ICTE in Regional Development, December 2013, Valmiera, Latvia

Competence development in a combined assessment and collaborative e-portfolio information system

Aleksandrs Gorbunovs*, Atis Kapenieks and Ieva Kudina

*Distance Education Study Centre, Riga Technical University, Azenes Str. 12, LV-1048, Riga, Latvia

Abstract

A necessity to find solutions on how to make the learning process more motivating and efficient still exists. This paper describes the scaffolding e-Portfolio system developed by the research team of Distance Education Study Centre, Riga Technical University. This system differs from traditional e-Portfolio systems by utilizing a motivating approach in the acquiring of competencies, involvement in group-work, increasing responsibility for personal and peer achievements, and comparison of scores, study results, assessments and competency development levels during the whole course study period in two educational systems: university study portal “ORTUS” and e-Portfolio. The introduced e-Portfolio algorithmic model ensures the collection of data (marks and recommendations) from all assessments and opening them up for the appropriate e-Portfolio group members; consequently, students have an excellent possibility to improve their outcomes and develop required competencies. The study also shows correlations between learner activity levels within e-Portfolio and their learning outcomes.

© 2013 The Authors. Published by Elsevier B.V. Open access under CC BY-NC-ND license.
Selection and peer-review under responsibility of the Sociotechnical Systems Engineering Institute of Vidzeme University of Applied Sciences

Keywords: e-Portfolio; Assessment; Competence development; Group-working; Information system.

1. Introduction

What actions should be taken to make learning more effective? This question addresses many researchers and education subject matter experts. By virtue of the new technology era and changing of the educational paradigm,

* Corresponding author.

E-mail address: aleksandrs.gorbunovs_1@rtu.lv, atis.kapenieks@rtu.lv, ieva.kudina@gmail.com
democracy features increasingly interpose the learning process. Despite positive effect which is gained by distance learning, i.e. technology enhanced learning, when a learner has several lists of learning courses to choose from and a possibility to audit and make learning more flexible, there is also a side-effect.

This is a question about a too high drop-out rate among students. Forrester report states that only twenty-five per cent of learners finish their e-learning courses. The situation in blended and hybrid learning seems a little bit better; although, it is also unpromising. This is why course developers are seeking appropriate advanced teaching methods to enhance learning. Teaching staff ought to analyse learners’ behaviour forms within the e-learning environment and find suitable ways to engage and motivate them.

Some possible solutions were proposed in earlier findings. Thus, to make the learning process more effective, it could be necessary to involve students in the knowledge acquisition process by means of active engaging and a motivating approach, find parallels in the learning process with real-life situations, simulate tasks related to daily real-life activities, analyse the problem and find an appropriate solution, act, think critically and reflect (find appropriate solutions).

Real-life situations further improve critical thinking skills. Collaborative learning methods are considered powerful learner engaging and learning process quality improvement instruments. Learners are encouraged actively participate in group-work, share and distribute their own ideas, and create new ideas and knowledge.

e-Portfolios initially devised as show-window instruments are nowadays changing important competence development systems. Ideally, they should not only be used as signboards. New e-Portfolio systems ought to ensure appropriate engagement and a motivating environment where learners could develop their competencies in the most efficient way. It might be done by actively involving students in collaborative group-work activities, tailoring tasks to real-life situations, encouraging learners to think critically and reflect on group members’ suggestions and improve learning outcomes. Multi-level assessment tools within the system might allow to track the learners’ competence enhancement process and compare it to their study activities.

2. Background

There were many endeavours to make the learning environment and knowledge acquisition process more effective and engaging. The high technology era brings new approaches, proposals and solutions.

To enhance student readiness for further professional duties, an assessment of their knowledge structure was proposed by the research team from the Department of Systems Theory and Design of Riga Technical University. Since 2005 they have been developing a semi-formal knowledge representation tool in the form of a concept map based intelligent knowledge assessment system built on Java modules. Students were asked to complete sequential tasks to assess their objective knowledge level on a particular Artificial Intelligence course theme. They were able to reduce the level of complexity instead of decreasing their potential final score at the end of the task. This computerized solution ensures student self-assessments and allows teaching staff to improve courses by analysing learning outcomes.

To find new ways in the knowledge acquisition and building process, in 2009 and 2010 Kagenieks implemented action research as a collaborative study method for Riga Technical University’s bachelor level students. The Google Docs application was used to determine the creative abilities and anxiety for knowledge of learners. In other words, it was anticipated that using the action research method learners would create living theory, and more successful students might set an example, motivate and encourage other students in the group to promote their creative activities. Kagenieks further developed Whitehead’s basic method of the living theory approach in education which stated that everybody might have their own problem solving explanation, and this might influence a person’s learning process.
Several student groups were formed with a task to work in small virtual teams where homework lists in a form of particular descriptions were submitted to a common group signboard. Other students were offered to make further constructive suggestions until the task is completed. It was found that the creativity in students’ actions led to knowledge creation. The implemented method stimulated learner motivation and increased the quality of mastering the course.

However, neither these studies, nor other research efforts provided a specific effective united instrument which would ensure learner competence development assessments, their creativity expression in the form of critical thinking notes and reflection, and improvement of learning outcomes. The Next sections will introduce our approach and possible solutions.

3. Development of an engaging collaborative information system

3.1. Assignments

After competence enhancement, an assignment to create engaging learning environment was formulated. The environment had to ensure a dynamic competence acquisition, measurement and assessment process for students. Course developers were asked to set up appropriate competence assessment forms and draw up a system which would engage students into a collaborative, creative group-working environment. This all had to be merged with an existing information system.

The first assignment part anticipated development of specific parameters which had to characterize main competencies improved during the course. This task could be done by filling out competence self-assessment forms and tailoring them to the course.

The second assignment part had the aim to create an additional information system, preferably in the form of an e-Portfolio system, which had to provide student group-working activities in small teams of four participants each where they would be asked to assess team member accomplishments (their homework / fulfilled tasks related to the corresponding course theme) and make self-assessments. These evaluations had to be realized by giving marks and filling in critical thinking notes within the e-Portfolio system.

3.2. Algorithmic model design

To achieve the goal of an engaging learning environment, which promotes competence development, an experimental e-Portfolio algorithmic model was created. This model, along with the external study portal information system "ORTUS", ensured the student competence development assessment process during the study course.

Fig. 1 shows a simplified scheme of the experimental e-Portfolio system’s algorithmic model. Extra external assessment processes (the right side of the Fig. 1) are added to this figure just to emphasize the importance of a likely full spectrum of assessment types which may vary in different learning environments and have crucial significance in competence development.

Students can upload the finished homework onto Riga Technical University’s educational portal ‘ORTUS’. Then, with the assistance of an e-Portfolio system administrator, the course tutor sends these works to the e-Portfolio system. E-Portfolio groups are formed based on a sequence of submitted homework.

Students should assess their group members’ accomplishments and make a self-assessment. Students have a possibility to see group member names with their achieved assessment results: marks, critical thinking notes and constructive suggestions. In this case, to verify details, students can also establish external mutual and/or e-communication contacts. Teaching staff support is also appreciated and might be used to render good offices to students.
Reflection is one of the key processes as “the real value of an e-Portfolio is in the reflection”\(^2\). A student is kindly asked to observe and collect group member feedback, listen to a tutor’s recommendations, estimate data and compare it with their own calculations, analyse information, select appropriate conclusions, add something well-formed or synthesize dandy things. As the result the number of improved homework ought are increased.

Fig. 1. Simplified scheme of the experimental e-Portfolio system’s algorithmic model.
There was also a necessity to analyse students’ competence development and change dynamics. To ensure the recording of the competence development process, the existing RTU student educational web portal ORTUS, which is built on open source Moodle software, was used and eight self-assessment survey lists were created.

3.3. System justification

The formulated system design task is to create an e-Portfolio system that ensures learner group-working activities in teams of four students each. In correspondence with the recognized five key subject areas (themes) there were five collaborative assignments during the course. Working in groups embraces a user’s authentication and the environment for self-assessment and other group members accomplished homework assessments in scores and text. The homework assessment environment includes the homework of all group members. Homework as well homework assessment criteria are finished and submitted almost at the same time.

The software preference statement was formulated based on considerations that a new application should be programmed to ensure student group work. This software would be exactly suited to students’ needs as an extra module for an existing module in the Moodle environment with the functionality of group work, which was not available in existing Moodle environment.

To illustrate user interactivity within the e-Portfolio system, an example, which characterizes student group work possibilities, will be shown in the next paragraphs. After each corresponding homework submission to the course’s Moodle e-environment in the university’s study portal ‘ORTUS’ related to the theme (stage), the user has to complete the next task – read, analyse and assess the homework of other three group members which they already uploaded to the course in the Moodle e-environment at about same time with the user. The same rules apply to a student’s self-assessment activities in all course modules.

3.4. System programming

The new application software for student activities in the e-Portfolio system is created using the open source integrated development environment ‘Netbeans’. The fundamental principle of the three-pronged architecture MVC (‘Model View Controller’) is applied. At the lower level (prong) of the system the object-oriented data model ‘Hibernate’ is utilized. At the system’s controller and view levels the JSF (Java Server Faces) libraries are used. The JSF offers a formation of auxiliary classes and controller classes, as well objects’ input into xhtml standard pages and their depiction at the view level. Navigation between pages is provided by defining JSF configuration in the file faces-config.xhtml. The JSF also ensures functional features of AJAX (Asynchronous JavaScript and XML). It gives an opportunity to avoid of JavaScript code which sometimes may confuse a user.

The created e-Portfolio system is hosted on a ‘Glassfish 3.1’ application server. Received data is collected, saved and maintained in a ‘MySQL 5.5.16’ data base. It allows easy data export and import from and to the Moodle MySQL database. Java ‘Development Kit 1.6.0_24’ is used.

Five data tables, classification tables and statistical tables are created within the database. Student homework data, i.e. name of author, group member or assessor names, assessments, marks and assessment remarks, e.g. feedback, critical thinking notes or encouragement are saved, collected, picked and read in the data tables. Appropriate assessment criteria for each of the five group works are placed in the classification tables. User data, login details, authentication information, such as beginning and end of the authentication, is placed in statistical tables.

The object-oriented data level enables application software to process data in a database using ‘Hibernate’ libraries. Business functions, which process data arrays, are placed at the controller level. Functions, which reproduce processed data arrays in an established format, are defined at the view level.
3.5. Software programming solutions

3.5.1. Authorisation.

The e-Portfolio system uses MySQL database authorisation. This means that all system users are defined in the database and their passwords are kept in cipher. Applying solutions of Hibernate libraries, a possibility to connect to the database with given account name and password is checked. If this possibility is accepted as enforceable, the user gains access to the system. In other cases the system remains closed. The Hibernate utility program and a method, which checks users and their existence and conformity in the database, is defined:

```java
public static SessionFactory getSessionFactory(String username, String pwd) {
    SessionFactory sessionFactoryWithUser;
    Configuration cfg;
    try {
        cfg = new Configuration();
        cfg.configure();
        cfg.setProperties(System.getProperties());
        System.setProperty("hibernate.connection.password", pwd);
        System.setProperty("hibernate.connection.username", username);
        System.setProperty("hibernate.dialect", "org.hibernate.dialect.MySQLDialect");
        System.setProperty("hibernate.connection.driver_class", "com.mysql.jdbc.Driver");
        System.setProperty("hibernate.show_sql", "true");
        System.setProperty("hibernate.current_session_context_class", "thread");
        System.setProperty("hibernate.query.factory_class", "org.hibernate.hql.classic.ClassicQueryTranslatorFactory");
        sessionFactoryWithUser = cfg.buildSessionFactory();
    } catch (Throwable ex) {
        // Log the exception.
        System.err.println("PtF.business.error:Initial SessionFactory creation WITH USER failed." + ex);
        throw new ExceptionInInitializerError(ex);
    }
    return sessionFactoryWithUser;
}
```

A constructor of an auxiliary class “PSimpleLoginBeanPaligs” is designed with the aid of the method “getSessionFactory(…,…)”:

```java
public class PSimpleLoginBeanPaligs implements Serializable {
    String currentUser;
    String currentPwd;
    String returnOutcome;
    Session session = null;
    public PSimpleLoginBeanPaligs(String currentUser, String currentPwd) {
        this.session = HibernateUtil.getSessionFactory(currentUser, currentPwd).getCurrentSession();
    }
}
```

In the auxiliary class we define a method, which checks whether a user with a password was able to read records from the table “PtTema” (reading authorisation to the user was given before). If reading was successful, the appropriate record in the statistical table of the database is made. If the reading was unsuccessful, a “String” type variable with the value „fail” is returned:
public String getUsernameDatiCheck(String currentUser, String currentPwd) {
    returnOutcome = "fail";
    List<JtStMacolaJedzieni> macolaList = null;
    try {
        org.hibernate.Transaction tx = session.beginTransaction();
        Query q = session.createQuery("select mj.jtJedzieni from JtStMacolaJedzieni as mj where mj.jtMacoli in (select mc.id from JtMacoli as mc where mc.id = " + 1 + ")");
        macolaList = (List<JtStMacolaJedzieni>) q.list();
        returnOutcome = "success";
        if (currentUser.equals("admin")) {
            returnOutcome = "admin";
        }
        System.out.print("ePortfolio:LOGIN SUCCESS: " + currentUser);
        session.close();
        // create a java calendar instance
        Calendar calendar = Calendar.getInstance();
        // get a java.util.Date from the calendar instance.
        // this date will represent the current instant, or "now".
        Date now = calendar.getTime();
        // a java current time (now) instance
        Timestamp currentTimestamp = new Timestamp(now.getTime());
        PfStatistikaLietotaji pfsl = new PfStatistikaLietotaji();
        pfsl.setLietotajaVardsLogin(currentUser);
        pfsl.setStatuss("login");
        session = HibernateUtil.getSessionFactory().openSession();
        org.hibernate.Transaction tx2 = session.beginTransaction();
        session.saveOrUpdate(pfsl);
        tx2.commit();
    } catch (Exception e) {
        System.out.print("ePortfolio:LOGIN FAILED:" + currentUser);
    }
    return returnOutcome;
}

The “String” type variable dependencies for Web pages are described in the configuration file faces-config.xml corresponding to JSF system architecture:

<managed-bean>
<managed-bean-name>PSimpleLoginBeanKontrolieris</managed-bean-name>
<managed-bean-class>controller.PSimpleLoginBeanKontrolieris</managed-bean-class>
<managed-bean-scope>session</managed-bean-scope>
</managed-bean>

<navigation-rule>
<from-view-id>/plogin.xhtml</from-view-id>
<navigation-case>
<from-action>#{PSimpleLoginBeanKontrolieris.getUsernameCheck}</from-action>
<from-outcome>success</from-outcome>
<to-view-id>/pmain.xhtml</to-view-id>
</navigation-case>
A method "getUsernameCheck()" is defined in the controller class. It checks a user’s name and password. Navigation rules are defined in JSF configuration file, and this method is called from the “plogin.xhtml” page. Depending on whether the method returns “success”, “fail” or “admin”, appropriate Web working pages appear:

```java
@ManagedBean
@SessionScoped
public class PSimpleLoginBeanKontroleris implements Serializable {
    String loginname;
    String password;
    String usernameCheck;
```
PSimpleLoginBeanPaligs helper;
FailuPaligs helper2;
public PSimpleLoginBeanKontrolieris() {
    
    public String getLoginname() {
        return loginname;
    }

    public void setLoginname(String loginname) {
        this.loginname = loginname;
    }

    public String getPassword() {
        return password;
    }

    public void setPassword(String password) {
        this.password = password;
    }

    public String getUsernameCheck() {
        helper2 = new FailuPaligs();
        helper = new PSimpleLoginBeanPaligs(loginname, password);
        usernameCheck = helper.getUsernameDatiCheck(loginname, password);
        if (loginname.equals("admin")) {
            helper2.importAll();
        }
        return usernameCheck;
    }
}

The control is defined at the View level (top level) of software architecture. It is used to call the Controller level (controller class) method “getUsernameCheck()”:

```
<h:form>
  <table>
    <tr>
      <td><h:outputText value="Enter Login ID: ">/</td>
      <td><h:inputText id="loginname" value="#{PSimpleLoginBeanKontrolieris.loginname}" /></td>
    </tr>
    <tr>
      <td><h:outputText value="Enter Password: ">/</td>
      <td><h:inputSecret id="password" value="#{PSimpleLoginBeanKontrolieris.password}" /></td>
    </tr>
    <tr>
      <td><h:commandButton value="Login" action="#{PSimpleLoginBeanKontrolieris.getUsernameCheck}" /></td>
    </tr>
  </table>
</h:form>
```
3.5.2. B. Data-entry.
After opening the Web page, a user sees links to five group-working activities, as well five links to view his/her homework assessments made by corresponding group members:

```html
<h: dataTable value="dt" rendered ="true">
  <h:column>
    <h:commandLink styleClass="h1" value="1. majas darbs" action="#{MajasDarbuKontrolieris.md1Saite}"/>
    <h:outputText  value="#{KopigaisKontrolieris.newLine}" escape="false" />
    <h:commandLink styleClass="h1" value="1. majas darbs - citu kolegu vertejumi" action="#{MdVertejumiNolasitiKontrolieris.md1SaiteVertejumi}"/>
    <h:outputText  value="#{KopigaisKontrolieris.newLine}" escape="false" />
    <h:commandLink styleClass="h1" value="2. majas darbs" action="#{KustibuKontrolieris.md2Saite}"/>
    <h:outputText  value="#{KopigaisKontrolieris.newLine}" escape="false" />
    <h:commandLink styleClass="h1" value="2. majas darbs - citu kolegu vertejumi" action="#{KustibuKontrolieris.md2VertejumiSaite}"/>
    <h:outputText  value="#{KopigaisKontrolieris.newLine}" escape="false" />
    <h:commandLink styleClass="h1" value="3. majas darbs" action="#{KustibuKontrolieris.md3Saite}"/>
    <h:outputText  value="#{KopigaisKontrolieris.newLine}" escape="false" />
    <h:commandLink styleClass="h1" value="3. majas darbs - citu kolegu vertejumi" action="#{KustibuKontrolieris.md3VertejumiSaite}"/>
    <h:outputText  value="#{KopigaisKontrolieris.newLine}" escape="false" />
    <h:commandLink styleClass="h1" value="4. majas darbs" action="#{KustibuKontrolieris.md4Saite}"/>
    <h:outputText  value="#{KopigaisKontrolieris.newLine}" escape="false" />
    <h:commandLink styleClass="h1" value="4. majas darbs - citu kolegu vertejumi" action="#{KustibuKontrolieris.md4VertejumiSaite}"/>
    <h:outputText  value="#{KopigaisKontrolieris.newLine}" escape="false" />
    <h:commandLink styleClass="h1" value="5. majas darbs" action="#{KustibuKontrolieris.md5Saite}"/>
    <h:outputText  value="#{KopigaisKontrolieris.newLine}" escape="false" />
    <h:commandLink styleClass="h1" value="5. majas darbs - citu kolegu vertejumi" action="#{KustibuKontrolieris.md5VertejumiSaite}"/>
    <h:outputText  value="#{KopigaisKontrolieris.newLine}" escape="false" />
  </h:column>
</h: dataTable>
```

The user can also see suggestions written by an e-Portfolio system administrator and tutors which have administrator rights:

```html
<h:outputText  value="#{KopigaisKontrolieris.newLine}" escape="false" />
<h:outputText  value="#{KopigaisKontrolieris.newLine}" escape="false" />
```
By opening the link to group-working activities (let’s look at an example of the second group-work) the user sees homework files which three other particular e-Portfolio group participants have uploaded to the Moodle environment in the University’s study portal “ORTUS” at about the same time as the user. Here the learner also sees three tables with assessment questions and a scale with values from 1 to 10, as well a text field to fill in critical thinking remarks “InputTextArea”:

```html
<h:outputText value="#{KopigaisKontrolieris.newLine}" escape="false" />
</h:column>
<h:inputHidden id="loginname2111222" value="#{PSimpleLoginBeanKontrolieris.loginname}" binding="#{IeteikumiKontrolieris.loginnamem}"

By opening the link to group-working activities (let’s look at an example of the second group-work) the user sees homework files which three other particular e-Portfolio group participants have uploaded to the Moodle environment in the University’s study portal “ORTUS” at about the same time as the user. Here the learner also sees three tables with assessment questions and a scale with values from 1 to 10, as well a text field to fill in critical thinking remarks “InputTextArea”:

```
After filling out the e-Portfolio homework assessment for the group’s first participant, the user should press the “Save” button (for instance, in the second group-working activity):

```
<h:commandButton value="Save"
action="#{BiznesaIdeja21Kontrolieris.saveBiznesaIdeja21}"/>
```

By pressing the “Save” button, a business method “saveBiznesaIdeja21()” is called from the business class (View Controller) level. In this class a constructor from the auxiliary class “BiznesaIdeja21Paligs” is designed. In this class a method “saveBiznesaIdeja21()” is defined, which saves the assessment of the first group participant in the second group-work, where a method “saveBiznesaIdeja21Paligs(....)” is called from auxiliary class:

```
public BiznesaIdeja21Kontrolieris()
{
    helper = new BiznesaIdeja21Paligs();
}

public BiznesaIdeja21Kontrolieris(HtmlInputHidden loginname, HtmlInputHidden biznesaIdejasNosaukums, HtmlInputHidden biznesaIdejasAutors, HtmlInputHidden biznesaIdejasVertetajs, HtmlDataTable ieraksti2, HtmlInputHidden rowIndex)
{
    helper = new BiznesaIdeja21Paligs();
    this.loginname = loginname;
    this.biznesaIdejasNosaukums = biznesaIdejasNosaukums;
    this.biznesaIdejasAutors = biznesaIdejasAutors;
    this.biznesaIdejasVertetajs = biznesaIdejasVertetajs;
}

public String saveBiznesaIdeja21()
{
    String biznesaIdejasNosaukumsString = biznesaIdejasNosaukums.getValue().toString();
    String biznesaIdejasAutorsString = biznesaIdejasAutors.getValue().toString();
    String biznesaIdejasVertetajsString = biznesaIdejasVertetajs.getValue().toString();
```
jj = 0;
ii = 0;
int i = jautajumaKomentarsList.size();
for (ii=0;ii<i;ii++) {
jautajumaKomentarsString = jautajumaKomentarsList.get(jj);
jautajumaNumurs = jj +1;
{
jautajumaVertejumsString = jautajumaVertejumsList.get(jj);
jj++;
String rez = helper.saveBiznesaIdeja21Paligs (biznesaIdejasNosaukumsString, biznesaIdejasAutorsString, biznesaIdejasVertetajsString, jautajumaVertejumsString, jautajumaKomentarsString, jautajumaNumurs);
}
}

In auxiliary class “BiznesaIdeja21Paligs” the constructor is designed pursuant to the current session:

public class BiznesaIdeja21Paligs {
    Session session = null;
    public BiznesaIdeja21Paligs() {
        this.session = HibernateUtil.getSessionFactory().getCurrentSession();
    }
}

In this auxiliary class the data is saved by utilizing Hibernate classes “PfJautajumi” and “Pf2mdDarbuVertejums”:

public String saveBiznesaIdeja21Paligs{
    String biznesaIdejasNosaukumsString, 
    String biznesaIdejasAutorsString, 
    String biznesaIdejasVertetajsString, 
    String jautajumaVertejumsString, 
    String jautajumaKomentarsString, 
    Integer jautajumaNumurs
    { 
        String rez = "";
        PfJautajumi pfJaut = new PfJautajumi();
        PfJautajumi pfJaut2 = new PfJautajumi();
        PfJautajumi pfJaut3 = new PfJautajumi();
        PfJautajumi pfJaut4 = new PfJautajumi();
        PfJautajumi pfJaut5 = new PfJautajumi();
        PfJautajumi pfJaut6 = new PfJautajumi();
        PfJautajumi pfJaut7 = new PfJautajumi();
        Pf2mdDarbuVertejums mdv = new Pf2mdDarbuVertejums();
        mdv.setBiznesaIdejasNosaukums(biznesaIdejasNosaukumsString);
        mdv.setBiznesaIdejasAutors(biznesaIdejasAutorsString);
        mdv.setBiznesaIdejasVertetajs(biznesaIdejasVertetajsString);
        mdv.setJautajumaVertejums(Integer.valueOf(jautajumaVertejumsString));
        mdv.setJautajumaKomentars(jautajumaKomentarsString);
        pfJaut.setId(jautajumaNumurs);
        mdv.setPfJautajumi(pfJaut);
        try {
            session = HibernateUtil.getSessionFactory().openSession();
            org.hibernate.Transaction tx = session.beginTransaction();
        }
It might be represented by a Hibernate class example:

```java
public class Pf2mdDarbuVertejums implements java.io.Serializable {
    private Integer id;
    private PfJautajumi pfJautajumi;
    private String biznesaIdejasNosaukums;
    private String biznesaIdejasAutors;
    private String biznesaIdejasVertetajs;
    private Integer jautajumaVertejums;
    private String jautajumaKomentars;
    public Pf2mdDarbuVertejums() {
    }
    public Pf2mdDarbuVertejums(PfJautajumi pfJautajumi, String biznesaIdejasNosaukums, String biznesaIdejasAutors, String biznesaIdejasVertetajs, Integer jautajumaVertejums, String jautajumaKomentars) {
        this.pfJautajumi = pfJautajumi;
        this.biznesaIdejasNosaukums = biznesaIdejasNosaukums;
        this.biznesaIdejasAutors = biznesaIdejasAutors;
        this.biznesaIdejasVertetajs = biznesaIdejasVertetajs;
        this.jautajumaVertejums = jautajumaVertejums;
        this.jautajumaKomentars = jautajumaKomentars;
    }
    public Integer getId() {
        return this.id;
    }
    public void setId(Integer id) {
        this.id = id;
    }
    public PfJautajumi getPfJautajumi() {
        return this.pfJautajumi;
    }
}
```

In case the user tries to open the second group-working assessment page from the main working page (user's personal desktop – “pmain.xml”), the learner picks up assessments done by other participants of the group. Hibernate class “Pf2mdDarbuVertejums” values of the following variables “biznesaIdejasVertetajs”, “jautajumaVertejums”, “jautajumaKomentars” are represented in a table:

```
<h:column>
    <f:facet name="header">
        <h:outputText value="Biznesa idejas vertetajs:"/>
    </f:facet>
    <h:outputText value="#{item.biznesaIdejasVertetajs}"/>
</h:column>
```
The e-Portfolio system administrator has a possibility to input suggestions for students respecting uploaded homework:

The administrator saves suggestions by utilizing the “Save” button. It ensures a call for the method “saveIeteikumus()” from the class “IeteikumiKontroleris”:

In the class “IeteikumiKontroleris” the constructor is built and a method of a call is defined. In the method’s call the method “saveIeteikumus(……..)” from auxiliary class “IeteikumiPaligs” is asked:
recordCount = 267;
helper = new IeteikumiPaligs();
ieraksti = null;
startId = 1;
endId = 10;
currentUser = null;
currentId = null;
}
public IeteikumiKontrolieris(HtmlInputHidden loginname, int startId, int endId)
{
    pageSize = 10;
    recordCount = 267;
    helper = new IeteikumiPaligs();
    this.loginname = loginname;
    ieraksti = null;
    this.startId = startId;
    this.endId = endId;
    currentUser = null;
    currentId = null;
}
public String saveIeteikums()
{
    rez = helper.saveIeteikumu(getIeteikumsMd1(), getIeteikumsMd2(), currentId,
    currentUser, getIeteikumsMd4(), getIeteikumsMd5(), getIeteikumsMd6());
    return rez;
}

The method “saveIeteikumu(……..)” is defined in the auxiliary class “IeteikumiPaligs”. The data is saved by utilizing Hibernate class “PfLietotajiIeteikumi” and its “set” methods:

public String saveIeteikumu(String ieteikumsMd1, String ieteikumsMd2, Integer id, String
lietotajaVards, String ieteikumsMd4, String ieteikumsMd5, String ieteikumsMd6)
{
    String rez = null;
PfLietotajiIeteikumi lie = new PfLietotajiIeteikumi();
lie.setIeteikumaMd2(ieteikumsMd2);
lie.setLietotajaVards(lietotajaVards);
l ie.setIeteikumaMd1(ieteikumaMd1);
l ie.setIeteikumaMd4(ieteikumaMd4);
l ie.setIeteikumaMd5(ieteikumaMd5);
l ie.setIeteikumaMd6(ieteikumaMd6);
l ie.setId(id);
try
{
    session = HibernateUtil.getSessionFactory().openSession();
    Transaction tx = session.beginTransaction();
    session.update(lie);
    ...
tx.commit();
rez = "success";
}
catch(Exception e)
{
    e.printStackTrace();
    rez = "fail";
}
return rez;

4. Experimental part

4.1. Testing in living lab environment

To test the prototype of the created e-Portfolio system, teaching staff at the Distance Education Study Centre (DESC), Riga Technical University (RTU) remodelled and enriched an existing „Business Planning for Open Markets” (BPOM) blended e-learning course for first year bachelor study programme students. The notion that Living Labs are intended to involve users in the innovation process, knowledge sharing, exploration, experimentation, assessment, and co-creation[12] furthered testing of our experimental system in such an environment. The new course was implemented from the September 5th, 2011 until January 27th, 2012, and was offered to Electronics and Telecommunications faculty students as required choice category studies, as well as for bachelor level students from other faculties as a voluntary category studies choice.

According to course objectives and structure, there were two main streams of assessments. First, eight self-assessment survey lists were created within the university’s study portal ‘ORTUS’ to analyse students’ subjective feelings about their BPOM competence levels at each stage of the course on a scale from 1 (the lowest valuation) to 10 (the highest valuation).

In total, there were seven BPOM competencies to be evaluated (for instance, competencies to estimate the viability of a business idea, determine a company’s ability to carry out business ideas, be aware of competitive factors, estimate financial resources, assess and develop a company’s ability to carry out business ideas, identify possible risks, and marketing competence). Learners had access to the self-assessments through links inside the BPOM course.

After the first initial self-assessment students had to fill out the questionnaire repeatedly every fortnight until the end of the course. A two week time frame was chosen due to the time period when new themes were introduced to the course in a stated sequence. Moreover, this approach allowed teaching staff to monitor and analyse the change dynamics of competencies, as well as, based on these observations, make conclusions about the significance level of each course theme and its impact on competency development within BPOM. The last eight self-assessments had to be filled out after the submission of final papers (completed business plan and presentation). Corresponding data, obtained from the database, was used to compare learner self-assessments to their test results and activities within e-Portfolio system.

Secondly, student competency development and assessment within the e-Portfolio system was also provided through the links from university’s study portal ‘ORTUS’. By opening the link students were guided to the e-Portfolio login page. Activation of actions starts from a person’s authentication and authorization. In the user profile he or she is asked to enter his/her login identification and password. For this purpose the authentication and authorization part of the ‘MySQL 5.5.16’ database was exploited, and an authority was assigned to the user’s corresponding users group. Similar login procedures also applied to administrators and tutors.
After login the e-Portfolio user’s main page with personal desktop is opened. It contains links to homework accomplishments with assessment tasks and homework assessments with feedback and suggestions given to the users by peers – particular group members.

By clicking on the appropriate homework task (for instance, in Fig. 4 – links: „1. majas darbs”, „2. majas darbs”, etc.), a user’s work page opens. Here the student can download and save the homework of his/her group members, read it, assess, mark the corresponding competence level, and write critical thinking notes to assist others in further improvement of their work. After completing of group member’s assessment (grading by points, filling out assessment forms, and saving), filled-out fields are no longer visible.

There was an obligation to assess not only group members, but also make a self-assessment: assess the personal competency level and describe personal considerations regarding achieved results. The obtained data could be contrasted and compared to self-assessments and test results made outside the e-Portfolio system (within the external Riga Technical University’s study portal ‘ORTUS’), and internal peer assessments within particular a e-Portfolio group.

By clicking on appropriate group participant assessment link peer assessment and feedback summary in the user’s page was opened. Here not only a group member’s remarks, but also a user’s own assessment scores and notes are displayed. Analysis of this data was incredibly useful for the learner to resolve a problem, get in touch with group members, other course students and tutors, more precisely assess the personal competency level, improve accomplished poor or even failed work, and develop required competencies.

Administrator and tutor login windows were similar to the users. However, after authentication and authorization, windows with a link for tutor guidance input opened.

By clicking on the guidance input link, the tutor’s guidance input page opened. The tutor had the possibility to write corresponding notes and suggestions related to specified work or a question, and leave links to external educational materials.

The groups of four students each for five group-work activities were formed in line with a sequence of submitted accomplished homework No.1 and No.2. After the second homework the composition of group participants did not change until the end of the course. The purpose of that was an assumption that assignments for group-work from the second to fifth were tightly tangled, the third homework was built on the conditions and results from the second homework, the fourth homework – on the third, and the fifth – on the fourth. Thus, from our point of view students were able to give a good account of themselves, become more familiar with group members and their work direction, provide information on suggesting better improvements for improving the work of group members and increase a dynamics of competence development. Here parallels could be drawn with project team building when project members are asked to contribute their proposals regarding project tasks – everybody should be on time and everything should be the pick of the bunch.

Students were asked to complete given tasks, upload their accomplished homework to the university’s study portal ‘ORTUS’, login into their e-Portfolio group create a self-assessment of their own work and competency levels, assess group members work, analyse, give them appropriate suggestions, and for their part acquaint themselves with group participant feedback, estimate, analyse and select constructive ideas, think about possibilities to improve their own work, and update initial drafts if possible.

In total, 254 first year bachelor study programme students were enrolled in the BPOM course. 173 students passed the examination at the end of course. Only 145 self-assessment questionnaires were acknowledged as valid. From all enrolled students, 56 learners took part in all group-work activities within the e-Portfolio system, i.e. all five times; 16 students were also very active – they participated in four group-work activities; 19 students were
rather moderate – three group-work activities; 27 students were less active – two activities; 39 students were inactive – only one group-work was done; and 97 did not participated in any of the e-Portfolio group-work activities.

4.2. Correlations

Pursuant to two main streams of assessments noted in the previous section here we will study this in two planes. Firstly, we examined respondents’ initial self-assessments and set them against their outcomes, test and final exam results. We have found that a correlation between self-assessments and test results does not exist. There was also no correlation between initial self-assessments and exam results. A lot of students had a lack of confidence, and, as a result, their initial self-assessment marks were far from real competency levels. It took time to get some confidence. Hardworking students had steady progress, which allowed them to acquire the required competencies and achieve remarkable final exam results. In many cases these results were even higher than in other groups. This especially applies to the comparison of the first test and initial self-assessment ratio.

Contrariwise, some students overleapt themselves. Starting from the second course module they made corrections to self-assessment questionnaires. Activities within the e-Portfolio system influenced a more precise adjustment of these changes. However, those students who assessed themselves the worst in the initial self-assessment also evaluated themselves the worst at the final self-assessment stage; while the most self-assured students remained the same in initial and final self-assessment stages. Fig. 3 shows these characteristics and waves through importance of activities and hard work within the e-Portfolio system. This enables good students who offer good accounts to assist students with a low level of confidence to achieve their aim (columns in Fig. 3 are formed by combining the data into clusters of successive initial self-assessment marks). Moreover, a higher level of group work activity and even responsibility let learners dash and catch up to the smartest students.

![Fig. 3. Correlations depending on initial self-assessments.](image)

By looking at the question further, it should be admitted that a correlation exists between students’ test marks, final exam results and achieved competencies, and their activities in the e-Portfolio system.
Secondly, our attention was guided to correlation findings of student e-Portfolio activities, their outcomes and achieved competency levels. We have found that the final self-assessments of students correlate with exam results and group-work activities within the e-Portfolio system. This also applies to the number of login files in the e-Portfolio system (see Fig. 4 and Fig. 5). It could be said that final self-assessments quite accurately indicate and characterize a student’s competency development. These relations can be best seen in Fig. 4.

There students were sorted based on terms of their activity within the e-Portfolio as follows:

- **Active users** – those who have participated in 4 or 5 group works;
- **Moderate users** – those who have participated in 2 or 3 group works;
- **Inactive users** – those who have not participated in any of group work or take part in only one activity.

Fig. 5 breakdown students according to a number of accomplished group works within e-Portfolio (e.g., 5 group works – the most active users, 4 – active, 3 – moderate, 2 – less active, 1 – inactive, and 0 – the most inactive).

We have also found that the number of improved homework has a direct correlation with the number of e-Portfolio login files. More active students take more part in the offered group work activities. It could be established that a user’s reflection on critical thinking notes and constructive suggestions leads to creativity, synthesis and competency development. As a result, the number of corrected, slightly improved or crucially processed products depends on a user’s activity level within the e-Portfolio system.

During the first e-Portfolio system testing in Living Lab we have received a total of 312 improved works. This number shows that:

- The second homework was improved 78 times;
- The third homework was improved 66 times;
- The fourth homework was improved 65 times;
- The fifth homework was improved 103 times.

There was no obligation to make any homework improvements after first group work activity.
After analysing statistics of improved works we were surprised that there were several improved works developed by students, who did not take part in any e-Portfolio group-work activities (for instance, the right column in the second set of columns from the right in Fig. 5).

![BPOM competence (on average of seven) development correlations with e-Portfolio activities.](image)

It might be explained by their interest in e-Portfolio working processes („what’s going on?”). Their accomplished draft homework items were placed in one of system’s groups in the e-Portfolio system. Despite their indisposition towards group-working, few of these students took a look at appropriate group members feedback, as we can see in the last column of the Fig. 5 („Login files (on average)”), looked over, selected ways for necessary corrections, and made improvements to the final version of the homework.

5. Conclusions and further work

The introduced e-Portfolio system has an invaluable positive impact on learner competency development. The group-work environment within e-Portfolio:

- Ensures collaboration activities within and outside e-Portfolio groups;
- Engages learners into the knowledge acquisition process;
- Enhances responsibility for one’s own actions and team work results;
- Motivates learners to think critically and reflect on things and on-going processes.

Testing in the Living Lab highlighted an effect of e-Portfolio usage on a user’s competence development. The more activities were performed within the e-Portfolio system, the better were the outcomes, and more useful assistance was given to other group members.

At the same time we ought to revise the e-Portfolio group formation procedure. The pilot phase of e-Portfolio system testing highlighted a problem with student drop-out rate. It was rather high and had an effect on actions within several groups. It is understood that groups which have lost one or two participants make worse progress than
the fully commissioned groups. Moreover, if a group loses three of its four participants, any group-work activities will be impossible. However, this situation might be imagined as a simulation environment in both international and domestic project developments. In this case, the loss of group participants may imitate joint project challenges and real-work conditions when project partners are working on the certain collaborative tasks and have to make appropriate contributions to work-packages. The project must go on even in case some project partners, who have contributed before, leave the project.

In the first experimental stage the option of delivering learning suggestions was enabled for tutors close to the end of the course. In addition to that we had a lack of available instructors who could be involved in group-work activities. Thereby this option was not exploited as extensively as we had planned. In next prototype testing stages we expect this tool to work better.

In perspective it might be useful to consider adding a recommendations or suggestions generation tool emulating the decision-making ability of a human expert. Apparently, e-Portfolio will be further presented with system artificial intelligence features.

The introduced e-Portfolio system could be implemented in other educational organisations. It might be installed in institution learning management systems or remain separately on servers. It would offer organisation teaching staff the opportunity to engage students in the competency development process.

Acknowledgement

This research has been partly supported by a grant from the European Regional Development Fund (ERDF), "E-technologies in innovative knowledge source and flow systems (ETM)" Project No. 2DP/2.1.1.1.0/10/APIA/VIAA/150 (RTU PVS ID 1534). (Contract No. 2010/0222/2DP/2.1.1.1.0/10/APIA/VIAA/150).

References