Using Screencasts in Computer Programming Courses

Preliminary results

Sanja Mohorovičić and Edvard Tijan
Department of Electrical Engineering, Automatics and IT
University of Rijeka, Faculty of Maritime Studies
Rijeka, Croatia
sanja@pfri.hr, etijan@pfri.hr

Abstract:—**The paper researches the use of screencasts in computer programming courses in higher education. Screencasts are video captures of the activity that is taking place on the computer screen, which can prove to be useful when presenting and demonstrating computer activity. A pilot research is currently taking place at the University of Rijeka, Faculty of maritime Studies, where teachers and students enrolled in the programming course are testing screencasts to assess their usability in higher education. Preliminary results show great potential, and the need for future research.**

Keywords:-screencasts; computer programming courses; screencasting tools; higher education

I. INTRODUCTION

Development of information and communication technologies (ICT) has had a tremendous impact on everyday life, work and education. E-learning has changed the traditional educational process and has introduced new methods that possess the potential to improve teaching and learning in higher education.

Modern students, called the Net Generation [1], use computers and Internet on a daily basis. They grew up with information technology and have different needs than previous generations of students. They are not satisfied with classic classroom courses, uniform and static learning content. Net Generation students demand greater flexibility while learning, they demand more diverse and dynamic materials with more visual elements within the learning content, not just text.

Numerous higher education institutions offer computer programming courses to their students. These courses are an important part of curriculums offered by departments associated with computer and information sciences. Several computer programming courses are offered to students within the program of Marine Electronics and Information Technologies at the University of Rijeka, Faculty of Maritime Studies in Croatia. Since these courses are among the more important ones in the curriculum of the mentioned program, it is imperative that these courses are delivered with the highest quality possible.

Teaching and learning programming is somewhat specific, requiring the teacher to take into account certain factors. In [2] some of the specific characteristics and problems are stated, which occur in computer programming courses. "Due to those reasons, teaching programming should not be exclusive and limited only to classical learning from books or to complete e-learning. The combined methods of teaching would enable them to better understand the concepts of programming and code writing. A blended (hybrid) course extends the possibilities of using various teaching methods to adapt to individual needs of students." [2] While learning programming, individual work and problem solving is of utmost importance and should not be dismissed.

When learning programming, the usage of integrated development environment (IDE) is needed. For example, for C++, Dev C++ could be used for free.

The preliminary research [2] was conducted in 2010, in which students of University of Rijeka, who attended at least one computer programming course, expressed their opinions regarding the usage of new technologies in teaching and learning programming languages in higher education. Results have shown substantial interest for new e-learning methods in teaching programming, mainly in the use of video tutorials. That was the main reason for pilot implementation of screencasts in several courses offered during the spring semester of 2011. The plan was to implement some of the new technologies in computer programming courses, i.e. Web 2.0 tools. Screencasting tool (Web 2.0 tool) was implemented first.

In this paper, the most popular free screencasting tools are researched and compared, a selected screencast tool is analyzed and the procedure of screencast creation is explained. Potential applications of screencasts in computer programming courses are examined. The paper presents students' initial experiences with the usage of screencasts in computer programming course "Algorithms and data structures", held during the spring semester of 2011 at the University of Rijeka, Faculty of Maritime Studies.

II. SCREENCASTING TOOLS

A screencast is a video capture of computer screen activity, with audio narration. Recorded videos are delivered as video files, usually via the Internet. There is an increasing trend of creating and using screencasts in various aspects of life (e.g. for presentation of products in commerce, recording tutorials), and also in higher education. Presentation, demonstration or any other computer activity can be recorded and easily published and shared online on any of the popular video sharing services (e.g. YouTube, Vimeo). "Along with cheap distribution over the Internet, the reusable aspect of screencasts makes them a low cost, highly efficient educational tool."[3] A teacher or a student who is producing a screencast doesn't need to possess any specific technical knowledge besides the basic computer operating skills.

In computer programming course which is analyzed in this paper, screencasts are used as a supplement to face-to-face teaching, with the intention of providing students with more options and flexibility in learning outside the classroom. This can be useful for part-time students, or when a student misses a lecture. Different students require different approaches, because they all have different learning styles. Screencasts can be easily adapted to students' learning styles.

All that is required to create a screencast is a computer, a screencasting tool and a microphone. Most screencasting tools allow screen capture of custom-selected area, or an entire screen. Some tools, besides the basic features, also possess additional functionalities such as the option of webcam capturing, editing screencasts after they are recorded (e.g. adding annotations), etc.

Various desktop applications and web-based (online) tools that don't require any installation exist. One of the most popular commercial screencasting applications is Camtasia Studio (http://www.techsmith.com/camtasia), and some of the most popular free applications are Jing (http://www.techsmith.com/jing), Wink (http://www.debugmode.com/wink), and CamStudio (http://camstudio.org).

While selecting a tool for pilot implementation in computer programming courses, a focus has been set on free online screencasting tools.

A. *Web-based Screencasting Tools*

A web-based tool was selected for this project, mainly because it doesn't require any installation, and students are not bound to a certain computer while they are recording screencasts. Online screencasting tools, along with other cloud computing services, are continually becoming more powerful, but they still lag behind the richer features and capabilities of desktop applications. On the other hand, online tools are very simple and easy to use, which was especially important for this project.

Several web-based screencasting tools that are available were tested, and their features were compared (Table I): Screencast-O-Matic (http://www.screencast-o-matic.com), ScreenCastle (http://screencastle.com) and Screenr (http://www.screenr.com). These tools require only an Internet connection and a web browser. Majority of these tools require the latest version of Java and Flash plugins installed within the web browser. Table I shows whether the tool requires registering and logging in, restrictions regarding the length of the recording, and the possibility of downloading the recorded video in various formats.

The online tool runs in a browser and provides quick and easy video creation and sharing. Most of the online screencasting tools are connected to free video hosting services and, if user desires, can instantly share produced video to these locations.

TABLE I. WEB-BASED SCREENCASTING TOOLS COMPARISON

Name	Login	Time limit	Download file formats
Screencast-O-Matic	No	15 min	MP4, AVI, FLV
ScreenCastle	No	-	-
Screenr	Yes	5 min	MP4

File format is selected according to a chosen delivery method – whether a user wishes to publish and share his video on one of the video sharing services, at a certain web site or inside a Learning Management System (LMS). Since the LMS that is used at the University of Rijeka (MudRi, adapted version of Moodle) already has the Flash player embedded (.flv video format), a screencasting tool with the ability of saving video in this format was selected. The aim was to have all electronic learning materials of the course in one location (LMS).

After review and analysis, Screencast-O-Matic has been selected as a tool that best fits the set criteria. Reasons in favor were: registration and login are not necessary, the ability to produce and download video in .flv format, and 15 minutes recording time limit fits the requirements of the course (Screenr only allows 5 minutes of recordings).

Screencast-O-Matic is a free, Java-based online screen recorder for one-click recording from a web browser on Windows, Mac, or Linux. The frame size of recording area can be changed (users can choose one of the preset sizes ranging from 640x480 to Full HD or full-screen, or define custom size) and positioned on screen.

B. *Screencast Creation*

Fig. 1 shows the procedure of screencast creation.

Planning is an important step that directly impacts screencast quality. The planning includes defining what is being recorded (teacher) or an assignment (student). A story script of the screencast to be recorded should be devised [4]. Preparation of all materials which are to be used in a screencast is required, as well as preparation of equipment (the microphone), checking if all equipment is working properly, etc.

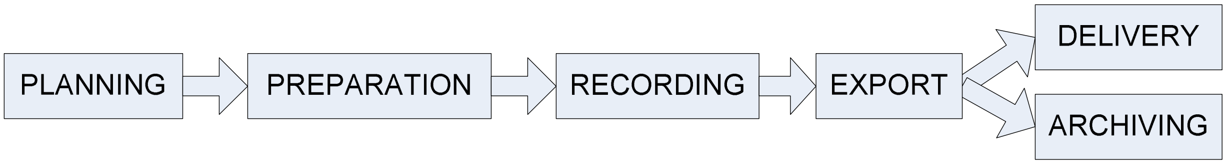


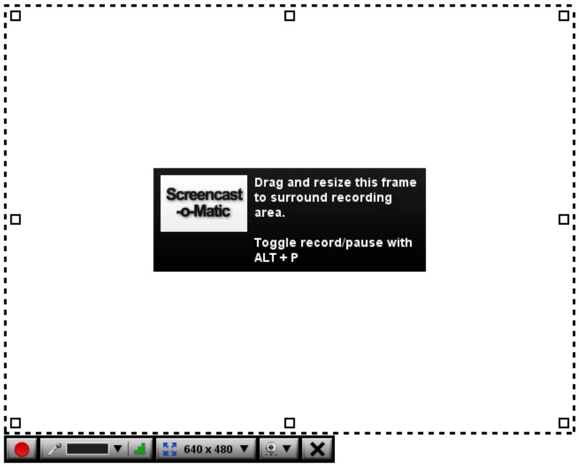
Figure 1. The procedure of screencast creation

Recording process is described based on the selected tool: Screencast-O-Matic. The steps in screencast recording are as follows. After selecting "Start recording", selecting the size of the area that will be captured, selecting an option to record audio or not, the Record button should be clicked. After clicking on the Record button, a recorder interface window is shown (Fig. 2) and the recording begins, with options to pause/continue or stop recording, restart recording, or to playback. After the end of recording, the user can share or download video or even go back to recording to add more content (to continue recording).

Export in intended file types (different file types can be offered to students, so students can transfer them to their mobile or handheld devices, eliminating the need for online viewing). Recorded videos should be archived, not just uploaded on the web, in order to be safe if eventually the video sharing service stops working, or similar.

There are some potential problems that have to be kept in mind when considering online tools, such as: unlike with desktop applications, there are no insurances that the online tool in question will be available for the extended period of time. Furthermore, if certain video sharing service is used to store the video, there is a possibility that its hosting space will not exist forever, and that the video, if it is not saved to the hard drive, will be lost. For this reason, backups are always encouraged.

Teachers could use a more advanced (desktop) screencasting tool in order to obtain higher quality screencasts. Several free and/or online tools possess the possibility of paid advanced features. For example, the selected Screencast-o-Matic offers (for 9$ per year): no watermark on export, various editing options, etc.



Drag and resize this frame to surround recording area.

Toggle record/pause with ALT + P

Figure 2. Recorder interface window

III. Potential Uses of Screencasts

Some of potential uses of screencast in computer programming courses are researched here. Creation of these videos has numerous benefits for both students and teachers [5]. Two types of screencasts will be considered: when screencast is produced by the teacher and by the student. Teacher could produce screencasts with the learning materials (curriculum explanations), and students could record screencasts with the elaboration of solved tasks.

When using screencasts in a course, students can view the screencasts whenever they want (24/7) and repeat the viewing as many times as needed. Screencast are especially useful for part-time students, who are unable to attend the courses regularly. If a student is absent, it would be useful for him/her to have a recording of the lecture, which would make it easier to catch-up on lost lectures. Furthermore, screencasts could be useful for exam preparation.

Teachers can create mini-lessons, "how to" tutorials in which they explain important topics of the course, difficult programming concepts, explanations of programming code examples, problem-solving strategies, clarifications of homework problems, correcting students homework and tasks with useful audio comments (as if students were present at consultations), etc.

Most programming concepts are easier to understand when they are explained visually. For instance, when teaching programming, teacher usually writes the code (instructions) in IDE, lectures through the process and solves problems in class. The teacher could instead produce screencasts that record, step by step, the code writing and program execution, along with detailed explanations, quite similar to what could be presented in class. Recorded videos can cover more important matters that the teacher delivered during classroom lectures, or additional tasks, for example additional materials for high achievers.

Screencasts can be used to explain topics that teachers repeat more than once (instead of explaining the topics several times) or for curriculum that is harder for students to understand. Recording and using screencasts could free up teacher's time [6]. Preparing text documents containing various explanations can be very time consuming, especially when teachers are describing a step-by-step process. Screencasts can, on the contrary, be very quickly recorded and delivered to students.

Students can create screencasts to explain solved tasks and homeworks which are graded. Instead of presenting live in the laboratory, students can record and explain code writing (all or key parts of the code).

IV. The Survey – Results of Initial Experiences

The pilot implementation of screencasts is currently being conducted in three courses throughout the spring semester of 2011 at the University of Rijeka, Faculty of Maritime Studies. The results of preliminary usage of screencasts in one of these courses are presented, a computer programming course "Algorithms and data structures", delivered during the first year of graduate study. The course covers algorithms and data structures in C++ programming language. All electronic materials for the course are available in the LMS: theory, exercise problems, homework, screencasts, etc.

The preliminary survey was conducted in March 2011, at the beginning of the semester, after three weeks of lectures, with the aim of assessing the students' opinions regarding using and creating screencasts in a computer programming course. Of 17 students that attended the course, 16 students participated in the research. Students' experiences and desire for using screencasts have been assessed. The purpose of the preliminary survey was, while respecting the students' opinions, to plan the future use of screencasts in this course.

The survey consisted of open and closed questions regarding the use of computers and the Internet, learning materials, video tutorials, screencasting tools and screencast creation. Some of the obtained results will be presented here.

Screencasts were recorded by both teacher and students. During the semester, teacher produced the screencasts and the files were uploaded to the LMS of the course, where students viewed videos. Considering that the "Algorithms and data structures" course is a follow up of the programming course that covers the basics of C++ programming language, students were given the task of creating a screencast at the beginning of semester. They were instructed on how to use Screencast-O-Matic and record screencasts, in which format (.flv) to download a file, and how to send a file to teacher via e-mail or to provide it on an USB memory stick. If students were unable to record screencasts at home, they were given an option to record them at the Faculty, because computers connected to the Internet and microphones were made available.

Students were issued different assignments: they were supposed to create a short video explaining the basics of C++ language (e.g. loops, operators, functions, 1D field, 2D field...). Their screencasts (put together) were supposed to cover the basics of C++, in order to refresh their knowledge and to facilitate future learning. This assignment enables the students not only to gain points themselves, but to learn from each other. Students' screencasts were tipically 2-7 minutes long. Fig. 3 shows an example of screencast recorded by one of the students.

A. Results

The results have shown that all students have PC's and broadband Internet access at home. 25% of the students do not have a microphone, but stated their willingness to purchase one.

Majority of the students prefer to receive the learning material in a paper form. On the other hand, all students stated that it is important that learning materials are available in electronic form online (in LMS).

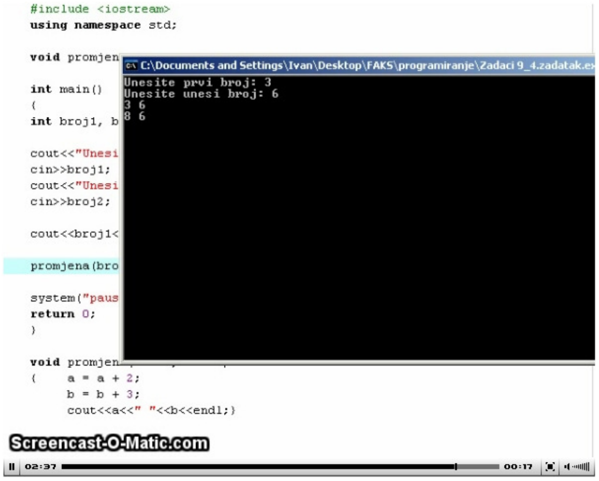
Prior to this course, only 2 students have used a screencasting tool. Even though that for most of the students, this course was the first to introduce screencasts, 75% of the students answered (a Likert-type question with five options provided: "very hard", "hard", "neither easy nor hard", "easy" and "very easy") that it was either easy or very easy to use a screencasting tool Screencast-O-Matic (Fig. 4).

Students were asked to grade (from 1 to 5) their screencasts, and the average grade was 3.56, which implies that they were relatively happy with their videos. Almost everyone (88%) thinks that screencasts enable faster learning of the curriculum and programming concepts, and the same number of students would like that more lectures are recorded by the teacher. All students agreed that it would be useful to use screencasts as an addendum to lectures, but 81% do not want the screencasts to replace face-to-face lessons

As the main benefit of screencasts, students pointed out: the simplicity of use, availability at any given moment, clear explanation of a problem within minutes, faster understanding and easier learning, and the possibility of repeated viewing

Students also stressed some negative aspects of screencasts: lack of interaction, inability to ask questions, problems with positioning and sizing of the recording frame (some parts were not visible), etc.

Following are some students' answers and comments. Students commented that the quality of screencasts varies, depending on the author. Sometimes the information can be incorrect (especially when screencasts were recorded by the students). Screencasts provide a better and simpler insight than



```
#include <iostream>
using namespace std;

void promjen

int main()
{
int broj1, b

cout<<"Unesi
cin>>broj1;
cout<<"Unesi
cin>>broj2;

cout<<broj1<

promjena(bro

system("paus
return 0;
}

void promjen
{     a = a + 2;
      b = b + 3;
      cout<<a<<" "<<b<<endl;}
```

C:\Documents and Settings\Ivan\Desktop\FAKS\programiranje\Zadaci 9_4.zadatak.ex

```
Unesite prvi broj: 3
Unesite unesi broj: 6
3 6
8 6
```

Figure 3. An example of screencast recorded by one of students

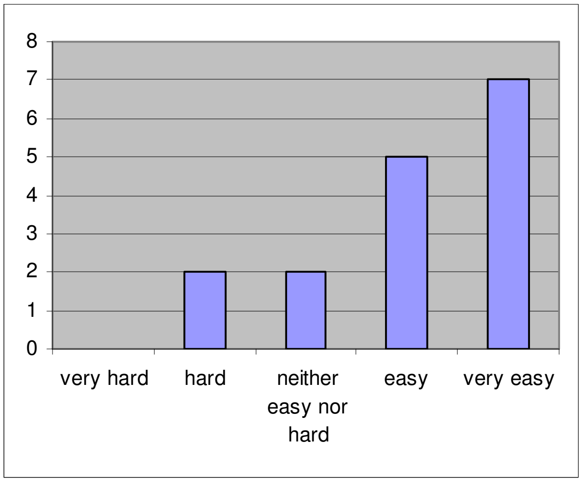


Figure 4. Students' responses on the easines of use of screencasting tool Screencast-O-Matic

static slides, and enable faster comprehension of harder parts of curriculum. Screencasts can be viewed repeatedly, and at any moment. Screencasts are an interesting approach to teaching and learning, and should be used more widely as a supplement to lectures.

The number of students in the forementioned programming course was not large enough to draw final conclusions, but it was sufficiently representative for a preliminary research, to indicate several trends. The results indicate that students show positive opinions about using screenasts in computer programming courses, and wish to use them further. Students were especially interested in screencasts produced by the teacher, demanding more.

Teacher's general impression was that students gladly accepted the new tool and sometimes even recorded the screencasts themselves, even when not required to. It was easier for them to record the screencasts containing homeworks, instead of presenting the homeworks in person.

Our results are consistent with other studies of using screencasts in education [5], [6], [7], [8], [9], [10], [11], which also show students' positive feedback in creating and/or using screencasts.

Since the general idea was to provide all of the learning material in one place (in a LMS), and not to be dependent on other sites (students prefer not to have to visit different pages or even use different accounts), a need arose and a proposition has been made to expand the MudRi LMS with additional functionalities which would enable the insertion of student's screencasts inside the e-courses (similar to homework).Those videos would be instantly available to other students enrolled in the e-course. Students shouldn't be allowed to insert screencasts without a prior approval of the teacher, or alternatively, teacher can upload the screencasts him/herself. An ideal solution would be the presence of a screencasting tool within the LMS, eliminating the need for uploading/downloading.

V. CONCLUSION AND FUTURE WORK

Although the results are preliminary, the positive opinions among students and willingness for future use of screencasts and their efficiency in computer programming courses are evident. Screencasts will be used for the whole semester, and at the end of the semester, all students who had attended the course will participate in a more detailed survey, which will investigate technology fit, screencast utilization and learning satisfaction.

After the three courses in which the screencasts will be used during the spring semester of 2011, the plan for the following year is to use the screencasts in more courses, and to intensify their use in the "Algorithms and data structures", primarilly on the teacher side (more examples and explanations).

Screencasts are especially suited for use in computer programming courses because the concepts, syntax and algorithms do not change often, therefore a recorded screencast could be used for a long time. The screencasts, either produced by teacher or by student, can be used in various courses, not only in ICT courses.

Integrating screencasts into the process of teaching and learning computer programming could increase learning efficiency, provide quicker and better understanding, motivate students to participate more actively (producing the screencasts themselves), make the courses more interesting and therefore enhance the students' overall motivation. The initial feedback of using screencasts that were produced by students was very positive. Students were less passive; they were not any longer just passive recipients of information, but creators of new learning material. The recording of screencasts might demonstrate the students' understanding of the course content and solved tasks.

From the students' point of view, it is very important that the screencasting tool is free and user friendly. The screencasting tool should be presented to students prior to assigning the task of recording a screencast. While watching a screencast, student can pause, fast-forward, skip or view the video several times. Particular attention should be paid to the sound quality, and the readability of text inside the video (if present).

Screencasts are useful, easy and quick to produce. The usage of screencasts in courses enhances the usage of ICT in higher education, and the level of ICT literacy of both teachers and students. Using screencasts in higher education can provide better insight into students' work and understanding of the curriculum, and can promote a more individual approach to studying. Teachers can quickly produce screencasts in response to student needs. Students have different styles of learning and screencasts can be easily adapted to their learning style.

Screencasts should not be used as a replacement of face-to-face teaching, but should be used as an additional supplement to the material which is available for example within the LMS, the classic "paper" material, etc.

REFERENCES

[1] D. G. Oblinger, J. L. Oblinger, Educating the Net Generation, EDUCASE, e-book, 2005, http://www.educause.edu/ir/library/pdf/pub7101.pdf

[2] S. Mohorovičić and E. Tijan, "New Technologies in Teaching University Level Programming", MIPRO 2010 Proceedings. Croatia, vol. 4, pp. 268-272, 2010

[3] M. Kopel, "The Paradigm of Screencasting in E-Learning", Advances in Multimedia and Network Information System Technologies. Springer Berlin/Heidelberg vol. 80, pp. 297-305, 2010

[4] R. Jesus and F. Moreira, "Best Practices of Blended Learning with Screencasts", Proceedings of the 2010 International Conference on Frontiers in Education: Computer Science & Computer Engineering. Las Vegas, Nevada, USA, pp. 310-315, 2010

[5] D. F. Mullamphy, P. J. Higgins, S. R. Belward and L. M. Ward, "To screencast or not to screencast", ANZIAM Journal, vol. 51, pp. 446-460, 2009

[6] S. Winterbottom, "Virtual Lecturing: Delivering lectures using screencasting and podcasting technology", Planet. no. 18, pp. 6-8, 2007

[7] L. Murphy and D. Wolff, Scaffolding active programming instruction with theoretically grounded screencasts and annotated worked examples, 2009, http://web.reed.edu/nwacc/programs/grants/final_reports08/murphy_final.pdf

[8] J. D. Palmer, " Touchcasting Digital Lecture Notes", The Journal of Computing Sciences in Colleges, vol. 26, no. 4, pp. 157-163, 2011

[9] J. Small, " Delivering library instruction with screencast software: a Jing is worth a thousand words!", paper presented to Discovery! Future tools, trends and options: 7th Health Libraries Inc. Conference, Melbourne, 2010

[10] T. Pinder-Grover, J. M. Millunchick, C. Bierwert and L. Shuller, "The Efficacy of Screencasts on Diverse Students in a Large Lecture Course," paper presented at the American Society of Engineering Education , 2009, http://soa.asee.org/paper/conference/paper-view.cfm?id=11305

[11] J. L. Falconer, J. deGrazia, J. W. Medlin, and M. P. Holmberg, " Using Screencasts in ChE Courses ", Chemical Engineering Education, vol. 43, no. 4, pp. 286-289, 2009