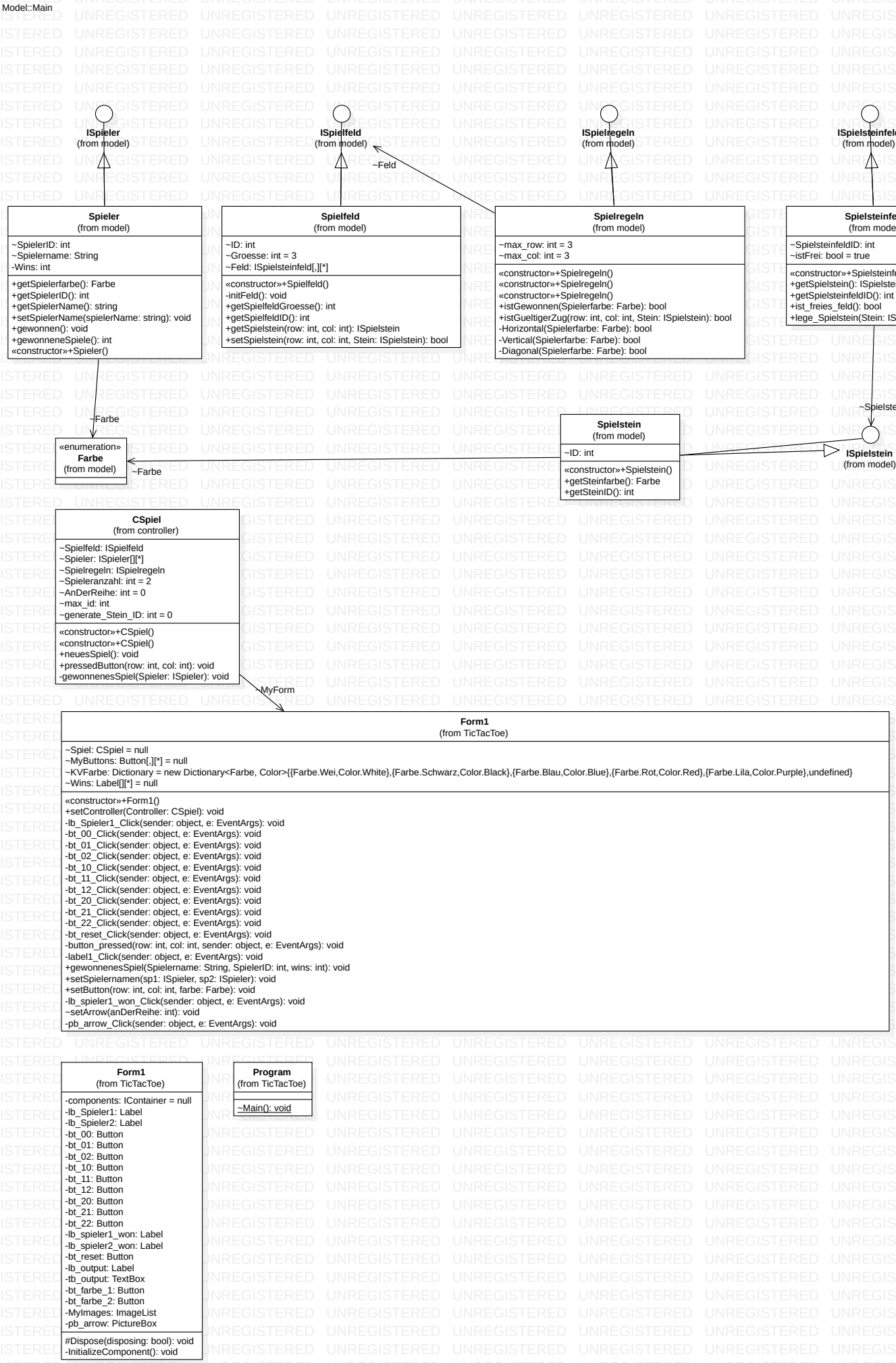
Model::Main
ISpieler
(from model)
ISpielfeld
(from model)
ISpielregeln
(from model)
ISpielsteinfeld
(from model)
~Feld
Spieler
(from model)
~SpielerID: int
~Spielername: String
-Wins: int
+getSpielerfarbe(): Farbe
+getSpielerID(): int
+getSpielerName(): string
+setSpielerName(spielerName: string): void
+gewonnen(): void
+gewonneneSpiele(): int
«constructor»+Spieler()
Spielfeld
(from model)
~ID: int
~Groesse: int = 3
~Feld: ISpielsteinfeld[,][*]
«constructor»+Spielfeld()
-initFeld(): void
+getSpielfeldGroesse(): int
+getSpielfeldID(): int
+getSpielstein(row: int, col: int): ISpielstein
+setSpielstein(row: int, col: int, Stein: ISpielstein): bool
Spielregeln
(from model)
~max_row: int = 3
~max_col: int = 3
«constructor»+Spielregeln()
«constructor»+Spielregeln()
«constructor»+Spielregeln()
+istGewonnen(Spielerfarbe: Farbe): bool
+istGueltigerZug(row: int, col: int, Stein: ISpielstein): bool
-Horizontal(Spielerfarbe: Farbe): bool
-Vertical(Spielerfarbe: Farbe): bool
-Diagonal(Spielerfarbe: Farbe): bool
Spielsteinfeld
(from model)
~SpielsteinfeldID: int
~istFrei: bool = true
«constructor»+Spielsteinfeld()
+getSpielstein(): ISpielstein
+getSpielsteinfeldID(): int
+ist_freies_feld(): bool
+lege_Spielstein(Stein: ISpielstein): bool
~Farbe
~Spielstein
«enumeration»
Farbe
(from model)
~Farbe
Spielstein
(from model)
~ID: int
«constructor»+Spielstein()
+getSteinfarbe(): Farbe
+getSteinID(): int
ISpielstein
(from model)
CSpiel
(from controller)
~Spielfeld: ISpielfeld
~Spieler: ISpieler[][*]
~Spielregeln: ISpielregeln
~Spieleranzahl: int = 2
~AnDerReihe: int = 0
~max_id: int
~generate_Stein_ID: int = 0
«constructor»+CSpiel()
«constructor»+CSpiel()
+neuesSpiel(): void
+pressedButton(row: int, col: int): void
-gewonnenesSpiel(Spieler: ISpieler): void
~MyForm
Form1
(from TicTacToe)
~Spiel: CSpiel = null
~MyButtons: Button[,][*] = null
~KVFarbe: Dictionary = new Dictionary<Farbe, Color>{{Farbe.Wei,Color.White},{Farbe.Schwarz,Color.Black},{Farbe.Blau,Color.Blue},{Farbe.Rot,Color.Red},{Farbe.Lila,Color.Purple},undefined}
~Wins: Label[][*] = null
«constructor»+Form1()
+setController(Controller: CSpiel): void
-lb_Spieler1_Click(sender: object, e: EventArgs): void
-bt_00_Click(sender: object, e: EventArgs): void
-bt_01_Click(sender: object, e: EventArgs): void
-bt_02_Click(sender: object, e: EventArgs): void
-bt_10_Click(sender: object, e: EventArgs): void
-bt_11_Click(sender: object, e: EventArgs): void
-bt_12_Click(sender: object, e: EventArgs): void
-bt_20_Click(sender: object, e: EventArgs): void
-bt_21_Click(sender: object, e: EventArgs): void
-bt_22_Click(sender: object, e: EventArgs): void
-bt_reset_Click(sender: object, e: EventArgs): void
-button_pressed(row: int, col: int, sender: object, e: EventArgs): void
-label1_Click(sender: object, e: EventArgs): void
+gewonnenesSpiel(Spielername: String, SpielerID: int, wins: int): void
+setSpielernamen(sp1: ISpieler, sp2: ISpieler): void
+setButton(row: int, col: int, farbe: Farbe): void
-lb_spieler1_won_Click(sender: object, e: EventArgs): void
~setArrow(anDerReihe: int): void
-pb_arrow_Click(sender: object, e: EventArgs): void
Form1
(from TicTacToe)
-components: IContainer = null
-lb_Spieler1: Label
-lb_Spieler2: Label
-bt_00: Button
-bt_01: Button
-bt_02: Button
-bt_10: Button
-bt_11: Button
-bt_12: Button
-bt_20: Button
-bt_21: Button
-bt_22: Button
-lb_spieler1_won: Label
-lb_spieler2_won: Label
-bt_reset: Button
-lb_output: Label
-tb_output: TextBox
-bt_farbe_1: Button
-bt_farbe_2: Button
-MyImages: ImageList
-pb_arrow: PictureBox
#Dispose(disposing: bool): void
-InitializeComponent(): void
Program
(from TicTacToe)
~Main(): void