Отчёт

1) Декомпозиция задачи.  
Для большего понимания программы был создан файл FlappyDiagram.html благодаря сайту <https://app.diagrams.net>

2) Время, затраченное на выполнение, с детализацией.  
Задание было выполнено в пятницу вечером и доделано в субботу утром, всё остальное время было затрачено на ознакомление с Zenject <https://github.com/modesttree/Zenject> , применение этого плагина на практике и написание отчёта с диаграмой.

3) Чем вы руководствовались при выборе архитектуры для игры?

Опытом.

4) Как вы построили работу с UI, почему именно так?  
Я посчитал что было бы проще всего создать один синголтон объект UIManager который хранит ссылки на ui элементы. Обращаясь к менеджеру можно открывать закрывать ui елементы и менять score.  
UIManager был выбран потому что, из одного объекта проще всего организовывать какие объекты должны появится на экране, а какие должны исчезнуть.

5) Какими паттернами объектно ориентированного дизайна вы воспользовались

при разработке проекта?

Strategy, Singleton и что то похожее на Factory, можно было бы добавить State но я не стал.

6) Соблюдаются ли в вашем коде, принципы SOLID, если да, какие из них?

Есть соблюдения и нарушения принципа S (single responsibility principle), нарушения происходят в моих синглтон менеджерах, где преувеличены обязанности. Соблюдения происходит например в классе DataManger->Data.

Есть соблюдения принципа O (open-closed principle), например в Player->Bird можно переопределить функцию Die() тем самым изменить поведение класса. Или Есть ещё пример с ColliderObject и его наследниками.

Есть соблюдения принципа L (liskov substitution principle), думаю хорошим примером будут функции OnTriggerEnter2D() и OnCollisionEnter2D() у Player, где вызываются функции Trigger() и Collide().  
  
Не соблюдается принцип I(interface segregation principle), я посчитал, что не было определённой сложной задачи, для решения которой необходимо создать отдельные интерфейсы.

Есть соблюдения принципа D (dependency inversion principle), тк благодаря DI(dependency inject) и Zenject классы по большей части не имеют жёстко прописанных зависимостей от объектов других классов. Что облегчает будущую поддержку кода.

7) Какие изъяны/технический долг есть в этом проекте с точки зрения разработки?  
1 Некорректная работа со звуком, я обычно использую для этого сторонние плагины.

2 Если всё время нажимать на кнопку паузу то есть шанс что препятствие не появятся или появятся но на другом препятствие.  
3 Выход в меню осуществляется перезагрузкой текущей сцены.  
Это конечно не точно, но вроде это все изъяны.  
   
8) Если бы это был коммерческий проект с гораздо большим бюджетом времени

на разработку, что бы вы сделали по другому?  
Переписал бы игрока, для того что бы была возможность играть за любой объект типа Player,  
будь то птица или самолёт и возможно разными поведениями при тапе.  
Так же переписал бы спавнер объектов, было бы классно если бы объекты спавнились не просто рандомно, а например с определённой сложностью или поведением.  
Переписал бы весь ui и добавил обучение.  
Естественно постарался бы исправить ошибки.

9) Что вызвало у вас наибольшие сложности при разработке этого задания?  
Реализовать паузу в игре.