**Updated information for RPG Game:**

**Summary of the application that your program will implement:**

The program will be a text-based adventure game.

First the user is asked to enter their username – The program ensures that this username was not used yet

Then they will receive a menu of starting the game, viewing the scoreboard and exiting the program. Assuming the user chooses the game option, they will be prompted with different scenarios and will be given the opportunity to choose between 2 choices. In some cases they will also be able to answer with words (riddles). From there the player will keep playing till their demise or the end of the game. Within each scenario there will be the opportunity for the player to gain points or loose points - they are not told whether or not they will gain or loose points from a decision until after the decision is made, sometimes they dont loose anything at all. But when they do win or loose points for a decision, a 30 sided dice will be rolled to decide upon the amount. Upon the completion of the game the user’s username, score, and timestamp will be written to a file, which then can be read out (by username or by scores) in the menu of the game through the scoreboard option.

**Description of input and output of the program:**

Starts with a menu with 3 options: [1] Start game, [2] View scoreboard [0] Exit

The user's input will be error-trapped, meaning that they can only answer with 1, 2 or 0.

If the user chooses to start the game, they will be “transported” into an adventure scenario and given the option to make a decision, an example of this would be:

“You awaken and find yourself deep within an ancient, mysterious forest.The towering trees block the sun, and the dense foliage obscures any sense of direction. You must navigate through the forest, where every path seems to hold both promise and peril. The forest suddenly opens up with 2 paths... One path opens up into a serene clearing revealing glimpses of sunlight while the other path winds through ancient trees and dense undergrowth. Which will you choose? [1] The clear path [2] The forested path:”

From there the user would make a choice, and continue along their journey until the games completion. Some choices will allow the player to enter words, and in some cases they will gain and loose points.

**Description of what ADTs will be used:**

A Binary Search Tree structure for this kind of branching storyline. Each node in the tree represents a story event, and child nodes represent choices that lead to different outcomes. This would not only enable me to explore a new ADT but it is the most fitting for such a game.

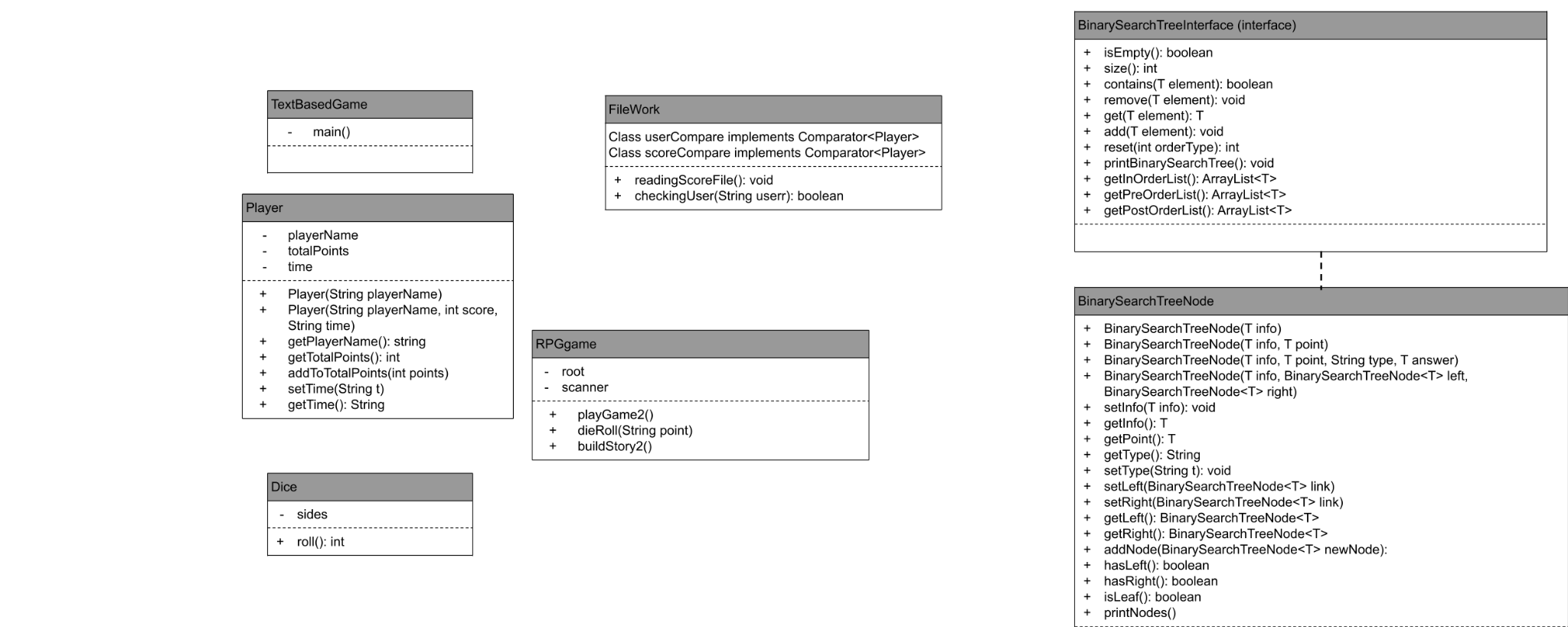
A dice class would also be used for the rolling of the 30 sided dice

**Description of programming tools:**

Programming language: Java

Software development environment: Eclipse

**Classes and their info:**

* TextBasedGame:
  + Contains the main method to start the game.
  + Contains the menu.
  + Manages the game's overall flow and control.
* RPGgame:
  + Sets up each node. - With scenarios, answer type (only for riddle questions), answers (again only for riddle or word answer questions/scenarios), and whether the choice is a win or lose points choice
* BinarySearchTreeNode
  + Used to setup each scenario and build up the chain
* FileWork:
  + This class reads out the score file to print out the score board (option in main menu),
  + Used when checking if a user name already exists (used in beginning of program)
* Player:
  + Represents the player's profile and keeps track of users info, their running points, their username, final score and their time
* Dice:
  + “Rolls” a dice to generate a random num of point for the user
* BinarySearchTreeInterface (Interface):
  + Defines methods for representing nodes in the decision tree, including getting descriptions, choices, and points.
* Random (Java API)
  + Used in Dice class for random number generation
* Scanner (Java API)
  + Used for user input and interaction with the player
* LinkedList (Java API):
  + Used for implementing the decision tree structure, such as maintaining the choices as linked lists.