**Задание №4: Разработка тест плана к программному продукту.**

**Вариант 1: Учет больничных листов на предприятии**

**Сергиенко Лев, 12 группа, МСС**

### **1. Идентификатор тестового плана**

* **Название документа:** Тестовый план для системы учёта больничных листов
* **Версия:** 1.0
* **Дата создания:** Март 2025

### **2. Ссылки**

* **Спецификация требований (SRS):** Документ «Анализ предметной области и требований к программному продукту» (Задание №3)
* **Документы проекта:** Технические задания, диаграммы потоков данных, инструкции по интеграции.

### **3. Введение**

Цель данного тестового плана – описать стратегию, методы и процедуры тестирования для системы учёта больничных листов. План определяет объем тестирования, перечень объектов, критерии прохождения тестов, риски, необходимые ресурсы и порядок согласования результатов. Тестирование направлено на обеспечение корректной регистрации больничных листов, расчёта пособий, формирования отчетности и интеграции с внешними системами, а также соблюдения нормативных требований.

### **4. Объекты тестирования**

* **Модуль регистрации больничных листов:** Ввод и сохранение данных (ФИО, табельный номер, даты, данные медицинской организации и т.д.).
* **Модуль валидации данных:** Проверка корректности заполнения обязательных полей, форматов дат, уникальности номера документа и т.п.
* **Модуль расчёта пособий:** Автоматизированный расчет суммы пособия с учетом среднего заработка, страхового стажа и длительности нетрудоспособности.

### **5. Проблемы и риски**

* **Технические риски:** Возможные ошибки в обработке данных, проблемы с интеграцией и синхронизацией с внешними системами.
* **Валидация данных:** Риск некорректного ввода данных пользователями, что может привести к ошибкам при расчётах и формировании отчетов.
* **Зависимости от внешних систем:** Нарушение работы интеграционных каналов может повлиять на корректность работы продукта.

### **6. Функции, которые нужно протестировать**

* **Регистрация и учет больничных листов:** Корректность ввода, сохранения, редактирования, продления, аннулирования и дублирования записей.
* **Валидация обязательных полей:** Проверка корректности форматов дат, уникальности номера, заполненности обязательных полей.
* **Модуль расчёта пособий:** Точность вычислений с учетом различных сценариев.
* **Формирование и экспорт отчетности:** Корректность формирования сводных отчетов, возможность фильтрации и сортировки данных, экспорт в заданных форматах.

### **7. Функции, которые НЕ нужно тестировать**

* **Внутренние функции бухгалтерского учета:** Тестирование проводок в бухгалтерской системе осуществляется отдельным модулем и не входит в сферу данного тест плана.
* **Интерфейсы сторонних систем:** Тестирование работы сторонних систем (например, ERP/HRM) проводится их специализированными командами; наша задача – проверить корректность обмена данными.
* **Нефункциональные аспекты, не связанные с безопасностью и производительностью:** Если такие аспекты уже протестированы в рамках общих нагрузочных испытаний.

### **8. Подходы**

* **Функциональное тестирование:** Проверка каждого модуля системы на соответствие требованиям.
* **Модульное тестирование:** Тестирование отдельных компонентов.
* **Интеграционное тестирование:** Проверка корректности обмена данными между модулями и внешними системами.
* **Системное тестирование:** Оценка работы всей системы в условиях, приближенных к боевым.
* **Приемочное тестирование:** Совместное тестирование с участием конечных пользователей для подтверждения соответствия бизнес-требованиям.
* **Автоматизированное тестирование:** Использование скриптов для повторяемых тестов.

### **9. Критерии прохождения тестов для объектов тестирования**

* **Функциональные тесты:** Все тест-кейсы должны быть выполнены успешно (статус «Pass»).
* **Критерии корректности:** Все рассчитанные суммы пособий должны соответствовать расчетным алгоритмам, подтвержденным в SRS.
* **Отсутствие критических ошибок:** Не допускается наличие дефектов, блокирующих ключевые функции.
* **Соответствие нормативным требованиям:** Все валидационные проверки и форматы данных должны соответствовать государственным стандартам.

### **10. Критерии остановки и требования для возобновления тестирования**

* **Критерии остановки:**
  + Обнаружение критической ошибки, влияющей на возможность регистрации больничных листов или расчёта пособий.
  + Нарушение работы интеграционных модулей, что делает невозможным проведение дальнейших тестов.
* **Требования для возобновления:**
  + Исправление выявленных критических ошибок и повторное прохождение предварительных проверок.
  + Подтверждение корректной работы системы на тестовом стенде до продолжения полного цикла тестирования.

### **11. Результаты тестирования**

* **Формат отчёта:** Отчёт включает перечень выполненных тест-кейсов, статус каждого теста, найденные дефекты с классификацией по критичности, статистику (количество пройденных/не пройденных тестов).
* **Инструменты:** Результаты фиксируются в системе баг-трекинга (например, JIRA, Bugzilla) и документируются в итоговом отчёте тестирования.

### **12. Оставшиеся задачи тестирования**

* **Нагрузочное тестирование:** Планируется проведение тестирования под высокой нагрузкой для оценки производительности системы.
* **Регрессионное тестирование:** После исправления ошибок – повторная проверка основных сценариев.
* **Тестирование интеграции с внешними системами:** Дополнительное тестирование в случае обновления внешних модулей или изменения протоколов обмена данными.

### **13. Требования среды**

* **Аппаратное обеспечение:** Тестовые серверы с конфигурацией, аналогичной боевой среде.
* **Программное обеспечение:** Операционные системы (Windows/Linux), СУБД, веб-серверы, браузеры (для UI-тестирования).
* **Инструменты тестирования:** Автоматизированные фреймворки (Selenium для UI, Postman для API, JMeter для нагрузочного тестирования), баг-трекинг система.

### **14. Требования по части кадров и их обучения**

* **Квалификация тестировщиков:** Опыт работы в тестировании информационных систем, понимание принципов автоматизации и методик функционального тестирования.
* **Обучение:** Проведение инструктажа по специфике системы учёта больничных листов, особенностям расчетов пособий и интеграции с внешними системами.
* **Поддержка:** Наличие методических материалов и документации по продукту.

### **15. Распределение обязанностей**

| **Имя** | **Должность** | **Обязанности** |
| --- | --- | --- |
| Сергиенко Л.Э. | Руководитель QA | Общее руководство тестированием, планирование. |
| Сергиенко Л.Э. | Инженер по тестированию | Разработка и выполнение тест-кейсов. |
| Сергиенко Л.Э. | Автоматизатор тестирования | Создание и поддержка автоматизированных тестов. |
| Сергиенко Л.Э. | Аналитик по качеству | Анализ результатов тестирования, составление отчётов. |

### **16. Расписание**

* **Подготовительный этап:** Сбор и анализ документации, настройка тестовой среды (1–2 недели).
* **Разработка тест-кейсов и скриптов:** 1 неделя.
* **Функциональное и интеграционное тестирование:** 2–3 недели.
* **Системное и приемочное тестирование:** 1–2 недели.
* **Итоговый отчёт и анализ:** 3–5 рабочих дней.

### **17. Планирование рисков и непредвиденных обстоятельств**

* **Идентификация рисков:** Регулярное проведение встреч для обсуждения возникших проблем и выявления рисков на ранних стадиях.
* **План реагирования:** В случае выявления критических ошибок – немедленное уведомление ответственных лиц, приостановка тестирования и фиксация дефектов в баг-трекере.
* **Планирование ресурсов:** Дополнительное привлечение специалистов и перераспределение задач в случае задержек или форс-мажорных обстоятельств.

### **18. Утверждение**

* **Разработчик тестового плана:** Сергиенко Л.Э., Руководитель QA

### **19. Глоссарий**

* **Больничный лист:** Официальный документ, подтверждающий временную нетрудоспособность сотрудника.
* **SRS:** Спецификация требований к программному продукту.
* **Регрессионное тестирование:** Проверка ранее протестированных функций после внесения изменений.
* **Smoke-тестирование:** Быстрая проверка основных функций системы для определения готовности к дальнейшему тестированию.
* **Интеграционное тестирование:** Проверка взаимодействия между отдельными модулями или системами.