	ServerModel		
	logger		gge
	server		erve
	questManager	StoryCardS	
	eventManager	StoryCardS	
	currentState	StoryCardS	
	adventureDeckDiscard	CardCollection <adventurec boo</adventurec 	
	inNextQ currentViewer	boo	in
	currentPlayer		in
	currentStage		in
	currentSponsor		in
	endTurnCounter		in
	gameWon	boo	olea
f o	AllyInPlaySirGalahad	boo	olea
	AllyInPlaySirLancelot	boo	oleai
f o	AllyInPlayKingArthur	boo	leai
	AllyInPlaySirTristan	boo	
	AllyInPlayKingPellinore	boo	
	AllyInPlaySirGawain	boo	
	AllyInPlaySirPercival	boo	
	AllyInPlayQueenGuinevere	boo	
	AllyInPlayQueenIseuIt AllyInPlayMerIin	boo	
	AllyInPlayMerlin numStages	boo	olea in
	numstages ServerModel(Server)		- 10
	instantiatePlayers(int)		voi
_	instantiateStages()		voi
	initialShuffle()		voi
	deal()		voi
	CardsTest()		voi
	resetCurrentStage()		voi
m 🚡	draw(String, int)		voi
m 🚡	removeFromParty(String, int)		voi
	removeShields(int, int)		voi
	party(String, int)		voi
	stage(String, int, int)	boo	
	unstage(String, int, int)		voi
	discard(String, int)		voi
	assassinate(String, int)		voi
	queue(String, int)		voi
	dequeue(String, int)	ollection <adventurecard>, String) boo</adventurecard>	
	containsSameweapon(CardC endTurn()		voi
	stageOver()		voi
	allysinPlay()		voi
	containsFoe(CardCollection<		
	containsAmour(CardCollectio		
	containsWeapon(CardCollect		
	getSubType(String, int)		tring
	playQuest()		voi
m ?	playEvent()		voi
	playGame()		voi
	nextPlayer()		voi
	nextStory()		voi
	setScenario1()		voi
	setScenario2()		voi
	setScenarioTest()		voi
	eventTesting()		
	ventureDeckDiscard pryDeckDiscard	CardCollection <adventurec CardCollection<storyc< td=""><td></td></storyc<></adventurec 	
	tivePlayer		ara: lave
p sta			Stat
P sta		ArrayList <cardcollection<adventurecar< td=""><td></td></cardcollection<adventurecar<>	
	rrentStage		in
	ventureDeck	Adventure	
	mPlayers		in
	rrentState	StoryCardS	Stat
		Pla	
p pla	iyer o	1 10	yer

	EventManager	
411	logger	Logger
411	players	Player[]
411	numPlayers	int
411	adventureDeck	AdventureDeck
41	adventureDeckDiscard	CardCollection <adventurecard></adventurecard>
411	nextQ	boolean
<u> </u>	currentPlayer	int
41	serverModel	ServerMode
411	EventManager(Abstract)	/lodel)
41	EventManager(ServerMo	del)
411	setPlayers()	void
411	handleEvent(String)	void
411	CourtCalled()	void
411	KingsRecognition()	void
411	QueenFavor()	voic
411	Pox()	voic
411	KingCallToArms()	voic
411	ProsperityThroughoutTh	eRealm() voic
41	ChilvarousDeed()	voic
411	Plague()	voic
41	handle()	voic
411	nextPlayer1()	Player
411	nextPlayer()	voic
411	setPlayer()	voic
411	canEndTurn()	boolean
41	resolveStage()	voic
41	increaseResponse()	voic
.P h	asSponsor	boolean

C B	QuestManager	
n a	logger	Logge
f a	ENDTURN	String
f o	serverModel	ServerMode
f o	players	Player[
f) a	questersReady	boolear
f) a	numOfansewers	in
f) ä	numberOfCardsToReturn	in
f) a	numberOfrequests	in
f) ä	questers	QuesterQueque
f) ä	numOfQuesterPotential	in
f) a	numOfQuester	in
f) ä	numOfRepsonders	in
f) a	nextPersonToDraw	in
m %	QuestManager(ServerModel)	
411	handle()	voic
41	nextPlayer()	void
411	stageHarder()	boolear
<u>411</u>	totalNumOfBP(CardCollection <advent< td=""><td>ureCard>) in</td></advent<>	ureCard>) in
411	isfoeEachStage()	boolear
411	numberOfCardsForSponsor()	in
<u> </u>	checkHandSize()	boolear
411	canEndTurn()	boolear
411	reset()	voi
411	setPlayer()	void
411	resolveQuest()	in
411	resolveStage()	void
411	increaseResponse()	void
p h	asSponsor	boolear

C	Server	
jn a	logger	Logg
f a	DEFAULT_PORT	i
n a	DEFAULT_PLAYERS_PER_TABLE	i
(n) a	maxPlayers	i
f) a	serverPort	i
f) a	playersPerGame	i
f o	clientThreads ArrayList <server< td=""><td>Thread</td></server<>	Thread
f) 'b	serverModel Serv	erMod
m %	Server(int, int)	
411	start()	VO
411	sendServerMessage(String)	VO
411	sendServerMessageToOne(String, int)	VO
411	sendServerMessageToAllButOne(String, int)	VO
411	getSponsorDecision()	VO
411	getQuesterDecison()	VO
411	main(String[])	VO
411	alert(String)	vo
411	updateViewState()	VO
411	resolveQuest()	vo
411	resolveStage()	VO
411	update()	VO
411	getTournamentDecision()	VO
₽ m	odel Serv	erMod

logger	Logger
ENDTURN	String
serverModel	ServerModel
players	Player[]
questersReady	boolean
numOfansewers	int
numberOfCardsToRetu	rn int
numberOfrequests	int
questers G	QuesterQueque
numOfQuesterPotentia	l int
numOfQuester	int
numOfRepsonders	int
nextPersonToDraw	int
TournamentManger(Ser	rverModel)
handle()	void
nextPlayer()	void
setPlayer()	void
canEndTurn()	boolean
resolveStage()	void
increaseResponse()	void
sSponsor	boolean
	ENDTURN serverModel players questersReady numofansewers numberOfCardsToRetu numberOfCardsToRetu numberOfGrequests questers Questers ContinumofQuesterPotentia numOfQuesterPotentia numOfQuester numOfRepsonders nextPersonToDraw TournamentManger(Se handle() nextPlayer() setPlayer() canEndTurn() resolveStage() increaseResponse()

C 1	ServerThread	
41	logger	Logger
411	socket	Socket
<u> </u>	server	Server
411	out	PrintWriter
411	in B	ufferedReader
411	ServerThread(Socket, Server, int)	
411	run()	void
411	getClientMessage(Strin	g) void
playerNumber int int		

C B	MainMenu	
411	logger	Logger
411	IMG_DIR	String
3	GIF	String
411	SQUIRE	String
411	KNIGHT	String
3	CHAMPION_KNIGHT	String
41	numberPlayers	int
40	MainMenu(View, Pane)	
41	numberSelected()	int