

Desarrollo Avanzado de Software

Trabajo final (en grupo) 40% de la nota

Descripción

Hay que desarrollar una aplicación en grupos de 3-4 personas. La aplicación tiene que resolver un reto que se describirá posteriormente añadiendo gran parte del contenido visto en la asignatura.

Para evitar problemas, **antes de comenzar el desarrollo de la aplicación** cada grupo deberá enviar un correo electrónico a iker.sobron@ehu.eus informando de quiénes conforman el grupo, cuál es la idea para resolver el reto, y las actividades y componentes principales que se piensan incluir en la aplicación.

Además de la aplicación, para la entrega final hay que realizar una presentación de 10 minutos delante del resto de la clase, explicando las características más importantes de vuestra aplicación. La presentación tendrá lugar el día **26 de mayo (lunes)** y es de asistencia obligatoria. La hora de comienzo de las presentaciones se determinará en función de la cantidad de presentaciones a realizar. La presentación también forma parte de la evaluación.

La nota obtenida en esta evaluación se corresponderá al 40% de la nota de la asignatura.

Reto UNIGO

Es un reto enmarcado en el aula Open Data de la EHU.

Objetivo: **Diseñar y desarrollar una aplicación Android para acceder al campus de Álava desde cualquier punto de la ciudad de Vitoria-Gasteiz mediante los siguientes medios:**

- A pie
- En bicicleta (Disponibles archivos de bicicarrit-bidegorri): https://www.vitoria-gasteiz.org/wb021/was/contenidoAction.do?uid=app_j34_0080&idioma=es
- En tranvía (Datos en Euskotren)
- En autobús urbano (Disponibles datos Tuvisa: <https://opendata.euskadi.eus/catalogo/-/moveuskadi-datos-de-la-red-de-transporte-publico-de-euskadi-operadores-horarios-paradas-calendario-tarifas-etc/>)

Desafío UNIGO



La aplicación puede también incluir un registro con el perfil e información relacionada al usuario en particular, así como cualquier otro servicio o funcionalidad complementaria que facilite la movilidad del estudiantado al campus.

Las mejores tres aplicaciones recibirán premios del ámbito tecnológico.

Desarrollo Avanzado de Software

Entrega

Los trabajos tienen que estar entregados antes de las **23:55 del día 18 de mayo (domingo)**, para que la semana siguiente se tenga tiempo para probar las aplicaciones. Se deben subir a la entrega de *eGela* los siguientes archivos:

- El fichero *.apk* que se genera en el directorio *build* que se encuentra en el interior del directorio del proyecto.
- Un fichero *.pdf* con una pequeña explicación de qué hace y cómo se usa la aplicación, indicando la versión de Android para la que está desarrollada la aplicación. Este documento se hará público en *eGela*, junto con el *.apk*, para que el resto de la clase pueda consultarlo.
- OTRO fichero *.pdf* que incluya un enlace a un repositorio de código Git con el proyecto completo de la aplicación. También podéis incluir alguna explicación adicional sobre el desarrollo o despliegue de la aplicación (p.e., si para funcionar necesita de servicios Web alojados en el servidor de la asignatura). Este documento no se hará público en *eGela*.

Evaluación

La nota de cada grupo se obtendrá usando la rúbrica que tenéis en la parte inferior de este documento. Cada estudiante tendrá que rellenar una rúbrica para cada grupo excepto para aquel del que forme parte.

Para obtener la nota de cada grupo se realizará una media entre las valoraciones de todos los compañeros y eso será el 50% de la nota. El otro 50% lo pondrá el profesor también mediante la rúbrica.

Desarrollo Avanzado de Software

Ejemplo de uso de la rúbrica:

	Estudiante 1	Estudiante 2	Estudiante 3	...	Profesor
Apariencia	A	A	B	...	A
Usabilidad	B	C	B	...	B
Complejidad	A	B	C	...	C
Presentación	A	B	B	...	C
Nota media según los compañeros: 7,48					Nota del profesor: 5,85
Nota final del grupo (media): 6,67					

Aspectos a puntuar:

Apariencia (10 %):

Se valorará si la estética está trabajada y se guarda cierta coherencia entre los colores, imágenes, etc. Se valorará el uso de estilos, temas, imágenes personalizadas.

Usabilidad (25 %):

Se puntuará que la app sea fácil de usar y de navegar desde la primera vez que se usa. Hay que penalizar errores de navegación (repeticiones de actividades por gestionar mal la pila de actividades, etc.), actividades que tengan funcionalidades muy parecidas, etc.

Complejidad (40 %):

En este apartado se valorará el trabajo técnico llevado a cabo por el grupo. Se tendrá en cuenta cuántas funcionalidades vistas en clase se han aplicado en la app, pero valorando el sentido práctico de las actividades (penalizando las funcionalidades que se han metido con “calzador”). Además, se valorará especialmente el hecho de incluir aspectos no trabajados en clase y su complejidad.

Presentación (25 %):

Se valorará el documento que acompaña a la app (el manual de usuario) y la presentación final que se realice.

	Sobresaliente (A)	Notable (B)	Aprobado (C)	Suspenso (D)
Apariencia (10%)	El aspecto gráfico de la aplicación está cuidado, es uniforme y diferente al dado por defecto.	La aplicación no tiene el aspecto por defecto. Tiene algún color de fondo o cosas básicas como cambio de letra, tamaño, etc.	La aplicación tiene el aspecto por defecto que genera la plantilla de Android Studio.	La aplicación no se ve correctamente porque hay errores en la disposición de elementos en IU como botones fuera de pantalla, listados sin scroll, etc.
Usabilidad (25%)	La aplicación es fácil de usar y el usuario sabe dónde está en todo momento.	La aplicación es fácil de usar, pero es difícil saber qué hace cada opción o cómo llegar a cada funcionalidad.	La aplicación funciona, pero no es sencillo su uso porque no se sabe qué hay que hacer. La aplicación es extremadamente simple debido a su poca complejidad técnica. La aplicación tiene ligeros errores que dificultan su uso.	La aplicación no funciona o da múltiples errores en su uso.
Complejidad (40%)	La aplicación usa el contenido visto en clase de manera exhaustiva y añade tres o más elementos no vistos o al menos uno con complejidad elevada.	La aplicación se limita a un uso exhaustivo de las cosas vistas en clase y a uno o dos conceptos no vistos de baja complejidad.	La aplicación se limita al uso de elementos vistos en clase.	Aunque implemente múltiples elementos vistos o no en clase, estos en su mayoría no funcionan o lo hacen de manera errónea.
Presentación (25%)	La presentación es clara en sus contenidos, está bien expuesta y realizada. El manual es claro y auto-explicativo.	El aspecto de la presentación es cuidado, pero se nota que hay errores por falta de ensayo o por desconocimiento. Aspectos de la app quedan sin explicar en el manual.	La presentación es básica y poco trabajada. El manual de funcionamiento de la app es igualmente básico.	La presentación no se hace o si se hace, hay errores de concepto. No hablan todos los miembros del equipo. No se entrega manual o si se hace, no es útil o/y resulta confuso.