



국민대학교  
소프트웨어융합대학  
소프트웨어학부


# 소프트웨어 프로젝트 2

## 프로젝트

프로젝트 명	무슨 색이게? (색 민감도 테스트)
팀 명	팀 SeokWoo (팀 A)
문서 제목	최종보고서

Version	1.3
Date	2021.12.6

팀원	20213075 전석환 (팀장)
	20213062 이찬우

 소프트웨어융합대학 소프트웨어학부 소프트웨어프로젝트 2	AD 프로젝트 결과보고서		
	프로젝트 명	무슨 색이게? (색 민감도 테스트)	
	팀 명	팀 SeokWoo (팀 A)	
	Confidential Restricted	Version 1.3	2021-12-6


#### CONFIDENTIALITY/SECURITY WARNING

이 문서에 포함되어 있는 정보는 국민대학교 소프트웨어융합대학 소프트웨어학부 및 소프트웨어학부 개설 교과목 소프트웨어 프로젝트 2 수강 학생 중 프로젝트 "무슨 색이게"를 수행하는 "팀 SeokWoo"의 팀원들의 자산입니다. 국민대학교 소프트웨어학부 및 "팀 A"의 팀원들의 서면 허락없이 사용되거나, 재가공 될 수 없습니다.

## 문서 정보 / 수정 내역


Filename	AD 프로젝트-무슨 색이게.docx
원안작성자	전석환, 이찬우
수정작업자	전석환, 이찬우

수정날짜	대표수정자	Revision	추가/수정 항목	내 용
2020-12-2	전석환	1.0	최초 작성	각자 무엇을 써올지에 대한 역할분담
2020-12-3	전석환	1.1	내용 추가	각자 작성해온 내용 확인 후 추가
2021-12-5	이찬우	1.2	내용 추가	목표 상세화, 개요, 개발 내용, 활용/개발된 기술, 부록 추가
2021-12-6	이찬우	1.3	내용 추가 및 최종 검토	개발 내용 추가, 시스템 구조 및 설계도 수정, 문서 스타일 통일, 참고문헌 작성, 머리/바닥글 수정

 소프트웨어융합대학 소프트웨어학부 소프트웨어프로젝트 2	AD 프로젝트 결과보고서		
	프로젝트 명	무슨 색이게? (색 민감도 테스트)	
	팀 명	팀 SeokWoo (팀 A)	
	Confidential Restricted	Version 1.3	2021-12-6

## 목 차

1	개요 .....	4
2	개발 내용 및 결과물 .....	5
2.1	목표 .....	5
2.2	개발 내용 및 결과물 .....	7
2.2.1	개발 내용 .....	7
2.2.2	시스템 구조 및 설계도 .....	11
2.2.3	활용/개발된 기술 .....	16
2.2.4	현실적 제한 요소 및 그 해결 방안 .....	21
2.2.5	결과물 목록 .....	22
3	자기평가 .....	24
4	참고 문헌 .....	26
5	부록 .....	27
5.1	사용자 매뉴얼 .....	27
5.2	설치 방법 .....	28


 <b>소프트웨어융합대학</b> <b>소프트웨어학부</b> <b>소프트웨어프로젝트</b> <b>2</b>	<b>AD 프로젝트 결과보고서</b>		
	<b>프로젝트 명</b>	무슨 색이게? (색 민감도 테스트)	
	<b>팀 명</b>	팀 SeokWoo (팀 A)	
	Confidential Restricted	Version 1.3	2021-12-6

## 1 개요

이번 AD 프로젝트를 통해 소프트웨어프로젝트 2 과목에서 배운 GUI 에 관한 내용들을 최대한 활용하면서 복습의 기회를 만들고 기존의 지식을 확장하고자 하는 마음이 있었다. 이를 위해 프로그램의 주제를 어떤 것으로 할지 많은 고민을 기울이게 되었고, 고민 끝에 정답 색을 기억해서 그 색을 찾아내는 “무슨 색이게?”라는 제목의 게임을 만들기로 결정하게 되었다.

본 프로젝트를 수행하는 과정에서 우리 팀은 요구사항 수집 및 분석, 구조 설계, 소프트웨어 설계, 소프트웨어 구현, 소프트웨어 시험의 단계를 거쳤다. 요구사항 수집 및 분석 단계에서는 요구사항명세서(SRS)를 작성하였고, 구조 설계 단계에서는 대략적인 소프트웨어의 구조를 설계했으며, 소프트웨어 설계의 단계에서는 클래스 인터페이스 설계 및 소프트웨어 구조 설계서(ADS)를 작성했다. 설계한 것들을 바탕으로 소프트웨어 구현 단계에서는 본격적인 코딩 작업에 들어갔으며, 팀원이 2 명이므로 전반적으로 한 명이 특정 부분의 설계 구현을 맡고, 다른 한 명이 다른 부분의 구현을 맡는 방식으로 진행했으며, 코드 작성, 코드의 개선이라는 주기를 반복적으로 수행했다. 최종적으로 프로그램이 완성되기까지 소프트웨어 시험 단계를 통해 각 소스 파일별 단위 테스트를 진행했고, 오류를 잡아내는 과정을 거쳤다. 이러한 단계를 순차적으로 거치며 프로젝트를 진행했다.

본 프로젝트에서 사용된 외부 라이브러리는 PyQt5 뿐이며, 다른 라이브러리들은 파이썬에 기본적으로 내장되어 있는 라이브러리이다. PyQt5 는 pip 를 이용하여 설치할 수 있는데, command 콘솔에서 pip install PyQt5, pip install PyQt5-tools 명령어를 순차적으로 입력하여 설치할 수 있다. 또는 파이참에서 File 버튼 > Settings > Project: [Project Name] > Python Interpreter 로 이동한 후, 패키지 목록에서 + 버튼을 눌러 PyQt5 를 추가할 수 있다.

 소프트웨어융합대학 소프트웨어학부 소프트웨어프로젝트 2	AD 프로젝트 결과보고서		
	프로젝트 명	무슨 색이게? (색 민감도 테스트)	
	팀 명	팀 SeokWoo (팀 A)	
	Confidential Restricted	Version 1.3	2021-12-6

## 2 개발 내용 및 결과물

### 2.1 목표

적용단계	내용	적용 여부
1 단계	요구사항 수집 및 분석	적용
2 단계	구조 설계	적용
3 단계	소프트웨어 설계	적용
4 단계	소프트웨어 구현	적용
5 단계	소프트웨어 시험	적용

#### 1) 요구사항 수집 및 분석 단계에서의 목표

**세부 목표 1-1)** PyQt5 를 이용하여 우리가 어느정도 수준의 프로그램을 만들 수 있는지 알아본 후 무슨 프로그램을 만들지 정하는 것을 목표로 하였다.

**세부 목표 1-2)** 그리고 “무슨 색이게”를 만들기로 결정한 후에는 “무슨 색이게”에 대한 요구사항 분석을 기능적 요구사항, 게임 로직의 기능적 요구사항, 사용자 인터페이스 요구사항, 비기능적 요구사항으로 나누어 분석함으로써 “무슨 색이게”가 무슨 프로그램인지에 대한 내용을 더욱더 구체화시키는 것을 목표로 하였다.

**세부 목표 1-3)** 그 다음 목표는 요구사항 분석을 한 내용을 요구사항명세서를 작성하면서 단계적으로 정리함으로써 게임의 전체적인 틀을 다듬는 것이었다.

#### 2) 구조 설계 단계에서의 목표

 소프트웨어융합대학 소프트웨어학부 소프트웨어프로젝트 2	AD 프로젝트 결과보고서		
	프로젝트 명	무슨 색이게? (색 민감도 테스트)	
	팀 명	팀 SeokWoo (팀 A)	
	Confidential Restricted	Version 1.3	2021-12-6

**세부 목표 2-1)** 요구사항명세서를 가지고 MVC 모델에 맞추어 소프트웨어 구조 설계를 하는 것을 목표로 하였다.

### 3) 소프트웨어 설계 단계에서의 목표

**세부 목표 3-1)** 소프트웨어 구조 설계한 것을 토대로 클래스 인터페이스 설계를 하는 것을 목표로 하였다.

**세부 목표 3-2)** 클래스 인터페이스 설계한 것을 토대로 소프트웨어 구조 설계서를 작성하는 것을 목표로 하였다.

**세부 목표 3-3)** 클래스 인터페이스 설계한 것을 토대로 구현 상계 설계를 하고 소프트웨어 상세 설계서 작성을 하는 것을 목표로 하였다.


### 4) 소프트웨어 구현 단계에서의 목표

**세부 목표 4-1)** 구현 상계 설계와 소프트웨어 상세 설계서 작성 대신 클래스 인터페이스 설계한 것과 소프트웨어 구조 설계서를 가지고 소프트웨어를 구현하는 것을 목표로 하였다.

**세부 목표 4-2)** 소프트웨어 구현을 할 때는 처음부터 우리가 설계한 소프트웨어를 한 번에 다 구현하지 않고 소프트웨어의 틀을 구현하고 거기에 추가적인 기능들을 구현하는 것을 목표로 하였다.

### 5) 소프트웨어 시험 단계에서의 목표

**세부 목표 5-1)** 소프트웨어를 다 구현하고 소프트웨어를 시험하는 것이 아니라, 소프트웨어를 구현하는 중간중간에 구현한 각각의 기능들이 잘 작동하나 시험해 보는 것을 목표로 하였다.

 <b>소프트웨어융합대학</b> <b>소프트웨어학부</b> <b>소프트웨어프로젝트</b> <b>2</b>	<b>AD 프로젝트 결과보고서</b>		
	<b>프로젝트 명</b>	무슨 색이게? (색 민감도 테스트)	
	<b>팀 명</b>	팀 SeokWoo (팀 A)	
	Confidential Restricted	Version 1.3	2021-12-6

**세부 목표 5-2)** 그리고 소프트웨어가 다 구현된 후에는 사용자가 이 소프트웨어를 사용하면서 겪게 될 상황들과 그 상황에서 소프트웨어는 어떤 식으로 작동해야 하는지 정하고 실제로 그 상황에 소프트웨어가 우리가 정한 대로 작동하는지 확인하는 것을 목표로 하였다.

## 2.2 개발 내용 및 결과물

### 2.2.1 개발 내용

#### 1) 요구사항 수집 및 분석 - 타임라인

**10 월 12 일:** 팀원 간 첫 만남을 가졌다. 아직 GUI 에 대하여 서로 익숙하지 않아서 16 일까지 GUI 에 대하여 서로 알아오기로 했다.

**10 월 16 일:** 서로 무슨 프로그램을 만들지에 대하여 서로 얘기해 봄으로써 프로젝트의 주제를 대략적으로 정하게 되었다. 이때 회의를 통해 나온 아이디어는 다음과 같다.


- 1. 화면에 글자가 나오고 사용자가 그 글자를 얼마나 빨리 치는지 알려주는 게임
- 2. 아이템이 있고 장애물이 움직이는 공룡 뽀틀 게임
- 3. 어떠한 색이 화면에 뜨면 다음 화면에서 그 색을 맞추어야 하는 색맹 테스트

**10 월 16 일 ~ 10 월 21 일:** 해당 기간 동안 한 가지의 아이디어가 더 추가되었는데, 해당 아이디어는 다음과 같다.

- 4. GUI 구성 요소인 버튼 및 슬라이더를 통해 떨어지는 미사일의 속도와 플레이어의 이동속도를 직접 설정할 수 있게끔 입력을 받는 슈팅 게임

발생된 아이디어들 중 최종적으로 어떤 것을 본 프로젝트의 주제로 결정할 지에 대해 추가적인 논의를 거쳤다.

**10 월 21 일:** PyQt5 내의 라이브러리들을 이용하여 많은 기능들을 접목시킬 수 있는지와 완성될 프로그램의 독창성 등 여러 요소를 고려해 보았을 때 여러 아이디어들 중 3. 색맹

 소프트웨어융합대학 소프트웨어학부 소프트웨어프로젝트 2	AD 프로젝트 결과보고서		
	프로젝트 명	무슨 색이게? (색 민감도 테스트)	
	팀 명	팀 SeokWoo (팀 A)	
	Confidential Restricted	Version 1.3	2021-12-6

테스트가 가장 적합할 것 같아서 색맹 테스트로 프로젝트를 결정하였고, 프로그램 이름을 '무슨 색이게?'로 설정하였다. 이로써 세부 목표 1-1 을 완료하였다.

**10 월 21 일 ~ 10 월 30 일:** 각자 '무슨 색이게?'에 대한 요구사항을 분석했다. 기능적 요구사항, 게임 로직의 기능적 요구사항, 사용자 인터페이스 요구사항, 비기능적 요구사항으로 나누어 작성하는 과정을 거쳤다.

**10 월 30 일:** 서로 분석해온 내용을 공유하고 병합함으로써 세부 목표 1-2 를 완료하였다 또한 프로그램의 레이아웃 틀에 대해서도 의논함으로써 페이지의 개수를 4 개로 설정할 것으로 결정했고 각 페이지마다 필요한 구성 요소들에 대한 이야기를 나누었다.

**11월 1일:** 10월 30일까지 수렴된 의견을 바탕으로 요구사항 명세서를 작성하였고, 내용은 다음과 같다. 이로써 세부 목표 1-3 을 완료하였다.

## 2) 구조 설계 – 타임라인


**11 월 4 일:** MVC(Model, View, Controller) 모델을 기반으로 하여 기본적인 소프트웨어의 구조를 설계했다. M, V, C 각각 필요할 것으로 예상되는 클래스, 파일을 설정했다. Model 에는 Time 클래스와 Difficulty 클래스를, View 에는 각 페이지 파일을, Controller 에는 필요할 것으로 예상되는 알고리즘들을 배치했다.

**11 월 7 일:** 11 월 4 일까지 설계한 부분을 수정하여 소프트웨어 구조 설계를 작성했다. 이로써 세부 목표 2-1 을 완료하였다.

## 3) 소프트웨어 설계 – 타임라인

**11 월 9 일:** 클래스 인터페이스 설계서 문서를 생성하여 Model 부분을 작성했다. 설정 창에서 시간 값을 받아오는 부분과 난이도 값을 받아오는 부분을 각각의 메서드로 분리하여 Value 클래스라는 한 클래스에 넣도록 했다.



 소프트웨어융합대학 소프트웨어학부 소프트웨어프로젝트 2	AD 프로젝트 결과보고서		
	프로젝트 명	무슨 색이게? (색 민감도 테스트)	
	팀 명	팀 SeokWoo (팀 A)	
	Confidential Restricted	Version 1.3	2021-12-6

**11 월 9 일 ~ 11 월 11 일:** 전석환은 클래스 인터페이스 설계서의 View 로 설정했던 부분을 작성했고, 이찬우는 클래스 인터페이스 설계서의 Controller 로 설정했던 부분을 작성했다.

**11 월 11 일:** 서로 작성해 온 클래스 인터페이스 설계서를 합침으로써 최종적인 클래스 인터페이스 설계서를 완성했다. 이로써 세부 목표 3-1 를 완료하였다.


**11 월 15 일:** 클래스 인터페이스 설계서에 기반하여 소프트웨어 구조 설계서를 완성했다. 이로써 세부 목표 3-2 를 완료하였다.

세부 목표 3-2 를 완료한 후 클래스 인터페이스 설계한 것을 토대로 구현 상계 설계를 하고 소프트웨어 상세 설계서를 작성하는 세부 목표 3-3 을 수행하려 했지만, 우리 팀원 모두 GUI 에 대한 코딩 경험이 부족했기 때문에 계획대로 수행한 것들이 제대로 적용되지 않을 가능성이 있었고, 더욱 상세하게 설계하게 된다면 코딩 시에 수정해야 할 부분이 매우 많아질 발생할 가능성이 있었다. 따라서 세부 목표 3-3 은 건너뛰고 소프트웨어 구조 설계서까지 작성한 것에 따라 코딩 단계로 넘어갔고, 다음 단계에서 추가적으로 필요한 변수나 메서드가 있을 경우 추가하고, 수정해야 할 부분이 발생할 경우 수정하기로 하였다.

#### 4) 소프트웨어 구현 - 타임라인

**11 월 19 일 ~ 11 월 22 일:** 11 월 19 일에 처음으로 학교에서 대면하였고, 전석환 팀원은 페이지 간의 전환 부분, 이찬우 팀원은 설정 창, 도움말 창 등 서브 창들을 제외한 각 페이지의 메인 레이아웃 부분을 맡아 코딩하였다. 남은 부분은 11 월 22 일까지 각자가 맡은 부분을 완성해 오는 것으로 하였다. 이 기간동안 팀원 각자 다음과 같이 역할을 수행했다.

- **전석환:** game.py 파일에서 각 페이지간 전환이 가능하도록 구현하기
- **이찬우:** 각 페이지 파일인 onepage.py, twopage.py, threepage.py, fourpage.py 파일 각각 기본 창의 레이아웃을 구성하기

 <b>소프트웨어융합대학</b> <b>소프트웨어학부</b> <b>소프트웨어프로젝트</b> <b>2</b>	<b>AD 프로젝트 결과보고서</b>		
	<b>프로젝트 명</b>	무슨 색이게? (색 민감도 테스트)	
	<b>팀 명</b>	팀 SeokWoo (팀 A)	
	Confidential Restricted	Version 1.3	2021-12-6

**11 월 22 일 ~ 11 월 25 일:** 11 월 22 일까지의 결과를 바탕으로 팀원 각자 다음과 같이 역할을 수행했다.

- **전석환:** onepage.py 및 threepage.py 에 발생된 폰트 적용 관련 코드의 반복 줄이기, 만들어진 각 페이지 파일을 game.py 와 연결시키기
- **이찬우:** 배경사진 변경한 것 적용하기, twopage.py 버튼에 구현되지 않았던 보기 버튼을 추가하여 맞춰야 할 색이 표시되게끔 구현하기, twopage.py 색 창의 크기 수정, onepage.py 에서 설정, 도움말 버튼과 fourpage.py 에서의 참고자료 버튼을 누르면 서브 창이 뜨도록 구현하기

**11 월 25 일 ~ 11 월 27 일:** 11 월 25 일까지의 결과를 바탕으로 팀원 각자 다음과 같이 역할을 수행했다.


- **전석환:** 문제가 진행됨에 따라 2 페이지 및 3 페이지의 좌측 상단에 뜨는 숫자 값을 바뀌도록 구현하기, colorDecision.py 구현하기
- **이찬우:** 4 페이지 수정, colorArrange.py 와 answerCheck.py 구현하기

이 과정에서 setting.txt 파일을 새로 만들어 설정 값을 저장하도록 구현했다.

**11 월 27 일 ~ 12 월 2 일:** 이 기간동안 팀원 각자 다음과 같이 역할을 수행했다.

- **전석환:** colorArrange.py, answerCheck.py, answerCheck.py 를 game.py, twopage.py, threepage.py 에 병합, 추가 텍스트 파일을 생성하여 게임 진행에 필요한 값들 저장되도록 구현하기, threepage.py 에서 compare 메서드를 통해 정답 색과의 비교를 구현하기
- **이찬우:** choicesColorDecision.py 파일 내 알고리즘 구현하기, 각 페이지 파일의 불필요한 주석 삭제하기, compare 메서드의 반복 제거하기, fourpage.py 에서 결과에 따라 다르게 표시되는 코멘트 구현하기

전반적으로 소프트웨어 구현 단계에서는 설계서에서 작성했던 것을 바탕으로 프로그램을 구현하려고 노력했지만 설계서에서 다루지 않았던 파일들을 생성하거나, 파일들을 병합하는 과정이 있었다. 따라서 설계서 때 작성했던 그대로 구현하지는 못했다

 <b>소프트웨어융합대학</b> <b>소프트웨어학부</b> <b>소프트웨어프로젝트</b> <b>2</b>	<b>AD 프로젝트 결과보고서</b>		
	<b>프로젝트 명</b>	무슨 색이게? (색 민감도 테스트)	
	<b>팀 명</b>	팀 SeokWoo (팀 A)	
	Confidential Restricted	Version 1.3	2021-12-6

때문에 세부 목표 4-1 은 부분적인 완료를 거두었다. 또한 프로그램의 전체적인 틀을 만들어 놓고 하나 하나씩 기능들을 추가로 코딩하여 구현했기 때문에 세부 목표 4-2 는 완료하는데 성공했다.

## 5) 소프트웨어 시험 – 타임라인

12 월 2 일 이후 프로젝트 종료 전까지 소프트웨어 시험 단계를 거쳤다. 각 페이지 파일별로 코드 하단에 `if __name__ == "__main__":` 를 두어 각 파일에서 정상적으로 작동되는지 검토를 거친 후 `game.py` 를 실행했고, 코드를 수정하거나 추가할 부분이 생길 때마다 이를 이용해 검토를 거쳤다. 따라서 세부 목표 5-1 을 완료하는데 성공했다.

또한 완성된 프로그램을 두고, 사용자가 정상적으로 모든 문제들을 순차적으로 풀 후 결과를 확인하는 경우 말고도 문제를 푸는 도중에 중단하고 프로그램을 종료했다가 다시 게임을 시작하는 경우 등 발생할 수 있는 다양한 경우에 오류가 발생하지 않도록 시험하는 과정을 거쳤다. 이로써 세부 목표 5-2 를 완료하는데 성공했다.


## 2.2.2 시스템 구조 및 설계도

### 1) 페이지 전환 (전석환)

본 프로젝트에서 구상한 “무슨 색이게?” 프로그램은 문제를 풀어야 하는 프로그램이므로 시작화면(`onpage.py`), 색이 보여지는 화면(`twopage.py`), 색을 고르는 화면(`threepage.py`), 결과화면(`fourpage.py`)인 이 4 페이지를 상황에 맞게 넘나들 수 있어야 했다.

그래서 `game.py`라는 파이썬 파일을 만들어서 이 기능을 구현하였다. 외부 코드를 참고 <sup>1</sup> 하였는데, `onpage.py`, `twopage.py`, `threepage.py`, `fourpage.py`마다 `switch_window = QtCore.pyqtSignal()`과 `self.switch_window.emit()`를 사용하여 각 페이지마다 특정버튼을 누르면 다음 페이지로 넘어가게 `game.py`와 버튼만 있는 `onpage.py`, `twopage.py`, `threepage.py`,

<sup>1</sup> <https://gist.github.com/MalloyDelacroix/2c509d6bcad35c7e35b1851dfc32d161>

 <b>소프트웨어융합대학</b> <b>소프트웨어학부</b> <b>소프트웨어프로젝트</b> <b>2</b>	<b>AD 프로젝트 결과보고서</b>		
	<b>프로젝트 명</b>	무슨 색이게? (색 민감도 테스트)	
	<b>팀 명</b>	팀 SeokWoo (팀 A)	
	Confidential Restricted	Version 1.3	2021-12-6

fourpage.py를 만들었다.

하지만 "무슨 색이게"는 총 10개의 문제를 푸는 프로그램이므로 3페이지에서 2페이지로 10번 돌아온 다음에 3페이지에서 4페이지로 넘어가야 한다. 그래서 game.py를 수정하여 question\_number.txt에서 불러온 값에 따라 if 문을 구성하여 리스트의 인덱스에 해당하는 값이 threepage.py에서 twopage.py로 10번 이동해야 threepage.py에서 fourpage.py로 연결되도록 했다.

## 2) 페이지 레이아웃 (이찬우)


onepage.py, twopage.py, threepage.py, fourpage.py의 레이아웃을 PyQt5를 이용하여 계획한 것과 같게 만들었다. 각각의 파일의 Layout 클래스마다 버튼 및 라벨 등의 객체를 생성하여 각 페이지의 기본적인 틀을 만들었다. 각각의 버튼 및 라벨마다 QFontDatabase와 QFont를 통해 '나눔스퀘어\_ac Bold' 폰트를 적용시켰고 외부 코드를 참조<sup>2</sup> 하여 각각의 레이아웃 클래스에 QPalette 객체를 생성하여 배경사진을 적용시켰다. 또한 각각의 창의 사이즈가 고정되도록 setFixedSize 메서드를 사용했고, 창의 제목을 setWindowTitle 메서드를 통해 적용했으며, setWindowIcon 메서드와 QIcon을 통해 창의 아이콘이 사진 파일로 적용되도록 했다. 또한 인터넷에서 코드를 참조<sup>3</sup> 하여 창이 moveCenter 메서드를 통해 모니터의 중앙에 위치되도록 적용했다. 그리고 각 페이지마다 주로 메인 레이아웃을 구성하는 서브 레이아웃을 나타내기 위해 QHBoxLayout과 QVBoxLayout을 이용했으며, 이러한 레이아웃들이 최종적으로 적용될 메인 레이아웃에는 QVBoxLayout을 사용하였다. 여기까지는 각 페이지 파일에 공통적으로 적용된 부분이며, 각 페이지별 적용 사항은 다음과 같다.

### 2-1) onepage.py

- QPushButton 객체인 self.settingButton, self.helpButton, self.startButton 버튼을 만들어 향후 각각 설정 버튼, 도움말 버튼, 시작 버튼의 기능을 수행할 수 있도록 했다.
- QLabel 객체인 self.settingLabel, self.helpLabel, self.titleLabel, self.subtitleLabel,

<sup>2</sup> <https://lsjsj92.tistory.com/319>

<sup>3</sup> <https://wikidocs.net/26684>

 <b>소프트웨어융합대학</b> <b>소프트웨어학부</b> <b>소프트웨어프로젝트</b> <b>2</b>	<b>AD 프로젝트 결과보고서</b>		
	<b>프로젝트 명</b>	무슨 색이게? (색 민감도 테스트)	
	<b>팀 명</b>	팀 SeokWoo (팀 A)	
	Confidential Restricted	Version 1.3	2021-12-6

self.ourName을 만들어 각각 설정, 도움말, 제목, 부제목에 해당하는 텍스트를 표시하도록 했다.

## 2-2) twopage.py

- QPushButton 객체인 self.showButton 버튼을 만들어 색 보기 버튼의 기능을 수행할 수 있도록 했다.
- QLabel 객체인 self.questionNumLabel, self.waitLabel, self.transparentColor, self.color를 만들어 각각 문제 번호, 상단 메시지, 색이 표시되기 전에 표시되는 투명한 색, 보여질 색을 표시하도록 했다. setStyleSheet를 이용해 색을 적용시켰다.

## 2-3) threepage.py

- QPushButton 객체인 각각의 색깔 버튼을 반복문을 통해 생성하여 기능을 수행할 수 있도록 했다.
- QLabel 객체인 self.questionNumLabel, self.label1 을 만들어 각각 문제 번호 ,상단 메시지를 표시하도록 했다.

## 2-4) fourpage.py

- QPushButton 객체인 self.restartButton, self.finishButton, self.reference 버튼을 만들어 향후 각각 재시작 버튼, 종료 버튼, 참고 자료 버튼의 기능을 수행할 수 있도록 했다.
- QLabel 객체인 self.resultLabel, self.scoreLabel, self.scoreLetterLabel, self.scoreCommentLabel, self.restartLabel, self.finishLabel 을 만들어 각각 게임 결과 위의 텍스트, 점수, 점수에 따른 코멘트, 다시 시작, 종료를 표시하도록 했다.


## 3) 1)과 2)를 합침 (전석환)

레이아웃이 제대로 구현된 onepage.py, twopage.py, threepage.py, fourpage.py를 game.py의 페이지 전환 메서드로 연결시켰다.

## 4) 각 페이지 마다 작동해야 하는 기능들을 구현 (이찬우, 전석환)

### 4-1) onepage.py

설정 버튼을 누르면 창이 뜨게 하고 그 창의 레이아웃을 구현하였다.(이찬우)

 <b>소프트웨어융합대학</b> <b>소프트웨어학부</b> <b>소프트웨어프로젝트</b> <b>2</b>	<b>AD 프로젝트 결과보고서</b>		
	<b>프로젝트 명</b>	무슨 색이게? (색 민감도 테스트)	
	<b>팀 명</b>	팀 SeokWoo (팀 A)	
	Confidential Restricted	Version 1.3	2021-12-6

- 수정한 부분: Layout 클래스에 clickedSetting 메서드 생성 및 SettingLayout에 연결  
 도움말 버튼을 누르면 창이 뜨게하고 그 창의 레이아웃을 구현하였다.(이찬우)

- 수정한 부분: Layout 클래스에 clickedHelp 메서드 생성 및 HelpLayout에 연결  
 setting.txt를 사용하여 설정 창에서 설정한 설정값이 유지되도록 하였다.(전석환)

- 수정한 부분: SettingLayout 클래스, changedTime 메서드, changedDiffi 메서드

#### 4-2) twopage.py

왼쪽위에 있는 문제번호가 문제의 번호에 맞게 갱신되게 question\_number.txt를 사용하여  
 구현하였다. (전석환)

- 수정한 부분: game.py의 ShowPages 클래스, show\_three\_to\_twopage 메서드,  
 twopage.py의 Layout 클래스 \_\_init\_\_ 메서드

색 보기 버튼을 누르면 색이 나타나는 기능을 구현 하였다.(이찬우)

- 수정한 부분: showColor 메서드를 Layout 클래스 내에 생성하여 색 보기 버튼에  
 커넥팅하도록 변경

색이 보여진 후 1페이지에서 설정한 시간이 지난 후에 3페이지로 넘어가는 기능을 구현하였다.  
 이때 외부 코드를 참고<sup>4</sup>하였다.(전석환)

- 수정한 부분: twopage.py Layout 클래스 showColor 메서드

보여지는 색이 랜덤하게 정해지는 알고리즘을 구현하였다.(이찬우)

- 수정한 부분: random 모듈의 randint 메서드를 사용하여 RGB 값을 무작위로  
 생성하도록 구현


colorList.txt를 사용하여 랜덤하게 정해진 색을 저장하는 기능을 구현하였다.(전석환)

- 수정한 부분: twopage.py의 Layout 클래스, showColor 메서드

#### 4-3) threepage.py

왼쪽 위에 있는 문제번호가 문제의 번호에 맞게 갱신되게 threepage.py 코드를 일부

<sup>4</sup> <https://makersweb.net/python/1098>

 소프트웨어융합대학 소프트웨어학부 소프트웨어프로젝트 2	AD 프로젝트 결과보고서		
	프로젝트 명	무슨 색이게? (색 민감도 테스트)	
	팀 명	팀 SeokWoo (팀 A)	
	Confidential Restricted	Version 1.3	2021-12-6

수정하였다.(전석환)

- 수정한 부분: game.py ShowPages 클래스 show\_three\_to\_twopage 메서드, twopage.py Layout 클래스 \_\_init\_\_ 메서드

2페이지에서 나온 색과 1페이지에 설정한 난이도에 맞게 3페이지에 나와야 하는 색과 배치를 정하는 알고리즘을 구현하였다.(이찬우)

- 수정한 부분: 5) choicesColorDecision.py 부분 참조 바람

#### 4-4) fourpage.py

result.txt를 이용하여 게임에서 게임을 진행하면서 맞춘 문제의 개수를 기억하고 4페이지에 나오게 구현하였다.(전석환)

- 수정한 부분: threepage.py Layout 클래스 \_\_init\_\_ 메서드, threepage.py Layout 클래스 compare 메서드

점수에 따라 나오는 멘트가 다르게 하는 기능을 구현하였다.(이찬우)

- 수정한 부분: if문을 통해 텍스트를 달리 하여 QLabel에 적용하도록 구현

다시하기 버튼을 누르면 1페이지로 돌아가고 문제 수와 맞춘 문제의 개수가 초기화되는 기능과 끝내기 버튼을 누르면 프로그램이 끝나고 문제 수와 맞춘 문제의 개수, 게임 설정이 초기화 되는 기능을 구현하였다.(전석환)


- 수정한 부분: fourpage.py Layout 클래스 \_\_init\_\_ 메서드, fourpage.py Layout 클래스 clickedFinish 메서드, fourpage.py Layout 클래스 fourpage 메서드, fourpage.py Layout 클래스 closeEvent 메서드, game.py ShowPages 클래스 show\_onepage 메서드, game.py ShowPages 클래스 show\_fourpage 메서드

참고자료 버튼을 누르면 창이 뜨게하고 그 창의 레이아웃을 구현하였다.(이찬우)

- 수정한 부분: Layout 클래스에 clickedReference 메서드 생성 및 ReferenceLayout에 연결

### 5) choicesColorDecision.py (이찬우)

#### 5-1) ChoicesColorDecision 클래스의 decide 메서드

 <b>소프트웨어융합대학</b> <b>소프트웨어학부</b> <b>소프트웨어프로젝트</b> <b>2</b>	<b>AD 프로젝트 결과보고서</b>		
	<b>프로젝트 명</b>	무슨 색이게? (색 민감도 테스트)	
	<b>팀 명</b>	팀 SeokWoo (팀 A)	
	Confidential Restricted	Version 1.3	2021-12-6

난이도와 정답 색을 파라미터로 받으면 그 정답 색과 11 개의 난이도에 따라 생성된 무작위 색상코드 11 개를 담은 리스트를 반환하는 메서드이다. 구현된 알고리즘은 다음과 같다.


- 파라미터로 난이도와 색상코드를 받아, 난이도는 self.diffi 에 할당하여 self.variance 에 난이도에 따른 편차의 합을 할당하고, 색상코드는 RGB 로 분리하여 각각 self.r, self.g, self.b 에 할당한다.
- 11 번의 반복을 하는 for 문 내에서 while 문에 따라 devilList 내의 원소의 합이 self.variance 가 되도록 무작위로 편차를 생성한다. 이때 self.variance 가 128 보다 클 경우 최종적으로 생성된 색이 0 ~ 255 범위를 벗어날 가능성이 있으므로 이 경우 0 에서 128 사이의 난수가 생성되도록 필터링한다.
- 부호 결정을 위해 곱해질 값인 -1 과 1 을 pn 리스트에 할당하고, 기존 색의 RGB 값을 color\_rgb 리스트에 담는다. 그리고 color\_rgb 리스트의 길이인 3 회 만큼의 반복을 수행하여 각각의 편차에 부호를 적용하여 기존 값에 더한다. 이때 무한루프에 진입하여 0 ~ 255 범위를 벗어나지 않을 때까지 반복한 후 빠져나오도록 필터링한다.
- 최종적으로 생성된 RGB 값을 16 진수 색상코드로 변환시키고, 생성된 색상코드를 self.colorList 에 추가한다.
- 11 개의 색상코드가 모두 생성되면, 파라미터로 받았던 기존 색상코드 값을 self.colorList 의 무작위 위치에 삽입한다.
- 완성된 리스트를 리턴한다.

### 2.2.3 활용/개발된 기술

#### 사용된 라이브러리

##### 1) PyQt5



 소프트웨어융합대학 소프트웨어학부 소프트웨어프로젝트 2	AD 프로젝트 결과보고서		
	프로젝트 명	무슨 색이게? (색 민감도 테스트)	
	팀 명	팀 SeokWoo (팀 A)	
	Confidential Restricted	Version 1.3	2021-12-6

PyQt5 는 Qt 애플리케이션 프레임워크의 파이썬 버전의 라이브러리이며, GUI 구현을 위한 툴킷이다. PyQt5 를 사용하여 만든 인터페이스는 Windows, MacOS, Linux 등의 플랫폼과 관계없이 구동될 수 있으므로 게임 형식의 GUI 를 만들고자 하는 본 프로젝트 수행을 위해 다루게 되었다.

### 1-1) QtWidgets

QtWidgets 는 PyQt5 내에 존재하는 파일이며, 기본적인 스타일의 사용자 인터페이스를 생성하는 UI 요소의 집합을 제공한다. 본 프로젝트에서는 게임 전체가 작동하는 파일인 game.py 와, 각 페이지 파일인 onepage.py, twopage.py, threepage.py, fourpage.py 에서 QtWidgets 이 사용되었으며 사용된 하위 클래스는 다음과 같다.

- **QWidget**

QWidget 의 윈도우를 생성하기 위해 각 페이지의 레이아웃 클래스에서 상속을 받는 용도로 사용하였다.

- **QLabel**

각 페이지 파일에서 짧은 길이의 텍스트들을 나타내기 위한 용도로 사용하였다. 또한 twopage.py 에서는 텍스트가 삽입되지 않은 QLabel 을 통해 색을 표시하는 용도로 사용하였으며, 이를 구현하기 위해 QWidget 클래스의 setStyleSheet 메서드를 통해 색상값을 설정하여 QLabel 객체의 배경색을 설정했다.


- **QSizePolicy**

QWidget 클래스 내의 setSizePolicy 메서드의 전달인자로 QSizePolicy 의 Expanding 을 사용하였다. 이를 통해 QLabel 또는 QPushButton 이 주어진 공간 내에서 가로, 세로로 팽창할 수 있도록 하였다.

- **QApplication**

프로그램을 실행하기 위한 App 객체를 만들기 위해 사용하였다.

- **QVBoxLayout**

 소프트웨어융합대학 소프트웨어학부 소프트웨어프로젝트 2	AD 프로젝트 결과보고서		
	프로젝트 명	무슨 색이게? (색 민감도 테스트)	
	팀 명	팀 SeokWoo (팀 A)	
	Confidential Restricted	Version 1.3	2021-12-6

각 페이지 파일에서 수직 레이아웃을 구성하기 위해 사용하였으며, 주로 하위 레이아웃들을 통합한 메인 레이아웃을 구성하기 위해 사용하였고, 경우에 따라 하위 레이아웃에서도 사용하였다.

#### - QHBoxLayout

각 페이지 파일에서 수평 레이아웃을 구성하기 위해 사용하였으며, 주로 메인 레이아웃을 구성하기 위해 사용되는 하위 레이아웃을 구성하기 위해 사용하였다.

#### - QGridLayout

threepage.py 의 메인 화면에 표시되는 12 개의 색깔을 그리드 형식으로 배치하기 위해 사용하였다.

#### - QDesktopWidget

창을 모니터의 중앙에 배치하는 과정에서 availableGeometry 메서드 및 center 메서드를 통해 화면의 중앙 좌표를 계산하기 위해 사용하였다.

#### - QPushButton


각 페이지에서 사용되는 버튼을 구현하기 위해 사용하였다. 각 버튼마다 정해진 작업을 수행하도록 커넥팅하였다.

#### - QDialog

QDialog 의 윈도우를 생성하기 위해 각 페이지의 서브 레이아웃 클래스에서 상속을 받는 용도로 사용하였다. 각 페이지마다 메인인 되는 화면을 구현하기 위해 QWidget 을 상속받은 반면, QDialog 는 onepage.py 의 설정 창, 도움말 창, fourpage.py 의 참고 자료 창과 같은 서브 창을 구현하기 위한 용도로 사용하였다.

#### - QSlider

onepage.py 의 설정 창에서, 사용자로부터 제한 시간 및 난이도를 슬라이더를 통해 입력받을 수 있도록 구현하기 위해 사용하였다. 각 슬라이더의 값이 변할 때마다 실제 설정 값으로 반영될 수 있도록 커넥팅하였다.

 소프트웨어융합대학 소프트웨어학부 소프트웨어프로젝트 2	AD 프로젝트 결과보고서		
	프로젝트 명	무슨 색이게? (색 민감도 테스트)	
	팀 명	팀 SeokWoo (팀 A)	
	Confidential Restricted	Version 1.3	2021-12-6

#### - QLineEdit

fourpage.py 의 참고 자료 창에서 각 참고 자료에 해당하는 링크들을 한 줄의 형태로 나타내기 위해 사용하였다.

#### - QTextEdit

onepage.py 의 도움말 창에서 도움말의 각 파트별로 분리하여 여러 줄의 텍스트로 나타낼 수 있게끔 구현하기 위해 사용하였다.

### 1-2) QtCore

QtCore 모듈에는 이벤트 루프나 Qt 의 신호 등 주요한 클래스들이 포함되어 있다. 본 프로젝트에서는 각 페이지 간의 전환 및 신호 발생, 타이머 사용 등을 목적으로 사용하였다.

#### - Qt

Qt.AlignCenter, Qt.Horizontal 등 객체를 특정 위치 또는 방향으로 정렬하는 것을 구현하기 위해 setAlignment 메서드 및 setOrientation 메서드와 함께 사용하였다.


#### - QSize

onepage.py 와 fourpage.py 의 버튼에 사용되는 아이콘의 크기를 설정하기 위해 사용하였다.

#### - QTimer

twopage.py 에서 색 보기 버튼을 눌렀을 때 타이머가 작동되도록 구현하기 위해 사용하였다.

### 1-3) QtGui

 <b>소프트웨어융합대학</b> <b>소프트웨어학부</b> <b>소프트웨어프로젝트</b> <b>2</b>	<b>AD 프로젝트 결과보고서</b>		
	<b>프로젝트 명</b>	무슨 색이게? (색 민감도 테스트)	
	<b>팀 명</b>	팀 SeokWoo (팀 A)	
	Confidential Restricted	Version 1.3	2021-12-6

QtGui 는 윈도우 시스템 통합, 이벤트 처리, 2D 그래픽, 기본 이미징, 폰트 및 텍스트를 위한 클래스가 포함되어 있는 모듈<sup>5</sup>이다. 본 프로젝트에서는 폰트, 이미지, 아이콘을 설정하기 위해 사용하였다.

#### - QFontDatabase, QFont

각 페이지에 있는 텍스트에 폰트를 적용시키기 위해 사용하였다. QFontDatabase 에 addApplicationFont 메서드를 이용하여 소스 파일과 같은 폴더에 존재하는 폰트 파일을 불러와 사용할 수 있도록 했고, QFont 를 이용하여 폰트 객체를 생성하는 방식으로 사용하였다.

#### - QImage, QPalette, QBrush

각 페이지마다 배경 사진을 적용하기 위해 사용하였다. QImage 의 입력인자로 사진 파일의 경로를 입력하여 객체를 생성했고, 생성된 객체가 QBrush 와 QPalette 객체를 통해 적용될 수 있게끔 구현했다.


#### - QIcon

프로그램을 실행했을 때 작업 표시줄에 표시되는 아이콘 및 프로그램 창의 좌측에 표시되는 아이콘을 설정하기 위해 사용했다.

## 2) random

파이썬에서 기본적으로 제공되는 모듈이며, 난수를 생성하는 모듈이다. 본 프로젝트에서는 설정한 범위 내의 정수 난수 1 개를 생성하는 메서드인 randint 메서드와, 여러 원소가 담긴 컬렉션에서 무작위 원소를 뽑는 메서드인 choice 메서드만을 사용하였다. twopage.py 에서 정답 색을 무작위로 발생시키기 위해 randint 메서드를 사용하였고, choicesColorDecision.py 에서는 정답 색에 따른 11 개의 무작위 색을 생성하는 과정에서 randint 메서드와 choice 메서드를 사용하였다.

<sup>5</sup> <https://docs.huihoo.com/pyqt/PyQt5/QtGui.html>

 소프트웨어융합대학 소프트웨어학부 소프트웨어프로젝트 2	AD 프로젝트 결과보고서		
	프로젝트 명	무슨 색이게? (색 민감도 테스트)	
	팀 명	팀 SeokWoo (팀 A)	
	Confidential Restricted	Version 1.3	2021-12-6

## 새롭게 고안한 알고리즘

choicesColorDecision.py 파일에 ChoicesColorDecision 클래스를 만들어 파라미터로 받은 색상 값과 난이도 값에 따라 11 개의 무작위 편차만큼 떨어진 색깔들을 생성하고 이 색깔들이 담긴 리스트에 정답 색을 추가하여 총 12 개의 색상코드가 담긴 리스트를 반환하는 decide 메서드를 만들어 threepage.py 의 12 개 버튼들을 나타내기 위해 사용하였다.

### 2.2.4 현실적 제한 요소 및 그 해결 방안

#### 1) 구현 상세 설계와 소프트웨어 상세 설계서 작성

우리는 소프트웨어에 관한 지식과 소프트웨어를 만들어본 경험이 적어서 구현 상세 설계와 소프트웨어 상세 설계서를 작성할 수 없었다. (우리가 생각한 기능을 만들려면 클래스 안에 어떠한 기능을 하는 메서드가 필요한지에 관한 판단을 할 수 없는 수준이었다.) 따라서 우리는 클래스 인터페이스 설계한 것과 소프트웨어 구조 설계서 작성을 한 것을 토대로 코드의 구현 단계에 들어가기로 하였다.

#### 2) 더 좋은 알고리즘 구현

2-1) 2 페이지에서 각 문제마다 나오는 10 가지 색이 서로 안 비슷했으면 좋을 것 같다고 생각을 하였는데 막상 알고리즘을 구현할 때 서로 안 비슷한 10 가지 색을 고르는 알고리즘을 만들 수 없어서 2 페이지에 나오는 색은 문제를 풀때마다 무작위로 선택되게 알고리즘을 구현하였다.

2-2) 2-1)에서의 문제가 비슷하게 3 페이지에서 나오는 정답 이외의 11 가지 색끼리 서로 안 비슷하게 알고리즘을 구현하려 하였지만 이것도 알고리즘을 구현하지 못하여 정답 색과만 안 비슷하게 나머지 색이 정해지도록 choicesColorDecision.py 의 알고리즘을 구현하였다.

2-3) choicesColorDecision.py 에서 ChoicesColorDecision 클래스의 decide 메서드를 구현하는 과정에서 편차 3 개를 생성하는 로직에서 과부하와 반복을 최대한 줄이기 위해 while 문을 이용한

 소프트웨어융합대학 소프트웨어학부 소프트웨어프로젝트 2	AD 프로젝트 결과보고서		
	프로젝트 명	무슨 색이게? (색 민감도 테스트)	
	팀 명	팀 SeokWoo (팀 A)	
	Confidential Restricted	Version 1.3	2021-12-6

난수 사용을 배제하려 했지만, while 문을 사용하지 않고 각각의 편차에 종속되는 난수를 생성한다면 각 편차 값이 균일한 조건에서 생성되지 않는 양상을 보이는 것을 확인했다. 따라서 기존에 계획했던 방법이 아닌 while 문을 이용한 방법을 채택하기로 했고, 실제로 프로그램이 발생시키는 과부하의 양이 눈에 띄지 않을 만큼 작고 생성 시간 또한 오래 걸리지 않아서 기존 계획이 아닌 while 문을 사용하는 방법으로 알고리즘을 구현하였다.

## 2.2.5 결과물 목록

### 1) 소스 코드

#### game.py

게임을 구동하기 위해 실행해야 하는 파일이며, 조건에 따라 각 페이지 파일인 onepage.py, twopage.py, threepage.py, fourpage.py의 레이아웃들을 전환하는 역할을 수행한다.

#### onepage.py

게임을 실행했을 때의 첫 화면에 나타나는 레이아웃을 담당하는 파일이다.

#### twopage.py

시작버튼을 누른 후, 맞추어야 할 색이 표시되는 화면에 나타나는 레이아웃을 담당하는 파일이다.

#### threepage.py


12 개의 색 중에서 정답 색을 골라야 하는 화면에 나타나는 레이아웃을 담당하는 파일이다.

#### fourpage.py

게임 결과가 표시되는 마지막 화면에 나타나는 레이아웃을 담당하는 파일이다.

#### choicesColorDecision.py

매개변수로 입력받는 난수도 값과 정답 색상코드에 따라 무작위로 나머지 11 개의 색상코드를 생성하고, 생성된 11 개의 색상코드 리스트에 정답 색을 무작위 위치에 삽입한 리스트를 반환하는 decide 메서드를 포함하는 클래스가 담긴 파일이다.

 소프트웨어융합대학 소프트웨어학부 소프트웨어프로젝트 2	AD 프로젝트 결과보고서		
	프로젝트 명	무슨 색이게? (색 민감도 테스트)	
	팀 명	팀 SeokWoo (팀 A)	
	Confidential Restricted	Version 1.3	2021-12-6

## 2) 텍스트 파일

### colorList.txt

정답 색의 색상코드가 입력되는 파일이다.

### question\_number.txt

현재까지 진행한 문제의 개수가 누적되는 파일이다.

### result.txt

맞은 문제의 개수가 누적되는 파일이다.

### setting.txt

설정값(제한시간, 난이도)이 저장되는 파일이다.

## 3) 이미지 파일

### colortable.png

프로그램의 아이콘으로 사용되는 파일이다.

### finish.png

fourpage.py 에서 종료 버튼의 이미지로 사용되는 파일이다.


### gray.png

프로그램의 배경 사진으로 사용되는 파일이다.

### questionIcon.png

onepage.py 에서 도움말 버튼의 이미지로 사용되는 파일이다.

### restart.png

 소프트웨어융합대학 소프트웨어학부 소프트웨어프로젝트 2	AD 프로젝트 결과보고서		
	프로젝트 명	무슨 색이게? (색 민감도 테스트)	
	팀 명	팀 SeokWoo (팀 A)	
	Confidential Restricted	Version 1.3	2021-12-6

fourpage.py 에서 다시하기 버튼의 이미지로 사용되는 파일이다.

**settingIcon.png**

onpage.py 에서 설정 버튼의 이미지로 사용되는 파일이다.

#### 4) 폰트 파일

**NanumSquare\_acB.ttf**

프로그램 전반에 적용되는 폰트 파일이며, 무료 폰트를 사용하였다.

### 3 자기평가


#### - 전석환(20213075) 자기평가

‘무슨 색이게’에서 나의 역할은 요구사항 분석, 요구사항명세서 작성, 소프트웨어 구조 설계, 클래스 인터페이스 M, V 부분 설계, 소프트웨어 구조 설계서 작성, 프로젝트 소프트웨어의 연결 부분 코딩, 프로젝트 회의방 만들기(구글 미팅), 발표, 보고서 작성이었다.

프로젝트 수행 시 아직 나는 소프트웨어에 관한 지식과 소프트웨어를 만들어본 경험이 적어서 소프트웨어의 자세한 부분까지 설계하는 과정이 어려웠다. 앞서 말한 이유와 같은 이유로 내가 머리 속으로 생각한 것들을 코딩을 통해 구현하는 과정도 어려웠다. 그리고 프로젝트를 만드는 과정들 간의 시간조절이 힘들었다. 어떤 프로젝트를 만들지 정하고 소프트웨어를 설계하는데 시간을 많이 써서 소프트웨어를 구현할 때 시간이 촉박했다.

나는 ‘무슨 색이게’에 적극적이고 주도적으로 참여하여 도움이 되었다. 먼저 적극적으로 의견을 내어 무슨 프로젝트를 할 것인지 정하였다. 그리고 언제 만나서 무엇을 회의할 지 정하였다. 회의를하기로 한 날에는 내가 구글 미팅을 통해 회의방을 개설하여 팀원을 초대해서 회의를 시작하였다.



 소프트웨어융합대학 소프트웨어학부 소프트웨어프로젝트 2	AD 프로젝트 결과보고서		
	프로젝트 명	무슨 색이게? (색 민감도 테스트)	
	팀 명	팀 SeokWoo (팀 A)	
	Confidential Restricted	Version 1.3	2021-12-6

내가 생각하기에는 3~4 학년 때 지금 만든 프로그램을 확인해보는 과정을 통해 프로그램을 다듬는 과정이 있으면 좋을 것 같다. 3~4 학년이 되어서 소프트웨어에 관한 지식과 경험이 풍부해진 다음에 지금 만든 프로그램을 다시보면 프로그램을 더 좋게 다듬을 수 있을 것 같기 때문이다.


### - 이찬우(20213062) 자기평가

본 프로젝트에서의 나의 역할은 요구사항 분석 및 요구사항 명세서 작성, 소프트웨어 구조 설계, 클래스 인터페이스 M, C 부분 설계, 프로그램의 각 페이지 파일인 onepage.py, twopage.py, threepage.py, fourpage.py 의 메인 레이아웃 및 서브 레이아웃 전반에 대한 코딩, choicesColorDecision.py 내 ChoicesColorDecision 클래스의 decide 메서드 알고리즘 구현, 보고서 작성 및 최종 검토였다.

프로젝트를 수행하는 과정에서 팀원과 원활히 소통하는데 어려움이 많았다. 그동안 협업을 하며 코딩을 하는 프로젝트를 했던 경험이 거의 없었고, 개인적인 문제이지만 말을 심하게 더듬는 습관도 가지고 있어서 팀원과 프로젝트에 관한 아이디어를 어떤 방식으로 나누고 프로젝트를 진행해야 하는지 확신이 서지 않았다. 다만 팀원과 여러 차례동안 화상 회의를 통해 만남을 가짐으로써 점차 스스로 내가 갖추어야 할 태도나 말아야 할 부분이 무엇인지 등의 방향을 찾게 되었고, 이것을 프로젝트 수행 과정에 차츰차츰 적용하려고 노력했다.

나는 PyQt5 를 통해 onepage.py, twopage.py, threepage.py, fourpage.py 에 필요한 레이아웃 전반에 대해 코딩하여 프로젝트의 전체적인 틀을 설정하였고, 이것이 후에 세부적인 변경 요소들을 덧붙이는 과정에 많은 도움이 되었다. 초반부터 이 4 개의 파일을 주도적으로 맡아 관리했는데, 레이아웃뿐 아니라 폰트, 배경사진, 아이콘 설정, 화면 위치 설정 등 세부적인 부분에 대한 코딩도 맡음으로써 PyQt5 를 활용한 코딩 능력을 최대한 발휘할 수 있었다. 또한 choicesColorDecision.py 에 사용된 알고리즘을 구현하는 등 소프트웨어 구현 과정에서 많은 도움이 되었다. 그리고 소프트웨어 구상 및 설계 과정에서 적극적으로 의견을 내었다.


프로젝트 초기에 예상했던 대로 본 프로젝트에서 개선이 필요한 점은 코딩 경험 부족으로 인해 설계서에 계획했던 대로 코딩되지 않은 부분이 많았다는 것과 설계의 수정이 잦았다는 것이었다.

 소프트웨어융합대학 소프트웨어학부 소프트웨어프로젝트 2	AD 프로젝트 결과보고서		
	프로젝트 명	무슨 색이게? (색 민감도 테스트)	
	팀 명	팀 SeokWoo (팀 A)	
	Confidential Restricted	Version 1.3	2021-12-6

다음 프로젝트에서는 코딩 경험을 더욱 많이 갖고 소프트웨어의 구조에 대한 이해도를 높여서 계획했던 것을 크게 수정하는 일들을 겪지 않도록 노력이 필요할 것 같다.

## 4 참고 문헌

번호	종류	제목	출처	발행년도	저자	기타
1	웹페이지	MalloyDelacroix / PyQt5WindowChangeExample.py	<a href="https://gist.github.com/MalloyDelacroix/2c509d6bcad35c7e35b1851dfc32d161">https://gist.github.com/MalloyDelacroix/2c509d6bcad35c7e35b1851dfc32d161</a>	2018	Malloy Delacroix	각 페이지를 넘나들 수 있도록 구현할 때 참고한 코드
2	웹페이지	파이썬 PyQt5 gui 배경 색깔 및 이미지 등 스타일과 디자인 - 1	<a href="https://lsjsj92.tistory.com/319">https://lsjsj92.tistory.com/319</a>	2018	이수진	배경 이미지를 설정할 때 참고한 코드
3	웹페이지	08) 창을 화면의 가운데로	<a href="https://wikidocs.net/26684">https://wikidocs.net/26684</a>	2020	WikiDocs	창이 모니터의 중앙에 배치되도록 구현할 때 참고한 코드

 <b>소프트웨어융합대학</b> <b>소프트웨어학부</b> <b>소프트웨어프로젝트</b> <b>2</b>	<b>AD 프로젝트 결과보고서</b>		
	<b>프로젝트 명</b>	무슨 색이게? (색 민감도 테스트)	
	<b>팀 명</b>	팀 SeokWoo (팀 A)	
	Confidential Restricted	Version 1.3	2021-12-6

4	웹페이지	[pyqt4]QTimer 예제 - 버튼을 누르면 3 초후 함수 또는 메소드 호출	<a href="https://makersweb.net/python/1098">https://makersweb.net/python/1098</a>	2015	Makersweb	타이머를 사용할 때 참고한 코드
5	웹페이지	QtGui	<a href="https://docs.huihoo.com/pyqt/PyQt5/QtGui.html">https://docs.huihoo.com/pyqt/PyQt5/QtGui.html</a>	2017	<a href="https://docs.huihoo.com/">https://docs.huihoo.com/</a>	QtGui 에 대한 설명

## 5 부록


### 5.1 사용자 매뉴얼

#### 게임 설명

- '무슨 색이게?' 게임은 주어지는 색을 잠깐 보고 그 색이 무슨 색이었는지 맞추는 게임입니다.

#### 게임 방법

- 설정 버튼을 통해 게임에 적용될 시간 제한과 난이도를 설정할 수 있다. 설정을 마친 후 Start 버튼을 누르게 되면 게임을 시작할 수 있습니다.
- 게임이 시작되면 각 라운드마다 맞춰야 할 색깔을 확인할 수 있는 버튼이 나타나고, 이용자가 설정한 시간 제한에 맞게 색이 보여진 후 자동으로 다음 화면으로 넘어갑니다.
- 그리고는 주어졌던 색을 12 개의 색 중에서 찾아서 버튼을 눌러야 하며, 10 개의 라운드가 끝나면 총 몇 개를 맞추었는지 확인할 수 있습니다.

 <b>소프트웨어융합대학</b> <b>소프트웨어학부</b> <b>소프트웨어프로젝트</b> <b>2</b>	<b>AD 프로젝트 결과보고서</b>		
	<b>프로젝트 명</b>	무슨 색이게? (색 민감도 테스트)	
	<b>팀 명</b>	팀 SeokWoo (팀 A)	
	Confidential Restricted	Version 1.3	2021-12-6

- 게임을 중간에 중단하여도 다시 실행하면 이전 과정을 이어서 플레이할 수 있습니다.

## 5.2 설치 방법

1. 파이참(Pycharm Community Edition)을 다운로드 후 설치한다.
2. 파이참에서 PyQt5 가 설치되어 있지 않다면 PyQt5 를 설치한다.
3. 무슨 색이게.zip 압축파일을 다운로드 받은 후, 압축을 푼다.
4. 생성된 '무슨 색이게' 폴더를 우클릭한 후 'Edit with Pycharm Community Edition'을 클릭하여 프로젝트를 연다.
5. game.py 파일을 실행한다.