**Remote Enclosure**

**Management System**

**프로젝트 최종보고서**

2012154010 김수혁

2013152008 김병욱

2013152015 김진하

2014156048 한준희

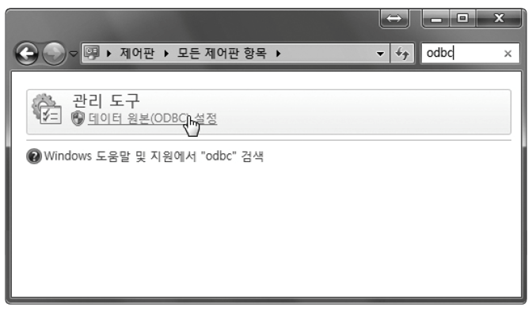
1. 프로젝트 개요

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **프로젝트 정보** | | | | | | | | |
| **프로젝트명** | Remote Enclosure Management System | | | | | | | |
| **주제영역** | □ 건강 □ 생산성 □ 생활  안전 □ 엔터테인먼트 | | | | | | | |
| **기술분야** | □ 이동통신  네트워크 | | □ 스마트미디어  □ 전파·위성 | | | SW·컴퓨팅  □ SW | □ 디지털콘텐츠  □ 정보보호 | □ 융합서비스  □ ICT 디바이스 |
| **달성성과** | □ 논문게재 및 포스터발표 □ 앱등록 □ 프로그램등록 □ 특허 □ 기술이전  □ 실용화 □ 공모전( )  기타( 교내 팀 프로젝트 ) | | | | | | | |
| **프로젝트명** | Remote Enclusure Management System | | | | | | | |
| **프로젝트**  **소개** | 최근 증가하고 있는 강도나 절도 범죄 및 상해 같은 범죄로 인해 CCTV 장비에 대한 중요성이 증대되고 있다. 하지만 현재 시중에 나와 있는, CCTV를 관리해주는 함체 관리 시스템은 대부분 효율적이지 못하고, 가격이 비싼 경우가 많다. 하지만 Remote Enclosure Management System은 가격이 비싸지 않고, 효율적으로 함체에 연결된 장비들에 대해 원격으로 관리할 수 있는 기능을 지원한다. 그리고 V-World Map View를 통해서 장비가 연결된 장소에 대한 정보를 확인할 수 있다. 그리고 기존 장비들은 점검을 요청하기가 매우 어려운 면이 있었는데, 효율적으로 장비의 점검을 요청하고, 요청된 장비들에 대한 내역을 확인해서 처리할 수 있게 하였다. | | | | | | | |
| **개발배경 및 필요성** | - 각 지역에 등록된 장비들에 대한 관리가 제대로 이루어지지 않는 경우에 절도나 상해 같은 문제가  일어날 수 있음  - 이로 인해 많은 사람들이 갈등을 빚게 될 수 있음  - 기존의 함체 관리 시스템은 장비들을 원활하게 관리해주지 않음 | | | | | | | |
| **프로젝트**  **주요기능** | 함체 등록 | | | 사용할 함체의 리스트를 출력하고 새로운 함체를 등록하는 다이얼로그 | | | | |
| 함체 장비 구성 | | | 함체에 어떤 장비들이 사용되고 있는지에 대한 정보를 출력하며, 장비의 수리 내역을 출력하는 다이얼로그 | | | | |
| 점검요청 | | | 수리가 필요한 장비의 점검을 요청하기 위해 장비 내역을 출력해주고, 점검을 요청하는 다이얼로그 | | | | |
| 점검요청 내역관리 | | | 점검요청 다이얼로그에서 요청한 점검요청에 대한 출력과 점검 완료 처리를 위한 다이얼로그 | | | | |
| **작품의 기대효과**  **및 용분야** | - **절도 및 상해 예방 :** 장비의 고장으로 인한 절도 및 상해 예방  - **효율적인 관리 :** Remote Control을 통한 효율적인 관리  - **점검 요청 / 확인 :** 복잡한 절차 없이 간편하게 장비 점검을 요청하고 확인  - **Map View :** 장비가 연결된 장소를 V-World Map을 통해서 확인  - **장비 정보 관리 :** 장비에 대한 정보를 효율적으로 관리 | | | | | | | |
| **프로젝트 수행방법** | **번호** | **성명** | | | **담당업무** | | | |
| 1 | 김수혁 | | | 기획서, 최종보고서 작성, 함체 정보 구성 다이얼로그 코드 작성 | | | |
| 2 | 김병욱 | | | MDI 상위 탭 기능 추가, 함체등록 다이얼로그 코드 작성 | | | |
| 3 | 김진하 | | | 장비점검요청내역 탭 다이얼로그 코드 작성 | | | |
| 4 | 한준희 | | | 장비유지관리 다이얼로그 코드 작성, 다이얼로그 통합 | | | |

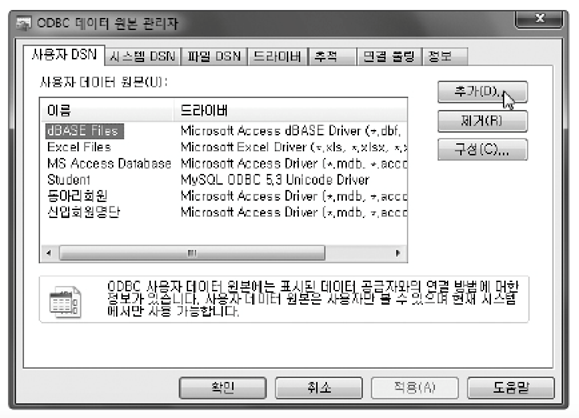
1. 프로젝트 수행 일정

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **구분** | **추진내용** | **수행일정** | | |
| **10월** | **11월** | **12월** |
| **계획** | 수행 계획서 작성 |  |  |  |
| **분석** | HW, SW 구성 구체화 |  |  |  |
| **설계** | SW 설계 |  |  |  |
| HW 설계 |  |  |  |
| **개발** | MDI를 이용한 최상위탭 개발 |  |  |  |
| 팀원 간 개발 탭 선정 |  |  |  |
| 작성된 탭 팀원간 검토 |  |  |  |
| 완성된 탭 통합 |  |  |  |
| 최종 기능 보정 |  |  |  |
| 카메라 + 모터 + 초음파 센서 연결 |  |  |  |
| **테스트** | 제품 테스트 |  |  |  |
| **종료** | 프로젝트 종료 |  |  |  |

1. 프로젝트 실행방법
2. 제어판에 있는 데이터 원본(ODBC) 설정 이용을 선택한다.



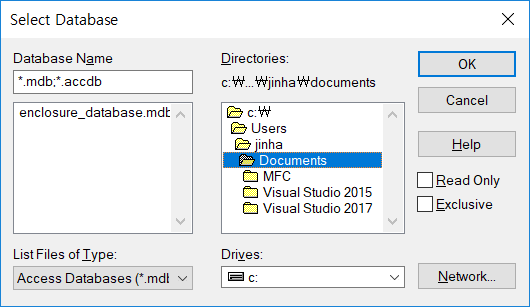
1. [ 추가 ] 버튼을 눌러서 새로운 사용자 데이터 원본을 추가한다.



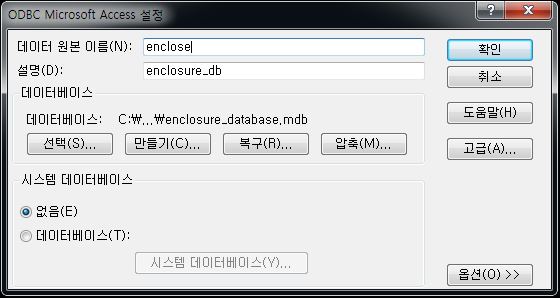
1. 데이터 원본의 드라이버를 ‘Microsoft Access Driver’로 선택한다.



1. [ 선택 ] 을 눌러서 다음과 같이 데이터베이스 파일을 선택해준다. 이 때, 반드시 파일의 경로는 C:\Users\(컴퓨터마다 다름)\Documents\enclosure\_database.mdb여야 한다.



1. ‘데이터 원본 이름’에 ‘enclosure’라고 설정해주고, [ 확인 ] 을 누른다.



1. 데이터 구성
2. 테이블 구성

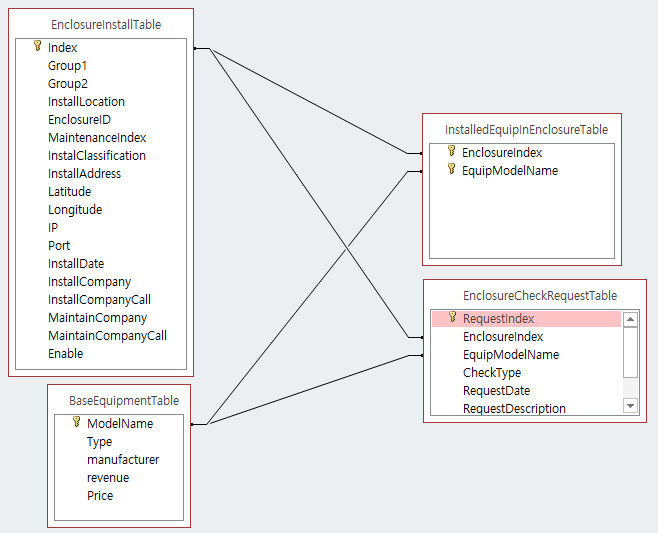


그림 1 : MS ACCESS 관계

우선 뷰를 구현하기 전에 데이터를 먼저 구성하였습니다. “설치된 함체”에 대한 테이블, 설치할 수 있는 “기본장비 정보”에 대한 테이블, “함체에 설치된 기본장비”에 대한 테이블, “함체에 설치된 기본장비의 점검 요청”에 대한 테이블 총 4가지의 테이블을 만들었습니다.

“함체에 설치된 기본장비”에 대한 테이블과 “함체에 설치된 기본장비의 점검 요청”는 외래키로 각각 테이블의 기본키인 함체 인덱스, 기본장비의 모델명인 외래키를 저장하여 구성됩니다.

1. 화면 구성



**그림 2 : REMS 로고**

REMS 프로그램을 실행했을 때 화면에 나타나는 로고입니다.

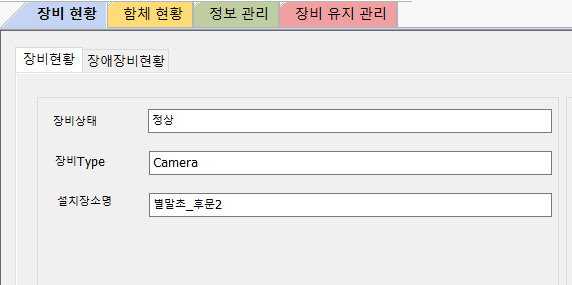


그림 3 : Tab in Tab

각각의 뷰들을 쉽게 보여주기 위해서 전체 기능의 측면에서의 탭과 해당 탭 안에서 서브 탭들을 구성하여 보기 쉽게 탭들을 구성하였습니다. 구현상으로는 여러 개의 MDI 템플릿을 넣어서 기존의 뷰 방식을 바꾸어 탭 방식으로 보여줍니다.

1. 뷰 구성

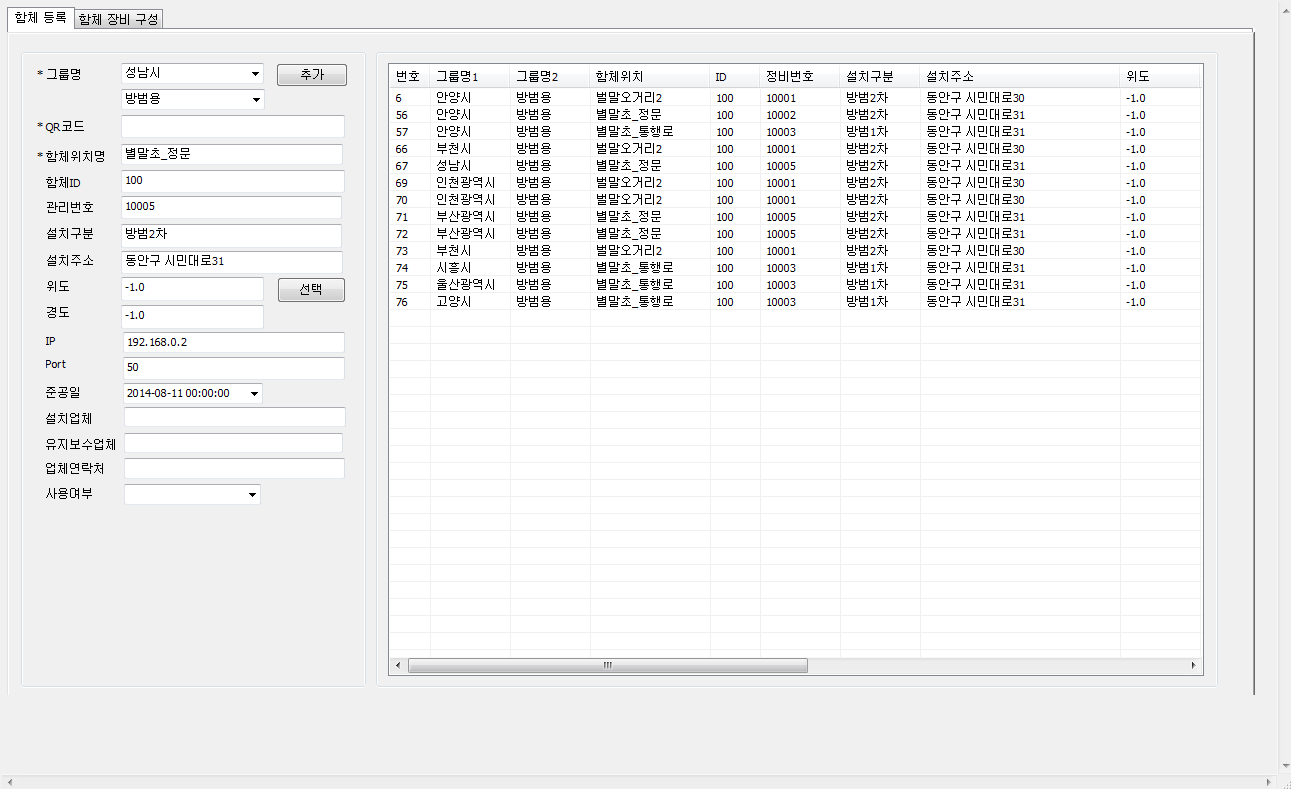


그림 4 : 설치된 함체 장비 관리 뷰

각각의 설치된 함체를 관리하는 뷰로 데이터를 추가하고 보여줄 수 있는 기능이 있습니다. 함체 데이터를 보여주고, 각각의 함체 데이터를 추가할 수도 있습니다.

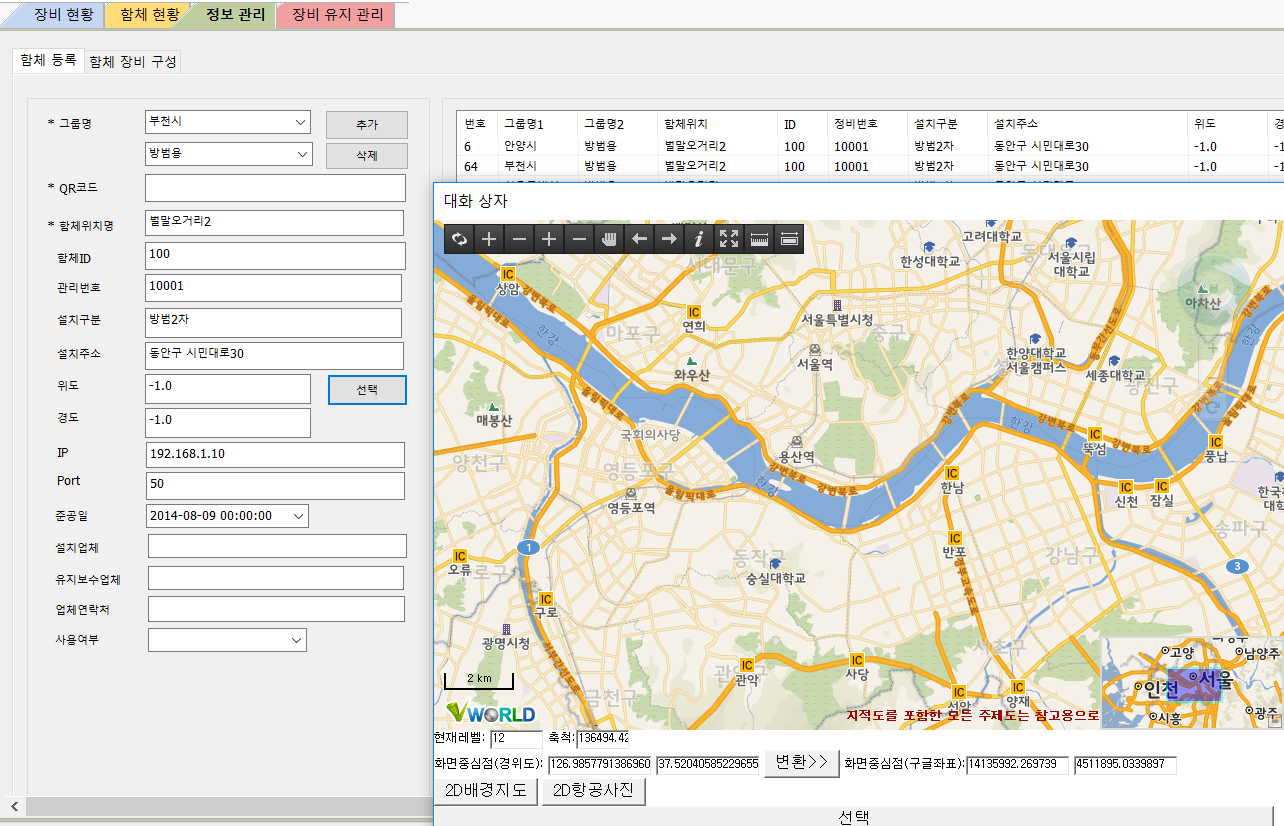


그림 5 : 함체 위치 표시용 지도

“설치된 함체 관리 뷰” 에서 위도 에딧박스 옆의 선택 버튼을 클릭하면 각각의 설치된 함체의 위치를 보여주기 위해 지도를 보여주는 기능을 넣었습니다. VWorld API를 사용하여서 위도와 경도를 받아와서 해당 위치를 2D 배경 지도와 2D 항공 사진으로 볼 수 있게 구현하였습니다.

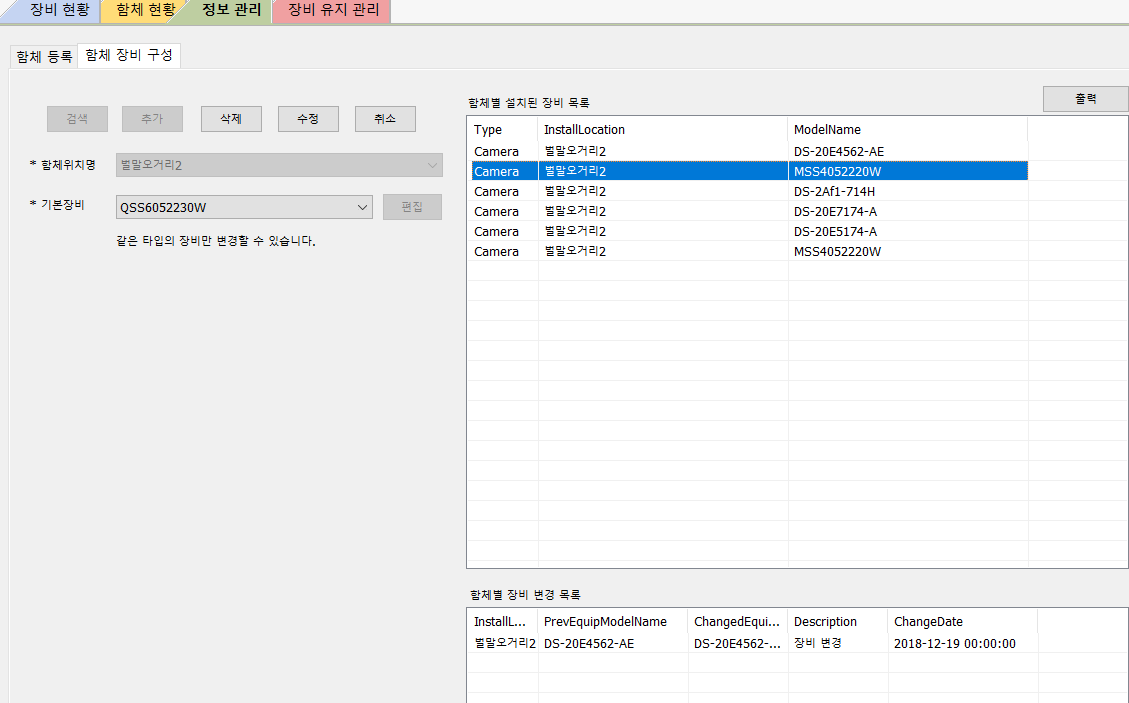


그림 6 : 함체에 설치된 각각의 장비 관리 뷰

이 화면은 “함체에 설치된 장비” 들을 보여주는 화면입니다. 함체 위치와 설치된 모델명을 보여줌으로써 구별할 수 있습니다. 옆의 입력하는 부분은 함체 위치명으로 설치된 함체들을 선택하고 기본장비 모델명으로 기본장비를 구별하여 해당 내용과 같은 것을 검색하거나 테이블에 추가를 할 수 있고, 선택하여 내용을 수정할 수도 있습니다. 아래 리스트에서는 해당 함체에서 장비가 무엇이 바뀌었는지에 대한 내역이 나옵니다. 이는 선택한 것들에 따라서 달라집니다.

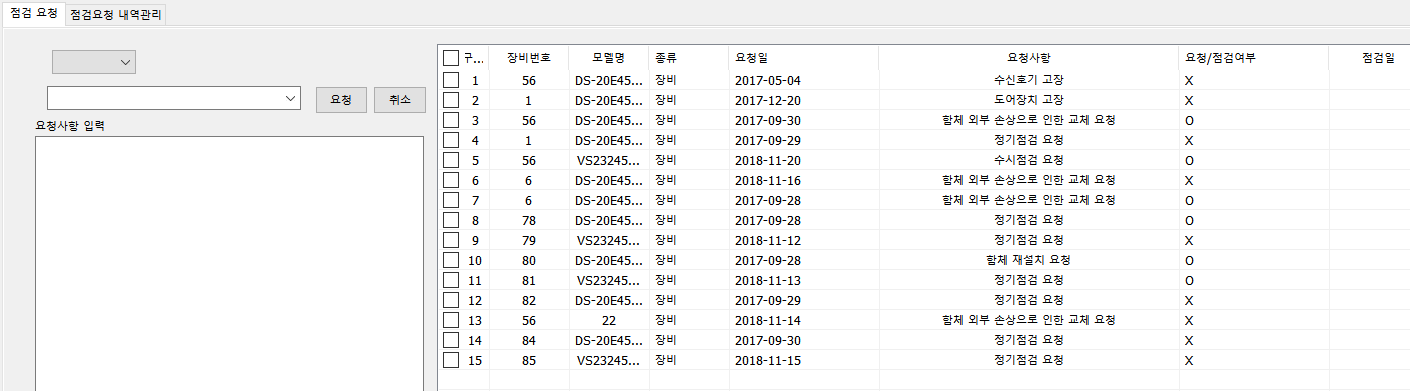


그림 7 : 점검 요청 뷰

이는 전체 “함체-장비” 와 같이 설치된 장비들을 보여줌으로써 한번에 점검된 여부에 따라서 보여줍니다. 이 뷰에서는 함체-장비에 따라서 점검을 요청할 수 있습니다. 또한 현재 장비의 점검이 요청된 것들을 한눈에 파악할 수 있습니다. 기능적인 부분으로는 설치된 장비를 리스트에서 하나의 리스트를 체크박스를 통해 선택하면 왼쪽 콤보박스에 선택 목록이 추가가 됩니다. 콤보박스에서 선택된 아이템을 눌러주고 요청사항이 있다면 아래의 요청사항 입력란에 기재 후 요청 버튼을 누르면 요청/점검여부의 X 표시가 O로 바뀌며 점검요청 내역관리에서 다룰 수 있는 데이터로 전환 됩니다. 데이터의 입장에서는 함체-장비에 따라서 릴레이션을 생성하고, 점검 요청 테이블에 Insert를 합니다.

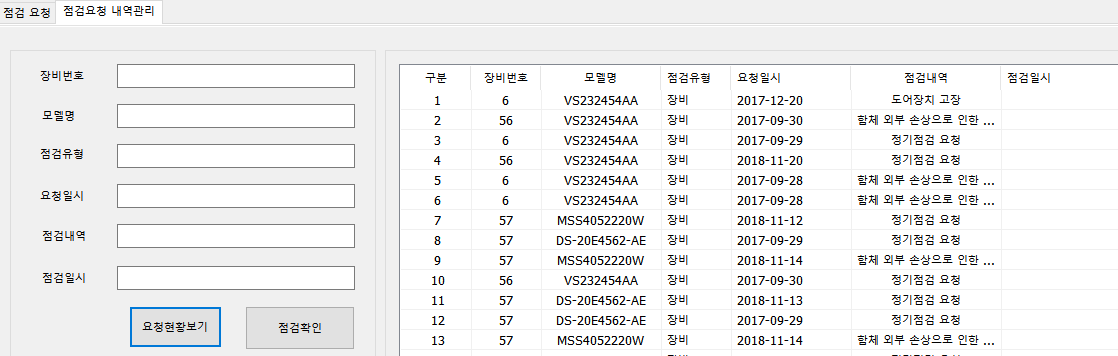


그림 8 : 점검 요청 내역 관리 뷰

여기서는 요청되었지만 점검이 실제로 되지 않는 요청 내역들을 확인할 수 있습니다. 실제 점검이 끝난 후에는 아이템을 선택하여 점검 확인을 하여 요청된 내역에 대해서 점검한 내역으로 체크할 수 있습니다. DB 의 입장에서는 하나의 레코드의 애트리뷰트를 바꿔 주기만 합니다.

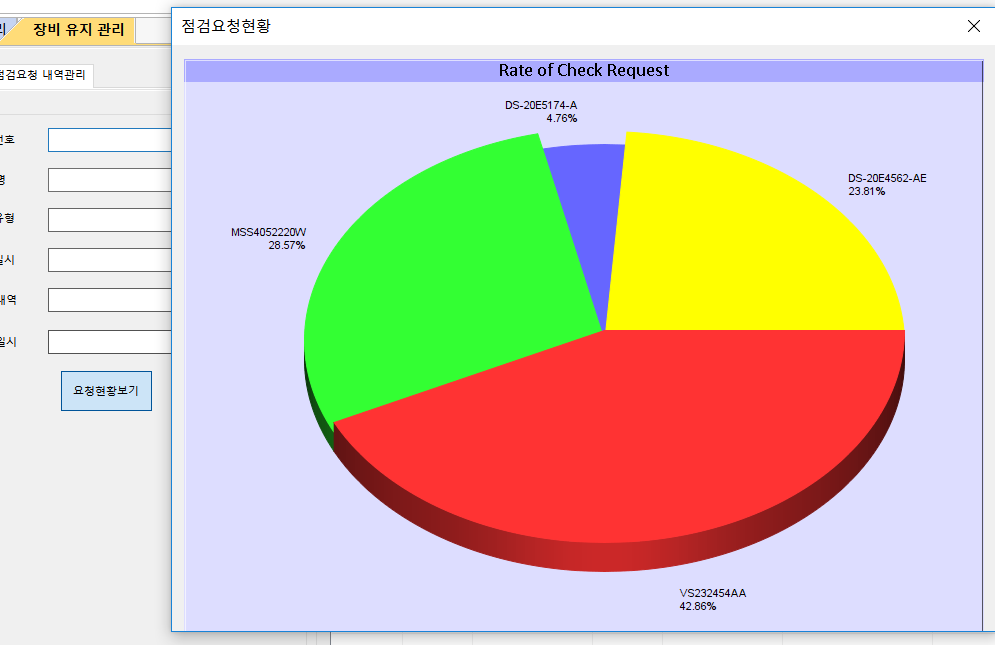


그림 9 : 점검 요청 현황 차트 뷰

점검이 요청된 장비들의 현황을 한눈에 볼 수 있도록 하기 위해 만들어졌습니다. 본 데이터는 차트 형태로 나타나며, 각 장비마다 어느 장비의 종류가 요청비율이 높은지 쉽게 파악할 수 있습니다. 이는 함체에 등록되는 장비들에 대해 보완사항이 있을 시 파악하기에 좋은 자료입니다.

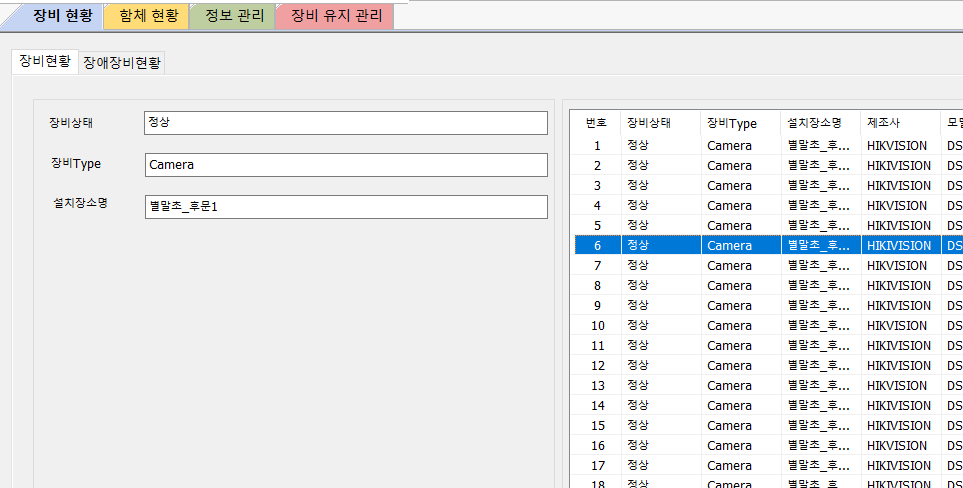


그림 10 :장비 현황 뷰

현재 함체가 연결된 장비들을 볼 수 있는 뷰입니다.

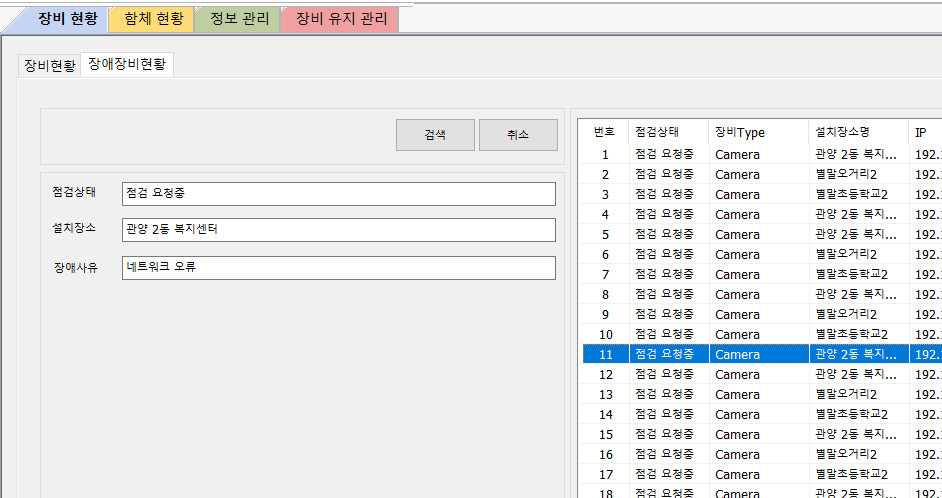


그림 11 : 장애 장비 현황 뷰

현재 함체가 연결된 장비들 중에서 장애가 있는 장비의 점검 상태에 대해서 나타내는 뷰입니다.

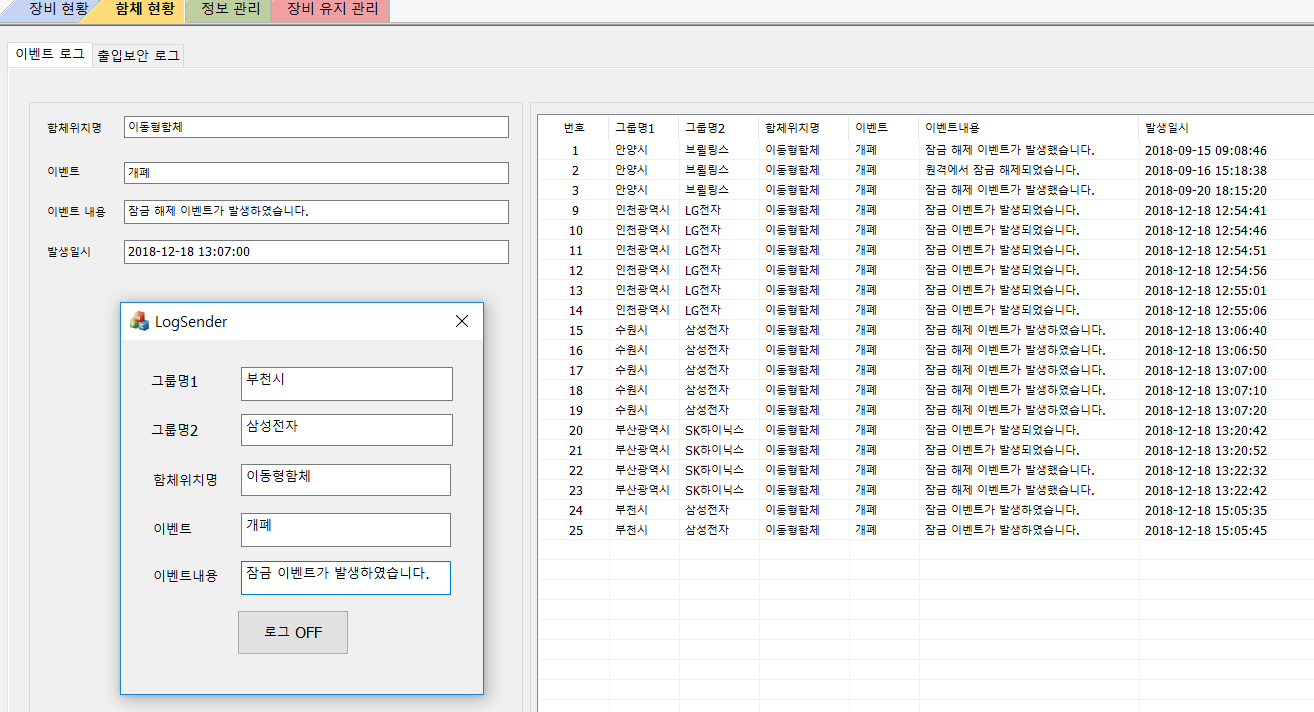


그림 12 : 이벤트 로그 뷰

장비에서 발생한 이벤트(개폐 등)에 대한 로그를 출력하는 뷰입니다. 저희 조는 UDP 소켓 프로그래밍을 이용한 MFC 프로젝트를 하나 더 만들어서 외부에서 이 프로젝트로 로그를 전달하면 이 프로젝트에서 받은 데이터를 데이터베이스 로그에 등록하고, 업데이트 결과를 즉시 보여주는 식으로 구현하였습니다.

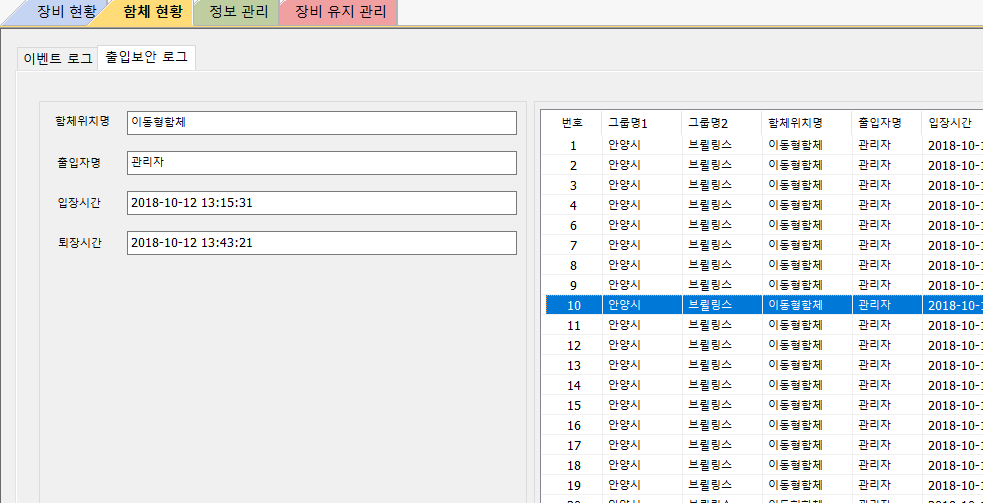


그림 13 : 출입보안 로그 뷰

함체에 출입한 로그를 보여주는 뷰입니다. 출입자와 입장 시간과 퇴장 시간에 대해서 표시해줍니다.

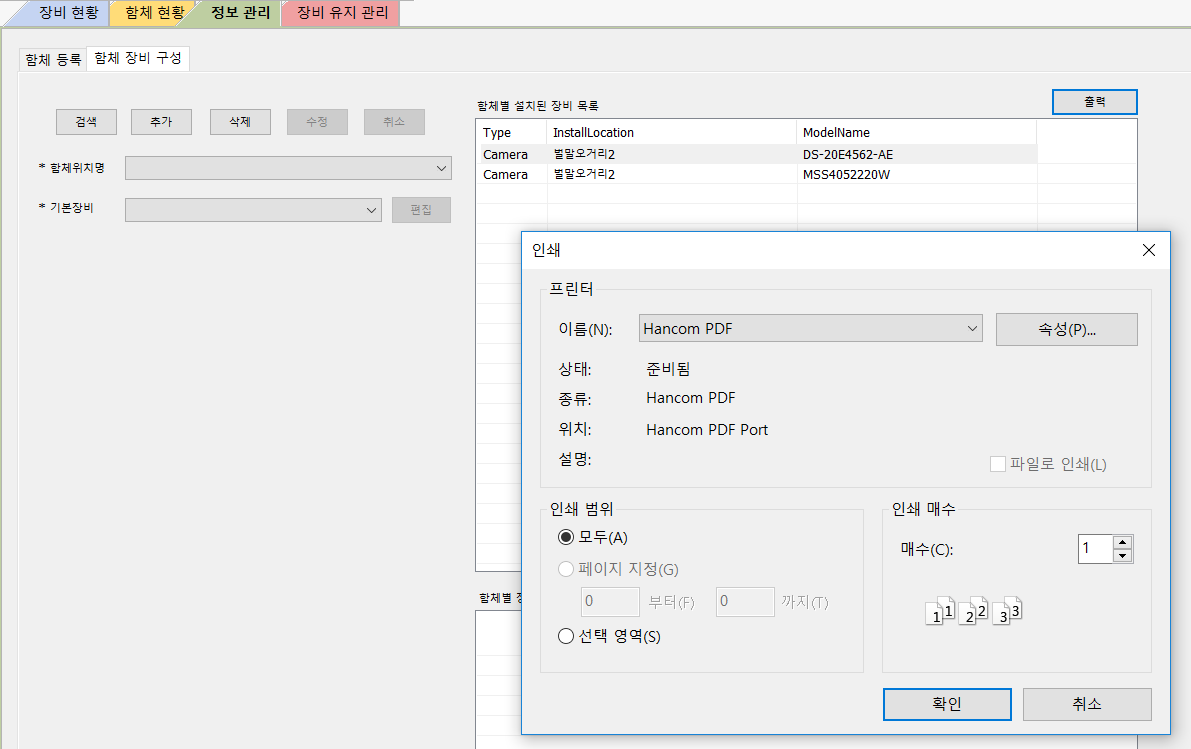


그림 14: 보고서 출력 뷰

함체 장비 구성 탭에서는 해당 함체에 등록된 장비들에 관한 보고서를 출력하는 기능을 구현하였습니다. 함체에 등록된 장비들을 선택하고 출력 버튼을 누르면 보고서를 출력할 수 있는 인쇄 페이지가 나타납니다.

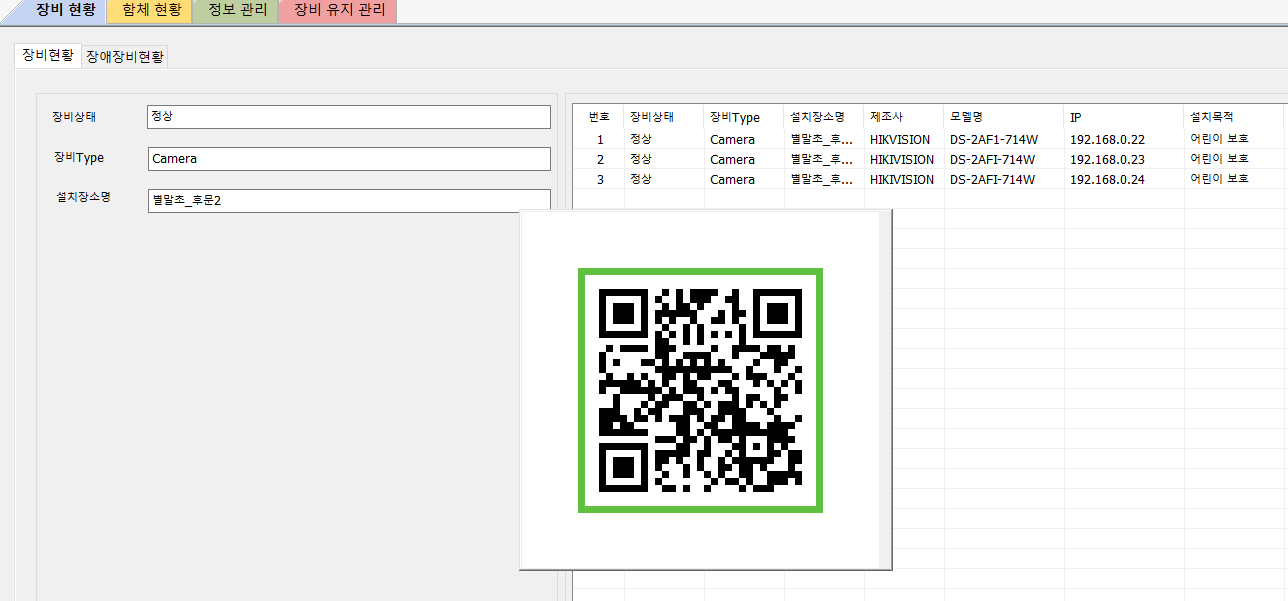


그림 15: QR 코드 출력 뷰

마지막으로 함체에 등록된 장비들의 QR Code를 출력하는 뷰입니다. 장비 현황 탭에서 리스트 컨트롤에 출력된 장비 목록에서 장비들을 더블클릭 하게 되면 해당 장비 모델명에 해당하는 QR Code를 출력합니다.

5. 프로젝트 수행 후기

|  |  |  |
| --- | --- | --- |
| 구분 | 이름 | 작성 내용 |
| 1 | 김수혁 | 이번 프로젝트를 통하여 여태까지 간단하게 과제로 했던 것과는 달리 실무에서 어떻게 하는지 방법을 알 수 있어 힘들지만 보람찼었다. |
| 2 | 김병욱 | 이번 프로젝트를 통해 이전에 접해보지 못했던 MFC에 대한 내용에 대해 많이 알게 되었기 때문에 공부에 있어서 굉장히 많은 도움이 되었던 것 같다. 특히 어렵게만 느껴졌던 윈도우 프로그래밍이 다양한 예제와 함께 진행하는 과정과, 팀프로젝트를 통해 협업하는 과정을 겪으면서 프로그래밍 실력이 많이 늘 수 있었던 좋은 기회가 되었던 것 같다. |
| 3 | 김진하 | 이번 학기에 MFC에 대해 처음 접했습니다. 작년에 모바일 안드로이드 프로그래밍이란 과목을 들었는데, 상당히 유사한 부분이 많아서 공부하면서 친숙해서 재미를 많이 느꼈습니다. 이번 기말 프로젝트를 진행하면서 수업시간에 다루지 않은 내용에 대해서 찾아가면서 코딩을 하는데 많은 애로사항이 있었지만, 그러한 시행착오를 통해서 많은 공부를 할 수 있었고 팀원들간에 협동심과 배려심을 기를 수 있었던 좋은 기회가 되었던 것 같습니다. |
| 4 | 한준희 | 실 세계의 모델을 기준으로 잡고, 구현하는 과정에서 다소 미숙한 프로그램 구현 수단을 이용하여 수행하려니 착잡하고 힘든 느낌이 많아 중간중간 포기 하고 싶은 마음도 있었다. 하지만 힘들어도 다 같이 하면 할 수 있다는 의의를 두고 열심히 함께 이 프로젝트에 임해준 팀원들 덕분에 포기하지 않고 열심히 했던 것 같아 힘든 만큼 뿌듯함을 느낄 수 있었다. |