

**ДОДАТОК А**  
Слайди презентації

**Дослідження методів  
оптимізації систем керування  
робочим процесом**

Виконав: ст. гр. Болотов Д.В. Керівник: проф. Бондарев В.М.

Рисунок А.1 – Вступний слайд

**Актуальність дослідження**

- керування робочим процесом;
- фактори що впливають на успішність проекту;
- методи у керуванні робочим процесом;
- пріоритезація та моніторинг витраченого на виконання задач часу.

2

Рисунок А.2 – Актуальність дослідження

## Керування проектами

Управління проектами – область знань з планування, організації та керування ресурсами з метою успішного досягнення цілей та завершення завдань

Рисунок А.3 – Керування проектами



Рисунок А.4 – Трикутник управління проектами

## Управління часом

5

Це дія або процес тренування контролю над часом, витраченим на конкретні види діяльності, основною ціллю якого є збільшення ефективності та продуктивності.

Рисунок А.5 – Управління часом

## Методи в управлінні часом

6

- метод Альп;
- ABC аналіз;
- матриця Ейзенхауера;
- Метод Помадоро.

Рисунок А.6 – Методи в управлінні часом

## Мета роботи

Метою роботи є проектування та реалізація додатку, основним напрямком якого є оптимізація процесів керування проектом сфокусувавшись на області управління часом.

Рисунок А.7 – Мета роботи

## Постановка задачі

- провести аналіз, концептуальне та UML-моделювання предметної галузі;
- розробити програмну реалізацію обраних методів оптимізації;
- розробити програмну реалізацію веб-системи;

Рисунок А.8 – Постановка задачі

## Аналіз аналогів | RescueTime

9

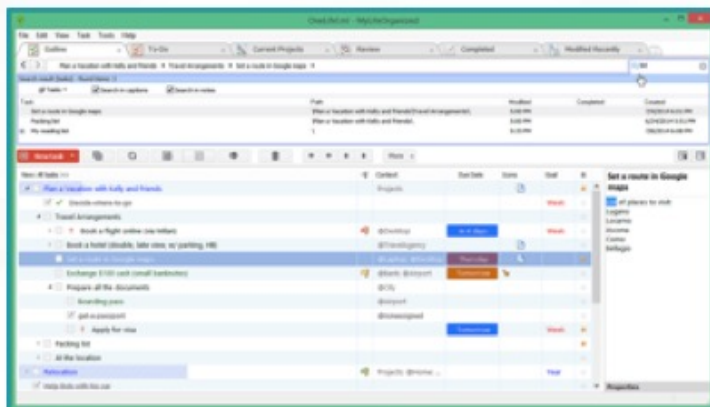


- таймери на виконання задач;
  - блокування відволікаючих сайтів;
  - аналіз витраченого часу.
- веб версія;
  - інструменти для пріоритизації задач;
  - занадто висока ціна.

Рисунок А.9 – Аналіз аналогів

## Аналіз аналогів | MyLifeOrganized

10



- нотифікації згідно місцезнаходження;
  - дерево списків.
- обмежений функціонал;
  - складний інтерфейс;
  - система моніторингу витраченого часу.

Рисунок А.10 – Аналіз аналогів

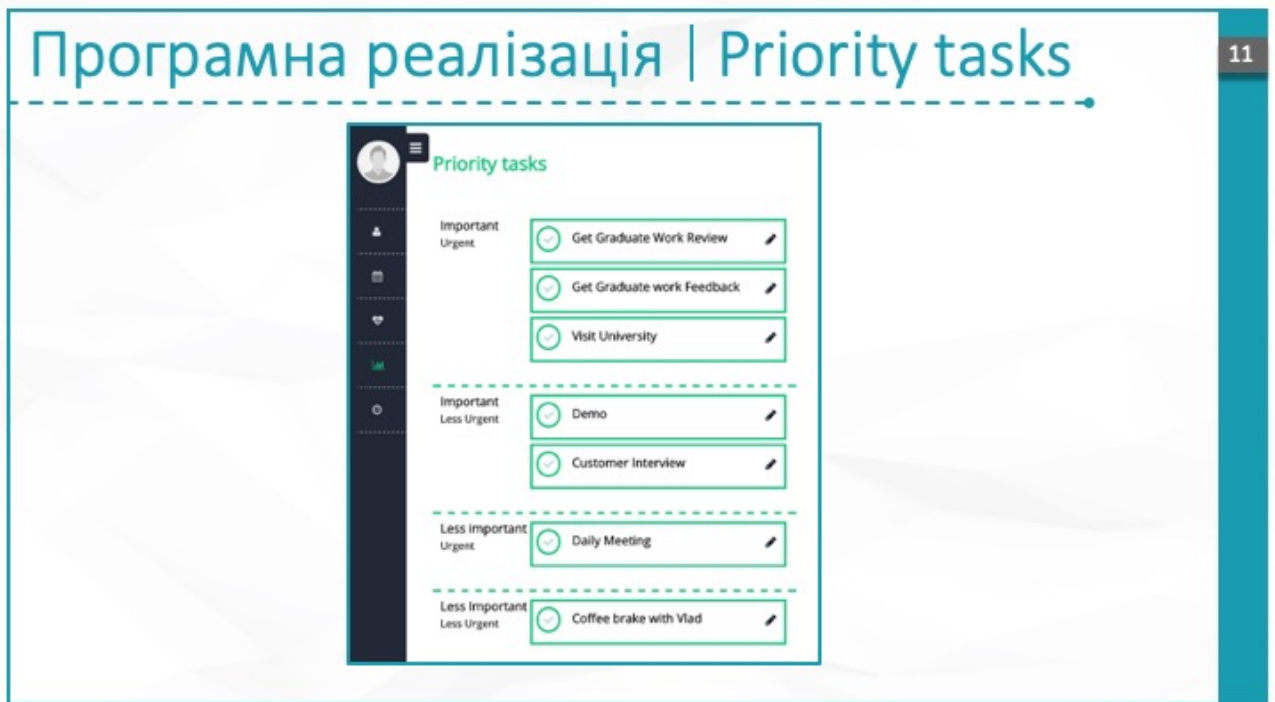


Рисунок А.11 – Програмна реалізація

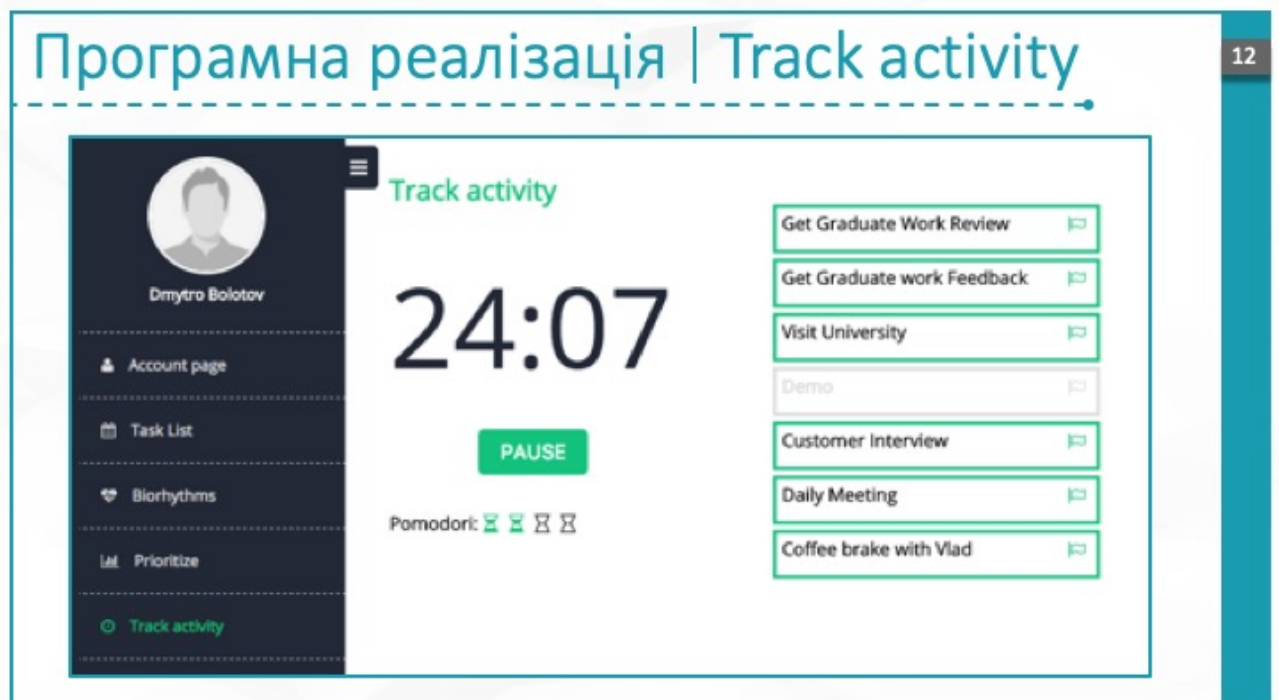


Рисунок А.12 – Програмна реалізація

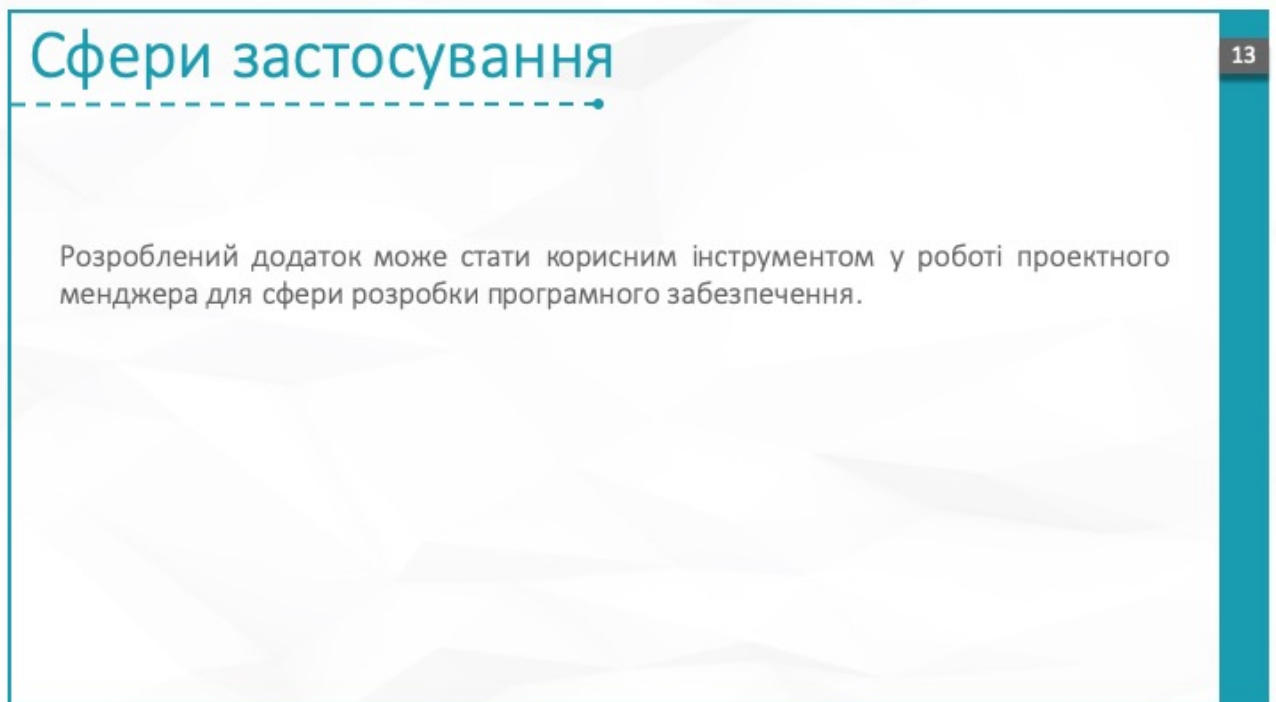


Рисунок А.13 – Сфери застосування

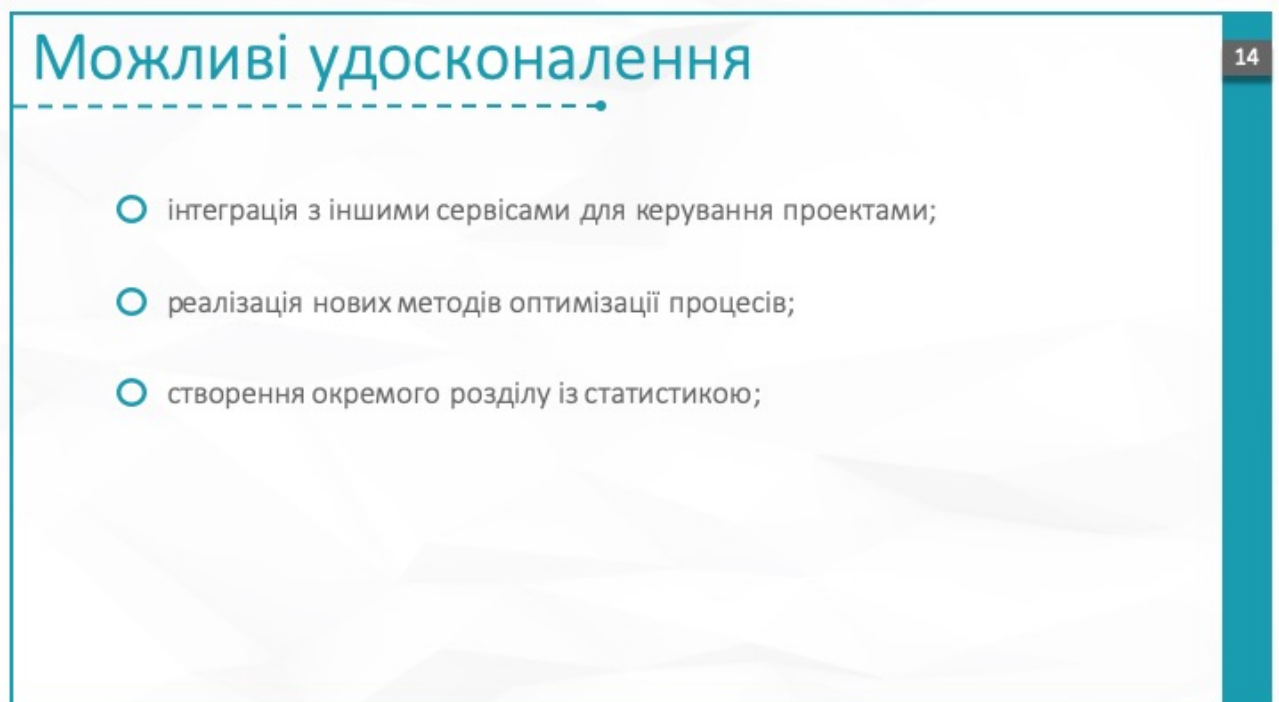


Рисунок А.14 – Можливості удосконалення

## Висновки

15

- проведено дослідження у сфері керування робочим процесом;
- проведено аналіз методів оптимізації керування робочим процесом;
- проведено концептуальне та UML-модельювання предметної галузі;
- реалізовано додаток основуючись на методах оптимізації управління часом.

Рисунок А.15 – Висновки

## ДОДАТОК Б

## Приклади програмного коду

```
const createList = (filter) => {  
  const ids = (state = [], action) => {  
    const handleToggle = (state, action) => {  
      const {result: toggledId, entities} = action;  
      const {completed} = entities[toggledId];  
      const shouldRemove = (  
        (completed && filter === 'active') ||  
        (!completed && filter === 'completed')  
      );  
  
      return shouldRemove  
        ? state.filter(id => id !== toggledId)  
        : state;  
    };  
  
    switch (action.type) {  
      case 'FETCH_TODO_SUCCESS':  
        return filter === action.filter  
          ? action.response.map(todo => todo.id)  
          : state;  
      case 'ADD_TODO_SUCCESS':  
        return filter !== 'completed'  
          ? [...state, action.response.id]  
          : state;  
      case 'TOGGLE_TODO_SUCCESS':  
        return handleToggle(state, action);  
      default:  
        return state;  
    }  
  };  
  
  const isFetching = (state = false, action) => {  
    if(action.filter !== filter) {  
      return state  
    }  
  
    switch (action.type) {  
      case 'FETCH_TODO_REQUEST':  
        return true;  
      case 'FETCH_TODO_SUCCESS':  
      case 'FETCH_TODO_FAILURE':  
        return false;  
      default:  
        return state;  
    }  
  };  
};
```

Рисунок Б.1 – Реалізація редьюсера для отримання списку задач

```

export const fetchTodos = filter => (dispatch, getState) => {
  if(isFetching(getState(), filter)) {
    return Promise.resolve();
  }

  dispatch({
    type: 'FETCH_TODOS_REQUEST',
    filter
  });

  return api.fetchTodos(filter)
    .then(
      todos => dispatch({
        type: 'FETCH_TODOS_SUCCESS',
        filter,
        todos
      }),
      error => ({
        type: 'FETCH_TODOS_FAILURE',
        filter,
        message: error.message || 'Something went wrong...'
      })
    );
};

export const addTodo = text => dispatch =>
  api.addTodo(text).then(response => ({
    type: 'ADD_TODO_SUCCESS',
    response
  }));

export const toggleTodo = id => dispatch =>
  api.toggleTodo(id).then(response => ({
    type: 'TOGGLE_TODO_SUCCESS',
    response
  }));

```

Рисунок Б.2 – Реалізація екшину для роботи з задачею

```

<nav class="sidebar-nav">
  <ul>
    <li class="nav-item active-nav">
      <a href="#about"> <i class="fa fa-user nav-item-icon"></i>
      <span class="nav-item-title">Account page</span> </a>
    </li>
    <li class="nav-item">
      <a href="#education">
        <i class="fa fa-calendar nav-item-icon"></i>
        <span class="nav-item-title">Task List</span> </a>
      </li>
    <li class="nav-item">
      <a href="#experience"> <i class="fa fa-heartbeat nav-item-icon"></i>
      <span class="nav-item-title">Biorhythms</span> </a>
    </li>
    <li class="nav-item">
      <a href="#skills"> <i class="fa fa-bar-chart nav-item-icon"></i>
      <span class="nav-item-title">Prioritize</span> </a>
    </li>
    <li class="nav-item">
      <a href="#portfolio"> <i class="fa fa-clock-o nav-item-icon"></i>
      <span class="nav-item-title">Track activity</span> </a>
    </li>
  </ul>
</nav>

```

Рисунок Б.3 – HTML для реалізація навігаційного меню

```
.main-info {
  padding: 40px;
  margin-left: 250px;
}

.btn-top {
  position: fixed;
  bottom: 0;
  right: 5px;
  width: 30px;
  height: 30px;
  border: 0;
  color: #fff;
  outline: none;
  border-radius: 5px 0 5px 0;
  cursor: pointer;
  background: $oxford-blue;
}

.section-title {
  margin-bottom: 50px;
  font: 25px/28px $font-stack;
  color: $shamrock;
  font-weight: 800;
}

.info-section {
  margin-bottom: 60px;
}

@media (max-width: 768px) {
  .sidebar {
    width: 75px;
  }

  .sidebar-avatar {
    height: 50px;
    width: 50px;
  }

  .nav-item-title,
  .back-btn-title,
  .sidebar-author-name {
```

Рисунок Б.4 – CSS стилі для реалізації інтерфейсу