Constrained Generative UI for the Cockpit: A Comparison of LLM-Assembled and Pre-Defined Interfaces

Akshil Shah
Mercedes-Benz AG
Boeblingen, Germany
Technische Hochschule Ingolstadt
Ingolstadt, Germany
hello@akshilshah.com

Sebastian Zepf
Human-Computer Interaction
Mercedes-Benz AG
Boeblingen, Baden-Wuerttemberg
Germany
sebastian.zepf@mercedes-benz.com

Andreas Riener
Human-Computer Interaction Group
Technische Hochschule Ingolstadt
Ingolstadt, Bavaria, Germany
andreas.riener@thi.de



Figure 1: Outputs from the proposed generative GUI system under the same event: the system communicates the cause of the congestion while presenting contextually relevant supplementary content to keep the driver informed during the wait.

Abstract

In-car infotainment systems are growing in complexity, making traditional design and engineering approaches costly and difficult to scale. Recent advances in large language models (LLMs) offer context-aware, dynamic capabilities, enabling interfaces to adapt in real-time and reducing the need for manual design. We propose a scalable framework for Constrained AI-Assembled Graphical User Interfaces (GUIs) that dynamically assembles the interface at runtime from a pre-validated library of components using an LLM. We evaluated this approach in a comparative user study ($N = 30$) in a simulated driving environment, measuring usability, as well as driver's cognitive load, distraction, and situational awareness. Quantitative results showed no significant differences between AI-assembled and conventional pre-designed interfaces regarding usability and driver's mental state. Our qualitative feedback highlighted occasional challenges in information prioritization. We conclude that AI-assembled GUIs have the potential to achieve performance similar to conventional pre-defined interfaces and provide design recommendations to guide future work.

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License.
AutomotiveUI '26, Gothenburg, Sweden
© 2026 Copyright held by the owner/author(s).
ACM ISBN 979-8-4007-2814-3/26/09
https://doi.org/10.1145/3828157.3828780

CCS Concepts

• **Human-centered computing** → **Graphical user interfaces**; **Empirical studies in HCI**.

Keywords

Automotive User Interfaces, Context-Aware User Interface, User Interface Design, Generative AI, Generative User Interfaces

ACM Reference Format:
Akshil Shah, Sebastian Zepf, and Andreas Riener. 2026. Constrained Generative UI for the Cockpit: A Comparison of LLM-Assembled and Pre-Defined Interfaces. In *18th International Conference on Automotive User Interfaces and Interactive Vehicular Applications (AutomotiveUI '26), September 20–23, 2026, Gothenburg, Sweden.* ACM, New York, NY, USA, 12 pages. https://doi.org/10.1145/3828157.3828780

1 Introduction

In the automotive domain, graphical user interfaces (GUIs) are widely used and remain a key channel of interaction, particularly for system output [13]. Designing high-quality GUIs that meet user expectations is time-consuming and requires dedicated design and engineering expertise [23]. This challenge is increasing as modern cockpits evolve into "digital living rooms," integrating expansive feature sets via GUI [15]. At the same time, strict safety requirements must be met, since driver interaction is safety-critical and can have severe consequences [13, 38], especially during manual driving. Current industry standards rely on a deterministic approach where designers manually specify layouts for each use case, and

developers hard-code the logic to trigger them [4, 21, 22]. Although effective in simple scenarios, these static designs are inflexible and struggle to adapt to varying contexts. Dynamically adapting the interface in response to changing context offers the potential to improve user experience by providing richer interaction in less demanding conditions or during autonomous driving and to reduce distraction in critical situations by minimizing interaction. However, adapting a GUI for every contextual factor currently requires extensive manual effort. With a growing number of context factors, the resulting combinations make fully adaptive interfaces infeasible with manual design. The potential in breaking up this rigidity is especially evident in complex driving scenarios, as illustrated in the following examples:

- **Scenario A – Driving in an unfamiliar city:** Sarah is driving through a city she has never visited before. Heavy traffic and unfamiliar streets are limiting her spatial orientation. She is stressed, the lanes are complex, and she has no mental map to fall back on. Knowing that Sarah has never visited this area before and drawing on dynamic information such as current congestion and lane complexity, the system simplifies the map, enlarges key labels, suppresses non-essential details, and highlights only the most critical navigation information for a quick glance.
- **Scenario B – Night driving in difficult conditions:** It is late, visibility is poor due to rain and fog, and Sarah's fuel warning has just come on. Reduced visibility, tighter margins for error, and low fuel are all compounding the risk. Her display dims to avoid adding to the visual strain, surfaces only the most critical safety information, and makes it immediately clear where the nearest reachable fuel station is.

These two short scenarios encapsulate the potential of an idealized fully context-aware system. Achieving this via manual design requires explicit 'if-then' rules for every possible combination of contextual variables, rendering manual design fundamentally infeasible at scale. As Lindley et al. [22] highlight, this bottleneck necessitates a radical shift in Human-Computer Interaction (HCI) practice: moving from traditional methods where practitioners manually craft interfaces for stable software ecosystems toward a trajectory where AI models generate and adjust user interfaces dynamically. Recent advances in Large Language Models (LLMs) offer new options to navigate this shift, as their contextual understanding could help overcome the need to design for every combination and potentially reduce development time. However, it is unclear how well generative AI performs in automotive UI design in comparison to pre-designed interfaces. In this paper, we propose a Constrained AI-Assembled GUI that uses an LLM to generate GUIs using components from a pre-validated GUI library, bridging manual design and open-ended generation. Specifically, to approach the kind of adaptation illustrated through our two exemplary scenarios, our system reads the current user input and a set of available context information, combines them with the component library and the system instructions into a single structured prompt. This prompt is fed into an LLM that reasons about which GUI elements are relevant, composes components accordingly, and allocates them across pre-defined spatial zones by priority, with only validated output reaching the rendering layer (full description in subsection 3.1). We evaluated a functional prototype of our framework through a user study (N=30) in a simulated driving environment.

Our contributions to the HCI community ([20], ch. 1.2) are as follows:

- **Artifact:** We present a novel architecture for a constrained generative UI system with dynamic LLM-based interface assembly from a validated component library operating under spatial, structural, and safety constraints.
- **Empirical:** We report results from a user study comparing our constrained generative UI system against a pre-designed GUI ($N = 30$). Assessing usability and driver's cognitive load, distraction, and situational awareness, we found that there are no significant differences between the two systems, indicating that they perform on a comparable level. We provide four concrete design recommendations for constrained AI-generated UIs in the automotive context based on our findings.

2 Related Work

Conventionally, user interfaces are fully specified during development, relying on manually designed GUIs for anticipated scenarios [4, 21, 22]. Meixner et al. [23] highlight that UI-related factors account for approximately 48% of source code, 45% of development time, and 37% of maintenance effort, underscoring the resource intensity of conventional design workflows. As requirements grow, designers often have to prioritize the "happy path", leaving complex contextual intersections as secondary concerns, resulting in rigid systems that are costly to update and scale.

This challenge is especially prevalent in the automotive domain, where modern cockpits have evolved into what Hermann and Singh [15] describe as "digital living rooms", integrating expansive feature sets across pillar-to-pillar displays. Wirtz et al. [39] note that infotainment systems are typically designed for broad general use rather than specific scenarios, shifting the burden of information filtering onto the driver. A systematic review by Krstačić et al. [19] found that visual-manual interaction remains the leading cause of driver distraction, while Bölükbaşı et al. [3] report a consistent pattern of lower situational awareness scores under high-complexity interfaces, establishing distraction and situational awareness as primary safety metrics against which any new interface approach must be evaluated. Context-adaptive interfaces are widely proposed as a solution to this trade-off [33, 38], yet rule-based adaptive approaches remain constrained by deterministic logic: as the number of contextual variables grows, the required interface variants multiply, making scalable manual specification practically infeasible.

2.1 Generative User Interfaces

Recent work has begun exploring whether LLMs can move beyond generating text to generating the user interface itself. Leviathan et al. [21] demonstrate that modern AI models can assemble task-specific interfaces on demand, pointing toward a paradigm where software reshapes itself around the user's immediate goals rather than requiring users to adapt to fixed structures. Similarly, Hojo et al. [16] show that dynamically generating HTML-based GUIs within a conversational AI system significantly reduces mental demand, effort, and task completion time compared to text-only clarification interfaces, particularly in scenarios where multiple clarifying

questions are needed. Similar interaction patterns are already observable in systems such as the Claude interface (Sonnet 4.6) [2], where, instead of presenting follow-up questions purely as text, the system generates interactive UI elements (e.g., multiple-choice selections) that allow users to respond through direct manipulation rather than typed input. These developments further support the potential of LLM-generated interfaces to enhance usability in general-purpose contexts.

While run-time generation offers immense adaptability, recent work by Lindley et al. [22] highlights a critical distinction between Generative UI used as a design-time prototyping tool versus a run-time mechanism. Generating interfaces directly for end-users in real-time raises fundamental open questions for the HCI community regarding how such systems can consistently apply design heuristics and strictly adhere to safety guidelines. Okopnyi et al. [29] argue that unconstrained GUI generation risks placing elements in unexpected locations, undermining learnability and increasing cognitive load. Furthermore, LLM limitations such as hallucinations and embedded biases [26] could potentially produce misleading or unsafe interface states. Architecturally, approaches that render LLM output directly, such as generating raw HTML [16], may introduce additional risks of syntactic or structural errors reaching the user. Recent qualitative evaluations by Sawicki et al. [34] provide empirical evidence of these limitations; when testing state-of-the-art LLMs to generate unconstrained HTML/JS interfaces, they observed frequent functional flaws such as non-functional buttons, absent field validations, and inconsistent language translations.

These concerns have led to a shift toward more controlled forms of generative UI. For instance, Cao et al. [5] critique approaches based on direct code generation, noting that such methods often result in opaque and inconsistent interfaces that are difficult to control. Instead, they propose a specification-based approach in which the model generates a structured "Task-Driven Data Model", a high-level representation of user goals, entities, and their relationships, which is then translated into concrete interface components through pre-defined mapping rules. Furthermore, a recent framework named MorphGUI from Calo et al. [4] leverages LLMs to enable real-time, interactive UI customization by allowing end-users to provide structured natural language instructions that explicitly separate functional ("what it should do") from visual ("how it should appear") modifications. This separation of generative reasoning from interface rendering enables greater control over layout and behavior while preserving the flexibility of LLM-based generation. This line of work reflects a broader trend toward constraining generative processes to improve predictability and control.

2.2 Research Gap and Hypotheses

Although generative GUI approaches show promise [16, 21] and recent works began to address more controlled forms of it [4, 5], these approaches have primarily been explored in open-ended, user-driven scenarios, where users iteratively refine or customize interfaces for exploratory tasks. In contrast, safety-critical domains such as automotive interaction impose fundamentally different requirements. Interfaces must not only be structurally valid, but also support rapid comprehension, stable spatial organization, and minimal cognitive overhead under time-constrained conditions.

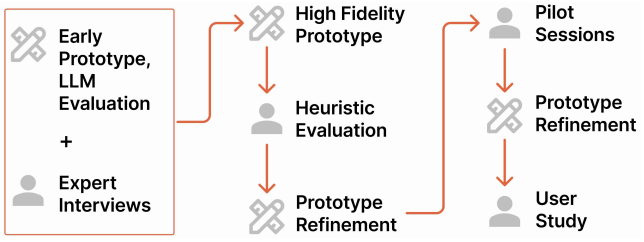


Figure 2: Overview of our human-centered design approach, illustrating the iterative progression from exploratory interviews and LLM prototyping to heuristic evaluation, pilot validation, and the main user study.

It remains unclear whether constrained, specification-based generative approaches can meet these requirements when interfaces are assembled automatically in real-time. To our knowledge, no prior work has empirically evaluated LLM-based GUI generation and assembly in a driving context. This gap motivates our Constrained AI-Assembled GUI framework, which automatically assembles context-aware interfaces using pre-defined atomic elements, while ensuring structural and safety constraints. By separating generative reasoning from controlled rendering, the approach aims to balance scalability and adaptability with the predictability required for safe in-vehicle interaction. To fill these gaps, this paper examines the performance of a constrained AI-assembled GUI in comparison to a pre-designed interface across usability and driver safety metrics based on the following hypotheses.

- **H1**: The AI-assembled GUI shows higher perceived usability (H1a), clarity (H1b), and informational value (H1c) than the pre-designed GUI.
- **H2**: The AI-assembled GUI causes lower cognitive load (H2a) and distraction (H2b), and higher situational awareness (H2c) than the pre-designed GUI.

3 System Design

The system was developed through an iterative, human-centered process involving multiple steps, as illustrated in Figure 2. We evaluated different LLMs [1, 12, 30, 31] in terms of their ability to generate and adapt the GUI, resulting in an early prototype. Each model was provided with an initial textual user input and thereafter a manually triggered contextual input, which led to the adaptation of the initial screen (see supplementary material for an example output). Results revealed that although GPT-5 performed best in generating and adapting the GUI, Gemini 2.5 Pro provided the best balance between GUI adaptivity and latency. We considered latency during model selection to ensure sufficiently smooth interactions within the user study and to avoid disruptions to the participant experience. However, latency was not treated as a primary factor in the final evaluation.

In parallel, we conducted expert interviews ($N = 7$) to explore and verify the problem space for AI-driven in-car user interfaces and to ground design decisions in expert insights (full list of questions in supplementary material). Two researchers performed thematic clustering, revealing three central themes:

- High interest in AI-generated GUI: The desire of "use-case-less" design was identified, aiming to lessen the need for layouts to be manually designed for every possible scenario.
- Balancing Freedom and Constraints: Concerns that too many constraints could produce results that are similar to rule-based systems emerged.
- Safety Constraints: The need to respect user control and follow design system guidelines was highlighted. The system should not overstep when the driver expects to be in control and must adhere to safety guidelines.

Findings from these stages informed the development of a high-fidelity prototype, which was subsequently evaluated through a heuristic review based on the materials and guidelines from Nielsen Norman Group [25, 27]. We made refinements following this evaluation to address the issues which received severity ratings of 2, 3 and 4 on the scale of 0-4 [28]. For instance, issues with the highest severity rating (4) included the lack of system feedback in response to user input and the use of an unlabeled icon within the list. A complete overview of all identified issues and their respective severity ratings is provided as supplementary material. Pilot sessions ($N = 3$) were then conducted to validate the experimental setup and identify remaining usability issues. Based on the observations, a standardized fixed delay of 32 seconds, matching the average latency of the AI-based system was applied to the pre-designed system to ensure comparable temporal behavior of the two systems.

3.1 Constrained AI-Driven GUI Assembly Architecture

To realize the constrained AI-assembled GUI, we chose the web-based architecture shown in Figure 3 over static simulation methods. This client-server approach was selected because it successfully balances the need for dynamic layout generation at runtime with the strict structural control required for automotive safety and brand identity. Moreover, it provides a higher degree of ecological validity by replicating how such LLM-driven pipelines might be implemented in modern, digitally connected vehicles. While not intended as a direct production implementation, the architecture reflects a technically plausible deployment model and therefore more closely mirrors real-world system behavior than purely prototypical approaches (e.g., interaction prototyping tools such as ProtoPie). As reliable connectivity was ensured in the study environment, the LLM was integrated as a cloud-hosted service.

In the following, we describe the considered system inputs, the processing via LLM, and the output generation.

- **Input (GUI Component Library, Contextual Data, Guidelines):** A pre-defined GUI library acts as the foundation for the system's selection (see supplementary material for a full list). A Python environment serves as the central hub where the current status of the screen, a variety of contextual data (environmental, user preferences, etc.), and user input are collected and structured for interpretation. This information is then bundled in a system prompt along with condensed safety guidelines [10, 18, 24] and output instructions.
- **Processing (LLM, GUI Generation):** The system prompt (provided as supplementary material) is sent to the agent (Gemini-2.5-pro [12]), acting as contextual interpreter and GUI architect. The prompt followed a modular structure, including the following distinct sections: system identity, non-negotiable rules, the task objective, capabilities, output specification, driver profile, layout principles, interaction guidelines, usability heuristics, condensed safety guidelines, and a final model instruction to self-validate the output before returning. By evaluating the prompt and real-time context, it reasons which atomic elements from the library are relevant and how they should be used to create components to match the content it defined. Furthermore, it dynamically allocates these components to one of two pre-defined spatial zones (a vertical-left space or a horizontal bottom bar) and to a space within these zones depending on the information's priority and complexity. On each turn, the current screen state and the conversation history were appended to support incremental, context-aware, and consistent updates.
- **Output (Validation, Controlled Rendering Layer):** Unlike recent approaches that render LLM-generated code directly to the user (e.g., [16]), we separate generative reasoning from rendering. As result of the LLM-based reasoning, a set of structured JSON component configurations is sent to the rendering layer. Similar to frameworks such as MorphGUI [4], the JSON is then interpreted by a deterministic React layer that renders the referenced components from the validated library. The component vocabulary is fixed, but their content, selection, composition, and arrangement is generated at runtime. Output that is missing or cannot be parsed as valid JSON is rejected rather than displayed. This ensures that only structurally valid interface states reach the driver, avoiding the syntax errors, broken rendering, and unintended artifacts associated with direct code generation.

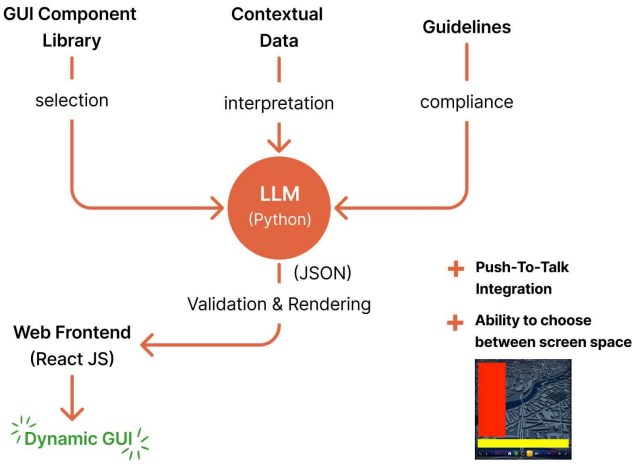


Figure 3: Overview of the constrained LLM-driven GUI assembly architecture.

Furthermore, the system was equipped with voice input as the primary interaction modality through a Push-to-Talk mechanism, supporting natural interaction. Touch input was included as a secondary modality for non-critical actions such as scrolling or dismissing panels. A short tone indicated that the system was listening, and a confirmation beep indicated a GUI update upon completion of processing. As the system required an average of 32 seconds to

generate a complete GUI state, these auditory cues were critical for making the user aware of the system status. As stated above, latency optimization was not a focus of our investigation.

4 User Study

We used a within-subjects design for our study, where each participant interacted with two interface conditions in randomized order:

- **Condition A (Pre-Defined GUI):** A baseline system with a rule-based adaptive interface. Spoken input was transcribed by a speech-to-text engine and then interpreted by Gemini-2.5-pro into a bounded set of intents (e.g., find rest stop, start navigation, or a named stop). Each intent triggered a corresponding pre-designed layout. In accordance with common automotive practice, the layouts prioritized information relevant to the primary user goal of reaching the destination (e.g., navigation instructions) over secondary infotainment information (e.g., latest news). Within these layouts, adaptation was handled through hard-coded rules for driver preferences. For example, the briefing and news panels expanded or condensed their content based on the driver's information-density preference, and rest-stop descriptions varied with the stored dietary preference.
- **Condition B (AI-Assembled GUI):** A real-time GUI in which an LLM assembled the layout at runtime from the shared component library, instead of mapping it to a fixed intent. Based on the current context and driver preferences, the model selected which components to display, composed them, and allocated them across the two screen zones by priority, producing a context-specific layout on each generation (see Section 3.1 for the full pipeline).

Both systems were based on the same component library and shared the same push-to-talk speech-recognition pipeline, in which spoken input was transcribed by a commercial speech-to-text engine (ElevenLabs, model scribe_v1 [8]).

4.1 Scenario & Task Design

The study employed the narrative of a user journey to immerse participants. The following three events were designed to produce a single-shot interface configuration, enabling evaluation of a specific GUI state and avoiding reconfiguration triggered by further user input:

- Briefing (Proactive): Upon vehicle entry, the system presented a daily briefing based on schedule and preferences.
- Search (User-initiated): In simulated highway driving, participants initiated a voice-based search for a breakfast stop.
- Notification (Proactive): A traffic jam triggered a system notification requiring re-prioritization of information.

While, as noted above, latency optimization was not a focus of our investigation, the study design was optimized to mitigate potential impact of the delay on user feedback. First, we deliberately chose events that are not time-critical. Second, participants were explicitly informed about latency as technical limitation and instructed to ignore it in their evaluations. Finally, Events 1 and 3 were moderator-triggered (simulated proactivity), with the trigger release aligned with the end of the scripted instructions, so that the output appeared approximately at the right narrative moment to keep the sessions immersive. As Event 2 was participant-initiated to account for different interaction scenarios, this mitigation was not applicable and the delay was experienced directly.

Variable Type	Variable	Levels / Measurement
Independent	Interface Type	AI-Assembled GUI, Pre-Defined GUI
Independent	Event Type	Briefing, Search, Notification
Dependent	Usability / Experience	UEQ+, UEQ-S scales(Clarity, Value, Pragmatic, Hedonic)
Dependent	Cognitive Load	NASA-TLX (Mental Demand)
Dependent	Distraction	7-point Likert (Event 2)
Dependent	Situational Awareness	3-item SA task accuracy (Event 2)
Dependent	Overall Preference	100-point allocation task (post-condition)

Table 1: Summary of independent and dependent variables used in the study.

To control for order effects and learning bias, two distinct video segments (V1, V2) of highway-driving were used during Event 2. Both segments were extracted from the same 50-minute driving video [37] and were reviewed by two independent experts to ensure comparable driving complexity, traffic density, and visual dynamics. A small quantitative comparison of observable characteristics (e.g., overtaking interactions) further confirmed similarity. The video segments were counterbalanced with interface conditions (A/B), resulting in four balanced experimental groups, additionally mitigating potential effects of the driving scenario.

4.2 Measures

Event-based measures included perceived GUI quality (UEQ+ module for graphical user interfaces [36], including Clarity, Value, Visual Aesthetics), cognitive load represented through the single item for Mental Demand from the NASA-TLX [14], and open-ended feedback. For Event 2 (driving monitoring) an item for self-reported distraction was added. All subjective ratings were collected based on 7-point Likert-Scales. In addition, we captured perceptive situational awareness via a three-item verification task (accuracy score 0–3) for Event 2, with an approach inspired by Level 1 of the Situation Awareness Global Assessment Technique (SAGAT) [9] (See Table 1). Unlike traditional SAGAT, a retrospective assessment was used to preserve ecological validity and avoid interrupting the continuous driving scenario and the interaction with the system. As post-condition measures, we used the pragmatics scale of the UEQ-S [35] to assess general usability. We also decided to include the corresponding hedonics scale for insights into non-instrumental qualities in addition to Visual Aesthetics. Finally, a 100-point allocation task captured the overall system preference. The full questionnaire is provided as supplementary material. The study was approved by the institutional ethics board and participation was voluntary under informed consent.

4.3 Apparatus

We used a Mercedes-Benz S580 as study vehicle. For safety reasons, the study was conducted with the vehicle stationary. Both interface conditions were displayed on the vehicle's 11.9-inch central head unit. A 65-inch external monitor positioned in front of the vehicle presented either a static image (Event 3) or counterbalanced highway driving footage (Event 2) (see illustration of the study setup in Figure 5). It was unused during Event 1, as driving had not yet commenced. To simulate a steering-wheel-mounted voice button

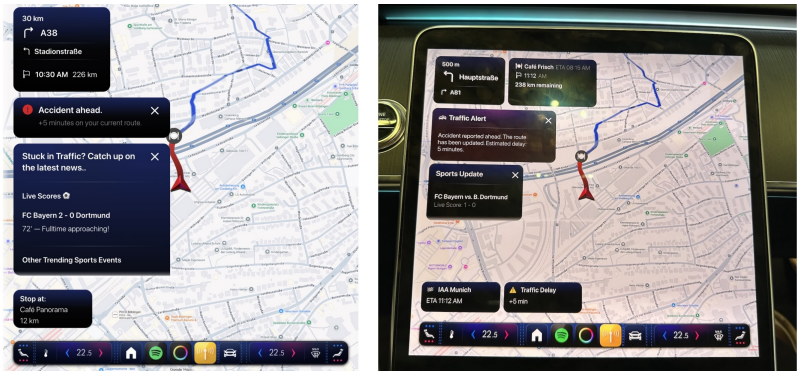


Figure 4: A Pre-designed GUI (Left) and an exemplary AI-assembled GUI (Right) under the same driving scenario.



Figure 5: Pictures of the study setup showing the exterior view of the study vehicle (left) as well as the participant perspective (right).

without hardware modification, a wireless keyboard was positioned in an ergonomic location within the center console. Participants initiated voice commands by pressing and holding the spacebar, ensuring they could provide natural language input without diverting their visual attention from the road. Voice output was deliberately omitted to isolate evaluation of the graphical interface.

4.4 Procedure

As illustrated in Figure 6, participants first received standardized instructions of the goal and the procedure of the upcoming study, as well as an introduction to the operation of the system. To reduce expectation bias, the two interface conditions were presented as "System A" and "System B". Also, participants were explicitly informed that response delays of both systems were due to technical limitations of the prototype and were instructed not to take this into account in their evaluations. Each session consisted of the three events described in subsection 4.1, followed by the questionnaire asking for immediate event-based feedback. After completing all events, participants filled out the post-condition questionnaire. The entire sequence was then repeated with the second interface condition using the alternate driving video segment. The session concluded with the final preference allocation task and a debriefing.

4.5 Participants

An a-priori power analysis (G*Power) indicated a target sample of $N = 34$ to detect medium effects ($d = 0.5$) with 80% power. Due to operational constraints, the final sample consisted of $N = 30$. A post-hoc sensitivity analysis confirmed that medium effects ($d \geq 0.53$) could be detected with 80% power. 30 participants (18 male, 12 female; age range 18–64 years) were recruited internally from Mercedes-Benz. None were involved in the development of the evaluated systems. Participants reported a mean driving experience of 15.1 years ($SD = 12.64$, range 0–40) and moderate AI familiarity ($M = 2.47$, $SD = 0.90$ on a 0–4 scale).

5 Results

All quantitative data was assessed for normality using the Shapiro-Wilk test (test results are provided as supplementary material). Paired-samples t-tests were applied to normally distributed data, while Wilcoxon Signed-Rank tests were used for non-normal data (significance level $\alpha = .05$). Table 2 summarizes the descriptive statistics and test results. Graphs for all measures are attached in Appendix A.

Pragmatics, Clarity, and Value: Overall, there were no significant differences in pragmatics, clarity, and value between the AI-assembled GUI and the pre-designed GUI. Considering the mean value, the AI-assembled GUI was rated slightly easier to use ($M = 5.53$ vs. $M = 5.43$, $p = .698$), while the baseline was marginally more efficient ($M = 5.27$ vs. $M = 5.17$, $p = .804$) and supportive ($M = 5.83$ vs. $M = 5.73$, $p = .528$). The largest numerical difference emerged in the Confusing-Clear dimension, favoring the baseline ($M = 5.30$ vs. $M = 4.87$, $p = .162$). A similar pattern emerged in the UEQ+ Clarity scale where the baseline scored numerically higher on perceived clarity, without a significant difference ($M = 5.52$ vs. $M = 5.30$, $p = .089$). The perceived informational value was nearly identical ($M = 5.44$ vs. $M = 5.47$, $p = .872$).

Cognitive Load, Distraction, and Situational Awareness: There was also no significant impact of the GUI variants on cognitive load, distraction, and situational awareness. In particular, the AI-assembled GUI generated a slightly higher reported mental load ($M = 3.91$ vs. $M = 3.66$, $p = .235$). Conversely, participants reported marginally lower distraction with the AI system ($M = 3.60$ vs. $M = 4.03$, $p = .146$). Although the difference was not statistically significant, it suggests that the system's dynamic context-filtering did not produce additional visual-manual distraction. Finally, situational awareness scores were marginally higher for the baseline ($M = 1.48$ vs. $M = 1.21$, $p = .310$).

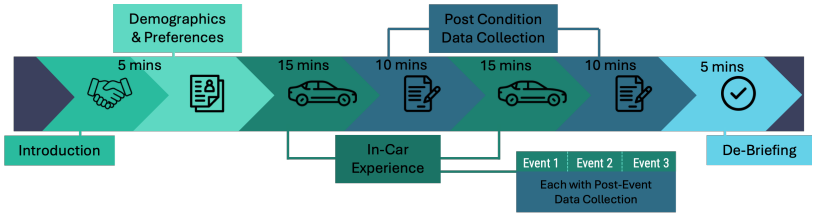


Figure 6: Schematic illustration of our study procedure centered around two in-car sessions in randomized order, each presenting one interface condition in three events.

Hedonics and Visual Aesthetics: The non-instrumental qualities followed the same pattern, yielding no significant differences. The AI-assembled GUI was rated marginally more exciting ($M = 4.67$ vs. $M = 4.47$, $p = .423$) and interesting ($M = 4.93$ vs. $M = 4.87$, $p = .801$), while the pre-designed baseline scored slightly higher on the Conventional-Inventive dimension ($M = 4.17$ vs. $M = 3.97$, $p = .573$). Both systems received identical scores on the Usual-Leading-edge dimension ($M = 3.90$, $p = 1.00$). Visual aesthetics were similarly comparable, with the pre-designed GUI scoring marginally higher on the aggregated UEQ+ scale ($M = 5.22$ vs. $M = 5.16$, $p = .346$), and no statistically significant differences found across any individual aesthetic item.

Overall Preference: The pattern of non-significant differences was also mirrored in subjective user preference. When asked to allocate 100 points to indicate their overall preferred system, participants split their points almost evenly between the AI-assembled ($M = 48.73$, $SD = 18.26$) and pre-designed ($M = 51.27$, $SD = 18.26$) GUIs, with 14 users favoring the AI, 15 favoring the baseline, and one remaining neutral. The baseline received five preferences ≥ 70 points compared to 4 for the AI-assembled system. The range of allocated points spanned 20 to 95 for the pre-designed GUI, and 5 to 80 for the AI-assembled GUI.

Open-ended Feedback: Of the 30 participants, 20 provided additional open-ended feedback. The qualitative data consisted of brief open-ended questionnaire responses rather than rich interview transcripts. Given the limited depth and length of the responses, a full thematic analysis (e.g., after Braun and Clarke [6]) was not considered appropriate. Instead, an inductive coding approach was used to identify recurring issues and participant concerns.

While interacting with the AI-assembled prototype, participants appreciated the dynamic content, including the morning greeting, weather, calendar summaries, restaurant ratings, and traffic updates, and several explicitly valued the system's dynamic visual adaptation, such as coloring the alert panel red during a safety notification (P14: "Red box was a good indicator"). Through our analysis of the open feedback, we found three themes highlighting potential areas for improvement, alongside which we report our qualitative findings including exemplary participant statements (number of the respective participants in brackets).

The most frequently raised theme, mentioned by 12 participants, concerned visual presentation and load. Here, participants found the simultaneous display of multiple generated blocks demanding, citing "too much info" (P4) and "too many blocks" (P28), and described entertainment content competing with safety alerts (P6: "split screen with alert was too much. Felt like it was yelling at me"). One participant captured the tension between richness and glanceability directly (P3: "more distracting since it was easy to read"), while others pointed to small fonts and low-contrast colors that were hard to read at a glance (P19: "grey color hard to read. Font too small").

As second theme, aspects around organization and glanceability were raised by 10 participants and can be connected to the quantitative trend toward lower clarity in the AI condition. The two most common single issues in the AI-condition feedback, each raised by three participants, were ungrouped navigation information (P2, P4, P11) and uncertainty about information hierarchy, i.e. what should appear at the top (P3, P10, P14). Participants further reported difficulty locating requested items at a glance (P11: "had to focus to locate name of the café", P13: "hard to distinguish navigation/meeting info from news"), duplicate content (P22), and the disappearance of navigation across generated states (P3: "navigation shouldn't disappear").

The third theme, content relevance and prioritization, was raised by 9 participants, several of whom questioned why preference-based content appeared during driving, with politics and sports the most frequent targets (P25: "election update not valuable", P11: "Don't know why soccer score update info is there at the bottom, am I suppose to drive there?"). Others asked for non-essential content to be reduced while driving (P1: "during driving, reduce other info like weather, news") and noted that core operational details were sometimes missing (P4: "no info on location, distance, detour").

6 Discussion

Our findings did not reveal statistically significant differences between the AI-assembled and pre-designed interfaces. The relatively small sample size of $N = 30$ in combination with the assumption of a medium effect size $d = 0.5$ is a potential reason why our study may not have had the power to detect small differences between the two interface variants. Alternatively, it is also possible that the two designs simply did not differ much, and that both reached a genuinely comparable level of quality. Rather than focusing on the absence of statistical significance, we discuss the practical implications of our findings. The AI-generated GUI performed comparably to the expert-designed GUI across the evaluated dimensions, with no significant differences between the two. For the question of whether an LLM-generated interface can serve as a viable alternative, this comparability is itself an interesting and relevant finding.

Looking specifically at usability, clarity, and value (H1), the two interfaces did not differ significantly, so H1 was not supported. This seems to conflict with recent research by Hojo et al. [16], who reported that their AI-based GUI system significantly improved

	AI-Assembled		Pre-Designed			
Metric	**M**	**SD**	**M**	**SD**	**Test Statistic**	**p-value**
UEQ-S Pragmatic Qualities						
Obstructive-Supportive	5.73	1.08	5.83	0.65	$W = 72.0$	.528
Complicated-Easy	5.53	1.25	5.43	1.22	$t(29) = 0.39$	.698
Inefficient-Efficient	5.17	1.42	5.27	1.23	$W = 153.5$	.804
Confusing-Clear	4.87	1.48	5.30	1.15	$t(29) = -1.43$	.162
UEQ-S Hedonic Quality						
Boring-Exciting	4.67	1.56	4.47	1.70	$t(29) = 0.81$	.423
Not interesting-Interesting	4.93	1.41	4.87	1.72	$W = 119.0$	.801
Conventional-Inventive	3.97	1.81	4.17	1.60	$W = 120.0$	.573
Usual-Leading-edge	3.90	1.71	3.90	1.90	$t(29) = 0.00$	1.00
UEQ+ Scales						
Clarity						
Aggregated scale	5.30	0.97	5.52	1.12	$W = 108.5$	.089
1. Poorly grouped-Grouped	5.39	1.02	5.50	1.21	$t(29) = -0.59$	.558
2. Unstructured-Structured	5.30	1.04	5.61	1.08	$t(29) = -1.78$	.086
3. Disordered-Ordered	5.26	1.16	5.43	1.19	$t(29) = -0.96$	.346
4. Disorganized-Organized	5.26	0.92	5.54	1.21	$W = 80.0$	.130
Value						
Aggregated scale	5.44	0.84	5.47	0.79	$t(29) = -0.16$	.872
1. Not elegant-Elegant	5.23	1.08	5.21	1.13	$t(29) = 0.12$	.906
2. Not presentable-Presentable	5.53	1.02	5.66	0.82	$t(29) = -0.81$	.425
3. Tasteless-Tasteful	5.43	1.01	5.39	0.93	$t(29) = 0.28$	.778
4. Inferior-Valuable	5.58	0.72	5.61	0.66	$t(29) = -0.21$	.835
Visual Aesthetics						
Aggregated scale	5.16	0.95	5.22	1.00	$W = 127.5$	.346
1. Ugly-Beautiful	5.19	0.91	5.27	0.86	$t(29) = -0.56$	.580
2. Unappealing-Appealing	5.24	1.02	5.18	1.26	$W = 137.0$	.976
3. Unpleasant-Pleasant	5.27	0.98	5.39	1.01	$W = 91.5$	.252
4. Lacking style-Stylish	4.96	1.07	5.03	1.08	$t(29) = -0.54$	.597
Mental Load						
Aggregated	3.91	1.17	3.66	1.23	$t(29) = 1.21$	.235
Event 1 (Briefing)	3.80	1.85	3.37	1.67	$t(29) = 1.12$	.274
Event 2 (Driving)	4.00	1.68	3.87	1.87	$W = 110.0$	.584
Event 3 (Traffic)	3.93	1.53	3.73	1.55	$W = 178.0$	.563
Distraction						
Event 2 (Driving)	3.60	1.61	4.03	1.99	$W = 110.5$	.146
Situational Awareness						
Event 2 (Driving)	1.21	1.01	1.48	0.99	$t(29) = -1.03$	.310

Table 2: Comparison of AI-Assembled and Pre-Designed interfaces across UEQ-S, UEQ+, and task-related measures (mental load, distraction, and situational awareness), showing descriptive statistics (M, SD) and inferential test results.

usability and reduced mental demand. However, they used a text-based conversational interface as baseline, where added GUI widgets provide a clear efficiency gain. In our study, the AI system was compared against a state-of-the-art GUI, which represents a stronger baseline than text-based conversational interfaces, potentially reducing the margin for observable improvement. Further, our findings align with work from other domains, such as Calò et al. [4], who achieved an 'acceptable' usability score ($SUS = 68$) for real-time GUI customization in a less safety-critical calendar app. In line with prior critiques of Generative GUIs arguing that it may increase cognitive overhead by violating prioritization and consistency of the layout [29], our qualitative findings and the numerically lower (non-significant) clarity scores point out that caution is advised and that there is definitely potential for improvement in AI-based GUIs when it comes to the details.

As the GUI variants did not show a significant effect on cognitive load, distraction, and situational awareness, H2 was not supported. While measures remained within comparable ranges across conditions, the interplay between distraction and situational awareness revealed an interesting finding. Participants reported numerically lower distraction with the AI-assembled GUI ($M = 3.60$ vs. $M = 4.03$), suggesting that it freed up the driver's cognitive resources. Paradoxically, this reduction in distraction was not accompanied by improved situational awareness, which remained marginally lower than the baseline. Based on the qualitative feedback, a potential reason for this discrepancy is that the successful generation of non-distracting functional GUI parts does not automatically translate to more attention to the driving scene. Several participants stated that their attention was caught by the personalized secondary content the AI brought into the GUI with varying priority, such as political news or sports scores. In some cases, this was perceived as "informational noise" in an active driving context. Some participants explicitly questioned the system's prioritization rationale, asking "Don't know why soccer score update info is

there at the bottom, am I suppose to drive there?" Another participant noted that "during driving, it would help when other info like weather or news is reduced". This was especially highlighted when critical operational details, such as distance and detour information, were noticeably absent. As noted in recent research by Ferrario [11], this dynamic can be understood through a 'triadic model' involving the human, the AI agent, and digital resources. In our constrained assembly, the agent acts as an active mediator of these resources. When the interface autonomously reorganizes information, it implicitly makes judgments about importance on behalf of the driver. When these mediation decisions occur without communicating their underlying rationale, this absence of an explanation of the "why" behind the layout prioritization can disrupt the driver's mental model and create tension, forcing the driver to "work harder" to interpret the layout. Over repeated interactions, this risks undermining trust in the system [32]. These observations align with recent findings by Sawicki et al. [34], who noted that while LLMs can adapt surface-level GUI aesthetics for different personas, they still lack a deep, nuanced understanding of user behavior and context, which is required for meaningful prioritization. Approaches discussed earlier [4, 5], attempt to solve these issues by allowing users to inspect and iteratively tailor the interface to their evolving needs. While this might be effective for open-ended information tasks, it is incompatible with automotive interfaces, where iterative manual UI would conflict with time-critical interactions and potentially increase distraction. A potential way to address this in our constrained AI-assembled approach could be additional guidance and restrictions regarding prioritization.

When looking at the outputs in detail, it is noteworthy that the AI-assembled system achieved layouts and usability comparable to the pre-designed interface even with a rather small set of predefined GUI components. On the one hand, this suggests that future work should expand the component library to increase the range of visual solutions for more tailored designs.

On the other hand, due to the non-determinism of the AI-based system, the GUI outputs for some of the repeated situations varied quite broadly. For instance, the "Additional Navigation" component exhibited especially strong structural differences (see Figure 7). While this element consistently displayed supplementary navigation information, its emphasis and information hierarchy changed across sessions. Since these inconsistencies primarily occurred between sessions rather than within a single interaction, they were rarely reported by our participants. Accordingly, it is important that future work conducts longitudinal studies with a focus on reproducibility of UI outputs. In particular, the right compromise between design freedom, component library scope, and consistency needs to be found, as this variability in generated layouts across contexts could potentially introduce interpretability challenges [29] and increase distraction, especially for driving-relevant functions and information.

6.1 Design Recommendations for Constrained Generative Interfaces

A recent initiative within the HCI community by Lindley et al. [22] challenges researchers to envision a new engineering and technical stack for generative UI that centers user needs and evaluates the trade-offs of different technical approaches. Following this initiative, we provide four concrete design recommendations we identified as valuable for generative automotive user interfaces based on our constrained AI approach.

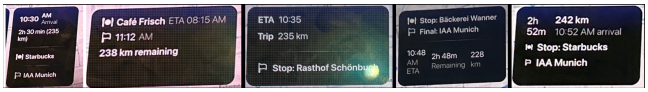


Figure 7: Variations of "Additional Navigation" component generated by AI-assembled GUI under Event-3 across different study sessions.

- **Bound the AI design space:** Rather than allowing an AI system to freely generate interface elements or code, generative reasoning should operate on a validated component library that complies with established automotive design and safety guidelines (e.g., [10, 18, 24]). In addition, both permissible and non-permissible configurations should be specified (e.g., as AI-readable part of the component library) to limit the possible interface states to a more controlled design space and to avoid unwanted outputs.
- **Constraint and validation mechanism:** In safety-critical domains, generative systems must operate under explicit constraints derived from domain guidelines and regulatory requirements. Embedding these constraints within the system prompt and rendering logic with clear prioritization rules, information density limits, and visual design standards (e.g., minimal font sizes) reduces the risk of safety violations. In addition, we found that core operational details such as the location and distance for the navigation might be missed out by our current approach. Thus, we recommend extending the validation mechanism to also consider potentially missing content and the detection of hallucinations.
- **Preserve a stable spatial grammar:** To avoid adaptation in unwanted areas (e.g., time-critical, driving-relevant information) and to preserve predictable output patterns, certain spatial structures should remain stable. This means defining fixed regions on the display for critical information, primary controls, and secondary content. This allows restricting generative assembly to pre-defined interface regions and enables contextual adaptation where the benefit of adaptation outweighs risks.
- **Separate reasoning from rendering:** LLM output should not be rendered directly to the user interface. Producing structured configuration data that is interpreted by a deterministic rendering layer prevents malformed output and ensures that only valid interface states reach the user. For system failures, dedicated fallback mechanisms are needed to avoid that relevant information does not reach the user at all.

6.2 Limitations and Future Work

As with every work, our research comes with limitations that have to be noted. First, our study focused on one-shot generation of interfaces. While this allowed a systematic comparison of system outputs, we did not capture potential longitudinal effects and impact of variances in the repeated generation of interfaces in the

same scenario (highlighted in the discussion based on the navigation module). Accordingly, longitudinal studies with repeated interactions are needed to examine whether this variance affects user focus and how it impacts trust, mental models, and acceptance.

Second, we used a stationary vehicle with a passive monitoring task rather than active driving for safety reasons. As a result, we did not involve physical control demands or real-time driving decisions, and the findings reflect performance under comparatively low operational workload. Evaluating constrained generative interfaces under genuine driving conditions is therefore an essential next step. Future studies employing active driving simulators, and ultimately on-road evaluation, are necessary to assess AI-assembled interfaces under higher workload and more realistic operational conditions.

Third, situational awareness was assessed using a short retrospective recall task immediately after each scenario in order to avoid interrupting the driving task. This approach focused primarily on Level 1 situational awareness (perception). Higher levels, such as comprehension and projection, were not assessed. Future research should incorporate multi-level situational awareness measures to provide a more comprehensive evaluation.

Fourth, all participants were recruited from within the institution where the study was conducted. While this included a substantial range in age as well as reasonably balanced gender ratio, the sample does not fully represent the diversity of real-world vehicle users. Future studies should thus include larger and more diverse samples to strengthen external validity.

Fifth, for the purpose of our study, any output that could not be parsed as valid JSON was rejected and the previous screen stays in place, so the driver never faces a broken or empty display after a failed generation. Further, no mechanisms to safeguard against hallucinations, which are still a major drawback of LLMs [7], were implemented. The constraints we impose regulate the structure of the interface, not the correctness of its content, so the model can still place accurate-looking but wrong information into a perfectly valid component, such as a misstated distance or a fuel station that does not exist [17]. Some hallucinations may be categorized as a usability problem, but in a moving vehicle a wrong navigation or safety cue that looks entirely trustworthy could result in bad driving decisions and severe safety risks. Accordingly, to bring generative UI approaches closer to a real-world application, future research should focus on failure-handling mechanisms.

Finally, the prototype relied on a cloud-hosted LLM and exhibited an average generation latency of approximately 32 seconds, which is not suitable for real-world driving contexts. A cloud-based approach also involves recurring usage-based inference costs, network bandwidth requirements, and vulnerability to degraded or absent connectivity in areas with poor coverage. In addition, privacy concerns may arise, as the contextual data may be sensitive. On-board or hybrid local–cloud inference could address these issues around latency and privacy by keeping time-critical generation and sensitive data within the vehicle. A smaller model, specifically fine-tuned for GUI assembly, could be a promising approach for future work in this regard.

7 Conclusion

Designing automotive infotainment interfaces that adapt to dynamic driving contexts remains challenging, as generative AI approaches must balance flexibility with safety and predictability. In this work, we presented a Constrained AI-Assembled GUI framework in which an LLM dynamically composes interfaces at runtime using a pre-validated component library. A comparative in-vehicle user study ($N = 30$) evaluated the system against a conventional predesigned interface. Results showed comparable performance across usability, cognitive load, driver distraction, and situational awareness, with no statistically significant differences and near-evenly split user preferences. These findings suggest that constrained generative assembly yields potential in automotive applications, being able to stay within a comparable level of distraction and to maintain glanceability without critical usability flaws in our one-shot generation setup. Future work should focus on optimizing latency of the UI generation and explore hybrid interface architectures in which safety-critical elements remain persistent while AI-driven assembly operates within designated dynamic regions to deliver context-aware content.

References

[1] Anthropic. 2025. Claude 4. https://www.anthropic.com/news/claude-4. Accessed 2026-02-25.

[2] Anthropic. 2026. Visual and Interactive Content in Claude. https://support.claude.com/en/articles/13641943-visual-and-interactive-content. Accessed: 2026-03-17.

[3] Elif Bölükbaşı, Diah Ayu Irawati, Michael A. Gerber, and Andreas Riener. 2024. Impact of Infotainment System Complexity on Driver Situation Awareness in Manual Vehicles. In *Adjunct Proceedings of the 16th International Conference on Automotive User Interfaces and Interactive Vehicular Applications* (Stanford, CA, USA) *(AutomotiveUI '24 Adjunct)*. Association for Computing Machinery, New York, NY, USA, 312–315. doi:10.1145/3641308.3680518

[4] Tommaso Calò, Andrea Sillano, and Luigi De Russis. 2026. MorphGUI: Real-time GUIs customization with large language models. *International Journal of Human-Computer Studies* 208 (2026), 103695. doi:10.1016/j.ijhcs.2025.103695

[5] Yining Cao, Peiling Jiang, and Haijun Xia. 2025. Generative and Malleable User Interfaces with Generative and Evolving Task-Driven Data Model. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems (CHI '25)*. Association for Computing Machinery, New York, NY, USA, Article 686, 20 pages. doi:10.1145/3706598.3713285

[6] Victoria Clarke and Virginia Braun. 2017. Thematic analysis. *The Journal of Positive Psychology* 12, 3 (2017), 297–298. arXiv:https://doi.org/10.1080/17439760.2016.1262613 doi:10.1080/17439760.2016.1262613

[7] Upol Ehsan, Elizabeth A Watkins, Philipp Wintersberger, Carina Manger, Nina Hubig, Saiph Savage, Justin D. Weisz, and Andreas Riener. 2025. New Frontiers of Human-centered Explainable AI (HCXAI): Participatory Civic AI, Benchmarking LLMs, XAI Hallucinations, and Responsible AI Audits *(CHI EA '25)*. Association for Computing Machinery, New York, NY, USA, Article 793, 6 pages. doi:10.1145/3706599.3706713

[8] ElevenLabs. 2025. Scribe: Speech-to-Text. https://elevenlabs.io/speech-to-text Model: scribe_v1. Accessed: 2025-10-01.

[9] M. R. Endsley. 1995. Toward a Theory of Situation Awareness in Dynamic Systems. *Human Factors: The Journal of the Human Factors and Ergonomics Society* 37, 1 (1995), 32–64. doi:10.1518/001872095779049543

[10] European Commission. 2008. Commission Recommendation of 26 May 2008 on safe and efficient in-vehicle information and communication systems: update of the European Statement of Principles on human-machine interface. Official Journal of the European Union, L 216/1. https://eur-lex.europa.eu/eli/reco/2008/653/oj 2008/653/EC.

[11] Andrea Ferrario. 2025. Expanding HCXAI in the Age of AI Agents: Challenges and Recommendations. doi:10.5281/zenodo.15170379

[12] Google Cloud. 2025. Gemini 2.5 Pro. https://docs.cloud.google.com/vertex-ai/generative-ai/docs/models/gemini/2-5-pro. Last updated December 30, 2025.

[13] Iwona Grobelna, David Mailland, and Mikołaj Horwat. 2025. Design of Automotive HMI: New Challenges in Enhancing User Experience, Safety, and Security. *Applied Sciences* 10 (2025). doi:10.3390/app15105572

[14] Sandra G Hart. 2006. NASA-task load index (NASA-TLX); 20 years later. In *Proceedings of the human factors and ergonomics society annual meeting*, Vol. 50.

Sage publications Sage CA: Los Angeles, CA, 904–908.
[15] David S. Hermann and Sandhya Singh. 2020. User Experience and HMI Technologies for the Future. In *2020 27th International Workshop on Active-Matrix Flatpanel Displays and Devices (AM-FPD)*. 39–44. doi:10.23919/AM-FPD49417.2020.9224505
[16] Nobukatsu Hojo, Kazutoshi Shinoda, Yoshihiro Yamazaki, Keita Suzuki, Hiroaki Sugiyama, Kyosuke Nishida, and Kuniko Saito. 2025. GenerativeGUI: Dynamic GUI Generation Leveraging LLMs for Enhanced User Interaction on Chat Interfaces. In *Proceedings of the Extended Abstracts of the CHI Conference on Human Factors in Computing Systems (CHI EA '25)*. Association for Computing Machinery, New York, NY, USA, Article 306, 9 pages. doi:10.1145/3706599.3719743
[17] Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin, and Ting Liu. 2025. A Survey on Hallucination in Large Language Models: Principles, Taxonomy, Challenges, and Open Questions. *ACM Trans. Inf. Syst.* 43, 2, Article 42 (Jan. 2025), 55 pages. doi:10.1145/3703155
[18] Japan Automobile Manufacturers Association (JAMA). 2004. *Guidelines for In-Vehicle Display Systems – Version 3.0.* Japan Automobile Manufacturers Association (JAMA), Tokyo, Japan. https://www.jama.or.jp/operation/safety/display_system/pdf/jama_guidelines_v30_en.pdf Version 3.0. Available online at https://www.jama.or.jp/operation/safety/display_system/pdf/jama_guidelines_v30_en.pdf.
[19] Rafael Krstačić, Alesandro Žužić, and Tihomir Orehovački. 2024. Safety Aspects of In-Vehicle Infotainment Systems: A Systematic Literature Review from 2012 to 2023. *Electronics* 13, 13 (2024). doi:10.3390/electronics13132563
[20] Jonathan Lazar, Jinjuan Heidi Feng, and Harry Hochheiser. 2017. *Research methods in human-computer interaction* (2nd ed.). Morgan Kaufmann, Cambridge, Mass. https://shop.elsevier.com/books/research-methods-in-human-computer-interaction/lazar/978-0-12-805390-4
[21] Yaniv Leviathan, Dani Valevski, Matan Kalman, Danny Lumen, Eyal Segalis, Eyal Molad, Shlomi Pasternak, Vishnu Natchu, Valerie Nygaard, Srinivasan Venkatachary, James Manyika, and Yossi Matias. 2026. Generative UI: LLMs are Effective UI Generators. (2026). doi:10.48550/arXiv.2604.09577
[22] Siân Lindley, J. Williams, Y. Cao, H. Xia, Elizabeth F. Churchill, Abigail Sellen, Jeffrey Nichols, and David R. Karger. 2026. What Does Generative UI Mean for HCI Practice?. In *Extended Abstracts of the 2026 CHI Conference on Human Factors in Computing Systems (CHI EA '26)*. ACM. doi:10.1145/3772363.3778757
[23] Gerrit Meixner, Fabio Paternò, and Jean Vanderdonckt. 2011. Past, Present, and Future of Model-Based User Interface Development. *i-com* 10, 3 (2011), 2–11. doi:doi:10.1524/icom.2011.0026
[24] National Highway Traffic Safety Administration (NHTSA). 2013. *Visual-Manual NHTSA Driver Distraction Guidelines for In-Vehicle Electronic Devices.* U.S. Department of Transportation, National Highway Traffic Safety Administration, Washington, DC, USA. https://www.federalregister.gov/documents/2013/04/26/2013-09883/visual-manual-nhtsa-driver-distraction-guidelines-for-in-vehicle-electronic-devices Federal Register Notice No. 2013-09883, 78 FR 24818. Available online at https://www.federalregister.gov/documents/2013/04/26/2013-09883/visual-manual-nhtsa-driver-distraction-guidelines-for-in-vehicle-electronic-devices.
[25] Nielsen Norman Group. 2023. Heuristic Evaluation Workbook. https://media.nngroup.com/media/articles/attachments/Heuristic_Evaluation_Workbook_1_Fillable.pdf. Accessed: 2026-01-07.
[26] Nielsen Norman Group. 2025. *AI Hallucinations: What Designers Need to Know.* https://www.nngroup.com/articles/ai-hallucinations/ [Online; accessed 2025-08-09].
[27] Nielsen Norman Group. n.d.. How to Conduct a Heuristic Evaluation. https://www.nngroup.com/articles/how-to-conduct-a-heuristic-evaluation/ Accessed: 2026-01-07.
[28] Nielsen Norman Group. n.d.. How to Rate the Severity of Usability Problems. https://www.nngroup.com/articles/how-to-rate-the-severity-of-usability-problems/ Accessed: 2026-01-07.
[29] Pavel Okopnyi, Oda Elise Nordberg, and Frode Guribye. 2024. Against Generative UI. In *Proceedings of the Halfway to the Future Symposium* (Santa Cruz, CA, USA) *(HttF '24)*. Association for Computing Machinery, New York, NY, USA, Article 12, 4 pages. doi:10.1145/3686169.3686184
[30] OpenAI. 2025. GPT-5. https://openai.com/gpt-5/. Accessed 2026-02-25.
[31] OpenAI. 2025. GPT-5 Mini — API Models. https://developers.openai.com/api/docs/models/gpt-5-mini. Accessed 2026-02-25.
[32] Jakob Peintner, Carina Manger, Melanie Berger, and Andreas Riener. 2025. Balancing transparency and control: The impact of AI explanation detail on user perception in automated vehicles. *Transportation Research Interdisciplinary Perspectives* 34 (2025), 101742. doi:10.1016/j.trip.2025.101742
[33] Lena Rittger, Doreen Engelhardt, and Robert Schwartz. 2022. Adaptive User Experience in the Car—Levels of Adaptivity and Adaptive HMI Design. *Trans. Intell. Transport. Sys.* 23, 5 (May 2022), 4866–4876. doi:10.1109/TITS.2021.3124990
[34] Bartosz Sawicki, Tomasz Les, Dariusz Parzych, Aleksandra Wycisk-Ficek, Pawel Trebacz, and Pawel Zawadzki. 2026. Qualitative Evaluation of LLM-Designed GUI. arXiv:2601.22759 [cs.HC] https://arxiv.org/abs/2601.22759
[35] Martin Schrepp, Andreas Hinderks, and Thomaschewski Jörg. 2017. Design and Evaluation of a Short Version of the User Experience Questionnaire (UEQ-S). *International Journal of Interactive Multimedia and Artificial Intelligence* 4, 6 (Dec. 2017), 103–108. doi:10.9781/ijimai.2017.09.001
[36] Martin Schrepp and Jörg Thomaschewski. 2019. Construction and first validation of extension scales for the user experience questionnaire (UEQ). *Unpublished. https://doi. org/10.13140/RG* 2, 19260.08325 (2019).
[37] Wanderlust Travel Videos. 2020. Scenic Drive Autobahn (A3), No Speed Limit! - Germany [4K HDR] Driving Tour. https://www.youtube.com/watch?v=Hx44jrWHaUE.
[38] Marco Wiedner, Sreerag V Naveenachandran, Philipp Hallgarten, Satiyabooshan Murugaboopathy, and Emilio Frazzoli. 2024. CARSI II: A Context-Driven Intelligent User Interface. In *Adjunct Proceedings of the 16th International Conference on Automotive User Interfaces and Interactive Vehicular Applications* (Stanford, CA, USA) *(AutomotiveUI '24 Adjunct)*. Association for Computing Machinery, New York, NY, USA, 128–135. doi:10.1145/3641308.3685036
[39] L. Wirtz, E. Sever, and L. Eckstein. 2024. Adaptive Human-Machine Interfaces and Inclusivity in the Automotive Field: A Review. In *Accessibility, Assistive Technology and Digital Environments (AHFE Open Access, Vol. 121)*, Matteo Zallio (Ed.). AHFE International, USA. doi:10.54941/ahfe1004613

A Graphs Visualizing the Results for the Subjective Quantitative Measures

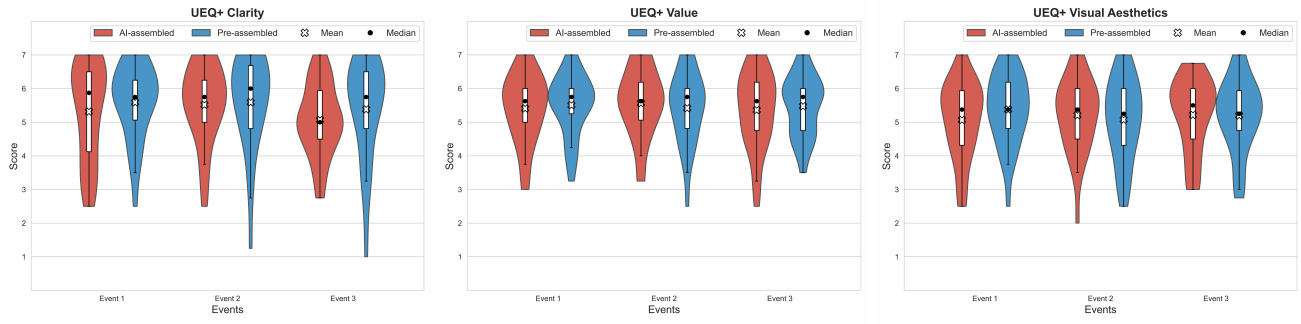


Figure 8: Violin charts illustrating the distribution of the user ratings on the UEQ+ scales for Clarity (left), Value (middle), and Aesthetics (right) per event for the AI-Assembled (red) and the Pre-Designed (blue) system.

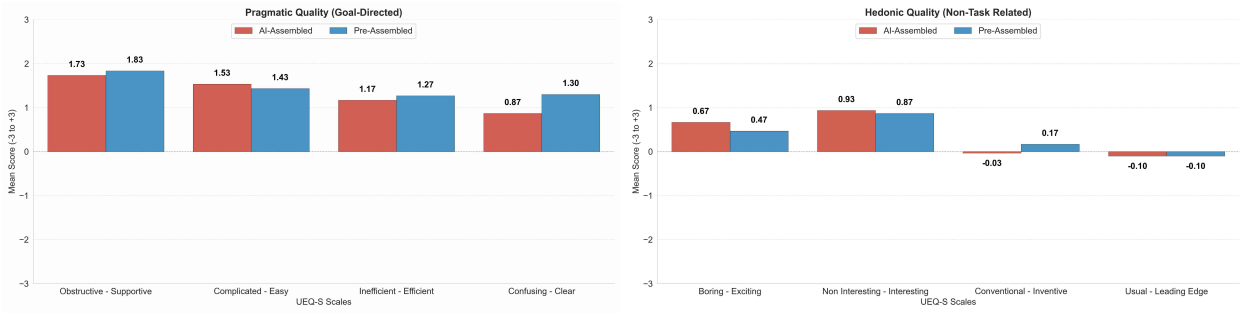


Figure 9: Bar charts illustrating the results for the pragmatic qualities (left) and the hedonic qualities (right) of UEQ-S questionnaire per event for the AI-Assembled (red) and the Pre-Designed (blue) system.

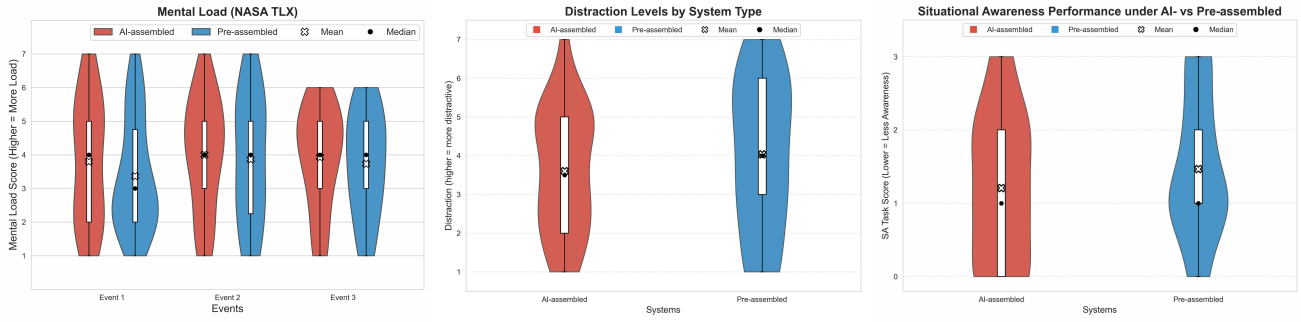


Figure 10: Violin charts illustrating the data distribution for our metrics focusing on user's state, in particular Mental Load (left, for all events), Distraction (middle, for Event 2), and Situational Awareness (right, for Event 2) for the AI-Assembled (red) and the Pre-Designed (blue) system.