

프로젝트명	블록체인 소셜 플랫폼 steemit 친구를 찾아주는 stemfriend 개발
참여기간	2017.07 ~ 2017.07
프로젝트 개요	블록체인 소셜 플랫폼 중 하나인 steemit 에서 가장 친한 친구를 찾아주는 웹 서비스를 만듦. 페이스북에서 가장 친한 친구를 찾아주는 서비스를 보고 아이디어를 생각하게 됨.
주요 역할 및 담당	Django 프레임워크를 이용하여 블록체인 데이터를 가져와 분석하고, 웹사이트에 뿌려주는 역할을 함.
발생 문제/ 해결방법	<p>※ 처음에 단순 구현 후에는 Bad Gateway 가 자주 발생. 한 유저가 들어간 후 몇 초간의 계산 후에 결과가 나오게 되는데, 동시 접속이 많은 경우 이러한 문제가 발생했음.</p> <p>▶ Nginx 및 Django 설정을 통해 다중 접속시에도 서버 접속이 원활해지도록 해결함.</p> <p>※ 사용자의 인터랙션 히스토리를 가져와야 하는데, steemit 블록체인에서 제공하는 API를 굉장히 많이 호출해야 했고, 그로 인해 한 사람당 30초 가까이 기다려야 하는 문제가 발생하였음.</p> <p>▶ steemit 블록체인 데이터를 SQL처럼 처리할 수 있는 SteemSQL을 사용하여 데이터 베이스 내에서 최대한 효율적인 처리를 한 후에 결과를 가져올 수 있도록 구현하여 해결함.</p>
성과	<p>블록체인에 올라가 있는 소셜 데이터에 접근해볼 경험을 가짐.</p> <p>Django 프레임워크로 개인 토이 프로젝트를 처음 해봄.</p> <p>웹 서버 엔진 Nginx, uwsgi 를 직접 세팅.</p> <p>속도 개선을 위한 고민을 해본 경험.</p>

<b>프로젝트명</b>	키다리 은행 클라이언트 및 서버 개발
<b>참여기간</b>	2016.06 ~ 2016.12
<b>프로젝트 개요</b>	<p>대학교 학생들을 위한 은행 협동 조합인 '키다리 은행' 아웃소싱 개발 해준 프로젝트. 앱으로 대출 신청을 할 수 있고, 대출을 해 줄 사람들은 어떤 대출 신청들이 올라와 있는지를 확인할 수 있어야 함. 본인의 키다리 은행 내 보관된 금액과 기존 기록들을 확인할 수 있어야 함. 다만 실제로 돈이 오고 가는 일은 기존의 조합원들이 수행하고, 클라이언트 앱은 계좌 관리 및 대출 정보 확인 등의 목적을 수행함.</p>
<b>주요 역할 및 담당</b>	<p>아이오닉 프레임워크를 이용하여 키다리 은행 클라이언트 앱을 개발하고, 앱 구동을 위한 API 서버, 그리고 관리자 페이지 등을 만들어 줌.</p>
<b>발생 문제/ 해결방법</b>	<p>※ 비록 인간이 개입하는 일에 보조적인 역할을 하는 서비스였지만, 그래도 돈이 오고 가는 일이라 잘못 동작할 경우 민감한 문제가 발생할 수 있었음.</p> <p>▶ 특정 버튼을 누르는 등의 동작을 할 때 중간에 문제가 발생하여 수행이 실패하더라도 나중에 기록을 보고 맞춰볼 수 있도록 로그 기록을 설계하였음. 또한, 이렇게 실패한 경우에도 문제가 없도록 반드시 상태가 롤백되도록 예외 처리를 함.</p> <p>※ 푸쉬 기능을 설계할 때, 한 유저가 여러 장치를 사용할 수 있는 상황을 고려하지 못하여 가장 마지막으로 로그인한 장치에 대해서만 푸쉬가 알람이 가는 문제가 발생.</p> <p>▶ 데이터베이스 테이블을 추가하여 각 유저별 여러 가지 디바이스 아이디를 기록하도록 구현하고, 푸쉬 기능에서 그 유저가 가지고 있는 모든 (푸쉬가 허락된) 디바이스에 보내도록 구현</p>
<b>성과</b>	<p>처음으로 프론트부터 서버까지 혼자 개발해본 경험</p> <p>메시지 푸쉬 서버를 개발한 경험</p>

프로젝트명	협업 필터링 추천 시스템 성능 향상 연구를 주제로 한 산학 프로젝트
참여기간	2015.01 ~ 2015.10
프로젝트 개요	<p>한양대학교 데이터 마이닝 연구실에서 협업 필터링 추천 시스템 성능 향상을 연구 주제로 해서 산학 프로젝트를 수행하였고, 이 주제로 졸업 논문을 작성하였다. 본 프로젝트에서는 신뢰 네트워크에서 직접 연결된 이웃들을 활용한 대치 방법(평점 행렬을 채우는 방법)을 제안한다.</p> <p>기존의 대치 방법은 많은 수의 이웃들이 평점을 남긴 아이템들에 대해서만 평점 값을 예측하여 대치하였다. 하지만 직접 연결된 이웃의 경우 단 한 명만이 평가한 아이템에 대해서라도 평점 값을 예측하여 대치한다.</p>
주요 역할 및 담당	<p>본 논문의 제안 방법의 알고리즘을 C# 및 MATLAB 코드로 작성하고, 실험 결과를 정리하여 논문을 쓰는 데 반영함.</p>
발생 문제/ 해결방법	<p>※ 신뢰 네트워크에서 접근할 수 있는 유저들이 많이 평가한 아이템을 prediction 하여 imputation 하는 것이 실험 방법의 핵심.</p> <p>※ 실험을 위해서 사용하는 데이터가 굉장히 커서 메모리 문제가 나는 경우가 있었음.</p> <p>➤ 신뢰 네트워크에서 특정 거리 안에 도달할 수 있는 유저들을 neighbor로 설정하고, prediction해야 할 아이템을 저장하고 관리하는 중에 Out Of Memory 예외처리가 많이 발생했고, 이를 해결하기 위해서, 각 원소마다 1비트를 사용하는 Bit Array를 사용하여 해결하였음.</p> <p>※ 몇몇 기능들(특정 사용자가 특정 아이템에 대해서 평가했는지? 몇점으로 평가했는지? 특정 사용자의 평균 평점 등) 을 O(1) 시간안에 작동하도록 구현하여야 했음.</p> <p>➤ 평점 행렬을 저장할 때, 이차원 배열이 아닌, Dictionary의 배열과 HashSet의 배열로 잡아서, 평가 여부와 평점 정보들을 동시에 관리하였음. 또, 사용자들의 평균 평점을 미리 계산.</p>
성과	<p>대학원 실험실에서 어떠한 연구를 하는지 알 수 있는 기회가 되었다. 이 때, 빅데이터를 다룰 때 발생하는 메모리, 수행 속도 문제를 해결하기 위해서 세밀한 부분에서 퍼포먼스를 고려하면서 구현해야 했다.</p> <p>이전까지는 특정 기술이나 언어를 이용해서 필요한 기능을 구현하는 데에 그쳤다면, 이 프로젝트를 하면서 자료구조, 메서드를 시간 공간 복잡도 측면에서 고민을 해가며 설계해야 했다. C#에서 제공하는 Collections 클래스들의 메서드들의 성능, 메모리 비용 등을 실험한 결과들을 많이 참고하면서 실험을 수행했다.</p>

<b>프로젝트명</b>	Waterful (애플 워치 응용프로그램 개발)
<b>참여기간</b>	2015.09 ~ 2015.11
<b>프로젝트 개요</b>	애플워치 응용프로그램 개발을 주제로 시작한 프로젝트, 사용자의 물 섭취를 유도하는 건강 어플리케이션이다. 애플 워치에서 제공하는 차세대 인터페이스들을 활용하여 기존의 유사 어플리케이션보다 더 손쉽고 간편하게 사용하는 것이 가능하다.
<b>주요 역할 및 담당</b>	사용자의 건강 정보와 iOS 의 헬스킷 프레임워크를 사용하여 사용자에게 유의미한 알림을 줄 수 있도록 물 섭취 패턴을 예측하는 알고리즘을 구현
<b>발생 문제/ 해결방법</b>	<p>※ 사용자의 자연스러운 물 섭취를 위해서는 백그라운드 모드에서의 로직이 많이 필요했는데, 헬스킷 프레임워크의 쿼리는 백그라운드 모드에서 사용할 수 없었다.</p> <p>▶ 헬스킷 쿼리 중 HKObserverQuery 를 사용하면 백그라운드에서도 헬스킷 데이터 샘플이 추가되는 것을 감지할 수 있다. 이 쿼리를 사용하여 사용자의 걸음과 물 섭취 기록에 대한 샘플이 추가되는 것을 수집하여 수집되는 '시간'을 기준으로 시계열 패턴을 수집하고, 사용자가 일어난 시간을 예측하거나 물 마실 시간을 패턴하는 알고리즘에 사용할 수 있었다.</p> <p>※ 사용자의 물 섭취 패턴을 적절하게 예측하기 위해서 필요한 알고리즘을 구현할 때, 임의로 평균값을 내서 구하는 것보다는 어느 정도 신뢰 있는 알고리즘을 사용하는 것이 좋을 것이다.</p> <p>▶ 기존의 연구 중 K-Means 클러스터링과 AprioriAll 알고리즘을 활용하여 재고 경향의 시계열 패턴을 분석하는 방법에 대한 연구 논문이 있어, 그 방법을 참고하여 물 섭취 패턴을 예측하는 알고리즘을 구현함.</p>
<b>성과</b>	이 프로젝트를 통해서 Swift 를 사용했고, iOS 의 헬스킷 프레임워크와 같은 애플의 신 기술을 사용해보는 경험을 얻을 수 있었다. 또, 차세대 플랫폼으로 등장하고 있는 애플 워치 플랫폼에서 개발을 해보는 경험도 얻었다.

프로젝트명	Apus (바이너리 데이터 조작 스크립트 형 언어 개발)
참여기간	2015.07 ~ 2015.08
프로젝트 개요	바이너리 데이터와 적절한 템플릿을 사용자가 입력하면, 사용자가 입력한 템플릿에 맞게 바이너리 데이터를 파싱하여 분석해주고, 그 데이터를 조작하는 스크립트 언어를 작성할 수 있도록 제공
주요 역할 및 담당	<p>바이너리 데이터에 특정 위치, 특정 형태로 값을 적절하게 읽어들이고, 시스템의 엔디안 방식을 고려하여 적절한 값으로 변환하여 넘겨주는 모듈을 작성</p> <p>사용자가 작성한 스크립트들을 스캐닝, 파싱한 후에 각 변수나 함수 사용에 대한 심볼 테이블을 구현함 (Semantic Analysis)</p>
발생 문제/ 해결방법	<p>※ 바이너리 데이터를 생성하는 환경과, 바이너리 데이터를 읽어들이려는 환경의 엔디안 방식이 다를 때, 읽어들이는 바이너리 데이터를 역순화하는 과정을 단순화할 필요가 있었다.</p> <p>▶ C/C++에서 제공하는 union 키워드를 사용하여 단순화</p>
성과	<p>기본적으로 사용자가 제공한 템플릿과 조작을 위한 스크립트들을 분석해야 했기 때문에 컴파일러의 스캐닝, 파싱, 시맨틱 분석 과정을 한 번 공부해야 했다. 또한, 스크립트 형 언어를 만든 것이기 때문에 한 스크립트가 해결될 때마다 시맨틱 분석이 끝날 때마다 스크립트의 실행을 진행하는 가상머신도 구현하였다.</p> <p>이 프로젝트는 SW 마에스트로 과정에서 수행했던 프로젝트로, 오픈 소스 개발자이신 담당 지도 멘토님께 Git 을 사용하는 것을 제대로 배우게 된 계기였다.</p>

프로젝트명	안드로이드 폰을 활용한 방법 시스템 개발
참여기간	2014.09 ~ 2014.12
프로젝트 개요	<p>안드로이드 폰 두 개와 블루투스 제어 전동차인 BeautoRover 로 방법 시스템을 개발하는 프로젝트.</p> <p>안드로이드 폰 하나는 CCTV 역할을, 다른 하나는 BeautoRover 를 제어하는 용도로 사용되었다. CCTV 역할의 안드로이드 폰이 모션 인식과 얼굴 인식을 하였고, 시스템에 등록되지 않은 사용자일 경우 BeautoRover 를 제어하는 안드로이드 폰과 통신하여 BeautoRover 를 작동시켜서 방법 기능을 하는 방식이다.</p>
주요 역할 및 담당	<p>사람 얼굴을 인식하는 기능 (Face Recognition) 과 BeautoRover 와 통신하는 기능 구현</p>
발생 문제/ 해결방법	<p>※ 2 개 이상의 디바이스와의 블루투스 통신을 제어하는 것이 어려움          ► 안드로이드 폰과의 통신은 TCP 통신을 통하여 구현함.</p> <p>※ 얼굴 인식을 위해 OpenCV 를 사용하려고 하였으나, 안드로이드 폰에서의 OpenCV 가 예상보다 속도나 정확도가 좋지 못하였음.          ► 안드로이드 기본 hardware 패키지에서 제공하는 Camera.Face 클래스, FaceDetectionListener 등을 활용하여 구현하였음. 이들을 사용하여 인식된 얼굴의 정보를 알 수 있었고, 그 정보를 기반으로 얼굴을 식별하고 구별하는 기능을 구현하였음.</p> <p>※ 모션 인식이 너무 민감한 문제가 발생하였음          ► 아주 작은 모션 인식으로도 감지가 되는 문제가 발생하여, 수 차례 실험을 통해 pixel, image threshold 를 제대로 설정하여 적절한 모션 인식을 유도함.</p>
성과	<p>안드로이드 플랫폼에서 블루투스 통신 기술, TCP 통신 기술, 얼굴, 모션 인식을 위한 기술 등 다양한 기술을 다뤄볼 수 있고, BeautoRover 를 작동시키기 위한 레퍼런스 문서와 OpenCV 를 대체할 다른 얼굴 인식 방법을 찾다가 안드로이드 hardware 패키지에 대한 레퍼런스 문서를 참고하였다.</p>

<b>프로젝트명</b>	OpenWrt 공유기를 활용한 통합 공유 파일 시스템 개발
<b>참여기간</b>	2014.03 ~ 2014.06
<b>프로젝트 개요</b>	<p>OpenWrt 운영체제가 올라가 있는 공유기에 연결된 모든 디바이스를 관리하고, 그 디바이스들을 통합하여 파일을 관리하는 시스템을 개발</p> <p>인터넷 공유기가 웹 서버 역할을 하고, 그 공유기에 인터넷을 리스하는 여러 모바일 디바이스들이 연결된다. 외부에서 공유기에 인터넷을 리스하는 이 여러 디바이스들의 파일 시스템들을 관리할 수 있는 기능을 제공한다.</p>
<b>주요 역할 및 담당</b>	DB 관리 부분과 서버 웹페이지에서 파일 시스템을 관리할 수 있는 기능을(클라이언트 부분) PHP 로 구현함.
<b>발생 문제/ 해결방법</b>	<p>※ 파일 관리를 FTP 로 하려고 했지만, 서버가 아닌 상황에서 외부에서의 접근이 어렵고, 크로스 도메인 문제가 발생함.</p> <p>▶ FTP 대신 CIFS 를 활용해서 파일 시스템을 원격 마운트하고, 파일 리스팅, 업로드, 다운로드, 삭제 기능들을 직접 구현하였음.</p> <p>※ 공유기에 연결된 디바이스들의 파일 시스템 사용 가능 상태를 확인하는 기능 구현이 어려웠음.</p> <p>▶ PHP 로 시스템 커맨드를 실행시켜서 시스템 파일에 접근한 후에, 얻은 파일을 파싱하여 디바이스들의 정보를 가져옴.</p>
<b>성과</b>	<p>웹 페이지를 구현하기 위해서 HTML, PHP 언어로 작성하였고, DB 기능 (서버에 연결된 디바이스들을 저장하는 역할)을 구현하기 위해서 SQLite3 를 사용하였다. (인터넷 공유기에 MySQL, Oracle DB 가 올라가지 않는다)</p> <p>또, CIFS 기술을 활용해서 연결된 모바일 디바이스의 파일 시스템을 마운트하는 방식으로 대체하면서, 파일 시스템을 관리하는 기능을 직접 구현하였다. PHP, HTML 언어에 한층 더 익숙해지는 계기를 갖게 되었고, SQLite3 와 같은 새로운 DB 도 사용하였다</p>