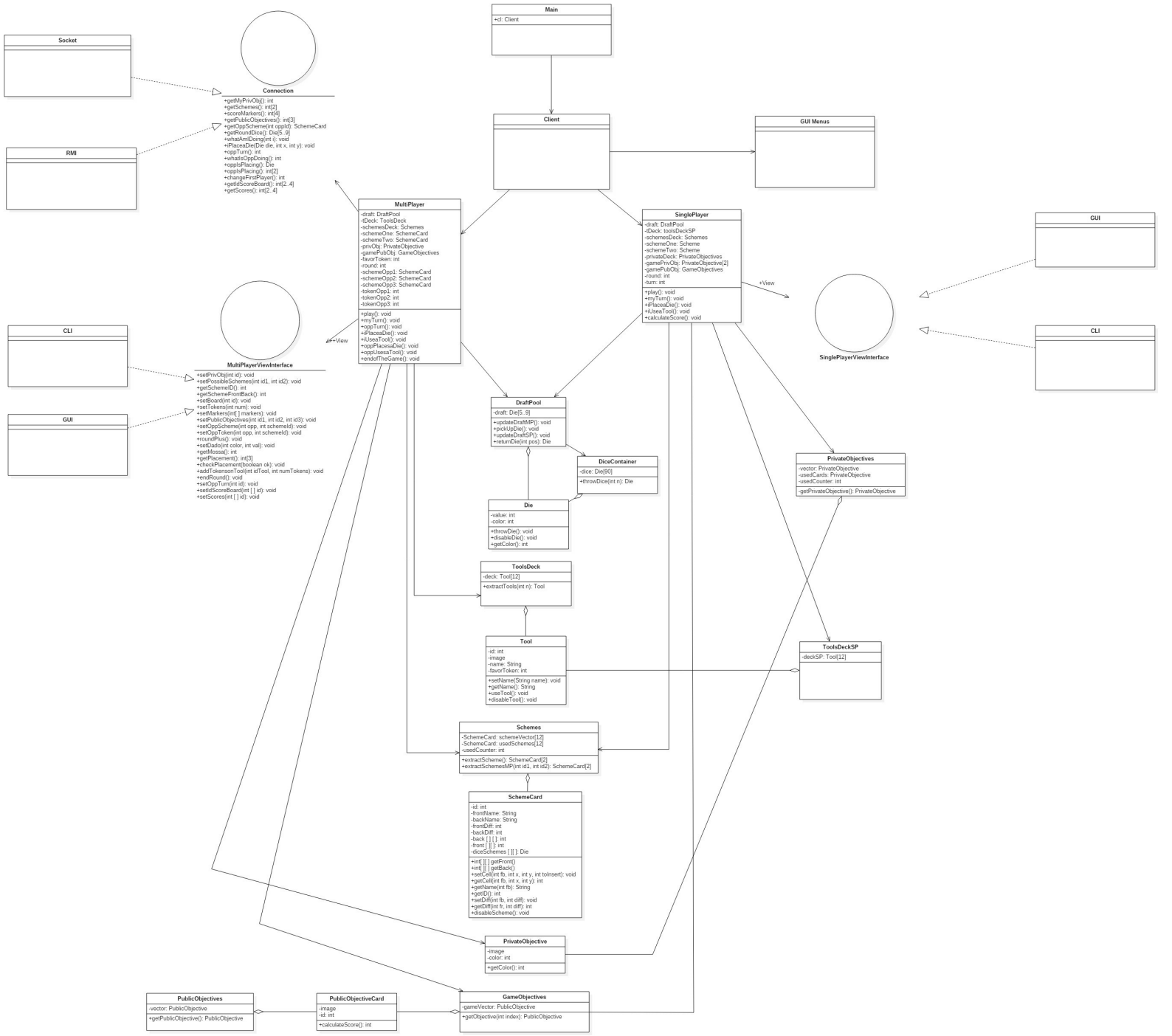
Main
+cl: Client
Socket
Connection
+getMyPrivObj(): int
+getSchemes(): int[2]
+scoreMarkers(): int[4]
+getPublicObjectives(): int[3]
+getOppScheme(int oppId): SchemeCard
+getRountDice(): Die[5..9]
+whatAmIDoing(int i): void
+iPlaceaDie(Die die, int x, int y): void
+oppTurn(): int
+whatIsOppDoing(): int
+oppIsPlacing(): Die
+oppIsPlacing(): int[2]
+changeFirstPlayer(): int
+getIdScoreBoard(): int[2..4]
+getScores(): int[2..4]
RMI
Client
GUI Menus
MultiPlayer
-draft: DraftPool
-tDeck: ToolsDeck
-schemesDeck: Schemes
-schemeOne: SchemeCard
-schemeTwo: SchemeCard
-privObj: PrivateObjective
-gamePubObj: GameObjectives
-favorToken: int
-round: int
-schemeOpp1: SchemeCard
-schemeOpp2: SchemeCard
-schemeOpp3: SchemeCard
-tokenOpp1: int
-tokenOpp2: int
-tokenOpp3: int
+play(): void
+myTurn(): void
+oppTurn(): void
+iPlaceaDie(): void
+iUseaTool(): void
+oppPlacesaDie(): void
+oppUsesaTool(): void
+endofTheGame(): void
+View
SinglePlayer
-draft: DraftPool
-tDeck: toolsDeckSP
-schemesDeck: Schemes
-schemeOne: Scheme
-schemeTwo: Scheme
-privateDeck: PrivateObjectives
-gamePrivObj: PrivateObjective[2]
-gamePubObj: GameObjectives
-round: int
-turn: int
+play(): void
+myTurn(): void
+iPlaceaDie(): void
+iUseaTool(): void
+calculateScore(): void
+View
GUI
CLI
SinglePlayerViewInterface
CLI
GUI
MultiPlayerViewInterface
+setPrivObj(int id): void
+setPossibleSchemes(int id1, int id2): void
+getSchemeID(): int
+getSchemeFrontBack(): int
+setBoard(int id): void
+setTokens(int num): void
+setMarkers(int[ ] markers): void
+setPublicObjectives(int id1, int id2, int id3): void
+setOppScheme(int opp, int schemeId): void
+setOppToken(int opp, int schemeId): void
+roundPlus(): void
+setDado(int color, int val): void
+getMossa(): int
+getPlacement(): int[3]
+checkPlacement(boolean ok): void
+addTokensonTool(int idTool, int numTokens): void
+endRound(): void
+setOppTurn(int id): void
+setIdScoreBoard(int [ ] id): void
+setScores(int [ ] id): void
DraftPool
-draft: Die[5..9]
+updateDraftMP(): void
+pickUpDie(): void
+updateDraftSP(): void
+returnDie(int pos): Die
DiceContainer
-dice: Die[90]
+throwDice(int n): Die
Die
-value: int
-color: int
+throwDie(): void
+disableDie(): void
+getColor(): int
PrivateObjectives
-vector: PrivateObjective
-usedCards: PrivateObjective
-usedCounter: int
-getPrivateObjective(): PrivateObjective
ToolsDeck
-deck: Tool[12]
+extractTools(int n): Tool
Tool
-id: int
-image
-name: String
-favorToken: int
+setName(String name): void
+getName(): String
+useTool(): void
+disableTool(): void
ToolsDeckSP
-deckSP: Tool[12]
Schemes
-SchemeCard: schemeVector[12]
-SchemeCard: usedSchemes[12]
-usedCounter: int
+extractScheme(): SchemeCard[2]
+extractSchemesMP(int id1, int id2): SchemeCard[2]
SchemeCard
-id: int
-frontName: String
-backName: String
-frontDiff: int
-backDiff: int
-back [ ] [ ]: int
-front [ ] [ ]: int
-diceSchemes [ ][ ]: Die
+int[ ][ ] getFront()
+int[ ][ ] getBack()
+setCell(int fb, int x, int y, int toInsert): void
+getCell(int fb, int x, int y): int
+getName(int fb): String
+getID(): int
+setDiff(int fb, int diff): void
+getDiff(int fr, int diff): int
+disableScheme(): void
PrivateObjective
-image
-color: int
+getColor(): int
PublicObjectives
-vector: PublicObjective
+getPublicObjective(): PublicObjective
PublicObjectiveCard
-image
-id: int
+calculateScore(): int
GameObjectives
-gameVector: PublicObjective
+getObjective(int index): PublicObjective