ТИПОВАЯ АРХИТЕКТУРА ЭОС ВУЗА

Дудников С.А., к.т.н., доцент

Вербицкий А.С., к.т.н., доцент

МГВОКУ, г. Москва, Россия

Вербицкий А.А.,студент

НИУ МИЭТ, г. Москва, Россия

Аннотация. В статье рассматриваются концептуальные основы, ключевые требования и принципы проектирования типовой архитектуры электронной образовательной среды (ЭОС) современного вуза. Особое внимание уделено разделению цифровой экосистемы на независимые функциональные модули, охватывающие учебные, коммуникационные, аналитические и административные процессы университета. Обоснована эффективность модели, сочетающей стабильное платформенное ядро информационной безопасности и управления данными с гибким набором подключаемых сервисов. Предложенный модульный подход позволяет высшим учебным заведениям поэтапно внедрять программные решения, избегать монопольной зависимости от

поставщиков софта и существенно оптимизировать затраты на модернизацию цифровой инфраструктуры.

Ключевые слова: электронная образовательная среда, типовая архитектура, модульная архитектура, интеграция информационных систем.

Электронная образовательная среда (ЭОС) призвано собрать воедино разрозненные элементы учебного процесса: от лекционных материалов и каналов связи до составления графиков занятий, выдачи заданий и мониторинга текущей успеваемости. Подобная платформа стирает границы между аудиторными занятиями, удаленным форматом и гибридным обучением, обеспечивая постоянный доступ к знаниям и делая взаимодействие обучающихся, преподавательского состава и руководства понятным и открытым. Стандартизированная модель построения среды задает единые правила функционирования и взаимодействия всех ее компонентов. Внедрение такого подхода нацелено на то, чтобы связать между собой различные вузовские сервисы, избавиться от лишнего дублирования записей, поднять уровень отказоустойчивости и защиты данных, а также сократить финансовые вложения на этапах разработки, сопровождения и последующей модернизации инфраструктуры.

Понимание фундаментальной роли ЭОС позволяет сформулировать ключевые ожидания от программного комплекса, которые затем фиксируются в проектных документах в виде детальных технических и качественных параметров. Если обобщить эти положения, требования сводятся к нескольким аспектам. Платформа обязана полностью закрывать рутинные учебные задачи: публикацию курсов, распределение проектных заданий, организацию рубежного контроля, фиксацию оценок, коммуникацию и ведение электронного журнала. Пользователям критически важно иметь стабильный канал связи с ресурсами независимо от текущего времени суток, географической точки или используемого гаджета. Комплекс должен бесперебойно работать при любом формате занятий, включая трансляцию вебинаров, проведение консультаций и групповую деятельность обучающихся, что особенно ценно для

обучающихся, находящихся на производственных практиках, в научных экспедициях или дальних поездках.

В то же время сервисы среды должны напрямую взаимодействовать с внешними базами: электронными библиотечными системами, графиком занятости аудиторий, кадровыми реестрами, системами документооборота и прочими вузовскими программами. Приоритетное внимание уделяется информационной безопасности: здесь необходимы сквозная учетная запись, централизованные механизмы входа и детально проработанное разграничение полномочий для разных категорий пользователей. Крайне важно обеспечить сохранность персональных сведений и академических записей, внедрить надежные алгоритмы криптографической защиты, фиксировать историю пользовательских действий, наладить создание резервных копий и пресекать попытки несанкционированного проникновения. Вместе с тем среда должна выдерживать пиковые сетевые нагрузки в период сессий и быстро восстанавливать штатную работу в случае программно-аппаратных сбоев. Сама конструкция системы проектируется с запасом прочности, чтобы увеличение числа обучающихся, учебных курсов или внешних сервисов не приводило к заметному падению быстродействия.

Отдельный акцент делается на эргономике и удобстве взаимодействия. Интерфейсные решения должны выглядеть единообразно, легко читаться, корректно масштабироваться на смартфонах и ПК, а также отвечать стандартам инклюзивности для людей с ограничениями по здоровью. Для поддержания стабильности требуются централизованные инструменты администрирования, регулярные обновления, четкие инструкции по интеграции сторонних решений и регламенты действий технической поддержки при инцидентах.

Перечисленный комплекс требований прямо диктует модульную структуру цифровой среды. Для ее практической реализации единый массив разделяется на самостоятельные блоки, отвечающие за конкретный спектр педагогических, административных и системных задач:

Единый портал вуза выступает в качестве входных ворот, открывая доступ к внутренним ресурсам, расписанию, новостной ленте и справочникам.

Персональный кабинет формирует индивидуальное виртуальное рабочее место обучающегося, преподавателя или сотрудника деканата с учетом его служебных полномочий.

Система управления учебным процессом (LMS) служит для администрирования онлайн-курсов, размещения файлов, проверки практических работ и фиксации результатов освоения программы.

Централизованное хранилище материалов предназначено для упорядочивания, каталогизации, быстрого поиска и повторного применения лекционных слайдов, текстовых документов и видеозаписей.

Электронная библиотека открывает доступ к учебникам, периодике, научным базам данных и внешним репозиториям.

Модуль аттестации и тестирования отвечает за проведение промежуточных срезов знаний, автоматизированную проверку ответов и сохранение баллов в ведомостях.

Коммуникационный шлюз объединяет групповые чаты, рассылки, форумы и

уведомления для постоянного диалога между участниками процесса.

Сервис видеоконференцсвязи применяется при дистанционных лекциях, онлайн-консультациях, публичных защитах дипломов и сохранении записей прошедших встреч.

Модуль диспетчеризации и расписания автоматизирует распределение учебных аудиторий, формирование сетки пар и экзаменационных дней.

Цифровое портфолио накапливает и систематизирует данные обо всех достижениях обучающегося: от успехов в учебе до научных публикаций и волонтерства.

Подсистема входного контроля работ обучающихся берет на себя автоматизированную приемку курсовых и отчетов, проверяя их соответствие стандартам оформления и выявляя заимствования.

Ядро учета контингента аккумулирует сведения об учебных группах, академических планах, преподавательской нагрузке и итоговых протоколах комиссий.

Блок внутреннего документооборота обеспечивает подготовку, маршрутизацию согласования, электронную подпись и надежное хранение приказов и распоряжений.

Аналитический модуль собирает статистику посещаемости и активности пользователей, генерирует сводные отчеты и помогает руководству объективно оценивать качество образовательных программ.

Служба идентификации и контроля доступа отвечает за учетные записи, сквозную авторизацию и матрицу ролей.

Интеграционная шина гарантирует бесперебойную синхронизацию данных между отдельными подсистемами среды и внешними информационными базами.

Служба управления данными поддерживает целостность мастер-справочников, следит за актуальностью сведений, их архивацией и резервным дублированием.

Контур информационной безопасности блокирует потенциальные угрозы, контролирует неизменяемость журналов событий и охраняет приватные сведения.

Модуль мониторинга следит за состоянием серверов, фиксирует замедления в обработке запросов и предлагает системным инженерам инструменты тонкой настройки.

Служба пользовательской поддержки принимает заявки от обучающихся и педагогов, предоставляет справочные регламенты и решает технические трудности.

Любой модуль в этой схеме представляет собой обособленную функциональную единицу с четко очерченными задачами, собственной внутренней архитектурой и стандартными протоколами сопряжения. Подобный подход дает возможность модернизировать, настраивать или заново переписывать конкретный сервис без риска нарушить функционирование соседних элементов. При этом изоляция не требует выделения под каждую задачу отдельного физического сервера: ключевое значение имеет строгое логическое разделение функций.

Одновременно с этим блок глубоко вовлечен в общее рабочее русло. Он непрерывно запрашивает параметры у соседних модулей и отправляет им результаты своих расчетов через утвержденные шины обмена. К примеру, модуль LMS извлекает списки зачисленных обучающихся и учебные планы из ядра управления контингентом, подгружает материалы из электронных библиотечных фондов и сразу передает экзаменационные баллы в аналитический модуль деканата. Тесная взаимосвязь достигается за счет сквозных идентификаторов, использования единых нормативных справочников и стандартизированных форматов передачи данных. За каждый информационный массив назначается единственный первичный источник, а сам процесс обмена сопровождается валидацией данных, фильтрацией ошибок и подтверждением доставки сообщений. Это на корню пресекает дублирование записей и снимает необходимость повторного ручного набора одних и тех же сведений.

В конечном счете элементы среды сочетают независимость внутренней структуры с четкостью внешних интерфейсов. Глубина интеграции здесь измеряется не числом прямых связей между серверами, а непрерывностью сквозных пользовательских сценариев: единичное действие в одной вкладке браузера автоматически запускает цепочку обновлений по всей экосистеме, создавая ощущение монолитного сервиса.

Компонентный принцип построения платформы позволяет адаптировать ее возможности под актуальные нужды конкретного учебного заведения. Ненужные опции можно отключить, новые сервисы – подсоединить по мере назревания задач, а уже работающие инструменты – обновить или сменить аналогами без остановки всего комплекса. В центре структуры всегда остается неизменяемое ядро, которое регулирует аутентификацию, распределение прав, синхронизацию справочников и обмен сетевыми пакетами. Трансформация внешних модулей никак не затрагивает фундамент среды при сохранении утвержденных протоколов и контроле перекрестных связей. На практике такая вариативность опирается на строгое разделение платформы на обязательный базис и подключаемые надстройки. Базис задает общие правила сетевой логики, тогда как набор надстроек подбирается индивидуально под масштаб

вуза, специфику факультетов и текущий уровень автоматизации. По этой причине типовая модель воспринимается как гибкий каркас для разных сценариев сборки, а не жестко зафиксированный перечень программ.

Интеграция очередного модуля выполняется через типовые интерфейсы: новый компонент регистрируется в системе безопасности, подключается к общим базам и начинает передавать данные смежным сервисам. Обновление или полная замена одного решения на другое точно так же не требует переделки ядра при условии соблюдения обратной совместимости форматов данных.

Отключение сервиса предваряется тщательным анализом его связей. Если другие компоненты зависят от его работы, эти каналы необходимо заблаговременно перенаправить на резервную систему или аккуратно заглушить, предварительно переместив накопленные архивы в безопасное хранилище. Таким образом, адаптивность платформы выражается в управляемом развитии ее состава, а не в хаотичном удалении компонентов. Университет получает возможность разворачивать среду последовательно: начать с минимальной рабочей сборки, а затем постепенно наращивать функционал без болезненной смены фундамента. Это распределяет нагрузку на бюджет во времени, избавляет от зависимости от одного разработчика и защищает накопленные массивы информации, обеспечивая преемственность академических процессов.

Таким обазом, рассмотренная типовая структура электронного образовательного пространства выступает надежной основой для развертывания комплексной вузовской экосистемы, сопровождающей все стороны обучения и управления. Ее проектирование

базируется на четко зафиксированных функциональных и инженерных параметрах, обеспечивающих доступность, кибербезопасность, стабильность под нагрузками и комфорт пользователей. Краеугольным камнем архитектуры остается баланс между автономностью отдельных программных блоков и их глубокой координацией. Общие правила авторизации, согласованные справочники, строгая дисциплина источников информации и унифицированные шлюзы обмена превращают разрозненный софт в единый слаженный механизм.

Разделение архитектуры на фундаментальное ядро и гибкую периферию создает условия для постоянного обновления среды без масштабных перестроек. Добавление, вывод из эксплуатации или модификация подсистем производятся с оглядкой на сетевые зависимости, сохранность баз данных и стабильность учебных процессов. Данная модель предлагает не замкнутый готовый продукт, а фундаментальные правила построения расширяемой и адаптируемой цифровой экосистемы. Такой вектор развития принимает во внимание уникальные особенности каждого университета, облегчает поэтапный ввод сервисов, оптимизирует расходы на техническую поддержку и гарантирует непрерывность цифровой трансформации высшей школы.

Литература

1. Потапова, С. Интеграция электронных библиотечных систем в цифровую образовательную среду вуза: подходы к мониторингу и оценке эффективности / С. Потапова, Н. Амбросенко, О. Лысак. — Текст : электронный // Russian Journal of Education and Psychology. — 2025. — Т. 16, № 3. — С. 7–32. — DOI 10.12731/2658-4034-2025-16-3-694. — URL: (дата обращения: 17.11.2026).

2. Шамсутдинова, Т. М. Цифровой след как источник больших данных (Big Data) в образовании / Т. М. Шамсутдинова. — Текст : электронный // Открытое образование. — 2024. — Т. 28, № 6. — С. 13–21. — DOI 10.21686/1818-4243-2024-6-13-21. — URL: (дата

обращения: 17.11.2026).