The Gruntled Product Spec

Table of Contents

Product Summary

Gruntled is a Chrome Browser Extension that lets the user record & share their reaction to a particular YouTube video. Users are able to consolidate reactions with friends with a shareable link. The final result being, both the reaction videos playing in-sync with the respective watching video.

I came up with this idea with a friend. We wanted to react to a UFC fight that neither of us had seen but had heard was a very good match. We wanted to be able to find a way to still watch & connect with each other despite not being able to meet in person or even at the same time, virtually. Thus, Gruntled doesn't have to be used in-sync to get a reaction video that is in fact, in-sync.

The final product should be a browser extension, preferably pinned on the user's extensions bar. After the user grants video permissions to Gruntled & navigate to a YouTube video, the user should be able to click the Extension & hit record. Record will automatically restart the video to the beginning & using the webcam, record the user's reaction to the video. The user then, can send a link to a friend that can consolidate both their reactions to the same video. The final key result is the consolidated reaction; both reaction videos are embedded within the watching video.

The MVP will have the two reaction videos, sized much smaller & placed below the respective watching video. As of this writing, we hope to have an MVP out by the end of April, 2022.

Business Case

Gruntled is in the space of allowing customers to seamlessly record their reaction to share & enjoy with their friends. Convenience is a huge commodity in the software industry; the ability to easily record & share a reaction instead of having to splice & edit could prove prosperous in a media-driven age. We want to create and iterate over a Chrome browser extension that allows the user to record & send a reaction to a friend for a particular video. We measure success by seeing how many users have installed our browser extension, as well as how many merged reaction videos our application creates. The browser extension will be the first product of the company. Since the user flow to download & use the extension is streamlined, we hope that many users will be familiar with the Gruntled brand.

User Stories

1) As a fight fan, I want to be able to see my friends react to insane UFC knockouts, so that I can feel a sense of connectedness & glee.
2) As a content creator, I want to be able to seamlessly consolidate my friends' reactions to videos so that I can focus on other aspects of my craft.
3) As a comic book fan, I want to be able to share my reaction to new movie trailers, so that I can show how excited I am for the release

User Personas

> **"**
> I've been learning to how to edit but I would appreciate a way to quickly put together a reaction video.
> **"**

Bio

Shanté is a lifestyle content creator. She loves to include her friends in her reaction videos. (i.e. certain scenes from her favorite movie she's watching.)

Shanté Quinn

22 • Content Creator • Fashionista

Motivations

CONVENIENCE	✳ ✳ ✳ ✳
PERSONALIZATION	✳ ✳ ✳
SOCIAL	✳ ✳ ✳ ✳ ✳

Goals

• to be able to put together a reaction video with friends for a particular scene

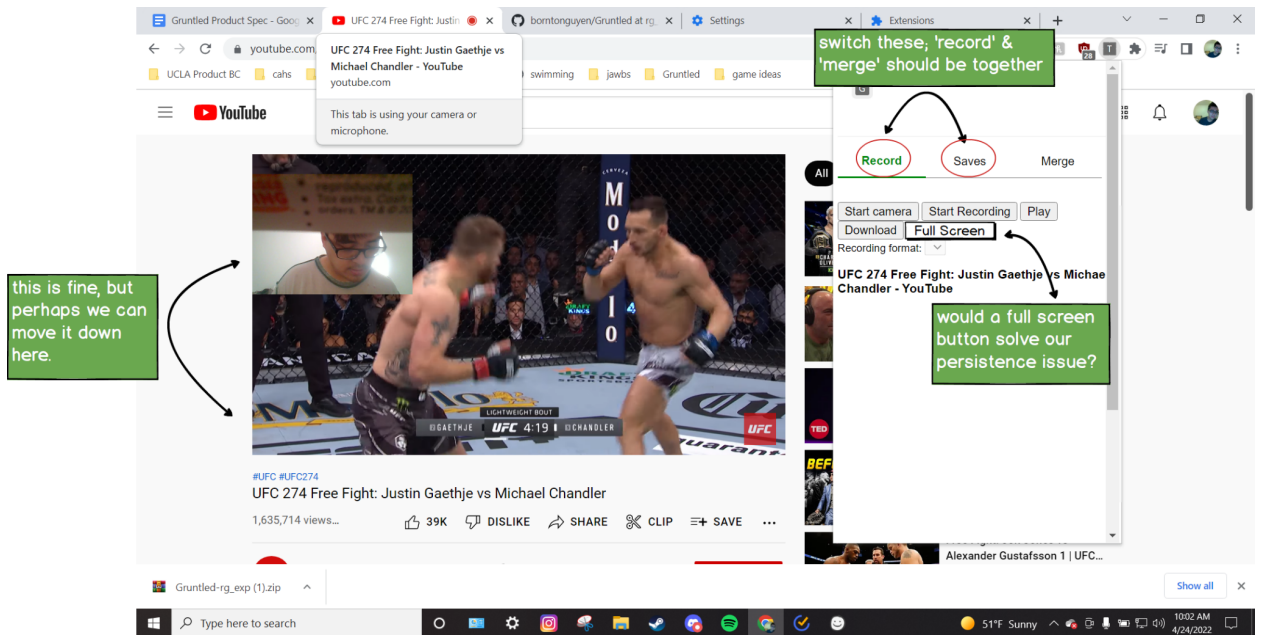• to be able to record anytime to share with others later

Pain Points

• difficult to meet friends because of time & distance constraints

• doesn't have the resources to buy/learn video editing software.

Type of Videos

• Live Fashion Drops

• Scenes from favorite sitcom

Product Design

this is fine, but perhaps we can move it down here.

switch these; 'record' & 'merge' should be together

would a full screen button solve our persistence issue?

github.com/borntonguyen/Gruntled

Functional Spec

popup.js ...currently implemented in this file

Should Go in Filename

POPUP.JS
- addCustomEventListener(html_element, custom_event_func)
- setupTabs()
- setupWebcam() [BROKEN]
- getSupportedMimeTypes()
- getCurrentTab() [async]

YT_CONTROL.JS
- play()
- pause()
- restart()

CONTENT.JS
- startRecording()
- handleDataAvailable(event)
- stopRecording()
- recordToggle()
- startCamera()
- playbackVideo()
- downloadVideo()

Youtube Webpage | **Injected Script**

Popup.js

On Extension Launch → yt_controls.js

On startRecording Button Press → Webcam.js

add condition → Content.js

yt_controls.js — ...g youtube video, injected ...n is clicked.

Webcam.js — ...ccess and set up Media stream, ...bcam preview.

content.js

github.com/borntonguyen/Gruntled