



Título del Trabajo Fin de Grado/Máster

Grado en Ingeniería Informática

Trabajo Fin de Grado

Autor:

Nombre Apellido1 Apellido2 (alumno)

Tutor/es:

Nombre Apellido1 Apellido2 (tutor1)

Nombre Apellido1 Apellido2 (tutor2)

28 de Mayo de 2019



**Facultad
Informática
Universidad
Murcia**

Título del Trabajo Fin de Grado/Máster

Subtítulo del proyecto

Autor

Nombre Apellido1 Apellido2 (alumno)

Tutor/es

Nombre Apellido1 Apellido2 (tutor1)

Departamento del tutor

Nombre Apellido1 Apellido2 (tutor2)

Departamento del cotutor



Grado en Ingeniería Informática



UNIVERSIDAD DE
MURCIA



Murcia, 28 de Mayo de 2019

Preámbulo

Poner aquí un texto breve que debe incluir entre otras:

las razones que han llevado a la realización del estudio, el tema, la finalidad y el alcance y también los agradecimientos por las ayudas, por ejemplo apoyo económico (becas y subvenciones) y las consultas y discusiones con los tutores y colegas de trabajo. [1]

Agradecimientos¹

Este trabajo no habría sido posible sin el apoyo y el estímulo de mi colega y amigo, Doctor Rudolf Fliesning, bajo cuya supervisión escogí este tema y comencé la tesis. Sr. Quentin Travers, mi consejero en las etapas finales del trabajo, también ha sido generosamente servicial, y me ha ayudado de numerosos modos, incluyendo el resumen del contenido de los documentos que no estaban disponibles para mi examen, y en particular por permitirme leer, en cuanto estuvieron disponibles, las copias de los recientes extractos de los diarios de campaña del Vigilante Rupert Giles y la actual Cazadora la señorita Buffy Summers, que se encontraron con William the Bloody en 1998, y por facilitarme el pleno acceso a los diarios de anteriores Vigilantes relevantes a la carrera de William the Bloody.

También me gustaría agradecerle al Consejo la concesión de Wyndham-Pryce como Compañero, el cual me ha apoyado durante mis dos años de investigación, y la concesión de dos subvenciones de viajes, una para estudiar documentos en los Archivos de Vigilantes sellados en Munich, y otra para la investigación en campaña en Praga. Me gustaría agradecer a Sr. Travers, otra vez, por facilitarme la acreditación de seguridad para el trabajo en los Archivos de Munich, y al Doctor Fliesning por su apoyo colegial y ayuda en ambos viajes de investigación.

No puedo terminar sin agradecer a mi familia, en cuyo estímulo constante y amor he confiado a lo largo de mis años en la Academia. Estoy agradecida también a los ejemplos de mis difuntos hermano, Desmond Chalmers, Vigilante en Entrenamiento, y padre, Albert Chalmers, Vigilante. Su coraje resuelto y convicción siempre me inspirarán, y espero seguir, a mi propio y pequeño modo, la noble misión por la que dieron sus vidas.

Es a ellos a quien dedico este trabajo.

¹Por si alguien tiene curiosidad, este “simpático” agradecimiento está tomado de la “Tesis de Lydia Chalmers” basada en el universo del programa de televisión Buffy, la Cazadora de Vampiros.<http://www.buffy-cazavampiros.com/Spiketesis/tesis.inicio.htm>

*A mi esposa Marganit, y a mis hijos Ella Rose y Daniel Adams,
sin los cuales habría podido acabar este libro dos años antes*²

²Dedicatoria de Joseph J. Roman en "An Introduction to Algebraic Topology"

*Si consigo ver más lejos
es porque he conseguido auparme
a hombros de gigantes*

Isaac Newton.

Declaración firmada sobre originalidad del trabajo

D./Dña. **Nombre Apellido1 Apellido2 (alumno)**, con DNI **00000000A**, estudiante de la titulación de **Grado en Ingeniería Informática** de la Universidad de Murcia y autor del TF titulado “**Título del Trabajo Fin de Grado/Máster**”.

De acuerdo con el Reglamento por el que se regulan los Trabajos Fin de Grado y de Fin de Máster en la Universidad de Murcia (aprobado C. de Gob. 30-04-2015, modificado 22-04-2016 y 28-09-2018), así como la normativa interna para la oferta, asignación, elaboración y defensa de los Trabajos Fin de Grado y Fin de Máster de las titulaciones impartidas en la Facultad de Informática de la Universidad de Murcia (aprobada en Junta de Facultad 27-11-2015)

DECLARO:

Que el Trabajo Fin de Grado presentado para su evaluación es original y de elaboración personal. Todas las fuentes utilizadas han sido debidamente citadas. Así mismo, declara que no incumple ningún contrato de confidencialidad, ni viola ningún derecho de propiedad intelectual e industrial

Murcia, a 28 de Mayo de 2019



Fdo.: **Nombre Apellido1 Apellido2 (alumno)**
Autor del TF

Resumen

Aqui ira el resumen en del TFG

Falta por hacer

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Aliquam elementum, lorem at aliquet convallis, ipsum est elementum orci, eget venenatis quam nisi et mauris. Nullam dignissim lectus nec ante efficitur semper. Aliquam sed nisl lectus. Donec vel pellentesque orci. Ut et tempus neque, vel semper arcu. Morbi varius tortor vel posuere bibendum. Integer et pharetra eros. Ut efficitur malesuada rhoncus. Cras ac quam hendrerit, tempor eros vel, condimentum sapien. Integer in velit facilisis, rutrum enim vel, varius ante.

Donec nec nulla diam. Integer sed nulla at elit posuere suscipit non quis nulla. Pellentesque sit amet dictum eros. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Sed dapibus pulvinar euismod. Donec ullamcorper cursus nunc quis ultrices. Cras ultrices dignissim justo at viverra. Suspendisse in felis enim. Mauris sodales vehicula laoreet. In suscipit dignissim augue eget laoreet. Sed sollicitudin fermentum turpis, ut malesuada nunc auctor nec. Nullam non justo cursus, dapibus nunc a, aliquet urna. Nulla semper tortor ut odio molestie tincidunt. Sed ut elit felis. Quisque turpis purus, suscipit vel gravida sit amet, laoreet sed massa.

In hac habitasse platea dictumst. Mauris sit amet pulvinar elit. Nam in neque gravida, imperdiet tortor sed, pellentesque diam. Quisque mattis orci ut felis aliquam malesuada. Ut a metus odio. Integer tincidunt, augue eu porttitor auctor, dolor tellus tincidunt dolor, cursus rhoncus ex leo in tellus. Donec est massa, consequat vitae nisi ut, posuere efficitur justo. Quisque non scelerisque purus. Nunc eu placerat magna. Sed facilisis metus vel pretium finibus. Sed eleifend bibendum dictum. Aenean odio odio, varius sit amet neque at, porttitor gravida mauris. In sit amet nisl magna.

Nulla metus nisi, faucibus volutpat vestibulum ut, efficitur a velit. Nam rhoncus dolor sit amet luctus pharetra. Nulla rutrum nibh at ullamcorper tincidunt. Cras et lectus ultrices, iaculis mi sit amet, lacinia nisi. Etiam enim neque, convallis ac sem a, accumsan pharetra purus. Integer quis odio libero. Mauris porttitor diam quis nisl posuere accumsan. Cras lorem tellus, condimentum at sem ac, iaculis elementum dolor. Vivamus in blandit tortor, vitae accumsan purus.

Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos himenaeos. Vivamus ornare urna a velit volutpat, vitae volutpat arcu semper. Nullam cursus ante sit amet purus consectetur, in aliquam nisi blandit. Etiam luctus maximus pharetra. Vivamus mattis, diam in faucibus tincidunt, lacus justo imperdiet massa, et bibendum lectus quam ut libero. Nulla et ante ac quam vehicula euismod et et felis.

Mauris ac nibh congue nulla commodo dapibus. In hac habitasse platea dictumst. Maecenas consectetur justo vel convallis rutrum. Suspendisse eget tortor arcu. Pellentesque tristique fringilla elit ac placerat. Phasellus commodo mauris a accumsan aliquam. Cras tincidunt rhoncus odio.

Extended Abstract

Resumen extendido en inglés bajo el título “Extended Abstract”. Este apartado se situará tras el apartado “Resumen” y tendrá una extensión mínima de 2000 palabras.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Mauris at ligula dolor. Cras sodales porttitor tellus, vel eleifend lectus porta vel. Cras dignissim nec ex at venenatis. Aliquam erat volutpat. Fusce aliquet bibendum mauris id convallis. Integer tempus maximus vehicula. Morbi eu lorem a nibh faucibus viverra. Proin tellus tellus, euismod in est id, ultricies blandit urna. Phasellus tincidunt nec massa a efficitur. Nulla non purus purus.

Sed vestibulum placerat malesuada. Quisque in libero nulla. Suspendisse rhoncus vitae ante ut pretium. Maecenas efficitur nisl non luctus eleifend. Nam convallis lobortis elit in pulvinar. Duis pellentesque dui ac iaculis bibendum. Etiam maximus viverra velit eu sollicitudin. Morbi pharetra, mi vel commodo tincidunt, urna dolor facilisis orci, et tristique odio nisl in ex.

Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Suspendisse vehicula non nibh vitae fringilla. Nulla fermentum dolor at rutrum pharetra. Aenean vel nulla lacus. Duis pharetra et metus et tempor. Nam eu lectus rutrum, mollis nulla at, consectetur orci. Nam auctor dictum iaculis. Integer quis urna nisi. Nunc at erat nibh. Sed a auctor nulla, quis aliquet odio. Vivamus enim mauris, ultricies auctor tellus in, aliquet placerat magna.

Etiam sagittis, purus in tincidunt commodo, erat ligula egestas massa, eu bibendum est velit interdum massa. Phasellus sapien purus, blandit non sem a, tristique cursus sapien. In elit velit, volutpat eu lorem vitae, gravida consequat erat. Sed eu lacinia quam. Duis sit amet urna ac nulla sollicitudin elementum. Nam blandit quam ac elit auctor tincidunt. Etiam dui nunc, blandit vitae purus in, commodo tincidunt ante. Vivamus viverra dui metus, vel dictum nisi congue a. Nam a dapibus mauris, eget feugiat enim. In hac habitasse platea dictumst. Etiam at mollis tortor. Aliquam erat volutpat. Aliquam molestie scelerisque tortor vel suscipit.

Cras sodales justo vitae ex egestas, sed ultricies metus suscipit. Donec sed est eget ex scelerisque pharetra. Donec scelerisque tempor mi eu malesuada. Interdum et malesuada fames ac ante ipsum primis in faucibus. Curabitur nisl ante, hendrerit sit amet tortor in, volutpat tincidunt lacus. Aenean rutrum volutpat velit et lobortis. Aliquam lorem magna, iaculis vel pellentesque ac, bibendum nec nibh. Proin dictum libero vel ante viverra, placerat pharetra augue tempor. Nunc vel erat sed felis lobortis tristique eu a odio. Morbi vel lorem nec eros gravida aliquet eget in risus. Aliquam in justo

Falta por hacer

volutpat, tempor dui et, convallis ex. Nulla id condimentum ipsum, nec hendrerit est. Praesent semper arcu sit amet tincidunt ultrices. Aenean quis cursus leo, ut pulvinar turpis.

Nulla vel ex sed sem consequat pretium. Duis lobortis rutrum mi, non efficitur tortor porta vitae. In hac habitasse platea dictumst. Aenean convallis felis a ante faucibus, a consectetur magna tempus. Praesent volutpat cursus elit, dictum mollis sapien ultrices sit amet. Nunc pharetra vestibulum mi eu iaculis. Proin commodo dui nisl, sit amet faucibus augue dignissim fermentum. Pellentesque ultricies elit sit amet eros interdum feugiat. Fusce urna nulla, iaculis et velit sed, euismod ultrices lorem. Nunc mi tortor, porttitor quis rutrum et, blandit vel nunc. Cras et enim faucibus, placerat justo vel, semper diam.

Suspendisse rutrum, nibh vel iaculis facilisis, diam quam sollicitudin neque, et fringilla nunc nisi ut diam. Duis porttitor arcu nulla, vitae viverra diam laoreet et. Nullam in ligula vel eros tincidunt egestas. Donec id magna sed risus pretium commodo. Fusce vulputate lectus eget ipsum porta accumsan. Nunc consequat, nisi eleifend auctor eleifend, ipsum metus semper dolor, eget scelerisque est libero ut turpis. Mauris facilisis a odio eget sodales. Quisque eget enim placerat, ornare ex vel, aliquam mi. Sed in lacinia leo. Vestibulum a egestas massa, faucibus dapibus enim. Pellentesque aliquet, tortor id dignissim tempus, orci quam mollis sem, sit amet dignissim odio urna in tellus. Cras non libero non quam tincidunt lobortis auctor in nulla.

Proin eu consectetur felis. Vivamus consequat neque ac diam viverra, sed venenatis risus mattis. Pellentesque cursus enim iaculis metus convallis dapibus. Suspendisse potenti. Praesent non metus porta tellus vehicula elementum. Curabitur fermentum erat eu consequat aliquet. Curabitur eget massa eu tortor vestibulum vulputate sit amet in nunc. Aenean fermentum sodales mauris at sodales. Nullam aliquet eros turpis, a posuere diam dignissim vel. Curabitur velit massa, sollicitudin et velit sed, condimentum dictum enim. Donec et dolor augue. Quisque vulputate scelerisque nunc.

Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos himenaeos. Aenean varius rhoncus quam, ut euismod magna pretium et. Aenean ultrices, enim id fermentum consequat, odio leo consectetur orci, et gravida nisi nisi nec felis. Pellentesque ipsum elit, sollicitudin ac metus non, aliquam sollicitudin odio. Phasellus ipsum ante, laoreet accumsan metus nec, ultricies faucibus nulla. Mauris a elementum ligula, vel cursus lectus. Sed id aliquet turpis, egestas hendrerit velit. Donec aliquet id dui a elementum. Duis facilisis fermentum sodales. Morbi eget placerat risus. Nulla facilisi. Etiam vestibulum massa et eros pulvinar gravida. Maecenas pellentesque ex eget enim congue sagittis.

Pellentesque sed dui laoreet, pretium massa sit amet, placerat quam. Nulla et enim in nulla ultrices rutrum id finibus ex. Integer tincidunt blandit nunc et tincidunt. Duis gravida hendrerit neque ut tincidunt. Interdum et malesuada fames ac ante ipsum primis in faucibus. Praesent dictum malesuada blandit. Praesent eget fermentum nisl. Nam in libero massa. Vivamus volutpat varius ante in fringilla. Sed eu mi risus. Donec cursus arcu quis quam congue tempus. Maecenas turpis nulla, rhoncus et mi ut,

accumsan vestibulum massa. Suspendisse aliquet ullamcorper metus ut dictum.

Nullam blandit quis tellus nec ullamcorper. Phasellus dapibus mauris sit amet lorem cursus, vitae imperdiet risus accumsan. Suspendisse mollis sollicitudin metus nec facilisis. Sed turpis nisl, posuere non purus ut, sagittis sodales erat. Nam sagittis sagittis quam quis ornare. Donec rhoncus turpis porta lacus congue dictum. Aliquam suscipit consectetur lobortis. Aliquam fringilla risus ut hendrerit aliquet. Nulla faucibus, ante vel congue volutpat, ex est facilisis odio, eget semper nibh elit in ipsum. Duis mollis, nulla a sollicitudin rutrum, ex diam efficitur massa, vitae varius est velit eu justo. Nunc sed neque nec libero semper cursus non vitae diam. Curabitur nec iaculis sem. Quisque ut dignissim urna.

Vivamus interdum vel turpis ac rutrum. Integer a metus ut odio consectetur facilisis et sed nulla. In et est ut mauris suscipit commodo non sit amet ex. Nulla facilisi. Sed id semper lectus. Fusce egestas dolor a scelerisque consequat. Nunc maximus mi eget tempor feugiat. Praesent mauris metus, congue ac interdum luctus, fermentum id ante. Nam porttitor dignissim leo et ultrices. Proin imperdiet lectus nisl, vel tristique lacus vestibulum ac. Vivamus urna turpis, sollicitudin ac dolor sit amet, tincidunt consequat lacus. Sed faucibus a ligula nec tincidunt.

Sed aliquet tincidunt nisl, nec pharetra elit molestie eget. Quisque justo ex, aliquet ut suscipit quis, dapibus sit amet diam. Mauris mollis egestas fringilla. Mauris ornare lorem id tellus consectetur malesuada. Suspendisse a orci nibh. Nullam placerat arcu in arcu cursus tempor. Aliquam non felis dolor. Sed luctus nisl at vehicula bibendum. Integer tellus ante, egestas eu sollicitudin ut, tristique vitae eros. Ut dolor diam, rhoncus vitae ante quis, tincidunt pretium dui. Morbi ipsum odio, viverra et tempor vel, dictum a justo. Cras aliquam quam sed justo bibendum feugiat. Quisque in nisl ut neque blandit accumsan imperdiet non leo.

Suspendisse laoreet nunc id ipsum gravida, sit amet euismod risus maximus. Cras pharetra ipsum et odio tempor tristique. Integer tempor erat eu malesuada imperdiet. Suspendisse scelerisque justo sit amet enim tempus dapibus. Etiam pellentesque maximus velit nec ullamcorper. Vestibulum quam nisi, accumsan porttitor magna sed, fermentum placerat enim. Ut iaculis egestas bibendum. Maecenas dapibus congue lectus laoreet sagittis. Fusce sit amet semper nunc. Donec varius mi tellus, quis cursus nulla aliquet id. Vivamus metus enim, fringilla quis sem sed, egestas vehicula purus. Fusce eget diam et purus tristique tincidunt id eget mauris. Aliquam laoreet consequat nibh ut varius. Morbi volutpat arcu sed pellentesque tempus. Curabitur semper, ipsum in lobortis rhoncus, nibh lacus posuere tellus, eget imperdiet nulla sem vel mauris.

Sed sed risus justo. Donec rutrum sagittis porttitor. Aenean aliquam rhoncus ligula. Proin ex dolor, ultricies in felis ut, porta eleifend justo. Aenean purus augue, viverra a massa nec, aliquam tempor magna. Integer a convallis lacus. Cras lacus sem, pellentesque eu ligula ut, pretium porttitor nisl. Sed volutpat lorem eget arcu gravida viverra. Nulla a scelerisque massa. Etiam ut massa dignissim, efficitur ligula quis, mollis dui. Nunc facilisis neque at dui fringilla, ac volutpat sapien ultrices. Vestibulum tristique mollis luctus. Vivamus eget dignissim odio, sit amet molestie ipsum. In ut est est. Nam

a accumsan tellus. Morbi maximus malesuada molestie.

In quis vestibulum nibh. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Praesent dolor purus, egestas at sapien vel, iaculis consecetur nulla. Vivamus non dictum urna, ac maximus tellus. Donec orci ipsum, volutpat ac nibh eget, hendrerit rutrum massa. Sed nulla quam, maximus a pharetra eget, sodales at ipsum. Nulla non arcu neque. Nam dapibus porttitor lacus in hendrerit. Donec at eros nunc. Suspendisse ut lorem ligula. Phasellus sapien mauris, tincidunt et dolor quis, euismod tincidunt leo. Nam aliquam augue purus, nec elementum est congue in. Morbi metus ex, fermentum vitae lorem sed, dignissim porttitor nibh. Ut gravida odio dui, eget iaculis est commodo vel. Aliquam fringilla elit ipsum, ut ornare lectus rhoncus a. Nulla luctus diam vitae lectus pretium varius.

Morbi hendrerit vestibulum metus eu rhoncus. Donec euismod diam ex, vitae ultrices sem consectetur et. Nam felis diam, efficitur in mauris vitae, ullamcorper scelerisque est. In fermentum velit dui, eget tincidunt risus elementum sed. Praesent eleifend imperdiet tellus. Vivamus eu dapibus urna. In egestas blandit metus id eleifend. Vestibulum massa dui, dictum sit amet ante vel, dictum auctor libero. Quisque laoreet, mi iaculis feugiat commodo, urna lectus pretium urna, quis ultrices ipsum erat eu turpis. Fusce aliquam libero in nisi viverra, vel ultrices arcu porttitor. Aliquam erat volutpat. Sed sagittis tellus nec libero ullamcorper tempor. Nullam accumsan nibh quis risus fermentum eleifend.

Curabitur leo elit, hendrerit semper purus sed, ullamcorper iaculis quam. Vivamus ac velit tristique, dapibus orci at, feugiat odio. Proin vel libero est. In hac habitasse platea dictumst. Suspendisse potenti. Fusce vestibulum elit in est accumsan pharetra. Maecenas ac fringilla lectus, at condimentum ipsum. Curabitur aliquet arcu ex. Nam fringilla interdum nibh ut vulputate.

Vivamus ut est nec elit volutpat ultrices. Proin ultrices viverra felis ut scelerisque. Suspendisse ut justo urna. Duis neque urna, congue eu velit vel, molestie rutrum enim. Ut pulvinar in justo vel dapibus. Duis at imperdiet augue, sed bibendum felis. Pellentesque venenatis nisl blandit arcu suscipit molestie. Curabitur tellus mauris, molestie tincidunt quam a, mattis porta tortor. Donec faucibus mauris id nisl elementum, at ornare metus vehicula. Phasellus vel ultrices quam. Praesent fermentum, dolor id gravida sagittis, nisl dui euismod dolor, quis tristique ex urna condimentum tellus. Etiam iaculis metus magna, ac porta velit rhoncus ut.

Sed tempus risus id elit fringilla suscipit. Suspendisse porta justo ut augue hendrerit faucibus. Donec fringilla leo vitae velit mattis tincidunt. Nam vestibulum erat tortor, vitae scelerisque sapien sodales sed. Interdum et malesuada fames ac ante ipsum primis in faucibus. Suspendisse fermentum orci nisi, id dapibus lectus ultrices eget. Nunc id magna lacinia, convallis augue sit amet, tempor ipsum. Donec rutrum maximus velit vel tincidunt. Donec blandit purus quis leo bibendum, nec ornare velit luctus. Cras molestie nunc ut est rutrum, at luctus justo porta. Quisque tristique nisi ac augue rutrum, id posuere quam vulputate. Mauris ultricies facilisis nisl sed dignissim. Nulla eleifend velit sed dui bibendum, ut consectetur libero posuere. Proin sit amet tempor

metus. Cras dignissim leo massa, non ultrices sem posuere a.

Sed vitae turpis nec dolor elementum sagittis vel quis eros. Nullam finibus maximus enim eu maximus. Aenean ac egestas risus. Aliquam rhoncus, elit quis vestibulum hendrerit, risus est feugiat ipsum, vitae egestas ipsum lorem vitae lacus. Maecenas vitae odio sit amet velit porta sodales. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Pellentesque quis aliquam urna.

Sed interdum sed orci eu tempor. Fusce imperdiet maximus euismod. Fusce ac portitor dolor, in placerat risus. In laoreet metus sed pulvinar cursus. Duis sollicitudin ex sit amet enim consequat tempus. Donec ipsum quam, tristique ut elit ut, laoreet aliquam augue. Suspendisse rhoncus massa eu ex vehicula sollicitudin vitae a ipsum. Quisque finibus quam vel nunc feugiat, sit amet posuere justo suscipit. Ut suscipit maximus ante, dictum tempus neque sodales vel. Aliquam erat volutpat.

Aliquam elementum rhoncus quam vel tempor. Vivamus eu erat eget tortor tincidunt congue nec semper lacus. Curabitur eu orci eu tellus rutrum euismod. Suspendisse maximus mi a magna semper tincidunt. Aliquam vitae odio id enim accumsan sodales. Suspendisse viverra libero vel lorem bibendum, nec varius sapien consequat. Donec ullamcorper pellentesque tellus, eu laoreet massa blandit sit amet.

Praesent eget purus at risus accumsan aliquet. In interdum sem non efficitur hendrerit. Nunc molestie convallis vulputate. Maecenas faucibus nec leo at tristique. Interdum et malesuada fames ac ante ipsum primis in faucibus. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Aliquam tincidunt faucibus nisl non ullamcorper.

Praesent quis enim quis mauris dapibus pharetra. Nam iaculis eros sit amet eleifend euismod. Integer bibendum leo in facilisis egestas. Proin cursus consequat velit, ut gravida turpis elementum non. Mauris semper pellentesque elementum. Pellentesque consequat, massa id volutpat cursus, dolor ligula feugiat ligula, in faucibus purus felis ut sem. Nunc vitae ultricies magna. Sed sit amet lectus velit. Duis scelerisque, quam hendrerit sollicitudin placerat, leo turpis sodales est, ut interdum leo tortor at mi. Nunc sit amet orci erat. Mauris leo justo, rutrum eu tortor a, tincidunt elementum sapien. Pellentesque euismod sapien eget dui eleifend, at pulvinar metus pretium. Sed dignissim erat sed quam iaculis mollis eget et augue. Etiam a arcu consectetur, tempus dui ac, finibus mauris.

Mauris placerat nunc aliquam dapibus convallis. Nulla hendrerit risus quis nisi sodales, ut euismod sapien faucibus. Vestibulum sollicitudin id quam id malesuada. Vivamus vestibulum, orci lacinia rutrum semper, ex urna semper justo, at faucibus mi felis ut augue. Nulla facilisi. Sed at.

Índice general

1. Introducción	1
1.1. Planteamiento del Problema	1
1.2. Objetivos	1
1.2.1. Objetivo General	1
1.2.2. Objetivos Específicos	1
2. Estado del arte	3
3. Marco Teórico	5
3.1. ¿Qué es la Raspberry?	5
3.2. ¿Qué es Arduino?	6
3.3. ¿Qué es un comunicador serial?	6
3.4. ¿Qué es un Protoboard?	8
3.5. Configuración de las comunicaciones serie entre Arduino y Python . . .	9
3.5.1. Configuración de Arduino con Python	9
3.5.2. Creación del Código	9
4. Diseño y resolución del trabajo realizado	11
4.1. Diagramas	12
4.1.1. Diagrama de Bloque	12
4.1.2. Diagrama Eléctrico	12
4.1.3. Diagrama Esquemático	13
4.1.4. Diagrama IML	13
4.2. Lista de Componentes	14
4.3. Mapa de Variables	14
4.4. Explicación del Código Fuente	15
4.5. Prerequisitos y configuración	15
5. Conclusiones y vías futuras	17
5.1. Aportaciones	17
5.2. Conclusiones	17
5.3. Recomendaciones	17
5.4. Manual de Usuario	17
5.5. Cronograma	18
6. Bibliografía	19

7. Anexos	21
Bibliografía	65
A. Anexo I	67
B. Páginas horizontales	69
C. Importar PDF	73

Índice de figuras

Índice de tablas

Índice de Códigos

1. Introducción

1.1. Planteamiento del Problema

¿De qué forma un comunicador serial permite enviar información mediante la creación de programas con Python?

1.2. Objetivos

1.2.1. Objetivo General

Crear aplicaciones "control de leds", para Arduino con Python en una Raspberry

1.2.2. Objetivos Específicos

- Realizar la configuración del Arduino en la Raspberry, para realizar una programación con Python
- Identificar el funcionamiento de los diferentes comandos en Python
- Realizar las conexiones entre Raspberry, Arduino y Protoboard para que se pueda ejecutar el comando de Python

2. Estado del arte

Las nuevas tecnologías de la información y las comunicaciones se han involucrado en todos los aspectos de la vida de las personas. Se encuentran inmersas (embebidas) tanto en los dispositivos médicos como en los electrodomésticos; en los equipos industriales como en los negocios. Los avances son muy significativos tanto en hardware como en software y han logrado proveer a los usuarios de elementos de tamaño y costo muy reducidos, de programación libre y de alta prestación de servicios.

Esta cartilla, con la presentación de los pasos necesarios para la instalación y configuración de dos placas con sistemas embebidos Raspberry Pi 3 y pcDuino, está dirigida a estudiantes de ingeniería mecatrónica, robótica o programación interesados en la utilización de las nuevas tecnologías de la información y las comunicaciones para desarrollar propuestas relacionadas con casas inteligentes (domótica), obtención de datos mediante sensores, Internet en la nube e Internet de las cosas. Ha sido desarrollada dentro del ambiente académico y de investigación formativa del Programa de Ingeniería Mecatrónica de la Universidad Piloto de Colombia, cuyos componentes principales son el Taller Mecatrónico, los semilleros de investigación y la ayuda a poblaciones vulnerables desde los espacios de la Proyección Social. A continuación, se presentan los pasos de instalación y configuración, seguidos de una aplicación de ejemplo de programación y funcionamiento para cada tarjetas, lo que dará a los lectores una información sencilla y completa, y les permitirá avanzar en el mundo de las aplicaciones para dar vía libre a su creatividad.

Los sistemas embebidos hacen referencia a una combinación de hardware y software de computador diseñada para realizar operaciones de cómputo, normalmente en tiempo real, y cumplir con una función específica; sus recursos son limitados, pero son muy útiles en la construcción de productos dedicados. Esta combinación puede ser reemplazada en muchos casos por un circuito integrado que realice la misma tarea, pero una de las ventajas de los sistemas embebidos es su flexibilidad, ya que a la hora de realizar alguna modificación resulta mucho más sencillo modificar unas líneas de código en el software que reemplazar todo el circuito integrado (Salas, 2015, p. 38).

Existen básicamente dos tipos de sistemas embebidos: los que corresponden a sistemas autónomos y que funcionan por sí solos, capaces de realizar varias funciones a la vez; y los que son parte de sistemas mayores, los cuales cumplen una funcionalidad.

Los miniPC son siempre una buena opción para disfrutar de toda la potencia de un ordenador, pero en un tamaño compacto. Pueden ser usados como proveedores de contenidos, conectarse a un televisor o como ordenadores personales. Entre los más populares se encuentra Raspberry Pi por su bajo costo y sus enormes posibilidades. Desde su lanzamiento en 2012 fue acogido por miles de entusiastas de los ordenadores de bajo costo que constituyen una comunidad que aporta valor al proyecto mediante innumerables aplicaciones que se pueden implementar con poco conocimiento.

La placa base de la Raspberry Pi aloja un chip Broadcom BCM2835, procesador ARM de hasta 1 GHz, GPU VideoCore IV y 512 Mb de memoria RAM. Este embebido se desarrolló en el Reino Unido por la Fundación Raspberry PI (Universidad de Cambridge) en el año 2011, con el objetivo de estimular la enseñanza de la informática en las escuelas, aunque su comercialización empezó hasta el año 2012. El software que utiliza es open source y su sistema operativo oficial es el Raspbian, sin embargo permite otros sistemas operativos, incluida una versión de Windows 10.

La Raspberry Pi no incluye un disco duro ni unidad de estado sólido, ya que usa una tarjeta SD para el almacenamiento permanente; tampoco incluye fuente de alimentación ni carcasa (Blog Historia de la Informática, 2013). No obstante, Raspberry Pi no es la única tarjeta que se encuentra en el mercado: pcDuino es una minicomputadora que puede correr Ubuntu o Android y, además, cuenta con herramientas compatibles con el Arduino. La pcDuino puede ser usada como una computadora y, como otros dispositivos de esta naturaleza, no tiene aceleración de gráficas por hardware, lo cual significa que podría haber afectación al reproducir videos o en el desempeño de gráficas 3D. Quizás la gran virtud de esta tarjeta/computadora es la relación estrecha que existe con los encabezados de Arduino, además que permite desarrollar proyectos y prototipos para distintas aplicaciones (Michelone, 2013).

3. Marco Teórico

3.1. ¿Qué es la Raspberry?

En marzo de 2018 la Raspberry Pi Foundation lanzaba el Raspberry Pi 3 Model B+. Este nuevo micro-ordenador ha supuesto un rediseño absoluto de la placa, manteniendo el mismo tamaño y la misma posición de los elementos que en el modelo Pi 3 (algo importante, ya que nos valen las mismas cajas), pero ha cambiado el procesador por otro más potente que funciona a 1.4 Ghz, y además elimina el cuello de botella de la conectividad incluyendo Bluetooth 4.2, BLE, Wi-Fi a doble banda 2.4 Ghz y 5 Ghz y, además, la tarjeta de red, Gigabit Ethernet, ya no está limitada a los 100 Mbps, sino que es capaz de alcanzar los 300 Mbps al funcionar sobre USB 2.0. Todas las especificaciones técnicas de este nuevo modelo de Raspberry Pi 3 son:

- CPU + GPU: Broadcom BCM2837B0, Cortex-A53 (ARMv8) 64-bit SoC @ 1.4GHz
- RAM: 1GB LPDDR2 SDRAM
- Wi-Fi + Bluetooth: 2.4GHz y 5GHz IEEE 802.11.b/g/n/ac, Bluetooth 4.2, BLE
- Ethernet: Gigabit Ethernet sobre USB 2.0 (300 Mbps)
- GPIO de 40 pines
- HDMI
- 4 puertos USB 2.0
- Puerto CSI para conectar una cámara
- Salida de audio estéreo y vídeo compuesto
- Micro-SD
- Power-over-Ethernet (PoE)

3.2. ¿Qué es Arduino?

Arduino es una plataforma de desarrollo basada en una placa electrónica de hardware libre que incorpora un microcontrolador re-programable y una serie de pines hembra, los que permiten establecer conexiones entre el microcontrolador y los diferentes sensores y actuadores de una manera muy sencilla (principalmente con cables dupont).

Una placa electrónica es una PCB (“Printed Circuit Board”, “Placa de Circuito Impreso” en español). Las PCBs superficies planas fabricadas en un material no conductor, la cual consta de distintas capas de material conductor. Una PCB es la forma más compacta y estable de construir un circuito electrónico. Así que la placa Arduino no es más que una PCB que implementa un determinado diseño de circuitería interna, de esta forma el usuario final no se debe preocupar por las conexiones eléctricas que necesita el microcontrolador para funcionar, y puede empezar directamente a desarrollar las diferentes aplicaciones electrónicas que necesite.

PCB de un Arduino UNO Cuando hablamos de “Arduino” deberíamos especificar el modelo concreto, ya que se han fabricado diferentes modelos de placas Arduino oficiales, cada una pensada con un propósito diferente y características variadas (como el tamaño físico, número de pines E/S, modelo del microcontrolador, etc). A pesar de las varias placas que existen todas pertenecen a la misma familia (microcontroladores AVR marca Atmel), esto significa que comparten la mayoría de sus características de software, como arquitectura, librerías y documentación.

3.3. ¿Qué es un comunicador serial?

La comunicación serial consiste en el envío de un bit de información de manera secuencial, ésto es, un bit a la vez y a un ritmo acordado entre el emisor y el receptor.

La comunicación serial en computadores ha seguido los estándares definidos en 1969 por el RS-232 (Recommended Standard 232) que establece niveles de voltaje, velocidad de transmisión de los datos, etc. Por ejemplo, este protocolo establece un nivel de -12v como un uno lógico y un nivel de voltaje de +12v como un cero lógico (por su parte, los microcontroladores emplean por lo general 5v como un uno lógico y 0v como un cero lógico).

Existen en la actualidad diferentes ejemplos de puertos que comunican información de manera serial (un bit a la vez). El conocido como “puerto serial” ha sido gradualmente reemplazado por el puerto USB (Universal Serial Bus) que permite mayor versatilidad en la conexión de múltiples dispositivos. Aunque en naturaleza serial, no suele referenciarse de esta manera ya que sigue sus propios estándares y no los establecidos por el RS-232.

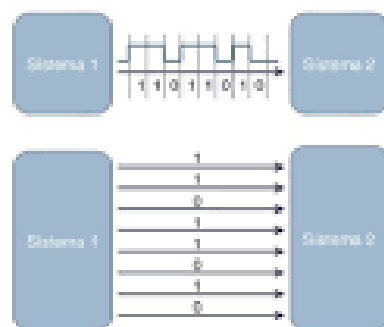
La mayoría de los microcontroladores, entre ellos Arduino, poseen un puerto de comunicación serial. Para comunicarse con los computadores personales actuales que poseen únicamente puerto USB requieren de un dispositivo “traductor”. Arduino emplea el integrado FT232R, el cual es un convertidor USB-Serial. A través de este integrado el microcontrolador puede recibir y enviar datos a un computador de manera serial.

En la comunicación con el computador Arduino emplea la comunicación asincrónica. Esto es, requiere de sólo dos líneas de conexión que corresponden con los pines 2 y 3: Pin 2 (Rx) pin de recepción y pin 3 (Tx) pin de transmisión, y del establecimiento de un nivel de tierra común con el computador, esto es, ambas tierras deben estar conectadas, estableciendo el mismo nivel de voltaje de referencia.

Además de realizar las conexiones físicas entre el microcontrolador y el computador, para que pueda establecerse la comunicación serial debe existir un acuerdo previo en la manera como van a ser enviados los datos. Este acuerdo debe incluir los niveles de voltaje que serán usados, el tamaño y formato de cada uno de los mensajes (número de bits que constituirán el tamaño de la palabra, existirá o no un bit de inicio y/o de parada, se empleará o no un bit de paridad), el tipo de lógica empleada (qué voltaje representará un cero o un uno), el orden en que serán enviados los datos (será enviado primero el bit de mayor peso o el de menor peso) y la velocidad de envío de datos.

Arduino facilita este proceso para que sólo sea necesario especificar la velocidad de envío de los datos. Esta velocidad es conocida como “baud rate” o rata de pulsos por segundo. Velocidades frecuentes de uso son 9600, 19200, 57600 y 115200.

Comunicación serial



3.4. ¿Qué es un Protoboard?

La Protoboard, llamada en inglés breadboard, es una placa de pruebas en la que se pueden insertar elementos electrónicos y cables con los que se arman circuitos sin la necesidad de soldar ninguno de los componentes. Las Protoboards tienen orificios conectados entre sí por medio de pequeñas laminas metálicas. Usualmente, estas placas siguen un arreglo en el que los orificios de una misma fila están conectados entre sí y los orificios en filas diferentes no. Los orificios de las placas normalmente están tienen una separación de 2.54 milímetros (0.1 pulgadas).

Una Protoboard es un instrumento que permite probar el diseño de un circuito sin la necesidad de soldar o desoldar componentes. Las conexiones en una Protoboard se hacen con solo insertar los componentes lo que permite armar y modificar circuitos con mayor velocidad. Normalmente estas placas son usadas para realizar pruebas experimentales. Si la prueba resulta satisfactoria el circuito se construye de una forma más permanente para evitar el riesgo de que algún componente pueda desconectarse. En caso de que la prueba no sea satisfactoria, puede modificarse el circuito fácilmente.

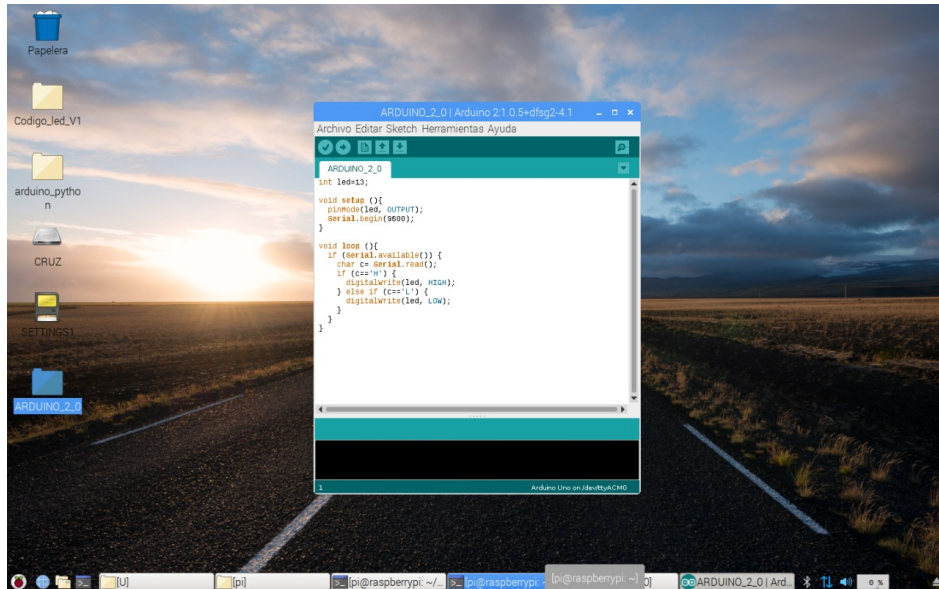
Las Protoboards tienen tres partes: el canal central, las pistas, y los buses. En el canal central, ubicado en la parte media, se conectan los circuitos integrados para mantener aislados los pines de ambos lados del circuito integrado. Los buses se encuentran en los lados de la Protoboard, y generalmente se emplean para conectar la tierra del circuito y su voltajes de alimentación. La mayoría de las veces los buses están indicados con franjas color negro o azul para indicar el bus de tierra, y con franjas color rojo para indicar el bus de voltaje positivo. El resto de los orificios de la Protoboard pertenecen a las pistas. Como se mencionó anteriormente, las pistas están separadas por filas. Las filas están indicadas con números y las columnas están indicadas con letras.

Las Protoboards presentan algunas ventajas y desventajas. Entre sus principales ventajas está que pueden utilizarse tantas veces como se requiera y que son de fácil manejo. Por otra parte, entre sus desventajas está el inconveniente de que en ocasiones puede haber falsos contactos, los cables empleados pueden tener mala conductividad o estar rotos, lo que hace que las conexiones no sean tan seguras como las de las pistas de un circuito impreso. Otra característica que hay que tomar en cuenta es que las Protoboards no están diseñadas para trabajar con componentes de gran potencia. La corriente con la que puede operar una Protoboard varía entre 3 y 5 A, y esto depende del fabricante. Suelen operar a bajas frecuencias, entre 10 – 20 MHz.

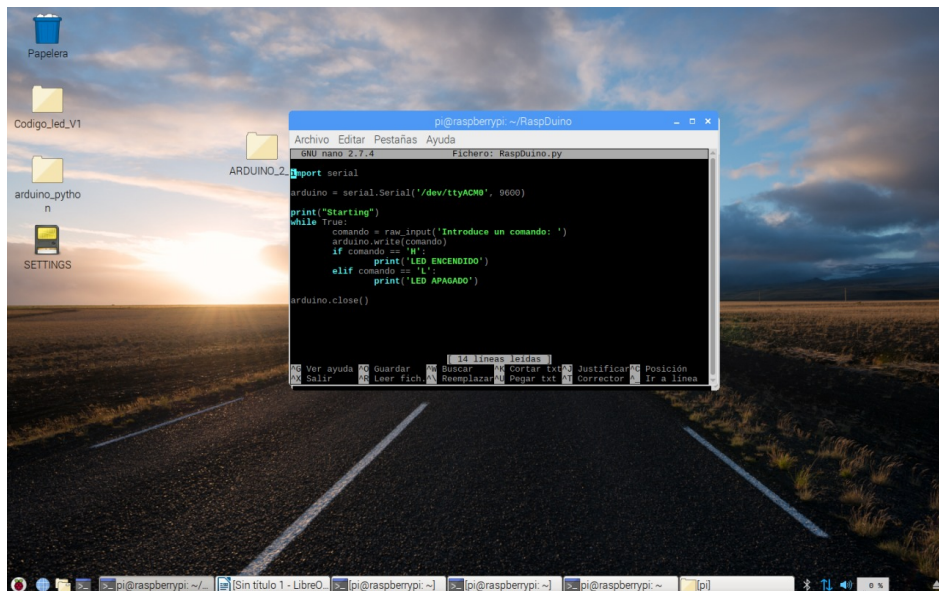
Es importante mencionar que en inglés, se entiende por Protoboard, una placa prefabricada para soldar componentes, en México se suele llamar a este tipo de componentes por el nombre de placas fenólicas perforadas.

3.5. Configuración de las comunicaciones serie entre Arduino y Python

3.5.1. Configuración de Arduino con Python



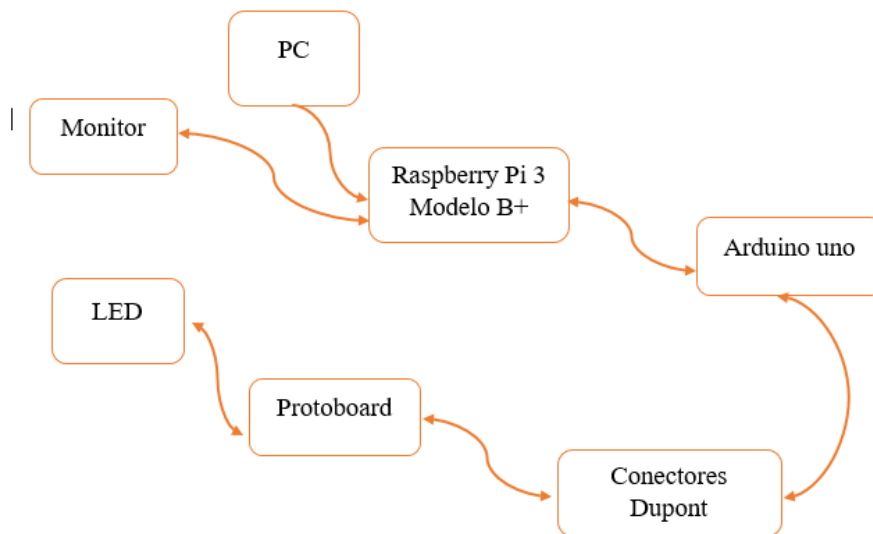
3.5.2. Creación del Código



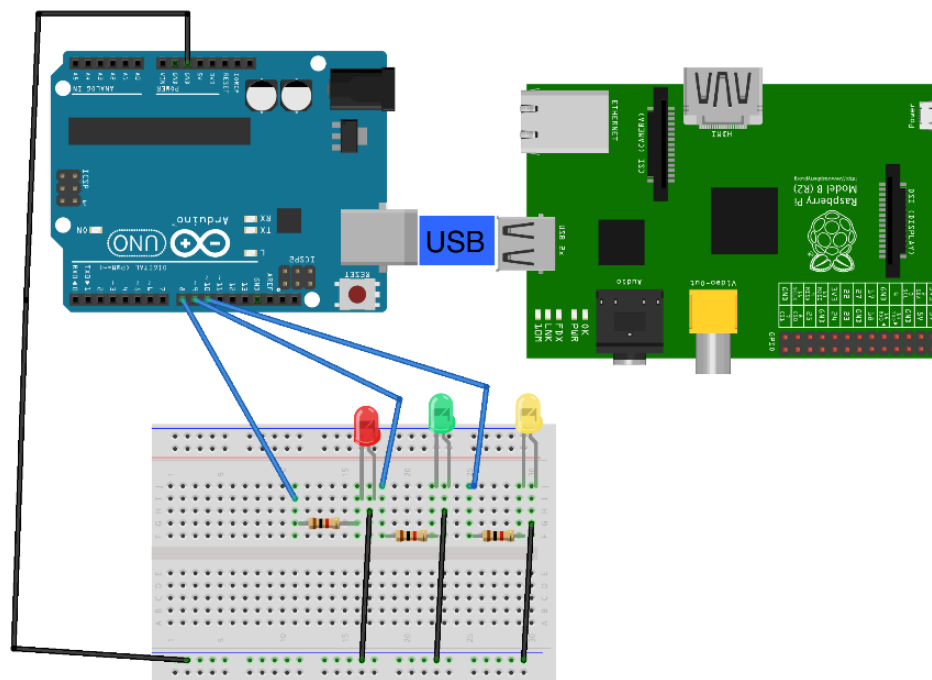
4. Diseño y resolución del trabajo realizado

4.1. Diagramas

4.1.1. Diagrama de Bloque



4.1.2. Diagrama Eléctrico

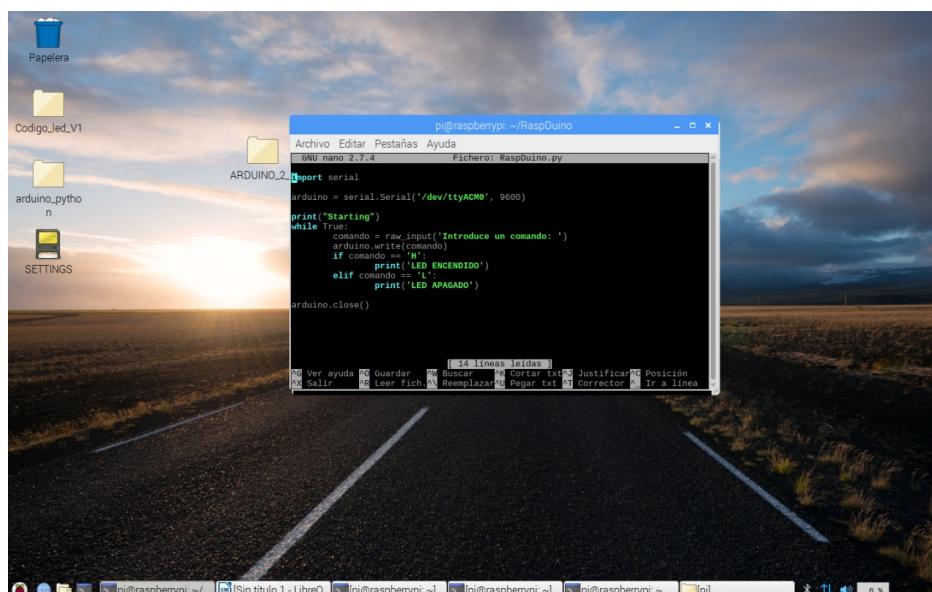


4.2. Lista de Componentes

- Raspberry Pi 3 Modelo B+
- Sistema Operativo Raspberry Pi
- Monitor
- Arduino Uno
- Protoboard
- Led
- Resistencias
- Conexiones DuPont
- Cable HDMI
- Fuente de 5v

4.3. Mapa de Variables

Las variables que se usan dentro de esta investigación están definidas en el programa Python que se ejecuta en la Raspberry.



4.4. Explicación del Código Fuente

A continuación se explicará el funcionamiento de los códigos dentro de la programación.

1. Import: este código permite importar las librerías de Arduino y poder empezar la programación en Python.
2. Print: este código sirve para presentar algún tipo de mensaje, que sirve como guía durante el desarrollo del programa.
3. While: Este bucle, se encarga de ejecutar una misma acción "mientras que" una determinada condición se cumpla
4. If-Elif-Else: estructuras de control de flujo condicionales, se definen mediante el uso de tres palabras claves reservadas, del lenguaje: if (si), elif (sino, si) y else (sino).

4.5. Prerequisitos y configuración

Para poder hacer la programación del Arduino con Python se necesitan tener las siguientes aplicaciones.

- Arduino IDE: esta aplicación nos permite hacer que la comunicación serial reciba datos desde Python. Descargar de la página de Arduino y Ejecutar
- PySerial-Master: Librería que permite los accesos de la Raspberry para poder realizar la configuración con Python. Descargar desde algún sitio web y posteriormente instalar.

Además debe tener el siguiente Hardware para que todo logre funcionar

- Raspberry Pi 3 Modelo B+
 - Monitor
 - Arduino Uno
 - Protoboard
 - Led
 - Resistencias
 - Conexiones DuPont
 - Fuente de 5v
-

5. Conclusiones y vías futuras

5.1. Aportaciones

La programación desde la Raspberry con Python necesita de varios componentes para poder ejecutarse, en esta investigación la aportación para poder solucionar este inconveniente es la instalación de las librerías "PySerial-Master".

Dentro del desarrollo de la investigación se menciona de manera poco frecuente a esta librería, pero su funcionamiento aplicación permitió que se pueda hacer una programación sencilla.

5.2. Conclusiones

La creación de aplicaciones para el control de LED's es posible gracias a la comunicación serial del Arduino mediante la implementación de la librería PySerial-Master, de esta forma se puede hacer programación con Python, el cual envía información al Arduino y este puede transmitirla de forma tal que, el Protoboard pueda entender la información y ejecutarla.

5.3. Recomendaciones

Para poder ejecutar los pasos que presenta la investigación, se recomienda:

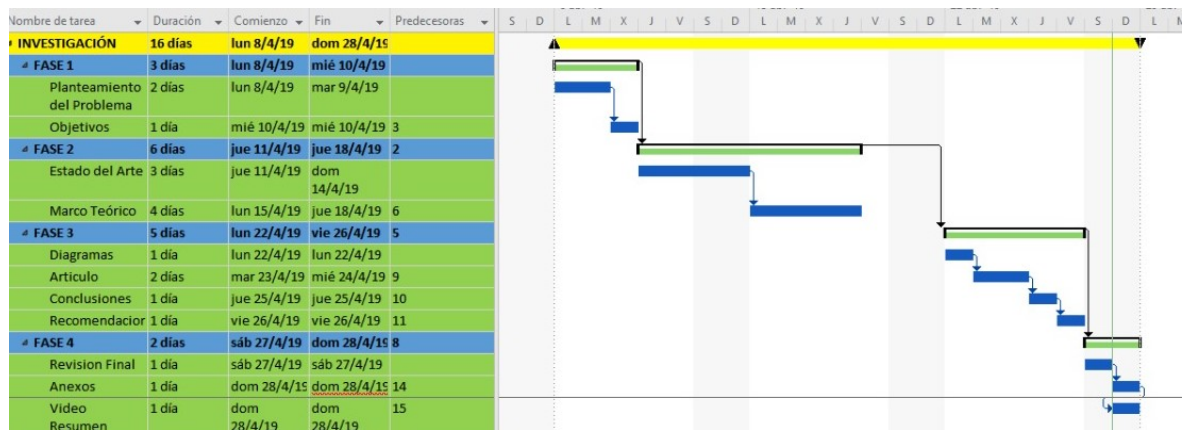
- Revisar que todo el Hardware (Raspberry, Arduino, Protoboard, Leds, conexiones) se encuentren en buen estado.
- Ubicar el Arduino y Raspberry en una zona donde no se concentre el calor
- Revisar de forma detallada el ingreso del código, la sintaxis debe estar correcta ya que de caso contrario no se podrá ejecutar la orden que se desea enviar

5.4. Manual de Usuario

1. Descargar el sistema operativo para la Raspberry Pi

2. Conectar la Raspberry al monitor
3. Descargar el Programa en la Raspberry Pi para Arduino
4. Programar en Python desde la Raspberry Pi un Led
5. Conectar la Raspberry Pi con el Arduino
6. Insertar la orden para el Arduino reciba la programación
7. Hacer la conexión desde el Arduino hacia la Protoboard
8. Enviar las ordenen para encender y apagar

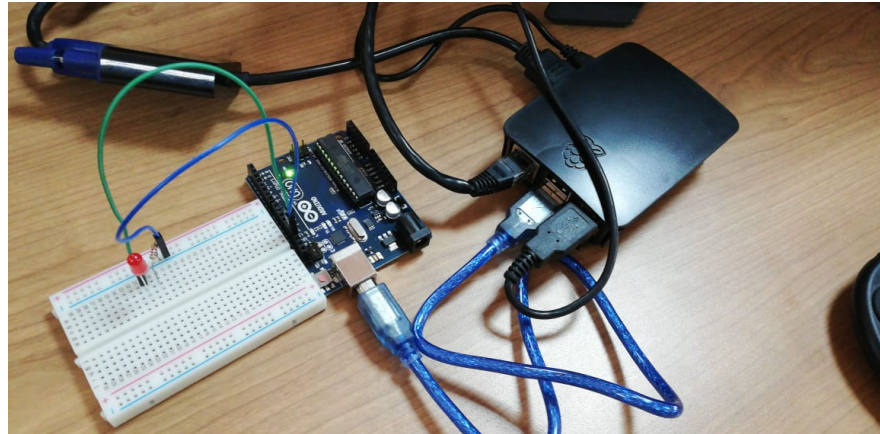
5.5. Cronograma



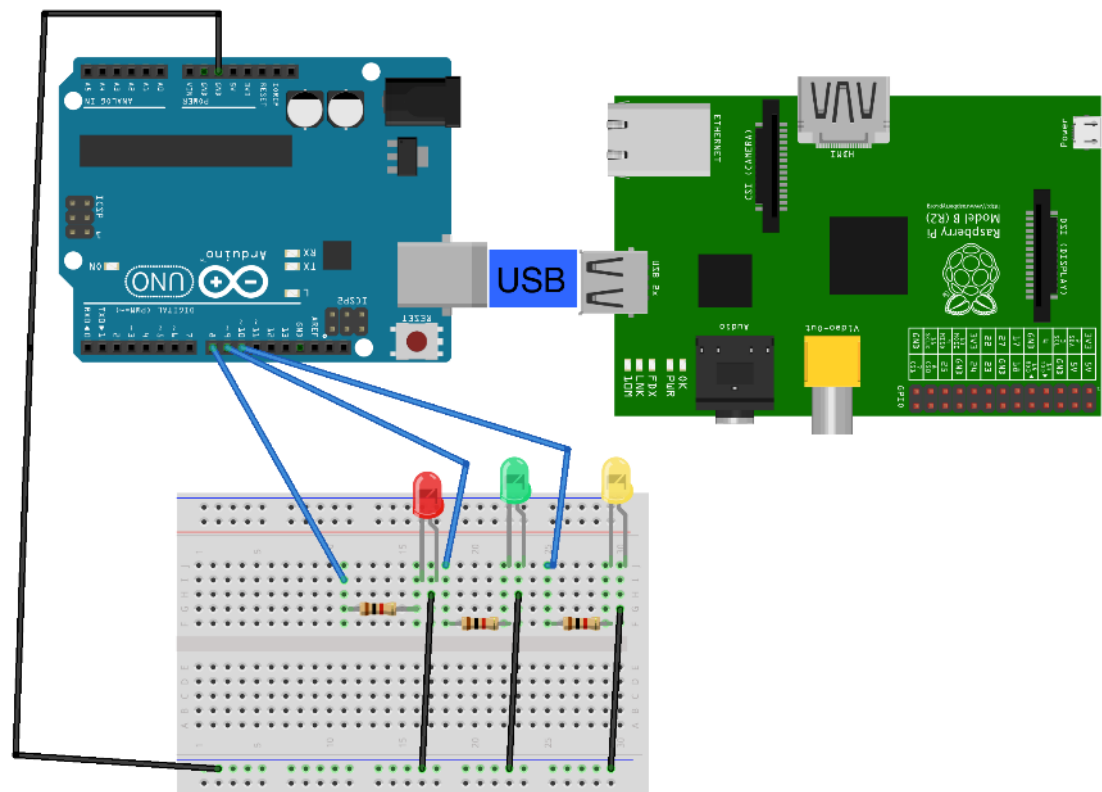
6. Bibliografía

- S.A. (2010). Download the Arduino IDE, recuperado de: <https://www.arduino.cc/en/Main/Software>
- Rubén E-Marmolejo. (2016). Arduino Serial Print en 5 minutos, recuperado de: <https://hetpro-store.com/TUTORIALES/arduino-serial-print/>
- 330ohms. (2016). Qué es un Protoboard, recuperado de: <https://blog.330ohms.com/2016/03/02/que-es-un-protoboard/>
- S.A. (2015). Análisis de Raspberry Pi3, recuperado de: <https://hardzone.es/reviews/perifericos/raspberry-pi-3-modelo-b/>
- Ariel Ortiz. (2014). Cómo programar a tu Arduino, recuperado de: <http://edupython.blogspot.com/2014/05/como-programar-tu-arduino.html>

7. Anexos

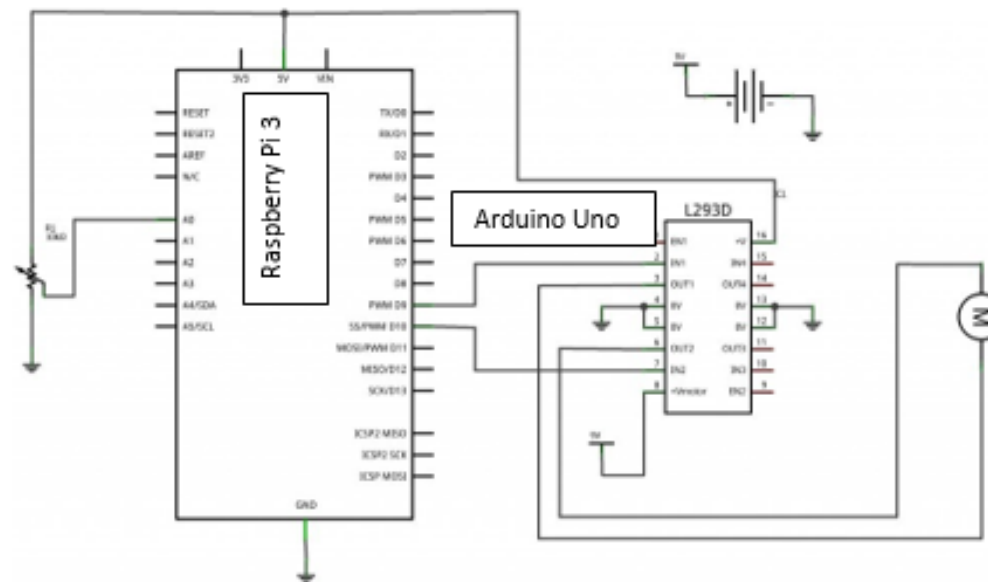


- Materiales



- Conexiones

fritzing



- Detalle de Conexión

6. Introducción (Con ejemplos de contenido)

Antes de comenzar la lectura de este documento debo agradecer el trabajo realizado por Pedro Pernías Peco en su plantilla de “tfg” que se puede ver en <https://github.com/lcg51/tfg>. Gracias a esa plantilla me he lanzado a crear mi versión. Algunos contenidos aquí mostrados han sido extraídos de la plantilla de Pedro.

Esta plantilla se ha diseñado de 0 y por ello no utiliza la misma estructura que la plantilla de Pedro. Pero la estructura de contenido para un TFG/TFM es la misma y a continuación se muestran las diferentes partes que debe tener un TFG/TFM redactado por Pedro.

6.1. ¡Importante!, leer primero

Este texto está escrito pensando en orientar a los alumnos que usarán \LaTeX para escribir su Trabajo Final de Grado (TFG) y Trabajo Final de Máster (TFM).

Contiene información útil para aquellos que no tengan experiencia previa en \LaTeX así como algunos datos acerca de cómo escribir mejor su TFG. A continuación, se ofrece una copia de la información que hay en el libro de estilo para la realización de los TFG de la EPS de la Universidad de Alicante.

En los capítulos siguientes encontrarás ejemplos de muchas de las cosas que se pueden realizar con \LaTeX . Con un poco de paciencia, estudia cómo se hacen estas cosas y luego aplícalas en tus documentos.

6.2. Estructura de un TFG

En caso de que el TFG/TFM tenga como finalidad la elaboración de un proyecto o un informe científico o técnico, deberá ajustarse a lo dispuesto en las normas UNE 157001:2002 y UNE 50135:1996 respectivamente.

Si el TFG/TFM tiene por finalidad la elaboración de un trabajo monográfico, el documento presentado deberá constar de las siguientes partes, teniendo como base la norma UNE 50136:1997.

Preámbulo: se describirán brevemente la motivación que ha originado la realización del TFG/TFM, así como una breve descripción de los objetivos generales que se quieren alcanzar con el trabajo presentado.

Agradecimientos: se podrán añadir las hojas necesarias para realizar los agradecimientos, a veces obligatorios, a las entidades y organismos colaboradores.

Dedicatoria: se podrá añadir una única hoja con dedicatorias, su alineación será derecha.

Citas: (frases célebres) se podrá añadir una única hoja con citas, su alineación será derecha.

Índices: cada índice debe comenzar en una nueva página, se incluirán los índices que se estimen necesarios (conforme UNE 50111:1989), en este orden:

Índice de contenidos: (obligatorio siempre) se incluirá un índice de las secciones de las que se componga el documento, la numeración de las divisiones y subdivisiones utilizarán cifras arábicas (según UNE 50132:1994) y harán mención a la página del documento donde se ubiquen.

Índice de figuras: si el documento incluye figuras se podrá incluir también un índice con su relación, indicando la página donde se ubiquen.

Índice de tablas: en caso de existir en el texto, ídem que el anterior.

Índice de abreviaturas, siglas, símbolos, etc.: en caso de ser necesarios se podrán incluir cada uno de ellos.

Cuerpo del documento: en el contenido del documento se da flexibilidad para su organización y se puede estructurar en las secciones que se considere. En todo caso obligatoriamente se deberá, al menos, incluir los siguientes contenidos:

Introducción: donde se hará énfasis a la importancia de la temática, su vigencia y actualidad; se planteará el problema a investigar, así como el propósito o finalidad de la investigación.

Marco teórico o Estado del arte: se hará mención a los elementos conceptuales que sirven de base para la investigación, estudios previos relacionados con el problema planteado, etc.

Objetivos: se establecerán el objetivo general y los específicos.

Metodología: se indicarán el tipo o tipos de investigación, las técnicas y los procedimientos que serán utilizados para llevarla a cabo; se identificarán la población y el tamaño de la muestra así como las técnicas e instrumentos de recolección de datos.

Resultados: incluirá los resultados de la investigación o trabajo, así como el análisis y la discusión de los mismos.

Conclusiones: obligatoriamente se incluirá una sección de conclusiones donde se realizará un resumen de los objetivos conseguidos así como de los resultados obtenidos si proceden.

Bibliografía y referencias: se incluirá también la relación de obras y materiales consultados y empleados en la elaboración de la memoria del TFG/TFM. La bibliografía y las referencias serán indexadas en orden alfabético (sistema nombre y fecha) o se numerará correlativamente según aparezca (sistema numérico). Se empleará la familia 1 como tipo de letra. Podrá utilizarse cualquier sistema bibliográfico normalizado predominante en la rama de conocimiento, estableciéndose como prioritarios el sistema ISO 690, sistema American Psychological Association (APA) o Harvard (no necesariamente en ese orden de preferencia). En esta plantilla Latex se propone usar el estilo APA indicándolo en la línea correspondiente como

```
\bibliographystyle{unsrtnat}
```

Anexos: se podrán incluir los anexos que se consideren oportunos.

6.3. Apartados dentro de los capítulos

En \LaTeX existen diferentes niveles de títulos para realizar secciones, subsecciones, etc. En esta web puedes ver más información al respecto https://en.wikibooks.org/wiki/LaTeX/Document_Structure

Para ello se utilizan los siguientes comandos;

```
\section{Esto es una sección}
Y este el contenido de la sección.
\subsection{Esto es una subsección}
Y este el contenido de la subsección.
\subsubsection{Esto es una subsubsección}
Y este el contenido de la subsubsección.
\paragraph{Esto es un paragraph}
Y este el contenido del paragraph. Que siempre se inicia en la misma línea que el título del mismo.
```

Y se genera lo siguiente:

6.4. Esto es una sección

Y este el contenido de la sección.

6.4.1. Esto es una subsección

Y este el contenido de la subsección.

Esto es una subsubsección

Y este el contenido de la subsubsección.

Esto es un paragraph Y este el contenido del paragraph. Que siempre se inicia en la misma línea que el título del mismo.

6.5. Citar bibliografía

Para citar la bibliografía tal como se define en el sistema APA (en esta web se indica como debe aparecer en el texto la cita: <http://guides.libraries.psu.edu/apasquickguide/intext>) se debe realizar con alguno de los comandos mostrados a continuación:

Esto es una cita estándar: `\citet{Shaw1996}`, que también puedes mostrar con paréntesis así: `\citep{Shaw↵↵ 1996}`. También se puede realizar una cita indicando a qué parte te refieres `\citep[ver][Cap. 2]{Shaw↵↵ 1996}` o `\cite[Cap. 2]{Shaw1996}` o `\citep[ver]{}{Shaw1996}`.

También puedes mostrar todos los autores cuando hay más de 2 autores añadiendo un asterisco después del ↵↵ comando como: `\citet*{Akyildiz2005}`, sin el asterisco quedaría así: `\citet{Akyildiz2005}`.

O puedes citar dos o más fuentes al mismo tiempo: `\citep{Barkan1995,Leighton2012}`
`\cite{UNE50136:97}`

Y \LaTeX genera lo siguiente:

Esto es una cita estándar: Shaw and Garlan [2], que también puedes mostrar con paréntesis así: [2]. También se puede realizar una cita indicando a qué parte te refieres [ver 2, Cap. 2] o [2, Cap. 2] o [ver 2].

También puedes mostrar todos los autores cuando hay más de 2 autores añadiendo un asterisco después del comando como: Akyildiz, Pompili, and Melodia [3], sin el asterisco quedaría así: Akyildiz et al. [3].

O puedes citar dos o más fuentes al mismo tiempo: [4, 5]

6.6. Notas a pie de página

Para introducir notas a pie de página se debe escribir lo siguiente:

La plantilla necesita el motor XeLaTeX `\footnote{Para más información sobre XeLaTeX visita \url{↵↵ https://es.sharelatex.com/learn/XeLaTeX}}` (el más recomendable actualmente), por lo que si el ↵↵ programa que utilizas compila la plantilla con el motor pdfLaTeX `\footnote{También puedes ↵↵ buscar más información en internet}` (el más habitual pero menos potente) debes cambiarlo por ↵↵ XeLaTeX en las opciones del programa. Si no sabes como hacerlo busca en el manual del ↵↵ programa o en google.

L^AT_EX genera lo siguiente (observa las notas a pie de página):

La plantilla necesita el motor XeLaTeX¹ (el más recomendable actualmente), por lo que si el programa que utilizas compila la plantilla con el motor pdfLaTeX² (el más habitual pero menos potente) debes cambiarlo por XeLaTeX en las opciones del programa. Si no sabes como hacerlo busca en el manual del programa o en google.

6.7. Estilos de texto

A continuación se muestran ejemplos de distintos estilos de texto:

- `\textit{Cursiva}` → *Cursiva*
- `\emph{Cursiva 2}` → *Cursiva 2*
- `\textbf{Negrita}` → **Negrita**
- `\texttt{Monoespacio}` → `Monoespacio`
- `\textsc{Mayúsculas capitales}` → MAYÚSCULAS CAPITALS
- `\uppercase{Todo mayúsculas}` → TODO MAYÚSCULAS

6.8. Acrónimos

Ahora vamos a ver cómo se ponen los acrónimos.

La norma dice que la primera vez que aparece un acrónimo debe ponerse su fórmula completa, es decir lo que significa, al lado del acrónimo. Después de ello, podemos usar sólo el acrónimo salvo cuando consideremos que debemos volver a usar la fórmula completa por alguna razón de legibilidad.

¿Cómo llevar la cuenta de cuándo es la primera vez que ponemos el acrónimo? si hacemos cambios en el doc es fácil que perdamos esa información así que lo mejor es que sea el propio L^AT_EX el que lleve esa cuenta. Para ello tenemos que hacer dos cosas:

Primero: creamos la entrada del acrónimo en el fichero `acronimos.tex`. Revisa los comentarios de su cabecera para saber cómo crear esa entrada. Básicamente lo que hacemos allí es poner la “fórmula corta” y la “fórmula larga” del acrónimo es decir, el propio acrónimo y su significado

Segundo: escribimos en el texto el acrónimo SIEMPRE diciendo que es un acrónimo y el tipo de fórmula que queremos usar. Por ejemplo, si siempre que queremos hacer referencia al IEEE escribimos

¹Para más información sobre XeLaTeX visita <https://es.sharelatex.com/learn/XeLaTeX>

²También puedes buscar más información en internet

```
\gls{ieeee}
```

se consigue que la primera vez que aparezca el acrónimo ponga las fórmulas larga y corta y en las siguientes ocasiones sólo aparecerá la corta.

Aquí va un ejemplo:

Si escribimos:

El `\gls{ieeee}` es una institución muy importante en el mundo de la ingeniería. El `\gls{ieeee}` lleva marcando normas y protocolos desde hace mucho tiempo. Pero el `\gls{ieeee}` no está solo en esta tarea. Además del `\gls{ieeee}` hay muchas otras instituciones para ello.

Obtendremos:

El Institute of Electrical and Electronics Engineers (IEEE) es una institución muy importante en el mundo de la ingeniería. El IEEE lleva marcando normas y protocolos desde hace mucho tiempo. Pero el IEEE no está solo en esta tarea. Además del IEEE hay muchas otras instituciones para ello.

6.9. Tareas por hacer

En esta plantilla se ha incluido un paquete para incluir notas/comentarios en el texto para recordar partes que hay que revisar o terminar de desarrollar. El uso es sencillo, el manual para conocer todos los comandos se encuentra en <http://osl.ugr.es/CTAN/macros/latex/contrib/todonotes/todonotes.pdf>, a continuación se muestran algunos ejemplos:

Para incluir un comentario sobre el texto:

Recomiendo utilizar programas LaTeX que permitan trabajar con sistema de archivos para poder editar el `\↔` conjunto de capítulos en la misma ventana. Este tipo de función lo tienen programas como `\↔` `\todo`TexStudio, es multiplataforma. `\todo`{Incluir más ejemplos de programas}

LaTeX genera lo siguiente:

Recomiendo utilizar programas LaTeX que permitan trabajar con sistema de archivos para poder editar el conjunto de capítulos en la misma ventana. Este tipo de función lo tienen programas como TexStudio, es multiplataforma.

Para incluir un comentario sobre el texto pero dentro del texto:

Recomiendo utilizar programas LaTeX que permitan trabajar con sistema de archivos para poder editar el `\↔` conjunto de capítulos en la misma ventana. Este tipo de función lo tienen programas como `\↔` `\todo`TexStudio, es multiplataforma. `\todo`[inline]{Incluir más ejemplos de programas}

L^AT_EX genera lo siguiente:

Recomiendo utilizar programas LaTeX que permitan trabajar con sistema de archivos para poder editar el conjunto de capítulos en la misma ventana. Este tipo de función lo tienen programas como TexStudio, es multiplataforma.

Incluir más ejemplos de programas

También se puede dejar indicado donde falta una imagen o figura, para incluirla más adelante del siguiente modo:

```
\missingfigure{Añadir gráfica de rendimiento}
```

L^AT_EX genera lo siguiente:



7. Marco Teórico (Con ejemplos de listas)

7.1. Listas

Hacer una lista es simple en \LaTeX . Para ello has de crear un entorno (así se llama) `itemize` con

```
\begin{itemize}  
...  
\end{itemize}
```

Y dentro de esa estructura, añadir cada elemento de la lista precedido de

```
\item primer ítem de lista  
\item segundo ítem de lista  
...  
\item ultimo ítem de lista
```

Es importante que revises este texto tal como aparece en la plantilla y relaciones el aspecto que tiene el PDF final con cómo está escrito el documento \LaTeX .

Aquí va una lista con subtérminos:

```
\begin{itemize}  
\item Ingeniería Informática.  
\item Ingeniería Sonido e Imagen en Telecomunicación.  
\item Ingeniería Multimedia.  
    \subitem Mención: Creación y ocio digital.  
    \subitem Mención: Gestión de Contenidos.  
\end{itemize}
```

El resultado es el siguiente:

- Ingeniería Informática.
- Ingeniería Sonido e Imagen en Telecomunicación.
- Ingeniería Multimedia.
 - Mención: Creación y ocio digital.
 - Mención: Gestión de Contenidos.

Aquí va una lista con subtérminos pero numerada:

```
\begin{enumerate}
  \item Ingeniería Informática.
  \item Ingeniería Sonido e Imagen en Telecomunicación.
  \item Ingeniería Multimedia.
  \begin{enumerate}
    \item Mención: Creación y ocio digital.
    \item Mención: Gestión de Contenidos.
  \end{enumerate}
\end{enumerate}
```

El resultado es el siguiente:

1. Ingeniería Informática.
2. Ingeniería Sonido e Imagen en Telecomunicación.
3. Ingeniería Multimedia.
 - a) Mención: Creación y ocio digital.
 - b) Mención: Gestión de Contenidos.

7.2. Listas de definición

Puedes realizar una lista de conceptos con su definición del siguiente modo:

```
\begin{description} % Inicio de la lista
  \item[MAPP XT:] Programa desarrollado por \textit{Meyer Sound} para el diseño y ajuste de sistemas ↵
    ↵ formados por altavoces de su marca.
  \begin{description} % Realiza una lista dentro de la lista
    \item[Ventajas:]~
      El programa permite realizar múltiples ajustes tal como se podría realizar en la realidad con un ↵
      ↵ procesador real.

      Permite analizar la fase recibida en cualquier punto y compararla con otras mediciones.

      Dispone de varios tipos de filtros, inversiones de fase, etc.
    \item[Inconvenientes:]~
      No existe una lista global de los altavoces ubicados en el plano, por lo tanto solo se pueden editar ↵
      ↵ seleccionándolos sobre el plano.

      Sólo permite diseñar en 2 dimensiones, principalmente sobre la vista lateral ya que los array de ↵
      ↵ altavoces no permite voltearlos.
  \end{description}
\end{description}
```

Y L^AT_EX genera lo siguiente:

MAPP XT: Programa desarrollado por *Meyer Sound* para el diseño y ajuste de sistemas formados por altavoces de su marca.

Ventajas: El programa permite realizar múltiples ajustes tal como se podría realizar en la realidad con un procesador real.

Permite analizar la fase recibida en cualquier punto y compararla con otras mediciones.

Dispone de varios tipos de filtros, inversiones de fase, etc.

Inconvenientes: No existe una lista global de los altavoces ubicados en el plano, por lo tanto solo se pueden editar seleccionándolos sobre el plano.

Sólo permite diseñar en 2 dimensiones, principalmente sobre la vista lateral ya que los array de altavoces no permite voltearlos.

8. Objetivos (Con ejemplos de tablas)

8.1. Tablas

Ahora veremos otra estructura más: las tablas.

Aquí va una tabla¹ para que se vea cómo insertar una tabla simple dentro del documento.

```
\begin{table}[h]
\centering
\begin{tabular}{llll}
& & & \\
& & & \\
& & & \\
\hline
& & & \\
& & & \\
& & & \\
& & & \\
\hline
\end{tabular}
\caption{Ejemplo de tabla.}
\label{tabladeejemplo}
\end{table}
```

	columna A	columna B	columna C
fila 1	fila 1, columna A	fila 1, columna B	fila 1, columna C
fila 2	fila 2, columna A	fila 2, columna B	fila 2, columna C
fila 3	fila 3, columna A	fila 3, columna B	fila 3, columna C

Tabla 8.1: Ejemplo de tabla.

L^AT_EX usa un sistema de parámetros para “decorar” las tablas. Puedes consultar estos parámetros en la tabla 8.2 de la página 40. La tabla se ubicará donde, a juicio de L^AT_EX, menos moleste por lo que puede no aparecer necesariamente donde se ha insertado en el texto original.

Existe la posibilidad de forzar que las tablas, figuras u otros objetos aparezcan en la zona del texto que se desea aunque en ocasiones puede dejar grandes espacios en blanco. El comando a utilizar es:

```
\FloatBarrier
```

Que introducido justo después de una tabla, figura, etc (después del comando `\end{...}`)

¹En <http://www.tablesgenerator.com/> se puede encontrar un generador On-Line de tablas para L^AT_EX

fuerza la aparición en el texto, empujando el contenido.

Parámetro	Significado
h	Situa el elemento flotante <i>preferentemente</i> (es decir, si es posible) en la situación exacta donde se incluye este
t	Sitúa el elemento en la parte de arriba de la página
b	Sitúa el elemento en la parte de abajo de la página
p	Sitúa el elemento en una página aparte dedicada sólo a elementos flotantes; en el caso del formato article , ésta se sitúa al final del documento, mientras que para el book es colocada al final de cada capítulo

Tabla 8.2: Parámetros optativos de los entornos flotantes

También es posible elegir el ancho de cada columna y la orientación del texto en cada una. Por ejemplo:

```
\begin{table}[ht]
\centering
\begin{tabular}{|C{2cm}|C{2cm}|C{2cm}|C{2cm}|} % 4 columnas de 2cm — texto centrado y con ↔
↔ bordes
\hline
\multicolumn{4}{|c|}{\textbf{\begin{tabular}{c}{@{}c@{}}FUENTE: TRÁFICO RODADO\\
↔ HORARIO: TARDE\end{tabular}}}\ \hline
\textbf{dB(A)} & \textbf{Población expuesta tarde} & \textbf{\%} & \textbf{\scriptsize↔
↔ CENTENAS}} \ \hline
\textbf{\textgreater70} & 0 & 0,000 & 0 \ \hline
\textbf{65 — 70} & 348,9 & 9,792 & 3 \ \hline
\textbf{60 — 65} & 1594,7 & 44,757 & 16 \ \hline
\textbf{55 — 60} & 322,1 & 9,040 & 3 \ \hline
\textbf{50 — 55} & 0 & 0,000 & 0 \ \hline
\textbf{\textgreater50} & 1297,3 & 36,410 & 13 \ \hline
\textbf{TOTAL} & 3563 & 100 & 35 \ \hline
\end{tabular}
\label{my-label}
\end{table}
```

L^AT_EX genera esto:

Donde `C{2cm}` indica que la columna tiene el texto centrado y un ancho de 2 cm. También se puede utilizar `L{}` o `R{}` para poner el texto a la izquierda o derecha y definir un ancho concreto.

Páginas como <https://www.tablesgenerator.com/> ayudan a realizar tablas fácilmente, es lo más recomendado, ahorra mucho tiempo de trabajo y luego si falta algún detalle se puede retocar en el documento.

El formato estándar de las columnas es `c`, `l` o `r`, así lo genera la web mencionada antes, pero una vez generada puedes cambiar ese formato por el definido anteriormente para ajustar el ancho de las columnas, o mantenerlo así si el resultado ya es el deseado.

Para conocer más sobre las tablas puedes leer manuales como este: <https://latexlive.files.wordpress.com/2009/04/tablas.pdf> que contiene muchos ejemplos y expli-

FUENTE: TRÁFICO RODADO HORARIO: TARDE			
dB(A)	Pobla- ción expuesta tarde	%	CENTENAS
>70	0	0,000	0
65 - 70	348,9	9,792	3
60 - 65	1594,7	44,757	16
55 - 60	322,1	9,040	3
50 - 55	0	0,000	0
>50	1297,3	36,410	13
TOTAL	3563	100	35

caciones.

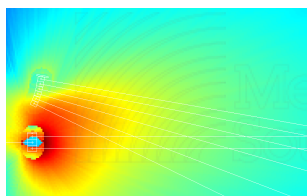
9. Metodología (Con ejemplos de figuras)

9.1. Inserción de figuras

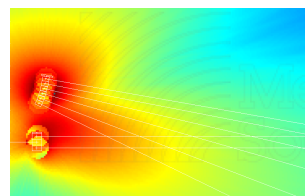
Las figuras son un caso un poco especial ya que \LaTeX busca el mejor lugar para ponerlas, no siendo necesariamente el lugar donde está la referencia. Por ello es importante añadirle un “caption” y un “label” para poder hacer referencia a ellas en el párrafo correspondiente. Nosotros ponemos la referencia a la figura 9.1 que está en la página 44, justo aquí debajo, pero \LaTeX puede que la ubique en otro lugar. (observa el código \LaTeX de este párrafo para observar como se realizan las referencias. Estos detalles también se aplican a tablas y otros objetos).

Existe también la posibilidad de realizarlo sin tablas, con subfiguras:

```
\begin{figure}[h]
  \centering
  \begin{subfigure}[b]{0.4\textwidth} % Espacio horizontal ocupado por la subfigura
    \centering
    \includegraphics[width=4cm]{archivos/subs-sin} % Tamaño de la imagen
    \caption{Sin procesado.}
    \label{fig:gull}
  \end{subfigure}
  ~ % Añadir el espacio deseado, si se deja la línea en blanco la siguiente subfigura irá en una nueva línea
  \begin{subfigure}[b]{0.4\textwidth} % Espacio horizontal ocupado por la subfigura
    \centering
    \includegraphics[width=4cm]{archivos/subs-con} % Tamaño de la imagen
    \caption{Con procesado.}
    \label{fig:tiger}
  \end{subfigure}
  \caption{Ejemplo de subfiguras} \label{systemass}
\end{figure}
```



(a) Sin procesado.



(b) Con procesado.

Figura 9.1: Ejemplo de subfiguras

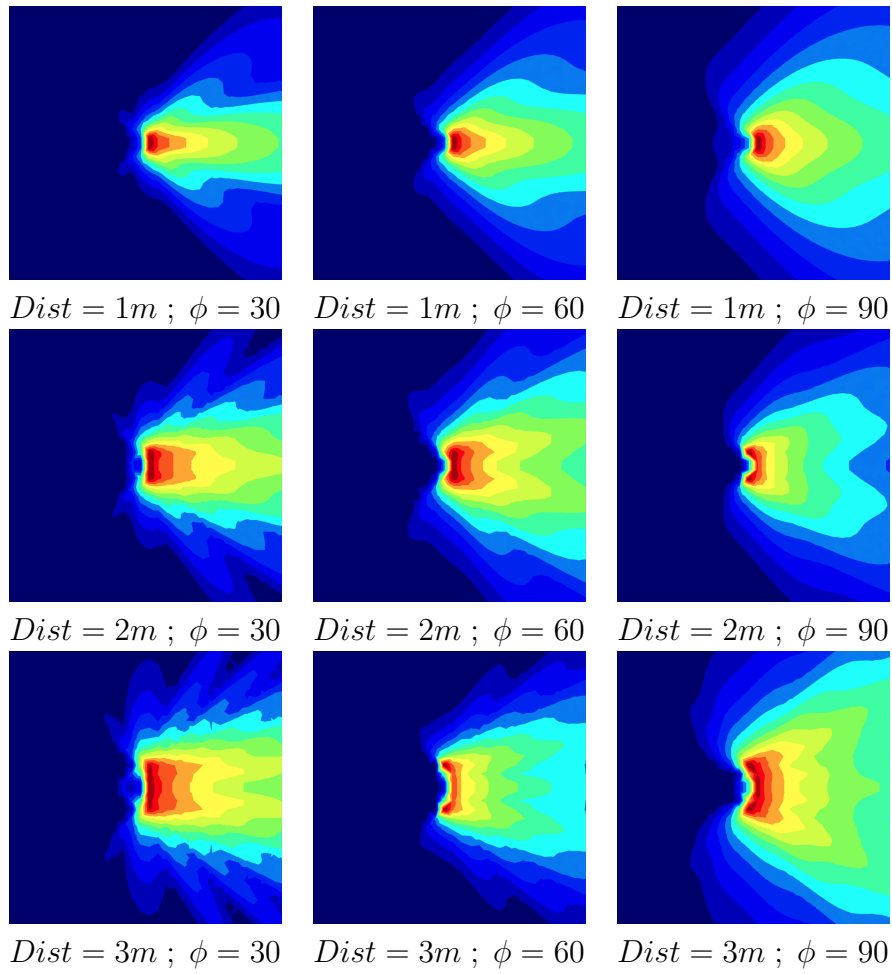
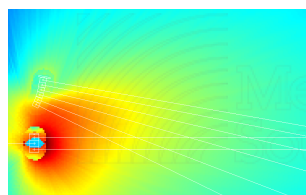
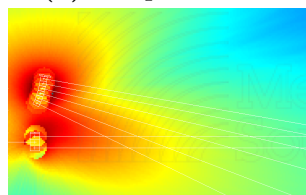


Tabla 9.1: Esta es una tabla con múltiples imágenes. Útil cuando se deben mostrar varias juntas.

Si eliminas la línea '`\caption`' de las subfiguras, tendrás las imágenes sin la información individual, aunque sí con la principal. Y obviamente, si eliminas el de la figura no se mostrará ninguna información.



(a) Sin procesado.



(b) Con procesado.

Figura 9.2: Ejemplo de subfiguras vertical

10. Desarrollo (Con ejemplos de código)

10.1. Inserción de código

A veces tendrás que insertar algún pedazo de código fuente para explicar algo relacionado con él. No sustituyas explicaciones con códigos enormes. Si pones algo de código en tu TFG que sea para demostrar algo o explicar alguna solución.

L^AT_EX te ayuda a escribir código de manera que su presentación tenga las marcas y tabulaciones propias de este tipo de texto. Para ello, debes poner el código que escribas DENTRO de un entorno que se llama “listings”. La plantilla ya tiene una serie de instrucciones para incluir el paquete “listings” y añadirle algunos modificadores por lo que no tienes que incluirlo tú. Simplemente, mete tu código en el entorno “lstlisting” y ya está. Puedes indicar el lenguaje en el que está escrito el código y así L^AT_EX lo mostrará mejor.

En el archivo *estiloscodigoprogramacion.tex* están definidos algunos lenguajes para mostrarlos con un diseño concreto, se pueden modificar para cambiar el coloreado del código, qué términos se ponen en negrita, etc. Si se quiere profundizar más en la función “listings” se puede consultar su manual en <http://osl.ugr.es/CTAN/macros/latex/contrib/listings/listings.pdf>, aunque hay mucha información en foros y blog’s que es más fácil de comprender.

Veamos un ejemplo en la figura 10.1:

```
\begin{lstlisting}[style=C, caption={ejemplo código C},label=C_code]
#include <stdio.h>
int main(int argc, char* argv[]) {
    puts("Hola mundo!");
}
\end{lstlisting}
```

El resultado será:

Código 10.1: ejemplo código C

```
1 #include <stdio.h>
2 // Comentario
3 int main(int argc, char* argv[]) {
4     puts("Hola mundo!");
5 }
```

Si lo quieres en color, está definido el estilo C-color en el archivo *estiloscodigoprogramacion.tex*, con algunos parámetros para mejorar la visualización:

```
\begin{lstlisting}[style=C-color, caption={ejemplo código C en color},label=C_code-color]
#include <stdio.h>
// Comentario
int main(int argc, char* argv[]) {
    puts("Hola mundo!");
}
\end{lstlisting}
```

Código 10.2: ejemplo código C en color

```
1  #include <stdio.h>
2  // Comentario
3  int main(int argc, char* argv[]) {
4  puts("Hola mundo!");
5  }
```

Por supuesto, puedes mejorar esta presentación utilizando más modificadores. En la sección 10.2 se indican algunos detalles.

Otro ejemplo, ahora para mostrar código PHP, sería escribir en tu fichero \LaTeX lo siguiente:

```
\begin{lstlisting}[style=PHP, caption={ejemplo código PHP},label=PHP_code]
/*
Ejemplo de código en PHP para escribir tu primer programa en este lenguaje
Copia este código en tu ordenador y ejecútalo
*/
<html>
<head>
<title>Prueba de PHP</title>
</head>
<body>
<?php echo '<p>Hola Mundo</p>'; ?> //esto lo escribe TODO el mundo
</body>
</html>
\end{lstlisting}
```

y el resultado es el siguiente:

Código 10.3: ejemplo código PHP

```
100 /*
101 Ejemplo de código en PHP para escribir tu primer programa en este lenguaje. Copia este código en tu ↵
    ↵ ordenador y ejecútalo
102 */
103 <html>
104 <head>
105 <title>Prueba de PHP</title>
106 </head>
107 <body>
108 <?php echo '<p>Hola Mundo</p>'; ?> //esto lo escribe TODO el mundo
109 </body>
110 </html>
```

O también en color:

Código 10.4: ejemplo código PHP

```

1 /*
2 Ejemplo de código en PHP para escribir tu primer programa en este lenguaje. Copia este código en tu ↵
   ↵ ordenador y ejecútalo
3 */
4 <html>
5 <head>
6 <title>Prueba de PHP</title>
7 </head>
8 <body>
9 <?php echo '<p>Hola Mundo</p>'; ?> //esto lo escribe TODO el mundo
10 </body>
11 </html>

```

Observa cómo \LaTeX ha puesto los comentarios en gris y ajustado el código para que se muestre más claro.

A continuación se muestran otros ejemplos:

Código 10.5: ejemplo código Matlab en color

```

1 %% Code sections are highlighted.
2 % System command are supported...
3 !touch testFile.txt
4 A = [1, 2, 3;... %... as is line continuation.
5      4, 5, 6];
6 fid = fopen('testFile.text', 'w');
7 for k=1:10
8     fprintf(fid, '%6.2f \n', k)
9 end
10 x=1; %% this is just a comment, not the start of a section
11 % Context-sensitive keywords get highlighted correctly...
12 p = properties(person); %(here, properties is a function)
13 x = linspace(0,1,101);
14 y = x(end:-1:1);
15 % ... even in nonsensical code.
16 ]end()((((end while { end )end ))))end (end
17 %{
18     block comments are supported
19 %} even
20 runaway block comments are

```

Código 10.6: ejemplo código Matlab en blanco y negro

```

1 %% Code sections are highlighted.
2 % System command are supported...
3 !touch testFile.txt
4 A = [1, 2, 3;... %... as is line continuation.
5      4, 5, 6];
6 fid = fopen('testFile.text', 'w');
7 for k=1:10
8     fprintf(fid, '%6.2f \n', k)
9 end
10 x=1; %% this is just a comment, not the start of a section
11 % Context-sensitive keywords get highlighted correctly...
12 p = properties(person); %(here, properties is a function)
13 x = linspace(0,1,101);
14 y = x(end:-1:1);

```

```
15% ... even in nonsensical code.  
16]end()(((end while { end }end )))end (end  
17%{  
18    block comments are supported  
19%} even  
20runaway block comments are
```

Código 10.7: ejemplo código Python en color

```

1 class Example (object):
2     def __init__ (self, account, password):
3         """e.g. account = 'bob@example.com/test'
4             password = 'bigbob'
5         """
6
7         reg = telepathy.client.ManagerRegistry()
8         reg.LoadManagers()
9
10        # get the gabble Connection Manager
11        self.cm = cm = reg.GetManager('gabble')
12
13        # get the parameters required to make a Jabber connection
14        # begin ex.basics.dbus.language-bindings.python.methods.call
15        cm[CONNECTION_MANAGER].RequestConnection('jabber',
16        {
17            'account': account,
18            'password': password,
19        },
20        reply_handler = self.request_connection_cb,
21        error_handler = self.error_cb)
22        # end ex.basics.dbus.language-bindings.python.methods.call

```

Código 10.8: ejemplo código Python en blanco y negro

```

1 class Example (object):
2     def __init__ (self, account, password):
3         """e.g. account = 'bob@example.com/test'
4             password = 'bigbob'
5         """
6
7         reg = telepathy.client.ManagerRegistry()
8         reg.LoadManagers()
9
10        # get the gabble Connection Manager
11        self.cm = cm = reg.GetManager('gabble')
12
13        # get the parameters required to make a Jabber connection
14        # begin ex.basics.dbus.language-bindings.python.methods.call
15        cm[CONNECTION_MANAGER].RequestConnection('jabber',
16        {
17            'account': account,
18            'password': password,
19        },
20        reply_handler = self.request_connection_cb,
21        error_handler = self.error_cb)
22        # end ex.basics.dbus.language-bindings.python.methods.call

```

10.2. Usos y personalización

El texto que acompaña al código puedes incluirlo o no, también puedes decidir si el texto va numerado o no. A continuación se muestra como:

```

% Con esta línea el código no tendrá título
\begin{lstlisting}[style=Python]
micodigo
\end{lstlisting}

```

```
1 micodigo
```

```
% Con esta línea el código tendrá el título abajo
\begin{lstlisting}[style=Python, caption={Ejemplo de título abajo},captionpos=b]
micodigo
\end{lstlisting}
```

```
1 micodigo
```

Código 10.9: Ejemplo de título abajo

```
% Con esta línea el código tendrá título no numerado
\begin{lstlisting}[style=Python, title={Ejemplo de título no numerado}]
micodigo
\end{lstlisting}
```

Ejemplo de título no numerado

```
1 micodigo
```

```
% Con esta línea el código no tendrá las líneas numeradas
\begin{lstlisting}[style=Python,numbers=none, title={Ejemplo de código sin número de líneas}]
micodigo
sin
número
de
líneas
\end{lstlisting}
```

Ejemplo de código sin número de líneas

```
micodigo
sin
número
de
líneas
```

10.3. Importar archivos fuente

Existe la posibilidad de importar un archivo de código en lugar de copiar su contenido y pegarlo en \LaTeX .

Para realizarlo debes escribir:

```
\lstinputlisting[style=C++-color,caption={Archivo C++ importado}]{archivos/ejemplos/holamundo.cpp}
```

Y se importará con el formato establecido entre los '[']':

Código 10.10: Archivo C++ importado

```

1 #include <stdio.h>
2 int main()
3 {
4     // printf() displays the string inside quotation
5     printf("Hello, World!");
6     return 0;
7 }

```

A continuación se muestran otros ejemplos

\lstinputlisting[style=Python-color,caption={Archivo Py importado},label=importado_py]{archivos/↵
↵ ejemplos/holamundo.py}

Código 10.11: Archivo Py importado

```

1 #-----↵
   ↵ -----
2 # helloworld.py
3 #-----↵
   ↵ -----
4
5 import stdio
6
7 # Write 'Hello, World' to standard output.
8 stdio.writeln('Hello, World')
9
10 #-----↵
   ↵ -----
11
12 # python helloworld.py
13 # Hello, World

```

\lstinputlisting[style=Matlab-color,caption={Archivo Matlab importado},label=importado_m]{archivos/↵
↵ ejemplos/holamundo.m}

Código 10.12: Archivo Matlab importado

```

1 function y = hello_world %#codegen
2
3 y = 'Hello World!';
4
5 end
6 % Copyright 2010 The MathWorks, Inc.

```


11. Resultados (Con ejemplos de gráficos)

11.1. Diagramas

Gracias al paquete *Tikz* se pueden incluir multitud de medios gráficos, diagramas, capas sobre imágenes, etc. Existen múltiples formas de realizarlo, para ello es recomendable consultar la guía de iniciación disponible aquí: <http://cremeronline.com/LaTeX/minimaltikz.pdf> y también el manual completo disponible aquí: <http://osl.ugr.es/CTAN/graphics/pgf/base/doc/pgfmanual.pdf>.

A continuación se muestran algunos ejemplos. Revisa el archivo .tex para ver cómo se utilizan.

Imagen a la que se le ha añadido cuadros y texto desde latex:

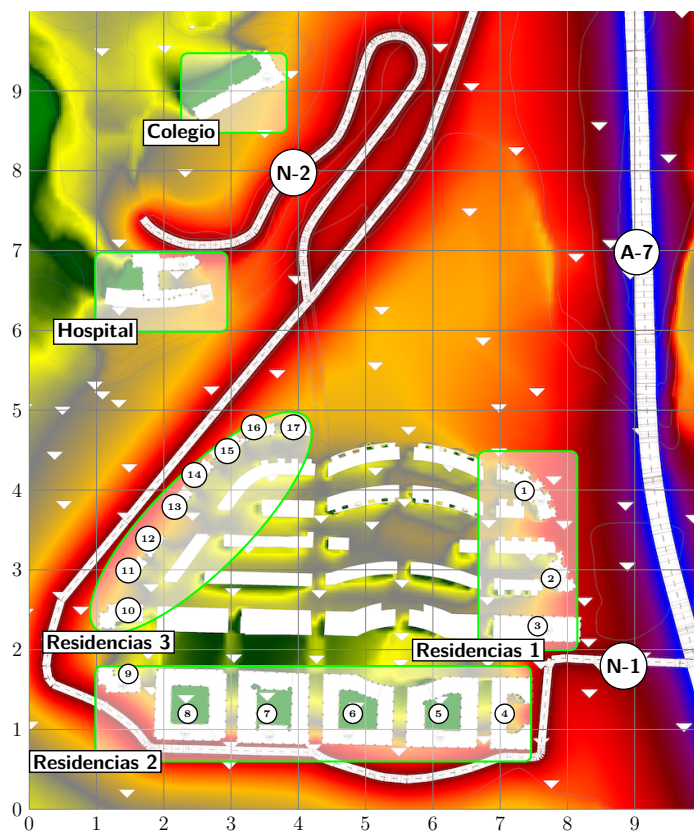
En muchas ocasiones es necesario realizar un diagrama de bloques, más abajo se muestra un ejemplo de ello. En la red hay multitud de ejemplos que pueden ser fácilmente modificables para un fin concreto, como por ejemplo en esta web: <http://www.texample.net/tikz/examples/tag/block-diagrams/>.

11.2. Gráficas

Existen múltiples formas de generar gráficas para latex. Hay disponibles herramientas como GeoGebra que dispone de la utilidad para exportar los gráficos en formato Tkiz. También funciones para Matlab que genera las gráficas que muestra habitualmente pero en código para Tkiz.

La forma más simple, aunque no sencilla cuando abarca muchos datos es la siguiente:

```
\begin{figure}[ht]
\centering
\begin{tikzpicture}
\begin{axis}
[ymin=0,ymax=5, % Límites del eje y
xmin=0,xmax=6, % Límites del eje x
ylabel= eje Y, % Nombre del eje y
xlabel= eje X] % Nombre del eje x
\addplot+[smooth] coordinates % Une los puntos curva suavizada
{(0,0) (1,2) (2,3) (4,3)}; % Puntos de la gráfica
\end{axis}
\end{tikzpicture}
\end{figure}
```



```

\end{tikzpicture}
\caption{Gráfica sencilla.}
\end{figure}

```

El resultado es el siguiente:

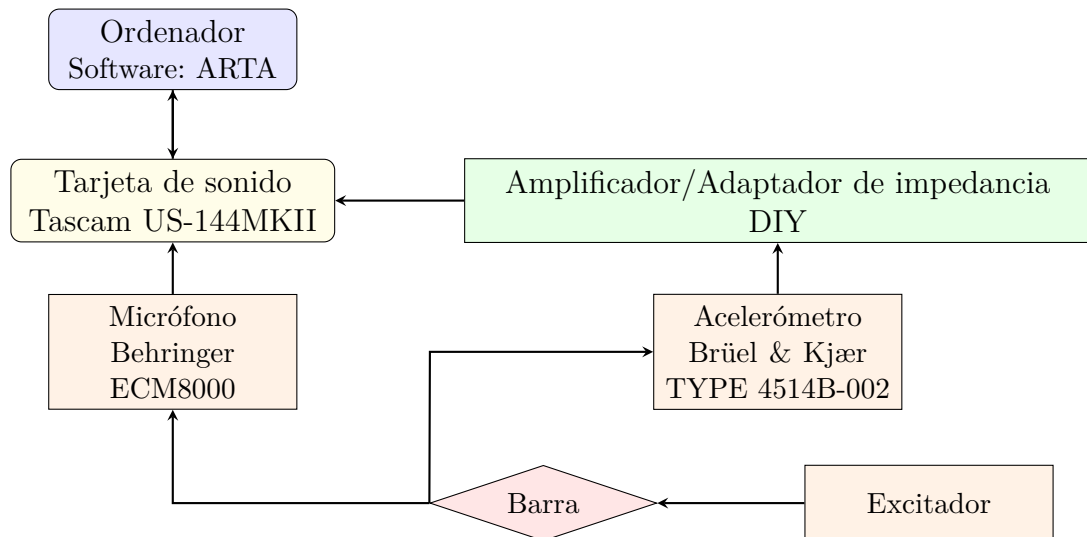


Figura 11.1: Diagrama realizado en latex con Tikz.

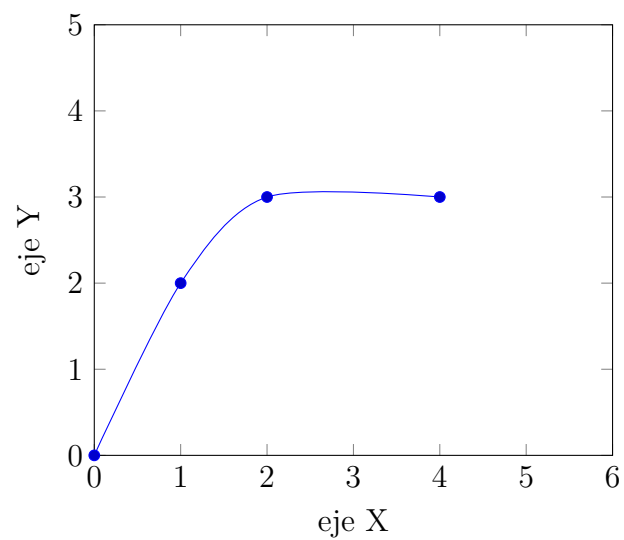


Figura 11.2: Gráfica sencilla.

Otro ejemplo es la gráfica de barras:

```
\begin{figure}[ht]
\centering
\begin{tikzpicture}
\begin{axis}[
ybar=12pt,
ymin=0,ymax=150,
xtick=data,
enlarge x limits={abs=2cm},
symbolic x coords={rubio, moreno},
bar width = 20pt,
ylabel= número,
xlabel= color de pelo,
ytick align=outside,
ytick pos=left,
major x tick style = transparent,
legend style={at={{(0.04,0.96)}},anchor=north west, font=\footnotesize, legend cell align=left,},
]
\addplot[ybar,fill=blue, area legend] coordinates {
(rubio,20)
(moreno,100)};
\addplot[ybar,fill=purple, area legend] coordinates {
(rubio,110)
(moreno,105)};
\legend{Chicos, Chicas}
\end{axis}
\end{tikzpicture}
\caption{Gráfica barras.}
\end{figure}
```

El resultado es el siguiente:

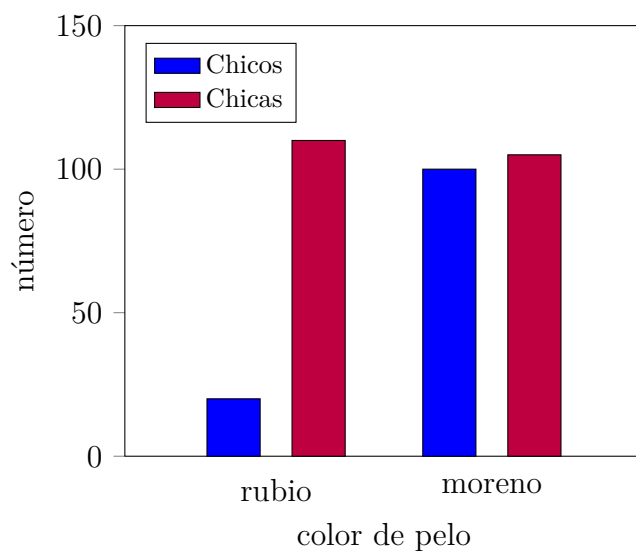


Figura 11.3: Gráfica barras.

11.3. Importados de Matlab

Gracias a la herramienta *matlab2tikz* (<https://es.mathworks.com/matlabcentral/fileexchange/22022-matlab2tikz-matlab2tikz>) se pueden exportar las gráficas de cualquier tipo de Matlab a latex. Después de incluir los archivos de *matlab2tikz* se debe escribir una llamada después de crear la figura tal que:

Código 11.1: Ejemplo de llamada a *matlab2tikz*

```
1 fig = plot(x,y);
2 matlab2tikz('figurehandle',fig,'NombreArchivo.tex','height','5cm','width','13.5cm','strict',true,'↔
↪ showHiddenStrings',true,'showInfo',false)
```

Y para utilizar el archivo generado por la función en este documento:

```
\begin{figure}[ht]
\centering
{\scalefont{0.8}\input{archivos/ejemplos/ParedFina} }
\caption{Ejemplo de gráfica obtenida con matlab2tikz.}
\end{figure}
```

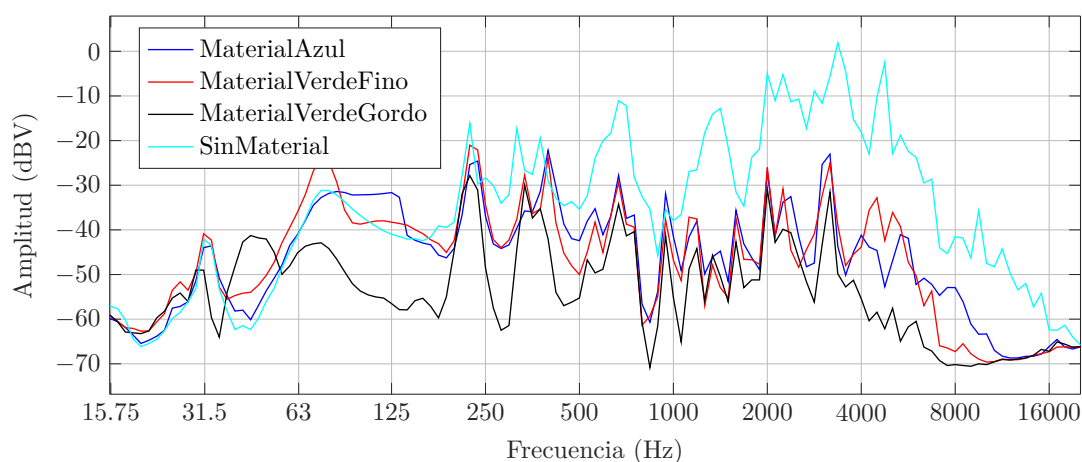


Figura 11.4: Ejemplo de gráfica obtenida con *matlab2tikz*.

Ejemplo de una gráfica 3D generada en Matlab y exportada por *matlab2tikz*:

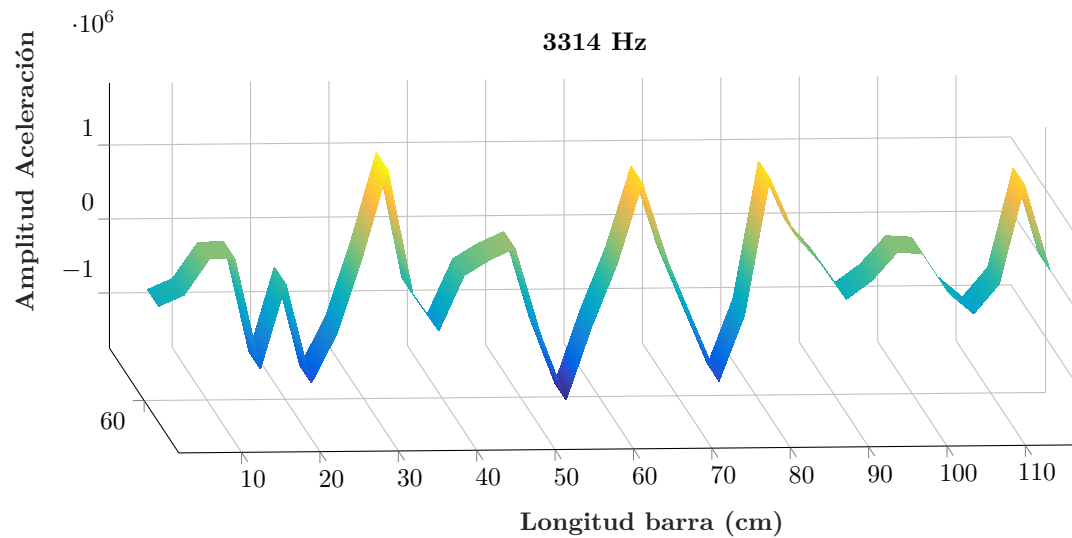


Figura 11.5: Amplitud de la aceleración en el modo número 8.

11.4. Ejemplo avanzado

El potencial del paquete *Tikz* es muy alto, se pueden realizar muchísimas cosas. En la red se facilitan muchos ejemplos para poder ver el funcionamiento y aprender. Existen hilos donde la gente publica sus mejores diseños de *Tikz* como en <https://tex.stackexchange.com/questions/158668/nice-scientific-pictures-show-off> o páginas donde facilitan muchas plantillas como <http://www.texample.net/tikz/examples/all/>.

Un ejemplo de lo que se puede llegar a conseguir es el siguiente:

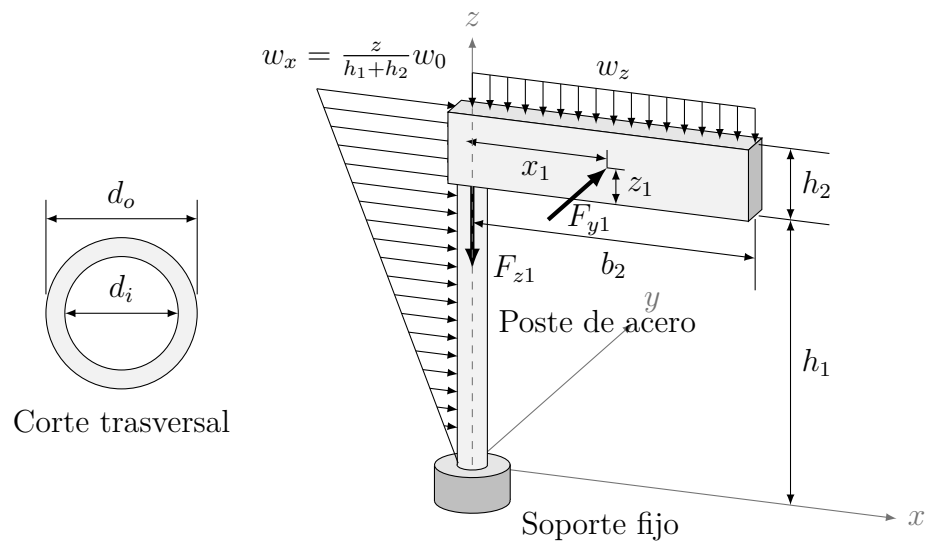


Figura 11.6: Señal realizada con Tikz, sin imágenes.

12. Conclusiones (Con ejemplos de matemáticas)

12.1. Matemáticas

En \LaTeX se pueden mostrar ecuaciones de varias formas, cada una de ellas para un fin concreto.

Antes de ver algunas de estas formas hay que conocer cómo se escriben fórmulas matemáticas en \LaTeX . Una fuente de información completa es la siguiente: <https://en.wikibooks.org/wiki/LaTeX/Mathematics>. También existen herramientas online que permiten realizar ecuaciones mediante interfaz gráfica como <http://www.hostmath.com/>, <https://www.mathcha.io/editor> o <https://www.latex4technics.com/>

Para mostrar una ecuación numerada se debe utilizar:

```
\begin{equation}
\nabla\times{\mathbf H}=\left[\frac{1}{r}\frac{\partial}{\partial r}(rH_{\theta})-\frac{1}{r}\frac{\partial H_r}{\partial \theta}\right]{\hat{\mathbf z}}
\label{ecuacion}
\end{equation}
```

$$\nabla \times \mathbf{H} = \left[\frac{1}{r} \frac{\partial}{\partial r} (r H_{\theta}) - \frac{1}{r} \frac{\partial H_r}{\partial \theta} \right] \hat{\mathbf{z}} \quad (12.1)$$

Si es necesario agrupar varias ecuaciones en un mismo índice se puede escribir del siguiente modo:

```
\begin{subequations}
\begin{eqnarray}
{\mathbf E}&=&E_z(r,\theta){\hat{\mathbf z}} \label{ecu1} \\
{\mathbf H}&=&H_r(r,\theta){\hat{\mathbf r}}+H_{\theta}(r,\theta){\hat{\mathbf \theta}} \label{ecu2}
\end{eqnarray}
\end{subequations}
% Se incluye '&' entre la igualdad para centrar las ecuaciones desde el '='.
```

$$\mathbf{E} = E_z(r, \theta) \hat{\mathbf{z}} \quad (12.2a)$$

$$\mathbf{H} = H_r(r, \theta) \hat{\mathbf{r}} + H_{\theta}(r, \theta) \hat{\boldsymbol{\theta}} \quad (12.2b)$$

Otras dos formas que son las habituales en muchos lugares para incluir ecuaciones son:

Ejemplo de fórmula en línea con el texto `\int_{a}^{b} f(x)dx = F(b) - F(a)`, esta ecuación quedará dentro \leftrightarrow del texto.

Esta otra, al utilizar dos `'$'`, se generará en una línea nueva `$$\int_{a}^{b} f(x)dx = F(b) - F(a)$$`

Ejemplo de fórmula en línea con el texto $\int_a^b f(x)dx = F(b) - F(a)$, esta ecuación quedará dentro del texto.

Esta otra, al utilizar dos `'$'`, se generará en una línea nueva

$$\int_a^b f(x)dx = F(b) - F(a)$$

También se puede añadir información adicional a una ecuación con la función *condiciones* creada para esta plantilla:

```
\begin{equation}
\underset{z=z_0}{\mathrm{Res}}\{f(z)\}=\frac{1}{(m-1)!}\lim_{z\rightarrow z_0}\left[\frac{\text{d}}{\text{d}z}
\right]^{m-1}\left\{\text{d}z^{m-1}f(z)\right\}
\end{equation}

\begin{condiciones}[donde:]
% Excepto 'Descripción y valor' el resto no es necesario el símbolo $para texto matemático.
% Item & Relación & Descripción o valor
m & \rightarrow & Es la multiplicidad del polo $z_0$ \\
z_0 & \rightarrow & Es la parte que se iguala a 0 con el polo. \\
f(z) & \rightarrow & Es la función contenida en la integral.
\end{condiciones}
```

$$\text{Res}_{z=z_0}(f(z)) = \frac{1}{(m-1)!} \lim_{z \rightarrow z_0} \left[\frac{d^{m-1}}{dz^{m-1}} [(z - z_0)^m f(z)] \right] \quad (12.3)$$

donde: m \rightarrow Es la multiplicidad del polo z_0

z_0 \rightarrow Es la parte que se iguala a 0 con el polo.

$f(z)$ \rightarrow Es la función contenida en la integral.

Si lo que deseas es una ecuación alineada a la izquierda o derecha puedes hacerlo con lo siguiente (el `'&'` simple es utilizado para alinear las ecuaciones desde ese punto, los iguales):

```
% Alineado a la izquierda al incluir al final el doble '&&'
```

```

\begin{flalign}
y_{h_1} &= \begin{bmatrix} 6 \cos(\sqrt{6} x) \\ -\sqrt{6} \sin(\sqrt{6} x) \end{bmatrix} e^x \quad & & \\
y_{h_2} &= \begin{bmatrix} 6 \sin(\sqrt{6} x) \\ \sqrt{6} \cos(\sqrt{6} x) \end{bmatrix} e^x \quad & & \\
\end{flalign}

% Alineado a la derecha al incluir al inicio el doble '&&'
\begin{flalign}
&& y_{h_1} &= \begin{bmatrix} 6 \cos(\sqrt{6} x) \\ -\sqrt{6} \sin(\sqrt{6} x) \end{bmatrix} e^x \quad & & \\
&& y_{h_2} &= \begin{bmatrix} 6 \sin(\sqrt{6} x) \\ \sqrt{6} \cos(\sqrt{6} x) \end{bmatrix} e^x \quad & & \\
\end{flalign}

```

$$y_{h_1} = \begin{bmatrix} 6 \cos(\sqrt{6}x) \\ -\sqrt{6} \sin(\sqrt{6}x) \end{bmatrix} e^x \quad (12.4)$$

$$y_{h_2} = \begin{bmatrix} 6 \sin(\sqrt{6}x) \\ \sqrt{6} \cos(\sqrt{6}x) \end{bmatrix} e^x \quad (12.5)$$

$$y_{h_1} = \begin{bmatrix} 6 \cos(\sqrt{6}x) \\ -\sqrt{6} \sin(\sqrt{6}x) \end{bmatrix} e^x \quad (12.6)$$

$$y_{h_2} = \begin{bmatrix} 6 \sin(\sqrt{6}x) \\ \sqrt{6} \cos(\sqrt{6}x) \end{bmatrix} e^x \quad (12.7)$$

Tanto con la función utilizada en (12.1,12.3), como en (12.2a,12.2b) y en las anteriores, si se les incluye un '*' después de 'equation', 'subequation' o 'flalign', se elimina la numeración de las ecuaciones pero manteniendo el resto de características.

Bibliografía

- [1] AENOR. norma une 50136:1997., 1997. URL http://docubib.uc3m.es/CURSOS/Documentos_cientificos/Normas%20y%20directrices/UNE_50136=ISO%207144.pdf.
- [2] Mary Shaw and David Garlan. *Software architecture: perspectives on an emerging discipline*, volume 1. Prentice Hall Englewood Cliffs, 1996.
- [3] Ian F Akyildiz, Dario Pompili, and Tommaso Melodia. Underwater acoustic sensor networks: research challenges. *Ad hoc networks*, 3(5):257–279, 2005.
- [4] A Barkan, Robert L Merlino, and N D’angelo. Laboratory observation of the dust-acoustic wave mode. *Physics of Plasmas*, 2(10):3563–3565, 1995.
- [5] Timothy Leighton. *The acoustic bubble*. Academic press, 2012.
- [6] BOE. Resolución de 7 de marzo de 2012, de la universidad de alicante, por la que se publica el plan de estudios de graduado en ingeniería multimedia. BOE, 22 marzo de 2012, marzo 2012. URL <http://www.boe.es/boe/dias/2012/03/22/pdfs/BOE-A-2012-4008.pdf>.

Bibliografía

- [1] AENOR. norma une 50136:1997., 1997. URL http://docubib.uc3m.es/CURSOS/Documentos_cientificos/Normas%20y%20directrices/UNE_50136=ISO%207144.pdf.
- [2] Ian F Akyildiz, Dario Pompili, and Tommaso Melodia. Underwater acoustic sensor networks: research challenges. *Ad hoc networks*, 3(5):257–279, 2005.
- [3] Mary Shaw and David Garlan. *Software architecture: perspectives on an emerging discipline*, volume 1. Prentice Hall Englewood Cliffs, 1996.
- [4] BOE. Resolución de 7 de marzo de 2012, de la universidad de alicante, por la que se publica el plan de estudios de graduado en ingeniería multimedia. BOE, 22 marzo de 2012, marzo 2012. URL <http://www.boe.es/boe/dias/2012/03/22/pdfs/BOE-A-2012-4008.pdf>.
- [5] A Barkan, Robert L Merlino, and N D’angelo. Laboratory observation of the dust-acoustic wave mode. *Physics of Plasmas*, 2(10):3563–3565, 1995.
- [6] Timothy Leighton. *The acoustic bubble*. Academic press, 2012.

A. Anexo I

Aquí vendría el anexo I

B. Páginas horizontales

Aquí se muestra cómo incluir páginas en horizontal.
Esta página está en vertical

Esta página está en horizontal

Esta página también está en horizontal

Esta página está de nuevo en vertical

C. Importar PDF

A continuación se muestra una página importada de un PDF externo. Observar los comentarios en el código de este anexo para más información. También puedes leer el manual con todas las opciones en <http://osl.ugr.es/CTAN/macros/latex/contrib/pdfpages/pdfpages.pdf>.