

Summary of the goal or purpose:

The goal of this assignment was to migrate my terminal based Yahtzee program into a fully functional Yahtzee program that can be played solely with a GUI. In addition, the goal was to meet the requirements of HW3 in terms of game logic, and to create a GUI that followed this game logic that is easy and intuitive for the player to use.

The purpose of this assignment was to grow familiar with Swing and learn more about the concepts of events and GUIs. In addition, the purpose was to learn about how to make a program that is user friendly to the non-programmer.

Overview of the general design for my program:

The general design of my program changed quite a bit from HW3. The flow of the program can take a few different routes depending on what the user decides to do, and how they decide to play. In general, the game starts off with a configurations frame that uses combo boxes so that the user can choose the number sides on a die, the number of dice in a hand, and the number of rolls per hand. When the user is ready to continue, they click on the play button. From here, the user's hand of dice is displayed via images to them. At this point the user has the option to click on the dice that they wish to keep and click the roll button, or they can click the View Scorecard button and it will show them their current scorecard. Once the user has reached the maximum number of rolls for their hand, a new frame is displayed. This new frame shows the user the result of their last roll, along with all of their scoring options. It is up to the user to decide which line they want to score on. The rolling and scoring continue until there are no more lines to score on. From here, the game over frame is displayed to the user along with their final scorecard for the game. If the user wants to play again, then all they have to do is click on the play again button and the process repeats until the user no longer wants to play.

Major design and/or programming issues:

This programming assignment was by far the most time consuming. My major design issue was that I did not design and plan enough before starting to program. Initially, I had drawn out what each of my frames would look like once the GUI had been implemented, but that was the extent of my preparation. I decided to learn swing as I progressed through the assignment rather than study up on it before programming. This decision probably costed me 10 hours of time because after 10 hours of trying to program a GUI, I had nothing to show for. So, I decided to actually go through the Horstmann text and understand how Swing works before my second attempt.

The main programming issue that I ran into was trying to use the images of the player's hand as JToggleButton. I eventually was able to do this, but only a single die could be selected at a time which is not ideal if the user wants to save all of the dice in their hand. So, after some thought I

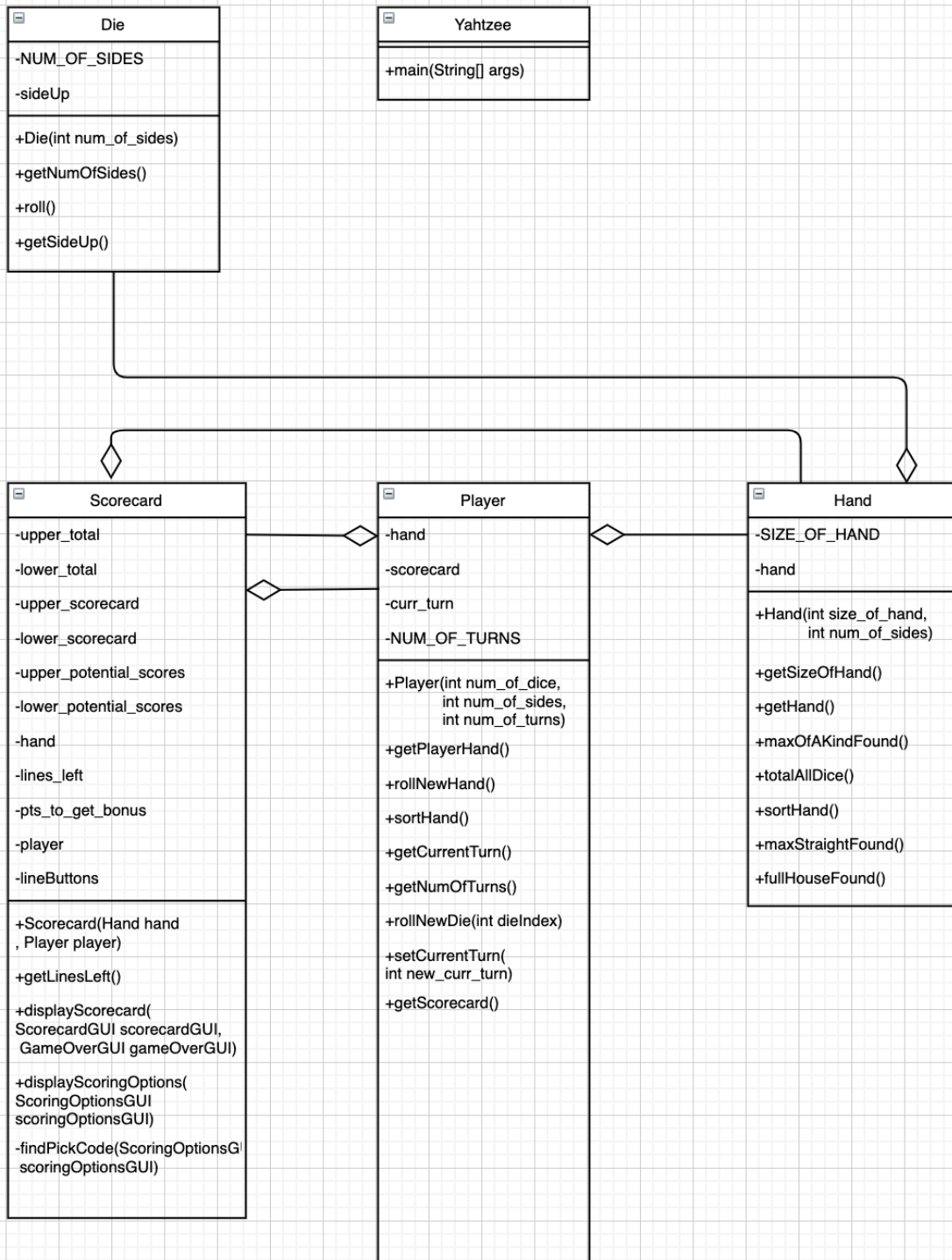
decided to settle for JToggleButtonButtons that are placed below each die, and the user is told to click on the buttons that correspond to the dice that they want to keep.

In retrospect:

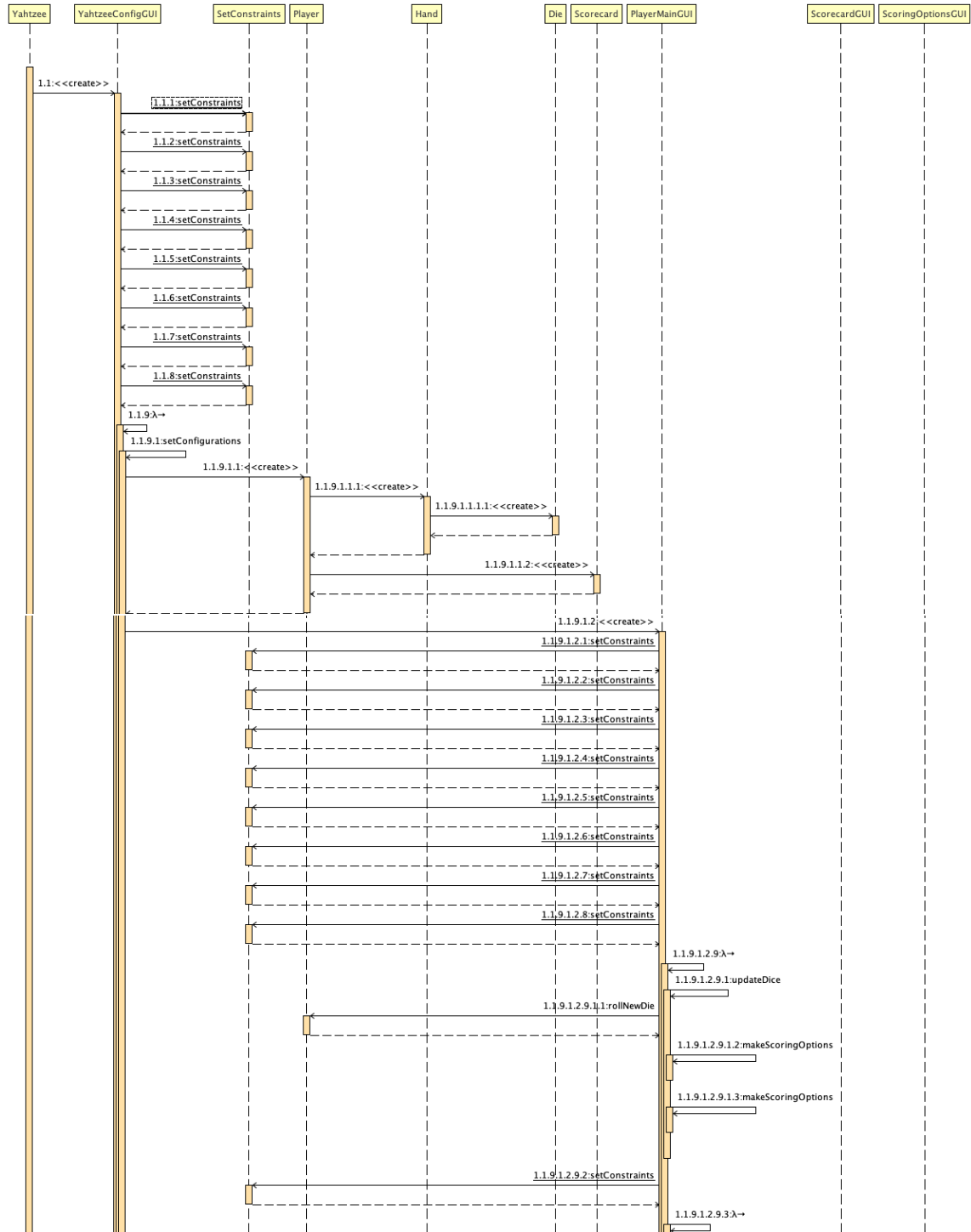
I would learn more Swing before jumping into the programming. Also, if I were given more time, I would like to learn more about other layouts, since for this assignment I mainly just used GridBagLayouts. GridBagLayouts were easy to learn, but as I progressed through the assignment, placing each component of the JFrame exactly where I wanted them to be became very tedious and time consuming.

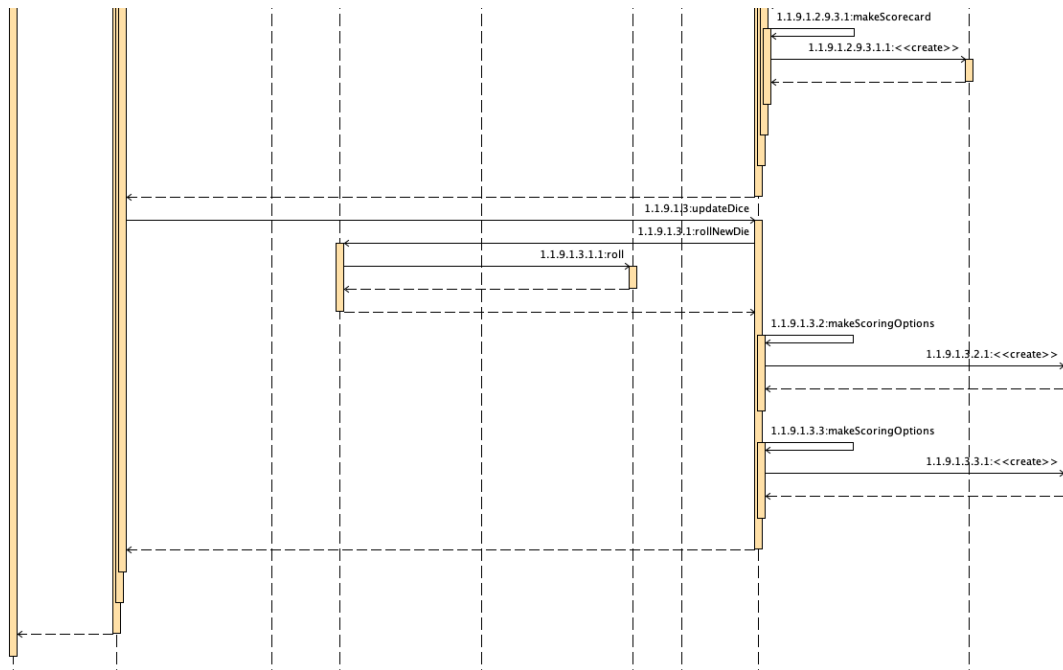
UML DIAGRAM ON NEXT PAGE (Doesn't include GUI components per Homework specifications)

Note: Yahtzee creates a YahtzeeConfigGUI object that starts the process of the game of Yahtzee.



UML SEQUENCE DIAGRAM NEXT PAGE





Note: You'll notice that `setConstraints` is called quite a bit. `setConstraints` is simply a helper function for setting the constraints of a `GridBagLayout` component. So essentially, all it is doing is positioning GUI components at a particular location on the `Frame`.