|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | | **작성자** | **2017180012 박주용**  **2017180029 이예준**  **2017182030 이수민** | **팀명** | FruitsPangPang |
| **주차** |  | **기간** | **2022.06.29~2022.07.05** | **지도교수** | **정 내 훈**(서명) |
| **이번주 한일** | * **박주용**   **액세서리 텍스처링, 로비 맵 수정, ai test, 기타 수정**   * **이예준**   **액세서리 모델링**   * **이수민**   **상점구현, 아이템 장착구현, Return Lobby기능구현, 잡다한 버그 수정** | | | | |

**<상세 수행내용>**

* + **공통**
* 테스트 및 (치명적인)버그 수정
  + **박주용**
* 액세서리 텍스처링(빨대, 막대사탕, 해바라기, 나뭇잎)

텍스트, 전자기기, 스크린샷이(가) 표시된 사진

자동 생성된 설명

* 로비 맵 수정

텍스트이(가) 표시된 사진

자동 생성된 설명

잔디, 하늘, 실외, 풀로덮인이(가) 표시된 사진

자동 생성된 설명

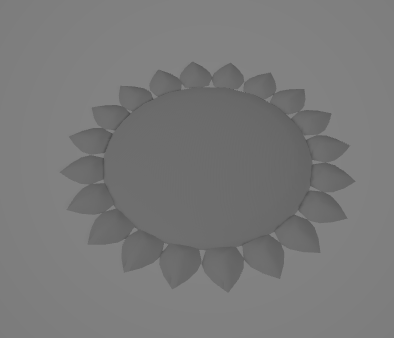
액세서리들 배치, 매칭 나무 표지판 수정 등.

* Ai test: ai perception test, smart ai test
* 기타 수정:

병원 내 trigger box와 과일이 충돌하는 버그 수정, heal spawner box collision, 구급상자와 과일이 충돌하는 버그 수정, 나무와 나무 상자에서 과일들 스폰 예쁘게 정렬되도록 수정.

* + **이예준**
* 액세서리 모델링(빨대, 막대사탕, 해바라기, 나뭇잎





* + **이수민**

db: item\_data 테이블 생성. (상점 NPC가 1명밖에 없기 때문에 이 테이블 전체가 상점 1개를 위한 테이블이다. 다른 NPC를 추가할 일은 없으니 이렇게 했다.)

---------------------

우리 게임의 상점은 플레이어가 아이템을 보유한다면 살 수 없어야 한다.

즉 수량이 있는 상점이다.

따라서 서버에서 수량이 맞는지 관리해 주어야한다.

---------------------

플레이어가 무슨 아이템을 가지고 있는지를 저장하는 테이블인 playerhaveitem 테이블에서 정보를 꺼내 와서 그 값을 기반으로 계산을 해야 한다.

플레이어가 로그인 할 때 db에서 playerhaveitem테이블 정보를 받고, 그 정보를 Character의 Shopinventory에 넣어준다.

ShopInventory를 기반으로 한 정보를 플레이어에게 보내서 상점 buttton을 enable, disable해준다.

플레이어가 구매를 시도하면 캐릭터의 Shopinventory를 검사하여 이미 구매한 물품이라면 계산하지 않고, 구매하지 않았을 때만 살 수 있도록 한다.

로비서버를 처음 킬 때, db에서 상점정보를 받아 NPC클래스에 넣고, 그 이후로는 이 npc클래스에 있는 정보를 사용한다.

----------------------------

-로비서버 구동 시: db서버에게 상점의 info를 요청한다. db에서 상점관련 db를 뽑아 로비서버에게 준다.

-플레이어가 로그인 시: db에서 플레이어가 보유한 아이템 목록을 뽑아서 login\_ok 패킷에 담아 보낸다. 이 때, 뽑은 아이템 목록은 자체적으로 Character 클래스에 저장해준다.

-------------클라이언트----------

-StoreItemBoxWidget과 StoreWidget의 C++화 insertItem() initializeitem()로 아이템을 widget에 업로드.

-기존 MainWidget에 그냥 존재하던 StoreWidget을 StoreBox(HorizontalBox)에 넣어서 스폰.

-MainWidget에 ScoreWidget 스폰코드 및 Setting 코드까지 추가.

- 각종 아이템 임시 Icon 추가

-파일을 기반으로 Store List를 만들고 파일을 읽어서 Store에 아이템들을 업로드. 어떤 구조로 할까 (txt, json ...) 고민하다가 수정하기 용이한 csv(comma separated values)를 사용하기로 결정. ue4에서 csv파일을 또 지원을 해준다. csv파일을 임포트 하게 되면, datatable(ue4가 지원해주는 구조)가 생긴다. 이때 기준이 되는 struct가 필요하기에 item.h에 해당 구조체를 만들어 주었다. (itemCode, itemName, Price) 그 다음에는 DataTable을 읽는 형식을 사용해서 파일입력을 해주었다. StoreList가 수정된다면, widget->Asset에 있는 Shop DataTable을 ReImport만 해주면 된다.

엄청났던 버그: #include <ppl.h>를 하면 소켓관련 프로그래밍이 버그가 난다??? 이유는 모르겠음.

서버의 상점과 클라의 상점의 기능을 동기화하는 작업.

---클라이언트---

- 아이템 구매 및 착용 기능 개발. (상대방의 착용 스킨도 적용돼서 보이도록 함.)

- Cash변수와 store의 cash 연결

- 소켓관련 변수 생성자에서 초기화 코드 추가.

---게임서버---

- 플레이어의 스킨 정보를 DB에서 얻어와야 했기에 DB서버와 연동. 그에 따라 필요한 세팅들 추가.

---DB서버---

- 게임서버에서 요청을 받고 정보를 줌. try\_login 프로시저로 충분히 구현 가능해서 재활용 했음.

DB구현 유의점: 프로시저의 EXEC Name Value1,Value2에서 value를 전달 할 때, varchar가 아닌 int, smallint 들은 값을 전달해줄 때, ASCII의 문자 값으로 보내야함. ex) 3을 전달하려면 0x03 이 아니라 '3'으로. 따라서 itoa를 통해서 int값을 문자열로 바꾸고 해당 문자열을 추가하는 식으로 해결.

- 아이템구매, 아이템 장착, 플레이어 정보 얻어 오기 프로시저를 구현.

--로비서버---

아이템구매, 아이템 장착 패킷을 받고, 정보 업데이트 및 DB업데이트를 구현

- 게임Result에서 Return Lobby 누를 시 로비로 이동 구현.

- 상점 R로 여는 것 F로 바꾸고 가까이 가서 열도록 수정

- 힐 팩 안 먹어지는 버그 및/InGAme에서 Editor Simulate Stop시, 로그인 화면 안 나오는 버그 수정

|  |  |  |  |
| --- | --- | --- | --- |
| **문제점 정리** |  | **해결 방안** |  |
| **다음 주차** |  | **다음 기간** | **2022.07.06~2022.07.12** |
| **다음주 할 일** |  | | |
| **지도교수**  **Comment** |  | | |