

**Фреймворк методов
интеллектуальной эвристической оптимизации iOpt**

**Описание процессов, обеспечивающих
поддержание жизненного цикла**

Нижний Новгород

2023 г.

Аннотация

Данный документ содержит описание процессов, обеспечивающих поддержание жизненного цикла Фреймворка методов интеллектуальной эвристической оптимизации iOpt, в том числе устранение неисправностей, выявленных в ходе эксплуатации ПО, совершенствование ПО, а также информацию о персонале, необходимом для обеспечения такой поддержки.

Оглавление

Перечень принятых сокращений	4
1. Общие сведения	5
1.1. Виды деятельности, функции	5
2. Документация iOpt	7
3. Комплект Программного обеспечения	8
4. Процессы жизненного цикла iOpt	9
4.1. Развитие iOpt	10
4.2. Процессы, обеспечивающие поддержание жизненного цикла iOpt	10
4.2.1. Процесс менеджмента документации	10
4.2.2. Процесс менеджмента конфигурации программного обеспечения	11
4.2.3. Процесс обеспечения гарантии качества программного обеспечения	12
4.2.4. Процесс верификации программного обеспечения	12
4.2.5. Процесс валидации программного обеспечения	13
4.2.6. Процесс ревизии программного обеспечения	14
4.2.7. Процесс аудита программного обеспечения	16
4.2.8. Процесс решения проблем в программном обеспечении	16
4.3. Эксплуатация и сопровождение системы	17
4.3.1. Гарантийное обслуживание	18
4.3.2. Техническая поддержка	19
4.4. Модернизация	20
4.5. Описание процесса работы	20
4.5.1. Каналы доставки запросов	21

Перечень принятых сокращений

ИС – Информационная система

ПО – программное обеспечение

СУБД – система управления базами данных

КСП – Кризисная ситуация и происшествие

ТП – Техническая поддержка

ОПО – Общесистемное программное обеспечение

1. Общие сведения

Фреймворк методов интеллектуальной эвристической оптимизации iOpt предназначен для выбора оптимальных (в заданной метрике) значений параметров сложных объектов и процессов, например, методов искусственного интеллекта и машинного обучения, а также – методов эвристической оптимизации. Фреймворк позволяет проводить точную настройку параметров моделей и методов, используемых в прикладных исследованиях в различных научных областях.

1.1. Виды деятельности, функции

Фреймворк обеспечивает реализацию следующих функциональных возможностей:

- автоматический выбор значений параметров математических моделей и методов ИИ и МО, используемых в промышленности;
- интеллектуальное управление процессом выбора оптимальных параметров для промышленных задач;
- интеграция с внешними библиотеками или фреймворками искусственного интеллекта и машинного обучения, а также предметными моделями;
- автоматизация предварительного анализа исследуемых моделей, например, выделение различных классов зависимостей модели от разных групп параметров;
- визуализация процесса выбора оптимальных параметров;
- фреймворк разработан на языке Python 3.8 с использованием библиотек numpy, scikit-learn, depq, kiwisolver, matplotlib.

Ограничения:

- фреймворк ориентирован на решение задач с числом параметров не более десяти;

- параметры могут быть как непрерывными, так и категориальными (число различных значений каждого категориального параметра должно ограничиваться единицами вариантов);
- фреймворк способен решать задачи с функциональными ограничениями на параметры (не более десяти ограничений);
- время проведения одного поискового испытания (т.е. работы настраиваемого алгоритма с одним набором параметров) должно составлять величину порядка минут, а не часов (иначе настройка займет неприемлемое время).

2. Документация iOpt

Пользовательская и эксплуатационная документация включает следующие документы:

- Руководство пользователя;
- Руководство по установке.

Комплект Программного обеспечения Для установки фреймворка iOpt может использоваться система управления пакетами PIP (https://pip.pypa.io/en/stable/cli/pip_install/).

Пример команды при настроенном окружении и установленной системе управления пакетами:

```
pip install iOpt
```

В случае, если требуется работа напрямую с исходным кодом проекта, могут использоваться варианты, указанные ниже.

На Unix-подобных операционных системах:

```
git clone https://github.com/aimclub/iOpt
cd iOpt
pip install virtualenv
virtualenv ioptenv
source ioptenv/bin/activate
python setup.py install
```

На операционных системах семейства Windows:

```
git clone https://github.com/aimclub/iOpt
cd iOpt
pip install virtualenv
virtualenv ioptenv
ioptenv\Scripts\activate.bat
```

Фреймворк методов интеллектуальной эвристической оптимизации iOpt
Описание процессов, обеспечивающих поддержание жизненного цикла

```
python setup.py install
```

4. Процессы жизненного цикла iOpt

Жизненный цикл iOpt представлен на Рис. 1.

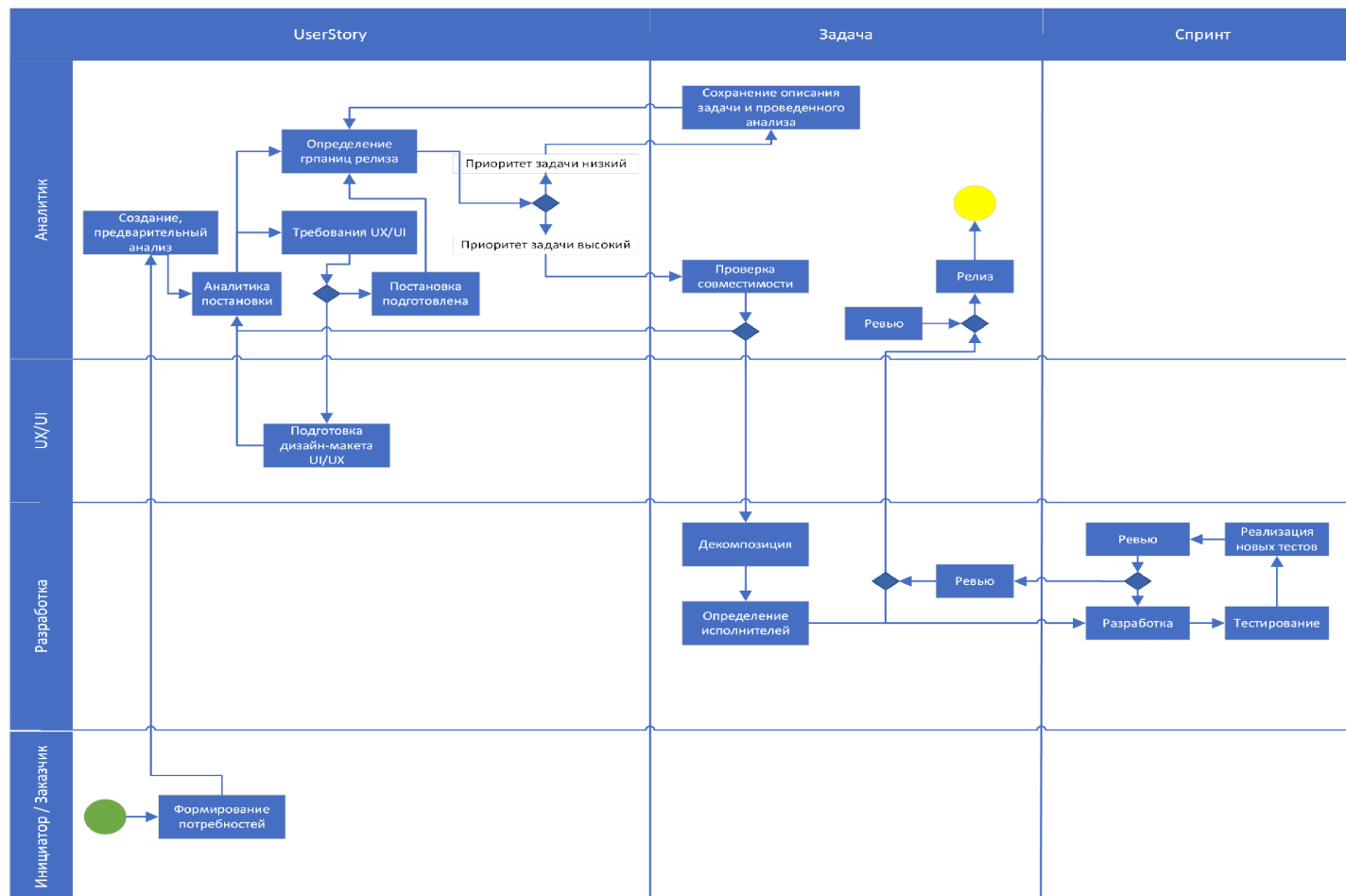


Рисунок 1 Схема процесса Жизненный цикл Системы

4.1. Развитие iOpt

Фреймворк iOpt разрабатывается в соответствии с внутренними планами команды-разработчика. Обновление фреймворка осуществляется по мере готовности к выходу очередного релиза.

При формировании планов учитываются:

- заявки клиентов, поступающие в ходе сопровождения;
- функциональные возможности, преимущества и недостатки аналогичных систем.

4.2. Процессы, обеспечивающие поддержание жизненного цикла iOpt

В основу процессов, обеспечивающих поддержание жизненного цикла фреймворка, заложен ГОСТ Р ИСО/МЭК 12207-2010 «Информационная технология. Системная и программная инженерия. Процессы жизненного цикла программных средств».

4.2.1. Процесс менеджмента документации

Цель процесса менеджмента документации – разработка и сопровождение зарегистрированной информации по программному обеспечению, сформированной в результате процессов ЖЦ.

Задачами процесса менеджмента документации являются:

- определение стандартов, которые применяются при разработке программной документации;

- определение документации, которая производится процессом ЖЦ;
- определение и утверждение содержания и целей всей документации;
- разработка документации и организация доступа к ней в соответствии с внутренними стандартами;
- сопровождение документации в соответствии с определенными критериями.

Стратегия менеджмента документации оформляется в соответствии с внутренними стандартами. Документация, включает в себя:

- заголовок или название;
- цели и содержание;
- круг пользователей, которым она предназначена;
- процедуры и ответственность при формировании исходных данных, разработке, ревизиях, модификации, утверждении, производстве, хранении, распределении, сопровождении и менеджменте конфигурации;
- графики создания промежуточных и окончательных версий.

Каждый идентифицированный документ разрабатывается в соответствии с подходящими стандартами на документацию, регламентирующими носители, форматы, описание содержания, нумерацию страниц, размещение рисунков и таблиц, пометки о правах собственности и секретности и другие элементы представления.

Подготовленные документы рассматриваются и редактируются по формату, техническому содержанию и стилю представления в соответствии со стандартами на документацию. Перед выпуском документов они проверяются уполномоченным лицом.

Документы разрабатываются и поставляются в соответствии с планом. При производстве и распределении документов могут использоваться бумажные, электронные или другие носители. Важные материалы хранятся в соответствии с требованиями по содержанию записей, защищенности, сопровождению и резервированию.

Изменения в документацию вносятся при выполнении процесса сопровождения программного обеспечения.

4.2.2. Процесс менеджмента конфигурации программного обеспечения

Цель процесса менеджмента конфигурации программного обеспечения - установление и сопровождение целостности программных объектов и обеспечение их доступности для заинтересованных сторон.

Задачами процесса менеджмента конфигурации программного обеспечения являются:

- определение схемы обозначения программного обеспечения и его модификаций (объектов программной конфигурации);
- контроль модификаций и выпусков программного обеспечения;
- регистрация и предоставление информации о статусе модификаций;
- контроль хранения, обработки и поставки модификаций.

Управление конфигурацией включает в себя:

- идентификацию и регистрацию заявок на изменения;
- анализ и оценку изменений;
- принятие или отклонение заявок;
- реализацию, верификацию и выпуск модифицированной составной части;
- проверочные испытания, на основании которых можно проследить каждую модификацию, ее причины и полномочия на проведение изменений;
- оценка законченности программного обеспечения с точки зрения реализации установленных к ним требований.

4.2.3. Процесс обеспечения гарантии качества программного обеспечения

Цель процесса обеспечения гарантии качества программного обеспечения – предоставление гарантии соответствия рабочей продукции и процессов предварительно определенным условиям.

Задачами процесса гарантии качества программного обеспечения являются:

- создание и поддержка свидетельств гарантии качества;
- идентификация и регистрация проблем и (или) несоответствий с требованиями;

- верификация соблюдения продукцией, процессами и действиями действующих стандартов, процедур и требований.

Процесс гарантии качества ведется координированно с процессами верификации программного обеспечения, валидации программного обеспечения, ревизии и аудита программного обеспечения.

4.2.4. Процесс верификации программного обеспечения

Цель процесса верификации программного обеспечения – подтверждение соответствия программного обеспечения заданным требованиям.

Задачами процесса верификации программного обеспечения являются:

- определение критериев верификации;
- выполнение требуемых действий по верификации;
- определение и регистрация дефектов;
- предоставление результатов верификации заинтересованным сторонам.

Процесс верификации включает следующие этапы:

- верификация требований (осуществляется с учетом следующих критериев: системные требования являются согласованными, выполнимыми и тестируемыми; системные требования соответственно распределены по техническим, программным элементам

согласно критериям проекта; требования к программному обеспечению согласованы, выполнимы, проверяемы и точно отражают системные требования; требования к программному обеспечению, связанные с безопасностью, защитой и критичностью, являются корректными);

- верификация кода (осуществляется с учетом следующих критериев: код является следствием проекта и требований тестируемости, правильности и соответствует установленным требованиям и стандартам, относящимся к кодированию; код осуществляет надлежащую последовательность событий, имеет согласованные интерфейсы, корректные данные и поток команд управления; код корректно реализует требования по безопасности, защищенности и другим критическим свойствам);
- верификация документации (осуществляется с учетом следующих критериев: документация является адекватной, полной и согласованной; подготовка документации осуществляется своевременно; менеджмент конфигурации документов следует установленным процедурам).

4.2.5. Процесс валидации программного обеспечения

Цель процесса валидации программного обеспечения – подтверждение выполнения требований для конкретного применения рабочего программного обеспечения.

Задачами процесса валидации программного обеспечения являются:

- определение критериев валидации;
- идентификация и регистрация проблем;

- обеспечение свидетельств того, что созданное программное обеспечение пригодно для применения по назначению;
- предоставление результатов действий по валидации заинтересованным сторонам.

В рамках работ разрабатывается план валидации для подтверждающей проверки программного обеспечения. Определяются задачи валидации, связанные с ними методы, технологии и инструментарий.

Проблемы и несоответствия, обнаруженные в процессе работ по валидации, передаются в процесс решения проблем в программном обеспечении.

При валидации на основе выбранных требований к тестированию разрабатываются тестовые примеры и спецификации для анализа результатов тестирования.

Процесс валидации включает в себя следующие проверки:

- тестирование в условиях повышенной нагрузки, граничных значений параметров и нестандартных входов;
- тестирование программного обеспечения на его способность изолировать и минимизировать влияние ошибок, то есть осуществлять плавную деградацию после отказов, в условиях повышенной нагрузки, граничных значений параметров и нестандартных входов;
- тестирование успешности решения намеченных задач основными пользователями программного обеспечения;
- тестирование программного обеспечения на соответствие своему назначению.

4.2.6. Процесс ревизии программного обеспечения

Цель процесса ревизии программного обеспечения – поддержание общего понимания у заинтересованных сторон относительно целей, достигнутых результатов и дальнейших действий. Ревизии программного обеспечения применяются как на уровне менеджмента проекта, так и на техническом уровне и проводятся в течение всей жизни проекта.

Задачами процесса ревизии программного обеспечения являются:

- выполнение технических ревизий и ревизий менеджмента на основе потребностей проекта;
- оценка состояния и результатов действий;
- предоставление результатов ревизии всем участвующим сторонам;
- идентификация и регистрация рисков и проблем.

Периодические ревизии проводятся в предварительно определенные сроки.

Для проведения каждой ревизии устанавливаются:

- план ревизии;
- состав программного обеспечения (результатов деятельности);
- проблемы, подлежащие обсуждению;

- области применения и процедуры;
- исходные и итоговые критерии для ревизии.

Проблемы, выявленные при проведении ревизии, регистрируются и передаются в процесс решения проблем в программном обеспечении.

Результаты ревизии документируются. Дается оценка адекватности ревизии (например, принятие, непринятие или условное принятие результатов ревизии). Результаты ревизии предоставляются заинтересованным сторонам. Участвующие стороны согласовывают итоговый результат ревизии, ответственность за позиции, требующие действий, и критерии завершения.

4.2.6.1. *Ревизии менеджмента проекта*

При ревизии менеджмента проекта проводится оценка проекта по отношению к планам проекта, графикам работ, стандартам и руководящим указаниям. Итоговые результаты ревизии представляют на рассмотрение соответствующему руководству, предусматривая:

- активизацию работ в соответствии с планом, основанную на оценке деятельности или состояния программного обеспечения;
- поддержание глобального управления проектом посредством соответствующего распределения ресурсов;
- изменение направления развития проекта или определение потребности в дополнительном планировании;
- оценку и руководство решением вопросов, связанных с риском, которые могут угрожать успеху проекта.

4.2.6.2. *Технические ревизии*

Технические ревизии проводятся для оценки программного обеспечения с позиции следующих критериев:

- полнота комплектации;
- соответствие принятым стандартам и спецификациям;
- соответствие процессу менеджмента конфигурации;
- соответствие установленному графику работ.

4.2.7. Процесс аудита программного обеспечения

Цель процесса аудита программного обеспечения – независимое определение соответствия программного обеспечения требованиям, планам и соглашениям.

Задачами процесса аудита программного обеспечения являются:

- определение соответствия программного обеспечения требованиям, планам и соглашениям;
- выявление проблем и передача их для решения ответственным сторонам.

Проверки проводятся в предварительно установленные сроки.

По каждому аудиту устанавливается:

- план проведения аудита;
- состав проверяемого программного обеспечения и результатов деятельности;
- область распространения и процедуры аудита;
- исходные и итоговые критерии проведения аудита.

Проблемы, выявленные при проведении аудитов, передаются процессу решения проблем в программном обеспечении.

Результаты аудита документально оформляются и представляются проверяемой стороне. Проверяемая сторона согласовывает представленный отчет и сообщает о планируемых решениях соответствующих проблем.

4.2.8. Процесс решения проблем в программном обеспечении

Цель процесса решения проблем в программном обеспечении – гарантированные идентификация, анализ, контроль и управление решением выявленных проблем.

Задачами процесса решения проблем в программном обеспечении являются:

- регистрация, идентификация и классификация проблем;
- анализ и оценка проблем для определения приемлемого решения (решений);
- выполнение решений проблем;

- отслеживание проблем вплоть до их закрытия.

Процесс решения проблем в программном обеспечении является циклическим. Обнаруженные в других процессах проблемы вводятся в процесс решения проблем. Каждая проблема классифицируется по категории и приоритету для облегчения анализа тенденций и решения проблем. По этим проблемам инициируются необходимые действия. При необходимости заинтересованные стороны информируются о существовании проблем. Проводится анализ тенденций в известных проблемах. Устанавливаются и анализируются причины проблем, которые далее, если возможно, устраняются. Состояние проблемы отслеживается и отражается в отчетах.

4.3. Эксплуатация и сопровождение системы

Эксплуатация системы осуществляется сотрудниками структурных подразделений Клиента. Техническое обслуживание Системы в процессе ее эксплуатации выполняется персоналом технических служб Клиента.

Состав работ в рамках сопровождения:

- Прием, регистрация, классификация и маршрутизация заявок (обращений) Клиента в отношении ПО.
- Предоставление Клиенту информации о ходе исполнения ранее принятой заявки.
- Информирование Клиента о результате обработки принятой заявки, подтверждение решения.
- Решение инцидентов, связанных с работой ПО:
 - анализ и устранение ошибок и сбоев в программных модулях ПО;
 - предоставление и реализация рекомендаций и/или временных решений по исправлению ошибок и сбоев в программных модулях ПО;

- проведение внутреннего тестирования версий ПО / патчей с исправленными ошибками;
- предоставление версий ПО / патчей с исправленными ошибками, их установка уделенным методом подключения, в случае необходимости установки версий ПО / патчей на объекте, предоставляется инструкция по установке;
- изменение и настройка параметров ПО в целях устранения сбоев;
- предоставление обновленной документации в случае, если при изменениях, выполняемыми в ходе оказания услуг или обновления ПО, данные в текущей документации становятся не актуальными.

Сотрудники и компетенции у правообладателя

№	Направление	Компетенции	Выполняемые работы	Количество специалистов
1	Разработка	Python, AI, ML	Техническая поддержка, Гарантийное обслуживание, Модернизация	2
2	Тестировщики	Опыт разработки автотестов, функционального и нагрузочного тестирования	Техническая поддержка, Гарантийное обслуживание, Модернизация	1
3	Технические писатели	Опыт разработки документации по ГОСТ 19, 34	Техническая поддержка, Гарантийное	1

			обслуживание	
--	--	--	--------------	--

Процесс сопровождения ПО включает в себя устранение выявленных ошибок в функционировании ПО путем обновления версии ПО / установкой патча с исправленными ошибками.

Работы по сопровождению не включают в себя работы по расширению существующей функциональности ПО.

Сопровождение выполняется по договору в соответствии с регламентом оказания услуг.

4.3.1. Гарантийное обслуживание

Гарантийное обслуживание обеспечивается специалистами ФГАО УВО "НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО" с учетом выполнения следующих требований для поступающих запросов:

- документально подтвержденное свидетельство наличия сбоя;
- подтверждение того, что сбой произошел по вине разработчика;
- последовательность действий пользователя в системе, вызывающих данную ошибку;
- корректная эксплуатация программного обеспечения в соответствии с инструкцией;
- отсутствие самостоятельного вмешательства клиента в устройство программного обеспечения.

Выполнение гарантийных обязательств обеспечивается специалистами, обозначенными в п. 4.3.

В рамках задач по гарантийному обслуживанию предоставляются услуги:

- Анализ выявленных и подтвержденных Клиентом ошибок и неисправностей в базовом программном обеспечении и доработки базового программного обеспечения.
- Исправление по запросу ошибок и неисправностей программного обеспечения. Под ошибкой понимается ситуация полной или частичной неработоспособности программного обеспечения, вызванная неверной логикой, заложенной в информационную систему.

Работы по исправлению ошибок не включают:

- Ситуации, вызванные неверным функционированием оборудования, операционной системы, системного программного обеспечения, которые влияют на функционирование системы.
- Ситуации, связанные с неверным функционированием системы при несоблюдении пользователями технических требований.
- Ситуации, связанные с неверным функционированием системы, возникшие в результате самостоятельного вмешательства пользователей в устройство программного обеспечения или неверной его настройкой.

4.3.2. Техническая поддержка

В рамках работ по технической поддержке пользователей обеспечивается помощь пользователям при работе в программном обеспечении.

Осуществляется решение возникающих конкретных проблем в работе пользователя с ПО, включая:

- устранении возникающих ошибок и неисправностей в работе программного обеспечения;
- помощь в настройке и обновлении ПО;
- консультации по работе продукта – при возникновении вопросов или сложностей с использованием конечными пользователями.

4.4. Модернизация

Выполнение работ по модернизации ПО обеспечивается специалистами, обозначенными в п. 4.3.

ФГАО УВО "НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО" производит работы по обновлению продукта, содержащие изменения ПО в части задач:

- улучшения функциональности ПО;
- адаптации ПО под обновление ОПО.

Сотрудники и компетенции у правообладателя

№	Направление	Компетенции	Выполняемые работы	Количество специалистов
1	Разработка	Python, AI, ML	Техническая поддержка, Гарантийное обслуживание, Модернизация	2
2	Тестировщики	Опыт разработки	Техническая	1

		автотестов, функционального и нагрузочного тестирования	поддержка, Гарантийное обслуживание, Модернизация	
3	Технические писатели	Опыт разработки документации по ГОСТ 19, 34	Техническая поддержка, Гарантийное обслуживание	1

4.5. Описание процесса работы

Ошибки, выявленные в ходе эксплуатации, устраняются в течение месяца, с учетом приоритетов. После того, как ошибка(-и) устранены, Клиенту оперативно передается обновление с учетом необходимых исправлений.

Выпуск стабильных версий производится с периодичностью раз в три месяца без автоматического обновления версий ПО, установленных на стороне клиентов.

С выпуском новой версии программного продукта производитель сопровождает ее следующими документами:

- «Описание версии ПО», в котором отражены изменения компонентов ПО.

Приоритеты заявок:

- **Критический** – к наивысшему приоритету относятся ситуации, в которых ПО находится в аварийном или предаварийном состоянии. Данное состояние оказывает влияние на всех пользователей ПО.
- **Низкий** – заявки, связанные с пожеланиями по скорости работы, способам отображения, реализацией дополнительных функций, не предусмотренных в работе ПО, а также другие типовые обращения, решаемые в ходе этого обращения, путем предоставления

инструкций, памяток и других простых материалов, и рекомендаций, при условии выполнения системой требований ТЗ по скорости работы ПО.

Предложения клиента на доработку ПО направляются по выделенным каналам согласно п. 4.5.1 производителя, оцениваются и включаются в план релизов iOpt.

Взаимодействие со службой технической поддержки разработчика, в том числе по вопросам возникающих ошибок, которые не могут быть устранены без доработки исходного кода, осуществляется по выделенному почтовому адресу.

Заявки от клиентов решаются последовательно друг за другом в порядке их поступления и приоритета. Время решения зависит от сложности заявки.

4.5.1. Каналы доставки запросов

Запросы в адрес разработчика могут поступать по каналам:

- Телефонная линия: +7 (812) 909 31 56
- Электронная почта: sai.center@itmo.ru

Адрес физического расположения поддержки:

ФГАО УВО "НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО"

Адрес: Г. САНКТ-ПЕТЕРБУРГ, ПРОСП. КРОНВЕРКСКИЙ, 49