# Java Coding Challenge

Hi, Backend developer! :slightly_smiling_face:

We need your help to develop **SMS Billing system** from scratch which should be a Spring app and we got stuck on a couple of user stories.

To give you a bit of a background, we need to be able to charge our customers in the end of the month for each SMS they send. For all customers prices are the same and 1 SMS costs 0.001 USD.

The actual charging of our customers is happening as per request of our Invoicing System once per month in the end of the month. Our Invoicing System is a separate microservice (already developed) that would send a request to our SMS Billing service to get the information about **t**he amount that a customer needs to pay for the current billing month. We need your help here to enable our customers to use the SMS Billing system but also our Invoicing system to charge our customers correctly.

**What we would expect you to do:**

1. Model of a Customer. Populate DB with 2 customers called “Bank” and “Shop”.
2. Create an endpoint that is enabling our customers to send SMS message. Please note that we do not expect to see the real SMS going through, it would be enough to just acknowledge that a message was sent by a customer with a price.
3. Create an endpoint that is returning the amount that a customer needs to pay for the current billing month in USD. Billing month starts on the 1st of the calendar month and ends on the last day of the calendar month.. This endpoint is meant to be used by our Invoicing System to charge our customers one per month. You do not need to create an invoice as a part of that endpoint - this would be the responsibility of the Invoicing System itself.
4. We want to be able to offer 3 different plans to our customers. There should be a **Basic** plan that enables our customers to send 0 free SMS messages per month (no free messages). **Silver** plan that allows a customer up to 100 free SMS message per month. **Gold** plan allows our customers to send up to 1000 free SMS messages per month. Plan itself does not cost anything; we want to be using “pay as you go” strategy. After this limit of free messages is exceeded, the price of 1 SMS would be as following
   * Basic plan customers would need to pay 0.001 USD for 1 SMS
   * Silver plan customers would need to pay 0.002 USD for 1 SMS
   * Gold plan customers would need to pay 0.003 USD for 1 SMS

Please adjust the above models and endpoints to be able to support these plans. You can assume that each customer is linked to only one plan and this plan cannot be changed.

**Additional tasks for bonus points you are not expected to do but we would be impressed if you do:**

* We would like to offer a discount to the most loyal customers. Only for those customers who are using Gold plan and only in the case that they sent more than 100k of SMS messages for a billing month, we should give them a discount and charge them exactly 0.0005 USD for 1 SMS counting forward after 100k threshold is exceeded. Their first SMS should still cost 0.003 USD and their 100000th SMS should still cost 0.003 USD, however their 100001st SMS and every message after that one should cost 0.0005 USD. In the billing month they would again need to send more than 100k of SMS messages to get the same discount. Please adjust the endpoint that is returning the amount that a customer needs to pay for the current billing month in USD to reflect the discount too.
* Dockerize the solution

**Technicalities**

* You are not expected to implement any kind of UI for this, this is a backend coding challenge.
* You can choose relational database of your choice.
* You can use any open-source library you think is useful for this.
* In order to spare some of your time, please note that you are not expected to make this API secured nor implement any authentication method nor you would get bonus points for that.

Once you are done, please push your solution to your own GitHub repository and make sure it is public. Then, share the link with us please.   
In case you want to keep this repository private, just make sure you invite these 2 GitHub users to review:  
- **gajicvedrana**  
**- BartekZ**  
  
In case you have any questions, we are here to help – please do not hesitate to contact us! And do not forget to come back to us via email when you are done. In that email include the link of the GitHub repository too.

Best of luck!