|  |
| --- |
| **1. 주제**  과외 및 멘토링 튜터를 위한 학습 관리 어플  **분반, 팀, 학번, 이름**  가반, 4팀, 20241978, 임수빈 |

|  |  |
| --- | --- |
| **2. 요약 (10점)**  - 목표 : 과외 및 멘토링 튜터들이 수업을 보다 효율적으로 관리하고 학습 성과를 극대화할 수 있도록 돕는 어플리케이션을 개발한다. 수업 시간표 관리, 출석 체크, 학생 집중도 확인, 자료 공유, 학부모에게 피드백을 전송하는 기능을 제공하는 것을 목표로 한다.  - 핵심 내용 : 튜터와 학생이 수업 시간을 등록하고 오프라인 및 온라인 수업 방식을 선택할 수 있다. 온라인 수업의 경우 줌(ZOOM)와 같은 플랫폼을 연동한 뒤 화면 실시간 체크를 통해, 오프라인 수업의 경우 전면 카메라를 통해 학생의 집중도를 모니터링한다. 튜터 간의 자료 공유 기능과 학부모에게 출석 및 피드백을 전송하는 기능이 포함된다.  - 중요성 : 이 어플리케이션은 튜터들의 수업 준비 및 학생 관리를 간소화하고, 학부모와의 소통을 통해 학생의 출석 및 학습 상황을 투명하게 관리할 수 있도록 하여 학습 성과를 높일 수 있다. | **3. 대표 그림**  - 개발 배경 : 과외 및 멘토링을 진행하는 튜터들이 학생의 집중도를 관리하거나 학부모에게 피드백을 전송하는 등의 과정에서 편리성을 도모하기 위한 플랫폼의 필요성을 느꼈다.  - 예상 결과 : 어플을 통해 이러한 문제를 해결하여 시간표 관리, 집중도 체크, 피드백 전송이 더 효율적으로 이루어질 것이다.    그림 1. 앱 내 UI |

|  |
| --- |
| **4. 서론**  - 배경 설명, 사례 분석 : 현재 교육 환경에서 과외와 멘토링의 비율이 높아지고 있는데, 학생 개개인의 학습 성향을 반영한 맞춤형 교육이 강조되고 있으며, 이에 따라 튜터들이 학생들의 학습 진행을 보다 세밀하게 관리할 수 있는 도구의 필요성이 대두되고 있다. 하지만 현재 교육 어플리케이션을 살펴보았을 때 학생의 학습을 도와주거나 과외 학생을 연결해주는 등의 어플리케이션밖에 찾지 못했고, 학생의 학습 상황에 대한 실시간 피드백 제공이나, 수업 중 학생의 집중도를 확인하는 등의 기능이 부족하다. 실제로 필자 본인과 주변인들의 사례를 살펴보면 학생의 집중도를 매번 파악하는 데 어려움을 겪은 사례도 있고 체계적으로 과외와 멘토링을 관리하기 어려워하는 사례도 있었다.  - 문제 정의 : 오프라인 수업에서는 학생과 튜터가 같은 공간에 있지만 수업을 하는 동시에 집중도를 일일이 파악하여 관리하기에 한계가 있다. 온라인 수업에서는 다른 공간에 있기 때문에 더욱더 어렵다. 학부모와의 소통에서도 체계적인 시스템이 없는 경우 피드백을 전달하는 데 불편함이 있다. 플랫폼을 통한 과외가 아닌 개인적인 과외 같은 경우에는 자료를 구하기 힘든 경우가 있는데, 비슷한 주제나 문제를 다루는 튜터들이 공유할 수 있는 기회가 적다보니 동일한 내용을 여러 번 준비해야 하거나, 수업 자료의 질에서 편차가 생기는 경우가 있다.  - 극복 방안 : 이를 극복하기 위해 우선 집중도 모니터링 기능을 도입한다. 오프라인 수업에서는 휴대폰의 전면 카메라를 통해 학생의 집중도를 실시간으로 모니터링하고, 온라인 수업에서는 화상 회의 플랫폼을 제공한 뒤 화면 감지를 통해 학생의 집중도를 모니터링한다. 학생이 집중하고 있지 않을 때 경고창을 띄우고 집중하지 않은 횟수를 체크하여 저장한 뒤 피드백 기능에서 저장된 내용을 확인하여 피드백 시 참고할 수 있게 한다. 또 자료 공유 및 협업 시스템을 도입한다. 튜터들이 수업 자료를 손쉽게 공유하고 피드백을 주고 받을 수 있는 공간을 제공한다. 이를 통해서 자료 준비에 소요되는 시간을 줄이고, 수업의 질을 상향 평준화할 수 있다. 학부모와의 피드백 소통 기능도 도입한다. 학생의 출석 여부 및 학습 상황을 수업마다 학부모에게 자동으로 전송함으로써 학습 상황에 대한 투명한 정보를 제공할 수 있다. 출석 여부 및 간단한 학습 상황 외에도 필요한 피드백을 전송할 수 있는 기능도 부가적으로 존재한다. |

|  |
| --- |
| **5. 본론**    그림 2. 시스템 개요 그림  <필요한 기술 요소 설명>  1. 집중도 모니터링 기술 : OpenCV 및 TensorFlow Lite 학생의 얼굴 인식과 표정 분석을 통해 실시간으로 집중도를 측정하고 분석한다.  2. 카테고리별 자료 공유 및 협업 시스템 : Google Cloud Storage 또는 AWS S3를 사용하여 대용량 자료를 안전하게 저장하고 공유한다. 파일 메타데이터는 Firebase Firestore나 Google Cloud Firestore에 저장된다.  3. 학부모와의 피드백 소통 시스템 : Firebase Realtime Database 또는 Firestore를 통해 학생의 출석 및 집중도 데이터를 기록하고, Firebase Functions로 자동 피드백을 생성하여 Firebase Cloud Messaging 또는 Email API로 알림을 보낸다.  4. 시간표 관리 시스템 : Google Calendar API 또는 Firebase를 사용하여 학생별로 색상 코딩된 일정표를 제공하고, 일정 충돌 시 Firebase Cloud Messaging을 사용하여 실시간 알림을 제공한다.  <구현 방법 및 개발 방향>  - Flutter 프레임워크를 사용해 크로스 플랫폼 어플리케이션을 개발하고, Provider 패턴을 통해 상태 관리를 간소화하여 수업 자료 관리, 집중도 모니터링, 화상 수업 기능을 효율적으로 제공한다. Node.js와 Express를 사용하여 서버를 구축하고, 클라우드 스토리지와 Firebase와의 연동을 통해 데이터를 처리한다. 이를 통해 튜터가 업로드한 수업 자료를 안전하게 저장하고, 학부모 피드백 전송을 위한 데이터를 실시간으로 관리할 수 있다.  1. 집중도 모니터링 기술 구현 : OpenCV와 TensorFlow Lite를 사용해 오프라인 수업에서는 얼굴 인식 및 표정 변화를 실시간 분석하고, 온라인 수업에서는 Zoom SDK나 Jitsi Meet API로 화면 참여 여부를 모니터링한다. 집중도가 떨어지면 알림을 보내준다.  2. 카테고리별 자료 공유 및 협업 시스템 구현 : 튜터가 원하는 자료를 검색할 때, Firebase Realtime Database 또는 Firestore에서 제공하는 메타데이터를 기반으로 카테고리별로 자료를 필터링할 수 있다. 각 자료에는 파일명, 업로드 날짜, 업로더 정보 등이 함께 표시되어, 필요한 자료를 선택해 Google Cloud Storage 또는 AWS S3에서 해당 파일을 다운로드할 수 있다.  3. 학부모와의 피드백 소통 시스템 구현 : 기록된 출석 정보 및 집중도 데이터를 기반으로 Firebase Functions를 통해 자동으로 피드백을 생성하는 시스템을 구현한다. 학생이 수업 중 집중하지 않았던 시간을 자동으로 분석해 피드백에 포함시키며, 출석 여부와 학습 성과도 함께 기록한다. 생성된 피드백은 Firebase Cloud Messaging 또는 Email API를 통해 학부모에게 푸시 알림 또는 이메일로 자동 전송된다.  4. 시간표 관리 시스템 구현 : Google Calendar API 또는 Firebase를 사용하여 튜터가 여러 학생의 수업 일정을 하나의 시간표에서 관리할 수 있도록 설계한다. 각 학생의 수업은 서로 다른 색상으로 구분되며, 튜터가 한눈에 모든 학생의 일정을 쉽게 파악할 수 있도록 한다. Firebase를 통해 시간표 데이터를 실시간으로 동기화하고, 일정이 중복되거나 충돌할 경우 Firebase Cloud Messaging를 통해 이를 즉시 감시하여 튜터에게 자동으로 알림을 보내어 조정할 수 있다. |

|  |
| --- |
| **6. 결론**  본 제안서에서 제안한 어플리케이션의 핵심 기능은 학생 집중도 모니터링, 카테고리별 자료 공 유, 학부모 피드백 자동화, 학생별 색상 코딩된 시간표 관리 시스템이다. 각 기능은 Flutter 프레임워크 기반으로 구현될 것이고, Google Cloud, Firebase와 같은 기술을 통해 안정적인 데이터 관리 및 실시간 동기화가 가능하도록 설계하였다. 이러한 기능을 통해 튜터들은 수업 운영의 효율성을 높이고, 학생들의 학습 상황을 더 쉽게 파악할 수 있게 될 것이다.  개발을 위해 우선 각 기능별 개발 및 테스트를 수행하고 이후 사용자 피드백을 반영하여 기능 개선 및 최적화를 진행할 것이다. 향후에는 사용자 요구에 맞게 새로운 기능을 추가하고 학습 성과를 분석할 수 있는 추가 모듈 개발을 검토할 계획이다. |

**7. 출처**

1. SCIENCEON <[한국특허] 눈동자 인식 프레임워크 장치 및 그 구현 방법>

2. ZOOM SDK, <https://developers.zoom.us/docs/meeting-sdk/>

3. Firebase, <https://firebase.google.com/?hl=ko>

4. Google Cloud, <https://cloud.google.com/storage>