УДК 004.9

ЭМЕРДЖЕНТНЫЕ МОДЕЛИ В СОСТАВЕ ТЕХНИЧЕСКИХ ПРЕДЛОЖЕНИЙ К ВЫПОЛНЕНИЮ ИНИЦИАТИВНОЙ НИР ПО КАФЕДРЕ ИИППО «МЕТОДОЛОГИЧЕСКИЕ ОСНОВЫ ПРОЕКТИРОВАНИЯ АРХИТЕКТУРЫ ИНФОРМАЦИОННЫХ СИСТЕМ»  
  
Болбаков Р.Г., Плотников С.Б.,   
Куликов А.А., Тигай М.

*МИРЭА – Российский технологический университет, 119454, Россия, г. Москва, проспект Вернадского, 78, tibult41@gmail.com*

Материалы статьи отображают вариативные подходы авторов в работе над выбором методик разработки информационных систем (ИС) в составе открываемой на период 2020 – 2022 гг. инициативной комплексной НИР кафедры ИиППО Института ИТ РТУ МИРЭА на тему: «Методологические основы проектирования архитектуры информационных систем (ИС)». Рассматриваются методологические аспекты оценки эмерджентности проектирования информационных систем.

Ключевые слова: эмерджентность, инжиниринг, методология, реинжиниринг, информационные системы.

EMERGENT MODELS AS PART OF TECHNICAL PROPOSALS FOR THE IMPLEMENTATION OF THE INITIATIVE SRW OF THE DEPARTMENT AND THE "METHODOLOGICAL BASIS FOR DESIGNING THE ARCHITECTURE OF INFORMATION SYSTEMS".  
  
Bolbakov R.G., Plotnikov S.B.,

Kulikov A.A., Tigai M.

*MIREA - Russian Technological University, 119454, Russia, Moscow, 78 Vernadsky Avenue, tibult41@gmail.com*

The materials of the article reflect the authors' variant approaches in the work on the choice of methods for developing information systems (IS) as part of the initiative complex research, which is being opened for the period 2020 - 2022, by the Department IIPPO of the Institute of IT RTU MIREA on the theme: "Methodological bases of designing the architecture of information systems (IS)". Methodological aspects of evaluation of the emergenence of information systems design are considered.

Keywords: Emergencies, engineering, methodology, re-engineering, information systems.

Успешность инжиниринга информационной системы (ИС) определяется проработанностью методики подхода, используемого в процессах моделирования, конструирования и проектирования. В этом процессе участвуют: пользователи, аналитики и разработчики. Современная тенденция разработки ИС характеризуется следующими признаками [1]:

– Автоматизация всех процессов предприятия: бизнес-процессов, процессов согласования и принятия решений.

– Построение единой модели ИС, понятное всем участникам процесса разработки.

– Комплексное решение вопросов формализации и информатизации процессов принятия решений.

– Возможность осуществления и реализация имитационного моделирования автоматизируемых процессов пользователями с возможностью использовать полученные результаты и знания на этапах автоматизации [3].

Рассмотрим следующие методологии инжиниринга ИС:

1. **Waterfall**(**WF**)(каскадная, водопадная) модель - одна из самых старых и проверенных практикой методологий, подразумевающая строгое последовательное выполнение всех этапов продвижения проекта, каждый из которых должен завершиться перед началом следующего. То есть переход на следующий этап означает полное завершение работ на предыдущем. На рисунке 1 показано 5 этапов модели:

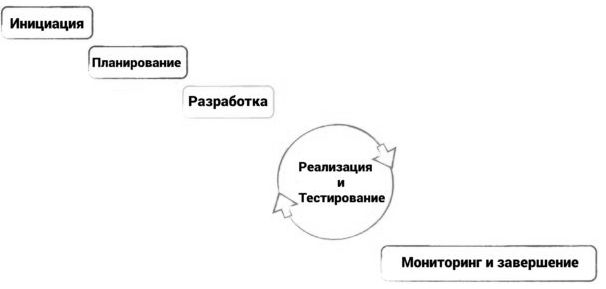


Рис.1. Схема процесса Waterfall

**Этап 1. Инициация.** На данном этапе определяются требования к информационной системе, ставится цель ИС, часто проводятся совещания на которых определяется представление ИС.

**Этап 2. Планирование.** На данном этапе формируется структура ИС, формируются этапы достижения цели, поставленной на предыдущем этапе. Происходит уточнение, детализация цели и результаты ИС, а также состав работ по ней. На основании данной информации формируется календарный план и бюджет, оцениваются риски и выявляются заинтересованные стороны.

**Этап 3. Разработка.** Данная стадия реализуется не для всех ИС — как правило она является частью фазы планирования. В фазе разработки, характерной для технологических ИС, определяется конфигурация будущего проекта и/или продукта и технические способы его достижения. Например, в ИТ-проектах на данном этапе выбирается язык программирования.

**Этап 4. Реализация и тестирование.** На этой фазе происходит собственно основная работа по ИС – написание кода, возведение здания и тому подобное. Следуя разработанным планам начинает создаваться содержание проекта, определённое ранее, проводится контроль по выбранным метрикам. Во второй части данной фазы происходит тестирование ИС, она проверяется на соответствие требованиям Заказчика и заинтересованных сторон. В части тестирования выявляются и исправляются недостатки продукта.

**Этап 5. Мониторинг и завершение проекта.** В зависимости от ИС данная фаза может состоять из простой передачи Заказчику результатов ИС или же из длительного процесса взаимодействия с клиентами по улучшению ИС и повышению их удовлетворённости, и поддержке результатов ИС.

Возвраты на следующие этапы не предусмотрены. Использовать такую систему рекомендуется в небольших проектах, где известны заранее требования и мала вероятность, что они будут менять.

1. **Scrum** (итеративно-инкрементальный метод) - методика разработки ИС, основанная на делении всего процесса на итерации, где в конце каждой из них есть возможность предоставить первую версию ИС. На рисунке 2 показаны этапы параллельной разработки, что позволяет в конце каждой итерации иметь готовую часть информационной. Спринт – итерации в Scrum, которые обычно длятся от 2 до 4 недель [4].

Основная структура процессов Scrum вращается вокруг пяти основных встреч: упорядочивания беклога, планирования Спринта, ежедневных встреч, подведения итогов Спринта и ретроспективы Спринта.

1. Встреча по упорядочиванию беклога (Backlog Refinement Meeting, «Backlog Grooming»): Эта встреча аналогична фазе планирования в классическом проектном управлении, и проводится в первый день каждого Спринта. На ней рассматривается – что уже было сделано по проекту в целом, что ещё осталось сделать и принимается решение о том, что же делать дальше. Определяются, какие задачи на данном этапе являются наиболее приоритетными. Данный процесс определяет эффективность Спринта, ведь именно от него зависит, какую ценность получит заказчик по итогам спринта.

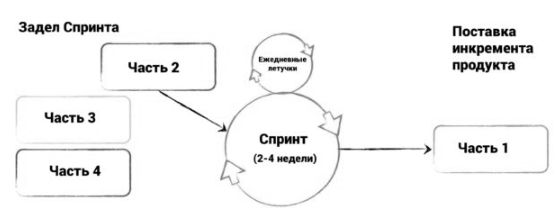


Рис2. Схема процесса Scrum

* Планирование Спринта: После того, как определены приоритеты, составляется план задач в грядущей, итерации, как достигнуть поставленной на предыдущей встрече цели. Возможно применение различных инструментов планирования и оценки на данном этапе, лишь бы они не противоречили принципам и логике Scrum.
* Ежедневные встречи: Каждый день спринта, в идеале, в одно и то же время, члены команды разработки ИС тратят 15 минут на то, чтобы поделиться информацией о статусе задач и состоянии проекта. На ней не происходит обсуждений проблем или принятия решений – если после встречи возникают вопросы и конфликты, Scrum Мастер и вовлечённые участники обсуждают их отдельно.
* Подведение итогов Спринта: Цель этапа – обследование и адаптация создаваемого продукта. Команда представляет результаты деятельности всем заинтересованным лицам. Основная задача – убедиться, что продукт этапа соответствует ожиданиям участников и согласуется с целями проекта.
* Ретроспектива Спринта: Проводится сразу после Подведения итогов спринта и до планирования следующего спринта. На нём команда выясняет, насколько чётко и слаженно проходил процесс реализации этапа. Обследованию подвергаются возникшие проблемы в работе, методологии и взаимодействии. Именно этот этап позволяет команде провести рефлексию и следующий Спринт провести эффективнее.

1. **Kanban -** система, построенная на визуализации процесса выполнения задач команды. Основная идея в этой системе уменьшать количество задач выполняющихся в данный момент (в колонке «in progress»). В scrum ориентация команды на успешное выполнение спринтов, в Канбане на первом месте задачи. В канбане задачи сдаются индивидуально. Задача независимо от других задач проходит по всем этапам на доске и как только она выполнена её можно показать [5,6].

Канбан доска состоит из колонок, каждая из которых это отдельный процесс разработки. На некоторые столбцы (например, in progress) вводят ограничения по количеству тасок, которые там могут находиться. Это помогает легко и быстро находить проблемные места в распределении задач. На картинке пример самой просто такой доски. Количество колонок и названия могут меняться, самые распространенные:

* To do – список задач, которые надо сделать
* In progress – задачи над которыми ведется работа в данный момент
* Code review – задачи, которые сделаны и отправлены на ревью
* In testing – задачи, готовые к тестированию
* Done – сделанные задачи.

Обобщая выше приведённое в качестве заключения к настоящему эссе уместно обозначить и выделить следующий набор отборочных критериев: модель процессов предприятия(A), которая описывает статические (B) и динамические (C) бизнес-процессы, а также модели лиц, принимающих решения (D);средства анализа процессов (E), включающие организационный реинжи-ниринг (F), анализ «узких мест» (G);возможность использования данных из модели предприятия при разработке ИС (H): в части динамических бизнес-процессов (I), в части модели лица, принимающего решение (J);использование структурного (K) и объектно-ориентированного подходов (L);результаты автоматизации (M): бизнес-процессы (N), согласование решений (O), процессы принятия решений –использование машины логического вывода(P).

Таблица 1. Сравнение методик разработки ИС

|  |  |  |  |
| --- | --- | --- | --- |
| Критерии сравнения | WF | Scrum | Kanban |
| A | да | да | да |
| B | да | да | да |
| C | нет | да | да |
| D | нет | да | нет |
| E | нет | да | да |
| F | нет | да | нет |
| G | нет | да | нет |
| H | нет | да | да |
| I | нет | да | да |
| J | нет | нет | нет |
| K | да | да | да |
| L | да | да | да |
| M | да | да | да |
| N | да | да | да |
| O | нет | да | да |
| P | да | да | да |

Из описанных методик разработок ИС, затрагивающие анализ процессов ОТС, динамику БП, анализ «узких» мест, использование информации из модели процессов ОТС в части динамических БП для разработки информационной системы, по мнению автора, более всего подходит методика разработки Scrum. Кроме того, обозначенный метод уделяет позиционирует признаки и показатели надежности, временной характеристики разработки информационных систем, обладающих самой различной архитектоникой.

Список литературы

1. Аксенов К.А., Спицина И.А., Крохин А.Л. Сравнительный анализ методов разработки информационных систем на основе мультиагентного подхода (205-207) с.
2. Швецов А.Н. Модели и методы построения корпоративных интеллектуальных систем поддержки принятия решений: дис. д-ра техн. наук: 05.13.01 / А.Н. Швецов. Санкт-Петербург, 2004. 461 с.
3. Royce, Winston (1970), Managing the Development of Large Software Systems
4. Майк Кон. Scrum: гибкая разработка ПО = Succeeding with Agile: Software Development Using Scrum. — М.: «Вильямс», 2011. — С. 576.
5. James P. Womack. The Machine That Changed the World. — 2007.
6. Мэри Поппендик, Toм Поппендик. Бережливое производство программного обеспечения: от идеи до прибыли / Вильямс, 2009 г

References

* + - 1. Aksenov K.A., Spitsina I.A., Krokhin A.L. Comparative analysis of methods for development of information systems based on the multi-agent approach (205-207) p.
      2. Shvetsov A.N. Models and methods of construction of corporate intellectual systems of decision-making support: Dr. Sci. St. Petersburg, 2004. 461 с.
      3. Royce, Winston (1970), Managing the Development of Large Software Systems
      4. Mike Cohn. Scrum: Flexible software development = Succeeding with Agile: Software Development Using Scrum. - M.: Williams, 2011. - — С. 576.
      5. James P. Womack. The Machine That Changed the World. - — 2007.

1. Mary Poppendick, Tom Poppendick. Lean Production Software: From Idea to Profit / Williams, 2009.