Dejamobile Take Home

We're excited to have your interest in Dejamobile, and we appreciate you taking the time to complete our take home.

About the technical interview

You will give a presentation to 2 or more Dejamobile team members on a technical piece of work that you have previously produced and delivered yourself. ● We prefer a project that will allow you to demonstrate your technical skills, your problem solving skills, and your creativity. The project will ideally showcase your work across different modules you chose to work on (front end, back end). ● Please share the code with the interviewers at least 24h before the interview. ● You can choose to prepare a slides deck to present your work. The presentation will be preferably in English (slides + commentary) and should talk about: ● The technical architecture and the key decisions made. ● Deep dive into one of the components of the system

Take Home project

Important warning

The project described below is way too broad to be executed properly. You will need to decide what you do / do not do, and how well. Some components can be replaced by simulators with dummy data.

Some developers may want to focus on the code, others may want to deliver simpler code but spend more time on deploying an actual, working application that they can demo. There is no right or wrong answer, just a set of decisions that will allow you to demonstrate your strengths.

Technical guidelines

There are no technical guidelines. You can choose whatever technical stack you want.

However, here is a list of stack components & patterns used at Dejamobile:

- Database: MySQL, Cassandra
- Message broker: Kafka, NATS
- Programing language: Golang, Java
- Architecture: Microservices
- Paradigm: Event Driven, Domain Driven Design

Business problem

Dejamobile aim to provide a platform to enable contactless payments using NFC technology. This platform should be used by **multiple customers** (basically banks). The solution of Dejamobile is composed of a mobile **SDK** integrated in the customer application and of a set of **backend services** that formed the Cloud Based Payment solution.

The goal of this platform is to enable the contactless payment on a electronic/digitized card that will be **generated** by the backend and **hosted** on the user mobile.

The Cloud Based Payment solution is connected to entities that are able to create new digitized cards for a given Primary Account Number of a real card. Those entities are called "Card Schemes", Visa and Mastercard are 2 of the majors card schemes. External calls to those systems can be mocked.

The payment system has the following features:

- a user can **enroll** to the service with some informations (email,...)
- a user can **add** one or multiple digitized cards
- a user can **see** a list of his digitized cards
- a user can **delete** one or multiple digitized cards
- a user can see some **analytics** data on his payments

Architecture overview