



**RUP өңдеу құрамын
анықтайтын халықаралық
және
мемлекеттік стандартты
құжаттары**

RUP келесі негізгі принциптарды орындау үшін қолданылады:

- Ертерек идентификациялау және үздіксіз негізгі қауіптерді жою (жоба аяқталмастан бұрын).
- Орындалатын программаға қойылатын тапсырыс берушінің талаптарын орындауға концентрациялау (прецеденттер (немесе қолдану варианттары) моделін құру).
- Талаптардағы өзгерістерді, жобалық шешімдерді күту, және өңдеу процессіндегі олардың таратылулары.
- Компоненттік архитектура, жобаның ертерек кезеңдеріндегі таратылатын және тестіленетін архитектурасы.
- Жобаның (өнімнің) барлық өңдеу кезеңдеріндегі сапасын тұрақты қамтамасыздандыру.

RUP технологиясы, оның фазасы және өңдеу құжаттары, құрал-саймандары.

Rational Unified Process Қазіргі заманғы ең танымал әдістемелердің бірі. Көптеген өнімдерін қолдау үшін Rational Software компаниясымен жасалған (атақтылардың арасында - Rational Rose және Requisite Pro). RUP үш белгілі адамдар – Гради Буч, Ивар Якобсон және Жеймс Рамбомен (Rumbaugh) жасалған.

RUP итеративтілігі, кез келген қазіргі ілгері процесстер сияқты, итеративті болып табылады. Бұл демек, жобаның жасауы бірнеше итерациялардың алдында болады. Әрбір итерация соңында жұмыс істейтін өнімнің пайда болады, бірақ функционалы толық емес. Келесі итерацияларында функционалды істеп бітіреді және соңында соңғы толық дайын өнім пайда болады. Итератив процессінің идеясын төмендегідей көрсетуге болады. (1.сурет)



Плюстар итератив өңдеуінде көп. Әрбір итерацияда тестіленетін релиздердің көбісі түпкі өнімнің сапасына ықпал етеді. Егер керек болса, ерте кезеңдерге қолданушылардың күтуін тексеруге болады және өнімге өзгеріс енгізуге болады. Бұдан басқа, жобаны жоспарлау оңай келеді, өйткені, бірінші итерациядан кейін барлығы анықты болады және жоба басқарушысы келесі итерациялардың аяқталуын нақты болжамдармен айта алады. RUP-та процесстің түзу итеративтілігі туралы тікелей айтылмаған. Демек, RUP-ты сатылардың бәрі бірінен кейін бірі жүретін және дайын өнім ең соңында шығатын құламасы процесс үшін де қолдануға болады. Сондықтан RUP настройкасында итеративтілікті дұрыс ендіруі керек

Қолданушылардың сценарилары. (Use Case) қолданушының сценарии - бұл нақты бір операцияның орындалу кезіндегі қолданушының тізбекті әсерлер сипаттамасы. Мысалы, жаңа құжатты ашуы үшін қолданушының сценарийін жазуға болады және тағы сол сияқтылар. RUP қолданушылардың сценариларымен жүргізіледі (немесе прецеденттермен). Қолданушылардың сценарилары өңдеушілерге жүйенің не істеуі және оны қалай істеу керектігін көрсетуге мүмкіндік береді. Қолданушының сценарилары жүйенің функционалдық спецификациялары бөлігі болып табылады. Мұндай сценарилар тіпті бағдарлама әзірлеуде өте пайдалы, өйткені

- тапсырма берушіге түсінікті және тапсырма берушімен және өңдеушінің арасындағы түсінетін ортақ тіл және бас байланыстыратын буын ретінде қызмет көрсетеді
- бағдарламаның логикасында қателерді ерте кезеңде табуға көмектеседі
- тапсырма берушінің бағдарламаға берген талаптарын нақты анықтауға көмектеседі
- интерфейс жасаудың базасына және тестілеу сценариларын жазуға

Use-case driven процесі RUP-та қолданушылардың сценариларында құрметті орын алады

RUP құрылымы. Процесс төрт фазадан тұрады: 1. Зерттеу (Inception) 2. Жоспардың түзетуі (Elaboration) 3. Құрастыру (Construction) 4. Жазу (Transition). Фазалардың әрқайсыларында негізгі ықылас әртүрлі процесстерге түседі. Зерттеулер фазасында талаптардың жиыны және талдауы жүргізіледі, жоспардың түзетуін фазасында - талаптардың талдауы және жүйенің жобалауы жүргізіледі, құрастыру фазасында - өңдеу және кодтау, жазу фазасында - тестеу және тарату. RUP әдістемесі 9 негізгі ағындарға негізделген: 1) Бизнес- талдау (қажеттіктердің талдауы); 2) талаптардың жиыны және талаптармен басқару (функционалдық спецификацияларға талаптар аудармасы); 3) талдау және пішіндеу (программалық үлгіге талаптарының аудармасы); 4) кодтау; 5) тестілеу (бағдарлама талаптарға сәйкестігін тексеру); 6) кескінмен басқару және өзгерту (өнімнің әртүрлі болжамдарындағы өзгерістерінің зерттеп отыруы); 7) жобаларды басқару; 8) өңдеу әртүрлі жобалар және өнімнің кодтауы; 9) жазу (өнімнің сәттілігіне

1. **Бастапқы кезең (Inception)**

- ▶ Жоба шекарасы және көрсетілімдері қалыптастырылады.
- ▶ Экономикалық негіздемелері құрылады.
- ▶ Өнімнің негізгі талаптары, шектеулері және кілттік функциональдылықтары анықталады.
- ▶ Прецеденттер моделінің базалық версиясы құрылады.
- ▶ Қауіптер бағаланады.
- ▶ Бастапқы кезең аяқталмас бұрын мақсаттың өмірлік циклының кезеңдерінің жетістіктері бағаланады

2. **Нақтылау кезеңі (Elaboration)**

Нақтылау кезеңінің фазасында пәндік аймақ талдауы жүргізіледі және орындалатын архитектураның құрылуы орындалады. Ол келесілерден тұрады:

- ▶ Талаптарды құжаттау (барлық прецеденттерді толық сипаттаудан тұрады).
- ▶ Жобаланған, таратылған және тестіленген орындалатын рхитектура алу.
- ▶ Жаңартылған экономикалық негіздемелер және нақты мерзімдері мен құны.
- ▶ Төмендетілген қауіптердің негіздемесі.
- ▶ Нақтылау кезеңінің дұрыс орындалуы архитектураның өмірлік циклын алуға мүмкіндіктер береді

► 3. Құрастыру кезеңі (Construction)

Құрастыру фазасында өнімнің функциональды бөлігінің таратылуы орындалады. Құрастыру фазасы жүйенің бірінші ішкі таратылуын және бастапқы функциональды дайындығын көрсетумен аяқталады

► 4. Ендіру кезеңі (Transition)

Ендіру фазасында өнімнің ең соңғы таратылатын версиясы құрылады және өндірушіден тапсырыс берушіге ұсынылады. Бұл кезең бета-тестілеу, пайдаланушыларды үйрету және өнімнің сапасын анықтаудан тұрады. Егер сапасы бастапқы фазаларда жасалған, пайдаланушылармен күтілген талаптарды немесе критерилерді қанағаттандырмаса онда ендіру қайта қайталанады. Ал барлық талаптардың дұрыс орындалғаны өңдеудің толық циклының аяқталғандығын білдіреді (Product Release).

RUP-та кез келген жоба төрт фазаны өтеді. Бұл фазалардан кейін барлық тоғыз ағындардан да өтеді. Әрбір фаза, өз кезегінде, итерацияға бөлінеді. Мысалы, егер "Зерттеу" фазасында бірінші итерацияны алсақ, онда негізгі ықылас бұл бизнес- талдау, талаптардың жиыны және пішіндеу, бірақ кодтау да бар. Егер "Құрастыру" фазасында соңғы итерациялардың бірін алсақ, онда негізгі ықылас кодтауға, тестілеуге және кескінмен басқаруға түседі. Басқа сөзбен айтқанда, жобаның даму барысында әрбір итерацияға екпіндер ауыстырылады. Бұл дұрыс, өйткені соңында талдайтын зат болмайды, ал талаптарды жинау кеш болады. (Artefact) артефакт – ПҚ жасау кезінде жасаланатын және қолданылатын өнім. Мысалы, артефакттарға құжаттар, модельдер, бастапқы кодтар жатады. Артефакттар мысалдары: UML класстарының диаграммасы, қолданушыға арналған нұсқау, тағы сол сияқтылар. RUP-тың ажырамас бөлігін **артефакттар** және **рөлдер** құрайды. Бағдарлама әзірлеулер кезінде әртүрлі артефакттар жасалады, және әр артефактқа нақты рөл жауап береді. Мысалы, класстар диаграммасын "Сәулетші" құрайды, тестілеу сценарийін "Тесттер дизайнері" жазады. Барлық визуалды пішіндеу CASE- құралдары арқылы іске асады. Оның негізгі қызмет ететін тілі UML (Unified Modeling Language), ол таң қалатындай емес, өйткені UML тілі RUP-тың авторларымен өңделген.

Итеративті өңдеу ерте кезеңде жұмыс істейтін өнім түрін алуға мүмкіндік береді және критикалық кемшіліктерін айқындайды, одан басқа, қорытынды өнім сапасы артады, өйткені база өнім қанша итерациядан өтсе, сонша рет тестілейді.

Талаптармен басқару – орташа өнімдерді өндіру арасында маңызды процесстердің бірі. Осының арқасында өнім тапсырыс берушінің күткеніндей болып шығады. Аспапты қолдау Requisite Pro қамтамасыз етіледі.

Модульдік архитектура теорияда кодты қайтадан пайдалануға мүмкіндік береді және жүйе ыңғайлы болады. Іс жүзінде жүзеге асыру мүмкін емес.

Визуалды модельдеу жүйелердің өсетін күрделілігімен белдесуге тиімді мүмкіндік береді. Модельдер жүйенің негізінде жұмыс істеуін түсінуге көмектеседі, яғни оның не істейді және қалай істейді. Одан басқа, өндүшілер арасында моделдер коммуникация құралдары болып табылады, бірақ олар барлығына түсінікті болу керек. Сол себепті RUP UML-де қолданылады, өндірушілерге бір тілде сөйлесуге мүмкіндік береді. Аспапты қолдау Rational Rose

RUP әдістемесі негізгі 9 ағынға негізделіп жасалады

- 1) Бизнес-талдау (керектіктің талдануы);
- 2) Талаптарды жинау және талаптарды басқару (талаптарды функционалдық спецификацияға ауыстыру);
- 3) Талдау және моделдеу (талаптарды программалық моделге ауыстыру);
- 4) Кодтау;
- 5) Тестілеу (программаның берілген талаптарға сәйкестігін тексеру);
- 6) Өзгертулерді және конфигурациясын басқару (өнімнің әртүрлі версияларындағы өзгерулерін тексеру);
- 7) Жобаны басқару ;
- 8) Өңдеу ортасын ұстану және құру;
- 9) Ашып қарау (өнімді беру немесе сату үшін керектінің барлығы)

Басы - бастапқы идея бойысында зерттеу фазасына өтпелі шешім қабылдау туралы (жобаның қатысушылары кем дегенде, көздің нүктесімен) жеткілікті дәлел беретін процесстің бірінші кезеңі.

Зерттеу - бұл процесстің екінші фазасы; өнімнің көрінуі және оның архитектурасы осы зерттеу арқылы анықталады. Негізгі ықылас басымдылықтардың жүйеге және талаптар қоюына конкретизациясын білдіреді.

Құрастыру - процесстің үшінші фазасы болып табылады. Атқарылатын сәулеттік түп тұлға ол қолданушыларға елестете алған формаға ие болады. Жүйеге, және бағалау белгісіне талаптар кәзіргі жағдайда, өзгертін қажеттіктермен сәйкес қайта қарауларға душар болады, белгінің кішірейтуі үшін қажетті қорлар адыраяды.

Енгізу – программалық қамтама құру процессінің төртінші кезеңі. Бірақ өңдеу ережесі аяқталмаған, тіпті анығында осы жүйе фаза бойында үздіксіз мінсіздікке жетеді, қателер түзетіледі және алдыңғы үлгілеріне енбеген функционалды мүмкіндіктері қосылады.

Барлық төрт фазада да программалық қамтаманы өңдеу әдісін ұйымдастыруды сипаттау үшін сипатталған элемент бар болып табылады, ол – интергация. Интеграция деп бағалау критерийлері мен анық құрылған жоспармен қоса нақты түрде анықталған әрекеттің тізбектілігі айтылады. Бұл әрекет ішкі және сыртқы қолданыс үшін жаңа үлгі құруға алып келеді. Бұл дегеніміз, өңдеудің өмірлік циклы жүйе архитектурасын жүзеге асыратын орындалатын үлгілердің үздіксіз ағынын көрсетеді.

Жобалаудың әдістері және программалардың өмірлік циклын қамтамасыз ету.

Қолданбалы стандартты қолдану - 5 негізгі (тапсырыс, жабдықтау, өңдеу, пайдалану, бақылап отыру)

8 қосалқы (кескіндермен құжаттау, басқару, мәселелердің сапаны қамтамасыз ету, верификация, іскерлік сараптау, бірлескен талдау, тексеру, шешімі)

4 ұйымдастыру процесстері (инфрақұрылымның жасауы, басқару, жетілдіру, үйрену)

Өңдеу программасы - талаптардың талдау, жобалау, программалау, құрастыру, тестеу, іске қосу, қабылдау.

Керемет тәжірибелер RUP-тың өзі алты ең жақсы тәжірибелерден (best practices тұрақтанады:

- ▶ Итеративті өңдеу
- ▶ Талаптармен басқару
- ▶ Модулдік сәулеттерді (архитектура) қолдану
- ▶ Визуалды моделдеу
- ▶ Сапаны тексеру
- ▶ Өзгерістердің зерттеп отыруы

Олар RUP-тың тікелей бір бөліктері болып табылмайды, бірақ олар процесстің күйге келтіруінде (настройка) дұрыс қарауға ұсынылады

RUP объективті-орналастырылған технологияны қолдайды. Көп визуалды үлгілер объективті- орналастырылған үлгілермен болып табылады, объекттердің тұжырымдамаларында, олардың арасындағы класс және қатынастар. Сонымен бірге Unified Modeling Language (UML) ортақ тіл болып табылады.

RUP жүйенің компонентті өңдеуін қамтамасыз етеді. Компоненттермен қарапайым емес модулдар болып есептеледі, ішкі жүйелері функционалдықпен жүйеге агрегаттала алады.

RUP құрылған жүйенің сапасының кепілдігі сияқты барлық құрылған материалдардың жобаларындағы сапаны бақылауына бағытталған. Сапаның бағасы әдістемеліктің процесстерінде орналастырылған.

RUP кәсіпорынның масштабының (ИЖ) күрделі ақпараттық жүйелерінің жасауында қолданылады және Rational Software-нің жобаның үстінде командалық жұмыс қамтамасыз ететін аспапты құралдарымен іске асады.

Программалық өнімдердің өңдеуінде қолданылатын халықаралық және отандық стандарттар. Өңдеуді анықтайтын сапаның ИСО стандарты. ИСО 4001-96. Сапа жүйелері. Үлгі жобалауда сапаны қамтамасыз етуді. ИСО МЭК 9126-93. ИТ. Программалық өнімнің бағасы. Сапалар және олардың қолдануына бойынша нұсқау мінездеме. ИСО МЭК 8402-94. Сапаны басқару және сапаны қамтамасыз ету. Сөздік. 34.601-90, 34.603-92, ИСО4001-96 және т.б стандарттар.

Аспапты құралдардағы стандарттарының іске асуы. Өңдеулердің визуалды жиын элементтерінің даярлаулары. Элементтердің кітапханаларының құрамның жедел күйге келтіруінің мүмкіндіктері. Жобалаудың болмаған үйреншікті элементтерінің қосымшалары. Өзгерістердің жедел қимылдаушылығы.

Ақпараттық қауіпсіздіктің стандарттары. P511.88-98. Ақпаратты қорғау. Компьютер вирустерінің бар болуына программалық құралдарының сынауы. P512.41- 98. Құралдар және бақылау жүйелері мен рұқсатты басқару. Классификация. Жалпы техникалық талаптар. Сынау әдістері. Рұқсат етілмеген басқаруды қорғаудың әдістері. Қолданылатын стандарттардың маңызы. Немесе басқару