

1. Objetivo General

- Desarrollar una aplicación que permita reafirmar el conocimiento del **paradigma de programación lógico**.

2. Objetivos Específicos

- Crear una aplicación que se comporte como un Controlador Aéreo Experto de un peculiar Aeropuerto ACE.
- Aplicar los conceptos de programación lógico.
- Crear y manipular listas como estructuras de datos.

3. Datos Generales

- El valor del proyecto: 7,5%
- **Nombre código: MayCEy**
- La tarea debe ser implementada en grupos de no más de 3 personas.
- La **fecha de entrega** es 6/Nov/2019.
- Cualquier indicio de copia será calificado con una nota de 0 y será procesado de acuerdo al reglamento.

4. Descripción del caso.

Los sistemas expertos (SE) son aplicaciones de cómputo que involucran experiencia no algorítmica, para resolver cierto tipo de problema. Por ejemplo, los sistemas expertos se usan para el diagnóstico al servicio de humanos y máquinas. Existen SE que juegan ajedrez, que plantean decisiones financieras, que configuran computadoras, que supervisan sistemas de tiempo real, que deciden políticas de seguros, y llevan a cabo demás tareas que requieren de experiencia humana.

La presente tarea tiene como objetivo desarrollar un controlador aéreo para el Aeropuerto de Computadores del Tec (ACE), la interfaz debe ser completamente natural utilizando el lenguaje español y el SE debe comportarse como un humano experto en la resolución de problemas de un controlador aéreo.

El usuario(piloto) presenta todas las características de la aeronave que pilota y pide permiso para despegar o aterrizar y dependiendo del avión el controlador aéreo le asignara una de las cuatro pistas que posee.

P1: pista de un kilómetro de largo donde pueden aterrizar/despegar aeronaves pequeñas como una Cessna, Beechcraft o un Embraer Phenom (Aviones pequeños).

P2-1 y P2-2: dos pistas de dos kilómetros de largo donde pueden aterrizar/despegar aeronaves de mediana envergadura como un Boeing 717, Embraer 190 o un Airbus A220 (Aviones Medianos), la pista P2-1 tiene una dirección de Este a Oeste y la P2-2 tiene dirección de Oeste a Este.

P3: pista de tres Kilometros de largo donde pueden aterrizar/despegar aeronaves de mediana envergadura como un Boeing 747, un Airbus A340 incluso un Airbus A380 (Aviones Grandes).

Si una Pista esta ocupada por ejemplo la P1 y un Embraer Phenom quiere aterrizar se le asignara una P2 si no hay un avión mediano solicitando por una pista P2, igual con la P3. Lo contrario no aplica si hay una emergencia en ese caso se pondrá en espera el avión de mayor envergadura para atender la emergencia del avión menor.

Otros factores aparte de la aeronave y la pista que deben tomarse en consideración para asignar una pista están, velocidad de la aeronave, velocidad y dirección del viento, peso y finalmente casos de emergencia (mayday-secuestro 7500).

Una conversación sería:

Usuario: Hola MayCEy!
MayCEy: Hola ¿en qué lo puedo ayudar?
Usuario: Solicito permiso para despegar.
MayCEy: Por favor identifíquese.
Usuario: Vuelo: 404, Aerolínea: TEC-Airlines, Matricula: Alfa Charlie Tango Lima.
MayCEy: Gracias, ¿Qué tipo de Aeronave es?
Usuario: Embraer 190
MayCEy: Por favor indique la hora de salida.
Usuario: 15:00 pm
MayCEy: Por favor indique su dirección de salida.
Usuario: Este a Oeste
MayCEy: P2-1 asignada por 5 minutos.
Usuario: Gracias.
MayCEy: Con gusto.

Usuario': Solicito permiso para despegar.
MayCEy: Por favor identifíquese.
Usuario': Vuelo: 400, Aerolínea: TEC-Airlines, Matricula: Tango Fox Lima Delta.
MayCEy: Gracias, ¿Qué tipo de Aeronave es?
Usuario': Boeing 717
MayCEy: Por favor indique la hora de salida.
Usuario': 15:00 pm
MayCEy: Por favor indique su dirección de salida.
Usuario': Este a Oeste
MayCEy: P3 asignada por 5 minutos.
Usuario': Gracias.

MayCEy: Con gusto.
Usuario: Cambio y Fuera.

Otra conversación sería:

Usuario: Hola MayCEy
MayCEy: Hola ¿en qué lo puedo ayudar?
Usuario: Quiero aterrizar. Mi matricula es Delta Echo Hotel Victor.
MayCEy: Me indica el tipo de Aeronave que es
Usuario: Airbus A220.
MayCEy: Me indica su velocidad, distancia a la pista y dirección.
Usuario: Mi velocidad es XYZ Km/H, distancia es 200Km y dirección Este a Oeste.
MayCEy: Su pista asignada es la P1-1, por favor disminuya su velocidad a DFE para que logre aterrizar a las 15:05pm posterior al vuelo Vuelo: 404, Aerolínea: ACE, Matricula: Alfa Charlie Tango Lima.
Usuario: Muchas Gracias.
MayCEy: Con gusto.
Usuario: Cambio y Fuera.

Otra conversación sería

Usuario: Mayday, mayday.
MayCEy: Buenas, por favor indique su emergencia.
Usuario: Perdí uno de los motores. Solicito permiso de aterrizar de inmediato.
MayCEy: Por favor identifíquese.
Usuario: Fox Bravo Hotel Romeo. Cessna
MayCEy: P1 asignada y equipo de ayuda enviado.
Usuario: Muchas GraCIASSS!
MayCEy: Con mucho gusto.
Usuario: Cambio y Fuera.

La presente tarea se compone de tres partes: la primera se refiere a escribir un programa que identifique palabras clave de entrada por ejemplo: inicio de una conversación [Hola|Buenos días|Buenas tardes], fin de una conversación [Gracias|Muchas gracias|Adios], preguntas [Que|Cual|Por que], emergencias [Mayday|7500].

La segunda parte es la definición de una base de datos (hechos y reglas) con al **menos** cuatro/cinco tablas principales:

Aeronaves: [Cessna, Beechcraft, Embraer Phenom, Boeing 717, Embraer 190, AirBus A220, Boeing 747, AirBus A340, Airbus A380 ...]

Pistas: [P1,P1-2(Este a Oeste),P2-2(Oeste a Este),P3...]

Emergencias: [Pérdida de motor, Parto en Medio Vuelo, Paro Cardíaco de Pasajero, Secuestro]

AtencionDeEmergencias: [Llamar a Bomberos, Llamar a médico, Llamar médico, Llamar a la OIJ y fuerza pública (ja ja ja...)]

Condiciones de aterrizaje: [Viento, Peso, LargoPista, Velocidad,...]

La tercera parte es la unión de las dos primeras partes en una interfaz de usuario donde el sistema reconoce palabras clave y determina si es una pregunta o una entrada, en caso de ser una pregunta da una respuesta al usuario utilizando la base de datos.

Es importante mencionar que el sistema puede no encontrar una palabra clave en la oración de entrada por lo que debe tener una oración conectora para poder proseguir con la conversación.

Se sugiere utilizar listas pero no se limita a estas.

5. Entregables

5.1. Código fuente comentado.

5.2. Manual de usuario.

6. Documentación

1. Se deberá entregar un documento que contenga:
 - 1.1. Manual de usuario: cómo ejecutar el programa.
 - 1.2. Descripción de los hechos y reglas implementadas.**
 - 1.3. Descripción de las estructuras de datos utilizadas.**
 - 1.4. Descripción detallada de los algoritmos desarrollados.**
 - 1.5. Problemas conocidos: En esta sección se detalla cualquier problema que no se ha podido solucionar en el trabajo y sus intentos de solución.
 - 1.6. Plan de Actividades realizadas por estudiante: Esta tarea debe realizarse antes de empezar con el proyecto y lo que busca es que se realice un plan, este debe contener al menos: la lista de actividades, responsable y fecha de entrega de la actividad.
 - 1.7. Problemas encontrados: descripción detallada, intentos de solución sin éxito, soluciones encontradas con su descripción detallada, recomendaciones, conclusiones y bibliografía consultada para este problema específico.
 - 1.8. Conclusiones y Recomendaciones del proyecto.
 - 1.9. Bibliografía consultada en todo el proyecto
2. Bitácora en digital, donde se describen las actividades realizadas, desde reuniones con el compañero de trabajo, investigaciones, consultas, etc. Se debe describir todo por más insignificante que sea, esto demostrará si ustedes están trabajando en realidad. Este es su diario de trabajo, llevan seguimiento de todo en el tiempo, imaginen que si un compañero los releva en su trabajo, le bastaría con leer sus bitácoras para seguir el trabajo.

7. Evaluación

1. El proyecto tendrá un valor de un 65% de la nota final, debe estar funcional.
2. La documentación tendrá un valor de un 20% de la nota final, cumplir con los requerimientos especificados en la documentación no significa que se tienen todos los puntos, se evaluará que la documentación sea coherente, acorde al tamaño del proyecto y el trabajo realizado, no escatimen en documentación.
3. La defensa tiene un valor de 15%, todos los miembros deben participar y todos deben conocer

- el código.
4. Cada grupo recibirá una nota en cada uno de los siguientes apartados Código y Documentación.
 5. El profesor no sólo evaluará la funcionalidad del proyecto, esto quiere decir que aunque el proyecto este 100% funcional esto no implica una nota de un 100, ya que se evaluarán aspectos de calidad de código, aplicación del **paradigma lógico** calidad de documentación interna y externa y trabajo en equipo.
 6. No se revisarán funcionalidades parciales, ni funcionalidades no integradas.
 7. Es responsabilidad de cada miembro del grupo conocer su código, el profesor puede preguntar a cualquier miembro del grupo que le explique alguna funcionalidad/porción de código.
 8. De las notas mencionadas en los puntos 1, 2 y 3 se calculará la Nota Final del Proyecto.
 9. Las citas de revisión oficiales serán determinadas por el profesor durante las lecciones o mediante algún medio electrónico.
 10. Aun cuando el código y la documentación tienen sus notas por separado, se aplican las siguientes restricciones
 - 10.1. Si no se entrega documentación, automáticamente se obtiene una nota de 0.
 - 10.2. Si no se entrega el punto 4 de la documentación se obtiene una nota de 0.
 - 10.3. Si el código y la documentación no se entregan en la fecha indicada se obtiene una nota de 0.
 - 10.4. Si el código no compila se obtendrá una nota de 0, por lo cual se recomienda realizar la defensa con un código funcional.
 - 10.5. Si el grupo no cuenta con los equipos necesarios para realizar la revisión y no avisó al profesor de esta situación obtendrá una nota de 0.
 - 10.6. El código debe ser desarrollado en **Prolog** utilizando el **paradigma de programación lógico**, en caso contrario se obtendrá una nota de 0.
 11. Cada grupo tendrá como máximo 30 minutos para exponer su trabajo al profesor y realizar la defensa de éste, es responsabilidad de los estudiantes mostrar todo el trabajo realizado, por lo cual se recomienda tener todo listo antes de ingresar a la defensa.
 12. Cada excepción o error que salga durante la ejecución del proyecto y que se considere debió haber sido contemplada durante el desarrollo del proyecto, se castigará con 2 puntos de la nota final del proyecto.
 13. Cada grupo es responsable de llevar los equipos requeridos para la revisión.
 14. Durante la revisión únicamente podrán participar los miembros del grupo, asistentes, otros profesores y el coordinador del área.
 15. Las revisiones se realizan con los estudiantes matriculados en el curso, cualquier persona fuera de estos y los mencionados en el punto 13, no pueden participar en la revisión.
 16. Después de enviada la nota final del proyecto el estudiante tendrá un máximo de 3 días hábiles para presentar un reclamo siempre y cuando la funcionalidad esté completa.

8. Referencias

Monferrer, T. E., Toledo, F., & Pacheco, J. (2001). *El lenguaje de programacion Prolog*. Valencia.

