**Архитектура**

Audio – сделать синглтон AudioManager

UI – понятно

Main Menu - понятно

Save System – лучше всего наверно через сохранение в Json. Пока сохранятся будут только пройденные уровни, но стоит продумать возможность сохранять количество звезд, полученных за каждый уровень.

Advert – надо разобраться с плагином от яндекса. Вроде на ютубе полно инструкций. Рекламу можно будет посмотреть при первой смерти, чтобы ожить.

Level Director – в этом классе будет метод FinishLevel(), который вызывается извне в случае успешного прохождения уровня. Это метод сохраняет пройденный уровень, проигрывает анимацию победы и тд. Если задача уровня – дойти до финиша, то метод вызывается триггером на финише. Если задача продержаться определенное время, то таймером. Если убить всех врагов, то счетчиком врагов и тд.

Player – принимает команды от инпута и перемещает героя/стреляет паутинкой/бросает предмет.

Input – считывает точку, на которую игрок кликнул мышкой/тыкнул пальцем. Управление мышкой и тачем лучше сделать независимо друг от друга разными классами, унаследованными от интерфейса IInputHandler (или абстрактного класса).

Platform – лучше сделать абстракцию IPlatform или PlatformBase, от которой наследовать конкретные типы платформ.

Enemy – сделать абстрактный класс EnemyBase и от него наследовать врагов.

Projectile – шипы, которыми стреляют враги и герой. Можно сделать два вида – перехватываемые и не перехватываемые. Для оптимизации спавн прожектайлов сделать через Object Pool.

Интерфейсы:

IDamagable – имеет метод GetDamage(). Вешается на игрока и все врагов, которых можно убить.

IDamager – наверно, необязательно его делать, но пусть будет

IGrabable – вешается на все, что игрок может схватить

Triggers – кнопки/рычаги, которые игрок может активировать