

USER MANUAL – INTEGRATIVE PROJECT

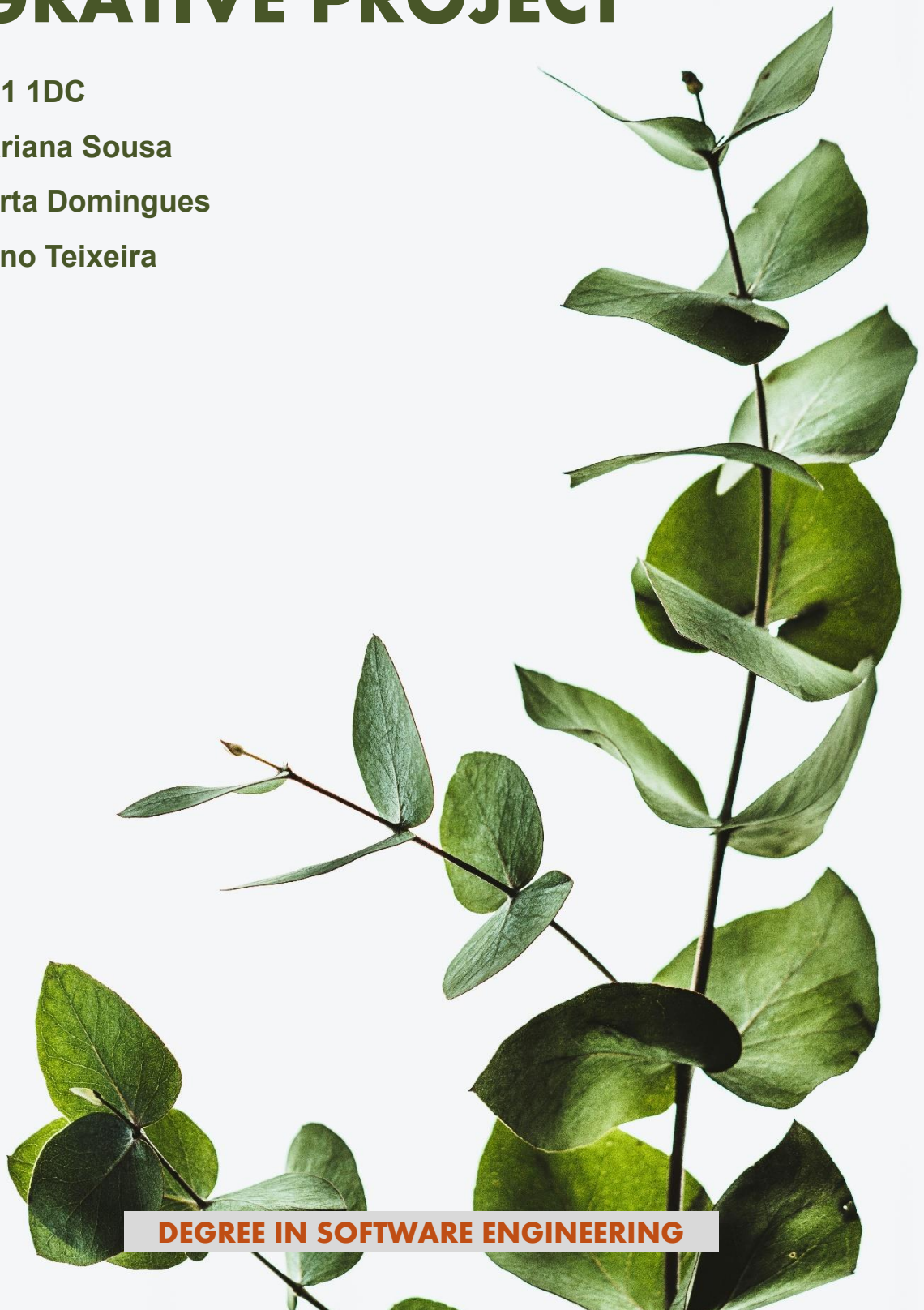
Team 3,14 G31 1DC

1230792 – Mariana Sousa

1231183 – Marta Domingues

1231375 – Nuno Teixeira

June 9, 2024



DEGREE IN SOFTWARE ENGINEERING

Index

Glossary	3
Introduction	14
System Requirements	14
System Overview.....	14
Features/Functions – step by step	18
Frequently Asked Questions	31
Troubleshooting	33
Attachments	33
Contacts/Support	48

Glossary

**Terms, Expressions and Acronyms (TEA) must be organized alphabetically.

TEA (EN)	TEA (PT)	Description (EN)	Description (PT)
Aerator	Aerador	Equipment used to increase soil oxygenation or aeration, typically employed in lawns, gardens, and cultivation areas.	Equipamento utilizado para aumentar a oxigenação ou aeração do solo, normalmente utilizado em gramados, jardins e áreas de cultivo.
Agenda	Agenda	List or plan of things to be done or considered.	Lista ou plano de coisas a serem feitas ou consideradas.
Agriculture phytopharmaceuticals	Fitofarmacêuticos agrícolas	Chemical substances used in agriculture to protect plants against pests, diseases, and weeds, as well as to promote crop growth and health.	Substâncias químicas usadas na agricultura para proteger as plantas contra pragas, doenças e ervas daninhas, bem como para promover o crescimento e a saúde das culturas.
Algorithms	Algoritmos	Step-by-step procedures for solving problems and completing tasks, essential in computer science for achieving specific outcomes.	Procedimentos passo a passo para resolver problemas e realizar tarefas, essenciais na ciência da computação para alcançar resultados específicos.
Application	Aplicação	Software designed to perform specific tasks for users on a computer or mobile device.	Software desenhado para executar tarefas específicas para utilizadores num computador ou dispositivo móvel.
Assembly Point	Ponto de Encontro	Designated location where individuals gather in case of emergency or for coordination before starting an activity.	Local designado onde indivíduos se reúnem em caso de emergência ou para coordenação antes de iniciar uma atividade.
Attribute	Atributo	Characteristic or quality that describes or identifies an object, entity, or phenomenon.	Característica ou qualidade que descreve ou identifica um objeto, entidade ou fenómeno.
Bandstand	Coreto	Structure, often in a park, used for outdoor musical performances or other events.	Estrutura, frequentemente em um parque, usada para apresentações musicais ao ar livre ou outros eventos.
Backhoe	Retroescavadora	Heavy equipment vehicle with a digging bucket at the end of a two-part articulated arm, used primarily for digging trenches and excavating soil.	Veículo pesado equipado com uma caçamba de escavação na extremidade de um braço articulado em duas partes, usado principalmente para escavar valas e retirar solo.

TEA (EN)	TEA (PT)	Description (EN)	Description (PT)
Barplot	Gráfico de barras	Graphical representation of data where bars of uniform width are used to represent the frequency or distribution of categorical or numerical variables.	Representação gráfica de dados onde barras de largura uniforme são usadas para representar a frequência ou distribuição de variáveis categóricas ou numéricas.
Bathroom	Casa de banho	Room in a public place with toilets and sinks for personal hygiene.	Espaço em um lugar público com sanitários e lavatórios para higiene pessoal.
Benches	Bancos	Long seats for multiple people, typically made of wood or metal, often found in parks or gardens.	Assentos longos para várias pessoas, geralmente feitos de madeira ou metal, frequentemente encontrados em parques ou jardins.
Blower	Soprador de folhas	Motorized device designed to blow air at high speed, typically used for removing leaves, debris, and grass clippings from outdoor areas such as yards, sidewalks, gardens, and parks.	Dispositivo motorizado projetado para soprar ar em alta velocidade, normalmente usado para remover folhas, detritos e restos de grama de áreas externas, como quintais, calçadas, jardins e parques.
Boxplot	Gráfico de caixa	Graph that shows the distribution of a dataset, highlighting the median, quartiles, and outliers.	Gráfico que mostra a distribuição de um conjunto de dados, destacando a mediana, os quartis e os outliers.
Bricklayer	Pedreiro	Responsible for assembling edifices by employing bricks, stones, or concrete blocks, adhering to architectural blueprints and construction standards to achieve structural stability and longevity.	Responsável por montar edifícios utilizando tijolos, pedras ou blocos de concreto, seguindo os projetos arquitetônicos e padrões de construção para alcançar estabilidade estrutural e longevidade.
Brush cutter	Roçadora	Cutting tools used to trim dense vegetation such as bushes, weeds, and tall grass.	Ferramentas de corte usadas para aparar vegetação densa, como arbustos, ervas daninhas e gramíneas altas.
Budget	Orçamento	Plan for managing income and expenses.	Plano para gerir rendimentos e despesas.
Business	Empresa	Organization or entity engaged in commercial, industrial, or professional activities.	Organização ou entidade envolvida em atividades comerciais, industriais ou profissionais.

TEA (EN)	TEA (PT)	Description (EN)	Description (PT)
Camelcase	Camelcase	Pattern of writing used in Java normally used to create variables. It consists of writing the first word in lower case and then writing upper case on the first letter of each word.	Padrão de escrita usado em Java normalmente utilizado para criar variáveis. Consiste em escrever a primeira palavra em minúsculas e então escrever em maiúsculas na primeira letra de cada palavra.
Cistern	Cisterna	Reervoir or tank used to store water, commonly found in residences, commercial buildings, farms, or urban areas.	Reservatório ou tanque usado para armazenar água, comumente encontrado em residências, edifícios comerciais, fazendas ou áreas urbanas.
Chainsaw	Motoserra	Portable mechanical saw that utilizes a rotating chain made up of teeth to cut through wood or other materials.	Serra mecânica portátil que utiliza uma corrente rotativa composta por dentes para cortar madeira ou outros materiais.
Closed Van	Carrinha fechada	Type of vehicle with an enclosed cargo area, typically used for transporting goods securely.	Tipo de veículo com uma área de carga fechada, geralmente usado para transportar mercadorias de forma segura.
Coefficient of skewness	Coefficiente de Assimetria	Measures the degree of asymmetry in a distribution of data.	Mede o grau de assimetria em uma distribuição de dados.
Collaborator	Colaborador	Person who is an employee of the organization and carries out design, construction, and/or maintenance tasks for green areas, depending on their skills.	Pessoa que é funcionário da organização e realiza tarefas de design, construção e/ou manutenção de áreas verdes, dependendo de suas habilidades.
Competency	Competencia	Knowledge, skills, abilities, and behaviors that enable individuals to perform tasks effectively in a specific context or role.	Conjunto de conhecimentos, habilidades, capacidades e comportamentos que permitem que indivíduos realizem tarefas de forma eficaz em um contexto ou função específica.
Construction Site	Local de construção	Area where building or infrastructure construction work is being carried out.	Área onde estão sendo realizados trabalhos de construção de edifícios ou infraestruturas.
Comma-separated values (CSV)	CSV	Represents a file format where data is organized into rows and columns, with each value separated by a comma.	Representa um formato de arquivo onde os dados estão organizados em linhas e colunas, com cada valor separado por vírgula.

TEA (EN)	TEA (PT)	Description (EN)	Description (PT)
Cycle lane	Ciclovia	Designated lane for bicycles, often separated from motor vehicle traffic, intended to provide safer cycling infrastructure.	Faixa designada para bicicletas, frequentemente separada do tráfego de veículos motorizados, destinada a fornecer uma infraestrutura de ciclismo mais segura.
Designer	Designer	Someone who creates and plans the visual appearance and functionality of objects, products, or systems.	Alguém que cria e planeja a aparência visual e funcionalidade de objetos, produtos ou sistemas.
Disc Harrow	Grade de discos	Farm implement used to break up and smooth soil, typically consisting of a series of rotating circular discs mounted on a frame.	Um implemento agrícola usado para quebrar e nivelar o solo, geralmente consistindo em uma série de discos circulares rotativos montados em um chassi.
Drainage system	Sistema de drenagem	Infrastructure designed to manage the flow of water.	Infraestrutura projetada para gerenciar o fluxo de água.
Drinking fountain	Bebedouro	Public structures designed to provide potable water for human consumption, commonly found in parks, squares, schools, train stations, and urban areas, facilitating easy access to drinking water and promoting hydration and well-being.	Estruturas públicas projetadas para fornecer água potável para consumo humano, comumente encontradas em parques, praças, escolas, estações de trem e áreas urbanas, facilitando o acesso fácil à água potável e promovendo a hidratação e o bem-estar.
Electrician	Eletricista	Someone who specializes in the installation, maintenance, and repair of electrical systems and equipment.	Alguém que se especializa na instalação, manutenção e reparação de sistemas e equipamentos elétricos.
Emergency Plan	Plano de emergência	Predetermined set of actions and procedures to follow in the event of a crisis or emergency situation.	Conjunto predeterminado de ações e procedimentos a serem seguidos em caso de crise ou situação de emergência.
Emergency Sign	Sinal de emergência	Visual indicator or symbol used to convey important information or instructions during emergency situations.	Indicador visual ou símbolo usado para transmitir informações importantes ou instruções durante situações de emergência.
Employee	Empregado	Person who works for another person or a company for wages or a salary.	Pessoa que trabalha para outra pessoa ou empresa por salário ou vencimento.

TEA (EN)	TEA (PT)	Description (EN)	Description (PT)
Equipment	Equipamento	Tools, machinery, or other durable items used in a profession or activity.	Ferramentas, máquinas ou outros itens duráveis usados em uma profissão ou atividade.
Evacuation	Evacuação	Act of moving people from a dangerous or hazardous area to a place of safety.	Ato de mover pessoas de uma área perigosa ou de risco para um local seguro.
Exercise Machine	Máquinas de exercício	Equipment designed for fitness or strength training, used indoors to perform various exercises.	Equipamento projetado para fitness ou treinamento de força, usado em ambientes fechados para realizar diversos exercícios.
Facility	Instalação	Place, building, or area that is designed and equipped to serve a specific purpose or function.	Lugar, prédio ou área projetada e equipada para servir a um propósito ou função específica.
Fence	Cercas	Structure typically made of wood, metal, or wire, used to enclose or divide an area, often used for privacy, security, or aesthetic reasons.	Estrutura geralmente feita de madeira, metal ou arame, usada para cercar ou dividir uma área, frequentemente utilizada por motivos de privacidade, segurança ou estética.
Filter (water)	Filtro (de água)	Device used to remove impurities or particles from water, improving its quality.	Dispositivo usado para remover impurezas ou partículas da água, melhorando sua qualidade.
Fleet	Frota	Group of vehicles or ships owned or operated by a company or organization for a specific purpose, such as transportation or delivery.	Grupo de veículos ou navios de propriedade ou operados por uma empresa ou organização para um propósito específico, como transporte ou entrega.
Flowerbed	Canteiro	Patch of soil in which flowers and other plants are grown, typically raised and bordered for decorative purposes.	Pedacço de solo no qual flores e outras plantas são cultivadas, geralmente elevado e cercado por motivos decorativos.
Fountain	Fonte	Ornamental structure that releases water into a basin or jets it into the air for decorative or dramatic effect, often found in parks, gardens, and public squares.	Estrutura ornamental que libera água em uma bacia ou a jorra no ar para efeito decorativo ou dramático, frequentemente encontrada em parques, jardins e praças públicas.

TEA (EN)	TEA (PT)	Description (EN)	Description (PT)
Garden	Jardim	Garden space with or without trees with little or no equipment (may have a basic irrigation system or sitting benches).	Espaço de jardim com ou sem árvores, com pouco ou nenhum equipamento (podendo ter um sistema básico de irrigação ou bancos para sentar).
Gardener	Jardineiro	Someone who cultivates and maintains plants, flowers, and landscapes, typically in gardens or outdoor spaces.	Alguém que cultiva e mantém plantas, flores e paisagens, geralmente em jardins ou espaços exteriores.
Chainsaw	Motosserra	Motorized tool used for cutting trees and wood, utilizing a toothed chain driven by a gasoline or electric motor.	Ferramenta motorizada para cortar árvores e madeira, usando uma corrente dentada movida por um motor a gasolina ou elétrico.
Graph	Grafo	Discrete structure consisting of a set of points and a set of lines joining pairs of points.	Estrutura discreta que consiste em um conjunto de pontos e um conjunto de linhas que unem pares de pontos.
Grass	Grama	Herbaceous plant with narrow leaves growing from the base, often used as a ground cover or for lawns.	Planta herbácea com folhas estreitas crescendo a partir da base, frequentemente usada como cobertura do solo ou para gramados.
Grass trimmer	Aparador de grama	Machine or tool used to trim grass, typically with a rotating blade or a line of nylon filament.	Máquina ou ferramenta utilizada para aparar a grama, normalmente com uma lâmina rotativa ou uma linha de filamento de nylon.
Green area	Área verde	Land, whether public or private, covered predominantly by grass, trees, or other vegetation, often used for recreation or conservation purposes.	Terreno, seja público ou privado, predominantemente coberto por grama, árvores ou outra vegetação, frequentemente usado para fins recreativos ou de conservação.
GSM	Gerente de Espaços Verdes	Acronym for Green Spaces Manager. Person responsible for managing the green spaces in charge of the organization.	A sigla para Gerente de Espaços Verdes. Pessoa responsável por gerenciar os espaços verdes sob responsabilidade da organização.

TEA (EN)	TEA (PT)	Description (EN)	Description (PT)
GSU	Usuário de Espaços Verdes	Acronym for Green Spaces User. Person who uses the green spaces managed by the organization and who can through the Portal, make comments or report faults in parks and gardens on the Portal.	A sigla para Usuário de Espaços Verdes. Pessoa que utiliza os espaços verdes gerenciados pela organização e que pode, através do Portal, fazer comentários ou relatar falhas em parques e jardins no Portal.
HRM	Gerente de Recursos Humanos	Acronym for Human Resources Manager. Person who manages human resources and defines teams based on the needs of ongoing projects and the skills of the employees.	A sigla para Gerente de Recursos Humanos. Pessoa que gerencia os recursos humanos e define equipes com base nas necessidades dos projetos em andamento e nas habilidades dos funcionários.
Infrastructure	Infraestrutura	Basic facilities and systems needed for a society to function, like roads, bridges, and utilities.	Instalações básicas e sistemas necessários para o funcionamento de uma sociedade, como estradas, pontes e serviços públicos..
JavaFX 11	JavaFX 11	Software development tool that provides the capability of running a program with a GUI. It is often used to develop web applications.	Ferramenta de desenvolvimento de software que fornece a capacidade de executar um programa com uma interface gráfica do usuário (GUI).
Job	Trabalho	Task or piece of work that one is employed to perform, typically for a regular wage or salary.	Tarefa ou peça de trabalho pela qual alguém é empregado para realizar, geralmente em troca de um salário ou remuneração regular.
JUnit 5 framework	Estrutura JUnit 5	Unit testing framework for Java programming language.	Estrutura de teste unitário para a linguagem de programação Java.
Fleet	Frota	Group of vehicles or ships owned or operated by a company or organization for a specific purpose, such as transportation or delivery.	Grupo de veículos ou navios de propriedade ou operados por uma empresa ou organização para um propósito específico, como transporte ou entrega.
Key Performance Indicator (KPI)	Indicador-chave de desempenho	Measurable value that indicates how effectively an organization is achieving its key business objectives.	Valor mensurável que indica quão eficazmente uma organização está alcançando seus principais objetivos de negócios.

TEA (EN)	TEA (PT)	Description (EN)	Description (PT)
Ladder	Escada	Vertical portable structure consisting of rungs or horizontal steps, designed to allow a person to reach higher areas.	Estrutura portátil vertical, consistindo de degraus ou passos horizontais, projetada para permitir que uma pessoa alcance áreas mais altas.
Lake	Lago	Fixed body of water.	Corpo de água fixo.
Liberal system	Sistema liberal	Political system that are based on liberal principles.	Sistema político baseado em princípios liberais.
Lifting Platform	Plataforma de Elevação	Multi-functional machinery that is used for lifting and loading people or goods in a vertical form.	Máquina multifuncional usada para levantar e carregar pessoas ou mercadorias de forma vertical.
Lighting System	Sistema de iluminação	Network of lighting fixtures and controls designed to illuminate indoor or outdoor spaces efficiently and effectively.	Rede de luminárias e controles projetados para iluminar espaços internos ou externos de forma eficiente e eficaz.
Machine	Máquina	Mechanical or electrical device that performs a task or function.	Dispositivo mecânico ou elétrico que realiza uma tarefa ou função.
Mean	Média	Numeric quantity representing the center of a collection of numbers and is intermediate to the extreme values of a set of numbers.	Quantidade numérica que representa o centro de uma coleção de números e está intermediária aos valores extremos de um conjunto de números.
Median	Mediana	The value separating the higher half from the lower half of a data sample, a population, or a probability distribution.	O valor que separa a metade superior da metade inferior de uma amostra de dados, uma população ou uma distribuição de probabilidade.
Meeting Point	Ponto de encontro	Meeting Point Area in a large public place where people can arrange to meet.	Área em um local público amplo onde as pessoas podem combinar de se encontrar.
Municipal Master Plans	Planos Diretores Municipais	Documents that outline the long-term development goals and strategies for a municipality or local government.	Documentos que delineiam os objetivos de desenvolvimento de longo prazo e as estratégias para um município ou governo local.
MS	MusgoSublime	Acronym for MusgoSublime. Organization dedicated to the planning, construction, and maintenance of green spaces for collective use.	Sigla para MusgoSublime. Organização dedicada ao planejamento, construção e manutenção de espaços verdes para uso coletivo.
Open Box Van	Carrinha de caixa aberta	Open body vans with no sides or roof to allow for easy loading and versatility.	Carrinhas abertas na laterais ou teto para permitir fácil carregamento e versatilidade.

TEA (EN)	TEA (PT)	Description (EN)	Description (PT)
Operating Machine	Máquinas operadoras	Heavy machinery designed to perform specific tasks.	Máquinas pesadas projetadas para realizar tarefas específicas.
Organization	Organização	Entity such as a company, or an institution comprising one or more people and having a particular purpose.	Entidade como uma empresa, ou uma instituição composta por uma ou mais pessoas e tendo um propósito específico.
Outliers	Valores atípicos	Data point that differs significantly from other observations.	Ponto de dados que difere significativamente de outras observações.
Park	Parque	Public space for recreation, usually featuring greenery, playgrounds, or walking paths.	Espaço público para recreação, geralmente com vegetação, parques infantis ou caminhos para caminhar.
Picnic Area	Área do picnic	Outdoor recreational space specifically designed for people to gather and enjoy meals, socialize, and relax in a natural setting.	Espaço recreativo ao ar livre especificamente projetado para as pessoas se reunirem e desfrutarem de refeições, socializar e relaxar em um ambiente natural.
Playground	Parque infantil	Area with structures, often resembling houses, designed as a recreational space for children, typically found in parks, playgrounds, or backyards.	Área com estruturas, frequentemente parecidas com casas, projetadas como área de recreação para crianças, geralmente encontradas em parques, playgrounds ou quintais.
Portal	Portal	Website or page on the internet that allows people, to get useful information and give their opinion about the spaces managed by the organization.	Website ou página na internet que permite às pessoas obterem informações úteis e darem sua opinião sobre os espaços gerenciados pela organização.
Power Supply	Fonte de energia	Electrical device that supplies electric power to an electrical load.	Dispositivo elétrico que fornece energia elétrica a uma carga elétrica.
Rainwater	Precipitação atmosférica	Water that falls from the atmosphere as precipitation.	Água que cai da atmosfera como precipitação
Report	Relatório	Report Structured document that provides information about a particular subject, situation, or event.	Documento estruturado que fornece informações sobre um determinado assunto, situação ou evento.
River	Rio	Large natural flowing watercourse, typically freshwater, flowing towards an ocean, sea, lake, or another river.	Grande curso de água natural, normalmente de água doce, fluindo em direção a um oceano, mar, lago ou outro rio.

TEA (EN)	TEA (PT)	Description (EN)	Description (PT)
Rural area	Área rural	Region or geographic area outside urban centers or cities.	Região ou área geográfica fora de centros urbanos ou cidades.
Scalable Vector Graphics (SVG)	Gráficos Vetoriais Escaláveis	XML-based vector image format for defining two-dimensional graphics.	Formato de imagem vetorial baseado em XML para definição de gráficos bidimensionais.
Scarifier	Escarificador	Equipment used to perform the soil scarification operation.	Equipamento usado para realizar a operação de escarificação do solo.
Semi-urban park	Parque semi-urbano	Park located on the outskirts or in proximity to urban areas, often serving as a transition between urban and rural landscapes.	Parque localizado nos arredores ou em proximidade de áreas urbanas, frequentemente servindo como uma transição entre paisagens urbanas e rurais.
Shrub	Arbusto	Small to medium-sized woody plant characterized by multiple stems and relatively low height.	Pequena a média planta lenhosa caracterizada por múltiplos caules e altura relativamente baixa.
Skill	Competência	A particular ability that is developed through training and experience and that is useful in a job.	Uma habilidade específica que é desenvolvida por meio de treinamento e experiência e que é útil em um emprego.
Software	Software	Collection of data or computer instructions that tell the computer how to work.	Coleção de dados ou instruções de computador que dizem ao computador como funcionar.
Software quality assessment team manager	Gerente da equipa de avaliação de qualidade de software	Individual responsible for overseeing the team responsible for assessing the quality of software products, ensuring adherence to standards, and implementing improvements in processes and methodologies.	Indivíduo responsável por supervisionar a equipe responsável por avaliar a qualidade dos produtos de software, garantindo aderência aos padrões e implementando melhorias nos processos e metodologias.
Swing	Baloiço	Suspended seat attached to ropes or chains, often found in playgrounds, parks, or backyard for children to play on.	Assento suspenso preso a cordas ou correntes, frequentemente encontrado em playgrounds, parques ou quintais para as crianças brincarem.
Tap (faucet)	Torneira	Valve or faucet for controlling the flow of liquid, typically water, from a pipe or container.	Válvula ou torneira para controlar o fluxo de líquido, tipicamente água, de um tubo ou recipiente.

TEA (EN)	TEA (PT)	Description (EN)	Description (PT)
Task	Tarefa	Piece of work to be done or undertaken.	Trabalho a ser realizado ou empreendido.
Team	Equipa	Group of individuals working together to achieve a common goal.	Grupo de indivíduos trabalhando juntos para alcançar um objetivo comum.
Toilet	Sanita	Fixture used for human waste disposal, typically consisting of a bowl and seat, and connected to a sewage system or septic tank.	Instalação usada para o descarte de resíduos humanos, tipicamente consistindo de uma bacia e assento, e conectada a um sistema de esgoto ou tanque séptico.
UN	Nações Unidas	Acronym for United Nations. International organization established to promote peace, cooperation, and sustainable development among nations worldwide.	A sigla para Nações Unidas. Organização internacional estabelecida para promover a paz, cooperação e desenvolvimento sustentável entre as nações em todo o mundo.
Urban park	Parque Urbano	Park located within urban areas, providing recreational and green spaces for residents and visitors.	Parque localizado dentro de áreas urbanas, fornecendo espaços recreativos e verdes para residentes e visitantes.
Vehicle	Veículo	Machine, usually motor-driven, used for transportation of people or goods.	Máquina, geralmente motorizada, usada para o transporte de pessoas ou bens.
VFM	Gestor de Frota de Veículos e Equipamentos	Acronym for Vehicle and Equipment Fleet Manager. Person who manages the fleet park, the machines, equipment, and vehicles, ensuring their good condition and assigning them to the tasks to be carried out.	A sigla para Gerente de Frota de Veículos e Equipamentos. Pessoa responsável por gerenciar o parque de frota, as máquinas, equipamentos e veículos, garantindo seu bom estado e atribuindo-os às tarefas a serem realizadas.
Water fountain	Chafariz	Ornamental structure that releases water into a basin or jets it into the air for decorative or dramatic effect, often found in parks, gardens, and public squares.	Estrutura ornamental que libera água em uma bacia ou a jorra no ar para efeito decorativo ou dramático, frequentemente encontrada em parques, jardins e praças públicas.
Weeder	Arrancador de ervas daninhas	Equipment used to remove weeds, which are unwanted plants growing in cultivated areas, gardens, or plots of land.	Equipamento utilizado para remover ervas daninhas, que são plantas indesejadas crescendo em áreas cultivadas, jardins ou terrenos.

Introduction

This document is intended to provide comprehensive guidance on how to use the developed software, covering everything from a general overview of the product to detailed instructions on user-specific functionalities. It caters to a wide range of user demographics, ensuring that all users can interact effectively with the application.

This manual is intended for individuals involved in the management of urban green spaces, such as human resources managers, fleet managers for vehicles and equipment, and collaborators responsible for maintaining green areas and related activities. It aims to provide valuable insights and operational instructions to maximise the utility of the software.

System requirements

The developed software requires a system with Java 8 or higher, Graphviz 10.0.1 or higher, a computer with at least 2GHz CPU (integrated CPU), 4GB RAM or more, and 5GB of free disk space (program installation not included).

System Overview

GENERAL DESCRIPTIONS

This application is a comprehensive global management tool for urban green spaces like parks, gardens, and public areas. Its primary goal is to streamline and improve the administration and maintenance of these green spaces, ensuring efficient resource management, sustainability, functionality, and aesthetic appeal.

This software will simplify and clarify various aspects of green space management, such as collaborator management, including their professions and relevant skills; work team management; vehicle and machinery management; vehicle maintenance; green space management; task management (agenda and to-do list); and notification service.

MAIN FEATURES

The main features of the application are:

- **Employee Registration and Management:**

The application simplifies the process of registering and managing employee information. It enables users to enter and maintain detailed records such as skills, qualifications, and work history. It centralises this data, allowing for efficient task and role assignment based on employee competencies, ensuring optimal workforce utilisation.

- **Vehicle Registration and Management:**

The app makes it simple to register and manage the vehicles required for green space management operations. This feature allows you to track vehicle details like the model, registration information, and maintenance schedules. By organising maintenance checks and vehicle-related tasks, the developed software ensures that vehicles are well-maintained, reducing downtime and increasing operational efficiency.

- **Team Proposal Generation:**

The platform facilitates the formation of work teams by automatically generating proposals based on required competencies. This feature evaluates the skills and qualifications of available employees and recommends the best team composition for specific tasks or projects. By optimising workforce allocation, the platform contributes to efficient and effective project completion.

- **Task Management:**

This programme provides tools for effective task management, which is critical for the timely maintenance of green spaces. Managers can use the To-Do List feature to register and track all necessary tasks, categorising them based on urgency and duration. The Agenda feature allows for task scheduling and monitoring, team assignment, and vehicle and equipment allocation, resulting in more comprehensive and organised task execution and improved overall green space functionality.

- **Green Space Management:**

This tool enables managers to register various green spaces, specifying their type (garden, medium-sized park, or large-sized park) and respective areas. This feature ensures that all green spaces are properly documented and easily accessible for management.

- **Team Assignment Notification:**

When a team is assigned a task in the Agenda, the software sends an email to all team members right away. This notification system supports various email services and is configurable to allow for flexibility in communication platforms, ensuring that all team members receive timely updates regardless of the email service used.

- **Data Analysis:**

The service provides advanced data analysis tools to help decision-makers with green space management.

- **Water Consumption Cost Analysis:**

The app includes tools for estimating water consumption costs in specific green spaces. By collecting and analysing water usage data, this feature assists managers in identifying areas of high consumption and potential inefficiencies. Managers can use cost analysis insights to optimise water usage, reduce costs, and promote sustainability.

- **Equipment Usage Analysis:**

The application's equipment usage analysis feature allows managers to monitor and analyse the use of equipment in green spaces. This feature provides valuable insights into equipment efficiency and effectiveness by monitoring variables such as equipment utilisation rates, maintenance schedules, and performance metrics. This information enables managers to make informed decisions about equipment maintenance, replacement, and utilisation strategies.

- **Monthly Water Consumption Cost Prediction:**

The software uses a linear regression model to estimate the average monthly water consumption cost for new parks based on their size. This feature uses historical data to provide accurate cost estimates, which help with budget planning and resource allocation.

- **Polynomial Regression for Data Fit:**

The service uses polynomial regression to find the best-fit line for each dataset. This feature enables managers to analyse complex data relationships, resulting in more accurate predictions and informed decisions.

- **User Portal Data Collection:**

The software makes it easier to collect data from user portals about park usage. This feature allows managers to collect information about visitor demographics, usage patterns, and preferences. Understanding how users interact with green spaces enables managers to tailor maintenance and management strategies to meet park visitors' needs and expectations.

- **Optimal Route Determination:**

The app uses advanced algorithms to calculate the best routes for installing pipes in green spaces. This feature ensures that water resources are distributed efficiently and effectively by taking into account terrain, distance, and water supply requirements. Optimised route planning helps to develop robust irrigation systems that meet the needs of green spaces while reducing resource waste.

- **Shortest Route to Assembly Points:**

This tool assists in defining and placing signs for the shortest evacuation routes from various points throughout the park to designated assembly points. The feature employs custom algorithms to identify and visualise the shortest paths, resulting in efficient and safe evacuations.

Features/Functions – Step by Step

LOGIN | ALL USERS

This procedure is required for all users wishing to use the software's functions.

Here are the **Step-by-Step** Guidelines:

1. Open the application from your desktop.
2. Select Login in the Main Menu.
3. Enter your administrator-registered email address and password.
4. Click the login button and wait for the page to load.

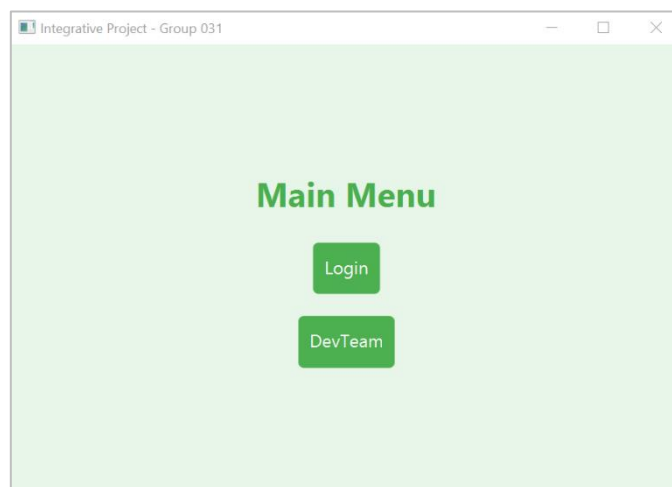


Illustration 1 – Main Menu view

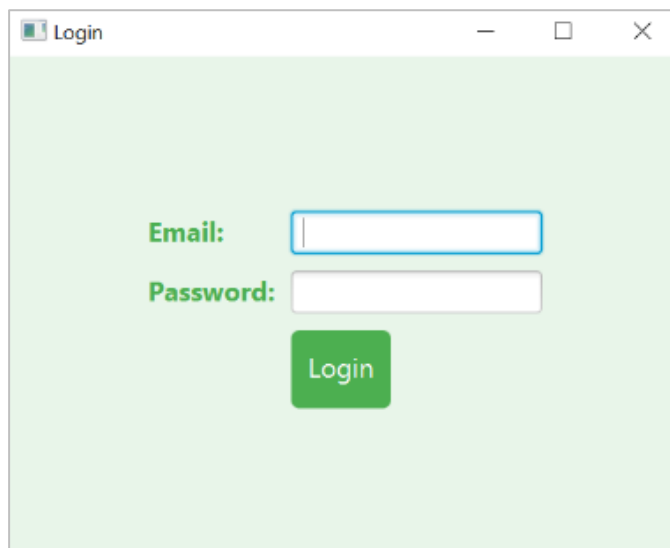


Illustration 2 - Login view

REGISTER SKILLS FOR COLLABORATOR | HUMAN RESOURCES MANAGER

This feature allows the Human Resources Manager (HRM) to add skills to the system database.

Here are the **Step-by-Step** Guidelines:

1. Open the application and login as HRM.
2. Inside of the HRM Menu select Create a Skill.
3. Write the name of the Skill that it's going to be added.
4. Click on the option Create Skill.
5. Upon the creation, the system will display the skill indicating the successful creation of the new skill.

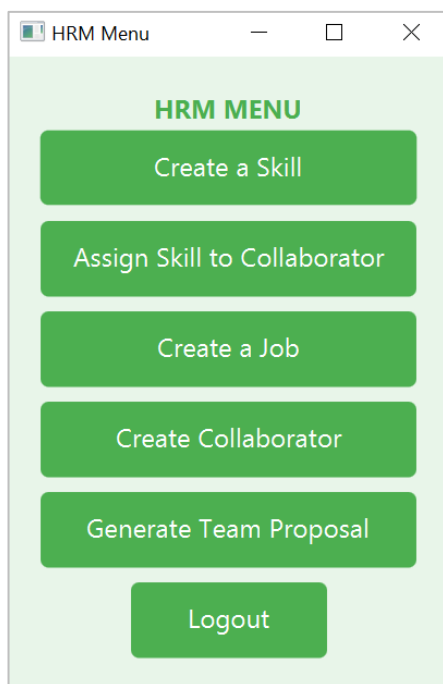


Illustration 3 – HRM Menu

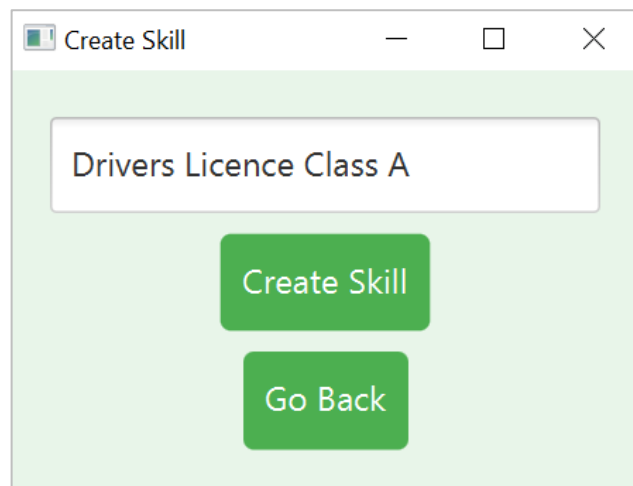


Illustration 4 – Create Skill

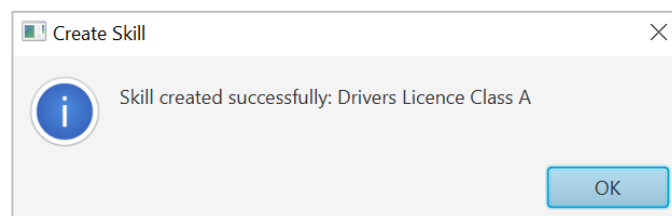


Illustration 5 – Confirmation Message

REGISTER JOB FOR COLLABORATOR | HUMAN RESOURCES MANAGER

This feature allows HRM to register a job in the system.

Here are the **Step-by-Step** Guidelines:

1. Open the application and login as HRM.
2. In the HRM Menu select “Create a Job”.
3. Write the name of the Job that it’s going to be added.
4. Click on the option “Create Job”.
5. Upon the creation, the system will display the job indicating the successful creation of the new skill.

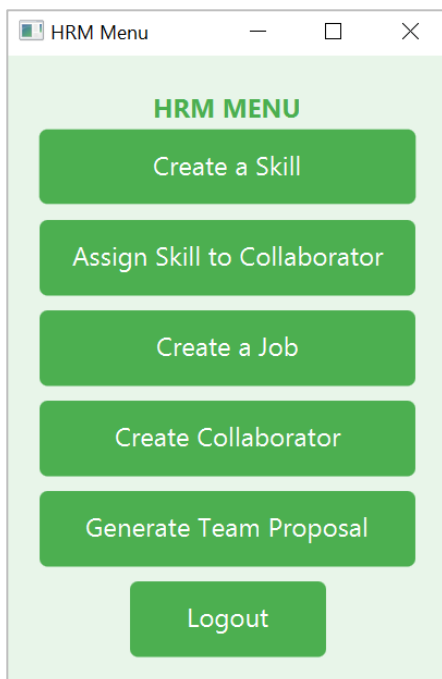


Illustration 3 – HRM Menu

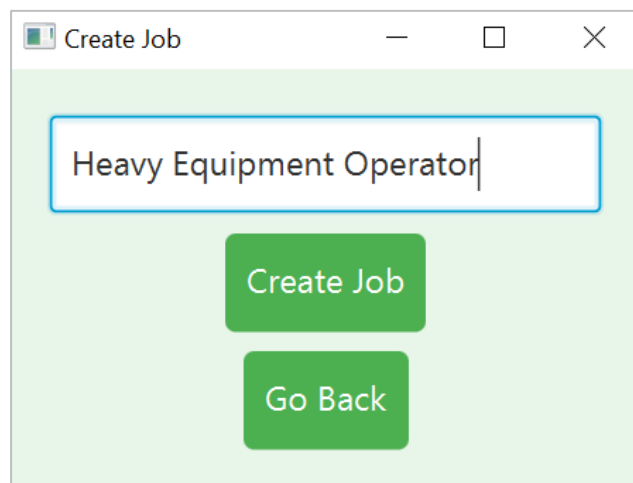


Illustration 6 – Create Job

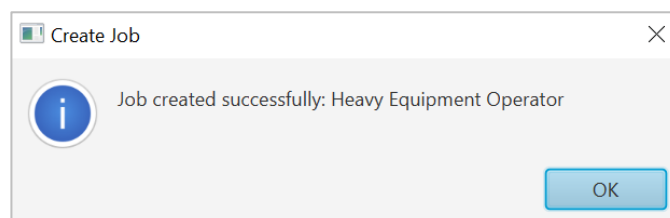


Illustration 7 – Confirmation Message

REGISTER COLLABORATOR WITH A JOB AND CHARACTERISTICS | HUMAN RESOURCES MANAGER

This feature allows the HRM to register a collaborator in the system.

Here are the **Step-by-Step** Guidelines:

1. Open the application and login as HRM.
2. In the HRM Menu go to “Create Colaborator.”
3. Enter the requested data for the new collaborator, including personal information and assigned job.
4. After entering the data, review all the data.
5. Confirm the provided data by selecting “Submit”.
6. Upon confirmation, the system will display the successful message.

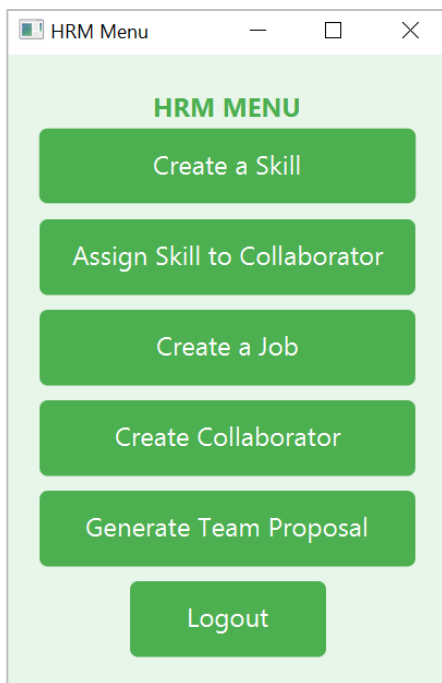


Illustration 3 – HRM Menu

Illustration 9 – Data for the creation of Collaboration and Assign Job

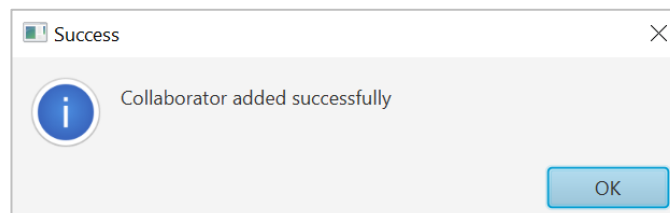


Illustration 10 – Confirmation Message

ASSIGN SKILL TO AN COLLABORATOR | HUMAN RESOURCES MANAGER

This feature allows HRM to assign one or more skills to a collaborator.

Here are the **Step-by-Step** Guidelines:

1. Open the application and login as HRM.
2. In the HRM Menu go to “Assign Skill to Collaborator”
3. Select one collaborator from the list.
4. Click on the skill(s) that it’s going to be added to the collaborator.
5. Select “Assign Skill”
6. Upon confirmation, the system will display the successful message with the confirmation of the assigned skill(s).

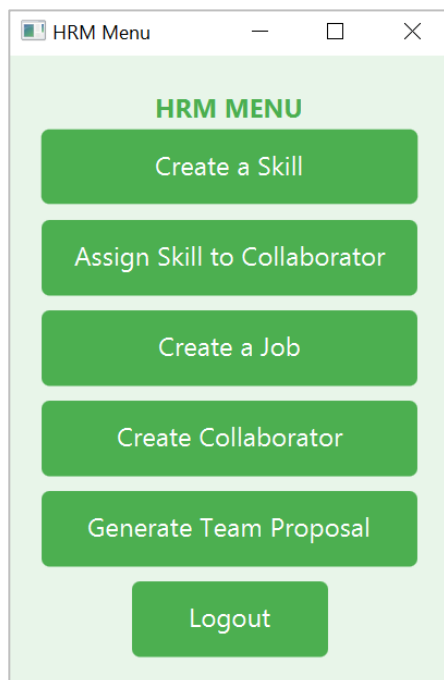


Illustration 3 – HRM Menu

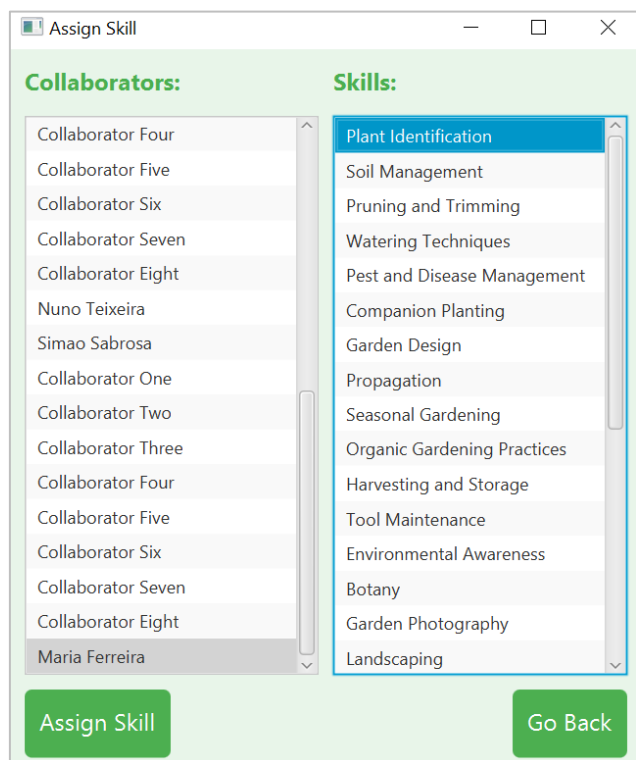


Illustration 11 – Assign Skill

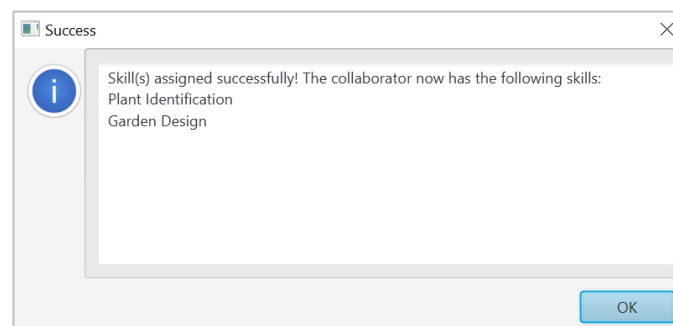


Illustration 12 – Confirmation message

GENERATE TEAM PROPOSAL | HUMAN RESOURCES MANAGER

This feature allows HRM to generate a team proposal.

Here are the **Step-by-Step** Guidelines:

1. Open the application and login as HRM.
2. Select “Generate Team Proposal”.
3. Enter the requested data, minimum number of collaborators, maximum number of collaborators.
4. From the list of skills that are available, select the one’s that are going to be added.
5. Click on “Generate Team Proposal”

(if needed the user can eliminate all the skills that were added by clicking in “clear skills” and start over)

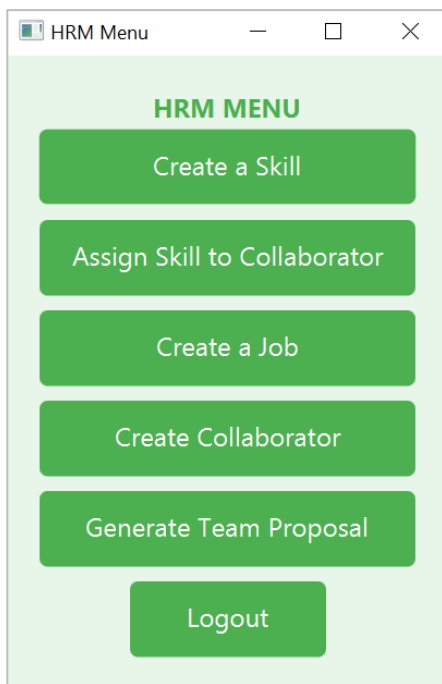


Illustration 3 – HRM Menu

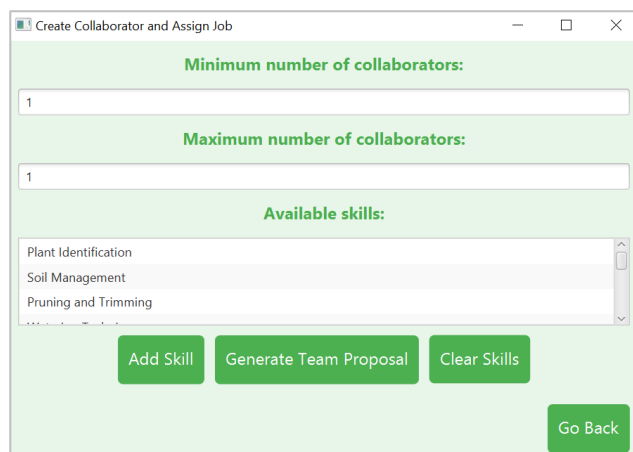


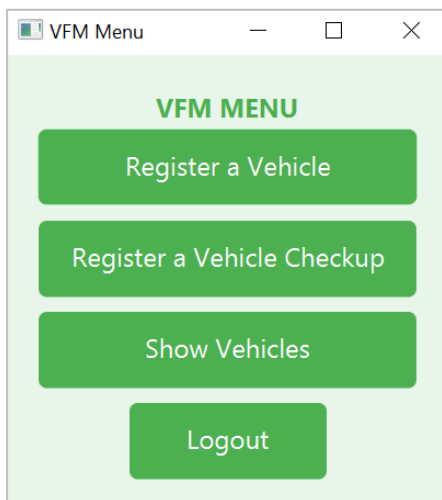
Illustration 13 – Create Collaborator and Assign Job

REGISTER A VEHICLE | VEHICLE AND EQUIPMENT MANAGER

This feature allows VFM to register a vehicle with various characteristics.

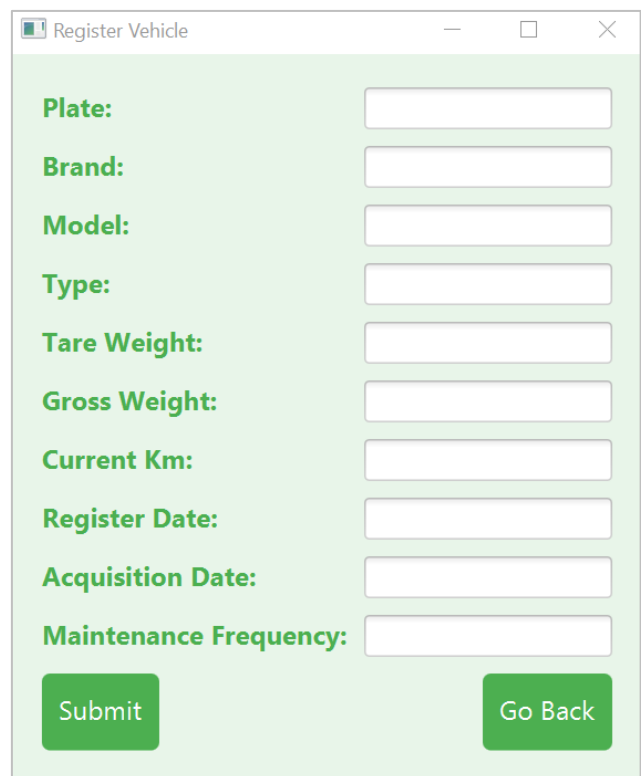
Here are the **Step-by-Step** Guidelines:

1. Open the application and login as VFM
2. Select “Register a Vehicle Checkup”
3. The system will request the vehicle data such as Vehicle Plate, Brand, Model, Type, Tare Weight, Gross Weight, Current Km, Register Date, Acquisition Date, Maintenance Frequency.
4. After that the system will display the successful message.



The screenshot shows a window titled "VFM Menu". Inside, there is a green header with the text "VFM MENU". Below the header, there are four green buttons stacked vertically: "Register a Vehicle", "Register a Vehicle Checkup", "Show Vehicles", and "Logout".

Illustration 14 – VFM Menu



The screenshot shows a window titled "Register Vehicle". It contains a list of labels followed by input fields: "Plate:", "Brand:", "Model:", "Type:", "Tare Weight:", "Gross Weight:", "Current Km:", "Register Date:", "Acquisition Date:", and "Maintenance Frequency:". At the bottom, there are two green buttons: "Submit" and "Go Back".

Illustration 15 – Register Vehicle

REGISTER A VEHICLE'S MAINTENANCE | VEHICLE AND EQUIPMENT MANAGER

This feature allows VFM to register a vehicle's maintenance.

Here are the **Step-by-Step** Guidelines:

1. Open the application and login as VFM.
2. Select "Register a Vehicle Checkup".
3. The system will request data like, the vehicle plate, the checkup date and the km at checkup.
4. Click "Submit".
5. Upon confirmation, the system will display the successful message.

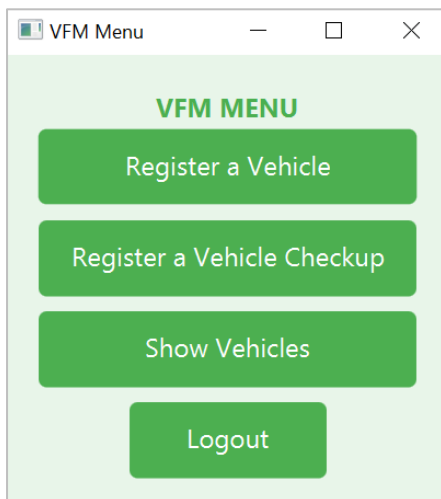


Illustration 14 – VFM Menu

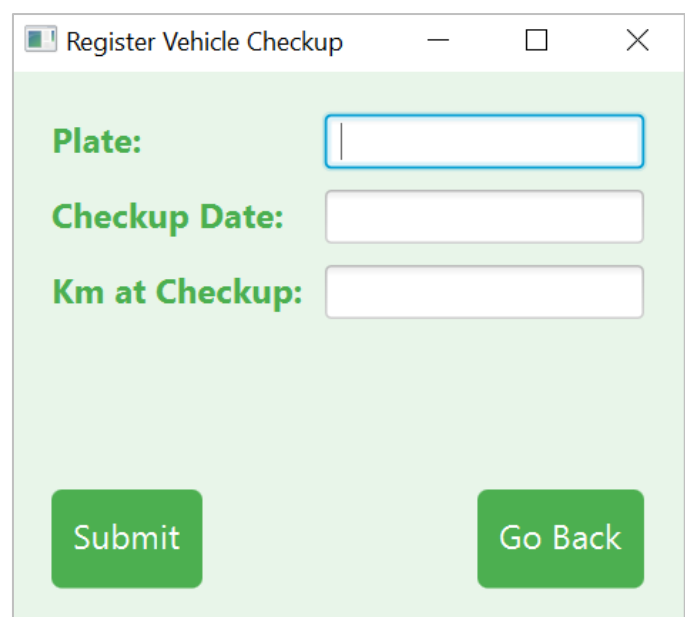
The screenshot shows a window titled "Register Vehicle Checkup". It has a light green background. There are three input fields with green labels: "Plate:", "Checkup Date:", and "Km at Checkup:". Below these fields are two green buttons: "Submit" and "Go Back".

Illustration 16 – Register Vehicle Checkup

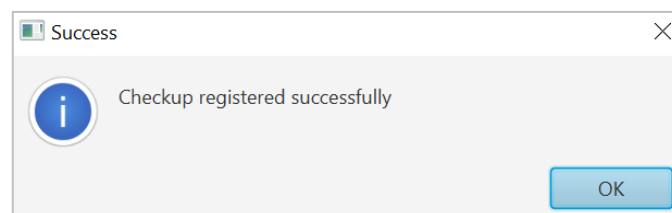


Illustration 17 – Success message

LIST THE VEHICLES NEEDING THE MAINTENANCE | VEHICLE AND EQUIPMENT MANAGER

This feature allows VFM to list the vehicles needing a maintenance.

Here are the **Step-by-Step** Guidelines:

1. Open the application and login as VFM.
2. Select “Show Vehicles”.
3. The system displays a list of registered vehicles, and a list of vehicles needing maintenance.

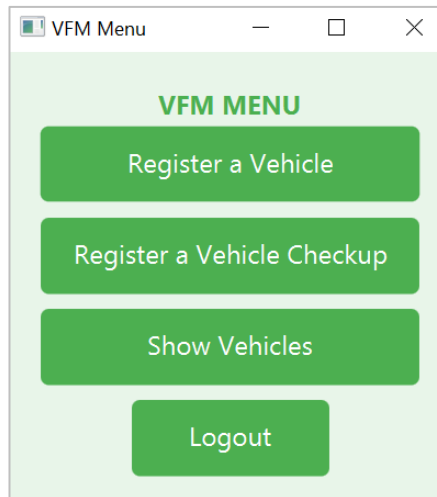


Illustration 14 – VFM Menu

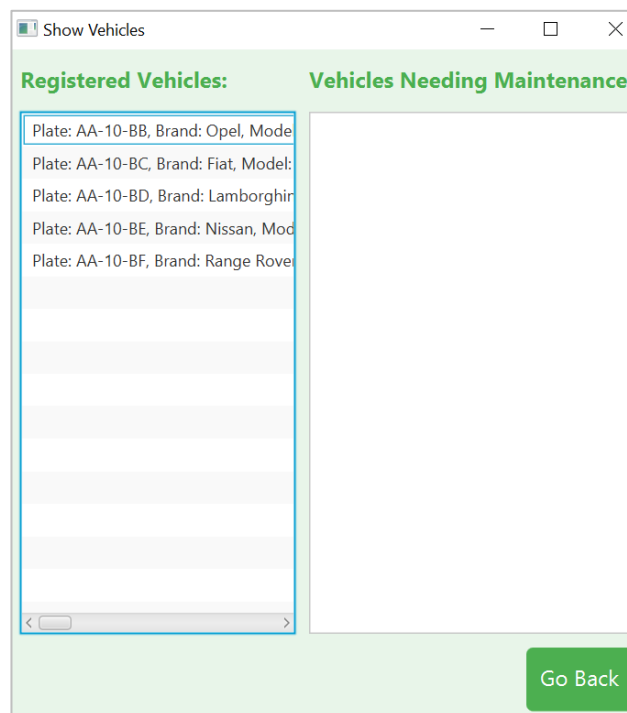


Illustration 18 – Show Vehicles

REGISTER A GREEN SPACE | GREEN SPACES MANAGER

This feature allows the GSM to register a green space (garden, medium-sized park, or large-sized park) and its respective area.

Here are the **Step-by-Step** Guidelines:

1. Open the application and login as GSM.
2. Select “Register Green Space”.
3. The system will request the green space name, its street name, its type and area.
4. After filling all that information click “Register Green Space”.
5. Upon confirmation, the system will display the successful message.

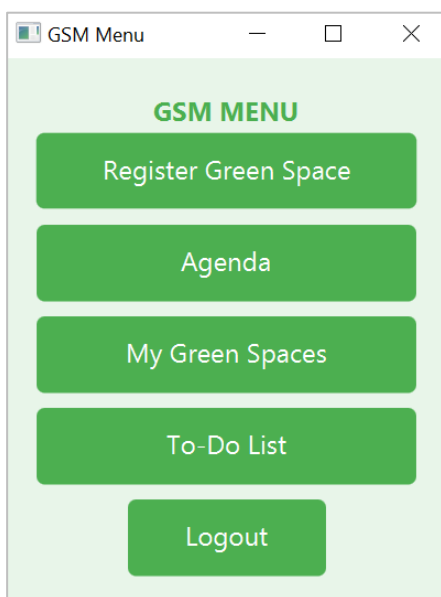


Illustration 19 – GSM Menu

A screenshot of a web application window titled "Register Gre...". The window has a light green background. It contains four input fields with green labels: "Green Space Name:", "Green Space Street:", "Green Space Type:" (which is a dropdown menu), and "Green Space Area:". Below these fields is a green button with white text: "Register Green Space". At the bottom right, there is a green button with white text: "Go Back".

Illustration 20 – Register Green Space

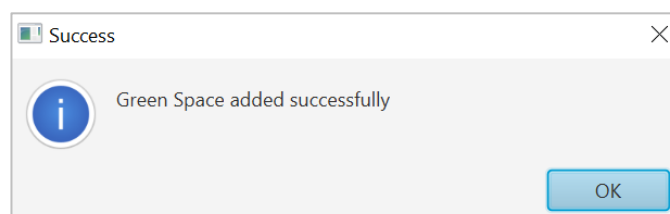


Illustration 21 – Success Message

ADD A NEW ENTRY TO THE TO-DO LIST | GREEN SPACES MANAGER

This feature allows the GSM to add a new entry to the To-Do-List.

Here are the **Step-by-Step** Guidelines:

1. Open the application and login as GSM.
2. Select “To-Do List”.
3. Click on the option “New Entry”.
4. The system will request the entry details such as task name, a task description, task duration, Task GreenSpace, task UrgencyDegree.
5. Upon confirmation, the system will display the successful message.

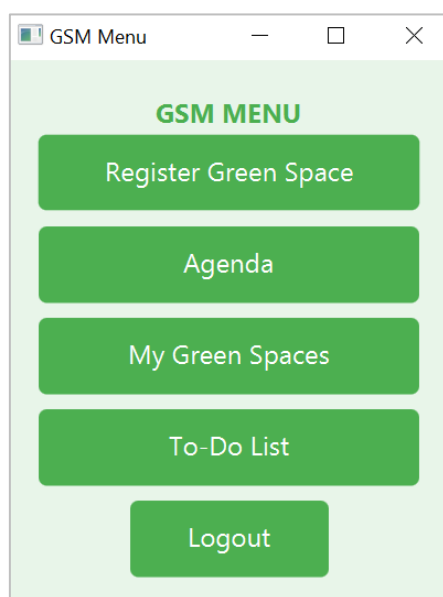


Illustration 19 – GSM Menu

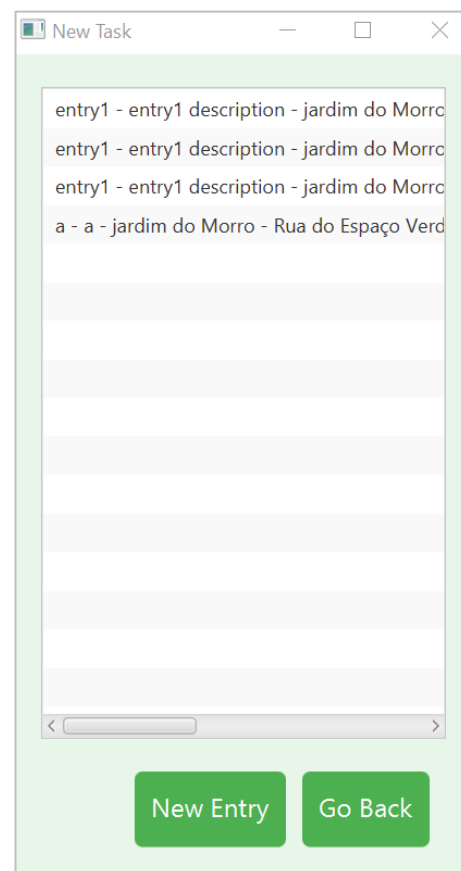


Illustration 22 – New Task

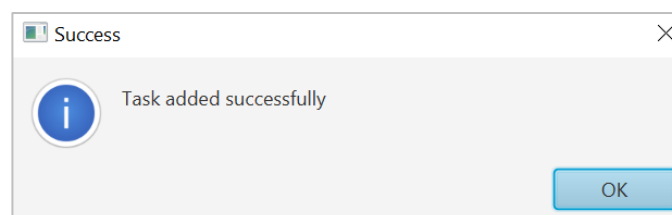


Illustration 23 – Success message

ADD A NEW ENTRY TO THE AGENDA | GREEN SPACES MANAGER

This feature allows the GSM to add a new entry to the agenda.

Here are the **Step-by-Step** Guidelines:

1. Open the application and login as GSM.
2. Select “Agenda”.
3. Click on the option “New Entry”.
4. Put the entry date and entry task, and submit.
5. Select the entry.
6. When selected the entry it will give the option to postpone, deactivate, assign team and assign vehicle(s).

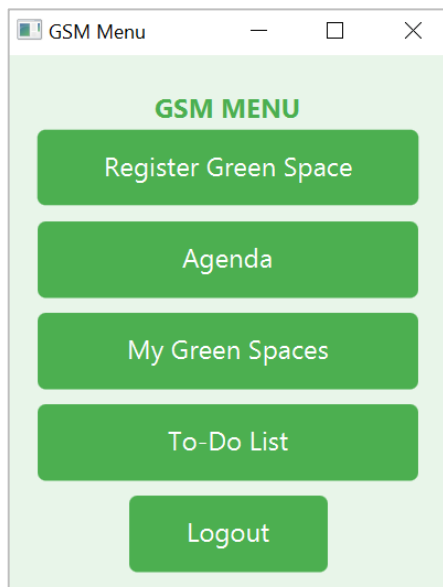


Illustration 19 – GSM Menu

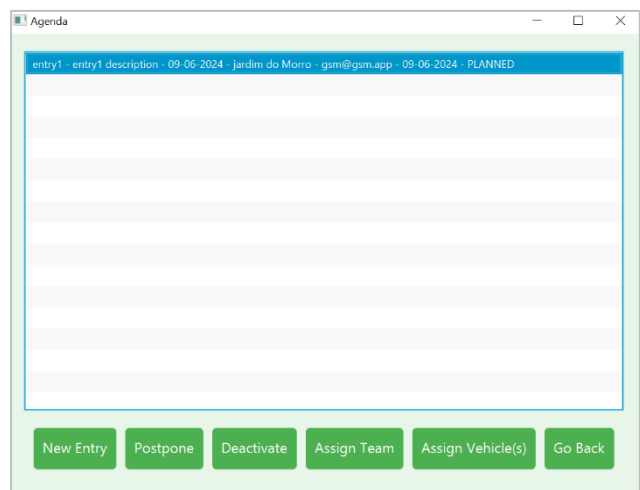


Illustration 24 – Agenda

LIST OF ALL GREEN SPACES MANAGED BY THE GSM | GREEN SPACES MANAGER

This feature allows the GSM to list all green spaces managed by them.

Here are the **Step-by-Step** Guidelines:

1. Open the application and login as GSM.
2. Select “My Green Spaces”.
3. The system will automatically display an updated list of green spaces managed by GSM.

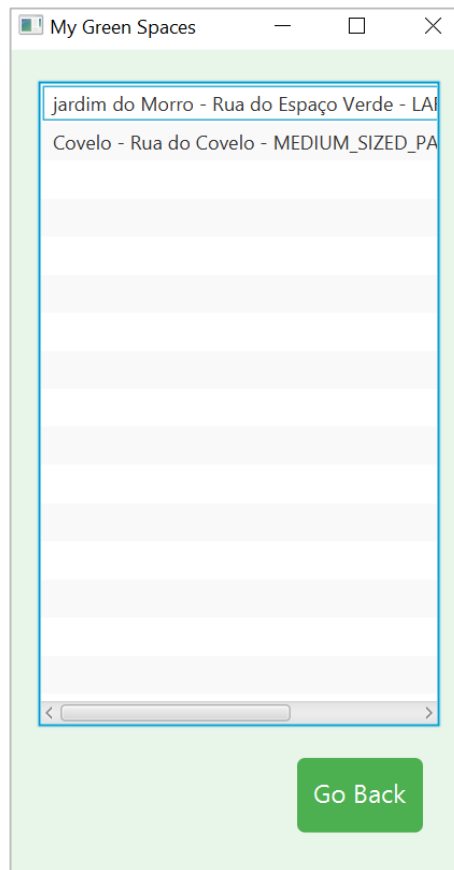


Illustration 25 – My Green Spaces

Frequently Asked Questions

This Frequently Asked Questions (FAQ) section for the developed software aims to address common questions and provide simple answers to help users navigate the platform smoothly. Whether you're new to the app or need clarification on a specific feature, this FAQ section provides concise answers to your questions. Explore the questions below to get the answers you need to improve your experience.

Question: What's the developed software and what does it do?

Answer: The application is a comprehensive management tool for urban green spaces, encompassing a wide range of functionalities to simplify and streamline the administration, maintenance, and resource management of parks, gardens, and public areas. It aims to promote sustainability, functionality, and aesthetic appeal in these green spaces.

Question: Who is the application aimed at?

Answer: The app is designed for various user groups involved in urban green space management, including Human Resources Managers (HRM), Vehicle and Equipment Fleet Managers (VFM), Green Space Managers (GSM) and Collaborators (CLB).

Question: Can I use the app on my mobile device?

Answer: Currently is primarily designed for desktop use.

Question: Can I use the software to manage tasks offline?

Answer: Currently it does not require an internet connection for full functionality.

Question: How should the CSV for the pipe routes be formatted?

Answer: The CSV file must contain the following organization on each line: "Water Point X;Water Point Y;Distance". For example: "A;B;13".

Question: How should the CSV for the cost matrix be formatted?

Answer: The CSV file must contain the following organization:

0;3;9;1

3;0;0;2

9;0;0;5...

Question: How should the CSV for the points names be formatted?

Answer: The CSV file must contain the following organization:

PointA;PointB;AP;PointD;AP1;...

Question: What should the zip code fields look like?

Answer: All postal codes must be in the following format: "1234-123".

Question: What should the VIN fields look like?

Answer: The VIN fields must have 17 alphanumeric characters.

Question: How should the plates be inserted?

Answer: The plates have 6 alphanumeric characters and must be valid. For example: “00AA00” for a vehicle registered between 2000 and 2005.

Question: How many digits does a Citizen Card number have?

Answer: A Citizen Card number consists of 9 digits followed by 2 letters and 1 digit, making a total of 12 characters.

Question: What format does a passport number follow?

Answer: A Passport number should be 6 to 9 alphanumeric characters.

Question: What is the format for an Identity Card number?

Answer: An Identity Card number consists of exactly 9 digits.

Question: Can an Other Type document number contain special characters?

Answer: No, an Other Type document number should contain only alphanumeric characters.

Question: Can an employee/collaborator be of any age?

Answer: No, an employee/collaborator must be at least 18 years old. The system checks that the difference between the registration date and the admission date is at least 18 years.

Troubleshooting

PROBLEM	POSSIBLE CAUSE	SOLUTION
If you cannot access your account.	Incorrect credentials (email or password).	Verify that your email and password are correct. Contact the administrator to make sure your account is activated.
If you cannot access Collaborator Section.	Insufficient permissions.	Verify that you are logged in as an HRM.
If you cannot access Tasks Section.		Verify that you are logged in as an GSM.
If you cannot access Vehicles Section		Verify that you are logged in as an VFM.
If you cannot access Spaces Section.		Verify that you are logged in as an GSM.
If you cannot access Team Section.		Verify that you are logged in as an HRM.
If you cannot register a new collaborator.	Required field not filled out.	Ensure all required fields are filled in, and valid before confirmation.
	Required field filled with Invalid inputs.	
	Collaborator already registered.	
If you cannot register a new skill.	Required field not filled in.	Ensure all required fields are filled out before confirmation. Check that there isn't already a skill with the same name.
	Required field filled with Invalid inputs.	
	Skill already registered.	
If you cannot register a new vehicle.	Required field not filled out.	Ensure all required fields are filled before confirmation. Check that there isn't already a skill with the same VIN or Plate.
	Required field filled with Invalid inputs.	
	Vehicle already registered.	
If you cannot register a new Maintenance.	No vehicle selected in vehicle list.	Select a vehicle in the vehicle list before clicking on option add maintenance.
	Required field not filled in.	Ensure all required fields are filled out before confirmation.
	Invalid inputs.	Ensure all required fields are valid before confirmation.
	Maintenance already registered.	
If you cannot register a new green space.	Required field not filled in.	Ensure all required fields are filled in before confirmation.
	Invalid inputs.	

	Green space already registered.	Check that there isn't already a Green Space with the same attributes.
If you cannot register an entry in the To-Do-List.	Required field not filled out.	Ensure all required fields are filled out before confirmation.
	Invalid inputs.	
If you cannot register an entry in the agenda.	No entry selected in To-Do list.	Select an entry in the To-Do list after clicking the option add new entry.
	Required field not filled out.	Ensure all required fields are filled out before confirmation.
	Invalid inputs.	Ensure all required fields are valid before confirmation.
	Entry already registered.	
If you cannot see the option assign Skill.	No Collaborator selected in collaborator list.	Select a collaborator in the collaborator list to see the option assign skill.
If you cannot assign a skill to a collaborator.	Required field not filled out.	Ensure all required fields are filled out before confirmation.
	Required field filled with Invalid inputs.	
	Skill already assigned.	Ensure all required fields are valid before confirmation.
If you cannot assign a vehicle to an entry in agenda.	No vehicle selected.	Select a vehicle after clicking the option assign vehicle.
If you cannot assign a team to an entry in agenda.	No entry selected in agenda.	Select an entry in the agenda and after, click the option assign team.
If you cannot Postpone an entry in agenda.	No entry selected in agenda.	Select an entry in the agenda and after that click the option to Postpone entry.
	Required field filled with invalid inputs.	
If you cannot complete an entry in agenda.	No entry selected in agenda.	Select an entry in the agenda and, after that, click in the option to complete Entry.

Attachments

MATCP

US15

Introdução

Na User Story 15 (US15), o objetivo principal é compreender a relação entre a área dos parques e o custo mensal associado ao consumo de água, possibilitando uma gestão mais eficiente dos recursos financeiros, sendo que é precisa estimar o custo médio mensal da água para um novo parque de **55 hectares**. Para isso, utilizaremos um modelo de regressão linear, considerando a área do parque como a variável independente e o custo médio mensal de água como a variável de resposta.

Os dados necessários para essa análise estão contidos em dois arquivos:

- **"water_consumption_updated.csv"**: Contém registos diários de consumo de água (em metros cúbicos) para diversos parques.
- **"Area.csv"**: Regista a área (em hectares) de cada parque.

O custo da água segue as regras definidas na US09:

- Para consumos mensais até 1000 m³ o custo é de **0,7 €/m³**.
- Para consumos superiores a este limite é aplicada uma taxa adicional de **15%** sobre o excedente.

Resultados esperados

A análise procura alcançar os seguintes resultados:

1. Previsão do custo mensal para um novo parque de 55 hectares:

Utilizando um modelo de regressão linear, prevemos o custo médio mensal de água para um parque de 55 hectares. Esta previsão é baseada na relação entre a área do parque e os custos médios mensais observados nos dados históricos.

-Input: Arquivos "water_consumption_updated.csv" e "Area.csv"

-Output: Previsão do custo mensal médio de água para um parque de 55 hectares

2. Visualização da relação entre a área do parque e o custo mensal:

Geração de um gráfico de dispersão que representa a relação entre a área dos parques e o custo mensal de água, acompanhado pela linha de regressão ajustada.

-Input: Arquivos "water_consumption_updated.csv" e "Area.csv"

-Output: Gráfico de dispersão com a linha de regressão

3. Análise da regressão linear:

A equação da reta de regressão permite-nos prever o custo médio mensal de água para um novo parque com base na sua área. A avaliação do modelo de regressão linear através do coeficiente de correlação (r) e do coeficiente de determinação (r^2). Estas medidas ajudam a entender a força e a precisão da relação entre a área do parque e o custo mensal de água.

-Input: Arquivos "water_consumption_updated.csv" e "Area.csv"

-Output: Equação da reta de regressão (y), coeficiente de correlação (r) e coeficiente de determinação (r^2)

Matéria teórica e fórmulas

Coeficiente de correlação (r)

$$r = \frac{S_{xy}}{\sqrt{S_{xx} \times S_{yy}}}, -1 \leq r \leq 1$$

Em que:

$$S_{xx} = \sum_{i=1}^n (x_i - \bar{x})^2, \quad S_{yy} = \sum_{i=1}^n (y_i - \bar{y})^2, \quad \text{e} \quad S_{xy} = \sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})$$

- **r > 0:** associação linear positiva entre as 2 variáveis. Quando os valores de uma das variáveis aumentam, existe tendência para que os valores da outra variável também aumentem;
- **r < 0:** associação linear negativa entre as 2 variáveis. Quando os valores de uma das variáveis aumentam, existe tendência para que os valores da outra variável diminuam.

Coeficiente de determinação (r²)

$$0 \leq r^2 \leq 1$$

- **r² = 1:** todas as observações estão sobre a reta de regressão (ajuste perfeito);
- **r² = 0:** o modelo de regressão não tem qualquer utilidade.

US16

Introdução

Na User Story 16 (US16), o objetivo principal é aplicar a regressão polinomial para determinar a melhor linha que se ajusta aos dados obtidos na US14 que consistiu em realizar testes para entradas de tamanho variável, a fim de observar o comportamento do tempo de execução do algoritmo da US13.

Os dados resultantes desses testes serão utilizados para ajustar uma linha de regressão polinomial que melhor represente o comportamento do tempo de execução em função do tamanho do input.

Resultados Esperados

A análise procura alcançar os seguintes resultados:

1. Ajuste da linha de regressão polinomial

Utilizando os dados obtidos nos testes da US14, aplicaremos a regressão polinomial para determinar a melhor linha que se ajusta aos dados. Essa linha representará o comportamento do tempo de execução do algoritmo em função do tamanho da entrada.

- Input: execution_times.csv com os dados de input e tempo de execução.
- Output: Equação da linha de regressão polinomial.

2. Visualização do ajuste da regressão polinomial

Geraremos um gráfico que represente o ajuste da regressão polinomial aos dados. Esse gráfico permitirá visualizar como a linha de regressão se adapta aos pontos de dados obtidos nos testes.

- Input: execution_times.csv com os dados de input e tempo de execução.
- Output: Gráfico que mostra o ajuste da regressão polinomial aos dados.

Matéria teórica e fórmulas

Na regressão polinomial, o objetivo é ajustar uma curva polinomial aos pontos de dados, em vez de uma linha reta como na regressão linear simples. A equação geral para um modelo de regressão polinomial é dada por:

$$y = \beta_0 + \beta_1 x + \beta_2 x^2 + \beta_3 x^3 + \dots + \beta_m x^m + \epsilon$$

Onde:

- y é a variável dependente (tempo de execução).
- x é a variável independente (tamanho do input).
- $\beta_0, \beta_1, \beta_2, \beta_3, \dots, \beta_m$ são os coeficientes do modelo.
- ϵ é o termo de erro ou resíduo, ou seja, a diferença entre o valor observado e o valor estimado pelo modelo.

MDISC

Este pdf contém a teoria estruturada e análise da complexidade do tempo do pior caso de procedimentos desenvolvidos nas respectivas User Stories. Os algoritmos de Dijkstra e Kruskal estão apresentados abaixo em pseudocódigo, com a sua complexidade analisada.

Antes de começar a analisar, temos abaixo a teoria usada para o desenvolvimento da análise feita ao detalhe.

As informações abaixo apresentam as operações principais dos algoritmos usados com as suas respectivas complexidades.

- atribuições (A);
- incrementos (I);
- comparações (C);
- retornos (R).

MÉTODO DO ALGORITMO DE KRUSKAL US13

Method calculateMinimumSpanningTree(edges, mstEdges)

edges = sort(edges by weight)

parent = new HashMap()

rank = new HashMap()

vertices = new HashSet()

for edge em edges

vertices.add(edge.from)

vertices.add(edge.to)

for edge em edges

rootX = find(parent, edge.from)

rootY = find(parent, edge.to)

if rootX \neq rootY

mstEdges.add(edge)

union(parent, rank, rootX, rootY)

if tamanho(mstEdges) = tamanho(vertices) – 1

break

Return mstEdges

PSEUDOCÓDIGO DA US13 E RESPETIVA ANÁLISE DETALHADA SOBRE SUA COMPLEXIDADE

Código	Análise
Method calculateMinimumSpanningTree(edges, mstEdges)	$O(n^2)$
edges = sort(edges by weight)	$O(n \log n)$
parent = new HashMap() rank = new HashMap() vertices = new HashSet()	nA
for edge em edges	$O(n)$
vertices.add(edge.from) vertices.add(edge.to)	$O(1)$
for edge em edges	$O(n)$
rootX = find(parent, edge.from) rootY = find(parent, edge.to)	$O(\log n)$
if rootX \neq rootY	C

mstEdges.add(edge)	A
union(parent, rank, rootX, rootY)	$O(\log n)$
if tamanho(mstEdges) = tamanho(vertices) - 1	C
break	-----
Return mstEdges	R

A complexidade de tempo $O(n \log n)$ para a operação de ordenação, como **edges = sort(edges by weight)**, decorre do uso de algoritmos de ordenação eficientes, como QuickSort, MergeSort. Agora passando a uma explicação mais detalhada:

No QuickSort:

- Pior caso: $O(n^2)$, mas com boas implementações e na prática, é $O(n \log n)$
- Média e melhor caso: $O(n \log n)$
- Funcionamento: QuickSort é um algoritmo de divisão e conquista que escolhe um “pivô” e particiona a lista em dois sub-arranjos: um com elementos menores que o pivô e outro com elementos maiores. Em seguida, ele ordena os sub-arranjos recursivamente.

No MergeSort:

- Complexidade de tempo garantida: $O(n \log)$ para o melhor, pior e casos médios.
- Funcionamento: MergeSort também é um algoritmo de divisão que divide a lista em duas metades, ordenando cada metade recursivamente e depois as combina (merge).

Esses algoritmos eficientes têm uma complexidade de tempo $O(n \log n)$ por causa das seguintes razões:

1) Divisão:

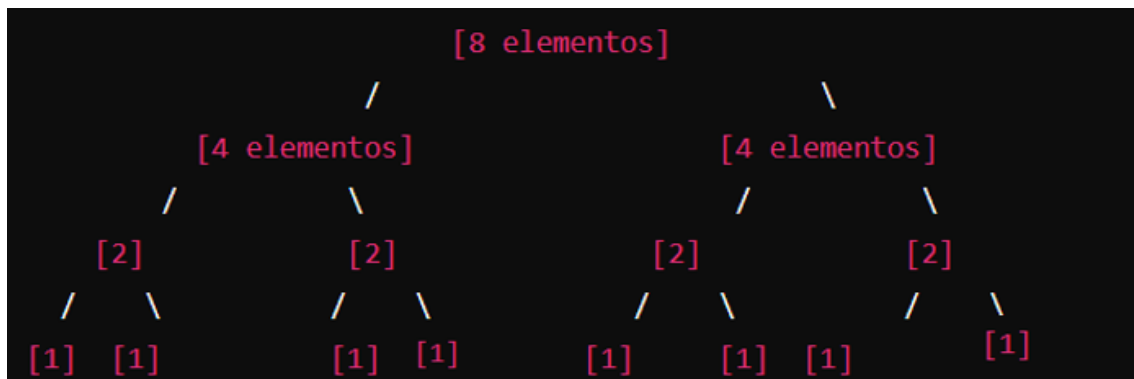
- Eles dividem a lista em partes menores ($\log n$ vezes)
- Em cada nível de divisão, o custo de combinar (merge) ou repartir a lista é $O(n)$

2) Profundidade de recursão:

- A profundidade da árvore de recursão é **$\log n$** .
- O custo de cada nível de árvore de recursão é proporcional a **$O(n)$** .

Concluindo, a complexidade total de tempo é **$O(n \log n)$** , onde **n** é o número de elementos na lista (número de arestas no caso da US).

Para dar um exemplo mais compreensível: Se tivermos 8 elementos, a árvore de recursão pode ser visualizada assim:



- Número de níveis: $\log_2(8) = 3$
- Operações por nível: $O(n)$ (combinando ou dividindo os elementos)

Multiplicando o número de níveis ($\log n$) pelas operações por nível (n), obtemos $O(n \log n)$, sendo assim o motivo para **edges = sort(edges by weight)** ter complexidade $O(n \log n)$.

Explicando agora a inicialização de **parent**, **rank**, **vertices** ser considerado $O(n)$ ou nA :

1) **parent = new HashMap():**

- **Explicação:** A inicialização de um **HashMap** em si é uma operação de tempo constante, $O(1)$. No entanto, o uso do **HashMap** será $O(n)$ ao longo do tempo, pois vamos armazenar cada nó individualmente no **HashMap** durante o processo das arestas.
- **Operação associada:** Durante o processamento, a inserção de n nós em um **HashMap** envolve n operações, cada uma sendo $O(1)$. Portanto, o uso geral do **HashMap** para armazenar os “pais” dos nós será $O(n)$.

2) **rank = new HashMap**

- **Explicação:** Similar ao **parent**, a inicialização do **HashMap** é uma operação de tempo constante, $O(1)$. No entanto, durante a execução do algoritmo, vamos armazenar e acessar informações de **rank** para cada nó, o que, acumulativamente, é $O(n)$.
- **Operação associada:** Durante o processamento, a inserção e acesso de n **ranks** de nós em um **HashMap** envolve n operações, cada uma sendo $O(1)$. Portanto, o uso geral do **HashMap** para armazenar os **ranks** dos nós será $O(n)$.

3) `vertices = new HashSet()`

- **Explicação:** A inicialização de um **HashSet** também é uma operação de tempo constante, $O(1)$. No entanto, o uso do **HashSet** será $O(n)$ ao longo do tempo, pois vamos armazenar cada nó individualmente no **HashSet** durante o processamento das arestas.
- **Operação associada:** Durante o processamento, a inserção de **n** nós em um **HashSet** envolve **n** operações, cada uma sendo $O(1)$. Portanto, o uso geral do **HashSet** para armazenar os nós será $O(n)$ (ou nA).

Concluindo, embora a criação inicial de cada estrutura (**HashMap** ou **HashSet**) seja uma operação de tempo constante, $O(1)$, o termo nA ou $O(n)$ reflete assim, o uso cumulativo dessas estruturas durante o processamento de todas as arestas no gráfico. Como vamos armazenar informações para cada nó (até **n** nós, onde **n** é o número de nós), a complexidade acumulativa para essas operações será sempre $O(n)$.

Antes de continuar gostaria de clarificar a diferença entre as notações de complexidade e a razão pela qual nA e $O(n)$ são utilizados de forma diferente no contexto de pseudocódigo.

$O(n)$: É uma notação assintótica que descreve o comportamento da complexidade de tempo ou espaço de um algoritmo à medida que no tamanho da entrada cresce. Indica que a complexidade é linear em relação ao tamanho da entrada.

nA : No contexto do pseudocódigo fornecido, nA é uma notação específica que está sendo usada para descrever operações acumulativas em termos de número de operações (A - Atribuições). Neste contexto, nA representa **n** operações de atribuição.

For edge em edges - $O(n)$:

- Aqui, estamos repetir sobre cada aresta na **lista edges**. Se há **n** arestas, essa operação de iteração ocorre **n** vezes.
- **Complexidade:** $O(n)$ onde x é o número de arestas.

`vertices.add(edge.from)` - $O(1)$:

- A operação **add** em um **HashSet** tem complexidade de tempo constante $O(1)$ no caso médio, devido ao uso de tabelas de **hash**.
- **Complexidade por operação:** $O(1)$.

`vertices.add(edge.to)` - $O(1)$:

- Semelhante à operação **add** para adicionar o nó **to** ao **HashSet** também tem complexidade $O(1)$.
- **Complexidade por operação:** $O(1)$.

```

for edge em edges → O(n)
rootX ← find(parent, edge.from) → O(log n)
rootY ← find(parent, edge.to) → O(log n)

```

O(n): Esta complexidade ocorre quando repetimos linearmente sobre uma coleção de tamanho **n**. No caso do **for edge em edge**, repetir sobre todas as arestas (edges) é uma operação O(n).

O(log n): Esta complexidade ocorre para a operação find no contexto da estrutura de dados Union-Find com compressão de caminho. Cada find pode ter que percorrer até log n nós na pior das hipóteses devido à compressão de caminho, tornando a operação eficientemente O(log n).

```

if rootX ≠ rootY
mstEdges.add(edge)

```

C: Condição

- Representa uma verificação condicional.
- Indica que uma operação está a ser feita para verificar uma condição lógica.

A: Atribuição

- Representa uma operação de atribuição ou adição a uma estrutura de dados.
- Indica que um valor está a ser atribuído ou adicionado.
- Indica que um valor está a ser atribuído ou adicionado.

If rootX /= rootY → C

Descrição: Esta linha verifica se rootX é diferente de rootY.

Motivo para C:

- **C** representa uma operação de condição ou verificação lógica.
- Aqui, a condição **rootX ≠ rootY** é avaliada para decidir se o código dentro do bloco **if** está a ser executado.
- A complexidade dessa verificação é O(1) (constante), mas em termos de pseudocódigo detalhado, é anotada como C para indicar que uma condição está a ser avaliada.

mstEdges.add(edge) → A

Descrição: Esta linha adiciona a aresta `edge` à lista `mstEdges`, que armazena as arestas da Árvore Geradora de Custo Mínimo.

Motivo para A:

- **A** representa uma operação de atribuição.
 - Aqui, **edge** está sendo adicionada à lista **mstEdges**.
 - A operação de adição a uma lista é uma operação de **atribuição** e é marcada como **A** para indicar que um valor está sendo atribuído ou adicionado a uma estrutura de dados.
 - A complexidade dessa operação é $O(1)$ (constante), mas em termos de pseudocódigo detalhado, é anotada como **A** para indicar que uma atribuição está ocorrendo.
-

union(parent, rank, rootX, rootY) → $O(\log n)$

A operação **union(parent, rank, rootX, rootY)** tem complexidade **$O(\log n)$** devido à necessidade de encontrar as raízes dos conjuntos dos nós envolvidos (o que é $O(\log n)$ por causa da compressão de caminho) e subquentemente, união dos conjuntos baseados em seus status/ranks (o que é $O(1)$).

Portanto, a complexidade total da operação **union** é dominada pela complexidade de encontrar as raízes, resultando em **$O(\log n)$** .

if tamanho(mstEdges) = tamanho(vertices) - 1 → C

Descrição: Verifica se o número de arestas na da Árvore Geradora de Custo Mínimo (`mstEdges`) é igual a $n - 1$, onde n é o número total de vértices no grafo. Isso é uma verificação para determinar se a árvore mínima já foi completamente construída.

Motivo para C:

- A complexidade dessa verificação é $O(1)$ porque envolve apenas comparação de dois valores.
- No entanto, em termos de notação detalhada, é marcada como **C** para indicar que uma condição está sendo avaliada.

Return mstEdges → R

Descrição: Esta linha vai retornar a Árvore Geradora de Custo Mínimo construída até o momento.

Motivo para R:

- A complexidade do retorno de uma estrutura de dados é **$O(1)$** porque não envolve nenhum processamento adicional, apenas a operação de retorno.
- No entanto, em termos de notação detalhada, é marcada como **R** para indicar que é uma operação de retorno.

MÉTODO DO ALGORITMO DE DIJKSTRA

```

Class DijkstraAlgorithm {
    List<Route> edges

Function findShortestPathToAnyEndPoint(origin: Point, endPoints: List<Point>, edges: List<Route>) -> List<Route> {
    // Assign the provided edges to the class variable
    this.edges = edges

    // Create a list of unique points from the edges
    List<Point> points = new List
    For each route in edges:
        addUniquePoint(points, route.getOrigin())
        addUniquePoint(points, route.getDestination())
    End For

    // Initialize arrays for costs, previous nodes, and visited nodes
    Array<double> costs = new Array of size points.size()
    Array<String> previousNodes = new Array of size points.size()
    Array<boolean> visitedNodes = new Array of size points.size()

    // Set default values for the arrays
    For i from 0 to points.size() - 1:
        costs[i] = Double.MAX_VALUE
        previousNodes[i] = null
        visitedNodes[i] = false
    End For

    // Set the cost of the origin point to 0
    Integer originIndex = getPointIndex(points, origin)
    costs[originIndex] = 0.0

    // Initialize a list to keep track of node costs
    List<NodeCost> nodeCosts = new List
    nodeCosts.add(new NodeCost(origin, 0.0))

```

```

Class DijkstraAlgorithm {
    List<Route> edges

Function findShortestPathToAnyEndPoint(origin: Point, endPoints: List<Point>, edges: List<Route>) -> List<Route> {
    // Assign the provided edges to the class variable
    this.edges = edges

    // Create a list of unique points from the edges
    List<Point> points = new List
    For each route in edges:
        addUniquePoint(points, route.getOrigin())
        addUniquePoint(points, route.getDestination())
    End For

    // Initialize arrays for costs, previous nodes, and visited nodes
    Array<double> costs = new Array of size points.size()
    Array<String> previousNodes = new Array of size points.size()
    Array<boolean> visitedNodes = new Array of size points.size()

    // Set default values for the arrays
    For i from 0 to points.size() - 1:
        costs[i] = Double.MAX_VALUE
        previousNodes[i] = null
        visitedNodes[i] = false
    End For

    // Set the cost of the origin point to 0
    Integer originIndex = getPointIndex(points, origin)
    costs[originIndex] = 0.0

    // Initialize a list to keep track of node costs
    List<NodeCost> nodeCosts = new List
    nodeCosts.add(new NodeCost(origin, 0.0))

```



```

        If costs[endIndex] < minCost:
            minCost = costs[endIndex]
            closestEndPoint = endPoint
        End If
    End For

    // Build and return the path to the closest endpoint
    Return buildPath(origin, closestEndPoint, points, previousNodes)

End Function

```

PSEUDOCÓDIGO DA US19 E RESPETIVA ANÁLISE DETALHADA SOBRE SUA COMPLEXIDADE

Linha de Código	Número de Iterações	Complexidades Acumuladas
this.edges = edges	1	$O(1)$
List points = new List	1	$O(1)$
For each route in edges:	n	$O(nC + nI)$
addUniquePoint(points, route.getOrigin())	n	$O(nA)$
addUniquePoint(points, route.getDestination())	n	$O(nA)$
Array costs = new Array of size points.size()	1	$O(1)$
Array previousNodes = new Array of size points.size()	1	$O(1)$
Array visitedNodes = new Array of size points.size()	1	$O(1)$
For i from 0 to points.size() - 1:	n	$O(nC + nI)$
costs[i] = Double.MAX_VALUE	n	$O(nA)$
previousNodes[i] = null	n	$O(nA)$
visitedNodes[i] = false	n	$O(nA)$
Integer originIndex = getPointIndex(points, origin)	1	$O(1)$

costs[originIndex] = 0.0	1	$O(1)$
List nodeCosts = new List	1	$O(1)$
nodeCosts.add(new NodeCost(origin, 0.0))	1	$O(1)$
While nodeCosts is not empty:	n	$O(nC)$
NodeCost currentNodeCost = getMinimumDistancePoint(nodeCosts)	n	$O(nA)$
Point currentPoint = currentNodeCost.point	n	$O(nA)$
Integer currentIndex = getPointIndex(points, currentPoint)	n	$O(nA)$
If visitedNodes[currentIndex] is true:	n	$O(nC)$
Continue	-	-
visitedNodes[currentIndex] = true	n	$O(nA)$
For each route in edges:	n^2	$O(n^2C + n^2I)$
If route.getOrigin() equals currentPoint:	n^2	$O(n^2C)$
Point neighbor = route.getDestination()	n^2	$O(n^2A)$
Integer neighborIndex = getPointIndex(points, neighbor)	n^2	$O(n^2A)$
Double newCost = costs[currentIndex] + route.getCost()	n^2	$O(n^2Op)$
If newCost < costs[neighborIndex]:	n^2	$O(n^2C)$
costs[neighborIndex] = newCost	n^2	$O(n^2A)$
previousNodes[neighborIndex] = currentPoint.getId()	n^2	$O(n^2A)$
nodeCosts.add(new NodeCost(neighbor, newCost))	n^2	$O(n^2A)$
For each endPoint in endPoints:	n	$O(nC + nI)$
Integer endIndex = getPointIndex(points, endPoint)	n	$O(nA)$
If costs[endIndex] < minCost:	n	$O(nC)$
minCost = costs[endIndex]	n	$O(nA)$
closestEndPoint = endPoint	n	$O(nA)$
Return buildPath(origin, closestEndPoint, points, previousNodes)	1	R

this.edges = edges: Isto é uma atribuição direta de uma variável sendo feito em tempo constante, $O(1)$, pois não depende do tamanho da entrada.

List points = new List: Neste caso, estamos apenas a inicializar uma lista o que também é feito em tempo constante, $O(1)$.

For each route in edges:: Este caso implica em percorrer em todos os elementos em "edges", que contém "n" elementos. Então, a complexidade é $O(nI + nC)$, onde nI é o custo a percorrer sobre cada elemento e nC é o custo de realizar operações em cada elemento.

addUniquePoint(points, route.getOrigin()) e addUniquePoint(points, route.getDestination()): Neste caso, as operações adicionam pontos à lista e se a lista ainda não contém o ponto, ele é adicionado. Se a lista for grande, pode haver uma busca para verificar a existência do ponto, resultando em complexidade $O(nA)$, onde nA é o custo de adicionar um ponto único.

Array costs = new Array of size points.size(): Iniciar uma matriz com o tamanho da lista de pontos é feito em tempo constante, $O(1)$.

Array previousNodes = new Array of size points.size() e Array visitedNodes = new Array of size points.size(): Ambas as operações envolvem a inicialização de arrays com base no tamanho da lista de pontos, então também são $O(1)$.

For i from 0 to points.size() - 1: Temos um Loop sobre todos os elementos da lista de pontos sendo a complexidade é $O(nI+nC)$, onde nI é o custo a percorrer sobre cada elemento e nC é o custo de realizar operações em cada elemento.

costs[i] = Double.MAX_VALUE, previousNodes[i] = null e visitedNodes[i] = false: Estas operações são executadas para cada elemento na lista de pontos, portanto, a complexidade é $O(nA)$, onde nA é o custo de atribuir um valor a cada elemento.

Integer originIndex = getPointIndex(points, origin) e costs[originIndex] = 0.0: São operações de busca em uma lista e atribuição de um valor, respectivamente. Portanto, cada uma é $O(1)$.

List nodeCosts = new List e nodeCosts.add(new NodeCost(origin, 0.0)): Novamente, essas operações são feitas em tempo constante, $O(1)$.

While nodeCosts is not empty: Um loop que continua até que "nodeCosts" esteja vazio. A complexidade é $O(nC)$, onde nC é o custo das operações realizadas em cada iteração.

NodeCost currentNodeCost = getMinimumDistancePoint(nodeCosts), Point currentPoint = currentNodeCost.point e Integer currentIndex = getPointIndex(points, currentPoint): Cada uma dessas operações é $O(nA)$, onde nA é o custo da operação.

If visitedNodes[currentIndex] is true:: Uma operação de verificação. Isso é feito em tempo constante, $O(1)$.

visitedNodes[currentIndex] = true: Atribuição de um valor a um elemento na lista "visitedNodes". Portanto, é $O(1)$.

For each route in edges: Um loop que percorre todos os elementos em "edges". A complexidade é $O(n^2C+n^2I)$, onde n^2I é o custo de iterar sobre cada elemento e n^2C é o custo de realizar operações em cada elemento.

If route.getOrigin() equals currentPoint:: Uma verificação. Isso é feito em tempo constante, $O(1)$.

Point neighbor = route.getDestination() e Integer neighborIndex = getPointIndex(points, neighbor): São operações que envolvem buscar informações em uma lista. Assim, cada uma é $O(1)$.

Double newCost = costs[currentIndex] + route.getCost(): Uma operação de adição. É $O(1)$.

If newCost < costs[neighborIndex]: Uma operação de comparação, C .

costs[neighborIndex] = newCost e previousNodes[neighborIndex] = currentPoint.getId(): Operações de atribuição A , cada uma é $O(1)$.

nodeCosts.add(new NodeCost(neighbor, newCost)): Adição de um elemento a uma lista. Isso é $O(1)$.

For each endPoint in endPoints: Temos um loop que percorre todos os elementos em "endPoints". A complexidade é $O(nC + nI)$, onde nI é o custo de percorrer sobre cada elemento e nC é o custo de realizar operações em cada elemento.

Integer endIndex = getPointIndex(points, endPoint): Uma operação de busca de uma informação em uma lista. É $O(1)$.

If costs[endIndex] < minCost: Uma operação de comparação, C . Isso é feito em tempo constante, $O(1)$.

minCost = costs[endIndex] e closestEndPoint = endPoint: Atribuição de valores. Cada uma é $O(1)$.

Return buildPath(origin, closestEndPoint, points, previousNodes): Uma operação de retorno, R . Isso é feito em tempo constante, $O(1)$.

Contacts/Support

Because this software is a proof of concept, it is still undergoing many improvements and is not yet complete. If you run into any problems or any bugs, please do not hesitate to contact us. We appreciate your feedback and remain committed to improving the software. For assistance, you can contact members of the group at the following email addresses:

1230792@isep.ipp.pt – Mariana Sousa

1231183@isep.ipp.pt – Marta Domingues

1231375@isep.ipp.pt – Nuno Teixeira