

# Einfache Marketing-Automation mit Mailchimp & Mandrill

Versende ab sofort automatisierte  
Eventreminder, Dankesmails u. v. m.!



TICKETINO.  
Everybody's Ticketing



## Inhalt dieser Anleitung kurz zusammengefasst

Im Rahmen dieser Anleitung wollen wir dir Schritt für Schritt zeigen, wie du deine Kontaktdaten aus dem TICKETINO-Cockpit via Webhook & Zapier an Mandrill übermitteln kannst. Mandrill ist eine Erweiterung zu Mailchimp. Falls du ein anderes Tool zur Marketingautomation nutzt, prüfe doch [hier](#), ob Zapier dieses ebenfalls unterstützt. Falls ja, dann sollte ein analoges Vorgehen möglich sein.

Diese Anleitung richtet sich an etwas versiertere Anwender/innen, da du ein klein wenig «Codieren» oder zumindest etwas im Code anpassen musst. Falls dir dies zu technisch, empfehlen wir dir die einfache Synchronisation deiner Kontakte mit Mailchimp. Von dort aus kannst du ebenfalls einfache Marketingautomation betreiben und bei Bedarf auch jederzeit die Mandrill-Erweiterung hinzufügen. Die Anleitung findest du neben vielen weiteren in der [TICKETINO-Academy](#) unter der Rubrik Whitepaper oder gleich hier unter diesem [Link](#).

Bitte sei dir bewusst, dass die nachfolgend beschriebenen Prozesse nicht unbegrenzt mit den Gratislösungen von Mailchimp und Zapier funktionieren. Nach einer allfälligen Testphase sind also mit «moderaten» Lizenzkosten zu rechnen. Falls du dies nicht investieren möchtest, kannst du deine Kontakte natürlich auch über den Export der Ticketliste übertragen und deine Mails manuell senden.

## Schritt 1: Erstelle sowohl einen Zapier wie auch einen Mailchimp Account

Wir gehen an dieser Stelle davon aus, dass du das TICKETINO-System bereits für deinen Ticketverkauf nutzt. Falls nicht, müsstest du [hier ebenfalls noch einen Veranstalter-Account](#) erstellen.

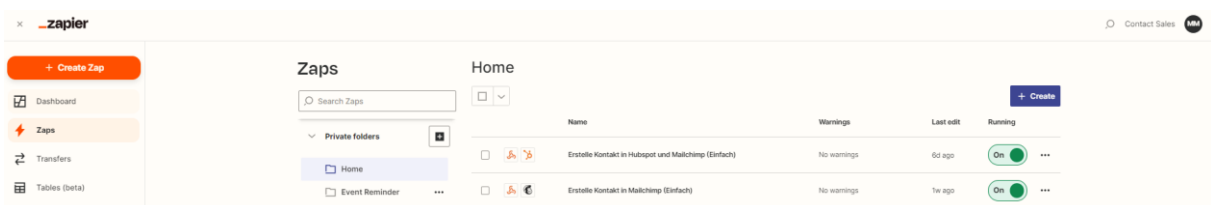
Falls du noch keinen Mailchimp-Account hast, kannst du [hier](#) einen erstellen und falls gewünscht deine bestehenden Kontaktdaten aus dem TICKETINO-Cockpit manuell übertragen. Das Vorgehen ist recht intuitiv. Für alle Fälle [verlinken wir dir hier dennoch eine Anleitung](#).

Es macht zudem Sinn, dass du neben deiner Marketingautomation über Mandrill auch gleich deine Kontaktdaten automatisiert bei Mailchimp aktualisierst. Wir empfehlen dir daher zuerst [den Zap in dieser Anleitung zu erstellen](#). Den Code für Mandrill kannst du dort dann einfach anhängen, bzw. der Mehraufwand ist im Vergleich zum Nutzen marginal. In der Annahme, dass du unserem Rat hier folgst, gestalten wir die weitere Anleitung aufbauend und erklären nicht erneut, wie du einen Webhook im TICKETINO-Cockpit erstellst etc.

## Schritt 2: Ergänze deinen Zap auf Zapier

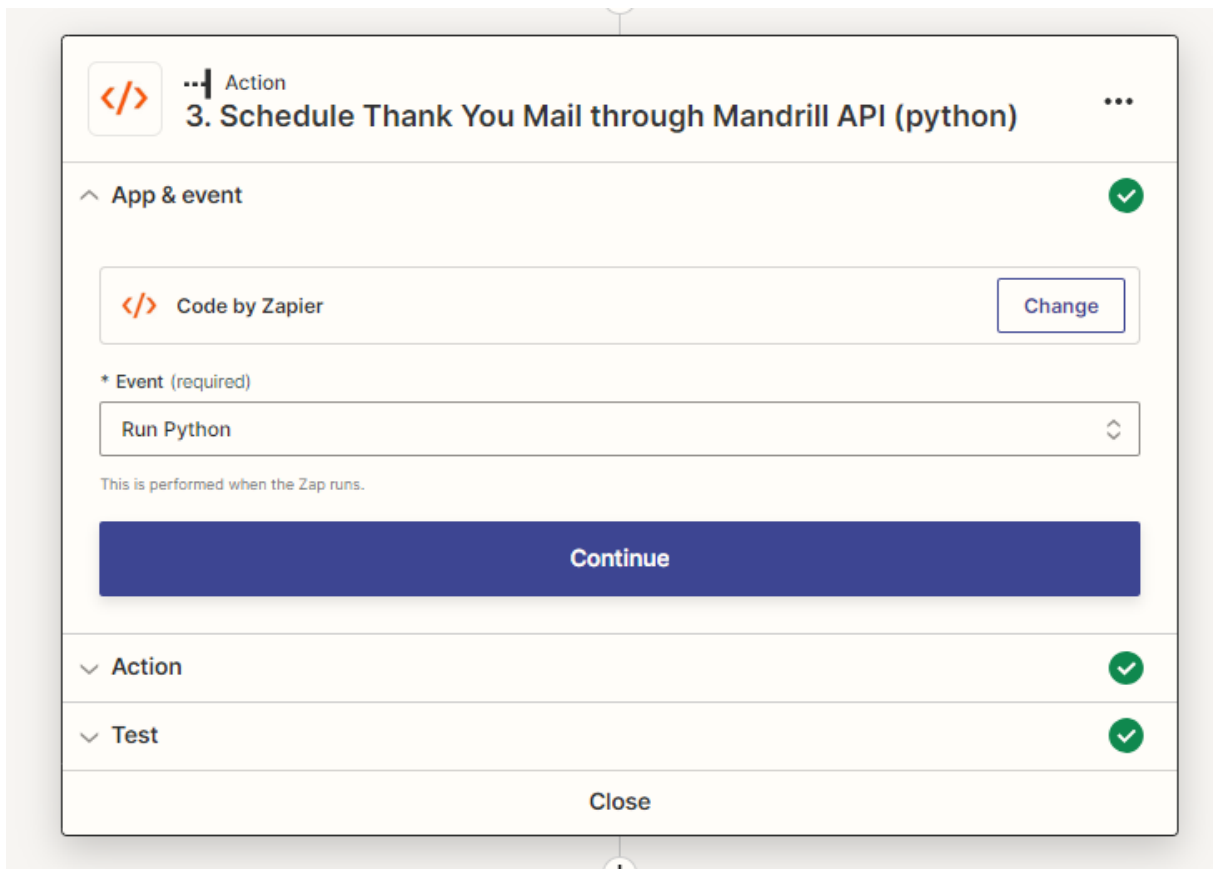
Sofern du unserem Rat gefolgt bist, hast du zuerst [die Schritte in dieser Anleitung](#) umgesetzt und kannst nun einfach deinen bestehenden Zap erweitern. Falls nicht, hole dies nun nach 😊.

Logge dich nun in deinen Zapier Account ein und klicke unter Zaps auf den zuvor erstellten:



Innerhalb des Zaps klickst du oben rechts auf «Edit draft» und kommst somit in den Bearbeitungsmodus.

Unten im Zap siehst du nun ein kleines Pluszeichen. Klicke darauf und es öffnet sich eine neue «Action». Wähle bei App & event die Option «Code by Zapier» und bei Event «Run Python».



Bestätige durch klicken auf «Continue».

Wähle dann die Felder, welche du an Mandrill übermitteln möchtest. Die Auswahl siehst du wiederum durch einen Klick in das Feld «Enter text or insert Data» und dann auf «Catch Hook in Webhooks by Zapier».

The screenshot shows the Zapier configuration interface for an advanced trigger. At the top, there is a section for 'App & event' with a dropdown arrow and a green checkmark. Below it is the 'Action' section, also with a dropdown arrow and a green checkmark. Under 'Action', there is an 'Input Data' section with a text input field containing 'Enter text or insert data...' and a 'x' button to the right. Below the input field is a '+' button. A note below the input field reads: 'Use the Input Data fields above to assign key names (left) and map values (right) from previous steps. Use notation... [more](#)'. Below this is a '\* Code (required)' section with a text area containing the JSON code: 

```
output = [{'id': 123, 'hello': 'world'}]
```

 Below the code area is a warning message: 'Warning! This is an advanced trigger! Please read the [Code documentation for more information](#).' At the bottom of the configuration area are two buttons: 'Refresh fields' and 'Continue'. At the very bottom of the interface, there is a 'Test' section with a dropdown arrow and a yellow warning triangle icon.

Da wir in den kommenden Schritten ein **Event-Reminder-Mail** auslösen wollen, übermitteln wir neben den Personendaten auch noch das Eventdatum (Siehe Feld «eventStart»). Je nach dem, was du machen möchtest, können aber auch andere Werte angebracht sein.

Benenne die «Input Data» Felder und matche die Werte wie in der Abbildung unten. Wenn du nicht so versiert bist, übernimm die Benennung und die Werte am besten 1 zu 1. Ansonsten musst du im Code unten noch weitere hier nicht erklärte Modifikationen vornehmen.

The screenshot shows the configuration interface for an action named "3. Eventreminder Mandrill". It features a section for "Input Data" with five rows, each containing a field name on the left and a value on the right. The values are mapped from a previous step. Below this is a code editor with a required code block. At the bottom, there are "Refresh fields" and "Continue" buttons, and a "Test" section with a warning icon.

Field Name	Value
eventStart	1. Tickets Ticket...ype Event Start: 2023-06-23T11:1...2.4308064+02:00
firstName	1. Organizer First Name: No data
lastName	1. Organizer Last Name: No data
email	1. Organizer Email: No data
fullName	1. Organizer First Name: No data 1. Organizer Last Name: No data

```
output = [{ 'id': 123, 'hello': 'world' }]
```

Warning! This is an advanced trigger! Please read the [Code documentation for more information.](#)

Bestätige anschliessend durch einen Klick auf Continue. **Pausiere hier am besten und erstelle zuerst deine Mail-Templates in Mailchimp** und übermittle dies an Mandrill (siehe «Schritt 4»). Der Grund liegt darin, dass du Kampagnenname etc. im Code hinterlegen musst.

Wir fahren hier in der Annahme fort, dass du entweder ein fortgeschrittener User bist oder «Schritt 4» eingeschoben hast.

In das Feld «Code» innerhalb deines Zaps kannst du anschliessend den Code-Textblock unten hineinkopieren. Diese musst du aber wie gesagt noch etwas anpassen. Den Code im Detail verstehen musst du nicht zwingend. Die nötigen Felder, die du anschauen musst, haben wir dir Gelb markiert, wobei der anzupassende Part jeweils in roter Schrift gekennzeichnet ist.

Vor dem Code-Textblock erklären wir dir noch kurz die einzelnen Felder. Die Anpassung kannst du entweder vor dem Einfügen bei Zapier oder auch danach vornehmen. Dem «Nicht-Profi», der schnell zum Ziel kommen möchte, empfehlen wir hier keine Experimente, bzw. auf saubere Arbeit zu achten. Schon ein falsches Zeichen kann dazu führen, dass gar nichts funktioniert 😊.

### Erklärung der Felder, die du anpassen solltest:

`date - timedelta(days=1)` = Hier definierst du, wann das Mail ausgelöst wird. In diesem Fall als minus (-) **1** Tag vor dem Eventdatum. Sollte die Frist zwischen Bestelldatum und Eventdatum kürzer als einen Tag sein, dann wird das Mail mit einer kleinen Verzögerung «quasi umgehend» ausgelöst. Durch eine Modifikation hier kannst du das Timing beeinflussen. Bspw. wäre für ein Dankesmail 2 Tag nach Event folgender Wert zielführend: `date + timedelta(days=2)`

`{{ "key": "md-AKmgnyHbpXXXXXXXXXX", "template_name": "Eventreminder1", "template_content": [ { "name": "template", "content": "Eventreminder1" } ] }` = Hier hinterlegst du den API Key deines Mandrill-Accounts und definierst, welches Template du versenden willst. Den API Key findest (bzw. erstellst) du in deinem Mandrill Account unter «Settings».

The screenshot shows the Mandrill account settings page. The top section is titled "SMTP & API Credentials" and includes instructions to send email using an API or SMTP. Below this, there are fields for Host (smtp.mandrillapp.com), Port (587), SMTP Username (any username will work - try "WOW MUSEUM Room for Illusions" for example), and SMTP Password (any valid API key). The bottom section is titled "API Keys" and features a "New API Key" button. Below the button is a table of "Valid API Keys" with columns for Key, Description, and Created. One key is listed: "md-aEiv" with a description of "Key" and a creation date of "Jun 9, 2023 12:52 pm". An "Edit" button is visible next to the key.

Den Namen des Templates findest du unter Outbound/Templates (bzw. hier kannst du eines erstellen, falls du es noch nicht von Mailchimp an Mandrill gesendet hast wie empfohlen(siehe «Schritt 4»).

Outbound

Activity Tags **Templates** Rules Split Tests URL Patterns Subaccounts

Dashboard

Reports

Outbound

Inbound

Settings

Account

You don't have any sending domains set up yet. To start sending through Mandrill, you need to [verify ownership](#) of your sending domain. [Set up your sending domain](#)

[+ Create a Template](#)

**Eventreminder1**  
 Published: Jun 9, 2023 12:49 pm  
 Draft last saved: Jun 9, 2023 12:49 pm

**ThankYou1**  
 Published: Jun 9, 2023 12:51 pm  
 Draft last saved: Jun 9, 2023 12:51 pm

`{"subject": "Mach dich bereit für deinen Event!"}` = Hier definierst du den Betreff deines Mailings.

`"from_email": "info@info.ticketino.com", "from_name": "Ticketino Marketing"` = Hier definierst du die Absender-Adresse sowie den Absender-Namen deiner Mailings.



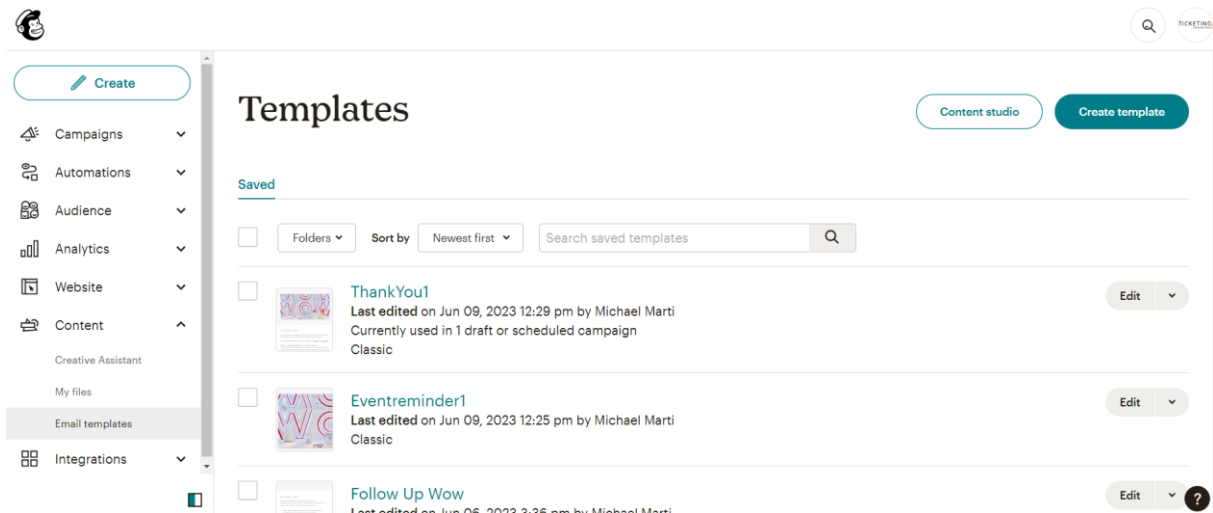
## Der Code-Textblock, welchen du anpassen und einfügen kannst

```
import http.client import json from datetime import datetime, timedelta #function to round the
time to nearest 15min def round_dt(dt, delta): return datetime.min + round((dt - datetime.min)
/ delta) * delta conn = http.client.HTTPSConnection("mandrillapp.com") #fetch data from
zapier eventStart = input_data['eventStart'] firstName = input_data['firstName'] lastName =
input_data['lastName'] email = input_data["email"] fullName = input_data["fullName"] start
= eventStart[0:18] #convert string to date date = datetime.strptime(start, '%Y-%m-
%dT%H:%M:%S') delta = timedelta(minutes=15) #calling the function to round the date date
= round_dt(date,delta) #reducing one day from the event start scheduleDate = date -
timedelta(days=1) #subtracting 2 hours to convert it to utc timezone scheduleDate =
scheduleDate - timedelta(hours=2) #formatting to ISO String 8601 scheduleDateFormatted =
scheduleDate.isoformat() sendAt = scheduleDate.strftime("%Y-%m-%d %H:%M:%S")
#configuring the mailchimp api request payload = json.dumps({ "key": "md-
AKmgnYHbpXXXXXXXXXXXX", "template_name": "Eventreminder1", "template_content":
[ { "name": "template", "content": "Eventreminder1" }, { "name": "firstname", "content":
firstName }, { "name": "lastname", "content": lastName } ], "message": { "subject": "Mach
dich bereit für deinen Event!", "merge_vars": [ { "rcpt": email, "vars": [ { "name": "FNAME",
"content": firstName }, { "name": "LNAME", "content": lastName } ] } ], "from_email":
"info@info.ticketino.com", "from_name": "Ticketino Marketing", "to": [ { "email": email,
"name": fullName } ] }, "send_at": sendAt }) headers = { 'Content-Type': 'application/json',
'Authorization': 'Bearer md-AKmgnYHbpE0oTnSWWAAHiw' } #here campaignId has to be
changed according to the event campaign conn.request("POST", "/api/1.0/messages/send-
template", payload, headers) res = conn.getresponse() data = res.read() print(data.decode("utf-
8")) output = [{"done": "true"}]
```

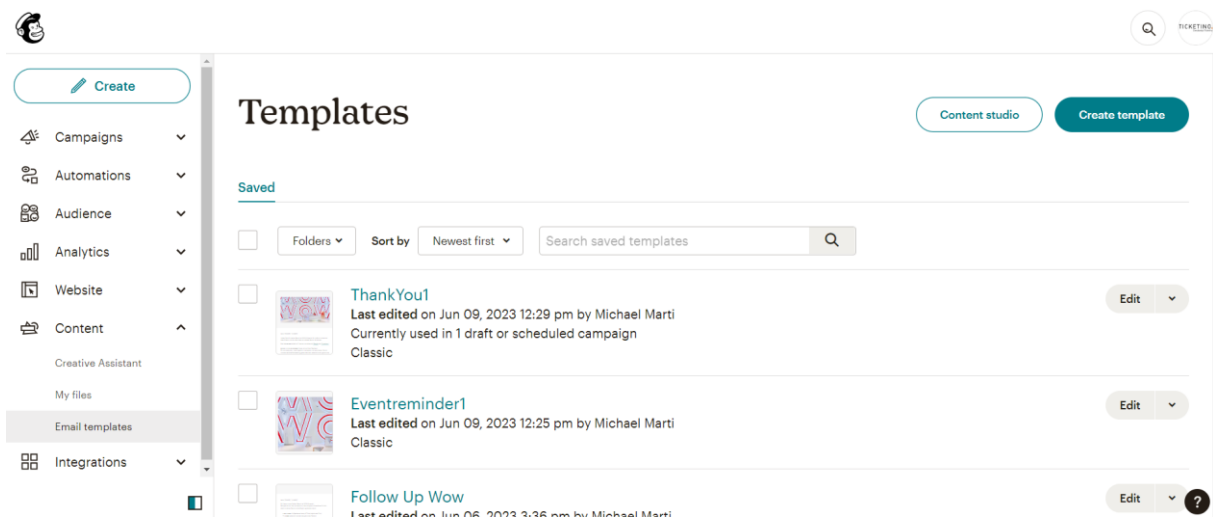
## Schritt 4: Erstelle dein Mail-Template

Je nach Fertigkeiten und Anforderungen an dein Mail kannst du hier natürlich auch etwas eigenes «Coden». Für den «Beginner» empfehlen wir die simple Mailchimp-Funktionalität zu nutzen.

Logge dich bei Mailchimp ein und klicke unter Content/Email templates auf «Create template».



Du kannst nun nach Lust und Laune dein Mail erstellen. Sobald du zufrieden bist, speicherst du ab. In der Übersicht kannst du das Template im Anschluss durch einen Klick auf «Send to Mandrill» an Mandrill übermitteln. Alternativ kannst du auch das HTML exportieren.



## Schritt 5: Teste die Automation

Gehe nun in den Ticketshop und löse eine Bestellung in deinem Event aus.

The screenshot shows a ticket purchase form with the following fields and values:

- Total inkl. MWST. CHF 1.00
- KÄUFERANGABEN
- Anrede: Herr
- Vorname: Michael
- Nachname: Marti
- Firma: TICKETINO AG
- Strasse: 470 Birmensdorferstrasse
- PLZ/Stadt: 8055 Zürich
- Land: Schweiz
- E-Mail: michael.marti@gmx.ch

A green callout box on the right contains the phone number 0900 441 441 (CHF 1.00/Min.)\* and a note: \* Publikation der Nummern nur gemäss Vorgaben.

A red button at the bottom right says "BESTELLUNG ABSCHLIESSEN".

Wenn du nun die deine Eventreminder-Mail zum gewünschten Zeitpunkt erhältst, dann scheint zumindest mal die Marketing-Automation zu funktionieren. Dies gilt aber nicht zwingend für die Synchronisation deiner Kontakte. Prüfe nun Layout, Betreff, Absender-Adresse, Antwort-Adresse und Personalisierung (falls eingebaut). **Unabhängig ob du das Mail erhalten hast, folgst du den weiteren Schritten dieser Anleitung.**

Gehe im Anschluss in deinen Mailchimp Account und prüfe die verknüpfte «Audience». Es müssten nun die von dir definierten Felder in der Mailchimp Datenbank ersichtlich sein. Wenn unter Source «Zapier» steht, ist dies ein sehr starkes Indiz, dass der erste Teil deines Zaps «Add/Update Subscriber in Mailchimp» läuft wie erwartet.

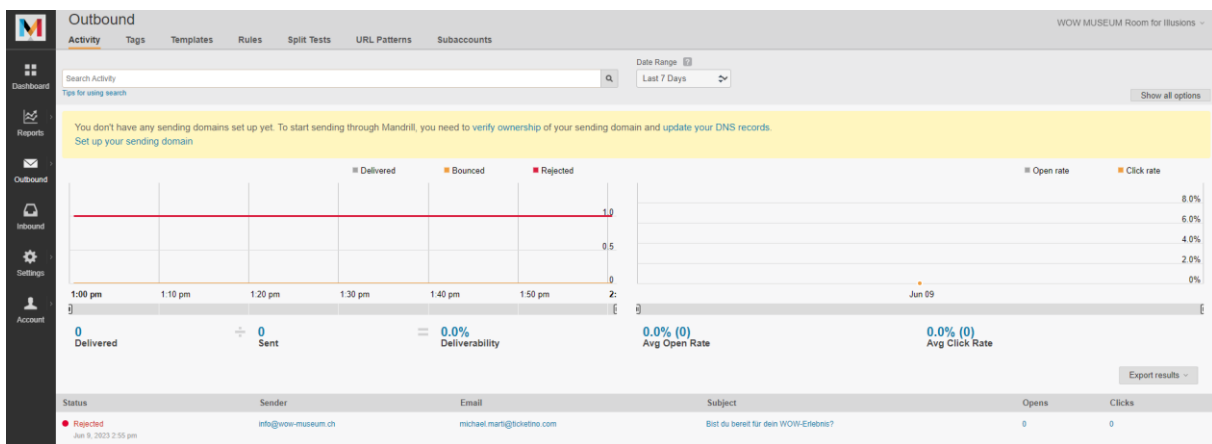
The screenshot shows the Mailchimp Audience interface for "WOW MUSEUM Room for illusions". The audience has 9,127 contacts, with 7,566 subscribers. The interface includes a sidebar with navigation options like Campaigns, Automations, and Audience. The main content area shows a table of contacts with the following columns: Filter by Tags, View Segment, New Segment, Email Address, First Name, Last Name, Address, Phone Number, Birthday, Tags, Email Marketing, and Source.

Filter by Tags	View Segment	New Segment	Email Address	First Name	Last Name	Address	Phone Number	Birthday	Tags	Email Marketing	Source
			michael.marti@ticketino.ch	Michael	Marti				Ticketino	Subscribed	Zapier

Gehe nun zu Mandrill (Direktlink oder Mailchimp/Automation/Transactional email und dann Klick auf «Launch App».

Klicke dort auf Outbound. Sofern dort deine Eventreminder-Mail aufgelistet ist, dann scheint auch der zweite Teil deines Zaps «Eventreminder Mail Zapier» wie erwartet zu funktionieren.

Solltest du dennoch kein Mail bekommen haben, wie in unserem abgebildeten Beispiel (Status «Rejected»), dann liegt dies in der Regel daran, dass du deine «sending domain» etc. noch nicht richtig aufgesetzt und verifiziert hast. Dies zeigt dir auch der gelbe Balken oberhalb des Graphs. Durch einen Klick auf die einzelnen verlinkten Satzteile innerhalb des gelben Balkens findest du weitere Anweisungen, Anleitungen und Codefragmente. Folge dort den Anweisungen von Mandrill/Mailchimp. Gegebenenfalls benötigst du hier etwas Hilfe von deinem Web-Admin (Person, welche deine Homepage erstellt hat).



## Schritt 6: Lehne dich zurück und lass Mandrill seine Arbeit machen

Deine Kontaktdatenbank auf Mailchimp wird nun bei jeder (bei dir) eingegangenen Bestellung über den [TICKETINO-Ticketshop](#) ergänzt und je nach Einstellung bei Zapier (Feld «Update Existing») auch aktualisiert, sobald eine erneute Bestellung mit derselben E-Mail-Adresse getätigt wird. Dies kann beispielsweise sinnvoll sein, wenn sich Werte wie «Nachname» bei einer Heirat ändern.

Du kannst dich nun zurücklehnen und zusehen, wie dein Mailingverteiler wächst. Parallel dazu versendet Mandrill nun bei jeder eingehenden Bestellung das Eventreminder-Mail.

Nun kannst du dich entweder zurück lehnen oder....

- Nach demselben Schema weitere automatisierte Mailings einrichten wie bspw. Feedback- & Dankesmailings.
- Über Mailchimp Newsletter gestalten und an deine wachsende Kundendatenbank senden.
- Weitere Zaps erstellen oder weiter Actions an den bestehenden anbauen etc.

Schon gewusst: an einen Zap kannst du verschiedenartige «Actions», also verschiedene Apps anhängen. Beispielsweise ein CRM, einen Kalender, ein Newsletter-Tool etc. Zudem kannst du [neben Webhooks für Bestelldaten auch Webhooks für Eventmutationsdaten generieren](#).

Weitere hoffentlich hilfreiche Whitepaper findest du in der [TICKETINO-Academy](#) oder im [TICKETINO-Veranstalterblog](#).

Benötigst du weitere Informationen zu Webhooks, API, Iframes sowie Codebeispiele? Dann besuche jetzt unsere [Website für Entwickler/innen](#).

Falls du Hilfe benötigst, bzw. die ganze Anbindung nicht selbst machen möchtest, dann gib diese Anleitung einem Web-Entwickler deines Vertrauens. Alternativ kannst du unter [marketing@ticketino.com](mailto:marketing@ticketino.com) eine Anfrage stellen und wir schauen, dass wir dir einen Programmierer gegen ein faires Stundensalär vermitteln können.

Das TICKETINO Team wünscht dir viel Erfolg mit deinem Onlinemarketing!