CPT: Controllable & Editable Design Variations with Language Models

Karthik Suresh
Adobe
karsures@adobe.com

Amine Ben Khalifa*
Atta
amine.benkhalifa@gmail.com

Li Zhang
Adobe
zhangli@adobe.com

Wei-ting Hsu
Adobe
whsu@adobe.com

Fangzheng Wu
Adobe
fangzhengw@adobe.com

Vinay More
Adobe
vmore@adobe.com

Asim Kadav
Adobe
akadav@adobe.com

Abstract

Designing visually diverse and high-quality designs remains a manual, time-consuming process, limiting scalability and personalization in creative workflows. We present a system for generating editable design variations using a decoder-only language model – the Creative Pre-trained Transformer (CPT) – trained to predict visual style attributes in design templates (Figure 1). At the core of our approach is a new representation called Creative Markup Language (CML), a compact, machine-learning–friendly format that captures canvas-level structure, page layout, and element-level details (text, images, and vector graphics), including both content and style. We fine-tune CPT on a large corpus of design templates authored by professional designers, enabling it to learn meaningful, context-aware predictions for attributes such as color schemes and font choices. The model produces semantically structured and stylistically coherent outputs, preserving internal consistency across elements. Unlike generative image models, our system yields fully editable design documents rather than pixel-only images, allowing users to iterate, personalize within a design editor. In experiments, our approach generates contextual color and font variations for existing templates and shows promise in adjusting layouts, all while maintaining design principles.

1 Introduction

Creating visually appealing content that aligns with brand guidelines is essential for creators, marketers, and businesses. However, producing content at scale remains manual and slow. Typically, expert designers are required to create and adapt templates for different contexts or audiences. As demand for high-quality personalized content grows across platforms, this manual workflow becomes a bottleneck that limits both creativity and productivity. Recent advances in generative AI have introduced powerful multimodal models [Achiam et al., 2023, OpenAI, 2024, Team et al., 2023] and tools capable of producing impressive visual artifacts [Lin et al., 2023, Gao et al., 2023], yet many of these systems operate in unstructured domains (e.g., image generation) or require extensive prompt engineering without offering editability or control over specific visual attributes. Other tools provide semi-automated recommendations for fonts [Zhao et al., 2018, Jiang et al., 2019] or context-aware asset search [Kovacs et al., 2018], but these require manual application and often lack contextual awareness or stylistic coherence throughout designs.

In this work, we propose a new perspective on the design variation problem: treating it as a structured sequence prediction task over a semantically rich representation of design documents. We introduce

*Work done while at Adobe.

39th Conference on Neural Information Processing Systems (NeurIPS 2025) Workshop: The First Workshop on Generative and Protective AI for Content Creation.

Figure 1: Our CPT model uses the context of the original template (far left) to generate font and color variations.

the Creative Pre-trained Transformer (CPT), a decoder-only language model fine-tuned to predict masked style attributes within editable design templates. Central to this approach is the Creative Markup Language (CML), a compact machine learning-friendly representation that encodes the structure, layout, content and styling details at the canvas level for text, images, and vector graphics, inspired by recent work on multimodal document understanding [Ye et al., 2023, Kikuchi et al., 2024].

By leveraging powerful large language models (LLMs) fine-tuned on designer-authored content, our approach incorporates rich world knowledge about color, typography, and visual design, allowing the generation of context-aware and stylistically sophisticated variations compared to traditional recommendation-based methods. Unlike traditional generation methods that produce static images, our system produces fully editable design documents that can be rendered, edited, and exported within a commercial design editor.

Our contributions are as follows:

- We propose a controllable and editable design-variation framework (CPT+CML): users choose which attributes to vary (color, font, layout), CPT predicts only those fields under optional brand constraints, and the resulting documents remain fully editable—unlike traditional LLM/diffusion systems that produce uncontrolled, non-editable outputs.
- We develop a complete engineering pipeline for converting raw production design documents to and from CML, enabling seamless integration with production design tools and editable output.
- We design a heuristics-based evaluation pipeline complemented by a GPT-powered filter and design scorer, which ensures generated variations meet aesthetic and usability standards.
- We demonstrate the effectiveness of our approach on generating stylistically coherent color and font variations for templates from a commercial design editor, and show preliminary results for layout variation on simple templates.

2 Background and Related Work

Early systems for automatic graphic design relied on hard–coded rules or template retrieval, limiting flexibility and scale [O'Donovan et al., 2014, Kovacs et al., 2018]. Recent work treats layout generation as a learning problem. **GAN-based** models such as LayoutGAN refine randomly initialised element boxes using a wire-frame discriminator [Li et al., 2019]. **Latent-variable** approaches (LayoutVAE) capture the distribution of scene layouts via a VAE, while **autoregressive transformers** (LayoutTransformer, VTN) model element sequences directly. Diffusion variants (LayoutDM) further boost diversity and realism through iterative refinement.

Most of these methods output abstract bounding boxes and raster previews, leaving detailed styling (color, typography) and editability to the user. To address full-fidelity outputs, CreatiPoster generates multi-layer posters by combining an LLM for structured JSON with a diffusion background model [Zhang et al., 2025]. Closer to our goal, Shimoda *et al.* propose a transformer that predicts coherent font-and-color assignments for placeholder text, but it omits images and layout metadata [Shimoda et al., 2024]. Context-aware recommenders handle specific style subtasks—e.g. font pairing [Zhao et al., 2018, Jiang et al., 2019] or harmonisation—but treat each element independently and do not scale to full-template variation.

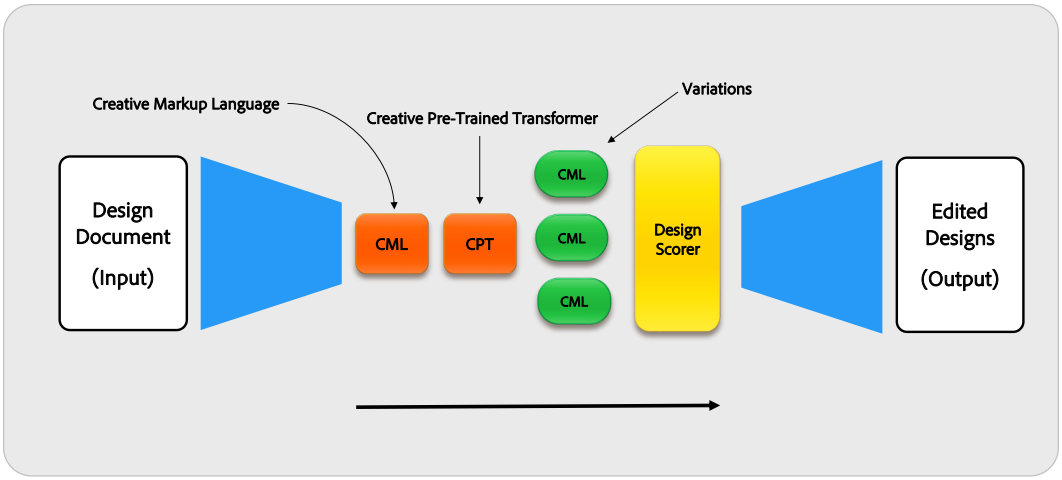

Figure 2: High-Level Overview of the Design Variations Pipeline

Prior systems differ in output fidelity and editability. LayoutDM and related diffusion/transformer models generate only abstract boxes or raster previews, offering limited control [Horita et al., 2024]. AutoPoster produces content-aware posters but as static images without editability [Lin et al., 2023]. CreatiPoster improves fidelity with layered JSON plus a diffusion background, yet control remains partial (layers only) [Zhang et al., 2025]. In contrast, CPT outputs fully structured, editable CML documents with fine-grained control (color, font, layout, brand), unifying editability and controllability in a single framework.

Our work unifies these threads: we cast template variation as a *masked sequence prediction* task over a compact, editable markup (CML) and fine-tune a 7B decoder-only LLM (CPT) to jointly predict color, font and layout attributes. Unlike prior image-centric generators, CPT outputs a fully structured document that designers can open and tweak directly, bridging large-language-model reasoning with professional creative tools.

3 Design

Our system consists of three main components: (1) a design representation (CML) that converts graphical templates into a linear token sequence suitable for a language model, and (2) the Creative Pre-trained Transformer (CPT) model, a decoder-only transformer that is fine-tuned on the CML sequences to output design variations, and (3) a design scorer to filter designs that do not meet aesthetic standards. We also outline our training procedure for CPT, which involves masking certain attributes in the CML and letting the model predict them, and describe how we generate new variations at inference time. An overview of the pipeline is illustrated in Figure 2.

3.1 Creative Markup Language (CML)

CML (Creative Markup Language) is a domain-specific, compact, and semantically rich representation for editable design templates. It linearizes design documents into sequences of tokens while preserving their hierarchical structure. A CML document begins with global definitions (e.g., canvas size, background), followed by structured element blocks such as `<text>`, `<image>`, or `<shape>`. Each block encodes both **content** (e.g., text string, image reference) and **style/layout attributes** (e.g., font, size, color, bounds, z-index). By masking attributes such as color, font, or bounds, CML provides controllable slots for CPT to predict stylistically coherent alternatives.

Compared to raw JSON or SVG, CML offers several advantages:

- **Machine-learning–friendly:** standardized tokens and normalized values (e.g., hex colors, canonical font names) allow efficient training.
- **Context-aware:** the order and grouping of tokens preserve structural relationships, enabling models to infer design conventions (e.g., titles and subtitles sharing a style).

- **Editable:** round-trips seamlessly back into production design tools, ensuring generated outputs remain fully manipulable.
- **High-fidelity:** provides a near-exact reverse mapping to renderable documents, preserving layout, styling, and content integrity.

Crucially, CML supports attribute masking (e.g., replacing color codes or font families with `<MASK_COLOR>` or `<MASK_FONT>`) to guide CPT in predicting stylistically coherent variations.

```
<text id="3">
  <bounds top="239" left="210" width="600" height="385"/>
  <style alignment="center"/>
  <p>
    <content>Pizza</content>
    <style color="<MASK_COLOR_0>" font="<MASK_FONT_0>" size="<MASK_FONTSIZE_0>">
  </p>
</text>
```

This masking strategy ensures predictions remain contextualized and coherent, while making CML a natural substrate for causal LLM training.

3.2 Creative Pre-trained Transformer (CPT)

CPT is a transformer-based language model that we fine-tuned for design generation. We start from a pre-trained decoder-only LLM with 7B parameters (Mistral-7B [Jiang et al., 2023]), chosen for its strong contextual reasoning and ability to generate well-structured text such as code or JSON [Wei et al., 2022]. Since CML is designed as a flat, non-nested XML representation, it avoids the hierarchical complexity that commonly causes parsing errors in LLM-generated JSON or SVG. Each element block (e.g., <text>, <image>, <shape>) maps directly to a design entity, keeping syntax shallow yet expressive. Fine-tuning CPT on over 220K professionally authored CML templates enables it to learn this grammar and structure, producing syntactically valid XML in over 99% of validation cases verified by schema checks. This flat schema and large-scale finetuning together yield reliably well-formed and semantically coherent design documents.

Fine-tuning Objective. CPT is fine-tuned on 220K professionally designed templates converted to CML using LoRA [Hu et al., 2021] for parameter-efficient adaptation. The task is formulated as masked sequence prediction: selected attributes are replaced with placeholder tokens, and the model autoregressively predicts only the masked values. These predictions are then *infilled* back into the original CML at the correct positions, preserving the rest of the canvas exactly. This masking-based infilling resembles the "Fill-in-the-Middle" (FIM) paradigm [Bavarian et al., 2022], but differs in that our approach uses *pre-defined, task-specific masks* (e.g., colors, fonts, layout) within structured XML (CML). The model outputs only the masked tokens, ensuring contextualized yet controlled design variations rather than free-form text completion. To explore different levels of consistency and diversity, we trained three variants of CPT:

- **CPT No Association:** every masked token is independent, giving the model maximal freedom but often yielding inconsistent styles.
- **CPT Local Association:** masks within the same element (e.g., all attributes in a `<text>` block such as color, font, size) share a mask ID, enforcing local consistency.
- **CPT Global Association:** attributes that match across elements share the same mask ID, enforcing global coherence (e.g., multiple boxes sharing the same color). This reduces diversity slightly but improves alignment, contrast, and design consistency—ensuring that identical attributes in the original remain identical in all variations. Such consistency is especially valuable in structured visuals like infographics or figures with legends, where these associations carry semantic meaning. (see Table 1).

This approach is akin to sequence "inpainting": the model sees a mostly complete XML canvas, outputs only the missing attributes, and these are seamlessly infilled into the masked positions, yielding a fully valid and renderable design document. Temperature further provides a controllable knob on creativity: lower values bias the model toward safer, brand-consistent predictions, while higher values encourage more diverse and exploratory style variations.

For illustration, consider the earlier <text> element. After masking font and color attributes, the model produces predictions that are then infilled back into the CML:

```
<text id="3">
  <bounds top="239" left="210" width="600" height="385"/>
  <style alignment="center"/>
  <p>
    <content>Pizza</content>
    <style color="<MASK_COLOR_0>" font="<MASK_FONT_0>" size="MASK_FONTSIZE_0"/>
  </p>
</text>

### OUTPUT:
MASK_COLOR_0: #fefcf0
MASK_FONT_0: MeowScript-Regular
MASK_FONTSIZE_0: 279
```

These predicted values are inserted back into the masked positions, yielding a fully valid and renderable CML document. For brevity, only a fragment of the full CML is shown here; a complete masked-to-infilled example is provided in the Appendix.

All models were fine-tuned for 3 epochs on hundreds of thousands of professionally designed templates from an online graphic design platform, covering diverse use cases (social media posts, flyers, ads, invitations, etc.). Each template was converted to CML, and multiple masked variants were generated to expose the model to a wide range of scenarios (colors only, fonts only, or combinations including layout attributes).

Training used a standard autoregressive loss with AdamW optimizer [Loshchilov and Hutter, 2017] and GELU activations [Hendrycks and Gimpel, 2016]. Mask placeholders (e.g., <MASK_COLOR>) remain in the input, and CPT learns to output only the missing values, which are then infilled into the original CML sequence. This allows the model to generate context-aware, stylistically coherent predictions while remaining faithful to the XML structure of CML.

3.3 Variation Generation and Design Scorer

At inference time, variations are generated by masking selected attributes in the input CML and letting CPT predict replacements, which are then infilled to produce a complete, editable design. Users can mask *color, font, layout, or any combination of these*, giving direct control over which aspects of the design are modified:

- **Color variations:** color attributes (background color, text color, effect color, etc.) are masked and repainted by CPT, typically yielding coherent palettes rather than arbitrary colors.
- **Font variations:** font attributes (family, size, leading, tracking, etc.) are masked, allowing CPT to propose new typographic styles while preserving layout and content.
- **Layout variations:** positional attributes (e.g., coordinates, sizes) can be masked to explore alternative arrangements. This is a more challenging problem, but CPT can still maintain balance in simple cases, suggesting it has learned basic spatial relationships [Lee et al., 2020].
- **Effect, brand-aware, and other variations:** the same masking strategy naturally extends beyond color, font, and layout. For instance, effect attributes (e.g., duotone, colorize, tint) can be masked to explore stylistic treatments, brand-aware variations can be guided by constraints in a dedicated <brand> section of the CML, and additional attributes can be incorporated as needed—making this a general, extensible framework for creative control.

  See Appendix A.2A.3 A.4 for qualitative illustrations of variation types.

The CML $\rightarrow$ design conversion is deterministic and implemented via a diff-and-apply pipeline. We first compute a structured diff between the original and infilled CML, translate these changes into atomic document-edit operations (e.g., style, font, or layout updates) bound to stable element IDs, and apply them through the platform's rendering APIs to produce updated renditions. Because each edit is schema-validated and uniquely targeted, the result is visually and structurally identical to

the baseline and remains fully editable. When no base document exists, we reconstruct the design deterministically by reversing the CML mapping to restore elements, geometry, styles, and asset references—ensuring a consistent baseline for subsequent diff-and-apply updates.

To guarantee quality, we employ a Design Scorer that leverages GPT-4o [OpenAI, 2024] to filter and rank rendered variations, assessing both usability and aesthetic soundness.

3.3.1 Filtering and Failure Detection

Despite strong performance, CPT can still produce occasional failures due to the inherent difficulty of modeling aesthetics, the limitations of text-only representations, and imperfections in data distribution. To mitigate these issues, we designed a system that automatically detects common failure modes before presenting the generated design variations to end users.

The filtering component operates on two levels: CML-based and rendition-based. This two-stage process enables early rejection of low-quality generations before costly rendering, while still ensuring rigorous post-render checks. The component is also customized to detect failures according to different generation modes. For conciseness, the following discussion focuses on the generation mode for color and font variations.

CML-based filtering. In the case of color and font variations, generations that lack sufficient diversity—such as those with colors or fonts too similar to either the original design or to other generated variations—are discarded.

Rendition-based filtering. This stage operates on rendered images of both the original template and the generated variations. Leveraging GPT-4o's multimodal capabilities, the system evaluates design quality using two targeted metrics specific to color and font variations.

- **Color Contrast Filtering:** Both the original and modified design images are input to GPT-4o with a specialized color contrast prompt. The model evaluates whether the color variations have introduced readability issues such as unreadable, faded, or missing text and design elements due to poor contrast. Each variation is categorized as either *pass* (no visibility issues detected) or *fail* (important elements are missing or unreadable). Only variations receiving a *pass* classification are retained for further consideration.
- **Alignment Filtering:** Using the original and modified images, GPT-4o checks for alignment issues introduced by the variation process with a dedicated alignment prompt. Importantly, the model flags only newly introduced alignment problems, not pre-existing flaws in the original template. Results are classified as *pass* (no usability or layout issues) or *fail* (obvious misalignment affecting usability or aesthetics). Variations that fail this stage are removed from the candidate set.

3.3.2 Aesthetic Ranking and Diversity Maximization

After filtering, the remaining variations for each template undergo a ranking process designed to balance aesthetics and diversity. Using GPT-4o with a diversity-emphasized prompt, variations are ranked to ensure the final set collectively presents a broad range of color palettes and stylistic approaches while maintaining quality standards.

This scorer ensures both quality assurance and diversity optimization: users receive variations that are technically sound while also offering stylistically meaningful alternatives. These results demonstrate that in-context learning with GPT-4 can reliably function as a design scorer [Haraguchi et al., 2024].

4 Results

4.1 Evaluation Methodology

To assess the quality of CPT's generated variations, we use a *three-pronged evaluation framework* that balances automated analysis with human-centered judgments: **Automated Heuristic Metrics**—quantitative checks used for checkpoint selection; **Human Evaluation**—thumbs-up/down ratings with comments; and **Qualitative Golden-Set Analysis**—manual inspection of challenging templates.

4.1.1 Automated Heuristic Metrics and Results

We designed a suite of heuristic-based metrics that capture critical design principles such as spatial layout integrity, text readability, and visual contrast. While useful for guiding model selection, these metrics cannot fully capture aesthetic quality. All metrics are reported as *chosen rates* (percentage of designs meeting a minimum quality threshold), following [Inoue et al., 2024, Horita et al., 2024].

- **General Overlap**: Measures the percentage of designs where elements do not inappropriately overlap, preserving visual clarity and structure.
- **Text Overflow**: Evaluates whether text fits within its bounding box without clipping horizontally or vertically.
- **Text Over Boundary**: Assesses the percentage of text elements within canvas boundaries.
- **Text Line Overlap**: Measures whether text lines interfere with other elements.
- **Color Contrast**: Evaluates if text has sufficient contrast against the background for readability.
- **Overall Chosen Rate**: A composite metric that requires passing all quality checks, serving as a conservative indicator of immediately usable designs.

Table 1 presents quantitative results of this evaluation. All reported results correspond to the CPT model trained for 3 epochs and evaluated at temperature 0.8, selected as the checkpoint with the lowest validation loss and strongest performance on heuristic metrics.

Table 1: CPT model performance on heuristic metrics (Chosen Rate %). Best non-human values in **bold**.

Model	Overall Chosen Rate	General Overlap	Text Overflow	Text Over Boundary	Text Line Overlap	Color Contrast
Human Templates (Gold Standard)	80.7	97.8	93.5	99.7	98.8	94.6
CPT Global Association	**58.3**	**93.5**	63.7	**99.6**	**98.1**	**41.3**
CPT Local Association	50.9	93.3	61.3	**99.6**	98.0	35.4
CPT No Association	35.0	92.5	**66.2**	99.3	96.5	26.3

4.1.2 Human Evaluation

To complement the automated metrics, we conducted a human evaluation with professional designers and engineers. In total, we collected **2974** ratings on variations filtered and ranked by the Design Scorer: **2698** thumbs up (**90.7%**) and **276** thumbs down (**9.3%**).

We also collected **107** free-form comments explaining thumbs-down ratings. Figure 3 summarizes the distribution of these comments, with the most frequent concerns being insufficient color contrast **(50%)**, misalignment issues **(30.8%)**, and lack of stylistic diversity **(18.7%)**.

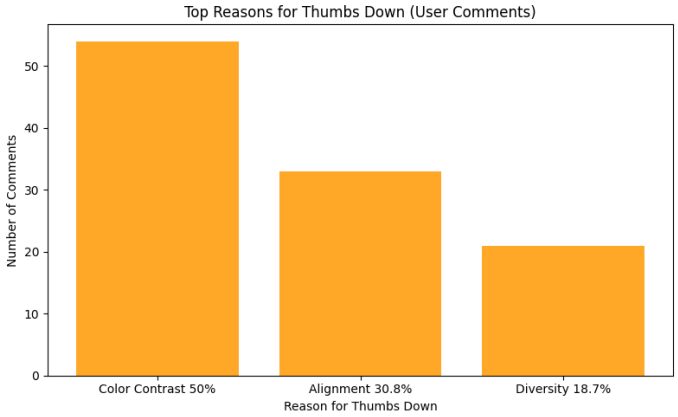

Figure 3: Human evaluation results

Validation of the Design Scorer. We validated the Design Scorer, which contains GPT-based filtering on color contrast and alignment, by corroborating its pass/fail judgments on those by human ratings (see Table 2); detailed prompt comparisons are in Appendix A.1

Table 2: Confusion matrix comparing GPT judgments (Pass vs. Fail; with Borderline treated as Pass) to human thumbs (Up vs. Down) for the best-performing prompt. Cells show *count* (row%).

	GPT: Fail	**GPT: Pass**
Human: Down	561 (18.9%)	665 (22.4%)
Human: Up	201 (6.8%)	1547 (52.0%)

4.1.3 Qualitative Evaluation on a Golden Set

Finally, we conduct qualitative comparisons using a curated "golden set" of challenging designs. These include edge cases such as dense layouts, extreme aspect ratios, and complex typography. We also include cases requiring world knowledge and design conventions, such as seasonal color themes (e.g., orange–purple for Halloween, green–red for Christmas) and domain-specific typography choices (e.g., playful fonts for children's events, serif fonts for formal announcements).

For each case, we manually compare CPT's outputs against human-designed references, enabling detailed analysis of strengths, weaknesses, and failure modes that aggregate scores cannot capture.

4.2 Analysis and Key Findings

- **Human upper bound:** Human templates achieve 80.7% chosen rate, setting the maximum benchmark for this evaluation.
- **Association matters:** Association-based models strongly outperform the no-association baseline (58.3% vs. 35.0%).
- **Color contrast:** Remains the largest gap (41.3% vs. 94.6%), though GPT-based filtering boosts satisfaction to 90.7%.

These results show that while CPT faces challenges in color contrast and certain aspects of typography, it reliably preserves structure and consistency. Automated metrics alone can underestimate aesthetic quality, but when combined with human evaluation and qualitative golden-set inspection, they provide a more complete picture of CPT's strengths and gaps. The Design Scorer further ensures that surfaced variations are stylistically diverse, structurally sound, and well-received—competitive with recent multimodal design generation systems [Cheng et al., 2025, Lin et al., 2025].

5 Conclusions

We presented CPT, a novel approach for generating editable design variations using fine-tuned language models and a structured design representation. Our contributions include the Creative Markup Language (CML), which enables compact and semantically rich encoding of design templates, and the Creative Pre-trained Transformer (CPT), which predicts contextually appropriate style attributes through masked sequence prediction. In addition, we introduce a filtering and ranking pipeline powered by GPT-4o that detects contrast and alignment issues, ranks variations for aesthetics and diversity, and ensures only high-quality outputs reach users.

We also design a comprehensive evaluation framework that combines automated heuristics, GPT-based assessments, and human judgments, offering a reliable picture of both functional and aesthetic quality. By producing fully editable outputs rather than static images, CPT addresses a key limitation of existing generative design tools. Fine-tuned on professionally designed templates, the system generates coherent color and font variations while maintaining design consistency, demonstrating how LLM-driven approaches can balance creativity with reliability in design generation.

Our work opens new possibilities for AI-assisted design tools that preserve creative control while accelerating content production, potentially transforming how designers approach template variation and personalization at scale.

5.1 Future Work

We see three directions. (1) *Multimodal + scale:* integrate visual signals from embedded photos and design assets and train at larger scale, so CPT can reason about text–background interactions and layering, mitigating color-contrast and alignment issues inherent to text-only CML. (2) *Richer CML + controllable inference:* extend CML with constraints and design relationships (e.g., component groupings) to enforce consistency, and expose these constraints at inference—incorporating user preferences and brand guidelines for personalized, brand-consistent outputs. (3) *Better evaluation + training loops:* develop metrics and stronger template-to-CML conversion to detect failures (color contrast, alignment, overlap) more accurately in the Design Scorer and during training, enabling reinforcement-style training to further improve quality.

Acknowledgements

We would like to thank the members of Adobe Enterprise Search engineering team including Rahul Gandhi, Vishwesh Nayak, and Nandaja Ananthanarayanan for building data and rendering pipelines. We also would like thank Tracy King, Gaurav Kukal and Vipul Dalal for their support and encouragement of this project.

References

J. Achiam, S. Adler, S. Agarwal, L. Ahmad, I. Akkaya, F. L. Aleman, D. Almeida, J. Altenschmidt, S. Altman, S. Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

M. Bavarian, H. Jun, N. Tezak, J. Schulman, C. McLeavey, J. Tworek, and M. Chen. Efficient training of language models to fill in the middle. *arXiv preprint arXiv:2207.14255*, 2022.

Y. Cheng, Z. Zhang, M. Yang, H. Nie, C. Li, X. Wu, and J. Shao. Graphic design with large multimodal model. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 2473–2481, 2025.

Y. Gao, J. Lin, M. Zhou, C. Liu, H. Xie, T. Ge, and Y. Jiang. Textpainter: Multimodal text image generation with visual-harmony and text-comprehension for poster design. In *Proceedings of the 31st ACM International Conference on Multimedia*, pages 7236–7246, 2023.

D. Haraguchi, N. Inoue, W. Shimoda, H. Mitani, S. Uchida, and K. Yamaguchi. Can gpts evaluate graphic design based on design principles? In *SIGGRAPH Asia 2024 Technical Communications*, pages 1–4. 2024.

D. Hendrycks and K. Gimpel. Gaussian error linear units (gelus). *arXiv preprint arXiv:1606.08415*, 2016.

D. Horita, N. Inoue, K. Kikuchi, K. Yamaguchi, and K. Aizawa. Retrieval-augmented layout transformer for content-aware layout generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 67–76, 2024.

E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, and W. Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021.

N. Inoue, K. Masui, W. Shimoda, and K. Yamaguchi. Opencole: Towards reproducible automatic graphic design generation. *arXiv preprint arXiv:2406.08232*, 2024.

A. Q. Jiang, A. Sablayrolles, A. Mensch, C. Bamford, D. S. Chaplot, D. de las Casas, F. Bressand, G. Lengyel, G. Lample, L. Saulnier, L. R. Lavaud, M.-A. Lachaux, P. Stock, T. Le Scao, T. Lavril, T. Wang, T. Lacroix, and W. El Sayed. Mistral 7b. *arXiv preprint arXiv:2310.06825*, 2023.

S. Jiang, Z. Wang, A. Hertzmann, H. Jin, and Y. Fu. Visual font pairing. *IEEE Transactions on Multimedia*, 22 (8):2086–2097, 2019.

K. Kikuchi, N. Inoue, M. Otani, E. Simo-Serra, and K. Yamaguchi. Multimodal markup document models for graphic design completion. *arXiv preprint arXiv:2409.19051*, 2024.

B. Kovacs, P. O'Donovan, K. Bala, and A. Hertzmann. Context-aware asset search for graphic design. *IEEE transactions on visualization and computer graphics*, 25(7):2419–2429, 2018.

H.-Y. Lee, L. Jiang, I. Essa, P. B. Le, H. Gong, M.-H. Yang, and W. Yang. Neural design network: Graphic layout generation with constraints. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part III 16*, pages 491–506. Springer, 2020.

J. Li, J. Yang, A. Hertzmann, J. Zhang, and T. Xu. LayoutGAN: Generating graphic layouts with wireframe discriminators. *CoRR*, abs/1901.06767, 2019. URL `https://arxiv.org/abs/1901.06767`. ICLR 2019.

J. Lin, M. Zhou, Y. Ma, Y. Gao, C. Fei, Y. Chen, Z. Yu, and T. Ge. AutoPoster: A highly automatic and content-aware design system for advertising poster generation. In *Proceedings of the 31st ACM International Conference on Multimedia*, 2023. doi: 10.1145/3581783.3611930. URL `https://arxiv.org/abs/2308.01095`.

J. Lin, S. Sun, D. Huang, T. Liu, J. Li, and J. Bian. From elements to design: A layered approach for automatic graphic design composition. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 8128–8137, 2025.

I. Loshchilov and F. Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

P. O'Donovan, A. Agarwala, and A. Hertzmann. Learning layouts for single-page graphic designs. *IEEE Transactions on Visualization and Computer Graphics*, 20(8):1200–1213, 2014.

OpenAI. Hello gpt-4o, 2024. URL `https://openai.com/index/hello-gpt4o/`. Accessed 04-11-2024.

W. Shimoda, D. Haraguchi, S. Uchida, and K. Yamaguchi. Towards diverse and consistent typography generation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 7296–7305, 2024.

G. Team, R. Anil, S. Borgeaud, J.-B. Alayrac, J. Yu, R. Soricut, J. Schalkwyk, A. M. Dai, A. Hauth, K. Millican, et al. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023.

J. Wei, X. Wang, D. Schuurmans, M. Bosma, F. Xia, E. Chi, Q. V. Le, D. Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35: 24824–24837, 2022.

J. Ye, A. Hu, H. Xu, Q. Ye, M. Yan, Y. Dan, C. Zhao, G. Xu, C. Li, J. Tian, et al. mplug-docowl: Modularized multimodal large language model for document understanding. *arXiv preprint arXiv:2307.02499*, 2023.

Z. Zhang, Y. Cheng, D. Hong, M. Yang, G. Shi, L. Ma, H. Zhang, J. Shao, and X. Wu. Creatiposter: Towards editable and controllable multi-layer graphic design generation. *arXiv preprint arXiv:2506.10890*, 2025.

N. Zhao, B. Liu, and X. Tang. Context-aware font recommendation for web design. In *ACM UIST*, 2018.

A Appendix

A.1 GPT Evaluation Prompts and Results

To assess reliability of the GPT-based filtering pipeline, we iterated over multiple prompt formulations. Below, we report detailed results for `prompt_v1` and the improved `prompt_v2`, evaluated against human thumbs up/down judgments for color contrast and alignment combined. Borderline cases are considered as passes.

Table 3: Confusion matrices comparing GPT judgments (Pass vs. Fail; Borderline = Pass) to human thumbs for `prompt_v1` and `prompt_v2`. Cells show *count* (row%).

Prompt_v1

	GPT: Fail	**GPT: Pass**
Human: Down	686 (23.1%)	1350 (45.4%)
Human: Up	76 (2.5%)	862 (29.0%)

Prompt_v2

	GPT: Fail	**GPT: Pass**
Human: Down	561 (18.9%)	665 (22.4%)
Human: Up	201 (6.8%)	1547 (52.0%)

Prompt_v1

Color contrast: "You will be provided with an original template (first image) and a modified template (second image) with changes in color for certain components. Determine if the second template has any missing or unreadable text/elements due to low contrast. Pay attention to all texts and object regardless of their size, maintaining a very high threshold for visibility. What you can do it try to recognize all texts in the first image, then do the same for the second image, independently. Compare if all extracted texts match one another. If any text/object is missing in the second image, response 'yes'. If ALL the design elements are readable, flag them as 'no.' If there are issues, specify which parts of the design lack sufficient contrast."

Alignment: "You will be provided with an original template (first image) and a modified template (second image) with changes in color and fonts. Determine if the second template has any text placement different from the original, resulting in misalignment with other objects and reduced layout harmony. Examine closely at each text, pay attention to any overlapping of texts and objects, off-center placement of texts, text extending beyond canvas or containers, text too small and not taking enough space in its designated containers, broken alignment with respect to other texts in the design, reduced readability due to text misplacement. Only respond 'yes' if any behavior listed above is found. If none of the above issues are found flag them as 'no.' If there are issues, specify which parts of the design has text misalignment issues."

Color contrast. "You will be provided with two images: the original_design_template and a modified_design_template with changes in color, font & font size. Your task is to determine whether the modified_design_template has any missing, unreadable, or significantly faded text or design elements due to very poor color contrast. Focus on real-world readability: flag only if the contrast in the modified_design_template makes the text significantly harder to perceive compared to the original. Minor reductions in clarity that do not affect readability should not be flagged. The original_design_template is provided for context to understand existing visibility levels. We are interested in knowing if the modified_design_template introduced new visibility issues that could impact user experience. Ignore spelling corrections or differences — evaluate visibility only. Even small text or visually distinct elements such as headers, badges, banners, or call-to-action buttons must be clearly readable against their background. Do not assume visibility based on typical layout or expected content. Text that remains clearly visible—even with bold, unconventional, or stylistic color choices—should be considered acceptable. Categorize your response using one of the following three buckets: 1. 'clear_pass' — all elements are present and fully readable with no visibility concerns; 2. 'borderline' — some text or elements are slightly harder to perceive but are readable and might still be acceptable; 3. 'clear_fail' — important text or elements are missing, unreadable, or significantly harder to perceive. Return your answer using this format: '<bucket>: <short explanation>'. Example1: clear_fail: The red text in the footer is unreadable against the dark background. Example2: borderline: text abc is slightly harder to read against the light background,

but it might still be acceptable. Example3: clear_pass: All elements are clearly visible and readable. **Do not evaluate for alignment or minor layout adjustments or pre-existing flaws in the original design.**"

Alignment. "You will be provided with two images: the original_design_template and a modified_design_template with changes in color, font, font size or minor layout adjustments. Your task is to determine whether the modified_design_template introduces **any new alignment issues** that negatively impact layout quality, visual balance or user perception. **Ignore the color contrast issues and spelling mistakes**. **Focus strictly on new alignment problems introduced in the modified_design_template which were not present in the original_design_template**, such as: Text or elements overlapping with each other; Text misaligned enough to appear visually far disconnected from related elements; Text or objects extending outside canvas area; Words split awkwardly across lines (e.g., a single word broken between lines). Categorize your response using **one of these buckets**: 1. 'clear_pass' — Layout remains visually coherent with no usability concerns 2. 'borderline' — Some misalignments are present but may still be acceptable 3. 'clear_fail' — Clear misalignment issues that disrupt readability or balance. **Respond using this exact format**: '<bucket>: <short explanation>'. **Examples:** Example 1: clear_fail: Footer text is misaligned and overlaps with page number. Example 2: borderline: Sidebar content is slightly shifted but still understandable. Example 3: clear_pass: All design elements are properly aligned and balanced. **Do not evaluate color, text contrast, spelling, or pre-existing flaws in the original design.**"

Analysis Compared to `prompt_v1`, `prompt_v2` introduced:

- Clearer instructions to *ignore non-relevant issues* (e.g., spelling, color when evaluating alignment).
- A structured triage scheme (`clear_pass`, `borderline`, `clear_fail`) instead of binary labels.
- Emphasis on real-world readability and user perception, rather than absolute pixel-level differences.

This change reduced false positives substantially (from 45.4% to 22.4%) and improved true positives (from 29.0% to 52.0%), at the cost of slightly higher false negatives. The result is a more reliable and user-aligned filter, ensuring that genuinely usable designs are surfaced while only discarding those with meaningful quality issues.

A.2 Qualitative Examples of Color and Font Variations

Figure 4: Our CPT model generates stylistic variations (right) from an original design (left). Each row shows a different example with either font or color variation. Notably, Example 2 illustrates the use of world knowledge to select a Halloween-inspired color palette, while Example 3 demonstrates context-aware typography, where playful fonts are chosen to match the event theme.

A.3 Examples for Layout Variations

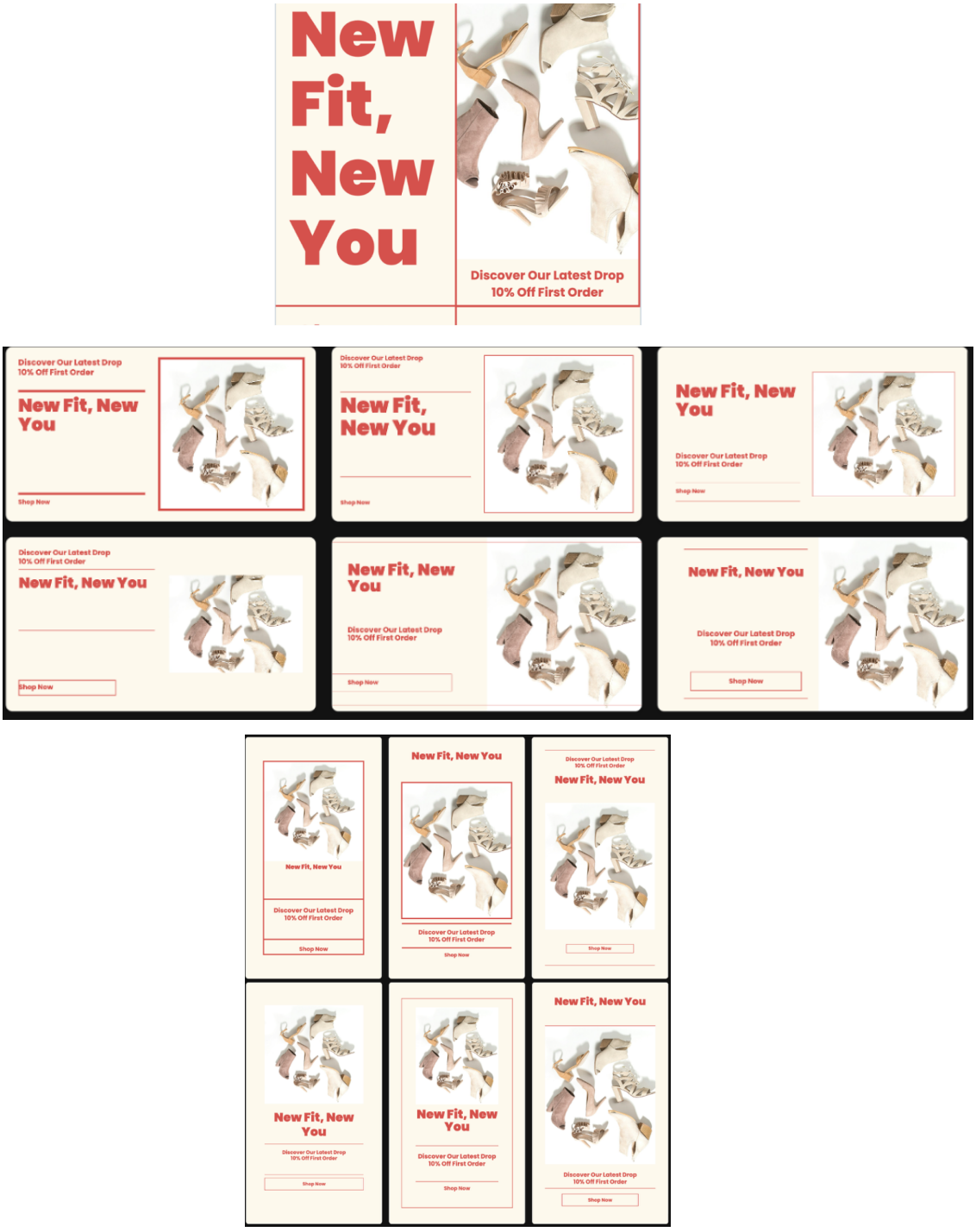

Figure 5: Examples of layout variations generated from a single template: the original square format (1:1) in the first row, adapted to a YouTube thumbnail in the second row, and to an Instagram Story in the third row.

A.4 Examples for Brand-Aware Color-Font Variations

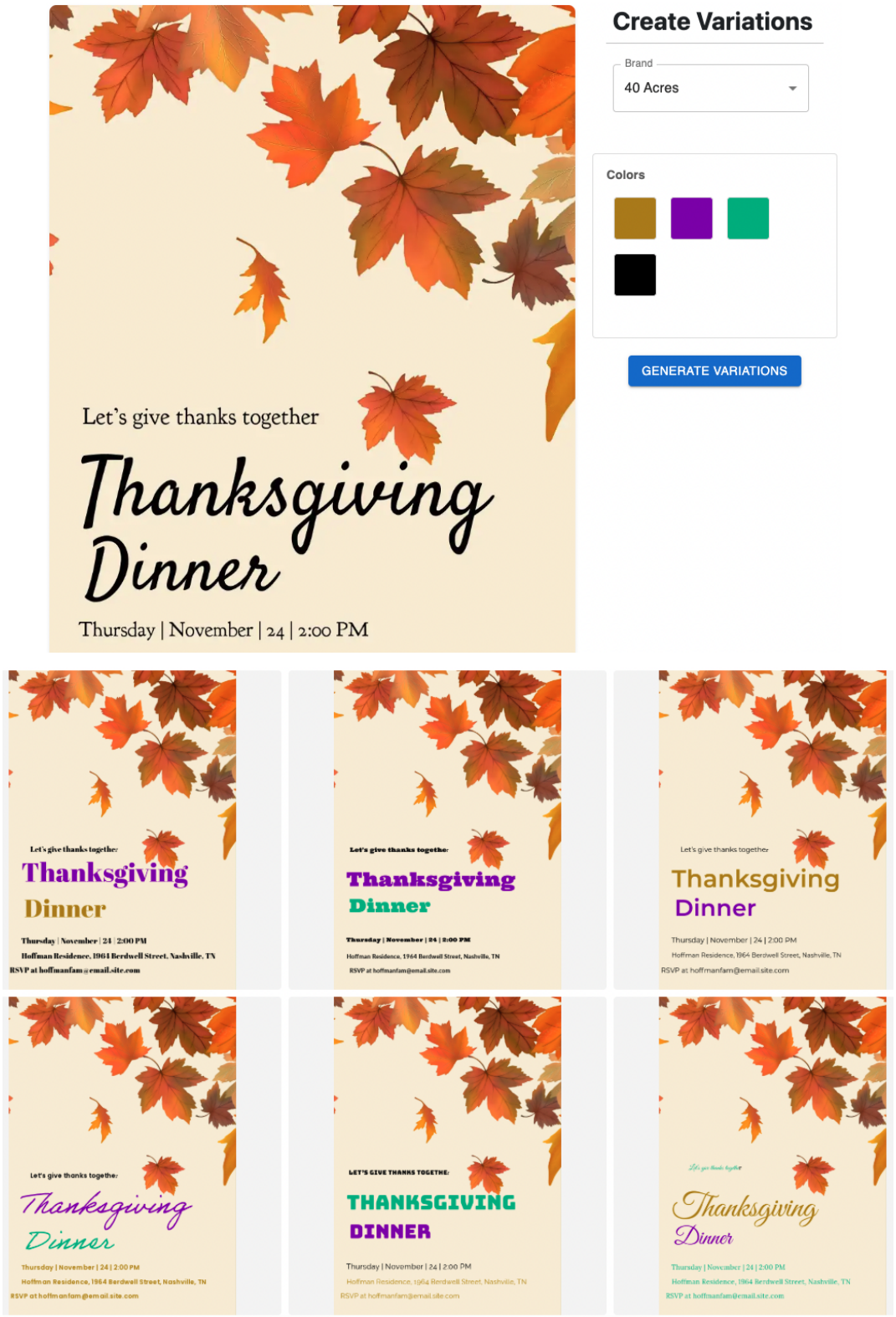

Figure 6: CPT applies a brand preset (colors + fonts) to generate variations that stay editable, contextually coherent, and consistent with brand identity.

A.5 Full Example of CML Representation

Below we include a complete Creative Markup Language (CML) specification for one of our generated designs. This highlights the structured representation of colors, fonts, shapes, and layout.

```
<cml v="3" numberPages="1">
    <page width="2550" height="3300" format="flyer" layoutID="unknown">
        <brand>
            <color value="#eceae3" />
            <color value="#400e04" />
            <color value="#a3665c" />
            <color value="#bd521d" />
            <color value="#400e05" />
            <color value="#e0e0e0" />
            <font name="Rubik-SemiBold" />
            <font name="Poppins-Regular" />
            <font name="Satisfy-Regular" />
            <font name="Poppins-Black" />
            <font name="Rubik-Italic" />
        </brand>
        <background type="color" id="0" entityId="b04d9fa0-6c15-459d-bb5f-fd4199c18616">
            <bounds top="0" left="0" width="2550" height="3300" rotation="0" z-index="0"
                />
            <style color="#fff6eb" />
        </background>
        <text id="1" entityId="26384d45-34d4-419f-9cbc-285588da6618">
            <bounds top="246" left="426" width="371" height="79" rotation="0" z-index="3"
                />
            <style alignment="left" layout="dynamic" />
            <p>
                <content>
                    E m e n
                </content>
                <style leading="1.2" color="#008045" font="Novecentosansnarrow-Bold" size
                    ="116" tracking="0" opacity="1" underline="false" fontSize="48" />
            </p>
        </text>
        <text id="2" entityId="0ed7c280-99ed-41d1-9b02-b77602ec6683">
            <bounds top="588" left="255" width="1287" height="58" rotation="0" z-index
                ="4" />
            <style alignment="left" layout="dynamic" />
            <p>
                <content>
                    Please join us for our annual
                </content>
                <style leading="1.2" color="#a8493f" font="Muli-Regular" size="100"
                    tracking="0" opacity="1" underline="false" fontSize="90" />
            </p>
        </text>
        <text id="3" entityId="2e175ef3-8f2e-465e-abbe-9eac674c1707">
            <bounds top="733" left="200" width="1850" height="498" rotation="0" z-index
                ="5" />
            <style alignment="left" layout="autoWidth" />
            <p>
                <content>
                    Thanksgiving
                </content>
                <style leading="1.2" color="#782010" font="Allura-Regular" size="374"
                    tracking="0" opacity="1" underline="false" fontSize="415" />
            </p>
        </text>
        <text id="4" entityId="23885bd6-6c78-4ee7-913e-6c3f7bfde03c">
            <bounds top="1650" left="227" width="1541" height="232" rotation="0" z-index
                ="6" />
            <style alignment="left" layout="autoHeight" />
            <p>
                <content>
                    Friday, Nov. 17th    |    12-2pm\r
                </content>
                <style leading="1.69" color="#8a362c" font="Muli-Bold" size="84" tracking
                    ="0" opacity="1" underline="false" fontSize="86" />
            </p>
            <p>
                <content>
                    Conference Room A
                </content>
                <style leading="1.69" color="#8a362c" font="Muli-Regular" size="84"
                    tracking="0" opacity="1" underline="false" fontSize="86" />
            </p>
        </text>
```

```
<text id="5" entityId="1193cfe6-366a-4a6d-808d-5e48a8fafa4e">
    <bounds top="2134" left="232" width="716" height="214" rotation="0" z-index
        ="7" />
    <style alignment="left" layout="autoHeight" />
    <p>
        <content>
            Don't forget to bring a dish!
        </content>
        <style leading="1.2" color="#782010" font="Muli-Regular" size="88"
            tracking="0" opacity="1" underline="false" fontSize="92" />
    </p>
</text>
<text id="6" entityId="8dbdd45b-6bdc-4e9c-a456-483b941ee8ea">
    <bounds top="338" left="422" width="375" height="54" rotation="0" z-index
        ="10" />
    <style alignment="left" layout="autoWidth" />
    <p>
        <content>
            Health Solutions
        </content>
        <style leading="1.2" color="#008045" font="NotoSans-Regular" size="48"
            tracking="0" opacity="1" underline="false" fontSize="46" />
    </p>
</text>
<text id="7" entityId="2498cdf5-2636-455b-837f-dded0229104e">
    <bounds top="1094" left="107" width="1465" height="498" rotation="0" z-index
        ="11" />
    <style alignment="left" layout="autoHeight" />
    <p>
        <content>
            Potluck
        </content>
        <style leading="1.04" color="#782010" font="Allura-Regular" size="374"
            tracking="0" opacity="1" underline="false" fontSize="415" />
    </p>
</text>
<image id="8" entityId="a4bed395-e000-4b23-a4c9-7079be7189bc" sourceType="
    designAsset" sourceId="529444607">
    <bounds top="1417" left="1073" width="1819" height="1801" rotation="0" z-
        index="13" />
    <content>
        watercolor pumpkin clipart
    </content>
    <style blendMode="normal" hasAlpha="true" />
    <colorGrid c1="#ffffff" c2="#ffffff" c3="#ffffff" c4="#ffffff" c5="#ebd286"
        c6="#ffffff" c7="#ffffff" c8="#ffffff" c9="#ffffff" />
    <effect name="shape" type="Rectangle" shape="" />
</image>
<image id="9" entityId="f49e9351-17e2-4db3-8e43-6f888f50c856" sourceType="
    designAsset" sourceId="546836347">
    <bounds top="-1022" left="2735" width="2304" height="1665" rotation="90" z-
        index="1" />
    <content>
        autumn leaves background vector | price 1 credit usd $1
    </content>
    <style blendMode="normal" hasAlpha="true" />
    <colorGrid c1="#ca4634" c2="#ffffff" c3="#ffffff" c4="#fbd278" c5="#ffffff"
        c6="#ffffff" c7="#c67029" c8="#ffffff" c9="#ffffff" />
    <effect name="shape" type="Rectangle" shape="" />
</image>
<image id="10" entityId="b18d5b08-4c33-44fe-ba86-09e71c170686" sourceType="
    designAsset" sourceId="546836347">
    <bounds top="2473" left="2715" width="1397" height="3408" rotation="90" z-
        index="12" />
    <content>
        autumn leaves background vector | price 1 credit usd $1
    </content>
    <style blendMode="normal" hasAlpha="true" />
    <colorGrid c1="#ca4634" c2="#ffffff" c3="#ffffff" c4="#fbd278" c5="#ffffff"
        c6="#ffffff" c7="#c67029" c8="#ffffff" c9="#ffffff" />
    <effect name="shape" type="Rectangle" shape="" />
</image>
<shape id="11" type="composite" entityId="b1f0a839-6464-4ef4-8536-9e1f8c677d74"
    sourceType="Adobe Stock" sourceId="596680639">
    <bounds top="261" left="345" width="53" height="106" rotation="53" z-index
        ="9" />
    <content>
        a pill icon on a white background
    </content>
    <search>
        pill icon
```

```
            </search>
            <style opacity="1" color="#008045" strokeColor="#008045" strokePosition="
                center" strokeWidth="1" strokeDashGeometryType="solid" />
        </shape>
        <shape id="12" type="Rectangle" entityId="2b32a5cc-476d-4f1e-98c2-2b67080e6f7e">
            <bounds top="94" left="-89" width="1162" height="594" rotation="0" z-index
                ="2" />
            <style opacity="1" color="#fff6eb" strokeColor="#e0e0e0" strokePosition="
                center" strokeWidth="0" strokeMiter="10" strokeDashGeometryType="solid"
                />
        </shape>
        <shape id="13" type="Ellipse" entityId="99c6361e-2f62-4b27-97db-5714d0276ce7">
            <geometry rx="70" ry="70" />
            <bounds top="240" left="245" width="149" height="149" rotation="0" z-index
                ="8" />
            <style opacity="1" color="transparent" strokeColor="#008045" strokePosition="
                center" strokeWidth="8.14" strokeMiter="10" strokeDashGeometryType="
                solid" />
        </shape>
        <shape id="14" type="Line" entityId="f6c37c37-c479-4311-a29a-f8b4705f2f6c">
            <geometry startX="0" startY="0" endX="765" endY="0" />
            <bounds top="2007" left="232" width="765" height="4" rotation="0" z-index
                ="14" />
            <style opacity="1" color="transparent" strokeColor="#782010" strokePosition="
                center" strokeWidth="4" strokeMiter="10" strokeDashGeometryType="solid"
                />
        </shape>
    </page>
</cml>
```