**ANEXO A. PLAN INDIVIDUAL DE TAREAS DEL ESTUDIANTE**

|  |  |  |
| --- | --- | --- |
| **Tareas** | **Fecha de entrega** | **Rol(es) que desarrolla(n) con la tarea** |
| Reunión de inicio de la práctica. | 29/4/2024 | JP |
| Asimilación de Python como lenguaje de programación. | 4/5/2024 | AN |
| Identificar las deficiencias que presenta el código de BHCVRP. | 8/5/2024 | AS, AR |
| Investigar si existen bibliotecas en Java que tengan implementadas algunas de las heurísticas existentes en BHCVRP y que estén disponibles. | 10/5/2024 | AS, AR |
| Estudiar patrones y principios de diseño para saber cuáles nuevos se pueden incorporar en el componente. | 12/5/2024 | AR |
| Diseñar los casos de prueba necesarios para validar el funcionamiento de las heurísticas de construcción existentes en BHCVRP para el Problema de Planificación de Rutas de Vehículos con Capacidades (Capacitated VRP, CVRP). | 15/5/2024 | PB |
| Diseñar los casos de prueba necesarios para validar el funcionamiento de las heurísticas de construcción existentes en BHCVRP para el Problema de Planificación de Rutas de Vehículos con Múltiples Depósitos (Multi Depot VRP, MDVRP). | 18/5/2024 | PB |
| Diseñar los casos de prueba necesarios para validar el funcionamiento de las heurísticas de construcción existentes en BHCVRP para el Problema de Planificación de Rutas de Camiones y Remolques (Truck and Trailer Routing Problem, TTRP). | 21/5/2024 | PB |
| Diseñar los casos de prueba necesarios para validar el funcionamiento de las heurísticas de construcción existentes en BHCVRP para el Problema de Planificación de Rutas de Vehículos con Flota Heterogénea (Heterogenous Fleet VRP, HFVRP). | 24/5/2024 | PB |
| Ejecutar en una herramienta los casos de prueba diseñados para validar el funcionamiento de las heurísticas de construcción existentes en BHCVRP para los distintos VRP: CVRP, MDVRP, TTRP y HFVRP. | 27/5/2024 | PB |
| Implementar en Python las clases necesarias para modelar CVRP, MDVRP, TTRP y HFVRP. | 3/6/2024 | AS, PG |
| Implementar en Python el patrón *Factory Method* y las clases necesarias para la creación de distancias y heurísticas de construcción existentes en la versión Java, utilizando tratamiento de excepciones. | 10/6/2024 | AS, PG |
| Implementar en Python las clases necesarias para modelar las rutas, soluciones y operadores de post-optimización, utilizando tratamiento de excepciones.. | 17/6/2024 | AS, PG |
| Implementar en Python la nueva heurística de construcción Inserción de Kilby, utilizando tratamiento de excepciones. | 24/6/2024 | PG |
| Implementar en Python la nueva heurística de construcción Algoritmo de Ahorro basado en *Matching*, utilizando tratamiento de excepciones. | 1/7/2024 | PG |
| Diseñar los casos de prueba necesarios para validar el funcionamiento de las nuevas heurísticas de construcción incorporadas para diferentes instancias de VRP. | 3/7/2024 | PB |
| Ejecutar en una herramienta los casos de prueba diseñados para validar el funcionamiento de las nuevas heurísticas de construcción incorporadas para diferentes instancias de VRP. | 8/7/2024 | PB |
| Comparar los resultados obtenidos con OR-Tools, aplicando pruebas estadísticas. | 10/7/2024 | PB |
| Elaborar el informe de la práctica profesional. | 13/7/2024 | EE |
| Entregar informe final de PP al tutor. | 15/7/2024 | JP |
| Rectificar señalamientos del informe de PP. | 20/7/2024 | EE |
| Entrega del informe final de PP y preparación para la defensa. | 21/7/2024 | EE |

|  |  |  |
| --- | --- | --- |
| \_\_\_\_\_\_Isis Torres Pérez\_\_\_\_\_\_\_\_\_  Nombre completo y firma del primer tutor | \_\_\_\_\_\_\_Alejandro Rosete Suárez\_\_\_\_  Nombre completo y firma del segundo tutor |  |
| Ananda de la Caridad Morales Morales  Nombre completo del estudiante | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Firma del estudiante |

|  |
| --- |
| En la columna Rol se deben poner la sigla del rol que contribuye a formar cada tarea. Las siglas de los roles son:  **AN**- Analista de negocio **AR**- Arquitecto **JP**-Jefe de proyecto  **AS**- Analista de sistema **DS**- Diseñador de software **ES**- Especialista de seguridad  **AD**- Analista de datos **DI**- Diseñador de Interfaz hombre – máquina **EE**- Escritor-expositor de trabajos técnicos  **PG**- Programador **DB**- Diseñador de base de datos **II**- Habilitador de Infraestructuras Informáticas  **PB**- Probador **GC**- Gestor de Configuración **TD**- Facilitador de la Toma de Decisiones |

**Breve descripción del trabajo**

Se está trabajando en la Biblioteca de Heurísticas de Construcción para Problemas de Planificación de Rutas de Vehículos (BHCVRP), que consta de una primera versión en 2016, desarrollada en el lenguaje de programación Java. Esta permite la reutilización de diferentes heurísticas de construcción en diferentes problemas de planificación de rutas de vehículos. Además, está diseñada de manera robusta para permitir la incorporación flexible de nuevas heurísticas y problemas de planificación de rutas de vehículos, así como su despliegue como parte de soluciones más complejas. Actualmente, la biblioteca contiene siete heurísticas de construcción: Inserción en Paralelo de Christofides, Mingozzi y Toth (CMT), Inserción Secuencial Mole & Jameson, Vecino más cercano con lista de candidatos restringidos, Método aleatorio, Algoritmo de barrido y Algoritmo de ahorros con sus versiones secuencial y paralela. Por otra parte, resuelve cuatro variantes de VRP: Problema de Planificación de Rutas de Vehículos con Capacidades (*Capacitated VRP*, CVRP), Problema de Planificación de Rutas de Vehículos con Múltiples Depósitos (*Multi Depot VRP*, MDVRP), Problema de Planificación de Rutas de Camiones y Remolques (*Truck and Trailer Routing Problem*, TTRP) y Problema de Planificación de Rutas de Vehículos con Flota Heterogénea (*Heterogenous Fleet VRP*, HFVRP). En la actualidad, debido al crecimiento de la comunidad en Python y dado que este lenguaje es bastante utilizado en temas de Inteligencia Artificial y problemas de optimización combinatoria, se propone lanzar una primera versión del componente BHCVRP para esta tecnología.

Se planea desarrollar la biblioteca en etapas progresivas, avanzando hacia el objetivo final conforme se avance en las prácticas. En la primera etapa, que corresponde a las prácticas profesionales, se llevarán a cabo las siguientes actividades:

* Crear un conjunto de casos de pruebas con los resultados de las heurísticas de construcción existentes para distintas instancias de los VRP implementados.
* Desarrollar una primera versión de BHCVRP en Python que contenga las mismas funcionalidades de la versión existente en Java.
* Crear la nueva versión en Python, con un diseño mejorado, es decir, refactorizado, modularizado, con nuevos patrones y principios de diseño, así como tratamiento de excepciones.
* Incorporar dos nuevas heurísticas de construcción: Inserción de Kilby y Algoritmo de Ahorro basado en *Matching*.
* Comparar resultados con OR-Tools.

Esto creará las condiciones de desarrollo futuro.