

INGENIERIA DE SOFTWARE I

Año 2023

Carrera/ Plan: (Dejar lo que corresponda)*Licenciatura en Informática Plan 2021/Plan 2015/Plan 2012*
Licenciatura en Sistemas Plan 2021/Plan 2015/Plan 2012
*Analista Programador Universitario Plan 2021/ Plan 2015/Plan 2007**Analista en Tecnologías de la Información y la Comunicación*
Plan 2021/Plan 2017**Año:** 2do**Régimen de Cursada:** Semestral**Carácter (Obligatoria/Optativa):** Obligatoria**Correlativas:** Taller de Programación**Profesor/es :** Patricia Pesado

Marcos Boracchia

Silvia Esponda

Ariel Pasini

Alejandro Gonzalez

Hs. semanales Teoría: 3hs**Hs. semanales Práctica:** 3hs**FUNDAMENTACIÓN**

Ingeniería de Software I, es una asignatura obligatoria correspondiente al segundo año. Los estudiantes aprenderán conocimientos generales acerca de la Ingeniería de Software. En la materia los estudiantes abordarán conocimientos tales como calidad, modelos de procesos, metodologías ágiles. Estos contenidos se vinculan entre sí, y son parte importante en el ciclo de vida de la ingeniería de software. Además, se abordarán temas tales como requerimientos y su forma de modelización. Se sentarán las bases teórico prácticas a desarrollar en la materia Ingeniería de Software 2.

OBJETIVOS GENERALES

Introducir al alumno en los conceptos fundamentales de la Ingeniería de Software.

En particular profundizar las primeras etapas del ciclo de vida (requerimientos, análisis y diseño de sistemas). Presentar los conceptos de calidad de software y de privacidad, integridad y seguridad de datos. El alumno desarrollará trabajos experimentales de análisis y diseño de sistemas.

RESULTADOS DE APRENDIZAJE

- 1.5. Demostrar sensibilización ante la necesidad de contar con amplios conocimientos a la hora de crear aplicaciones informáticas en otras áreas temáticas (Básico).
- 2.1. Utilizar una serie de técnicas con las que identificar las necesidades de problemas reales, analizar su complejidad y evaluar la viabilidad de las posibles soluciones mediante técnicas informáticas (Adecuado).
- 2.2. Describir un determinado problema y su solución a varios niveles de abstracción (Adecuado).
- 2.3. Seleccionar y utilizar los correspondientes métodos analíticos, de simulación y de modelización (Adecuado).
- 2.4. Escoger los patrones de solución, algoritmos y estructuras de datos apropiados (Adecuado).
- 3.1. Definir y diseñar hardware/software informático/de red que cumpla con los requisitos establecidos (Básico).
- 3.2. Describir las fases implicadas en distintos modelos de ciclo de vida con respecto a la definición, construcción, análisis y puesta en marcha de nuevos sistemas y el mantenimiento de sistemas existentes (Básico).

- 3.3. Elegir y utilizar modelos de proceso adecuados, entornos de programación y técnicas de gestión de datos con respecto a proyectos que impliquen aplicaciones tradicionales, así como aplicaciones emergentes (Básico).
- 3.4. Describir y explicar el diseño de sistemas e interfaces para interacción persona-ordenador y ordenador-ordenador (Básico).
- 4.4. Explicar la importancia de la confidencialidad de la información y cuestiones relativas a la seguridad con respecto al diseño, desarrollo, mantenimiento, supervisión y uso de sistemas informáticos (Básico).
- 5.1. Demostrar conocimientos sobre los códigos y estándares de cumplimiento del sector (Básico).
- 5.2. Describir y explicar las técnicas de gestión correspondientes al diseño, implementación, análisis, uso y mantenimiento de sistemas informáticos, incluyendo gestión de proyectos, de configuración y de cambios, así como las técnicas de automatización correspondientes (Adecuado).
- 5.3. Identificar los riesgos, incluyendo riesgos de seguridad, laborales, medioambientales y comerciales y llevar a cabo una evaluación de riesgos, reducción de riesgos y técnicas de gestión de riesgos (Adecuado).
- 6.2. Comunicar mensajes de forma efectiva tanto oralmente como por medio de otros medios de comunicación ante distintas audiencias (Básico).
- 6.4. Identificar las distintas maneras de organizar equipos y los distintos roles dentro de dichos equipos (Básico).
- 6.5. Participar de manera efectiva en grupos de trabajo informático (Básico).

COMPETENCIAS

- CGS2- Comunicarse con efectividad en forma oral y escrita.
- CGT1- Identificar, formular y resolver problemas de Informática.
- CGT2- Concebir, diseñar y desarrollar proyectos de Informática.
- CGT5- Utilizar de manera efectiva las técnicas y herramientas de aplicación de la Informática.
- CGT6 – Capacidad para identificar y gestionar los riesgos en Informática (ambientales, laborales, de seguridad y económicos).
- CGT9 Capacidad de organización de equipos de trabajo en proyectos de Informática, definiendo los roles en los mismos.
- LI- CE4 – Planificar, dirigir, realizar y/o evaluar proyectos de relevamiento de problemas del mundo real, especificación formal de los mismos, diseño, implementación, prueba, verificación, validación, mantenimiento y control de calidad de sistemas de software/sistemas de información que se ejecuten sobre equipos de procesamiento de datos, con capacidad de incorporación de tecnologías emergentes del cambio tecnológico. Capacidad de análisis, diseño y evaluación de interfases humano computador y computador-computador.
- LI- CE6 – Controlar las normas de calidad en el software o software integrado a otros componentes. Capacidad de evaluación de performance de sistemas de software y sistemas que integren hardware y software.
- LS - CE1 – Planificar, dirigir, realizar y/o evaluar proyectos de relevamiento de problemas del mundo real. Especificación formal, diseño, implementación, prueba, verificación, validación, mantenimiento y control de calidad de sistemas de software que se ejecuten sobre sistemas de procesamiento de datos, con capacidad de incorporación de tecnologías emergentes del cambio tecnológico. Capacidad de análisis, diseño y evaluación de interfases humano computador y computador-computador.
- LS - CE5 – Establecer métricas y normas de calidad y seguridad de software, contralando las mismas a fin de tener un producto industrial que respete las normas nacionales e internacionales. Control de la especificación formal del producto, del proceso de diseño, desarrollo, implementación y mantenimiento. Establecimiento de métricas de validación y certificación de calidad. Capacidad de evaluación de performance de sistemas de software y sistemas que integren hardware y software.

CONTENIDOS MINIMOS (de acuerdo al Plan de Estudios)

- El proceso de software. Ciclos de vida de software.
- Herramientas para el proceso de software.

- Ingeniería de requerimientos. Introducción a los métodos formales.
- Metodologías de Análisis y diseño. Lenguajes de modelado.
- Sistemas de Tiempo Real.
- Conceptos de calidad de software.
- Conceptos de Teoría General de Sistemas. Definición de Sistemas de Información
- Conceptos de Privacidad, Integridad y Seguridad en Sistemas de Información

PROGRAMA ANALÍTICO

1- Conceptos de software e ingeniería de software.

- Evolución del software. Características. Componentes.
- Definición de Ingeniería de Software. Evolución.
- Software de alta calidad.
- Ingeniería de Sistemas.

2- Procesos del Software.

- El significado de proceso.
- Modelos de proceso. Modelo de cascada.
- Iteración de procesos. Modelos incrementales. Modelos Evolutivos.
- Prototipación. Metodologías ágiles. Desarrollo basado en componentes.
- Métodos formales. Desarrollo orientado a aspectos. Proceso unificado.
- Actividades del Proceso. Especificación. Diseño. Implementación. Validación. Evolución.
- Herramientas y técnicas para modelado de procesos.
- Ingeniería de Software Asistida por computadora.

3- Ingeniería de Requerimientos.

- El proceso de requerimientos.
- Tipos de requerimientos. Requerimientos funcionales, no funcionales, del usuario, del sistema.
- Características de los requerimientos. Obtención y análisis de los requerimientos.
- Técnicas de comunicación. Los problemas de la comunicación. Elicitación de requisitos. Entrevistas, cuestionarios, JAD, brainstorming.
- Validación de requerimientos. Gestión de requerimientos. Medición de requerimientos.
- Documentos de Especificación de requerimientos. STD 830.

4- Modelos del Sistema.

- Modelo de contexto, de comportamiento, de datos, de objetos.
- Técnicas de especificación de requerimientos: estáticas, dinámicas, relacionales, orientadas a estados, formales.
- Tablas de Decisión, Diagramas de Transición de Estados, Redes de Petri.
- Casos de Uso, Historias de Usuarios, Diagramas de Flujos de Datos, Diagramas de Flujos de Control, UML.
- Prototipado de los requerimientos. Técnicas de construcción rápida.

5- Calidad.

- Conceptos de Calidad y Calidad Total.
- Calidad del proceso y del producto.
- Aseguramiento y estándares de calidad.
- Planeamiento de la calidad.
- Control de la calidad.
- Revisiones Técnicas formales. Inspecciones.
- Modelos de madurez para las organizaciones de desarrollo de software (CMM-CMMI).
- Normas ISO 9000.

6- Sistemas de Información.

- Teoría General de Sistemas.
- Clasificación de los sistemas organizacionales y de información administrativa.
- Privacidad, Integridad y Seguridad.

BIBLIOGRAFÍA

- Ingeniería de Software. 10ma Edición. Ian Sommerville. Pearson. 2016.
- Software Engineering: Theory and Practice. 4th Edition. Shari Pfleeger. Prentice Hall. 2009. (Edición en castellano: Ingeniería de Software. Teoría y Práctica. Shari Pfleeger. Pearson Education. 2002)
- Ingeniería de Software. Un enfoque práctico. 9na Edition. Roger Pressman. McGraw-Hill. 2021.
- Sistemas de Información Administrativa. Murdick R. Prentice Hall. 1988
- Systems Analysis and Design, 9/E. Kendall & Kendall. Pearson. 2013. (Edición en castellano: Análisis y diseño de sistemas. 8va Edición. Kendall & Kendall. Pearson. 2011)

METODOLOGÍA DE ENSEÑANZA MODALIDAD PRESENCIAL

El curso consta de clases teóricas, explicaciones de práctica y clases prácticas.

Las explicaciones de práctica apuntan a brindar las herramientas necesarias para la realización de los trabajos prácticos.

Las actividades prácticas comprenden la realización de ejercicios de modelización de requerimientos aplicando diferentes herramientas explicadas en la teoría.

En la cátedra se pone énfasis en el proceso de identificación de problemas del mundo real, especificación de los mismos como problemas resolubles desde la informática y en el desarrollo de soluciones verificables para los mismos utilizando las técnicas de especificación de requerimientos establecidas en el programa de la materia

Los ejercicios de la práctica son evaluados teniendo en cuenta el contenido técnico, pero también la estructura, organización, sintaxis, claridad conceptual.

Además, los alumnos deben realizar una actividad grupal de elicitation de requerimientos que es evaluada por el docente a cargo de la actividad.

Todas las evaluaciones orales realizadas se reflejan en planillas escritas que conforman documentación de evaluación del trabajo.

EVALUACIÓN

Los alumnos obtienen la cursada aprobando un examen práctico el cual cuenta con dos (2) recuperatorios.

Para la aprobación final de la asignatura los alumnos tienen dos posibilidades:

Alumnos por promoción:

Deben concurrir al 80% de las clases teóricas.

Deben realizar y aprobar una actividad grupal

Deben rendir y aprobar una evaluación teórica con nota 6 o superior (con una recuperación).

Alumnos regulares:

Deben rendir y aprobar un examen final.

CRONOGRAMA DE CLASES Y EVALUACIONES (Tentativo)

Semana	Teoría	Práctica
14/8	Conceptos de I.S.	
	Fases genéricas	
21/8	Técnicas de Comunicación	Técnicas de elicitación
28/8	Técnicas de Comunicación-Tablas de decisión	Técnicas de elicitación
04/9	Requerimientos-DTE	Tablas de decisión
11/9		DTE
18/9	Requerimientos-Redes de Petri	DTE + RP
25/9	Casos de Uso- Historias de Usuarios	
02/10	DFD-DFC	
09/10	Modelos de Procesos	RP + CU
16/10		Casos de Uso- HU
23/10	Metodologías Ágiles	Historias de Usuario
30/10	Calidad y Sistemas de Información	Consulta
06/11	Repaso	Consulta
13/11	Repaso	
20/11		Consulta
27/11	Repaso	

Evaluaciones previstas	Semana
Entrega de Trabajo Práctico de Teoría	02/10
Parcial Práctico	18/10
Re Entrega de Trabajo Práctico de Teoría	30/10
Evaluación Teórica	06/11
Primer Recuperatorio del Parcial Práctico	10/11
Recuperatorio Evaluación Teórica	20/11
Recuperatorio Segunda Evaluación Teórica)	06/12

Contacto de la cátedra (mail, sitio WEB, plataforma virtual de gestión de cursos):

<https://asignaturas.info.unlp.edu.ar/my/sesponda@info.unlp.edu.ar>
marcosb@lidi.info.unlp.edu.ar
agonzalez@lidi.info.unlp.edu.ar



Silvia Esponda
Firma del/los profesor/es

INGENIERIA DE SOFTWARE I
Redictado**Año 2023****Carrera/Plan:**

Licenciatura en Informática Planes 2012-2015 -2021
Licenciatura en Sistemas Planes 2012-2015- 2021
Analista Programador Universitario Planes 2015 - 2021
Analista en Tecnologías de la Información y la
Comunicación Planes 2017 - 2021

Año: 2°**Régimen de cursada:** Semestral**Carácter:** Obligatoria**Correlativas:**

Taller de Programación

Profesores: Patricia Pesado

Marcos Boracchia

Silvia Esponda

Ariel Pasini

Alejandro Gonzalez

Hs. semanales teoría: 2 Hs.**Hs. semanales práctica:** 4 Hs**OBJETIVOS GENERALES**

Introducir al alumno en los conceptos fundamentales de la Ingeniería de Software. En particular profundizar las primeras etapas del ciclo de vida (requerimientos, análisis y diseño de sistemas). Presentar los conceptos de calidad de software y de privacidad, integridad y seguridad de datos. El alumno desarrollará trabajos experimentales de análisis y diseño de sistemas.

RESULTADOS DE APRENDIZAJE

- 1.5. Demostrar sensibilización ante la necesidad de contar con amplios conocimientos a la hora de crear aplicaciones informáticas en otras áreas temáticas (Básico).
- 2.1. Utilizar una serie de técnicas con las que identificar las necesidades de problemas reales, analizar su complejidad y evaluar la viabilidad de las posibles soluciones mediante técnicas informáticas (Adecuado)
- 2.2. Describir un determinado problema y su solución a varios niveles de abstracción (Adecuado).
- 2.3. Seleccionar y utilizar los correspondientes métodos analíticos, de simulación y de modelización (Adecuado).
- 2.4. Escoger los patrones de solución, algoritmos y estructuras de datos apropiados (Adecuado).
- 3.1. Definir y diseñar hardware/software informático/de red que cumpla con los requisitos establecidos (Básico).
- 3.2. Describir las fases implicadas en distintos modelos de ciclo de vida con respecto a la definición, construcción, análisis y puesta en marcha de nuevos sistemas y el mantenimiento de sistemas existentes (Básico).
- 3.3. Elegir y utilizar modelos de proceso adecuados, entornos de programación y técnicas de gestión de datos con respecto a proyectos que impliquen aplicaciones tradicionales así como aplicaciones emergentes (Básico).
- 3.4. Describir y explicar el diseño de sistemas e interfaces para interacción persona-ordenador y ordenador-ordenador (Básico).
- 4.4. Explicar la importancia de la confidencialidad de la información y cuestiones relativas a la seguridad con respecto al diseño, desarrollo, mantenimiento, supervisión y uso de sistemas informáticos (Básico).
- 5.1. Demostrar conocimientos sobre los códigos y estándares de cumplimiento del sector (Básico).
- 5.2. Describir y explicar las técnicas de gestión correspondientes al diseño, implementación, análisis, uso y mantenimiento de sistemas informáticos, incluyendo gestión de proyectos, de configuración y de cambios, así como las técnicas de automatización correspondientes (Adecuado).

- 5.3. Identificar los riesgos, incluyendo riesgos de seguridad, laborales, medioambientales y comerciales y llevar a cabo una evaluación de riesgos, reducción de riesgos y técnicas de gestión de riesgos (Adecuado)
- 6.2. Comunicar mensajes de forma efectiva tanto oralmente como por medio de otros medios de comunicación ante distintas audiencias (Básico).
- 6.4. Identificar las distintas maneras de organizar equipos y los distintos roles dentro de dichos equipos (Básico).
- 6.5. Participar de manera efectiva en grupos de trabajo informático (Básico).

COMPETENCIAS

- CGS2- Comunicarse con efectividad en forma oral y escrita.
- CGT1- Identificar, formular y resolver problemas de Informática.
- CGT2- Concebir, diseñar y desarrollar proyectos de Informática.
- CGT5- Utilizar de manera efectiva las técnicas y herramientas de aplicación de la Informática.
- CGT6 – Capacidad para identificar y gestionar los riesgos en Informática (ambientales, laborales, de seguridad y económicos).
- CGT9 Capacidad de organización de equipos de trabajo en proyectos de Informática, definiendo los roles en los mismos.
- LI- CE4 – Planificar, dirigir, realizar y/o evaluar proyectos de relevamiento de problemas del mundo real, especificación formal de los mismos, diseño, implementación, prueba, verificación, validación, mantenimiento y control de calidad de sistemas de software/sistemas de información que se ejecuten sobre equipos de procesamiento de datos, con capacidad de incorporación de tecnologías emergentes del cambio tecnológico. Capacidad de análisis, diseño y evaluación de interfases humano computador y computador-computador.
- LI- CE6 – Controlar las normas de calidad en el software o software integrado a otros componentes. Capacidad de evaluación de performance de sistemas de software y sistemas que integren hardware y software.
- LS - CE1 – Planificar, dirigir, realizar y/o evaluar proyectos de relevamiento de problemas del mundo real. Especificación formal, diseño, implementación, prueba, verificación, validación, mantenimiento y control de calidad de sistemas de software que se ejecuten sobre sistemas de procesamiento de datos, con capacidad de incorporación de tecnologías emergentes del cambio tecnológico. Capacidad de análisis, diseño y evaluación de interfases humano computador y computador-computador.
- LS - CE5 – Establecer métricas y normas de calidad y seguridad de software, contralando las mismas a fin de tener un producto industrial que respete las normas nacionales e internacionales. Control de la especificación formal del producto, del proceso de diseño, desarrollo, implementación y mantenimiento. Establecimiento de métricas de validación y certificación de calidad. Capacidad de evaluación de performance de sistemas de software y sistemas que integren hardware y software.

CONTENIDOS MINIMOS (de acuerdo al Plan de Estudios)

- El proceso de software. Ciclos de vida de software.
- Herramientas para el proceso de software.
- Ingeniería de requerimientos. Introducción a los métodos formales.
- Metodologías de Análisis y diseño. Lenguajes de modelado.
- Sistemas de Tiempo Real.
- Conceptos de calidad de software.
- Conceptos de Teoría General de Sistemas. Definición de Sistemas de Información
- Conceptos de Privacidad, Integridad y Seguridad en Sistemas de Información

PROGRAMA ANALÍTICO

1- Conceptos de software e ingeniería de software.

- Evolución del software. Características. Componentes.
- Definición de Ingeniería de Software. Evolución.
- Software de alta calidad.
- Ingeniería de Sistemas.

2- Procesos del Software.

- El significado de proceso.
- Modelos de proceso. Modelo de cascada.
- Iteración de procesos. Modelos incrementales. Modelos Evolutivos.
- Prototipación. Metodologías ágiles. Desarrollo basado en componentes.
- Métodos formales. Desarrollo orientado a aspectos. Proceso unificado.
- Actividades del Proceso. Especificación. Diseño. Implementación. Validación. Evolución.
- Herramientas y técnicas para modelado de procesos.
- Ingeniería de Software Asistida por computadora.

3- Ingeniería de Requerimientos.

- El proceso de requerimientos.
- Tipos de requerimientos. Requerimientos funcionales, no funcionales, del usuario, del sistema.
- Características de los requerimientos. Obtención y análisis de los requerimientos.
- Técnicas de comunicación. Los problemas de la comunicación. Elicitación de requisitos. Entrevistas, cuestionarios, JAD, brainstorming.
- Validación de requerimientos. Gestión de requerimientos. Medición de requerimientos.
- Documentos de Especificación de requerimientos. STD 830.

4- Modelos del Sistema.

- Modelo de contexto, de comportamiento, de datos, de objetos.
- Técnicas de especificación de requerimientos: estáticas, dinámicas, relacionales, orientadas a estados, formales.
- Tablas de Decisión, Diagramas de Transición de Estados, Redes de Petri.
- Casos de Uso, Historias de Usuarios, Diagramas de Flujos de Datos, Diagramas de Flujos de Control, UML.
- Prototipado de los requerimientos. Técnicas de construcción rápida.

5- Calidad.

- Conceptos de Calidad y Calidad Total.
- Calidad del proceso y del producto.
- Aseguramiento y estándares de calidad.
- Planeamiento de la calidad.
- Control de la calidad.
- Revisiones Técnicas formales. Inspecciones.
- Modelos de madurez para las organizaciones de desarrollo de software (CMM-CMMI).
- Normas ISO 9000.

6- Sistemas de Información.

- Teoría General de Sistemas.
- Clasificación de los sistemas organizacionales y de información administrativa.
- Privacidad, Integridad y Seguridad.

BIBLIOGRAFÍA

- Ingeniería de Software. 9na Edición. Ian Sommerville. Pearson. 2011.
- Software Engineering: Theory and Practice. 4th Edition. Shari Pfleeger. Prentice Hall. 2009. (Edición en castellano: Ingeniería de Software. Teoría y Práctica. Shari Pfleeger. Pearson Education. 2002)
- Ingeniería de Software. Un enfoque práctico. 7ma Edición. Roger Pressman. McGraw-Hill. 2010.
- Sistemas de Información Administrativa. Murdick R. Prentice Hall. 1988
- Systems Analysis and Design, 9/E. Kendall & Kendall. Pearson. 2013. (Edición en castellano: Análisis y diseño de sistemas. 8va Edición. Kendall & Kendall. Pearson. 2011)

CONDICIONES DE ACCESO AL RE-DICTADO

Para acceder al re-dictado de la materia, los alumnos haber aprobado UN tema del parcial de Ingeniería de software I del curso dictado en el semestre anterior.

METODOLOGÍA DE ENSEÑANZA

El curso consta de clases teóricas, explicaciones de práctica y clases prácticas.

Las explicaciones de práctica apuntan a brindar las herramientas necesarias para la realización de los trabajos prácticos.

Las actividades prácticas comprenden la realización de ejercicios de modelización de requerimientos aplicando diferentes herramientas explicadas en la teoría.

En la cátedra se pone énfasis en el proceso de identificación de problemas del mundo real, especificación de los mismos como problemas resolubles desde la informática y en el desarrollo de soluciones verificables para los mismos utilizando las técnicas de especificación de requerimientos establecidas en el programa de la materia, incluyendo la evaluación de riesgos y las técnicas de gestión y reducción del riesgo. Asimismo se considera en particular la confidencialidad de la información y cuestiones relacionadas con la seguridad

Los ejercicios de la práctica son evaluados teniendo en cuenta el contenido técnico, pero también la estructura, organización, sintaxis, claridad conceptual.

Además los alumnos deben realizar una actividad grupal de elicitation de requerimientos que es evaluada de manera oral por el ayudante a cargo.

Todas las evaluaciones orales realizadas se reflejan en planillas escritas que conforman documentación de evaluación del trabajo.

Las actividades podrán ser realizadas de forma presencial o mediante videoconferencia, en función de las disposiciones de la Facultad.

La asignatura utiliza la plataforma MOODLE de la Facultad de Informática y herramientas de video conferencias

EVALUACIÓN

Los alumnos obtienen la cursada aprobando dos exámenes prácticos en los cuales se evalúan, en cada uno, dos (2) temas. Los mismos cuentan con un recuperatorio donde se evalúan los temas no aprobados en el parcial. Además, se cuenta con un parcial integrador, al final de la cursada, en caso de no lograr la aprobación de los cuatro (4) temas de los dos (2) parciales.

Para la aprobación final de la asignatura los alumnos tienen dos posibilidades:

Alumnos por promoción:

Deben concurrir al 80% de las clases teóricas (presenciales o virtuales).

Deben rendir y aprobar una evaluación teórica con nota 6 o superior con un recuperatorio

Alumnos regulares:

Deben rendir y aprobar un examen final.

CRONOGRAMA DE CLASES Y EVALUACIONES

Inicio de clases estimado: Semana del 27 de marzo
Planificación tentativa del curso:

	Teoría - martes		Práctica - miércoles
28/3	Introducción - Conceptos de I.S. Ingeniería de requerimientos Técnicas de Comunicación	29/3	Practica TE
4/4	Técnicas de especificación de requerimientos Historias de Usuarios - TD	5/4	Practica TD
11/4	Explicación de HU	12/4	Practica HU
18/4	Casos de Uso	19/4	Practica HU
25/4	Explicación CU	26/4	Practica HU
2/5		3/5	Práctica CU
9/5	DTE – RPs	10/5	Práctica HU-CU
16/5	No hay teoría o Consultas HU-CU	17/5	Examen HU-CU
23/5	Explicación DTE – RPs	24/5	Practica DTE RPs
30/5	MP- MA	31/5	Practica DTE RPs
6/6	Calidad	7/6	Practica DTE RPs
13/6	No hay teoría o Consultas DTE - RPs	14/6	Examen DTE RPs
20/6	Feriado	21/6	Consultas
27/6		28/6	Examen HU-CU
4/7		5/7	Examen DTE RPs
12/7	Evaluación Teórica	13/7	Consultas
		2/8	Recuperatorio Global
4/8	Recuperatorio - Evaluación Teórica		

Evaluaciones previstas	Semana
Examen HU-CU	17/5
Examen DTE RPs	14/6
Examen HU-CU – Rec	28/6
Examen DTE RPs – Rec	5/7
Recuperatorio Global	2/8
Evaluación Teórica	13/7
Evaluación Teórica - Rec	4/8

Contacto de la cátedra (mail, sitio WEB, plataforma virtual de gestión de cursos):

Mail: apasini@lidi.info.unlp.edu.ar

Plataforma: Moodle Facultad de Informática (MFI)

Firma del/los profesor/es



Ariel Pasini