

Universidades de Burgos, León y
Valladolid

Máster universitario

Inteligencia de Negocio y Big Data en Entornos Seguros



Trabajo Fin de Máster

**Algo muy largo y no sé que más
puedo hacer aquí**

Presentado por José Miguel Ramírez Sanz
en Universidad de Burgos — 26 de febrero
de 2020

Tutor: Dr. José Francisco Díez Pastor
Dr. Álgar Arnaiz Gonzalez

Universidades de Burgos, León y Valladolid



Máster universitario en Inteligencia de Negocio y Big Data en Entornos Seguros

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

Expone:

Que el alumno D. José Miguel Ramírez Sanz, con DNI 71303106R, ha realizado el Trabajo final de Máster en Inteligencia de Negocio y Big Data en Entornos Seguros titulado título de TFM.

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, 26 de febrero de 2020

Vº. Bº. del Tutor:

Vº. Bº. del co-tutor:

Dr. José Francisco Díez Pastor

Dr. Álgar Arnaiz Gonzalez

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
Memoria	1
1. Introducción	3
2. Objetivos del proyecto	5
3. Conceptos teóricos	7
3.1. Secciones	7
3.2. Referencias	7
3.3. Imágenes	8
3.4. Listas de items	8
3.5. Tablas	9
4. Técnicas y herramientas	11
4.1. Visión por computador ~ Detección de movimiento	11
5. Aspectos relevantes del desarrollo del proyecto	15
6. Trabajos relacionados	17
7. Conclusiones y Líneas de trabajo futuras	19

Apéndices	20
Apéndice A Plan de Proyecto Software	23
A.1. Introducción	23
A.2. Planificación temporal	23
A.3. Estudio de viabilidad	24
Apéndice B Especificación de Requisitos	25
B.1. Introducción	25
B.2. Objetivos generales	25
B.3. Catalogo de requisitos	25
B.4. Especificación de requisitos	25
Apéndice C Especificación de diseño	27
C.1. Introducción	27
C.2. Diseño de datos	27
C.3. Diseño procedimental	27
C.4. Diseño arquitectónico	27
Apéndice D Documentación técnica de programación	29
D.1. Introducción	29
D.2. Estructura de directorios	29
D.3. Manual del programador	29
D.4. Compilación, instalación y ejecución del proyecto	29
D.5. Pruebas del sistema	29
Apéndice E Documentación de usuario	31
E.1. Introducción	31
E.2. Requisitos de usuarios	31
E.3. Instalación	31
E.4. Manual del usuario	31
Bibliografía	33

Índice de figuras

3.1. Autómata para una expresión vacía	8
4.2. Logo de Detectron2.	12
4.3. Logo de TensorFlow.	12

Índice de tablas

3.1. Herramientas y tecnologías utilizadas en cada parte del proyecto	10
---	----

Memoria

Introducción

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

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_EX¹.

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 [3].

¹Créditos 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.

▪

3.5. Tablas

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

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

Técnicas y herramientas

En este apartado se van a comentar las distintas técnicas y herramientas utilizadas en el desarrollo del proyecto.

4.1. Visión por computador ~ Detección de movimiento

La visión por computador, también llamada visión artificial, es la parte de la ciencia de la computación orientada a la recogida y al tratamiento de imágenes y vídeos [?].

Por la tipología de los datos con los que se trabaja en la visión por computador el procesamiento de éstos es muy costoso, este elevado coste computacional es causado en gran parte por la calidad de las imágenes y en el caso de los vídeos, además de la calidad de la imagen, por la cantidad de fotogramas por segundo con el que se ha grabado éste. ¿AÑADIR AQUI QUE POR ESO SE HA USADO GAMMA (TENDRÍA QUE EXPLICAR ANTES QUE ES GAMMA)?

Una de las principales funciones que tiene la visión artificial es la detección de movimiento, la cual consiste en primero detectar la posición de cada una de las partes del cuerpo (depende del propio algoritmo la división que se quiera realizar sobre el cuerpo) para posteriormente realizar un seguimiento de estos elementos.

Existen distintos algoritmos de detección de movimiento, cada uno diferenciado por el uso de distintos lenguajes de programación y diferentes métodos de inteligencia artificial con los que entrenar el modelo. Como en este

proyecto se ha querido trabajar en *Python* se han investigado los siguientes algoritmos que están implementados en este lenguaje de programación.

Detectron

Detectron es un proyecto de inteligencia artificial de *Facebook* centrado en la detección de objetos y personas con el uso de algoritmos de *deep learning* [2]. El proyecto se apoya en una de las librerías de inteligencia artificial más usadas en *Python*, *PyTorch*.

El proyecto cuenta ya con una segunda versión llamada Detectron2, figura 4.2, en la cual se han mejorado y añadido más modelos [6].



Figura 4.2: Logo de Detectron2.

PoseNet

PoseNet es un proyecto de *Google* orientado unicamente al seguimiento del movimiento de las personas centrando su análisis en los puntos claves del cuerpo humano. El proyecto se basa en la librería de inteligencia artificial de la propia compañía *Google* llamada *TensorFlow*, figura 4.3 [1, 4].



Figura 4.3: Logo de TensorFlow.

4.1. VISION POR COMPUTADOR ~ DETECCIÓN DE MOVIMIENTO 13

TF-Pose-Estimator

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 de máster no parece tan 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.

Apéndice

Apéndice A

Plan de Proyecto Software

A.1. Introducción

La planificación de un proyecto es una fase esencial en desarrollo de éste, ya que permite comprobar y guiar el proyecto según las necesidades actuales y futuras.

Dentro de este plan para el desarrollo del proyecto se pueden diferenciar dos puntos sobre los que se puede enfocar este estudio:

- **Temporal:** enfoque en el que se analiza la evolución del proyecto en el tiempo. Al utilizar una metodología SCRUM, se ha dividido el desarrollo en *sprints* de 2 semanas.
- **Viabilidad:** comprobación de la adecuación del proyecto tanto desde el ámbito económico como desde el ámbito legal.

A.2. Planificación temporal

Como ya se ha comentado, la planificación temporal se ha dividido en distintos *sprints* de 2 semanas cada uno, en lo cuales se realizaron distintas tareas y reuniones. Cabe destacar que antes de realizar el primer *sprint* se desarrolló la aplicación FIS sobre la cuál será implementado en el futuro el proyecto desarrollado.

Desarrollo de la aplicación FIS

¿Esto lo documento mas?

***Sprint1* - 17/02/2020 ->26/02/2020**

Este es el primer *sprint* de desarrollo de la aplicación, en el se realizaron las primeras tareas de creación del repositorio en *GitHub* y la creación de la estructura del mismo. Además, se creó el documento *LaTeX* a partir de la plantilla de la asignatura. Las tareas más relevantes en este primer *sprint* fueron las relacionadas con la configuración de *Gamma* y los primeros pasos en la investigación de algoritmos de visión por computador y detección de movimiento, como son Detectron2 y PoseNet.

A.3. Estudio de viabilidad

Viabilidad económica

Viabilidad legal

Apéndice B

Especificación de Requisitos

- B.1. Introducción
- B.2. Objetivos generales
- B.3. Catalogo de requisitos
- B.4. Especificación de requisitos

Apéndice C

Especificación de diseño

- C.1. Introducción
- C.2. Diseño de datos
- C.3. Diseño procedimental
- C.4. Diseño arquitectónico

Apéndice D

Documentación técnica de programación

- D.1. Introducción
- D.2. Estructura de directorios
- D.3. Manual del programador
- D.4. Compilación, instalación y ejecución
del proyecto
- D.5. Pruebas del sistema

Apéndice E

Documentación de usuario

- E.1. Introducción
- E.2. Requisitos de usuarios
- E.3. Instalación
- E.4. Manual del usuario

Bibliografía

- [1] Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dandelion Mané, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. Software available from tensorflow.org.
- [2] Ross Girshick, Ilija Radosavovic, Georgia Gkioxari, Piotr Dollár, and Kaiming He. Detectron. <https://github.com/facebookresearch/detectron>, 2018.
- [3] John R. Koza. *Genetic Programming: On the Programming of Computers by Means of Natural Selection*. MIT Press, 1992.
- [4] TensorFlow. Pose estimation. https://www.tensorflow.org/lite/models/pose_estimation/overview, feb 2020.
- [5] Wikipedia. Latex — wikipedia, la enciclopedia libre, 2015. [Internet; descargado 30-septiembre-2015].
- [6] Yuxin Wu, Alexander Kirillov, Francisco Massa, Wan-Yen Lo, and Ross Girshick. Detectron2. <https://github.com/facebookresearch/detectron2>, 2019.