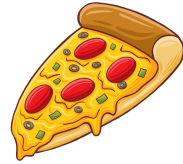


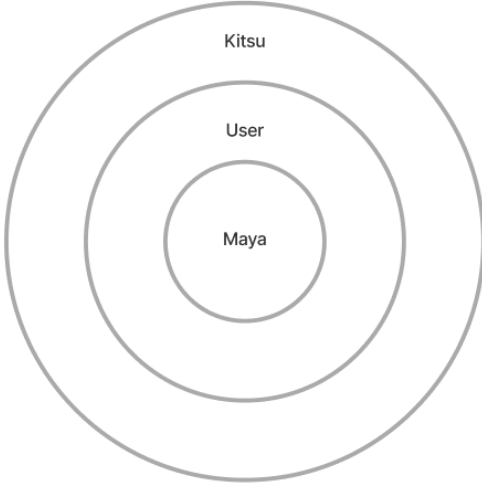
TEAM PIZZA 프로젝트 기획서



프로젝트명	Pizza API (Layout 팀을 위한 자동화 API 개발 및 솔루션)
프로젝트 시행목적	<p>3D MAYA 프로그램을 사용하는 Layout 팀을 위한 API 및 UI 제작.</p> <p>스크립트 클릭 한 번으로 데이터베이스(Kitsu)에 기록된 정보를 토대로 아티스트가 작업해야 할 작업물을 불러오거나 저장 및 퍼블리시 할 수 있고, 파이프라인을 구축함으로써 아티스트의 작업 속도 향상과 편의 증대</p>
문제점 및 제작 계기	<p>레이아웃 아티스트로서 Asset을 관리하고 Workflow를 구성하는 것은 매우 복잡한 작업입니다. 가져와야 하는 에셋이 너무 많아 여러 에셋을 관리하는 것이 부담스러울 수 있고, 이외에도 많은 수의 샷을 처리해야 하는 문제도 있습니다. 작업할 샷이 많기 때문에 모든 샷을 구성하고 관리하기 위한 효과적인 시스템을 갖추는 것이 중요합니다.</p> <p>각 샷에 대한 올바른 Asset을 사용할 수 있고, 해당 Asset이 적절한 폴더에 저장되는지 확인하는 것이 필요합니다.</p> <p>Maya Scene 작업을 효율적으로 하려면 Camera/Undistortion Image Sequence/Asset을 올바르게 가져오고 저장 및 퍼블리시 해야 합니다.</p> <p>또한 MatchMove 팀에서 작업한 모든 샷에 대해 Camera, Undistortion Image Sequence을 저장하는 것이 중요한데, 여러 폴더에 접근하여 Working, Output, Preview 파일을 따로 저장해야 하는 번거로움이 있어 제작하게 되었습니다.</p>

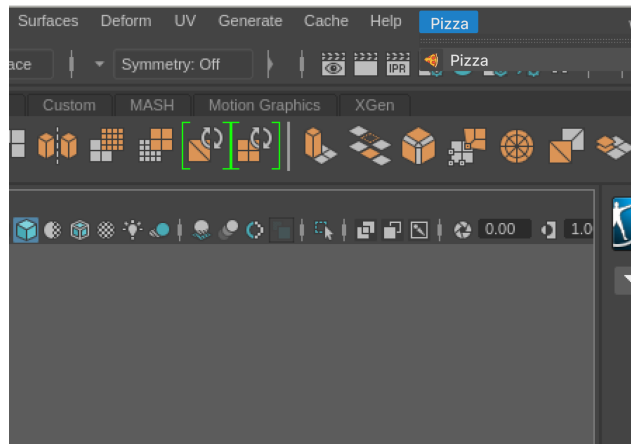
<p>해결 방안</p>	<p>Maya 메뉴 바에 있는 스크립트를 클릭하는 간단한 행동으로 쉽게 모든 Camera/Undistortion Image/Asset을 가져오거나 저장 및 퍼블리시 할 수 있는 기능을 구현합니다.</p> <p>Maya의 Python 스크립팅 언어를 사용하여 이러한 작업을 자동화하는 스크립트를 생성하여 시간을 절약하고 편의성을 증대시키고자 합니다.</p>
<p>프로젝트 특징</p>	<ul style="list-style-type: none"> ● 로깅 & 로그인 설정 및 관리 <ul style="list-style-type: none"> => 로그인 기능을 통해 나에게 assign된 작업만 진행 가능 => 동일한 샷에서 작업하는 팀원 간의 충돌 방지 ● 작업에 필요한 파일 전반(Camera/Undistortion Image/Asset) 중 원하는 파일만 선별하여 로드 가능 ● 작업 결과물 올바른 파일 형식과 버전으로 저장 및 퍼블리시 ● 작업한 샷의 프리뷰 영상 생성, 저장 및 퍼블리시
<p>사용자 편의성</p>	<ul style="list-style-type: none"> ➔ 파일을 로드하고 저장하기 위한 사용자 친화적인 인터페이스를 제공하여 파일 관리 프로세스 단순화 ➔ 사용자가 다른 프로젝트 파일 사이를 빠르게 전환할 수 있도록 하여 작업의 흐름 효율성 개선 ➔ 저장 또는 로드할 작업물을 선택하는 기능을 포함하여 작업자의 자유도를 높여 로드 및 퍼블리시 ➔ 썸네일을 보여주어 작업자가 한눈에 파악하기 좋음 ➔ 로드하기 전에도 파일 정보의 중요한 것들을 보여줌

<p>프로젝트 기대효과</p>	<ul style="list-style-type: none"> ➔ 파일 관리 프로세스를 단순화하여 레이아웃 팀의 워크플로우 효율성 크게 향상 ➔ 전반적으로 레이아웃 팀의 작업 흐름에 긍정적인 영향을 미치므로 보다 효율적이고 효과적으로 작업 가능 ➔ 저장 및 퍼블리시 할 때 수동 작업 시 발생할 수 있는 실수나 오류 감소
<p>프로젝트 예상결과</p>	<ul style="list-style-type: none"> ● 사용자가 Maya 메뉴 표시줄에서 스크립트를 클릭하는 간단한 행동으로 모든 Working, Output 파일 (Camera/Undistortion Image/Asset)을 쉽게 로드하거나 저장 및 퍼블리시 할 수 있는 완전한 기능 ● API에 대한 문서 및 사용 설명서 ● API에 대한 유닛테스트 및 로그 파일 기록
<p>API 구현 계획</p>	<p>1. API 제작 전 아래의 상황을 정의합니다.</p> <ul style="list-style-type: none"> - PM이 지정해준 Task가 존재하며, 이 Task의 Entity는 Asset이다. - Layout Asset에 필요한 임시 Asset들이 Casting 되어 있다. - 이 Asset은 사용되는 각각의 Shot에 Casting 되어 있다. - 이 Asset은 Sequence에 Casting 되어 있다. - 필요한 Output Type, Task Type, Folder Tree 구조 등이 모두 존재한다. - Camera 설정, Undistortion Image, Asset 파일 등 관리해야 하는 파일 유형 식별한다. - 파일 로드 및 저장과 같이 API가 지원해야 하는 작업을 결정한다.

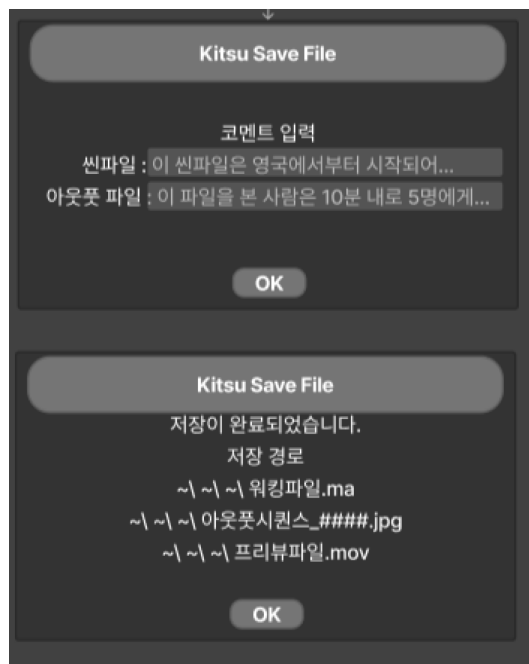
	<ul style="list-style-type: none"> - 작업이 완료된 working, output 파일의 정보를 저장한다. <p>2. API를 구현합니다.</p> <ul style="list-style-type: none"> - Python을 사용하여 필요한 작업을 구현하는 스크립트 제작 - 아티스트가 스크립트에 쉽게 액세스하고 필요한 작업을 실행할 수 있는 사용자 인터페이스 제작 - Maya의 Python API를 사용하여 필요한 파일 및 데이터와 상호 작용
API 실행구조	 <p>Kitsu에서 기록된 정보를 가져온다.</p> <p>-> ui에 띄운다</p> <p>-> 사용자의 선택에 따라 원하는 파일을 Load 후</p> <p>레이아웃 작업을 진행한다.</p> <p>-> 작업물을 저장하고 Kitsu에 작업 내역을 Publish 한다.</p>

UI PROTOTYPE

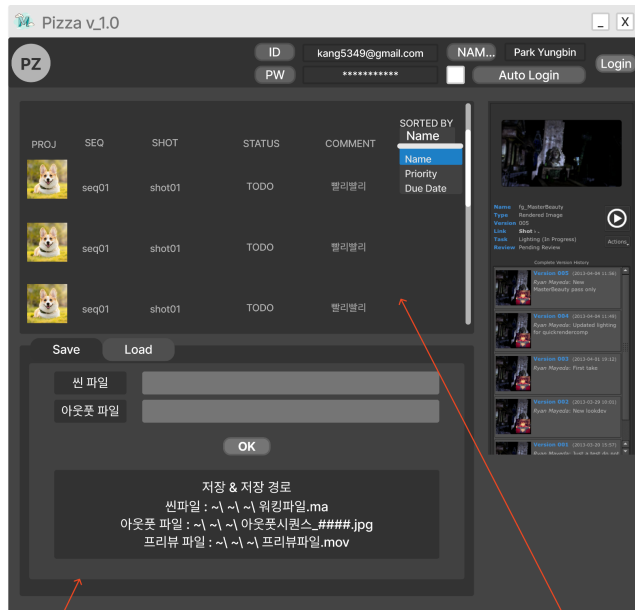
1. Menu Icon



2. Pop Up



3. Main Window



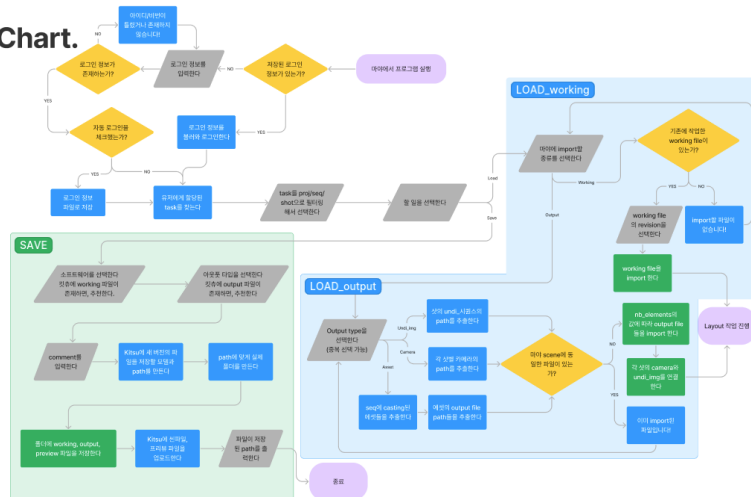
Save & Load 는 같은곳에서 모든것이 이루어짐

task 선택하기 전에는 Save / Load 창 안 뜸

CASTING LIST
Load할 파일 선택하기 전까지는 안 뜸

FLOW CHART

Flow Chart.



개발형태	python2.7
개발도구	Gazu, Pyside2, Maya2020
프로젝트 참여인원	<ul style="list-style-type: none"> ● 총 인원 : 5명 ● 팀장 : 조아현 ● 팀원 : 강경욱, 박정태, 이보름, 이의민
프로젝트 기간	2023년 2월 1일 ~ 3월 17일