

Integrantes:

Manuel Arango Gomez

Andrés Camilo Álvarez Vásquez

Jhon Fredy Giraldo Osorio

ESTRATEGIAS Primera parte

A la hora de reunirnos para charlar y debatir sobre el proyecto del Scrabble, lo primero que realizamos fue leer detenidamente cada parte del documento en el cual se hablaba del nuevo proyecto a entregar para este semestre, luego de que cada integrante leyera el documento nos empezamos a preguntar varias cosas como: ¿Cómo se juega el scrabble? ahí fue cuando llego uno de los compañeros y se acordó que tenía el juego de mesa Scrabble y nos empezó a leer las normas del juego y ahí fue cuando empezamos a tomar varias de estas ideas como los valores que tendrá cada letra, formas de generar palabras, cuantas letras tendrá cada jugador, pero cuando leímos que habían letras como: "RR", "Ñ", "CH", "LL" y nos dimos cuenta que estas letras no las podríamos utilizar en nuestro código, así que empezamos a crear nuestra propia versión de Scrabble y ahí fue cuando nació "Scrabble Proyect", empezamos con el diccionario, nos preguntamos ¿Cómo leemos un documento en java? luego de investigar y navegar un buen rato nos dimos cuenta que las soluciones que nos proponían no nos funcionaba ya que cuando lo implementábamos el código no leía la totalidad de las palabras, lo que nos llevó a que un de los integrantes del equipo nos mostrara un código el cual le había mostrado y explicado una profesora de desarrollo de software, así fue como encontramos la solución para leer estas palabras, luego empezamos a pensar el cómo saber si las letras que tiene la persona en mano puede formar letras disponibles en el diccionario, dado esto creamos un código el cual es recursivo que recorre el diccionario y luego valida si la palabra puede ser formada con las letras del usuario, con esto en mente debíamos buscar la manera de presentarle a usuario las 10 mejores posibilidades lo que nos llevo a asignarle un valor a cada letra para luego formar un arreglo en el cual se guardaban las 10 mejores palabras para así poder imprimirlas con el puntaje que tiene cada una y que el jugador escoja una de las 10 mejores palabras que podría escoger, luego de que el jugador escoja esta palabra, se retiraran las letras que contiene la palabra de las letras que tiene el jugador en mano, el programa guarda la palabra que se escoge y también guarda el puntaje acumulado de cada jugador, para así con esto el jugador poder escoger siempre la mejor opción en cada round luego de llevar todo esto a código nos encontramos con algunos problemas a los cuales les dimos solución rápidamente, uno de los primero problemas que tuvimos fue a la hora de hacer la permutación de las letras ya que lo hacíamos con ciclos, pero nos dimos cuenta que a la hora de ejecutar el programa este hacia muchas iteraciones y se demoraba mucho en presentar a la persona las mejores 10 posibilidades, a lo cual recurrimos como dijimos anteriormente a la

recursividad, fue difícil encontrar la forma correcta de hacer la permutación ya que a la hora de que el sistema realizara la permutación lo hacia sin repetición, lo cual nos llevo a pensar como hacer que la recursividad que estábamos implementábamos hiciera todas las repeticiones que debía hacer.

Luego de mucho trabajo y tiempo logramos encontrar solución a cada uno de los problemas que nos surgían y de esta manera pudimos formar nuestra propia forma de ver el juego y cada vez que hacíamos algo nos sentíamos orgullosos, pero siempre queríamos mejorar el código, pulirlo, poner mejores mensajes al usuario para que a la hora de hacer uso de nuestro código se pudiera entender todo perfectamente, además tratamos de comentar cada línea del código para así la persona que lo lea y lo trate de entender lo haga fácilmente.

ESTRATEGIAS Segunda parte

Luego de hacer la entrega de la primera parte nos reunimos para pensar como haríamos la segunda entrega en la cual debíamos de crear un tablero en el que iban a jugar y interactuar las personas, para esto decidimos crear una matriz la cual se deberá ir actualizando con la opción elegida por el usuario siempre guardando la jugada anterior hecha por el usuario y se crearon unas reglas que se debían seguir como por ejemplo al inicio del juego el sistema pedirá que la persona ubique la palabra pero si este la ubica en otro lado que no sea el centro el sistema estará encargado de poner la palabra en todo el centro, cuando codificamos esto seguimos algunas reglas del juego original ya que este siempre debía iniciar en el centro, luego nos empezamos a preguntar más cosas como ¿Qué pasaría si la palabra el usuario escoja no cabe en el tablero? ya que al inicio pensamos y creamos el tablero de 10 x 10 pero luego de analizar las palabras del diccionario llegamos a que seria mejor expandir el tablero no solo para que el juego sea mas largo si no también por que nos dimos cuenta que habían palabras que no cabían en el tablero, luego de agrandar el tablero nos preguntamos ¿Cuándo una persona quiere ingresar la palabra en una posición x pero a partir de esta no cabe la palabra que pasaría? pues lo pensamos y llegamos a que el programa medirá la palabra y comparara si esta cabe en el tablero dependiendo de la posición en la que se encuentre y si esta palabra no cabe se le mostrara al usuario que la palabra no cabe y tendrá la opción de escoger de nuevo la palabra que desee y posicionarla en un mejor lugar, luego pensamos en el juego original y vimos que este tenia muchas reglas y reglas que no nos gustaron así que decidimos crear algo nuevo para innovar y por que vimos que a varias personas les gusto este idea y fue que cuando una persona va a ingresar una palabra esta no podrá estar al lado de otra deberá estar separada por un espacio ya que si esto pasaba se creaban mas palabras, pero vimos la posibilidad de cambiarlo para que esta regla no existiera y poder hacer el juego más rápido y que las personas que ganen tengan cualidades diferentes a las habituales y esto porque queríamos crear nuestras propias reglas mas avanzadas. Algo que sucedió fue que cuando creamos la matriz nos dio un poco de problema ya que debíamos de actualizar la matriz cada turno de cada jugador y además agregar unos números a los laterales de la matriz para así poder ayudar al usuario a organizar más fácilmente su palabra.

luego de tomar y pensar bien todas estas decisiones y llevarlo a código nos dimos cuenta de que el juego no tendría un final como tal así que de las 10 opciones de palabras que la persona puede escoger agregamos una opción 11 la cual da la posibilidad a la persona de pasar el turno y cuando todos los jugadores escojan esta opción en el mismo round se acabara el juego y le mostrara al usuario los puntos de cada persona. y otra decisión que tomamos para hacer el juego un poco mas largo y divertido es que cuando una persona escoja la palabra las letras que esta contenga serán cambiadas de las letras que la persona tenga disponibles esto para crear muchas más palabras y poder dar conocimiento al usuario de cuanta variedad de palabras hay.