



UNIVERSIDAD NACIONAL MAYOR DE SAN MARCOS
Universidad del Perú, DECANA DE AMÉRICA
FACULTAD DE INGENIERIA DE SISTEMAS E INFORMÁTICA
UNIDAD DE POSGRADO

CONSTANCIA DE ESTUDIOS

N° 0121/ 2017

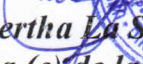
La Directora (e) de la Unidad de Posgrado de la Facultad de Ingeniería de Sistemas e Informática de la Universidad Nacional Mayor de San Marcos, deja constancia que
Don:

ADALBERTO ADOLFO TORRES GONZÁLEZ

Con código 17207016 inició sus estudios en la Maestría en Ingeniería de Sistemas e Informática con mención en Ingeniería de Software en el proceso de admisión 2017-I.

Se expide la presente Constancia a solicitud de la interesada, para los fines que considere conveniente.

Ciudad Universitaria, 08 de mayo de 2017


Dra. Nora Bertha La Serna Palomino
Directora (e) de la UPG-FISI



/kloy

Maestría en Ingeniería de Sistemas e Informática

Mención: Ingeniería de Software

>> PRESENTACIÓN

Esta mención contempla el estudio integral del desarrollo y adaptación de métodos, ambientes y gestión para la construcción de Software. Algunas de las metas de la Ingeniería de Software son mejorar la calidad del producto y del proceso, de los costos y del cronograma. Dentro de este contexto se desarrollan actividades de enseñanza e investigación orientadas a la evaluación, creación de métodos, técnicas y herramientas para mejorar los procesos de construcción de Software. Se da énfasis a las tendencias tecnológicas y de gestión de Software y de proyectos.

>> OBJETIVO GENERAL

Formar profesionales con alto nivel académico, según las tendencias internacionales, capaces de enfrentar con éxito los retos que involucra la gestión, la investigación, el desarrollo y el dominio de las nuevas tecnologías. En específico debe ser capaz de gerenciar y desarrollar proyectos de Software mediante la aplicación de principios y métodos que permitan obtener Software económico, robusto y confiable.

>> PERFIL DEL EGRESADO

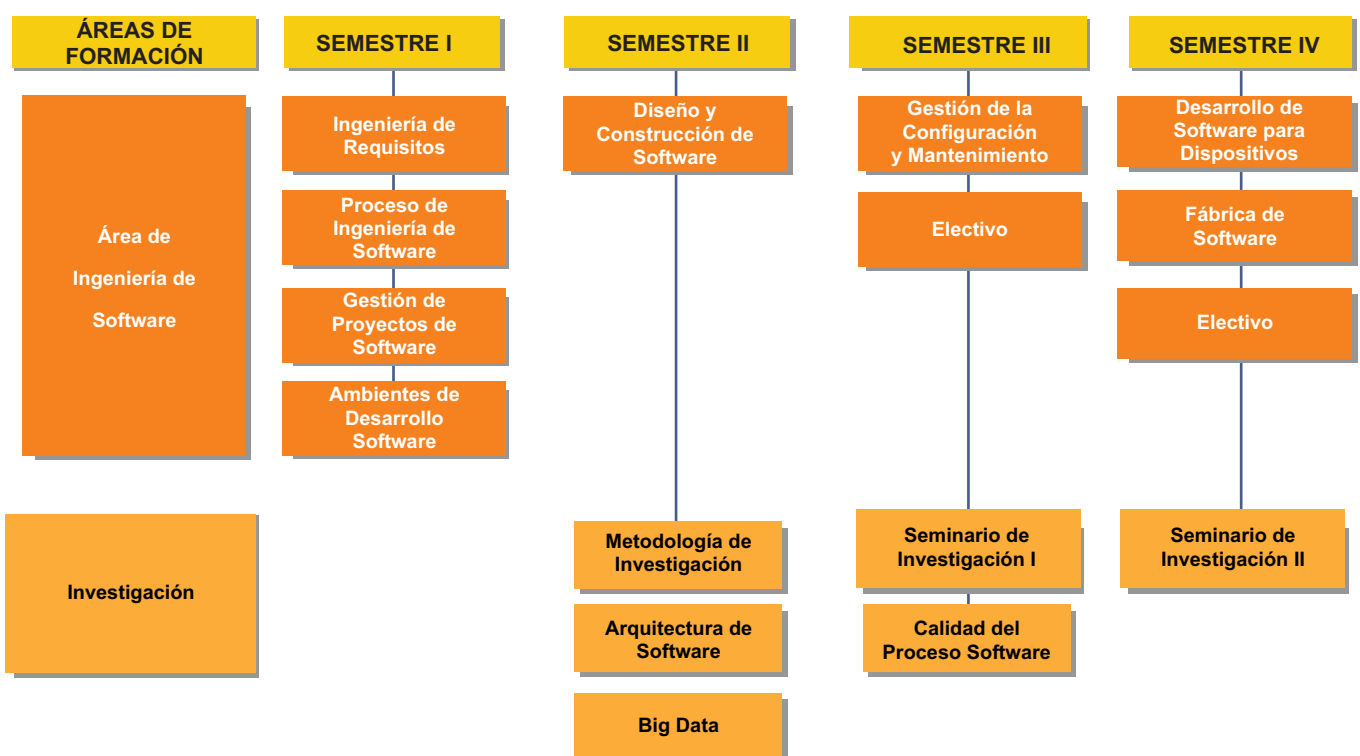
El magíster en Ingeniería de Sistemas e Informática con mención en Ingeniería de Software, estará en capacidad de:

- Liderar equipos de investigación vinculados al desarrollo de proyectos de sistemas de información.
- Formular políticas y estrategias para el desarrollo de la industria del software.
- Gerenciar y desarrollar proyectos de software alineados a los estándares internacionales de calidad.
- Constituir empresas de investigación, desarrollo y consultoría de software.
- Apoyar las actividades científicas multidisciplinarias que requieran de la Ingeniería de Sistemas y la Computación.
- Ejercer la docencia de alto nivel académico en las universidades.

>> GRADO ACADÉMICO

Magíster en Ingeniería de Sistemas e Informática con mención en Ingeniería de Software.

PLAN DE ESTUDIOS (DOS AÑOS ACADÉMICOS)





Universidad Nacional Mayor de San Marcos
 Universidad del Perú, DECANA DE AMÉRICA
SISTEMA ÚNICO DE MATRÍCULA
 FACULTAD DE INGENIERÍA DE SISTEMAS E INFORMÁTICA
 Maestría en Ingeniería de Sistemas e Informática

Período Académico : 2017-1

Reporte de Matrícula

Alumno:

17207016 - TORRES / GONZÁLEZ / ADALBERTO ADOLFO

Plan de Estudios:

Plan de Estudios 2008 - Régimen : Semestral

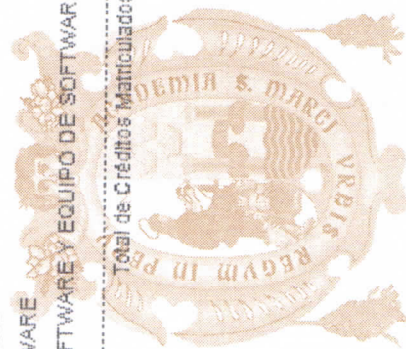
Situación: Regular

Especialidad de Ingeniería del Software

Usuario del Sistema : NLASERINAM
 Fecha de Impresión : 08/05/2017 15:39

Fecha : 8/05/2017 15:39:22

Ciclo	Código	Nombre de la Asignatura	Crédito	Sec.	Aula	Docente
1	J71124	AMBIENTES DE DESARROLLO DE SOFTWARE	4	1		-- / -- / --
1	J71118	INGENIERIA DE REQUISITOS Y ANALISIS	4	1		-- / -- / --
1	J71119	GESTION DE PROYECTOS DE SOFTWARE	4	1		-- / -- / --
1	J71000	PROCESOS DE DESARROLLO DE SOFTWARE Y EQUIPO DE SOFTWARE	4	1		-- / -- / --
Cursos Matriculados : 4			Total de Créditos Matriculados : 16			



Oficina de Matrícula

Firma del Alumno



Universidad Nacional Mayor de San Marcos
 Universidad del Perú, DECANA DE AMÉRICA
 FACULTAD DE INGENIERÍA DE SISTEMAS E INFORMÁTICA
 Maestría en Ingeniería de Sistemas e Informática

Periodo Académico : 2017-2

Reporte de Matrícula

17207016 - TORRES / GONZALEZ / ADALBERTO ADOLFO

Alumno:

Plan de Estudios:

Plan de Estudios 2006 - Régimen : Semestral

Especialidad de Ingeniería del Software

Situación: Regular

Fecha : 25/08/2017 10:52:

J. Juano de Sistema: ZMAMANI
 Fecha de Impresión: 25-08-2017 13:08

Ciclo	Código	Nombre de la Asignatura	Crédito	Seq.	Aula	Docente
2	J71128	SEMINARIO DE INVESTIGACIÓN	4	1		-- / -- / --
2	J71001	INGENIERÍA DE SOFTWARE BASADO EN LA WEB	4	1		-- / -- / --
2	J71129	ARQUITECTURA DE SOFTWARE	4	1		-- / -- / --
DISEÑO DE SOFTWARE Y PATRONES			4			-- / -- / --
Cursos Matriculados : 4			Total de Créditos Matriculados :		24	



Oficina de Matrícula

[Handwritten Signature]

Firma del Alumno



UNIVERSIDAD NACIONAL MAYOR DE SAN MARCOS
FACULTAD DE INGENIERÍA DE SISTEMAS E INFORMÁTICA
UNIDAD DE POSTGRADO

Sílabo

1. DATOS GENERALES

• Curso	:	ARQUITECTURA DE SOFTWARE
• Maestría	:	Maestría en Ingeniería de Sistemas e Informática
• Mención	:	Ingeniería de Software
• Semestre	:	2017 - 2
• Horas Semanales	:	3
• Semestre	:	II
• Profesor(a)	:	Félix Melchor Santos López
• Correo electrónico	:	fsantos2374@gmail.com

2. SUMILLA

Asignatura del área de especialidad con carácter teórico-práctico. Tiene el propósito de brindar los conocimientos para realizar el diseño de la arquitectura de un producto software aplicando las mejores prácticas internacionales. Abarca los siguientes aspectos: conceptos básicos de Arquitectura de Software, diferencia con otros tipos de Arquitectura (Arquitectura Empresarial y Arquitectura de Sistemas), Ciclo de Influencia de la Arquitectura (AIC), Vistas, Puntos de Vistas, Estilos Arquitectónicos, Patrones de Arquitectura, Antipatrones, Arquitecturas de Referencia, Escenarios de Atributos de Calidad, Tácticas, Taller de Atributos de Calidad, Diseño Arquitectural (ADD), Documentación de la Arquitectura (VaB), UML para Arquitectos, Evaluación de Arquitecturas de Software SAAM, ARID y ATAM, Tópicos de Arquitectura de Software y Revisión de artículos Científicos en Arquitectura de Software. Los alumnos realizan una presentación de un caso real aplicando los conocimientos adquiridos.

3. OBJETIVOS

- 1) Identificar los elementos que componen una Arquitectura de Software y su Ciclo de Influencia.
- 2) Conocer y analizar la aplicabilidad de diversos patrones de Arquitectura de Software, Arquitecturas de Referencia, así como el desarrollo de tácticas.
- 3) Desarrollar criterios y entender buenas prácticas para el manejo de stakeholders en la obtención de los atributos de calidad.
- 4) Evaluar arquitecturas de software y seguir los lineamientos proveídos por el *Software Engineering Institute (SEI) – Carnegie Mellon University*.

4. CONTENIDOS Y CRONOGRAMA

SEMANA	CONTENIDOS CONCEPTUALES	ACTIVIDADES/TALLERES /LECTURAS
1	Presentación del docente y alumnos. Introducción al contenido de la asignatura, el silabo, fechas importantes de evaluación, criterio de evaluación. Introducción a conceptos básicos de arquitectura de software.	
2	Implicación de la definición de arquitectura de software y tipos de estructuras.	
3	Vistas arquitecturales y el Ciclo de Influencia de la Arquitectura de software (AIC).	<i>1er Control de Lectura.</i>
4	Patrones de Arquitectura I.	<i>2do Control de Lectura.</i>
5	Patrones de Arquitectura II (microservicios) y Antipatrones.	
6	Atributos de Calidad y Atributos de Calidad del Negocio.	EXAMEN PARCIAL
7	Taller de Atributos de Calidad (QWA) y Tácticas.	<i>PRESENTACIÓN GRUPAL</i>
8	Diseño de Arquitectura de Software Dirigido por Atributos (ADD) y Arquitecturas de Referencia.	<i>3er Control de Lectura.</i> <i>PRESENTACIÓN GRUPAL</i>
9	ADD casos prácticos. Documentación en Arquitectura de Software (VaB).	<i>4to Control de Lectura.</i> <i>PRESENTACIÓN GRUPAL</i>
10	UML y AADL para arquitectos.	<i>PRESENTACIÓN GRUPAL</i>
11	Evaluación de Arquitecturas de Software (ATAM, SAAM, ARID).	<i>PRESENTACIÓN GRUPAL</i>
12	Presentación de los papers de revisión bibliográfica.	EXAMEN FINAL

5. MÉTODO DE DESARROLLO DEL CURSO

El curso se desarrollará de manera presencial mediante clases expositivas y debates grupales. Se utilizarán diapositivas, videos y pizarra. Se llevarán a cabo talleres y desarrollo de casos de estudios.

6. EVALUACIÓN

Criterios

- Asistencia y puntualidad como mínimo al 70 % de las clases.
- Entrega y/o exposición oportuna de trabajos asignados.
- Evaluación de controles de lectura (se elimina la nota más baja).
- Paper de revisión bibliográfica (trabajo individual, exposición e informe).

Promedio de la nota final

CRITERIOS DE EVALUACIÓN	PORCENTAJE
1. Evaluación Parcial	15 %
2. Evaluación Final	20 %
3. Controles de Lecturas	20 %
4. Presentación de Trabajo Grupal	20 %
5. Paper de Revisión Bibliográfica	25 %

7. BIBLIOGRAFÍA

- Bass, L., Clements, P., & Kazman, R. (2012). *Software Architecture in Practice*. Reading, MA: Addison-Wesley.
- Bass, L., Weber, I. & Zhu, L. (2015). *DevOps: A Software Architecture's Perspective*, Addison Wesley.
- Cervantes. H. & Kazman., R. (2016). *Designing Software Architectures: A Practical Approach*, Addison Wesley.
- Clements, P., Bachmann, Felix., Bass, L., Garlan, D., Ivers, J., Little, R., Merson, P., Nord, R., & Stafford, J. (2011). *Documenting Software Architecture: Views and Beyond*, MA: Addison-Wesley.
- Clements. P., Kazman. R. & Klein. M. (2002). *Evaluating Software Architecture: Methods and Case Studies*, Addison Wesley.
- Dugan, D. (2012). *Enterprise Software, Architecture and Design: Entities, Services and Resources*, Wiley.
- Fairbanks, G. (2010). *Just Enough Software Architecture: A Risk-Driven Approach*, Marshall & Brainerd.
- Feiler, P. & Gluch, D. (2013). *Model-Based Engineering with AADL*, Addison Wesley.
- Gonzalez, P. (2016). *Arquitectura y Diseño del Software*, Kindle Edition.
- Gorton, I. (2006). *Essential software architecture*. Berlin: Springer.
- Richards, M. (2013). *Software Architecture Patterns: Understanding Common Architecture Patterns and When to Use Them*. Hoboken, O'reilly.
- Rozanski, N. & Woods, E. (2012). *Software Systems Architecture: Working with Stakeholders Using Viewpoints and Perspectives*, Addison Wesley.
- Taylor, R., Medvidovic, N. & Dashofy, E. (2010). *Software Architecture: Foundations, Theory and Practice*, Wiley.

Artículos para controles de lectura

Control 1:

- Ahmed, Lina Khalid. "The Relationship Between AIC and The Quality Product." In Proceedings of the International Conference on Software Engineering Research and Practice (SERP), p. 152. The Steering Committee of The World Congress in Computer Science, Computer Engineering and Applied Computing (WorldComp), 2016.
- Fernandez, L., 2006. Arquitectura de Software. Software Guru, 2(3), pp.40-45.
- Gorton, I., 2006. Essential software architecture, Chapter 1: Understanding Software Architecture. Springer Science & Business Media.
- Weyrich, Michael, and Christof Ebert. "Reference architectures for the internet of things." IEEE Software 33, no. 1 (2016): 112-116.

Control 2:

- Dragoni, Nicola, Saverio Giallorenzo, Alberto Lluch Lafuente, Manuel Mazzara, Fabrizio Montesi, Ruslan Mustafin, and Larisa Safina. "Microservices: yesterday, today, and tomorrow." arXiv preprint arXiv:1606.04036 (2016).
- O'Brien, Liam, Paulo Merson, and Len Bass. "Quality attributes for service-oriented architectures." In Proceedings of the international Workshop on Systems Development in SOA Environments, p. 3. IEEE Computer Society, 2007.
- Schmidt, Douglas C., Henney, Kevlin and Frank Buschmann. Pattern-Oriented Software Architecture, A Pattern Language for Distributed Computing. Vol. 4. Chapter 2: On Distributed Systems. John Wiley & Sons, 2013.

Control 3:

- Kiran, Mariam, Peter Murphy, Inder Monga, Jon Dugan, and Sartaj Singh Baveja. "Lambda architecture for cost-effective batch and speed big data processing." In Big Data (Big Data), 2015 IEEE International Conference on, pp. 2785-2792. IEEE, 2015.
- Liu, F., Tong, J., Mao, J., Bohn, R., Messina, J., Badger, L. and Leaf, D., 2011. NIST cloud computing reference architecture. NIST special publication, 500(2011), p.292.
- Wu, Weihang, and Tim Kelly. "Safety tactics for software architecture design." In Computer Software and Applications Conference, 2004. COMPSAC 2004. Proceedings of the 28th Annual International, pp. 368-375. IEEE, 2004.

Control 4:

- Babar, M A , & Gorton, I. 2004. Comparison of Scenario-Based Software Architecture Evaluation Methods Engineering Conference, 2004. 11th Asia-Pacific.
- Knodel, J and Naab, M., 2016. Pragmatic evaluation of software architectures (pp. 21-34). Springer.
- Nord, R.L., Barbaacci, M R, Clements, P., Kazman, R. and Klein, M., 2003. Integrating the Architecture Tradeoff Analysis Method (ATAM) with the cost benefit analysis method (CBAM) (No. CMU/SEI-2003-TN-038). CARNEGIE-MELLON UNIV PITTSBURGH PA SOFTWARE ENGINEERING INST