

Programa del curso

Semestre 2022-20

Nombre del curso:	Arquitectura de Software
Créditos:	4
Profesor:	Darío Correal
Versión PDF	Click Aquí

Descripción

El propósito de este curso es realizar una revisión avanzada de los conceptos relevantes asociados a una arquitectura de software. Específicamente, este curso presenta los principales procesos de diseño de una arquitectura de software, así como las características principales de una buena arquitectura de software. El curso hace énfasis en el diseño de arquitecturas de software, particularmente en componentes, conectores, estilos de arquitectura y tácticas relevantes para el logro de requerimientos de calidad en aplicaciones de software actuales. El curso gira en torno al diseño y prototipado de una aplicación de software. Esta aplicación sirve para ilustrar aspectos como la identificación de requerimientos de calidad, el uso de patrones y tácticas de arquitectura y la documentación y presentación de decisiones de diseño críticas para el éxito de un proyecto de software.

Objetivo

El propósito de este curso es presentar conceptos avanzados de arquitectura de software. Este curso presenta elementos relevantes asociados a los procesos de diseño de una arquitectura de software, así como las características principales de una buena arquitectura de software. El curso hace énfasis en la documentación detallada y profesional tanto de los modelos que componen la arquitectura, como del conocimiento y razonamiento que hicieron posibles dichos modelos de arquitectura.

Metodología

El curso gira en torno a 3 grandes retos que buscan solucionar requerimientos de calidad en situaciones específicas de un proyecto de software. Cada reto tiene asociado el cumplimiento de unos requerimientos de calidad, los cuales deben ser diseñados y posteriormente implementados para validar si la solución propuesta cumple con las expectativas de los usuarios. La tecnología seleccionada para la implementación de la validación es decisión de cada grupo de trabajo. Durante la última parte del curso, se

trabaja en el diseño y experimentación de un mini-proyecto de cierre de semestre, en el que se busca integrar todos los conceptos vistos durante el curso. En este proyecto de cierre, cada grupo deberá diseñar una arquitectura completa, satisfaciendo diferentes atributos de calidad, e implementar experimentos que permitan demostrar la validez de su diseño. El curso se trabaja en grupos de 4 estudiantes mínimo, los cuales deben distribuir clara y equitativamente las actividades de diseño e implementación de los retos y del proyecto de cierre del semestre. Se espera que los estudiantes revisen el material de referencia antes de las clases. Durante la clase, se dedicará un tiempo para la solución de dudas. El resto de la clase se enfocará en el diseño de los modelos de arquitectura asociados a cada uno de los retos del curso, mediante trabajos en grupo.

Evaluación

La evaluación está distribuida de la siguiente forma: □ 2 parciales individuales cada uno de (15%) para un total de 30% □ 3 retos cada uno 15% para un total de 45%. Cada reto se divide en modelos de arquitectura (10%) y experimentación (5%) □ Proyecto de cierre 25% **NOTA:** Para aprobar el curso es necesario obtener una nota de 3.0. No hay aproximaciones de notas inferiores a 3.0, es decir que 2.99 corresponde a 2.5. Las notas definitivas del curso varían entre 1.50 a 5.00, en intervalos de 0.5 (por ejemplo 4.0, 4.5, 5.0) Las aproximaciones se realizan de acuerdo a las siguientes reglas: 3:00 a 3.24 = 3.0 3.24 a 3.74 = 3.5 3.75 a 4.24 = 4.0 4.25 a 4.74 = 4.5 4.74 a 5.00 = 5.0 Para pasar el curso es indispensable lograr en el puntaje ponderado 3.00/5.00. No existe aproximación automática en la nota definitiva por debajo de 3.0. En particular, no hay aproximación a 3.00 de puntajes menores a esta nota (e.g., 2.99 no es 3.00, es 2.5). El curso se pierde con una inasistencia del 20%. Nota: El único medio oficial de entrega de las tareas es sicuaplus. No se aceptan trabajos entregados fuera de horario al correo electrónico del profesor o del monitor.

Calendario

Semana	Fecha	Temas	Actividades
1	Enero 21,	Introducción	Conceptos Fundamentales, Proyecto
2	Enero 28,	Inicio reto 1	Desempeño: Estilos, Tácticas, Documentación, Trabajo en Grupo
3	Febrero 4,	Avance reto 1	Estilos y Tácticas, Presentación grupos
4	Febrero 11,	Avance reto 1	Estilos y Tácticas, Presentación grupos
5	Febrero 18,	Cierre reto 1	Evaluación reto 1
6	Febrero 25,	Inicio reto 2	Disponibilidad y Seguridad: Estilos, Tácticas y Documentación, Trabajo en Grupo
7	Marzo 4,	Avance reto 2	Estilos y Tácticas, Presentación grupos
8	Marzo 11,	Parcial 1	Evaluación reto 2
9	Marzo 18,	Inicio reto 3	Flexibilidad e Interoperabilidad: Estilos, Tácticas y Documentación, Trabajo en Grupo
10	Marzo 25,	Avance reto 3	Estilos y tácticas, Presentación grupos
11	Abril 1,	Cierre reto 3	Evaluación reto 3
12	Abril 8,	Proyecto de cierre	Trabajo en grupo
13	Abril 15,	Semana de trabajo individual	
14	Abril 22,	Presentación avance proyecto	cierre: Requerimientos y Estilos, Trabajo en grupos
15	Abril 29,	Parcial 2	Revisión avance del proyecto
16	Mayo 6,	Presentación Proyecto de cierre	

Bibliografía

- [1] Bass, L. Clements, P., Kazman, R., [Software Architecture in Practice], Addison-Wesley, Third Edition. [2] Rozanski Nick, Eoin woods. Software Systems Architecture: Working With Stakeholders Using Viewpoints and Perspectives (2nd Edition). Addison-Wesley Professional; 2 edition (November 4, 2011) [3] Kazman R., Cervantes H. Designing Software Architectures: A practical approach. 2016 [4] Paul Clements, et al. Documenting Software Architectures: Views and Beyond (2nd Edition), Addison-Wesley Professional; 2 edition (October 15, 2010) [5] Anthony Lattanze. Architecting Software Intensive Systems: A Practitioners Guide. Auerbach Publications (November 18, 2008) [6] Richard Taylor, Nenad Medvidovic, Eric Dashofy. [Software Architecture Foundations, Theory and Practice, 2009. [7] George Fairbanks, Just Enough Software Architecture: A risk-driven approach. 2010 [8] Bhuvan Uhhelkar, Software Engineering with UML, CRC Press, 2018