## **Сквозной проект. Семинары 10-12**

**Семинар 10,11 и 12** представляют из себя сквозной проект.

**Цель задания**: научиться проектировать облачное приложение.

**Задание:** Спроектировать облачное приложение с интерфейсами в браузере и нативными интерфейсами в мобильных устройствах.

* Необходимо спроектировать облачный сервис домашнего робота пылесоса для уборки помещений.
* Результатом должны быть: компоненты интерфейсов, доменная модель, Use case, компонентные диаграммы, EDR, API контракты, тестовые сценарии.
* Выполняется постепенно, в течении 3-х семинаров (блоков).

### **Блок 1 / Семинар 10**

### **Задание:**

a. Спроектировать пользовательский интерфейс (web-SPA, native mobile), основные компоненты (подключение робота, управление помещениями, расписание работы, сервисное обслуживание робота, история уборок),<https://www.figma.com/> или<https://app.diagrams.net/>.

b. Спроектировать доменную модель, в виде текста Домен – атрибуты.

c. Спроектировать сценарии (Use case)(подключение, выбор помещения, программы уборки, настройка расписания, просмотр статистики..), в виде Актор – Прецедент (из первой лекции).

d. Спроектировать слой API Gateway (mobile, web), сформировать REST запросы: GET, POST, PUT, DELETE (<https://swagger.io>).

* (дополнительно, по желанию) Разработать REST контракты API между компонентами и сгенерировавать (автоматически на ресурсе<https://swagger.io>) код на разных языках программирования.

e. Спроектировать компоненты бизнес-логики и связать их API Gateway с применением паттерна BFF<https://app.diagrams.net/>.

f. Определить состав информации для кеширования на уровне приложения пользователя, API Gateway, уровня бизнес-логики и уровня репозитория. Список.

g. Спроектировать ER модель (<https://www.dbdesigner.net/>), запросы в БД и уровень хранения данных (СУБД).

### **Домашнее задание:**

Доработать пункты задания Блока 1: a, b, c, d, e, f, g.

Инструменты:

* [https://www.figma.com/](https://www.figma.com/**)
* [https://app.diagrams.net/](https://app.diagrams.net/**)
* [https://www.dbdesigner.net/](https://www.dbdesigner.net/**)
* [**https://swagger.io**](https://swagger.io)

### **Блок 2 / Семинар 11**

### **Задание:**

h. Спроектировать компонент подключения по API к производителю роботов-пылесосов, для обновления прошивок и передачи диагностических данных<https://app.diagrams.net/>.

i. Спроектировать адаптеры интерфейсов в виде компонент (подключение к БД, шлюзы к интерфейсу, шлюзы к третьим источникам)<https://app.diagrams.net/>.

j. Спроектировать компонент внешнего конфигурирования для настройки компонент бизнес-логики (параметры конкретного робота)<https://app.diagrams.net/>.

k. Спроектировать и имплементировать компонент самодиагностики (Self-checking). Который собирает статус работоспоспособности и времени запроса на каждый компонент уровня API Gateway, бизнес-логики и слоя управления данными. Включая технологический интерфейс для мониторинга<https://grafana.com/> или<https://prometheus.io/> .

l. Выбрать метрики для наблюдения за инфраструктурой на которой будет развёрнуто решение (процессор, память, диск).

### **Домашнее задание:**

Доработать пункты задания h, i, j, k, l.

Инструменты:

* <https://app.diagrams.net/>
* <https://grafana.com/>
* <https://prometheus.io/>

### **Блок 3 / Семинар 12**

### **Задание:**

m. Разработать Unit Test для сервисов бизнес-логики.

n. Предложить и разработать E2E тесты<https://www.postman.com/> или аналоги.

o. Предложить и разработать приёмочные тесты (UAT) для сценариев Use case<https://www.gurock.com/testrail/> или аналоги.

p. Предложить тесты для регрессионного тестирования.

### **Домашнее задание:**

Доработать пункты задания m, n, o, p.

Инструменты:

* <https://app.diagrams.net/>
* <https://www.gurock.com/testrail/>
* <https://www.postman.com/>