|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **PROGRAMA UNIDAD CURRICULAR** | | | | | | | |
| **Unidad Académica** | | | DEPARTAMENTO DE ECONOMÍA, PRODUCCIÓN E INNOVACIÓN TECNOLÓGICA | | | | |
| **Carrera/s** | | | LICENCIATURA EN GESTIÓN DE TECNOLOGÍAS DE LA INFORMACIÓN | | | | |
| **Plan de Estudios** | | | Resolución (CS) 220/2019 | | | | |
| 1. **Datos sobre la unidad curricular** | | | | | | | |
| **Nombre** | **Laboratorio de Programación y Lenguajes** | | | | **Código** | 6017 | |
| **Modalidad** | PRESENCIAL | | **Régimen** | | Cuatrimestral | | |
|
| **Equipo responsable** | | | **GERARDO MARTIN GONZALEZ TULIAN** | | | | |
| **Año de presentación del programa** | | | **2023-03** | | | | |
| 1. **Carga horaria** | | | | | | | |
| **Horas de clase semanales** | | 4 | |  | | |  |
| **Horas de clase totales** | | 64 | | Horas totales teóricas | | |  |
| Horas totales prácticas | | |  |
| Otras horas totales (laboratorio, trabajo de campo, etc.) | | |  |

|  |  |
| --- | --- |
| 1. **Unidades correlativas** precedentes en el Plan de Estudios | |
| Denominación | Código |
| Paradigmas de Programación | 6015 |
|  |  |

|  |
| --- |
| 1. **Contenidos mínimos** según Plan de Estudios |
| Aplicación de lenguajes de programación. Estudios del lenguajes de programación según distintas necesidades y aplicaciones. Problemáticas de compatibilidad entre lenguajes. Instalación y configuración de lenguajes en distintos sistemas operativos. Desarrollo de un proyecto de programación en un lenguaje particular. |

|  |
| --- |
| 1. **Fundamentación** |
| La asignatura **Laboratorio de Programación y Lenguajes** se ubica en el segundo cuatrimestre del segundo año según el plan vigente (cuarto cuatrimestre), además tiene como correlativa precedente a la asignatura Paradigmas de Programación, y como espacio curricular subsiguiente a la asignatura Seguridad informática..  La asignatura reviste vital importancia para el perfil de egresado, ya que en los entornos dinámicos y cambiante de las organizaciones los problemas requieren soluciones de desarrollo de software que utilizan distintos lenguajes de programación. Es indispensable que el alumno comprenda y conozca los distintos lenguajes de programación que se pueden aplicar en la resolución de problemas informáticos.  El principal enfoque de enseñanza que se utilizan para llevar adelante la asignatura es la resolución de problemas basados en ejemplos prácticos. Se proporcionan diversos ejemplos que los estudiantes analizan y comprenden cómo funcionan, luego trabajan en solucionar problemas utilizando las técnicas de los distintos paradigmas que le permiten aprende a aplicar los conceptos analizados y comprendidos. |

|  |
| --- |
| 1. **Objetivos** |
| Que los estudiantes logren:   * Comprender la base teórica y los fundamentos del lenguajes de programación. En Particular los lenguajes, NodeJs y Java * Conocer la sintaxis y semántica de los lenguajes de programación y su uso en la práctica. * Utilizar entornos de desarrollo que permitan la utilización de distintos lenguajes en cada sistemas operativo. * Resolver problemas lógicos de mediana y alta complejidad utilizando las herramientas que proveen cada lenguaje de programación * Enfrentar situaciones no intuitivas y adquiera pericia para modelar y desarrollar soluciones donde resulta apropiado relacionar múltiples lenguajes entre sí. |

|  |
| --- |
| 1. **Contenidos (organizados por unidades)** |
| **UNIDAD I: Presentación de los lenguajes NodeJs y java**  Instalación y configuración de los lenguajes NodeJs y Java es los sistemas operativos Windows y Linux. Motor de ejecución de NodeJs. Diferencia entre JRE y JDE. Instalación y configuración de los IDEs Visual Studio Code e Intellij para la utilización con NodeJs y Java en los distintos sistemas operativos.  **UNIDAD II: Gestión de Paquetes y Dependencias**  Utilización de los Gestores de paquetes npm y yarm para NodeJs.  Utilización de los Gestores de paquetes Maven y Gradle para Java.  Instalación de las dependencias con cada gestor de paquete de cada lenguaje.  Comprender el sistema de dependencia entre los paquetes en cada lenguaje de programación, analizar y resolver conflictos de versiones entre dependencias. Comprender la utilización de Dependencias de Desarrollo y Producción.  **UNIDAD III: Compatibilidad en los lenguajes con orientación al desarrollo web.**  Desarrollo Web el NodeJs con la librería express. Desarrollo web con el framework spring boot en Java. Compatibilidad entre la rutas de una API Rest entre los lenguajes. Utilización de interfaces en la capa de servicio que pueden implementarse en cada uno de los lenguajes. Sintaxis avanzada y funcional en el manejo de colecciones en lo distintos lenguajes y compatibilidad.  **UNIDAD IV: Desarrollo de un Proyecto de API-RESTful**  Desarrollo de una API-REST en NodeJs o Java con el framework spring-boot que permita realizar las operaciones básicas de un CRUD sobre varios recursos a determinar. Deberá tener manejo de colecciones en memoria. El código deberá estar publicado en un repositorio publico de internet con las instrucciones detalladas para ser probado y usado. El contrato de la API podrá ser reemplado por el desarrollo en otro lenguaje. |

|  |
| --- |
| 1. **Bibliografía obligatoria y complementaria (organizada por unidades)** |
| **UNIDAD I**  López Quintero, Ismael (2013). Node.JS – JavaScript en el lado del servidor – Manual práctico Avanzado. Alfaomega, Altaria Editorial  Deitel, P., & Deitel, H. (2016). Java: cómo programar (10ª ed.). Pearson Educación.  **UNIDAD II**  Rodríguez, P. (2018). Gestión de paquetes en Node.js: Utilizando npm y yarn. Ediciones ENI.  Rivas, F. (2018). Maven y Gradle: Gestión de dependencias en proyectos Java. Ediciones Anaya Multimedia.  **UNIDAD III**  Morales, G (2020) - Creando API con Node. js, Express y MongoDB. [Independently Published](https://www.google.com.ar/search?hl=es-419&q=inpublisher:%22Independently+Published%22&tbm=bks&sa=X&ved=2ahUKEwjG772Mz67_AhXoppUCHarNAgYQmxMoAHoECBEQAg&sxsrf=APwXEdeAsB9uPgmPlskDkcaOdcZjbRYtkg:1686053594667)  Perez Martínez, R. (2016). Desarrollo de aplicaciones mediante framework de spring. Editorial Ra-Ma.  **UNIDAD IV**  Rodríguez, P. (2018). Desarrollo de API RESTful con Java y Spring Boot. Ediciones ENI.  Casciaro, M., & Mammino, L. (2014). "Node.js Design Patterns". Packt Publishing |

|  |
| --- |
| 1. **Metodología de trabajo** |
| La asignatura está constituida por 4 Unidades, las cuales se dictarán durante un cuatrimestre con resolución de problemas a cargo de los alumnos.  Todas las clases se realizarán prácticas en laboratorio de computadoras. Se utilizan PC, proyector y pizarrón o pizarra virtual o programas para grabar las clases en caso de ser necesario.  Los ejercicios propuestos por el docente se resolverán en forma individual. Se utilizarán las herramientas informáticas adecuadas para la resolución de los problemas y la justificación de las respuestas obtenidas.  Trabajo Práctico  El ejercicio práctico se desarrollara durante 4 semanas donde se realizarán puntos de control y ajustes para ir controlando el correcto avance del mismo.  Adicionalmente se utiliza un entorno virtual de información-aprendizaje Campus Virtual de UNPAZ, donde estarán disponibles las guías, avisos, material complementario, videos, etc., que se harán visibles a medida que avancemos cronológicamente con lo planificado. |

|  |
| --- |
| 1. **Evaluación** (Criterios de evaluación y requisitos para la acreditación) |
| Se realizan dos instancias de evaluación parcial. La modalidad será práctica, donde se plantean distintas escenarios donde los alumnos deben desarrollar las respuestas asociando los distintos temas vistos en clase.  Adicionalmente se solicitarán un trabajo práctico obligatorio.  La modalidad de evaluación y requisitos es articulada según Res. C.S. N°150/18) y Res. C.S. N°154/22  El régimen de aprobación de la UUCC podrá ser por:  Según ARTÍCULO 31. C.S. N°150/18) y ARTÍCULO 4 Res. C.S. N°154/22   1. mediante promoción directa; 2. mediante aprobación de examen integrador; 3. mediante examen final.   Regular: aprobó la cursada pero no la materia, con una nota menor a 7 y mayor e igual a 4,  ARTÍCULO 21. Res. C.S. N°150/18). La UC será regularizada cuando el/la estudiante haya cumplido con un mínimo del 75% (setenta y cinco por ciento) de la asistencia y haya obtenido en las instancias evaluatorias parciales (o sus recuperatorios) una calificación de 4 (cuatro) puntos o superior.  Desaprobada: ARTÍCULO 22 Res. C.S. N°150/18). La UC será desaprobada cuando el/la estudiante haya cumplido con un mínimo del 75% (setenta y cinco por ciento) de la asistencia y haya obtenido en alguna de las instancias evaluatorias parciales (o sus recuperatorios) una calificación menor a 4 (cuatro) puntos.  Ausente ARTÍCULO 23. Res. C.S. N°150/18)- Serán considerados ausentes los/as estudiantes que no hayan cumplido con el mínimo del 75% setenta y cinco por ciento de la asistencia o que no hubieren rendido alguno de los exámenes parciales o sus respectivos recuperatorios.  Promociona  ARTÍCULO 35.- Res. C.S. N°150/18. Estarán aprobados mediante promoción directa, aquellos/as estudiantes que:   1. hayan mantenido su condición de regularidad al final del curso conforme lo previsto en el artículo 21 y, 2. hayan obtenido una calificación de 7 (siete) o más puntos como promedio de todas las instancias evaluativas, sean éstas parciales o sus recuperatorios, debiendo obtener una nota igual o mayor a 6 (seis) puntos en cada una de éstas.   Evaluación integradora  ARTÍCULO 36. Res. C.S. N°150/18. Quedarán habilitados automáticamente para rendir la evaluación integradora aquellos/as estudiantes que:   1. hayan mantenido su condición de regularidad al final del curso (conforme lo previsto en el artículo 21); y, 2. hayan obtenido una calificación entre 4 (cuatro) y 6 (seis) puntos en promedio de las instancias parciales y como mínimo un 4 (cuatro) en cada instancia o en sus respectivos recuperatorios.   Examen final  ARTÍCULO 39 Res. C.S. N°150/18). Podrán aprobar la UC mediante examen final los/as estudiantes que: hayan mantenido su condición de regularidad al final del curso conforme lo previsto en el artículo 21; hayan obtenido una calificación entre 4 (cuatro) y 6 (seis) en los respectivos exámenes parciales y/o sus recuperatorios, pero no hubieren aprobado o asistido a la instancia del examen integrador.  ARTÍCULO 40 Res. C.S. N°150/18). Los/as estudiantes podrán inscribirse en 4 (cuatro) oportunidades para rendir el examen final de la UC que hayan regularizado, y por un período de 2 (dos) años desde que haya concluido el curso. En caso de ausencia o desaprobación en ambas instancias, el/la estudiante deberá recursar la UC o rendirla en modalidad de examen libre.  EXÁMENES LIBRES  ARTÍCULO 43 Res. C.S. N°150/18. Los/as estudiantes podrán inscribirse para rendir una UC como libres bajo las siguientes condiciones:   1. tener aprobadas las correlatividades correspondientes a la UC a la que se inscriben; 2. no haber aprobado mediante la modalidad de evaluación libre el veinticinco por ciento (25%) o más de las UUCC que integran el Plan de Estudios de la Carrera; 3. que no esté establecido por el Plan de Estudios de la Carrera ni en el Programa de la UC aprobado por el Consejo Departamental, la imposibilidad de rendir dicha asignatura en la condición de libre.   ARTÍCULO 44. Res. C.S. N°150/18. La modalidad del examen libre será escrita y oral, siendo la primera instancia de carácter previa y eliminatoria. Se evaluarán todos los contenidos establecidos en el programa correspondiente a la fecha del examen. La calificación mínima establecida para la aprobación de la asignatura en examen libre es de 4 (cuatro) puntos. |

|  |
| --- |
| 1. **Instancias de práctica** (si corresponde) |
|  |

|  |  |
| --- | --- |
| 1. **Cronograma** de actividades teóricas y prácticas | |
| Semana 1 | Presentación de la materia.  Instalación y configuración de los lenguajes NodeJs y Java en los sistemas operativos Windows y Linux.  Compilación de java, Interpretación de NodeJs y concepto de transpilación. |
| Semana 2 | Instalación y configuración de los IDEs Visual Studio Code e Intellij para la utilización con NodeJs y Java en los distintos sistemas operativos Windows y Linux,  Practica con los lenguajes NodeJs y Java dentro de los IDEs. |
| Semana 3 | Funcionamiento del Motor de ejecución de NodeJS.  Diferencia entre JRE y JDE. Configuración de distintas versiones de Java sobre los distintos IDEs.  Ejercitación con los lenguajes NodeJs y Java dentro de los IDEs |
| Semana 4 | Presentación de los Gestores de Paquetes.  Explicación de las dependencias funcionales y transitivas. Conflictos de versiones entre la dependencias.  Dependencias de Desarrollo y de producción. |
| Semana 5 | Instalación de los Gestores de paquetes npm y yarm para NodeJs en Windows y Linux.  Instalación de los Gestores de paquetes Maven y Gradle para Java en Windows y Linux.  Instalación de dependencias varias para los entornos desarrollo y de producción.  Casos prácticos de resolución de conflictos entre dependencias y versiones |
| Semana 6 | Practica con ejercicios tipo parcial de las UNIDADES I Y II. |
| Semana 7 | Primera evaluación parcial de las UNIDADES I y II. |
| Semana 8 | Entrega de notas del primer parcial de las UNIDADES I y II.  Discusión y solución conjunta del primer Parcial.  Práctica de ejercicios para el recuperatorio de las UNIDADES I y II. |
| Semana 9 | Desarrollo web con la librería express en NodeJS.  Configuración de la aplicación Web, puerto de escucha (listener)  Definiciones de Rutas, Controladores y Servicios.  Implementación de operaciones sobre colecciones en forma funcional. |
| Semana 10 | Desarrollo web con el FrameWork spring-boot en Java.  Configuración del contender y servidor web Tomcat y Jetty.  Definiciones anotaciones para Controladores y Servicios.  Implementación de operaciones sobre colecciones en forma funcional. |
| Semana 11 | Diferencias y similitudes en el desarrollo realizados.  Practica para el Segundo parcial UNIDAD III  Presentación del Trabajo Practico y elección del lenguaje a utilizar. |
| Semana 12 | Segunda evaluación Parcial UNIDAD III. |
| Semana 13 | Discusión y solución conjunta del segundo Parcial.  Práctica con ejercicios para el recuperatorio del segundo parcial de la UNIDAD III  Seguimiento y control de avance del Trabajo Práctico. |
| Semana 14 | Recuperatorio del primer parcial (UNIDADES I y II) y/o segundo parcial (UNIDAD III) según corresponda a cada alumno. |
| Semana 15 | Seguimiento y control de avance del Trabajo Práctico. |
| Semana 16 | Entrega y revisión final del Trabajo Práctico (UNIDAD IV)  Conclusiones Finales.  Cierre de Notas finales. |

|  |  |
| --- | --- |
| *A partir de aquí completar únicamente las unidades curriculares con régimen anual* | |
| Semana 17 |  |
| Semana 18 |  |
| Semana 19 |  |
| Semana 20 |  |
| Semana 21 |  |
| Semana 22 |  |
| Semana 23 |  |
| Semana 24 |  |
| Semana 25 |  |
| Semana 26 |  |
| Semana 27 |  |
| Semana 28 |  |
| Semana 29 |  |
| Semana 30 |  |
| Semana 31 |  |
| Semana 32 |  |

Firma del docente/s responsable/s:



González Tulian, Gerardo Martin

DNI: 24911146