**Игра “Города” (Cities game)**

**Описание:**

Приложение представляет собой html-страницу, на которой располагаются карта, данные пользователя и компьютера, кнопки управления. При вводе игроком названия города, он автоматически отображается на карте. Победитель определяется в соответствие с [правилами игры](https://ru.wikipedia.org/wiki/%D0%93%D0%BE%D1%80%D0%BE%D0%B4%D0%B0_(%D0%B8%D0%B3%D1%80%D0%B0)). Игрок имеет возможность сдаться или начать игру сначала.

**Репозиторий на Github:**

Перед началом выполнения задания необходимо создать папку с названием *“Task 9 (Cities game)”* в вашем репозитории на GitHub и добавить [меня](https://github.com/AlexMozheiko) как *collaborator-а*.  
  
**Условия реализации:**

* перед началом игры система предлагает ввести имя игрока. Если пользователь пропускает данный шаг, то система автоматически присваивает ему имя (например, *player-1*).
* система случайным образом определяет, кто ходит первым: пользователь или компьютер.
* в качестве карты необходимо использовать *Google Maps* или *Yandex Maps*.
* карта должна располагаться посередине страницы и быть адаптивной.
* в верхнем левом углу располагаются данные игрока: имя, поле для ввода названия городов с кнопкой подтверждения, а ниже список уже введенных пользователем городов.
* в верхнем правом углу располагаются данные компьютера: строка *“Компьютер”*, а под ним список уже введенных компьютером городов.
* вверху страницы посередине располагаются кнопки управления: *“Завершить игру”* (для пользователя), *“Начать игру сначала”*.
* после того, как пользователь ввел название города и нажал на кнопку подтверждения, карта автоматически перемещается к указанному городу, масштаб уменьшается/увеличивается (по необходимости, чтобы были видны границы города) и устанавливается маркер в центр указанного города. Аналогичные действия происходят при ходе компьютера.
* в случае, если пользователь ввел несуществующий город (или уже названный), система показывает диалоговое окно с сообщением о том, что данный город не существует (или он уже был назван) и просит пользователя ввести другой город.
* список введенных пользователем/компьютером городов должен быть “скроллящимся”, при этом остальной контент страницы должен оставаться на своих местах.
* в случае победы игрока/компьютера система выводит диалоговое окно с указанием имени победителя и предложением начать игру заново.
* при клике на маркер, система отображает информационное окно ([Map infowindow](https://developers.google.com/maps/documentation/javascript/examples/infowindow-simple)) с названием данного города.
* базу городов для компьютера можно взять, например, [отсюда](https://github.com/lutangar/cities.json).

**Задания повышенной сложности:**

* Добавить ограничение по времени (например, 1 минуту) на каждый ход игрока/компьютера. По истечении времени, если не введен город, то игроку/компьютеру засчитывается поражение.
* Реализовать возможность голосового ввода городов для пользователя. Для этого использовать [Web Speech API](https://developer.mozilla.org/en-US/docs/Web/API/Web_Speech_API) (пример использования есть [тут](https://mdn.github.io/web-speech-api/speech-color-changer/)).

**Используемые технологии:**

* HTML5
* CSS3 (Необходимо использовать [SASS-препроцессор](https://sass-lang.com/guide), а также [CSS variables](https://developer.mozilla.org/en-US/docs/Web/CSS/Using_CSS_variables), [Grids](https://css-tricks.com/snippets/css/complete-guide-grid/) и [Flexboxes](https://css-tricks.com/snippets/css/a-guide-to-flexbox/). Имена классов задавать по [БЭМ-методологии](https://en.bem.info/methodology/css/)).
* JS (Использовать “vanilla” JS, [Google Maps API](https://developers.google.com/maps/documentation/javascript/tutorial) или [Yandex Maps API](https://tech.yandex.com/maps/jsapi/). Необходимо использовать возможности из современных спецификаций ES6, ES7, ES8, а также оформить код по данному [style guide](https://github.com/airbnb/javascript)).