



# Back-End Developer Case Study

## Description

You are tasked with building 2 applications: one that receives events and one that stores the events. These 2 applications must communicate with each other via Google Protocol Buffers.

The first application receives logs of events via HTTP and must pass them to the second application for storing them in the database (for the case study we will not use a database, logging them to a file on the disk will suffice).

## Implementation details

The first application must be built in JAVA as an API. The API must have 1 route where a user may send events (JSON objects with data) via POST calls.

Here is an example event:

```
{
  "timestamp" : 1518609008,
  "userId" : 1123,
  "event" : "2 hours of downtime occurred due to the release of version 1.0.5 of the system"
}
```

Once an event is received it must be serialized in Google Protocol Buffers format and passed on to the second application for persistence.

The second application must be built in another programming language of your choice and must receive data from the JAVA API in GPB format, deserialize it and log it to a file on the disk.

The communication protocol between the two applications is not specified but it shouldn't be HTTP, you may choose whatever suits you best and makes your life easier, the important bit is to use Google Protocol Buffers for whatever protocol you choose.

## How to deliver the results

In delivering the results please make sure that you have the following:

- A **git repository** of your work
- A **Readme.md** file that specifies how the application should be run
- A **documentation** of the implemented code



## Notes

Please keep in mind that this page is part of a more complex application and the code should be organized appropriately.

Please make sure that the code provided is self-sufficient and does not depend on 3rd party resources that can be taken down between the time of the coding and the time of the review.

**Happy coding! :)**