

ООО «ДиБиЭс Разработка»

Юр. адрес: 121609, г. Москва, вн.тер.г. Муниципальный Округ Крылатское,
ш. Рублёвское, д. 28

Телефон: +7(495) 665-94-09

ИНН/КПП: 9731146530/773101001

ОГРН: 1257700019279



Описание функциональных характеристик программного обеспечения Rf-b2b-onboarding 1.0

(версия документа 1.0)

г. Москва, 2026г.

Оглавление

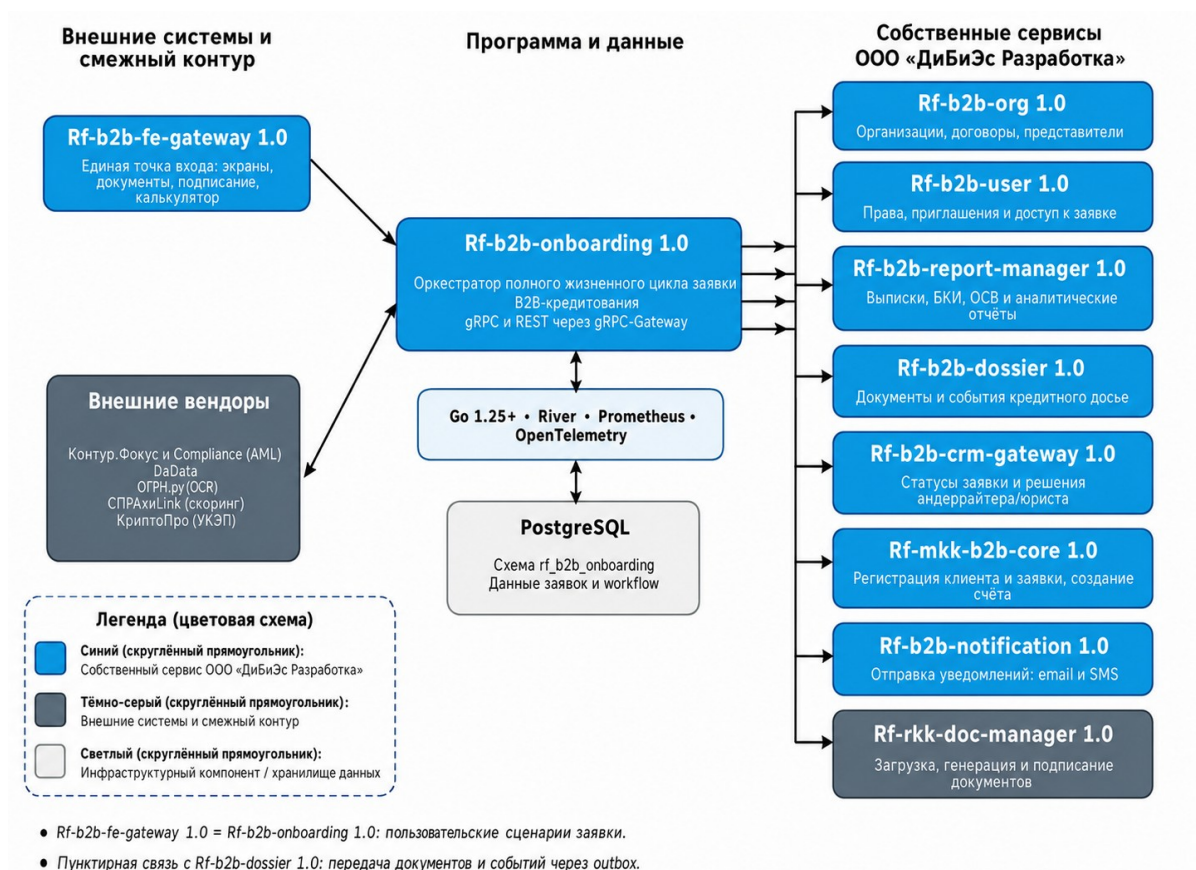
1. Глоссарий:.....	3
2. Верхнеуровневая архитектура использования Программы.....	4
3. Описание функциональных характеристик программного обеспечения.....	4
3.1. Цели и назначение.....	4
3.2. Подробное описание функциональных характеристик.....	4
3.2.1. Управление заявками и формами.....	4
3.2.2. Офферы, условия и договоры.....	4
3.2.3. Продукты, расчёты и интеграции.....	5
4. Характеристика функциональной структуры Программы.....	5
5. Сведения о программных модулях, являющихся неотъемлемыми компонентами программного обеспечения.....	5
6. Режим функционирования программного обеспечения.....	5
Приложение 1.....	6
API для интеграции с Rf-b2b-onboarding 1.0.....	6
1. Методы API.....	6
2. Описание объектов.....	7

Настоящий документ «Описание функциональных характеристик программного обеспечения Rf-b2b-onboarding 1.0» предназначен для описания функциональных характеристик программного обеспечения Rf-b2b-onboarding 1.0 (далее – Продукт, Программа) в части базового функционала, программных интерфейсов, подсистем и компонентов.

1. Глоссарий:

Разработчик, DBS	ООО «ДиБиЭс Разработка» (ОГРН: 1257700019279).
Rf-b2b-onboarding 1.0	Для целей создания интернет-сервиса, обеспечивающего полный жизненный цикл заявки на В2В-кредитование — от предварительного расчета условий финансирования и сбора данных клиента до принятия кредитного решения, выполнения условий выдачи и подписания договорной документации.
БД Rf-b2b-onboarding 1.0	PostgreSQL-база данных Программы со схемой rf_b2b_onboarding.
Пользователь	Пользователь клиентской системы, действующий от имени юридического лица или индивидуального предпринимателя.
Заявка	Агрегат процесса онбординга, включающий форму, оффер и договор.
Форма	Часть заявки со сведениями об организации, контактных лицах, адресах и документах.
Оффер	Часть заявки с решением, параметрами финансирования, документами, бенефициарами и условиями.
Договор	Часть заявки, содержащая параметры заключаемого договора, комплект договорных документов, сведения о подписантах и статусы подписания.
Workflow	Последовательность шагов обработки формы, оффера и договора с сохранением статусов.

2. Верхнеуровневая архитектура использования Программы



Rf-b2b-onboarding 1.0 реализован как серверное приложение на Go с gRPC-интерфейсом и REST-интерфейсом, формируемым gRPC-Gateway. Программа использует PostgreSQL, фоновые задания River, метрики Prometheus и трассировку OpenTelemetry.

Внутренние компоненты разделены на delivery-слой обработчиков, прикладные сервисы, доменную модель, workflow форм, офферов и договоров, репозитории PostgreSQL, фоновые задания и клиенты интеграций.

Входной точкой для пользовательских каналов является Rf-b2b-fe-gateway 1.0. В рамках кредитного процесса Программа взаимодействует с Rf-b2b-org 1.0, Rf-b2b-user 1.0, Rf-b2b-report-manager 1.0, Rf-b2b-dossier 1.0, Rf-b2b-crm-gateway 1.0, Rf-mkk-b2b-core 1.0, Rf-rkk-doc-manager 1.0 и Rf-b2b-notification 1.0. Контракты взаимодействия Программы определены в Protocol Buffers и опубликованы в Swagger 2.0.

3. Описание функциональных характеристик программного обеспечения

3.1. Цели и назначение

Программа Rf-b2b-onboarding 1.0 предназначена для автоматизации B2B-онбординга юридических лиц и индивидуальных предпринимателей, обработки заявок на финансирование и взаимодействия со смежными системами. Основными функциональными характеристиками Программы являются:

1. Управление заявками, формами и статусами workflow
2. Обработка офферов, документов, бенефициаров, условий и договоров
3. Управление продуктами, расчёт параметров финансирования и интеграции

Программа Rf-b2b-onboarding 1.0 обеспечивает валидацию входных данных, работу с UUID-идентификаторами, хранение состояния в PostgreSQL и интеграцию по REST/gRPC, как описано в разделе «Подробное описание функциональных характеристик».

3.2. Подробное описание функциональных характеристик

3.2.1. Управление заявками и формами

Программа Rf-b2b-onboarding 1.0 предоставляет полный цикл управления заявкой и формой:

- GetOrCreate находит существующую заявку по user_id и ИНН либо создаёт новую заявку; GetApplication, GetApplications, GetUserApplicationIds и GetApplicationOrgID возвращают состояние и идентификаторы заявок;
- SubmitAdditionalData сохраняет сведения анкеты;
- UploadDocument, DeleteDocument, GetDocumentForSign и DownloadDocuments обеспечивают работу с документами; SignDocuments выполняет подписание документов УКЭП;
- ReturnToEdit возвращает заявку на редактирование;
- GetApplicationWorkflowData выдаёт диагностические данные workflow, CheckMoratorium проверяет мораторий, а DeleteApplication доступен при включённой настройке APPLICATION_ALLOW_DELETE.

3.2.2. Офферы, условия и договоры

Программа Rf-b2b-onboarding 1.0 реализует обработку оффера после завершения формы:

- SubmitOfferAdditionalDocuments принимает сведения о бенефициарах и дополнительных документах; ConfirmBeneficiar и ReturnBeneficiarToEdit управляют этапом подтверждения бенефициаров;
- SubmitWaitingAdditionalDocuments, SubmitWaitingReCheckDocuments и SubmitConditionDocuments переводят оффер между этапами ожидания и проверки;
- SubmitDecision принимает решение из Rf-b2b-crm-gateway 1.0;
- UploadOfferDocument и DeleteOfferDocument управляют документами оффера; AddCondition ведёт каталог условий;
- SignContract подписывает документы договора.

3.2.3. Продукты, расчёты и интеграции

- CreateProduct создаёт продукт с лимитами суммы и срока, критериями, обеспечением и настройками процесса;
- Calculate рассчитывает доступные лимиты и платёж;
- TrancheCalculate — полный график платежей. Интеграционные клиенты обеспечивают обмен с компонентами B2B, системами проверки, распознавания и электронной подписи.

4. Характеристика функциональной структуры Программы

Программа представляет собой контейнеризованное серверное приложение для Linux. Клиент-серверная архитектура обеспечивает доступ по gRPC и HTTP/JSON; прикладное

состояние хранится в PostgreSQL. Программа предназначена для работы в защищённом контуре с контролируемым доступом к внешним системам.

5. Сведения о программных модулях, являющихся неотъемлемыми компонентами программного обеспечения

Во время работы Rf-b2b-onboarding 1.0 взаимодействует с Rf-b2b-org 1.0, Rf-b2b-user 1.0, Rf-rkk-doc-manager 1.0, Rf-b2b-report-manager 1.0, Rf-b2b-notification 1.0, Rf-mkk-b2b-core 1.0, опционально с Rf-b2b-dossier 1.0 и Rf-b2b-crm-gateway 1.0, а также с внешними системами DaData, DBrain, DMS/AxiLink, Kontur Focus Compliance и КриптоПро.

Если зависимая система недоступна или возвращает ошибку, Rf-b2b-onboarding 1.0 возвращает ошибку вызывающей системе либо сохраняет/повторяет выполнение через workflow и фоновые задания в соответствии с реализацией конкретной операции.

6. Режим функционирования программного обеспечения

В основном режиме функционирования программное обеспечение обеспечивает:

- непрерывную работу в режиме – 24 часа в день, 7 дней в неделю (24x7);
- выполнение всех функций в полном объеме, за исключением периодов проведения профилактических и других работ, а также устранения возникших нештатных ситуаций.

Неисправности, выявленные в ходе эксплуатации Программы, могут быть исправлены двумя способами:

- Массовое автоматическое обновление компонентов Программы;
- Единичная работа специалиста службы технической поддержки по запросу заказчик.

Приложение 1

API для интеграции с Rf-b2b-onboarding 1.0

1. МЕТОДЫ API

Сервис предоставляет следующий набор методов через конечные точки (endpoints):

Метод	HTTP-метод	Endpoint	Краткое описание	Основные параметры запроса
ApplicationService_DownloadDocuments	GET	/api/v1/applications/{appId}/documents/download	DownloadDocuments скачивает документы определённого типа из заявки в виде ZIP-архива	appId (string); docType (string)
ApplicationService_GetDocumentForSign	GET	/api/v1/applications/{appId}/documents/{docId}/for-sign	Получение документа для подписания	appId (string); docId (string); docKind (string)
ApplicationService_AddCondition	POST	/api/v1/conditions/catalog	Регистрация/обновление записи в каталоге условий (админская ручка). Если code уже есть в active-версии — старая запись деактивируется, создаётся новая с supersedes_id.	code (string); name (string); description (string); conditionType (v1ConditionType); verificationType (v1VerificationType); sanctionTypes (array<v1SanctionType>); department (v1Department)
CRMIntegrationService_SubmitDecision	POST	/api/v1/offers/{offerId}/decision	Решение по заявке из CRM	offerId (string); isApproved (boolean); approvedAmount (string/int64); approvedTerm (string/int64); monthlyRate (typeDecimal); conditionIds (array<string>); requestedDocTypes (array<v1DocType>); comment (string); customConditions (array<object>); rejectReason (v1CrmRejectReason); conditionResults (array<object>)
ApplicationService_GetApplication	GET	/application	Получение заявки	appId (string); offerId (string)
ApplicationService_DeleteApplication	DELETE	/application	Удаление заявки	appId (string); inn (string)
ApplicationService_SubmitAdditionalData	POST	/application/additional	Отправка дополнительных данных	appId (string); addressMatchesBusiness (boolean); ceoInn (string); registrationAddress (string); businessAddress (string); contactFullName (string); contactPhone (string); contactEmail (string); submitComplete (boolean); taxResidentConfirmed (boolean); pdlConfirmed (boolean); ceoPhone (string); taxSystem (v1TaxSystem); addresses (array<object>); hasPsn (boolean); businessAddressType (v1AddressType)
ApplicationService_ConfirmPassportData	POST	/application/confirm_passport	Deprecated: шаг WaitingApprove удалён	appId (string); confirm (boolean)
ApplicationService_DeleteDocument	DELETE	/application/document	Удаление документа	appId (string); docType (string); docId (string)
ApplicationService_UploadDocument	POST	/application/document	Загрузка документа	appId (string); docType (v1DocType); file (string/byte); fileName (string)
ApplicationService_GetOrCreate	POST	/application/get_or_create	Найти существующую заявку по (user_id, inn) или создать новую. Покрывает BR-072 / BR-073 / BR-074 / BR-208 / FR-111: дерево решений «своя/приглашённая active form» → «ЛК уровня (ВКЛ или подписанная форма)» → «создание». Ответ XOR: ровно одно из app_id или lk выставлено.	userId (string); inn (string); phone (string); term (string/int64); amount (string/int64); leadId (string)
ApplicationService_CheckMoratorium	GET	/application/moratorium	Проверка моратория по ИНН	inn (string)
ApplicationService_AddMoratorium	POST	/application/moratorium/add	Тестовый метод для QA: добавляет ИНН в мораторий на 7 дней.	inn (string)
ApplicationService_RemoveMoratorium	POST	/application/moratorium/remove	Тестовый метод для QA: удаляет ИНН из моратория.	inn (string)
ApplicationService_SignDocuments	POST	/application/sign	Подписание документов УКЭП	appId (string); signatures (array<object>);

Метод	HTTP-метод	Endpoint	Краткое описание	Основные параметры запроса
				beneficiaryStatementConfirmed (boolean); userId (string)
ApplicationService_GetApplicationWorkflowData	GET	/application/workflow-data	Получение сырых данных workflow заявки (для диагностики / QA)	appId (string)
ApplicationService_GetApplicationOrgID	GET	/application/{appId}/org_id	Получение org_id заявки. Лёгкий метод для проверки прав в gateway.	appId (string)
ApplicationService_GetApplications	GET	/applications	Получение списка заявок по ID	appIds (array); userId (string); stage (string); orgId (string)
ApplicationService_GetUserApplicationIds	GET	/applications/user	Получение активных id заявок пользователя в указанной организации (own + invited+accepted). Проверка прав делается внутри: вызывающему gateway не нужно дублировать логику.	userId (string); orgId (string)
ApplicationService_SignContract	POST	/contract/sign	Подписание документов контракта	appId (string); signatures (array<object>); financingConditionsConfirmed (boolean); userId (string)
ApplicationService_SubmitOfferAdditionalDocuments	POST	/offer/additional	Отправка дополнительных данных оффера (сумма лимита, бенефициары, галочки)	offerId (string); appId (string); ceolsSoleBeneficiary (boolean); beneficiaries (array<object>); submitComplete (boolean)
ApplicationService_ConfirmBeneficiar	POST	/offer/beneficiar/confirm	Подтверждение бенефициаров (кнопка "Продолжить")	offerId (string); appId (string)
ApplicationService_ReturnBeneficiarToEdit	POST	/offer/beneficiar/return-to-edit	Возврат на редактирование с экрана подтверждения бенефициаров	offerId (string); appId (string)
ApplicationService_SubmitConditionDocuments	POST	/offer/condition-documents	Отправка данных на шаге отлагательных условий (кнопка "Продолжить")	offerId (string); appId (string); submitComplete (boolean)
ApplicationService_DeleteOfferDocument	DELETE	/offer/document	Удаление документа оффера	offerId (string); appId (string); docType (string); docId (string)
ApplicationService_UploadOfferDocument	POST	/offer/document	Загрузка документа оффера (устав, решение, договор аренды)	offerId (string); appId (string); docType (v1DocType); file (string/byte); fileName (string)
ApplicationService_SubmitWaitingAdditionalDocuments	POST	/offer/waiting-additional-documents	Отправка данных на шаге ожидания дополнительных документов (кнопка "Продолжить")	offerId (string); appId (string); submitComplete (boolean)
ApplicationService_SubmitWaitingRecheckDocuments	POST	/offer/waiting-recheck-documents	Отправка данных на шаге повторной проверки документов (кнопка "Продолжить", после юр. проверки)	offerId (string); appId (string); submitComplete (boolean)
ProductService_CreateProduct	POST	/product	Создаёт конфигурацию кредитного продукта.	code (string); name (string); createdBy (string); amount.minAmount (string); amount.maxAmount (string); term.minTerm (integer); term.maxTerm (integer); criteria.isMsp (boolean); criteria.opf (array); collateral.beneficiaryGuarantee (boolean); collateral.companyGroupGuarantee (boolean); collateral.movablePropertyPledge (boolean); collateral.realEstatePledge (boolean); process.lenderType (string); process.processType (string); meta.title (string); meta.isActive (boolean)
ApplicationService_ReturnToEdit	POST	/v1/applications/{appId}/return-to-edit	Возврат на редактирование с шага WaitingSign	appId (string); ApplicationServiceReturnToEditBody
CalculatorService_Calculate	POST	/v1/calculator	Calculate рассчитывает график платежей по заданной сумме и сроку	amount (string/int64); term (integer/int32)
CalculatorService_TrancheCalculate	POST	/v1/calculator/tranche	TrancheCalculate рассчитывает полный график платежей для экрана выдачи транша	appId (string); amount (string/int64); term (integer/int32)

2. ОПИСАНИЕ ОБЪЕКТОВ

1. GetOrCreateRequest — запрос поиска или создания заявки.

- Поля: userId (UUID), inn (10 или 12 цифр), phone, term, amount, leadId (UUID).
- Ответ GetOrCreateResponse содержит либо appId, либо данные личного кабинета lk.

2. Application — агрегат заявки.

- Содержит идентификатор, бизнес-номер, Form, Offer, Contract, даты создания/истечения и идентификатор организации.

3. Form и FormData — форма заявки и её бизнес-данные.

- Содержат статус, недостающие поля, причину отказа, данные организации, адреса, контактные сведения, налоговый режим, документы и распознанные паспортные данные.

4. Offer и OfferData — оффер и сведения этапов оффера.

- Содержат одобренную сумму, сведения о бенефициарах, документы, отлагательные условия, сроки и комментарии проверки.

5. Contract и ContractData — договор и параметры финансирования.

- Содержат сумму, срок, ставку, документы для подписания, номер и дату договора, условия финансирования и срок действия.

6. UploadDocumentRequest и UploadOfferDocumentRequest — запросы загрузки документов.

- Поля включают appId/offerId, docType, file в формате bytes и fileName.

7. SignDocumentsRequest и SignContractRequest — запросы электронной подписи.

- Поля включают appId, массив signatures, подтверждающие признаки и userId.

8. CalculateRequest/Response — расчёт параметров финансирования.

- Запрос содержит amount и term; ответ — лимиты продуктов, минимальный ежемесячный платёж, месячную ставку и дату последнего платежа.

9. TrancheCalculateRequest/Response — расчёт графика транша.

- Ответ содержит массив платежей с датой, основным долгом, комиссией и итогом, а также ставку и суммы комиссий.

10. SubmitDecisionRequest — решение CRM по офферу.

- Содержит offerId, признак одобрения, сумму, срок, ставку, условия, запрошенные документы, комментарий, причину отказа и результаты проверки условий.