РЕКФЭЙСИС

ID-GUARD

Справка по обеспечению жизненного цикла разработки ПО в компании Рекфейсис

IDGD.TI.DOC.996

СОДЕРЖАНИЕ

1 СВЕДЕНИЯ О ПРОЦЕССЕ РАЗРАБОТКИ И ПОДДЕРЖКИ

1.1 ДАННЫЕ О ПЕРСОНАЛЕ, ЗАДЕЙСТВОВАННОМ В ПРОЦЕССЕ РАЗРАБОТКИ

Для разработки и поддержки ПО, разрабатываемого в компании Рекфейсис, помимо административного персонала задействованы следующие специалисты:

1. Системный архитектор (Architect) – 1 чел. (высшее образование)

2. Аналитики (Team Lead + Business) – 3 чел. (высшее образование)

3. Отдел разработки ПО (Software developers) – 4 чел. (высшее образование)

4. Инженер службы безопасности (Security engineer) - 1 чел. (высшее образование)

5. Отдел тестирования (QA) – 3 чел. (высшее образование)

6. Отдел сопровождения – 3 чел. (высшее образование)

Средний стаж сотрудников, занятых в разработке, отладке, тестировании ПО и технической поддержке составляет >10 лет.

1.2 ДАННЫЕ ОБ АДРЕСЕ, ПО КОТОРОМУ ОСУЩЕСТВЛЯЕТСЯ ПРОЦЕСС РАЗРАБОТКИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

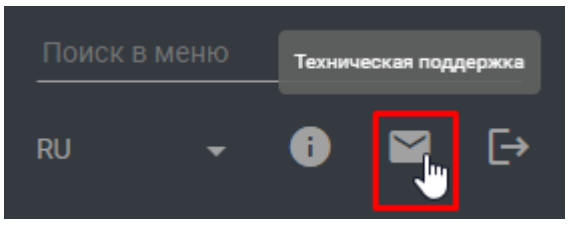
Разработка производится по адресу: 119334, Москва, 5-й Донской пр-д, 21Б, стр.10

1.3 НЕОБХОДИМЫЕ СВЕДЕНИЯ О ПРОЦЕССЕ СОПРОВОЖДЕНИЯ ПО

Техническая и консультационная поддержка осуществляется в формате online по следующим каналам коммуникации:

Электронная почта по продукту Id-Guard: Id-Guard@RecFaces.ru

Заявка также может быть отправлена из интерфейса Id-Guard, нажатием на кнопку с изображением конверта.



Режим работы технической поддержки: с понедельника по пятницу с 9 до 18 по Московскому времени. За исключением общегосударственных выходных и праздничных дней.

Техническая поддержка по умолчанию оказывается на русском языке.

В процессе сопровождения задействованы специалисты отдела сопровождения в полном объеме (3 чел).

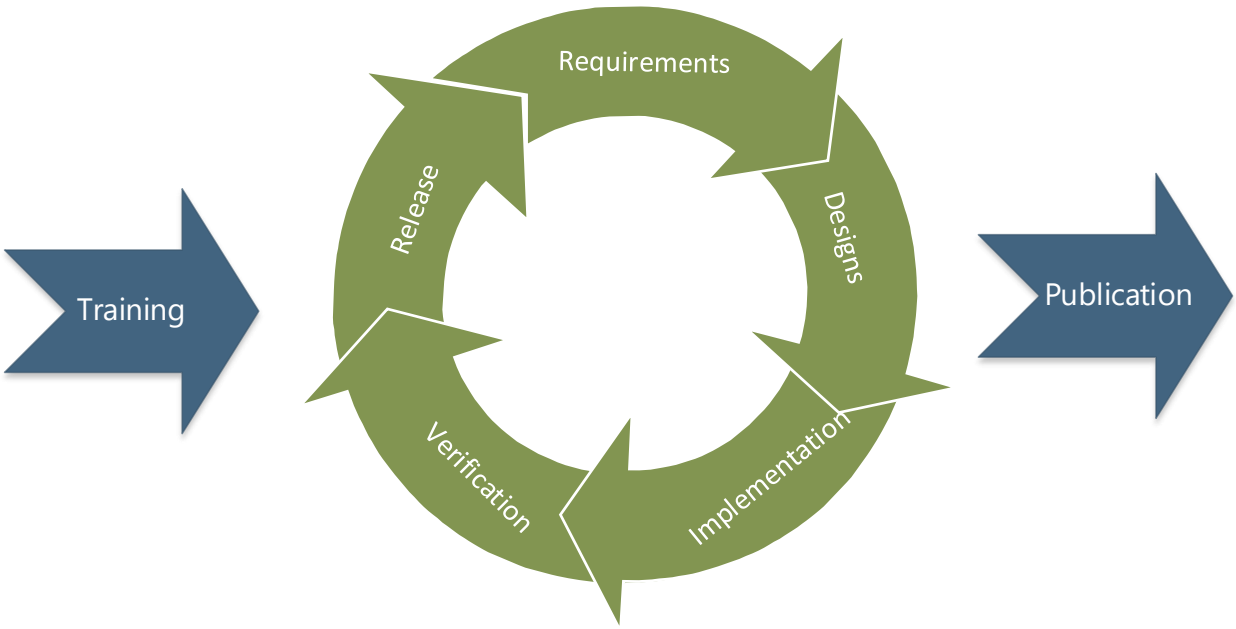
В зависимости от сложности запроса и необходимой квалификации, дополнительно могут быть задействованы специалисты других подразделений.

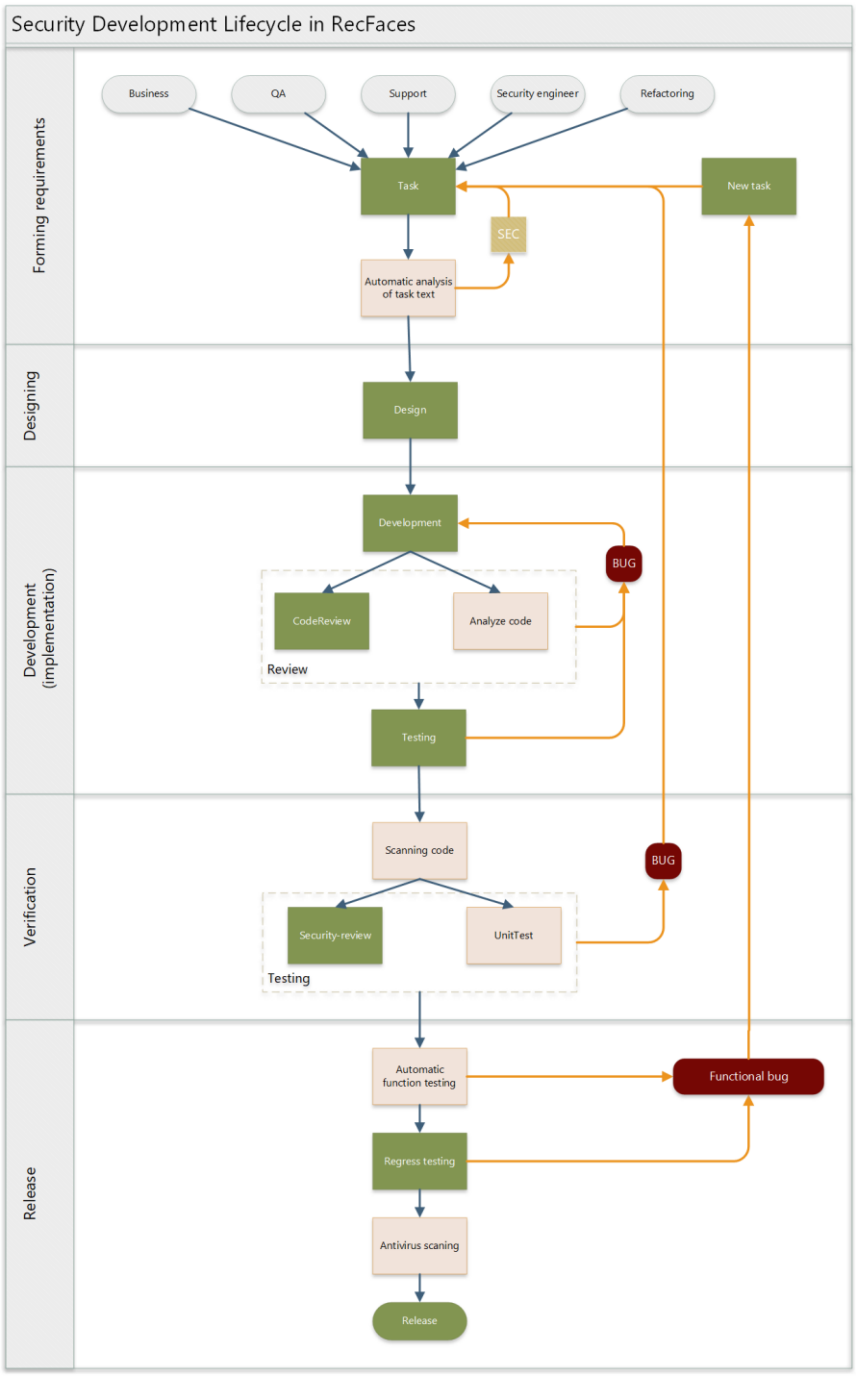
1.4 ДАННЫЕ ОБ АДРЕСЕ, ПО КОТОРОМУ ОСУЩЕСТВЛЯЕТСЯ ПОДДЕРЖКА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

Сопровождение осуществляется специалистами, располагающимися по адресу: 119334, Москва, 5-й Донской пр-д, 21Б, стр.10

2 СВЕДЕНИЯ О ЦИКЛЕ РАЗРАБОТКИ ПО, ПРИНЯТОМ В КОМПАНИИ РЕКФЕЙСИС

На базе существующих в мире практик по циклу безопасной разработки, в компании сформировался свой подход, охватывающий ключевые потребности и позволяющий сформировать максимально качественное и безопасное решение.





2.1 ОБУЧЕНИЕ (TRAINING)

Этап обучения является неотъемлемым элементом SDL в компании и нацелен на повышение общего уровня знаний и компетенций сотрудников.

2.1.1 ПОДГОТОВКА GUIDE LINE

Для предотвращения формирования скрытых ошибок и возможных проблем по информационной безопасности, по всем возможным направлениям сформированы требования и рекомендации.

Для аналитиков:

- Требования к постановке задач в JIRA.
- Требования к оформлению статей в Confluence.
- Рекомендации по тегированию задач признаком #SEC.

Для инженеров и специалистов службы поддержки:

- Требования к формированию паролей.
- Рекомендации по информационной безопасности при консультации заказчиков.
- Список вопросов и ответов.

Для архитекторов и специалистов информационной безопасности:

- Правила контроля требований.
- Правила проверок по информационной безопасности.

Для тестировщиков:

- Требования к информационной безопасности при проверке задач.
- Требования к подготовке чек-листов (с учетом ИБ).
- Требования к подготовке тест-кейсов (с учетом ИБ).

2.1.2 РЕГУЛЯРНОЕ ОБУЧЕНИЕ СОТРУДНИКОВ

Для всех технических специалистов мы проводим регулярные обучения и внутренние митинги по вопросам информационной безопасности.

В соответствии со внутренним регламентом компании установлены следующие правила проведения обучения:

1. Один раз в год все сотрудники проходят обучение привлеченным сотрудником информационной безопасности.
2. Один раз в 3 месяца руководитель группы (Teamlead) проводит инструктаж членов команды по правилам разработки.
3. Один раз в месяц проводятся внутренние митинги с обсуждением вопросов ИБ.

2.1.3 ФОРМИРОВАНИЕ BEST PRACTICE ДЛЯ ПРОДУКТА

Все члены команды и инженер службы информационной безопасности повышают свой уровень знаний и на базе этого формируют набор дополнительных правил, которые позволяют нивелировать возможные проблемы информационной безопасности при разработке и эксплуатации продуктов.

Все вновь выработанные практики доводятся до сведений всех членов команды и в случае необходимости используются в процессе разработки.

2.1.4 УЧАСТИЕ В КОНФЕРЕНЦИЯХ И ВЕБИНАРАХ

На регулярной основе наши специалисты посещают вебинары и конференции по информационной безопасности и защите информации (очно и онлайн).

Цель участия сотрудников в вебинарах и конференциях - повышение компетенции в рамках изучения тенденций информационной безопасности.

Компания заинтересована в повышении эффективности и безопасности использования информационных и коммуникационных технологий

2.2 ФОРМИРОВАНИЕ ТРЕБОВАНИЙ (REQUIREMENTS)

Для повышения качества продукта компания непрерывно развивает функциональные возможности, решения, улучшает и упрощает пользовательский интерфейс, расширяет список интегрированных систем, а также, устраняет программные и архитектурные проблемы. Источниками новых изменений продукта выступают:

- Business – при анализе потребностей рынка, потребностей заказчиков и новых технологий запросы формируется road map по развитию продуктов.
- QA – в процессе тестирования формируются как список ошибок, так и список новых доработок.
- Support – со стороны службы поддержки приходят запросы на устранение проблем и добавление новых функций.
- Security engineer – инженер по информационной безопасности анализирует работу продуктов в компании, а также, на территории заказчика, формирует требования по модернизации продукта с целью предотвращения потенциальных проблем.
- Refactoring – разработчики в процессе разработки могут локализовать проблемные участки кода и формировать задачи по его оптимизации.

Учет запросов осуществляется в менеджере задач Atlassian Jira, что позволяет организовать рабочий процесс с учетом требований SDL.

Для учета задач, которые могут затрагивать информационную безопасность используется метка #SEC. Данная метка может быть установлена:

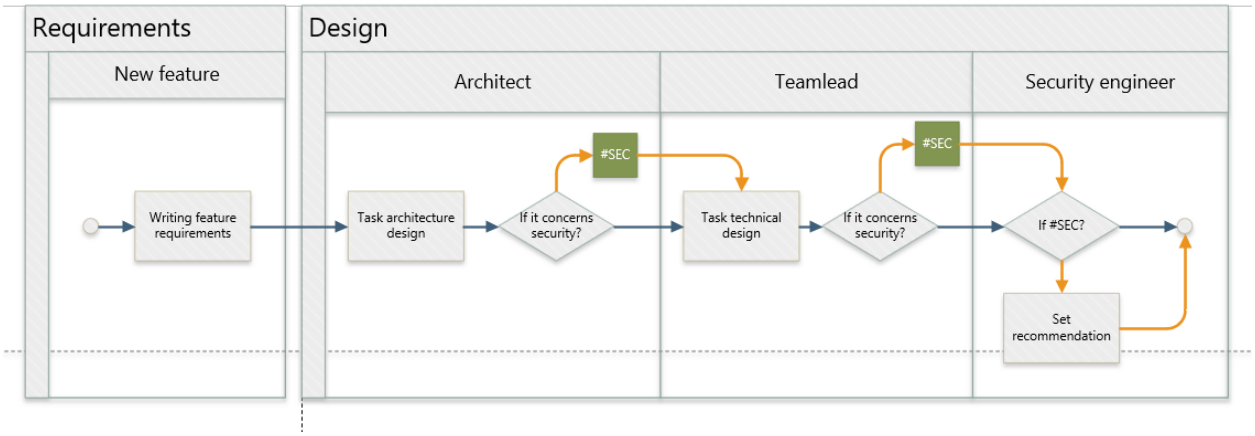
- Инженером службы безопасности – в случае решения проблемы ИБ.
- Архитектором – если изменения затрагивают ИБ.
- Аналитиком – при подозрении, что решение задачи коснется ИБ.
- Автоматическим сканером текста задачи – при обнаружении в тексте задач ключевых слов, относящихся к ИБ.

Наличие метки #SEC сигнализирует всем участникам, что в ходе проектирования, разработки и тестирования необходимо обратить повышенное внимание на ИБ.

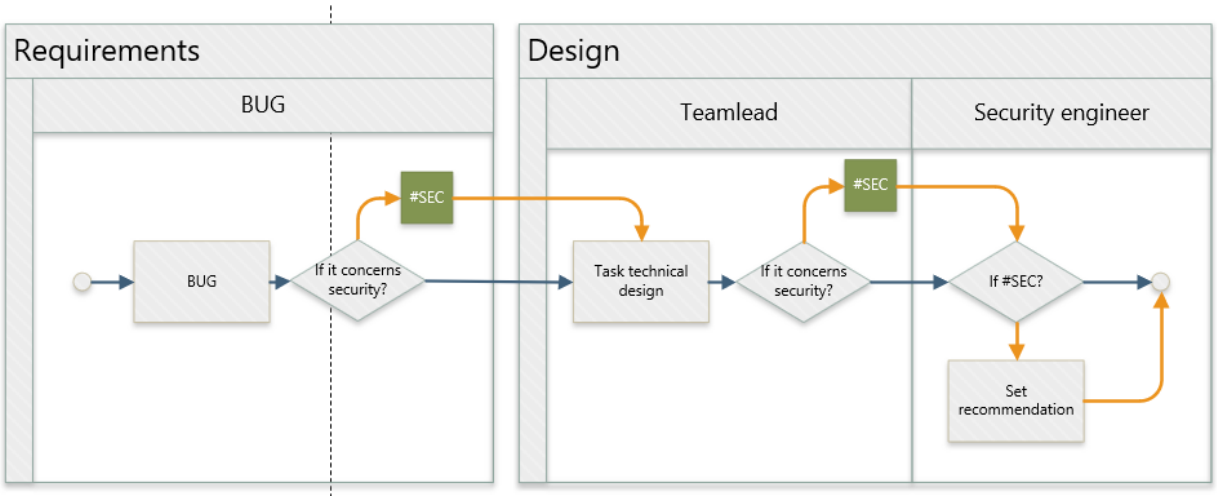
2.3 ПРОЕКТИРОВАНИЕ (DESIGN)

Проектирование является обязательным этапом работ в жизненном цикле разработки, и через него проходят все доработки и исправление ошибок. Но, в зависимости от типа задачи, в проектировании могут участвовать разные сотрудники.

2.3.1 ДОРАБОТКИ



2.3.2 ОШИБКИ



- Architect – По сформулированным бизнес-требованиям проектирует архитектуру доработки, прикрепляя к задаче связи на необходимые регламенты и требования. В случае, если задача может затронуть информационную безопасность, ставит задаче метку #SEC.
- Teamlead – производит техническое проектирование реализации задачи. В случае, если реализация задачи затрагивает компоненты информационной безопасности, устанавливает задаче метку #SEC.
- Security engineer – По всем задачам с меткой #SEC выдает рекомендации по информационной безопасности.

2.4 РАЗРАБОТКА (IMPLEMENTATION)

2.4.1 РЕАЛИЗАЦИЯ ЗАДАЧИ

Реализация задачи с учетом тренингов и консультаций со стороны ИБ.

2.4.2 CODEREVIEW

Все задачи, решенные программистом, проходят обязательную проверку кода, что позволяет разработать эффективное и качественное решение. При проверке кода используется комбинированный подход, т.е. код проверяется как руководителем команды, так и технической командой.

Руководитель группы разработки проверят важные фрагменты кода и задачи, помеченные тегом #SEC. Данный подход позволяет проводить критичную проверку важных фрагментов и не создает эффект замыливания при регулярной проверке.

Проверка кода коллегами создает комплексный эффект, позволяя поднимать общий уровень команды разработчиков, т.к. происходит фоновая передача знаний от одного члена команды другому.

2.4.3 СКАНИРОВАНИЕ КОДА

Весь программный код, разработанный программистами, проходит обязательную проверку при помощи статического анализатора кода.

Анализатор кода диагностирует различные проблемы и делит их на типы.

Type:

- Bug – устранение.
- Vulnerability – устранение.
- Code Smell – анализ и устранение в случае наличия глубоких проблем.

Severity:

- Blocker– устранение.
- Critical – устранение.
- Major – анализ и устранение.
- Minor – анализ и устранение.
- Info – анализ и устранение.

2.5 ПРОВЕРКА (VERIFICATION)

2.5.1 СКАНИРОВАНИЕ КОДА

Статическое тестирование безопасности приложений Static Application Security Testing (SAST) используется для сканирования (тестирования) программного кода продукта на наличие уязвимостей на этапе проверки. Когда разработанный код находится в своем исходном состоянии, статическое тестирование предназначено для поиска недостатков, таких как SQL-инъекция и т.д. Применяемое решение SAST в проектах компании предоставляет информацию, где и как исправить уязвимости в исходном коде. Встроенное в процесс разработки решение gosec позволяет автоматически возвращать задачи на доработку, помечать задачи меткой #SEC, или автоматически создавать ошибки.

2.5.2 SECURITY-REVIEW

Проверка изменений на информационную безопасность производится инженером по информационной безопасности в выборочном режиме по критичным задачам и блокам кода, затрагивающим механизмы:

- Авторизации.
- Идентификации.
- Шифрования.
- Хранения и обработке закрытых ключей.
- Хранения биометрических данных.
- Хранения и обработки данных пользователей.

2.5.3 МОДУЛЬНОЕ ТЕСТИРОВАНИЕ UNIT TEST

При разработке программного кода программисты параллельно реализуют Unit Test для всех критичных процедур и функций, а также обеспечивают 100% покрытие unit test'ами кода задач с меткой #SEC.

Модульное тестирование выполняется для проверки на корректность отдельных единиц работы исходного кода программы, минимальными из которых являются процедуры и функции.

2.6 РЕЛИЗ (RELEASE)

2.6.1 РЕГРЕССИОННОЕ ТЕСТИРОВАНИЕ

Регрессионное тестирование выполняется по подготовленному набору тестов, направленных на обнаружение дефектов в уже протестированных участках приложения. Данный вид тестирования выполняется для поиска и исправления регрессионных ошибок, появляющихся при добавлении в существующую сборку нового участка программы, или исправления других багов.

Таким образом, для нас цель регрессионного тестирования - убедиться, что исправление одних багов не стало причиной возникновения других, и что обновление сборки не создало новых дефектов в уже проверенном коде.

Применяются несколько видов регрессионных тестов:

- Верификационные тесты - проводятся для проверки исправления обнаруженного и открытого ранее бага.
- Тестирование верификации версии - содержит принципы дымного тестирования и тестирование сборки: проверка работоспособности основной функциональности программы в каждой новой сборке.
- Непосредственно само регрессионное тестирование – повторное выполнение всех тестов, которые были написаны и проведены ранее. Они выполняются по уже существующим тест-кейсам независимо от того, были в ходе их прохождения найдены баги, или нет.

- Тестирование уже исправленных багов в новой сборке. Выполняется для того, чтобы проверить, не возобновило ли обновление сборки старые дефекты.

2.6.2 АВТОМАТИЗИРОВАННОЕ ФУНКЦИОНАЛЬНОЕ ТЕСТИРОВАНИЕ (FUNCTIONAL AUTOMATION TESTING)

Применение автоматизированного функционального тестирования при проверке релиза позволяет решить несколько целей:

- Обеспечение повторяемости тестов – все написанные тесты всегда выполняются однообразно, то есть исключен «человеческий фактор». Тестировщик не пропустит тест по неосторожности и ничего не напутает в результатах.
- Быстрое тестирование – автоматизированному скрипту или тесту не требуется сверяться с инструкциями и документациями, это сильно экономит время выполнения.
- Меньшие затраты на поддержку – когда автоматические скрипты уже написаны, на их поддержку и анализ результатов требуется, как правило, меньшее время, чем на проведение того же объема тестирования вручную.
- Формирование отчетов и багов – автоматически формируемые отчеты помещаются в раздел релизов bconfluence, а при наличии ошибки, автоматически заводятся ошибка в jira.

2.6.3 СКАНИРОВАНИЕ АНТИВИРУСОМ

Для предотвращения заражения оборудования заказчика вредоносными программами, все файлы проходят обязательную антивирусную проверку двумя независимыми антивирусами:

- ESET File Security для Linux.
- Clam AntiVirus.

Сканирование всех файлов, встроенное в процесс выпуска продукта, и не зависящее от действий DevOps, позволяет избежать ошибки инженера при обнаружении подозрений о вредоносных элементах.

2.7 ПУБЛИКАЦИЯ (PUBLICATION)

Этап публикации продукта для доступа партнером и заказчиком является одним из важных этапов всего цикла безопасной разработки и требует особой тщательности и внимания. На данном этапе выполняются следующие обязательные действия:

- Подписание всех исполняемых файлов электронной подписью.
- Подписание всех pdf файлов электронной цифровой подписью.
- Формирование контрольной суммы по всем файлам, входящим в дистрибутив.
- Публикация инсталляторов релиза.
- Публикация списка изменений релиза.
- Публикация пресс-релиза.

3 АВАРИЙНЫЕ СИТУАЦИИ

Система обеспечивает корректную обработку аварийных ситуаций, вызванных неверными действиями администратора, неверным форматом или недопустимыми значениями входных данных. В указанных случаях администратору должны выдаваться соответствующие аварийные сообщения, после чего возвращаться в рабочее состояние, предшествовавшее неверной (недопустимой) команде или некорректному вводу данных. Аварийные ситуации могут возникать как из-за ошибок в программных продуктах, так и из-за неправильной настройки.

Признаками аварийной ситуации являются:

1. Отсутствие на экране необходимой страницы;
2. Окна с сообщениями о нештатной ситуации;
3. Окна с нечитаемыми сообщениями на английском языке;
4. Сообщение об отсутствии прав на действия.

4 СЛУЖБА ТЕХНИЧЕСКОЙ ПОДДЕРЖКИ

Если при эксплуатации Системы у вас возникли дополнительные вопросы или у вас появилась проблема, вы можете обратиться в письменной форме в службу технической поддержки. Для этого необходимо нажать на надпись «Служба поддержки» в нижней правой части экрана (Рисунок 1) и отправить письмо по email.

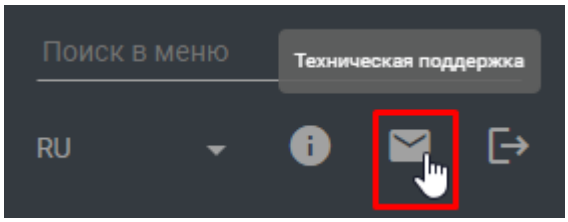


Рисунок 1. Служба поддержки

Также можно ознакомиться с доступной информацией на сайте www.recfaces.ru

ВНИМАНИЕ! Для быстрого и корректного ответа просьба в письме указывать номер версии, находящейся в окне, информация о Системе (Рисунок 2).

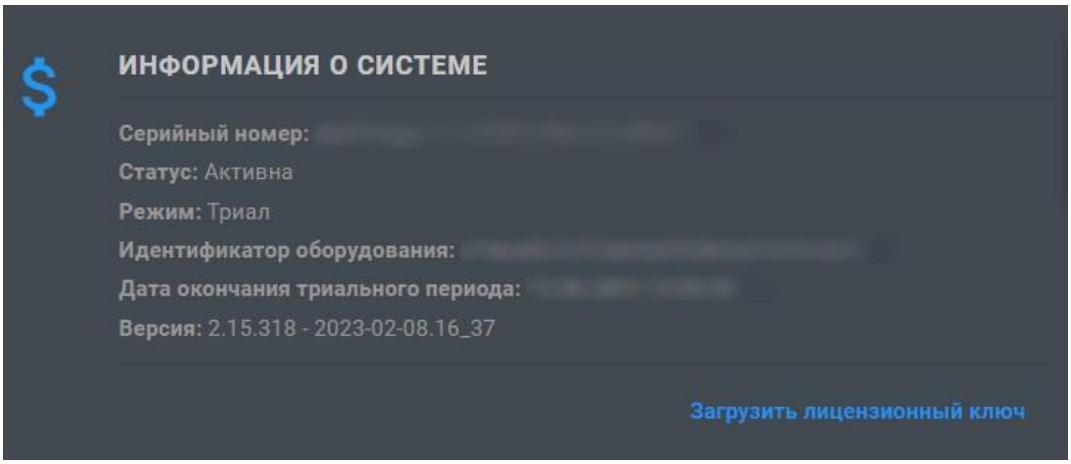


Рисунок 2. Номер версии