과제 제출 시스템

Team CtrlZ

|  |  |  |
| --- | --- | --- |
| 담당 교수님 |  | 최윤자 교수님 |
| 팀장 |  | 유동규 2011105062 |
| 팀원 |  | 박동원 2011105038 |
| 팀원 |  | 윤준호 2011105065 |
| 팀원 |  | 이승진 2011105072 |

**INDEX**

1. Overview
2. Requirements
3. Subjects (\*tentative)
4. Task Plan and Distribution
5. Strategies
6. Special (\*tentative)
7. **Overview**

과제 제출 시스템은 기존 ABEEK사이트를 이용한 과제 제출시 부족했던 보안 및 데이터 관리 등을 보완하기 위해 계획 및 제작되었다. 그러나 다음과 같은 개선사항들이 발견되었다.

1. 현재까지 개발 된 과제 제출 시스템은 네트워크 연동과 보안에 관한 기능이 구현되어있지 않다.
2. 과제 게시물의 등록 및 채점 시에 사용자들이 그 사실을 확인할 수 있는 알림 기능이 필요하다.
3. 게시물의 수정 및 삭제 기능이 존재하지 않는 데 따른 불편함이 있다.
4. 설계 측면에 있어 MVC 패턴을 적용하여 유지보수 측면에서의 용이성을 끌어올릴 필요가 있다.

이번 프로젝트는 위에서 언급된 부분들의 추가 구현 및 개선에 대한 계획이 포함되어있다.

1. **Requirements**

|  |  |  |  |
| --- | --- | --- | --- |
|  | 기존  요구사항 |  | 1. 교수, 조교, 학생이 시스템을 사용한다. 2. 교수, 조교는 과제를 등록한다. 3. 학생은 교수, 조교가 올린 과제를 확인하고, 제출한다. 4. 교수, 조교는 학생이 제출한 과제를 확인하고 평가 및 코멘트를 단다. 5. 교수, 조교가 과제를 평가하면 학생에게 알림이 간다. 6. 학생은 교수, 조교에게 질문이 가능하다. 7. 교수, 조교는 학생의 질문에 답변이 가능하다. 8. 학생은 자기 자신의 과제 및 채점 결과만 볼 수 있다. |
| 추가  요구사항 |  | 1. 서버에 데이터를 저장하여 네트워크 통신을 이용할 수 있게 한다. 2. 각 게시물의 변화에 대한 알림이 사용자들에게 가도록 한다. 3. 게시물 보안 및 통신 데이터의 암호화 처리를 한다. |
|  |

1. **Subjects**

|  |  |  |
| --- | --- | --- |
| 개발 |  | 1. 네트워크  * 로그인한 모든 사용자들은 같은 네트워크를 공유함 * 사용자들의 계정 정보는 서버에 저장됨 * 게시물 데이터는 서버에 저장됨  1. 보안  * 통신 되는 모든 데이터는 암호화 처리 됨 * 학생은 자신의 과제와 결과, 또한 전체 평균만을 확인할 수 있음  1. 알림  * 학생이 과제 제출 시 교수, 조교에게 알림이 감 * 학생의 과제가 성공적으로 제출 되었을 경우 제출 된 시간을 포함하고 있는 알림이 본인에게 감 * 교수, 조교가 과제를 채점할 시 학생에게 채점 완료 알림이 감 * 알림이 가야 할 상대가 접속해있지 않을 경우 서버에 알림 데이터를 저장해놓고 추후 접속 시 알림이 가도록 함 |
| 개선 |  | 1. MVC 패턴 적용  * 오브젝트들을 Model, View, Controller 세 가지 요소로 분리함  1. 글 수정 및 삭제 기능 추가  * 학생이 올린 과제 게시물의 수정 및 삭제 기능 추가 * 교수, 조교의 과제 평가에 대한 수정 및 삭제 기능 추가 * 질문 게시물의 수정 및 삭제 기능 추가 * 댓글의 수정 및 삭제 기능 추가 |

**4. Task Plan and Distribution**

**Task Priority**

1. 네트워크 기능 구현
2. 보안 시스템 추가
3. 수정 및 삭제 기능 추가
4. 알림 기능 추가

**Task Distribution**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 유동규 |  | 박동원 |  | 윤준호 |  | 이승진 |
|  |
| 네트워크  보안 |  | 네트워크 |  | 알림 |  | 수정 및 삭제 |
|  |

**Member Allocation Chart**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | 5월 1주 | 5월 2주 | 5월 3주 | 5월 4주 | 6월 1주 | 6월 2주 | 6월 3주 |
| 유동규 | 보안 | | | | 네트워크 | 마무리 작업 | |
| 박동원 | 네트워크 | | | | | 마무리 작업 | |
| 윤준호 |  | 알림 | | | | 마무리 작업 | |
| 이승진 | 수정 및 삭제 | | | | 마무리 작업 | | |

**5. Strategies**

|  |  |  |  |
| --- | --- | --- | --- |
|  | MVC 패턴 공부 |  | 시스템 개선사항으로서 MVC 패턴 적용이 있으므로 MVC패턴의 형태나 적용법 등에 대해 확실히 알아두어야 한다. |
| 자바 네트워크 공부 |  | 네트워크를 기반으로 하는 서버와의 통신이 가능토록 시스템을 개발하기 위해 자바 네트워크 프로그래밍에 숙달되어야 한다. |
| 기존 시스템의 이해 |  | 기존 시스템을 깊이 이해하는 것이 개선은 물론 기능의 추가에 있어서도 반드시 필요하다. |
| 정확한 업무분담 및 확인 |  | 각자의 능력과 장점에 맞는 업무를 분담하고 진행 정도를 자주 확인하여 효율과 진행속도를 높인다. |
| 요구사항 재분석 및 테스트 |  | 요구사항의 재분석과 테스트를 통해 잘못 파악했거나 수정된 사항을 즉각 반영할 수 있어야 한다. |
| TDD를 활용한 개발 |  | 테스트 코드를 작성하여 원하는 부분의 완성여부를 빠르게 파악하고 신뢰도를 높일 수 있다. |

**6. Special**