슬라이드1

안녕하세요 저희는 자바프로그래밍 실습 7조입니다. 저희는 이번 실습 프로젝트로 동물 이름 맞추기 게임을 만들었습니다.

슬라이드 2

먼저 목차는 다음과 같습니다. 이번 프로젝트의 주제, 목표 등 소개 그리고 자바 코드의 구조, 구현시 실행 모습을 두 가지 측면: player 1과 player 2의 모습에서 바라봤을 때 어떻게 실행됐는지 보여드리고 마지막으로 각 조원의 역할을 설명해 드리겠습니다.

슬라이드3

먼저 이 게임의 이름은 call of animal입니다. 이 게임은 5승제로 게임 판수가 진행되는데 플레이어 1은 방(server)를 여는 사람을 말합니다. 플레이어 1은 포트넘버와 자신의 이름을 입력하여 서버를 열게되고 플레이어 2는 플레이어 1이 만든 방의 포트넘버와 자신의 이름을 입력하여 서버에 접속합니다. 플레이어 1이 gamestart버튼을 누르면 게임이 시작됩니다.

슬라이드4

이 게임은 기본적으로 1:1 방식으로 두 플레이어가 겨루는 방식입니다. 랜덤으로 화면에 나오는 동물 사진을 보고 동물의 이름을 상대보다 먼저 입력하면 점수가 1점씩오르고 각 라운드 마다 30초의 시간제한이 있는데 그안에 둘다 정답을 입력하지 못하면 무승부로 처리되고 둘다 점수는 오르지 않습니다. 무승부로 처리되거나 한사람이 정답을 입력하면 동물의 울음소리와 함께 정답이 화면에 출력됩니다. 한 플레이어가 5점에 먼저 도달하면 게임은 끝이나고 승자가 정해집니다.

슬라이드5

다음은 게임중의 화면입니다. 화면 상단과 같이 라운드수, 남은 시간이 출력되고, 동물의 사진이 화면에 나옵니다. 이때 플레이어 1이 먼저 정답을 맞추면

슬라이드 6

다음 화면과 같이 동물의 울음소리가 소리와 텍스트로 출력이 되고 플레이어 1의 점수가 1점 올라갑니다.

슬라이드7

만약 30초안에 둘다 정답을 입력하지 못했을 시 같은 방식으로 동물의 울음소리가 소리와 텍스트로 출력이되지만 둘다 점수는 올라가지 않습니다.

슬라이드 8

이 게임의 목표는 도심에서 다양한 동물들을 접할 수 없는 초등학생들이 늘어남에 따라 자바로 구현된 해당 프로젝트로 간접적을 동물의 모습과 울음소리에 대해 배울 수 있습니다. 또한 동물의 이름을 직접 영어로 입력하는 과정을 통해 초등학교 학생들의 영어 학습을 증진시킬 수 있습니다. 앞에 소개했듯이 각 라운드마다 30초라는 제한시간과 상대방보다 먼저 정답을 입력해야한다는 요건 때문에 순발력과 영어실력을 요구하는 게임이 될 수 있고 경쟁심을 불러 일으킬 수 있습니다.

슬라이드 9

다음은 이 프로그램의 전체 코드 구조에 대하여 설명하겠습니다.

슬라이드 10

전체적인 클래스 다이어그램은 다음과 같습니다. 메인 클래스에서 메인화면을 띄우고 메인화면에 두가지 입력선택이 있습니다.

슬라이드 11

플레이어가 게임시작 버튼을 누르면 게임스크린 클래스를 불러서 서버를 운영, 또는 게임의 흐름을 조정하는데 이때 game, random, result, setting의 클래스가 필요합니다. 만약 플레이어가 answer 버튼을 누르면 메인 화면은 answer화면을 띄우는데 이 화면은 게임 시작전 정답 예시를 보여주는 화면입니다. 이화면을 통해 플레이어들은 게임 시작전에 미리 동물에 대해 공부를 할 수 있습니다.

슬라이드 12

게임 스크린 클래스는 총 3가지 클래스를 부릅니다. Setting 클래스는 서버를 관리하는데 다른 플레이어와의 연결을 관리힙니다. result클래스는 플레이어 1과 플레이어2의 점수를 받아 승자를 판단하고 game 클래스는 각 플레이어의 점수, 각 라운드수, 남은 시간 등을 관리하고 각 플레이어가 입력한 텍스트가 정답인지 판단합니다.

슬라이드 13

게임스크린 클래스 안에는 총 3개의 클래스가 있는데 server client timer 가 있습니다. Server는 run, relaymessages 함수를 사용합니다. run함수는 socket을 이용해 서버를생성하고 mutithread로 relaymessage를 구현합니다. Relaymessage 함수는 게임도중, 플레이어 1 이 cat을 입력하면 ”client1: cat”을 양측 플레이어 화면에 출력해주는 등 메시지를 전송해주는 함수입니다. Client는 서버에 접속을 해주며 각 플레이어에 대한 클래스이고 timer는 30초를 새는 타이머 역할을 하는 함수입니다.

슬라이드 14

Setting 패널에서는 클라이언트를 위한 소켈을 생성합니다. 이때 멀티스레드를 이용하여 구현합니다.

슬라이드 15

추가로 부연설명을 하자면 다음의 각 함수들은 player1(server)에 대한 함수와 player2(client)에 대한 함수로 나뉩니다. Create room은 local ip에 대해 방을 만듭니다. 이때 서로 다른 공유기는 연결이 되지 않습니다. Your’s Name은 호스트의 이름을 설정하고 create는 클라이언트에게 퀴즈 화면을 전송합니다. 이 과정으로 생성된 서버에 아래 5개의 함수를 이용하여 클라이언트가 생성되고 서버에 접속을 합니다.

슬라이드 16

다음은 java로 구현한 모습입니다.

슬라이드 17

게임 패널은 채팅박스를 구현합니다. 각 클라이언트들은 이 채팅박스를 통해 시간초과 정답여부등을 알 수 있습니다. 상대방이 입력한 것을 이 채팅박스로 기록을 확인 할 수 있으며 그것이 정답인지 아닌지를 알 수 있습니다.

슬라이드 18

result패널에서는 게임종료후 결과에 대해 알려줍니다. 승패 혹은 무승부 여부를 확인 할 수 있고 각 플레이어는 quit버튼을 통해 게임을 종료 할 수 있습니다.

슬라이드 19

다음은 result패널을 구현한 모습입니다.

슬라이드 20

Answer 창에서는 게임의 정답 예시를 확인을 할 수 있습니다. 이때 animalList와 cryList는 각 동물의 이름, 소리에 대한 정보를 담고있는 배열이며 이 배열을 통해 랜덤으로 게임진행 시 문제를 랜덤으로 출력하거나 answer창에서 정답 예시를 순서대로 보여줍니다.

슬라이드 21

다음은 같은 공유기 네트워크에서 실제 서로 다른 노트북으로 게임을 실행했을 때 모습을 보여드리겠습니다.

슬라이드 22

우선 게임을 실행하고 게임스타트 버튼을 누르면 다음과 같이 create room 버튼 connect room 버튼을 볼 수 있습니다. 방을 만드는 사람은 포트번호와 이름을 적고 방을 만들고, 들어가는 사람은 상대방의 아이피 포트번호 이름을 입력하여 방에 들어갈 수 있습니다. 두 명이 들어오게 되면 방을 만든 사람의 화면에서 game start 버튼이 보이게 되고 이 버튼을 누르면 게임이 시작됩니다. 처음 문제가 시작되면 라운드, 스코어, 시간제한을 볼 수 있고 다음과 같이 submit 버튼을 통해 각자 정답이라고 생각하는 동물의 이름을 입력합니다. 정답은 rattle snake였고 client1이 맞추었기때문에 이 동물의 소리와 함께 로그에 정답을 맞춘 사람의 이름이 뜹니다. 다시 client1 화면에 next버튼이 생성되고 이 버튼을 통해 다음 라운드로 이동할 수 있습니다. 이때 라운드, 스코어는 갱신되고 시간제한이 다시 흐르기 시작합니다. 이번에는 두 사람이 열심히 정답을 맞추려 노력하고 있습니다만 정답이 나오지 않고 있습니다. 이렇게 되어 시간이 다 흐른 경우에는 타임아웃과 함께 이 라운드는 무승부 처리되어집니다. 따라서 다시 next 버튼을 눌러 다음 라운드로 이동해도 스코어는 갱신되지 않습니다. 한번 더 라운드가 진행되는 상황을 지켜봅시다.

슬라이드 23

이렇게 라운드가 흘러 한 사람이 먼저 5점의 점수를 따내게 되면 이긴 사람에게는 승리 결과창이

슬라이드 24

진 사람에게는 패배 결과창이 뜨게 됩니다. 이외에도 라운드가 모두 흐르고 여전히 두 사람의 스코어가 갔다면 양쪽 다 비겼다는 결과창을 볼 수 있습니다.

슬라이드 25,26

마지막으로 저희 팀의 조원들은 역할을 나누어 분담을 하였습니다. 아이디어를 다같이 토의하여 하나씩 개선점을 추가해서 이 주제를 생각해내었고, 코드 구현과 발표준비를 다음과 같이 나눠서

시행하였습니다

이상으로 발표를 들어주셔서 감사합니다.