

Научная статья

Статья в открытом доступе

УДК 519.65

doi: 10.30987/2658-6436-2022-4-70-78

АНАЛИЗ ПОДХОДОВ К ПОСТРОЕНИЮ АРХИТЕКТУРЫ ПРОГРАММНОГО КОМПЛЕКСА ПО УПРАВЛЕНИЮ РЕМОНТАМИ В СЕРВИСНЫХ КОМПАНИЯХ

Дмитрий Викторович Титарев¹, Сергей Олегович Кривцанов²

^{1,2} Брянский государственный технический университет, г. Брянск, Россия

¹ titaryovdv@mail.ru, <http://orcid.org/0000-0001-5502-2037>

² krivtsanovso@gmail.com, <http://orcid.org/0000-0002-7061-8856>

Аннотация. Изложен анализ подходов к построению архитектуры системы для сервисных компаний. Выполнен краткий обзор предметной области, выявлены проблемы таких компаний. Приведено обоснование, что задача выбора архитектуры программного комплекса по своей природе многокритериальна. Введены основные варианты архитектуры программного комплекса, для которых были выявлены и проанализированы их достоинства и недостатки, а также описаны критерии для выполнения их оценивания. Для решения задачи многокритериальной оптимизации был выбран метод взвешенной свертки типа «расстояние до идеала». В результате решения задачи было выяснено, что наиболее подходящей архитектурой программного комплекса для сервисных компаний является EAM-система с разработанной подсистемой службы Service Desk и гибкой интеграции с внешними Service Desk системами для расширения функционала при необходимости.

Ключевые слова: EAM, ITSM, ITIL, Service desk, управление ремонтами, управление сервисами, многокритериальная оптимизация

Для цитирования: Титарев Д.В., Кривцанов С.О. Анализ подходов к построению архитектуры программного комплекса по управлению ремонтами в сервисных компаниях // Автоматизация и моделирование в проектировании и управлении. 2022. №4 (18). С. 70-78. doi: 10.30987/2658-6436-2022-4-70-78.

Original article

Open Access Article

ANALYSING APPROACHES TO BUILDING THE SOFTWARE COMPLEX ARCHITECTURE FOR MANAGING REPAIR IN SERVICE COMPANIES

Dmitry V. Titarev¹, Sergey O. Krivtsanov²

^{1,2} Bryansk State Technical University, Bryansk, Russia

¹titaryovdv@mail.ru, ²krivtsanovso@gmail.com

Abstract. The paper presents an analysis of approaches to building an architecture system for service companies. A brief review of the subject area is made; the problems of such companies are identified. The article substantiates that the problem of choosing the software complex architecture is multi-criteria in nature. The main variants of the software package architecture are introduced, for which their advantages and disadvantages are identified and analysed, and the criteria for their evaluation are described. To solve the problem of multi-criteria optimisation, a weighted convolution method of the «distance to ideal» type is chosen. As a result, it is found that the most suitable software architecture for service companies is EAM-system with the developed Service Desk subsystem and flexible integration with external Service Desk systems to expand functionality in appropriate cases.

Keywords: EAM, ITSM, ITIL, Service desk, repair management, service management, multi-criteria optimization

For citation: Titarev D.V., Krivtsanov S.O. Analysing approaches to building the software complex architecture for managing repair in service companies. Automation and modeling in design and management, 2022, no. 4 (18). pp. 70-78. doi: 10.30987/2658-6436-2022-4-70-78.

Введение

Для удовлетворения потребностей современного человека компаниям приходится каждый день увеличивать качество своей продукции. Для того, чтобы это было возможно, они вынуждены автоматизировать свои процессы с помощью современных технологий и оборудования. Это обусловлено тем, что ручные операции увеличивают вероятность ошибок и уменьшают скорость производства, что означает потерю конкурентоспособности. Для

предотвращения таких ситуаций компании наращивают свои производственные мощности и, что немаловажно, модернизируют свое производство. Наращивание мощностей позволяет увеличить объем выпускаемой продукции, а благодаря модернизации компания улучшает качество продукции и делает ее более разнообразной.

Однако для обеспечения бесперебойной работоспособности любого оборудования его необходимо обслуживать. От качества обслуживания зависит срок эксплуатации. Пренебрежение данным фактом или обслуживание ненадлежащим образом приводят к серьезным техническим неисправностям, простоям оборудования или технологических процессов, если оборудование является частью конвейерного производства.

Оборудование влияет только на продукцию, а для эффективного функционирования всего предприятия в рабочем, исправном состоянии должны находиться все материальные активы: помещения, здания, инфраструктура и т.д.

Для решения этой проблемы используется методология ЕАМ, а также ЕАМ-системы [1, 2]. Данные системы предназначены для автоматизации бизнес-процессов учета и технического обслуживания и ремонта (ТОИР) основных фондов, позволяя уменьшить простои оборудования, сократить затраты на обслуживание, ремонты и материально-техническое обеспечение. Они нацелены на производственные предприятия, где компания сама использует оборудование и производит какие-то ценности.

При возникновении потребности в определенном материальном активе у предприятия есть выбор: приобрести его и поставить на собственный баланс или арендовать. Принимая решение, компания руководствуется принципами экономической целесообразности, и в случае больших затрат на приобретение аренда оказывается более предпочтительной.

Помимо производственных предприятий существуют так называемые сервисные компании, которые представлены практически во всех сферах жизни человека. Они специализируются на работе с клиентами и оказании им разного рода услуг. Одним из видов являются услуги по предоставлению оборудования, техники, инструментов и любых других активов в краткосрочную (для разового использования) и долгосрочную аренду, а также по оказанию клиентам технического обслуживания (как арендованных, так и собственных активов). Именно потребность в аренде оборудования и привела к появлению на рынке таких компаний.

В зависимости от оказываемых услуг данные компании делятся на три категории:

- компании, которые занимаются исключительно техническим обслуживанием активов клиентов;
- компании, владеющие некоторыми активами, которые предоставляют в аренду клиентам и ответственны за их техническое обслуживание;
- ситуацию, при которой и клиентом, и сервисной компанией (диспетчерской) являются части одной организации.

Можно заметить, что все данные компании/подразделения работают по принципу диспетчерской: получают сигнал о какой-то аварийной ситуации, обрабатывают его и отправляют бригаду на место. Обычно такие компании имеют как диспетчеров, отвечающих за получение информации от клиентов, так и ремонтный персонал, выполняющий обслуживание.

В любом случае они должны иметь возможность получать актуальную информацию о текущем состоянии оборудования для осуществления планового технического обслуживания или оперативного внепланового ремонта. Эту информацию необходимо получать непосредственно от клиентов, т.к. именно они эксплуатируют оборудование.

Таким образом, ЕАМ-системы подходят и для автоматизации бизнес-процессов компаний, специализирующихся на оказании услуг технического обслуживания. Проблема применения этих систем в данном случае заключается в необходимости получать информацию о техническом состоянии оборудования от клиента, т.к. сама компания непосредственно его не использует.

В методологии ЕАМ рекомендаций для решения этой задачи нет. В ней описан учет дефектов, которые выявляются во время осмотра или эксплуатации. Но методики, раскрывающей, как получать эту информацию от клиентов, нет. Это приводит к тому, что системы, реализованные на основе данной методологии, не имеют функционала предоставления клиентами информации о состоянии оборудования. Поэтому сервисным компаниям приходится вести учет обращений либо в бумажном виде, либо в отдельной

системе, после чего вручную переносить эту информацию в ТОИР-систему для инициации ремонта, а также самостоятельно отслеживать их выполнение и информировать клиентов.

Но при ручных операциях возрастает вероятность ошибки, а также значительно увеличивается время регистрации обращений и получения информации о них.

В связи с этим у сервисных компаний есть потребность в программном продукте, который предоставлял бы две связанные между собой подсистемы ТОИР и диспетчерской. Процессы по управлению услугами описаны в методологии ITSM, на их основе реализованы так называемые системы Service Desk. В ITSM содержится большое количество процессов, но для реализации диспетчерской службы сервисных компаний достаточно использовать следующие из них: управление обращениями, управление уровнями сервиса, управление каталогом услуг [5 – 10].

Задача выбора архитектуры

Один из первых и наиболее важных этапов разработки – выбор будущей архитектуры. Именно на нем закладываются основные концепции, которые в результате будут влиять на разработку, поддержку, удобство использования и скорость работы будущей системы. У каждого заказчика есть свои требования к системе и представления о том, как она должна работать и какой функционал предоставлять. Ввиду разных требований, зачастую не пересекающихся и не повторяющихся, у множества компаний универсальной архитектуры, подходящей всем, нет.

Поэтому если сервисной компании необходим функционал Service Desk систем, то они вынуждены реализовывать все с нуля. Этот подход имеет одно неоспоримое преимущество – в данном случае будут учтены все требования и выбрана наилучшая архитектура для конкретной компании. Но процесс разработки подобного решения является долгим и сложным (с точки зрения разработки), а также крайне дорогим. Поэтому его могут выбрать только крупные компании, которых не так много.

При разработке программного продукта для сервисных компаний необходимо учитывать, что это должно быть тиражное решение, т.е. подходить всем заказчикам. Система должна иметь минимально необходимый функционал, но достаточный для охвата всех основных потребностей сервисных компаний.

Но при таком подходе итоговая архитектура всегда является компромиссом, т.к. она не будет эталонной для каждой отдельно взятой компании, но позволит максимизировать среднее значение общей удовлетворенности всех заказчиков.

Таким образом, задача выбора архитектуры по своей природе является многокритериальной. Она может быть формализована в виде задачи многокритериальной оптимизации (в диалоге с экспертами) [11]. Для решения данной задачи необходимо выделить и обосновать альтернативы (архитектуры) и критерии, по которым они будут оцениваться.

Обзор возможных архитектур программного комплекса

Для эффективного управления ремонтами и техническим обслуживанием оборудования сервисному предприятию нужен программный комплекс, удовлетворяющий следующим требованиям:

- автоматизация бизнес-процессов ТОИР;
- автоматизация регистрации обращений пользователей;
- минимизация расходов на разработку, внедрение и обслуживание программного комплекса.

Таким образом, автоматизированный программный комплекс должен включать в себя как блок ТОИР, так и службу технической поддержки. Данные блоки могут быть представлены как отдельной системой, так и встроенной подсистемой/модулем. Исходя из этого может быть предложено три подхода к построению архитектуры для разработки проектируемого программного комплекса, отличающиеся концептуально друг от друга сущностью блоков.

Так как в программном комплексе должны быть реализованы и взаимосвязаны две методологии, то требуется проанализировать различные архитектуры, подразумевающие использование каждой из методик в той или иной форме. Необходимо уточнить, что при рассмотрении архитектур, в которых одна из методологий представлена в качестве

отдельного блока/подсистемы, подразумевается реализация не всех процессов, описанных в стандартах, а лишь небольшой части. Это связано с тем, что все затраты на разработку, приобретение и внедрение необходимо минимизировать, а в случае реализации полноценной системы в качестве отдельного блока будет получен обратный эффект. В данном случае можно взять готовую вторую систему и реализовать с ней интеграцию.

Независимые EAM и ITSM системы. В данном случае программный комплекс состоит из двух независимых систем (рис. 1), обменивающихся информацией.

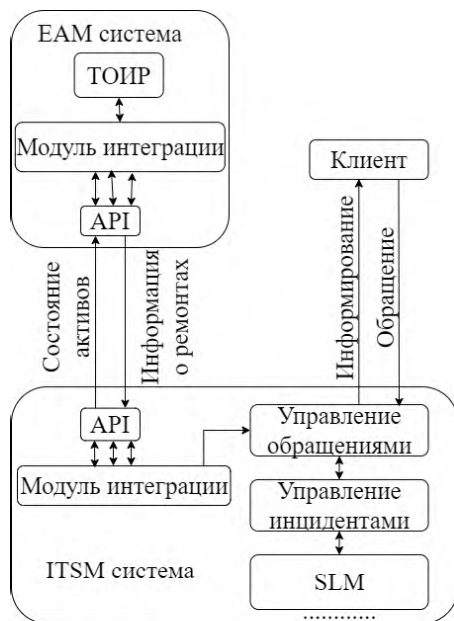


Рис. 1. Архитектура программного комплекса с двумя независимыми системами
Fig. 1. Architecture of the software package with two independent systems

ITSM система используется для управления обращениями клиентов. Клиент регистрирует обращение любым удобным и доступным ему способом, после чего эта информация отправляется в ТОИР систему.

EAM система автоматизирует бизнес-процессы по управлению ремонтами и техническим обслуживанием оборудования. Информация о плановых мероприятиях хранится в самой системе, извне поступает только информация от клиентов о неисправностях оборудования для оперативного выполнения ремонта.

Помимо этого, для корректного функционирования программного комплекса необходима настроенная интеграция между двумя системами для передачи информации: после регистрации обращения информация переносится в EAM систему, после выполнения ремонта в ITSM систему отправляется факт окончания работ для дальнейшего информирования клиента.

При такой архитектуре есть несколько вариантов работы с программным комплексом.

В первом варианте подразумевается полноценная работа сотрудников с системами. С ITSM-системой работают диспетчера: обрабатывают запросы клиентов, регистрируют обращения, получают обратную связь от них. В EAM-системе работает ремонтный персонал: автоматически создаются нужные документы на основании информации из другой системы (с помощью регулярного обмена данными), которые они используют в процессах управления ремонтами. В данном случае используется весь функционал обеих систем, что позволяет в автоматическом режиме определять необходимые для анализа параметры, а также управлять качеством оказываемых услуг.

Второй вариант использует так называемую бесшовную интеграцию, которая позволяет в интерфейсе одной системы просматривать и работать с данными другой. Применительно к проектируемому программному комплексу это реализуется с помощью веб-сервиса, который через API получает необходимую информацию из ITSM-системы. То есть в EAM-системе в отдельном окне отображается информация о запросах клиентов, с помощью определенных команд можно оперировать этими данными (так же с помощью веб-сервиса). Вторая система работает в «фоновом» режиме, получение информации происходит с помощью

альтернативных каналов связи (почта, мобильное приложение, чат-боты в мессенджерах, либо диспетчером в EAM-системе), и используется для выполнения сложных вычислений (расчет крайнего срока и т.д.), чтобы не разрабатывать эти механизмы в системе управления ремонтами.

Таким образом, проектируемый программный комплекс использует две готовые системы, поэтому нет необходимости их дорабатывать. Главной задачей является разработка и настройка интеграции, что оказывается более выгодным с точки зрения затрат вариантом по сравнению с другими подходами.

С другой стороны, приобретение сразу двух систем потребует больших расходов со стороны клиента. Это обусловлено тем, что одна полноценная система будет стоить хоть и дороже, чем каждая по отдельности, но заметно дешевле чем две вместе (также для каждой придется приобретать лицензии и т.д.). А стоимость дальнейшего обслуживания программного комплекса делает такую архитектуру экономически нецелесообразной. Также в базовом варианте для решения поставленной задачи не требуется весь функционал ITSM системы: необходима возможность регистрации обращений клиентов и связывание ее с бизнес-процессами ТОИР, а нагруженный интерфейс с лишними возможностями снизит пользовательскую удовлетворенность. Таким образом, для среднестатистической сервисной компании программный комплекс с такой архитектурой не подходит.

ITSM система с модулем ТОИР. В данном случае программный комплекс состоит из одной ITSM системы, для которой разработан модуль ТОИР (рис. 2). Работа с программным комплексом аналогична предыдущему варианту, за исключением того, что информация о регистрации обращения и выполнении работ сразу становится доступной в соответствующей подсистеме и не требует передачи в другую систему.

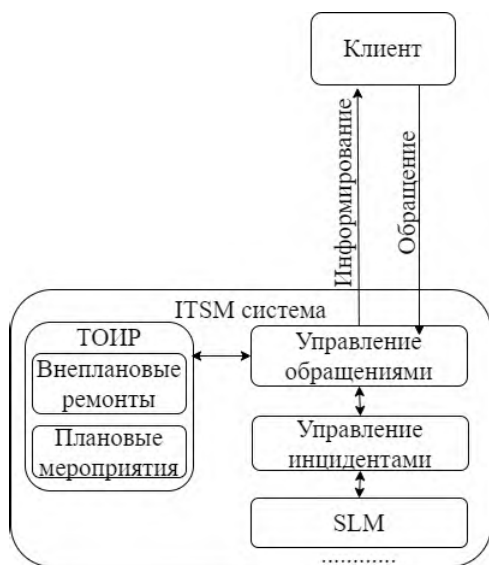


Рис. 2. Архитектура программного комплекса, представленная ITSM системой с модулем ТОИР
Fig. 2. The architecture of the software package, represented by the ITSM system with the EAM module

Такая архитектура программного комплекса экономически нецелесообразна по следующим причинам:

- приобретение целой системы, большая часть функционала которой не нужна;
- необходим развитый функционал ТОИР, что потребует долгой и дорогостоящей разработки нового модуля.

EAM система с модулем Service Desk. В отличие от предыдущего варианта в данной архитектуре основной является EAM система, имеющая отдельный модуль Service Desk [10] для управления обращениями клиентов (рис. 3).

Кроме упрощенного интерфейса дополнительного модуля, для пользователя отличий от предыдущей архитектуры нет. С финансовой точки зрения данная архитектура наиболее предпочтительна по следующим причинам:

- приобретение одной системы, все возможности которой будут использованы;
- быстрая и недорогостоящая разработка нового модуля;
- быстрое внедрение и последующее недорогое обслуживание.

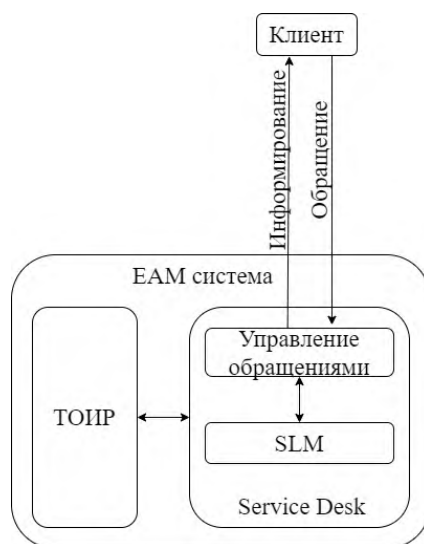


Рис. 3. Архитектура программного комплекса, представленная EAM системой с модулем Service Desk

Fig. 3. The architecture of the software package, represented by the EAM system with the Service Desk module

Критерии

Период внедрения. Время, затраченное на внедрение системы в инфраструктуру компании и ввод ее в эксплуатацию, напрямую зависит от архитектуры: если она монолитная, т.е. когда обе подсистемы реализованы в рамках одной системы, время будет меньше, т.к. для двух независимых систем в большинстве случаев понадобится подготовить два отдельных сервера, настроить их, установить и настроить системы.

Трудозатраты (человеко-часы) на разработку. При выборе архитектуры с двумя независимыми системами количество трудозатрат на разработку гораздо меньше, т.к. необходимо реализовать только интеграцию, с помощью которой они будут синхронизировать данные. А при разработке на базе одной из двух систем нужно будет полностью с нуля реализовать отдельный блок, что гораздо дольше, чем разработка интеграции.

Стоимость (внедрения и поддержки). Две независимые системы дороже и при внедрении, и при поддержке. В этом случае и лицензии нужно будет покупать для двух систем, и оформлять поддержку. А так как скорее всего компании-разработчики разные, то стоимость будет гораздо больше, чем для монолитной системы. Но, если заказчик решит использовать альтернативные каналы регистрации обращений (например, мобильное приложение, или сайт), то такие альтернативы будут дороже варианта программного комплекса с двумя независимыми системами.

«Порог вхождения». В готовых Service Desk системах используется терминология из ITSM, но в большинстве случаев с ней будут работать диспетчера, которые не знакомы с методологией. Поэтому при разработке отдельной подсистемы Service Desk данное требование будет учтено, что является преимуществом.

Удовлетворенность работой системы. У двух независимых систем лучше отказоустойчивость: если на одном из серверов нужно провести технические работы или выполнить обновление системы, что приведет к ее недоступности какое-то время, то вторая система будет работать и ею можно пользоваться (особенно это важно для диспетчерской, т.к. работают непосредственно с клиентами, которые не знают о каких-то неисправностях).

Решение задачи многокритериальной оптимизации

Для решения задачи многокритериальной оптимизации был выбран метод взвешенной свертки типа «расстояние до идеала» [11], которая в качестве оптимального решения выбирает то, которое ближе всех расположено к некоторому «эталону качества» в пространстве критериальных оценок. Возможны и другие методы, которые выбираются в зависимости от особенностей системы предпочтений ЛПР. Ниже приведена формальная модель задачи.

Альтернативы:

1. Разработка программного комплекса, состоящего из двух независимых систем – ЕАМ и Service Desk – между которыми настроен регулярный обмен (x_1).
2. Разработка программного комплекса на базе системы Service Desk с подсистемой ТОИР, тесно связанной с бизнес-процессами по оказанию услуг (x_2).
3. Разработка типового универсального модуля Service Desk для минимизации трудозатрат на разработку при внедрении проекта (x_3).
4. Разработка программного комплекса на базе ЕАМ-системы с модулем службы Service Desk, тесно связанным с бизнес-процессами ТОИР. Разработка гибкой интеграции с внешними Service Desk системами для расширения функционала системы (x_4).
5. Разработка мобильного приложения для клиентов сервисной компании (x_5).
6. Альтернатива, объединяющая решения 1 и 5 (x_6).
7. Альтернатива, объединяющая решения 2 и 5 (x_7).

Критерии:

1. Период внедрения – f_1 .
2. Трудозатраты (чел. ч) – f_2 .
3. Стоимость (внедрения и поддержки) – f_3 .
4. «Порог вхождения» – f_4 .
5. Удовлетворенность – f_5 .

Выставление оценок по каждому критерию выполнялось сотрудником компании-разработчика ТОИР-системы, который является консультантом и занимается поддержкой, т.е. непосредственно взаимодействует с пользователями, отвечает на их вопросы, разбирается в их проблемах. Консультант имеет представление о потребностях заказчиков и может дать оценку тому или иному критерию для каждой из альтернатив исходя из своего опыта. В табл. 1 приведены выставленные оценки. Использовалась шкала от 0 до 10 для выставления более точной оценки.

Сопоставление альтернатив с целью

Таблица 1

Comparison of alternatives with the goal

Table 1

Альтернативы	Период внедрения	Трудозатраты (чел. ч)	Стоимость	«Порог вхождения»	Удовлетворенность
1	2	2	6	5	4
2	7	7	8	8	8
3	10	9	0	0	7
4	10	10	10	7	9
5	0	0	10	5	8
6	4	4	7	5	6
7	7	7	10	7	9

Решение задачи. Для решения задачи необходимо определить вес каждого критерия. Они были оценены методом пропорциональных коэффициентов в диалоге с экспертом: $f_1 = 0,267$, $f_2 = 0,2$, $f_3 = 0,333$, $f_4 = 0,067$, $f_5 = 0,133$.

Для каждого критерия будем считать «идеальной» наилучшую из имеющихся по нему оценок. С учетом этого перейдем от исходных критериев f к вспомогательным критериям φ по формуле:

$$\varphi_j(x_i) = |f_j(x_i) - f_j^*|$$

Далее производим нормировку отклонений, поделив эти значения на «идеал» по каждому критерию.

В конце применяем оператор свертки расстояния до идеала (табл. 2). Так как данный критерий минимизируется, то в результате получаем альтернативу x_4 .

Результат задачи многокритериальной оптимизации

Таблица 2

Result of the multicriteria optimization problem

Table 2

Альтернативы	1	2	3	4	5	6	7
Результат	0,703	0,246	0,651	0,032	0,694	0,505	0,208

Выводы

Как видно, лучшей альтернативой является проектирование программного комплекса на базе EAM-системы для сервисных компаний и разработка подсистемы службы Service Desk, тесно связанной с бизнес-процессами ТОИР. Также данная система должна иметь гибкую интеграцию с внешними Service Desk системами для расширения функционала системы. Это требуется для того, чтобы модуль был небольшим и имел только основной функционал, связанный с выделенными из методологии ITSM процессами. Интеграция позволит при необходимости вести учет обращений в другой системе, в которой будет реализован весь функционал. При этом для клиентов должны быть реализованы разные альтернативные каналы для создания обращений, чтобы удовлетворить максимальное количество пользователей.

Данная альтернатива имеет следующие преимущества:

- приобретение одной системы, все возможности которой будут использованы;
- быстрая и недорогостоящая разработка нового модуля;
- быстрое внедрение и последующее недорогое обслуживание.

Проектируемый программный комплекс должен улучшить сферу услуг, связанную с арендой и техническим обслуживанием оборудования, путем автоматизации и цифровизации бизнес-процессов сервисных компаний. Это позволит повысить их уровень сервиса, что скажется на удовлетворенности клиента.

СПИСОК ИСТОЧНИКОВ

References:

1. Enterprise Asset Management. Системы управления основными фондами предприятия: [Электронный ресурс]. URL: <http://www.tadviser.ru/index.php/EAM> (Дата обращения: 21.09.2020).
2. Design and implementation of EAM system based on Web Service: [Электронный ресурс]. URL: <https://ieeexplore.ieee.org/document/6013637> (Дата обращения: 03.10.2020).
3. CMMS система: [Электронный ресурс]. URL: <http://www.novosoft.ru/nerpa/cmms-sistema.shtml> (Дата обращения: 21.09.2020).
4. Automated inspection planning system for CMMS: [Электронный ресурс]. URL: <https://ieeexplore.ieee.org/document/6396139> (Дата обращения: 02.10.2020).
5. IT Service Management. Системы управления ИТ-службой: [Электронный ресурс]. URL: <http://www.tadviser.ru/index.php/ITSM> (Дата обращения: 20.09.2020).
6. ITSM и ITIL. Как использовать? В чем отличия и суть?: [Электронный ресурс]. URL: <https://okdesk.ru/blog/itsmitil> (Дата обращения: 20.09.2020).
7. Increasing Information Systems Availability Through Accuracy, Awareness, Completeness and Manageability of ITSM: [Электронный ресурс]. URL: <https://ieeexplore.ieee.org/document/8760686> (Дата обращения: 02.10.2020).
8. Ковалев, А.В. Доступный ITIL. Настольная книга ИТ руководителя. Часть 1. Эксплуатация сервисов. / А.В. Ковалев. – Москва: Тезаурис, 2018. – 450 с.
9. Эвес, Д. ITIL Поддержка услуг. / Д. Эвес, Ж. Пойнтер. – London: TSO, 2005. – 418 с.
10. ITIL для не ИТ-подразделений: [Электронный ресурс]. URL: <https://cleverics.ru/digital/2016/07/itil-dlya-ne-it-podrazdelenij/> (Дата обращения: 21.09.2020).
11. Андрейчиков А.В., Андрейчикова О.Н. Наука и искусство принятия решений. Книга 1: Принятие решений: Условия определенности. Условия риска. - М.: Ленанд, 2021. - 240 с.
1. Enterprise Asset Management [Internet] [cited 2020 Sep 21]. Available from: <http://www.tadviser.ru/index.php/EAM>
2. Design and Implementation of EAM System Based on Web Service [Internet] [cited 2020 Oct 3]. Available from: <https://ieeexplore.ieee.org/document/6013637>
3. CMMS System [Internet] [cited 2020 Sep 21]. Available from: <http://www.novosoft.ru/nerpa/cmms-sistema.shtml>
4. Automated Inspection Planning System for CMMS [Internet] [cited 2020 Oct 2]. Available from: <https://ieeexplore.ieee.org/document/6396139>
5. IT Service Management [Internet] [cited 2020 Sep 20]. Available from: <http://www.tadviser.ru/index.php/ITSM>
6. Service Management (ITSM) vs ITIL: What's the Difference? [Internet] [cited 2020 Sep 20]. Available from: <https://okdesk.ru/blog/itsmitil>
7. Increasing Information Systems Availability through Accuracy, Awareness, Completeness and Manageability of ITSM [Internet] [cited 2020 Oct 2]. Available from: <https://ieeexplore.ieee.org/document/8760686>
8. Kovalev AV. Accessible ITIL. Desk Book of IT Manager. Moscow: Thesaurus; 2018. Part 1, Service Operation.
9. Ewes D., Pointer J. ITIL Service Support. London: TSO; 2005.
10. ITIL for non-IT Departments [Internet] [cited 2020 Sep 9]. Available from: <https://cleverics.ru/digital/2016/07/itil-dlya-ne-it-podrazdelenij/>
11. Andreichikov A.V., Andreichikova O.N. Science and Art of Decision Making. Book 1: Decision Making: Conditions of Certainty. Conditions of and Risk. Moscow: Lenand; 2021.

Информация об авторах:

Дмитрий Викторович Титарёв

кандидат технических наук, доцент кафедры «Информатика и программное обеспечение» ФГБОУ ВО «Брянский государственный технический университет», ORCID: 0000-0001-5502-2037.

Сергей Олегович Кривцанов

магистрант кафедры «Информатика и программное обеспечение» ФГБОУ ВО «Брянский государственный технический университет», ORCID: 0000-0002-7061-8856.

Information about authors:

Dmitry Viktorovich Titarev

Candidate of Technical Sciences, Associate Professor of the Department «Computer Science and Software» of the Federal State Budgetary Educational Institution of Higher Education «Bryansk State Technical University», ORCID-iD: 0000-0001-5502-2037.

Sergey Olegovich Krivtsanov

master's student of the Department «Computer Science and Software» of the Federal State Budgetary Educational Institution of Higher Education «Bryansk State Technical University», ORCID-iD: 0000-0002-7061-8856.

Вклад авторов: все авторы сделали эквивалентный вклад в подготовку публикации.

Contribution of the authors: the authors contributed equally to this article.

Авторы заявляют об отсутствии конфликта интересов.

The authors declare no conflicts of interests.

Статья поступила в редакцию 08.09.2022; одобрена после рецензирования 28.10.2022; принята к публикации 04.11.2022.

The article was submitted 08.09.2022; approved after reviewing 28.10.2022; accepted for publication 04.11.2022.

Рецензент – Лозбинец Ф.Ю., доктор технических наук, профессор, Брянский филиал Российской академии народного хозяйства и государственной службы при Президенте Российской Федерации.

Reviewer – Lozbinets F.Yu., Doctor of Technical Sciences, Professor, Bryansk Branch of Russian Presidential Academy of National Economy and Public Administration.