Lift Control System

General Description

The Lift Control System simulates the operation of two elevators in a 7-story building (floors 0–6). The system is implemented using modern web technologies: HTML, CSS, JavaScript, and jQuery.

It responds to user interactions, handles elevator calls and destinations, manages state changes, and provides both visual and audio feedback.

System Architecture

Models

Building – Represents the entire building, including floors and elevators.
Elevator – Represents an individual elevator with its own state and logic.
Floor – Represents a floor with call buttons and relevant data.

Controllers

ElevatorController – Monitors elevator states and events, updates the UI.
ElevatorScheduler – Selects the most suitable elevator for a call.
EmergencyHandler – Handles emergency-related events and transitions.
AnimationController – Manages visual animations such as elevator movement and door actions.

Helper Classes

Settings – Stores configurable system parameters such as speed and door timings.

Basic Operation

1. User Interaction
    - Elevators can be called from each floor using up/down buttons
    - Destination floors can be selected from inside the elevator
2. Decision Logic for Elevator Selection
    - Distance from the calling floor is considered
    - Current movement direction of each elevator is taken into account
    - In case of equal distance, the elevator on the lower floor has priority
3. Elevator States: IDLE, MOVING_UP, MOVING_DOWN, DOOR_OPENING, DOOR_CLOSING, LOADING, EMERGENCY.

Functionality

Core Features

- Elevator call from floors

- Destination selection from inside the elevator
- Manual door open/close
- Emergency handling
- Light/Dark mode switching
- Adjustable elevator speed and door open time

Additional Features

- Sound effects (door, bell, arrival, emergency)
- Responsive UI for both desktop and mobile
- Visual indicators for direction and state

Elevator Scheduling Logic

1. An elevator completes all requests in its current direction before changing direction.
2. Destinations are ordered according to the current movement direction.
3. If multiple elevators are available, selection is based on:
    - Proximity to the calling floor
    - Direction of movement
    - Lower position in the building (if equidistant)

Event Handling

- The system uses event-based communication between components.
- Elevators emit events when their state changes.
- The ElevatorController listens to these events to update the interface and trigger animations or further logic accordingly.