



UNIVERSIDAD DE BURGOS
ESCUELA POLITÉCNICA SUPERIOR
Grado en Ingeniería Informática



TFG del Grado en Ingeniería
Informática

Análisis y predicción de datos
obtenidos del funcionamiento
de un AGV



Presentado por Gonzalo Burgos de la Hera
en Universidad de Burgos — 24 de marzo
de 2023

Tutor: Bruno Baruque Zanón y Jesús Enrique
Sierra García



UNIVERSIDAD DE BURGOS
ESCUELA POLITÉCNICA SUPERIOR
Grado en Ingeniería Informática



D. nombre tutor, profesor del departamento de nombre departamento, área de nombre área.

Expone:

Que el alumno D. Gonzalo Burgos de la Hera, con DNI dni, ha realizado el Trabajo final de Grado en Ingeniería Informática titulado título de TFG.

Y que dicho trabajo ha sido realizado por el alumno bajo la dirección del que suscribe, en virtud de lo cual se autoriza su presentación y defensa.

En Burgos, 24 de marzo de 2023

Vº. Bº. del Tutor:

Vº. Bº. del co-tutor:

D. nombre tutor

D. nombre co-tutor

Resumen

En este primer apartado se hace una **breve** presentación del tema que se aborda en el proyecto.

Descriptores

Palabras separadas por comas que identifiquen el contenido del proyecto Ej: servidor web, buscador de vuelos, android ...

Abstract

A **brief** presentation of the topic addressed in the project.

Keywords

keywords separated by commas.

Índice general

Índice general	iii
Índice de figuras	v
Índice de tablas	vi
Introducción	1
1.1. INTRO TEMPORAL	1
1.2. Data storing	1
Objetivos del proyecto	7
Conceptos teóricos	9
3.1. Secciones	9
3.2. Referencias	9
3.3. Imágenes	10
3.4. Listas de items	10
3.5. Tablas	11
Técnicas y herramientas	13
4.1. Sistema Gestor de Base de Datos	13
Aspectos relevantes del desarrollo del proyecto	15
Trabajos relacionados	17
Conclusiones y Líneas de trabajo futuras	19

Bibliografía	21
---------------------	-----------

Índice de figuras

3.1. Autómata para una expresión vacía	10
--	----

Índice de tablas

1.1. General information comparison	3
1.2. Data model comparison	3
1.3. Support	4
1.4. Other	4
3.1. Herramientas y tecnologías utilizadas en cada parte del proyecto	11
4.1. Comparación gestores de bases de datos	13

Introducción

Descripción del contenido del trabajo y del estructura de la memoria y del resto de materiales entregados.

1.1. INTRO TEMPORAL

Our goal is to collect data on a fleet of Autonomous Guided Vehicles (AGVs) through log messages sent from each vehicle. These messages contain information about key parameters of the vehicle, and are indexed over time. Later, we intend to use this information to predict future behaviour, so the fleet can be managed in a more optimal way. Because access to an AGV is not always possible, we also intend to develop a simulator, which will be useful to gather the data needed to train the model.

The first part of this work will address which database model and software to use to store the information, and the second will address the data analysis and prediction part.

1.2. Data storing

Database model

The first thing to choose is the database model to use, as this will have a key importance when deciding which database management system to use. The next models will be considered:

1. Relational database management systems: in these systems, data is stored in relations. A relation, also sometimes called a table, is a collection of tuples, or rows. Those tables are defined by their name, and

a set of fixed number of attributes, or columns, with fixed data types. These systems respect the ACID properties (Atomicity, Consistency, Isolation and Durability), and have basic operations defined, like selection, projection and join.

2. Document stores: the principal characteristic of this kind of system is their schema-free organization of data. This means that, contrary that in relational databases, records don't need to have a uniform structure, columns can have more than one value, and even records can have a nested structure.
3. Key-Value stores: they are the most basic and simplest database management systems, and can only store key-value pairs. Those are usually not feasible for complex applications, but usually have good performance due to their simplicity.
4. Search engines: their main usage is the search for data content, and are typically NoSQL database management systems, meaning that they don't use a relational data model.
5. Time Series database management systems: this database systems are optimized for storing time series data[3]. Although they are typically NoSQL, some systems of this kind follow the relational data model.

Time series data, like the data we receive from the AGVs, can be managed with any of those models. However, Time Series DBMS are specialized in this kind of work, making them the best suit for our needs. Knowing this, we will make a comparison in the following chapter between the top 5 most used TSDBMS, according to DB-Engines[1].

DataBase Management System

The next DBMS will be compared:

- InfluxDB.
- Kdb.
- Prometheus.
- Graphite.
- TimescaleDB.

General information

Systems	Developer	Launch	Latest version	Licence
InfluxDB	InfluxData	2013	2022	OSS ¹
Kdb	Kx Systems	2000	2018	Commercial ²
Prometheus		2015	2023	OSS
Graphite		2006	2022	OSS
TimescaleDB	Timescale	2017	2022	OSS

Tabla 1.1: General information comparison

Only Open Source Software will be considered for this work. Kdb, therefore, will not be used, and will not appear in the following comparisons. For the same reason, only the features of the Community Edition of InfluxDB will be used in those comparisons, and features of the Enterprise Edition, like sharding, will not be taken in mind.

Data model

System	Model	Schema	Typing	Secondary Index
InfluxDB	Multidimensional	Schema-free	Numeric and string	No
Prometheus	Multidimensional	Yes	Numeric	No
Graphite	Key-Value	Yes	Numeric	No
TimescaleDB	Relational	Yes	All	Yes

Tabla 1.2: Data model comparison

¹Commercial Enterprise Edition.

²Free 32 bit version.

Support

System	OS	Python	API	SQL
InfluxDB	Linux OS x	Yes	HTTP API Json UDP	SQL-like
Prometheus	Linux Windows	Yes	REST HTTP JSON API	No
Graphite	Linux Unix	Yes	HTTP API Sockets	No
TimescaleDB	Linux OS X Windows	Yes	JDBC ODBC C Library	Yes

Tabla 1.3: Support

Other

System	InfluxDB	Prometheus	Graphite	TimescaleDB
Server scripts	No	No	No	Yes
Partitioning	No ³	Sharding	No	Yes
Replication	No ⁴	Yes	No	Yes
Consistency	Eventual ⁵	No	No	Innmediate
Transaction concepts	No	No	No	ACID
Concurrency	Yes	Yes	Yes ⁶	Yes
Durability	Yes	Yes	Yes	Yes
User rights	Rights via user accounts	No	No	Access rights SQL-standard

Tabla 1.4: Other

³Only in Enterprise Edition.

⁴Only in Enterprise Edition.

⁵According to the InfluxDB documentation, InfluxDB prioritizes read and write performance over strong consistency. However, it is ensured that data is eventually consistent.^[2]

⁶Concurrency by blocking.

Performance

Conclusion

Objetivos del proyecto

Este apartado explica de forma precisa y concisa cuales son los objetivos que se persiguen con la realización del proyecto. Se puede distinguir entre los objetivos marcados por los requisitos del software a construir y los objetivos de carácter técnico que plantea a la hora de llevar a la práctica el proyecto.

Conceptos teóricos

En aquellos proyectos que necesiten para su comprensión y desarrollo de unos conceptos teóricos de una determinada materia o de un determinado dominio de conocimiento, debe existir un apartado que sintetice dichos conceptos.

Algunos conceptos teóricos de L^AT_EX⁷.

3.1. Secciones

Las secciones se incluyen con el comando `section`.

Subsecciones

Además de secciones tenemos subsecciones.

Subsubsecciones

Y subsecciones.

3.2. Referencias

Las referencias se incluyen en el texto usando `cite` [5]. Para citar webs, artículos o libros [4].

⁷Créditos a los proyectos de Álvaro López Cantero: Configurador de Presupuestos y Roberto Izquierdo Amo: PLQuiz

3.3. Imágenes

Se pueden incluir imágenes con los comandos standard de \LaTeX , pero esta plantilla dispone de comandos propios como por ejemplo el siguiente:



Figura 3.1: Autómata para una expresión vacía

3.4. Listas de items

Existen tres posibilidades:

- primer item.
- segundo item.

1. primer item.
2. segundo item.

Primer item más información sobre el primer item.

Segundo item más información sobre el segundo item.

■

Herramientas	App	AngularJS	API REST	BD	Memoria
HTML5		X			
CSS3		X			
BOOTSTRAP		X			
JavaScript		X			
AngularJS		X			
Bower		X			
PHP			X		
Karma + Jasmine		X			
Slim framework			X		
Idiorm			X		
Composer			X		
JSON		X	X		
PhpStorm		X	X		
MySQL				X	
PhpMyAdmin				X	
Git + BitBucket		X	X	X	X
MikTeX					X
TeXMaker					X
Astah					X
Balsamiq Mockups		X			
VersionOne		X	X	X	X

Tabla 3.1: Herramientas y tecnologías utilizadas en cada parte del proyecto

3.5. Tablas

Igualmente se pueden usar los comandos específicos de \LaTeX o bien usar alguno de los comandos de la plantilla.

Técnicas y herramientas

4.1. Sistema Gestor de Base de Datos

A continuación, se muestra una pequeña comparación de los sistemas gestores de bases de datos planteados para su uso en este trabajo.

Gestores	SQLite	Elasticsearch	InfluxDB	MongoDB
Relacional	Si	No	No	No
SQL	Sí	No	No	Lectura
Modelo	RDBMS	Search engine	TSDBMS	Document Store
Open Source	Sí	Sí	Sí	Sí
Python	Sí	Sí	Sí	Sí
UDP	No	No	Sí	No
Visor	No	Kibana	Chronograf	Charts
Server Scripts	No	Sí	No	Sí
Docker	No	Sí	Sí	Sí
Linux	Sí	Sí	Sí	Sí

Tabla 4.1: Comparación gestores de bases de datos

Otro punto importante que no se ve reflejado en la tabla, es la disponibilidad de documentación. Todos estos sistemas son bastante conocidos y utilizados en sus respectivos campos, por lo que todos disponen de documentación completa y fácil de entender.

Al final, el sistema elegido ha sido InfluxDB. Como se ve en la tabla, InfluxDB es un Sistema Gestor de Bases de Datos de Series Temporales

(Time Series Data Base Management System, o TSDBMS). En estos tipos de sistemas, el tiempo tiene una especial importancia, pues cada dato tendrá un timestamp asociado. Esto viene muy bien para un sistema dedicado a guardar logs, pues cada mensaje recibido tendrá el timestamp del momento en el que se generó.

Aspectos relevantes del desarrollo del proyecto

Este apartado pretende recoger los aspectos más interesantes del desarrollo del proyecto, comentados por los autores del mismo. Debe incluir desde la exposición del ciclo de vida utilizado, hasta los detalles de mayor relevancia de las fases de análisis, diseño e implementación. Se busca que no sea una mera operación de copiar y pegar diagramas y extractos del código fuente, sino que realmente se justifiquen los caminos de solución que se han tomado, especialmente aquellos que no sean triviales. Puede ser el lugar más adecuado para documentar los aspectos más interesantes del diseño y de la implementación, con un mayor hincapié en aspectos tales como el tipo de arquitectura elegido, los índices de las tablas de la base de datos, normalización y desnormalización, distribución en ficheros³, reglas de negocio dentro de las bases de datos (EDVHV GH GDWRV DFWLYDV), aspectos de desarrollo relacionados con el WWW... Este apartado, debe convertirse en el resumen de la experiencia práctica del proyecto, y por sí mismo justifica que la memoria se convierta en un documento útil, fuente de referencia para los autores, los tutores y futuros alumnos.

Trabajos relacionados

Este apartado sería parecido a un estado del arte de una tesis o tesina. En un trabajo final grado no parece obligada su presencia, aunque se puede dejar a juicio del tutor el incluir un pequeño resumen comentado de los trabajos y proyectos ya realizados en el campo del proyecto en curso.

Conclusiones y Líneas de trabajo futuras

Todo proyecto debe incluir las conclusiones que se derivan de su desarrollo. Éstas pueden ser de diferente índole, dependiendo de la tipología del proyecto, pero normalmente van a estar presentes un conjunto de conclusiones relacionadas con los resultados del proyecto y un conjunto de conclusiones técnicas. Además, resulta muy útil realizar un informe crítico indicando cómo se puede mejorar el proyecto, o cómo se puede continuar trabajando en la línea del proyecto realizado.

Bibliografía

- [1] DB-Engines. Db-engines ranking of time series dbms, March 2023. [Internet; read on 22-march-2023].
- [2] InfluxData. Influxdb design principles. [Internet; read on 22-march-2023].
- [3] InfluxData. What is time series data? [Internet; read on 22-march-2023].
- [4] John R. Koza. *Genetic Programming: On the Programming of Computers by Means of Natural Selection*. MIT Press, 1992.
- [5] Wikipedia. Latex — wikipedia, la enciclopedia libre, 2015. [Internet; descargado 30-septiembre-2015].