|  |  |  |  |
| --- | --- | --- | --- |
| 도둑들 | | | |
| **2월 2주** | **2023. 02. 05 ~ 2023. 02. 11** | **작성자** | **정극훈** |
| **이번 주**  **목표** | **이도영**: send packet 관련 cpp코드 및 메인 함수를 실험 가능한 정도까지 작성  서버 공부  **정극훈**: 클라이언트 프레임워크 Rendering 부분까지 완성  **김혁동**: FBXLoader 개발 시작( ~2월 28일) | | |
| **2/5**  **일** | **이도영**: Send packet관련 cpp코드와 서버 main함수 작성  DB 코딩 시작  DB Test Client 제작 및 계획 수정 | | |
| **2/6**  **월** | **이도영**: DB로 로그인 및 회원가입 기능 구현 | | |
| **2/7**  **화** | **정극훈**: Resources class 추가   * resource와 관련된 부분은 Resources에서 관리   Object class 추가   * 모든 객체의 최상위 부모 역할을 하는 클래스   **김혁동**: FBX SDK를 바탕으로 FBX Loader 제작 | | |
| **2/8**  **수** | **이도영**: DB오류 수정 및 Local에서 테스트 완료  **정극훈**: Lighting class 추가   * 게임 화면에서 빛의 표현 구현   Normal Mapping 기능 구현: 객체에 normal map을 적용해 normal map에 표현된 질감을 구현  SkyBox: 게임의 배경인 skybox 구현 | | |
| **2/9**  **목** | **이도영**: DB 테스트    **정극훈**: 디자인패턴 공부  4장 직렬화 프로토타입 팩토리 학습  5장 싱글턴 디자인 패턴: 특정 컴포넌트의 인스턴스가 프로그램 전체에서 단 하나만 존재해야 하는 상황을 처리하기 위해 고안됨  싱글턴 패턴의 기본적인 구현 공부  **김혁동**: FBX SDK를 바탕으로 FBX Loader 제작 | | |
| **2/10**  **금** | **정극훈**: Frustum class 추가   * Frustum Culling 기능: 보이지 않는 물체를 렌더링하지 않아 프로그램의 성능 향상   카메라에 직교 투영 구현: 기초적인 UI 기반 마련 | | |
| **2/11**  **토** | **정극훈**: RenderTargetGroup class 추가  Forward Rendering에서 Deferred Rendering으로 렌더링방식 변경  화면에 객체를 그릴 때 객체마다 계산 후 그리지 않고 깊이, 노말 맵핑, 빛 등의 계산을 임의의 텍스쳐에 저장한 후 여러 객체에 한번에 적용한 후 최종 결과물을 화면에 표시  **도둑들 정기회의 (19시):**   * **로그인 UI 화면 회의** * **클라이언트에 적용할 서버 코드 관련 회의** * **다음 주 회의시간 조정** * **서버 코드 일정 회의** | | |
| **이번 주 문제점 및 해결방안** | **이도영**: Local에서는 작동하나 다른 컴퓨터에서 접속 시 서버에 연결이 안되는 오류 발생 | | |
| **다음 주 계획** | **이도영**: DB오류 수정, Astar관련 코드 학습, test클라이언트를 이용 패킷전송 확인  **정극훈**: 클라이언트 프레임워크 Particle, Instancing, Terrain 부분까지 완성  **김혁동**: FBX Loader 완성 | | |
| **비고** | Github 도둑들 주소 : <https://github.com/rmrgns/Thieves.git>  Github 졸업작품 회의록 주소: https://github.com/rmrgns/gameproject\_proceedings.git | | |