

Report

Urban development focuses on planning, building, and managing city infrastructure to achieve sustainable growth that balances resident needs, economic goals, and environmental health. Core principles include **Sustainable Growth**, ensuring that cities expand without overusing resources; **Resource Management**, involving efficient use of energy, water, and materials; **Citizen Satisfaction**, ensuring accessible housing, healthcare, and public services to maintain a thriving population; and **Government Policy**, where tax and budget allocation drive development aligned with social equity and environmental goals.

Role of City Components

- **Buildings:** Residential, commercial, and industrial buildings serve different purposes and support population and economic growth.
- **Utilities:** Essential services like power, water, and waste management are vital for livability and functionality.
- **Transportation:** Roads, transit, and airports reduce congestion and improve access to jobs and services.
- **Citizens:** Population needs drive demand for housing, jobs, and amenities, with satisfaction influencing city growth.
- **Government:** Manages taxes, budget allocation, and policy to shape economic stability and citizen welfare.
- **Resources:** Materials, energy, and budget must be managed sustainably to support growth and infrastructure.
- **Taxes:** Funding from taxes supports city operations, with rates influencing both citizen satisfaction and economic stability.
- **City Growth:** Population, housing, and infrastructure must grow together to maintain service quality and economic stability.

Influence on Design

Components like **citizen satisfaction** informed resource allocation and policy impact on economic growth, while utilities and transportation systems addressed livability and productivity. The government's role in taxes and policies enabled dynamic management of resources and citizen satisfaction.

Assumptions and Key Design Decisions

- **Sustainable Growth** and **Dynamic Citizen Needs** guided resource allocation.
- **Modular Building System** and **Resource Tracking** provided flexibility and realistic constraints.
- **Policy and Tax Management** directly impacted citizen satisfaction and growth.

These choices created a balanced model emphasizing the interconnectedness of city components, simulating realistic urban development.

🔍 **Factory Pattern** - *Building Creation*

- The Factory pattern is used to create different types of buildings, such as residential, commercial, and industrial buildings. This design choice allows the program to dynamically create buildings based on player or system input, enhancing modularity and scalability.
- In the UML, this is represented by classes like ResidentialBuilding, IndustrialBuilding, and CommercialBuilding, which could be produced by a building factory.

🔍 **Singleton Pattern** - *CityHall*

- The Singleton pattern ensures that only one instance of CityHall exists throughout the simulation. This is likely because CityHall serves as the central administration point, coordinating high-level decisions for the city.
- In the diagram, CityHall is depicted as a unique component, signifying its centralized role.

🔍 **Mediator Pattern** - *Inter-Component Communication*

- The Mediator pattern is applied to facilitate communication between various components, such as when roads or buildings are added to the simulation. It helps coordinate changes without direct dependencies between components.
- This is suggested in the structure, where CityHall (central mediator) component communicates with other elements like ResidentialComponent or IndustrialBuilding.

🔍 **State Pattern** - *Road Traffic State*

- Roads in the city use the State pattern to manage traffic conditions, like low, moderate, and high traffic states. This pattern enables dynamic transitions between states, depending on the road usage, and allows other components to react based on the current state.
- The states LowTrafficState, ModerateTrafficState, and HighTrafficState are shown in the UML diagram as subclasses of a common TrafficState.

❓ **Strategy Pattern** - *Political System*

- The Strategy pattern provides flexibility in handling different political approaches within the city's governance. Political strategies like Authoritarian, Communist, and Democratic can be swapped dynamically, allowing the simulation to reflect different governance styles.
- The UML shows these strategies as distinct classes under the PoliticalSystem interface.

❓ **Composite Pattern** - *Residential Building and Complex Management*

- The Composite pattern organizes buildings within a hierarchy, such as grouping multiple residential buildings into a ResidentialComplex. This pattern allows the program to treat individual buildings and complexes uniformly, streamlining management and interactions.
- The UML diagram includes ResidentialComponent as a composite, with ResidentialBuilding and ResidentialComplex as leaf and composite elements, respectively.

❓ **Decorator Pattern** - *Extending Industrial Building Functionality*

- The Decorator pattern is used to add additional functionalities to industrial buildings. This could include enhancements like adding pollution control or boosting productivity without modifying the base IndustrialBuilding class.
- The OperationDecorator in the UML indicates this use, with IndustrialBuilding as a base class and the decorator adding extra operations.

❓ **Chain of Responsibility Pattern** - *Satisfaction Handler*

- The Chain of Responsibility pattern is applied to manage citizen satisfaction in a chain, where different handlers address various satisfaction-related concerns. This approach allows for flexible processing of citizen complaints or feedback, with each handler focusing on a specific aspect.
- Although not explicitly labeled in the UML, there may be a satisfaction chain where issues flow through multiple classes that address different needs.

❓ **Prototype Pattern** - *Presets for Residential Complex*

- The Prototype pattern is used to store a copy of a ResidentialComplex configuration as a preset. This allows for efficient cloning of preset configurations for new residential complexes, enhancing usability.
- A copy mechanism might be implemented in ResidentialComplex to facilitate this cloning.

❓ **Iterator Pattern** - *Iterating Over Building Satisfaction*

- The Iterator pattern is used to traverse through buildings, for calculating citizen satisfaction. This pattern simplifies accessing each building in a consistent manner, without exposing the underlying structure.
- The UML implies iteration over ResidentialBuilding components for satisfaction calculation.

Strategy for Branching

We implement a branching strategy where a dedicated branch is created for each design pattern. In addition, we establish supplementary branches specifically for merging and debugging purposes. Once the development and testing are complete, the finalized code is merged and pushed to the main branch. This approach allows for organized and efficient development while maintaining a clear history of changes and features.