🏨

Coding & Architecture Challenge

We are about to start a business around insuring hotel bookings. We've already got our first client who wants us to insure their bookings. They have an IT department so they can send to us their booking information.

User Stories

They are ordered according to priority:

US1: Insure confirmed bookings

Definition of Done

- Given a booking, when a confirmation arrives, then insure the booking.

US2: Reinsure bookings according to modifications

Definition of Done

- Given an insured booking, when a modification arrives, then reinsure the booking with the newer information.

US3: Invoice insured bookings

Definitions of Done

- Given some bookings insured, when someone ask for them, then list all insured bookings with its premium amount.

US4: Generate a report of cancelled insured bookings

Definition of Done

- Given some cancelled insured bookings, when someone ask for them, then list them all.

Specifications

Hotel communication contract

The hotel is going to send to us every booking that should be insured. We agreed to provide them of an HTTP endpoint, and only one, that accepts JSON messages representing booking notifications.

Those messages contain an `action` attribute which determines the action that triggered them. Its value can be any of the following:

- `confirmation`
- `modification`
- `cancellation`

Example Request

```
{
  "reference": "A83K1C",
  "action": "confirmation",
  "check_in": "2022-09-13",
  "check_out": "2022-09-17",
  "people": 3
}
```

Response

The hotel expects to receive a 200 HTTP status response for every requests. In any other case they will retry until they receive the 200 HTTP status.

Insurance API

We've already reach an agreement with an insurance company and they sent to us their API documentation. They have created an ad-hoc product for us that is called `FLEXIBLE_INSURED`. However, the API is not ready yet. They will provide us the exact URI in the future. Meanwhile, we need to start developing and be ready for the moment we can point to their real API.

Premium amount calculation

The `premium_amount` value should be calculated according to number of nights and people insured.

- Every person adds 0,12 €.
- Every night adds 0,08 €.

Example

Request

```
{
  "reference": "A83K1C",
  "check_in": "2022-09-13",
  "check_out": "2022-09-17",
  "people": 3,
  "premium_amount": 0.68,
  "currency": "EUR",
  "product": "FLEXIBLE_INSURED"
}
```

Response

```
{
  "reference": "A83K1C",
  "policy": "BI-281921"
}
```

Clarifications

- The purpose of this exercise is to understand how you approach problems and have something to discuss in an interview.
- Focus on providing the value contained within the User Stories. Be pragmatic finding a balance between engineering and product trade-offs. Neither over-engineer nor mortgage future growth. Testing is important though. Do not bother implementing authentications of any kind or being defensive regarding the external world.
- The exercise explanations are vague on purpose. We expect you to have questions and we'll provide you with all support needed.
- We encourage you to spend no more of 4 hours working on it. This is not because we are measuring the lines of code you can write per hour, it is because we respect your spare time.
- Make yourself comfortable and use your preferred programing language.

Deliverables

Architecture

- Draw an architecture diagram of how the system you are about to built will look like. Do not bother using flashy tooling, do it using pen and paper and take a picture of it. This will serve as a starting point to begin an architecture discussion.

Coding

- Start by the first User Story and code until the time is up. Have a clear idea of how your design will look like since we'll get deeper in the interview.
- You do not need to finish the whole exercise, although you should think through the remaining User Stories and how you would implement them since it will part of a later discussion.