## 

## 

## Visió global

## Gestió projecte

* Taiga amb èpiques, sprints, històries i tasques
* Com fem els punts? (planning poker -> 1,2,3,5,8,13,21,34 coffee)
* A quan equival 1 punt = 1 hora
* DoD

## Gestió versions

* Utilització de github → Descriure els repositoris que tindrem
* Representació dels commits (com les tagejem i les linkegem amb les tasques del taiga)
* Gestió de branques
* Estructura de carpetes
* Utilització d'issues per a crear feature per èpica i subbranques per historia?

## Comunicació equip

* Utilització discord
* Grup de whatsapp
* 3 reunions a la setmana, una d’elles review
* retros

## Gestió de la qualitat

* Utilització de intellij code analysis

## Gestió de proves

* Tests unitaris
* Jocs de prova
* Github CI

## Gestió de configuracions

Utilització de github CD

com despleguem

## Interacció amb altres companys → Isslam

* Responsable de l’equip
* Interaccions continues

## Gestió de bugs

* Report detallat a discord
* Utilització resolució de bugs a github (issues -> bug)

## Ides:

* Pycharm
* WebStorm

## Llenguatges

* React native (javascript) o Flutter (Dart) o Android natiu (Java) -> app
* Django (python) o Java Servlet (java) o express (javascript) o Spring Boot (java) o Ruby on Rails (Ruby) o …-> Backend
* React (javascript) o sense framework o …-> admin panel

## Arquitectura física

* Muntem sobre dockers?
* Necessitem:
  + Allotjament backend
  + Allotjament ddbb
  + Allotjament frontend web admin
* APIS?

## Gestió BBDD

* Utilització de migrations
* SQL o NoSQL?
* Una bbdd o varies?
* ús de SPs

## SERVEIS

* A obtenir
  + Google Auth
  + Raco auth
  + Estacions de carrega publiques
  + Google maps
  + Cotxes elèctrics
  + Google places
  + Contaminació
* A donar
  + Navegació per mapa de punt A a punt B

## Requeriments no funcionals:

Llista històries d’usuari classificades per èpiques → Anotar-les al taiga amb descripció i criteris d’acceptació → UML