

Projecte IPOP – Planificació i organització del treball en equip



Autors: Xavier, Climent, Martí i Fabián

Aquest document recull una **visió tècnica exhaustiva del projecte**, oferint un recorregut pel desenvolupament de les funcionalitats sprint rere sprint, exposant les principals dificultats tècniques que hem trobat i com les hem superat, i descrivint els requisits i enfocaments tecnològics que han sostingut la construcció del sistema.

Evolució de les funcionalitats al llarg dels Sprints

El desenvolupament s'ha estructurat en diferents sprints que han permès una evolució progressiva del producte.

- **En els primers sprints**, ens vam centrar a establir una **base tècnica sòlida**: la configuració de l'entorn de treball, la definició dels primers components, la connexió amb la base de dades i l'establiment d'una arquitectura clara i modular.
 - **A mesura que avançava el projecte**, vam afegir funcionalitats essencials com la navegació entre vistes, la gestió d'usuaris, l'enviament i la recepció de dades entre el frontend i el backend, així com la visualització d'aquestes dades a l'interfície d'usuari.
 - També vam implementar la **interacció amb dades dinàmiques** provinents de simulacions (com sensors de temperatura, CO₂ i volum), integrant un sistema de pop-ups interactius amb codificació per colors, permetent una interpretació ràpida dels valors.
 - **Als darrers sprints**, ens vam enfocar en millores d'usabilitat, comprovacions d'errors, i proves d'usuari per garantir una experiència fluida i estable. També es van optimitzar els temps de càrrega i es van revisar detalls visuals i tècnics.
-

Problemes i solucions

Durant el desenvolupament vam haver d'afrontar diversos **problemes tècnics** que van posar a prova la nostra capacitat de resolució i adaptació.

Projecte IPOP – Planificació i organització del treball en equip



Autors: Xavier, Climent, Martí i Fabián

1. Problemes de comunicació entre el frontend i el backend

Inicialment vam tenir dificultats per aconseguir una comunicació fluida entre el frontend (fet amb Vue.js) i el backend (implementat amb Node.js i Express). Hi havia errors de CORS, malentès en les rutes d'API i problemes amb l'enviament de dades JSON correctes.

Solució: vam revisar tota l'estructura del backend, afegint headers CORS adequats, ajustant els mètodes HTTP utilitzats (GET, POST...) i validant els objectes enviats des de Vue amb eines de debugging com Postman.

2. Gestió de l'estat i dades en temps real

Una altra dificultat va ser mantenir **actualitzada la informació mostrada** en pantalla (dades de sensors simulats, colors dinàmics, etc.). Inicialment, els canvis no es reflectien correctament, o es desincronitzaven entre components.

👉 **Solució:** vam aplicar patrons de gestió d'estat utilitzant variables reactives, emetent esdeveniments, en alguns casos, reestructurant el flux de dades per evitar bucles o actualitzacions incorrectes.

3. Problemes de compatibilitat i rendiment amb les imatges del mapa

Les imatges dels plànols del centre eren grans i en alguns dispositius no es carregaven bé o es desplaçaven malament. A més, els pop-ups es posicionaven incorrectament segons la mida de la pantalla.

👉 **Solució:** vam reescalar les imatges, ajustar el de càrrega, afegir scroll dinàmic i adaptar els punts interactius a posicions relatives. També vam millorar el rendiment carregant imatges de manera asíncrona i optimitzant el component de mapa amb

Projecte IPOP – Planificació i organització del treball en equip



Autors: Xavier, Climent, Martí i Fabián

4. Errors visuals i validacions incompletes

Alguns inputs no tenien validació, i es permetia enviar dades errònies o buides, generant errors interns.

👉 **Solució:** vam implementar validació de formularis tant al frontend com al backend, afegint gestió d'errors clara i missatges d'advertència a l'usuari. També vam revisar cada interacció per garantir que les dades fossin consistents.

Aspectes tècnics i requisits

El projecte ha estat desenvolupat amb una arquitectura basada en components reutilitzables utilitzant Vue.js per al frontend, i Node.js amb Express per al backend. La base de dades principal ha estat MySQL per a la gestió d'usuaris i entitats principals, mentre que MongoDB s'ha utilitzat per al registre de dades estadístiques provinents dels sensors (simulats).

S'han tingut en compte els següents requisits tècnics:

- **Modularitat i separació de responsabilitats** clares entre frontend i backend.
- **Connexió a base de dades** robusta amb mètodes segurs i protegits.
- **Autenticació d'usuaris** mitjançant tokens JWT i encriptació de contrasenyes amb bcrypt.
- **Simulació de dades en temps real**, tractant i representant les dades amb codificació visual (colors i estils).
- **Testing d'usuari** i depuració contínua per millorar l'experiència d'usuari.

El resultat és una aplicació tècnicament estable, clara i funcional, capaç d'adaptar-se i escalar si fos necessari, mantenint sempre una estructura clara i mantenible.