

AGILE-МЕТОДОЛОГІЇ В У ПРОЦЕСІ НАВЧАННЯ ІНЖЕНЕРІВ-ПРОГРАМІСТІВ

Чорна Альона Віталіївна

асистент кафедри інформатики і кібернетики

МДПУ ім. Б. Хмельницького

Анотація. У статті висвітлена загальна концепція та принципи використання Agile-методології розробки програмного забезпечення, як платформи при підготовці майбутніх інженерів-програмістів. Обґрунтовані переваги і недоліки зазначеної методології. Визначений зв'язок використання agile-методології при формування управлінської компетентності інженерів-програмістів під час вивчення фахових дисциплін. Описані функції управління за умов впровадження Agile-методології у процес навчання інженерів-програмістів. Наведена схема роботи agile-методології під час вивчення дисциплін.

Ключові слова: Agile-методології, інженери-програмісти, процес навчання, управлінська компетентність.

Вступ. В підготовці кваліфікованого інженера-програміста велику увагу слід приділяти двом видам діяльності: проектна та операційна. До операційної діяльності відносять затверджений навчальний план, методичні матеріали по дисциплінам, контрольні точки, що розбиті на семестри. Проектна діяльність змінюється щорічно у зв'язку із оновленням навчальних програм, переліком дисциплін, зміною технологій в області інформаційних технологій, професійних вимог («soft skills» та «hard skills») до кваліфікованих працівників від потенційних роботодавців. Для вирішення поставлених перед освітнім процесом проблем підходить використання Agile-методології, як найбільш актуальної на сьогодні методології управління проектами.

Питаннями розробки і впровадження Agile-методологій у процесі розробки програмного забезпечення висвітлені в працях науковців: Б. Вольсон [1], К. Бек, М. Фаулер, М. Кон [3], Р. Мартін, Х. Кніберг. Використання Agile-підходу в освітню діяльність висвітлюють Н. Гульчевська, Г. Греф, А. Делхій [2], В. Війнадс та ін.

Мета роботи – проаналізувати особливості використання agile-методології у процесі навчання інженерів-програмістів під час формування управлінської компетентності.

Основна частина.

Кожен інженер-програміст на сьогоднішній день повинен мати цілком певні уявлення про управлінські аспекти своєї професійної діяльності, про основні теорії управління, про практику сучасного управління, що базуються, у великих корпораціях, насамперед на використанні інформаційних технологій. Знання про закономірності управління, про психологічні особливості поведінки людини в організаціях повинні розглядатися сьогодні як необхідний і невід'ємний компонент загальної культури особистості фахівця будь-якого профілю (розробник програмного забезпечення, розробник мобільних додатків, розробник, комп'ютерних ігор, системний адміністратор, дизайнер та ін.), який визначається відповідно до вимог, що пред'являються в сфері його професійної компетентності. Очевидно, що де б згодом не працював майбутній інженер-програміст, він завжди включається в управлінський контур деякої організації, займаючи в ній певне місце, часто - керівне. Тому формування управлінської компетентності є одним із головним завданням підготовки майбутніх інженерів-програмістів.

Управлінська компетентність інженерів-програмістів полягає в інтегруванні управлінських (навички керування часом, делегування повноважень, керування проектами, прийняття управлінських рішень, робота в групах при розробці складних програм) та організаторських умінь (оптимізація діяльності людей у колективі) [4, с. 73].

Проаналізувавши праці науковців, ми дійшли висновку, що формування управлінської компетентності майбутніх інженерів-програмістів відбувається в процесі вивчення фахових дисциплін («Вступ до спеціальності», «Операційні системи та системне програмування» та «Управління програмними проектами») у підготовці майбутніх інженерів-програмістів.

Для ефективного формування управлінської компетентності слід використовувати Agile методології, що базується на ітеративній розробці, тобто робота виконується серією коротких циклів, що складаються з планування, реалізації, перевірки і оцінки. До основних функцій управління за умов впровадження Agile-методології у процес навчання інженерів-програмістів можна віднести: планування (набуття студентами нових сучасних та актуальних на кожний момент часу компетентностей; впровадження практичної спрямованості у процес навчання; отримання студентами першого практичного досвіду роботи), організація (розбиття навчального процесу на більш короткотривалі проміжки часу - спринти; розділення студентів на невеликі групи; призначення відповідальних за створення та формування команд; закріплення за кожною групою наставника (куратора) для ефективного виконання принципів Agile-методології; розподілення ролей, робот у команді), мотивація (новий спосіб отримання теоретичних знань, практичних умінь, навичок; максимальне наближення навчального процесу до робочого; отримання вмінь роботи в команді та самостановлення в колективі; зручний робочий простір для більш ефективної самоорганізації), контроль (контроль за ефективним виконання принципів agile-методології; оцінка роботи кожного учасника команди куратором та складання рейтингу за студентами; оцінка злагодженості та успішності роботи команд, складання рейтингу команд)

Agile-методологія має декілька особливостей, за допомогою яких можна ефективно управляти навчальним процесом майбутніх інженерів-програмістів: спринти (обмежений фіксований часовий період, протягом якого команда має виконати поставлені об'єми роботи); Stand-up мітинги (щоденний командний

мітинг для перевірки повноти виконаного завдання), сценарії користувачів (можливість швидко аналізувати потреби користувача) [5].

Для ефективного впровадження Agile-методології потрібно виконати наступну організацію. Група студентів розбивається на команди, кожній команді необхідно освоїти всі теми предмета. Черговість вивчення, спосіб вивчення делегується самим командам. Scrum-дошка містить стікери зі всіма темами, такими, що підлягають вивченню. Група переміщує кожен стікер від стовпчика «до вивчення», «в прогресі» до колонки «освоєно». Скрам-дошка може бути матеріальною, а може бути віртуальною. Викладач лише контролює останню стадію, чи дійсно команда освоїла тему. Також завдання викладача скоординувати роботу команд. Кожна команда може розподіляти ролі відповідно до того, хто що краще вміє робити, проводити обговорення по Agile-методології. Основні принципи роботи Agile-методології показано на рис. 1.

Схема Agile-методології

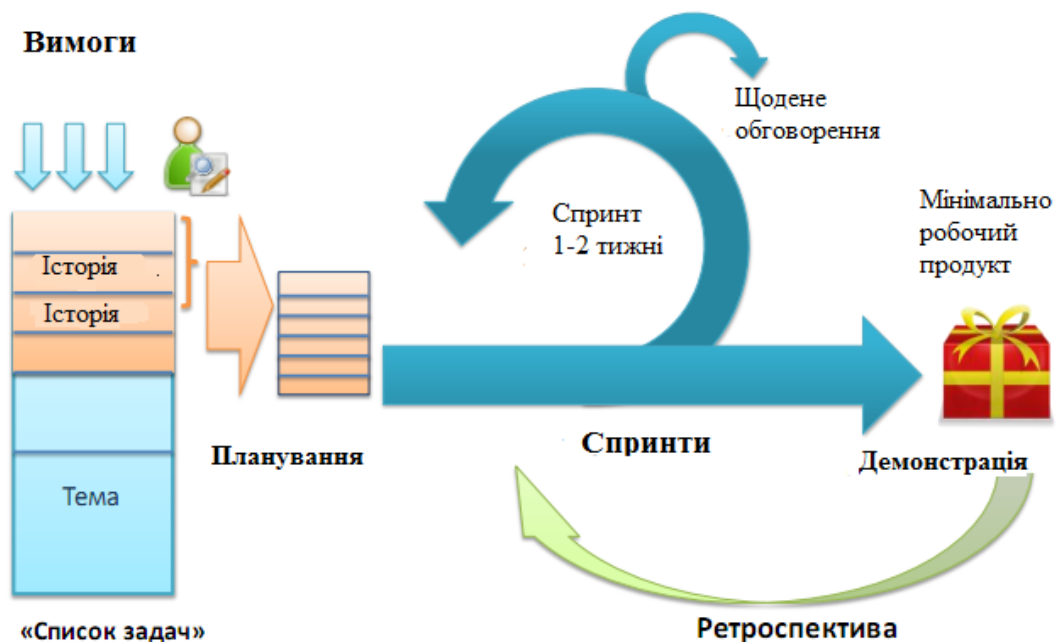


Рис. 1. Схема Agile-методології

Для кожної команди повинно бути організовано робочий простір - аудиторія, де буде: робоче місце з необхідним обладнанням та програмним забезпеченням для кожного студента; дошка завдань (agile board) для наочного

зображення поточного стану виконання завдань; місце для проведення stand-up мітингів.

Також необхідне використання електронних та Інтернет-ресурсів для організації виконання завдань, перевірки та оцінювання, швидкого доступу до необхідних матеріалів, наприклад, таких як Skype, Moodle, Google drive, Google docs/sheets та інші.

Висновки. Таким чином, при правильному впровадженні Agile-методології та дотриманні усіх її вимог, процес навчання інженерів-програмістів (набуття управлінської компетентності) може сприяти підвищенню рівня підготовки випускників та їх конкурентоспроможності на ринку праці. Безперечно, запропонований підхід може і надалі вдосконалюватися і потребує подальших досліджень та модернізацій відповідно до національних, технологічних та економічних особливостей.

Література

1. Вольсон Б. Гибкие методологии разработки [Електронний ресурс]. – Режим доступу: <http://adm-lib.ru/books/10/Gibkie-metodologii.pdf>
2. Делхий А. Руководство eduScrum «Правила игры» / А. Делхий, Рини ван Солиген, В. Вийнадс [Електронний ресурс]. – Режим доступу: http://eduscrum.nl/file/CKFiles/The_eduScrum_Guide_RU_1.2.pdf.
3. Кон М. Scrum: гибкая разработка ПО. Пер. с англ. / М. Кон – М.: ООО “И.Д. Вильямс”, 2011. – 576 с.
4. Круглик В.С. Структура професійної компетентності майбутнього інженера-програміста/ В.С. Круглик, В.В. Осадчий // Педагогічний дискурс: зб. наук. праць за ред. І.М.Шоробура – Хмельницький: ХГПА, 2016. – Вип.21. – С. 69-75.
5. Сазерленд Д. Scrum. Революционный метод управления проектами / Д. Сазерленд ; пер. с англ. М. Гескиной — М.: Манн, Иванов и Фербер, 2017. – с. 272.