

병원예약관리 시스템

주제: 파일 입출력 기반의 데이터 처리 자바 콘솔 프로젝트 (병원 예약 시스템 구현)

1. 개발환경

JAVA version (Language)	JDK 11.0.2
-------------------------	------------

2. 개요 및 기획 의도(목적)

개요	어플의 단점을 보완하고 등록기관(병원)보다 사용자의 편의에 맞춘 프로그램 구현
문제점 및 기획 의도	1. 문제점: a. 대리 예약 시스템의 미비 b. 병원 리뷰에 관한 정보 및 환경의 불충분성
	1. 기획의도 a. 가족정보를 등록하고 추가한 가족을 대리 예약할 수 있는 시스템을 구현 b. 병원 리뷰 시스템을 구현하여 방문 전 병원에 대한 정보를 얻는 환경 구축 2. 추가 사항 a. 예약 뿐만 아니라 시스템을 세분화하여 효율성 극대화

3. 핵심 업무

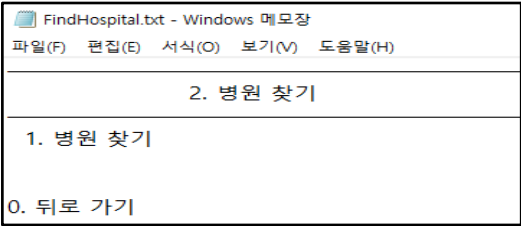
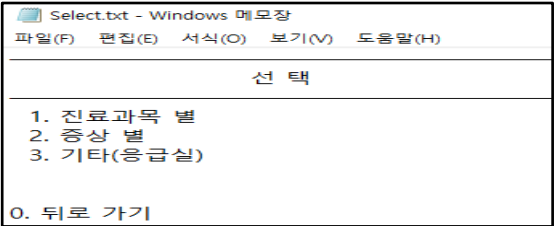
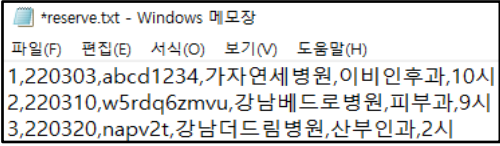
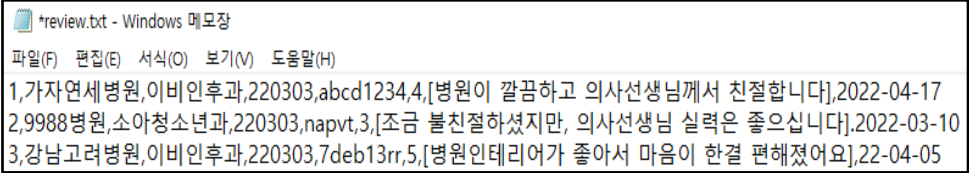
목차	
업무	상세내용
[회원등록]	
1. 회원가입	시스템 이용을 위한 회원으로 등록하는 기능
2. ID·PW 찾기	등록된 아이디와 비밀번호를 찾는 기능
3. 로그인	등록된 아이디와 비밀번호로 시스템에 접속하는 기능
[마이페이지]	
1. 개인정보 수정	회원데이터를 읽어 로그인한 사용자의 정보를 새로운 정보로 덮어쓰는 기능
2. 가족 추가	로그인 된 회원의 아이디로 가족코드를 부여하여 가족을 추가하는 기능
3. 진료내역 확인	현재 예약한 내역과 과거 진료받았던 기록을 확인하는 기능
4. 리뷰 작성	병원의 이름을 입력한 후 평점과 리뷰를 작성하는 기능
[병원 찾기]	
1. 원하는 병원 찾기	위치적 상황적 부분을 모두 고려하여 세부적으로 카테고리를 나열하는 기능
2. 의료기관정보출력	의료기관의 기본정보를 출력하고 상세정보를 조회/출력하는 기능
3. 병원 예약	예약 가능한 날짜를 확인하여 병원을 예약하는 기능

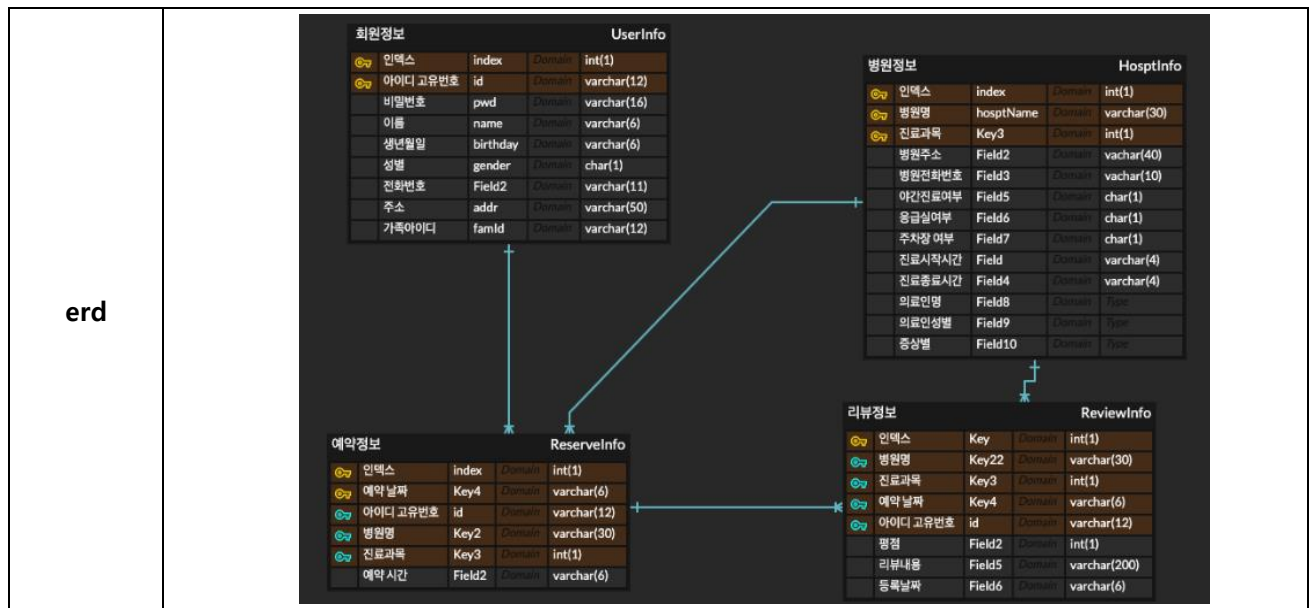
[관리자]	
1. 병원 데이터 등록	병원 데이터를 추가하기 위한 기능
2. 병원 데이터 수정	병원 데이터를 수정하기 위한 기능
3. 회원 정보 관리	회원들의 데이터를 관리하기 위한 기능

4. 담당 업무

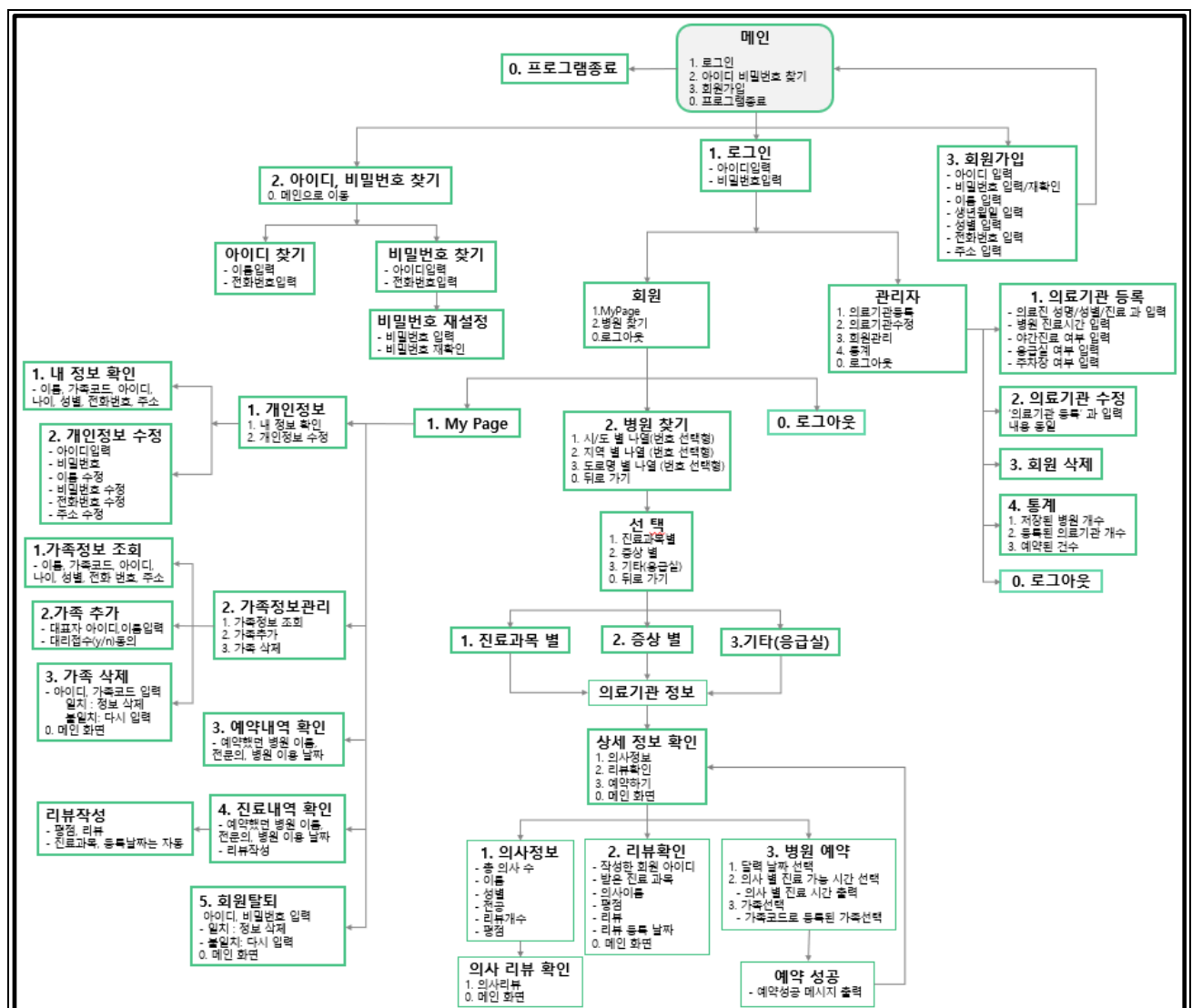
김은지 [병원목차]	조장 및 총괄 문서: 화면 설계서 취합, 발표 대본 취합, 자료조사, 전체문서 검수(수정) 구현: 지역 세부상 및 사항 목차 구현
권민채 [마이페이지]	문서: 구현: 개인정보 수정, 가족 추가, 진료내역 확인 및 리뷰작성, 회원탈퇴
김성태 [병원 찾기]	문서: 기능명세서 취합, 의료진 진료시간 데이터 생성 구현: 병원 예약 시스템 구현
백서영 [회원등록]	문서: 순서도 합본, ppt 제작 및 취합, 회원 데이터 생성 구현: 로그인, 회원가입, ID·PW 찾기
염규준 [관리자]	문서: 데이터 관계도 설계(erd), 개발문서 취합, 코드 취합 구현: 관리자 병원등록, 병원수정, 회원정보, 통계관리 기능 구현 및 데이터 설계
차수연 [병원찾기]	문서: 발표, 코드 취합 구현: 의료기관 기본 정보 확인, 의료기관 의사 정보 확인, 의료기관 리뷰 정보 확인

5. 데이터 구조

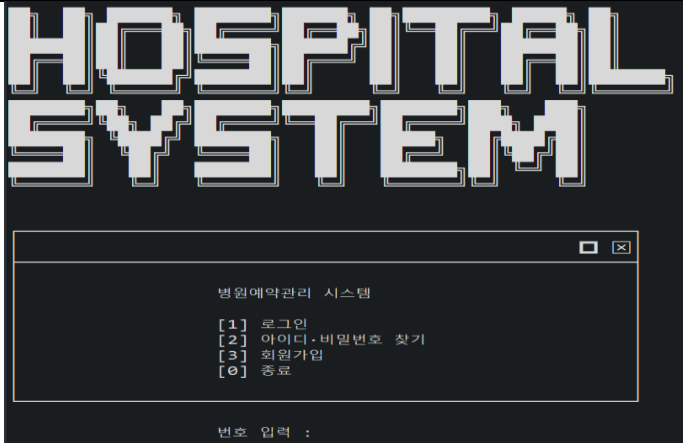
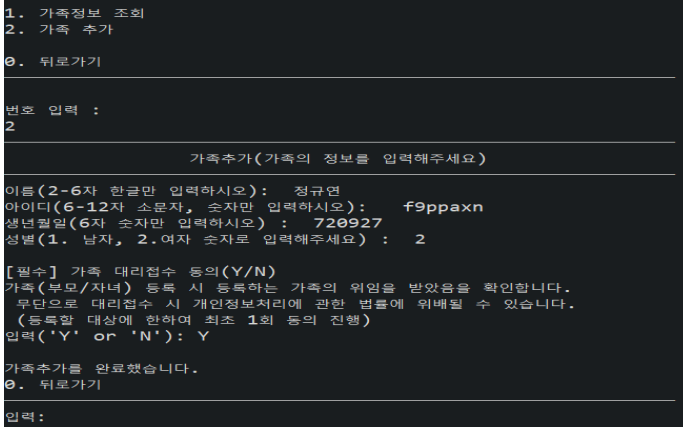
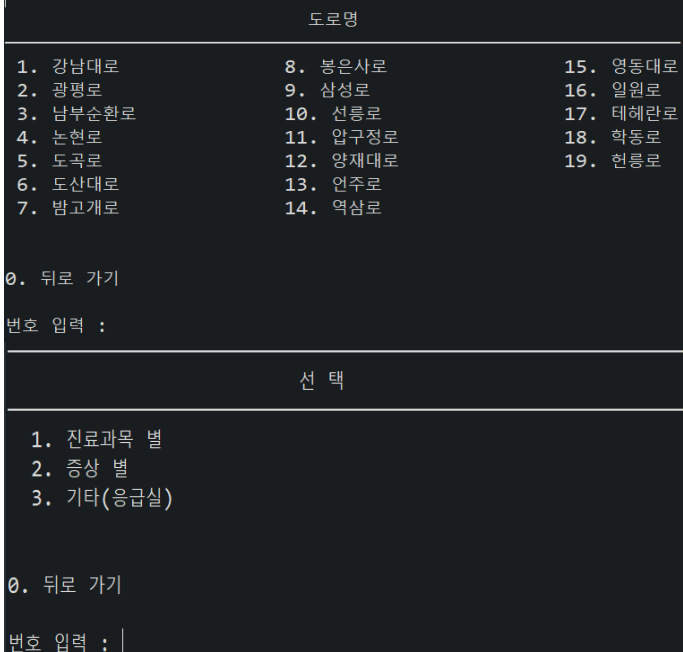
txt 파일	FindHospital.txt 	Select.txt 
	reserve.txt 	
	review.txt 	



6. 순서도



7. 구동화면

화면	설명
	<p>프로그램 시작 시 출력되는 메인 화면</p> <p>사용 기술: Scanner</p>
	<p>세부내용: 가족 추가 기능(기획 의도)</p> <p>1. 가족의 정보를 입력 받아 회원데이터 정보와 일치하면 로그인한 회원의 아이디가 가족코드가 되어 가족 추가 가능</p> <p>사용 기술: scanner, filewriter</p>
	<p>세부내용: 위치 세분화 및 선택형 목차</p> <p>1. 지리적 위치까지 도로명으로 세분화 2. 위치 선택 후 선택지도 세분화</p> <p>사용기술: 데이터를 이용하여 관련 정보 추출(강점: 정보 수정 용이)</p>

상세정보를 확인하고 싶은 병원 번호를 입력해주세요.
번호 입력 : 259

1. 의사정보 확인

2. 리뷰확인

3. 예약하기

0. 메인으로 이동

번호 입력 : 1

의사정보

번호 : 259

이름 : 현재마

성별 : 남

※ [Enter]를 입력하면 이전 화면으로 이동합니다. ※

1. 의사정보 확인

2. 리뷰확인

3. 예약하기

0. 메인으로 이동

번호 입력 : 2

22년04월12일 게시

평점 : ★★★★★

아이디 : 6dczt982

받은 진료 : 치과

의사 이름 : 현재마

리뷰 : 원장님 과잉진료 안하시고, 정말 꼼꼼하게 봐주세요.

이번에 신경치료하고 크라운 씌우는데 제작부터

교함까지 죄송할 정도로 신경 너무 써주셨어요.

너무 감사합니다.

※ [Enter]를 입력하면 이전 화면으로 이동합니다. ※

예약을 희망하는 날짜를 선택해주세요.

오늘은 17일 입니다.

월

화

수

목

금

토

일

18

19

20

21

22

23

휴무

날짜 입력 : 20

현재 고객님의 가족 구성원입니다.
예약을 희망하시는 구성원을 선택해주세요.

1. 아이디 : ofsu6gxm , 이름 : 권준연 , 주민번호 : 670916

2. 아이디 : 6ovfvj , 이름 : 권용현 , 주민번호 : 950515

번호 입력 : 1

예약이 완료 되었습니다.

예약 내역은 MyPage에서 확인 하실 수 있습니다.

권준연님 20일, 11시

전태찬 선생님과 예약 되었습니다.

[Enter] 를 입력하시면 의료기관 정보 페이지로 이동합니다.

세부내용: 상세내역 확인

1. 상세정보를 확인하고 싶은 의료기관
번호를 선택 후, 상세정보 메뉴의 번호 입력
> 해당 정보(의사정보, 리뷰) 출력

사용기술: Scanner, Buffered Reader,
FileReader, split, equals

세부내용: 병원 예약 시 날짜 선택 화면

1. 희망하는 날짜를 선택 후 입력
* 당일 기준 7 일만 출력, 일요일은 휴무로
예약불가

사용기술: Calendar, ArrayList, Scanner

세부내용: 대리 예약 시스템(기획 의도)

1. 대리 예약을 위해 가족 정보 출력

2. 번호로 대상 지정 가능

3. 완료 내역 출력으로 대리 예약 재확인
가능

시용기술: Scanner, FileReader, FileWriter

<div>관리자</div> <div> 번호를 선택해주세요. 1. 의료기관 등록 2. 의료기관 수정 3. 회원 관리 4. 통계 0. 로그아웃 </div> <div>번호 입력 :</div>	<p>세부 내용: 관리자 메인 화면</p> <ol style="list-style-type: none"> 1. 관리자 권한으로 진행 가능한 목록 2. 각 항목 선택시 항목에 해당하는 화면으로 이동 <p>사용기술: Scanner</p>
<div>진료병원 : 김이박이비인후과</div> <div>병원 이용 날짜 : 220415</div> <div>별점 : ★ ★ ★ ★ ★</div> <div>리뷰를 삭제하시겠습니까?</div> <div> 1. 예 2. 아니오 </div>	<p>세부 내용 : 회원 리뷰 삭제</p> <ol style="list-style-type: none"> 1. 회원이 등록한 리뷰 관리자 권한으로 삭제하는 기능 <p>사용 기술 : ArrayList, remove()</p>
<div>저장된 전체 병원 개수 : 8개</div> <div>가입한 회원 수 : 5명</div> <div>예약된 건 수 : 1건</div> <div>[Enter]를 누르면 메인메뉴로 돌아갑니다.</div>	<p>세부 내용 : 통계 화면</p> <ol style="list-style-type: none"> 1. 저장되어 있는 데이터의 통계 출력 2. 데이터 수정 시 자동으로 업데이트 <p>사용 기술 : ArrayList, BufferedReader, FileReader</p>

8. 후기

첫 프로젝트라 모두가 서툴고 의견내기 어려웠지만 서로를 의지했으며 자신이 모른다는 것에 대해 부끄러워하지 않고 먼저 도움을 청했기 때문에 오류가 적었고 빠르게 해결이 가능해 기획의도와 맞는 프로그램을 짜는 것이 가능했다.

꼼꼼한 요구분석도와 순서도로 인해 코딩 구현 자체는 빨랐으며 진행했던 프로젝트를 발표할 땐 정확하게 기획한 의도를 드러냈으며 문제점 파악과 그것에 대한 해결 그리고 오류 없는 구현까지 완벽하다고 말할 순 없었지만 오류를 잡아냈다는 것에 대해 큰 의의를 둔다.

그러나 잘못 생각했던 점은 터미 데이터를 미리 만들었음에도 불구하고 각자 프로그램을 구현할 때 따로 데이터를 추가했다는 것이다. 그 점까지는 생각에 미치지 못했다. 단순히 프로그램 구현을 완성시키는 데 중점을 두었기 때문에 그런 사실을 나중에야 알아차리게 되었다는 점을 아쉽다고 뱉을 수 있다. 이유는 역시 오류가 나서 결과를 산출하지 못할 거 같다는 불안감이 더 컸기 때문인 거 같다. 다음 프로젝트에서는 이 점을 교훈 삼아 발전한 개발자의 모습이 될 거 같다.