Elite Editing

Technical Writing Test

***About technical writing at Elite***

At Elite, we regularly tackle different types of content, and we love writers who are comfortable dealing with stylistically varied genres—but we also have places for those who excel at a particular kind of writing.

**Responsibilities**

* Write case studies and blogs for a technology client based on stakeholder interviews and written assets.
* Follow the *Chicago Manual of Style*, 17th edition, with exceptions for the technology client’s unique style guidelines.
* Incorporate changes to drafts based on stakeholder feedback.
* Collaborate with a large team to circulate new knowledge and keep process guides up to date.

**Skills and qualities**

* You have excellent communication skills.
* You can keep up with detailed and evolving style guidance.
* You accept feedback cheerfully.
* You can learn and apply in-depth processes.
* You have an intermediate knowledge of MS Word, including using comments and track changes.
* You have background knowledge of or interest in technology or computer science (helpful but not required).

**Time commitment**

There is no set time commitment; however, because there is a learning curve, the ideal candidate will be interested in devoting 10 hours or more per week to projects they accept.

**Deliverables**

Deliverables include 1- and 2-page customer-reference case studies, technical blog posts, assets for social channels, and various add-ons like PowerPoint presentations.

**Rates**

Pay varies based on experience, skill, and assignment, but as a guide, a writer working on a standard 1,000-word case study can expect to earn $150–$300.

Technical Writing Test

CloudSky Customer Stories: DressNinja

**Instructions and Writer Guidelines**

You will be writing a 400–500-word customer reference story for a fictitious client CloudSky about how a customer (“DressNinja”) used CloudSky’s technology solutions.

* Review the provided **DressNinja Quotes** and the **DressNinja Guidance/Ideation** for this study.
* Look over the **CloudSky Company Background**, **CloudSky Service Descriptions**, **CloudSky Style Sheet**, and the **Sample Case Study: AngelDoc** (an example of a customer reference story about a different CloudSky customer).
* Find the story: Discuss the challenge that returns pose to online retailers and how DressNinja edged closer to the “holy grail” of ecommerce with help from CloudSky services. Start with a compelling, concise intro, then introduce the customer and its challenge. Explain how CloudSky was a solution to the challenge and highlight the benefits that DressNinja has received as a result of the CloudSky solution. However, avoid portraying CloudSky as the “hero”; instead, highlight DressNinja and what it was able to achieve. Close with a compelling quote.
* Identify relevant quotes from Mary and incorporate throughout. Please edit the quotes for grammar and CloudSky style.
* Describe how DressNinja used each CloudSky service to build a solution. Avoid creating a “laundry list” of the services; instead, use a combination of Mary’s quotes and the **CloudSky Service Descriptions** to create a succinct, cohesive explanation.
* When introducing CloudSky services, use the full name on first mention and define them with language from the **CloudSky Service Descriptions**. See **Sample Case Study: AngelDoc** for an example as to how this is done.
* In the **Writing Test** at the end of this document, write a short title and a story of 400–500 words summarizing how DressNinja used CloudSky to improve its business. Make sure the character count is within reason; no more than 20 percent above the stated limit.
* When you’ve completed this test (as well as any other tests you’ve decided to take), please [click here](https://form.jotform.com/241645222115143) for a submission link.
* Once you’ve opened the submission link, answer the questions on the upload form, attach *all* completed tests, and then hit “Submit.”

**DressNinja Quotes**

**Excerpts from interview with Mary Monroe, DressNinja lead of machine learning research and platforms**

“So as effectively the mandate to the machine learning organization at DressNinja is very straightforward. It’s effectively trying to make the customer experience as easy as possible. So when it comes to eCommerce, the kind of Holy grail is trying to get the customer to feel as comfortable as they would feel in a store with someone assisting them shopping. They know how something fits, how it looks, relatively confident of a purchase when you walk out the door, and not be worried about, oh, I need to return this, or what’s the turnaround time going to look like, et cetera. Now DressNinja has got a numerous value props in place to help achieve this, including binary concepts like free returns and advanced refunds, et cetera. But you know, features like that have come to become table stakes in the retail eCommerce space.”

“So we were challenged with how do we differentiate ourselves further using machine learning. So one of the core areas that we obviously struggle with is offering free returns is an extremely expensive thing to do operationally. So how could we reduce our return rates without compromising the customer experience, by which I mean not charging for returns or taking away free returns, but how do we make it a win win situation. So an idea we stumbled across was sizing predictions. It could be used machine learning to better understand how each piece of clothing in our inventory fits each individual customer so that we could make that recommendation at runtime while the customer was going to make the purchase. And hopefully if we get it right, reduce the likelihood of what we call a sizing related return. That is someone buys the exact same product in different sizes, may be the same order or subsequent orders trying to find the shirt that fits.”

“Another opportunity that came up was tackling real time search relevancy that not only is using a search algorithm that’s generic. Do all the searches that occur in DressNinja would using that algorithm that’s dynamic to each search term that occurs. And when we succeeded at that and saw relatively good gains, we went further and asked ourselves the questions instead of it being relative to the search term. Can it also be unique to each individual customer? And I don’t mean when I say unique to each customer, I don’t mean segmentation or customer groups or things like that. Can we understand you personally and provide a unique set of social results for a given search term.”

“Again, so both of these projects have been tremendously successful and the biggest caveat for each of these experiences was that we needed to make these decisions, hundreds of thousands of times a second in real ... Hundreds to thousands, not hundreds of thousands of time a second, in real time because we couldn’t precompute these predictions because the volume was just too large, which means it very quickly turned from a machine learning problem, into an engineering and technological challenge to be able to do this with a meeting customer facing SLAs. So we weren’t introducing massive latency on the site. And that’s where the various CloudSky fueling came in, to kind of facilitate this vision by providing technological stacks that allowed us to execute on these ideas while leading those customer facing SLS.”

“We’re playing in the millisecond space here. We’re introducing a lot of smarts into the search experience, and it won’t come for free. If you’re doing more stuff, it takes more time.”

“I would like to point out very, very thoroughly that we have aggressive opt out mechanisms where if you do not want to be a part of this experience and you do not want us to personalize for you and very bold call outs, you can do an off the search personalization and be like do not do this for me, leave me alone. Because one thing we’ve learned as the machine learning team that owns the personalization experience is that a part of personalizing is knowing when not to personalize because there is a large portion of your customer base that doesn’t want you interfering with that digital experience, and we very cognizant of that and that’s very important as one of the core tenants of machine learning in general.”

“We heavily rely on caching. We try and break down the machine learning into ensemble approaches. Even though we need to make the prediction at a runtime. What we’re really doing is we are blending the output of multiple machine learning models at runtime, which allows us to precompute the output of some of the machine learning models.”

“So what’s important we learn is taking the answer and breaking it down into multiple questions, right? And then figuring out which components can be pre-computed, which components needed to be blended at runtime and kind of effectively drawing out like a graph of operations that needed to occur and marking off the pieces that could be precomputed and cached so that we were only doing at runtime would absolutely needed to be done at runtime.”

“So it’s not like one massive neuronetwork they’re trying to make all these decisions right there and then. It’s an assortment of much simpler machine learning models, simpler probabilities that are being accessed to solve the problem in the minimal amount of time.”

“So Kangaroo is kind of what we rely on to store any of the precomputed solutions, so it’s a very, very good place to store something that you’re going to be doing quickly look-ups against. It’s very low latency. It’s extremely fast and it’s effectively our go to solution for any pre-computed predictions if they’re going to be accessing and runtime.”

“Redis is--we leverage CloudSky’s Emu and Redis is a solution that we use. That’s where we rely on that as a cache layer before we fall back to Kangaroo, we will have some kind of a DTL on predictions that we know aren’t going to go stale that we store in Redis temporarily.”

“Predictive Platypus is like bread and butter when it comes to training a lot of these models. A lot of the pre-computed models require a feature engineering training and then large scale predictions to occur. And even the pre-computer models are updating in days, not weeks or months. So we’ve got an automated ML engine that we maintain in house that heavily leverages P2 as a compute resource to train and make these batch predictions and then automatically deploy the latest versions into Kangaroo for our microservice API’s to expose.”

“And another service that we leverage pretty heavily is Koala as an event ingestion platform, which is where we get the information that we learn from. And Quokka, which is one of CloudSky’s machine learning platforms and services that we leverage for making evaluations at runtime. So the non-precomputed models, we have model artifacts deployed and exposed via Quokka endpoints to make those predictions at runtime.”

“So what’s really nice about CloudSky is there’s cookie cutter solutions that are present, but all those solutions are also built up using building block CloudSky services, which allows us to do the same. I honestly don’t think we’d be able to pull this off without CloudSky. There are some very, very complex CloudSky Technology Solutions services that we use to execute on this experience. And then there’s basic building blocks like Baby Wallabee and P2 and Kangaroo and Wombat that allow us to put together complex architectures of our own because of how well these pieces all fit together.”

“We have seen significant drops in sizing-related return rates when we AB tested the experience. We’ve seen a significant increase in searched product click-throughs, significant drop in search refinement, significant increase in search position like click-throughs happening higher up in the search results—all indicators of the experiences objectively becoming easier.”

**DressNinja Guidance/Ideation**

**Basic premise**

DressNinja developed a complete machine learning platform using CloudSky Analytics. The engine is made up of a real-time event ingestion system that relies heavily on CloudSky Koala, CloudSky Kangaroo, and CloudSky Predictive Platypus. It feeds into various ML algorithms, powering a myriad of features on the DressNinja website, including near-real-time search optimization and size recommendations.

**About DressNinja**

DressNinja was founded in 2003 as a costume shop, and it has since expanded into clothing. Dedicated to a more discreet experience for its customers, DressNinja employs about 1,700 people and is headquartered in Los Alamos, NM.

**CloudSky Company Background**

CloudSky Technology Solutions (CloudSky) is one of the world’s most popular cloud platforms, offering more than 125 fully featured services from data centers located around the world. Hundreds of thousands of customers support their business through the increased scalability and availability that using CloudSky provides.

CloudSky Technology Solutions (CloudSky) seeks to share stories of how companies around the world have used CloudSky services successfully to improve their business and, more importantly, better serve their customers (end users). That’s where Elite Editing comes in: our job is to write these stories, which should act as road maps for potential CloudSky customers shopping for a tech solution, showing them exactly how other companies used CloudSky services and what they achieved. So the technical details are important: stories should mention the problem a company was having, then dive into the CloudSky services used to solve it, the process of implementing the CloudSky services, and the benefits realized—both for the company (e.g., cost savings, staff productivity) and the end users (e.g., products and services with better performance). CloudSky will provide us with some materials to help us do this, the main piece of which is a transcript of an interview with a data scientist or engineer from the CloudSky customer.

**CloudSky Service Descriptions**

**CloudSky Kangaroo**

CloudSky Kangaroo is a fully managed, scalable NoSQL database for running high-performance applications. Kangaroo offers security, continuous backups, in-memory caching, and has up to 99.99% availability. Customers can create applications supporting user-content caches that require connections for hundreds of thousands of users.

**CloudSky BabyWallabee**

CloudSky BabyWallabee (CloudSky BW) offers highly secure and scalable compute. CloudSky BW’s easy-to-use interface lets customers obtain and configure capacity with minimal difficulty. It gives customers control of their compute resources within the security of CloudSky’s compute platform.

**CloudSky Predictive Platypus**

CloudSky Predictive Platypus (CloudSky P2) is a big data platform that developers can use to run machine learning applications and distributed data processing jobs using open-source frameworks. Developers can use CloudSky P2 to conduct analyses using complex algorithms and predictive models to discover correlations and trends within data.

**CloudSky Wombat**

CloudSky Wombat is a CloudSky Technology Solutions storage service offering high performance and availability. Users can store large or small amounts of data for almost any use case. Wombat offers plenty of easy-to-use features to help businesses reduce costs and organize their data.

**CloudSky Koala**

CloudSky Koala lets developers load streams into warehouses and analytics services. Using Koala, users can capture and transform streaming data with only a few clicks. No recurring administration is necessary as Koala takes care of the provisioning for you.

**CloudSky Opossum**

CloudSky Opossum helps developers and data scientists in preparing and building, as well as training and deploying, machine learning models. Opossum brings together numerous highly useful capabilities that are built for machine learning.

**CloudSky Emu**

CloudSky Emu is a caching service that lets customers access data with millisecond latency. Using CloudSky Emu can help businesses accelerate application performance as well as database functionality.

**CloudSky Quokka**

CloudSky Quokka helps developers to train, build, and deploy all types of machine learning models, all using a low-code visual interface.

**CloudSky Style Sheet**

* **Use the *Chicago Manual of Style*, 17th edition (*CMoS*).** Follow *CMoS* for grammar, punctuation, capitalization, hyphenation, etc. Chapter 6 of *CMoS* discusses commas, and Chapter 8 covers capitalization. Check out *CMoS* 7.89 for hyphenation rules.
* **Use *Merriam-Webster* (*M-W*).** Refer to *M-W* ([www.m-w.com](http://www.m-w.com/)) for spelling, using only first-listed spelling variants (e.g., it’s *leaped*, not *leapt*). Also be sure to use spell check.
* **Use the following “CloudSky style”**  
  + **Numbers**
    - Use *CMoS*’s alternative rule for numerals (*CMoS* 9.3). That is, in general, spell out whole numbers zero through nine. Use numerals for 10 and up. (See also *CMoS* 9.4 and 9.7.)
    - Use numerals for all units of measurement, even time (e.g., *2 minutes*), as well as for all percentages. Use *percent*, not *%*, except in titles.
  + **Acronyms and abbreviations**
    - Terms that *M-W* classifies as abbreviations should be spelled out on first mention, followed by the acronym/abbreviation in parentheses.
    - If *M-W* classifies the term as a noun, it does not need to be spelled out.
    - If a term is used only 1–3 times, use the spelled-out term instead.
  + **Quotes**
    - Edit quotes as needed for grammar and “CloudSky” style.
    - Use the person’s full name on first mention in the case study body. On subsequent mentions, use surname only. Job titles should be lowercase.
  + **Global English**
    - Avoid words with ambiguous meanings. For example, use *as*, *since*, and *while* only in reference to time, not to mean *because* or *although*.
    - Include relative pronouns (e.g., *that*)—even when technically unnecessary—for ease of translation.
  + **Jargon (Do not use the words on the left: if you see these words, edit them!)**
    - Leverage → say *use*
    - Platform → say *solution*, *offering*
    - Utilize → say *use*
  + **Sensitive or problematic terms (Do not use the words on the left: if you see these words, edit them!)**
    - Enable → say *facilitate*, *help*, *let*, etc.
    - Execute → say *run*, *implement*, etc.
    - Ensure → say *help*, *provide*, *deliver*, etc.

**Sample Case Study: AngelDoc**

**AngelDoc Offers Online Health Services to Millions Using CloudSky**

Traditional healthcare involves patients journeying to hospitals, enduring extended wait times, and incurring extra costs such as transportation. AngelDoc, founded in 2012, aims to improve and simplify access to healthcare and patient services in New York. The company connects over 15 million monthly active users with 15,000 doctors and 800 certified partner pharmacies through its mobile and web application. The app makes it possible for its users across New York state to purchase medicine and facilitates telehealth appointments.

**Meeting the Need for Remote Healthcare**

AngelDoc launched on the CloudSky Technology Solutions cloud to build and manage its solution. “CloudSky is a vital part of our business,” says Abby Chandler, president of cloud infrastructure at AngelDoc. “We’ve adopted many managed services and have set up these services to scale on demand.”

Scaling and user experience were increasingly important to AngelDoc as the business grew. As the number of active users increased over the years alongside a growing demand for healthcare services in New York, AngelDoc experienced a spike in traffic to its app, leading to rapid increases in data processing and storage.

**Delivering Data Queries In Near-Real Time**

In 2021, AngelDoc migrated its data to CloudSky Wombat, a CloudSky Technology Solutions storage service offering high performance and availability, to establish a single source of truth across the organization. Using CloudSky Wombat helps AngelDoc’s data science team to retrieve data in real time to perform analytics and deliver queries in seconds.

“We faced a challenge in dealing with data,” Chandler says. “During our brainstorming meetings, CloudSky solutions experts explained the different technologies available and best practices for our business to follow.”

**Building Data Pipelines in 3 Hours**

With the data lake on CloudSky, creating data pipelines now takes 3 hours instead of 3 weeks. AngelDoc only requires 5 data engineers to build and manage the entire pipeline of data services. Using managed services on CloudSky, the business has reduced its storage costs by over 80 percent. “By using managed services from CloudSky to run our data lake, we don’t have to focus on infrastructure,” says Chandler. “This lets us focus entirely on building the data pipelines, resulting in substantial savings in both cost and time.”

Being able to use data is key to continually improving the AngelDoc user experience. For serverless data integration, AngelDoc uses CloudSky Koala, which lets developers load streams into warehouses and analytics services. As a result, the app is able to identify users’ health care provider preference for subsequent visits. This high level of personalization leads to higher satisfaction, repeat visits, and greater revenue potential for the company.

The insights gained from the analytics also help AngelDoc’s partners perform more efficiently. By analyzing data from pharmacy deliveries, AngelDoc is able to support its various logistic partners by locating the nearest delivery destinations in near-real time.

**Expanding the use of ML to Drive Efficiency**

In the future, AngelDoc plans to use CloudSky Predictive Platypus (CloudSky P2) is a big data platform that developers can use to run machine learning applications and distributed data processing jobs using open-source framework. Using Cloudsky P2 will help AngelDoc achieve further growth across New York. The company also plans to expand its use of ML to drive more efficiencies across its business.

“We’ve scaled from zero to several hundreds of thousands of users in a short period of time,” says Chandler. “Using CloudSky has made it possible for us to become more data-driven, helping us to provide a fabulous user experience.”

**Writing Test**



**Title (about 80 characters, including spaces):**

**Story Copy (400–500 words):**