

Escuela de Ingeniería

Obligatorio de: Ingeniería de Software en la Práctica

Código de materia: 6454

Fecha: 11/03/2019

Grupo: Todos

Hoja 1 de 8

OBJETIVOS

Con la finalidad de promover la innovación y el emprendimiento, e intentando darle un alcance que exceda en el tiempo al dictado de la asignatura, es que se propone como trabajo obligatorio la construcción de una aplicación no pre determinada y a elección de los alumnos.

Se busca que este trabajo sirva como base para futuros productos o como prueba de concepto para proyectos finales, lo que motiva el carácter abierto de la propuesta.

TRABAJO

El trabajo consta de la construcción de una aplicación móvil, que debe poder ser ejecutada en un dispositivo corriendo en Android. La versión a soportar queda a elección de los desarrolladores, lo que sí es obligatorio es que la aplicación pueda ser ejecutada en el entorno de desarrollo que cada grupo elija.

Es importante aclarar que la aplicación debe tener un componente que haga las veces de *back end*. Para este o estos componentes se debe utilizar la plataforma de desarrollo .NET.

Se espera que se desarrolle la aplicación completa.

ALCANCE

El obligatorio está dividido en dos entregas, y se espera que el producto se encuentre terminado para la segunda entrega. Dado que cada trabajo será diferente de los demás, es necesario que cada grupo acuerde el alcance de cada una de las entregas con los docentes, quienes deben aprobar la propuesta.

Para ello cada grupo debe proponer por escrito de manera la aplicación que desea construir, y en base a esta propuesta se acordará que se espera para cada una de las entregas. A partir de este momento se pueden presentar propuestas (cuanto antes mejor), las que podrán ser modificadas en función de los comentarios de los docentes.

SOBRE LA PROPUESTA DE TRABAJO

Estudio de la tecnología

La materia requiere un alto grado de **autoestudio** de la tecnología seleccionada, parte de la propuesta de trabajo implica tener la capacidad de familiarizarse rápidamente con una nueva tecnología y tener la habilidad de resolver problemas por medio de la ayuda, de la documentación provista en el web, artículos, libros, etc.

Tecnologías y herramientas de desarrollo

- *Front end*: Desarrollo *mobile* con Android Studio
- *Back end*: ASP.NET Web API con Visual Studio / Base de datos SQL Server

Escuela de Ingeniería		
Obligatorio de: Ingeniería de Software en la Práctica	Código de materia: 6454	
Fecha: 11/03/2019	Grupo: Todos	Hoja 2 de 8

ENTREGAS

La propuesta de trabajo consta de dos entregas, el principal motivo de estas dos entregas es asegurar la adecuada distribución de los tiempos para cada etapa del obligatorio a lo largo del curso así como la validación por parte de los docentes del alcance final.

El obligatorio tiene un **alto grado de complejidad** a nivel de **proceso, diseño e implementación** y también un **alto grado de exigencia** en lo referente a la **documentación, justificaciones e investigación**.

El contenido de las entregas dependerá del enfoque metodológico que el equipo seleccione pudiendo este estar basado en planes o en un enfoque ágil. A continuación se brindan los elementos típicos que se pueden esperar para cada enfoque y se apela al conocimiento adquirido por los estudiantes para complementar los entregables de cada fase.

Enfoque basado en los planes

Área	Entregable	Detalle
Análisis del problema	Documento de especificación de requerimientos	Selección de una metodología de gestión de los requerimientos y su correspondiente justificación.
Gestión de proyecto	Cronograma	Descripción del ciclo de vida a utilizar, cronograma reflejando los dos hitos y entregables para cada etapa del proyecto. Debe establecerse el alcance del proyecto para las dos entregas oficiales del producto.
	Avance	Si corresponde: descripción de los cambios realizados en lo entregado respecto a la primera entrega. Se debe actualizar todos los diagramas que corresponda, y agregar los que se considere necesario para mostrar los cambios realizados y justificar las decisiones tomadas .
	Gestión de riesgos	Identificación y mitigación de los principales riesgos identificados y su evolución en el tiempo.
Gestión de la configuración	Control de cambios y de versiones	Estructura del repositorio, liberaciones identificadas de acuerdo al cronograma. Utilización y evidencia de uso de una herramienta de registro de defectos.
Aseguramiento de calidad	Proceso y producto	Identificación y uso de estándares.
	Pruebas de software	<p>Resultado de la ejecución de las pruebas unitarias, integración y aceptación. Evidencia del código de pruebas unitarias, reporte de la herramienta de cobertura y análisis de los resultados.</p> <p>El documento debe cumplir con los siguientes elementos del Documento 302 de la facultad (http://www.ort.edu.uy/fi/pdf/documento302facultaddeingenieria.pdf), en especial:</p> <ul style="list-style-type: none"> – Capítulo 1, secciones 1.1 (sin la leyenda), 1.5, 1.7, 1.8 y 1.9 – Capítulos 4, 5 y 9

Escuela de Ingeniería		
Obligatorio de: Ingeniería de Software en la Práctica		Código de materia: 6454
Fecha: 11/03/2019	Grupo: Todos	Hoja 3 de 8

Arquitectura	Documentación de diseño	<p>Descripción de la arquitectura y diseño propuesto con su justificación correspondiente.</p> <p>Arquitectura y diseño de toda la aplicación y justificación de las decisiones de diseño tomadas. El suficiente para poder entender adecuadamente el problema, hacer diagramas que aporten para comprender y documentar el diseño. Utilización de principios como los SOLID, patrones de diseño, criterios de diseño de las APIs de front end y back end, etc.</p> <p>Documentación de diseño de UI y UX.</p> <p>Modelo de tablas de la estructura de la base de datos.</p>
Implementación	Código	<p>Aplicación y evidencia de Clean Code.</p> <p>Repositorio compartido con los docentes.</p>

Enfoque basado en metodologías ágiles

Área	Entregable	Detalle
Análisis del problema	Especificación de historias de usuario	<p><i>Product backlog</i> incluyendo historias de usuario estimadas utilizando una escala apropiada.</p> <p>Bocetos de interfaz de usuario asociados a las historias de usuario.</p>
Gestión de proyecto	<i>Release Plan</i>	<i>Release plan</i> correspondiente a las dos entregas pactadas. Se deben identificar las historias de usuario, épicas para cada <i>release</i> en base a la velocidad estimada del equipo.
	<i>Sprint planning</i>	Evidencia de la planificación de los sprints. Se debe definir el largo del sprint y mostrar la evidencia de que se planificó las HU y tareas a realizar en cada uno. Burnout chart para cada sprint.
	Gestión de riesgos	Identificación y mitigación de los principales riesgos identificados para cada sprint.
Gestión de la configuración	Control de cambios y de versiones	Estructura del repositorio, liberaciones identificadas de acuerdo a los sprints. Utilización y evidencia de uso de una herramienta de registro de defectos.
Aseguramiento de calidad	Proceso y producto	Identificación y uso de estándares.
	Pruebas de software	<p>Resultado de la ejecución de las pruebas unitarias, integración y aceptación para cada sprint realizado. Evidencia del código de pruebas unitarias, reporte de la herramienta de cobertura y análisis de los resultados.</p> <p>El documento debe cumplir con los siguientes elementos del Documento 302 de la facultad (http://www.ort.edu.uy/fi/pdf/documento302facultaddeingenieria.pdf), en especial:</p> <ul style="list-style-type: none"> – Capítulo 1, secciones 1.1 (sin la leyenda), 1.5, 1.7, 1.8 y 1.9

Escuela de Ingeniería		
Obligatorio de: Ingeniería de Software en la Práctica	Código de materia: 6454	
Fecha: 11/03/2019	Grupo: Todos	Hoja 4 de 8

		– Capítulos 4, 5 y 9
Arquitectura	Documentación de diseño	<p>Descripción de la arquitectura y diseño propuesto con su justificación correspondiente. La documentación debe presentarse en un formato adecuado al uso de metodologías ágiles y mostrando la evolución del diseño durante los sprints ejecutados.</p> <p>Arquitectura y diseño de toda la aplicación y justificación de las decisiones de diseño tomadas. El suficiente para poder entender adecuadamente el problema, hacer diagramas que aporten para comprender y documentar el diseño. Utilización de principios como los SOLID, patrones de diseño, criterios de diseño de las APIs de front end y back end, etc.</p> <p>Documentación de diseño de UI y UX.</p> <p>Modelo de tablas de la estructura de la base de datos.</p>
Implementación	Código	<p>Aplicación y evidencia de Clean Code.</p> <p>Repositorio compartido con los docentes.</p>

Las condiciones de entrega serán evaluadas como si se le estuviese entregando a un **cliente real**: prolijidad, claridad y profesionalismo.

La propia empresa eventualmente hará cambios sobre el sistema, por lo que se requiere un alto grado de mantenibilidad, flexibilidad, calidad, claridad del código y documentación adecuada.

La entrega debe ser en medio de almacenamiento CD o DVD (una sola copia como respaldo, ya que se tiene acceso al repositorio), dar acceso a los docentes al repositorio Git utilizado por el grupo y según corresponda en cada una de las dos entregas se debe incluir:

- Aplicaciones compiladas en *release* con instructivo de instalación, todo lo necesario para poder hacer el *deploy* sin utilizar Android Studio o Visual Studio.
- Se debe entregar el código fuente de todos los componentes de la aplicación así como cualquier otro producto necesario para su ejecución (servidores, bases de datos, etc.).
- Documentación entregada en **un solo** documento impreso y digital.
- Base de datos: entregar un archivo de base de datos vacío y otro con datos de prueba.

Importante: Solo se aceptará lo entregado en bedelía y en el repositorio Git, por lo que es de suma importancia revisar los medios y la documentación que se entrega antes de hacerlo.

El foco de la segunda entrega debe ser la aplicación terminada con el alcance acordado con los docentes. Si se hacen modificaciones o revisiones en el alcance y/o diseño de la aplicación, los mismos deben ser acordados con los docentes y estar debidamente documentados y aclarados en esta entrega.

Datos de prueba

El sistema debe entregarse con una fuente de datos con datos de prueba que demuestre la prueba de la solución, y de manera de poder comenzar las pruebas sin tener que definir una cantidad de datos iniciales. Dichos datos de prueba deben estar adecuadamente especificados en la documentación entregada.

Escuela de Ingeniería

Obligatorio de: Ingeniería de Software en la Práctica

Código de materia: 6454

Fecha: 11/03/2019

Grupo: Todos

Hoja 5 de 8

Pruebas unitarias

Se requiere escribir los casos de prueba automatizados con un *framework* de *unit tests*, documentando y justificando las pruebas realizadas.

Como parte de la evaluación se va a revisar el nivel de cobertura de los test sobre el código entregado. Por esto se debe entregar un reporte y un análisis de la cobertura de las pruebas.

Control de versiones

La gestión del código del obligatorio debe realizarse utilizando el repositorio Git de **Bitbucket**.

Luego de la entrega se le debe dar acceso a los docentes al repositorio. Las cuentas en Bitbucket correspondientes a los docentes son (solo es necesario darle acceso a los docentes de su dictado):

Luis Barragué – luisbarrague

Ignacio Valle – ignaciov

EVALUACIÓN

Se evaluarán los siguientes aspectos de cada trabajo:

- Porcentaje de cumplimiento con lo acordado funcionalmente.
- Funcionamiento de la aplicación, con especial énfasis en la experiencia de usuario (IU, fluidez, ergonomía, etc.).
- Calidad de diseño e implementación.
- Correcto uso de la tecnología.
- Usabilidad de la aplicación.
- Completitud y calidad de la documentación, teniendo en cuenta:
 - Documentación de análisis.
 - Documentación del proceso.
 - Descripción de la arquitectura.
 - Documentación de diseño detallada de cada componente.
 - Documentación de mecanismos de comunicación de los componentes de su solución o con terceros.
 - Justificación de las decisiones importantes de diseño e implementación.
 - Documentación o cualquier otro elemento importante de la solución.
- Se valorará especialmente la originalidad de las propuestas.

Las siguientes tablas muestran la distribución de puntos a utilizar para la corrección. Las mismas serán tomadas como guía por los docentes al momento de calificar y debe ser tenida en cuenta por los alumnos al plantear el trabajo.

RÚBRICA OBLIGATORIO 1

Escuela de Ingeniería		
Obligatorio de: Ingeniería de Software en la Práctica		Código de materia: 6454
Fecha: 11/03/2019	Grupo: Todos	Hoja 6 de 8

Tema	Descripción	%
Calidad documentación	Redacción, ortografía y forma.	15%
Alcance	Análisis del problema, documentación de los requerimientos, detalle, claridad, completitud, mockups, navegación, etc.	35%
Alcance	Originalidad de la aplicación, complejidad, uso del dispositivo (GPS, cámara, acelerómetro, almacenamiento, audio, voz, etc.).	10%
Proceso	Selección, documentación y justificación del modelo de proceso aplicado.	40%

RÚBRICA OBLIGATORIO 2

Tema	Descripción	%
Calidad documentación, alcance y proceso	Redacción, ortografía y forma. Análisis del problema. Evidencia de gestión e incorporación de las prácticas de ingeniería de software. Claridad, completitud y prolijidad de la documentación. Organización de la documentación (debe tener un orden lógico y un índice). Descripción de los cambios realizados en lo entregado respecto a la primera entrega (si corresponde y acordados con los docentes).	13%
Diseño y arquitectura	Justificación adecuada de la solución y diseño. Calidad del diseño. Modelo de tablas. Documentación de diseño de UI y UX. Informe de Clean Code y pruebas. Instructivo de instalación. Diseño de cada componente, mecanismos de comunicación, estructura de datos, etc.	18%
Implementación	Buenas prácticas de estilo y codificación y su impacto en la mantenibilidad (Clean Code). Correcto uso de las tecnologías. Claridad del código. Concordancia con el diseño.	13%
Demo al cliente	Cada equipo deberá realizar una demostración al cliente de su trabajo. Dentro de los aspectos que un cliente espera se encuentran: <ul style="list-style-type: none"> • Ver la demo cuando él esté listo y no tener que esperar a que el equipo se apronte. • Probar la solución y que la misma funcione sin problemas o con problemas mínimos, y de buena calidad de interfaz de usuario. • Funcionamiento de la aplicación, con especial énfasis en la experiencia de usuario (IU, fluidez, ergonomía, etc.). 	13%

Escuela de Ingeniería		
Obligatorio de: Ingeniería de Software en la Práctica		Código de materia: 6454
Fecha: 11/03/2019	Grupo: Todos	Hoja 7 de 8

	<ul style="list-style-type: none"> • Conocer las capacidades de cada uno de los integrantes del equipo, pudiendo preguntar a cualquier integrantes sobre la solución, su diseño, el código y sobre cómo fue construida, y así apreciar que fue un trabajo en equipo. Todos los integrantes deben conocer toda la solución y proceso. • Verificar el aporte individual al trabajo por parte de cada uno de los integrantes del equipo y en función de los resultados, se podrán otorgar distintas notas a los integrantes del grupo. Se espera que cada uno de los integrantes haya participado en la codificación de parte significativa del obligatorio. <p>Esta demostración al cliente hará las veces de defensa del trabajo.</p> <p>NOTA: El incorrecto funcionamiento de la instalación puede significar la no corrección de la funcionalidad. En el caso de defensa en el laboratorio, durante la defensa cada grupo contará con 15 minutos para la instalación de la aplicación. Luego de transcurridos los mismos se restan puntos al trabajo.</p>	
Aplicación	Porcentaje de cumplimiento con la funcionalidad acordada.	33%
Avance	Cada 3 semanas, el alumno debe presentar al docente brevemente un reporte de avance indicando lo avanzado en ese periodo	10%

RÚBRICA INFORME (CORRECCIÓN CRUZADA)

La cátedra asignará a cada grupo un obligatorio de otro grupo, de forma que puedan realizar un análisis exhaustivo y crítico del mismo en un plazo dado (ver fechas de entregas al final del presente documento). Para esto, cada grupo dará acceso a su repositorio a los integrantes del otro grupo al finalizar la última entrega (obligatorio 2).

Este informe comprende los siguientes puntos:

- Corrección de la funcionalidad acordada. Se debe ejecutar la aplicación (desde la carpeta con la aplicación compilada en release) y probar toda la funcionalidad marcando si fue implementada correctamente o no fue implementada o si fue implementada con errores (especificando los mismos).
- Evaluación del proyecto y proceso en base a la documentación, código fuente, y repositorio.
- Calidad de la documentación (análisis, proceso, diseño, arquitectura, etc.).
- Incorporación de las prácticas de ingeniería de software.
- Conformidad Clean Code. Se debe analizar el código de la solución para realizar un informe en cuanto al cumplimiento de los lineamiento de Clean Code y calidad del código.
- Pueden tomar como referencia los puntos detallados en el presente documento en la sección "Entregas" y en las "Rúbricas".

Escuela de Ingeniería

Obligatorio de: Ingeniería de Software en la Práctica

Código de materia: 6454

Fecha: 11/03/2019

Grupo: Todos

Hoja 8 de 8

Información importante

Lectura de Obligatorio: 11/03/2019

Defensas: a coordinar.

Obligatorio 1: Plazo máximo de entrega 24/04/2019

Puntaje mínimo/máximo: 0/10 puntos.

Obligatorio 2: Plazo máximo de entrega 18/06/2019

Puntaje mínimo/máximo: 15/30 puntos.

Informe: Plazo máximo de entrega 1/07/2019

Puntaje mínimo/máximo: 0/15 puntos.

Los obligatorios se forman como máximo por grupos de 3 estudiantes.

Todas las entregas se realizan exclusivamente via gestión y hasta las 20.00 hs del día de la entrega.