

## **Tema 0 - Presentación (DCA)**

HISTORIAL DE REVISIONES			
NÚMERO	FECHA	MODIFICACIONES	NOMBRE

## Índice

<b>1. Desarrollo Colaborativo de Aplicaciones</b>	<b>1</b>
<b>2. Objetivos</b>	<b>1</b>
<b>3. Temario</b>	<b>1</b>
<b>4. Primer período de evaluación (I)</b>	<b>2</b>
<b>5. Primer período de evaluación (II)</b>	<b>2</b>
<b>6. Cálculo de la nota final</b>	<b>3</b>
<b>7. Segundo período de evaluación (I)</b>	<b>3</b>
<b>8. Segundo período de evaluación (II)</b>	<b>3</b>
<b>9. Sistema de evaluación que se aplicará en la convocatoria de diciembre</b>	<b>3</b>
<b>10. Recomendaciones bibliográficas</b>	<b>4</b>
<b>11. Enlaces de interés</b>	<b>4</b>
<b>12. Aclaraciones</b>	<b>5</b>

Logo DLSI

**Tema 0 - Presentación (DCA)**  
**Curso 2018-2019**

## 1. Desarrollo Colaborativo de Aplicaciones

- **Asignatura optativa**
- **6 créditos ECTS**

**Aula de teoría: B01, Aulario-II**

**Laboratorios de prácticas: L23 (15:00-17:00, 19:00-21:00), Politécnica-I**

**Ficha UA**

**Web de la asignatura**

## 2. Objetivos

- Aprender a trabajar en grupo, adquirir habilidades sociales, saber identificar roles dentro de un grupo de trabajo y adaptarse a ellos, etc. . .
- Aprender a identificar y corregir los fallos de una aplicación así como recibir informes de fallos desde usuarios de la misma.
- Aprender a trabajar con ramas de desarrollo.
- Aprender a optimizar los tiempos de compilación de una aplicación.
- Aprender a internacionalizar una aplicación de manera eficiente.
- Aprender a “interrogar” (introspección) al sistema donde se compilará una aplicación para adaptarla a él, así como crear instaladores de aplicaciones para hacer más sencilla su distribución y uso por parte de usuarios noveles.
- Aprender a crear y usar tests.
- Conocer y saber usar el concepto de diseño por contrato.
- Aprender a usar sistemas de control de versiones para aplicarlos en el proceso de desarrollo del software creado, así como aprender a decidir cual de ellos puede ser el más apropiado en un momento determinado para un desarrollo concreto.

## 3. Temario

- **Tema-I:** Estructura de un equipo de desarrollo - **1 semana**
- **Tema-II:** Despliegue / Instalación - **1 semana**
  - **Versiones de desarrollo vs estables, Ramas de desarrollo / corrección de fallos, Versiones debug / release**
- **Tema-III:** Bugtracking - **1 semana**
  - **Bugzilla, Trac, tickets, issues**
- **Tema-IV:** Sistemas de paquetes - **1 semana**

- **tar, tgz, deb, rpm**
- **Tema-V:** Compilación de grandes proyectos - **2 semanas**
  - **make, distcc, ccache**
- **Tema-VI:** Configuración y adaptación automática de proyectos - **2 semanas**
  - **Autotools, CMake, Meson**
- **Tema-VII:** Generación y paso de tests - **2 semanas**
  - **junit/cppunit, boost::test, GLib.Test**
- **Tema-VIII:** Diseño por contrato - **1 semana**
  - **Precondiciones, postcondiciones, invariantes**
- **Tema-IX:** Internacionalización (I18N) - **1 semana**
  - **Gettext, ficheros .po y traducciones**
- **Tema-X:** Sistemas de Control de Versiones de última generación - **3 semanas**
  - **Centralizados vs. distribuidos, Operaciones basicas, git, mercurial, darcs**

## 4. Primer período de evaluación (I)

(HASTA PRIMER EXAMEN)

- **Realización de prácticas con ordenador (nota 1ª): Cuatro prácticas.** El profesor establecerá una nota entre 0 y 10 puntos para cada una de ellas. En la formación de la nota final del alumno la nota de la primera práctica tendrá un peso del 2 %, la de la segunda del 5 %, la de la tercera del 8 % y la de la cuarta de un 10 %. No se exigirá ninguna nota mínima para superar la asignatura respecto de esta actividad de evaluación. **Esta parte NO es recuperable.** Todas las prácticas se entregarán en **pracdlsi** en las fechas allí indicadas.
- **Primera prueba de conocimientos (nota 2ª):** Se llevará a cabo aproximadamente a mitad de semestre. Consistirá en un examen tipo test que comprenderá la materia explicada hasta ese momento. Es necesario recordar que no habrá suspensión de clases con motivo de la realización de esta primera prueba, por lo tanto, el profesor deberá realizarla durante sus horas de clase. El profesor establecerá una nota entre 0 y 10 puntos. Esta nota tendrá un peso del 25 % en la formación de la nota final del alumno. No obstante, no se podrá aprobar la asignatura si en dicho control no se obtiene al menos un 4, aunque la nota final del alumno sea superior a 5. **Esta parte SÍ es recuperable.**

## 5. Primer período de evaluación (II)

(DESDE PRIMER EXAMEN)

- **Realización de prácticas con ordenador (nota 3ª): Cinco prácticas.** El profesor establecerá una nota entre 0 y 10 puntos para cada una de ellas. En la formación de la nota final del alumno la nota de la primera práctica tendrá un peso del 7 %, la de la segunda del 5 %, la de la tercera del 3 %, la de la cuarta del 3 % y la de la quinta de un 7 %. No se exigirá ninguna nota mínima para superar la asignatura respecto de esta actividad de evaluación. **Esta parte NO es recuperable.** Todas las prácticas se entregarán en **pracdlsi** en las fechas allí indicadas.
- **Segunda prueba de conocimientos (nota 4ª):** Esta prueba se desarrollará siempre durante las tres semanas siguientes a la finalización del período de docencia, cuando ya han concluido las clases, en la fecha oficial asignada por la escuela politécnica. Consistirá en un examen tipo test que comprenderá la materia explicada en la segunda mitad del semestre. El profesor establecerá una nota entre 0 y 10 puntos. Esta nota tendrá un peso del 25 % en la formación de la nota final del alumno. No obstante, no se podrá aprobar la asignatura si en dicho control no se obtiene al menos un 4, aunque la nota final del alumno sea superior a 5. **Esta parte SÍ es recuperable.**

## 6. Cálculo de la nota final

- **Nota final = nota 1ª (25 %) + nota 2ª (25 %) + nota 3ª (25 %) + nota 4ª (25 %) = 100 %**
- El aprobado se obtendrá cuando la nota final sea igual o superior a 5,0, siempre además que el alumno haya obtenido una nota igual o superior a 4,0 en las notas 2ª y 4ª.
- Si el alumno obtiene una nota final igual o superior a 5,0, pero en una de las pruebas 2ª o 4ª su nota es inferior a 4, entonces el alumno habrá suspendido el primer período de evaluación. En este caso, el profesor pondrá al alumno una nota final de 4,5.
- Si el alumno no ha obtenido calificación en ninguna de las cuatro actividades de evaluación, entonces el profesor calificará al alumno como “ *sin presencialidad* ”.

## 7. Segundo período de evaluación (I)

- El alumno tendrá derecho a ser evaluado en un segundo período de evaluación durante el mes de julio sólo cuando haya suspendido la asignatura en el primer período de evaluación. Por lo tanto, no debe admitirse una segunda evaluación que sirva sólo para mejorar la nota de un alumno que haya aprobado.
  1. Las notas correspondientes a las primeras cuatro prácticas (nota 1ª) y de las segundas cinco prácticas (nota 3ª) no son recuperables. Por lo tanto, se utilizará siempre la misma nota que se obtuvo durante el primer período de evaluación en ellas.
  2. Las notas correspondientes a las pruebas de conocimientos primera (nota 2ª) y segunda (nota 4ª) son siempre recuperables. Por lo tanto, el profesor deberá preparar un examen que contenga dos partes diferenciadas correspondientes cada una de ellas a las materias que se evaluaron en cada una de ellas.

## 8. Segundo período de evaluación (II)

- El alumno entonces sólo tendrá que realizar las partes del examen correspondiente a las pruebas cuya nota sea menor que 4,0, de manera que la nueva nota obtenida ponderada en el 25 % para cada prueba de conocimientos, se utilizará para calcular la nueva nota final. Debe tenerse en cuenta que el alumno no podrá examinarse de la parte aprobada, ya que no cabe realizar el examen con el único objetivo de mejorar la nota de una parte ya aprobada. Por lo tanto, la nota obtenida en el primer período de evaluación respecto de la prueba que obtuvo una nota mayor o igual que 4,0 será necesariamente la que se tendrá en cuenta para calcular la nueva nota final, ponderada lógicamente en el 25 %.
- De igual modo que sucede en el primer período de evaluación, el alumno sólo aprobará la asignatura si la nueva nota final es igual o superior a 5,0 y además el alumno ha obtenido una nota igual o superior a 4,0 en cada una de las partes de las que se haya examinado.
- Caso especial: cuando el alumno hubiera obtenido una nota final inferior a 5,0 en el primer período de evaluación, pero hubiera aprobado las dos pruebas de conocimientos, podrá examinarse de nuevo de las dos partes del examen con el objetivo de darle la oportunidad de superar la asignatura en el segundo período de evaluación.

## 9. Sistema de evaluación que se aplicará en la convocatoria de diciembre

- Un examen tipo test que comprenderá la materia explicada en toda la asignatura (el peso de su puntuación será el mismo que el de ambas pruebas de conocimientos) y un examen tipo test que comprenderá los conocimientos adquiridos a lo largo de todas las prácticas con ordenador hechas durante toda la asignatura (el peso de su puntuación será el mismo que el de las prácticas con ordenador).

## 10. Recomendaciones bibliográficas

1. Managing the Testing Process: Practical Tools and Techniques for Managing Hardware and Software Testing. Rex Black. Ed. Wiley. ISBN-10: 0470404159. ISBN-13: 978-0470404157. Edition: 3
2. Implementing Automated Software Testing: How to Save Time and Lower Costs While Raising Quality Elfriede Dustin, Thom Garrett, Bernie Gauf. Ed. Addison-Wesley Professional. ISBN-10: 0321580516. ISBN-13: 978-032158051. Edition: 1
3. Software Project Management For Dummies. Teresa Luckey, Joseph Phillips. Ed. For Dummies. ISBN-10: 0471749346. ISBN-13: 978-0471749349, 1 edition
4. Guía Rápida de Bugzilla (Spanish Edition). Adrián Ruffinatti. Ed. Amazon Digital Services, Inc. ASIN: B0077MBO1S.
5. Mastering Cmake. Ken Martin, Bill Hoffman. Ed. Kitware, Inc.. 5th edition. ISBN-10: 193093422X. ISBN-13: 978-1930934221.
6. Autotools: A Practioner's Guide to GNU Autoconf, Automake, and Libtool. John Calcote. Ed. No Starch Press; 1 edition. ISBN-10: 1593272065. ISBN-13: 978-1593272067.
7. Managing Projects with GNU Make (Nutshell Handbooks). Robert Mecklenburg. Ed. O'Reilly Media; Third Edition edition. ISBN-10: 0596006101. ISBN-13: 978-0596006105.
8. The Boost C++ Libraries. Boris Schäling. Ed. XML Press. ISBN-10: 0982219199. ISBN-13: 978-0982219195.
9. GNOME 3 Application Development Beginner's Guide. Mohammad Anwari. Ed. Packt Publishing. ISBN-10: 1849519420. ISBN-13: 978-1849519427.
10. Design by Contract, by Example. Richard Mitchell, Jim McKim. Ed. Addison-Wesley Publishing Company; 1st edition. ISBN-10: 0201634600. ISBN-13: 978-0201634600.
11. Pro Git (Expert's Voice in Software Development). Scott Chacon. Ed. Apress; 1 edition. ISBN-10: 1430218339. ISBN-13: 978-1430218333.
12. Mercurial: The Definitive Guide. Bryan O'Sullivan. Ed. O'Reilly Media. ISBN-10: 0596800673. ASIN: B00CVE1FNO.

## 11. Enlaces de interés

1. <https://code.google.com/p/distcc/>
  2. <http://en.wikipedia.org/wiki/Distcc>
  3. <http://ccache.samba.org/>
  4. <http://es.wikipedia.org/wiki/Trac>
  5. <http://www.bugzilla.org/>
  6. <http://neugierig.org/software/chromium/notes/2011/02/ninja.html>
  7. <http://wiki.debian.org/HowToPackageForDebian>
  8. <http://www.debian.org/doc/manuals/maint-guide/build.en.html>
  9. [http://fedoraproject.org/wiki/How\\_to\\_create\\_an\\_RPM\\_package/es](http://fedoraproject.org/wiki/How_to_create_an_RPM_package/es)
  10. <http://es.wikipedia.org/wiki/Gettext>
  11. <http://www.gnu.org/software/gettext/>
  12. <http://git-scm.com/>
  13. <http://git-scm.com/book>
-

14. <http://www.monotone.ca/>
15. <http://mercurial.selenic.com/>
16. <http://en.wikipedia.org/wiki/Mercurial>
17. <http://en.wikipedia.org/wiki/Darcs>
18. <http://darcs.net/>
19. <http://www-cs-students.stanford.edu/~blynn/gitmagic/>
20. <http://gitready.com/>
21. [http://www.boost.org/doc/libs/1\\_53\\_0/libs/test/doc/html/index.html](http://www.boost.org/doc/libs/1_53_0/libs/test/doc/html/index.html)
22. [http://sourceforge.net/apps/mediawiki/cppunit/index.php?title=Main\\_Page](http://sourceforge.net/apps/mediawiki/cppunit/index.php?title=Main_Page)
23. <http://es.wikipedia.org/wiki/JUnit>
24. <http://junit.org/>
25. <https://developer.gnome.org/glib/2.36/glib-Testing.html>
26. <http://www.gnu.org/software/make/manual/make.html>
27. <http://www.gnu.org/software/autoconf/manual/autoconf.html>
28. <http://sources.redhat.com/automake/automake.html>
29. <http://www.cmake.org/>
30. [http://www.cmake.org/cmake/help/cmake\\_tutorial.html](http://www.cmake.org/cmake/help/cmake_tutorial.html)
31. [http://en.wikipedia.org/wiki/Design\\_by\\_contract](http://en.wikipedia.org/wiki/Design_by_contract)
32. <http://dlang.org/dbc.html>
33. [https://live.gnome.org/Vala/Tutorial#Assertions\\_and\\_Contract\\_Programming](https://live.gnome.org/Vala/Tutorial#Assertions_and_Contract_Programming)
34. <http://www.informit.com/articles/article.aspx?p=23953>
35. <http://archive.cs.st-andrews.ac.uk/STSE-Handbook/SoftwareDevAndDeploy/>
36. <http://www.ambysoft.com/essays/deploymentTips.html>
37. [http://en.wikipedia.org/wiki/Software\\_deployment](http://en.wikipedia.org/wiki/Software_deployment)

## 12. Aclaraciones

EN NINGÚN CASO ESTAS TRANSPARENCIAS SON LA BIBLIOGRAFÍA DE LA ASIGNATURA.

- Debes estudiar, aclarar y ampliar los conceptos que en ellas encuentres empleando los enlaces web y bibliografía recomendada que puedes consultar en la página web de la [ficha de la asignatura](#) y en la [web propia de la asignatura](#).