

**Normes dels lliuraments de Projectes de Programació**  
**Darrera modificació: 1/setembre/2025**

**Normes generals:**

1. Si no es diu el contrari, els lliuraments es faran sempre abans de les 23:59 del dia previst, via Racó. En el cas del tercer lliurament, es farà també una correcció interactiva posterior.
2. El lliurament que sigui incomplet, que es faci després de la data prevista o en format incorrecte és equivalent al no lliurament.
3. El tutor pot demanar revisions al primer lliurament i fixar un termini raonable per rebre-les. La primera repetició d'un lliurament serà puntuada sobre 6 (en comptes de sobre 10). Una eventual segona repetició serà puntuada sobre 4.
4. Cada classe estarà programada per una sola persona, i cada membre de l'equip ha de programar aproximadament el mateix. En cas de desviació important, es podrà considerar que l'estudiant no s'ha presentat a l'assignatura.
5. Els programes s'han de poder compilar i executar des de terminal en la versió 21.\* de Java, i no poden tenir errors de lectura (cal portar més d'una còpia com a precaució). L'incompliment d'aquesta norma farà que es consideri no lliurat el projecte. Així mateix, s'utilitzarà la versió 4.\* de la llibreria JUnit.
6. Es obligatori l'ús del repositori GitLab de la FIB. Es crearan comptes per cada equip. Es proporcionarà un template de projecte Gradle que podeu usar com a punt de partida.
7. Al lliuraments interactius s'hi han de presentar tots els membres de l'equip. La no assistència implicarà que l'estudiant no s'ha presentat a l'assignatura.
8. Si un estudiant d'un equip deixa de complir amb les seves responsabilitats, els altres han d'avalar immediatament el tutor perquè reorganitzi la feina. No s'ha d'esperar fins al final del lliurament per avisar del problema.
9. Les hores destinades pel professor a tutoritzar projectes són exclusivament les de les classes de laboratori. Les hores de consulta es destinen solament a qüestions generals de l'assignatura.
10. Tot lliurament de documentació tindrà una portada amb la següent informació:
  - Nom de la pràctica
  - Identificador de l'equip
  - Nom, cognoms i username de cada membre de l'equip (no el dni)
  - Versió del lliurament 9.9 (número de lliurament.revisió 0 o 1)
  - Totes les pàgines han d'estar numerades.
  - Després de la tapa, un índex del contingut amb la numeració de cada apartat.
11. Per incorporar al projecte qualsevol codi (font, objecte, executable, o llibreries) no desenvolupat per l'equip, caldrà comptar amb el permís explícit del tutor, per escrit o per correu electrònic. Això inclou el codi generat per eines altres que el compilador del llenguatge de referència, com per exemple, programes per dissenyar interfícies que

generen codi font o objecte. S'exceptua el codi instal·lat junt amb el llenguatge de referència en els PCs de les sales d'usuaris de l'LCFIB, i aquell que es designi explícitament en una llista que es podrà publicar a la pàgina de l'assignatura. Incorporar codi extern al projecte sense permís es considerarà un intent d'engany i estarà subjecte a la consideració de còpia recollida en les normatives acadèmiques de la universitat. En cas de dubte, cal consultar el tutor.

---

### **Contingut i puntuació de cada lliurament:**

Cada tutor pot especificar que algun document s'ha de lliurar en paper a més de via Racó.

Primer lliurament (40% de la nota final del projecte)<sup>1</sup>:

- Diagrama de casos d'ús (fet amb Rational Rose o eina similar) i breu descripció de cada cas d'ús (fet amb processador de textos).
- Diagrama estàtic complet del model conceptual de dades en versió disseny (fet amb Rational Rose o eina similar) i breu descripció d'atributs i mètodes de cada classe (*setters* i *getters* es poden obviar).
- Relació de les classes implementades per cada membre de l'equip.
- Breu descripció de les estructures de dades i algorismes utilitzats per a implementar les funcionalitats principals d'aquesta entrega.
- Codi de totes les classes del model.
- Codi de les classes de domini associades a les funcionalitats principals de l'aplicació. Aquestes funcionalitats principals es detallaran a l'enunciat.
- Tests unitaris de JUnit per totes les classes del model (amb *stubs* o *mocks* si és necessari). Els test cases inclosos han de proporcionar una bona cobertura (comprovar casos convencionals i casos extrems).
- Executables que permeten provar les funcionalitats principals implementades, mitjançant *drivers* interactius que utilitzin els controladors pertinents del domini.
- Tests (jocs de proves) que l'equip ha fet servir per provar les funcionalitats principals (a afegir als *drivers*: una cosa no exclou l'altre)

Segon lliurament (15% de la nota final del projecte)<sup>2</sup>:

- Els tres diagrames següents (separadament, cadascun en un arxiu diferent):
  - Domini i controladors del domini.
  - Vistes i controladors de presentació.
  - Gestors de dades.

---

<sup>1</sup> Veure punt 7.1 de la normativa d'estructura dels lliuraments

<sup>2</sup> Veure punt 7.2 de la normativa d'estructura dels lliuraments

- Breu descripció d'atributs i mètodes de cada classe de la capa de domini (*setters* i *getters* es poden obviar). Breu descripció de cada classe de les altres dues capes.
- Breu descripció de les estructures de dades i algorismes utilitzats per a implementar les funcionalitats principals de tot el projecte.

Tercer lliurament (45% de la nota final del projecte)<sup>3</sup>:

- El dia de la correcció interactiva, hi ha solament una prova. Cada error que es trobi baixarà nota en funció de la seva gravetat.
  - Manual d'usuari.
  - Relació de les classes implementades per cada membre de l'equip.
  - Codi de totes les classes del projecte.
  - Executable del programa. Jocs d'assaig complets per a provar-lo.
  - Opcional (recomanat): carpeta de documentació generada per Javadoc/Doxygen o similar.
- 

### Estructura dels lliuraments en format digital

1. En el directori arrel hi haurà un fitxer amb la descripció de l'equip. Contindrà el nom i dos cognoms dels membres de l'equip (en el format: cognoms, nom), un per línia, en ordre alfàbetí. Després hi haurà les adreces e-mail de la facultat (és a dir aquelles que acaben amb @estudiantat.upc.edu) de tots els membres de l'equip, en el mateix ordre que els cognoms, és a dir la primera adreça corresponderà al primer alumne, la segona al segon... també una per línia. Finalment podeu afegir-hi adreces e-mail alternatives (upcnet, gmail, etc.) també una per línia, però en l'ordre que vulgueu.
2. Tot directori contindrà un fitxer de nom index.txt (format text pla i sense accentuar) o index.htm o index.html amb la descripció del directori i de tots els seus fitxers –un per un– on podeu afegir les observacions especials que vulgueu per facilitar-ne la comprensió o l'ús.
3. Si es vol usar un compressor de fitxers no estàndard (estàndards es consideren els .zip, .jar, .gz i .tgz) per reduir els requisits d'espai en disc del lliurament, cal consultar amb el tutor si cal adjuntar l'eina necessària per a la descompressió en el directori arrel. El fitxer amb la descripció de l'equip, el fitxer index.txt (index.htm o index.html) del directori arrel i l'eina de descompressió estaran sense comprimir.
4. En tots els textos cal tenir molta cura amb l'ortografia. No useu accents en els fitxers .txt.
5. No podeu suposar *paths* absoluts.
6. Tot lliurament amb virus serà considerat com a no presentat.
7. Estructura dels directoris.
  - 7.1. El lliurament 1 tindrà tres directoris amb el següent nom i contingut:

---

<sup>3</sup> Veure punt 7.3 de la normativa d'estructura dels lliuraments

- DOCS: Els documents ja descrits en el contingut de l'entrega. Aquest directori no conté cap codi.
- FONTS: Classes implementades per l'equip fins al moment (incloent-hi *stubs* i *drivers*). Els subdirectoris han de seguir l'estructura de *packages*, perquè el codi sigui recompilable directament. S'ha d'incloure un fitxer de comandes (.bat, .sh o makefile) amb les opcions per compilar els *drivers*.

En cas d'estar usant Gradle, es pot proporcionar directament el seu directori arrel, incloent-hi els directoris *bin* i *src*.

- EXE: Executables per provar les classes o funcionalitats implementades i jocs d'assaig. Hi haurà un subdirectorí per cada classe a provar, amb els executables necessaris, les dades del joc d'assaig i la descripció d'aquest (veure l'apartat de descripció dels jocs de proves).

7.2. El lliurament 2 tindrà un directori amb el següent nom i contingut:

- DOCS: Els documents ja descrits en el contingut de l'entrega. Aquest directori no conté cap codi.

7.3. El lliurament 3 tindrà tres directoris amb el següent nom i contingut:

- DOCS: Els documents ja descrits en el contingut de l'entrega. Aquest directori no conté cap codi.
- FONTS: Classes definitives del projecte. Els subdirectoris han de seguir l'estructura de *packages*, perquè el codi sigui recompilable directament. S'ha d'incloure un fitxer de comandes (.bat, .sh o makefile) amb les opcions per compilar l'aplicació completa. No cal tornar a incloure els *stubs* i *divers* de la primera entrega.

En cas d'estar usant Gradle, es pot proporcionar directament el seu directori arrel, incloent-hi els directoris *bin* i *src*.

- EXE: Executable/s del projecte i jocs d'assaig complets.

Tot lliurament que deixi de satisfer una d'aquestes condicions serà penalitzat amb fins a 2 punts i podrà ser retornat per defecte de forma.

Tota millora que, sense contravenir aquestes condicions, faciliti la lectura i comprensió del contingut del lliurament repercutirà positivament a la nota.

---

### Descripció dels jocs de proves

Cada *driver* executable del primer lliurament (exceptuant tests unitaris) ha d'anar acompanyat d'un text que descrigui:

- **Objecte de la prova:** casos d'ús i classes, o integració de conjunts, provats anteriorment, que es proven
- **Altres elements integrats a la prova:** classes ja provades o que s'han provat d'alguna altra manera –reutilitzats, etc.– integrades en aquest executable (si n'hi ha)
- **Fitxers de dades necessaris:** nom dels fitxers de dades, si n'hi ha, i en tal cas les

classes que s'emmagatzemen en cadascun

- **Valors estudiats:** amb quins valors es prova i quina missió tenen, aquí és on s'hauria de parlar de caixa blanca o negra, conjunts de dades de comportament homogeni, etc.
- **Efectes estudiats:** funcionalitats com ara vistes, moviments per la pantalla, selecció en llistes, etc. que no són dades concretes
- **Operativa:** petit manual o descripció del funcionament de l'executable, molt bàsica

L'executable del projecte del tercer lliurament ha d'anar acompanyat d'un document que descrigui exhaustivament les proves efectuades, incloent-hi per a cadascuna:

- **Objecte de la prova:** casos d'ús que es proven
- **Fitxers de dades necessaris:** nom dels fitxers de dades, si n'hi ha, i en tal cas les classes que s'emmagatzemen en cadascun
- **Valors estudiats:** amb quins valors es prova i quina missió tenen, aquí és on s'hauria de parlar de caixa blanca o negra, conjunts de dades de comportament homogeni, etc.
- **Efectes estudiats:** funcionalitats com ara vistes, moviments per la pantalla, selecció en llistes, etc. que no són dades concretes