

УДК 004.054
DOI: 10.15827/0236-235X.140.533-540

Дата подачи статьи: 14.07.22, после доработки: 03.09.22
2022. Т. 35. № 4. С. 533–540

Организационные проблемы реализации гибких подходов в разработке прикладного программного обеспечения

О.В. Саяпин¹, д.т.н., профессор, tmaec@mail.ru

О.В. Тиханьчев², к.т.н., зам. начальника отдела управления перспективных разработок, towb5@yandex.ru

А.А. Безвесильная¹, к.п.н., зав. кафедрой, a.bezvesilnaia@amchs.ru

¹ Академия гражданской защиты МЧС России, г. Химки, 141435, Россия

² Компания «Т1 Интеграция», г. Москва, 111395, Россия

В статье рассмотрены особенности каскадных и гибких подходов к организации разработки прикладного программного обеспечения автоматизированных систем управления, выявлены их положительные стороны и недостатки. Особенно критичными из недостатков представляются увеличение сроков разработки и слабое взаимодействие между заказчиком и разработчиком при использовании каскадного подхода. В то же время существующими нормативными документами именно такой подход определяется как основной.

С использованием общенаучных методов анализа и синтеза обеспечено получение количественных и качественных оценок по времени и ожидаемому результату применения при разработке ПО каскадного и гибкого подходов. По результатам сравнения полученных оценок сформирован вывод о том, что рациональным решением может служить замена каскадного подхода на гибкий. В то же время, как показал анализ нормативно-технической документации, применению при разработке прикладных программ гибких подходов препятствуют имеющиеся организационные проблемы, связанные не только с требованиями нормативных документов, но и со сложностью координации работ, выполняемых распределенными коллективами исполнителей. По результатам анализа типового процесса разработки прикладного ПО автоматизированных систем управления авторами сформулированы предложения о возможных вариантах замены каскадных подходов на гибкие или комбинированные.

Новизна предложенного подхода к организации разработки заключается в обеспечении гибкости реализации требований заказчика, а его внедрение позволит выстроить систему разработки, обеспечивающую повышение заинтересованности всех участников процесса в конечном результате и осуществляющую процесс непрерывного уточнения требований.

Ключевые слова: автоматизация управления, разработка программного обеспечения, гибкая методология разработки, поддержка принятия решений, организационные проблемы.

Автоматизация управления считается одним из основных методов повышения его эффективности. Материальной основой процесса автоматизации служит использование АСУ, включающих в свой состав технические и программные компоненты. В свою очередь, программные компоненты, особенно прикладные программы, являются средствами, обеспечивающими выполнение основных функций управления и придающими системе способность настраиваться под требования пользователей. Но, несмотря на накопленный опыт разработки ПО, широкое разнообразие имеющихся методов и технологий в данной области, в настоящее время остается ряд проблем, сдерживающих развитие автоматизации управления.

Анализ состояния предметной области и практический опыт участия авторов в разработке АСУ позволяют сделать вывод, что су-

щественная часть проблем разработки прикладного ПО (ППО) как компонента, обеспечивающего основной функционал пользователей, определяется методологией организации разработки. А конкретнее, использованием при разработке ПО каскадного подхода, оставшегося в наследство от разработки продукции машиностроения и регулируемого ГОСТами серии 34, специализированными ГОСТами по областям техники и другими нормативными и регламентирующими документами [1–3].

Отказ от каскадного подхода, переход к активно применяемому и показавшим практическую эффективность гибким технологиям разработки [4–6] может потребовать отказа от использования указанных документов или их существенной переработки. Однако, несмотря на возможные сложности, отказ от каскадных подходов может в перспективе повысить эф-

фективность разработки прикладных программ.

Проблема существующего подхода к организации разработки ППО

Сущность проблемы использования устаревших методологий разработки ПО предлагается рассмотреть, сравнивая каскадные и гибкие подходы к организации этого процесса. Сравнительный анализ показывает, что данные подходы имеют ряд существенных отличий, влияющих на разработку ПО.

Во-первых, это различия в сроках разработки, влияющие на время внедрения новых методик управления, разрабатываемых для реализации в составе ПО АСУ.

В настоящее время данные сроки определяются задаваемым руководящими документами каскадным порядком разработки ППО. Обработка статистических данных по нескольким десяткам НИОКР по созданию АСУ промышленными объектами и транспортом, выполняемых в рамках госзаказа, с применением классического каскадного подхода и их сравнение со сроками обновления коммерческих программных продуктов класса ERP, таких как SAP ERP ECC, SAP NetWeaver 2004, SAP S/4HANA, Oracle ERP Cloud, Microsoft Dynamics 365, 1С.ERP «Управление предприятием», создаваемых и модернизируемых по гибким технологиям (см. таблицу), позволяет сделать вывод,

что использование гибких методов потенциально может изменить длительность работ как по этапам разработки, так и в целом. Для формирования численных оценок использовались известные статистические формулы расчета математического ожидания и дисперсии времени выполнения работ, полученные из открытых источников. Впрочем, сущность использования гибких подходов даже не в сроках – их сокращение является необязательным следствием внедрения новых методологий, а в ожидаемом результате разработки, определяемом в том числе возможностью вносить изменения в продукт на любом этапе. Сформированная экспертная оценка позволяет сделать вывод о повышении степени соответствия разработанного продукта актуальным на момент сдачи работы потребностям заказчика, что и должно являться главным итогом предлагаемых изменений.

Во-вторых, имеются существенные отличия в части гибкости реализации требований заказчика и/или функций пользователя при их изменении в ходе разработки и эксплуатации системы, при уточнении реализованного ранее в составе ПО АСУ математического аппарата.

В настоящее время по ГОСТу 15.304 доработка программ осуществляется в ходе авторского надзора в минимальном объеме с целью устранения недостатков. Серьезная модификация заложенного математического аппарата действующими нормативно-техническими до-

Сравнительный анализ сроков разработки ППО для разных подходов к организации процесса

Comparative analysis of the applied software development time for different approaches to process organization

Существующий подход		Гибкая методология (RUP)	
Этап	Срок	Этап	Срок
Выявление заказчиком проблемы и задание поисковой НИР	Минимум полгода	Выявление заказчиком проблемы, формулировка задачи поиска решения и разработка прототипа	Не более полугодом
Выполнение НИР по формированию исходных данных для предприятий промышленности	Минимум год	Апробация прототипа, анализ результатов и разработка макета	До полугодом
Организация конкурса на выполнение опытно-конструкторской работы (ОКР)	До полугодом	Организация конкурса на выполнение ОКР	До полугодом
Выполнение ОКР предприятиями промышленности	Минимум два года	Выполнение ОКР по доработке макета и его адаптации в ПО АСУ	Год
Результат			
Итого: общий срок от 4 лет и более, в соответствии с требованиями, сформулированными на начальном этапе, за 3,5 года до сдачи продукта		Итого: около 2,5 года, с постоянной актуализацией требований и функционала должностных лиц в ходе выполнения работы	

кументами не предусмотрена. Как уже отмечалось, ситуация сложилась исторически в силу того, что во времена разработки этих документов не были развиты гибкие методы, а существующие ГОСТы по разработке ПО создавались на основе применяемых ранее документов по проектированию и постановке на производство изделий машиностроения. Для них выполнение требований проводилось в основном на этапе НИР и аванпроектов, так как управлять требованиями заказчика на этапе создания опытного образца было проблематично и затратно. Совсем другая ситуация складывается при разработке ПО, когда проведение доработки разрабатываемого продукта теоретически возможно до самого последнего этапа работ – проведения приемо-сдаточных испытаний. Но реализации этой возможности на практике препятствует действующая *нормативно-техническая и регламентирующая документация* (НТД и РД).

В сложившейся ситуации при появлении новых или уточнении ранее сформулированных требований заказчика, отличающихся от заложенных в документацию, требуется открывать новые НИР и ОКР.

При использовании гибкого подхода, как показывает практика, модификация заложенного в ППО математического аппарата может производиться до последнего этапа разработки, превращая дискретный процесс совершенствования ПО АСУ практически в непрерывный.

Это особенно актуально при разработке ППО для АСУ специального назначения. В настоящее время она проводится по техническому заданию и постановкам на разработку задач. В существующей НТД и РД этот процесс организован так, чтобы максимально полно сформулировать требования к ППО на начальном этапе разработки. Но, в отличие от ситуаций разработки общего ПО, пользователем которого отчасти является и сам разработчик, в случае ППО АСУ специального назначения имеется специфика предметной области, недоступная разработчику. А определяемый НТД и РД порядок разработки, ориентированный на каскадный подход, ограничивает возможности уточнения и детализации требований по мере появления проблемных вопросов. В то же время есть вероятность, что именно гибкие подходы могут решить эту проблему [7, 8].

В-третьих, различные методологии по-разному обеспечивают уровень заинтересованности всех участников процесса разработки в конечном результате.

В существующей структуре нормативной и регламентирующей документации, нацеленной на каскадный подход, каждый исполнитель отвечает только за качество реализации своего этапа: постановки задачи в виде технического задания, разработки постановок задач, программы, пользовательской и рабочей конструкторской документации. При гибком подходе, как показывает мировая практика разработки ПО, системно работают взаимодействующие команды, решая проблемы совместно.

Указанные проблемы, на взгляд авторов, порождаются тем, что организация работ в рамках каскадного подхода имеет следующие особенности:

- разрыв между постановкой задачи и контролем результата выполнения по каждой отдельной программе;

- относительно автономная разработка компонентов программ, часто приводящая к отсутствию системности результата;

- участие в составлении и процессе согласования постановок на разработку прикладных программ ограниченного круга специалистов со своими требованиями и пожеланиями, обладающих ограниченными возможностями по работе в смежных областях и по взаимодействию с другими участниками процесса.

Как показывает анализ практического опыта разработки ПО, указанные проблемы могут быть решены в рамках гибкого подхода. При использовании гибких методов, как правило, распределенно работают смешанные команды, формируемые на каждом этапе под конкретный результат, в том числе с привлечением заказчика и конечного потребителя. Соответственно, обеспечено привлечение широкого круга специалистов, заинтересованных в достижении результата на всех этапах разработки.

При использовании гибких методов разработки круг участников расширяется, на всех этапах можно выяснять, что возможно сделать на практике, а что нет, какие дополнительные опции можно реализовать, оперативно решать вопросы системного характера. Сейчас делать это проблематично, так как использование гибкого подхода не предусмотрено существующими НТД и РД.

Следует отметить еще одну особенность использования различных подходов к разработке: при каскадном подходе окончание работы определяется сроками ОКР, а при гибком – достижением результата. Последнее с высокой вероятностью гарантирует успешное заверше-

ние работы, удовлетворяющее потребности заказчика.

Преимущества и проблемы гибкого подхода

Таким образом, практика показывает существенные преимущества гибких методов разработки ПО АСУ перед каскадными. Но в их реализации могут возникать и проблемы, определяемые как структурой разрабатываемых систем, так и особенностями самого гибкого подхода:

- гибкий подход сложнее в организации и финансировании;
- требуется периодическое отвлечение заказчика от основной деятельности для участия в процессе разработки;
- результаты работы могут отклоняться от ожиданий заказчика, сформулированных при задании разработки.

Последнее, впрочем, чаще случается именно при использовании каскадного подхода, причем в более критичном варианте и с большей вероятностью. А в случае использования гибких подходов данное несоответствие в большинстве случаев оказывается некритичным, так как заказчик актуализирует требования в процессе разработки и соответствие продукта его первоначальным пожеланиям становится необязательным. Остается только зафиксировать данный факт документально.

Вторую из описанных проблем можно считать некритичной: затраты времени заказчика окупятся тем, что он получит систему, соответствующую его потребностям.

Наиболее существенной представляется первая из перечисленных проблем, особенно актуальная при разработке сложных систем, включающих различные виды программных и технических компонентов, особенно, когда разработке или модернизации подлежат сразу несколько из них. Такая ситуация нетипична

при разработке коммерческих ERP- или CRM-систем, но часто возникает при разработке АСУ специального назначения.

Именно подобная ситуация порождает одну из сложностей в применении гибких подходов – соблюдение баланса между интересами разработчика и потребностями заказчика, определяемое тем, что в рамках гибкого подхода к разработке и те, и другие непрерывно уточняются. И при этом, учитывая сложную структуру ПО и определенную инертность процесса его разработки, необходимо соблюдать тонкую грань между гибкостью управления и переходом процесса в хаос.

Как показывает практика разработки АСУ, соблюдение этого баланса может быть обеспечено только реализацией скоординированных по времени и задачам мероприятий при планировании и в ходе выполнения НИОКР. Вариант соотношения мероприятий по разработке приведен на рисунке. Для наглядности ППО разделено на *специальное ПО* (СПО) и *общесистемное ПО* (ОСПО), как это сделано в некоторых ведомственных ГОСТах.

При формировании данных предложений по организации разработки (см. рисунок) учтено, что при использовании гибкого подхода (в отличие от каскадного) построение мероприятий не является заранее спланированной линейной последовательностью операций. Последовательность определяется требованием периодического уточнения функций различных компонентов ПО, обуславливающих смену базы, относительно которой происходят доработки:

- выбор ОПО, обеспечивающего функционирование разрабатываемых программ на применяемом комплекте технических средств;
- разработка (доработка) ОСПО как средства информационной поддержки принятия решений и информационного фундамента для функционирования расчетных задач и моделей СПО;



– разработка СПО как основного средства обеспечения поддержки принятия управленческих решений с одновременной доработкой взаимодействующих с ним компонентов ОСПО и при крайней необходимости ОПО.

При организации разработки ПО АСУ с использованием гибкого подхода возникает важная научно-практическая задача – определение соотношения по времени и объему между этапами разработки и доработки разных компонентов АСУ. Ее решение потенциально обеспечивает организацию разработки, позволяющую реализовывать меняющиеся требования заказчика при соблюдении интересов разработчика. При анализе возможности и выборе методов решения сформулированной задачи предлагается учесть практику работ по информатизации различных прикладных процессов [9–11].

Возможное решение задачи использования гибких подходов

Сформулированная в статье задача рациональной организации разработки и доработки компонентов ПО АСУ при использовании гибких подходов может быть решена на основе анализа практики выполнения подобного рода работ.

Определенные примеры реализации гибких подходов в разработке СПО АСУ специального назначения, хотя и в неявном виде, можно найти и в настоящее время. Например, при планировании работ сроки проведения приемосдаточных испытаний иногда намеренно увеличивают, чтобы иметь возможность устранять замечания и выполнять рекомендации представителей заказчика в ходе приемки, дорабатывая ПО под актуальные на момент испытаний требования пользователя. При этом разработчик старается привлекать заказчика уже к самым ранним этапам испытаний, которые проводятся еще на предприятии, – к предварительным испытаниям. Это, хоть и не явный, но вполне работоспособный элемент гибких подходов, реализуемый на практике явочным порядком. Пусть формально не узаконенный, но повышающий эффективность разработки и в очередной раз доказывающий преимущество гибкости организации процесса создания СПО АСУ, а также возможности реализации гибких подходов на практике. Анализ указанной информации, ее сопоставление с опытом разработки коммерческого ПО позволяют сделать вывод: рационально при планировании предусмотреть этапы доработки ОПО и СПО как ба-

зовых элементов каждого последующего этапа (см. рисунок) с длительностью 5–20 % от запланированной продолжительности этапа. Конкретные значения длительности этапов доработки могут быть оценены исходя из показателей новизны разрабатываемого ППО, определяемой в ходе предпроектных исследований в рамках технико-экономического обоснования работ при планировании НИОКР.

Примеры применения AGILE-методологии, то есть того же гибкого подхода, в разработке ПО коммерческого назначения, при создании промышленных ERP- и CRM-систем, с одной стороны, являются более очевидными и наглядными [11–13]. С другой стороны, необходимо отметить, что коммерческое ПО обычно проще, чем ППО для АСУ в части ограничений на процесс создания: при его разработке, как правило, используются готовые компоненты общего ПО без их доработки, практически нет ограничений на привлечение экспертов предметной области и широкого круга потенциальных пользователей для проведения α - и β -тестирования, менее критичны запреты по конфиденциальности работ. Но функции коммерческого продукта по масштабу и сложности обычно не уступают промышленному ППО, поэтому учет опыта его разработки при уточнении принципов использования гибкого подхода представляется вполне корректным, в том числе при разработке подходов к координации выполняемых работ. Например, в части использования системного подхода, когда итерации (scrum) разработки ППО выстраиваются не по отдельным последовательно разрабатываемым программным компонентам, а как этапы создания АСУ в целом, параллельно усложняя все входящие в систему компоненты от прототипов экранных форм до полнофункциональных задач с учетом связей между ними. Это только один из возможных подходов к совершенствованию организации процесса разработки, реализуемого в рамках гибкой методологии, практика подскажет и другие методы, потенциально применимые для решения сформулированной задачи [14, 15].

Заключение

Как показывает практика, использование зарекомендовавших себя RUP-стандартов и гибких методов организации разработки может повысить эффективность реализации имеющегося и вновь создаваемого научно-техниче-

ского задела в части автоматизации управления и обеспечения поддержки принятия решений.

В то же время реализация гибкого подхода при разработке ППО сопряжена с определенными проблемами. Наиболее значимая из них – сложность организации процесса разработки, особенно для систем, требующих в процессе разработки ППО модернизации общего ПО. Возможно, практическое решение данной проблемы даже потребует наложить некоторые ограничения на гибкость подхода к разработке ПО, сделав его не каскадным или гибким в чистом виде, а комбинированным. Может быть, придется не только менять содержание и продолжительность этапов разработки, но и совместить некоторые из них, например, разработку и тестирование, изменив при этом методы контроля качества на разработку через тестирование. Конкретные меры подскажет практика. Потенциальные трудности компенсируются тем, что в итоге заказчик получает именно тот программный продукт, который ему требуется

для работы, а не разработанный по описанию требований, сформированных на начальном этапе разработки системы и, возможно, устаревших.

Таким образом, сформулированная в статье постановка задачи разработки принципов регулирования работ, основанных на использовании гибкого подхода к их организации, является актуальной и требует развития. Для решения сформулированной задачи и успешной реализации гибкого подхода потребуются корректировка НТД и РД, в первую очередь – в части разработки методик планирования НИОКР, а также описания принципов управления требованиями и непрерывности контроля качества разрабатываемого ПО, но результат обещает компенсировать затраты. Применение гибкой методологии позволит реализовать процессы динамичного управления требованиями к разрабатываемым компонентам ППО, а рациональная организация этого процесса сделать его приемлемым как для заказчика, так и для разработчика программной продукции.

Литература

1. Лукинова О.В. Методологические аспекты управления жизненным циклом информационной системы на основе инструментов функциональной стандартизации // Программные продукты и системы. 2016. Т. 29. № 4. С. 27–35. DOI: 10.15827/0236-235X.116.027-035.
2. Тиханычев О.В., Саяпин О.В., Гахов В.Р. Современные технологии информационного обследования как фактор успеха автоматизации управления // Информатизация и связь. 2013. № 6. С. 90–93.
3. Первухин Д.В., Исаев Е.А., Рытиков Г.О., Филюгина Е.К., Айрапетян Д.А. Сравнительный анализ теоретических моделей каскадных, итеративных и гибридных подходов к управлению жизненным циклом ИТ-проекта // Информационные системы и технологии в бизнесе. 2020. Т. 14. № 1. С. 32–40. DOI: 10.17323/2587-814X.2020.1.32.40.
4. Morien R. A retrospective on constructing a personal narrative on agile development. Proc. KICSS. AISC, 2018, vol. 685, pp. 290–304. DOI: 10.1007/978-3-319-70019-9_24.
5. Mahadevan L., Ketinger W.J., Meservy T.O. Running on hybrid: Control changes when introducing an agile methodology in a traditional “waterfall” system development environment. Communications of the Association for Information Systems, 2015, no. 36, pp. 77–103. DOI: 10.17705/1CAIS.03605.
6. Nicula D., Ghimiși S.S. Command and control vs self management. IOP Conf. Ser.: Materials Science and Engineering, 2019, vol. 514, art. 012039, pp. 1–6. DOI: 10.1088/1757-899X/514/1/012039.
7. Schuh G., Rebentisch E., Riesener M., Diels F., Dille C., Eich S. Agile-waterfall hybrid product development in the manufacturing industry – Introducing guidelines for implementation of parallel use of the two models. Proc. IEEE Int. Conf. IEEM, 2017, pp. 725–729. DOI: 10.1109/IEEM.2017.8289986.
8. Ramamoorthy B.T., Mayilvahanan P. Comparative study on agile scrum over traditional waterfall lifecycle projects. J. of Advanced Research in Dynamical and Control Systems, 2019, vol. 11, no. 4, pp. 524–529.
9. Сазерленд Д. Scrum. Революционный метод управления проектами; [пер. с англ.]. М.: Манн, Иванов и Фербер, 2016. 288 с.
10. Рубин К. Основы Scrum: Практическое руководство по гибкой разработке ПО; [пер. с англ.]. М.: Вильямс, 2016. 544 с.
11. Тиханычев О.В. О «гибких» технологиях в разработке программного обеспечения систем поддержки принятия решений // Программные системы и вычислительные методы. 2018. Т. 2. № 2. С. 51–59. DOI: 10.7256/2454-0714.2018.2.23743.

12. Добрынин А.С., Койнов Р.С., Пургина М.В. Принцип открытого управления в рейтинговых системах // Программные системы и вычислительные методы. 2016. Т. 3. № 3. С. 258–267. DOI: 10.7256/2305-6061.2016.3.19847.

13. Götz O., Wai Y., Klein S., Romehl M., Basten D. The (Go)SMART way to agility: Managing a Scrum subproject in a waterfall environment. *J. of Information Technology Teaching Cases*, 2018, vol. 8, no. 2, pp. 149–160. DOI: 10.1057/s41266-018-0035-9.

14. Симанков В.С., Толкачев Д.М. Информационное обеспечение ситуационного центра с использованием сети Интернет // Программные системы и вычислительные методы. 2015. № 3. С. 267–272. DOI: 10.7256/2305-6061.2015.3.16979.

15. Тиханычев О.В., Макарец Л.В., Гахов В.Р. Рациональная организация процесса разработки прикладного программного обеспечения как предпосылка успешной автоматизации поддержки принятия решений // Программные продукты и системы. 2017. Т. 30. № 4. С. 706–710. DOI: 10.15827/0236-235X.120.706-710.

Software & Systems
DOI: 10.15827/0236-235X.140.533-540

Received 14.07.22, Revised 03.09.22
2022, vol. 35, no. 4, pp. 533–540

Organizational problems of implementing flexible approaches in the applied software development

*O.V. Sayapin*¹, *Dr.Sc. (Engineering), Professor, tmaec@mail.ru*

*O.V. Tikhanychev*², *Ph.D. (Engineering), Deputy Head of the Department of Advanced Development, tow65@yandex.ru*

*A.A. Bezvesilnaya*¹, *Ph.D. (Education), Associate Professor, Head of the Department, a.bezvesilnaia@amchs.ru*

¹ *Civil Defence Academy EMERCOM of Russia, Khimki, 141435, Russian Federation*

² *TI Integration Group, Moscow, 111395, Russian Federation*

Abstract. The article analyzes and considers the features of cascade and flexible approaches to organizing applied software development for automated control systems, their positive aspects and disadvantages. The most critical shortcomings are the increase in development time and poor interaction between a customer and a developer when using the cascade approach. At the same time, the existing regulatory documents define this approach as the main one.

The use of general scientific methods of analysis and synthesis provides obtaining quantitative and qualitative estimates in terms of time and the expected result of using cascade and flexible approaches in software development. Based on the results of comparing the obtained estimates, the authors make a conclusion that a replacement of the cascade approach with a flexible one could be a rational solution. At the same time, the analysis of regulatory and technical documentation showed that the use of flexible approaches in developing application programs is hindered by existing organizational problems associated not only with the regulatory requirements, but also with the complexity of coordinating the work performed by distributed teams of performers. Based on the results of the analysis of a typical process of developing application software for automated control systems, the authors formulate proposals on possible options for replacing cascade approaches with flexible or combined ones.

The novelty of the proposed approach is in its complexity. Its implementation will allow building a development system that will increase the interest of all participants in the process as a result and implement this system through a process of continuous specification of requirements.

Keywords: control automation, software development, agile, decision support, organizational development problems.

References

1. Lukinova O.V. Methodological aspects of information system life cycle management based on functional standardization tools. *Software & Systems*, 2016, vol. 29, no. 4, pp. 27–35. DOI: 10.15827/0236-235X.116.027-035 (in Russ.).

2. Tikhanychev O.V., Sayapin O.V., Gakhov V.R. Modern information survey technologies as a factor in the success of management automation. *Informatization and Communication*, 2013, no. 6, pp. 90–93 (in Russ.).
3. Pervukhin D.V., Isaev E., Rytikov G., Filyugina E., Hayrapetyan D. Theoretical comparative analysis of cascading, iterative, and hybrid approaches to IT project life cycle management. *Business Informatics*, 2020, vol. 14, no. 1, pp. 32–40. DOI: 10.17323/2587-814X.2020.1.32.40 (in Russ.).
4. Morien R. A retrospective on constructing a personal narrative on agile development. *Proc. KICSS. AISC*, 2018, vol. 685, pp. 290–304. DOI: 10.1007/978-3-319-70019-9_24.
5. Mahadevan L., Kettinger W.J., Meservy T.O. Running on hybrid: Control changes when introducing an agile methodology in a traditional “waterfall” system development environment. *Communications of the Association for Information Systems*, 2015, no. 36, pp. 77–103. DOI: 10.17705/1CAIS.03605.
6. Nicula D., Ghimiși S.S. Command and control vs self management. *IOP Conf. Ser.: Materials Science and Engineering*, 2019, vol. 514, art. 012039, pp. 1–6. DOI: 10.1088/1757-899X/514/1/012039.
7. Schuh G., Rebentisch E., Riesener M., Diels F., Dlle C., Eich S. Agile-waterfall hybrid product development in the manufacturing industry – Introducing guidelines for implementation of parallel use of the two models. *Proc. IEEE Int. Conf. IEEM*, 2017, pp. 725–729. DOI: 10.1109/IEEM.2017.8289986.
8. Ramamoorthy B.T., Mayilvahanan P. Comparative study on agile scrum over traditional waterfall lifecycle projects. *J. of Advanced Research in Dynamical and Control Systems*, 2019, vol. 11, no. 4, pp. 524–529.
9. Sutherland J. *Scrum. The art of Doing Twice the Work in Half the Time*. NY, Crown Business Publ., 2014, 248 p. (Russ. ed.: Moscow, 2016, 288 p.).
10. Rubin K.S. *Essential Scrum: A Practical Guide to the Most Popular Agile Process*. NY, Addison-Wesley Professional Publ., 2012, 496 p. (Russ. ed.: Moscow, 2016, 544 p.).
11. Tikhanychev O.V. Agile technologies in software development of decision support systems. *Software Systems and Computational Methods*, 2018, vol. 2, no. 2, pp. 51–59. DOI: 10.7256/2454-0714.2018.2.23743 (in Russ.).
12. Dobrynin A.S., Koynov R.S., Purgina M.V. The principle of open management in rating systems. *Software Systems and Computational Methods*, 2016, vol. 3, no. 3, pp. 258–267. DOI: 10.7256/2305-6061.2016.3.19847 (in Russ.).
13. Götz O., Wai Y., Klein S., Romehl M., Basten D. The (Go)SMART way to agility: Managing a Scrum subproject in a waterfall environment. *J. of Information Technology Teaching Cases*, 2018, vol. 8, no. 2, pp. 149–160. DOI: 10.1057/s41266-018-0035-9.
14. Simankov V.S., Tolkachev D.M. Information support of situational center with the use of internet. *Software Systems and Computational Methods*, 2015, no. 3, pp. 267–272. DOI: 10.7256/2305-6061.2015.3.16979 (in Russ.).
15. Tikhanychev O.V., Makartsev L.V., Gakhov V.R. Rational organization of an application software development process for decision support successful automation. *Software & Systems*, 2017, vol. 30, no. 4, pp. 706–710 (in Russ.).

Для цитирования

Саяпин О.В., Тиханьчев О.В., Безвесильная А.А. Организационные проблемы реализации гибких подходов в разработке прикладного программного обеспечения // Программные продукты и системы. 2022. Т. 35. № 4. С. 533–540. DOI: 10.15827/0236-235X.140.533-540.

For citation

Sayapin O.V., Tikhanychev O.V., Bezvesilnaya A.A. Organizational problems of implementing flexible approaches in the applied software development. *Software & Systems*, 2022, vol. 35, no. 4, pp. 533–540 (in Russ.). DOI: 10.15827/0236-235X.140.533-540.