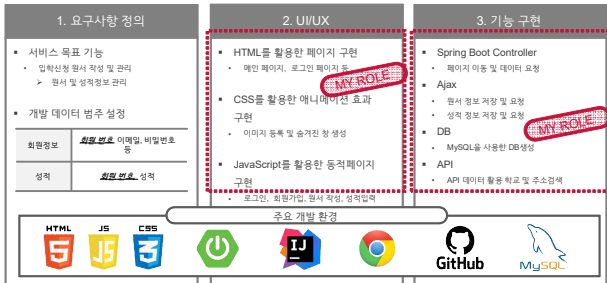


고등학교 신입생 입학신청 사이트 및 관리자 사이트 통해
웹 디자인 / 기능 구현 역량을 갖추고자 노력하였습니다.



본 프로젝트를 통해 기본적인 프론트,백 / 웹 구현에 필요한 API 자료 탐색 및 적용의 과정을 경험할 수 있었습니다.

개발 역량은 다양한 소스에 대한 탐색과 적용 노력을 통해 얻을 수 있다고 생각합니다.

이러한 노력을 웹서비스의 기존 코드 리뷰 및 외부 소스에 대한 적용으로 활용하고자 합니다.

Ajax 를 통한 로그인, 비밀번호 찾기 서비스 개발을 통해 프론트,백 엔드 개발 역량을 갖추고자 노력하였습니다.

로그인 Ajax 코드

```
$.ajax({
  url    : "/user/login",
  data   : sendData,
  type   : "GET",
  success : function(result) {
    $('#load').hide();
    if(result.loginResult == "1"){
      location.href = "/pd/modify";
      location.href = "/dss/main";
    }else if(result.loginResult == "0"){
      swal({
        text: "계정 정보가 일치하지 않습니다.",
        icon: "error"
      });
    }
  },
  error: function(request, status, error){
    $('#load').hide();
    swal({
      text: "로그인 도중 서버에 문제가 발생했습니다.",
      icon: "warning" // "info, success, warning, error" @ 역1
    });
  }
});
```

- Ajax 비동기 활용 데이터 전송 및 결과값 리턴

로그인 페이지

본인확인

핸드폰번호

-없이 입력해주세요.

핸드폰인증시 사용했던 휴대폰 번호를 입력해주세요.

비밀번호

확인버튼 클릭시 저장한 연락처로 이메일로 발송합니다.

처음

확인

비밀번호 찾기

- 로그인, 비밀번호 찾기 기능
- 로그인시 작성한 원서화면으로 이동, 비밀번호 찾기 클릭시 숨겨진 모달창이 보여지며 핸드폰 SMS 인증 및 비밀번호 재설정

Ajax, API 를 통한 원서 작성 및 주소, 학교검색 서비스 개발을 통해 프론트,백 엔드 개발 역량을 갖추하고자 노력하였습니다.

입학원서 작성 및 수정 페이지

[illegible]

- 입학원서 사진등록, 정보 저장 및 수정
- API 활용 주소 검색 및 학교 검색

Ajax 를 통한 학생들이 작성한 원서 리스트 출력 서비스 개발을 통해 프론트,백 엔드 개발 역량을 갖추하고자 노력하였습니다.

```

@GetMapping("/{personId}")
public String PersonData(Model model, Pageable pageable, HttpServletResponse session,
    @RequestParam(required = false, defaultValue = "0", value = "page") int page,
    @RequestParam(required = false, defaultValue = "", value = "select") String select,
    @RequestParam(required = false, defaultValue = "", value = "title") String title) {
    if (!session.checkLoginSessionCheckSession()) {
        pageable = PageRequest.of(page, 10, Sort.by("studentId").descending());
        int answer;
        Page<PersonDataEntity> memberEntities = personDataService.selectAllTableByName(pageable);
        Pagination pagination = new Pagination(memberEntities.getTotalPages(), page);
        model.addAttribute("pageId", "pageId", attributeValue("log"));
        model.addAttribute("pageId", "pageId", attributeValue("/{personId}"));
        model.addAttribute("personId", memberEntities);
        model.addAttribute("thisPage", pagination.getPage()); // 현재 몇 페이지인지 있는지 확인할 때
        model.addAttribute("isSelect", pagination.isSelect()); //각 페이지에 무슨 권한이 있는지
        model.addAttribute("isPreSelect", pagination.isPreSelect()); //이전페이지로 이동
        model.addAttribute("firstPage", pagination.getFirstPage()); //첫 페이지로 이동
        model.addAttribute("lastPage", pagination.getLastPage()); //마지막 페이지로 이동
        model.addAttribute("totalPage", pagination.getTotalPages()); //총 페이지
        return "{personId}";
    } else {
        return "redirect:/";
    }
}

```

- [illegible]

- 리스트를 가져와 뷰에 출력 후 페이징 및 검색 기능
- 해당 파일 다운로드 및 수정 기능