

Jose Pablo Murillo

Bitácora

| Fecha | Decisión tomada | Comentario |
|-------------------|--|---|
| 31/08/2018 | Realizar investigación acerca de bibliotecas de reconocimiento de voz en Python | |
| 1/08/2018 | Realizar investigación acerca de bibliotecas que pudieran convertir el audio recibido a texto | |
| 4/08/2018 | Iniciar desarrollo de agente para recibir comandos | El agente recibe los valores de alto, largo y tamaño |
| 7/08/2018 | Inicio de elaboración de clase cuadrícula que se encargará de correr el algoritmo de ruta más corta | |
| 11/08/2018 | Algoritmo de ruta más corta implementado | |
| 14/08/2018 | Desarrolló de cambio de posición para piezas de inicio y de fin | |
| 17/08/2018 | Validaciones acerca de los comandos | |
| 20/08/2018 | Desarrollo de interfaz gráfica | |
| 21/08/2018 | Realizar investigación acerca de la herramienta Canvas de Tkinter para Python | |
| 24/08/2018 | Incorporar lógica de la aplicación junto con interfaz gráfica | |
| 28/08/2018 | Redacción de manual de usuario | |

Lecciones aprendidas

Las experiencias pasadas que he tenido con otros grupos de trabajo me ayudó a establecer, de primera, maneras en que se pudiera contactar y trabajar con el grupo de modo que todos estuviéramos al tanto de qué se estaba haciendo y qué faltaba por hacer. El uso de WhatsApp, ayudó a que los miembros del grupo pudiéramos tener un mecanismo oficial de comunicación y así mismo tener una manera de dejar plasmado lo que se discutía durante el desarrollo del proyecto.

La división de tareas ayudó mucho ya que los integrantes no lograron coordinar un tiempo para reunirse a programar juntos. Se estableció un estándar para que así fuera más fácil entender el código ajeno y así mantener la legibilidad y facilitar el mantenimiento de código

La investigación acerca de los reconocedores de voz me ayudaron a entender de que hay que tomar en cuenta diferentes factores a la hora de seleccionar el indicado. Reconocedores como apiai o wit ofrecen mecanismos de procesamiento de lenguaje ya incorporados en la biblioteca. Se seleccionó la biblioteca de SpeechRecognition porque este ofrecía una manera sencilla de capturar el audio del usuario que debe ser capturado en tiempo real. SpeechRecognition trabaja con diferentes wrappers para así poder utilizar diferentes reconocedores de voz como de Google, Bing, ibm, etc. Su facilidad de uso y curva de aprendizaje corta ayudó al equipo a poder incorporar el elemento de reconocedor con los otros elementos del sistema de manera sencilla y rápida.

Algo que tenía preocupado al equipo de desarrollo era el poder convertir texto a voz ya que ninguno había trabajado con ninguna biblioteca. El resultado de la investigación trajo a la luz la biblioteca de Pyttsx. Esta biblioteca usa diferentes engines basados en sistemas operativos. Incorpora nsss para sistemas operativos Mac OS, sapi5 para sistemas operativos de Windows XP, Vista y 7, y espeak para sistemas operativos basados en Linux.

En las pruebas, el algoritmo A* muestra poder presentar una solución de manera rápida en cuestión de segundos. Las pruebas incluían tableros de 20x10 y 30x20 agregando obstáculos dentro del tablero. Como el algoritmo trabaja con nodos que va recorriendo, el tiempo de computación es reducido drásticamente ya que no tiene que explorar cada camino sino que solo debe recordar por donde ha pasado.