WEBVTT

00:00:00.000 --> 00:00:07.000

Қайырлы күн, құрметті студенттер. Біз Practice Test сұрағын талдауды бастаймыз.

00:00:07.000 --> 00:00:11.000

Сұрақтың шарты бойынша бізде three-tier web application, яғни үш деңгейлі веб-қосымшасы бар.

00:00:11.000 --> 00:00:15.000

Front-end бірінші деңгейінде бізде статикалық веб-парақшалар бар.

00:00:15.000 --> 00:00:22.000

Екінші деңгейде, логикалық деңгейде қосымша бір EC2 инстанс машинасында жұмыс істейді.

00:00:22.000 --> 00:00:26.000

Бұл деңгейде өте ұзақ тапсырмалар орындалады.

00:00:26.000 --> 00:00:33.000

Соңғы деректер деңгейінде бізде MySQL дерекқоры жұмыс істейді.

00:00:33.000 --> 00:00:58.000

Solutions Architect ретінде сіз decoupling арқылы бүкіл қолданбаны highly available ету, яғни жоғары қолжетімді ету, яғни оларды бір-бірімен әлсіз байланыстыру тапсырмасын алдыңыз.

00:00:58.000 --> 00:01:00.000

Бұл үшін не істеуге болады?

00:01:00.000 --> 00:01:05.000

Берілген нұсқалардан бір дұрыс жауапты таңдауымыз керек.

00:01:05.000 --> 00:01:12.000

Бұл слайдта сіз жауап нұсқаларын көресіз, сонымен қатар екінші жауап нұсқасы дұрыс екенін көресіз.

00:01:12.000 --> 00:01:17.000

Дегенмен, әр нұсқаны қарастырайық.

00:01:17.000 --> 00:01:28.000

Жауаптардың құрылымына қарасаңыз, олардың барлығы үш бөліктен тұрады және әрбір бөлік біздің веб-қосымшаның үш деңгейінің әрқайсысына қатысты.

00:01:28.000 --> 00:01:33.000

Бірінші сөйлем front-end деңгейі туралы айтады.

00:01:33.000 --> 00:01:50.000

Біз бірінші нұсқа үшін біздің барлық веб-парақшаларымызды, ассеттерімізді, яғни суреттерді, бейнелерді back-end деңгейіне тасымалдап, EC2 инстансы ішіндегі қолданбаның жанына орналастырып, инстанстың өлшемін ұлғайту ұсынылатынын көреміз.

00:01:50.000 --> 00:02:03.000

Бұл сұрақта талап етілгенге қайшы келеді, біз компоненттерді бөліп, оларды әлсіз біріктіруіміз керек еді, бірақ бұл жағдайда, керісінше, біз оларды біріктіреміз, сондықтан біз бұл нұсқаны бірден алып тастай аламыз.

00:02:03.000 --> 00:02:10.000

Екінші нұсқаны қарасаңыз, барлық статикалық ассеттер мен веб-парақшаларды S3-ге тасымалдау ұсынылады.

00:02:10.000 --> 00:02:17.000

Иә, S3 жағында бізде статикалық веб-сайттарды орналастыру мүмкіндігі бар, біз оны қосып, пайдалана аламыз.

00:02:17.000 --> 00:02:22.000

Бұл шешім веб-парақшаларды EC2-де орналастырудан әлдеқайда арзан.

00:02:22.000 --> 00:02:26.000

Сондықтан біз бұл нұсқаны әзірге қалдырып, үшіншісін қарастырамыз.

00:02:26.000 --> 00:02:33.000

Үшінші нұсқа веб-парақшаларды емес, барлық статикалық ассеттерді S3-ге тасымалдауды ұсынады.

00:02:33.000 --> 00:02:44.000

Мүмкін, бұл қате болуы мүмкін, бірақ егер бұл қате емес, әдейі болса, онда бұл нұсқа дұрыс емес, өйткені S3 тек ассеттерді ғана емес, веб-парақшаларды да орналастыра алады.

00:02:44.000 --> 00:02:51.000

Төртінші нұсқаны қарастыратын болсақ, ол барлық статикалық ассеттер мен веб-парақшаларды CloudFront-қа тасымалдауды ұсынады.

00:02:54.000 --> 00:03:11.000

Бұл нұсқа логикалық тұрғыдан мүлдем дұрыс емес, өйткені CloudFront өзі Content Delivery Network болып табылады. Идея мынада: бұл кэші бар серверлер желісі және кэшке кейбір деректерді қою үшін бізге көз қажет.

00:03:11.000 --> 00:03:19.000

Amazon S3 әдетте көз ретінде пайдаланылады, олар үйлесімде өте жақсы жұмыс істейді, яғни S3 және CloudFront.

00:03:19.000 --> 00:03:23.000

Сондықтан төртінші нұсқа логикалық тұрғыдан мүлдем дұрыс емес, біз оны алып тастаймыз.

00:03:23.000 --> 00:03:26.000

Бізде екінші және үшінші жауап нұсқалары қалады.

00:03:26.000 --> 00:03:28.000

Екінші бөлігін қарастырайық.

00:03:28.000 --> 00:03:38.000

Екінші нұсқаның екінші бөлігінде Elastic Container Service сервисін пайдалану және сол жерде Auto Scaling баптау ұсынылады.

00:03:38.000 --> 00:03:45.000

Біз сыни немесе қарама-қайшы ештеңе көрмейміз, біз қолданбамызды контейнерлерге тасымалдай аламыз.

00:03:45.000 --> 00:03:50.000

Үшінші нұсқаның екінші бөлігінде не айтылғанын көрейік.

00:03:50.000 --> 00:03:55.000

Мұнда concurrency limit теңшеп, қолданбамызды Lambda-ға тасымалдау ұсынылады.

00:03:55.000 --> 00:03:59.000

Осылайша, serverless архитектураға ауысамыз.

00:03:59.000 --> 00:04:08.000

Бұл нұсқа сәйкес келмейді, себебі Lambda максимум 15 минут, яғни 900 секунд жұмыс істей алады.

00:04:08.000 --> 00:04:15.000

Сұрақта біз ұзаққа созылатын процестермен жұмыс істеп жатқанымызды айтады, яғни бұл 15 минуттан астам уақыт болуы мүмкін.

00:04:15.000 --> 00:04:19.000

Яғни, мұнда Lambda қолданбауға сілтеме болды.

00:04:19.000 --> 00:04:24.000

Сондықтан үшінші нұсқаны алып тастаймыз, екінші нұсқа қалады.

00:04:24.000 --> 00:04:31.000

Дегенмен, басқа нұсқалардың үшінші сөйлемін қарастырайық.

00:04:31.000 --> 00:04:40.000

Екінші нұсқа дерекқорды Amazon RDS жүйесіне көшіруді және Multi-AZ deployment қосуды ұсынады.

00:04:40.000 --> 00:04:46.000

Иә, бұл - managed service пайдалану әкімшілік тапсырмалардан босатуға көмектесетін жақсы шешім.

00:04:46.000 --> 00:04:56.000

Сонымен қатар, дайын Multi-AZ шешімдері бар, бұл сіздің деректер қорыңызды highly available, яғни жоғары қолжетімді етуге көмектеседі.

00:04:56.000 --> 00:05:05.000

Бір availability zone жағында проблема пайда болса, автоматты түрде failover орын алады және secondary database негізгі болады.

00:05:05.000 --> 00:05:08.000

Осылайша дерекқор жұмысын жалғастырады.

00:05:08.000 --> 00:05:21.000

Үшінші нұсқадағы үшінші сөйлемді қарасақ, ол біздің дерекқорды DynamoDB-ге көшіруді ұсынады.

00:05:21.000 --> 00:05:39.000

DynamoDB - NoSQL дерекқоры, сондықтан MySQL-де орналастырылған реляционды дерекқорды RDS сервисіне тасымалдау оңайырақ.

00:05:39.000 --> 00:05:42.000

DynamoDB – бұл NoSQL, сондықтан деректерді модельдеу жұмыстарын жүргізу қажет және бүкіл құрылым өзгереді.

00:05:42.000 --> 00:05:48.000

Сондықтан күш-жігер тұрғысынан қиын.

00:05:48.000 --> 00:05:53.000

Осыған сүйене отырып, екінші нұсқаны дұрыс деп анық айта аламыз.

00:05:53.000 --> 00:06:08.000

Келесі слайдтарда бұл мәселенің мәтіндік талдауы болады, сіз видеоны тоқтатып, оны оқи аласыз.

00:06:08.000 --> 00:06:13.000

Осымен Practice Test сұрағын талдауымыз аяқталды. Назарларыңызға рахмет.