



Universidad Politécnica
de Madrid

**Escuela Técnica Superior de
Ingenieros Informáticos**



Grado en Ingeniería Informática

Trabajo Fin de Grado

MODELO de SIMULACIÓN de un TALLER de MANTENIMIENTO

Autor: Eduardo Marquina García

Tutor(a): Juan Antonio Fernández del Pozo de Salamanca

Madrid, Febrero 2024

Este Trabajo Fin de Grado se ha depositado en la ETSI Informáticos de la Universidad Politécnica de Madrid para su defensa.

Trabajo Fin de Grado
Grado en Ingeniería Informática

Título: MODELO de SIMULACIÓN de un TALLER de MANTENIMIENTO
Febrero 2024

Autor: Eduardo Marquina García
Tutor: Juan Antonio Fernández del Pozo de Salamanca
Departamento de Inteligencia Artificial
Escuela Técnica Superior de Ingenieros Informáticos
Universidad Politécnica de Madrid

Resumen

La simulación es una herramienta computacional que permite imitar el funcionamiento de un sistema real, teórico o ficticio con el fin de poder predecir el comportamiento futuro de un proceso antes de que el mismo ocurra. Esta técnica es de gran valía ya que ofrece la posibilidad de examinar sistemas que se pretenden instalar antes de su puesta en marcha efectiva, lo que nos otorga la capacidad de corregir y mejorar posibles problemas en los cálculos, procedimientos, diseño, etc. [1]

Es fundamental conocer los distintos componentes que forman el marco de una simulación; Las variables de estado son aquellos elementos que plasman las dinámicas que se producen en el sistema, sus distintos valores provocan múltiples escenarios debido a las diferentes casuísticas que se producen. Los eventos son los hechos que suceden en un momento determinado y que provocan cambios en el sistema, los cuales se registran en el estado del sistema. Además, este último nos proporciona una captura de las características y condiciones del sistema en un instante preciso.

Obtener una distribución estadística que se ajuste a las cualidades de nuestro sistema es sumamente complejo, por ello, tras la ejecución del mismo se analizan detalladamente los datos estadísticos recogidos. Esto se hace mediante la visualización de los datos a través de gráficas, diagramas, etc. Esta parte es fundamental para comprender como trabaja nuestro sistema ya que el entendimiento de los datos en crudo es mucho más costoso.

Este Trabajo de Fin de Grado se enfoca en la creación de un modelo de simulación de un taller de mantenimiento en tiempo continuo. Con ello se pretende identificar patrones de flujo de trabajo, conocer la eficiencia operacional del sistema y optimizar los tiempos de respuesta al cliente con el objetivo final de elevar la calidad del servicio.

Abstract

Simulation is a computational tool that makes it possible to imitate the operation of a real, theoretical or fictitious system in order to be able to predict the future behaviour of a process before it happens. This technique is of great value as it offers the possibility of examining systems that are intended to be installed before their effective start-up, which gives us the ability to correct and improve possible problems in the calculations, procedures, design, etc. [1]

It is essential to be aware of the different components that make up the framework of a simulation; state variables are those elements that capture the dynamics that occur in the system, their different values cause multiple scenarios due to the different casuistry that occur.

Events are the facts that happen at a given time and cause changes in the system, which are recorded in the state of the system. This provides us a snapshot of the characteristics and conditions of the system at a precise instant.

Obtaining a statistical distribution that fits the qualities of our system is extremely complex, therefore, after the execution of the same, the collected statistical data is analysed in detail. This is done by visualising the data by means of graphs, diagrams, etc. This step is crucial to understand how our system works as understanding the raw data is much more costly.

This Final Degree Project focuses on the creation of a simulation model of a maintenance workshop based on continuous time. The aim is to identify workflow patterns, understand the operational efficiency of the system and optimise customer response times with the ultimate goal of improving the quality of service.

Tabla de contenidos

1. Introducción	1
1.1. Objetivos	2
1.2. Técnicas de Modelado y Simulación	3
1.2.1. Según su Método de Procesamiento	3
1.2.2. Según sus Atributos Fundamentales	4
1.3. Entorno de trabajo	6
1.3.1. Librerías y paquetes usados	7
2. Estado del Arte	9
2.1. La simulación de sistemas	9
2.1.1. Modelado y Simulación de Sistemas	9
2.1.2. Simulación de Sistemas Discretos	10
2.1.3. Simulación. Métodos y Aplicaciones	11
2.2. Simulación de Procesos	12
2.2.1. Simulación de Procesos Industriales	12
2.3. Distribuciones de Probabilidad	13
2.3.1. Distribución de Poisson	13
2.3.2. Distribución Exponencial	14
2.3.3. Distribución Binomial	15
2.3.4. Distribución Uniforme	16
3. Modelo Empirico	19
3.1. Visión General del Sistema	19
3.2. Árbol de Decisión	20
3.3. Escenarios	22
3.3.1. Prototipo 1	22
3.3.2. Prototipo 2	23
4. Prototipado	25
4.1. Diseño Prototipo 1	25
4.1.1. Evaluación de Necesidades del Vehículo	25
4.1.2. Flujo de Trabajo Mantenimiento	26
4.1.3. Flujo de Trabajo Chapa y Pintura	28
4.1.4. Flujo de Trabajo Lavado	29
4.2. Diseño Prototipo 2	30
4.2.1. Flujo de Trabajo ITV	30

TABLA DE CONTENIDOS

4.2.2. Ampliación Flujo de Trabajo Lavado	31
4.3. Implementación	31
4.3.1. Funciones probabilísticas	31
4.3.2. Funciones de las trayectorias	33
4.4. Simulación	34
4.4.1. Fichero de Simulación - simmer.R	34
4.4.2. Fichero de Interfaz Gráfica - app.R	38
5. Análisis de Datos	41
5.1. Características de la Simulación	41
5.1.1. Parámetros del Sistema	42
5.2. Datos Estadísticos	44
5.3. Visualización de Gráficas	48
5.3.1. Distribución en el Tiempo del Uso de Recursos	49
5.3.2. Distribución de Llegadas	50
5.3.3. Densidad Tiempo de Servicio	50
5.3.4. Análisis de Gastos	51
5.3.5. Uso de Recursos	52
5.3.6. Tiempo de Actividad	54
5.3.7. Tiempo de Espera	55
6. Conclusiones y trabajo futuro	57
6.1. Evaluación de Objetivos del TFG	57
6.2. Lineas Futuras	58
6.3. Validación y Verificación	59
6.4. Evaluación del Proceso de Realización del TFG	60
Bibliografía	63
Anexos	67
A. Primer anexo	67
B. Segundo anexo	83

Índice de figuras

1.1. Integrador[3]	3
1.2. Pasos Simulación de Monte Carlo[6]	5
1.3. Caras dado	6
1.4. Logo RStudio	7
2.1. Principles of Modeling and Simulation [9]	10
2.2. Simulación de Sistemas Discretos [10]	11
2.3. Principales Símbolos de un Diagrama de Flujo [15]	12
2.4. Distribución Uniforme [20]	17
3.1. Visión General del Sistema Prototipo 1	19
3.2. Árbol de Decisión Prototipo 1	21
3.3. Prototipo 1 Escenario 1 - Valoración en la <i>Inspección</i> negativa	22
3.4. Prototipo 1 Escenario 2 - Cliente atraviesa varias trayectorias	23
3.5. Prototipo 2 - Visión General del Sistema Prototipo	23
3.6. Diagrama de Flujo <i>Prototipo 2 Escenario 1 - ITV con Fallo Grave</i>	24
4.1. Diagrama de Flujo <i>Evaluación</i>	26
4.2. Ilustración <i>Mantenimiento</i>	26
4.3. Diagrama de flujo <i>Espera de Piezas</i>	27
4.4. Diagrama de Flujo <i>Reparación</i>	27
4.5. Diagrama de Flujo <i>Inspección</i>	28
4.6. Diagrama de Flujo <i>Chapa y Pintura</i>	28
4.7. Diagrama de Flujo <i>Pulido</i>	29
4.8. Diagrama de Flujo <i>Lavado</i>	29
4.9. Diagrama de Flujo <i>Revisión ITV</i>	30
4.10 Diagrama de Flujo <i>Lavado</i>	31
4.11.Código para la generación de clientes y tiempo de actividad	32
4.12.Código para la toma de decisiones	33
4.13Nube de Contenido Entidades	33
4.14Ficheros generados tras la ejecución del código	35
4.15Ejemplo Fichero de Trazas	37
4.16Portada Interfaz Gráfica de Shiny	38
4.17Panel <i>Seleccionar Simulación</i>	39
4.18.Opciones panel <i>Seleccionar tipo de Gráfico Específico</i>	39
5.1. Resultados Estadísticos bajo Condiciones Iniciales	45

ÍNDICE DE FIGURAS

5.2. Resumen Estadístico Cola y Coste Cond. Inic. 1	46
5.3. Resultados Estadísticos bajo Condiciones Iniciales 2	47
5.4. Resumen Estadístico Cola y Coste Cond. Inic. 2	48
5.5. Gráfico Circular Prototipo 1	49
5.6. Histograma de Llegadas	50
5.7. Densidad Tiempo de Servicio	51
5.8. Prototipo 1: Distribución de Gastos según el tipo de cliente	51
5.9. Porcentaje de uso de cada recurso	53
5.10.Tiempo de Actividad por encima de la Mediana	54
5.11.Tiempo de Actividad por debajo de la Mediana	55
5.12.Tiempo de Espera por encima de la Mediana	55
5.13.Tiempo de Espera por debajo de la Mediana	56
B.1. Informe de Originalidad Turnitin	84

Capítulo 1

Introducción

Este Trabajo de Fin de Grado (TFG) se centra en el estudio y aplicación de la simulación de un sistema en tiempo continuo dentro de un taller de mantenimiento de vehículos. Esta aproximación metodológica busca no solo ofrecer una visión de los procesos internos que definen la operativa diaria del taller, sino también identificar estrategias para mejorar la calidad del servicio y la satisfacción del cliente.

La simulación computacional en tiempo continuo está compuesta por entidades y eventos; Las **entidades** nos aportan información sobre el estado del sistema y poseen atributos que rigen su comportamiento; Los **eventos** por otra parte son sucesos que provocan los cambios en las variables de estado[2]. El conjunto de estas variables conforma el estado del sistema, el cual nos indica la situación del sistema en un instante determinado.

Además, en este tipo de simulaciones de tiempo continuo los eventos se registran en cualquier momento dentro de una línea temporal constante. A diferencia de la simulación de eventos discretos, donde los eventos se registran en ciclos de tiempo de amplitud fija, en la simulación en tiempo continuo los eventos pueden ocurrir en cualquier punto del tiempo, lo que permite una mayor precisión en la modelación de procesos y poder obtener unos resultados más exactos del sistema.

Al replicar de manera virtual el flujo de trabajo y las interacciones dentro del sistema, este estudio nos proporciona conocimientos valiosos sobre el rendimiento y funcionamiento del mismo y destaca oportunidades de mejora que, de otra manera, podrían permanecer ocultas.

La importancia de adoptar tales herramientas de simulación radica en su capacidad para predecir el comportamiento del sistema ante variadas condiciones, permitiendo así la implementación de ajustes y optimizaciones antes de su aplicación en el entorno real. Este enfoque pro activo no solo contribuye a la reducción de costos y tiempos de inactividad, sino que también mejora la capacidad de respuesta frente a las demandas de los clientes y eleva la eficacia general del servicio.

En este TFG y a través de la implementación de distintos prototipos se busca realizar un análisis detallado de nuestro modelo de simulación de un taller de reparación, así como extraer conclusiones objetivas mediante los datos recogidos sobre el desempeño actual del sistema con el fin de proponer mejoras específicas dirigidas a optimizar los procesos del taller.

1.1. Objetivos

Este trabajo tiene como **objetivo principal** desarrollar un modelo de simulación lo más completo posible mediante la realización de prototipos de complejidad creciente con el objetivo final de que la simulación sea lo más realista posible, asegurando así que abarque un amplio espectro de variables y escenarios potenciales. Con ello, se espera no solo comprender mejor el fenómeno estudiado, sino también proveer una herramienta para la experimentación y análisis en un entorno controlado capaz de adaptarse a las necesidades cambiantes del estudio.

A su vez, los **objetivos específicos** que se van a abordar en este Trabajo Fin de Grado incluyen:

- **Elaboración Modelo Empírico:** Este objetivo se basa en investigar y documentar las principales características del modelo de simulación a realizar, así como de la representación del mismo mediante figuras e imágenes descriptivas del sistema en sus distintas fases y casuísticas. Esta parte es fundamental para explicar qué funciones va a tener el modelo y bajo qué condiciones va a trabajar en cada una de esas funciones.
- **Desarrollo Modelo Matemático y Computacional:** En esta sección se pretende dar forma a todo lo descrito en el apartado anterior mediante la traducción del Modelo Empírico a un conjunto de ecuaciones matemáticas que describan las interacciones y funcionamiento del modelo. Esta parte busca elaborar un código en un entorno de simulación computacional que use este conjunto de ecuaciones o distribuciones matemáticas.
- **Implementación de la Simulación:** Este objetivo se basa en la correcta definición de los escenarios a simular que reflejen las distintas condiciones en las que puede suceder un fenómeno o casuística. Más tarde se ejecuta en repetidas veces y mediante ciclos de ejecución la simulación que se ha codificado y se recogen de los datos que nos aporta la misma.
- **Análisis de los datos y Visualización:** Una vez recogidos los datos se procederá al análisis de los mismos. Esto nos permitirá sacar conclusiones del modelo así como modificar y adaptar los procesos para que sean más eficientes. La parte de visualización posterior es fundamental para poder traducir el volumen de datos en una representación de la información más clara y comprensible.

1.2. Técnicas de Modelado y Simulación

Este campo ha evolucionado significativamente a lo largo de los años, introduciendo una variedad de herramientas avanzadas que nos permiten reproducir y estudiar comportamientos y procesos complejos en entornos virtuales controlados. Desde el modelado matemático hasta las simulaciones computacionales y el uso de inteligencia artificial para optimizar estos procesos, cada técnica ofrece perspectivas y soluciones únicas a problemas concretos, lo que permite una comprensión y análisis más profundo del fenómeno que se está estudiando.

Las simulaciones se pueden clasificar según distintos factores:

1.2.1. Según su Método de Procesamiento

- **Simulación Analógica:** Los primeros simuladores que existieron fueron analógicos, estos se utilizan para resolver sistemas de ecuaciones diferenciales ordinarias (no algebraicas) beneficiándose del hecho de que las ecuaciones que describen ciertos dispositivos electrónicos son idénticas a las que describen otros sistemas de interés. Conectando los componentes electrónicos (tales como condensadores, amplificadores, potenciómetros y resistencias) de diversas maneras, es posible realizar operaciones matemáticas complejas usando las respuestas físicas de los circuitos. Entre ellas, la Suma o la Integración, mediante el uso de esta última se consiguen resolver ecuaciones diferenciales y por tanto simular comportamientos específicos del sistema a modelar.

En la figura 1.1 se ilustra la creación de un circuito integrador. Las resistencias R_1 , R_2 , y R_3 están configuradas para condicionar la señal de entrada. El condensador C se encarga de acumular la carga eléctrica, y junto con el amplificador operacional, se establece la integración de la señal.

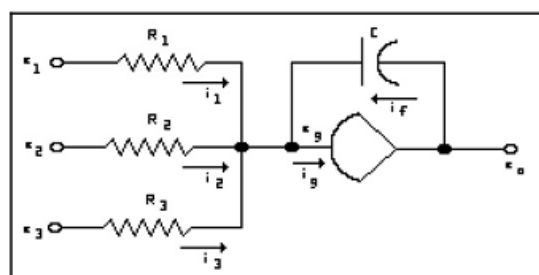


Figura 8 Bloque integrador

Figura 1.1: Integrador[3]

- **Simulación Numérica:** Esta técnica usada desde los años 60 y que tuvo su gran auge con la llegada de las computadoras digitales, sirve para validar modelos mediante pequeñas variaciones o derivaciones sobre las fórmulas de las ecuaciones diferenciales. Estos métodos entre los que se encuentra el método de Euler entre muchos otros, proveen aproximaciones para resolver

las ecuaciones, es decir, mediante el uso de modelos matemáticos se consigue describir el comportamiento de los sistemas.

La ingeniería geotécnica es una rama de la ingeniería que estudia las propiedades y características del suelo y las rocas y es una de las grandes beneficiadas en el uso de este tipo de simulación, ya que mediante la modelización numérica se consigue realizar un análisis de las infraestructuras tales como túneles, puentes, presas, diques, etc. optimizando el proceso de construcción de las mismas. Otro ejemplo es el campo económico, el modelo matemático se basará en principios como la inflación, compra-venta y otros factores económicos clave mediante los cuales poder prever el impacto de políticas económicas y/o anticiparse a la toma de decisiones, por eso tiene una parte importante de su aplicación en el análisis de riesgo financiero.

Además, las simulaciones numéricas actualmente evolucionan hacia la integración en los modelos de técnicas de *machine learning* e *inteligencia artificial* que otorgan nuevas posibilidades de estudio y conducen a predicciones más precisas.

- **Simulación Analógica Digital:** Con la llegada de las computadoras, se abrieron nuevas vías de investigación en el campo del modelado y la simulación, así es como, a partir del año 1955 se comienzan a desarrollar los primeros programas que integran componentes analógicos y digitales, aunque el impulso decisivo en este campo se alcanzó años más tarde cuando los PC's poseían una mayor capacidad de cómputo y por tanto un mayor rendimiento. Destacan la aparición de programas como *SIMULINK* [4] herramienta que se integra en *MATLAB* [5] y la cual permite construir modelos virtuales en una alta gama de disciplinas, para simular y probar sistemas, también concede la posibilidad de generar código en distintos lenguajes de programación, lo cual ofrece una gran flexibilidad

1.2.2. Según sus Atributos Fundamentales

- **Determinista o Estocástico:**
 - *Determinista:* Este modelo se basa en datos puramente deterministas, es decir, establecidos y fijos, lo que significa que no existe ninguna variabilidad de aleatoriedad en los mismos, los datos de entrada son determinados como la propia palabra indica, es decir, no son azarosos.
 - *Estocástico:* En el modelo estocástico por el contrario los valores de las variables se rigen por el azar, determinados normalmente por distribuciones estadísticas, por tanto, en cada iteración que se produzca sobre la misma simulación se generarán unos datos distintos de entrada. Entre las distribuciones más extendidas y también las más empleadas en términos de simulación tendríamos la distribución Normal o Gaussiana, la Uniforme, la Exponencial, la de Poisson... Cada una posee distintas características y por tanto, cada una describe mejor un comportamiento que otras. Sin embargo, el número de distribuciones existentes es infinito, ya que para

1.2. Técnicas de Modelado y Simulación

cada situación única se puede teóricamente construir una distribución de probabilidad única que la describa.

- *Simulaciones de Monte Carlo*: Dentro de los modelos estocásticos cabe remarcar el "Método de Monte Carlo", el cual lo forman un conjunto de métodos creados para el desarrollo de armas nucleares en los años 40. Este método se basa en la resolución de un problema mediante juegos de azar ya que la probabilidad que subyace en estos puede determinar el comportamiento de algún suceso real que también se rija por dicha probabilidad. Además, la simulación de Monte Carlo toma su nombre del emblemático Casino de Montecarlo en Mónaco, donde la ruleta, un clásico generador de números aleatorios, es una metáfora para la simulación, que replica la naturaleza aleatoria de sucesos reales mediante el uso de modelos probabilísticos.

Este método requiere que se siga una metodología clave para su correcto desarrollo, primero se deben identificar y definir las variables de estado, estas variables aleatorias determinarán el proceder de nuestra simulación. Después se debe asignar una distribución probabilística a estas variables, esta distribución determina todos los posibles sucesos y la probabilidad de que cada uno de ellos suceda. La elección de la distribución depende del modelo a simular, así como de los objetivos y las conclusiones que se pretenden alcanzar. Finalmente, se deben generar las muestras aleatorias para cada variable en base a las distribuciones asignadas y realizar la simulación del modelo.[6]

En esta figura (1.2) podemos ver la metodología mencionada de forma ilustrada.

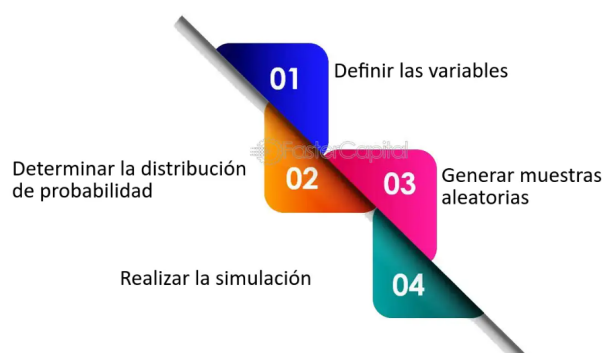


Figura 1.2: Pasos Simulación de Monte Carlo[6]

■ Estático o Dinámico:

- *Estático*: Este modelo representa los resultados mediante las relaciones del sistema cuándo está quieto o en equilibrio, es decir, las condiciones son fijas. Este tipo de simulación se suele emplear en situaciones en las que el tiempo no es una variable influyente. Un ejemplo que nos puede ayu-

dar a entender este concepto, es un dado, figura en forma cubo, en la que independientemente del tiempo y los factores externos, la respuesta al lanzarlo siempre será una de las caras del mismo. En esta imagen lo vemos representado (1.3).

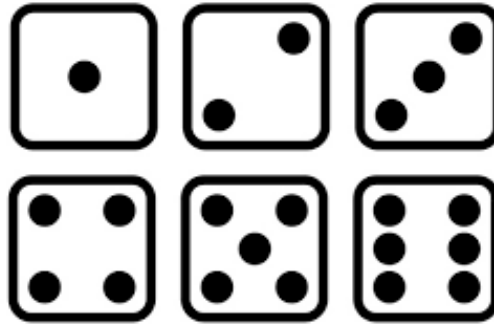


Figura 1.3: Caras dado

- **Dinámico:** Esta por el contrario es aquella simulación en la que el estado del sistema cambia en el tiempo y por tanto, se estudia la respuesta y evolución en su estado relativa a las condiciones del sistema en ese instante preciso.
- **Continuo o Discreto:**
 - **Continuo:** El modelado en tiempo continuo implica cambios en el estado del sistema constantemente en el tiempo, y se usa regularmente para simulaciones naturales como el clima, tráfico de vehículos, etc. Las simulaciones en este contexto son más fluidas y tienden a menos cambios abruptos.
 - **Discreto:** A diferencia de la anterior, en las simulaciones de eventos discretos (DES) las variables de estado cambian únicamente en determinados momentos, regidas por qué un evento suceda. Este tipo de simulación es útil en sistemas que usan colas de espera, sistemas de inventario, o en los que los eventos suceden en tiempos aleatorios.

1.3. Entorno de trabajo

En el contexto de este proyecto es importante utilizar una herramienta con la que no solo sea fácil la manipulación de los datos en sí, sino también su análisis y su visualización posterior. Por tanto, el lenguaje de programación que mejor considero se ajusta a nuestras necesidades, es R.

R surge a través del lenguaje de programación S, el cual se originó en los Laboratorios Bell, los mismos en los que se había creado casi 30 años atrás el sistema operativo Unix. Sin embargo, S era propiedad de dicho laboratorio y Ross Ihaka y Robert Gentleman, de la Universidad de Auckland en Nueva Zelanda, deciden crear una versión abierta y gratuita del mismo, lo cual da origen a R en el año

1992, aunque no sería hasta los 2000 que conseguirían obtener una versión final estable. [7]

Existen muchas razones por las que elegir R. Se fundamenta en el análisis estadístico; nos ofrece la posibilidad de usarse en los sistemas operativos más extendidos (Microsoft, Linux y/o Mac); Tiene grandes cualidades para la importación, procesamiento y exportación de datos; Posee una gran capacidad de visualización avanzada;

Asimismo se considera que el entorno de desarrollo integrado más cómodo para usar R es RStudio (1.4) debido a su cómoda interfaz que consta de distintas ventanas como la de comandos, terminal, workspace, espacio para salidas y gráficas, etc. Además de ser un entorno previamente conocido por su utilización en la asignatura de "Probabilidad y Estadística I y II" durante el grado de Ingeniería Informática.



Figura 1.4: Logo RStudio

1.3.1. Librerías y paquetes usados

Además de todas las cualidades mencionadas anteriormente R también nos da acceso a una amplia gama de librerías y paquetes que nos ofrecen todo tipo de posibilidades variadas según el objetivo que se busque en la implementación.

- **Paquetes de simulación:** Estos paquetes nos permiten crear modelos detallados de procesos y operaciones. En mi caso he escogido el paquete *simmer* [8], el cual ofrece una gran cantidad de posibilidades para generar los distintos recursos y atributos que componen el modelo, así como para ir modificándolos según avance la simulación.

El paquete *simmer* [8] lleva la simulación de eventos discretos a R. Está diseñado como un marco genérico pero potente orientado a procesos. La arquitectura encierra un núcleo de simulación escrito en C++ con capacidades de monitorización automática. Proporciona una API (interfaz de programación de aplicaciones) de R rica y flexible que gira en torno al concepto de trayectoria, un camino común en el modelo de simulación para entidades del mismo tipo.

- **Paquetes de análisis de datos:** La recogida y el manejo de los datos es crucial para más tarde poder obtener conclusiones a través de ellos. Por eso, en este caso me parece adecuado el uso de *dplyr*, un paquete que permite hacer uso de sus funciones para resumir, agrupar y transformar datos. Lo que nos permite tras la ejecución obtener métricas estadísticas avanzadas a través de las cuales se calculan el tiempo medio de un cliente en el

sistema, la tasa de paso, distribuciones y demás información relevante.

- **Paquetes de visualización:** La visualización de los datos se convierte también en un factor igual o más importantes que los mencionados anteriormente, ya que nos ayudan a comprender y proyectar todos los datos obtenidos en crudo. Para ello he escogido el paquete *simmer.plot*, la cual se encuentra dentro de *simmer*, pero es muy eficiente para visualizar todo lo sucedido en la simulación, lo cual con *plot* seria mucho más laborioso. También uso el paquete *plotly* el cual es muy útil para realizar visualizaciones más dinámicas y atractivas.
- **Interfaz gráfica:** Para la interfaz gráfica he escogido un paquete ya integrado en RStudio llamado *shiny*. Este, nos ofrece una plataforma flexible para construir aplicaciones web interactivas sin necesidad de tener conocimientos expresos en diseño web, facilitando así la exploración y comprensión de los datos.

Capítulo 2

Estado del Arte

Este apartado está destinado a realizar un repaso de distintos conocimientos relacionados con nuestro campo de estudio. Se pretende dar al lector una visión detallada de los fundamentos de la modelización y la simulación, a través de las investigaciones, estudios y avances recogidos dentro de nuestro área de indagación que nos permitan dar contexto al proyecto.

2.1. La simulación de sistemas

2.1.1. Modelado y Simulación de Sistemas

La obra "Principles of Modeling and Simulation" [9] se convierte en una pieza clave para tener una primera introducción a la teoría, las aplicaciones del modelado y la simulación. También, resulta de gran utilidad ya que trata los cuatro preceptos en los que se basa nuestro trabajo: modelado, simulación, visualización y análisis.

El Capítulo 4, titulado "Queue Modeling and Simulation", profundiza en los modelos de colas y su simulación lo que nos ofrece diferentes perspectivas sobre métodos y técnicas específicas para manejar la modelización de colas, lo cual es vital en entornos donde los vehículos deben ser atendidos secuencialmente, replicando una situación típica de un taller donde los vehículos esperan para ser reparados o revisados.

Otro de los capítulos más relevantes de este libro es el Capítulo 8: "Modeling and Simulation: Real-World Examples", el mismo se centra en el modelado de simulaciones de tráfico y nos permite analizar aspectos como los retrasos y congestiones que podrían enfrentar los conductores en una red vial, ofreciendo un amplio nivel de detalle que se convierte en crítico para nuestro modelo de simulación en aspectos como los flujos de trabajo y como tratarlos, la planificación de la capacidad de los procesos, así como distintos enfoques para el mejor manejo de emergencias y escenarios imprevistos.

A continuación, en la figura 2.1 podemos ver la portada del libro comentado.

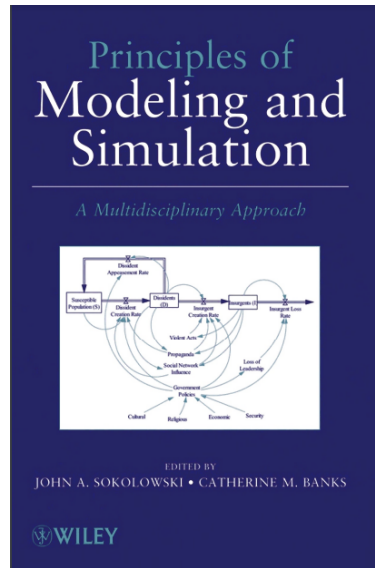


Figura 2.1: Principles of Modeling and Simulation [9]

2.1.2. Simulación de Sistemas Discretos

El libro “Simulación de Sistemas Discretos” publicado en 1996 [10] profundiza en la metodología de construcción de modelos de simulación, enfocándose en la simulación de sistemas discretos, lo cuál hace de él una fuente muy consistente de información. El mismo es obra de Jaime Barceló [11], quien destaca no solo por su contribución a la literatura académica en ingeniería de sistemas y simulación sino también por su reconocida trayectoria como académico e investigador. Barceló es catedrático en el Departamento de Estadística e Investigación Operativa de la Universidad Politécnica de Cataluña, una institución que ha sido cuna de numerosos avances en el campo de la ingeniería y la investigación operativa en España y en el ámbito internacional.

Este libro no solo ofrece una guía para entender los principios subyacentes de la simulación de sistemas discretos sino que también explora en profundidad la construcción y validación de modelos de simulación. Ofrece un repaso de los componentes aleatorios que influyen en los sistemas discretos y cómo deben ser modelados adecuadamente y habla de la aleatoriedad inherente a todas las variables de la simulación.

También es muy enriquecedor el capítulo “Modelización de la Aleatoriedad de los Sistemas”, sección en la que habla de los distintos patrones de aleatoriedad que pueden regir nuestro sistema dependiendo de la distribución estadística escogida. La cuál recalca es de suma importancia escoger adecuadamente según el proceso que se busque describir.

En la imagen siguiente (2.2) podemos apreciar la portada del libro comentado.

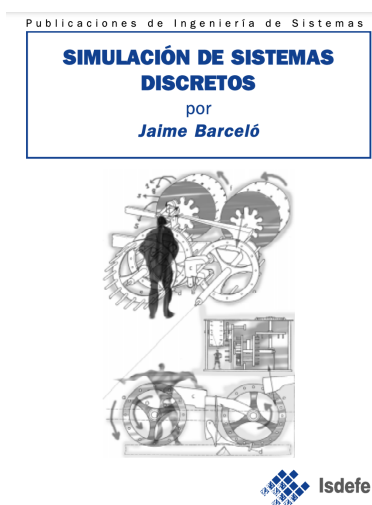


Figura 2.2: Simulación de Sistemas Discretos [10]

2.1.3. Simulación. Métodos y Aplicaciones

El libro "Simulación. Métodos y aplicaciones" (2ª Edición) [12] es una obra que aborda la creación de modelos informáticos para simular comportamientos de sistemas complejos. Su contenido se enfoca en métodos y aplicaciones de la simulación, introduciendo desde la generación de números aleatorios hasta las "Cadenas de Markov" y el "Muestreador de Gibbs".

El "**Muestreador de Gibbs**" es un algoritmo perteneciente al grupo MCMC (*Markov Chain Montecarlo*), grupo de algoritmos relacionados con el muestreo de distribuciones de probabilidades. En concreto este es un algoritmo iterativo que obtiene muestras de la distribución posterior mediante la combinación de varias distribuciones condicionales completas, esto es de suma utilidad en casos en los que la distribución conjunta es inaccesible o muy difícil de muestrear directamente y se usa con gran frecuencia en la estadística bayesiana ya que facilita la generación de muestras de distribuciones de alta dimensión sin necesidad de cálculos complicados de normalización [13]. Además, este algoritmo ha permitido calcular grandes modelos jerárquicos que requieren integraciones sobre cientos o miles de parámetros desconocidos.

Finalmente el libro examina aplicaciones prácticas como la simulación de sucesos discretos y la optimización global basada en simulación mediante ejemplos e incluyendo técnicas avanzadas como el "recocido simulado" y algoritmos genéticos. Además, abarca un conjunto de aplicaciones variadas y presenta métodos para analizar resultados de simulaciones, técnicas de reducción de la varianza y la planificación de experimentos.

2.2. Simulación de Procesos

La simulación de procesos es una técnica ampliamente usada en el sector empresarial y en el de la investigación para probar modelos matemáticos sobre procesos químicos, industriales y/o operaciones de fabricación de alta complejidad, entre otras muchas áreas. Para desarrollar este tipo de simulación se requiere tener un alto conocimiento del campo a simular, así como de las propiedades de los componentes que combinados y nos permiten realizar cálculos en el ordenador sobre los procesos involucrados [14].

Estos procesos se describen mediante diagramas de flujo o flujogramas. Este tipo de diagrama se usa para explicar el flujo de un proceso mediante símbolos y elementos estandarizados describiendo los distintos pasos y decisiones del proceso. Existen múltiples variantes de este diagrama dependiendo del sistema a describir, sin embargo, hay unos elementos que suelen ser fijos en estos, como el Proceso, Decisión, las Flechas de Flujo o las Bases de Datos [15].

En la imagen 2.3 que se muestra a continuación podemos ver algunos de los símbolos más estandarizados.

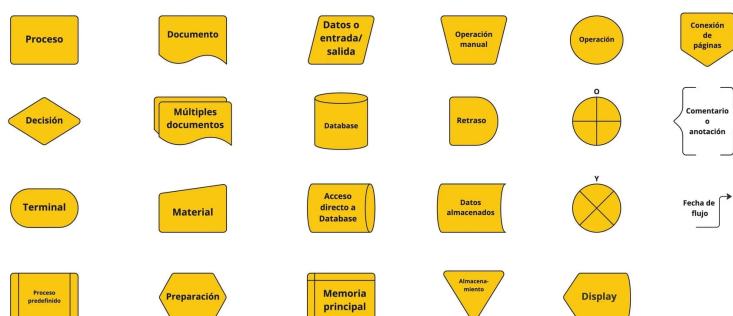


Figura 2.3: Principales Símbolos de un Diagrama de Flujo [15]

2.2.1. Simulación de Procesos Industriales

En concreto la simulación de procesos industriales busca simular procesos relacionados con la logística, la automatización de procesos, y la optimización de los procesos de fabricación como los sistemas de producción o las líneas de montaje. Estas características se asemejan a las de nuestra simulación.

En estas simulaciones se pretende averiguar por ejemplo cómo afectan y que impacto final tienen los distintos elementos que se pretenden añadir en un sistema o analizar una cierta situación empresarial para tomar decisiones estratégicas a largo plazo. Esto permite que se tenga una referencia y un apoyo sobre el que se sustentan posibles decisiones a tomar en base a los datos recogidos por la simulación [16].

En concreto vamos a hablar de las distintas **aplicaciones** que pueden tener las simulaciones en el contexto de Procesos Industriales [17]:

- **Optimización de máquinas en operación:** Esto permite la mejora conti-

nua de los equipos industriales ante aspectos como el ahorro energético, la capacidad y el rendimiento operativo, así como el mantenimiento preventivo de la máquinas usadas.

- **Mediciones de Rendimiento en Complejos Industriales:** Mediante el uso de simuladores que emplean interfaces gráficas en 2D o 3D, las empresas representan sus complejos industriales para mejorar y optimizar sus operaciones. De este modo se consiguen identificar áreas ineficientes, posibles fallos del sistema, situaciones anómalas en la demanda, incluso situaciones climatológicas adversas, como un incendio o una inundación. La revisión de todos estos casos busca que las industrias sean mas seguras, eficientes y duraderas.
- **Optimización de la Capacidad de Producción:** Este paso pretende la revisión de la maquinaria de trabajo para adaptar su capacidad productiva. Esto permite identificar ineficiencias en la producción, o la búsqueda de cambios en la cadena de producción como la adición de sistemas de espera o reposo a determinadas horas del día o ante un factor determinado.

2.3. Distribuciones de Probabilidad

2.3.1. Distribución de Poisson

La distribución de Poisson es una distribución de probabilidad que describe el número de eventos que ocurren durante un espacio de tiempo determinado, asumiendo que estos eventos ocurren con una tasa constante y de manera independiente entre sí. Esta distribución es especialmente útil para modelar el número de ocurrencias de eventos que son infrecuentes o "raros" [18].

Considérese $\lambda > 0$ y X una variable aleatoria discreta, si la variable aleatoria tiene una distribución de Poisson con parámetro λ esta se denotara por:

$$X \sim \text{Poisson}(\lambda)$$

Propiedades:

- **Media y varianza:** Para una variable aleatoria X que sigue una distribución de Poisson, la media y la varianza son ambas igual a λ .
- **Relación con la distribución exponencial:** Si el tiempo entre eventos sucesivos sigue una distribución exponencial con parámetro λ , entonces el número de eventos en un intervalo fijo sigue una distribución de Poisson con el mismo parámetro.
- **Función de probabilidad:** La probabilidad de observar exactamente k eventos viene dada por la fórmula:

$$P(X = k) = \frac{\lambda^k e^{-\lambda}}{k!}$$

donde k es el número de ocurrencias del evento, λ es el número promedio de ocurrencias en un intervalo dado, y e es la base del logaritmo natural.

- **Aproximación a la Distribución Normal:** Para valores de λ mayores que 5, la distribución de Poisson puede aproximarse a una distribución normal con media λ y desviación estándar $\sqrt{\lambda}$. Esta propiedad es útil en el mundo de la simulación ya que permite simplificar los cálculos utilizando la distribución normal, que es más fácil de manejar en términos computacionales.

$$P(X \approx N(\lambda, \sqrt{\lambda}))$$

- **Suma de Variables Poisson Independientes:** Una propiedad importante de la distribución de Poisson es que la suma de variables Poisson independientes también sigue una distribución Poisson. Si X_i son variables independientes con distribución $P(\lambda_i)$, entonces la suma de estas variables $\sum X_i$ sigue una distribución $P(\sum \lambda_i)$.

$$\sum X_i \sim P\left(\sum \lambda_i\right)$$

Esta propiedad permite agregar eventos independientes en la simulación y ajustar los resultados combinados a una única distribución Poisson, simplificando el análisis y la implementación.

2.3.2. Distribución Exponencial

La distribución exponencial está estrechamente relacionada con la Distribución de Poisson ya que esta distribución de probabilidad sirve para modelar el tiempo de espera entre dos hechos que sigan un proceso de Poisson. Es una distribución continua ideal para describir la cantidad de tiempo hasta que ocurre el próximo evento, por ejemplo en la vida real, el tiempo desde ahora mismo hasta que sucede un terremoto lo describe la Distribución Exponencial.

Sea una variable aleatoria continua X que tiene una distribución exponencial con parámetro λ , escribimos

$$X \sim \text{Exp}(\lambda), \quad \lambda > 0$$

si su función de densidad es

$$f_X(x) = \lambda e^{-\lambda x}, \quad x \geq 0$$

Propiedades:

- **Pérdida de memoria:** Un proceso exponencial tiene la propiedad de falta de memoria, lo que significa que la probabilidad de que ocurra un cierto evento en el futuro es independiente del tiempo que ya ha transcurrido.
- **Suma de Variables Exponenciales Independientes:** Si X_i son n variables independientes e idénticamente distribuidas con una distribución exponencial $\text{Exp}(\lambda)$, entonces la suma de estas variables sigue una distribución

Gamma con parámetros λ y n . Cuando n es un entero positivo, esta distribución Gamma se llama distribución Erlang:

$$\sum_{i=1}^n X_i \sim \text{Gamma}(\lambda, n) \quad \text{o} \quad \sum_{i=1}^n X_i \sim \text{Erlang}(\lambda, n)$$

Esta propiedad es útil en la simulación para agregar tiempos y modelar procesos que suman tiempos de eventos exponenciales.

- **Distribuciones con Memoria:** Si nuestro proceso tiene memoria, es decir, factores como desgaste, fatiga u obsolescencia que afectan al proceso, es más apropiado utilizar otras distribuciones. Algunas de estas distribuciones son:
 - **Distribución Gamma:** Adecuada para modelar tiempos de vida de procesos con factores acumulativos.
 - **Distribución Weibull:** Utilizada para modelar tiempos de vida de productos y materiales que se deterioran con el tiempo.
 - **Distribución Rayleigh:** Comúnmente utilizada en el análisis de datos de vida y en estudios de confiabilidad.

2.3.3. Distribución Binomial

La distribución binomial o binómica describe el número de éxitos en una secuencia de n ensayos independientes, con una probabilidad fija de p probabilidades de éxito en dichos ensayos. Realmente aunque la probabilidad de éxito pueda ser una u otra sólo existen dos posibles casos en esta distribución, el éxito que tiene una probabilidad de aparición de p y el fracaso, que tiene una probabilidad de $q = 1 - p$. Su notación es la siguiente:

$$Y \sim \text{Binomial}(n, p), \quad n > 0, \quad 0 \leq p \leq 1$$

Propiedades:

- **Función de masa de probabilidad:** En cada prueba de un experimento de Bernoulli, se asigna el valor 1 a una variable aleatoria si el resultado es un éxito y 0 si es un fracaso. La suma de estas variables aleatorias individuales da como resultado una nueva variable aleatoria que representa el total de éxitos obtenidos, la cual se distribuye de manera binomial [19]. Por lo tanto, se puede calcular la probabilidad de obtener exactamente k éxitos en n pruebas mediante la expresión:

$$P(X = k) = \binom{n}{k} p^k (1 - p)^{n-k}$$

Esta ecuación incorpora el coeficiente binomial $\binom{n}{k}$, lo que da el nombre de la distribución binomial.

- **Relación con Poisson:** La distribución de Poisson puede ser usada para aproximar la distribución binomial cuando el número de ensayos es grande y la probabilidad de éxito es pequeña.
- **Relación con la Distribución Normal:** Cuando n es grande ($n > 30$) y p es pequeño ($p < 0,01$), la distribución binomial puede ser aproximada por una distribución normal con media $\mu = np$ y varianza $\sigma^2 = np(1 - p)$:

$$X \sim N(np, \sqrt{np(1 - p)})$$

Esta aproximación es muy útil para calcular probabilidades de manera eficiente, tanto numéricamente (precisión) como computacionalmente (tiempo). Usar la distribución normal en lugar de la binomial reduce la complejidad de los cálculos, especialmente en simulaciones que involucren un gran número de ensayos.

- **Ejemplo de Uso:** Si queremos calcular la probabilidad de obtener un cierto número de éxitos en un gran número de ensayos con una baja probabilidad de éxito en cada ensayo, la aproximación normal nos permite realizar este cálculo de manera más rápida y precisa.

2.3.4. Distribución Uniforme

Esta distribución de probabilidad está fundamentada en que todos los valores de un intervalo específico tienen la misma probabilidad de aparecer. El intervalo puede ser tanto abierto como cerrado, es decir, el intervalo puede estar comprendido entre los valores "a" y "b" estando alguno de ellos incluido, ninguno o ambos. [20]

Para una variable aleatoria continua X que se distribuye uniformemente en el intervalo $[a, b]$, su notación es la siguiente:

$$X \sim \text{Uniforme}(a, b), \quad -\infty < a < b < \infty$$

Propiedades:

- **Función de densidad de probabilidad:** La función de densidad de probabilidad (FDP) para una distribución uniforme continua en el intervalo $[a, b]$ se define como:

$$f(x) = \begin{cases} \frac{1}{b-a} & \text{si } a \leq x \leq b \\ 0 & \text{en otro caso} \end{cases}$$

Esta función indica que la probabilidad es constante para todos los valores en el intervalo $[a, b]$.

En la siguiente imagen 2.4 podemos ver representada la Distribución Uniforme, en ella vemos definido el intervalo que la comprende, el cual está compuesto por las variables "a" y "b", que son sus valores mínimo y máximo respectivamente.

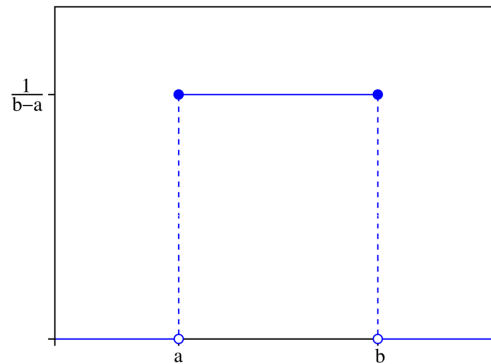


Figura 2.4: Distribución Uniforme [20]

- **Aplicaciones y Generación de Números Aleatorios:** La generación de números aleatorios, es fundamental en la simulación de procesos estocásticos. La distribución uniforme se utiliza en términos de simulación cuando se requiere un conjunto de valores equiprobables dentro de un intervalo. La **independencia estadística** y la **aleatoriedad** son dos propiedades clave en la generación de números aleatorios. La independencia asegura que la aparición de un número en la secuencia no influye en la aparición de otro, lo que es esencial para evitar patrones predecibles. La aleatoriedad implica que cada número en el conjunto tiene la misma probabilidad de aparecer. Estas propiedades garantizan que los números generados simulen adecuadamente el comportamiento de fenómenos naturales y sistemas complejos. Además, los números aleatorios uniformemente distribuidos son a menudo transformados para generar otras distribuciones más complejas, como la distribución normal o la exponencial, mediante técnicas como la transformación inversa y el método de aceptación-rechazo.

- **Base de la Generación de Datos Aleatorios** La generación de números aleatorios es la base de la simulación de procesos y la forma de lograrlo es mediante el uso de algoritmos que generan secuencias de números pseudoaleatorios. Estos números, aunque generados de manera determinista, pasan diversas pruebas estadísticas que los hacen indistinguibles de números aleatorios verdaderos para la mayoría de aplicaciones prácticas.

Una propiedad interesante es que la media de 12 números aleatorios uniformemente distribuidos se distribuye aproximadamente como una distribución normal. Esta propiedad es una manifestación del **Teorema Central del Límite**, que establece que la suma de un gran número de variables aleatorias independientes e idénticamente distribuidas tiende a una distribución normal, independientemente de la distribución original de las variables.

- **Estadística Computacional** La estadística computacional es un campo que se enfoca en el uso de computadoras para realizar inferencias estadísticas. La inferencia estadística es el proceso de analizar datos de una muestra para hacer estimaciones, hipótesis o predicciones sobre un campo

Capítulo 2. Estado del Arte

más amplio. La estadística computacional supone la base de la simulación ya que permite generar aleatoriedad de manera controlada y utilizar esta aleatoriedad para explorar el comportamiento de sistemas complejos.

Un recurso útil para aprender sobre estadística computacional es el curso "Estadística Computacional" disponible en [este enlace](#) [21] en el cuál se proporciona una base sólida de la generación de números aleatorios para la simulación y el análisis de datos.

En este Trabajo Fin de Grado, se han utilizado técnicas de estadística computacional para generar datos aleatorios que alimentan el modelo de simulación del taller de mantenimiento de vehículos y asegurar que los datos generados sean representativos y permitan realizar inferencias precisas sobre el comportamiento del sistema.

Capítulo 3

Modelo Empirico

En este capítulo se busca dar una visión descriptiva y visual del modelo de simulación implementado, presentando los distintos elementos que forman el mismo.

3.1. Visión General del Sistema

Esta primera figura 3.1 representa el sistema de un Taller de Mantenimiento de Vehículos, dividido en diferentes trayectorias y procesos que puede seguir el vehículo de un cliente desde su Entrada hasta su Salida.

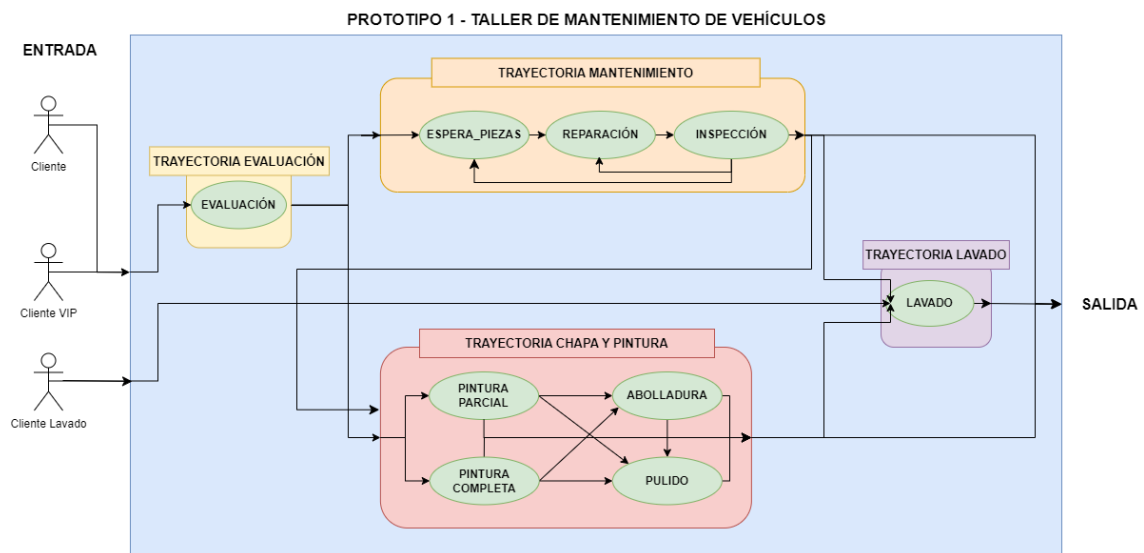


Figura 3.1: Visión General del Sistema Prototipo 1

En este diagrama de flujo podemos observar la organización y el funcionamiento interno del Taller, diferenciando entre los distintos tipos de clientes y servicios. Es importante comprender que cada uno de los sucesos por los que puede ir el cliente están determinados en base a una cierta probabilidad, y cuando un

cliente pasa por dichos eventos los mismos generan cambios en el sistema, tanto en el tiempo empleado por cada cliente en cada uno de los eventos como en el coste que tienen cada uno de los servicios para dicho cliente.

- **ENTRADA:** En esta parte se diferencian 3 tipos distintos de posibles entradas o clientes; El cliente (normal) y el cliente VIP siguen la misma trayectoria, sin embargo el segundo goza una mayor prioridad mayor para hacer uso de un servicio con respecto al primero; Por otra parte, el cliente Lavado únicamente va al evento de Lavado directamente, teniendo una prioridad para ser atendido menor a la del cliente y el cliente VIP.
- **TRAYECTORIAS:** Dentro de esta sección se definen las distintas variantes o caminos que pueden seguir los distintos tipos de clientes, según la casuística.
 - *Evaluación:* Todos los vehículos excepto aquellos exclusivamente que quieren lavado, son sometidos a una *Evaluación* inicial para determinar las necesidades de mantenimiento o adecuación del vehículo, siendo redirigidos a la trayectoria de *Mantenimiento* general del vehículo o a la trayectoria de *Chapa y Pintura*.
 - *Mantenimiento:* Esta trayectoria esta compuesta a su vez por distintas fases o eventos, en primera instancia se trata de determinar si las piezas necesarias para esta reparación se hayan en el propio taller o por el contrario se deben pedir al proveedor de confianza. Una vez que tenemos las piezas identificadas y a nuestra disposición se avanza al evento de *Reparación* en el cual se realizará el mantenimiento pertinente a cada automóvil; Por último el cliente pasa por el evento de *Inspección* en el cual se determina si se ha realizado bien el mantenimiento sobre el vehículo o por el contrario se ha identificado algún error y el cliente debe pedir nuevamente piezas para la reparación posterior o simplemente debe repararse algo que no se hizo de forma precisa.
 - *Chapa y Pintura:* Esta trayectoria está destinada a clientes que quieren arreglar la carrocería exterior de su vehículo y en la cual se le ofrecen los servicios de *Pintura Parcial*, *Pintura Completa*, *Abolladura* y/o *Pulido* del vehículo.
 - *Lavado:* Finalmente en esta trayectoria se procede al lavado del automóvil, pudiendo acceder a este servicio directamente o tras haber recibido los servicios de *Mantenimiento* y/o de *Chapa y Pintura*.
- **SALIDA:** Una vez que el vehículo a pasado por las etapas de servicio pertinentes el cliente sale del sistema y se le devuelve el vehículo al mismo.

3.2. Árbol de Decisión

La simulación tiene como objetivo final realizar un estudio sobre una casuística y acercarse lo máximo posible a la realidad. Por ello, como he mencionado anteriormente los acontecimientos suceden en base a cierta probabilidad pa-

3.2. Árbol de Decisión

ra representar así la naturaleza aleatoria de la simulación así como ocurre en cualquier situación de la vida real.

Por tanto creo que la forma más idónea de representar visualmente estos sucesos basados en datos estadísticas es mediante un Árbol de Decisión. En la figura 3.2 podemos ver representadas las distintas fases de las posibles trayectorias que puede seguir un cliente. Además, en las ramas del árbol podemos observar distintas características que diferencian a estas ramas, como son la leyenda que las acompaña y el color de las mismas.

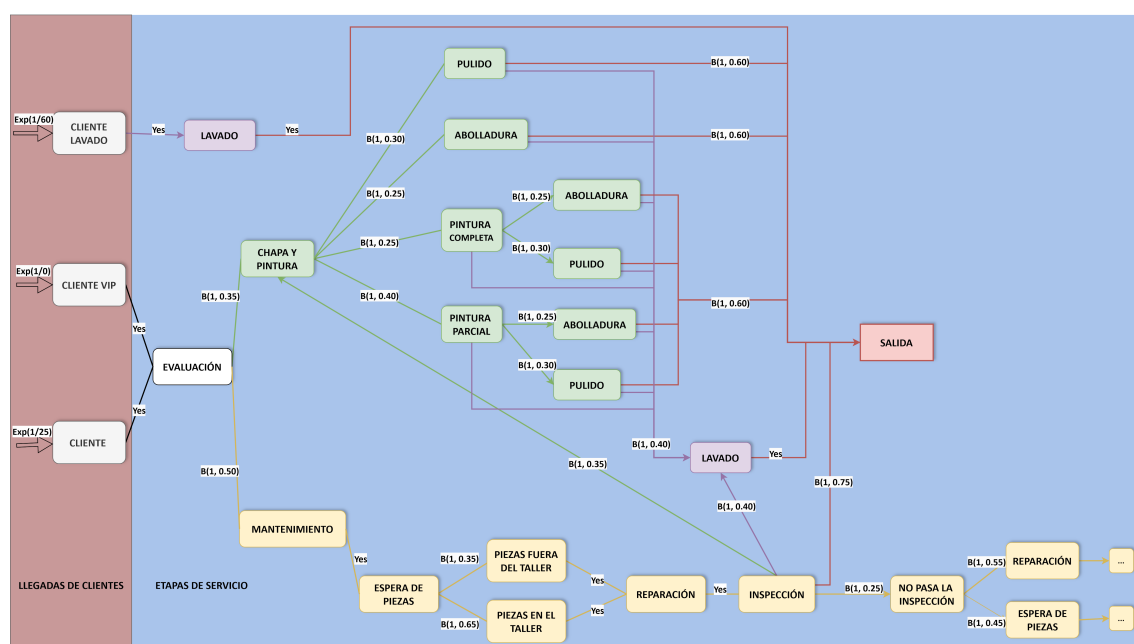


Figura 3.2: Árbol de Decisión Prototipo 1

Llegadas de Clientes

En esta sección diferenciada en el dibujo con un color de fondo granate podemos observar 3 flechas más grandes que las del resto del diagrama que identifican las llegadas de los clientes. Cada una de esas flechas viene acompañada por una leyenda en la que se expresa la frecuencia de llegadas de estos clientes, la cuál viene regida por la función exponencial, "**Exp**" con tasa λ . En nuestro caso las tasas de llegadas se calculan mediante el uso de la función `timeout()`, a la cuál se le pasa como parámetro de entrada la función exponencial con tasas de 25, 40 y 60 minutos para los clientes, clientes VIP y clientes de lavado respectivamente.

Etapas de Servicio

Esta parte está diferenciada por el color azul del fondo y en ella se distinguen en color Verde las entidades respectivas a la trayectoria de "Chapa y Pintura", en color Amarillo las respectivas a la de "Mantenimiento", en color Morado la de "Lavado", en color Rojo la "Salida" y por último en un color Gris más neutro la fase de "Evaluación". Por otra parte también se puede apreciar que las ramas

que unen las distintas entidades son también de dichos colores, según el color de las mismas se indica al tipo de entidad al que se dirige dependiendo de la trayectoria a la que pertenezca la entidad, esto se hace para que el conjunto de la figura sea fácilmente comprensible. En las etapas de servicio la leyenda que acompaña a las ramas es la distribución estadística que siguen esas decisiones, en este caso todas son de tipo binomial o **"B"** en la figura. El segundo argumento de esta función es el que nos indica la tasa p de esa decisión, 0.40 por ejemplo sería un 40% en base a la distribución binomial. También se puede observar el mensaje **"Yes"**, el cual indica que el cliente ira en esa dirección por defecto.

Salida

Tras completar todos los servicios necesarios, los vehículos son dirigidos a la salida.

3.3. Escenarios

Como ya he comentado en numerosas ocasiones, cuando un cliente entra en el sistema sigue una trayectoria u otra dependiendo de las decisiones basadas en cierta probabilidad, las diferentes opciones de los clientes generan distintos escenarios. Principalmente el cliente puede seguir la trayectoria de Mantenimiento, la de Chapa y Pintura o bien una combinación de ambas, además cabe remarcar que en Mantenimiento existe la fase de Inspección la cuál si es desfavorable tiene un efecto directo en el gasto en tiempo y dinero del cliente.

3.3.1. Prototipo 1

Escenario 1 - Inspección Deficiente

En este escenario representado mediante la figura 3.3 se supone que la inspección del vehículo tras su paso por la reparación no es satisfactoria. Además, en la etapa de "Espera de Piezas" existen dos posibilidades, que las piezas estén en el taller o que no lo estén, en nuestro caso el precio y tiempo tienen los valores referidos al caso de que las piezas si están en el taller. Además la flecha discontinua viene a representar la posibilidad de que un cliente repita varias veces el mismo recorrido debido a inspecciones desfavorables.

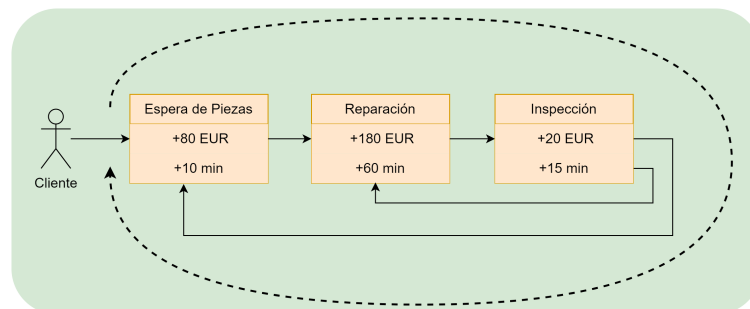


Figura 3.3: Prototipo 1 Escenario 1 - Valoración en la *Inspección* negativa

Escenario 2 - Combinación de Trayectorias

En la siguiente figura 3.4 que representa otro escenario los valores afectados son también el dinero y el tiempo. En este caso el cliente pasaría primero por la trayectoria de Mantenimiento y en segunda instancia podría requerir el servicio de Chapa y Pintura, el de Lavado o la salida del sistema. En concreto nosotros ilustraremos el caso más completo, en el cual el cliente pasaría por las 3 trayectorias secuencialmente. Además, a esto habría que sumarle distintas casuísticas dentro de cada trayectoria las cuales no se especifican por simplicidad.

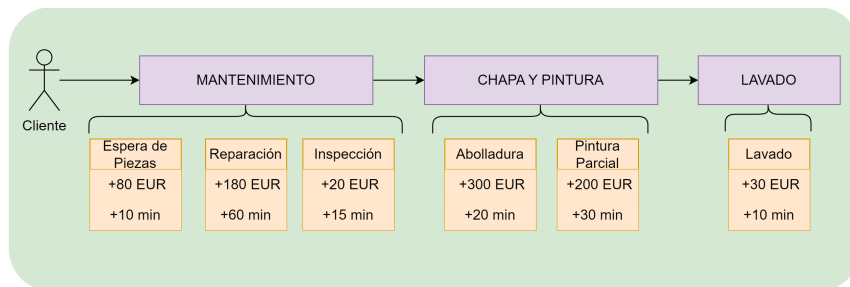


Figura 3.4: Prototipo 1 Escenario 2 - Cliente atraviesa varias trayectorias

3.3.2. Prototipo 2

En el Prototipo 2 se añade complejidad al sistema añadiendo entre otras cosas un nuevo tipo de cliente, el "Cliente ITV" el cual sigue una nueva trayectoria ITV. Además, se amplía la trayectoria de *Lavado* con nuevas tareas pudiendo elegir el cliente entre un servicio de lavado de su vehículo mediante túnel de lavado, pistola o por el contrario un lavado completo que implica lavado por un empleado del exterior y del interior del automóvil.

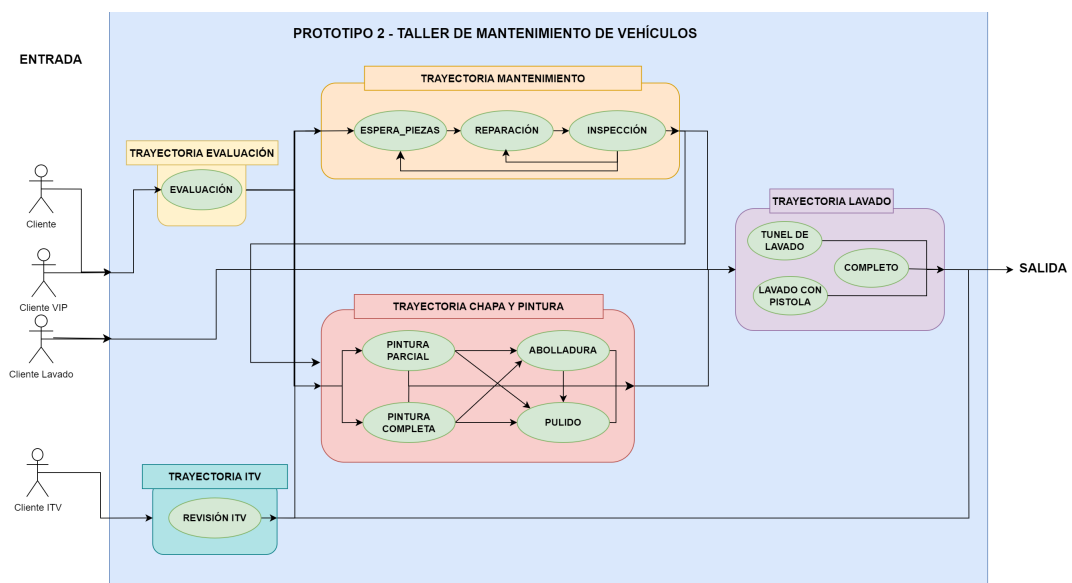


Figura 3.5: Prototipo 2 - Visión General del Sistema Prototipo

Replicando el diagrama 3.1 se muestra en la figura anterior (3.5) la ampliación del sistema de los servicios mencionados con respecto a las etapas que componen el de *Lavado* y de *ITV*.

Escenario 1 - Revisión ITV con Fallo Grave

En esta figura 3.6 se pueden ver representadas las características del nuevo cliente "Cliente ITV", cada uno de ellos tienen asociado un vehículo con los atributos "Tipovehículo" y "Antigüedad" a través de una función generadora de vehículos aleatorios, esto sólo sucede para este tipo de clientes.

Además en este escenario se ilustra lo que sucedería si se encuentra un fallo grave con el cual el cliente no pasaría la ITV "real". Por tanto este automóvil es redirigido a la trayectoria de Mantenimiento en la que se realizarían los distintos servicios ya conocidos. En caso de que se encontrasen fallos leves o ningún fallo en el vehículo el cliente abandonaría en ese momento el sistema.

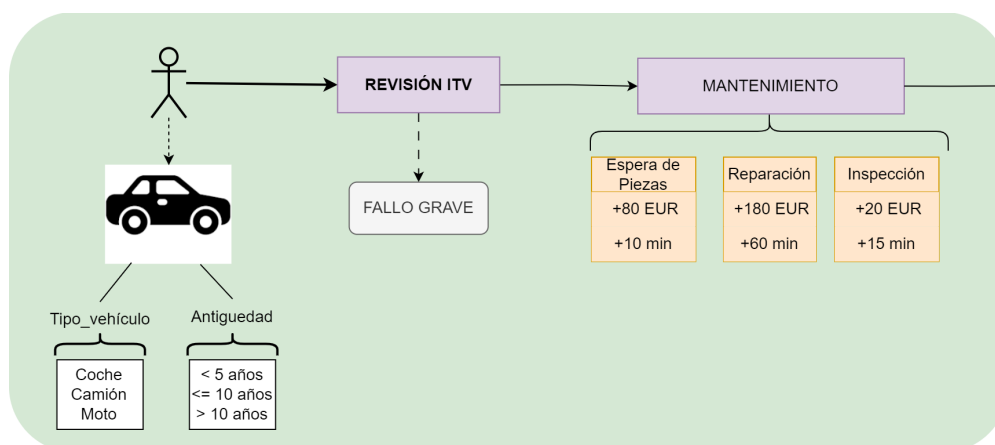


Figura 3.6: Diagrama de Flujo *Prototipo 2 Escenario 1 - ITV con Fallo Grave*

En lo referente a la figura 3.6 podemos observar que el vehículo posee dos atributos fundamentales, "Tipo_vehículo" que nos indica el tipo de automóvil que posee el cliente y "Antigüedad" que como la propia palabra indica la edad del coche, mediante un valor entero en el intervalo [0,20], aunque se divide la evaluación en menor de 5 años, menor o igual a 10 años y mayor a 15 años ya que estos 3 intervalos son críticos en relación con el número de accidentes de cada tipo de coche.

En el desarrollo de la simulación y en esta sección se han tenido en cuenta únicamente un **subconjunto de todos los escenarios** posibles dentro de un Taller de Mantenimiento de Vehículos. Se han modelado escenarios simples y combinaciones más complejas de estos, sin embargo, cabe remarcar que no todos los escenarios y las condiciones de los mismos son realistas o necesariamente existen. Es decir, los escenarios desarrollados nos permiten **evaluar el desempeño** de determinadas tareas en un contexto operativo bajo un entorno de simulación, pero existen **infinitos escenarios**, posibilidades y casuísticas más **que este modelo no cubre**.

Capítulo 4

Prototipado

Esta sección se centra en las fases de diseño, implementación y simulación del sistema propuesto, explorando las estrategias de diseño adoptadas y la lógica subyacente, detallando también la implementación y el desarrollo técnico del proyecto. Este apartado es crucial ya que no solo explica cómo se ha construido la simulación y las justificaciones para la elección de determinadas técnicas, sino también cómo cada componente y función construyen el modelo simulado.

4.1. Diseño Prototipo 1

En la sección de diseño se establecen los fundamentos metodológicos y técnicos que dirigirán el desarrollo e implementación del modelo de simulación. En este apartado se representan los flujos que pueden seguir los clientes dependiendo de las decisiones que se toman y sus consecuencias en términos de simulación. En las figuras los rectángulos con fondo rojo son las funciones desarrolladas (detalladas en la Implementación 4.3) que determinan cada una de esas decisiones. A continuación, se ilustran las distintas etapas de *Evaluación*, *Chapa y Pintura* e *Inspección* mediante Diagramas de Flujo o Entidad-Relación.

4.1.1. Evaluación de Necesidades del Vehículo

La evaluación forma parte de la primera fase por la que los clientes y clientes VIP tienen que pasar. En primera instancia se evalúa si el automóvil necesita Mantenimiento o solamente labores de Chapa y Pintura 4.1.3 relacionadas únicamente con reparaciones sobre la carrocería del mismo. Una vez que se determina cual de los dos servicios (Mantenimiento o Chapa y Pintura) necesita el cliente es dirigido a una trayectoria o a otra. En el caso de seguir la trayectoria de Mantenimiento el vehículo sigue una serie de fases explicadas en la sección 4.1.2 y tras salir del Mantenimiento es evaluado nuevamente por si necesitase arreglos de Chapa y Pintura. Todo se representa en la siguiente figura 4.1.

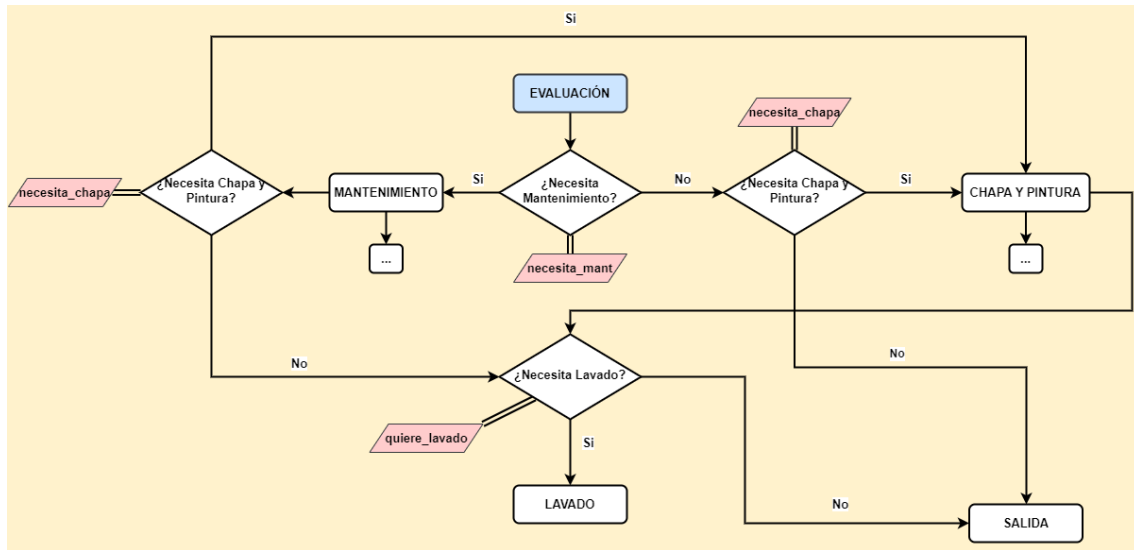


Figura 4.1: Diagrama de Flujo *Evaluación*

Esta implementación se hace de esta manera ya que se estima oportuno que si el cliente necesita Mantenimiento y Chapa y Pintura, el segundo sea ofrecido de forma posterior al primero.

4.1.2. Flujo de Trabajo Mantenimiento

La trayectoria de mantenimiento se compone de las fases de espera de piezas, en la cual se obtienen las piezas necesarias para la reparación, ya sean mediante un pedido de las mismas o porque ya se encuentren en el taller. Mas tarde se realizaría la reparación del vehículo y en última instancia se realiza una inspección que determina si todo el mantenimiento ha sido satisfactorio y el vehículo abandona esta trayectoria en óptimas condiciones o si por el contrario todavía hace falta alguna tarea por hacer sobre el automóvil. La unión de estas 3 fases se ve reflejada en la figura 4.2 y más adelante veremos en más detalle cada una de ellas.

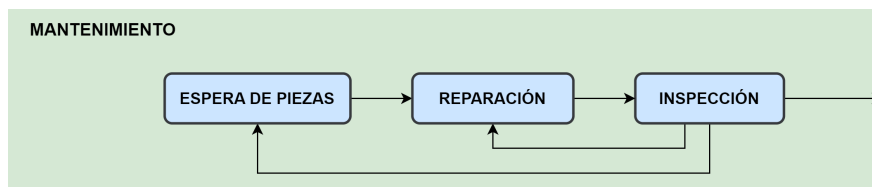


Figura 4.2: Ilustración *Mantenimiento*

Flujo de Trabajo Espera de Piezas

En la siguiente imagen (4.3) se representa el flujo de trabajo en la fase de *Espera de piezas* en la cuál se cuestiona si las piezas requeridas para el mantenimiento del coche se ubican ya en el taller o por el contrario se deben pedir al proveedor.

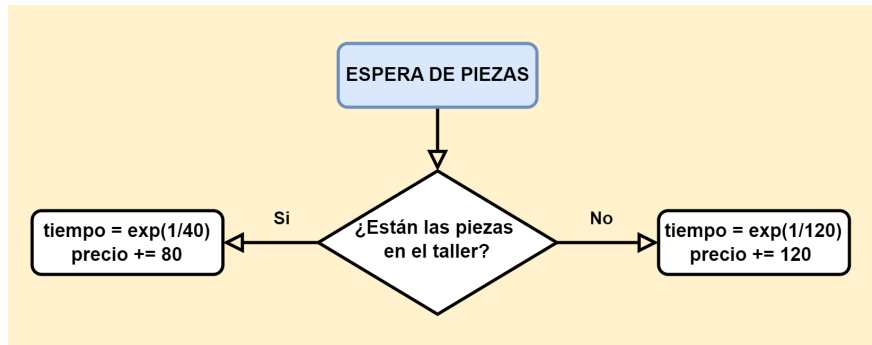


Figura 4.3: Diagrama de flujo *Espera de Piezas*

En el primer caso tanto el tiempo como el precio serán menores ya que las piezas ya se encontraban en el taller previamente y por tanto no hay que esperar a que lleguen, así como tampoco hay que pagar ningún envío por ellas. Podemos ver como según lo mencionado las variables "tiempo" y "precio" varían de distinta manera, además las unidades son minutos y euros correspondientemente.

Flujo de Trabajo Reparación

Durante la reparación (representada mediante la figura 4.4) he considerado que no había ningún condicionante y por tanto durante la ejecución de esta fase únicamente se aumenta el tiempo y el dinero de factura que el cliente deberá pagar.

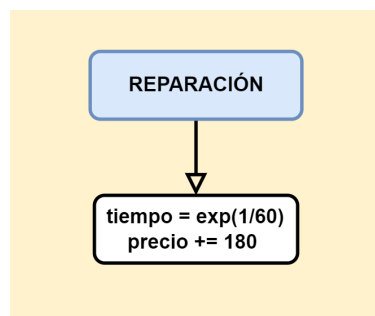


Figura 4.4: Diagrama de Flujo *Reparación*

Flujo de Trabajo Inspección

Durante la Inspección se evalúa si las tareas sobre el vehículo se han hecho de forma adecuada, en caso de que dicha comprobación sea exitosa el cliente sale del sistema. En caso contrario se determina si el cliente únicamente necesita reparación sin necesidad de más piezas o si necesita adquirir las piezas y realizar de nuevo la reparación. Esto se ve representado en la siguiente imagen 4.5.

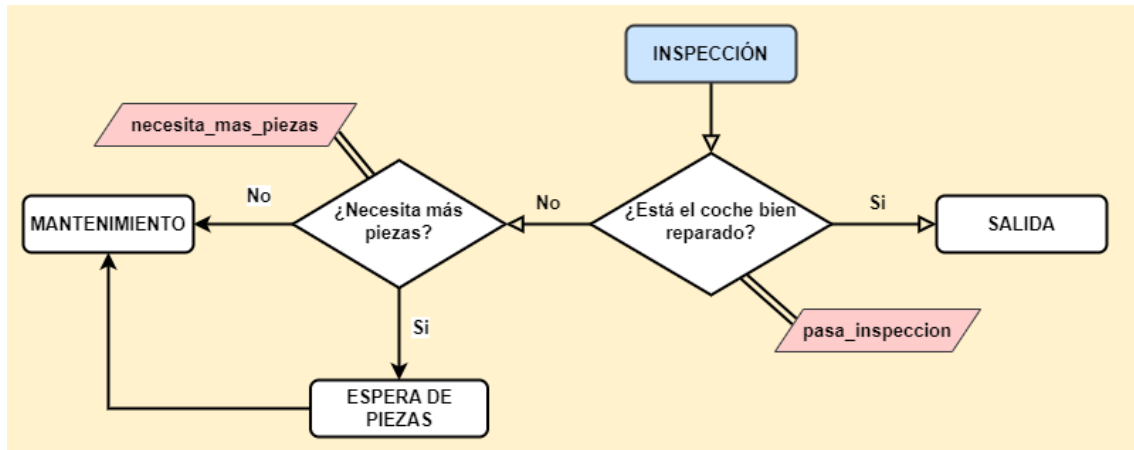


Figura 4.5: Diagrama de Flujo *Inspección*

4.1.3. Flujo de Trabajo Chapa y Pintura

La fase de *Chapa y Pintura* representada a continuación (4.6) está compuesta a su vez por distintas etapas dependiendo de los requerimientos del vehículo, por motivos de diseño y lógica se ha decidido que tanto la pintura parcial como la completa puedan suceder únicamente como etapas finales de esta fase. Además, ambas dos son excluyentes entre sí, debido a que si se realiza pintura parcial sobre el vehículo no tiene sentido que se realice una pintura completa y viceversa.

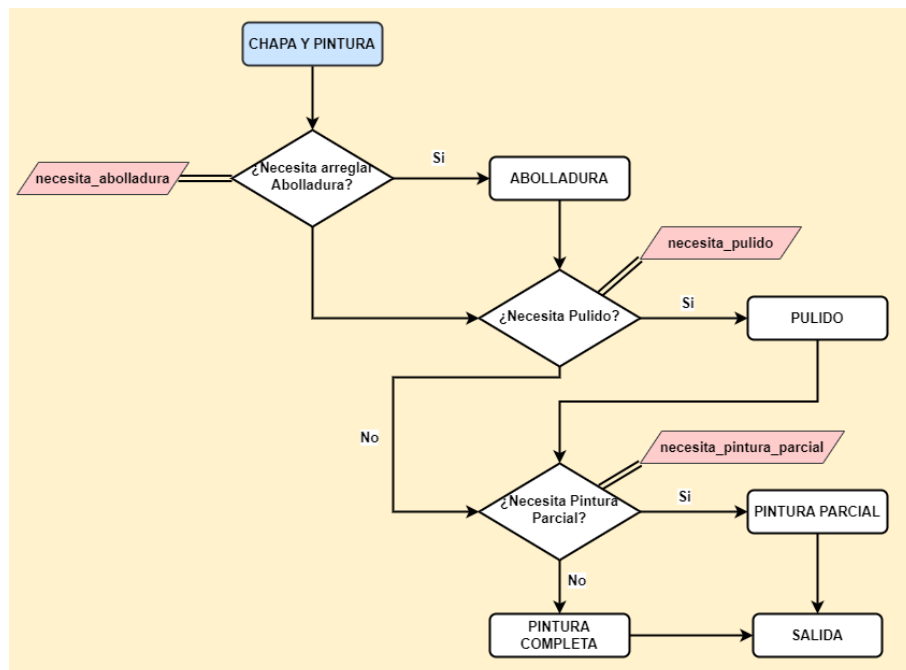


Figura 4.6: Diagrama de Flujo *Chapa y Pintura*

Flujo de Trabajo Componentes Chapa y Pintura

Dentro de la trayectoria de chapa y pintura los distintos servicios que se ofrecen carecen dentro de ellos de ninguna evaluación o decisión por tanto por simplicidad voy a ilustrar únicamente uno de estos procesos, el *Pulido* en la siguiente imagen 4.7.

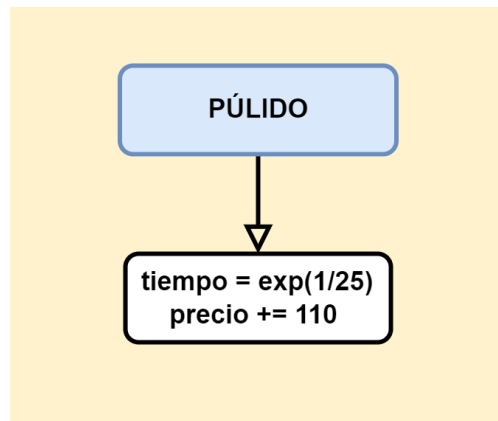


Figura 4.7: Diagrama de Flujo *Pulido*

Con respecto al orden que siguen los clientes en esta trayectoria se ha definido así por un tema de diseño, y siguiendo la lógica de que si un cliente entra en esta trayectoria los servicios de Pintura, bien sea parcial o completa serían los últimos que un cliente emplearía, ya que carece de sentido aplicar la pintura al coche para después arreglarle la abolladura.

4.1.4. Flujo de Trabajo Lavado

En este caso la trayectoria de lavado, únicamente esta compuesta por el servicio de Lavado, el cual no está sujeto a ninguna decisión. Por tanto, al igual que en el apartado anterior solo se ilustran en la imagen 4.8 las consecuencias en términos de coste monetario y de tiempo del uso de este servicio.

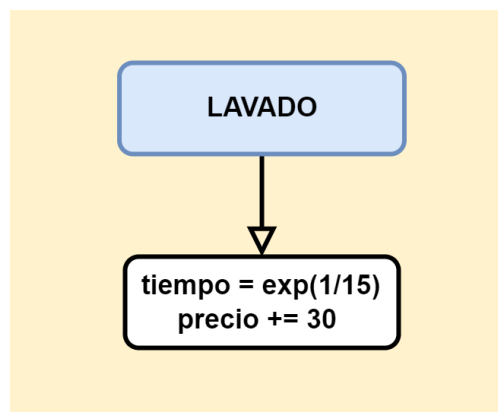


Figura 4.8: Diagrama de Flujo *Lavado*

4.2. Diseño Prototipo 2

Con respecto al primer prototipo los cambios más relevantes del sistema son la adición de una nueva trayectoria y tipo de cliente enfocado en simular la llegada de clientes que pretenden hacer una revisión de su vehículo previo a hacer la revisión oficial de la ITV, estos clientes además poseen unas características que no poseen el resto de clientes, ya que son "Impacientes" o lo que es lo mismo, necesitan ser atendidos de forma relativamente rápida, esto se ha modelado así porque se presupone que a estos clientes se los ha citado próximamente para realizar la ITV y necesitan que se les haga una evaluación previa antes de ir a la revisión oficial, además nos permite añadir un nuevo tipo de cliente con distintas características a los anteriores.

4.2.1. Flujo de Trabajo ITV

En este apartado podemos ver representadas los distintos factores que provocan que el cliente tenga cierta probabilidad de que se encuentre un fallo leve o grave en la revisión del vehículo, estos dos factores fundamentales son el tipo de vehículo y la antigüedad del mismo. En cuyo caso se determine que el cliente posee un fallo grave en su automóvil es redirigido a la trayectoria de *Mantenimiento* en la cual pueda ser tratado y se reparen los fallos graves que no le permitirían superar satisfactoriamente la revisión oficial.

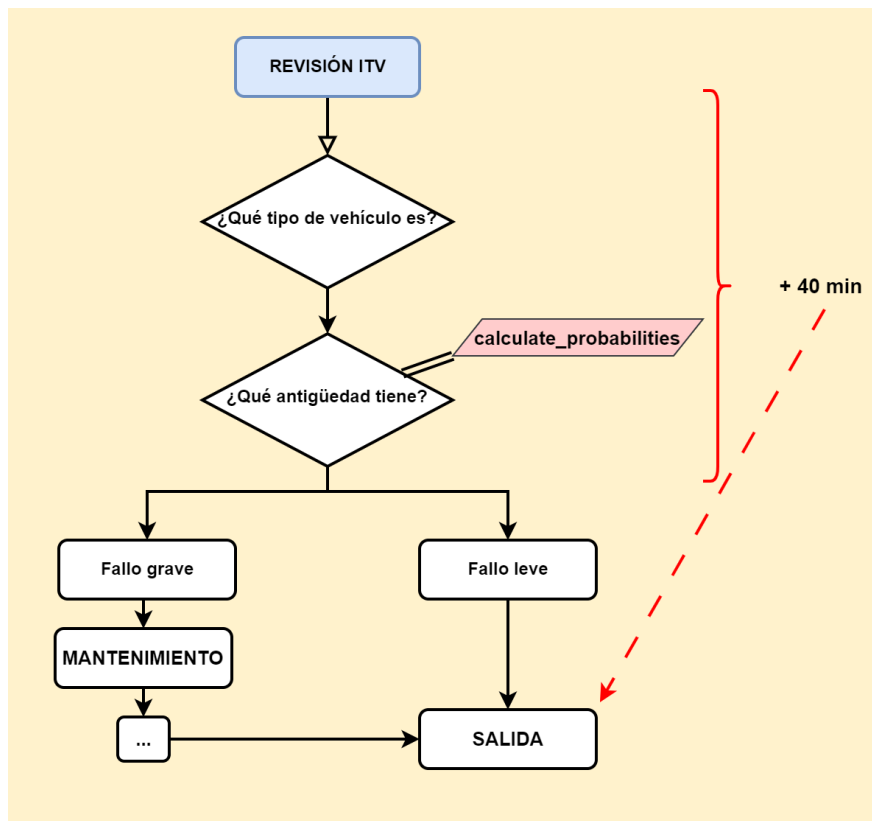


Figura 4.9: Diagrama de Flujo *Revisión ITV*

Por otra parte, como también se puede observar en la figura 4.9 este tipo de clientes si pasan más de 40 minutos en cola se impacientarán u se irán del sistema sin recibir ningún servicio y sin pagar.

4.2.2. Ampliación Flujo de Trabajo Lavado

En la siguiente imagen 4.10 podemos ver la ampliación realizada sobre la entidad de *Lavado* implementada en el prototipo 2, en la misma el cliente "decide" que tipo de lavado prefiere, teniendo un efecto variable tanto en el tiempo como en el coste gastado por el cliente en este servicio.

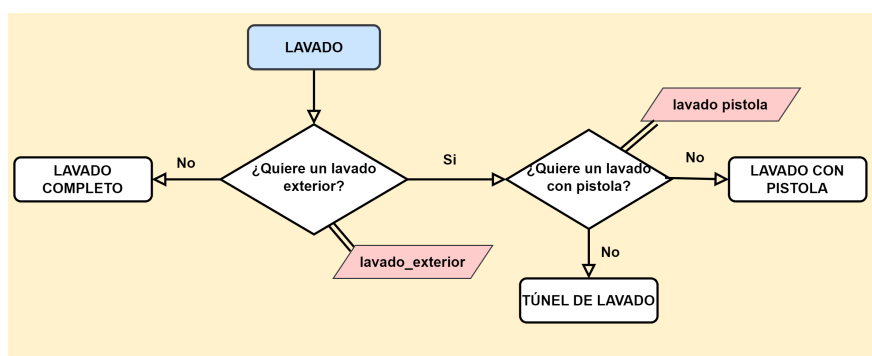


Figura 4.10: Diagrama de Flujo *Lavado*

Por simplicidad no se añaden las consecuencias concretas de la elección de cada tipo de lavado.

4.3. Implementación

En este apartado se detalla la implementación de la simulación realizada, describiendo tanto la arquitectura de software como las funciones específicas desarrolladas para modelar los procesos del sistema. Se hace mención de las funciones que se han usado o implementado durante el desarrollo de este proyecto. Por una parte las funciones basadas en distribuciones estadísticas, así como las de las trayectorias y las que rigen las decisiones del modelo.

4.3.1. Funciones probabilísticas

- `rexp`: La función `rexp` parte del paquete base de R y genera números aleatorios a partir de una distribución exponencial.
 - **Parámetros:**
 - `n`: número entero que indica la cantidad de observaciones aleatorias a generar.
 - `rate`: la tasa de ocurrencia, λ , el inverso de la media. Debe ser un número positivo.

Capítulo 4. Prototipado

- **Valor de retorno:** Un vector de números aleatorios de longitud n que siguen una distribución exponencial con la tasa especificada.
- **Ejemplo de uso:** En nuestro modelo de simulación estas funciones sirven para modelar tanto el tiempo de llegadas entre clientes como el tiempo que tarda cada cliente en cada actividad, esto se representa en la siguiente figura (4.11) :

```
# Generador de clientes con una tasa de 25
# minutos entre cada cliente
add_generator("cliente",
  trajectory() %>%
    set_attribute("tipo_cliente", 1) %>%
    join(trayectoria_evaluacion()),
  function() rexp(n = 1, rate = 1/25),
  priority = 2, mon = 2)

# Tasa en minutos que tarda un cliente en
# realizar actividad concreta
timeout(function() rexp(1, rate = 1/15)) ##15 mins
```

Figura 4.11: Código para la generación de clientes y tiempo de actividad

- `rbinom`: La función `rbinom` también es parte del paquete base en R y se utiliza para generar números aleatorios en base a la distribución binomial.

- **Parámetros:**

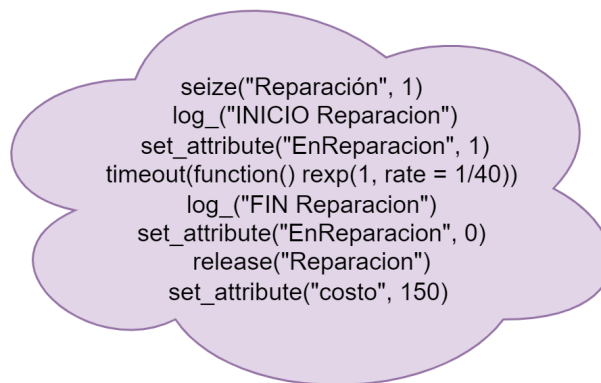
- `n`: número entero que indica cuántas realizaciones de la distribución binomial se generarán.
 - `size`: número entero que representa el número de ensayos en cada realización.
 - `prob`: este parámetro indica probabilidad de éxito en cada ensayo, y debe ser un valor entre 0 y 1.
- **Valor de retorno:** Un vector de enteros donde cada elemento es el número de éxitos en los correspondientes `size` ensayos.
 - **Ejemplo de uso:** En nuestra implementación en `simmer` [8] se usa para modelar la probabilidad de la ocurrencia de los distintos factores que afectan al sistema, es decir, actúan como funciones de decisión. En esta figura 4.12 podemos ver uno de estos ejemplos de decisión:

```
# Determina si se necesita mantenimiento con  
# una probabilidad del 50  
necesita_mant <- function() rbinom(1, 1, 0.50) == 1
```

Figura 4.12: Código para la toma de decisiones

4.3.2. Funciones de las trayectorias

Cada trayectoria está compuesta por distintas fases o *entidades* las cuales siguen la estructura de la imagen 4.13 que sirve tanto para registrar los datos estadísticos de la simulación como para generar las trazas que más tarde servirán para seguir la trayectoria seguida por el cliente durante su paso por las distintas etapas así como para la posible depuración del código.



```
seize("Reparación", 1)  
log_("INICIO Reparacion")  
set_attribute("EnReparacion", 1)  
timeout(function() rexp(1, rate = 1/40))  
log_("FIN Reparacion")  
set_attribute("EnReparacion", 0)  
release("Reparacion")  
set_attribute("costo", 150)
```

Figura 4.13: Nube de Contenido Entidades

- `log_`: Utilizada para registrar eventos o actividades durante la ejecución del programa. Es común en entornos que requieren un seguimiento detallado del comportamiento del sistema.

- **Parámetros:**

- `mensaje`: El mensaje a registrar.

- **Ejemplo de uso:**

```
log_('INICIO ....')
```

- `timeout`: Usado para gestionar el tiempo máximo de ejecución de tareas que pueden ser prolongadas, evitando que procesos se ejecuten indefinidamente y manteniendo un control sobre los mismos.
- `seize` y `release`: Estas funciones se utilizan para gestionar la adquisición y liberación de recursos en un entorno de simulación.
 - `seize`: Reserva un recurso para una entidad. Esto es útil en contextos donde los recursos son limitados y deben ser gestionados cuidadosamente.

- **Parámetros:**

- ◊ recurso: El nombre del recurso a reservar.
- ◊ cantidad: La cantidad del recurso a reservar (opcional, por defecto es 1).

- **Ejemplo de uso:**

```
seize('Reparación', 1)
```

- **release:** Libera un recurso previamente reservado por una entidad, permitiendo que otros puedan usarlo.

- **Parámetros:**

- ◊ recurso: El nombre del recurso a liberar.
- ◊ cantidad: La cantidad del recurso a liberar (opcional, por defecto es 1).

- **Ejemplo de uso:**

```
release('Reparación')
```

- **set_attribute:** Esta función se utiliza para establecer atributos personalizados para las entidades en la simulación. Los atributos pueden ser cualquier tipo de dato y se utilizan para almacenar información específica sobre cada entidad.

- **Parámetros:**

- nombre: El nombre del atributo.
- valor: El valor del atributo.

- **Ejemplo de uso:**

```
set_attribute('EnReparacion', 1)
```

4.4. Simulación

Nuestro modelo de simulación se desarrolla en dos ficheros principales, **simmer.R** y **app.R** en el primero se desarrolla la implementación de la simulación en sí misma, mientras que el segundo fichero sirve para la visualización de los datos.

4.4.1. Fichero de Simulación - simmer.R

Tras la ejecución de la simulación usando el paquete `simmer` [8] se genera por cada muestra o repetición así como de todas las simulaciones, una serie de ficheros en los que se almacenan los datos capturada más relevante, cada uno

de estos ficheros almacena un tipo concreto de información. En la siguiente imagen 4.14 vemos la disposición de los ficheros de una simulación concreta









 arrivals	26/05/2024 18:03	Archivo de valores...	5 KB
 atributos	26/05/2024 18:03	Archivo de valores...	43 KB
 costos_promedio	26/05/2024 18:03	Archivo de valores...	1 KB
 facturas	26/05/2024 18:03	Archivo de valores...	2 KB
 info_adicional	26/05/2024 18:03	Archivo de valores...	1 KB
 recursos	26/05/2024 18:03	Archivo de valores...	22 KB
 tiempos_espera	26/05/2024 18:03	Archivo de valores...	3 KB
 TRAZAS_sim_004	26/05/2024 18:03	Documento de tex...	35 KB

Figura 4.14: Ficheros generados tras la ejecución del código

A continuación se detalla una descripción de cada fichero así como los campos que los componen.

Llegadas (arrivals.csv) En este fichero se almacena la información relevante a las llegadas de los clientes, almacenándose sobre ellos su nombre o identificador, el momento de llegada, el momento de finalización y el tiempo de actividad de ese determinado cliente.

- **name:** Identificador del cliente.
- **start_time:** Tiempo de llegada.
- **end_time:** Tiempo de salida.
- **activity_time:** Tiempo de actividad.
- **finished:** Estado de finalización.
- **replication:** Réplica a la que pertenece el dato.

Atributos (atributos.csv) En este fichero se almacena toda la información referente a los atributos tanto de la los clientes como de las entidades, en ella se almacena por ejemplo el tipo de cliente, o la entidad en la que se encuentra el cliente en cada momento indicando el tiempo de su entrada y su salida de dicha entidad.

- **time:** Tiempo.
- **name:** Identificador del cliente.
- **key:** Nombre del atributo.
- **value:** Valor del atributo.

Capítulo 4. Prototipado

- **replication:** Réplica a la que pertenece el dato.

Costos Promedio (costos_promedio.csv) En el mismo se almacena el cálculo del coste promedio de cada tipo de cliente.

- **tipo_cliente:** Tipo de cliente.
- **costo_promedio:** Costo promedio.

Facturas (facturas.csv) En este se lleva un registro de la suma del gasto de cada cliente, según por las entidades por las que el mismo pase.

- **name:** Identificador del cliente.
- **costo_total:** Costo total acumulado por el cliente.

Información Adicional (infoadicional.csv) Este fichero guarda información adicional de las entidad que conforman nuestra simulación, como su capacidad, la longitud media de su cola, el tiempo medio de servicio por entidad o el número de personas que pasa por cada uno de estos recursos.

- **recurso:** Nombre del recurso.
- **capacidad:** Capacidad del recurso.
- **utilizacion:** Porcentaje de utilización del recurso.
- **longitud_media_cola:** Longitud media de la cola para el recurso.
- **tiempo_medio_servicio:** Tiempo medio de servicio.
- **tiempo_total_uso:** Tiempo total de uso del recurso.
- **num_served:** Número total de clientes servidos.

Recursos (recursos.csv) En este fichero se almacena también la información principal de los recursos, como el tiempo en que se accede a los mismos y su estado en esos determinados momentos.

- **resource:** Nombre del recurso.
- **time:** Tiempo.
- **server:** Número de servidores activos.
- **queue:** Longitud de la cola.
- **capacity:** Capacidad del recurso.
- **queue_size:** Tamaño de la cola.
- **system:** Estado del sistema.
- **limit:** Límite del recurso.
- **replication:** Réplica a la que pertenece el dato.

Tiempos de Espera (tiempos_espera.csv) En el mismo se refleja la información de cada cliente, con respecto a los tiempos medios de espera y de servicio.

- **name:** Identificador del cliente.
- **mean_waiting_time:** Tiempo medio de espera.
- **mean_time_in_system:** Tiempo medio en el sistema.

Fichero de Trazas (TRAZASSim00X.txt) Por último en el único fichero que no es un csv se imprimen las trazas en tiempo de ejecución de cada simulación, esto significa, que por cada punto de la trayectoria en que un cliente pasa por una fase se registra en este fichero. Se imprime en el fichero El INICIO y FIN de ese cliente en los determinados recursos o servicios, además de registrarse también información relevante con respecto a las decisiones que toma el cliente en su paso por el sistema. También se registra el tiempo de simulación en que ocurren dichos mensajes que se imprimen. Este documento ha sido de gran ayuda desde el primer momento para poder hacer un correcto seguimiento de los clientes y comprobar que hacían lo que se requería así como para ajustes posteriores en base a las observaciones sobre el mismo. A continuación, se muestra una parte de un fichero concreto de trazas 4.15.

```
5.31341: cliente0: INICIO Evaluacion
7.3294: cliente VIP0: INICIO Evaluacion
11.0563: cliente0: FIN Evaluacion
11.0563: cliente0: Necesita Mantenimiento
11.0563: cliente0: INICIO EsperaPiezas
11.0563: cliente0: No está la pieza en el taller, necesitan ser pedidas
11.0563: cliente VIP1: INICIO Evaluacion
18.1267: cliente VIP0: FIN Evaluacion
18.1267: cliente VIP0: Necesita Mantenimiento
18.1267: cliente VIP0: INICIO EsperaPiezas
18.1267: cliente VIP0: No está la pieza en el taller, necesitan ser pedidas
19.0774: cliente VIP1: FIN Evaluacion
19.0774: cliente VIP1: No necesita Mantenimiento, procede a Chapa y Pintura
19.0774: cliente VIP1: INICIO Chapa y Pintura
19.0774: cliente VIP1: INICIO Pintura Completa
21.1957: cliente ITV0: Inicio Revisión ITV
22.8377: cliente ITV0: Cliente impaciente, abandona el sistema
36.898: cliente Lavado0: INICIO Lavado
38.6387: cliente VIP2: INICIO Evaluacion
41.995: cliente VIP3: INICIO Evaluacion
50.226: cliente ITV1: Inicio Revisión ITV
51.5872: cliente VIP2: FIN Evaluacion
51.5872: cliente VIP2: Necesita Mantenimiento
51.5872: cliente VIP2: INICIO EsperaPiezas
51.5872: cliente VIP2: Piezas disponibles en el taller.
51.9724: cliente ITV1: Fin Revisión ITV
51.9724: cliente ITV1: Fallo leve o sin fallos, cliente finaliza proceso ITV
```

Figura 4.15: Ejemplo Fichero de Trazas

4.4.2. Fichero de Interfaz Gráfica - app.R

La apariencia de nuestro sistema como ya menciono en el apartado 1.3.1 se ha desarrollado mediante el paquete **shiny** [22] el cual nos proporciona una interfaz gráfica moldeable a los gustos del usuario en el que representar la información que queramos. Además es importante remarcar que shiny nos permite visualizar la interfaz gráfica de la aplicación a través de una dirección HTTP local, lo cuál también es una ventaja.

En la imagen 4.16 vemos una primera portada en la que se diferencian dos paneles de elección a la izquierda y a la derecha más arriba dos pestañas entre las que elegir también, una "Resumen y Estadísticas" para observar distintas métricas tomadas de la simulación y la otra "Gráficos" en la que el usuario puede ver distintas representaciones de la información gráficos.

TALLER de MANTENIMIENTO de VEHÍCULOS

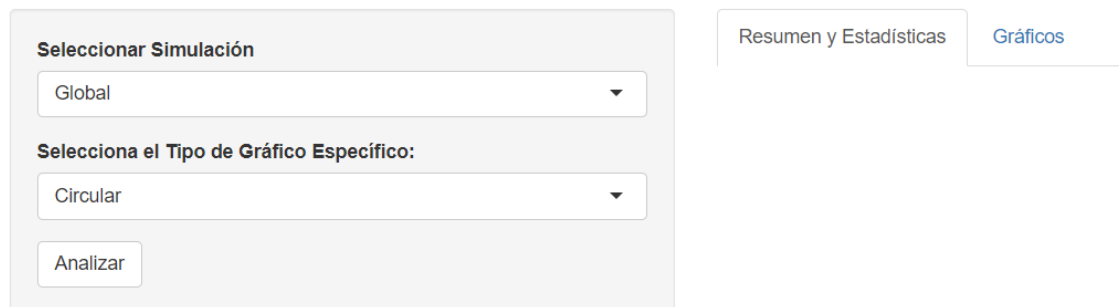


Figura 4.16: Portada Interfaz Gráfica de Shiny

Fijándonos de forma más atenta en los paneles de elección de la izquierda concretamente en el que se encuentra más arriba, *Seleccionar Simulación* se puede elegir entre si observar la información de una muestra o simulación concreta de las 10 réplicas implementadas, o por el contrario la información del computo *Global* de todas las simulaciones. Esta opción es válida tanto para la pestaña "Resumen y Estadísticas" como para la de "Gráficos" y se representa en la siguiente figura 4.17:

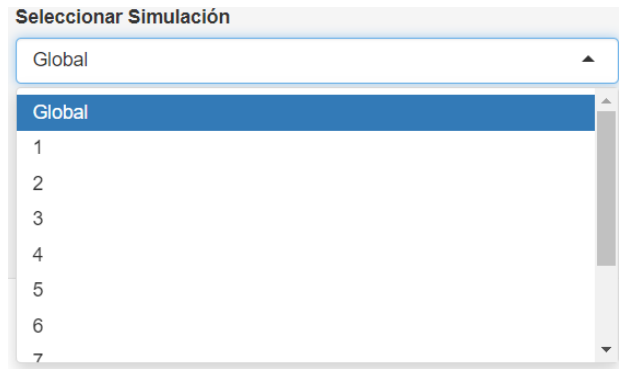
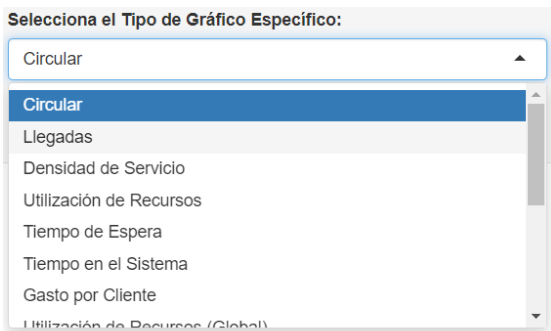


Figura 4.17: Panel *Seleccionar Simulación*

Además, en el panel inferior se puede leer *Seleccionar tipo de Gráfico Específico*, esta opción es únicamente válida para la pestaña "Gráficos" que como ya hemos mencionado sirve para mostrar distintos tipos de representaciones de la información. Es importante remarcar que tanto por riqueza de datos como por simplicidad de implementación las opciones en las que al final pone (*Global*) son únicamente seleccionables para la simulación *Global*. En la imagen 4.18a podemos ver las opciones de las representaciones a elegir y en la 4.18b una continuación de las opciones disponibles en este panel.



(a) Opciones Gráficos 1



(b) Opciones Gráficos 2

Figura 4.18: Opciones panel *Seleccionar tipo de Gráfico Específico*

Capítulo 5

Análisis de Datos

En esta parte se discutirá la retroalimentación obtenida en cada etapa del prototipado mediante el análisis de los datos extraídos de la simulación, esto nos permite enfocarnos en los aspectos a optimizar buscando la mejora del modelo hasta alcanzar una representación óptima del mismo.

5.1. Características de la Simulación

Es fundamental definir y presentar las características de la simulación para entender bien el contexto operativo de la misma. A continuación, se describen las configuraciones utilizadas en nuestro estudio:

Tiempo de Simulación:

La simulación se ejecuta durante un periodo de un mes suponiendo que el taller abriese todos los días 12 horas, es decir, 21.600 minutos. Esto se hace suponiendo que el establecimiento abriese de forma continuada y sin interrupciones. También se podrían ver como 15 días de forma continuada trabajando.

Número de Réplicas

Para que se pueda hacer un análisis de los datos y que exista una validez estadística en los resultados se realizan 10 réplicas independientes de la simulación, cada una con distinta semilla para garantizar que las ejecuciones son diferentes. Cabe remarcar que aunque las réplicas sean independientes y diferentes entre sí se hacen bajo las mismas condiciones. Es decir, si cada réplica la enfocásemos como si fuese un mes, podríamos decir que siempre simulamos un mes de Enero (por ejemplo) distinto.

Este número de réplicas es suficiente para reducir la varianza de las estimaciones y obtener intervalos de confianza precisos para las métricas clave del sistema, como tiempos de espera y utilización de recursos.

Ciclo de Simulación

El ciclo de simulación se define de forma continua ininterrumpida, es decir, en todo momento se capturan los resultados que se generan. Este nivel de granularidad permite capturar de forma mucho más detallada y precisa las variables de estado del sistema, como los tiempos de servicio y de espera, estados de las colas,

Tamaño de la Salida

Cada una de estas réplicas genera un archivo de registro (log file) que contiene las trazas de ejecución de los distintos clientes a lo largo de la simulación, los mensajes los he añadido de forma manual y sirven como registro para realizar un seguimiento de cada cliente. Aparte se generan una serie de archivos csv por cada simulación en los que se guarda la información más relevante de cada réplica como se detalla en el apartado 4.4.1. Además, todos estos ficheros csv también se agrupan para conseguir una información de carácter global que nos permita hacer el análisis con el conjunto de todos los datos.

5.1.1. Parámetros del Sistema

Los parámetros del sistema se configuran de la siguiente manera:

Recursos

- **Evaluación, Reparación, EsperaPiezas, Inspección, PinturaParcial, PinturaCompleta, Abolladura, Pulido, ITV:** Capacidad de 2 unidades, con capacidad infinita en la cola.

Tasas de llegadas

- **Cliente normal:** cada 25 minutos en promedio.
- **Cliente VIP:** cada 40 minutos en promedio.
- **Cliente Lavado:** cada 60 minutos en promedio.
- **Cliente ITV:** cada 30 minutos en promedio.

Tiempo de Servicio

- **Evaluación:** 20 minutos.
- **Mantenimiento:** 60 minutos.
- **EsperaPiezas (sin pedido):** 10 minutos.
- **EsperaPiezas (con pedido):** 120 minutos.
- **Reparación:** 60 minutos.
- **Inspección:** 15 minutos.

5.1. Características de la Simulación

■ Chapa y Pintura

- **PinturaPar**: 30 minutos.
- **PinturaComp**: 90 minutos.
- **Abolladura**: 35 minutos.
- **Pulido**: 25 minutos.

■ Lavado

- **Tunel de Lavado (Exterior)**: 25 minutos.
- **Lavado con Pistola (Exterior)**: 30 minutos.
- **Completo (Exterior + Interior)**: 50 minutos.

■ Revisión ITV: 20 minutos.

Ahora vamos a analizar y visualizar los datos de ambos prototipos bajo las mismas condiciones iniciales. En ambos casos las celdas señaladas mediante líneas discontinuas se refieren a los datos incluidos únicamente en el Prototipo 2.

En la siguiente tabla 5.1 podemos observar las condiciones asociadas a los clientes, los clientes VIP parten de un precio base de 50 euros porque se supone que pagan una tasa mensual de 50 euros, las tasas de llegadas de cada cliente se definen mediante la distribución exponencial y las prioridades están referenciadas en orden creciente, a mayor valor en el campo "Prioridad" mayor prioridad tendrá en la cola para ser atendido. Se presupone por tanto que los Clientes ITV al necesitar un servicio más rápido tengan mayor prioridad que los clientes normales.

Tipo de Cliente	Precio Base (EUR)	Tasa de Llegada(min)	Prioridad
Cliente Normal	0	Exponencial(25)	2
Cliente VIP	50	Exponencial(40)	4
Cliente Lavado	0	Exponencial(60)	1
Cliente ITV	0	Exponencial(60)	3

Cuadro 5.1: Condiciones Iniciales Clientes

En la tabla 5.2 vemos representadas las características que envuelven a las entidades de servicio, la tasa de probabilidad se refiere al porcentaje de que ocurra cierto suceso, en esta misma columna mediante un * se señala que esto se refiere a los clientes y clientes VIP, ya que son los únicos que pasan por dicho servicio, además en el servicio de *Inspección* la tasa de probabilidad se refiere a que suceda dicho evento una vez el cliente ya forma parte de la trayectoria de *Mantenimiento*, no a la probabilidad de que un cliente pase dicha inspección.

Además, la columna "Capacidad" se refiere a la capacidad del servidor, la capacidad de la cola no se ha añadido pero en esta primera ejecución se supone a infinita. Las siguientes celdas indican el tiempo que se tarda en desarrollar una

Capítulo 5. Análisis de Datos

tarea concreta y el coste económico que le supone al cliente dicho servicio. Toda esta información mencionada se ve reflejada en la siguiente tabla 5.2.

Servicio	Tasa de Probabilidad (%)	Capacidad	Tiempo (min)	Costo (EUR)
Evaluación	100*	2	20	10
Mantenimiento	50			
EsperaPiezas (sin pedido)	65	2	10	80
EsperaPiezas (con pedido)	35	2	120	120
Reparación	100	2	60	180
Inspección*	100	2	15	20
Chapa y Pintura	35			
PinturaPar	40	2	30	200
PinturaComp	25	2	90	400
Abolladura	35	2	35	250
Pulido	30	2	25	110
Lavado	40	2	10	30
Tunel de Lavado (Exterior)	50		10	15
Lavado con Pistola (Exterior)	50		20	20
Completo (Exterior + Interior)	50		50	60
Revisión ITV	100	2	20	75

Cuadro 5.2: Condiciones Iniciales Entidades

5.2. Datos Estadísticos

A través de los csv generados en los que se recaba los datos de simulación más relevantes analizaremos mediante cálculos que nos permitan obtener ciertas conclusiones de nuestro modelo. Se representa información sobre el tiempo en el sistema, de servicio y de cola, al igual que sobre el coste de cada tipo de sistema. Para cada uno de estos aspectos se obtendrán respectivamente sus valores "Mínimo", "Máximo", "Promedio" así como la "Varianza" y la "Mediana".

Prototipo 1

Vamos a analizar los datos expuestos en la tabla 5.1. De la primera columna podemos obtener una idea de las llegadas por muestra, aproximadamente unas 800 para Clientes normales, unas 500 para Clientes VIP y presumiblemente más de 300 llegadas de Clientes Lavado.

1. Tiempos en el Sistema y de Servicio - [5.1]

- **Cliente:** Con respecto a los valores mínimo y máximo del "Tiempo en el Sistema" se puede ver una alta variabilidad, sin embargo, pueden ser datos de casos aislados, la Varianza es la que nos indica verdaderamente esto y sugiere que existen casos extremos de demora significativa. Por otra parte al comparar estos datos con los referentes al tiempo de servicio podemos ver que en el segundo existe una mayor estabilidad, lo cuál nos puede indicar que las demoras se deben más a las colas y tiempos de espera.

- **Cliente Lavado:** Este tipo de clientes poseen un tiempo en el sistema mucho menor a los clientes normales, esto se debe a la naturaleza más simple y rápida del lavado. Además el tiempo de servicio es mucho menor y estable, lo que concuerda totalmente con el servicio de *Lavado* que carece de opciones.
- **Cliente VIP:** En los datos referentes a estos clientes se puede apreciar que tanto el tiempo de actividad máximo en el sistema como el promedio se reducen considerablemente con respecto al cliente "normal", sin embargo al comparar estos datos con los tiempos de servicio se puede apreciar un factor muy interesante y es que los tiempos de servicio entre los tipos de cliente es muy parecido ya que no existe preferencia en el servicio, lo que varía de forma sustancial es la prioridad de atención en la cola lo que provoca que los tiempos de servicio casi no varíen, pero el tiempo en el sistema sí.

Toda esta información relativa a los tiempos en el sistema y tiempos de servicio de cada tipo de cliente la vemos reflejada en la siguiente tabla 5.1:

tipo_cliente	Total Llegadas	Tiempo en el Sistema Mínimo (min)	Tiempo en el Sistema Máximo (min)	Tiempo en el Sistema Promedio (min)	Mediana Tiempo en el Sistema (min)	Tiempo en el Sistema (min ²) Varianza	Tiempo de Servicio Mínimo (min)	Tiempo de Servicio Máximo (min)	Tiempo de Servicio Promedio (min)	Mediana Tiempo de Servicio (min)	Tiempo de Servicio (min ²) Varianza
Cliente	8470	0.06	4702.41	393.04	177.27	265437.03	0.05	1256.38	110.70	86.71	9004.26
Cliente Lavado	3633	0.00	194.27	17.01	11.79	299.39	0.00	194.27	15.30	10.28	254.71
Cliente VIP	5517	0.00	965.46	161.71	126.65	17692.46	0.00	798.59	110.47	88.13	8481.16

Figura 5.1: Resultados Estadísticos bajo Condiciones Iniciales

2. Tiempos de Cola - [5.2a]

- **Cliente:** En lo referente a los tiempos de cola en los clientes se observa que los mismos son muy elevados, con un promedio de espera de más de 200 min, o lo que es lo mismo mayor a 3 horas. Esto nos indica que existen cuellos de botella importantes en el sistema, probablemente debido a una cantidad de recursos insuficientes.
- **Cliente Lavado:** En este caso los tiempos de cola son casi insignificantes lo cual refleja que la entidad de *Lavado* tiene buena capacidad para la demanda que enfrenta.
- **Cliente VIP:** Con respecto a los clientes normales los VIP mejoran bastante, aún así observamos un promedio aproximado de 50 minutos lo que sugiere que puedan existir picos de demanda.

En la tabla de la izquierda (5.2a) vemos representada los datos que acabamos de analizar respectivos a los tiempos de cola de cada tipo de cliente.

Capítulo 5. Análisis de Datos

Tiempo de Cola Mínimo (min)	Tiempo de Cola Máximo (min)	Tiempo de Cola Promedio (min)	Mediana Tiempo en Cola (min)	Tiempo de Cola (min^2) Varianza	Costo Mínimo (EUR)	Costo Máximo (EUR)	Costo Medio (EUR)	Mediana Costo (EUR)	Varianza Costo (EUR^2)
0.00	4561.40	282.34	51.54	237526.50	10.00	1350.00	325.37	290.00	53806.61
0.00	87.97	1.71	0.00	43.71	30.00	30.00	30.00	30.00	0.00
0.00	632.89	51.24	16.68	6220.73	50.00	1400.00	378.43	340.00	53458.93

(a) Resultado Colas Cond. Inic. 1

(b) Resultado Coste Cond. Inic. 1

Figura 5.2: Resumen Estadístico Cola y Coste Cond. Inic. 1

En lo referente a la tabla 5.2b situada a la derecha encontramos los datos y métricas relativos a cada tipo de cliente para los gastos de los mismos.

3. Costos - [5.2b]

- **Cliente:** La gran variabilidad representada mediante la varianza refleja la diversidad de trayectorias y servicios que puede seguir un cliente y que implican gastos totalmente diferentes. El coste promedio es en general más costoso debido a la combinación de múltiples servicios.
- **Cliente Lavado:** Con un costo promedio de 30 euros y sin variabilidad en el precio de este servicio, es el más barato y estandarizado del modelo.
- **Cliente VIP:** El coste que asume el cliente VIP es más elevado que los del resto, aparte de por su condición de "VIP" por la variabilidad en las combinaciones del servicio como en los clientes normales.

Prototipo 2

En la información referente a la tabla 5.3 y comparando los resultados con la misma tabla del Prototipo 1 (5.1) podemos ver que la afluencia de todos los tipos de clientes es la misma, esto se debe a que la capacidad de las colas es infinita para todas las entidades por tanto si durante el mismo periodo de tiempo (en este caso un mes por cada muestra) se genera un nuevo tipo de cliente, el resto de llegadas de clientes no se verán afectadas si continúan con la tasa de generación previa a la creación del nuevo tipo de cliente.

1. Tiempos en el Sistema y de Servicio - [5.3]

- **Cliente:** Con respecto al prototipo anterior es importante remarcar que el cliente pierde un rango en la prioridad de atención en cola ya que se presupone como ya he comentado con anterioridad que el cliente ITV tiene una mayor necesidad de ser atendido cuánto antes. Esto repercute directamente en los tiempos en el sistema los cuales aumentan respecto al otro prototipo.

- **Cliente ITV:** El nuevo tipo de cliente tiene un tiempo en el sistema bastante estable y controlado, la variabilidad que existe en cuánto al mismo es si se le encuentra un fallo grave y es redirigido a la trayectoria de "Mantenimiento" o si por el contrario únicamente se detectan fallos leves o la ausencia de fallos en cuyo caso seguirá su trayectoria de la ITV y no variara su coste y tiempo de servicio en estos casos.
- **Cliente Lavado:** Con respecto a los clientes de Lavado al incluir distintas posibilidades en las trayectorias que pueden seguir dichos clientes y tener entre todos los clientes la menor prioridad de atención en cola, sufren un aumento considerable tanto en el tiempo en el sistema como en el de servicio.
- **Cliente VIP:** El cliente VIP se mantienen constantes en lo respectivo a los mismos datos del prototipo 1.

Toda esta información relativa a los tiempos en el sistema y tiempos de servicio de cada tipo de cliente para el prototipo 2 la vemos reflejada en la siguiente tabla 5.3:

tipo_cliente	Total Llegadas	Tiempo en el Sistema Mínimo (min)	Tiempo en el Sistema Máximo (min)	Tiempo en el Sistema Promedio (min)	Mediana Tiempo en el Sistema (min)	Tiempo en el Sistema (min ²) Varianza	Tiempo de Servicio Mínimo (min)	Tiempo de Servicio Máximo (min)	Tiempo de Servicio Promedio (min)	Mediana Tiempo de Servicio (min)	Tiempo de Servicio (min ²) Varianza
Cliente	8449	0.06	5344.20	674.45	250.02	725930.75	0.00	1004.70	118.05	92.36	10123.89
Cliente ITV	3609	0.00	1747.60	63.47	12.83	22118.75	0.00	594.34	31.53	12.55	3276.32
Cliente Lavado	3526	0.00	790.16	99.63	52.70	15128.70	0.00	404.41	36.15	18.59	2211.45
Cliente VIP	5409	0.08	998.82	181.97	147.36	20638.66	0.06	901.61	119.19	93.53	9986.41

Figura 5.3: Resultados Estadísticos bajo Condiciones Iniciales 2

2. Tiempos de Cola - [5.4a]

- **Cliente:** En lo referente al tiempo en cola este también se ve incrementado con respecto al primer prototipo, como ya he mencionado debido a su bajada en la prioridad de la cola y a una mayor carga del sistema ante nuevas llegadas antes no valoradas.
- **Cliente ITV:** Este tipo de clientes mantienen un tiempo en cola mucho menor ya que gozan de una buena prioridad y salvo que el vehículo posea un error grave y sea redirigido al "Mantenimiento" su tiempo de cola sera estable.
- **Cliente Lavado:** El mismo es el más afectado en este segundo prototipo debido a la adición de distintas posibilidades de lavado, así como del cliente ITV. Todos sus tiempos en cola aumentan los más reseñables el tiempo promedio y la varianza.
- **Cliente VIP:** Los clientes VIP se mantienen estables con una ligera mejora en lo respectivo al primer prototipo.

Capítulo 5. Análisis de Datos

En la tabla de la izquierda (5.4a) vemos representada los datos que acabamos de analizar respectivos a los tiempos de cola de cada tipo de cliente.

Tiempo de Cola Mínimo (min)	Tiempo de Cola Máximo (min)	Tiempo de Cola Promedio (min)	Mediana Tiempo en Cola (min)	Tiempo de Cola (min^2) Varianza	Costo Mínimo (EUR)	Costo Máximo (EUR)	Costo Medio (EUR)	Mediana Costo (EUR)	Varianza Costo (EUR^2)
0.00	5232.10	556.40	101.23	679752.73	10.00	1445.00	317.18	290.00	55031.37
0.00	1570.47	31.94	0.00	13301.58	75.00	615.00	145.33	75.00	17032.53
0.00	776.16	63.48	5.37	12891.32	15.00	50.00	33.48	20.00	266.74
0.00	607.90	62.78	27.70	7106.30	50.00	1480.00	372.99	340.00	53704.43

(a) Resultado Colas Cond. Inic. 2

(b) Resultado Coste Cond. Inic. 2

Figura 5.4: Resumen Estadístico Cola y Coste Cond. Inic. 2

En lo referente a la tabla 5.4b situada a la derecha encontramos los datos y métricas relativos al prototipo 2 para cada tipo de cliente y los gastos de los mismos.

3. Costos - [5.4b]

- **Cliente:** Los datos se mantienen constantes con respecto al prototipo 1 y la varianza de coste sigue indicando la diversidad de los servicios disponibles.
- **Cliente Lavado:** En este la varianza ha pasado de un valor nulo ya que solo existía un tipo de opción de lavado a tener distintas opciones, esto también ha favorecido a que el precio mínimo que puede gastarse el cliente para un lavado disminuya.
- **Cliente ITV:** Los costes para los clientes ITV no son elevados, además estos son predecibles y estandarizados. Aunque en algunos casos su coste pueda ser más elevado su coste promedio es aproximadamente la mitad del coste de un Cliente Normal o VIP.
- **Cliente VIP:** Seguimos observando aquí la diferencia de gasto entre el Cliente Normal y el VIP al ser el de este último algo más elevado.

5.3. Visualización de Gráficas

Los datos obtenidos del apartado anterior así como el resto de información registrada en los archivos csv se puede representar y visualizar mediante distintas gráficas. En el siguiente apartado podremos estudiar con mayor nivel de detalle y de forma más visual los resultados obtenidos de nuestra simulación. Todas las gráficas pertenecen al análisis realizado únicamente sobre el **Prototipo 2**, debido a que es el más completo de ambos y del que podemos obtener una mayor cantidad de información, además considero que es más apropiado hacerlo

de esta forma para no ser repetitivo en la visualización de las gráficas de ambos prototipos ya que no existen grandes cambios en el flujo de trabajo general de uno al otro ya que se a trabajado mas en la ampliación y adición de tareas respecto

5.3.1. Distribución en el Tiempo del Uso de Recursos

Se analiza la distribución del uso de los recursos en el tiempo, es decir, del tiempo total de simulación, el porcentaje de tiempo que usa cada recurso. Los porcentajes indican por tanto la proporción de tiempo de uso de cada recurso y esto se representa mediante un diagrama circular o sectorial, en el que cada Entidad o Servicio viene representado mediante un color y su correspondiente porcentaje usado del sistema impreso encima de su sector correspondiente.

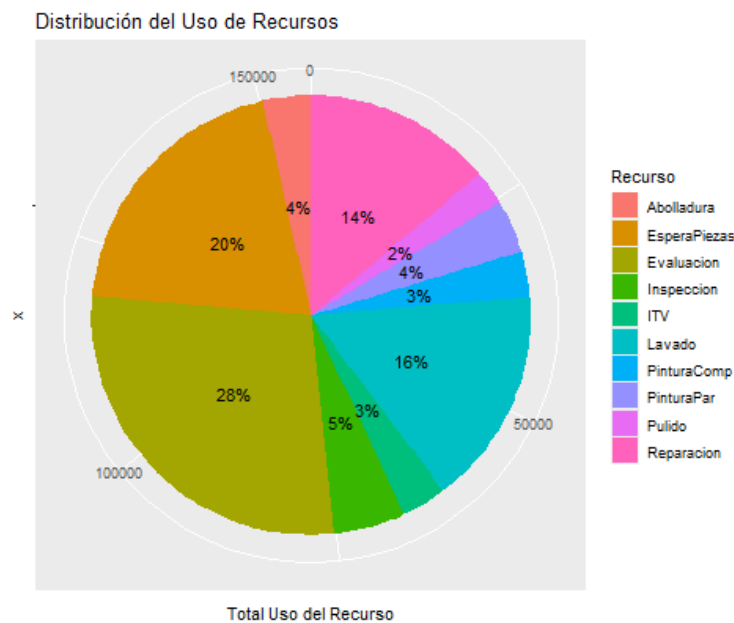


Figura 5.5: Gráfico Circular Prototipo 1

En la imagen 5.5 podemos observar por ejemplo como la *Evaluación* conforma el 28% del tiempo de simulación mientras que la *Espera de Piezas* supone el 20% de este tiempo. Ambas componen la parte más significativa del tiempo de uso del sistema, lo cual es coherente si se piensa que todos los Clientes y Clientes VIP siempre pasan por la etapa de *Evaluación* y muchos de estos clientes son redirigidos a la fase de *Mantenimiento* en la cuál la *Espera de Piezas* puede demorarse mucho si las mismas no están en el taller. Además, podemos concluir que ambas son las etapas más propensas a ser cuellos de botella para el sistema y que por tanto ralenticen la eficiencia del mismo.

Por otra parte entidades como el *Lavado* (16%) o la *Reparación* (14%) también tienen un peso considerable en el sistema, mientras que todos los servicios relativos a la trayectoria de Chapa y Pintura (*Pintura Parcial*, *Pintura Completa*,

Pulido y Abolladura) se encuentran entre el (2%) y el (4%) de uso, así como la *ITV* que conforma un 3% lo que significa que todos estos recursos emplean un menor tiempo en dar su servicio y/o tienen un menor tráfico de clientes y cuando se usan son más eficientes.

5.3.2. Distribución de Llegadas

A continuación podemos ver una representación mediante un histograma del número total de llegadas de los clientes a lo largo del tiempo de simulación en intervalos de minutos.

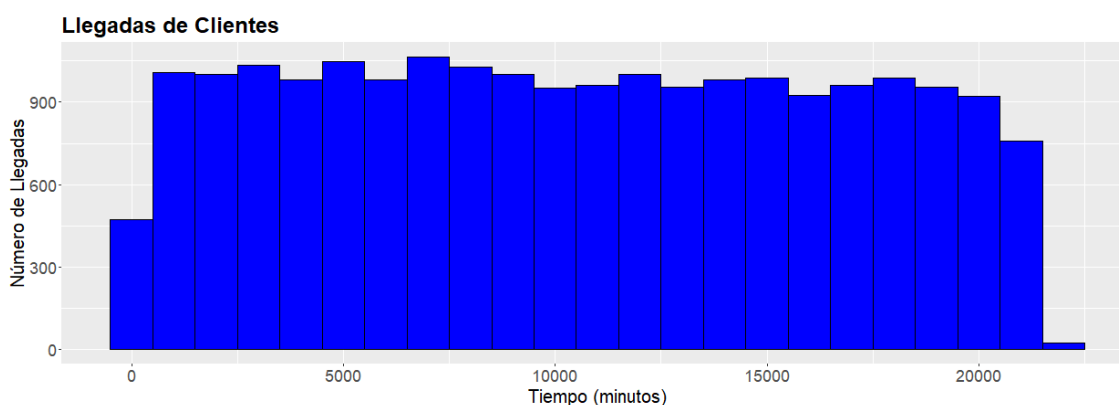


Figura 5.6: Histograma de Llegadas

En la figura 5.6 se puede observar que las llegadas siguen una distribución que parece relativamente uniforme a lo largo del tiempo, con ciertos picos y valles menores. Esto sugiere que la distribución de llegadas es constante, lo cual encaja con un proceso de Poisson. En un proceso de Poisson, los intervalos entre llegadas son aleatorios y siguen una distribución exponencial, resultando en una tasa constante de llegadas por unidad de tiempo.

Este análisis nos permite llegar a la conclusión de que necesitamos un sistema que soporte una carga constante de trabajo, lo que favorece la planificación de los recursos. Es importante considerar que la demanda homogénea de servicios puede influir en la distribución de llegadas observada. En otros sistemas, estos esfuerzos podrían estar distribuidos en el tiempo, según distintos factores, como la variabilidad en la demanda y los diferentes tipos de servicios requeridos.

5.3.3. Densidad Tiempo de Servicio

Analizamos ahora la distribución del tiempo de servicio a los clientes. La gráfica 5.7 que se ve a continuación muestra un alta densidad de tiempos de servicio bajos, sin embargo, se aprecia un gran pico lo que implica que aunque la mayoría de los servicios son significativamente rápidos existe una minoría que toman mucho más tiempo. La alta densidad de tiempos de servicio bajos es un buen indicador de eficiencia para la mayoría de los casos, pero el pico alto sugiere

que hay excepciones que necesitan ser abordadas para evitar tiempos de espera prolongados e ineficiencias en el sistema.

Además, esta forma tan característica, donde la densidad cae en picado, es típica de la distribución exponencial y esta forma se respalda sabiendo que los tiempos de servicio en el sistema siguen una distribución exponencial.

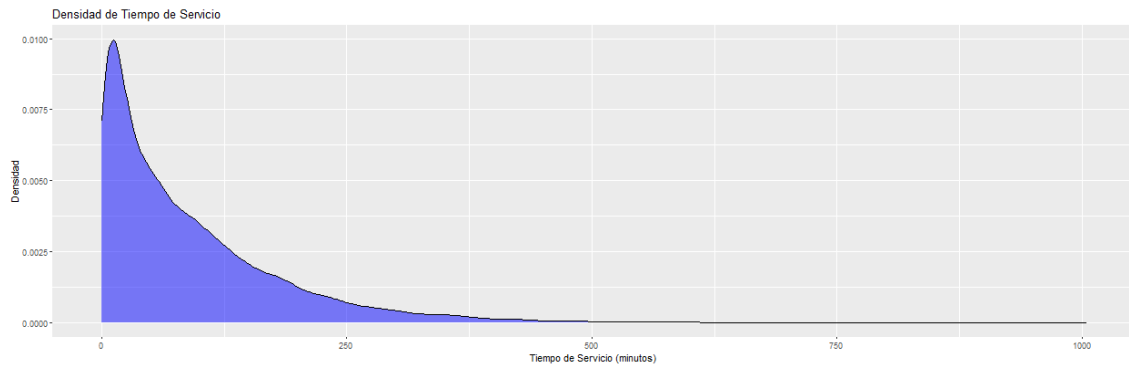


Figura 5.7: Densidad Tiempo de Servicio

5.3.4. Análisis de Gastos

Ahora se estudian los distintos niveles de gastos según el tipo de cliente. Cada una de las cajas plasma el rango intercuartílico (IQR) del gasto, la línea del interior de la caja indica la mediana del gasto, mientras que los bigotes (o líneas perpendiculares a la caja) se extienden hasta un 1,5 veces el IQR viendo como los puntos que se dibujan fuera de ese rango son considerados valores atípicos.

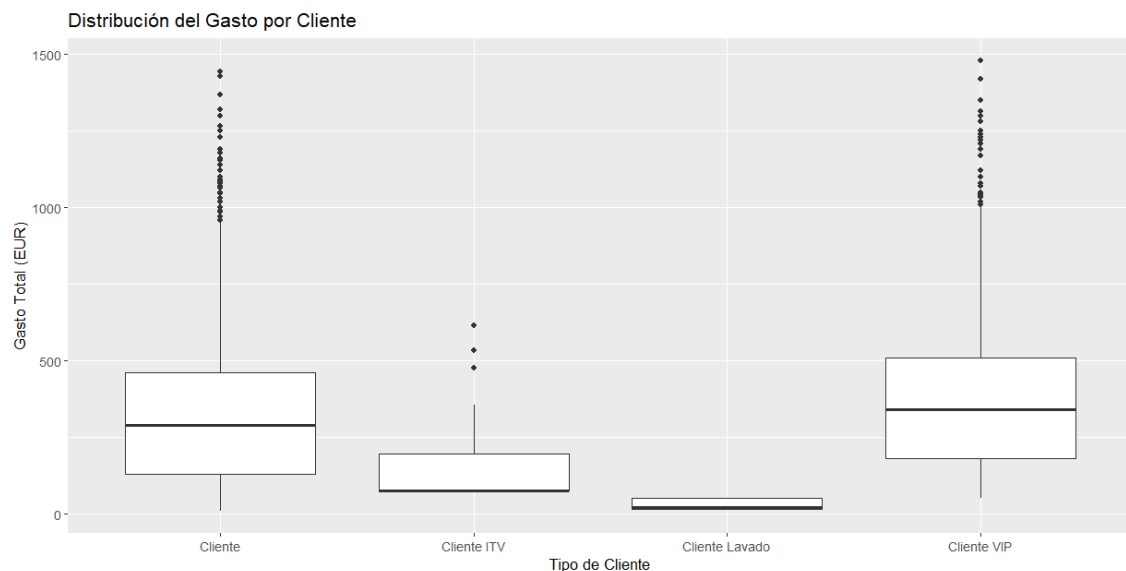


Figura 5.8: Prototipo 1: Distribución de Gastos según el tipo de cliente

En la siguiente figura 5.8 vemos representado mediante un diagrama de cajas

y bigotes esta información sobre los tipos de clientes "Cliente", "Cliente ITV", "Cliente Lavado" y "Cliente VIP" así que los datos reflejados en la imagen son referentes a ellos.

Con respecto al **Cliente** podemos observar que su mediana es algo superior a los 250 EUR, mientras que el IQR está aproximadamente entre 150 y 500 EUR, lo que sugiere que el 50% central de estos clientes gastan entre estas cantidades. La longitud del bigote superior es mucho más larga que la del bigote inferior, lo que indica que hay una mayor dispersión de datos en los gastos más altos. Por otra parte existen diversos puntos por encima de los 1000 EUR hasta los casi 1500 EUR lo que nos indica que aunque la mayoría de los clientes gastan dentro de un rango moderado algunos de ellos pueden tener unos gastos más elevados debido al uso de servicios adicionales o reparaciones más costosas.

En lo que se refiere al **Cliente ITV** podemos apreciar una mediana muy baja, cercana a los 75 EUR, lo cual es el gasto mínimo de estos clientes, esto indica que la mayoría de ellos paga lo mínimo. El rango intercuartílico es más compacto para estos clientes, entre los 100 y los 200 EUR, lo que sugiere también una menor variabilidad de gasto comparado con los clientes generales. También con respecto a los valores atípicos estos son poco numerosos y no tan extremos lo que refuerza la idea de que sus gastos son más uniformes y moderados.

El **Cliente Lavado**, aunque parezca que no se está representando nada podemos sacar varias conclusiones de este tipo de cliente. La mediana, el rango intercuartílico, así como la ausencia de valores atípicos nos indican que este tipo de clientes no incurren en gastos adicionales aparte del costo base del servicio de lavado, se observa que la caja es casi inexistente lo que nos sugiere que no existe variabilidad en el gasto de estos clientes y que son predecibles. Además, con respecto al Prototipo 1 su representación en este mismo gráfico es una única línea ya que siempre usaba el mismo servicio con el mismo coste al no existir más posibilidades.

Por último sobre el **Cliente VIP** podemos observar lo esperado, tanto su mediana como su rango intercuartílico es algo mayor que los mismos datos del Cliente "normal", por otra parte sus valores atípicos parecen también algo superiores a los del Cliente sin embargo existen menos clientes VIP con valores atípicos, lo que quiere decir que aunque los costes puedan llegar a ser más elevados para la mayoría existen menos Clientes VIP que se salgan del marco previsto de gasto en los servicios del taller.

5.3.5. Uso de Recursos

En este apartado analizaremos la información referente al uso de los recursos pero no con respecto al tiempo que se gasta en ellos como se analiza en el punto 5.3.1 de esta misma sección si no con respecto al uso de la capacidad de cada recurso. Como podemos observar en este gráfico de barras (5.9) los recursos más utilizados con respecto a su capacidad son los de **EsperaPiezas**, **Reparación**, **Lavado** y **Evaluación** con un 95%, 80%, 68% y 63% respectivamente. Esto nos sugiere que estos recursos mencionados son críticos y podrían representar

cuellos de botella en nuestro sistema.

Los intervalos de confianza mostrados en el gráfico 5.9 indican la variabilidad de la utilización de cada recurso. Un intervalo de confianza más amplio sugiere una mayor variabilidad en el uso del recurso, mientras que un intervalo de confianza corto nos indica la poca variabilidad o estabilidad de ese recurso.

Aplicando esto a nuestra representación podemos observar que el recurso **EsperaPiezas** aparte de ser el que más utilización emplea es el que posee el rango más alto de variabilidad sugiriendo por tanto la necesidad de mejora en cuanto a la gestión de dicha entidad. Por el contrario recursos como **Inspección**, **Lavado** y/o **Pulido** entre otros tienen intervalos de confianza muy cortos, lo que implica que son estables y poco variables.

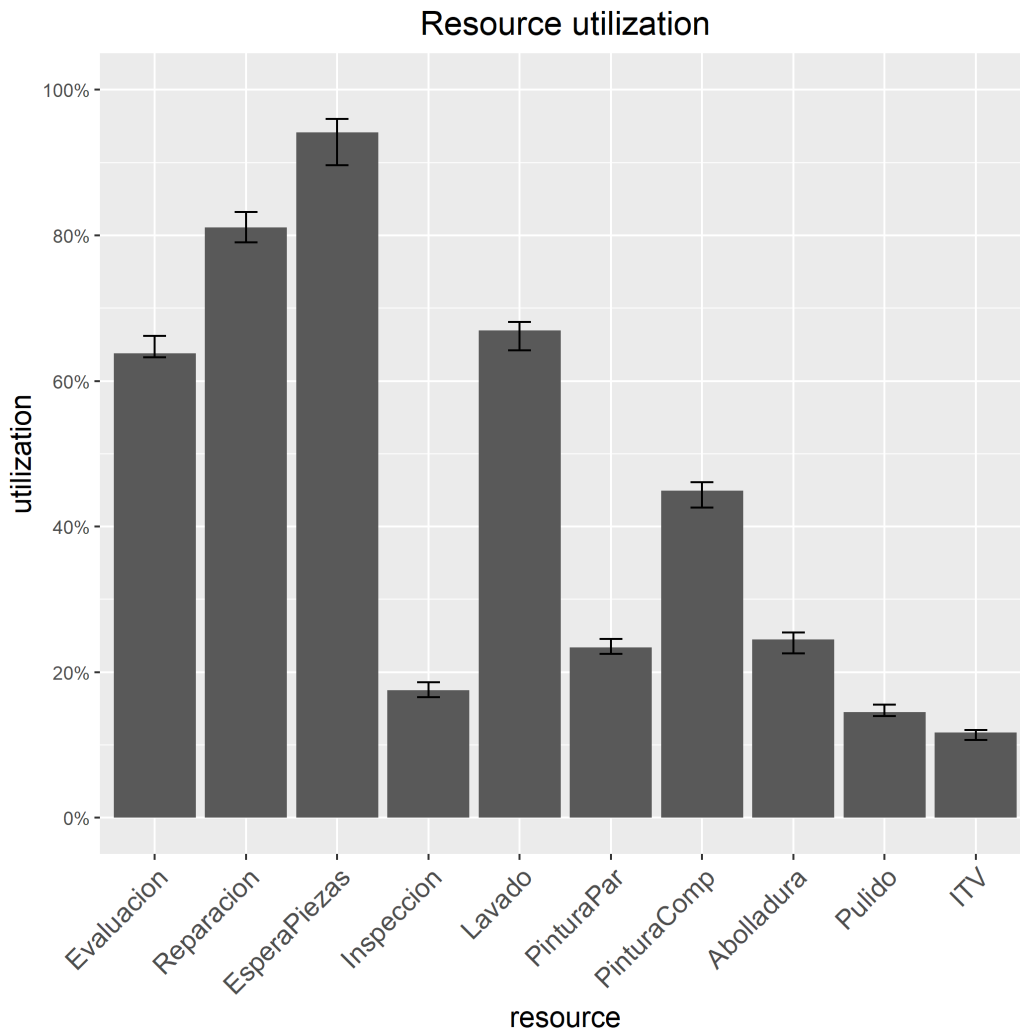


Figura 5.9: Porcentaje de uso de cada recurso

Sin embargo, se puede obtener una conclusión más allá de eso y es que los últimos recursos mencionados son apenas usados entre un 20 % y un 40 %, es

decir, no llegan ni a la mitad del uso del recurso, lo que puede implicar que sean entidades que están siendo infrautilizadas con respecto a su capacidad y que se podrían reajustar para que su uso fuese más proporcional a su capacidad

5.3.6. Tiempo de Actividad

En este apartado analizaremos la distribución de los tiempos de actividad por encima y por debajo de la mediana respectivamente para todos los clientes del sistema.

En la primera figura representada 5.10 podemos observar como la gráfica exhibe una distribución que decae de forma exponencial, en la que la mayoría de los tiempos de actividad o servicio por encima de la mediana están concentrados en el rango de 0 a 250 minutos, disminuyendo de manera progresiva hasta los 1250 minutos. Lo que nos confirma lo observado en el apartado 5.3.3, los casos en los que los clientes requieren tiempos de servicio extremadamente largos son casos raros. El pico inicial se encuentra en un rango de 0 a 250 minutos, es decir, la mayoría de clientes tiene una duración en el taller relativamente moderada. Después de los 250 minutos la frecuencia disminuye rápidamente, lo que nos indica que los tiempos de actividad superior a los 250 minutos son mucho menos comunes.

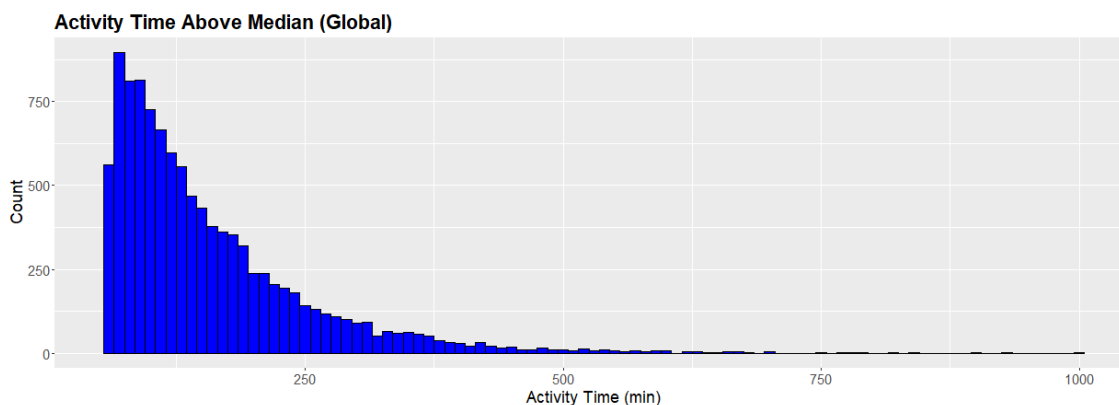


Figura 5.10: Tiempo de Actividad por encima de la Mediana

Por otra parte con respecto al tiempo de actividad por debajo de la mediana representados en la figura 5.11 se observa una distribución del tiempo entre los rangos de 0 a 60 minutos, con valores algo superiores en el rango de 10 a 20 minutos. Esto nos sugiere que los tiempos de actividad cortos son más comunes en comparación con los tiempos más largos. Estos valores mucho más estables nos indican que hay un conjunto mayoritario de tareas que se completan de forma efectiva y rápida, lo que favorece al funcionamiento del taller y a la mayor rotación de los clientes.

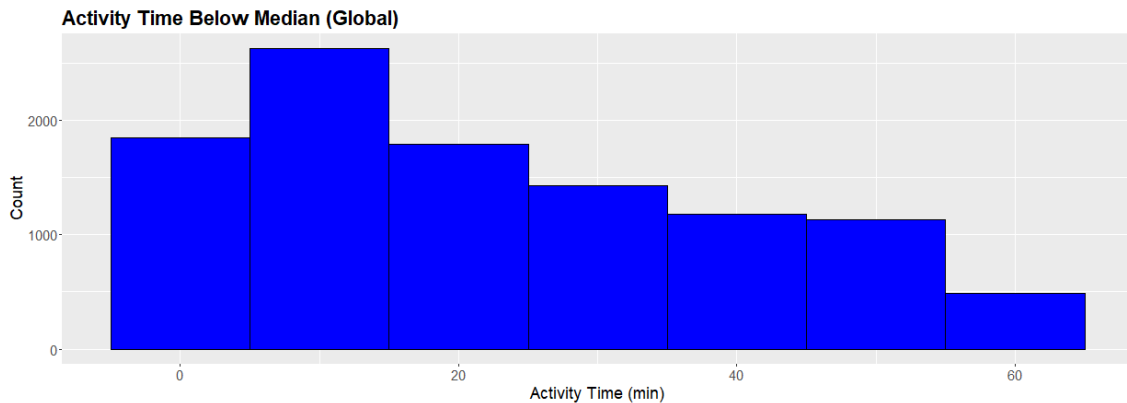


Figura 5.11: Tiempo de Actividad por debajo de la Mediana

5.3.7. Tiempo de Espera

El gráfico de la distribución del tiempo de espera por encima de la mediana muestra una clara tendencia hacia tiempos de espera más largos. La mayor parte de los clientes esperan entre 0 y 800 minutos, y la frecuencia disminuye a medida que aumenta el tiempo de espera. Esta distribución sugiere que, aunque un gran número de clientes tienen tiempos de espera bastante cortos, hay algunos clientes cuyos tiempos sugieren largas colas de espera. Estos resultados se ven representados en la gráfica inferior (5.12):

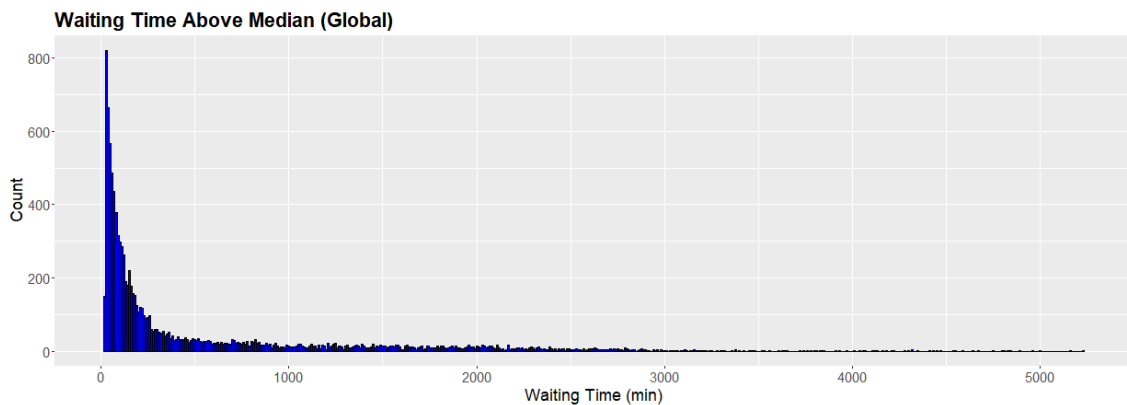


Figura 5.12: Tiempo de Espera por encima de la Mediana

Este comportamiento del gráfico puede explicarse por la variabilidad intrínseca de los tiempos de servicio y la limitación de capacidad de los recursos. Cuando la demanda es elevada, algunos clientes pueden sufrir grandes retrasos si se consumen los recursos disponibles. Esto se podría mitigar mediante distintas estrategias, como la asignación de recursos o mecanismos de priorización más eficaces o mediante el bloqueo de nuevos clientes al sistema ante situaciones de grandes colas.

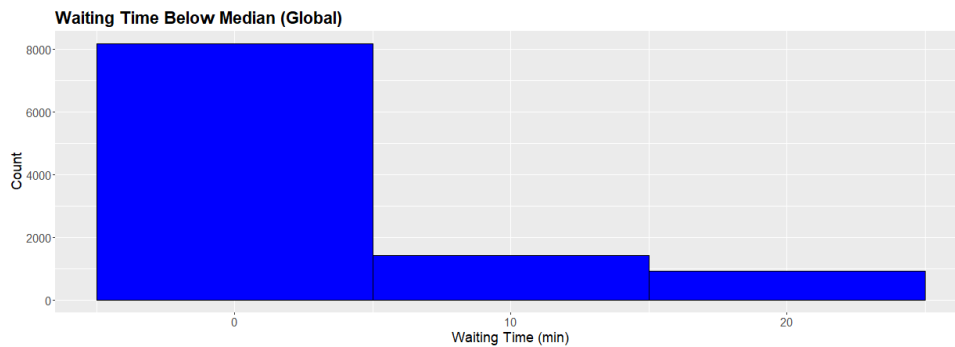


Figura 5.13: Tiempo de Espera por debajo de la Mediana

Sin embargo, el gráfico (5.13) en el que se representa la distribución del tiempo de espera por debajo de la mediana está muy compactado en su representación. La mayor parte del tiempo de espera se concentra en torno a valores muy bajos, con altas concentraciones de tiempo de espera en torno a casi cero. Esto indicaría que para el mayor número de clientes, el tiempo de espera es mínimo, lo que en términos de eficacia del sistema es un buen resultado.

Estos datos pueden explicarse por el hecho de que la capacidad es suficiente y eficiente a la hora de atender a los clientes en periodos de baja demanda. Cuando los recursos no están saturados se puede atender a los clientes casi de inmediato, lo que reduce drásticamente el tiempo de espera.

Capítulo 6

Conclusiones y trabajo futuro

Resumen de los resultados obtenidos en el trabajo, cumplimiento de los objetivos planteados, conclusiones obtenidas y análisis potencial de las líneas futuras del proyecto.

6.1. Evaluación de Objetivos del TFG

Se valorará el cumplimiento de los objetivos expuestos en la sección de Objetivos (1.1) al principio del documento.

El principal objetivo era la **creación de un modelo de simulación** para un taller de mantenimiento de vehículos. Este objetivo se ha cumplido ya que se ha llevado a cabo la implementación de dos prototipos de complejidad creciente que permiten observar y analizar el comportamiento del taller bajo diferentes condiciones.

Con respecto a los objetivos específicos, la elaboración del **Modelo Empírico** se llevó a cabo satisfactoriamente, representando el sistema mediante figuras y diagramas descriptivos. Este paso nos permitió definir una base sólida sobre la cual se desarrollaron los prototipos y se implementó el modelo.

En lo relativo al **Modelo Matemático y Computacional** se consiguieron desarrollar una serie de funciones basadas en distintas distribuciones probabilísticas para modelar: las llegadas de los clientes, el tiempo consumido por estos en las distintas etapas de simulación y las decisiones que toman dentro del sistema.

Para la parte de **Implementación** se definieron distintos escenarios para representar las múltiples condiciones que pueden suceder en un sistema y se ha logrado a través de dos prototipos en los que se ha logrado replicar de manera efectiva los procesos internos del taller, proporcionando una herramienta para la experimentación y el análisis en un entorno controlado.

Por último, la parte de **Análisis y Visualización de los Datos** nos ha permitido a través de distintas métricas y representaciones obtener conclusiones objetivas del modelo tanto para una optimización posterior si se quisiese, como para estu-

diar mediante medidas de rendimiento la efectividad del sistema, así como para un correcto entendimiento del funcionamiento interno del sistema.

6.2. Líneas Futuras

Dentro de lo conseguido en nuestro modelo de simulación siempre existen factores que podrían hacer del mismo un sistema más completo, dentro de esos factores se encuentran:

1. Mayor Complejidad del Sistema:

- **Proveedores y Logística:** Añadiendo diferentes proveedores que sigan distintas políticas de suministro de piezas, entrega y coste se conseguiría un sistema con más factores a analizar, diversos escenarios antes no contemplados que pondrían en valor la eficiencia de cada proveedor así como el coste logístico de cada operación. Todo esto promueve un sistema más cercano a la realidad.
- **Gestión de Personal:** Se podrían también integrar lógicas relacionadas con los empleados del taller, cada uno definido por su puesto de trabajo con diferentes tareas a llevar a cabo, habilidades y rangos de experiencia. Esto permitiría realizar combinaciones con empleados de distintos sectores en el taller para analizar la productividad de los mismos así como una mayor gestión y control del gasto con respecto a sus sueldos.
- **Simulación de Turnos:** Además actualmente en nuestro sistema se simula un periodo de un mes en tiempo continuo y sin paradas, se podrían implementar turnos de trabajo siendo estos intervalos fijos de tiempo con paradas en determinados puntos del tiempo para simular el periodo de noche en el que los talleres convencionales no suelen estar operativos. Concepto que también se podría combinar con la idea de implementar distintos tipos de puesto y personal, lo que enriquecería mucho el modelo.
- **Simulación de Situaciones de Emergencia:** Otra de las opciones a barajar pasa por el planteamiento de simular situaciones de parada repentina de ciertos procesos simultáneamente o no. Si se quisiese llevar el modelo a un plano real habría también que valorar todas las situaciones de emergencia posibles, incendio, inundación, corte en el suministro eléctrico.
- **Expansión del Taller:** Nuestro taller podría pasar por fases de expansión de instalaciones así como de maquinaria y funciones, todo esto supondría afrontar nuevos retos asociadas a dicha expansión.

2. Simulación Gráfica en Tiempo Real:

- **Uso de Herramientas Avanzadas:** El uso de herramientas como AnyLogic[23] o Tecnomatix orientadas a simulaciones en 3D sería muy nutritivo para el desarrollo del proyecto ya que nos permite además de

la distribución en el espacio de las máquinas y el personal en el taller, una observación del flujo de trabajo en tiempo real. Estas herramientas también disponen de la capacidad de cambiar en tiempo real los tiempos de servicio, capacidad de las colas, así como todos los factores que envuelven el modelo.

- **Interfaz de Usuario Mejorada:** Con respecto a la interfaz desarrollada por mi cuenta en shiny siempre se puede mejorar, desde desarrollar una interfaz más vistosa con distintas opciones hasta hacer diferentes representaciones para los datos o que se ubiquen de otra manera en las pestañas.

3. Distribuciones Estadísticas Avanzadas:

- **Mixturas de Distribuciones:** Dentro de la estadística como ya hemos tratado existen distintos tipos de distribuciones, sin embargo también existen combinaciones o mixturas de varias de estas distribuciones, factor que podría favorecer a la creación de un modelo más exacto que pudiese representar con mayor fidelidad el comportamiento de los clientes o el reparto de tiempo en los distintos procesos.
- **Pruebas de Bondad de Ajuste:** La bondad de ajuste pretende evaluar cuan bien se ajusta un modelo a un conjunto de observaciones. Existen diversas técnicas para llevar a cabo esta evaluación y las mismas buscan validar y mejorar el modelo mediante la comparación de los valores esperados a los valores reales obtenidos

4. Optimización del Sistema:

- **Algoritmos de Optimización:** En el mundo de la simulación existen muchas maneras de mejorar la eficiencia de los procesos, si se hubiese integrado alguna de estas técnicas probablemente hubiésemos descubierto puntos a mejorar que podrían haber significado por ejemplo un cambio en el diseño de alguno de los prototipos o de ambos.
- **Análisis de Sensibilidad:** Realizar análisis de sensibilidad para identificar los parámetros más críticos del sistema y cómo afectan al rendimiento general. Esto ayudará a focalizar los esfuerzos de mejora en las áreas que más lo necesitan.

6.3. Validación y Verificación

Para que nuestro proyecto este correctamente desarrollado y se ajuste a lo que realmente se busca se comprueban una serie de requisitos que veremos a continuación. La **validación** busca asegurar que se siga una lógica de simulación y que los parámetros utilizados sean representativos de la realidad.

- **Parámetros Realistas:** Los parámetros del modelo, como las tasas de llegada de clientes, los tiempos de servicio y los costes de cada servicio, son orientativos y no se ajustan a todas las realidades, sin embargo si puede encajar en según que modelos y se basan en observaciones y búsquedas

Capítulo 6. Conclusiones y trabajo futuro

realizadas sobre el ámbito de la reparación de automóviles, por tanto no son valores escogidos con aleatoriedad y tienen su fundamento teórico subyacente.

- **Consultas al Tutor:** La orientación que he recibido ha sido de una persona involucrada dentro del departamento de "Probabilidad y Estadística" y la cual ha realizado unos estudios en base a ello, además de apasionado por el mundo de la simulación. Todo ello me ha servido para tener una base coherente de los parámetros usados.
- **Comparación con Datos Reales:** Para que la validación fuese completa se debería comparar los resultados de la simulación con datos reales, sin embargo, debido a las limitaciones del proyecto esto no ha sido posible y se plantea como una línea futura como se menciona en el apartado anterior 6.2.

Por otra parte la **verificación**, se enfoca en asegurar que el modelo de simulación esté libre de errores y que los resultados producidos tengan sentido. Para conseguir esto, se usaron los siguientes métodos:

- **Revisión del Código:** Cada vez que se realizaba un cambio se comprobaba que no hubiese problemas de compilación ni "Warnings" y que los cambios realizados en la implementación tuviesen sentido más adelante en la recogida de datos y en la impresión del fichero de trazas.
- **Pruebas Secuenciales:** Como el código se ha elaborado de forma secuencial y a medida que avanzaban las semanas descubría nuevas funcionalidades que poder implementar, las nuevas modificaciones daban lugar a pruebas sobre las funciones para comprobar que funcionaban correctamente y devolvían lo que se esperaba de ellas.
- **Simulaciones de Prueba:** Se llevaron a cabo simulaciones de prueba tanto con distintos paquetes de RStudio como con escenarios más simples para verificar que el modelo produjese resultados lógicos y consistentes y así avanzar hacia implementaciones más complejas.
- **Documentación y Transparencia:** Toda la lógica del modelo, los supuestos y los parámetros han sido documentados. Esto aparte de facilitar la verificación también permite que otros investigadores o estudiantes reproduzcan y validen el estudio en un futuro supuesto.

En resumen, mediante los distintos pasos seguidos considero que el proyecto se ha ajustado correctamente a la lógica que se pretendía, los datos utilizados y el modelo por ende siguen una estructura coherente y realista de lo que podría ser un sistema real.

6.4. Evaluación del Proceso de Realización del TFG

El desarrollo de este Trabajo Fin de Grado ha sido un proceso enriquecedor y desafiante a partes iguales en el cuál he tenido que ir superando las distintas etapas planteadas mencionadas en la sección anterior 6.1.

En lo referente a la **planificación y gestión del proyecto** desde un primer momento se establecieron los objetivos a conseguir tanto a corto como a largo plazo y los recursos y materiales disponibles para mantener el trabajo en curso. A medida que las semanas fueron pasando mi tutor me iba orientando en los siguientes retos y tareas a abordar lo cuál me resultó de gran ayuda para ir cumpliendo los objetivos en términos de contenido de forma secuencial. Desde el primer momento las fechas y plazos planteados se han ido cumpliendo y no se ha tenido que realizar ningún cambio con respecto a la planificación del proyecto lo cuál creo que habla de forma positiva tanto del planteamiento de mi tutor como de la ejecución por mi parte.

Por otra parte en lo relativo al **Desarrollo Técnico** si es cierto que en primera instancia la elección del lenguaje de programación en el que desarrollar el trabajo me generó muchas dudas, las principales opciones eran *Python* y *R*, cada cuál con sus puntos fuertes y débiles propios, sin embargo, al final me decanté por *R* por sus potentes herramientas de análisis de datos así como por su origen estadístico. Durante el desarrollo de los prototipos a lo largo de la implementación ha habido cambios sustanciales, el más importante el cambio del tipo de simulación ya que originalmente **iba a ser un modelo de simulación basado en eventos discretos** y finalmente se ha desarrollado un modelo en tiempo continuo, lo cuál ha tenido un efecto importante en la recogida de datos, así como en el planteamiento de la ejecución del código. También ha habido cambios en la sección de Diseño por motivos de comodidad y eficiencia en la implementación.

Otro aspecto importante a remarcar es el **Aprendizaje y Conocimientos adquiridos**, en este ámbito el proyecto ha supuesto un gran beneficio en diversas facetas, tanto a nivel de implementación en el que he aprendido un lenguaje casi desconocido para mi hasta la fecha como era *R* en el que únicamente había programado una vez años atrás, como a nivel teórico, ya que he recordado y estudiado muchos conceptos diferentes, algunos del mundo de la probabilidad y la estadística, como la Teoría de Redes de Colas, o las distintas distribuciones estadísticas y sus características o como los distintos tipos de simulación y sus elementos, etc.

En **conclusión**, considero que este Trabajo Fin de Grado me ha aportado muchos aspectos beneficiosos en diversos campos, me ha permitido conocer el mundo de la simulación el cual era desconocido para mi y, en el que todavía hay mucho por decir, he sido capaz de desarrollar un proyecto en un lenguaje nada cómodo hasta el momento y creo que he podido aprender de un área tan compleja como llamativa y enriquecedora, lo que me motiva a seguir explorándola y nutriéndome de ella.

Bibliografía

- [1] Reptil.mx. «Modelos de simulación». (2021), dirección: <https://industriasgsl.com/blogs/automatizacion/%20modelos-de-simulacion> (visitado 16-03-2024).
- [2] UPC y F. Garriga Garzón, *Tome la mejor decisión experimentando previamente sus consecuencias*, es. OmniaScience, ISBN: 978-84-946352-4-3. DOI: 10.3926/oss.35. dirección: <https://www.omniascience.com/books/index.php/scholar/catalog%20book/48> (visitado 16-04-2024).
- [3] «Simulación Analógica», en U. de La Laguna, ed. dirección: https://fjarabo.webs.ull.es/CV/4_Documentos%20elaborados/%20Esquemas%20y%20Res%C3%BAmenes/1989%20-%20Autom%C3%A1tica%20-%20Teor%C3%ADa.pdf.
- [4] *Simulación y diseño basado en modelos con Simulink*. dirección: <https://es.mathworks.com/products/simulink.html> (visitado 14-04-2024).
- [5] *MATLAB - El lenguaje del cálculo técnico*, es. dirección: <https://es.mathworks.com/products/matlab.html> (visitado 14-04-2024).
- [6] *Simulacion de Monte Carlo prevision del retorno de la inversion utilizando tecnicas de simulacion de Monte Carlo*, es. dirección: <https://fastercapital.com/es/contenido/Simulacion-de-Monte-Carlo--prevision-del-retorno-de-la-inversion-utilizando-tecnicas-de-simulacion-de-Monte-Carlo.html> (visitado 14-04-2024).
- [7] J. B. M. Vega. «La importancia de R en la ciencia de datos». (2022), dirección: <https://geofabio.com/2022/12/17/la-importancia-%20de-r-en-la-ciencia-de-datos/> (visitado 18-03-2024).
- [8] I. Ucar, B. Smeets y A. Azcorra, «**simmer** : Discrete-Event Simulation for R», en, *Journal of Statistical Software*, vol. 90, 2019, ISSN: 1548-7660. DOI: 10.18637/jss.v090.i02. dirección: <http://www.jstatsoft.org/v90/i02/> (visitado 30-05-2024).
- [9] J. Sokolowski y C. Banks, *Principles of Modeling and Simulation: A Multidisciplinary Approach*. Wiley, 2011, ISBN: 9781118210949. dirección: <https://books.google.es/books?id=QrTJiwTwSG8C>.
- [10] J. Barcelo, *Simulacion de sistemas discretos*, es. Isdefe, 1996, ISBN: 978-84-89338-12-8.

- [11] «Biografía y CV de Jaume Barceló». (2019), dirección: <https://www.agrupacionciteec.udc.es/wp-content/uploads/2019/09/Resumen-CV-Jaume-Barcelo-Bugeda.pdf> (visitado 12-09-2019).
- [12] D. Insua, S. Insua, A. Martín y J. Jimenez, *Simulación. Métodos y aplicaciones (2a edición)* (Textos Universitarios). RA-MA S.A. Editorial y Publicaciones, 2008, ISBN: 9788478978953. dirección: <https://books.google.es/books?id=q3OqPgAACAAJ>.
- [13] M. A. M. Arevalo, *Descripción e implementación del Muestreador de Gibbs en versión bivariada*. dic. de 2016, Section: BLOG INDUSTRIA 4.0. dirección: <https://repository.udistrital.edu.co/bitstream/handle%20/11349/5572/Thesis.pdf?sequence=1>.
- [14] *Simulación de procesos*, es, Page Version ID: 155395600, 2023. dirección: https://es.wikipedia.org/w/index.php?title=Simulaci%C3%B3n_de_procesos&oldid=155395600.
- [15] *¿Qué es un diagrama de flujo? Tipos, símbolos y ejemplos* | Miro, es. dirección: <https://miro.com/es/diagrama-de-flujo/que-%20es-diagrama-de-flujo/>.
- [16] *Qué es la simulación de procesos industriales y cómo se realiza*, Section: BLOG INDUSTRIA 4.0, mayo de 2022. dirección: <https://www.cursosaula21.com/que-es-la-simulacion-de-procesos-industriales/>.
- [17] G. Oyarzún, *Simulación de procesos industriales: fiabilidad para tomar decisiones*, ene. de 2024. dirección: <https://blog.comparasoftware.com/simulacion%20-procesos-industriales/>.
- [18] Wikipedia, *Distribución de Poisson* — Wikipedia, La enciclopedia libre, [Internet; descargado 8-abril-2024], 2024. dirección: https://es.wikipedia.org/w/index.php?title=Distribuci%C3%B3n_de_Poisson&oldid=159301959.
- [19] Wikipedia, *Distribución binomial* — Wikipedia, La enciclopedia libre, [Internet; descargado 7-mayo-2024], 2024. dirección: https://es.wikipedia.org/w/index.php?title=Distribuci%C3%B3n_binomial&oldid=159973725.
- [20] Wikipedia, *Distribución uniforme continua* — Wikipedia, La enciclopedia libre, [Internet; descargado 30-abril-2024], 2024. dirección: https://es.wikipedia.org/w/index.php?title=Distribuci%C3%B3n_uniforme_continua&oldid=159817288.
- [21] M. T. Ortiz, *Estadística Computacional*. dirección: <https://tereom.github.io/est-computacional-2018/> (visitado 30-05-2024).
- [22] *Shiny*, en. dirección: <https://shiny.posit.co/> (visitado 30-05-2024).
- [23] *AnyLogic: Simulation Modeling Software Tools & Solutions for Business*, en. dirección: <https://www.anylogic.com/> (visitado 26-05-2024).

Anexos

Apéndice A

Primer anexo

```
library("simmer")
library("simmer.plot")
library("gridExtra")
library("dplyr")
library("plotly")
library("stringr")
library("tidyr")

# Funciones auxiliares
necesita_mant <- function() rbinom(1, 1, 0.50) == 1
necesita_chapa <- function() rbinom(1, 1, 0.35) == 1
necesita_pedir_piezas <- function() rbinom(1, 1, 0.35) == 1
pasa_inspeccion <- function() rbinom(1, 1, 0.75) == 1
necesita_mas_piezas <- function() rbinom(1, 1, 0.45) == 1
quiere_lavado <- function() rbinom(1, 1, 0.40) == 1
necesita_pulido <- function() rbinom(1, 1, 0.30) == 1
necesita_abolladura <- function() rbinom(1, 1, 0.35) == 1
necesita_pintura_parcial <- function() rbinom(1, 1, 0.40) == 1
necesita_pintura_completa <- function() rbinom(1, 1, 0.25) == 1
lavado_exterior <- function() rbinom(1, 1, 0.5) == 1
lavado_pistola <- function() rbinom(1, 1, 0.5) == 1

CHECK_VIP <- function() {
  ifelse(get_attribute("tipo_cliente") == 2, 4, # Cliente VIP
    ifelse(get_attribute("tipo_cliente") == 4, 3, # Cliente ITV
      ifelse(get_attribute("tipo_cliente") == 1, 2, # Cliente
        ifelse(get_attribute("tipo_cliente") == 3, 1, 2))))
  # Cliente lavado
}

CHECK_FALLO_GRAVE <- function() {
  if (get_attribute("fallo_tipo") == 2) {
    return(1)
  } else {
    return(0)
  }
}
```

```
# -----
# -----VEHÍCULO-----
# -----
Vehicle <- setRefClass("Vehicle",
  fields = list(
    tipo = "character",
    antiguedad = "numeric",
    prob_fallo_grave = "numeric",
    prob_fallo_leve = "numeric"
  ),
  methods = list(
    initialize = function(tipo, antiguedad) {
      .self$tipo <- tipo
      .self$antiguedad <- antiguedad
      .self$calculate_probabilities()
    },
    calculate_probabilities = function() {
      stop("Debe ser implementado por subclases.")
    }
  )
)

# Definir subclase Coche
Coche <- Vehicle$copy()
# Utiliza copy para crear una subclase
Coche$methods(calculate_probabilities = function() {
  if (.self$antiguedad < 5) {
    .self$prob_fallo_grave <- rbinom(1, 1, 0.063)
    .self$prob_fallo_leve <- rbinom(1, 1, 0.030)
  } else if (.self$antiguedad <= 10) {
    .self$prob_fallo_grave <- rbinom(1, 1, 0.099)
    .self$prob_fallo_leve <- rbinom(1, 1, 0.050)
  } else if (.self$antiguedad <= 15) {
    .self$prob_fallo_grave <- rbinom(1, 1, 0.175)
    .self$prob_fallo_leve <- rbinom(1, 1, 0.088)
  } else {
    .self$prob_fallo_grave <- rbinom(1, 1, 0.235)
    .self$prob_fallo_leve <- rbinom(1, 1, 0.118)
  }
})

# Similarmente, definir Moto
Moto <- Vehicle$copy()
Moto$methods(calculate_probabilities = function() {
  if (.self$antiguedad < 5) {
    .self$prob_fallo_grave <- rbinom(1, 1, 0.245)
    .self$prob_fallo_leve <- rbinom(1, 1, 0.122)
  } else if (.self$antiguedad <= 10) {
    .self$prob_fallo_grave <- rbinom(1, 1, 0.272)
    .self$prob_fallo_leve <- rbinom(1, 1, 0.136)
  } else {
    .self$prob_fallo_grave <- rbinom(1, 1, 0.283)
  }
})
```

```

        .self$prob_fallo_leve <- rbinom(1, 1, 0.141)
    }
})

# Definir subclase Camion
Camion <- Vehicle$copy()
Camion$methods(calculate_probabilities = function() {
    if (.self$antiguedad <= 10) {
        .self$prob_fallo_grave <- rbinom(1, 1, 0.234)
        .self$prob_fallo_leve <- rbinom(1, 1, 0.117)
    } else {
        .self$prob_fallo_grave <- rbinom(1, 1, 0.284)
        .self$prob_fallo_leve <- rbinom(1, 1, 0.142)
    }
})

# Asignar tipo de vehículo
asignar_tipo_vehiculo <- function(tipo) {
    if (tipo == "Coche") {
        return(1)
    } else if (tipo == "Camion") {
        return(2)
    } else {
        return(3)
    }
}

generate_vehicle_attributes <- function() {
    type <- sample(c("Coche", "Camion", "Moto"), 1,
        prob = c(0.5, 0.15, 0.35))
    age <- sample(1:20, 1)
    vehiculo <- switch(type,
        "Coche" = Coche$new(tipo = type, antiguedad = age),
        "Camion" = Camion$new(tipo = type, antiguedad = age),
        "Moto" = Moto$new(tipo = type, antiguedad = age))
    vehiculo$calculate_probabilities()
    tipo <- asignar_tipo_vehiculo(vehiculo$tipo)
    antiguedad <- vehiculo$antiguedad
    fallo_tipo <- ifelse(vehiculo$prob_fallo_grave ==
        1, 2, ifelse(vehiculo$prob_fallo_leve == 1, 1, 0))
}

# -----
# -----ITV-----
# -----
trayectoria_itv <- function() {
    trajectory() %>%
        set_attribute("vehiculo_tipo", function() {
            generate_vehicle_attributes()
            return(tipo)
        }) %>%
        set_attribute("vehiculo_antiguedad", function() {
            return(antiguedad)
        })
}

```

Capítulo A. Primer anexo

```
}) %>%
set_attribute("fallo_tipo", function() {
  return(fallo_tipo)
}) %>%
seize("ITV", 1, priority = CHECK_VIP) %>%
log_("Inicio Revisión ITV") %>%
set_attribute("EnITV", 1) %>%
renege_in(
  function() { rexp(n = 1, rate = 1/40) },
  out = trajectory("Impaciente") %>%
  set_attribute("Impaciente", 1) %>%
  log_("Cliente impaciente, abandona el sistema")
) %>%
timeout(function() rexp(1, rate = 1/20)) %>%
renege_abort() %>%
log_("Fin Revisión ITV") %>%
set_attribute("EnITV", 0) %>%
release("ITV") %>%
set_attribute("costo", 75) %>%
branch(
  function() {
    fallo_tipo == 2
  },
  continue = FALSE,
  trajectory() %>%
  log_("Fallo grave, uniendo a
    trayectoria principal") %>%
  join(trayectoria_principal())
) %>%
log_("Fallo leve o sin fallos, cliente
  finaliza proceso ITV")
}

# -----
# -----MANTENIMIENTO-----
# -----
trayectoria_evaluacion <- function() {
  trajectory() %>%
  seize("Evaluacion", 1, priority = CHECK_VIP) %>%
  log_("INICIO Evaluacion") %>%
  set_attribute("EnEvaluacion", 1) %>%
  timeout(function() rexp(1, rate = 1/20)) %>%
  log_("FIN Evaluacion") %>%
  set_attribute("EnEvaluacion", 0) %>%
  release("Evaluacion") %>%
  set_attribute("costo", 10) %>%
  branch(
    necesita_mant,
    continue = FALSE,
    trajectory() %>%
    log_("Necesita Mantenimiento") %>%
    join(trayectoria_principal()) %>%
  branch(
```

```

necesita_chapa, # Si necesita chapa tras mantenimiento
continue = FALSE,
trajectory() %>%
  log_("Procede a Chapa y Pintura después del
Mantenimiento") %>%
  join(trayectoria_chapa_y_pint()) %>%
  branch(
    quiere_lavado, # Decide si quiere lavado después
    de chapa y pintura
    continue = TRUE,
    trajectory() %>%
      log_("Opta por Lavado después de Chapa
y Pintura") %>%
      join(trayectoria_lavado())
  )
) %>%
branch(
  quiere_lavado, # Si quiere lavado directamente
  después del mantenimiento
  continue = TRUE,
  trajectory() %>%
    log_("Opta por Lavado después del
Mantenimiento") %>%
    join(trayectoria_lavado())
  )
) %>%
log_("No necesita Mantenimiento, procede a Chapa
y Pintura") %>%
join(trayectoria_chapa_y_pint()) %>%
branch(
  quiere_lavado,
  continue = TRUE,
  trajectory() %>%
    log_("Opta por Lavado después de
Chapa y Pintura") %>%
    join(trayectoria_lavado())
  )
) %>%
log_("SALE DEL SISTEMA")
}

trayectoria_espera_piezas <- function() {
  trajectory() %>%
  branch(
    necesita_pedir_piezas,
    continue = FALSE,
    trajectory() %>%
      log_("INICIO EsperaPiezas") %>%
      log_("No está la pieza en el taller,
necesitan ser pedidas") %>%
      seize("EsperaPiezas", 1, priority = CHECK_VIP) %>%
      set_attribute("EnEsperaPiezas", 1) %>%
      timeout(function() rexp(1, rate = 1/120)) %>%
      # Mayor tiempo de espera por envío

```

Capítulo A. Primer anexo

```
    set_attribute("costo", 120) %>%
    # Costo adicional por envío
    log_("FIN Espera Piezas con pedido") %>%
    set_attribute("EnEsperaPiezas", 0) %>%
    release("EsperaPiezas")
  ) %>%
  log_("INICIO EsperaPiezas") %>%
  log_("Piezas disponibles en el taller.") %>%
  seize("EsperaPiezas", 1, priority = CHECK_VIP) %>%
  set_attribute("EnEsperaPiezas", 1) %>%
  timeout(function() rexp(1, rate = 1/10)) %>%
  # Menor tiempo de espera
  set_attribute("costo", 80) %>%
  # Costo reducido sin envío
  log_("FIN Espera Piezas sin pedido") %>%
  set_attribute("EnEsperaPiezas", 0) %>%
  release("EsperaPiezas")
}

trayectoria_reparacion <- function() {
  trajectory() %>%
  seize("Reparacion", 1, priority = CHECK_VIP) %>%
  ## Prioridad si espero por piezas
  log_("INICIO Reparacion") %>%
  set_attribute("EnReparacion", 1) %>%
  timeout(function() rexp(1, rate = 1/60)) %>%
  ## 60 minutos
  log_("FIN Reparacion") %>%
  set_attribute("EnReparacion", 0) %>%
  release("Reparacion") %>%
  set_attribute("costo", 180) ## 180 EUR
}

trayectoria_inspeccion <- function() {
  trajectory() %>%
  seize("Inspeccion", 1, priority = CHECK_VIP) %>%
  log_("INICIO Inspeccion") %>%
  set_attribute("EnInspeccion", 1) %>%
  timeout(function() rexp(1, rate = 1/15)) %>%
  ## 15 minutos
  release("Inspeccion") %>%
  set_attribute("costo", 20) %>% ## 20 EUR

  branch( function() !pasa_inspeccion(),
    continue = TRUE,
    trajectory() %>%
    branch( necesita_mas_piezas,
      continue = FALSE,
      trajectory() %>%
      log_("FIN Inspeccion --> necesita más piezas") %>%
      set_attribute("EnInspeccion", 0) %>%
      join(trayectoria_espera_piezas()) %>%

```

```

        join(trayectoria_reparacion()) %>%
        log_("FIN Inspeccion --> Inspección Aprobada") %>%
        log_("FIN Mantenimiento") %>%
        set_attribute("EnInspeccion", 0)
    ) %>%
    log_("FIN Inspeccion --> necesita reparación
    adicional") %>%
    set_attribute("EnInspeccion", 0) %>%
    join(trayectoria_reparacion())
    ) %>%
    log_("FIN Inspeccion --> Inspección Aprobada") %>%
    log_("FIN Mantenimiento") %>%
    set_attribute("EnInspeccion", 0)
}

# -----
# -----CHAPA Y PINTURA -----
# -----
# Trayectoria principal de chapa y pintura
trayectoria_chapa_y_pint <- function() {
    trajectory("Proceso Chapa y Pintura") %>%
    log_("INICIO Chapa y Pintura") %>%
    set_attribute("EnChapaPintura", 1) %>%
    branch(
        necesita_pintura_parcial,
        continue = TRUE,
        trajectory() %>%
        join(trayectoria_pintura_parcial())
    ) %>%
    branch(
        necesita_pintura_completa,
        continue = TRUE,
        trajectory() %>%
        join(trayectoria_pintura_completa())
    ) %>%
    branch(
        necesita_abolladura,
        continue = TRUE,
        trajectory() %>%
        join(trayectoria_abolladura())
    ) %>%
    branch(
        necesita_pulido,
        continue = TRUE,
        trajectory() %>%
        join(trayectoria_pulido())
    ) %>%
    log_("FIN Chapa y Pintura") %>%
    set_attribute("EnChapaPintura", 0)
}

# Trayectorias específicas para cada servicio dentro
de chapa y pintura

```

Capítulo A. Primer anexo

```
trayectoria_pintura_parcial <- function() {
  trajectory("Pintura Parcial") %>%
    seize("PinturaPar", 1, priority = CHECK_VIP) %>%
    log_("INICIO Pintura Parcial") %>%
    set_attribute("EnPinturaPar", 1) %>%
    set_attribute("pintoParcialmente", 1) %>%
    timeout(function() rexp(1, rate = 1/30)) %>%
    log_("FIN Pintura Parcial") %>%
    set_attribute("EnPinturaPar", 0) %>%
    release("PinturaPar") %>%
    set_attribute("costo", 200)
}

trayectoria_pintura_completa <- function() {
  trajectory("Pintura Completa") %>%
    seize("PinturaComp", 1, priority = CHECK_VIP) %>%
    log_("INICIO Pintura Completa") %>%
    set_attribute("EnPinturaComp", 1) %>%
    timeout(function() rexp(1, rate = 1/90)) %>%
    log_("FIN Pintura Completa") %>%
    set_attribute("EnPinturaComp", 0) %>%
    release("PinturaComp") %>%
    set_attribute("costo", 400)
}

trayectoria_abolladura <- function() {
  trajectory("Reparación de Abolladuras") %>%
    seize("Abolladura", 1, priority = CHECK_VIP) %>%
    log_("INICIO Abolladura") %>%
    set_attribute("EnAbolladura", 1) %>%
    timeout(function() rexp(1, rate = 1/35)) %>%
    log_("FIN Abolladura") %>%
    set_attribute("EnAbolladura", 0) %>%
    release("Abolladura") %>%
    set_attribute("costo", 250)
}

trayectoria_pulido <- function() {
  trajectory("Pulido") %>%
    seize("Pulido", 1, priority = CHECK_VIP) %>%
    log_("INICIO Pulido") %>%
    set_attribute("EnPulido", 1) %>%
    timeout(function() rexp(1, rate = 1/25)) %>%
    log_("FIN Pulido") %>%
    set_attribute("EnPulido", 0) %>%
    release("Pulido") %>%
    set_attribute("costo", 110)
}

# -----
# -----LAVADO-----
# -----
```

```

trayectoria_lavado <- function() {
  trajectory() %>%
    seize("Lavado", 1, priority = CHECK_VIP) %>%
    log_("INICIO Lavado") %>%
    set_attribute("EnLavado", 1) %>%
    branch(
      lavado_exterior,
      continue = FALSE,
      trajectory() %>%
        log_("Lavado Exterior seleccionado") %>%
        branch(
          lavado_pistola,
          continue = FALSE,
          trajectory() %>%
            log_("Lavado con Pistola seleccionado") %>%
            timeout(function() rexp(1, rate = 1/20)) %>%
            # Tiempo para lavado con pistola
            set_attribute("costo", 20) %>%
            log_("FIN Lavado") %>%
            set_attribute("EnLavado", 0) %>%
            release("Lavado") %>%
            log_("SALE DEL SISTEMA")
        ) %>%
        log_("Túnel de Lavado seleccionado") %>%
        timeout(function() rexp(1, rate = 1/10)) %>%
        # Tiempo para túnel de lavado
        set_attribute("costo", 15) %>%
        log_("FIN Lavado") %>%
        set_attribute("EnLavado", 0) %>%
        release("Lavado") %>%
        log_("SALE DEL SISTEMA")
    ) %>%
    log_("Lavado Completo seleccionado") %>%
    timeout(function() rexp(1, rate = 1/60)) %>%
    # Tiempo para lavado completo
    set_attribute("costo", 50) %>%
    log_("FIN Lavado") %>%
    set_attribute("EnLavado", 0) %>%
    release("Lavado") %>%
    log_("SALE DEL SISTEMA")
}

# -----
# -----TRAYECTORIA PRINCIPAL-----
# -----
# # Trayectoria principal que une todas las partes
trayectoria_principal <- function() {
  trajectory("Proceso Principal") %>%
    join(trayectoria_espera_piezas()) %>%
    join(trayectoria_reparacion()) %>%
    join(trayectoria_inspeccion())
}

```

Capítulo A. Primer anexo

```
trayectoria_vip <- function() {
  trajectory() %>%
    set_attribute("costo", 50) %>% # Costo inicial para VIPs
    join(trayectoria_evaluacion())
}

# -----
# -----REPETICIONES-----
# -----

base_output_folder <- "D:/UNIVERSIDAD/TFG/Codigo/Prototipo2"

# Función para correr una simulación
correr_simulacion <- function(replica_id, tiempo) {
  sim_folder <- file.path(base_output_folder,
    sprintf("Simulacion_%03d", replica_id))

  if (!dir.exists(sim_folder)) {
    dir.create(sim_folder, recursive = TRUE)
  }

  # Establecer una semilla única para cada réplica
  set.seed(replica_id)

  # Crear el archivo de log
  log_file <- file.path(sim_folder,
    sprintf("TRAZAS_sim_%03d.txt", replica_id))
  sink(log_file)
  on.exit(sink(), add = TRUE)

  # Creación del entorno de simulación
  env <- simmer("Taller de Vehiculos")
  recursos_list <- c("Evaluacion", "Reparacion",
    "EsperaPiezas", "Inspeccion", "Lavado",
    "PinturaPar", "PinturaComp", "Abolladura",
    "Pulido", "ITV")

  capturar_datos <- function(env) {
    capacity <- sapply(recursos_list,
      function(res) get_capacity(env, res))
    utilization <- sapply(recursos_list,
      function(res) get_server_count(env, res)
        / get_capacity(env, res))
    queue_lengths <- sapply(recursos_list,
      function(res) mean(get_mon_resources(env)
        %>% filter(resource == res) %>% pull(queue)))
    service_times <- sapply(recursos_list,
      function(res) mean(get_mon_resources(env)
        %>% filter(resource == res) %>% pull(time)))
    total_usage_times <- sapply(recursos_list,
      function(res) sum(get_mon_resources(env) %>%
        filter(resource == res) %>% pull(time)))
    num_served <- sapply(recursos_list,
      function(res) nrow(get_mon_resources(env) %>%
```

```

filter(resource == res, server > 0)))

arrivals <- get_mon_arrivals(env)
waiting_times <- arrivals %>%
  mutate(
    time_in_system = end_time - start_time,
    waiting_time = pmax(0, end_time - start_time
      - activity_time)
  ) %>%
  group_by(name) %>%
  summarise(
    mean_waiting_time = mean(waiting_time, na.rm = TRUE),
    mean_time_in_system = mean(time_in_system, na.rm = TRUE)
  )

list(
  capacity = capacity,
  utilization = utilization,
  queue_lengths = queue_lengths,
  service_times = service_times,
  total_usage_times = total_usage_times,
  num_served = num_served,
  waiting_times = waiting_times
)
}

for (recurso in recursos_list) {
  env <- env %>% add_resource(recurso, 2, queue_size = Inf)
}

env <- env %>%
  add_generator("cliente",
    trajectory() %>%
      set_attribute("tipo_cliente", 1) %>%
      join(trayectoria_evaluacion()),
    function() rexp(n = 1, rate = 1/25),
    priority = 2, mon = 2) %>%
  add_generator("cliente VIP",
    trajectory() %>%
      set_attribute("tipo_cliente", 2) %>%
      join(trayectoria_vip()),
    function() rexp(n = 1, rate = 1/40),
    priority = 4, mon = 2) %>%
  add_generator("cliente Lavado",
    trajectory() %>%
      set_attribute("tipo_cliente", 3) %>%
      join(trayectoria_lavado()),
    function() rexp(n = 1, rate = 1/60),
    priority = 1, mon = 2) %>%
  add_generator("cliente ITV",
    trajectory() %>%
      set_attribute("tipo_cliente", 4) %>%
      join(trayectoria_itv()),

```

Capítulo A. Primer anexo

```
function() rexp(n = 1, rate = 1/60),
priority = 3, mon = 2)

# Correr la simulación
env %>% run(until = tiempo)

# Guardar resultados
recursos <- get_mon_resources(env)
arrivals <- get_mon_arrivals(env)
atributos <- get_mon_attributes(env)

costos_por_cliente <- atributos %>%
  filter(key == "costo") %>%
  group_by(name) %>%
  summarise(costo_total = sum(as.numeric(value)))

calcular_costo_promedio <- function(tipo_cliente) {
  total_cost <- sum(costos_por_cliente %>%
    filter(grepl(tipo_cliente, name)) %>%
    pull(costo_total))
  total_arrivals <- nrow(arrivals %>%
    filter(grepl(tipo_cliente, name)))
  if (total_arrivals > 0) {
    total_cost / total_arrivals
  } else {
    NA
  }
}

costos_promedio <- data.frame(
  tipo_cliente = c("Cliente", "Cliente VIP",
    "Cliente Lavado", "Cliente ITV"),
  costo_promedio = c(
    calcular_costo_promedio("cliente"),
    calcular_costo_promedio("cliente VIP"),
    calcular_costo_promedio("cliente Lavado"),
    calcular_costo_promedio("cliente ITV")
  )
)

# Obtener información adicional
datos_adicionales <- capturar_datos(env)

# Crear un dataframe para la información adicional
info_adicional <- data.frame(
  recurso = recursos_list,
  capacidad = datos_adicionales$capacity,
  utilizacion = datos_adicionales$utilization,
  longitud_mediaCola = datos_adicionales$queue_lengths,
  tiempo_medio_servicio = datos_adicionales$service_times,
  tiempo_total_uso = datos_adicionales$total_usage_times,
  num_served = datos_adicionales$num_served
)
```

```

# Guardar en archivos CSV
write.csv(costos_por_cliente, file.path(sim_folder,
"facturas.csv"), row.names = FALSE)
write.csv(arrivals, file.path(sim_folder, "arrivals.csv"),
row.names = FALSE)
write.csv(recursos, file.path(sim_folder, "recursos.csv"),
row.names = FALSE)
write.csv(atributos, file.path(sim_folder, "atributos.csv"),
row.names = FALSE)
write.csv(info_adicional, file.path(sim_folder,
"info_adicional.csv"), row.names = FALSE)
write.csv(datos_adicionales$waiting_times,
file.path(sim_folder, "tiempos_espera.csv"),
row.names = FALSE)
write.csv(costos_promedio, file.path(sim_folder,
"costos_promedio.csv"), row.names = FALSE)

itv_clients <- atributos %>%
  filter(grepl("cliente ITV", name)) %>%
  filter(key %in% c("vehiculo_tipo",
"vehiculo_antiguedad", "fallo_tipo")) %>%
  pivot_wider(names_from = key,
values_from = value) %>%
  filter(!is.na(vehiculo_tipo) &
!is.na(vehiculo_antiguedad) &
!is.na(fallo_tipo)) %>%
  # Filtra solo las filas con todos los atributos
  mutate(
    vehiculo_tipo = recode(vehiculo_tipo, `1` = "Coche",
`2` = "Camion", `3` = "Moto"),
    fallo_tipo = recode(fallo_tipo, `2` = "grave",
`1` = "leve", `0` = "sin fallos")
  )

# Guardar en archivo CSV
write.csv(itv_clients, file.path(sim_folder,
"itv_clients.csv"), row.names = FALSE)

return(env)
}

# Ejecutar las réplicas
numero_de_replicas <- 10
tiempo_de_simulacion <- 30 * 12 * 60 # 30 días en minutos

reps <- lapply(1:numero_de_replicas, function(i) {
  correr_simulacion(i, tiempo_de_simulacion)
})

all_recursos <- get_mon_resources(reps)
all_arrivals <- get_mon_arrivals(reps)
all_atributos <- get_mon_attributes(reps)

```

Capítulo A. Primer anexo

```
# Guardar gráficos y resultados globales
global_output_folder <- file.path(base_output_folder, "Global")
if (!dir.exists(global_output_folder)) {
  dir.create(global_output_folder, recursive = TRUE)
}

# Supongamos que 'p5' es un objeto ggplot
p5 <- plot(all_recursos, metric = "utilization",
c("Evaluacion", "Reparacion", "EsperaPiezas", "Inspeccion",
"Lavado", "PinturaPar", "PinturaComp", "Abolladura",
"Pulido", "ITV"))

# Convertir el gráfico a ggplot y ajustar el ancho
de las barras (si es necesario)
p5 <- p5 +
  theme(axis.text.x = element_text(angle = 45, hjust = 1,
size = 13), # Rotar las etiquetas del eje x
axis.title.x = element_text(size = 14),
# Aumentar tamaño del título del eje x
axis.title.y = element_text(size = 14),
# Aumentar tamaño del título del eje y
plot.title = element_text(size = 16, hjust = 0.5))
# Aumentar tamaño del título del gráfico

# Ajustar el tamaño de la ventana del gráfico
file_path_p5 <- file.path(
global_output_folder,
"utilization_plot_global.png")
if (file.exists(file_path_p5)) file.remove(file_path_p5)

# Ajustar el tamaño de la imagen guardada
ggsave(file_path_p5, plot = p5)

p6 <- plot(all_recursos, metric = "usage", c("Evaluacion",
"Reparacion", "EsperaPiezas"), items = c("queue", "server"))
file_path_p6 <- file.path(global_output_folder,
"usage_plot_global.png")
if (file.exists(file_path_p6)) file.remove(file_path_p6)
ggsave(file_path_p6, plot = p6)

p7 <- plot(all_recursos, metric = "usage", c("Inspeccion",
"Lavado", "PinturaPar"), items = c("queue", "server"))
file_path_p7 <- file.path(global_output_folder,
"usage_plot_global2.png")
if (file.exists(file_path_p7)) file.remove(file_path_p7)
ggsave(file_path_p7, plot = p7)

p8 <- plot(all_recursos, metric = "usage", c("PinturaComp",
"Abolladura", "Pulido"), items = c("queue", "server"))
file_path_p8 <- file.path(global_output_folder,
"usage_plot_global3.png")
if (file.exists(file_path_p8)) file.remove(file_path_p8)
```

```
ggsave(file_path_p8, plot = p8)

p9 <- plot(all_recurcos, metric = "usage", c("ITV"),
  items = c("queue", "server"))
file_path_p8 <- file.path(global_output_folder,
  "usage_plot_global4.png")
if (file.exists(file_path_p8)) file.remove(file_path_p8)
ggsave(file_path_p8, plot = p8)


p10 <- plot(all_arrivals, metric = "waiting_time")
file_path_p9 <- file.path
(global_output_folder,
  "waiting_time_plot_global.png")
if (file.exists(file_path_p9))
file.remove(file_path_p9)
ggsave(file_path_p9, plot = p9)

p11 <- plot(all_arrivals, metric = "activity_time")
file_path_p10 <- file.path(
global_output_folder,
  "activity_time_plot_global.png")
if (file.exists(file_path_p10))
file.remove(file_path_p10)
ggsave(file_path_p10, plot = p10)

write.csv(all_recurcos, file.path(
global_output_folder, "recurcos_global.csv"),
row.names = FALSE)
write.csv(all_arrivals, file.path(
global_output_folder, "arrivals_global.csv"),
row.names = FALSE)
write.csv(all_atributos, file.path(
global_output_folder, "atributos_global.csv"),
row.names = FALSE)
```


Apéndice B

Segundo anexo




Recibo digital

Este recibo confirma que su trabajo ha sido recibido por Turnitin. A continuación podrá ver la información del recibo con respecto a su entrega.

La primera página de tus entregas se muestra abajo.

Autor de la entrega:	EDUARDO MARQUINA GARCIA
Título del ejercicio:	Turnitin Memoria Final (Moodle PP)
Título de la entrega:	1475_tfg_latex_etsiinf_2023_06_01.pdf
Nombre del archivo:	31222_EDUARDO_MARQUINA_GARCIA_1475_tfg_latex_etsiinf...
Tamaño del archivo:	3.43M
Total páginas:	91
Total de palabras:	22,446
Total de caracteres:	121,587
Fecha de entrega:	25-jun.-2024 03:20p. m. (UTC+0200)
Identificador de la entre...	2408450447



Universidad Politécnica
de Madrid

Escuela Técnica Superior de
Ingenieros Informáticos


Grado en Ingeniería Informática

Trabajo Fin de Grado

**MODELO de SIMULACIÓN de un
TALLER de MANTENIMIENTO**

Autor: Eduardo Marquina García
Revisor: Juan Antonio Fernández del Pozo de Salazar

Madrid, febrero 2024



Derechos de autor 2024 Turnitin. Todos los derechos reservados.

Figura B.1: Informe de Originalidad Turnitin

Este documento esta firmado por



Firmante	CN=tfgm.fi.upm.es, OU=CCFI, O=ETS Ingenieros Informaticos - UPM, C=ES
Fecha/Hora	Sun Jun 30 20:41:08 CEST 2024
Emisor del Certificado	EMAILADDRESS=camanager@etsiinf.upm.es, CN=CA ETS Ingenieros Informaticos, O=ETS Ingenieros Informaticos - UPM, C=ES
Numero de Serie	561
Metodo	urn:adobe.com:Adobe.PPKLite:adbe.pkcs7.sha1 (Adobe Signature)