

⇒ Spring boot Batch processing (For reading, Processing and writing huge amount of data batch by batch)

⇒ Spring boot Security (For Enabling Security on the web Apps,
Distributed Apps)

1 - -> Authentication & Authorization

I -> checking the access^x of a user
Permission

1 → Checking the identity of a User.

Given for Advanced Security Operations
like SSO Impl

⇒ Spring boot JWT (JSON web Tokens) like SSO Impl
↓
(Single Sign On)

→ Spring Boot Cloud (For developing Micro Services Architecture Apps)

⇒ Spring boot JMS

⇒ Springboot apache Kafka

→ For Enabling Text based Comm

below the apps or projects

⇒ Spring Boot Integration with Angular | React JS

⇒ Spring boot webflux C for reactive programming

⇒ Mini Projects (4-5)

⇒ Design Patterns (Best practices of Coding)

⇒ Spring boot Integration with API

and etc..

lot tools (maven, junit, gradle, log4j, slf4j and etc..)

lot common topics (resume preparation, Interview topics)

dos and Do nots after Joining the Company,

how to gap as experience and etc..)

— x — x — x — x — x — x —

⇒ To read data from different sources, to process data

and write data to different destinations batch by

batch we need to work with Spring boot batch

module.

reading LIC Policy holders Info from DB Sql (Oracle)
batch by batch.

UseCase ::

Identifying matured policies and config the

⇒ SSOC (Single Sign On) Feature makes the developers to sign in to one or another social media app and allows to use that Identity to operate other Apps.

Eg:: login to gmail and use that Identity to operate other google products

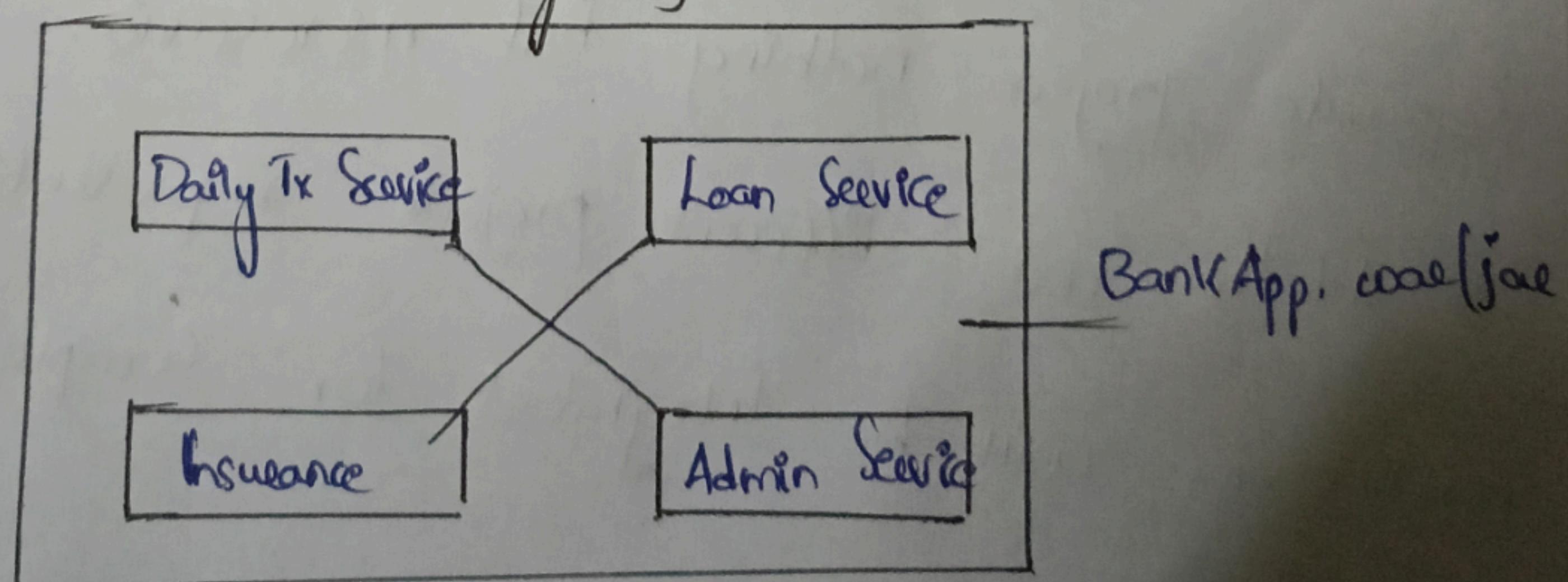
⇒ Two popular architecture for project development are.

a) Monolith Architecture

b) Microservice Architecture.

⇒ In Monolith (Monolith all the services / modules of the project will be packed into single unit as jar file / war file as shown below)

Banking Project



→ Few limitations of Monolith Architecture

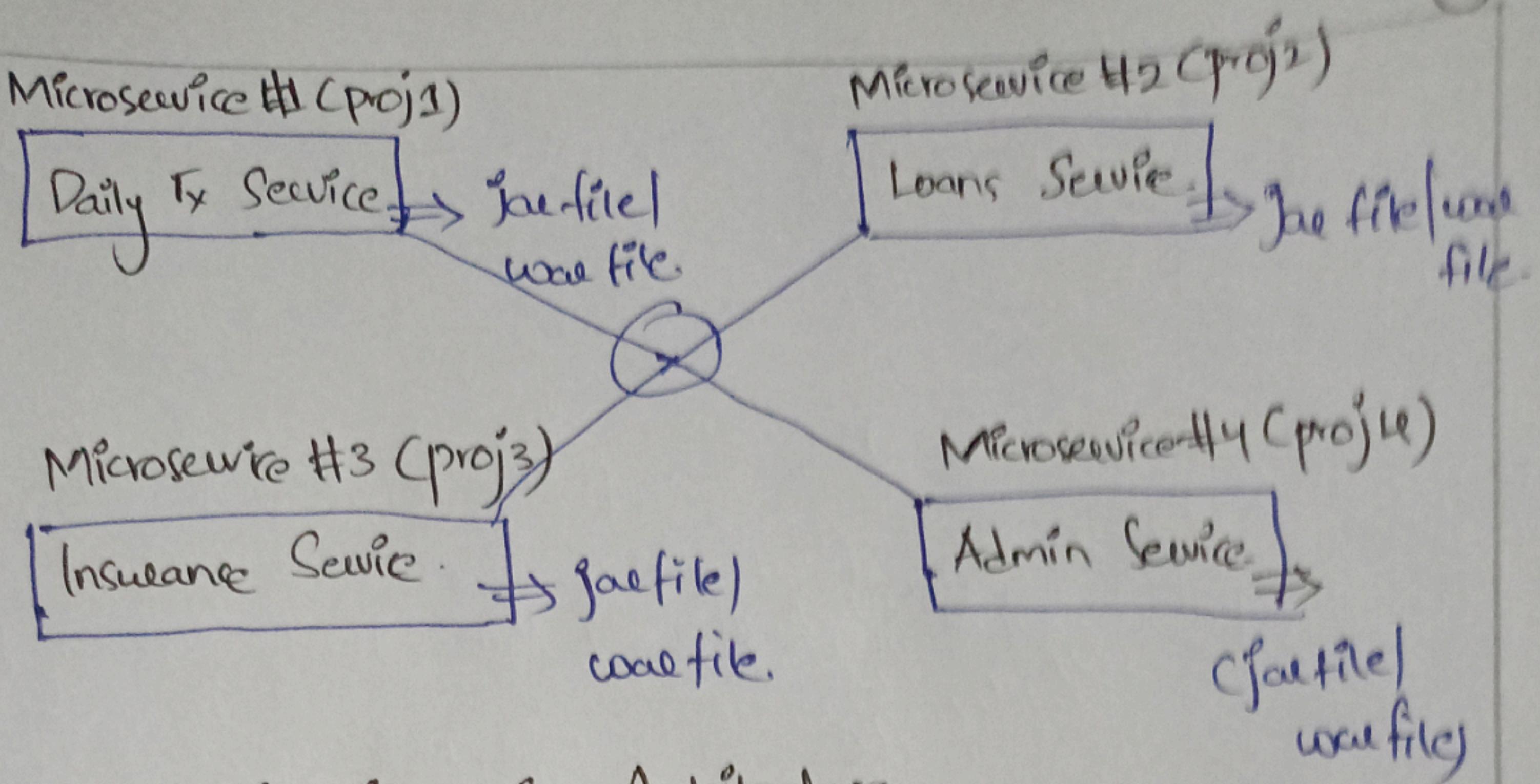
- (a) If one service/module is down then other service will be effected.
- (b) All services/modules must be developed using same language/technology/framework because all the services are integrated into single unit.
- (c) Other projects can not use specific module/service of our project.

⇒ To overcome the problems of monolith architecture take

the support of microservices architecture.

⇒ Here each service/module will be developed as the separate project nothing but microservice.

⇒ At the end different projects representing different microservices will be integrated for completing the project.



- ⇒ Advantages of Microservice Architecture.
- ⇒ If service/microservice is down, that does not effect other microservices
- ⇒ Different micro services can be developed using different languages/technologies/frameworks
- ⇒ Every micro service can be used in different projects