Guten Tag Frau Shiryagina Salini

herzlichen Dank für Ihre Nachricht.

Sehr gerne sende ich Ihnen heute kurz vor dem Wochenende noch die Aufgabenstellung zu. Bitte lesen Sie sich alles sorgfältig durch, bei Fragen können Sie jederzeit gerne auf mich zukommen.

Sobald Sie wissen, wann es Ihnen für eine Präsentation passen würde, finden wir gerne gemeinsam einen Termin, der Ihnen passt.

**Scenario**

The following scenario is fictional but also applicable to a typical customer scenario of PRODYNA.

PRODYNA's customer is a rapidly growing textile cleaning company named "Clean It! AG". Clean It! AG asks for an application to handle their daily business in their local cleaning shops, distributed over ten cities in Switzerland.

New customers have to be registered. Textiles get received, cleaned and returned. A cleaning order/job has to be linked to a customer account. Employees of Clean It! AG should have a user account to manage the cleaning jobs. A local cleaning shop manager needs to overview all orders and their processing.

Should you have questions regarding the scenario, feel free to make assumptions or ask us directly.

**Your Task and its Delimitations**

You are asked to develop a client-server application in the scenario described above. Implement two use cases (e.g. "reception of a cleaned textile", "order overview for a local shop manager"). Please note that we do not want you to implement the whole application! We also do not expect you to invest more than 10 hours in implementing.

Present your solution as you would in front of a real customer who ordered your services. Show the two use cases you have implemented.

A live presentation of your application is preferred. However, we also accept a prerecorded video or screenshots as an alternative. To reduce the effort on your side, you can simulate technical dependencies. For example, you are allowed to hard-code the state of a customer object instead of reading it from a database.

Plan to have the following audience as part of your customer presentation:

·         The CEO, owner and purchaser of the project

·         The head of the relatively small IT department of Clean It! AG

Be prepared that the head of IT might ask you why you didn't choose his personal favorite (nest.js) for the implementation.

Besides the "official" customer presentation, we would like you to show your program code to us.

**Technical Requirements**

For the front-end, we ask you to use Angular and Angular Material. Asses if you want to use NgRx and prepare to explain your decision.

Find a fitting use case to utilize the RxJs operator "debounceTime". Find a second use case in which you adapt another RxJs operator free of choice.

For the back-end, you are free to use one of the following programming languages/platforms:

·         TypeScript / node.js

·         Java / Spring Boot

·         C# / .NET

·         Go (Lang)

Optional: Demonstrate that the application runs in at least one Docker container.

Auch wenn die Aufgabenstellung in Englisch ist, erwarten wir nicht, dass du die Aufgabe in Englisch präsentierst.

Falls du Fragen hast oder etwas unklar ist, kannst du uns jederzeit gerne fragen – oder Annahmen treffen.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Sehr gerne sende ich Ihnen heute kurz vor dem Wochenende noch die Aufgabenstellung zu. Bitte lesen Sie sich alles sorgfältig durch, bei Fragen können Sie jederzeit gerne auf mich zukommen.

Sobald Sie wissen, wann es Ihnen für eine Präsentation passen würde, finden wir gerne gemeinsam einen Termin, der Ihnen passt.

Сценарий

Следующий сценарий является вымышленным, но он также применим к типичному сценарию клиента PRODYNA.

Клиентом PRODYNA является быстрорастущая компания по чистке текстиля под названием «Clean It! AG». Убери это! AG запрашивает приложение для ведения повседневных дел в местных клининговых мастерских, расположенных в десяти городах Швейцарии.

Новые клиенты должны быть зарегистрированы. Текстиль получают, чистят и возвращают. Заказ/задание на уборку должно быть связано с учетной записью клиента. Сотрудники Clean It! AG должна иметь учетную запись пользователя для управления заданиями по очистке. Местный менеджер клининговой мастерской должен контролировать все заказы и их обработку.

Если у вас есть вопросы относительно сценария, не стесняйтесь делать предположения или задавайте их нам напрямую.

Ваша задача и ее ограничения

Вам предлагается разработать клиент-серверное приложение по описанному выше сценарию. Реализуйте два варианта использования (например, «прием выстиранного текстиля», «обзор заказов для местного менеджера магазина»). Обратите внимание, что мы не хотим, чтобы вы реализовывали приложение целиком! Мы также не ожидаем, что вы потратите более 10 часов на внедрение.

Представьте свое решение так, как если бы это было перед реальным клиентом, который заказал ваши услуги. Покажите два варианта использования, которые вы реализовали.

Предпочтительна живая презентация вашего приложения. Однако в качестве альтернативы мы также принимаем предварительно записанное видео или скриншоты. Чтобы уменьшить усилия с вашей стороны, вы можете смоделировать технические зависимости. Например, вы можете жестко запрограммировать состояние объекта клиента вместо того, чтобы считывать его из базы данных.

Запланируйте участие следующей аудитории в рамках презентации для клиентов:

• Генеральный директор, владелец и покупатель проекта

• Руководитель относительно небольшого ИТ-отдела Clean It! АГ

Будьте готовы к тому, что руководитель отдела ИТ может спросить вас, почему вы не выбрали для реализации его личный фаворит (nest.js).

Помимо «официальной» презентации для клиентов, мы хотели бы, чтобы вы показали нам код вашей программы.

Технические требования

Для внешнего интерфейса мы просим вас использовать Angular и Angular Material. Оцените, хотите ли вы использовать NgRx, и приготовьтесь объяснить свое решение.

Найдите подходящий вариант использования оператора RxJs «debounceTime». Найдите второй вариант использования, в котором вы адаптируете другой оператор RxJs по своему выбору.

Для серверной части вы можете использовать один из следующих языков/платформ программирования:

• TypeScript / node.js

• Java/весенняя загрузка

• С#/.NET

• Идти (язык)

Необязательно: продемонстрируйте, что приложение работает как минимум в одном контейнере Docker.

Auch wenn die Aufgabenstellung in English ist, erwarten wir nicht, dass du die Aufgabe in English präsentierst.

Falls du Fragen имеет некий oder etwas unklar ist, kannst du uns jederzeit gerne fragen – oder Annahmen treffen.