|  |
| --- |
| **1. 주제**  지도 기반 캠퍼스 유기동물 돌봄 관리 시스템 구축  **분반, 팀, 학번, 이름**  (나)분반, 12팀, 20251903, 정구현 |

|  |  |
| --- | --- |
| **2. 요약**  본 프로젝트는 대학 캠퍼스 내 유기동물에 대한 자발적인 돌봄 활동을 체계화하고 효율성을 극대화하기 위해 ‘지도 기반 캠퍼스 유기동물 돌봄 관리 시스템’을 구축하는 것을 목표로 함.  시스템의 핵심 내용은 실시간으로 업데이트되는 캠퍼스 지도를 중심으로 구성됨. 사용자는 지도 위에 아이콘으로 표시된 각 동물의 위치를 직관적으로 파악할 수 있음. 각 아이콘을 통해 해당 동물의 상세 정보(사진, 특징, 주요 출몰 위치) 등을 조회하고 ‘구조요청’, ‘사료 급여’ 등과 같이 돌봄 활동을 기록 및 다른 사용자들과 공유할 수 있음.  본 시스템 구축을 통해 유기동물 및 길거리동물들의 안전을 확보하고 캠퍼스를 보다 깨끗하게 유지할 수 있음. | **3. 대표 그림**    그림 1.  매일 보이던 길고양이의 부재로 인한 학생들의 걱정  그림 2.  지도 기반 캠퍼스 유기동물 돌봄 관리 시스템을 통해 한결 쾌적해진 환경 |

|  |
| --- |
| **4. 서론**  현재 숭실대학교 캠퍼스 내에는 ‘숭냥이’라고 불리는 길고양이들이 다수 존재한다. 많은 학생들이 자발적으로 숭냥이들에게 먹이를 챙겨주고 따뜻한 관심을 가지며 이 작은 생명체들과 공존을 실천하고 있다. 이처럼 숭냥이는 단순한 동물을 넘어 삭막할 수 있는 캠퍼스 생활에 활력과 따뜻함을 더해주는 소중한 구성원으로 자리 잡음.  하지만 이러한 긍정적인 교감의 이면에는 체계적인 돌봄 시스템의 부재라는 그림자가 존재함.매일 아침 인사를 건네던 숭냥이가 갑자기 보이지 않을 때 학생들은 그저 막연한 걱정을 안고 소식을 기다릴 수밖에 없음. 제작년에 학생들의 소통창구인 ‘에브리타임’에 게시되었던 교통사고로 다리를 다친 숭냥이의 소식을 애타게 찾던 한 학생은 이러한 문제점을 단적으로 보여주는 사례임. (그림 1 참조) 임시보호자의 연락처를 모르고 현재 상태를 알 길이 없어 오프라인 게시물을 붙이고 온라인으로도 글을 작성해 익명의 제보에만 의존해야만 했던 이 사건은 선한 의지를 가진 다수의 개인이 존재함에도 불구하고 그 정보가 연결되지 못하는 현재 돌봄 방식의 명확한 한계를 드러냄.  이러한 배경과 사례를 통해 현재 캠퍼스 내 길고양이 돌봄 활동의 문제점을 다음과 같이 정의할 수 있음. 첫번째로 정보 전달의 한계와 파편화가 있는데 고양이의 건강 상태나 위치, 사고 소식 등 중요한 정보가 특정 개인이나 소수 집단에 머물러 있거나 대자보와 같은 오프라인 수단에 의존하여 전파 속도가 느리고 범위가 제한적임. 이는 공동체 전체의 신속하고 일관된 대응을 어렵게 만듬. 둘째로 돌봄 활동의 비효율과 자원 낭비, 누가 언제 어디서 먹이를 주었는지에 대한 정보가 공유되지 않아 특정 고양이가 중복으로 과식을 하거나 다른 고양이는 굶는 등 비효율적인 상황이 발생함. 이는 학생들의 선의와 자원이 낭비되는 결과로 이어짐. 마지막으로 위기 상황 발생 시 신속한 대응 체계 부재가 가장 중요함. '숭냥이(치즈냥이)'의 사례처럼 고양이가 다치거나 아픈 위기 상황이 발생했을 때 이를 발견한 사람이 효과적으로 도움을 요청하고 공동체의 지원을 이끌어낼 공식적인 소통 창구가 없음. 이는 응급 상황의 골든타임을 놓치는 치명적인 결과로 이어질 수 있음.  따라서 본 프로젝트는 위에서 정의된 문제들을 기술적으로 해결하기 위해 '지도 기반 캠퍼스 유기동물 돌봄 관리 시스템'의 구축을 제안함. 이 시스템은 캠퍼스 맵 위에 숭냥이들의 프로필과 현재 위치를 시각적으로 표시하고, 모든 구성원이 각 개체의 상태를 실시간으로 기록하고 공유할 수 있는 통합 플랫폼을 기획중임.  이를 통해 파편화된 정보를 한곳에 모아 정보의 비대칭을 해소하고 중복 돌봄을 방지하여 자원의 효율성을 높일 것임. 또한 위기 상황 발생 시 앱을 통한 긴급 알림 기능으로 공동체의 신속한 대응을 유도하여 소중한 생명을 구할 기회를 확보하고자 함. 궁극적으로 본 시스템은 학생들의 자발적인 돌봄 활동을 체계적으로 지원하여 인간과 동물이 더욱 안전하고 행복하게 공존하는 캠퍼스 문화를 만들어 나가는 핵심적인 도구가 되기를 기대함. |

|  |
| --- |
| **5. 본론**  본 프로젝트가 제안하는 시스템은 사용자인 학생과 관리자가 웹 및 모바일 애플리케이션을 통해 서버와 상호작용하는 클라이언트-서버 아키텍처를 기반으로 할 예정임. 사용자는 개인 스마트폰이나 PC를 통해 캠퍼스 지도에 시각화된 고양이 정보를 확인하고 자신의 돌봄 활동을 실시간으로 기록할 수 있습니다. 모든 데이터는 중앙 서버의 데이터베이스에 저장되어 공동체 구성원 전체에게 투명하게 공유됨. 또한 관리자는 별도의 관리자 페이지를 통해 동물의 프로필을 공식 등록하거나 위기 상황 시 전체 공지를 발송하는 등 체계적인 운영을 담당함.  시스템 개요 및 필요한 기술 요소    **프론트엔드**: 웹 페이지의 뼈대, 디자인, 동작을 구현하는 가장 기본적인 핵심 기술. 별도의 복잡한 프레임워크 학습 없이 웹 개발의 기초를 탄탄히 다지면서 프로젝트의 모든 핵심 기능을 충분히 구현할 수 있음.  **백엔드 및 데이터베이스**; Google Firebase: 서버를 직접 구축하고 관리하는 어려운 과정 없이 백엔드 기능을 빌려 쓸 수 있는 서비스를 이용 예정  **Firestore**: 실시간 데이터베이스로 고양이 정보나 활동 기록을 저장하고 모든 사용자가 즉시 업데이트된 내용을 볼 수 있음  **Firebase Hosting**: 완성된 웹사이트를 명령어 한 줄로 인터넷에 쉽게 배포 및 게시할 수 있게 해주는 서비스  구현 방법 및 개발 방향; 프로젝트는 애자일(Agile) 방식에 기반하여 총 3단계의 점진적 개발 로드맵을 설정했음. 핵심 기능부터 빠르게 구현하고 사용자 피드백을 반영하여 서비스를 고도화하는 것을 목표로 함.  1단계: MVP(최소 기능 제품) 개발로 핵심 기능의 신속한 구현 및 사용성 검증.  주요 기능: HTML/CSS/JS를 이용한 기본 UI 구현, Firebase 연동을 통한 회원가입/로그인, 숭냥이 프로필 등록/조회, 지도 기반 위치 확인, 간단한 돌봄 활동 기록 및 타임라인 기능. Firebase Hosting을 통해 웹사이트를 빠르게 배포하고 초기 사용자 피드백 수집.  2단계: 기능 고도화 및 커뮤니티 활성화로 사용자 편의성 증대 및 커뮤니티 기능 강화.  주요 기능: Firestore의 실시간 업데이트 기능을 활용한 댓글 및 사진 공유 기능 강화, 익명 정보 공유 게시판 개발, 특정 조건 발생 시 알림 기능 구현.  3단계: 서비스 안정화 및 데이터 활용으로 서비스 안정성 확보 및 데이터 시각화.  주요 기능: 사용자 피드백을 바탕으로 UI/UX 개선 및 버그 수정. Firestore에 축적된 데이터를 바탕으로 간단한 통계를 보여주는 관리자용 페이지 개발.  전체 개발 과정은 Git 및 GitHub를 활용하여 체계적인 버전 관리 및 협업을 수행할 예정 |

|  |
| --- |
| **6. 결론**  본 제안서는 숭실대학교 캠퍼스 내 '숭냥이' 돌봄 활동의 비효율성과 정보 공유의 한계를 해결하고자 지도 기반 유기동물 돌봄 관리 시스템 구축을 제안하였음. HTML/CSS/JS 프론트엔드와 Firebase 백엔드, Kakao Maps API 연동을 통해 실시간 정보 공유 및 체계적인 관리를 목표하며 이는 궁극적으로 캠퍼스 동물의 안전을 보장하고 학생들의 협력적 공존 문화를 조성하는 데 기여할 것임.  향후 프로젝트는 Git/GitHub를 활용한 개발 환경 구축 후 Firebase 기반의 회원가입, 숭냥이 프로필/위치 조회, 활동 기록 등의 핵심 기능으로 MVP를 우선 개발할 계획임. 이후 사용자 피드백을 반영하여 실시간 알림, 커뮤니티 기능 등을 추가하며 시스템을 고도화하고, 최종적으로 안정성 테스트를 거쳐 결과물을 발표할 예정 |

**7. 출처**

[1] 에브리타임, 익명게시판, ‘치즈 숭냥이 근황’, 2023.07.28 <https://everytime.kr/370455/v/310944720>

[2] Google. (n.d.). Firebase Documentation. Retrieved from <https://firebase.google.com/docs>

[3] MDN Web Docs. (n.d.). HTML: HyperText Markup Language. Retrieved from <https://developer.mozilla.org/ko/docs/Web/HTML>

[4] MDN Web Docs. (n.d.). *CSS: Cascading Style Sheets*. Retrieved from <https://developer.mozilla.org/ko/docs/Web/CSS>

[5] MDN Web Docs. (n.d.). JavaScript: JavaScript and web programming. Retrieved from <https://developer.mozilla.org/ko/docs/Web/JavaScript>