Proyecto de Iniciación a la Investigación y a la Innovación

Título descriptivo:

Apellidos, Nombre del estudiante: García-Bermejo Mazorra, Víctor
Apellidos, Nombre del Tutor: Gómez Martínez, María Elena

Cuatrimestre: 1 Curso: 2020/21

Fecha: (2020/10/19)

1 Antecedentes, motivación y objetivos

Los sistemas críticos son aquellos sistemas en los que, un fallo puede propiciar consecuencias fatales en un amplio abanico de elementos como son los bienes materiales, la integridad de las personas o la estabilidad de una empresa. Por ello, estos sistemas deben ser confiables, seguros y capaces de mantener sus funciones bajo cualquier circunstancia. Una forma correcta de asegurarse que un sistema crítico tiene las características anteriores, es diseñarlo teniéndolas en cuenta como requisitos no funcionales del sistema.

La evaluación, validación y verificación de la seguridad es vital en los sistemas críticos. Un sistema crítico debe poder cumplir su misión en cualquier problema de seguridad, teniendo la capacidad de recuperarse en caso de fallo. Además, la verificación de las propiedades de seguridad durante el desarrollo de un sistema supone un gran aumento en sus costes.

Actualmente, para la especificación de la seguridad de los sistemas software, se utiliza el denominado lenguaje *Object Constraint Language* (OCL). Mediante este lenguaje se expresan en el Lenguaje de Modelado Unificado (UML) restricciones o contratos que cumplirán el modelo diseñado. Un contrato sirve para añadir precondiciones o post-condiciones a los diversos elementos de un modelo y sus relaciones. Con ello, se podrá efectuar una validación y una verificación temprana de un sistema crítico, pudiendo detectar así, los problemas potenciales durante la fase de diseño. Paralelamente a esta mejora en la fase de diseño, la industria del software está empezando a utilizar métodos formales de la ingeniería del software como el *model checking*. Teóricamente, esta técnica permite verificar un sistema explorando todas sus posibles soluciones.

Para este proyecto de iniciación a la investigación se planea realizar una serie de tareas relacionadas con el desarrollo de técnicas para la validación y verificación durante la fase de diseño de sistemas críticos basados en métodos formales, siendo acompañadas de una metodología para su sistematización. Para ello, se usarán conocimientos aprendidos en la asignatura de Desarrollo de Software Dirigido por Modelos, como por ejemplo la generación de modelos mediante UML y OCL o la representación de los sistemas críticos mediante redes de Petri estocásticas. Todas las tareas realizadas forman parte de un proyecto a la espera de su aprobación.

2 Actividades y cronograma

En el siguiente cronograma se plantearán las actividades que se pretenden realizar a lo largo de este proyecto.

Cronograma

Tareas / subtareas	Primer mes	Segundo mes	Tercer mes	Cuarto mes
T1 Desarrollo de técnicas y métodos para la verificación y validación en la fase de diseño de sistemas críticos.				
T1.1 Análisis en profundidad de las técnicas actuales. Se tratarán de entender tanto los métodos formales como los estándares industriales.	15 h	20 h		
T1.2 Desarrollo de técnicas y métodos para la automatización de contratos de seguridad los cuales serán especificados usando estándares (OCL).		30 h	30 h	
T2. Asesoramiento de la seguridad				
T2.1. Desarrollo de una metodología para la evaluación automática de los aspectos de seguridad de los sistemas críticos.			15 h	40 h
Total	15 h	50 h	30 h	40 h

3 Bibliografía inicial

1. W. Damm, H. Hungar, B. Josko, T. Peikenkamp y I. Stierand, «Using contract-based component specifications for virtual integration testing and architecture design,» de Procs. of Design, Automation & Test in Europe, 2011
2. S. Bauer, A. David, R. Hennicker, K. Guldstrand Larsen, A. Legay, U. Nyman y A. Wasowski, «Moving from specifications to contracts in component-based design,» de Proceedings of the 15th international conference on Fundamental Approaches to Software Engineering (FASE'12), 2012.
3. I. Bate, R. Hawkins y J. McDermid, «A Contract-based approach to designing safe systems,» de Proceedings of the 8th Australian workshop on safety critical systems and software - Volume 33, 2003.
4. T. Bouabana-Tebibel y M. Belmesk, «Integration of the association ends within UML state diagrams,» International Arabic Journal of Information Technology, vol. 5, nº 1, pp. 7-15, 2008.
5. E. Clarke y J. Wing, «Formal methods: State of the art and future directions,» ACM Computing Surveys, vol. 28, nº 4, pp. 626-643, 1996.
6. E. Clarke Jr, O. Grumberg, D. Kroening, D. Peled y H. Veith, Model checking, MIT press, 2018.
7. B. Gallina, E. Gomez-Martinez y C. Benac-Earle, «Promoting MBA in the rail sector by deriving process-related evidence via MDSafeCer,» Computer Standards & Interfaces, vol. 54, pp. 119-128, 2017.
8. E. Gómez-Martínez, R. González-Cabero y J. Merseguer, «Performance Assessment of an Architecture with Adaptative Interfaces for People with Special Needs,» Empirical Software Engineering, vol. 19, pp. 1967-2018, 2014.
9. E. Gómez-Martínez, R. Rodríguez, C. Benac-Earle, L. Etxeberria y M. Illarramendi, «A methodology for model-based verification of safety contracts and performance requirements,» Journal of Risk and Reliability, vol. 232, nº 3, pp. 227-247, 2018.
10. E. Gómez-Martínez y J. Merseguer, «ArgoSPE: Model-based Software Performance Engineering,» de 27th Int. Conf. on Applications and Theory of Petri Nets and Other Models of Concurrency, 2006.

4 Seminarios de investigación

Adicionalmente a las actividades anteriormente detalladas asistiré a un mínimo de 4 Seminarios de Posgrado de la EPS.

Firma del/de la estudiante:

VºBº del tutor/res

GOMEZ MARTINEZ MARIA ELENA - 25168718W

Firmado digitalmente por GOMEZ MARTINEZ MARIA ELENA - 25168718W
Fecha: 2020.10.19 15:50:44 +02'00'