НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ

«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ»

ФАКУЛЬТЕТ ІНФОРМАТИКИ ТА ОБЧИСЛЮВАЛЬНОЇ ТЕХНІКИ

КАФЕДРА ОБЧИСЛЮВАЛЬНОЇ ТЕХНІКИ

**РОЗРАХУНКОВО-ГРАФІЧНА РОБОТА**

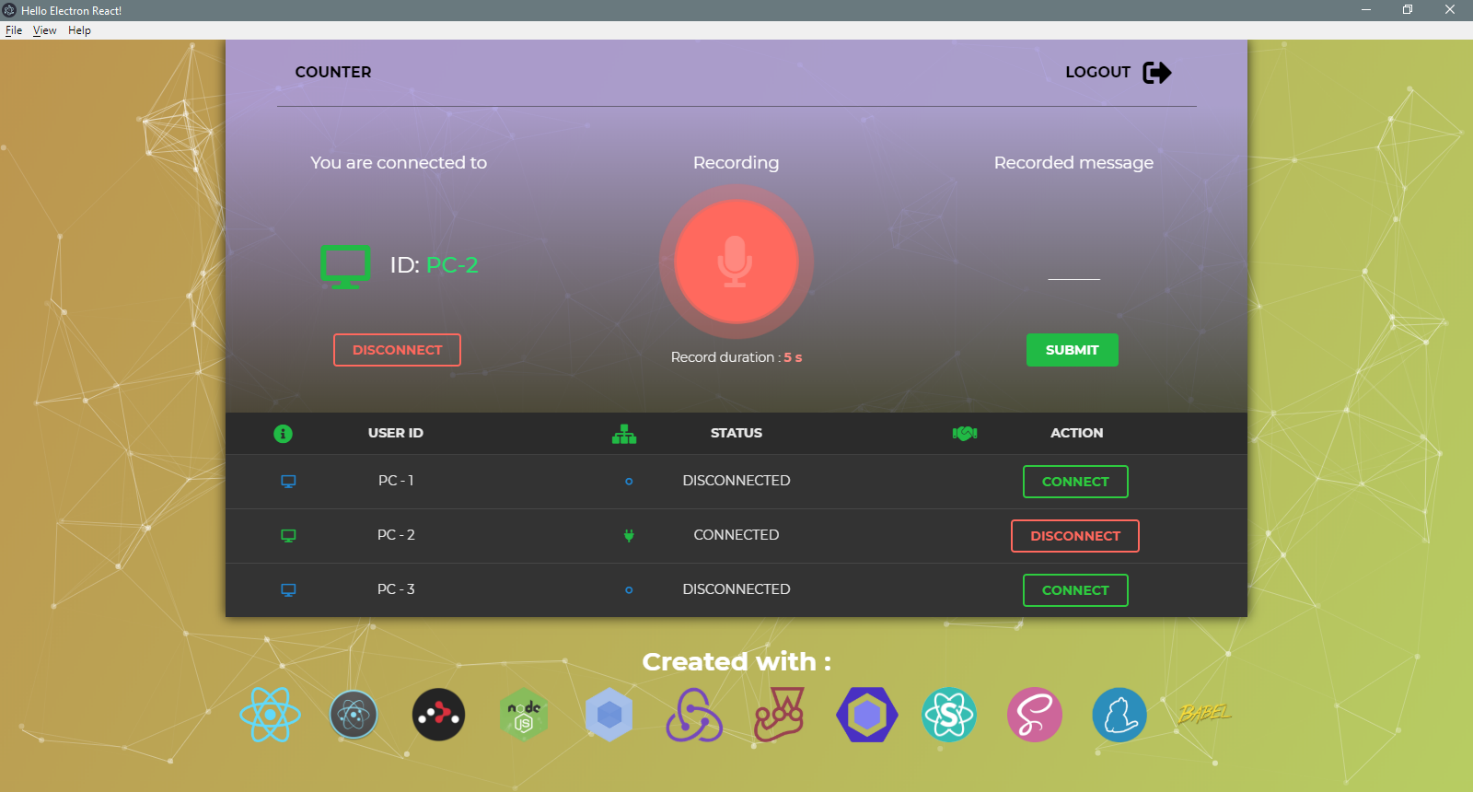
з курсу «ІНТЕГРАЦІЙНІ ПРОГРАМНІ СИСТЕМИ »

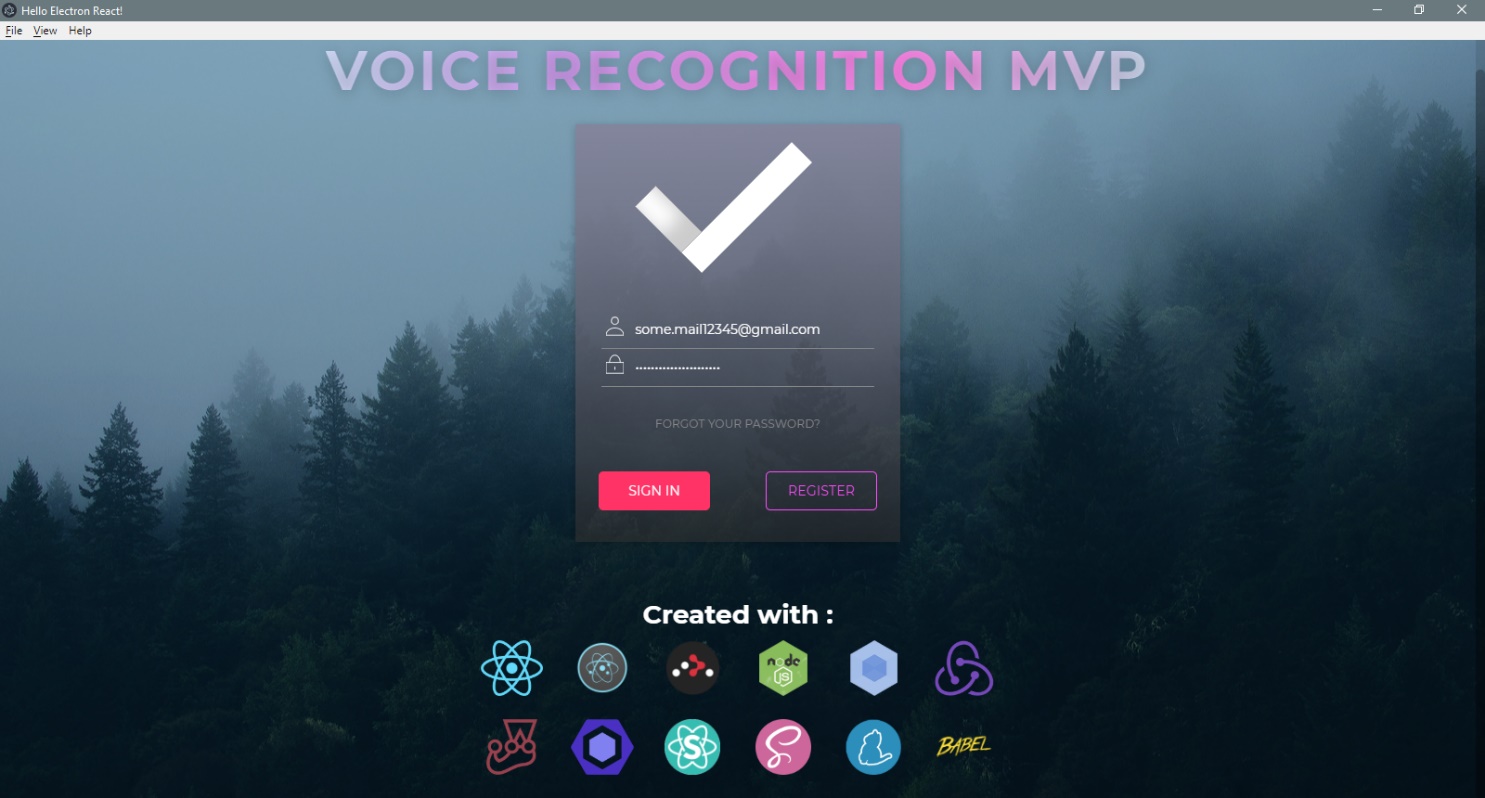
|  |
| --- |
| **Виконали**:  студенти IV курсу  ФІОТ гр. ІО-52  **Бригада**  *the\_drifter* в складі:  Василенко Д.Є.  Прасолов А.A.  Віцюк Р.В  Маневич В.А. |

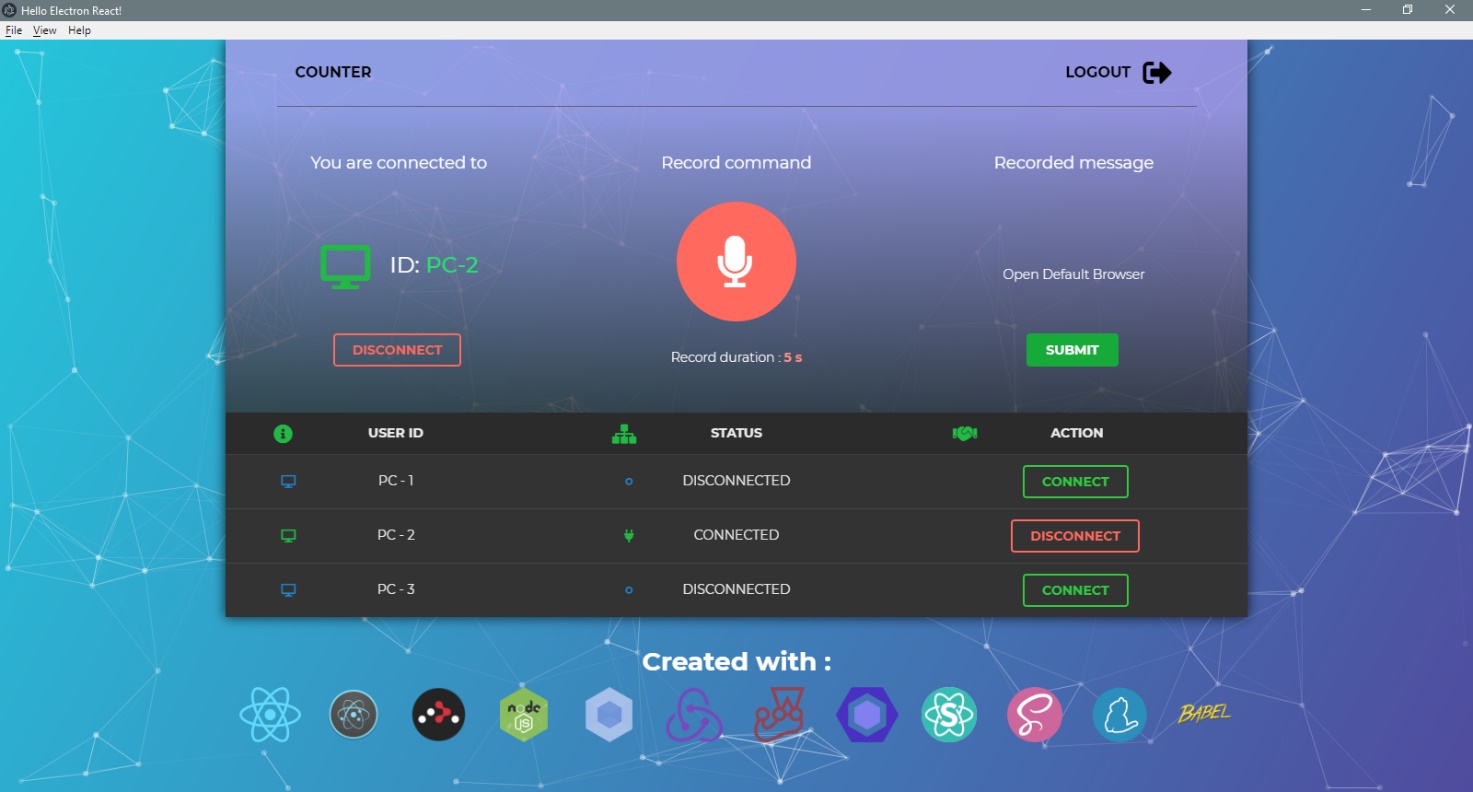
**Київ 2018**

1. Опис проекту

Розроблений нами проект є голосовим асистентом що може виконувати певні голосові команди







Скріншоти сервісу

Серверна чатина складаєтся з 2 мікросервісів що базуються на AWS EC2 з пайплайном на Circle CI. 1 мікросервіс – stateless jwt token аутентифікація написана на Clojure з використанням фреймворку Luminus. 2 мікросервіс, що є певним middleware для Google Speech api, на Scala з выкористання Akka http. Обидва мікросервіси реалізуют Onion архітектуру.

На Aws EC2 встановлений t2 micro instance. З встановленим Docker та розвернутими контейнерами до яких відкритий доступ з портів 80, 8080.

2. Система атоматичної збірки. Sbt

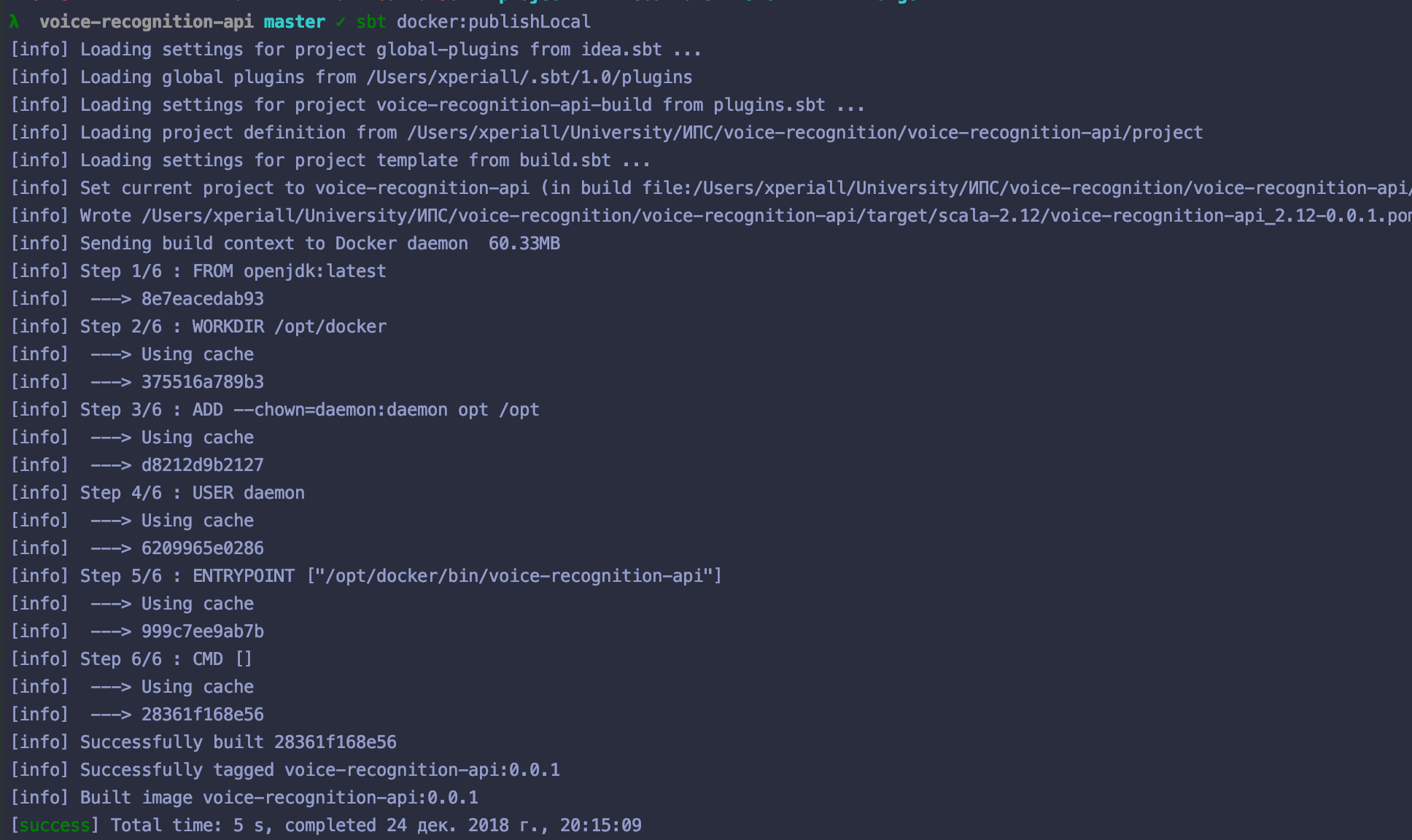
**sbt** — інструмент складання з відкритим кодом для проектів мовами програмування Scala і Java, подібний до Maven чи Gradle.

Основними властивостями є:

* вбудована підтримка компілювання мови Скала та інтеграція з багатьма інструментами тестування для Scala
* опис складання написаний на Scala з використаннями DSL
* Керування залежностями за допомогою Ivy (який підтримує репозитарії у форматі Maven)

sbt є де-факто стандартним інструментом складання у спільноті Scala.

Результат docker build.



3. Cервер безперервної інтеграції. Circle CI

Як виявилося, термін «continuous integration» досить старий. Він був введений Мартіном Фаулером (Martin Fowler) у 2000-му році і викладений у статті «Continuous Integration». Це частина процесу розробки, в якій розробляється проект збирається / тестується в різних середовищах виконання автоматично і безперервно. Задумувалася дана методика для найбільш швидкого виявлення помилок / протиріч інтеграції проекту, а отже зниження витрат на наступні простої.

Принцип досить простий: на окремій машині працює якась служба, в обов'язки якої входить отримання вихідного коду проекту, його збірка, тестування, логування, а також можливість надати для аналізу дані виконання перерахованих операцій.

Звичайно ж, безкоштовний сир буває тільки в мишоловці і за зручність необхідно платити: виділити окремий сервер і підтримувати його в робочому стані, забезпечити наявність необхідних програмних комплексів, налаштувати середовища виконання, робити резервні копії даних і т.д. Все це вимагає чимало часу і ресурсів. І цілком логічним здається можливість делегувати цю відповідальність на сторонні сервіси. От якраз таким і є circle-ci - «хостинг безперервної інтеграції для open source співтовариства». Настав час подивитися на нього ближче.

Circle-ci підтримує безліч мов програмування серед яких є і ruby ​​(що не дивно, тому що спочатку він розроблявся для ruby-проектів). Почати користуватися сервісом дуже просто. Потрібно всього лише зробити кілька кроків, які детально описані у власному Гайд проекту. Я лише опишу процес в цілому.

Circle CI виконує :

1. Збірку контейнерів
2. Тести
3. Деплой на Heroku мікросервісу аутентифікації

4. Експоненціальна витримка

Для вирішення задачі раптового зникнення з’єднання з базою даних було вирішено задачу експоненціальної витримки (для того, щоб не потрібно було виконувати перезапуск сервер. Вона автоматично обробляється драйвером для Postgresql для Clojure.

Формула витримки складетеся лише з двох параметрів. Мінімальний час у степені кількості спроб. Отриманий графік можна переглянути нижче