

1. Document history

Version	Description	Date	Created by	Approved by
1.0.0	Initial version	2019-07-07	Rodrigo Pio	<PM>

2. References

[Toggler - Requirements gathering \(v1.0.0\).docx](#)

3. Context and objectives

XPTO company has a digital platform built under SOA that currently contains 62 services/applications inside it. When a team needs to implement a new feature in his service, they implement it under a toggle so that the feature can be turned “on” (true) or “off” (false) as a fail-safe. To make a change in the toggle value on a production environment, the team needs to change it in a static file and restart the service/application in each server so that the changes take effect.

XPTO company wants to change the system and is asking you to implement a new REST Web Api in the technology of your choice (preferably C# or Java) that could manage in real-time and dynamically the platform toggles.

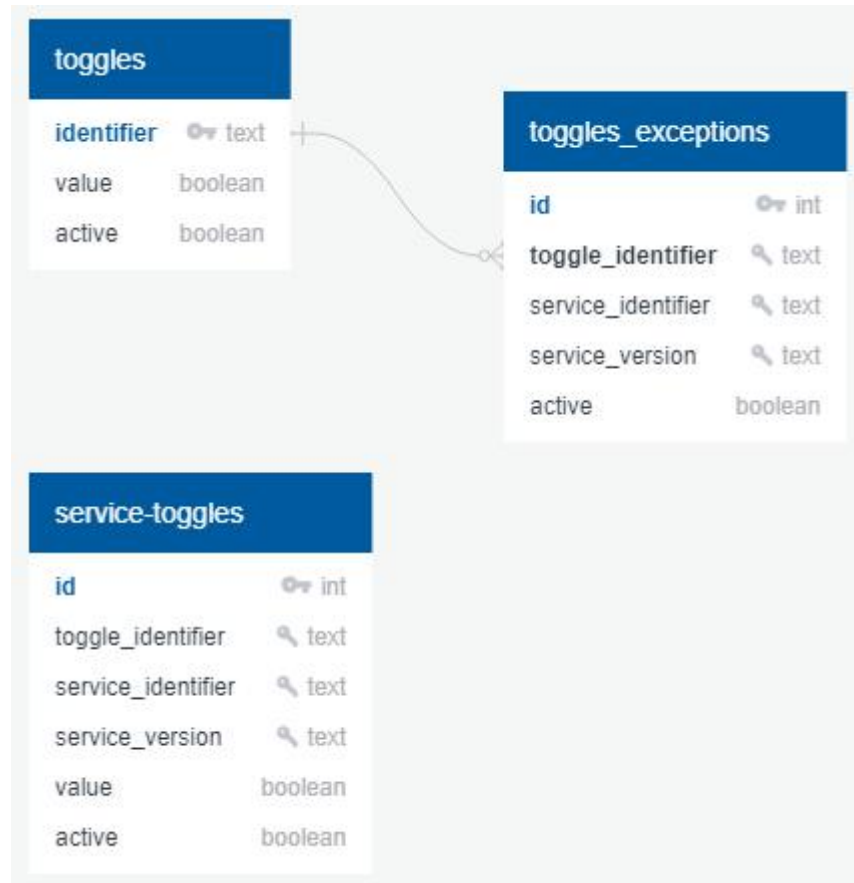
4. Business Requirements

#	Requirement	Description
1	CRUD operations	This service must be able to register new toggles that can be used by one or more services or applications with a hierarchy.
2	Status query	When the services/applications request their toggles, they will only provide their identifiers and version.
3	Documentation	The company has multiple development teams around the world, and with the different time-zones they will not be able to be in contact with you, so the service must have a clear documentation for them to use.
4	Decoupling	There will be a team that will maintain this service, and they must be able to easily change the code and deploy this service with confidence that they will not break any contract with the other services in the platform.

5. Technical Requirements

- Object oriented programming;
- SOLID Principles;
- Clean Code;
- Unit/Automation/Integration Testing;
- Design Patterns;
- REST API;

6. Database model



7. REST API

#	Method	Resource type	Action	Request	Response
	POST	toggles	Creates or replaces a toggle for all services and applications	https://<host>/toggles <pre>{ "identifier": "<toggle_identifier>", "value": <true false>, "active": <true false> }</pre>	<pre>{ "identifier": "<toggle_identifier>", "value": <true false>, "active": true, "_links": { "self": { "href": "http://<host>/toggles/<toggle_identifier>" } }, "toggles": { "href": "https://<host>/toggles" } }</pre>
	PUT	toggles	Turns a toggle “on” (if the service or application informed has an override, only the value of the override will be changed)	https://<host>/toggles/<toggle_identifier>/on <pre>{ "identifier": "<toggle_identifier>", "serviceidentifier": "<service_identifier>", "serviceVersion": "<service_version>" }</pre>	<pre>{ "identifier": "<toggle_identifier>", "serviceidentifier": "<service_identifier>", "serviceVersion": "<service_version>", "value": <true false>, "_links": { "self": { "href": "http://<host>/toggles/<toggle_identifier>" } }, "toggles": { "href": "https://<host>/toggles" } }</pre>
	PUT	toggles	Turns a toggle “off” (if the service or application informed has an override, only the value of the override will be changed)	https://<host>/toggles/<toggle_identifier>/off <pre>{ "identifier": "<toggle_identifier>", "serviceidentifier": "<service_identifier>", "serviceVersion": "<service_version>" }</pre>	<pre>{ "identifier": "<toggle_identifier>", "serviceidentifier": "<service_identifier>", "serviceVersion": "<service_version>", "value": <true false>, "_links": { "self": { "href": "http://<host>/toggles/<toggle_identifier>" } }, "toggles": { "href": "https://<host>/toggles" } }</pre>
	GET	toggles	Get the list of all toggles	https://<host>/toggles	<pre>{ "_embedded": { "toggleList": [{ "toggleIdentifier": "<toggle_identifier>", "value": <true false>, </pre>

TECHNICAL SPECIFICATION

TOGGLER

#	Method	Resource type	Action	Request	Response
					<pre> "active": <true false>, "_links": { "self": { "href": "http://localhost:8080/toggles/<toggle_identifier>" }, "toggles": { "href": "http://localhost:8080/toggles" } }, {...}] }, "_links": { "self": { "href": "http://localhost:8080/toggles" } } } </pre>
	GET	toggles	Query the value of a toggle	<pre> https://<host>/toggles/<toggle_identifier> HTTP headers: "service-identifier"=<service_identifier> "service-version"=<service_version> </pre>	<pre> { "identifier": "<toggle_identifier>", "serviceidentifier": "<service_identifier>", "serviceVersion": "<service_version>", "value": <true false>, "_links": { "self": { "href": "http://<host>/toggles/<toggle_identifier>" }, "toggles": { "href": "https://<host>/toggles" } } } </pre>
	DELETE	toggles	Deactivate (disable) a toggle for all services and applications	<pre> https://<host>/toggles/<toggle_identifier>/deactivate </pre>	<pre> { "identifier": "<toggle_identifier>", "value": <true false>, "active": false, "_links": { "self": { "href": "http://<host>/toggles/<toggle_identifier>" }, "toggles": { "href": "https://<host>/toggles" } } } </pre>
	PUT	toggles	Activates (enable) a toggle for all services and applications	<pre> https://<host>/toggles/<toggle_identifier>/activate { "identifier": "<toggle_identifier>", </pre>	<pre> { "identifier": "<toggle_identifier>", "value": <true false>, "active": true, </pre>

TECHNICAL SPECIFICATION

TOGGLER

#	Method	Resource type	Action	Request	Response
				<pre>"value": <true false>, "active": <true false> }</pre>	<pre>"_links": { "self": { "href": "http://<host>/toggles/<toggle_identifier>" }, "toggles": { "href": "https://<host>/toggles" } } }</pre>
	GET	toggles-exclusions	Get the list of all exclusions of services or applications for toggles	https://<host>/toggles-exclusions	<pre>{ "_embedded": { "toggleExclusionList": [{ "id": "<toggles_exclusions_id>", "toggleIdentifier": "<toggle_identifier>", "serviceIdentifier": "ABC", "serviceVersion": "1.0.0", "active": <true false>, "_links": { "self": { "href": "http://localhost:8080/toggles-exclusions/<toggles_exclusions_id>" }, "toggle-exclusion": { "href": "http://localhost:8080/toggles-exclusions" } }, {...} }], "_links": { "self": { "href": "http://localhost:8080/toggles-exclusions" } } } }</pre>
	POST	toggles-exclusions	Creates an exclusion of a service or application for a toggle	https://<host>/toggles-exclusions <pre>{ "identifier": "<toggle_identifier>", "serviceIdentifier": "<service_identifier>", "serviceVersion": "<service_version>" }</pre>	<pre>{ "id": <toggles_exclusions_id>, "toggleIdentifier": "<toggle_identifier>", "serviceIdentifier": "<service_identifier>", "serviceVersion": "<service_version>", "active": <true false>, "_links": { "self": { "href": "http://<host>/toggles-exclusions/<toggles_exclusions_id>" }, "toggle-exclusions": { "href": "https://<host>/toggles-exclusions" } } }</pre>

TECHNICAL SPECIFICATION

TOGGLER

#	Method	Resource type	Action	Request	Response
					<pre> } } } </pre>
	DELETE	toggles-exclusions	Deactivate (disable) an exclusion of a service or application for a toggle	<pre> https://<host>/toggles/<toggle_exclusion_id>/deactivate </pre>	<pre> { "id": <toggle_exclusion_id>, "toggleIdentifier": "<toggle_identifier>", "serviceIdentifier": "<service_identifier>", "serviceVersion": "<service_version>", "active": false, "_links": { "self": { "href": "http://<host>/toggles-exclusions/<toggles_exclusions_id>" }, "toggles-exclusions": { "href": "https://<host>/toggles-exclusions" } } } </pre>
	PUT	toggles-exclusions	Activate (enable) an exclusion of a service or application for a toggle	<pre> https://<host>/toggles-exclusions/<toggle_exclusion_id>/activate { "identifier": "<toggle_identifier>", "serviceIdentifier": "<service_identifier>", "serviceVersion": "<service_version>" } </pre>	<pre> { "id": <toggle_exclusion_id>, "toggleIdentifier": "<toggle_identifier>", "serviceIdentifier": "<service_identifier>", "serviceVersion": "<service_version>", "active": true, "_links": { "self": { "href": "http://<host>/toggles-exclusions/<toggles_exclusions_id>" }, "toggles-exclusions": { "href": "https://<host>/toggles-exclusions" } } } </pre>
	GET	toggles-overrides	Get the list of all overrides of services or applications for toggles	<pre> https://<host>/toggles-overrides </pre>	<pre> { "_embedded": { "toggleExclusionList": [{ "id": "<toggles_exclusions_id>", "toggleIdentifier": "<toggle_identifier>", "serviceIdentifier": "ABC", "serviceVersion": "1.0.0", "active": <true false>, "_links": { "self": { "href": "http://localhost:8080/toggles-exclusions/<toggles_exclusions_id>" }, "toggles-exclusion": { "href": </pre>

TECHNICAL SPECIFICATION

TOGGLER

#	Method	Resource type	Action	Request	Response
					<pre>"http://localhost:8080/toggles-exclusions" } }, {...}] }, "_links": { "self": { "href": "http://localhost:8080/toggles-exclusions" } }</pre>
	POST	toggles-overrides	Creates a toggle for a specific service or application	<pre>https://<host>/toggles-overrides { "identifier": "<toggle_identifier>", "serviceIdentifier": "<service_identifier>", "serviceVersion": "<service_version>", "value": <true false>, "active": <true false> }</pre>	<pre>{ "id": <toggles_overrides_id>, "toggleIdentifier": "<toggle_identifier>", "serviceIdentifier": "<service_identifier>", "serviceVersion": "<service_version>", "value": <true false>, "active": <true false>, "_links": { "self": { "href": "http://<host>/toggles-overrides/<toggle_identifier>" }, "toggles": { "href": "https://<host>/toggles-overrides" } } }</pre>
	DELETE	toggles-overrides	Deactivate (disable) a toggle for a specific service or application	<pre>https://<host>/toggles-overrides/<toggle_override_id>/deactivate</pre>	<pre>{ "id": <toggle_override_id>, "toggleIdentifier": "<toggle_identifier>", "serviceIdentifier": "<service_identifier>", "serviceVersion": "<service_version>", "active": false, "value": <true false>, "_links": { "self": { "href": "http://<host>/toggles-overrides/<toggle_identifier>" }, "toggles": { "href": "https://<host>/toggles-overrides" } } }</pre>
	PUT	toggles-overrides	Activate (enable) a toggle for a specific service or application	<pre>https://<host>/toggles-overrides/<toggle_override_id>/activate</pre>	<pre>{ "id": <toggle_override_id>, "toggleIdentifier": "<toggle_identifier>", "serviceIdentifier": "<service_identifier>",</pre>

#	Method	Resource type	Action	Request	Response
					<pre>{ "serviceVersion": "<service_version>", "active": true, "value": <true false>, "_links": { "self": { "href": "http://<host>/toggles-overrides/<toggle_identifier>" }, "toggles": { "href": "https://<host>/toggles-overrides" } } }</pre>

8. Documentation

Online documentation for the REST API is available after deploy at the following link: <http://<host>:8080/swagger-ui.html>

9. Tests

a) Scenarios

#	Scenario	Initial condition	Outcome
1	Create toggle "isButtonBlue" for all services and applications. The toggle must be set "on" (true).	Toggler receives a request with the following information: <ul style="list-style-type: none">Toggle identifier "isButtonBlue"	A new toggle is created for all services and platforms HTTP Status code 200 OK The response from Toggler contains the following information: <ul style="list-style-type: none">Toggle identifierToggle value "off" (false)
2	Create toggle "isButtonBlue" for service "ABC"	Toggler receives a request with the following	A new toggle is created for all services and platforms

#	Scenario	Initial condition	Outcome
	version "1.0.0". The toggle must be set "off" (false).	information: <ul style="list-style-type: none"> ● Toggle identifier "isButtonBlue" ● Service identifier "ABC" ● Service version "1.0.0" 	<p>HTTP Status code 200 OK</p> <p>The response from Toggler contains the following information:</p> <ul style="list-style-type: none"> ● Toggle identifier ● Toggle value "off" (false)
3	Create toggle "isButtonGreen" for service "ABC" version "1.0.0". The toggle must be set "on" (true).	Toggler receives a request with the following information: <ul style="list-style-type: none"> ● Toggle identifier "isButtonGreen" ● Service identifier "ABC" ● Service version "1.0.0" 	<p>A new toggle is created for all services and platforms</p> <p>HTTP Status code 200 OK</p> <p>The response from Toggler contains the following information:</p> <ul style="list-style-type: none"> ● Toggle identifier ● Toggle value "on" (true)
4	Create toggle "isButtonRed" for all services and applications except for "ABC". The toggle must be already "on" (true).	Toggler receives a request with the following information: <ul style="list-style-type: none"> ● Toggle identifier "isButtonRed" ● Exclusion filled with the following element: <ul style="list-style-type: none"> ■ Service identifier "ABC" ■ Service version "1.0.0" 	<p>A new toggle is created for all services and platforms</p> <p>HTTP Status code 200 OK</p> <p>The response from Toggler contains the following information:</p> <ul style="list-style-type: none"> ● Toggle identifier ● Toggle value "on" (true)
5	Query the status of toggle "isButtonBlue" as service "XYZ" version "1.0.0"	Toggler receives a request with the following information: <ul style="list-style-type: none"> ● Toggle identifier "isButtonBlue" ● Service identifier "XYZ" ● Service version "1.0.0" 	<p>HTTP Status code 200 OK</p> <p>The response from Toggler contains the following information:</p> <ul style="list-style-type: none"> ● Toggle identifier ● Toggle value "on" (true)
6	Query the status of toggle "isButtonBlue" as service "ABC" version "1.0.0".	Toggler receives a request with the following information:	HTTP Status code 200 OK

TECHNICAL SPECIFICATION

TOGGLER

#	Scenario	Initial condition	Outcome
		<ul style="list-style-type: none">● Toggle identifier "isButtonBlue"● Service identifier "ABC"● Service version "1.0.0"	The response from Toggler contains the following information: <ul style="list-style-type: none">● Toggle identifier● Toggle value "off" (false)
7	Query the status of toggle "isButtonGreen" as service "ABC" version "1.0.0".	Toggler receives a request with the following information: <ul style="list-style-type: none">● Toggle identifier "isButtonGreen"● Service identifier "ABC"● Service version "1.0.0"	HTTP Status code 200 OK The response from Toggler contains the following information: <ul style="list-style-type: none">● Toggle identifier● Toggle value "on" (true)
8	Query the status of toggle "isButtonGreen" as service "XYZ" version "1.0.0".	Toggler receives a request with the following information: <ul style="list-style-type: none">● Toggle identifier "isButtonGreen"● Service identifier "XYZ"● Service version "1.0.0"	HTTP Status code 404 Not Found
9	Query the status of toggle "isButtonRed" as service "XYZ" version "1.0.0".	Toggler receives a request with the following information: <ul style="list-style-type: none">● Toggle identifier "isButtonRed"● Service identifier "XYZ"● Service version "1.0.0"	HTTP Status code 200 OK The response from Toggler contains the following information: <ul style="list-style-type: none">● Toggle identifier● Toggle value "on" (true)
10	Query the status of toggle "isButtonRed" as service "ABC" version "1.0.0".	Toggler receives a request with the following information: <ul style="list-style-type: none">● Toggle identifier "isButtonRed"● Service identifier "ABC"● Service version "1.0.0"	HTTP Status code 404 Not Found
11	Create toggle "isButtonBlue" for all services and applications. The toggle must be set "on" (true).	Toggler receives a request with the following information: <ul style="list-style-type: none">● Toggle identifier "isButtonBlue"	HTTP Status code 409 Conflict The response from Toggler contains the following information: <ul style="list-style-type: none">● Error code and description

TECHNICAL SPECIFICATION

TOGGLER

#	Scenario	Initial condition	Outcome
12	Create toggle "isButtonBlue" for service "ABC" version "1.0.0". The toggle must be set ""off" (false)".	Toggler receives a request with the following information: <ul style="list-style-type: none"> ● Toggle identifier "isButtonBlue" ● Service identifier "ABC" ● Service version "1.0.0" 	HTTP Status code 409 Conflict The response from Toggler contains the following information: <ul style="list-style-type: none"> ● Error code and description
13	Create toggle "isButtonGreen" for service "ABC" version "1.0.0". The toggle must be set ""on" (true)".	Toggler receives a request with the following information: <ul style="list-style-type: none"> ● Toggle identifier "isButtonGreen" ● Service identifier "ABC" ● Service version "1.0.0" 	HTTP Status code 409 Conflict The response from Toggler contains the following information: <ul style="list-style-type: none"> ● Error code and description
14	Create toggle "isButtonRed" for all services and applications except for "ABC". The toggle must be already "on" (true).	Toggler receives a request with the following information: <ul style="list-style-type: none"> ● Toggle identifier "isButtonRed" ● Exclusion filled with the following element: <ul style="list-style-type: none"> ■ Service identifier "ABC" ■ Service version "1.0.0" 	HTTP Status code 409 Conflict The response from Toggler contains the following information: <ul style="list-style-type: none"> ● Error code and description
15	Create toggle "" for all services and applications. The toggle must be set "on" (true).	Toggler receives a request with the following information: <ul style="list-style-type: none"> ● Toggle identifier "" 	HTTP Status code 400 Bad Request The response from Toggler contains the following information: <ul style="list-style-type: none"> ● Error code and description
16	Create toggle "" for service "ABC" version "1.0.0". The toggle must be set "off" (false).	Toggler receives a request with the following information: <ul style="list-style-type: none"> ● Toggle identifier "" ● Service identifier "ABC" ● Service version "1.0.0" 	HTTP Status code 400 Bad Request The response from Toggler contains the following information: <ul style="list-style-type: none"> ● Error code and description
17	Create toggle "" for all except service "ABC" version "1.0.0". The toggle must be set "on" (true).	Toggler receives a request with the following information: <ul style="list-style-type: none"> ● Toggle identifier "" ● Exclusion filled with the following element: <ul style="list-style-type: none"> ■ Service identifier "ABC" 	HTTP Status code 400 Bad Request The response from Toggler contains the following information: <ul style="list-style-type: none"> ● Error code and description

#	Scenario	Initial condition	Outcome
		■ Service version "1.0.0"	

10. Technological stack

- Java 11
- Spring Framework (Boot, Data)
- Swagger (Swagger-UI)
- Apache Tomcat 9.0.21 (development)
- Oracle WebLogic Server 12c (production)
- PostgreSQL 11

11. Design Patterns

- Model-View-Controller
- Dependency Injection
- Singleton
- Front Controller

12. Future releases

- **User and roles**
- **Authentication** through OAuth2 and OpenID Connect
- **Toggle groups** will allow services or applications obtain the value of several correlated toggles with one request
- **Toggle value notifications** will provide a Publisher/Subscriber mechanism for messaging to send updates to services, applications or users about change in toggles states