**UML uitleg**

**Design patterns**

De volgende design patterns zijn geïmplementeerd in HotPinkPong.

**Factory en Visitor**

Via de Option Interface is de Visitor design pattern geïmplementeerd. De Visitor gaat bij elke Entity langs en haalt de attributes op. De entities worden door de EntityFactory aangemaakt en door de EntityConstructor in de List entities gezet. De entities zijn Ball, Paddle, Borderline, ScoreLeft en ScoreRight. Om de entities te updaten en te tekenen gebruiken we de Visitor. Dit gaat doormiddel van de interface IDrawVisitor en DefaultDrawVisitor. Aan de andere kant hebben we IUpdateable, IUpdateVisitor en DefaultUpdateVisitor.

**Iterator**

De Iterator is een interface die wordt geïmplementeerd door List. Deze gaat langs elke Entity in de lijst en werkt daarmee samen met de Visitor.

**Adapter**

Er zijn vier verschillende adapters om de applicatie cross-platform te maken. Alle code die hierboven is beschreven is SharedLogic. Daarnaast hebben we de Monogame- en SFMLLogic. De SFMLLogic bestaat uit een SFMLDrawingAdapter die ervoor zorgt dat de desktopapplicatie kan tekenen met SFML. De MonogameDrawingAdapter wordt gebruikt door de MonogameLogic om de applicatie op een Android platform te tekenen.