Capstone Design

Space Log 최종 보고 및 방학 계획

**목차**

1. 최종보고서
2. 방학계획서

201321333 한재선, 201021340 원용률

2016-06-26

1. 최종 보고
   1. 클라이언트
      1. 회원가입, 로그인, 로그아웃, 중복로그인
      2. 행성 정보 받아서 출력
      3. 방향키 입력에 따른 동작
   2. 서버
      1. 로그인  
         - 클라이언트로부터 정보를 받을 때 사용하는 메시지는 ‘login\_msg’   
         - DB MEMBER Collection에서 위의 정보(username , password)를 조회  
         - 일치하는 정보가 있으면 소켓 메시지 ‘login\_res’를 클라이언트에게 전송할 때 {response : ‘true’}를 전송 / 없으면 {response : ‘false’} 를 전송
      2. 로그아웃 및 중복로그인  
         - MEMBER Collection 에서 마지막 필드에 accessing을 추가하여 사용자가 접속중인지 아닌지를 판별.  
         - 회원가입을 할 때 기본적으로 accessing 필드 값은 flase 로 저장하여 접속하지 않은 상태임을 표시.   
         - 로그인을 하면 true 로 변경하여 현재 접속중임을 표시.
      3. 회원가입  
         - 클라이언트로부터 정보를 받을 때 사용하는 메시지는 ‘join\_msg’.  
         - MEMBER Collection 에서 위의 정보 중 ‘username’이 존재하는지 조회하고 이미 해당 username 이 존재하면, 클라이언트에게 전송할 때 join\_res 메시지로 {response : ‘false’}를 전송 / 존재하지 않으면 MEMBER Collection에 받은 정보를 모두 저장한 뒤 , {response : ‘true’}를 전송
      4. 행성 랜덤 생성   
         - 새로 생성하는 행성의 위치와 자원양을 랜덤으로 생성하기 위해 random 함수를 사용하고  
         - random 함수만 사용하면 일정 패턴이 반복되기 때문에 행성의 위치가 겹치는 문제가 발생.  
         - getTime 함수를 행성 생성 수식에 추가하여 위의 문제를 해결
      5. 행성정보 전송  
         - PLANET Collection 에서 모든 document를 조회하여 객체형식으로 클라이언트에게 모두 전송
2. 방학 계획
   1. 방학 중 진행 할 부분 목록
      1. 유저가 이동함에 따라 화면도 이동하기. 그리고 게임 종료시에 유저의 마지막 좌표값을 서버에게 전송하여 MEM\_INFO Collection의 좌표값 없데이트 하기.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| username | exp | mineral | gas | unknown | Location\_x | Location\_y |
|  |  |  |  |  | ★ | ★ |

* + 1. 유저가 개척한 행성 정보를 받아서 MEM\_PLANET Collection에 입력 및 업데이트

|  |  |
| --- | --- |
| 행성번호 | 유저 ID |
|  |  |

* + 1. 클라이언트는 유저가 개척하여 얻은 자원량 정보를 로컬스토리지에 저장하고 있다가 일정 시간이 지나면(ex. 1분 혹은 3분) 현재까지 유저가 얻은 자원량 정보를 서버에게 한꺼번에 전송. 이렇게 하는 이유는 유저가 자원을 획득하는 속도는 1초에 각 자원을 10씩 얻는데 (즉, mineral +10 , gas + 10, unknown + 10 . 이런 식으로 증가) 매번 자원량의 변동이 있을 때마다 그 정보를 계속 서버에게 전송하는 것은 비효율 적이므로 1분이나 3분에 한 번씩 누적 획득량을 서버에게 전송하는 게 좋다고 판단했다.

**서버**

1분 혹은 3분에 한 번 씩 전송

유저가 게임에서 획득한 자원양 정보

(로컬 스토리지)

* + 1. 유저가 개척해서 얻은 자원량 정보를 받아서 MEM\_INFO Collection에 해당 유저의 정보 업데이트

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| username | exp | mineral | gas | unknown | Location\_x | Location\_y |
|  | ★ | ★ | ★ | ★ |  |  |

* + 1. 서버에서는 상점에 들어갈 여러 큐브 데이터를 MISSILE Collection 과 SHIELD Collection에 저장하고, 클라이언트에게 이 데이터들을 전송하여 상점에 물품 목록이 뜨도록 하기

|  |  |
| --- | --- |
| missile | offensive |
|  |  |
| shield | defensive |
|  |  |

* + 1. MEM\_SHIP Collection 에 유저가 구입한 방어 큐브와 발사체 큐브의 공격력과 방어력을 계산한 값과 함선의 내구도 정보를 저장

|  |  |  |  |
| --- | --- | --- | --- |
| username | durability | defensive | Offensive |
|  |  |  |  |

* 1. 진행 계획 달력

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **7월**  **// : 보완 작업**  **ⅰ ⅱ ⅲ ⅳ ⅴ ⅵ ⅶ ⅷ ⅸ : 위의 리스트 항목 번호** | | | | | | | |
| **일** | **월** | **화** | **수** | **목** | **금** | **토** |
|  |  |  |  |  | 1  **ⅰ** | 2  **ⅰ** |
| 3  **ⅰ** | 4  **ⅰ** | 5  **ⅰ** | 6  **ⅰ** | 7  **ⅰ** | 8  **ⅰ** | 9  **ⅰ** |
| 10  **ⅱ** | 11  **ⅱ** | 12  **ⅱ** | 13  **ⅱ** | 14  **ⅱ** | 15  **ⅱ** | 16  **ⅲ** |
| 17  **ⅲ** | 18  **ⅲ** | 19  **ⅲ** | 20  **ⅲ** | 21  **ⅲ** | 22  **ⅲ** | 23  **ⅲ** |
| 24  **ⅲ** | 25  **//** | 26  **//** | 27  **//** | 28  **//** | 29  **//** | 30  **//** |
| 31  **//** |  |  |  |  |  |  |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **8월**  **// : 보완 작업**  **ⅰ ⅱ ⅲ ⅳ ⅴ ⅵ ⅶ ⅷ ⅸ : 위의 리스트 항목 번호** | | | | | | | |
| **일** | **월** | **화** | **수** | **목** | **금** | **토** |
|  | 1  **ⅳ** | 2  **ⅳ** | 3  **ⅳ** | 4  **ⅳ** | 5  **ⅳ** | 6  **ⅳ** |
| 7  **ⅳ** | 8  **ⅴ** | 9  **ⅴ** | 10  **ⅴ** | 11  **ⅴ** | 12  **ⅴ** | 13  **ⅴ** |
| 14  **ⅴ** | 15  **ⅵ** | 16  **ⅵ** | 17  **ⅵ** | 18  **ⅵ** | 19  **ⅵ** | 20  **ⅵ** |
| 21  **ⅵ** | 22 **//** | 23  **//** | 24  **//** | 25  **//** | 26  **//** | 27  **//** |
| 28  **//** | 29 | 30 | 31 |  |  |  |