

УДК 629.7  
ББК 65.290-86  
JEL C53

**Анализ подходов применения программного обеспечения для проектирования изделия под заданную стоимость на этапах жизненного цикла**

**Масленников Евгений Сергеевич**, магистрант кафедры 101 «Проектирование и сертификация авиационной техники», Московский авиационный институт (Национальный исследовательский университет), Волоколамское шоссе, 4, Москва, Российская Федерация, 125080  
E-mail: [maslennikov.e.s@gmail.com](mailto:maslennikov.e.s@gmail.com)

**Гязова Марина Мухарбиевна**, кандидат экономических наук, доцент кафедры 505 «Инновационная экономика, финансы и управление проектами», Московский авиационный институт (национальный исследовательский университет), Волоколамское шоссе, 4, Москва, Российская Федерация, 125080  
ORCID: [0000-0002-0183-2082](https://orcid.org/0000-0002-0183-2082)  
E-mail: [mgyazova@gmail.com](mailto:mgyazova@gmail.com)

**Аннотация:** В статье представлены возможные подходы применения программного обеспечения для анализа проектирования изделия под заданную стоимость на этапах жизненного цикла и ее основные функции, а также модели проектирования программного обеспечения с учетом процесса разработки программного продукта. Показаны решения для сбора, ввода и анализа информации.

**Ключевые слова:** программное обеспечение; анализ затрат; модели проектирования; экономика; жизненный цикл продукции; научно-технический прогресс.

**Analysis of approaches to the use of software for designing a product for a given cost at the stages of the life cycle**

**Evgeniy S. Maslennikov**, Master's Degree student of Department 101 «Aircraft Design and Certification», Moscow Aviation Institute (National Research University), Volokolamskoeshosse, 4, Moscow, Russian Federation, 125080  
E-mail: [maslennikov.e.s@gmail.com](mailto:maslennikov.e.s@gmail.com)

**Marina M. Gyazova**, PhD in Economic sciences, Associate Professor of Department 505 "Innovative Economics, Finance and Project Management", Moscow Aviation Institute (National Research University), Volokolamskoe shosse, 4, Moscow, Russian Federation, 125080  
ORCID: [0000-0002-0183-2082](https://orcid.org/0000-0002-0183-2082)  
E-mail: [mgyazova@gmail.com](mailto:mgyazova@gmail.com)

**Abstract:** The article presents possible approaches to using software for analyzing the design of a product at a given cost at the stages of the life cycle and its main functions, as well as software design models taking into account the process of developing a software product. Solutions for collecting, entering and analyzing information are shown.

**Keywords:** software; cost analysis; modeling; economics; product life cycle; scientific and technological progress.

**Введение**

За последние тридцать лет в экономике наметилась четкая тенденция к значительному сокращению жизненного цикла (ЖЦ) продукции в результате ускорения научно-технического прогресса, в связи с чем возможность управлять жизненным циклом продукта и осуществлять его разработку в более короткие сроки становится важным конкурентным преимуществом.

Возникает необходимость в принятии решений в быстро меняющихся условиях, что требует применение новых методов, подходов, одним из которых является анализ затрат жизненного цикла (Life cycle cost analysis – LCCA). LCCA это инструмент для определения наиболее экономичного варианта среди различных конкурирующих альтернатив покупки, владения, эксплуатации, обслуживания и, наконец, утилизации объекта или процесса [1].

Целью данного исследования является анализ функций, механизмов применения программного обеспечения (ПО) для оценки стоимости жизненного цикла изделия.

### Материалы и методы

Постановка задачи по разработке программного модуля для оценки стоимости жизненного цикла требует рассмотрение следующих вопросов:

1) Рассмотреть существующие механизмы применения программного обеспечения для оценки стоимости жизненного цикла изделия в разных отраслях.

2) Дать описание процессам разработки программного обеспечения для оценки стоимости жизненного цикла изделия, с учетом выбора облачного сервис.

3) Выбор программного метода.

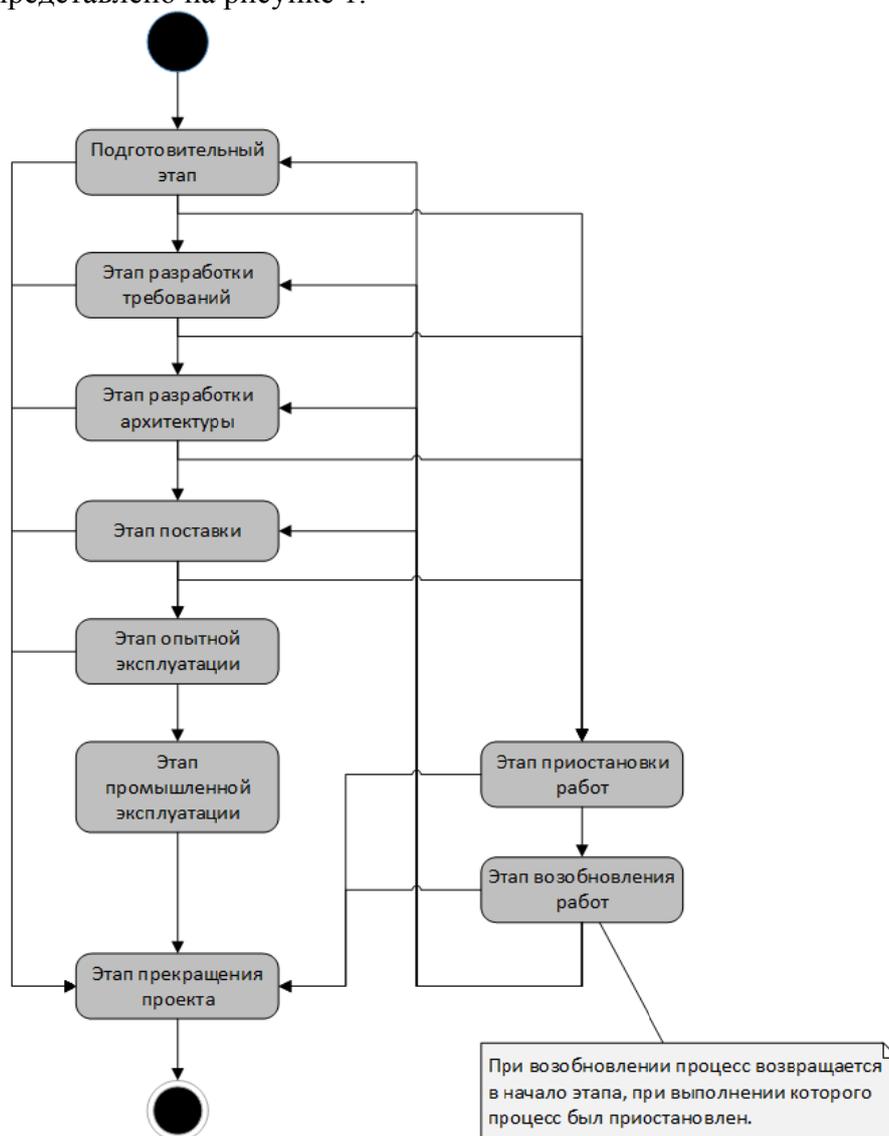
В настоящее время в мировой практике применяются следующие методы проектирования программного обеспечения, сравнительный анализ которых представлен в таблице 1.

**Таблица 1 Сравнение методов проектирования ПО**

<b>Методы проектирования – программного обеспечения</b>	<b>Условия использования методики</b>
Модель водопада	<ul style="list-style-type: none"> <li>• только когда требования известны, поняты и фиксированы. Нет противоречивых требований</li> <li>• нет проблем с наличием программистов необходимой квалификации</li> <li>• в относительно небольших проектах</li> </ul>
V-модель	<ul style="list-style-type: none"> <li>• если требуется тщательное тестирование продукта, то V-модель оправдывает основную идею: валидация и верификация</li> <li>• для малых и средних проектов, где требования четко определены и зафиксированы</li> <li>• в условиях наличия инженеров с необходимой квалификацией, особенно тестировщиков</li> </ul>
Инкрементальная модель	<ul style="list-style-type: none"> <li>• когда основные требования к системе четко определены и поняты. В то же время некоторые детали могут быть улучшены со временем.</li> <li>• необходим ранний вывод продукта на рынок</li> <li>• есть несколько рискованных функций или целей</li> </ul>
Модель RAD	только в присутствии высококвалифицированных и узкоспециализированных архитекторов. Бюджет проекта большой, чтобы оплатить этих специалистов, равно как и стоимость готовых автоматизированных сборочных инструментов. Модель RAD можно выбрать при уверенном знании целевого бизнеса и необходимости срочного производства системы в течение 2-3 месяцев.
Гибкая модель (Agile)	<ul style="list-style-type: none"> <li>• когда потребности пользователей постоянно меняются в динамичном бизнесе</li> <li>• гибкие изменения реализуются с меньшими затратами из-за частых приращений</li> <li>• в отличие от модели водопада, в гибкой модели достаточно лишь небольшого планирования, чтобы начать проект</li> </ul>
Итерационная модель	<ul style="list-style-type: none"> <li>• повторные требования к окончательной системе четко определены и понятны заранее</li> </ul>

	<ul style="list-style-type: none"> <li>• проект большой или очень большой</li> <li>• основная цель должна быть определена, но детали реализации могут изменяться со временем</li> </ul>
Спиральная модель	<ul style="list-style-type: none"> <li>• эта модель не подходит для небольших проектов, она целесообразна для сложных и дорогостоящих, например, таких как разработка системы документооборота для банка, когда каждый следующий шаг требует больше анализа для оценки последствий, чем программирования</li> </ul>

Описание этапов разработки программного обеспечения для расчета стоимости жизненного цикла изделия представлено на рисунке 1.



**Рис. 1. Этапы разработки программного обеспечения для расчета стоимости жизненного цикла изделия [3]**

Работа над проектом начинается с подготовительного этапа. Цель этапа - создать на основе предложений определенную концепцию будущей системы и на основе этой концепции оценить актуальность и целесообразность проекта. Если решение о привлечении подрядчика принимает заказчик, то предварительный этап включает формирование необходимой документации.

Проект, концепция которого выглядит приемлемой для реализации, входит в стадию разработки требований. Цели этапа - выявить все скрытые потребности, разрешить конфликты требований, сформировать целостное техническое решение и проанализировать осуществимость подготовленного решения [4].

Если техническое решение найдено, приступаем к разработке архитектуры будущей системы. Цель этого этапа – определить логическую и физическую архитектуру высокого уровня, которая полностью покрывает все установленные требования. При разработке архитектуры проводится обзор и уточнение концепции, требований и предварительного технического решения, что позволяет предотвратить наиболее опасные риски.

Если после завершения архитектурного проектирования баланс был найден, и удалось создать приемлемую архитектуру системы, можно переходить к внедрению системы, реализация которой может проходить в один или несколько этапов. Для небольших проектов одностадийная поставка всей функциональности системы может быть вполне приемлемой. Однако чем крупнее проект, тем выше зависимость подсистем внутри создаваемой системы. В этих условиях внедрение следует разделить на несколько этапов, чтобы в конце каждого этапа у команды разработчиков был продукт, готовый к поставке. Затем полностью отлаженная и настроенная система вводится в промышленную эксплуатацию.

Завершающая фаза проекта направлена на анализ результатов, внесение изменений в процесс разработки на основе полученного опыта и пополнение базы знаний разработчика новыми эффективными решениями и предупреждениями, а также новыми готовыми компонентами, которые можно использовать в будущих проектах.

Системы электронных облаков бывают нескольких типов, где поставщики предлагают услуги общедоступного облака, а предприятия размещают в своих центрах обработки данных локальные частные облака. Есть и промежуточный вариант - гибрид, когда одна часть задач решается в публичном облаке, а другая – в частном.

Что касается приложений жизненного цикла изделия (Productlifecycle – PLM), частные облака больше подходят для систем автоматизированного проектирования (САПР), поскольку они имеют более широкие каналы передачи данных, или гибридные, в которых рабочие станции проектировщиков будут расположены локально, а другие рабочие станции и хранилища данных будут находиться в публичном облаке. Но для этого сначала нужно решить вопрос с производительностью обработки графики на виртуальных рабочих станциях [5].

Для инженерных расчетов наиболее эффективными будут общедоступные облака, которые могут обеспечить высокую вычислительную мощность, избавляя предприятие от необходимости покупать дорогостоящее оборудование и программное обеспечение. Но отметим, что частное облако имеет важное преимущество: его всегда можно настроить и отрегулировать для повышения производительности.

Чтобы разработать продукт в кратчайшие сроки без снижения качества и увеличения затрат, нужно понимать, каким будет его жизненный цикл. Точно сформулированные требования позволят выбрать в программе методы для наиболее оптимального решения поставленных задач. Наиболее важными функциями программного обеспечения для расчета стоимости жизненного цикла являются:

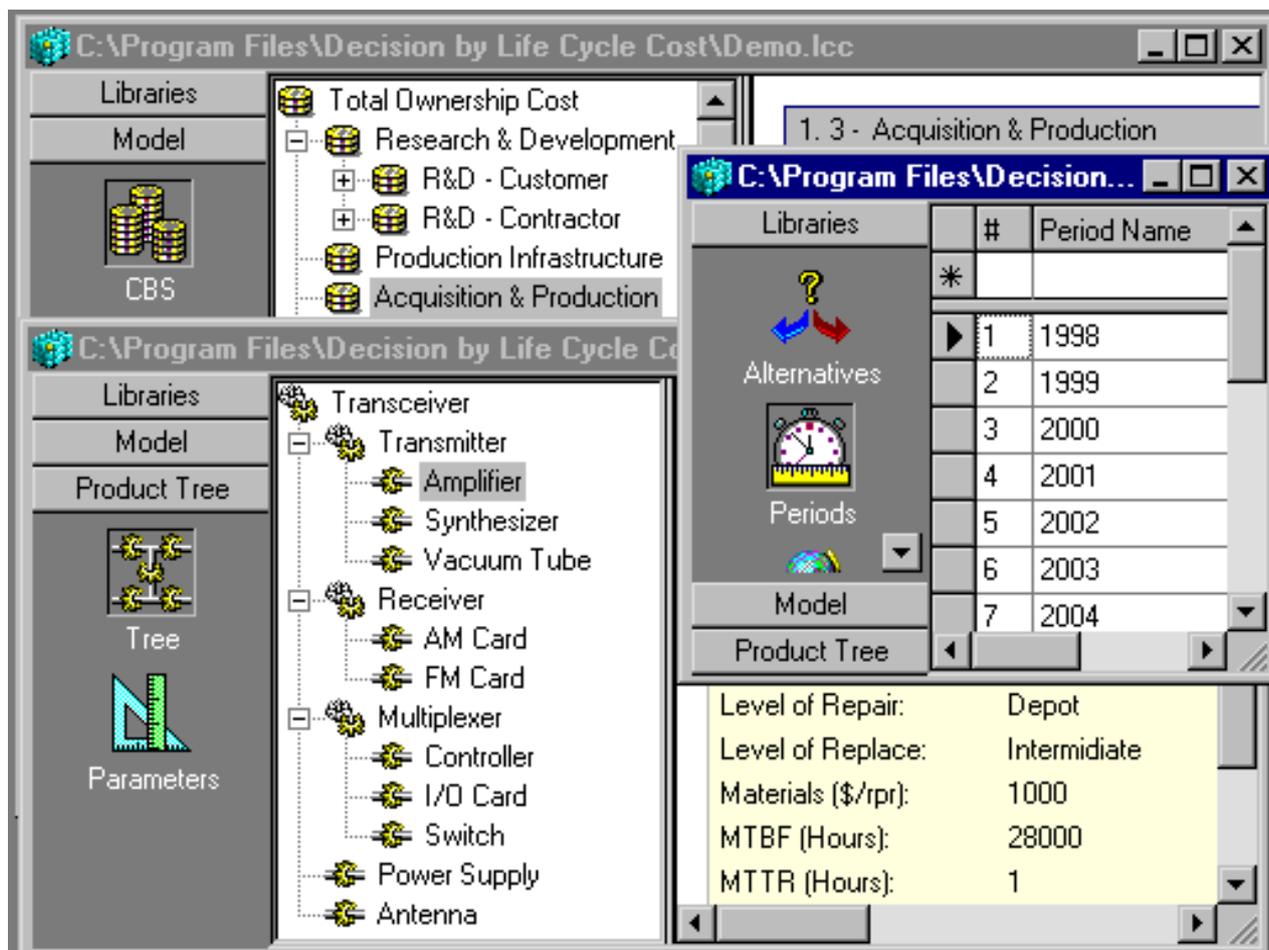
- расчет стоимости жизненного цикла – анализ общей стоимости владения различными конструктивными альтернативами и компонентами системы на протяжении всего прогнозируемого жизненного цикла системы;
- анализ затрат – расчет стоимости дерева продуктов, которая позволяет встроить параметры дерева продуктов в модель жизненного цикла и рассчитать любые требуемые категории / элементы затрат (например, стоимость запасных частей для каждого уровня ремонта) для всех компонентов дерева продуктов;
- общая стоимость владения - рассчитывается с использованием существующей или вновь созданной структуры распределения затрат (Cost breakdown structure – CBS) и присвоения переменных затрат каждому первичному элементу CBS. Система обеспечивает оценку затрат в порядке возрастания, поддерживает подробное изучение затрат и параметров, влияющих на жизненный цикл, выполняет анализ чистых текущих затрат;
- чистая приведенная стоимость (Net Present Cost – NPC). При анализе бюджета и финансов необходимым требованием является определение текущей стоимости будущих денежных потоков, а также сравнение вариантов с разными уровнями инфляции и ставками

дисконтирования, который сопровождается анализом чувствительности ставки инфляции и ставки дисконтирования;

- оценка прибыльности, которая обычно рассчитывается на основе операционных затрат на единицу затрат;
- анализ удельной стоимости – инструмент для расчета затрат по конкретной статье бюджета. Функция анализа удельных затрат рассчитывает все компоненты, например, стоимость рабочей силы или материала. Модуль генерирует результаты для элементов, которые затем добавляются ко всему проекту;
- анализ чувствительности – рассчитывает изменения в жизненном цикле / общей стоимости владения на основе изменений определенных глобальных переменных. Анализ чувствительности определяет основные детерминанты стоимости (принцип Парето 20/80), поддерживает сравнительный анализ и показывает результаты изменений конкретного критического параметра и ключевых допущений;
- библиотека данных – набор глобальных переменных, которые приемлемы для модели стоимости и влияют на большое количество элементов, например, инфляцию и ставки дисконтирования [6].

Возможность ввода собственных методов расчета позволит оценить жизненный цикл продукта, если он имеет некоторые особенности, не входящие в общий список механизмов.

Примеры функций программного обеспечения представлены ниже:



**Рис. 2. Структуру распределения затрат (CBS) и структурную декомпозицию изделия (PBS), предоставляет расчет по восходящей с учетом времени (фаз жизненного цикла) на примере D-LCC**

### Cost Profile

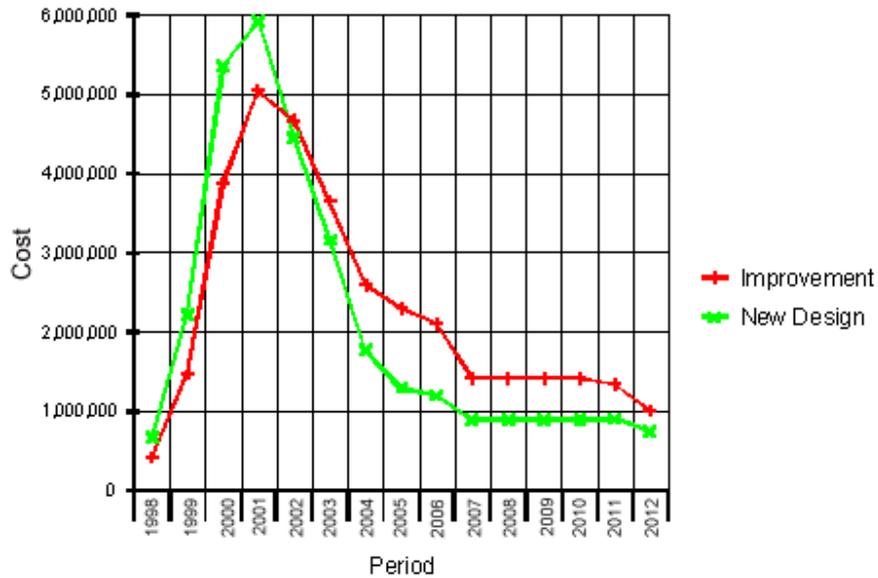


Рис. 3. Оценка кривой затрат

### Sensitivity Analysis

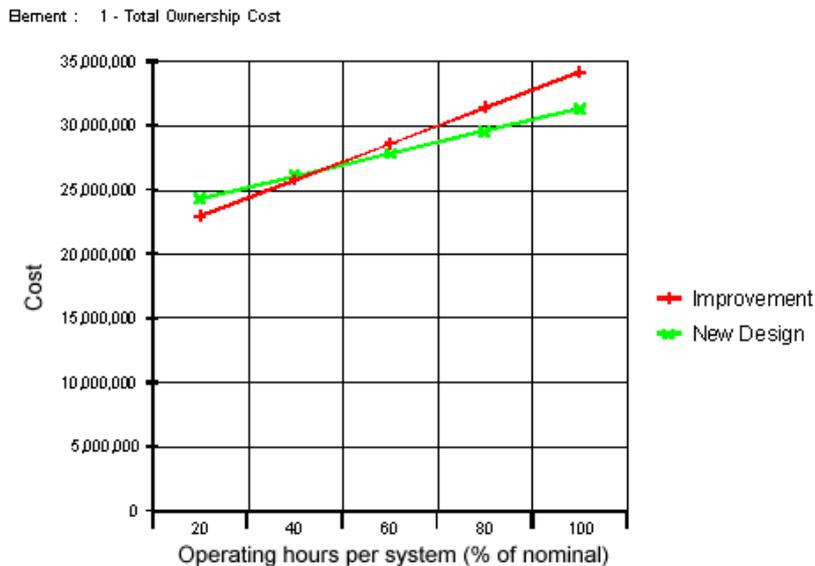


Рис. 4. Анализ чувствительности

### Результаты

Рассмотренные методы проектирования программного обеспечения, описывают процесс разработки программного обеспечения, типы облачных систем хранения, важные функции для оценки жизненного цикла программного обеспечения, а также бизнес-процессы в существующем программном обеспечении. Из рассмотренных методов проектирования ПО для оценки стоимости жизненного цикла, чаще остальных используется гибкий метод (Agile), расположение ПО на общественных облаках позволяет осуществлять работу удаленно и с наименьшими затратами на развертывание дополнительных серверов. Рассмотренные функции позволяют осуществить анализ и оценку стоимости жизненного цикла изделия, а так же провести разработку изделия под заданную стоимость.

## **Заключение**

В данной статье рассматриваются только основные практические подходы к созданию программного обеспечения для расчета стоимости жизненного цикла. Предложенные методики по совмещению функций могут быть использованы при проведении технико-экономического обоснования создания инновационной продукции на предприятиях авиационно-строительной отрасли, а также могут служить методологической основой для проведения технико-экономического обоснования инновационной продукции и процессов и применимы для организаций-производителей авиационной техники.

## **Библиография**

1. Багиев Г.Л. Факторы повышения конкурентоспособности промышленных корпораций: Учебное пособие. разрешение. – М.: ГАЗ и ЖКК, 2017. – 141 с. ISBN 978-5-7996-1098-2.
2. Берг Д.Б., Ульянова Е.А., Добряк П.В. Модели жизненного цикла: учебник. – Екатеринбург: Уральское изд-во. Университет, 2014. – 74 [2] с. ISBN 978-5-7996-1311-2.
3. Фридлянд А.А., Гязова М.М. (2019) Модель для оценки влияния результатов НИОКР на техническое и эксплуатационное совершенство авиационного двигателя и стоимость жизненного цикла самолета // Российский экономический интернет-журнал. – №1. eISSN: 2218-5402.
4. Кузнецов М.И. Обзор процесса разработки программного обеспечения, 2015 г. URL: <https://habr.com/ru/post/255991/>.
5. Погосян М.А. и др. (2018) Самолетостроение: учебник. М.: Инновационная инженерия. 864 с. ISBN 978-5-6040281-5-5.
6. Зиглинде Ф. Национальный институт стандартов и технологий (NIST), 2016 г. URL: <https://wbdg.org/resources/life-cycle-cost-analysis-lcca>.

## **References**

1. Bagiev G.L. Factors of increasing the competitiveness of industrial corporations: textbook. allowance. M.: GAS and ZhKK, 2017. 141 p. ISBN 978-5-7996-1098-2.
2. Berg D.B., Ulyanova E.A., Dobryak P.V. Life cycle models: textbook. allowance /Yekateriburg: Ural Publishing House. University, 2014. 74 [2] p. ISBN 978-5-7996-1311-2.
3. Fridlyand A.A., Gyazova M.M. (2019) A model for assessing the impact of R&D results on the technical and operational excellence of an aircraft engine and the cost of the life cycle of an aircraft // Russian Economic Internet Journal. №1. eISSN: 2218-5402.
4. Kuznetsov M.I. Overview of the software development process, 2015. URL: <https://habr.com/ru/post/255991/>.
5. Pogosyan M.A. et al. (2018) Aircraft design: a textbook. M.: InnovativeEngineering. 864 p. ISBN 978-5-6040281-5-5.
6. Sieglinde F. National Institute of Standards and Technology (NIST), 2016. URL: <https://wbdg.org/resources/life-cycle-cost-analysis-lcca>.