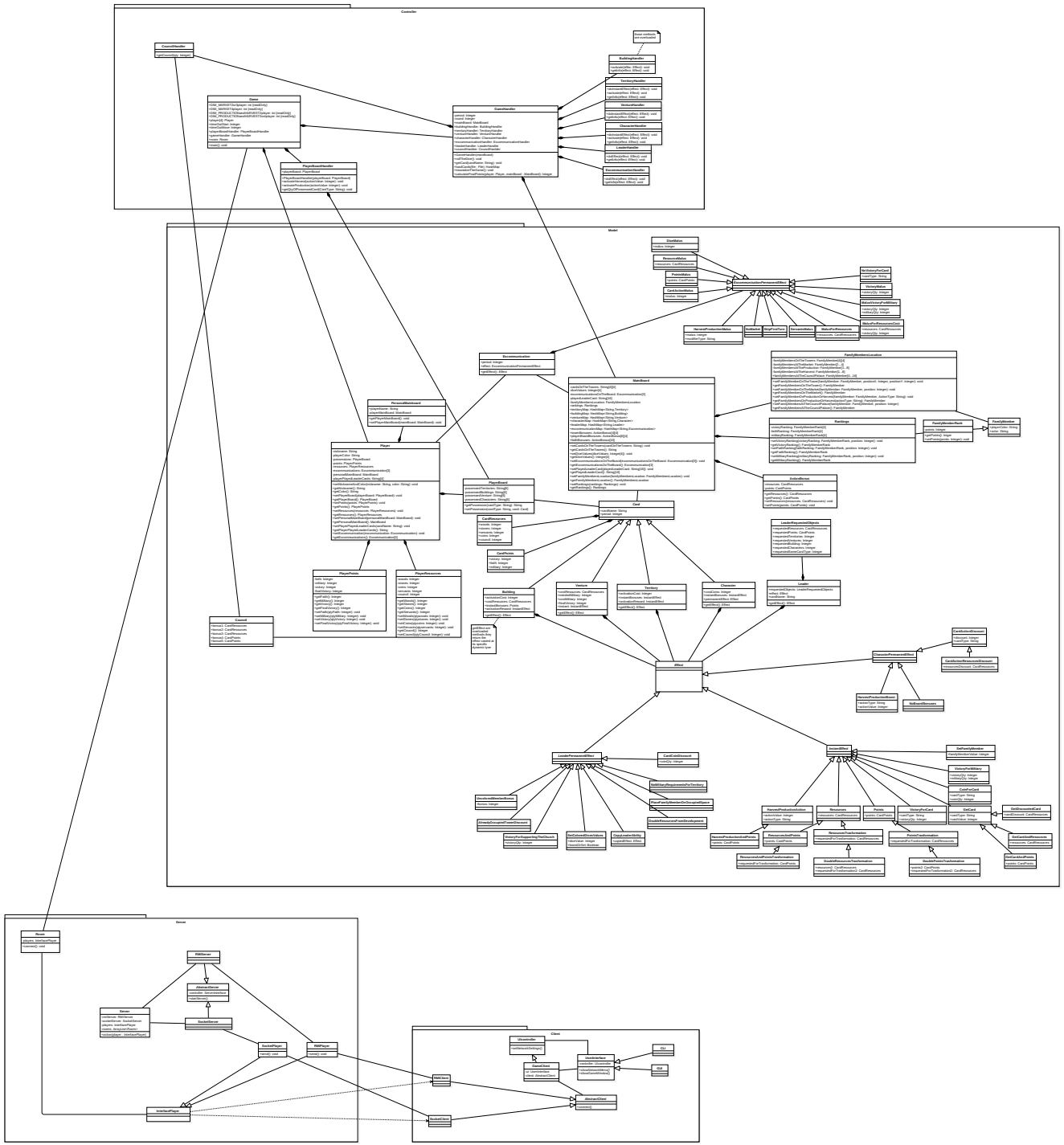
Controller
GameHandler
getGameState: Integer
Game
+DISC_MINISTERSplayer: int[maxDisc]
+DISC_MINISTERSpos: int[maxDisc]
+DISC_PRODUCTIONandMINISTERSplayer: int[maxDisc]
+DISC_PRODUCTIONandMINISTERSpos: int[maxDisc]
+players[4]: Player
+timerDuration: Integer
+currentPlayer: Integer
+playerBoardHandler: PlayerBoardHandler
+gameHandler: GameHandler
+init(): Room
+reset(): void
PlayerBoardHandler
+playerBoard: PlayerBoard
+*PlayerBoardHandler(playerBoard: PlayerBoard)
+addPlayerToBoard(player: Player): void
+actionProduction(actionValue: Integer): void
+getQuiDiPresenteCard(cardType: String): void
GameHandler
period: Integer
round: Integer
mainBoard: MainBoard
+buildingHandler: BuildingHandler
+tenteringHandler: TenteringHandler
+ventureHandler: VentureHandler
+characterHandler: CharacterHandler
+excommunicationHandler: ExcommunicationHandler
+leaderHandler: LeaderHandler
+recommendationHandler: RecommendationHandler
+*GameHandler(mainBoard)
+initTheBoard(): void
+getCardsonRoster(s: String): void
+loadCardsOn_Flat(): HashMap
+initPlayers(): void
+calculateFinalPoints(player: Player, mainBoard: MainBoard): Integer
BuildingHandler
+activateEffect: Effect): void
+getEffect: Effect): void
TenteringHandler
+activateEffect(effect: Effect): void
+getEffect(effect: Effect): void
+getAllEffect: Effect): void
VentureHandler
+activateEffect(effect: Effect): void
+getEffect(effect: Effect): void
CharacterHandler
+activateEffect(effect: Effect): void
+getEffect(effect: Effect): void
+getAllEffect: Effect): void
LeaderHandler
+initEffect(effect: Effect): void
+getEffect(effect: Effect): void
RecommendationHandler
+initEffect(effect: Effect): void
+getEffect(effect: Effect): void
Model
DiscStatus
+index: Integer
ResourceMalus
+resources: CardResources
PointsMalus
+points: CardPoints
CardOrderMalus
+index: Integer
ExcommunicationPenaltyEffect
HarvestProductionMalus
+index: Integer
+leaderType: String
SetBonus
RightTurnTiles
ExcommMalus
MakeForResources
+resource: CardResources
NetVictoryPerCard
+cardType: String
VictoryMalus
+victoryQty: Integer
MaterialVictoryFaithMilitary
+victoryQty: Integer
+victoryQty: Integer
MaterialPerResourceVictory
+resources: CardResources
+victoryQty: Integer
Recommendation
+period: Integer
+effect: ExcommunicationPenaltyEffect
+getEffect(): Effect
MainBoard
+cardsOnTheTowers: String[][4]
+discMalus: Integer[3]
+excommunicationOnTheBoard: Excommunication[2]
+familyMembersOnTheBoard: FamilyMembersLocation
+buildings: Rankings
+tenteringTiles: HashMap<String,Tenteringt>
+buildingTiles: HashMap<String,Building>
+ventureMap: HashMap<String,Venture>
+characterMap: HashMap<String,Character>
+leaderMap: HashMap<String,Leader>
+excommunicationMap: HashMap<String,Excommunication>
+haveBonusin: ActionBonus[4]
+haveBonusOut: ActionBonus[4]
+taskBonuses: ActionBonus[12]
+initCardsOnTheTowers(cardsOnTheTowers: String): void
+getCardsOnTheTowers(): String
+initDiscMalus(discMalus: Integer[3]): void
+getDiscMalus(): Integer[3]
+initExcommunicationsOnTheBoard(excommunicationsOnTheBoard: Excommunication[2]): void
+getExcommunicationsOnTheBoard(): Excommunication[2]
+initTheFamilyMembersOnTheBoard(familyMembersOnTheBoard: FamilyMembersLocation): void
+getTheFamilyMembersOnTheBoard(): FamilyMembersLocation
+initRankings(rankings: Rankings): void
+getRankings(): Rankings
FamilyMembersLocation
[family]MembersOnTheTowers: FamilyMember[4][4]
[family]MembersOnTheBoard: FamilyMember[2]
[family]MembersOnTheProduction: FamilyMember[2...8]
[family]MembersOnTheHarvest: FamilyMember[2...8]
[family]MembersOnTheCouncilPalace: FamilyMember[2...12]
+setFamilyMembersOnTheTowers(family[Member: FamilyMember, positionT: Integer, positionT: Integer): void
+getFamilyMembersOnTheTowers(): void
+setFamilyMembersOnTheBoard(family[Member: FamilyMember, position: Integer): void
+getFamilyMembersOnTheBoard(): FamilyMember
+setFamilyMembersOnTheProduction(family[Member: FamilyMember, actionType: String): FamilyMember
+setFamilyMembersOnTheHarvest(family[Member: FamilyMember, position: Integer): void
+setFamilyMembersOnTheCouncilPalace(): FamilyMember
Rankings
victoryRanking: FamilyMembersRank[5]
militaryRanking: FamilyMembersRank[5]
faithRanking: FamilyMembersRank[5]
+setVictoryRanking(victoryRanking: FamilyMembersRank, position: Integer): void
+getVictoryRanking(): FamilyMembersRank
+setFaithRanking(faithRanking: FamilyMembersRank, position: Integer): void
+getFaithRanking(): FamilyMembersRank
+setMilitaryRanking(militaryRanking: FamilyMembersRank, position: Integer): void
+getMilitaryRanking(): FamilyMembersRank
FamilyMembersRank
+points: Integer
+position: Integer
+setPoints(): Integer
+setPoints(points: Integer): void
FamilyMember
+playerColor: String
+color: String
PersonalBoard
+playerBoard: PlayerBoard
+*PersonalBoard(playerBoard: PlayerBoard)
+getPlayerBoard(): void
+setPlayerBoard(playerBoard: MainBoard): void
Player
+nickname: String
+playerColor: String
+personalBoard: PlayerBoard
+familyMembers: FamilyMember[4]
+resources: PlayerResources
+excommunications: Excommunication[3]
+actualPlayerPoints: PlayerPoints
+playerPlayedLeaderCards: String[4]
+setNickname(nickname: String): void
+setPlayerColor(playerColor: String, color: String): void
+getColor(): String
+setPlayerBoard(playerBoard: PlayerBoard): void
+getPlayerBoard(): PlayerBoard
+setPlayerPoints(points: PlayerPoints): void
+getPoints(): PlayerPoints
+setPlayerResources(resources: PlayerResources): void
+getPlayerResources(): PlayerResources
+setPersonalBoardBeforeEffectsAfterMainBoard(): MainBoard: void
+getPersonalBoard(): PersonalBoard
+setPlayerPlayedLeaderCardsOrderDiscard(card: String): void
+getPlayerPlayedLeaderCards(): String
+setExcommunications(excommunicationsNumber: Excommunication): void
+getExcommunications(): Excommunication[3]
PlayerBoard
+personalDiscMalus: Integer[3]
+personalCardPoints: String[3]
+personalDiscount: String[3]
+personalCharacters: String[3]
+getPlacementsCard(cardType: String): String
+setPlacementsCard(cardType: String, card: Card): void
Card
+cardName: String
+period: Integer
CardResources
+coins: Integer
+stones: Integer
+servants: Integer
+wood: Integer
+coins: Integer
+stone: Integer
+servants: Integer
+wood: Integer
CardPoints
+victory: Integer
+faith: Integer
+military: Integer
ActionBonus
+resources: CardResources
+points: CardPoints
+getResources(): CardResources
+setResources(): void
+setPlacementsCardsBonus: CardResources): void
+setPoints(points: CardPoints): void
LeaderRequirementSignals
+requiredPlaceresources: CardResources
+requiredPlaceCards: Integer
+requiredTerritories: Integer
+requiredUnitsports: Integer
+requiredCoins: Integer
+requiredPlacepoints: Integer
+requiredStoneCardType: Integer
PlayerPoints
faith: Integer
military: Integer
victory: Integer
totalVictory: Integer
+getFaithQ(): Integer
+getMilitaryQ(): Integer
+getVictoryQ(): Integer
+getVictoryQ(): Integer
+getTotalVictory(): Integer
+setFaithQ(faithQty: Integer): void
+setMilitaryQ(militaryQty: Integer): void
+setVictoryQ(victoryQty: Integer): void
+setTotalVictory(totalTotalVictory: Integer): void
PlayerResources
wounds: Integer
stones: Integer
coins: Integer
servants: Integer
coins: Integer
stone: Integer
servants: Integer
wood: Integer
+getCoins(): Integer
+getStones(): Integer
+getCoins(): Integer
+getServants(): Integer
+setWoods(woods: Integer): void
+setStones(stones: Integer): void
+setCoins(coins: Integer): void
+setServants(servants: Integer): void
+getCoins(): Integer
+setCoins(coins: Integer): void
Council
+Name1: CardResources
+Name2: CardResources
+Name3: CardPoints
+Name4: CardPoints
Building
+activationCost: Integer
+costResources: CardResources
+resourcesBonuses: Points
+costAlternativeResources: InstantEffect
+getEffect(): Effect
Venture
+costResources: CardResources
+costMaleMilitary: Integer
+costFaithQty: Integer
+finalVictory: Integer
+instant: InstantEffect
+getEffect(): Effect
Territory
+activationCost: Integer
+costResources: InstantEffect
+activationBonus: InstantEffect
+getEffect(): Effect
Character
+costCoins: Integer
+costResources: InstantEffect
+activationBonus: InstantEffect
+getEffect(): Effect
Leader
+requiredObjects: LeaderPlacementObjects
+value: Integer
+cardGamer: String
+getEffect(): Effect
getEffect.or
getEffect(effect: String, effect cancel,at effect, dynamic type
Effect
CharacterPermanentEffect
CardAlternativeDiscount
+discount: Integer
+cardType: String
CardSubstituteResourceDiscount
+resourceDiscount: CardResources
HarvestProductionBonus
+leaderType: String
+actionValue: Integer
SetBonuses
LeaderPermanentEffect
CardCostDiscount
+cardQty: Integer
UncoloredMinisterBonus
+Bonus: Integer
SetMilitaryRequirementForTerritory
PlacePartyMembersOnOccupiedSpace
AlreadyOccupiedTowerBonus
DoubleResourceFromOccupiedSpace
VictoryPlacepoppingTheChurch
+victoryQty: Integer
SetColoredDiscsValue
+discValue: Integer
+bonusColor: Boolean
CopyLeaderAbility
+copiedAbility: Effect
InstantEffect
SetFamilyMember
+familyMemberValue: Integer
VictoryFaithMilitary
+victoryQty: Integer
+victoryQty: Integer
CardPerCard
+cardType: String
+victoryQty: Integer
HarvestProductionValue
+actionValue: Integer
+leaderType: String
Resources
+resources: CardResources
Points
+points: CardPoints
VictoryPerCard
+cardType: String
+victoryQty: Integer
GetCard
+cardType: String
+cardValue: Integer
GetDiscountedCard
+cardDiscount: CardResources
HarvestProductionPoints
+points: CardPoints
ResourceAndPoints
+points: CardPoints
ResourcesTransformation
+requiredForTransformation: CardResources
PointsTransformation
+requiredForTransformation: CardResources
ResourceAndPointsTransformation
+requiredForTransformation: CardPoints
DoubleResourcesTransformation
+resources2: CardResources
+requiredForTransformation1: CardResources
DoublePointsTransformation
+points2: CardPoints
+requiredForTransformation2: CardResources
GetCardAndPoints
+points: CardPoints
Server
Room
+players: InterfacePlayer
+*room(): void
RMIServer
SocketServer
+socketPort: SocketServer
+startServer()
Server
+rmiServer: RMIServer
+socketServer: SocketServer
+players: ArrayList<Player>
+socketplayer: InterfacePlayer
SocketPlayer
+*run(): void
RMIPlayer
+*run(): void
InterfacePlayer
Client
UserInterface
+startNewGame(arg1)
CLI
GUI
UserInterface
+controller: UserInterface
+*UserInterface(arg1)
+*UserInterface(arg1)
RMIClient
SocketClient
AbstractClient
+run()