

I. Synopsys

High-level description of what a data layer is:

<https://developers.google.com/tag-platform/tag-manager/datalayer>

We will be following GA4 dataLayer format

<https://developers.google.com/analytics/devguides/collection/ga4/reference/events>

+ some additions for user data and/or custom parameters

Note: For marketing platforms (Facebook, Google ADS, TikTok, Snapchat, etc.) it is extremely important to have user data along with events.

Therefore, it is very desirable to add this data on events where the user has already specified it or was, like, authorized on the site.

All user data that can be used and its format can be found at the very end of this document.

We propose not to remove current events from the dataLayer, but to add new events alongside them. This way we will not interfere with the current tracking during the transition. In the future the old data layer events can be removed from the code.

To do this, we propose to simply use custom 'event' parameters in dataLayer events. That is for example instead of view_item use view_item_v2 or view_item_new - any of these or similar variants.

The structure, locations and format of the data layer are described below.

II. Data Layer structure

view_item_list

- When viewing an collection / category page with product list

JavaScript

```
dataLayer = window.dataLayer || [];
```

```

dataLayer.push({ecommerce: null}); // we want to null out the ecommerce object,
so there's no overlap if events happen on the same page
dataLayer.push({
  event: 'view_item_list',
  ecommerce: {
    value: 12.34, // Number, two decimals, required
    currency: 'USD', // String, required
    items: [
      {
        item_name: 'Mock Item name', // String, required
        item_id: '1235', // String, required
        item_sku: 'sku_123', // String, required
        price: 12.34, // Number, two decimals, required
        quantity: 1, // Integer, required
        item_category: 'Mock Item category', // String, optional but advised
        if available
        item_variant: 'Mock Item variant', // String, optional
        item_brand: 'Brand name', // String, optional
        image_url: 'https://yourdomain.com/files/1235.png' //Link to product
        main image
      },
      {
        //next item from list...up to 10 items
      }
    ]
  },
  user_data: { // we're going to have this object with all events that
    have user-data already available, for example if user is logged in
    first_name: 'John',
    last_name: 'Mikros',
    email_address: 'john@stape.io',
    user_id: '12345', // this is if you have a consistent internal
    user-id, recommended
    customer_status: 'new'
    //and any other available user data
  }
});

```

view_item

- When viewing an product page

JavaScript

```
dataLayer = window.dataLayer || [];  
dataLayer.push({ecommerce: null}); // we want to null out the ecommerce object,  
so there's no overlap if events happen on the same page  
dataLayer.push({  
  event: 'view_item_v2',  
  ecommerce: {  
    value: 12.34, // Number, two decimals, required  
    currency: 'USD', // String, required  
    items: [  
      {  
        item_name: 'Mock Item name', // String, required  
        item_id: '1235', // String, required  
        item_sku: 'sku_123', // String, required  
        price: 12.34, // Number, two decimals, required  
        quantity: 1, // Integer, required  
        item_category: 'Mock Item category', // String, optional but advised  
        if available  
        item_variant: 'Mock Item variant', // String, optional  
        item_brand: 'Brand name', // String, optional  
        image_url: 'https://yourdomain.com/files/1235.png' //Link to product  
        main image  
      }  
    ],  
  },  
  user_data: { // we're going to have this object with all events that have  
    user-data  
    already available, for example if user is logged in  
    first_name: 'John', // I'm outlining the maximum PII list, you  
    obviously need to push only what's available  
    last_name: 'Mikros',  
    email_address: 'john@stape.io',  
    phone_number: "1555555555", // Remove symbols, letters, and any  
    leading zeros. Phone numbers must include a country code  
    country: "US", // 2-letter country codes in ISO 3166-1 alpha-2.  
    city: "Wilmington",  
    state: 'DE',  
    postal_code: "19808", // no spaces and no dash  
    user_id: '12345', // this is if you have a consistent internal  
    user-id, recommended  
    customer_status: 'new' // user has done at least 1 purchase ?  
    'returning' : 'new'  
  }  
});
```

add_to_cart

- When the user adds the product to the basket;
- Since after clicking on add to cart the page reloads - it is necessary that this event is triggered after reloading, so that all event tags have time to work and send data.

JavaScript

```
dataLayer = window.dataLayer || [];  
dataLayer.push({ecommerce: null}); // we want to null out the ecommerce object,  
so there's no overlap if events happen on the same page  
dataLayer.push({  
  event: 'add_to_cart_v2',  
  ecommerce: {  
    value: 12.34, // Number, two decimals, required  
    currency: 'USD', // String, required  
    items: [  
      {  
        item_name: 'Mock Item name', // String, required  
        item_id: '1235', // String, required  
        item_sku: 'sku_123', // String, required  
        price: 12.34, // Number, two decimals, required  
        quantity: 1, // Integer, required  
        item_category: 'Mock Item category', // String, optional but advised  
        if available  
        item_variant: 'Mock Item variant', // String, optional  
        item_brand: 'Brand name', // String, optional  
        image_url: 'https://yourdomain.com/files/1235.png' //Link to product  
        main image  
      }  
    ],  
  },  
  user_data: { // we're going to have this object with all events that have  
    user-data already available, for example if user is logged in  
    first_name: 'John', // I'm outlining the maximum PII list, you  
    obviously need to push only what's available  
    last_name: 'Mikros',  
    email_address: 'john@stape.io',  
    phone_number: "1555555555", // Remove symbols, letters, and any  
    leading zeros. Phone numbers must include a country code
```

```

        country: "US", // 2-letter country codes in ISO 3166-1 alpha-2.
        city: "Wilmington",
        state: 'DE',
        postal_code: "19808", // no spaces and no dash
        user_id: '12345', // this is if you have a consistent internal
user-id, recommended
        customer_status: 'new' // user has done at least 1 purchase ?
'returning' : 'new'
    }
});

```

remove_from_cart

- When user removes the product from the basket;

```

JavaScript
dataLayer = window.dataLayer || [];
dataLayer.push({ecommerce: null}); // we want to null out the ecommerce object,
so there's no overlap if events happen on the same page
dataLayer.push({
    event: 'remove_from_cart_v2',
    ecommerce: {
        value: 12.34, // Number, two decimals, required
        currency: 'USD', // String, required
        items: [
            {
                item_name: 'Mock Item name', // String, required
                item_id: '1235', // String, required
                item_sku: 'sku_123', // String, required
                price: 12.34, // Number, two decimals, required
                quantity: 1, // Integer, required
                item_category: 'Mock Item category', // String, optional but advised
if available
                item_variant: 'Mock Item variant', // String, optional
                item_brand: 'Brand name', // String, optional
                image_url: 'https://yourdomain.com/files/1235.png' //Link to product
main image
            }
        ]
    }
});

```

```

    ]
  },
  user_data: { // we're going to have this object with all events that have
    user-data already available, for example if user is logged in
    first_name: 'John', // I'm outlining the maximum PII list, you
    obviously need to push only what's available
    last_name: 'Mikros',
    email_address: 'john@stape.io',
    phone_number: "1555555555", // Remove symbols, letters, and any
    leading zeros. Phone numbers must include a country code
    country: "US", // 2-letter country codes in ISO 3166-1 alpha-2.
    city: "Wilmington",
    state: 'DE',
    postal_code: "19808", // no spaces and no dash
    user_id: '12345', // this is if you have a consistent internal
    user-id, recommended
    customer_status: 'new' // user has done at least 1 purchase ?
    'returning' : 'new'
  }
});

```

begin_checkout

- Should trigger when the user goes to first checkout step;
- Items reflect full cart state.

JavaScript

```

dataLayer = window.dataLayer || [];
dataLayer.push({ecommerce: null}); // we want to null out the ecommerce object,
so there's no overlap if events happen on the same page
dataLayer.push({
  event: 'begin_checkout',
  ecommerce: {
    value: 12.34, // Number, two decimals, required, total value of cart
    currency: 'USD', // String, required
    items: [
      {
        item_name: 'Mock Item name', // String, required

```

```

        item_id: '1235', // String, required
        price: 12.34, // Number, two decimals, required
        quantity: 1, // Integer, required
        item_category: 'Mock Item category', // String, optional but advised
        if available
            item_brand: 'Mock Item brand', // String, optional, might be useful
        if you sell different brands
            item_variant: 'Mock Item variant', // String, optional
            image_url: 'https://yourdomain.com/files/1235.png' //Link to product
        main image
    }
  ],
  user_data: { // we're going to have this object with all events that have
    user-data already available, for example if user is logged in
      first_name: 'John',
      last_name: 'Mikros',
      email_address: 'john@stape.io',
      user_id: '12345', // this is if you have a consistent internal
    user-id, recommended
      customer_status: 'new'
    //and any other available user data
  }
});

```

add_payment_info

- Should trigger after user load the page with payment info step on checkout flow.
- User data is always available in this step - so it should always be in the data layer in this step. Should be the data of the user who makes the booking, not of the visitors
- Items reflects full cart state.

JavaScript

```
dataLayer = window.dataLayer || [];
```

```

dataLayer.push({ecommerce: null}); // we want to null out the ecommerce object,
so there's no overlap if events happen on the same page
dataLayer.push({
  event: 'add_shipping_info',
  ecommerce: {
    value: 12.34, // Number, two decimals, required, total value of cart
    currency: 'USD', // String, required
    items: [
      {
        item_name: 'Mock Item name', // String, required
        item_id: '1235', // String, required
        price: 12.34, // Number, two decimals, required
        quantity: 1, // Integer, required
        item_category: 'Mock Item category', // String, optional but advised
        if available
        item_brand: 'Mock Item brand', // String, optional, might be useful
        if you sell different brands
        item_variant: 'Mock Item variant', // String, optional
        image_url: 'https://yourdomain.com/files/1235.png' //Link to product
        main image
      }
    ],
  },
  user_data: { // we're going to have this object with all events that have
    user-data already available, for example if user is logged in
    first_name: 'John', // I'm outlining the maximum PII list, you
    obviously need to push only what's available
    last_name: 'Mikros',
    email_address: 'john@stape.io',
    phone_number: "1555555555", // Remove symbols, letters, and any
    leading zeros. Phone numbers must include a country code
    country: "US", // 2-letter country codes in ISO 3166-1 alpha-2.
    city: "Wilmington",
    state: 'DE',
    postal_code: "19808", // no spaces and no dash
    user_id: '12345', // this is if you have a consistent internal
    user-id, recommended
    customer_status: 'new' // user has done at least 1 purchase ?
    'returning' : 'new'
  }
});

```

purchase

- When purchase is complete, usually fired on the thank-you page. **Make sure it fires once (don't push if I refresh the page);**
- Items reflects full cart state;
- Additional fields like transaction_id, tax, etc.;
- As this is a main conversion - it is essential that as much user data as possible is available.

JavaScript

```
dataLayer = window.dataLayer || [];  
dataLayer.push({ecommerce: null}); // we want to null out the ecommerce object,  
so there's no overlap if events happen on the same page  
dataLayer.push({  
  event: 'purchase',  
  ecommerce: {  
    value: 22.64, // Number, two decimals, sum of transaction. required  
    currency: 'USD', // String, required  
    tax: 2.00, // Number, two decimals, required. Push 0.00 if  
non-applicable  
    shipping: 0.00, // Number, two decimals, required. Push 0.00 if  
non-applicable  
    transaction_id: '7890410RD', // String, required, unique identifier  
of order/transaction  
    coupon: 'My mock coupon', // String, optional, push if available  
    payment_type: 'Visa', // String, optional, push if available  
    items: [  
      {  
        item_name: 'Mock Item name', // String, required  
        item_id: '1235', // String, required  
        price: 12.34, // Number, two decimals, required  
        quantity: 1, // Integer, required  
        item_category: 'Mock Item category', // String, optional but advised  
if available  
        item_brand: 'Mock Item brand', // String, optional, might be useful  
if you sell different brands  
        item_variant: 'Mock Item variant', // String, optional  
        image_url: 'https://yourdomain.com/files/1235.png' //Link to product  
main image  
      }  
    ]  
  }  
});
```

```

    }
  ],
  },
  user_data: { // we're going to have this object with all events that have
user-data already available, for example if user is logged in
    first_name: 'John', // I'm outlining the maximum PII list, you
obviously need to push only what's available
    last_name: 'Mikros',
    email_address: 'john@stape.io',
    phone_number: "1555555555", // Remove symbols, letters, and any
leading zeros. Phone numbers must include a country code
    country: "US", // 2-letter country codes in ISO 3166-1 alpha-2.
    city: "Wilmington",
    state: 'DE',
    postal_code: "19808", // no spaces and no dash
    user_id: '12345', // this is if you have a consistent internal
user-id, recommended
    customer_status: 'new' // user has done at least 1 purchase ?
'returning' : 'new'
  }
});

```

lead_submitted

After submit lead forms

- In user_data, add any available relevant user data from the form

```

JavaScript
dataLayer.push({
  event: 'lead_submitted_v2',
  lead_form: 'contact_us', //optional, some form id or name if needed
  user_data: { // user data from lead form
    first_name: 'John',
    last_name: 'Mikros',
    email_address: 'john@stape.io'
  }
});

```

login

- Must be triggered after user logged in

```
JavaScript
dataLayer.push({
  event: 'login_v2',
  user_data: { // user data from account
    first_name: 'John',
    last_name: 'Mikros',
    email_address: 'john@stape.io',
    user_id: '12345', // this is if you have a consistent internal
user-id, recommended
  }
});
```

Elements click events

You have a set of events that work simply by clicking on elements. Ideally you should also have them in the data layer so that the tracking does not depend on changes on the frontend.

These data layer pushes can only contain the 'event' parameter, for example:

- buy_button

```
JavaScript
dataLayer.push({
  event: 'buy_button'
});
```

III. User data

Below is a table with all the user data that can be used in any marketing platforms and the format of their use.

Name	Format
Email	-
Phone	Remove symbols, letters, and any leading zeros. Phone numbers must include a country code. Example: Input: US phone number (650)555-1212 Normalized format: 16505551212
First name	-
Last name	-
Date of birth	YYYYMMDD Example: Input: 2/16/1997 Normalized format: 19970216
Gender	Example: f for female m for male
City	-
State/Region	All lowercase letters Exception: 2-letter abbreviated state names for the US
Street	-
Zip code	Use lowercase with no spaces and no dash.
Country	Use the lowercase, 2-letter country codes in ISO 3166-1 alpha-2. Example: Input: United States Normalized format: us

Name	Format
User Id	Any unique ID from the advertiser, such as loyalty membership IDs, user IDs
New customer	True or false
Customer lifetime value	The amount for which the user has made purchases for the whole time. Number with dot Example: 223.69