**TRAINING ROBxTASK**

Das Projekt **ROBxTASK** entwickelt eine Plattform, auf der mehrere Roboter oder andere Geräte zusammen gesteuert werden können. Dadurch ist es möglich, auf einfachem Weg eine Interaktion zwischen Geräten zu „programmieren“, ohne dazu besondere Kenntnisse zu benötigen. In dieser Testung soll diese Steuerung anhand kleiner Interaktionen bzw. Aufgaben ausprobiert werden.

**DIE ROBxTASK PLATTFORM**

Die **ROBxTASK Plattform** für die Studie **nach dem Training** ist unter folgendem Link erreichbar:

[**https://robxtask.salzburgresearch.at/**](%20https://robxtask.salzburgresearch.at/)

Die eigene Registrierung ist über Eingabe in ein Formular möglich, auch die UID Nummer der eigenen Organisation sollte eingegeben werden.

**GERÄTE KURZBESCHREIBUNG**

Name: **ARTI Chasi**

Geräteart: **Roboter**  
Skills:

* Fahren zu einer bestimmbaren Position/Örtlichkeit
* Transportieren
* Umgebungserkennung und Routenfinder
* Kommunizieren (Befehl erhalten, Befehl übermitteln)
* Demo: <https://www.youtube.com/watch?v=sohpgx6j_lk>

Name: **Camera System**

Geräteart: **sonstige Technologie**  
Skills:

* Bildaufnahme, Objekterkennung
* Bildübermittlung, Kalibrierung
* Kommunizieren (Befehl erhalten, Befehl übermitteln)

Name: **Fitbit Smartwatch**

Geräteart: **sonstige Technologie**  
Skills:

* Messung von Vitaldaten und Dehydrierung
* Aktivitätstracking, GPS Tracking
* Benachrichtigungen
* Kommunizieren (Befehl erhalten, Befehl übermitteln)

Name: **Franka Panda**

Geräteart: **Roboterarm**Skills:

* Dinge greifen, aufnehmen und platzieren
* Befestigung z.B. von Schrauben
* Gelenkmodule, lernfähig
* Kommunizieren (Befehl erhalten, Befehl übermitteln)
* Demo: <https://youtu.be/zHKHkRBbYMM>

Name: **MiR 100 Mobile Platform**

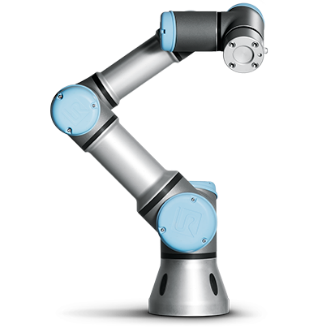
Geräteart: **Roboter**  
Skills:

* Fahren
* Transportieren
* Umgebungserkennung und Routenfinder
* Aufsatzmodule, u.a. Roboterarm kann befestigt werden
* Kommunizieren (Befehl erhalten, Befehl übermitteln)
* Demo: <https://www.youtube.com/watch?v=aMnhhpU51t0>

Name: **Q.bo One**

Geräteart: **sozialer Roboter**Skills:

* Sprachsteuerung, Sprachausgabe
* Mimik, Emotionserkennung und Reaktion
* Sich mittels Rollen zu einem anderen Platz bewegen
* Kommunizieren (Befehl erhalten, Befehl übermitteln)
* Demo: <https://www.youtube.com/watch?v=-AQy-LcculU>

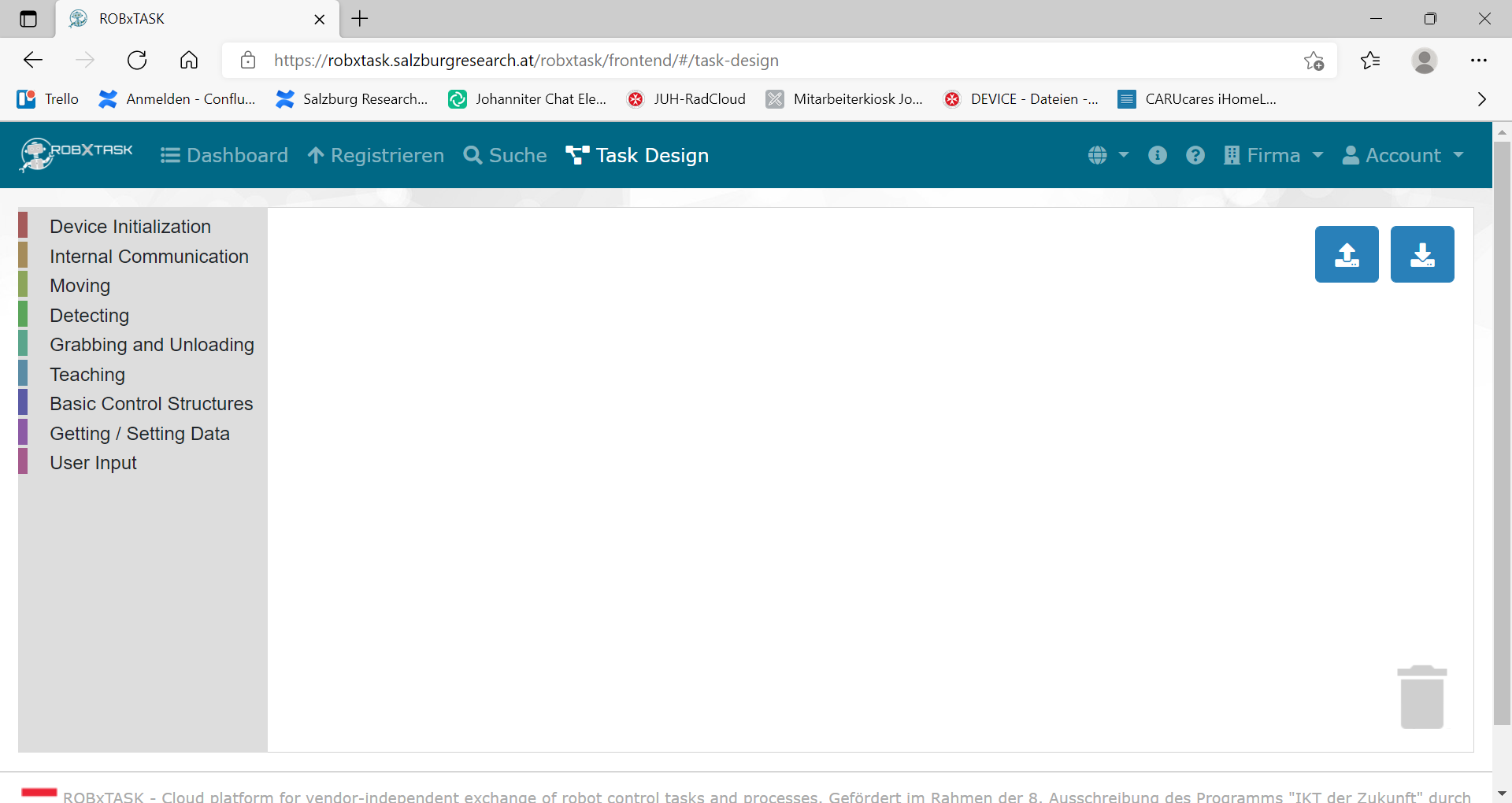
Name: **Universal Robot Manipulator UR10 CB3**

Geräteart: **Roboterarm**  
Skills:

* Dinge aufnehmen und platzieren
* Befestigung z.B. von Schrauben
* Kommunizieren (Befehl erhalten, Befehl übermitteln)
* Demo: <https://www.youtube.com/watch?v=PtncirKiBXQ>

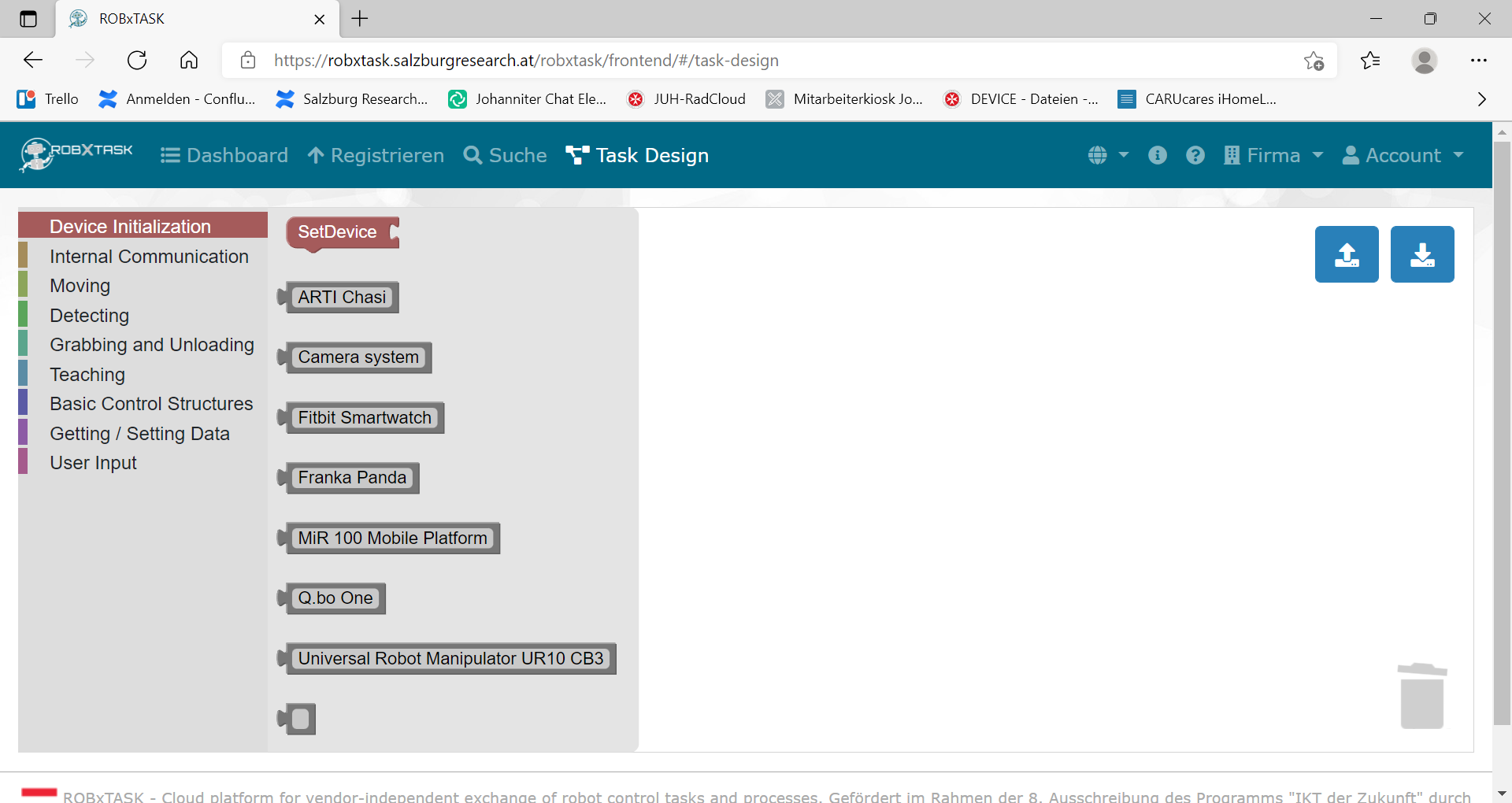
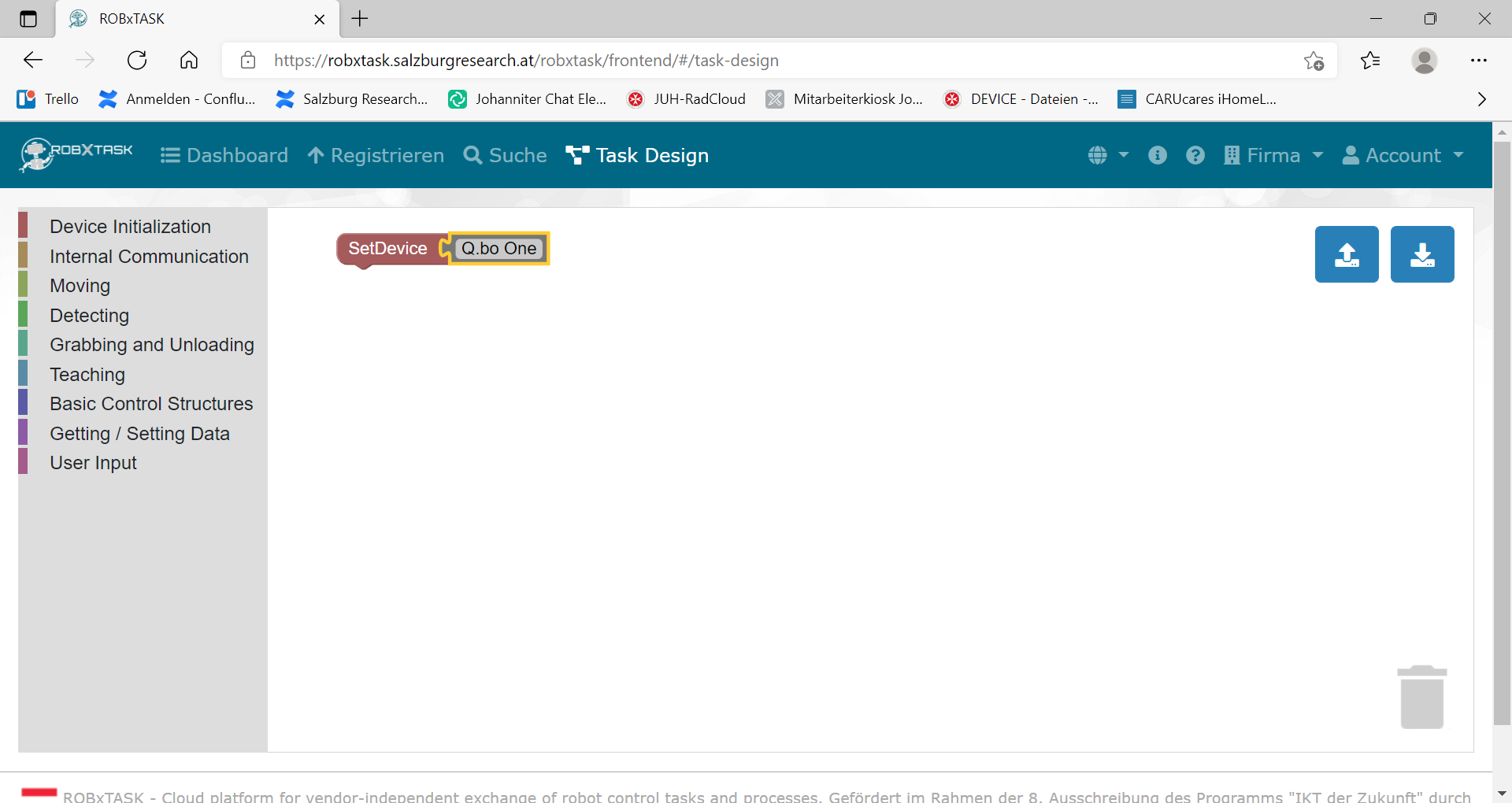
**TASK DESIGN EDITOR**

Auf der ROBxTASK Plattform steht ein TASK DESIGN EDITOR zur Verfügung.



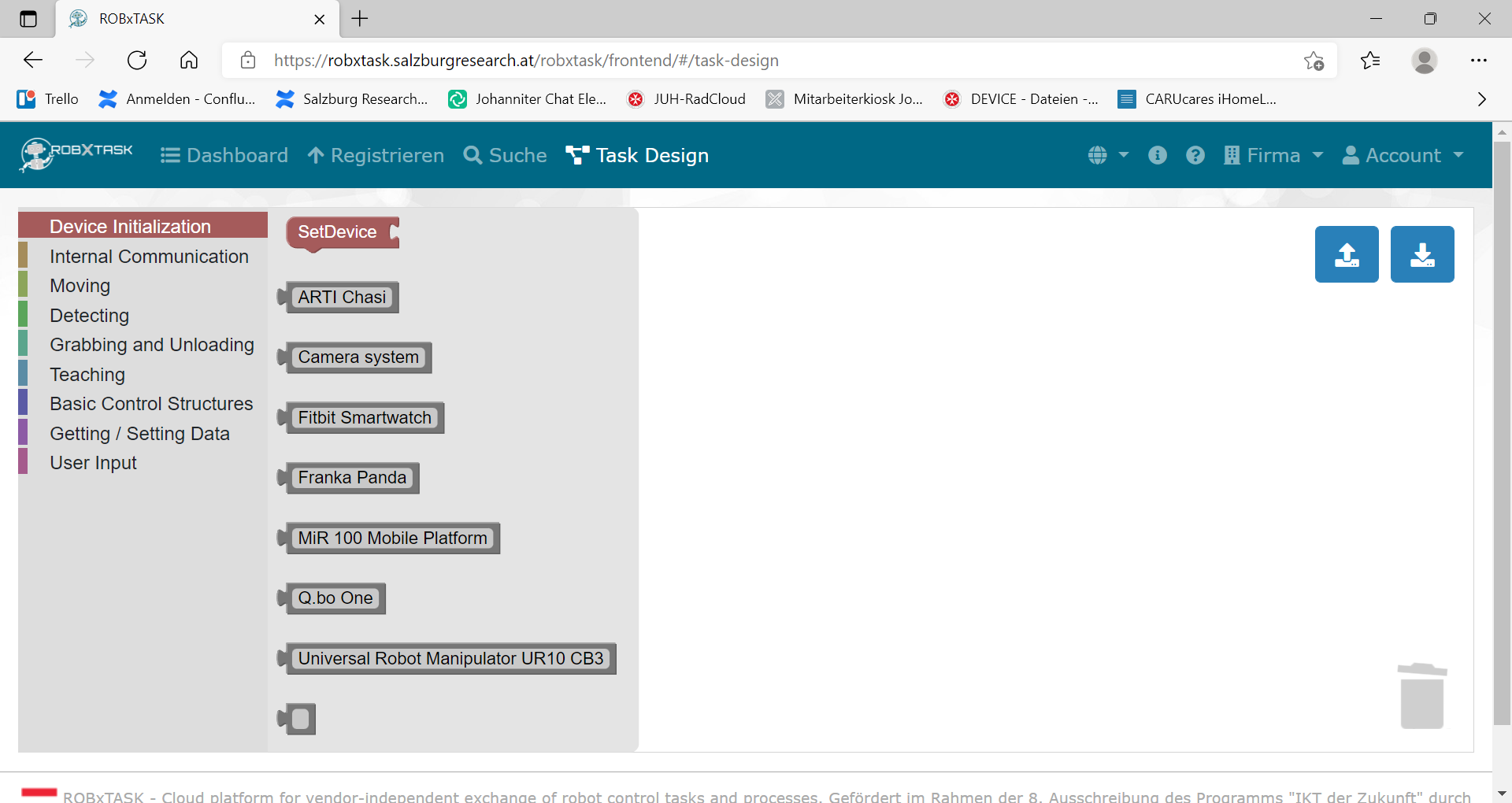
Über die Auswahl auf der rechten Seite können nun Aufgaben Robotern zugewiesen werden und unterschiedliche Geräte miteinander verbunden werden.

Die „Programmierung“ geschieht mittels Blöcken, die wie Puzzleteile ineinander greifen. Per **drag and drop** werden die Blöcke in das Fenster des Task Design Editors gezogen:

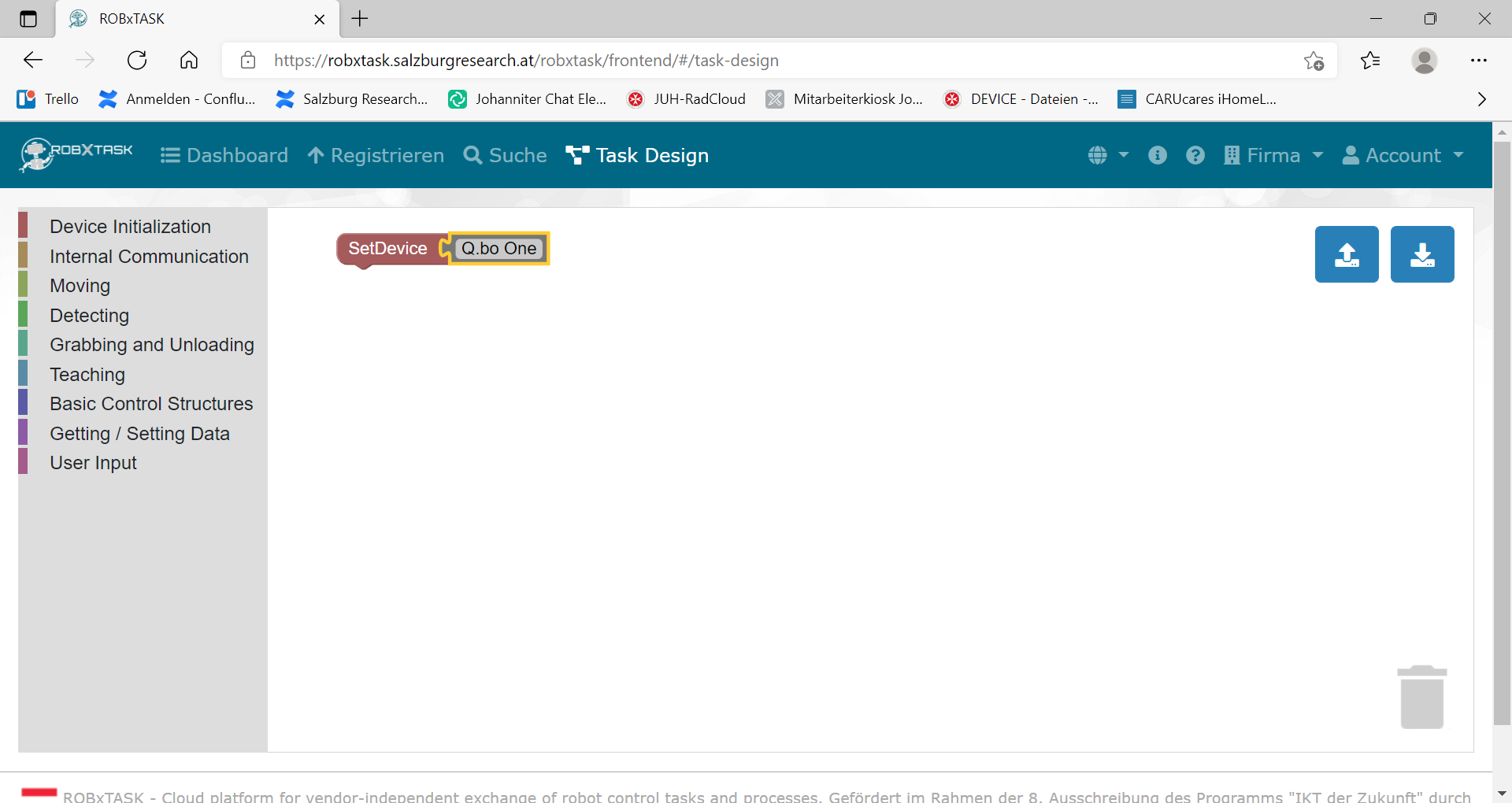
Die einzelnen **„Puzzleteile“** sollen dabei ineinander greifen.

Ein leeres Puzzleteil kann dazu verwendet werden, eigenen Text einzugeben, z.B. einen Text zur Kommunikation für die Übergabe einer Aufgabe zwischen 2 Geräten:

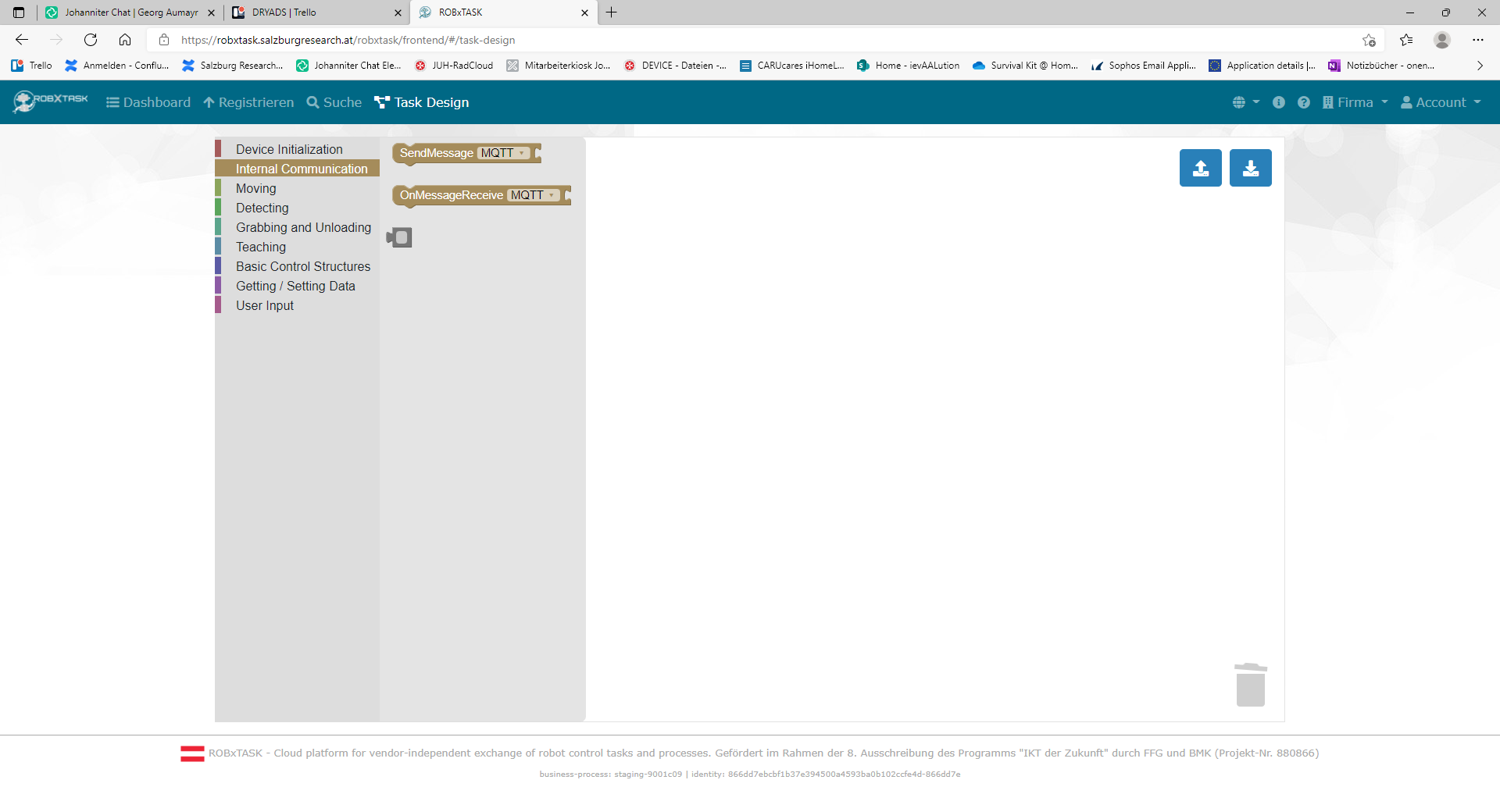
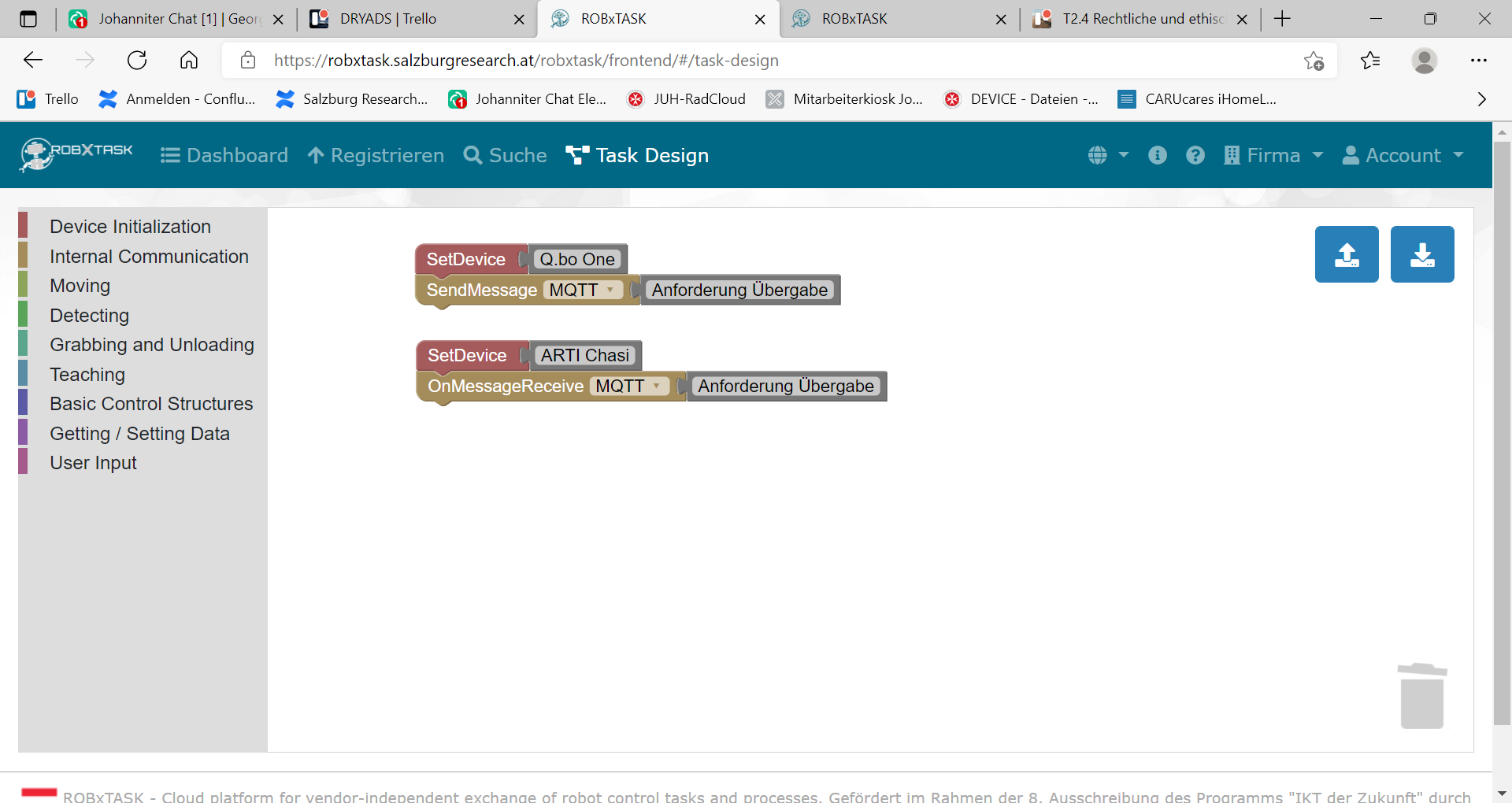


**Folgende AKTIONEN können im Task Design Editor gemacht werden:**

1. **Device Initialization:** Hier wird das jeweilige Gerät ausgewählt. Mit „SetDevice“ wird eine Auswahl befohlen, das jeweilige Gerät muss ausgewählt und hinzugefügt werden. Abbildung siehe oben: Hier wurde Q.bo One als Gerät durch SetDevice ausgewählt:

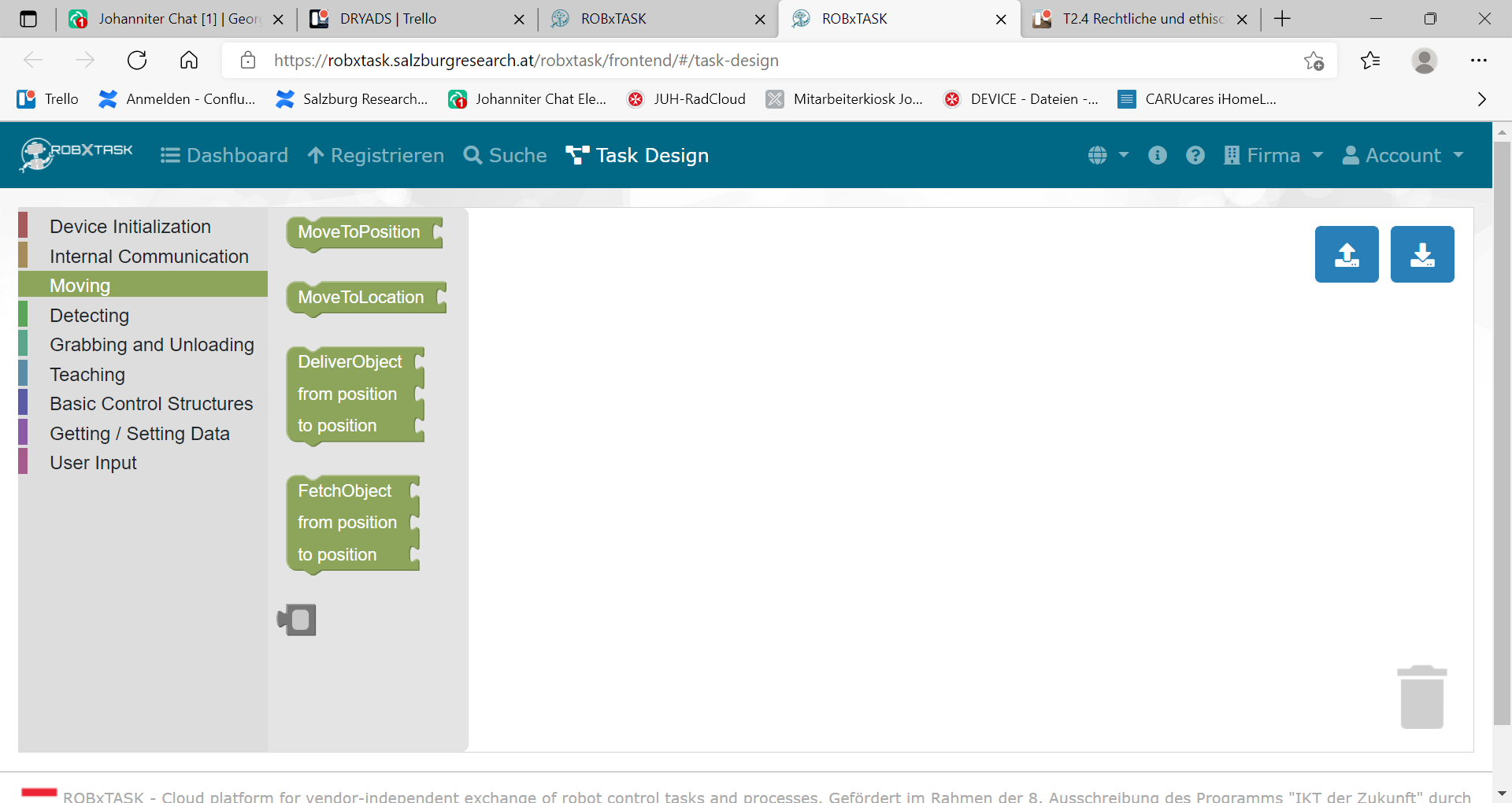
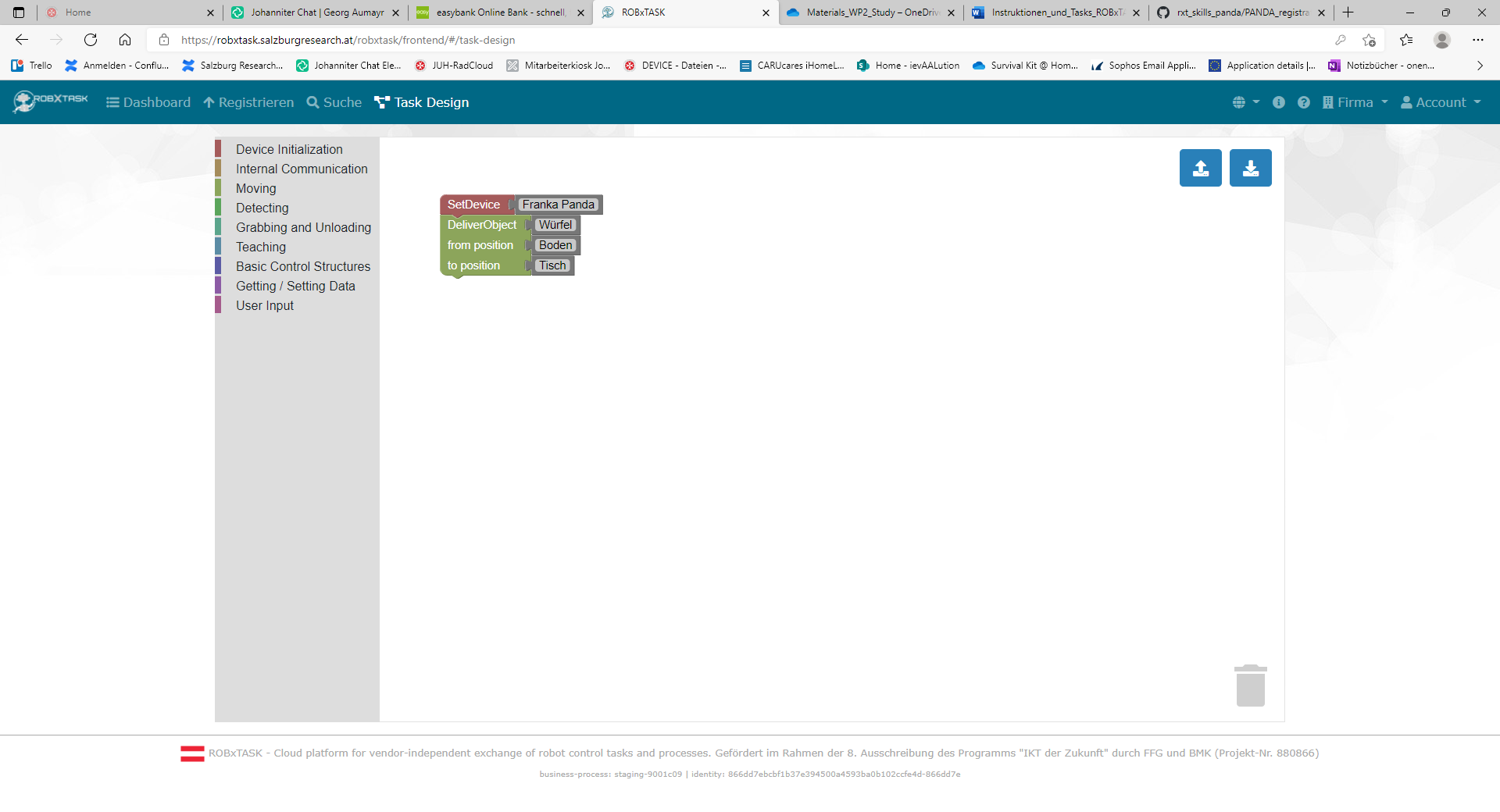


1. **Internal Communication:** Ein Gerät kann ein anderes Gerät mittels Textnachricht aktivieren. Dazu muss ein Gerät einen Text senden [SendMessage], das andere Gerät diesen Text verstehen [OnMessageReceive]. Es kann zwischen zwei Kommunikationsstandards ausgewählt werden: OPC-UA und MQTT. Die Nachricht, die gesendet wird, muss denselben Wortlaut und denselben Standard haben, wie die empfangene Nachricht. Im Beispiel wird von Q.bo One das Gerät Franka Panda über den Text „Anforderung Übergabe“ angesprochen.

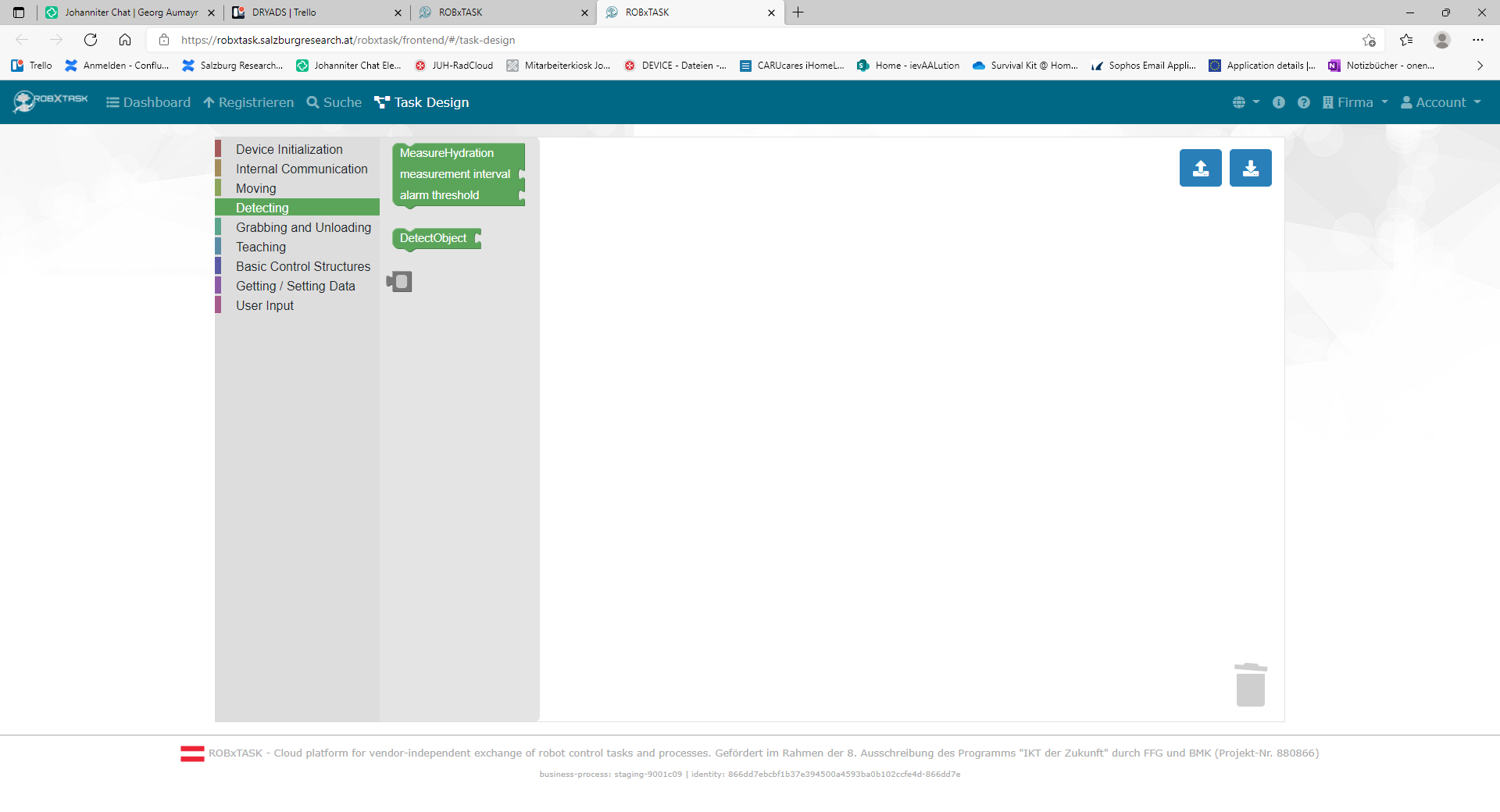
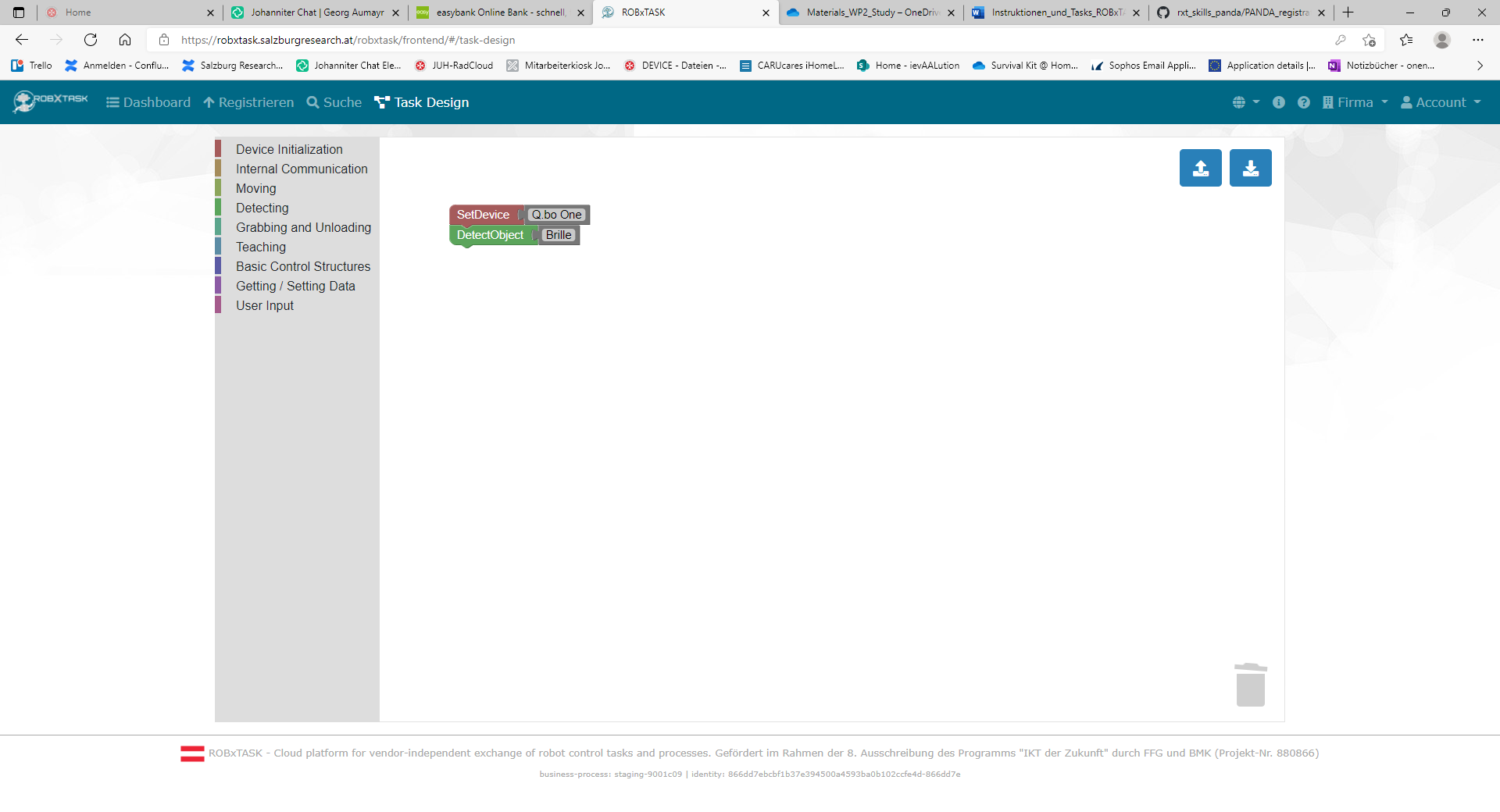


1. **Moving:**

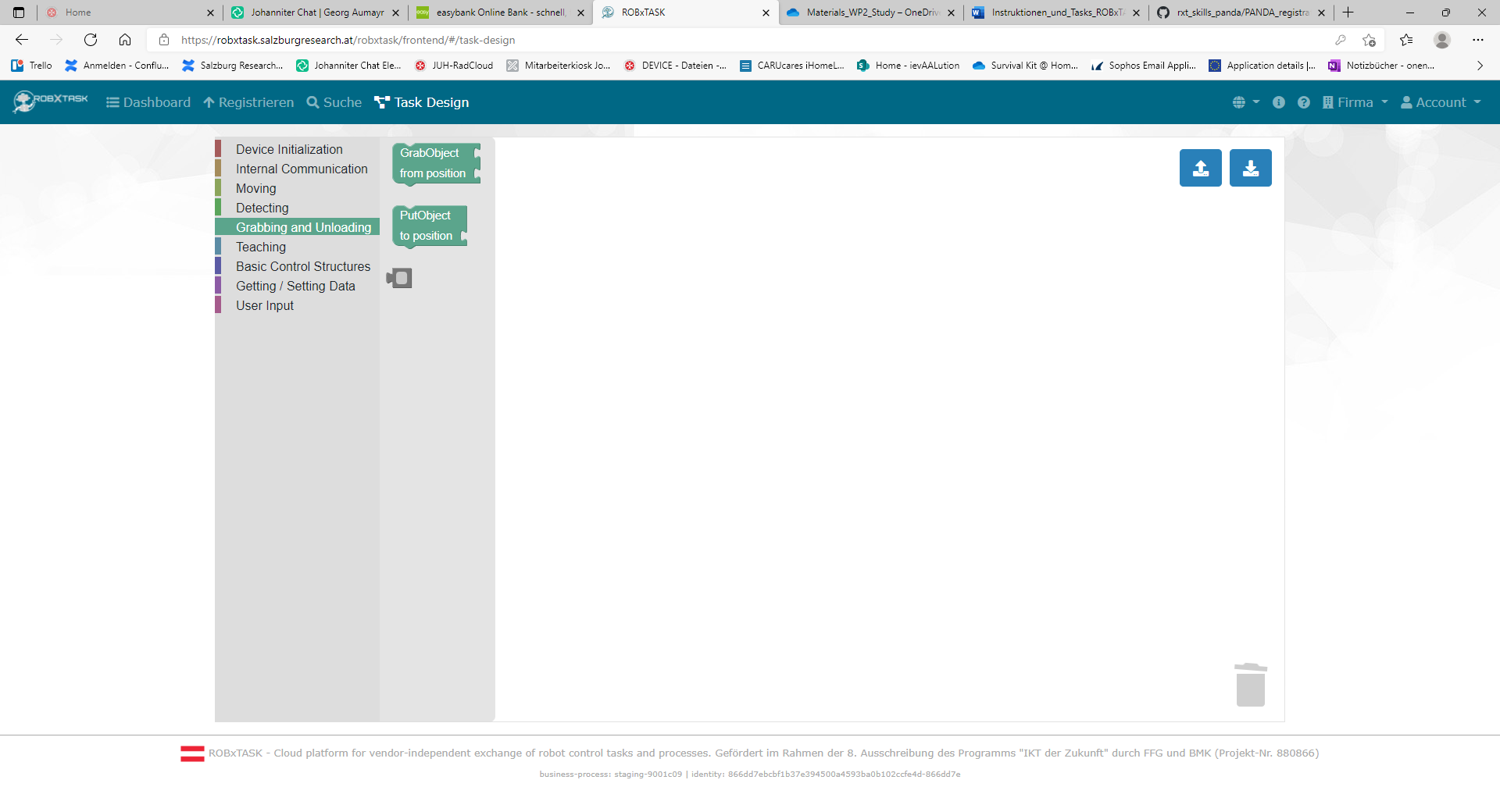
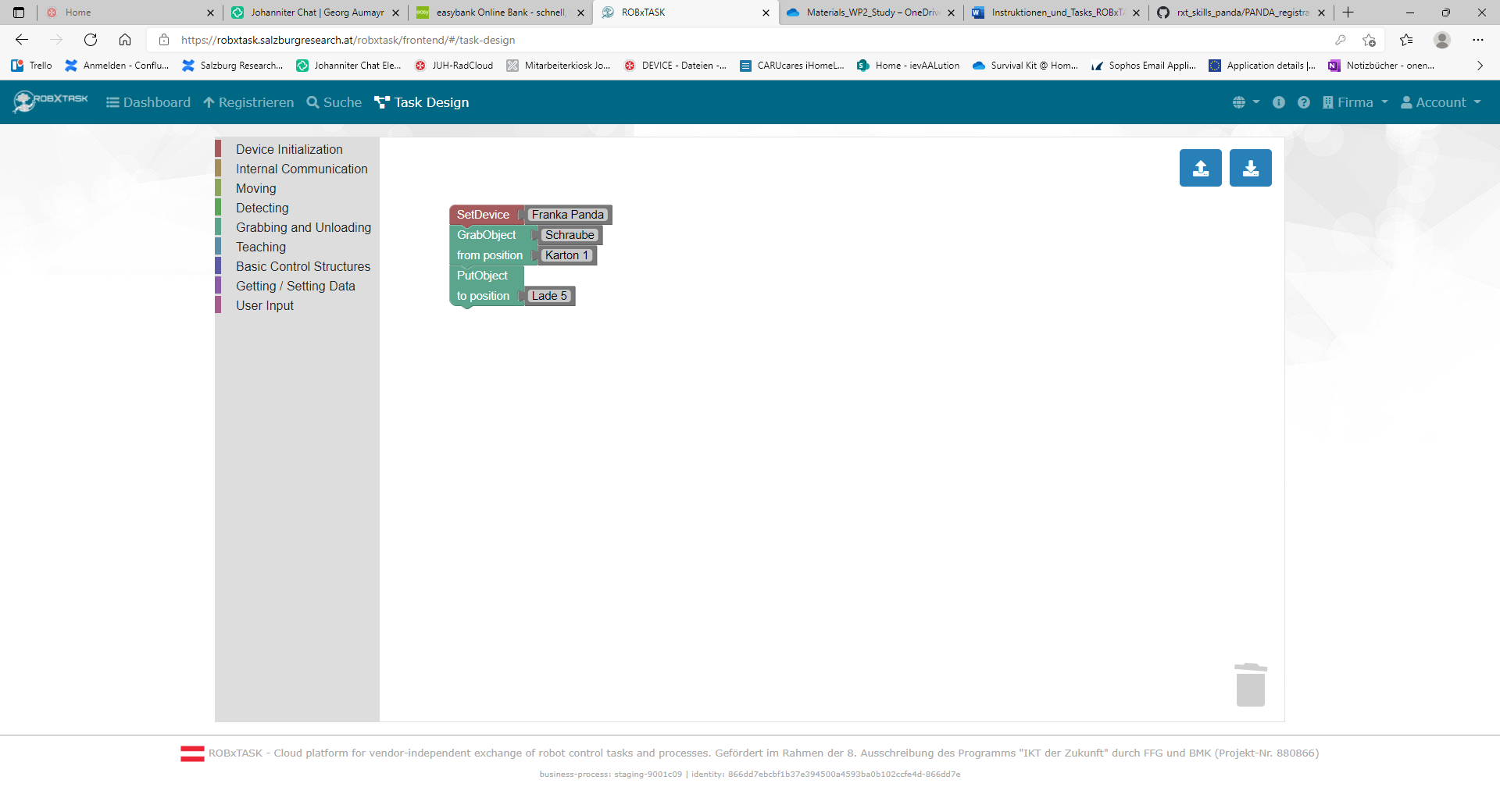
Einige der Geräte können sich selbst oder Gegenstände wo anders hinbewegen. Diese können mit „Moving“ gesteuert werden.



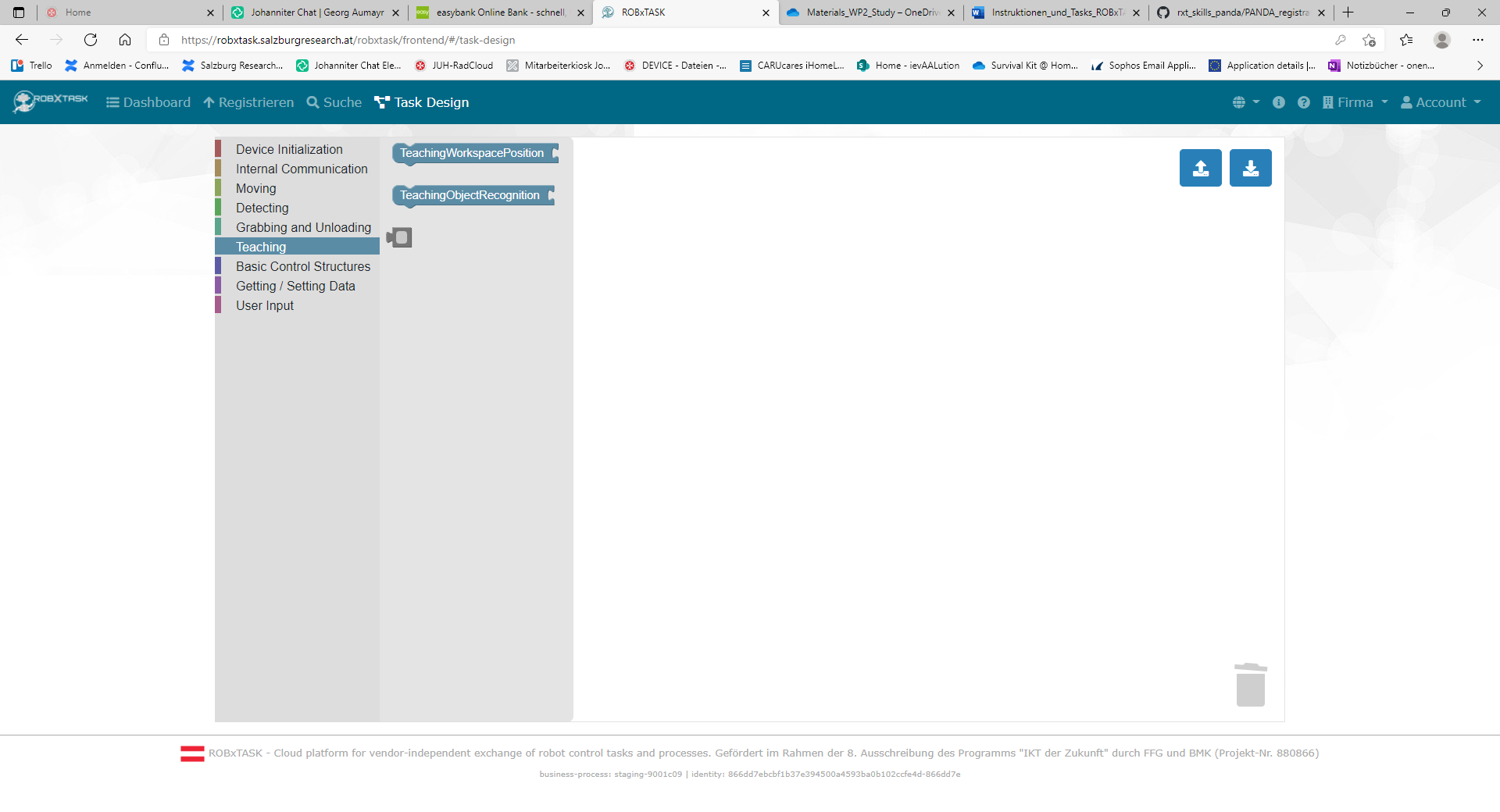
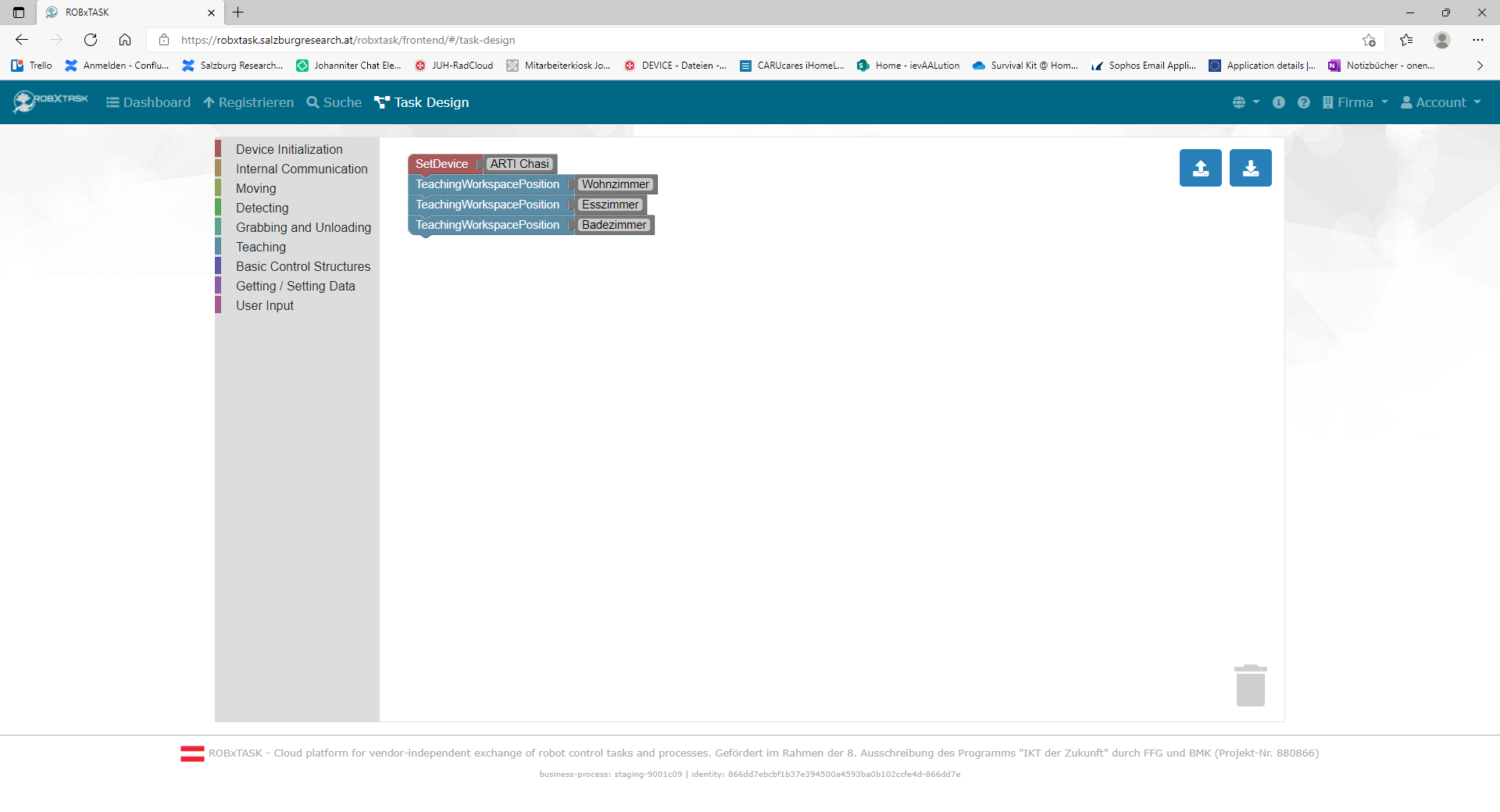
1. **Detecting:** Mit diesem Befehl können z.B. Gegenstände erkannt werden, die sich in ihrer Form unterscheiden oder auch Positionen. Es können auch Messerwerte erhoben werden, z.B. der Hydrationsstatus einer Person.



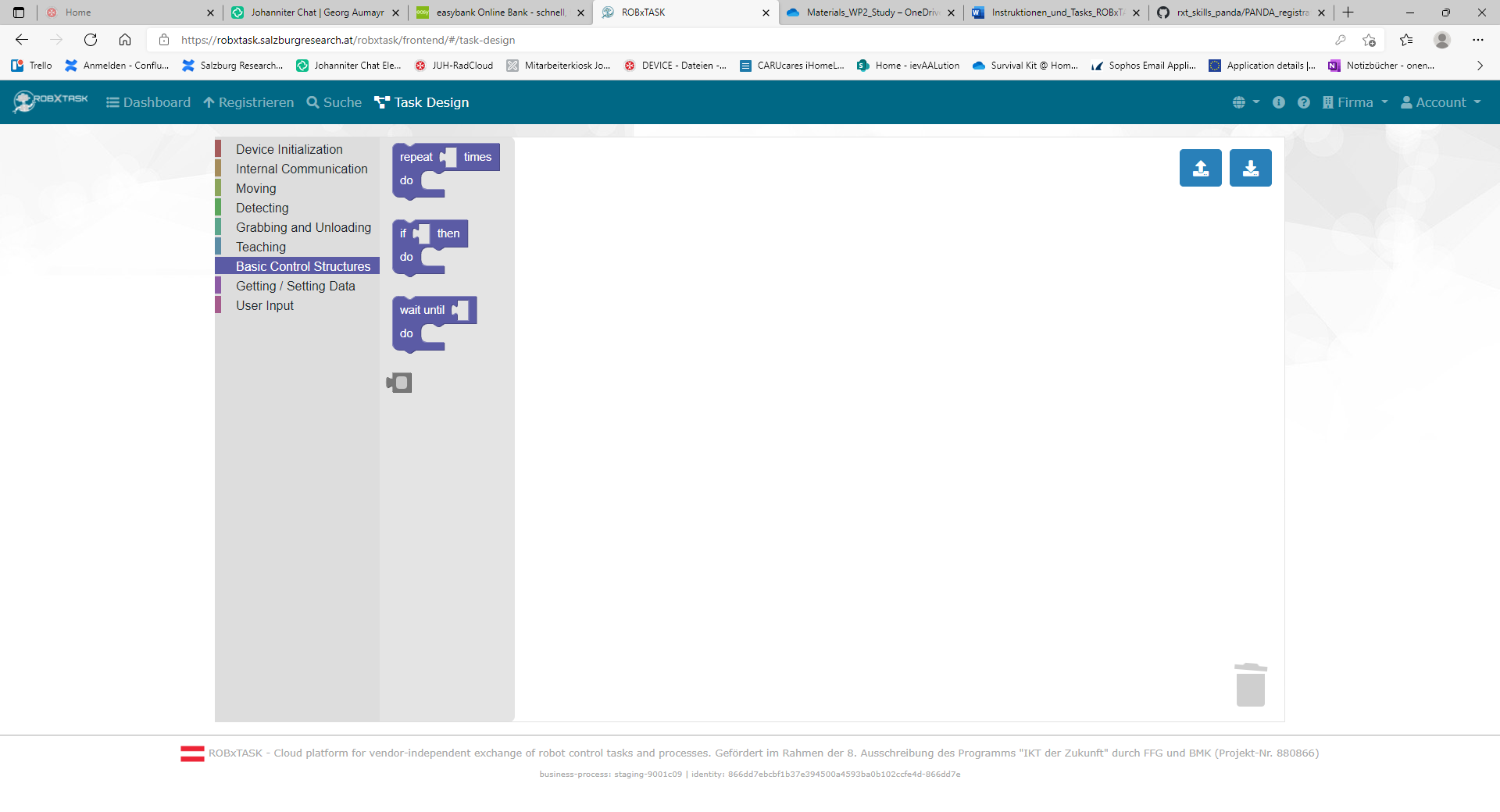
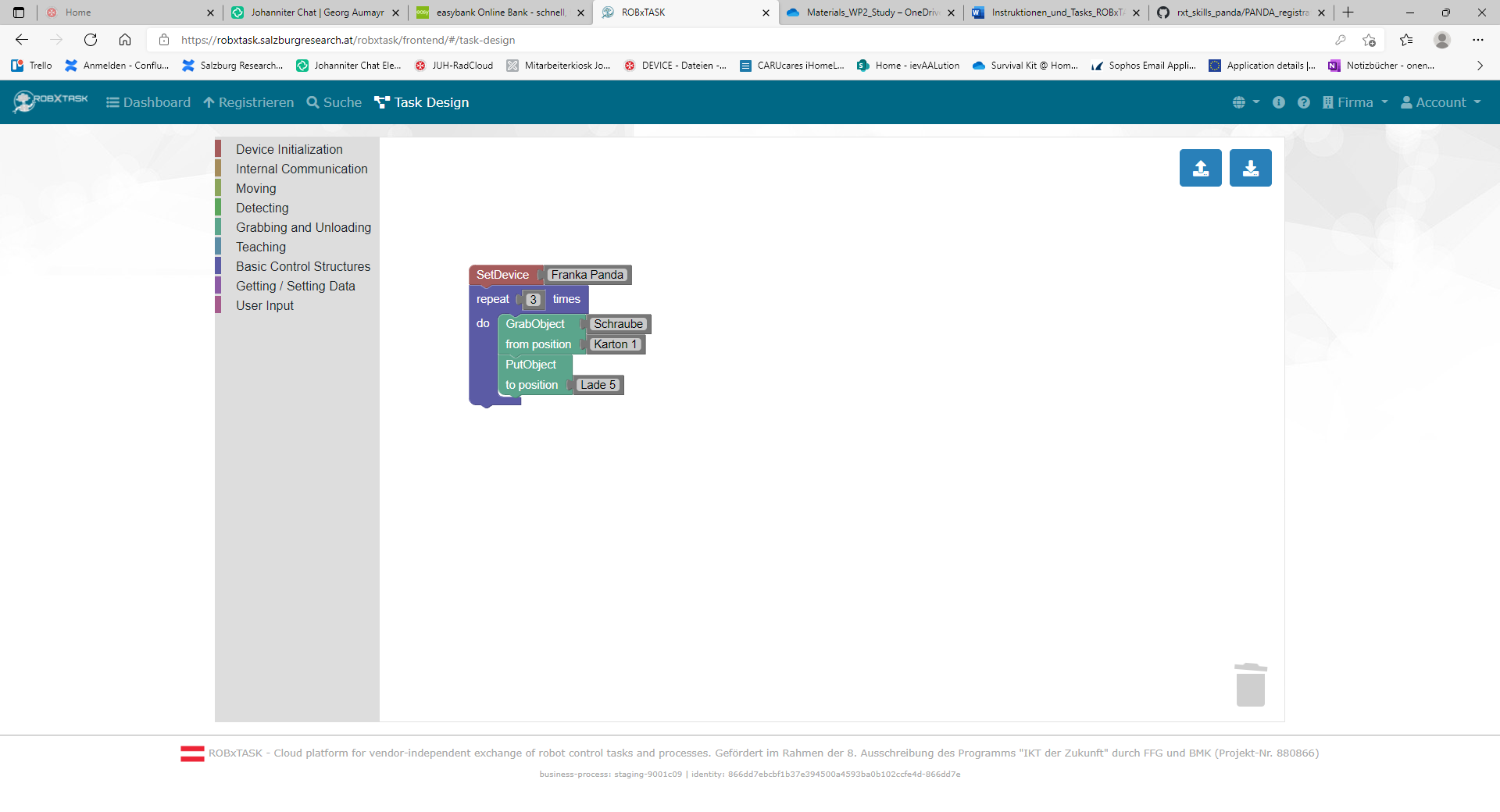
1. **Grabbing and Unloading:** Gegenstände können bewegt und an der bestimmten Position wieder abgeladen werden.



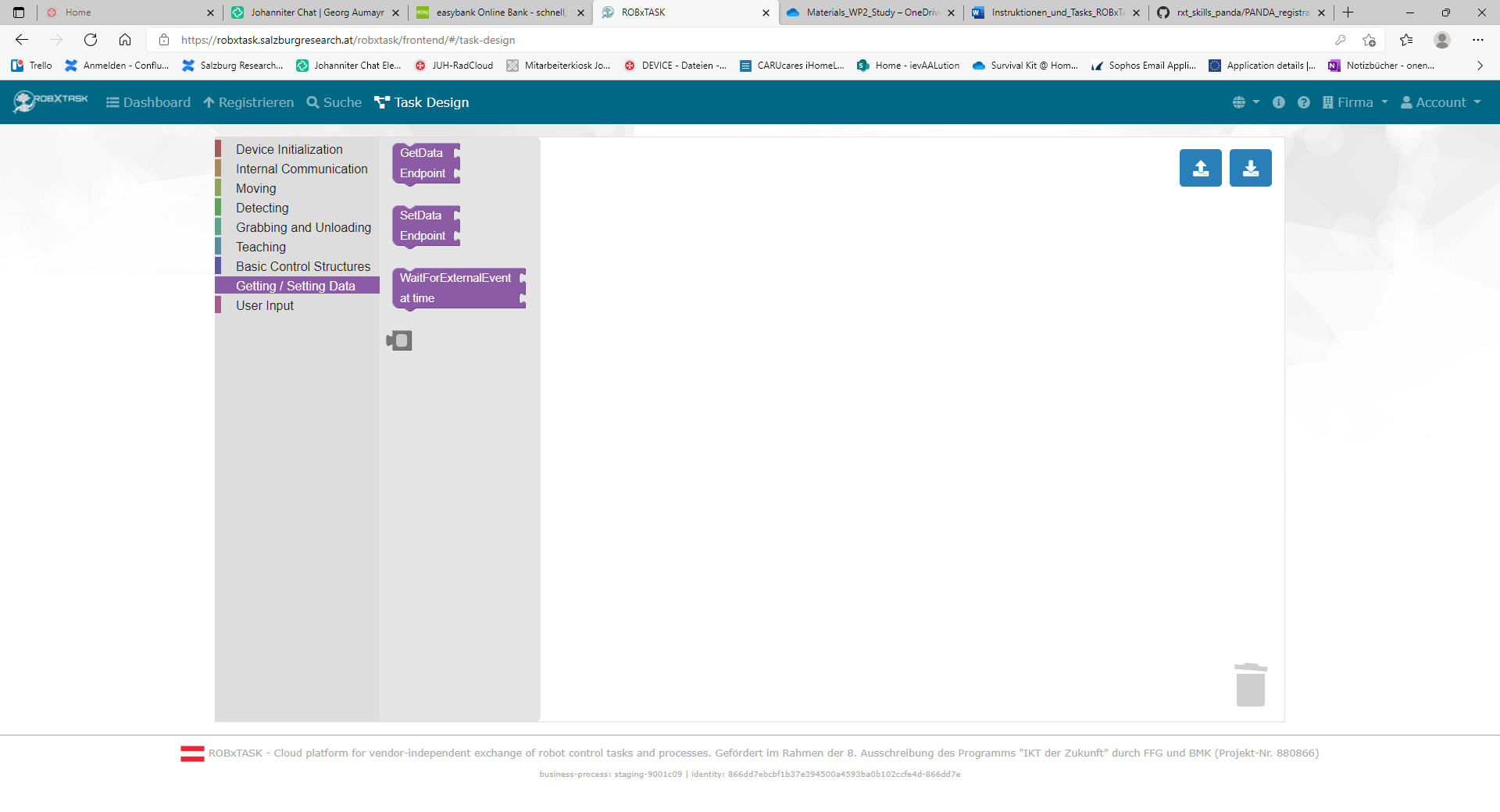
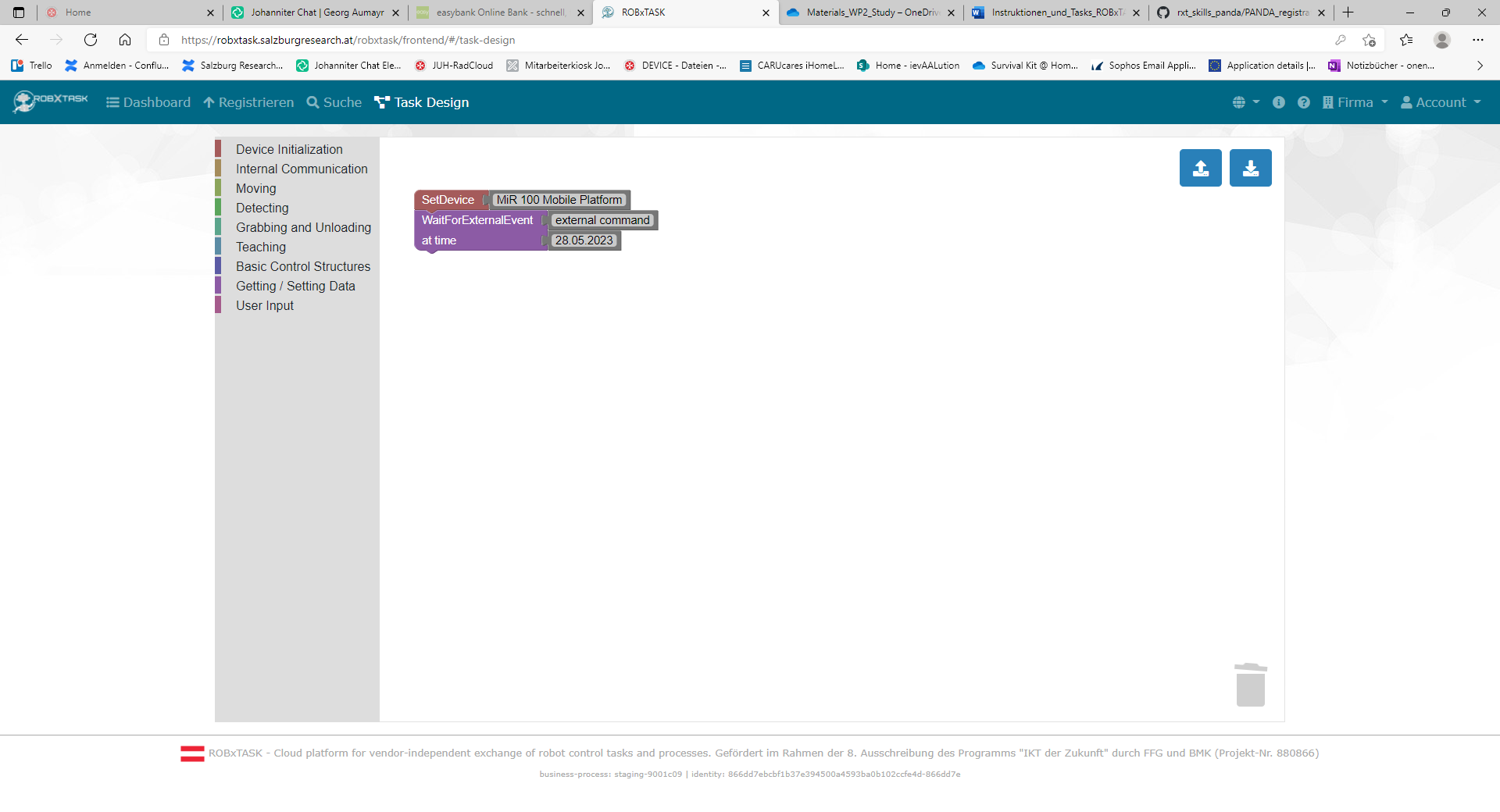
1. **Teaching:** Durch dieses Kommando werden für das Gerät relevante Positionen definiert. Diese Positionen können danach für weitere Befehlsketten verwendet werden:



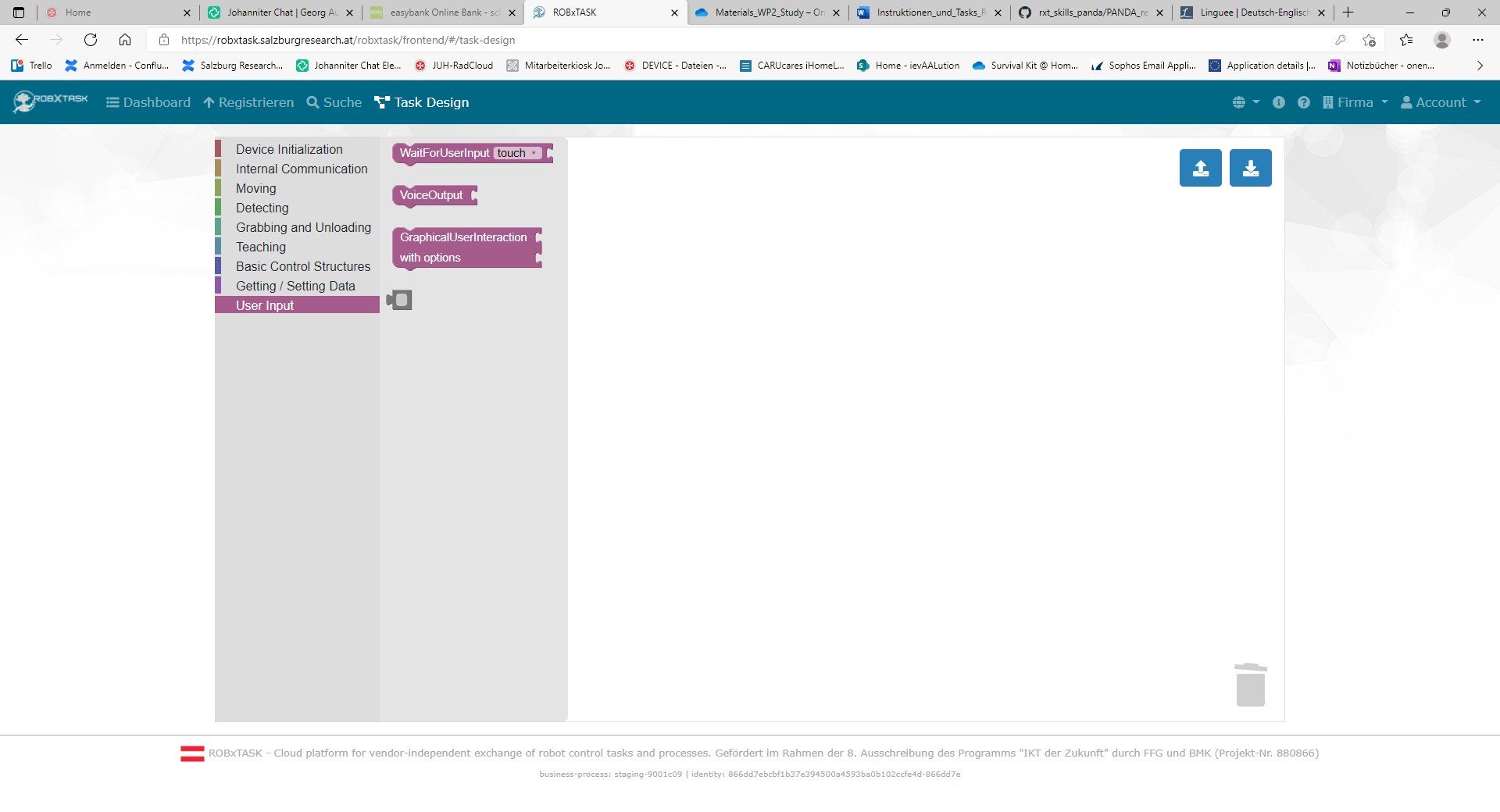
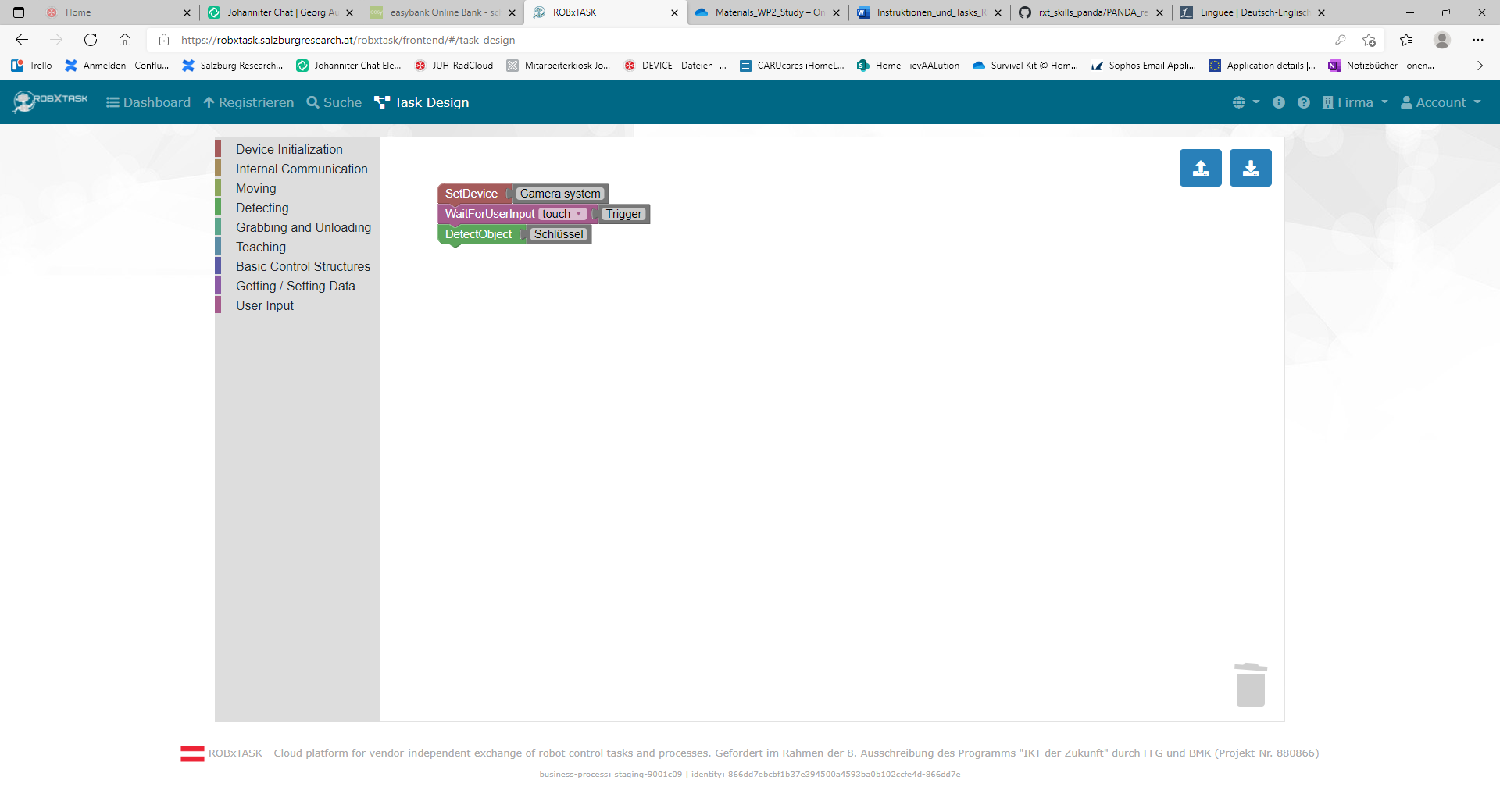
1. **Basic Control Structures:** Damit können z.B. Wiederholungen oder Bedingungen erstellt werden.



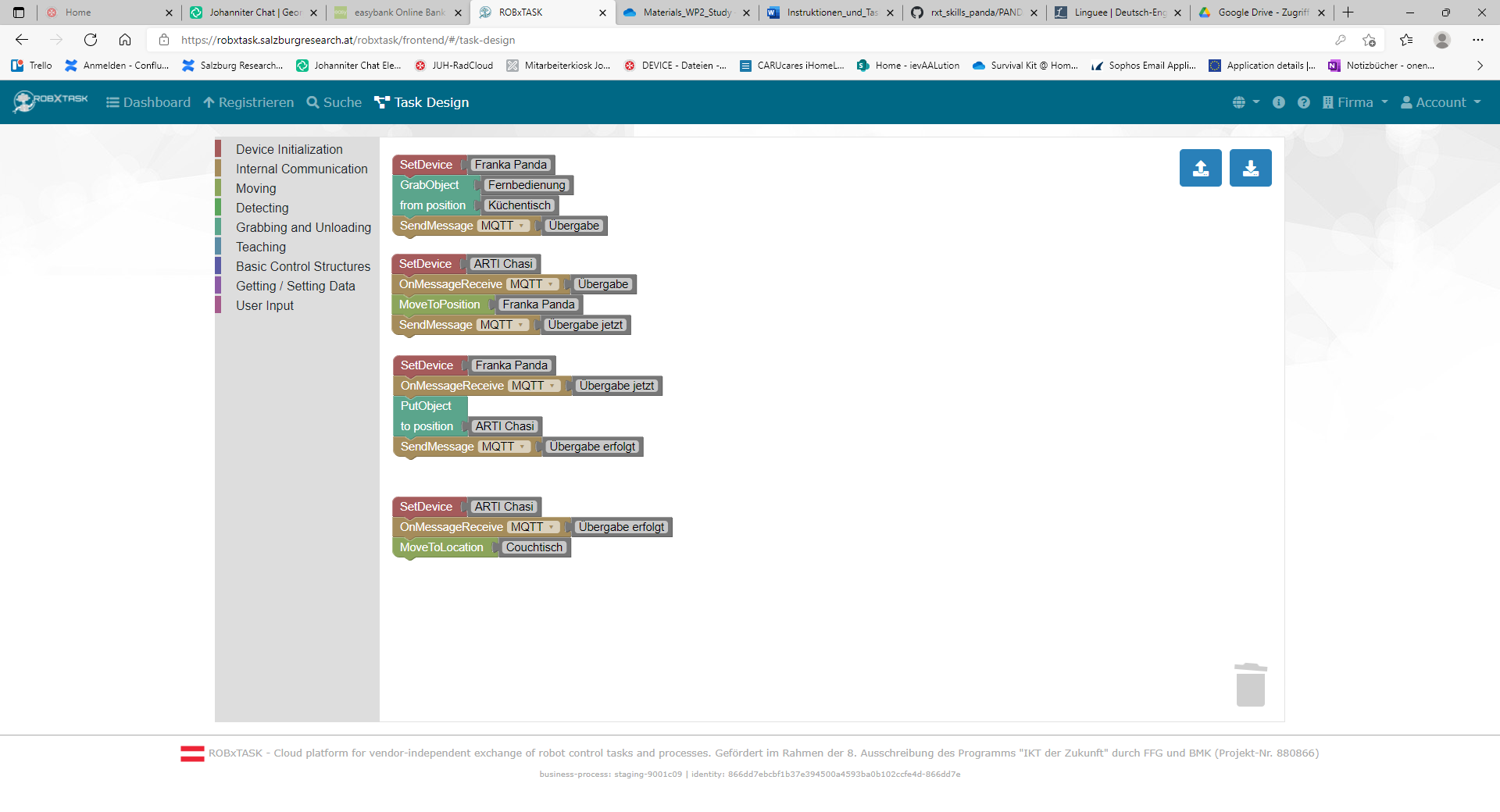
1. **Getting/Setting Data:** der aktuelle Status (z.B. betriebsbereit, standby) eines Gerätes kann übertragen werden, oder auch das Warten auf einen Trigger bestimmt werden.

1. **User Input:** Das Gerät reagiert auf eine bestimmte Aktion des Benutzenden, z.B. auf Sprache, auf eine Berührung oder die Eingabe auf einer Tastatur. Im Fall von Sprache kann auch eine Antwort durch manche Geräte erfolgen.

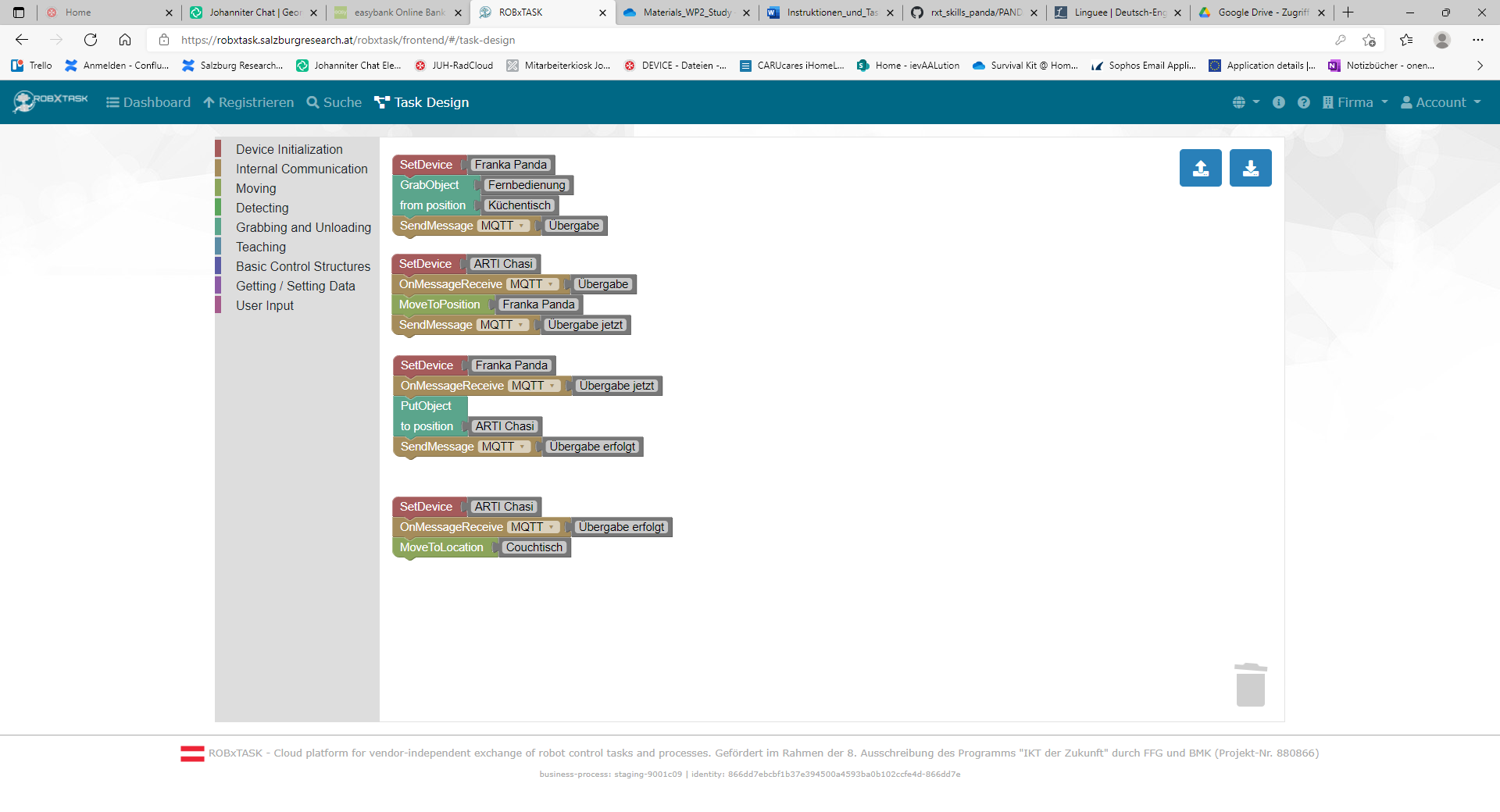


**Hier ist ein Beispiel für ein Szenario, das im Task Designer dargestellt wurde. Versuchen Sie, zu erkennen, was hier geschieht!**

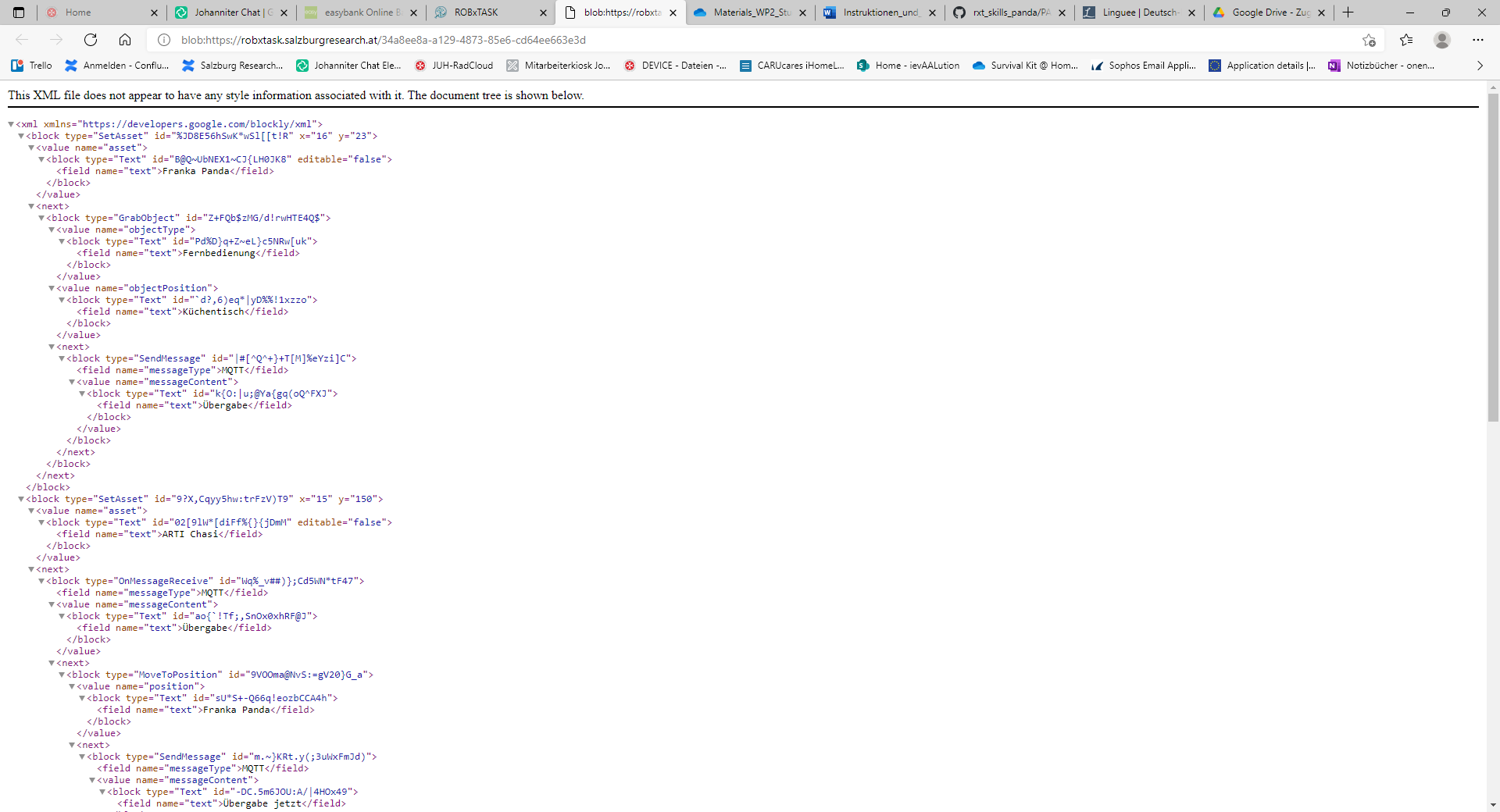


**XML FILE speichern**

Damit ein Projekt bzw. **die Ausführung einer erledigten Aufgabe gespeichert** und wieder aufgerufen werden kann ist es notwendig, ein XML File daraus zu generieren. Achtung, die Einrückung bzw. Formatierung muss beibehalten werden. Klicken Sie dazu auf den Pfeil nach unten:



Es öffnet sich ein neuer Tab mit dem XML File in ihrem Browser:



Benutzen Sie nun einen XML Formatter, z.B. [Free Online XML Formatter (liquid-technologies.com)](https://www.liquid-technologies.com/online-xml-formatter) oder [www.freeformatter.com](https://www.freeformatter.com/xml-formatter.html)

Klicken Sie ggf. auf „formatieren“. Das formatierte XML wird danach in einen Editor kopiert – z.B. txt editor – und danach abgespeichert und abgelegt. Nun kann bei erneutem Aufruf, der XML Code per Upload Pfeil wieder im Task Design Editor hochgeladen werden und das Projekt wird wieder angezeigt.