|  |
| --- |
| 1. **주제**   일반인과 대학생 프로그래머를 위한 아이디어 공유 웹/앱 개발 제안  **분반, 팀, 학번, 이름**  오픈소스기초설계 (나)반, 2팀, 20230734 변지현 |

|  |  |
| --- | --- |
| **2. 요약**  아이디어에 늘 목마른 대학생 프로그래머와 좋은 아이디어를 가지고 있으나 구현에 이르지 못하는 비전공자(일반인)의 니즈를 모두 충족하는 것을 목표로 하는 ‘아이디어 공유 플랫폼’입니다.  일반인 사용자가 일상생활 속에서 떠오른 좋은 아이디어를 플랫폼 상에 업로드 합니다. 이후 해당 아이디어에 관심이 있는 프로그래머 사용자가 자신의 의사를 댓글로 남긴 후, 아이디어를 구현하여 결과물을 업로드하는 방식으로 작동합니다.  일반인들에게는 본인의 아이디어가 실현으로 이루어질 수 있다는 성취감을 안겨줄 수 있고, 프로그래머들에게는 포트폴리오를 풍성하게 할 수 있는 기회가 될 수 있습니다. | **3. 대표 그림** |

|  |
| --- |
| **4. 서론**  **배경설명**  오픈소스 프로젝트 팀이 결성되자마자, 팀원들 각자 주제를 구상하여 공유하는 방향으로 진행하였습니다. 팀원들 모두 4주간 열심히 고민했지만, 결국 좋은 아이디어를 찾지 못해서 난항을 겪었습니다. 하지만 놀랍게도 IT 전공과 거리가 먼, 다른 학과 친구들이 일생생활 속에서 좋은 아이디어를 많이 생각해내는 모습을 보게 되었습니다. 예를 들어, ‘카페의 일회용 컵이 길에 버려져 있는 모습을 보고, 지도 어플에 공공 쓰레기통 위치도 표기해 주었으면 좋겠다’와 같은 아이디어를 떠올렸습니다. 그러나 여기서 도출된 좋은 아이디어들이 그저 생각하는 단계에서 끝나고 버려진다는 사실이 아쉽게 느껴졌습니다. 따라서 이러한 문제를 해결할 수 있도록, 아이디어에 늘 목마른 대학생 프로그래머들과 좋은 아이디어를 가지고 있으나 구현하지 못하는 일반인의 니즈를 모두 충족할 수 있는 아이디어 공유 웹/앱을 기획하게 되었습니다.  **사례분석**  아이디어를 공유하는 플랫폼은 이미 많이 존재합니다. 가장 대표적으로 개발자들의 커뮤니티인 Github, 프로젝트 아이디어를 제안하고, 사용자들에게 후원을 받는 Kickstarter, 그리고 대한민국 특허청에서 운영하는 아이디어로가 있습니다. 그러나 Github와 Kickstarter의 경우에는 프로그래밍 지식을 어느정도 갖춘 프로그래머들을 위한 플랫폼입니다. 프로그래밍 지식이 전무한 일반인들이 접근하기에는 어렵습니다. 또한 특허청에서 운영하는 아이디어로의 경우에는, 개인의 아이디어를 기업에서 경매를 통해 구입하는 방식입니다. 경매 시 거래 금액이 매우 크기 때문에, 기업에게 꼭 필요한 아이디어가 아닌 경우 버려지게 됩니다. 이러한 일종의 장벽들은 일반인들이 아이디어를 공유하기 어렵게 만들고, 결국 좋은 아이디어가 세상에 나오는 것을 억제합니다.  **문제정의**  위에서 기술했듯이, 기존 플랫폼에서 가장 큰 문제점은 어느정도 프로그래밍 지식을 갖춘 사용자 위주이기 때문에 일반인의 접근성이 떨어진다는 것입니다. 이는 아이디어 공유의 위축으로 이어집니다. 또한 아이디어 공유 플랫폼을 제작하게 되면, 부가적으로 두가지의 문제가 발생합니다. 첫번째로, 해당 플랫폼에서 제작된 결과물에 대한 라이선스 문제입니다. 아이디어 공유자와 프로그래머 사이에 분쟁이 발생할 수 있기 때문입니다. 두번째로는, 아이디어의 실현 가능성 문제입니다. 프로그래밍 지식이 전무한 일반인이 구현 단계를 고려하지 않고 낸 아이디어가 과연 실현 가능한가에 대한 의문이 들 수 있기 때문입니다.  **극복방안**  일반인의 접근성 문제를 해결하기 위해, 기존 플랫폼의 전문적인 분위기 보다는 자유롭게 아이디어를 공유할 수 있는 분위기를 조성할 예정입니다. 또한 누구나 쉽게 플랫폼을 이용할 수 있도록 사용자 친화적 UI를 제공하기 위해 본문과 댓글의 간단한 형식으로 플랫폼 구조를 형성하려고 합니다. 일상생활 속에서 짧은 시간 내에 아이디어를 업로드 할 수 있도록 업로드 절차를 간소화할 예정입니다. 더불어 아이디어 공유와 구현에 대해 인센티브를 줄 수 있도록 마일리지 및 보상 시스템을 구현하려 합니다. 두번째로 라이선스 문제의 경우에는, 수업 때 배웠던 오픈소스 라이선스를 활용하려고 합니다. 일반인 사용자에게 오픈 소스 라이선스에 대한 설명을 제공하고, 직접 명시할 수 있도록 합니다. 또한 해당 플랫폼에서 창출된 결과물은 오픈소스로 제한할 예정입니다. 마지막으로, 아이디어 실현 가능성에 대한 문제는 가이드라인을 통해 해결할 예정입니다. 아이디어 업로드 창에 구현 과정에서 어떤 것들을 고려해야 하는지 가이드라인을 제공하려고 합니다. |

|  |
| --- |
| **5. 본론**  **시스템 개요 그림**  **필요한 기술 요소**  시스템 개요도의 프로그램 동작 부분에 기술했듯이, 사용자 PC로부터의 요청을 전송하는 프론트엔드, 요청에 대한 응답을 반환하는 백엔드, 그리고 데이터를 저장하고 관리할 데이터베이스(DB)가 필요합니다. 또한 버전 관리 및 협업을 위해 Git, Github를 활용해야 하고, 하이브리드 앱 제작시 https 보안 인증서가 필요합니다. 더불어 구현 후 코드를 검사할 Jest, Mocha와 같은 테스트 도구가 필요합니다.  **플랫폼 동작 과정**   1. 아이디어 작성 및 등록: 아이디어를 공유하고자 하는 사용자가 플랫폼(웹/앱) 내에서 아이디어를 작성합니다. 아이디어 작성 시 구현 과정을 고려할 수 있도록 가이드라인을 참고합니다. 2. 아이디어 탐색: 게시판을 탐색하던 프로그래머 사용자가 특정 아이디어에 흥미를 느끼면, 자신이 구현하겠다는 의사를 댓글 형식으로 남깁니다. 3. 프로그램 구현: 프로그래머 사용자는 선택한 아이디어를 바탕으로 구현을 시작합니다. 4. 결과물 업로드: 프로그래머 사용자가 구현을 마치면, 이전에 작성했던 댓글에 이어서 결과물을 업로드합니다. 5. 결과물 확인 및 활용: 아이디어를 제공한 사용자는 업로드 된 결과물을 확인하고, 이를 일상생활 속에서 활용합니다. 6. 보상 지급: 아이디어가 채택되거나 프로그램 구현을 완료한 경우, 사용자는 그에 따른 보상을 받습니다.   따라서 플랫폼의 최종적인 목표는 일상생활 속에서 떠오른 유용한 아이디어가 구현까지 이루어질 수 있도록 일반인 사용자와 프로그래머 사용자를 중개하는 것입니다.  **구현 방법 및 개발 방향**  사용자들의 접근성과 개발 비용을 고려하여 하이브리드 앱으로 제작할 예정입니다. 하이브리드 앱이란 웹 프로그래밍을 통해 개발된 애플리케이션으로, 웹 구현만으로도 사용자에게 웹과 앱 두 가지 형태를 모두 제공할 수 있습니다. 실질적인 동작은 웹에서 이루어집니다.   1. **프론트엔드**    1. 아이디어 업로드 기능을 비롯하여 사용자로부터 텍스트를 입력 받기 위해 <form>, <input>, <textarea>등의 태그 묶음으로 이루어진 html 파일을 구현합니다.    2. CSS로 UI를 스타일링하여 사용자 친화적인 인터페이스를 갖추도록 합니다. 또한 오픈소스 라이브러리인 Bootstrap을 활용하여 반응형 디자인을 구현합니다.    3. JavaScript로 웹 페이지 내 발생한 이벤트 처리하고, 동적 웹 페이지를 구현합니다. 사용자가 실시간으로 자신의 등록 내용을 확인하고, 다른 사용자들의 아이디어 및 결과물도 즉시 확인할 수 있도록 합니다. 2. **백엔드**    1. 웹페이지 사용자(클라이언트)로부터 발생한 요청을 처리할 백엔드는 Node.js를 활용할 예정입니다.    2. 플랫폼 내의 댓글 작성, 아이디어 및 구현 결과 업로드 등에서 실시간 통신이 필요하므로 Node.js 내의 Socket.IO 라이브러리를 사용할 예정입니다.    3. 구현 결과 등의 파일 업로드는 Multer 라이브러리를 이용하려고 합니다. 3. **데이터베이스**    1. 현재 프로토타입을 제작하는 것이 목표이며, 데이터 구조가 안정적이기 때문에 SQLite를 이용할 예정입니다.   개발 순서의 경우 프론트엔드 개발, 백엔드 개발, 데이터 베이스 연결, 보안 설정(https), 테스트, 배포 순으로 진행할 예정입니다. |

|  |
| --- |
| **6. 결론**  기존의 아이디어 공유 플랫폼은 프로그래밍 지식을 가진 개발자들을 위한 공간입니다. 따라서 일반인의 경우 접근하기 어렵고, 좋은 아이디어가 있어도 구현 단계까지 이어지기 어렵습니다. 반면에, 대학생 프로그래머들은 늘 새로운 프로젝트 주제를 찾느라 고민합니다. 따라서 이 둘의 니즈를 충족할 수 있는 플랫폼을 제안하고자 합니다. 팀원들과 프론트, 백, DB 3개의 파트로 역할을 분담하고, UI 디자인에 들어가는 것으로 프로젝트를 시작하려고 합니다. |

**7. 출처**

[1] 고경희, 『Do it! 지옥에서 온 문서 관리자 깃 & 깃허브 입문 [전면 개정판] 』, 이지스퍼블리싱, 2022, 137 ~ 141쪽.

[2] 제임스 웹 영, 『아이디어 생산법』, 윌북, 2018, 17~24쪽.

[3] Node.js v22.9.0 documentation - HTTPS version, <https://nodejs.org/docs/latest/api/https.html>, (2024.10.14 접속)

[4] W3schools – HTML Input Attributes, <https://www.w3schools.com/html/html_form_attributes.asp>, (2024.10.15 접속)

[5] Bootstrap v5.3 documentation, <https://getbootstrap.com/docs/5.3/getting-started/introduction/>, (2024.10.15 접속)

**\***저희 2팀의 경우, 하나의 프로젝트에 대해 파트를 나누어 개별 보고서를 작성하였습니다. 따라서 큰 줄기는 같습니다. 팀장인 제가 아이디어 공유 플랫폼의 메인 메뉴이자 전체적인 동작 방식을 기술하였고, 나머지 2명의 팀원이 플랫폼 내의 다른 주요한 기능들에 집중하여 작성하였습니다.