스마트 ERP 시스템

종합설계 프로젝트

2조

과목명 : 종합설계

학과 : 컴퓨터공학과

이름 : 맹주완, 김지수, 추다혜

담당교수님 : 김우성 교수님

제출일 : 2019-12-09

목차

[1. 소개 3](#_Toc26784902)

[1.1 제안 배경 3](#_Toc26784903)

[1.2 유사 프로젝트 3](#_Toc26784904)

[1.3 최종 목표 4](#_Toc26784905)

[2. 설계 4](#_Toc26784906)

[2.1 시스템 구성도 4](#_Toc26784907)

[2.2 구성 요소별 처리 과정 4](#_Toc26784908)

[3. 구현 결과 4](#_Toc26784909)

[3.1 구현 환경 4](#_Toc26784910)

[3.2 적용된 주요 기법 4](#_Toc26784911)

[3.3 구현 결과 4](#_Toc26784912)

[4. 업무별 참여 및 완성도 4](#_Toc26784913)

[5. 배운 것 및 느낀 점 4](#_Toc26784914)

1. 소개
   1. 제안 배경

현재 우리나라에서 사용하는 ERP 시스템들은 하드디스크에 설치해서 사용하는 프로그램들이 대부분이고 웹을 이용한 ERP 시스템은 하나 밖에 없었고 해당 프로그램도 인공지능을 이용한 기능과 앱과 블루투스 비콘을 이용해 재고의 위치를 추적하는 기능은 없었다.

그래서 우리는 이점에서 착안해 인공지능과 앱을 기존에 있는 웹 ERP 시스템에 추가해 기존의 웹 ERP 시스템이 가지고 있던 기능인 입고, 출고 관리 기능이나 인사관리 기능, 입고 물품 추가나, 출고 물품 추가 등의 기능뿐만 아니라 인공지능을 이용해 여태까지의 입고내역, 출고내역을 분석해서 달마다 무엇을 입고하면 좋을지 추천해주고 또한 현황을 그래프로 표시해주는 기능과 웹 ERP와 앱을 연동하여 블루투스 비콘을 활용해서 창고 안에 있는 재고의 위치를 표시해주는 기능까지 추가해 새로운 웹 ERP 시스템을 만들고자 했다.

* 1. 유사 프로젝트

※이카운트 ERP



웹 기반이여서 인터넷만 연결되면 사무실 밖에서도 언제든지 업무를 조회하고 처리할 수 있으며, 별도의 설치 과정이 필요하지 않기 때문에 도입하는 즉시 사용이 가능하다는 장점을 공유한다.

또한 개인 pc가 아닌 서버에 자료가 저장되기 때문에 따로 백업이 필요가 없다는 장점과 모바일 앱과 연동이 가능한 부분 또한 해당 프로젝트와 공유한다.

차이점으로는 이카운트 ERP는 인공지능을 이용해서 입고나 출고 현황을 분석해 물품을 추천하는 기능은 가지고 있지 않았으며 또한 창고 내에서 물품의 위치를 파악해서 그것을 웹 상에서 표시해주는 기능 또한 가지고 있지 않았다.

1.3 최종 목표

※웹 기능

1. 메인 화면

ERP 시스템의 메인화면이며, 공지사항, 달력 그리고 금일 물품 입출고 현황을 표시하게 했다.

2. 로그인

ERP 시스템은 지정된 사용자 아니면 사용할 수 없어야 하기 때문에 로그인을 하지 않으면 ERP 시스템 자체를 이용할 수 없게 했다. 권한은 일반 사원 권한과 관리자 권한이 있다.

3. 회원정보 변경

회원정보를 변경할 수 있는 페이지며 여기서 직급이나 부서 이름이나 핸드폰 번호 등의 개인정보도 바꿀 수 있다.

4. 상품관리

4.1 입고목록

입고물품을 확인하는 창이며, 검색은 물품명 날짜로 하게끔 만들었다.

4.2 출고목록

출고물품을 확인하는 창이며, 검색은 물품명 날짜로 하게끔 만들었다.

4.3 불량목록

불량물품을 확인하는 창이며, 검색은 물품명 날짜로 하게끔 만들었다.

4.4 신규물품등록

신규물품등록을 하는 창이며 해당 물품의 번호나 이름 그리고 물품의 사진까지 여기서 등록한다.

5. 스마트관리

5.1 제품추천 목록

인공지능으로 돌린 제품추천목록을 날짜를 입력하면 출력하게 되어있으며, 해당 정보를 그래프와 리스트로 표현해서 보여준다.

6. 인사관리

6.1 인사정보

인사정보를 확인하는 창이며 여기에서 사원들의 개인정보나 부서 등을 열람이 가능하다.

6.2 가입승인

가입을 요청해 승인이 필요한 사람들이 명단을 해당 페이지에서 보여주며, 일관선택해 승인이 가능하도록 했다.

7. 조회관리

7.1 공지사항

공지사항 글들을 보여주고 검색할 수 있게 하며 또한 새로 쓸 수 있게 해주는 창이다.

7.2 입고량조회

입고량을 날짜별로 검색해 자료를 가지고와 리스트와 그래프로 표현해주는 창이다.

7.3 출고량조회

출고량을 날짜별로 검색해 자료를 가지고와 리스트와 그래프로 표현해주는 창이다.

7.4 재고량조회

재고량을 날짜별로 검색해 자료를 가지고와 리스트와 그래프로 표현해주는 창이다.

※사용자 앱 기능

1. 제품 추천 목록

원래라면 제품 Rssi 위치를 이용해서 제품의 위치를 대략적으로 잡아주는 기능이었으나 Rssi 위치가 값이 굉장히 부정확해서 위치가 제대로 안 나왔기에 도중에 웹의 제품추천목록을 앱 서버를 통해 받아 앱에 띄워주는 기능으로 바꾸었다.

2. 출고 확인

웹의 출고 확인 페이지에 있는 리스트 항목들을 가져와서 보여주고 해당 물품을 카카오톡으로 전송하는 기능.

3. 입고 확인

웹의 입고 확인 페이지에 있는 리스트 항목들을 가져와서 보여주고 해당 물품을 카카오톡으로 전송하는 기능.

4. 로그인 기능

로그인을 하지 않으면 앱에 접속할 수 없게 만들었으며, 앱 서버로부터 오는 모든 정보도 로그인을 한 후에 전송되게끔 만들었다.

5. 공지확인 기능

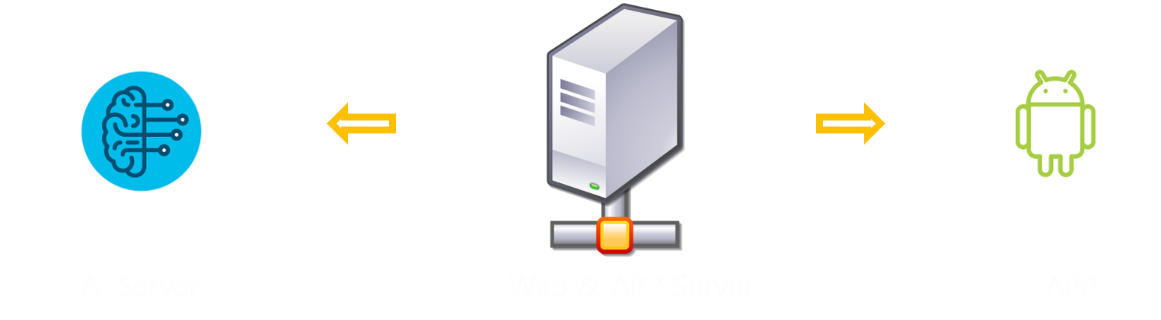
웹에 있는 공지를 가지고 와서 보여주는 기능

6. 카카오톡 연동기능

리스트 항목 클릭시 카카오톡으로 해당 물품관련 정보 전송 기능

1. 설계

2.1 시스템 구성도



웹을 통해 입고, 출고 등의 정보를 DB에 입력하면, 해당 DB를 열람해서 인공지능이 정보를 분석해 분석한 결과를 다시 웹에 돌려주고 웹에서 나온 결과를 앱 서버를 거쳐서 사용자 앱에 보여주게 했다.

웹은 서버와 프론트가 합쳐져 있으며, DB에 기록된 정보를 서버가 받아오면 프론트에 표시한다.

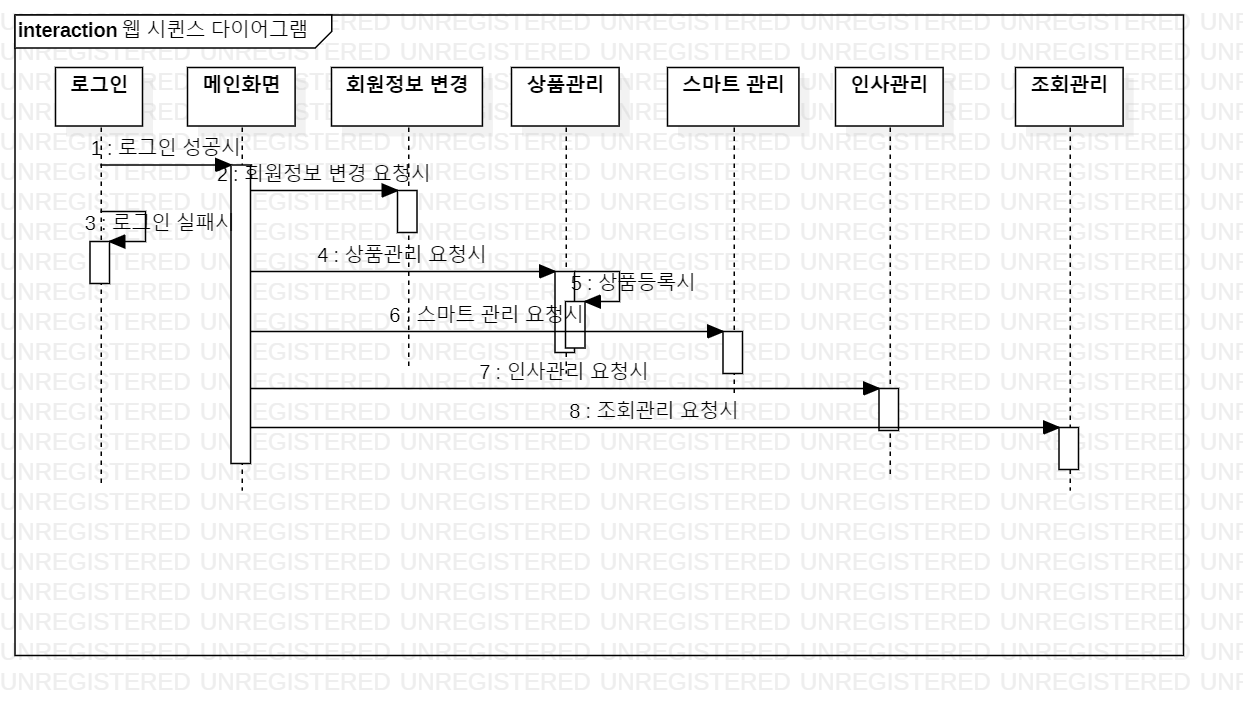
인공지능은 AI 서버가 별도로 있으며 해당 서버가 돌아가 다음 추천 목록을 계산해내는 것은 자정 12시이며, 앱은 메뉴를 열람하면 해당 메뉴의 정보가 업데이트되게 끔 만들었다.

앱은 웹이 서버와 프론트가 합쳐져 있는 관계로 앱 서버와 웹 서버가 별도로 있으며

앱서버에서 DB값을 받아와 앱 화면에 표현해주는 형식이다.

2.2 구성 요소별 처리 과정

※웹 시퀸스 다이어그램



※앱 시퀸스 다이어그램



1. 구현 결과

3.1 구현 환경

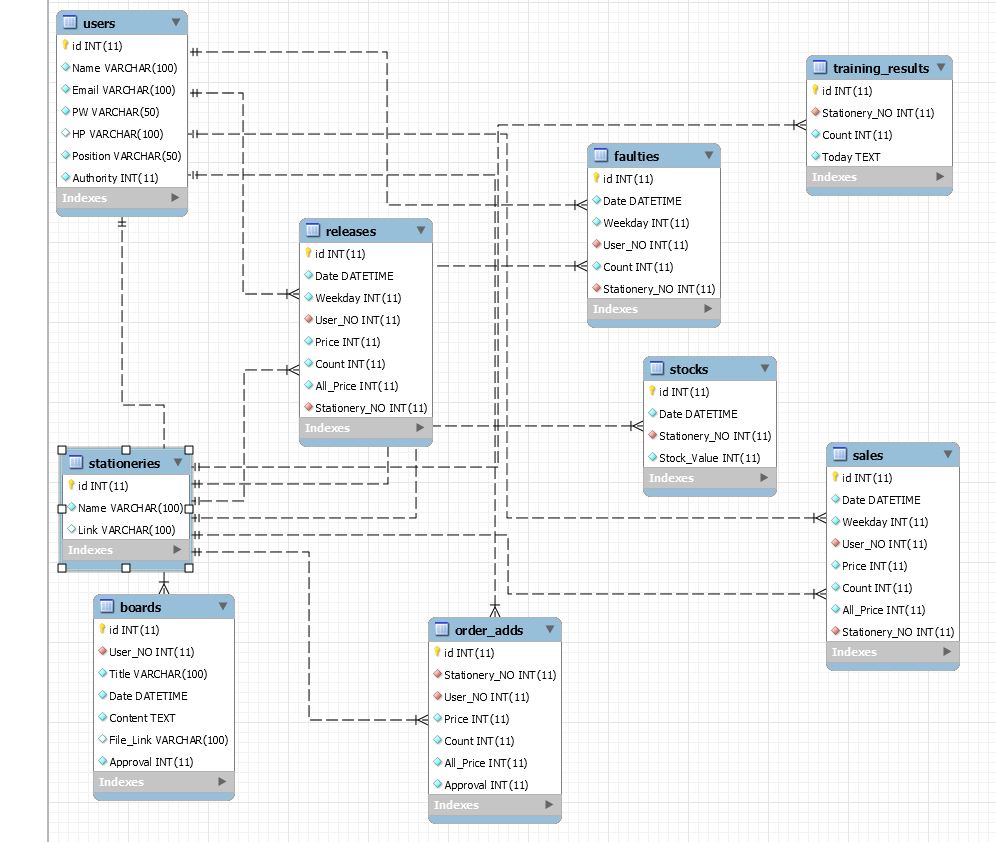
※웹 : 크롬(플랫폼),Visual Studio Code(개발도구), Node.js, Javascript, CSS(부트스트랩), Html

웹의 프론트는 Html과 CSS(부트스트랩) 언어는 Javascript 사용하였으며, 서버는 Node.js Express를 사용하고 언어는 Javascript 사용하였다.

주로 사용하는 웹의 플랫폼은 파이어폭스, 인터넷 익스플로어, 크롬 중 크롬을 선택했으며, 개발 도구는 무료이면서 탁월한 툴인 Visual Studio Code를 사용했다.

※DB : MySql(개발도구)

처음에 구축한 DataBase 모델 초안이 있었지만, 개발을 진행을 하던 도중, 필요한 Table이 생기기 시작하여, 지금의 DataBase처럼 모델링이 완성되었다.



총 9개의 Table으로 입고, 재고, 출고, 출고대기, 사용자, 물품, 학습결과, 게시판, 불량 게시판으로 구성하였다.

관계는 BCNF 정규형으로 분해하여 Join을 하였으며, BCNF 정규형이란 후보키가 아닌 결정자를 제거해주는 것이므로, 즉 일부 데이터가 갱신, 삽입, 삭제가 되어도 이상이 일어나지 않게 분해를 하였다.

※인공지능 : Keras LSTM 모델, Python

사용언어는 Python, 인공지능 툴은 Keras를 썼으며 모델은 LSTM 모델을 썼다.

LSTM은 Long Short Term Memory 의 약자로써 RNN (Recurrent Neural Network)를 한 단계 더 발전한 모델으로 우리는 이 모델으로 만들어진 금융분석 ai를 응용해서 사용했다.

※앱 : 안드로이드(플랫폼), 안드로이드 스튜디오(개발도구), Java

앱은 전적으로 안드로이드 플랫폼을 기준으로 만들었으며, 개발도구는 안드로이드 스튜디오를 썼다.

일반적으로 activity를 이용해서 만드는 것과 달리 이번에는 fragment라는 개념을 activity 대신 사용해보았다.

비콘은 안드로이드의 블루투스 비콘 라이브러리인 altBeacon을 써서 핸드폰의 블루투스 통신 장치를 비콘처럼 작동하게 했으며, 거리 측정은 당 라이브러리에 있는 위치측위 함수를 이용해 만든 앱을 다른 핸드폰에 이식함.

3.2 적용된 주요 기법

※인공지능

AI System은 앞서 언급했던 Keras의 LSTM 모델을 사용하여 구현하였다.

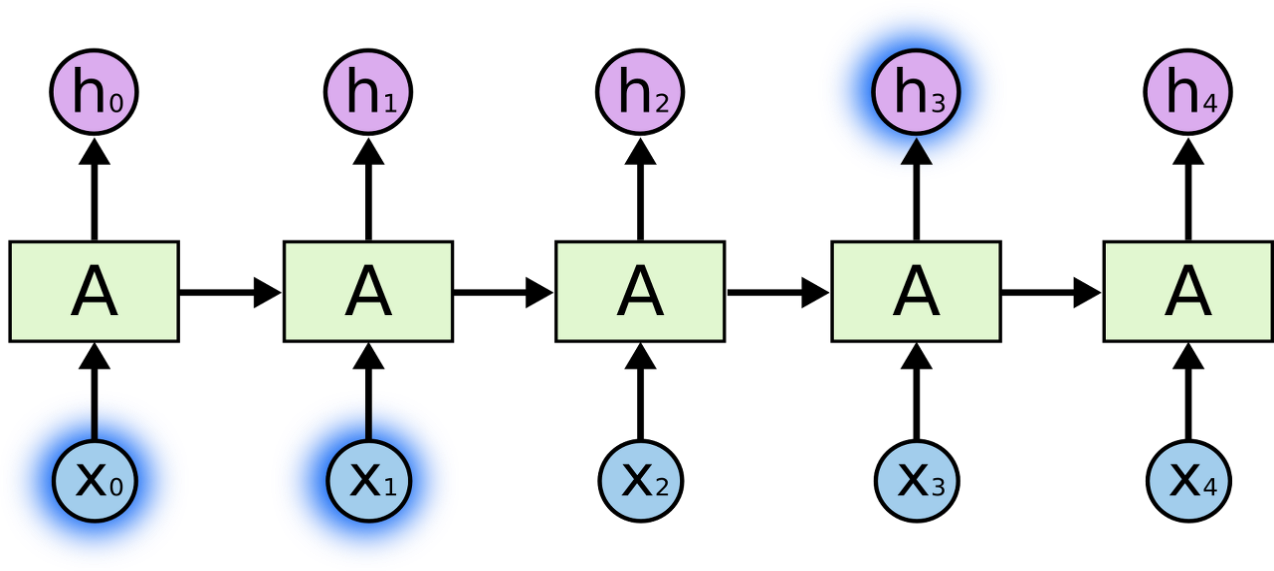
LSTM은 Long Short Term Memory 의 약자로써 RNN (Recurrent Neural Network)를 한 단계 더 발전한 모델이다.

RNN의 개념부터 설명하자면 자기 스스로 반복하면서 이전 단계에서 얻은 정보를 지속되도록 하는 것이다. 예를 들어 우리가 하루에 일어나는 일들을 분류하고 싶다고 해보자.

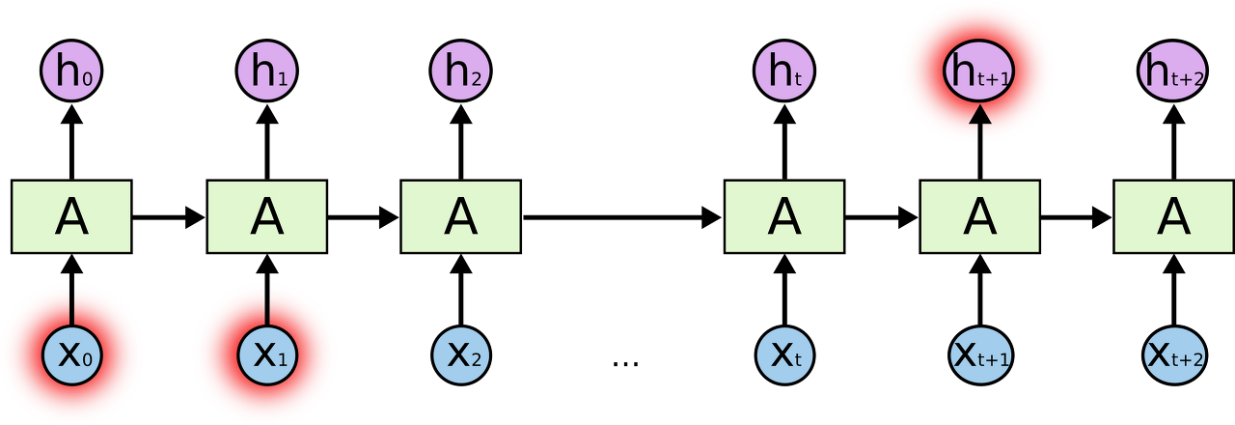
전통적인 Neural Network는 이전에 일어난 사건을 바탕으로 나중에 일어나는 사건을 생각하지 못한다 즉 미래를 생각을 못한다고 볼 수 있다.

이러한 문제점을 해결하기 위해서 RNN이 모델이 개발 되었고, 음성 인식, 언어 모델링, 번역, 이미지 주석 생성 등등 다양한 분야에서 사용되고 있다.

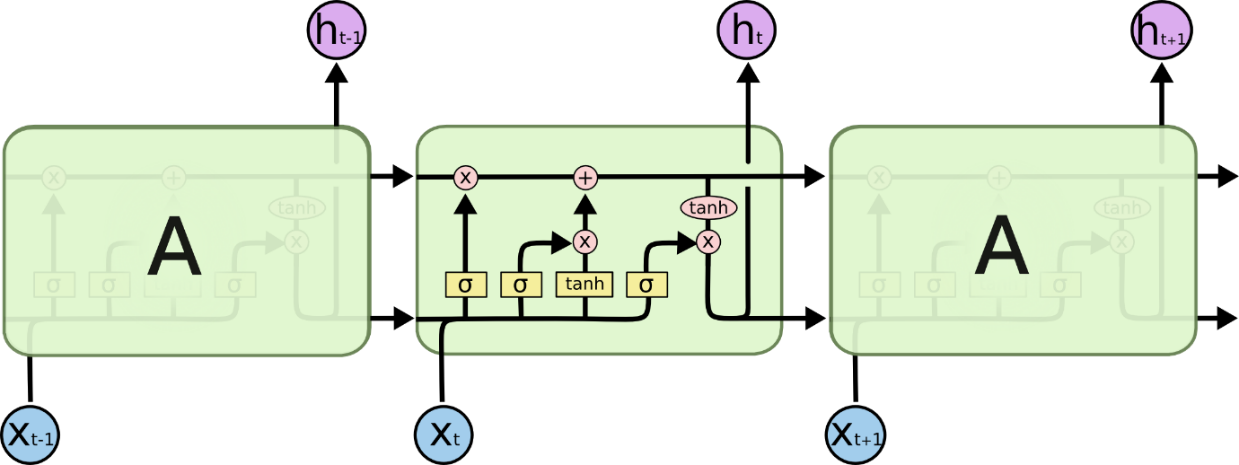
하지만 RNN의 문제점은 긴 의존기간으로 인해 문제점이 생기는데, 긴 의존기간의 문제점이란 기간이 긴 데이터를 예측할 시 시간격차가 굉장히 커지게 된다고 한다.



짧은 시간의 RNN 작동 방식



긴 시간의 RNN의 구현 방식



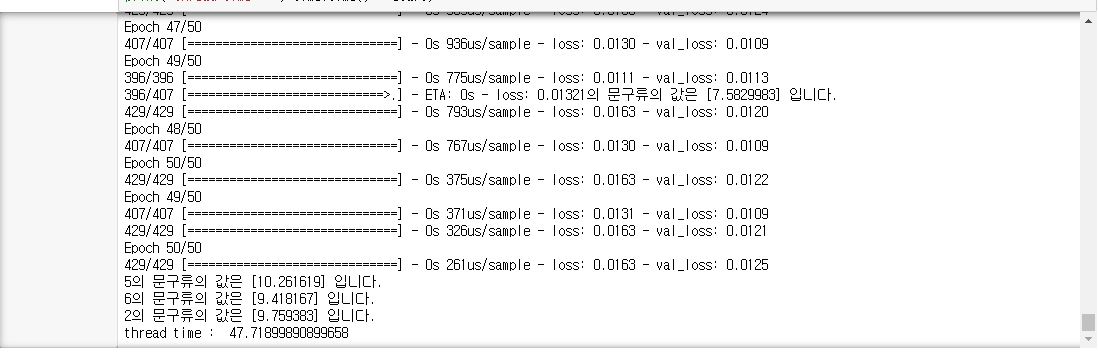
LSTM 구현 방식

그래서 이런 긴 기간의 데이터를 예측할 때 시간 격차를 좁히기 위해서 개발된 모델이 LSTM 모델이라고 한다.

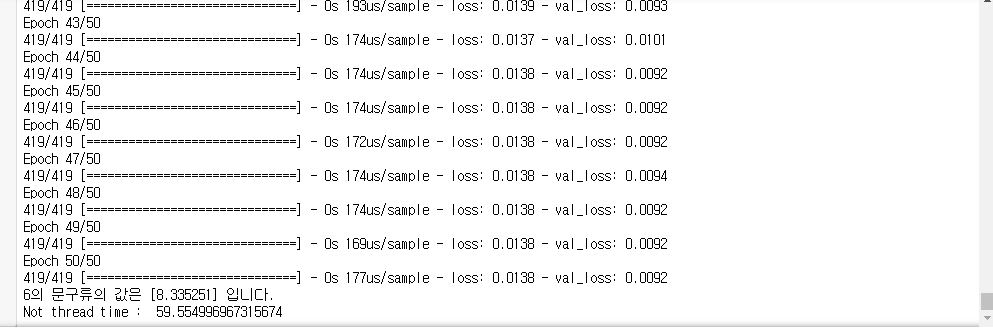
우리는 오래된 데이터까지 학습하여 결과를 추출을 해야 되기 때문에, LSTM 모델을 선택을 하였고, 추출한 데이터를 학습을 시켜보았다.

데이터 학습을 시킨 컴퓨터의 성능은 CPU는 I5-6500, GPU GTX650, RAM 8G의 성능을 가진 PC이다.

학습을 해본 결과 속도는 약 60초 정도 나왔다. RNN보다는 만족할 수는 있겠지만, 한 번 더 생각을 해보기로 했다. ‘물품당 Thread를 생성해서 학습을 시키면 더 빠르지 않을까?’라는 생각을 했고 물품당 Thread를 생성하여 학습을 해본 결과 약 50초 정도의 매우 빠른 속도로 학습이 되었다.



Thread로 학습 시 약 47초 시간이 소요되었다. (요일마다 데이터셋이 다르니 약 50초 생각하면 될 듯하다.)



Thread를 미 사용 학습 시 약 59초 시간이 소요되었다. (요일마다 데이터셋이 다르니 약 60초 생각하면 될 듯하다.)

학습된 데이터의 결과를 DB에 INSERT 하면 다음과 같이 데이터가 INSERT가 된 걸 확인할 수 있다.

※Rssi 위치 측위

위치를 하는 방식에는 총 세가지 방식이 있는데, 첫째로 각도를 활용한 방식(AOA), 둘째 신호의 전송시간을 활용한 방식(TOA), 셋째로 신호의 세기를 측정하여 위치를 추적하는 방식인(RSSI)가 있다.

우리는 이 중에서 RSSI 방식을 채택해서 했으며, altBeacon 라이브러리에 있는 위치 공식인 txpower를 이용해서 했다.

다만 Rssi 특징상 오차가 굉장히 심했으며, 이 때문에 어느 정도 나온 결과값이 부정확한 것을 전제로 하게 되었다.

아두이노 블루투스 비콘이 하나밖에 없어서 핸드폰을 비콘으로 썼으며, 이를 위해서 altBeacon 라이브러리를 이용해 해당 핸드폰들에 있는 블루투스 모듈들이 비콘기능을 지원하는지 전부 검사했다.

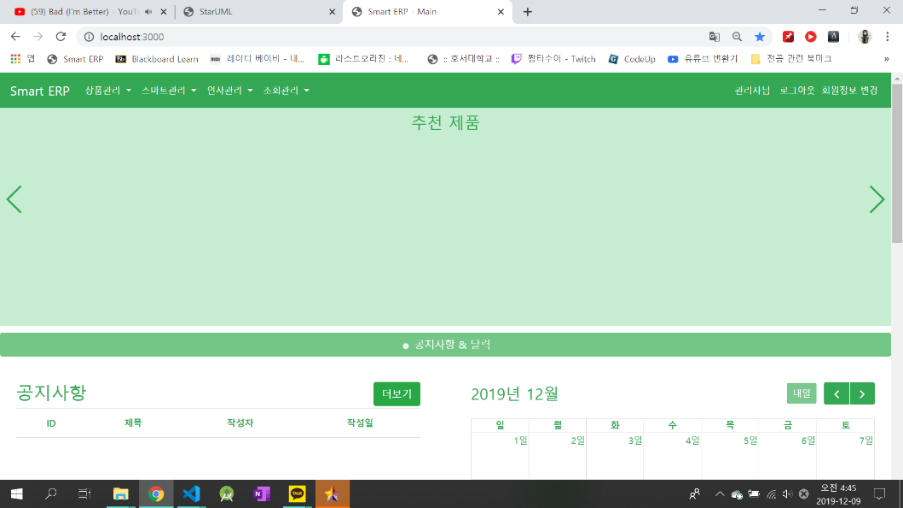
총 3개의 핸드폰이 통과했고 그 핸드폰들엔 altBeacon 라이브러리를 이용해서 주기적으로 비콘 신호를 내보내도록 만드는 앱을 깔아 비콘처럼 동작하게 만들었다.

이후 rssi 값을 측위하는 앱이 깔린 핸드폰으로 각 비콘들과의 rssi 값을 측정했으며 핑거프린팅 방식으로 rssi 표를 작성했다.

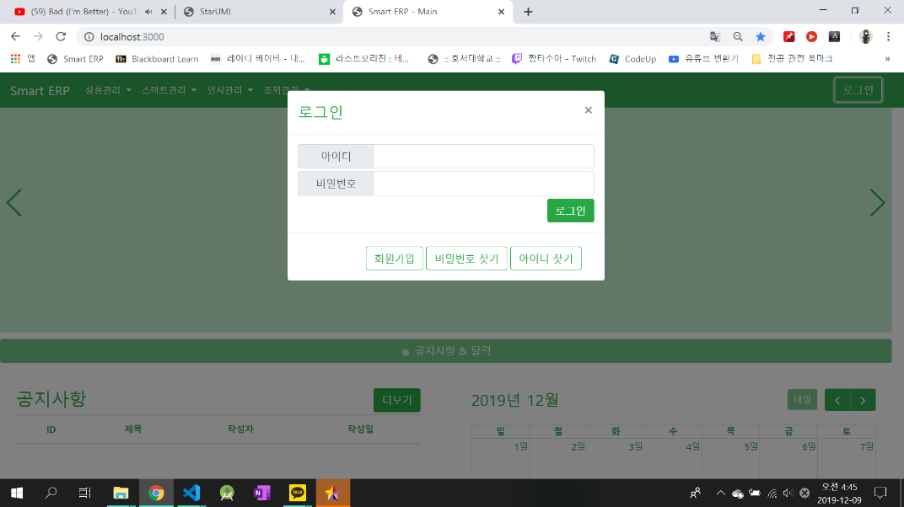
3.3 구현 결과

※웹

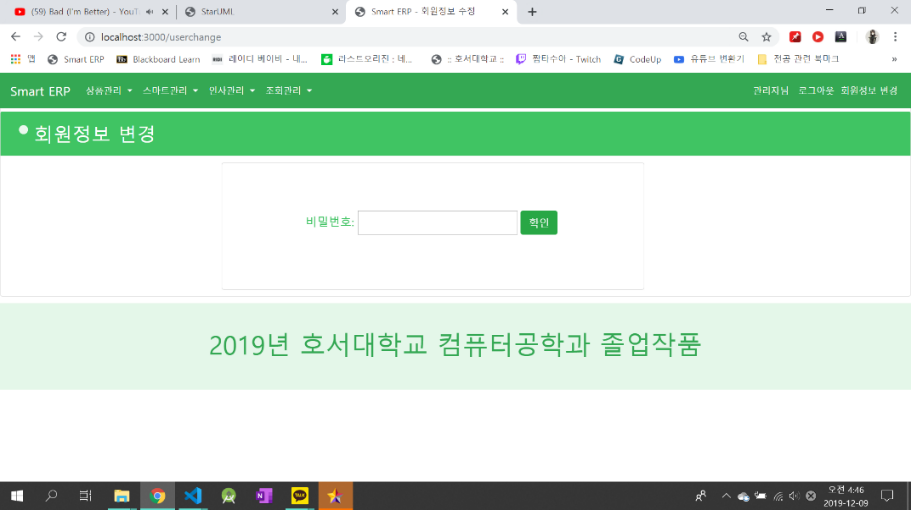
메인 화면

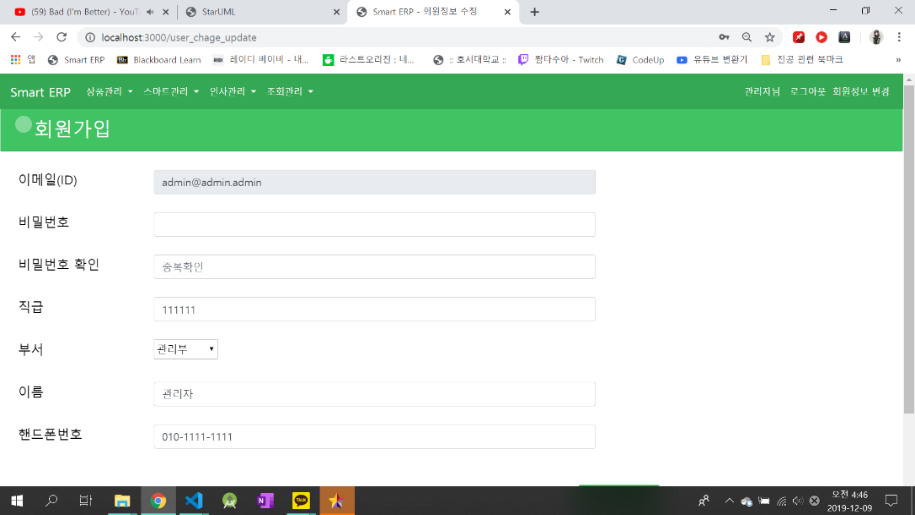


로그인



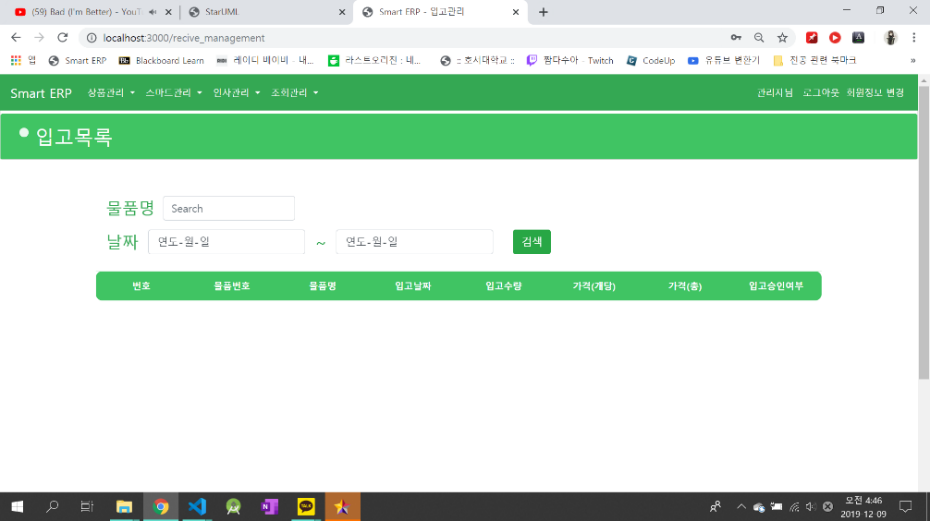
회원정보 변경



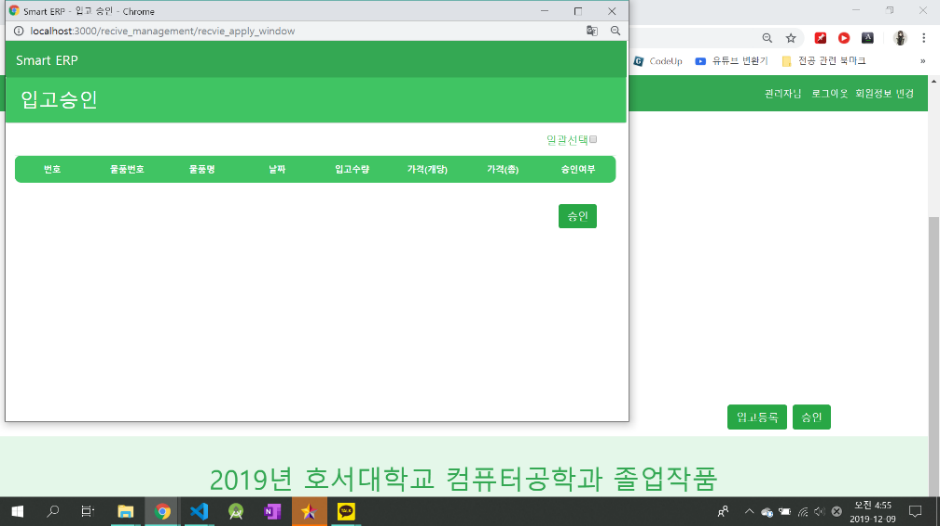


상품관리

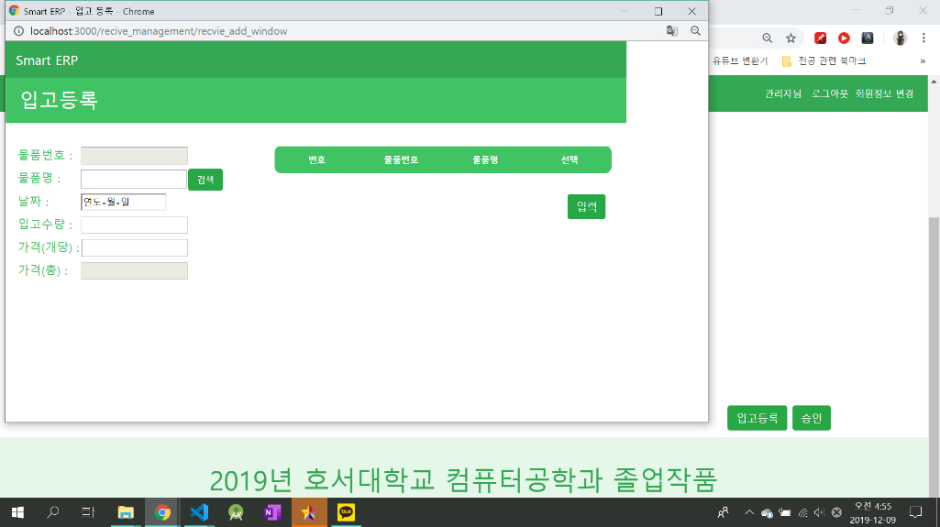
입고목록



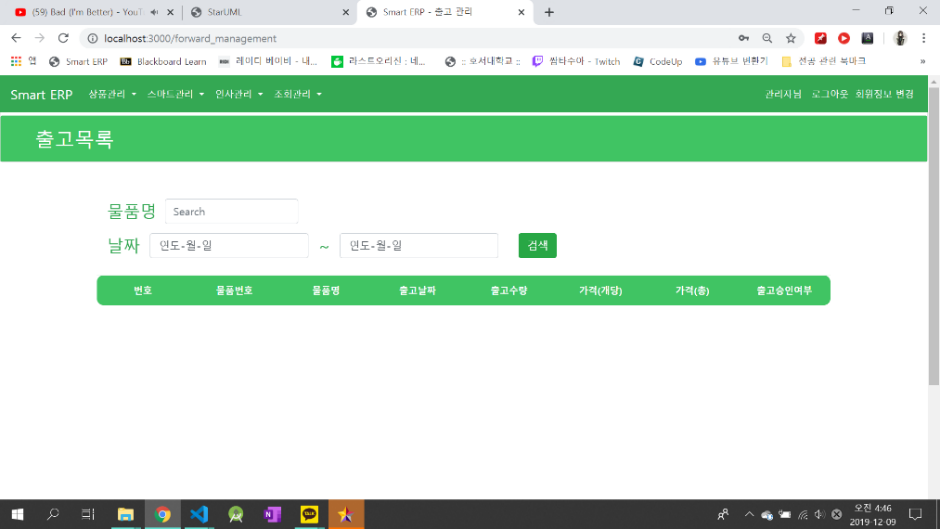
승인



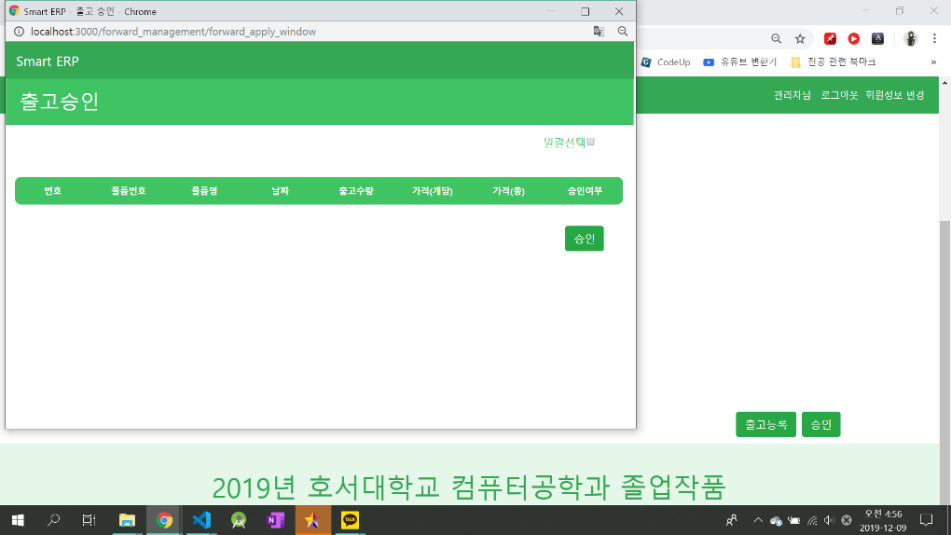
등록



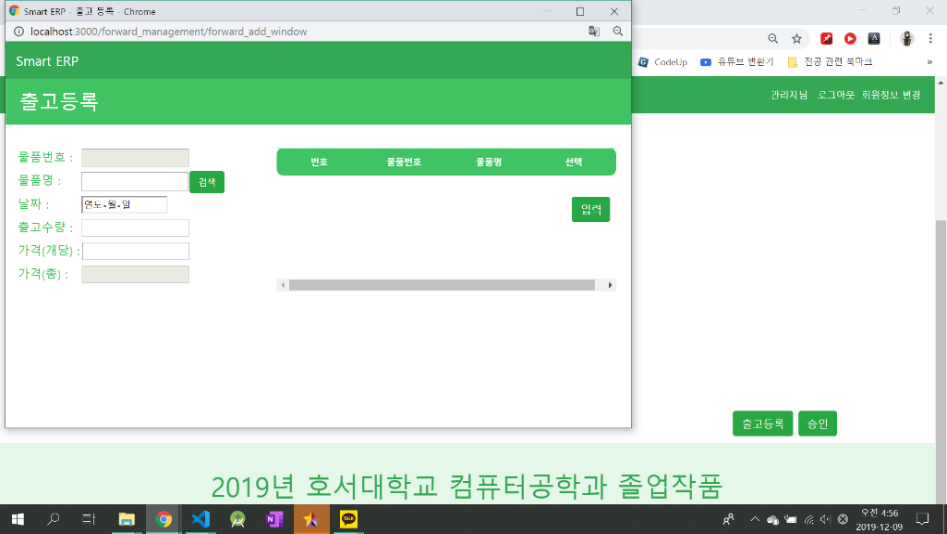
출고목록



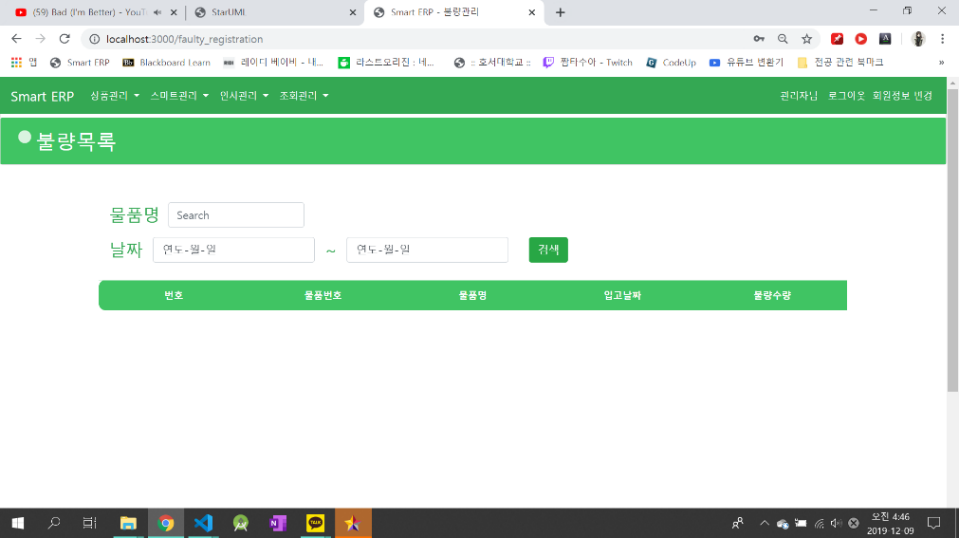
승인



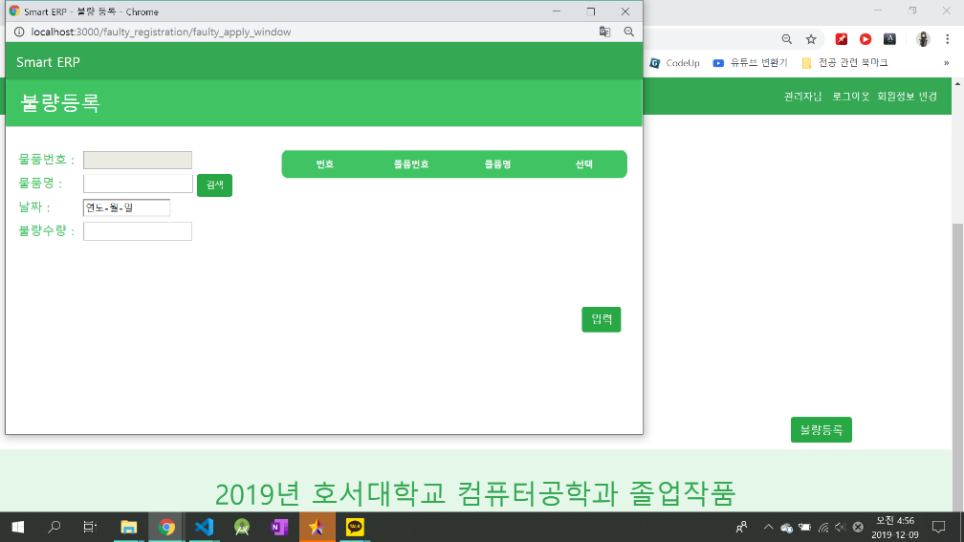
등록



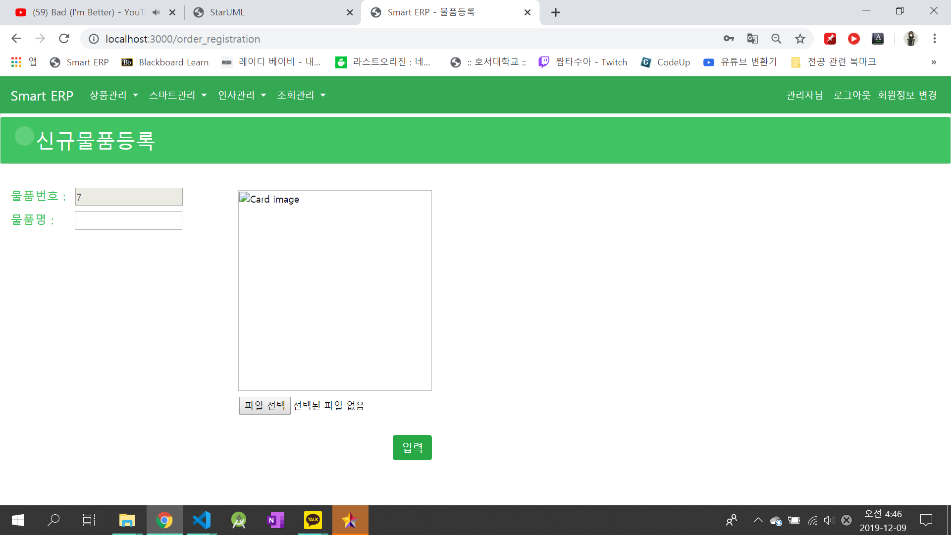
불량목록



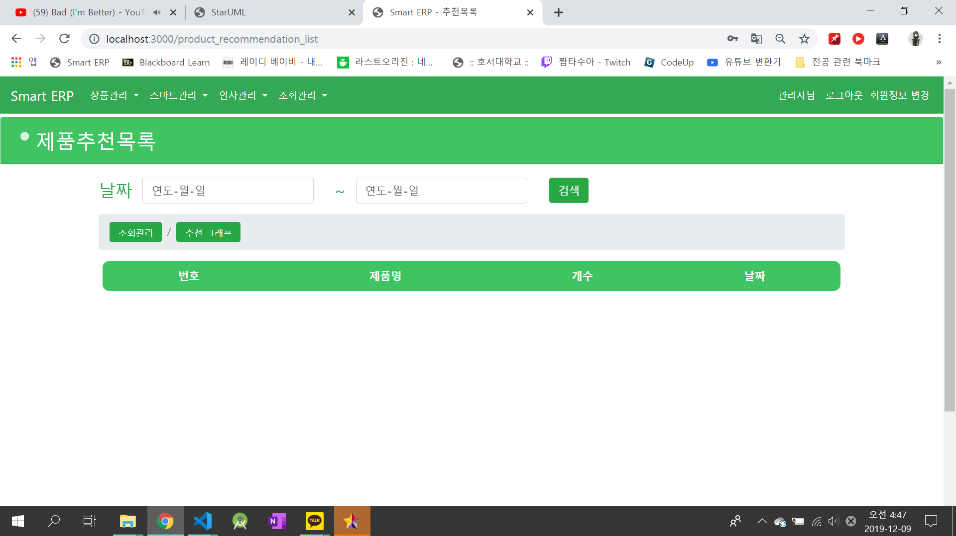
등록



신규물품등록

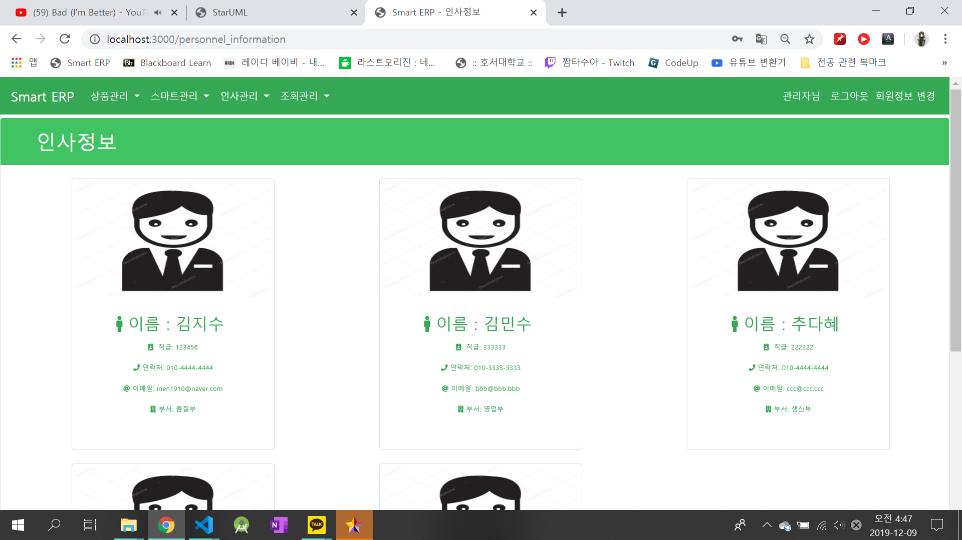


스마트관리-제품추천목록

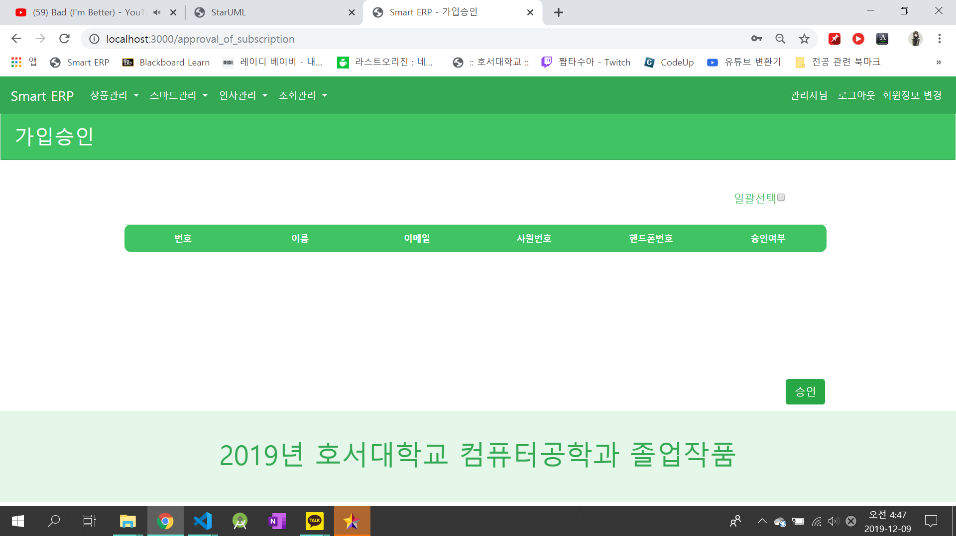


인사관리

인사정보

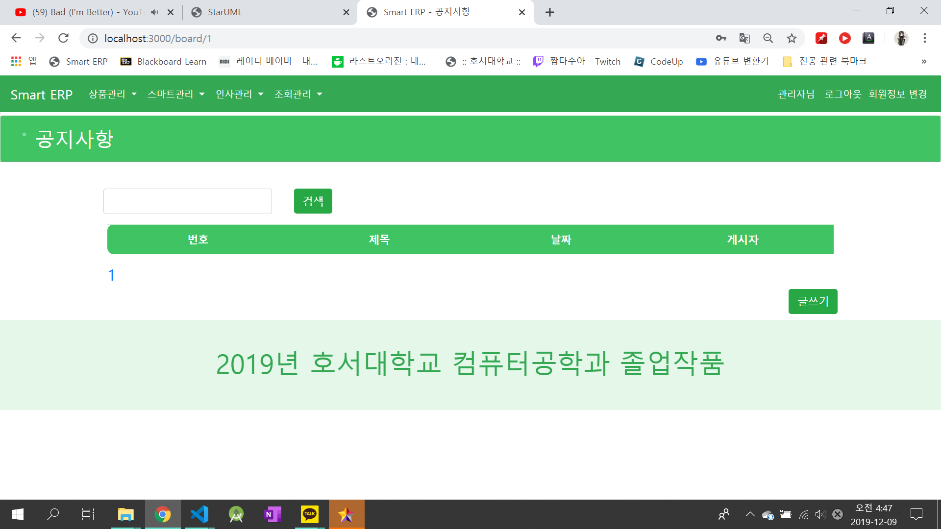


가입승인

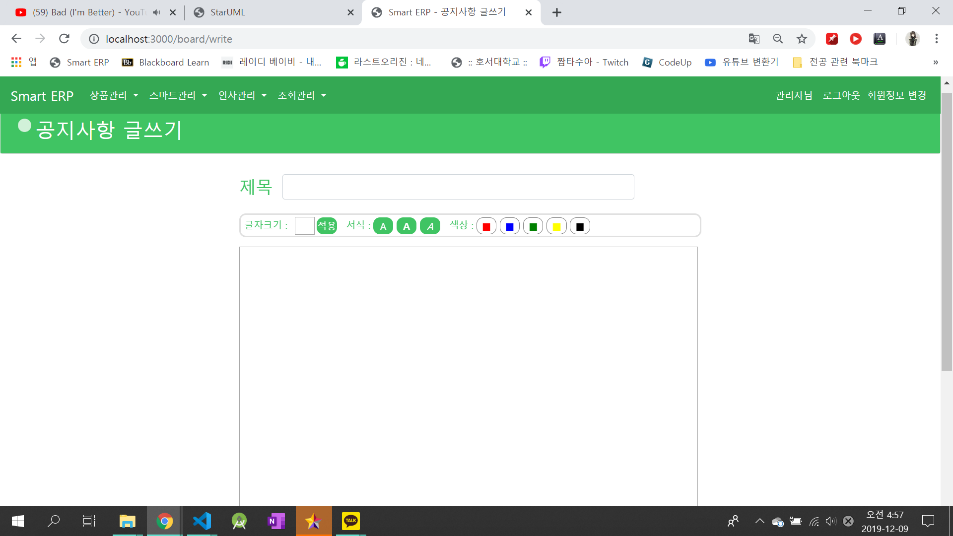


조회관리

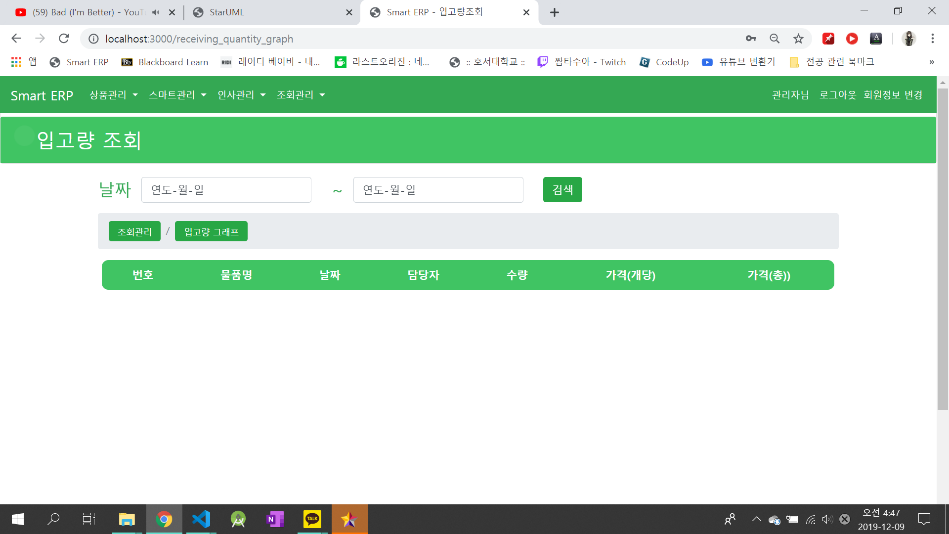
공지사항



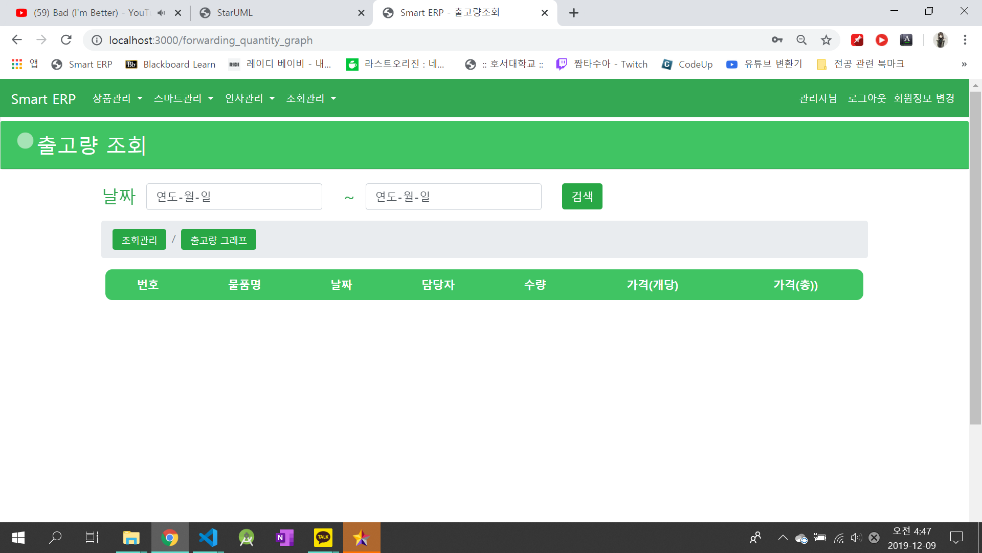
글쓰기



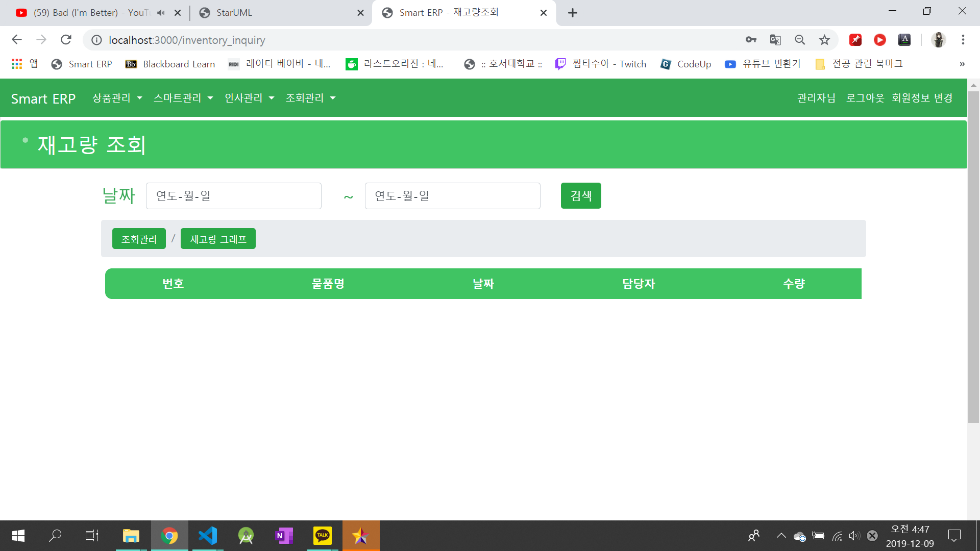
입고량조회



출고량조회



재고량조회



※사용자 앱 기능

제품 추천 목록



출고 확인



입고 확인

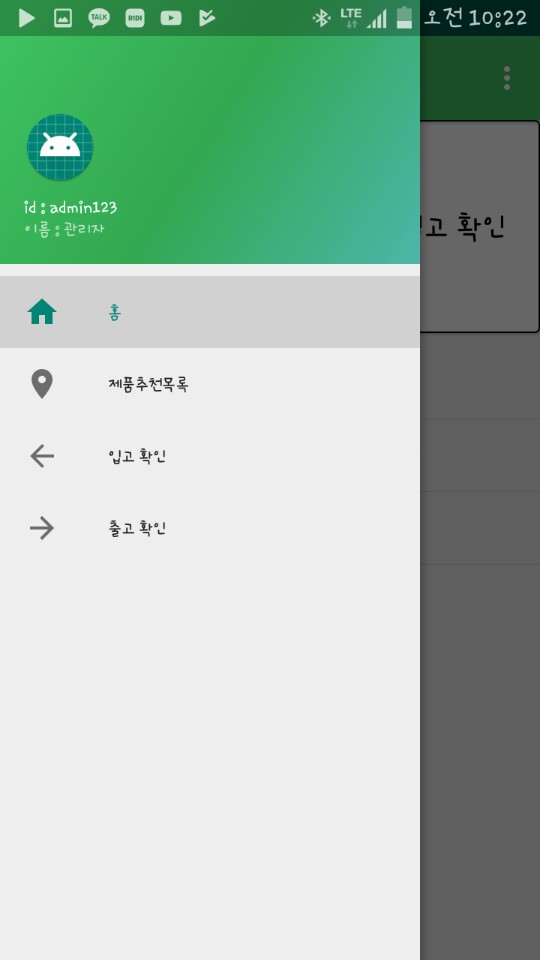


로그인 기능



공지확인 기능





.카카오톡 연동기능



1. 업무별 참여 및 완성도

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 이름  업무내용 | 맹주완  (참여도) | 김지수  (참여도) | 추다혜  (참여도) | 완성도  (시연여부) |
| 사용자 앱 | ⅹ | ○(100%) | ⅹ | 80%(○) |
| 비콘 앱 | ⅹ | ○(100%) | ⅹ | 100%(ⅹ) |
| 웹 백앤드 및 서버 | ○(100%) | ⅹ | ⅹ | 100%(○) |
| 웹 프론트 | ○(40%) | ○(60%) | ⅹ | 100%(○) |
| DB | ○(100%) | ⅹ | ⅹ | 100%(○) |
| 인공지능 서버 | ○(100%) | ⅹ | ⅹ | 100%(○) |
| 앱 서버 | ○(20%) | ⅹ | ⅹ | 100%(ⅹ) |
| 중간보고서 | ⅹ | ○(70%) | ○(30%) | 100%(○) |
| 최종보고서 | ⅹ | ○(100%) | ⅹ | 100%(○) |

1. 배운 것 및 느낀 점

맹주완(팀장)

- 평소에 관심 가지고 있던 AI를 공부만 하고 있었는데, 공부했던 내용들을 직접 프로젝트에 접목을 시켜서 개발을 했다는 거에 보람을 느꼈다.

- Keras LSTM 모델을 사용하여 회귀분석을 하였는데, 정확한 데이터셋만 가지고 학습을 한다면 상용화할 정도로 매우 만족한 결과를 보여줬다.

- Node.js로 웹 개발을 처음 해봤는데, 되게 매력적이라고 생각이 들었다. 지원하는 라이브러리가 많아서 편하게 개발을 할 수 있었던 것이 최고의 장점으로 느껴졌다. 기초지식만 있어도 빨리빨리 개발할 수 있어서 더 매력적으로 다가왔다.

- DataSet이 없어서 구하는 데 힘이 들었지만, 임의적으로 잘 만들어서 DataSet도 생각보다 금방 확보할 수 있었고, 평소에 공부하면서 놓친 부분은 개발을 하면서 다시 리마인드 할 수 있어서 정리가 되었다.

- 소통 부분에서 많은 문제점이 있었음 프로젝트는 체계적으로 나눠서 해야 되는데, 개인주의를 가진 애들이 많아서 그런지 자기밖에 생각을 못함 협업하는데 매우 힘이 들었음

- 졸업 프로젝트 하기 전에 개인사정으로 인해 휴학자는 프로젝트 진행하기 전에 이름을 빼고 시작하는 게 좋을 듯함 아님 시작하였다면 끝까지 책임 지겠다는 각서를 쓰는 것도 좋을 듯

- 졸업 프로젝트가 문제점이 매우 많다고 생각을 한다 그 중 최고 문제점은 어차피 안 해도 졸업은 한다는 마인드를 가진 애들이 많아서 책임의식이 없음 또한 못하는 애들은 잘하는 애들에게 묻어가면 된다고 생각을 해서 업무를 줘도 하지를 않음 그럼 결국 잘하는 애들만 프로젝트를 다 수행하게 되어서 정신적으로 피로도가 심하고, 개인 시간을 보장하지 못함 못하는 애들은 받아먹기만 하고 실력 향상에는 도움이 되지 않고, 졸업프로젝트에 힘을 쓰지 않는 악순환이 생김 졸업프로젝트 개선점이 필요하다고 생각함 (개선방안은 참여 안 하는 팀원은 졸업을 못 시키거나, 다른 방안으로 졸업을 시킬 수 있는 방법을 검토)

김지수

- 하나의 플랫폼에서만 만들다 보니까 다른 플랫폼과 연결해서 만들어야 할 때의 지식이 심각하게 부족하다는 것을 느끼게 되었으며, 또한 웹 프론트에 대한 전반적인 지식을 얻을 수 있는 좋은 기회였다.

- 서버에 대한 지식은 매우 부족했다는 것을 다시 한번 느꼈고 해당 공부는 더 해야함을 다시 한번 통감했다.

- 또 안드로이드 앱을 이번엔 네이티브 앱으로 만들었는데 원래의 의도와 살짝 맞지 않아 다음에 비슷한 프로젝트를 할 시에는 웹앱과 네이티브앱 둘의 장점이 합해진 하이브리드 앱으로 만들 계획이다.

- 조원과의 협업 면에서는 무척이나 힘들었다.

- 아직까지는 분담이나 협업에 익숙하지 않아서 그런 것도 있었지만 무엇보다 제일 답답했던 것은 말로 했던 것을 직접 실천할 수 있던 지식이 없던 나 자신이었다.

- 여러모로 부족함을 많이 느낄 수 있던 프로젝트였으며 이 경험은 두고두고 교훈으로 삼을 것이다.

추다혜

- 프로그래밍을 하는데 처음 접하는 부분이여서 힘든점이 많았다. 나의 실력도 많이 부족했었던 것은 맞다. nodejs를 이용하여 html디자인을 하였는데 ejs파일을 만들어서 창과 버튼, 날짜 등을 설정할 수 있고, 각 버튼들을 누르면 해당 창으로 이동하게끔 하고, 메일을 전송할 수 있는 창을 만들어보았다. 처음에는 자신이 정말 없던 부분들이었지만 옆에서 친구가 할수있다고 응원해주고 잘 알려준 덕분에 조금이라도 더 힘을 내서 도전을 할 수 있었다. 그래서인지 포기를 하지 않으려고 노력을 했던 것 같다.