

**ДВАДЕСЕТ И ПЕТА УЧЕНИЧЕСКА СЕКЦИЯ**

**на Пролетната конференция на СМБ**

**УС'25**

**31 март – 3 април 2025 г., Варна**

**ТЕМА НА ПРОЕКТА**

**Образователна платформа за обучение по програмиране**

**Автор(и):**

**Георги Иванов Шишков, ПГЕЕ - Пловдив, 12 клас**

(трите имена, училище, град, клас)

**Научен ръководител (консултант):**

**Магдалена Мъглижанова, учител по теоретично обучение,  
ПГЕЕ - Пловдив**

(име, фамилия, длъжност, месторабота)

# СЪДЪРЖАНИЕ

<b>СЪДЪРЖАНИЕ.....</b>	<b>2</b>
<b>РЕЗЮМЕ.....</b>	<b>3</b>
<b>УВОД.....</b>	<b>3</b>
<b>ГЛАВА 1 – ОСНОВНИ ЕТАПИ В РЕАЛИЗИРАНЕТО НА ПРОЕКТА.....</b>	<b>5</b>
1.1.Среда за разработка на приложението.....	5
1.2. Програмиране и уеб технологии.....	5
1.3. Контейнеризация чрез Podman.....	6
1.4. ChatBot.....	6
<b>ГЛАВА 2 – РЕАЛИЗАЦИЯ НА ПРОЕКТА.....</b>	<b>7</b>
2.1. Персонализирани функционалности.....	7
2.2. Модерен подход към обучението.....	7
2.3. Необходими библиотеки.....	7
2.4. Файлова структура.....	8
2.5. Хостинг и домейн на сайта – Smartasp.....	8
2.6. Домейн на сайта.....	9
2.7. База данни.....	9
3.1. Структура на съдържанието на приложението.....	10
3.2. Бъдещите състезания по програмиране.....	12
<b>ГЛАВА 4 – ЗАКЛЮЧЕНИЕ.....</b>	<b>13</b>
<b>ГЛАВА 5 - ИЗПОЛЗВАНИ ИЗТОЧНИЦИ.....</b>	<b>14</b>

## РЕЗЮМЕ

Платформата е иновативно приложение, което съчетава най-новите технологии с модерни образователни методи. Основната ѝ цел е да подпомага ученици и преподаватели в усвояването на ключови умения, развитието на креативност и аналитично мислене, както и разбирането на сложността на съвременния свят. Освен това, ще насърчава творческото мислене, съвместната работа и критичното анализиране на информация, като подготвя младите хора за успешна реализация в съвременната дигитална среда. Това приложение ще помага за обмен на знания между преподаватели и ученици, насърчавайки изграждането на образователна общност, ориентирана към бъдещата професия "Програмист".

## УВОД

Светът се променя. По-бързо, отколкото сме си представяли, и често в неочаквани посоки. Дигитализацията, автоматизацията и изкуственият интелект преобразяват не само начина, по който живеем, но и как работим. **Професиите на бъдещето** изискват умения, които досега бяха рядкост или изобщо не съществуваха.

Особено видима е промяната при програмистите. От специалисти, които трябва да разбират от алгоритми, те се превръщат в архитекти на дигиталния свят. Сега работата им е свързана с огромни масиви от данни, създаване и управление на изкуствени интелекти и други сложни технологии. Тези изисквания водят до необходимостта от постоянен процес на учене и адаптация към иновациите, които бързо изместват традиционните подходи.

Иновациите създават предизвикателства за професиите и налагат нови подходи към образованието. Ученици и преподаватели трябва да се подготвят за тези промени и да се възползват от възможността да развият нови умения, които ще ги направят конкурентоспособни в бъдещият свят.

**Цел на сайта:**

Създаването на платформата е продиктувано от нуждата от модерно образователно пространство, което интегрира най-новите технологии и предоставя възможности за развитие. Тя е предназначена да подпомага ученици и преподаватели в усвояването на ключови умения, развиването на креативност и аналитично мислене, както и разбирането на сложността на съвременния свят.

**Целева група:**

Платформата е насочена към ученици и преподаватели, които искат да се адаптират към технологиите на бъдещето. Тя предоставя инструменти за овладяване на умения, необходими за новите професии, и насърчава иновативни подходи към ученето и преподаването.

**Съдържание на сайта:**

- Интерактивни модули с обратна връзка в реално време: За по-добро усвояване на уроците;
- Обучение в модерни технологии и практики;
- Изкуствен интелект и машинно обучение: Основни концепции и практики за работа с AI и ML модели.
- Работа с големи данни (Big Data);
- Възможност за създаване на автоматизирани тестове и интерактивни задачи за улеснение на ученици и учители

**Резултат от създаването на сайта:**

Резултатът от създаването на платформата ще бъде от голямо значение както за учениците, така и за преподавателите в тяхната подготовка. Тя ще предоставя възможности за постоянно развитие на нови умения и ще подкрепя творческите и интелектуалните способности на тези, които я използват. Ще улесни учителите при добавянето на материали, тестове и упражнения. Също така ще помогне значително при проверяването им чрез автоматизирани тестове за програмните задачи, спестявайки на учителя времето, което се отделя за сваляне на архиви с решения и ръчното им проверяване.

# ГЛАВА 1 – ОСНОВНИ ЕТАПИ В РЕАЛИЗИРАНЕТО НА ПРОЕКТА

## 1.1.Среда за разработка на приложението.

За изграждането на този проект основно са използвани **ASP.NET** и езикът за програмиране **C#**.

**ASP.NET** е популярна платформа за разработка на уеб приложения, създадена от **Microsoft**. Тя предлага вградени функции за защита на данни, като автентикация и авторизация, които са от съществено значение за осигуряване на сигурността, особено ако платформата ще обработва чувствителна информация за потребителите. Освен това, **ASP.NET** е лесно съвместим с различни технологии и бази данни, като предоставя възможности за обработка на големи обеми данни, интеграция с изкуствен интелект (ИИ) и други иновации.

## 1.2. Програмиране и уеб технологии.

**C#**, от своя страна, е основният език за програмиране, използван с **ASP.NET**. Той е силно типизиран език, който предлага богата функционалност за обектно-ориентирано програмиране и е широко използван при разработката на уеб приложения. **C#** се отличава с ясен и изчистен синтаксис, който улеснява както разработката, така и поддръжката на кода. Освен това, **C#** предлага отлична производителност при изграждането на големи и сложни приложения, подходящ е за работа с бази данни и различни услуги, което го прави идеален за платформи, които изискват обработка на големи обеми данни и интеграция с различни източници на информация.

**HTML (HyperText Markup Language)** – Основният език за изграждане на уеб страници. Този език определя структурата на съдържанието в уеб сайтовете и работи заедно с **CSS** и **JavaScript** за създаване на динамични и интерактивни уеб приложения.

**JavaScript** – Скриптов език, който позволява динамичното взаимодействие на потребителите с уеб страниците. Използва се за разработка на интерактивни уеб приложения, а в комбинация с **HTML** и **CSS** позволява създаването на модерни уеб сайтове.

### 1.3. Контейнеризация чрез Podman.

Чрез следните технологии в платформата се създава виртуална среда за разработка на потребителите:

- 1) **Docker** – Водеща платформа за контейнеризация, която позволява разработчиците да изграждат, пакетират и разпространяват приложения в изолирани среди. Docker улеснява процесите на разработка, тестване и внедряване на софтуер в различни среди, като елиминира проблемите с несъвместимости.
- 2) **Podman** – Модерен инструмент за управление на контейнери, който осигурява сигурна алтернатива на Docker. Поддържа rootless контейнери, което повишава сигурността, и позволява създаването и управлението на OCI - съвместими контейнери без нужда от демон процес.

### 1.4. ChatBot

В приложението е добавен чат бот, който помага на потребителя при решаване на задачите. Информацията че бъде ограничена, за да не улеснява прекалено много работата на ученика. За целта е специализиран модел.

Проектът използва SignalR за комуникация в реално време между клиента и сървъра, Gemini API за генериране на отговори от AI и ASP.NET Core за изграждане на уеб приложението. Дизайнът на чата е реализиран с HTML, CSS и JavaScript, като JavaScript обработва изпращането и получаването на съобщения чрез SignalR.

Чат ботът работи като посредник между потребителя и ИИ. Когато някой изпрати съобщение, то се препраща към сървъра. След като получи отговор, сървърът го връща обратно в чата, където се визуализира в реално време, без нужда от презареждане на страницата. Това позволява плавна и интерактивна комуникация.

## ГЛАВА 2 – РЕАЛИЗАЦИЯ НА ПРОЕКТА

Съществуващите платформи като Coursera и Codecademy предлагат различни компоненти на обучението, но настоящият проект обединява всички тези елементи и предоставя уникални възможности. Той съчетава най-добрите практики и нови технологии, като интеграция с ИИ, осигуряване на практическо обучение чрез работа с реални данни.

**Тази платформа е уникална, защото комбинира персонализирани компоненти и нови технологии по няколко начина:**

### 2.1. Персонализирани функционалности.

Учителите имат възможност да създават свои собствени курсове, адаптирани към нуждите и интересите на учениците (в случая от специалност „Приложно програмиране“), да проследяват напредъка и да оценяват според различни компетентности.

Например: дигитални компетентности, инициативност и предприемчивост и др.

### 2.2. Модерен подход към обучението.

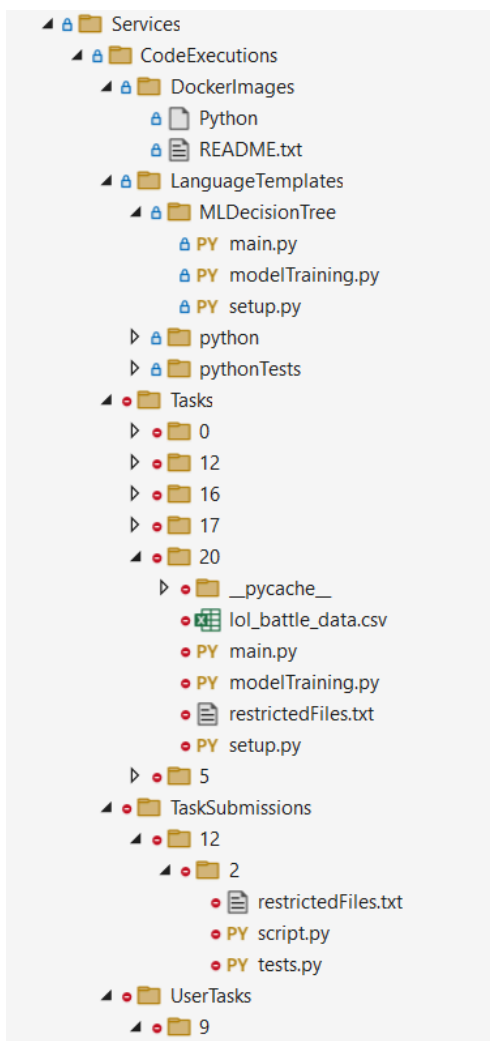
Интеграция на **ИИ асистент**, който помага, проверява кода, генерира предложения за подобрения, помага за проследяване на напредъка и персонализиране на задачите спрямо нуждите на потребителите.

### 2.3. Необходими библиотеки.

Всички библиотеки биват свалени в **Docker images**, с цел подобряване на ефикасността и избягване на ненужно инсталиране. Освен това предварителното дефиниране на библиотеките с и със **съображения за сигурност**. По този начин потребителите не биха могли умишлено или по грешка да инсталират нелицензиран или опасен софтуер.

Екзекутирането на потребителският код се извършва във **виртуална среда** с помощта на **Podman**. Всяка екзекуция заема място в отделен **виртуален контейнер**. По този начин се осигурява **безопасна среда**. Също така всеки **контейнер**, както и всеки **сървър** имат ограничения на **брой ядра, памет, време** и други.

## 2.4. Файлова структура.



(фиг. 1 - Файлова структура)

В папката **Code Executions** се намират всички файлове свързани с компилирането и екзекутирането на потребителския код.

Папката **Docker Images** съдържа моделите за различните docker изображения. В тях се задават всички библиотеки за даденият език.

**Language Templates** съдържа шаблоните достъпни до учителите при създаването на задачи. Когато създават нова задача те избират един от тях и пишат върху него.

В **Tasks** се записват всички задачи след като бъдат зададени. Авторът им може да продължава да ги редактира.

**User Tasks** е директорията в която всеки потребител си създава копие на задачите по които работи.

При коректно решение на задача ученика може да предаде своето решение което ще се озове в **TaskSubmissions**.

## 2.5. Хостинг и домейн на сайта – Smartasp

**SmartASP** предлага хостване на уеб приложения и платформи като настоящата разработка.

**SmartASP** е оптимизирана платформа за работа с технологии - ASP.NET. Това гарантира бързо и ефективно изпълнение на уеб приложенията.



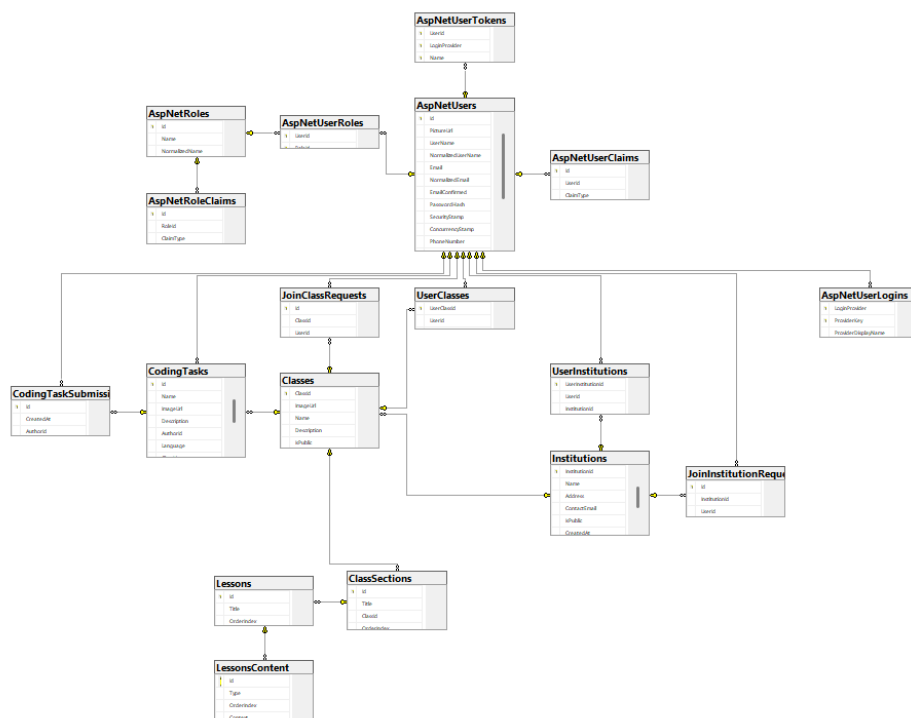
Платформата предлага хостинг услуги, които могат да поддържат интензивни процеси, като например интеграция на AI технологии и машинно обучение, които са част от функционалността на платформата.

Хостинг решенията на SmartASP включват различни механизми за защита на данните, като SSL сертификати, защитени мрежи и автентикация на потребителите, което е изключително важно за настоящата платформа, тъй като ще обработва чувствителна информация за потребителите и техните проекти.

## 2.6. Домейн на сайта.

SmartASP предлага лесен начин за регистрация на домейн и управление на DNS настройки.

## 2.7. База данни.



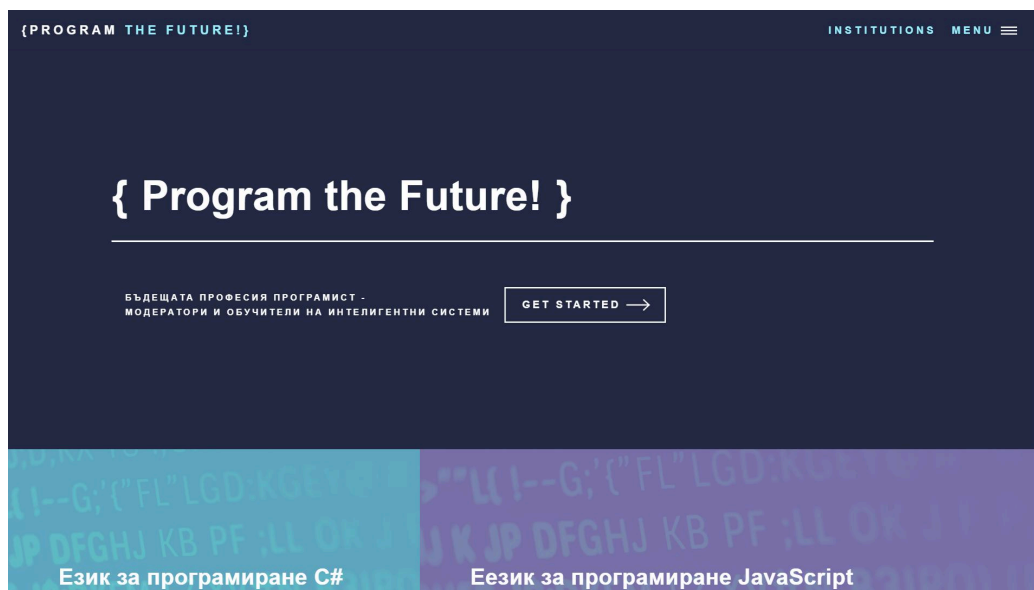
(фиг. 2 - База данни на приложението)

За база от данни се използва SQL Server. Това е релационна база от данни, което гарантира интегритет на данните. В базата има таблици за потребители, институции, класове, задачи, уроци, изпратени решения и др. Таблиците свързани със задачите биват асоциирани със съответните им файлове чрез id-то им.

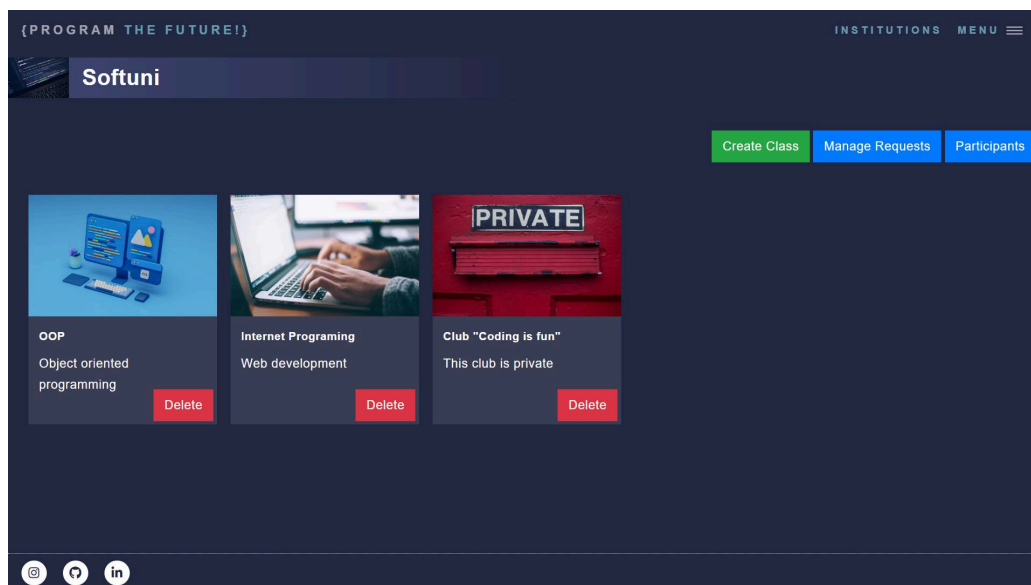
## ГЛАВА 3 - ЛОГИЧЕСКО И ФУНКЦИОНАЛНО ОПИСАНИЕ НА РЕШЕНИЕТО

### 3.1. Структура на съдържанието на приложението.

→ Страници



(фиг. 3 -Начална страница)



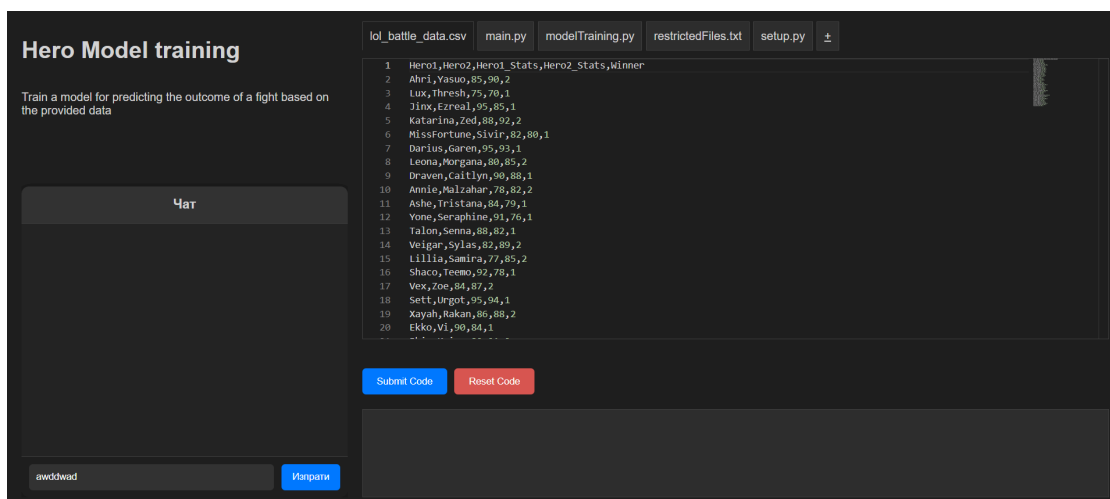
(фиг. 4 - Създаване на класове)

Чрез специално създадени страници учителите ще могат да:

- Добавят и управляват задания за учениците
- Добавят ученици към частни класове
- Добавят ученици към частни институции
- Одобряват или отхвърлят заявки за присъединяване на ученици към институция
- Качват и организират уроци
- Преглеждат и оценяват изпълнените задачи
- Следят напредъка на учениците чрез статистики и анализи
- Създават и управляват класове

Учениците ще могат да:

- Изпращат заявки за присъединяване към различните публични институции
- Достъпват качените уроци и учебни материали
- Изпълняват и предават задания
- Получавам обратна връзка и оценки
- Участват в интерактивни модули и практически проекти
- При нужда получава помощ от вграденият в платформата чат бот



(фиг. 5 - Среда за програмиране)

Учителят има възможност да създава класове, като при частните класове се генерира код за достъп. Той може да добавя уроци, да включва други колеги към даден клас и да добавя ученици от списъка на институцията. Освен това, учителят може да оценява учениците и да създава задания, които да бъдат изпълнявани от тях.

Учениците имат достъп до списък с публичните институции и могат да изпращат заявки за присъединяване. След като учителят одобри заявката, ученикът получава достъп до уроците, заданията и може да изпраща своите решения за оценка.

Задачите по програмиране се създават от учители и администратори. При създаването им учителят определя име, записва описание и избира програмен език и шаблон. След като заданието бъде създадено, авторът му има възможност да редактира създадените файлове. Всеки шаблон съдържа специален файл с име `restricted.txt`, който е достъпен само за автора. В този файл чрез запетайки могат да бъдат записани имената на файлове, които трябва да останат скрити от останалите потребители, като например файловете с тестове, предназначени за автоматична проверка на решенията.

### **3.2. Бъдещите състезания по програмиране**

Състезателното програмиране дълго време беше свързано с демонстрация на умения в основни алгоритмични и математически понятия. Но с развитието на изкуствения интелект (ИИ) и машинното обучение, традиционните състезания по програмиране може да се променят. В бъдеще успехът няма да се измерва само в способността да се решават дадени задачи, а в умението да се разработват, настройват и подобряват модели на машинно обучение.

## ГЛАВА 4 – ЗАКЛЮЧЕНИЕ

1. Традиционните алгоритмични задачи ще отстъпят място на моделирането на данни – способността да се изграждат и подобряват модели ще бъде основно умение.
2. Разбирането на машинното обучение дава конкурентно предимство – бъдещите състезатели няма да решават само задачи, но и ще намират оптимални подходи за анализ на данни.
3. Индустриалната значимост на ИИ расте – компании предпочитат специалисти с умения за работа с машинно обучение.
4. Състезателите развиват аналитично и експериментално мислене – настройването на хипер параметри и анализът на резултати става основно умение.

Това не означава, че класическото програмиране ще изчезне, но че ще бъде разширено с машинно обучение, което ще стане неразделна част от бъдещите състезания.

За тази цел обаче, досегашните методи за обучение няма да са достатъчни. С помощта на съвременни платформи, съдържащи интерактивни упражнения и задачи, тази цел става по-постижима. Една такава платформа помага както на ученика да разбере материала, чрез практическа работа. Така и спестява време и помага на учителя при създаване на материали и оценяване на учениците.

## ГЛАВА 5 - ИЗПОЛЗВАНИ ИЗТОЧНИЦИ

- Gantikow, H., Walter, S., & Reich, C. (2020, June). Rootless containers with Podman for HPC. In *International Conference on High Performance Computing* (pp. 343-354). Cham: Springer International Publishing.
- Anand, Yuvanesh, et al. "Gpt4all: Training an assistant-style chatbot with large scale data distillation from gpt-3.5-turbo." GitHub <https://github.com/nomic-ai/gpt4all> (2023).
- Price, Mark J. *C# 8.0 and .NET Core 3.0—Modern Cross-Platform Development: Build applications with C#, .NET Core, Entity Framework Core, ASP.NET Core, and ML.NET using Visual Studio Code*. Packt Publishing Ltd, 2019.
- Mukherjee, Sourav. "SQL Server Development Best Practices." *International Journal of Innovative Research in Computer and Communication Engineering* 10.2 (2019): 27-35.