**Documentation – Quest of the Round Table**

Team 35 – Luvai Hassanali 10092392

Wednesday Apri 11th

Table of contents:

1. Language and libraries used

2. Use cases

3. Client / server diagrams

4. Architecture Notes

Languages used:

Java on server side and Javascript on client side.

Libraries used:

Client:

- jQuery (used for image manipulation)

Server:

- Springboot Framework

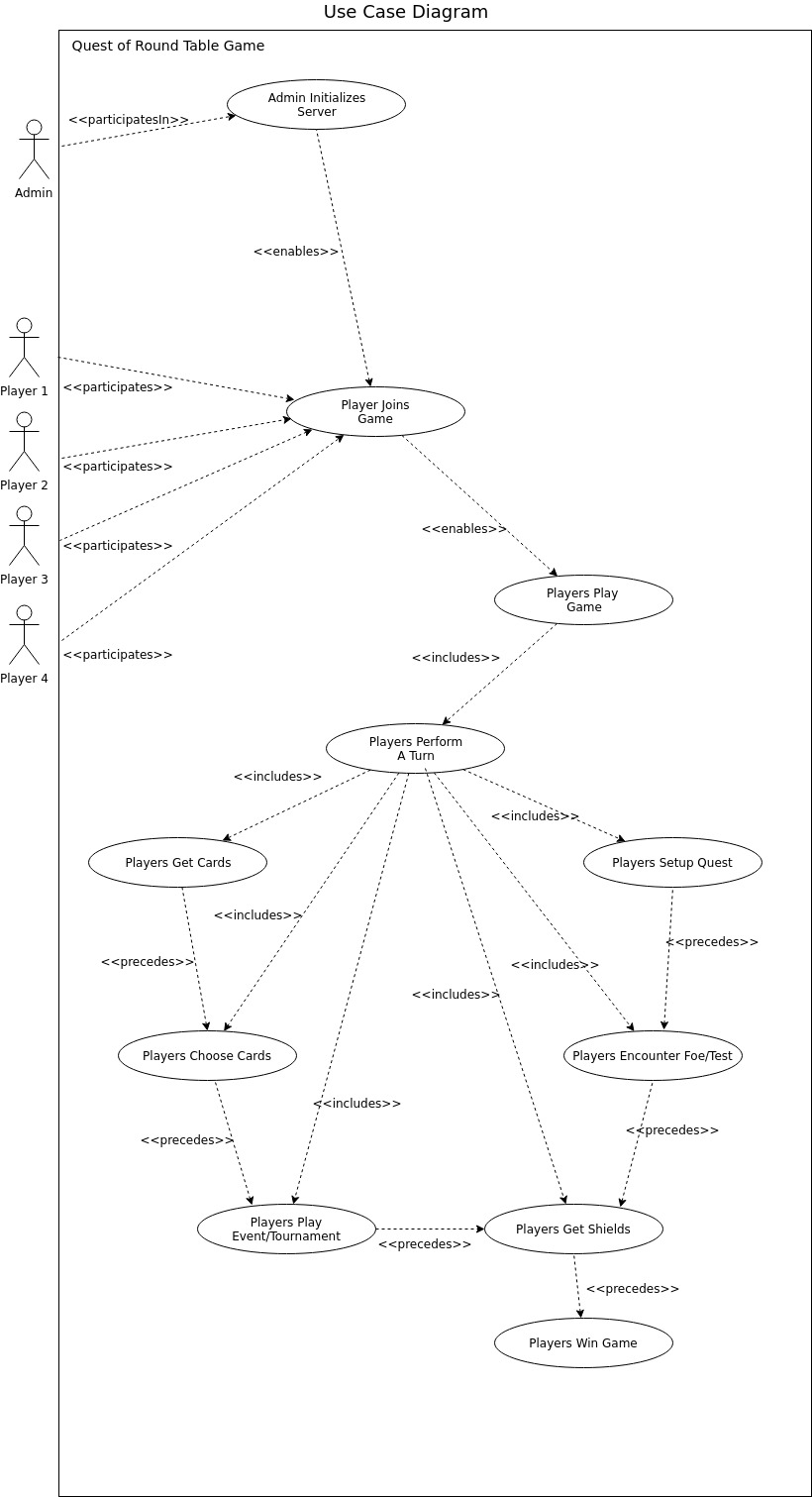
- Log4j2 logging

- JSTL (used to render view)

- Apache Tomcat

- Spring Websockets

- Google json (JSON functionality)



Use Cases (rules with reference tags below)

|  |  |
| --- | --- |
| **UC-01** | **Admin Initializes Server** |
| Description | Player (as Admin) starts server application so that other players can connect to game. |
| Actors | Player |
| Triggering Event | Admin decides to start game |
| Main Sequence | 1. Player runs the server.  2. Server ready to accept player connections.  3. Server shuffles story deck & adventure deck for game start.  4. Player enrolls into game as Player 1. |
| Resulting Event | Server and game decks are now setup and waits for connections from remaining players. |
| Traceability | Game components - A01 |

|  |  |
| --- | --- |
| **UC-02** | **Players Joins Game** |
| Description | Player connects to server and enrolls into game. |
| Actors | Player |
| Triggering Event | Player opens webpage to game/connects to server. |
| Main Sequence | 1. Player session connects to server.  2. Player enters nickname for game.  3. Player is added to game participant list. |
| Resulting Event | If 4 players connected, game starts. Else wait for more players to connect. |
| Traceability | Logfile: scenario1.txt(46-60) |

|  |  |
| --- | --- |
| **UC-03** | **Players Play Game** |
| Description | All players have connected, game view is rendered, hands are dealt, and first player flips story deck. |
| Actors | 4 players |
| Triggering Event | Final player joins game. |
| Main Sequence | 1. Fourth player enrolls into game.  2. Each player is dealt a hand.  3. First player flips story deck card.  4. Game view is rendered (cards in hand, story deck face up card, & rank card face up for each player) |
| Resulting Event | Face up card is triggered whether it be event, quest, or tournament. |
| Traceability | Game setup - A03 |

|  |  |
| --- | --- |
| **UC-04** | **Players Perform A Turn** |
| Description | After story card is flipped, player must make a decision to either: setup quest, participate in quest, or participate in tournament. |
| Actors | Player |
| Triggering Event | Story card being flipped. |
| Main Sequence | 1. Story card flipped.  2. Player performs a turn.  3. Next player in turn performs a turn till all player decisions collected. If quest → player setup quest |
| Resulting Event | If quest: A setup quest based off flipped story deck card → all remaining players perform a turn to accept/decline  Else: Next player in turn performs a turn (accept/decline tounrnament OR flip story deck) |
| Traceability | Game sequence - A04 |

|  |  |
| --- | --- |
| **UC-05** | **Players Get Cards** |
| Description | Player picks up card(s) into hand from adventure deck. |
| Actors | 1-4 Players |
| Triggering Event | If player is enrolled in quest then at each stage, before tournaments, or else by certain events. |
| Main Sequence | 1. Player is dealt X number of cards.  2. If player is holding more than 12 cards they must discard the appropraite number of cards.  3. Player discards to complete event or sends test bids or battle equipment for quest. |
| Resulting Event | Event is completed and next player in turn flips story deck or information for quest is sent to server. |
| Traceability | Playing a quest, Game sequence - A05 |

|  |  |
| --- | --- |
| **UC-06** | **Players Choose Cards** |
| Description | Player chooses cards for use in game or to discard because hand contains over 12 cards. |
| Actors | Player |
| Triggering Event | Quest setup, event, and quest/tournament participation. |
| Main Sequence | 1. Player chooses card based off current situation.  2. Card is removed from player hand and added into gameplay or discard pile. |
| Resulting Event | Player loses a card |
| Traceability | Setup Quest, Play Quest, Tournaments - A06 |

|  |  |
| --- | --- |
| **UC-07** | **Players Setup Quest** |
| Description | Player chooses cards for number of stages on face up quest card. |
| Actors | Player |
| Triggering Event | Player chooses to accept sponsoring quest. |
| Main Sequence | 1. For number of stages listed on current quest card Player chooses either Test Card or Foe along with weapons. |
| Resulting Event | Quest is setup and remaining players are asked to participate. |
| Traceability | Setup Quest - A07 |
| **UC-08** | **Players Encounter Foe** |
| Description | Player is participating in current quest and sponsor has placed a foe for current stage. |
| Actors | 1-3 Players |
| Triggering Event | Current stage for quest contains foe. |
| Main Sequence | 1. Player chooses cards to use as equipment (weapon, amour, ally) for foe OR does not choose anything.  2. Player chooses cards to increase battle foints for upcoming battle.  3. Goes into battle concurrently after all other players have chosen equipment. |
| Resulting Event | If all quest participants have chosen equipment, battle screen renders. Else, player waits till all other participants have chosen equipment. If player battle points are equal to or higher than foe, player advances to next stage. Else player is ejected from quest. If last stage then player wins quest and receives shields. |
| Traceability | A foe is encountered - A08 |

|  |  |
| --- | --- |
| **UC-9** | **Player Encounter Test** |
| Description | Player is participating in current quest and the current stage in play has a test card. |
| Actors | Player |
| Triggering Event | Current stage for quest contains test. |
| Main Sequence | 1. Player can choose to drop out of test or bid according to current minimum bid.  2. Player chooses cards to bid.  3. Waits for other participants to finish bidding. |
| Resulting Event | Player sends bid to server and next participant in turn is asked to bid. If bid is higher that previous current minimum bid, new minimum bid is set. If all other participants have dropped out, test is won and player advances to next stage. If last stage then player has won quest and recieves shield. |
| Traceability | A test is encountered - A09 |

|  |  |
| --- | --- |
| **UC-10** | **Players Play Event** |
| Description | An event card was flipped which contains rules that includes current player. (number of shields, rank, etc.) |
| Actors | 1-4 Players |
| Triggering Event | Player fits decription stated by event card and so must play event. |
| Main Sequence | 1. Event card is flipped up from story deck.  2. Player must follow rules and do what event states. |
| Resulting Event | Is dictated by current event card. (Could recieve/lose shields) |
| Traceability | Game sequences - A10 |

|  |  |
| --- | --- |
| **UC-11** | **Players Play Tournament** |
| Description | All players who accept to participate choose equipment (weapons, amour, ally) to use in Tournament. Player with highest battle points wins. If tie-breaker, players involved discard weapons and re-equip and battle again. |
| Actors | 1-4 Players |
| Triggering Event | Tournament card is flipped up from story deck. |
| Main Sequence | 1. All players are asked to participate.  2. Participants choose equipment to increase battle points.  3. Participant with highest amount wins the tournament. |
| Resulting Event | Player who wins receives numbers of shields equal to participants plus any bonus shields listed on current Tournament Card. |
| Traceability | Tournaments - A11 |

|  |  |
| --- | --- |
| **UC-12** | **Players Get Shields** |
| Description | After completing event, tournament, or quest a player recieves X number of shields. |
| Actors | 1-3 Players |
| Triggering Event | Completed event, wining quest or tournament. |
| Main Sequence | 1. Player is given X amount of shields.  2. Player is then checked for rank advancement and approriate rank cards are switched and battle points increased if true. |
| Resulting Event | Players number of shields increases. If player shields is equal to greater than defined rank, player is advances in rank. If player advances to higher than 10 shields the game is over. |
| Traceability | Shields & Rank - A12 |

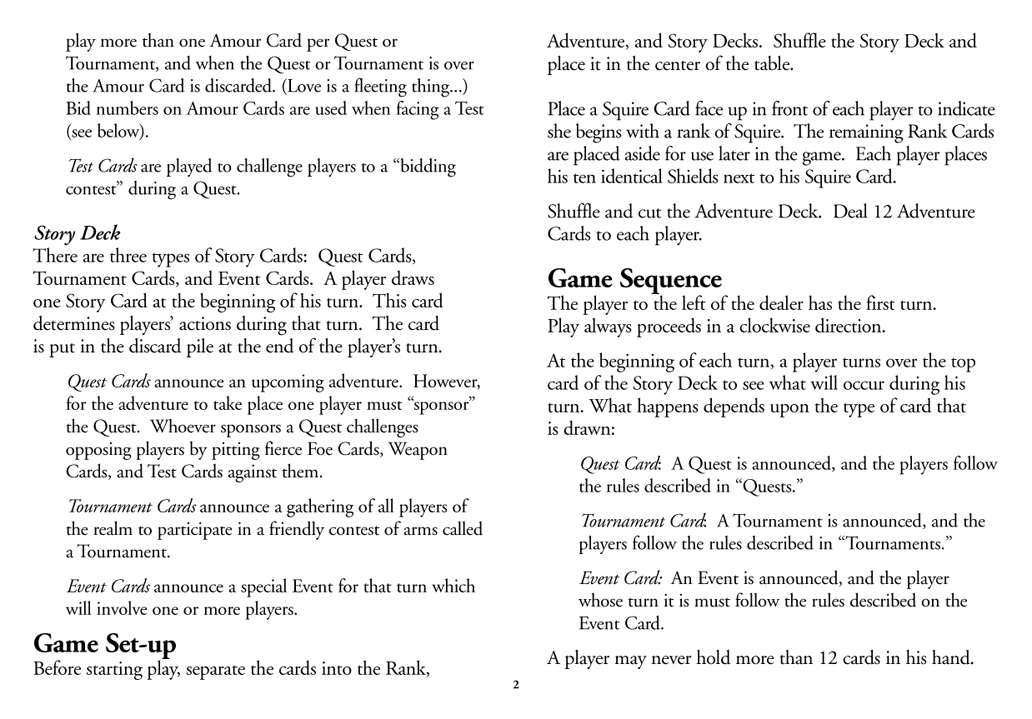
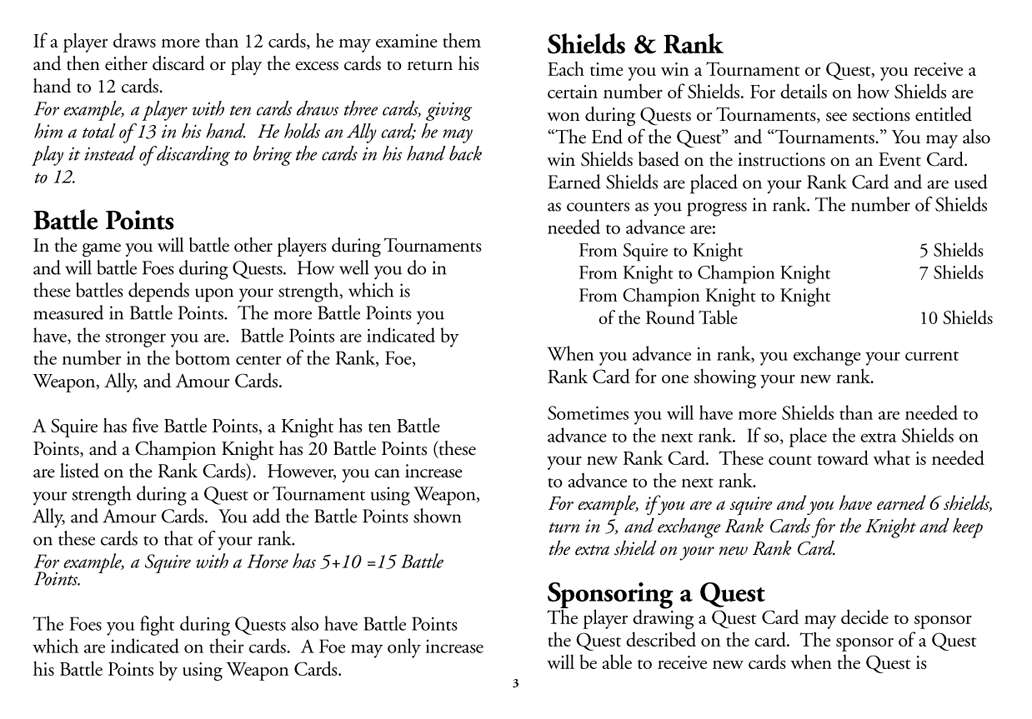
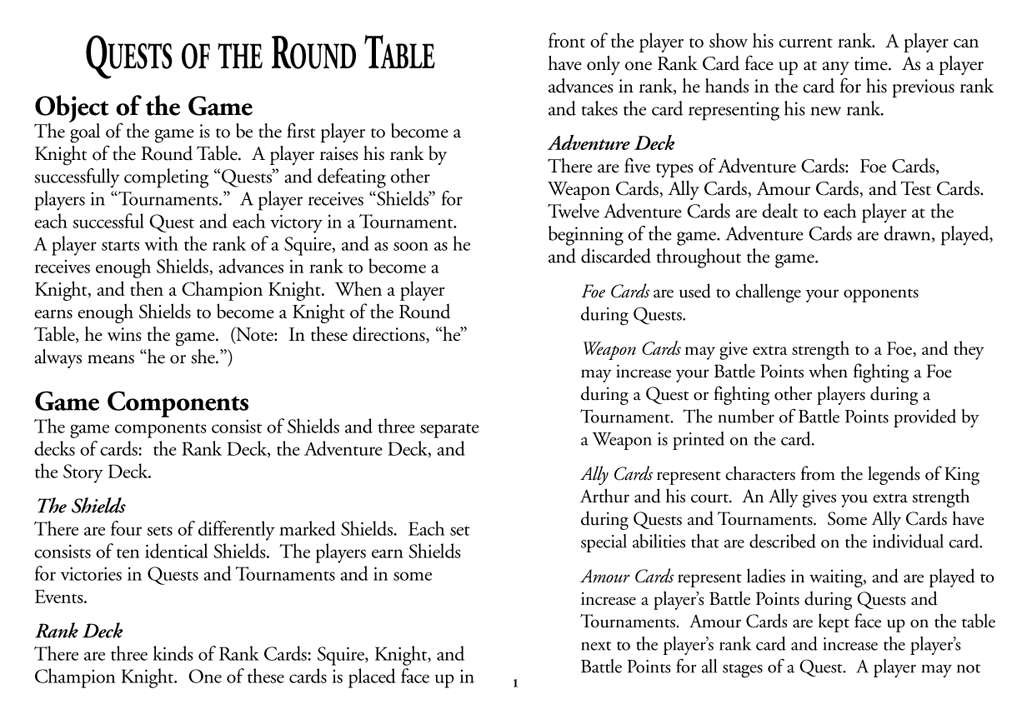
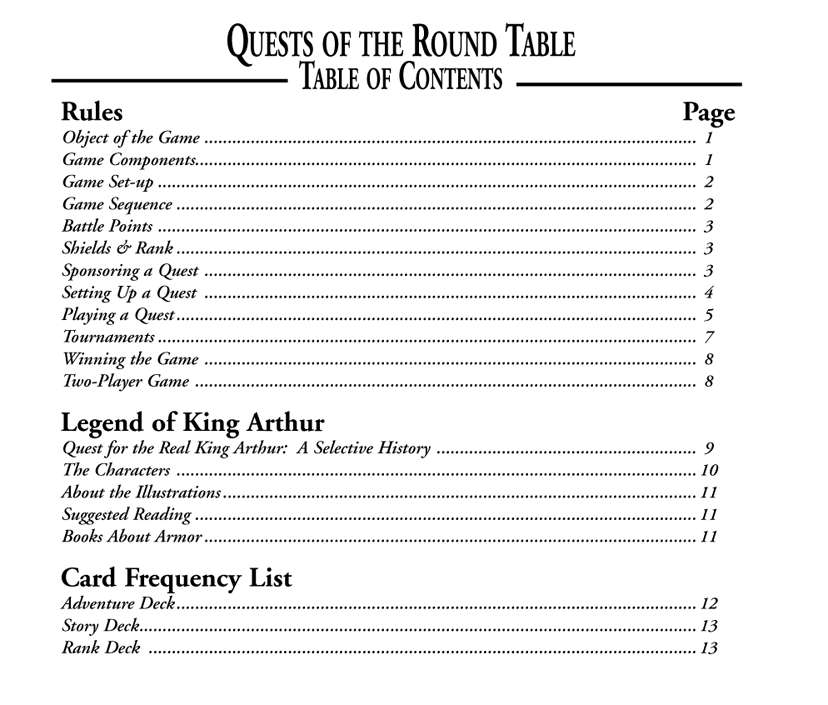
|  |  |
| --- | --- |
| **UC-13** | **Players Win Game** |
| Description | When players gain 10 shields they advance to rank of Knight of the Round Table and win the game. If there is a tie, those of the last rank must participate in a final tournament. |
| Actors | 1-3 Players |
| Triggering Event | Player gains 10 shields and advances to rank of Knight of the Round Table. |
| Main Sequence | 1. Player gains shields by means of event or winning quest or tournament.  2. Player is checked for rank advancement and has enough shields (10) to reach next rank which is Knight of the Round Table. |
| Resulting Event | Player is winner of the game. If more than one player advances rank to Knight of the Round table at the same time, those players participate in a final tournament. |
| Traceability | Winning game - A13 |

Test Cases

|  |  |
| --- | --- |
| **TC-01** | **Foes with Extra Battle Points** |
| Description | Some foe cards contain two battle points values. When a foe is chosen for a stage of a quest if the current Quest names the foe the second value of battle poitns is chosen. |
| Actors | 1 Player |
| Triggering Event | Player sponsoring quest chooses foe for stage where foe is equal to the one listed on quest card. |
| Main Sequence | 1. Player chooses cards for quest setup.  2. Player chooses foe which equals to that named on quest card.  3. Foes battle points equal to the second higher value listed on the card. |
| Resulting Event | Foe battle points higher than regular during battle phases. |
| Traceability | Logfiles: scenario1.txt(71-77, 86, 140) [Boar with sword and dagger = 30 because of extra battle points] |

|  |  |
| --- | --- |
| **TC-02** | **Ally/Amour extra bids** |
| Description | When bidding for a test, an ally card or amour will list a bonus bid. If the test is won the number of cards needed to discard is reduced by combined number of bonus bids. |
| Actors | Player |
| Triggering Event | Player wins test. |
| Main Sequence | 1. Player wins test.  2. Player discards needed for test is now minus bonus bids |
| Resulting Event | Less cards lost during bidding of tests |
| Traceability | Logfiles: scenario3.txt(48, 54-56) [player only discards 3 due to bonus bids of King Arthur] |

|  |  |
| --- | --- |
| **TC-03** | **Merlin** |
| Description | Before answering to participate in quest, if player has Merlin in hand then they are given option to preview one stage of setup quest. |
| Actors | Player |
| Triggering Event | Player is asked to participate in quest while Merlin is in hand. |
| Main Sequence | 1. Player is prompted if they would like to preview stage.  2. Player clicks corresponding button to the stage they want to preview.  3. Preview is displayed on screen for 10 seconds. |
| Resulting Event | Player gains knowledge of one stage of setup quest then can decide whether to participate. |
| Traceability | Logfiles: scenario1.txt(83-86), merlin\_log.txt(81-84), merlin\_log(88-91) |



A-01

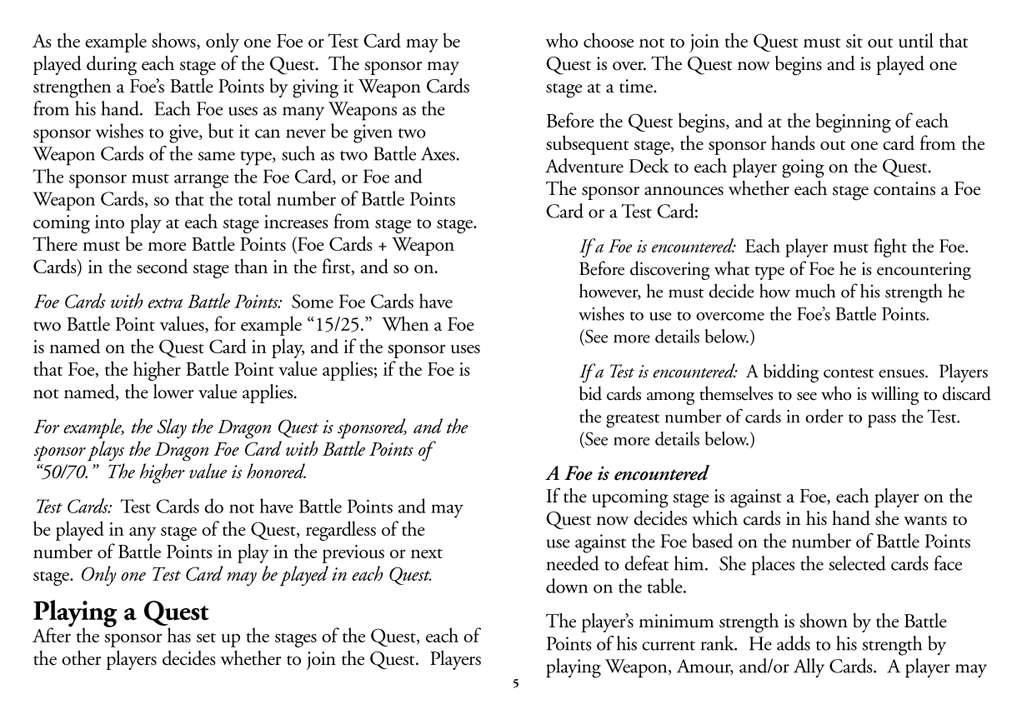
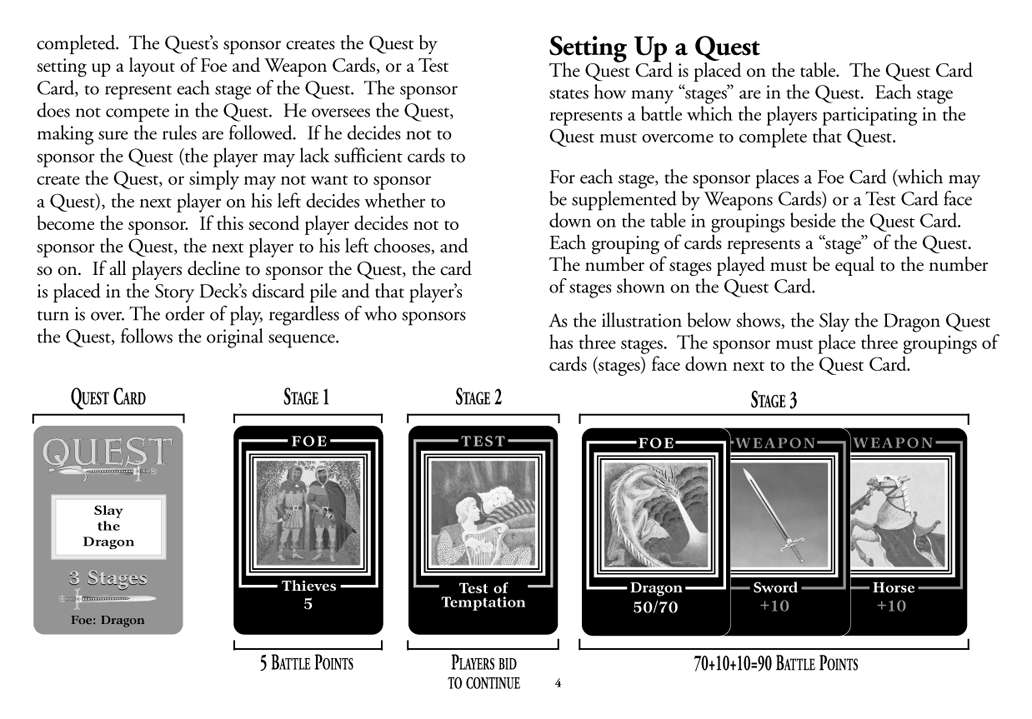
A-03

A-04

A-10

A-12

A-05

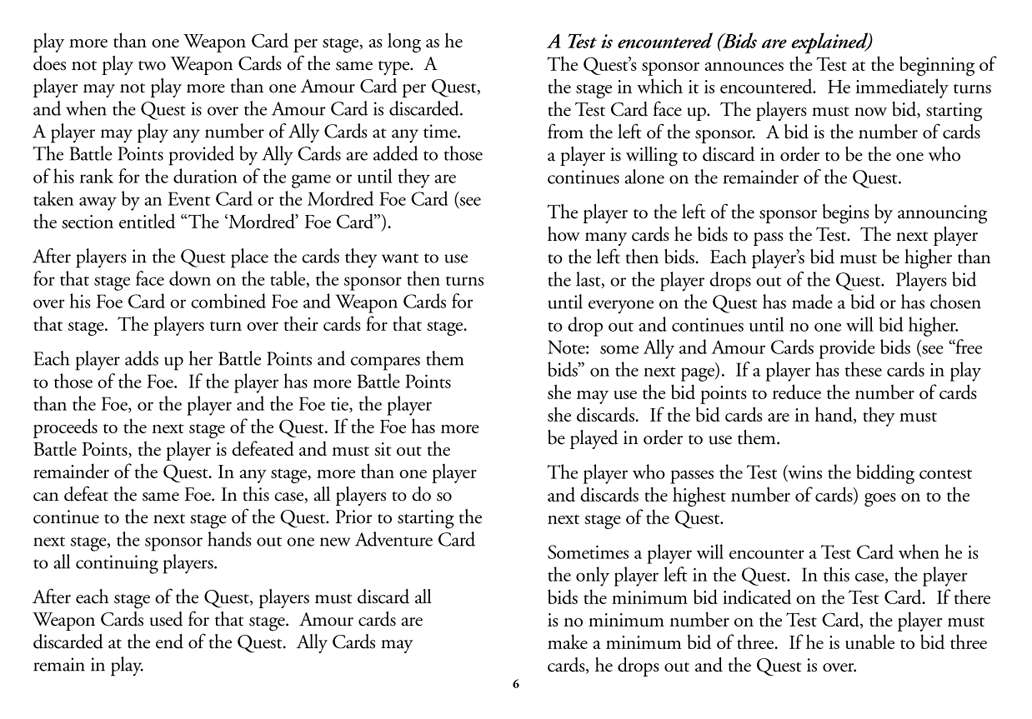
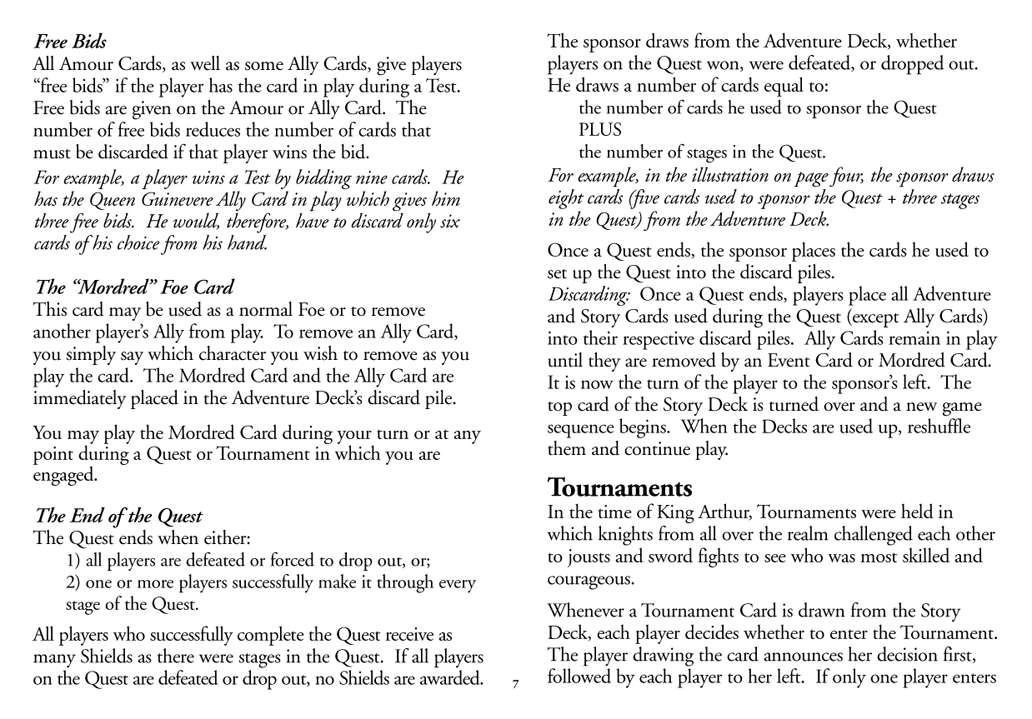


A-07

A-06

A-08

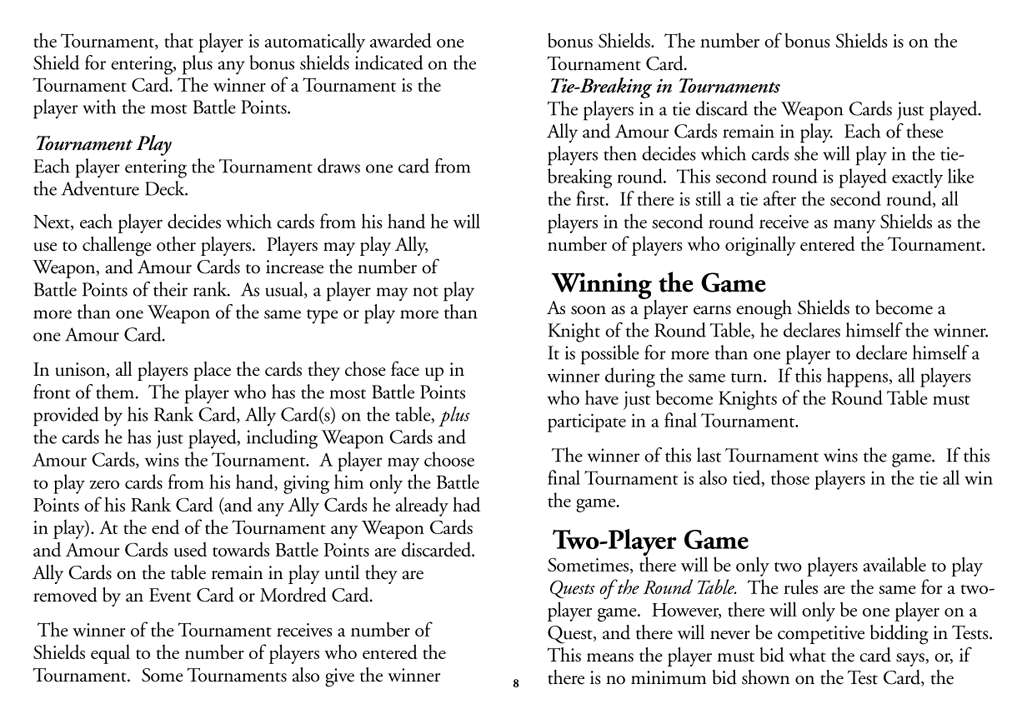
A-06



A-09

A-06

A-05

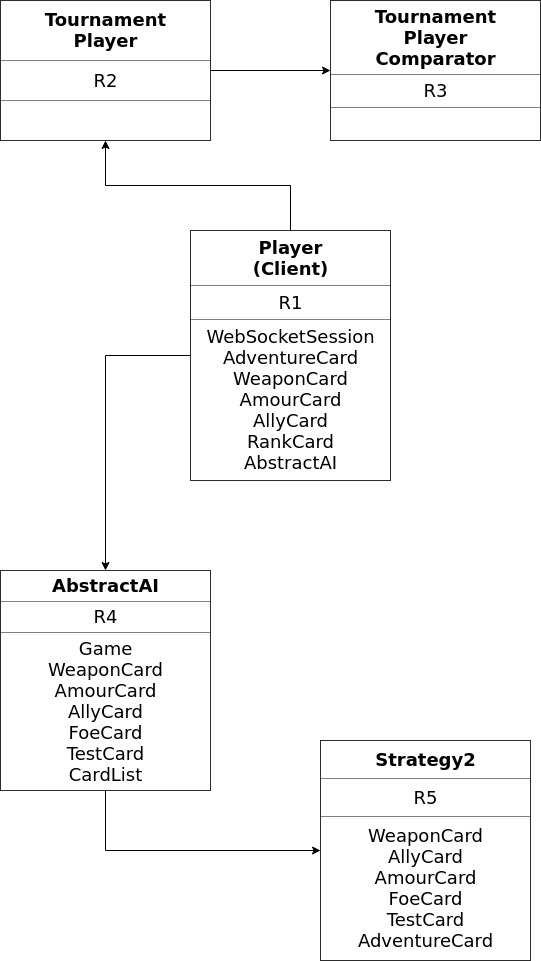


A-11

A-06

A-13

Client Diagram



Client responsibilities

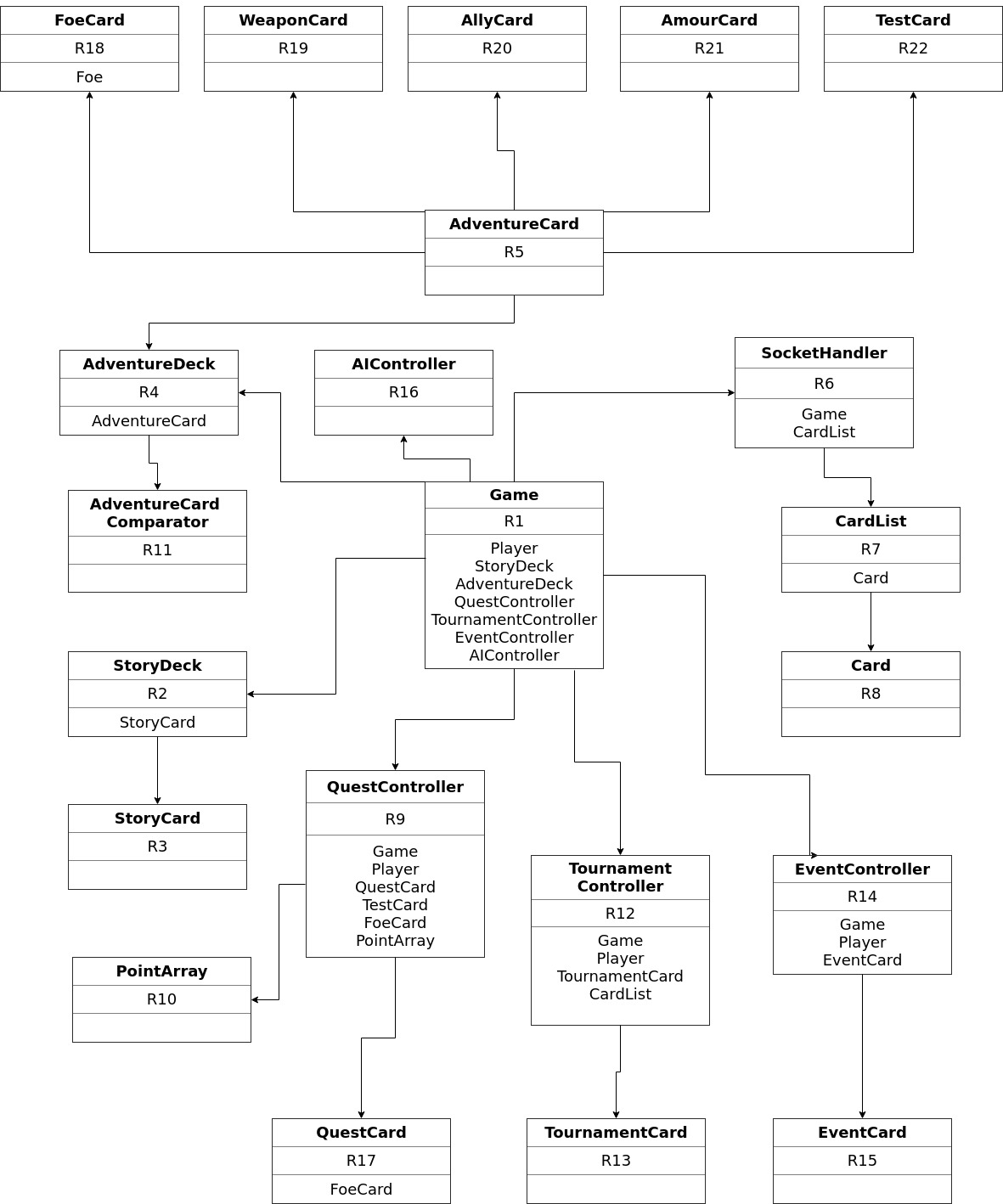
R1 – Represents the client in game. Holds all relelvant player information and statistics needed for gameplay. As the game is played the class updates information to keep track of progress.

R2 – For tournaments to store contestant information. Used alongside Tournament Player Comparator to determine winner.

R3 – Organizes tournament contestants by increasing battle points to determine winner.

R4 – Provides an outline for various methods to which AI Player strategies are implemented.

R5 – Follows AbstractAI class and implements a specified stratergy pattern for non-human players of game.

Server diagram

Server responsibilities

R1 – Game class holds game utilities such as deck, player objects, controllers for quest, tournament, and events. Controls turns and flow of the game. The game class is also responsible for initiating the game, dealing players hands, shuffling decks, etc.

R2 – Story deck contains story cards which are flipped up in game. This class also controls deck when rigged game is initiated.

R3 – Story cards can be quest, tournament, or event. Main component of game, card is flipped from story deck at commencement of player’s turn.

R4 – Adventure deck class holds the adventure cards which make up the deck for the game. Shuffled to ensure randomness.

R5 – Adventure cards make up the hand of player. Can be of different types each performing a different function.

R6 – Class used to communicate with clients using web sockets.

R7 – CardList class contains all card information. Lists all cards used in gameplay and also implements a finder function to return card object by string name.

R8 – Card is the base object used by all specific card types. Contains a name and image string.

R9 – Controls current quest. Stores all quest information created by sponsor and participants and performs approriate calculations to find winner and then send results to client side.

R10 – Used to store information of foes and corresponding points when setting up quest.

R11 – Compares adventure cards based on battle points. Sorts in increasing order.

R12 - Controls current tournament. Stores all challengers information then performs calculations to determine winner, deal with tie breaker then send resulting information to be displayed for clients.

R13 – Tournament card placed in story deck to intitiate a tournament. Has name, picture, and sometimes bonus shields to be awarded to winner.

R14 – Event controller regulates the flipped event card. Ensures all players perform necessary action and then executes the results dictated by the card.

R15 – Event card from story deck used to initiate tournament. Holds all relevenat information needed for players to perform event.

R16 – AIController used to parse information from AI player clients then pass that information to the corresponsing strategy class. The results are then parsed by the controller to make the AI player perform the dictated outcome.

R17 – Quest card contains quest information along with number of stages or any special foes. Flipped up from story card and can be setup by current player.

R18 – FoeCard dictates foes in the game. Contains name, image, battle points, bonus points (if applicable) and any specials the card may have.

R19 – WeaponCard dicates weapons in the game to be used by players or foes. Contains battle points, name, and image.

R20 – AllyCard dictates allies for the game. Can be equipped for length of game to strengthen battle points. Some allies contains bids used in tests and other various specials.

R21 – AmourCard used as defense within game. Can be used during any battle to increase battlepoints by ten. Is discarded after quest stage/tournament completion.

R22 – TestCard used in tests to create a challenge where players bid cards against one another. Used in quest setup.

Architecture Notes

What client does:

- client receives messages from server then parses that information to get player choices and/or render displays

- maintains connection with server socket in order to be able to receive and send messages back and forth throughout the game

What server does:

- receives messages from clients then parses that information to store information about player choices then use than information to execute game logic then send results back to clients

- server maintains a one socket connection with players and stores client session information to easily send messages

How they interact:

- client and server both send information in order to advance game

all client messages are routed through server, server controls relay messages back to clients

- client sends player choice information while server parses then sends results in order to display results in all client screens