Sociedad Peruana de Computación (SPC)
Programa Profesional de
Ciencia de la Computación
Sílabo 2021-I

1. CURSO

CS2B1. Desarrollo Basado en Plataformas (Obligatorio)

2. INFORMACIÓN GENERAL

2.1 Créditos	:	3
2.2 Horas de teoría	:	1 (Semanal)
2.3 Horas de práctica	:	2 (Semanal)
2.4 Duración del periodo	:	16 semanas
2.5 Condición	:	Obligatorio
2.6 Modalidad	:	Presencial
2.7 Prerrequisitos	:	CS112. Ciencia de la Computación I. (2do Sem)

3. PROFESORES

Atención previa coordinación con el profesor

4. INTRODUCCIÓN AL CURSO

El mundo ha cambiado debido al uso de la web y tecnologías relacionadas, el acceso rápido, oportuno y personalizado de la información, a través de la tecnología web, ubícuo y pervasiva; han cambiado la forma de ¿cómo hacemos las cosas?, ¿cómo pensamos? y ¿cómo la industria se desarrolla?.

Las tecnologías web, ubicuo y pervasivo se basan en el desarrollo de servicios web, aplicaciones web y aplicaciones móviles, las cuales son necesarias entender la arquitectura, el diseño, y la implementación de servicios web, aplicaciones web y aplicaciones móviles.

5. OBJETIVOS

- Que el alumno sea capaz de diseño e implementación de servicios, aplicaciones web utilizando herramientas y lenguajes como HTML, CSS, JavaScript (incluyendo AJAX) , back-end scripting y una base de datos, a un nivel intermedio.
- Que el alumno sea capaz de desarrollar aplicaciones móviles, administrar servidores web en sistemas basados en UNIX y aplicar técnicas de seguridad en la web a un nivel intermedio.

6. COMPETENCIAS

c) Diseñar, implementar y evaluar un sistema, proceso, componente o programa computacional para alcanzar las necesidades deseadas. (**Usar**)

d) Trabajar efectivamente en equipos para cumplir con un objetivo común. (**Usar**)

g) Analizar el impacto local y global de la computación sobre los individuos, organizaciones y sociedad. (**Usar**)

i) Utilizar técnicas y herramientas actuales necesarias para la práctica de la computación. (**Usar**)

7. COMPETENCIAS ESPECÍFICAS

c3) Utilizar distintas herramientas y lenguajes de programación en los componentes de software (*Full stack*).

c4) Diseñar e implementar arquitecturas de software escalables debilmente acopladas, flexibles, seguras y de facil mantenimiento en distintas plataformas.

c5) Describir como el desarrollo basado en plataformas difiere del propósito general de programación.

c6) Aplicar las ventajas y desventajas de las restricciones de diversas plataformas.

c7) Aplicar o implementar las restricciones de las plataformas Web en el desarrollo de software.

c8) Aplicar los estándares de la web.

c9) Aplicar los estándares de desarrollo para dispositivos móviles.

c10) Implementar software como un servicio.

d1) Desarrollo colaborativo de software utilizando repositorios de código y gestión de versiones (ej. Git, Bitbucket, SVN).

d2) Desarrollar presentaciones grupales e informes sobre tópicos específicos.

g1) Desarrollar soluciones que resuelvan un problema existente en nuestra sociedad.

i2) Utilizar lenguajes y entornos de programación que permitan la implementación y depuración de las soluciones.

8. TEMAS

Unidad 1: Introducción (5)	
Competencias esperadas: g	
Temas	**Objetivos de Aprendizaje**
• Visión general de plataformas (ejemplo, Web, Mobil, Juegos, Industrial) • Programacíon a través de APIs específicos. • Visión general de lenguajes de plataforma (ejemplo, Objective C, HTML5) • Pogramacíón bajo restricciónes de plataforma.	• Describir cómo el desarrollo basado en plataforma difiere de la programación de proposito general [Familiarizarse] • Listar las características de lenguajes de plataforma [Familiarizarse] • Escribir y ejecutar un programa simple basado en plataforma [Familiarizarse] • Listar las ventajas y desventajas de la programación con restricciones de plataforma [Familiarizarse]
Lecturas : [Gro09], [ADC13], [TC15]	

Unidad 2: Plataformas web (5)	
Competencias esperadas: c,g,i	
Temas	**Objetivos de Aprendizaje**
• Lenguajes de programación web (e.g., HTML5, Javascript, PHP, CSS) • Restricciones de las plataformas web: Client-Server, Stateless-Stateful, Caché, Uniform Interface, Layered System, Code on Demand, ReST. • Restricción de plataformas web. • Software como servicio. • Estándares web.	• Diseñar e implementar una aplicación web sencilla [Familiarizarse] • Describir las limitaciones que la web pone a los desarrolladores [Familiarizarse] • Comparar y contrastar la programación web con la programación de proposito general [Familiarizarse] • Describir las diferencias entre software como un servicio y productos de software tradicionales [Familiarizarse] • Discutir cómo los estándares de web impactan el desarrollo de software [Familiarizarse] • Revise una aplicación web existente con un estándar web actual [Familiarizarse]
Lecturas : [Fie00]	

Unidad 3: Desarrollo de servicios y aplicaciones web (25)	
Competencias esperadas: c,d,g,i	
Temas	**Objetivos de Aprendizaje**
• Describir, identificar y depurar problemas relacionados con el desarrollo de aplicaciones web. • Diseño y desarrollo de aplicaciones web interactivas usando HTML5 y Python. • Utilice MySQL para la gestión de datos y manipular MySQL con Python. • Diseño y desarrollo de aplicaciones web asíncronos utilizando técnicas Ajax. • Uso del lado del cliente dinámico lenguaje de script Javascript y del lado del servidor lenguaje de scripting python con Ajax. • Aplicar las tecnologías XML / JSON para la gestión de datos. • Utilizar los servicios, APIs Web, Ajax y aplicar los patrones de diseño para el desarrollo de aplicaciones web.	• Del lado del servidor lenguaje de scripting python: variables, tipos de datos, operaciones, cadenas, funciones, sentencias de control, matrices, archivos y el acceso a directorios, mantener el estado. [Usar] • Enfoque de programación web usando python incrustado. [Usar] • El acceso y la manipulación de MySQL. [Usar] • El enfoque de desarrollo de aplicaciones web Ajax. [Usar] • DOM y CSS utilizan en JavaScript. [Usar] • Tecnologías de actualización de contenido asíncrono. [Usar] • Objetos XMLHttpRequest utilizar para comunicarse entre clientes y servidores. [Usar] • XML y JSON. [Usar] • XSLT y XPath como mecanismos para transformar documentos XML. [Usar] • Servicios web y APIs (especialmente Google Maps). [Usar] • Marcos Ajax para el desarrollo de aplicaciones web contemporánea. [Usar] • Los patrones de diseño utilizados en aplicaciones web. [Usar]
Lecturas : [FR11]	

Unidad 4: Plataformas móviles (5)	
Competencias esperadas: c,d,g,i	
Temas	**Objetivos de Aprendizaje**
• Lenguajes de Programación para Móviles. • Principios de diseño: Segregación de Interfaces, Responsabilidad Única, Separación de Responsabilidades, Inversión de Dependencias. • Desafíos con mobilidad y comunicación inalámbrica. • Aplicaciones Location-aware. • Rendimiento / Compensación de Potencia. • Restricciones de las Plataformas Móviles. • Tecnologías Emergentes.	• Diseñar e implementar una aplicación móvil para una plataforma móvil dada [Familiarizarse] • Discutir las limitaciones que las plataformas móviles ponen a los desarrolladores [Familiarizarse] • Discutir los principios de diseño que guian la construcción de aplicaciones móviles [Familiarizarse] • Discutir el rendimiento vs perdida de potencia [Familiarizarse] • Compare y contraste la programación móvil con la programación de proposito general [Familiarizarse]
Lecturas : [Mar17]	

Unidad 5: Aplicaciones Móviles para dispositivos Android (25)	
Competencias esperadas: c,d,g,i	
Temas	**Objetivos de Aprendizaje**
• The Android Platform • The Android Development Environment • Application Fundamentals • The Activity Class • The Intent Class • Permissions • The Fragment Class • User Interface Classes • User Notifications • The BroadcastReceiver Class • Threads, AsyncTask & Handlers • Alarms • Networking (http class) • Multi-touch & Gestures • Sensors • Location & Maps	• Los estudiantes identifican software necesario y lo instalan en sus ordenadores personales. Los estudiantes realizan varias tareas para familiarizarse con la plataforma Android y Ambiente para el Desarrollo. [Usar] • Los estudiantes construyen aplicaciones que trazan los métodos de devolución de llamada de ciclo de vida emitidas por la plataforma Android y que demuestran el comportamiento de Android cuando los cambios de configuración de dispositivos (por ejemplo, cuando el dispositivo se mueve de vertical a horizontal y viceversa). [Usar] • Los estudiantes construyen aplicaciones que requieren iniciar múltiples actividades a través de ambos métodos estándar y personalizados. [Usar] • Los estudiantes construyen aplicaciones que requieren permisos estándar y personalizados. [Usar] • Los estudiantes construyen una aplicación que utiliza una única base de código, sino que crea diferentes interfaces de usuario dependiendo del tamaño de la pantalla de un dispositivo. [Usar] • Los estudiantes construyen un gestor de listas de tareas pendientes utilizando los elementos de la interfaz de usuario discutidos en clase. La aplicación permite a los usuarios crear nuevos elementos y para mostrarlos en un ListView. [Usar] • Los estudiantes construyen una aplicación que utiliza la información de ubicación para recoger latitud, longitud de los lugares que visitan. [Usar]
Lecturas : [ADC13], [TC15]	

9. PLAN DE TRABAJO

9.1 Metodología

Se fomenta la participación individual y en equipo para exponer sus ideas, motivándolos con puntos adicionales en las diferentes etapas de la evaluación del curso.

9.2 Sesiones Teóricas

Las sesiones de teoría se llevan a cabo en clases magistrales donde se realizarán actividades que propicien un aprendizaje activo, con dinámicas que permitan a los estudiantes interiorizar los conceptos.

9.3 Sesiones Prácticas

Las sesiones prácticas se llevan en clase donde se desarrollan una serie de ejercicios y/o conceptos prácticos mediante planteamiento de problemas, la resolución de problemas, ejercicios puntuales y/o en contextos aplicativos.

10. SISTEMA DE EVALUACIÓN

********* EVALUATION MISSING ********

11. BIBLIOGRAFÍA BÁSICA

[ADC13] J. Annuzzi, L. Darcey, and S. Conder. *Introduction to Android Application Development: Android Essentials*. Developer's Library. Pearson Education, 2013. ISBN: 9780133477337.

[Fie00] Roy Thomas Fielding. "Fielding dissertation: Chapter 5: Representational state transfer (rest)". In: *http://www. ics. uci. edu/˜ fielding/pubs/dissertation/rest_arch_style. htm* (2000).

[FR11] Eric Freeman and Elisabeth Robson. *Head first HTML5 programming: building web apps with JavaScript.* " O'Reilly Media, Inc.", 2011.

[Gro09] R. Grove. *Web Based Application Development.* Jones & Bartlett Learning, 2009. ISBN: 9780763759407.

[Mar17] Robert C Martin. *Clean architecture: a craftsman's guide to software structure and design.* Prentice Hall Press, 2017.

[TC15] Trish and Richard Cornez. *Android Programming Concepts.* Jones and Bartlett Publishers, 2015. ISBN: 9781284070705.