캡스톤설계프로젝트

Kingo Manager

Team AIU

목차

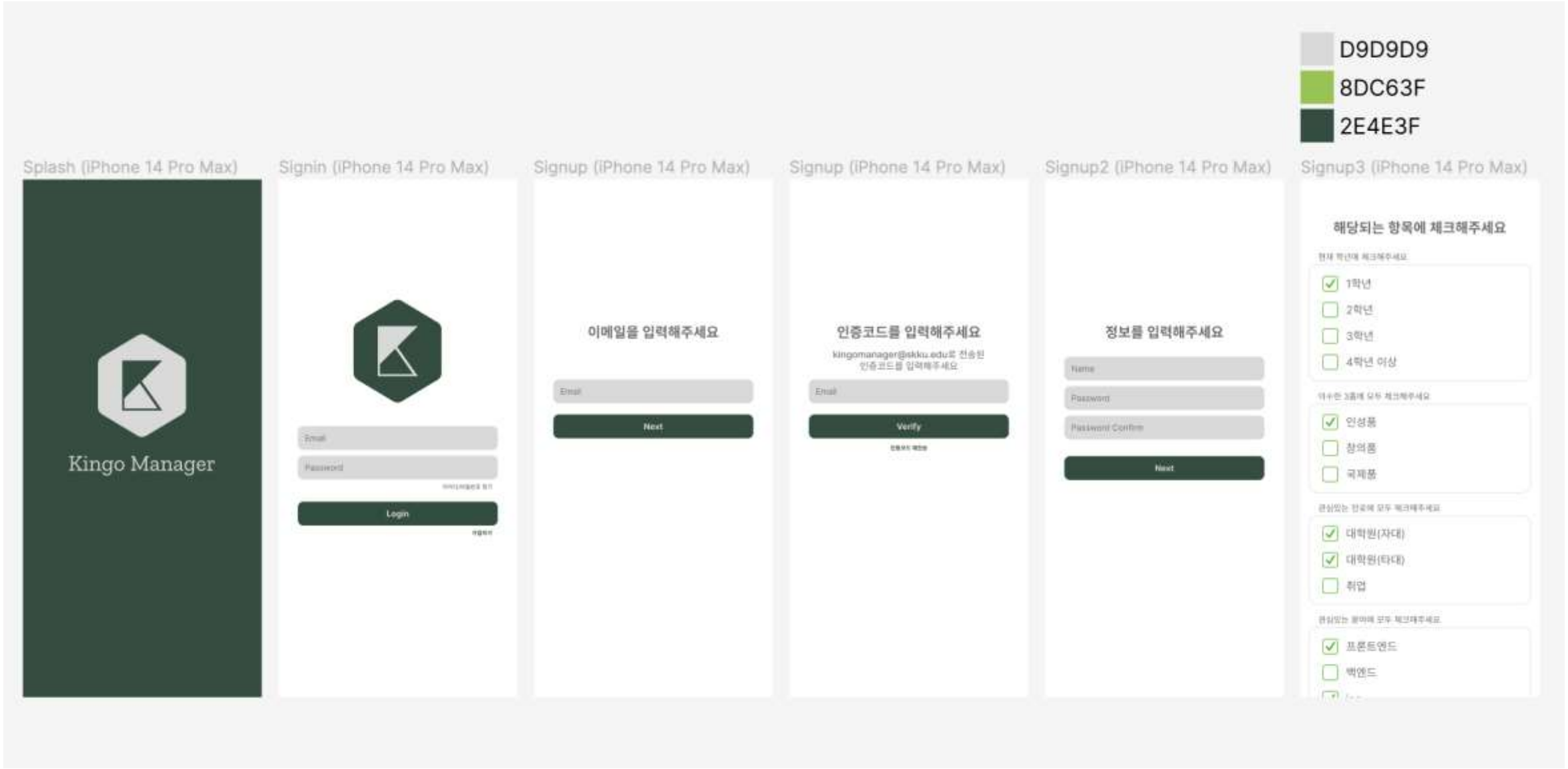
1. Frontend - UI & UX : 회원가입

1. Frontend - UI & UX : 회원가입

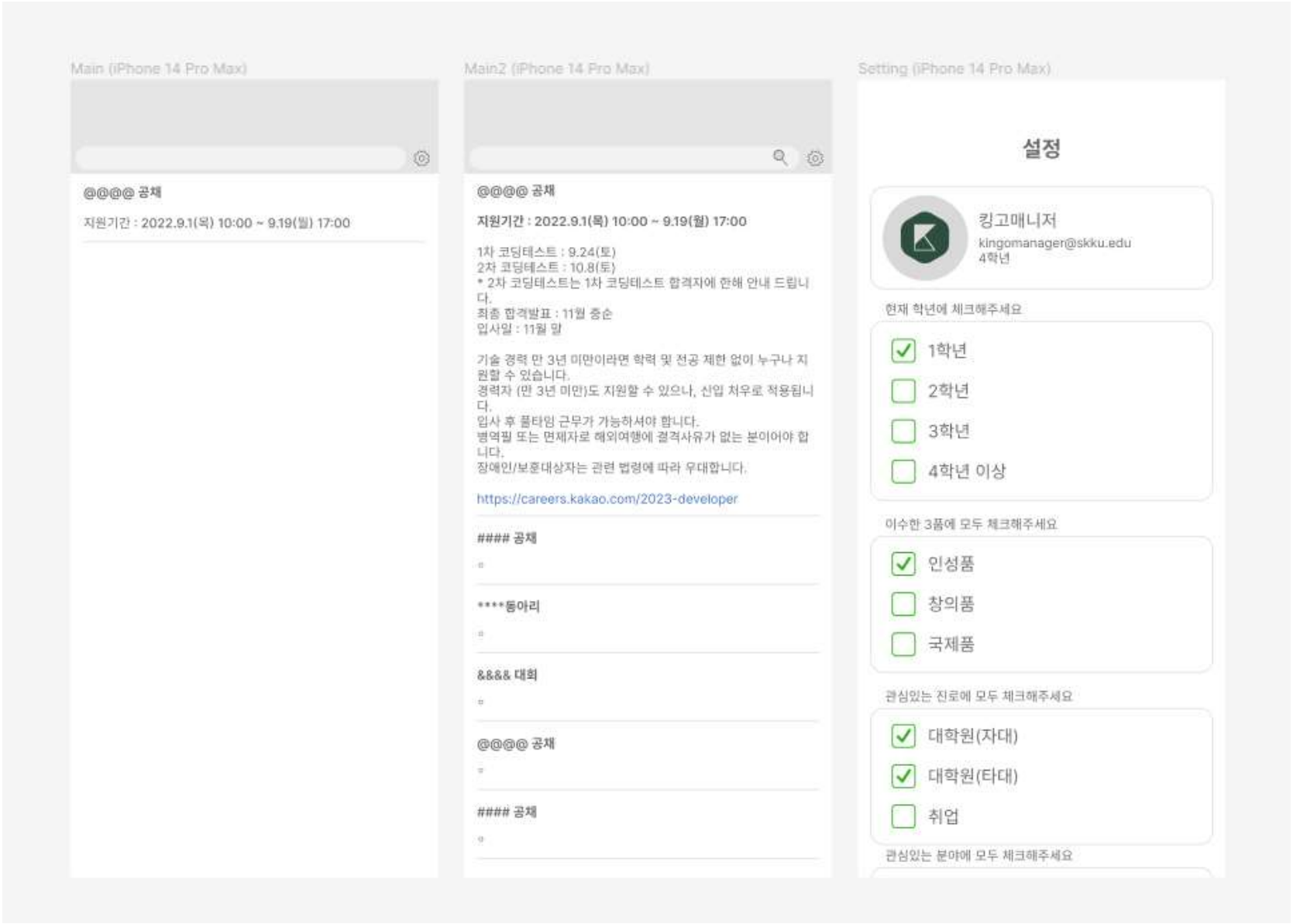
```
const _handleSignupButtonPress = async (event) => {
    setLoading(true);
    try {
        const response = await axios.post(`${baseURL}/adduser`, {
            name: name,
            email: email,
            password: password,
            grade: checkgrade,
            insung: insung,
            changeui: changeui,
            gukje: gukje,
            interest: interest,
        });
        Alert.alert('Success');
        navigation.navigate('Signin');
    } catch (error) {
        if (error.response) {
            const errorResponse = error.response;
            console.log('data : ', errorResponse.data);
            console.log('status : ', errorResponse.status);
            console.log('headers : ', errorResponse.headers);
        } else if (error.request) {
            console.log('request : ', error.request);
        } else {
            console.log('meassage : ', error.message);
        }
    } finally {
      setLoading(false);
    }
};
```

1. 로그인/로그아웃을 처음에 firebase로 구현하였다가 axios를 이용한 http통신으로 변경하였습니다.

1. Frontend - UI & UX : 회원가입



1. Fontend - UI & UX : 메인화면



현재 아직 구현하지 못한 부분

1. Async-storage를 이용한 게시물의 로컬 저장

2. 가입시 이메일을 통한 인증

Trial & error

1.지금까지 받은 추천 목록의 저장법에 대해 고민

(1) 로그인할 때마다 유저에 대한 데이터를 전부 불러온다
-> 추후 데이터의 양이 쌓이면 로그인시 시간이 상당히 소요된다. (x)

(2) 카카오톡처럼 거의 로그인할 일이 없기 때문에 토큰을 이용하여 로그인 상태를 유지하고 데이터는 로컬에 저장한다. (o)

2. 가입시 이메일을 통한 인증

Table Column redefine

```python
# define Table : class User, users
class User(Base):
    __tablename__ = "users"
    id = Column(Integer, primary_key=True)
    email = Column(String(50), unique=True)
    password = Column(String(50), unique=False)
    name = Column(String(50), unique=False)
    grade = Column(Integer, unique=False)
    insung = Column(Boolean, unique=False)
    changeui = Column(Boolean, unique=False)
    gukje = Column(Boolean, unique=False)
    interest = Column(Integer, unique=False)
```

```python
# define Table : class Recommend, recommends
class Recommend(Base):
    __tablename__ = "recommends"
    id = Column(Integer, primary_key=True)
    email = Column(String(50), unique=False)
    title = Column(String(100), unique=False)
    category = Column(String(100), unique=False)
    contents = Column(String(500), unique=False)
    link = Column(String(100), unique=False)
```

Encoding issue

Utf-8

성균관대학교
공지사항

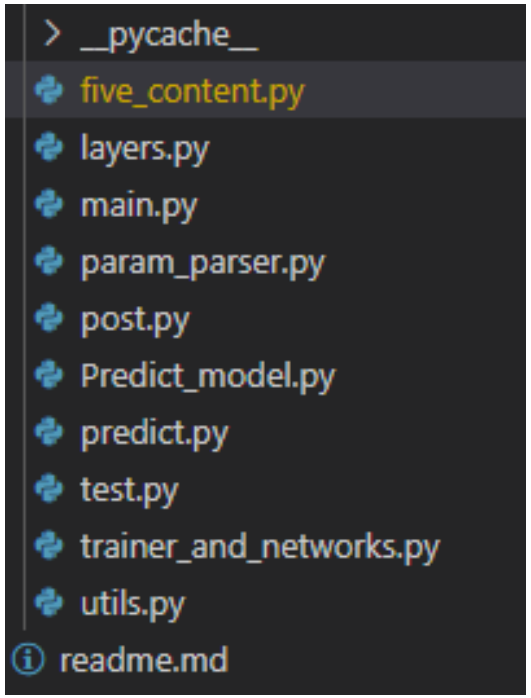
%EC%84%B1%EA%B7
%A0%EA%B4%80%EB
%8C%80%ED%95%99
%EA%B5%90%20%EA
%B3%B5%EC%A7%80
%EC%82%AC%ED%95
%AD

성균관대학교
공지사항

encoding

decoding

Model to Post




```python
def Post(input):
    for i in input:
        headers = {'Content-Type': 'application/json',}

        #joson_data = input
        json_data = i

        url = 'https://3llwwh4zd4.execute-api.ap-northeast

        response = requests.post(url=url, headers=headers,

        #print(response.text)
        #print(response.status_code)
```

```python
def Choose_five():
    if(main() == 1):
```

```python
return json_list
```

Thank you!