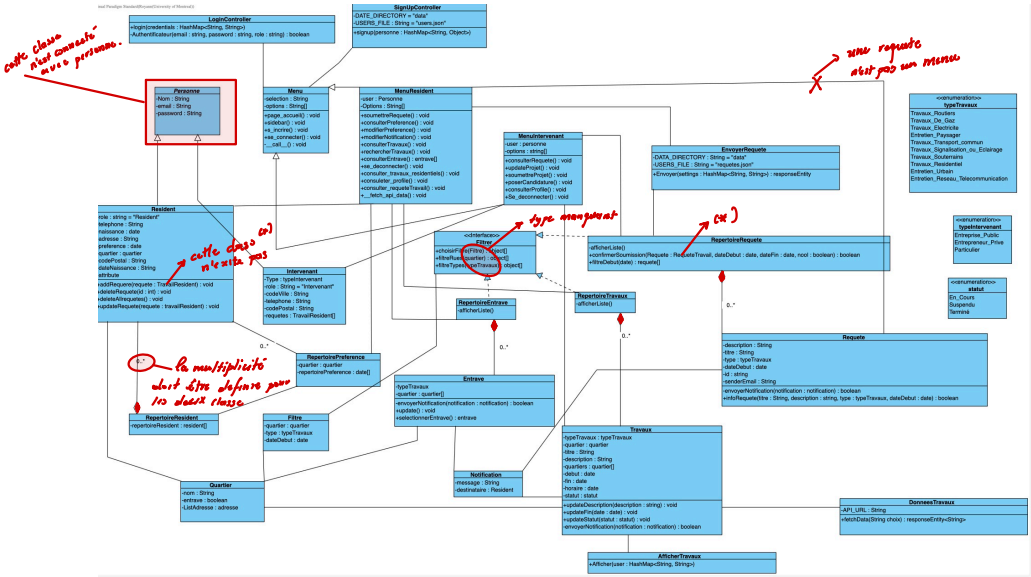
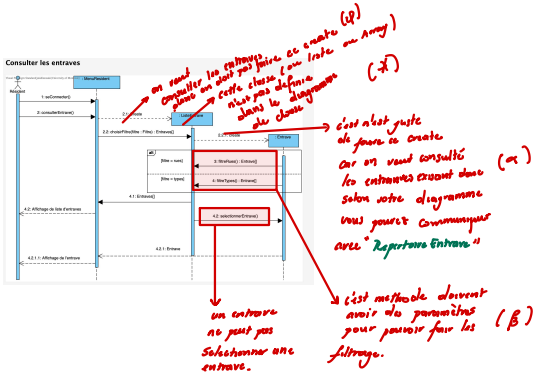
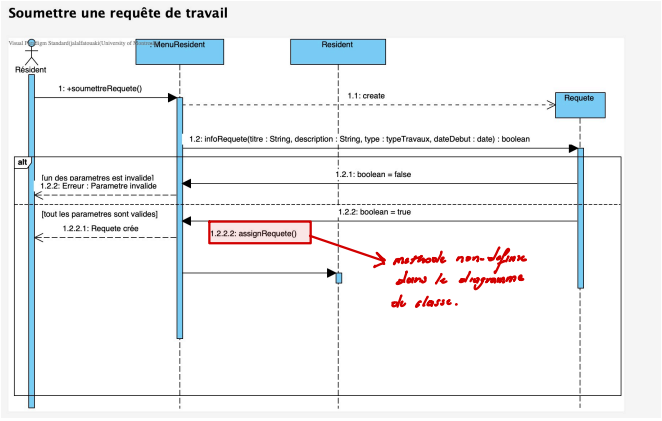
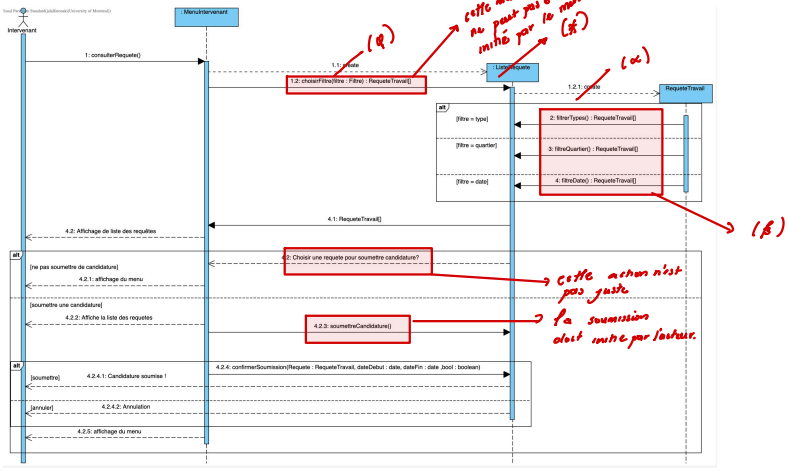

LoginController
+login(credentials : HashMap<String, String>)
-Authentificator(email : string, password : string, role : string) : boolean
SignUpController
-DATE_DIRECTORY = "data"
-USERS_FILE : String = "users.json"
+signup(personne : HashMap<String, Object>)
Personne
-Nom : String
-email : String
-password : String
Menu
-selection : String
-options : String[]
+page_accueil() : void
+sidebar() : void
+s_inscrire() : void
+se_connecter() : void
+__call_() : void
MenuResident
-user : Personne
-Options : String[]
+soumettreRequete() : void
+consulterPreference() : void
+modifierPreference() : void
+modifierNotification() : void
+consulterTravaux() : void
+rechercherTravaux() : void
+consulterEntraves() : entrave[]
+se_deconnecter() : void
+consulter_travaux_residentiels() : void
+consulater_profile() : void
+consulter_requetesTravail() : void
+__fetch_api_data() : void
MenuIntervenant
-user : personne
-options : string[]
+consulterRequetes() : void
+updateProjet() : void
+soumettreProjet() : void
+postulerCandidature() : void
+consulterProfile() : void
+Se_deconnecter() : void
EnvoyerRequete
-DATA_DIRECTORY : String = "data"
-USERS_FILE : String = "requetes.json"
+Envoyer(settings : HashMap<String, String>) : responseEntity
«enumeration»
typeTravaux
Travaux_Routiers
Travaux_De_Gaz
Travaux_Electricite
Entretien_Paysager
Travaux_Transport_commun
Travaux_Signalisation_ou_Eclairage
Travaux_Souterrains
Travaux_Residentiel
Entretien_Urbain
Entretien_Reseau_Telecommunication
Resident
role : string = "Resident"
telephone : String
naissance : date
adresse : String
preference : date
quartier : quartier
codePostal : String
dateNaissance : String
attribute
+addRequete(requete : TravailResident) : void
+deleteRequete(id : int) : void
+deleteAllrequetes() : void
+updateRequete(requete : travailResident) : void
Intervenant
-Type : typeintervenant
-role : String = "Intervenant"
-codeVille : String
-telephone : String
-codePostal : String
-requetes : TravailResident[]
«interface»
Filtre
+choisirFiltreFiltre() : object[]
+filtreRues(quartier : quartier[])
+filtreTypeTravaux(typeTravaux[]) : object[]
RepertoireRequete
-afficherListe()
+confirmerSoumission(Requete : RequeteTravail, dateDebut : date, dateFin : date, nocl : boolean) : boolean
+filtreDebut(date) : requete[]
«enumeration»
typeIntervenant
Entreprise_Public
Entrepreneur_Prive
Particulier
«enumeration»
statut
En_Cours
Suspendu
Terminé
RepertoireTravaux
-afficherListe()
RepertoireEntrave
-afficherListe()
RepertoirePreference
-quartier : quartier
-repertoirePreference : date[]
RepertoireResident
-repertoireResident : resident[]
Filtre
-quartier : quartier
-type : typeTravaux
-dateDebut : date
Entrave
-typeTravaux
-quartier : quartier[]
-envoyerNotification(notification : notification) : boolean
+update() : void
+selectionnerEntrave() : entrave
Requete
-description : String
-titre : String
-type : typeTravaux
-dateDebut : date
-id : string
-senderEmail : String
+envoyerNotification(notification : notification) : boolean
+infoRequete(titre : String, description : string, type : typeTravaux, dateDebut : date) : boolean
Quartier
-nom : String
-entrave : boolean
-ListeAdresse : adresse
Notification
-message : String
-destinataire : Resident
Travaux
-typeTravaux : typeTravaux
-quartier : quartier
-titre : String
-description : String
-quartiers : quartier[]
-debut : date
-fin : date
-horaire : date
-statut : statut
+updateDescription(description : string) : void
+updateFin(date : date) : void
+updateStatut(statut : statut) : void
+envoyerNotification(notification : notification) : boolean
DonneesTravaux
-API_URL : String
+fetchData(String choix) : responseEntity<String>
AfficherTravaux
+Afficher(user : HashMap<String, String>)
cette classe n'est connecté avec personne.
une requete était pas sur menu
cette classe n'existe pas
type manquant
(α)
la multiplicité doit être définie pour les deux classe
1..*
0..*


Consulter les entraves
MenuResident
Résident
1: seConnecter()
2: consulterEntrave()
2.1: choisir
ListEntrave
2.2: choisirFiltreFiltre : (Filre) : Entrave[]
2.2.1: choisir
Entrave
alt
[filre = rues]
3: filtreRues() : Entrave[]
[filre = types]
4: filtreTypes() : Entrave[]
4.1 : Entrave[]
4.2: Affichage de liste d'entraves
4.2: selectionnerEntrave()
4.2.1 : Entrave
4.2.1.1: Affichage de l'entrave
on veut consulter les entraves, donc on doit pas faire ce create (γ)
cette classe n'est pas définie dans le diagramme de classe (*)
c'est n'est juste de faire ce create car on veut consulté les entraves existant donc selon votre diagramme vous pouvez communiquer avec "Repertoire Entrave" (α)
un entrave ne peut pas sélectionner une entrave.
c'est methode doivent avoir des parametres pour pouvoir faire les filtrage. (β)


Soumettre une requête de travail
Résident
MenuResident
Resident
1: +soumettreRequete()
1.1: create
Requete
1.2: infoRequete(titre : String, description : String, type : typeTravaux, dateDebut : date) : boolean
alt
[un des paramètres est invalide]
1.2.2: Erreur : Parametre invalide
1.2.1: boolean = false
[tout les parametres sont valides]
1.2.2: boolean = true
1.2.2.1: Requete crée
1.2.2.2: assignRequete()
methode non-définie dans le diagramme de classe.


Intervenant
: MenuIntervenant
: LireRequete
RequeteTravail
1: consulterRequete()
1.1: create
1.2: choisirFiltre(filtre : Filtre) : RequeteTravail[]
1.2.1: create
alt
[filtre = type]
2: filtrerTypes() : RequeteTravail[]
[filtre = quartier]
3: filtreQuartier() : RequeteTravail[]
[filtre = date]
4: filtreDate() : RequeteTravail[]
4.1: RequeteTravail[]
4.2: Affichage de liste des requêtes
alt
[ne pas soumettre de candidature]
4.2.1: affichage du menu
[soumettre une candidature]
4.2.2: Affiche la liste des requetes
4.2.3: soumettreCandidature()
4.2.4: confirmerSoumission(Requete : RequeteTravail, dateDebut : date, dateFin : date ,bool : boolean)
[soumettre]
4.2.4.1: Candidature soumise !
[annuler]
4.2.4.2: Annulation
4.2.5: affichage du menu
(α)
(β)
(γ)
(δ)
cette methode ne peut pas être initié par le menu
cette action n'est pas juste
la soumission doit initié par l'acteur.