Twitter Thread by Pat Walls

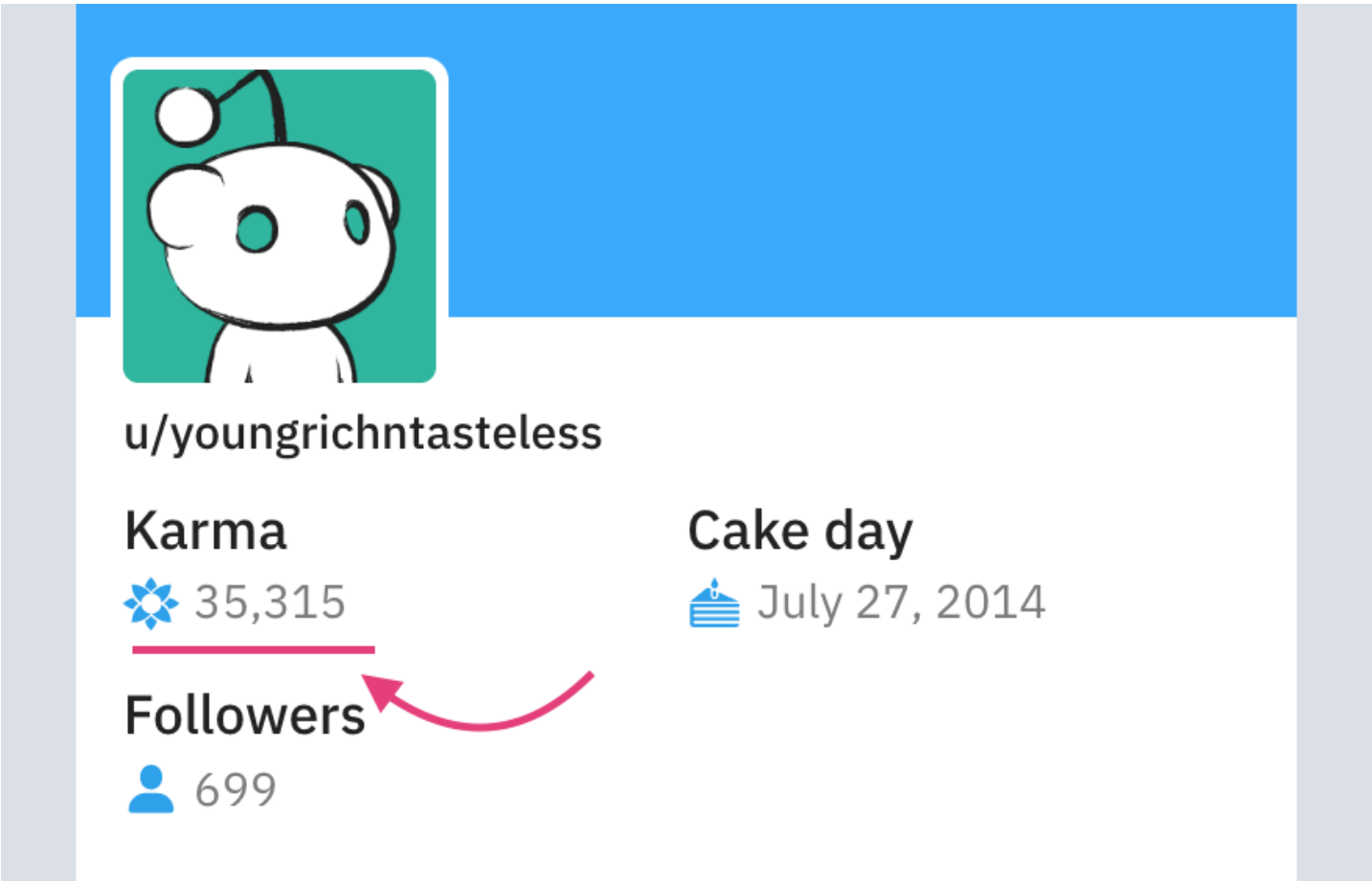


Pat Walls
@thepatwalls

How I auto-share ~120 blog posts to Reddit every month, reached 35k karma, and gotten millions of views.

And why I love building automation so much!

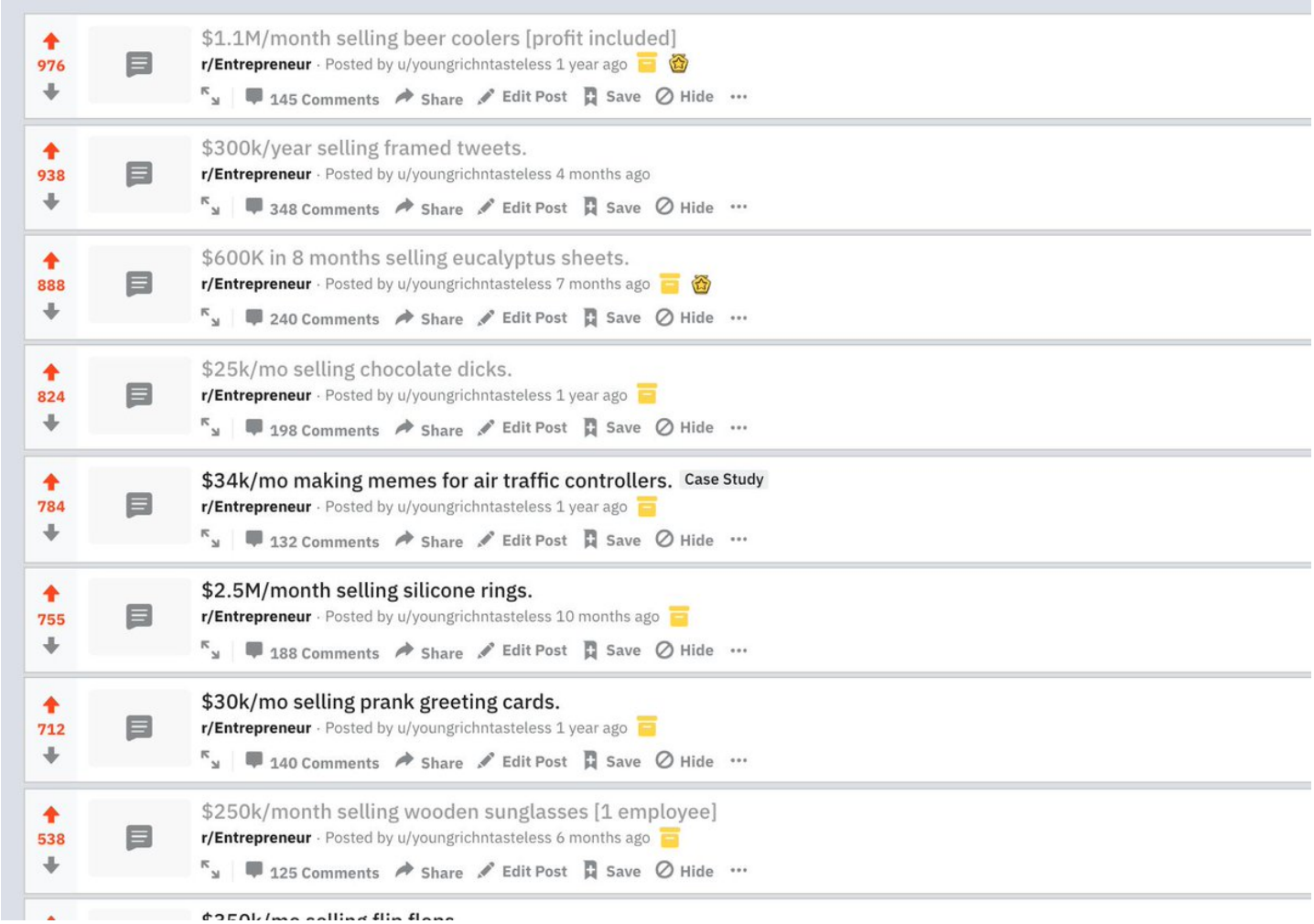
A thread!



I have hit the top of Reddit hundreds of times.

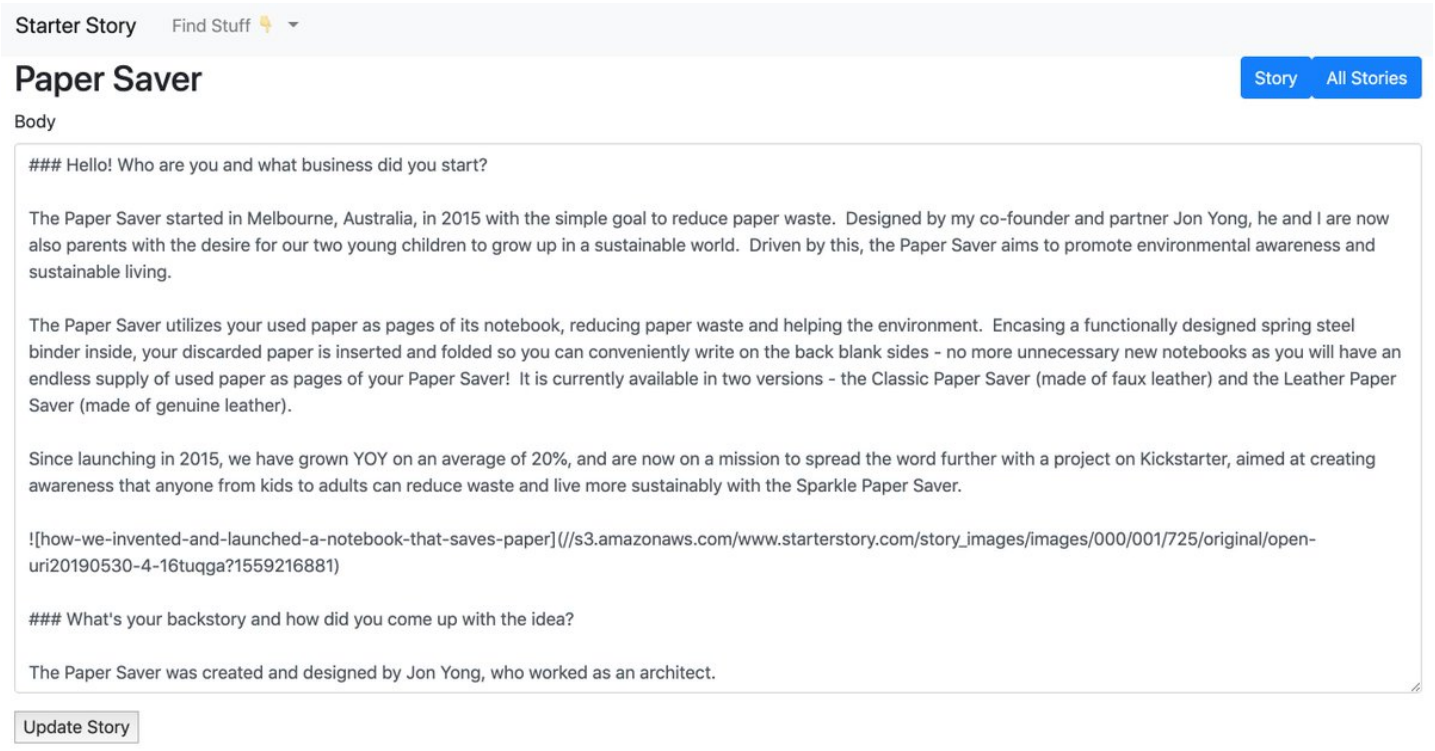
Through all of the love and hate that I get on the platform, I think I've found that it's here to stay.

After years of posting on Reddit, I've slowly built an automated system that is now 100% hands off.



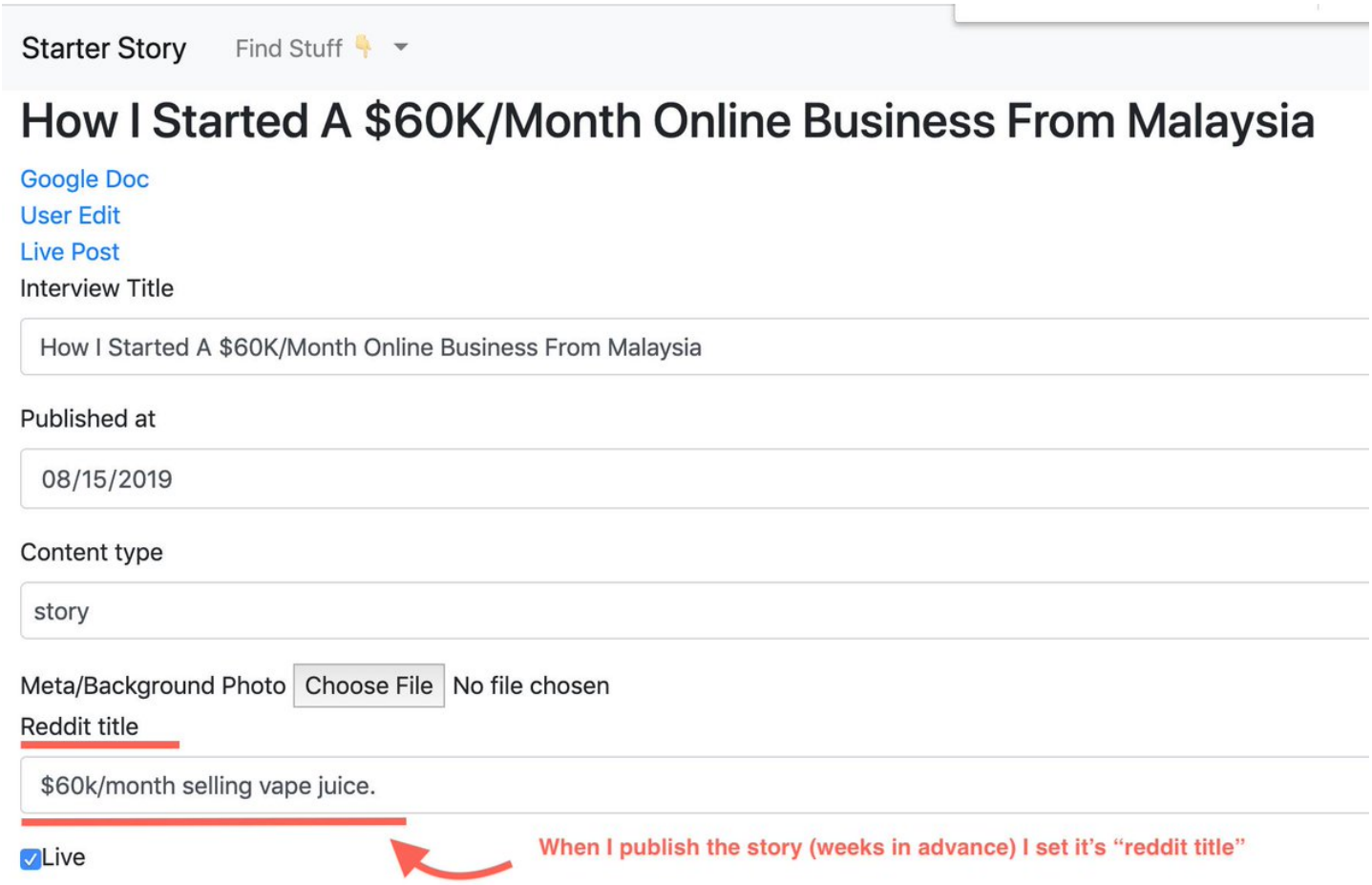
This is my Starter Story backend, where I save the content of my stories in a markdown format.

This is great for Reddit because Reddit is also in a markdown format!



While I edit and publish starter stories, I set the "reddit title" in advance.

This is so I don't really ever have to think about it again. Once that's set, the system knows it should post this one to reddit, and that should be the title.



I also create a "Reddit intro" and "outro"

Although the content of the blog post is stored in my database, I still should introduce myself in the Reddit post.

I use other variables in my database to construct this.

```
  def reddit_intro_text(story)
    "Hey - Pat from [StarterStory.com](https://www.starterstory.com) here with another interview.

Today's interview is with #{story.starter.name}#{story.starter.reddit_username.present? ? " (/u/#{story.starter.reddit_username})" : ""}

**Some stats:**

* Product: #{story.business.description}
* #{story.business.measurement&.capitalize}: #{number_to_currency(story.business.revenue, precision: 0)}
* Started: #{story.business.business_started_at&.strftime("%B %Y")}
* Location: #{story.business.location}
* Founders: #{story.business.num_founders}
* Employees: #{story.business.num_employees}\n\n"
  end

  def reddit_outro_text(story)
    "\n\n---

Liked this text interview? Check out the [full interview with photos](#{story_url_with_deprecated(story)})."
  end
```

I even tag the Reddit user (just recently starting tracking this data in my database) as they will get a notification when their post goes up.

This (and other data) is filled out when the interviewee is plugging all their info.



Here's another fun thing.

Reddit has an algorithm that will "shadow ban" (google it) your posts if you have certain links.

I have a list of links that I will strip from the markdown automatically:

(this list has been built from trial and error mostly)

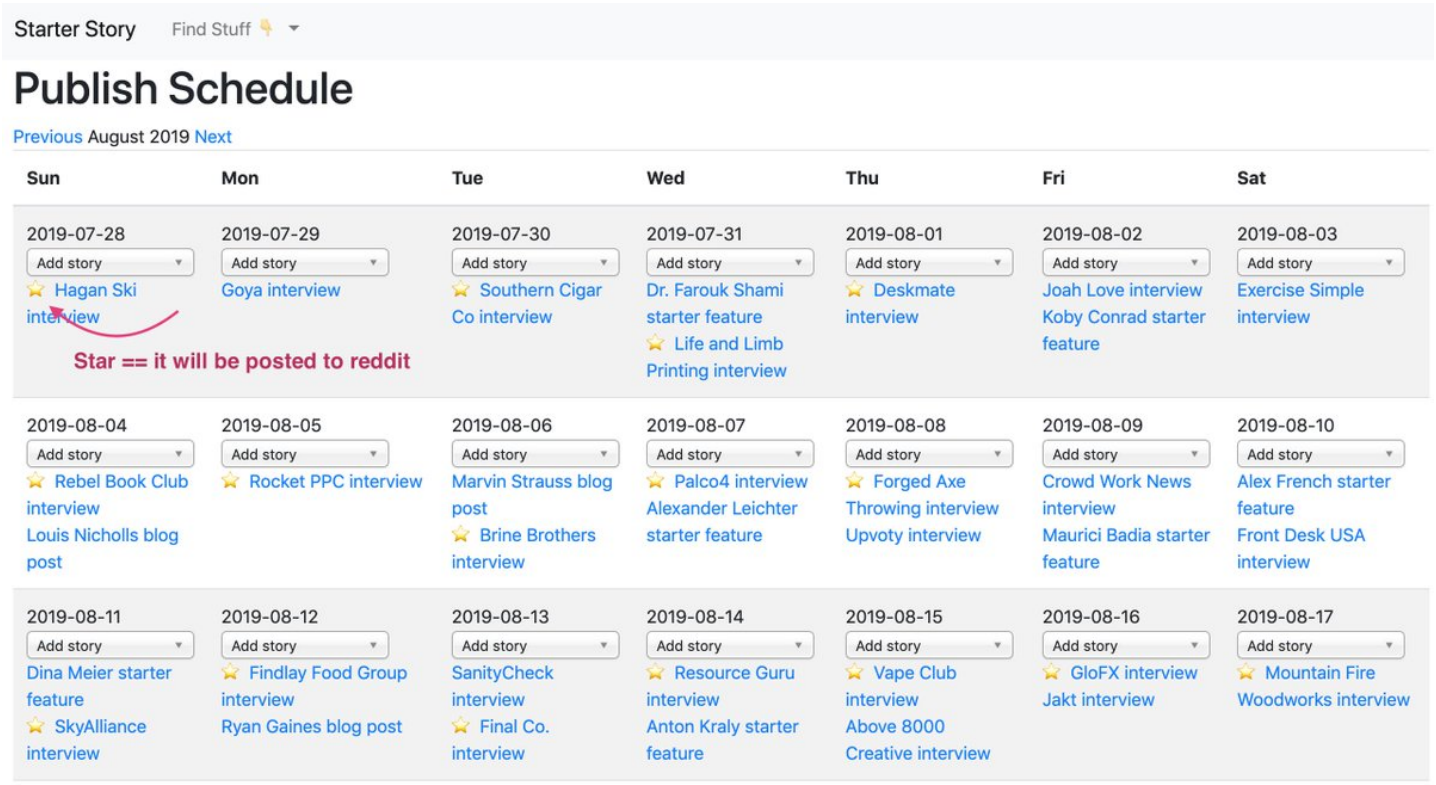
```
class Reddit < ApplicationRecord
  BLACKLISTED_DOMAINS = [
    "shopify.com",
    "apps.shopify.com",
    "amazon.com",
    "starterstory.com",
    "squarespace.com",
    "mailchimp.com",
    "shipstation.com",
    "google.com",
    "klaviyo.com",
    "etsy.com",
    "intuit.com",
    "rechargepayments.com",
    "basecamp.com",
    "goldmansachs.com",
    "gimletmedia.com",
    "slack.com",
    "streak.com",
    "lastpass.com",
    "judge.me",
    "boldcommerce.com",
    "audible.com",
    "evernote.com",
    "wordpress.com",
    "woocommerce.com",
    "codecanyon.net",
    "woowaitlist.com",
    "tawk.to",
    "activecampaign.com",
    "magento.com",
    "canva.com",
    "mixpanel.com"
```

I also need to make sure we only post one story per day to Reddit. Maybe one day I will increase this.

This is our content calendar.

When you see a■■that means it will be automatically posted to Reddit.

This content calendar allows us to schedule WEEKS in advance with stories.



Then, once a day, a worker runs that will post that story to 4 different subreddits (you can see which ones in screenshot)

I run these at 15 minute intervals to avoid tripping any spam things.

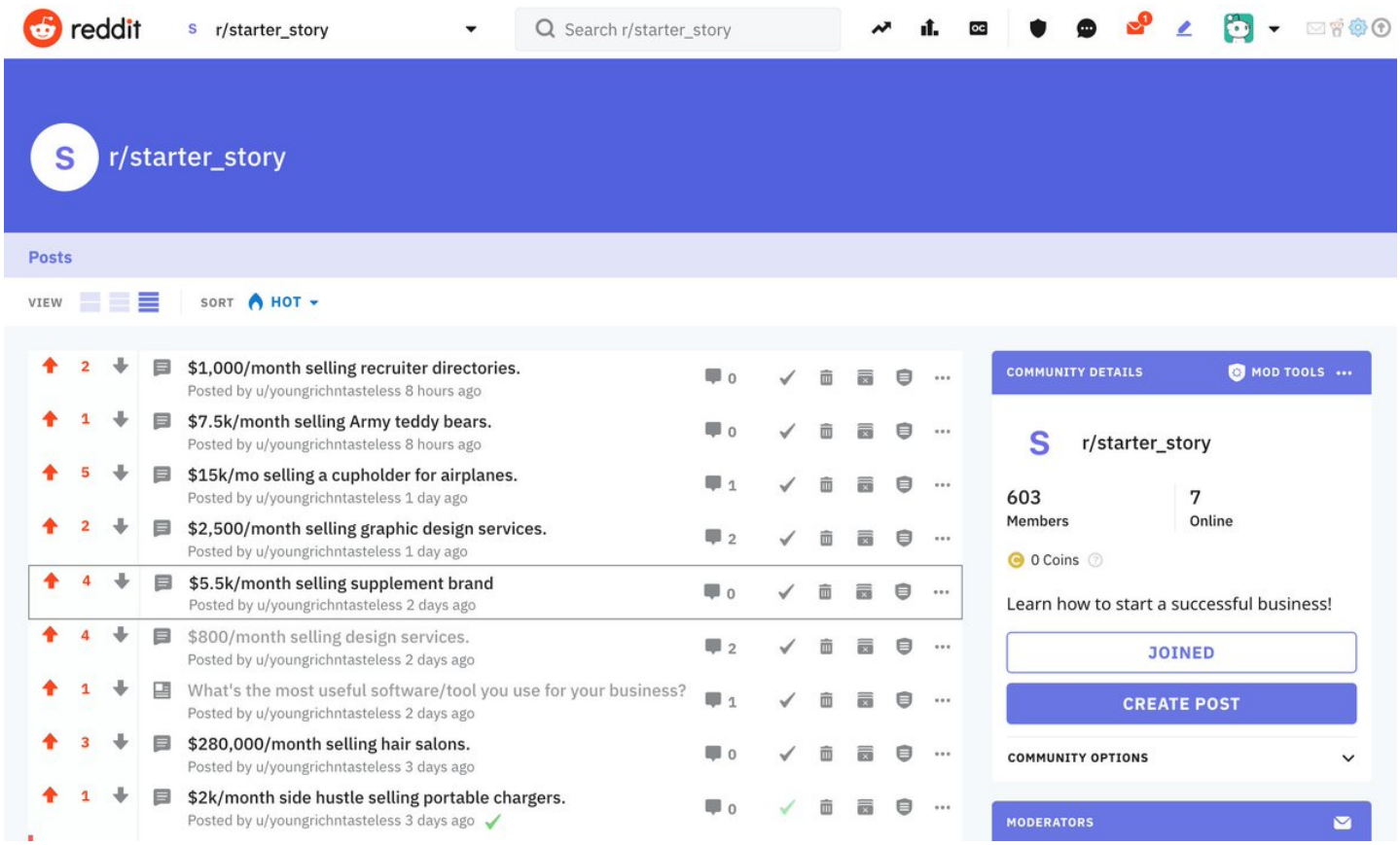
Here's the code. Just a few lines.

```
def perform
  return unless Rails.env.production?

  ["entrepreneur", "EntrepreneurRideAlong", "business_ideas", "starter_story"].each_with_index do |subreddit, idx|
    stories_published_today.each do |story|
      next if story.reddit_title.blank? && subreddit != "starter_story"

      title = story.reddit_title.present? ? story.reddit_title : generate_reddit_title(story)
      text = reddit_full_text(story)

      RedditWorker.perform_at((Time.now + (idx * 15).minutes), subreddit, title, text)
    end
  end
```

I also started my own subreddit!

https://t.co/wkIMCyNfw3

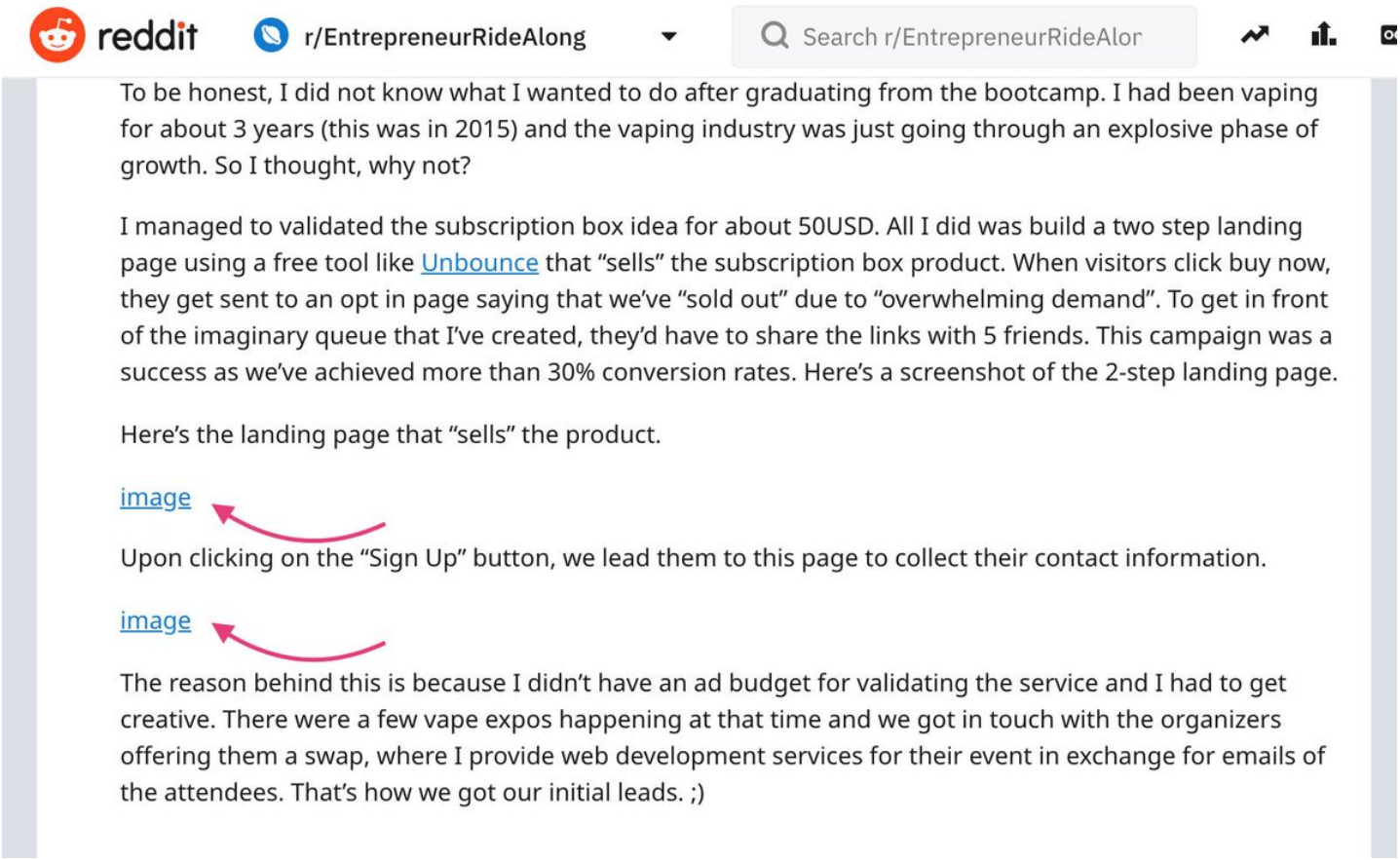
I post ALL stories there, and we just passed 600 members. Small, but growing a little every day :)



In addition to stripping blacklisted links for the reddit post, I also do a few other things:

- strip out affiliate link params
- remove blockquotes
- add links for the images

This gives the post a "less bloggy" feel IMO.



The automation timeline!

This has all been a process:

1. at first I manually posted and found what works
2. then i set reminders to do it
3. then i built the code that writes the intro (copy/paste)
4. then i configured the API

5. then i made it so i could just click "post to reddit"
6. then i created the worker

This is high level, but I slowly did this and tweaked over the last year. Many bumps in the road.

Automating things is an iterative process!!!!

The benefits of automating?

Other than the obvious benefits (save time/money) the biggest benefit is that I'm not constantly checking Reddit anymore, seeing how upvotes are going, seeing the latest hater comments, etc.

I'll often check the site and find my post hitting the front page and I had no idea!

Anyways, that's it. Hope you found something interesting out my fun little automation.

Even though it's simple, I'm proud of it!

Happy to answer any questions. What do you think of this?