

Design Document – Crossover Log API

The objective of this document is to describe the design of an Elixir based REST API. As a design strategy, we give the highest priority in minimization of development effort.

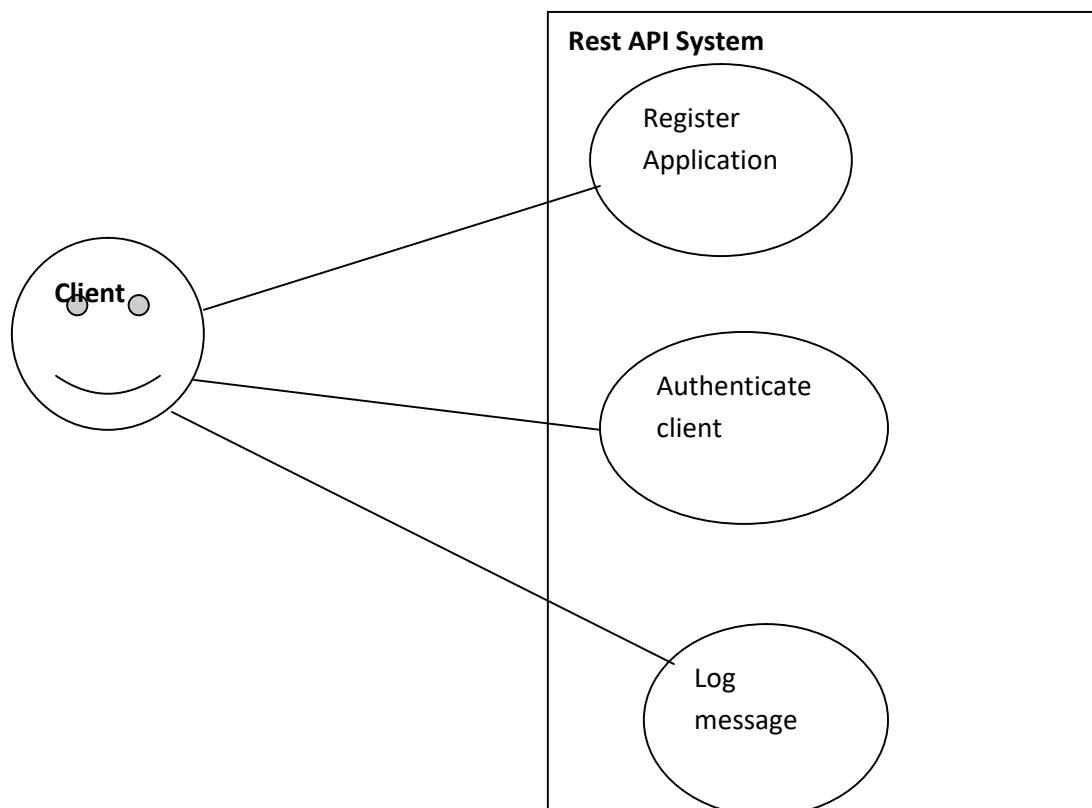
This API has three functionalities.

1. A register interface where a client will send the display name of an application. The system will generate two random strings of 32 characters as application id and application secret and respond back three parameters as a JSON to the client.
2. An auth interface where a client will send the application id and application secret of a registered application and the system will return an access token to the client if input parameters are valid.
3. A log interface where a client will send a valid access token as HTTP header and log parameters as JSON body. On successful logging, the system will return true otherwise false.

Our assumption:

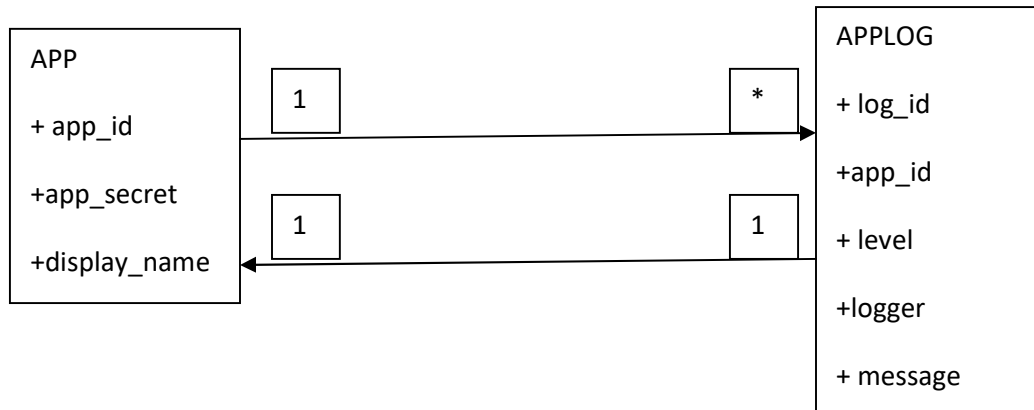
1. Input message will come in UTF-8 encoded format.
2. Access token will be valid for 10 min.

Below use case diagram will describe the system functionalities.



Structural View:

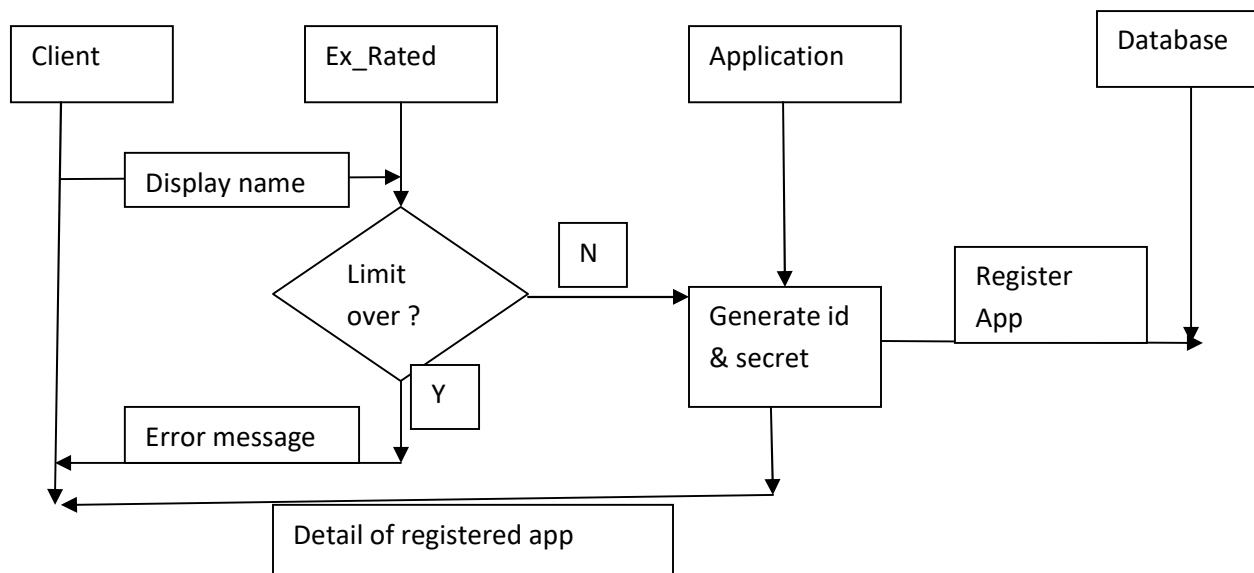
The structural view of the system is described by below class diagram. The system has two entities - App (Application) and Aplog (Application Log). They are associated with two relations. One App has many Applog. One Applog can have only one App.



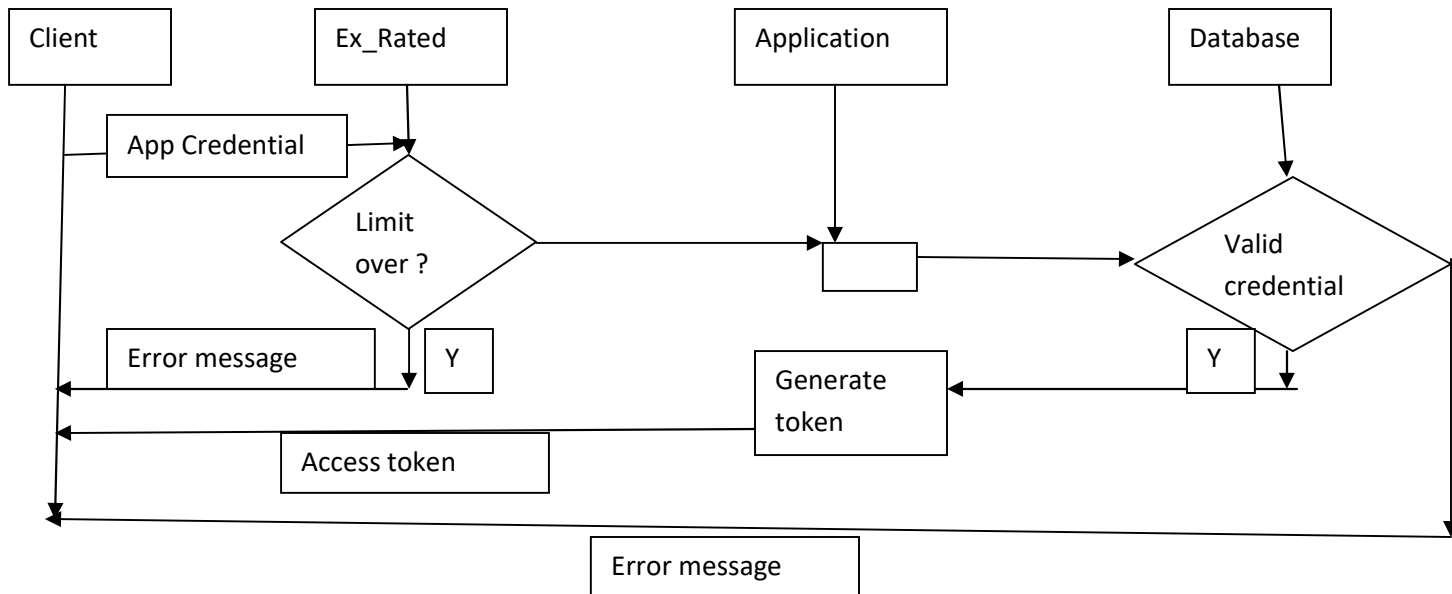
Behavioral view:

We described the behavioral view by sequence diagrams. Along with three interface system also has one constraint. If any point of time for any of the end point request rate is greater than 60/min, then the system will ignore the request for that endpoint. We implement this request checking by ExRated, the Elixir OTP GenServer.

Sequence diagram for register end point.



Sequence Diagram of auth end point



Two implement temporary access token we generate the access token by concatenating application secret with current server time in millisecond. So when a new log request comes with an access token, system split the token into application secret and time stamp. If application secret is valid and time stamp is not older than 10 min, the system then only processes the request.

Testing Strategy:

For testing purpose we create two additional end point

1. Get /applications – get details of all stored applications
2. Get /applogs - get details of all stored application logs

Using this two API we can easily check result of each transaction.