

 Portfolio

Park  
Minchul

# □ Contents

- Intro
- Software Technical Stack
- Backend Develop
  - ejabberd & Tsung - XMPP Signaling
  - IU - user info management server
  - Gateway - service gateway by graphql
  - Birthday - give a gift to friend's birthday
  - Dimple - Data aggregation server
  - Gaia - Cloud VSaaS
- Frontend Develop
  - Gift Reaction Project
  - Order Enhancement Project
- Infra Develop
  - Docker System using DC/OS
  - Building SmartCam Cloud System
  - Building Monitoring System using Zipkin
  - Service Java Version 8 UpgraPde
- Others
  - 26Area
  - Agile
  - S - Cube Project
  - Network Camera Video Coference
  - uC-OS II Porting
  - DPY - Do Performance Yourself

# □ Intro

- Park Minchul
  - 1989.02.20
- Career
  - Kakao
    - 2014. 09 ~ Now
  - Hanwha Techwin
    - 2014. 01 ~ 2016. 9
  - Soongsil University
    - 2007.03 ~ 2014.02
  - Samsung Software Memebership
    - 2013.12 ~ 2014.01
- Homepage : [knightpop.github.io/home](http://knightpop.github.io/home)
- LinkedIn : [linkedin.com/in/parkminchul](https://linkedin.com/in/parkminchul)
- Blog : [type-class.com](http://type-class.com)
- Github : [github.com/knightpop](https://github.com/knightpop)
- HackerRank : [hackerrank.com/knightpop](https://hackerrank.com/knightpop)



# □ Software Technical

- Language
  - Scala
  - Java
  - TypeScript
  - Python
  - PHP
  - Use any suitable language to solve the problem
- Backend
  - Play Framework & Akka
  - Finatra
  - Spring
  - Serverless
  - RxJava
- Frontend
  - React & Redux
- Infra
  - AWS & Azure
  - Mesos & Marathon & Docker



# BACKEND DEVELOP



XMPP Signaling



# EJABBERD & TSUNG

# □ ejabberd & Tsung

- Development Environments
  - OS : AWS(Ubuntu 14.04 LTS)
  - language : Erlang, Scala
  - Framework : OTP, Akka
  - base open source solution - ejabberd
- Project Goal
  - Develop solution which replaces old Samsung SmartCam XMPP Server solution.
  - Upgrade and additional development Open Source Solution, ejabberd to meet service plan
  - Develop the Load Test Program to verify Signaling Server which can control the millions camera
    - upgrade and additional development Tsung - Open source Load Test Program to meet service plan
    - Develop new Load Test Program using Scala & Akka
- Project Achievement
  - Upgrade and additional development ejabberd to meet service plan - 30,000 TCP Connection per 1 instance(AWS c4.large)
  - Upgrade and additional development Tsung to meet service plan - 60,000 TCP Connection per 1 instance(AWS r4.large)
  - Develop new XMPP Load Test Program to meet service plan base on Akka - 20,000 TCP Connection per 1 instance(AWS c4.large)
- Project Feature
  - Occur massive TCP traffic using Erlang / OPT
  - System & Erlang VM Configuration to accept massive TCP traffic and obtain resilience
  - The experience of massive traffic and handling.

# □ ejabberd & Tsung

- Development Role
  - ejabberd - Open Source Solution
    - Can maintain 30,000 TCP Connection per instance.
    - Develop the system for 20 instances can handle 60,000 TCP Connection
  - Develop and change Tsung - Open Source Load Test Program
    - Make 60,000 TCP Connection per 1 instance
    - Can occur 200 TCP Connection per second
  - Develop New Load Test Program Using Scala & Akka
    - Make 20,000 TCP Connection per 1 instance
    - Can occur 800 TCP Connection per second

User Info Management Server

kakao

IU

# □ IU - User Info Management Server

- Development Environments
  - OS : CentOS 7.2
  - Language : Scala
  - Framework : Finatra
  - Protocol: finagle-thrift
- Project Goal
  - CRUD and manage user information in Kakao commerce(gift)
  - Gradually migrate legacy Monolithic Service with Micro Service
    - Develop Micro Service using Finatra
- Project Result
  - Replace Monolithic Service Feature with Micro Service Server
  - Introduce finagle-thrift to team
  - Introduce Zookeeper to team
- Project Feature
  - Develop Micro Service Using Finatra
  - Server Cluster Management using Zookeeper
  - Fast response calling API and comfortable Integration with finagle-thrift

# □ IU - User Info Management Server

- Development Role
  - Server
    - Build micro server using Finatra
  - Make ‘Easy to Use’ API by Write Thrift IDL
  - API Client
    - Write both Scala client and Java client to achieve ‘Easy to Use’ API
    - Develop the new concept of the user account to Kakao Gift Service
    - Make new API(V2) to Develop the new concept of the user account. But compatible with Legacy System
  - Migrate legacy system to micros ever

Graphql API Server

kakao

GATEWAY

# □ Gateway - Graphql API Server

- Development Environments
  - OS : CentOS 7.2 on Docker
  - Language : Typescript
  - Framework : Node + Graphql-Yoga
  - Protocol: graphql
- Project Goal
  - Serving data to inner other services in Kakao
  - Request schema is different from each service. (Because of each service has own domain)
  - The project run on DC/OS for high availability
- Project Achievement
  - #Tab(Shopping tab, request high traffic in Kakao) start use Gateway service
- Project Feature
  - The flexible protocol using Graphql
  - Achieve high availability using DC/OS and Docker

kakao

# □ Gateway - Graphql API Server

- Development Role
  - Building a Docker system
    - Build high availability Docker Service
  - Auto-generate Graphql Shema + Quert
    - Auto-generate Graphql Schema and Query using Typed-Graphql
  - Dependency Injection using Typedi

Give a gift to friend's birthday

kakao

BIRTHDAY

# □ Birthday

- Development Environments
  - Language : Scala, Typescript
  - Framework : Play, React, Finatra
- Project Goal
  - Inform to user friend's birthday in the first tab
  - Make user give a gift by Kakao Gift and make additional revenue
- Project Achievement
  - Improve usability
    - Inform user friend birthday instead of Facebook
  - Improve revenue
    - 18 billion additional income every year
- Project Feature
  - Receive huge traffic every 12 AM because people want to give gift A.S.A.P
  - Require rapid release. So, develop 1<sup>st</sup>, 2<sup>nd</sup> project parallelly

kakao

# □ Birthday - 1<sup>st</sup>

- Develop Role
  - Develop alone except design
  - Front Page
    - Make 1<sup>st</sup> project page using React & Redux
  - Build Micro Server recommend user proper gift(from best data)
    - Because huge traffic every 12 AM so, adapt cache in the recommendation
    - Build service failover
  - Load Clock Log to analyze user data
  - Add Feedback API to check if the user really use birthday page

# □ 1st Page View

KT LTE 오후 3:24 76%

강혜영\_Jul..님의 생일을 축하해주세요!

6월 7일

인기있는 생일선물

**롯데시네마**  
롯데시네마 주말예매권(금~일)  
**10,000원**

**프렌즈리빙**  
[6월특가] 카카오프렌즈 언더 더씨 드링크자 4종 택1  
**8,900원**

**peripera**  
1+1+1

KT LTE 오후 3:25 76%

**GS25**  
GS25 모바일 상품권 1만원권  
**10,000원**

**경복-사과**  
사과100%사과꽃향기 사과즙 100mlX50팩 (옐로우기프...  
**13,900원**

**스타벅스**  
아이스 카페아메리카노 Tall  
**4,100원**

**KAKAO FRIENDS**

KT LTE 오후 3:25 76%

**베스킨라빈스**  
페밀리 아이스크림  
**19,500원**

**문화상품권**  
온라인문화상품권 100,000원권  
**100,000원**

**GS25**  
GS25 모바일 상품권 5천원권  
**5,000원**

선물하기 홈에서 더보기

## □ Birthday - 2<sup>nd</sup>

- Development Role
  - Project to improve service and revenue
    - Make 2<sup>nd</sup> project page using React & Redux
  - Build Micro Server recommend user proper gift(from user clustering algorithm)
    - Because huge traffic every 12 AM so, adapt cache in the recommendation
  - Load Clock Log to analyze user data
  - Add Feedback API to check if the user really use birthday page

Data aggregation Server

kakao

DIMPLE

# □ Dimple - Data Aggregation Server

- Development Environments
  - OS : CentOS 7.2
  - Language : Scala
  - Framework : Finatra
  - Protocol : http rest
- Project Goal
  - Service layer which manufactures page view data from various Micro Service In Kakao Gift Service
  - Make new Service Layer, which can manufacture various data not from the view layer server but from Dimple Server
- Project Achievement
  - Make new module used prom reaction project
- Project Feature
  - aggregate many Micro Services data and make one JSON Object.

# □ Dimple - Data Aggregation Server

- Development Role
  - Gift Inbox Detail Page Data
    - Make the new module in Dimple which can manufacture data
  - Change Architecture
    - Change Server Architecture to DDD Architecture

Cloud VSaaS



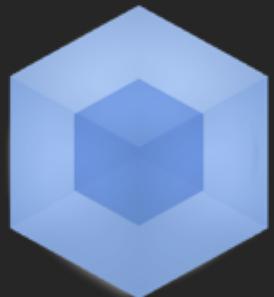
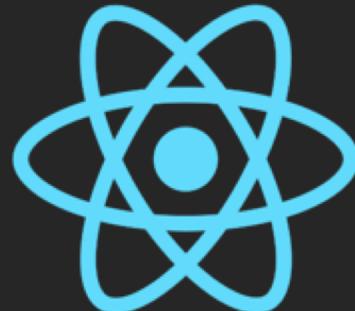
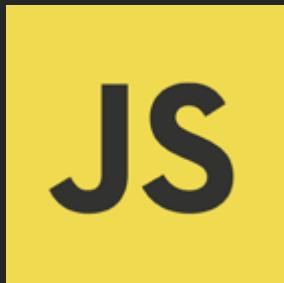
GAIA

# □ Gaia - Cloud VSaaS

- Development Environments
  - OS : AWS(Ubuntu 14.04 LTS) & Azure(Ubuntu 15.10)
  - Language : Scala, Java
  - Framework : Play, Akka
- Project Goal
  - VSaaS Live Video Streaming on AWS Cloud
  - Playback Server Accept RTSP Stream and store video
- Project Result
  - Implement Auto Scaling and we can playback fluently even serviced in 3g data communication. It's Prototype
  - Develop playback server accept RTSP Video Stream and convert to MPEG-dash and send.
- Project feature
  - Service Signaling server using ejabberd.
  - Make EC2 Cluster Concurrent Server by Akka, Play
  - WebRTC Adaptive Streaming with Kurento Media Server

# □ Gaia - Cloud VSaaS

- Development Role
  - Design and implement Live Streaming Server(POC)
    - Design and implement RTSP to WebRTC Transcoding, the live streaming server using Kurento Media Server
  - Design and implement AWS Instance Cluster(POC) Management and Business Logic Server
    - Design and implement Auto Scaling Instance management and Business Logic Server using Play Framework
  - Design and implement RTSP Endpoint server to playback RTSP Camera Video(POC)
    - Design and implement RTSP Endpoint using GStreamer
  - Design and implement Playback Server(POC)
    - Design and implement Playback Server connected with AWS S3 by changing Open Source eDash-Packager



# FRONTEND DEVELOP



Renewal Gift InBox Page

kakao

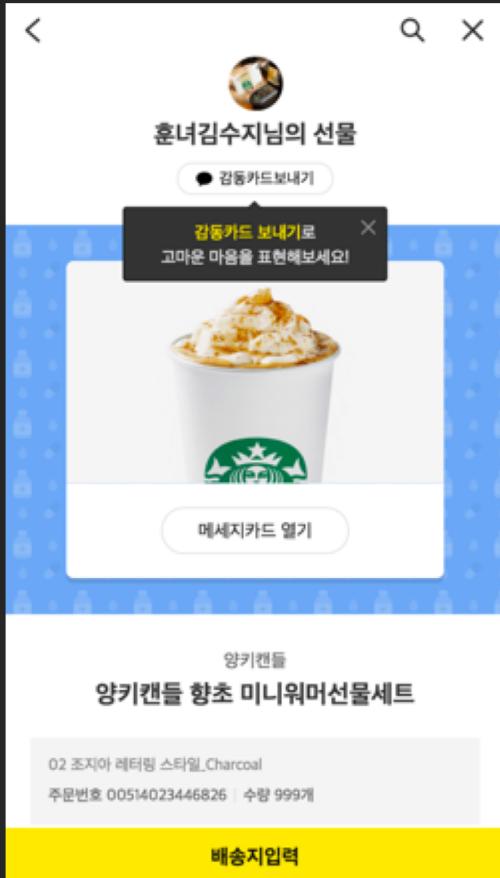
# GIFT REACTION

# □ Gift Reaction

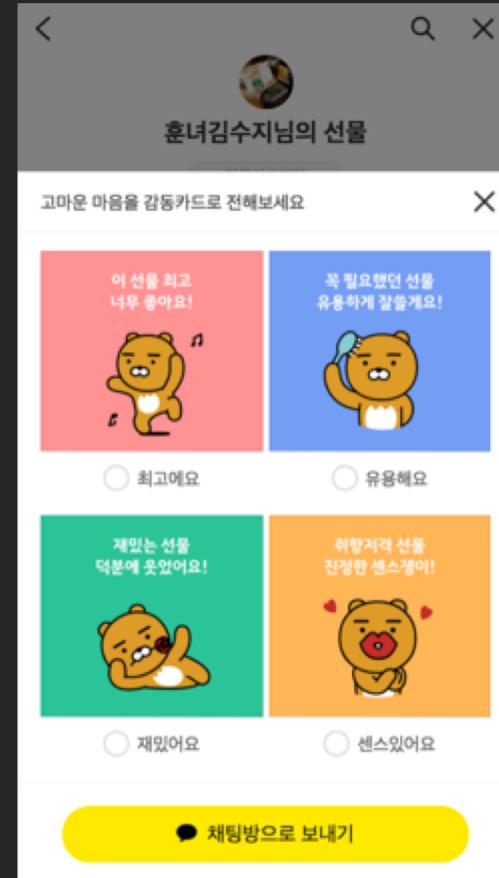
- Develop Environment
  - Language: TypeScript
  - Framework : React, Redux
- Project Goal
  - Project to aggregate data about people's feeling who receive the gift in Kakao Gift Service
  - Renewal page to replace legacy code(Angular JS 1) with React
- Project Achievement
  - Gift Reaction
    - Renewal Gift Inbox page.
    - Aggregate people's feeling who receive the gift in Kakao Gift Service
- Project Feature
  - Make the front page using TypeScript

kakao

# □ Page Views



Inbox detail page



Inbox reaction page

kakao

Renewal Item page and Best tab

kakao

# ORDER ENHANCEMENT PROJECT

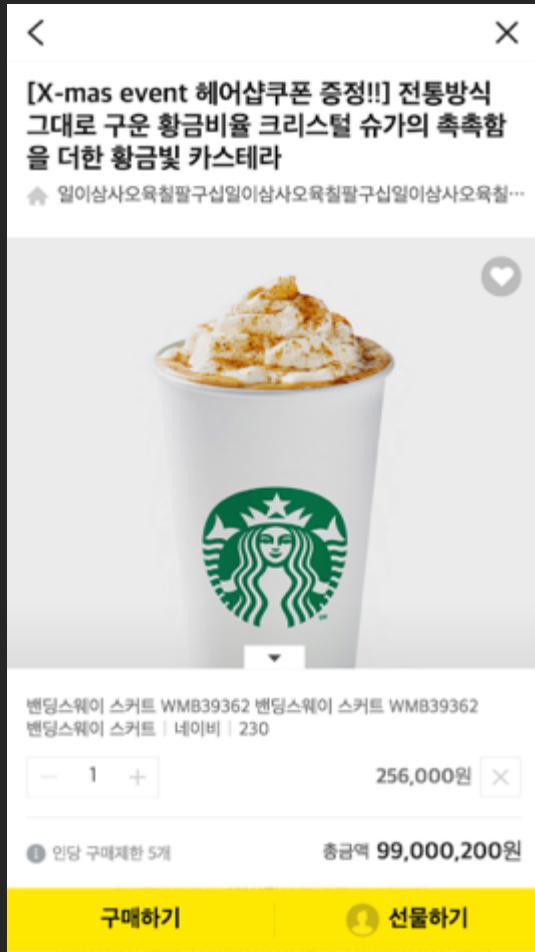
# □ Order Enhancement Project

- Development Environments
  - Language : javascript es6
  - Framework : React, Redux
- Project Goal
  - Develop Web App frontend for Kakao commerce(gift)
  - Renewal page to replace legacy code(Angular JS 1) with React
- Project Result
  - Best tab page
    - Develop new Best Tab web page to launch new service
  - Renewal Order Project
    - Develop new Item Detail page to enhance UX
- Project Feature
  - Use React, Redux, es6 to Develop modularized front page

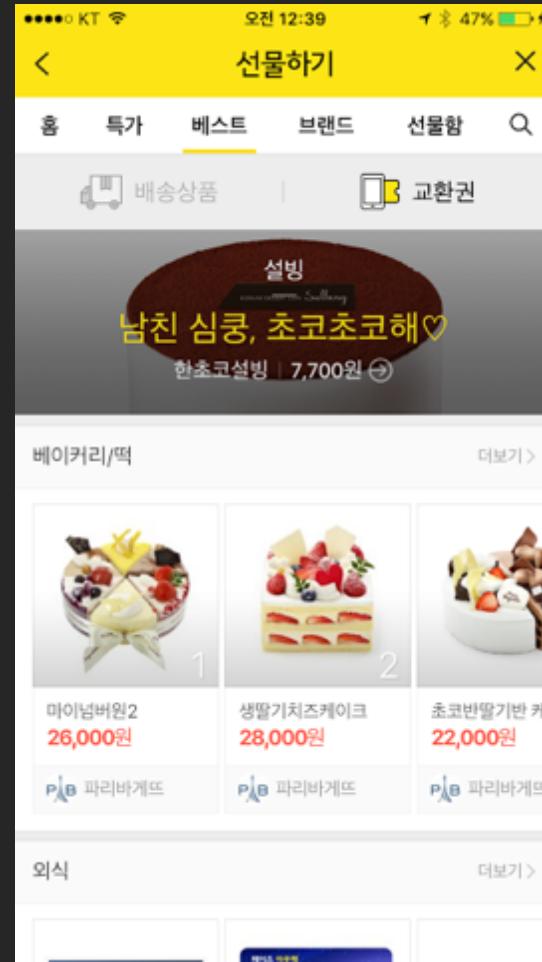
kakao

# □ Page Views

Item Detail Page

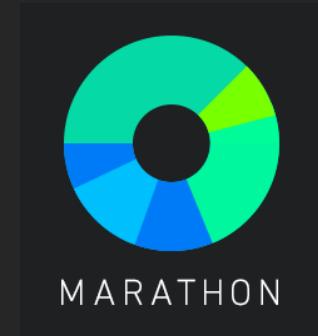


Gift Best Tab





# INFRA DEVELOP



Build Container System

kakao

KAKAO GIFT DC/OS

# □ Build DC/OS in Kakao Gift

- Development Environment
  - Mesos + Marathon + Docker
  - Kakao DK/OS
- Project Goal
  - Manage server and resource in Kakao Gift Service Efficiently
  - Easy and Fast to deploy
- Project Goal
  - Admin, Worker Develop Server Migration. (Primary)
  - Migrate all API server
- Project Feature
  - Migrate to Container Platform to legacy service with changing Container Native Service

AWS Cloud System



SAMSUNG SMARTCAM

# □ Samsung SmartCam V2 Architecture

- Development Environment
  - Cloud Platform: AWS
  - EC2, RDS(RDS), ElasticCache(Redis)
- Project Goal
  - With eJabberd, build new Architecture in Samsung SmartCam
- Project Feature
  - Compatible with legacy System
  - The affordable system which 4 million Camera connect at the same time

Micro Service Monitoring System

kakao

ZIPKIN

# □ Zipkin - Monitoring Service

- Develop Environment
  - Zipkin
- Project Goal
  - Trace Micro Service and find service bottleneck in real time
- Project Achievement
  - Integrate all server (except 1 legacy server). And find various service bottleneck
- Project Feature
  - Measure Round Time in Play, Akka, Finatra

Java 7 to Java 8 Version Up

kakao

# SERVICE JAVA VERSION UP

# □ Kakao Gift Service legacy Java Version up

- Development Environment
  - Play 1.4
  - Java 7
- Project Goal
  - Java version up to use API Client using Java 8
  - Migrate legacy system to Java 8 from Java 7
- Project Achievement
  - Migrate to Java 8 from Play 1.4 & Java 7
- Project Feature
  - Analysis Application base on Java migration document
  - Achieve resilience service by long-run test



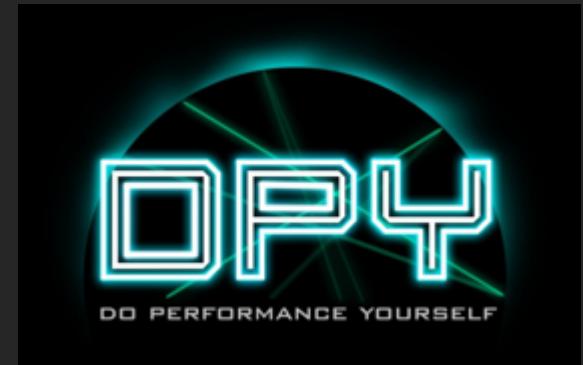
# 26Area



## OTHERS



SAMSUNG  
SOFTWARE  
MEMBERSHIP



B Class Humor Community

**26Area**

26AREA.COM

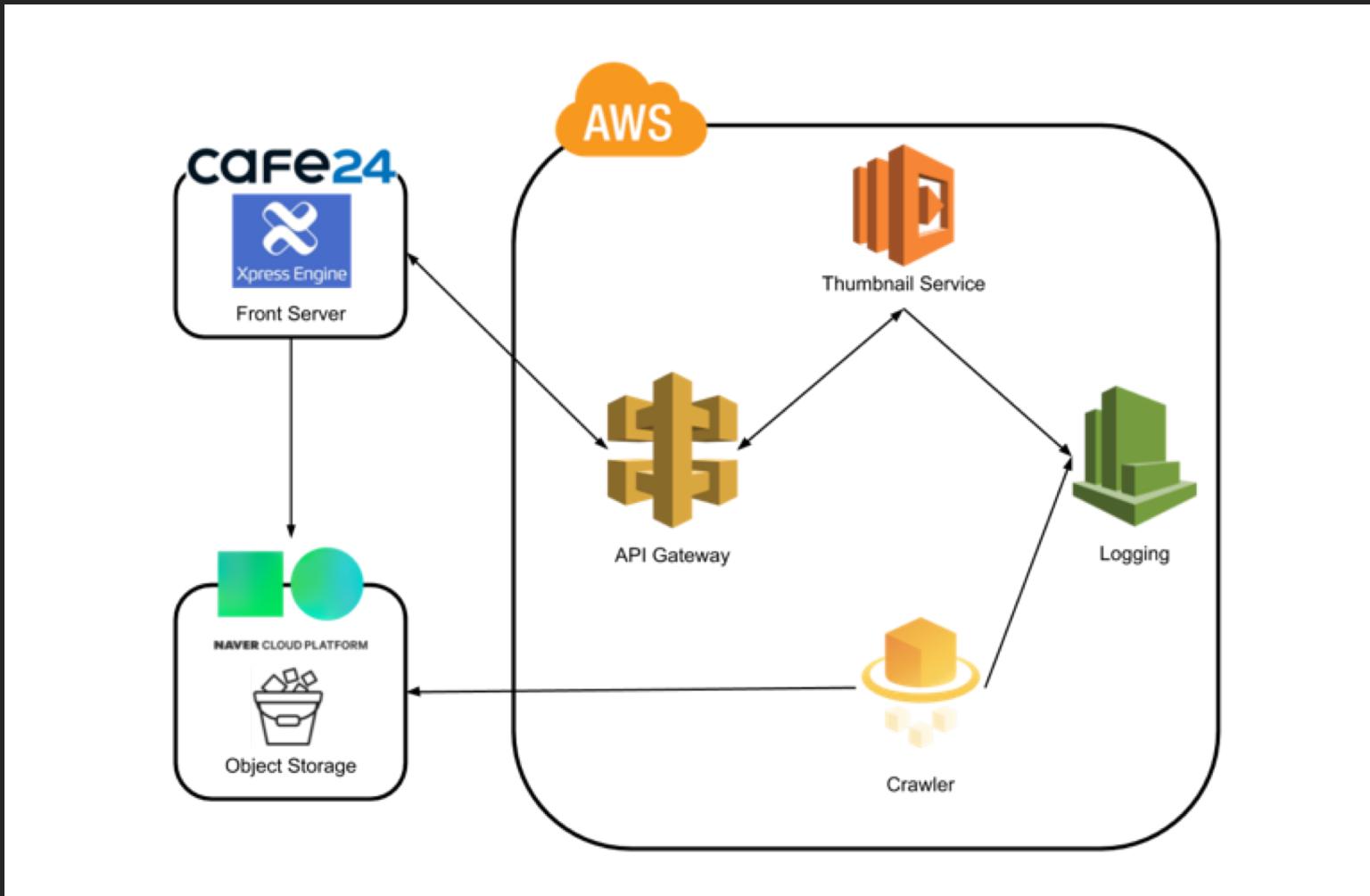
# □ 26Area.com

- Development Environment
  - XE-Core(CMS Solution, PHP)
  - Serverless(Typescript)
  - Scrapy(Python)
- Project Goal
  - Gathering Humor Document In Korea Internet
  - Build community site to make user communicate each other
- Project Achievement
  - Fast release using XE
  - Serverless Docker using AWS Fragate
  - Auto-deploy using Serverless Framework
  - Currently(2018-08-26) 200 user
- Project Feature
  - Short money because it is side-project(\$20 / month)
  - Using various Cloud platform(AWS, NCloud, Cafe24)

# □ 26Area.com

- Development Role
  - Build Site Using XE(CMS Solution)
    - Fast building using XE made by PHP
    - Using Cafe24 to reduce cost
  - Website Crawling Using Scrapy
    - Crawling Framework made by Python
    - Using AWS Fragate to reduce instance cost
  - Make Thumbnail Service Using Serverless Framework
    - Service on AWS Lambda
    - Smart Crop base on Shannon entropy
  - Operate Service
    - Operate stand alone because personal side-project

## □ 26Area.com Architecture



Introducing agile to team

kakao

AGILE

# □ Introducing agile to team

- Background:
  - Hard to make a balance between CS issue and technical issue
  - Number of bugs are growing because of technical debt
  - Become lower of Engineer's condition
- How to solve:
  - Adapt Agile to considering Kako Gift service environment.
- Introduction:
  - Scrum with product manager
  - Sprint Meeting
  - Introducing Scrum master
  - Introducing Scrum Board
  - Introducing Demo day

NVR/DVR Common



# S - CUBE PROJECT

# □ S - Cube Project

- Development Environments
  - OS : Ubuntu 12.04 LTS
  - Language : C++
- Project Goal
  - Make new generation NVR Platform To substitute existing Samsung Techwin NVR Platform, Sejong
- Project Result
  - Re-Design to concrete abstract Layer in existing architecture, and export hardware specific feature by XML to implement One Source Multiple Use
- Project feature
  - Collaboration with SRIB, India(Develop in India, in the field)
  - Aim One Source, Multiple Model

# □ S - Cube Project

- Development Role
  - Camera Manager
    - Design and implement Network Camera management and PTZ control in the same network using protocol ONVIF and open source gSoap
  - Log Service
    - Design and implement the module to manage, store and search all logs made by NVR using SQLite and open source Kompex Wrapper
  - Code Coverage Test Case and Management
    - Code Test code to process Test Driven Development using Gcov, CPPUnit

Samsung Techwin



# NETWORK CAMERA VIDEO CONFERENCE

# □ Network Camera Video Conference

- Development Environments
  - OS : Window 7
  - Language : C#(WPF)
- Project Goal
  - Develop Program which supports existed Samsung Techwin CCTV to using Video Conference
- Project Result
  - Video Conference multiple people using CCTV and RTSP
- Project Feature
  - Re-Use Existed CCTV



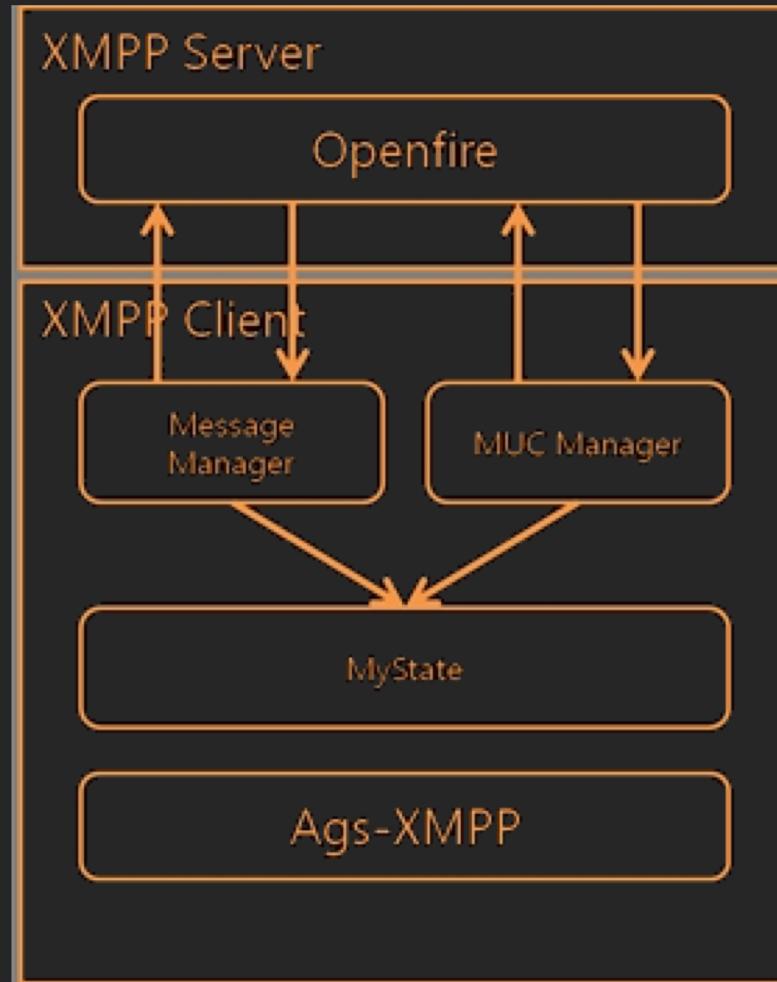
# □ Network Camera Video Conference

- Project Role
  - Program View and Business Logic
    - Design and implement Program View using WPF and All the Model Controller, Business Logic Module
  - Chatting Module
    - Design and implement to manage member, chatting room in the chatting server
    - Design and implement Openfire chatting client using XMPP Library, AGS-XMPP
  - Vote Module
    - Design and implement vote module to vote specific subject in Videoconference



# □ Network Camera Video Conference

- Chatting Module Block Diagram



# □ Network Camera Video Conference

- UI Design



ARM Porting



UC - II OS

# □ uC - II OS ARM Porting

- Development Environments
  - OS : Ubuntu 12.04 LTS
  - Language : C / ARM Assembly
  - Target Board : Odroid 7
- Project Goal
  - Port one of the RTOS operated in Window, uC - II OS to ARM Chip.
- Project Result
  - Make uC-OS II only operated in Power PC and Intel Chip to operate in ARM Architecture
  - Implement Dynamic scheduling like Linux nice value. Before, its burden to engineer
- Develop Environment
  - Use s5pc110 Chip, used in Samsung Galaxy S
  - Use ARM-none-eabi Cross Compiler

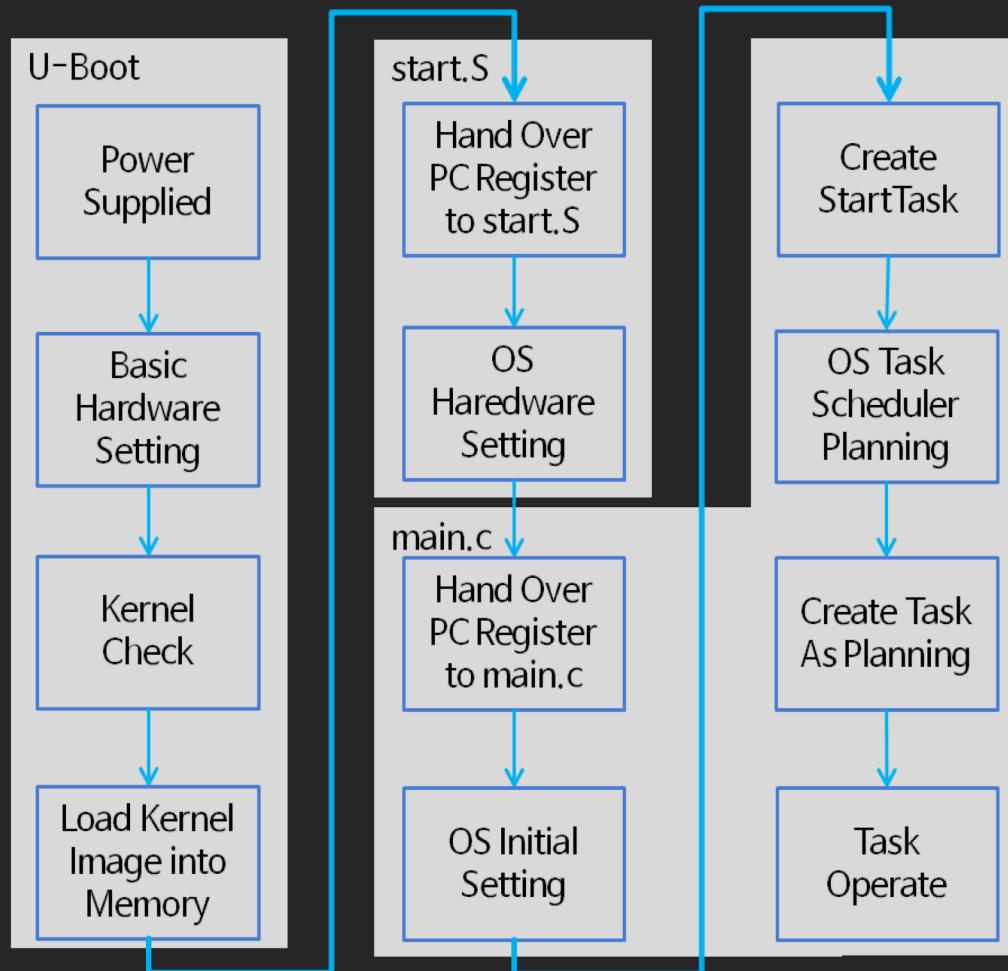


# □ uC - II OS ARM Porting

- Develop Role
  - Revise U-Boot and make it load uC - OS Kernel image in s5pc110 chip
  - Write uC - OS - II chip dependent hardware setting code by ARM Assembly to enter main entry
  - Make chip dependent driver like U-ART
  - Write interrupt code to context switch
  - Change uC-OS II Scheduler to use the nice value like Linux
- Performance Test
  - Using Timer inside Board to Performance Test

# □ uC - II OS ARM Porting

- uC - OS II Porting Kernel Operation



Do Performance Yourself



DPY

# DPY - Do Performance Yourself

- Development Environments
  - OS : Window 7
  - Language : C#(Unity3D)
- Project Goal
  - Develop Interaction Media Performance Content using Kinect
- Project Result
  - Develop interactive digital art React to user behavior
- Project Feature
  - User motion capture using Machine Learning
  - Character Follow User by Unity 3D



# □ DPY - Do Performance Yourself

- Project Role
  - Implement 3D Interactive Interface
    - Implement user UI, 3D motion, effect using Unity 3D
  - Implement module interact user and character
    - Implement interact module by Kinect and Zigfu
  - Implement Business Logic combine with another module
    - Combine machine Learning Module and DSP Module



# □ DPY - Do Performance Yourself

- UI

