

RESUME

iOS Developer - 한영신



팀 프로젝트로 진행한 **iOS 앱을 2번 실제로 출시한 경험**이 있고,
더 **퀄리티 높은 코드**를 작성하기 위해 나아가고 있습니다.

주어진 상황을 불평하기 보다는 **긍정적으로** 생각하고, 새로운 것을 배우길 좋아합니다.
모바일 QA 엔지니어로 5년 간 근무하여 협업 경험이 많고,
Android, iOS 생태계를 잘 알고 있습니다.

[Index]

- [1. 프로젝트](#)
- [2. 보유 기술](#)
- [3. 경험](#)

기술 스택

#익숙한 것들

- Swift, AutoLayout, Xib, Profiler, REST API
- 리팩토링, App LifeCycle, UI개발
- Tool : Xcode, Slack, JIRA, Zpelin

#활용가능한 것들

- Git, GitLabCI, Linux

#코드를 읽고 이해할 수 있는 것들

- Rx&MVVM, Android&Kotlin, Python

STUDY

2021 년

SSAC 라이징프로그래머스 iOS 개발 과정 수료 (4개월)
수익형 앱 런칭 동아리 MakeUS iOS 개발자로 참여
소프트스퀘어드 코드리뷰 스터디 참여
한양사이버대학교 컴퓨터공학 학사 졸업

2020년

정보처리기사 자격증 취득
리눅스마스터(2급) 자격증 취득



Work Experience

모바일 QA Engineer - 1년

- Android, iOS 개발 사항 검증(블랙박스, 화이트박스 테스트)
- OS 성능 최적화 테스트(Xcode Profiler)
- appium 기반 UI 자동화 테스트 개발
- 자동화 테스트 CI 파이프라인 구축

모바일 게임 QA Engineer - 4년

- 모바일 게임/플랫폼(Android, iOS) QA
- 모바일 게임 보안 취약점 분석 업무
- 10개 이상의 프로젝트 런칭, 유관 부서와의 협업
- 올해의 우수 사원 / 우수 신입 사원상 수상

iOS



Projects

- OfMe(앱스토어 출시)
- 리딩피스(앱스토어 출시)
- Grip 클론 코딩
- 개인 프로젝트 & 기술 스택
 - 위치 기반 날씨 정보 앱
 - 푸시 기능(FCM)
 - 비동기 데이터 처리
 - Multi Thread를 활용한 API 호출
 - Firebase

OfMe

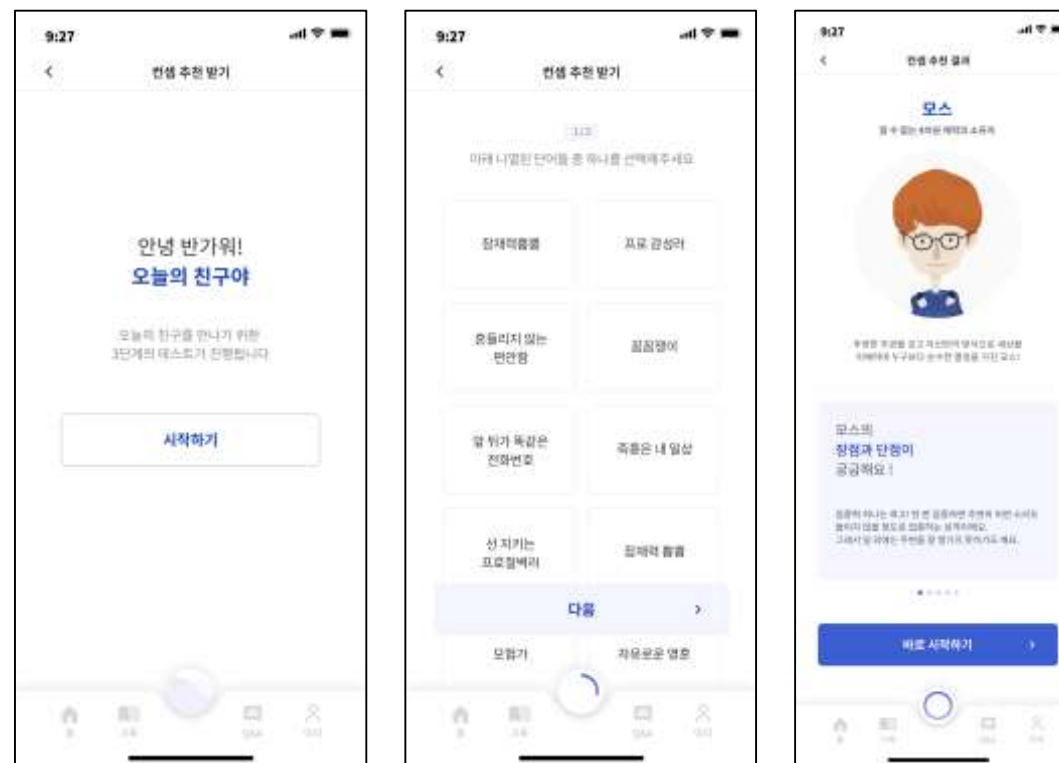


팀 프로젝트 [\[Repo\]](#) [\[프로젝트소개\]](#)

기간 1 개월

Skills

- 프로토타입 개발이 끝난 프로젝트에 합류하여 리팩토링 진행
리팩토링 과정에서 클린아키텍처와 리팩토링의 중요성 확인
- 소셜로그인(카카오, 애플) 개발, 성격테스트 기능 리팩토링
 - 클린아키텍처, MVVM, Firebase App Distribution



[리팩토링 포인트]

1) 네이밍 컨벤션 통일

비슷한 버튼 클릭 액션을 구현하더라도 사람에 따라서
[click, touchdown, selected] 등 다양한 네이밍 컨벤션 존재
다양한 네이밍 때문에 헷갈릴 수 있는 부분을 통일

2) Massive ViewController의 책임 분산

하드코딩 형태로 ViewController에 집중된 코드를
다른 파일에 적절하게 위임하여 코드의 가독성을 높임
Manager클래스를 생성하고, View와 관련 없는 로직은 최대한 덜어냄

3) 확장 가능한 Type 생성

확장 가능성이 높은 로그인 방식을 별도의 Custom Type으로 분리해서
추후 확장할 때 해당 구조체만 수정하면 대응 가능하게 변경

```
class SocialLoginManager: NSObject {  
    var loginType: SocialLoginType  
    var viewController: UIViewController  
  
    init(_ viewController : UIViewController, socialLoginType: SocialLoginType) {  
        self.loginType = socialLoginType  
        self.viewController = viewController  
    }  
  
    func login() {  
        if loginType == .apple {  
            appleLogin()  
        } else {  
            kakaoLogin()  
        }  
    }  
}
```

리딩피스



리딩피스 
주식회사 소프트웨어
Designed for iPhone
Free

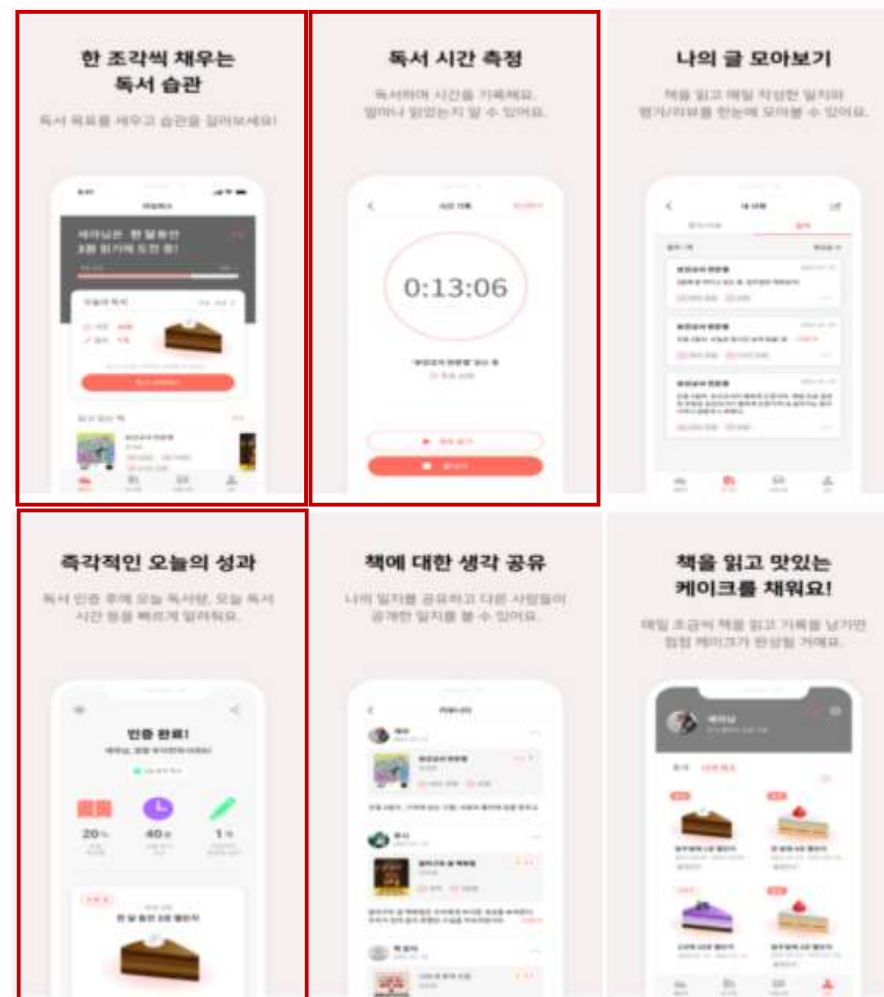
팀 프로젝트 [\[Repo\]](#)

기간 2 개월

Skills

- SSAC iOS개발자 과정 최종 팀 프로젝트로
독서 내용과 후기를 공유하는 커뮤니티 앱
- 스토리보드 Scene기반 협업, SpriteKit
 - 제네릭을 활용한 API 통신, 앱스토어 배포 프로세스

Features



리딩피스

What I Learned

1. Generic을 활용한 네트워크 모듈 개발

01 Issue

- 여러 개발자가 API를 연동하면서 코드 일관성 저하 우려
- 여러 ViewController에 API 호출에 필요한 코드가 산재

02 Solution

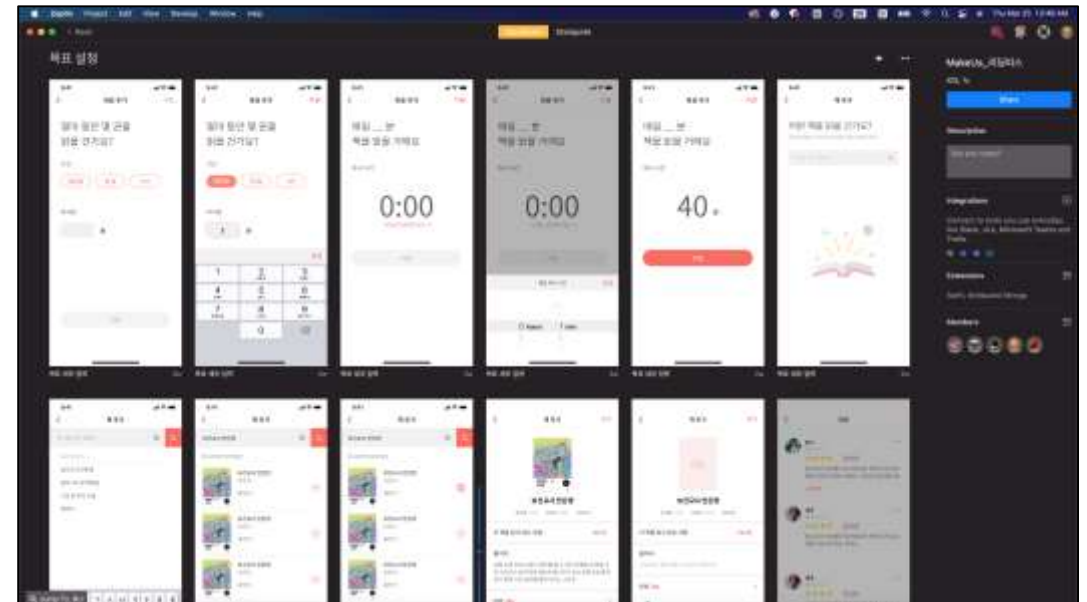
- Generic, Protocol을 이용해 공통으로 사용 가능한 객체 생성
- 다양한 HTTP메소드에 대응 가능하도록 모듈화

Generic Code

```
protocol Requestable {  
    associatedtype ResponseType: Codable  
  
    var endpoint: String { get }  
    var method: Network.Method { get }  
    var query: Network.QueryType { get }  
    var parameters: [String: Any]? { get }  
    var headers: [String: String]? { get }  
    var baseUrl: URL { get }  
    var timeout: TimeInterval { get }  
    var cachePolicy: NSURLRequest.CachePolicy { get }  
}  
  
extension Requestable {  
    var defaultJSONHeader: [String: String] {  
        return ["Content-Type": "application/json"]  
    }  
}  
  
enum NetworkResult<T> {  
    case success(T)  
    case cancel(Error?)  
    case failure(Error?)  
}
```

2.디자이너, 개발자간 협업

- : Git을 이용한 소스코드 관리
- : 디자이너와의 협업 (제플린, 피그마)
- : 서버 개발자와의 협업 (API구조 논의)



Grip



개인방송, 쇼핑몰 앱 클론 코딩

팀 프로젝트

기간 2 주

Skills

SSAC iOS개발자 과정 중 진행한 클론 코딩 팀 프로젝트
인기있는 라이브커머스 앱 Grip을 선정하여 진행
- Xib기반 UI개발, Protocol 활용, 네이버 소셜로그인



[개발포인트 : Protocol 활용한 데이터 전달]

01 Issue

- 리뷰 필터링을 위해 TableView와 TableViewCell 간의 데이터 전달 필요

02 Solution

- Protocol Oriented Programming 활용하여
- TableView와 TableViewCell 간에 Delegate로 데이터 전달

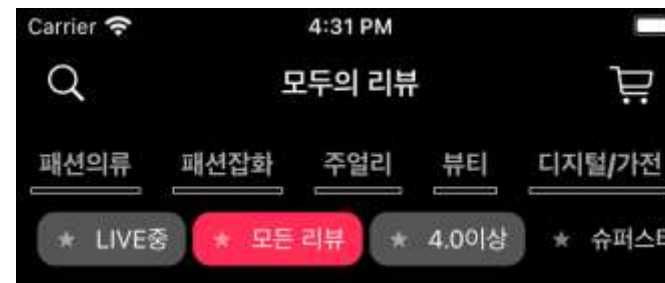
Protocol Code

```
import Foundation

protocol ReviewFilterCellDelegate {
    func selectedReviewFilter(selectedFilter:String)
}

protocol ReviewCategoryFilterCellDelegate {
    func selectedCategoryFilter(selectedCategory:String)
}

class ReviewFilterTableViewCell: UITableViewCell {
    @IBOutlet weak var reviewFilterCollectionView: UICollectionView
    static var identifier: String = "ReviewFilterTableViewCell"
    var cellDelegate: ReviewFilterCellDelegate?
```



[개인 프로젝트]

GPS와 푸시메세지 기능 구현 [\[Repo\]](#)

주변 날씨와 미세먼지 경보(푸시)를 알려주는 앱

Skills

CoreLocation, Info.plist 권한 처리, FCM, 푸시서버(AWS EC2 + Flask)

알림	상태 ?	플랫폼
➤ 테스트 푸시	✓ 완료됨	iOS
➤ 미세먼지 좋음!!	✓ 완료됨	Android

Multi Thread를 활용한 API 호출 [\[Repo\]](#)

게임사 API를 활용하여, 여러 유명 게임의 닉네임 중복 검사를 한번에 진행

Skills

Dispatch Group 기반 Multi-Thread 프로그래밍
클린아키텍처를 참고하여 코드 리팩토링 진행 중

```
func getNicknameData(nickname: String, completion:@escaping ([Response]) -> Void) {
    let group = DispatchGroup()
    let apiQueue = DispatchQueue(label: "api")
    var result: [Response] = []

    group.enter()
    apiQueue.async {
        NexonAPI().serachInFifaKartRider(nickname: nickname) { (data) in
            result.append(data)
            group.leave()
        }
    }

    group.enter()
    apiQueue.async {
        NexonAPI().serachInFifaOnline(nickname: nickname) { (data) in
            result.append(data)
            group.leave()
        }
    }

    group.enter()
    apiQueue.async {
        NexonAPI().serachInDnf(nickname: nickname) { (data) in
            result.append(data)
            group.leave()
        }
    }
}
```

Singlethone패턴 활용

Firebase로 CRUD를 구성한 커뮤니티 앱

Skills

Firebase인스턴스 접근시 자주쓰는 기능을 Singlethone으로 구현

Singleton Code

```
import Foundation
import FirebaseFirestore
import FirebaseAuth

class FirebaseManager {

    static let shared : FirebaseManager = FirebaseManager()

    var isCommented : Bool? = nil

    var brandList : [Brands] = [] {
        didSet {
            print("브랜드 정보 불러오기 완료")
        }
    }

    var productList : [Products] = [] {
        didSet {
            print("상품 정보 불러오기 완료")
        }
    }

    var productDetail : ProductDetail? = nil {
        didSet {
            print("상품 상세 정보 불러오기 완료")
        }
    }

    var reviewList : [UserReview] = [] {
        didSet {
            print("유저 리뷰 정보 불러오기 완료")
        }
    }

    var famousProductList : [ProductDetail] = [] {
        didSet {
            print("인기 상품 정보 불러오기 완료")
        }
    }
}
```

이미지 비동기 처리 & Data prefetch

주요 음원 사이트별 랭킹 정보를 크롤링하여 제공하는 앱

Skills

이미지 로딩 지연을 방지하기 위해서 Data prefetch 적용
Search Controller, 고차 함수 활용

Search Controller Code

```
// MARK: SearchController Setting
extension MelonViewController : UISearchResultsUpdating {
    func updateSearchResults(for searchController: UISearchController) {
        guard let text = searchController.searchBar.text else { return }
        self.filteredSongs = self.songs.filter { $0.title.lowercased().hasPrefix(text) }
        self.melonTableView.reloadData()
    }

    func searchBarIsEmpty() -> Bool {
        // 검색 결과가 없는 경우
        return searchController.searchBar.text?.isEmpty ?? true
    }

    func filterContentForSearchText(_ searchText: String, scope: String = "All") {
        songs = songs.filter({ song : Song -> Bool in
            return song.title.lowercased().contains(searchText.lowercased())
        })
    }
}
```



Skills

- 앱 성능 측정
- UI 자동화테스트 개발
- CI Pipeline 구축
- 해킹 취약점 진단/분석

앱 성능 측정

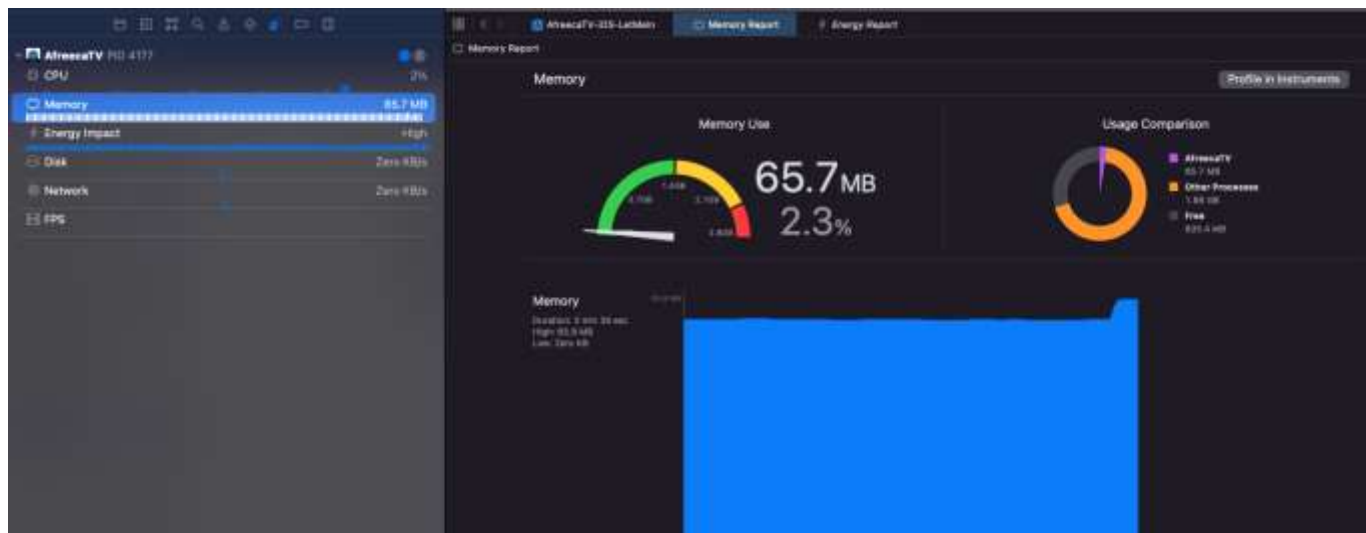
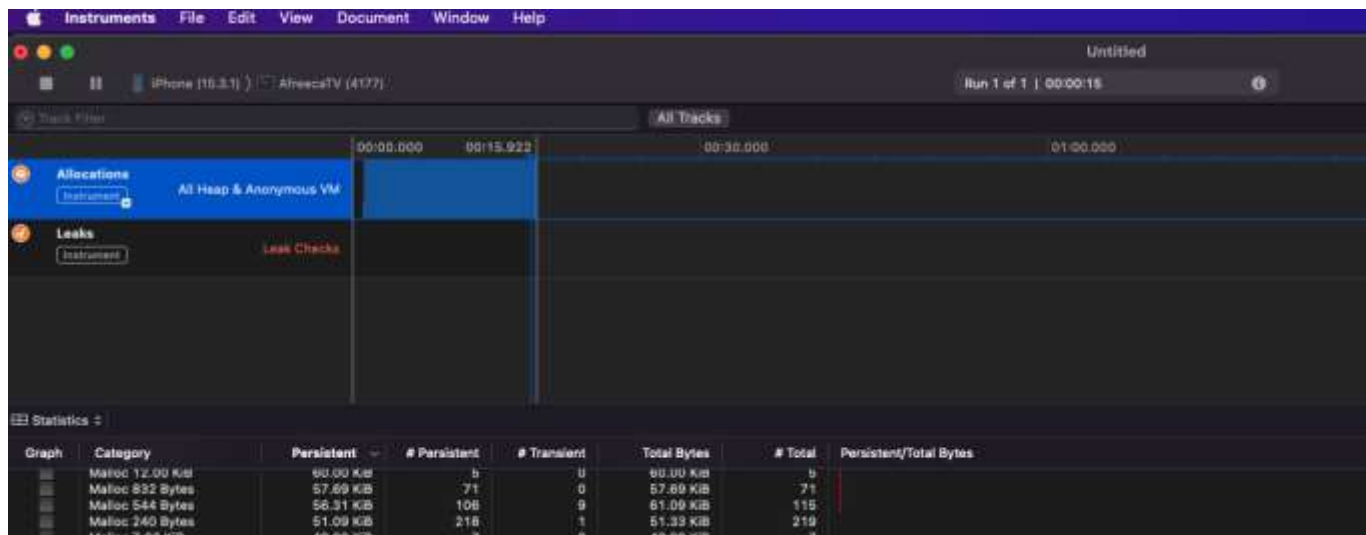
Skills

개요

- 정기적으로 성능 측정 업무 진행
- GPU로드가 많은 스트리밍 기능에 대한 성능 점검 진행

주요 내용

- Xcode Profiler, Instruments 활용
- 메모리 누수, GPU로드율 체크



```

from selenium import webdriver
import time
import urllib.request
from bs4 import BeautifulSoup

# ErrorHandler : 크롤링 도중 에러 발생시 보따를 손으로 받아들이고, 코딩하는 사람이 용이함. Yes, 3 months ago
Yes, 3 months ago · 1 author (You)

class ErrorHandler :
    crawling_fail_text = "크롤링 실패"
    crawling_completed_text = "크롤링 완료"
    connection_fail_err_text = "Error : 타겟 사이트 접속 불가\n"

# Review : 리뷰의 일자, 별점, 내용 : date(일자 날짜), comment(내용 내용), stars(별점)
Yes, 3 months ago · 1 author (You)

class Review:
    date = ''
    comment = ''
    stars = 0

    def __init__(self, date, comment, stars):
        self.date = date
        self.comment = comment
        self.stars = stars

# 크롤링 시작 & 크롤링 결과 리스트를 받고 있는 시작
Yes, 3 months ago · 1 author (You)

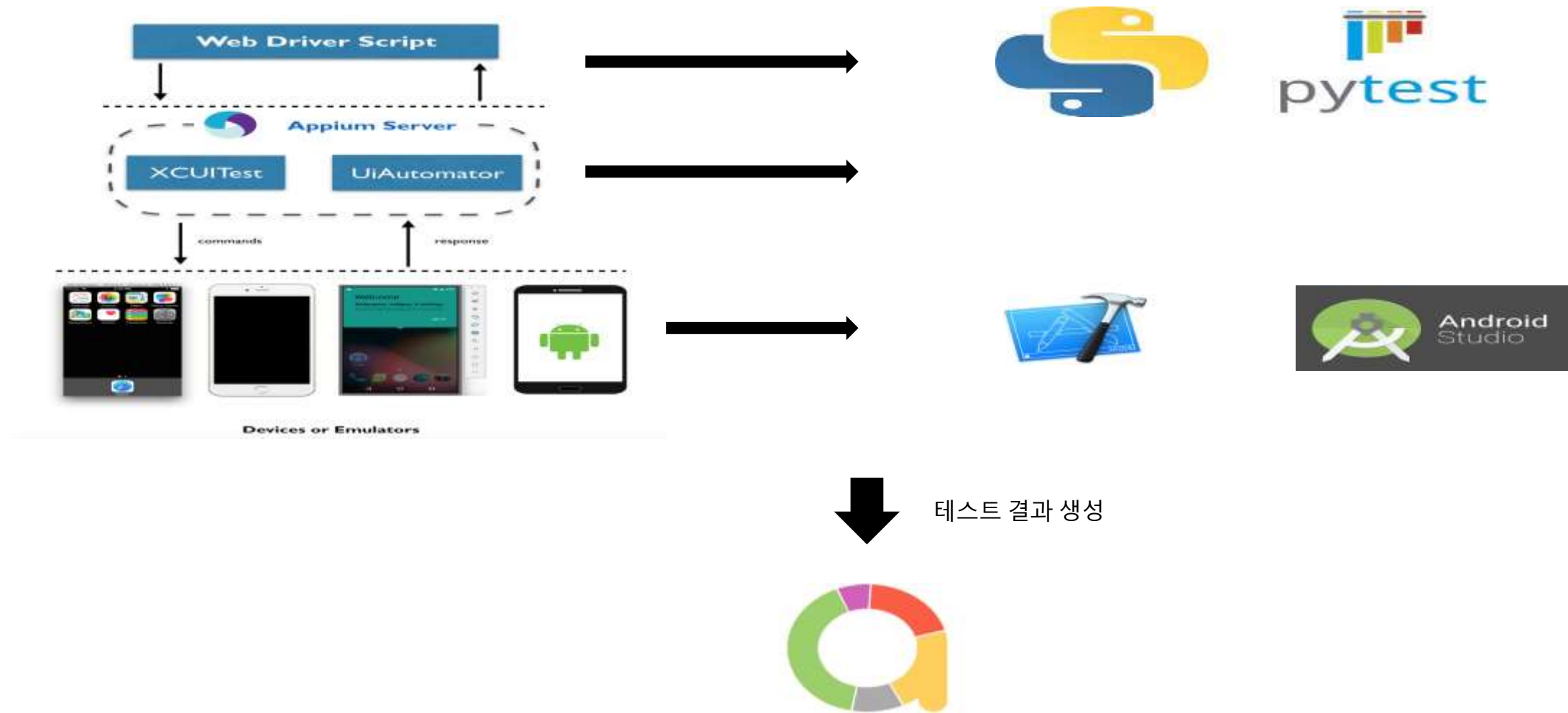
class Reviews :
    url = ''
    reviews = []

    # timeout_limit : 해당 사이트 #url로 target_url에 해당이 없으면, 사이트가 연결이 안되고 건너뛰고 크롤링을 함
    timeout_limit = 5
    crawled_reviews_count = -2

    def __init__(self, url):
        self.url = url

```

앱 자동화테스트 구조

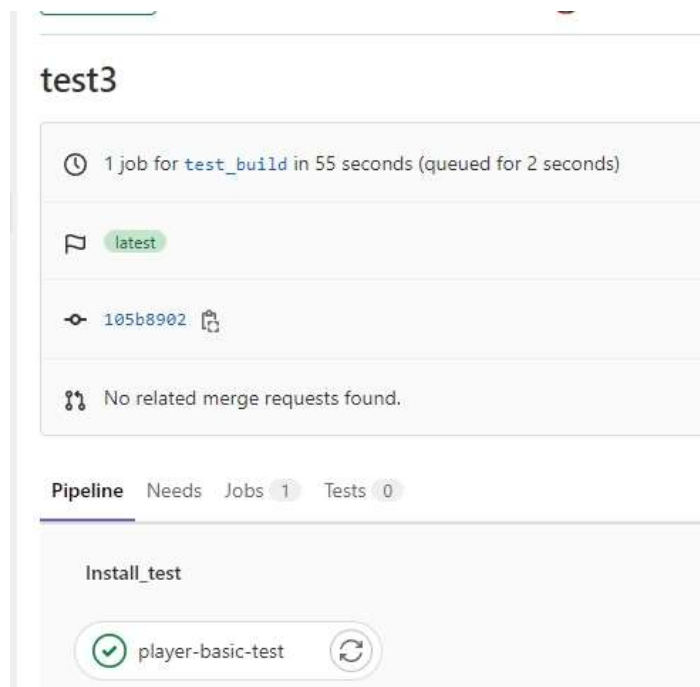


개요

- GitLab CI/CD 기반으로 자동화테스트 CI Pipeline 구축
- 아프리카TV앱 UI 테스트 진행

주요 내용

- APK 업로드시 UI 테스트를 자동으로 수행하는 Pipeline 구축
- 다양한 해상도, OS 기기 별로 동시에 호환성 테스트 진행



보안 취약점 분석 업무

Projects

개요

- 넥슨 모바일게임 출시 전 보안 취약점 분석 업무
- Android와 유니티 취약점 위주로 진행

주요 내용

- Android 앱 클라이언트 사이드 취약점 검증 업무
 - 메모리 변조, 스피드해킹, 루팅, 탈옥, 앱 리패키징, 디컴파일
- 서버 사이드 취약점 검증 업무
 - WireShark를 이용한 네트워크 패킷 난독화 여부 검증

```
21 public void OnSkillCondition(MyVal condition As UnitSkillCondition, MyVal exchangeRateCallback As Action<UnitSkillCondition, Boolean>)
22 {
23     if (Me.effectiveConditions.Contains(condition)) Then
24         Return
25     End If
26     Dim conditionData As SkillActiveStrategyComponent.ConditionData = New SkillActiveStrategyComponent.ConditionData() With
27     {
28         .onChangeState = onExchangeRateCallback
29     }
30     Me.effectiveConditions.Add(condition, conditionData)
31     Me.OnExchangeRateCallback(condition)
32 End Sub
33
34 Protected Function IsSatisfyCompareValue(MyVal value As Single, MyVal condition As UnitSkillCondition) As Boolean
35 Dim (single) As Single = Single.Parse(condition.compareValue)
36 Select Case condition.compareOp
37     Case UnitSkillCompareOp.GreaterThanSingle
38         Return value >= (single)
39     Case UnitSkillCompareOp.LessThanSingle
40         Return value <= (single)
41     Case UnitSkillCompareOp.Equals
42         Return value = (single)
43     Case UnitSkillCompareOp.NotEquals
44         Return value <> (single)
45 End Select
46 Return False
47 End Function
48
49 Protected Function IsSatisfyCompareValue(MyVal value As Integer, MyVal condition As UnitSkillCondition) As Boolean
50 Dim num As Integer = Integer.Parse(condition.compareValue)
51 Select Case condition.compareOp
52     Case UnitSkillCompareOp.GreaterThanSingle
53         Return value >= num
54     Case UnitSkillCompareOp.LessThanSingle
55         Return value <= num
56     Case UnitSkillCompareOp.Equals
57         Return value = num
58     Case UnitSkillCompareOp.NotEquals
59         Return value <> num
60 End Function
```



CQA "E%0*.1%>uÇÄ_CÑu

Experience

- SSAC 라이징프로그래머 iOS개발자 과정
- 많은 협업 경험

개요

- 서울산업진흥원(SSAC) iOS개발자 부트캠프 과정 수료 [\[후기글\]](#)

주요 내용

- 자기주도적 학습 이후에 클론코딩 & 팀프로젝트 진행
- 팀프로젝트는 디자이너, 기획자, 서버개발자가 팀을 매칭해서 진행
- 데모데이를 통해 최종 프로젝트 발표



합업합동

Experience

개요

- 여러 프로젝트, 유관부서, 관계사와 협업하면서 다양한 협업툴 활용

주요 내용

- JIRA, Slack, Confluence, GitLab 업무 환경에 대한 높은 숙련도
- Git Flow 활용 (PR, Merge, Branch 관리)
- 업무 프로세스에 필요한 다양한 내용 문서화 진행
- 비즈니스 매너 숙지

①. 작성 방법					
<ul style="list-style-type: none"> 담당자 : @ 해시태그 기능을 이용해서 이름 입력 최종 경산일 : 가장 최근에 페이지를 경산한 날짜 경산 사람 : 페이지 경산 필요 시점 비고 : 담당자 변경, 페이지 삭제 등의 특이사항 발생시 비고란에 기입 					
분류1	분류2	분류3	담당자	최종 경산일	경산 사항
0. 공통	1) 일정		@유우동 [melona]	2018.11.14	페이지 일정이 정해졌을 시
1. QA 프로세스	1) JIRA 사용법		@이태현 [jogus]	2018.06.27	JIRA 기능 업데이트 시
	2) 개발 JIRA 사용법		@이태현 [jogus]	2018.06.03	버그리포팅 프로세스 변경 시
	3) 버그리포팅 가이드		@이태현 [jogus]		
	4) 이슈 처리 규칙		@오혜진 [enoughon]	2018.09.18	규칙 개정 시
2. 테스트 환경	1) 테스트 계획		@이태현 [jogus]		테스트 계획 정보 반영 시
		테스트 계획 사용법	@이태현 [jogus]		테스트 계획 정보 반영 시
	2) 결함 세팅		@오혜진 [enoughon]		지정 담당자, 계획 변경 시
	3) 테스트용 앱 & 치트		담당자 전체		신규 Tip, 치트 추가 시
3. 테스트 가이드	1) 캔버스 테스트	작용 배크	@이태현 [jogus]		신규 배크 추가 시

[illegible]

E-MAIL yshan4329@gmail.com

Github <https://github.com/RobertHan96>

Blog <https://blog.naver.com/yshan1008>

감사합니다