

ҚАЗАҚСТАН РЕСПУБЛИКАСЫ БІЛІМ ЖӘНЕ ҒЫЛЫМ МИНИСТРЛІГІ

Сәтбаев университеті

Кибернетика және ақпараттық технологиялар институты

Киберқауіпсіздік, ақпаратты өңдеу және сақтау кафедрасы

Болатов Мәдияр Ерланұлы

«ИТ технологияларды қазақ тілінде үйрететін веб-ресурс жобалау»

ДИПЛОМДЫҚ ЖҰМЫС

5B070300—«Ақпараттық жүйелер» мамандығы

Алматы 2020

ҚАЗАҚСТАН РЕСПУБЛИКАСЫ БІЛІМ ЖӘНЕ ҒЫЛЫМ МИНИСТРЛІГІ

Сәтбаев университеті

Кибернетика және ақпараттық технологиялар институты

Киберқауіпсіздік, ақпаратты өңдеу және сақтау кафедрасы

ҚОРҒАУҒА ЖІБЕРІЛДІ
КАӨЖС кафедра меңгерушісі,
тех.ғыл.канд., ассоц.
профессор
_____ Н.А.Сейлова
« _____ » _____ 2020ж.

ДИПЛОМДЫҚ ЖҰМЫС

Тақырыбы: «ИТ технологияларды қазақ тілінде үйрететін веб-ресурс жобалау»

5B070300 – «Ақпараттық жүйелер» мамандығы бойынша

Орындаған:
Пікір беруші:

« _____ » _____ 2020 ж.

Болатов М.Е.
Ғылыми жетекші:
_____ Таубалды Н.
« _____ » _____ 2020 ж.

Алматы 2020

ҚАЗАҚСТАН РЕСПУБЛИКАСЫ БІЛІМ ЖӘНЕ ҒЫЛЫМ МИНИСТРЛІГІ

Сәтбаев университеті

Кибернетика және ақпараттық технологиялар институты

Киберқауіпсіздік, ақпаратты өңдеу және сақтау кафедрасы

5B070300 – «Ақпараттық жүйелер» мамандығы

БЕКІТЕМІН

КАӨЖС кафедра меңгерушісі,
тех.ғыл.канд., ассоц.
профессор

_____ Н.А.Сейлова
«_____» _____ 2020 ж.

**Дипломдық жұмысты орындауға
ТАПСЫРМА**

Білім алушы: Болатов Мәдияр Ерланұлы

Тақырыбы: «ИТ технологияларды қазақ тілінде үйрететін веб-ресурс жобалау»

Университет Ректоры: 2020 жылғы «27» қаңтар №762-б бұйрығымен бекітілген

Аяқталған жұмысты тапсыру мерзімі: 2020 жылы «__» мамыр

Дипломдық жұмыстың бастапқы берілістері: диплом алдындағы практикалық жұмыс қорытындысы, тақырып бойынша әдебиеттерге шолу нәтижелері, теориялық мәліметтердің жиыны

Дипломдық жұмыста қарастырылатын мәселелердің тізімі:

а) қойылған мәселенің қазіргі жағдайын пайымдау

б) ақпараттық қамтаманы құру

Сызбалық материалдар тізімі: Power Point бағдарламасындағы слайдтар

Сызба материалдар: _____ слайдпен көрсетілген

Ұсынылатын негізгі әдебиет: 11 атау

Протокол анализа Отчета подобия

заведующего кафедрой / начальника структурного подразделения

Заведующий кафедрой / начальник структурного подразделения заявляет, что ознакомился(-ась) с Полным отчетом подобия, который был сгенерирован Системой выявления и предотвращения плагиата в отношении работы:

Автор: Болатов Мадияр

Название: Разработка Web ресурса обучающий ИТ технологиям на казахском языке

Координатор: Нұрсұлтан Таубалды

Коэффициент подобия 1:2,9

Коэффициент подобия 2:1,3

Замена букв:2

Интервалы:0

Микропробелы:1

Белые знаки:0

После анализа отчета подобия заведующий кафедрой / начальник структурного подразделения констатирует следующее:

- обнаруженные в работе заимствования являются добросовестными и не обладают признаками плагиата. В связи с чем, работа признается самостоятельной и допускается к защите;
- обнаруженные в работе заимствования не обладают признаками плагиата, но их чрезмерное количество вызывает сомнения в отношении ценности работы по существу и отсутствием самостоятельности ее автора. В связи с чем, работа должна быть вновь отредактирована с целью ограничения заимствований;
- обнаруженные в работе заимствования являются недобросовестными и обладают признаками плагиата, или в ней содержатся преднамеренные искажения текста, указывающие на попытки сокрытия недобросовестных заимствований. В связи с чем, работа не допускается к защите.

Обоснование:

.....
.....
.....
.....
.....
.....
.....

.....

.....

Дата

Подпись заведующего кафедрой /

начальника структурного подразделения

Окончательное решение в отношении допуска к защите, включая обоснование:

.....
.....
.....
.....
.....
.....
.....

.....

.....

Дата

Подпись заведующего кафедрой /

начальника структурного подразделения

Протокол анализа Отчета подобия Научным руководителем

Заявляю, что я ознакомился(-ась) с Полным отчетом подобия, который был сгенерирован Системой выявления и предотвращения плагиата в отношении работы:

Автор: Болатов Мадияр

Название: Разработка Web ресурса обучающий ИТ технологиям на казахском языке

Координатор: Нұрсұлтан Таубалды

Коэффициент подобия 1:2,9

Коэффициент подобия 2:1,3

Замена букв:2

Интервалы:0

Микропробелы:1

Белые знаки: 0

После анализа Отчета подобия констатирую следующее:

- обнаруженные в работе заимствования являются добросовестными и не обладают признаками плагиата. В связи с чем, признаю работу самостоятельной и допускаю ее к защите;
- обнаруженные в работе заимствования не обладают признаками плагиата, но их чрезмерное количество вызывает сомнения в отношении ценности работы по существу и отсутствием самостоятельности ее автора. В связи с чем, работа должна быть вновь отредактирована с целью ограничения заимствований;
- обнаруженные в работе заимствования являются недобросовестными и обладают признаками плагиата, или в ней содержатся преднамеренные искажения текста, указывающие на попытки сокрытия недобросовестных заимствований. В связи с чем, не допускаю работу к защите.

Обоснование:

.....
.....

.....

Дата

.....

Подпись Научного руководителя

REPORT DATE: 2020-05-26 11:32:03

TITLE:
 Разработка Web ресурса обучающий ИТ технологиям на казахском языке

AUTHOR(S):
 Болатов Мадияр

PROMOTER:
 Нұрсұлтан Таубалды

ORGANIZATIONAL UNIT:
 ИКИИТ

FILE UPLOAD DATE:
 2020-05-26 11:25:58

NUMBER OF VERIFICATIONS: 
 1

SKIPPED URL ADDRESSES: 

Record of similarities

Please note that high coefficient values do not automatically mean plagiarism. The report must be analyzed by an authorized person.



Alerts

In this section, you can find information regarding text modifications that may aim at temper with the analysis results. Invisible to the person evaluating the content of the document on a printout or in a file, they influence the phrases compared during text analysis (by causing intended misspellings) to conceal borrowings as well as to falsify values in the Similarity Report. It should be assessed whether the modifications are intentional or not.

Characters from another alphabet number of characters from other alphabets which may imitate letters from the alphabet relevant to the document, causing misspellings in the text, please verify their validity	2	show in the text
Spreads number of increased distances between letters - please verify whether they imitate spaces, indicating as merging the words in the Report	0	show in the text
Micro spaces number of spaces with zero length - please verify whether they are placed inside words and cause word division in the text	1	show in the text
White characters number of characters with a white font color - please verify whether they are used instead of spaces, causing merge of the words (in the Report the color of the letters is changed to black in order to show them)	0	show in the text

Active lists of similarities

Scroll the list and analyze especially the fragments that exceed the SC 2 (marked in bold). Use the link "Mark fragment" and see if they are short phrases scattered in the document (coincidental similarities), numerous short phrases near each other (mosaic plagiarism) or extensive fragments without indicating the source (direct plagiarism).

The 10 longest fragments (2,94 %)

Ten longest fragments found in all available resources.

NO	TITLE OR SOURCE URL (DATABASE)	AUTHOR(S)	NUMBER OF IDENTICAL WORDS	
1	https://stud.kz/referat/show/67031		33	0,76 %
2	https://stud.kz/referat/show/67032		25	0,58 %
3	https://b-ok.org/book/4984953/e3b221		15	0,35 %
4	https://stud.kz/referat/show/67031		11	0,25 %
5	https://stud.kz/referat/show/67031		10	0,23 %
6	https://stud.kz/referat/show/67032		10	0,23 %
7	https://stud.kz/referat/show/67032		10	0,23 %
8	https://stud.kz/referat/show/67031		8	0,19 %
9	https://b-ok.org/book/4984953/e3b221		5	0,12 %

from RefBooks database (0,00 %)

All fragments found in RefBooks database, which contains over 3 millions academic text including scientific articles published in popular databases.

NO	TITLE	AUTHOR(S)	NUMBER OF IDENTICAL WORDS (NUMBER OF FRAGMENTS)
----	-------	-----------	---

NO SIMILARITIES FOUND

from the home database (0,00 %)

All fragments found in the database of your institution.

NO	TITLE	AUTHOR(S)	INDEXING DATE	IDENTICAL WORDS (FRAGMENTS)
----	-------	-----------	---------------	-----------------------------

NO SIMILARITIES FOUND

from the Database Exchange Program (0,00 %)

All fragments found in the database of other institutions.

NO	TITLE DATABASE NAME	AUTHOR(S)	INDEXING DATE	NUMBER OF IDENTICAL WORDS (NUMBER OF FRAGMENTS)
----	------------------------	-----------	---------------	---

NO SIMILARITIES FOUND

from the Internet (2,94 %)

All fragments found in the open access global internet resources.

NO	SOURCE URL	IDENTICAL WORDS (FRAGMENTS)	
1	https://stud.kz/referat/show/67031	62 (4)	1,43 %
2	https://stud.kz/referat/show/67032	45 (3)	1,04 %
3	https://b-ok.org/book/4984953/e3b221	20 (2)	0,46 %

Дипломдық жұмысты дайындау

КЕСТЕСІ

Бөлімдер атауы, қарастырылатын мәселелер тізімі	Ғылыми жетекші мен кеңесшілерге көрсету мерзімдері	Ескерту
Мәселенің қазіргі жағдайына шолу және оны талдау		
Ақпараттық қамтаманы құру		
Программалық қамтаманы құру		

Дипломдық жұмысының бөлімдерінің кеңесшілері мен норма бақылаушыларының аяқталған жобаға қойған **қолтаңбалары**

Бөлімдер атауы	Кеңесшілер, аты, әкесінің аты, тегі (ғылыми дәрежесі, атағы)	Қол қойылған күні	Қолы
Норма бақылаушы	Бауыржан М. Б., тьютор		
Программалық қамтама	Кабдуллин М.А., ассистент		

Ғылыми жетекші _____ Таубалды Н.

Тапсырманы орындауға алған білім алушы _____ Болатов М.Е.

Күні

«__» мамыр 2020 ж.

АНДАТПА

Бұл дипломдық жұмыста қазақ тілінде ИТ технологияларды үйренуге арналған веб платформа ұсынылған. Қазіргі уақытта технологиялар қарқынды дамып, жылдан жылға білім ескіріп жатыр. ИТ компаниялардың басым көпшілігі батыс елдерінде болғандықтан, жаңа технологияларға байланысты курстар, мақалалар, кітаптар және басқа да білім көздері ағылшын тілінде ұсынылады. Ал қазақ тіліне бұл мәліметтердің көбі аударылмайды. Сол себепті біздің елде жаңа мамандарды дайындау, басқа елдермен бәсекелес болу қиынға соғады. Жақсы маман болу үшін, біздің елде қазақ тілінен басқа ағылшын тілін білу өте қажет. Ал басқа тілдерді үйренуге тағы да артық уақыт кетеді. Соның кесірінен қарапайым студент немесе оқушы жақсы ИТ маман болу үшін басқа елдің адамдарына қарағанда көбірек уақыт жұмсайды.

Бұл мәселені шешу үшін бұл дипломдық тақырыпта ұсынылған веб платформаны жобалау керек. Бұл платформада заманауи технологиялар бойынша қазақ тілінде курстар, қазақша кітаптар және соңғы жаңалықтар мен мақалалар ұсынылады. Оған қоса бұл жерде ИТ саласымен қызығатын басқа адамдармен байланысып, танысуға немесе қиындықтар туындағанда сұрақ қоюға болады.

АННОТАЦИЯ

В этой дипломной работе рассматривается разработка веб-платформы для обучения IT-технологиям на казахском языке. В настоящее время технологии развиваются быстрыми темпами, и знания из года в год устаревают. Поскольку подавляющее большинство IT-компаний находятся в западных странах, курсы, статьи, книги и другие источники знаний, связанных с новыми технологиями, предлагаются именно на английском языке. Большая часть этой информации не переведена на казахский язык. Поэтому в нашей стране сложно готовить новых специалистов и конкурировать с другими странами. Чтобы стать хорошим специалистом в нашей стране, помимо казахского языка необходимо знать и английский язык. Изучение других языков занимает еще больше времени. В результате, средний студент тратит больше времени, чем студенты в других странах, чтобы стать хорошим IT-специалистом.

Чтобы решить эту проблему, необходимо разработать веб-платформу, которая представлена в этой дипломной работе. Эта платформа предлагает курсы по современным технологиям на казахском языке, казахские книги и последние новости и статьи. Кроме того, здесь вы можете связаться с другими людьми, интересующимися IT-индустрией или задать вопросы в случае затруднений.

SUMMARY

This thesis deals with the development of a web platform for teaching IT technologies in the Kazakh language. Currently, technology is developing rapidly, and knowledge is becoming obsolete from year to year. Since the vast majority of IT companies are located in Western countries, courses, articles, books and other sources of knowledge related to new technologies are offered in English. Most of this information has not been translated into Kazakh. Therefore, it is difficult in our country to train new specialists and compete with other countries. To become a good specialist in our country, in addition to the Kazakh language, you need to know English. Learning other languages takes even longer. As a result, the average student spends more time than students in other countries to become a good IT specialist.

To solve this problem, it is necessary to develop a web platform, which is presented in this thesis. This platform offers courses on modern technologies in the Kazakh language, Kazakh books and the latest news and articles. In addition, here you can contact other people interested in the IT industry or ask questions in case of difficulty.

МАЗМҰНЫ

КІРІСПЕ	8
1 Аналитикалық бөлім	9
1.1 Есептің қойылымы	9
1.2 Ақпараттық жүйені жобалау құралын негіздеу	9
1.3 Қолданылған бағдарламалық тілдер	10
1.4 Қолданылған жаңа технологиялар	12
2 Жобалау бөлімі	14
2.1 Деректер базасын концептуалды жобалау	14
2.2 Веб-сайтты концептуалды жобалау	15
2.3 Веб-сайтты қолдану	17
3 Жобаның экономикалық тиімділігін негіздеу	29
3.1 Бағдарламаны енгізудің экономикалық тиімділігін есептеу	29
ҚОРЫТЫНДЫ	31
ПАЙДАЛАНЫЛҒАН ӘДЕБИЕТТЕР ТІЗІМІ	32
Қосымша А	33

КІРІСПЕ

Жылдан жылға білім беру саласында IT технологиялардың қолданылуы артып келе жатыр. Қазіргі уақытта көптеген адамдар өз білімдерін интернет арқылы жетілдіруде. Білім беру центрлеріне, курстарға барудың орнына, интернеттен керекті білімін іздеп тауып алады. Оны біз coursera, udemy, geekbrains сияқты жобалардың қарқынды дамуынан көре аламыз. Бұл жобалардың ескі білім алу әдістеріне қарағанда артықшылығы өте көп. Мысалы, технологиялардың өте жылдам дамуына байланысты адам білімі тез ескіреді, ал бұл жобаларда сол пайда болып жатқан жаңа технологияларға қатысты курстар бірден пайда болады. Оған қоса, сіз бір затты үйрену үшін арнайы уақыт бөлмей, кез келген бос уақыттыңызда, кез келген жерде Интернет арқылы білім ала аласыз. Бірақ мұның бәрі ағылшын, орыс тілдерінде ұсынылған. Қазақ тілінде жаңадан шығып жатқан технологиялар жайлы курстар олай бірден пайда болмайды. Тіптен пайда болмайды деп те айтуға болады.

Осындай жобалардың тағы бір үлкен артықшылығын осы жылы көрдік. Бүкіл әлемді жаулаған коронавирус пандемиясының кесірінен студенттер мен оқушыларға үйде отырып білім алуға тура келді. Көптеген білім беру жобалары өздерінің қорларын тегін қолдануды ұсынды. Ал Қазақстанды алып қарасақ, бізде қазақ тілінде ондай жобалар жоқ. Әрбір жоғарғы оқу орны өз күшімен өз студенттеріне түрлі әдістермен білім берді. Ал мектеп оқушылары тіптен, телеарналар арқылы оқу оқуға мәжбүр болды.

Осы себептерге байланысты бұл дипломдық жұмыстың өзектілігі бар деп ойлаймын.

Дипломдық жұмыстың тақырыбы: IT технологияларды қазақ тілінде үйрететін веб-ресурс жобалау.

Мақсаты: басқа тілдегі аналогтарды зерттей отырып қазақ тілінде IT технологияларды үйренуге болатын веб-ресурс жобалау. Бұл ресурста тіркелген қолданушылар ұсынылған курстарды үйрене алуы тиіс, білмеген заттары болса сұрақ қойып немесе басқа қолданушыларға көмектесіп, олардың сұрақтарына жауап бере алуы керек, IT жаңалықтарды қазақ тілінде оқи алуы тиіс және қазақша кітаптарды жүктей алуы керек.

Дипломдық жұмыстың жаңалығы: веб-ресурсты жобалау барысында қазіргі заманауи технологиялар қолданылуы тиіс. Оларға Node.js, Angular, React және т.б. технологияларды жатқызуға болады.

1 Аналитикалық бөлім

1.1 Есептің қойылымы

Пәндік аймақ: ИТ технологияларды қазақ тілінде үйрететін веб-ресурс.

Пәндік аймақтың сипаттамасы: Қазіргі уақытта, жылдан жылға интернет арқылы курстарды меңгеру, білім алу платформаларының қолданушылар саны артып жатыр. Оның басты себебі, барған сайын интернет арқылы білім алушы санының артуында және де жаңа технологиялардың тез дамып, бірін бірі алмастыруда. Технологиялардың тез даму нәтижесінде білім де тез ескіреді. Сол білімді жаңартып отыру үшін осындай платформалар өте қолайлы. Ондай платформаларға Coursera, Udemy, Udacity, Geekbrains және т.б. веб-сайттар жатады. Осындай платформалардың ең басты артықшылығы – сіз білім алу үшін белгілі бір уақытыңызды бөліп бір жерге барып оқудың қажеті жоқ, кез келген жерден интернет арқылы бос уақытыңызда білім ала беруіңізге болады.

Бірақ осындай білім алу ресурстарының басым көпшілігі ағылшын немесе орыс тілінде ғана кездеседі. Қазақ тілінде мәлімет өте аз және сол бар аз мәлімет тез ескіреді. Сіз қазіргі ИТ салада қолданылып жатқан технологияларды қазақ тілінде үйренейін десеңіз ауқымды мәлімет таба алмайсыз.

Осы сияқты платформалардың тағы бір артықшылығы осы жылы пандемия кесірінен студенттер мен оқушылар карантинде үйде отырып білім алған кезде көрдік. Жоғарыда аталған платформалар өз қорларын қолдануды барлығына тегін қылса, қазақ тілінде ондай ауқымды білімді беретін платформа табылмады. Әр жоғарғы оқу орныны өз студенттерін өз күштерімен біліммен қамтамасыз етіп отырды. Ал мектеп оқушыларына теледидар каналдары арқылы сабақ оқуына тура келді.

Осы мәселелерді шешу үшін, қазақ тілінде контент санын арттырып, қазақ тілін дамыту үшін қазақ тілінде ИТ сала бойынша білім беретін веб-платформа жобалап шығару керек.

1.2 Ақпараттық жүйені жобалау құралын негіздеу

Дипломдық жоба үшін белгілі бір платформаға арналған бағдарламалау ортасында бағдарлама жасаудың орнына веб-сайт жобаланады деп шешілді. Себебі бағдарламалау орталарында жасалған бағдарламалар белгілі бір операциялық жүйеде немесе платформада ғана орындалады.

Мысалы, Delphi бағдарламалау ортасында жасалған бағдарламалар көбінде Windows операциялық жүйесінде ғана жұмыс істейді. Бұл ортада жобаланған бағдарламаны Windows операциялық жүйесінен басқа операциялық жүйені қолданатын клиенттер қолдана алмайды. Сол сияқты, қолданушылардың барлығында Windows операциялық жүйесінде істейтін дербес компьютер болмауы мүмкін. Оның орнына оларда: macOS, Linux,

Ubuntu, Android, iOS, Symbian, J2ME, Tizen OS, Blackberry OS және т.б. операциялық жүйеде істейтін құрылғы болуы мүмкін. Ондай адамдардың барлығы Windows ОЖ-сіне арнап жобаланған бағдарламаны қолдана алмайды.

Бұл мәселені шешудің екі жолы бар:

– әрбір операциялық жүйеге арнап сол платформада ғана жұмыс істейтін бағдарлама құру;

– кез келген операциялық жүйеде орындалатын кроссплатформалы ақпараттық жүйе құру.

Бірінші әдіс бойынша істеген жағдайда, әрбір операциялық жүйеге арналып құрылған бағдарламалардың әрқайсысын құруға өте көп уақыт, бағдарламашылар және көп қаражат кетеді. Ал веб-сайт барлық операциялық жүйелерде бірдей жұмыс істегендігінен, оған уақыт, бағдарламашылар және ақша бірінші әдістегіден қарағанда біршама аз кетеді. Он бағдарламаны қамтамасыз еткенше, бір веб-сайтты қамтамасыз еткен өте тиімді болып табылады.

Оған қоса веб-технологиялар қазіргі уақытта ең тез дамып жатқан сала болып табылады. Ең көп қолданылатын бағдарламалау тілдерін JavaScript, Python және PHP веб-бағдарламалау тілдері бастайды. Сонымен қатар басқа бағдарламашыларға қарағанда, веб-сайттар жобалайтын веб-бағдарламашылар өте көп кездеседі. Ал бүкіл әлем бойынша ең бай компаниялардың қатарына Google және Amazon сияқты компаниялар кіреді.

Кез келген адам компьютерде жұмыс істеу уақытының көбінде браузерлер арқылы сайттармен жұмыс істеп отырады. Веб-сайтқа қатынау үшін интернет пен браузерден басқа ештеңе керек жоқ. Веб сайтқа кез келген смартфоннан, ұялы телефоннан, компьютерден және т.б. көптеген құрылғылардан кіруге болады. Сондықтан ол қолданушылардың ауқымды көлеміне қолжетімді болады.

1.3 Қолданылған бағдарламалық тілдер

Бүгінгі уақытта веб-сайт құруға арналған біршама бағдарламалау тілдері бар. Олардың ішінде Python (Django), Ruby (Ruby on rails), Perl (Mojolicious), PHP (Laravel, Zend, Yii) және де JavaScript (Node.js). Python, Perl, PHP, Ruby-лардан қарағанда, JavaScript қазіргі уақытты әсіресе көп қолданылуда. Сайттың серверлік жағын (backend) және клиент жағын да (frontend) JavaScript арқылы жасауға болады. Ол Node.js программалық платформасының қарқынды дамуының арқасында JavaScript те дамып жатыр. Сол себептен JavaScript тілінде веб-сайттар жасау қазір дұрыс шешім.

Дипломдық жұмыста JavaScript тілін қолдану себептері:

– 2018 жылы StackOverflow сайтының программистері арасындағы бүкіләлемдік ең ауқымды сауалнама нәтижесінде Node.js ең танымал фреймворк екені анықталды. Бұл технологияны 50 пайызға жуық программистер қолданады [2].

– Веб-хостингтердің 95 пайызы JavaScript тілін қамтамасыз етеді. Егер сайт басқа тілде жазылған болса, оған хостинг табуға қиын болушы еді.

– Барлық ең танымал операциялық жүйелерімен қолданады.

– Көптеген қосымшалар мен дайын кітапханалар бар.

JavaScript – объектілі-бағытталған, императивті және функционалды стильдерді қамтамасыз ететін мультипарадигмалық бағдарламалау тілі. JavaScript бағдарламалаудың көптеген салаларында қолданылады. Веб-бағдарламалауда ол HTML тілі ұсына алмайтын заттарды жүзеге асыру үшін, яғни веб-парақтарға интерактивтілік беру үшін қолданылады. Соңғы уақытта JavaScript тілі сервер мен клиенттік жақта да толығымен қолданылатын болды. Ол Angular, React, Vue, Svelte сияқты фреймворктардың күшімен жүзеге асырылады.

TypeScript – JavaScript тілінің мүмкіндіктерін кеңейтетін веб-қосымшаларды жобалауға арналған бағдарламалау тілі. Ол 2012 жылы Microsoft компаниясымен ұсынылған. TypeScript тілін жобалаушы кезінде Turbo Pascal, Delphi, C# тілдерін жобалаған Андерс Хейлсберг болып табылады.

TypeScript-та жазылған код JavaScript-қа компиляцияланады. Код компиляцияланғаннан кейін оны кез келген заманауи браузерлерде немесе Node.js серверлік платформасында қолдана беруге болады. Компилятор Apache лицензиясы бойынша таратылады және оның коды ашық түрде GitHub сервисінде салынған.

TypeScript-тің JavaScript-тен айырмашылығы көп. Олар: мәліметтер типін статикалық көрсету, яғни бір типтегі айнымалыға басқа типтегі мәліметті еңгізуге болмайды, дәстүрлі объектіге-бағытталған бағдарламалау тілдеріндегідей класстарды қамтамасыз етеді, модульдерді қосу мүмкіндігі бар, рефакторинг және кодты қайта қолдануға болады.

HTML (HyperText Markup Language) – Ғаламтордағы құжаттарды белгілеу стандартталған тілі. Веб-парақтардың белгіленуі көбінде HTML тілінде сипатталады. HTML тілін компьютер немесе телефон мониторуна өңдеп шығаратын бағдарлама браузер деп аталады. HTML парақтар HTTP немесе HTTPS протоколы арқылы браузерға жай немесе шифрленген мәтін түрінде жіберіледі. Дипломдық жұмыста HTML тілінің ең соңғы, бесінші нұсқасы қолданылады.

CSS (Cascading Style Sheets – каскадтық кестелер стилі) – белгілеу тілі арқылы жазылған құжаттың сыртқы көрінісін сипаттауға арналған формалдық тіл. Көбінесе HTML немесе XHTML белгілеу тілінде жазылған веб-парақтардың сыртқы көрінісін безендіру, сипаттау құралы ретінде қолданылады. Бірақ оған қоса кез келген XML-құжаттарын безендіруде де қолданыла алады. Мысалы, SVG немесе XUL. CSS веб-парақтарды құрушылармен веб-парақтарға түстер, шрифттер, блоктардың орналасуын және басқа да сыртқы көрінсін сипаттау үшін қолданылады.

SQL (Structured Query Language – құрылымдық сұраныстар тілі) – реляциондық деректер базасын басқару жүйесіндегі деректерді басқаруда, құруда және өзгертуде қолданылатын бағдарламалау тілі.

1.4 Қолданылған жаңа технологиялар

Angular – Google компаниясының программистері мен басқа да компаниялардың программистер қауымымен жобаланған, TypeScript тілінде жазылған, веб-қосымшаларды жобалауға арналған еркін және ашық платформа. Angular – бұл AngularJS фреймворгын жасаған программистер тобының толығымен қайта жазылған фреймворгы. Алғашында AngularJS-тің екінші нұсқасы ретінде жасалып бастаған. Оның архитектурасы AngularJS-тен бөлек және кері байланысы жоқ. Сол себептен адамдар шатаспас үшін оны бөлек фреймворк ретінде дамыту шешілді.

Angular ұсынатын функционалдылықтарға келесілер жатады: екіжақты байланыс, шаблондар, маршрутизация және т.б. Angular көбінесе SPA (Single Page Application) жобалау кезінде қолданылады. Ал мен оны админ-панель жасау кезінде қолданамын.

React – қолданушы интерфейстерін жобалауға арналған ашық кодты JavaScript-кітапхана. React кітапханасы Facebook, Instagram және басқа да программистер мен корпорациялармен жобаланып, қамтамасыз етіледі.

React бір беттік қосымшалар (SPA) жобалауда да, мобильдік қосымшалар жобалауда да қолданыла алады. Оның мақсаты – тез жылдамдық, қарапайымдылық және масштабталуды ұсыну. Қолданушылық интерфейстерді жобалау кезінде React басқа да кітапханаларды қолдана алады. Оған мысал ретінде Redux кітапханасын қарастыруға болады.

React-тың артықшылықтары:

1) Декларативтік.

React арқылы интерактивті қолданушы интерфейстерін жобалау ыңғайлы. Ол үшін интерфейснің белгілі бір бөліктері әртүрлі жағдайларда қандай болу керегін сипаттасақ болды, React керекті кезде оларды жаңартып, деректерді өзгертіп отырады.

Декларативті көрініс кодты өте ұғынықты қылуға және де ретке келтіруді жеңілдетеді.

2) Компоненттер негізінде қаланған.

Жеке жағдайы бар инкапсуляцияланған компоненттер құрып, оларды үлкен, әрі қиын қолданушылық интерфейстерге біріктіруге болады.

Компоненттің логикасы JavaScript-та жазылғандықтан, мәліметтерді бүкіл қосымша бойынша жіберіп, компоненттің жағдайын DOM-нан тыс сақтауға болады.

3) Кроссплатформалық.

Қосымшаның технологиялық құралдарын білмесек те, React негізінде ескі кодты өзгертпей жаңа функционалдылық жобалауға болады.

Оған қоса, React Node.js арқылы серверде және React Native арқылы мобильдік платформаларда жұмыс істей алады.

SCSS – SASS тілінің «диалектісі». SASS дегеніміз HAML тіліне ұқсас, CSS-кодты жеңілдетуге арналған тіл. Қарапайым сөзбен айтқанда, арнайы ruby-программа арқылы SASS тіліндегі кодты CSS-кодқа аударады. Бұл тілдің

синтаксисі өте икемді. Оған қоса SASS тілінде логика (@if, each), айнималылар және математика бар (сандарды, жолдарды және түстерді қосу, азайту).

Node немесе Node.js – JavaScript тілін тар бағыттағы тілден жалпы мақсаттағы тілге айналдыратын, V8 (JavaScript кодты машиналық кодқа айналдырады) негізінде қаланған программалық платформа. Node.js көмегімен JavaScript сыртқы еңгізу-шығару құрылғыларымен API арқылы байланыса алады, басқа тілде жазылған сыртқы кітапханаларды қоса алады. Node.js көбінесе серверде веб-сервер қызметін атқарып қолданылады. Бірақ та Node.js арқылы дербес компьютерлерге терезелік қосымшалар (NW.js, AppJS немесе Electron арқылы) жобалау мүмкіндігі бар. Оған мысал ретінде: Visual Studio Code, Discord бағдарламалары Electron және Node.js арқылы жасалған.

Nest.js – эффективті, сенімді және масштабталатын серверлік қосымшалар құруға арналған Node.js прогрессивті ортасы. Басқаша айтқанда, Nest.js – бұл дұрыс архитектуралық амалдарды қолданатын, программистердің өмірін жеңілдетуге арналған фреймворк.

Сол себепті Nest.js – бұл жай ғана бэкендке арналған фреймворк емес, бұлDDD, Event sourcing және микросервистік архитектура сияқты заманауи концепцияларды түсінуге мүмкіндік беретін орта.

Nest.js-тің басты артықшылықтары:

1) Созылмалы. Модульдік архитектура арқасында, сіз кез келген басқа кітапханаларды қоса аласыз.

2) Жан-жақты. Барлық серверлік қосымшалар түріне толық магистраль ұсынатын икемді экосистема.

3) Прогрессивті. JavaScript тілінің соңғы мүмкіндіктерін қолдана отырып, Node.js платформасына жобалау шаблондары мен жаңа шешімдер еңгізеді.

Nest.js фреймворгының архитектурасы Angular-дың архитектурасына ұқсас. Ол да TypeScript тілін, декораторлар, модульдер мен компоненттерден құралады.

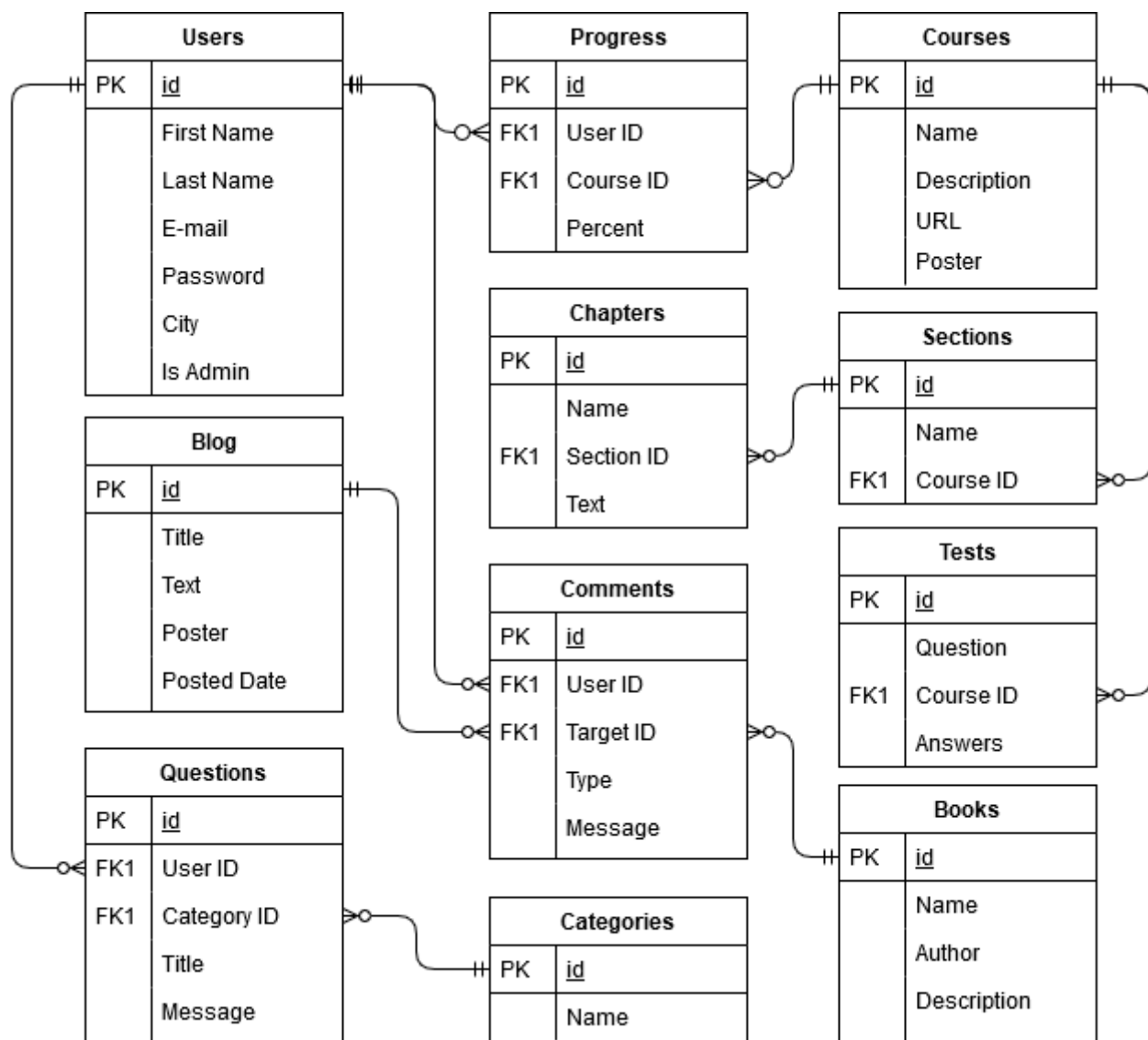
TypeORM – бұл Node.js, PhoneGap, Cordova, React Native, Ionic, NativeScript, Expo, Electron платформаларында және TypeScript, JavaScript тілдерімен қолданыла алатын ORM (Объект-реляциялық теңестіру). Оның басты мақсаты – JavaScript тілінің ең соңғы функцияларын қамтамасыз ету және деректер базасын қолданатын кез келген қосымшалар (бірнеше кестесі бар кішігірім қосымшалардан бастап, бірнеше деректер базасын қолданатын үлкен корпоративті қосымшаларға дейін) жобалауға көмектесетін қосымша функцияларды ұсыну.

TinyMCE (Tiny Moxiecode Content Editor) – Web негізіндегі кросс-платформалы JavaScript HTML WYSIWYG мәтін редакторы. Басты ерекшеліктеріне үлгілер, тілдер, модульдер қосу мүмкіндігі бар. Редактор суреттер, кестелер, түсті мәтіндер, видео және т.б. заттар қосуға мүмкіндік береді. Бұл редактормен Microsoft, IBM, LinkedIn, Evernote, WordPress және т.б. үлкен компаниялар қолданады. Барлығы жүз миллионнан астам сайт қолданады.

2 Жобалау бөлімі

2.1 Деректер базасын концептуалды жобалау

Нысан-байланыс моделі (Entity-Relationship model) – бағдарламалық қосымшаны жобалаудағы дерекқорды сипаттау үлгісі. Ол пәндік аймақтың концептуалды кескіндерін сипаттауға мүмкіндік беретін деректер моделі. Оның көмегімен негізгі түйіндер мен олардың арасында орнатыла алатын байланыстарды белгілеп алуға болады (1-сурет).



1 сурет – ER модель

Жоғарыдағы суретте көрсетілген ER-модель негізінде деректер базасының кестелері құрылды. Барлығы он кесте құрылды. Оларға қысқаша сипаттама:

– Users (Қолданушылар) – бұл кестеде қолданушының аты-жөні, поштасы, құпиясөзі, қаласы енгізіледі.

– Blog (Мақалалар) – бұл кестеде мақалалар мен жаңалықтар сақталады. Мақаланың тақырыбы, мәтіні, суреті және шыққан күні болады.

– Books (Кітаптар) – бұл кестеде кітаптың аты, сипаты, авторы, суреті және сілтемесі жазылады.

– Questions (Сұрақтар) – бұл жерде қолданушылар қоятын сұрақтар сақталады. Сұрақтың тақырыбы, толық сипаттамасы, сұрақ қойған қолданушы және оның категориясы болады.

– Categories (Категориялар) – бұл кестеде сұрақтардың категориялары сақталады. Мысалы, веб-жобалау, операциялық жүйелер және т.б.

– Comments (Пікірлер) – бұл кестеде қолданушылардың белгілі бір жазбаға қалтырған пікірлері сақталады. Пікірді кітап, мақала, курс немесе қолданушының парағында қалтыруға болады.

– Courses (Курстар) – мына кестеде курстың аты, сипаттамасы, суреті және сілтемесі сақталады.

– Sections (Тақырыптар) – бұл кестеде белгілі бір курстың тақырыптары сақталады.

– Chapters (Бөлімдер) – бұл кестеде сол тақырыптардың ішіндегі бөлімдер сақталған. Бөлімдердің аты және мәтіні болады.

– Tests (Тесттер) – бұл кестеде курс бойынша тесттер сақталады. Тесттің сұрағы, дұрыс жауабы және дұрыс емес жауаптары болады.

– Progress (Нәтиже) – бұл кестеде тесттің нәтижесі сақталады. Нәтиже қолданушы дұрыс жауап берген сұрақтардың барлық сұрақтар санына қатысты пайыз арқылы есептеледі.

2.2 Веб-сайтты концептуалды жобалау

Мақсаты:

1. Студенттерге, мектеп оқушыларына немесе кез келген ИТ саласымен қызығатын адамдарға білім беруге арналған ыңғайлы сайт жобалау.

2. Қазақ тіліндегі курстар, мақалалар және кітаптарға қол жеткізу мүмкіндігін ұсыну.

3. Оқу барысында қиыншылықтар туындаған кезде басқа білім алушылардан немесе білімі жоғары мамандардан қазақ тілінде көмек сұрау мүмкіндігін қамтамасыз ету.

4. ИТ саласымен қызығатын басқа адамдармен танысуға және байланысуға мүмкіндік туғызу.

5. Өз білімін қазақ тіліндегі тесттер көмегімен тексеру және қадағалап отыруға мүмкіндік беру.

Қолданушылар:

– ИТ саласына қызығатын және жаңа технологияларды үйренігсі келетін адамдар. Оларға: студенттер, оқушылар немесе жай адамдар жатады.

– Сайттың администраторлары. Сайт администраторлары сайтқа жаңа

мақалалар, кітаптар, курстар және т.б. мәлімет еңгізеді.

Пайдалану сценарийлері. Мұнда сайттың келесі қолданушылар тобына қажетті бөлімдері сипатталған.

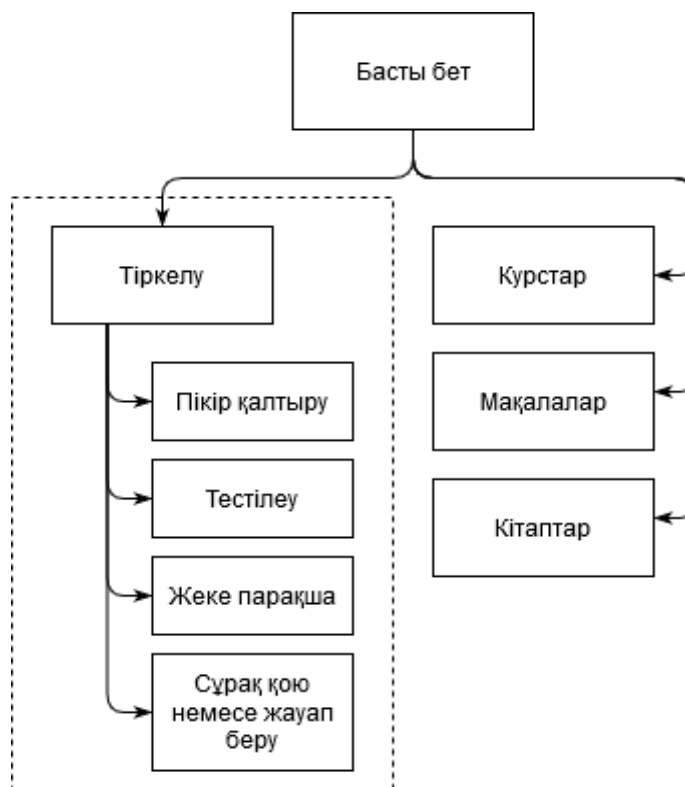
Қолданушылар:

- қазақ тілінде курстар оқу;
- курстар бойынша өз білімін тест арқылы тексеру;
- қазақ тіліндегі ИТ саласына қатысты кітаптар жүктеу;
- оқу барысында сұрақ туындаса басқалардан көмек сұрау;
- қазақша мақалалар және жаңалықтар оқу.

Администраторлар:

- жаңа кітаптар еңгізу;
- жаңа мақалалар мен жаңалықтар жазу;
- кітаптар мен мақалаларды өзгерту және жою.

2-суретте веб-сайтты пайдалану сценарийлерін зерттеу нәтижесінде туындаған веб-сайттың парақтарының орналасуы көрсетілген. Сол жағында көрсетілген, үзіксызықты тіктөртбұрыштың ішінде тек қана сайтқа тіркелген қолданушыларға ғана қолжетімді парақтар көрсетілген. Ал оң жақтағы парақтарды кез келген қолданушы көре алады.



2 сурет – Веб-сайттың концептуалды схемасы

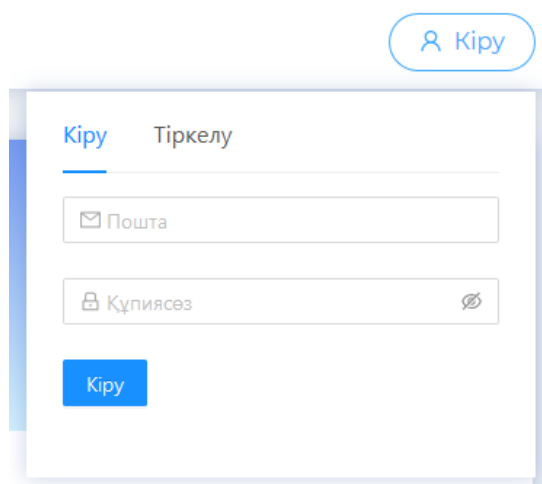
Концептуалды схемаға байланысты сайттың архитектурасы мен бизнес-логикасы құрылады, және ол бойынша сайттың коды жазылады. Сілтемелер «адамға түсінікті URL» концепциясы бойынша жасалды.

2.3 Веб-сайтты қолдану

2.3.1 Қолданушыларға арналған басты веб-қосымша

Жалпы дипломдық жұмыс барысында екі веб-қосымша жобаланды. Біреуі администраторлар үшін және екіншісі барлық қолданушыларға. Бұл бөлімде React кітапханасы арқылы жобаланған, қолданушыларға арналған веб-қосымша қарастырылады.

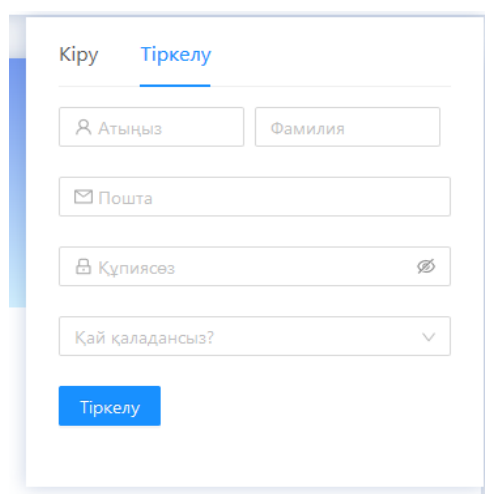
Төменгі суретте басты бетте орналасқан сайтқа кіру батырмасы көрсетілген. Батырманы басатын болсақ сайтқа кіру формасы ашылады. Бұл жерде электрондық пошта мен құпиясөз еңгізіліп, көк батырма басылады.



The image shows a login form with a blue header bar containing a magnifying glass icon and the text 'Кіру'. Below the header, there are two tabs: 'Кіру' (selected) and 'Тіркелу'. The form contains three input fields: 'Пошта' (Email) with an envelope icon, 'Құпиясөз' (Password) with a lock icon and a toggle eye icon, and a blue 'Кіру' button.

3 сурет – Сайтқа кіру формасы

Егер қолданушы сайтқа бірінші рет кіріп отырса, тіркелу сілтемесін басып тіркеледі. Бұл жерде ол өз аты-жөнін, поштасын, қаласын және құпиясөзін еңгізеді.



The image shows a registration form with a blue header bar containing a magnifying glass icon and the text 'Тіркелу'. Below the header, there are two tabs: 'Кіру' and 'Тіркелу' (selected). The form contains four input fields: 'Атыңыз' (Name) with a magnifying glass icon, 'Фамилия' (Surname), 'Пошта' (Email) with an envelope icon, and 'Құпиясөз' (Password) with a lock icon and a toggle eye icon. There is also a dropdown menu labeled 'Қай қаладансыз?' (Which city are you from?). At the bottom, there is a blue 'Тіркелу' button.

4 сурет – Жаңа қолданушы тіркелетін форма

Сайтқа кіргеннен кейін қолданушының парағы ашылады. Қолданушының парағын 5 суретте көре аласыз. Бұл жерте оның суреті, аты-жөні, поштасы, қаласы және өткен тесттер нәтижесі көрсетіледі.



The image shows a user profile card for Madiyar Bolatov. It features a circular profile picture of a dog sitting at a desk with a laptop. To the right of the picture, the name 'Мадияр Болатов' is displayed in large, bold black letters. Below the name, there is a breadcrumb trail: 'Қолданушы / Мадияр Болатов'. Underneath the breadcrumb, there are two buttons: 'E-mail' with an envelope icon and 'Алматы' with a globe icon. Below the profile card, there is a circular progress indicator showing 60% completion, with the word 'Python' written below it.

5 сурет – Қолданушы парағы

Төменге қарай түсетін болсаңыз, қолданушының парағында басқа қолданушылар жазып кеткен пікірлерді көруге болады. Ол 6 суретте көрсетілген. Бұл қолданушылар өзара байланысу үшін арналған.

Пікірлер



Асқар Тулеев

Қазір қандай тілді үйреніп жүрсің?

☰ Жауап беру ❤️ 0

Сіздің пікіріңіз

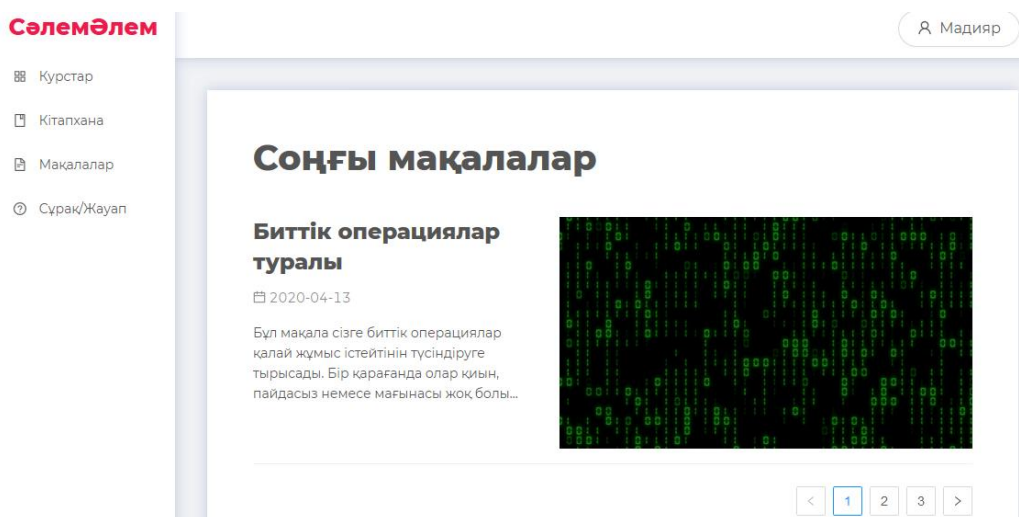
Жазу

6 сурет – Қолданушы парағындағы пікірлер

Төмендегі, 7 суретте басты бет көрсетілген. Бұл бетте соңғы мақалалар, соңғы үш курс және төрт кітап көрсетіледі. Басты бет арқылы сайтта болып жатқан соңғы өзгерістерді көруге болады.

Соңғы мақалалар бөлігінде ең соңғы енгізілген мақала жалғыз тұралы. Оның астыңғы жағындағы сілтемелерді басу арқылы одан ерте жазылған мақалаларды көруге болады. Мақаланың суреті, тақырыбы, шығарылған күні және қысқа мәтіні көрсетіледі.

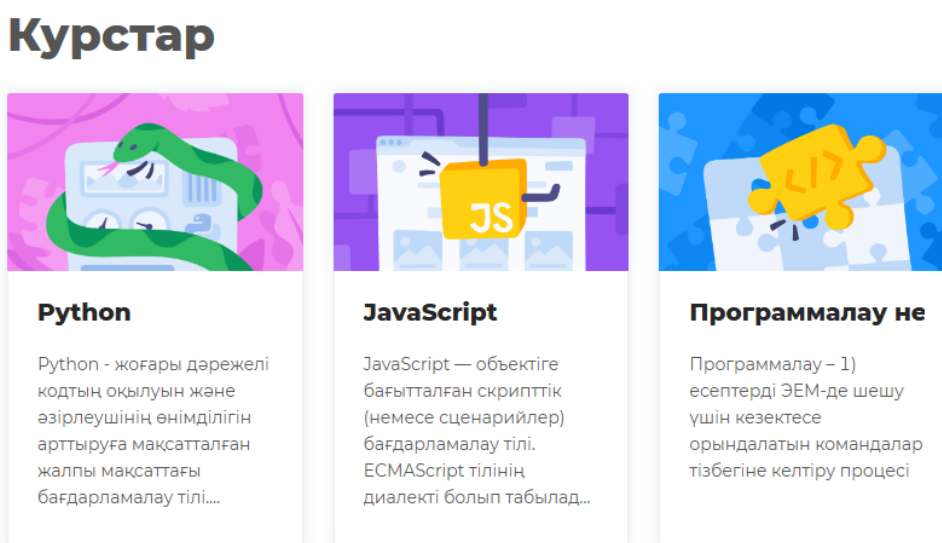
Сол жағында сайттың басты бөлімдеріне сілтемелер көрсетілген.



7 сурет – Басты бет

Сегізінші суретте басты беттегі соңғы үш курс көрсетілген. Егер «Курстар» сөзін басатын болсақ, сайттағы барлық курстар тізімі бар парақ ашылады.

Курстардың атауы, қысқаша сипаттамасы және де суреті болады.



8 сурет – Соңғы курстар тізімі

Курстар парағына кіріп кез келген бір курсты таңдайық. Мысалы, 9 суретте Питон тілі бойынша курстың парағы көрсетілген. Бұл жерде курстың атауы, қысқаша сипаттамасы көрсетілген. Төменгі жақта осы курс жайлы пікірлер жазылған. Ал оң жақта осы курсқа енетін бөлімдер мен тестке сілтеме көрсетілген.

9 сурет – Курстың парақшасы

Курсты өту үшін оң жақтағы бөлімдер тізімінен бір тақырыпты таңдаймыз. Мысалға, 10 суретте бірінші тақырып таңдалды. Тақырып таңдаған кезі пікірлердің орнына тақырыптың мәтіні көрсетіледі.

10 сурет – Курстың бірінші тақырыбы

Курста ұсынылған барлық бөлімдерді меңгеріп біткеннен кейін тестілеу батырмасын басып өз біліміңізді тексеруге болады. Питон бойынша тест 11 суретте көрсетілген.

Python

🏠 / 📖 Курстар / Python / Тест

Python - жоғары дәрежелі кодтың оқылуын және әзірлеушінің өнімділігін арттыруға мақсатталған жалпы мақсаттағы бағдарламалау тілі. Python тілі аз синтаксисті талап етеді. Бірақ сол уақытта стандартты к... [Expand](#)

1) Python тілінің қазіргі уақытта қай нұсқасы қолданылады?

- 2.5
- 4.8
- 3.7

2) "print" командасы не істейді?

11 сурет – Питон курсы бойынша тест

Тесттің барлық сұрақтарына жауап бергеннен кейін, 12 суретте көрсетілген «Жауап беру» батырмасы басылады. Батырманы басқаннан кейін, қолданушы қанша сұраққа дұрыс жауап бергені пайыз бойынша есептеліп, қолданушының парағына еңгізіледі. Оның нәтижесі 5 суретте көрсетілген.

- Монти Пайтон шоуы атына
- Питон алгоритмін қолдану себебі

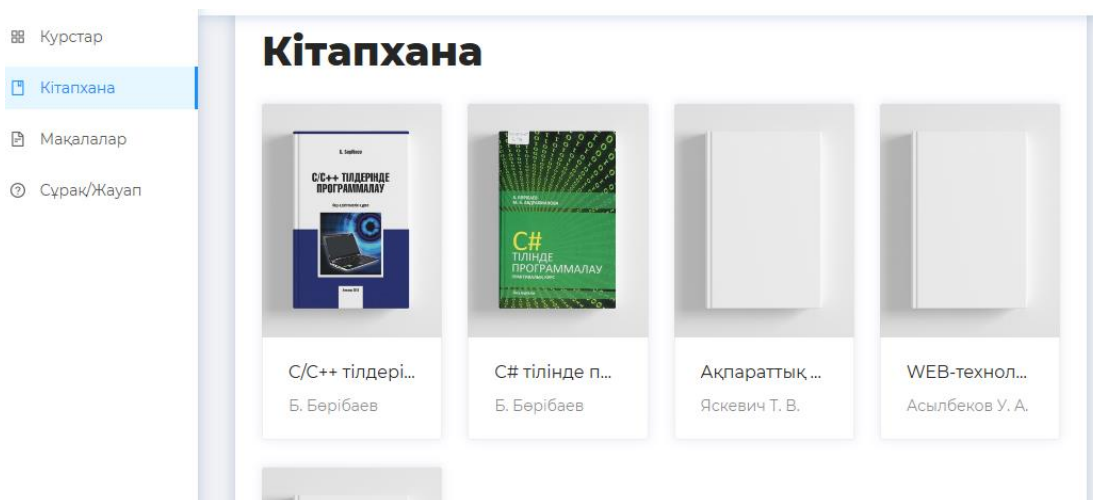
5) Питонның ерекшелігін таңда

- ООП негізінде жасалған
- Үтірлі нүкте қолданылмайды
- Анимациялар жасауға арналған

Жауап беру

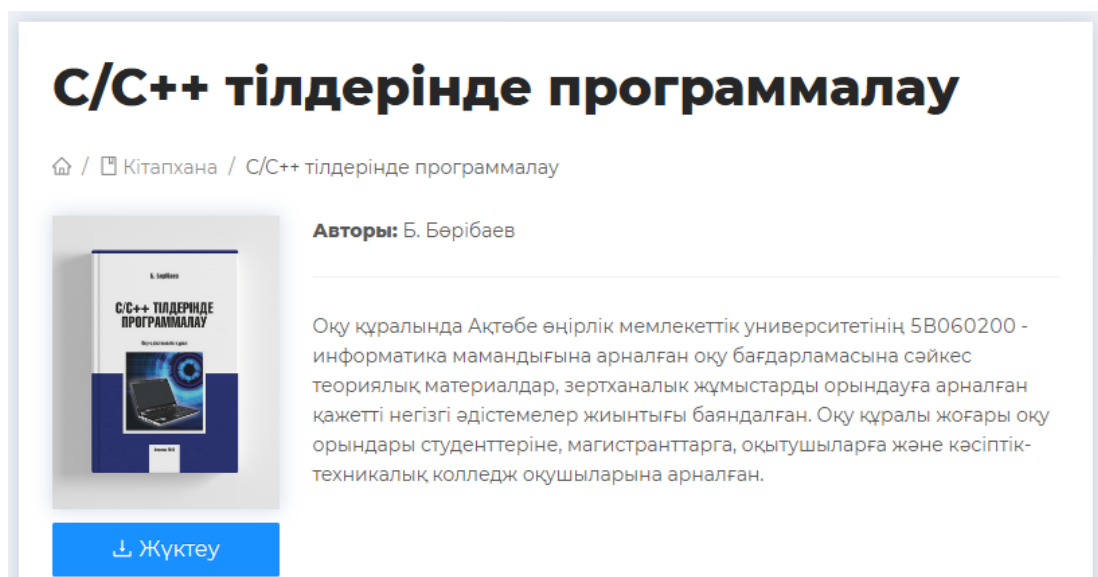
12 сурет – Тестті аяқтау батырмасы

Кітапхана сілтемесін ашсаңыз кітаптар тізімі көрсетіледі. Ол төмендегі 13 суретте көрсетілген. Бұл жерте кітаптың суреті, авторы және атауы беріледі.



13 сурет – Кітаптар тізімі

Төмендегі, 14 суретте кез келген бір кітаптың парағы көрсетілген. Бұл жерде атауы, суреті және авторынан басқа, кітаптың қысқаша сипаттамасы мен жүктеуге арналған сілтемесі орналасқан. Барлық кітаптар ашық ақпарат көздерінен алынған және жүктеу сілтемесі міндетті түрде файлға емес, кітап авторының жеке сайтына да сілтеуі мүмкін.



14 сурет – Кітап парақшасы

Кітаптың сипаттамасынан төмен қарай түсетін болсақ, бұл жерде осы кітап жайлы қолданушылар өз пікірлерін қалтырғанын көруге болады. Бұл бөлікте кез келген қолданушы өз пікірін қалтырып немесе басқа қолданушылармен осы кітап жайлы пікір таласуына болады.

Жалпы пікір қалтыру функционалдылығы жалғыз кітаптардың астында ғана емес, оған қоса жоғарыда көргендеріңіздей әр қолданушының жеке парағында, мақалалар мен жаңалықтардың астында, курстардың парақшасында да істейді.

Мысалы, төмендегі 15 суретте кітаптың астында қалтырылған пікір көрсетілген. Пікірді жазған адам өз пікірін «Жою» батырмасы арқылы жоя алады және де «Жауап беру» батырмасы арқылы басқа адамдарға жауап бере алады.

Пікірлер



Мадияр Болатов

Жақсы кітап

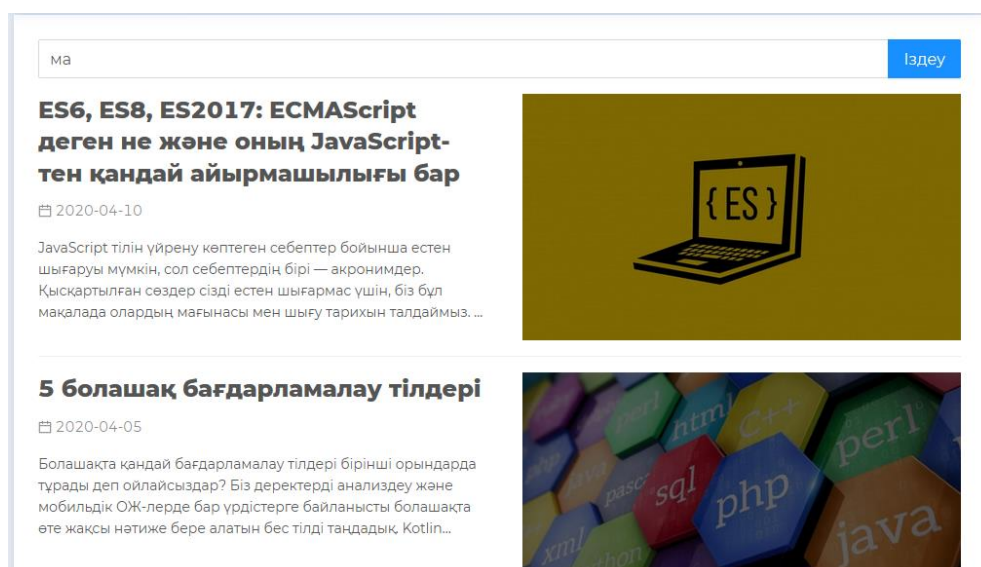
🗑 Жою ❤️ 0

Жазу

15 сурет – Пікір қалтыру формасы

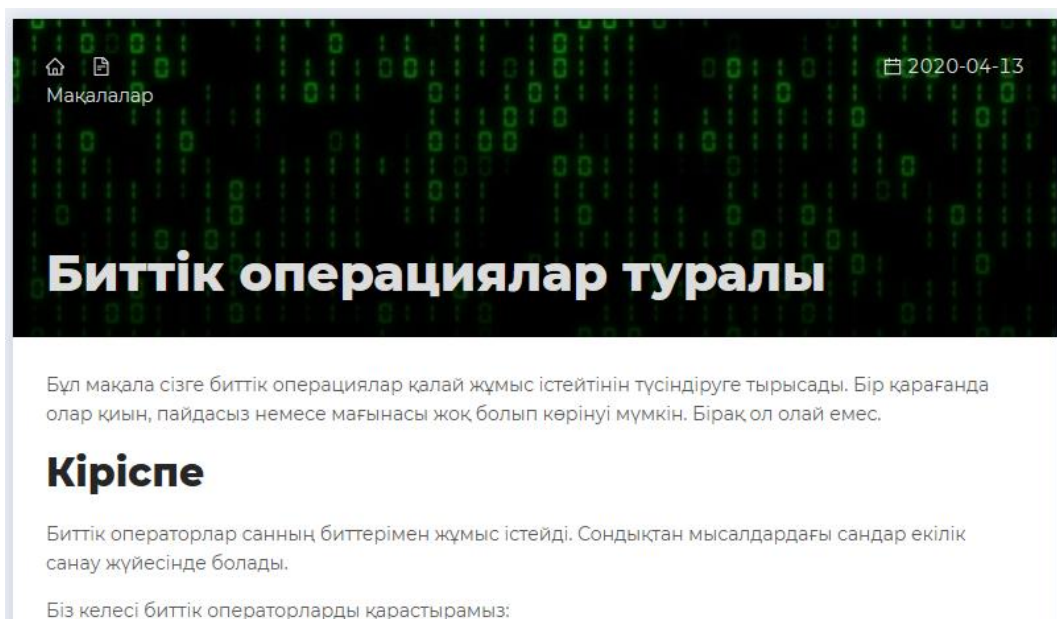
Мақалалар парағына кіретін болсақ барлық мақалалар тізімі көрсетіледі. Оған қоса, бұл парақта мақалаларды тақырыбы бойынша іздеуге болады. Ол төменде 16 суретте көрсетілген.

Бұл жерде мақалалардан басқа ИТ саласына қатысты жаңалықтар да еңгізіледі.



16 сурет – Мақалалар парағы

Мақаланың парақшасына кіретін болсақ, төменгі 17 суретте көрсетілгендей көріністі көресіз. Бұл жерде мақаланың тақырыбы, мәтіні, шыққан күні және суреті көрсетіледі. Мақаланың ең соңында пікір таласуға болады.

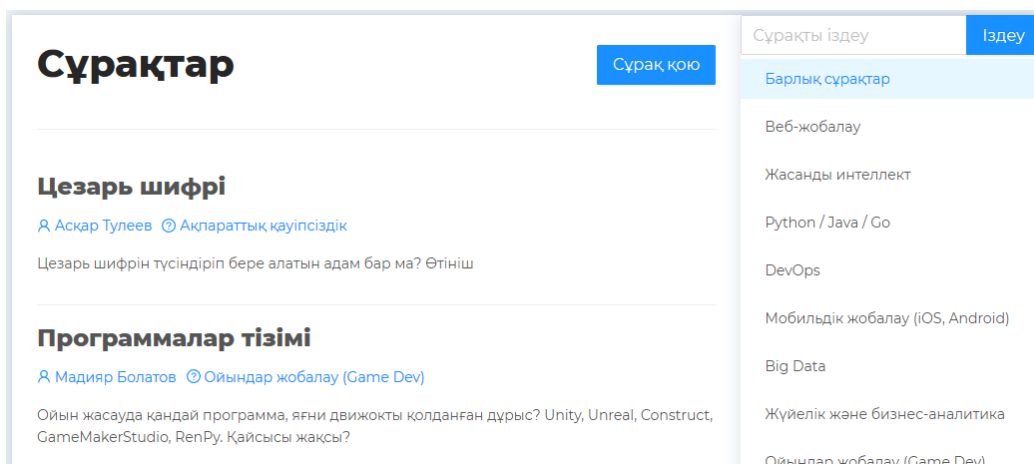


17 сурет – Мақала парақшасы

Кез келген қолданушы оқу барысында қиыншылықтарға сүрінуі мүмкін. Сұрақтар парағы сондай уақыттарда басқалардан көмек сұрау үшін қажет. Бұл жерде кез келген қолданушы ИТ саласына байланысты сұрақ қоя алады.

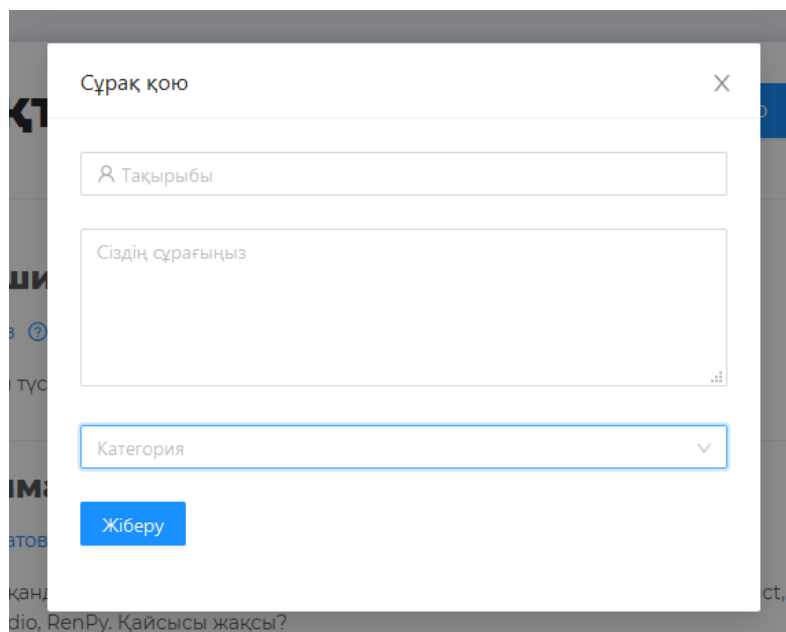
18 суретте көрсетілген сұрақтар парағында соңғы сұрақтар тізімі көрсетілген. Ал оң жағында жалпы тақырыптар тізімі ұсынылған.

Тақырыптар тізімінің жоғарғы жағында іздеу жолағы орналасқан. Ол сізді мазалайтын сұрақты бұрын талқылап қойды ма екенін білуге арналған. Егер өз сұрағыңызды тапсаңыз, тез жауап табуға болады.



18 сурет – Сұрақтар парағы


Ал егер сұрағыңыздың жауабын таппасаңыз, өз сұрағыңызды қоюға болады. Ол үшін «Сұрақ қою» батырмасын басасыз. Батырманы басқан кезі 19 суретте көрсетілген форма ашылады. Бұл жерде сұрақтың тақырыбын, сұрақты толық ашып мәтінін және категориясын таңдайсыз. «Жіберу» батырмасын басқанда сұрақ енгізіледі.




19 сурет – Сұрақ қою формасы

Сұрақ қою батырмасын басқаннан кейін сұрақтың парағы ашылады. Бұл жерде басқа қолданушылар сіздің сұрағыңызға жауап бере алады. Төменде, 20 суретте сұрақ парақшасы көрсетілген.

🏠 / 🗨️ Сұрақтар / Ақпараттық қауіпсіздік /

 **Цезарь шифрі**
[Аскар Тулеев](#)
Цезарь шифрін түсіндіріп бере алатын адам бар ма? Өтініш

Пікірлер

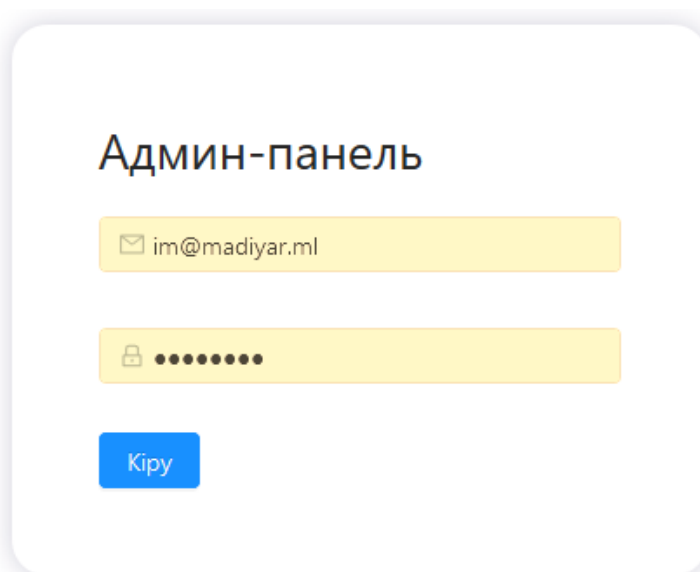
 **Мадияр Болатов**
Википедиядан оқысаң болады. Сол жерде жақсы түсіндірілген
👤 Жою 🍷 0

20 сурет – Сұрақ парақшасы

2.3.2 Администраторларға арналған веб-қосымша

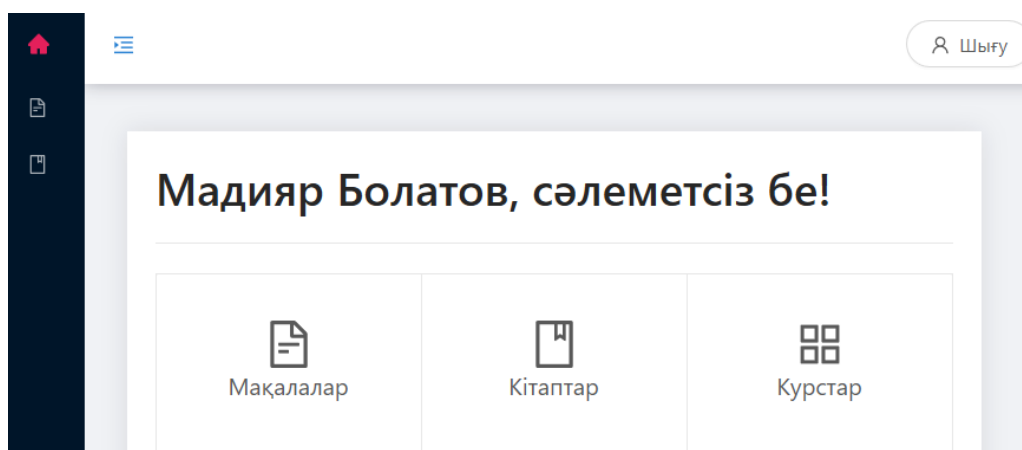
Бұл бөлімде Angular кітапханасы арқылы жобаланған, администраторларға арналған веб-қосымша қарастырылады.

Төменгі суретте басты бетте орналасқан администратор үшін арналған сайтқа кіру формасы көрсетілген. Бұл жерде электрондық пошта мен құпиясөз еңгізіліп, көк батырма басылады. Қолданушы администратор болатын болса, оның сайтқа кіруі расталады. Қолданушы администратор ма немесе қарапайым қолданушы ма екені «isAdmin»бағаны бойынша анықталады. Егер ол баған true болса, демек қолданушы администратор дегенді білдіреді. Ал false болса, керісінше, қарапайым қолданушы болып табылады.



21 сурет – Админ-панельге кіру формасы

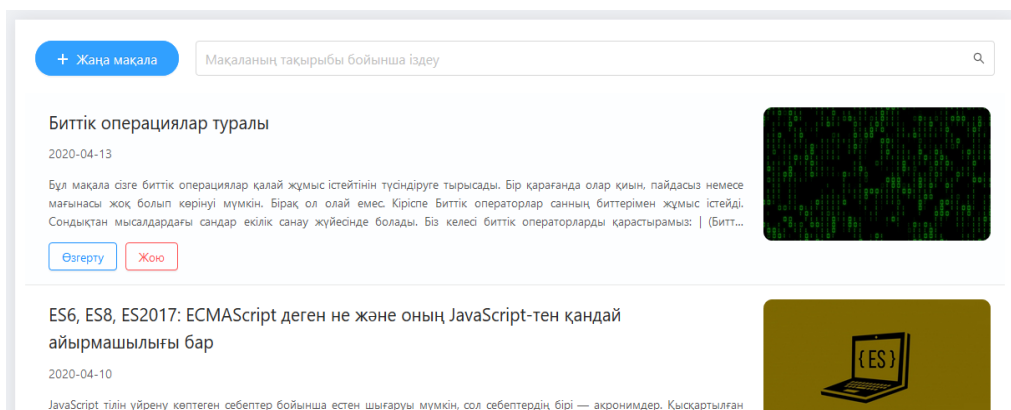
Сайтқа кіргеннен кейін басты бетте меню көре аласыз. Ол төменгі, 22 суретте көрсетілген. Үстіңгі оң жағында сайттан шығу батырмасы бар.



22 сурет – Админ-панельдің басты беті

Мақалалар парағын ашатын болсақ, сайтқа енгізілген мақалалар тізімі көрсетіледі. Оған қоса бұл жерде мақаланы тақырыбы бойынша іздеуге болады. Мақаланың қысқаша мәтінінің астында жою батырмасы бар. Оны басатын болсақ мақала жойылады.

Мақаланы тақырыбы бойынша іздеу жолағы екіжақты байланыс принципі арқылы жасалған. Яғни, біз сөзді енгізе бастай салысымен, төмендегі мақалалар тізімі бірден өзгеріп отырады.

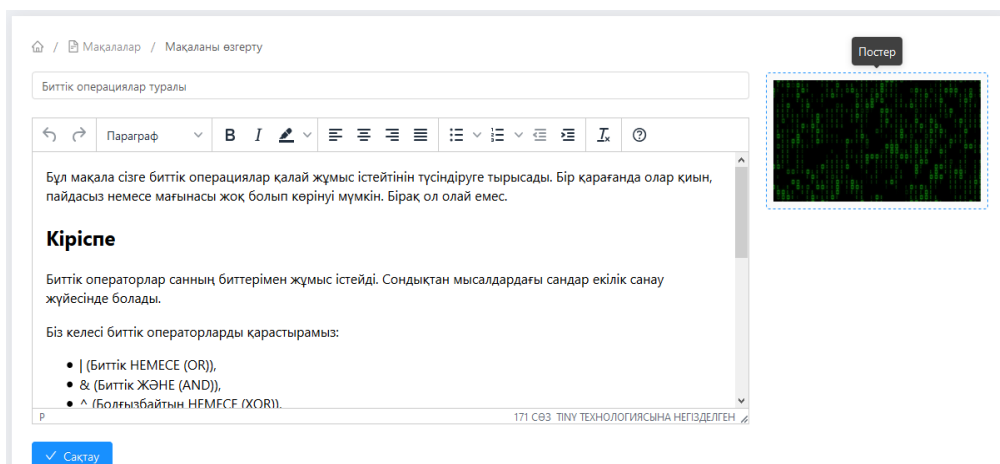


23 сурет – Мақалалар тізімі

Мақаланы өзгерту немесе жаңа мақала енгізу батырмасын бассақ, төмендегі, 24 суреттегі парақ ашылады. Бұл жерде жаңа мақала енгізуге немесе мақаланы өзгертуге болады. Оң жағында «Постер» батырмасын бассақ, мақаланың суретін жүктеуге болады.

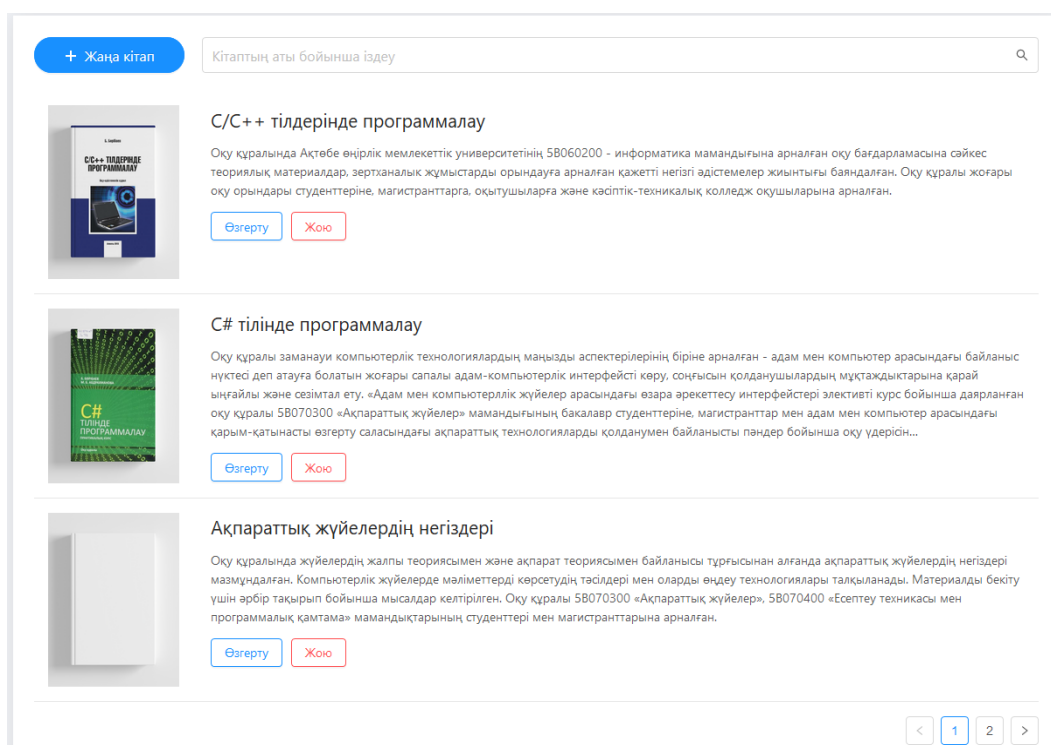
Мақаланың мәтінін енгізу үшін TinyMCE деп аталатын WYSIWIG редакторы қолданылған. Ол жайлы қысқаша 1.4 бөлімде айтылып кеткен. Жалпы айтқанда бұл редактор Microsoft Word-тың веб-нұсқасы деп айтуға болады.

Бұл жерде мәтіннің жуандығын, түсін таңдауға болады, параграф, тақырыптар, тізімдер, суреттер, видеолар енгізуге болады.



24 сурет – Мақаланы өзгерту және енгізу парағы

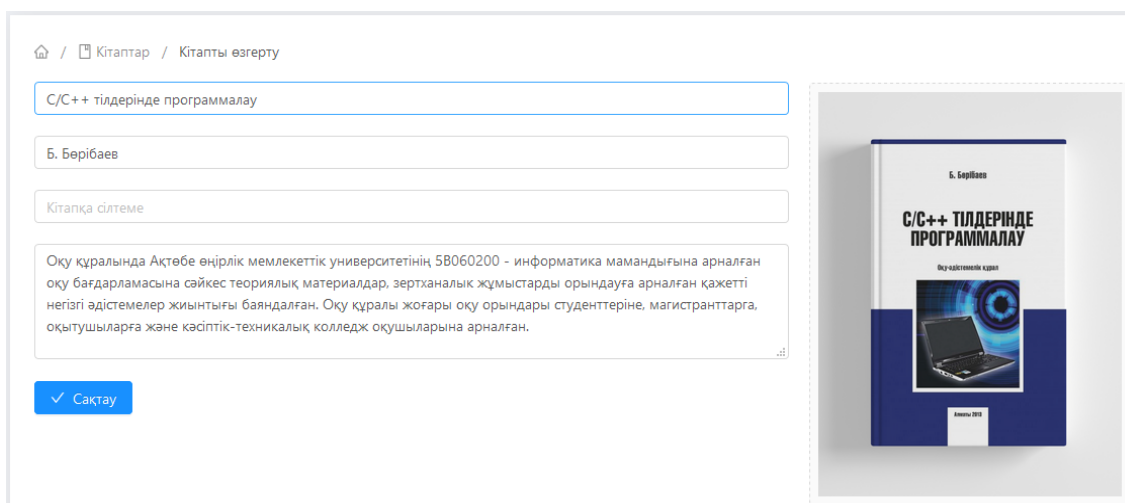
Кітаптар парағы мақалалар парағына өте ұқсас. Бұл жерде кітаптың суреті, атауы, қысқаша сипаттамасы көрсетілген. Оған қоса жаңа кітап еңгізу, кітапты жою, өзгерту батырмалары және кітапты аты бойынша іздеу формасы көрсетілген. Ол 25 суретте көрсетілген.



25 сурет – Кітаптар тізімі

Кітапты өзгерту немесе кітап еңгізу парағында кітап аты, авторы, сипаттамасы, сілтемесі және суреті таңдалады. Кейін 26 суреттегі «Сақтау» батырмасын басу қажет.

Кітаптың сипаттамасын еңгізу үшін TinyMCE редакторы қолданылмаған.



26 сурет – Кітапты өзгерту және еңгізу парағы

3 Жобаның экономикалық тиімділігін негіздеу

3.1 Бағдарламаны еңгізудің экономикалық тиімділігін есептеу

Веб-сайт шығындарының ең негізгісін хостинг пен домен құрайды. Веб-сайтты хостингке орналастыру Қазақстанның ең танымал үш хостингінің бағалары мен ұсынатын қызметтерін салыстыру жүзінде таңдалды. Оларға: hoster.kz, host.kz және ps.kz (nurhost) жатады.

Хостингтерді салыстыруда дискідегі орыны 500 Мб-тан кем емес тарифтері қарастырылды.

1 кесте – Қазақстан хостингтерін салыстыру

Хостинг	Дискідегі орын (Мб)	Пошта саны	MySQL ДБ саны	Қосымша сыйлықтар	Бір айдағы бағасы (тг)	Бір жылдағы бағасы (тг)
Hoster.kz	500	∞	∞	Бір жылға .kz домені	500	6000
Host.kz	500	50	5	–	700	7000 (16% жеңілдік)
PS.kz	500	∞	∞	SSL қауіпсіздік протоколы	600	6480 (10% жеңілдік)

1-кестеде көрсетілген хостингтердің ішінде баға мен сапа жағынан hoster.kz және ps.kz өте тиімді. Host.kz хостингі ең қымбат болғанның өзінде, ұсынатын қызметтері бойынша өзінің бәсекелестерінен біршама ұтылады. Оған қоса, оларда қосымша ешқандай сыйлықтар сыйламайды. Hoster.kz және ps.kz салыстыратын болсақ, олардың тек қана екі айырмашылығы бар. Ол: бағасы мен қосымша сыйлығы. Оған қоса hoster.kz хостингіне қарағанда, ps.kz хостингті бір жылға тапсырыс бергенде 10% жеңілдік жасайды.

SSL (Secure Sockets Layer – қорғаулы сокеттер деңгейі) – аса қауіпсіз байланысты түсіндіретін криптографиялық протокол. Ол алмасу кілттерін аутентификациялау үшін ассиметриялық криптографияны, құпиялықты сақтау үшін симметриялық шифрлеуді, хаттардың бүтіндігі үшін хаттарды аутентификациялау кодтарын қолданады.

PS.kz сайтынан Symantec компаниясы ұсынатын SSL-сертификаттарының бағасы жылына 178000 теңгеден бастап 620000 теңгеге дейін жетеді.

Қазақстанның домен тіркеуші ұйымдарының көбінде .kz доменінің бір жылдағы бағасы 3360 теңге. PS.kz домен тіркеуші және хостинг ұсынушы ұйымының сайтындағы .kz, .com.kz, .org.kz, .kaz домендерінің бағасы жылына 3388 теңге тұрады.

Ал SSL-сертификаттарының ең арзаны 150 мың теңге шамасында болады. Сондықтан PS.kz хостингінің B-1 тарифі қолданылады. Хостингтің B-1

тарифінің сипаттамасы мен бағасы 27-суретте көрсетілген. Оған қоса, hoster.kz пен ps.kz хостингтерінің бір жылдағы бағаларының арасындағы айырмашылық көп емес.

	B-1	B-2	B-3
SSL бесплатно для всех тарифных планов	600 тг/мес	1 200 тг/мес	1 560 тг/мес
Место на диске	500 MB	1000 MB	1500 MB
Количество сайтов	5	10	15
Баз данных MySQL	∞	∞	∞
Почтовых ящиков	∞	∞	∞
	Заказать	Заказать	Заказать
За год	6 480 тг/год экономия 720 тг	12 960 тг/год экономия 1440 тг	16 850 тг/год экономия 1870 тг

27 сурет – PS.kz хостерінің хостинг тарифтерінің сипаттамасы

Веб-сайтты Интернет желісіне ұйымдастыру үшін 9868 теңге жұмсалады. Қанша ақша қайда жұмсалғанын 2-кестеден көрсеніздер болады.

2 кесте – Веб-сайтты жобалауға кеткен шығын

Атауы	Уақыты	Бағасы (тг)
Хостинг	1 жыл	6480
Домен	1 жыл	3388
Барлығы	1 жыл	9868

Хостинг пен доменді бір жыл өткен соң уақытын созу керек. Оларды тағы бір жылға созу үшін 9868 теңге жұмсалады.

ҚОРЫТЫНДЫ

Дипломдық жұмыс барысында түрлі заманауи технологиялар қолданыла отырып ИТ технологияларды қазақ тілінде үйренетін веб-платформа жобаланды. Қолданылған технологияларға: React, Angular, Node.js, Nest.js, TypeORM, TypeScript, TinyMCE және т.б. жатады.

Дипломдық жұмыстың алдында ең басты мәселе – бұл қазақ тілінде білім ұсыну болып табылды. Себебі, қазіргі уақытта технологиялардың жылдан жылға қарқынды даму барысында, білім тез ескіреді. Сондықтан осындай уақытта біз де өз тілімізде білім санын арттыруымыз қажет.

Бұл мәселені шешу үшін алдымен басқа елдердегі, басқа тілдерде ұсынылған білім беру сайттары қарастырылды. Олардың ішінде: coursera, udemy, geekbrains, metanit, habrahabr, tproger, proglib, techcrunch, toster.ru, stackoverflow және т.б сайттар кіреді. Бұл сайттардың барлығының ерекшеліктері ескеріле отырып, жақсы жақтары есепке алынды. Дипломдық жобаның нәтижесінде жобаланған веб-ресурстың ішінде осы сайттардың әр қайсысының бір бөлігі бар деп те айтуға болады. Мысалға, habrahabr сайтынан тек жаңалықтар мен мақалалар оқуға болса, stackoverflow сайтында тек сұрақ қоюға болса, дипломдық жобада бұл сайттың екеуінің де ерекшелігі бар.

Басқа тілдегі сайттар зерттелгеннен кейін пәндік аймақ зерттеліп, оған байланысты ақпарат жинақталды. Веб-сайт пен бағдарламаны құру үшін жаңа веб-технологиялар зерттелді және оларды веб-сайтқа еңгізу қарастылды.

Пәндік аймақ, жаңа технологиялар және басқа да жобаға керекті мәліметтер жинақталғаннан кейін веб-сайт пен деректер базасы концептуалды жобаланды (ER-модель, шектеулер, қолданушылар тізімі және т.б.), деректер базасы машиналық жобаланды және ол жинақталған мәліметпен толтырылды.

Деректер базасы құрылғаннан кейін зерттелген жаңа технологияларды қолдана отырып веб-сайт жобаланды. Жобаланған веб-сайт тестілеуден өтті және қателері табылып, түзетілді.

Қорытындылай келе, дипломдық жұмыс барысында жобаланған веб-сайт бұл үлкен мәселені толығымен қанағаттандырмаса да, бұл мәселені шешудің дұрыс жолын көрсетті. Біздің тіліміз дамып, жаңа мықты мамандар саны арту үшін ең алдымен мемлекеттік тілдегі білімге көп мән беруіміз керек. Егер бұл жобаны ары қарай жетілдіретін болса, бұл білім беру саласында өте пайдасы зор платформалардың бірі болып саналады. Егер тілді дамымаса, мемлекет те дамымайды.

Ал жалпы өзім туралы айтатын болсақ, мен дипломдық жұмысты жасау барысында ЖОО-нда алған барлық білімімді сұрыптап, естен кеткенді еске түсіріп, жаңадан заманауи, жұмыс орындарында қажетті технологияларды меңгердім.

ПАЙДАЛАНЫЛҒАН ӘДЕБИЕТТЕР ТІЗІМІ

- 1 Васвани В.И. MySQL: использование и администрирование. – М.: Питер, 2011. – 368 с.
- 2 Node.js: самые трендовые инструменты на 2019 год http://suhorukov.com/news_akademy/nodejs-samyue-trendovyye-instrumenty-na-2019-god.
- 3 Воройский Ф.С. Информатика: энциклопедический систематизированный словарь-справочник. – М.: Физмалит, 2006. – 945 с.
- 4 Рейтинг языков программирования № 7 // Сообщество программистов DOU. [2005—2020]. URL: <https://dou.ua/lenta/articles/language-rating-jan-2016/> (дата обращения: 14.04.2020).
- 5 August 2011 Web Server Survey // NetCraft. [1995—2020]. Дата обновления: 05.08.2011. URL: <https://goo.gl/Y3Kyd6> (дата обращения: 14.04.2020).
- 6 CSS // Википедия. [2020—2020]. Дата обновления: 08.05.2020. URL: <http://ru.wikipedia.org/?oldid=85868641> (дата обращения: 08.05.2020).
- 7 HTML // Википедия. [2020—2020]. Дата обновления: 26.02.2020. URL: <http://ru.wikipedia.org/?oldid=83906736> (дата обращения: 26.02.2020).
- 8 Мардан Азат. React быстро. Веб-приложения на React, JSX, Redux и GraphQL. — СПб.: «Питер», 2019. — С. 560. — ISBN 978-5-4461-0952-4.
- 9 Бэнкс Алекс, Порселло Ева. GraphQL: язык запросов для современных веб-приложений. — СПб.: «Питер», 2019. — С. 240. — ISBN 978-5-4461-1143-5.
- 10 JavaScript // Википедия. [2020—2020]. Дата обновления: 06.04.2020. URL: <http://ru.wikipedia.org/?oldid=85831928> (дата обращения: 06.04.2020).
- 11 SQL // Википедия. [2020—2020]. Дата обновления: 12.05.2020. URL: <http://ru.wikipedia.org/?oldid=85930735> (дата обращения: 12.05.2020).

Қосымша А

Веб-қосымша листингі

```
import { Module } from '@nestjs/common';
import { TypeOrmModule } from '@nestjs/typeorm';
import { MulterModule } from '@nestjs/platform-express';
import { ServeStaticModule } from '@nestjs/serve-static';
import { join } from 'path';

import { AppController } from './app.controller';
import { AppService } from './app.service';

// Modules
import { CourseModule } from './salem/course.module';

// Entities
import { Course } from './salem/entities/Course';
import { Section } from './salem/entities/Section';
import { Chapter } from './salem/entities/Chapter';
import { User } from './salem/entities/User';
import { Progress } from './salem/entities/Progress';
import { Book } from './salem/entities/Book';
import { Blog } from './salem/entities/Blog';
import { Comment } from './salem/entities/Comment';
import { QuestionCategory } from './salem/entities/QuestionCategory';
import { Question } from './salem/entities/Question';
import { Test } from './salem/entities/Test';
import { Liked } from './salem/entities/Liked';

@Module({
  imports: [
    TypeOrmModule.forRoot({
      type: 'mysql',
      host: '127.0.0.1',
      username: 'root',
      password: '',
      database: 'diplom',
      entities: [
        Course, Section, Chapter,
        User, Progress, Book, Blog,
        Comment, QuestionCategory,
        Question, Test, Liked
      ],
      synchronize: true,
```



```

    }),
    CourseModule,
    MulterModule.register({
      dest: './uploads'
    }),
    ServeStaticModule.forRoot({
      rootPath: join(__dirname, '..', 'uploads')
    })
  ],
  controllers: [AppController],
  providers: [AppService],
})
export class AppModule {}
import { Controller, Get, Post, Body, Param, Put, Delete } from "@nestjs/common";
import { CourseService } from "../services/course.service";
import { SectionService } from "../services/section.service";
import { TestService } from "../services/test.service";

@Controller('courses')
export class CoursesController {

  constructor(
    private readonly courseService: CourseService,
    private readonly sectionService: SectionService,
    private readonly testService: TestService,
  ) {}

  @Get()
  getAll() {
    return this.courseService.getAll();
  }

  @Get(':url')
  getByUrl(@Param('url') url: string) {
    return this.courseService.getOne({ where: { url: url } });
  }

  @Get(':id/sections')
  getSections(@Param('id') id: number) {
    return this.sectionService.getAll({ where: { courseId: id } });
  }

  @Get(':id/tests')
  getTests(@Param('id') id: number) {

```

```

    return this.testService.getAll({ where: { courseId: id } });
  }

  @Post()
  createCourse(@Body() course) {
    return this.courseService.create(course);
  }

  @Put('/:id')
  updateCourse(@Param('id') id: number, @Body() course) {
    return this.courseService.update(id, course);
  }

  @Delete('/:id')
  deleteCourse(@Param('id') id: number) {
    return this.courseService.delete(id);
  }
}

import { Entity, PrimaryGeneratedColumn, Column, OneToMany } from "typeorm";
import { Section } from "./Section";
import { Progress } from "./Progress";

@Entity({name: 'courses'})
export class Course {

  @PrimaryGeneratedColumn()
  id: number;

  @Column()
  name: string; // Название курса. Н/р: JavaScript

  @Column()
  description: string; // Описание курса

  @Column()
  poster: string; // Изображение курса

  @Column()
  url: string; // Коротка ссылка на курс

  @OneToMany(type => Section, section => section.courseId) //+
  sections: Promise<Section[]>;

```

```

    @OneToMany(type => Progress, progress => progress.courseId) //+
    progress: Promise<Progress[]>;
  }
import { Injectable } from "@nestjs/common";
import { InjectRepository } from "@nestjs/typeorm";
import { Repository } from "typeorm";
import { Course } from "../entities/Course";

@Injectable()
export class CourseService {

  constructor(
    @InjectRepository(Course)
    private readonly repository: Repository<Course>
  ) {}

  // GET ALL IN TABLE
  getAll(criteria = {}) {
    return this.repository.find(criteria);
  }

  // GET BY ID
  getById(id: number) {
    return this.repository.findOne(id);
  }

  getOne(criteria = {}) {
    return this.repository.findOne(criteria);
  }

  // INSERT INTO
  create(data: Course) {
    return this.repository.save(data);
  }

  // UPDATE
  update(id: number, data: Partial<Course>) {
    return this.repository.update(id, data);
  }

  // DELETE BY ID
  delete(id: number) {
    return this.repository.delete(id);
  }
}

```

```

}
import { extname } from 'path';
import { uuid } from 'uuidv4';

// Filter files by extension
export const imageFileFilter = (req, file, callback) => {
  if(!file.originalname.match(/\.(jpg|jpeg|png|gif)$/)) {
    return callback(new Error('Only image files allowed'), false);
  }
  callback(null, true);
}

// Unique name for file
export const editFileName = (req, file, callback) => {
  const fileExtName = extname(file.originalname); // extension
  const randomName = uuid(); // random 5592268923
  callback(null, `${randomName}${fileExtName}`); // 5592268923.jpg
}
import { Module } from "@nestjs/common";
import { TypeOrmModule } from "@nestjs/typeorm";

// Entities
import { Course } from "./entities/Course";
import { Section } from "./entities/Section";
import { Chapter } from "./entities/Chapter";
import { User } from "./entities/User";
import { Progress } from "./entities/Progress";
import { Book } from "./entities/Book";
import { Blog } from "./entities/Blog";
import { Comment } from "./entities/Comment";
import { QuestionCategory } from "./entities/QuestionCategory";
import { Question } from "./entities/Question";
import { Test } from "./entities/Test";
import { Liked } from "./entities/Liked";

// Controllers
import { CoursesController } from "./controllers/courses.controller";
import { SectionsController } from "./controllers/sections.controller";
import { ChaptersController } from "./controllers/chapters.controller";
import { UsersController } from "./controllers/users.controller";
import { ProgressController } from "./controllers/progress.controller";
import { BooksController } from "./controllers/books.controller";
import { BlogController } from "./controllers/blog.controller";

```

```

import { UploaderController } from "./controllers/uploader.controller";
import { CommentsController } from "./controllers/comments.controller";
import { QuestionCategoryController } from
"./controllers/question.category.controller";
import { QuestionsController } from "./controllers/questions.controller";
import { TestsController } from "./controllers/tests.controller";
import { LikedController } from "./controllers/liked.controller";

// Services
import { CourseService } from "./services/course.service";
import { SectionService } from "./services/section.service";
import { ChapterService } from "./services/chapter.service";
import { UserService } from "./services/user.service";
import { ProgressService } from "./services/progress.service";
import { BookService } from "./services/book.service";
import { BlogService } from "./services/blog.service";
import { CommentService } from "./services/comment.service";
import { QuestionCategoryService } from "./services/question.category.service";
import { QuestionService } from "./services/question.service";
import { TestService } from "./services/test.service";
import { LikedService } from "./services/liked.service";

@Module({
  imports: [
    TypeOrmModule.forFeature([
      Course, Section, Chapter,
      User, Progress, Book, Blog,
      Comment, QuestionCategory,
      Question, Test, Liked
    ])
  ],
  exports: [TypeOrmModule],
  controllers: [
    CoursesController,
    SectionsController,
    ChaptersController,
    UsersController,
    ProgressController,
    BooksController,
    BlogController,
    UploaderController,
    CommentsController,
    QuestionCategoryController,
    QuestionsController,

```

```
    TestsController,  
    LikedController  
  ],  
  providers: [  
    CourseService,  
    SectionService,  
    ChapterService,  
    UserService,  
    ProgressService,  
    BookService,  
    BlogService,  
    CommentService,  
    QuestionCategoryService,  
    QuestionService,  
    TestService,  
    LikedService  
  ]  
})  
export class CourseModule {}
```