

РЕАЛИЗАЦИЯ СИСТЕМЫ АВТОМАТИЗАЦИИ УПРАВЛЕНИЯ ПРОЕКТАМИ С ИСПОЛЬЗОВАНИЕМ AGILE-ТЕХНОЛОГИЙ

Степанов А.А., магистрант,
Редькин Ю.В., к.т.н., доцент
ФГБОУ ВО «ГМУ им. адм. Ф.Ф. Ушакова»,
г. Новороссийск, Россия

Аннотация. В условиях нестабильной сетевой связи на морских судах требуется обеспечить отказоустойчивость и автоматизацию управления проектами на судне в автономном плавании. С этой целью в работе рассмотрена система автоматизации управления проектами, в основе которой лежит двухузловая структура с применением Agile-технологий. Проанализированы существующие методы управления проектами, прототипирование с использованием прокси-сервиса, доставки обновлений с помощью патчей. Предложена и обоснована система, которая позволяет устранить единую точку отказа структуры, снижение времени обновления, а также интеграцию Open Project и GitHub с целью прозрачного отслеживания хода разработки.

Ключевые слова: Open Project, отказоустойчивость, Agile, CI/CD, прокси-сервис.

В системах управления проектами с выраженной проблемой доступа к сети, характерной для морских судов, стандартные и привычные подходы к разработке и обновления программного обеспечения сталкиваются со множеством ограничений. Например, централизованная архитектура с единым сервером создает единую точку отказа, а обновление в ручном режиме приводит к увеличению времени простоя и риску рассинхронизации данных. Инструменты управления проектами на основе облачных технологий, такие как Jira, в сильной степени зависят от постоянного доступа к облачной

инфраструктуре, что делает невозможным их применения в условиях неустойчивой связи [6].

Вследствие этого, для морских судов, находящиеся длительное время в автономном плавании, требуется разработка специальной системы, обеспечивающих гибкое управление проектами. Такая система должна иметь отказоустойчивую аппаратную архитектуру и включать в себя современные и проверенные инструменты управления проектами, а также адаптированные механизмы доставки обновления [3, 4]. Важным условием является то, что все компоненты системы большую часть времени должны функционировать в изолированной среде без постоянного доступа к ресурсам береговых центров.

Для решения данной проблемы, предлагается использовать двухузловую серверную архитектуру системы с резервированием [7]. В этой архитектуре один сервер используется в качестве основного узла, а второй является резервным, поддерживающим актуальное состояние путем синхронизации. На рис. 1 представлена общая архитектура предлагаемой системы, где показано взаимодействие клиентских приложений, двух серверных узлов и прокси-сервис.

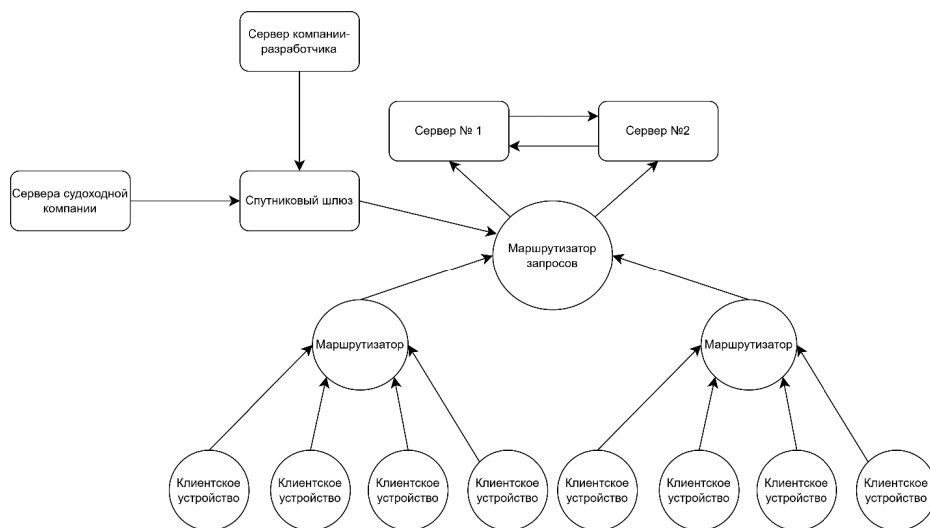


Рис. 1. Обобщенная двухсерверная архитектура системы управления проектами в изолированной среде

Одной из основных частей предлагаемой системы является прокси-сервис, разработанный на Python, который выполняет функции маршрутизации

сетевого трафика и обеспечивает отказоустойчивость и баланс нагрузки на систему. Сервис периодически проверяет доступность основного и резервного серверов: если один из серверов оказывается недоступен, происходит переключение на резервный сервер. Благодаря прокси-сервису обеспечивается повторная отправка запросов в условиях нестабильной связи и адаптация к поведению системы. Основные функции прокси-сервиса системы представлены на рис. 2.

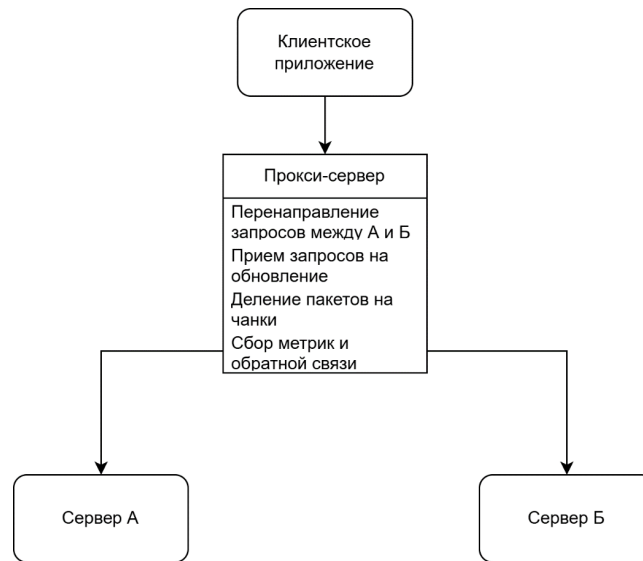


Рис. 2. Функции прокси-сервиса системы

Для доставки обновления и патчей был разработан механизм, позволяющий передавать только небольшие изменения вместо полных бинарных сборок. С его помощью можно выполнять сценарии при изменении состояния репозитория: если вносятся изменения и отправляются на выделенную ветку, формируется патч-файл, который содержит описание изменений [5]. Полученный таким образом файл, сжатый стандартными алгоритмами, имеет намного меньший размер по сравнению с полной сборкой. Затем патч доставляется на узел, где применяется к локальной базе, далее выполняется сборка и развертывание.

Чтобы управлять проектами и отслеживать работу со стороны заказчика, в системе используется платформа Open Project. Она отличается поддержкой гибких методологий Scrum и Kanban, возможностью локального

развертывания, а также наличие удобного программного интерфейса с возможностью интеграции в другие проекты [1, 2]. Благодаря интеграции с GitHub можно сгенерировать API-токен и затем добавить в репозиторий вебхук. Когда создается задача в Open Project, разработчиком при оформлении запроса на слияние указывается соответствующий тег и номер задачи, далее вебхук фиксирует это событие и на странице задачи в Open Project отображаются внесенные изменения. Процесс синхронизации изменений между Open Project и GitHub, а также графическое отображение работы над программным обеспечением представлены на рис. 3 и рис. 4 соответственно.

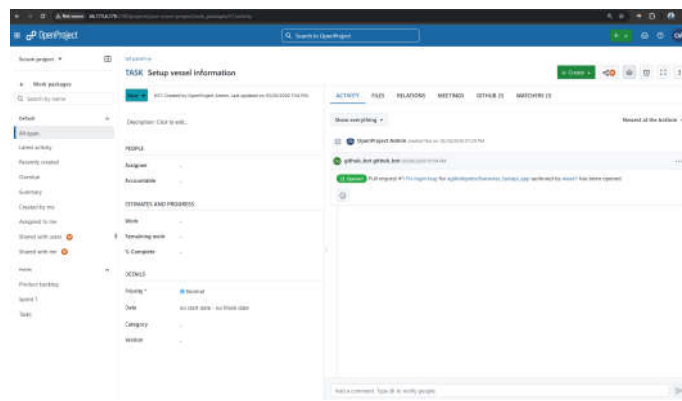


Рис. 3. Синхронизация изменений между Open Project и GitHub

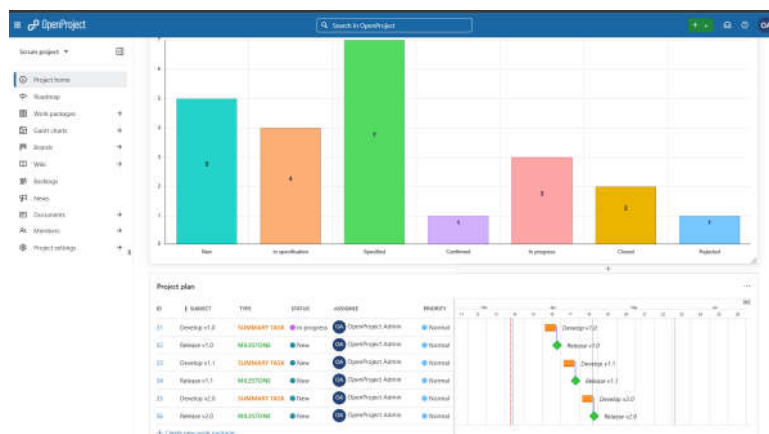


Рис. 4. Графическое отображение выполнения задач в системе

Таким образом, разработанная система позволяет решить ряд проблем систем управления проектами в условиях неустойчивой связи с глобальной сетью. Так, предлагаемая система с двумя узлами, прокси-сервисом для автоматического переключения и доступный сервер позволяет убрать единую

точку отказа, которая обычно присутствует в централизованных архитектурах. Благодаря применению небольших патч-файлов обновления появляется возможность сократить объем передаваемых данных, что критически важно в условиях неустойчивой или недоступной связи с береговым центром. Интеграция Open Project и GitHub позволяет отслеживать ход работы над задачами в реальном времени, а автоматизированные отчеты и понятные дашборды делают систему проще, понятнее и доступнее для пользователей, имеющих разный уровень знаний и навыков. В заключение следует отметить, что данная система является готовым апробированным решением, пригодным для внедрения в судовых условиях, которое может быть также масштабировано на другие промышленные объекты, работающие в условиях неустойчивой связи.

Литература

1. Джефф Сазерленд. Scrum. Революционный метод управления проектами. М.: МИФ, 2025. 288 с.
2. Дэвид Андерсон. Канбан. Альтернативный путь в Agile. М.: МИФ, 2017. 336 с.
3. Кен Швабер. Скрам. Гибкое управление продуктом и бизнесом. М.: Альпина Паблишер, 2024. 236 с.
4. Коул Р. Блистательный Agile. Гибкое управление проектами с помощью Agile, Scrum и Kanban. Санкт-Петербург : Питер, 2019. 304 с.
5. Ленц Мориц. Python. Непрерывная интеграция и доставка. М.: ДМК Пресс, 2020. 168 с.
6. Redkin Y.V. About the problems of data transmission over wireless industrial networks // Modern informatization problems in the technological and telecommunication systems analysis and synthesis (MIP-2020'AS): Proceedings of the XXV-th International Open Science Conference. Yelm, WA, USA: Science Book Publishing House, 2020. P. 260-265

7. Redkin Y. V. Redundancy features of real-time maritime information systems based on Ethernet technology // Modern informatization problems in the technological and telecommunication systems analysis and synthesis : Proceedings of the XX-th International Open Science Conference. Yelm, WA, USA: Science Book Publishing House LLC, 2015. P. 376-382.