**Практическое задание №1**

**Процесс сопровождения в жизненном цикле разработки**

**информационной системы.**

1. **Анализ текущего состояния:** Важным этапом анализа текущего состояния информационной системы является оценка ее документации, комментариев в коде, стабильности и безопасности.  
   * *Документация:* Проверьте наличие и актуальность документации. Оцените полноту и понятность инструкций по установке, настройке и сопровождению системы. Обратите внимание на описание архитектуры, модулей и API.
   * *Комментарии в коде:* Анализируйте наличие комментариев в исходном коде. Отсутствие комментариев или их недостаточное количество может затруднить понимание логики работы системы.
   * *Стабильность:* Оцените стабильность системы, выявив частоту и причины сбоев. Исследуйте журналы ошибок и мониторинг производительности.
   * *Безопасность:* Проверьте соответствие системы современным стандартам безопасности. Анализируйте наличие обновлений, мер безопасности и механизмов защиты от внешних атак.
2. **Выявление областей для улучшения:**
   * *Улучшение документации:* Если документация оказалась неполной или устаревшей, необходимо обновить её, добавив новые инструкции, описания и схемы. Это улучшит понимание системы и упростит процессы сопровождения.
   * *Уровень комментариев в коде:* В случае отсутствия комментариев или их недостаточного количества, рекомендуется добавить пояснения к коду. Это поможет новым разработчикам быстрее входить в проект и уменьшит вероятность ошибок.
   * *Обновление безопасности:* Если система уязвима к угрозам безопасности, необходимо провести аудит безопасности и внедрить соответствующие меры (например, обновления, патчи, усиление авторизации).
3. **Внесение изменений:**
   * *Добавление дополнительной документации:* Разработайте дополнительные материалы, такие как руководства пользователя, описание API, схемы базы данных. Это поможет как конечным пользователям, так и разработчикам.
   * *Улучшение механизма отслеживания изменений:* Внедрите систему управления версиями (например, Git) и установите процессы ревизии кода. Это обеспечит более эффективное отслеживание изменений и контроль качества кода.
   * *Обновление безопасности:* Регулярно обновляйте зависимости и библиотеки, а также внедряйте механизмы защиты от известных угроз.
4. **Тестирование изменений:**
   * *Тестовый план:* Разработайте тестовый план, включающий тест-кейсы для проверки обновленной документации, новых механизмов отслеживания изменений и обновлений безопасности.
   * *Проведение тестирования:* Перед внедрением изменений проведите тестирование в соответствии с разработанным планом. Это может включать функциональное тестирование, тестирование производительности и тестирование безопасности.
   * *Оценка результатов:* Анализируйте результаты тестирования и убедитесь, что внесенные изменения не привели к новым проблемам. При необходимости внесите коррективы и повторите тестирование.

### **Отчет:**

#### **Обзор текущего состояния проекта:**

Подробный обзор текущего состояния проекта, включая особенности документации, стабильности и безопасности.

#### **Обоснование выбранных областей для улучшения:**

Обоснование выбора трех ключевых областей для улучшения, учитывая потребности пользователей и риски.

#### **Предложения по внесению изменений:**

Предложения по внесению конкретных изменений в каждую из выбранных областей с описанием ожидаемых выгод.

#### **Описание проведенного тестирования с результатами и рекомендациями:**

Подробное описание тестов, результатов, выявленных проблем и рекомендации по дальнейшим шагам.