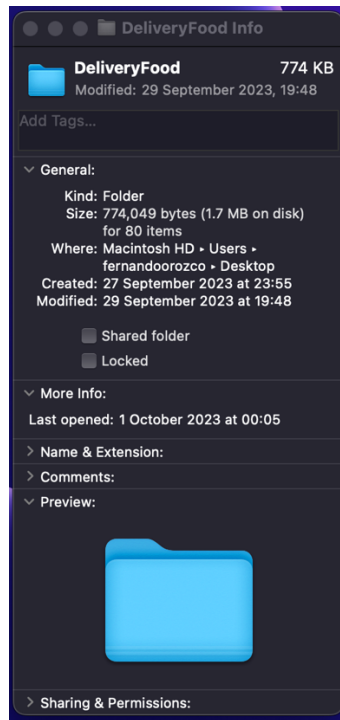
Manual Técnico

El proyecto fue realizado en su totalidad utilizando el lenguaje de programación Java en su versión 19.0.2 en adición de librerías para manejar documentos .bin y creación y consumo de Hilos para realizar distintas tareas al mismo tiempo

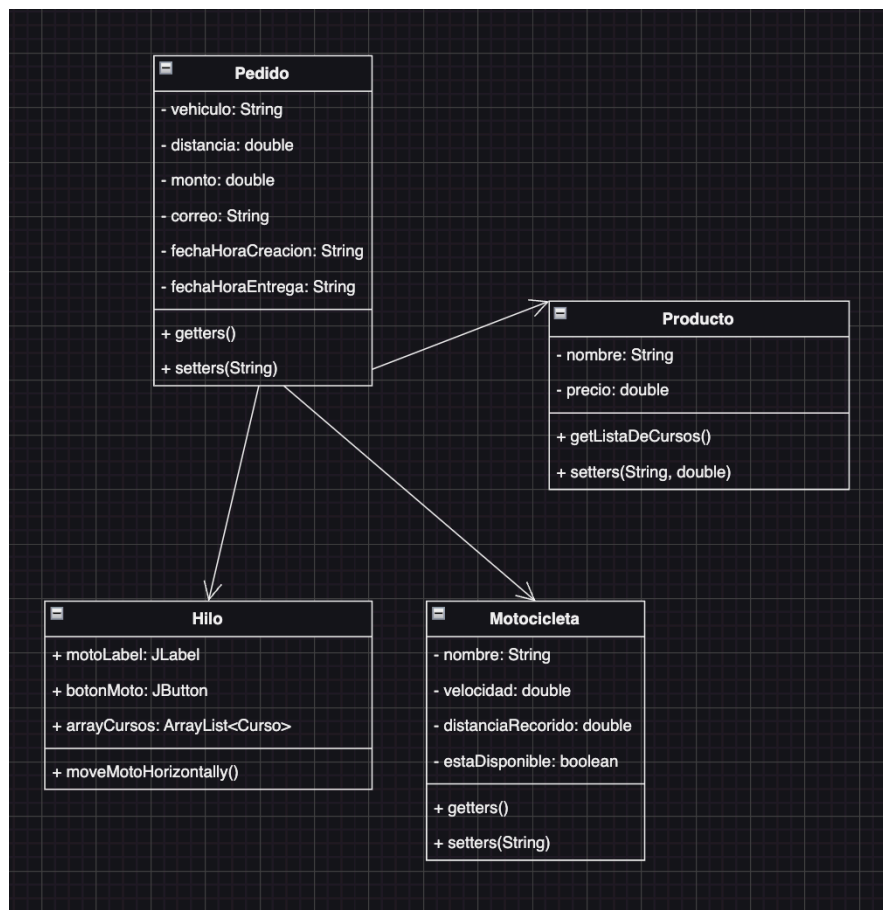


```
1 package pantallas;
2
3 import clases.Motocicleta;
4 import clases.Producto;
5 import java.io.FileInputStream;
6 import java.io.IOException;
7 import java.io.ObjectInputStream;
8 import java.util.ArrayList;
9 import static pantallas.RegistrarProductos.imprimirNombreProductos;
10
11 public class MainMenu extends javax.swing.JFrame {
12     public static ArrayList<Producto> arrayProductos = new ArrayList<>();
13     public static ArrayList<Motocicleta> arrayMotocicleta = new ArrayList<>();
14
15     RegistrarProductos pantallaRegistrar;
16     PrepararPedido pantallaDePedidos;
17     SeguirEnvios pantallaSeguirEnvios;
18     HistorialEnvios pantallaHistorialEnvios;
19
20
21     public MainMenu() {
22         initComponents();
23         recuperarProductos();
24         HistorialEnvios.recuperarHistorialPedidos();
25         crearCuatroProductosPorDefecto();
26         crearTresMotocicletas();
27     }
28
29     @SuppressWarnings("unchecked")
30     Generated Code
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109 private void registrarNuevoProductoActionPerformed(java.awt.event.ActionEvent evt) {
110     pantallaRegistrar = new RegistrarProductos();
111     this.pantallaRegistrar.setVisible( b:true);
112     this.setVisible( b:false);
113 }
114
115 private void prepararEnvioActionPerformed(java.awt.event.ActionEvent evt) {
116     pantallaDePedidos = new PrepararPedido();
117     this.pantallaDePedidos.setVisible( b:true);
118     this.setVisible( b:false);
119 }
120
121 private void hacerSeguimientoEnviosActionPerformed(java.awt.event.ActionEvent evt) {
122     pantallaSeguirEnvios = new SeguirEnvios();
123     this.pantallaSeguirEnvios.setVisible( b:true);
124     this.setVisible( b:false);
125 }
126
127 private void historialPedidosActionPerformed(java.awt.event.ActionEvent evt) {
128     pantallaHistorialEnvios = new HistorialEnvios();
129     this.pantallaHistorialEnvios.setVisible( b:true);
130     this.setVisible( b:false);
131 }
132
133
134
135 public static void crearCuatroProductosPorDefecto() {
136     if(arrayProductos.size() < 4){
137         Producto producto1 = new Producto( nombre:"Papas Fritas", precio:10);
138         Producto producto2 = new Producto( nombre:"Helado", precio:5);
139         Producto producto3 = new Producto( nombre:"Coca Cola", precio:3);
140         Producto producto4 = new Producto( nombre:"Papa de Agua", precio:2);
141         arrayProductos.add(producto1);
142         arrayProductos.add(producto2);
143         arrayProductos.add(producto3);
144         arrayProductos.add(producto4);
145     }
146 }
```

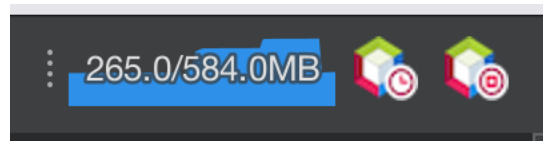
El código fuente del proyecto tiene un peso aproximado de 774 KB



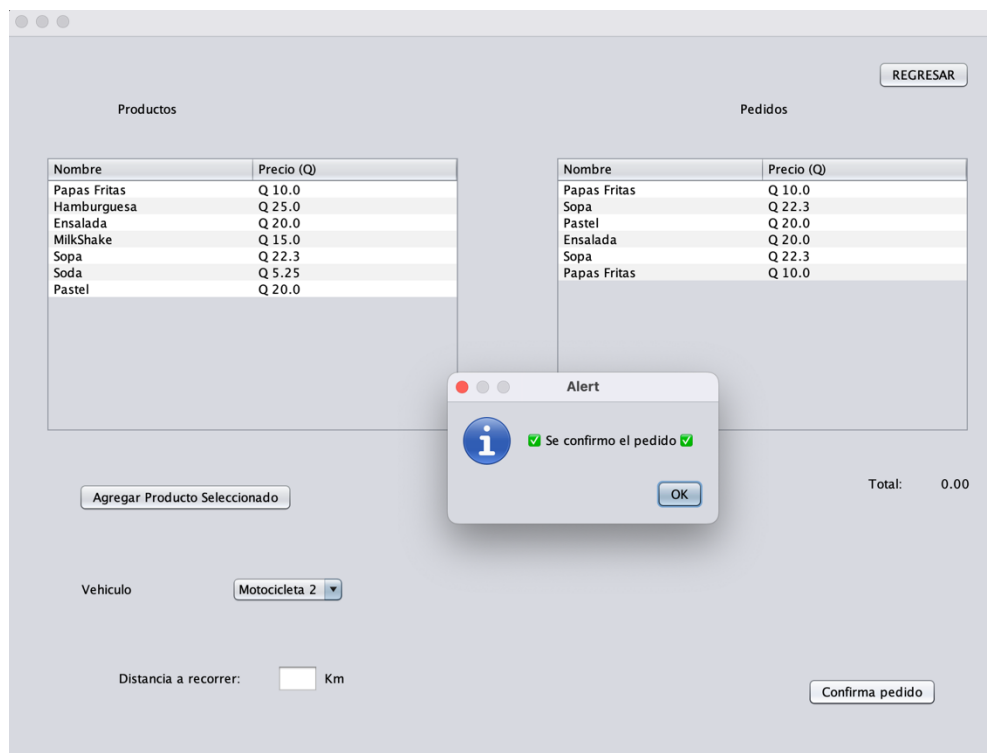
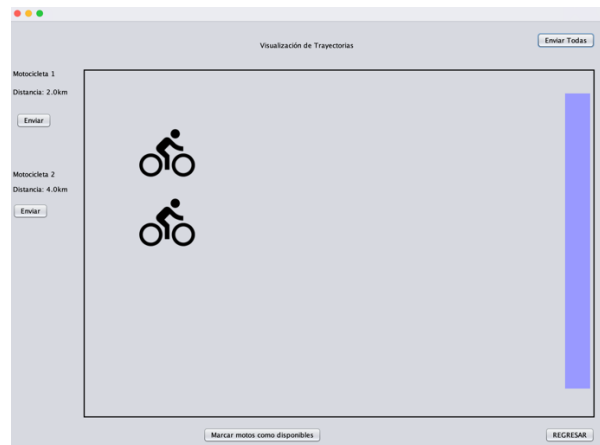
Se realizó un diagrama de clases que representa el flujo de las clases y sus métodos que dan funcionalidad a toda la aplicación, así como las clases que se encargan de la interfaz que esta utiliza.



Al ejecutarse el programa la memoria que ocupa es de menos de 265.0 MB, existe aspectos para poder optimizarlo con el fin de reducir estos recursos



El programa se desarrolló con el uso de una interfaz gráfica a diferencia de la primera práctica la cual solo uso la consola como medio de interacción con el usuario. Desde un simple Log-in hasta manejo de tablas



Se implementaron ciertas librerías para el desarrollo de la aplicación, como por ejemplo las librerías awt, ArrayList, swing y io.FileInputStream

```
package pantallas;

import clases.Hilo;
import clases.Pedido;
import java.io.FileInputStream;
import java.io.IOException;
import java.io.ObjectInputStream;
import java.util.ArrayList;
import javax.swing.table.DefaultTableModel;

public class HistorialEnvios extends javax.swing.JFrame {
```

Se implementaron los conceptos de modularidad y el paradigma de programación de “Programación Orientada a Objetos”

