

System Structure & Architecture

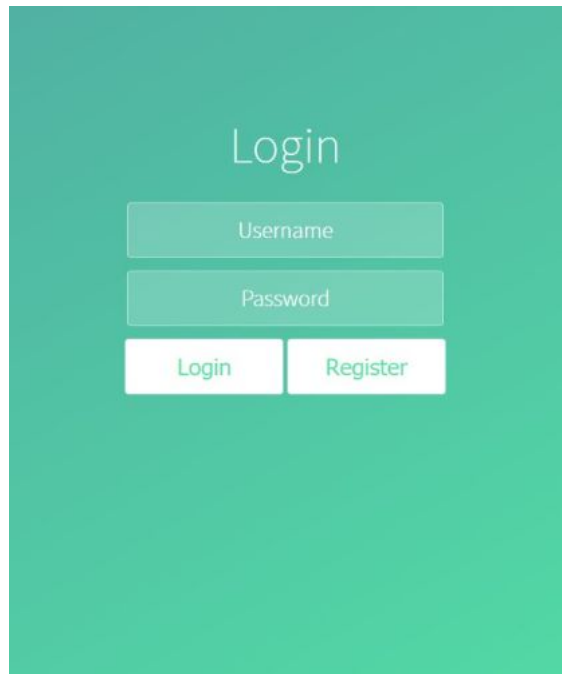
- FrontEnd Structure
- Architecture
- Database Structure

Terminology

- Sign up, in page
- Main page
- Group List Page: 그룹들 목록 페이지
- Group's page: 해당 그룹 개별 페이지. 책들의 목록과 이 달의 책을 보여준다.
- 독후감 목록 페이지: 개별 그룹 페이지의 개별 책에 대한 독후감들의 목록을 보여주는 페이지.
- 독후감 내용 페이지:

User Stories

- 1 (1) As a user, I can sign up, in, and out
 - Actor: User
 - Input: Email & PW input
 - Output:
 - Up : Login Page
 - In : Blog main page or pw again
 - Description: 유저가 이메일이랑 패스워드를 쳐서 로그인 버튼을 누르거나/싸인업 버튼을 눌러서 signup페이지로 들어가 signup작성



-
- Scenario:
 - Sign in
 - 사용자가 sign in 버튼을 누름
 - 사용자가 이메일 & 패스워드를 입력
 - 사용자가 confirm버튼을 누름
 - 맞는 이메일/pw인 경우 블로그 메인 페이지로 이동해야 함.
 - Sign up
 - 사용자가 sign in 버튼을 누름
 - 사용자가 sign up 버튼을 누름
 - 사용자가 이메일 & pw를 입력
 - (확인 이메일 -> backlog)
 - 사용자가 sign in 페이지로 이동해야 함.
 - Sign Out
 - 사용자가 로그인한 상태에서 sign out버튼을 누름
 - 사용자가 sign in 페이지로 이동.
- Task
 - Task 1: UserCreationForm을 이용하여 sign up 을 구현한다
 - Task 2: User model을 이용하여 in, out 을 구현한다
- 2 (1) As a user, I can see "this month's book"s of every group on the main page
 - Actor: User
 - Input: 그룹 클릭 / 로그인
 - Output:
 - 메인 페이지 상단에 이번달 책이 다 뜸
 - Description: 사용자가 처음에 자신이 속해 있는 모든 그룹의 이번달 책을 볼 수 있고, 그룹 내에 들어가면 각 그룹의 이 달의 책을 볼 수 있음.
 - Scenario:
 - 로그인

- 사용자가 로그인한다. (이 때 테스트 유저는 테스트를 위해 하나 이상의 그룹에 속해 있어야 한다)
 - 유저는 자신이 속한 모든 그룹의 이 달의 책을 메인 페이지에서 확인할 수 있어야 한다.
 - Task
 - Task 1: Main view 에서 유저가 속한 그룹 목록을 띄운다. 생성된지 얼마 안된 그룹이 위로
 - Task 2: Main view 에서 그룹 목록 중 하나를 클릭하면 관련 그룹 뷰로 들어간다
 - Task 3: Main view 에서 그룹 목록 위에 또 다른 목록인 이달의 책 목록이 떠야한다. 순서는 최신 순
 - Note
 - 이거 구현순위를 위로 좀 올려야할 거 같다 적어도 에디팅 위로
- 3 (1) As a user, I can upload a book report through the editor and I can see other members' book reports
 - Actor: User
 - Input: 책 클릭 + 제목 + Text + editor input/ 그룹 클릭 + 책 클릭 + 리포트 클릭
 - Output: 해당 글 페이지/ 리포트 창
 - Description: 유저가 에디터를 사용해서(혹은 그냥 텍스트를 사용하여) 해당 책에 대한 리포트를 업로드한다.
 - Scenario:
 - 유저가 글 쓰고 싶은 그룹으로 들어간다.
 - 유저는 그룹 페이지에서 책 목록을 볼 수 있어야 한다. 위로 갈 수록 최신. 이 달의 책은 다르게 구분하여 볼 수 있다.
 - 유저가 책 목록 내의 책을 클릭하면, 책에 대한 독후감 목록을 확인할 수 있다.
 - 유저가 독후감 목록 내의 새 글 쓰기 버튼을 누른다.
 - 유저가 (텍스트 에디터 -> backlog)텍스트 박스에 글을 쓰고 submit을 누른다.
 - 취소를 누르는 경우 글 쓰고 싶은 그룹으로 돌아가야 한다.
 - 그러면, 유저가 자신의 글을 볼 수 있어야 한다.
 - Task
 - Task 1: 유저가 그룹을 클릭하면 Main view 에서 특정 Group view 로 들어가도록 구현
 - Group view 에는 책 목록이 나온다
 - Task 2: 독후감을 클릭하면 독후감을 띄우는 페이지로 가도록 만든다
 - Task 3: 새 글 쓰기 버튼을 만들고 새 글 view 를 구현한다. 새 글을 작성하면 그 독후감을 바로 볼 수 있도록 만든다.
 - Task 4: 새 글은 독후감 목록의 최상단으로 올라간다
 - Task 5: Main view 에서 이달의 책 목록을 클릭하면 바로 독후감 목록을 볼 수 있다(그룹 클릭 없이)
 - Note:
 - 텍스트 에디터, 이거 분명 이미 인터넷에 돌아다니는거 있다
 - 테스트 할 때는 책 목록을 임의로 만들어 두자 (차후에 그룹 리더가 책 목록을 업로드 할 수 있다)

- 4 (1) As a user, I can make a group so that I am the leader of the group
 - Actor: User
 - Input: 그룹 만들기 버튼 + 그룹 이름, 그룹 소개
 - Output: 그룹 페이지
 - Description: 유저가 그룹을 만들 수 있다. 만든 그룹에서 해당 유저는 leader의 직위를 가진다.
 - Scenario:
 - 유저는 메인 페이지에서 create new group 버튼을 누른다.
 - 유저는 새 그룹을 만드는 창으로 이동해야 한다.
 - 유저가 그룹 이름과 그룹 소개글을 form안에 채워 넣고 submit버튼을 누른다.
 - cancel을 누를 시, 메인 페이지로 다시 라우팅한다.
 - 유저는 만든 그룹 페이지로 이동해야 한다.
 - Task
 - Task 1: Main view 에서 create new group 버튼을 만든다. 버튼을 클릭하면 새 그룹을 만드는 view 로 이동한다
 - Task 2: 새 그룹을 만드는 view 에서 채워 넣을 수 있는 form 들을 구현한다
 - Task 3: cancel 버튼은 메인 페이지로, submit 버튼을 누르면 해당 그룹 페이지로 이동한다. 그룹 페이지 초기에는 아무 책도 지정되어 있지 않아야한다.
 - Note:
 - (Backlog): 비밀 그룹을 만들 수 있다.
 - 이 경우 그룹을 만들 때 비밀 코드를 받아서 유저가 join할 경우 그 코드를 통해 그룹에 조인할 수 있도록 한다.

- 5 (1) As a user, I can join a group
 - Actor: User
 - Input: Join Group 버튼 + 해당 그룹 클릭
 - Output: 조인되었습니다 메시지(혹은 팝업)와 메인 페이지
 - Description: 유저가 그룹에 조인할 수 있다.
 - Scenario:
 - 유저가 메인 페이지에서 Join Group 버튼을 누른다.
 - 유저는 그룹 목록 창으로 이동해야 한다. 여기서 그룹의 간단한 소개글과 그룹 이름을 확인할 수 있다.
 - (backlog -> 그룹 목록 창에서 검색이 가능하다)
 - (backlog -> pagination)
 - (backlog -> searching)
 - 유저는 각 그룹 오른쪽의 join버튼을 눌러서 조인할 수 있다.
 - 유저는 메인 페이지로 돌아가면서 조인되었습니다 메시지(혹은 팝업)을 받는다.
 - 메인 페이지에서 새로 가입한 그룹도 볼 수 있어야 한다
 - Task
 - Task 1: 메인 뷰에 Join Group 버튼을 만든다. 버튼을 클릭하면 그룹 목록 뷰로 이동한다.

- Task 2: 그룹 목록 뷰를 구현한다. 그룹을 쭉 나열하고(최신순) 그룹명 오른쪽의 join 버튼을 구현한다. 버튼을 클릭하면 유저 정보를 바꿔서 그룹에 join 한 것으로 한다. 적절한 메시지를 구현한다.
- 6 (1) As a leader of a group, I can upload "this month's book"
 - Actor: Leader
 - Input: 그룹 클릭 + 책 이름 + 책 내용
 - Output: refresh된 그룹 페이지
 - Description: 리더는 자신의 그룹의 이 달의 책에 대한 정보를 업로드할 수 있다. 그룹 멤버는 불가능.
 - Scenario:
 - 유저는 메인 페이지에서 자신의 그룹을 누른다.
 - 그룹 페이지에서, 리더인 경우 책 추가 버튼을 볼 수 있다.
 - 리더가 아닌 경우 책 추가 버튼을 볼 수 없다.
 - 책 추가 버튼을 누르면, 책 추가 페이지로 이동하여 책 이름과 description을 적고, confirm 버튼을 누른다.
 - 뒤로 가기 버튼을 누르는 경우 그룹 페이지로 다시 라우팅한다.
 - 그룹 페이지로 이동해야 하고, 이 때 추가된 책을 볼 수 있어야 한다.
 - 비동기 처리를 할 것인가?
 - Task
 - Task 1: 리더와 일반 유저를 구분 해야한다. 각 그룹을 관리하는 db를 만들고 그룹 이름과 리더의 id(?)를 저장해둔다. 해당 그룹의 리더만 책 목록 페이지에서 이 달의 책을 업로드 할 수 있는 버튼이 뜬다
 - Task 2: 업로드 뷰를 만들고 이 달의 책에 관해 채워 넣을 수 있는 form 들을 구현한다
- 7 (2) As a user, I can edit a book report after uploading
 - Actor: User(리포트를 하나라도 올린)
 - Input: 자신이 올린 북 리포트 + edit 버튼 클릭 + confirm
 - Output: edit된 북 리포트 페이지.
 - Description: 유저는 자신이 올린 리포트를 edit할 수 있다. 비동기 처리할 것인지는 아직 모름.
 - Scenario:
 - 유저는 메인 페이지에서 자신의 그룹을 누르고, 책 목록에서 책을 누르고, 자신이 쓴 리포트를 누른다.
 - edit 버튼을 누른다.
 - 자신이 쓴 리포트가 아닌 경우, edit버튼이 없어야 한다.
 - 유저는 텍스트 박스에서 자신이 썼던 글을 확인해야 한다. 그 상태에서 edit가능. confirm 버튼을 누른다.
 - 뒤로가기를 누르면 리포트 페이지로 라우팅한다.
 - 변형된 리포트 페이지로 이동해야 한다.
 - 비동기 처리를 할 것인가?
 - Task
 - Task 1: 독후감 내용 페이지에 edit 버튼을 만든다.
 - Task 2:
 - Note
 - edit 버튼을 띄울 때 자신이 쓴 건지 아닌지는 db 구조를 결정한 뒤에

- 8 (3) As a leader of a group, I can see the progress of members
 - Actor: Leader
 - Input: Progress 버튼
 - Output: 각 멤버들의 독후감 차트 -> 형식은 어떻게?
 - Description: 리더는 자신의 그룹 내에서 각 멤버들의 progress를 확인할 수 있다. 각 책에 대해서 얼마나 독후감을 진행했는지, 독후감의 수로 확인 가능하다. 형식은 여기다 추후 기술 바람.
 - Scenario:
 - 리더는 메인 페이지에서 자신의 그룹을 누르고, 그룹 페이지에서 progress버튼을 볼 수 있다.
 - 리더가 아닌 경우 이 버튼을 볼 수 없어야 한다.
 - 리더는 Description에서 서술한 progress형식에 따른 멤버별 progress표를 확인해야 한다.
- 9 (3) As a leader of a group, I can delete the user from the group
 - Actor: Leader
 - Input: Members + delete
 - Output: Members 페이지
 - Description: 리더는 멤버스 페이지에서 delete버튼을 볼 수 있고, 이것을 눌러서 멤버를 삭제할 수 있다. 이 경우 실수 방지를 위해 정말 삭제하시겠습니까 띄워주기
 - Scenario:
 - 유저는 메인 페이지에서 자신의 그룹을 누르고, 멤버스 버튼을 눌러서 Members페이지로 들어갈 수 있다.
 - 멤버 정보
 - 자기에 대한 설명
 - 아이디
 - (프사)
 - 리더의 경우 각 멤버 정보 옆에 delete버튼을 볼 수 있다.
 - 리더가 아닌 경우 버튼을 보면 안 된다.
 - 리더는 delete버튼을 누르면 정말 삭제하시겠습니까? 라는 창을 확인해야 한다. 그 상태에서 취소를 누르면 팝업창만 없어짐. 확인을 누르면 members페이지를 리다이렉트 시킨다.
 - 리더는 바뀐 멤버 페이지를 확인해야 한다.
 - Task
 -
- 10 (3) As a leader of a group, I can approve a user who joined a group
 - Actor: Leader
 - Input: Members + approve
 - Output: Members 페이지
 - Description: 리더는 멤버스 페이지에서 승인 대기 중인 멤버 목록을 하단에 확인할 수 있다. 이 경우 멤버 정보는 10과 동일. 이 상태에서 승인을 누르면 리다이렉트 된 멤버스 페이지를 확인한다.
 - Scenario:

- 리더는 메인 페이지에서 자신의 그룹을 누르고, 멤버스 페이지를 들어간다.
- 리더인 경우 승인 대기 중인 멤버 목록을 볼 수 있다. 리더가 아닌 경우 보면 안 된다.
- 리더가 승인 대기 중인 멤버 오른쪽의 승인 버튼을 누르면, members페이지로 리다이렉트 된다. 이 경우 승인 대기 목록에서 해당 멤버는 지워지고, 그룹 멤버 목록에 해당 멤버가 업데이트된다.
- (backlog -> 그룹 멤버의 이메일로 그룹 승인 이메일을 보낸다.)

- Task

-

- 11 (3) As a user, I want to see well-designed application

- Actor: User

- Input: -

- Output: 예쁜 UI

- Scenario:

- 이쁨

- Task

-