***Un simulateur OFDMA pour la transmission de données en liaison descendante dans un réseau LTE.***

1. **Les concepts clés :**
   * OFDMA (Orthogonal Frequency Division Multiplexing Access) : Technique de multiplexage pour les communications sans fil.
   * LTE (Long Term Evolution) : Standard de communication sans fil de 4e génération.
   * Allocation de porteuse : Processus d'attribution des sous-porteuses aux utilisateurs.
   * Trafic hétérogène asynchrone : Trafic avec des exigences différentes en termes de débit, de latence et de qualité de service.
2. **Exigences du simulateur :**
   * Le simulateur doit simuler l'envoi de données en liaison descendante via OFDMA depuis une seule antenne vers plusieurs dispositifs utilisateurs.
   * Il doit gérer l'arrivée de nouvelles communications avec différentes tailles de données (Mo) et différents schémas de codage.
   * Il doit représenter l'état de l'allocation des ressources radio sous forme de matrice, avec le nombre de sous-porteuses comme entrée.
   * Il doit afficher des informations sur l'arrivée des nouvelles communications, en associant une couleur différente à chacune d'elles.
3. **Développer le simulateur :**
   * Créez une interface graphique utilisateur (GUI) pour faciliter l'interaction avec le simulateur.
   * Implémentez un générateur de trafic hétérogène asynchrone pour simuler l'arrivée de nouvelles communications.
   * Implémentez un mécanisme d'allocation dynamique des ressources radio en fonction des besoins des communications en cours.
   * Visualisez l'état de l'allocation des ressources radio sous forme de matrice, en faisant glisser les colonnes pour montrer la progression dans le temps.
   * Affichez les informations sur l'arrivée des nouvelles communications, en associant une couleur différente à chacune d'elles.

Côté serveur (PHP) :

* + Gérer la logique de simulation, y compris la génération de trafic, l'allocation des ressources radio et le traitement des données.
  + Implémenter les mécanismes d'arrivée des nouvelles communications et d'allocation dynamique des ressources radio.
  + Créer des API pour communiquer avec le côté client et envoyer les données de simulation.

Côté client (JavaScript) :

* + Utiliser JavaScript pour créer une interface graphique utilisateur (GUI) qui interagit avec l'utilisateur et affiche les résultats de la simulation.
  + Utiliser des bibliothèques JavaScript telles que React, Angular ou Vue.js pour structurer l'interface et gérer les interactions utilisateur.
  + Communiquez avec les API côté serveur pour récupérer les données de simulation et les afficher dans l'interface graphique.