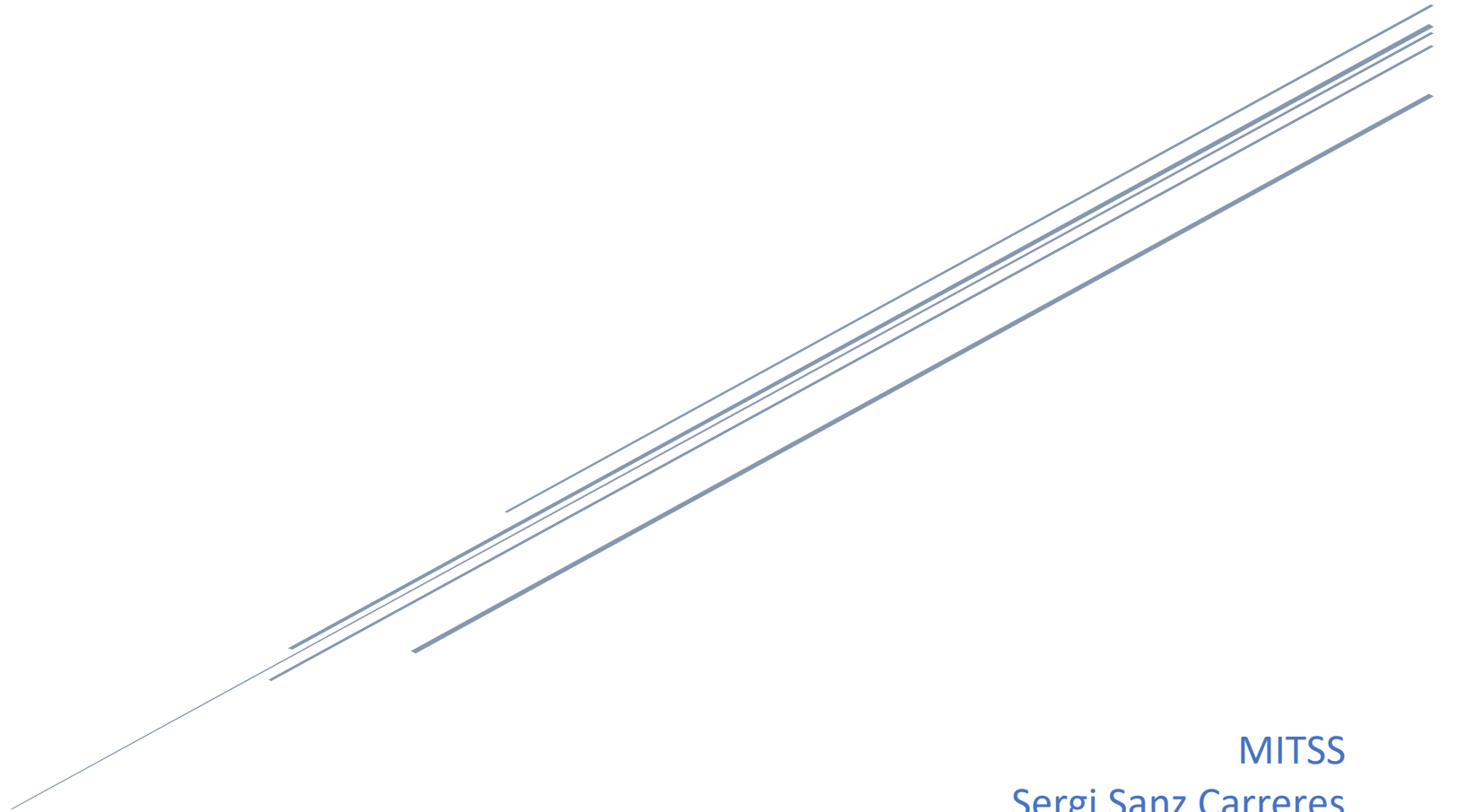


# TRABAJO ACADEMICO 1

SUA



MITSS

Sergi Sanz Carreres

Revista:	Edición	Título	Autor/es	Año
International Journal of Human – Computer Interaction	Volume 31, 2015	<a href="#">Can Autonomous Vehicles Be Safe and Trustworthy? Effects of Appearance and Autonomy of Unmanned Driving Systems</a>	Jae-Gil Lee, KiJoon Kim, Sangwon Lee, and Dong-Hee Shin	05 Oct 2015
	Volume 36, 2020	<a href="#">Autonomous Taxi Service Design and User Experience</a>	Sangwon Kim, Jennifer Jah Eun Chang, Hyun Ho Park, Seon Uk Song, Chang Bae Cha, Ji Won Kim, Namwoo Kang	14 Aug 2019
	Volume 31, 2015	<a href="#">Investigating the Importance of Trust on Adopting an Autonomous Vehicle</a>	Jong Kyu Choi, Yong Gu Ji	05 Oct 2015
HAL-Inria	27 Mar 2018	<a href="#">An Empirical Evaluation of OSGi Dependencies Best Practices in the Eclipse IDE</a>	Lina Ochoa, Thomas Degueule, Jurgen Vinju	2018
	N° 6231, June 2007	<a href="#">Java Components Vulnerabilities - An Experimental Classification Targeted at the OSGi Platform</a>	Pierre Parrend, Stéphane Frénrot	27 Jul 2007
	N° 6801, Jan 2009	<a href="#">I-JVM: a Java Virtual Machine for Component Isolation in OSGi</a>	Nicolas Geoffray, Gaël Thomas, Gilles Muller, Pierre Parrend, Stéphane Frénrot, Bertil Folliot	2009
ACM	ICAC '10: Proceedings of the 7th international conference on Autonomic computing. June 2010 Pages 79–88	<a href="#">Application heartbeats: a generic interface for specifying program performance and goals in autonomous computing environments</a>	Henry Hoffmann, Jonathan Eastep, Marco Domenico Santambrogio, Jason Eric Miller, Anant Agarwal	2010
	SEAMS '08: Proceedings of the 2008 international workshop on Software engineering for adaptive and self-managing systems May 2008	<a href="#">Towards goal-oriented development of self-adaptive systems</a>	Mirko Morandini, Loris Penserini, Anna Perini	2008
	C3S2E '12: Proceedings of the Fifth International C* Conference on Computer Science and Software Engineering June 2012	<a href="#">A survey of formal methods in self-adaptive systems</a>	Danny Weyns, M Usman Iftikhar, Didac Gil De La Iglesia, Tanvir Ahmad	2012

# Artículos seleccionados:

## [Towards goal-oriented development of self-adaptive systems](#)

A lo largo de este artículo se ofrece una visión sobre como funciona el software autoadaptable (o autoadaptativo), el cual tiene como objetivo anticipar los cambios que ocurren en un entorno complejo y tratarlos automáticamente en tiempo de ejecución, para ello toman automáticamente las acciones correctas basadas en el conocimiento de lo que está sucediendo en el sistema, guiados por los objetivos y necesidades de las partes interesadas.

Los sistemas autoadaptativos se caracterizan por tres funcionalidades centrales, supervisar (detectar) el entorno para reconocer "problemas"; tomar decisiones sobre qué comportamiento exhibir; darse cuenta del cambio de comportamiento por adaptación.

Asimismo, el objetivo del artículo es definir un proceso y un marco de diseño basado en herramientas para desarrollar sistemas autoadaptativos, que consideren los modelos de agente Belief-Desire-Intention como arquitecturas de referencia.

El dominio en el que se sitúa este artículo es en el de desarrollo de software autoadaptativo, mientras que, a mi punto de vista, la relación que se produce con la asignatura es el echo de hacer uso de la computación autónoma en especial del bucle de control, donde parte de las decisiones se toman en tiempo de runtime y los sistemas razonan acerca de su estado.

## An Empirical Evaluation of OSGi Dependencies Best Practices in the Eclipse IDE

Este artículo se centra en ofrecer una revisión sistemática de las mejores prácticas relacionadas con la gestión de la dependencia emitidas por la OSGi Alliance y las organizaciones respaldadas por OSGi, contribuyendo a una validación empírica de las mejores prácticas OSGi estándar de la industria. OSGi es un sistema de módulos y un marco de servicio que tiene como objetivo llenar la falta de soporte de Java para el desarrollo modular. Usando OSGi, los desarrolladores dividen el software en múltiples paquetes que declaran dependencias restringidas hacia otros paquetes. Sin embargo, hay varias formas de declarar y administrar tales dependencias, y puede ser confuso para los desarrolladores elegir una sobre otra.

El dominio en el que se sitúa este artículo es en el de desarrollo de software con capacidades de computación autónoma, mientras que, a mi punto de vista, la relación que se produce con la asignatura es el echo de hacer uso de una importante herramienta como es OSGi que se ha abordado a lo largo de la asignatura y que juega un papel importante para todos los proyectos que vamos a desarrollar haciendo uso de esta tecnología.