

## Test y Calidad, Curso 2020-2021

### Primer proyecto de Software

La primera mitad de la parte práctica de la asignatura estará centrada en el desarrollo desde cero de un pequeño proyecto de software utilizando las diversas técnicas de test que se explican en la primera mitad del curso. Los grupos son de dos personas. El proyecto se desarrollará a lo largo de tres sesiones de prácticas (es decir, aproximadamente un mes y medio).

El primer proyecto de software a desarrollar (que se entregará antes de la cuarta sesión de prácticas, la semana del 16 de noviembre de 2020) el puede elegir de la siguiente lista:

**MasterMind, Guerra de Barcos (Hundir la flota), animal- vegetal-Mineral, Buscaminas, Candy Crush, Ajedrez, Othello, Tetris, Go, Solitario, etc.**

Si decide definir libremente un proyecto de software, debe ser un proyecto con una complejidad media que necesite de un mínimo de unas 15-20 horas de trabajo, similares a los de la lista que se propone. No se aceptarán proyectos de software de alguna asignatura en la que el proyecto viene predefinido por el profesor (y que por tanto bastantes alumnos esté desarrollando el mismo software). Es decir, debe ser un proyecto **original** suyo. Puede desarrollar un juego donde la interfaz puede ser tanto gráfica como de tipo consola. En cualquiera de los dos casos, debe desarrollar el código utilizando una arquitectura tipo **Modelo-Vista-Controlador**. Esto permite desarrollar código *testable* sin necesidad de las demás partes, es decir, debe poder desarrollar la parte del Modelo bajo el paradigma TDD (y su código de *test* asociado) sin necesidad de tener implementada la parte de la Vista. Esto implica que el código de test de la práctica tiene que estar centrado en las partes Modelo y Controlador. Se realizará test sobre la parte gráfica en la segunda práctica.

Aquellos que elija definir libremente un proyecto, envíe por e-mail (uno por cada grupo de prácticas) en su profesor de prácticas una definición escrita de su proyecto que ocupe aproximadamente una hoja. Indique cuál es el objetivo del software, que funcionalidades tendrá qué lenguaje desea utilizar, etc. Envíe el e-mail antes de la primera sesión de prácticas.

### Sesiones

Las sesiones primera, segunda y tercera serán sesiones de tutoría donde los profesores harán un seguimiento del trabajo realizado de manera autónoma. No se realizará ninguna evaluación en estas sesiones. La asistencia NO es obligatoria pero se realizará un control de asistencia.

### Entrega

El límite de entrega del proyecto es el **lunes 16 de noviembre de 2020 a las 23:55**. En la cuarta sesión de prácticas se hará una evaluación individual sobre la práctica entregada el día indicado.

En este primer proyecto se deberá utilizar un servidor de versiones estilo GitHub donde el alumno mostrará las diferentes versiones del código de prueba y del código desarrollado a medida que se aplica TDD. Cada método deberá mostrar, tantas veces como sea necesario, una versión del código de prueba y luego otra versión del código desarrollado a partir de este código de prueba. Por cada método es obligatorio realizar un mínimo de dos versiones (excepto que se trate de métodos muy simples como por ejemplo *getters* y *setters*) tanto del código de prueba como del código desarrollado para evaluar cómo se ha realizado el TDD.

El repositorio de GitHub donde tendrá su proyecto debe ser un repositorio público. Debe invitar a este repositorio los profesores KEYA Li (usuario de GitHub @ \*\*\*\*\*) en Juan Manuel Vicente (usuario @juanmav) y Javier Otazu (@xavierotazuGDS).

Tanto el código desarrollado como el código de test deben estar documentados con comentarios explicativos. Estos comentarios deben indicar, por ejemplo en un código de test, qué caso de prueba se está comprobando, si se está testeando un valor de partición equivalente o un caso límite / frontera, si se realiza una prueba de caja blanca, etc. De manera similar, el código desarrollado también debe ser comentado de manera autoexplicativa.

Se debe entregar un informe donde se explican todos y cada uno de los tipos de test realizados. En las aulas Moodle del Campus Virtual se podrá encontrar un *template* para confeccionar este informe.

Cada alumno (a nivel individual) llenará y entregará una hoja de propuesta de nota (se publicará un *template* en las aulas Moodle) donde indicará en qué temas ha trabajado en la práctica y en qué nota máxima opta.

Aquellos que estás en el grupo de practicas en inglés y desea optar a la nota NAngles (ver siguiente apartado), debe entregar todos los materiales (código, informe, etc) de manera íntegra en este idioma.

## Evaluación

Los criterios para la evaluación se publicarán por los medios habituales (Aulas Moodle del Campus Virtual).

Durante la evaluación individual de prácticas (sesión cuarta) se evaluarán algunos de los temas que el alumno ha marcado en su hoja de propuesta de nota (el profesor decidirá libremente qué temas evalúa y no necesariamente serán todos los marcados por el alumno ). El formato de la evaluación individual es libre, es decir, el profesor podrá decidir si se trata de un pequeño caso práctico y / o de un examen oral. La nota individual de la entrega del proyecto de software es

$$NLAB(i) = NEntrega(i) * NFactorEx * NTest2 * NModelVistaControlador + NIngles,$$

donde NEntrega (i) es la nota de la entrega telemática (proyecto en el servidor GitHub) teniendo sólo en cuenta los temas del proyecto en los que ha trabajado el alumno (indicados

en su hoja de autoevaluación) y NFactorEx es la nota de la evaluación individual (entre cero y 1). NModelVistaControlador tendrá un valor de 1 o 0 en caso de que se haya aplicado esta arquitectura o no, respectivamente.

El proyecto de software debe ser funcional, es decir, debe poder ejecutarse sin problemas y debe tener las funcionalidades correctamente implementadas. Se recuerda, tal como se explicó el día de la presentación de curso, que este software será testeado por otros compañeros durante la segunda parte del curso, y que el resultados de este segundo test influirá sobre la nota de esta primera entrega. Por lo tanto, si el código que se entrega en este primer proyecto no se puede testear, se penalizará la nota de esta primera entrega. Se recuerda que no se penalizará el hecho de que el testing hecho en la segunda mitad del curso descubra nuevos errores en el software de esta primera entrega, pero sí se penalizará el hecho de que no se pueda hacer testing de manera adecuada (p.ej . mock objects dentro del código desarrollado, menús u opciones que no funcionan, funcionalidades no acabadas de implementar, etc). Los criterios de corrección sobre esta entrega debidos al testing hechos en la segunda mitad del curso se publicarán en el Campus Virtual. La nota obtenida en la primera entrega se verá multiplicada después de la segunda entrega por un factor NTest2 que nunca será inferior a 0.75.

NIngles tiene un valor de 0 si no ha realizado las prácticas en un grupo en inglés, y un valor de 0.5 si la ha realizado en este grupo.