

## Objektorientētas programmēšanas praktikums (studiju projekts)

Dr.sc.ing., prof. Oksana Ņikiforova  
DITF LDI  
Lietišķo datorzinātņu katedra  
Rīga - LV1048, Meža 1/3, 510.kab., tel.67 08 95 98  
[oksana.nikiforova@rtu.lv](mailto:oksana.nikiforova@rtu.lv)

## Kursa mērķi

- Nostiprināt studiju procesā apgūtās zināšanas programmatūras izstrādē,
  - darbojoties komandā izstrādes dzīves cikla posmos,
  - praktiski realizējot uzdotas sistēmas risinājumu programmatūrā,
  - izmantojot
    - objektorientētās tehnoloģijas principus,
    - programmatūras standartus un
    - objektorientētās programmēšanas vides un rīkus.
- Uzdevumi:
  - iegūt pieredzi programmatūras prasību specifikēšanā, lietojot UML lietošanas gadījumu notāciju un apraksta formu;
  - iegūt pieredzi programmatūras sistēmas arhitektūras definēšanā turpmākai sistēmas realizācijai objektorientētā programmēšanas vidē;
  - iegūt pieredzi sistēmas realizācijā atbilstoši iepriekš izstrādātai prasību specifikācijai un projektējumam

Oksana Ņikiforova

"Objektorientētas programmēšanas praktikums (studiju projekts)"

2.

## Trīs projekti

- RTU studentu konferences atbalsta sistēma
  - Projekta vadītāja - Tatjana Jakona
  - Tikai programmēšana
  - Ātra izstrāde -
    - Līdz 17.februārim - ieviests pirmais būvējums
    - Līdz 17.martam - ieviests otrs būvējums
    - līdz 17.aprīlim - ieviesta strādājoša sistēma
    - Teicamas MS Visual Studio (ASP.NET, C#)
  - 3-4 cilvēki - konkurss pirmā uzdevuma izpildē
  - SCRUM
- Bakalaura darba tēmas izvēles atbalsta sistēma
  - Projekta vadītāja vieta ir vakanta
  - PPS, PPA, realizācija, projekta izstrādes vides atbalsts
  - No integrēšanas ar ORTUS-u atkarīga tehnoloģiju izvēle
  - RUP
- Aizstāvēšanas komisijas darbības atbalsta sistēma
  - Projekta vadītājs - Vladislavs Nazaruks
  - PPS, PPA, realizācija, projekta vides atbalsts
  - Brīva tehnoloģiju izvēle
  - RUP

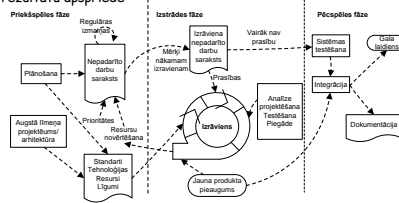
Oksana Ņikiforova

"Objektorientētas programmēšanas praktikums (studiju projekts)"

3.

## Spējas metodoloģijas

- SCRUM (1986, 1995: Ken Schwaber, Mike Beedle; [controlChaos.com](http://controlChaos.com))
  - Objektorientēta izstrāde
  - Projekts dalās iterācijās (sprints) 30 dienas katrā - viena sprinta gaitā tiek realizēts noteikts sistēmas funkcionēšanas bloks ar nosacījumu, ka prasības netiek mainītas
  - Katru dienu komandā savācas uz 15 minūtēm (scrum), kurā notiek problēmu un rezultātu apspriede



Oksana Ņikiforova

"Objektorientētas programmēšanas praktikums (studiju projekts)"

4.

## SCRUM dzīves cikls: Pirms-spēles fāze

- **Plānošana** - augsta līmeņa sistēmas vadāmības radīšana, kas balstās uz mūsdienās pazīstamām produkta prasībām un īpašībām kopā ar tā izmaksām un termiņiem.
  - Veidots produkta nepadarīto darbu saraksts (angl. Produkt backlog), kurā nonāk visas prasības no klienta.
  - Prasības prioritizētas un novērtētas piepildes, kas nepieciešamas šo prasību realizācijai.
    - Šis saraksts pastāvīgi atjaunots ar jauniem datiem no klienta, attiecīgi mainās arī uzdevumu prioritātes un to izpildes izvērtējums.
  - Projekta komandas noteikšana, tehnoloģiju un instrumentu izvēle, risku izvērtēšana un vadības aktivitātes.
  - Katras jaunās iterācijas gaitā projekta komanda pārskata atjaunoto darba sarakstu, nosakot uzdevumu apjomu nākošai iterācijai.
- **Augsta līmeņa projektēšana** - plānota sistēmas arhitektūra, pamatojoties uz prasībām produkta nepadarīto darbu sarakstā.
  - Esošās sistēmas modernizācijas gadījumā, pie nepieciešamo izmaiņu un jaunievedumu analīzes, tiek analizēti arī riski, kas var izraisīt šīs izmaiņas.

Oksana Ņikiforova

"Objektorientētas programmēšanas praktikums (studiju projekts)"

5.

## SCRUM dzīves cikls: Izstrādes (spēles) fāze

- Izstrāde notiek iterācijās, t.s. izrāvienos (angl. sprint).
- Izrāviens ir projekta iteratīvie cikli konkrēta uzdevumu apjoma izstrādei.
- Katrs izrāviens satur tradicionālas programmatūras izstrādes fāzes:
  - prasību analīze, projektēšana, kodēšana, testēšana un gatavās strādājošas programmatūras versijas piegāde.
- Katrā jaunā izrāvienā programmatūra apguv ar jaunu funkcionalitāti un iespējām,
  - to var demonstrēt Klientam un saņemt atsaucēmas par padarīto darbu, norādījumus un vēlmis turpmākai attīstībai.
- Izrāviens parasti ir ap 30 dienu garš izstrādes periods.
  - Sākas ar sapulci, kur kopā tiekas klients un izstrādātāju komanda.
  - Klients izvēlas viņam pašam prioritārākos uzdevumus no produkta nepadarīto darbu saraksta.
  - Komanda nosaka, kuri no šiem uzdevumiem var tikt realizēti šī izrāviens ietvaros.
- Darbība izrāviens ietvaros ir empirisks process, un daudzi procesi tajā ir nenoteikti un nekontrolējami, tas ir sava veida melnā kaste, kam ir nepieciešama ārēja kontrole.
  - Attiecīgi nepieciešamā kontrole, iekšējais risku vadīšanu, ir uzlikta uz katru izrāvienu, lai uzturētu kārtību pie nepieciešamās elastības paaugstināšanas.



Oksana Ņikiforova

"Objektorientētas programmēšanas praktikums (studiju projekts)"

6.

7.

## SCRUM dzīves cikls: Pēc-spēles fāze

- sagatavošanās jaunas programmatūras fināla versijas piegādei, kas iekļauj fināla dokumentāciju, pirmspiegādes testēšanu un piegādi.
- sākas, kad ir izpildītas visas klienta prasības, t.i. visa funkcionalitāte ir realizēta tādā veidā, kā to vēlas redzēt klients.
- Pēc fāzes sākuma nekādas jaunas prasības no klienta puses nevar rasties, visas pūles tiek virzītas, lai sagatavotos programmatūras piegādei - rūpīga testēšana, integrācija, dokumentācijas sagatavošana, lietotāju instrukcijas uzrakstīšana.

Oksana Nīkiforova "Objektorientētas programēšanas praktikums (studiju projekts)"

8.

## IBM/Rational vienots process IBM/Rational Unified Process - RUP (1998)

Organizācija laika secībā

Oksana Nīkiforova "Objektorientētas programēšanas praktikums (studiju projekts)"

9.

## RUP Sākuma fāze

- Mērķi:**
  - Biznes-plāna sastādīšana
  - Risku novērtēšana
- Jāatbild uz jautājumiem:**
  - Vai programmatūras ieviešanas palpa pārsniedz programmatūras izstrādes izmaksas?
  - Vai programmatūra tiks ieviesta tirgū pietiekami ātri, lai šo palpu sagremtu?
- Sākuma fāzes darbības:**
  - Sistēmas konteksta saprašana
  - Sistēmas lietošanas apgabala definēšana
  - Sistēmas funkcionālo un nefunkcionālo prasību uzmetums
  - Iespējamo risku samazināšana
  - Sākotnēja projekta plāna izstrāde
- Sākuma fāzes rezultāts:**
  - Sistēmas konteksta modelis (biznesprocesi un konceptuālo kļau modelis)
  - Sistēmas realizācijas un testēšanas uzmetumi, kas var turpmāk būt izmainīti
  - Sistēmas darbības konceptuālais prototips
  - Bāzes risku saraksts un prasību prioritāšu saraksts
  - Biznesplāns ar projekta vides izstrādes aprakstu un veiksmes kritēriju novērtēšanu (prognozējama palpa, tirgus izpēte, projekta novērtēšana)

Oksana Nīkiforova "Objektorientētas programēšanas praktikums (studiju projekts)"

10.

## RUP Izvēšanas fāze

- Mērķi:**
  - Noteikt pārejas prasības (kas nebija atklātas sākuma fāzē)
  - Uzsākt arhitektūras veidošanu
  - Turpināt novērtēt riskus
- Izvēšanas fāzes darbības:**
  - Projekta izstrādes komandas veidošana
  - Izstrādes vides modifēšana (pielāgošana) atbilstoši projekta būtībai
  - Prasību apstrāde:
    - Vai visas prasības ir nodefinētas?
    - Vai to detalizācijas pakāpe atbilst vajadzībām?
  - Iespējamo risku samazināšana
  - Biznesplāna atbilstības pārbaude
- Izvēšanas fāzes rezultāts:**
  - Sistēmas konteksta pilns modelis (biznesprocesi un konceptuālo kļau modelis)
  - Sistēmas analīzes un projektēšanas jauns modelis (dažādu sistēmas aspektu diagrammas UML valodā)
  - Sistēmas realizācijas un testēšanas uzmetumi, kas var turpmāk būt izmainīti
  - Sistēmas lietošanas sākotnējā interfeisa prototips
  - Sākotnējā sistēmas lietošanas instrukcija
  - Pārstrādāts risku saraksts
  - Pabeigts biznesplāns ar projekta vides izstrādes aprakstu un veiksmes kritēriju novērtēšanu (prognozējama palpa, tirgus izpēte, projekta novērtēšana)

Oksana Nīkiforova "Objektorientētas programēšanas praktikums (studiju projekts)"

11.

## RUP Konstruēšanas fāze

- Mērķi:**
  - Radīt programmas produktu, kas ir gatavs sākotnējai ieviešanai un β- testēšanai
- Konstruēšanas fāzes darbības:**
  - Atlikušo prasību daļas definēšana (parasti ap 20% no visām prasībām)
  - Interfeisa prototipa veidošana
  - Sistēmas analīzes un projektēšanas modeļa bagātināšana
  - Sistēmas arhitektūras projektēšana
  - Prioritāro prasību funkcionēšanas realizācija
  - Testēšanas uzdevumu izpilde
  - Kopējas sistēmas testēšana
  - Testa rezultātu novērtēšana
- Konstruēšanas fāzes rezultāts:**
  - Ieviešanas plānošana
  - Pati programmatūra - versija, kas ietver bāzes funkcionēšanu
  - Uzturams un minimāli modificējams sistēmas arhitektūras apraksts
  - Lietošanas instrukcija β-versijas testēšanai

Oksana Nīkiforova "Objektorientētas programēšanas praktikums (studiju projekts)"

12.

## RUP Pārejas fāze

- Mērķi:**
  - Salīdzināt sistēmas funkcionēšanu ar prasībām un noskaidrot lietderīguma pakāpi
  - Apskatīt visus jautājumus, kas ir nepieciešami lietotājiem, lai turpmāk strādātu ar programmatūru, ieskaitot β- testēšanas rezultātus
- Sistēmas ieviešana tiek klasificēta divos veidos:**
  - Produkts izplatīšanai tirgū, kas var būt ļoti liela mēroga
  - Sistēmas izstrāde pēc pasūtījuma
- Pieņem, ka produkti tiek ieviesti ar zināmu defektu daļu, un paredzēt produkta attīstību turpmākās versijās - uzturēšanas aktivitātes**
- Konstruēšanas fāzes rezultāts:**
  - Programmatūra ar instalācijas iespējām
  - Juridiskā dokumentācija - līgumi, licences piekrišanās, atteikumi no pretenzijām, garantijas
  - Pilns sistēmas analīzes un projektēšanas modelis
  - Pilns un izlabots sistēmas arhitektūras apraksts
  - Lietotāju, operatoru, sistēmas administratoru instrukcijas, kā arī mācību materiāli
  - Lietotāju atbalsta adreses un Internet lapas, kur var dabūt papildus informāciju par produktu, paziņot par kļūdām un iegūt jaunas programmatūras versijas

Oksana Nīkiforova "Objektorientētas programēšanas praktikums (studiju projekts)"