

[제22회 임베디드 소프트웨어 경진대회]
부문별 세부 안내사항

- webOS 부문 -

webOS 오픈소스를 활용한 스마트 팜 솔루션 개발

2024. 05

1. 부문별 목적 및 개요

가. 목적 : LG전자에서는 학생들이 webOS 오픈소스를 활용한 다양한 서비스와 기능 등에 대한 기획 및 개발, 시제품 제작 등 제품 개발의 전반적인 프로세스를 경험하게 하고 우수한 webOS 관련 작품을 발굴·선정합니다.

나. 개요

- 주제 : webOS 오픈소스를 활용한 스마트 팜 솔루션 개발

생산, 유통, 소비 등 농업활동 전반에 걸쳐 ICT 기술과 접목하여 농업을 보다 지능화하고 생산성을 향상시키기 위해 webOS 오픈소스를 활용하여 사용자 친화적 인터페이스와 데이터를 기반으로 시스템을 제어하거나 자동화 하는 솔루션을 개발하는 것을 목표로 함

- 개발 플랫폼 : webOS OSE 2.26.0 이상 (필수)

- 개발 언어 : 플랫폼 애플리케이션 - HTML / JavaScript / Enact 등
 플랫폼 서비스 - Node.js
 그 외 - 별도 제한 없음

- 개발 재료 : 플랫폼 - 라즈베리파이 4 x 1 ea
 - HDMI 지원 FHD 터치지원 디스플레이 1~2 ea
 카메라 - V4L2 지원 웹 캠 x 1 ea
 ESP32-CAM x 1 ea

아두이노 - 필요 시 제공 (아두이노 R3, ESP32, ESP8266)

그 외 - 별도 제한 없음.

단, 쉽게 부러지거나 다칠 위험이 있는 재료는 피해 주시기 바랍니다.

- 개발 규격

·소프트웨어

필수	webOS 플랫폼에서 동작하는 하나 이상의 web 기반의 앱을 개발해야 합니다.
가산점	webOS 플랫폼에서 동작하는 하나 이상의 Node.js 기반의 서비스를 개발해야 합니다. (서비스 레벨에서의 센서 통신 / 다른 webOS 앱과의 연동 / Notification 등)
가산점	AI/Interactive 기술을 적용해야 합니다.

·하드웨어

기본으로 라즈베리파이(webOS 탑재) 및 카메라 사용 이외의 별도의 제한은 없습니다. 단, 결선 장소로 가져오는데 어려움이 있는 크기는 피해 주시기 바랍니다. (전시부스 규격 : 2m × 2m, 변경될 수 있음)

2. 참가자격

가. 참가자격 : 대한민국 국민, 성인 이상 참가 가능

나. 팀 구성 : 참가 자격을 갖춘 자로 구성된 1~5인 (팀장 포함)

<팀 구성 규칙>

- 팀장 및 팀원은 모두 홈페이지에서 회원 가입을 완료해야 합니다.
- 팀장은 홈페이지에서 팀을 생성해야 합니다.
- 팀원은 팀에 가입 신청 하고, 팀장은 신청된 팀원 중 필요 인원을 승인해야 합니다.
- 승인된 팀원만 참가자로 인정됩니다. (팀원 간 소속 무관)
- 팀 구성 시 팀명은 아래 규칙을 준수해야 합니다.
 - 20Byte 이내 (한글 기준 10글자)
 - 띄어쓰기, 특수문자, 기호 사용 불가
 - 영문 사용 시, 대문자만 사용
 - 팀명 마지막에 '팀'을 붙이지 않는다. / ex) CONTEST (가능), CONTEST팀 (불가능)

<팀원 조정 규칙>

- 팀장 변경 시, 팀장이 홈페이지에서 팀장 변경을 신청해야 하며, 사무국 승인 하에 변경 가능합니다.
- 팀원 변경 시, 팀원이 홈페이지에서 팀 변경을 신청해야 하며, 팀장 승인 하에 변경 가능합니다.
- 예선을 통과한 결선진출팀 팀원의 변경은 결선 개발완료보고서 제출 전까지 가능합니다. (단, 개인적인 사정에 의한 일부 팀원의 중도 포기는 사무국 승인 시 가능)

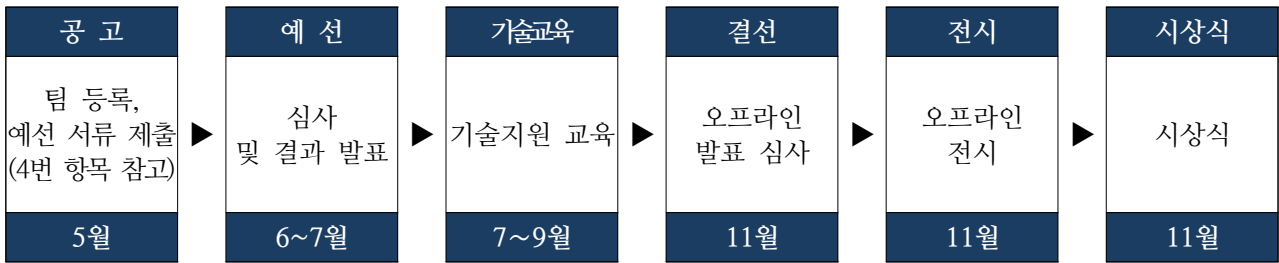
다. 참가자는 타 부문과 복수의 팀에 중복하여 참가할 수 있습니다.

(단, 예선 통과 시, 1개 부문만 선택 필요)

라. 참가등록방법 : 홈페이지>참가등록>참가안내를 참고하시기 바랍니다.

마. 경진대회 참가 등록은 규정 제3장(신청 및 접수)을 기본으로 합니다.

3. 경진대회 진행 일정



※ 상기 내용은 사정에 따라 조정될 수 있으며, 조정된 내용은 홈페이지를 통해 공지됨

- 가. 공 고 : 5월, 경진대회 홈페이지(eswcontest.or.kr)를 통해 공고
- 나. 예 선 : 6~7월, 접수된 제출서류(4번 항목 참고)를 토대로 후원 기업 실무자 및 해당 분야 전문가 서류심사 진행 및 결과발표
- 다. 기술교육 : 7~9월, 예선에 통과한 팀에 한하여, 분야별 기술 교육 및 컨설팅 진행(온·오프라인 교육 / 2회 내외)
- 라. 결선준비 : 10월, 개발완료보고서 및 동영상 등 결선심사를 위한 서류 제출
- 마. 결 선 : 11월, 접수된 제출서류(4번 항목 참고)를 토대로 오프라인 발표 평가 진행
- 바. 전 시 : 11월, 상위팀(장려상 이상) 오프라인 전시 진행(수상팀 반드시 참석)
 - * 전시 불참 팀은 수상이 취소될 수 있음
- 사. 시 상 식 : 11월, 시상식 진행 (수상 팀 반드시 참석)
 - * 전시 및 시상식은 2024 대한민국 산업기술 R&D대전 내 운영되며 사정에 따라 변동될 수 있음

4. 경진대회 심사 안내 및 제출 서류

구분	심사형태	제출서류	진행일정	비 고
예선	서류심사	참가신청서, 개발계획서	- 서류제출: ~6월7일 - 결과발표: 7월	온라인
결선	발표심사	개발완료보고서(PPT), 작품소개서, 소스코드, 최종 시연동영상, 개발작품(실물)	- 서류제출: 10월 - 결과발표: 11월	오프라인

※ 상기 내용은 사정에 따라 조정될 수 있으며, 조정된 내용은 홈페이지를 통해 공지됨

가. 심사형태 안내

- 서류심사 : 제출된 서류를 토대로 후원 기업 실무자 및 해당 분야 전문가로 구성된 심사위원들이 심사 진행
- 발표심사 : 현장에서 심사위원에게 작품에 대한 내용을 설명하고 답변하는 형태로 심사 진행(개발완료보고서 PPT로 발표 진행)

나. 제출서류

구분	제출 서류	분량	파일타입	파일명	업로드 위치
예선	참가신청서	1page	온라인	-	-
	개발계획서	10page 이내	PDF	2024ESWContest_webOS_팀번호_팀명_개발계획서	홈페이지
결선	개발완료보고서 (PPT)	20page 이내	PDF	2024ESWContest_webOS_팀번호_팀명_개발완료보고서	홈페이지
	작품소개서	2page	별도안내	2024ESWContest_webOS_팀번호_팀명_작품소개서	홈페이지
	소스코드	-	-	(GitHub 주소) github.com/사용자이름/2024ESWContest_webOS_팀명 또는 팀번호(자유) ※ 작품소개서에 해당 링크 삽입	GitHub
	최종 시연동영상	3분 이내	MP4	2024ESWContest_webOS_팀번호_팀명_최종시연동영상	YouTube
	개발 작품	최소 1개 이상	-	-	-

※ 상기 서류는 사정에 따라 조정될 수 있으며, 조정된 내용은 홈페이지를 통해 공지됨

※ 파일명 : 파란 부분만 해당 팀에서 수정하여 작성

※ 서류 제출 시 주최 측이 제시한 양식 및 기간을 준수하지 않을 경우 불이익(감점, 심사 대상 제외, 장비 지급 불가 등) 제공

※ 소스코드 및 동영상 관련 상세 내용은 “홈페이지 공지사항” 게시판 참고

- 참가신청서 : 홈페이지 상 온라인 신청(서류 접수 시 자동 생성)
- 개발계획서 : 개발 작품에 대한 계획서

- 개발완료보고서 : 개발 작품에 대한 전체적인 내용(PPT 형식)

No.	필수항목	분량
1	개발개요(프로젝트 개발 배경, 동기, 목표, 필요성 등) 소스코드 및 시연동영상 링크	3P
2	개발 환경 설명(HW, SW구성, 기능 설계도, 개발 환경 등)	10P
3	개발 프로그램 설명(파일 구성, 함수별 기능, 흐름도, 기술적 차별성)	
3-1	프로젝트에서 webOS를 활용한 부분 (별도 작성 또는 각 페이지 구분지어 표기)	
4	개발 중 발생한 장애요인과 해결방안	
5	개발 결과물의 차별성(작품의 차별성 및 우수성 설명, 타 유사 제품 및 서비스와의 차별점 설명)	5P
6	개발 결과물의 파급력 및 기대효과(판매가치, 시장성, 활용성, 발전 가능성 등)	
7	개발 일정 및 업무 분장	2P

- 소스코드 : 개발에 사용한 프로그램(언어) 소스코드
- 최종시연동영상(화질 : 720p) : 실제 작품의 시연 영상 + 작품 설명 영상
- 작품소개서 : 발표 심사 및 전시에서 보여줄 작품에 대한 소개 자료
- 개발 작품 : 발표 심사에 보여줄 실제 작품

다. 심사 항목 안내

구분	심사 항목	배점	항목별 설명	비고
예선	독창성 및 문제해결능력	30	- 현실에서 발생하는 다양한 문제점 분석 - 문제점 분석에 따른 독창적이며 현실적인 해결방안 제시	
	기술성	30	- 임베디드SW에 대한 기술 적응력 및 적용 예정 기술에 대한 이해도 (webOS에 대한 이해도 및 활용 계획 포함) - 제시된 아이디어를 구현하기 위한 기능적 차별성	
	문서완성도	30	- 개발 계획(개발의도, 개발방향, 기대효과)에 대한 명확성 - 개발 계획 문서의 가독성	
	팀 구성 및 팀 역량	10	- 개발 계획에 따른 적절한 팀 역할 배분 및 해당 팀원의 역량	
결선	완성도	60	- 개발 계획 목적에 맞는 기능의 구현 정도 및 완성도 (webOS플랫폼 활용정도, 개발계획서에 제시된 기술의 구현정도, 개발 중 발생한 장애요인 극복방안 적절성, 개발 일정 및 업무 분장의 적절성 등)	
	독창성	25	- 적용 기술의 차별성 및 우수성	
	활용성	15	- 작품의 기대효과, 실생활에서의 활용-사용성 및 추후발전가능성	
	가산점	각 5	- webOS 플랫폼에서 동작하는 하나 이상의 Node.js기반의 서비스를 개발 - AI/Interactive 기술 적용	

※ 상기 내용은 사정에 따라 조정될 수 있으며, 조정된 내용은 홈페이지를 통해 공지됨

5. 후원 기업 지원 내용(안) : 결선 진출 10팀

가. 기술 장비 지원 (후원 기업의 사정으로 변경될 수 있음)

- RaspberryPi 4 (with Power Adapter) 1대
- SD Card 16GB
- HDMI Cable
- V4L2 지원 카메라 1대
- ESP32 CAM 1대
- HDMI 지원 FHD 터치지원 디스플레이 (7인치) 1~2대
- 아두이노 R3, ESP32 또는 ESP8266 필요수량 만큼

* 대회 종료 또는 중도 포기 시, 장비 반납 / 분실 및 파손 시 동일 제품으로 반납

나. 기술 지원 세미나 및 기술 교육

- 장비 대여, webOS 기술 교육
- 진도 체크 및 질의응답

6. 소스코드 공개 관련 안내사항

가. 수상작은 참가팀과 사무국이 동일한 공개 여부 결정 권한을 가지며 Github를 통해 공개될 수 있습니다. (공개 정책에 대해서는 협의 가능하며 기업 후원 수상작은 후원기업에도 동일한 권한이 주어짐)

나. 본 대회에서 소스코드 공개의 의미는 (1)오픈소스로서의 공개와 (2)자작소스로서의 공개 중 하나이며, 참가팀의 동의 여부(의사 및 협의)에 따라서는 소스코드의 전체 또는 일부(선택적 또는 부분적으로)를 공개할 수 있고, (1)의 경우는 오픈소스로서 (제3자)의 활용을 제한해서는 안됩니다. (기업 후원 수상작은 후원기업의 요청에 따라 활용될 수 있음)

다. 위 내용과 같이 소스코드는 전부가 아닌 선택적 또는 부분적 공개가 가능하나 (1)의 경우는 오픈소스로서 반드시 핵심 부분이 공개범위에 포함되어야 합니다.

라. 단, 개발한 소스코드에 오픈소스가 포함되어 있을 경우 활용한 오픈소스 라이선스 규정을 위반해서는 안됩니다. (공개 대상인 수상작이 아닌 경우에도 오픈소스를 사용한 모든 참가팀에 해당)

마. 전 부문 수상작은 소스코드 관련하여 Github 공개범위를 Public(전체공개)로 유지합니다. (임의로 비공개 또는 Private으로 변경 시 수상 철회될 수 있음)

바. 제출된 소스코드는 경진대회 종료 후 홈페이지 수상작 게시판 內 개발완료보고서의 Github 링크를 통해 공개되도록 합니다.

7. 주의사항

- 가. 관련 안내 사항은 임베디드SW경진대회 공통 규정을 기반으로 합니다.
- 나. 접수 마감 이후, 팀 주제는 변경이 불가합니다.
- 다. 본 세부 안내사항 내용이 조정될 경우, 경진대회 홈페이지를 통해 안내될 수 있습니다.
- 라. 참가 팀은 공통 규정 제12조 4항에 해당하는 경우, 참가자격이 상실될 수 있습니다.
- 마. 중도 포기 시, 참가자는 발생하는 불이익을 감수해야 합니다.
- 바. 필요시 개인정보가 제3자에게 제공될 수 있습니다.

* 제공 시점 개별 공지 예정

〈중도 포기 시, 주의사항〉

- 포기 팀은 개발 장비를 사무국에 반납해야 합니다.
- 포기 팀은 반납확인증과 포기증서를 작성하여 제출해야 합니다.
- 포기 팀은 당해년도 또는 차년도 참가에 불이익이 있을 수 있습니다.

8. 예시

※ 아래는 스마트 팜 시스템에 관한 일부 예시이며, 이밖에 더욱 다양한 기술을 적용한 스마트 팜 시스템을 구축할 수 있습니다.

1. 자동제어 시스템

자동제어 시스템은 입력된 정보를 기반으로 다양한 장치와 시스템을 자동으로 제어하는 기술입니다. 이 시스템은 특히 스마트 팜에서 중요한 역할을 하며, 온도, 습도, CO2 농도 등의 환경 요인을 자동으로 조절하여 최적의 농작물 성장 환경을 유지할 수 있습니다.

사용 시나리오

1. 센서가 온도, 습도, 토양 수분, 광량 등을 실시간으로 감지합니다. (환경 모니터링)
2. 감지한 데이터를 중앙 제어 시스템으로 전송합니다. (데이터 전송)
3. 중앙 시스템은 데이터를 분석하고, 이상 징후를 감지하면 관련 장비(난방, 관수, 조명 시스템 등)를 자동으로 조절합니다. (분석 및 제어)
4. 관리자는 웹 앱을 통해 실시간 데이터를 확인하고 필요시 수동으로 개입할 수 있습니다. (사용자 피드백)

주요기능

- 환경 자동 조절: 기준치에 따라 온도, 습도, CO2 농도를 자동 조절합니다.
- 실시간 모니터링: 센서에서 측정된 데이터를 실시간으로 모니터링하고 사용자에게 제공합니다.
- 경보 시스템: 환경 요소가 임계값을 벗어날 경우 경보를 발생시켜 즉각적인 조치를 유도합니다.
- 데이터 분석 및 보고: 수집된 데이터를 분석하여 성장 패턴, 수확 시기 예측 등의 보고서를 생성합니다.
- 사용자 인터페이스: 사용자가 쉽게 시스템을 모니터링하고 제어할 수 있는 직관적인 인터페이스를 제공합니다.

2. 에너지 효율화 시스템

에너지 효율화 시스템은 스마트 팜의 전력 소비를 최적화하고 에너지 사용을 감소시키기 위해 설계된 자동 제어 시스템입니다. 이 시스템은 전력 사용 패턴을 분석하여 불필요한 에너지 낭비를 줄이고, 지속 가능한 농업 활동을 지원합니다.

사용 시나리오

1. 에너지 사용량을 측정하는 센서가 스마트 팜의 곳곳에 설치된 기기의 전력 소비 수준을 실시간으로 감지합니다. (에너지 소비 모니터링)
2. 수집된 에너지 사용 데이터는 중앙 처리 시스템으로 전송되어 분석됩니다. (데이터 분석)
3. 분석 결과를 바탕으로 불필요한 에너지 소비가 발생하는 장비는 자동으로 전력 공급이 조절되거나 꺼집니다. (효율적인 에너지 배분)
4. 시스템은 에너지 사용 보고서를 생성하고, 사용자는 이를 통해 더욱 효율적인 에너지 사용 계획을 수립할 수 있습니다. (사용자 결정 지원)

주요기능

- 실시간 에너지 모니터링: 모든 에너지 소비 포인트를 모니터링하여 실시간 데이터를 제공합니다.
- 에너지 소비 분석: 장기적인 데이터 분석을 통해 에너지 사용 패턴을 파악하고 이상 징후를 감지합니다.
- 자동 전력 조절: 에너지 소비가 높은 시간대에 비효율적인 장비의 전력을 자동으로 줄여 에너지 소비를 최적화합니다.
- 경고 시스템: 과도한 에너지 사용이 감지되면 사용자에게 알림을 보내 조치를 취할 수 있도록 합니다.
- 사용자 친화적 인터페이스: 에너지 사용 보고서, 현재 에너지 소비 상태, 조절 옵션을 쉽게 접근할 수 있는 사용자 인터페이스를 제공합니다.

[개발 방향]

webOS 플랫폼의 장점은 web Application과 web Service를 지원한다는 점입니다. Web page를 만들 듯 Application을 개발하고, Application이 동작하지 않아도 항상 동작하는 서비스를 Node.js 기반으로 개발할 수 있습니다. Node 기반 서버

스는 Application에 필요한 기능을 Application이 실행되지 않은 상태에서도 수행할 수 있습니다. 지속적으로 모니터링이 필요한 스마트 팜의 특성에 맞게 웹 앱이 동작하지 않는 상황에도 데이터를 측정할 수 있는 구조를 구축하여 다양한 방식으로 데이터를 활용하면 좋을 것 같습니다. webOS OSE 플랫폼이 제공하는 다양한 API를 활용하여 영상 정보 및 다양한 센서 정보를 기반으로 이를 활용하는 다양한 시나리오를 고민해보고 새로운 스마트 팜 운영자를 위한 Interactive UX/UI도 함께 고민하고 개발해 보기 바랍니다.

9. 참고자료

가. webOS OSE가 지원하는 주요 기능들

(<https://www.webosose.org/docs/reference/ls2-api/ls2-api-index/>)

○ Storage

- web(Browser) - LocalStorage (약 5Mb 정도)
- PDM - USB 저장장치 지원
 - <https://www.webosose.org/docs/reference/ls2-api/com-webos-service-pdm>
- Google Drive 지원
 - <https://www.webosose.org/docs/reference/ls2-api/com-webos-service-storageaccess/>

○ Database

- web(Browser) - Indexed DB
- DB8 - 로컬 데이터 베이스 제공
 - <https://www.webosose.org/docs/reference/ls2-api/com-webos-service-db>

○ Camera Support

- 라즈베리파이 카메라 / USB 카메라 지원
 - <https://www.webosose.org/docs/reference/ls2-api/com-webos-service-camera2/>
 - <https://www.webosose.org/docs/reference/ls2-api/com-webos-media/>

○ TTS

- webOS 플랫폼 API 활용
 - <https://www.webosose.org/docs/reference/ls2-api/com-webos-service-tts>
- Google Cloud Account 필요

○ Voice Assistant

- webOS 플랫폼 API 활용
 - <https://www.webosose.org/docs/reference/ls2-api/com-webos-service-ai-voice>
- Google Cloud Account 필요

○ Alarm / Scheduler

- <https://www.webosose.org/docs/reference/ls2-api/com-webos-service-alarm>
- <https://www.webosose.org/docs/reference/ls2-api/com-webos-service-activitymanager/>

○ Bluetooth

- <https://www.webosose.org/docs/reference/ls2-api/com-webos-service-bluetooth2>

○ Notification

- <https://www.webosose.org/docs/reference/ls2-api/com-webos-notification>

○ Media Indexing

- <https://www.webosose.org/docs/reference/ls2-api/com-webos-service-mediaindexer/>

나. 학습 자료

○ web application framework

- Enact: <https://enactjs.com>
- React: <https://reactjs.org>
- Vue: <https://vuejs.org>
- jQuery: <https://jquery.com>

○ web design Framework

- bootstrap: <http://getbootstrap.com>
- Bulma: <https://bulma.io>
- Material Design: <https://material.io/develop>

○ web Technology Study

- HTML5: <https://developer.mozilla.org/en-US/docs/Web/Guide/HTML/HTML5>
- CSS3: <https://developer.mozilla.org/en-US/docs/Web/CSS>
- JavaScript: <https://developer.mozilla.org/en-US/docs/Web/JavaScript>
- w3schools: <https://www.w3schools.com>
- 생활코딩: <https://opentutorialsorg/course/1>

○ Platform Study

- webOS OSE
 - 오픈소스 버전의 webOS 개발을 위한 개발자 사이트
 - <https://www.webosose.org>
- webOS TV
 - webOS TV 어플리케이션 개발을 위한 개발자 사이트
 - OSE 버전의 API는 다를 수 있으므로 필요한 개념만 확인/학습
 - <http://webostv.developer.lge.com>

○ Native HMI Framework (필요시)

- QML Application 개발
 - <https://www.webosose.org/docs/guides/development/qml-apps/qml-app-overview>

○ Service Component

- JavaScript Service
 - Node.js 기반의 서비스를 개발할 수 있음(Node.js / JavaScript 개발 필)
 - Node.js 개발자 사이트 : <https://nodejs.org/en>
 - 플랫폼 빌드 없이도 서비스 추가 가능
 - <https://www.webosose.org/docs/guides/development/js-services/js-service-overview>

○ webOS web app & service 가이드 (한글)

- 한글 가이드
 - <https://webos-supporters.tistory.com/>