1. **Обзор и объем (Scope and overview)**: объем работ по тестированию и обзор тестируемой функциональности;

Новая функциональность заключается в возможности развернуть базу данных PostgreSQL в облачном сервисе хранения и обработки информации в БД, используя форму ввода данных в веб-интерфейсе и кнопки «создать». Обязательные поля ввода: имя базы данных, регион размещения, размер.

Объем работ содержит в себе все описанные ниже подходы к тестированию и их виды, составление списка требуемых инструментов тестирования и метрик, а также характеристики отраслевых стандартов и анализа рисков, которые могут возникнуть в ходе тестирования новой функциональности.

1. **Подходы к тестированию (Test Approach)**: виды тестирования, уровни тестирования, требования к окружениям, роли и обязанности каждого члена команды;

* Уровни тестирования (Test levels); Исходя из возможностей новой функциональности и опираясь на существующие требования, в общем виде описываем следующие уровни тестирования для новой функциональности:
  + Модульное тестирование(Unit testing): Проверка отдельных компонентов тестируемой функциональности, таких как форма ввода (имя базы данных, регион размещения, размер) и кнопка создания базы данных. Также можно проверить функции, отвечающие за обработку введенных данных, создание конфигурации и развертку базы данных.
  + Интеграционное тестирование(Integration testing): Проверка взаимодействия новой функциональности с другими модулями системы. В данном контексте, можно проверить, как сервисы на бэкэнде, отвечающие за обработку ввода данных взаимодействуют с сервисами,, отвечающими за создание и развертывание базы данных, как фронтэнд взаимодействует с бэкэндом и т.д.
  + Системное тестирование(System testing): Проверка работы новой функциональности в контексте всей системы. В данном случае, можно проверить, работает ли весь процесс создания базы данных со стороны бэкэнда как ожидается, когда пользователь заполняет форму и нажимает кнопку создать.
  + Приемочное тестирование(Acceptance testing): Проверка того, что новая функциональность соответствует требованиям бизнеса и удовлетворяет ожидания пользователей.
* Виды тестирования (Test Types); описание видов тестирования и подходы к тестированию новой функциональности:

| **Функциональное тестирование** | |
| --- | --- |
| **Объект** | **Тесты** |
| Поле ввода имени Базы Данных | 1) Поле принимает только допустимые значения (принимает буквы, цифры, не принимает спец. символы, вызывающие ошибки при работе с БД)  2) Поле не может быть оставлено пустым  3) Имя БД должно быть уникальным |
| Поле ввода региона размещения | 1) Поле принимает только доступные регионы размещения  2) Поле не может быть оставлено пустым |
| Поле ввода размера | 1) Поле принимает только допустимые значения (целые числа, соответствующие допустимым значениям)  2) Поле не может быть оставлено пустым  3) Размер БД не превышает максимально допустимый размер |
| Кнопка "Создать" | 1) Клик на кнопку "Создать" при условии корректно заполненной формы ввода приводит к успешному созданию БД  2) Попытка нажатия на кнопку "Создать" при не заполненной или не полностью заполненной форме ввода приводит к ошибке и появлению соответствующего сообщения |
| Функциональность Базы Данных | 1) Созданная БД корректно отображается в облачном сервисе  2) БД правильно работает, а также обрабатывает операции, связанные с хранением информации  3) Данные могут быть добавлены, изменены и удалены из БД |
| **Нефункциональное тестирование** | |
| **Вид тестирования** | **Описание и назначение** |
| Тестирование производительности | Тест необходим для оценки времени создания БД и времени отклика веб-интерфейса при различных уровнях нагрузки  содержит проверку стабильности работы сервиса при различных размерах и регионах размещения,  проверку при создании большого количества баз данных одновременно.  В приоритете объемное, нагрузочное, и тестирование стабильности |
| Тестирование безопасности | Проверка на наличие уязвимостей и потенциальных рисков, связанных с внедрением новой функциональности  Тест включает проверку на SQL-инъекции  проверку прав доступа и аутентификации  проверка используемых методов защиты данных, таких как шифрование  Данное тестирование необходимо для проверки того, что данные пользователей обрабатываются безопасно и доступ к ним ограничен и защищен |
| Конфигурационное тестирование | Проверка поможет убедиться в корректности взаимодействия внедряемой функциональности в различных окружениях  проверка формы создания бд в разных браузерах и их различных версиях(зависит от требований) например, Chrome 114.0.5, Firefox 114.0.1, Safari 14.1.2  проверка адаптивной верстки при разных разрешениях экрана для формы создания бд  развертывание бд |

* Роли и обязанности (Roles and responsibilities);описание ролей и обязанностей каждого члена команды на протяжении всего цикла тестирования новой функциональности:
  + Инженеры тестирования: Создание и выполнение тестовых сценариев, документирование и отслеживание обнаруженных ошибок.
  + Разработчики: Исправление обнаруженных ошибок, участие в процессе тестирования.
  + Менеджеры проекта: Координация процесса тестирования, обеспечение своевременного выполнения задач.
  + Аналитики: Определение требований к новой функциональности, участие в процессе тестирования.
* Требования к окружениям (Environment requirements); окружения, необходимые для более качественного тестирования новой функциональности:
  + Тестовое окружение: Должно быть максимально приближено к реальному окружению, в котором будет работать новая функциональность. Это включает соответствующую версию PostgreSQL, а также другие компоненты системы.
  + Продуктивное окружение: После успешного тестирования новая функциональность должна быть развернута в продуктивном окружении.
  + Окружение для тестирования производительности: Требуемое окружение должно быть способно симулировать высокую нагрузку и большие объемы данных.
  + Окружение для тестирования безопасности: Должно быть способно симулировать различные угрозы безопасности.

1. **Инструменты тестирования (Testing tools):** инструменты, необходимые для проведения тестирования новой функциональности облачного сервиса хранения и обработки информации в базах данных (TMS, багтрекинговая система, стек автоматизации), выбираются в зависимости от проекта и стека;

* Система управления тестированием (TMS): Это может быть инструмент, такой как TestRail или Zephyr, который помогает в управлении, отслеживании и отчетности по тестовым случаям.
* Система отслеживания ошибок (багтрекинговая система): Jira, Bugzilla или Mantis могут быть использованы для отслеживания и управления ошибками, обнаруженными во время тестирования.
* Стек автоматизации: Можно использовать различные инструменты и подходы, в зависимости от того, тестируем ли мы API или веб-интерфейс:
  + стеки для API тестирования:
    - Python(с использованием библиотек Requests, Pytest).
    - фреймворки для тестирования API, такие как Postman, SoapUI, Swagger.
  + стеки для тестирования Веб-интерфейса:
    - Python
    - инструменты и библиотеки для автоматизации, такие как Selenium, Playwright, Cypress.

1. **Отраслевые стандарты, которым необходимо следовать (Industry standards to follow)**: В этом разделе описывается отраслевой стандарт, а также законы, на основании которых услуга должна предоставляться.

* Закон [ФЗ-149](http://docs.cntd.ru/document/901990051) «Об информации, информационных технологиях и о защите информации».
* Закон [ФЗ-152](http://docs.cntd.ru/document/901990046) «О персональных данных». Закон [ФЗ-242](http://docs.cntd.ru/document/420208796), который уточняет ФЗ-152 и обязывает операторов персональных данных обрабатывать и хранить персональные данные россиян с использованием баз данных, размещенных на территории РФ.
* Стандарт безопасности индустрии платежных карт (PCI DSS).
* Генеральный регламент о защите данных (GDPR).
* Стандарт ISO 27001.
* Стандарт ISO 27017.
* Стандарт ISO 27018.
* ГОСТ 57580.1-2017.
* Приказы ФСТЭК России [№17](http://fstec.ru/normotvorcheskaya/akty/53-prikazy/702-prikaz-fstek-rossii-ot-11-fevralya-2013-g-n-17) и [№21](http://fstec.ru/normotvorcheskaya/akty/53-prikazy/691-prikaz-fstek-rossii-ot-18-fevralya-2013-g-n-21).

1. **Критерии начала и окончания тестирования:** описание критериев, требуемых для начала и окончания тестирования новой функциональности.

* Критерии начала тестирования:
  + готовность новой функциональности к тестированию
  + законченность разработки тестового функционала
  + все требования с учетом отраслевых стандартов и законов уточнены
* Критерии окончания тестирования:
  + установленный объем тестирования функциональности выполнен
  + дефекты с высоким уровнем приоритета закрыты и соответствующие регрессионные сценарии успешно проведены
  + оговоренные сроки, отведенные на тестирование, истекли
  + требуемая документация по тестированию, например, отчет о тестировании, подготовлена

1. **Результаты тестирования (Test deliverables)**: документация, которую необходимо создать до, во время и по окончании тестирования новой функциональности;

* До тестирования: Тестовый план, который описывает стратегию, подход, ресурсы, расписание и критерии приемки для тестирования.
* Во время тестирования: Тестовые сценарии и тестовые случаи, которые описывают конкретные условия или переменные для проверки функциональности.
* После тестирования: Отчет о тестировании, который содержит подробную информацию о проведенных тестах, обнаруженных ошибках и рекомендациях.

1. **Метрики тестирования (Testing metrics)**: метрики, которые следует использовать в проекте для анализа статуса проекта и анализа эффективности процесса тестирования новой функциональности;

* Процент выполненных тестов: метрика демонстрирует количество выполненных тестов от общего числа запланированных тестов.
* Passed/Failed Test Cases: метрика рассчитывает отношение удачно пройденных тестов к завершившимся с ошибками
* Open/Closed Bugs: метрика формируется из отношения открытых багов к закрытым.
* Время реакции на ошибки: метрика показывает, сколько времени в среднем требуется для реагирования на обнаруженные ошибки.
* Время исправления ошибок: метрика показывает, сколько времени в среднем требуется для исправления обнаруженных ошибок.

1. **Анализ рисков и способы их снижения (Risk and mitigation):** все риски, возникающие в ходе тестирования и план по их снижению;
2. Продуктовые риски
   * Несанкционированный доступ: если учетные данные для доступа к базе данных становятся доступными для несанкционированных лиц, они могут получить доступ к конфиденциальной информации или даже удалить данные.
     + Меры: тестирование ролей и доступа к базе данных
   * Недостаточное шифрование: если данные не шифруются должным образом, они могут быть прочитаны в случае перехвата.
     + Меры: проверка шифрования чувствительных данных пользователей, например, персональные данные.
   * Скорость и качество разработки;   
     Не проработанные требования повлекут за собой увеличение сроков разработки, тестирования, так как при проектировании не будут учтены все функции.
     + Меры: нужно проанализировать и уточнить требования с учетом отраслевых стандартов и законов. Ниже представлен список.
       1. Доступ к форме создания базы данных
          1. Какой тип авторизации используется для доступа к форме создания базы данных?
       2. Конфигурация для создания базы данных:
          1. Есть ли возможность настроить кластер в котором будет находиться база данных? Имя кластера? Тип физического процессора и набор допустимых конфигураций vCPU и RAM? Какие опции конфигурации доступны для выбора при создании базы данных, например, версия, настройки СУБД? Тип хранилища(на локальных SSD-дисках, на сетевых SSD и HDD-дисках и т.д)?
          2. Какие варианты подключения к БД(сети общего пользования (публичный интернет), защищенное VPN-подключение поверх интернета (IPSec VPN), защищенный L2 VPN-канал связи, сеть IP VPN, через внутреннюю сеть)?
          3. Какие требования к имени базы данных? В системе, для которой разрабатывается новая функциональность, могут быть дополнительные ограничения, которые стоит учесть. Допустимая длина названия - от 1 до 63 символов по спецификации Postgre, название может содержать строчные и прописные буквы латинского алфавита, цифры, нижние подчеркивания и дефисы.
          4. Какие регионы размещения доступны для выбора при создании базы данных?
          5. Какие размеры базы данных доступны?
          6. Какие инструменты управления базой данных доступны для использования, например, консоль управления, графический интерфейс? Мониторинг доступности БД? Мониторинг состояния БД? Резервное копирование?
          7. Какие опции безопасности доступны для выбора при создании базы данных, например, шифрование данных, доступ к базе данных?
          8. Какие статусы может иметь база данных после создания в интерфейсе? Что увидит пользователь после отправки формы?
          9. В случае появления ошибок при создании базы данных что будет показано пользователю?
       3. Оплата за использование базы данных:
          1. Это будет платная услуга? Постоплата или предоплата?
3. Организационные риски
   * Несоответствие законодательству:
     + Меры: проведение внутреннего и внешнего аудита
   * Неверная оценка трудозатрат на разработку:
     + Меры: учет времени всех работ для завершения функциональности(дизайн, аналитика, тестирование, развертывание проекта девопсами)
   * Увольнения, отпуска и болезни сотрудников:
     + Меры: организация команды без bus-фактора, найм сотрудников
   * Проблемы с квалификацией сотрудников:
     + Меры: инструктаж, подготовка подробных документаций, определение необходимых компетенций для реализации новой функциональности и отбор по ним.
   * Нестабильность окружения:
     + Меры: подготовка и тестирование окружения до начала разработки
4. **Инструмент отчетности (Reporting tool):** инструмент отчетности для облачного сервиса хранения и обработки информации в базах данных с новой функциональностью развертывания базы данных PostgreSQL;

* Dashboard управления: управленческий дашборд может предоставлять обзор текущего состояния базы данных, включая использование ресурсов, производительность и статус безопасности. Он также может включать предупреждения о любых текущих или потенциальных проблемах.
* Система отслеживания ошибок: Система отслеживания ошибок может автоматически отслеживать и документировать любые ошибки или проблемы, возникающие в базе данных. Она может включать в себя информацию о том, когда и где произошла ошибка, какие действия были предприняты для ее устранения, и какие шаги следует предпринять для предотвращения повторения ошибки в будущем.
* Отчеты о производительности: Отчеты о производительности могут анализировать использование ресурсов базой данных и как это влияет на ее производительность. Это может помочь в определении областей, требующих оптимизации, и в планировании будущего масштабирования.

1. **Заключение.**

Так как документ о стратегии тестирования это статический документ и не должен меняться на протяжении жизненного цикла проекта, для финальной версии данной тест-стратегии необходимо уточнение требований.

В текущей версии документа описаны возможные процедуры обеспечения качества надежности и безопасности системы вместе с новой функциональностью, которая позволяет создать базу данных в облаке, но по мере уточнения требований список процедур и проверок должен быть скорректирован.