

## Лекція №12. Ігрові симулятори, тренажери, навчальні ігри.

Десятки років тому відомий американський інженер-програміст та вчений-інформатик Фредерік Брукс (Frederick Brooks) у своєму класичному творі "Легендарний людино-місяць" (The Mythical Man-Month) написав: "Найважчою складовою процесу побудови програмної системи є прийняття однозначного рішення про те, що саме необхідно побудувати. Жодна з інших складових роботи над концепцією не є настільки складною, як визначення детальних технічних вимог, включаючи всі аспекти зіткнення продукту з людьми, машинами та іншими програмними системами. Жодна інша складова цієї роботи не може завдати такої значної шкоди отриманій у результаті системі, якщо дана складова виконана неправильно. Саме цю складову процесу розробки найважче виправити на більш пізніх етапах" [3].

Розробку програмного забезпечення, а особливо великих систем програмного забезпечення, вважають суто технічним завданням. На практиці виявляється інакше: розробка великих (і не тільки) систем є завданням здебільшого "нетехнічного" характеру. Тобто визначальними є аналіз і формулювання вимог до програмного забезпечення, управління розробкою програмного забезпечення, методології, а також моделі та методи розробки ПЗ. Це пов'язано з тим, що планування й аналіз мають надзвичайно важливе значення при розробці ПЗ. Ті, хто брав участь у реальних проектах погоджуються, що погане планування є головною причиною провалів проектів [7]. Другим критичним чинником, що призводить до невдачі, є недостатнє розуміння потреб користувачів ПЗ [5]. Комунікація та управління є важливими в проектах будь-яких сфер, та особливого значення вони набувають саме в проектах із розробки ПЗ. Управління проектами програмного забезпечення означає: насамперед збір усіх необхідних даних, належну поінформованість персоналу, а також координацію індивідуальних потреб і завдань із загальними цілями.

Тоді як більшість підходів до навчання майбутніх інженерів-програмістів спрямовані на додавання реалізму в практичні заняття в аудиторії, деякі автори (М. Баррос (M. Barros), А. Бейкер (A. Baker), К. Вернер (C. Werner), А. Дантас (A. Dantas), Е. Наварро (E. Navarro), А. ван дер Хук (A. van der Hoek)) стверджують, що для студентів єдиним можливим способом набуття досвіду участі в реальних процесах розробки ПЗ в академічному середовищі є використання ігрових симуляторів у поєднанні з лекціями й навчальними проектами.

Розглянемо детальніше ідею надання студентам досвіду участі в реальних процесах розробки ПЗ в академічному середовищі за допомогою використання ігрових симуляторів.

Відповідно до праць декількох авторів (С. Колфілд (C. Caulfield), С. Мей (S. Maj), Дж. Ся (J. Xia) та Д. Віл (D. Veal)), існує також декілька ігрових стимуляторів у галузі навчання та підготовки майбутніх інженерів-програмістів, що представлені в їх систематизованому огляді літератури [2, с. 32].

Ігрові симулятори можуть стати і засобом набуття досвіду, а також, враховуючи їх привабливий характер, і засобом мотивації студентів. Крім того, ігрові симулятори дозволяють у процесі навчання брати участь у реальних сценаріях без ризиків.

У 2015 році Р. Атал (R. Atal) і А. Сурека (A. Sureka) проаналізували роботи дослідників із 2000 року до 2013 року, опубліковані за темою навчання розробки ПЗ із використанням концепції моделювання гри [1, с. 65-67].

Розглянемо більш детально виділені у п. 1.3. ігрові симулятори та їх роль у формуванні професійних м'яких компетентностей майбутніх інженерів-програмістів.

**SESAM** (Software Engineering Simulation by Animated Models), створений А. Драппа (A. Drappa) та Дж. Людвіг (J. Ludewig), є одним із перших симуляторів, розроблених для освітніх цілей у 2000 році. Являє собою

середовище моделювання процесу розробки програмного забезпечення (див. рис. 1), у якому студенти керують командою віртуальних співробітників для виконання віртуального проекту за графіком, у межах бюджету, а також на рівні або вище необхідного рівня якості. Даний симулятор використовує дуже гнучку й виразну мову, але процес побудови моделі є трудомістким, вимагає певного навчання та написання коду в текстовому редакторі. SESAM – це перший приклад мови моделювання процесу програмного забезпечення, який є розпорядчим, прогнозованим та інтерактивним (але не графічним).

```
Aug 10> ask Patrick about his experience
Patrick Murphy ponders over the question, before he answers:
"Well, I think my experience is average."
Aug 10> let specify
Who shall specify? Patrick
Patrick Murphy starts working on the specification.
Aug 10> proceed 14 days
Aug 24>ask Patrick about his work
Patrick Murphy reports: The specs are now 30 pages long, I think it
will soon be finished
Aug 24> proceed 21 days
The customer has called, he has received a 63-pages specs on
Sep 7. He will check it.
Sep 14>
```

*Рис. 1. Симулятор SESAM*

У 2004 році Е. Наварро (E. Navarro) розробляє **SimSE** – інтерактивний, графічний та гнучкий у налаштуванні ігровий симулятор процесу розробки програмного забезпечення для освітніх цілей. У SimSE є можливість вирішення проблеми відсутності вивчення процесу розробки програмного забезпечення в типовому курсі програмної інженерії, надаючи студентам практичний досвід реалістичного процесу розробки програмного забезпечення в привабливому вигляді. Це дозволяє студентам практикувати "віртуальний" процес розробки програмного забезпечення в повністю графічному, інтерактивному середовищі, де прямий і зворотній зв'язок у графічному вигляді дозволяє їм вивчити складні причинно-наслідкові зв'язки, що лежать в основі процесів програмної інженерії (див. рис. 2).

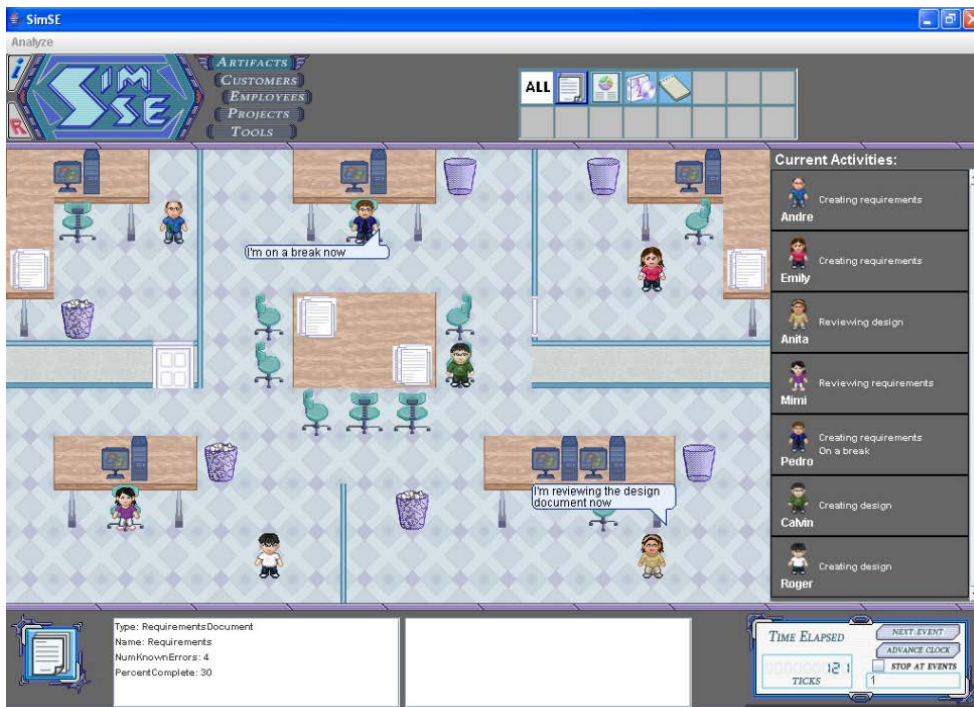


Рис.2. Симулятор SimSE

У цьому ж році інший симулятор **The Incredible Manager** був уведений Марсіо Барросом (Marcio Barros), Александре Дантасом (Alexandre Dantas) та Клаудією Вернер (Claudia Werner). Це ігровий симулятор, розроблений спеціально для підготовки менеджерів проектів програмного забезпечення. Він більшою мірою зосереджений на управлінні проектами, ніж на процесі розробки програмного забезпечення. По суті, це промисловий симулятор із додатковим графічним користувацьким інтерфейсом, подібним до ігрового. **The Incredible Manager** (див. рис. 3) дозволяє налаштовувати й імітаційні моделі через свій текстовий інтерфейс.

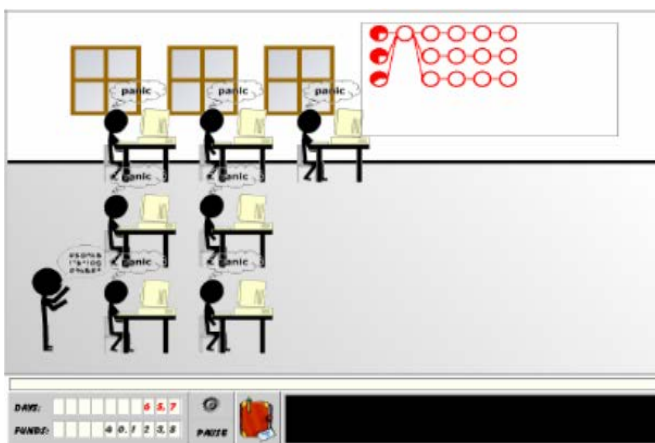


Рис. 3. Симулятор *The Incredible Manager*

**SimjavaSP** – це інтерактивний, веб-інтерфейсний, графічний симулятор процесу розробки програмного забезпечення, який з'явився в кінці 2005 року. Мета студента, який діє в SimjavaSP як менеджер проекту – розробити заданий проект програмного забезпечення протягом заданого часу та, використовуючи заданий бюджет, досягти при цьому відповідної запланованої якості. Користувацький інтерфейс (див. рис. 4) для SimjavaSP являє собою поєднання графічного та текстового зворотного зв'язку.

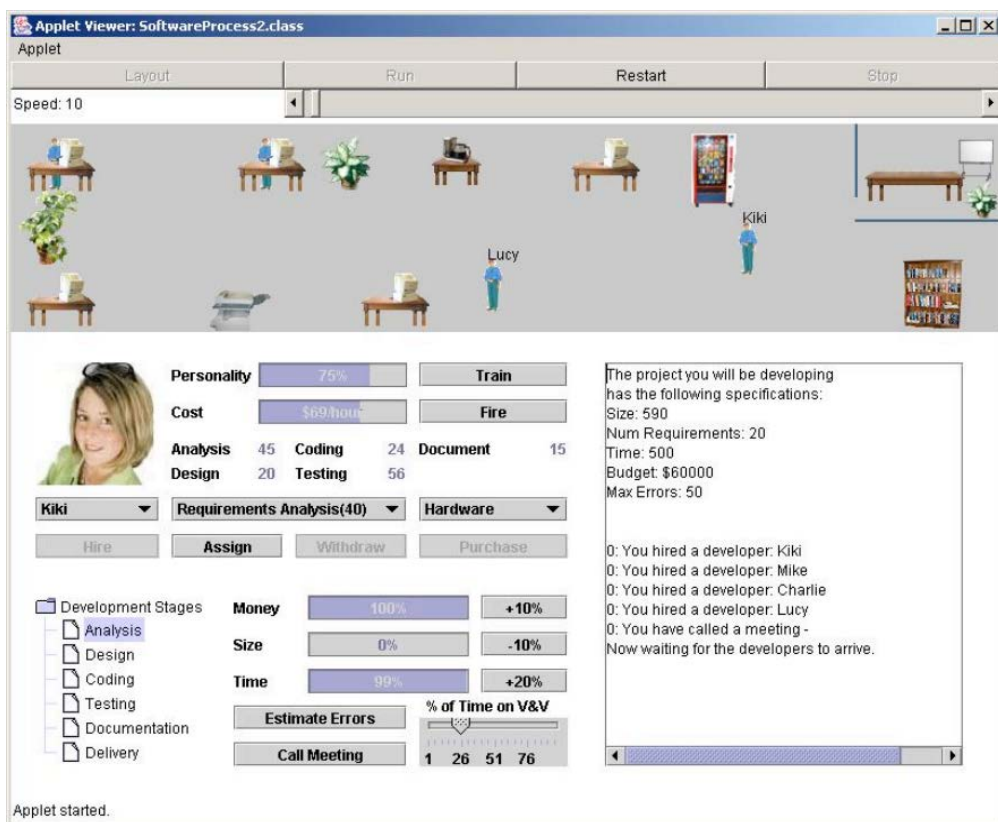


Рис. 4. Симулятор *SimjavaSP*

**ProMaSi** (Project Management Simulator), розроблений Н. Петалідісом (N. Petalidis), дозволяє студентам активно перевіряти свої навички в управлінні розробкою програмного забезпечення за допомогою виконання відповідних керуючих дій у симуляторі (див. рис. 5). Варто зазначити, що одним із нових підходів, запропонованих у даному симуляторі, є поділ складових частин симулятора на архітектурні модулі, що дозволяє третім сторонам створювати та додавати свої власні доробки.

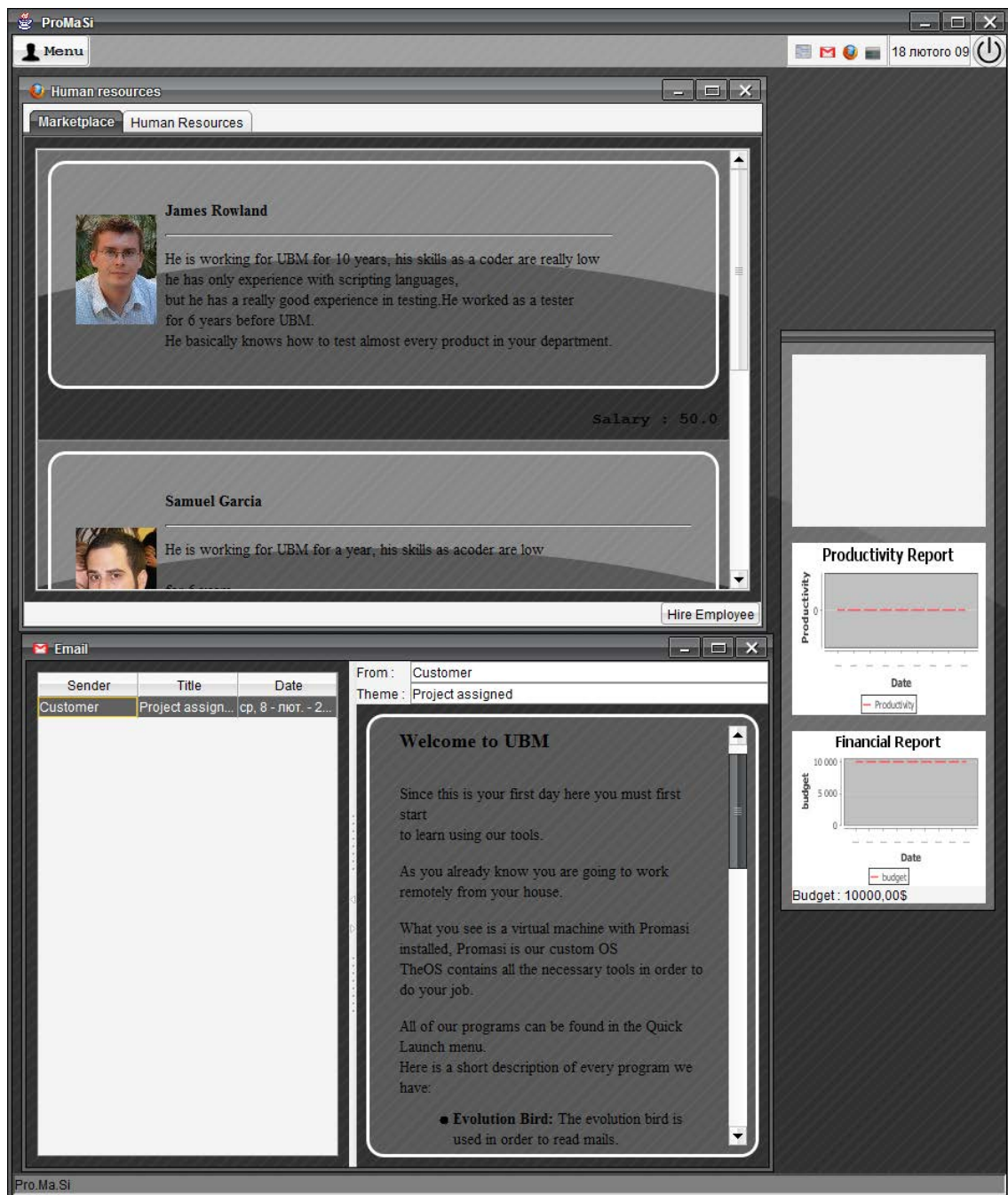
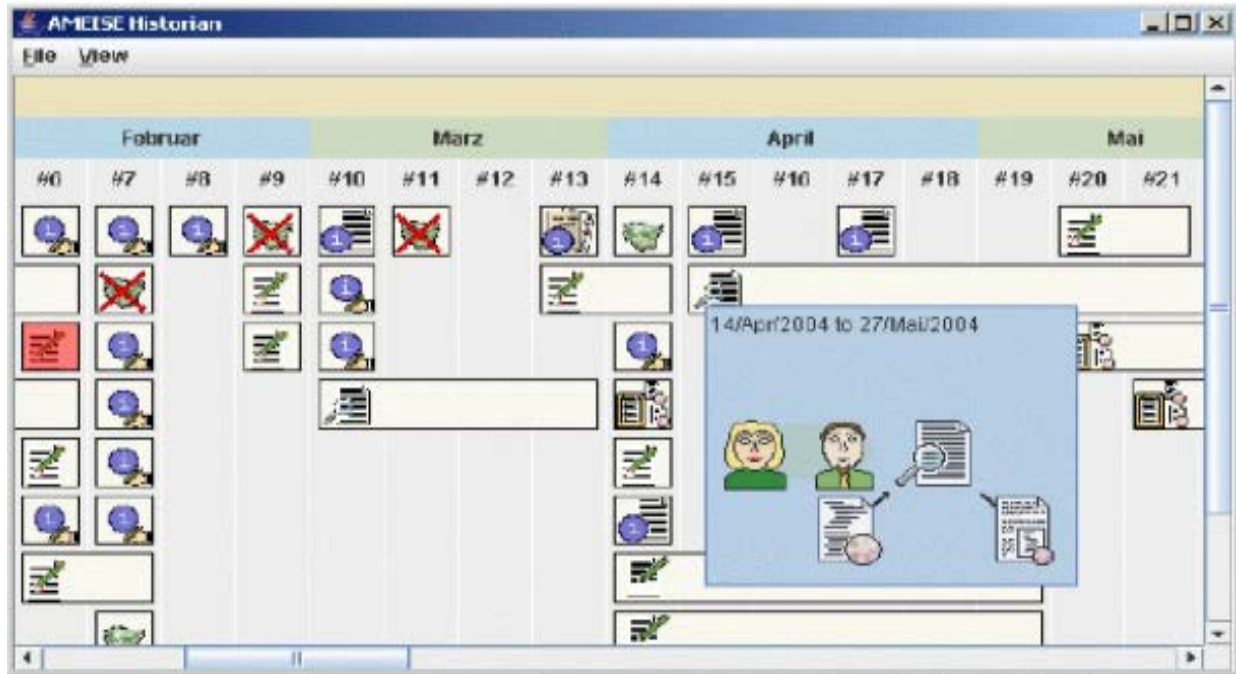


Рис. 5. Симулятор ProMaSi

У 2013 році А. Боллін (A. Bollin) розробив **AMEISE** (A Media Education Initiative for Software Engineering) – ігровий симулятор, де студенти беруть на себе роль технічного керівника проекту. Під час ігрової симуляції вони стикаються з труднощами, що виникають у процесі керування проектом відповідно до конкретної моделі завдання, обраної викладачем. Інструктор або викладач має змогу вибрати кількість випробувань (запусків моделювання) для вирішення поставлених завдань у межах заданих обмежень. Студенти

можуть вчитися на основі власних попередніх запусків ігрового симулятора, змінювати стратегії та вимірювати свій власний успіх за допомогою функції самооцінки (див. рис. 6).



*Рис. 6. Гра-симулятор AMEISE*

**PRODEC:** ігровий симулятор для підготовки менеджерів розробки програмного забезпечення, розроблений Алехандро Кальдероном, Мерседес Руїсом. Автори зазначають, що хоча є деякі роботи, пов'язані із застосуванням ігрових симуляторів для навчання управління програмними проектами, на сьогодні не вистачає інструментів, що поєднують у собі навчання й оцінювання в одному інструменті, а також які забезпечують для студентів можливість експериментувати в прийнятті рішень в умовах, близьких до реальних. Project Decision (PRODEC) – це ігровий симулятор (див. рис. 7), створений з метою підготовки й оцінювання студентів у галузі розробки та управління програмними проектами.



Рис. 7. Гра-симулятор PRODEC:

Основне завдання цього стимулятора полягає в тому, щоб, використовуючи привабливий характер гри, розмістити учнів у віртуальній організації, де вони можуть управляти програмними проектами й вирішувати реальні проблеми у середовищі без ризиків. Для викладачів PRODEC – це інструмент, що допомагає розглядати такі питання, як керівництво, менеджмент завдання та команди, моніторинг проектів, контроль і управління ризиками. Також даний симулятор допомагає викладачам оцінити знання та навички, які студенти розвивають та здобувають у даному ігровому симуляторі. Після будь-якої спроби PRODEC пропонує повний звіт, що



представляє кожне рішення, яке зробили гравці, а також результат відповідно до застосованих критеріїв оцінки, заданих викладачем на початку гри.

**Simsoft** (Caulfield, Veal, & Maj, 2011b) – ігровий симулятор, розроблений, щоб з'ясувати, як такі симулятори можуть вплинути на якість підготовки майбутніх інженерів-програмістів і менеджерів проектів. Simsoft поставляється у двох частинах. По-перше, це роздруковане ігрове поле розміру A0, навколо якого гравці збираються разом, щоб обговорити поточний стан свого проекту й розглянути питання про свій наступний крок. Поле показує хід гри, тоді як пластикові лічильники використовуються для індикації кількості співробітників проекту. Покерні фішки являють собою бюджет команди, за допомогою якого вона може придбати більше співробітників, а також звідки кошти можуть забиратися в залежності від рішень, прийнятих під час гри. Існує також простий Java додаток (рис. 8), за допомогою якого гравці можуть побачити поточний та історичний стан проекту в серії звітів і повідомлень, а також де гравці можуть налаштувати параметри проекту. Програмною основою Simsoft є модель динаміки системи, яка втілює в собі невеликий набір фундаментальних причинно-наслідкових зв'язків доволі простих проектів розробки програмного забезпечення. У процесі проходження симуляції Simsoft команди студентів, практикуючих менеджерів проектів та інженери-програмісти керують гіпотетичним проектом розробки програмного забезпечення з метою завершення проекту в зазначений термін і в межах бюджету. На основі вхідного сценарію ігрового симулятора, відомостей, що надаються під час ігрової симуляції, та власного досвіду, гравці приймають рішення про те, як варто діяти: наймати більше співробітників або зменшити їхню кількість, скільки часу витратити на розробку тощо.

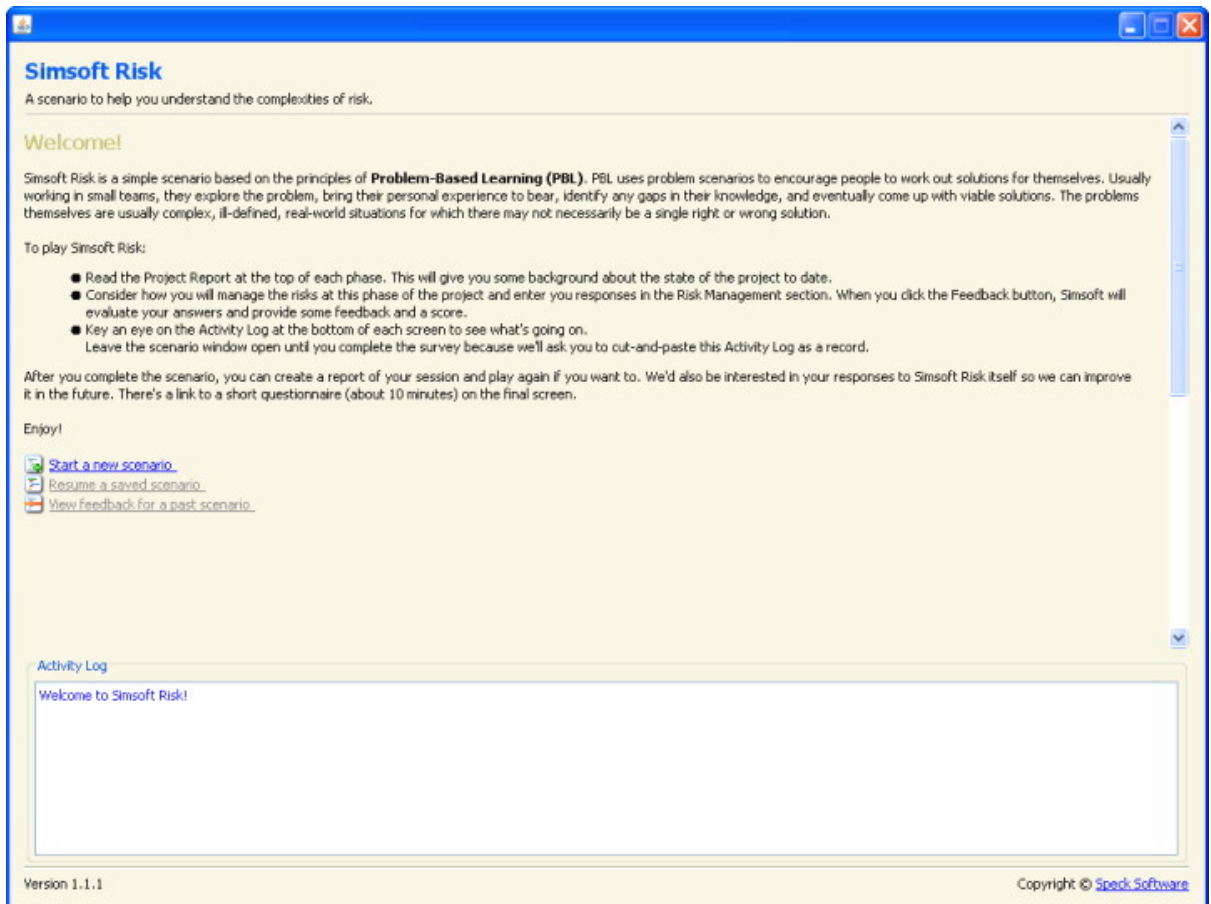


Рис. 8. Симулятор Simsoft

Під час використання розглянутих вище ігрових симуляторів у навчальному процесі формуються такі професійні м'які компетентності:

- 1) **Здатність до адаптації** – у процесі ігрової симуляції гравець повинен швидко адаптуватися до змін і розглядати нові підходи.
- 2) **Здатність до вирішення проблем** – гравцю необхідно виявляти проблеми, а також використовувати логіку, здоровий глузд і наявні дані для оцінки альтернативних рішень досягнення поставленої мети та бажаного результату.
- 3) **Здатність до збирання й аналізу даних** – під час ігрової симуляції гравцю необхідно шукати, збирати та синтезувати дані й знання і неупереджено доходити висновків, мети або суджень, що є базою для прийняття рішень і формування стратегії.

4) **Здатність до звітності** – гравець бере на себе відповідальність за успішне виконання поставлених цілей та досягнення результатів, установлюючи високі стандарти роботи для себе й віртуальних співробітників.

5) **Здатність до орієнтації на кінцевий результат** – гравцю необхідно фокусуватись на бажаному результаті, ставити складні стимулюючі цілі та досягати їх.

6) **Здатність до використання правил та процедур** – гравець повинен розуміти та застосовувати знання про положення, правила і процедури гри та процес розробки програмного забезпечення.

7) **Здатність до зовнішньої та організаційної поінформованості** – для успішного проходження ігрової симуляції гравець повинен визначати і розуміти, як внутрішні та зовнішні тенденції (наприклад, економічні, політичні, соціальні тощо) впливають на роботу організації.

8) **Здатність до планування та пріоритезації** – під час проходження ігрової симуляції гравець планує й організовує трудову діяльність, а також повинен бути здатним керувати декількома завданнями одночасно.

9) **Здатність до політичної кмітливості** – гравець повинен проявляти впевненість та професійну дипломатію і водночас налагоджувати ефективні взаємини з людьми на всіх рівнях, всередині та зовні організації.

10) **Здатність до прийняття рішень** – для успішного проходження ігрової симуляції гравцеві потрібно здобувати необхідні знання, уміння та навички, визначати ключові питання й наслідки для прийняття обґрунтованих та об'єктивних рішень.

11) **Здатність до роботи в команді** – гравцю необхідно працювати разом з іншими віртуальними співробітниками й сприяти досягненню спільних поставлених цілей.

12) **Здатність до співпраці** – у процесі ігрової симуляції гравець працює спільно з іншими, всередині та за межами організації, для досягнення цілей, що ведуть до створення й підтримки взаємовигідних партнерських взаємин, максимально ефективного використання знань та досягнення відповідних результатів.

13) **Здатність до стійкості** – для успішного проходження ігрової симуляції гравцеві необхідно зберігати високу продуктивність та самоконтроль під тиском або під час негараздів.

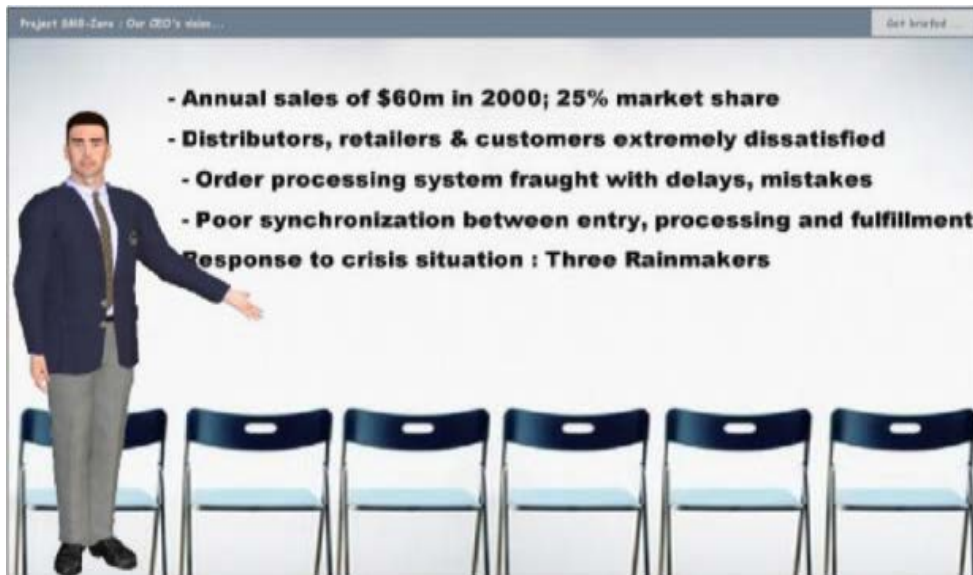
14) **Здатність до звернення уваги до дрібниць** – у процесі ігрової симуляції гравець повинен гарантувати повноту й точність даних і знань, а також слідкувати, щоб домовленості та зобов'язання були виконані.

15) **Здатність до управління змінами** – гравцю необхідно розуміти необхідність змін, планувати та пристосуватися до них настільки творчо й позитивно, наскільки це можливо.

16) **Здатність до якісного та кількісного аналізу** – для успішного проходження ігрової симуляції гравцю необхідно аналізувати й оцінювати наявні дані для керування та досягнення необхідних результатів.

У 2006 році А. Джейн (A. Jain) спеціально розробив симулятор **SimVBSE**, щоб навчити студентів теорії вартості в основах програмної інженерії. SimVBSE має повністю графічний користувацький інтерфейс, що включає також анімацію та аудіо (див. рис. 9). Ігровий симулятор відтворює один конкретний приклад із реального світу та не включає в себе опцій налаштування.

При використанні в навчальному процесі даного ігрового симулятора, формуються аналогічно такі здатності до: адаптації, вирішення проблем, збору та аналізу даних, звітності, орієнтації на кінцевий результат, планування та пріоритезації; політичної кмітливості, прийняття рішень, стійкості, звернення уваги до дрібниць, якісного та кількісного аналізу.



*Рис. 9. Симулятор SimVBSE*

**ANUKARNA** – ігровий симулятор процесу розробки програмного забезпечення для практичного навчання основ прийняття рішень у процесі виконання експертної оцінки коду (Peer Code Review), розроблений Рітіка Аталом (Ritika Atal). Автор створив веб-інтерактивну освітню гру-симулятор процесу розробки програмного забезпечення (рис. 10) для пояснення та викладання переваг і передових практик процесу експертної оцінки коду (Peer Code Review). Р. Атал описує основи та модель навчання, що засновані на "навчанні у процесі відкриття", навчанні на помилках, аргументах та міркуваннях щодо викладання основних концепцій практики експертної оцінки коду. Також він оцінює запропоновані основи, модель і симулятор шляхом проведення експериментів та збору зворотного зв'язку від користувачів і представляє отримані результати.

При використанні в навчальному процесі даного симулятора формуються аналогічно такі компетентності: здатність до вирішення проблем,

здатність до збору та аналізу даних, здатність до орієнтації на кінцевий результат, здатність до прийняття рішень, здатність до роботи в команді, здатність до співпраці, здатність до стійкості, здатність до звернення уваги до дрібниць, здатність до управління змінами, здатність до якісного та кількісного аналізу, здатність до використання правил та процедур.

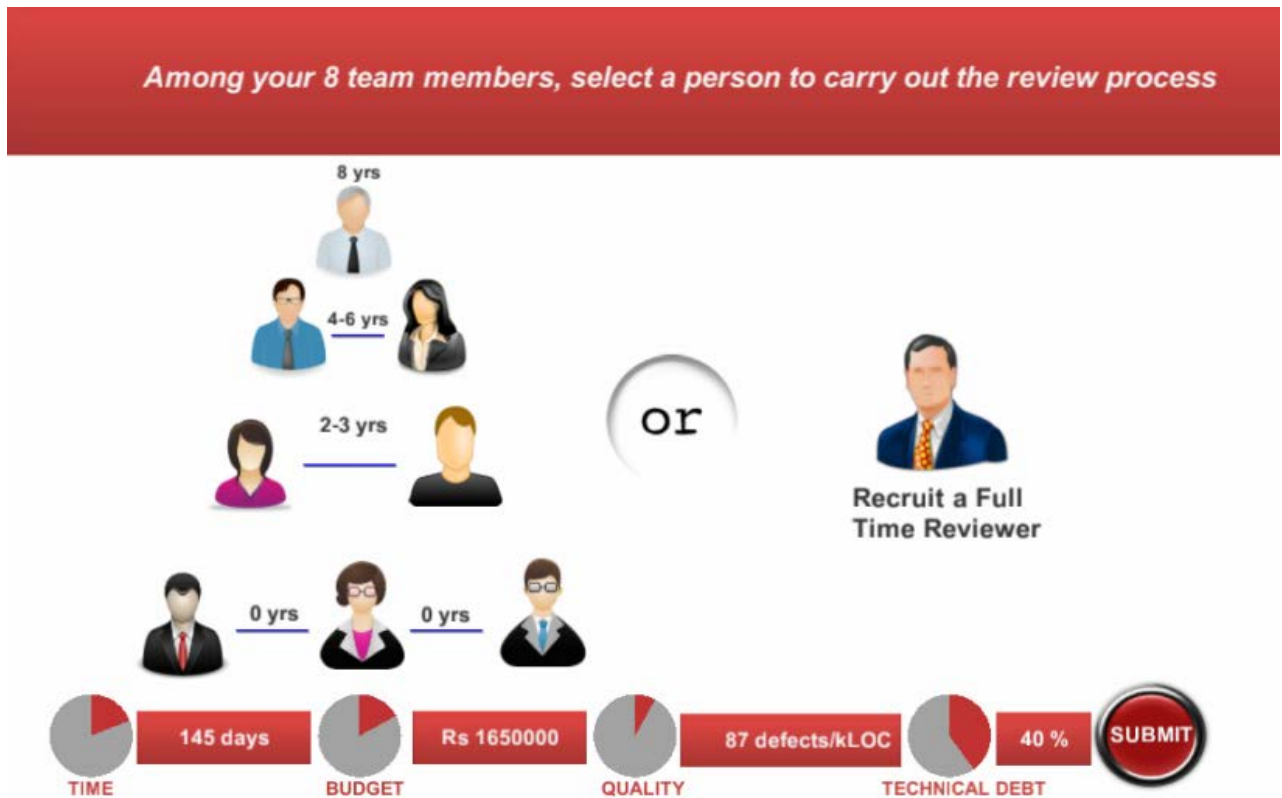


Рис. 10. Симулятор ANUKARNA

**Game Dev Tycoon** – ігровий бізнес-симулятор, що симулює діяльність та процес управління компанією із розробки ігор. Даний ігровий симулятор пропонує привабливий користувацький інтерфейс, а також інтерактивний і привабливий ігровий процес. У Game Dev Tycoon (рис. 11) студенти беруть на себе роль власника та менеджера власної компанії з розробки ігрового програмного забезпечення. На початку ігрової симуляції студенти є незалежними розробниками програмного забезпечення, що працюють самі на себе. У міру того як власна студія розробки програмного забезпечення стає успішнішою, з'являється все більше можливостей для розширення бізнесу вже разом із декількома співробітниками та на більш великій робочій площі.

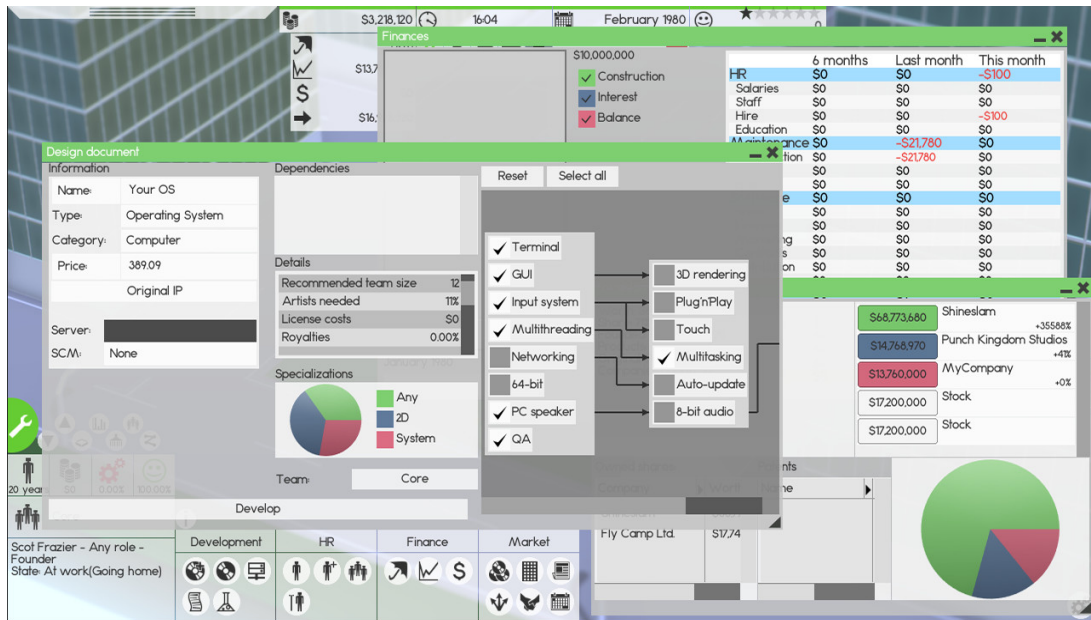
При використанні в навчальному процесі даного ігрового симулятора, формуються аналогічно такі компетентності: здатність до роботи в команді, здатність до співпраці, здатність до прояву професійної чесності та етики, здатність до планування та пріоритезації, здатність до вирішення проблем, здатність до комунікативності, здатність до прийняття рішень, здатність до орієнтації на кінцевий результат, здатність до підтримки міжособистісних відносин, здатність до звітності, здатність до адаптації, здатність до звернення уваги до дрібниць, здатність до обслуговування клієнтів, здатність до ініціативності, здатність до стійкості, здатність до інноваційності, здатність до зовнішньої та організаційної поінформованості, здатність до використання правил та процедур.



*Рис. 11. Графічний користувацький інтерфейс ігрового симулятора Game Dev Tycoon*

**Software Inc** – це ігровий симулятор (див. рис. 12) на тему управління компанією, що займається розробкою програмного забезпечення, від її заснування до перетворення на велику корпорацію в цифровому світі. Ігрова симуляція починається зі створення невеликого офісу з декількома

інженерами-програмістами (або лише із засновником компанії) та з розробкою програмного забезпечення на замовлення. Коли компанія досягне певного успіху, наступним рівнем стане розробка власних програмних продуктів.



*Рис. 12. Графічний користувацький інтерфейс ігрового стимулятора Software Inc*

При використанні в навчальному процесі даного ігрового симулятора, формуються аналогічно такі компетентності: здатність до роботи у команді, здатність до співпраці, здатність до планування та пріоритезації, здатність до вирішення проблем, здатність до прийняття рішень, здатність до орієнтації на кінцевий результат, здатність до звітності, здатність до адаптації, здатність до звернення уваги до дрібниць, здатність до обслуговування клієнтів, здатність до ініціативності, здатність до стійкості, здатність до інноваційності, а також здатність до використання правил та процедур.



### Список використаних джерел:

1. Atal R., Sureka A. Anukarna: A Software Engineering Simulation Game for Teaching Practical Decision Making in Peer Code Review. *1st International Workshop on Case Method for Computing Education (CMCE)*, 2015. P. 63-70.
2. Caulfield C., Xia J. C., Veal D., Maj S. P. A Systematic Survey of Games Used for Software Engineering Education. *Modern Applied Science*, 2011. № 5 (6). Pp. 28-43.
3. Frederick P. Brooks. *The Mythical Man-Month: Essays on Software Engineering, Anniversary Edition (2nd Edition)* by Frederick P. Brooks
4. 140. Metzger, P.W., Boddie, J. *Managing a Programming Project (3rd Ed.)*. Prentice Hall, Englewood Cliffs, New Jersey, 1996.
5. Концедайло В. В. Застосування ігрових симуляторів у формуванні професійних компетентностей майбутніх інженерів-програмістів. Дисертація на здобуття наукового ступеня кандидата педагогічних наук (доктора філософії) за спеціальністю 13.00.10 "Інформаційно-комунікаційні технології в освіті" (011 – Освітні, педагогічні науки). – Київ, 2018.
6. Концедайло В.В, Вакалюк Т. А. Методичні рекомендації з добору та впровадження ігрових симуляторів у навчальний процес підготовки майбутніх інженерів-програмістів. Житомир, 2018. 60 с.