

# РОЗДІЛ 1

## ЗАПИТИ ЗАЦІКАВЛЕНИХ ОСІБ

### Введення

У цьому документі описані запити зацікавлених осіб по відношенню до розроблюваної системи «Організація вуличних змагань WorkOut», в якості яких виступають: організатори цих змагань, судді, учасники та гості(глядачі).

#### 1.1. Мета

Метою документа є визначення основних вимог щодо організації даних спортивних змагань, а саме таких аспектів, як забезпечення інформацією усіх зацікавлених осіб, їх взаємодії за посередництвом системи.

#### 1.2. Контекст

Перелік вимог, перерахованих у цьому документі, є основою технічного завдання для розробки системи для супроводу вуличних спортивних змагань, виставлення балів (оцінок) суддями, забезпечення інформацією учасників та гостей.

#### 1.3. Короткий огляд продукту

Систему «Організація вуличних змагань WorkOut» можна умовно розділити на дві частини програмного забезпечення:

Сайт, який використовують зацікавлені особи в якості інтерфейсу доступу до усіх даних, що пов'язані із змаганнями, інструменту для роботи з ними;

База даних на головному сервері, що відображатиметься на сайті у вигляді турнірної таблиці.

#### 1.4. Ділові правила і приписи

##### 1.4.1. Призначення системи організації вуличних змагань

Система призначена для забезпечення взаємодії між базою даних та сайтом змагань, а також встановлення формальних правил доступу до даних та можливостей роботи з ними для потенційних користувачів системою.

Така організація зберігання даних про учасників, місце, час змагань та іншої інформації виключає можливість втрати інформації або порушення її конфіденційності.

|               |  |  |  |  |  |              |             |          |        |     |                                  |      |   |
|---------------|--|--|--|--|--|--------------|-------------|----------|--------|-----|----------------------------------|------|---|
| Підп. і дата  |  |  |  |  |  | Підп. і дата | Інв. № підп |          |        |     |                                  |      |   |
|               |  |  |  |  |  |              |             |          |        |     |                                  |      |   |
|               |  |  |  |  |  |              |             |          |        |     |                                  |      |   |
|               |  |  |  |  |  |              |             |          |        |     |                                  |      |   |
| Взаєм. інв. № |  |  |  |  |  | Зм           | Арк.        | № докум. | Підпис | Дат | 6.050102 «Комп'ютерна інженерія» | Арк. | 3 |
|               |  |  |  |  |  |              |             |          |        |     |                                  |      |   |



4. Організатор отримує інформацію.
5. Організатор надсилає на адресу електронної пошти гостя запит про підтвердження реєстрації.
6. Гість підтверджує факт реєстрації у змаганнях.
7. Організатор відкриває базу даних системи та вносить відповідні дані про учасника.
8. Організатор створює для учасника особистий електронний кабінет.
9. Учасник отримує персональний ідентифікаційний номер та пароль для подальшого доступу до особистого кабінету.
10. Профіль учасника автоматично активується.
11. По завершенню роботи організатор закриває базу даних.
12. Організатор перевіряє наявність змін у турнірній таблиці на сайті.

#### **Виключні ситуації:**

1. Гість невірно заповнив поля анкети або залишив їх порожніми.
2. Гість не підтвердив факт реєстрації у змаганнях.
3. Порушено зв'язок між сервером із базою даних та сайтом.

#### **1.5.2. Сценарій доступу до особистого кабінету**

**Назва:** сценарій доступу до особистого кабінету.

**Учасники:** учасник.

**Передумови:** учасник зареєстрований у змаганнях.

**Результат:** перевірено поточні дані учасника.

#### **Сценарій:**

1. Учасник вводить ідентифікаційний номер та пароль.
2. Учасник отримує доступ до особистого кабінету.
3. Автоматично створюється сеанс роботи.

|               |      |          |        |     |                                  |
|---------------|------|----------|--------|-----|----------------------------------|
| Підп. і дата  |      |          |        |     |                                  |
|               |      |          |        |     |                                  |
| Взаєм. інв. № |      |          |        |     |                                  |
|               |      |          |        |     |                                  |
| Інв. № дубл.  |      |          |        |     |                                  |
|               |      |          |        |     |                                  |
| Підп. і дата  |      |          |        |     |                                  |
|               |      |          |        |     |                                  |
| Інв. № підп   |      |          |        |     |                                  |
|               |      |          |        |     |                                  |
| Зм            | Арк. | № докум. | Підпис | Дат | 6.050102 «Комп'ютерна інженерія» |
|               |      |          |        |     | Арк.                             |
|               |      |          |        |     | 5                                |

4. Учасник переглядає інформацію щодо своїх успіхів, а також місця і часу проведення наступного етапу змагань, у якому він бере участь.

5. Учасник виходить із особистого кабінету.

6. Сеанс роботи автоматично завершується.

Виключні ситуації:

Учасник невірно ввів ідентифікаційний номер та пароль або залишив поля порожніми.

### **1.5.3. Сценарій відмови від участі у змаганнях.**

**Назва:** сценарій відмови від участі у змаганнях.

**Учасники:** учасник, організатор.

**Передумови:** учасник зареєстрований у змаганнях.

**Результат:** учасник усунений від участі.

**Сценарій:**

1. Учасник заходить до особистого кабінету.

2. Учасник надсилає організатору повідомлення про бажання відмовитись від участі у змаганнях.

3. Учасник виходить із особистого кабінету.

4. Організатор отримує повідомлення.

5. Організатор виключає запис учасника із бази даних.

6. Організатор видаляє особистий кабінет учасника.

7. Організатор перевіряє наявність змін у турнірній таблиці на сайті.

**Виключні ситуації:**

1. Порушено зв'язок між сервером із базою даних та сайтом.

2. Учасник не вийшов з особистого кабінету.

### **1.5.4. Сценарій адміністрування ресурсів організаторами**

**Назва:** сценарій адміністрування ресурсів організаторами.

**Учасники:** організатор.

|              |      |               |        |              |                                  |              |  |             |  |  |  |  |  |  |      |
|--------------|------|---------------|--------|--------------|----------------------------------|--------------|--|-------------|--|--|--|--|--|--|------|
| Підп. і дата |      | Взаєм. інв. № |        | Інв. № дубл. |                                  | Підп. і дата |  | Інв. № підп |  |  |  |  |  |  | Арк. |
|              |      |               |        |              |                                  |              |  |             |  |  |  |  |  |  | 6    |
| Зм           | Арк. | № докум.      | Підпис | Дат          | 6.050102 «Комп'ютерна інженерія» |              |  |             |  |  |  |  |  |  |      |

**Передумови:** організатор має доступ до адміністрування.

**Результат:** внесено зміни до даних на сайті.

**Сценарій:**

1. Організатор вводить логін та пароль для доступу до бази даних.
2. Організатор вносить зміни до бази даних відповідно до повідомлень з боку учасників та судей.
3. Організатор завершує роботу та залишає сервер з базою даних.
4. Організатор вводить шестизначний логін та персональний пароль для доступу до адміністрування на сайті.
5. Автоматично створюється сеанс роботи.
6. Організатор працює з ресурсами сайту відповідно до своїх прав доступу.
7. Організатор закінчує роботу.
8. Сеанс роботи автоматично завершується.

**Виключні ситуації:**

1. Організатор невірно ввів логін та пароль або залишив поля порожніми.
2. Порушено зв'язок між сервером із базою даних та сайтом.

#### **1.5.5. Сценарій адміністрування ресурсів суддями**

**Назва:** сценарій адміністрування ресурсів суддями.

**Учасники:** суддя.

**Передумови:** суддя має частковий доступ до адміністрування.

**Результат:** внесено зміни до турнірної таблиці.

**Сценарій:**

1. Суддя вводить шестизначний логін та персональний пароль для отримання часткового доступу до адміністрування на сайті.
2. Автоматично створюється сеанс роботи.

|              |  |               |  |              |  |              |  |             |  |      |          |        |     |                                  |      |   |
|--------------|--|---------------|--|--------------|--|--------------|--|-------------|--|------|----------|--------|-----|----------------------------------|------|---|
| Підп. і дата |  | Взаєм. інв. № |  | Інв. № дубл. |  | Підп. і дата |  | Інв. № підп |  |      |          |        |     |                                  |      |   |
|              |  |               |  |              |  |              |  |             |  |      |          |        |     |                                  |      |   |
| Зм           |  |               |  |              |  |              |  |             |  | Арк. | № докум. | Підпис | Дат | 6.050102 «Комп'ютерна інженерія» | Арк. | 7 |

3. Суддя працює з оцінками учасників змагань.
4. Суддя надсилає організатору запит на внесення змін до турнірної таблиці.
5. Суддя закінчує роботу.
6. Сеанс роботи автоматично завершується.

**Виключні ситуації:**

1. Суддя невірно ввів логін та пароль або залишив поля порожніми.
2. порушено зв'язок між сайтом та сервером із базою даних.

**1.5.6. Сценарій переходу з категорії «учасник» до категорії «організатор».**

**Назва:** сценарій переходу з категорії «учасник» до категорії «організатор».

**Учасники:** учасник, організатор.

**Передумови:** учасник бажає організувати власні змагання.

**Результат:** зареєстровано нового організатора.

**Сценарій:**

1. Учасник заходить до особистого кабінету.
2. Учасник надсилає організаторам повідомлення, що він бажає стати організатором власних змагань.
3. Учасник виходить з особистого кабінету.
4. Організатор проводить співбесіду з іншими організаторами.
5. Організатор надсилає учасникові підтвердження, персональні логін та пароль для доступу до адміністрування на сайті та дані для доступу до бази даних.
6. Кожному учасникові, який перейшов у категорію організаторів, на перший час надається персональний консультант.

**Виключні ситуації:**

|              |               |              |        |     |                                  |
|--------------|---------------|--------------|--------|-----|----------------------------------|
| Підп. і дата |               |              |        |     |                                  |
|              | Взаєм. інв. № |              |        |     |                                  |
|              |               | Інв. № дубл. |        |     |                                  |
|              |               |              |        |     |                                  |
| Підп. і дата |               |              |        |     |                                  |
|              |               |              |        |     |                                  |
| Інв. № підп  |               |              |        |     |                                  |
|              |               |              |        |     |                                  |
| Зм           | Арк.          | № докум.     | Підпис | Дат | 6.050102 «Комп'ютерна інженерія» |
|              |               |              |        |     | Арк.                             |
|              |               |              |        |     | 8                                |



### 1.6.1. Структура сайту системи «Організація вуличних змаганьWorkOut»

Сайт системи повинен мати наступні розділи:

1. *Реєстрація* – розділ, в якому будь-який бажаючий зможе зареєструватися у змаганнях.
2. *Вхід до особистого кабінету* – дає можливість переглянути персональну сторінку учасника.
3. *Турнірна таблиця* – інформація про оцінки кожного учасника та відомість про поточне місце учасника в турнірі.
4. *Керування та оцінювання* – розділ виключно для організаторів та суддів, в ньому можна зареєструвати учасника, внести зміни до турнірної таблиці, опублікувати оголошення щодо проведення змагань.
5. *Гостям турніру* – розділ, в якому гість може забронювати сидяче місце на змаганні.
6. *Оголошення* – інформація про адресу та час проведення змагань.
7. *Новини* – інформація про поточний хід подій на змаганнях.
8. *Контакти* – телефони для довідок організаторів змагання, адреса за яким можна звертатися.
9. *Архіви змагань* – інформація про турніри, які вже відбулися.

### 1.6.2. Можливості зацікавлених осіб

#### А)Можливості учасника

Реєструватися для участі у змаганнях

Мати доступ до свого особистого кабінету

Надіслати організаторам запит щодо організації власних змагань

Надіслати організаторам запит щодо переходу у категорію судей

Переглядати новини

Переглядати оголошення

Коментувати новини

Переглядати турнірну таблицю

Переглядати архів попередніх змагань

|               |      |          |        |     |      |
|---------------|------|----------|--------|-----|------|
| Підп. і дата  |      |          |        |     |      |
|               |      |          |        |     |      |
| Взаєм. інв. № |      |          |        |     |      |
|               |      |          |        |     |      |
| Інв. № дубл.  |      |          |        |     |      |
|               |      |          |        |     |      |
| Підп. і дата  |      |          |        |     |      |
|               |      |          |        |     |      |
| Інв. № підп   |      |          |        |     |      |
|               |      |          |        |     |      |
|               |      |          |        |     | Арк. |
|               |      |          |        |     | 10   |
| Зм            | Арк. | № докум. | Підпис | Дат |      |



Переглядати контактну інформацію.

### **Б)Можливості гостя (глядача)**

Переглядати новини

Переглядати оголошення

Переглядати архів попередніх змагань

Переглядати контактну інформацію

Реєструватися для участі у змаганнях

Переглядати турнірну таблицю.

### **В)Можливості організаторів**

Мати персональний доступ до керування сайтом

Реєструвати учасників

Публікувати новини

Публікувати оголошення

Переглядати новини

Переглядати оголошення

Вносити зміни до турнірної таблиці

Переглядати турнірну таблицю

Завантажувати архів змагань

Переглядати архів змагань

Публікувати контактну інформацію

Переглядати контактну інформацію.

### **Г)Можливості суддів**

Мати персональний доступ до редагування турнірної таблиці

Переглядати новини

Переглядати оголошення

Вносити зміни до турнірної таблиці

|               |  |
|---------------|--|
| Підп. і дата  |  |
| Взаєм. інв. № |  |
| Інв. № дубл.  |  |
| Підп. і дата  |  |
| Інв. № підп   |  |

|    |      |          |        |     |
|----|------|----------|--------|-----|
|    |      |          |        |     |
| Зм | Арк. | № докум. | Підпис | Дат |

Переглядати турнірну таблицю

Переглядати архів змагань

Переглядати контактну інформацію.

## 1.7. Практичність

### 1.7.1. Стандартизація

Система надає доступ до інформації кожному її користувачеві відповідно до його приналежності до однієї з чотирьох категорій.

### 1.7.2. Інтерфейс користувача

Інтерфейс сайту системи «Організація вуличних змагань WorkOut» відповідає наступним вимогам:

Зрозумілий і не допускає двозначного тлумачення.

Виконаний з урахуванням ергономічних вимог.

Всі кодовані параметри або елементи, наведені скорочення повинні мати розшифрування або вікно-підказку, що буде з'являтися після наведення курсору на елемент або після натискання спеціальної клавіші.

## 1.8. Надійність.

Протягом терміну зберігання архівів змагань повинна бути забезпечена їх цілісність.

Протягом терміну проведення одного турніру повинна бути забезпечена недоторканність та достовірність особистих сторінок учасників поточних змагань.

Для забезпечення збереження та цілісності організаторами буде використовуватися метод резервного копіювання.

Для забезпечення незмінності та достовірності організаторами буде використовуватися комплекс технологічних і адміністративних процедур, що перешкоджають випадковій або навмисній зміні збережених даних із бази.

Також повинна бути забезпечена конфіденційність персональної інформації. Надання доступу до персональних даних організатори здійснюють у відповідності з правами доступу відвідувача.

|  |               |          |        |     |                                  |    |
|--|---------------|----------|--------|-----|----------------------------------|----|
| Інв. № підп  | Підп. і дата  |          |        |     | Арк.                             |    |
|  | Взаєм. інв. № |          |        |     |                                  |    |
|  | Інв. № дубл.  |          |        |     |                                  |    |
|  | Підп. і дата  |          |        |     |                                  |    |
|  | Підп. і дата  |          |        |     |                                  |    |
| <p><b>1.8. Надійність.</b></p> <p>Протягом терміну зберігання архівів змагань повинна бути забезпечена їх цілісність.</p> <p>Протягом терміну проведення одного турніру повинна бути забезпечена недоторканність та достовірність особистих сторінок учасників поточних змагань.</p> <p>Для забезпечення збереження та цілісності організаторами буде використовуватися метод резервного копіювання.</p> <p>Для забезпечення незмінності та достовірності організаторами буде використовуватися комплекс технологічних і адміністративних процедур, що перешкоджають випадковій або навмисній зміні збережених даних із бази.</p> <p>Також повинна бути забезпечена конфіденційність персональної інформації. Надання доступу до персональних даних організатори здійснюють у відповідності з правами доступу відвідувача.</p> |               |          |        |     |                                  |    |
| Зм   | Арк.          | № докум. | Підпис | Дат | 6.050102 «Комп'ютерна інженерія» | 12 |

## РОЗДІЛ 2 РОЗРОБКА ІНФОРМАЦІЙНОГО ЗАБЕЗПЕЧЕННЯ

### 2.1. Загальна схема прецедентів

Загальна схема прецедентів для ролі гостя показує можливі послідовності дій даної категорії та встановлює зв'язок дій між різноманітними категоріями. Схема прецедентів представлена на рис. 2.1.

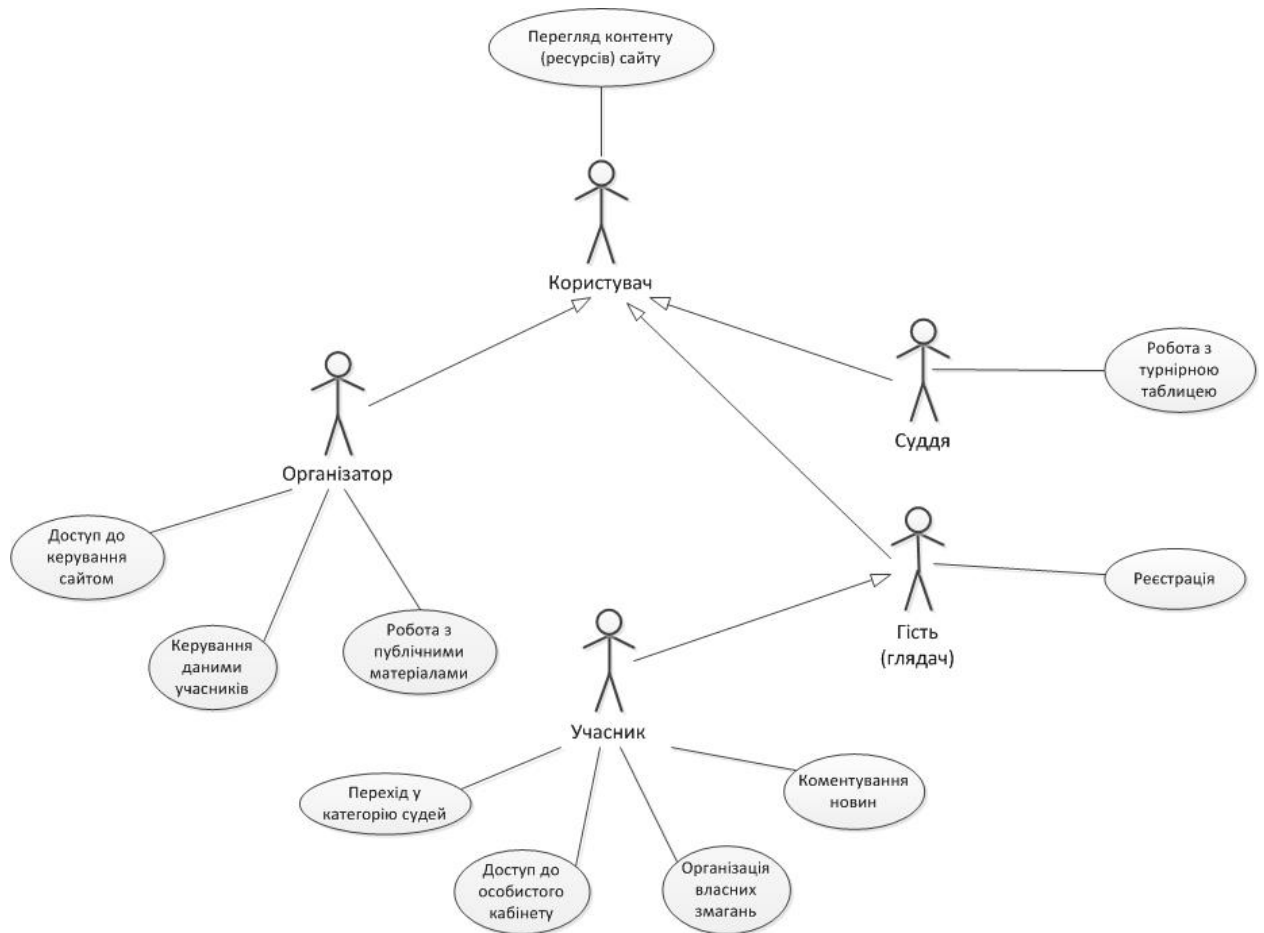


Рис. 2.1 – Загальна схема прецедентів для ролі головного адміністратора

### 2.2. Прецеденти для ролі гостя

Нижче описані процедури для ролі користувача з вказаними передумовами, результатом, виключними ситуаціями та детальним описом послідовності дій.

#### Перецент №1 Реєстрація у змаганнях

**Назва:** Реєстрація у змаганнях.

**Учасники:** гість, система.

**Передумови:** гість не зареєстрований у системі.

|               |  |
|---------------|--|
| Підп. і дата  |  |
| Взаєм. інв. № |  |
| Інв. № дубл.  |  |
| Підп. і дата  |  |
| Інв. № підп   |  |

|    |      |          |        |     |
|----|------|----------|--------|-----|
| Зм | Арк. | № докум. | Підпис | Дат |
|    |      |          |        |     |

**Результат:** зареєстровано нового учасника змагань.

**Основний сценарій:**

1. Гість заповнює анкету своїми даними на сайті. Для підтвердження особи він прикріплює фотографію або скан будь-якого документу, що засвідчує особу.
2. Гість надсилає анкету організаторам.
3. Автоматично створюється неактивний профіль учасника.
4. Система надсилає на адресу електронної пошти гостя запит про підтвердження реєстрації.
5. Гість підтверджує факт реєстрації у змаганнях.
6. Система створює для учасника особистий електронний кабінет.
7. Учасник отримує персональний ідентифікаційний номер та пароль для подальшого доступу до особистого кабінету.
8. Система активує профіль учасника.

**Виключні ситуації:**

1. Гість невірно заповнив поля анкети або залишив їх порожніми.
2. Гість не підтвердив факт реєстрації у змаганнях.
3. Порушено зв'язок між сервером із базою даних та сайтом.

|             |              |              |               |              |                                  |  |  |  |  |      |
|-------------|--------------|--------------|---------------|--------------|----------------------------------|--|--|--|--|------|
| Інв. № підп | Підп. і дата | Інв. № дубл. | Взаєм. інв. № | Підп. і дата |                                  |  |  |  |  | Арк. |
|             |              |              |               |              |                                  |  |  |  |  | 14   |
|             |              |              |               |              |                                  |  |  |  |  |      |
|             |              |              |               |              |                                  |  |  |  |  |      |
| Зм          | Арк.         | № докум.     | Підпис        | Дат          | 6.050102 «Комп'ютерна інженерія» |  |  |  |  |      |

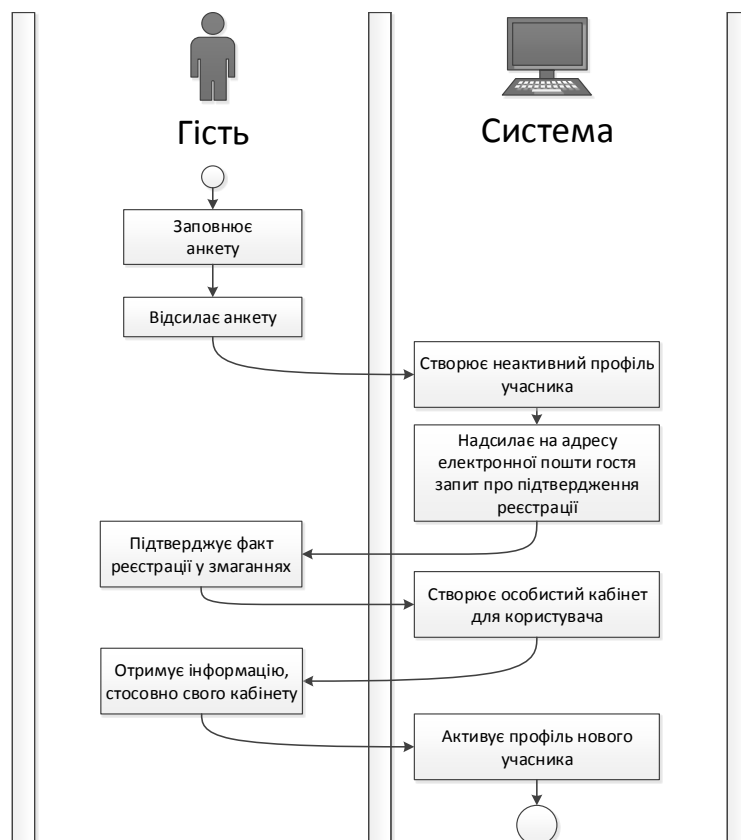


Рис. 2.2 – Схема реєстрації

## Прецидент №2. Доступ до особистого кабінету

**Назва:** Доступ до особистого кабінету.

**Учасники:** учасник, система.

**Передумови:** учасник зареєстрований у змаганнях.

**Результат:** перевірено поточні дані учасника.

**Сценарій:**

1. Учасник входить в систему.
2. Учасник вводить ідентифікаційний номер та пароль.
3. Учасник отримує доступ до особистого кабінету.
4. Система створює сеанс роботи.
5. Учасник переглядає інформацію щодо своїх успіхів, а також місця і часу проведення наступного етапу змагань, у якому він бере участь.
6. Учасник виходить із особистого кабінету.

Підп. і дата

Взаєм. інв. №

Інв. № дубл.

Підп. і дата

Інв. № підп

7. Система завершує сеанс роботи.

### Виключні ситуації:

Учасник невірно ввів ідентифікаційний номер та пароль або залишив поля порожніми.



Рис. 2.3 – Схема доступу до кабінету

|               |  |
|---------------|--|
| Підп. і дата  |  |
| Взаєм. інв. № |  |
| Інв. № дубл.  |  |
| Підп. і дата  |  |
| Інв. № підп   |  |

|    |      |          |        |     |
|----|------|----------|--------|-----|
| Зм | Арк. | № докум. | Підпис | Дат |
|    |      |          |        |     |

### 2.3. Діаграма бізнес-сутностей

Дана діаграма створюється на етапі бізнес моделювання. Вона відображає основні сутності та взаємозв'язки між ними. В даному випадку вона демонструє зв'язок між сутностями: організатор, суддя, гість та учасник. Діаграма бізнес-сутностей проекту зображена на рис. 2.8.

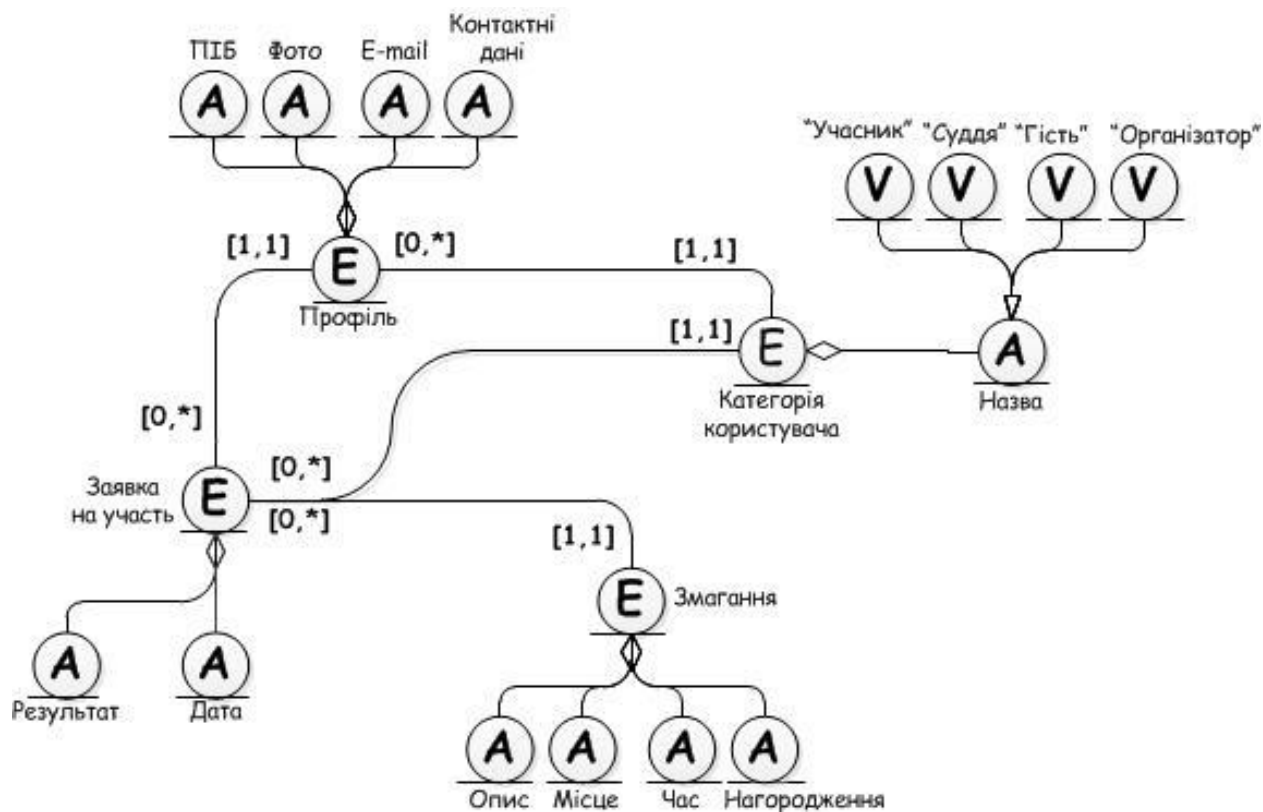


Рис. 2.8 – Діаграма бізнес-сутностей

### 2.4. Реляційна модель бази даних

Реляційна модель бази даних (рис 2.3) зображує структуру таблиць бази даних, взаємозв'язки між ними та поля кожної з таблиць. Наведена діаграма має багато схожого з діаграмою бізнес-сутностей. Кожній основній бізнес-сутності відповідає таблиця баз даних.

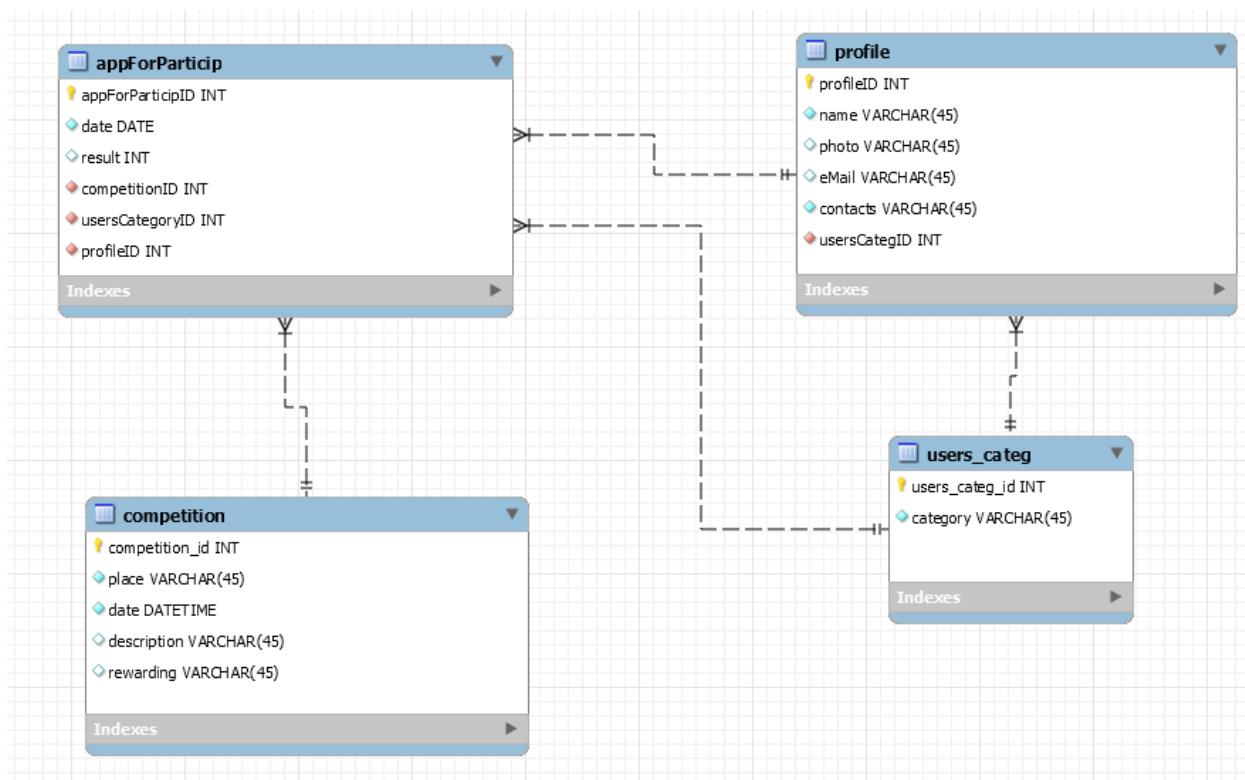


Рис 2.9 – Реляційна модель

## 2.5. Специфікація таблиць бази даних

Специфікація таблиць бази даних включає в себе інформацію про назви колонок таблиці, їхній тип, інформацію про те, чи є ця колонка первинним ключем, чи поле може бути пустим, чи значення поля автоматично збільшується та коментар щодо призначення колонки. Таблиці зі специфікаціями наведені нижче.

Таблиця 2.1

Таблиця **appForParticip** (application for participation)

| appForParticip | date       | result | competitionID | usersCategoryID | profileID |
|----------------|------------|--------|---------------|-----------------|-----------|
| 1              | 2014-11-15 | NULL   | 1             | 1               | 1         |
| 2              | 2014-11-19 | NULL   | 2             | 1               | 1         |
| 3              | 2014-11-15 | NULL   | 1             | 2               | 2         |
| 4              | 2014-11-19 | NULL   | 2             | 2               | 2         |
| 5              | 2014-11-15 | NULL   | 1             | 3               | 3         |
| 6              | 2014-11-19 | NULL   | 2             | 3               | 3         |
| 7              | 2014-11-15 | NULL   | 1             | 4               | 4         |
| 8              | 2014-11-19 | NULL   | 2             | 4               | 4         |



Таблиця 2.2

Таблиця **Profile**

| profileID | name            | photo | e-mail | contacts   | UsersCategID |
|-----------|-----------------|-------|--------|------------|--------------|
| 1         | Pavluchkov Vlad | NULL  | NULL   | 0981223425 | 1            |
| 2         | Zmeul Evgeniy   | NULL  | NULL   | 0671234569 | 2            |
| 3         | Morozov Maks    | NULL  | NULL   | 0671234565 | 3            |
| 4         | Popenko Ruslan  | NULL  | NULL   | 0984582499 | 4            |

Таблиця 2.3

Таблиця **competition**

| competition_id | Place                          | Date                   | Description | Rewarding |
|----------------|--------------------------------|------------------------|-------------|-----------|
| 1              | Stadium Start                  | 2014-11-15<br>19:00:00 | NULL        | NULL      |
| 2              | Metro Gidropark bus<br>station | 2014-11-19<br>19:00:00 | NULL        | NULL      |

Таблиця 2.4

Таблиця **users\_categ**

| users_categ_id | category    |
|----------------|-------------|
| 1              | participant |
| 2              | judge       |
| 3              | guest       |
| 4              | organizer   |

|               |  |
|---------------|--|
| Підп. і дата  |  |
| Взаєм. інв. № |  |
| Інв. № дубл.  |  |
| Підп. і дата  |  |
| Інв. № підп   |  |

|    |      |          |        |     |
|----|------|----------|--------|-----|
|    |      |          |        |     |
| Зм | Арк. | № докум. | Підпис | Дат |

## РОЗДІЛ 3

### РОЗРОБКА ПРОГРАМНОГО ПРОДУКТУ

#### 3.1. Реляційно-об'єктне відображення

Для реляційно-об'єктного відображення в програмі використовується бібліотека Hibernate. Вона надає можливість легко встановити зв'язок з будь-якою базою даних та створити відображення між об'єктно-орієнтованою моделлю та традиційною реляційною моделлю баз даних. На рис. 3.1 зображено діаграму Entity класів. Детальна специфікація (JavaDoc) наведена нижче.

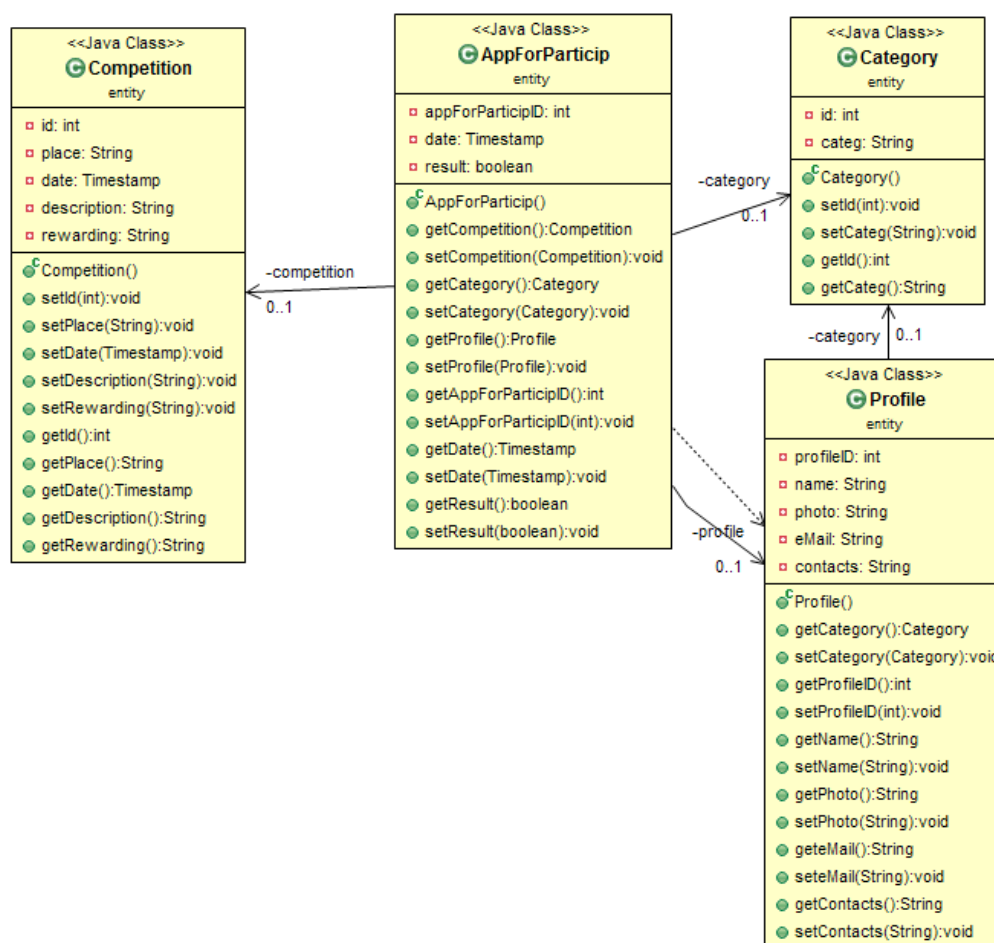


Рис 3.1 – Діаграма entity класів

##### 3.1.1. Клас «AppForParticip»

```

@Entity
public class AppForParticip
extends java.lang.Object

```

Клас представляє сутність Заявка на участь

Since:  
2015-04-14

Version:  
1.0

Author:  
Руслан Попенко

#### Field Summary

##### Fields

| Modifier and Type          | Field and Description                              |
|----------------------------|--|
| private int                | appForParticipID<br>Ідентифікатор заявки на участь |
| private Category           | category<br>Ідентифікатор категорії користувача    |
| private Competition        | competition<br>Ідентифікатор змагання              |
| private java.sql.Timestamp | date<br>Дата створення заявки                      |
| private Profile            | profile<br>Ідентифікатор профілю                   |
| private boolean            | result<br>Результат реєстрації заявки              |

#### Constructor Summary

##### Constructors

| Constructor and Description |
|-----------------------------|
| AppForParticip()            |

#### Method Summary

| All Methods        | Instance Methods                          | Concrete Methods |
|--------------------|---|------------------|
| Modifier and Type  | Method and Description                    |                  |
| int                | getAppForParticipID()                     |                  |
| Category           | getCategory()                             |                  |
| Competition        | getCompetition()                          |                  |
| java.sql.Timestamp | getDate()                                 |                  |
| Profile            | getProfile()                              |                  |
| boolean            | getResult()                               |                  |
| void               | setAppForParticipID(int appForParticipID) |                  |
| void               | setCategory(Category category)            |                  |
| void               | setCompetition(Competition competition)   |                  |
| void               | setDate(java.sql.Timestamp date)          |                  |
| void               | setProfile(Profile profile)               |                  |
| void               | setResult(boolean result)                 |                  |

### 3.1.2. Клас «Category»

```
@Entity
public class Category
extends java.lang.Object
```

Клас представляє сутність Категорія

Since:

2015-04-14

Version:

1.0

Author:

Руслан Попенко

#### Field Summary

##### Fields

| Modifier and Type        | Field and Description          |
|--------------------------|--------------------------------|
| private java.lang.String | categ<br>Категорія користувача |
| private int              | id<br>Ідентифікатор категорії  |

#### Constructor Summary

##### Constructors

| Constructor and Description |
|-----------------------------|
| Category()                  |

#### Method Summary

##### All Methods

##### Instance Methods

##### Concrete Methods

| Modifier and Type | Method and Description           |
|-------------------|----------------------------------|
| java.lang.String  | getCateg()                       |
| int               | getId()                          |
| void              | setCateg(java.lang.String categ) |
| void              | setId(int id)                    |

### 3.1.3. Клас «Competition»

```
@Entity
public class Competition
extends java.lang.Object
```

Клас представляє сутність Змагання

Since:

2015-04-14

Version:

1.0

Author:

Руслан Попенко

##### Fields

| Modifier and Type          | Field and Description                               |
|----------------------------|---|
| private java.sql.Timestamp | <b>date</b><br>Дата змагання                        |
| private java.lang.String   | <b>description</b><br>Опис змагання                 |
| private int                | <b>id</b><br>Ідентифікатор змагання                 |
| private java.lang.String   | <b>place</b><br>Місце змагання                      |
| private java.lang.String   | <b>rewarding</b><br>Нагородження учасників змагання |

### Method Summary

| All Methods        | Instance Methods | Concrete Methods  |
|--------------------|------------------|---|
| Modifier and Type  |                  | Method and Description                                    |
| java.sql.Timestamp |                  | <code>getDate()</code>                                    |
| java.lang.String   |                  | <code>getDescription()</code>                             |
| int                |                  | <code>getId()</code>                                      |
| java.lang.String   |                  | <code>getPlace()</code>                                   |
| java.lang.String   |                  | <code>getRewarding()</code>                               |
| void               |                  | <code>setDate(java.sql.Timestamp date)</code>             |
| void               |                  | <code>setDescription(java.lang.String description)</code> |
| void               |                  | <code>setId(int id)</code>                                |
| void               |                  | <code>setPlace(java.lang.String place)</code>             |
| void               |                  | <code>setRewarding(java.lang.String rewarding)</code>     |

### 3.1.4. Клас «Profile»

```
@Entity
public class Profile
extends java.lang.Object
```

Клас представляє сутність Профіль

Since:

2015-04-14

Version:

1.0

Author:

Руслан Попенко

| Fields                   |  |
|--------------------------|--|
| Modifier and Type        | Field and Description                            |
| private Category         | <code>category</code><br>Ідентифікатор категорії |
| private java.lang.String | <code>contacts</code><br>Контактний телефон      |
| private java.lang.String | <code>eMail</code><br>Пошта                      |
| private java.lang.String | <code>name</code><br>Ім'я                        |
| private java.lang.String | <code>photo</code><br>Фото                       |
| private int              | <code>profileID</code><br>Ідентифікатор профіля  |

### Constructor Summary

#### Constructors

Constructor and Description

`Profile()`

Підп. і дата

Взаєм. інв. №

Інв. № дубл.

Підп. і дата

Інв. № підп

|    |      |          |        |     |
|----|------|----------|--------|-----|
|    |      |          |        |     |
| Зм | Арк. | № докум. | Підпис | Дат |

6.050102 «Комп'ютерна інженерія»

Арк.

23

### 3.2. Специфікація HibernateUtil класу

Клас HibernateUtil слугує для того, щоб при взаємодії із файлами конфігурації створювати hibernate-сесію. Діаграму цього інтерфейса можна побачити на рис. 3.2. Детальна специфікація (JavaDoc) наведена нижче.

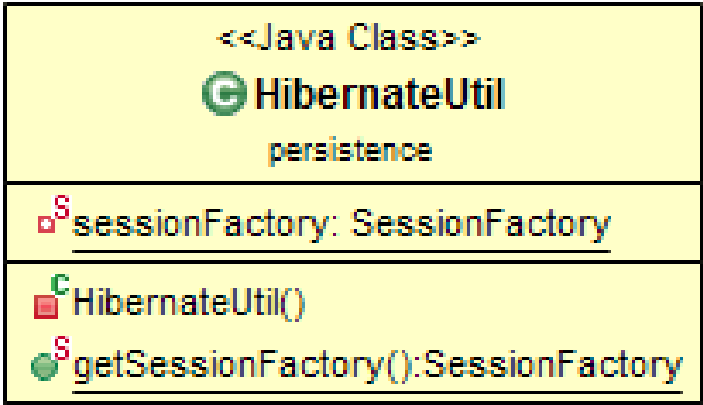


Рис 3.2. – Діаграма HibernateUnit класу

#### 3.2.1. Клас «HibernateUtil»

```
public class HibernateUtil
extends java.lang.Object
```

Клас для взаємодії з конфіг файлами і створення об'єкту SessionFactory, котрий відповідає за створення hibernate-сесії

Since:

2015-04-14

Version:

1.0

Author:

Руслан Попенко

##### Field Summary

###### Fields

| Modifier and Type                           | Field and Description  |
|---|--|
| private static org.hibernate.SessionFactory | <code>sessionFactory</code><br>об'єкт SessionFactory, котрий відповідає за створення hibernate-сесії |

##### Constructor Summary

###### Constructors

| Modifier | Constructor and Description  |
|----------|------------------------------|
| private  | <code>HibernateUtil()</code> |

##### Method Summary

###### All Methods

###### Static Methods

###### Concrete Methods

| Modifier and Type                   | Method and Description   |
|-------------------------------------|--|
| static org.hibernate.SessionFactory | <code>getSessionFactory()</code><br>Повертає об'єкт SessionFactory |

### 3.3. Специфікація DAO-класів

Класи, що тут представлені, містять методи для роботи з базою даних. Ці методи використовують бібліотеку Hibernate. Діаграму цих класів можна побачити на рис. 3.3. Детальна специфікація наведена в додатку В.

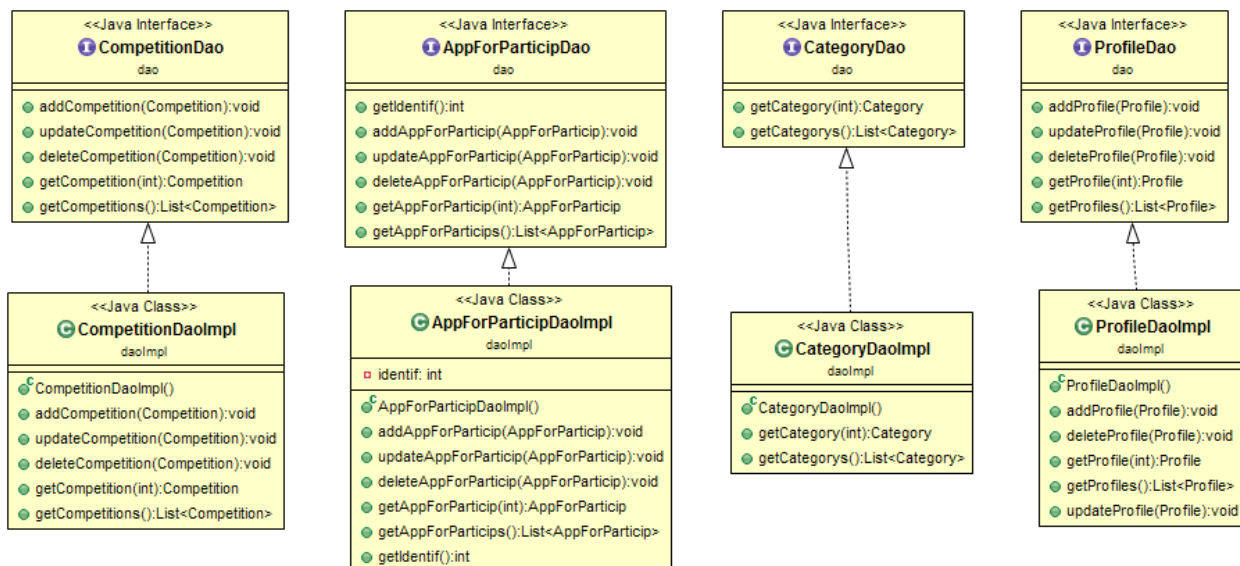


Рис 3.3 – Діаграма DAO класів

### 3.4. Класи контролерів та їх специфікація

Дані класи призначені для створення зв'язку між сервером та клієнтом. Діаграму цих класів можна побачити на рис. 3.4. Детальна специфікація наведена в додатку В.

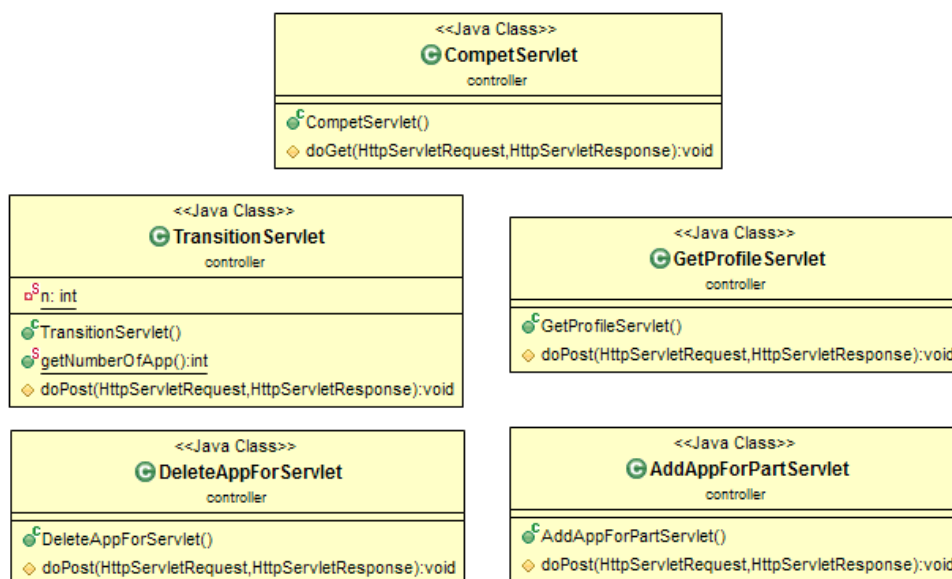


Рис. 3.4 – Діаграма класів контролерів

|               |  |
|---------------|--|
| Підп. і дата  |  |
| Взаєм. інв. № |  |
| Інв. № дубл.  |  |
| Підп. і дата  |  |
| Інв. № підп   |  |

|    |      |          |        |     |
|----|------|----------|--------|-----|
| Зм | Арк. | № докум. | Підпис | Дат |
|----|------|----------|--------|-----|



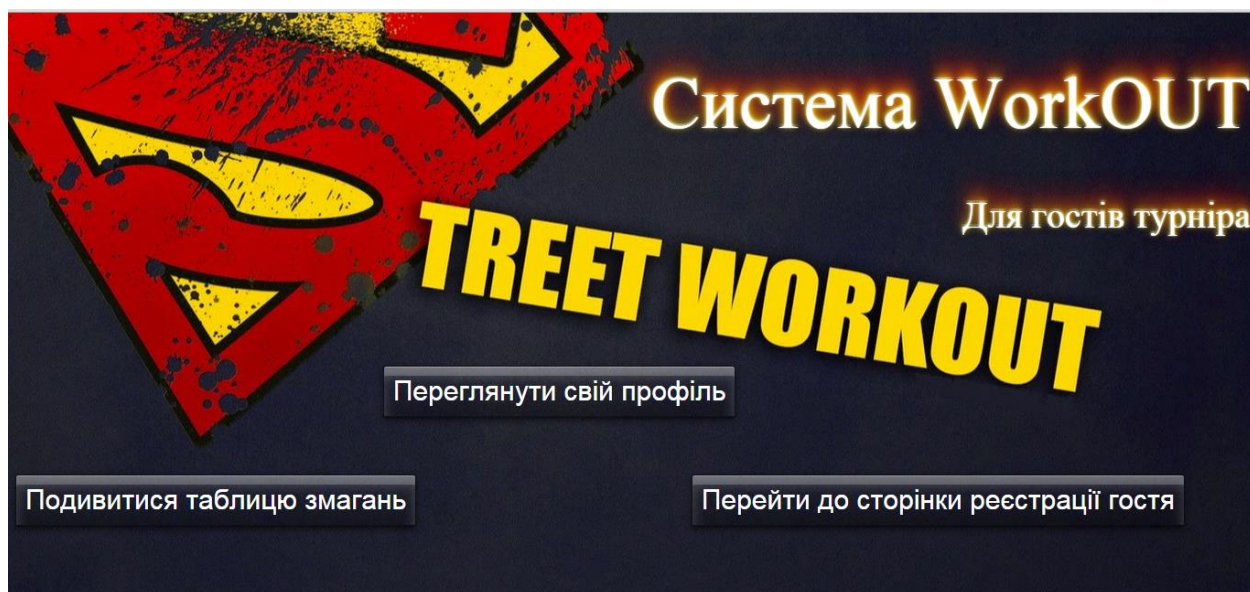
## РОЗДІЛ 4

### ІЛЮСТРАЦІЯ РОБОТИ ПРОГРАМИ

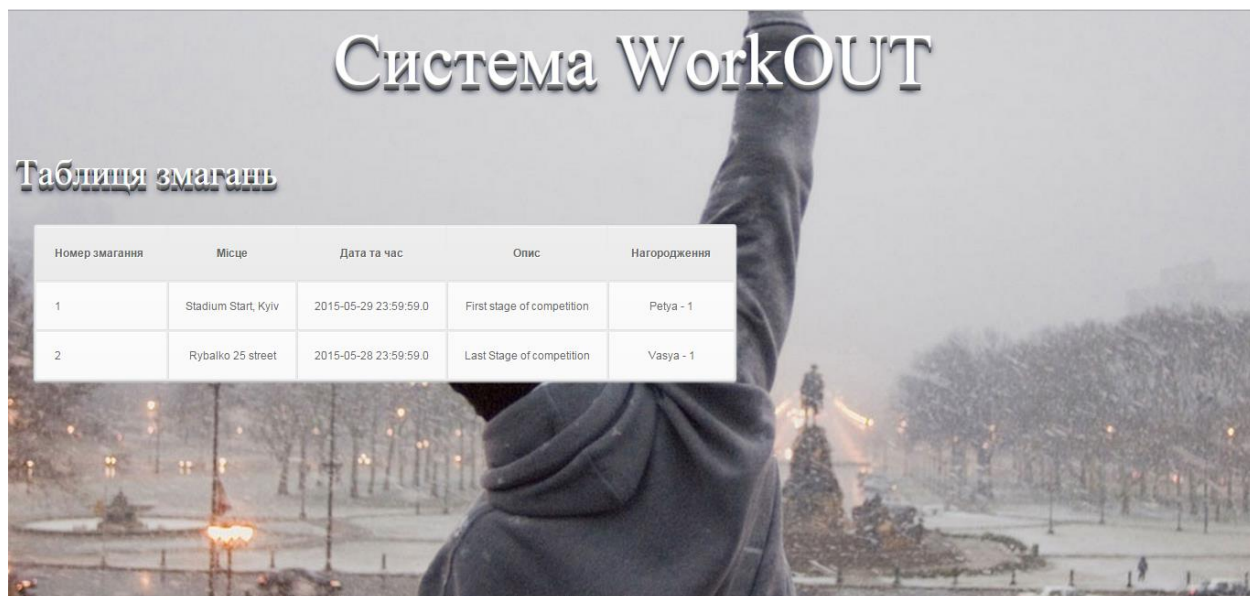
Для ілюстрації роботи програми в цьому розділі наведено графічні сценарії роботи проекту.

#### 4.1. Взаємодія гостя турніра і системи.

**4.1.1.** Головна сторінка, на якій можна зробити вибір, яку дію потрібно зробити користувачу.



**4.1.2.** Перегляд турнірної таблиці.





**4.1.3.** Реєстрація гостя. На сторінці зроблений годинник, для того щоб користувач міг подивитися час реєстрації.

**Система WorkOUT**

**2:28:10**

**СТРЕТ-ВОРКУАТ МІГРАЦІЯ**

**Реєстрація гостя**

Заповніть всі поля

Ім'я:

E-mail:

Контактний телефон

(продовження форми)

**Реєстрація гостя**

Заповніть всі поля

Ім'я:

E-mail:

Контактний телефон

Введіть номер змагання

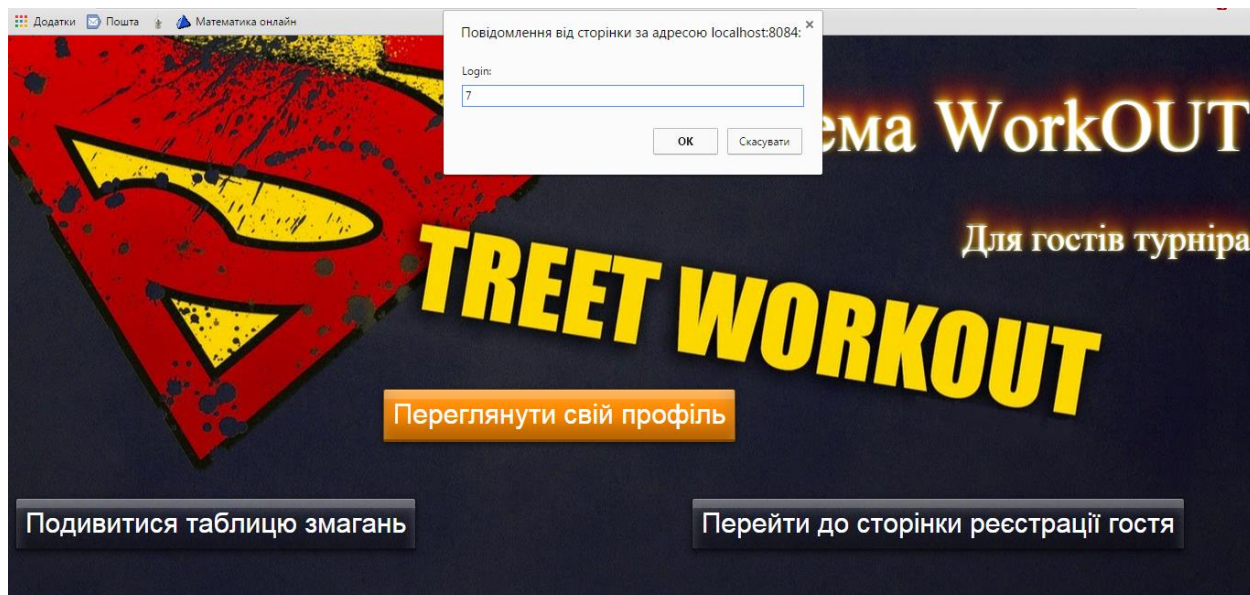
**Зареєструватися**

**4.1.4.** Перегляд заявки і профіля гостя.

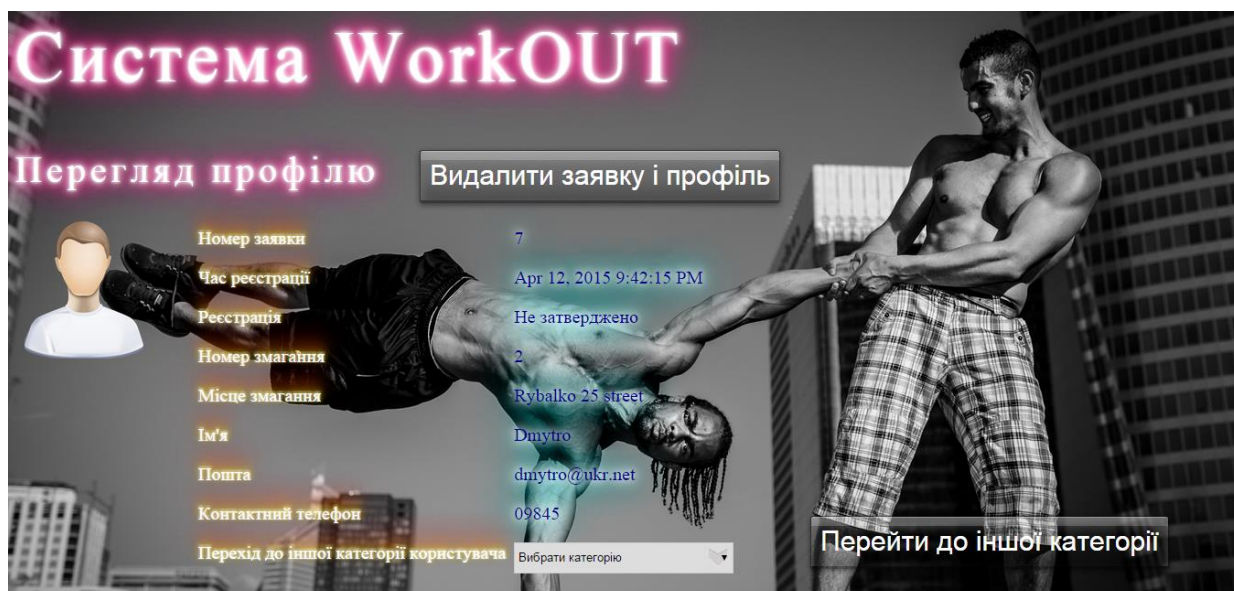
4.1.4.1. Спочатку треба пройти авторизацію

|              |               |              |              |             |
|--------------|---------------|--------------|--------------|-------------|
| Підп. і дата | Взаєм. інв. № | Інв. № дубл. | Підп. і дата | Інв. № підп |
|              |               |              |              |             |

|    |      |          |        |     |
|----|------|----------|--------|-----|
|    |      |          |        |     |
| Зм | Арк. | № докум. | Підпис | Дат |



4.1.4.2. Потім з'явиться інформація про гостя.



|               |  |
|---------------|--|
| Підп. і дата  |  |
| Взаєм. інв. № |  |
| Інв. № дубл.  |  |
| Підп. і дата  |  |
| Інв. № підп   |  |

|    |      |          |        |     |
|----|------|----------|--------|-----|
|    |      |          |        |     |
| Зм | Арк. | № докум. | Підпис | Дат |

## СПИСОК ІНФОРМАЦІЙНИХ ДЖЕРЕЛ

1. Apache Tomcat. – Посилання: <http://tomcat.apache.org>
2. Hibernate. – Посилання: <http://hibernate.org>
3. Maven. – Посилання: <https://maven.apache.org>
4. Git. – Посилання: <http://git-scm.com>
5. GitHub. – Посилання: <https://github.com>

|             |              |  |              |               |              |
|-------------|--------------|--|--------------|---------------|--------------|
| Інв. № пілп | Підп. і дата |  | Інв. № дубл. | Взаєм. інв. № | Підп. і дата |
|             |              |  |              |               |              |
|             |              |  |              |               |              |

|    |      |          |        |     |                                  |      |
|----|------|----------|--------|-----|----------------------------------|------|
|    |      |          |        |     | 6.050102 «Комп'ютерна інженерія» | Арк. |
| Зм | Арк. | № докум. | Підпис | Дат |                                  | 29   |