1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ПОСТАНОВКА ЗАДАЧІ

1.1 Аналіз задачі

Оскільки ціллю даної роботи є розробка компонентів системи [1], яка б дозволила автоматизувати процес управління проектами у виробничому процесі, то для її досягнення необхідно реалізувати наступні задачі:

– розробити календарний план створення системи;

– розробити модель системи;

– визначити методи рішення для створення підсистеми управління проектами, визначити засоби програмування, тестування;

– розробити алгоритм вибору найкращої методології управління проектом на основі даних введених користувачем;

– розробити програмні модулі системи;

– провести тестування системи (тестування системи на різні рекомендації, щодо вибору методології на основі різних вхідних даних від користувача, а також перевірка на відмово стійкість та безпеку);

– скласти проектну документацію.

1.2 Огляд і аналіз сучасного стану проблеми, що розглядається, а також існуючих методів і засобів вирішення задачі

Впровадження комп’ютерних технологій в усі сфери людської діяльності, призвели до того, що популярним стало ведення проекту у онлайн-режимі. Системи ведення проекту – це ресурси, які дозволяють вести задачі проекту, відстежувати час, ресурси, фінанси та звітність проекту і т.п. Даний спосіб має ряд переваг по відношенню до ведення проектів «на аркушах» як раніше. Переваги систем ведення проектів:

* Введення проекту на основі найкращих з існуючих методологій ведення проектів.
* Можливість організації ефективної системи багаторівневого планування, оптимізації і контролю виконання проектних робіт.
* Сучасні методи моніторингу стану проектів і робіт на основі аналізу контрольних точок.
* Управління необмеженою кількістю проектів будь-якої складності, з необмеженою кількістю робіт і ресурсів.
* Одночасне планування декількох проектів з урахуванням їх пріоритетів і обмежень на ресурси.
* Узгодження планів підрозділів і результатів зверху-вниз і знизу-вгору. Можна проаналізувати як зміни нижчестоящих планів вплинули на вищі і прийняти або відхилити ці зміни.
* Підбір співробітників потрібної кваліфікації або ролі, гнучке перерозподіл навантаження на фахівців.
* Пошук і призначення необхідного виду обладнання з урахуванням реального завантаження.
* Планування проектів як «вперед», визначаючи можливі терміни закінчення робіт, так і «назад», відштовхуючись від тимчасових обмежень.
* Пов'язане планування невиробничих робіт і виробництва.
* База знань по проектам і можливість створення проектів на основі типових фрагментів проектів і проектів-прототипів.
* Можливість отримання керівництвом підприємств візуальної інформації про хід проектів на всіх етапах робіт за допомогою діаграм Ганта, методу drill-down і методу контрольних точок.
* Взаємодія учасників проекту на основі регламентного документообігу, обміну повідомленнями, системи управління бізнес процесами.
* Можливість інтеграції з іншими інформаційними системами.

На даний момент для вирішення всіх цих проблем вимагає використання декількох програмних засобів, та витрачати багато часу на перенесення інформації с однієї до іншої.

Розглянемо ряд вже існуючих систем, що включають компоненти управління ходом виконання проекту, управлінням затрат часу, рекомендаційний компонент системи з вибору оптимальної методології управління.

1.2.1 «Microsoft Project»

Microsoft Project(MSP) – програма управління проектами, розроблена корпорацією Microsoft.

Основною задачею MSP є допомога менеджеру проекту у розробці планів, розподілі ресурсів по задачам, відстежуванні прогресу і аналізу обємів робіт. Дане програмне забепечення створює розклад критичного шляху та візуалізується в даграмі Ганта (рис. 1.1).

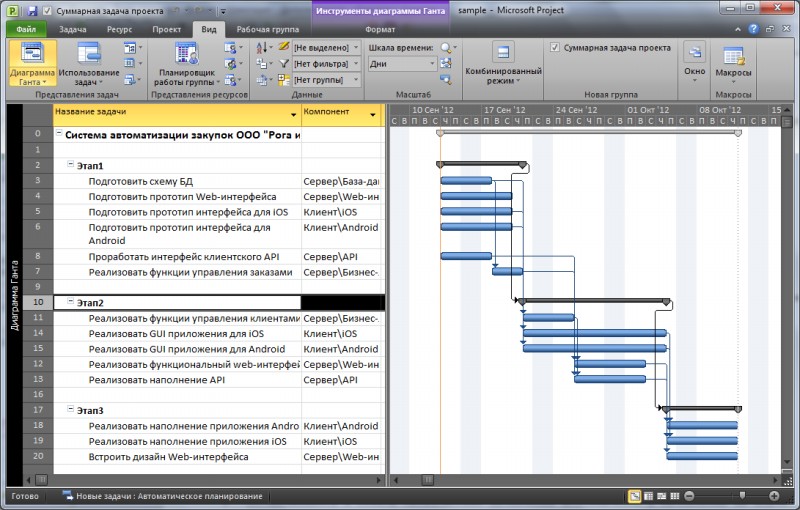


Рисунок 1.1 – Інтерфейс системи «MS Project»

MS Project має такі переваги:

- покроково розробляти проект, користуючись інтелектуальними підказками на різних рівнях;

- створювати завдання, розподіляти ресурси, вказувати терміни виконання та отримувати модель, яка буде максимально наближена до реальності;

- створювати мережеву діаграму проекту, тобто вказувати зв'язку завдань і пріоритетів з вирівнюванням ресурсів у міру реалізації проекту;

- розраховувати критичний шлях;

- представляти за допомогою графічних індикаторів проблемні завдання;

- міняти вказаний період виконання завдань у разі потреби;

- аналізувати як поточний стан справ, так і тенденції розвитку проекту за допомогою автоматизованих засобів підготовки звітів (причому, програма дозволяє запам'ятати проміжне поточний стан проекту, щоб повернутися до його аналізу пізніше);

- моделювати варіанти будь-яких рішень (наприклад, по зміні схеми фінансування, заміні механізмів і поставок);

MS Project має такі недоліки:

- застосовується при реалізації відносно невеликих проектних замислів, ящо більше – велика вартість ПЗ;

- реалізує класичний підхід до проектного менеджменту;

- не дає можливості створювати таски, сабтаски та назначати їх на участників команди;

- не має вбудованої баг-трекингової системи;

- система не дає можливості логування часу затрачуваного в дійсності на задачі;

- MS Project корисна лише на етапі проектування і не може бути самостійним інструментом для управління проектом на всіх етапах.

1.2.2 «Atlassian JIRA »

Atlassian JIRA — система відстеження помилок, призначена для організації спілкування з користувачами, хоча в деяких випадках може бути використана для управління проектами, розроблена компанією Atlassian Software Systems.

JIRA Software (рис 1.2) - це потужна комерційна система управління для розробників програмного забезпечення. Вона призначена для agile-проектів, а її інструменти підтримують такі методики, як kanban, scrum, а також ваші власні унікальні розробки.

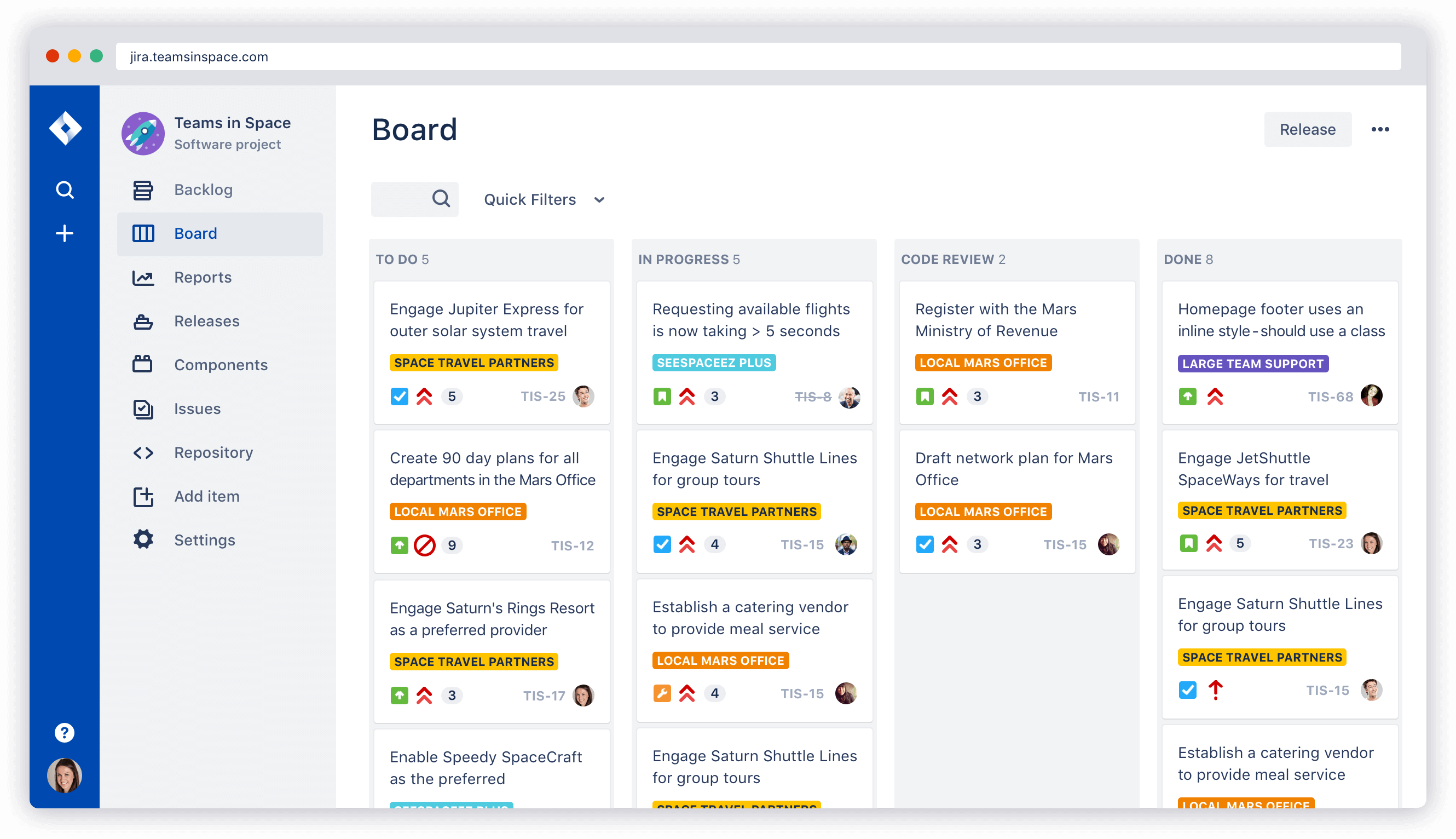


Рисунок 1.2 – Інтерфейс системи «Atlassian JIRA » на основі методології Канбан

Переваги ПЗ:

- можливість ведення одночасно декількох проектів одночасно;

- є можливість вибору методології ведення проекту;

- можливість створення тасків, сабтасків, багів, визначення їх пріоритетів та критичності виконання;

- створення наочних дашбордів, для слідкування за прогресом проекту;

- можливість розбиття задач на спринти;

- для швидкого здійснення ітерацій і поставки нових змін використовуються настроюються scrum-дошки;

- для досягнення максимального результату в стислі терміни застосовуються kanban-дошки.

- команди отримують десятки вбудованих звітів з необхідною інформацією.

- Portfolio for JIRA дозволяє контролювати всі команди і проекти, які стають видні як на долоні.

Недоліками JIRA є:

- слабка інтеграція методологій управління проектами;

- неможливість формування звітів про затрати часу на основі виконання завдань.

1.2.3 «Worksection»

Українська онлайн-система для управління проектами. Містить dashboard, завдання з коментарями, календар, сховище файлів, систему обліку часу, теги. Підтримка SSL, FTP користувача, субдоменів користувача. Відрізняється простотою і доступністю.

Worksection (рис.1.3) - система має ряд переваг: зручний і розвинений функціонал, легкість управління проектами, важливими процесами і завданнями. Система дозволяє працювати з файлами проекту і враховує історію версій файлів. Відмінна риса це захист даних, зручна система для планування часу та складання календарного плана.

Ідеально система підійде тим, хто працює над довгостроковими проектами і завданнями, якщо вам немає необхідності до точного дотримання термінів.

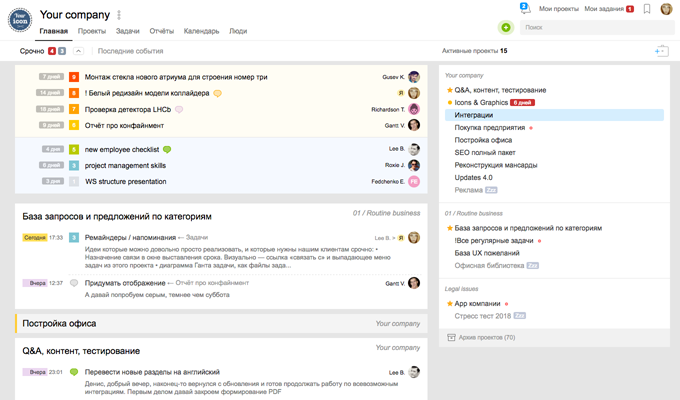


Рисунок 1.3 – Інтерфейс ПЗ Worksection

Плюси системи:

- можливість організації робіт з великою кількістю людей і над великою кількістю проектів;

- допомагає росту проектів і персоналу;

- дуже зручно використовувати для віддаленої роботи над проектом;

- адаптуватися в систему дуже легко, все нові співробітники швидко освоюються в системі;

- присутні мобільні версії.

Мінуси системи:

- незручний пошук;

- немає можливості поставити точний час для виконання завдань;

- немає можливості розсилати повідомлення про дедлайн на e-mail;

В наш існує велимка кількість систем управління проектами (СУП). Кожна з яких маж ряд своїх переваг та недоліків. На основі аналізу існуючих систем, можно зробити висновок, що веб-додаток розроблюємий у рамках атестаційної роботи має основну перевагу – це рекомендація методології вибору проекту на основі аналізу вхідних даних о проекті.

1.3 Різновиди та методи прийняття найбільш підходящої методології

Проект – це процес створення продукту. Це те, що робить команда, щоб видати замовнику продукт.

Продукт - те, що хоче отримати замовник. Продукт – результат (чи набір результатів) поставки по контракту.

Метою будь-якого проекту є отримання кінцевого продукту.

Всі проекти різні. Не існує ідеальної системи управління проектами, що підходить для кожного з видів проектів. Також не існує системи, яка б підходила кожному керівнику і була зручна для всіх членів команди. Однак за час існування проектного управління було створено чимало ефективних підходів, методик і стандартів, які можна взяти на озброєння. Розроблені підходи сильно відрізняються один від одного. Вони розрізняються по областях застосування, деталізованості, самодостатності і формалізації. У заголовку ми назвали їх «методами» для зручності, але насправді в статті представлені стандарти, концепції, методи і фреймворки, які застосовуються в управлінні проектами. В данний час найбільш популярними методологіями є:

- Класичне проектне управління

- Agile

- Scrum

- Lean

- Kanban

- Six Sigma

- PRINCE2

1.3.1 Класичне проектне управління

Найбільш очевидний спосіб зробити свій проект більш керованим – це розбити процес його виконання на послідовні етапи. Саме на такий лінійній структурі базується традиційне проектне управління. У цьому сенсі вона нагадує комп'ютерну гру – можна перейти на наступний рівень завершивши попередній. Схема робочого процесу наведена на рисунку 1.4.



Рисунок 1.4 - Класичний підхід до проектного менеджменту

Даний підхід орієнтований на проекти, в яких є суворі обмеження щодо послідовності виконання завдань.

Зазвичай виділяють 5 етапів класичного проектного управління, але можна додавати і додаткові етапи, якщо того вимагає проект.

Етап 1. Ініціація. Керівник проекту та команда визначають вимоги до проекту.

Етап 2. Планування. На даному етапі команда вирішує, як вона буде досягати мети, поставленої на попередньому етапі.

Етап 3. Розробка. Ця стадія реалізується не для всіх проектів — як правило вона є частиною фази планування. У фазі розробки, характерною для технологічних проектів, визначається конфігурація майбутнього проекту та/або продукту і технічні способи його досягнення.

Етап 4. Реалізація і тестування. На цій фазі відбувається власне основна робота по проекту – написання коду, зведення будівлі тощо. Слідуючи розробленим планам починає створюватися зміст проекту, визначений раніше, проводиться контроль за обраним метрик. У другій частині даної фази відбувається тестування продукту, він перевіряється на відповідність вимогам Замовника та зацікавлених сторін. У частині тестування виявляються і виправляються недоліки продукту.

Етап 5. Моніторинг і завершення проекту. В залежності від проекту дана фаза може складатися з простої передачі Замовнику результатів проекту або ж з тривалого процесу взаємодії з клієнтами щодо поліпшення проекту і підвищення їх задоволеності, і підтримці результатів проекту. Останнє відноситься до проектів у сфері клієнтського сервісу та програмного забезпечення.

Завдяки тому, що класичний проектний менеджмент суворо прив'язаний до часу виконання завдань, як правило, заздалегідь визначеним на етапі планування, для реалізації проектів в рамках даного підходу відмінно підходять інструменти календарно-мережного планування. Найпоширенішим інструментом календарно-мережного планування є вже згадана раніше діаграма Гантта. Існує безліч інструментів для її побудови – від простих таблиць начебто Excel і Smartsheet до професійних програмних пакетів кшталт Microsoft Project і Primavera.

Перваги:

* вимагає від Замовника та керівництва компанії визначити, що ж вони хочуть отримати, вже на першому етапі проекту;
* підхід передбачає моніторинг показників і тестування, що абсолютно необхідно для реальних проектів різного масштабу.
* класичний підхід дозволяє уникнути стресів через наявність запасного часу на кожному етапі, закладеного на випадок яких-небудь ускладнень і реалізації ризиків.

Недоліки:

- нетолерантність до змін;

- жорсткі рамки часу.

1.2.2 Scrum

Гнучкий фреймворк, створений в 1986 році, вважається самим структурованим з сімейства Agile. Створений в 1986 році, він поєднує в собі елементи класичного процесу і ідеї гнучкого підходу до управління проектами. У підсумку вийшло дуже збалансоване поєднання гнучкості і структурованості.

Слідуючи заповітам Agile, Scrum розбиває проект на частини, які відразу можуть бути використані Замовником для отримання цінності, звані заделами продуктів (product backlog). І незважаючи на те, що «зачепив продукту» - досить вірний переклад і використовується у фаховій літературі, в російській практиці найчастіше використовується просто «беклог». Потім ці частини пріоретізіруются Власником продукту - представником Замовника в команді. Найважливіші «шматочки» першими відбираються для виконання в спринті - так називаються ітерації в Scrum, що тривають від 2 до 4 тижнів. В кінці Спринту Замовнику видається робочий інкремент продукту - ті найважливіші «шматочки», які вже можна використовувати. Наприклад, сайт з частиною функціоналу або програма, яка вже працює, нехай і частково. Після цього команда проекту приступає до наступного спринту. Тривалість у Спринту фіксована, але команда вибирає її самостійно на початку проекту, виходячи з проекту і власної продуктивності. Принцип роботи представлено на рисунку 1.5.

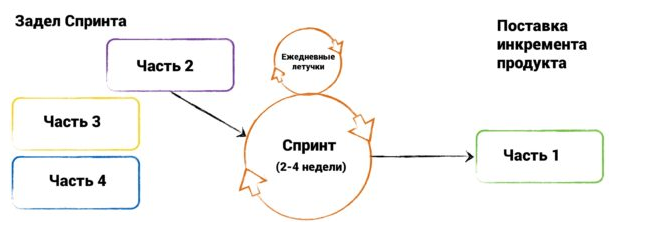


Рисунок 1.5- Методолігя Scrum

Структура процесів Scrum обертається навколо 5 основних зустрічей: упорядкування беклога, планування Спринту, щоденних летючок, підведення підсумків Спринту і ретроспективи Спринту.

1. Зустріч з упорядкування беклога (Backlog Refinement Meeting, «Backlog Grooming»): Ця зустріч аналогічна фазі планування в класичному проектному управлінні, і проводиться в перший день кожного Спринту. На ній розглядається - що вже було зроблено за проектом в цілому, що ще залишилося зробити і приймається рішення про те, що ж робити далі. Власник продукту визначає, які завдання на даному етапі є найбільш пріоритетними. Даний процес визначає ефективність Спринту, адже саме від нього нього залежить, яку цінність отримає Замовник за підсумками спринту.

2. Планування Спринту: Після того, як Власник продукту визначив пріоритети, команда спільно вирішує, що ж конкретно вони робитимуть під час прийдешньої ітерації, як досягти поставленої на попередній зустрічі мети. Команди можуть застосовувати різні інструменти планування і оцінки на даному етапі, аби вони не суперечили принципам і логіці Scrum. Планування Спринту проводиться на самому початку ітерації, після Зустрічі з упорядкування продукту.

3. Щоденні летучки: Кожен день спринту, в ідеалі, в один і той же час, члени команди витрачають 15 хвилин на те, щоб поділитися інформацією про статус завдань і стан проекту. На ній не відбувається обговорень проблем або прийняття рішень - якщо після зустрічі виникають питання і конфлікти, Scrum Майстер і залучені учасники обговорюють їх окремо. Летучка ж потрібна для обміну інформацією і підтримки всіх членів команди в курсі стану проекту.

4. Підведення підсумків Спринту: Мета етапу - обстеження та адаптація створюваного продукту. Команда представляє результати діяльності всім зацікавленим особам. Основне завдання - переконатися, що продукт етапу відповідає очікуванням учасників і узгоджується з цілями проекту.

5. Ретроспектива Спринту: Проводиться відразу після Підведення підсумків спринту і до планування наступного спринту. На ньому команда з'ясовує, наскільки чітко і злагоджено проходив процес реалізації етапу. Обстеженню піддаються виникли проблеми в роботі, методології та взаємодії. Саме цей етап дозволяє команді провести рефлексію і наступний Спринт провести ефективніше.

1.3.4 Kanban

Створений інженером компанії Toyota Тайічі Воно (Taiichi Ono) в 1953 році, Kanban дуже схожий на схему промислового виробництва. На вході в цей процес потрапляє шматочок металу, а на виході виходить готова деталь. Також і в Kanban, інкремент продукту передається вперед з етапу на етап, а в кінці виходить готовий до постачання елемент.

Крім того, творець Kanban надихався супермаркетами, а саме їх принципом - «тримай на полицях тільки те, що потрібно клієнту». А тому в Kanban дозволяється залишити незакінчену завдання на одному з етапів, якщо її пріоритет змінився і є інші термінові завдання. Невідредагована стаття для блогу, підвішена без дати публікації або частина коду функції, яку можливо не включатимуть в продукт - все це нормально для роботи по Kanban (рис.1.6).



Рисунок 1.6 – Методологія Канбан

Kanban набагато менш суворий, ніж Scrum - він не обмежує час спринтів, немає ролей, за винятком власника продукту. Kanban навіть дозволяє члену команди вести кілька завдань одночасно, чого не дозволяє Scrum. Також ніяк не регламентовані зустрічі за статусом проекту - можна робити це як Вам зручно, а можна не робити взагалі.

Для роботи з Kanban необхідно визначити етапи потоку операцій (workflow). У Kanban вони зображуються як стовпці, а завдання позначають спеціальні картки. Картка переміщається по етапах, подібно деталі на заводі, що переходить від верстата до верстата, і на кожному етапі відсоток завершення стає вище. На виході ми отримуємо готовий до постачання замовнику елемент продукту. Дошка зі стовпцями і картками може бути як справжній, так і електронної - навіть тут Kanban не накладаються ніяких обмежень на користувачів.

Ваша власна система Kanban може бути настільки гнучкою, наскільки Ви самі того забажаєте - адже багато в чому Kanban є візуалізацією ідеї Agile. Але у Kanban є 4 стовпи, на яких тримається вся система:

1.Картки: Для кожного завдання створюється індивідуальна картка, в яку заноситься вся необхідна інформація про завдання. Таким чином, вся потрібна інформація про завдання завжди під рукою.

2. Обмеження на кількість завдань на етапі: Кількість карток на одному етапі строго регламентовано. Завдяки цьому відразу стає видно, коли в потоці операцій виникає «затор», який оперативно усувається.

3. Безперервний потік: Завдання з беклога потрапляють в потік в порядку пріоритету.

1.4 Методи пошуку рекомендації оптимальної методології на основі експертних систем

Продукційна модель представлення знань є однією з найпоширеніших . Представлення знань за допомогою правил-продукцій має в деяких відносинах подібність із правилами виводу логічних моделей. Це дозволяє за допомогою продукцій виконувати ефективний вивід і, крім того, завдяки природній аналогії процесу міркувань людини дані моделі наочніше відбивають знання.

Правила забезпечують формальний спосіб представлення рекомендацій, вказівок або стратегій. Вони ідеально підходять в тих випадках, коли знання предметної області виникають з емпіричних асоціацій, накопичених за роки роботи по вирішенню завдань у тій чи іншій галузі.

Структура продукційного правила може бути описана наступним чином:

<ЯКЩО А, ТО В>,

де А – передумова (посилка чи антецедент), В – висновок (дія чи консеквент). Передмову та висновок виражаюсь через факти, якими описують предметну область.

Як умова, так і дія правила можуть враховувати кілька фактів, об'єднаних логічними зв'язками І, АБО, НІ. Логічний висновок полягає у використанні правила modus ponens: якщо відомо, що істинним є твердження А, та існує правило виду «Якщо А, то В», тоді твердження В також є істинним.

У загальному випадку продукціонна система включає наступні компоненти:

– базу продукційних правил;

– базу даних (робочу пам’ять);

– інтерпретатор.

Множину продукційних правил утворить базу правил, кожне з який представляє відособлений фрагмент знань про розв’язувану проблему. Психологи називають такі фрагменти чанками (від англ. chunk). Вважається, що чан – це об’єктивно існуюча одиниця знань, що вигляділяється людиною в процесі пізнання навколишнього світу.

Передумова правила часто розглядається як зразок. Це деяка інформаційна структура, що визначає узагальнену ситуацію (умова, стан і т.п.) навколишньої дійсності, при якій активізується правило. Робоча пам’ять (глобальна база даних) відбиває конкретні ситуації (стану, умови), що виникають у зовнішньому середовищі. Інформаційна структура, що представляє конкретну ситуацію зовнішнього середовища в робочій пам’яті, називається образом.

Інтерпретатор реалізує логічний вивід. Процес виводу є циклічним і називається пошуком за зразком. Розглянемо його в спрощеній формі. Поточний стан предметної області, що моделюється відбивається в робочій пам’яті у вигляді сукупності образів, кожний з яких представляється за допомогою фактів. Робоча пам’ять ініціалізується фактами, що описують задачу. Потім вибираються ті правила, для яких зразки, що представляються передумовами правил, порівнянні з образами в робочій пам’яті. Данні правила утворюють конфліктну множину. Усі правила, що входять у конфліктну множину можуть бути активізовані. Відповідно до обраного механізму дозволу конфлікту активізується одне з правил. Виконання дії, що міститься у виводу правила, приводить до зміни стану робочої пам’яті. Надалі цикл керування виведенням повторюється. Зазначений процес завершується, коли не виявиться правил, передумови яких порівнянні з образами робочої пам’яті (рис.1.7).

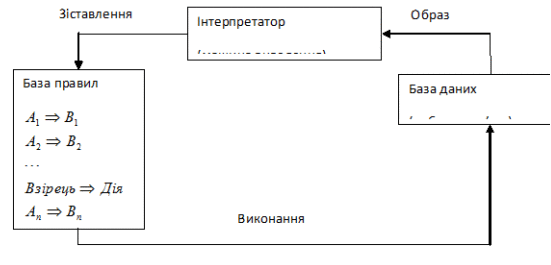


Рисунок 1.7 – Структура продукційної системи