1-Définir me modèle d’architectures en couche et en tiers

2-Préconiser des motifs de conception pour les couches

3-Préconiser les librairies, composants

**Stock information :**

Le logiciel implémenté va servir de parseur des données du marché. Il devra pouvoir rendre persistante ces données et fournir des services de lecture et écriture sur ces données. Il doit aussi passer des ordres sur le marché.

Le programme étant exécuté sur le système d’exploitation z/OS et nécessitant de très bonne performance en terme de rapidité sera implémenté avec le langage C++.

**Couches :**

|  |  |
| --- | --- |
| **Présentation** | N/A : aucune interaction avec l’utilisateur |
| **Contrôleur** | Doit contrôler l’appel de services, et doit contrôlé que le carnet d’ordre présent dans la base de données est à jours. |
| **Services** | Web services REST   * Récupérer un carnet d’ordre * Passer un ordre |
| **Domaine** | **Objets métiers :**   * Carnet d’ordres (liste d’ordres)   Utilisation de la libraire STL   * Ordre   **Fourniture des moyens d’accès à l’information :**  Oracle Call Interface (OCI) est une API fournie par Oracle permettant aux développeurs de créer des applications en utilisant des appels C++ bas niveau afin d'accéder à des bases de données Oracle. |
| **Persistance** | Librairie OCILIB encapsule OCI afin de fournir une interface beaucoup plus simple à coder et facilite la maintenance. |

**Tiers :**

Modèle 2-tiers.

1 serveur Logique : IBM CICS z/OS : application C++

1 serveur Données : Oracle

**Transaction sytem :**

Le logiciel implémenté va servir de contrôleur afin d’autoriser les ordres avant que ceux-ci ne soit passé au Stock Information.

**Couches :**

|  |  |
| --- | --- |
| **Présentation** | N/A : aucune interaction avec l’utilisateur |
| **Contrôleur** | Doit contrôler l’appel de services, Contrôler la validité des données et appelant les services de bloc Account Information |
| **Services** | Web services REST   * Passer un ordre(ordre, client) |
| **Domaine** | **Objets métiers :**   * Carnet d’ordres (liste d’ordres)   Utilisation de la libraire STL   * Ordre * Client   **Fourniture des moyens d’accès à l’information :**  Les accès à l’information son géré par le bloc Account Information |
| **Persistance** | La persistance des données est gérée par le bloc Account Information |

**Tiers :**

Modèle 1-tiers.

1 serveur Logique : IBM CICS z/OS : application C++

**Account information :**

Le logiciel implémenté va servir de persisté toutes les données relatives aux clients.

Le programme étant exécuté sur le système d’exploitation z/OS et nécessitant de très bonne performance en terme de rapidité sera implémenté avec le langage C++.

**Couches :**

|  |  |
| --- | --- |
| **Présentation** | N/A : aucune interaction avec l’utilisateur |
| **Contrôleur** | Doit contrôler l’appel de services, et doit permettre de créer, rechercher, modifier et supprimer des composants d’objets métiers dans le respect des propriétés transactionnelles. |
| **Services** | Web services REST   * Créer un client * Supprimer un client * Modifier un client * Ajouter un ordre * Supprimer un ordre * Modifier un ordre |
| **Domaine** | **Objets métiers :**   * Carnet d’ordres (liste d’ordres)   Utilisation de la libraire STL   * Ordre * Client   **Fourniture des moyens d’accès à l’information :**  Oracle Call Interface (OCI) est une API fournie par Oracle permettant aux développeurs de créer des applications en utilisant des appels C++ bas niveau afin d'accéder à des bases de données Oracle. |
| **Persistance** | Librairie OCILIB encapsule OCI afin de fournir une interface beaucoup plus simple à coder et facilite la maintenance. |

**Tiers :**

Modèle 1-tiers.

1 serveur Données : Oracle SPARC T4-1