

# 여기요

# 목차



기획의도



구현 및 기술



보완사항 및 시연



**"나와 같은 진도를 나가고 있는 사람들은 오늘은 어디에 앉아 있을까?"**

# 같이 공부하고 싶은데 나와 함께할 카뎃이 어디 있죠?

동일한  
서브젝트를 함께  
진행하면 효율이  
매우 매우 매우  
매우 매우 높다!

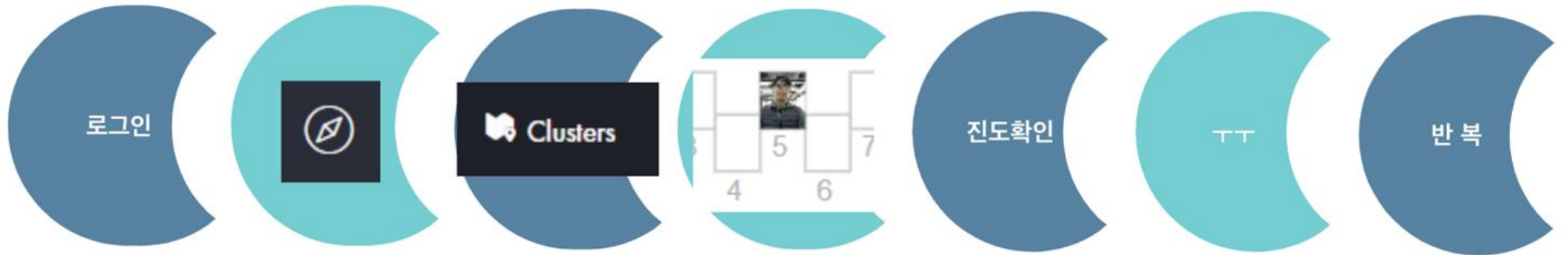
바로 옆 사람에게  
질문할 수 있고  
함께 고민할 수  
있어요

과제를 진행하다  
막혔을 때, 해당  
과제를 완료했거나  
진행중인 카뎃을  
찾는 방법은 현재  
조금 번거롭다

옆자리에 앉아 있는  
사람이 나와 같은  
진도를 나가면  
커뮤니케이션  
하기가 쉬워요 !!

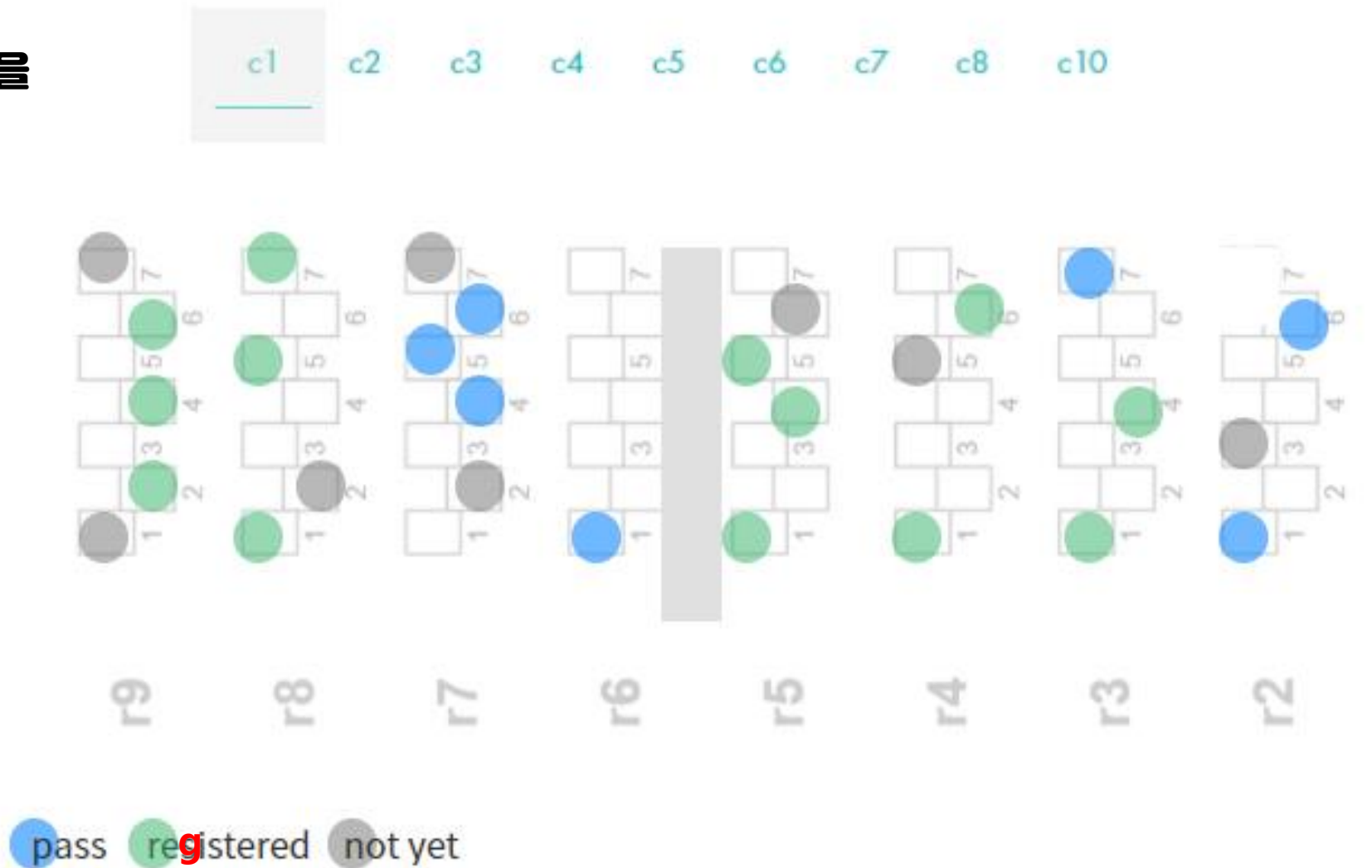
but

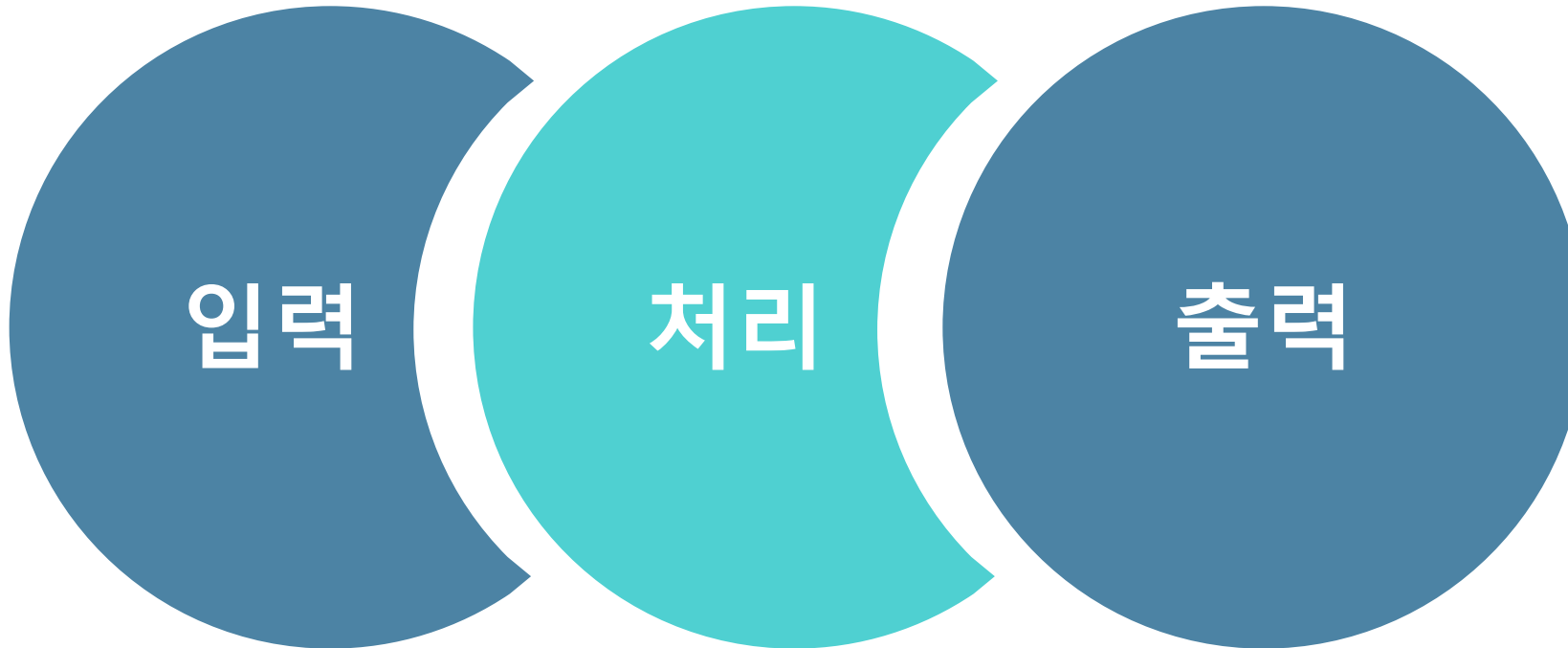
## 현재의 방법



# Concept

같은 진도를 나가는 사람을  
색상으로 구분해주자





## 임무 분담



iOS 환경에서 Subject를 입력 받고 로그인 화면 구성



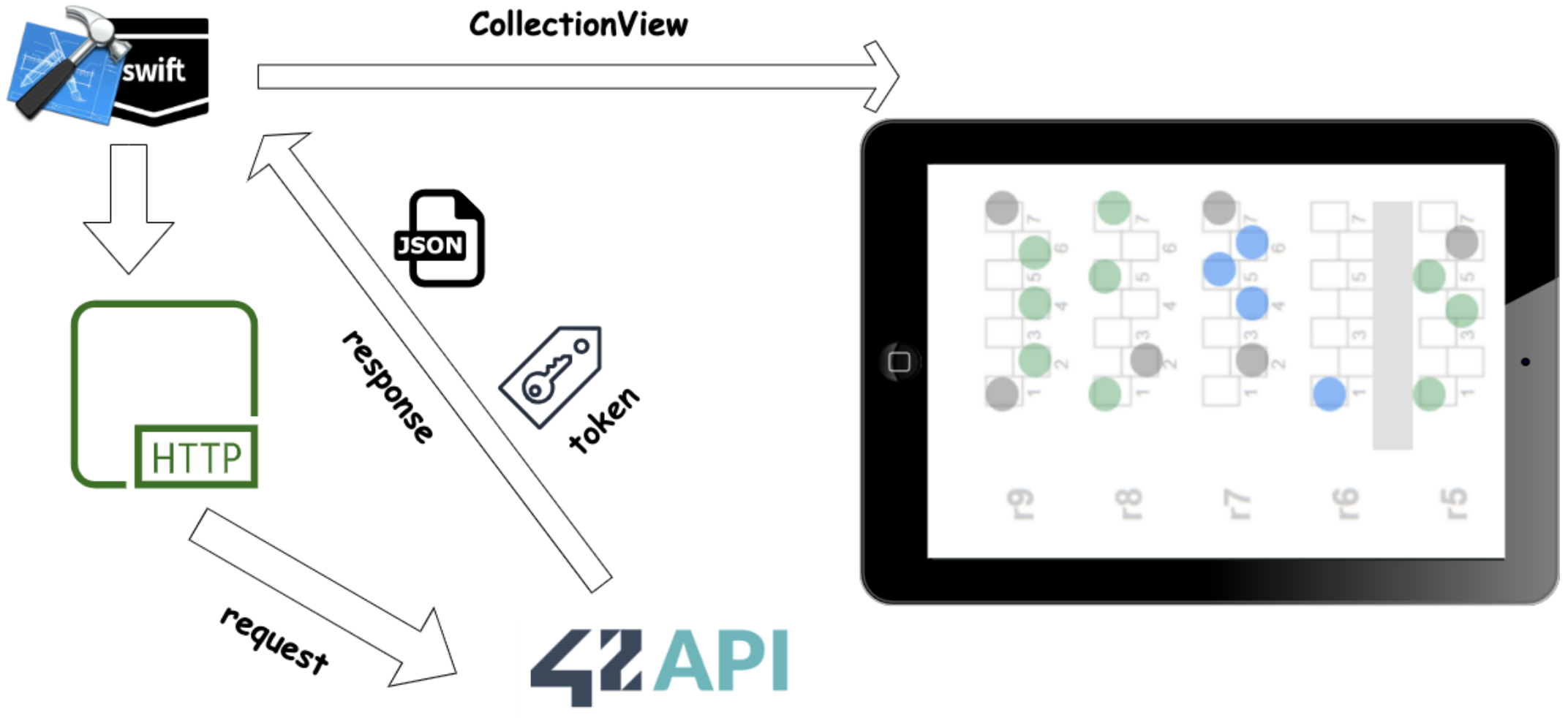
API 통신으로 현재 접속한 인원 확인 후, 과제에 대한 상태 처리



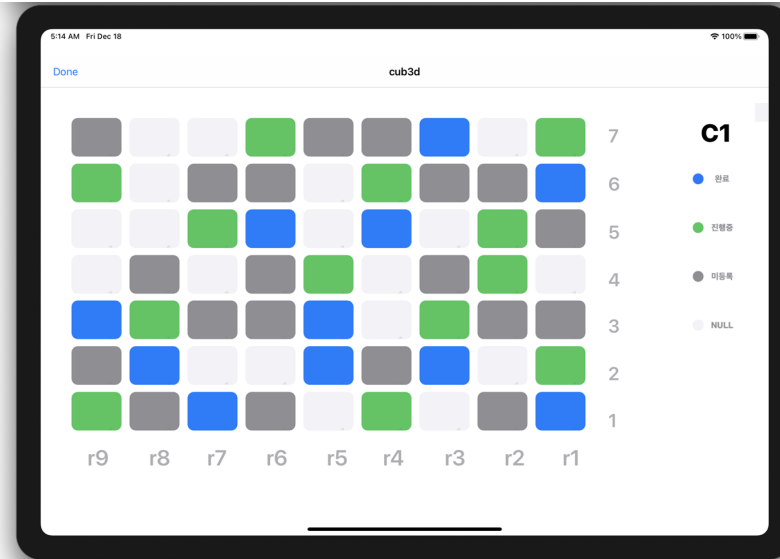
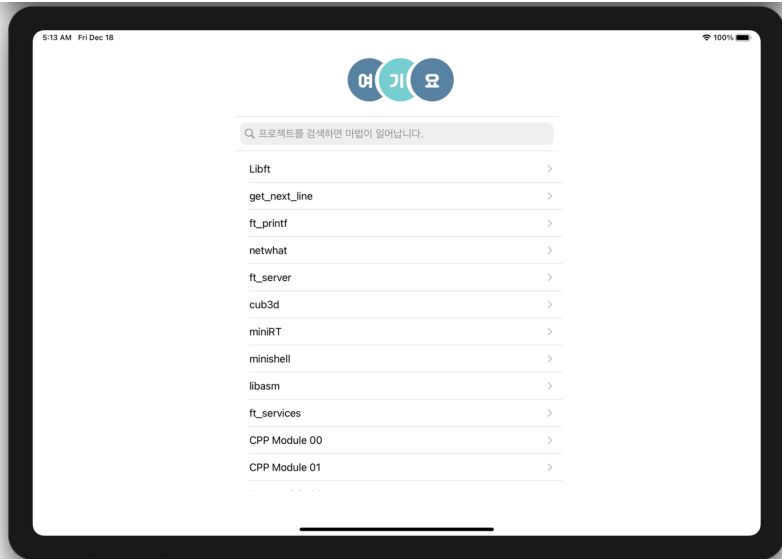
클러스터의 좌석을 보여주고 상태에 따른 색상을 입히기



# 구동 과정



iOS



**Simplicity is the best.**

**세상에서 가장 쉬운 검색,  
터치 한번이면 마법같은 일이 벌어집니다.**

## 자동완성 검색 기능

진행 중인 42 프로젝트를 검색하세요.

Q Ft

Libft >

ft\_printf >

ft\_server >

ft\_services >

ft\_containers >

ft\_irc >

ft\_transcendence >

```
func tableView(_ tableView: UITableView, cellForRowAt
indexPath: IndexPath) -> UITableViewCell {
    let cell =
        tableView.dequeueReusableCell(withIdentifier:
            "TableViewCell", for: indexPath) as UITableViewCell
    cell.textLabel?.text = filteredData[indexPath.row]
    return cell
}

func tableView(_ tableView: UITableView,
numberOfRowsInSection section: Int) -> Int {
    return filteredData.count
}

func searchBar(_ searchBar: UISearchBar, textDidChange
searchText: String) {
    filteredData = searchText.isEmpty ? data : data.filter
    { (item: String) -> Bool in
        // If dataItem matches the searchText, return true
        to include it
        return item.range(of: searchText, options:
            .caseInsensitive, range: nil, locale: nil) !=
            nil
    }

    tableView.reloadData()
}
```

## 기술 구현 - API

iOS

API

- **API 통신**을 하기위해 HTTP 연결 필요
- 따라서, Swift의 UIKit 에서 제공하는 **URLRequest** 사용
- 42 API가 제공하는 데이터 타입은 **JSON**
- JSON으로 제공되는 여러 데이터들 중 기능에 필요한 부분만 추출
- JsonSerialization을 통해 **JSON 데이터 Parsing**
- 기능에 필요한 정보만 가져옴. -> (user\_id, login, host, url, status)
- **User 객체**를 만들어서 저장
- 좌석별로 각각 객체를 만들어, 자리에 사람이 있는 경우 필요한 자료들 Setting
- 좌석 객체의 **Status**를 collectionView에 전달

## API 구현 코드

### HTTP 통신을 이용한 API 수집

```
func getCluster(url: URL, token: String) -> [[String: Any]]? {
    var request: URLRequest = URLRequest(url: url)
    request.httpMethod = "GET"
    let sem = DispatchSemaphore.init(value: 0)
    let appMessage = "Bearer " + token
    request.setValue(appMessage, forHTTPHeaderField: "Authorization")
    request.addValue("application/json; charset=utf-8", forHTTPHeaderField:
        "Content-Type")
    var json: [[String: Any]]?
    session.dataTask(with: request, completionHandler: { (data, response,
        error) in
        defer { sem.signal() }
        guard let data = data else { print("nothing");return }
        json = try? JSONSerialization.jsonObject(with: data, options: [])
            as? [[String: Any]]
    }).resume()
    sem.wait()
    return json
}
```

### JSON 데이터 Parsing

```
i = 1
while i < c.count
{
    j = 0
    while j < jsonCount - 1
    {
        let host : String? = jsonCluster?[j]["host"] as? String
        let end_at : String? = jsonCluster?[j]["end_at"] as? String
        if (c[i].compareSeats(host: host!) && end_at == nil)
        {
            let usrDic: NSDictionary? = (jsonCluster?[j]["user"])! as?
                NSDictionary
            let usrId : Int? = usrDic?["id"] as? Int
            let usrLogin : String? = usrDic?["login"] as? String
            let usrUrl : String? = "https://profile.intra.42.fr/users/" +
                usrLogin!
            c[i].setValidUser(id: usrId!, host: host!, url: usrUrl!, login:
                usrLogin!)
            c[i].showAll()
        }
        j += 1
    }
    i += 1
}
```

## 기술 구현 – Collection View

iOS

Collection View

API

- UICollectionView
- UICollectionViewCell
- UICollectionViewReusableView
- UICollectionViewFlowLayout
- UICollectionViewDataSource
- UICollectionViewDelegate
- UICollectionViewDelegateFlowLayout

**“컬렉션 뷰를 통해서 클러스터의 좌석을 각각의 셀로 구현”**

```
// 셀 크기 정하기 (n분의 1)
func collectionView(_ collectionView: UICollectionView, layout collectionViewLayout: UICollectionViewLayout, sizeForItemAt indexPath: IndexPath) CGSize {
    let height = collectionView.frame.height / 8 - 1 // n등분
    let size = CGSize(width: height, height: height)

    return size
}
// 셀 갯수
func collectionView(_ collectionView: UICollectionView, numberOfItemsInSection: Int) Int {
    return self.items.count
}
```

**셀 크기를 높이를 기준으로 동적으로 지정**

```
// 셀 속성
func collectionView(_ collectionView: UICollectionView, cellForItemAt indexPath: IndexPath) UICollectionViewCell {
    // get a reference to our storyboard cell
    let cell: MyCollectionViewCell = collectionView.dequeueReusableCell(at: indexPath, forCellWithIdentifier: "cell", reuseIdentifier: "cell")
    // 배경색
    if (items[indexPath.item] == "1") {
        cell.backgroundColor = UIColor.systemGray //make cell
    } else if (items[indexPath.item] == "2"){
        cell.backgroundColor = UIColor.systemGreen
    } else if (items[indexPath.item] == "3"){
        cell.backgroundColor = UIColor.systemBlue
    } else {
        cell.backgroundColor = UIColor.systemGray3
    }
}
```

**전달받은 좌석의 상태값에 따라 Cell 의 배경색상 변경**

```
return cell
}
// 셀 액션
func collectionView(_ collectionView: UICollectionView, didSelectItemAt indexPath: IndexPath) {
    //click 했을 때 특정행동 수행.
    if (items[indexPath.item] == "7") {
        if let url = URL(string: "http://www.naver.com") {
            UIApplication.shared.open(url, options: [:])
        }
    }
}
```

**셀 클릭시 전달받은 42intra user + id로 링크 이동**

```
// 셀 재사용 식별자
```

## 보완해야할 점



층을 탭으로 구분하여 표시



API를 받아오는 속도문제, 서버로 해결



과제이름 뿐만아니라 Intra ID로 서칭 기능



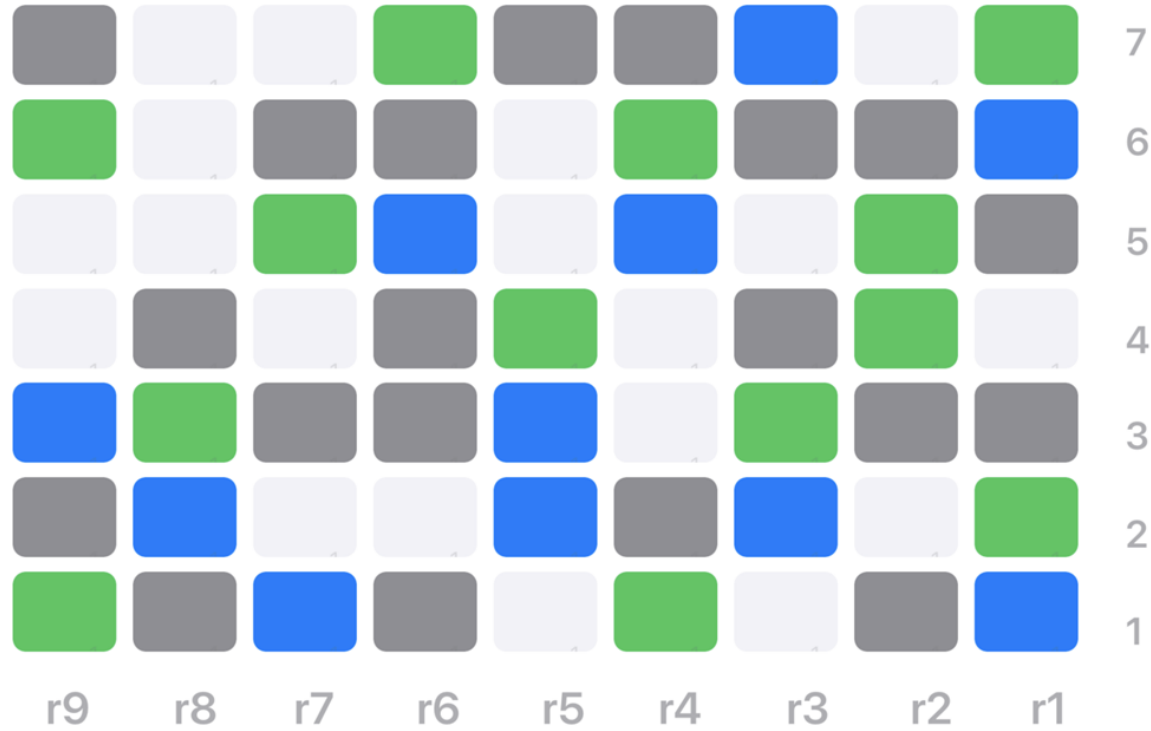


프로젝트를 검색하면 마법이 일어납니다.

- Libft >
- get\_next\_line >
- ft\_printf >
- netwhat >
- ft\_server >
- cub3d >
- miniRT >
- minishell >
- libasm >
- ft\_services >
- CPP Module 00 >
- CPP Module 01 >

Done

cub3d



C1

- Blue circle: 완료
- Green circle: 진행중
- Gray circle: 미등록
- Light Gray circle: NULL

# 시연



**감사합니다 !!!**

