

# ООП Проекти

Всички проекти са индивидуални. Трябва да изберете един проект. Проекта ви трябва да бъде в гитхъб. Там ще пушвате по него и накрая ще можем да проследим кой колко е работил, гледайки броя къмита на всеки. Всички проекти трябва да използват принципите на ООП, да имат класове, пакети, интерфейси. Проекта ви не трябва да може да бъде счупен от обикновения потребител на вашето приложение - трябва да има всички необходими валидации на данните. Към проектите трябва да има юнит тестове, покриващи поне 70% от кода.

## Проект, измислен от вас

Ако не си харесате проект от изброените по-долу или имате оригинална идея, която си заслужава да бъде реализирана, може да изберете да работите по собствен проект. За да работите по проект измислен от вас, го опишете възможно по-подробно и ни го пратете. Едва след като потвърдим, че е на нужното ниво, може да започнете работа по него. Проекти, за които не сте ни уведомили, няма да бъдат приемани.

Всяко описание на проект трябва да съдържа:

- Резюме на идеята
- Цел на проекта
- Целева група
- Списък с различните функции на програмата

## Шах с двама играчи

Шахът е древна и все още популярна игра. Реализирайте програма, която позволява на двама играчи, редувайки се, да играят шах.

- Двамата играчи се контролират фигурите от мишката на компютъра.
- За да се селектира фигура се клика върху нея, следващия клик указва къде да отиде.
- Трябва да се спазват всички правила на играта шах.
- Таймер отброява кой играч колко време общо е играл.
- Играта може да се запише във файл и при следващо стартиране да се зареди от файла и да се продължи до където е стигнала.
- Бонус: Играта може да се играе от един играч.

# Прогноза за времето

Програма за визуализиране на данни за времето - в миналото, в момента и в бъдещето. Основната функция на програмата е да показва текущото време, но потребителя може да намери и подробна информация за други периоди.

Програмата трябва да може:

- Не е нужно да има графичен интерфейс.
- Да зарежда информацията си от текстови файл. Ние ще ви предоставим текстовият файл.
- При стартиране програмата показва времето в момента, времето за следващия период от време, времето за утре, за вдругиден и за вчера.

Пример:

-----  
Днес: пн. 27.04.2020, 13:44

Температура: 22

Валежи: 14%

-----  
Довечера: 20 градуса, 12% валежи

Утре: 24/11 градуса, 33% валежи

29.04.2020: 22/10 градуса, 3% валежи

-----  
Вчера: 20/8, 11% валежи

-----  
За подробна прогноза за утре: 1

За 10-дневна прогноза: 2

За 10 дни назад: 3

- Потребителят има три опции
- Ако избере 1, ще види прогнозата за утре в 7, 12, 18 и 22 часа.
- Ако избере 2, ще види прогнозата за 10 дни в бъдещето, като ще има по една усреднена стойност за ден (както е на примера горе за утре)
- Ако избере 3, ще види 10 дни назад времето, като отново ще има по една стойност за ден
- Информацията ще бъде предоставена в текстови файл weather.csv, който може да намерите ТУК. Форматът на данните ще е следният:
  - Всеки нов ред е отделна информационна единица за определен период от време

- На реда, отделните данни са разделени с точка и запетая
- На всеки ред има дата, час, максимална температура, минимална температура, процент вероятност за валеж

## Система за запазване на билети в кино салон

Всяко модерно кино трябва да има дигитална система за резервиране на билети. Така се улесняват и клиентите, и персонала на киното.

В системата трябва да може:

- Да се въвежда филм, времена и дати на прожекциите.
- Данните да се запомнят и след изключване на програмата.
- Да има различни видове потребители: Администратори, които могат да въвеждат филми, зали (с различни размери), цени и Потребители, които да си купуват билети и история на поръчките.
- Да се купуват билети:
  - Избира се филма от падащо меню, съдържащо само филмите, които не са започнали все още за днешния ден
  - Избира се брой места
  - Отваря се екран със схема със свободните и заетите места за съответната прожекция.
  - Генерира се билет - текстов файл, който съдържа билетите: името на филма, прожекцията, дата и време, редове и места