



Розробка ядра CRM системи із візуалізацією процесу виконання проєктів для компанії Omega Telecom

ВИКОНАЛА: МАТВИЄНКО ІРИНА ВАЛЕНТИНІВНА, БП-4, ІПЗ

НАУКОВИЙ КЕРІВНИК: СТ. ВИКЛ. БОРОЗЕННИЙ СЕРГІЙ ОЛЕКСАНДРОВИЧ

Постановка задачі

- ▶ Спроекувати та розробити ядро CRM системи у вигляді веб-застосунку для компанії Omega Telecom. Система повинна автоматизувати та візуалізувати процес виконання проєктів, що складаються із послідовності завдань.

Недоліки проаналізованих CRM систем

► SendPulse

- всі працівники можуть редагувати всі угоди
- є лише угоди, що не розподіляються на завдання
- відсутня функціональність генерації звітів

► Monday work management

- надлишкова функціональність

► KeepinCRM

- максимум два користувачі
- є лише завдання, що не групуються за проектами

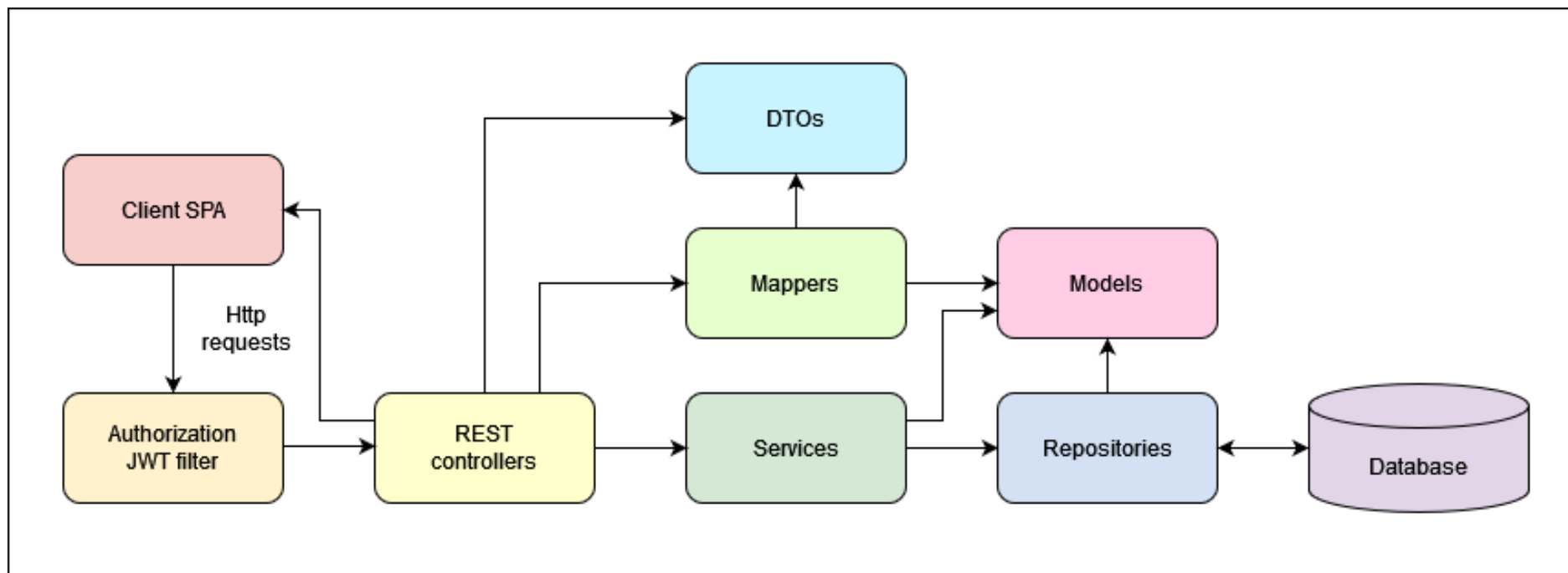
Всі

- Kanban-дошка
- відсутня перевірка завдань більш кваліфікованими працівниками
- немає обов'язкового прикріплення договорів із контрагентами

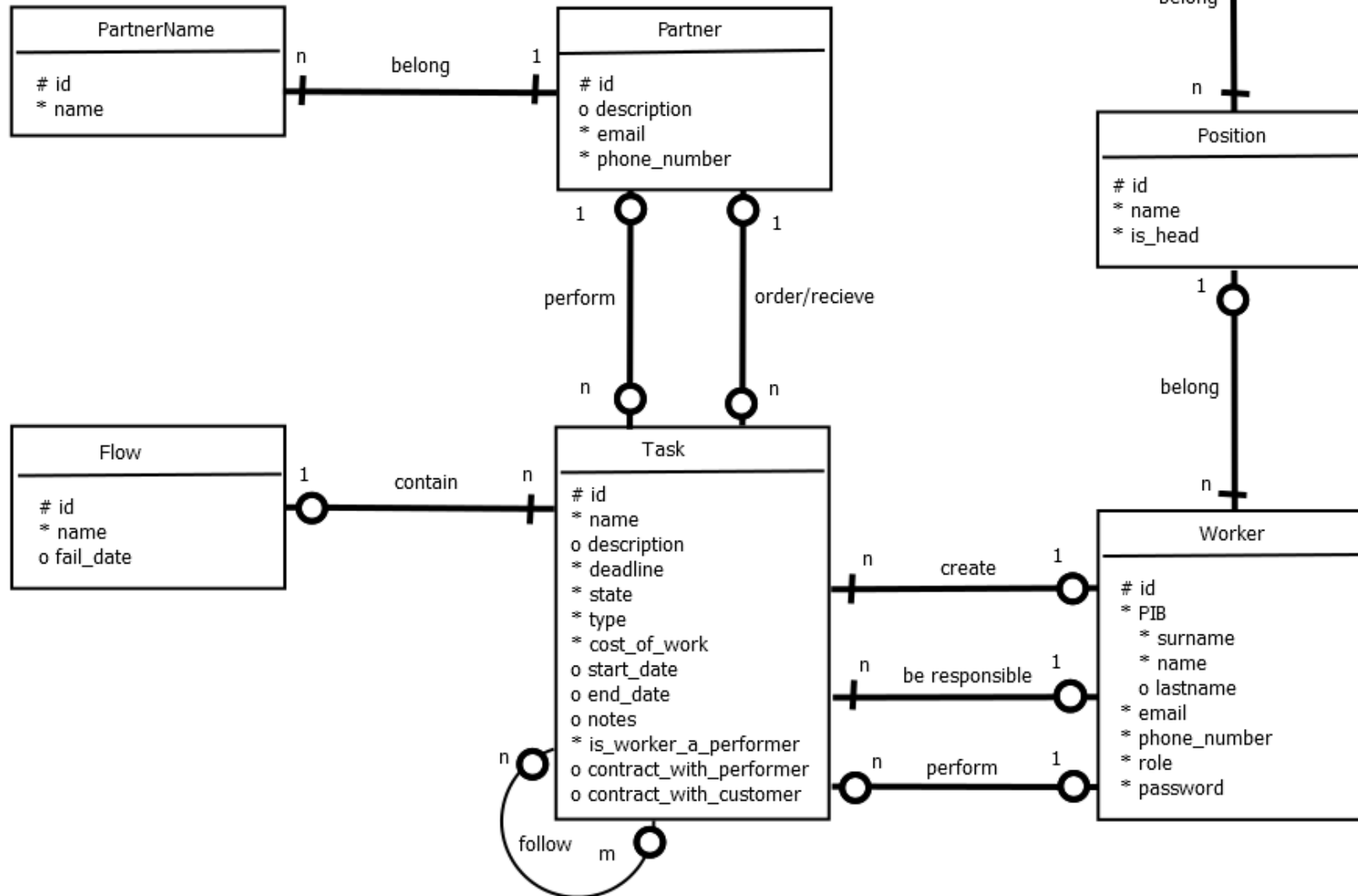
Підходи до вирішення задачі

- ▶ Архітектурні підходи
 - моноліт
 - мікросервіси
 - **клієнт-серверна архітектура**
- ▶ Інструменти для серверу
 - Node.js
 - **Spring Boot**
- ▶ Створення діаграми послідовності завдань
 - самотійно
 - **стороння бібліотека**

Архітектура веб-застосунку

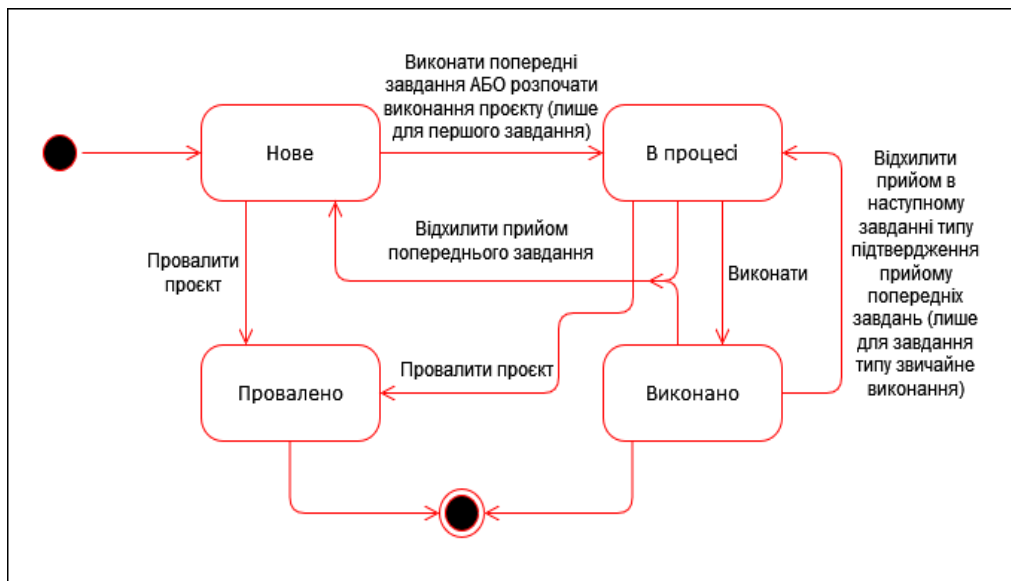


ER-МОДЕЛЬ

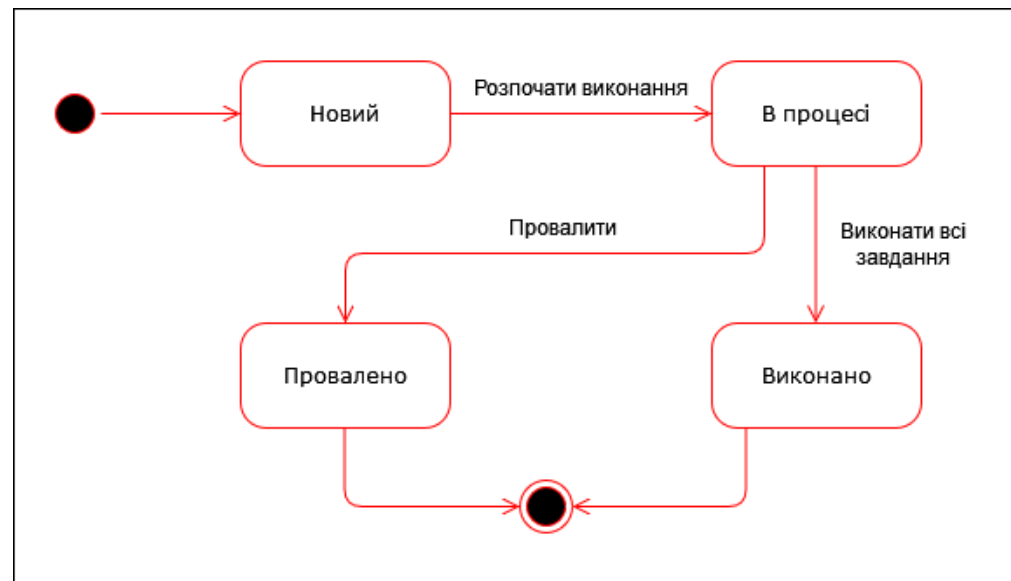


Діаграми станів

Завдання



Проєкт

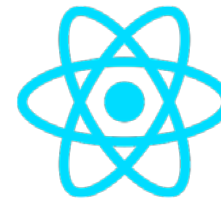


Використаний інструментарій

- ▶ Java + Spring Boot
- ▶ Spring MVC
- ▶ Spring Security
- ▶ Java Jwt
- ▶ iText
- ▶ PostgreSQL
- ▶ React + Redux
- ▶ GoJS
- ▶ Bootstrap



PostgreSQL



React JS



Bootstrap

API (1)

▶ Облікові записи:

- POST /api/login
- PUT /api/password

▶ Працівники:

- GET /api/workers
- GET /api/workers/:id
- GET /api/workers/me
- POST /api/workers
- PUT /api/workers/:id
- DELETE /api/workers/:id

▶ Відділи:

- GET /api/departments
- GET /api/departments/:id
- POST /api/departments
- PUT /api/departments/:id
- DELETE /api/departments/:id

▶ Посади:

- GET /api/positions
- GET /api/positions/:id
- POST /api/positions
- PUT /api/positions/:id
- DELETE /api/positions/:id

API (2)

▶ Компанії-контрагенти

- GET /api/partners
- GET /api/partners/:id
- POST /api/partners
- PUT /api/partners/:id
- DELETE /api/partners/:id

▶ Завдання

- PATCH /api/tasks/:id/notes
- PATCH /api/tasks/:id/performer
- POST /api/tasks/:id/finish
- POST /api/tasks/:id/approve_previous
- POST /api/tasks/:id/reject_previous

▶ Проекти

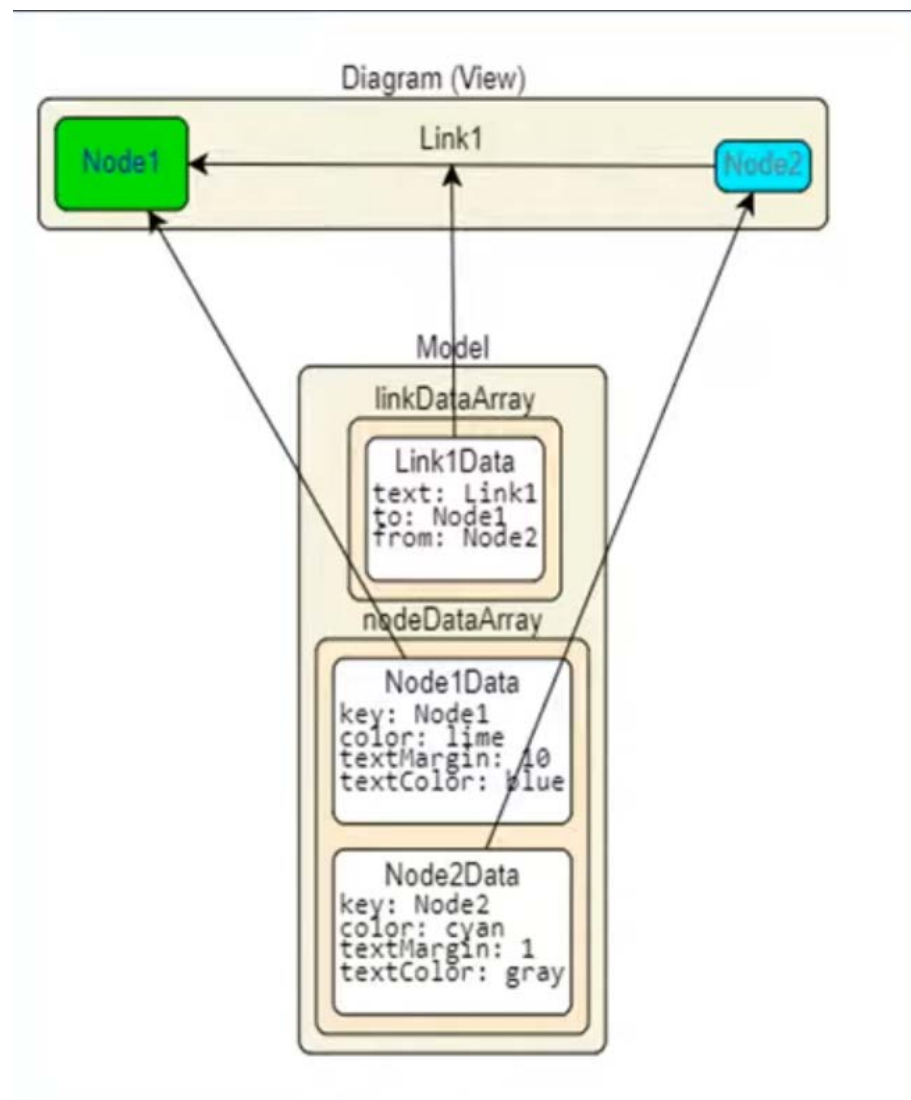
- GET /api/flows/all_allowed
- GET /api/flows/:id
- POST /api/flows
- PUT /api/flows/:id
- DELETE /api/flows/:id
- POST /api/flows/start
- POST /api/flows/fail

▶ Звіти

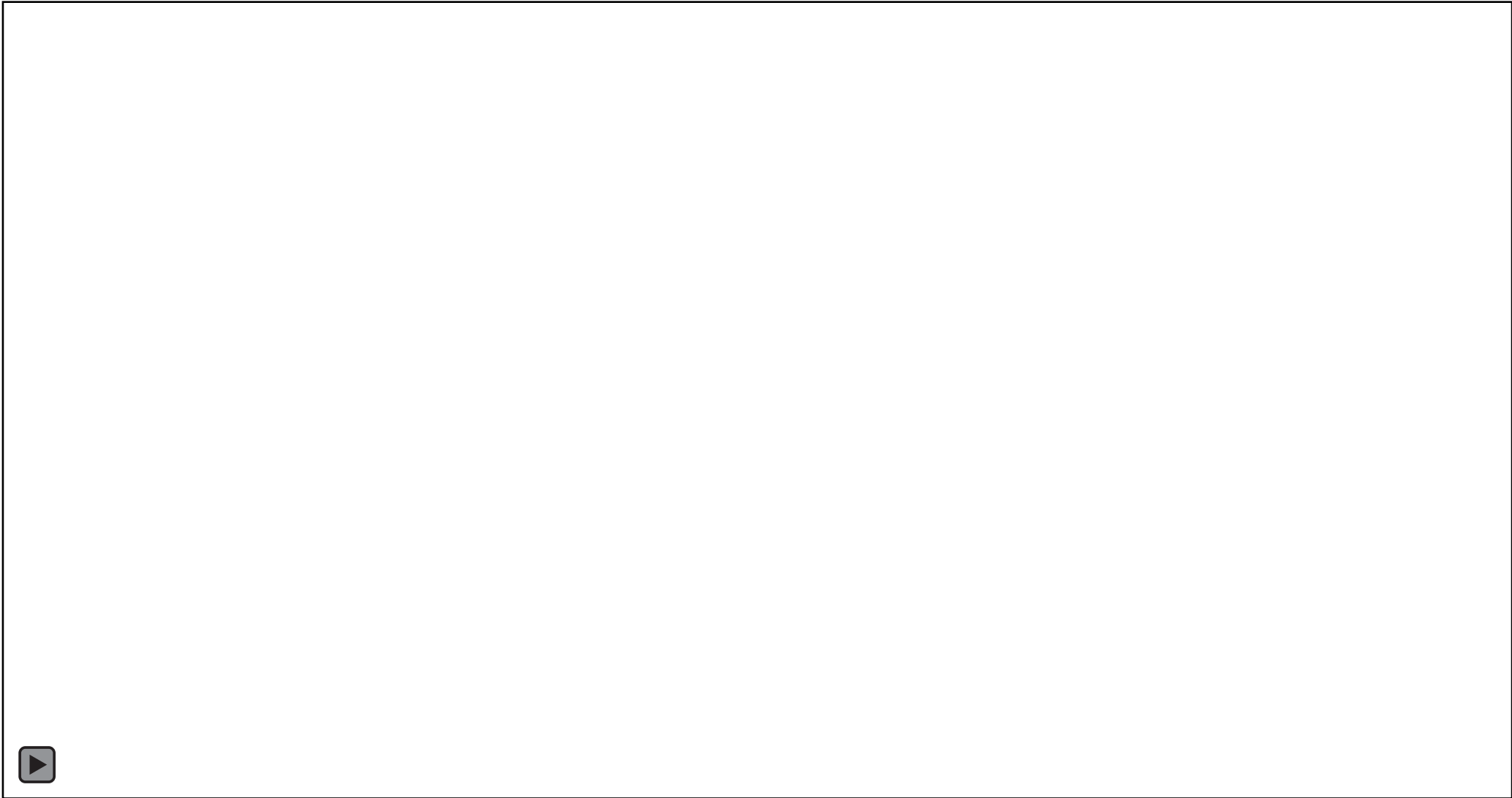
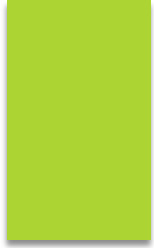
- GET /api/reports/workers/generate
- GET /api/reports/flows/generate
- GET /api/reports/workers
- GET /api/reports/flows

Бібліотека GoJS

- ▶ Canvas в основі
- ▶ model-view архітектура
- ▶ шаблони для nodes та зв'язків
- ▶ bindings
- ▶ обробники подій



Діаграма послідовності завдань



Результати

- ▶ Спроектовано та розроблено ядро CRM-системи у вигляді веб-застосунку із повним запланованим функціоналом, що покриває всі визначені вимоги користувачів для трьох ролей із чітким розподіленням прав доступу.
- ▶ Застосунок візуалізує послідовність завдань та поточну стадію виконання проєктів за допомогою діаграм GoJS, що спрощує контроль над прогресом проєкту та покращує розуміння для працівників, коли потрібно виконувати свої задачі.
- ▶ Реалізоване автоматичне формування звітів полегшує оцінку продуктивності роботи окремих працівників та компанії в цілому.

ВИСНОВКИ

- ▶ Бібліотека GoJS має детальну документацію, достатню кількість готових діаграм-прикладів із коментарями до коду, зручна у використанні та надає величезний набір параметрів для кастомізації діаграм.
- ▶ Розроблений веб-застосунок готовий до впровадження у роботу компанії Omega Telecom. За потреби, система може бути легко розширена додатковою функціональністю (наприклад, шаблонами послідовності завдань для проєктів) завдяки чіткому розподілу відповідальності між її рівнями та класами. Система також може використовуватися (можливо, з певними модифікаціями) в інших компаніях, робота яких побудована у форматі виконання проєктів.