

## Тестовое задание

В мире clojure не приняты фреймворки, каждый инженер собирает их сознательно под себя и под задачу из библиотек. Необходимо собрать свой полный Clojure/ClojureScript стек с PostgreSQL и реализовать на нем CRUD пациента (dataset описан ниже). На UI должны быть возможности: просмотреть список пациентов, создать, удалить, изменить пациента. Желательно с функциональными тестами и CI (сборка по коммиту). Подготовить приложение к развертыванию в K8s.

Dataset:

- ФИО пациента
- Пол
- Дата рождения
- Адрес
- Номер полиса ОМС

### Дополнительные ограничения (опционально).

REPL-driven разработка в **emacs** или **vim** будет дополнительным плюсом (как-то так <http://www.parens-of-the-dead.com/>)

Все что описано выше, это исходные ограничения, все остальные решения/выбор на каждом этапе создания приложения тебе придется принимать самому!

### Демо

По завершении проекта, необходимо презентовать **демо** работающего приложения, как ты его разрабатывал в repl, код, тесты и **обосновать** принятые ключевые решения в рамках создания приложения.

### Возможность получить консультацию!

На протяжении всего проекта, у тебя есть **1 консультация** (созвон/15-20 минут) с специалистом из Health Samurai. Сам решай когда и как ей воспользоваться!