1. **Analyse fonctionnelle (2h)**
2. *Reformulation du besoin*

La société NEGOSUD est un négociant en vin situé en Gascogne.

M. CLAIN, œnologue de métier, de par ses relations avec des domaines viticoles, a décidé d’ouvrir un entrepôt pour favoriser la découverte et la vente des produits du terroir, dans un lieu fortement touristique, en complément de ces domaines qui vendent déjà sur leurs propriétés.

En accord, avec ses partenaires (*Les domaines de Tariquet, Pelleheaut, Joy, Vignoble Fontan et Uby*), il propose la dégustation et la vente de l’ensemble de leurs produits.

L’entrepôt dispose d’un PC Windows et le gérant souhaite gérer le stock des produits qu’il propose. La société NEGOSUD nous a contacté pour réaliser son projet.

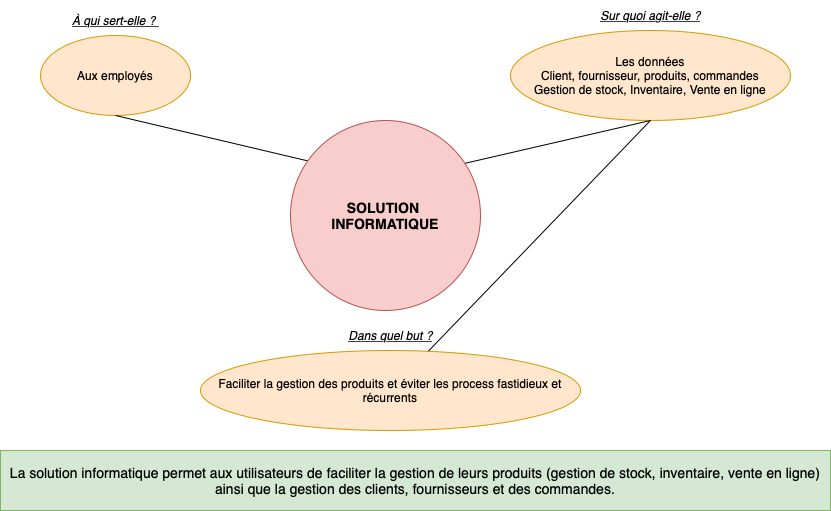
Secteur où le digital n’a pas énormément de place et du coup pour intégrer le digital dans ce domaine, l’entreprise souhaiterait informatiser sa gestion de stock.

Pb : comment améliorer la visibilité des employés sur les stocks de l’entreprise ?

Notre solution informatique permettra à NEGOSUD de gérer plus facilement ses produits et éviter les process fastidieux et récurrents (Gestion de stock, Inventaire, Vente en ligne).

Nous allons donc concevoir et développer une solution logicielle permettant à l’utilisateur de gérer les clients, les fournisseurs, les stocks, les inventaires et les commandes.

1. *Bête à corne*



1. *Les fonctionnalités*

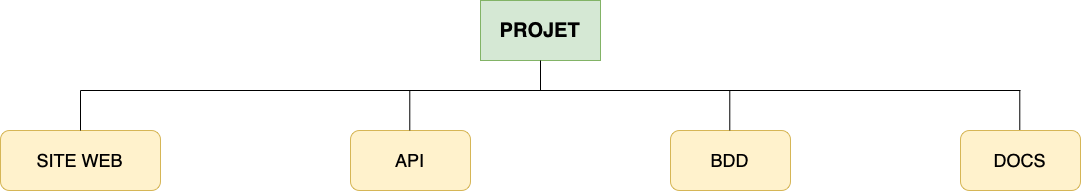
* CRUD sur les clients
* CRUD sur les fournisseurs
* CRUD sur les familles (rouge, rosé, blanc, pétillants, digestifs)
* CRUD sur les articles (avec références, quantité, date (millésime), prix et fournisseurs)
* Modification des stocks unitairement (ajouter et supprimer une quantité d’un article)
* Passer des commandes (référence-quantité-date) auprès d’un fournisseur
* Inventorier l’ensemble des articles
* Régularisation des stocks en masse (ajouter et supprimer une grande quantité d’articles - en carton ?)
* Commande client même si stock insuffisant
* Réapprovisionnement automatique : envoie automatique d’une commande fournisseur si stock insuffisant ou stock au seuil minimum
* BONUS : désactivation du réapprovisionnement automatique

1. **Conception (2j)**
2. *Architecture de la solution*

Étape 1 : PBS (Product Breakdown Structure)

Pour réaliser la solution logicielle répondant au cahier des charges, nous avons besoin de :

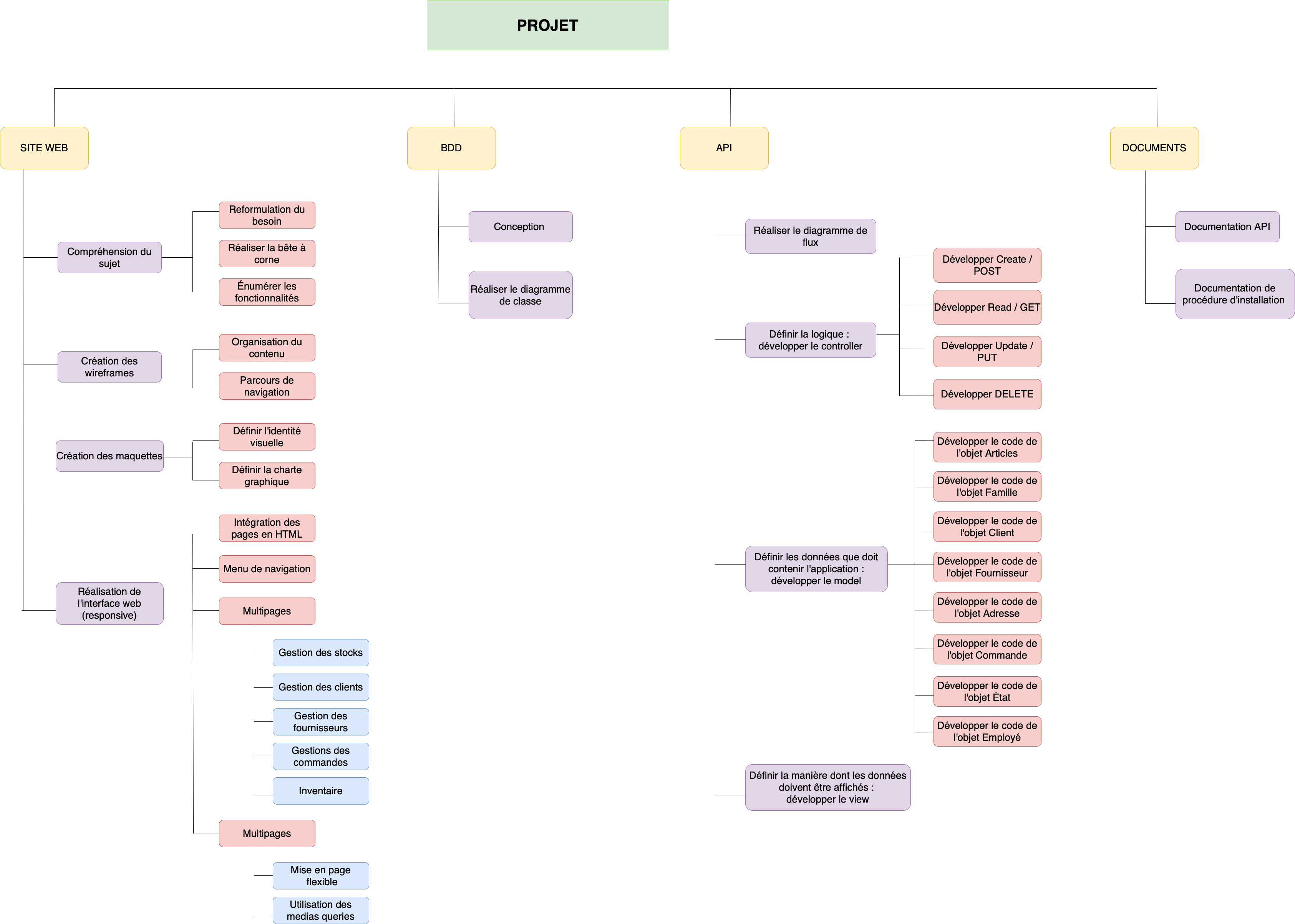
* Un site web
* Une API (application programming interface)
* Une Base De Données
* Une documentation



Étape 2 : WBS (Work Breakdown Structure)

De ces premiers niveaux des tâches à réaliser en découle d’autres.

* SITE WEB
  + Compréhension du projet
    - Reformulation du besoin
    - Réalisation de la bête à corne
    - Énumérations des fonctionnalités attendues
  + Création des wireframes
    - Organisation du contenu
    - Parcours de navigation
  + Création des maquettes
    - Définir l’identité visuelle
    - Définir la charte graphique
  + Réaliser l’interface web (responsive)
    - Intégration des pages en HTML
    - Menu de navigation
    - Multipages
      * Gestion des stocks
      * Inventaire
      * Gestion des clients
      * Gestion des fournisseurs
      * Gestion des commandes
    - Création d’une feuille de style
      * Mise en page flexible (unités relatives : %, em, rem)
      * Utilisation des medias queries
* BDD
  + Conceptualiser la base de données
  + Réaliser le diagramme de classe
* API REST
  + Réaliser le diagramme de flux
  + Définir la logique qui met à jour le modèle et/ou la vue en réponse aux entrées des utilisateurs de l'application.
    - Développer le Controller
      * Développer le CREATE/POST
      * Développer le READ / GET
      * Développer l’UPDATE / PUT
      * Développer le DELETE
  + Définir les données que l'application doit contenir
    - Développer le Model
      * Développer le code de l’objet Article
        + Attributs (nom, référence, volume, prix unitaire, prix au carton, quantité, famille)
      * Développer le code de l’objet Famille
        + Attributs (nom)
      * Développer le code de l’objet Client
        + Attributs (nom, prénom, téléphone, adresse, mail, mot de passe)
      * Développer le code de l’objet Fournisseur
        + Attributs (nom, téléphone, mail, adresse, numéro SIRET)
      * Développer le code de l’objet Adresse
        + Attributs (rue, ville, code postale)
      * Développer le code de l’objet Commande
        + Attributs (numéro de commande, date/heure, état de la commande)
      * Développer le code de l’objet État
        + Attributs (nom)
      * Développer le code de l’objet Employés
        + Attributs (nom, prénom, téléphone, mail, mot de passe)
  + Définir la manière dont les données de l'application doivent être affichées
    - Développer le View
* DOCS
  + Documentation API
  + Documentation de procédure d’installation pour une mise en production



1. *MCD-MLD-MPD*
2. *Diagramme UML : diagramme de classe*
3. *Création de wireframes*
4. Page de connexion
5. Page d’accueil : menu de navigation avec :   
   vue des stock  
   articles  
   clients  
   fournisseurs  
   commandes  
     
   onglet recherche + onglet “mon compte”
6. Sous menu de navigation :   
   dans articles : ajouter, retirer, inventaire  
   dans clients : ajouter, retirer, liste  
   dans fournisseurs : ajouter, retirer, liste  
   dans commande : commande client, commande fournisseur
7. Page Articles > Ajouter : entrer:  
   nom  
   appellation  
   type  
   millésime  
   volume  
   fournisseur  
   prix unitaire  
   prix au carton  
   quantité
8. Page Articles > Retirer : entrer :   
   sélectionner le vin  
   date de sortie  
   quantité
9. Page Articles > Inventaire :   
   liste des articles avec leurs caractéristiques  
   ajouter/enlever une quantité ou entrer directement la quantité  
   quand je clique sur un article de la liste ça me renvoie vers une page ou y a toutes les infos de l’articles répertorier et la possibilité de modifier ces données
10. Pages concernant client et fournisseur c’est un peu la même chose juste différent au niveau des caractéristiques   
    caractéristique client :   
    nom prénom téléphone mail mdp adresse date de naissance  
    caractéristique fournisseur :   
    nom mail téléphone SIRET adresse
11. Page vue des stocks :   
    liste des articles
12. Page de Commande > Commande clients :   
    liste des commandes   
    bouton supprimer la commande (envoie de mail au client pour dire commande annulé)  
    bouton modifier --> ouvre une page avec les caractéristiques de la commande qu’on peut modifier  
    bouton ajouter --> ouvre une page d’ajout de commande avec : sélectionner un client, sélectionner les articles

(on accèdera aux bouton modif et suppr une fois dans la liste de commande)

1. Page de Commande > Commande fournisseurs : pareil que pour commande client
2. *Maquettes site*
3. **Planification**
4. *Qui est à l’aise avec quoi ?*
5. *Création d’un projet sur Jira ou Gitlab*
6. *Création des tâches à effectuer détaillées*
7. *Assignation et datation des tâches*
8. **Développement**
9. Page d’authentification
10. Page de visualisation des commandes
11. Page des stocks – inventaire
12. Page clients
13. Page fournisseur
14. Passage de commande au fournisseur si stock insuffisant et commande client passé, si besoin d’un restocke
15. CRUD sur ces pages
16. **Rédaction**
17. Rédiger le rapport
18. Rédiger une documentation de l’API
19. Rédiger une procédure d’installation pour une mise en production
20. Relire le rapport
21. Réaliser le PPT
22. S’entraîner à l’oral

diagramme UML

Module = différentes fonctionnalités

Client lourd = exécutable

Client riche = navigateur

Client léger =

Architecture n tiers : ihm, API (REST), bdd

API reçoit l’appel http

Object Relation Mapping = fait le CRUD

Méthode create article

Diagramme attendu : diagramme de classe (attributs, objets métiers)

<https://www.youtube.com/watch?v=-5hLjYZ2xZY>

Tuto API REST C#