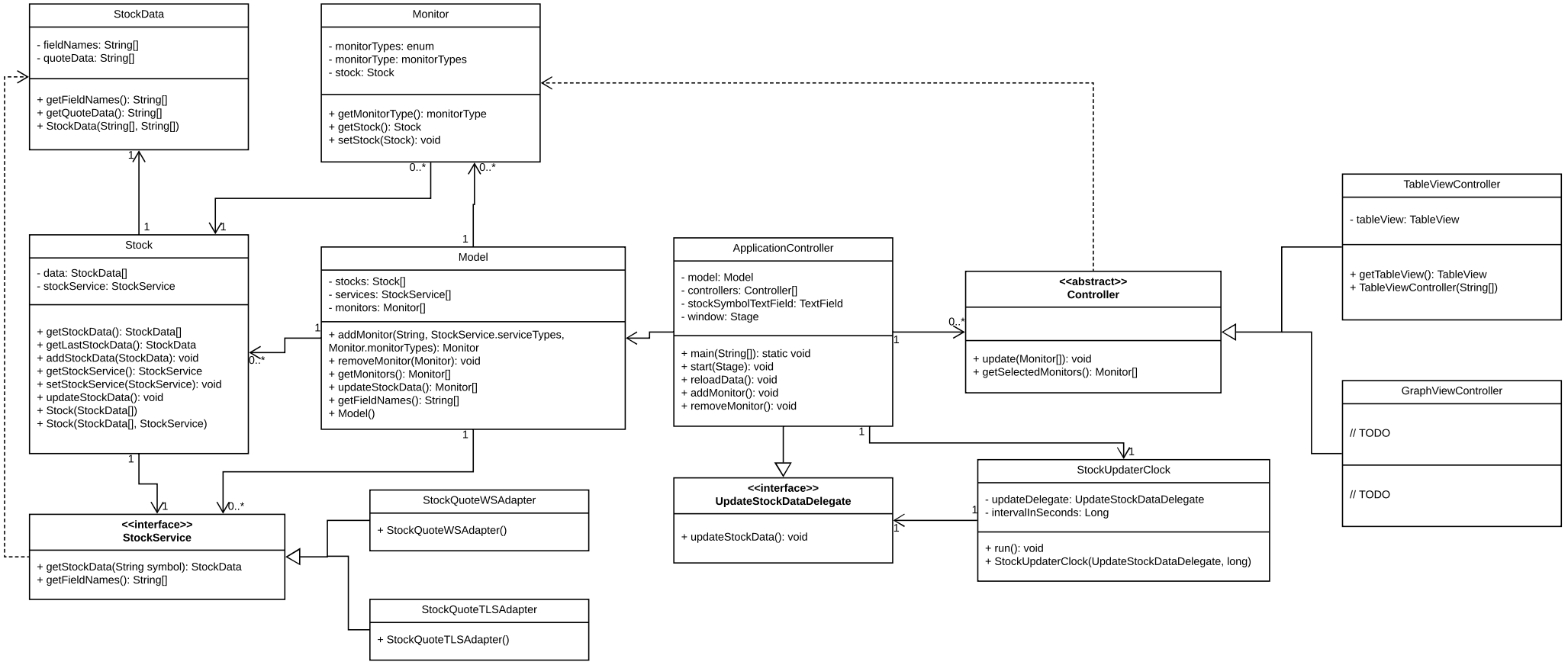
StockData
- fieldNames: String[]
- quoteData: String[]
+ getFieldNames(): String[]
+ getQuoteData(): String[]
+ StockData(String[], String[])
Monitor
- monitorTypes: enum
- monitorType: monitorTypes
- stock: Stock
+ getMonitorType(): monitorType
+ getStock(): Stock
+ setStock(Stock): void
Stock
- data: StockData[]
- stockService: StockService
+ getStockData(): StockData[]
+ getLastStockData(): StockData
+ addStockData(StockData): void
+ getStockService(): StockService
+ setStockService(StockService): void
+ updateStockData(): void
+ Stock(StockData[])
+ Stock(StockData[], StockService)
Model
- stocks: Stock[]
- services: StockService[]
- monitors: Monitor[]
+ addMonitor(String, StockService.serviceTypes, Monitor.monitorTypes): Monitor
+ removeMonitor(Monitor): void
+ getMonitors(): Monitor[]
+ updateStockData(): Monitor[]
+ getFieldNames(): String[]
+ Model()
ApplicationController
- model: Model
- controllers: Controller[]
- stockSymbolTextField: TextField
- window: Stage
+ main(String[]): static void
+ start(Stage): void
+ reloadData(): void
+ addMonitor(): void
+ removeMonitor(): void
<<abstract>>
Controller
+ update(Monitor[]): void
+ getSelectedMonitors(): Monitor[]
TableViewController
- tableView: TableView
+ getTableView(): TableView
+ TableViewController(String[])
GraphViewController
// TODO
// TODO
<<interface>>
StockService
+ getStockData(String symbol): StockData
+ getFieldNames(): String[]
StockQuoteWSAdapter
+ StockQuoteWSAdapter()
StockQuoteTLSAdapter
+ StockQuoteTLSAdapter()
<<interface>>
UpdateStockDataDelegate
+ updateStockData(): void
StockUpdaterClock
- updateDelegate: UpdateStockDataDelegate
- intervalInSeconds: Long
+ run(): void
+ StockUpdaterClock(UpdateStockDataDelegate, long)
1
0..*