Қазақстан Республикасының Ғылым және жоғары білім министрлігі

Әл-Фараби атындағы Қазақ ұлттық университеті

Эльбрус Е.Э.

APACHE SPARK ПЛАТФОРМАСЫ НЕГІЗІНДЕ АЛМАТЫ ҚАЛАСЫ ТЕАТРЛАРЫНЫҢ СПЕКТАКЛДЕРІНДЕ ТАПСЫРЫСТАРДЫ РЕСІМДЕУ ҮШІН ЗИЯТКЕРЛІК ПОРТАЛ ӘЗІРЛЕУ

ДИПЛОМДЫҚ ЖОБА

Мамандығы 6B06106 – «Жасанды интеллектcі бар жоғары жүктелген ақпараттық жүйелер»

Алматы, 2024

Қазақстан Республикасының Ғылым және жоғары білім министрлігі

Әл-Фараби атындағы Қазақ ұлттық университеті

Ақпараттық технологиялар факультеті

Ақпараттық жүйелер кафедрасы

ДИПЛОМДЫҚ ЖОБА

тақырыбы: «Apache Spark платформасы негізінде Алматы қаласы театрларының спектакльдеріне тапсырыстарды ресімдеу үшін зияткерлік портал әзірлеу»

Мамандығы 6B06106 – «Жасанды интеллектcі бар жоғары жүктелген ақпараттық жүйелер»

Орындаған Эльбрус Е.Э.

(қолы)

Орындаған Келменбетов А.Е.

(қолы)

Ғылыми жетекші Алпысбай Г.Е.

аға оқытушы (қолы)

Қорғауға жіберілді:

Хаттама № \_\_\_ , «\_\_\_»\_\_\_\_\_\_\_ 2024ж.

Кафедра меңгерушісі \_\_\_\_\_\_\_\_\_\_\_\_ Мусиралиева Ш.Ж.

(қолы және мөрі)

Норма бақылаушы Кенжебаева М.O.

(қолы)

Алматы 2024ж

**ТҮЙІНДЕМЕ**

Дипломдық жоба 40 беттен, 19 суреттен және 18 пайдаланылған әдебиеттер тізімінен тұрады.

APACHE SPARK ПЛАТФОРМАСЫ НЕГІЗІНДЕ АЛМАТЫ ҚАЛАСЫ ТЕАТРЛАРЫНЫҢ СПЕКТАКЛЬДЕРІНЕ ТАПСЫРЫСТАРДЫ РЕСІМДЕУ ҮШІН ЗИЯТКЕРЛІК ПОРТАЛ ӘЗІРЛЕУ

***Дипломдық жұмыстың мақсаты:***

Бұл дипломдық жұмыстың мақсаты-пайдаланушыларға темпераментіне, жынысына және жасына қарай Фильмдер ұсынатын ұсыныс жүйесін құру. Жүйе пайдаланушылардың жеке ерекшеліктеріне сәйкес келетін жекелендірілген ұсыныстарды қамтамасыз ету үшін машиналық оқыту және психологиялық профильдеу әдістерін пайдаланады.

Нысан: профильдерін, демографиясын, шолу тарихын және психологиялық сауалнамаларға жауаптарын қамтитын кино ағындық платформаны пайдаланушылар.

Тақырып: пайдаланушылардың темпераментіне, жынысына және жасына негізделген ұсыныс алгоритмдерін әзірлеу және енгізу.

Жоба бағытталған:

Дәл және әртүрлі фильм ұсыныстары үшін бірлескен сүзуді, мазмұнға негізделген сүзуді және демографиялық сүзуді біріктіретін гибридті ұсыныс жүйесін әзірлеу.

Интроверсия, экстраверсия, эмоционалды тұрақтылық және тәжірибеге ашықтық сияқты пайдаланушылардың темпераменттік сипаттамаларын бағалау үшін психологиялық сауалнаманы біріктіру.

Пайдаланушыларға ұсынылған фильмдерді бағалауға және шолуға мүмкіндік беретін динамикалық кері байланыс механизмін енгізу, бұл ұсыныс алгоритмдерінің үздіксіз жақсаруына ықпал етеді.

Нәтижелері:

* Пайдаланушылардың темпераментіне, жынысына және жасына негізделген фильмдерді дәл болжауға және ұсынуға қабілетті толыққанды ұсыныс жүйесін әзірлеу.
* Жоғары сәйкес және жекелендірілген ұсыныстар беру арқылы пайдаланушылардың қанағаттанушылығын арттыру.
* Пайдаланушыларды мазмұнға терең тарту, бұл фильмдерді көрудің жекелендірілген және жағымды тәжірибесіне әкеледі.

**РЕФЕРАТ**

Дипломная работа состоит из 40 страниц, 19 рисунков и 18 источников.

РАЗРАБОТКА ИНТЕЛЛЕКТУАЛЬНОГО ПОРТАЛА ДЛЯ ОФОРМЛЕНИЯ ЗАКАЗОВ НА СПЕКТАКЛИ ТЕАТРОВ ГОРОДА АЛМАТЫ НА ОСНОВЕ ПЛАТФОРМЫ APACHE SPARK

***Цель дипломной работы:***

Целью данной дипломной работы является создание рекомендательной системы, которая будет предлагать фильмы пользователям на основе их темперамента, пола и возраста. Система использует методы машинного обучения и психологического профилирования для предоставления персонализированных рекомендаций, соответствующих индивидуальным особенностям пользователей.

Объект: Пользователи кино-стриминговой платформы, включающие их профили, демографические данные, историю просмотров и ответы на психологические опросы.

Предмет: Разработка и внедрение алгоритмов рекомендаций, основанных на темпераменте, поле и возрасте пользователей.

Проект направлен на:

Разработку гибридной рекомендательной системы, объединяющей коллаборативную фильтрацию, фильтрацию на основе контента и демографическую фильтрацию для точных и разнообразных рекомендаций фильмов.

Интеграцию психологического опроса для оценки темпераментных характеристик пользователей, таких как интроверсия, экстраверсия, эмоциональная стабильность и открытость к опыту.

Внедрение динамического механизма обратной связи, позволяющего пользователям оценивать и рецензировать рекомендованные фильмы, что способствует непрерывному улучшению алгоритмов рекомендаций.

Полученные результаты:

* Разработка полнофункциональной рекомендательной системы, способной точно предсказывать и предлагать фильмы на основе темперамента, пола и возраста пользователей.
* Повышение удовлетворенности пользователей за счет предоставления высоко релевантных и персонализированных рекомендаций.
* Углубленное вовлечение пользователей в контент, что ведет к более персонализированному и приятному опыту просмотра фильмов.

**ABSTRACT**

Diploma work consists of 40 pages, 19 pictures and 18 sources.

DEVELOPMENT OF AN INTELLIGENT PORTAL FOR PLACING ORDERS FOR PERFORMANCES OF THEATERS IN ALMATY BASED ON THE APACHE SPARK PLATFORM

***The purpose of the thesis:***

The purpose of this thesis is to create a recommendation system that will offer films to users based on their temperament, gender and age. The system uses machine learning and psychological profiling techniques to provide personalized recommendations that match the individual characteristics of users.

Object: Users of the movie streaming platform, including their profiles, demographic data, viewing history and responses to psychological surveys.

Subject: Development and implementation of recommendation algorithms based on the temperament, gender and age of users.

The project aims to:

Develop a hybrid recommendation system combining collaborative filtering, content-based filtering and demographic filtering for accurate and diverse movie recommendations.

Integration of a psychological survey to assess the temperamental characteristics of users, such as introversion, extraversion, emotional stability and openness to experience.

The introduction of a dynamic feedback mechanism that allows users to evaluate and review recommended films, which contributes to the continuous improvement of recommendation algorithms.

Results:

* Development of a fully functional recommendation system capable of accurately predicting and suggesting movies based on the temperament, gender and age of users.
* Increase user satisfaction by providing highly relevant and personalized recommendations.
* In-depth user engagement in content, leading to a more personalized and enjoyable movie watching experience.

Мазмұны

[Кіріспе 7](#_Toc168305771)

[1 ЗИЯТКЕРЛІК БАҒДАРЛАМАҒА КІРІСПЕ 9](#_Toc168305772)

[1.1 Жүйені интеграциялау және пайдалану 9](#_Toc168305773)

[1.2 Ұсыныс жүйесін веб-қосымшаға интеграциялау 13](#_Toc168305774)

[1.3 Жүйедегі тапсырыс бетін жасау 14](#_Toc168305775)

[1.3.1 Тіркелу және авторизация 15](#_Toc168305776)

[1.3.3 Тапсырмаларды иерархиялық талдау 19](#_Toc168305777)

[2 КОНТЕЙНЕРЛІК ТЕХНОЛОГИЯНЫ ҚОЛДАНУ ЕРЕЖЕЛЕРІ 21](#_Toc168305778)

[2.1 Бір түйінді желі 22](#_Toc168305779)

[2.2 Kubernetes оркестрациясын кірістіру технологиясы 25](#_Toc168305780)

[2.2.1 Dockerfile және minikube қолдану принциптері 26](#_Toc168305781)

[2.2.2 Автоматты масштабтау және конфигурацияны басқару 27](#_Toc168305782)

[2.2.3 Нұсқалар мен жаңартуларды басқару 29](#_Toc168305783)

[2.2.4 Желілерді Басқару 30](#_Toc168305784)

[3 UML ДИАГРАММАЛАРЫН ПАЙДАЛАНУ 31](#_Toc168305785)

[3.1 Класстар диаграммасы 31](#_Toc168305786)

[3.2 Компоненттер диаграммасы 33](#_Toc168305787)

[3.3 Реттілік диаграммасы 35](#_Toc168305788)

[Қорытынды 37](#_Toc168305789)

[Пайдаланылған әдебиеттер тізімі 38](#_Toc168305790)

[Қосымша А 39](#_Toc168305791)

# **Кіріспе**

Қазіргі заман оларды ұйымдастырудың және қол жеткізудің жаңа тәсілдерін қажет ететін ақпарат пен деректер көлемінің қарқынды өсуімен сипатталады. Бұл әсіресе Мәдени іс-шаралар саласында өзекті болып отыр, мұнда пайдаланушылар жеке ұсыныстар мен билеттерге тапсырыс беру үшін ыңғайлы қызмет күтеді. Осыған байланысты бұл зерттеу Apache Spark платформасы негізінде Алматы қаласындағы театрлардың спектакльдеріне тапсырыстарды ресімдеу үшін зияткерлік портал әзірлеуге арналған.

Бұл жобаның негізгі мақсаты нақты уақыт режимінде жекелендірілген ұсыныстар бере алатын жоғары өнімді, масштабталатын және сенімді жүйені құру болып табылады. Бұл нұсқаулар пайдаланушының көңіл-күйі мен темпераментін талдауға негізделген, бұл пайдаланушы тәжірибесін айтарлықтай жақсартуға мүмкіндік береді. Микросервистерді басқару және шешімнің икемділігі мен тұрақтылығын қамтамасыз ету үшін Kubernetes, ал деректерді тиімді талдау үшін Apache Spark қолданылады.

Мұндай порталды әзірлеу бірнеше негізгі кезеңдерді қамтиды. Біріншіден, ұсыныстарды біріктіру және билеттерге тапсырыс беру функционалдығы бар веб-қосымшаны құру. Екіншіден, жүйенің тұрақты және икемді жұмысын қамтамасыз ететін заманауи контейнерлеу және оркестрлеу технологияларын қолдану. Маңыздысы, Kubernetes және Apache Spark пайдалану деректерді тиімді масштабтауға және талдауға мүмкіндік береді, бұл үлкен көлемдегі ақпаратпен жұмыс істеу үшін өте маңызды.

Мәдени іс-шаралар саласында жекелендіру үлкен маңызға ие. Қазіргі қолданушылар өздерінің қалауы мен қазіргі эмоционалды жағдайларын ескере отырып, жеке көзқарасты күтеді. Осы зерттеу аясында әзірленген интеллектуалды портал пайдаланушыларға ең маңызды ұсыныстарды ұсына отырып және олардың қызметпен өзара әрекеттесуін жақсарта отырып, осы факторларды ескеруге мүмкіндік береді. Осылайша, бұл жүйе пайдаланушылардың қанағаттану деңгейін арттырып қана қоймай, мәдени іс-шараларға тапсырыстар санының артуына ықпал етеді.

Жобаны сәтті жүзеге асыру үшін жүйенің архитектурасын нақты құрылымдауға және негізгі компоненттер мен олардың өзара әрекеттесуін визуализациялауға көмектесетін UML диаграммалары қолданылады. Бұл жүйені әзірлеу мен енгізуді тиімді жоспарлауға мүмкіндік береді, сонымен қатар оны одан әрі дамыту мен қолдауға мүмкіндік береді.

Жоба сонымен қатар веб-қосымшаларға ұсыныс жүйелерін біріктіруді, пайдаланушыларды тіркеу және авторизациялау үшін беттер құруды, сеанс кестесін қарауды және залдағы орындарды таңдауды қарастырады. Docker және Kubernetes пайдалану микросервистерді контейнерлеуге және оркестрлеуге мүмкіндік береді, бұл жүйені икемді және ауқымды етеді. Бұл тұрғыда, деректер қауіпсіздігі және тиімді конфигурацияны басқару мәселелеріне ерекше назар аударылады, бұл жүйенің сенімді жұмыс істеуі үшін өте маңызды. Apache Spark және Kubernetes базасында театр қойылымдарына тапсырыстарды рәсімдеу үшін зияткерлік порталды әзірлеу және енгізу Мәдени іс-шаралар саласындағы цифрлық шешімдерді дамытуға елеулі үлес болып табылады. Бұл жүйе пайдаланушы тәжірибесін жақсартып қана қоймайды, сонымен қатар осы салада ұсынылатын қызметтердің тиімділігі мен жекелендірілуіне айтарлықтай үлес қосады.

**1 ЗИЯТКЕРЛІК БАҒДАРЛАМАҒА КІРІСПЕ**

Қазіргі дәуір ақпарат пен білім көлемінің тез өсуімен сипатталады, оларды ұйымдастырудың және қол жеткізудің жаңа тәсілдерін қажет етеді. Жаһандану және технологиялардың кеңінен енуі жағдайында білімнің әртүрлі салаларын біріктіруге және пайдаланушыларға өзекті және дәлелденген ақпаратқа қол жеткізуге мүмкіндік беретін интеграциялық платформаларды құру өзекті болып отыр.

Интеллектуалды порталдар мұндай интеграцияны қамтамасыз етуде маңызды рөл атқарады. Бұл платформалар жеке пәндер бойынша білімді тереңдетуге ықпал етіп қана қоймайды, сонымен қатар пәнаралық зерттеулер үшін жаңа көкжиектер ашады. Интеллектуалды порталдарда жасанды интеллект технологияларын пайдалану пайдаланушыларға бейімделген ұсыныстар мен ақпаратты терең талдауды қамтамасыз ету арқылы деректерді іздеу және талдау процесін жекелендіруге мүмкіндік береді. Бұл өз кезегінде білімді тиімді оқытуға және практикалық қолдануға ықпал етеді.

Мұндай платформалардың негізгі міндеті ақпараттың қол жетімділігі мен түсінікті болуын қамтамасыз ету болып табылады. Материалдардың күрделілігі мен қол жетімсіздігіне байланысты кедергілерді жою маңызды. Заманауи интеллектуалды порталдар ұсынатын интерактивті элементтер мен құралдар күрделі тұжырымдамаларды түсінуді жеңілдетеді және пайдаланушыларға материалдармен тереңірек байланыс орнатуға мүмкіндік береді.

Интеллектуалды порталдар таным мен білім алмасу саласындағы ортақ мүдделері бар адамдарды біріктіретін динамикалық қауымдастықтар жасайды. Пікірталастарға қатысу, Бірлескен зерттеу және идеялармен бөлісу ұжымдық интеллект пен жаңа білімді қалыптастыруға ықпал етеді. Пайдаланушылар ақпаратқа қол жеткізіп қана қоймай, порталдардың дамуына өз табыстары мен жаңалықтарымен бөлісе алады.

Осылайша, интеллектуалды порталдар үнемі даму мен танымға ұмтылатындар үшін маңызды құралға айналады. Олар ғылым өнермен кездесетін және технология гуманитарлық ғылымдармен араласатын бірегей білім экожүйелерін жасайды. Бұл платформалар жаңа білім құруға және оларды әртүрлі салаларда тиімді қолдануға ықпал етеді, зерттеу мен инновацияның шексіз мүмкіндіктерін ашады. Платформаның құрылу негізіне веб бағдарламаның: frontend және backend бөліктері кіреді.

**1.1 Жүйені интеграциялау және пайдалану**

Веб-сайтты құру-бұл даму мен интеграцияның көптеген аспектілерін қамтитын күрделі процесс. Веб-сайтты белгілі бір аудитория үшін ақпаратты, интерактивті элементтерді және қызметтерді біріктіретін кешенді портал ретінде қарастыруға болады. Бұл процесс пайдаланушының жүйемен визуалды және функционалды өзара әрекеттесуін қамтамасыз ететін алдыңғы жағын жобалау мен дамытудан басталады.

Алдыңғы жағын әзірлеу Басты бет, байланыс беті және ақпарат беті сияқты әртүрлі беттерді жасауды қамтиды. Бұл беттер бір функционалды жүйеге біріктіріліп, пайдаланушыларға оңай навигация мен қол жетімділікті қамтамасыз етеді. Әрбір бет сәтті веб-жобаның негізгі факторы болып табылатын ыңғайлылық пен интуитивтілік принциптерін ескере отырып жасалған. Пайдаланушылар сайтпен ұсынылған қызметтер туралы ақпарат алу немесе байланыс парағы арқылы әкімшілікпен байланысу арқылы байланысады.

Дамудың техникалық аспектілері сайтты әр түрлі құрылғылар мен экрандарға бейімдеуді қамтиды, бұл барлық платформаларда оңтайлы дисплейге мүмкіндік береді. Бұған пайдаланушы құрылғысының экран өлшемдеріне интерфейс элементтерін автоматты түрде реттейтін жауапты дизайн арқылы қол жеткізіледі. Беттерді жылдам жүктеуге және пайдаланушы тәжірибесін жақсартуға ықпал ететін сайттың жұмысын оңтайландыру да маңызды.

Веб-сайтты құру сонымен қатар болашақ қажеттіліктер мен технологиялық тенденцияларды ескеруді талап етеді. Сайтты үнемі жаңарту және қолдау тек техникалық қызмет көрсетуді ғана емес, сонымен қатар мазмұнды үнемі жаңартуды да қамтиды. Бұл ақпараттың өзектілігін сақтау және жаңа келушілерді тарту үшін қажет. Сайтты одан әрі дамыту жоспарлары қосымша қызметтерді біріктіруді немесе жаңа технологиялар арқылы пайдаланушы тәжірибесін жақсартуды қамтуы мүмкін. Бұл жобаның өзгерістерге бейімделу және пайдаланушылардың қажеттіліктеріне сәйкес даму қабілетін көрсетеді. Веб-сайт тек ақпараттық платформа ғана емес, сонымен қатар технологиялық прогресс пен аудиторияның өзгеретін қажеттіліктерімен бірге дамитын тірі құрал болып табылады. [8]

Изображение выглядит как текст, снимок экрана, Мультимедийное программное обеспечение, программное обеспечение

Автоматически созданное описание

1.1 сурет. Жүйенің басты беті

Сайттың барлық элементтерін біріктіру-ақпараттық беттер, байланыс формалары, кіру және тіркеу функциялары — пайдаланушыға логикалық түрде құрылған және оңай шарлау тәжірибесін ұсынады. Бұл веб-жобаларды құру және басқару процестерін терең түсінуді көрсете отырып, веб-жүйелерді жобалауға және қолдауға кешенді көзқарастың маңыздылығын көрсетеді.

Веб-сайттың басты беті оның барлық функционалдығына негіз болады. Негізгі элементтердің бірі-пайдаланушы тәжірибесін жақсарту үшін біріктірілген ұсыныс жүйесі. Қазіргі цифрлық әлемде қол жетімді мазмұн күн сайын өсіп келе жатқанда, пайдаланушыларға таңдау процесін жеңілдететін құралдарды ұсыну маңызды. Сауалнама нәтижелерін пайдаланатын Нұсқаулық жүйесі келушілердің жеке қалауына сәйкес келетін фильмдерді ұсынуға мүмкіндік береді.

Мұндай жүйені дамыту пайдаланушылардың жауаптарын талдай алатын және олардың негізінде жеке ұсыныстар жасай алатын күрделі алгоритмдерді құруды талап етеді. Бұл тұрғыда ұсыныстардың жоғары дәлдігін қамтамасыз ете алатын Машиналық оқыту және деректерді талдау әдістері қарастырылады. Алгоритмдер ең маңызды мазмұнды ұсыну үшін пайдаланушының қалауы, алдыңғы фильм ұпайлары және сайттағы мінез-құлық сияқты көптеген факторларды ескереді.

Басты бет портал туралы пікірлер мен пікірлер қалдыруға мүмкіндік береді. Бұл функционалдылық пайдаланушыларға өз тәжірибелерімен және пікірлерімен бөлісуге, сондай-ақ Сайт әкімшілігімен өзара әрекеттесуге мүмкіндік береді. Егер пайдаланушылар проблемаларға тап болса немесе шағымдары болса, олар оларды шешу үшін әкімшілермен тікелей байланыса алады. Бұл пайдаланушы тәжірибесін жақсартып қана қоймайды, сонымен қатар аудиторияның қазіргі тенденциялары мен қалауларын бақылауға көмектеседі.

Ұсыныс жүйесін біріктіру және кері байланыс мүмкіндігі пайдаланушылардың сайтпен өзара әрекеттесуін жақсартуға көмектесетін маңызды элементтер болып табылады. Бұл мүмкіндіктер пайдаланушыларды ұстап қалудың және қанағаттандырудың негізгі факторы болып табылатын жекелендірілген және интерактивті тәжірибені қамтамасыз етеді. Мұндай жүйелерді енгізу мұқият жоспарлауды және озық технологияларды қолдануды талап етеді, бұл веб-сайттарды әзірлеу мен қолдаудың кешенді тәсілінің маңыздылығын көрсетеді. Сайттың басты беті пайдаланушылар үшін кіру нүктесі ретінде ғана емес, сонымен қатар олардың мазмұнмен өзара әрекеттесуін жақсарту үшін маңызды функцияларды орындайды. Бұған Деректерді талдаудың заманауи технологиялары мен әдістерін қолдану арқылы қол жеткізіледі, бұл Пайдаланушының жекелендірілген және тиімді тәжірибесін жасауға мүмкіндік береді.

Изображение выглядит как текст, снимок экрана, Мультимедийное программное обеспечение, программное обеспечение

Автоматически созданное описание

1.2 сурет. Колданушылардың әкімшімен байланысы

Суретте келушілермен байланысты қамтамасыз ету үшін кез келген сайттың пайдаланушы интерфейсінің маңызды құрамдас бөлігі болып табылатын контактілер веб-беті көрсетілген. Бұл бет веб-сайттың бренді мен дизайнын көрсетіп қана қоймайды, сонымен қатар ұйым мен оның аудиториясы арасындағы көпір функциясын орындайды, бұл пайдаланушыларға берілген кері байланыс нысаны арқылы сұрауларды, түсініктемелерді немесе сұрақтарды оңай жіберуге мүмкіндік береді.

Беттің дизайны веб-сайттың корпоративті түстерінде жасалған, бұл брендтің біркелкілігі мен танылуын қамтамасыз етеді. Экранның жоғарғы жағында сайттың басқа бөлімдеріне кіру үшін интуитивті навигация мәзірі бар, бұл пайдаланушыға негізгі бетке оралмай-ақ сайтты шарлау ыңғайлылығын қамтамасыз етеді.

Беттегі орталық орынды аты, тегі, электрондық пошта мекенжайы және хабарламаны енгізуге арналған өрістерден тұратын кері байланыс нысаны алады. Бұл құрылым сайтқа кірушілерге компания өкілдерімен оңай байланысуға мүмкіндік беретін стандартты тәжірибе болып табылады. Кері байланыс формасының астында телефон нөмірі, электрондық пошта мекенжайы және физикалық орналасқан жері бар Байланыс ақпараты орналастырылған. Бұл пайдаланушыларға бірнеше байланыс арналарын беру арқылы сенімділікті арттырады.

Мұндай құрылым және байланыс ақпаратын ұсыну сайтқа кірушілердің оң өзара әрекеттесуді құру үшін қажетті қолдау мен ақпаратты оңай алуын қамтамасыз етеді. Бұл сонымен қатар компанияның немесе сайтты ұйымдастырушылардың өз аудиториясына ашықтығы мен қол жетімділігін көрсетеді. Контактілер парағы тек практикалық ғана емес, сонымен қатар стратегиялық функцияны орындайды, бұл пайдаланушылардың қажеттіліктері мен тілектеріне назар аудару әсерін тудырады. Контактілер беті пайдаланушы тәжірибесін жақсартуда және тұтынушылардың қанағаттану деңгейін арттыруда маңызды рөл атқаратынын ескеру маңызды. Байланыстың бірнеше әдісін ұсыну арқылы пайдаланушылар өздері үшін ең қолайлы байланыс әдісін таңдай алады. Бұл өзара әрекеттесуді жақсартып қана қоймайды, сонымен қатар Компанияның оң имиджін қалыптастыруға ықпал етеді, оның ашық диалогқа және пайдаланушыларда туындайтын мәселелер мен мәселелерді тез шешуге дайын екендігін көрсетеді. Веб-сайттағы контактілер беті пайдаланушы тәжірибесін жақсартуға және брендке деген сенімділікті арттыруға ықпал ететін маңызды элемент болып табылады. Оның құрылымы, дизайны және функционалдығы пайдаланушының оң тәжірибесін құруда және аудиторияның қанағаттану деңгейін арттыруда шешуші рөл атқарады.

## **1.2 Ұсыныс жүйесін веб-қосымшаға интеграциялау**

Қызмет санаттары көрсетілген және машиналық оқытуды қолдана отырып, Фильмдер, спектакльдер және басқа да іс-шараларға ұсыныстар беретін сауалнама жүргізілетін веб-бет көрсетілген. Берілген веб-беттің құрылымы және ұсыныс жүйесінің интеграциясы үздіксіз және интуитивті пайдаланушы интерфейсін қамтамасыз етеді.

Сауалнама нәтижелерін пайдаланатын ұсыныс жүйесі пайдаланушылардың қалауын талдайды және олардың негізінде жеке ұсыныстар береді. Бұл, әсіресе, қол жетімді мазмұнның саны тез өсіп, таңдау процесін қиындататын қазіргі дәуірге қатысты. Машиналық оқыту алгоритмдері пайдаланушылардың жауаптарын талдайды және пайдаланушыларға олардың талғамы мен көңіл-күйіне сәйкес мазмұнды табуға мүмкіндік беретін арнайы ұсыныстар жасайды.

Веб-сайттың құрылымына және ұсыныс жүйесінің оған қалай енетініне ерекше назар аударылады. Бұл навигацияның қарапайымдылығын ғана емес, сонымен қатар пайдаланушылардың сайтты жалпы қабылдауын жақсартады. Жаңа шығарылымдар мен пайдаланушылардың пікірлерін ескере отырып, ұсыныстар жүйесі ұсыныстардың өзектілігін қамтамасыз ете отырып, фильмдер мен басқа да іс-шаралар туралы деректерді үнемі жаңартып отырады.

Веб-сайттың дизайны мен функционалдығы пайдаланушыларды тартуда маңызды рөл атқарады. Көрнекіліктер мен интерактивті элементтер ұсынылған фильмдерді көру және іс-шараларға қатысу туралы шешім қабылдауға ықпал етеді. Бұл таңдау процесін жағымды және аз уақытты қажет етеді, мәдени мазмұнға терең енуге ықпал етеді.

Зерттеу мұндай жүйелердің цифрлық мәдениет пен кинематографияны дамытуға қосқан үлесі туралы ойлаумен аяқталады. Жекелендірілген ұсыныс қызметтері кинематографиялық туындыларды ілгерілету және медиа мазмұнмен өзара әрекеттесу үшін жаңа мүмкіндіктер ашады. Бұл инновациялар жеке және қанағаттанарлық пайдаланушы тәжірибесін ұсына отырып, мазмұнды тұтыну тәсілін өзгерте алады.[7]

Осылайша, веб-сайтта ұсыныстар жүйесін енгізу пайдаланушылардың мазмұнмен өзара әрекеттесуін жақсартып қана қоймайды, сонымен қатар жекелендірілген ұсыныстар мен медиа мазмұнды ілгерілету саласында жаңа перспективалар мен мүмкіндіктерді ұсына отырып, Цифрлық мәдениетті дамытуға ықпал етеді.

Изображение выглядит как текст, снимок экрана, программное обеспечение, Мультимедийное программное обеспечение

Автоматически созданное описание

1.3 сурет. Ұсыныс жүйесі

## **1.3 Жүйедегі тапсырыс бетін жасау**

Заманауи цифрлық технологиялар әртүрлі іс-шараларға, соның ішінде киносеанстарға Билеттерді сатып алу процесін айтарлықтай жеңілдетеді. Онлайн билеттерді брондау платформалары пайдаланушыларға кинотеатр кассаларына баруды болдырмай, фильм билеттерін оңай және ыңғайлы таңдауға және сатып алуға мүмкіндік береді. Бұл жұмыста киносеанстарға билеттерді онлайн-брондау жүйесінің жұмыс істеуі қарастырылады, оның архитектурасы, функционалдығы және қамтамасыз етілетін қауіпсіздік деңгейі сипатталады.

Киносеанстарға билеттерді онлайн брондау жүйесі серверлік және клиенттік бөліктерді қамтитын көп деңгейлі архитектура болып табылады.

Жүйенің архитектурасы заманауи веб-технологияларға негізделген. Сервер бөлігі Node.js фреймворк қолдана отырып жүзеге асырылады. Пайдаланушы сұрауларын өңдеуге және дерекқормен өзара әрекеттесуге жауап беретін фильмдер, сеанстар және брондау деректерін сақтау үшін реляциялық мәліметтер базасы қолданылады. Киносеанстарға билеттерді онлайн брондау жүйесі пайдаланушыларға келесі негізгі функционалдылықты ұсынады:

1. Тіркелу және авторизация: Пайдаланушылар жүйенің функцияларына жеке қол жетімділікті қамтамасыз ететін есептік жазбалар жасай алады. Тіркеу процесі жеке деректерді енгізуді және электрондық поштаны Растауды қамтиды.

2. Сеанстар кестесін қарау: пайдаланушылар ағымдағы киносеанстар кестесін көре алады, сеанстарды фильмнің күні, уақыты және жанры бойынша сүзе алады.

3. Залдағы орындарды таңдау: жүйе пайдаланушыларға фильм көру үшін қажетті орындарды таңдауға мүмкіндік беретін интерактивті зал схемасын ұсынады.

4. Онлайн төлем: қаржылық транзакциялардың ыңғайлылығы мен қауіпсіздігін қамтамасыз ететін Банктік карталар мен электрондық әмияндарды қоса алғанда, әртүрлі төлем әдістеріне қолдау көрсетіледі.

Жүйенің Пайдаланушы интерфейсінің дизайны қазіргі заманғы ыңғайлылық талаптарын ескере отырып жасалған. Интерфейс қарапайым және интуитивті, бұл пайдаланушыларға жүйені оңай басқаруға және қажетсіз күш жұмсамай сатып алуға мүмкіндік береді. Жауап беретін дизайн компьютерлерді, планшеттерді және смартфондарды қоса алғанда, әртүрлі құрылғыларда интерфейстің дұрыс көрсетілуін қамтамасыз етеді.

Пайдаланушы деректерінің қауіпсіздігін қамтамасыз ету билеттерді онлайн брондау жүйесін әзірлеудің негізгі аспектісі болып табылады. Жүйе деректерді шифрлау және екі факторлы аутентификация сияқты ақпаратты қорғаудың заманауи әдістерін қолданады. Бұл шаралар пайдаланушылардың жеке ақпаратына рұқсатсыз қол жеткізуді болдырмауға және қаржылық операциялардың құпиялылығын қамтамасыз етуге бағытталған.

### **1.3.1 Тіркелу және авторизация**

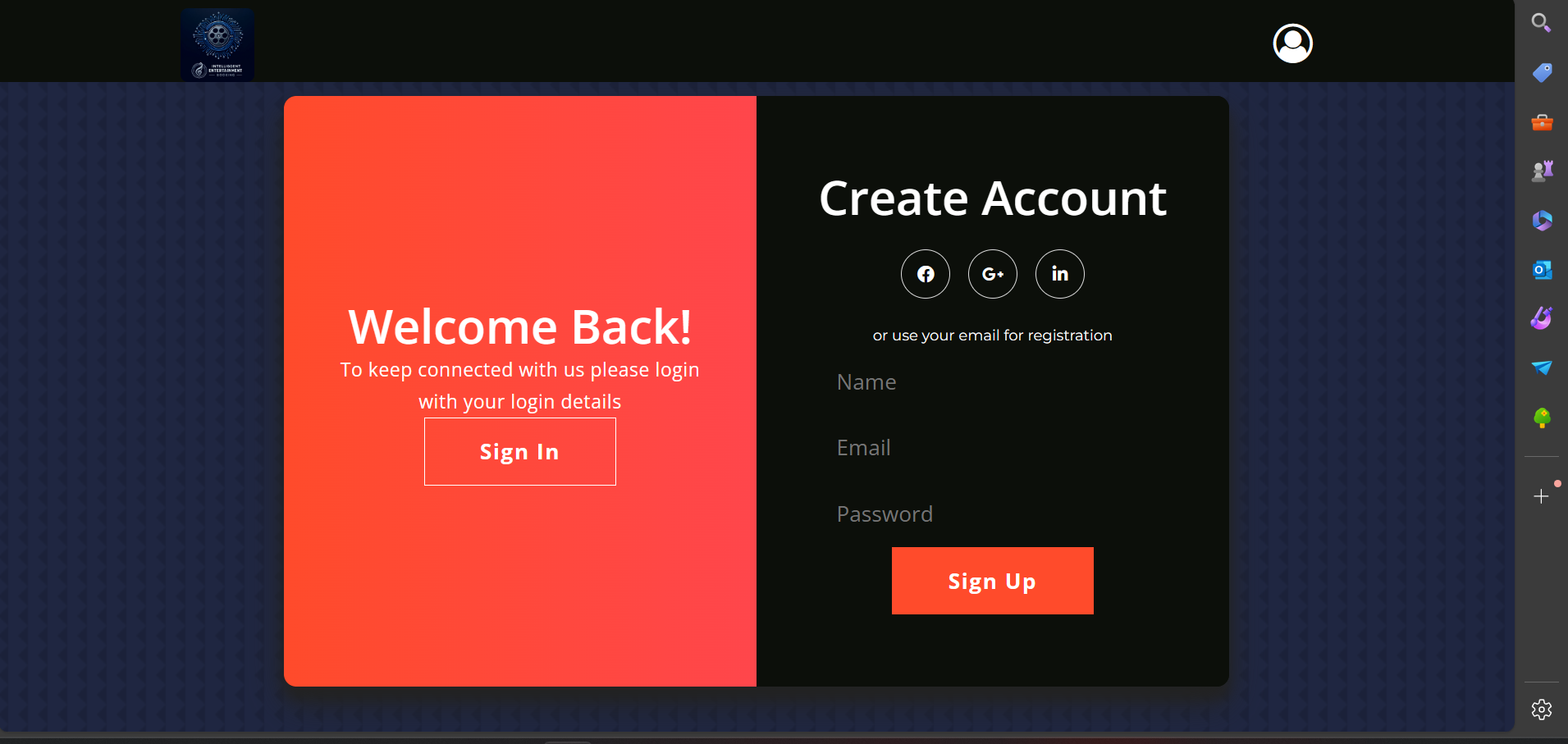
Заманауи веб-қосымшаларда, әсіресе пайдаланушының өзара әрекеттесуін және құпия деректерді өңдеуді қажет ететіндерде, сенімді және қауіпсіз авторизация жүйесін енгізу өте маңызды міндет болып табылады. Онлайн киносеанс билеттерін брондауға арналған Веб-қосымшалар пайдаланушыларға тіркелуге, жүйеге кіруге және жекелендірілген деректер мен қызметтерге қол жеткізуге мүмкіндік беретін авторизация функциясын қамтиды.

Авторизация процесі пайдаланушы аты мен құпия сөз сияқты пайдаланушы тіркелгі деректерін енгізу формасына негізделген. Бұл форма кіру бетінде ұсынылған және логин мен парольді енгізу үшін міндетті өрістерді, сондай-ақ серверге деректерді жіберу батырмасын қамтиды. Деректерді беру үшін post әдісін қолдану қауіпсіздікті қамтамасыз етеді, өйткені деректер сұрау денесінде жіберіледі және URL мекенжайында көрсетілмейді.

Сервер жағында пайдаланушыдан алынған деректер олардың дұрыстығын тексеру үшін өңделеді. Бұл процесс дерекқорда сақталған жазбалармен енгізілген тіркелгі деректерін тексеруді қамтиды. Сәтті тексеру кезінде сервер пайдаланушы үшін сеанс жасайды, бұл авторизация күйін сақтауға және бір сессия шеңберінде қорғалған ресурстарға қол жеткізуге мүмкіндік береді. Сеанстарды пайдалану сонымен қатар пайдаланушы туралы ақпаратты сервер жағында сақтауға мүмкіндік береді, бұл қауіпсіздікті одан әрі жақсартады, өйткені құпия деректер клиент жағында сақталмайды.

Авторизацияны іске асырудың маңызды аспектісі деректердің қауіпсіздігін қамтамасыз ету болып табылады. Құпия сөздерді сақтау үшін шифрлауды қолдану, сондай-ақ SQL инъекциясы және сайтаралық сценарий (XSS) сияқты шабуылдан қорғау механизмдерін енгізу стандартты тәжірибе болып табылады. Осы шаралардан басқа, қауіпсіздік деңгейін жақсарту үшін екі факторлы аутентификацияны енгізуге болады.

Пайдаланушы тәжірибесін жақсарту және берілетін деректердің қауіпсіздігін қамтамасыз ету үшін JavaScript авторизация процесінде маңызды рөл атқарады. Клиент жағындағы сценарийлерді серверге жібермес бұрын пішін деректерін тексеру үшін пайдалануға болады, бұл дұрыс емес деректердің енгізілуіне жол бермейді және серверге жүктемені азайтады. Мысалы, JavaScript енгізілген логин мен пароль мәндерінің парольдің минималды ұзындығы сияқты берілген талаптарға сәйкес келуін тексере алады. Онлайн киносеанс билеттерін брондауға арналған веб - қосымшалардағы авторизация жүйесі қауіпсіздік пен ыңғайлылықты қамтамасыз ететін клиент пен сервердің өзара әрекеттесуін қамтитын күрделі процесс болып табылады. Заманауи веб-технологиялар мен деректерді қорғау әдістерін қолдану пайдаланушылардың құпия ақпаратын қорғауды және қызметтерге қол жетімділікті қамтамасыз ететін сенімді және қауіпсіз авторизация жүйелерін құруға мүмкіндік береді.



1.4 сурет. Тіркелу бөлімі

**1.3.2 Сеанстар кестесін қарау және залдан орын таңдау**

Онлайн киносеанс билеттерін брондауға арналған веб-қосымшаларда маңызды функционалды компоненттер сеанс кестесін қарау және залдағы орындарды таңдау болып табылады. Бұл мүмкіндіктер пайдаланушыларға қолайлы сеанстарды оңай табуға және фильмдерді көруге ыңғайлы орындарды таңдауға мүмкіндік береді.

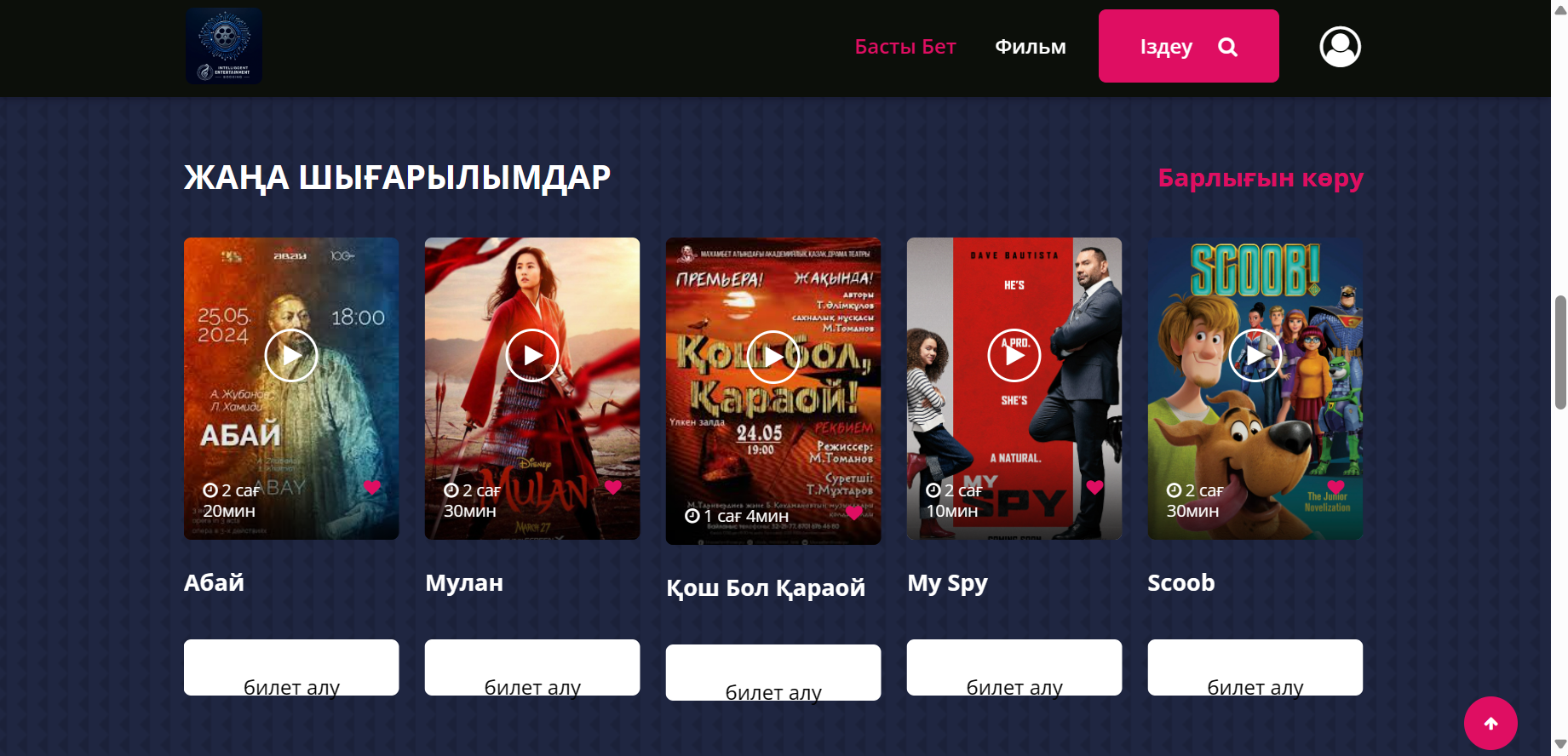
Сеанс кестесін қарау процесі негізгі бетте ағымдағы Фильмдер кестесін көрсетуден басталады. Фильм атаулары, көрсету уақыты және қол жетімді залдар сияқты сеанс деректері дерекқордан алынады және веб-бетте динамикалық түрде көрсетіледі. Пайдаланушының ыңғайлылығын жақсарту үшін сеанстарды фильмнің күні, уақыты немесе жанры бойынша сұрыптауға мүмкіндік беретін деректерді сүзу қолданылады. Бұл қызығушылық сеансын тез және дәл табуға мүмкіндік береді.

Тиісті сеансты таңдағаннан кейін пайдаланушыға залдағы орындарды таңдау мүмкіндігі беріледі. Бұл функция веб-бетте көрсетілген интерактивті зал схемасы арқылы жүзеге асырылады. Зал схемасы-брондау үшін қол жетімді орындарды көрсететін графикалық кескін. Әрбір орын бос, бос емес немесе қолжетімсіз деп белгіленуі мүмкін. Пайдаланушы схемадағы тиісті аймақтарды басу арқылы бір немесе бірнеше орынды таңдай алады.

Орындарды таңдау кезінде JavaScript интерактивтілікті қамтамасыз етуде және нақты уақыттағы ақпаратты жаңартуда шешуші рөл атқарады. Клиент жағындағы сценарийлер таңдалған орындардың күйін өзгерту және жаңартылған ақпаратты серверге жіберу арқылы басу оқиғаларын өңдейді. Бұл басқа пайдаланушылардың брондауына жол бермей, таңдалған орындарды уақтылы блоктауға мүмкіндік береді. Сервер жағында таңдалған орындардың қол жетімділігі тексеріліп, мәліметтер базасы жаңартылады. Егер барлық таңдалған орындар қол жетімді болса, жүйе брондауды растайды және пайдаланушыға электронды түрде жіберілуі мүмкін билетті жасайды. Әйтпесе, пайдаланушыға таңдалған орындарды брондау мүмкін .стігі туралы хабарланады және басқа қолжетімді орындарды таңдау ұсынылады.

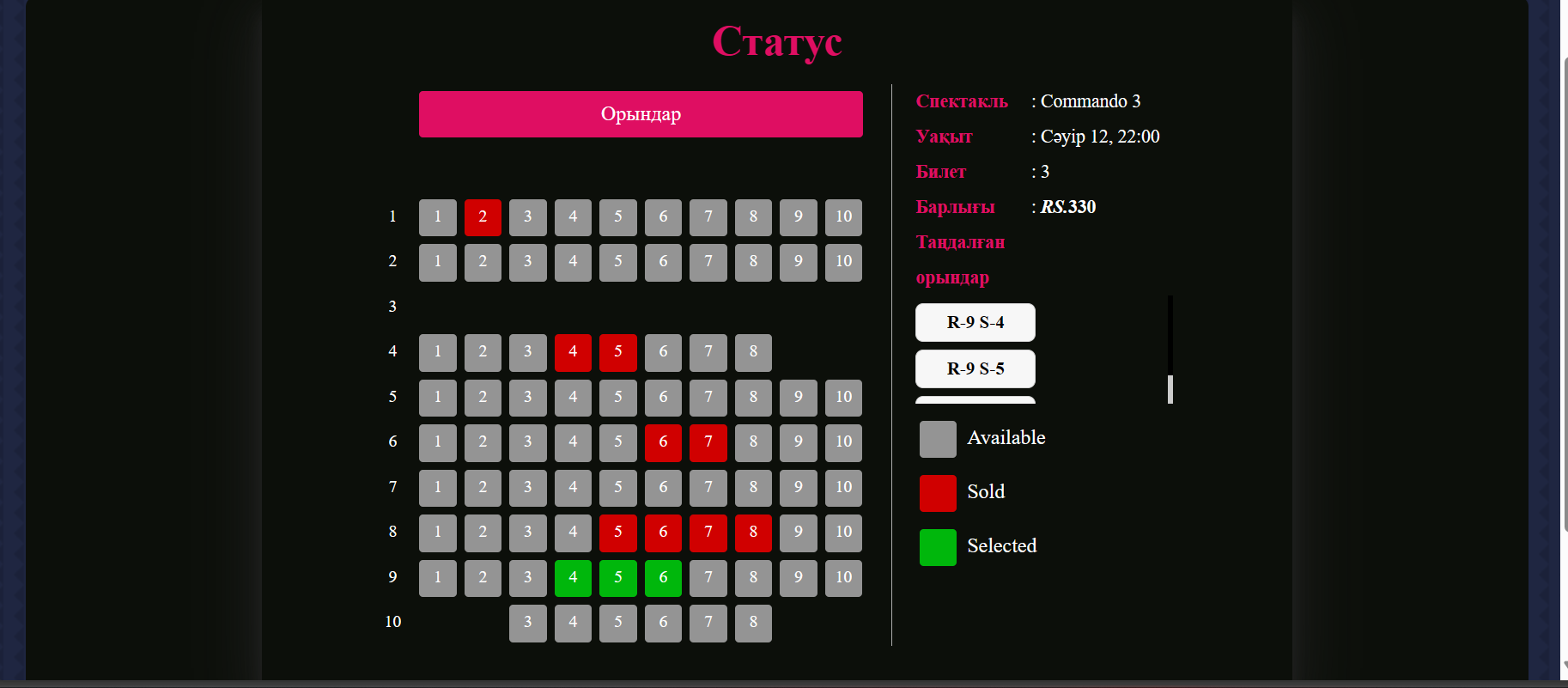
Деректерді жаңарту және Клиент пен сервер арасындағы ақпаратты синхрондау үшін AJAX (asynchronous JavaScript and XML) қолданылады. Бұл тәсіл бетті толығымен қайта жүктемей-ақ Веб-беттегі деректерді жаңартуға мүмкіндік береді, бұл пайдаланушы тәжірибесін айтарлықтай жақсартады және қолданбаның өнімділігін арттырады.

Осылайша, онлайн билеттерді брондау үшін веб - қосымшаларда сеанс кестесін қарау және залдағы орындарды таңдау мүмкіндігі пайдаланушыларға ыңғайлылық пен икемділікті қамтамасыз етеді. AJAX және JavaScript сияқты заманауи веб-технологияларды пайдалану жаңартылған ақпаратқа жылдам қол жеткізуге және брондау процесін тиімді басқаруға мүмкіндік беретін интерактивті және динамикалық интерфейстер жасауға мүмкіндік береді.



1.5 сурет. Көрсетілімдер тізімі

HTML, CSS және JavaScript көмегімен жүзеге асырылатын кинотеатрдағы орындарды таңдауға арналған веб-бет бар. HTML коды парақтың негізгі құрылымын, соның ішінде залдағы орындарды, түймелерді және күй тақталарын жасайды. Орындар пайдаланушылар таңдау үшін баса алатын түймелер немесе дивалар сияқты интерактивті элементтер ретінде ұсынылған. CSS осы элементтерді сәндеу үшін қолданылады, олардың сыртқы түрін, соның ішінде қол жетімді, бос емес және таңдалған орындарға арналған түстерді, сондай-ақ беттегі элементтердің дизайны мен орналасуын анықтайды. JavaScript беттің интерактивтілігін жергілікті шертулерді өңдеу, олардың күйін өзгерту (қол жетімді, бос емес, таңдалған) және таңдалған орындар туралы ақпаратты көрсететін күй жолағын жаңарту арқылы қамтамасыз етеді. Сонымен қатар, JavaScript орындарды тексеру және түпкілікті брондау үшін деректерді серверге жібере алады. Бұл компоненттердің барлығы бірге жұмыс істейді, пайдаланушыларға ыңғайлы және интерактивті интерфейс жасайды, бұл кинотеатрдағы орындарды оңай таңдауға және олардың таңдау күйі туралы жаңартылған ақпаратты алуға мүмкіндік береді.



1.6 сурет. Орындар тізімі

### **1.3.3 Тапсырмаларды иерархиялық талдау**

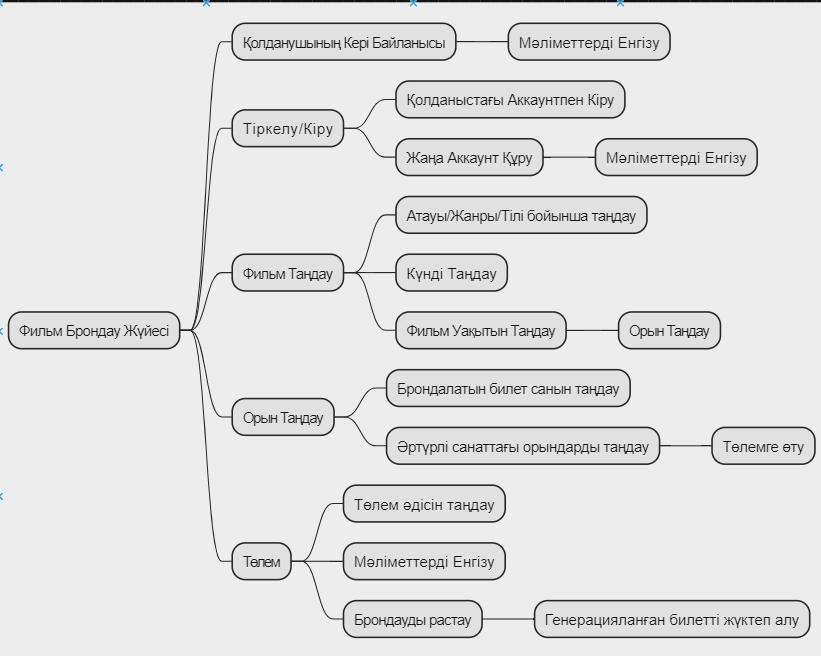
Тапсырмаларды иерархиялық талдау диаграммасы негізгі құрамдас бөліктер мен ішкі қадамдарға бөлінген киносеанстарға билеттерді брондау процесінің құрылымдық дисплейі болып табылады. Иерархияның шыңы - негізгі міндет — бірнеше негізгі компоненттерге бөлінген фильмдерді брондау жүйесі.

Бірінші компонент пайдаланушының деректерді енгізу процесін қамтиды. Бұл пайдаланушы жүйе туралы өз пікірлерін немесе ескертулерін беретін кері байланыс нысанын толтыруды білдіреді. Мұндағы маңызды элемент-пайдаланушы тәжірибесін жақсарту және қызмет сапасын жақсарту үшін кері байланысты жинау және өңдеу.

Келесі маңызды компонент тіркеу және кіру процестерін қамтиды. Кіру бұрыннан бар есептік жазбамен немесе Жаңа тіркелгі жасау арқылы жүзеге асырылуы мүмкін. Жаңа тіркелгі жасау үшін пайдаланушының аты, электрондық пошта мекенжайы және құпия сөз сияқты жеке деректерін енгізу қажет. Бұл процесс жүйе ұсынатын қызметтердің қауіпсіздігі мен жекелендірілуін қамтамасыз ету үшін маңызды.

Көру үшін фильмді таңдау процесі бірнеше ішкі кезеңдерден тұрады. Пайдаланушы алдымен фильмді аты, жанры немесе тілі сияқты әртүрлі критерийлер бойынша таңдайды. Фильмді таңдағаннан кейін көрсетілім күні мен уақытын таңдау керек, бұл пайдаланушыға кинотеатрға барудың ең қолайлы уақытын табуға мүмкіндік береді. Әрі қарай, пайдаланушы залдағы орындарды таңдауға көшеді. Бұл кезең VIP, стандартты немесе бюджеттік орындар сияқты әртүрлі санаттардан брондау үшін билеттер санын және залдағы нақты орындарды таңдауды қамтиды.

Соңғы компонент төлем процесін қамтиды. Пайдаланушы банктік карталарды, электрондық әмияндарды немесе басқа қол жетімді опцияларды қамтуы мүмкін төлем әдісін таңдайды. Осыдан кейін карта нөмірі және CVV коды сияқты Төлем деректерін енгізу қажет. Деректерді енгізгеннен кейін жүйе қателерді болдырмау және транзакцияның дұрыстығын қамтамасыз ету үшін брондауды растауды сұрайды. Процесс пайдаланушының кинотеатрға кіру үшін пайдалануға болатын электронды билетті жүктеуімен аяқталады. Тапсырмаларды иерархиялық талдау диаграммасы киносеанстарға билеттерді брондау процесінің әртүрлі кезеңдерінің дәйектілігі мен өзара байланысын айқын көрсетеді. Бұл жүйенің құрылымы мен логикасын біртұтас түсінуге, жақсартудың ықтимал қиындықтары мен бағыттарын анықтауға және жүйенің функционалды компоненттерін тиімді жоспарлауға және жүзеге асыруға көмектеседі.[4]



1.7 сурет. Талдау диаграммасы

**2 КОНТЕЙНЕРЛІК ТЕХНОЛОГИЯНЫ ҚОЛДАНУ ЕРЕЖЕЛЕРІ**

Docker контейнерлер негізгі технология, бағдарламалық қамтамасыз етуді әзірлеу және орналастыру процесі болып табылады. Контейнерлер қолданбаларды және олардың тәуелділіктерін дербес және тасымалданатын пакеттерге оқшаулауға мүмкіндік береді, оларды орындау ортасындағы айырмашылықтарға қарамастан Docker қолдайтын кез келген жүйеде іске қосуға болады.

Docker контейнерлерін пайдаланудың басты артықшылығы-олардың қолданбаның консистенциясы мен болжамын қамтамасыз ету мүмкіндігі. Дамудың дәстүрлі тәсілінде операциялық жүйелердің конфигурациясындағы, орнатылған кітапханалардағы және басқа факторлардағы айырмашылықтар үйлесімділік мәселелеріне және күтпеген қателерге әкелуі мүмкін. Docker бұл мәселені әр қолданба үшін оқшауланған ортаны құру арқылы шешеді, бұл әзірлеушілерге олардың коды жергілікті машиналарда да, өндірістік ортадағы серверде де бірдей жұмыс істейтініне сенімді болуға мүмкіндік береді.

Docker контейнерлер қолданбаны іске қосу үшін қажет нәрсенің барлығын қамтиды: бастапқы код, кітапханалар, тәуелділіктер, құралдар және конфигурация файлдары. Бұған барлық қажетті компоненттерді қамтитын файлдық жүйенің суреті ретінде ұсынылатын Docker кескінін жасау арқылы қол жеткізіледі. Контейнерлер Осы кескіндер негізінде іске қосылады және қосымшаларды бір-бірінен және негізгі операциялық жүйеден оқшаулауды қамтамасыз етеді.

Контейнерлерді оқшаулаудың көптеген артықшылықтары бар. Біріншіден, ол қауіпсіздікті қамтамасыз етеді, өйткені контейнерлер арнайы конфигурацияланбаған жағдайда басқа контейнерлердің немесе хосттың деректері мен процестеріне қол жеткізе алмайды. Бұл бір контейнердің бұзылуынан туындауы мүмкін зиянды шектейді. Екіншіден, оқшаулау тәуелділікті басқаруды жеңілдетеді. Әр қосымшаның өзіндік кітапханалары мен тәуелділіктері бар, бұл нұсқа қайшылықтары мен үйлесімділік мәселелерін шешеді. Үшіншіден, контейнерлер виртуалды машиналармен салыстырғанда жеңіл. Олар хост ресурстарын тиімдірек пайдаланады, өйткені олар амалдық жүйенің ядросын бөліседі және әр данасы үшін бөлек амалдық жүйені іске қосуды қажет етпейді.

Фильм, театр билеттерін брондау сайты болып табылатын "Portal" жобасында Docker контейнерлерін пайдалану мысалын қарастырыңыз. Мұндай қосымшаның жұмыс істеуін қамтамасыз ету үшін пайдаланушылардың сұраныстарын өңдеуге арналған веб-сервер, фильмдер мен брондау туралы ақпаратты сақтауға арналған мәліметтер базасы және пайдаланушылардың аутентификациясы мен төлемдерді өңдеуге арналған көмекші қызметтер сияқты бірнеше компоненттерді орналастыру қажет. Бұл компоненттердің әрқайсысын жеке Docker контейнеріне салуға болады, бұл оларды дербес басқаруға және кез-келген серверге оңай орналастыруға мүмкіндік береді.

Мысалы, веб-сервер мен дерекқорды бір Docker көпір желісіне қосылған жеке контейнерлерде орналастыруға болады. Бұл олардың сервердегі басқа процестерден оқшаулануын қамтамасыз етеді, сонымен қатар желі арқылы бір-бірімен байланысуға мүмкіндік береді. Веб-сервер пайдаланушылардан HTTP сұрауларына қызмет етеді және брондау туралы ақпаратты алу және сақтау үшін дерекқормен өзара әрекеттеседі. Қосымша контейнерлерді төлемдерді өңдеу және аутентификацияны басқару сияқты тапсырмалар үшін орналастыруға болады, бұл олардың тәуелсіздігін және қажет болған жағдайда масштабтау мүмкіндігін қамтамасыз етеді.

"Portal" жобасында Docker контейнерлерін пайдалану қолданбаның құрамдас бөліктерін орналастыру және басқару процесін жеңілдетіп қана қоймайды, сонымен қатар шешімнің икемділігі мен ауқымдылығын қамтамасыз етеді. Контейнерлеудің арқасында жеке компоненттерді оңай жаңартуға, жаңа нұсқаларды тексеруге және қажет болған жағдайда алдыңғы нұсқаларға оралуға болады, бұл үнемі өзгеріп отыратын талаптар мен жүктемелер жағдайында қосымшаның сенімділігі мен тұрақтылығын едәуір арттырады.

**2.1 Бір түйінді желі**

Docker-дегі бір түйінді желі (single Host Networking) бір хостта контейнерлердің өзара әрекеттесуін ұйымдастыруда шешуші рөл атқарады. Бұл контейнерлерге деректерді тиімді бөлісуге және үйлестіруге мүмкіндік береді, бұл бүкіл қосымшаның тұтастығы мен дәйектілігін қамтамасыз етеді.

Біріншіден, бір түйінді желі контейнерлердің желілік трафигін сыртқы ортадан және бір-бірінен оқшаулауды қамтамасыз етеді, бұл қауіпсіздік деңгейін арттырады. Контейнерлер басқа контейнерлердің деректеріне байқаусызда қол жеткізу мүмкіндігін болдырмай, тек арнайы конфигурацияланған желі арқылы өзара әрекеттесе алады. Бұл әсіресе күрделі көп компонентті жүйелердегі шабуылдар мен деректердің бұзылуын болдырмау үшін өте маңызды. Пайдаланушы көпір желісін құру (user-Defined Bridge Network) IP мекенжайларын және басқа контейнерлік желілік ресурстарды орталықтан басқаруға мүмкіндік береді. Бұл қолданбаларды орнатуды және масштабтауды жеңілдетеді, өйткені барлық контейнерлер IP мекенжайларын берілген ауқымнан автоматты түрде алады және бір-бірінам бойынша оңай таба алады. Желі конфигурациясы өзгертілген немесе жаңа контейнерлер қосылған жағдайда, әкімшіге IP мекенжайларын қолмен тағайындаудың және маршруттауды конфигурациялаудың қажеті жоқ.

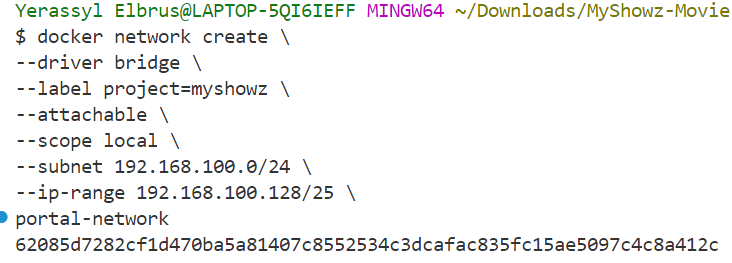
Бір түйінді желі микросервистік архитектураны қолданатын қосымшаларды орналастыру үшін өте қолайлы, мұнда қосымшаның әр компоненті бөлек контейнерде орындалады. "Portal" жобасы аясында веб-сервер, дерекқор, аутентификация және төлем қызметтері сияқты компоненттер икемділік пен масштабталуды қамтамасыз ететін ортақ желі арқылы өзара әрекеттесе алады. Мысалы, веб-сервер фильмдер мен брондау туралы ақпарат алу үшін дерекқорға сұраныстар жасай алады, ал аутентификация қызметі сол желі арқылы жіберілген пайдаланушы деректерін тексере алады.

Бір түйінді желіні пайдалану сонымен қатар контейнерлерді жөндеу және бақылау процесін жеңілдетеді. Әкімшілер желі белсенділігін оңай бақылай алады, желі трафигін талдай алады және контейнерлер арасындағы өзара әрекеттесу мәселелерін анықтай алады. Контейнерлер арасындағы байланыстың жоғалуы немесе дұрыс бағытталмауы сияқты мәселелер анықталған жағдайда, желіні тексеруге және қажетті өзгерістерді жедел енгізуге болады.

Бір түйінді желінің арқасында контейнерлерді басқа контейнерлерді қайта құруды немесе қайта жүктеуді қажет етпестен желіден оңай қосуға немесе жоюға болады. Бұл жүйенің жоғары икемділігі мен бейімделуін қамтамасыз етеді, бұл талаптар мен жүктемелердегі өзгерістерге жылдам жауап беруге мүмкіндік береді. Пайдаланушылар немесе деректер көбейген жағдайда, әкімшілер қолданбаның үздіксіз жұмыс істеуін қамтамасыз ете отырып, бар желіге жаңа контейнерлерді қоса алады.

"Portal" жобасында бір түйінді желі веб-серверге, дерекқорға және басқа компоненттерге бір-бірімен тиімді әрекеттесуге мүмкіндік береді. Жеке көпір желісін құру оқшаулау мен қауіпсіздікті қамтамасыз етеді, желілік ресурстарды басқаруды жеңілдетеді және қосымшаның микросервистік архитектурасын қолдайды. Мысалы, веб-сервер пайдаланушылардың HTTP сұрауларына қызмет ете алады және брондау туралы ақпаратты алу және сақтау үшін дерекқормен әрекеттесе алады. Конфигурация өзгерген немесе жаңа қызметтер қосылған жағдайда, бір түйінді желі қолданбаның тұрақты және тиімді жұмысын қамтамасыз ете отырып, жаңа жағдайларға оңай бейімделуге мүмкіндік береді.

Docker-дегі бір түйінді желі бір хосттағы контейнерлерге арнайы жасалған көпір желісі арқылы бір-бірімен байланысуға мүмкіндік береді. Бұл желі контейнерлерді сыртқы желілік Трафиктен оқшаулайды және олардың өзара әрекеттесуін басқаруға мүмкіндік береді.

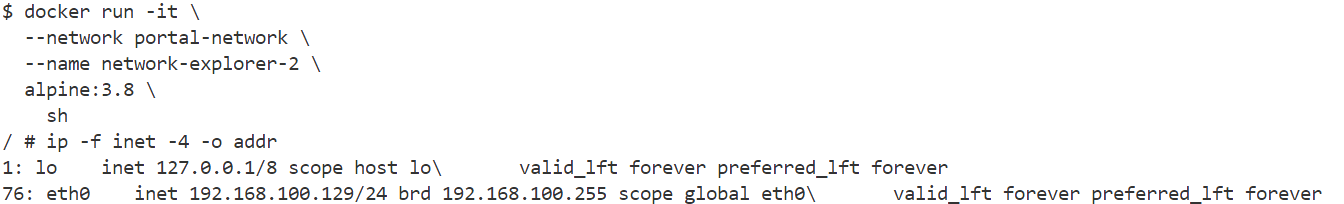


2.1 сурет. Көпір желісін салу

Бір түйінді желіні орнатудың алғашқы қадамы-көпір желісін құру. Мұны ішкі желі және IP мекенжай ауқымы сияқты белгілі бір параметрлері бар жаңа желіні жасайтын `docker network create` пәрменін пайдалану арқылы жасауға болады. Бұл желі контейнерлерді оқшаулауды қамтамасыз етеді және оларға тағайындалған IP мекенжайлары арқылы байланысуға мүмкіндік береді.

Желі құрылғаннан кейін контейнерлер оған қосылуы мүмкін. Бұл контейнерлерді іске қосу кезінде жасалады, онда олар пайдаланушы желісіне қосылады және берілген ауқымнан IP мекенжайларын алады. Контейнерлерді бір желіге қосу оларға деректерді бөлісуге және бір-бірімен өзара әрекеттесуге мүмкіндік береді.

Желілік конфигурацияны тексеру және зерттеу үшін контейнерге тағайындалған желілік интерфейстер мен IP мекенжайлары туралы ақпарат алу үшін `IP addr` сияқты диагностикалық командаларды орындайтын контейнерлерді пайдалануға болады. Бұл контейнерлердің желіге дұрыс қосылғанына және бір-бірімен әрекеттесе алатынына көз жеткізуге көмектеседі.



2.2 сурет. Желілік конфигурацияны тексеру және зерттеу

Хостқа кіретін трафикті өңдеу контейнер порттарын жариялау арқылы жүзеге асырылады. Бұл сыртқы пайдаланушыларға хост порттары арқылы контейнерлерде жұмыс істейтін қызметтерге қол жеткізуге мүмкіндік береді. Мысалы, контейнер ішінде жұмыс істейтін веб-сервер контейнер ішіндегі 80 портына сәйкес келетін 8080 хост портында қол жетімді болуы мүмкін.

Сондай-ақ, контейнерлерге арналған DNS серверлерін олардың желілік параметрлерін басқаруға және домен атауларының дұрыс ажыратымдылығын қамтамасыз етуге конфигурациялауға болады. Бұл әсіресе нақты DNS конфигурациясын қажет ететін күрделі желілік конфигурацияларда пайдалы.



2.3 сурет. Кіріс трафигін өңдеу және реттелетін DNS орнату

Соңында, желі конфигурациясын тексеру үшін "docker network inspect" пәрмені пайдаланылады, ол құрылған желі, оған қосылған контейнерлер және олардың желі параметрлері туралы толық ақпарат береді. Бұл әкімшіге дұрыс орнатылғанына көз жеткізуге және желі конфигурациясындағы ықтимал мәселелерді анықтауға мүмкіндік береді.



2.4 сурет. Желі конфигурациясын тексеру

Осылайша, Docker-де бір түйінді желіні орнату контейнерлерді оқшаулауға, олардың өзара әрекеттесуін басқаруға және желілік параметрлерді басқаруға мүмкіндік береді, бұл өндірістік ортада контейнерленген қосымшаларды орналастырудың маңызды бөлігі болып табылады.[3]

**2.2 Kubernetes оркестрациясын кірістіру технологиясы**

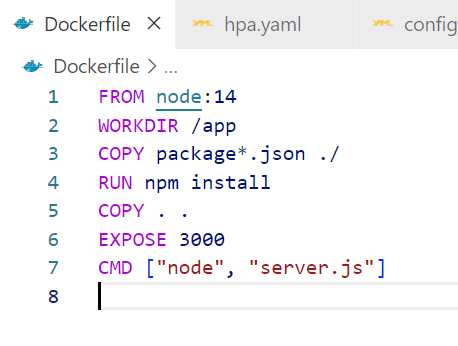
Kubernetes-кластердегі контейнерленген қолданбаларды басқаруға арналған қуатты жүйе. "Kubernetes" сөзі гректің "рульдік" сөзінен шыққан, бұл оның контейнерлерді басқарудағы рөлін білдіреді. Жүйені Google әзірледі және қазір Cloud Native Computing Foundation қолдайды. Kubernetes-тің негізгі мақсаты-контейнерлерге салынған қосымшаларды орналастыруды, басқаруды және масштабтауды автоматтандыру. Қолданбалар физикалық немесе виртуалды серверлерде жұмыс істейтін дәстүрлі тәсілден айырмашылығы, контейнерлеу қолданбаны және оның барлық тәуелділіктерін бір кескінге жинауға мүмкіндік береді. Бұл болжамды мінез-құлықты қамтамасыз етеді және қолданбаларды әртүрлі орталарда орналастыруды жеңілдетеді. Ол бұл контейнерлерді подиум деп аталатын логикалық бірліктерге топтастыру арқылы басқарады. Әрқайсысы астында ресурстар мен желіні бөлісетін бір немесе бірнеше контейнерлер болуы мүмкін. Kubernetes-тің негізгі міндеті-қолданба көшірмелерінің берілген саны әрқашан жұмыс істейтініне көз жеткізу, егер олар істен шыққан болса, контейнерлерді қайта іске қосу және қажет болған жағдайда оларды кластер түйіндері арасында жылжыту.

Жүйе жүктемені теңестіру, Автоматты масштабтау, өзін-өзі қалпына келтіру, конфигурация мен құпияларды басқару және тоқтаусыз жаңартулар сияқты көптеген компоненттер мен мүмкіндіктерді қамтиды. Kubernetes инфрақұрылымды абстракциялайды, бұл әзірлеушілер мен операторларға серверді басқаруға емес, қолданбаға назар аударуға мүмкіндік береді.

Kubernetes кластерлері басқару түйіндері мен жұмыс түйіндерінен тұрады. Басқару түйіндері кластердің күйін бақылайды және қосымшаларды орналастыру және масштабтау туралы шешім қабылдайды. Жұмыс түйіндері контейнерлерді орындайды және олардың күйін басқару түйіндеріне хабарлайды.

**2.2.1 Dockerfile және minikube қолдану принциптері**

Dockerfile контейнерлер мен желілік ресурстарды басқаруға арналған арнайы бағдарламалық жасақтаманы қолдана отырып, бағдарламалық жасақтаманы орындау үшін ортаны құру және конфигурациялау процесін сипаттайды. Негізгі ортаны орнатудан басталады, содан кейін тәуелділіктер мен конфигурациялардың сипаттамалары сияқты қажетті файлдар көшіріледі. Осыдан кейін жүйе барлық қажетті тәуелділіктерді автоматты түрде жүктейді және орнатады. Әрі қарай, қосымшаның жұмыс істеуі үшін қажетті барлық басқа файлдардың көшірмесі жасалады. Бұл қолданбаға кіру үшін қандайжелі порты пайдаланылатыны анықталады.

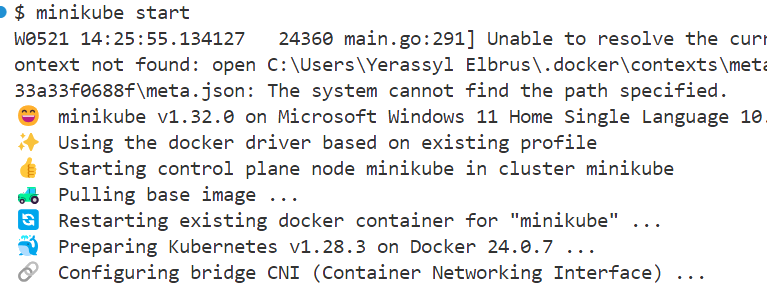


2.5 сурет. Dockerfile жазылуы

Minikube-бұл жергілікті Kubernetes кластерін орналастыруды жеңілдетуге арналған құрал. "Minikube start" командасы жергілікті бір түйінді Kubernetes кластерін іске қосады, бұл әзірлеушілерге нақты кластердің әрекетін имитациялайтын ортада қолданбаларды сынауға және әзірлеуге мүмкіндік береді. Бұл процесс виртуалды машинаны құруды және конфигурациялауды, қажетті Kubernetes компоненттерін жүктеуді және оларды іске қосуды қамтиды. Нәтижесінде қолданбаларды орналастыру және сынау үшін пайдалануға дайын толық жұмыс істейтін Kubernetes кластері пайда болады.

"Minikube start" пәрмені сонымен қатар kubectl пәрмен жолы құралын автоматты түрде конфигурациялайды, осылайша ол орналастырылатын қолданбаларды басқару және бақылау үшін ыңғайлы интерфейсті қамтамасыз ете отырып, сол кластермен әрекеттесе алады.[1]

Кітаптағы нұсқаулар "minikube start" жергілікті Kubernetes кластерін іске қосатынын және оны пайдалану үшін kubectl конфигурациялайтынын растайды, бұл Kubernetes ортасында қолданбаларды әзірлеу және сынау процесін айтарлықтай жеңілдетеді.



2.6 сурет. Minikube start командасы

**2.2.2 Автоматты масштабтау және конфигурацияны басқару**

YAML файлдары конфигурацияға декларативті тәсілді қамтамасыз ете отырып, Kubernetes нысандарын басқаруда шешуші рөл атқарады. Олар pods, қызметтер, конфигурация карталары және құпиялар сияқты нысандарды анықтау арқылы кластер ұмтылуы керек күйді сипаттайды. Олардың арқасында әзірлеушілер мен операциялық топтар әртүрлі орталар үшін жалпы конфигурацияны қолдана отырып, бір контексте жұмыс істей алады: әзірлеу, тестілеу және өндіріс. Бұл өмірлік циклдің барлық кезеңдерінде олардың тұрақты және дәйекті жұмысын қамтамасыз ете отырып, қолданбаларды біріктіру және жеткізу процесін айтарлықтай жеңілдетеді. Осылайша, YAML Kubernetes-тегі файлдар кластерді декларативті басқарудың қуатты құралы болып табылады, бұл қазіргі заманғы бұлтты қосымшаларды орналастыру және басқару процестерін автоматтандыруға, икемділікке және сенімділікке ықпал етеді.

Kubernetes-те автоматты масштабтау ағымдағы жүктемеге байланысты қолданбалар даналарының (көшірмелерінің) санын динамикалық түрде реттеуге мүмкіндік береді. Бұған процессор уақыты немесе жад сияқты ресурстарды пайдалануды бақылайтын және оңтайлы өнімділікті сақтау үшін жұмыс істейтін контейнерлер санын автоматты түрде көбейтетін немесе азайтатын механизм арқылы қол жеткізіледі.

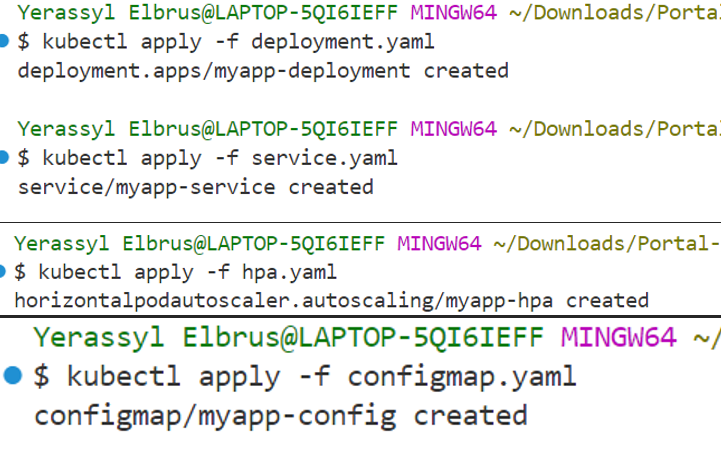
Көлденең автоматты масштабтау (Horizontal Pod Autoscaler) CPU жүктеу немесе жад сияқты көрсеткіштерді үнемі қадағалап отырады және кіріс сұрауларын өңдеуге жеткілікті қуат беру үшін pods көшірмелерінің санын реттейді. Мысалы, егер процессордың барлық жиектерде жүктелуі 80% - ға жетсе, жүйе жүктемені азайту және өнімділікті жақсарту үшін автоматты түрде қосымша жиектер жасай алады. Жүктеме азайған кезде, артық ресурстарды пайдаланбау үшін ҚҚС саны сәйкесінше азаяды.

Тік Автоматты масштабтау (Vertical pod Autoscaler) олардың санын өзгертудің орнына әр подқа бөлінген ресурстарды реттейді. Ол қолданбаның қажеттіліктеріне байланысты CPU және жад бөлуді арттыруы немесе азайтуы мүмкін. Бұл кластердегі ресурстарды тиімдірек пайдалану үшін пайдалы болуы мүмкін қосымша қадаларды жасамай-ақ қолданбалардың өнімділігін сақтауға мүмкіндік береді.

Kubernetes ішіндегі конфигурацияны басқару қолданба кодынан конфигурация деректерін бөлуге мүмкіндік береді, бұл қолданбаларды басқаруды және орналастыруды жеңілдетеді.

ConfigMaps конфигурация деректерін кілт-мән жұптары ретінде сақтау үшін пайдаланылады, оларды контейнерлерге қоршаған орта айнымалылары, пәрмен жолы аргументтері немесе файлдар ретінде беруге болады. Бұл кодты өзгертпестен немесе контейнерлерді қайта жинамай-ақ қолданба конфигурациясын өзгертуге мүмкіндік береді. ConfigMaps қолданба параметрлері, дерекқор URL мекенжайлары және басқа опциялар сияқты маңызды емес деректерді сақтауға жарамды.

Бұл конфигурацияны басқару тетіктері қолданба параметрлерін жаңарту процесін жеңілдетеді, қоршаған ортаның өзгеруіне жылдам жауап беруге мүмкіндік береді және сезімтал деректермен жұмыс істеу кезінде қауіпсіздіктің жоғары деңгейін қамтамасыз етеді.



2.7 сурет. deployment, service, configmap файлдарын құру

**2.2.3 Нұсқалар мен жаңартуларды басқару**

Kubernetes-те нұсқалар мен жаңартуларды басқару қолданбаларды жаңарту процесін жаңа нұсқаларды енгізуге байланысты үзілістер мен тәуекелдерді азайту үшін ұйымдастыруға мүмкіндік береді. Kubernetes-те blue-green deployments сияқты осы мақсатқа жетуге көмектесетін бірнеше орналастыру стратегиялары бар.

Blue-green deployments екі бірдей ортаның болуын білдіреді (көк және жасыл). Біреуі (мысалы, көк) қолданушыларға белсенді түрде қызмет етеді, ал екіншісі (жасыл) қосымшаның жаңа нұсқасын шығарады. Жаңа нұсқаны сәтті орналастырғаннан және тексергеннен кейін трафик көк ортадан жасылға ауысады. Бұл жаңа нұсқаға жылдам ауысуды және проблемалар туындаған жағдайда алдыңғы нұсқаға жылдам оралуды қамтамасыз етеді. Оның артықшылықтары:

1. Лезде ауысу: трафикті бір нұсқадан екіншісіне ауыстыру лезде жүреді, бұл бос уақытты азайтады және пайдаланушыларға қызметтің үздіксіз қол жетімділігін қамтамасыз етеді.

2. Жаңарту қауіпсіздігі: жаңа нұсқа толығымен оқшауланған ортада (мысалы, жасыл) орналастырылған, бұл оны пайдаланушыларға қызмет көрсете бастағанға дейін жауынгерлік жағдайларға мүмкіндігінше жақын жағдайда мұқият тексеруге мүмкіндік береді.

3. Кері қайтарудың қарапайымдылығы: егер жаңа нұсқада проблемалар табылса, трафикті пайдаланушыларға әсерін азайта отырып, алдыңғы тұрақты нұсқаға (көк орта) тез және оңай қайтаруға болады.

4. Параллельді тестілеу және әзірлеу: Blue-green deployments қолданбаның екі жұмыс нұсқасын бір уақытта қолдауға мүмкіндік береді, бұл қазіргі пайдаланушыларға әсер етпестен жаңа мүмкіндіктер мен түзетулерді тексеруді жеңілдетеді.

5. Тәуекелдерді азайту: жаңа нұсқа оқшауланған ортада толық сынақтан өткендіктен, ықтимал тұрақсыз нұсқаны енгізу қаупі азаяды.

Бұл механизмдер Kubernetes қолданбасын жаңарту процесін сенімдірек, икемді және қауіпсіз етеді, бұл әсіресе заманауи қолданбалар мен қызметтердің динамикалық ортасында маңызды.

Изображение выглядит как текст, Шрифт, снимок экрана, линия

Автоматически созданное описание

2.8 сурет. Нұсқалар мен жаңартуларды басқару файлдарын құру

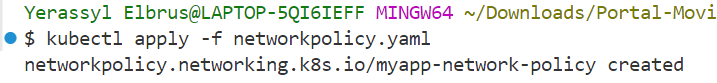
### **2.2.4 Желілерді Басқару**

Kubernetes желісін басқару қолданба құрамдастары арасындағы қауіпсіз және тиімді өзара әрекеттесуді, сондай-ақ олардың сыртқы қызметтермен өзара әрекеттесуін қамтамасыз етеді. Kubernetes-те желінің өзара әрекеттесуі networkpolicies көмегімен басқарылады, ол қандай подтардың бір-бірімен әрекеттесе алатынын анықтайды. Бұл қолданбаларды рұқсатсыз кіруден және желілік шабуылдардан қорғай отырып, кластер ішінде оқшауланған желілік орталарды жасауға мүмкіндік береді.

Желілік саясаттар подтар үшін кіріс және шығыс трафикті басқаратын ережелерді орнатуға мүмкіндік береді. Мысалы, белгілі бір подтарға тек белгілі бір затбелгілерге қол жеткізуді шектеуге немесе тек белгілі бір порттарға кіруге рұқсат беруге болады. Бұл қауіпсіздік деңгейін едәуір жақсартады, өйткені қандай қызметтер мен компоненттердің бір-бірімен байланыса алатындығын дәл бақылауға болады.

Kubernetes сонымен қатар жүктемені теңестіруді қолдайды, бұл трафикті берілістер арасында біркелкі бөлуге мүмкіндік береді. Бұған pods тобы үшін Тұрақты IP мекенжайын және DNS атауын қамтамасыз ететін қызметтер (Қызметтер) арқылы қол жеткізіледі, бұл олардың қол жетімділігі мен жеке pods ақауларына төзімділігін қамтамасыз етеді.

Ingress контроллері кіріс HTTP және HTTPS трафигін басқару арқылы кластердің сыртынан жіберулерге қол жеткізуге мүмкіндік береді. Олар URL немесе домендік атауларға негізделген әр түрлі қызметтерге сұраныстарды бағыттауға мүмкіндік береді, бұл сіздің қосымшаңыздың икемділігі мен ауқымдылығын қамтамасыз етеді. Kubernetes-тегі желіні басқару қауіпсіздік үшін желілік саясатты анықтауды және қолдануды, трафикті оңтайлы бөлу үшін жүктемені теңестіруді және сыртқы қол жетімділікті басқару үшін Ingress-ті орнатуды қамтиды. Бұл механизмдер контейнерленген қосымшалар үшін қауіпсіз, басқарылатын және тиімді желілік инфрақұрылымды құруға көмектеседі. Осы жұмыстардың жиынтығы networkpolicy.yaml файлында сақталады.[2]



2.9 сурет. Желілерді басқару файлын құру

**3 UML ДИАГРАММАЛАРЫН ПАЙДАЛАНУ**

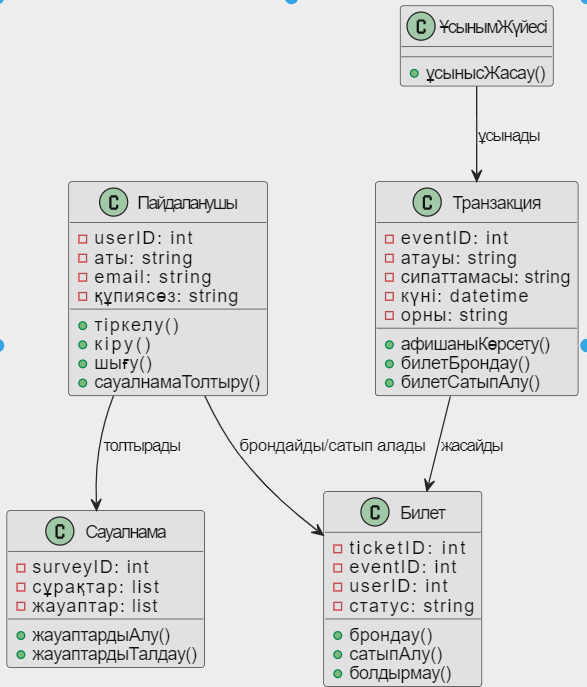
UML (Unified Modeling Language) — бағдарламалық жасақтама компоненттерін сипаттау, визуализациялау, әзірлеу және құжаттау үшін қолданылатын стандартты визуалды модельдеу тілі. UML диаграммалардың кең ауқымын ұсынады, олардың әрқайсысы бағдарламалық жасақтаманы әзірлеу процесінде белгілі бір мақсатқа қызмет етеді. UML-дің негізгі мақсаты-дизайн процесін жеңілдету, топ мүшелері арасындағы байланысты жақсарту және күрделі жүйелерді жүзеге асыруда айқындықты қамтамасыз ету. UML диаграммалары екі негізгі категорияға бөлінеді: құрылымдық және мінез-құлық. Құрылымдық диаграммалар сыныптар, компоненттер және олардың өзара байланысы сияқты жүйенің статикалық аспектілерін сипаттайды. Оларға сынып, объект, компонент және орналастыру диаграммалары кіреді. Мінез-құлық диаграммалары, керісінше, жүйенің динамикалық аспектілеріне назар аударады, мысалы, объектілер арасындағы өзара әрекеттесу, басқару ағындары және күйлер. Бұл санатқа реттілік диаграммалары, әрекеттер, күйлер және өзара әрекеттесу диаграммалары кіреді.

UML диаграммаларын қолданудың көптеген артықшылықтары бар. Біріншіден, олар жүйенің визуалды көрінісін қамтамасыз етеді, бұл бағдарламалық жасақтама архитектурасын түсіну мен талдауды жеңілдетеді. Бұл әсіресе күрделі жобалармен жұмыс істеу кезінде өте маңызды, мұнда визуализация дамудың алғашқы кезеңдерінде ықтимал проблемаларды анықтауға көмектеседі. Екіншіден, UML диаграммалар топ мүшелері арасындағы жақсы байланысқа ықпал етеді, өйткені олар жүйенің әртүрлі аспектілерін сипаттайтын бірыңғай стандартты ұсынады. Бұл адам факторымен байланысты түсініспеушіліктер мен қателіктердің ықтималдығын азайтады. Соңында, UML диаграммалар жоба құжаттамасында шешуші рөл атқарады. Олар жүйенің нақты және егжей-тегжейлі сипаттамасын ұсынады, бұл оны қолдау мен болашақта жаңартуды жеңілдетеді.

Осылайша, диаграммалар бағдарламалық жасақтама жасаушылардың арсеналындағы қуатты құрал болып табылады, құрылымдық дизайн тәсілін қамтамасыз етеді, байланыс сапасын жақсартады және құжаттау процесін жеңілдетеді. Оларды пайдалану сенімді және оңай қолдау көрсетілетін жүйелерді құруға ықпал етеді, нәтижесінде жобалар сәтті жүзеге асырылады.[5]

**3.1 Класстар диаграммасы**

Класстар диаграммасы-бұл жүйенің құрылымын сипаттайтын және негізгі сыныптарды, олардың атрибуттары мен әдістерін және олардың арасындағы қатынастарды көрсететін статикалық модель. Бұл жағдайда билеттерді брондау және сатып алу функционалдығын, сондай-ақ сауалнамаға негізделген ұсыныстарды қамтитын фильмдерге, спектакльдерге және концерттерге билеттерді сатудың зияткерлік портал жүйесі қарастырылады.



3.1 сурет. Класстар диаграммасы

"Пайдаланушы" классы - бұл сынып жүйенің пайдаланушылары туралы ақпаратты ұсынады. Негізгі атрибуттарға пайдаланушы идентификаторы (userID), аты (аты), электрондық пошта (email) және құпия сөз (құпиясөз) кіреді. Сынып әдістеріне тіркеу (тіркелу), жүйеге кіру (кіру), жүйеден шығу (шығу) және сауалнаманы толтыру (толық сауалнама) функциялары кіреді.

"Транзакция" классы - бұл сынып билеттерді сатып алуға болатын әрекеттерді сипаттайды. Атрибуттарға оқиға идентификаторы (eventID), атауы (атауы), сипаттама (сипаттама), өткізу күні (Күн) және өткізу орны (орны) кіреді. Әдістер афишаларды көрсетуге (афишаныКөрсету), билеттерді брондауға (билетБроньдау) және билеттерді сатып алуға (билетСатыпАлу) мүмкіндік береді.

"Билет" сыныбы билеттер туралы ақпаратты сақтауға арналған. Оған билет идентификаторы (ticketID), оқиға идентификаторы (eventID), пайдаланушы идентификаторы (userID) және билет күйі (статус) атрибуттары кіреді. Бұл сыныптың әдістері брондауға (броньдау), сатып алуға (сатыпАлу) және билеттерден бас тартуға (Болдырмау) мүмкіндік береді.

"Ұсыныс жасау" сыныбы пайдаланушыларға ұсыныстар жасауға жауап береді. ұсынысЖасау() әдісі ең қолайлы әрекеттерді ұсыну үшін пайдаланушылардың деректерін және олардың сауалнамаларға жауаптарын талдайды.

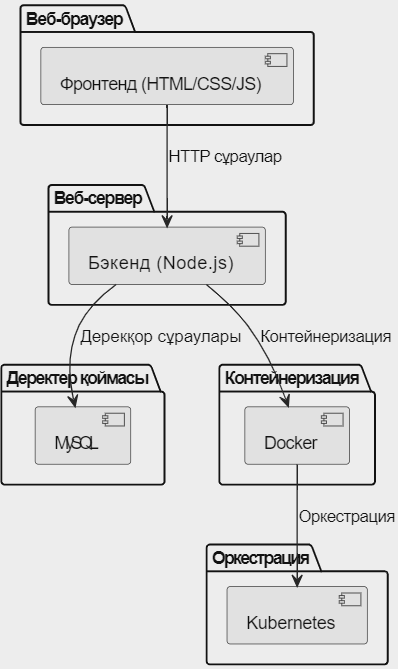
"Сауалнама" сыныбында пайдаланушылардың сұрақтары мен жауаптары туралы ақпарат бар. Атрибуттарға сауалнама идентификаторы (surveyID), сұрақтар тізімі (сұрақтар) және жауаптар тізімі (жауаптар) кіреді. Сынып әдістеріне пайдаланушылардың жауаптарын алу (жауаптардыАлу) және оларды талдау (жауаптардыТалдау) кіреді.

Сыныптар арасындағы қатынастар объектілердің бір-бірімен қалай әрекеттесетінін көрсетеді. Мысалы, пайдаланушы сауалнаманы толтырады, оқиғаларға билеттерді брондайды және сатып алады, ал ұсыныс қозғалтқышы пайдаланушылардың жауаптарын талдайды және ұсыныстар жасайды. Сынып диаграммасы жүйенің функционалдығын тиімді жоспарлауға және жүзеге асыруға мүмкіндік беретін негізгі элементтер мен олардың өзара байланыстарын анықтау арқылы жүйенің құрылымын жақсы түсінуге көмектеседі.[6]

**3.2 Компоненттер диаграммасы**

Компоненттер Диаграммасы - бұл жүйенің құрылымдық компоненттерін және олардың өзара байланысын бейнелейтін UML диаграммасының түрі. Ол Модульдер, сыныптар немесе кітапханалар сияқты бағдарламалық жасақтаманың жеке бөліктері бір-бірімен қалай әрекеттесетінін және олардың арасында қандай тәуелділіктер бар екенін көрсетеді. Мұндай диаграммалар жүйенің архитектурасын түсінуге және оның дамуын жоспарлауға көмектеседі.

UML құрамдас диаграммасы жүйелерді модельдеудің көптеген артықшылықтарын ұсынады. Бұл бағдарламалық жасақтаманың архитектурасын визуализациялауға мүмкіндік береді, әр түрлі компоненттер арасындағы өзара әрекеттесу мен тәуелділіктер туралы нақты түсінік береді. Бұл жүйенің құрылымын түсінуді жеңілдетеді және ықтимал мәселелерді анықтауды жеңілдетеді. Компоненттер диаграммасы топтық жұмыстың тиімдірек болуына ықпал етеді, өйткені ол архитектуралық шешімдерді талқылау мен құжаттаудың жалпы негізі болып табылады. Ол сондай-ақ жүйені өзгерту және кеңейту процесін жеңілдетеді, өйткені ол өзара байланысты бөліктерді оңай анықтауға мүмкіндік береді. Мұндай диаграммаларды пайдалану тестілеу мен қосымшаларды орналастыру процесін жақсартады, өйткені компоненттерді бөлек тексеруге және орналастыруға болады.



3.2 сурет. Компоненттер диаграммасы

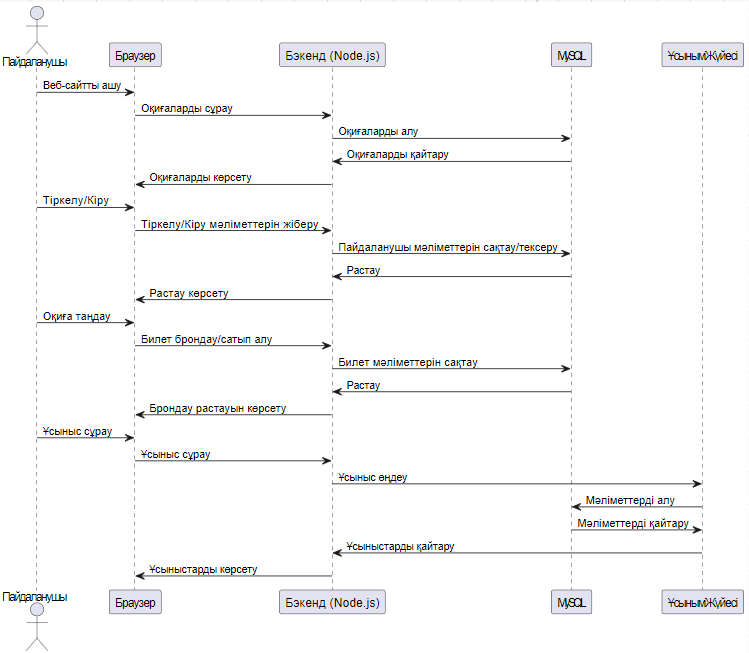
Бұл компоненттер диаграммасы веб-қосымшаның архитектурасын ұсынады, оның әртүрлі бөліктері мен технологиялары арасындағы байланысты көрсетеді. Диаграмманың жоғарғы жағында қолданбаның алдыңғы жағымен әрекеттесетін веб-шолғыш орналасқан. HTML, CSS және JavaScript көмегімен жүзеге асырылған алдыңғы жағы пайдаланушы интерфейсін өңдейді және веб-серверге HTTP сұрауларын жібереді.

Node.js-те жазылған сервер қосымшасы жұмыс істейтін Веб-сервер, осы сұрауларды қабылдайды және тиісті операцияларды орындайды. Деректерді сақтау және басқару үшін сервер MySQL дерекқорымен өзара әрекеттеседі, оған деректерді оқу және жазу үшін сұраулар жібереді. Сонымен қатар, артқы жағы контейнерлеу үшін Docker-ді пайдаланады, бұл қолданбаны және оның тәуелділіктерін контейнерлерге салып, олардың тасымалдануы мен оқшаулануын қамтамасыз етеді. Контейнерлерді басқару және оларды оркестрлеу үшін Kubernetes қолданылады. Kubernetes контейнерленген қосымшаларды орналастыруды, масштабтауды және басқаруды автоматтандыруды қамтамасыз етеді, бұл бүкіл жүйенің сенімділігі мен тиімділігін арттырады.

Бұл диаграмма осылай веб-қосымшаның әртүрлі компоненттерінің пайдаланушы интерфейсінен бастап мәліметтер базасы мен контейнерлерді оркестрлеу жүйесіне дейінгі өзара әрекеттесуін айқын көрсетеді.[6]

**3.3 Реттілік диаграммасы**

Реттілік диаграммасы-бұл жүйедегі объектілер арасындағы өзара әрекеттесуді уақыт бойынша көрсететін UML диаграмма түрі. Ол өзара әрекеттесуге қатысатын объектілерді және олардың арасындағы хабарламаларды көрсетеді, бұл өзара әрекеттесу тәртібін көрсетеді. Диаграмма жүйеде процестердің қалай орындалатынын және белгілі бір сценарий шеңберінде объектілердің бір-бірімен қалай әрекеттесетінін түсінуге көмектеседі.



* 1. сурет. Реттілік диаграммасы

Бұл реттілік диаграммасы пайдаланушы, браузер, Node.js сервері арасындағы өзара әрекеттесу процесін көрсетеді, MySQL дерекқоры және сыртқы ұсыныс жүйесі. Ол билеттерді брондау және ұсыныстарды сұрау үшін веб-сайтпен жұмыс істеу кезінде пайдаланушының қадамдарын көрсетеді.

Диаграмма пайдаланушының веб-сайтты ашуынан басталады. Браузер оқиғаларды алу туралы сұранысты артқы жағына жібереді, содан кейін оқиғалар туралы ақпарат алу үшін MySQL дерекқорына жүгінеді. Осыдан кейін дерекқор оқиғалар туралы деректерді пайдаланушыға көрсету үшін браузерге жіберетін бэкендке қайтарады.

Келесі қадам-тіркелу немесе пайдаланушыға кіру. Пайдаланушы өз деректерін енгізеді, оны браузер серверге жібереді. Сервер MySQL дерекқорына кіру арқылы пайдаланушы деректерін сақтайды және тексереді, содан кейін Растауды шолғышқа жібереді.

Әрі қарай, пайдаланушы оқиғаны таңдап, билетті брондау немесе сатып алу процесін бастайды. Браузер брондау деректерін MySQL дерекқорында сақтайтын серверге сұрау жібереді. Ақпаратты сәтті сақтағаннан кейін дерекқор Растауды бэкендке қайтарады, ол өз кезегінде браузерге хабарлайды және пайдаланушыға брондау нәтижелерін көрсетеді.

Соңғы қадамда пайдаланушы ұсыныстар сұрайды. Браузер серверге сұраныс жібереді, ол оны өңдейді және сыртқы ұсыныстар жүйесіне қол жеткізеді. Сыртқы жүйе қажетті деректерді артқы жағына қайтарады, содан кейін пайдаланушыға көрсету үшін ұсыныстарды браузерге жібереді.

Бұл реттілік диаграммасы жүйенің әртүрлі компоненттері арасындағы өзара әрекеттесу мен деректермен алмасудың бүкіл процесін анық көрсетеді, бұл веб-қосымшаның ішінде брондау және ұсыныстарды сұрау процесінің қалай жүзеге асырылатынын түсінуге көмектеседі.

# **Қорытынды**

Осы дипломдық жұмысты орындау барысында Алматы қаласы театрларының спектакльдеріне тапсырыстарды ресімдеу үшін Apache Spark базасында Зияткерлік жүйе әзірленіп, енгізілді. Пайдаланушы тәжірибесін жақсарту үшін пайдаланушының көңіл-күйі мен темпераментін ескеретін жеке ұсыныстар жүйесін құруға баса назар аударылады.

Жұмыстың негізгі аспектілерінің бірі жүйенің икемділігін, масштабталуын және тұрақтылығын қамтамасыз ететін микросервистерді оркестрлеу үшін Kubernetes пайдалану болды. Бұл технология Apache Spark-тың үлкен деректерді өңдеу және талдау мүмкіндіктерімен бірге жоғары өнімді және сенімді платформаны құруға мүмкіндік берді.

Әзірленген жүйеге интуитивті интерфейсі бар веб-бағдарлама, интеграцияланған ұсыныстар жүйесі және пайдаланушыларды тіркеу және авторизациялау, сеанстар кестесін қарау және залдағы орындарды таңдау мүмкіндігі кіреді. Даму барысында Docker көмегімен контейнерлеудің заманауи әдістері қолданылды, бұл жүйенің компоненттерін орналастыруды және басқаруды жеңілдетті.

Деректердің қауіпсіздігі мен конфигурацияны тиімді басқару мәселелеріне көп көңіл бөлінеді, бұл жүйенің нақты жұмыс жағдайында сенімді жұмысын қамтамасыз етті. UML диаграммаларын пайдалану жүйенің архитектурасын нақты құрылымдауға және негізгі компоненттер мен олардың өзара әрекеттесуін визуализациялауға мүмкіндік берді, бұл жобаны тиімді жоспарлауға және жүзеге асыруға ықпал етті.

Жұмыстың нәтижелері Kubernetes және Apache Spark сияқты заманауи технологияларды пайдалану деректердің үлкен көлемін басқару және жекелендірілген қызметтерді ұсыну үшін тиімді және сенімді шешімдер жасауға мүмкіндік беретінін растайды. Осы жоба шеңберінде әзірленген зияткерлік жүйе пайдаланушы тәжірибесін жақсартып қана қоймай, мәдени іс-шаралар саласындағы цифрлық шешімдердің дамуына елеулі үлес қосты.

Осылайша, бұл зерттеу пайдаланушылардың жеке және ыңғайлы қызметтерге өсіп келе жатқан қажеттіліктерін қанағаттандыра алатын жоғары өнімді жүйелерді құру үшін озық технологияларды сәтті қолдануды көрсетеді. Әзірленген платформа әртүрлі салалардағы интеллектуалды ұсыныстар жүйелері мен тапсырыстарды басқару саласындағы одан әрі зерттеулер мен әзірлемелерге негіз бола алады, бұл оның қазіргі қоғам үшін өзектілігі мен маңыздылығын көрсетеді.

# **Пайдаланылған әдебиеттер тізімі**

1 Джей Вьяс, Крис Лав. Kubernetes изнутри. ДМК Пресс, 2022. – 378 с.

2 Марко Лукша. Kubernetes в действии. ДМК Пресс, 2019. – 672 с.

3 JEFF NICKOLOFF STEPHEN KUENZLI. Docker in action. Manning Publications Co. 2019. – 300c.

4 Jim Arlow and Ila Neustadt. UML 2 и Унифицированный процесс. Практический объектно­ориентирован­ный анализ и проектирование, 2­е издание. – Пер. с англ. – СПб: Символ­ Плюс, 2007. – 624 с.

5 Крэг Ларман. Применение UML 2.0 и шаблонов проектирования. Практическое руководство.- 3-е издание.: Пер. с англ. М.: ООО "И.Д. Вильямс", 2013. 736 с.

6 Мартин Р., Мартин М. Принципы, паттерны и методики гибкой разработки на языке C#. – Пер. с англ. – СПб.: Символ-Плюс, 2011. – 768 с.

7 Дэвид Флэнаган. JavaScript Полное руководство. ООО “Диалектика”, 195027, Санкт-Петербург, Магнитогорская ул., д. 30. 2021. – 714c. URL: https://codelibrary.info/online?url=L2Rvd25sb2FkLzE0MTRfamF2YXNjcmlwdC1wb2xub2UtcnVrb3ZvZHN0dm8ucGRm

8 How To Make a Booking Website: From Domains to Marketing. URL: <https://www.websitebuilderexpert.com/building-websites/how-to-make-a-booking-website/> (26.05.2024)

9 What is a UML Diagram? | Different Types and Benefits – Miro. – URL: <https://miro.com/ru/> (26.05.2024)

10 What is Docker? – URL: <https://www.ibm.com/topics/docker> (26.05.2024)

11 Adomavicius, G., & Tuzhilin, A. (2005). "Toward the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions." IEEE Transactions on Knowledge and Data Engineering, 734-749.

12 Pazzani, M. J., & Billsus, D. (2007). Content-Based Recommendation Systems. In The Adaptive Web: Methods and Strategies of Web Personalization. Springer.

13 Collaborative Filtering, Encyclopedia of Machine Learning and Data Mining, Springer, 2017. Mohamed Sarwat and Mohamed Mokbel, Collaborative Filtering, Encyclopedia of Database Systems, Springer, 2018.

14 Ricci, F., Rokach, L., & Shapira, B. (2011). Recommender Systems Handbook.

15 Bishop, C. M. Pattern Recognition and Machine Learning. Springer, 2006,- 738 p.

16 Chambers, B., & Zaharia, M. (2018). Spark: The Definitive Guide: Big Data Processing Made Simple. O'Reilly Media.

17 Zečević, P., & Bonaći, M. (2016). Spark in Action. Manning Publications.

18 Ryza, S., Laserson, U., Owen, S., & Wills, J. Advanced Analytics with Spark: Patterns for Learning from Data at Scale. O'Reilly Media, 2015, -275 p.

# Қосымша А

Kubernetes үшін басқару файлдарының құрамы

apiVersion: apps/v1

kind: Deployment

metadata:

  name: myapp-deployment

  labels:

    app: myapp

spec:

  replicas: 3

  selector:

    matchLabels:

      app: myapp

  template:

    metadata:

      labels:

        app: myapp

    spec:

      containers:

      - name: myapp-container

        image: your-docker-image:latest

        ports:

        - containerPort: 80

        resources:

          requests:

            cpu: "100m"

            memory: "200Mi"

          limits:

            cpu: "500m"

            memory: "500Mi"

apiVersion: v1

kind: ConfigMap

metadata:

  name: myapp-config

data:

  DATABASE\_URL: "postgres://user:password@hostname:5432/dbname"

apiVersion: autoscaling/v2

kind: HorizontalPodAutoscaler

metadata:

  name: myapp-hpa

spec:

  scaleTargetRef:

    apiVersion: apps/v1

    kind: Deployment

    name: myapp-deployment

  minReplicas: 3

  maxReplicas: 10

  metrics:

  - type: Resource

    resource:

      name: cpu

      target:

        type: Utilization

        averageUtilization: 50