

선박운항 시뮬레이터용 웹 프레임워크 개발에 관한 연구

김혜진* · 오재용**† · 박세길**

*, ** 한국해양과학기술원 부설 선박해양플랜트연구소

Development of Ship-Handling Simulator Web Framework

Hye-Jin Kim* · Jaeyong Oh**† · Sekil Park**

*, ** Korea Research Institute of Ships and Ocean Engineering, Daejeon 34103, Korea

요약 : 본 논문은 시뮬레이터 시스템이 가지는 복잡성과 확장성 문제를 해결하기 위한 방법으로 웹 기술을 이용한 선박운항 시뮬레이터 프레임워크를 제안한다. 기존 시뮬레이터 시스템의 필수 기능을 분석하고, 이를 대체할 수 있는 웹 기반 기술을 선별하여 프레임워크를 구성하였다. 또한, 이를 바탕으로 서버와 클라이언트 핵심 기술을 구현하였으며, 이를 통합하여 웹 브라우저를 통해 사용이 가능한 선박운항 시뮬레이터 시스템의 시작품을 제작하였다. 제작된 시작품은 시험평가를 통해 시간과 장소의 제약을 받지 않고 다수의 사용자가 동시에 시뮬레이션 서비스를 이용할 수 있음을 확인하였으며, 향후 다양한 시뮬레이션 분야에 적용될 수 있기를 기대한다.

핵심용어 : 선박운항 시뮬레이터, 해양 안전, 플랫폼, 웹 기반, 프레임워크, 시뮬레이션 서비스

Abstract : This paper proposes the ship-handling simulator framework using web technology to solve the complexity and non-scalability problems of a common simulator system. We analyze the essential functions of the existing simulator system, and selects web technologies to replace them. Based on this framework, we implemented core technologies for server and client system, and integrated a prototype of a ship-handling simulator system that can be accessed from a web browser. The prototype has verified that the simulation service can be used by multiple users at the same time without being restricted by time and place through the evaluation test, and it is hoped that it can be applied to various simulation fields in the future.

Key Words : Ship-handling simulator, Maritime safety, Platform, Web-based, Framework, Simulation service

1. 서 론

선박운항 시뮬레이션 기술은 선박의 움직임을 수학적으로 모델링하여 이를 사실적으로 모사하는 기술이다. 이러한 기술을 기반으로 선박의 선교(bridge)와 유사하게 제작된 환경에서 선박운항을 가상으로 체험할 수 있도록 설계된 시스템이 선박운항 시뮬레이터 시스템이다. 이는 주로 항해사의 운항 및 선원의 선교 업무 숙지 훈련에 활용되며, 안전 항로와 같은 항만 시설의 통항 안전성을 평가하기 위한 목적으로 사용되기도 한다. 최근 해양사고의 규모가 커지고 해양 안전에 대한 관심이 증가하면서 선박운항 시뮬레이터 시스템

을 활용하고자 하는 시도가 다양해지고 있다. 이에 따라 시뮬레이터 시스템에 대한 요구사항이 증가하고 있으며, 이와 관련한 요소 기술들에 대한 연구가 활발히 진행되고 있다. 그러나 기존의 선박운항 시뮬레이터 시스템은 그 구조가 복잡하고 폐쇄적인 경우가 대부분이어서 기능의 수정과 확장이 어려우며, 이 때문에 다양한 분야에의 활용이 거의 불가능한 실정이다. 실제로 현재 운영되는 일부 선박운항 시뮬레이터 시스템의 경우, 하드웨어는 물론 소프트웨어의 업그레이드가 거의 불가능하여 기본적으로 제공되는 기능 이외에는 활용되지 못하고 있으며, 시뮬레이터 시스템을 운영하는 시간과 공간이 제한되어 있어서 그 활용에 제약이 따르기도 한다.

이러한 문제점을 개선하기 위해 본 논문에서는 웹 환경에서 운영될 수 있는 선박운항 시뮬레이터용 프레임워크를 제안하며, 시작품 제작 및 시험 평가를 통해 적용 가능성과 효용성을 고찰하고자 한다.

* First Author : hjk@kriso.re.kr, 042-866-3649

† Corresponding Author : ojyong@kriso.re.kr, 042-866-3648

* 이 논문은 “해기반 선박운항 시뮬레이터 시스템의 설계 및 구현”이란 제목으로 “2016년 해양환경안전학회 춘계학술발표회(2016.04, pp. 149)”에 발표되었음.

2. 선행 연구 분석

일반적인 선박운항 시뮬레이터 시스템은 Fig. 1과 같이 조종실과 통제실로 구분되며, 개별적인 기능을 수행하는 수십대의 장치들이 네트워크에 연결되어 시뮬레이션 작업을 수행하게 된다. 따라서 작은 변경 사항에도 시스템 내의 모든 장치들이 영향을 받게 되며, 이러한 복잡성 때문에 시뮬레이터 시스템의 기능을 수정하거나 추가하는 작업이 어려워지게 된다.

이러한 시스템의 유지보수 문제는 선박운항 시뮬레이터의 설계 단계에서부터 고려되어야 하며, 이를 위해 시스템의 유지보수 관점에서의 설계 기법이 제안되기도 했다(Park et al, 2012). 또한, HLA(High Level Architecture) 혹은 DIS(Distributed Interactive Simulation)와 같은 규격화 된 연동 체계를 이용하여 시스템의 확장성을 확보하기 위한 연구가 진행되고 있지만, 시스템이 더 복잡해지고, 추가적인 통신 부하가 발생하는 문제점이 있다(Kim et al, 2009).

한편, 웹 기반 기술은 사용자의 플랫폼에 제약을 받지 않고 다수의 사용자가 동시에 사용할 수 있기 때문에 시스템의 활용도를 극대화 할 수 있으며, 서버 시스템에서 데이터가 통합 관리되어 시스템의 확장과 변형이 용이한 특징을 가지고 있다. 특히, 하드웨어 의존성이 낮기 때문에 모바일 장치를 위한 소프트웨어 플랫폼으로도 널리 활용되고 있다.

이러한 웹 기술은 MSaaS(M&S as a Service)의 개념으로 군사용 훈련 시뮬레이션 분야에의 적용이 시도되고 있다. 네트워크를 통해 서버에 접속하여 시뮬레이션 서비스를 제공 받기 때문에 시스템의 구축비용이 절감되고, 사용자 접근성을 향상시키는 효과가 있으며, 이 기술은 현재 NATO의 M&S 그룹에 의해 표준화가 진행 중이다(NATO, 2017).

이와 같은 배경에서 본 논문에서는 시뮬레이터 시스템의 접근성과 확장성을 개선하기 위한 방법으로 웹 기술을 적용한 시뮬레이터 프레임워크를 설계하고, 이를 기반으로 시뮬레이터 시스템을 구현하여 활용 가능성을 확인하고자 한다.

3. 웹 기반 선박운항 시뮬레이터 시스템

3.1 프레임워크 설계

본 연구에서는 웹 기반 프레임워크를 설계하기에 앞서 선박운항 시뮬레이터를 구성하는 요소 기술을 식별하고 각 기술별 요구사항을 분석하였다. 분석 대상은 선박해양플랜트 연구소(KRISO)의 FMB(Full Mission Bridge)급 선박운항 시뮬레이터이며, 세부 사항은 다음과 같다.

시뮬레이터 시스템은 기본적인 시뮬레이션 기능을 수행하기 위해 시뮬레이션 제어기술, 선박운동 모의기술, 맵 전시기술, 데이터베이스 관리기술, 네트워크 기술, 3D 가시화 기술 등이 통합되어 하나의 시뮬레이션 작업을 수행하도록 되어있다.

먼저, 시뮬레이션 제어 기술은 시뮬레이션 시간, 운항 환경 등을 제어하는 기술이며, 다양하고 복잡한 작업을 동시에 수행하는 경우가 많아서 사용자 친화적인 인터페이스가 필수적이다. 맵 전시 기술은 전자해도(ENC, Electronic Navigation Chart) 데이터를 처리할 수 있는 기술과 실시간 데이터를 중첩 표시할 수 있는 기능을 포함해야한다. 또한 데이터베이스 기술은 다양한 조건의 시뮬레이션 시나리오를 체계적으로 관리할 수 있어야 하며, 대용량의 시뮬레이션 수행 결과를 효과적으로 저장하고 검색할 수 있어야 한다.

이와 같은 요구사항들을 반영하여 프레임워크를 구성하는 웹 기술들을 선별하였으며, Fig. 2 와 같이 4 단계의 계층 구조를 가지는 선박운항 시뮬레이터 개발용 프레임워크를 설계하였다. 각 계층의 세부사항은 다음과 같다.

1) 데이터베이스 계층

최하위 계층인 데이터베이스 계층은 데이터의 물리적인 저장소를 의미하며, GIS 데이터의 저장을 위한 GEO 데이터베이스, 시뮬레이션 데이터를 저장하는 시뮬레이션 데이터베이스, 3D 모델 데이터베이스 시스템으로 구성된다.

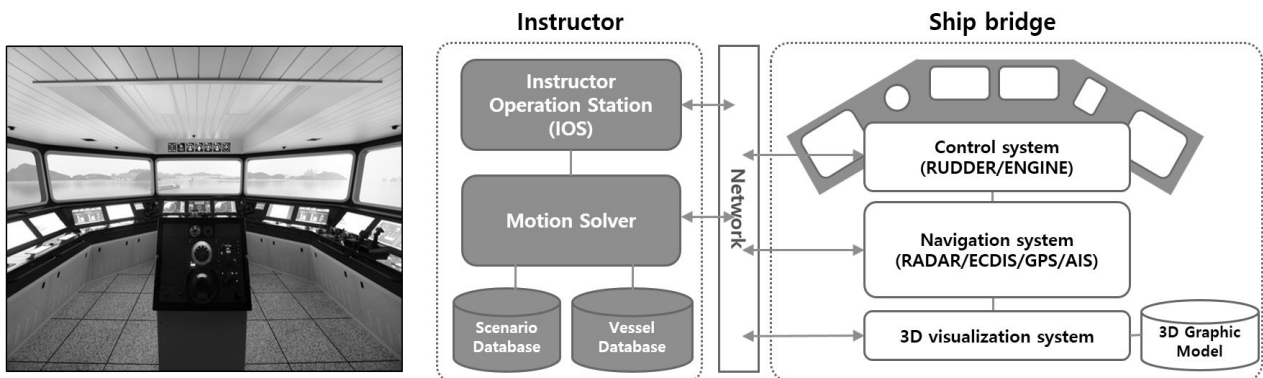


Fig. 1. Configuration of ship-handling simulator system by KRISO.

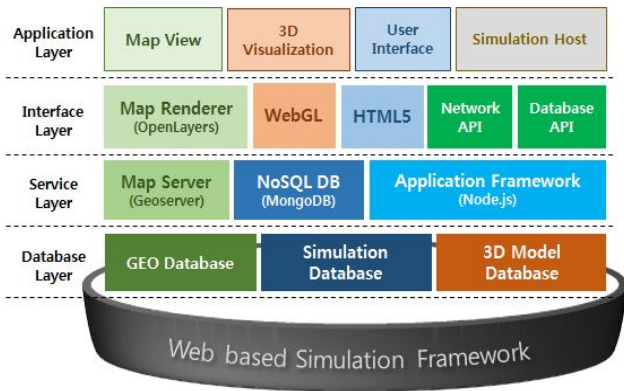


Fig. 2. Web-based simulator framework.

2) 서비스 계층

서비스 계층은 데이터베이스 계층의 데이터를 효율적으로 관리하고 상위 계층의 요청에 따라 빠르고 정확하게 데이터를 처리하는 기능을 수행하며, 웹 맵 서비스와 데이터베이스 서비스, Node.js 기반의 네트워크 프레임워크 서비스로 구성된다.

3) 인터페이스 계층

인터페이스 계층은 응용 프로그램 계층으로부터 접근이 가능한 프로그래밍 인터페이스를 제공하는 역할을 하며, 웹 환경에서의 접근성을 고려하여 자바 스크립트 언어를 사용한다. 인터페이스 계층은 웹 페이지 생성을 위한 HTML5 기술과 데이터베이스의 접근 및 관리를 위한 API(Application Programming Interface)를 제공하며, TCP와 UDP를 이용한 네트워크 통신 기능을 포함하고 있다.

4) 어플리케이션 계층

어플리케이션 계층은 인터페이스 계층의 API를 이용하여 구현된 선박운항 시뮬레이터 필수 응용 프로그램으로 구성된다. 맵 전시 기술, 3D 가시화 기술, 사용자 인터페이스 기술, 시뮬레이션 호스트 기술과 같이 앞서 언급된 시뮬레이터 시스템의 요소 기술들이 구현되어 있으며, 어플리케이션 계층의 구성에 따라 다양한 형태의 시뮬레이터 시스템을 구현할 수 있게 된다.

3.2 서버 시스템

서버 시스템은 크게 시뮬레이션 호스트 모듈, 네트워크 모듈, 데이터베이스 모듈로 구성되며, 프레임워크를 기반으로 웹을 통해 선박운항 시뮬레이션 서비스를 제공하는 역할을 수행한다. 웹 기반 서비스를 수행하기 위해서는 보다 빠른 이벤트 처리가 필요하며, 동시에 접속하는 많은 사용자에게 원활한 서비스를 제공할 수 있어야 한다. 이러한 요구

사항을 충족하기 위해 본 연구에서는 Node.js 기반의 웹 서비스 모듈을 설계하고 구현하였으며, 이를 중심으로 NoSQL 데이터베이스인 MongoDB를 연동하여 동작하도록 하였다. 이렇게 서버 시스템을 기반으로 하는 중앙 집중형 데이터 관리 방식은 시스템 자원의 사용률을 극대화하여 시스템 구축비용을 절감할 수 있으며, 시스템의 유지보수가 용이해지는 효과가 있다.

Fig. 3은 시뮬레이션 서버 시스템의 동작 절차를 나타낸다. 첫 번째 단계로, 웹 브라우저와 웹 서버의 데이터 통신에 의해 사용자의 입력이 시뮬레이션 호스트 모듈에 전달되면, 선박운동 모의 모듈에 의해 생성된 시뮬레이션 데이터가 데이터베이스 모듈과 네트워크 모듈을 거친 후, 최종적으로 웹 프로토콜을 통해 사용자의 웹 브라우저에 전달된다. 시뮬레이션 호스트 모듈은 전체 시뮬레이션을 통제하는 기능을 수행하며, 선박운동 모의 모듈과 연동하여 시뮬레이션 결과 데이터를 실시간 생성한다. 또한, 데이터베이스 시스템과 연동하여 시뮬레이션 시나리오 및 결과 데이터를 관리하도록 하였으며, 여러 개의 시나리오가 동시에 실행될 수 있도록 다중-스레드(multi-thread) 구조를 적용하여 구현하였다. 네트워크 모듈은 모듈 간 데이터 통신을 위한 인터페이스를 제공한다. 웹 브라우저와 웹 서버 모듈은 Socket.io 프로토콜을 사용하며, 선박운동 모의 모듈은 UDP 방식으로 데이터를 송·수신한다. 송수신 되는 데이터는 JSON(Javascript Object Notation) 포맷을 사용하여 MongoDB와의 호환성을 유지하도록 하였다. 데이터베이스 모듈은 시뮬레이션 시나리오 데이터 뿐만 아니라 시뮬레이션에서 생성되는 모든 결과 데이터를 저장하고 관리한다. 기존의 시스템은 시뮬레이션 시나리오의 관리 체계가 복잡하고 대용량의 데이터 백업 기능을 지원하지 않는 문제점이 있었다. 이를 개선하기 위해 본 연구에서는 대용량 데이터의 분산 처리가 가능한 MongoDB 엔진을 적용하였다. 데이터베이스에 저장되는 데이터는 시나리오 데이터, 시뮬레이션 결과 데이터, 선박운동 특성 데이터이며, 기존 선박운항 시뮬레이터 시스템의 데이터 모델을 기반으로 Fig. 4와 같이 모델링하였다.

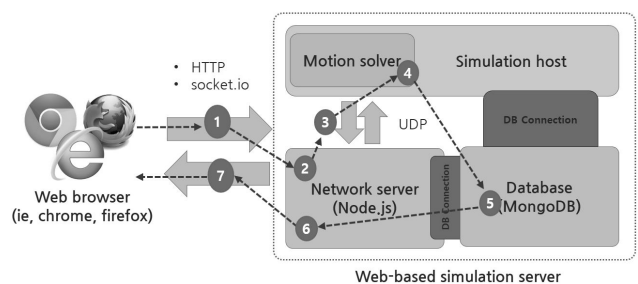


Fig. 3. Operation of web-based simulation server.

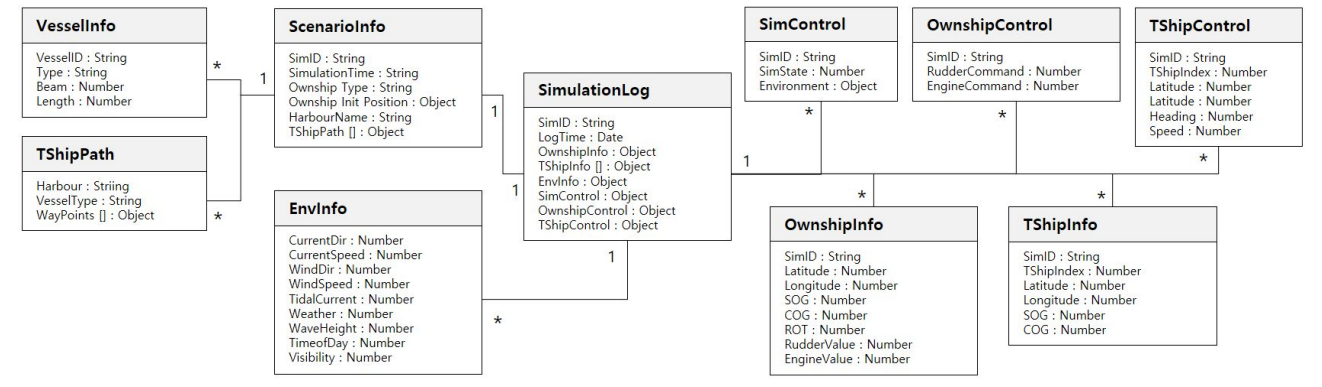


Fig. 4. Data model of ship-handling simulation service.

3.3 클라이언트 모듈

클라이언트 모듈은 맵 전시 모듈, 3D 가시화 모듈, 사용자 인터페이스 모듈로 구성되며, 사용자의 웹 브라우저를 통해 시뮬레이션을 제어하고, 서버로부터 시뮬레이션 데이터를 실시간 수신하여 시뮬레이션 상태를 전자해도 혹은 3차원 공간상에 전시하는 기능을 수행한다.

맵 전시 모듈은 전자해도 데이터를 바탕으로 실시간 시뮬레이션 데이터를 중첩하여 표시하는 기능을 수행한다. 일반적으로 전자해도 데이터는 지리참조, 속성, 공간정보로 구성되며, 화면 표시 방법 또한 표준화 되어있다. 이러한 전자해도 데이터는 항해 장비에의 적용을 목적으로 개발되었기 때문에, 웹 서비스용 데이터 포맷으로는 적합하지 않다. 따라서 본 연구에서는 오픈소스 GIS 소프트웨어인 QGIS를 이용한 전자해도 데이터 가공 절차를 정의하였으며, 이를 체계화하고, 자동화 할 수 있도록 하였다.

웹 서비스용 맵 데이터를 생성하기 위한 첫 번째 절차로 전자해도 파일의 레이어를 선별한다. 전자해도 파일은 각 데이터가 특성별로 분류되어 있고, 점, 선, 면 정보를 포함한 다양한 레이어로 구성되어 있으며, 이러한 레이어 정보 중 웹 서비스에 불필요한 레이어를 삭제하고, 표시 순서를 설정하여 웹 서비스 성능을 향상시킬 수 있게 된다. 다음 단계로 전자해도에 표시되는 심볼을 벡터 이미지 형태로 생성하며, 각 레이어의 색상 및 표시 방법 등의 스타일을 설정한다. 이때 전자해도 심볼은 IHO ECDIS Presentation Library 설정 파일을 참고하여 벡터 이미지 형태의 심볼로 변환하여 사용한다.

한편, 전자해도 데이터는 셀 단위로 분할되어 있으며, 웹 맵 서비스를 위해서 각 데이터마다 레이어를 선별하고, 스타일을 지정하며, 심볼을 변환하는 과정을 거쳐야 한다. 이러한 작업은 단순한 반복 작업이 대부분이어서 수동으로 작업하는 경우 오류가 발생할 수도 있게 된다. 이에 본 연

구에서는 QGIS의 파이썬 스크립트 기능을 이용하여 일련의 처리 과정을 체계화하고, 이를 자동화 할 수 있는 스크립트 코드를 작성하여 작업 시간 및 오류를 줄일 수 있도록 하였다.

이렇게 변환된 전자해도 데이터를 웹을 통해 서비스하기 위해서는 웹 맵 서비스용(WMS, Web Map Service) 서버 소프트웨어가 필요하며, 본 연구에서는 오픈소스 GIS 서버인 지오서버를 사용하였다. 맵 서비스를 위해서는 맵 데이터별 서버 저장소에 등록하는 절차를 거쳐야하며, 본 연구에서는 지오서버에서 제공하는 REST(REpresentational State Transfer) API를 이용하여 대용량의 전자해도 데이터를 맵 서버에 자동으로 등록할 수 있도록 하였다.

한편, 클라이언트의 웹 브라우저에서 맵 데이터를 출력하기 위해 오픈 레이어즈(OpenLayers)를 사용하였다. 오픈 레이어즈는 웹 브라우저 상에 지도 데이터를 표시할 수 있도록 하는 오픈 소스 자바스크립트 라이브러리이며, 이를 이용하여 지오서버에 등록된 전자해도를 바탕에 출력하고 그 위에 서버로부터 수신된 시뮬레이션 데이터를 중첩하여 표시함으로써, Fig. 5와 같이 시뮬레이션 상황을 실시간으로 모니터링 할 수 있도록 하였다.

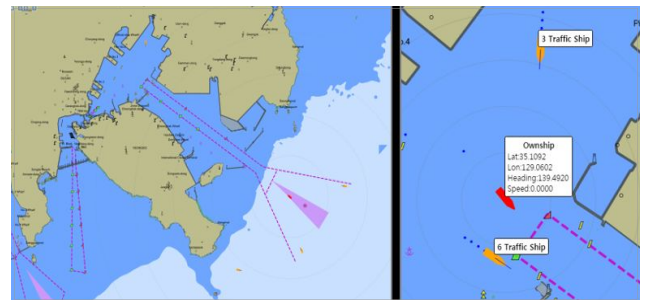


Fig. 5. Example of map display.

3D 가시화 모듈은 가상의 3차원 공간상에 선박운항 시뮬레이션 상황을 실시간 재현하는 모듈이다. 3차원 가시화 모듈 또한 웹 브라우저에서 구동되기 때문에 렌더링 성능의 최적화가 필수적이기 때문에 기존의 선박운항 시뮬레이터 시스템의 가시화 기능을 분석하여 주요 가시화 요소를 식별하였다. 식별된 주요 가시화 요소는 다음 표와 같다.

Table 1. Visualization elements of 3D module

| item | details |
|-------------|--|
| Weather | clear, wind, rain, storm, snow, fog, cloud, lightning |
| Time of day | day, sunset, dusk, night |
| Ocean | ocean surface, waves, reflection, refraction, floating effects |
| Lights | navigation lights, search lights, AtoN |
| Effects | flame, smoke |
| 3D models | harbor, vessels, AtoN |

3차원 가시화 모듈은 주요 웹 브라우저 호환성을 고려하여 Unity3D를 가시화 엔진으로 사용하였다. Unity3D는 크로스 플랫폼을 지원하는 3차원 그래픽스 엔진으로 iOS, Windows, Linux, Android 등 다양한 운영체제를 지원하며, 특히 웹 브라우저에서 3차원 가시화 소프트웨어를 구동할 수 있는 플러그인 소프트웨어(web player)를 제공한다. 웹 플레이어는 사용자의 웹 브라우저에 최초 실행 시에만 설치되어 구동되며, 3차원 모델 데이터를 서버로부터 다운로드하여 실행되기 때문에 클라이언트의 렌더링 성능을 극대화할 수 있다. 또한, 선박운항 시뮬레이터를 위한 가시화 모듈은 해양 환경에 대한 가시화가 주된 기능이며, 본 연구에서는 웹 플레이어를 지원하고, 스프레이, 스플래시 등의 효과 기능을 포함하고 있는 “NeoOcean” 에셋을 사용하여 구

현하였다. 이렇게 구현된 가시화 모듈은 네트워크 모듈의 Socket.io 기능을 사용하여 시뮬레이션 데이터를 서버로부터 실시간 수신하며, 수신된 데이터는 웹 플레이어 객체에 전달되어 시뮬레이션 호스트와 연동된다.

사용자 인터페이스 모듈은 클라이언트의 웹 브라우저에 출력되는 화면을 구성하는 모듈이며, HTML5와 자바 스크립트 기술을 이용하여 구현하였다. 특히 모바일 장치의 연동을 위해 QR 코드를 이용한 모바일 인터페이스를 구현하였다. 모바일 장치는 휴대성이 높아서 그 활용도가 높지만, 하드웨어적인 특성상 키보드를 이용한 사용자 입력이 자유롭지 못한 단점이 있다. 시뮬레이션 시스템에서는 시나리오 리스트를 검색하거나 해당 시나리오 정보를 조회하기 위해서는 시뮬레이션 ID를 입력하는 과정이 필수적이지만 모바일 장치에서는 매우 번거로운 작업이다. 본 연구에서는 이러한 문제점을 해결하기 위해 QR 코드에 시뮬레이션 시나리오 ID 정보를 삽입하여 사용자가 해당 시나리오 정보를 조회할 수 있도록 하였다.

3.4 웹 기반 선박운항 시뮬레이터 시작품 제작

웹 기반 프레임워크를 기반으로 구현된 서버 모듈과 클라이언트 모듈을 모두 통합하여 웹 기반의 선박운항 시뮬레이터 시스템의 시작품을 제작하고, 이를 시험 평가할 수 있는 테스트베드를 구축하였다. 시작품은 웹 서버를 통해 제공되는 웹 페이지 형태로 제작하였으며, 사용자 로그인, 시뮬레이션 시나리오 목록 검색, 시나리오 생성, 타선 경로 관리, 시뮬레이션 제어, 맵 전시, 3D 가시화, 서버 시스템 모니터링 기능을 각 페이지에 구분하여 구현하였다. 테스트베드는 MS windows server 기반의 서버 시스템에 Node.js 기반 웹 서버, MongoDB 서버, 지오서버를 설치하여 운영하며, 다양한 운영체제를 가진 모바일 장치를 포함하여 모바일 연동 기능을 테스트 할 수 있도록 하였다. 제작된 시작품의 주요 화면은 Fig. 6과 같다.



(a) Scenario creation page

(b) Simulation control page

(c) 3D visualization page

Fig. 6. Screenshots of prototype.

4. 성능 평가 및 결과 고찰

구축된 테스트베드를 이용하여 시작품의 구현 기능을 확인하고, 성능을 평가하였다. 실험은 다음 표와 같은 조건에서 수행되었으며, 서버 시스템의 부하 및 웹 브라우저 호환성을 평가하였다.

Table 2. Experiment condition

| conditions | | details | |
|------------|----------------------|------------------------------------|--------------------|
| server | H/W spec. | CPU | Intel Zeon 3.0GHz |
| | | MEMORY | 24 GB |
| | | OS | MS windows 10 pro. |
| | concurrent users | 10 users with each scenario | |
| ships | 5 ships per scenario | | |
| client | OS | windows, android, iOS | |
| | browser type | internet explorer, firefox, chrome | |
| | browser tabs | 4 tabs per client | |

서버 시스템의 부하 테스트는 10명의 사용자에게 의해 생성된 서로 다른 시뮬레이션 시나리오 10개를 동시에 실행하는 환경에서 진행되었다. 각 시나리오는 5척의 선박을 포함하고 있으며, 접속하는 클라이언트 시스템에서는 실시간 데이터를 수신하는 4개의 페이지를 동시에 실행한 후 서버 시스템의 CPU, 메모리, I/O에 대한 사용률을 측정하였다. 다음 그림은 각 항목의 사용률 그래프이다. 실험이 진행되는 동안 모든 항목에서 평균 10% 미만의 사용률을 보였으며, 모든 사용자 화면이 정상적으로 동작함을 확인하였다.

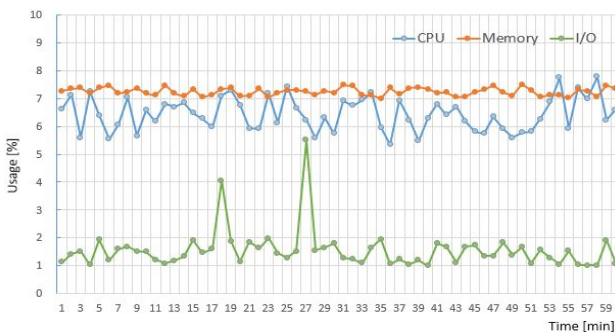


Fig. 7. Resource usage graph of server system.

Table 3 는 클라이언트의 운영체제(3종)와 주요 웹 브라우저(3종)에 대한 호환성 실험 결과이다. 웹 UI, map view 페이지, 3D 가시화 페이지에 대한 호환성 테스트를 수행하였으며, 대부분의 페이지가 플랫폼에 제약을 받지 않고 동작함을 확인하였다. 그러나 3D 가시화를 위한 Unity3D의 웹

플레이어가 MS windows의 IE 환경에서만 정상적으로 동작하였으며, 다른 웹 브라우저에서는 지원하지 않는 문제점이 발생하였다. 대부분의 웹 브라우저에서 외부 플러그인 방식을 지양하고 있음을 고려할 때, 향후 호환성 확보를 위해서는 WebGL과 같은 표준화 된 기술을 적용해야 할 것으로 판단된다.

Table 3. Experiment results of browser compatability

| OS | browser | web UI | map view | 3D |
|---------|---------|--------|----------|----|
| windows | IE | ○ | ○ | ○ |
| | chrome | ○ | ○ | × |
| | firefox | ○ | ○ | △ |
| android | chrome | ○ | ○ | × |
| | firefox | ○ | ○ | × |
| iOS | safari | ○ | ○ | × |
| | chrome | ○ | ○ | × |

5. 결론

웹 환경은 플랫폼의 제약이 적으며, 사용자가 쉽게 접근할 수 있다는 장점을 가지고 있다. 최근 웹 기술의 발달로 다양한 기능들의 구현이 가능해졌으며, 응용 프로그램으로서의 충분한 성능을 가지게 되었다. 본 논문에서는 이러한 웹 기술을 이용한 선박운항 시뮬레이터 프레임워크를 제안하였으며, 시작품 제작과 시험평가를 통해 적용 가능성을 확인하였다. 본 연구에서 개발된 시뮬레이터 시스템은 웹을 통해 시간과 장소의 제약을 받지 않고 다수의 사용자가 동시에 사용할 수 있는 장점이 있으며, 이러한 장점으로 다양한 시뮬레이션 분야에 효과적으로 적용될 수 있기를 기대한다.

사 사

본 논문은 선박해양플랜트연구소의 주요사업인 “해양사고 재현 및 분석 시뮬레이션 핵심기술개발-2단계”에 의해 수행되었습니다(PES8960).

References

[1] Kim, M. K, J. H. Kim and S. H. Chae(2009), Object management Method for Air Traffic Control Simulation Based on HLA/RTI, The Journal of Korea Navigation Institute, Vol. 13, No. 2, pp. 165-171.

- [2] NATO(2017), Modelling and Simulation as a Service (MSaaS) Rapid deployment of interoperable and credible simulation environments (MSG-136).
- [3] Park, S. K, J. Y. Oh, H. J. Kim and A. Y. Kim(2012), The study on the design and development of ship-handling simulator for effective maintenance, Proceedings of Korean Society of Marine Environment & Safety, pp. 171-174.

Received : 2017. 03. 07.

Revised : 2017. 04. 11. (1st)

: 2017. 04. 21. (2nd)

Accepted : 2017. 04. 27.