

## 4.7 M.ADI 산출물 적용

### 4.7.1 [M.ADI Maven Repository](#)

Android Studio에서 Gradle을 통해 사내에서 개발된 library나 plugin을 M.ADI Maven repository에서 검색하고 다운로드한다.

M.ADI Maven 저장소에서 관리되는 library와 plugin의 업데이트가 필요할 경우, version catalog를 통해 쉽게 최신 버전으로 업데이트할 수 있다.

[GAT 적용 사례 살펴보기](#)

#### settings.gradle.kts

```
pluginManagement {
    repositories {
        maven(url = "https://artifacts.mobis.co.kr:443/artifactory/MADI_maven")
    }
}

dependencyResolutionManagement {
    repositories {
        maven(url = "https://artifacts.mobis.co.kr:443/artifactory/MADI_maven")
    }
}
```

### 4.7.2 [Basic GUI Component](#)

M.ADI에서 모델 사양에 맞춰 정의한 GUI component를 사용함으로써 개발에 필요한 리소스 절감 및 UI 품질 일관성을 확보할 수 있다.

GAT에서는 다음과 같이 Basic GUI Component의 'TonalButton'을 사용하는 예제를 제공한다.

[GAT 적용 사례 살펴보기](#)

Module2NextScreen.kt

#### Module2NextScreen.kt

```
TonalButton(onClick = onNextClick) {
    Text(
        text = "Go back",
        style = ButtonText,
        modifier = Modifier.padding(horizontal = 60.dp, vertical = 15.dp),
    )
}
```

### 4.7.3 [Language Translator](#)

다국어 파일 신규 생성, 다국어 사양 신규 발행 등 적용이 필요한 시점에 간단한 Gradle 명령어를 통해 손쉽게 다국어 [strings.xml](#) 을 생성할 수 있다.

GAT에서는 다음과 같이 Language Translator plugin을 통해 생성한 strings.xml을 text에 적용하는 예제를 제공한다.

[GAT 적용 사례 살펴보기](#)

Module2Screen.kt

#### Module2Screen.kt

```
TonalButton(onClick = onNextClick) {  
    Text(  
        text = "Go to " + stringResource(id = R.string.LID0053779),  
        /* ... */  
    )  
}
```

#### 4.7.4 Publish Plugin

Library 및 Application을 [M.ADI Maven Repository](#)에 Publish 할 때 공통적으로 필요한 build 관련 feature들을 하나의 Publish라는 이름의 custom gradle plugin을 통해 개발 편의성을 제공한다.

GAT에서는 [Publish plugin이 제공하는 feature](#) 들 중

- Jetpack Compose Library의 version 호환성을 auto correction 하는 기능
- productFlavor 자동 생성.
- ProductFlavor 에 맞는 source folder 생성
- publish build logic 구성
- git tag 버전에 맞는 version 생성

을 적용한다.

[GAT 적용 사례 살펴보기](#)