

Article

TwinCity: An Urban Digital Twin Framework for Data-Scarce Environments—A Case Study of Benguerir, Morocco

Ouzougarh Badreddine ^{1,2} , Hassan Radoine ³  and Rafika Hajji ^{1,*} 

¹ College of Geomatic Sciences and Surveying Engineering, Hassan II Institute of Agronomy and Veterinary Medicine, Rabat 10101, Morocco; badred-edine.ouzougarh@hcu-hamburg.de

² Computational Methods Lab, Hafencity University Hamburg, 20457 Hamburg, Germany

³ CITINNOV for Integrated Territorial Planning and Smart Cities, Mohammed VI Polytechnic University (UM6P), Campus of Mohammed VI Polytechnic University Rabat, Rabat 11103, Morocco; hassan.radoine@um6p.ma

* Correspondence: r.hajji@iav.ac.ma

Highlights

What are the main findings?

- Identification and analysis of the gap between theoretical definitions and real-world implementations of Urban Digital Twins, including technical and operational limitations.
- Investigation of whether an Urban Digital Twin can be developed in data-scarce environments, and exploration of mitigation strategies to enable appropriate use-case selection under limited data conditions.

What is the implication of the main findings?

- Improved understanding of data interoperability challenges and development constraints in Urban Digital Twin systems.
- Guidance for adopting Urban Digital Twin solutions to address real urban challenges, even where data availability is limited.

Abstract

Urban Digital Twins (UDTs) are emerging as a new paradigm in smart city strategies, enabling real-time interaction with urban environments and supporting data-driven decision-making. By expanding beyond traditional smart functions, UDTs facilitate the analysis and simulation of urban resilience and sustainability indicators within a virtual city ecosystem, addressing both immediate urban challenges and long-term planning goals. This paper introduces TwinCity, a city-scale Urban Digital Twin framework developed and validated through a case study of the Green City of Benguerir, Morocco. The framework incorporates a technical architecture based on semantic 3D city models, data integration, and simulation scenarios to analyse the solar energy potential of the rooftop, the energy consumption of the building and the morphological indicators. A user-friendly web interface was developed to visualise and interact with the UDT, ensuring its accessibility. By bridging the gap between technical challenges (such as data scarcity) and practical applications, this work offers a replicable model for cities in the Global South.

Keywords: Urban Digital Twins; semantic 3D city models; smart city; sustainability



Academic Editors: Pierluigi Siano, Songnian Li and Zhen Chen

Received: 20 October 2025

Revised: 6 January 2026

Accepted: 7 January 2026

Published: 26 January 2026

Copyright: © 2026 by the authors.

Licensee MDPI, Basel, Switzerland.

This article is an open access article distributed under the terms and conditions of the [Creative Commons Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

1. Introduction

Rapid and unprecedented urbanisation is intensifying the challenges faced by cities. Across the African continent, various urban centres are struggling with inadequate infras-

structure, and poorly planned facilities, which in turn exacerbate the impacts of climate change and environmental degradation. In response to these complex urban issues, UDT have emerged as essential tools for urban management and planning. By providing a dynamic and data-driven representation of the urban environment, UDTs enable city decision-makers to make informed decisions in governance and infrastructure development [1].

As a result, several cities worldwide have initiated the development of their own digital twins. Notable examples include Zurich, which provides a digital twin platform that allows decision-makers to interactively explore and evaluate planning strategies [2]; the Helsinki-Espoo metropolitan area in Finland explores various energy management applications [3]; Virtual Singapore in Malaysia [4]; and 3D Amsterdam and Delft in the Netherlands [5], and Munich, Melbourne, and others.

A validated semantic 3D city model is considered the foundational element of an Urban Digital Twin (UDT), providing a comprehensive spatial representation of urban elements and enabling advanced spatial analysis and simulations [6]. However, the development of UDTs is still in its early stages, due to challenges such as the lack of comprehensive urban data standards, the high cost of data acquisition and management, and the need for software capable of supporting the full UDT lifecycle [7,8]. To address these issues, recent research has proposed frameworks for data integration, categorizing approaches into model extension, data transformation, and front-end integration aligned with UDT lifecycle phases [9]. In parallel, emerging studies explore the use of Large Language Models (LLMs) to automatically enrich and query CityGML models, facilitating the incorporation of heterogeneous data sources without requiring extensive domain expertise [10]. Furthermore, agentic AI has been proposed to dynamically generate, execute, and adapt urban workflows within UDTs, enabling autonomous, knowledge-augmented management of complex urban systems [11]. While the research community is actively developing open-source solutions and standards to overcome these challenges, currently available approaches remain limited.

This study presents a case example of creating an Urban Digital Twin for Benguerir city in Morocco, built around the new “Green City project”, which is a national pilot project that emerges as a new urban model built on advanced smart infrastructure designed to realise the principles of resilience, sustainability, and innovation (SADV Green City Report). For city decision-makers and stakeholders, the development of the Green City of Benguerir underscores the need for new strategies based on advanced spatial infrastructure and data-driven tools, such as UDT. These strategies are indispensable to align the city’s master plan with its sustainability objectives, address the disparities between newly developed urban areas and the existing old city model, and monitor urban progress in the face of evolving challenges, including, but not limited to urban planning, resource management (energy performance, water crisis in the region), and new mobility infrastructure solutions, all of which are integral to city green strategies.

To that end, this paper proposes a workflow for developing a UDT for the Green City, addressing the challenges encountered at each stage, particularly those related to data gaps. It explores multiple use cases in urban planning and highlights the broader applications of UDTs, such as in energy management.

The structure of the paper is as follows: Section 2 provides an overview of the background and related work on Urban Digital Twins and their applications. Section 3 outlines the methodology used to develop the UDT. Section 4 presents the findings and their implications, while Section 5 concludes the paper by summarising key insights and suggesting directions for future research.

2. Background and Related Work

2.1. Global City Trends

Global urbanisation presents a dual challenge: while cities generate 85% of the world's Gross Domestic Product (GDP) (<https://data.worldbank.org/indicator/NY.GDP.MKTP.CD?end=2023&start=1960&view=chart>) (accessed on 15 July 2025), they are responsible for 75% of global carbon emissions (https://capacity4dev.europa.eu/library/uneps-annual-report-2023-keeping-promise_en) (accessed on 15 July 2025). This tension is especially acute in the Global South, where urban populations have surged from 34% in 1960 to 57% in 2023 (<https://data.worldbank.org/indicator/SP.URB.TOTL.IN.ZS>) (accessed on 15 July 2025), often without commensurate infrastructure development. As a result, these cities face severe socioeconomic and environmental challenges. In response, many have aligned their urban policies with the Sustainable Development Goals (SDGs), particularly Goal 11, which focuses on making cities inclusive, safe, resilient, and sustainable [12].

While urban transitions are inherently spatio-temporal phenomena; cities must be understood as dynamic systems that evolve based on region-specific conditions and timelines [13,14]. Globally, policymakers have increasingly turned to ICT (Information and Communication Technology)-enabled solutions manifested in intelligent, digital, connected, or smart city initiatives to monitor and manage urban dynamics. Notable examples include Singapore's Smarter Future City initiative, launched in 2015 as part of the Intelligent Nation plan [15]; Japan's i-Japan Strategy, which emphasizes citizen-centric digital infrastructure [16]; and the European Commission's promotion of smart cities with a focus on energy efficiency and climate change mitigation [17].

These strategies converge on the principle that smart cities should efficiently serve the needs of their citizens. While current smart city systems rely heavily on ICT for real-time monitoring, many still lack robust mechanisms for representing the built environment in an interactive and holistic manner. UDTs offer a promising response to this gap. By creating a virtual replica of the city, UDTs enable bidirectional data interaction and near real-time feedback, supported by semantic 3D models that provide an optimal level of abstraction for spatial analysis [8]. However, a key limitation of current UDT implementations lies in their limited capacity for human interaction, which restricts their ability to fully capture and adapt to the behaviors and needs of urban residents.

2.2. UDTs in Underexplored Contexts: The Moroccan Imperative

In alignment with global trends, Morocco has launched several smart and sustainable urban development initiatives. Notable examples include the Benguerir Green City [18] and the Zenata Eco-City project [19]. However, it remains unclear whether UDTs have been implemented in these initiatives, or to what extent they are integrated into national or municipal urban planning frameworks.

To contextualise the adoption of UDTs in Morocco, existing studies offer insight into specific, often small-scale applications. For instance, UDTs have been used at the building level to monitor solar energy production systems [20], with similar approaches proposed for tracking rooftop solar panel performance in urban areas. Other research has explored the use of digital twins for monitoring road infrastructure [21] and for preserving cultural heritage sites [22]. Furthermore, this study [23] discusses the potential and challenges of developing a spatial data infrastructure to support data sharing and improve coordination among stakeholders.

These examples suggest that most UDT-related efforts in Morocco remain siloed or narrowly scoped, lacking integration into comprehensive urban-scale models that could support holistic planning development and still using traditional legal frameworks as fundamental sources for planning. A key barrier to advancing UDTs in this context is the

limited availability and accessibility of reliable and interoperable urban data, which is a common challenge in many developing countries.

In the case of Benguerir Green City, the University of Mohammed VI Polytechnic (UM6P) has played a central role in addressing urban challenges through research and innovation. However, ongoing studies indicate that a comprehensive UDT capable of holistically representing the city has not yet been developed, and that no robust, data-driven system currently exists to link physical city components to a dynamic, semantically rich urban model.

2.3. Technical Foundations of UDTs

A Digital Twin is a virtual replica of a physical environment, system, or process that mirrors its real-time state and performance through continuous data exchange [24–26]. When applied to urban environments, this concept evolves into an Urban Digital Twin or City Digital Twin (CDT), operating at various scales from individual buildings to entire cities [8]. UDTs provide dynamic, interactive representations of cities by integrating data from diverse urban sensors, thus transforming static 3D city models into live, real-time systems [1,27].

Developing a UDT should be a purpose-driven approach; it begins with identifying specific applications and outlining relevant use cases. Each application requires carefully defined spatial and non-spatial datasets to build a semantically rich city model that serves as the core of the UDT.

However, implementing such a workflow is non-trivial. It demands a robust software architecture and an efficient data pipeline that ensures seamless data collection, storage, streaming, and visualisation. Raw data must be cleaned, processed, and stored in standardised formats to enable interoperability and exchangeability, while also addressing issues of data security and privacy [7].

In this context, CityGML has emerged as the most widely adopted standard for representing 3D city models, while CityJSON offers a more web-friendly alternative with efficient geometry representation and better suitability for online interaction [28]. Enriching these models involves integrating multisource datasets at various architectural layers [29]. At the conceptual data model (CDM) level, several Application Domain Extensions (ADEs) support domain-specific data: the Noise ADE allows storage of noise data [30], while the Energy ADE focuses on energy-related attributes of buildings [31].

CityGML 3.0 introduces the Dynamizer module, enabling integration of real-time or time-series data from external sources such as spreadsheets or CSV files [32]. Alternative approaches include integrating IoT sensor data using the SensorThings API. For example, Ref. [33] proposed the CityThings concept, which binds 3D city features to sensor data using unique identifiers. Server-side data integration is also common, often achieved via RESTful APIs or MQTT protocols for IoT and spatial data streams [34]. These solutions support the development of geoportals and immersive platforms based on game engines [2,34].

Storing and managing the large volume of data associated with UDTs remains a central research challenge. The 3DCityDB project provides a robust data management framework for CityGML-based models [35], but its complex schema comprising over 66 tables can make data loading and querying cumbersome. In contrast CJBB a Python-based importer/exporter of CityJSONL files, offers a simplified alternative for CityJSON models, reducing schema complexity to just two tables and facilitating more efficient database operations [36].

Despite these advancements, UDT development faces several technical and non-technical challenges. Technically, ensuring interoperability across heterogeneous data

sources such as BIM, GIS, and IoT systems remains difficult [8]. Additional barriers include data quality, update frequency, and system scalability [7]. On the non-technical side, challenges include limited citizen engagement [37], organisational fragmentation, and a lack of cross-sectoral collaboration [38,39]. Privacy concerns and data confidentiality further complicate data sharing and system integration [40].

In summary, the UDT can be conceptualised as a virtual city actuator Figure 1) that transforms raw data into actionable insights, while incorporating feedback from stakeholders and citizens. These users not only consume data but also contribute to its acquisition and control, especially when empowered by AI-driven tools for proactive decision support and smart urban governance.

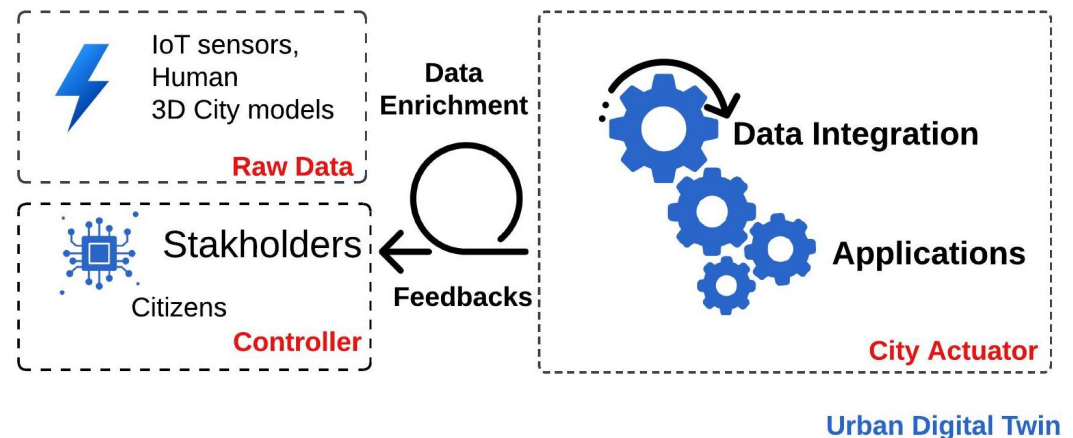


Figure 1. Urban digital Twin System Components.

3. Materials and Methods

3.1. Study Area and Data Sources

This study focuses on the city of Benguerir, Morocco, which was selected as a representative case for developing and testing an UDT framework in a data-scarce context. Benguerir is part of the Green City (GC) initiative, a large-scale urban development project that emphasizes sustainability, resilience, and innovation. Figure 2 illustrates the geographical extent of the study area.

To support the proposed methodology, multiple datasets were collected and integrated. These datasets are categorized according to their spatial representation (spatial vs. non-spatial) and their temporal characteristics (static, periodic, and dynamic). Each category serves a specific role within the UDT framework and was obtained from different sources and tools. In cases where real-world data were unavailable or incomplete, synthetic datasets were generated to demonstrate the functionality of the system and validate the methodological workflow. An overview of the datasets used in this study is provided in Table 1.

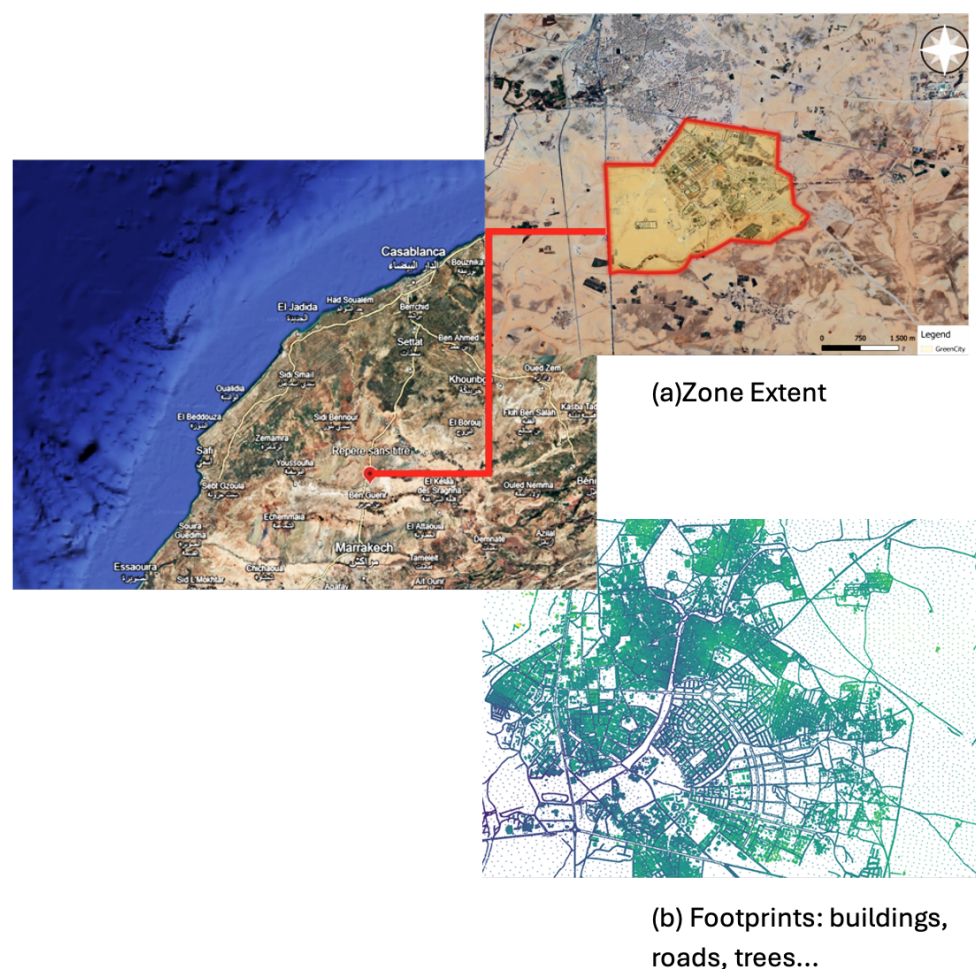
3.2. Methodological Framework

The overall methodology was designed based on prior research projects and an extensive review of related literature on UDTs, 3D city modeling, and urban analytics. The workflow is structured into two main components.

The first component focuses on the development of the UDT technical architecture and data pipeline. This process is divided into four sequential steps, detailed in Sections 3.3.1–3.3.4, covering data preparation, semantic modeling, database storage, and web-based visualization.

Table 1. Summary of datasets used in the study.

Dataset	Type	Description
Restitution and urban zoning plans	2D spatial data (static)	Derived from photogrammetric restitution and urban planning documents; used as base data for constructing the 3D city model and semantic enrichment.
Solar radiation data	Periodic raster data	Periodic solar radiation rasters generated using the Solar Radiation Analysis tool in ArcGIS Pro 3.5; used for rooftop solar potential estimation.
Sensor data	Dynamic data (REST API)	Synthetic real-time data streams generated using Node-RED to simulate IoT sensors (e.g., energy and environmental parameters).

**Figure 2.** Geographical extent of the Green City of Benguerir.

The second component applies the developed UDT framework to a set of urban planning and sustainability-oriented use cases. These include the analysis of building morphological indicators, building typology distribution, and construction age classification. In addition, the study explores energy-related applications such as rooftop solar potential estimation, building energy performance assessment, and shadow simulation. The complete methodological workflow is summarized in Figure 3.

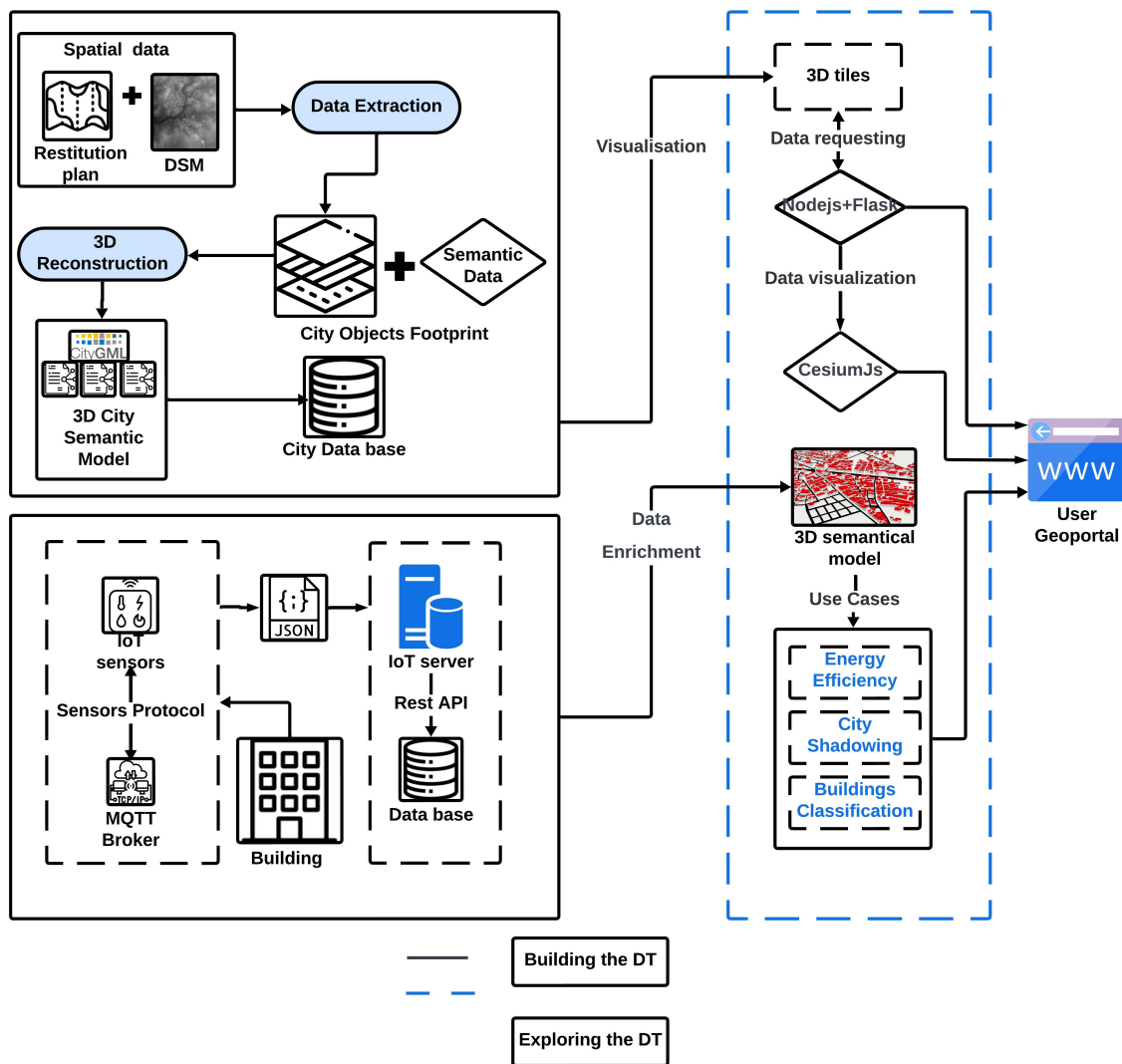


Figure 3. Overview of the methodological workflow for the Urban Digital Twin framework.

3.3. Use Case Design Framework

UDTs offer significant potential across multiple application domains. In this study, the primary focus is on urban planning-oriented use cases, while also illustrating complementary scenarios relevant to sustainability and decision support. The selection of use cases is guided by observed urban development challenges in Benguerir and constrained by the current availability of data, technical infrastructure, and institutional readiness.

To ensure practical relevance, the proposed use cases are designed to be technically feasible within data-scarce urban environments, such as those commonly found in Moroccan cities. Rather than aiming for fully operational simulations, the selected scenarios emphasize realistic, incremental implementations that can serve as proof-of-concept demonstrations. These use cases illustrate how UDTs can support planning analysis, regulatory assessment, and strategic urban interventions even under limited data conditions.

Table 2 summarizes the selected use cases, highlighting their objectives and the associated technical requirements and spatial operations. Detailed implementation workflows and system operations for these scenarios are presented in the following sections.

Table 2. Representative Urban Planning Use Cases Supported by the UDT Framework.

Use Case	Technical Requirements and Spatial Operations
Building Visibility and Shadow Analysis	• Semantic 3D city model generation with building geometry and height attributes • Sunlight and shadow simulation to assess visual and shading impacts • Integration of auxiliary datasets (e.g., property or economic data via APIs) • Web-based visualization dashboard with layered spatial information • Zoning compliance checks for height and floor-area-ratio (FAR) constraints
Optimization of Building Height Regulations	• Semantic extrusion of building footprints into 3D volumes • Computation of morphological indicators, including Height Homogeneity Index (HHI) and Area-to-Height Ratio (AHR) • Interactive spatial selection tools for zone-based analysis • 3D visualization with rule-based color coding (e.g., non-compliant buildings highlighted) • Automated reporting of regulatory compliance and scenario comparison (before/after simulations)
Urban Renewal and Redevelopment Strategy	• Semantic enrichment of buildings by function, age, and typology • Visual classification using color and shading schemes to represent use and construction period • Interactive 3D web map for filtering by attributes and temporal categories • User interface components (sliders, dropdowns) for exploratory analysis • Identification of underutilized or obsolete urban parcels to support redevelopment policies

3.3.1. Data Preparation

Modeling the urban ecosystem constitutes the foundational stage of the proposed UDT framework and begins with raw data acquisition, preparation, and cleaning. In the Benguerir case study, regulatory restrictions on drone operations prevented the use of UAV-based photogrammetry. Consequently, an alternative modeling strategy was adopted based on existing restitution plans provided by the Urban Agency of Benguerir. These restitution plans, delivered in DXF (Drawing Exchange Format), were originally produced through photogrammetric processes and exhibit an absolute horizontal accuracy of approximately 15 cm, consistent with the orthophoto from which they were derived. The dataset contains a multi-layer representation of urban features, including building footprints, road networks, green spaces, water bodies, and other thematic layers (see Table 3). To transform the raw CAD data into interoperable geospatial formats, a sequence of extraction, cleaning, completion, and conversion operations was applied. The processed datasets were converted into GeoPackage format and further refined to comply with Open Geospatial Consortium (OGC) Simple Feature standards (<https://docs.ogc.org/as/17-087r13/17-087r13.html> (accessed on 15 July 2025)). Building footprints were first converted into polygonal geometries and subsequently enriched with elevation values derived from raster-based height data. Due to fragmented or disconnected boundaries present in the source DXF files, a geometry correction workflow was implemented consisting of (i) boundary extraction as line geometries, (ii) line merging and topology correction, and (iii) polygon reconstruction from the

merged boundaries. This process proved effective in restoring valid polygonal geometries suitable for three-dimensional reconstruction. All geometries were then normalized to ensure compliance with ISO 19107 [41] geometry primitives. In particular, polygon and multipolygon features were unified into MultiSurface and MultiSolid representations to prevent inconsistencies during 3D modeling and database ingestion. For elevation modeling, a Digital Surface Model (DSM) was generated from raster data. To enhance elevation continuity, additional distributed Z-points were generated within feature geometries such as rooftops, roads, and walls. These points were converted into LAS point cloud format, enabling the generation of a higher-resolution DSM. Given the predominantly flat topography of Benguerir, this approach provided sufficient vertical accuracy while maintaining computational efficiency. Elevation values were originally referenced to the Moroccan National Geodetic Leveling Network (NGM). Due to the absence of common control points between the NGM and WGS84 reference systems, an absolute vertical transformation was not feasible. Consequently, normalized elevation values were used for feature extrusion. Z-values were extracted by intersecting building footprints with the normalized DSM, and mean elevation values were assigned to each building footprint. Three-dimensional building geometries were generated using a Boundary Representation (BRep) approach by extruding building footprints and constructing wall and roof surfaces to form LoD2 representations where possible. The resulting geometries were structured as polygon Z (footprints) and multipolygon Z (walls and roofs). The complete 3D reconstruction workflow was implemented using FME Workbench, ensuring compliance with the CityGML v2.0 standard. Subsequently, the models were converted to CityGML v3.0 using the citygml2-to-citygml3 conversion tool (<https://github.com/tum-gis/citygml2-to-citygml3> (accessed on 15 July 2025)). Other urban features were modeled following similar principles but with simplified representations dictated by data limitations. Transportation infrastructure was generated at LoD1 as continuous surface geometries, as the raw road network data consisted of undifferentiated polylines. Vegetation and city furniture were represented using point-based geometries. In the case of trees, the available dataset included only X, Y, and Z coordinates, without attributes such as height or trunk diameter. To address this limitation, 3D symbology instances were employed using glTF models, randomly scaled within the database. This approach enabled visually realistic rendering when deployed as 3D Tiles while maintaining low data complexity.

Table 3. Geometry Issues and Final Geometry for Filtered Layers.

Filtered Layers	Geometry Type	Data Limitation/ Geometry Issues	Final Geometry
Building Footprints	Polylines	Unjoined line segments	Z Polygon
Green Spaces	Polylines	Self-intersecting geometries	Z Polygon
Water Bodies	Polygons	Low vertex count	Z Polygon
Roads	Polylines	No segmentation; single-line representation	LoD1 Surface
Trees	Points	No Z attribute	Z Points (glTF instances)
City Furniture	Points	Incomplete data model	TIN or glTF Instances
Relief	Raster/Contours	Irregular spacing	TIN/DEM

3.3.2. Data Integration

To support diverse urban analysis and planning applications, the UDT requires an enriched semantic 3D city model. This phase focuses on integrating additional semantic attributes derived from static, periodic, and real-time data sources. As discussed previously (see Section 2.3), data integration can be implemented at three complementary levels:

directly within the City Model Database (CMD), at the underlying database layer, or at the server level hosting the geoportal and visualization services.

- **Static data:** This category includes time-invariant attributes such as construction year, number of building floors, volume, surface area, zoning designation, and address. These attributes were integrated directly into the building objects as semantic properties. Authoritative sources include urban development plans, cadastral datasets, and official urban planning regulations. The number of building floors was derived using a deterministic, rule-based calculation rather than statistical inference. The computation relies on building elevation values extracted from the Digital Surface Model (DSM), combined with zoning-specific planning constraints defined in official regulatory documents. These constraints include maximum allowable building height, foundation height, minimum and maximum floor heights, and maximum permitted number of floors per zoning class. The complete sequence of algorithmic steps used for floor estimation is detailed in Algorithm 1. In addition, several geometric attributes such as footprint area, perimeter, and enclosed volume were computed directly using standard spatial geometry operations.

Algorithm 1 Rule-Based Estimation of Building Floor Count

Require: B : set of building features with height attributes

Require: Z : set of zoning areas

Require: R : set of planning rules indexed by zoning class

Ensure: B' : buildings enriched with a rule-based number of floors

```

1: for each building  $b \in B$  do
2:   Identify zoning area  $z \in Z$  such that  $b \subset z$ 
3:   Retrieve applicable planning rules  $r \in R$  for zoning class  $z$ 
4:   if height attribute of  $b$  is missing or invalid then
5:     Assign null to the number of floors of  $b$ 
6:     continue
7:   end if
8:   Compute effective building height  $h_{\text{eff}}$  by adjusting total height according to foundation constraints in  $r$ 
9:   Select representative floor height  $h_{\text{floor}}$  within the allowable range defined by  $r$ 
10:  Estimate preliminary floor count  $n = \left\lfloor \frac{h_{\text{eff}}}{h_{\text{floor}}} \right\rfloor$ 
11:  if  $n > r.\text{max\_floors}$  then
12:     $n \leftarrow r.\text{max\_floors}$ 
13:  end if
14:  Assign  $n$  as the number of floors of  $b$ 
15:  Flag the attribute as rule-based
16: end for

```

- **Periodic data:** Solar radiation data were integrated as an example of periodically varying environmental information. Using the Solar Radiation Analysis tool in ArcGIS Pro (<https://pro.arcgis.com/en/pro-app/latest/tool-reference/spatial-analyst/raster-solar-radiation.htm> (accessed on 15 July 2025)), two raster outputs were generated:
 - Monthly direct irradiation, representing direct solar energy received (Wh/m^2);
 - Monthly diffuse irradiation, representing scattered solar energy (Wh/m^2).

Additionally a raster layer representing the duration of direct sunlight exposure (in hours) was produced. These raster datasets were spatially intersected with building footprints to assign solar exposure metrics to individual buildings within the city model.

- Real-time data: To simulate dynamic sensing capabilities in the absence of a deployed IoT infrastructure, synthetic real-time sensor streams were generated using Node-RED. These data streams were managed through the SensorThingsAPI, implemented using the Fraunhofer Open SensorThings (FROST) server (<https://github.com/FraunhoferIOSB/FROST-Server>) (accessed on 15 July 2025). This linkage is established through the properties attribute in the Sensors and Things tables. In these tables, the CityGML_ID serves as a unique identifier for the buildings. This identifier is added as a property of both the Thing and the Sensors, as illustrated in the JSON snippet below Listing 1.

Listing 1. Example of Sensor Properties.

```
{
  "citythings": {
    "image": "building5951_image.png",
    "gml_id": "building5951",
    "sensors": {
      "Energy_counter": {
        "location": "Auditorium",
        "sensor_id": "sensor18880",
        "sensor_image": "sensor124_image.png"
      },
      "sensor_Temperature": {
        "location": "Center of the rooftop Auditorium",
        "sensor_id": "sensor18881",
        "sensor_image": "sensor123_image.png"
      }
    },
    "location": "Auditorium Amphi"
  }
}
```

After associating the corresponding gml_id of the buildings with the respective sensor properties, we attempt to fetch the observations linked to the buildings using JavaScript code implemented on the server side of our application.

The primary simulated sensor stream represents:

- Building-level energy consumption, with daily and monthly values generated per building.
- Sensor data were configured in Node-RED, stored in a dedicated sensor database, and dynamically linked to the 3D city model through the CityThing framework. This integration enables bidirectional interaction between semantic city objects and time-dependent sensor observations. The overall architecture of the sensor integration workflow is illustrated in the UML diagram shown in Figure 4.

3.3.3. Data Storage

Following the modeling phase, an efficient data storage strategy is required to preserve, manage, and provide scalable access to heterogeneous 3D urban datasets within the UDT system. Given the volume, semantic richness, and dynamic nature of the integrated data sources, ensuring interoperability and performance during storage and streaming constitutes a key technical challenge. Addressing this challenge necessitates a storage approach aligned with both semantic modeling standards and web-based visualization requirements.

In this study, two complementary storage strategies were investigated: (i) a file-based approach using the CesiumIon API, and (ii) a database-driven approach using 3DCityDB.

The file-based strategy relies on the CesiumIon cloud service, which supports uploading, hosting, and streaming 3D Tiles directly to web clients. While this solution offers ease of deployment and high rendering performance, it remains inherently static and is therefore unsuitable for scenarios involving frequent data updates, dynamic attributes, or tight integration with analytical workflows.

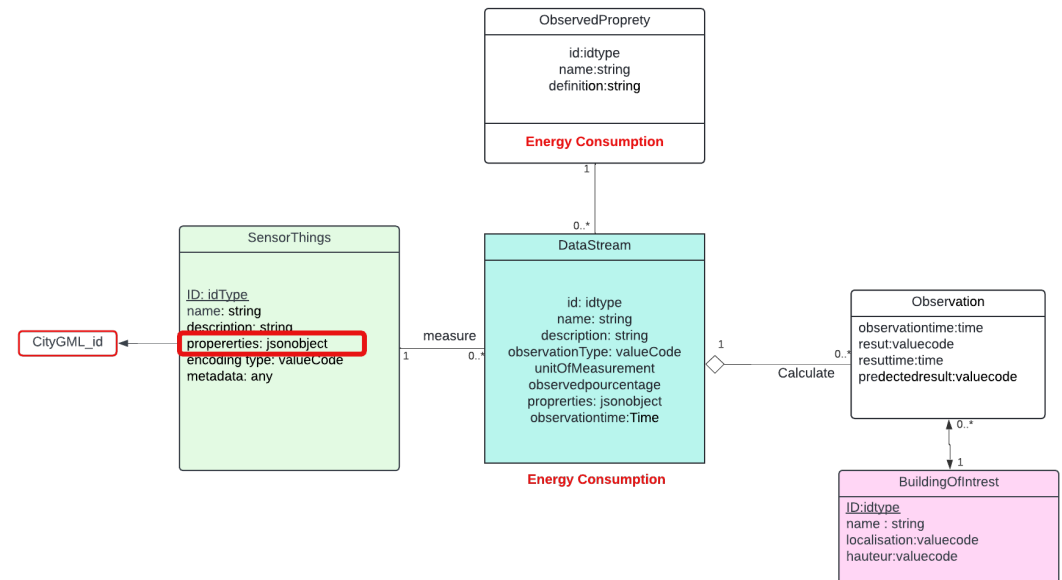


Figure 4. UML diagram illustrating the integration of sensor data with the Urban Digital Twin.

To overcome these limitations, a database-driven approach was adopted as the primary storage solution. The 3DCityDB platform was selected due to its native support for the CityGML data model and its ability to preserve semantic richness through a normalized relational schema. Built on PostgreSQL with the PostGIS extension, 3DCityDB distributes city objects across more than 60 interrelated tables, enabling detailed thematic classification and spatial querying.

All datasets were validated and imported using the 3DCityDB Importer/Exporter tool, ensuring compliance with CityGML version 2.0.0. Despite its strong semantic consistency, the complexity of the schema introduced significant performance bottlenecks, particularly during querying and rendering via the 3DCityDB Web Feature Service (WFS) and the 3DCityDB Web Map Client (<https://github.com/3dcitydb/3dcitydb-web-map> (accessed on 15 July 2025)) (Figure 5). These limitations motivated the adoption of an alternative strategy optimized for web-based visualization.

To improve visualization performance and support scalable streaming, an alternative pipeline was implemented to generate and serve 3D Tiles directly from a database structure designed for tiling workflows. The 3D Tiles generation process follows the methodology proposed by [42] and builds upon the open-source implementation available on GitHub (<https://github.com/yangzoe/3dtiles> (accessed on 15 July 2025)) (Figure 6).

This pipeline comprises four main stages:

1. Geometric computation and data extraction: Object geometries are extracted from the database, triangulated, and enriched with auxiliary geometric attributes such as surface normals and bounding volumes.
2. Spatial clustering: City objects are grouped into spatial clusters to generate multiscale tilesets supporting multiple Levels of Detail (LODs), thereby improving performance across different zoom levels.
3. Tileset hierarchy construction: A hierarchical spatial index is constructed to organize tiles from coarse to fine granularity, enabling efficient traversal and selective loading.

4. Web-based streaming: The resulting tileset is served through a server–client architecture compatible with CesiumJS.

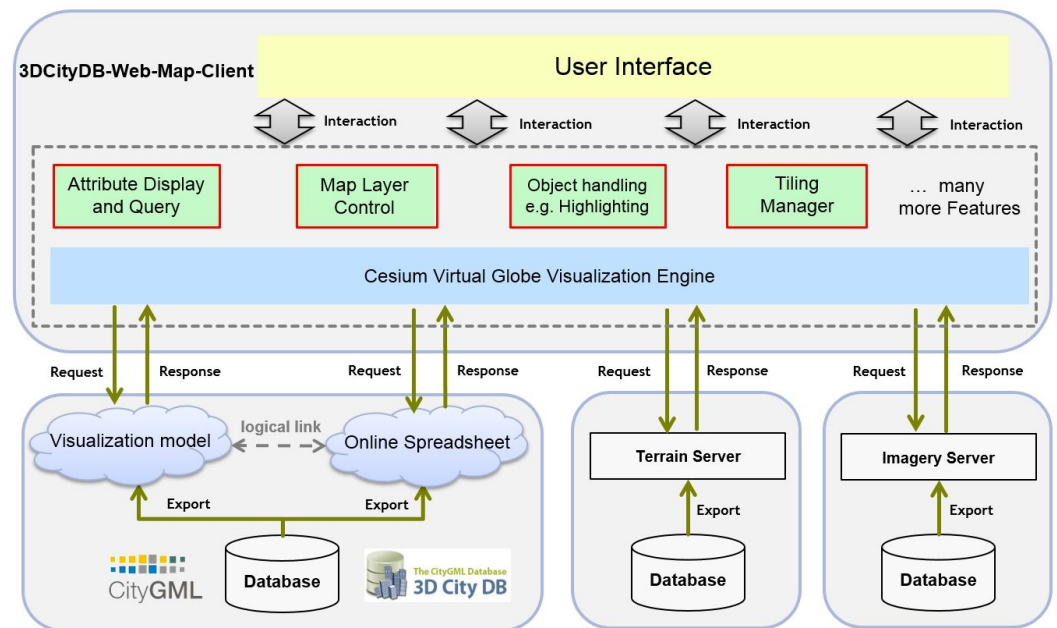


Figure 5. 3DCityDB Web Map Client architecture.

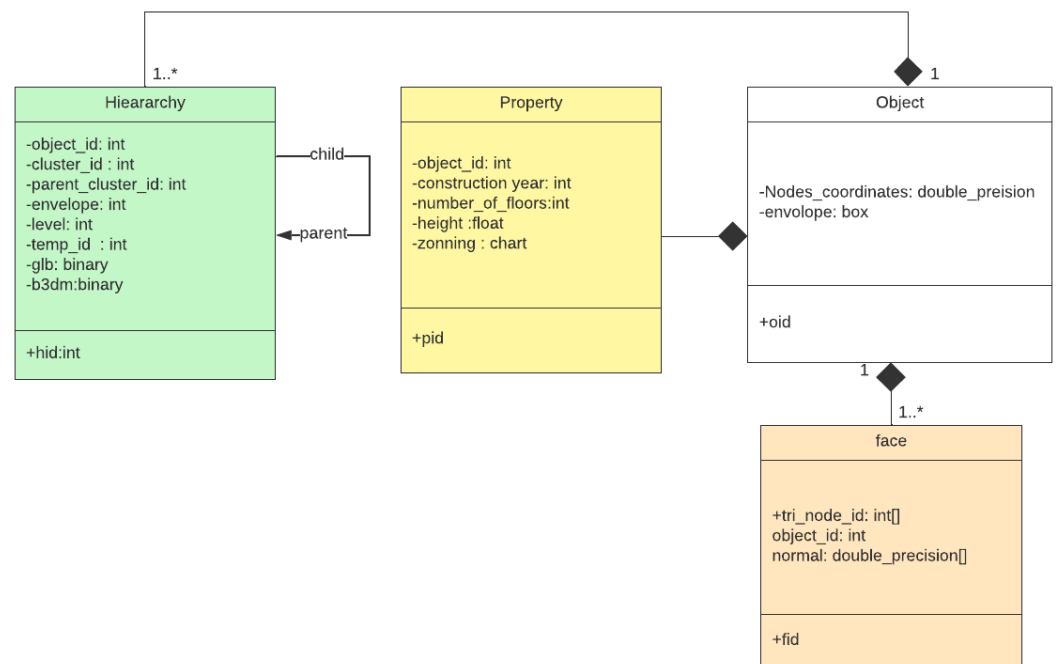


Figure 6. UML diagram of the database-driven 3D Tiles generation workflow.

The adopted storage model preserves explicit object-level geometry together with triangulated topology, following an object-oriented abstraction. Each physical object is represented by its body topology, while vertex coordinates are stored as nodes. For triangulated objects, faces reference node indices directly. This design avoids repeated reconstruction of geometry from normalized tables and significantly reduces traversal overhead during tile extraction. For spatial indexing, GiST-based R-tree structures in PostGIS were initially considered. While effective for spatial querying, GiST indexes implicitly manage node partitioning and do not provide explicit control over cluster size or hierarchy depth. More-

over, R-tree indexing does not directly align with the hierarchical structure required by the 3D Tiles specification, where tile granularity must be explicitly controlled as depicted in Listing 2.

Listing 2. Example of Input JSON for the Web Data Serving Component.

```
{
  "Ben": {
    "description": "This is a dataset of ALamal neighborhood
buildings",
    "lod": "lod1",
    "lod_description": "lod1",
    "mode": 1,
    "mode_description": {
      "1": "server and creator",
      "0": "server"
    },
    "cluster_number": [
      1,
      1
    ],
    "triangulation_flag": "tessellation",
    "index_flag": 0,
    "index_flag_description": {
      "0": "non-indexed",
      "1": "indexed"
    },
    "b3dm_flag": 1,
    "glb_flag": 1,
    "b3dm/glb_flag_description": {
      "1": "composed",
      "-1": "not composed"
    },
    "property": [
      "height",
      "construction_year",
      "building_type"
    ],
    "filter": "",
    "custom_parameter": ""
  }
}
```

To address these constraints, clustering-based partitioning strategies were explored. The PostGIS ST_ClusterKMeans function was selected over density-based methods (e.g., DBSCAN) because it guarantees complete object assignment without labeling outliers. However, standard k-means clustering does not enforce minimum or maximum cluster sizes, resulting in heterogeneous tile payloads. To mitigate this limitation, a hierarchical k-means clustering strategy was implemented. In this approach, city objects are first partitioned into coarse clusters at the top level, followed by recursive subdivision into finer clusters. Lower-level clusters are constrained to remain within the spatial extent of their parent clusters, ensuring spatial coherence across hierarchy levels. This top-down clustering directly defines the tileset hierarchy required by the 3D Tiles specification. Finally, the resulting cluster hierarchy is mapped to the tileset JSON structure, where each node corresponds to a spatial cluster and its associated geometric payload. This organization

enables progressive loading, reduces client-side parsing overhead, and improves rendering stability by maintaining balanced tile sizes and strong spatial locality. Concerning the web data serving component, a connection was established between the database and the application server using Flask on the backend to fetch tilesets based on the client's routing path. The main application function accepts an input JSON file, depicted as follows; the cited parameters are described in Appendix A, which also includes an example of the code used to fetch the created tiles.

3.3.4. 3D Visualisation

To ensure broad accessibility of the UDT for both stakeholders and citizens, a web-based application was developed to support interactive 3D visualisation, environmental monitoring, and urban data analysis. The system follows a modular, service-oriented architecture. The back-end combines Node.js and Flask services, while the front-end is implemented using React and CesiumJS to deliver an interactive and immersive 3D user interface.

Rendering and streaming large-scale 3D geospatial data in web environments present significant challenges, primarily due to data volume, complex spatial formats, and limited client-side computational resources. To address these challenges, an optimized visualization pipeline was implemented to enable scalable content delivery, efficient rendering, and responsive user interaction.

The adopted pipeline integrates two complementary visualization strategies, as illustrated in the sequence diagram shown in Figure 7:

- **Cesium Ion API:** This cloud-based approach is used to stream preprocessed 3D Tiles, enabling rapid visualization of static 3D city models. It provides a low-configuration solution suitable for demonstration purposes and scenarios where frequent data updates are not required.
- **Database-driven 3D Tiles streaming:** In this approach, 3D Tiles are generated from a custom PostgreSQL/PostGIS database and stored in the batched 3D model (B3DM) format. Tiles are served dynamically through a custom API in response to client requests, allowing integration of updated semantic attributes and supporting near real-time rendering. The client-side visualization is handled by CesiumJS, which enables progressive loading and level-of-detail management.

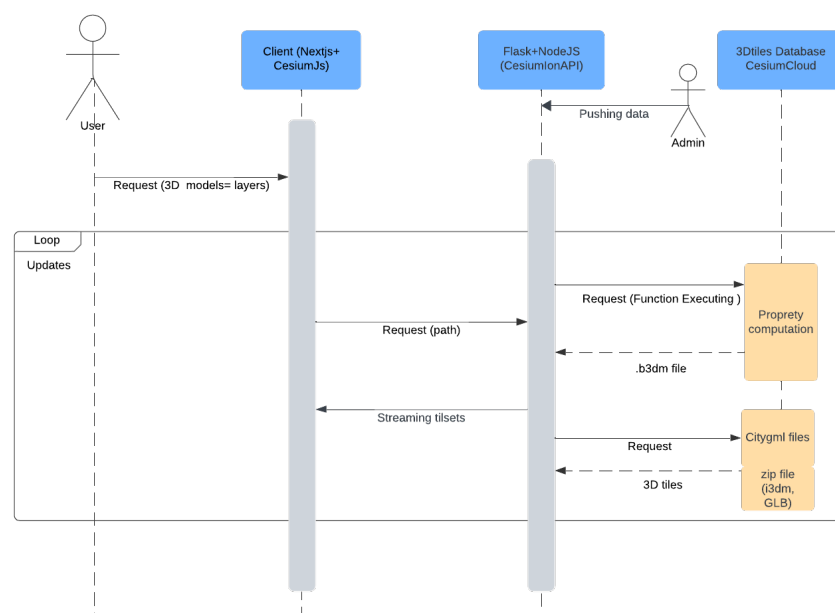


Figure 7. Sequence diagram illustrating the database-driven 3D visualization workflow.

This dual strategy allows the system to balance deployment simplicity with flexibility and performance. While Cesium Ion supports rapid visualization of static datasets, the database-driven approach provides greater control over data updates, querying, and tile organization, making it more suitable for Urban Digital Twin applications requiring frequent data interaction.

3.4. UDT Applications

This section demonstrates the practical implementation of the proposed UDT framework through a set of representative use cases developed for the Benguerir study area. The objective is to illustrate how the integrated 3D city model, semantic enrichment, and visualization components support urban analysis and decision-making under data-constrained conditions.

The implemented use cases are grouped into two primary domains: urban planning and energy management. Each use case leverages different combinations of static, periodic, and dynamic data, highlighting the flexibility of the proposed framework. Table 4 summarizes the implemented use cases, the input datasets employed, and the corresponding analytical outputs.

Table 4. Overview of UDT use cases, input data, and outputs.

Use Case	Input Data	Output
Building Classification and Morphology Indicators	3D building geometries, cadastral data, semantic building attributes (e.g., construction year, height, typology)	Building classification maps and quantitative morphology indicators (e.g., height homogeneity, area-to-height ratio)
Building Energy Efficiency	3D building geometries, simulated sensor data, monthly solar radiation metrics	Energy performance indicators and interactive visual analytics dashboards

3.4.1. Building Classification and Morphological Indicators

This use case supports urban planning analysis by classifying buildings according to enriched semantic attributes, including construction year, building height, and functional typology. These attributes enable zoning-related assessments and facilitate spatial decision-making processes at multiple urban scales. In addition, a set of morphological indicators is computed to quantitatively characterize the structural and vertical organization of the built environment within a predefined spatial unit (e.g., city block, district, or user-defined analysis zone).

Let n denote the total number of buildings within the selected analysis area.

Morphological Indicators

Two complementary indicators are employed:

- **Height Homogeneity Factor (HHF):** This indicator quantifies the degree of uniformity in building heights within the analysis area. It is expressed as the coefficient of variation in building heights. Lower values indicate a more homogeneous urban fabric, while higher values reflect greater vertical heterogeneity, which may influence visual coherence and skyline continuity.
- **Area-to-Height Ratio (A/H):** This indicator characterizes the vertical compactness of buildings by relating footprint area to building height. It provides insights into urban density patterns and potential impacts on environmental factors such as sunlight penetration and airflow.

Mathematical Formulation

Let H_i denote the height of building i and A_i its footprint area.

Height Homogeneity Factor:

$$\bar{H} = \frac{1}{n} \sum_{i=1}^n H_i \quad (1)$$

$$\sigma_H = \sqrt{\frac{1}{n} \sum_{i=1}^n (H_i - \bar{H})^2} \quad (2)$$

$$CV_H = \frac{\sigma_H}{\bar{H}} \quad (3)$$

where CV_H represents the Height Homogeneity Factor.

Area-to-Height Ratio Indicator:

$$\overline{A/H} = \frac{1}{n} \sum_{i=1}^n \frac{A_i}{H_i} \quad (4)$$

$$\sigma_{A/H} = \sqrt{\frac{1}{n} \sum_{i=1}^n \left(\frac{A_i}{H_i} - \overline{A/H} \right)^2} \quad (5)$$

$$CV_{A/H} = \frac{\sigma_{A/H}}{\overline{A/H}} \quad (6)$$

where $CV_{A/H}$ expresses the variability in the area-to-height ratio within the selected spatial unit.

Interpretation and Visualization

The computed indicators are reported as numerical metrics and visualized using Gaussian distribution curves to illustrate the dispersion and frequency of building heights and area-to-height ratios within the selected analysis zones. These outputs provide urban planners with quantitative measures to assess spatial coherence, vertical regularity, and potential impacts on urban visibility, solar access, and microclimatic conditions at neighborhood and district scales.

3.4.2. Building Monthly Solar Potential and Energy Consumption

This use case focuses on supporting urban energy management strategies by analyzing rooftop solar potential and comparing it with building energy consumption. Monthly solar irradiation estimates are used as a proxy for potential renewable energy production, while simulated energy consumption values represent operational demand at the building level.

The application integrates semantic 3D building models, simulated sensor data, and auxiliary datasets such as solar radiation rasters. Each sensor is linked to its corresponding building through a unique CityGML_id and is integrated into a SensorThings API-compliant data structure, ensuring standardized data exchange and interoperability.

On the server side, monthly solar radiation data is processed to estimate potential energy production for each building. These estimates are compared against observed or simulated energy consumption values to assess relative building energy performance. In addition, shadow analysis is integrated using CesiumJS, enabling interactive visualization of solar exposure. Users can dynamically adjust simulation parameters such as date, time, and playback speed to observe temporal variations in shadow patterns and their influence on building surfaces.

4. Results

This section presents the key outcomes of our UDT implementation for Benguerir, demonstrating both the technical feasibility and practical applications of the developed framework. Our results are structured across three principal components: (1) the generation and validation of Semantic 3D City Models (S3DCMs) incorporating multiple thematic urban classes; (2) the implementation of a functional geoportal enabling dynamic visualization and analysis; and (3) specific urban planning applications that leverage the UDT's capabilities for morphological analysis and energy performance assessment. Through quantitative metrics and qualitative visualizations, we illustrate how the integration of semantic 3D modeling with real-time data streams creates an operational decision-support tool that addresses Benguerir's unique urban challenges while providing a replicable model for similar contexts in Morocco.

4.1. 3D City Semantic Models

The accurate generation of 3D city models is critical for ensuring their usability in downstream applications. Errors or inaccuracies in the modeling process can significantly limit model effectiveness. Therefore, we placed strong emphasis on validating both geometric accuracy and semantic consistency of our 3DCSMs.

Relying on numbers during the initial validation stage, only approximately 35% of the building geometries were compliant with ISO19107 standards, primarily due to geometric and topological inconsistencies in the raw input data. Through an extensive geometry correction and topology repair process focusing on boundary reconstruction, polygon validation, and geometry normalization, the compliance rate was improved to approximately 87% for building objects only after verification at the CityGML importer/exporter tool.

In this study, we generated detailed 3D city models across multiple thematic classes to create a comprehensive virtual representation of the city (see Figure 8). Beyond visual representation, our models support analysis of spatial interactions and thematic relationships.

We adopted the CityGML standard for data storage and management, which provides a high accurate data structure and geometric-semantic coherence and widely accepted format for built-environment modeling. The validation process using the 3D City Database (3DCityDB) Importer/Exporter tool identified a small number of non-valid geometries, primarily related to minor topological inconsistencies, which were subsequently corrected to ensure model integrity.

Integration of the 3DCSM into the geoportal is accomplished via a RESTful API interfacing with the Cesium Ion service, enabling real-time, on-demand data fetching and rendering.

4.2. Application in Urban Planning

We conducted urban morphology analyses to characterize building stock based on key parameters including building height, year of construction, and typology (Figure 9). These classifications provide stakeholders with a comprehensive overview of the current urban fabric.

Furthermore, we calculated morphological indicators such as the Height Homogeneity Indicator (HHI) and Area-Height Ratio (AHR). These indicators support various analyses including solar exposure assessment, visibility studies, and obstruction evaluations, providing valuable insights for urban planners and designers.

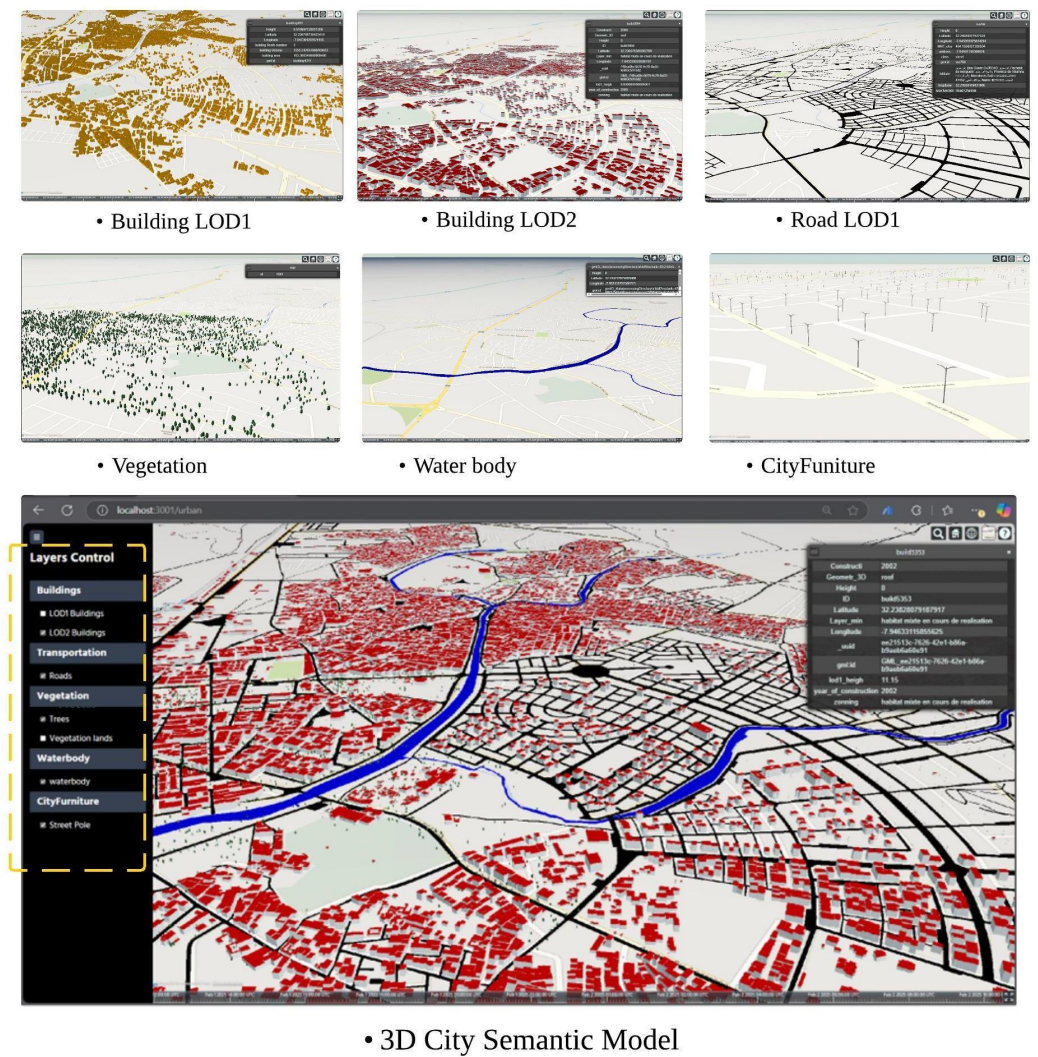


Figure 8. Semantic 3D City Model showing multiple feature classes within the dashed yellow line.

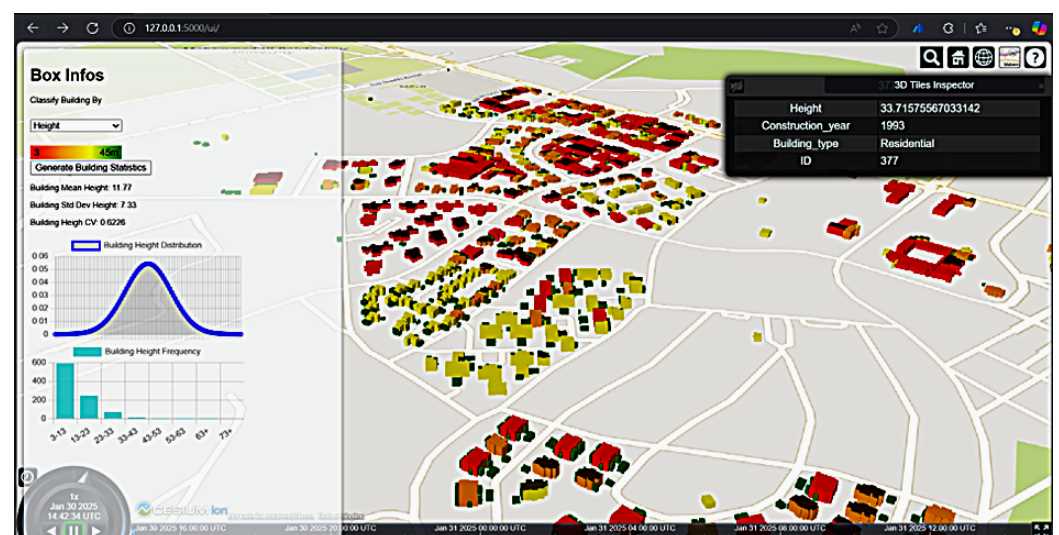


Figure 9. Building classification and morphological indicators.

Figure 9 illustrates the spatial classification of buildings by height in a selected area of Benguerir, visualized using the UDT application. The buildings are color-coded according to height ranges, revealing diverse vertical morphology across the urban fabric. Taller buildings (shown in red) are concentrated primarily in the central district, likely indicating

commercial or institutional zones, while shorter structures (in green and yellow) dominate surrounding residential areas.

The statistical analysis displayed in the left panel shows a mean building height of approximately 11.77 m (standard deviation = 7.33), suggesting moderate variation in vertical development. The height coefficient of variation ($CV = 62.26\%$) confirms heterogeneous building heights, further emphasized by the normal distribution curve of height values.

The Building Height Frequency chart reveals that the most common height class (6–12 m) reflects a predominantly low-rise urban typology, consistent with the residential nature of most buildings in the area. However, outlier buildings exceeding 30 m introduce local visual inconsistencies and potential shading impacts, particularly in denser areas.

These insights are critical for urban zoning analysis and skyline regulation. The observed spatial height variations can inform policies regarding view corridor protection, solar access, or vertical densification strategies. Moreover, integrating 3D classified buildings with their metadata (e.g., construction year, type) provides a dynamic basis for simulating redevelopment scenarios or assessing future vertical expansion impacts.

Our energy analysis evaluates both solar irradiation potential on building surfaces and corresponding monthly/daily energy consumption patterns (Figure 10). By comparing potential solar energy production with actual consumption, stakeholders can identify opportunities to optimize energy use and promote renewable solar adoption. Additionally, we developed a real-time shadow simulation tool to evaluate shading impacts on the urban environment, (the sub-mentioned utilities are depicted in Figure 10).



Figure 10. Energy Analysis: Shadow simulation, monthly solar potential estimation, and daily/-monthly building energy consumption.

4.3. The Geoportal

The geoportal serves as the primary user interface of the UDT, providing agnostic and intuitive access to urban data. It acts as the essential link between backend technical architecture and end-user experience, with key design principles including reusability, ease of maintenance, and transferability.

Our geoportal architecture follows a modular approach, separating the system into two backend frameworks and a frontend application. The frontend is decomposed into distinct components (viewer, header, toolbox, footer, charts) to ensure maintainability and facilitate future development. During development, it demonstrated strong performance and efficient rendering of complex 3D content while hosted locally.

The geoportal comprises several key services accessible via three main routes:

1. Urban Data Model: 3D representation integrating all major urban components as static data enriched with semantic attributes.
2. Urban Analysis: Facilitates building classification by semantic attributes and urban morphology indicator calculations.
3. Energy Analysis: Provides solar irradiation estimates on building roofs, analyzes monthly energy consumption, and compares potential solar production to usage ratios.

5. Discussion

This study led to the development of a UDT framework to analyze the urban fabric of Benguerir City. Starting with the creation of a 3DCSM, the modeling process was based on raw data from the 2D restitution plan produced through photogrammetry, successfully generating 3D models for multiple thematic classes at LoD1. Additionally, due to the simple building structures, we attempted to generate LoD2 buildings by extracting roofprints and footprints separately, then linking them according to the hierarchical structure defined by the standards. The generated models were successfully validated and conformed to CityGML v2.0; however, we encountered several challenges in geometry validation, as described previously in Section 3.3.1. We adopted the CityGML standard due to its structured data model, semantic-geometry coherence, standardized documentation, and active user community. Despite these advantages, CityGML has several limitations, including frequent geometry validation errors, large data sizes, limited software support for data storage, and a lack of efficient data viewers. To address this, CityGML is used as the authoritative semantic representation and database schema, while CityJSON and 3D Tiles are derived products optimized for querying and visualization. To illustrate this, we provide a comparative overview in Tables 5 and 6 according to multiple criteria.

Table 5. Comparison between CityGML, CityJSON.

Aspect	CityGML (XML)	CityJSON
Semantic richness	Very high	High (same conceptual model)
File size	Large	Compact
Web parsing performance	Slow	Fast
Direct querying capability	Poor	Moderate
Visualization quality	Limited	Good
Role in the framework	Authoritative semantic model	Web-friendly encoding

Table 6. Comparison of CityGML and CityJSON file sizes with compression factors.

Dataset	CityGML Size (MB)	CityJSON Size (MB)	Coeffeciant (CityGML/ CityJSON)
LOD1 Building	79.5	14.8	5.37
LOD2 Building	107	30.1	3.56
Roads	6.0	2.81	2.14

For example, 3DCityDB does not yet support CityGML v3.0.0, and even for older versions, it suffers from low-performance rendering when coupled with visualization tools. Similarly, existing CityGML viewers like FZK Viewer can only handle limited data

sizes, while SimStadt's new version lacks advanced data exploration tools. Furthermore, the Dynamizer module introduced in CityGML v3.0.0 was expected to improve multi-source data integration, but we faced significant visualization and validation issues. As an alternative, CityJSON provides a more streamlined pipeline for data validation, storage, and visualization, addressing many of CityGML's challenges.

To assess the suitability of different data representations for web-based Urban Digital Twin applications, an exploratory evaluation was conducted focusing on parsing, visualization, and querying of building features. The same buildings dataset and level of detail were used across all formats, and tests were performed under identical hardware (MSI laptop with 16 RAM and 16 RAM GPUs, RTX3050) and browser conditions in a single-user environment. The evaluation compared CityGML (XML), CityJSON, and CityGML-derived 3D Tiles, with performance measured in terms of initial load time, parsing behavior, response time, and interaction stability. The results are intended to provide implementation-level insights rather than exhaustive benchmarking in the Table 7.

Table 7. Comparison of Parsing and Visualization Approaches for Building Features.

Format	Encoding/Access Method	Parsing Strategy	Initial Load Time	Rendering Stability
CityGML	XML (WFS/file-based)	Server-side XML parsing	>600 s (timeout)	Unstable
CityJSON	JSON (file-based)	Client-side JSON parsing	~12–18 s	Stable
CityGML → 3D Tiles	Batched binary tiles (b3dm)	Preprocessed tile streaming	~5–8 s	Highly stable

The second part of our study focused on the UDT software architecture, particularly on data integration, storage, and visualization. While 3DCityDB was primarily used for storing 3D city models, rendering urban data models dynamically required an optimized database design. In our case, the use of a 3D Tiles database effectively supported the requirements of urban analysis. Integrating the SensorThings API with 3D city models on both the database and server sides proved to be a practical approach, aligning well with our use case (see Table 8). Initially, we stored the data using the CityGML standard within 3DCityDB, which relies on a relational database system (PostgreSQL with the PostGIS extension). 3DCityDB employs a detailed schema, distributing data across 55 tables according to object classes. For instance, Level of Detail 1 (LoD1) buildings are distributed across four key tables: cityobject, building, geometry, and cityobjectgenericattribute. Although 3DCityDB provides strong geospatial consistency through PostGIS's 3D geodata management capabilities, its complex schema makes it less suited for handling big data or dynamic changes. Alternative solutions, such as CJDB and Measur3D with Cerbere, offer simplified schemas better suited for scalability. CJDB, for example, significantly reduces the complexity of the 3DCityDB schema, facilitating easier export of CityJSON data into two tables. However, this simplicity comes at the cost of data consistency and limits the use of PostGIS's 3D functions. In contrast, Measur3D employs a document-oriented database (MongoDB) for improved support of big data, while the middleware Cerbere ensures consistency by validating transactions against the CityGML schema. We rely on previous studies [36,43] and our use case study to validate the comparison between these database systems, as presented in the following table:

While 3DCityDB imposes a storage requirement of 66 tables, consuming 23 MB, CJDB and Measur3D offer a more lightweight approach. For instance, the schema for Measur3D only requires 12 KB to create the three initial empty collections, making it a more efficient

solution for large-scale or dynamic urban data management. For user accessibility, our geoportal offers enhanced flexibility, with CesiumJS enabling robust 3D visualization capabilities. However, certain limitations persist, such as the absence of advanced spatial tools (e.g., area selection and measurement). These limitations can be mitigated by integrating CesiumJS with complementary libraries like OpenLayers or Mapbox to extend its spatial functionality. Overall, the software architecture has proven efficient for our use case, demonstrating strong performance in data integration, visualization, and analysis. Beyond its immediate applications, UDTs exhibit significant potential for analyzing urban planning and infrastructure management.

Table 8. Comparison of databases for 3D geospatial data handling, Ratings: * moderate, ** better, *** best [43].

Database Schemas	3DCityDB	CJDB	MongoDB	Mesured3D + Cerberre
Consistency	***	*	***	*
Scalability	*	***	***	*
Flexibility	***	*	***	***
2D Geospatial Operations	***	*	**	***
3D Geospatial Operations	***	*	*	***
Data Import/Export	***	*	**	***

Regarding to the UDT explored use cases to improve accuracy and generate more reliable insights in future work, enriching the model with precise, standardized data is essential. For example, urban planning use cases could benefit from additional semantic information such as building renovation timelines, real estate prices, and population statistics. In energy-related applications like estimating solar potential, UDTs can provide actionable insights to promote green energy. By tracking energy production and consumption at the building level, UDTs can identify optimal locations for solar panel installation. Furthermore, developing advanced spatial solutions for shadow simulation will enable stakeholders to evaluate solar exposure for various urban components. This is particularly relevant in hot climates like Benguerir, where assessing building sun exposure can inform housing market analysis and renovation strategies. Urban planners can also use these insights to identify optimal locations for tree planting along roads, enhancing pedestrian comfort in a city known for its high temperatures. Finally, classifying buildings and calculating morphological parameters can provide valuable indicators to support urban structure optimization, renovation strategies, and sustainable city planning.

On the other hand developing a UDT in a data-constrained context such as Benguerir revealed that; the primary challenges extend beyond purely technical and non-technical considerations are largely rooted in organizational, institutional, and data governance limitations as depicted in this study [8]. Although the city is officially promoted as a smart and sustainable urban environment, our implementation experience demonstrated significant gaps in geospatial data availability, interoperability, and accessibility. In particular, fragmented data ownership across governmental institutions, the absence of updated authoritative datasets, and limited mechanisms for data sharing and real-time data access substantially affected the workflow development. These constraints necessitated pragmatic design decisions and adaptive mitigation strategies, including periodic data acquisition, and the use of inferred data within machine learning techniques or synthetic data for validation purposes. To provide a structured and transparent overview, the most criti-

cal challenges encountered during the implementation of the proposed UDT framework are classified into technical and non-technical categories and summarized alongside the adopted mitigation strategies in Table 9.

Table 9. Key challenges encountered during UDT development in Benguerir and adopted mitigation strategies.

Category	Identified Challenges	Mitigation Strategies in This Study
Data Collection	• Restricted access to authoritative geospatial datasets • Lack of updated urban plans (missing newly developed districts) • Absence of accessible real-time sensor data	• Hybrid use of authoritative datasets and newly collected data • Establishment of an open-source urban dataset intended to be curated and extended by the research and local community • Use of inferred data (e.g., building heights, number of floors) derived from remote sensing products and urban rules • Use of synthetic sensor data streams to validate the UDT architecture and data integration workflows
Data Integration and Interoperability	• Fragmented data ownership across public institutions • Lack of semantic information. • Lack of interoperable geospatial data infrastructure • Absence of standardized data provider services	• Adoption of standardized data models for long-term interoperability (CityGML as an authoritative semantic model) • Use of CityJSON and 3D Tiles as complementary representations optimized for web-based querying and visualization • Centralized mediation layer to support gradual integration of future real-time and institutional data sources • Modular system architecture allowing incremental data enrichment over time
Institutional and Organizational	• Limited coordination between Urban Agencies and Cadastral authorities • Lack of open data policies and formal data-sharing mechanisms • Administrative barriers to deploying and maintaining physical sensor networks	• Definition of a dedicated coordination team responsible for supervising the different stages of the Digital Twin lifecycle • Phased UDT development strategy aligned with institutional readiness and data availability • Use of proof-of-concept implementations to demonstrate feasibility and encourage stakeholder engagement • Design of an extensible sensor integration framework to support future physical sensor deployment

6. Conclusions

Urban Digital Twins represent a continuum of digital representation, ranging from basic virtual city models to fully interactive, bidirectional digital twins. In this study, we developed a comprehensive workflow for UDTs that encompasses the entire process; from 3D modelling of diverse city objects to their practical application in urban planning; enabling the creation of a virtual city that fosters data-driven solutions to real-world urban challenges. Our use cases in urban planning demonstrate that UDTs are essential for identifying urban deficiencies and supporting informed decision-making by city stakeholders.

From a technical perspective, building such a workflow requires a robust, reusable, and scalable software architecture that must efficiently split into connected components across

various instances, promote scalability and adaptability, and ensure fluent data pipelines for interoperability between disparate data sources. Our initial exploration of 3D modelling standards like CityGML successfully generated semantic 3D city models; however, we encountered challenges in data validation, integration, storage, and visualisation. This led us to investigate alternative approaches, such as CityJSON, which offers a comprehensive toolkit for optimised data storage, parsing in common languages (e.g., JavaScript and Python), web streaming, and rendering. Despite these advances, further research is needed to improve performance, mainly through on-the-fly 3D tile computation for additional abstraction levels and the development of optimised data schemas.

Additionally, significant challenges remain on the data integration side, particularly for dynamic data. Although the CityThings concept simplifies data pipelines for many use cases, future directions should focus on enhancing integration methods, especially when dealing with diverse datasets. For example, integrating data at various levels, as attempted in the CMD database, is promising, though current solutions like CityADEs and Dynamizer still have limitations.

On the web visualisation side, existing frameworks provide solutions for data sharing but currently struggle with 3D rendering and multi-source data integration. Emerging technologies, such as web-based game engines, promise to address these limitations. Moreover, initiatives like the Urban Digital Twins Interoperability Pilot, led by the OGC community, are critical to creating a complete UDT framework that ensures interoperability throughout the UDT lifecycle (OGC Urban Digital Twins Interoperability Pilot, <https://www.ogc.org/fr/initiatives/ogc-urban-digital-twin-interoperability/>).

Non-technically, the development of UDTs is impeded by limited data availability and the lack of clear legal frameworks for managing data confidentiality, sharing, and presentation. In addition, challenges arising from organizational hierarchies and collaboration further restrict progress. To address this need, we recommend the urgent establishment of accessible, interoperable geospatial data infrastructure systems to mitigate these gaps.

In conclusion, while our work has established a foundational workflow for the development of UDT and demonstrated their feasibility for implementation in Moroccan environments, and their significant potential to address urban challenges, further research and cross-disciplinary collaboration remain imperative. A comprehensive, interoperable, and resilient UDT framework could transform urban planning and management.

Limitations and Future Work

Despite the promising results achieved in this study, several limitations should be acknowledged: First, the developed UDT primarily relies on static datasets derived from photogrammetric restitution plans and existing urban planning documents. While this approach proved sufficient for generating geometrically and semantically consistent LoD1 and partially LoD2 city models, it limits the ability to represent dynamic urban processes or to perform real-time micro-simulations. The absence of dense, real-time, multi-source sensor data (e.g., traffic, energy consumption, environmental sensors) constrained the scope of system-level simulations and bidirectional data feedback within the UDT framework.

Second, the semantic enrichment process partially depends on rule-based assumptions derived from zoning regulations and planning documents, such as the estimation of the number of building floors based on building height and standardized floor-height thresholds. Although these rules are aligned with local urban planning regulations and were applied conservatively, they may introduce uncertainties when generalized to other urban contexts with different regulatory frameworks or construction practices.

Third, the evaluation of parsing, visualization, and performance was conducted in a controlled single-user environment under fixed hardware and browser conditions.

While this setup enabled a fair comparison between CityGML, CityJSON, and 3D Tiles representations, it does not fully capture scalability issues related to concurrent users, cloud-based deployments, or large-scale city-wide datasets.

Finally, although CesiumJS provided robust visualization capabilities, the current geoportal lacks advanced spatial analysis and interaction tools, such as on-the-fly spatial queries, area-based selection, and detailed measurement functions. These limitations restrict certain analytical workflows that are relevant for urban planners and decision-makers.

Future work will focus on addressing these limitations by integrating real-time and near-real-time data streams through standardized interfaces such as the SensorThings API, enabling dynamic simulations and feedback mechanisms within the UDT. In addition, extending the framework toward CityGML 3.0 and CityJSON-based dynamic modules will be explored to better support temporal data and evolving urban states. Performance evaluations will be expanded to multi-user and cloud-based environments to assess scalability and operational robustness. Finally, enhancing the geoportal with advanced spatial analysis tools and coupling it with complementary web-mapping libraries will further improve user interaction and analytical capabilities, strengthening the applicability of the proposed UDT framework for sustainable urban planning and infrastructure management.

Author Contributions: Conceptualization, R.H. and O.B.; Methodology, R.H. and O.B.; Software, O.B.; Validation, R.H. and O.B.; Formal Analysis, O.B.; Investigation, R.H. and O.B.; Resources, R.H. and H.R.; Data Curation, R.H. and O.B.; Writing—Original Draft Preparation, O.B.; Writing—Review and Editing, R.H., O.B. and H.R.; Visualization, R.H. and H.R.; Supervision, R.H. and H.R.; Project Administration, R.H.; Funding Acquisition, H.R. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by UM6P-Citinnov.

Data Availability Statement: The data presented in this study are available on request from the corresponding author.

Acknowledgments: The research work was undertaken by a very limited team. I was fortunate to have the consistent support and motivation of Bouchra Monadel, formerly a GIS Engineer at the TOP Lab, although not an official member of the project team, our frequent and insightful technical discussions, often over coffee, were invaluable. I also extend my gratitude to the lab assistants and administrative agents for their essential support. Last but certainly not least, I extend my sincere thanks to Walid El Berz, a student assistant in my university department, for his dependable assistance.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

DSM	Digital Surface Model
API	Application Programming Interface
ADE	Application Domain Extension
BIM	Building Information Modeling
CDT	City Digital Twin
GIS	Geographic Information System
GML	Geography Markup Language
ICT	Information and Communication Technology
JSON	JavaScript Object Notation
LoD	Level of Detail
MQTT	Message Queuing Telemetry Transport
UDT	Urban Digital Twin
XML	Extensible Markup Language

Appendix A

Appendix A.1. 3D Tiles Parameter Overview

- Theme: Controls custom input behavior, specified through the Flask API.
- Level of Detail (LOD): Defines the detail level of the dataset; in our case, the LOD1.
- Mode: The Flask API operates in two modes: 1 (creator and server) or 0 (server only). In mode 1, the tile creator runs before the server, whereas mode 0 is used when the 3D tiles are already prepared in the database and only need to be served.
- Cluster Number: Specifies the number of clusters at the top and bottom levels when applying the hierarchical k-means algorithm.
- Index Flag: Determines if the b3dm or binary glTF is indexed (1) or non-indexed (0).
- b3dm and glb Flags: Controls whether the b3dm or glTF is pre-composed (1) or not composed (−1). For instance, if both the index flag and glb flag are set to 1, an indexed binary glTF will be composed ahead of time in the database.
- Property: Defines the properties to query from the dataset, provided as a list. An exception is raised if a specified property is unsupported.
- Filter: Allows for attribute or spatial filtering. For example: “and height > 10”. If left blank (the default), no filter is applied.

The main application function, created using Flask, uses the JSON file as input to configure the data fetching process based on the configurations represented in the JSON file.

Appendix A.2. Example of 3D Tiles Streaming Code

A code example used to fetch the created 3Dtiles.

Listing A1. Flask Routing for Fetching Precomposed b3dm Files.

```
@app.route("/tiles/<string:tile_name>.b3dm")
def tiles_one_tile(tile_name):
    route_start_time = time.time()
    # database connection
    conn = get_db()
    # Create a cursor object
    cursor = conn.cursor()
    tile_id = int(tile_name)
    if b3dm_flag == 1:
        sql = "SELECT b3dm from hierarchy where temp_tid = {0} and level
=2".format(tile_id)
        cursor.execute(sql)
        results = cursor.fetchall()
        b3dm_bytes = results[0][0]
        conn.commit()
        print("Approach 3: fetch precomposed b3dm successfully:
{0}".format(tile_id))
```

References

1. Shahat, E.; Hyun, C.T.; Yeom, C. City digital twin potentials: A review and research agenda. *Sustainability* **2021**, *13*, 3386. [CrossRef]
2. Schrotter, G.; Hürzeler, C. The digital twin of the city of Zurich for urban planning. *PFG—J. Photogramm. Remote Sens. Geoinf. Sci.* **2020**, *88*, 99–112. [CrossRef]
3. Hämäläinen, M. Smart city development with digital twin technology. In *Proceedings of the 33rd Bled eConference-Enabling Technology for a Sustainable Society, Online, 28–29 June 2020*; University of Maribor: Maribor, Slovenia, 2020.
4. Gobeawan, L.; Lin, E.S.; Tandon, A.; Yee, A.T.K.; Khoo, V.H.S.; Teo, S.N.; Yi, S.; Lim, C.W.; Wong, S.T.; Wise, D.J.; et al. Modeling trees for virtual singapore: From data acquisition to citygml models. *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.* **2018**, *42*, 55–62. [CrossRef]

5. D'Hauwers, R.; Walravens, N.; Ballon, P. From an inside-in towards an outside-out urban digital twin: Business models and implementation challenges. *ISPRS Ann. Photogramm. Remote Sens. Spat. Inf. Sci.* **2021**, *8*, 25–32. [\[CrossRef\]](#)
6. Biljecki, F.; Stoter, J.; Ledoux, H.; Zlatanova, S.; Cöltekin, A. Applications of 3D city models: State of the art review. *ISPRS Int. J.-Geo-Inf.* **2015**, *4*, 2842–2889. [\[CrossRef\]](#)
7. Lei, B.; Janssen, P.; Stoter, J.; Biljecki, F. Challenges of urban digital twins: A systematic review and a Delphi expert survey. *Autom. Constr.* **2023**, *147*, 104716. [\[CrossRef\]](#)
8. Weil, C.; Bibri, S.E.; Longchamp, R.; Golay, F.; Alahi, A. Urban digital twin challenges: A systematic review and perspectives for sustainable smart cities. *Sustain. Cities Soc.* **2023**, *99*, 104862. [\[CrossRef\]](#)
9. Jeddoub, I.; Nys, G.-A.; Hajji, R.; Billen, R. Data integration across urban digital twin lifecycle: A comprehensive review of current initiatives. *Ann. GIS* **2025**, *31*, 367–386. [\[CrossRef\]](#)
10. Kanna, K.; Kolbe, T.H. Automatic Enrichment of Semantic 3D City Models using Large Language Models. *ISPRS Ann. Photogramm. Remote Sens. Spat. Inf. Sci.* **2025**, *10*, 105–112. [\[CrossRef\]](#)
11. Xu, H.; Zlatanova, S.; Li, X.; Wachowicz, M.; Batty, M. Towards Fully Automated City Operations: Integrating Agentic AI with Urban Digital Twins. *SSRN Electron. J.* 2025. Available online: https://papers.ssrn.com/sol3/papers.cfm?abstract_id=5596992 (accessed on).
12. Our World in Data Team. *Make Cities Inclusive, Safe, Resilient and Sustainable*; Our World in Data: Oxford, UK, 2023. Available online: <https://ourworldindata.org/sdgs/sustainable-cities> (accessed on 15 July 2025).
13. Hassan, A.M.; Lee, H. The paradox of the sustainable city: Definitions and examples. *Environ. Dev. Sustain.* **2015**, *17*, 1267–1285. [\[CrossRef\]](#)
14. Zheng, H.W.; Shen, G.Q.; Wang, H. A review of recent studies on sustainable urban renewal. *Habitat Int.* **2014**, *41*, 272–279. [\[CrossRef\]](#)
15. Ng, P.T. Embracing emerging technologies: The case of the Singapore Intelligent Nation 2015 vision. In *Regional Innovation Systems and Sustainable Development: Emerging Technologies*; IGI Global: Hershey, PA, USA, 2011; pp. 115–123.
16. IT Strategic Headquarters. *I-Japan Strategy 2015*; Striving to Create a Citizen-Driven, Reassuring & Vibrant Digital Society; Prime Minister's Office of Japan: Tokyo, Japan, 2009.
17. Kylili, A.; Fokaides, P.A. European smart cities: The role of zero energy buildings. *Sustain. Cities Soc.* **2015**, *15*, 86–95. [\[CrossRef\]](#)
18. Rharbi, N.; Günseli Demirkol, H. Impact of Sustainability Transition in Moroccan Cities' Identity: The Case of Benguerir. *ICONARP Int. J. Archit. Plan.* **2023**, *11*, 88–106. [\[CrossRef\]](#)
19. Loudyi, N. La quête de la ville durable: Le besoin de vocation. *Afr. Mediterr. J. Archit. Urban.* **2022**, *4*, 146–154, ISSN 2665-7953
20. Kaitouni, S.I.; Ait Abdelmoula, I.; Es-sakali, N.; Mghazli, M.O.; Er-retby, H.; Zoubir, Z.; El Mansouri, F.; Ahachad, M.; Brigui, J. Implementing a Digital Twin-based fault detection and diagnosis approach for optimal operation and maintenance of urban distributed solar photovoltaics. *Renew. Energy Focus* **2024**, *48*, 100530. [\[CrossRef\]](#)
21. Sofia, H.; Anas, E.; Faïz, O. Mobile mapping, machine learning and digital twin for road infrastructure monitoring and maintenance: Case study of mohammed VI bridge in Morocco. In Proceedings of the 2020 IEEE International Conference of Moroccan Geomatics (Morgeo), Rabat, Morocco, 11–13 May 2020; pp. 1–6.
22. Serbouti, I.; Chenal, J.; Tazi, S.A.; Baik, A.; Hakdaoui, M. Digital Transformation in African Heritage Preservation: A Digital Twin Framework for a Sustainable Bab Al-Mansour in Meknes City, Morocco. *Smart Cities* **2025**, *8*, 29. [\[CrossRef\]](#)
23. Elomari, E.; El Mansouri, L.; Ghinane, H.; El-Ayachi, M. Opportunities and challenges of implementing a spatial data infrastructure case study of Rabat-Salé-Kenitra state in Morocco. *Afr. J. Land Policy Geospat. Sci.* **2021**, *4*, 212–231.
24. Grieves, M.; Vickers, J. Origins of the digital twin concept. *Fla. Inst. Technol.* **2016**, *8*, 3–20.
25. Qi, Q.; Tao, F. Digital twin and big data towards smart manufacturing and industry 4.0: 360 degree comparison. *IEEE Access* **2018**, *6*, 3585–3593. [\[CrossRef\]](#)
26. Rosen, R.; Von Wichert, G.; Lo, G.; Bettenhausen, K.D. About the importance of autonomy and digital twins for the future of manufacturing. *IFAC-Pap.* **2015**, *48*, 567–572. [\[CrossRef\]](#)
27. Singh, M.; Fuenmayor, E.; Hinchy, E.P.; Qiao, Y.; Murray, N.; Devine, D. Digital twin: Origin to future. *Appl. Syst. Innov.* **2021**, *4*, 36. [\[CrossRef\]](#)
28. Ledoux, H.; Arroyo Otori, K.; Kumar, K.; Dukai, B.; Labetski, A.; Vitalis, S. CityJSON: A compact and easy-to-use encoding of the CityGML data model. *Open Geospat. Data Softw. Stand.* **2019**, *4*, 4. [\[CrossRef\]](#)
29. Jeddoub, I.; Nys, G.A.; Hajji, R.; Billen, R. Digital twins for cities: Analyzing the gap between concepts and current implementations with a specific focus on data integration. *Int. J. Appl. Earth Obs. Geoinf.* **2023**, *122*, 103440. [\[CrossRef\]](#)
30. Biljecki, F.; Kumar, K.; Nagel, C. CityGML application domain extension (ADE): Overview of developments. *Open Geospat. Data Softw. Stand.* **2018**, *3*, 13. [\[CrossRef\]](#)
31. Agugiaro, G.; Benner, J.; Cipriano, P.; Nouvel, R. The Energy Application Domain Extension for CityGML: Enhancing interoperability for urban energy simulations. *Open Geospat. Data Softw. Stand.* **2018**, *3*, 2. [\[CrossRef\]](#)

32. Kutzner, T.; Chaturvedi, K.; Kolbe, T.H. CityGML 3.0: New functions open up new applications. *PFG—J. Photogramm. Remote Sens. Geoinf. Sci.* **2020**, *88*, 43–61. [[CrossRef](#)]
33. Santhanavanich, T.; Coors, V. CityThings: An integration of the dynamic sensor data to the 3D city model. *Environ. Plan. B Urban Anal. City Sci.* **2021**, *48*, 417–432. [[CrossRef](#)]
34. Rantanen, T.; Julin, A.; Virtanen, J.P.; Hyyppä, H.; Vaaja, M.T. Open geospatial data integration in game engine for urban digital twin applications. *ISPRS Int. J.-Geo-Inf.* **2023**, *12*, 310. [[CrossRef](#)]
35. Yao, Z.; Nagel, C.; Kunde, F.; Hudra, G.; Willkomm, P.; Donaubaue, A.; Adolphi, T.; Kolbe, T.H. 3DCityDB—A 3D geodatabase solution for the management, analysis, and visualization of semantic 3D city models based on CityGML. *Open Geospat. Data Softw. Stand.* **2018**, *3*, 5. [[CrossRef](#)]
36. Powalka, L.; Poon, C.; Xia, Y.; Meines, S.; Yan, L.; Cai, Y.; Stavropoulou, G.; Dukai, B.; Ledoux, H. cjdb: A simple, fast, and lean database solution for the CityGML data model. In Proceedings of the International 3D GeoInfo Conference, London, UK, 11–13 September 2023; pp. 781–796.
37. Charitonidou, M. Urban scale digital twins in data-driven society: Challenging digital universalism in urban planning decision-making. *Int. J. Archit. Comput.* **2022**, *20*, 238–253. [[CrossRef](#)]
38. Mylonas, G.; Kalogeras, A.; Kalogeras, G.; Anagnostopoulos, C.; Alexakos, C.; Muñoz, L. Digital twins from smart manufacturing to smart cities: A survey. *IEEE Access* **2021**, *9*, 143222–143249. [[CrossRef](#)]
39. Petrova-Antonova, D.; Ilieva, S. Methodological framework for digital transition and performance assessment of smart cities. In Proceedings of the 2019 4th International Conference on Smart and Sustainable Technologies (SpliTech), Split, Croatia, 18–21 June 2019; pp. 1–6.
40. Major, P.; Li, G.; Hildre, H.P.; Zhang, H. The use of a data-driven digital twin of a smart city: A case study of Ålesund, Norway. *IEEE Instrum. Meas. Mag.* **2021**, *24*, 39–49. [[CrossRef](#)]
41. ISO 19107:2003; Geographic Information-Spatial Schema. International Organization for Standardization (ISO): Geneva, Switzerland, 2003.
42. Yang, Y. Directly Serving 3D Tiles from a Geo-DBMS. Master's Thesis, Delft University of Technology, Faculty of Architecture and the Built Environment, Delft, The Netherlands, 2024. Available online: <https://resolver.tudelft.nl/uuid:096553de-711c-49fd-8428-50ee6203b4ce> (accessed on 15 June 2024).
43. Kasprzyk, J.P.; Nys, G.A.; Billen, R. Towards a multi-database CityGML environment adapted to big geodata issues of urban digital twins. *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.* **2024**, *48*, 101–106. [[CrossRef](#)]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.