1. Надо продумать загрузки моделей
2. Нужно научиться грузить текстурки

То есть если это в glut правда легко, юзаем его. Если это запарно, то просто на opengl быстро все напишем

И вообще внешний класс Application, который можно сделать синглтоном

1. Система юнитов

Точно нужен класс сцены

1. Загрузиться из файла, сохраниться в файл (?)
2. Добавить юнита, удалить юнита
3. Нарисоваться
4. Обновляться
5. Загрузиться – ну просто продумать формат.

Сцена должна рекурсивно вызывать для каждого юнита. Тогда у каждого юнита должна быть функция сереализации. Тоже получается в базовый интерфес. Но так он должен уметь и дереалсиерилизироваться.

1. Добавлять

std::map<std::string, std::shared\_ptr<Unit>> - структура данных

1. Просто рисует всех юнитов

Нужен класс юнитов – он будет полиморфным (с вирт функциями)

1. Констуктор/деструктор (деструктор вирутальный!!!)
2. Обновляться (Update) – виртуальная
3. Рисоваться (Draw) – виртуальная
4. (Де)сереиалиация

Конкреткные юниты:

class Person : public Unit

Есть поле position.

Можно взять его позицию. *vec2 get\_position()*.

Можно заставить говорить (оповестить об этом) *void Talk(Peolpe \*talker)*.

Если мы стоим рядом с ним, он выбрасывает фразу (какую именно он решает сам).

1. Рендер
2. Юниты, сцены, и продумать все что может понадобиться для итоговой игры

Application – сингтон. Объединяет сцены и рендер. Возможно сам класс рендера даже не нужен, можно все внутри этого класса.

1. Конструктор, деструктор
2. Start (принимает сцену)
3. Get/Set current scene
4. Timer
5. Create/Delete model (по факту пусть вся логика в конструкторах/деструкторах, но они приватные, а аппка просто возвращает указатели)
6. Draw model (поворот, перенос, скейл еще принимает)

Model:

Конструкторы:

1. Из массива вершин
2. Принимает файл модели
3. Принимает какой-то enum с конкретной тополгией

Деструктор

Возможность нацепить текстурку (добавить/удалить)

Хранит угол и ось поворота, позицию и скейл. Есть сеттеры/геттеры этой темы. Но возможно они не для всех:

С позициями надо продумать. Вероятно у нас все будет по клеточкам. Есть пердметы на 2 клеточки, большинство наверное на 1. Это может сильно упросить многое. Возможно объект вручную перемещать нельзя будет (только по высоте мб), а скейл будет относительно скейла кубика, где все рисуется, ну и дальше можно задать его. А сцена уже будет просчитывать абсолютный перенос и абсолютный скейл объекта.