

**Секція 10. СТРАТЕГІЯ СТІЙКОГО РОЗВИТКУ ПІДПРИЄМСТВ,
ГАЛУЗЕЙ, КОМПЛЕКСІВ**

*Колодинський Сергій Борисович,
доктор економічних наук, професор
Агеєнко Антоніна Сергіївна,
студентка 4 курсу*

Одеська державна академія будівництва та архітектури

**ПРОЕКТ ФУНКЦІОНУВАННЯ ТА ІМПЛЕМЕНТАЦІЇ МЕХАНІЗМІВ
АВТОМАТИЗАЦІЇ ТЕСТУВАННЯ В ПРОЕКТНІЙ ОРГАНІЗАЦІЇ**

На сьогодні багато уваги слід приділяти саме управлінню ризиками якості, людського фактору та автоматизації тестування. Програмні продукти проектів мають схильність до регресії. Зміна чи нарощування функціоналу в одній частині продукту проекту може призвести до збоїв у інших частинах, які вже були протестовані, та удосконалені раніше. Залежно від комплексності системи обсяг регресійних дефектів може сягати 35% від загального. Тому тестувати одне й те саме протягом циклу розробки проекту потрібно декілька разів. Регресійне тестування є одним із найдорожчих заходів відділу якості. Також слід пам'ятати, що інженери з контролю якості, як будь-яка людина піддаються «людському фактору». Тобто інколи люди помиляються. На сьогодні в умовах стрімких змін, невизначеності і нестабільності розвитку багатьох галузей, тому все більшого значення набуває розвиток методів управління гнучкими, адаптивними проектами в контексті м'якого компонента проекту. М'який компонент проекту – людина, фахівець [1].

Помилка при тестуванні – потенційний збиток в проекті. Тому зростають вимоги до рівня кваліфікації інженерів саме із контролю якості. Автоматизація тестування є ключовим елементом вчасного виконання проекту на базі гнучкої, адаптивної моделі. Бажано автоматизувати нову функціональність у тому ж спринті у яку вона розробляється, та використовувати процес безперервної інтеграції та випуску (англ. Continuous Integration / Continuous Delivery).

Цей процес забезпечує щоденну збірку продукту з урахуванням усіх останніх змін, автоматичне тестування та звітування у неробочий час. Тобто кожного дня команда проекту має у своєму розпорядженні актуальну у даний момент часу версію продукту проекту, та інформацію про загальний статус якості цієї версії. Вартість регресійного тестування мінімізується. Програмісти значно скоріше отримують зворотній зв'язок з приводу регресії. Регресійні дефекти виправляються у тому – ж спринті. Так само мінімізується вартість планування та управління стабілізацією продукту проекту. Структурна схема процесу безперервної інтеграції та випуску надана на рисунку 1.

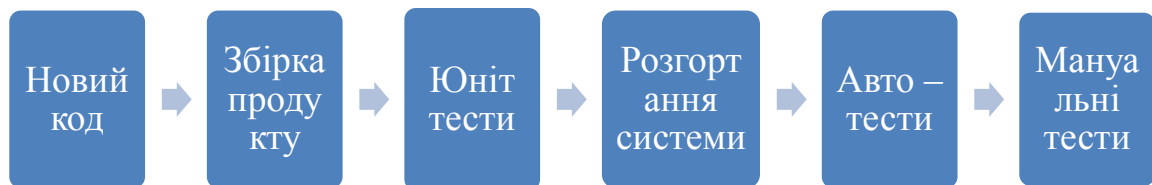


Рисунок 1. – Continuous Integration / Continuous Delivery

Кожний з етапів проекту, окрім першого, із зазначених на рис. 1. генерує зворотній зв'язок з приводу якості системи. Будь яка несправність, знайдена у процесі призведе до зміни програмного коду, потім до нової збірки, запуску «юніт» тестів і так далі. Гнучкі методи займають лідируючі позиції у сучасній індустрії виробництва програмних продуктів. Значна більшість дискретних проєктів, які мають бути почати з нуля, та не орієнтуються на багаторічний розвиток та глобальні метаморфози продукту проекту може бути виконана на базі гнучкої методології управління проєктів [2].

Проте у деяких випадках використання гнучкого підходу є не оптимальним, або не можливим. Зазвичай це гібридні проєкти – «довгожителі» із великим процентом операційної складової у процесах, пов'язані із підтримкою та розвитком сталого бренду на ринку програмного забезпечення. Такі проєкти часто мають досить розподілені проєктні команди із відчутною різницею у культурі та часовому поясі, велику спадщину застарілого функціоналу розробленого за допомогою архаїчних технологій, як слід низький

рівень автоматизації тестування. Адже кожна технологія розробки програмного забезпечення, якщо провести аналогію із машинобудуванням, є окремою «конвеєрною лінією». Кожна «лінія» потребує компліментарний інструментарій автоматизації проекту.

Основаючись на характеристиці наданої вище означимо проблематику використання Scrum у даному випадку. Істотна різниця у часових поясах розташування проектних команд лімітує комунікаційні можливості, ускладнює використання щоденних зустрічей – ключового управлінського інструменту. Також на обмін інформацією негативно впливає мовний бар'єр, великий обсяг виробничої групи та культурні відмінності між локаціями. Ще один фактор впливу часового поясу – швидкість обробки зворотного зв'язку отриманого від тестів. Часто на момент знайдення проблеми відповідальний інженер вже закінчив робочий день. Низький обсяг покриття автоматичного тестування та великий обсяг регресійного тестування нівелює концепцію спринт проекту [2, 3].

Оптимізація процесів (прийнятність) у такому проекті можлива за допомогою моделі виробництва із третьої групи – процесно – орієнтованих із фокусом на продукт проекту, додаючи специфічні до конкретного випадку вдосконалення продукту проекту.

Друга група моделей базується на ідеї розглядання комплексного проекту, як програми проектів меншого обсягу. Виділення під – продуктів у рамках продукту проекту, та виконання кожного з них за допомогою найбільш ефективної, прибуткової моделі. Зазвичай рекомендується виділити наступні під – проекти:

- автоматизації регресійного контенту; розробки нових функцій проекту;
- модернізації виробничих процесів;
- «ко–локації» або зосередження розробки та управління під проектом суто у одній географічній локації.

Останній пункт надає можливість реалізовувати під – проекти суто у рамках гнучких методів, використовуючи усі корисні властивості притаманні цій групі методів. Проекти віднесені до третьої групи, не зважаючи на існуючу проблематику, являють собою досить цікавий сегмент ринку. Як правило автоматизувати такий продукт за допомогою дешевого, або безкоштовно, інструментарію не ефективно – багато чого у продукті цей інструментарій не в змозі буде «подолати». А придбання дорогого продукту від лідерів ринку, такого як Hewlett – Packard Unified Functional Testing, Hewlett – Packard LeanFT, тощо, може бути за межами операційного бюджету. Адже одна копія такого продукту коштуватиме від 3000 – 7000 доларів США на рік.

Інша альтернатива – використовувати для тестування пропозицію Software as Service, та запрошувати необхідний інструментарій у сервісної компанії. Іншими словами це означає почасову оренду виробничих ресурсів у публічній хмарі. Така пропозиція може зацікавити команди, проекти яких не потребують використання процесу CI/CD, та взагалі рідко проводять регресійні цикли. Також це мають бути проекти із низьким рівнем конфіденційності, пов'язаним із продуктом і виконуваними бізнес–процесами [4]. Фінансовий сектор, юридичні установи, різноманітні державні структури, фармацевтичний сектор – не повний лист галузей які не в змозі вивести свої проекти до публічної хмари. Та узяти на себе ризик передачі конфіденційних даних через мережу інтернет до сервісної організації, постачальника виробничих ресурсів проектів [3].

Проекти наступної методологічної групи, також як і складні проекти першої групи можуть отримати необхідний інструментарій саме завдяки продукту будь-якого проекту – багатофункціональної системи із відкритим вихідним кодом, інтегруючою у собі сучасні технологічні тренди із веб, та «десктоп» автоматизації у поєднанні із спеціальними функціями, які в змозі адресувати проблеми застарілих програмних продуктів. Варто зазначити, що можливість переробляти вихідний код надає широкі можливості еволюції

інструментарію автоматизації на конкретному об'єкті впровадження, забезпечує довго – тривалість економічного ефекту його імплементації. На основі аналізу сучасного стану, тенденцій розвитку та впровадження механізмів автоматизації тестування в проектній організації [5].

На основі впровадження інформаційної моделі, було підвищено ефективність створення і управління проектами. Авторами розкрито особливості планування якості у гібридному проекті великого обсягу, проведено дослідження споживачів, проведено аналіз конкурентів, дослідження попиту, проведено аналіз проекту, аналіз витрат і доходів організації, розроблений план виконання робіт, обґрунтовано концепцію проекту, доведено здійснимість та ефективність даного проекту.

Список джерел:

1. Рудніченко М.Д., Крамський С.О. Концептуальна модель управління людським капіталом у виробничій ІТ-організації на платформі нечітких множин // Збірник наук. праць. Управління розвитком складних систем. Київ: КНУБА. 2017. 32. С.32-41.
2. Муравецький С.А. Планування процесів забезпечення якості у великих та географічно розподілених гібридних ІТ–проектах // Вісник НТУ «ХП», Харків: 2016. № 1 (1173). С.106-109.
3. Boyko V., Rudnichenko N., Kramskoy S., Hrechukha Ye., Shibaeva. Concept implementation of decision support software for the risk management of complex technical system. *Springer international publishing book*, 2017. Pp.255 – 269. doi:org/10.1007/978-3-319-45991-2_17.
4. Крамской С.А. Метод оценки компетенций ролевого состава специалистов для комплектации ИТ-компании с использованием нечеткой логики // Управление развитием сложных систем. Київ: КНУБА. 2016. 28. С.81–89.
5. Крамський С.О. Застосування комплексної оцінки ризик-орієнтованих засобів в управлінні інноваційними ІТ-проектами. // Управління розвитком складних систем. 2017. 29. С.71–77.