|  |
| --- |
|  |
| **문제정의서(연구계획서)** |
|  |

|  |  |
| --- | --- |
| **과제명** | **SW 자율학습 강의 Docker+웹 서비스 개발** |

|  |  |
| --- | --- |
| 조 | 야간자율학습 조 |
| 지도교수 | 이영석 교수님 (서명) |
| 조원 | 201701972 고태완  201702002 김지혜  201702080 최수연 |

**1. 연구의 필요성**

현재 세계 이러닝 시장이 성장세를 이루면서 SW 자율학습 플랫폼 또한 증가하고 있다. 흔히 자주 쓰이는 SW 자율학습 플랫폼으로 udemy, coursera와 같은 MOOC와, codecademy, inflearn, 생활코딩 등이 있다. 이러한 SW 자율학습 플랫폼들로 인해 많은 사람들이 인터넷으로도 쉽고 간편하게 SW 학습을 할 수 있게 되었다. 하지만 이를 포함한 많은 SW 자율학습 플랫폼들에서 몇 가지 한계점이 드러난다.

먼저, SW 학습의 상당 부분이 실습 위주인 것에도 불구하고 실습 환경을 제공하는 서비스가 드물다는 것이다. 그래서 많은 사용자들이 SW 실습 강의를 따라가기도 전에 환경 설정에서 막히는 경험을 한다. 학습 강사가 환경 설정에 대한 강의를 해 주는 경우도 있으나, 사용자들마다 자신의 개발 환경 상태가 다를 것이므로 환경 설정 강의로는 한계가 있다. 또한 사용자가 자신의 수준을 인지하기 어려워서 수준에 맞지 않는 학습을 할 수 있다. 강의의 수준과 자신의 수준을 정확히 파악하지 못하고 강의를 학습하는 것에 어려움을 겪는 경우도 있다. 이렇게 사용자의 자율학습을 돕는 것에도 한계가 드러난다. 자율학습을 돕는 것에서 뿐만 아니라, 서비스 자체적으로도 한계점이 있다. 사용자가 강의를 직접 업로드 함으로써 서비스 사용자들이 자체적으로 서비스를 업데이트하는 기능 또한 모든 SW 자율학습 플랫폼들이 갖고 있진 않다. 이렇게 되면 서비스 운영자가 직접 서비스를 업데이트해야 하는 데, 그것 만으로는 지속적으로 업데이트를 하는 데 한계가 있다.

4차산업혁명이 현실화가 되어가면서 세계 각국에서 SW 역량을 갖춘 인재 양성에 힘쓰고 있다. 국내에서도 코딩 교육의 중요성을 인지하여 코딩 교육을 의무화하고 있다. 많은 재능 있는 SW 인재를 육성하는 것이 사회의 발전에 기여하는 것 뿐만 아니라, 더 나아가 국가 경쟁력 또한 확보 할 수 있다. 우리는 이러한 현재 SW 시장 흐름에 맞춰 기존에 SW 자율학습 플랫폼들의 한계점을 개선하여 SW 자율학습을 돕고 경쟁력 있는 SW 인재를 육성하는 SW 자율학습 플랫폼 서비스를 개발할 것이다.

**2. 연구의 목표 및 내용**

본 연구는 궁극적으로 SW 자율학습 웹 서비스인 “Code at Once”의 개발을 목표로 한다. 본 웹 서비스는 오픈형 플랫폼으로 사용자가 자율학습을 진행할 수 있도록 양과 질이 보장된 컨텐츠를 지속적으로 업데이트하는 것을 지향하며 꾸준한 사용자의 유입으로 서비스 유지를 목표로 한다.

본 연구는 사용자가 문서 형식의 강의를 제작하여 SW 과목에 대해 실습 환경을 제공하는 오픈형 플랫폼이다. 이를 통해 고정적이거나 무분별한 컨텐츠를 배제할 수 있으며 컨텐츠가 지속적으로 갱신될 수 있다.

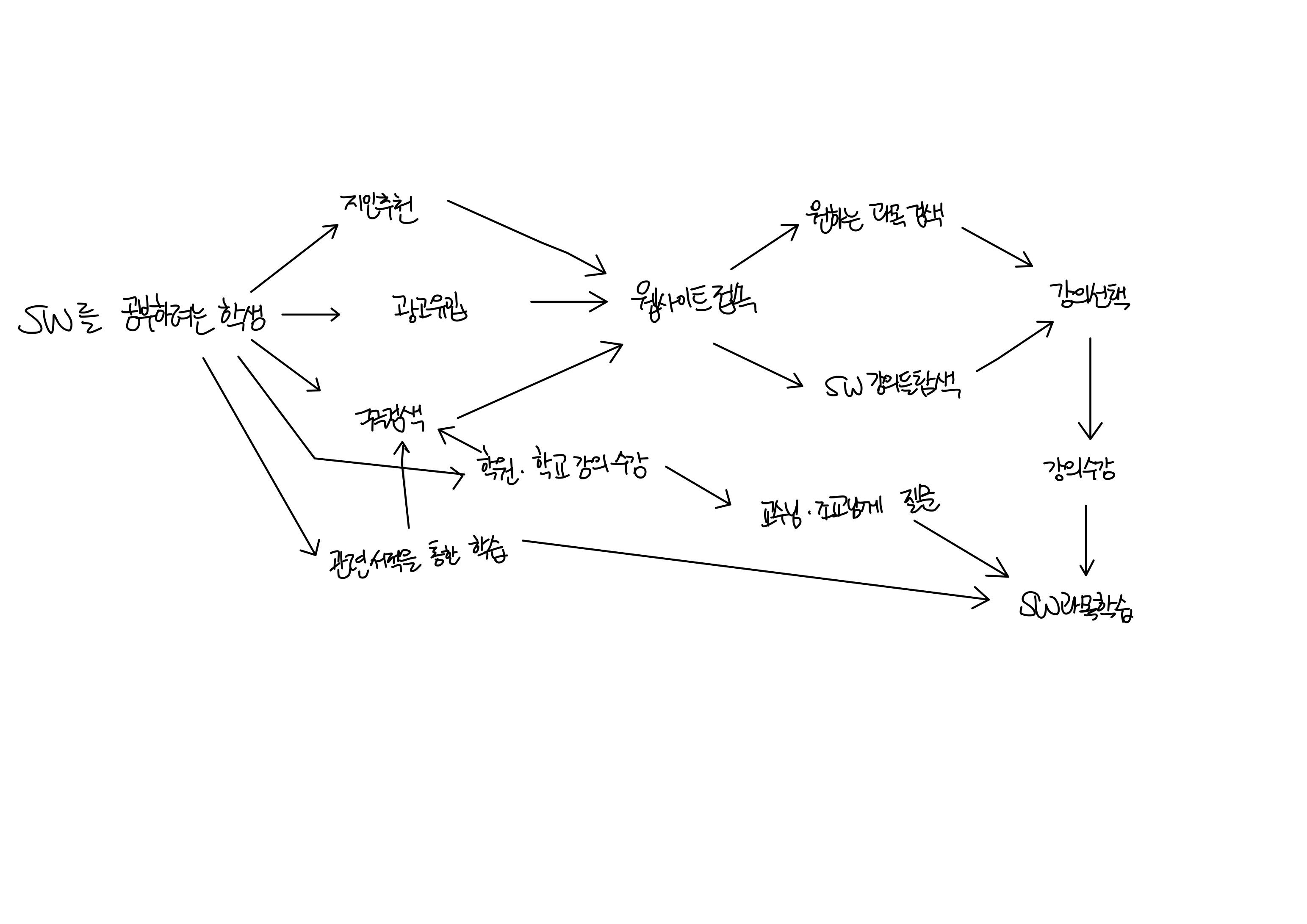
본 연구에서는 SW 과목을 공부하고자 하는 사용자가 자신의 수준을 인지하지 못해 최초 학습 시작 시 자가 수준 측정이 어려운 경우를 고려하여 Level Test 기능을 구현한다. 이 기능을 통해 서비스 이용자간의 평균 대비 자신의 수준을 인지할 수 있게끔 할 뿐만 아니라 사용자의 수준에 적절한 강의 목록을 제공한다. 강의에서 Docker를 이용하여 SW의 각 과목 실습 수강이 가능한 환경을 구축하여 코딩 실습 진행 시 Dokcer를 이용해 사용자 개개인이 독립적인 환경을 제공하여 환경 설정의 불편함을 해소한다.

편의 기능으로는 마이페이지에서 최근에 들었던 강좌로 이동하는 기능, 질의 응답을 모아서 보는 기능이 있다.

**3. 연구의 추진전략 및 방법**

구글에서 사용하는 의사 결정 방식인 Design Sprint를 이용한다.

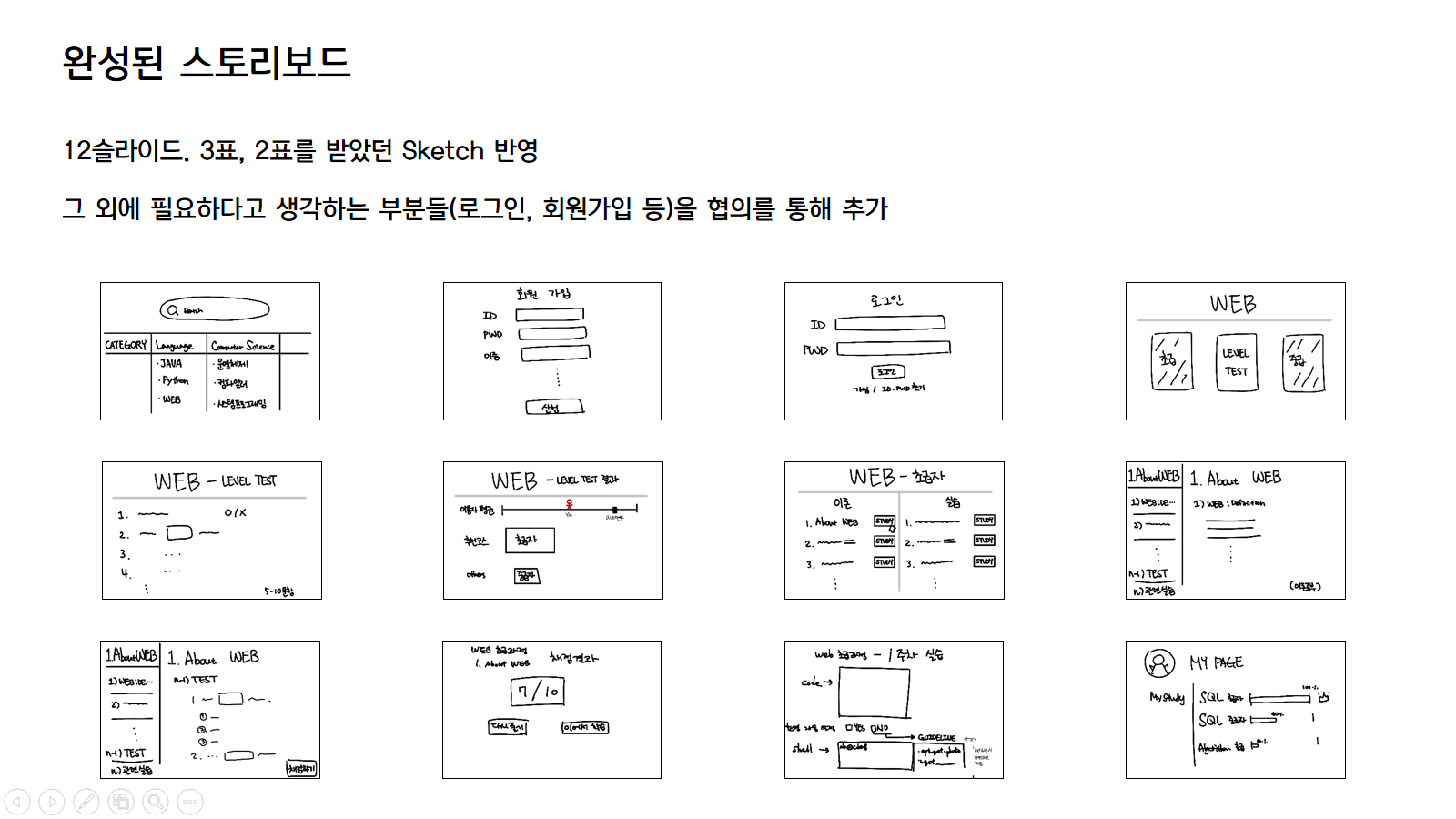
1일차: Map, 지도를 그려서 문제를 정의하고 타깃을 정한다.



2일차: Sketch, crazy 8's를 통해 각자 솔루션을 생성한다.



3일차: Decide, 가장 좋은 솔루션을 도출한다.



4일차: Prototype, 3일차에서 선택한 솔루션의 프로토타입을 만든다.



5일차: Test, Feedback을 통해 부족한 점을 보완하고 강점을 강화한다.



Design Sprint를 통한 의사 결정이 끝난 후, SW 공학적인 계획을 수립하여 목표 기간을 설정한다. 이 때 목표한 기능을 완수할 때 마다 마일 스톤을 세워 계획을 점검한다.

담당 교수 1명, 참여 인원 3명 그리고 담당 멘토 1명과 함께 지속적인 미팅 및 진척 상황 공유를 통해 협업한다. 이 때 협업은 Slack, Github, Zoom과 같은 협업 도구를 이용한다.

멘토링으로 받은 피드백으로 문서 수정 및 프로토타입을 재설정했다.

**4. 연구 팀의 구성 및 과제 추진 일정**

프로젝트의 구성 인원은 다음과 같다.

- 이영석: 담당 지도 교수

- 고태완: 클라이언트 개발 담당

- 김지혜: 서버 개발 담당

- 최수연: 서기 및 로직 구현

- 하석재: 담당 기업 멘토, 멘토링

추진 일정은 다음과 같다.



**- 참고문헌(Reference)**