**Тема 17. Организационно управление на софтуерни проекти. Човешки ресурс.**

**1. Организационно управление**

**1.1. Основни понятия**

Всеки **проект** има определена цел, която трябва да се постигне с крайни и ограничени ресурси. **Целта** на управлението на проекти е те да се осъществяват систематично и контролирано чрез постъпкови процедури. **Ефективни проекти** са тези, които водят до повишаване на производителността и осигуряване на качествени крайни продукти. Предпоставки за тях са:

* създаване на благоприятна среда за работа;
* формулиране на ясни, точни и непротиворечиви изисквания за работата;
* управляване на проекта чрез планиране и контрол;
* прилагане на ефективни, проверени и стандартизирани процедури;
* осигуряване на адекватни ресурси;
* мотивиране на участниците в проекта;

**1.2. Управление на софтуерни проекти**

**1.2.1. Методология на управлението на проекти**

1) Управлението на човешките ресурси - признаване на ролята на човешкия фактор в разработването и използването на софтуера;

2)Управлението на продукта - определяне на предназначението, основните функции и изисквания към новия продукт;

3)Управлението на процеса на разработване - включва планиране и ефективно осъществяване на всички дейности.

4)Управлението на проекта – основната цел е да се държат нещата под контрол. Управлението на проекти е в съответствие с избрана **методология** — съвкупност от принципи, подходи, методи, процедури, правила и указания. Методологията е **формална**, ако е представена в писмен вид и се прилага систематично в дадена организация. Преимуществата на прилагане на формална методология са, че тя:

* подпомага избирането на проекти;
* позволява разпределение на ресурсите и контролирането на риска;
* управлява проекта чрез дефиниране на контролни точки;
* улеснява комуникациите между участниците;
* подпомага счетоводно-финансовите дейности;
* осигурява създаване на необходимата документация по време на проекта;
* подобрява качеството на създавания ПП;
* намалява разходите за разработване;
* улеснява натрупването на статистическа информация;
* дава възможност за сравнителен анализ на различни проекти;
* осигурява систематичност и управляемост на проектите.

**1.2.2. Управленческа структура**

Традиционната управленска структура е йерархична, като нивата на йерархията. Има 2 нива:

1)Ръководителите на фирмата отговарят за:

* определяне на дългосрочните и краткосрочни бизнес програми;
* сключване на договори за разработване;
* назначаване на ръководители на проекти;
* ръководство на основни фирмени дейности;

2)Ръководителите на проекти отговарят за:

* подготовката и съставянето на офертата за разработката;
* оценка на стойността и продължителността на проекта и необходимите за осъществяването му ресурси;
* сформиране на екип(и) за проекта и оценяване на работещите;
* съставяне на планове и графици за проекта;
* текущо управление на проекта;
* представяне на проекта и развитието му пред външни инстанции;

**1.3. Планиране**

Планирането обхваща дейностите по съставяне и проследяване на всички видове планове.

Недостатъците на планирането са:

* че отнема много време;
* че е скъпо;
* ограничава гъвкавостта;
* че се основава на предположения, които може и да не са верни;
* че потиска творческия подход;
* и доколкото всеки проект е уникален, не може да се използва рационално опитът от предишни проекти;

 Преимуществата на планирането са, че то:

* подпомага осъществяването на интегрирани, целенасочени действия и
резултати;
* служи за координация и контрол на всички дейности;
* увеличава ефективността, като предотвратява повторно извършване на определени действия;
* осигурява ,прозрачност" на управлението на проектите.

В зависимост от целите и предназначението си плановете се делят на общи и конкретни.

Общите планове се отнасят до софтуерната организация като цяло и са целева програма, стратегически и тактически план.

 Целевата програма отговаря на въпроса, защо съществува софтуерната организация. Най-често целта е разработване на софтуер, който да се разпространява с цел да се реализира печалба.

Стратегическият план отговаря на въпроса, какво да създава софтуерната организация. Този план се създава за относително по-дълъг период от време и определя в какви приложни области ще бъдат разработваните ПП, дали и как ще се разширява дейността на организацията, какви ще са нейните основни и странични дейности.

Тактическият план отговаря на въпроса, как ще се осъществяват планираните софтуерни проекти. Той обикновено се съставя за една календарна година.

Конкретните планове се отнасят до отделен софтуерен проект. Освен плана за проекта се съставят бюджет, календарни планове и графици за групите, реализиращи проекта, както и индивидуални планове за всеки участник. Съществен конкретен план е бюджетът. Съставянето му се подчинява на следните правила:

* съставя се от ръководителя на проекта със съдействието на финансист;
* представлява писмен документ, който се утвърждава от ръководството на фирмата и спазването му е задължително;
* бюджетът описва необходимите средства и начина на осигуряването им. Разходите са описани по видове, наречени пера. Основни пера са: за трудови възнаграждения, за закупуване или наемане на техника, за поддържане на средата за разработване, за консумативи, за командировъчни, за допълнителни услуги .Обикновено средствата са „боядисани" и не се допуска безконтролното им прехвърляне от едно перо в друго.
* ако проектът се осъществява в условия на инфлация и ще продължи повече от година, се препоръчва съставянето му в някакви условни единици;
* препоръчва се да се предвидят контролни точки от проекта, в които да се анализира изпълнението на бюджета и ако е необходимо, да се актуализира;

**1.4. Анализ и управление на риска**

Под **риск** се разбира потенциален проблем, който може да възникне и да застраши успешното завършване на проекта. В зависимост от това, какво засягат, рисковете се делят на рискове на проекта, технически или бизнес рискове. Класификация на рисковете е в зависимост от степента на прогнозируемост:

1) Очаквани рискове - могат да се идентифицират чрез анализ на проекта;

2) Предвидими рискове - могат да се опишат въз основа на опита от предишни проекти;

3) Непредвидими рисковете - просто се случват.

Основната идея на управлението на риска е да се анализират някои от най-вероятните рискови ситуации и да се направят опити за избягването им или за намаляване на последиците от тях. Управлението на риска се осъществява в три стъпки:

1)Идентифициране - систематичен опит да се установи какво може да застраши успешната реализация на проекта. Рисковите фактори могат да се разделят на две групи: общи за всички софтуерни проекти и специфични за даден проект. Основна техника за идентифициране на риска е съставянето на въпросници, свързани с характеристики на основните застрашени обекти.

2)Оценяване - данните за риска се систематизират и се преобразуват в информация за вземане на решения. За всеки риск се определя вероятността и последиците от риска , въз основа на които се определя и приоритетът му.

3)Предотвратяване на последиците - изисква планиране на управлението на риска. По време на разработването се следят идентифицираните рискови фактори и се натрупва информация за всеки от тях, за да се установи дали и как се променя вероятността за настъпване на рисковата ситуация.

**1.5. Съставяне на графици**

 Съставянето на графици минава през следните етапи:

1. Декомпозиране на всички дейности, необходими за реализацията на проекта.
2. Определяне на зависимостите между задачите.
3. Определяне на трудоемкостта на всяка задача в избрана мерна единица и дефиниране на начална и крайна дата за завършване.
4. Разпределяне на работата между членовете на групата.
5. Определяне на персонални отговорници за всяка задача.
6. Определяне на получаваните изходи от всяка задача.
7. Дефиниране на контролни точки, при достигането на които ще се анализират и оценяват получените междинни продукти.

**1.6. Проследяване на проекта**

 Проследяването на проекта става чрез регламентиране на система за контрол. Контролират се основните параметри на проекта – размер на междинните продукти, производителност на разработниците, цена на разработката, продължителност. Текущото състояние се сравнява с планираното и при несъответствие се планират коригиращи действия. Развитието на проекта може да се следи чрез т.н. „earned value analysis”, който се извършва в определени контролни точки. Той се базира на следните мерки:

* Бюджетна трудоемкост на задачата (**BCW**): прогнозната трудоемкост на всяка задача;
* Бюджетна трудоемкост на завършените до момента задачи(**BCWS**): сумата от прогнозната трудоемкост на тези задачи, които според графиката трябва да са завършени до момента.
* Бюджетна трудоемкост на проекта(**BAC**): сумата от всички BCWS и следователно, оценката за трудоемкост на проекта като цяло;
* Планирана стойност (**PV**): частта за всяка задача PV=BCW/BAC;
* Бюджетна трудоемкост за извършената работа (**BCWP**): сумата от реалните трудоемкости за задачите завършени до момента;
* Реална трудоемкост за извършената работа(**ACWP**): сумата от реалните трудоемкости на завършените задачи;

Могат да се изчисляват следните индикатори за развитието на проекта:

* Рентабилност (**EV**): EV = BCWP/BAC
* Индекс на изпълнение на графиката(**SPI**): SPI = BCWP/BCWS
* Вариране на графика(**SV**): SV = BCWP - BCWS
* Индекс на изпълнение(**CPI**): CPI = BCWP/ACWP
* Вариране на трудоемкост(**CV**): CV = BCWP - ACWP

**1.7. Примерен план на софтуерен продукт**

1. Въведение.

2. Оценка на проекта

3. Рискове за проекта

4. Управление на проекта чрез календарни планове и графици

5. Ресурси

6.Организация на екипа, работещ по проекта

7. Проследяване и контролиране на проекта

8. Приложения

**1.8. Препоръчителни мениджърски практики**

* Управлявай риска.
* Регистрирай междинните резултати и ги сравнявай с планираните.
* Регистрирай дефектите.
* Помни, хората за най-важния ресурс.
* Проследявай изискванията.
* Проектирай два пъти, програмирай веднъж.
* Проверявай изискванията и проектите.
* Осъществявай тестването като непрекъснат процес.
* Компилирай и тествай често. И др.

**1.9. Правила за софтуерни проекти**

1. Комуникации.
2. Определяне на крайния срок.
3. Възлагане на проект.
4. Подизпълнители.
5. Осигуряване на необходимата документация.
6. Финансова дисциплина.
7. Задължения на изпълнителя.
8. Конфиденциалност.
9. Публични изяви.
10. Индустриална и интелектуална собственост.
11. Независимост.
12. Работна група.
13. Проблеми.
14. Отчети.
15. Закъсняване на проекти.
16. Удължаване на срока.
17. Изменения на характеристики на проекта.
18. Ползване на отпуск.
19. Плащания.
20. Прекратяване на договора.
21. Особени ситуации.
22. Разрешаване на противоречия.

**2.Човешки фактор в софтуерното производство**

**2.1 Наемане на добри софтуерни специалисти**

При вътрешното набиране след определяне на длъжностните характеристики за вакантните позиции се разглеждат възможностите за преназначаване. Използваните техники могат да бъдат:

* поддържане на информация за всички служители ;
* вътрешен конкурс. Свободните места се обявяват и всички служители, които се интересуват, могат да подават съответните документи;
* назначаване на основа на препоръки .

При външното набиране целта е да се потърсят и назначат нови специалисти. Основни техники са:

* използване на връзки със съответните университети и привличане на най-способните завършващи студенти;
* използване на услугите на специализирани фирми за подбор на кадри ;
* примамване на висококвалифицирани ;

За избор на подходящи служители се предлага използването на т. нар. психологически профили. Психологическият профил е съвкупност от индивидуални личностни качества, за които се смята, че са важни за съответната професия. Не се препоръчва изборът на софтуерни специалисти да се основава изцяло на психологическия профил поради следните причини:

* личностният психологически профил се променя във времето;
* различни качества са ценни за различните позиции в софтуерното производство;
* тестовете за психологически профил могат да бъдат разучени ;

**2.2. Определяне на структурата и състава на работните места**

Идентифицирани са четири типа комуникативно поведение при изпълнение на групови дейности:

1. обособяващ - хората са с устойчивата нагласа към индивидуализъм. При разпределението на функциите на такъв член на групата трябва да се възлагат задачи, които са относително самостоятелни;
2. водим – хората имат подчертана нагласа към доброволно подчинение и силно изразено предпочитание към изпълнение на възложена работа под непосредственото ръководство на лидера на групата.
3. лидерски - комуникативното поведение на хората от този тип е изключително важно за планиране на дейността на групата. Психологическият профил на лидерите е с нагласа към доминиране и предполага доброволно подчинение на останалите.
4. сътрудничество – хората от този тип показват стабилен стремеж към колективно решаване на общата задача, бързо ориентиране към рационалните решения.

Числеността и разпределението на задълженията в групата зависят от размера и сложността на конкретния софтуерен проект. Препоръчва се да се определи съставът на групата чрез "длъжностни характеристики" за дейностите, които ще се извършват, и след това да се изберат членовете на групата с отчитане на:

* възможността и желанието да се извършва дадена дейност
* опит от реализацията на подобни приложения ;
* стил на работа.

**2.3. Управление на взаимоотношенията в работната група**

От психологическа гледна точка са определени две нива на взаимоотношенията — делови функционални контакти и междуличностни взаимоотношения. Някои задължения и права на членовете на групата са:

* да знаят какво се очаква от тях;
* да имат възможност да обсъждат възлаганите им задачи;
* да разполагат с необходимите им ресурси;
* да бъдат информирани за състоянието на проекта; и др.

Някои от задълженията на мениджъра са:

* да определя задачите и функциите ясно и точно;
* да дискутира проблемите и възможните решение с всички засегнати;
* да разработва ясни процедури и да ги описва с подходящо ниво на детайлизация;
* да оценява периодично работата на всеки участник; и др.

Могат да се разграничат четири вида управленски профила:

1. **насочващ**, когато мениджърът дава инструкции и следи за тяхното изпълнение;
2. **съветващ**, когато мениджърът обяснява решенията си и поощрява обсъжданията и даването на предложенията;
3. **подкрепящ**, когато мениджърът улеснява и поддържа усилията на подчинените си и се чувства отговорен за вземаните от тях решения;
4. **делегиращ**, когато мениджърът прехвърля на подчинените си отговорността за вземане на решения при възникващите проблеми;

Основна функция на мениджъра е да организира изпълнението на определени задачи, насочвайки подчинените си и отчитайки мотивацията им. Фактори, определящи мотивацията, могат да бъдат:

* работата, която е възложена, и начинът, по който е възложена;
* стилът на контролиране и управление;
* хората, с които се работи;
* начинът на общуване с мениджъра;
* работната среда;
* организиране на конкретни проекти; и др.

Мениджърът носи отговорността за проекта като цяло, но той може да делегира част от правата и задълженията си на други членове на работната група.Преимущество на делегирането е, че мениджърът може да се съсредоточи върху същинските управленски функции, а членовете на групата са с повишена мотивация, защото се чувстват оценени, самостоятелни и отговорни.

**2.4. Организиране на комуникациите в групата**

Формите и интензивността на контактите зависят от:

* числеността на групата. За група от N членове възможните контакти саN(N-1)/2 ;
* структурата на групата;
* състава на групата;

Регламентираните формални контакти могат да бъдат:

1. **централизирани** - определя се координатор, който приема всички съобщения и решава към кои членове на групата да ги пренасочи. Координиращото лице може да поеме и допълнителни функции, като присвояване на приоритет на съобщенията, документиране и архивиране на разменяната информация.
2. **свободни** - всеки член на групата може да общува директно с всички останали. Недостатък на тази форма е големият поток неструктурирана информация, който изисква допълнително време за класифициране на съобщенията и може да забави обработката и очакваните като реакция действия.

**2.5.Организиране на сбирки и заседания**

а) планиране на сбирката - преди всичко трябва да се определи необходимо ли е провеждането на сбирката и тя да се свиква само ако проблемите не могат да се решат чрез индивидуални срещи, разговори по телефона или размяна на съобщения;

б) провеждане на сбирката – за ефективното протичане се определя водещ, който:

* открива сбирката, представяйки целите и начина на протичането й;
* ръководи сбирката;

в) закриване на сбирката - Сбирката се прекратява, когато свърши заплануваното време, когато се изчерпи дневният ред или когато групата няма ресурси да продължи. Задължително е резюмиране на постигнатите резултати и взетите решения и насрочване на следваща сбирка, ако е необходимо.

**2.6. Разработване на софтуера – teamwork**

 Съгласно CMMI-модела систематичното и управляемото разработване на качествен софтуер трябва да се осъществява от екипи, включващи специалисти с допълващи се знания, умения и опит, които си сътрудничат през целия ЖЦ на ПП. Сформирането на екип започва с идентифицирането на задачите, необходимите за тях знания и опит и назначаване на подходящите хора. Следващата стъпка е team-building – създаване на „екипност”, така че:

* ролята на всеки член на екипа да е точно определена;
* целите на екипа да са ясно формулирани;
* решенията да се вземат с консенсус;
* качеството на създаваните продукти и/или услуги да се подобрява; и др.

**2.7. Егрономика**

Основната цел на ергономиката е да изследва и да предложи подходяща работна среда, така че човешките ресурси да се използват по най-ефективен начин.От значение са:

* индивидуалното достъпно пространство;
* мебелировката;
* вътрешното оформление на помещението;
* контролираното ниво на шум и температура;
* чистотата и подредеността;
* наличието на помещение за почивка с достъп до кафе машини;

Работните места могат да бъдат в общо помещение или в индивидуални стаи за всеки разработчик. Двата вида организация са с различни възможности за контрол на работата и за комуникации. Освен характеристиките на физическата среда, от значение са и наличните комуникационни, хардуерни и софтуерни средства, предоставени на всеки участник в софтуерната разработка.

**2.8. PM-CMM модела за управление на човешките ресурси**

 PM-CMM модела описва прагматичен подход, целта на който е подобряване на цялостното функциониране на софтуерна организация чрез повишаване на производителността на най-важния управляем ресурс – хората.

**2.9. Професионализъм и етично поведение**

Проблемът за професионалната етика е дискутиран със софтуерни специалисти от цял свят. Много от тях не само са потвърдили, че такива случаи са често явление, но и смятат, че описаното поведение е оправдано в повечето случаи. Това не са просто примери за лоши управленски подходи, а посочване на по-сериозен проблем — липса на правила и стандарти за етично поведение. Във всяка организация се натрупват прецеденти и повтарящи се реакции и те се предават от по-опитните на пo-младите софтуерни специалисти. Такова поведение и стил на работа обикновено се улесняват от липсата на стандартизирано управление на софтуерните проекти, на процедури за оценка на качеството, оценка на разходите и др. По време на реализация на софтуерен проект вместо решения, основаващи се на това, което би било по-добро за клиента, много софтуерни специалисти избират решения, които биха улеснили самите разработчици или софтуерната разработка. Липсата на правилно етично поведение има огромно влияние върху развитието на организацията. Друго следствие от липсата на утвърден етичен правилник е деформирането на компютърния специалист като личност. Едно възможно решение е в софтуерната организация да се обсъждат проблемите от морално-етично естество и да се приемат правила за етично поведение, основани на:

* конфиденциалност;
* компетентност;
* неприкосновеност на интелектуалната собственост;
* недопускане на злоупотреба при работа с компютрите;

 Тези правила не могат да бъдат задължителни, но поне биха определили приемливо поведение.