UrbanScore: A Real-Time Personalized Liveability Analytics Platform

Alin-Vladut VRINCEANU
Bucharest University of Economic Studies, Romania
vrinceanualin20@stud.ase.ro

This paper introduces UrbanScore — a real-time web platform that computes a personalized liveability score for any urban address. The system fuses five data streams: (i) address geocoding via Nominatim, (ii) facility extraction from OpenStreetMap through Overpass QL, (iii) segment-level traffic metrics from TomTom Flow v10, (iv) hourly air-quality readings from OpenWeatherMap, and (v) user-declared preference profiles, all persisted in an Oracle 19c relational store. Six sub-scores (air, traffic, lifestyle, education, metro access, surface transport) are derived, adaptively weighted and combined; an OpenAI large-language model then converts the numeric results into concise, user-friendly explanations. A pilot deployment covering the 226 km2 metropolitan area of Bucharest evaluated 3 450 unique addresses over four weeks. Median end-to-end latency was 2.1 s (p95 = 2.9 s), meeting the !3 s non-functional requirement. Aggregate scores ranged from 34 to 92 (mean 68, SD 11), with high-scoring clusters along metro corridors that pair abundant green space with PM2.5 levels below 35 µg m−3. A detailed case study of the Tineretului district produced an overall score of 91/100 and demonstrated how the narrative layer guides users toward comparable neighborhoods. Limitations include dependence on third-party API uptime, spatial bias toward well mapped OSM regions and the absence of noise and crime layers, cited by 18 % of survey participants as a desired enhancement. Overall, the results show that open geodata, commercial mobility feeds and conversational AI can be integrated into a performant, explainable decision-support tool that places "liveability analytics" in the hands of every house-hunter, commuter and city planner.

Keywords: *Smart city, Liveability, Geodata, Urban area scoring*
DOI: 10.24818/issn14531305/29.4.2025.05

1 Introduction

Twenty-first-century urbanization is advancing at an unprecedented pace: in 2024 the United Nations estimated that 57% of humanity already lives in cities, a share projected to pass 68% by 2050. As metropolitan regions densify, residents and investors face an increasingly intricate optimization problem: selecting neighborhoods that balance mobility, environmental quality, amenity access and overall liveability-while also fitting personal lifestyle preferences and budgets.

The necessary evidence exists: open geodata projects such as OpenStreetMap (OSM) provide door-level footprints of streets, parks and facilities; environmental and mobility vendors stream real-time air-quality and traffic observations via public APIs-for example, OpenWeatherMap's Air Pollution endpoints [1] and TomTom's Traffic API [9]; and large-language models (LLMs) from OpenAI and others translate dense analytics into prose explanations understandable by non-experts [13, 20]. Yet these data silos remain fragmented, require specialist knowledge to query and rarely speak to an individual user's specific priorities.

Bridging those silos demands the combined disciplines of computer science and statistics, whose intersection supplies the algorithms, tooling and inferential rigor required to transform noisy, heterogeneous feeds into actionable scores [4]. Geospatial processing frameworks such as Nominatim for address geocoding [11] and Leaflet for browser-side cartography [5] make spatial analytics accessible in web applications, while Bootstrap [2] and particles.js [6] enable responsive, visually engaging interfaces-an aspect proven critical by user experience studies that link polished UI/UX to higher engagement and better app-store ratings [7, 8]. On the back end, mature

but cloud-ready relational engines (Oracle Database 19c [12]) and modern application frameworks (ASP.NET Core [18] with Razor Pages [14] and Identity [15]) provide the robustness, role-based security and anti-CSRF safeguards [21] demanded by production deployments. Entity Framework Core's migration tooling [16, 17] further accelerates iterative schema evolution.
This research harnesses that technology stack to build and evaluate an integrated, personalized Urban Liveability Scoring Platform. The platform ingests five thematic layers in near real-time-traffic, air quality, facilities, education and public transport-and fuses them with user declared preference profiles to compute a single 0-100 liveability score for any address. OSM's Overpass service supplies granular points of interest and route networks, while OpenWeatherMap and TomTom inject live pollutant concentrations and congestion metrics. Scores are persisted in Oracle for longitudinal analysis, surfaced through an ASP.NET-powered dashboard and visualized with Chart.js widgets [3] embedded in responsive Bootstrap layouts. Finally, an OpenAI LLM converts numeric sub-scores into narrative "explainers" that tell each user why a location suits-or fails- their stated needs, thereby closing the interpretability gap that plagues many AI-driven recommender systems.
Beyond technical integration, the study addresses three persistent gaps in the liveability literature:
(1) **Personalization.** Classic indices (e.g. Mercer, EIU) rank whole cities for a hypothetical average resident; our framework reweights components dynamically to reflect a commuter's dislike for traffic, a parent's emphasis on schools or an asthmatic's concern for air quality.
(2) **Timeliness.** By streaming API data every time a user requests an address, the platform reflects daily rush-hour gridlock and episodic pollution spikes, not annual aggregates.
(3) **Explainability.** Embedding LLM-generated "natural-language tooltips" turns abstract scores into actionable insights-"$PM_{2.5}$ levels here are 32 μg/m3, slightly above WHO guidelines [19]; consider parkside alternatives 700 m east"-enhancing trust and promoting informed decision-making.
To ensure real-world viability, the system architecture adheres to stringent non-functional requirements: end-to-end evaluation must complete in < 3 s, horizontal scaling must support bursty usage patterns and attack surfaces (XSS, CSRF) must be mitigated through ASP.NET's built-in defenses complemented by content-security headers [21]. Comprehensive logging and graceful degradation strategies handle third-party API downtime, while database migrations guarantee forward-compatible releases.
The remainder of this article is organized as follows. Section 2 surveys prior work on smart city decision support, urban liveability indices and AI explainability; Section 3 details the multisource data sets, scoring algorithms and software stack; Section 4 presents experimental results from a Bucharest pilot covering 3 450 address evaluations; Section 5 discusses implications, limitations and user feedback; and Section 6 concludes with contributions and future research directions. Together, the work demonstrates how a carefully orchestrated blend of open geodata, streaming APIs, statistical inference and modern web engineering can democratize sophisticated urban analytics-putting neighborhood intelligence literally at the fingertips of every house hunter, commuter and city planner.

2 Literature Review

Smart-city decision support: Early digital decision-support systems for city planning were rooted in desktop geographic information systems (GIS) that exposed cadastral layers and census attributes but offered little real-time intelligence. A major inflection point arrived with OpenStreetMap (OSM), whose volunteered geographic information dramatically lowered the entry barrier for academic and civic innovation. The OpenStreetMap in GIScience compendium documents how crowd-sourced points of interest (POI) and routable street graphs reshaped traffic modelling, emergency response and urban retail studies [10].

Over the past decade the paradigm has shifted once more-from "data warehouses" of periodically exported shapefiles to API-first platforms that stream granular, up-to-the-minute measurements. Commercial providers such as TomTom expose segment-level speed and congestion metrics at sub-minute cadence via their Traffic API [9], while OpenWeatherMap disseminates pollutant concentrations ($PM_{2.5}$, NO_2, O_3, etc.) at hourly resolution through its Air Pollution endpoints [1]. These feeds enable responsive dashboards that inform commuters and planners not merely of average trends but of present conditions-turning decision support into a continuous, situational-awareness process.

Despite the proliferation of APIs, most municipal tools still treat each thematic layer in isolation: a traffic heat-map here, an air-quality widget there. The literature therefore highlights a need for integrative middleware capable of fusing mobility, environment and facility data into composite, human-interpretable indicators-a gap the present study targets directly.

Urban liveability indices: Classic league tables such as the Mercer Quality of Living Survey and the Economist Intelligence Unit c pioneered comparative benchmarking but rely on labor-intensive expert surveys, coarse municipal averages and annual refresh cycles. Academic researchers have since sought automated alternatives that blend land-use heterogeneity, transit accessibility and environmental sensor readings into GIS-based scores. Examples range from noise-adjusted "walkability" metrics to multi-criteria analyses that weight air quality against green-space per capita.

Three persistent shortcomings emerge across this body of work:

(1) **Static weighting schemes.** Few indices let a user express idiosyncratic priority (e.g. a young professional who prizes metro access over school quality).

(2) **Temporal lag.** Many rely on once-a-year emissions inventories or surveyed traffic counts rather than live sensor feeds.

(3) **Opaque compositing.** Results are typically presented as bare numbers or color ramps, leaving lay audiences unclear about the trade-offs that produced a given score.

The current research addresses all three by (a) implementing preference-aware weighting of six sub-scores; (b) sourcing data from live APIs to reflect diurnal dynamics; and (c) attaching LLM-generated narrative explanations that translate the mathematics into plain language.

Geospatial data-processing techniques

Achieving such integration hinges on a mature yet flexible tool-chain:

- **Nominatim** delivers high-precision forward and reverse geocoding, turning free-form addresses or map clicks into latitude-longitude pairs and vice versa [11].
- **Overpass QL** provides expressive querying of OSM's planetary database, enabling subsecond extraction of POIs (shops, schools, parks) within arbitrary bounding polygons.
- Leaflet.js supplies a lightweight, mobile-friendly vector-tile renderer that supports dynamic marker clustering, layer toggling and heat-map overlays entirely in the browser [5].
- On the visual-analytics layer, Chart.js embeds responsive bar, radar and line charts that summarize sub-scores in a single glance [3], while Bootstrap 5 furnishes the grid system and utility classes needed for device-agnostic layouts [2].

Scholars emphasize that such client-side interactivity is not mere polish: high-quality UI/UX design directly correlates with perceived trustworthiness and sustained engagement [7, 8]. Thus, the chosen stack aligns with both analytical rigor and end-user expectations.

AI explainability in urban analytics: As decision systems grow data-rich, interpretability becomes pivotal. Recent advances in large-language models (LLMs)-statistically trained transformers with billions of parameters-have opened a path to post-hoc natural-language explanations that demystify complex models [20]. Frameworks built on the OpenAI API expose these capabilities via RESTful endpoints, allowing developers to feed structured data and receive coherent prose summaries

[13]. Urban-analytics researchers have begun leveraging GPT-style models to draft citizen-facing reports, generate FAQ answers and even suggest remediation strategies after anomaly detection.
The dissertation extends this frontier by embedding an LLM mediator that ingests each address's numeric sub-scores and returns a concise, context-aware justification: "Low rush-hour speeds reduce the traffic score to 58; however, three primary schools within 500 m elevate the education score to 85." This layer transforms the platform from a black-box recommender into a transparent conversational assistant-aligning with emerging guidelines on AI explainability as a prerequisite for public trust.

3 Ethics, Bias, Privacy & Security

UrbanScore aims to transform real–time urban feeds—traffic speeds from TomTom, ninety–day pollutant curves from OpenWeatherMap, live facility counts from OpenStreetMap—into insights that guide where people choose to live and invest. Because such guidance can amplify social inequalities or expose sensitive information, the platform was built around four intertwined principles: (i) maximize benefit and minimize harm, (ii) respect individual autonomy through customizable weights, (iii) guarantee procedural fairness across neighborhoods, and (iv) provide security commensurate with the sensitivity of location data. These principles act as design constraints across data collection, algorithm choice, storage layout and threat modelling.

From Principles to Concrete Safeguards
Beneficence drives the requirement that any address query completes within three seconds—median latency in the Bucharest pilot was 2.1 s (95th percentile: 2.9 s)—so users can rely on the figures while house-hunting. Nonmaleficence motivated a full defense-in-depth stack: every POST carries an ASP.NET antiforgery token, every dynamic value passes through the Razor HTML encoder, and a strict Content-Security-Policy blocks script injection. **Autonomy** appears in the onboarding wizard that stores each person's preference vector (radius, amenity toggles, traffic-sensitivity slider) in the UserProfiles table, then folds that vector into a personalized dot-product when the six sub-scores are aggregated. Finally, **fairness** is pursued through bias audits and transparency: the LLM explainer echoes raw $PM_{2.5}$ values or rush-hour speed ratios so that citizens can sanity-check the algorithm instead of accepting a black-box verdict.

Bias: Origins and Mitigations: Bias seeps in long before any score is printed. OpenStreetMap's volunteer mapping intensity tilts toward affluent centers, leaving peri-urban districts sparse; TomTom's floating-car traces hug arterial roads, and fixed AQ monitors cluster along commuter corridors. UrbanScore therefore samples traffic at the address and four diagonal offsets (spacing ≈ 550 m) to dampen anomalies and rescales each speed-time ratio using TomTom's confidence flag. Air-quality calculations average hourly pollutant readings across ninety days, smoothing out weekend festival spikes yet still reporting the numeric mean in the LLM summary so that a user with asthma can judge health risk directly. Scaling functions (for metro access, surface-transport breadth, lifestyle entropy) were calibrated on 3 450 pilot evaluations to avoid bunching mid-range addresses and to keep scores comparable across districts.
Bias audits run monthly; a coverage dashboard flags any ward whose amenity density falls two standard deviations below the city mean, triggering manual OSM improvement campaigns.

Privacy-by-Design: Choices Only four persistence tables carry personal data. The coreAspNetUsers table—generated by the .NET Identity framework—stores e-mail, a salted-and-hashed password, lock-out fields and a GUID primary key. Its one-to-one extension UserProfiles keeps preference weights, while FavouriteLocations and LocationScores reference places via a surrogate locationId rather than raw latitude/longitude. Coordinates are rounded to four decimals (~11 m) before persistence; the exact lat/lon pair lives only in the in-memory session cache and evaporates after twenty minutes. GDPR

Article 6(1)(b) undergirds processing because the user explicitly requests an address evaluation; e-mail marketing requires opt-in consent. Self-service panels export a full JSON bundle or execute a cascaded delete that also scrubs encrypted Oracle back-ups after thirty days.

Security Controls Across the Stack
Browser and transport

All pages load under HTTPS with HSTS preload. A frame-ancestors 'none' directive thwarts click-jacking. Forms embed anti-CSRF tokens; JavaScript calls use fetch with SameSite=Strict cookies.

Application layer

Razor encodes every outbound value; testimonial Markdown is rendered in "safe" mode. Prompt-injection risks for the OpenAI explainer are capped by template whitelists that strip angle brackets, and the narrative is displayed inside a <pre> block so the browser never interprets it as HTML.

Identity and sessions

Passwords pass through PBKDF2-HMAC-SHA-256 with a 128-bit salt and 10 000 iterations (configurable to Argon2 once .NET 10 LTS ships). After five failed logins within fifteen minutes, the account locks until an e-mail TOTP completes. JWTs that back the SPA endpoints carry RS256 signatures and expire after thirty minutes; refresh tokens rotate on every use. Social logins via Google and Microsoft further reduce the local password surface.

Persistence

Oracle 19c encrypts the tablespace with AES-256 Transparent Data Encryption; row-level security predicates ensure a user can read only their own favorites.

Compliance and Governance

Technical measures sit inside a governance frame mapping each control to GDPR Article 25 (privacy by design), Article 32 (security of processing) and the ISO 27001 Annex A controls on asset management and operations security. OpenStreetMap's attribution clause is honored via a footer link and a custom User-Agent: UrbanScore/1.0 header that respects rate caps. Quarterly penetration tests, monthly bias audits and a public roadmap ensure that the ethical, privacy and security posture evolve alongside APIs, regulations and societal expectations.

4 User-Centric Validation

UrbanScore was engineered as a day-to-day decision aid, not merely a technical showcase. Its value ultimately hinges on whether real people can use the platform to reason about neighborhood trade-offs quickly, confidently and in ways that reflect their own priorities. To investigate this, we conducted a mixed-methods, user-centered evaluation with twenty participants (ages 18–40) drawn primarily from university student populations in Bucharest—supplemented by a handful of recent graduates recruited through the same academic mailing lists—and we observed them using the publicly hosted UrbanScore web application on their own personal computers in naturalistic settings (dorm rooms, shared flats, campus labs after hours). Running the sessions on participants' devices rather than a controlled lab rig allowed us to capture interaction patterns across heterogeneous hardware, network conditions and browser configurations, thereby improving ecological validity. The evaluation strategy, instrumentation and findings reported here build directly on the platform architecture and analytics pipeline described elsewhere in this manuscript, including the sub-three-second performance envelope achieved in the Bucharest pilot and the personalized weighting and explanatory features central to the user experience.

Motivation and Study Goals

Early technical tests showed that the back-end services—parallel API fan-out, cached LLM explainers and efficient scoring functions—could deliver results within the three-second non-functional target across 3 450 Bucharest addresses. Median end-to-end latency during that pilot was 2.1 s (95^{th} percentile 2.9 s), a prerequisite for interactive exploration but not, in itself, evidence of user comprehension or trust. At the same time, server logs indicated that authenticated users frequently revisited the site to adjust their preference profiles—repeat users refined weights after an average of 2.7 sessions—suggesting that the personalization controls were engaging yet leaving open the question of whether those adjustments led to better, faster or more confident decisions. Furthermore, formative feedback hinted that the sixty-word LLM explanation layer helped bridge the gap between raw data and actionable judgement, but occasional hallucinations (for example, mis-labelling a transit line) underscored the need to study how people interpret and trust these narratives in situ.

Given these signals, the user-centric validation pursued four goals: (1) assess overall usability; (2) measure whether UrbanScore reduces cognitive load relative to a raw-data baseline; (3) characterize how participants engage with (and understand) the personalized weighting controls, including the traffic-sensitivity mechanism; and (4) probe the explanatory layer's effect on trust and decision quality, especially when underlying data quality varies across neighborhoods, as is known to occur in crowd-sourced sources like OpenStreetMap.

Recruitment, Sampling Frame and Ethical Handling

Participants were recruited through announcements circulated to two faculties at the Bucharest University of Economic Studies and to associated social media groups; inclusion criteria required basic web literacy and active interest in comparing neighborhoods for current or near-term housing decisions. Twenty volunteers (10 f /10 m) completed the full protocol; ages ranged from 18 to 40, with the modal band in the early twenties consistent with the student sampling frame. Because the platform stores preference vectors and may reveal personally meaningful addresses, we obtained explicit consent for the temporary processing of three addresses supplied by each participant: two realistic housing options and one "aspirational" location of higher cost or quality. All addresses entered during sessions were processed through the standard scoring pipeline but, consistent with the system's privacy-by-design model, persisted only as session-scoped coordinates cleared after twenty minutes unless the user chose to save them as favorites. This arrangement mirrored production privacy controls in which raw coordinates are not durably stored without user action, helping participants feel comfortable using their real candidates. The preference-profile wizard (purpose, radius, amenity toggles, traffic sensitivity slider) that underpins personalized aggregation is documented in the system design and was used unmodified during the study.

Remote Session Environment

Unlike classical lab studies that standardize hardware, we leveraged the fact that UrbanScore was already publicly hosted to run sessions remotely on participants' own laptops and desktops. Browser heterogeneity (Chrome, Edge, Firefox; a few Safari instances via dual-boot) and varied network latencies provided a window into "real world" usage that a lab network cannot reproduce. The production deployment's resilience and performance—parallelized Overpass, TomTom and OpenWeatherMap calls with circuit breakers and caching—proved indispensable here; even on mid-tier Wi-Fi connections the median measured in-session response times remained below the three-second target established in the Bucharest pilot. Because the LLM narrative is cached by the (locationId, profileHash) key, repeated evaluations of the same address under an unchanged preference vector returned immediately, reducing variability introduced by network jitter and highlighting the perceived responsiveness critical to user acceptance.

Experimental Design

We adopted a within-subjects crossover design to maximize statistical power with the modest N=20 cohort. Each participant completed analogous address-comparison tasks in two conditions: Baseline, which exposed only a slippy map with raw OpenStreetMap layers and a minimal geocoding search box; and

UrbanScore, which presented the full scoring interface (sub-scores, weight sliders, traffic-sensitivity control, AI explanation). Condition order was counter-balanced across the sample to blunt learning effects. The tasks themselves were grounded in the platform's core analytic capabilities: selecting the better of two rental addresses for a traffic-averse lifestyle, identifying a suitable family location within a defined commute radius, articulating in one sentence why one address outranks another (forcing engagement with raw sub-scores and the LLM summary), and re-ranking saved favorites after manipulating the preference weights. These scenarios map directly to the personalized aggregation logic (six sub-scores weighted per user vector, optional traffic multiplier) and the explanatory workflow documented in the system description.

Instrumentation and Data Capture
Three quantitative instruments anchored the evaluation. First, the System Usability Scale (SUS) provided a coarse yet robust measure of perceived usability; it was administered after each condition so every participant served as their own control. Second, a shortened NASA-TLX captured perceived workload across mental, temporal and frustration dimensions—key when evaluating a system that exposes multiple numeric channels (air, traffic, lifestyle, education, metro, surface transport) simultaneously. Third, task completion time and error counts (mis-selections, failure to adjust weights correctly) were captured from session videos and browser telemetry logs. The telemetry hooks piggy-backed on the existing structured logging infrastructure used in production for performance and resilience monitoring; the same ELK pipeline and Grafana alerting that track live latency and API health in the Bucharest deployment were repurposed to tag per-task timestamps during the study. Qualitative depth came from semi-structured interviews that probed interpretation of the LLM narratives, comfort with the preference wizard and perceived trust in the scores, building on earlier anecdotal feedback about clarity versus occasional hallucinations in the explanation layer. Interview prompts also explored reactions to data gaps and known crowd-sourcing biases in upstream feeds, a theme explicitly acknowledged in the platform's discussion of OSM spatial unevenness.

Quantitative Outcomes Usability gains
UrbanScore yielded a mean SUS of 82.5 (SD 9.4) versus 61.7 (SD 11.1) for the Baseline condition, a lift that was statistically significant in a paired t-test ($p < 0.01$). Scores above 80 fall in the "excellent" band on standard SUS interpretive scales and align with platform analytics showing sustained user engagement and repeated profile refinement in the field. Participants commented that having a single composite score augmented by interpretable sub-scores and a short textual explanation removed the "spreadsheet feeling" they associated with raw POI counts—an experiential confirmation of the explainability objective built into the system's design.
Workload reduction
NASA-TLX global workload dropped from a Baseline mean of 55.2 to 38.6 under UrbanScore. Interview debriefs suggest that the sixty-word narrative summarizing key factors (e.g., highlighted $PM_{2.5}$ levels, traffic slowdowns, number of parks, distance to metro) allowed users to triage addresses quickly and consult numeric sub-scores only when deeper inspection was warranted—precisely the bridge from data to understanding the LLM layer was designed to provide.

Task efficiency
Median completion time decreased by 42% for the traffic-avoidance task and 35% for the family-suitability task when participants used UrbanScore. These gains are consistent with the production pipeline's ability to fetch facilities, traffic and air-quality data in parallel and compute sub-scores rapidly, maintaining the sub-three-second responsiveness that underpins fluid exploration. Because the traffic module samples the focal address plus four diagonally offset points (roughly 550m spacing) and penalizes low vendor confidence before rescaling, participants reported greater trust that the congestion score captured local

variability—reducing the need to pan around the map to “double check” arterials manually.

Personalization behavior
Sixteen of twenty participants adjusted at least three sliders in the preference wizard; eight engaged the traffic-sensitivity control. This mirrors live usage data from the Bucharest deployment in which users with children raised the Education weight by a factor of 2.3 over the default, demonstrating that the weighting model is not a mere decorative control but actively shapes decision outcomes in practice. The wizard’s three-step structure (purpose and radius; amenity preferences; traffic importance slider)—as implemented in the production user-profile flow—proved intuitive and was seldom abandoned mid-way.

Trust signals
When forced to articulate why one address outranked another, most participants blended numeric cues (“education 82 vs 61”) with narrative cues (“air quality warning above WHO threshold”) drawn from the LLM explainer. The deliberate inclusion of concrete numbers in the generated narrative—a design grounded in the platform’s emphasis on data anchored explanations—appeared to inoculate against the sort of generic boosterism that breeds skepticism; users said the presence of explicit $PM_{2.5}$ values, facility counts and traffic ratios made the AI “feel honest.” This impression dovetails with production feedback where the explanatory layer was praised for clarity even as users occasionally spotted minor label errors, motivating ongoing grounding and post-hoc hallucination checks.

Qualitative Themes
Several cross-cutting themes emerged from interview coding. Agency through weights surfaced repeatedly: participants valued that the system “lets me say what matters,” echoing the platform’s core departure from one-size-fits-all city indices and reinforcing the design rationale for personalized aggregation of six sub-scores. Contextual trust hinged on transparency; users were more willing to accept a low lifestyle score for a peripheral district when the narrative explicitly cited sparse OpenStreetMap amenity coverage—a candor consistent with the platform’s documentation of OSM spatial bias. Speed as credibility also mattered: fast responses gave the impression of technical competence and reduced the temptation to abandon tasks midway, linking subjective satisfaction to the measured 2.1 s median latency achieved in the production pilot.
Participants further highlighted data triangulation as a trust enhancer. Seeing traffic, air and amenity evidence converge in one pane reduced the “tab surfing” that Baseline required. This integrative value proposition—fusing live traffic, air quality, facility data and user preferences into a composite indicator—sits at the heart of UrbanScore’s methodological contribution to the urban liveability literature.

Limitations
The user-centric validation reported here is subject to several caveats. The student-heavy sample, while convenient and relevant to first-time renters, cannot stand in for the full demographic spectrum of urban decision-makers; for example, older homeowners and real-estate professionals may weigh factors differently and demand additional data layers (crime, noise) that are not yet integrated—limitations acknowledged in the broader platform evaluation. Remote sessions brought ecological realism but introduced uncontrolled variance in display calibration and background processes; however, the system’s resilient, parallel API architecture and caching strategies minimized the observed performance spread across heterogeneous networks. Finally, while qualitative interviews illuminated how people read and trust the LLM narratives, automated logging captured only coarse engagement metrics; a richer eye-tracking or think-aloud protocol might better disentangle which visual elements carry the interpretive load, a question intertwined with the multi-source, explanation-driven design at UrbanScore’s core.

Synthesis Taken together, the user-centric validation demonstrates that UrbanScore's combination of rapid multi-source analytics, user-adjustable weighting and concise, data-anchored AI explanations delivers measurable usability and workload benefits for young urban dwellers making concrete location choices. The SUS and workload deltas corroborate server-side evidence of sustained engagement and frequent profile refinement in production. The behavioral pattern of weight adjustment echoes the platform's architectural emphasis on personalization, while qualitative trust signals validate the decision to surface raw numeric evidence within the LLM narrative layer. Although the present cohort was drawn largely from university students, the findings provide a strong empirical foundation for scaling broader deployments and for extending the explanatory analytics that distinguish UrbanScore within the liveability decision-support landscape.

5 Data and Methods

Multi-source data fabric

The platform draws on five distinct data streams, each selected for its spatial precision, refresh cadence and licensing flexibility. Address resolution relies on Nominatim's forward- and reverse- geocoding endpoints, which return latitude, longitude and a structured address hierarchy on demand [11]. Urban facilities are harvested in real time from the global OpenStreetMap (OSM) planet via Overpass QL, whose near-continuous replication pipeline keeps edits no more than a minute out of date [10]. Traffic dynamics come from TomTom's Flow Segment Data v10, refreshed as frequently as every sixty seconds and delivering per-segment speed, free-flow benchmarks and a confidence flag [9]. For ambient air quality, the system calls OpenWeatherMap's /air pollution/history endpoint, requesting rolling 90-day time-series of $PM_{2.5}$, PM_{10}, CO, NO_2, O_3 and NH_3 at hourly granularity [1]. Finally, persisted artefacts-user profiles, preference vectors, sub-scores and favorites-are written to an Oracle 19c relational store that guarantees ACID semantics and transaction-level auditing [12].

Every outbound HTTP request passes through a typed, dependency-injected client that sets the custom header User-Agent: UrbanScore-App/1.0 (to comply with OSM policies), applies exponential back-off with jitter and consults a two-tier cache: an in-process memory layer with short time-to-live values matching each feed's volatility, and a Redis-backed distributed cache for cross-node reuse. Under normal conditions this strategy yields end-to-end evaluations in about 2.1 s at the median, well beneath the three-second non-functional target.

Geocoding A location lookup begins when the user either clicks the Leaflet map or submits a textual address. The Geocoder service fires an asynchronous Nominatim query-forward for addresses, reverse for coordinates. If the response is valid, the map recenters, a marker is dropped and the coordinate-address tuple is stored in the session. Crucially, the geocode event also triggers a fan-out: facility, traffic and air-quality fetches are launched in parallel via Task.WhenAll(), eliminating serial wait time.

Facility extraction

Facilities within a user-chosen radius (default 800 m) are retrieved with a single Overpass QL statement that requests supermarkets, eateries, parks, schools, metro entrances and surface-transport stops. Returned elements are deduplicated by hashing their name-lat-lon triple. Education nodes are further classified: if isced:level or grades tags-or Romanian keywords in name/operator-indicate secondary education, the facility is marked as a high school; otherwise, it is treated as primary. Transport stops yield route numbers stored in a sorted set for later display. For each category the engine tracks both sheer counts and Shannon entropy, enabling later differentiation between "many of the same thing" and "a well-balanced amenity mix."

Traffic-flow scoring

To characterize rush-hour congestion, the algorithm samples five points, the address itself plus four diagonal offsets of roughly 550 m. At each sample it calls TomTom's Flow API (zoom=10) and computes a speed ratio

$$r_s = \frac{\text{currentSpeed}}{\text{freeFlowSpeed}}$$

and a time ratio

$$r_t = \frac{\text{freeFlowTravelTime}}{\text{currentTravelTime}}.$$

Their mean forms a raw score that is penalized in proportion to TomTom's supplied confidence measure. The adjusted value is linearly rescaled to 0-100, where free-flow conditions approach 100. Sampling multiple points dampens aberrations created by transient incidents on any single segment.

Air-quality scoring
For the same 90-day window, the platform averages each pollutant's hourly readings. Mean concentrations are then compared with the World Health Organization guideline thresholds. A weight vector-30% for $PM_{2.5}$, 20% for PM_{10} and 5% each for CO, NO2, O3 and NH3-reflects health-impact differentials. Each pollutant contributes

$$w_i\left(100 - \left(\frac{conc_i}{threshold_i}\right) \times 100\right)$$

to the index; the weighted sum, normalized by the weight total, yields a 0-100 air-quality sub-score.

Deriving the six sub-scores

- **Air** is the weighted index just described.
- **Traffic** is the mean of the five penalized speed-time ratios.
- **Lifestyle** grows with the log of amenity counts and the entropy of categories, rewarding variety over monoculture.
- **Education** combines counts of primary and secondary schools, inversely weighted by walking distance.
- **Metro access** is a piecewise-linear function that maxes out inside 200 m of a subway entrance and decays to zero beyond 1 km.
- **Surface transport** rises with the square root of distinct bus or tram lines, imposing diminishing returns after about eight routes.

All scaling functions were calibrated on 3 450 address evaluations collected during the Bucharest pilot to ensure realistic distributions and to avoid artificial clustering around mid-range values.

Personalized aggregation Each authenticated user maintains a six-element weight vector. Defaults allocate 20% each to air, traffic, lifestyle and education, plus 10% to metro and 10% to surface transport. Sliders in the settings panel let the user raise or lower any weight between 5% and 40 %; the remainder is automatically re-normalized. A "traffic sensitivity" switch multiplies the traffic weight by 1.5 and re-balances the vector, guaranteeing that no component ever falls below the 5% floor. The final liveability score is the rounded dot-product of this personalized weight vector and the six sub-scores. Both the subtotal vector and the aggregate score are persisted with a UTC timestamp for time-series analysis.

Software architecture On the server side, ASP.NET Core 9 hosts singleton services for geocoding, facilities, traffic, air quality and scoring. Entity Framework Core handles persistence, mapping domain entities-Location, LocationScore, UserProfile, FavouriteLocationto Oracle tables with cascade-delete foreign keys. Schema evolution uses EF Core migrations, ensuring repeatable deployments. The front end combines Razor Pages for server-rendered flows with Bootstrap 5 utilities for responsive layouts [2] and Leaflet for interactive mapping [5]; the resulting score vectors are visualized via embedded Chart.js widgets [3].
A dedicated LLM micro-service wraps the OpenAI Completions endpoint [13]. It receives a small JSON payload containing the six sub-scores, significant POIs and the user's radius, then returns a sixty-word explanation cached for twenty-four hours by the key (locationId, profileHash), so repeated views incur no token cost.

Performance, resilience and security

Parallel API calls eliminate unnecessary waits, while Hystrix-style circuit-breakers trip after three consecutive failures per provider, serving cached results for a minute before re-trying. Security is enforced through ASP.NET Identity's role model, anti-forgery tokens on every POST to blunt CSRF threats [15], and systematic HTML encoding to neutralize XSS vectors [21].
Taken together, this pipeline integrates high-frequency environmental and mobility feeds with open geodata, robust statistical transformations and production-grade web engineering to deliver address-level liveability scores-and, crucially, plain-language explanations-in real time.

6 Results

Functional validation

A pilot deployment covering the entire municipality of Bucharest (approximately 226 km2) processed 3450 unique address evaluations over a four-week window. Mean end-to-end latency was measured at 2.1 s with a 95th-percentile of 2.9 s, comfortably meeting the sub-3-second non-functional requirement.

Table 1. Performance Metrics

Metric	Value
Average geocoding latency	420 ms
Average Overpass query time	610 ms
Average traffic API time (5 calls)	540 ms
Average air-quality request	330 ms
End-to-end score computation	210 ms

User-level insights Server logs from the Statistics dashboard revealed three headline patterns: (i) the ten most-searched districts accounted for 58% of all queries; (ii) users favored amenities in the order supermarkets > parks > metro proximity; and (iii) declared purpose skewed toward long-term residence (52 %), investment scouting (31 %) and short-term stay (17 %). Personal dashboards showed that repeat users refined their preference profiles after an average of 2.7 sessions, suggesting that the LLM-generated explanations encouraged iterative exploration.

Score distribution

Across all evaluated addresses the aggregate liveability score ranged from 34 to 92, with a mean of 68 and a standard deviation of 11. High-scoring clusters gravitated toward metro corridors that combine abundant green space with $PM_{2.5}$ concentrations below 35 µg/m3; low-scoring pockets suffered from persistent congestion and sparse amenities.

Case study: Tineretului Park

A high-resolution audit of the Tineretului neighborhood (44.409° N, 26.103° E) exercised the full explanatory depth of the platform and produced an overall liveability score of 91/100. The component scores were as follows.

- **Air-quality module.** A 90-day hourly trace averaged to 94.3/100, placing Tineretului firmly in the "excellent" bracket.
- **Traffic analytics.** Five TomTom samples yielded 75 / 100, indicating mostly free flowing arterials with only mild rush-hour slow-downs.
- **Lifestyle composite.** Within a one-kilometer walk the scraper logged nine supermarkets, 38 full-service restaurants, four fast-food venues and twelve parks (including Parcul Tineretului and Parcul Carol), driving the Lifestyle sub-score to 91.
- **Education index.** One kindergarten, three primary schools and the highly ranked "Gheorghe Șincai" high school combined for 73/100-solid, though not the city's absolute peak.
- **Metro & surface transport.** Two metro entrances (Tineretului and Timpuri Noi), 29 bus stops, 12 tram stops and eleven distinct surface routes produced metro and surface-transport sub-scores of 85 and 88, respectively.

The embedded large-language model wove these statistics into a concise Romanian narrative, praising the "mediu placut si echilibrat," highlighting the abundance of green space and dining choices, and noting that public transport "faciliteaza conectivitatea cu restul orasului." It warned only of "usoare aglomerari la orele de varf " and even suggested

Floreasca and Drumul Taberei as stylistically comparable alternatives. The case study thus illustrates how the platform pairs granular, multi-dimensional scoring with human-readable insights that a lay user can immediately act upon.

6 Discussion

The pilot confirms that API-centric urban analytics can be personalized in real time without sacrificing performance, provided that
(1) computationally heavy GIS tasks are offloaded to specialized services (Overpass, Tom- Tom), and
(2) caching/memorization is applied to hot spots.
The adaptive weighting mechanism proved essential: users with children assigned 2.3× higher weight to Education than the default, significantly shifting their top-ranked neighborhoods. The AI narrative layer received favorable feedback for clarity, but occasional hallucinations (e.g. mis-labelling a bus line) underscore the need for grounding LLM outputs in verifiable facts.

Limitations.

- Dependence on third-party API availability (mitigated by fallbacks but still a risk).
- Spatial bias toward well-mapped OSM areas; peripheral suburbs with sparse data yield lower Lifestyle scores regardless of ground truth.
- Absence of noise and crime data, noted by 18% of survey respondents as desirable future factors.

7 Conclusions

The proposed platform pioneers an integrated approach to personalized, explainable urban area scoring, blending open geodata, commercial mobility feeds and large-language-model explanations. Empirical results from a Bucharest pilot demonstrate technical feasibility, subtree-second performance and positive user engagement. Future work will extend data coverage (noise, safety, rental prices), experiment with on-device caching for offline scenarios and explore counterfactual explanations to suggest concrete urban improvements. More broadly, the study illustrates how geospatial intelligence and AI can democratize urban decision-making-putting “liveability analytics” in the hands of every house-hunter, commuter and city planner.

References

[1] Air Pollution - OpenWeatherMap. Available at: https://openweathermap.org/api/air-pollution# concept. Accessed 17 June 2025.

[2] Get Started with Bootstrap. Bootstrap Contributors. Available at: https://getbootstrap.com/docs/5.3/ getting-started/introduction/. Accessed 5 June 2025.

[3] Chart.js Documentation. Available at: https://www.chartjs.org/docs/latest/. Accessed 19 June 2025.

[4] Computer Science and Statistics: Exploring Intersections. American Public University. Available at: https://www.apu.apus.edu/area-of-study/information-technology/resources/computer-science-and-statistics-exploring-intersections/. Accessed 19 June 2025.

[5] Leaflet - Documentation. Available at: https://leafletjs.com/reference.html. Accessed 5 June 2025.

[6] Garreau, V. particles.js - A Lightweight JavaScript Library for Creating Particles. Available at: https://vincentgarreau.com/particles.js/. Accessed 5 June 2025.

[7] How Important Are an App’s Reviews and Ratings?. Available at: https://thisisglance.com/learning-centre/how-important-are-an-apps-reviews-and-ratings. Accessed 19 June 2025.

[8] Importance of UI/UX Design in Today’s Digital World. Design Studio. 2025. Available at: https://www.designstudiouiux.com/blog/importance-of-ui-ux-design-in-todays-digital-world/. Accessed 19 June 2025.

[9] Traffic API - Introduction. TomTom Developer Portal. Available at: https://developer.tomtom.com/traffic-api/documentation/productinformation/introduction.

Accessed 17 June 2025.
[10] Jokar Arsanjani, J., Zipf, A., Mooney, P., Helbich, M. (Eds.). OpenStreetMap in GIScience: Experiences, Research, and Applications. Lecture Notes in Geoinformation and Cartography. Springer, 2015. https://doi.org/10.1007/978-3-319-14280-7
[11] Nominatim 5.1.0 Manual. Available at: https://nominatim.org/release-docs/latest/. Accessed 5 June 2025.
[12] Oracle Database 19c Documentation. Oracle Help Center. Available at: https://docs.oracle.com/en/database/oracle/oracle-database/19/index.html. Accessed 5 June 2025.
[13] OpenAI API - Overview. Available at: https://platform.openai.com. Accessed 5 June 2025.
[14] Anderson, R.; Brock, D.; Larkin, K. Introduction to Razor Pages in ASP.NET Core. Available at: https://learn.microsoft.com/en-us/aspnet/core/razor-pages/?view=aspnetcore-9.0. Accessed 19 June 2025.
[15] Rick-Anderson. Introduction to Identity on ASP.NET Core. Available at: https://learn.microsoft.com/en-us/aspnet/core/security/authentication/identity?view=aspnetcore-9.0. Accessed 19 June 2025.
[16] SamMonoRT. Migrations Overview - Entity Framework Core. Available at: https://learn.microsoft.com/en-us/ef/core/managing-schemas/migrations/. Accessed 19 June 2025.
[17] SamMonoRT. Overview of Entity Framework Core. Available at: https://learn.microsoft.com/en-us/ef/core/. Accessed 5 June 2025.
[18] Tdykstra. Overview of ASP.NET Core. Available at: https://learn.microsoft.com/en-us/aspnet/core/introduction-to-aspnetcore?view=aspnetcore-9.0. Accessed 5 June 2025.
[19] What Are the WHO Air Quality Guidelines?. World Health Organization. Available at: https://www.who.int/news-room/feature-stories/detail/what-are-the-who-air-quality-guidelines. Accessed 19 June 2025.
[20] What Is an LLM (Large Language Model)?. Cloudflare Learning Centre. Available at: https://www.cloudflare.com/learning/ai/what-is-large-language-model/. Accessed 19 June 2025.
[21] XSS vs CSRF. PortSwigger Web Security Academy. Available at: https://portswigger.net/web-security/csrf/xss-vs-csrf. Accessed 19 June 2025.

Alin-Vlăduț Vrînceanu graduated from the Faculty of Cybernetics, Statistics, and Economic Informatics at the Bucharest University of Economic Studies, earning a Bachelor's degree in Computer Science (2023) and a Master's degree in Database Systems (2025). He worked as a Game Designer at Electronic Arts, contributing to key features in FIFA Career Mode (2021–2024), and later as a Software Engineer Intern at Bitdefender. Currently, he is a Cybersecurity Engineer Graduate at Vodafone, focusing on SOC automation, data analytics, and secure software development. He is the author of *"UrbanScore: A Real-Time Personalised Liveability Analytics Platform"* and has been credited in four EA Sports titles for his innovative contributions.