# Backend Engineer Coding Assignment Overview

## Context

Axmed is set to accelerate healthcare and pharmaceutical access in LMICs.

At the forefront of innovation, Axmed’s primary strength is a technology platform that enables buyers to launch tenders for pharmaceutical they need, and sellers to bid on that.

The added transparency puts the buyers – for the first time – in a strong position when negotiating the prices for the medicines they need.

## The assignment

At the core of Axmed’s platform is a medication [SKU catalogue](https://www.lawinsider.com/dictionary/product-sku), allowing sellers to quickly find the medicine they want to buy and sellers to bid for prices and win tenders launched by the buyers. Through our backoffice solution, we can easily manage such catalogues by editing/inserting entries manually.

As part of this assignment, we invite the candidate to recreate a simplified version of the services that a frontend application will use to manage (CRUD) and bulk-upload SKUs to our catalogue.

## Requirements

The following list is what we expect to see for the assignment

**Functionalities**

* Enable users to create, read, update, or delete single medication SKUs through dedicated APIs
* Enable users to bulk create medication SKUs through a dedicated API

**Technology**

* Backend APIs with Python, ideally with Django to manipulate the data for the frontend.
* Ensure data persistence using a database of your choice.
* Make sure to submit your solution’s full git history via Github
* Documentation on how to deploy and run locally the solution

**Data Model**

* The fields for the medication SKU object are: Medication Name, Presentation, Dose and Unit
  + Examples: Amoxicillin, Tablet, 50, mg
* Two different SKUs cannot have the same combination of Medication Name, Presentation, Dose and Unit
* Medication names are also unique

**Testing**

* Ability to run tests on your solution

**CI/CD and Deployment**

* No constraints are put on this part

## Bonus Points:

**CI/CD and Deployment**

* Using Docker for deployments is not needed but is a nice addition
* Automating test + deployment, showing how this could work in a real scenario would be a great plus

**Next-step Review**

* We like to understand what the candidate would have prioritised, if given more time to build the solution