

Maestría en Ingeniería de Software

Principios de ingeniería de software automatizada y ágil

1^{er} Semestre

El primer semestre busca establecer una línea base para todos los participantes, garantizando un conocimiento básico mínimo en prácticas esenciales en Ingeniería de Software, particularmente en prácticas fundamentales de diseño, arquitectura de software, metodologías y herramientas ágiles y aplicaciones Web.

Ciclo 1:

- ① **Curso - Prácticas esenciales de desarrollo de ingeniería de software para el agilismo.**
- ② **Curso - Principios de diseño y arquitectura de software.**

Ciclo 2:

- ① **Curso - Ingeniería de software para la web.**
- ② **Curso - Pruebas automatizadas.**

Maestría en Ingeniería de Software

Principios de ingeniería de software automatizada y ágil

1er Semestre

Ciclo 1:

① Curso - Prácticas esenciales de desarrollo de ingeniería de software para el agilismo.

Dedicación

8 semanas - 2 créditos: 12 horas por semana de estudio

Descripción

En este curso se estudiarán 4 prácticas de ingeniería de software: Historias de usuario, flujos de trabajo, desarrollo basado en pruebas e integración continua. Para cada práctica se estudian los conceptos asociados, se realizan tutoriales prácticos y se aplican en un proyecto, en parejas, que se desarrolla a lo largo del curso.

Tecnologías principales

- Python, SQL Alchemy
- Framework pruebas: Unittest
- Git y Github
- Integración

En este curso aprenderás a

Utilizar prácticas esenciales de ingeniería de software para ser un miembro efectivo y eficiente en un equipo de desarrollo ágil (la documentación de requisitos en historias de usuario, flujos de trabajo sobre un manejador de versiones, el desarrollo orientado a pruebas y la práctica de la integración continua).

Metodología del curso

El elemento integrador de las prácticas es el proyecto. Es un proyecto pequeño, donde su alcance está definido y para el cual, los estudiantes analizarán el enunciado construyendo un modelo conceptual en UML y definen unas historias de usuario.

Cada estudiante debe desarrollar un conjunto de HU cuya implementación se hará utilizando la práctica de desarrollo basado en pruebas. El proceso de desarrollo es incremental. Para coordinar el trabajo, deben usar un sistema de control de versiones, un sitio para documentación del proyecto y definir un flujo de trabajo que les permita ir

Maestría en Ingeniería de Software

Principios de ingeniería de software automatizada y ágil

1^{er} Semestre

integrando de forma tal que el proyecto funcione en todo momento. Esta verificación se hace de manera automatizada utilizando una herramienta de integración continua que construye el proyecto y ejecuta las pruebas definidas, reportando el resultado.

② Curso - Principios de diseño y arquitectura de software.

Dedicación

8 semanas - 2 créditos: 12 horas por semana de estudio

Descripción

En este curso, aprenderemos cómo usar eficientemente los estilos y tácticas de arquitectura para garantizar el desempeño de las soluciones construidas, sin perder la capacidad de cambiar y evolucionar al ritmo de hoy. Veremos aplicaciones prácticas de patrones de diseño detallado y su relación con una arquitectura de software moderna. Durante el mismo, tendrás los roles de arquitecto y de diseñador de software, y deberás presentar y sustentar decisiones de diseño, explicando los beneficios de negocio y la viabilidad tecnológica de las propuestas.

Tecnologías principales

- UML

En este curso aprenderás a

- Documentar de forma clara y sin ambigüedades requisitos de calidad críticos para el éxito de un sistema
- Diseñar y documentar una arquitectura de software que satisfaga requisitos de calidad, especialmente relacionados con el desempeño y la facilidad de evolución del software
- Utilizar estilos y tácticas de arquitectura para satisfacer requisitos de calidad
- Aplicar patrones de diseño detallado como mecanismo de refinamiento de tácticas de arquitectura
- Se estudiarán principalmente los atributos de calidad de desempeño y facilidad de modificación

Maestría en Ingeniería de Software

Principios de ingeniería de software automatizada y ágil

1er Semestre

Metodología del curso

A partir de un proyecto de curso, se presentan gradualmente los conceptos de arquitectura, estilos y tácticas, hasta llegar a patrones de diseño detallado. Durante este recorrido, el estudiante analizará las decisiones de diseño tomadas en términos de los beneficios obtenidos y los riesgos generados.

Ciclo 2:

① Curso - Ingeniería de software para la web.

Dedicación

8 semanas - 2 créditos: 12 horas por semana de estudio

Descripción

En este curso se estudian elementos propios del desarrollo de aplicaciones web y se integran los conceptos vistos en los demás cursos del Master Track. El curso se enfoca en Implementar, a partir de una lista de historias de usuario, una aplicación web.

Tecnologías principales

- Desarrollo web: HTML, CSS, Bootstrap, Typescript, Angular, Node.js, Nest.js
- Pruebas: Protractor, Karma, Jasmine
- Integración continua: Jenkins, Git, Github, npm
- IDE: Visual Studio Code

Lo que aprenderás

- Crear el diseño visual de una aplicación web responsive, SPA, accesible, internacionalizable y fácil de usar.
- Diseñar la arquitectura de la aplicación web e implementarla utilizando un framework de desarrollo.
- Seguir un proceso iterativo ágil, que incluye el desarrollo de pruebas automatizadas (unitarias y end-to-end) y prácticas de integración continua.

Metodología del curso

A lo largo de este curso, se desarrolla un proyecto en equipos de tres integrantes cuyo producto es una aplicación web de tamaño mediano. El proyecto, que está definido por

Maestría en Ingeniería de Software

Principios de ingeniería de software automatizada y ágil

1er Semestre

un conjunto de historias de usuario, se lleva a cabo en iteraciones donde se aplican las prácticas de integración continua, entre ellas las pruebas automatizadas y el uso de Gitflow para sincronizar el trabajo. Semanalmente se hará un plan basado en los hitos de la semana y se distribuirá el trabajo entre los integrantes.

② Curso - Pruebas automatizadas.

Dedicación

8 semanas - 2 créditos: 12 horas por semana de estudio

Descripción

Este curso es un primer paso para diseñar estrategias que buscan balancear esfuerzos manuales y automatizados para realizar pruebas de software. Aprende sobre conceptos y técnicas de pruebas automatizadas, estrategias de pruebas, pruebas exploratorias, pruebas de reconocimiento, y pruebas de "extremo a extremo".

Tecnologías principales

- Cypress, Playwright, RUPuppet, Cucumber, Kraken, ResembleJS, FakeJS, Podam, Mockaroo

En este curso aprenderás a

- Diseñar estrategias de pruebas que usen pruebas manuales, de reconocimiento y de "extremo a extremo" (E2E), en el contexto dado por una aplicación bajo pruebas.
- Construir y ejecutar pruebas de reconocimiento y E2E usando frameworks de automatización existentes.
- Experimentar con diferentes herramientas para automatización de pruebas en aplicaciones web.

Metodología del curso

A lo largo de 8 semanas aprenderás diferentes conceptos y técnicas, y realizarás diferentes actividades orientadas a entender las ventajas y limitaciones de diferentes enfoques de pruebas. Las actividades incluyen análisis de casos, lecturas, talleres, ejercicios, presentación de testimonios de profesionales en la industria. Este curso también incluye un proyecto en grupo orientado a diseñar, construir y ejecutar pruebas manuales y automatizadas sobre una aplicación web.