### Background

In Pixelz, We help ecommerce studio professionals beat deadlines by providing reliable, AI-powered retouching at scale. We have a portal application that allows our customers to create the order, do the payment and get the recurring billing…

### Scenario outline

Our clients need to checkout their orders (payout the orders) to push their orders to our Production system.

When they create a checkout order successfully, we need to send out an email to them to let them know that it is successful. We also have to create an invoice to our invoice system, otherwise, we need to call our internal Production system (Production service) to push this order to be processed by our internal system.

Example:

1. The Art Director goes to our system and searches the orders that they have already created.
2. The Art Director could find the order by order name.
3. The Art Director checks out the order and if the payment is successful, then the order should be pushed to our Production system.

Notes:

* You could mock the Payment Service Provider to simulate both failed and successful payment transactions.
* You could mock the Email service too.
* You could mock the API of our internal system (Production service) to simulate both failed and successful cases.

### Functional Requirements

* Orders could be searched/filtered by name
* If the order is checked out successfully, we need to call our internal system (Production system) to update the state of the order in the internal system.
* Client has to receive the email if the payment is successful.

### What to submit

* System design(s) to meet the requirements above
  + Appropriate data models and schemas.
  + Relevant HTTP endpoints (REST, gRPC, GraphQL etc)
  + The components and their interactions within the system.
  + Any assumptions you’ve made, and how you might validate them.
* Plan for delivery
* Track and include how much time you spent on this

### How to submit

* Use whatever tools you’re comfortable with to put your submission together, as long as it is easily accessible and shared with us.
  + e.g Github repo, Google Doc, Notion, Figma, Lucid chart etc.
  + Please no attached zip files or downloads.
* Email us a link to your submission.

### Additional things to think about

* **Show off your skills and knowledge!!!**
* **This codebase might be handled by a team of 10 or 20 engineers. How do you structure the code-base?**
* How will you manage the integrity of data?
* How will you manage performance?
* Go into some detail *where necessary* to showcase your skills.