M.ADI

Project Common Info

Exported on 09/12/2024

Table of Contents

1 1. Goal

- SW Development Members[1]는 AAOS를 기반으로 한 Multi Platform 및 Multi OEM을 대상으로, HMI App. 개발의 생산성, 안정성, 성능, Memory 최적화, 재사용성, 변경 용이성, 테스트 용이성 등등의 품질 요소들을 강화하기 위해
  필요한 요소들을 식별하고 설계 및 구현 또는 개발 지침, 개발 정책, 개발 프로세스 및 개발 가이드 등등을 정의한다.
- 위와 관련된 모든 일련의 활동들은 M.ADI(Mobis/Modern Android/Application Development Infrastructure)를 규정하고 개발하는 것으로 통칭한다.
  [Modern Application Development 란] Click here to expand...

  App Modernization(애플리케이션 현대화)

  - 현재의 비즈니스 필요를 가장 잘 지원하는 기술과 방법으로 구형 애플리케이션을 재구성하는 작업
  - Modern Application은 더 빠르게, 자주, 일관되게, 안전하게 가치를 실현할 수 있도록 하는 최신 기술, 아키텍처, 소프트웨어, 제공 방식, 운영 프로세스의 조합
  - Modern Android Application Development[2]
- 즉, M.ADI는 App. 개발자가 고품질의 App.을 쉽고 빠르게 개발 및 유지보수 할 수 있도록 돕는 모든 Infrastructure를 구성함으로써 최종적으로 App. 개발 및 유지 보수에 관한 전체 Process를 고도화하는 것을 목표로 한다.

1 https://confluence.mobis.co.kr/display/PCI/SW+Development+Members
2 https://developer.android.com/modern-android-development

2 2. Principles of Development Direction

Principal ID	Description
P1	AAOS를 기반으로 하기에, 구글의 Android App. 개발 시 지향하는 최신의 빌드, 개발 언어, 방법론 및 가이드 라인등등을 특별한 이유가 없다면 최대한 따른다.
P2	HMI App.의 Architecture Design은 Code Level로 강제화 될 수 있어야 하며, 모든 HMI App.들의 필수적인 공통 요소들(Essential Common Components)은 단일화된 Code들로 제공될 수 있어야 한다. • Architecture Design을 Code Level로 강제화하고, Essential Common Components들이 제공되기 위한 수단으로 App. Framework (see page 0)를 사용한다. • Android에서도 기본적으로 App. Framework가 제공되지만, 자유도가 높기에 요구사항들을 대응하면서 HMI App. 별로 Architecture Design이 파편화 되거나 공통으로 필요한 필수 요소들의 공용화가 어려워 진다. • 위와 같은 문제를 개선하기 위해 Mobis/Modern App. Framework(M.AF)이 필요하며, HMI App.은 M.AF에 App. Specific한 Feature들을 추가하여 개발되어야 한다.
	M.AF는 모든 HMI App.이 공통으로 필요한 필수 요소(Essential Common Components)들을 Archiecture Design(ex. MVVM)에 맞게 구성하여 제공해야하며, P1에 따라 가능한한 Android Jetpack Components (see page 0) 를 이용 또는 응용 하는것을 고려한다. • 특정한 공통 필수 요소 기능이 Android Jetpack Components (see page 0) 을 통해 제공되지 않는 경우, de-facto로 사용되는 public library(ex. retrofit[3], filament[4])들이 이용 또는 응용 될 수 있다.
	M.AT는 Archiecture Design에 맞게 구성된 Build Script, Directory Structure, Essential Common Components들에 대한 Template을 가능한한 Project Wizard로 제공하는 것을 고려한다. • Project Wizard에서 제공하는 App. Template은 기본적인 template 및 HMI App. Categroy 별 특성을 고려한 template들의 제공을 고려하며, 이를 Mobis/Modern App. Template(M.AT)로 지칭한다.
	M.AT의 Build Script는 M.AP를 이용하여 Multi Platform 및 Multi OEM을 고려한 Build Variants[5]를 제공하고, 이 Build Variants를 기반으로 HMI App.이 build 될 수 있도록 한다.

3 https://mvnrepository.com/artifact/com.squareup.retrofit2
4 https://github.com/google/filament
5 https://developer.android.com/build/build-variants

Principal ID	Description
P3	HMI App.은 Multi Platform 및 Multi OEM에 유연하게 대응하기 위해, Monolithic 형태를 지양하고 Modularization[6] 형태를 지향한다. ▪ App. specific한 feature들의 경우 Multi Platform 및 Multi OEM를 고려하여 변동부 및 고정부를 분리하고, 고정부는 Module화하여 개발한다. ▪ 고정부에 대한 Module화 및 적용 방안은 HMI App.에 대한 Multi Platform 및 Multi OEM 별 요구사항에 따라서 하기와 같이 두 가지 case로 나뉜다. • HMI App.을 구성하는 module들 및 module 내부에서 다수가 변동부여서 customization 처리가 오히려 복잡도를 상승 시키는 경우, • 고정부 Module은 library화 해서 M.ADI Maven Repository에 upload • HMI App.은 Multi Platform 및 Multi OEM 별 별도의 HMI App.으로 분리하고 library화된 고정부 module을 import 해서 사용 • HMI App.을 구성하는 module들 및 module 내부에서 소수가 변동부여서 customization 처리가 복잡도를 상승 시키지 않는 경우, • Build Variants[7] 를 이용하여 Module들을 조합하여 최종 HMI App.이 생성되도록 한다. --- 상기 개발 방향성에 대해서 상세화된 사항은 하기를 참조한다. ▪ APP 모듈화(고정부/변동부)를 위한 설계 원칙[8] ▪ APP 모듈화(고정부/변동부) 설계에 대한 Reference Design Document[9]
P4	AAOS SDK 및 M.AF를 통해 제공되지 않지만 HMI App. Category 별로 공통으로 필요한 Feature의 경우, copy & paste를 최대한 지양하고 library화를 지향한다. Library화는, • Public(ex. Android Jetpack[10])한 library들이 제공되는지를 우선적으로 검토하고, 이를 이용 또는 응용하여 개발한다. • Pubilc 한 library들이 없는 경우, Mobis Private Library를 개발한다.
P5	Runtime에 여러 HMI App.들의 concurrent한 request에 대해서 serialization하여 처리가 필요한 Feature의 경우, 특정 App.내 Service 제공(App.간 의존 관계 생성)은 불가하고 독립된 Service(Process)화를 고려해야 한다. • Service의 경우, Platform 내 Service(System Service)화의 필요성 여부에 대해서 Application SW Architecture 협의체에서 선 논의한 이후, App. & Platform SW Architecture 협의체 안건[11]으로 상정하여 논의한다.

6 https://developer.android.com/topic/modularization?hl=ko
7 https://developer.android.com/build/build-variants
8 https://confluence.mobis.co.kr/pages/viewpage.action?pageId=487558867#APP모듈화(고정부/변동부)설계구조자료-1.DesignPrinciple
9 https://confluence.mobis.co.kr/pages/viewpage.action?pageId=487558867#APP모듈화(고정부/변동부)설계구조자료-2.ReferenceDesignDocument
10 https://developer.android.com/jetpack/androidx/explorer
11 https://confluence.mobis.co.kr/pages/viewpage.action?pageId=455875008

Principal ID	Description
P6	HMI App. 개발의 선행 단계에서 생산성 강화를 위해 공통으로 필요한 Tool들의 경우, Gradle 및 IDE와 연동되는 Plugin들로 개발하는 것을 고려하고 이를 Mobis/Modern App. Plugin(M.AP)로 지칭한다.

3 3. Requirements

- 기존 Android/ccNC Platform에서의 HMI App. 개발 관련 Pain Points들을 기반으로 요구사항들을 도출하고 요구사항들에 대한 우선 순위를 정한다.

	Category	Description	Pirority	Note
1	Multi Langauge	신규 플랫폼에서 다국어 반영에 대한 프로세스 검토 필요. • 표5W 기준으로 UX팀에서 배포한 다국어 사양서에 대해서 개발자가 일일이 수동으로 strings.xml 파일에 반영하기에 누락이 되는 경우가 종종 발생함		• Language Translator[12][13]
2	SCM (Source Code Management)	특정 이슈 수정이나 사양 변경 건으로 추가된 코드에 대해서 히스토리 관리 필요 • 특정 이슈 수정이나 사양 변경 건으로 추가된 코드에 대해서 히스토리 파악이 어려워서 코드 관리 측면에서 어려움이 있음 • 특별한 comment없이 추가된 코드들을 삭제/수정을 했을 때 문제 발생 소지가 있음		• Commit Message Rule & Tool[14][15]

12 https://confluence.mobis.co.kr/display/PCI/Language+Translator
13 https://confluence.mobis.co.kr/display/PCI/Language+Translator
14 https://confluence.mobis.co.kr/pages/viewpage.action?pageId=377139871
15 https://confluence.mobis.co.kr/pages/viewpage.action?pageId=377139871

	Category	Description	Piro rity	Note
3	UX_Multi Langauge	[요청 목적] • 다국어 길이 넘침, 자동 개행 및 임의적 개행 삽입으로 인한 다국어 이슈 감소 • 현재 수작업을 통한 다국어 폰트 입력 작업을 자동화 하여 작업 효율화 [해결 방안] • 다국어 폰트 정보 자동 추출 필요 • 표시 공간 자유도 높임 다국어 번역 프로세스_표준 다국어 (ccNC) .pptx[16][17]		91. UX components (see page 4) 
4	UX_Validation	CR 혹은 변경요청에 대해서 UX 수정 사항의 SW 반영사항을 UX인원이 확인할 수 있는 방법 방안 필요		
5	Collaboration between PL and App	Platform 별 상이한 사양의 경우 각 차종 PL 분들의 사양 문의가 해당 모듈 담당자에게 근거 자료와 함께 요청됨. • 차종별 PL 분들이 상이하니 해당 모듈 담당자는 매번 동일한 내용을 공유해야 함. • 고객사 담당자 분들도 각각 문의 요청됨.		
6	Collaboration between QE and App	모든 이슈는 Application 모듈 담당자가 전부 할당 받아 재 분배하는 과정		

16 https://confluence.mobis.co.kr/download/attachments/346970528/%EB%8B%A4%EA%B5%AD%EC%96%B4%20%EB%B2%88%EC%97%AD%20%ED%94%84%EB%A1%9C%EC%84%B8%EC%8A%A4_%ED%91%9C%EC%A4%80%20%EB%8B%A4%EA%B5%AD%EC%96%B4%28ccNC%29%20.pptx?api=v2&modificationDate=1693179054000&version=1

17 https://confluence.mobis.co.kr/download/attachments/346970528/%EB%8B%A4%EA%B5%AD%EC%96%B4%20%EB%B2%88%EC%97%AD%20%ED%94%84%EB%A1%9C%EC%84%B8%EC%8A%A4_%ED%91%9C%EC%A4%80%20%EB%8B%A4%EA%B5%AD%EC%96%B4%28ccNC%29%20.pptx?api=v2&modificationDate=1693179054000&version=1

	Category	Description	Piro rity	Note
7	Collaboration between QE and App	Jira 이슈 할당 시 이슈 영상 및 로그, 재현 경로, 이슈 발생 시점의 충분한 설명이 명시되어 있는지 확인 필요		
8	Collaboration between QE and App	Application 문제 사항에 대해서 검증자 별로 중복 할당되는 경우가 있는데, 이미 할당된 문제 사항은 중복으로 Jira 생성되지 않을 수 있는 방법		
9	Collaboration between GUI and App	• GUI 사양이 오배포 되는 경우가 종종 있으며, 이에 대한 검증 시스템 개선 필요 • GUI 사양서의 부정확한 정보 또는 까다로운 중첩 Layer 표현 등의 한계점 존재 • 명확하게 관리 되지 않은 이미지 파일 관리 (중복 이미지 또는 해상도만 다른 동일 이미지 존재, 사용서에 삭제된 이미지 존재, 이미지 파일/디렉터리 path naming의 명확성, 애니메이션 이미지 적합성 등)		
10	Collaboration between UX and App	• 사양 검색의 어려우며 (특히 일반 사양들), PPT가 아닌 다른 System 구축과 함께 편리한 검색 시스템의 도입 필요 • 모호한 사양의 경우, ES 사양서 참조 대신 UX 사양서에 최대한 기술 또는 ES사양서 링크의 필요성 • CR로 인해 GUI/UX 사양이 변경됐으며 공식 Release의 시차로 인해 품질 이슈로 잡히게 되며, 이를 개발자가 Issue Reporter에서 설명해야 됨 (사양서 시스템의 변화가 필요한건 아닌지)		

	Category	Description	Piro rity	Note
11	DX (Developer Experience)	• 해당 Project에 대한 기술 자료가 파편화 되지 않고 하나의 구심점을 중심으로 분야/모듈 별로 잘 정리 및 관리 되었는지? (문서의 위치, 가독성 및 충분한 설명 등) (e.g. Repository 및 빌드가이드, Image-download 및 각종 Troubleshoot 문서, 인터페이스 문서, Know-how, Know-where, Tip 등등) • 개발자 사이의 R&R Argument 발생의 경우, 명확한 프로젝트 개발 방향이나 Guideline이 문서화 되어 있다면 사전에 예방 가능하지 않을지? 혹은 기능/모듈/경우에 따라 최적의 해결안 도출을 위한 시스템이 있으면 좋지 않을지?		▪ M.ADI Home[1819] ▪ AA 회의록[2021] ▪ AA 검토 결과[2223]
12	Collaboration between Platform and App	자체 SDK 배포 여부 및 Vendor API 지원 여부 확인 필요. Vendor API가 포함된 SDK배포시 App개발 속도를 향상 시킬수 있음.		▪ Platform SDK libraries published to M.ADI Maven[2425]
13	Collaboration between QE and App	• 설계/검증/평가팀 간 정기적 미팅을 통해 사양 공유 및 Q&A 진행을 통한 평가자 역량 향상 지원 • 신규 검증/평가 담당자에 대한 검증/평가 자체 교육 필요.		
14	Development Environment	Normal Boot / Flash Mode 제어 가능한 debug board 개발 필요 야간 빌드/다운로드 자동화로 개발시간 확보 및 재택근무 시 업무 가능 범위 확대 가능		

18 https://confluence.mobis.co.kr/display/PCI/M.ADI+Home
19 https://confluence.mobis.co.kr/display/PCI/M.ADI+Home
20 https://confluence.mobis.co.kr/pages/viewpage.action?pageId=346970561
21 https://confluence.mobis.co.kr/pages/viewpage.action?pageId=346970561
22 https://confluence.mobis.co.kr/pages/viewpage.action?pageId=422242343
23 https://confluence.mobis.co.kr/pages/viewpage.action?pageId=422242343
24 https://confluence.mobis.co.kr/display/PCI/Platform+SDK+libraries+published+to+M.ADI+Maven
25 https://confluence.mobis.co.kr/display/PCI/Platform+SDK+libraries+published+to+M.ADI+Maven

	Category	Description	Piro rity	Note
15	Develop ment Environm ent	개발시 필요한 공동 자재 정기적 구매 또는 상시 불출 프로세스 필요 (Ex : 디버그보드, 하네스, 키패드 PCB, 샤크 안테나 등)		
16	Collabora tion between GUI and App	전체 GUI 배포 일정 단축 방안 확인 필요. (1개월내) 기존 최소 2~3개월 소요되어 충분한 APP개발 일정 확보 불가. (대부분 고객사에서 사양 배포 지연으로 발생)		
17	Collabora tion between PL and App	브랜치 아웃 기준 수립 및 본 협의체와 협의 후 결정 필요 PL 성향에 따라 브랜치 아웃 기준이 달라 SW 반영 관리가 힘듬.		
18	Collabora tion between UX and App	• UX 사양서의 Scene ID 를 수동으로 관리하고 있음 • IA 구조로 자동으로 관리되어 할당되고 관리 되는 방안 필요		
19	Collabora tion between UX and App	• 고객사 요청으로 VCRM 기능이 존재하나 관련 데이터를 모비스도 수집하여 설계에 반영이 필요함 • 모비스 설계를 위해 사용자 사용 빈도 및 패턴 분석을 위한 사용자 모니터링 기능(VCRM)		

	Category	Description	Piro rity	Note
20	Collaboration between UX and App	• 다국어의 적용 시, 단어 단위의 자동 개행은 적용되고 있음 • 그러나 1개 단어의 표시 공간 부족에 대한 로직 적용 필요 • 1개 단어가 표시 공간 초과 시, 데쉬 적용 후 개행 처리될 수 있는 로직 개발 필요 • https://jira.ccos.dev/browse/CCNCUX-826 (예시) AAAAAAA (표시공간 초과) -->로직 적용 후 표시 AAA- AAA		
21	Collaboration between UX and App	• 시스템과 내비가 동일하게 사용하는 다국어가 동일하지 않는 경우가 있음. • 각 사별로 다국어 파일을 따로 관리하다보니 반영 시점 등이 달라 이슈가 발생함 • 내비 쪽의 다국어 OR 개발소스를 반영하여 바로 H/U SW 반영 방안 필요 • 시스템/내비의 다국어를 1개로 관리 (오토에버 협의 필요)		
22	Collaboration between UX and App	• 기능 적용 지역/차종 파악이 안되 다국어 제공 시점의 어려움이 있음. • 그리고 설정 사양 중에 default 값 등은 UX 사양으로 관리하지 않는데 UX에 확인 요청이 많음 • CR/신규 사양 적용 시, 적용 지역/차종 별 사양을 모든 담당자(SW/UX/GUI/평가)들이 파악할 수 있는 방법이 있었으면 함 (설정의 default 값 등 포함)		

	Category	Description	Piro rity	Note
23	Collaboration between UX and App	• String 배포 시, 마스터 바전에 반영 여부는 확인이 가능하나 브랜치/평가 SW에 어떤 string이 반영되었는지는 매번 확인 필요함. • 각 브랜치 SW(평가 SW 포함)별로 반영된 string 버전을 확인할 수 있는 방법 필요		
24	Collaboration between UX and GUI	• 공통 GUI 가이드를 따로 관리하는 것도 좋지만 GUI 가이드 확인 시에 여러 문서를 다 찾아봐야 함. 예: 리스트 팝업 확인 시에 해당 모드 가이드를 찾고, 팝업 가이드 이후에 공통(리스트) 가이드를 모두 확인해야 함 • GUI 가이드 문서를 좀 더 효율적으로 확인할 수 있는 방법 필요		
25	Collaboration between UX and App	• s/w는 일괄적으로 적용할 수 있음에도 불구하고 특정 다국어 번역이 안된 경우 다국어 번역이 완료될때까지 s/w에서 관리 필요함. • 해외향 전반적인 다국어 변경 시 다국어 변경을 s/w에서 찾아서 확인해야 되는 불편함이 있음. • 다국어 변경 시마다 s/w 에서 검토하는 시간 단축할 수 있도록 UX에서 다국어를 시스템 공용 리소스로 직접 등록하고 S/W는 UX에서 이미 정의한 LID를 사용하는 방식으로 검토 필요. (STEP에 따라 다국어 변경되는 경우도 고려 필요)		

	Category	Description	Pirority	Note
26	UX_Multi Langauge	타겟 환경에서 언어를 쉽게 바꿔서 다국어 구문 확인이 가능하도록 설정 인터페이스 제공 필요. • 향지별 지원 언어 목록이 다름. 전체 언어 테스트를 위해서는 향지 변경 후 리부팅이 여러번 필요 • 엔지니어링 모드 설정 등을 통해서 전체 언어 변경이 가능하도록 하는 다국어 확인 모드 추가 • 다국어 확인 모드에서는 SEEK UP/DOWN 등의 H/W 키로 언어 변경이 가능하도록 기능 제공 필요		
27	Collaboration between PL and App	여러 모듈이 협업해야 하는 복합 사양 구현 시 해당 기능 개발을 책임지고 관리할 수 있는 인원이 필요. 복합 사양의 경우 책임 주체가 불분명해 진행 자체가 더뎌지는 경우가 많음. • PL 중 1인이 전체 진행 상황을 관리 또는 • 사양 구현 담당자 중 구현 비율이 높은 1인이 관리 및 이에 맞는 권한 부여 필요		
28	Collaboration between PL and App	전체 App 공통 적용 사양 등의 경우 목표 일정을 명확히 하여 단기간 내 동시 적용할 수 있도록 관리 필요. 개발자가 처한 상황이나 성향에 따라 적용 시점이 달라, 지연 되는 모듈에 의해 먼저 적용한 앱들의 적용 사항을 롤백해야 하는 경우가 생김		
29				

4 4. Key Decisions

- Principles of Development Direction과 Requirements를 기반으로, M.ADI를 위한 주요 결정 사항들에 대해서 기술한다.

4.1 4.1 Development Environment

4.1.1 4.1.1 Build Tool

- 개발 생산성 측면에서 SDK 기반으로 Android Studio IDE에서 HMI App.을 개발하는 것이 효율적이기에, Android Studio IDE에서 지원되는 gradle을 build tool로 선택한다.
- 단, HMI App.의 개발 산출물을 system image에 integration 하기 위한 build tool은 soong을 이용하며, 자세한 사항은 4.1.3 에 기술된 내용을 참조한다.

4.1.2 4.1.2 IDE

- System App Development Environment on Android Studio[26]

4.1.3 4.1.3 Rationale for Decision

- Pre-Installed Application Development Environment[27]

4.2 4.2 UI Toolkit & Language

4.2.1 4.2.1 UI Toolkit

- 선언형 기반의 확장성이 좋은 Jetpack Compose를 Principles of Development Direction of M.ADI (see page 5)의 P1에 따라서 기본 UI Toolkit으로 사용한다.
- 단, 과거 제품에서 Layout XML 기반으로 기 개발된 HMI App.을 customization 또는 AAOS 내장 HMI App.을 customization하여 개발하는 것이 더 효용성이 높은 경우는, Layout XML 을 UI Toolkit으로 사용한다.

4.2.2 4.2.2 Language

- Kotlin 및 Java 중 개발자가 선호하는 언어를 사용해도 되지만, Principles of Development Direction of M.ADI (see page 5)의 P1에 따라서 kotlin을 사용하는 것을 권장한다.

---

26 https://confluence.mobis.co.kr/display/PCI/System+App+Development+Environment+on+Android+Studio
27 https://confluence.mobis.co.kr/display/PCI/Pre-Installed+Application+Development+Environment

4.2.3 4.2.3 Rationale for Decision

- UI Toolkit & Language[28]

4.3 4.3 SCM (Source Code Management)

4.3.1 4.3.1 Commit Message Rule & Tool

- Commit Message Rule & Tool[29]

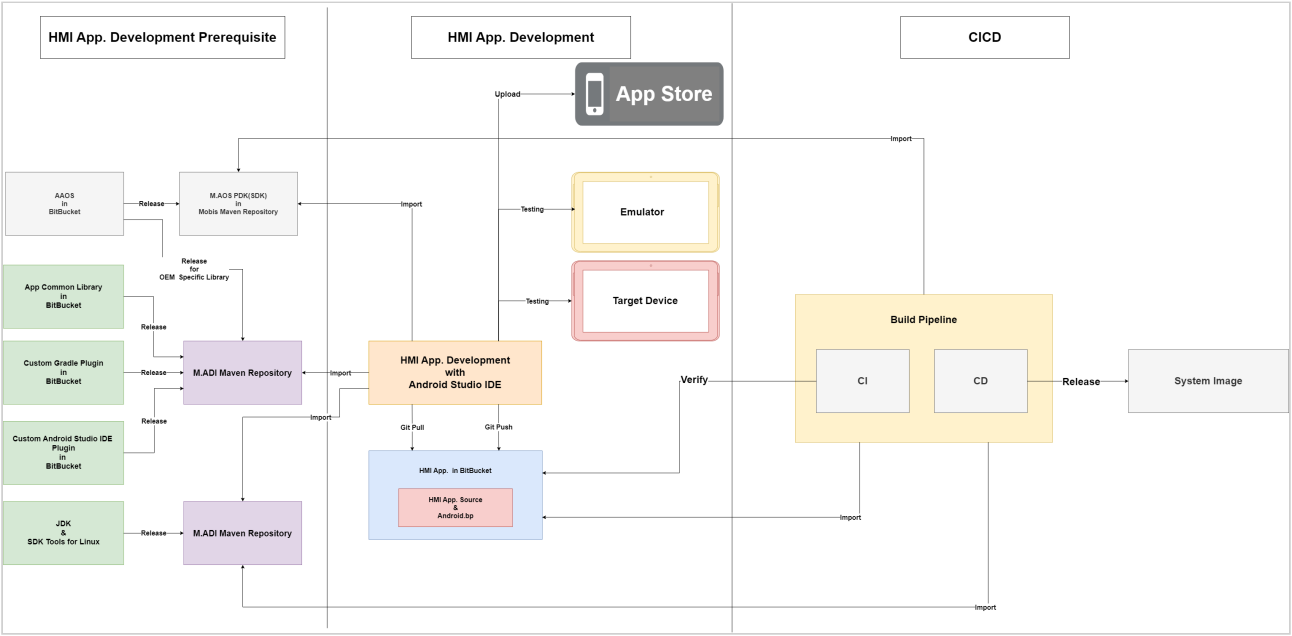
4.3.2 4.3.2 Lint for Coding Convention

- Ktlint (Kotlin Lint)[30]

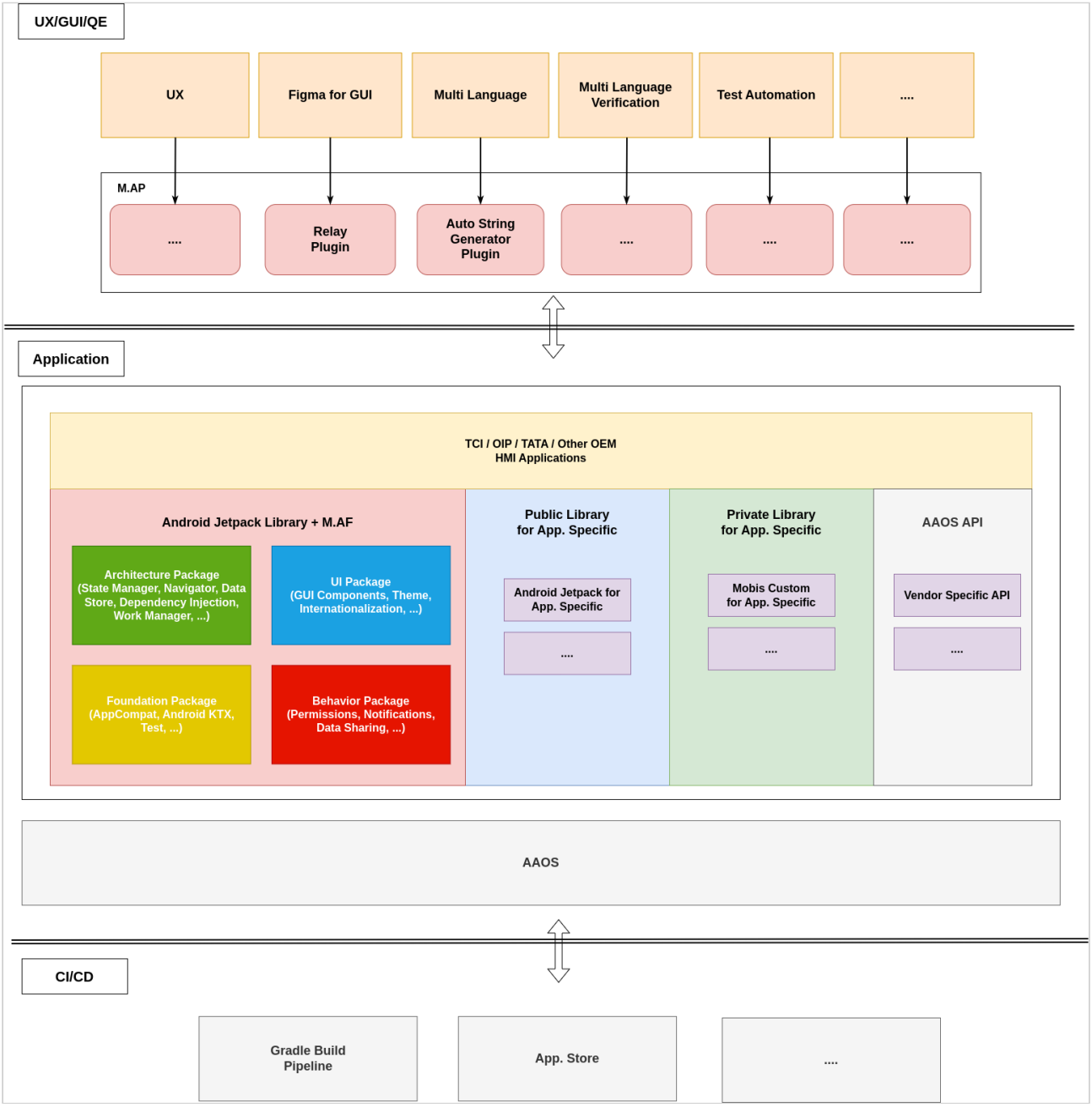
28 https://confluence.mobis.co.kr/pages/viewpage.action?pageId=346970539
29 https://confluence.mobis.co.kr/pages/viewpage.action?pageId=377139871
30 https://confluence.mobis.co.kr/pages/viewpage.action?pageId=460855358

5 5. Big Picutre Overview

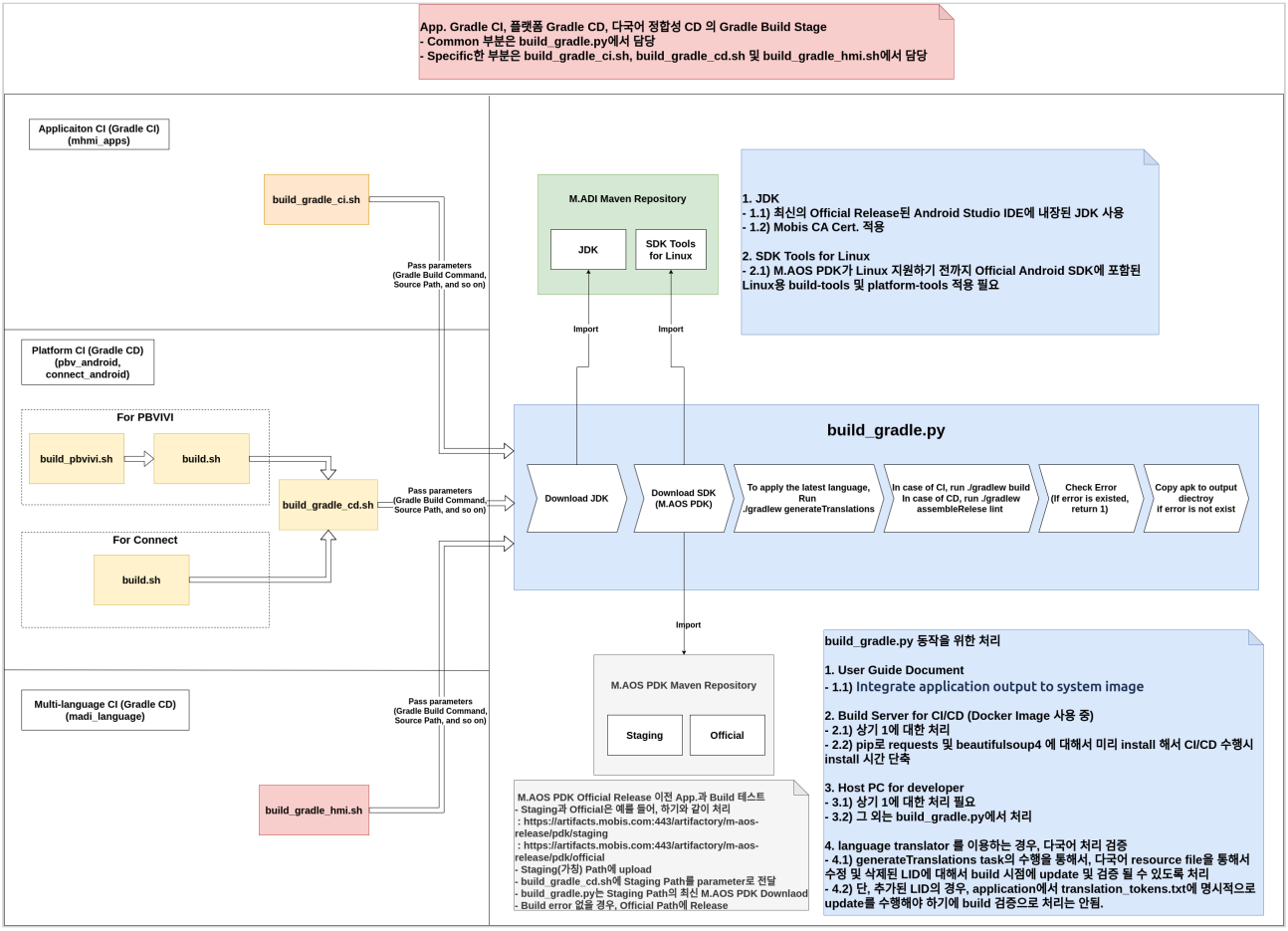
5.1 5.1 Application Development Flow Overview



5.2 5.2 Application Internal Overview



5.3 5.3 Application CI/CD Overview


App. Gradle CI, 플랫폼 Gradle CD, 다국어 정합성 CD 의 Gradle Build Stage
- Common 부분은 build_gradle.py에서 담당
- Specific한 부분은 build_gradle_ci.sh, build_gradle_cd.sh 및 build_gradle_hmi.sh에서 담당
Applicaiton CI (Gradle CI)
(mhmi_apps)
build_gradle_ci.sh
Pass parameters
(Gradle Build Command,
Source Path, and so on)
Platform CI (Gradle CD)
(pbv_android,
connect_android)
For PBVIVI
build_pbvivi.sh
build.sh
build_gradle_cd.sh
Pass parameters
(Gradle Build Command,
Source Path, and so on)
For Connect
build.sh
Multi-language CI (Gradle CD)
(madi_language)
Pass parameters
(Gradle Build Command,
Source Path, and so on)
build_gradle_hmi.sh
M.ADI Maven Repository
JDK
SDK Tools
for Linux
Import
Import
1. JDK
- 1.1) 최신의 Official Release된 Android Studio IDE에 내장된 JDK 사용
- 1.2) Mobis CA Cert. 적용
2. SDK Tools for Linux
- 2.1) M.AOS PDK가 Linux 지원하기 전까지 Official Android SDK에 포함된
Linux용 build-tools 및 platform-tools 적용 필요
build_gradle.py
Download JDK
Download SDK
(M.AOS PDK)
To apply the latest language,
Run
./gradlew generateTranslations
In case of CI, run ./gradlew build
In case of CD, run ./gradlew
assembleRelese lint
Check Error
(If error is existed,
return 1)
Copy apk to output
diectroy
if error is not exist
Import
M.AOS PDK Maven Repository
Staging
Official
M.AOS PDK Official Release 이전 App.과 Build 테스트
- Staging과 Official은 예를 들어, 하기와 같이 처리
: https://artifacts.mobis.com:443/artifactory/m-aos-release/pdk/staging
: https://artifacts.mobis.com:443/artifactory/m-aos-release/pdk/official
- Staging(가정) Path에 upload
- build_gradle_cd.sh에 Staging Path를 parameter로 전달
- build_gradle.py는 Staging Path의 최신 M.AOS PDK Downlaod
- Build error 없을 경우, Official Path에 Release
build_gradle.py 동작을 위한 처리
1. User Guide Document
- 1.1) Integrate application output to system image
2. Build Server for CI/CD (Docker Image 사용 중)
- 2.1) 상기 1에 대한 처리
- 2.2) pip로 requests 및 beautifulsoup4 에 대해서 미리 install 해서 CI/CD 수행시 install 시간 단축
3. Host PC for developer
- 3.1) 상기 1에 대한 처리 필요
- 3.2) 그 외는 build_gradle.py에서 처리
4. language translator 를 이용하는 경우, 다국어 처리 검증
- 4.1) generateTranslations task의 수행을 통해서, 다국어 resource file을 통해서 수정 및 삭제된 LID에 대해서 build 시점에 update 및 검증 될 수 있도록 처리
- 4.2) 단, 추가된 LID의 경우, application에서 translation_tokens.txt에 명시적으로 update를 수행해야 하기에 build 검증으로 처리는 안됨.

6 6. Architecture Design

- HMI App.의 Architecture Deign에 대해서 기술한다.

6.1 6.1 Concept of Architecture Design

Unable to render include or excerpt-include. Could not retrieve page.

6.2 6.2 Architecture Design with Template

- Application Template[31]

31 https://confluence.mobis.co.kr/display/PCI/Application+Template

7 7. Output of M.ADI

- Maven Repository for M.ADI[32]
- Release Note for M.ADI[33]
- User Guide for M.ADI[34]
- R&D for M.ADI[35]
- Bitbucket of M.ADI[36]

32 https://confluence.mobis.co.kr/display/PCI/Maven+Repository+for+M.ADI
33 https://confluence.mobis.co.kr/display/PCI/Release+Note+for+M.ADI
34 https://confluence.mobis.co.kr/display/PCI/User+Guide+for+M.ADI
35 https://confluence.mobis.co.kr/pages/viewpage.action?pageId=346970535
36 https://bitbucket.mobis.co.kr/projects/M_ADI

8 8. Development Roadmap

8.1 8.1 Rule of Roadmap Construction

Rule ID	Description
R1	가능한 HMI App. 개발에 바로 사용 될 수 있도록 MVP(Minimum Viable Product)로 단계적으로 Release 한다. 이를 위해 요구사항들의 우선 순위를 고려하여 동작 가능한 Feature 단위로 분할하여 개발한다. • 단, 개발 timeline을 고려시 guide line 및 reference architecture design 등과 같이 document 및 example code등이 먼저 제공되고 이후 MVP로 제공하는 것을 고려할 수 있다.
R2	개발 Roadmap은 Short Term, Medium Term, Long Term으로 3 phase 로 나눠서 진행한다. ▪ 각 개발 Phase에서는 개발 일정, 개발 Resource 및 산출물에 대해서 기술한다. ▪ 각 개발 Phase에서는 개발 산출물에 대한 Timeline을 단축하기 위해서, 가능한한 paralle 하게 진행 가능하도록 Planning 한다. ▪ 각 개발 항목별로 M.ADI Jira Project[37]에서 Epic & Stroy/Task로 Jira Ticket으로 관리하며, code 산출물은 bitbucket을 통해서 release 한다. ▪ M.ADI에 대한 Jira project에 대한 권한은 M.ADI Jira & Confluence 권한 신청[38] 을 참조 ▪ 각 개발 Phase 별로 개발 항목에 대한 output이 빠르게 release되면, 개발 plan을 조정하여 앞의 phase에서 개발 될 수 있도록 한다.

8.2 8.2 Projects of M.ADI

	Project	Description
1	M.ADP (Mobis/Modern Android/ Application Development Process)	HMI App. 개발 지침, 개발 정책, 개발 프로세스 및 개발 가이드 등을 포함한 문서들

37 https://jira.mobis.co.kr/secure/RapidBoard.jspa?rapidView=2399&view=planning.nodetail&issueLimit=100
38 https://confluence.mobis.co.kr/display/PCI/M.ADI+Home#M.ADIHome-2.M.ADIJira&Confluence권한신청

	Project	Description
2	M.AF (Mobis/Modern Android/ Application Framework)	HMI App.의 Architecture Design을 Code Level로 강제화 하고, 모든 HMI App.들의 필수적인 공통 요소들(Essential Common Components)을 포함한 App. Framework
3	M.AT (Mobis/Modern Android/ Application Template)	Project Wizard를 통해 제공되는 App. Template • M.AF를 기반으로 제공하는 기본적인 template • HMI App. Categroy 별 특성들을 고려하여 M.AF + Public Library + Private Library 등등을 조합한 template들
4	M.AP (Mobis/Modern Android/ Application Plugin)	HMI App.의 개발 생산성을 강화하기 위해 필요한 도구들
5	M.AE (Mobis/Modern Android/ Application Examples)	HMI App. 개발자들에게 example code들을 통해서 다양한 use case에서 M.ADI의 산출물들의 "How to Use"에 대한 이해도를 높이기 위해 제공하는 application • 예를 들어, M.AF의 GUI Component들에 대한 "How to Use"에 대한 다양한 use case 기반 example code 제공
6	M.ADC (Mobis/Modern Android/ Application Design Compose)	구글의 Automotive Design for Compose Toolkit[39](이하 DesinCompose Toolkit)을 이용하여, 개발 Tool 기반으로 UI와 Data를 SoC(Separation of Concerns)하는 개발 패러다임의 전환을 개발 Process 정립

8.3 8.3 SW Development Members

- M.ADI의 SW 개발을 진행하는 Members들에 대해서 기술한다.

	Cell	Member
1	**인포홈/설정APP팀**	민정기(MIN JUNG KI) 인포홈/설정APP팀
2		이재학(LEE JAE HAK) 인포홈/설정APP팀
3		이정원(LEE JUNG WON) 인포홈/설정APP팀
4		김인규(KIM IN KYU) 인포홈/설정APP팀

39 https://google.github.io/automotive-design-compose/

	Cell	Member
5		최승구(CHOI SEUNG KU) 인포홈/설정APP팀
6		최영림(CHOI YOUNG RIM) 인포홈/설정APP팀
7		신철규(SHIN CHEOL GYU) 인포홈/설정APP팀
8	**인포방송/미디어APP팀**	서종원(SEO JONG WON) 인포방송/미디어APP팀
9		이하영(LEE HA YOUNG) 인포방송/미디어APP팀
10	**인포차량연동APP팀**	김경호(KIM KYUNG HO) 인포차량연동APP팀
11	**인포커넥티비티APP1팀**	정요셉(JUNG JOSEPH) 인포커넥티비티APP1팀
12	**인포커넥티비티APP2팀**	조란(JO RAN) 인포커넥티비티APP2팀
13	**인포CCS APP팀**	고석현(KHO SEOK HYUN) 인포CCS APP팀
14	**SWH** (M.ADI Work Package)	Kim Byung Joo (PO)
15		Pashchenko Andrii
16		Hetman Taras
17		Denys Kalashnyk
18		Soloviov Glib
19		Spektor Dmytro
20	**SWH (**CI/CD IVI Work Package**)**	Yunyeong Kim

8.4 8.4 Short Term

8.4.1 8.4.1 Milestone

- 개발 Resource 및 Due Date이 결정되면, 개발 대상 항목들을 선정하고 개발 항목들에 대한 Milestone을 세운다.
- 개발 Due Date: 2023. 11. 30

8.4.2 8.4.2 M.ADP

	Output	Description	Ticket
1	Development Environment ▪ Build Tool ▪ IDE ▪ CI/CD	▪ HMI App. 개발에 대한 Build Tool 정의 ▪ IDE 기반 System HMI App. 개발을 위한 permission, key signing 방법 정립 및 가이드 ▪ CI/CD와 연동 검토 및 정의	key summary status Jira 프로젝트가 존재하지 않거나 볼 권한이 없습니다. Jira에서 이러한 이슈 보기[40] key summary status Jira 프로젝트가 존재하지 않거나 볼 권한이 없습니다. Jira에서 이러한 이슈 보기[41]

40 https://jira.mobis.co.kr/secure/IssueNavigator.jspa?reset=true&jqlQuery=project+%3D+MADI+AND+%22Epic+Link%22+%3D+MADI-6++++++&src=confmacro

41 https://jira.mobis.co.kr/secure/IssueNavigator.jspa?reset=true&jqlQuery=project+%3D+MADI+AND+%22Epic+Link%22+%3D+MADI-107+++++++&src=confmacro

	Output	Description	Ticket
2	UI Toolkit & Language	▪ HMI App.개발을 위한 UI Toolkit 정의 ▪ HMI App. 개발을 위한 Language 정의	key summary status ⚠ Jira 프로젝트가 존재하지 않거나 볼 권한이 없습니다. Jira에서 이러한 이슈 보기[42]
3	UX & GUI & SW Collaboration Work Model	▪ SW 개발 과정에서 HMI, GUI, 개발자 사이의 효율적인 협업 모델에 대해 도출	key summary status ⚠ Jira 프로젝트가 존재하지 않거나 볼 권한이 없습니다. Jira에서 이러한 이슈 보기[43]

42 https://jira.mobis.co.kr/secure/IssueNavigator.jspa?reset=true&jqlQuery=project+%3D+MADI+AND+%22Epic+Link%22+%3D+MADI-7+++&src=confmacro

43 https://jira.mobis.co.kr/secure/IssueNavigator.jspa?reset=true&jqlQuery=project+%3D+MADI+AND+%22Epic+Link%22+%3D+MADI-9+&src=confmacro

	Output	Description	Ticket
4	Open Source Management	• Open Source Managment에 관해 제품 개발 관점에서 관리 필요한 항목들을 식별 • 식별된 항목들 별로 Mangement에 대한 전략에 따라 필요한 사항들에 대해 정의	key summary status ⚠ Jira 프로젝트가 존재하지 않거나 볼 권한이 없습니다. Jira에서 이러한 이슈 보기[44]
5	SCM (Source Code Management)	• Source Code Management에 관해 제품 개발 관점에서 필요한 항목들을 식별 • 식별된 항목들 중에서 우선 순위가 높은 항목들에 대해서 검토	key summary status ⚠ Jira 프로젝트가 존재하지 않거나 볼 권한이 없습니다. Jira에서 이러한 이슈 보기[45]

44 https://jira.mobis.co.kr/secure/IssueNavigator.jspa?reset=true&jqlQuery=project+%3D+MADI+AND+%22Epic+Link%22+%3D+MADI-87+&src=confmacro

45 https://jira.mobis.co.kr/secure/IssueNavigator.jspa?reset=true&jqlQuery=project+%3D+MADI+AND+%22Epic+Link%22+%3D+MADI-8+&src=confmacro

8.4.3 8.4.3 M.AF

	Output	Description	Ticket
1	Theme, Image Asset and Font Development	▪ Theme, Image Asset 및 Font 등의 개발 방향성에 대한 수립 및 개발에 필요한 요소들을 식별하고 설계 및 구현	key summary status Jira 프로젝트가 존재하지 않거나 볼 권한이 없습니다. Jira에서 이러한 이슈 보기[46]

46 https://jira.mobis.co.kr/secure/IssueNavigator.jspa?reset=true&jqlQuery=project+%3D+MADI+AND+%22Epic+Link%22+%3D+MADI-60++&src=confmacro

	Output	Description	Ticket
2	GUI Components of UI Package	▪ GUI Components에 대한 HMI/UX 사양 정의 ▪ GUI Components에 대한 Development Workflow 정의 ▪ GUI Components에 Code 산출물	key summary status Jira 프로젝트가 존재하지 않거나 볼 권한이 없습니다. Jira에서 이러한 이슈 보기[47] key summary status Jira 프로젝트가 존재하지 않거나 볼 권한이 없습니다. Jira에서 이러한 이슈 보기[48]

47 https://jira.mobis.co.kr/secure/IssueNavigator.jspa?reset=true&jqlQuery=project+%3D+MADI+AND+%22Epic+Link%22+%3D+MADI-11+&src=confmacro

48 https://jira.mobis.co.kr/secure/IssueNavigator.jspa?reset=true&jqlQuery=project+%3D+MADI+AND+%22Epic+Link%22+%3D+MADI-112++&src=confmacro

	Output	Description	Ticket
3	Reference Analysis for Architecture Design	▪ Modern Android App Architecture에 대한 분석 ▪ Android App Architecture의 구성요소에 대해 분석 ▪ Android Architecture Sample들과 Sample App들을 분석	key summary status Jira 프로젝트가 존재하지 않거나 볼 권한이 없습니다. Jira에서 이러한 이슈 보기[49]
4	Logger	• Android HMI App. 개발시 사용 가능한 Logger들을 검토 • 과거 Android 기반 플랫폼 로그 시스템 및 불편한 점에 대해 검토 • 검토된 내용을 기반으로 Logger 방향을 결정하고 이를 바탕으로 개발	key summary status Jira 프로젝트가 존재하지 않거나 볼 권한이 없습니다. Jira에서 이러한 이슈 보기[50]

49 https://jira.mobis.co.kr/secure/IssueNavigator.jspa?reset=true&jqlQuery=project+%3D+MADI+AND+%22Epic+Link%22+%3D+MADI-12+&src=confmacro

50 https://jira.mobis.co.kr/secure/IssueNavigator.jspa?reset=true&jqlQuery=project+%3D+MADI+AND+%22Epic+Link%22+%3D+MADI-18+&src=confmacro

8.4.4 8.4.4 M.AP

	Output	Description	Ticket
1	Generic Plugin Development	▪ Custom한 plugin을 개발하기 위한 guide 및 sample 개발	key summary status Jira 프로젝트가 존재하지 않거나 볼 권한이 없습니다. Jira에서 이러한 이슈 보기[51]
2	Auto Multi-Langague Generator Plugin	• HMI App.의 Multi-Language 자동 생성 Plugin 개발	key summary status Jira 프로젝트가 존재하지 않거나 볼 권한이 없습니다. Jira에서 이러한 이슈 보기[52]

51 https://jira.mobis.co.kr/secure/IssueNavigator.jspa?reset=true&jqlQuery=project+%3D+MADI+AND+%22Epic+Link%22+%3D+MADI-14+&src=confmacro

52 https://jira.mobis.co.kr/secure/IssueNavigator.jspa?reset=true&jqlQuery=project+%3D+MADI+AND+%22Epic+Link%22+%3D+MADI-15+&src=confmacro

8.4.5 8.4.5 M.AT

	Output	Description	Ticket
1	Custom Application Template Plugin Development	• Android Stuido IDE의 Project Wizard를 통해서 custom한 App. Template을 생성하기 위한 App.Template Plugin 개발	key summary status Jira 프로젝트가 존재하지 않거나 볼 권한이 없습니다. Jira에서 이러한 이슈 보기 [53]
2	Generic Application Template Development	▪ 구글 MAD(Modern Android App Development)에 기반한 MVVM + Modularization Template 개발	key summary status Jira 프로젝트가 존재하지 않거나 볼 권한이 없습니다. Jira에서 이러한 이슈 보기 [54]

53 https://jira.mobis.co.kr/secure/IssueNavigator.jspa?reset=true&jqlQuery=project+%3D+MADI+AND+%22Epic+Link%22+%3D+MADI-16++&src=confmacro

54 https://jira.mobis.co.kr/secure/IssueNavigator.jspa?reset=true&jqlQuery=project+%3D+MADI+AND+%22Epic+Link%22+%3D+MADI-17++&src=confmacro

8.4.6 8.4.6 M.AE

	Output	Description	Ticket
1	Example Container Application Development	• HMI App. 개발자들에게 M.ADI의 산출물들의 "How to Use"에 대한 이해도를 높이기 위한 example code들이 실행 가능한 하나의 container application을 개발	key summary status Jira 프로젝트가 존재하지 않거나 볼 권한이 없습니다. Jira에서 이러한 이슈 보기[55]
2	Basic GUI Component Examples Development	▪ HMI App. 개발자들에게 Basic GUI Component들의 "How to Use"에 대한 이해도를 높이기 위한 example code들을 adi.examples.app에서 개발	key summary status Jira 프로젝트가 존재하지 않거나 볼 권한이 없습니다. Jira에서 이러한 이슈 보기[56]

55 https://jira.mobis.co.kr/secure/IssueNavigator.jspa?reset=true&jqlQuery=project+%3D+MADI+AND+%22Epic+Link%22+%3D+MADI-51+&src=confmacro

56 https://jira.mobis.co.kr/secure/IssueNavigator.jspa?reset=true&jqlQuery=project+%3D+MADI+AND+%22Epic+Link%22+%3D+MADI-52++&src=confmacro

8.5 8.5 Medium Term

8.5.1 8.5.1 Milestone

- TBD

8.6 8.6 Long Term

8.6.1 8.6.1 Milestone

- TBD

9 9. Reference

9.1 9.1 M.AOS

- M.AOS Home[57]

9.2 9.2 Android Tutorial

- Modern Android Development[58]
  - https://developer.android.com/jetpack
  - https://developer.android.com/courses/android-basics-compose/course
  - https://medium.com/androiddevelopers/tagged/mad-skills
- https://www.geeksforgeeks.org/android-tutorial/?ref=lbp

9.3 9.3 Application Framework

9.3.1 9.3.1 What is the Application Framework?

[Framework vs Library] Click here to expand...

Sorry, the widget is not supported in this export.
But you can reach it using the following URL:

http://youtube.com/watch?v=t9cclykXTCM

- 프레임워크 라이브러리 차이[59]

9.3.2 9.3.2 Examples of Application Framework

- GetX[60]

57 https://confluence.mobis.co.kr/display/MAOS/M.AOS+Home
58 https://developer.android.com/modern-android-development
59 https://digit2sight.com/%ED%94%84%EB%A0%88%EC%9E%84%EC%9B%8C%ED%81%AC-%EB%9D%BC%EC%9D%B4%EB%B8%8C%EB%9F%AC%EB%A6%AC-%EC%B0%A8%EC%9D%B4%EC%99%80-%EC%A2%85%EB%A5%98/
60 https://pub.dev/packages/get#about-get

9.3.3 9.3.3 Jetpack Library

- 개발자가 고품질 앱을 쉽고 빠르게 개발할 수 있게 돕는 라이브러리, 도구, 가이드 모음으로서 Jetpack은 Android 플랫폼 API와는 별도로 제공되는 androidx.*[61] 패키지 라이브러리로 구성되며, Google's Maven Repository[62] 에 존재
  The `androidx` namespace comprises the Android Jetpack[63] libraries. Like the Support Library, libraries in the `androidx` namespace ship separately from the Android platform and provide backward compatibility across Android releases.

특징	설명
앱 개발 생산성	• AOSP는 Android 운영 체제의 핵심 기능을 제공하는 것에 중점을 두는 반면, Jetpack library는 개발자들의 생산성을 높이기 위한 목적으로 디자인. • 즉, 개발자들이 앱을 보다 효율적으로 개발하고 관리할 수 있는 도구와 기능을 제공하는 것에 목표로 함.
모듈화 및 유지보수 용이성	• Jetpack library는 개별 모듈로 구성되어 있으며, 앱 개발자들이 필요한 기능만 선택하여 사용할 수 있도록 설계.
다양성과 일반성	• Android는 다양한 기기 및 하드웨어에서 작동해야 하기 때문에, AOSP는 가능한 한 일반적이고 범용적인 기능들을 제공하려는 경향. • 반면, Jetpack library리는 더 다양한 앱 개발 시나리오와 사용자 경험을 지원하기 위해 특정한 기능들을 제공.
업데이트 및 유지보수	• Jetpack library는 AOSP와는 독립적으로 업데이트 및 개선되기에, 더 빠르게 새로운 기능들의 사용 및 대응 가능.

	Library	Description
1	activity *[64]	Access composable APIs built on top of Activity.

61 https://developer.android.com/jetpack/androidx
62 https://maven.google.com/web/index.html
63 https://developer.android.com/jetpack
64 https://developer.android.com/jetpack/androidx/releases/activity

	Library	Description
2	ads[65]	Get an advertising ID with or without Play Services.
3	annotation[66]	Expose metadata that helps tools and other developers understand your app's code.
4	appcompat *[67]	Allows access to new APIs on older API versions of the platform (many using Material Design).
5	appsearch *[68]	Build custom in-app search capabilities for your users.
6	arch.core[69]	Helper for other arch dependencies, including JUnit test rules that can be used with LiveData.
7	asynclayoutinflater[70]	Inflate layouts asynchronously to avoid jank in the UI.
8	autofill[71]	Improve autofill accuracy via extending hints.
9	benchmark[72]	Accurately measure your code's performance within Android Studio.
10	biometric[73]	Authenticate with biometrics or device credentials, and perform cryptographic operations.
11	browser[74]	Display webpages in the user's default browser.
12	car-app[75]	Build navigation and point of interest apps for Android Auto and Android Automotive OS.

65 https://developer.android.com/jetpack/androidx/releases/ads
66 https://developer.android.com/jetpack/androidx/releases/annotation
67 https://developer.android.com/jetpack/androidx/releases/appcompat
68 https://developer.android.com/jetpack/androidx/releases/appsearch
69 https://developer.android.com/jetpack/androidx/releases/arch-core
70 https://developer.android.com/jetpack/androidx/releases/asynclayoutinflater
71 https://developer.android.com/jetpack/androidx/releases/autofill
72 https://developer.android.com/jetpack/androidx/releases/benchmark
73 https://developer.android.com/jetpack/androidx/releases/biometric
74 https://developer.android.com/jetpack/androidx/releases/browser
75 https://developer.android.com/jetpack/androidx/releases/car-app

	Library	Description
13	camera *[76]	Build mobile camera apps.
14	cardview[77]	Implement the Material Design card pattern with round corners and drop shadows.
15	collection[78]	Reduce the memory impact of existing and new collections that are small.
16	compose *[79]	Define your UI programmatically with composable functions that describe its shape and data dependencies.
17	compose.animation[80]	Build animations in their Jetpack Compose applications to enrich the user experience.
18	compose.compiler[81]	Transform @Composable functions and enable optimizations with a Kotlin compiler plugin.
19	compose.foundation[82]	Write Jetpack Compose applications with ready to use building blocks and extend foundation to build your own design system pieces.
20	compose.material[83]	Build Jetpack Compose UIs with ready to use Material Design Components. This is the higher level entry point of Compose, designed to provide components that match those described at www.material.io[84].
21	compose.material3[85]	Build Jetpack Compose UIs with Material Design 3 Components, the next evolution of Material Design. Material 3 includes updated theming and components and Material You personalization features like dynamic color, and is designed to be cohesive with the new Android 12 visual style and system UI.

76 https://developer.android.com/jetpack/androidx/releases/camera
77 https://developer.android.com/jetpack/androidx/releases/cardview
78 https://developer.android.com/jetpack/androidx/releases/collection
79 https://developer.android.com/jetpack/androidx/releases/compose
80 https://developer.android.com/jetpack/androidx/releases/compose-animation
81 https://developer.android.com/jetpack/androidx/releases/compose-compiler
82 https://developer.android.com/jetpack/androidx/releases/compose-foundation
83 https://developer.android.com/jetpack/androidx/releases/compose-material
84 http://www.material.io
85 https://developer.android.com/jetpack/androidx/releases/compose-material3

	Library	Description
22	compose.runtime[86]	Fundamental building blocks of Compose's programming model and state management, and core runtime for the Compose Compiler Plugin to target.
23	compose.ui[87]	Fundamental components of compose UI needed to interact with the device, including layout, drawing, and input.
24	concurrent[88]	Move tasks off the main thread with coroutines and take advantage of ListenableFuture.
25	constraintlayout[89]	Position and size widgets in a flexible way with relative positioning.
26	contentpager[90]	Load and page across ContentProvider data in a background thread.
27	coordinatorlayout[91]	Position top-level application widgets, such as AppBarLayout and FloatingActionButton.
28	core[92]	Target the latest platform features and APIs while also supporting older devices.
29	core.uwb[93]	Implement UWB (ultra-wideband) on supported devices.
30	credentials[94]	This library provides unified access to a user's credentials. This can include passwords, passkeys and federated credentials. This library should be used to provide seamless and secure sign-in experiences.
31	cursoradapter[95]	Expose Cursor data to a ListView widget.

86 https://developer.android.com/jetpack/androidx/releases/compose-runtime
87 https://developer.android.com/jetpack/androidx/releases/compose-ui
88 https://developer.android.com/jetpack/androidx/releases/concurrent
89 https://developer.android.com/jetpack/androidx/releases/constraintlayout
90 https://developer.android.com/jetpack/androidx/releases/contentpager
91 https://developer.android.com/jetpack/androidx/releases/coordinatorlayout
92 https://developer.android.com/jetpack/androidx/releases/core
93 https://developer.android.com/jetpack/androidx/releases/core-uwb
94 https://developer.android.com/jetpack/androidx/releases/credentials
95 https://developer.android.com/jetpack/androidx/releases/cursoradapter

	Library	Description
32	customview[96]	Implement custom views.
33	databinding *[97]	Bind UI components in your layouts to data sources in your app using a declarative format.
34	datastore[98]	Store data asynchronously, consistently, and transactionally, overcoming some of the drawbacks of SharedPreferences
35	documentfile[99]	View a file document.
36	draganddrop[100]	Accept drag-and-drop data from another app or within an app, and show a consistent drop target affordance.
37	drawerlayout[101]	Implement a Material Design drawer widget.
38	dynamicanimation[102]	Create smooth animations with a physics-based animation API.
39	emoji[103]	Display emoji in current and older devices.
40	emoji2[104]	Display emoji in current and older devices.
41	enterprise[105]	Create enterprise-ready applications.
42	exifinterface[106]	Read and write image file EXIF tags.

96 https://developer.android.com/jetpack/androidx/releases/customview
97 https://developer.android.com/jetpack/androidx/releases/databinding
98 https://developer.android.com/jetpack/androidx/releases/datastore
99 https://developer.android.com/jetpack/androidx/releases/documentfile
100 https://developer.android.com/jetpack/androidx/releases/draganddrop
101 https://developer.android.com/jetpack/androidx/releases/drawerlayout
102 https://developer.android.com/jetpack/androidx/releases/dynamicanimation
103 https://developer.android.com/jetpack/androidx/releases/emoji
104 https://developer.android.com/jetpack/androidx/releases/emoji2
105 https://developer.android.com/jetpack/androidx/releases/enterprise
106 https://developer.android.com/jetpack/androidx/releases/exifinterface

	Library	Description
43	fragment *[107]	Segment your app into multiple, independent screens that are hosted within an Activity.
44	games[108]	Use the Android Game SDK natively in your app to perform complex games tasks, like Frame Pacing.
45	glance[109]	Build layouts for remote surfaces using a Jetpack Compose-style API.
46	graphics[110]	Leverage graphics facilities across multiple Android platform releases
47	gridlayout[111]	Implement a grid layout.
48	health[112]	Create performant health applications in a platform agnostic way.
49	health.connect[113]	Allows developers to read or write user's health and fitness records.
50	heifwriter[114]	Encode an image or image collection in HEIF format using the available codecs on the Android device.
51	hilt *[115]	Extend the functionality of Dagger Hilt to enable dependency injection of certain classes from the androidx libraries.
52	input[116]	Reduce the latency of input interactions by predicting future MotionEvents
53	interpolator[117]	Use animation interpolators on older platforms.

107 https://developer.android.com/jetpack/androidx/releases/fragment
108 https://developer.android.com/jetpack/androidx/releases/games
109 https://developer.android.com/jetpack/androidx/releases/glance
110 https://developer.android.com/jetpack/androidx/releases/graphics
111 https://developer.android.com/jetpack/androidx/releases/gridlayout
112 https://developer.android.com/jetpack/androidx/releases/health
113 https://developer.android.com/jetpack/androidx/releases/health-connect
114 https://developer.android.com/jetpack/androidx/releases/heifwriter
115 https://developer.android.com/jetpack/androidx/releases/hilt
116 https://developer.android.com/jetpack/androidx/releases/input
117 https://developer.android.com/jetpack/androidx/releases/interpolator

	Library	Description
54	javascriptengine[118]	Enable your Android app to evaluate JavaScript.
55	jetifier[119]	A standalone tool that migrates a library's dependencies on the deprecated support library to equivalent AndroidX dependencies.
56	leanback[120]	Write apps for Android TV devices using dpad-friendly widgets and template fragments.
57	legacy[121]	This artifact and its classes are deprecated. Starting with Android 8, background check restrictions make this class no longer useful.
58	lifecycle *[122]	Build lifecycle-aware components that can adjust behavior based on the current lifecycle state of an activity or fragment.
59	loader[123]	Load data for your UI that survives configuration changes.
60	localbroadcastmanager[124]	This artifact and its classes are deprecated. Use LiveData or reactive streams instead.
61	media[125]	Share media contents and controls with other apps. Superseded by media2.
62	media2[126]	Share media contents and controls with other apps.
63	media3 *[127]	Support libraries for media use cases.

118 https://developer.android.com/jetpack/androidx/releases/javascriptengine
119 https://developer.android.com/jetpack/androidx/releases/jetifier
120 https://developer.android.com/jetpack/androidx/releases/leanback
121 https://developer.android.com/jetpack/androidx/releases/legacy
122 https://developer.android.com/jetpack/androidx/releases/lifecycle
123 https://developer.android.com/jetpack/androidx/releases/loader
124 https://developer.android.com/jetpack/androidx/releases/localbroadcastmanager
125 https://developer.android.com/jetpack/androidx/releases/media
126 https://developer.android.com/jetpack/androidx/releases/media2
127 https://developer.android.com/jetpack/androidx/releases/media3

	Library	Description
64	mediarouter[128]	Enable media display and playback on remote receiver devices using a common user interface.
65	multidex[129]	Deploy applications with multiple dex files on pre-Android 5 devices.
66	metrics[130]	Track and report various runtime metrics for your application
67	navigation *[131]	Build and structure your in-app UI, handle deep links, and navigate between screens.
68	paging *[132]	Load data in pages, and present it in a RecyclerView.
69	palette[133]	Extract representative color palettes from images.
70	percentlayout[134]	This artifact and its classes are deprecated. Use ConstraintLayout and associated layouts instead.
71	preference[135]	Build interactive settings screens without needing to interact with device storage or manage the UI.
72	print[136]	Print photos, docs, and other graphics and images from your app.
73	privacysandbox.ads[137]	This library enables integration with Privacy Preserving APIs, which are part of Privacy Sandbox on Android.
74	privacysandbox.plugins[138]	Android Privacy Sandbox Sdk Library Gradle Plugin

128 https://developer.android.com/jetpack/androidx/releases/mediarouter
129 https://developer.android.com/jetpack/androidx/releases/multidex
130 https://developer.android.com/jetpack/androidx/releases/metrics
131 https://developer.android.com/jetpack/androidx/releases/navigation
132 https://developer.android.com/jetpack/androidx/releases/paging
133 https://developer.android.com/jetpack/androidx/releases/palette
134 https://developer.android.com/jetpack/androidx/releases/percentlayout
135 https://developer.android.com/jetpack/androidx/releases/preference
136 https://developer.android.com/jetpack/androidx/releases/print
137 https://developer.android.com/jetpack/androidx/releases/privacysandbox-ads
138 https://developer.android.com/jetpack/androidx/releases/privacysandbox-plugins

	Library	Description
75	privacysandbox.sdkruntime[139]	This library provides components for SdkRuntime aware consumers
76	privacysandbox.tools[140]	A library to utilize the Privacy Sandbox functionality in Android
77	privacysandbox.ui[141]	TODO
78	profileinstaller[142]	Enables libraries to prepopulate ahead of time compilation traces to be read by ART.
79	recommendation[143]	Promote content to the Android TV Launcher home screen.
80	recyclerview[144]	Display large sets of data in your UI while minimizing memory usage.
81	remotecallback[145]	Create a wrapper that makes it easier for developers to provide a PendingIntent.
82	resourceinspection[146]	Surface the attributes of custom views in Android Studio's Live Layout Inspector.
83	room *[147]	Create, store, and manage persistent data backed by a SQLite database.
84	savedstate[148]	Write pluggable components that save the UI state when a process dies, and restore it when the process restarts.

139 https://developer.android.com/jetpack/androidx/releases/privacysandbox-sdkruntime
140 https://developer.android.com/jetpack/androidx/releases/privacysandbox-tools
141 https://developer.android.com/jetpack/androidx/releases/privacysandbox-ui
142 https://developer.android.com/jetpack/androidx/releases/profileinstaller
143 https://developer.android.com/jetpack/androidx/releases/recommendation
144 https://developer.android.com/jetpack/androidx/releases/recyclerview
145 https://developer.android.com/jetpack/androidx/releases/remotecallback
146 https://developer.android.com/jetpack/androidx/releases/resourceinspection
147 https://developer.android.com/jetpack/androidx/releases/room
148 https://developer.android.com/jetpack/androidx/releases/savedstate

	Library	Description
85	security[149]	Safely manage keys and encrypt files and sharedpreferences.
86	sharetarget[150]	Provide backwards compatibility for using shortcuts as direct share targets.
87	slice[151]	Display templated UI elements outside your app.
88	slidingpanelayout[152]	Implement a sliding pane UI pattern.
89	startup[153]	Implement a straightforward, performant way to initialize components at app startup.
90	sqlite[154]	Work with local SQLite databases. If possible, use Room instead.
91	swiperefreshlayout[155]	Implement the swipe-to-refresh UI pattern.
92	test *[156]	Testing in Android.
93	test.uiautomator[157]	Framework for cross app functional UI testing
94	textclassifier[158]	Identifies conversations, links, selections, and other similar constructs in text.
95	tracing[159]	Write trace events to the system trace buffer.
96	transition[160]	Animate motion in the UI with starting and ending layouts.

149 https://developer.android.com/jetpack/androidx/releases/security
150 https://developer.android.com/jetpack/androidx/releases/sharetarget
151 https://developer.android.com/jetpack/androidx/releases/slice
152 https://developer.android.com/jetpack/androidx/releases/slidingpanelayout
153 https://developer.android.com/jetpack/androidx/releases/startup
154 https://developer.android.com/jetpack/androidx/releases/sqlite
155 https://developer.android.com/jetpack/androidx/releases/swiperefreshlayout
156 https://developer.android.com/jetpack/androidx/releases/test
157 https://developer.android.com/jetpack/androidx/releases/test-uiautomator
158 https://developer.android.com/jetpack/androidx/releases/textclassifier
159 https://developer.android.com/jetpack/androidx/releases/tracing
160 https://developer.android.com/jetpack/androidx/releases/transition

	Library	Description
97	tv[161]	Provides developers with Compose and Material design functionalities in order to write applications for TV
98	tvprovider[162]	Provide Android TV channels.
99	vectordrawable[163]	Render vector graphics.
100	versionedparcelable[164]	Provides a stable and compact binary serialization format that can be passed across processes or persisted safely.
101	viewpager[165]	Display Views or Fragments in a swipeable format. If possible, use viewpager2 instead.
102	viewpager2[166]	Display Views or Fragments in a swipeable format.
103	wear[167]	Create applications for Wear OS by Google smartwatches.
104	wear.compose[168]	Write Jetpack Compose applications for Wearable devices by providing functionality to support wearable specific devices, sizes, shapes and navigation gestures.
105	wear.protolayout[169]	This library allows defining a set of UI layouts and non-UI expressions to be rendered/evaluated on a remote surfaces.
106	wear.tiles[170]	Create applications for Wear OS by Google smartwatches.

161 https://developer.android.com/jetpack/androidx/releases/tv
162 https://developer.android.com/jetpack/androidx/releases/tvprovider
163 https://developer.android.com/jetpack/androidx/releases/vectordrawable
164 https://developer.android.com/jetpack/androidx/releases/versionedparcelable
165 https://developer.android.com/jetpack/androidx/releases/viewpager
166 https://developer.android.com/jetpack/androidx/releases/viewpager2
167 https://developer.android.com/jetpack/androidx/releases/wear
168 https://developer.android.com/jetpack/androidx/releases/wear-compose
169 https://developer.android.com/jetpack/androidx/releases/wear-protolayout
170 https://developer.android.com/jetpack/androidx/releases/wear-tiles

	Library	Description
107	wear.watchface[171]	Create applications for Wear OS by Google smartwatches.
108	webkit[172]	Work with modern WebView APIs on Android 5 and above.
109	window[173]	Helps support different device form factors such as foldable devices.
110	window.extensions.core[174]	The Core APIs for Window Manager Library Extensions
111	work *[175]	Schedule and execute deferrable, constraint-based background tasks.
112	Material Design Components *[176]	Modular and customizable Material Design UI components for Android.

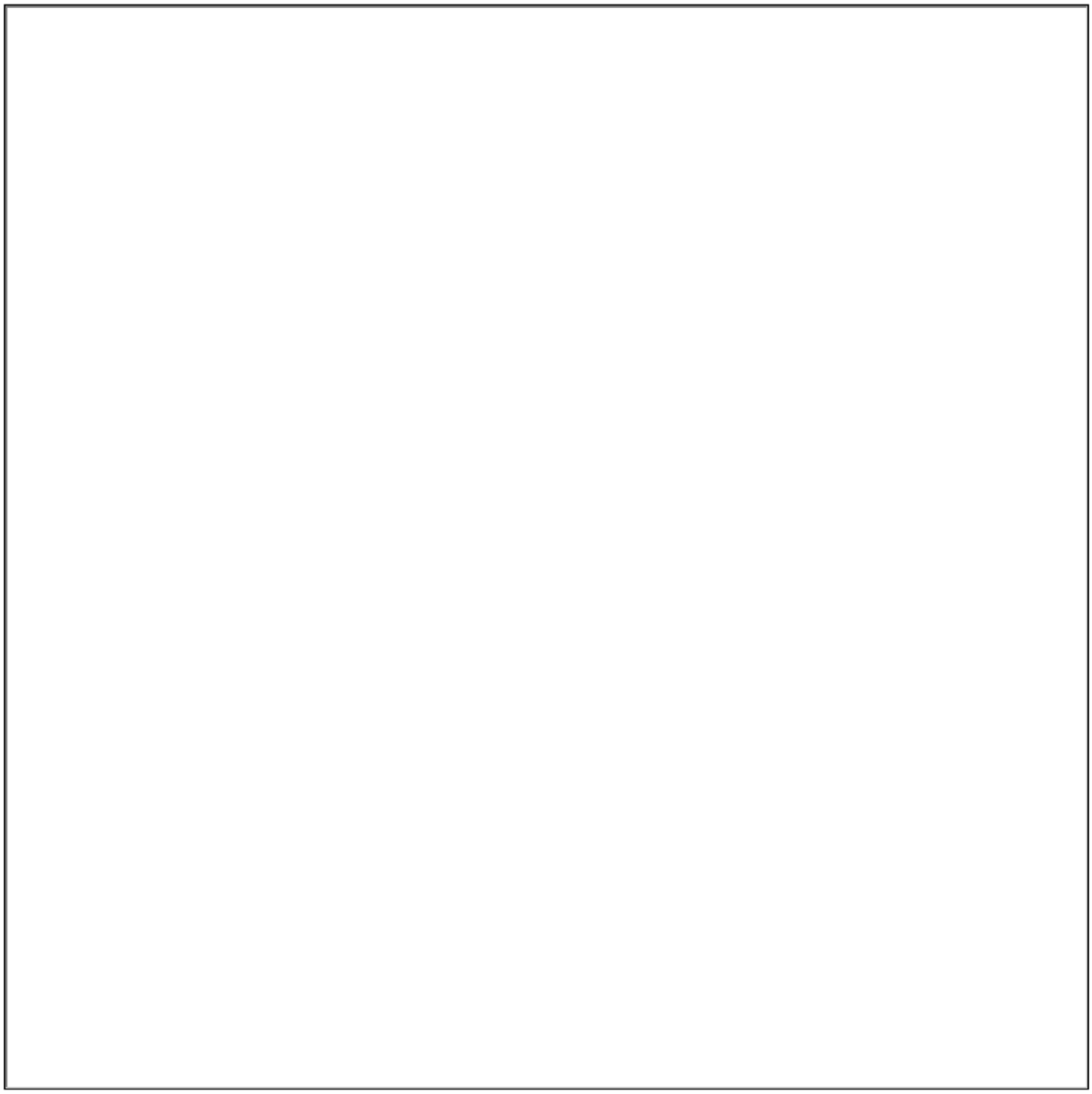
- Guide Link
  - https://developer.android.com/jetpack
  - Jetpack & architecture document[177]

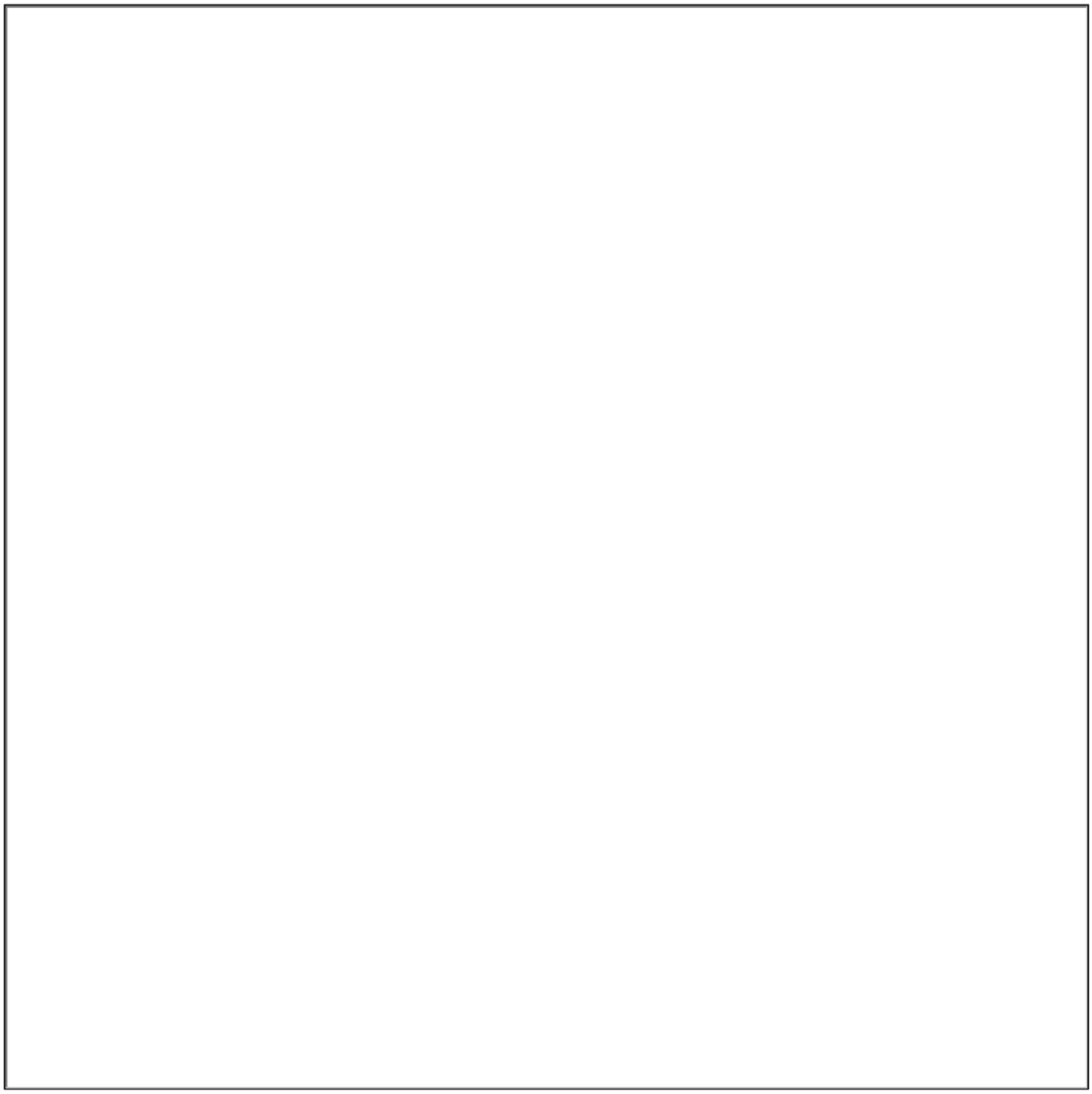
9.3.4 9.3.4 Architecture Design Examples with Jetpack Compose

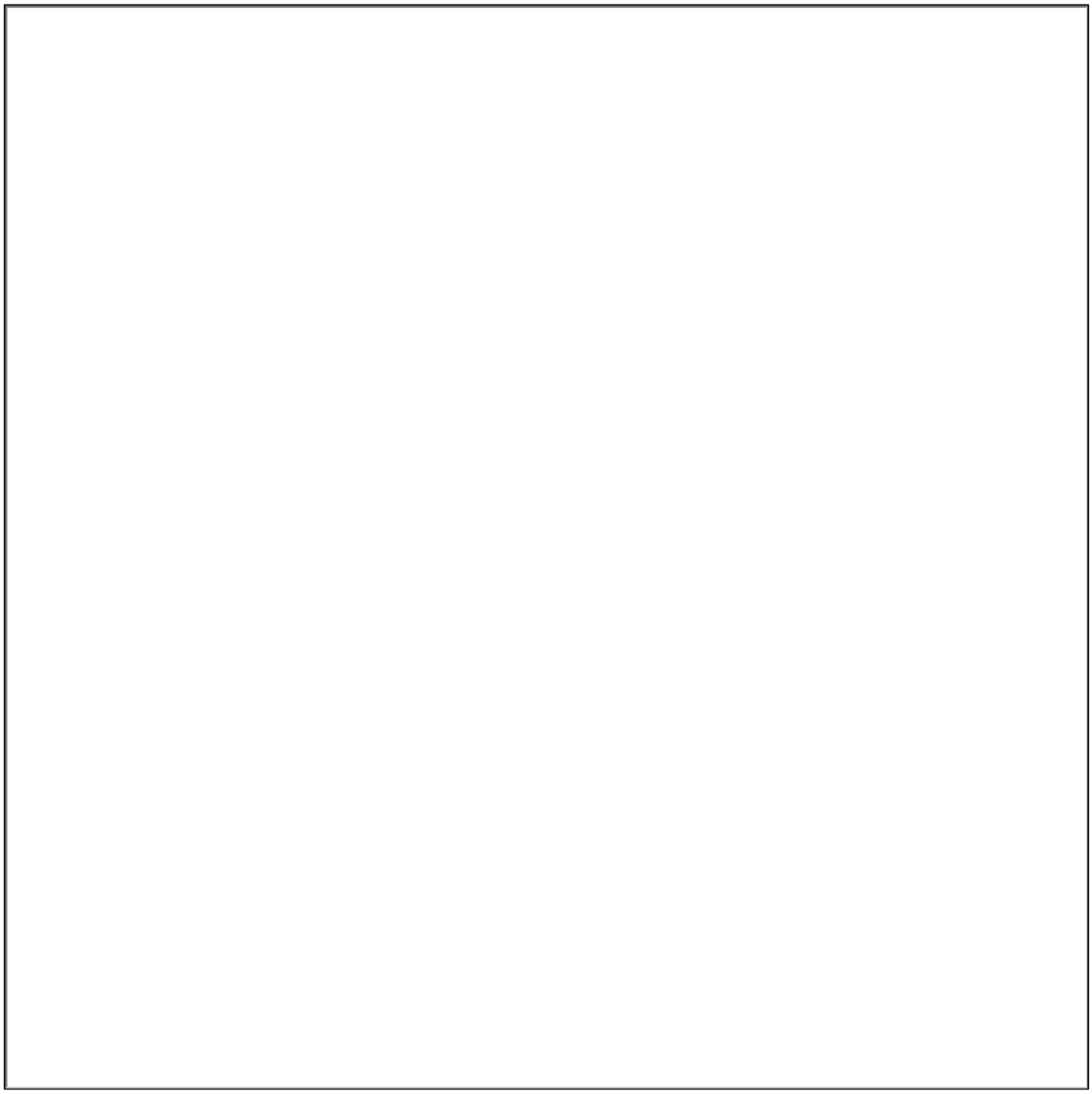
- Example app: Designed and built in the design case study, architecture learning journey and modularization learning journey[178]
  [Architecture Learning Journey] Click here to expand...

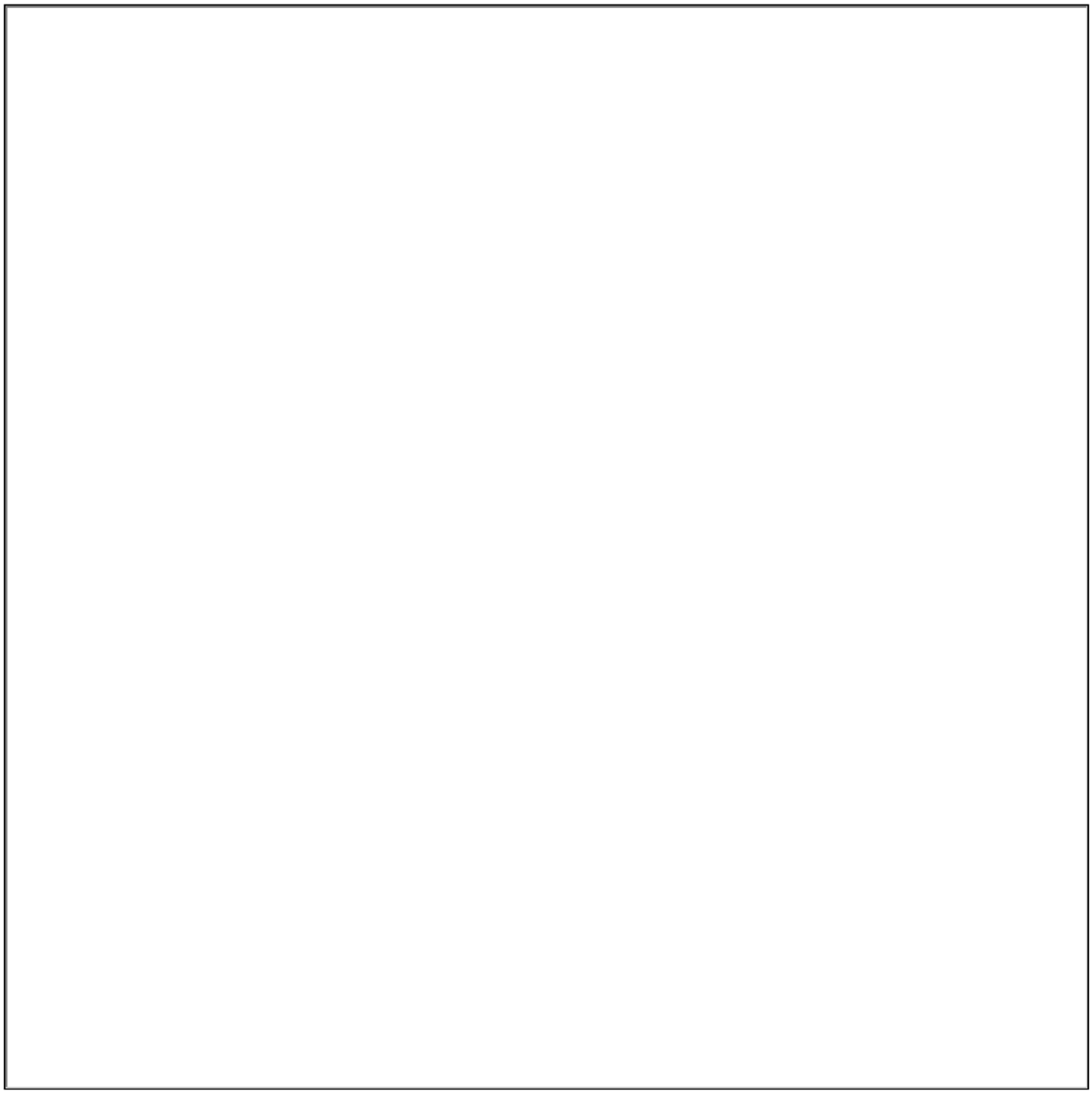
  https://github.com/android/nowinandroid/blob/main/docs/ArchitectureLearningJourney.md

171 https://developer.android.com/jetpack/androidx/releases/wear-watchface
172 https://developer.android.com/jetpack/androidx/releases/webkit
173 https://developer.android.com/jetpack/androidx/releases/window
174 https://developer.android.com/jetpack/androidx/releases/window-extensions-core
175 https://developer.android.com/jetpack/androidx/releases/work
176 https://m3.material.io/develop/android/jetpack-compose
177 https://www.geeksforgeeks.org/introduction-to-android-jetpack/?ref=lbp
178 https://github.com/android/nowinandroid



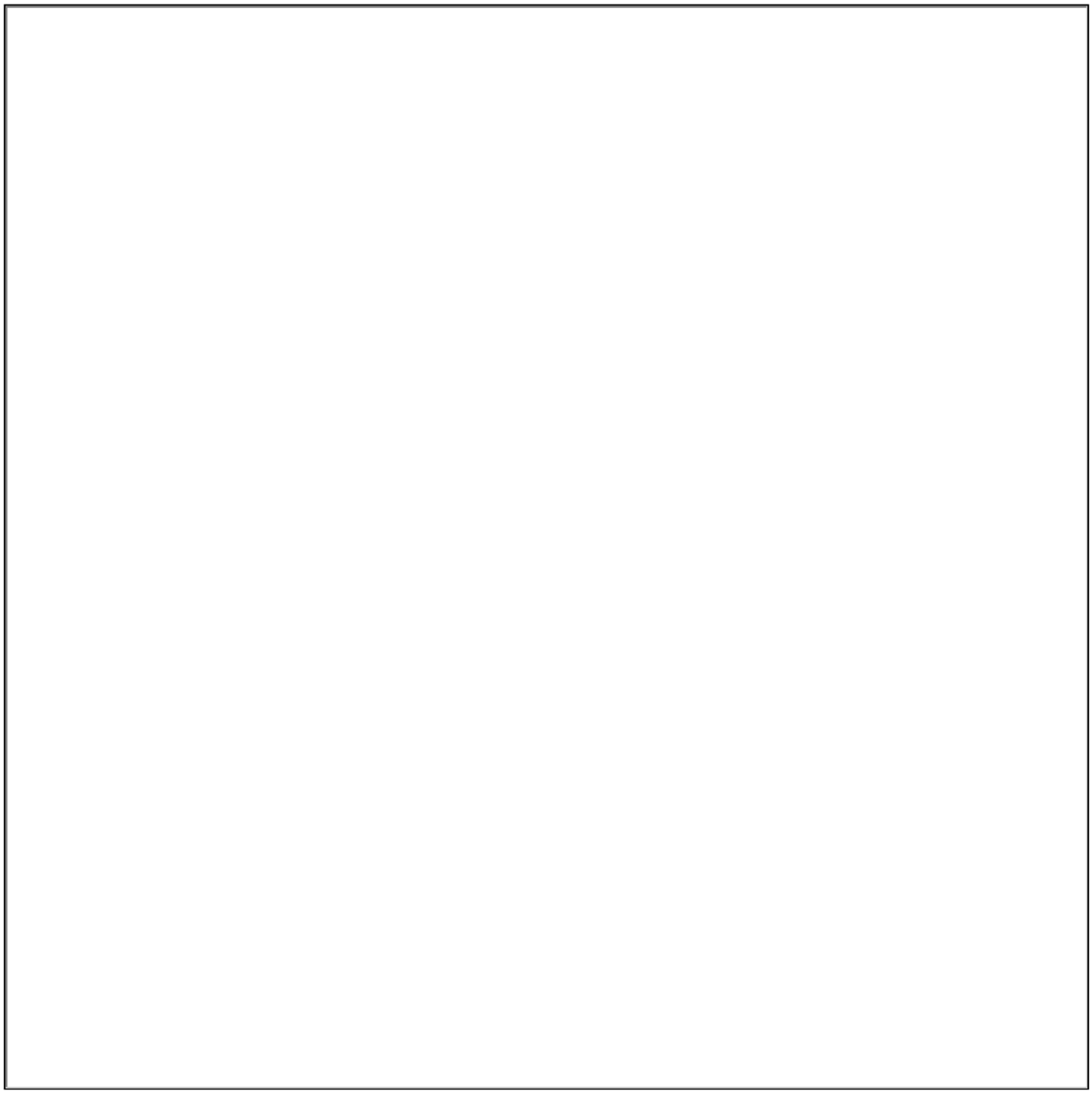






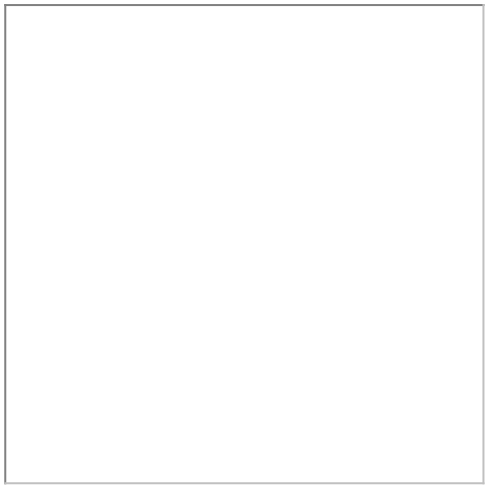
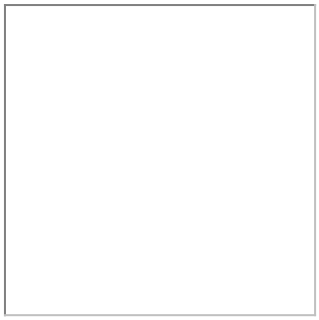
[Modularization learning journey] Click here to expand...

https://github.com/android/nowinandroid/blob/main/docs/ModularizationLearningJourney.md

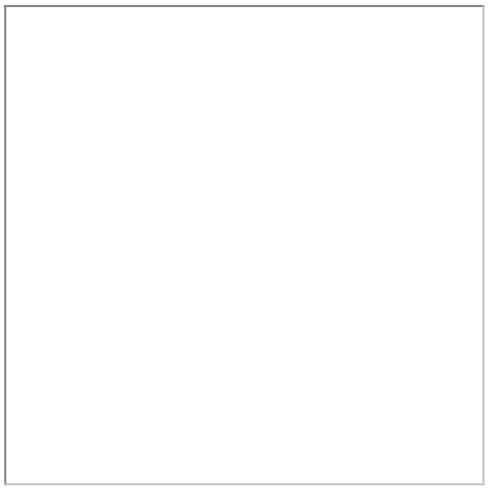
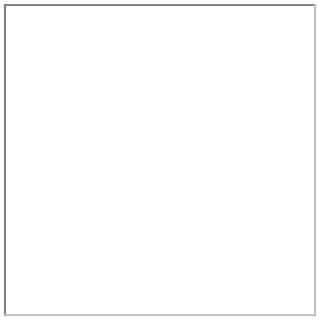


- Jetpack Compose Samples[179] (https://developer.android.com/jetpack/compose/samples)

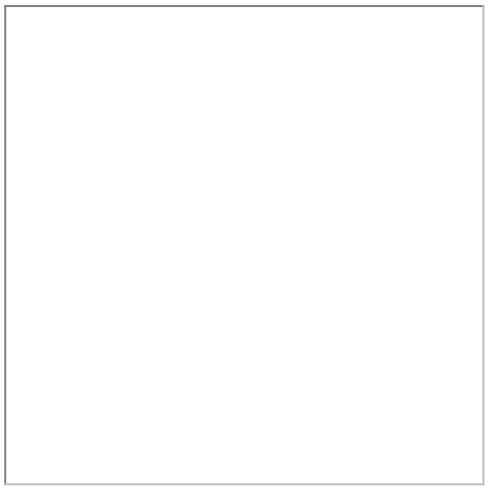
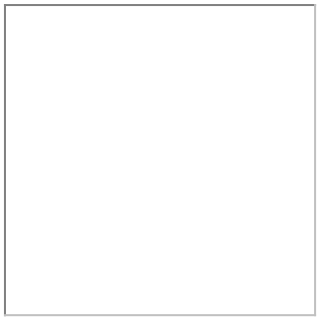
179 https://github.com/android/compose-samples

Project	
 A sample blog post viewer that demonstrates the use of Compose with a typical Material app and real-world architecture. • Medium complexity • Varied UI • Light & dark themes • Resource loading • UI Testing > Browse[180]	

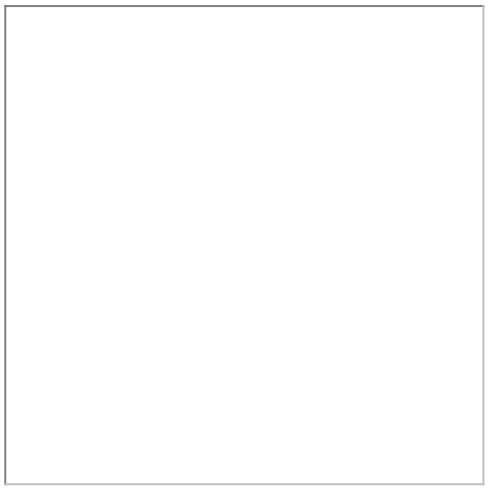
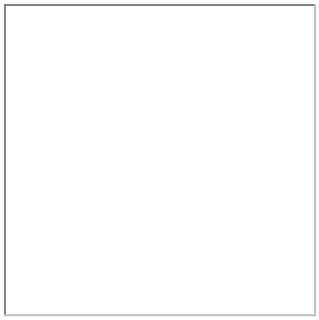
180 https://github.com/android/compose-samples/blob/main/JetNews

Project	
 A sample chat app that focuses on UI state patterns and text input. • Low complexity • Material Design 3 theme and Material You dynamic color • Resource loading • Back button handling • Integration with Architecture Components: Navigation, Fragments, LiveData, ViewModel • Animation • UI Testing > Browse[181]	

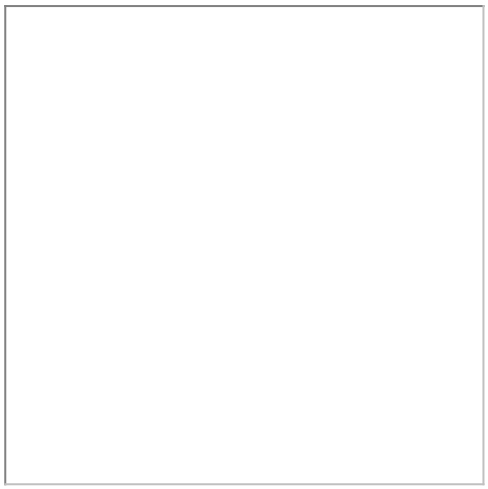
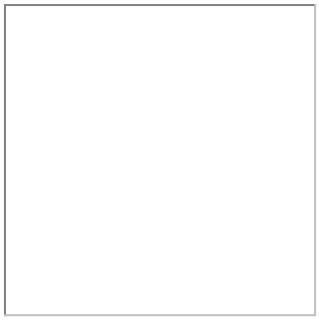
181 https://github.com/android/compose-samples/blob/main/Jetchat

Project	
 A sample survey app that showcases text input, validation and UI state management in Compose. • Low complexity • `TextField` and form validation • Snackbar implementation • Element reusability and styling • Various form elements > Browse[182]	

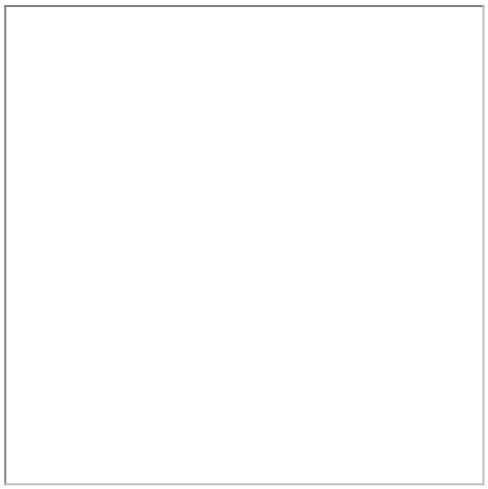
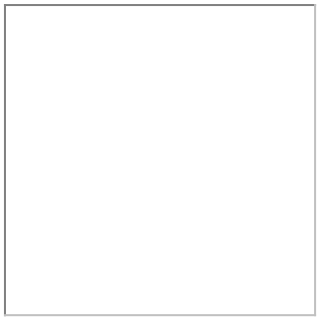
182 https://github.com/android/compose-samples/blob/main/Jetsurvey

Project	
 Jetsnack is a sample snack ordering app built with Compose. • Medium complexity • Custom design system • Custom layouts • Animation > Browse[183]	

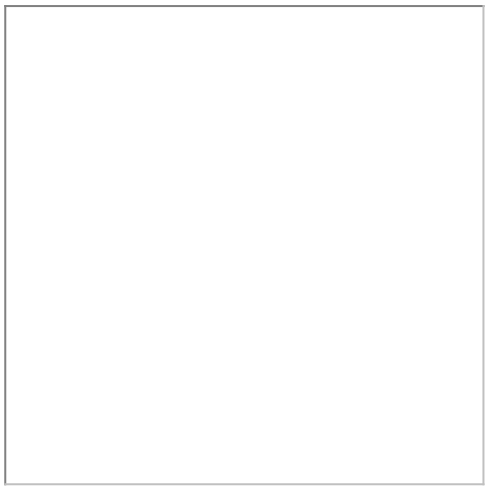
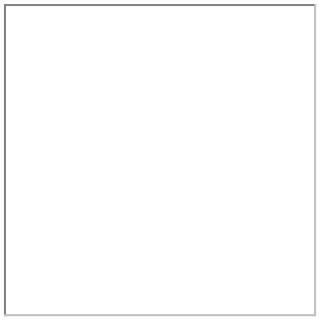
183 https://github.com/android/compose-samples/blob/main/Jetsnack

Project	
 A sample podcast app that features a full-featured, Redux-style architecture and showcases dynamic themes. • Advanced sample • Dynamic theming using podcast artwork • Image fetching • `WindowInsets`[184] support • Coroutines • Local storage with Room > Browse[185]	

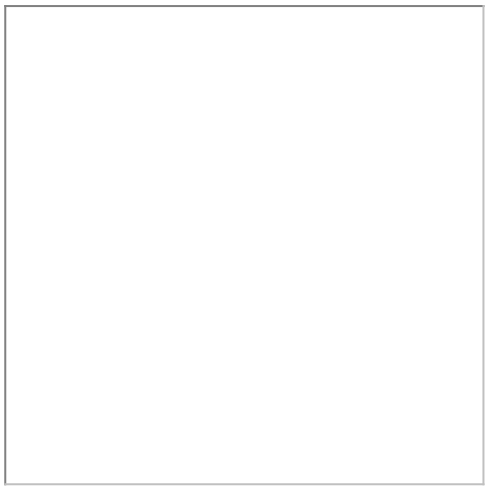
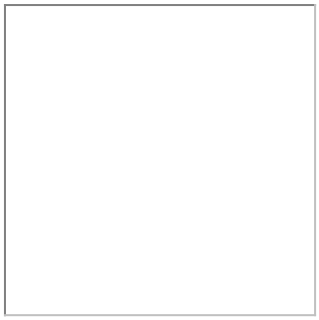
184 https://developer.android.com/reference/kotlin/android/view/WindowInsets
185 https://github.com/android/compose-samples/blob/main/Jetcaster

Project	
 A Compose implementation of the Rally Material study, a financial app that focuses on data, charts, reusability and animations. • Low complexity • Material theming with a dark-only theme • Custom layouts and reusable elements • Charts and tables • Animations • Screenshot tests > Browse[186]	

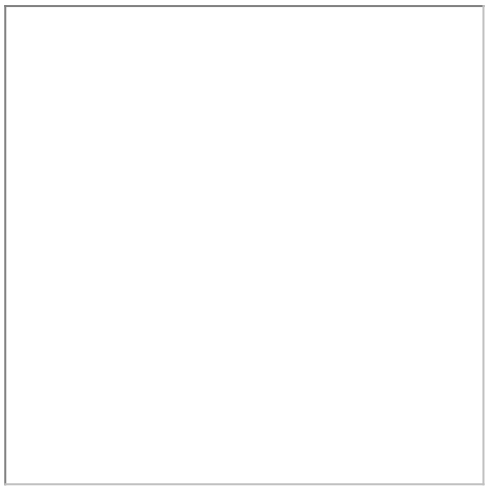
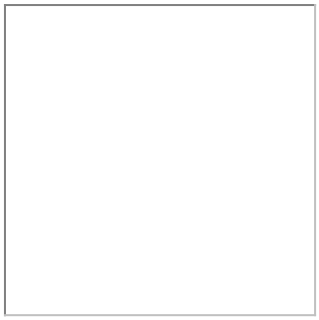
186 https://github.com/android/compose-samples/blob/main/Rally

Project	
 A Compose implementation of the Crane Material study, a travel app that uses Material Design components and Material Theming to create a personalized, on-brand experience. • Medium complexity • Draggable UI elements • UI state handling • UI Tests > Browse[187]	

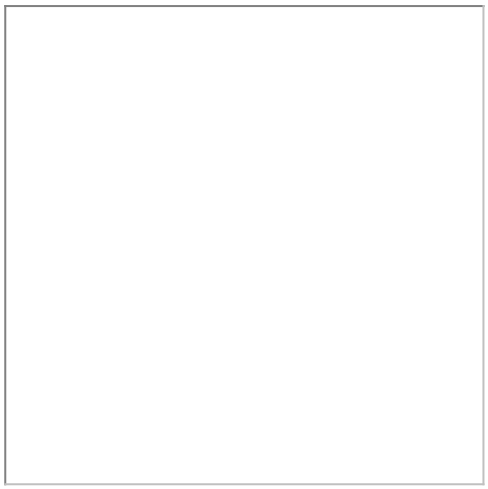
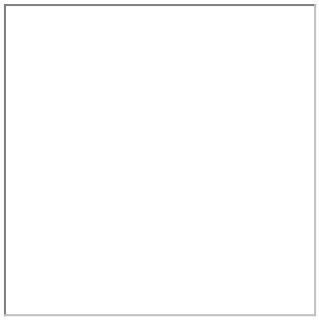
187 https://github.com/android/compose-samples/blob/main/Crane

Project	
 A Compose implementation of the Owl Material study. The Owl brand uses bold color, shape, and typography to express its brand attributes: energy, daring, and fun. • Medium complexity • Material theming & light/dark themes • Custom layout • Animation > Browse[188]	

188 https://github.com/android/compose-samples/blob/main/Owl

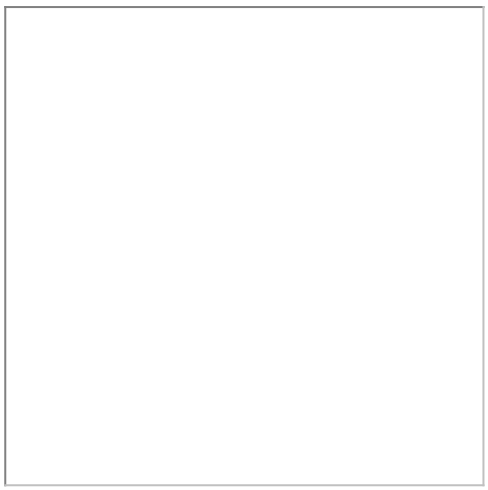
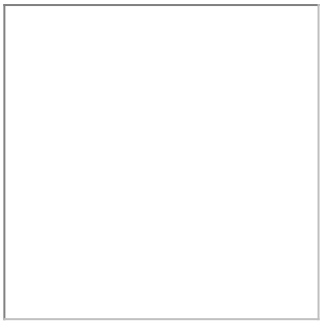
Project	
 A compose implementation of the Reply material study, an email client app that focuses on adaptive design for mobile, tablets and foldables. It also showcases brand new Material design 3 theming, dynamic colors and navigation components. • Medium complexity • Adaptive UI for phones, tablet and desktops • Foldable support • Material 3 theming & Components • Dynamic colors and Light/Dark theme support > Browse[189]	

189 https://github.com/android/compose-samples/blob/main/Reply

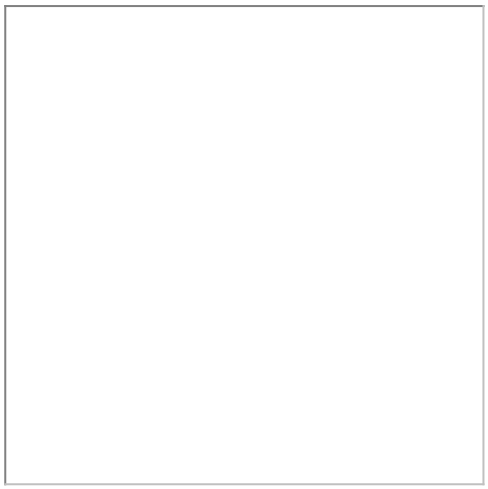
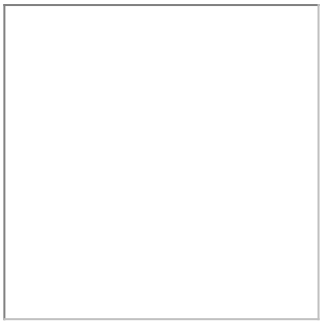
Project	
 A sample sleep tracker app, showcasing how to create custom layouts and graphics in Compose • Custom Layouts • Graphs with Paths > Browse[190]	

☐ Additional samples

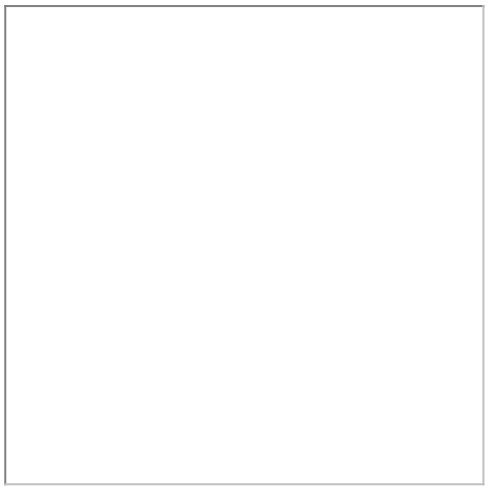
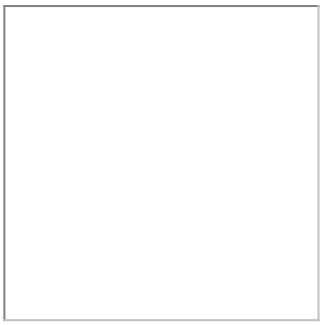
190 https://github.com/android/compose-samples/blob/main/JetLagged

Project	
 An app for keeping up to date with the latest news and developments in Android. • Jetpack Compose[191] first app. • Implements the recommended Android Architecture Guidelines[192] • Integrates Jetpack Libraries[193] holistically in the context of a real world app  > Browse[194]	

191 https://developer.android.com/jetpack/compose
192 https://developer.android.com/topic/architecture
193 https://developer.android.com/jetpack
194 https://github.com/android/nowinandroid

Project	
 A catalog of Material Design components and features available in Jetpack Compose. See how to implement them and how they look and behave on real devices. • Lives in AOSP—always up to date • Uses the same samples as API reference docs • Theme picker to change Material Theming values at runtime • Links to guidelines, docs, source code, and issue tracker  > Browse on AOSP[195]	

195 https://cs.android.com/androidx/platform/frameworks/support/+/androidx-main:compose/integration-tests/material-catalog

Project	
 A gardening app illustrating Android development best practices with Android Jetpack. The `compose` branch is partially built with Jetpack Compose. • Compose working in an existing app: including integration with strings, resources, and themes and styles • Integration with an existing architecture based on Jetpack libraries • `CollapsingToolbarLayout` behavior implemented manually with Compose • Showing Snackbars[196] with Compose > Browse[197]	

- Plugin : Jetpack Compose UI Architecture Templates[198]

9.3.5 9.3.5 Architecture Design Examples on GitHub

- https://github.com/android/architecture-samples
- https://github.com/android/architecture-components-samples

196 https://material.io/components/snackbars
197 https://goo.gle/sunflower-compose
198 https://plugins.jetbrains.com/plugin/19034-jetpack-compose-ui-architecture-templates

9.4 9.4 Code Template

9.4.1 9.4.1 How to Create the Custom Template

- Custom Template Plugin for Android Studio[199]
- Accommodate your project-specific needs with custom templates[200]
- Annotation Processor 만들기[201]

9.4.2 9.4.2 Inspection

- MAD Scorecard[202]

9.5 9.5 Design System

- Atomic Design[203]
- **[Design System with Jetapck Compose] Click here to expand...**

Sorry, the widget is not supported in this export.
But you can reach it using the following URL:

https://www.youtube.com/watch?v=O1yPOr7J3gY

199 https://velog.io/@devoks/Custom-Template-Plugin-for-Android-Studio
200 https://proandroiddev.com/accommodate-your-project-specific-needs-with-custom-templates-46cfdccc8363
201 https://www.charlezz.com/?p=1167
202 https://plugins.jetbrains.com/plugin/15549-mad-scorecard
203 https://atomicdesign.bradfrost.com/chapter-2/