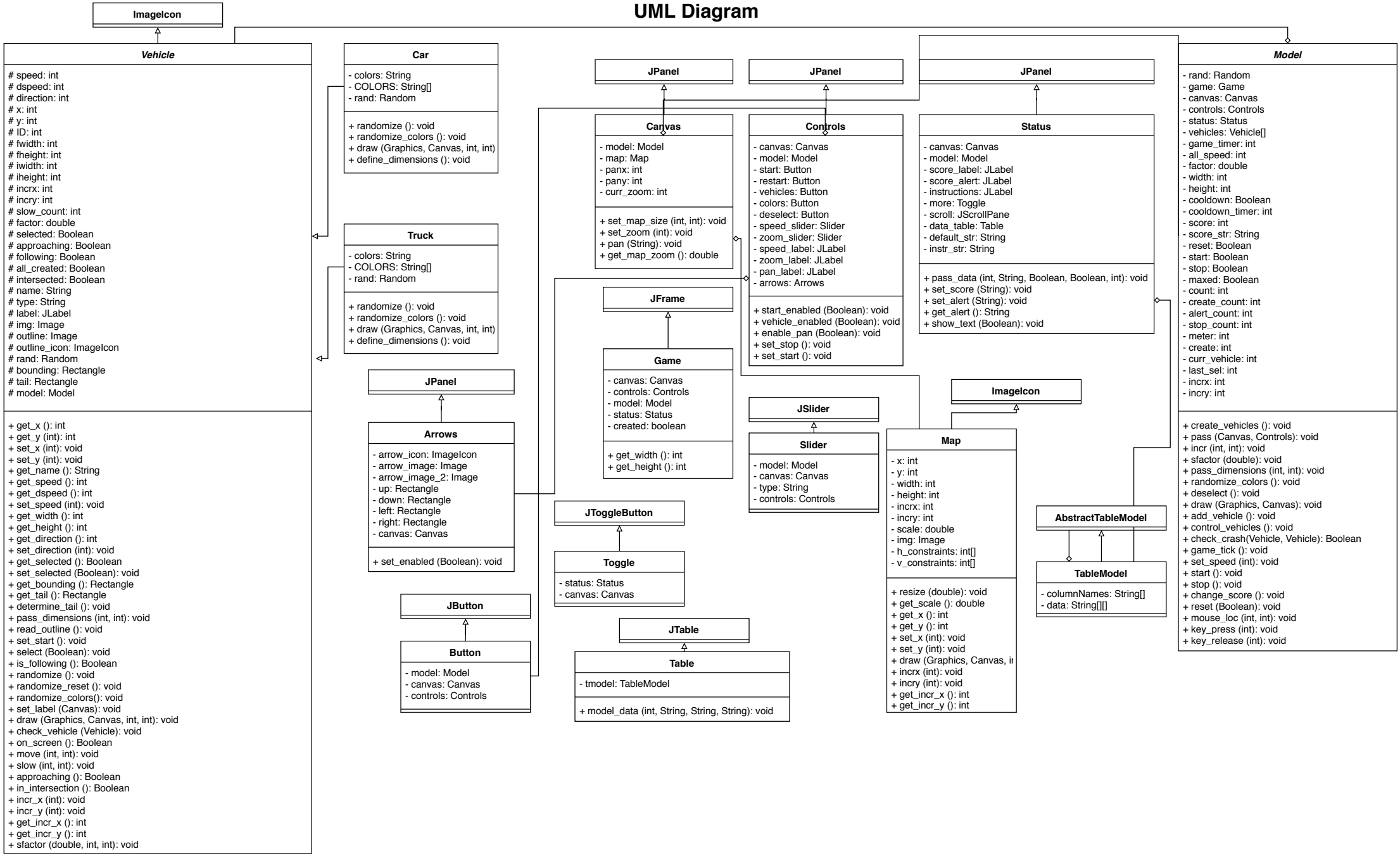
UML Diagram


ImageIcon
Vehicle
speed: int
dspeed: int
direction: int
x: int
y: int
ID: int
fwidth: int
fheight: int
iwidth: int
iheight: int
incrx: int
incry: int
slow_count: int
factor: double
selected: Boolean
approaching: Boolean
following: Boolean
all_created: Boolean
intersected: Boolean
name: String
type: String
label: JLabel
img: Image
outline: Image
outline_icon: ImageIcon
rand: Random
bounding: Rectangle
tail: Rectangle
model: Model
+ get_x (): int
+ get_y (int): int
+ set_x (int): void
+ set_y (int): void
+ get_name (): String
+ get_speed (): int
+ get_dspeed (): int
+ set_speed (int): void
+ get_width (): int
+ get_height (): int
+ get_direction (): int
+ set_direction (int): void
+ get_selected (): Boolean
+ set_selected (Boolean): void
+ get_bounding (): Rectangle
+ get_tail (): Rectangle
+ determine_tail (): void
+ pass_dimensions (int, int): void
+ read_outline (): void
+ set_start (): void
+ select (Boolean): void
+ is_following (): Boolean
+ randomize (): void
+ randomize_reset (): void
+ randomize_colors(): void
+ set_label (Canvas): void
+ draw (Graphics, Canvas, int, int): void
+ check_vehicle (Vehicle): void
+ on_screen (): Boolean
+ move (int, int): void
+ slow (int, int): void
+ approaching (): Boolean
+ in_intersection (): Boolean
+ incr_x (int): void
+ incr_y (int): void
+ get_incr_x (): int
+ get_incr_y (): int
+ sfactor (double, int, int): void
Car
- colors: String
- COLORS: String[]
- rand: Random
+ randomize (): void
+ randomize_colors (): void
+ draw (Graphics, Canvas, int, int)
+ define_dimensions (): void
Truck
- colors: String
- COLORS: String[]
- rand: Random
+ randomize (): void
+ randomize_colors (): void
+ draw (Graphics, Canvas, int, int)
+ define_dimensions (): void
JPanel
Arrows
- arrow_icon: ImageIcon
- arrow_image: Image
- arrow_image_2: Image
- up: Rectangle
- down: Rectangle
- left: Rectangle
- right: Rectangle
- canvas: Canvas
+ set_enabled (Boolean): void
JButton
Button
- model: Model
- canvas: Canvas
- controls: Controls
JPanel
Canvas
- model: Model
- map: Map
- panx: int
- pany: int
- curr_zoom: int
+ set_map_size (int, int): void
+ set_zoom (int): void
+ pan (String): void
+ get_map_zoom (): double
JFrame
Game
- canvas: Canvas
- controls: Controls
- model: Model
- status: Status
- created: boolean
+ get_width (): int
+ get_height (): int
JToggleButton
Toggle
- status: Status
- canvas: Canvas
JTable
Table
- tmodel: TableModel
+ model_data (int, String, String, String): void
JPanel
Controls
- canvas: Canvas
- model: Model
- start: Button
- restart: Button
- vehicles: Button
- colors: Button
- deselect: Button
- speed_slider: Slider
- zoom_slider: Slider
- speed_label: JLabel
- zoom_label: JLabel
- pan_label: JLabel
- arrows: Arrows
+ start_enabled (Boolean): void
+ vehicle_enabled (Boolean): void
+ enable_pan (Boolean): void
+ set_stop (): void
+ set_start (): void
JSlider
Slider
- model: Model
- canvas: Canvas
- type: String
- controls: Controls
JPanel
Status
- canvas: Canvas
- model: Model
- score_label: JLabel
- score_alert: JLabel
- instructions: JLabel
- more: Toggle
- scroll: JScrollPane
- data_table: Table
- default_str: String
- instr_str: String
+ pass_data (int, String, Boolean, Boolean, int): void
+ set_score (String): void
+ set_alert (String): void
+ get_alert (): String
+ show_text (Boolean): void
ImageIcon
Map
- x: int
- y: int
- width: int
- height: int
- incrx: int
- incry: int
- scale: double
- img: Image
- h_constraints: int[]
- v_constraints: int[]
+ resize (double): void
+ get_scale (): double
+ get_x (): int
+ get_y (): int
+ set_x (int): void
+ set_y (int): void
+ draw (Graphics, Canvas, i
+ incrx (int): void
+ incry (int): void
+ get_incr_x (): int
+ get_incr_y (): int
AbstractTableModel
TableModel
- columnNames: String[]
- data: String[][]
Model
- rand: Random
- game: Game
- canvas: Canvas
- controls: Controls
- status: Status
- vehicles: Vehicle[]
- game_timer: int
- all_speed: int
- factor: double
- width: int
- height: int
- cooldown: Boolean
- cooldown_timer: int
- score: int
- score_str: String
- reset: Boolean
- start: Boolean
- stop: Boolean
- maxed: Boolean
- count: int
- create_count: int
- alert_count: int
- stop_count: int
- meter: int
- create: int
- curr_vehicle: int
- last_sel: int
- incrx: int
- incry: int
+ create_vehicles (): void
+ pass (Canvas, Controls): void
+ incr (int, int): void
+ sfactor (double): void
+ pass_dimensions (int, int): void
+ randomize_colors (): void
+ deselect (): void
+ draw (Graphics, Canvas): void
+ add_vehicle (): void
+ control_vehicles (): void
+ check_crash(Vehicle, Vehicle): Boolean
+ game_tick (): void
+ set_speed (int): void
+ start (): void
+ stop (): void
+ change_score (): void
+ reset (Boolean): void
+ mouse_loc (int, int): void
+ key_press (int): void
+ key_release (int): void