

Principe d'une application

Sommaire

- 1. Les différents types d'applications**
- 2. Les Composants d'une application**
- 3. Architecture et Modélisation**
- 4. Déploiement et Maintenance**

Les différents types d'applications

Les différents types d'applications



Les différents types d'applications

1. Applications Web

- **Définition et Caractéristiques :** Fonctionnent via un navigateur web et ne nécessitent pas de téléchargement ou d'installation sur le dispositif de l'utilisateur.
Peuvent être responsives pour s'adapter à différents dispositifs (ordinateur, tablette, mobile).
- **Utilisation dans le Low Code :**
 - Rapidité de développement grâce à des modèles prêts à l'emploi.
 - Mise à jour facile sans action de l'utilisateur.

2. Applications Mobiles

- **Définition et Caractéristiques :**
 - Conçues spécifiquement pour les smartphones et les tablettes.
 - Peuvent être natives, hybrides ou web mobiles.
- **Utilisation dans le Low Code :**
 - Les plateformes de low code permettent souvent de créer des applications hybrides qui fonctionnent sur plusieurs systèmes d'exploitation.

Les différents types d'applications

3. Applications de Bureau (Desktop)

- **Définition et Caractéristiques :**

- Installées et exécutées sur un ordinateur personnel ou de bureau.
- Conçues pour tirer pleinement parti des ressources système de l'ordinateur.

- **Utilisation dans le Low Code :**

- Moins courantes en low code, mais certaines plateformes permettent de créer des applications de bureau multiplateformes.

Les différents types d'applications

4. Applications Entreprise (Enterprise Applications)

- **Définition et Caractéristiques :**

- Conçues pour répondre aux besoins des organisations et des entreprises.
- Comprend souvent des applications pour la gestion de la relation client (CRM), planification des ressources d'entreprise (ERP), automatisation des services, etc.

- **Utilisation dans le Low Code :**

- Permet une intégration avec des systèmes existants et une évolutivité pour répondre à des besoins complexes.
- Favorise l'alignement rapide avec les processus d'affaires grâce à la possibilité de personnalisations.

Les différents types d'applications

5. Applications de l'Internet des Objets (IoT)

- **Définition et Caractéristiques :**

- Permettent de contrôler et de collecter des données depuis des dispositifs connectés à Internet.
- Sont souvent employées dans l'automatisation des habitations, la surveillance, la maintenance prédictive, etc.

- **Utilisation dans le Low Code :**

- Les plateformes low code peuvent faciliter l'intégration avec l'IoT en fournissant des connecteurs et des API prédéfinis.

Les différents types d'applications

6. Applications de Commerce Électronique (E-commerce)

- **Définition et Caractéristiques :**

- Spécialement conçues pour acheter et vendre des biens ou services en ligne.
- Intègrent des systèmes de gestion de produits, des paniers d'achat, et des modules de paiement.

- **Utilisation dans le Low Code :**

- Les solutions low code peuvent accélérer le développement de boutiques en ligne avec des fonctionnalités standardisées et personnalisables.

Les différents types d'applications

7. Applications sur Mesure (Custom Applications)

- **Définition et Caractéristiques :**

- Développées pour répondre à des besoins spécifiques qui ne sont pas couverts par des applications standard.
- Souvent personnalisées pour un processus d'affaires unique ou un groupe d'utilisateurs spécifique.

- **Utilisation dans le Low Code :**

- Le low code permet de créer rapidement des solutions sur mesure avec un effort de développement réduit et une approche modulaire.

TP Les différents types d'applications

Digitalisation de "Globetrotter Adventures"

Contexte : Vous travaillez pour "Globetrotter Adventures", une entreprise spécialisée dans l'organisation de voyages d'aventure personnalisés. Avec l'expansion de l'entreprise, "Globetrotter Adventures" souhaite digitaliser ses services pour améliorer l'expérience client et optimiser les processus internes.

Projet :

Votre équipe a été chargée de proposer une solution digitale qui couvre les aspects suivants de l'entreprise :

1. **Interface Client :** Une plateforme permettant aux clients de parcourir les offres de voyage, de personnaliser et de réserver leurs aventures.
 2. **Outil de Gestion Interne :** Un système pour gérer les réservations, les itinéraires de voyage, et le suivi des clients.
 3. **Système d'Analyse des Données :** Un outil pour analyser les tendances de réservation, les retours clients et les performances des voyages.
- Chaque groupe de stagiaires doit identifier et justifier le type d'application le plus adapté pour chaque aspect du projet en utilisant les principes du low code/no code. Ils doivent prendre en compte la facilité de mise à jour, la maintenance, l'intégration avec d'autres systèmes, et l'expérience utilisateur.

Les Composants d'une application

Les Composants d'une application - Composants Front-End

1. Interface Utilisateur (UI)

- L'Interface Utilisateur est la partie de l'application avec laquelle vous interagissez directement. Elle comprend tout ce que vous pouvez voir et manipuler, comme les boutons, les textes, les images, et les champs de formulaire. Dans le développement low code, vous n'avez pas besoin d'écrire manuellement le code pour créer ces éléments. Au lieu de cela, vous utilisez des outils visuels pour assembler des pages en sélectionnant, en plaçant et en configurant des éléments à partir d'une bibliothèque d'interface.

2. Fonctionnalités Front-End

- Ces fonctionnalités sont les réactions de l'application à vos actions, comme cliquer sur un bouton ou entrer des informations dans un formulaire. Les plateformes low code vous permettent de définir ces fonctionnalités à travers des configurations simples ou des logiques visuelles. Par exemple, vous pouvez indiquer ce qui se passe après un clic de bouton en choisissant parmi une liste d'actions prédéfinies, sans écrire de code.

Les Composants d'une application - Composants Front-End

3. Frameworks et Bibliothèques

- Les frameworks et bibliothèques comme React ou Angular sont des collections de code pré-écrit qui facilitent la création d'applications web en fournissant des structures et des composants réutilisables. Dans le low code, ces frameworks peuvent être intégrés en arrière-plan, vous n'avez donc pas besoin de les comprendre en profondeur. Au lieu de cela, vous travaillez avec une interface qui encapsule ces technologies, vous permettant de construire des applications sans connaître les détails du code qui les fait fonctionner.

Les Composants d'une application - Composants Back-End

1. Serveur, Application, et Base de Données (SABD)

- Le back-end d'une application est la partie que vous ne voyez pas en tant qu'utilisateur mais qui est cruciale pour le fonctionnement de l'application. Il s'agit des serveurs qui exécutent l'application, de la logique de l'application qui traite les données, et des bases de données qui stockent ces données. Dans un environnement low code, vous gérez cela à travers des interfaces graphiques où vous pouvez définir des modèles de données, des règles métier et des processus sans écrire de code complexe.

2. Logique Métier

- La logique métier représente les règles qui déterminent comment les données sont créées, stockées et modifiées. Imaginez une boutique en ligne; la logique métier détermine ce qu'il se passe quand un article est ajouté au panier. Dans un environnement low code, vous implémentez la logique métier en utilisant des outils visuels pour créer des workflows et des processus.

Les Composants d'une application - Composants Back-End

3. Intégrations et API

- Les intégrations et les API permettent à votre application de communiquer avec d'autres systèmes et services. Une API est comme un menu dans un restaurant; elle vous dit ce que vous pouvez demander et la manière de le demander. Dans le low code, les intégrations sont souvent simplifiées par des connecteurs prêts à l'emploi qui vous permettent de relier votre application à des systèmes externes sans avoir besoin de comprendre en détail comment ces connexions fonctionnent au niveau du code.

Les Composants d'une application - Composants de Gestion de Données

1. Base de Données

- La base de données est l'endroit où toutes les informations de votre application sont stockées. Cela peut inclure des informations sur les utilisateurs, des produits, des commandes, etc. Les plateformes low code offrent généralement une manière simplifiée de créer et de gérer des bases de données à travers des interfaces graphiques, où vous pouvez créer des structures de données et définir comment elles se rapportent les unes aux autres sans avoir à écrire de requêtes SQL complexes.

2. Gestion des États

- La gestion de l'état est la manière dont votre application garde la trace de ce que l'utilisateur fait. Par exemple, si vous vous connectez à une application, la gestion de l'état se souvient que vous êtes connecté. Dans les plateformes low code, vous pouvez gérer l'état à travers des sessions utilisateur ou des variables globales qui peuvent être configurées visuellement sans toucher au code.

Les Composants d'une application - Composants de Sécurité

1. Authentification et Autorisation

- Ces deux composants sont essentiels pour la sécurité de votre application. L'authentification vérifie qui vous êtes (par exemple, en vérifiant votre nom d'utilisateur et votre mot de passe), tandis que l'autorisation détermine ce que vous êtes autorisé à faire une fois connecté. Dans le développement low code, des modules intégrés vous permettent de mettre en place ces mesures de sécurité facilement.

2. Cryptographie et Sécurité des Données

- La cryptographie est utilisée pour protéger les informations sensibles, comme les mots de passe ou les données personnelles, en les transformant en codes qui ne peuvent pas être lus sans une clé de décryptage. En low code, des fonctions de cryptographie peuvent être appliquées automatiquement ou via des paramètres simples, sans que vous ayez à manipuler directement les algorithmes de cryptage.

Les Composants d'une application - Composants de Performance et Optimisation

1. Caching et Stockage

- Le caching est une technique qui stocke temporairement des copies de données pour accélérer l'accès aux informations fréquemment utilisées. Le stockage fait référence à la manière dont les données sont conservées sur le long terme. Dans un contexte low code, des solutions de caching et de stockage peuvent être configurées via des interfaces graphiques.

2. Optimisation du Code et des Ressources

- L'optimisation implique d'ajuster votre application pour qu'elle fonctionne aussi efficacement que possible. Cela peut concerner le code lui-même ou la manière dont les ressources, comme les images et les scripts, sont chargées. Les plateformes low code gèrent souvent cette optimisation pour vous, s'assurant que l'application est rapide et réactive sans intervention manuelle.

Architecture et Modélisation

Architecture et Modélisation

1. Modèles de Conception

- Ces modèles sont des plans ou des templates pour organiser le code de votre application d'une manière qui est à la fois logique et efficace. Les modèles courants incluent MVC (Modèle-Vue-Contrôleur) et MVVM (Modèle-Vue-VueModèle). Dans le low code, l'architecture est souvent prédéfinie et simplifiée pour que vous puissiez vous concentrer sur la logique métier plutôt que sur la structure de l'application.

2. Schémas de Base de Données

- Le schéma de base de données est le plan détaillé de la manière dont les données sont organisées dans la base de données. Cela inclut les tables, les champs, les types de données, et les relations entre les tables. Avec le low code, vous définissez souvent ces schémas à l'aide d'outils graphiques qui génèrent le schéma réel en arrière-plan.

Déploiement et Maintenance

Déploiement et Maintenance

1. Infrastructure et Plateformes de Déploiement

- C'est ici que votre application vit - sur des serveurs et des plateformes qui la rendent accessible aux utilisateurs. Le déploiement dans le cloud, par exemple, signifie que votre application est hébergée sur des serveurs gérés par des fournisseurs comme Amazon AWS ou Microsoft Azure. Les outils low code peuvent simplifier le déploiement en automatisant la plupart des étapes nécessaires.

2. Suivi et Logs

- Après le déploiement, il est crucial de surveiller comment votre application fonctionne dans le monde réel. Le suivi et les logs sont comme un journal de bord qui enregistre les activités et les erreurs. Les plateformes low code fournissent des tableaux de bord et des outils de rapport pour vous aider à comprendre la performance de votre application sans avoir à plonger dans des fichiers de log complexes.