Views

JFrame
// External Swing component

AbstractBorder
// External Swing component

RoundedBorder
- radius: int

+ RoundedBorder(radius: int)
+ paintBorder(c: Component, g: Graphics,
x: int, y: int, width: int, height: int): void
+ getBorderInsets(c: Component): Insets
+ getBorderInsets(c: Component, insets: Insets): Insets

Gui
- SCREEN_WIDTH: int = 900
- SCREEN_HEIGHT: int = 600
- RESEARCH_PANEL_WIDTH: int = 350
- MIN_SCREEN_WIDTH: int = 600
- MIN_SCREEN_HEIGHT: int = 300
- PARIS_CENTER: Coordinate = new Coordinate(48.8566, 2.3522)
- DEFAULT_ZOOM: int = 10
- textStart: JScrollPane
- textEnd: JScrollPane
- contentPanel: JPanel
- textItineraryPanel: JPanel
- textItineraryScrollPane: JScrollPane
- contentScrollPane: JScrollPane
- mapViewer: JMapViewer
- researchButton: JButton
- distCheckBox: JCheckBox
- timeCheckBox: JCheckBox
- showAllLinesCheckBox: JCheckBox
- numLine: JScrollPane
- comboBoxHours: JComboBox<String>
- comboBoxMinutes: JComboBox<String>
- lineTypeDropdown: JComboBox<String>
- viewLineButton: JButton
- textColor: Color = new Color(11, 22, 44)
- bordeColor: Color = new Color(88, 88, 88)
- primaryBackgroundColor: Color = new Color(240, 240, 240)
- accentColor: Color = new Color(76, 175, 80)
- floatingWindow: JDialog

+ Gui()
- init(): void
+ isShowAllLinesSelected(): boolean
- createFloatingWindow(): void
+ toggleFloatingWindow(visible: boolean): void
+ toggleCheckmark(menuItem: JMenuItem): void
+ isCheckmarkEnabled(menuItem: JMenuItem): boolean
- createMapViewer(): JMapViewer
- createTextAreaInput(text: String): JScrollPane
- createTextAreaOutput(text: String): JTextArea
- createSearchButton(): JButton
+ addMarker(latitude: double, longitude: double, color: Color): void
+ displayTransportType(lines: ArrayList<Line>, type: TransportTypes): void
+ addPolygonLine(start: Coordinate, end: Coordinate, color: Color, isDash: boolean): void
+ cleanMap(): void
+ displayPath(segments: ArrayList<SegmentItineraire>): JPanel
+ getResourcePathForTransportTypeIcones(tty: TransportTypes): String
+ displayTextItinerary(segments: ArrayList<SegmentItineraire>): JPanel
+ getIconAndLine(subline: Subline): JPanel
+ displayLine(lines: ArrayList<Line>, type: TransportTypes, lineNum: String): void
- displaySpecificLine(lines: ArrayList<Line>, type: TransportTypes, lineNum: String): boolean
- addToggleFunctionality(jp: JPanel, timesPanel: JPanel): void
+ displayListOfStopDeparture(stop: Stop): JPanel
- drawLines(g2d: Graphics2D, points: List<Point>): void
+ getMenuItem(menuIndex: int, itemIndex: int): JMenuItem
+ launch(): void

JPanel
// External Swing component

SolidCircle
- color: Color

+ SolidCircle(color: Color)
# paintComponent(g: Graphics): void

SolidColoredLine
- color: Color

+ SolidColoredLine(color: Color)
# paintComponent(g: Graphics): void

DashedColoredLine
- color: Color

+ DashedColoredLine(color: Color)
# paintComponent(g: Graphics): void

SolidDownwardsArrow
- color: Color

+ SolidDownwardsArrow(color: Color)
# paintComponent(g: Graphics): void

ImagePanel
- image: BufferedImage

+ ImagePanel(imagePath: String)
# paintComponent(g: Graphics): void

ColoredMapPolygon
- color: Color

+ ColoredMapPolygon(start: Coordinate, end: Coordinate, middle: Coordinate, colo
+ paint(g: Graphics, points: List<Point>): void
- drawLines(g2d: Graphics2D, points: List<Point>): void

GrayDashedMapPolygon
- GRAY_COLOR: Color = Color.GRAY

+ GrayDashedMapPolygon(start: Coordinate, end: Coordinate, middle: Coordinate)
+ paint(g: Graphics, points: List<Point>): void

MapPolygonImpl
// External OpenStreetMap component
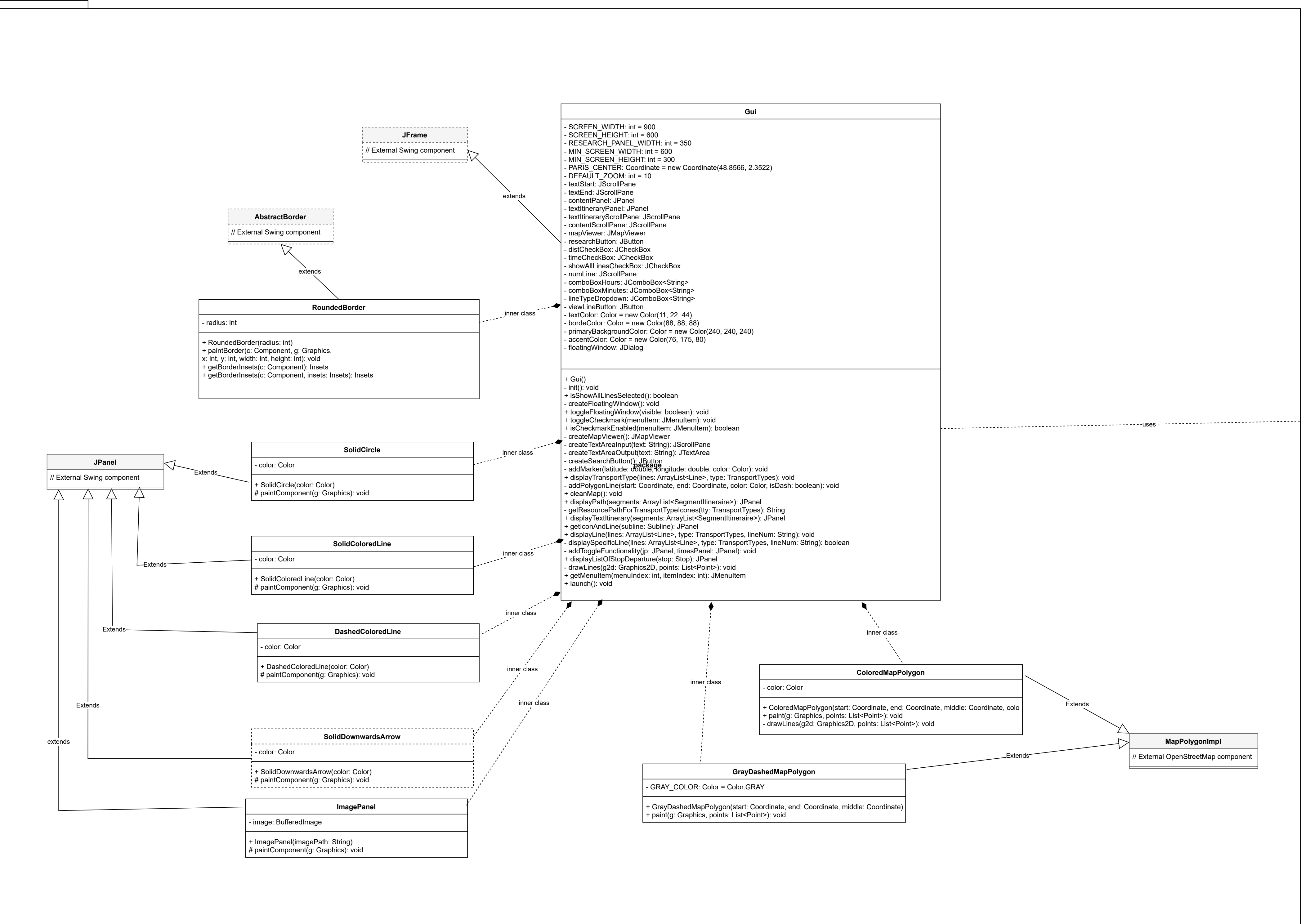
- SegmentItineraire
- Line
- Stop
- Subline
- TransportTypes

These are model classes from
the fr.u_paris.gla.project.graph
and fr.u_paris.gla.project.astar
packages.

extends
inner class
Extends
uses