

수업명 수요의 카페

프로젝트 제안서

제출일자: 12/15
제출자명: 황윤규
제출자학번: 214968

1. 프로젝트 목표

1) 배경 및 필요성 (14 pt)

고등학생 시절 게임이론을 게임으로 만들어 보여주는 게임을 해본 적이 있습니다. 특정 학문의 이론을 게임으로 접하게 되면서 재미도 있고 글로만 접하는 것이 아니라 더 이해하기 쉬웠던 경험이 있습니다. 이에 착안하여 제가 1학년 시절 경제학개론에서 어려움을 겪었던 기억이 있습니다. 그중에서 가장 핵심이었다고 생각이 드는 수요 공급 곡선과 관련된 내용 재미있게 만들면 좋겠다는 생각이 들어 게임으로 만들고자 합니다. 컨셉으로는 카페를 경영하는 게임으로 수요 공급 곡선에서 가격과 공급자가 원하는 가격인 균형 지점이라는 곳이 있습니다. 이때 플레이어는 카페 메뉴를 균형지점에 해당하는 가격을 찾아내어 돈을 가장 많이 버는 것을 목표로 하는 게임입니다. 거기에 다른 추가적인 베블런 효과나 외부효과와 같은 것들을 추가하여 색다른 재미와 동시에 더 추가적인 효과를 기대할 수 있습니다.

2) 프로젝트 목표

수요 공급 곡선의 핵심 개념인 가격이 오르면 소비자의 수요가 줄어듦과 가격을 내리면 수요가 증가한다는 개념을 게임을 통해 전하는 것을 목표로 합니다. 그리고 후에 엔딩에서 해당 개념을 설명해줌으로써 더 쉽고 재미있게 전하고자 합니다. 게임의 몰입도는 해치지 않되 게임을 진행함으로써 재미있고 유익한 시간이 되도록 하고 싶습니다.

3) 차별점

기존에 학습을 목표로 한 게임들은 대다수 그렇게 인지도를 가지지 못하였다고 알고 있습니다. 제가 느끼기에는 이런 게임들이 큰 반향을 일으키지 못한 까닭은 너무 학습에만 목표를 잡고 교육의 개념을 게임 내에 지속적으로 노출시켜 게임의 몰입감이 떨어졌기 때문이라고 생각합니다. 따라서 재미를 줄 수 있는 게임에 일종에 요소로 개념을 집어넣고 이를 핵심으로 두면 자연스럽게 해당 개념을 접하게되고 원리를 이해시킨 후에 엔딩에 이와 관련된 내용을 알려줌으로써 게임은 게임대로 몰입감을 더하고 마지막에 가르침도 얻을 수 있도록 하는 점이 차별점이라고 생각합니다.

2. 기능 계획

1) 기능 1 (플레이어 행동)

- 설명 플레이어의 행동 요소로 선택하여 행동하는 기능입니다.

(1) 세부 기능

- 플레이어의 행동은 낮과 밤으로 나뉘어지고 행동 요소가 달라집니다.

낮 : 1. 카페 메뉴 만들기 ,2. 디저트 만들기 2. 카페 메뉴 확인, 3. 정산 , 4. 미니 게임, 5, 판매 시작

밤 : 1. 메뉴 가격 수정 , 2. 직원 고용 , 3. 카페 메뉴 확인, 4. 정산, 5. 신문 확인하기, 6. 잠자기

===== (낮) =====

1) 기능 2 (카페 메뉴 만들기)

- 설명 : 카페의 메뉴의 이름과 가격을 입력받아 메뉴를 제작합니다.

(1) 세부 기능

- 메뉴의 가격과 이름을 입력받게되면 수요 공급 곡선에 따라서 가격에 맞추어 판매량이 결정 됩니다.

- 후에 플레이어 행동 양식(낮) 5번 판매 시작 시에 정해진 판매량에 따라 판매됩니다.

1) 기능 3 (디저트 만들기)

- 설명 : 디저트의 이름을 입력받아 메뉴를 제작합니다.

(1) 세부 기능

- 디저트는 커피에 절반만큼 팔립니다.
- 하루에 한번 나오는 신문에서 나온 특정 단어를 조합하여 만들면 더 많이 팔립니다.
- 디저트는 한 종류만 제작이 가능합니다.

1) 기능 4 (카페 메뉴 확인)

- 설명 : 현재 존재하는 카페 메뉴를 모두 보여줍니다.

(1) 세부 기능

- 메뉴의 이름과 가격을 보여줍니다.
- 만일 메뉴가 이미 한번 판매를 한 전적이 있다면 판매량또한 보여줍니다.
- 이는 후에 플레이어 행동 양식(밤)에 1번 메뉴 가격 수정시에 용이하게 사용됩니다.

1) 기능 5 (정산)

- 설명 : 지금까지 판매한 메뉴들의 총 판매량을 모두 합산하여 보여줍니다.

(1) 세부 기능

- 현재까지 판매한 판매량을 종합하여 확인할 수 있도록 합니다.

1) 기능 6 (미니게임)

- 설명 : 미니게임을 실행합니다.

(1) 세부 기능

- 미니게임을 진행하여 성공 시에 플레이에 도움이 되는 아이템을 지급합니다.

1) 기능 7 (판매 시작)

- 설명 : 현재 게시된 메뉴들로 판매를 시작합니다.

(1) 세부 기능

- 현재 게시된 메뉴들로 판매를 시작하되 판매량은 가격에 따라 정해진 양만 판매되어집니다.
- 판매량은 후에 플레이어 행동 양식(낮/밤) 카페 메뉴 확인을 통해 확인이 가능합니다.

=====-(밤)=====

1) 기능 8 (메뉴 가격 수정)

- 설명 : 메뉴의 가격을 수정합니다.

(1) 세부 기능

- 메뉴의 가격을 다시 수정할 수 있습니다. 커피는 가격에 따른 판매량도 재조정됩니다.

- 메뉴의 수정은 메뉴별로 한 번만 가능합니다.

1) 기능 9 (직원 고용)

- 설명 : 돈을 일정 소모하여 직원을 고용합니다.

(1) 세부 기능

- 직원 고용 시에 카페 메뉴를 추가로 더 제작이 가능해집니다.

- 직원 고용 시 3명의 직원을 보여주며 고용 비용은 무작위로 나옵니다.

1) 기능 10 (신문 확인하기)

- 설명 : 하루에 한번 신문이 나오게 됩니다.

(1) 세부 기능

- 신문에는 그날 유행하는 단어가 나오게 됩니다.

- 랜덤하게 하나의 신문을 제공하여 줍니다.

- "오늘은 사과가 맛있습니다."와 같은 신문이 나오면 디저트에 사과타르트와 같은 메뉴를 제작하면 더 많이 팔리게 됩니다.

1) 기능 11 (낮과 밤)

- 설명 : 플레이어는 낮과 밤에 따라 행동양식이 달라집니다.

(1) 세부 기능

- 플레이어의 행동은 낮과 밤으로 나뉘어지고 행동 요소가 달라집니다.

- 낮에는 플레이어 행동 양식(낮) 5번 판매 시작을 통해 판매가 끝나면 밤으로 이동합니다.

- 밤에는 플레이어 행동 양식(밤) 5번 잠자기를 통해 낮으로 이동합니다.

1) 기능 12 (시간)

- 설명 : 플레이어가 행동할 수 있는 시간은 7일로 제한되어 있습니다.

(1) 세부 기능

- 낮과 밤이 1일이며 밤이 지나면 1일이 지나가게 됩니다.
- 7일이 모두 지나게 되면 엔딩이 나오게 됩니다. 주요 공급 곡선에 관련된 내용을 소개해 줍니다.
- 제작자가 정해 놓은 가격을 달성시에 최종 엔딩이 나오게 됩니다.

1) 기능 13 (고객 피드백 시스템)

- 설명 : 판매 이후에 고객들이 메뉴에 대해 피드백을 남깁니다.

(1) 세부 기능

- 긍정 혹은 부정 피드백을 남깁니다.
- 긍정 피드백의 경우에는 균형 가격에 가까울수록 많아집니다. 반대로 멀수록 부정 피드백이 증가합니다.

1) 기능 14 (균형 가격 설정)

- 설명 : 새로운 메뉴를 만들수록 더 비싼 메뉴를 제작하여 균형 가격이 각기 다릅니다.

(1) 세부 기능

- 새로 만든 메뉴일수록 균형가격이 조금 더 비싸게 조정됩니다.

1) 기능 15 (베블런 효과)

- 설명 : 특별한 효과로 특정 메뉴를 가격을 높게 설정했을 때 판매량이 증가합니다.

(1) 세부 기능

- 미니게임에서 특정 이름의 메뉴의 힌트를 제공하며 해당 이름의 메뉴의 경우에는 가격을 높게 설정할수록 판매량이 증가합니다.

1) 기능 16 (외부 효과)

- 설명 : 하루에 한 번 신문이 나오며 그 신문에 단어에 대한 힌트가 제공됩니다.

(1) 세부 기능

- 신문에는 사과가 제철과 같은 단어가 나오면 사과가 들어간 디저트류는 더 많이 팔립니다.

수정 사항

1) 기능 6 (미니게임)

- 설명 : 미니게임을 실행합니다.

(1) 세부 기능

~~- 미니게임을 진행하여 성공 시에 플레이어에 도움이 되는 아이템을 지급합니다.~~

-> 미니게임을 통해서 그날 신문에 따라 정해지는 키워드를 맞추는 미니게임을 진행합니다,

-> 베블런 효과에 따른 미니게임을 고려중입니다.

1) 기능 14 (균형 가격 설정)

- 설명 : 새로운 메뉴를 만들수록 더 비싼 메뉴를 제작하여 균형 가격이 각기 다릅니다.

(1) 세부 기능

~~- 새로 만든 메뉴일수록 균형가격이 조금 더 싸게 조정됩니다.~~

- 균형 가격의 경우에는 랜덤으로 제공되어집니다.

1) 기능 16 (외부 효과)

~~- 설명 : 하루에 한 번 신문이 나오며 그 신문에 단어에 대한 힌트가 제공됩니다.~~

- 설명 : 하루에 한 번 신문이 나오며 신문을 통해 단어를 유추하거나 미니게임에서 그 단어 맞추기 게임이 진행됩니다.

(1) 세부 기능

- 신문에는 사과가 제철과 같은 단어가 나오면 사과가 들어간 디저트류는 더 많이 팔립니다.

3. 진척 사항

1) 기능 구현

(1) 기능 6 미니 게임

설명

-> 미니게임을 통해서 그날 신문에 따라 정해지는 키워드를 맞추는 미니게임을 진행합니다.

-> 베블런 효과에 따른 미니게임을 고려 중입니다.

적용된 배운내용

1) while 문

정답이거나 정해진 횟수 전까지 반복합니다.

2) 클래스

클래스 내의 멤버에 대한 접근 지정자를 통하여 접근이 가능하게 합니다.

3) map

각 신문의 이름을 key로 두어 값(value)를 나타냅니다.

index를 통해 값을 찾아냅니다.

코드 스크린샷

```
// 미니게임을 실행합니다.
void Event::MiniGame(int index) {
    // map에서 index에 해당하는 키 찾기
    auto it = newspaperKeywords.begin();
    advance(it, index);

    string keyword = it->second; // 해당 키워드
    string guessed(keyword.size(), '_');

    // 사용할 수 있는 횟수이며 10번으로 제한합니다.
    int attempts_left = 10;

    // 추측한 단어를 저장합니다.
    string guessed_letters = "";

    // 맞추거나 횟수를 모두 소진할 때까지 반복하여 실행합니다.
    while (attempts_left > 0 && guessed != keyword) {
        // 현재 상태 출력
        cout << "\n단어: ";
        for (char c : guessed) {
            cout << c << ' ';
        }
        cout << "\n남은 시도 횟수: " << attempts_left << "\n";
        cout << "추측한 글자: " << guessed_letters << "\n";
        cout << "알파벳 한 글자를 입력하세요: ";

        // 추측할 단어를 입력받습니다.
        char guess;
        cin >> guess;
        guess = tolower(guess);

        // 잘못된 입력이거나 이미 추측한 단어일 경우에 다시 진행합니다.
        if (!isalpha(guess) || guessed_letters.find(guess) != std::string::npos) {
            cout << "잘못된 입력입니다. 한 글자 알파벳만 입력하거나, 이미 추측한 글자를 제외해주세요.\n";
            continue;
        }
    }
}
```

1) 기능 10 (신문 확인하기)

- 설명 : 하루에 한번 신문이 나오게 됩니다.

(1) 세부 기능

- 신문에는 그날 유행하는 단어가 나오게 됩니다.

- 랜덤하게 하나의 신문을 제공하여 줍니다.

- "오늘은 사과가 맛있습니다."와 같은 신문이 나오면 디저트에 사과타르트와 같은

```

// 추측한 단어는 담아줍니다.
guessed_letters += guess;

// 추측한 알파벳이 포함되어져 있는 곳을 찾습니다.
bool found = false;
for (size_t i = 0; i < keyword.size(); ++i) {
    if (keyword[i] == guess) {
        guessed[i] = guess;
        found = true;
    }
}

// 포함 여부를 보여줍니다.
if (found) {
    cout << "좋아요! '" << guess << "'는 단어에 포함되어 있습니다.\n";
} else {
    cout << "안타깝습니다. '" << guess << "'는 단어에 없습니다.\n";
}

// 횟수를 소진합니다.
attempts_left--;
}

// 성공 여부를 보여줍니다.
if (guessed == keyword) {
    cout << "\n축하합니다! 단어를 맞추셨습니다: " << keyword << "\n";
} else {
    cout << "\n게임 오버! 다시 도전해주세요!!: " << "\n";
}
}

```

메뉴를 제작하면 더 많이 팔리게 됩니다.

적용된 배운내용

1) 파일 입출력

index를 통해 신문과 값을 가져와서 해당 파일이 존재하면 열어 해당 파일의 내용을 보여줌

2) map

각 신문의 이름을 key로 두어 값(value)를 나타냅니다.

index를 통해 값을 찾아냅니다.

코드 스크린샷

1) 기능 11 (낮과 밤)

- 설명 : 플레이어는 낮과 밤에 따라 행동양식이 달라집니다.

(1) 세부 기능


```

// 각종 이벤트와 관련된 클래스입니다.
class Event {
public:
    // 신문 파일과 키워드 저장합니다.
    map<string, string> newspaperKeywords = {
        {"newspaper1.txt", "nobel"},
        {"newspaper2.txt", "newjeans"},
        {"newspaper3.txt", "ott"},
        {"newspaper4.txt", "chuncheon"},
        {"newspaper5.txt", "electric"},
        {"newspaper6.txt", "apple"},
        {"newspaper7.txt", "ai"},
        {"newspaper8.txt", "martial law"},
        {"newspaper9.txt", "bitcoin"},
        {"newspaper10.txt", "subway"}
    };

    // 신문을 보여주는데 사용되어집니다. index의 경우에는 하루에 랜덤으로 선택되어집니다.
    void ShowNewspaper(int index);

    // 미니게임을 실행합니다.
    void MiniGame(int index);

    // 그날의 keyword를 반환합니다.
    string ReturnKeyword(int index);
};

```

```

// 신문을 보여주는 함수입니다.
void Event::ShowNewspaper(int index) {
    // map에서 index에 해당하는 키 찾기
    auto it = newspaperKeywords.begin();
    advance(it, index);

    string news = it->first; // 선택된 파일 이름
    string keyword = it->second; // 해당 키워드

    // 선택되어지는 신문을 확인합니다.
    //cout << "선택된 신문 파일: " << news << endl;
    //cout << "핵심 키워드: " << keyword << endl;

    // 파일 읽기
    ifstream is(news);
    if (!is) {
        cerr << "파일 오픈에 실패하였습니다: " << news << endl;
        return;
    }

    cout << "=====| 신문 내용 |===== " << endl;
    char c;
    while (is.get(c)) {
        cout << c;
    }
    cout << endl;
}

```

- 플레이어의 행동은 낮과 밤으로 나뉘어지고 행동 요소가 달라집니다.

- 낮에는 플레이어 행동 양식(낮) 5번 판매 시작을 통해 판매가 끝나면 밤으로 이동합니다.
- 밤에는 플레이어 행동 양식(밤) 5번 잠자기를 통해 낮으로 이동합니다.

적용된 배운내용

1) case문

밤에 case6를 실행하면 하루를 더해 날짜를 표시합니다.

2) while문

시간 기한인 7일이 되기 전까지 반복합니다.

코드 스크린샷

```
// 전체적인 행동 요소를 나타내는 함수 입니다.
void Run::run(){
    srand(static_cast<unsigned int>(time(nullptr)));
    int r = rand() % 10; // 인덱스는 0~9 범위
    // keyword를 받아옴

    string key_word = event.ReturnKeyword(r);
    // 디저트의 가격을 초기화 시킵니다.
    menu.ResetDessertPrice();

    int menu_select;

    int time = 0;

    while(day < 7){
case 6:
    // 시간을 낮으로 만듭니다.
    //anim.anim_text("자는중");
    time = 0;
    day++;
    break;
```

1) 기능 15 (베블런 효과)

- 설명 : 특별한 효과로 특정 메뉴를 가격을 높게 설정했을 때 판매량이 증

가합니다.

(1) 세부 기능

- 미니게임에서 특정 이름의 메뉴의 힌트를 제공하며 해당 이름의 메뉴의 경우에는 가격을 높게 설정할수록 판매량이 증가합니다.

적용된 배운내용

1) class

class를 사용하여 같은 객체를 여러 개 사용하기 위해서 활용하였습니다.

2) for문

for 문에서 메뉴의 숫자만큼 돌고, veblen 효과를 적용 시킬 단어를 보냅니다.

3) 참조자 매개변수

참조자 매개변수로 활용하여 함수에 veblen 효과가 한번만 적용되도록 (예를 들면 베블런 효과가 적용되는 단어가 메뉴 2개에 들어있을 경우) check_veblen 함수를 보내 확인합니다.

코드 스크린샷

```
void Menu::SellMenu(){
    bool check_veblen = true;

    // 디저트의 판매량은 커피의 절반만큼 팔립니다.
    // 다만 커피의 판매량은 가격에 따라 바뀔 수 있으므로 0으로 초기화하고 다시 받아줍니다.
    dessert_sell = 0;

    // 메뉴를 전부 돌면서 정해진
    for(int i = 0; i < menu_price.size(); i++){
        SupplyDemand(i, check_veblen);
    }

    // 커피의 모든 판매량의 절반만큼 판매되어 저장합니다.
    if(check_dessert_make){
        dessert_total_sell += 5000 * dessert_sell;
    }
}
```

1) 기능 16 (외부 효과)

~~- 설명 : 하루에 한 번 신문이 나오며 그 신문에 단어에 대한 힌트가 제공됩니다.~~

- 설명 : 하루에 한 번 신문이 나오며 신문을 통해 단어를 유추하거나 미니 게임에서 그 단어 맞추기 게임이 진행됩니다.

(1) 세부 기능

- 신문에는 사과가 제철과 같은 단어가 나오면 사과가 들어간 디저트류는 더 많이 팔립니다.

적용된 배운내용

1) getline()

메뉴 이름 작성시에 띄어쓰기도 입력을 받기 위해 사용하였습니다.

2) if문

함수를 통해 해당 키워드가 포함되어져 있는지를 체크합니다.

코드 스크린샷

```
// 디저트를 만듭니다.
void Menu::MakeDessert(string key){
    int price;
    string name;

    cout << "=====| 주의 사항 |===== " << endl;
    cout << "디저트의 이름은 영어로 작성하여 주시길 바랍니다." << endl;
    cout << "왜 한글이 안되는 것일까요...." << endl;
    cout << "===== " << endl;
    cout << endl;

    // 제작할 메뉴를 입력받습니다.
    cout << "디저트를 입력해주세요 : " << endl;
    cin.ignore(); // 이전 입력이 있다면 버퍼를 비웁니다.
    getline(cin, name);
    dessert_name = name;

    cout << "기본 디저트 가격은 5000원입니다. " << endl;

    // 그날의 키워드가 포함되어져 있는 지 체크합니다.
    if(containsWord(key, dessert_name)){
        cout << "디저트가 더 비싼 가격에 팔립니다!!" << endl;
        dessert_price = 7000;
    }
    else{
        dessert_price = 5000;
    }

    // 디저트 확인 시에 디저트의 여부를 판별
    check_dessert_make = true;
}
```

```
// 특정 단어가 문자열에 포함되어 있는지 확인하는 함수
bool Menu::containsWord(const string& targetWord, const string& Word) {
    istringstream stream(Word);
    string word;

    while (stream >> word) {
        if (word == targetWord) {
            return true;
        }
    }
    // 만일 단어가 포함되어 있지 않다면 false를 반환합니다.
    return false;
}
```

3. 프로젝트 일정 (참고: 간트차트)

업무	11/3	11/10	11/17	11/20	11/27
제안서 작성	(완료)				
기능 1		(완료)			
기능 2		(완료)			

업무	11/10	11/17	11/24	12/1	12/8
기능 3		(완료)			
기능 4		(완료)			
기능 5			(완료)		

업무	11/17	11/24	12/1	12/8	12/15
기능 6		(일부)			

기능 7		(완료)			
기능 8		(완료)			
업무	11/17	11/24	12/1	12/8	12/15
기능 9			(완료)		
기능 10			(완료)		
기능 11			(완료)		

업무	11/24	12/1	12/8	12/15	12/22
기능 12			(완료)		
기능 13			----->		
기능 14			(완료)		
업무	11/24	12/1	12/8	12/15	12/22
기능 15				(일부)	
기능 16				(완료)	