

실시간 드론 관제 시스템 웹 애플리케이션

Real-time drone control system web application

요 약

최근 글로벌 기업인 구글, 아마존, DHL 등이 드론의 상업적 활용 사업을 선언하고 개인이 수월하게 운용할 수 있는 소형 드론이 개발되어 민수, 민간용 드론 시장이 매우 커지고 대중의 관심이 집중되고 있다. 점차적으로 상업적, 민간용으로 운용되는 드론의 개수가 증가한다면 여러 사용자가 드론의 위치 및 상태 정보를 얻을 수 있는 시스템이 필요할 것이다. 따라서 본 연구에서는 드론으로부터 실시간으로 정보를 받아와 디스플레이하는 웹 어플리케이션을 제시하고자 한다. [1]

1. 서 론

1.1 연구배경

무인항공기로 불리는 드론은 1900년대 초반 군사적 용도로 전투와 경찰의 목적으로 사용되기 시작하였다. 현재는 건설, 물류, 시설 관리, 재난구조, 교통 관측, 과학연구, 농업, 촬영, 취미 등 다양한 분야에서 활용되고 있고 최근 드론 관련 연구나 드론을 활용한 기술들이 쏟아져 나오고 있다. 아마존과 구글에서도 드론을 택배업에 활용할 계획을 세우고 있고 국내에서도 K-드론시스템 구축을 진행하며 드론 택배, 드론 배송 사업을 꿈꾸고 있는 추세이다.[2]



위 사진은 항공 관제 시스템 중 하나로 전 세계에서 운행 중인 항공기의 정보를 실시간으로 모니터링 하는 시스템이다.[3] 실시간 위치 뿐만 아니라 항공기와 항공편에 대한 정보까지 제공한다. 본 연구에서는 항공기가 아닌 드론을 대상으로 관제 웹 어플리케이션을 구축하고자 한다. 실시간으로 드론들의 위치를 추적하고 드론에 대한 정보 및 통계를 실시간으로 보여줄 것이다. 현재 본 연구를 진행하는 기업인 (주)트웬티온스에서 임무 특화 드론 훈련 시뮬레이터를 개발하여 활용하고 있다. 일반적인 상황에서 사용하는 드론이 아닌 어려운 상황이 바탕이 되는 시나리오에서 임무 특화적인 드론을 훈련할 수 있는

교육을 위한 가상 훈련 콘텐츠이다. 이후 본 프로젝트를 통하여 드론 데이터를 웹 어플리케이션에 전달하여 활용할 방안이며 더 나아가 국내 드론 정보를 등록하여 드론 관제 시스템 솔루션을 구축하는 것을 목표로 한다.

1.2 연구목표

사용자에게 실시간으로 드론의 위치를 보여주는 것을 목적으로 지도에 드론 위치 정보를 제공하여 사용자가 쉽게 눈으로 볼 수 있게 만들 것이며 웹으로 표현하여 어디서든 확인하도록 할 것이다.

첫 번째 목표는 지도에 드론 위치를 실시간으로 보여주는 것이다. 실시간으로 지도에 많은 데이터를 한 번에 표시하는 데 있어서 끊임이나 버벅거림을 최소화하여 사용자가 느끼기에 편하게 만드는 것이다.

두 번째 목표는 대용량 데이터를 저장하고 전송하는 데 있어서 빠르고 정확하게 데이터가 전송되어야 한다. 이를 위해 데이터를 안정적으로 저장하고 전송하는 시스템을 만드는 것이다.

2. 관련 연구

2.1. PX4 Autopilot

본 프로젝트와 유사한 소프트웨어인 드론 관제 시스템은 Pixhawk의 오픈소스 PX4 Autopilot를 확장하여 개발하는 경우가 많은데, PX4 Autopilot은 오픈 소스로 만들어진 드론 또는 기타 무인 차량을 제어하기 위한 소프트웨어이며, 드론 개발자들을 위해 드론 어플리케이션 제작을 위한 유연한 도구들을 제공한다.

이 소프트웨어는 드론 등과 같은 비행 차량에 대한 오픈 소스 소프트웨어 사용을 촉진하기 위해 만들어졌으며, Linux 재단에서 관리하는 비영리 조직인 Dronecode의 일부이다.

PX4의 주요 특징으로는 오픈 소스이며, 개발자를 위한 API와 SDK가 제공되어 있어 개발자 친화적이고 모듈식 구조를 취하

고 있어 확장 및 설정이 간편하다. 또한 BSD-3-Clause 라이선스 정책을 취하고 있어 상업적 사용 및 수정에 자유로우며 안정성이 보장되어 세계적으로 PX4 기반 시스템이 많이 배포되고 있다.

국내 기관들에서 운용하는 임무 전용 드론 관제 시스템은 이 Pixhawk Autopilot을 확장하여 개발된 소프트웨어를 사용하고 있다. 개별적인 드론 조종/설정 등을 하는데 효율적인 도구이나, 대량의 무인기/드론 정보 조회, 통계 기능을 제공하기에는 기능이 부족하다고 볼 수 있다.[4]

2.2. 웹소켓 통신

웹 소켓(Web Socket)은 순수 웹 환경에서 웹 클라이언트 - 웹 서버간 실시간 양방향 통신을 지원하는 프로토콜이다. 기존 브라우저의 통신 방식은 HTTP 요청(HTTP Request)에 대한 HTTP 응답(HTTP Response)을 받아서 브라우저의 화면을 지우고 받은 내용을 새로 표시하는 방식이다. 내용은 지우고 다시 그리면 브라우저의 깜빡임이 생기게 되며 이러한 깜빡임 없이 원하는 부분만 다시 그리고 실시간으로 사용자와 상호작용하는 방식이 웹 소켓(Web Socket)이다.

HTTP 프로토콜이 아닌 웹 소켓(Web Socket) 프로토콜을 사용하면 기존의 Polling 기법이나 Long Polling 기법의 불필요한 요청으로 인한 부하를 줄일 수 있다. 웹 소켓(Web Socket) 방식은 최초 연결 시 HTTP 요청(HTTP Request)의 Upgrade 헤더를 사용하여 웹 서버에 요청하고, 서버는 이를 확인하고 토큰을 생성하여 응답한다. 이러한 웹 소켓(Web Socket) 핸드셰이크 과정에서만 요청과 응답 헤더가 있고 그 이후 데이터 송수신에는 최소 2바이트 크기의 프레임을 사용하므로 네트워크의 부하가 감소하게 된다.

2.3. OpenStreetMap

OpenStreetMap(OSM)은 영국에서 출범한 오픈 소스 방식의 참여형 무료 지도 서비스이다. 개방형 공동 작업 프로젝트로 전 세계의 다양한 사람들이 집단 지성을 이용하여 만들어진다. 빠르게 업데이트 되고 무료이면서 공개적으로 사용할 수 있다는 장점이 있고 웹페이지나 GPS장치 오프라인 등에서도 지도 데이터를 사용할 수 있는 GPS좌표와 연동하는 API도 제공한다. 실제 유명한 모바일 게임인 ‘포켓몬 고’를 국내에서 서비스하기 위하여 구글 지도 대신 OSM을 사용하였다.[5]



2.4. Nuxt.js

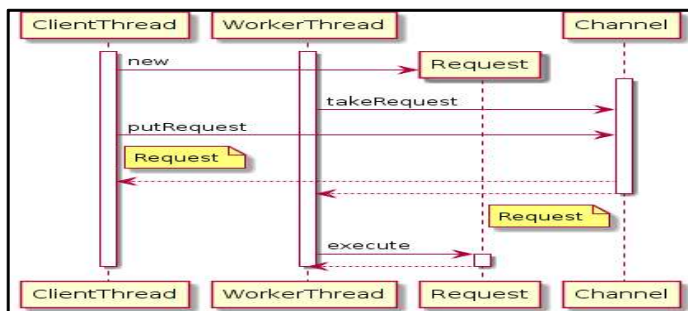
Nuxt.js는 Vue를 기반으로 한 웹 개발을 좀 더 쉽고 강력하게 만들어주는 오픈소스 프레임워크이다.

Nuxt의 주요 특징은 추상화된 Nuxt 전용 설정을 통하여 Vue, Webpack, Babel, Server 등의 웹 어플리케이션을 만들기 위한 설정을 간편하게 할 수 있다. 기본적으로 설정된 값은 일반적인 상황에서 Vue.js 웹 어플리케이션을 개발하기 위한 최적의 값으로 되어 있으며, 모든 설정 값은 확장 및 오버라이드가 가능하다.

Nuxt의 가장 큰 장점은 SPA(Single Page Application)으로 구성된 Vue 어플리케이션을 SSR(Server Side Rendering)로 프로젝트를 구성할 수 있다는 것이다. SSR을 통하여 Nuxt 프로젝트는 사용자의 초기 진입 경험을 개선할 수 있고, SEO(Search Engine Optimization)을 설정할 수 있다. Nuxt는 프로젝트를 생성할 때 SPA, Universal, SSG(Static Site Generate) 모드를 선택할 수 있는데 Universal 모드는 최초의 페이지 접근 시 서버에서 렌더링하여 로드되고 Rehydration 이후에는 SPA의 방식을 취하여 SSR과 SPA의 장점을 모두 취하고 있다. 그리고 Universal 모드로 프로젝트를 구성할 시, 프론트엔드 서버가 활성화 되어 http 통신에서의 header 등의 서버 레벨에서 설정할 수 있었던 것들을 프론트엔드 개발에서도 가능해진다.

또한 Nuxt는 Vue 어플리케이션과 달리 정해진 폴더 구조가 있으며 Nuxt에서 정해놓은 규칙대로 개발할 필요성이 있다. 이 규칙 중 가장 특징적인 부분이 파일 시스템 기반 라우팅과 저장소 구조인데, 생성한 폴더 및 파일의 이름을 기반으로 계층적 라우트 및 저장소 구조를 자동으로 생성해주어 해당 설정의 수고로움을 덜어준다. 이러한 Nuxt의 장점으로 많은 개발자들이 웹 프론트 엔드 개발의 프레임워크로 Nuxt를 채용하고 있으며 stateofjs의 2020년 자바스크립트 프레임워크 설문결과에서 88%의 높은 만족도를 보여주었다.[6]

2.5 Node.js Event Loop

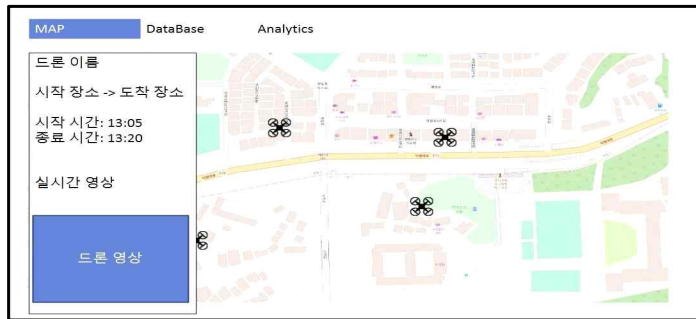


대용량의 데이터를 실시간으로 전송하면서 빠르고 손실없이 제공하며, 또한 기록되어야 한다. 하나의 프로세스에 있어서 작업 처리시간이 오래 걸리거나, 다양한 곳에서 같은 로직을 처리해야 하는 상황이 발생한다. 이러한 상황에서 Producer/Consumer 패턴, Worker Thread 패턴을 사용하여 큰 데이터 흐름을 사용자에게 오랜 시간동안 기다리지 않고 뒤에서 처리하는 패턴들이 있다. 대용량의 데이터는 사용자에게 전

달되어야 하며 전달되는 도중 해당 데이터는 추가적으로 데이터베이스에 저장되어야 한다. 데이터베이스에 저장되는 시간동안 사용자에게 데이터 전달에 있어서 시간이 지연 될 것이다. 데이터를 전송하는 프로세스에서 데이터를 저장하는 기능은 worker로 전달만 하여 데이터 전송지연이 없도록 처리하며 worker는 데이터 전송과는 별개로 저장하는 작업을 진행한다. 이러한 패턴들에 대해서 Node.js는 Event Loop라는 기능을 제공한다. Node.js는 Single Thread 기반의 Non-Blocking I/O 방식으로 이와같은 패턴을 코드로 해결할 수 있다. 데이터를 제공하면서 기록하는 부분을 별도의 Event Loop로 전송하여 별도로 데이터를 저장을 할 수 있다.

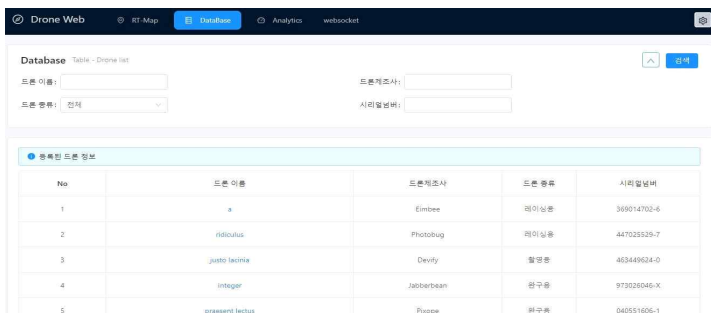
3. 프로젝트 내용

3.1 Main-Map



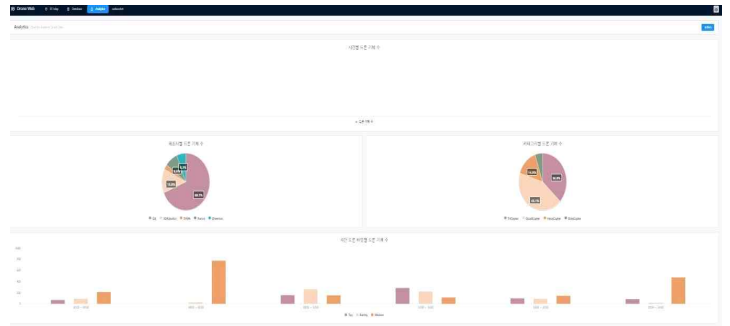
웹 어플리케이션의 메인 화면이다. 메인화면에는 지도화면과 현재 비행하고 있는 드론의 움직임을 지도위에 보여준다. 드론 아이콘을 클릭하면 그림과 같은 화면이 출력된다. 위 그림은 클릭한 드론에 대한 정보를 왼쪽 레이아웃에 표시해주도록 한다.

3.2. Database



DataBase는 드론정보, 실시간 데이터, 비행 스케줄에 대한 데이터를 보여주는 UI이다. 우선, 상단의 탭에 따라 드론기체, 실시간데이터, 비행스케줄의 정보를 테이블의 형태가 다르게 나타난다. 테이블 위에는 필터링 옵션이 들어가있고, 각기 입력한 정보에 따라 이에 맞는 드론의 리스트가 화면에 출력된다.

3.3 Analytics



Analytics는 드론의 통계정보를 담은 화면이다. 화면 중앙에는 작은선 그래프가 표시되도록 하며, 하단에는 원형 차트가 2개 정도 배치되도록 한다. 꺾은선 그래프의 값에 접근이 이것이 무엇을 의미하는지에 대해 스티커 메시지 형태로 보여주도록 한다.

4. 결론 및 기대효과

본 연구의 어플리케이션은 가상현실 속 드론 혹은 실제 드론의 실시간 위치와 상태(배터리 상태, IMU 및 GPS 신호, 목적지 정보 등)를 추적하여 사용자에게 웹으로 정보를 제공한다. 제공된 정보를 통하여 운용 중인 드론들을 파악함으로써 드론 관련 사고 및 오용을 예방할 수 있고, 사고 발생시 위치 파악 및 즉각 대응이 가능할 것이다.

또한 실시간으로 수집한 드론 데이터를 빅데이터화하여 분석 및 시각화를 통해 드론 산업 현황을 파악할 수 있을 것이다.

현재 드론 비행의 허가 및 관리에 대한 법제화가 되지 않은 상태이지만, 앞으로 드론 비행을 하기 위해 신고를 하고 실시간 정보를 제공해야 하게 될 것으로 보고 여러 방법으로 유용하게 활용될 것이라고 예측한다.

참 고 문 헌

- [1] 윤광준, 드론 핵심 기술 및 향후 과제, 광학세계, 158권, 52-54, 2015년.
- [2] 정삼철, 드론산업 발전 동향과 활용 사례, 충북연구원, 2015년.
- [3] "항공편 추적기/항공편 상태"[FlightAware]. (2021.04.25). URL: <https://ko.flightaware.com/>
- [4] "Open Source Autopilot for Drones"[PX4 Autopilot]. (2021.04.25). URL: <https://px4.io/>
- [5] "Nuxt.js" [NUXTJS]. (2021.04.25.) URL: <https://ko.nuxtjs.org/>
- [6] "오픈스트리트맵"[OpenStreetMap]. (2021.04.25.). URL: <https://www.openstreetmap.org/>