

---

# CDKDRIVE REPAIR ORDER API

## Developer Guide

### ABOUT

CDKDrive Repair Orders V1 is a transactional API enabling service write-up workflows on CDK Drive. You can perform the following:

- Query open repair orders by Service Advisor ID
- Retrieve a specified repair order
- Open (create) a new repair order (specify customer, vehicle, advisor, transport)
- Query, Add, Update, or Delete Service Lines (which include labor operations and preassigned parts)
- Update Header (change customer, vehicle, advisor, transport details)

<https://apidocs.fortellis.io/specs/bof20c99-ed1f-49de-aoe8-6776d2ee0fe3>

### Key Concepts

#### REPAIR ORDERS

A repair order records maintenance, repairs, or government tests performed on a vehicle for a customer. The repair order includes the list of service items with parts and labor information.

### Potential Use Cases

Use the CDKDrive Repair Orders API to create solutions to handle the following repair order functions:

#### WORKING WITH REPAIR ORDERS

- Query repair orders by service advisor
- Query a repair order
- Update a repair order
- Create a new repair order

#### WORKING WITH SERVICE LINES

- Add a service line to the repair order
- Query a service line
- Update a service line
- Delete a service line

#### QUERYING RELATED DATA

- Query dispatch codes

- Query a dispatch code
- Query dispatch make codes
- Query labor types
- Query a labor type
- Query technicians
- Query a technician
- Query service advisors
- Query a service advisor

## Related APIs

Use the following APIs to collect or create records required for CDKDrive Repair Orders API workflows.

### CDKDRIVE CUSTOMER

Find, retrieve, and filter service customers in CDK Drive. Create a basic customer record.

<https://apidocs.fortellis.io/specs/od488dd1-c609-4618-9061-68be8b61c2a9>

### CDKDRIVE SERVICE VEHICLES

Create, update, query, and retrieve service domain vehicle information in CDK Drive.

<https://apidocs.fortellis.io/specs/54b70ee1-ac17-4be2-9314-45c947692c5d>

### CDKDRIVE OPCODES

Query CDK Drive service opcodes to return data such as the description and hours to complete the operation.

<https://apidocs.fortellis.io/specs/5cef9348-9ced-4f26-9d28-75e08a4714e>

## GETTING STARTED

This topic guides you through making your first test call to a Repair Orders API endpoint. The call from this exercise uses a test subscription against Fortellis API test endpoints, which simply returns sample data from the API Spec. Learn more about testing in Fortellis.

## Prerequisites

Completing this exercise requires the following:

- Familiarity with programming concepts
- An approved Fortellis account
- A solution registered in the Fortellis Developer Network
- The API key and secret from your registered solution

- A tool for testing APIs

**Note:** These instructions guide you through using Postman to make a call to the Repair Orders API, but you can use a similar tool, cURL, or your preferred programming language. You can download postman from [getpostman.com](https://getpostman.com).

If you're already comfortable with the prerequisites, you can skip ahead to Sending a Request to the Repair Orders API. Otherwise, continue through the prerequisites section.

## Fortellis API Test Endpoint

When using the test Subscription-Id to make calls to Fortellis API endpoints, the response is provided only from sample data as defined in the API spec. The data returned is not correlated to or affected by the values in your request.

For more information, see [Test Endpoint](#).

### FORTELLIS ACCOUNT

For help with creating a Fortellis account, see [Getting Started](#).

### SOLUTION AND AUTHORIZATION

Authorization to API test endpoints requires the API key and secret from a registered solution that is connected to the Repair Orders API. You can use a solution you have previously created or register a new one.

For help with registering a new solution, see [Registering Your Solution](#).

For the next few tasks in this section, go to your solution:

1. Log in to the Fortellis Developer Network.
2. Hover over your name, and select [Developer Account](#).
3. To the right of **Solutions\***, select View All\*\*.
4. Select the solution you want to view.

### VIEWING CONNECTED APIS

In the expanded details of your solution, you can review the list of APIs connected to your solution.

- If you see the Repair Orders API, skip ahead to [Recording Your API Key and Secret](#)
- If you do not see the Repair Orders API, proceed to [Connecting the Repair Orders API](#).

### CONNECTING THE REPAIR ORDERS API

To connect the Repair Orders API to your solution:

1. In the menu for your solution, select **Edit**.
2. In the **APIs** tab, search for **CDKDrive Repair Orders V1**, and then select **Update**.

### RECORDING YOUR API KEY AND SECRET

Record your API key and secret to use later in this exercise.

1. In the expanded details of your solution, find the **Other Specifications** section.
2. Next to the value for **API Key**, select the copy to clipboard button.
3. Next to the masked value for **API Secret**, select the show secret button, and then select the copy to clipboard button.

## POSTMAN

These instructions guide you through using Postman to make a call to the Repair Orders API, but you can use a similar tool, cURL, or your preferred programming language. You can download postman from [getpostman.com](https://getpostman.com).

## Sending a Request to the Repair Orders API

As mentioned earlier, Fortellis API test endpoints return sample data as configured in the API Spec. For this exercise, you will use Postman to make a call to the Repair Orders API using the test subscription ID and basic authorization with your API key and secret.

**Note:** Your final solution will use a more sophisticated authorization flow. See [Authentication & Authorization](#) for more information.

## CONFIGURING POSTMAN BUILDING BLOCKS

Create a Postman collection and request.

### CREATING A POSTMAN COLLECTION

1. Open Postman.
2. Select the **New** menu, and then select **Collection**.
3. In the name field, enter **Fortellis Repair Orders API Test**.
4. Select the **Authorization** tab.
5. Set the type to **Basic Auth**.
6. Enter your API key into the username field.
7. Enter your API secret into the password field.
8. Select **Create**.

### CREATING A REQUEST

1. Select the **New** menu, and then select **Request**.
2. In the request name field, enter **Create Repair Order**.
3. Select the collection you created earlier.
4. Select **Save to Fortellis Repair Orders API Test**.

## BUILDING YOUR REQUEST

**Note:** This procedure uses the request body example from the Repair Orders API Create Repair Order endpoint at [Repair Orders API Reference](#).

1. Open the request you created earlier.
2. Set the method to **POST**.
3. Enter the request URL <https://api.fortellis.io/cdkdrive/service/v1/repair-orders/>.

4. Select the Headers tab.
5. Add a new key for Subscription-Id with the value test.
6. Select the **Body** tab.
7. Select **raw**.
8. Paste the following request body into the field.

**Note:** The simulator platform validates the route, method, and basic JSON syntax. It does not validate the contents of your request.

```
{
  "appointmentId": "8888",
  "blockIVRFlag": false,
  "comments": "Use synthetic 5W-20 engine oil",
  "customerHref": "https://api.fortellis.io/cdkdrive/crm/v1/customers/800245",
  "vehicleHref": "https://api.fortellis.io/cdkdrive/service/vehicles/4G263088",
  "customerContactInfo": "8302852330",
  "dropOff": {
    "dateTime": "2018-08-12T20:13:45",
    "address": "678 Montgomery St",
    "city": "Jersey City",
    "note": "Call before pickup",
    "phone": "8302852330",
    "vanNumber": "8302852330"
  },
  "estimate": {
    "authorized": 200,
    "labor": 80,
    "parts": 40,
    "misc": 30,
    "lube": 30,
    "sublet": 10,
    "tax": 10
  },
  "mileageIn": {
    "value": 87665,
    "units": "MI"
  },
  "pickUp": {
    "dateTime": "2018-08-12T20:19:45",
    "address": "678 Montgomery St",
    "city": "Jersey City",
    "note": "Call before pickup",
    "phone": "8302852330",
    "vanNumber": "8302852330"
  },
  "promiseDateTime": "2018-08-12T20:17:45",
  "serviceAdvisorId": "6164",
  "tagNum": "3088",
  "remarks": "sample remarks",
  "transportType": "DROPOFF",
  "serviceLineItems": [
    {
      "cause": "Braking issue",
      "comebackFlag": false,
      "dispatchCode": "S104",
      "estimatedDuration": 1.5,
      "laborEstimate": 10,
      "laborType": "CEC",
    }
  ]
}
```

```

        "lubeEstimate": 10,
        "miscEstimate": 10,
        "partsEstimate": 10,
        "serviceEstimate": 10,
        "status": {
            "code": "I93",
            "description": "PREASSIGNED"
        },
        "serviceRequest": "CUSTOMER STATES THERE IS A CLUNKING SOUND IN FRONT END OVER
BUMPS",
        "subletEstimate": 10,
        "taxEstimate": 10,
        "technicianId": "253",
        "laborOperations": [
            {
                "forceShopChargeFlag": false,
                "laborType": "CEC",
                "opCode": "SSTR",
                "opCodeDesc": "SUSPENSION AND STEERING REPAIR",
                "soldHours": 0.2,
                "saleAmount": 7,
                "saleOverrideFlag": false,
                "includedParts": [
                    {
                        "fixedSaleFlag": false,
                        "includesCoreFlag": false,
                        "mfrCode": "FR",
                        "partNumber": "P9488",
                        "partNumberDescription": "CLUTCH PLATE",
                        "partQty": 1,
                        "saleAmount": 30
                    },
                    {
                        "fixedSaleFlag": false,
                        "includesCoreFlag": false,
                        "mfrCode": "FR",
                        "partNumber": "7488",
                        "partNumberDescription": "BRAKEPAD",
                        "partQty": 2,
                        "saleAmount": 50
                    }
                ]
            }
        ]
    }
}

```

9. Save your request.

## SENDING YOUR REQUEST

In the request you created earlier, select Send. Your request is sent to Fortellis, and the test subscription ID routes the request to the simulator platform.

## EVALUATING YOUR REQUEST

After you send your request, you should see your response body in the lower pane. The response should look like this:

```
{
  "repairOrderId": "5656",
  "status": {
    "code": "I93",
    "description": "PREASSIGNED"
  },
  "links": {
    "self": {
      "href": "https://api.fortellis.io/cdkdrive/service/v1/repair-orders/123"
    }
  },
  "blockIVRFlag": false,
  "comments": "Use synthetic 5W-20 engine oil",
  "customerHref": "https://api.fortellis.io/cdkdrive/crm/v1/customers/800245",
  "customerContactInfo": "8302852330",
  "vehicleHref": "https://api.fortellis.io/cdkdrive/service/vehicles/4G263088",
  "dispatchMakeCode": "C",
  "estimate": {
    "estimateOverrideFlag": false,
    "authorized": 200,
    "labor": 80,
    "parts": 40,
    "misc": 30,
    "lube": 30,
    "sublet": 10,
    "tax": 10
  },
  "mileageIn": {
    "value": 87665,
    "units": "MI"
  },
  "poNumber": "5334",
  "promiseDateTime": "2018-08-12T20:17:45",
  "remarks": "sample remarks",
  "serviceAdvisorId": "6164",
  "tagNum": "3088",
  "transportType": "WAITER",
  "serviceLineItems": [
    {
      "serviceLineId": "A111",
      "cause": "Braking issue",
      "comebackFlag": false,
      "dispatchCode": "S104",
      "estimatedDuration": 1.5,
      "laborEstimate": 10,
      "laborType": "CEC",
      "lubeEstimate": 10,
      "miscEstimate": 10,
      "partsEstimate": 10,
      "serviceEstimate": 10,
      "status": {
        "code": "I93",
```

```

        "description": "PREASSIGNED"
    },
    "serviceRequest": "CUSTOMER STATES THERE IS A CLUNKING SOUND IN FRONT END OVER
BUMPS",
    "subletEstimate": 10,
    "taxEstimate": 10,
    "technicianId": "253",
    "laborOperations": [
        {
            "forceShopChargeFlag": false,
            "laborType": "CEC",
            "opCode": "SSTR",
            "opCodeDesc": "SUSPENSION AND STEERING REPAIR",
            "soldHours": 0.2,
            "saleAmount": 7,
            "saleOverrideFlag": false,
            "includedParts": [
                {
                    "partRefId": "1",
                    "fixedSaleFlag": false,
                    "includesCoreFlag": false,
                    "mfrCode": "FR",
                    "partNumber": "P9488",
                    "partNumberDescription": "CLUTCH PLATE",
                    "partQty": 1,
                    "saleAmount": 30
                },
                {
                    "partRefId": "2",
                    "fixedSaleFlag": false,
                    "includesCoreFlag": false,
                    "mfrCode": "FR",
                    "partNumber": "7488",
                    "partNumberDescription": "BRAKEPAD",
                    "partQty": 2,
                    "saleAmount": 50
                }
            ]
        }
    ]
}

```

*Remember that Fortellis API test endpoints send prefilled sample data as provided in the API Spec. The response will not necessarily correspond to the values from your request.*

If you received an error, use the table below for help troubleshooting.

Error	Possible Cause	Solution
The request could not be satisfied	Incorrect method	Make sure your method is set to <b>POST</b> .



Cannot POST {someURL}	Invalid request URL (endpoint)	Confirm your request URL is correct.
Authorization header required	Misconfiguration of authorization	Make sure your authorization type is set to <b>Basic Auth</b> .
Solution has not requested access to this API. Please select the API from your Developer Account.	Invalid API key in the username field or solution is not connected the this API	Confirm your solution is connected the Repair Orders API and that you have the correct API key from that solution.
Unauthorized	Invalid API secret in the password field	Confirm you have the correct API secret from a solution that is connected to the Repair Orders API.
Subscription-Id header required	Subscription-Id is missing or there is a typo in the key	Confirm you have a header parameter with the key <b>Subscription-Id</b> .
401 Unauthorized	Invalid subscription ID	Confirm the value for your subscription ID header parameter is <b>test</b> .
Unexpected token {someCharacter} in JSON at position {someNumber}	Your json payload is malformed or has errors	Confirm you copied the code sample correctly. If you have trouble copying from this topic, you can get the same sample request from the API reference page.

**Note:** Because of how the API test endpoints work, you can send a request without a JSON payload in the body and still receive a successful response.

## AUTHENTICATION AND AUTHORIZATION FOR THE CDKDRIVE REPAIR ORDERS API

Authentication and authorization for all APIs in Fortellis are managed through the platform. Review the [Authentication & Authorization](#) topic to learn about your options.

## STATUS AND ERROR CODES FOR THE CDKDRIVE REPAIR ORDERS API

Fortellis issues standard status and error codes as outlined in [RFC 7231](#).

### Error Code Listing

The Repair Orders API returns the following codes:

#### POST METHOD ENDPOINTS

Code	Description
201	Details of the successful post
400	Bad Request
401	Unauthorized

403	Forbidden
-----	-----------

#### GET METHOD ENDPOINTS (EXCEPT FOR PING)

Code	Description
200	OK
400	Bad Request
401	Unauthorized
403	Forbidden

#### DELETE METHOD ENDPOINTS

Code	Description
204	No Content
400	Bad Request
401	Unauthorized
403	Forbidden

#### PING ENDPOINT

Code	Description
200	Connection OK
400	Connection Not OK

## Troubleshooting

Use the following tips to help troubleshoot errors.

### 400 - BAD REQUEST

This is likely caused by a malformed request. This error tends to be a bit vague, but you can start by confirming that your request syntax is valid.

### 401 - UNAUTHORIZED OR 403 - FORBIDDEN

There is a problem with authentication or authorization to the API. Check the response body for more information.

Review the [Authentication & Authorization](#) topic for further help.

### 404 - NOT FOUND

This is likely to be a problem with the request URL or method used in your request.

## CDKDRIVE REPAIR ORDERS API WORKFLOWS

This topic provides an overview of workflows related to the Repair Orders API.

## Searching for Repair Orders

A common and simple workflow using the Repair Orders API is to search for repair orders within the dealership service application. You can recall a single repair order if you have the ID, or you can query for a list of repair orders for a specified service advisor.

### GETTING A LIST OF OPEN REPAIR ORDERS

Use the Query Repair Orders endpoint to get a list of all open repair orders and for a specified service advisor. For each result in the response you will find:

- Basic information about the repair order
- A link in the attribute **links.self.href** to details for the repair order, such as labor operations and parts

Here is an example request to get a list of repair orders.

```
curl -X GET \
  'https://api.fortellis.io/cdkdrive/service/v1/repair-
  orders/?serviceAdvisorId={serviceAdvisorId}' \
  -H 'Authorization: {oauth2.0Token}' \
  -H 'subscription-id: {subscriptionId}'
```

To see an example response, test the endpoint in the CDKDrive Repair Orders API Reference.

### RECALLING A SINGLE REPAIR ORDER

Use the Query Repair Order By Id endpoint to get detailed information about a specific repair order. In the response you will find:

- Repair order details
- One or more service line items
- For each service line item, one or more labor operations
- For each labor operation, zero or more parts
- The hyperlink to associate customer details customerHref
- The hyperlink to associate vehicle details vehicleHref
- The direct hyperlink to this record **links.self.href**

Here is an example request to get the details of a single repair order.

```
curl -X GET \
  "https://api.fortellis.io/cdkdrive/service/v1/repair-orders/{repairOrderId}/" \
  -H 'Authorization: {oauth2.0Token}' \
  -H 'subscription-id: {subscriptionId}'
```

To see an example response, test the endpoint in the CDKDrive Repair Orders API Reference.

## Gathering Required Information

Repair orders contain several different types of information that come from various sources. This section explains where to get the information you need to build a repair order request.

**Note:** Important fields are noted below. For a complete list of fields and their definitions, or to see an example request and response, see the [Repair Orders API Spec and Reference](#).

- Repair order header details
  - Customer link
  - Vehicle link
  - Dispatch make code
  - Service advisor ID
- Service line details
  - Dispatch code
  - Technician ID
  - Labor type
- Labor operation details
  - Operation code
  - Operation description
  - Labor type
  - Parts (must come from a list of pre-assigned parts)

### GETTING A CUSTOMER RECORD

Use the [CDKDrive Customer API](#) to get a customer record for your repair order. You can search for existing customers or create a new customer record.

### SEARCHING FOR EXISTING CUSTOMERS

If you have the customer ID, you can recall a single customer using the **Query Customer By Id** endpoint.

Here is an example request where we are searching for a single customer by customerId.

```
curl -X GET \
  "https://api.fortellis.io/cdkdrive/crm/v1/customers/{customerId}/" \
  -H 'Authorization: {oauth2.0Token}' \
  -H 'subscription-id: {subscriptionId}'
```

If you don't know the customer ID, you can use the **Query Customers** endpoint to get a list of customers matching at least one of the following:

- Name of the customer
- Phone number of the customer
- Email address of the customer

Here is an example request where we are searching for a list of customers with the last name of **DALTON**.

```
curl -X GET \
```

```
"https://api.fortellis.io/cdkdrive/crm/v1/customers/?name=DALTON" \
-H 'Authorization: {oauth2.0Token}' \
-H 'subscription-id: {subscriptionId}'
```

If the customer exists in CDK Drive, record the hyperlink for the customer from the response field **links.self.href**. You will use this link when creating your repair order.

## CREATING A NEW CUSTOMER

Create a new customer to use with your repair order using the Create Customer endpoint.

When you create your new customer, record the hyperlink for the customer from the response field **links.self.href**. You will use this link when creating your repair order.

## GETTING A VEHICLE RECORD

Use the [CDKDrive Service Vehicles API](#) to get a vehicle record for your repair order. You can search for existing vehicles or create a new vehicle record.

This section provides a basic overview of the CDKDrive Service Vehicles API for use with creating repair orders.

## SEARCHING FOR EXISTING VEHICLES

If you have the vehicle ID, you can recall a single vehicle using the **Query Vehicle By Id** endpoint.

Here is an example request where we are searching for a single vehicle by **customerId**.

```
curl -X GET \
  "https://api.fortellis.io/cdkdrive/service/v1/vehicles/{vehicleId}/" \
  -H 'Authorization: {oauth2.0Token}' \
  -H 'subscription-id: {subscriptionId}'
```

If you don't know the vehicle ID, you can use the **Query Vehicles** endpoint to get a list of vehicles matching at least one of the following:

- VIN number
- License plate number
- Customer ID

Here is an example request where we are searching for a list of vehicles with a customer ID of **001234**.

```
curl -X GET \
  "https://api.fortellis.io/cdkdrive/service/v1/vehicles/?customerId=001234" \
  -H 'Authorization: {oauth2.0Token}' \
  -H 'subscription-id: {subscriptionId}'
```

If the vehicle exists in CDK Drive, record the hyperlink for the vehicle from the response field **links.self.href**. You will use this link when creating your repair order.

## CREATING A NEW VEHICLE

You can create a new vehicle to use with your repair order using the **Create Vehicle** endpoint.

When you create your new vehicle, record the hyperlink for the vehicle from the response field **links.self.href**. You will use this link when creating your repair order.

## GETTING DISPATCH MAKE CODES

Get dispatch make codes from the [Repair Orders API](#) Query Dispatch Make Codes endpoint.

Record the value from the **dispatchMakeCode** field for record you need.

## GETTING SERVICE ADVISOR IDS

Get service advisor IDs from the [Repair Orders API](#) Query Service Advisors or Query Service Advisor By Id endpoint.

Record the value from the **serviceAdvisorId** field for record you need.

## GETTING DISPATCH CODES

Get dispatch codes from the [Repair Orders API](#) Query Dispatch Codes or Query Dispatch Code By Id endpoint.

Record the value from the **dispatchCode** field for record you need.

## GETTING TECHNICIAN IDS

Get technician IDs from the [Repair Orders API](#) Query Technicians or Query Technician By Id endpoint.

Record the value from the **technicianId** field for record you need.

## GETTING LABOR TYPES

Get labor types from the [Repair Orders API](#) Query Labor Types or Query Labor Type By Id endpoint.

Record the value from the **laborType** field for record you need.

## GETTING LABOR OPERATION CODES

This section provides a basic overview of the CDKDrive OpCodes API for use with creating repair orders

Get labor operation codes (OpCodes) from the [CDKDrive OpCodes API](#).

If you have the OpCode for a particular labor operation, you can recall that record using the **Query Op Code By Code** endpoint.

Here is an example request where we are searching for a single service operation code with the OpCode **ABC**.

```
curl -X GET \
  "https://api.fortellis.io/cdkdrive/service/catalog/v1/opcodes/ABC" \
  -H 'Authorization: {oauth2.0Token}' \
  -H 'subscription-id: {subscriptionId}'
```

If you don't have the OpCode, you can use the **Query Op Codes** endpoint to get a list of labor operations matching a text filter on the description.

**Note:** Wildcards are implicit at the beginning and end of your text filter value.

Here is an example request where we are searching for all service operation codes including the phrase **Air Filter**.

```
curl -X GET \
  "https://api.fortellis.io/cdkdrive/service/catalog/v1/opcodes/?desc="Air
  Filter"&page=&pageSize=" \
  -H 'Authorization: {oauth2.0Token}' \
  -H 'subscription-id: {subscriptionId}'
```

Record the values from the **opCode** and **opCodeDesc** fields for record you need.

## Creating a Repair Order

Combine the information you collected in the section Gathering Required Information with other details for your repair order and its service lines.

Review the [Repair Orders API Spec and Reference](#) for a list of all fields and their descriptions.

Use the **Create Repair Order** endpoint to send your request.

Here is an example request to create a repair order.

```
curl -X POST \
  "https://api.fortellis.io/cdkdrive/service/v1/repair-orders/001234/" \
  -H "If-Match: {if-MatchETagString}" \
  -H 'Authorization: {oauth2.0Token}' \
  -H 'subscription-id: {subscriptionId}' \
  -d '{
    "promiseDateTime": "2018-08-12T20:17:45",
    "serviceLineItems": [
      {
        "taxEstimate": 10,
        "laborType": "CEC",
        "partsEstimate": 10,
        "miscEstimate": 10,
        "subletEstimate": 10,
        "laborEstimate": 10,
        "cause": "Braking issue",
        "dispatchCode": "S104",
        "lubeEstimate": 10,
        "estimatedDuration": 1.5,
        "serviceLineId": "A11",
        "laborOperations": [
          {
            "opCodeDesc": "SUSPENSION AND STEERING REPAIR",
            "saleAmount": 7,
            "laborType": "CEC",
            "opCode": "SSTR",
            "saleOverrideFlag": false,
            "soldHours": 0.2,
            "forceShopChargeFlag": false,
            "includedParts": [
              {
                "saleAmount": 30,
```

```

        "fixedSaleFlag": false,
        "partNumberDescription": "CLUTCH PLATE",
        "partRefId": "1",
        "includesCoreFlag": false,
        "mfrCode": "FR",
        "partNumber": "P9488",
        "partQty": 1
    },
    {
        "saleAmount": 50,
        "fixedSaleFlag": false,
        "partNumberDescription": "BRAKEPAD",
        "partRefId": "2",
        "includesCoreFlag": false,
        "mfrCode": "FR",
        "partNumber": "7488",
        "partQty": 2
    }
]
},
"comebackFlag": false,
"serviceEstimate": 10,
"serviceRequest": "CUSTOMER STATES THERE IS A CLUNKING SOUND IN FRONT END OVER
BUMPS",
"technicianId": "253",
"status": {
    "description": "PREASSIGNED",
    "code": "I93"
}
}
],
"comments": "Use synthetic 5W-20 engine oil",
"vehicleHref": "https://api.fortellis.io/cdkdrive/service/vehicles/4G263088",
"customerHref": "https://api.fortellis.io/cdkdrive/crm/v1/customers/800245",
"mileageIn": {
    "value": 87665,
    "units": "MI"
},
"repairOrderId": "5656",
"blockIVRFlag": false,
"serviceAdvisorId": "6164",
>tagNum": "3088",
"estimate": {
    "estimateOverrideFlag": false,
    "authorized": 200,
    "sublet": 10,
    "parts": 40,
    "tax": 10,
    "lube": 30,
    "labor": 80,
    "misc": 30
},
"links": {
    "self": {
        "href": "https://api.fortellis.io/cdkdrive/service/v1/repair-orders/123"
    }
},
"dispatchMakeCode": "C",

```



```
"transportType": "WAITER",
"customerContactInfo": "8302852330",
"poNumber": "5334",
"remarks": "sample remarks",
"status": {
  "description": "PREASSIGNED",
  "code": "I93"
}
}'
```

A successful call includes a **201 Created** response and the repair order object.

## Updating a Repair Order

You can update existing repair orders as follows:

- Update header information (such as comments, remarks, tag numbers, etc.) using the Update Repair Order endpoint
- Add a new service line using the Add Service Line endpoint
- Update an existing service line using the Update Service Line endpoint
- Delete an existing service line using the Delete Service Line endpoint

### UPDATING REPAIR ORDER HEADER INFORMATION

Here is an example request to update a repair order having a `repairOrderId` of **001234**.

```
curl -X POST \
  "https://api.fortellis.io/cdkdrive/service/v1/repair-orders/001234/" \
  -H "If-Match: {if-MatchETagString}" \
  -H 'Authorization: {oauth2.0Token}' \
  -H 'subscription-id: {subscriptionId}' \
  -d '{
    "blockIVRFlag": false,
    "comments": "Use synthetic 5W-20 engine oil",
    "customerHref": "https://api.fortellis.io/cdkdrive/crm/v1/customers/800245",
    "vehicleHref": "https://api.fortellis.io/cdkdrive/service/vehicles/4G263088",
    "customerContactInfo": "8302852330",
    "dropOff": {
      "dateTime": "2018-08-12T20:13:45",
      "address": "678 Montgomery St",
      "city": "Jersey City",
      "note": "Call before pickup",
      "phone": "8302852330",
      "vanNumber": "8302852330"
    },
    "estimate": {
      "authorized": 200,
      "labor": 80,
      "parts": 40,
      "misc": 30,
      "lube": 30,
      "sublet": 10,
      "tax": 10
    },
    "mileageIn": {
      "value": 87665,
      "units": "KM"
    }
  }
```

```

    },
    "pickUp": {
      "dateTime": "2018-08-12T20:19:45",
      "address": "678 Montgomery St",
      "city": "Jersey City",
      "note": "Call before pickup",
      "phone": "8302852330",
      "vanNumber": "8302852330"
    },
    "promiseDateTime": "2018-08-12T20:17:45",
    "serviceAdvisorId": "6164",
    "tagNum": "3088",
    "remarks": "sample remarks",
    "transportType": "DROPOFF"
  }
}

```

A successful response includes the repair order object with updated values.

## ADDING A NEW SERVICE LINE

Here is an example request to add a service line to a repair order having a repairOrderId of **001234**.

```

curl -X POST \
  "https://api.fortellis.io/cdkdrive/service/v1/repair-orders/001234/service-lines" \
  -H "If-Match: {if-MatchETagString}" \
  -H 'Authorization: {oauth2.0Token}' \
  -H 'subscription-id: {subscriptionId}' \
  -d '{
    "cause": "Unusual sound occurring when clutch is engaged",
    "comebackFlag": false,
    "dispatchCode": "S102",
    "estimatedDuration": 1.5,
    "laborEstimate": 10,
    "laborType": "CEC",
    "lubeEstimate": 10,
    "miscEstimate": 10,
    "partsEstimate": 10,
    "serviceEstimate": 10,
    "status": {
      "code": "I93",
      "description": "PREASSIGNED"
    },
    "serviceRequest": "Investigate sound coming from clutch",
    "subletEstimate": 10,
    "taxEstimate": 10,
    "technicianId": "253",
    "laborOperations": [
      {
        "forceShopChargeFlag": false,
        "laborType": "CEC",
        "opCode": "SSTR",
        "opCodeDesc": "SUSPENSION AND STEERING REPAIR",
        "soldHours": 0.2,
        "saleAmount": 7,
        "saleOverrideFlag": false,
        "includedParts": [
          {
            "fixedSaleFlag": false,

```

```

    "includesCoreFlag": false,
    "mfrCode": "FR",
    "partNumber": "P9488",
    "partNumberDescription": "CLUTCH PLATE",
    "partQty": 1,
    "saleAmount": 30
  },
  {
    "fixedSaleFlag": false,
    "includesCoreFlag": false,
    "mfrCode": "FR",
    "partNumber": "7488",
    "partNumberDescription": "BRAKEPAD",
    "partQty": 2,
    "saleAmount": 50
  }
]
}
]
}',

```

A successful response includes the service line object.

## UPDATING AN EXISTING SERVICE LINE

Here is an example request to update an existing service line having a `serviceLineId` of 321 to a repair order having a `repairOrderId` of 001234.

```

curl -X POST \
  "https://api.fortellis.io/cdkdrive/service/v1/repair-orders/001234/service-lines/321/" \
  -H "If-Match: {if-MatchETagString}" \
  -H 'Authorization: {oauth2.0Token}' \
  -H 'subscription-id: {subscriptionId}' \
  -d '{
    "cause": "Braking issue",
    "comebackFlag": false,
    "dispatchCode": "S104",
    "estimatedDuration": 1.5,
    "laborEstimate": 10,
    "laborType": "CEC",
    "lubeEstimate": 10,
    "miscEstimate": 10,
    "partsEstimate": 10,
    "serviceEstimate": 10,
    "subletEstimate": 10,
    "taxEstimate": 10,
    "status": {
      "code": "I93",
      "description": "PREASSIGNED"
    },
    "serviceRequest": "CUSTOMER STATES THERE IS A CLUNKING SOUND IN FRONT END OVER BUMPS",
    "technicianId": "253",
    "laborOperations": [
      {
        "laborOperationId": "11",
        "forceShopChargeFlag": false,
        "laborType": "CEC",

```

```

"opCode": "SSTR",
"opCodeDesc": "SUSPENSION AND STEERING REPAIR",
"soldHours": 0.2,
"saleAmount": 7,
"saleOverrideFlag": false,
"includedParts": [
  {
    "partRefId": "1",
    "fixedSaleFlag": false,
    "includesCoreFlag": false,
    "mfrCode": "FR",
    "partNumber": "P9488",
    "partNumberDescription": "CLUTCH PLATE",
    "partQty": 1,
    "saleAmount": 30
  },
  {
    "partRefId": "2",
    "fixedSaleFlag": false,
    "includesCoreFlag": false,
    "mfrCode": "FR",
    "partNumber": "7488",
    "partNumberDescription": "BRAKEPAD",
    "partQty": 2,
    "saleAmount": 50
  }
]
}
}
]
}'

```

A successful response includes the repair order object with updated values.

## DELETING AN EXISTING SERVICE LINE

Here is an example request to delete an existing service line having a `serviceLineId` of **987** to a repair order having a `repairOrderId` of **001234**.

```
curl -X POST \
  "https://api.fortellis.io/cdkdrive/service/v1/repair-orders/001234/service-lines/987/" \
  -H "If-Match: {if-MatchETagString}" \
  -H 'Authorization: {oauth2.0Token}' \
  -H 'subscription-id: {subscriptionId}'
```

A successful deletion returns a 204 No Content response.

## USING THE CDKDRIVE REPAIR ORDERS API WITH CDK DRIVE

For field definitions, see the [CDKDrive Repair Orders API Reference](#).

Consider the following details to use the CDKDrive Repair Orders API with the CDK Drive DMS.

## Estimates

The repair order and service line estimates are the estimated charges to the customer for work performed on a vehicle. These estimated values are provided to the customer on the repair order document.

The repair order `authorized` field is the total estimate for the repair order and is printed on most work orders.

The Service Line `serviceEstimate` field is the total service estimate for the particular service line. Each line can have its own estimate.

## Estimate Dealer Configuration

The CDK Drive DMS provides a configuration option for each dealer to make the Repair Order Estimate optional or required under Drive Service account function UES (Update Estimating Options). This UES Function allows for dealers to set predefined Estimating options in Drive. Please consult with each dealer to understand the configuration setup for each store.

### REPAIR ORDER ESTIMATE AMOUNTS

When preparing the `estimate` object for a repair order, the amount in `authorized` must match the sum of the other estimate fields:

- `labor`
- `parts`
- `misc`
- `lube`
- `sublet`
- `tax`

If the amounts do not match, CDK Drive returns an error and will not create or update the record.

### SERVICE LINE ITEM ESTIMATE AMOUNTS

When preparing the `serviceLineItems` object for a repair order, the amount in `serviceEstimate` must match the sum of the other service line item estimate fields:

- `laborEstimate`
- `lubeEstimate`
- `miscEstimate`
- `partsEstimate`
- `subletEstimate`
- `taxEstimate`

If the amounts do not match, CDK Drive returns an error and will not create or update the record.

## ESTIMATE TOTALS HANDLING

The CDKDrive Repair Orders API compares the value in the `authorized` field to the sum of the values in the service line item `serviceEstimate` fields.

If the sum of the `serviceEstimate` fields matches the value in the `authorized` field, the `estimateOverrideFlag` field is sent with a value of `false`, and the repair order is created normally.

If the sum of the `serviceEstimate` fields does not match the value in the `authorized` field, the `estimateOverrideFlag` field is sent with a value of `true`, and the repair order is created with an override. The service advisor must then adjust the estimates in CDK Drive when working on the repair order.

## PRINTER SETUP

You can assign one printer per store as the work order printer. If enabled, this configuration allows you to print a work order to that specified printer when the work order is created. If you have document archiving (DSDA) enabled for the store, the document is also archived at this time.

Each store can have only 1 work order printer and only 1 parts printer configured. To configure a work order printer, you must provide the following information to Fortellis who will configure this on each dealer subscription.

- Indication if the work order is to print or not (Y or N)
- Work order printer number
- Number of work order copies to print
- Pre-assigned parts printer Name (only used if pre-assigned parts are passed on the create repair order request)

**Note:** If your configurations are invalid for the work order printer number or the number of work order copies to print, CDK Drive returns an error.