

フレームワーク

- 1.RAILS...基本理念は「同じことを繰り返さない」・「設定より規約」・「定義などの作業は一回だけで済ませる」である。＊開発者の決定すべきことを減少させ、単純にするが柔軟性を失わせない。Railsでは、全く何もないところからアプリケーションを作成していくのではなく、アプリケーションの雛形を自動でまず作成し、そのあとで、必要な箇所を修正するという手順を踏む。インストール方法は、まず「Rudy」をインストールした後、コマンドプロンプトを使用する。
- 2.JAVAE...プログラミング言語の Javaによって、企業システムを開発するための標準仕様。企業システムの機能として実装されるコンポーネントを作るための約束事、コンポーネントからデータベースなどのインフラサービスにアクセスするための APIなどが、標準化されているため、一度開発したアプリケーションを JAVAEに対応した様々な実行環境の上で、動かすことが可能となる。移植性が高く、サーバマシンやアプリケーションサーバを変えても使い続けられる。
- 3.Spring ...DIと呼ばれる仕組みの導入により、依存性のあるプログラムは外部から取り入れるようになった Frameworkているので、プログラムの修正や変更が必要になっても、最低限で済む。また、プログラムが一連の処理を終えるかどうか、結合テストすることができ、フレームワークの基本的な機能のほとんどが「インターフェース」なので、拡張できる。さらに、主に必要な処理と、そうでない処理を分けることで、コードがわかりやすくなり、バグや不具合があっても、原因が突き止めやすいなど、保守性が高く、共通のプログラムをまとめやすいという点でも、再利用性が高い。