

Design decisions

Version 8 of java is used as I am currently working in same version.

SpringBoot is used as it makes it easy to create spring based applications without much fuss. Version **2.5.4** is used as it is the latest version as of August 2021.

Project **Lombok** is used to avoid writing boilerplate code like getters, setters, constructors etc.

AS-1: As TNT, I want to be able to query all services in a single network call to optimise network traffic.

A single call to this application contains parameters for three backend services. I am collecting them in three different parameters, which will be placed in three separate queues. From each queue there will be a rest call to backend and response is collected in a field. **Resttemplate** is used to make calls. Should have perhaps switched to webclient as that is going to be default now with spring.

After placing the request in queue, call is redirected to check if we have there is a response available for all backend services. Once response is available, call is redirected to get the aggregated response and return it.

AS-2: as TNT, I want service calls to be throttled and bulked into consolidated requests per respective API to prevent services from being overloaded.

Queue mentioned in the previous solution is **ArrayBlockingQueue** with capacity as 5. Blocking queue waits for the queue to be available before putting new one. This helps us in consolidating 5 requests into one. Each request is attached to a UUID which helps us mapping backend requests to main request by the user. Each backend request is executed as a new thread to enable parallel processing so that main request can get response quickly.

AS-3: as TNT, I want service calls to be scheduled periodically even if the queue is not full to prevent overly-long response times.

Application stores the time when first entry to the queue was made. There is a scheduler which checks every second whether first entry to queue is there for more than 5 seconds.

If yes, it empties the queue collecting all the requests to make a consolidated request. And the time is set to zero till there is a new entry in the queue.