

# Connect[Sticky]

with Webpack



# 목차

1

Connect[Sticky]

2

프로젝트 개요

3

설계 문서

4

주요 정책

5

Webpack

# Connect

Sticky, ChatBot... 등의 사용자를 위한 맞춤 서비스

# Sticky

사용자의 페이지, 행동에 따른 추천을 해주기 위한 서비스

# Connect

Sticky

ChatBot

Item

Page

Page

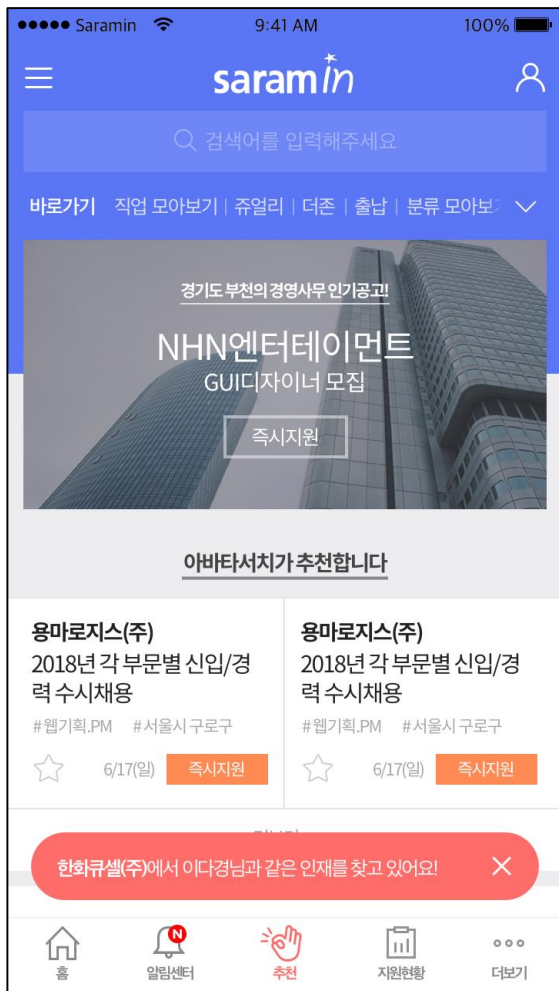
Page

Situation

Situation

Situation

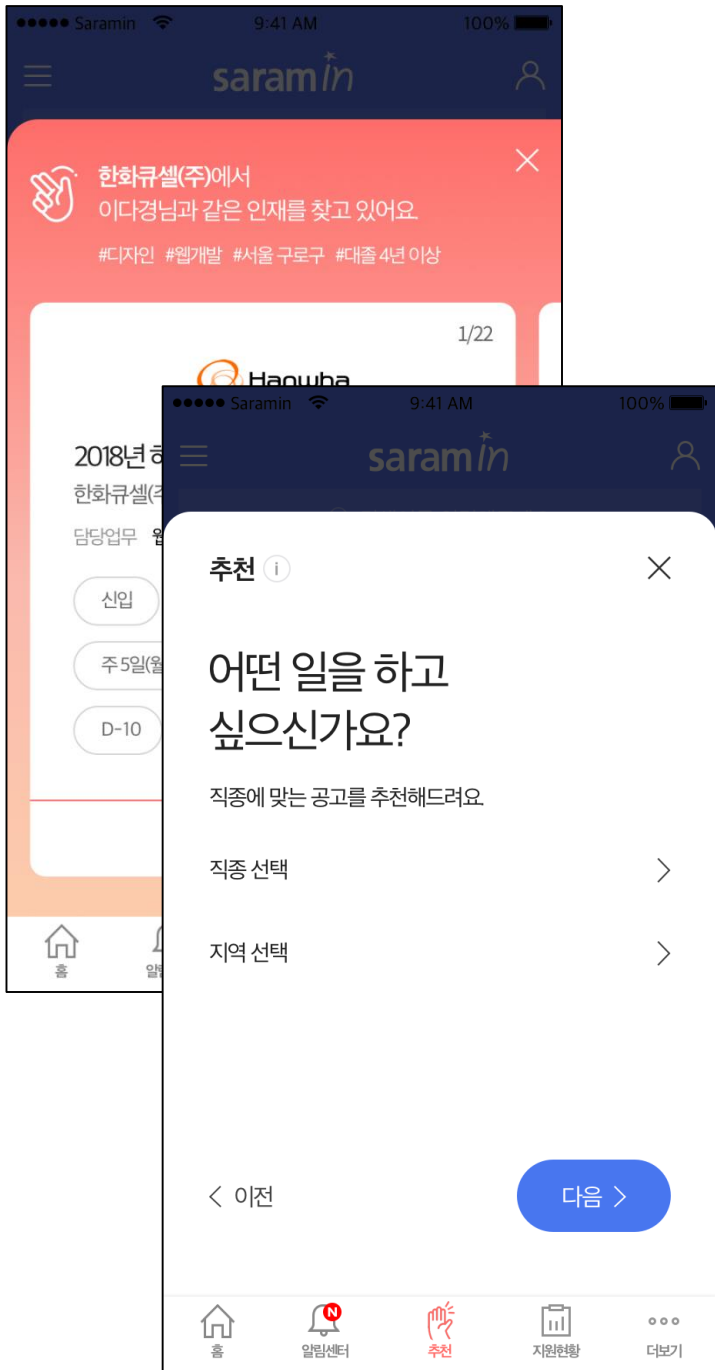
아이템, 페이지, 행동에 따라 각각 다른 이벤트를 처리



# 모바일 전체 페이지 포함

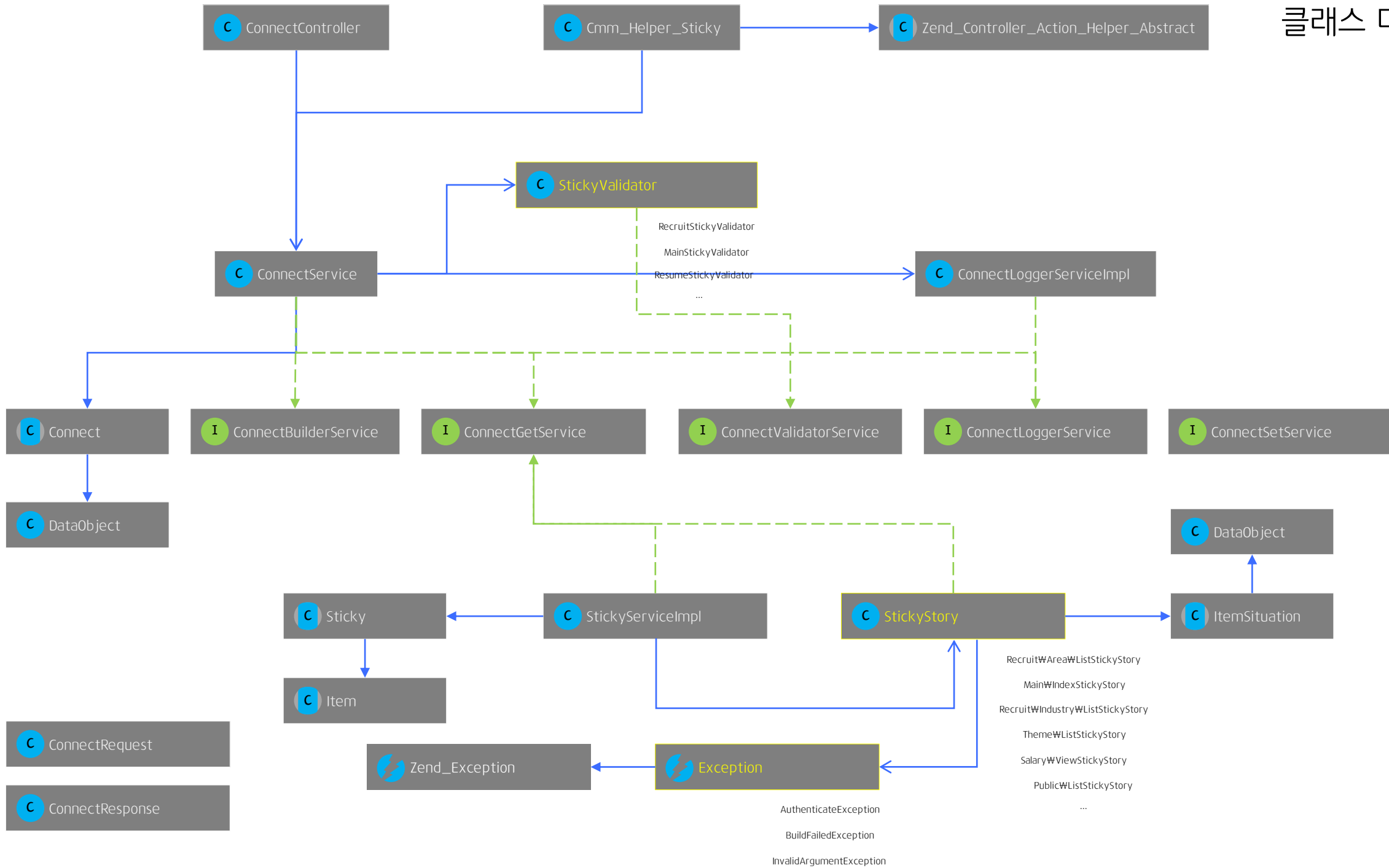
- ※ 네이티브 하단 tab\_bar 제거 → 웹에서 tab\_bar 노출로 변경
- ※ tab\_bar 구조 변경 (추천 아이콘 추가)
- ※ btn\_back, btn\_home, btn\_top 조건부 제거
- ※ 공통적으로 모바일 전체 포함시키는 Sticky 뷰파일 추가 (Middleware)
- ※ 사용자 현재 페이지별 추천 문구 노출 (아이콘 노출 변경 포함)
- ※ 사용자 행동별 추천 문구 노출 (이전 페이지 행동 포함)
- ※ 추천 문구 영역 클릭할 경우 추천 레이어 노출 (브라우저 히스토리 X)
- ※ 최초 진입 사용자를 위한 기본정보 입력 프로세스 추가 (별이 + Sticky)
- ※ ~~입사지원 완료 프로세스 변경~~ CSS 변경으로 처리
- ※ 확장성 고려 (챗봇, 콘텐츠 확장, IA 변경 및 간소화 예정)

# 모바일 전체 페이지 포함



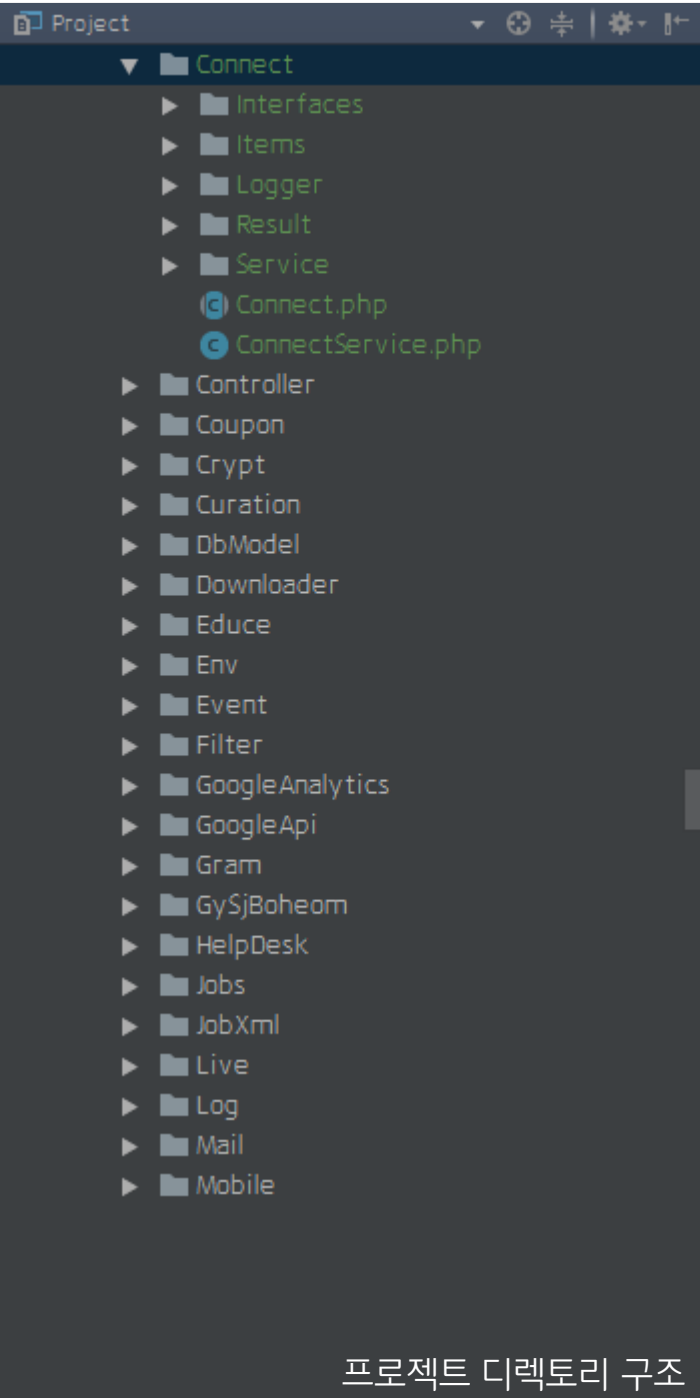
- ※ 네이티브 하단 tab\_bar 제거 → 웹에서 tab\_bar 노출로 변경
- ※ tab\_bar 구조 변경 (추천 아이콘 추가)
- ※ btn\_back, btn\_home, btn\_top 조건부 제거
- ※ 공통적으로 모바일 전체 포함시키는 Sticky 뷰파일 추가 (Middleware)
- ※ 사용자 현재 페이지별 추천 문구 노출 (아이콘 노출 변경 포함)
- ※ 사용자 행동별 추천 문구 노출 (이전 페이지 행동 포함)
- ※ 추천 문구 영역 클릭할 경우 추천 레이어 노출 (브라우저 히스토리 X)
- ※ 최초 진입 사용자를 위한 기본정보 입력 프로세스 추가 (별이 + Sticky)
- ※ ~~입사지원 완료 프로세스 변경~~ CSS 변경으로 처리
- ※ 확장성 고려 (챗봇, 콘텐츠 확장, IA 변경 및 간소화 예정)

# 클래스 다이어그램

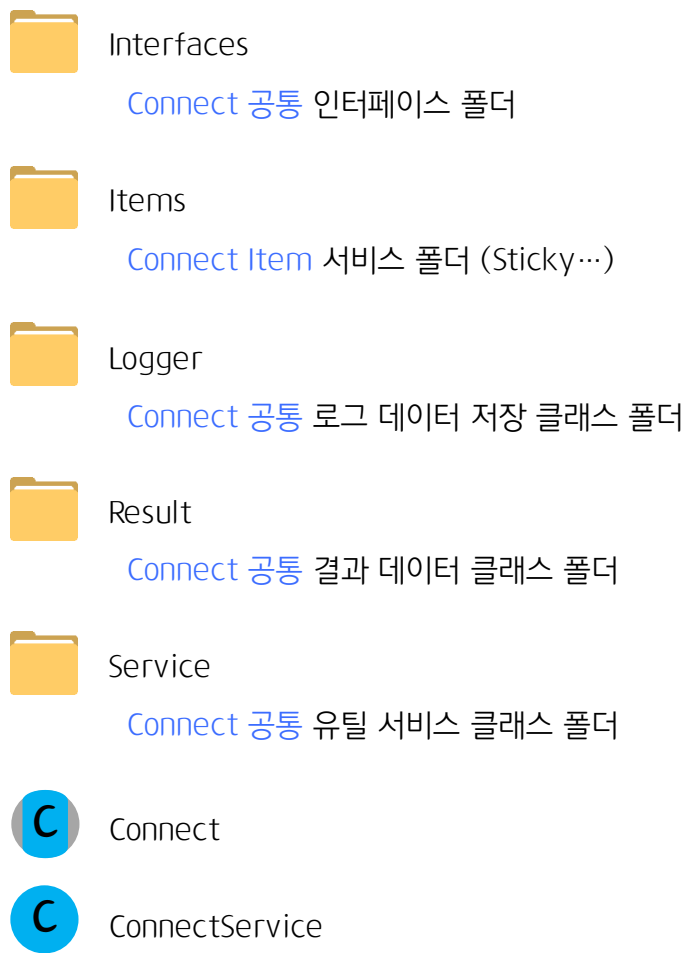




## 프로젝트 디렉토리 구조



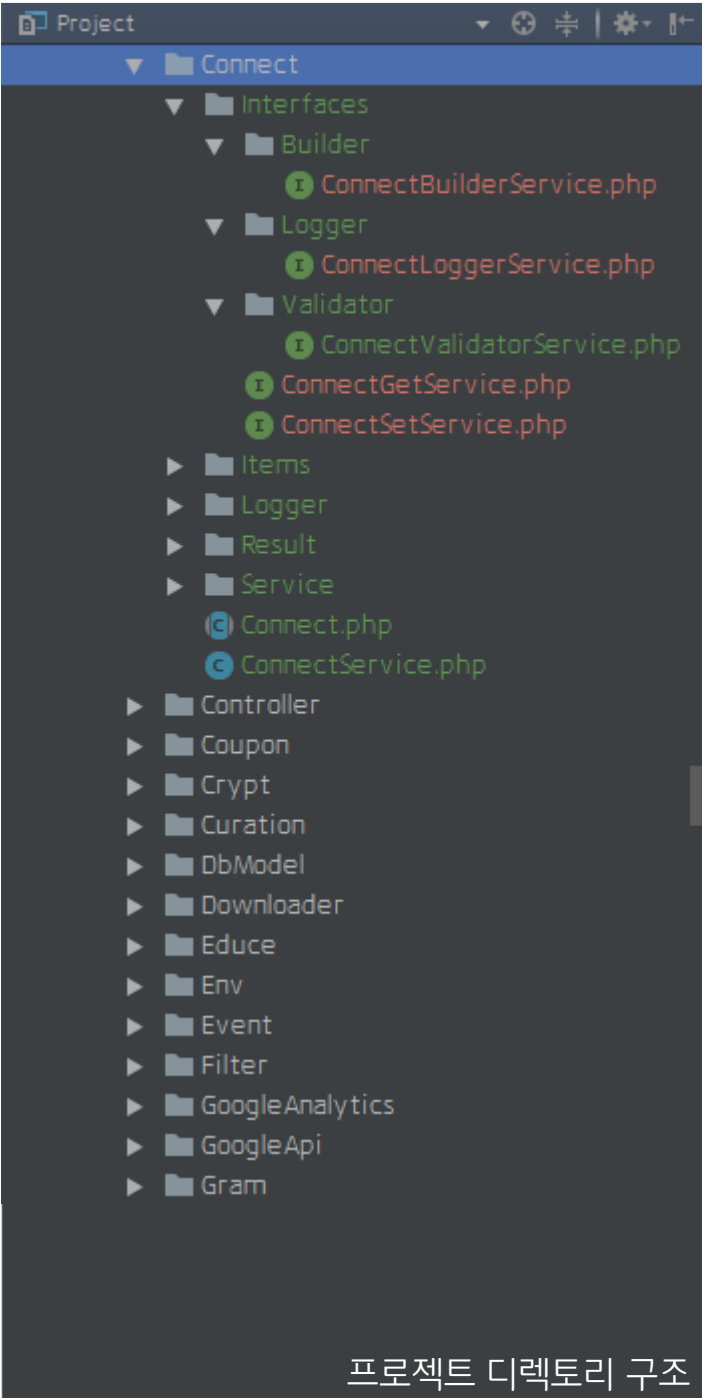
프로젝트 디렉토리 구조



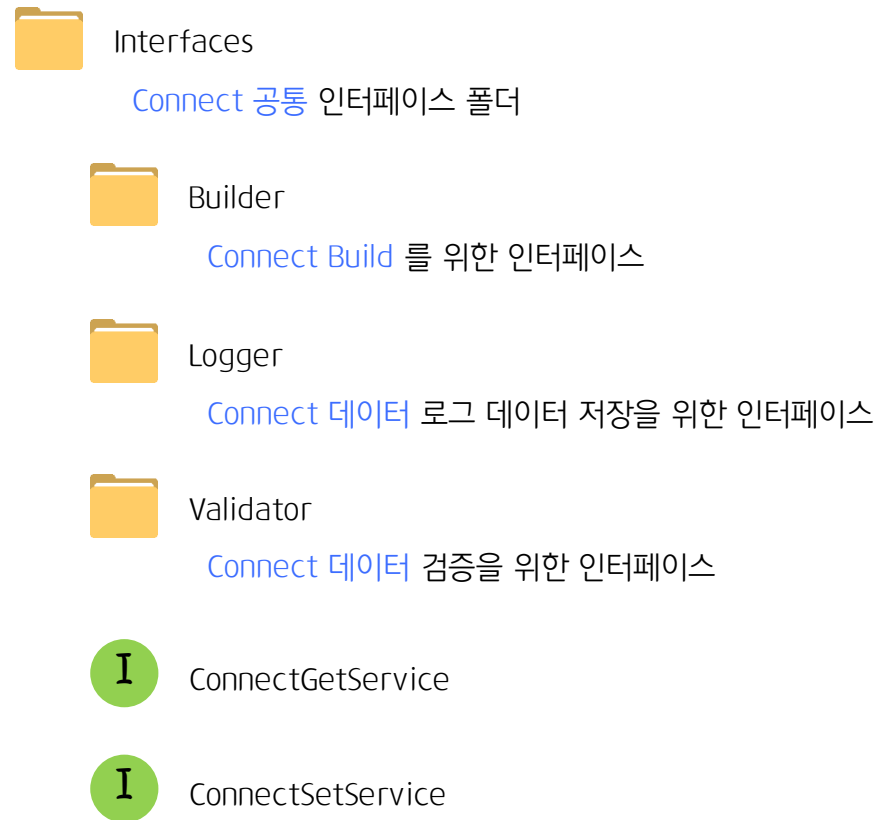
### ConnectService

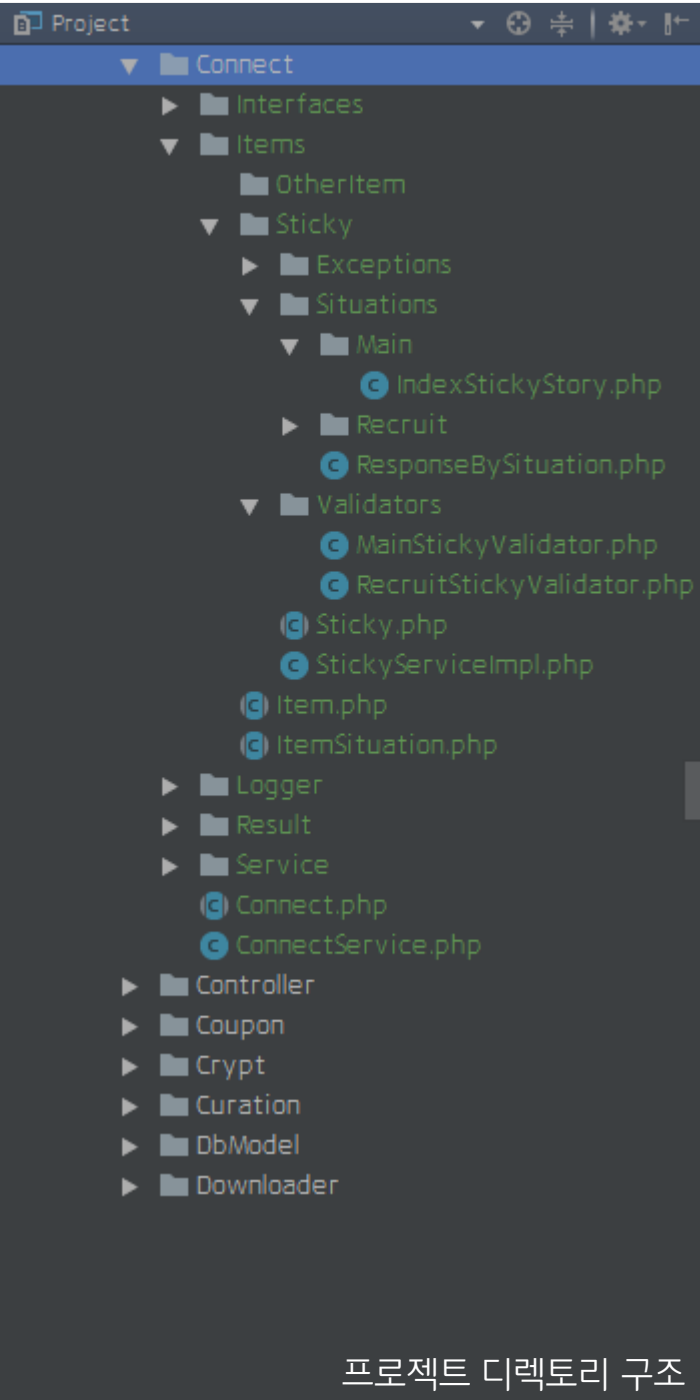
```
+ getItem();  
+ setPage();  
+ setIdentity();  
+ setSituation();  
+ build();  
+ isBuilt();  
+ validate();  
+ get();  
+ logging();  
  
- buildAfterProcess();  
- throwExceptionIfNotExistsClass  
- throwExceptionIfNotBuilt()
```

## 프로젝트 디렉토리 구조



프로젝트 디렉토리 구조





프로젝트 디렉토리 구조



구분	페이지	비고
main_index	'/', '/index'	
recruit_area_list	'/location-job/recently-list', '/location-job/domestic', '/location-job/domestic-list', '/location-job/overseas', '/location-job/overseas-list'	
recruit_job_list	'/job-industry/recently-list', '/job-industry/job', '/job-industry/job-list'	
recruit_industry_list	'/job-industry/recently-list', '/job-industry/ids', '/job-industry/ids-list',	
recruit_special_list	'/special-job/all', '/special-job/all-list'	
recruit_search_list	'/search'	
recruit_publication_list	'/group-job/calendar', '/group-job/top1000', '/group-job/major', '/group-job/major-list', '/group-job/top-group', '/group-job/live-list', '/group-job/schedule', '/job-fair/schedule', '/real-coverletter', '/job-fair', '/career-information/job-interview-list', '/pass-data/search-spec-self', '/successful-interview', '/interview-knowhow', '/aftermote-aptitude'	

구분	페이지	비고
myhome_index	'/my-service'	
resume_list	'/resume'	
apply_list	'/apply/state'	
read_list	'/lately-job/list'	
scrap_list	'/scrap-job/list', '/scrap-job/calendar'	
favor_list	'/favor-company/list'	
suited_list	'/member/suited-recruit-mail/list'	
company_info	'/job-search/company-info-view'	
recruit_view	'/job-search/view'	
contents_view	'/real-coverletter/view'	
theme_view	'/curation/view'	
theme_list	'/curation'	
salary_list	'/salaries/total-salary/list'	
salary_view	'/salaries/total-salary/view'	
tool_view	'/helper-tool/unemp', '/helper-tool/retire', '/helper-tool/holi', '/helper-tool/gradu', '/helper-tool/grade-converter', '/helper-tool/text-count', '/helper-tool/salary-calculator'	

2018. 09. 04 기준 적용 페이지



## Situations

구분별로 클래스를 생성하여 이벤트를 처리

※ recruit\_special\_list → **RecruitSpecialListStickySituation Class**

구분 코드	텍스트	링크	액션	옵션	비고
enter	'반가워요. 사람입니다.' '사람인에 오신걸 환영해요.'	"	'request_sticky'	'code' => 'enter', 'icon' => 'hello'	
first	'당신이 누구인지 간단히 알려주세요.'	'/connect/welcome-star'	'popup_layer'	'code' => 'first', 'icon' => 'hello'	
main	'오늘의 추천 공고를 확인하세요.'	'/connect/today-recommend-recruit'	'popup_layer'	'code' => 'main', 'icon' => 'default'	
register	'회원가입 완료! 지금 추천 공고를 확인하세요.'	'/connect/today-recommend-recruit'	'popup_layer'	'code' => 'register', 'icon' => 'complete'	
application	'지원하신 공고와 비슷한 공고를 지금 확인하세요.'	'/connect/similar-recommend-recruit'	'popup_layer'	'code' => 'application', 'icon' => 'complete'	
resume	'이력서를 분석! 최적의 공고를 확인하세요.'	'/connect/recommend-recruit?type=member_resume'	'popup_layer'	'code' => 'resume', 'icon' => 'complete'	
job_recruit_list	'추천 공고를 확인하세요.'	'/connect/recommend-recruit?type=job'	'popup_layer'	'code' => 'job_recruit_list', 'icon' => 'default'	
area_recruit_list	'추천 공고를 확인하세요.'	'/connect/recommend-recruit?type=area'	'popup_layer'	'code' => 'area_recruit_list', 'icon' => 'default'	
public_recruit_list	'새로운 추천 공채를 확인하세요.'	'/connect/recommend-recruit?type=public'	'popup_layer'	'code' => 'public_recruit_list', 'icon' => 'default'	
recruit_view	'이런 공고는 어떠신가요?'	'/connect/recommend-recruit?type=default'	'popup_layer'	'code' => 'recruit_view', 'icon' => 'default'	
content_list	'나에게 가장 맞는 공고를 확인하세요.'	'/connect/recommend-recruit?type=content'	'popup_layer'	'code' => 'content_list', 'icon' => 'default'	
content_view	'나에게 가장 맞는 공고를 확인하세요.'	'/connect/recommend-recruit?type=content'	'popup_layer'	'code' => 'content_view', 'icon' => 'default'	
scrap	'지금 스크랩하기 가장 적합한 공고입니다.'	'/connect/recommend-recruit?type=scrap'	'popup_layer'	'code' => 'scrap', 'icon' => 'suggestion'	
company_view	'당신을 찾고 있는 다른 기업 공고들입니다.'	'/connect/recommend-recruit?type=company_view'	'popup_layer'	'code' => 'company_view', 'icon' => 'default'	
favor_list	'관심 기업과 유사한 공고입니다.'	'/connect/recommend-recruit?type=favor'	'popup_layer'	'code' => 'favor_list', 'icon' => 'suggestion'	
search	'검색어 추천 공고를 확인하세요.'	'/connect/recommend-recruit?type=search'	'popup_layer'	'code' => 'search', 'icon' => 'suggestion'	

2018. 09. 04 기준 추천 문구

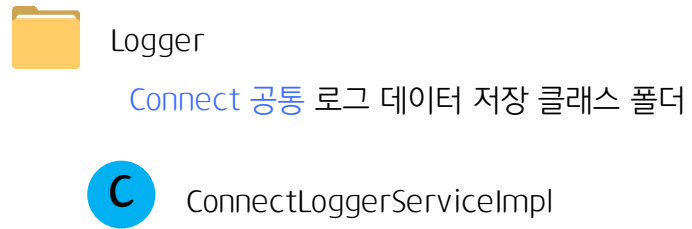
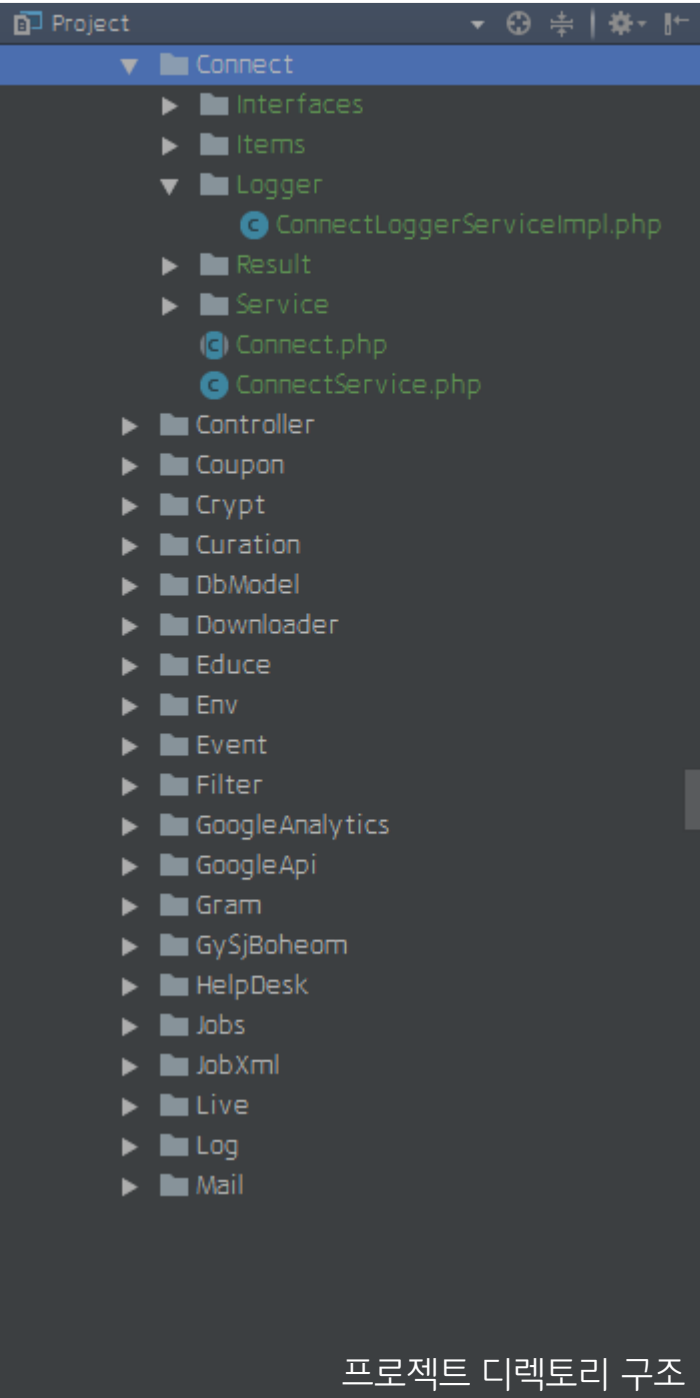


## Situations

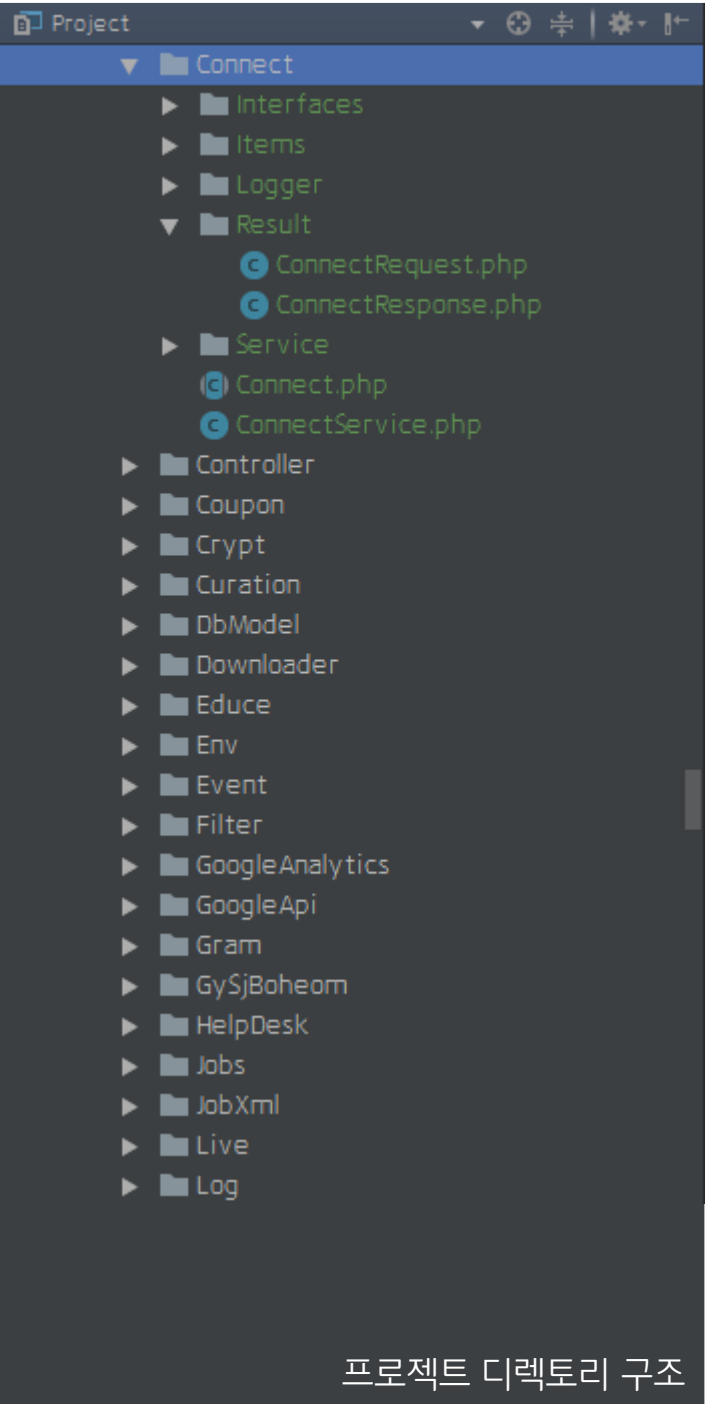
구분별로 텍스트와 링크, 액션 값, 추천 아이콘 등을 변경해줘야 함.

※ 관리자 페이지에서 실시간으로 변경할 수 있어야 함/ 파일 (JSON) 으로 관리

## 프로젝트 디렉토리 구조



프로젝트 디렉토리 구조



프로젝트 디렉토리 구조



Result

Connect 공통 결과 데이터 클래스 폴더



ConnectRequest

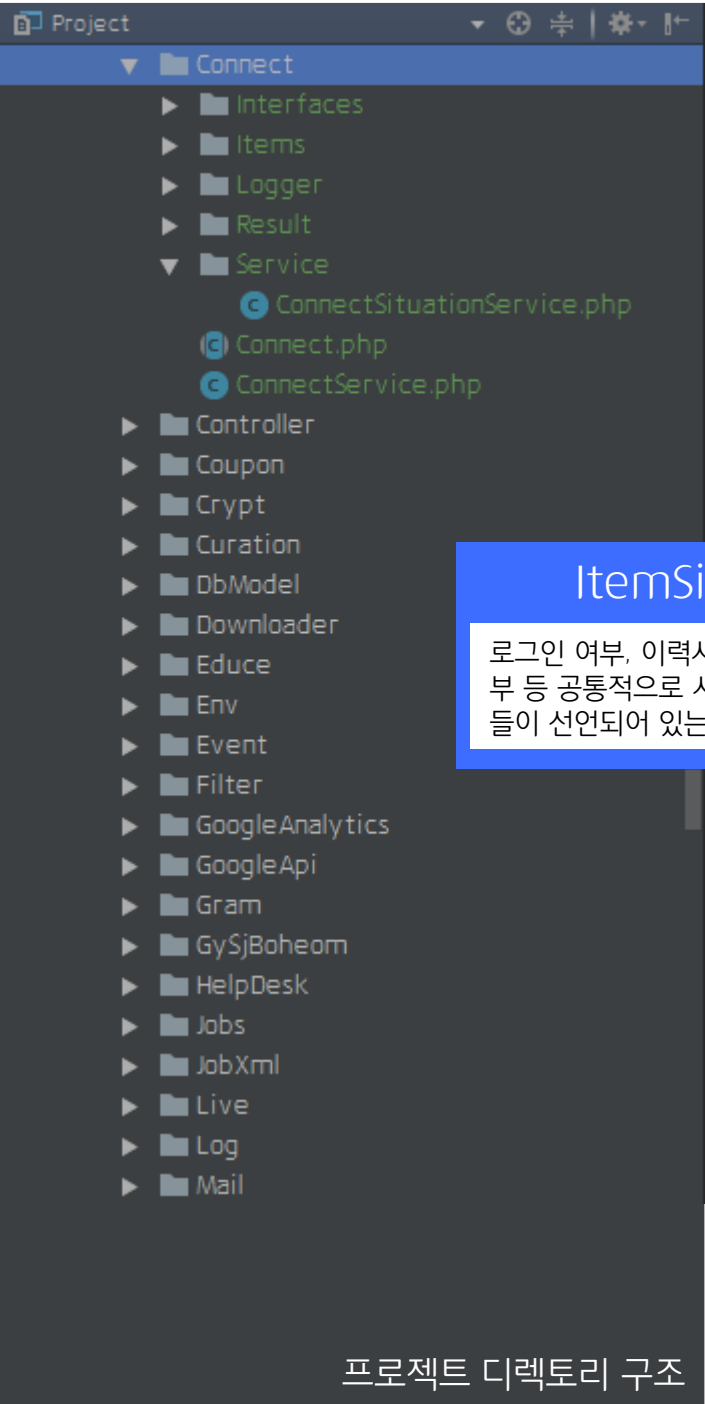


ConnectResponse

<공통> 결과 데이터

```
result = [
    'request' => ConnectRequest($attributes),
    'response' => ConnectResponse($message, $code, $data)
];

result = [
    request => [
        'item'    => ['sticky', 'chatbot', ...],
        'page'    => ['main_index', 'recruit_area_list', ...],
        /* 입력받지 못했을 경우 또는 잘못 입력한 경우 Service 단에서 자동 판단하여 입력 처리 */
        'situation' => ['apply', 'join', 'resume', ...],
        /* 입력받지 못했을 경우 또는 잘못 입력한 경우 Service 단에서 자동 판단하여 입력 처리 */
        'identify' => identify
    ],
    response => [
        'code'     => [400, 404, 200, ...],
        'status'   => ['OK', 'No Content', 'Bad Request', ...],
        'message'  => ['성공', '콘텐츠 없음', '잘못된 요청', ...],
        /* IANA 가 지정한 HTTP STATUS CODE */
        'data'     => [
            'text'   => '당신이 누구인지 간단히 알려주세요.',
            'link'   => 'http://www.saramin.co.kr',
            'action' => 'popup_layer',
            'options' => ''
        ]
    ]
];
```



Service

Connect 공통 유틸 서비스 클래스 폴더



ConnectSituationService

다양한 사용자 데이터를 반환 & 검증해주는 함수를 작성해놓은  
<공통> ItemSituation abstract class 를 extends 한 뒤에 함수를 호출해서 사용

**ItemSituation**  
로그인 여부, 이력서 여부, 입사지원 여부 등 공통적으로 사용할 수 있는 함수들이 선언되어 있는 abstract 클래스

```
class IndexStory extends ItemSituation implements ConnectGetService
{
    public function __construct($attributes = [])
    {
        parent::__construct($attributes);
    }

    public function get()
    {
        if (!$this->isAuthenticated()) {
            /* @Situation 비회원 */
            return $this->getSituationResponse(self::FIRST_VISITED_NON_MEMBER_SITUATION);
        }

        if (!$this->isExistsResume()) {
            /* @Situation 회원, 이력서 미존재 */
            return $this->getSituationResponse(self::NOT_EXISTS_RESUME_MEMBER_SITUATION);
        }

        if ($this->isAppliedThreeDays()) {
            /* @Situation 회원, 이력서 존재, 3일내 입사지원 기록 보유 */
            return $this->getSituationResponse(self::IS_APPLIED_BY_THREE_DAYS_SITUATION);
        }

        if (!$this->isAppliedThreeDays() && $this->getAvatarAttribute('직종')) {
            /* @Situation 회원, 이력서 존재, 3일내 입사지원 기록 미보유, 아바타 직종 속성 보유 */
            return $this->getSituationResponse(self::JIKJONG_SITUATION);
        }

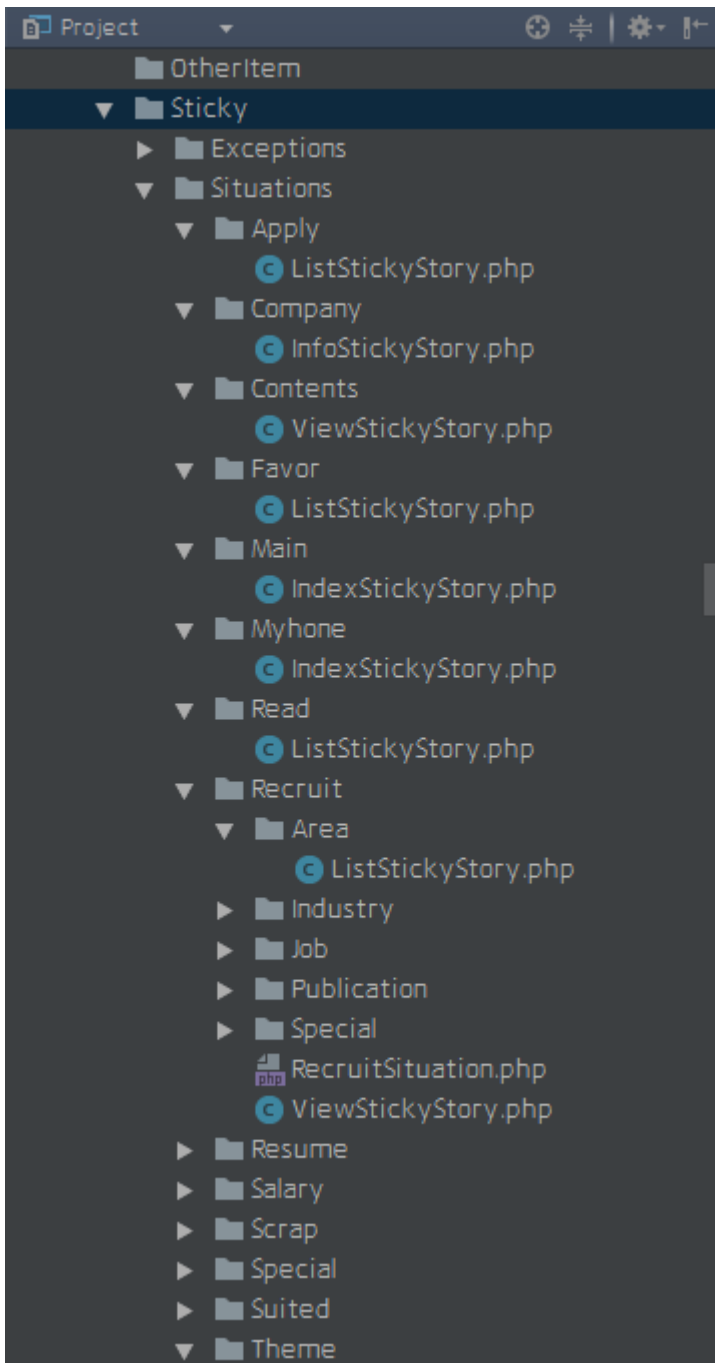
        if (!$this->isAppliedThreeDays() && $this->getAvatarAttribute('지역')) {
            /* @Situation 회원, 이력서 존재, 3일내 입사지원 기록 미보유, 아바타 지역 속성 보유 */
            return $this->getSituationResponse(self::AREA_SITUATION);
        }

        if (!$this->isAppliedThreeDays()) {
            /* @Situation 회원, 이력서 존재, 3일내 입사지원 기록 미보유, 아바타 직종, 지역 속성 미보유 */
            return $this->getSituationResponse(self::NO_AREA_AND_NO_JIKJONG_SITUATION);
        }

        return $this;
    }
}
```

```
class ResponseBySituation
{
    public static $response = [
        'enter' => [
            'text' => [
                '반가워요. 사람입니다.',
                '사람인에 오신걸 환영해요.'
            ],
            'link' => '',
            'action' => 'nothing',
            'options' => ''
        ],
        'first' => [
            'text' => '당신이 누구인지 간단히 알려주세요.',
            'link' => 'http://92xrdv.mmt.pdev2.saramin.co.kr/connect/welcome-star',
            'action' => 'popup_layer',
            'options' => ''
        ],
        'main' => [
            'text' => '오늘의 추천 공고를 확인하세요.',
            'link' => '/job-search/view?rec_id=27607773&t_ref=myhome&t_ref_content=recently_recruit',
            'action' => 'popup_layer',
            'options' => ''
        ],
        'register' => [
            'text' => '회원가입 완료! 지금 추천 공고를 확인하세요.',
            'link' => '/job-search/view?rec_id=27607773&t_ref=myhome&t_ref_content=recently_recruit',
            'action' => 'popup_layer',
            'options' => ''
        ],
        'application' => [
            'text' => '지원하신 공고와 비슷한 공고를 지금 확인하세요.',
            'link' => '/job-search/view?rec_id=27607773&t_ref=myhome&t_ref_content=recently_recruit',
            'action' => 'popup_layer',
            'options' => ''
        ],
        'resume' => [
            'text' => '이력서를 분석! 최적의 공고를 확인하세요.',
            'link' => '/job-search/view?rec_id=27607773&t_ref=myhome&t_ref_content=recently_recruit',
            'action' => 'popup_layer',
            'options' => ''
        ]
    ];
}
```





abstract class ItemSituation extends DataObject {

protected boolean 공통 Situation Response 코드();

protected boolean 로그인 여부 체크();

protected boolean 외부 페이지 진입 체크();

protected boolean 이력서 존재 여부 체크();

protected boolean 3일간 입사지원 존재 여부 체크();

protected boolean 개인화 속성 존재 여부 체크();

여러가지 공통적으로 쓸 수 있는 함수를 제공

(조합하여 사용 또는 새롭게 정의하여 사용)

```

class IndexStickyStory extends ItemSituation implements ConnectDataService {
{
    public function __construct($attributes = [])
    {
        parent::__construct($attributes);
    }

    public function get()
    {
        if ($this->isAuthenticated()) {
            /* Situation 비회원 */
            return $this->getSituationResponse(self::FIRST_VISITED_NON_MEMBER_SITUATION);
        }

        if ($this->isExistsResume()) {
            /* Situation 회원, 이력서 미존재 */
            return $this->getSituationResponse(self::NOT_EXISTS_RESUME_MEMBER_SITUATION);
        }

        if ($this->isAppliedThreeDays()) {
            /* Situation 회원, 이력서 존재, 3일내 입사지원 기록 보유 */
            return $this->getSituationResponse(self::IS_APPLIED_BY_THREE_DAYS_SITUATION);
        }

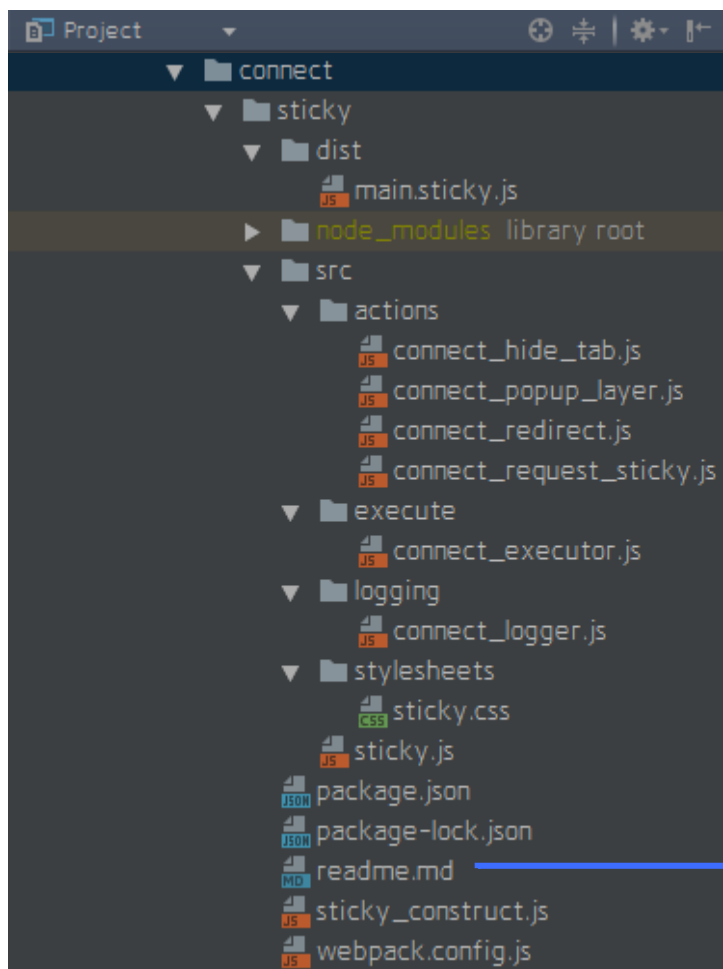
        if ($this->isAppliedThreeDays() && $this->getAvatarAttribute('직종')) {
            /* Situation 회원, 이력서 존재, 3일내 입사지원 기록 미보유, 아바타 직종 속성 보유 */
            return $this->getSituationResponse(self::JIKJONG_SITUATION);
        }

        if ($this->isAppliedThreeDays() && $this->getAvatarAttribute('지역')) {
            /* Situation 회원, 이력서 존재, 3일내 입사지원 기록 미보유, 아바타 지역 속성 보유 */
            return $this->getSituationResponse(self::AREA_SITUATION);
        }

        if ($this->isAppliedThreeDays()) {
            /* Situation 회원, 이력서 존재, 3일내 입사지원 기록 미보유, 아바타 직종, 지역 속성 미보유 */
            return $this->getSituationResponse(self::NO_AREA_AND_NO_JIKJONG_SITUATION);
        }

        return $this;
    }
}
  
```

구분 코드 반환



J readme.md

## connect sticky webpack build

### 디렉토리

```

├── dist
│   └── main.sticky.js
├── src
│   ├── actions
│   │   ├── connect_hide_tab.js
│   │   ├── connect_popup_layer.js
│   │   ├── connect_redirect.js
│   │   └── connect_request_sticky.js
│   ├── execute
│   │   └── connect_executor.js
│   ├── logging
│   │   └── connect_logger.js
│   ├── stylesheets
│   │   └── sticky.css
│   └── sticky.js
├── sticky_construct.js
├── webpack.config.js
└── readme.md
    
```

### 빌드 방법

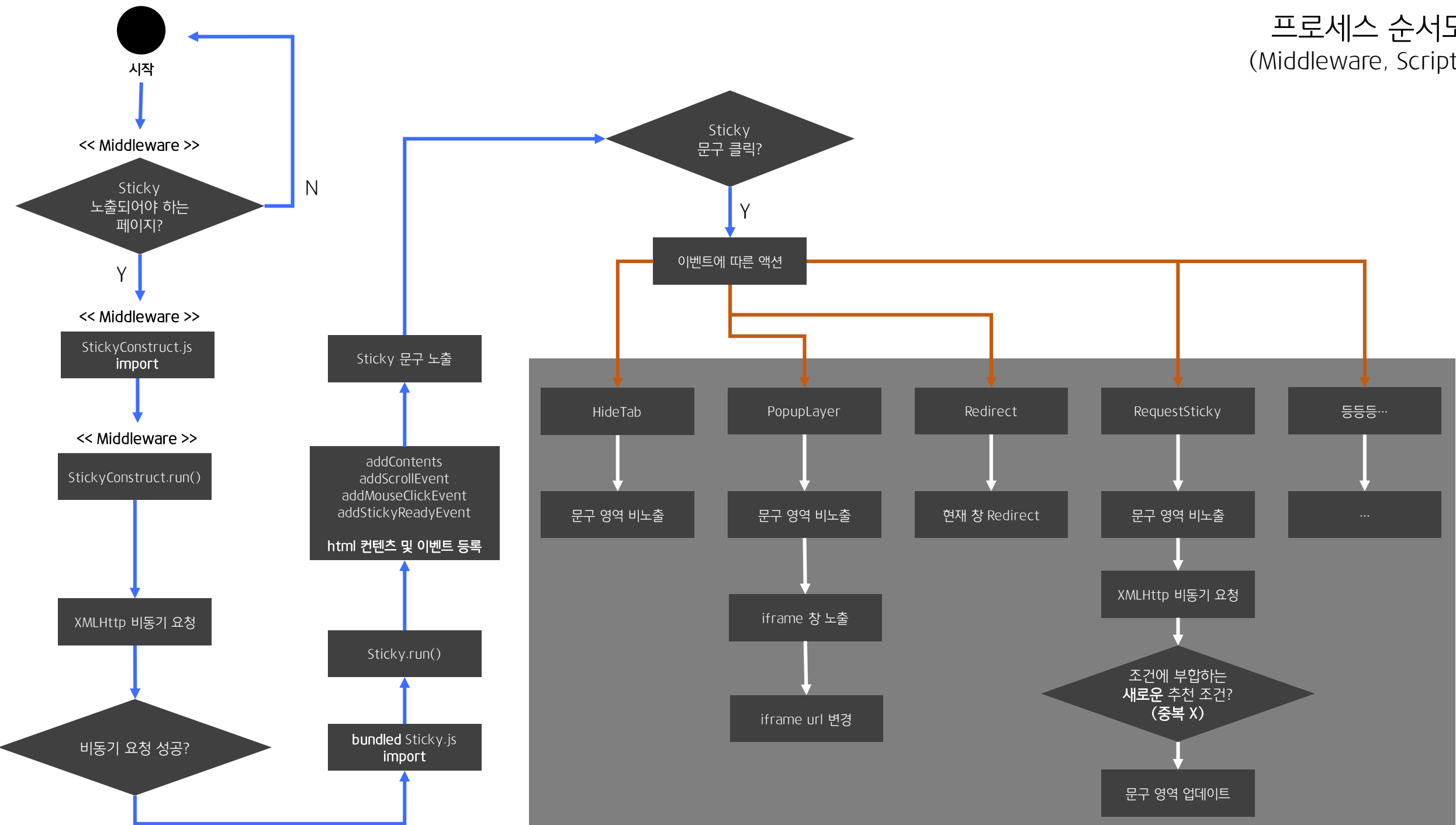
0. root path => webroot\mobile\js\connect\sticky\
1. node.js → npm 설치
2. webpack 설치
3. \node\_modules Excluded
4. install webpack-cli
5. loader 설치
  - npm install style-loader css-loader --save-dev
  - npm install file-loader --save-dev
  - npm install clean-webpack-plugin
  - npm install uglifyjs-webpack-plugin
  - npm install chunkhash-replace-webpack-plugin --save-dev

### 빌드

0. cd webroot\mobile\js\connect\sticky\
1. npm run-script build

빌드 후 **main.sticky.js** 파일로 서비스됨.

프로세스 순서도  
(Middleware, Script)

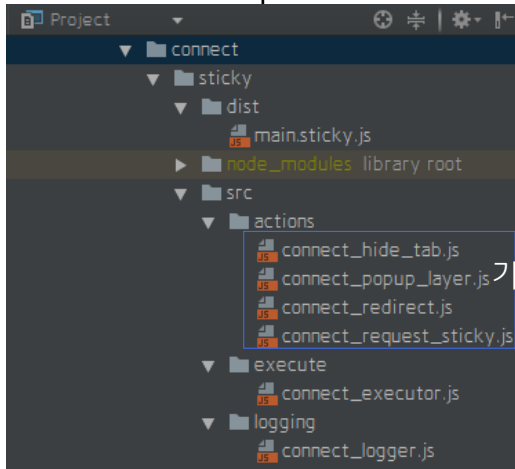


render view

import main.sticky.js



webpack bundling



기능별로 스크립트 파일 생성하여 사용



(Webpack Bundle) Javascript file

```
run : function (__d, options) {
  if (typeof __d.response.data === "undefined") {
    return false;
  }

  var response = __d.response.data;
  try {
    ConnectExecutor.doExecute(response, []);
  } catch (ex) {
    ConnectLogger.run(__d, ex);
  }
},
doExecute: function (response) {
  var action = response.action;

  if (!action || !this.__actions[action]) {
    Sticky.hideTabs();

    __waitProcess = false;
    return false;
  }

  switch (action) {
    case 'nothing':
      __waitProcess = false;
      break;

    case 'hide_tab':
      ConnectHideTab.run(response, []);
      break;

    case 'popup_layer':
      ConnectPopupLayer.run(response, []);
      break;

    case 'redirect':
      ConnectRedirect.run(response, []);
      break;
  }
}
```

스크립트 작성 방식

```
var Sticky = {
  __d : {},
  __version : '1.0.0',

  chaining: function (__d) {...},
  run: function (data) {
    if (!data) {
      return false;
    }

    Sticky.setData(data);

    if (!__isStickyEvent) {
      Sticky.addStickyEvent();
    }

    Sticky.addStickyReadyEvent();

    return false;
  },
  addStickyEvent: function () {...},
  addContents: function () {...},
  addScrollEvent: function () {
    $(window).scroll(function () {
      if (Sticky.isFloating()) {
        return false;
      }

      if (__waitProcess) {
        return false;
      }

      var data = Sticky.getData();
      if (!Sticky.isSuccess(data)) {
        return false;
      }

      Sticky.showTabs();
    });

    return this;
  },
  addMouseEvent: function () {
```

스크립트 작성 방식

## <공통> 요청 방법

```
$connector = new ConnectService('sticky');
$connector
->setPage($page)
/* 빈값일 경우 Service 단에서 임의 입력 처리 */
->setSituation($situations)
/* 빈값일 경우 Service 단에서 임의 입력 처리 */
->setIdentify($identify)
->build();

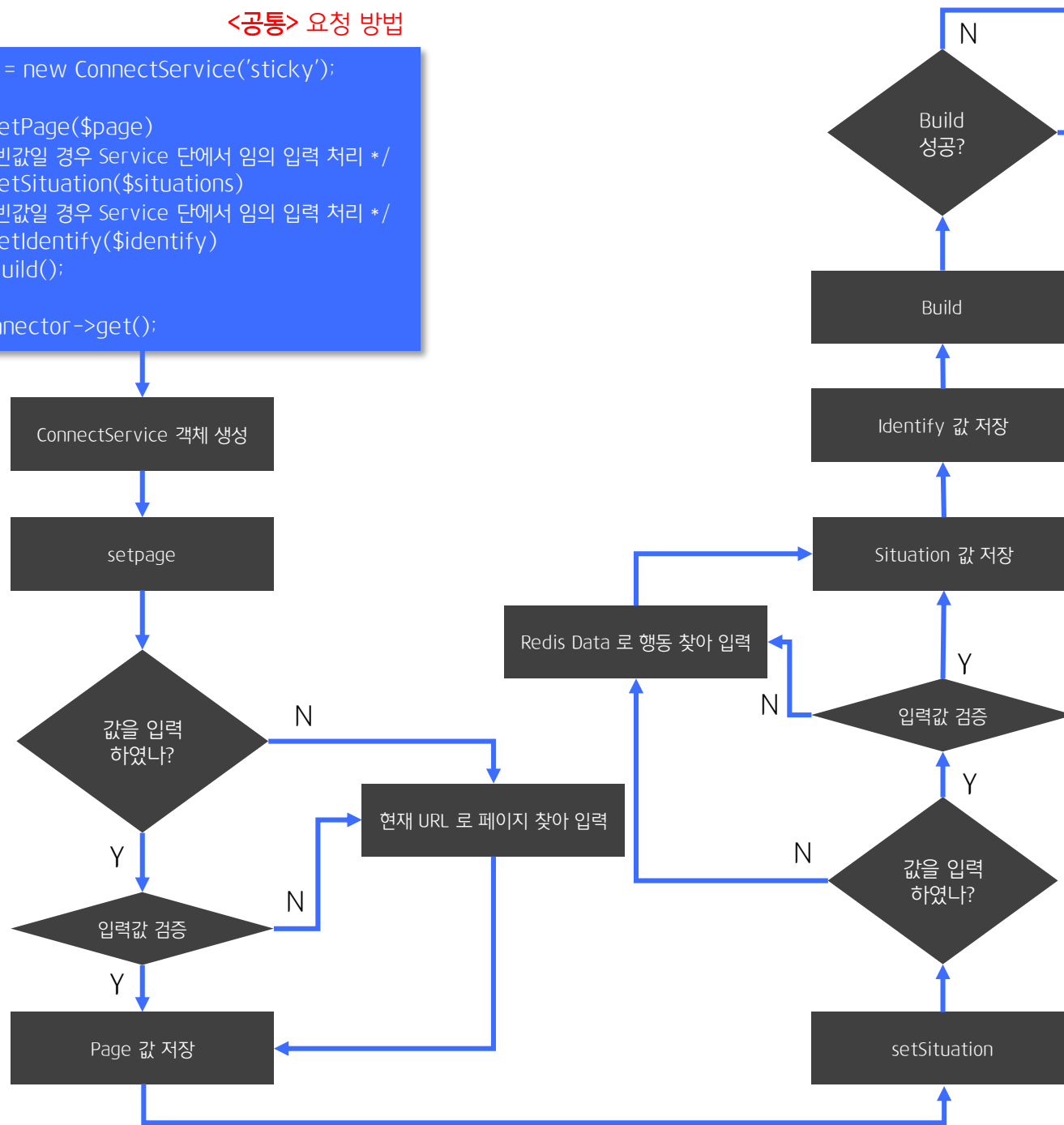
return $connector->get();
```

## 프로세스 순서도 (Service Request)

## <공통> 결과 데이터

```
result = [
  'request' => ConnectRequest($attributes),
  'response' => ConnectResponse($message, $code, $data)
];

result = [
  request => [
    'item'    => ['sticky', 'chatbot', ...],
    'page'    => ['main_index', 'recruit_area_list', ...],
    /* 입력받지 못했을 경우 또는 잘못 입력한 경우 Service 단에서 자동 판단하여 입력 처리 */
    'situation' => ['apply', 'join', 'resume', ...],
    /* 입력받지 못했을 경우 또는 잘못 입력한 경우 Service 단에서 자동 판단하여 입력 처리 */
    'identify' => identify
  ],
  response => [
    'code'    => [400, 404, 200, ...],
    'status'  => ['OK', 'No Content', 'Bad Request', ...],
    'message' => ['성공', '콘텐츠 없음', '잘못된 요청', ...],
    /* IANA 가 지정한 HTTP STATUS CODE */
    'data'    => [
      'text'   => '당신이 누구인지 간단히 알려주세요.',
      'link'   => 'http://www.saramin.co.kr',
      'action' => 'popup_layer',
      'options' => ''
    ]
  ]
];
```



\$\_SERVER['SCRIPT\_URL'];

setpage

C ConnectSituationParameters

```
class ConnectSituationParameters
{
    public static $params = [
        'page' => [
            'main_index' => [
                '/'
            ],
            'recruit_area_list' => [
                '/location-job/recently-list',
                '/location-job/domestic-list',
            ],
            'recruit_job_list' => [
                '/job-industry/recently-list',
                '/job-industry/job-list',
                '/job-industry/job'
            ],
            'recruit_industry_list' => [
                '/job-industry/ids-list',
                '/job-industry/ids'
            ],
            'recruit_special_list' => [
                '/special-job/all',
            ],
            'recruit_publication_list' => [
                '/group-job/calendar',
                '/group-job/major',
                '/group-job/major-list',
            ],
            'theme' => [
                '/curation'
            ],
        ],
        'behavior' => [
        ]
    ];
}
```

다양한 사용자 데이터를 반환 & 검증해주는 함수를 작성해놓은  
<공통> ItemSituation abstract class 를 extends 한 뒤에 함수를 호출해서 사용

```
class IndexStory extends ItemSituation implements ConnectGetService
{
    public function __construct($attributes = [])
    {
        parent::__construct($attributes);
    }

    public function get()
    {
        if (!$this->isAuthenticated()) {
            /* @Situation 비회원, */
            return $this->getSituationResponse(self::FIRST_VISITED_NON_MEMBER_SITUATION);
        }

        if (!$this->isExistsResume()) {
            /* @Situation 회원, 이력서 미등록 */
            return $this->getSituationResponse(self::NOT_EXISTS_RESUME_MEMBER_SITUATION);
        }

        if ($this->isAppliedThreeDays()) {
            /* @Situation 회원, 이력서 존재, 3일내 입사지원 기록 보유 */
            return $this->getSituationResponse(self::IS_APPLIED_BY_THREE_DAYS_SITUATION);
        }

        if ($this->isAppliedThreeDays() && $this->getAvatarAttribute('직종')) {
            /* @Situation 회원, 이력서 존재, 3일내 입사지원 기록 미보유, 아바타 직종 속성 보유 */
            return $this->getSituationResponse(self::JIKJONG_SITUATION);
        }

        if ($this->isAppliedThreeDays() && $this->getAvatarAttribute('지역')) {
            /* @Situation 회원, 이력서 존재, 3일내 입사지원 기록 미보유, 아바타 지역 속성 보유 */
            return $this->getSituationResponse(self::AREA_SITUATION);
        }

        if ($this->isAppliedThreeDays()) {
            /* @Situation 회원, 이력서 존재, 3일내 입사지원 미보유, 아바타 직종, 지역 속성 미보유 */
            return $this->getSituationResponse(self::NO_AREA_AND_NO_JIKJONG_SITUATION);
        }

        return $this;
    }
}
```

C ConnectSituationService

setSituation

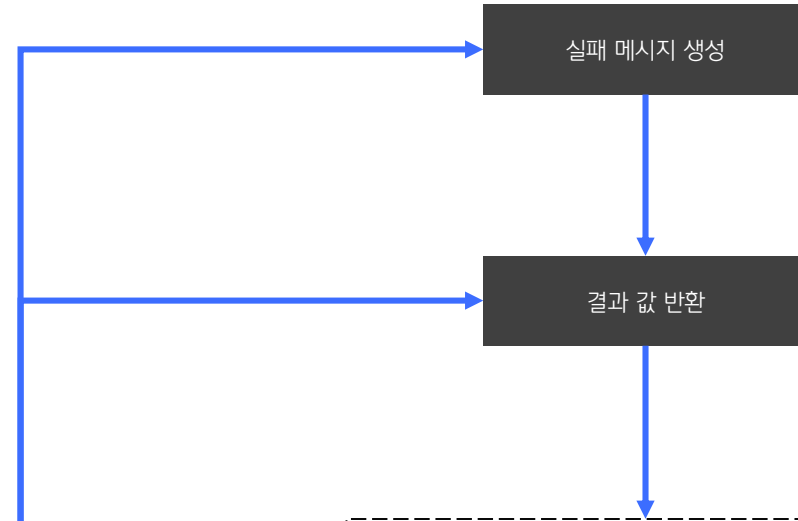
```
class ResponseBySituation
{
    public static $response = [
        'enter' => [
            'text' => [
                '반가워요. 사람입니다.',
                '사람인에 오신걸 환영해요.'
            ],
            'link' => '',
            'action' => 'nothing',
            'options' => ''
        ],
        'first' => [
            'text' => '당신이 누구인지 간단히 알려주세요.',
            'link' => 'http://33crdev.w-at.pdev2.saramin.co.kr/connect/welcome-star',
            'action' => 'popup_layer',
            'options' => ''
        ],
        'main' => [
            'text' => '오늘의 추천 공고를 확인하세요.',
            'link' => '/job-search/view?rec_id=27607773&t_ref=myhome&t_ref_content=recently_recruit',
            'action' => 'popup_layer',
            'options' => ''
        ],
        'register' => [
            'text' => '회원가입 완료! 지금 추천 공고를 확인하세요.',
            'link' => '/job-search/view?rec_id=27607773&t_ref=myhome&t_ref_content=recently_recruit',
            'action' => 'popup_layer',
            'options' => ''
        ],
        'application' => [
            'text' => '지원하신 공고와 비슷한 공고를 지금 확인하세요.',
            'link' => '/job-search/view?rec_id=27607773&t_ref=myhome&t_ref_content=recently_recruit',
            'action' => 'popup_layer',
            'options' => ''
        ],
        'resume' => [
            'text' => '이력서를 분석! 최적의 공고를 확인하세요.',
            'link' => '/job-search/view?rec_id=27607773&t_ref=myhome&t_ref_content=recently_recruit',
            'action' => 'popup_layer',
            'options' => ''
        ],
    ];
}
```

REDIS & DATABASE DATA



```
public static $codeDefinitions = [
    '1' => [
        '100' => [...],
        '101' => [...],
        '102' => [...],
    ],
    '2' => [
        '200' => [
            'status' => 'OK',
            'message' => '성공',
        ],
        '201' => [...],
        '202' => [...],
        '203' => [...],
        '204' => [
            'status' => 'No Content',
            'message' => '콘텐츠 없음',
        ],
        '205' => [...],
        '206' => [...],
        '207' => [...],
        '208' => [...],
        '209' => [...],
    ],
    '3' => [
        '300' => [...],
        '301' => [
            'status' => 'Moved Permanently',
            'message' => '영구 이동',
        ],
        '302' => [...],
        '303' => [...],
        '304' => [...],
        '305' => [...],
        '306' => [...],
        '307' => [...],
        '308' => [...],
    ],
];
```

HTTP STATUS CODE



<공통> 결과 데이터

```
result = [
    'request' => ConnectRequest($attributes),
    'response' => ConnectResponse($message, $code, $data)
];

result = [
    request => [
        'item' => ['sticky', 'chatbot', ...],
        'page' => ['main_index', 'recruit_area_list', ...],
        /* 입력받지 못했을 경우 또는 잘못 입력한 경우 Service 단에서 자동 판단하여 입력 처리 */
        'situation' => ['apply', 'join', 'resume', ...],
        /* 입력받지 못했을 경우 또는 잘못 입력한 경우 Service 단에서 자동 판단하여 입력 처리 */
        'identify' => identify
    ],
    response => [
        'code' => [400, 404, 200, ...],
        'status' => ['OK', 'No Content', 'Bad Request', ...],
        'message' => ['성공', '콘텐츠 없음', '잘못된 요청', ...],
        /* IANA 가 지정한 HTTP STATUS CODE */
        'data' => [
            'text' => '당신이 누구인지 간단히 알려주세요.',
            'link' => 'http://www.saramin.co.kr',
            'action' => 'popup_layer',
            'options' => ''
        ]
    ]
];
```

<http://sri-wiki.saraminhr.co.kr/display/~92.kr.dev/Sticky>



<https://jacobs902207.github.io/posts/webpack.html>

# Connect[Sticky]

with Webpack

