Описание процессов жизненного цикла программного обеспечения «Справочно-информационная система Абиограм»

Аннотация

Настоящий документ является описанием процессов жизненного цикла программного обеспечения (ПО) «Справочно-информационная система Абиограм» и содержит сведения о жизненном цикле программного обеспечения, в том числе устранение неисправностей, выявленных в ходе эксплуатации программного обеспечения, а также информацию о персонале, необходимом для обеспечения такой поддержки.

Оглавление

Сокращения и термины

ЛИС	лабораторная информационная система
МИС	медицинская информационная система
Git	распределённая система управления версиями и управления исходным кодом
Гитлаб флоу	методология работы с Git, в ней определяется, какие виды веток необходимы проекту и как выполнять слияние между ними.
Бэклог	перечень рабочих задач, расположенных в порядке важности, для разработчиков
Релиз	конечная стадия разработки программного обеспечения
Smoke-тестирование	проверка программного обеспечения на стабильность и наличие явных ошибок
Регрессионное тестирование	проверка ранее протестированной программы, позволяющая убедиться, что внесенные изменения не повлекли за собой появления дефектов в неизмененной части программы

Введение

Ключевая идея программного обеспечения «Справочно-информационная система Абиограм» заключается в интеграции с существующими лабораторными информационными системами (ЛИС) и/или медицинскими информационными системами (МИС) и микробиологическими анализаторами. Также возможно прямое взаимодействие медицинского персонала с интерфейсом ПО. Итогом работы программного обеспечения является формирование микробиологического заключения с оценкой результатов на основе установленных правил.

Программное обеспечение использует следующую информацию:

- Вид возбудителя;
- Внутренний номер возбудителя;
- Результаты определения чувствительности к антибиотикам для данного возбудителя;
- Используемые критерии для определения чувствительности;
- Идентификатор пациента.

Используя полученную информацию ПО проводит интерпретацию результатов на основе установленных критериев в соответствующих документах и осуществляет валидацию результата с применением установленных правил. Результатом работы ПО является сформированное микробиологическое заключение, которое отправляется в ЛИС/МИС,

или в виде графического отображение представляется персоналу (в случае непосредственного использования веб-интерфейса ПО).

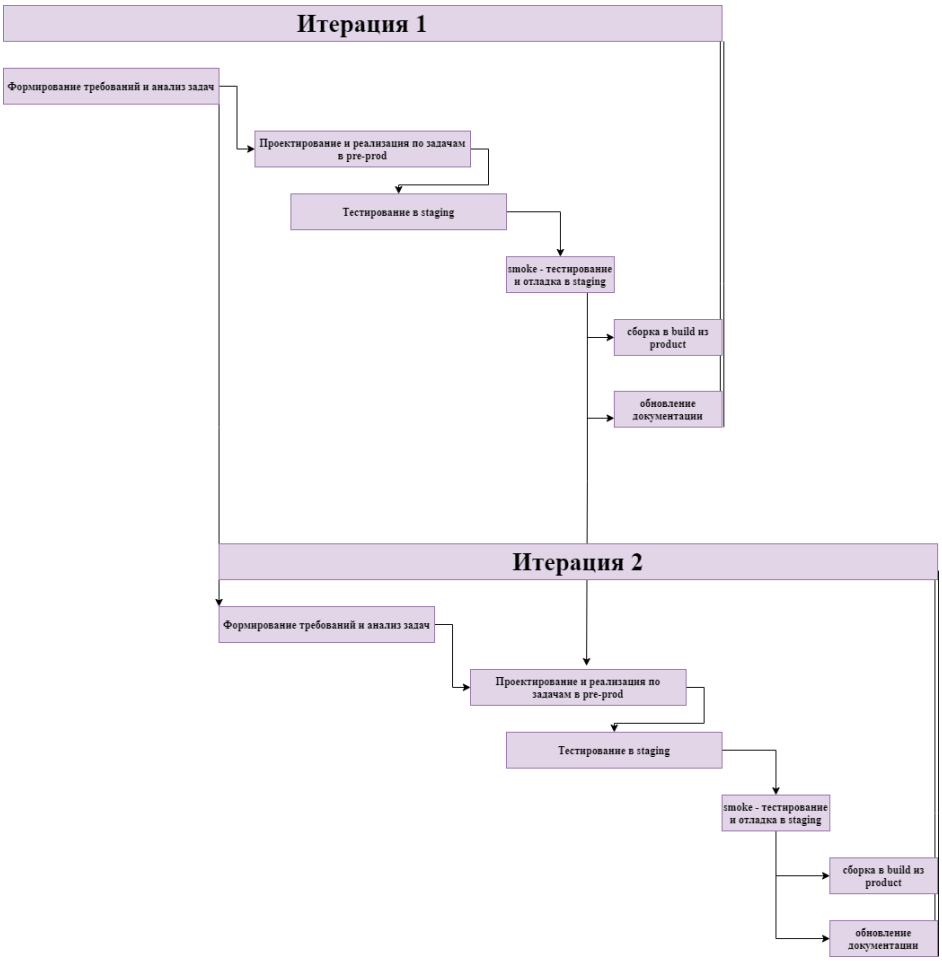
1. Планирование процессов жизненного цикла разработки

Жизненный цикл (ЖЦ) включает период создания и использования «Справочно-информационная система Абиограм», начиная с момента возникновения потребности в продукте, заканчивая
разработкой, тестированием и отладкой, поставкой программной продукции, ее эксплуатацией на объектах Заказчика и технической поддержкой.

Жизненный цикл определен с учетом положений следующих стандартов:

- ГОСТ Р ИСО/МЭК 12207-2010 «Информационная технология. Системная и программная инженерия. Процессы жизненного цикла программных средств»;
- ГОСТ Р 56939-2016 «Защита информации. Разработка безопасного программного обеспечения. Общие требования»;
- ГОСТ Р ИСО 9001-2015 «Системы менеджмента качества. Требования».

Используется итерационная модель жизненного цикла (рисунок 1).



Выпуск новой версии продукта ведется в среднем каждые две недели согласно итерационной модели. Итерация включает этапы:

- формирования требований и оценку задач;
- проектирование и реализацию по задачам;
- тестирование и отладку;
- smoke-тестирование;
- обновление документации.

Процессы управления конфигурацией сервиса проводятся с использованием репозитория дистрибутивов, сервера сборки и системы контроля версий.

2. Формирование требований и анализ задач

В рамках формирования требований и анализа задач выполняется следующий порядок задач:

- аналитик определяет задачи и требования к задачам, которые следует выполнить в итерацию;

- технический менеджер, аналитик совместно с разработчиком проводят анализ задач;
- технический менеджер определяет ответственность за разработку, назначает задачи и сроки выполнения разработчику;
- технический менеджер проводит мониторинг процесса.

Управление задачами ведётся с использованием релизов.

Не включенные в релиз задачи остаются в бэклоге и не участвуют в разработке, а используются для планирования будущих работ.
Результаты этапа:

1. Сформирован бэклог на итерацию и релиз;
2. Составлена спецификация требований и/или техническое описание по каждой задаче;
3. Назначены задачи и сроки выполнения.

3. Проектирование и реализация

В рамках проектирования и реализации выполняется следующий порядок задач:

- разработка программной архитектуры и решений по построению всех составных компонент;
- разработка исходных текстов, написание файлов спецификации для сборки программного обеспечения;
- сборка программного обеспечения и добавление в репозиторий программного обеспечения;
- сборка дистрибутивов из репозитория программного обеспечения.

Разработка по задачам идёт до тех пор, пока все задачи в рамках итерации не будут закрыты.

При работе с кодом команда придерживается методологии, известной как гитлаб флоу (https://about.gitlab.com/topics/version-control/what-is-gitlab-flow/).
В каждом репозитории есть три основных ветки и соответствующих им окружения:

- Мастер (с англ. Master, в терминологии гитлаб флоу) - ветка последнего стабильного
  релиза;
- Пре-продакшн (с англ. pre-production , в терминологии гитлаб флоу) - ветка для разработки, отправки на тестирование и готовящегося релиза;
- Продакшн (с англ. production, в терминологии гитлаб флоу) - ветка окончательного релиза.

4. Тестирование и отладка

Выполнение тестирования является обязательным перед передачей новой версии потребителю. Тестирование проводится лицом, ответственным за проведение тестирования программной продукции (инженер-тестировщик). Для тестирования и отладки продукции выделяется сервер, выдается задание на тестирование. По результатам тестирования осуществляется устранение ошибок и осуществляется (при необходимости) доработка программного обеспечения.
Процессом тестирования и отладки определен следующий порядок:
● сборка дистрибутивов программного обеспечения - ответственный инженер по внедрению проводит сборку в pre-production окружении;
● проведение тестирования программного обеспечения - ответственный инженер-тестировщик проводит тестирование в staging, а также smoke-тестирование в staging всего продукта;
● устранение выявленных недостатков программного обеспечения - в случае обнаружения недостатков инженер-тестировщик формирует задачу с описанием дефекта (недостатка ПО), далее проводится повторное smoke-тестирование (ретест);
● При завершении smoke-тестирования проводится регрессионное тестирование всей функциональности продукта;
● При положительном результате регрессионного тестирования проводится добавление в репозиторий эталонных версий дистрибутивов и исходных текстов программного обеспечения;
● ответственный инженер проводит слияние pre-production в master и сборку (building) в production окружении;
● корректировка программной документации.

5. Эксплуатация и сопровождение

5.1. Техническая поддержка

Техническая поддержка пользователей осуществляется в формате консультирования пользователей и администраторов сервиса по вопросам установки, переустановки, администрирования и эксплуатации программного обеспечения по электронной почте support@abiogram.ru.

В рамках технической поддержки сервиса оказываются услуги:
● помощь в настройке и администрировании программного обеспечения;
● помощь в установке обновлений программного обеспечения;
● помощь в поиске и устранении проблем в случае некорректной установки обновления программного обеспечения;
● описание функционала программного обеспечения, помощь в эксплуатации;
● предоставление актуальной документации по настройке/работе программного обеспечения.

В заявке на техническую поддержку пользователь должен указать следующую информацию:
● описание проблемы;
● предпринятые попытки решения проблемы;
● релевантная дополнительная информация.
Завершенный запрос переходит в состояние закрытого после получения подтверждения от пользователя о решении запроса. В случае отсутствия ответа
пользователя о завершении запроса в течение 14 рабочих дней, в случае если иное не оговорено в соглашении о расширенной технической поддержки, запрос считается закрытым. Закрытие запроса может инициировать пользователь, если необходимость в ответе на запрос по каким-либо причинам более не требуется.

5.2. Персонал, обеспечивающий работу на местах

Пользователи ПО «Справочно-информационная система Абиограм» должны обладать навыками работы с персональным компьютером на уровне пользователя. Для работы с Системой пользователю необходимо изучить руководство пользователя «Справочно-информационная система Абиограм». Администратор Системы должен владеть навыками работы с персональным компьютером на уровне уверенного пользователя. Обязательно знание основ работы вычислительной техники и программного обеспечения в локальных сетях, а также настроек системной политики прав пользователей в операционных системах семейства Windows и Linux.

5.3. Персонал, обеспечивающий техническую поддержку и развитие системы

Специалисты, обеспечивающие техническую поддержку и развитие Системы, должны обладать следующими знаниями и навыками:

1. Владение персональным компьютером на уровне уверенного пользователя;
2. Знание функциональных возможностей Системы и особенностей работы с ними;
3. Знание языков программирования;
4. Знание реляционных БД;
5. Значение протоколов обмена данных;
6. Знание средств восстановления баз данных и мониторинга производительности серверов.