

Семинар #2

Внедрение зависимостей: DIP, IoC, DI

Формула #2

Нужно решить все ДЗ на ≥ 4 . Тогда итог рассчитывается с учетом приведения к единице:

```
result_estimation = math_round(0.2 * seminars_estimation + 0.8 * tasks_estimation)
```

- `seminars_estimation` – математическое среднее всех оценок за семинары. В конце семинаров будут мини-тесты на 3-5 вопросов по материалам лекций/семинаров. Тест не сдан или дано 0 правильных ответов - 0, \leq половины правильных ответов - 5, $>$ половины правильных ответов - 8, все ответы верны - 10.
- `tasks_estimation` – математическое среднее оценок за ДЗ. Тут будут заранее известные критерии и достаточно жесткая проверка на списывание. Претендующие на отл или авторы подозрительных работ будут их защищать. Примерная развесовка домашек: ДЗ-1 = 0.2, ДЗ-2 = 0.2, ДЗ-3 = 0.3, ДЗ-4 = 0.3

Итоговая оценка меньше 4 округляется вниз.

В случае пересдачи, вам нужно будет просто доделать все ДЗ за зимние каникулы.

Предложение от которого можно отказаться

На КПВ по iOS разработке очень хотят сделать геймтон, но им нужен общий сервер к которому они будут подключать свои клиенты. В качестве игры предполагается сделать что-то на тему бомбербена (кто-то вообще помнит такую игру?)

Основные фичи:

- Онлайн на 2-4 игрока
- Подключение по вебсокетам или signalr
- В идеале сделать несколько карт
- Как только все стены взорваны (или по таймеру), начинают спауниться рандомные бомбы
- Побеждает последний выживший
- Монстров добавлять не нужно
- Нужно добавить какую-то реализацию спектратор мода

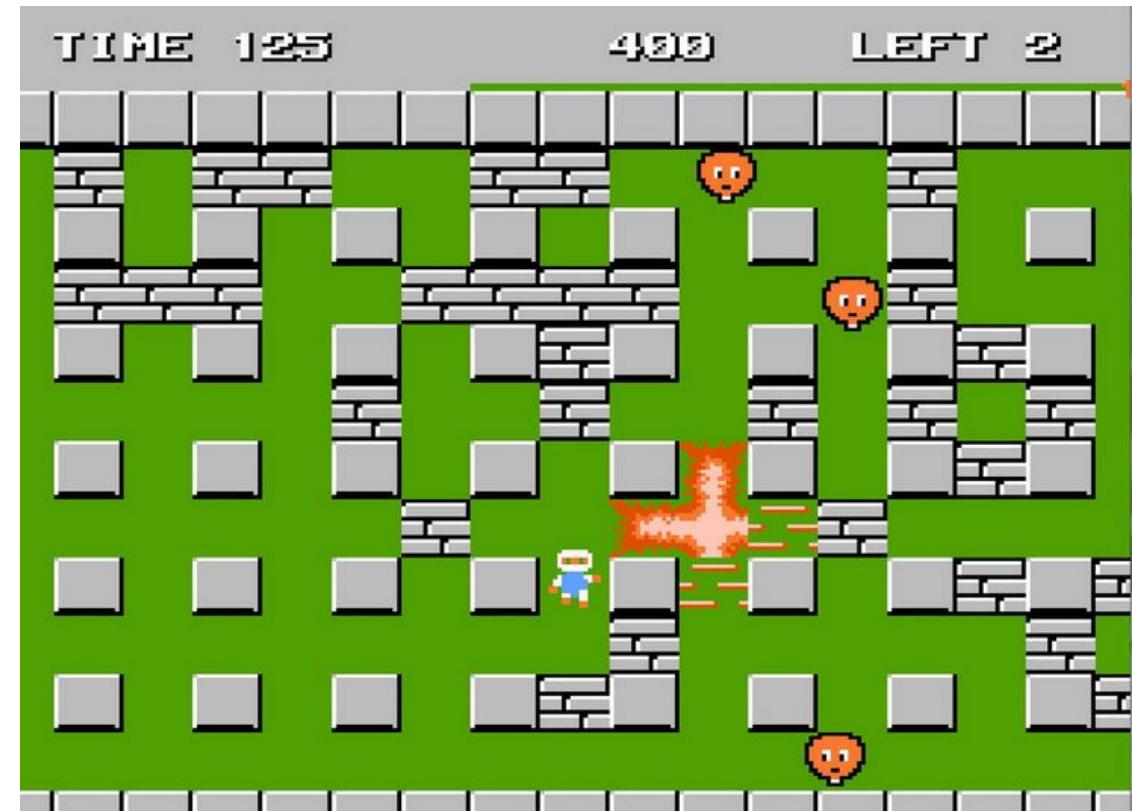
Это задание засчитается за любую ДЗ

Можно просрочить дедлайн ДЗ1 без штрафов

Можно взять этот проект в качестве курсового

Возможно у нас в декабре тоже будет геймтон :)

Срок разработки – до конца октября



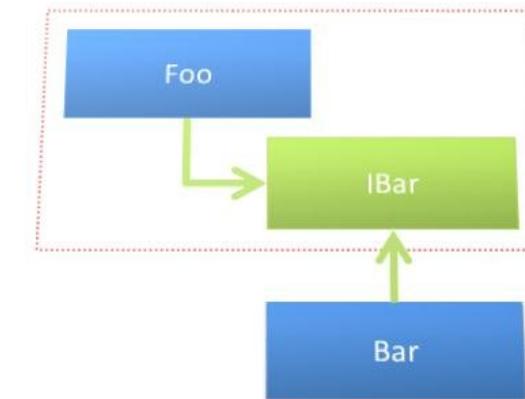
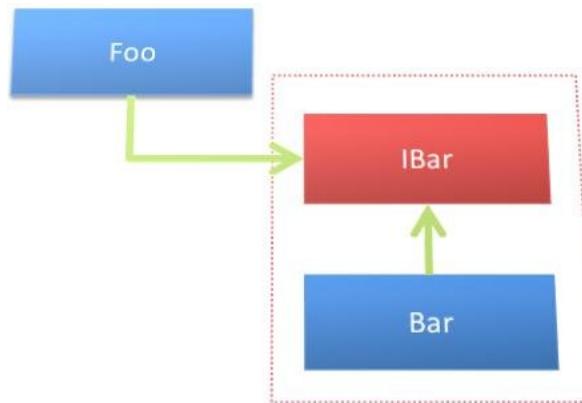
Dependency Inversion Principle (Инверсия зависимостей)

Формулировка:

- Модули верхних уровней не должны зависеть от модулей нижних уровней. Оба типа модулей должны зависеть от абстракций.
- Абстракции не должны зависеть от деталей. Детали должны зависеть от абстракций.

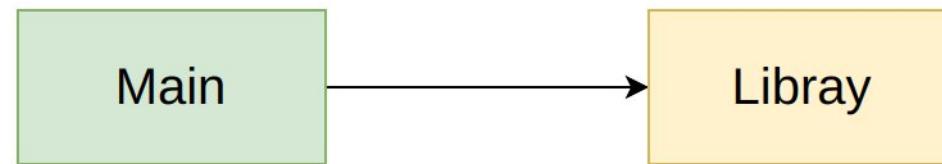
Менее академично:

- DIP: модули верхнего уровня (наша бизнес-логика) не зависят от модулей нижнего уровня (работа с бд, файловой системой и тд.)
- Абстракции вместо деталей: зависимости строятся на интерфейсах
- Детали зависят от абстракций: реализации подчиняются контрактам интерфейсов



Inversion of Control (Инверсия управления)

При инверсии управления поток управления программы контролируется некой внешней по отношению к коду сущностью. По сути это просто способ автоматизация внедрения зависимостей (в контексте внедрения зависимостей).

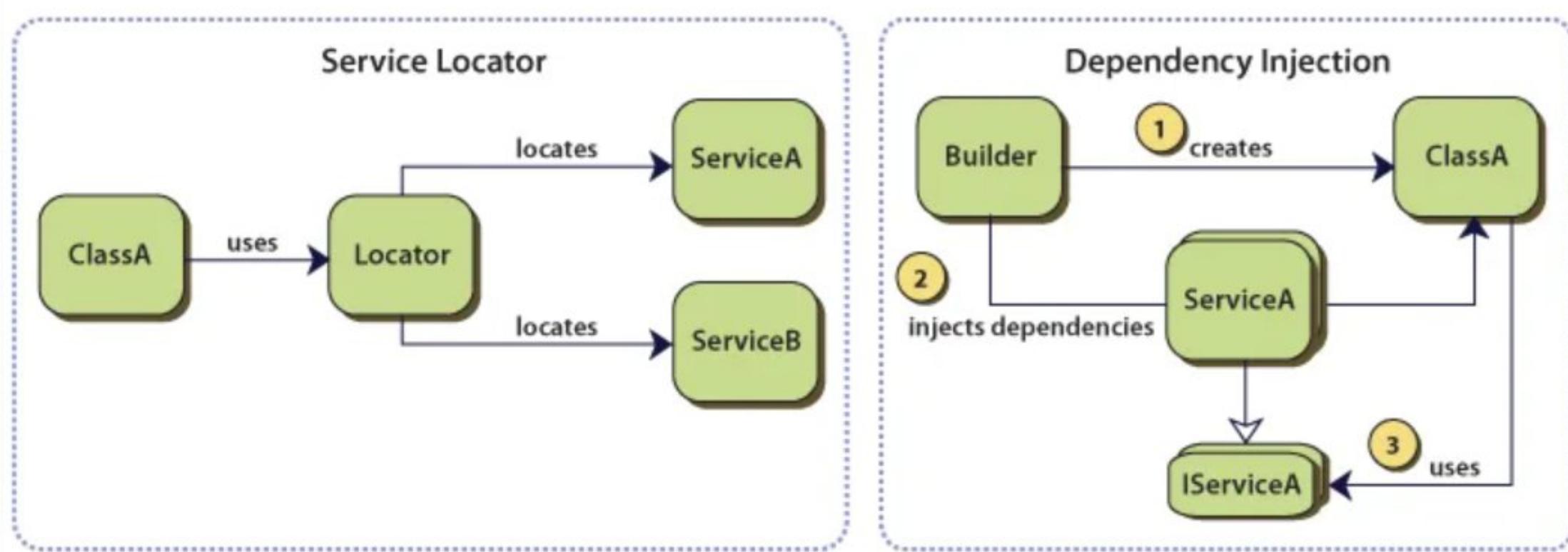


Прямой поток управления



Инверсия потока управления

Service Locator vs Dependency Injection



Дополнительные материалы

Для понимания понятий DIP, IoC, DI:

<https://habr.com/ru/articles/321344/>