

BLOC A

Creació d'Interfícies amb IA

Track Product Owners

Programa de formació en IA · Àrea d'Enginyeria i Desenvolupament

Corporació Catalana de Mitjans Audiovisuals

Maig 2026

Avui aprenem a anar més ràpid.

Del problema a la hipòtesi en una hora, no en una setmana.



Una hipòtesi de producte

La idea inicial: 'i si tinguessim una eina per a X?'



Un prototip funcional

En 30-60 minuts. Iterable amb l'usuari davant.



User stories per dev

5-10 stories format Jira llestes per a programar.

Què és aquest bloc, què no és

Per a evitar expectatives equivocades.

ÉS

- ✓ Velocitat per a anar del problema a la hipòtesi
- ✓ Eina per a sessions amb usuaris en directe
- ✓ Generació automàtica de user stories
- ✓ Capacitat de validar idees abans d'invertir

NO ÉS

- ✗ Substitut de l'equip UX/UI
- ✗ Substitut de l'equip de development
- ✗ Eina per a generar producte final
- ✗ Solució per a tot el cicle de producte

Panorama d'eines

Quina trio per a què. Avui ens centrarem en la primera.

Lovable

Aplicacions web completes

Iteració conversacional
URL pública compartible

*Sessions amb usuaris
Prototips ràpids*

v0 (Vercel)

Components UI individuals

Estètica polida
Codi Next.js/React copiable

*Components específics
Disseny inspirat en una imatge*

Bolt.new

Apps amb codi descarregable

Similar a Lovable
ZIP descarregable per a dev

Quan vols entregar codi al dev directament

Copilot Agent

Generació dins l'IDE

Per a devs
Skills i Design System de l'organització

Sessions conjuntes amb dev (no stand-alone per PO)

4 regles del bon prompt

Aplicables a Lovable, v0, Bolt o qualsevol altra eina conversacional.

1

Concret > abstracte

'Un panell amb 3 columnes' funciona millor que 'un panell modern'.

2

Domini explícit

Sempre menciona 3Cat / mitjà audiovisual. Posa el context.

3

Comportaments, no només aparença

Diu què passa quan algú clica, no només com es veu.

4

Iteració curta

Comença simple. Afegeix complexitat al següent prompt.

Avís de confidencialitat

Tot el que envieu a aquestes eines es processa als seus servidors.



Regles d'ús a 3Cat



Mai dades reals d'usuaris

Useu placeholders. Mai noms, emails, IDs interns reals.



Mai informació estratègica confidencial

Acords amb proveïdors, números sensibles, contrasenyes.



Prototips amb URL pública

Accessibles a qualsevol que conegui la URL. Configureu privacitat.



Per a casos amb dades sensibles reals

Consulteu Plataforma o Seguretat ABANS d'usar les eines.

Instruccions a Lovable

El que vau aprendre al Bloc 2 existeix a Lovable, amb un altre nom.

COPILOT (BLOC 2)

GitHub Copilot

.github/copilot-instructions.md

fitxer de configuració

.prompt.md

fitxer de skill per component

Repo GitHub

on viuen les skills

Copilot el llegeix

en generar codi a l'IDE

Developers + UX/UI

qui el manté



*mateix
concepte*

LOVABLE

Lovable

Knowledge

secció de configuració del projecte

Secció per component

dins del mateix Knowledge

Project Settings

on viu el Knowledge

Lovable el llegeix

a cada prompt que envies

POs + UX/UI

qui el manté

Skills a Lovable

El mateix concepte que vas aprendre al Bloc 2, aplicat component a component.

BLOC 2 · ARQUITECTURA EN 3 CAPES

Capa Global

Tokens, idioma, convencions d'organització

Capa Producte

Instruccions per component (.prompt.md)

Capa Subproducte

Variants i casos d'ús específics

Governança

PR + review UX/UI. Skills com a codi.



*mateix
patró*

LOVABLE · KNOWLEDGE

Tokens i regles globals

Colors, espaiat, llengua, vocabulari 3Cat

Secció per component

Button · ChannelBadge · ProgramCard · LiveIndicator

Variants i casos d'ús

Quan usar-lo, quan NO, errors típics de la IA

Actualització

UX/UI vàlida. Si el DS canvia, la skill canvia.

Exportació des de Lovable

El prototip no queda tancat dins Lovable. Aquí teniu com treure'l.

1

URL pública compartible

Cada prototip té una URL activa al moment. Envieu-la per Slack o porteu-la a una reunió amb usuaris. Cap instal·lació, cap login per a qui la rep.

Immediatament disponible

2

Connexió a GitHub

Lovable pot sincronitzar el codi generat amb un repositori de GitHub. Els developers clonen el repo i continuen des d'on el PO ha deixat el prototip.

Recomanat per a 3Cat

3

Exportació ZIP (codi font)

Descàrrega directa del codi React generat com a fitxers .tsx i .css. Útil si no voleu usar GitHub o voleu inspeccionar el codi abans de lliurar-lo.

Alternativa sense GitHub

4

Connector Atlassian → User Stories a Jira

Un cop validat el prototip amb usuaris, Lovable pot generar automàticament User Stories en format Jira amb criteris d'acceptació. Sense copiar i enganxar manualment.

El que canvia el flux de treball

Enterprise: el Design System com a organització de fitxers

El mateix concepte que vas veure al Bloc 2, però natiu a Lovable. Sense configuració extra.

```
.lovable/
├── system.md (~500 línies màx)
│   ├── Installation instructions (10 línies)
│   │   └── Package setup · Env variables · .npmrc
│   ├── Asset links (fonts, icones, ...)
│   ├── Directory guide (20 línies)
│   └── General guidelines
│       └── Code patterns · Design principles
└── rules/
    ├── components/
    │   ├── button.md input.md modal.md
    │   └── patterns/
    │       ├── forms.md navigation.md
    │       └── styling/
    │           ├── colors.md typography.md
```

EQUIVALENT AL DS DE SKILLS

system.md	=	copilot-instructions.md (Capa 1 Org Global)
rules/components/ button.md	=	components/ button.md (Bloc B)
rules/styling/ colors.md	=	components/ tokens.json (Bloc B)

Per què és important

- Actualitza un fitxer → tots els prototips futurs ja el respecten
- Les skills del Bloc B van aquí directament

Tier Enterprise: la carpeta **.lovable/** viu al repo, es versiona amb git i es revisa com qualsevol altre fitxer de codi.

Connectors

Com els MCPs que han construït els developers, però per a vosaltres. Sense codi.

Supabase

Base de dades real

El prototip guarda i llegeix dades reals.
Formularis que persisteixen, llistes que s'actualitzen.

Stripe

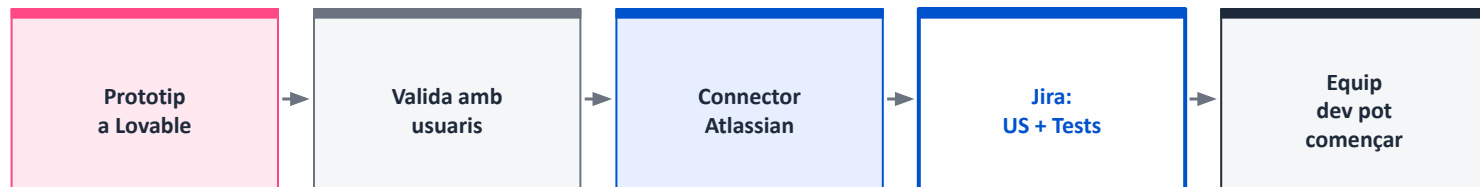
Pagaments reals

El prototip pot fer transaccions reals.
Útil per a prototips de subscripció o e-commerce.

★ Atlassian / Jira

El que canvia el flux de treball

- User Stories generades automàticament al Jira
- Tests d'acceptació inclosos a cada ticket
- Criteris d'acceptació



El cicle complet: *del prototip validat als tickets de Jira, sense sortir de Lovable ni obrir cap eina addicional.*

El cas d'ús que treballarem

Una eina interna de gestió de continguts. Cas inspirat en eines reals de 3Cat.



Dashboard amb sidebar

Panell intern amb seccions: Continguts, Calendari, Drets, Estadístiques, Configuració



Formulari de contingut

Crear continguts amb metadades: títol, tipus, canal, dates, durada, drets, subtítols



Graella de programació

Vista setmanal amb programes per franja horària. Codis de colors per tipus.



Generació de stories

Convertir el prototip en 5-10 user stories format Jira

Alternativa: repo real + GitHub Copilot Enterprise

React/Next.js sobre el repositori de producció, sense Lovable.

AVANTATGES

✓ Zero gap de sincronització

El que el PO genera és codi final. No hi ha traducció de prototip a producció.

✓ DS, skills, MCPs i agents ja integrats

Tota la infraestructura del Bloc 2 i 3 disponible sense configuració extra.

✓ Codi reutilitzable pels developers

El PO contribueix directament al repo. No hi ha descart de feina.

✓ Historial de decisions al git

Cada iteració queda al log. Es pot revertir, ramificar, comparar.

✓ Menys capes d'adaptació

Del prompt al component React: un sol pas. Sense exportació ni copia-enganxa.

✓ El PO aprèn vocabulari real

Components, props, rutes. Millora la comunicació amb developers a llarg termini.

DESAVANTATGES

! Més passos tècnics per al PO

Node, git, editor, npm run dev... Lovable és un navegador. Barrera d'entrada real.

! Velocitat d'iteració menor

Cada canvi requereix commit, potser PR, potser review. Explorar és més lent.

! Sense preview en directe

Lovable mostra el resultat al moment. Aquí cal interpretar codi. Menys 'wow effect'.

! Risc de conflictes de merge

Treballar en paral·lel amb developers complica la gestió de branques.

! Planificació de convencions

Cal definir on va el codi del PO, quines branques usa, quan es revisa.

! Menys paral·lelisme

El PO bloqueja o és bloquejat pels developers si treballen al mateix fitxer.

Recomanat quan: el PO té context tècnic (sap usar git i un editor) i el prototip ha d'acabar directament a producció sense pas intermedi. **Per a la majoria de POs de 3Cat, Lovable primer.**

Que us emporteu del Bloc A

1

Passos per treballar amb Lovable

Estructura de 4 regles + prompts per iteració. Adaptable a v0 i Bolt. Punt de partida real per a la pròxima sessió amb usuaris.

2

Com configurar el Knowledge al vostre projecte

Paral·lelisme entre el que hem après als Blocs 1-4 i el flux de treball de Lovable.

3

Un flux de treball validat: prototip → Jira

Connector Atlassian. 5–10 user stories en format Jira amb criteris d'acceptació, llestes per a l'equip de dev. Sense pas manual.

4

Una experiència real d'iteració

Heu provat l'eina amb un cas real de 3Cat, no un exemple inventat. I sabeu quan usar Lovable i quan passar directament al repo amb Copilot.

PROPER PAS

Ens veiem al

Hackathon!

prepareu les idees que voleu desenvolupar

i organitzeu els grups de treball
