



“Единственное, чего люди не забывают, это нечто нераскрытое. Ничто не живет так долго, как неразгаданная тайна.”

— [\(С\) Джон Фаулз.](#)

Тестовое задание на вакансию стажёр веб-разработчик.

Шифр Цезаря

Шифр Цезаря один из наиболее древнейших известных шифров. Схема шифрования очень проста - используется сдвиг буквы алфавита на фиксированное число позиций. Используемое преобразование обычно обозначают как ROTN, где N - сдвиг, ROT - сокращение от слова ROTATE, в данном случае “циклический сдвиг”.

Алфавит действительно закикливается, то есть буквы в конце алфавита преобразуются в буквы начала алфавита. Например, обозначение ROT2 обозначает сдвиг на 2 позиции, то есть “а” превращается в “в”, “б” в “г”, и так далее, и в конце “ю” превращается в “а” а “я” - в “б”. Число разных преобразований конечно и зависит от длины алфавита. Для русского языка возможно 32 разных преобразования (преобразования ROT0 и ROT33 сохраняют исходный текст, а дальше начинаются уже повторения). В связи с этим шифр является крайне слабым и исходный текст можно восстановить просто проверив все возможные преобразования.

Задача

Спроектировать и разработать одностраничное веб приложение позволяющее, шифровать и расшифровывать тексты *только* на английском языке используя шифр Цезаря.

Основные характеристики веб-приложения:

- состоит из клиентской и серверной частей, шифрование и дешифровка должны проходить на стороне

сервера;

- для реализации этой части допускается использование любого языка программирования (Python, Ruby, Java, C#, Php);
- обмен данными между клиентской и серверной частями происходит с использованием формата JSON;
- для хранения данных на сервере используется какое-либо хранилище: реляционная/нереляционная БД, простой файл, оперативная память или др.;
- клиентская часть по возможности использует любой JavaScript-фреймворк на ваш вкус: AngularJS будет плюсом.

Клиентская часть

Пользователь взаимодействует с приложением посредством единственной страницы, через которую подаются входные параметры и отображаются выходные данные.

Макет страницы может иметь вид как на рисунке ниже:

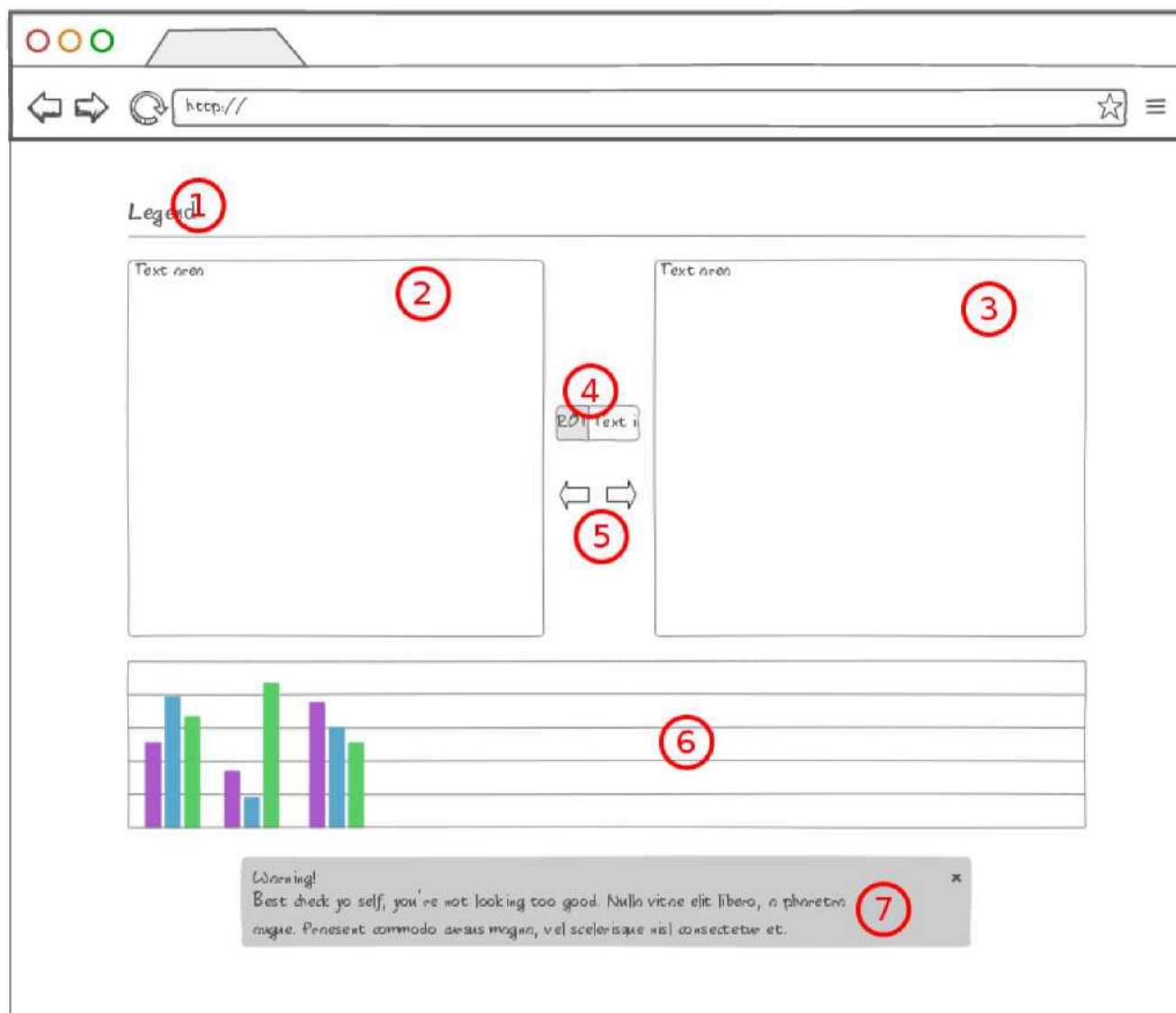


Figure 1: Макет приложения

Цифрами на изображении указаны следующие элементы управления:

1. Зона для заголовка и короткого описания
2. Текстовое поле для ввода текста
3. Текстовое поле для отображения результата (только чтение)
4. Текстовое поле для указания смещения
5. Кнопки позволяющие зашифровать и расшифровать текст
6. Место для частотной диаграммы
7. Место для вывода информационного сообщения с предположением нужного смещения.

Основной сценарий использования

Пользователь вводит текст в поле 2. Выбирает смещение, и нажимает кнопку расшифровать или зашифровать. В окошке 3 появляется результат.

Дополнительные сценарии

- После ввода текста в поле, в области 6 появляется частотная диаграмма для заданного текста, которая отображает частоту каждого символа в тексте.
- После ввода текста в поле, программа пытается определить является ли данный текст зашифрованным кодом или нет. Если является, она отображает сообщение с попыткой угадать смещение.

Требования к оформлению и критерии оценки

Программа должна быть рабочей. Если присланный код не работает, или его не удаётся запустить из-за отсутствия необходимых инструкций. Тестовое задание не рассматривается.

Код должен соответствовать общепринятым стилевым и организационным стандартам, действующим для выбранных вами языков и технологий.

По возможности код должен сопровождаться разумными комментариями, юнит-тестами, прочими инструкциями.

Оцениваться тестовое задание будет по следующим критериям:

- За полностью рабочий основной сценарий - максимум 3 бала;
- За каждый полностью рабочий дополнительный - максимум 2 бала;
- За качество кода клиентской части, внешний вид, соответствие общепринятым стандартам - максимум 2 бала;
- За качество кода серверной части и проектирование архитектуры - максимум 2 бала;
- За креативность -1 бал.

Для прохождения отбора соискатель должен набрать 7 баллов.