<팀 그릴그린 - 캡스톤 디자인 최종발표 대본>

|  |
| --- |
| 인사말 / 목차 / 팀 구조 |
| 팀 그릴그린의 팀장 오혜진입니다. 발표를 시작하겠습니다.  목차는 팀 구조, 프로젝트 설명, 산출물 소개, 결과물 및 테스팅 순서입니다.  그릴그린은 프로젝트별 조직이라고 가정합니다. 각자 맡은 역할은 오혜진 백엔드, 천세륜 프론트엔드, 오규진 빅데이터 분석 및 시각화입니다. |

|  |
| --- |
| CorpCollector란? / 프로젝트 기획 배경 / 사용한 HW, SW 목록 |
| CorpCollector(코프콜렉터)는 저희 팀이 기획한 프로젝트로, 소비자 관점에서 선한 기업 정보를 제공하는 웹 사이트입니다. 선한 기업의 예시로는 사회적 기업, 녹색 기업 등이 있습니다. 이런 기업 목록은 공공데이터 포털에서 파일 데이터, 또는 API로 제공합니다. 일반 사용자들이 보기에는 가독성이 떨어지는 부분을 고려해 웹으로 만들었습니다. 또한 데이터 분석 자료도 제공해서 자료의 질을 높였습니다.  이 서비스를 기획하게 된 배경을 설명하겠습니다. 특정 기업들의 논란으로 인해, 소비자들의 불매 행위가 불거지면서 해당 기업들의 매출이 하락하는 증세를 보입니다. 이는 부도덕한 행위를 한 기업 제품보다는 다른 기업 제품을 소비하자는 의견이 있었기 때문입니다. 따라서 선한 기업 목록을 CorpCollector에서 제공하여 사람들의 이로운 소비 행위 및 기업의 사회 환원을 장려하고자 합니다.  사용한 하드웨어는 개인용 개발 PC 3대, 소프트웨어는 깃허브, 태블로, DBMS, 이클립스 외 여러 툴이 있습니다. WAS(웹 애플리케이션 서버)는 아파치 톰캣을 사용했습니다. |

|  |
| --- |
| 퍼트 차트 & 간트 차트 / 사이트맵 구조 |
| 이어서 프로젝트 산출물을 소개하겠습니다.  전체 개발 주기를 표현한 퍼트 차트입니다. 사이트 맵 구성 이후에는 각 파트에 따라 설계 작업을 진행했습니다. 이후에는 약 1주일 단위로 개발/보완/수정 주기를 거치며 작업했습니다.  또한, 11월 초에 마감 기간이 늘어났기 때문에 유지보수와 추가 작업을 진행했습니다.  추가 일정까지 표시한 간트 차트는 이런 모습입니다. 4차 주기에서 백엔드 마무리 작업이 늦어졌지만, 이후 일정은 원활하게 진행되었습니다.  사이트 맵은 총 5가지 주메뉴로 이루어져 있습니다. 주 메뉴 카테고리로는 서비스 소개, 기업 찾기, 정보나눔, 커뮤니티, 마이페이지, 로그인 및 회원가입이 있습니다. 상세한 설명은 다음 챕터에서 설명하겠습니다. |

<프론트엔드>

|  |
| --- |
| 프론트엔드 |
| 이제 결과물 및 테스트 결과를 설명하겠습니다. 먼저, 프론트엔드 파트입니다. 프론트엔드 담당자 천세륜은 이클립스 개발 환경에서 jsp, js, css 등을 이용해 사이트를 구현했습니다. |

|  |
| --- |
| 화면 구성 소개 |
| 인덱스 페이지부터 간략하게 설명하겠습니다. 인덱스 페이지의 가운데에 ‘CorpCollector’ 로고와 검색바를 배치하여 사이트의 메인 기능인 검색 서비스를 빠르게 이용할 수 있도록 했습니다.  서비스 소개의 하위 메뉴인 ‘서비스 개요’에는 사이트 제작 배경, 각 메뉴 설명이 있고 ‘개발 과정’에는 각 개발 파트에 대한 간략한 설명과 설계도 등의 산출물이 있습니다.  다음으로 상위 메뉴인 ‘기업 찾기’에서는 기업 종류에 상관없는 통합 결과가 나타납니다. 사이드 바 하단의 기업 순위 정보를 참고하여 검색에 도움을 받을 수 있습니다.  기업 종류를 선택하면 선택한 기업 종류에 해당하는 리스트가 나타납니다. 예시로 ‘녹색 기업’을 클릭하면, 녹색 기업 리스트만 나타나고 녹색 기업 내의 검색 결과가 출력됩니다.  기업 리스트 중 하나를 선택하면 상세 정보 페이지로 이동하고 로그인 한 유저는 관심 기업 등록을 할 수 있습니다.  관심 기업으로 등록한 기업은 채워진 별 표시가 됩니다. 목록의 별을 직접 클릭해서 관심 기업 등록, 삭제도 가능합니다.  정보 나눔 메뉴의 ‘기업 데이터 분석’에서는 기업 종류별로 데이터를 분석하여 그래프로 나타내 사용자에게 정제된 정보를 전달합니다.  다음 하위 메뉴인 ‘기업 기사 모음’은 기업들의 최근 기사를 리스트로 정리한 페이지로 링크를 클릭하면 기사 원문 사이트로 이동합니다.  ‘커뮤니티’ 메뉴를 설명하겠습니다. 공지사항 게시판에는 관리자만 글을 쓸 수 있으며, 그 외에는 고객후기 게시판과 같은 기능을 가집니다. 공개 글은 모든 유저가 볼 수 있고 비공개 글은 작성자와 관리자만 확인할 수 있습니다.  모든 글 작성 페이지는 로그인 한 사용자만 이용 가능하며 체크박스로 글과 작성자 공개/비공개를 설정할 수 있습니다.  마지막 메뉴인 ‘마이페이지’는 기업과 관련된 두 가지 하위 메뉴만 설명하겠습니다.  먼저, ‘관심 기업’ 페이지에서는 ‘기업 찾기’에서 관심 기업으로 등록한 기업 목록을 확인할 수 있습니다. 기업 종류별로 분류되어 나타나며 체크박스를 이용해 선택 기업을 관심 기업에서 삭제할 수 있습니다.  ‘최근 검색 기업’ 페이지는 로그인 한 사용자가 ‘기업 찾기’에서 상세 정보를 확인한 최근 50개 기업 목록을 보여줍니다. |

<백엔드>

|  |
| --- |
| 백엔드 |
| 백엔드 파트입니다. 백엔드 담당자 오혜진은 웹 사이트의 기능 구성 및 구현을 진행했습니다. 해당 파트는 웹 사이트 구조, 클래스 다이어그램 등으로 이루어져 있습니다. |

|  |
| --- |
| 웹 구조 |
| 우선, CorpCollector의 웹 구조를 말씀드리겠습니다. 브라우저가 컨텐츠를 요청하면 웹 서버가 정적인 컨텐츠를 제공하거나 요청을 웹 컨테이너에 보냅니다. 그중 웹 컨테이너는 DB에서 데이터를 가져와서 비즈니스 로직에 맞는 결과를 만들어 제공합니다. 여기에 사용된 웹 애플리케이션 서버는 아파치 톰캣, 데이터베이스는 MariaDB입니다. |

|  |
| --- |
| 디자인 패턴 |
| 디자인 패턴은 MVC 패턴과 프론트 컨트롤러, 커맨드 패턴을 사용했습니다. 모델, 뷰, 컨트롤러를 분리해 관리가 편하도록 설정했습니다. 모든 요청을 프론트 컨트롤러로 보내서 URI 매핑을 통해 어떤 커맨드를 실행할지 결정합니다.  커맨드 패턴은 프론트 컨트롤러 서블릿이 직접 요청을 처리하지 않고, 커맨드 클래스가 처리하게 해줍니다. 이러한 MVC 모델을 사용하면 용도에 따라 코드를 분리할 수 있고, 따라서 중복코딩을 최소화할 수 있습니다. 또한, 유지보수성과 유연성이 향상하고, 프론트엔드 작업자도 뷰, 화면 인터페이스만 보고 작업할 수 있으므로 팀 작업에 적합하다고 판단했습니다. |

|  |
| --- |
| 클래스 다이어그램 |
| 회원 관리 커맨드들은 MemberDAO를 호출합니다. 회원가입 시에 아이디 중복 방지를 위해 아이디 체크 커맨드를 사용합니다. 그 외에는 회원 관리에 필요한 기본적인 커맨드들입니다.  다음으로, 이메일 인증 기능을 보겠습니다. 회원가입에 기재한 이메일의 계정주만 개인 정보에 접근할 수 있게 해주는 기능입니다. ServerLogin 클래스에 인증 코드를 적재한 뒤, 이메일 체크 커맨드에서 사용자가 입력한 값과 비교합니다.  기업 찾기 클래스 다이어그램은 예시로 녹색 기업, 통합기업 관련 클래스만 다루겠습니다. 기업 유형만 다르고 기본적인 컨트롤 로직은 비슷하므로, 한 커맨드에서 기업유형별 검색을 수행할 수 있도록 구성했습니다.  마이페이지에는 여러 테이블이 복합적으로 사용되므로 커맨드별 쓰이는 DAO가 다양합니다. FavoriteCorpDAO의 메소드를 이용해서 관심 기업 등록, 삭제를 진행하고 RecentSearchCorpDAO 메소드로 상세 정보를 본 기업 데이터를 기록합니다. |

<빅데이터>

|  |
| --- |
| 빅데이터 |
| 마지막 부 챕터, 빅데이터 파트입니다. 빅데이터 파트 담당자 오규진은 데이터 분석 및 시각화, 기사 크롤링을 진행했습니다. 내용은 데이터 전처리 및 시각화 과정 등으로 구성되어 있습니다. |

|  |
| --- |
| 데이터 프로세스 구조 |
| 데이터 프로세스 구조를 설명하겠습니다.  기업 데이터는 공공데이터 포털에 있는 기업 데이터를 사용했습니다.  프로세스 순서는 다음과 같습니다. 파이썬으로 open api를 사용하여 공공데이터 포털에서 데이터들을 받습니다.  데이터들의 raw데이터들을 DB에 적재할 수 있도록 데이터 전처리를 진행합니다. 그 후, 파이썬 pymysql(파이마이sql), sqlalchemy(sql알케미)를 사용하여 DB에 연결, 적재하였습니다. |

|  |
| --- |
| 데이터 전처리, 적재 과정(코드) |
| Raw데이터를 수집하고 DB에 적재하기 위해선 전처리 과정이 필요합니다. Raw 데이터의Null값과 중복 데이터는 직접입력 하거나 삭제 등으로 결측치 처리하였고, 지역 구분을 정확하게 하기 위한 텍스트 전처리도 진행했습니다.  [데이터 적재 코드로 넘어가며]  데이터 적재는 아까 말씀드린 것처럼 파이썬 pymysql(파이마이sql), sqlalchemy(sql알케미) 라이브러리를 사용하여 SQL에 연결 후, 각 카테고리 기업 테이블에 데이터를 적재하였습니다. |

|  |
| --- |
| 데이터 시각화-대시보드 |
| 데이터 시각화 대시보드 구성입니다.  데이터를 시각화하는 툴(Tool)은 Tableau(태블로)를 사용했습니다.  태블로는 여러 그래프를 만들어 하나의 대시보드로 표현할 수 있습니다.  구성된 그래프를 설명하겠습니다. 첫 번째 그래프는 기업들의 업종 통계를 나타내는 그래프이고 트리 그래프로 표현하였습니다.  [다음 장면으로 넘어가며]  두 번째 그래프는 지역별로 기업의 업종 통계를 나타내는 그래프이고 누적 막대그래프로 표현하였습니다.  [다음 장면으로 넘어가며]  세 번째 그래프는 해당 카테고리에 선정된 기업들의 숫자를 나타낸 그래프이고 막대 그래프로 표현하였습니다.  네 번째 그래프는 기업의 지역 분포를 숫자로 표현하는 시각화이고 해당 그래프는 표로 표현하였습니다. |

|  |
| --- |
| 기사 모음(크롤링) 과정 |
| 기업 기사를 크롤링하는 과정입니다.  네이버 뉴스에서 기업 기사를 크롤링하여 DB에 적재했습니다. 크롤링하는 과정에서 파이썬 beautifulsoup(뷰티풀숲), request라이브러리를 사용하였습니다. |

|  |
| --- |
| 맺음말 |
| 이상으로 그릴그린의 웹 프로젝트 발표였습니다. 기타 자세한 프로젝트 작업물은 아래 첨부한 사이트로 들어가서 볼 수 있습니다. 감사합니다. |