

# IBM Worklight V5

## 기술 문서



### 목차

- 1 IBM Worklight 개요
- 2 IBM Worklight 구성요소
- 2 개발도구
- 8 런타임 서버 환경
- 10 보안 및 인증 메커니즘
- 11 시스템 요구 사항

### IBM Worklight 개요

IBM Worklight는 다양한 스마트폰과 태블릿을 위한 HTML5, 하이브리드 및 네이티브 애플리케이션(이하 앱)을 통해 기업이 비즈니스를 확장시킬 수 있도록 지원합니다. 또한 백엔드 통합 및 인증을 통해 개발에서부터 사후 배포 관리에 이르는 전체 애플리케이션 라이프사이클을 지원할 수 있습니다.

**개발:** IBM Worklight Studio 및 IBM Worklight SDK는 iOS, 안드로이드, 블랙베리 및 윈도우폰을 포함하여 다양한 모바일 플랫폼 전반에 걸쳐 웹, 하이브리드 및 네이티브 앱의 개발을 간단하게 만들어줍니다. IBM Worklight 최적화 프레임워크는 다양한 환경에서 요구하는 스타일에 맞는 풍부한 사용자 경험을 제공할 수 있습니다. 이러한 과정에서 IBM Worklight는 개발과 시장 진출에 필요한 시간 및 지속적인 관리 노력을 효과적으로 감소시키면서 하나의 환경에서 다른 환경에 이르는 코드 베이스의 공유를 극대화할 수 있습니다.

**통합:** IBM Worklight의 서버 아키텍처와 어댑터 기술로 백엔드 엔터프라이즈 시스템 및 클라우드 기반 서비스와 앱의 통합을 간단하게 수행할 수 있습니다. IBM Worklight Server는 기업의 IT 인프라에 맞도록 유연하게 설계되었기 때문에 기존 자원의 활용을 극대화할 수 있습니다. 독립형 백엔드 통합 계층을 커스터마이징할 수 있으며, 다양한 애플리케이션 간에 공유할 수 있습니다. 또한 IBM Worklight Adapters는 2가지 유형의 데이터 전송 메커니즘(디바이스 요청 및 푸시 알림)을 지원합니다.

**런타임:** IBM Worklight Studio는 다양한 공개 앱스토어 및 사설 배포 저장소로 앱을 업로드하기 위해 필요한 애플리케이션 파일 작성을 지원합니다. 애플리케이션을 실행하면 IBM Worklight Server를 통해 모든 엔터프라이즈 백엔드 시스템 및 클라우드 기반 서비스와 커뮤니케이션을 할 수 있습니다. 이는 모바일 전송을 위해 데이터를 최적화시키며, 민감한 사용자 데이터를 보호하기 위해 다양한 보안 기능이 제공됩니다.

**관리:** 일단 배포되면, 관리자들은 다양한 앱과 디바이스의 백엔드 시스템 액세스를 제어할 수 있습니다. 또한 미리 정의된 룰을 바탕으로 직접 업데이트를 하거나 비활성화를 시킬 수 있으며, IBM Worklight Console이라는 하나의 중앙 웹 인터페이스에서 모든 푸시 서비스와 이벤트 소스를 제어할 수 있습니다. 또한 관리자들은 기본으로 제공되는 맞춤형 보고서를 사용하여, 설치된 앱 기반과 사용자들에 관한 사용 정보를 액세스할 수 있습니다.



## IBM Worklight 구성요소

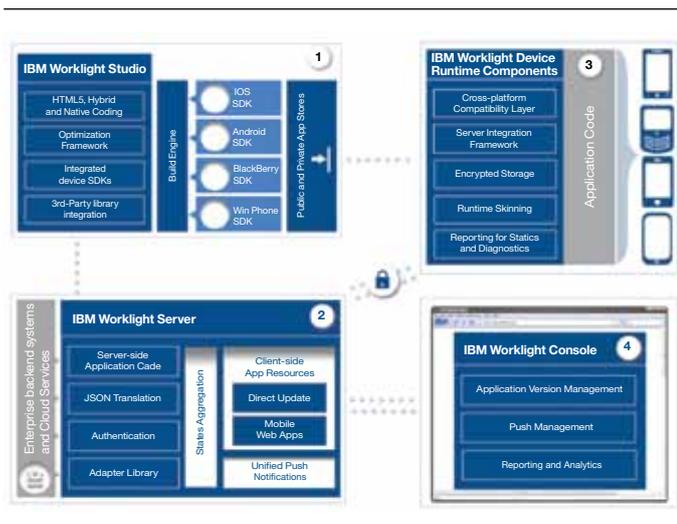
IBM Worklight 아키텍처는 다음과 같은 4가지 핵심 요소들로 구성됩니다.

- IBM Worklight Studio: 플랫폼을 위한 통합개발환경(IDE)
- IBM Worklight Server: 앱, 백엔드 시스템 및 클라우드 서비스 간의 게이트웨이
- IBM Worklight Device Runtime Component: 클라이언트 기능을 통해 서버 보안
- IBM Worklight Console: 웹 기반 관리 인터페이스

## 개발도구

### IBM Worklight Studio

IBM Worklight Studio는 Eclipse 기반의 통합개발환경이며, 개발자가 직원용 및 고객용 애플리케이션을 개발하기 위해 요구되는 모든 코딩 및 통합 작업을 수행할 수 있도록 지원합니다. IBM Worklight Studio는 IBM Worklight 플러그인에 의해 제공되는 다양한 엔터프라이즈 수준의 기능들과 익숙한 Eclipse 툴을 강화시키며, 이를 통해 애플리케이션의 개발을 간소화시키고 기존 기업 내부 시스템과 연결합니다.



IBM Worklight Studio가 지원하는 핵심 기능들 중 일부는 다음과 같습니다.

### 크로스 플랫폼 지원

IBM Worklight Studio는 iOS 안드로이드, 블랙베리, 윈도우폰, 스마트폰 및 태블릿에 사용되는 풍부한 하이브리드 애플리케이션과 네이티브 모바일 애플리케이션의 개발을 지원합니다. HTML4, CSS 2.1 및 JavaScript 1.5 버전 이상을 실행할 수 있는 웹 브라우저들을 사용하는 최신 디바이스 지원을 위해 모바일 HTML 웹 앱을 개발할 수 있습니다.

IBM Worklight는 자체 최적화 프레임워크(Optimization Framework)를 사용하기 때문에 최소한의 기능만을 제공하는 시장의 다른 기술들과 차별화됩니다. 이는 개발자들이 사용자 경험이나 애플리케이션 기능을 타협하지 않고, 다양한 환경에 걸쳐 애플리케이션 코드의 대부분을 공유할 수 있도록 지원함으로써 가능해집니다.

최적화 프레임워크는 IBM Worklight Studio의 독특한 파일 구조를 사용하여 개발자들이 코드 베이스의 재사용을 극대화하면서 애플리케이션을 다양한 모바일 환경에 맞게 조정할 수 있도록 지원합니다. 개발자들은 일반적으로 공유된 코드를 재작성하거나 증가시킬 수 있는 지정된 폴더에 특정 환경 코드를 분리시키면서, 다양한 환경들 사이에서 일반적인 애플리케이션 코드를 공유할 수 있습니다. 결과적으로 사용자 인터페이스가 고유의 방식으로 작용하고, 최종 사용자의 기대치를 충족시키는 독특한 기능 및 디바이스의 디자인 가이드라인을 고수하며, 동시에 애플리케이션 로직은 다양한 환경 사이에서 일관성을 유지할 수 있습니다.

애플리케이션 개발자들은 최신 디바이스들이 제공하는 API에 직접 액세스할 수 있습니다. 또한 공개적으로 사용할 수 있거나 커스터마이징된 타사의 라이브러리, 프레임워크 및 툴을 손쉽게 통합할 수 있으며, 이를 통해 각 조직의 필요에 따라 구축된 고급 모바일 애플리케이션을 만들 수 있습니다.

개발자들은 크로스 컴파일러(Cross-compiler) 또는 인터프리터 등의 런타임 계층이나 중간 빌드 타임에 의존하지 않기 때문에, IBM Worklight를 사용하여 새로운 모바일 OS 버전 또는 타사의 라이브러리 출시에 맞춰 네이티브 API에 액세스할 수 있습니다. 또한 이 앱의 웹 코드는 모바일 브라우저에 의해 직접 실행되기 때문에, 개발자들이 HTML DOM(Document Object Model)에 직접 액세스할 수 있으며, 모든 JavaScript API 또는 타사의 JavaScript 툴킷과 프레임워크를 자유롭게 사용할 수 있습니다.

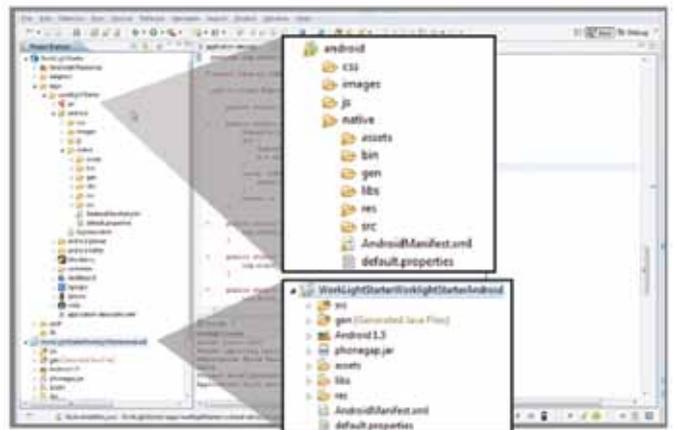
### 하이브리드 코딩

오늘날 모바일 디바이스는 끊임없이 진화하고 있고 운영 시스템은 다양하게 변화하고 있어, 애플리케이션 개발은 많은 비용이 요구됩니다. 이러한 과정은 급속히 성장하고 있는 크로스 플랫폼 모바일 개발 솔루션이라는 시장을 만들어 냈습니다.

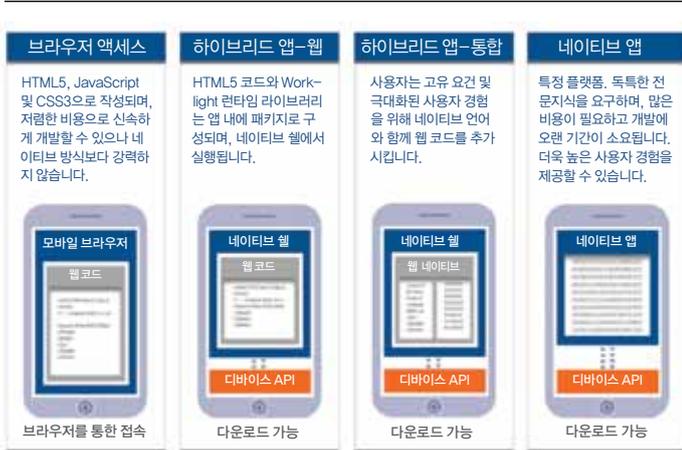
그러나 크로스 플랫폼 기능을 달성하기 위해서 시장의 많은 솔루션들은 양식 기반의 통합개발환경, WYSIWYG(What You See Is What You Get) 툴 및 간단하게 미리 패키징된 앱과 같은 독점적 툴에 제한적으로 의지하고 있습니다. 이로 인해 사용자 경험과 멀티플랫폼 적용 범위의 파할 수 없는 상충 관계가 생성되게 됩니다. 개발자들은 IBM Worklight를

사용하여 순수한 네이티브 코드(Objective-C, Java 또는 C#) 또는 표준 웹 기술(HTML5, CSS3 및 JavaScript)을 사용할 수 있으며, 동일한 앱에 이 2가지 조합을 사용하는 방법도 선택할 수 있습니다. 이를 통해 개발자들은 개발 효율성과 앱 기능성 사이에서 원하는 균형을 찾을 수 있습니다.

개발자들은 HTML 기반의 페이지로부터 네이티브 코드를 호출할 수 있으며, 동일한 애플리케이션에서 HTML과 네이티브 기반의 페이지들을 조합할 수 있습니다. 또한 동일한 페이지에 HTML과 네이티브 컴포넌트를 같이 표시할 수 있습니다.



어떠한 단일 개발 접근 방식도 더욱 큰 규모에 대한 완벽한 솔루션을 제공할 수 없습니다. 그러나 IBM Worklight가 제공하는 하이브리드 코딩을 위한 독특한 지원을 통해, 조직은 당면한 프로젝트의 특정 요건을 기반으로 하는 다양한 앱 유형들을 개발, 실행 및 관리하기 위해 동일한 모바일 플랫폼을 사용할 수 있습니다.



### 런타임 스킨

런타임 스킨을 활용하여 IBM Worklight Studio 내에서 더욱 최적화된 앱을 사용할 수 있습니다. 이러한 스킨들은 앱의 실행 가능한 기능과 함께 패키지로 구성되며, 모바일 앱에 적용되고, 동일한 OS 제품군에서 다양한 디바이스에 맞게 자동으로 조정할 수 있습니다. 런타임 스킨을 통해 혜택을 얻을 수 있는 일반적인 시나리오는 다음과 같습니다.



### HTML5 지원

IBM Worklight이 지원하는 HTML5를 통해 표준 기반의 접근 방식을 강화하고, 개발자들이 독점적 인터프리터나 코드 변환기의 제한 없이 앱에 직접 HTML5 코드를 작성할 수 있으며, 다음과 같은 이점을 얻을 수 있습니다.

- 더욱 명확하고 읽기 쉬우며 일관성이 있는 HTML 코드를 작성할 수 있습니다.
- 기존에는 네이티브 코드 방식으로만 사용이 가능했던 풍부한 미디어(오디오 및 비디오)에 액세스할 수 있습니다.
- 브라우저에 기본적으로 구현된 데이터 수집기, 슬라이더 기능 및 텍스트 생략 기호를 제공하는 편집 상자 등 고급 UI 컴포넌트를 사용할 수 있습니다.
- 앱의 크기를 감소시키고 앱의 응답 속도를 향상시키기 위해 CSS3 스타일 및 CSS3 기반의 애니메이션을 사용할 수 있습니다.
- 다양한 앱스토어 및 이로 인한 시간 소비, 제한 사항들을 뛰어 넘는 앱 유통 채널
- 위치 정보 서비스 지원
- 오프라인 저장 기능

IBM Worklight는 애플리케이션의 가용성을 향상시키기 위해 디바이스 암호화 및 오프라인 사용자 인증 등과 같은 엔터프라이즈 수준의 유틸리티와 함께 이러한 기능들을 증가시킵니다.

**타사의 JavaScript 툴킷과 UI 프레임워크 지원**

IBM Worklight 플랫폼은 HTML5 지원뿐만 아니라 jQuery Mobile, Sencha Touch 및 dojox.mobile 등과 같은 UI 프레임워크와의 유연한 통합을 제공합니다. 개발자들은 JavaScript UI 프레임워크를 자신이 선택할 수 있으며, IBM Worklight Studio를 통해 자신이 선택한 프레임워크를 이용하여 애플리케이션을 개발할 수 있습니다.

**네이티브 디바이스 SDK 통합**

IBM Worklight Studio는 IBM Worklight가 지원하는 모바일 디바이스의 SDK와 완벽하게 통합됩니다. 이를 통해 개발자들은 개발 환경을 벗어나지 않고 네이티브 코드 기능과 최고의 개발 툴 및 모바일 디바이스에 기본으로 적용되는 테스트와 디버깅 메커니즘을 최대한 활용할 수 있습니다.

**표준화된 데이터 검색**

IBM Worklight Studio는 개발자들이 백엔드 시스템에서부터 JSON(Javascript Object Notation) 포맷에 이르기까지 검색된 계층적 데이터를 변환하기 위해 XSL 변환과 JavaScript 코드를 사용할 수 있게 해주며, 이를 통해 앱의 소비에 대응하기 위한 준비를 할 수 있습니다. 개발자들은 IBM Worklight Studio에서 직접 백엔드 서비스를 호출할 수 있으며, XML에서 초기 결과값을 획득하거나, JSON 포맷에서 처리된 결과값(XSL 변환 및 JavaScript를 사용하여 JSON으로 변환한 후에)을 얻을 수 있습니다.



개발자들은 다양한 백엔드 애플리케이션으로부터 데이터를 수집하기 위해 JavaScript에서 서버 매시업을 수행할 수 있으며, 이를 통해 속도가 느린 모바일 네트워크에 생성된 요청 수를 줄이고 앱의 응답 속도를 향상시킵니다.

또한 개발자들은 JavaScript가 아닌 Java에서 서버측 백엔드 통합과 인증 코드의 실행을 선택할 수 있습니다.

**통합 푸시 알림**

통합 어댑터를 생성하는 과정에서 사용자들은 단일화된 중앙 인터페이스로부터 자동 알림을 미리 설정하기 위해 IBM Worklight의 균일한 푸시 아키텍처를 사용할 수 있습니다. IBM Worklight는 지원되는 디바이스를 위한 자체 통합 푸시 API의 강화를 통해 모바일 업체와의 전체 커뮤니케이션 과정을 개발자에게 완전히 공개합니다.



### 분산 개발

엔터프라이즈 모바일 개발은 거의 대부분 한 명의 개발자가 실행할 수 있는 간단한 과정이 아닙니다. 복잡한 엔터프라이즈 개발 환경은 앱의 서로 다른 부분을 작업하는 다수의 개발팀, 테스트팀 및 QA팀으로 구성되어 있는 경우가 가장 일반적이며, 때로는 심지어 서로 다른 지리적 위치에서 작업하는 경우도 있습니다. IBM Worklight는 다른 IBM 협력 툴과의 통합을 포함하여, 다양한 특징과 기능을 통해 이러한 시나리오들을 지원하도록 설계되었습니다.

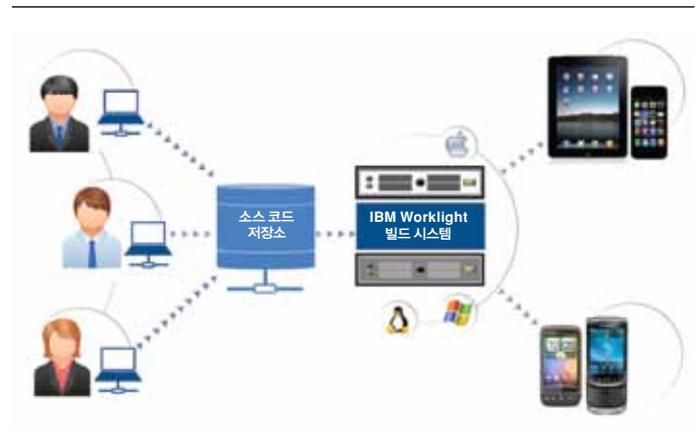
### 중앙 집중식 구축

IBM Worklight Builder는 독립 실행형 애플리케이션이며, IBM Rational Jazz Builder, Hudson 및 Lintbuild 등의 일반적인 중앙 구축 서비스에 더욱 쉽게 통합될 수 있습니다. 개발, 테스트 및 QA 단계에 참여하는 서로 다른 팀들은 중앙 집중식 구축 기능을 활용하여 하나의 공통된 코드 버전을 마무리할 수 있으며, 내부 애플리케이션 개발 과정의 협력과 자동화를 효과적으로 개선시킬 수 있습니다.

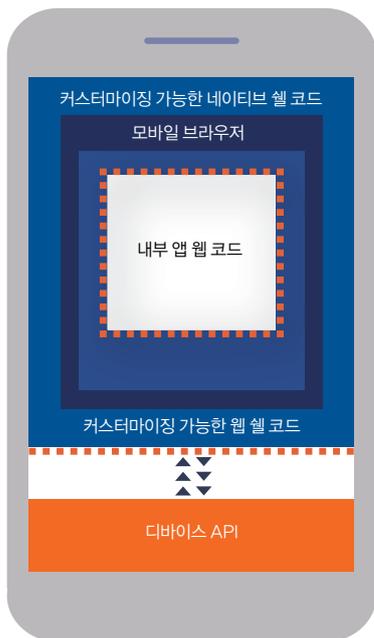
### 셸 방식

기업들은 자주 다양한 기술과 경험을 갖춘 다수의 개발 팀들을 고용합니다. "셸(Shell)" 방식은 이러한 엔터프라이즈들이 모바일 개발의 내부 장애물을 줄이고, 기술과 책임의 분류를 통해 조직 전체에 이 방식을 적용할 수 있도록 해줍니다.

IBM Worklight의 셸 방식은 앱 개발을 두 부분(외부 셸과 내부 애플리케이션)으로 분할시킵니다.



셀은 디바이스의 기본 기능에 JavaScript 액세스를 제공하는 커스터마이징 가능한 컨테이너로 구성되어 있습니다. 전문 개발자들로 구성된 전담 팀은 브랜딩, 보안 구성, 감사 및 인증 프레임워크에 대한 책임이 있습니다. 이러한 팀은 각각 다양한 정책과 브랜딩을 가지고 있는 다양한 셀을 생성할 수 있으며, 자동으로 파라미터에 대응하기 위해 각 셀 내에서 내부 앱을 강제로 실행할 수 있습니다. 이러한 파라미터들은 데이터 액세스, 특정 API 사용 및 다양한 브랜딩 등의 제한을 포함할 수 있습니다.



셀에 의해 실행된 엔터프라이즈 정책을 통해 부서별 개발팀들은 웹 언어 이외의 어떠한 것도 사용하지 않고 내부 앱을 손쉽게 구축할 수 있습니다. 이러한 팀들은 사용자 인터페이스, 비즈니스 로직 및 잠재적으로는 데이터 통합에만 집중해야 합니다. 앱의 배포는 세 개의 다른 채널을 통해 이루어질 수 있습니다.

- 새로운 버전의 내부 앱은 엔드 유저의 디바이스로 직접 전송 및 업데이트되는 반면(서비스 공급 업체의 조건에 따라 다름), 내부 앱은 중앙 집중식 구축 서버에 의해 셀로 통합될 수 있으며, 사설 또는 공개 앱스토어로 업로드될 수 있습니다.
- 셀은 기업이 인가한 애플리케이션의 디렉토리나 패키지로 구성될 수 있으며, 사용자들이 필요에 따라 다양한 내부 앱을 선택할 수 있도록 해줍니다.
- 셀은 비어 있는 채로 사용자에게 배포될 수 있으며, 그 후 사용자는 서버에 보관된 애플리케이션 저장소에 액세스하게 됩니다.

### IBM Worklight SDK

IBM Worklight SDK는 개발 프로세스를 지원하는 라이브러리 및 툴 세트로 구성되며, 백엔드 시스템, 클라우드 기반 서비스 및 인증 메커니즘과 클라이언트 코드의 통합을 간결하게 해줍니다. IBM Worklight SDK의 구성은 다음과 같습니다.

- JavaScript 클라이언트 API를 통해 IBM Worklight 서비스와 특정 환경에서의 기능 및 크로스 플랫폼 UI 유틸리티 세트에 대한 액세스를 제공합니다.
- Objective-C 클라이언트 API를 통해 Objective-C로 개발된 iOS 앱을 위한 IBM Worklight 서비스에 대한 액세스를 제공합니다.
- JavaScript 클라이언트 API를 통해 네이티브 Java 코드로 개발된 안드로이드 OS 애플리케이션을 위한 IBM Worklight 서비스에 대한 액세스를 제공합니다.

- JavaScript 서버 API를 통해 애플리케이션으로부터의 요청을 백엔드 시스템으로 보내기 전에 미리 처리할 수 있게 해주며, 백엔드 데이터를 애플리케이션으로 전송하기 전에 후처리를 수행할 수 있게 해줍니다. 또한 모바일 네트워크의 트래픽 부하 및 대기 시간을 줄이기 위해 다양한 소스에서 얻어진 백엔드 데이터의 메시지를 가능하게 해줌으로써 앱의 응답 속도를 향상시킵니다.
- 오프라인 액세스 API를 통해 개발자들이 액세스 실패 루틴을 지정할 수 있게 해주며, 실행 시간 및 자동 연결 재설정 중에 연결 상태를 확인할 수 있게 해줍니다.
- 서버와 클라이언트 통합 푸시 API를 통해 알림 메시지를 처리하고, 푸시 서비스에 등록하며, 통합 관리를 제공을 지원합니다.
- 위치 API를 통해 모바일 디바이스의 위치를 감지합니다.
- 어댑터 구성을 위한 스키마를 확인합니다.
- 개발 및 통합 프로세스와 모든 API를 문서화합니다.
- 프레젠테이션, 예제 및 샘플 코드를 포함하는 완전한 교육자료를 제공합니다.

## 런타임 서버 환경

### IBM Worklight Server

Java 기반의 IBM Worklight Server는 앱과 외부 서비스 및 엔터프라이즈 사이에 확장 가능한 게이트웨이입니다. IBM Worklight Server는 다양한 보안 기능들에 의해 제공되는 암호화된 커뮤니케이션, 백엔드 연결, 데이터 활용, 인증, 분석 및 운영 관리 기능을 지원합니다. IBM Worklight Server는 하기의 사항을 지원합니다.

- 클라우드 기반의 서비스뿐만 아니라, SOAP, REST, SQL, LDAP, SAP 등과 같이 광범위하게 사용되는 통합 기술들에 대한 다양한 엔터프라이즈 정보 시스템에 연결하는 어댑터 기술을 제공합니다.

- 여러 개의 데이터 스트림을 하나로 통합하고, 그것을 애플리케이션 사용자에게 제공하기 위한 멀티소스 데이터 매시업을 가능하게 해줍니다. 멀티소스 데이터 매시업은 사용자에게 올바른 데이터를 제공하기 위한 효과적인 방법일 뿐만 아니라, 시스템의 전체 트래픽을 줄여줍니다.
- 개발자들이 모바일 소비를 위한 백엔드 데이터를 제공하기 위해 요구되는 맞춤형 서버 로직을 추가할 수 있게 해줍니다. 이는 클라이언트와 서버 사이에 프로세스 배포에 도움이 될 뿐만 아니라, 조직 내의 데이터 보안 규정 처리에도 도움이 됩니다.
- 트랜잭션 호출 이외에, 애플리케이션 및 데이터 액세스를 보호하기 위해 엔터프라이즈 인증 인프라와 통합합니다. IBM Worklight 인증 인프라는 앱의 가용성을 높이기 위해 사용자의 오프라인 인증뿐만 아니라 다원적 로그인이나 다단계 로그인 프로세스에서부터 비인터랙티브 싱글 사인온(SSO) 통합에 이르기까지 다양한 유형의 인증 프로세스를 지원할 수 있을 만큼 유연합니다. 또한 IBM Worklight Server는 인증이 필요한 HTTP 기반 서비스와의 통합을 간단하게 만듭니다. Kerberos, Windows NTLM(NT LAN Manager), Basic 및 Digest 인증과의 통합은 서버 코드를 작성하지 않고 간단한 HTTP 어댑터 구성만으로 더욱 손쉽게 달성할 수 있습니다. 게다가 디바이스 기반의 애플리케이션 SSO를 지원하며, 만약 기존에 인증된 세션을 동일한 모바일 디바이스에서 이미 사용할 수 있을 경우, 자동적으로 앱을 인증할 수 있게 해줍니다.
- 공격(보안에 대한)을 방지하기 위해 표준 및 독점 보안 메커니즘을 사용합니다.
- 물리적 클러스터링을 통해 수십만의 사용자들과 다양한 애플리케이션을 지원하기 위해 더욱 손쉽게 확장됩니다.
- IBM Worklight Console에 의해 관리되고 액세스되는 앱 배포 기능과 버전 제어 기능을 제공합니다.

- IBM Worklight Server의 지속력과 애플리케이션에 제공되는 서비스들을 확인하는 모니터링 및 성능 관리 시스템과 통합합니다.
- 감사 및 보고서의 목적을 위해 사용자 채택 및 사용 데이터를 자동으로 수집하고, 보고 매트릭스의 커스터마이징 구성을 허용합니다. 원자료(Raw Data)는 추가 분석을 위해 조직이 사용하는 다양한 비즈니스 인텔리전스 툴을 통해 내보내기를 수행할 수 있습니다.

### IBM Worklight Console

IBM Worklight Console은 IBM Worklight Server와 배포된 앱, 어댑터 및 푸시 알림 서비스의 지속적인 관리를 위한 전용 웹 기반 사용자 인터페이스입니다. IBM Worklight Console을 통해 관리자가 수행할 수 있는 작업은 다음과 같습니다.

- 배포된 모든 어댑터와 애플리케이션을 모니터링하는 관리 용도의 대시보드에 대한 액세스가 가능합니다.
- 모든 푸시 알림 서비스, 이벤트 소스 및 관련 애플리케이션을 제어하고 모니터링할 수 있습니다.
- 동일한 애플리케이션의 다양한 버전들을 관리할 수 있으며, 모바일 운영 시스템의 버전과 유형에 따라 원격으로 애플리케이션을 사용하지 않도록 설정할 수 있습니다.
- 내장된 애플리케이션 채택 및 사용 보고서에 대한 액세스가 가능합니다.
- 앱의 액세스를 제어하기 위해 디바이스 기반의 액세스 제어 정책을 정의할 수 있습니다.

### IBM Worklight Device Runtime Component

IBM Worklight는 HTML5, 하이브리드 또는 네이티브 앱을 공급하는 클라이언트 런타임 코드를 제공하며, 다음과 같은 기능들이 포함됩니다.

- **액세스백엔드 데이터 및 트랜잭션:** 백엔드 시스템에 대한 IBM Worklight 서비스의 호출, 데이터의 검색 및 트랜잭션 실행을 위한 API입니다.
- **인증 및 보안:** 애플리케이션 데이터와 IBM Worklight Server에 대한 링크의 보안과 인증 절차의 관리를 위한 API와 코드입니다.
- **애플리케이션 관리:** IBM Worklight Console에 정의된 정책에 따라 새로운 애플리케이션 버전을 적용하고 애플리케이션을 사용하지 않도록 설정하기 위한 API와 코드입니다.
- **문제 해결:** 앱 내의 런타임 연결 문제들을 감지하고, 앱과 디바이스에 관한 문제 해결 정보를 수집하기 위한 코드입니다.
- **감사 및 분석을 위한 사용 보고:** 감사 및 분석을 위해 IBM Worklight Server의 앱으로부터 기본 데이터 및 맞춤형 데이터를 수집하는 API입니다.
- **크로스 플랫폼 호환성 API:** 디바이스 기능과 유용한 UI 작업을 위해 API를 통합하며, 서로 다른 환경에 존재하는 차이점을 드러내지 않습니다.
- **스킨애플리케이션:** 개발자들이 런타임 동안 디바이스의 폼 팩터에 대한 앱의 특징과 기능들을 조정할 수 있게 해주며, 동일한 디바이스 제품군의 다양한 버전에 맞게 최적화시킵니다.

런타임 클라이언트 환경은 다음과 같은 요소들로 구성되어 있습니다.

- **JavaScript 라이브러리:** JavaScript API를 실행하는 JavaScript 라이브러리 세트입니다. 이러한 라이브러리들은 대부분의 런타임 환경에서 사용할 수 있습니다(Javascript 라이브러리를 요구하지 않으면서, 각각 Objective-C와 Java로 작성되는 네이티브 iPhone 및 안드로이드 앱은 예외).

- **하이브리드 앱을 위한 네이티브 라이브러리:** 특정 디바이스 기능에 대한 액세스를 제공하는 네이티브 라이브러리 세트(iOS 및 안드로이드용)입니다. JavaScript로 작성된 앱은 이러한 라이브러리들에 직접 액세스를 하는 것이 아니라, 관련 JavaScript API를 통해 액세스합니다. 어떤 경우에는, 개발자가 제공한 웹 코드가 네이티브 코드에 의해 실행됩니다.
- **네이티브 앱을 위한 네이티브 라이브러리:** 기본으로 작성된 앱을 위한 IBM Worklight Server 기능에 액세스를 제공하는 iOS 및 안드로이드용 네이티브 라이브러리 세트입니다.
- **네이티브 코드 템플릿:** 개발자가 제공한 웹 코드를 실행하는 브라우저를 압축하는 iOS, 안드로이드, 블랙베리 및 윈도우폰 디바이스용 네이티브 코드 템플릿입니다.

**보안 및 인증 메커니즘**

IBM Worklight은 안전한 애플리케이션의 생성을 지원하는 다양한 메커니즘과 툴을 제공합니다.

다음은 플랫폼의 핵심 보안 기능 목록입니다.

메커니즘	이점	세부사항
암호화된 디바이스 스토리지	악성 코드 공격 및 디바이스 도난으로부터 민감한 정보를 보호합니다.	<ul style="list-style-type: none"> <li>• 앱에서 생성된 정보를 디바이스에 저장하기 위해 AES256 및 PKCS #5에 의해 생성된 암호화 키를 사용합니다.</li> <li>• 오프라인 사용자 인증을 허용합니다.</li> <li>• 선택적인 기본 성능 개선을 통해 JavaScript(매우 난해함)에서 실행됩니다.</li> </ul>
직접 업데이트	전체 설치 기반으로 업데이트된 앱 버전의 시기 적절한 전파를 보장합니다.	<ul style="list-style-type: none"> <li>• 새로운 버전의 코드는 앱의 수동 업데이트 없이 배포할 수 있습니다(웹 리소스에 적용 가능).</li> </ul>
원격 사용 불가능	전체 설치 기반에 대한 핵심 보안 업데이트를 적시에 채택하여 시행합니다.	<ul style="list-style-type: none"> <li>• 서버 콘솔은 허용된 앱 버전의 구성을 승인합니다. 관리자는 사용자가 네이티브 코드에 보안 업데이트를 설치하도록 강제로 요구할 수 있습니다.</li> </ul>
인증 프레임워크	인증 인프라와의 통합에 관련된 전체 비용 및 복잡성을 감소시킵니다.	<ul style="list-style-type: none"> <li>• JAAS (Java Authentication and Authorization Service) 개념과 인증 영역을 기반으로 하는 백엔드 인증 인프라와의 통합을 위해 설계된 서버측 아키텍처입니다.</li> <li>• Kerberos, NTLM, Basic 및 Digest 인증과의 통합을 실행할 준비가 되어 있습니다.</li> <li>• Web Services Security(WSS) 표준에 따라 X509 인증서와 서버 간 SOAP 커뮤니케이션을 암호화할 수 있습니다.</li> <li>• 세션 만료에 대한 비동기 로그인 요청을 위한 클라이언트 프레임워크입니다.</li> </ul>
서버 보호 수단	SQL 삽입을 방지하고 XSRF(Cross-site Request Forgery)로부터 보호합니다.	<ul style="list-style-type: none"> <li>• 준비된 질의문</li> <li>• 실행세션 쿠키에 대한 제출 자료의 유효성 검사</li> </ul>

메커니즘	이점	세부사항
엔터프라이즈 SSO 통합	기존의 엔터프라이즈 인증 및 사용자 자격 증명을 강화하고, 직원 소유의 디바이스들을 활성화시킵니다.	<ul style="list-style-type: none"> <li>클라이언트 메커니즘은 사용자 자격 증명을 획득 및 암호 화하고, 요청과 함께 서버로 전송합니다.</li> <li>암호화는 사용자가 제공한 개인식별번호(PIN), 서버 기밀 및 디바이스 ID를 통합합니다.</li> <li>자격 증명은 분실 또는 도난 당한 디바이스에서 검색되지 않습니다.</li> </ul>
가상 사설 네트워크(VPN)	직원이 소유한 디바이스들이나 엔터프라이즈 네트워크에서 허용되지 않는 유형의 디바이스를 위한 모바일 앱의 제공 및 운영을 가능하게 하며, 모바일 디바이스에 VPN 클라이언트를 설치할 수 없거나 관리하기 복잡할 경우에 제공됩니다.	<ul style="list-style-type: none"> <li>클라이언트 프레임워크와 서버 프레임워크는 SSL 기반의 VPN 역할을 수행합니다.</li> <li>클라이언트 프레임워크 계층에 미리 구성된 네트워크 액세스 제어 및 정책</li> <li>서버 프레임워크를 통해 업데이트된 네트워크 액세스 및 보안 조치</li> <li>민감한 데이터의 손상을 방지하기 위한 디바이스 암호화 스토리지</li> </ul>

IT 시스템 보안은 클라이언트 엔터프라이즈 내부에서 발생하는 부적절한 액세스에 대한 예방, 보호 및 대응을 통해 시스템과 정보를 보호합니다. 부적절한 액세스는 정보의 변경, 파괴 및 남용을 유발하거나, 다른 시스템 공격에 오용되는 결과를 발생시킬 수 있습니다. 보안에 대한 종합적인 접근 방식이 없다면, 어떠한 IT 시스템이나 제품도 완전히 안전한 것으로 여기지 말아야 하며, 어떠한 단일 제품이나 보안 조치들이라도 부적절한 액세스 방지에 완벽히 효과적일 수는 없습니다. IBM Worklight 시스템과 제품들은 종합적인 보안 접근 방식의 일부로 설계되었으며, 추가적인 운영 절차가 반드시 필요하고, 가장 효과적인 다른 시스템, 제품 또는 서비스가 필요할 수도 있습니다. IBM Worklight은 어떠한 악의적인 행위나 불법적인 행위로부터 영향을 받지 않는다는 것을 보증하지는 않습니다.

## 시스템 요구 사항

### 실제 운영 환경

IBM Worklight Server는 다음 운영 시스템에 설치할 수 있습니다.

- 윈도우 서버 2008, 64비트
- REHL(Red Hat Enterprise Linux) V4 SP3 또는 그 이상(SP4 권장), 64비트

IBM Worklight Server는 메타 데이터와 캐시된 백엔드 데이터를 저장하기 위해 다음과 같은 데이터베이스가 요구됩니다.

- My SQL 5.0.22 또는 그 이상(5.0.67 권장)
- Oracle 10g 및 11g

IBM Worklight Server는 다음 애플리케이션 서버에서 실행할 수 있습니다.

- Tomcat 7
- IBM WebSphere Application Server V7 이상

IBM Worklight Server는 높은 가용성과 확장성을 달성하기 위해 통합할 수 있습니다. 이러한 경우에는, 부하 분산 디바이스가 필요합니다. 이것은 스티키 세션(Sticky Session)을 지원하는 모든 상업적 부하 분산 디바이스, 소프트웨어 및 하드웨어가 모두 해당됩니다. 부하 분산디바이스는 리버스 프록시 및 SSL 가속기 역할을 선택적으로 할 수 있습니다.

### 개발 환경

IBM Worklight 개발 환경에는 IBM Worklight Server와 Eclipse 기반 Studio가 포함됩니다. 개발 환경은 다음 운영 시스템에서 지원됩니다.

- 윈도우 7, 비스타 또는 XP, 32비트 또는 64비트
- 맥킨토시 환경

개발을 위해 다음과 같은 데이터베이스가 지원됩니다.

- My SQL 5.0.22 또는 그 이상의 버전 (5.0.67 권장)

IBM Worklight Studio는 다음과 같은 Eclipse 배포가 필요합니다.

- Eclipse for Java EE Developers, Indigo 버전

IBM Worklight는 비즈니스 내에서 완벽한 모바일 개발, 배포 및 관리에 요구되는 모든 필수 요소들을 제공하는 IBM Mobile Foundation 제품군의 일부입니다. IBM Mobile Foundation은 다음 사항을 지원하는 기능들로 구성됩니다.

- IBM Worklight: 모바일 애플리케이션 개발 및 제공
- IBM Endpoint Manager: 완벽한 엔드-투-엔드 모바일 디바이스 관리 (MDM)
- IBM WebSphere Cast Iron Hypervisor Edition: 모바일 디바이스에 최적화된 백엔드 시스템과 클라우드 기반 서비스에 대한 고급 연결

추가적인 정보는 다음 웹사이트를 참고하시기 바랍니다.

[ibm.com/software/mobile-solutions/worklight](http://ibm.com/software/mobile-solutions/worklight)

## 추가 정보

모바일 애플리케이션 개발을 위한 IBM Worklight 자산에 관련된 추가적인 정보는 IBM 마케팅 담당자 및 IBM 비즈니스 파트너에게 문의하시거나, 다음 웹사이트에서 확인하실 수 있습니다.

[ibm.com/software/solutions/mobile-enterprise](http://ibm.com/software/solutions/mobile-enterprise)

IBM Global Financing은 가장 효율적인 비용과 전략적인 방법을 통해 귀하의 비즈니스에 필요한 소프트웨어를 확보할 수 있도록 지원합니다. 신뢰할 수 있는 클라이언트들과 파트너 관계를 유지하여 귀하의 비즈니스 및 개발 목표에 적합한 금융 솔루션을 제공하고, 이를 통해 효과적인 현금 관리가 가능해지며, 총소유비용을 개선시킬 수 있습니다. 핵심 IT에 투자하여 귀하의 비즈니스를 추진해 보시기 바랍니다. 추가적인 정보를 위해서는 다음 웹사이트를 참고할 수 있습니다.

[ibm.com/financing](http://ibm.com/financing)



IBM Corporation Software Group Route 100  
Somers, NY 10589

Produced in the United States of America  
April 2012

IBM, IBM 로고, ibm.com, Cast Iron, Jazz, Rational, Tivoli 및 WebSphere는 전세계 여러 국가에 등록된 International Business Machines Corp.의 상표입니다. 기타 제품 및 서비스 이름은 IBM 또는 해당 회사의 상표입니다. 현재 IBM 상표 목록은 웹 "저작권 및 상표 정보"([www.ibm.com/legal/copytrade.shtml](http://www.ibm.com/legal/copytrade.shtml))에 있습니다.

Linux는 미국 또는 기타 국가에서 Linus Torvalds의 등록 상표입니다.

Microsoft, Windows 및 Windows NT는 미국 또는 기타 국가에서 Microsoft Corporation의 상표입니다.

Java 및 모든 Java 기반 상표 및 로고는 Oracle 및/또는 그 자회사의 상표 또는 등록 상표입니다.

본 문서는 발행일 기준으로 최신이고 IBM은 이를 통지없이 변경할 수 있습니다. 본 문서에서 언급된 모든 오퍼링이 IBM이 영업하고 있는 모든 국가에서 제공된다는 것을 의미하지는 않습니다.

IBM 제품 및 프로그램과 함께 사용된 모든 제품 또는 프로그램의 운영에 관한 평가 및 검증은 전적으로 고객의 책임입니다. 본 문서의 모든 정보는 타인의 권리 침해, 상품성 및 특정 목적에의 적합성에 대한 묵시적 보증을 포함하여 묵시적이든 명시적이든 어떠한 종류의 보증 없이 "현상대로" 제공됩니다. IBM 제품은 제공된 제품에 적용된 계약의 이용 약관에 따라 보증됩니다.



Please Recycle