**Факультет прикладної математики**

**Кафедра програмного забезпечення комп’ютерних систем**

**ШАБЛОНИ ПРОЕКТУВАННЯ В ООП.**

**Гра “П’ятнашки”**

**Технічне завдання**

“ПОГОДЖЕНО”

Керівник:

\_\_\_\_\_\_\_\_\_\_ Т.М. Заболотня

Виконавець:

“\_\_” \_\_\_\_\_\_\_\_\_\_ 2018 р. \_\_\_\_\_\_\_\_\_\_ М.В. Свинарчук

КИЇВ 2018

**ЗМІСТ**

стор.

1. НАЙМЕНУВАННЯ ТА ГАЛУЗЬ ЗАСТОСУВАННЯ 3

2. ПІДСТАВИ ДЛЯ РОЗРОБЛЕННЯ 3

3. ПРИЗНАЧЕННЯ РОЗРОБКИ 3

4. ТЕХНІЧНІ ВИМОГИ 3

5. ВИМОГИ ДО ПРОЕКТНОЇ ДОКУМЕНТАЦІЇ 6

6. ЕТАПИ РОЗРОБЛЕННЯ 6

**1.** **НАЙМЕНУВАННЯ ТА ГАЛУЗЬ ЗАСТОСУВАННЯ**

Найменування: Гра “П’ятнашки”.

Галузь застосування: ігрова сфера.

**2.** **ПІДСТАВИ ДЛЯ РОЗРОБЛЕННЯ**

Підставою для розроблення є завдання на виконання курсової роботи з дисципліни «Об’єктно-орієнтоване програмування» студентами ІІ курсу кафедри програмного забезпечення комп’ютерних систем Національного технічного університету України «Київський політехнічний інститут».

**3.** **ПРИЗНАЧЕННЯ РОЗРОБКИ**

Розробка призначена для використання як ігровий додаток з метою приємного проведення вільного часу, розвитку моторики очей та логічного складу розуму гравця.

**4.** **ТЕХНІЧНІ ВИМОГИ**

**4.1.** **Вимоги до програмного продукту, що розробляється**

Розроблене програмне забезпечення повинне дозволяти працювати з такими об’єктами:

* **модель гри** – об’єкт, який власне і представляє сутність всієї гри. Містить у собі всю логіку, яка побудована на основі взаємодії інших об’єктів: ігрове поле (дошка), контроллер гри, тощо. Даний об’єкт може перебувати в 4 станах: гра очікує на початок, гра в процесі, гра призупинена та гра завершена.
* **ігрове поле (дошка)** – об’єкт, який містить у собі фіксований набір кнопок та інформацію про їх місцерозташування, обов’язкова містить заблоковану кнопку, яка представляє собою порожню клітинку на полі. Відповідальний за те, як гравцю буде відображено гру. Є два типи: текстова дошка (містить кнопки з числовими значеннями) та дошка з зображенням (містить кнопки з частинами певного зображення).
* **кнопка** – об’єкт, власне з яким і взаємодіє гравець. Натискаючи на кнопку відбувається переміщення з іншими кнопками (за виконання певних умов). Є два типи: текстова кнопка (з порядковим номером) та кнопка з зображенням.
* **контроллер гри –** об’єкт, відповідальний за те, яким чином будуть переміщуватися кнопки на ігровому полі, контролює підсвічення кнопок, запам’ятовує порядок переміщення та за потреби може відмінити зроблені раніше ходи гравця, також перевіряє стан ігрового поля та в разі чого повідомляє повідомляє **модель гри** про завершення (момент, коли всі кнопки на ігровому полі стали в правильні позиції – картинка зібрана або числа розташовані в правильному порядку).
* **налаштування –** сутність, яка напряму впливає на **модель гри**. Відповідає за складність гри, визначає чи будуть підсвічуватися кнопки, відтворюватися звукові ефекти, чи увімкнений режим hardmode, який тип ігрового поля буде відображений та яким чином будуть переміщуватися кнопки. Гравець може в будь-який момент змінювати налаштування, які в свою чергу відразу змінять поведінку гри. Зберігається в окремий файл, відповідно відновлює свій стан під час запуску гри.
* **таблиця рекордів –** об’єкт, який містить інформацію про результати гравців, що вже пройшли гру. Підтримує сортування за різними параметрами. Зберігається в окремий файл, відповідно відновлює свій стан під час запуску гри.

Розроблювана програма повинна забезпечувати:

* користувацький інтерфейс;
* можливість перезапуску гри в будь-який час;
* можливість зупинити гру в будь-який час;
* вибір типу ігрового поля: нумероване поле, та поле з зображенням;
* вибір складності гри: 3х3 чи 4х4;
* відтворення звукових ефектів;
* підсвічування кнопок, що стоять на правильних позиціях;
* два способи переміщення кнопок;
* зміну налаштувань гри (див. вище) під час виконання самої програми, та збереження користувацьких налаштувань з відтворенням їх стану при наступному запуску гри;
* підрахунок часу та кількості перемикань під час гри;
* збереження результату гри в таблицю рейтингу, яка записується в окремий файл та відтворюється при кожному запуску гри.

**4.2.** **Вимоги до програмного забезпечення**

* операційна система Windows, Linux, Mac OS ;
* встановлений JDK 8u171

**4.3. Вимоги до апаратної частини**

* процесор Intel Pentium II і вище;
* оперативна пам’ять не менше 128 Мбайт;
* вільний простір жорсткого диска не менше 25 Мбайт;

**5.** **ВИМОГИ ДО ПРОЕКТНОЇ ДОКУМЕНТАЦІЇ**

У процесі виконання курсової роботи має бути розроблена така документація:

1. пояснювальна записка;
2. керівництво користувача (опис інтерфейсу користувача);
3. графічний матеріал:
4. структурна схема розробленого ПЗ;
5. діаграма класів;
6. діаграми реалізованих шаблонів проектування;
7. ілюстрації до опису функціональних можливостей ПЗ та інтерфейсу користувача.

**6.** **ЕТАПИ РОЗРОБЛЕННЯ**

|  |  |
| --- | --- |
| **№** | **Назва етапів розроблення** |
| **1** | Затвердження теми курсової роботи. Опрацювання відповідної літератури. Розроблення та узгодження технічного завдання, оформлення аркуша завдання. |
| **2** | Аналіз постановки задачі, виділення абстракцій в предметній області. |
| **3** | Вибір та дослідження методів створення програми, вибір відповідних структур даних та шаблонів проектування. Перше узгодження з керівником. |
| **4** | Доопрацювання структури модулів та класів з урахуванням пропозицій керівника. Проектування інтерфейсу користувача. |
| **5** | Розроблення основних алгоритмів роботи програми та проектування інтерфейсу. Друге узгодження з керівником. |
| **№** | **Назва етапів розроблення** |
| **6** | Програмна реалізація. |
| **7** | Демонстрування першого варіанту. Третє узгодження з керівником. |
| **8** | Тестування програми. |
| **9** | Аналіз результатів. Підготовка матеріалів КР та оформлення документації |
| **10** | Захист курсової роботи |