**ПРЕДМЕТ: РАЗРАБОТКА НА СОФТУЕР**

**ДЕЙНОСТ: УЧЕНИЧЕСКИ ПРАКТИКИ 2**

***2021-2022***

**Тема: „Модел на приложение за шах“**

**Курсов проект**

*Янислав Стоянов*

БургасСъдържание

[1 Въведение 3](#_Toc96509792)

[2 Цели и обхват на софтуерното приложение 3](#_Toc96509793)

[3 Анализ на решението 3](#_Toc96509794)

[3.1 Потребителски изисквания и работен процес 3](#_Toc96509795)

[3.2 Примерен потребителски интерфейс 4](#_Toc96509796)

[3.3 Диаграми на анализа 4](#_Toc96509797)

[3.4 Модел на съдържанието / данните 5](#_Toc96509798)

[4 Дизайн 5](#_Toc96509799)

[4.1 Реализация на структура на приложението (3-layer), Разделение на кода според предназначението му 5](#_Toc96509800)

[4.2 Организация и код на заявките към база от данни 5](#_Toc96509801)

[4.3 Наличие и интуитивност на потребителски интерфейс (конзолен, графичен, уеб) 5](#_Toc96509802)

[5 Тестване 6](#_Toc96509803)

[6 Заключение и възможно бъдещо развитие 6](#_Toc96509804)

[7 Използвани литературни източници и Уеб сайтове 6](#_Toc96509805)

[8 Приложения 6](#_Toc96509806)

[9 Критерии и показатели за оценяване 8](#_Toc96509807)

# Въведение

Моделът за шах представлява приложение, целящо да запознае потребителите как определена програма за шах може да изглежда, тя може да представи обучение и възможност за подобряване на техните шахматни умения. Ето някои от основните функционалности, които такова приложение може да предложи:

Игра срещу компютър: Приложението може да предоставя възможност за игра срещу компютъра на различни нива на трудност.

Игра срещу втори играч на локално ниво: Потребителите могат да играят шах срещу други играчи от същото устройство.

Обучение и тренировки: Приложението може да предлага обучителни материали, които помагат на потребителите да научат основите на шаха и да развият своите умения. Това може да включва интерактивни уроци, задачи за решаване и тренировъчни режими.

Логин и регистър форма: Приложението предоставя два визуални модела за логин и регистър форми, които играчът може да избере.

# Цели и обхват на софтуерното приложение

Софтуерното приложение има за различни цели и функционалности, които може да има:

Игра с изкуствен интелект: Една от възможните цели на приложението е да предостави потребителите възможността да играят шах с изкуствен интелект. Приложението осигурява удобна и лесна за използване интерфейсна система.

Обучение и тренировки: Важна цел на приложението е предоставянето на материали за обучение, както и увод за начинаещи.

Игра срещу друг човек на локално ниво: Приложението може да позволява на потребителите да играят шах срещу други играчи онлайн. Това включва възможността за търсене на опоненти, създаване на игрови стаи и комуникация между играчите.

База данни и логин/регистър форма: Приложението може да предлага база данни със записи на влезлите и регистрираните потребители, както на техните имейли и пароли.

# Анализ на решението

## Потребителски изисквания и работен процес

1. Диаграмата на последователност илюстрира последователността на съобщенията между обектите и итерациите. Действието започва, след като играчът направи ход, тогава системата информира изкуствения интелект, а той прави ход. Алтернатива: променяне, сменяне на играта. Информира се системата, спира/ сменя се играта.
2. Чрез клас диаграмата се описват класове, обекти, пакети и връзките между тях. Клас „Шах“: обекти: дъска(цвят, фигури, ходове), играч(бял, черен). Фигури: Цар, дама, офицер, кон, топ, пешка. Квадрат: ред, колона.
3. Чрез диаграмата на случаи на използване се описват връзките между участниците и случаите. Изкуственият интелект може да направи ход, специфични ходове: промотиране на пешка, рокадо. Играчът може да прави същите ходове, както изкуствения интелект, като той избира дали да играе или да се откаже. Позволени са три репетиции на ход, след което играта бива спряна.
4. Диаграмата субект-връзка (или ER модел) описва взаимосвързани неща, които представляват интерес в конкретна област на знанието. Изкуствен интелект: мислене, отговор, избор, взимане, местене. Обратна връзка: ИД, бот\_ИД, ниво на бот, потребител, срок на годност, отговор. Движение: Пешка, кон, офицер, топ, дама, цар. Играч срещу играч: Вземане, местене, рокадо, ело. Потребител: ИД, име, парола, логин, връзка, потребителско име. Специални движения: рокадо, промотиране. Група: ид на група, ид на потребител, име, фамилия, ело, имейл.

## Примерен потребителски интерфейс



## Диаграми на анализа

<https://lucid.app/lucidchart/7587e6e8-a1f8-468c-8e75-71bfd85a5e11/edit?invitationId=inv_a86ad2f7-165f-4092-8e21-9940a8b5f11c>

<https://lucid.app/lucidchart/83d0e1bb-a044-4992-9cd4-828ad7065c34/edit?invitationId=inv_2ea79f74-d3a5-4e8f-96e4-c3eb868c1770>

<https://lucid.app/lucidchart/cf827fee-b898-4af6-902d-b4ba40ed4c54/edit?invitationId=inv_038d72f9-72bb-4963-ac8a-987b9338d022>

https://lucid.app/lucidchart/18292491-d708-42a3-bb57-19b5e35ad946/edit?invitationId=inv\_96daf26a-def6-422d-a43e-5cd83596bb74

* Онагледяване на базата данни чрез ER, клас, последователност и комуникация диаграма

## Модел на съдържанието / данните

Тук опишете модела на данните/съдържанието - текстово, графично и евентуално аудио/видео съдържание), което ще представите в проекта си. Опишете размера и типа на данните/файловете и начина на кодиране за всеки от ресурсите.

Ако ползвате в проекта си текстово и/или мултимедийно съдържание от различни типове, представете неговата структура, напр. посредством таксономия, типология, онтология или други схеми за представяне на структурата от категории, под-категории, типове и т.н., както и техните взаимовръзки с други категории или типове, напр. географски региони и дялове, области/сегменти от промишлеността, и др.

След структурата опишете и възможните характеристики, атрибути и честота на срещане на всеки един ресурс в съдържанието (категория, тип, екземпляр, връзка/релация и т.н.).

# Дизайн

Използван е графичен потребителски интерфейс. Класът ChessMenuGUI създава графичен интерфейс на менюто, което се използва за навигация .Класът Board (Дъска), представляващ шахматната дъска. Класът LoginRegisterGUI има за цел да регистрира определен потребител с помощта на база данни, която сравнява минали влизания от всички потребители.

## Реализация на структура на приложението (3-layer), Разделение на кода според предназначението му

Слой на потребителския интерфейс:

Описание: Този слой се грижи за визуалната част на приложението, като предоставя графичен интерфейс (GUI) за взаимодействие с потребителя.

Библиотеки и методи: За разработката на GUI може да се използва Maven, JavaFX и други библиотеки.

Слой на бизнес логиката:

Описание: Този слой съдържа логиката на приложението, която обработва правилата на шаха, валидацията на ходовете, изчислението на възможните ходове и т.н.

Слой на достъп до данни:

Описание: Този слой се грижи за взаимодействието със съхраняваните данни, като потребителските профили и други. Тук могат да се използват бази данни за съхранение на данните.

Библиотеки и методи: За достъп до бази данни може да се използва JDBC (Java Database Connectivity). За работа с файлова система може да се използва стандартната библиотека на Java или специализирани библиотеки за работа с файлови формати.

## Организация и код на заявките към база от данни

В Java проекта за шах, когато трябва да се осъществи връзка с база данни, се използва JDBC (Java Database Connectivity). Това е интерфейс на Java, който позволява взаимодействието с различни релационни бази данни.

## Наличие и интуитивност на потребителски интерфейс (конзолен, графичен, уеб)

Използван е изцяло графичен потребителски интерфейс за реализацията на приложението за шах. Основните функционалности биват:

Шахматна дъска: Показване на дъска с 64 бели и черни квадрата, разположени в 8x8 матрица.

Избор на фигури: Възможност за избор на фигура, като например пешка, топ, кон, офицер, царица или крал.

Игрови опции: Възможност за настройка на различни игрови опции като игра срещу изкуствен интелект или човек на локално ниво, контрол, включване/изключване играта, и др.

Логин/Регистър форма: сравняване на имейл, име, парола и минали влизания от базата данни.

# Тестване

Тук се *включват тестовите случаи* и какви видове тестване предвиждате в реалното изпълнение на проекта, напр. с колко и какви документи, в какви браузъри, с какви приставки, и т.н.

# Заключение и възможно бъдещо развитие

Предимства за използването на Java за този проект:

Платформена независимост: Java е платформено независим език, който позволява приложението да работи на различни операционни системи.

Обширна библиотека: Java разполага с обширна библиотека, която предоставя множество инструменти и функционалности, които могат да бъдат използвани за разработка на шахматно приложение.

Недостатъци за използването на Java за този проект:  
По-сложен синтаксис: Синтаксисът на Java може да бъде по-сложен за начинаещи програмисти, което може да удължи и усложни разработката и поддръжката на приложението.

За бъдещото развитие на шахматното приложение може да се вземе предвид следните решения:

1. Подобрена визуализация и интерфейс: Развитието на по-добри и интуитивни графични интерфейси, които правят играта по-привлекателна и достъпна за потребителите.
2. Разширяване на функционалността на приложението: Добавяне на допълнителни материали за обучение, например видеоуроци, и разнообразни задачи за развитие на тактическите и стратегическите умения.
3. Изкуствен интелект и алгоритми: Подобрение на шахматните двигатели и алгоритми, които се използват за компютърна анализа и оценка на партиите. Развитието на по-силни и интелигентни шахматни програми, които предлагат по-предизвикателни и интересни игри.

# Използвани литературни източници и Уеб сайтове

1. Уеб сайт на GeekForGeeks, адрес https://www.geeksforgeeks.org/
2. Уеб сайт на Wikipedia адрес https://en.wikipedia.org/wiki/Chess
3. Литературен източник 1 - https://en.wikipedia.org/wiki/Chess

# Приложения

При необходимост можете да добавите и допълнителни секции под формата на апендикси. Таблица с диаграми, таблици и графики

*Забележка:*

1. *Документацията на проекта се предава само в електронен вид в MS Word, чрез качването на архив с документа и останалите файлове по проекта, в задание за предаване на проект, в канала на екипа в Teams.*
2. *Кода на проекта, базата данни и документацията трябва да са налични в репозитори в GitHub, което е копие на заданието генерирано в организацията.*

# Критерии и показатели за оценяване

|  |  |  |
| --- | --- | --- |
| **Показател** | **точки** | **срок** |
| 2. [Цели и обхват на софтуерното приложение](file:///O:/Admin204.1/Проекти/2021-2022/Ученически%20практики%202/Ученическа%20документация/user_name_Software%20Development.docx#_Цели_и_обхват)  3.1 [Потребителски изисквания и работен процес](file:///O:/Admin204.1/Проекти/2021-2022/Ученически%20практики%202/Ученическа%20документация/user_name_Software%20Development.docx#_Потребителски_изисквания_и) | 5  5 | 04.03.2022 |
| 3.2 [Примерен потребителски интерфейс](file:///O:/Admin204.1/Проекти/2021-2022/Ученически%20практики%202/Ученическа%20документация/user_name_Software%20Development.docx#_Примерен_потребителски_интерфейс)  3.3 [Диаграми на анализа](file:///O:/Admin204.1/Проекти/2021-2022/Ученически%20практики%202/Ученическа%20документация/user_name_Software%20Development.docx#_Диаграми_на_анализа)  3.4 [Модел на съдържанието/данните](file:///O:/Admin204.1/Проекти/2021-2022/Ученически%20практики%202/Ученическа%20документация/user_name_Software%20Development.docx#_Модел_на_съдържанието) | 5  5  5 | 18.03.2022 |
| 4.1 [Реализация на структура на приложението (3-layer),  Разделение на кода според предназначението му.  Допълване на Class диаграми/3.3/](file:///O:/Admin204.1/Проекти/2021-2022/Ученически%20практики%202/Ученическа%20документация/user_name_Software%20Development.docx#_Дизайн). | 10  10 | 16.04.2022 |
| 4.2 [Организация и код на заявките към база от данни](file:///O:/Admin204.1/Проекти/2021-2022/Ученически%20практики%202/Ученическа%20документация/user_name_Software%20Development.docx#_Дизайн). | 15 | 30.04.2022 |
| 4.3 [Наличие и интуитивност на потребителски интерфейс (конзолен, графичен, уеб)](file:///O:/Admin204.1/Проекти/2021-2022/Ученически%20практики%202/Ученическа%20документация/user_name_Software%20Development.docx#_Дизайн). | 10 | 31.05.2022 |
| 5. [Наличие и организация на автоматизирани тестове](file:///O:/Admin204.1/Проекти/2021-2022/Ученически%20практики%202/Ученическа%20документация/user_name_Software%20Development.docx#_Тестване). | 15 | 15.06.2022 |
| 6. Организация на проекта в система за контрол на изходния код и употреба на добри практики (merge requests, code reviews, branching strategy) | 10 | 25.06.2022 |
| 1. [Въведение. Ниво на завършеност на проекта](file:///O:/Admin204.1/Проекти/2021-2022/Ученически%20практики%202/Ученическа%20документация/user_name_Software%20Development.docx#_Въведение) |  | 30.06.2022 |
| Документация на проекта (XML comments, wiki, etc.) | 5 | текущо |
| Презентация на проекта |  | 30.06.2022 |
| Общо | 100 | Финал на първа фаза |