



# Selenium

**Christoph Schwarz – 83352**

**Florian Janot – 80785**

**Pauline Neumann – 82884**

**Philip Ritzer – 83127**

Hochschule Aalen  
Softwareengineering und Tools  
Prof. Dr. Marc Fernandes  
15-06-2022

# Agenda

**1. Beschreibung**

**2. Nutzen im Continuous Integration Prozess**

**3. Installation**

**4. Erste Schritte**

**5. Demo**

# 1. Beschreibung

- Testautomatisierungs Tools und Libraries für Webanwendungen
- Open Source und frei erhältlich
- Ursprünglich veröffentlicht im Jahr 2004 unter dem Namen “*JavaScriptTestRunner*”
- Unterstützt verschiedene Programmiersprachen wie Java, Python, C#, etc.
- Kann auf Windows, Mac OS und Linux implementiert werden



# 1.1 Selenium WebDriver

- Ermöglicht Tests von Websites aller Art
- Unterstützt eine weite Anzahl an Browsern
- Nutzt die Automatisierungs-APIs des jeweiligen Browsers
- Simuliert Eingaben seitens des Nutzers
- Braucht dafür keinen Source-Code der Website → testet das finale Produkt



## 1.2 Selenium IDE

- Möglichkeit, Aktionen eines Benutzers auf einer Website aufzuzeichnen
- Aufzeichnungen können beliebig abgespielt werden → einfacher Test von Webanwendungen
- “Stress-Tests” ebenfalls möglich
  - Es werden verschiedene Locator ausgewählt und beim Abspielen alle durchlaufen bis einer erfolgreich war
- Mit Plugins kann die IDE erweitert werden



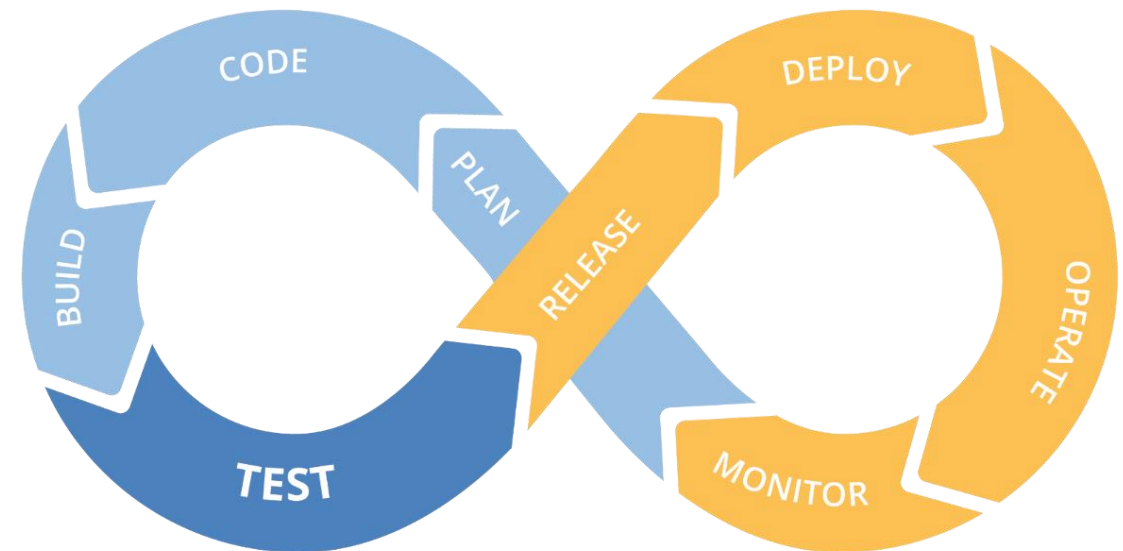
## 1.3 Selenium Grid

- Erlaubt es, Tests auf mehreren Servern unabhängig von der Plattform durchzuführen
- Test wird lokal gestartet, aber dann durch die Server ausgeführt
- Kann die Tests parallel ausführen → kürzere Durchlaufzeiten
- Ermöglicht es, verschiedene Kombinationen von OS und Browser zu testen



## 2. Nutzen im Continuous Integration Prozess

- Aktueller Code steht regelmäßig zur Verfügung
- Früherkennung von Bugs
- Kein großer Fortschrittsverlust bei Fehlern
- Projekt frisst weniger Zeit
- Qualität der Anwendung wird gesteigert



## 3.1 Installation Selenium IDE

- Verfügbar für Firefox und Chrome
- Bereitstellung als Browser Plugin
- Download Firefox: <https://addons.mozilla.org/en-GB/firefox/addon/selenium-ide/>
- Download Chrome: <https://chrome.google.com/webstore/detail/selenium-ide/mooikfkahbdckldjjndioackbalphokd>
- Hinzufügen zum Browser



## 3.2 Installation Selenium WebDriver

- Vor der Nutzung in Java müssen folgende Tools installiert werden:
  - Selenium Library
  - Webdriver

## 3.2.1 Installation Selenium Library

- Installation für Java via Gradle:

- build.gradle öffnen
- Folgende dependency hinzufügen:

```
dependencies {  
  
    compile group: 'org.seleniumhq.selenium', name: 'selenium-  
java',  
    version: '4.0.0'
```

- build.gradle speichern
  - Gradle Refresh
- Alternativ Installation via Maven ebenfalls möglich

## 3.2.2 Installation Browser Driver

- Wichtig: Es ist wichtig, immer die richtige bzw. passende Version des Webtreibers zur Browser Version zu installieren
- Für Google Chrome:
  - Download des Webtreibers unter: <https://chromedriver.storage.googleapis.com/index.html>
  - Zip Datei entpacken
  - Datei an gewünschten Ort verschieben, vorzugsweise ins Projektverzeichnis
- Für Mozilla Firefox:
  - Webtreiber für Firefox nennt sich “Geckodriver”
  - Download via GitHub unter: <https://github.com/mozilla/geckodriver/releases>
  - Schritte sonst identisch mit Google Chrome

## 3.2.3 Treiber einbinden

- Zum Einbinden der Treiber in Java gibt es insgesamt 3 Methoden: Driver Management Software, PATH Variable und Hardcoded Pfad. Folgend 2 Beispiele:
- Via PATH Variable:
  - CMD/Terminal öffnen
  - Anzeigen was bereits in PATH inkludiert ist: `echo %PATH%`
  - Folgenden Eintrag zu PATH hinzufügen: `setx PATH "%PATH%;C:\...\[Pfad zur Datei]"`
  - Prüfen auf Erfolg: `chromedriver.exe` in Terminal eingeben -> Webtreiber sollte gestartet werden
- Via Hardcoded Pfad:
  - Folgende Codezeilen in Java einfügen:

```
System.setProperty("webdriver.chrome.driver", "/path/to/chromedriver");  
  
ChromeDriver driver = new ChromeDriver();
```

## 3.2.4 Öffnen des Browsers als Funktionstest mit Java

- Für Google Chrome:

- Muster Source Code zum Öffnen und Schließen des Chrome Browsers im Testmode

```
ChromeOptions options = new ChromeOptions();  
  
driver = new ChromeDriver(options);  
  
driver.manage().timeouts().implicitlyWait(Duration.ofMillis(5000));  
  
driver.quit();
```

- Für Mozilla Firefox:

- Muster Source Code zum Öffnen und Schließen des Firefox Browsers im Testmode

```
FirefoxOptions options = new FirefoxOptions();  
  
driver = new FirefoxDriver(options);  
  
driver.manage().timeouts().implicitlyWait(Duration.ofMillis(5000));  
  
driver.quit();
```

## 4. Erste Schritte mit Selenium

- Was wollen wir nun gemeinsam Programmieren?
  - Automatisches Testen, ob die Google Suche nach “Hochschule Aalen” funktioniert
- Im Folgenden Step-by-Step Guide + vollständiger Code zum Nachmachen

## 4.1 Session starten

- Zuerst muss eine neue Session gestartet werden
- Source Code:

```
driver = new ChromeDriver();
```

## 4.2 Website aufrufen

- Dann eine Website aufgerufen werden; in unserem Falle: google.com
- Source Code:

```
driver.get("https://google.com");
```



## 4.3 Browserinformationen abrufen

- Um zu schauen, ob man auf der richtigen Website ist, ruft man Browserinformationen ab mit z.

B. `getTitle` oder `getCurrentUrl`

- SourceCode:

```
String title = driver.getTitle();
```

## 4.4 Wartestrategie

- Die Wartestrategie in Browserbasierten Tests ist essentiell. Generell ist es wichtig, gewisse Wartezeiten zu implementieren, wo diese erwartet werden.
- Sourcecode:

```
driver.manage().timeouts().implicitlyWait(Duration.ofMillis(500));
```

## 4.5 Element finden

- Um ein Element zu benutzen, muss das Element vorher ausgewählt werden
- Source Code:

```
WebElement searchBox = driver.findElement(By.name("q"));
```

```
WebElement searchButton = driver.findElement(By.name("btnK"));
```

## 4.6 Mit Element interagieren

- Nachdem die Elemente ausgewählt wurden, kann man mit ihnen interagieren
- Source Code:

```
searchBox.sendKeys("Hochschule Aalen");
```

```
searchButton.click();
```

## 4.7 Elementinformationen abfragen

- Es ist möglich, Elementinformationen abzufragen, wie z. B. `getAttribute`, `isSelected`, `isDisplayed`:
- Source Code:

```
String value = searchBox.getAttribute("value")
```

## 4.8 Session beenden

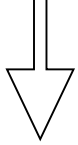
- Damit der Testlauf automatisch beendet wird, muss die Session geschlossen werden
- Source Code:

```
driver.quit();
```

## 4.9 Zusammenstellung SourceCode (inkl. Cookie) + Test

```
public class ErsteSchritteTest {  
    public WebDriver driver;  
    @Test  
    public void alleSchritte() {  
        System.setProperty("webdriver.chrome.driver", "...\\chromedriver.exe");  
        driver = new ChromeDriver();  
  
        driver.get("https://google.com");  
        String title = driver.getTitle();  
        Assertions.assertEquals("Google", title);  
  
        driver.manage().timeouts().implicitlyWait(Duration.ofMillis(500));  
        WebElement cookie = driver.findElement(By.id("L2AGLb"));  
        cookie.click();  
        driver.manage().timeouts().implicitlyWait(Duration.ofMillis(500));  
    }  
}
```

Pfad einfügen



## 4.9 Zusammenstellung SourceCode (inkl. Cookie) + Test

...

```
WebElement searchBox = driver.findElement(By.name("q"));
WebElement searchButton = driver.findElement(
By.xpath("/html/body/div[1]/div[3]/form/div[1]/div[1]/div[3]/center/input[1]"));

searchBox.sendKeys("Hochschule Aalen");
searchButton.click();

searchBox = driver.findElement(By.name("q"));
String value = searchBox.getAttribute("value");
Assertions.assertEquals("Hochschule Aalen", value);
}
}
```



## 5. Live Demo eines Testprozesses

- Getestet werden soll die anonyme Uploadfunktion der Plattform [imgur.com](https://imgur.com)
- Die Website besitzt in Upload Limit von 20 MB pro JPEG.
- Aus einem Ordner, in welchem Bilder in verschiedenen Größen liegen, sollen die Bilder seriell hochgeladen werden. Wenn ein Bild zu groß ist, soll mit dem nächsten Bild fortgefahren werden
- Dies ist ein perfektes Beispiel für eine Testanwendung sowie eine Automatisierungsanwendung


## 5.1 Driver setup


```
System.setProperty("webdriver.chrome.driver","...\\chromedriver.exe");
```


```
ChromeDriver driver = new ChromeDriver();
```


```
driver.get("https://imgur.com/");
```

```
driver.manage().timeouts().implicitlyWait(Duration.ofMillis(500));
```






Images, #tags, @users oh my!
 




Go Ad-Free
 Sign In
 Sign up

## "Hello, is this the Krusty Krab?" "No, this is Imgur"


EXPLORE TAGS




**Fail**  
 FEATURED • 24,032 Posts




**Archery**  
 770 Posts




**Dog**  
 212,569 Posts




**Otters**  
 534 Posts




**Science And Tech**  
 45,162 Posts




**Aww**  
 765,309 Posts




**Memes**  
 575,774 Posts




**Gaming**  
 299,080 Posts




**Funny**  
 2,700,803 Posts




**Awesome**  
 753,805 Posts



**Staff Picks**  
 6,564 Posts




**Oc**  
 40,481 Posts




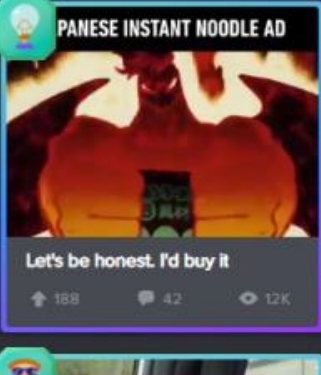
**Unmuted**  
 14,646 Posts


MORE TAGS +

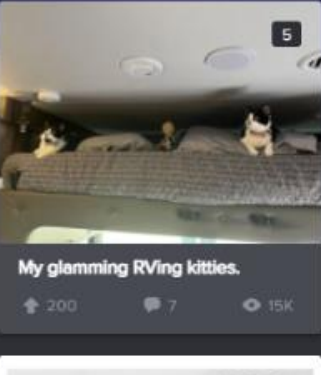
MOST VIRAL




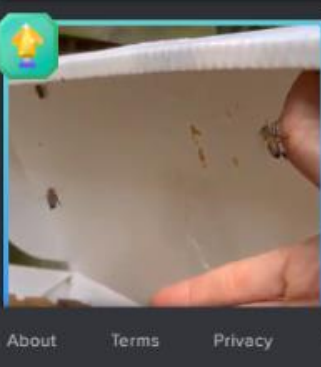


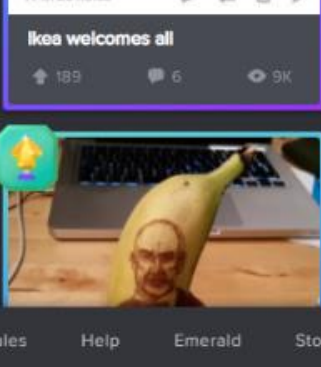


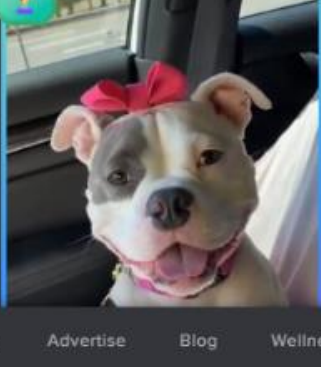


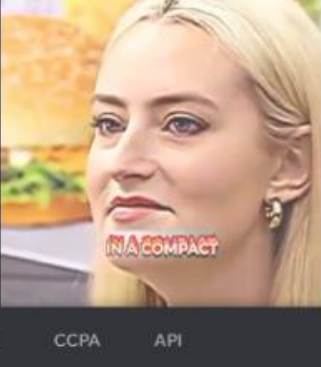


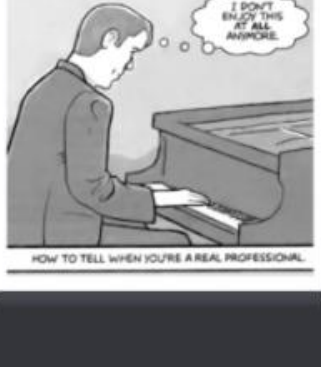






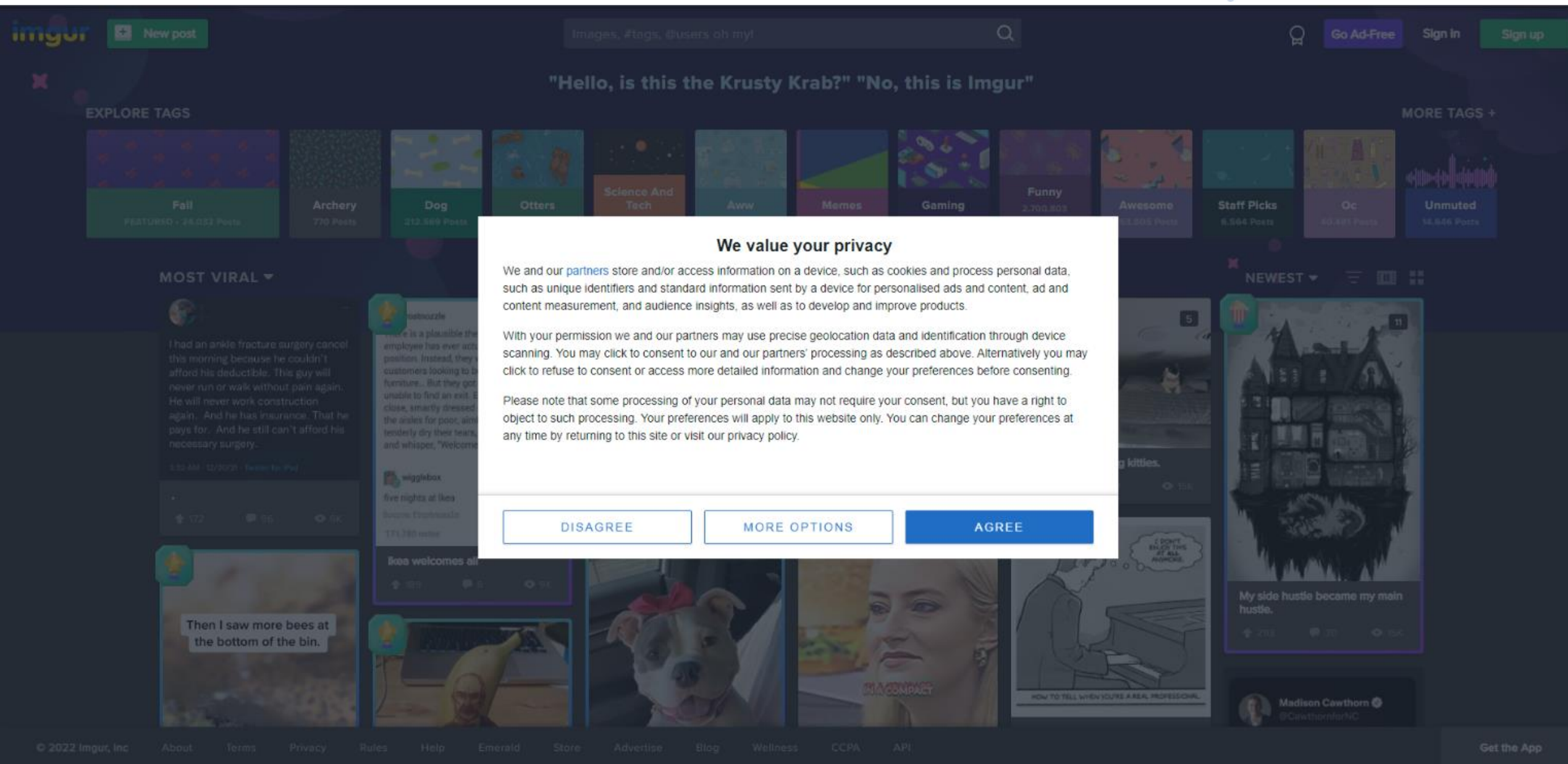






© 2022 Imgur, Inc
 About
 Terms
 Privacy
 Rules
 Help
 Emerald
 Store
 Advertise
 Blog
 Wellness
 CCPA
 API

Get
 Privacy



The image shows the Imgur homepage with a central privacy notice overlay. The background features the 'imgur' logo, a search bar, and navigation links like 'New post', 'Go Ad-Free', 'Sign in', and 'Sign up'. Below these are sections for 'EXPLORE TAGS' (Fall, Archery, Dog, Otters, Science And Tech, Awr, Memes, Gaming, Funny, Awesome, Staff Picks, Oc, Unmuted) and 'MOST VIRAL' posts. The privacy notice is a white box with a title 'We value your privacy' and three buttons: 'DISAGREE', 'MORE OPTIONS', and 'AGREE'.

**We value your privacy**

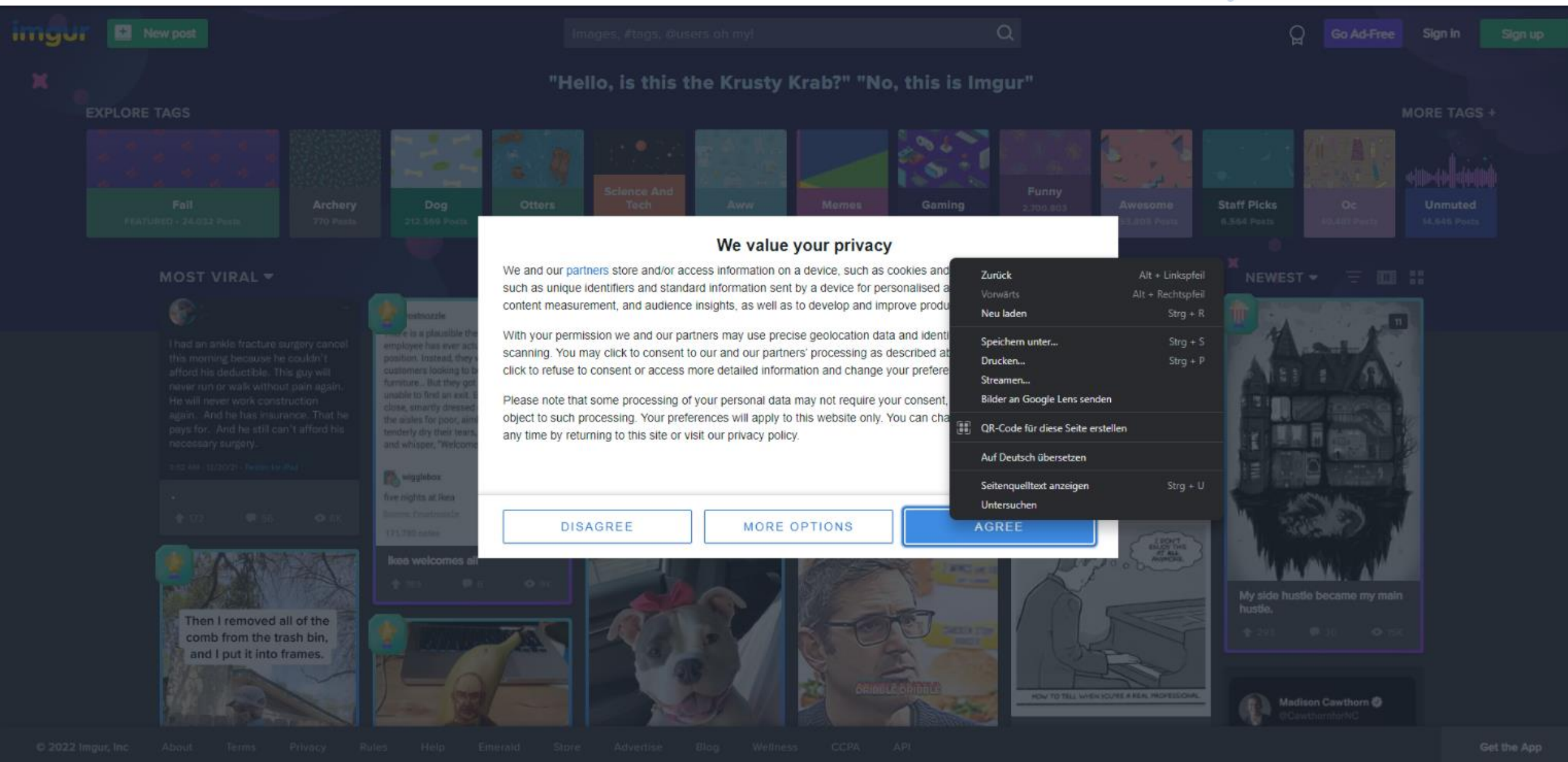
We and our [partners](#) store and/or access information on a device, such as cookies and process personal data, such as unique identifiers and standard information sent by a device for personalised ads and content, ad and content measurement, and audience insights, as well as to develop and improve products.

With your permission we and our partners may use precise geolocation data and identification through device scanning. You may click to consent to our and our partners' processing as described above. Alternatively you may click to refuse to consent or access more detailed information and change your preferences before consenting.

Please note that some processing of your personal data may not require your consent, but you have a right to object to such processing. Your preferences will apply to this website only. You can change your preferences at any time by returning to this site or visit our privacy policy.

**DISAGREE** **MORE OPTIONS** **AGREE**





The screenshot shows the Imgur homepage with a dark theme. At the top, there's a navigation bar with the Imgur logo, a 'New post' button, a search bar, and links for 'Go Ad-Free', 'Sign in', and 'Sign up'. Below the navigation bar is a banner that reads '"Hello, is this the Krusty Krab?" "No, this is Imgur"'. The main content area is divided into sections: 'EXPLORE TAGS' with various category tiles (Fail, Archery, Dog, Otters, Science And Tech, Aww, Memes, Gaming, Funny, Awesome, Staff Picks, Oc, Unmuted), 'MOST VIRAL' with a featured post about an ankle fracture surgery, and 'NEWEST' with a post about a side hustle. A central white modal box displays a privacy notice with three buttons: 'DISAGREE', 'MORE OPTIONS', and 'AGREE'. A right-click context menu is open over the 'AGREE' button, showing options like 'Zurück', 'Vorwärts', 'Neu laden', 'Speichern unter...', 'Drucken...', 'Streamen...', 'Bilder an Google Lens senden', 'QR-Code für diese Seite erstellen', 'Auf Deutsch übersetzen', 'Seitenquelltext anzeigen', and 'Untersuchen'.

**We value your privacy**

We and our [partners](#) store and/or access information on a device, such as cookies and such as unique identifiers and standard information sent by a device for personalised advertisement, content measurement, and audience insights, as well as to develop and improve our products.

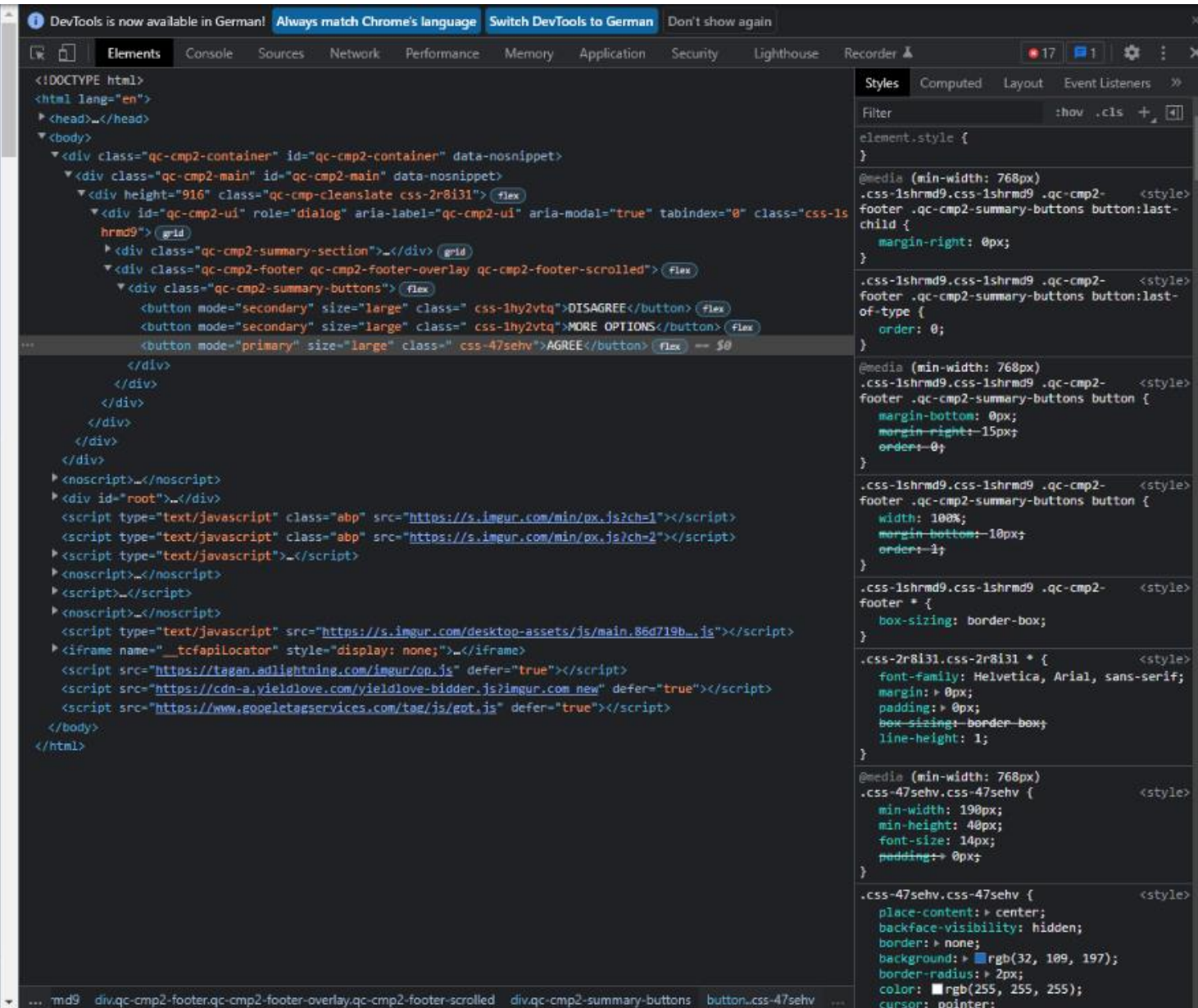
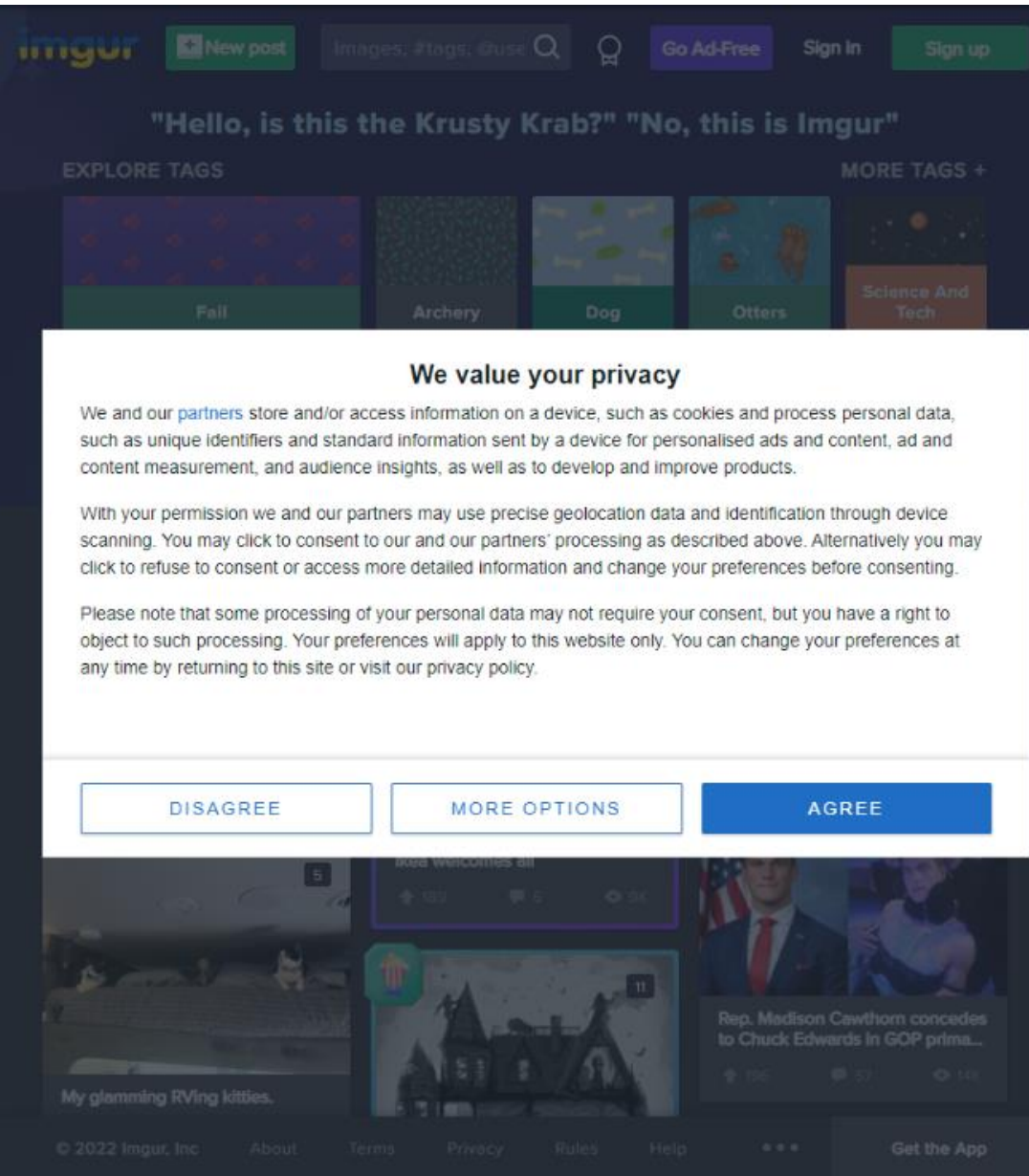
With your permission we and our partners may use precise geolocation data and identify your IP address. You may click to consent to our and our partners' processing as described above or you may click to refuse to consent or access more detailed information and change your preferences at any time.

Please note that some processing of your personal data may not require your consent, but we will still ask you to consent to it for our purposes. You can check which preferences you have already selected at any time by returning to this site or visit our privacy policy.

**DISAGREE** **MORE OPTIONS** **AGREE**

**Context Menu Options:**

- Zurück (Alt + Linkspfeil)
- Vorwärts (Alt + Rechtspfeil)
- Neu laden (Strg + R)
- Speichern unter... (Strg + S)
- Drucken... (Strg + P)
- Streamen...
- Bilder an Google Lens senden
- QR-Code für diese Seite erstellen
- Auf Deutsch übersetzen
- Seitenquelltext anzeigen (Strg + U)
- Untersuchen



## 5.2 Cookies akzeptieren

```
WebElement cookie = driver.findElement(By.className("css-47sehv"));  
  
cookie.click();
```



**MORE TAGS +**

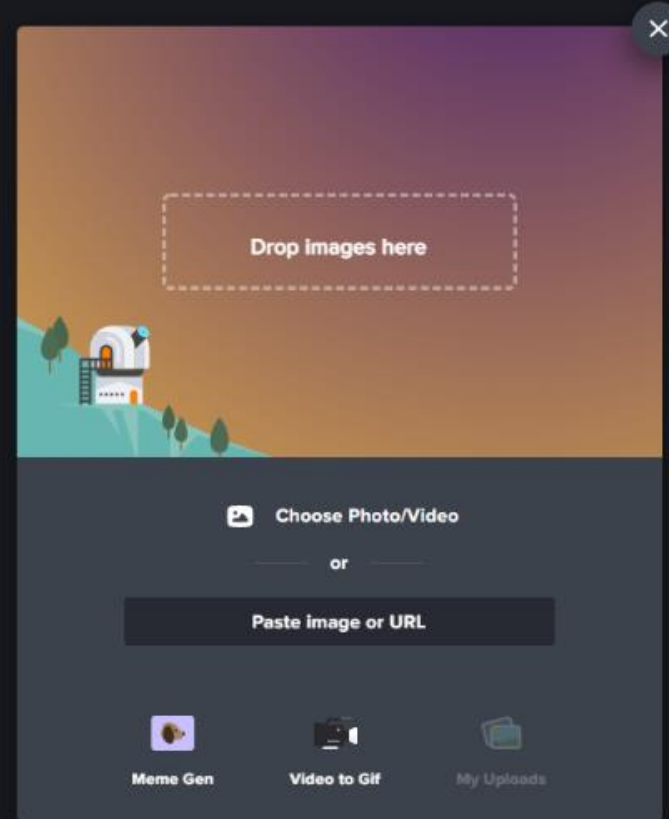
NEWEST ▼

Adam Grant   
@AdamMGGrant



## 5.3 Neuen Post auswählen

```
WebElement newPost = driver.findElement(By.className("newPostLabel"));  
  
newPost.click();
```

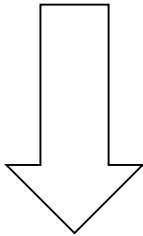


By creating a post, you agree to imgur's [Terms of Service](#) and [Privacy Policy](#)

## 5.4 Datei auswählen

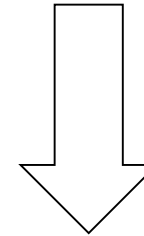
```
WebElement chooseFile = driver.findElement(By.xpath("./input[@type='file']"));  
  
chooseFile.sendKeys("\\Hund.png");
```

7 MB Bild



Kein Fehler

20 MB Bild



Fehler beim Testen

[Go Ad-Free](#)[Sign In](#)[Sign up](#)

Give your post a title...



Add a description

+ Add Image

#### POST

To Community

Grab Link

↪ Your post is currently **Hidden**

☐ Mature (?)

#### ADD TAGS

+ Tag

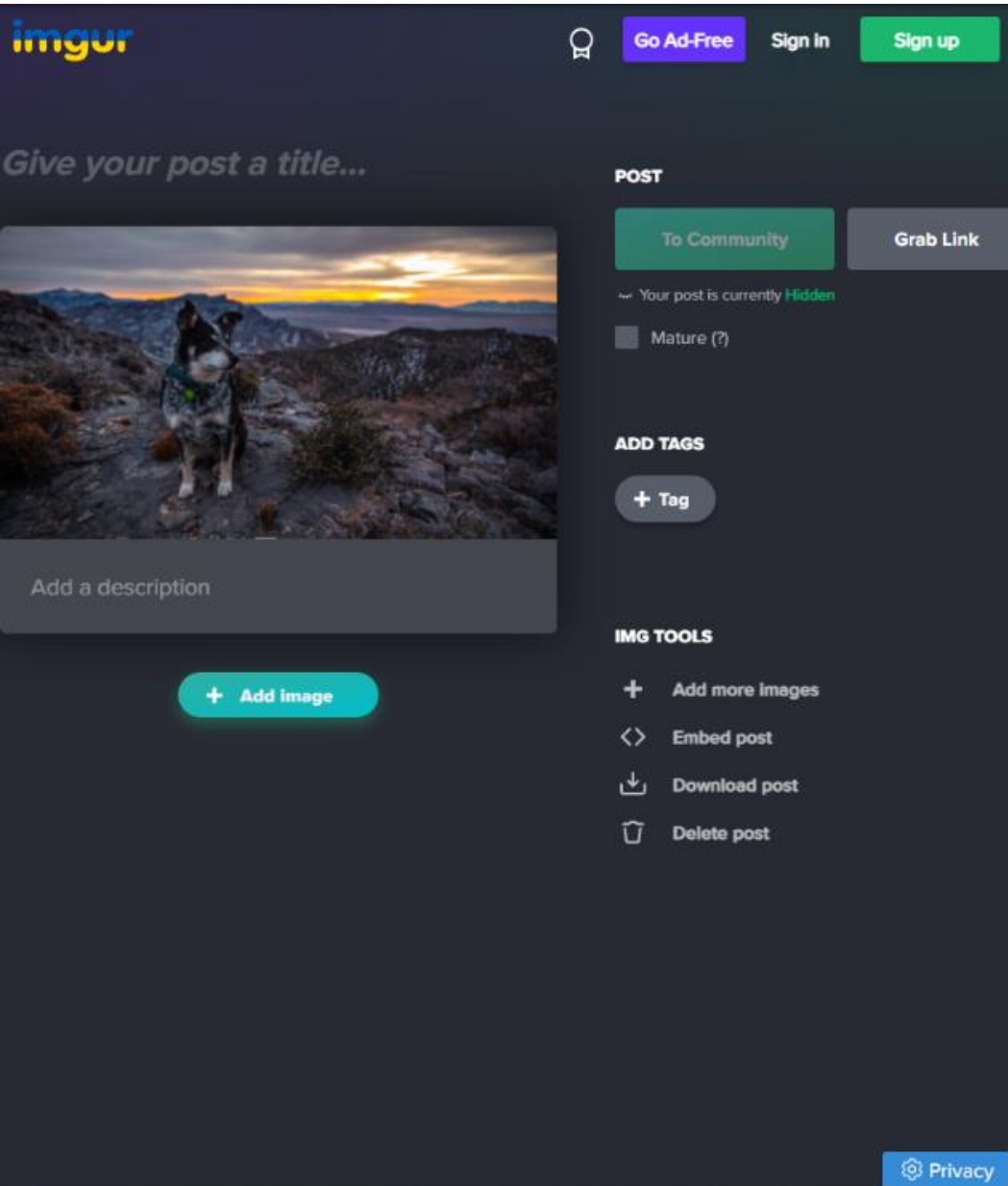
#### IMG TOOLS

+ Add more Images

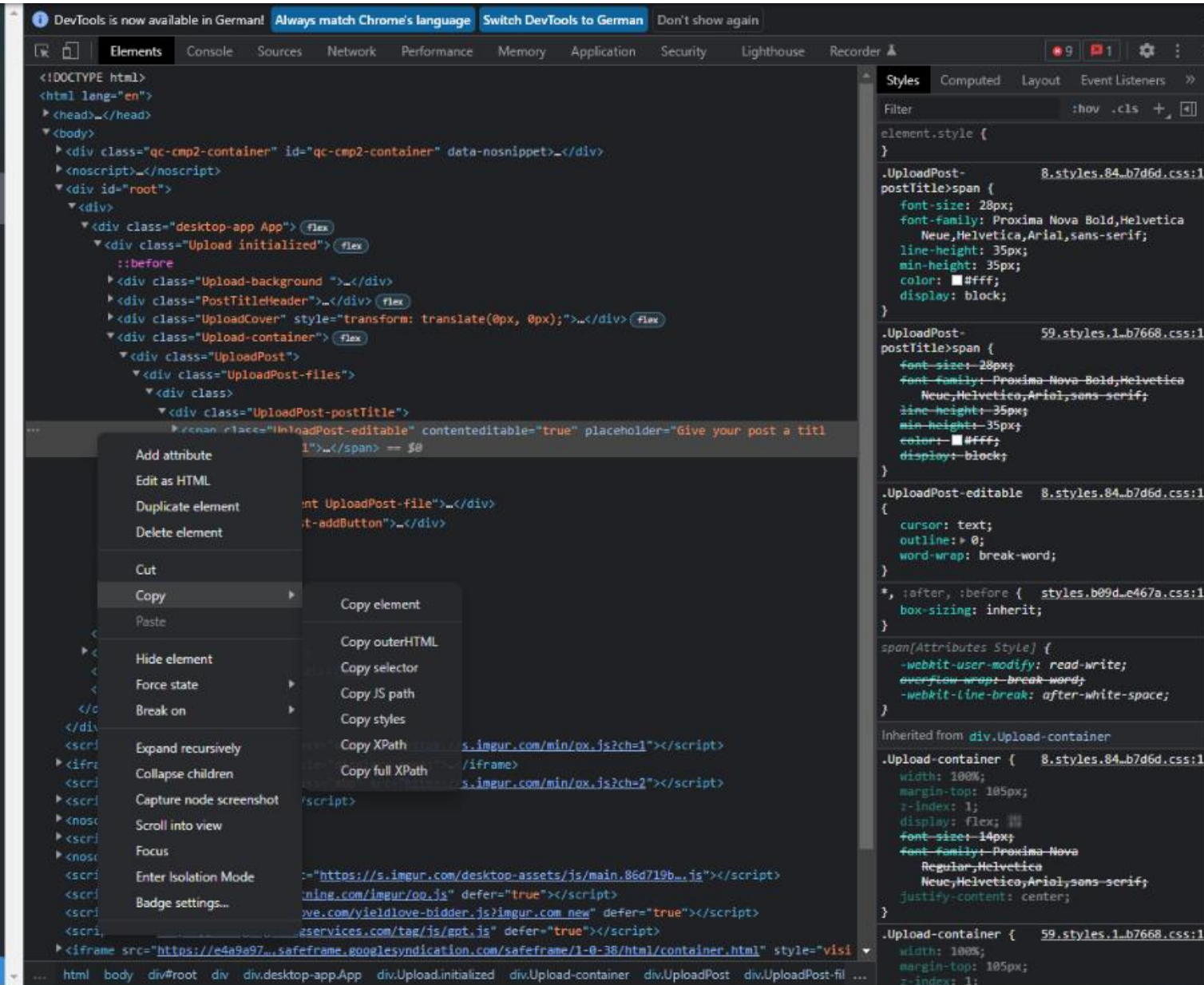
<> Embed post

↓ Download post

🗑 Delete post



The image shows the imgur.com web interface. At the top, there's a navigation bar with 'Go Ad-Free', 'Sign In', and 'Sign up' buttons. Below this, a large image of a dog is displayed. To the right of the image, there's a 'POST' section with 'To Community' and 'Grab Link' buttons. Below the image, there's a 'Mature (?)' toggle and an 'ADD TAGS' section with a '+ Tag' button. At the bottom, there's an 'IMG TOOLS' section with buttons for 'Add more images', 'Embed post', 'Download post', and 'Delete post'. A 'Privacy' link is at the bottom right.



The image shows the Chrome DevTools interface. The 'Elements' panel is open, showing the HTML structure of the page. A context menu is open over the 'UploadPost-postTitle' element, with options like 'Add attribute', 'Edit as HTML', 'Duplicate element', 'Delete element', 'Cut', 'Copy', 'Paste', 'Hide element', 'Force state', 'Break on', 'Expand recursively', 'Collapse children', 'Capture node screenshot', 'Scroll into view', 'Focus', 'Enter Isolation Mode', and 'Badge settings...'. The 'Styles' panel is also open, showing the CSS rules for the selected element. The 'Computed' tab is selected, showing the final styles applied to the element.

## 5.5 Titel und Beschreibung hinzufügen

```
WebElement title =  
driver.findElement(By.xpath("//*[@id=\"root\"]/div/div[1]/div/div[4]/div[1]/div/div[1]/div/s  
pan"));  
  
title.sendKeys("Bild eines Hundes");
```

```
WebElement description =  
driver.findElement(By.xpath("//*[@id=\"root\"]/div/div[1]/div/div[4]/div[1]/div/div[2]/div[2  
]/div[1]"));  
  
description.sendKeys("Ein Bild eines Hundes, der in den Bergen sitzt");
```



[Go Ad-Free](#)[Sign In](#)[Sign up](#)

## Bild eines Hundes



Ein Bild eines Hundes, der in den Bergen sitzt

[+ Add Image](#)

### POST

[To Community](#)[Grab Link](#)

↩ Your post is currently **Hidden**

☐ Mature (?)

### ADD TAGS

[+ Tag](#)

### IMG TOOLS

[+ Add more images](#)[<> Embed post](#)[⬇ Download post](#)[🗑 Delete post](#)

## 5.6 Titel Eingabe Test

```
WebElement titleInput = driver.findElement(By.className("UploadPost-postTitle"));  
  
String pictureTitle = titleInput.getText();
```



## 5.7 Tests

```
boolean thrown = false;

try {

    WebElement test = driver.findElement(By.className("UploadPost-
postTitle"));

} catch (org.openqa.selenium.NoSuchElementException e) {

    thrown = true;

}

Assertions.assertTrue(thrown);
```

[Go Ad-Free](#)[Sign In](#)[Sign up](#)

Give your post a title...



Add a description

+ Add Image

#### POST

To Community

Grab Link

↪ Your post is currently **Hidden**

☐ Mature (?)

#### ADD TAGS

+ Tag

#### IMG TOOLS


+ Add more Images

<> Embed post


↓ Download post


🗑 Delete post






Drop Images here





 Choose Photo/Video

or


Paste Image or URL

 Meme Gen

 Video to GIF

 My Uploads

By creating a post, you agree to Imgur's [Terms of Service](#) and [Privacy Policy](#)

Woah there! Your file is too large. [Learn more.](#) 

## 5.7 Tests

```
title.sendKeys("Bild eines Hundes");

driver.manage().timeouts().implicitlyWait(Duration.ofMillis(10000));

WebElement titleInput = driver.findElement(By.className("UploadPost-postTitle"));

String pictureTitle = titleInput.getText();

Assertions.assertEquals("Bild eines Hundes", pictureTitle);
```



# Selenium

**Danke für Ihre Aufmerksamkeit.**

**Christoph Schwarz – 83352**

**Florian Janot – 80785**

**Pauline Neumann – 82884**

**Philip Ritzer – 83127**

Hochschule Aalen  
Softwareengineering und Tools  
Prof. Dr. Marc Fernandes  
15-06-2022

# Quellen

- <https://www.testautomatisierung.org/wp-content/uploads/263.png>
- <https://www.selenium.dev/projects/>
- <https://testautomatisierung-gewusst-wie.de/testautomatisierung-von-webanwendungen-mit-selenium/>
- <https://entwickler.de/java/selenium-ide-capture-and-replay-testing/>
- <https://www.cabotsolutions.com/continuous-integration-testing-using-selenium>
- <https://blog.itil.org/wp-content/uploads/2016/07/DevOps-Cycle.png>