

Інформаційна система управління проєктами

Виконав студент групи 2ІСТ-19м

Лишак Олександр Михайлович

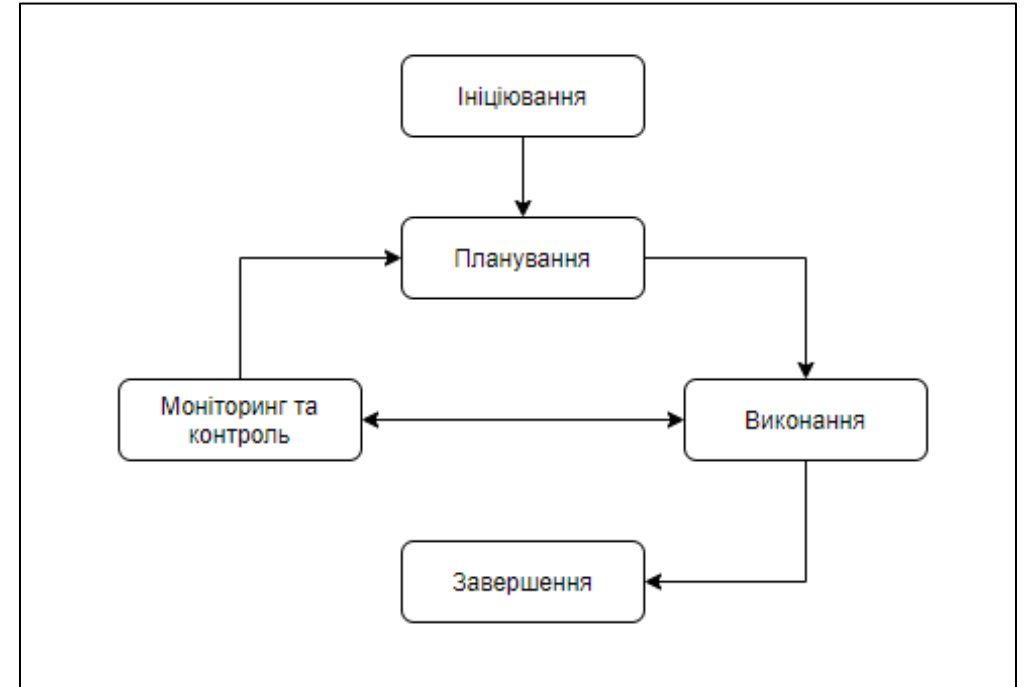
науковий керівник: д.т.н., проф. Мокін О. Б.

Вступ

У сучасному світі часто виникає питання поліпшення робочого процесу для покращення ефективності розподілу робочого часу. Інформаційна система управління проектами, це інструмент який допомагає спростити процес роботи.

Використовуючи такі сервіси користувачі будуть заощаджувати свій час, які вони могли потратити на непотрібні наради та обговорення. Менеджеру не потрібно особисто повідомляти працівників про нові задачі, про помилки в реалізації того чи іншого завдання і так далі. Також деякі платформи надають функціонал аналізу процесів проекту, що також заощаджує час керівника та спрощує створення звітів. Крім цього аналіз проекту може допомогти своєчасно побачити проблему в організації процесу, що зможе заощадити виділені ресурси.

Отже, розробка інформаційної системи управління проектами, яка буде мати простий та зрозумілий інтерфейс, а також буде надавати можливість перегляду в інформацію про процеси які виконуються в проекті в зручному вигляді, є актуальною.



Порядок етапів виконання проектів

Об'єктом дослідження є процес управління проєктами.

Предметом є інформаційна система управління проєктами.

Метою є підвищення ефективності роботи команди над проєктом за допомогою інформаційної системи управління проєктами, яка відрізняється від існуючих спрощеним інтерфейсом та розширеним модулем аналітики.

Для досягнення поставленої мети необхідно вирішити такі задачі:

- обґрунтування доцільності створення інформаційної системи управління проєктами;
- проєктування інформаційної системи управління проєктами;
- розробка інформаційної системи управління проєктами;

Огляд існуючих аналогів

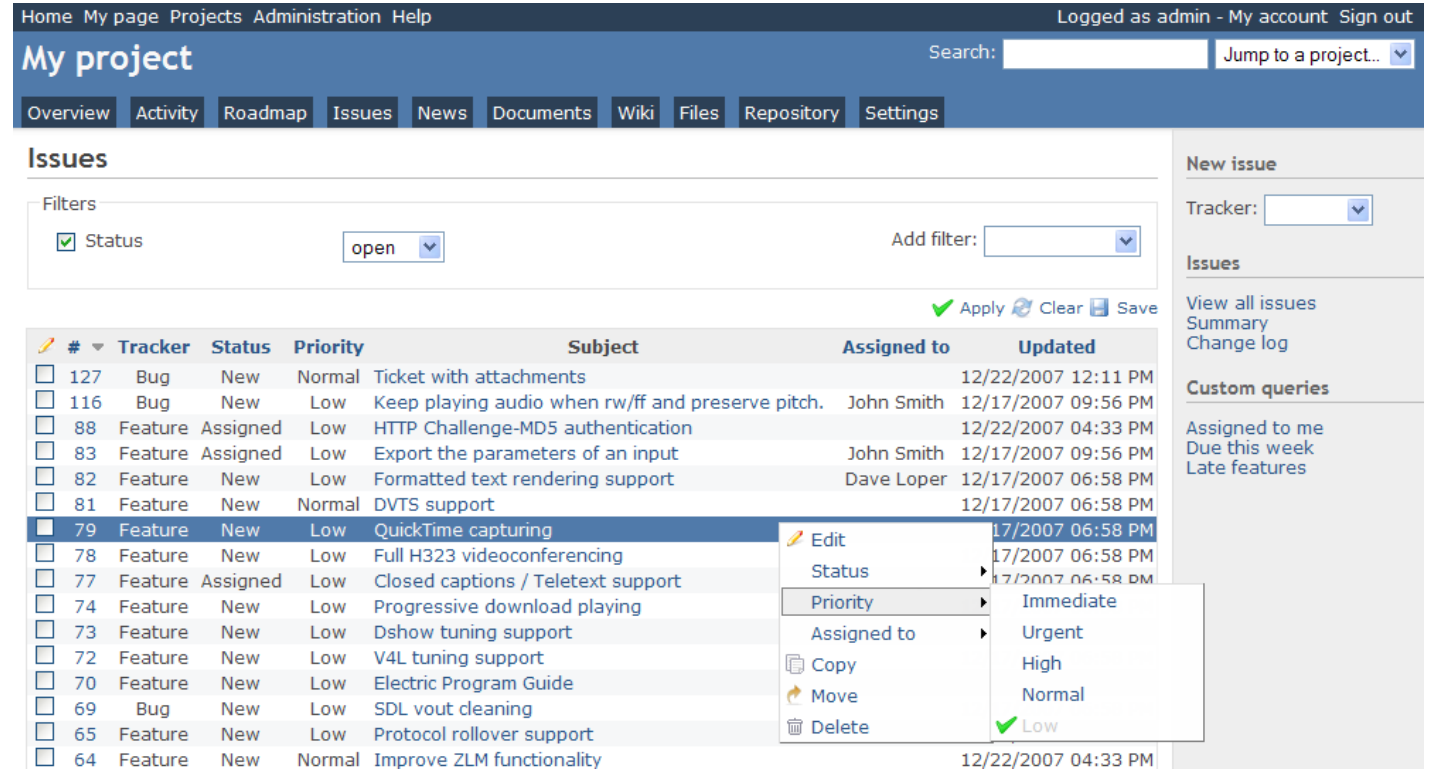
Redmine

Переваги:

- сервіс є безкоштовним;
- створення форумів і вікі-сторінок проєктів.

Недоліки:

- застарілий та не зручний інтерфейс;
- відсутня будь-яка статистика проєктів;
- нестабільність роботи.



Home My page Projects Administration Help Logged as admin - My account Sign out

My project Search: [] Jump to a project.. [v]

Overview Activity Roadmap Issues News Documents Wiki Files Repository Settings

Issues

Filters

Status open [v] Add filter: [] [v]

Apply Clear Save

#	Tracker	Status	Priority	Subject	Assigned to	Updated
<input type="checkbox"/> 127	Bug	New	Normal	Ticket with attachments		12/22/2007 12:11 PM
<input type="checkbox"/> 116	Bug	New	Low	Keep playing audio when rw/ff and preserve pitch.	John Smith	12/17/2007 09:56 PM
<input type="checkbox"/> 88	Feature	Assigned	Low	HTTP Challenge-MD5 authentication		12/22/2007 04:33 PM
<input type="checkbox"/> 83	Feature	Assigned	Low	Export the parameters of an input	John Smith	12/17/2007 09:56 PM
<input type="checkbox"/> 82	Feature	New	Low	Formatted text rendering support	Dave Loper	12/17/2007 06:58 PM
<input type="checkbox"/> 81	Feature	New	Normal	DVTS support		12/17/2007 06:58 PM
<input checked="" type="checkbox"/> 79	Feature	New	Low	QuickTime capturing		12/17/2007 06:58 PM
<input type="checkbox"/> 78	Feature	New	Low	Full H323 videoconferencing		12/17/2007 06:58 PM
<input type="checkbox"/> 77	Feature	Assigned	Low	Closed captions / Teletext support		12/17/2007 06:58 PM
<input type="checkbox"/> 74	Feature	New	Low	Progressive download playing		
<input type="checkbox"/> 73	Feature	New	Low	Dshow tuning support		
<input type="checkbox"/> 72	Feature	New	Low	V4L tuning support		
<input type="checkbox"/> 70	Feature	New	Low	Electric Program Guide		
<input type="checkbox"/> 69	Bug	New	Low	SDL vout cleaning		
<input type="checkbox"/> 65	Feature	New	Low	Protocol rollover support		
<input type="checkbox"/> 64	Feature	New	Normal	Improve ZLM functionality		12/22/2007 04:33 PM

Context menu for issue #79:

- Edit
- Status
- Priority
 - Immediate
 - Urgent
 - High
 - Normal
 - Low (checked)
- Assigned to
- Copy
- Move
- Delete

Огляд існуючих аналогів

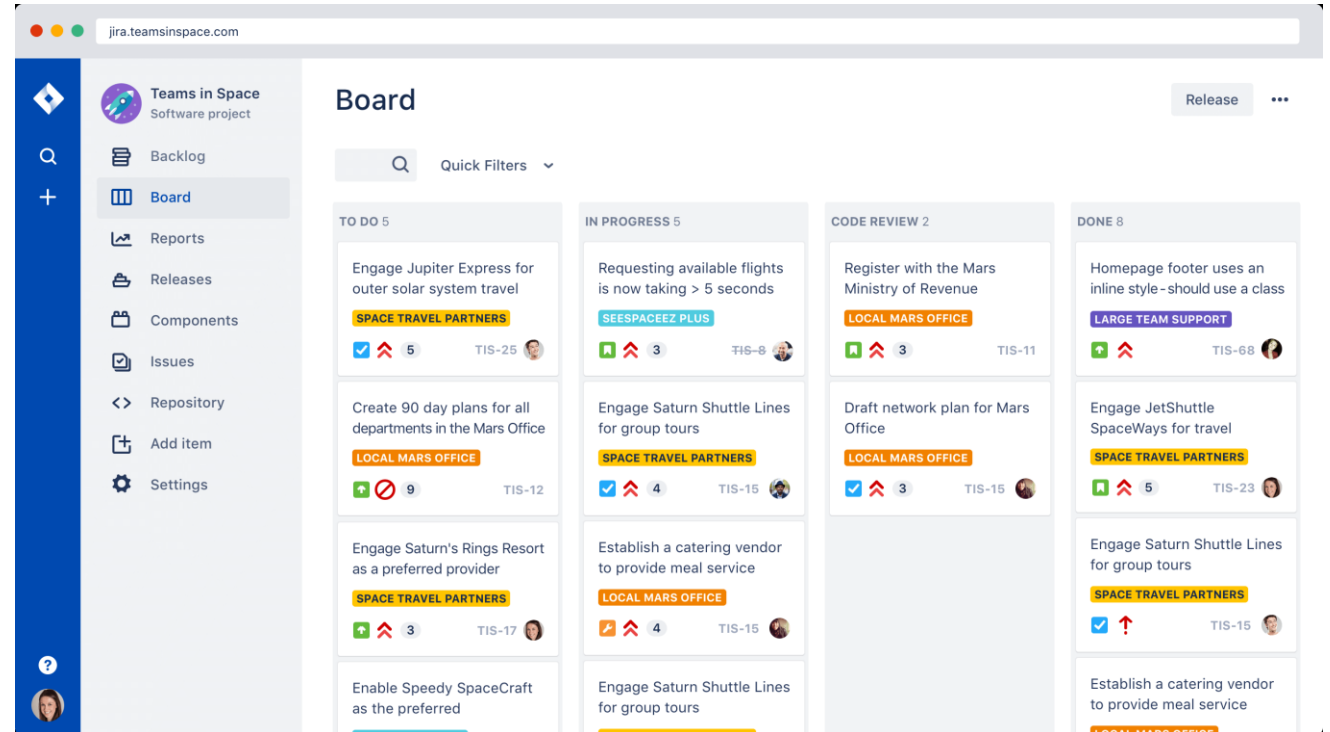
Jira

Переваги:

- змішані методологій і scrum;
- система оцінювання.

Недоліки:

- складність налаштування;
- велика кількість функціоналу доступна тільки при використанні сторонніх плагінів;
- відсутня українська локалізація.



Огляд існуючих аналогів

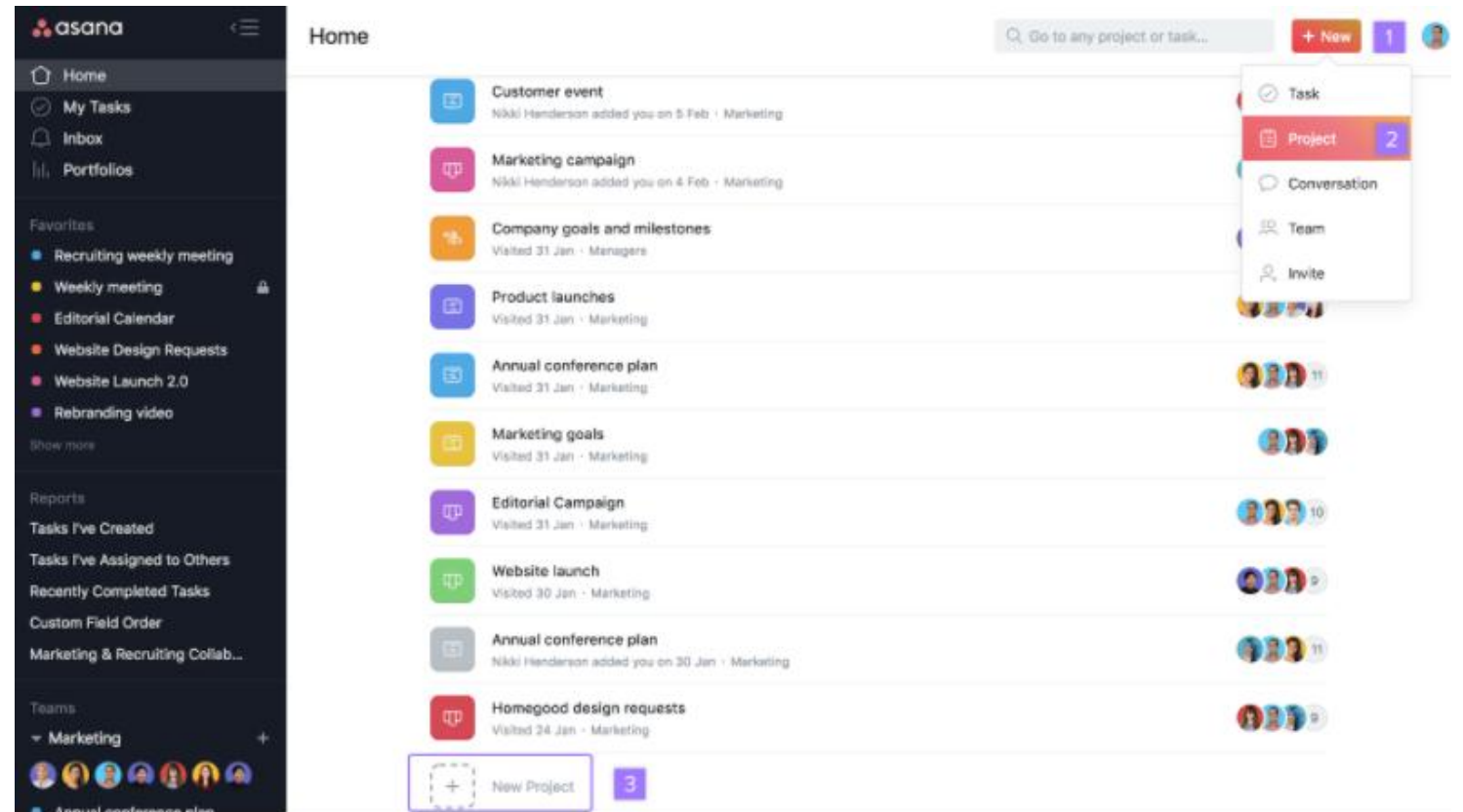
Asana

Переваги:

- повторювані завдання;
- відстеження термінів.

Недоліки:

- неможливість призначати завдання декільком користувачам відразу;
- відсутня українська локалізація.



Огляд існуючих аналогів

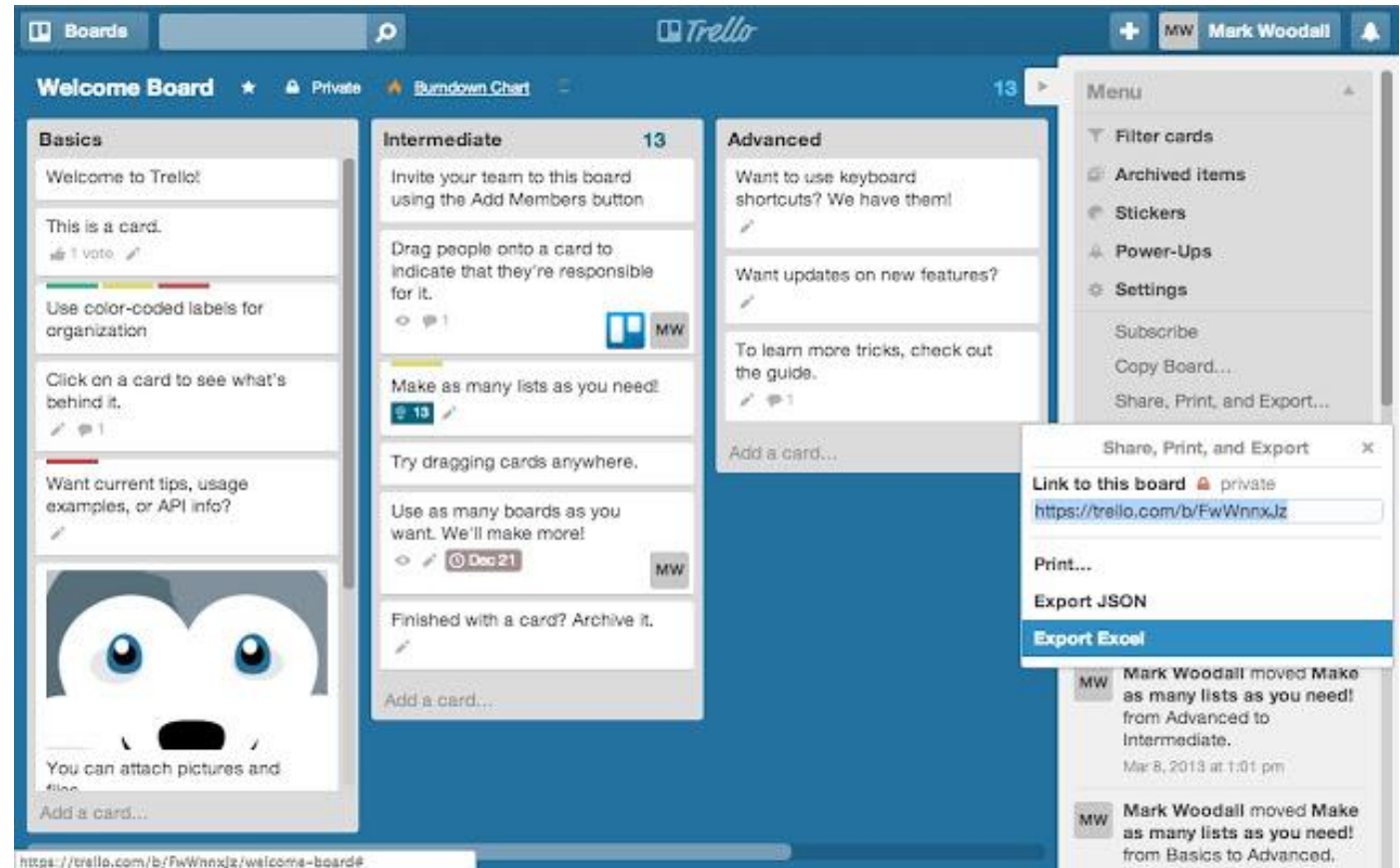
Trello

Переваги:

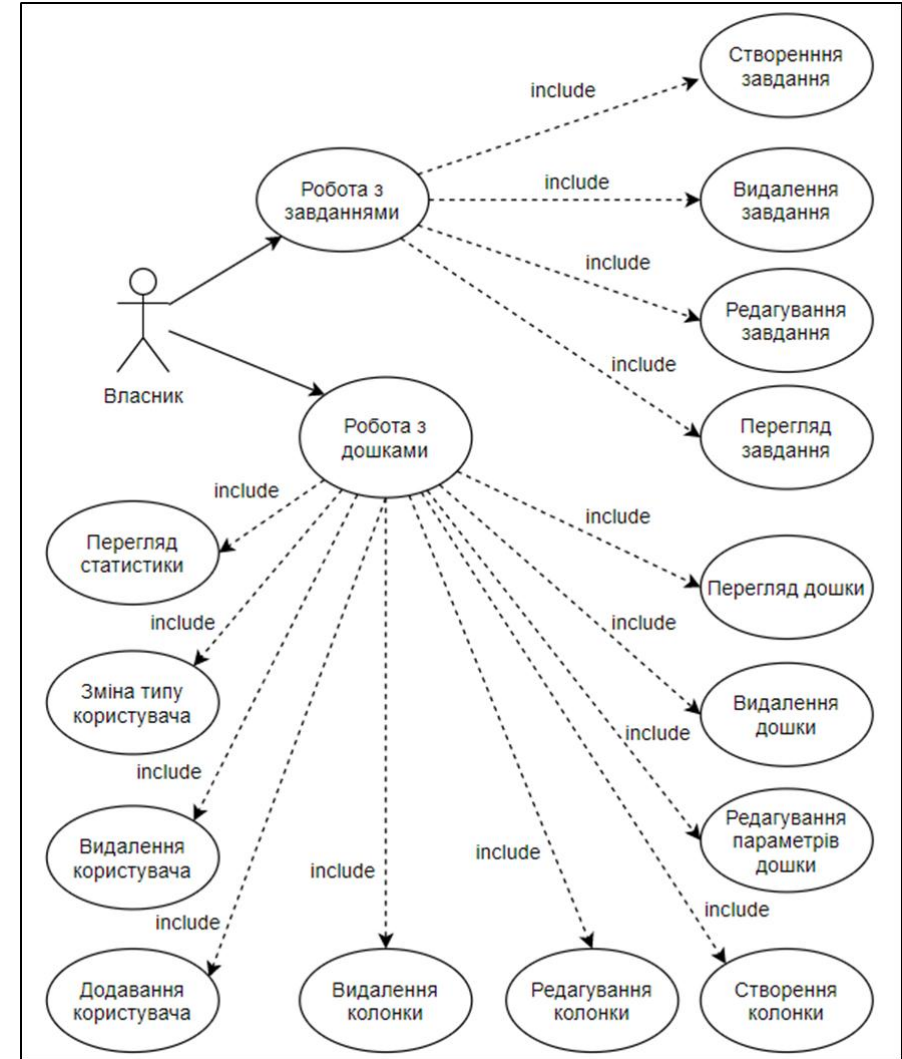
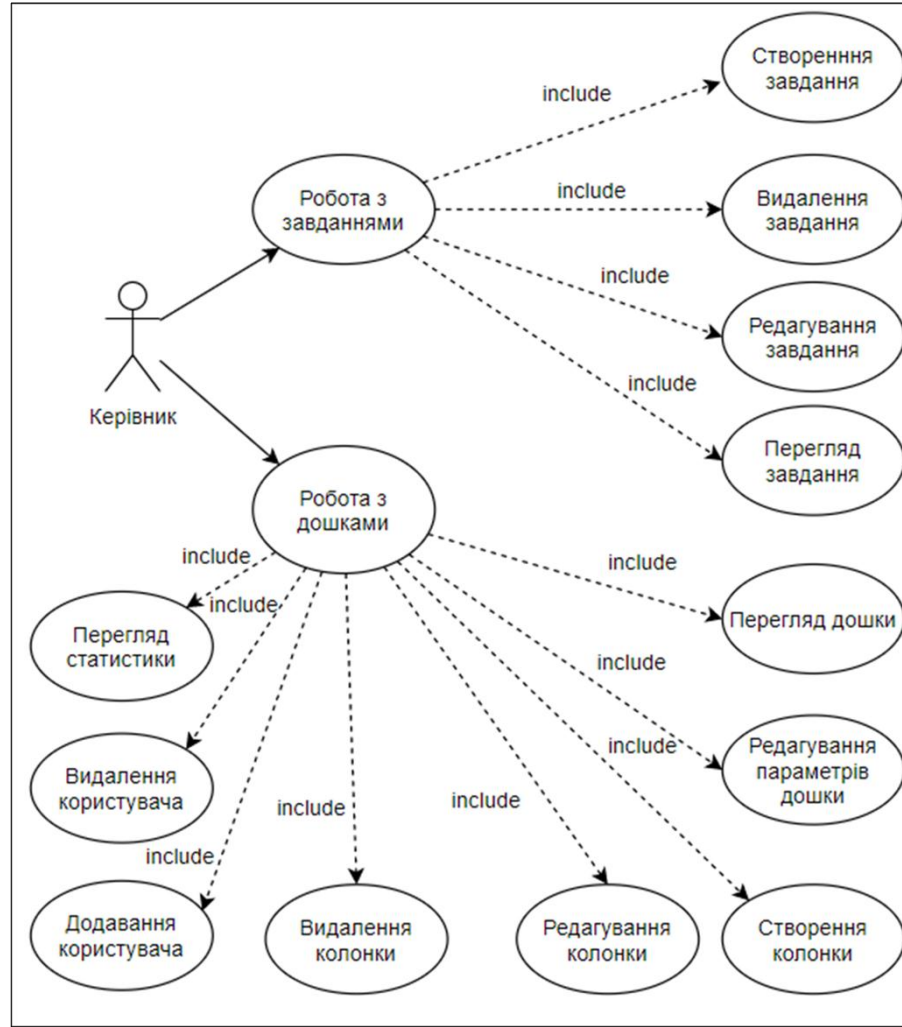
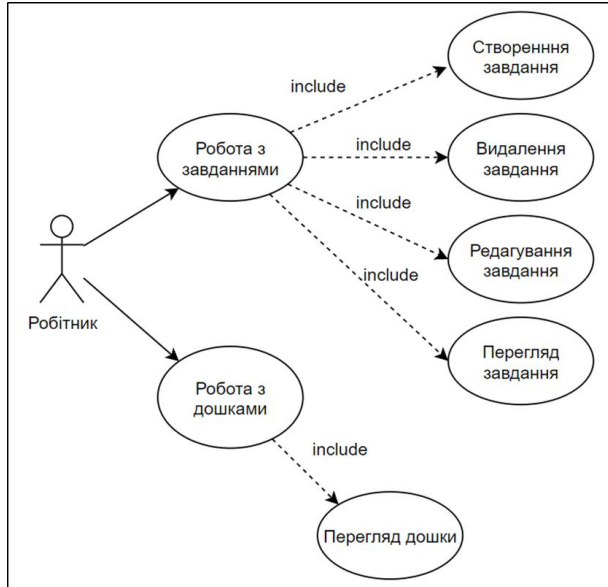
- наочність роботи та управління;
- створення власних приватних дощок.

Недоліки:

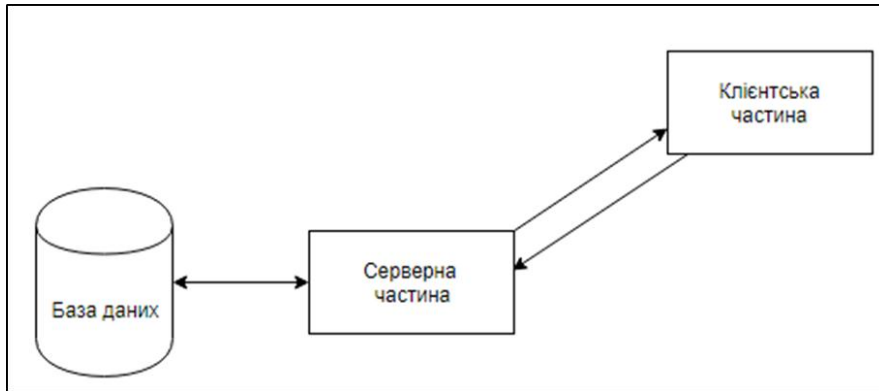
- відсутня можливість відстеження часу;
- відсутня будь-яка статистика проєктів;
- немає можливості додавати опис проєктів.



Формування процесів та методів роботи користувачів



Моделювання структури програмного забезпечення



Загальна структурна схема
інформаційної системи

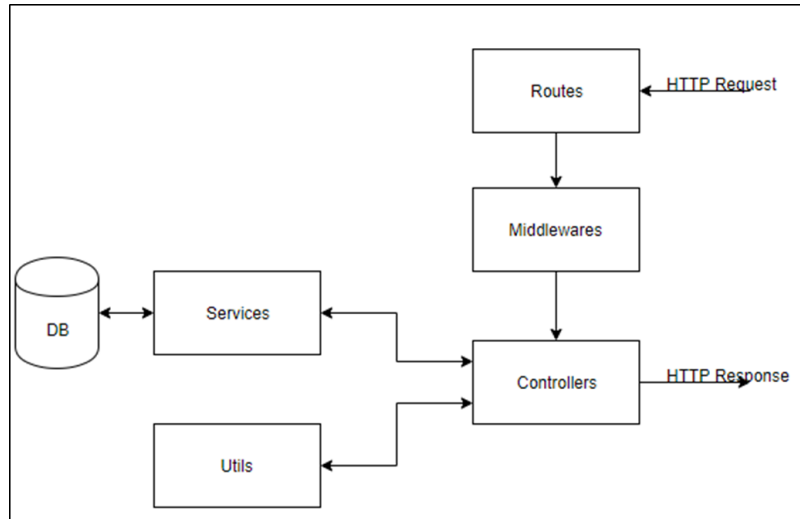
REST - це стиль архітектури програмного забезпечення для розподілених систем.

У загальному випадку REST - це дуже простий інтерфейс для управління інформацією без використання додаткових внутрішніх шарів. Кожна інформаційна одиниця однозначно ідентифікується глобальним ідентифікатором, таким як URL-адреса. Кожна URL-адреса, у свою чергу, має строго визначений формат.

Архітектурні обмеження:

- клієнт-серверна архітектура;
- відсутність стану;
- кешування;
- однорідний інтерфейс;
- шари абстракції.

Розробка серверної частини інформаційної системи



Структурна схема серверної частини інформаційної системи



Node.js це платформа з відкритим кодом для виконання JavaScript, яка побудована на основі движка JavaScript Chrome V8, яка дозволяє транслювати мову JavaScript в машинний код. Node.js перш за все призначений для створення серверних додатків на мові JavaScript. Хоча також існують проекти по написанню настільних додатків (Electron) і навіть зі створення коду для мікроконтролерів.

Express - це мінімалістичний і гнучкий веб-фреймворк для додатків Node.js, що надає великий набір функцій для мобільних і веб-додатків.

Розробка серверної частини інформаційної системи

Для роботи авторизації за основу було взято бібліотеки passport.js та bcrypt.js. За допомогою та bcrypt.js створено методи для хешування створення та перевірки токенів користувача. Для посилення безпеки даних користувачів буде використовуватись 2 види токенів:

- токен доступу;
- токен оновлення.



```
const cert = fs.readFileSync('private.pem');
const certRefresh = fs.readFileSync('private-refresh.pem');

module.exports.verify = function verify(token) {
  return new Promise((resolve, reject) => {
    jwt.verify(token, cert, (err, decoded) => {
      if (err) {
        reject(err);
      } else {
        resolve(decoded);
      }
    });
  });
};

module.exports.verifyRefresh = function verify(token) {
  return new Promise((resolve, reject) => {
    jwt.verify(token, certRefresh, (err, decoded) => {
      if (err) {
        reject(err);
      } else {
        resolve(decoded);
      }
    });
  });
};

module.exports.sign = function sign(object, options) {
  return jwt.sign(object, cert, options);
};

module.exports.signRefresh = function sign(object, options) {
  return jwt.sign(object, certRefresh, options);
};
```

Реалізація методів для створення та перевірки токенів

Розробка бази даних інформаційної системи управління проєктами

Sequelize - це ORM-бібліотека для додатків на Node.js, яка здійснює зіставлення таблиць в базі даних з відповідними їм класами. При використанні Sequelize можна не писати SQL-запити, а працювати з даними як зі звичайними об'єктами. Причому Sequelize може працювати з рядом СУБД - MySQL, Postgres, MariaDB, SQLite, MS SQL Server.



Sequelize

```
const fs = require('fs');
const path = require('path');
const Sequelize = require('sequelize');

const env = process.env.NODE_ENV || 'dev';
const config = require('.../config/' + env + '.js').db;

const basename = path.basename(__filename);
const db = {};

let sequelize;

sequelize = new Sequelize(config.database, config.username, config.password, config);

fs
  .readdirSync(__dirname + '/models')
  .filter((file) => (file.indexOf('.') !== 0) && (file !== basename) && (file.slice(-3) === '.js'))
  .forEach((file) => {
    const model = sequelize.import(path.join(__dirname + '/models', file));
    db[model.name] = model;
  });

Object.keys(db).forEach((modelName) => {
  if (db[modelName].associate) {
    db[modelName].associate(db);
  }
});

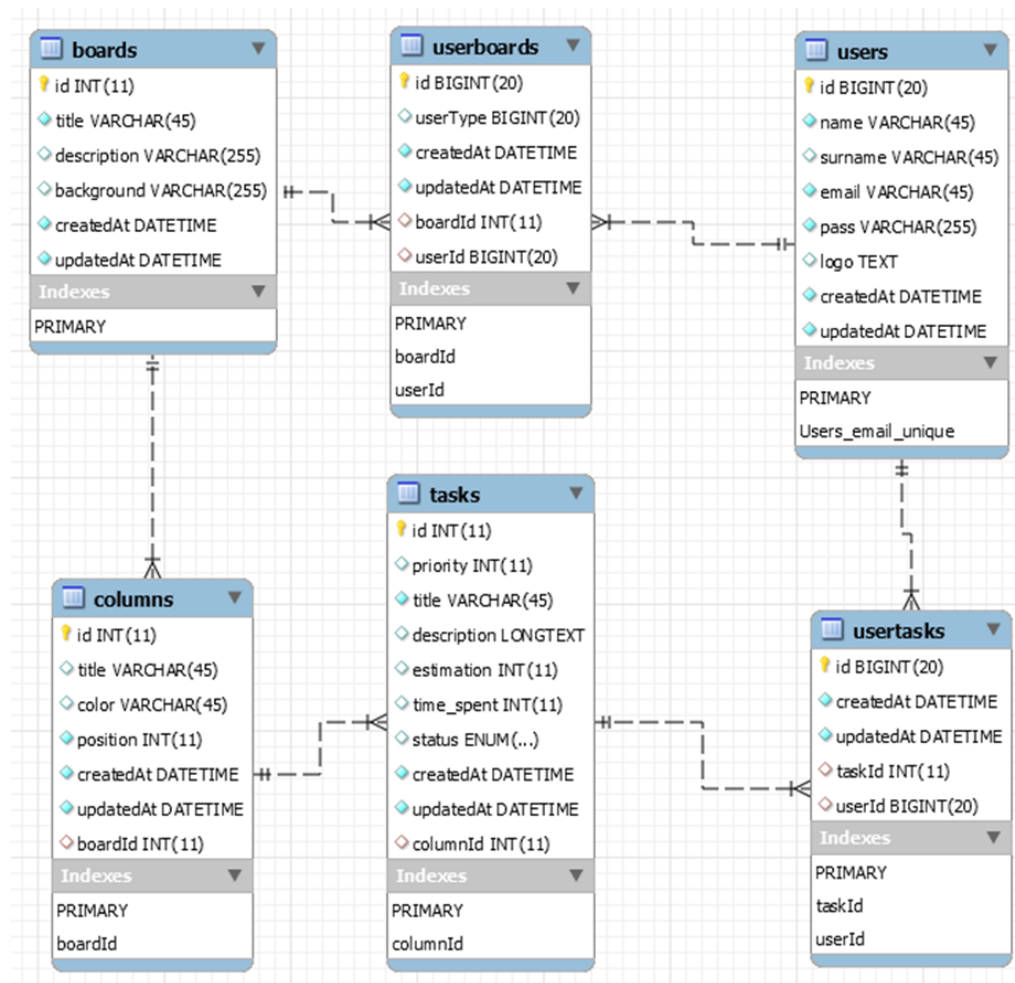
db.sequelize = sequelize;
db.Sequelize = Sequelize;

module.exports = db;
```

Модуль підключення бази даних

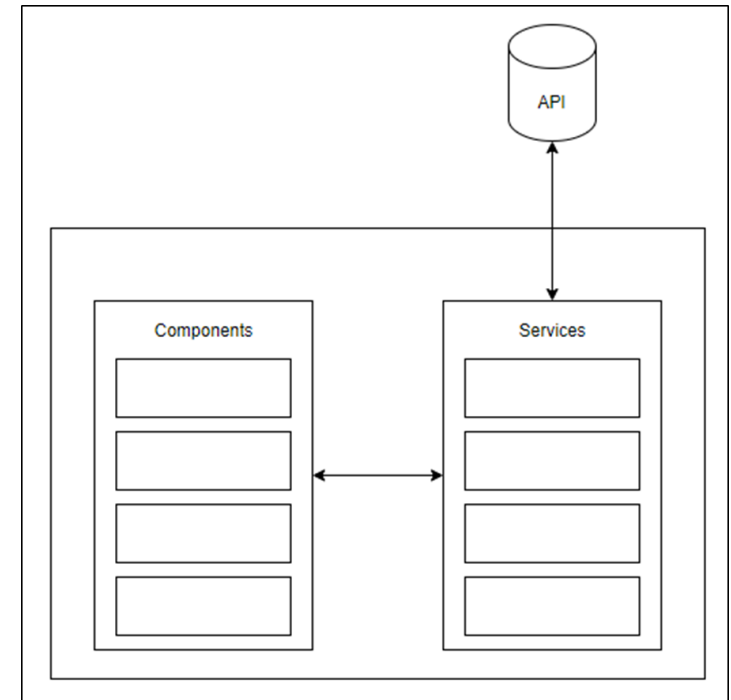
Розробка бази даних інформаційної системи управління проєктами

Ім'я сутності 1	Ім'я сутності 2	Тип зв'язку	Ім'я зв'язку	Клас належності
Користувач	Дошка	Б:Б	Має	Обов'язковий
Дошка	Колонка	Б:1	Містить	Обов'язковий
Колонка	Завдання	Б:1	Містить	Обов'язковий
Завдання	Користувач	Б:Б	Має	Можливий



Розробка клієнтської частини інформаційної системи

Angular — написаний на TypeScript front-end фреймворк з відкритим кодом, який розробляється під керівництвом Angular Team у компанії Google, а також спільнотою приватних розробників та корпорацій. Angular — це AngularJS, який переосмислили та який був повністю переписаний тією ж командою розробників.



Структурна схема клієнтської частини інформаційної системи

Розробка клієнтської частини інформаційної системи

WorkManager © 2020 Українська

Дошки

- Адміністраторські дошки
- Менеджерські дошки
- Користувачькі дошки

СТВОРИТИ НОВУ ДОШКУ

Назва
Розробка інтелектуальної системи

Опис
Дошка для розробки системи

СТВОРИТИ

WorkManager © 2020 Українська

Дошки

ДОДАТИ ЗАВДАННЯ

Заголовок
Оновити шапку сайту

Опис

B I N |

```
1 Потрібно оновити такі елементи в шапці сайту
2 - заголовок
3 - посилання
4
5 ![Приклад дизайну](http://site.ua/1.jpg)
6 [Посилання на сайт](http://site.ua/)
7
8
```

Колонка
Завдання

Пріоритет
1

Оцінка часу
12

Витрачений час
0

Користувачі

+ Додати користувача

СТВОРИТИ

WorkManager © 2020 Українська

Дошки

Розробка інтелектуальної системи

Дошка для розробки системи

- Завдання
- В розробці
- Тестування

Оновити шапку сайту
Пріоритет: 1

+ Додати завдання

+ Додати завдання

+ Додати завдання

Вкладки сторінки редагування дошки

The screenshot shows the 'Користувачі' (Users) tab. The main content area contains the following fields:

- Заголовок:** Розробка інтелектуальної системи
- Опис:** Дошка для розробки системи
- Фон проекту:** (empty)

At the bottom of the content area, there are two large buttons: a blue one labeled 'ЗБЕРЕГТИ' (Save) and a red one labeled 'ВИДАЛИТИ' (Delete).

The footer contains 'WorkManager © 2020' and a Ukrainian flag icon with the text 'Українська'.

The screenshot shows the 'Користувачі' (Users) tab. It features a user profile card for 'Геральт з Рівії' with the email 'test@gmail.com' and role 'Адміністратор'. Below the card is a green button labeled '+ Додати користувача' (Add user).

The footer contains 'WorkManager © 2020' and a Ukrainian flag icon with the text 'Українська'.

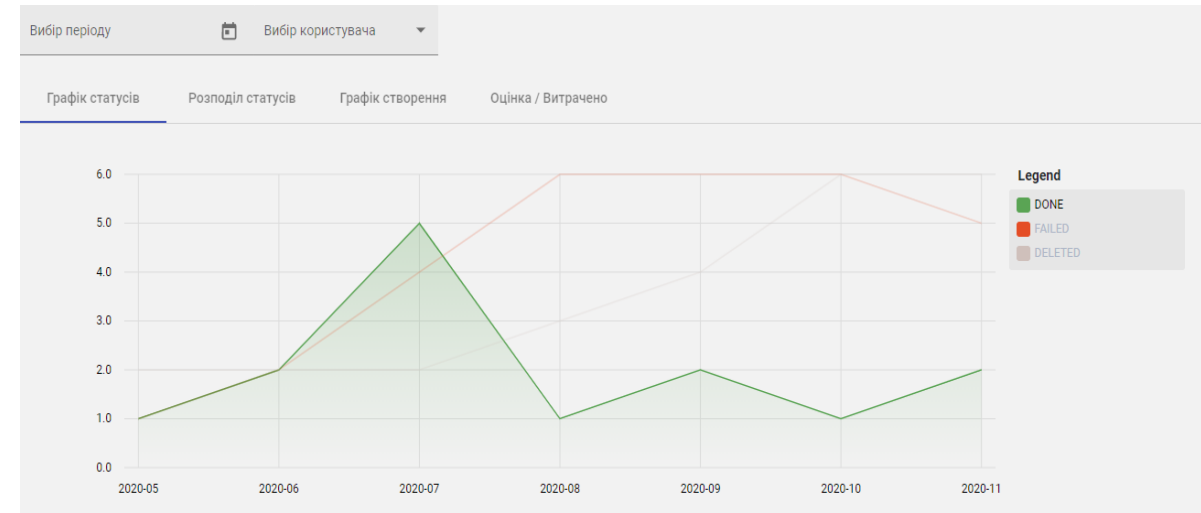
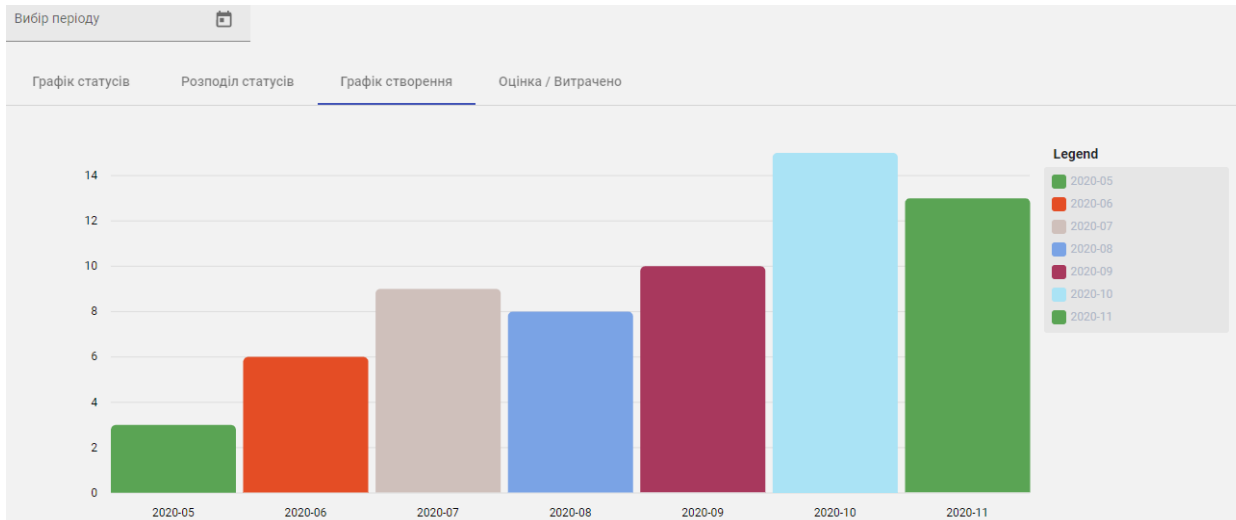
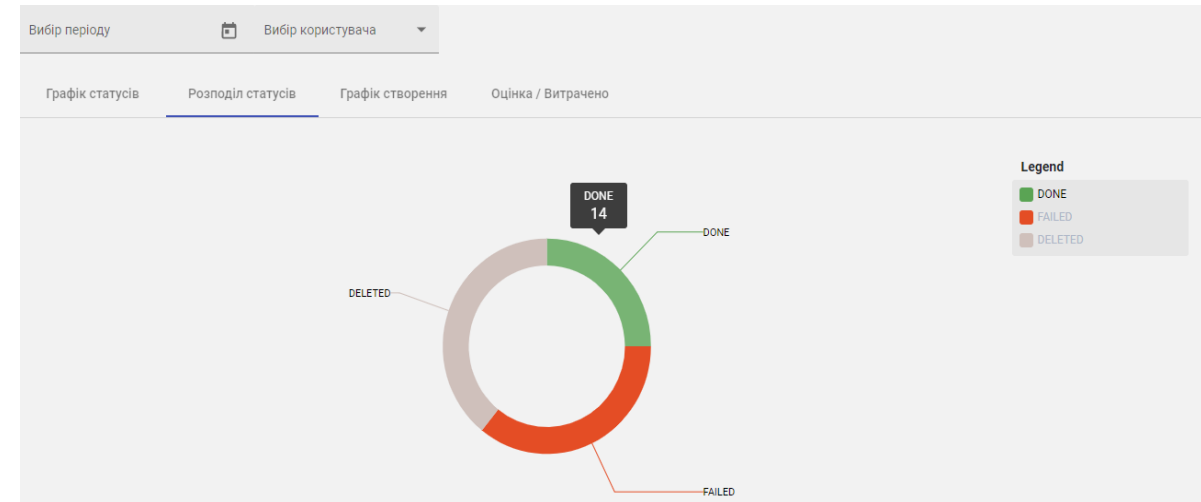
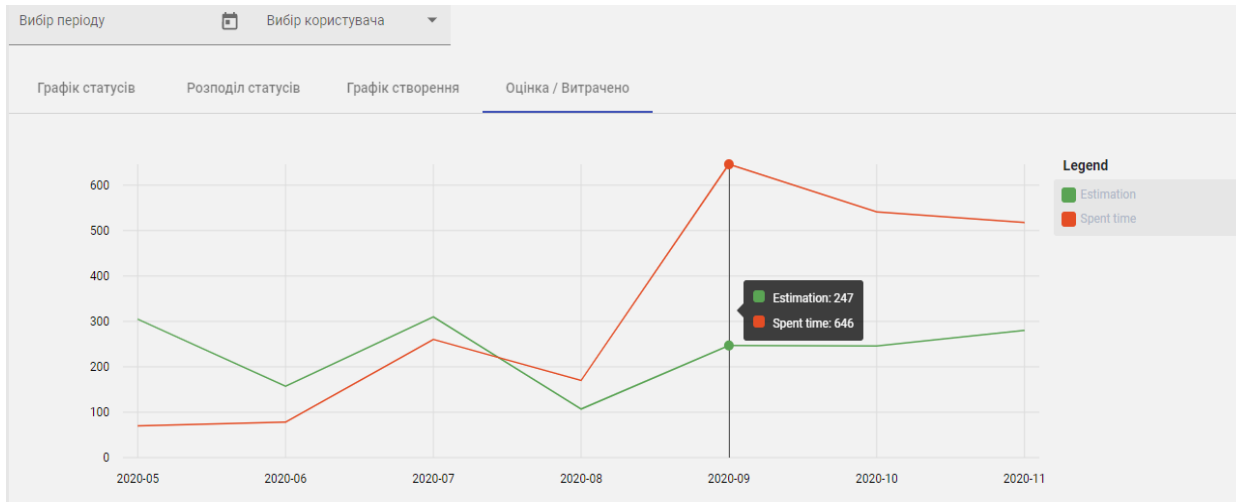
The screenshot shows the 'Колонки' (Columns) tab. It displays a list of four columns, each with a colored header and edit/delete icons:

- Завдання (red header)
- В розробці (blue header)
- Тестування (purple header)
- Виконано (yellow header)

At the bottom of the list is a green button labeled '+ Додати колонку' (Add column).

The footer contains 'WorkManager © 2020' and a Ukrainian flag icon with the text 'Українська'.

Вкладки сторінки статистики



Наукова новизна

Основні результати, які були отримані в процесі вирішення поставлених завдань та становлять наукову новизну дослідження, полягають у вдосконаленні інформаційної система управління проєктами, яка відрізняється від існуючих спрощеним інтерфейсом та розширеним модулем аналітики, що дозволяє підвищити ефективність роботи команди над проєктом.

Практичне значення

Практичне значення одержаних результатів полягає у більш ефективному використанні часу та кращій координації роботи команди за рахунок впровадження таких особливостей інформаційної системою управління проєктом:

- реалізовано більш простий та зручний інтерфейс;
- реалізовано модуль розширеної аналітики;
- побудовано UML-діаграму класів, яка дозволяє швидко відслідковувати взаємодію елементів проєкту.

За результатами магістерської кваліфікаційної роботи опубліковано: 1 тези на XV міжнародній конференції "Контроль і управління в складних системах" (КУСС-2020).

Висновки

В ході виконання магістерської кваліфікаційної роботи було розроблено механізми для роботи власника, керівника та робітника в інформаційній системі управління проєктами.

В результаті виконання магістерської кваліфікаційної роботи було розроблено та реалізовано:

- базу даних для оптимального представлення усієї інформації предметної області;
- прикладний програмний інтерфейс, що має можливість швидко читати/записувати інформацію бази даних та легко обмінюватися цими даними з клієнтською частиною;
- клієнтську частину, що має зручний та зрозумілий інтерфейс, який може забезпечувати в режимі реального часу та швидку взаємодію між користувачами, системою та її базою даних

За результатами магістерської кваліфікаційної роботи опубліковано тези доповідей на науковій конференції.

ДЯКУЮ ЗА УВАГУ!!!