pág

[Servicios de datos para la Plataforma 2 2](#_Toc163032324)

[Servicio get\_data\_geoinformation\_service 2](#_Toc163032325)

[Microservicio get\_information\_from\_coordinate 2](#_Toc163032326)

[Microservicio get\_geoinformation\_from\_polygon 4](#_Toc163032327)

[Microservicio get\_entities\_ids\_from\_coordinate 6](#_Toc163032328)

[Microservicio get\_coffee\_recommended\_prices 7](#_Toc163032329)

[Microservicio get\_user\_trends 7](#_Toc163032330)

[Microservicio get\_sorted\_lots\_based\_on\_buyer\_preferences 9](#_Toc163032331)

[Microservicio get\_lot\_coding 9](#_Toc163032332)

[Microservicio service\_get\_farmers\_farm\_info 10](#_Toc163032333)

[Microservicio train\_recommendation\_engine 10](#_Toc163032334)

[Microservicio get\_recommended\_lots\_from\_engine 11](#_Toc163032335)

[Servicio consume\_businessman\_profile\_data\_microservice 12](#_Toc163032336)

[Microservicio get\_available\_items\_for\_lots 12](#_Toc163032337)

[Microservicio get\_lot\_information 12](#_Toc163032338)

[Microservicio get\_products\_by\_search\_filter 13](#_Toc163032339)

[Servicio consume\_farmer\_profile\_data\_microservice 15](#_Toc163032340)

[Microservicio get\_farm\_info 15](#_Toc163032341)

[Servicio consume\_learner\_profile\_data\_microservice 16](#_Toc163032342)

[Microservicio estimate\_learning\_style 16](#_Toc163032343)

[Microservicio make\_a\_chatbot\_query 18](#_Toc163032344)

[Microservicio obtain\_success\_rate\_of\_evaluation\_results 18](#_Toc163032345)

[Microservicio estimate\_course\_progress 20](#_Toc163032346)

# Servicios de datos para la Plataforma 2

Para los servicios de datos para plataforma 2 se encuentran disponibles cinco rutas (correspondientes a los endpoints desarrollados por el componente backend). Estas son:

* get\_data\_geoinformation\_service [POST]
* /charts/<filename> [GET]
* consume\_farmer\_profile\_data\_microservice [POST]
* consume\_businessman\_profile\_data\_microservice [POST]
* consume\_learner\_profile\_data\_microservice [POST]

En las dos primeras rutas se especifican los microservicios, ya integrados y en funcionamiento, que se tienen para la plataforma con corte en finales del año 2023. Las últimas tres rutas son las más recientes y se platearon pensando en la migración progresiva de los microservicios ya funcionales que comprenden las otras dos rutas. Se proyecta que, una vez finalizada la migración, sólo se cuente con rutas referentes a los perfiles que se manejan en la plataforma, que de momento son caficultor, empresario y aprendiz. Si surge una nueva figura o perfil, por ejemplo administrador, se creará una nueva ruta. Esto con el fin de independizar los microservicios y facilitar los procesos de desarrollo, mantenimiento y testeo con miras a una arquitectura basada en microservicios.

En este documento se exponen únicamente los microservicios que ya se encuentran disponibles y se consumen en entorno de producción.

## Servicio get\_data\_geoinformation\_service

### Microservicio get\_information\_from\_coordinate

Dada una coordenada (longitud, latitud), retorna el valor de temperatura, altitud y clima para la localización respectiva.

{

"service": "get\_information\_from\_coordinate",

"longitude": <valor\_longitud>,

"latitude": <valor\_latitud>

}

**Ejemplo 1**

Cuerpo de la petición:

{

    "service": "get\_information\_from\_coordinate",

    "longitude": -76.5113,

    "latitude": 2.4213

}

Respuesta:

{

    "altitude": 1992,

    "climate": "Semihúmedo",

    "temperature": "12 a 16 °C"

}

**Ejemplo 2**

Si se provee una coordenada para la que no se tenga la información solicitada, es decir, fuera de los límites de Colombia, devuelve el valor -99 para las tres claves en el Respuesta:

Cuerpo de la petición:

{

    "service": "get\_information\_from\_coordinate",

    "longitude": -76.5113,

    "latitude": 32.4213

}

Respuesta:

{

    "altitude": -99,

    "climate": -99,

    "temperature": -99

}

### Microservicio get\_geoinformation\_from\_polygon

Dado un id, que puede ser de un departamento (id\_state), municipio (id\_city) o vereda (id\_village), devuelve la geoinformación respectiva. Esto incluye el bounding box (conjunto de coordenadas que construyen un rectángulo que encierra el polígono) y el centroide. Esta información se usa en la etapa en la que un caficultor especifica la ubicación de su finca.

{

"service": "get\_geoinformation\_from\_polygon",

"id\_state": <valor\_id>,

"id\_city": <valor\_id>,

"id\_village": <valor\_id>

}

Sólo se le puede pasar un id bien sea del departamento, municipio o vereda. Si se le pasa más de un id, el servicio toma en cuenta únicamente el último id recibido.

**Ejemplo 1**

Cuerpo de la petición:

{

    "service": "get\_geoinformation\_from\_polygon",

    "id\_state": 790,

    "id\_city": null,

    "id\_village": null

}

Respuesta:

{

    "bbox": {

        "xmax": -75.74782424722925,

        "xmin": -77.9283381527739,

        "ymax": 3.3289411577604913,

        "ymin": 0.9580284644383605

    },

    "centroid": {

        "x": -76.83068719603754,

        "y": 2.3979493332932007

    }

}

**Ejemplo 2**

Cuerpo de la petición:

{

    "service": "get\_geoinformation\_from\_polygon",

    "id\_state": null,

    "id\_city": 19965,

    "id\_village": null

}

Respuesta:

{

    "bbox": {

        "xmax": -75.74782424722925,

        "xmin": -76.25456084307095,

        "ymax": 3.046906891863681,

        "ymin": 2.425065377805925

    },

    "centroid": {

        "x": -75.99368263795537,

        "y": 2.7338877006493765

    }

}

**Ejemplo 3**

Cuerpo de la petición:

{

    "service": "get\_geoinformation\_from\_polygon",

    "id\_state": null,

    "id\_city": null,

    "id\_village": 12555

}

Respuesta:

{

    "bbox": {

        "xmax": -75.90480778699998,

        "xmin": -75.94701824299995,

        "ymax": 2.7830887920000578,

        "ymin": 2.667556963000038

    },

    "centroid": {

        "x": -75.9251036394809,

        "y": 2.7193815269619934

    }

}

### Microservicio get\_entities\_ids\_from\_coordinate

Dada una coordenada (longitud, latitud), devuelve los ids de la vereda, municipio y departamento en los que cae la coordenada.

{

"service": "get\_entities\_ids\_from\_coordinate",

"longitude": <valor\_longitud>,

"latitude": <valor\_latitud>

}

**Ejemplo 1**

Cuerpo de la petición:

{

    "service": "get\_entities\_ids\_from\_coordinate",

    "longitude": -75.9297,

    "latitude": 2.6924

}

Respuesta:

{

    "id\_city": 19965,

    "id\_state": 790,

    "id\_village": 12555

}

**Ejemplo 2**

Si se da una coordenada que cae sobre una ciudad o casco urbano, ya que este tipo de entidades no poseen veredas, se devolverá -99 para la salida del id de la vereda:

Cuerpo de la petición:

{

    "service": "get\_entities\_ids\_from\_coordinate",

    "longitude": -75.97075,

    "latitude": 2.64149

}

Respuesta:

{

    "id\_city": 19965,

    "id\_state": 790,

    "id\_village": -99

}

**Ejemplo 3**

Si se proveen coordenadas por fuera de los límites de Colombia, se devuelve -99 para todas las salidas:

Cuerpo de la petición:

{

    "service": "get\_entities\_ids\_from\_coordinate",

    "longitude": -15.9297,

    "latitude": 82.6924

}

Respuesta

{

    "id\_city": -99,

    "id\_state": -99,

    "id\_village": -99

}

### Microservicio get\_coffee\_recommended\_prices

Este microservicio devuelve un rango de precios recomendados para un producto de café con base a la variedad del mismo y a algunos indicadores económicos actuales.

{

"service": "get\_coffee\_recommended\_prices",

"variety\_name": <nombre\_variedad>

}

**Ejemplo 1**

Cuerpo de la petición

{

    "service": "get\_coffee\_recommended\_prices",

    "variety\_name": "Pacamara"

}

Respuesta:

{

    "higher\_price": 35314.61,

    "lower\_price": 8256.37,

    "recommended\_price": 17657.3

}

### Microservicio get\_user\_trends

Devuelve las tendencias bien sea de caficultores o empresarios con base a los datos de compras/ventas registradas en base de datos

{

"service": "get\_user\_trends",

"id\_role": <id\_rol\_de\_interes>

}

**Ejemplo 1**

Si id\_rol = 1, se devuelve el ranking de caficultores:

Cuerpo de la petición:

{

    "service": "get\_user\_trends",

    "id\_role": 1

}

Respuesta:

{

    "ranking": [

        {

            "id\_user": 13,

            "lots": [

                {

                    "id\_lot": 51,

                    "lot\_photo": "uploads/lot\_photo\_2.png",

                    "profile\_name": "Dulzón",

                    "roasting\_name": "Tueste claro",

                    "sales\_quantity": 129,

                    "variety\_name": "Blue Mountain"

                },

**Ejemplo 2**

Si id\_role = 2, devuelve el ranking de empresarios, a la vez que genera los charts correspondientes para su visualización.

Cuerpo de la petición:

{

    "service": "get\_user\_trends",

    "id\_role": 2

}

Respuesta:

{

    "ranking": [

        {

            "chart\_paths": {

                "chart\_profile\_path": "./generated\_charts/stats\_23\_profile.png",

                "chart\_roast\_path": "./generated\_charts/stats\_23\_roast.png",

                "chart\_variety\_path": "./generated\_charts/stats\_23\_variety.png"

            },

            "id\_user": 23,

            "stats": {

                "profile": [

                    {

                        "color": "#01DE98",

### Microservicio get\_sorted\_lots\_based\_on\_buyer\_preferences

Para un caficultor en particular, devuelve la listas de sus productos ordenados con base a las preferencias de un empresario de su interés.

{

"service": "get\_sorted\_lots\_based\_on\_buyer\_preferences",

"id\_seller": <id\_caficultor>,

"id\_buyer": <id\_empresario>

}

**Ejemplo 1**

Cuerpo de la petición:

{

    "service": "get\_sorted\_lots\_based\_on\_buyer\_preferences",

    "id\_seller": 1,

    "id\_buyer": 36

}

Respuesta:

{

    "sorted\_lots": [

        {

            "cup\_profile\_name": "Cítrico",

            "id\_lot": 1,

            "lot\_number": 1,

            "lot\_photo": "uploads/lot\_photo\_3.png",

            "lot\_price": 7614,

            "lot\_score": null

        },

        {

            "cup\_profile\_name": "Caramelo",

### Microservicio get\_lot\_coding

Con base a las características del café y de su propietario, genera un código alfanumérico único para identificar un producto.

{

"service": "get\_lot\_coding",

"id\_lot": <id\_lote>

}

**Ejemplo 1**

Cuerpo de la petición:

{

    "service": "get\_lot\_coding",

    "id\_lot": 10

}

Respuesta:

{

    "lot\_coding": "AG1-70473-V4P12T2-3N2"

}

### Microservicio service\_get\_farmers\_farm\_info

Dado el id de un caficultor, devuelve una lista de las fincas registradas que le pertenecen

{

"service": "service\_get\_farmers\_farm\_info",

"id\_user": <id\_caficultor>

}

**Ejemplo 1**

Cuerpo de la petición:

{

    "service": "service\_get\_farmers\_farm\_info",

    "id\_user": 8

}

Respuesta:

{

    "1": {

        "altitude": 1830,

        "cilmate": "Húmedo",

        "city\_name": "ACEVEDO",

        "farm\_name": "Finca Helio 1",

        "id\_farm": 14,

        "number\_of\_lots": 3,

        "state\_name": "HUILA",

        "temperature": "16 a 20 °C",

        "village\_name": null

    },

    "2": {

        "altitude": 179,

### Microservicio train\_recommendation\_engine

Entrena el modelo de recomendación de productos

{

"service": "train\_recommendation\_engine"

}

**Ejemplo 1**

Cuerpo de la petición:

{

    "service": "train\_recommendation\_engine"

}

Respuesta:

{

    "success": "model trained"

}

### Microservicio get\_recommended\_lots\_from\_engine

Usa el motor de recomendaciones de productos para devolver una lista de productos que pueden ser de interés a un empresario.

{

"service": "get\_recommended\_lots\_from\_engine",

"id\_user": 10

}

**Ejemplo 1**

Cuerpo de la petición:

{

    "service": "get\_recommended\_lots\_from\_engine",

    "id\_user": 10

}

Respuesta:

{

    "ids\_recommended\_lots": [

        21,

        25,

        44,

        56,

        75,

        83,

        91,

        21,

        23,

        31

    ]

}

## Servicio consume\_businessman\_profile\_data\_microservice

### Microservicio get\_available\_items\_for\_lots

Hace un sondeo por la base de datos y devuelve la disponibilidad de elementos para características de productos, origen y rangos de precio. No requiere parametrizaciones.

{

    "microservice": "get\_available\_items\_for\_lots"

}

Respuesta:

{

    "available\_price\_range": {

        "max\_price": 7967.0,

        "min\_price": 4056.0

    },

    "available\_states": [

        {

            "id\_state": "780",

            "state\_name": "ANTIOQUIA"

        },

        {

            "id\_state": "785",

            "state\_name": "BOLÍVAR"

### Microservicio get\_lot\_information

Obtiene la mayor información posible para un lote específico

{

"microservice": "get\_lot\_information",

"id\_lot": <id\_lote>

}

**Ejemplo 1**

Cuerpo de la petición:

{

    "microservice": "get\_lot\_information",

    "id\_lot": 5

}

Respuesta:

{

    "farm\_info": {

        "farm\_name": "Finca Luis 3",

        "id\_farm": "3"

    },

    "id\_lot": "5",

    "location\_info": {

        "city\_name": "PAILITAS",

        "id\_city": "20001",

        "id\_state": "791",

        "id\_village": "13730",

        "state\_name": "CESAR",

### Microservicio get\_products\_by\_search\_filter

Obtiene productos registrados en base de datos que cumplan con los filtros de búsqueda recibidos.

{

"microservice": "get\_products\_by\_filtered\_search",

"lot\_properties": {

"id\_variety": <ids\_variedades>,

"id\_profile": <ids\_perfiles>,

"id\_roast": <ids\_tipos\_de\_tostion>

},

"price\_range": {

"min\_price": <precio\_minimo>,

"max\_price":<precio\_maximo>

},

"id\_state": <ids\_de\_departamentos>

}

**Ejemplo 1**

Cuerpo de la petición:

{

    "microservice": "get\_products\_by\_filtered\_search",

    "lot\_properties": {

        "id\_variety": [1,2,3,4,5,6,36],

        "id\_profile": [2,3],

        "id\_roast": [1,2]

    },

    "price\_range": {

        "min\_price": 0,

        "max\_price":5000

    },

    "id\_state": [791, 192]

}

Respuesta:

{

    "lots\_information": [

        {

            "farm\_info": {

                "farm\_name": "Finca Maximino 1",

                "id\_farm": "20"

            },

            "id\_lot": "47",

            "location\_info": {

                "city\_name": "RÍO DE ORO",

                "id\_city": "20004",

                "id\_state": "791",

                "id\_village": "13804",

                "state\_name": "CESAR",

**Ejemplo 2**

Cuerpo de la petición:

{

    "microservice": "get\_products\_by\_filtered\_search",

    "lot\_properties": {

        "id\_variety": [1,2,3,4,5,6,36],

        "id\_profile": [2,3],

        "id\_roast": [1,2]

    },

    "price\_range": {

        "min\_price": 0,

        "max\_price":5000

    },

    "id\_state": null

}

Respuesta

{

    "lots\_information": [

        {

            "farm\_info": {

                "farm\_name": "Finca Maximino 1",

                "id\_farm": "20"

            },

            "id\_lot": "47",

            "location\_info": {

                "city\_name": "RÍO DE ORO",

                "id\_city": "20004",

                "id\_state": "791",

**Ejemplo 3**

Cuerpo de la petición:

{

    "microservice": "get\_products\_by\_filtered\_search",

    "lot\_properties": {

        "id\_variety": [1,2,3,4,5,6,36],

        "id\_profile": null,

        "id\_roast": [1,2]

    },

    "price\_range": {

        "min\_price": 5000,

        "max\_price":null

    },

    "id\_state": null

}

Respuesta

{

    "lots\_information": [

        {

            "farm\_info": {

                "farm\_name": "Finca Olga 1",

                "id\_farm": "5"

            },

            "id\_lot": "10",

            "location\_info": {

                "city\_name": "MORROA",

                "id\_city": "20553",

                "id\_state": "807",

                "id\_village": "30066",

                "state\_name": "SUCRE",

                "village\_name": "LA LOMA - LAS BRUJAS"

## Servicio consume\_farmer\_profile\_data\_microservice

Servicio asociado al perfil de caficultor.

### Microservicio get\_farm\_info

Devuelve la información asociada a una finca mediante su id en base de datos.

{

    "microservice": "get\_farm\_info",

    "id\_farm": <id de la finca>

}

**Ejemplo 1**

Cuerpo de la petición:

{

    "microservice": "get\_farm\_info",

    "id\_farm": 1

}

Respueta:

{

    "altitude": 1500,

    "city\_name": "GONZÁLEZ",

    "climate": "Semihúmedo",

    "farm\_name": "Finca Luis 1",

    "farm\_number\_lots": 2,

    "id\_city": 19996,

    "id\_farm": 1,

    "id\_state": 791,

    "id\_user": 1,

    "id\_village": 13591,

    "name\_provided\_by\_user": null,

    "state\_name": "CESAR",

    "temperature": 25,

    "village\_name": "BUJARAVITA"

}

## Servicio consume\_learner\_profile\_data\_microservice

Servicio asociado al perfil de aprendiz.

### Microservicio estimate\_learning\_style

El usuario responde una serie de preguntas, cada una con tres posibilidades de respuesta seleccionable, para determinar su estilo de aprendizaje y mostrarle contenidos de interés en consecuencia. Las respuestas proporcionadas se pasan a este microservicio, el cual retorna el nombre del estilo de aprendizaje correspondiente.

{

    "microservice": " estimate\_learning\_style",

"answers": [

        {

            "id\_question": 1,

            "id\_answer": <id\_respuesta\_de\_pregunta1>

        },

        {

            "id\_question": 2,

            "id\_answer": <id\_respuesta\_de\_pregunta2>

        },

}

**Ejemplo 1**

Cuerpo de la petición:

{

    "microservice": "estimate\_learning\_style",

    "answers": [

        {

            "id\_question": 1,

            "id\_answer":2

        },

        {

            "id\_question": 2,

            "id\_answer":1

        },

        {

            "id\_question": 3,

            "id\_answer":3

        },

        {

            "id\_question": 4,

            "id\_answer":3

        },

        {

            "id\_question": 5,

            "id\_answer":3

        },

        {

            "id\_question": 6,

            "id\_answer":2

        },

        {

            "id\_question": 7,

            "id\_answer":1

        },

        {

            "id\_question": 8,

            "id\_answer":3

        },

        {

            "id\_question": 9,

            "id\_answer":3

        },

        {

            "id\_question": 10,

            "id\_answer":3

        }

    ]

}

Respuesta:

{

    "learning\_style": "cinestésico"

}

### Microservicio make\_a\_chatbot\_query

Dada una pregunta realizada por el aprendiz, la remite a ChatGPT para obtener su respectiva respuesta.

**Ejemplo 1**

Cuerpo de la petición:

{

    "microservice": "make\_a\_chatbot\_query",

    "question": "Qué es la certificación FairTrade?"

}

Respuesta:

{

    "answer": "La certificación Fair Trade es un sello que garantiza que un producto ha sido producido y comercializado siguiendo estándares de comercio justo, que promueven condiciones laborales dignas y sostenibilidad ambiental."

}

**Ejemplo 2**

Si el usuario hace una pregunta que no esté relacionada con la Certificación FairTrade, devuelve un mensaje aclarando que no acepta este tipo de preguntas:

Cuerpo de la petición:

{

    "microservice": "make\_a\_chatbot\_query",

    "question": "Cuál es el tamaño de la Tierra?"

}

Respuesta:

{

    "answer": "Lo siento, solo respondo preguntas relacionadas con la certificación Fair Trade. ¿Tienes alguna consulta sobre ese tema?"

}

### Microservicio obtain\_success\_rate\_of\_evaluation\_results

Obtiene, a partir de las respuestas seleccionadas por el aprendiz durante la evaluación de un submódulo, el porcentaje de acierto basado en las respuestas correctas almacenadas en la base de datos y la validez de las respuestas recibidas. Si se detecta que se debe almacenar un nuevo registro en la base de datos, o si el registro ya existe para el usuario y submódulo en cuestión, se hacen las operaciones respectivas.

Cuerpo petition

{

    "microservice": "obtain\_success\_rate\_of\_evaluation\_results",

"id\_user": <id del aprendiz>,

    "id\_submodule": <id del submodulo>,

    "answers": [

        {

            "id\_question": <id pregunta1>,

            "id\_answer": <id respuesta seleccionada por usuario>

        },

        {

            "id\_question": <id pregunta2>,

            "id\_answer": <id respuesta seleccionada por usuario>

},

...

    ]

}

Cuerpo respuesta

{

    "data\_updated": <true o false según se haya hecho o no una inserción/actualización en base de datos>,

    "succes\_rate": <porcentaje de acierto>

}

**Ejemplo 1**

Cuerpo de la petición:

{

    "microservice": "obtain\_success\_rate\_of\_evaluation\_results",

    "id\_submodule": 1,

    "id\_user": 50,

    "answers": [

        {

            "id\_question": 1,

            "id\_answer": 2

        },

        {

            "id\_question": 2,

            "id\_answer": [5,6]

        },

         {

            "id\_question": 3,

            "id\_answer": 8

        }

    ]

}

Respuesta:

{

    "answers\_validity": {

        "¿Cuál es uno de los objetivos principales de prohibir el trabajo infantil en la certificación Fair Trade?": true,

        "¿Qué medidas toma Fair Trade para prevenir el trabajo infantil?": true,

        "¿Qué prohíbe la certificación Fair Trade en relación al trabajo infantil?": true

    },

    "data\_updated": false,

    "succes\_rate": 100

}

### Microservicio estimate\_course\_progress

Obtiene el grado de avance de completitud del curso por parte de un aprendiz.

Cuerpo petición

{

    "microservice": "estimate\_course\_progress",

    "id\_course": <id del curso>,

    "id\_user": <id del usuario>

}

Cuerpo respuesta

{

    "progress\_rate": <grado de progreso>

}

**Ejemplo 1**

Cuerpo de la petición:

{

    "microservice": "estimate\_course\_progress",

    "id\_course": 1,

    "id\_user": 50

}

Respuesta:

{

    "progress\_rate": 23

}