Assessing Large Project Courses: Model, Activities, and Lessons Learned

Maria Vasilevskaya, David Broman and Kristian Sandahl

Original Publication:

Maria Vasilevskaya, David Broman and Kristian Sandahl, Assessing Large Project Courses: Model, Activities, and Lessons Learned, 2015, ACM Transactions on Computing Education, (15), 4, 20:1-20:30.
http://dx.doi.org/10.1145/2732156

A

Assessing Large Project Courses: Model, Activities, and Lessons Learned[1]

MARIA VASILEVSKAYA, Linköping University
DAVID BROMAN, KTH Royal Institute of Technology and University of California, Berkeley
KRISTIAN SANDAHL, Linköping University

In a modern computing curriculum, large project courses are essential to give students hands-on experience of working in a realistic software engineering project. Assessing such projects is, however, extremely challenging. There are various aspects and tradeoffs of assessments that can affect the course quality. Individual assessments can give fair grading of individuals, but may loose focus of the project as a group activity. Extensive teacher involvement is necessary for objective assessment, but may affect the way students are working. Continuous feedback to students can enhance learning, but may be hard to combine with fair assessment. Most previous work focuses on some specific assessment aspect, whereas we in this paper present an assessment model that consists of a collection of assessment activities, each covering different aspects. We have applied, developed, and improved these activities during a seven-year period. To evaluate the usefulness of the model, we perform questionnaire-based surveys over a two-years period. Furthermore, we design and execute an experiment that studies to what extent students can perform fair peer assessment and to what degree the assessments of students and teachers agree. We analyze the results, discuss findings, and summarize lessons learned.

Categories and Subject Descriptors: K.3.2 [**Computers and Education**]: Computer and Information Science Education—*computer science education, self-assessment*

General Terms: Student and team assessment; Methods and tools to support team project courses

Additional Key Words and Phrases: Project Courses; Assessment; Software Engineering

ACM Reference Format:
Maria Vasilevskaya, David Broman, Kristian Sandahl, 2014. Assessing Large Project Courses: Model, Activities, and Lessons Learned. *ACM Trans. Comput. Educ.* V, N, Article A (January YYYY), 29 pages.
DOI : http://dx.doi.org/10.1145/0000000.0000000

1. INTRODUCTION

In today's computing curricula, project-based courses are essential when preparing students for real-world jobs in industry. Capstone project courses [Todd et al. 1995; Umphress et al. 2002] or large scale software engineering projects [Broman 2010; Broman et al. 2012; Meawad 2011] make it possible for student to learn about industrially important areas such as project management, team management, and communication. However, teaching large project courses is difficult. In particular, assessing student performance in terms of learning outcomes is a challenging, non-trivial task.

[1]This article is an extension of a conference paper that appeared in SIGCSE 2014 [Vasilevskaya et al. 2014]. The main extensions compared to the previous paper are (1) the peer assessment experiment, (2) analysis of relationships between activities, (3) the lessons learned section, and (4) one additional year of survey data with additional questions and analysis.

This work was supported in part by Department of Computer and Information Science, Linköping University, Sweden.
Authors' addresses: Maria Vasilevskaya, Computer Science Department, Linköping University, Sweden; David Broman, School of Information and Communication Technology, KTH Royal Institute of Technology, Sweden and EECS, University of California, Berkeley, USA; Kristian Sandahl, Computer Science Department, Linköping University, Sweden;

DOI : http://dx.doi.org/10.1145/0000000.0000000

Project courses are typically organized by dividing students into groups, where each group develops their own software product. Assessing projects solely on a group's final product does not, however, recognize the difference between individual students' contributions [LeJeune 2006]. Certain individuals may contribute significantly, whereas others fly under the radar, adding little to the project. Strictly individual assessment and grading may, on the other hand, result in too much individual focus, thus sacrificing the common group goal of producing a high quality product. As a consequence, good assessment techniques must balance between *individual* and *group assessment* aspects.

Teachers can only assess project parts that are observable. Students are usually working with little teacher involvement, making it very hard to assess what is really happening in a project. Getting teachers extensively involved in all project activities may not be possible within resource constraints. Teacher involvement can also change students' behavior: a group meeting with or without an observing teacher can be drastically different. An alternative is that the students perform the assessment themselves, using peer or self assessments [Clark et al. 2005; Wilkins and Lawhead 2000]. Clearly, assessment techniques need to involve both *teachers* and *students* and it is essential that such assessment techniques are both fair and accountable [Farrell et al. 2012].

The primary purpose of assessment activities is usually to assess student performance, that is, to generate *feedout* for grading purposes. Such assessments are said to be *summative*. Alternatively, assessments may also be *formative*, meaning that the assessment activity gives *feedback* to students for improved learning [Knight 2002]. Both summative and formative assessments are essential, but can be hard to combine.

Although tempting, it is very hard to find a single assessment activity that balances between individual and group assessment, involves both teachers and students, and is summative as well as formative. Previous work focuses on specific aspects, such as self and peer assessments [Clark et al. 2005; Herbert 2007; Wilkins and Lawhead 2000], formative vs. summative assessment [Knight 2002], or grading of written projects [Smith 2008]. Instead, we propose that a project course should include a collection of assessment activities, each covering certain aspects of the assessment landscape. In particular, we make the following contributions:

— We present a new assessment model that consists of ten *assessment activities*, categorized into three dimensions: (1) group/individual assessment, (2) teacher/student involvement, and (3) formative/summative assessment. We describe the essential parts of each activity, give rationales for the categorization, and analyze the relationships between activities (Section 3).
— We evaluate the developed assessment model and its activities by performing questionnaire-based surveys with closed questions. The surveys were conducted at the end of the course in two consequent years. In total, 162 questionnaires were collected that corresponds to 81% response rate. We thoroughly study the results and discuss findings and observed tendencies (Sections 4 and 5).
— We perform an in-depth study of using *peer assessment* [Clark et al. 2005] in large project courses. The main research questions are: (1) to what extent can student perform fair peer assessment in large project groups, and (2) are students' and teachers' assessments giving similar results? In 2013, we conducted an experiment where students and teachers performed anonymous assessments of all students in their respective group. We analyze the results of the experiment in detail and discuss how peer assessment may be applied in large project courses (Sections 4 and 6).
— From the experience of applying and refining the assessment activities during a seven-year period, we summarize *lessons learned*. We content that this information can be practically very useful for educators when improving and designing assessment activities in computing education courses (Section 7).

2. RELATED WORK

The problem of assessing individuals in project groups has been studied before. In the context of computer science capstone projects, Farell *et al.* [Farrell et al. 2012] use a model called the Team Contribution System (TCS) that collects three types of documents from the students: Contribution

Statements, Peer Group Assessment (including self-assessment), and Meeting Minutes. All documentation is collected periodically in learning cycles to provide formative assessment. For summative assessment, the peer rated data is never used directly; the final judgment is always made by academic personnel.

The TCS is implemented as an on-line tool to facilitate more frequent and more accurate input of data [Farrell et al. 2013]. In their present version, contribution statements are collected weekly. Apart from the peer-assessment, the students can quickly view their own data and there is also a meeting scheduler integrated into the portal to make sure meetings are scheduled and reported. An evaluation done with questionnaire and focus groups shows that the reception amongst students and teachers is positive.

The groups using TCS are fairly small, 3-5 students, which suggests that all group members have a good grasp of the capacity of each other, at least after a while. It is a challenge to extend it for larger groups where there is also an asymmetric aspect in peer-assessment; everyone knows the project leader's performance quite well, whereas the project leader does not know the detailed performance of each individual.

As with our model, TCS acknowledge the need for assessing groups and individuals, involving students in assessment, and giving both formative and summative assessment. However, nothing is mentioned about multiple teachers observing student meetings and teachers giving feed-back for specific artifacts. TCS collects data frequently, but is not synchronizing this with an iterative development process, which is the way formative assessment is made in the professional environment. The use of specific software engineering methods is not mandated.

Clark *et al.* [Clark et al. 2005] have inspired the development of TCS. The major difference compared to TCS is that the weighting of the final assessment is following an exact formula. For a survey of different approaches to calculate final grades, see [Lejk et al. 1996]. The course design is well connected to the software engineering domain by using two iterations and assessing both product quality as well as process adherence. Clark has more students, up to 136 students in 33 teams. Data from a program feed-back survey shows that 85-90% found the documents useful, both in the project and to give input for the final assessment.

In the Engineering Projects in Community Service (EPICS) course [Slivovsky et al. 2003], students in different years and different technical areas participate in a broad range of projects for non-profit organizations. The projects run over several semesters and the students can get 1-2 US credits per semester. The credit system encourages long-term participation. The assessment model takes input from many different sources, where peer-assessment is an important part for the individual grade. It is interesting that EPIC groups can be as large as 20 students. The lack of student's knowledge of peers is handled in two ways: the students record their confidence in their assessment and students having leadership roles get a group grade weighted into their final individual grade. Some EPICS artifacts are software-intensive, but the focus is not to teach software engineering in an intensive course. Thus, EPICS does not use iterative methods or software engineering practices. In addition, the EPICS projects focus on the design of a solution, not in releasing a tested product; some products are released after several semesters. As with TCS, EPICS assessment does not explicitly use observations made by teachers in student meetings.

There is a generally available service called SPARK^{PLUS} where students can submit peer- and self-assessment data and written comments anonymously. The service can be licensed from a non-profit organization. The research and practice about SPARK^{PLUS} has been reported by Wiley and Gardner [Willey and Gardner 2009] using two factors: the Self- and Peer Assessment (SPA) factor; and the Self-Assessment to Peer Assessment (SAPA) factor. The SPA factor can be used both for summative and formative assessment and reflects the individual's contribution to the team project. The SAPA factor shows the degree of agreement between the participant's self-assessment and the average assessment provided by the peers.

The tool comes with recommendations on how to deal with the results. A series of workshops is proposed where individuals and the entire group calibrate their assessment and reflect on the results; all with the final goal of obtaining constructive areas of improvement for the group and its members.

It has been tested in design projects with 8 participants and is appreciated amongst the students. A large majority state that their ability to perform assessments and to improve by analyzing the result has developed during the course. The discussions during workshops are specially rewarding. Put in relation to our model, this work is specialized for the student assessment part only but making it more smooth and reliable to perform.

A totally different approach of obtaining data about individuals and group work is suggested by Purto *et al.* [Purtro et al. 2014]. Their assessment model is built on a course design where all students are collaborating with a Wiki-based tool used for communication, planning, and repository. The teachers use the log file of the wiki to retrieve metrics about the activities of individuals and their interaction. The final product is assessed both from the perspective of how well it meets the requirements, but also the degree of how individual contributions from all members have been integrated in the final product. We use similar data from task management systems and code repositories as a complement to other assessment activities. We doubt the value of only using this type of formal data, or as Wilkins and Lawhead put it: ”Perhaps no tool can replace that of the instructor going to meetings of each group and simply observing what they do and how they interact.” [Wilkins and Lawhead 2000]. There is no evaluation of the assessment model in the paper.

The handbook in computer science project work [Fincher et al. 2001] covers a very wide scope of possible course settings and some practical ”mechanisms”. The book is a compilation of the body of knowledge from the Effective Projectwork in Computer Science (EPCoS) project . According to their classification our project can be described as a ”Process-based Project” with its strong emphasis of team work and industrial practice. This is one out of 11 different types of projects. The assessment of this type of projects comprises: a high degree of reflection and critique, mostly with a formative intention. Assessment data are continually collected during the project and peer- and self-assessments are often used as a component in grading.

Following the project classification, the handbook describes the ”mechanisms”, describing an abstraction of elements present in building and running a project course. For the assessment part the key mechanisms are:

— Continuous assessment
— Summary assessment
— Formative assessment
— Individual deliverables
— Group deliverables
— A list of different types of deliverables that can be assessed, for instance, documents and code
— Organization of supervisors marking: single supervisor, groups of supervisors, and use of a moderator.
— Self- and peer-assessment
— Marking by external people, for instance, industry partners and clients.

The second part of the handbook comprises a set of ”bundles”, that describe a set of different practices that can be used to solve problems or to improve a project course design. Examples from the assessment section are: (i) Let groups asses their own and other groups’ achievements and (ii) Use a fine-grained marking schema, to get better agreement between assessors.

The handbook is very easy to read and apply, but as a consequence it does not contain many references and lack evaluation data. Most of the ”mechanisms” correspond to the dimensions in our model. The reader gets the advice to combine them, but not how to balance the different ”mechanisms”. The continuous assessment is implemented in our process, see Figure 2, and our grading criteria comprise the creation of several of the deliverables listed. The bundles are more practical advice that are neither put in relation to the ”mechanisms” or each other.

Our work takes a step forward by uniting the different perspectives on assessing software engineering project courses from previous work. We deliberately try to spread the activities along the dimensions of assessing both groups and individuals, involving both students and teachers, and use the information collected for both formative and summative assessment. Depicting these three di-

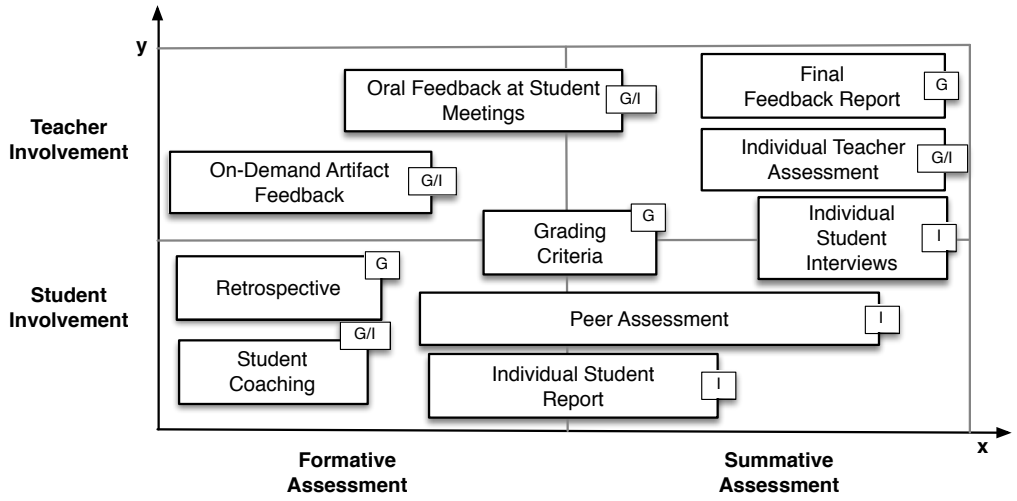

Fig. 1. An assessment model of the relationship between formative/summative assessment and teacher/student involvement. Each rectangle defines an assessment activity. The different activities are marked as group-based (G), individual (I), or a combination (G/I).

mensions together is unique, as far as we know, and gives us a valuable tool for improving our course. The model is the basis for making detailed student surveys where we evaluate how the students receive the assessment activities.

The target of our model is large scale software engineering projects, such as the company approach [Broman et al. 2012]. When designing the assessment activities we have also used the methods for process improvement as they occur in agile processes in practice. For instance, we use the retrospective meetings in Scrum, whereas other work emphasize regularity of learning cycle intervals irrespectively of how the software is developed. Just as Clark *et al.* [Clark et al. 2005], we also assess how well the students work with the processes, but we also put an emphasis on assessing *what* methods, tools, and techniques that the students are using. In our model we have represented this as grading criteria used by students to set the ambition level for the project. To the best of our knowledge, this has not been used before. Only Wilkins and Lawhead [Wilkins and Lawhead 2000] propose using input from teachers with different specialties who are attending the students' meetings in deciding the final grading, but they do not present empirical evidence. Individual student interviews are not mentioned in the work cited above.

Peer assessment can be problematic. Farrell *et al*. [Farrell et al. 2013] and Herbert [Herbert 2007] claim that they are fair and accurate, building on surveys among students (see for instance Clark *et al.* [Clark et al. 2005] and Basholli *et al.* [Basholli et al. 2013]). LeJune [LeJeune 2006] has collected data that compares students' assessments with teacher assessments. The observation by LeJune is that peer and teacher rating agree surprisingly well. To further investigate this topic we collected data from several teachers and students and made a statistical analysis.

3. ASSESSMENT MODEL

In this section, we first give an overview of the proposed assessment model and explain in general how it may be used. This is followed by a brief explanation of each assessment activity within the model. Finally, based on a real software engineering project course, we discuss usage and relationships of the different activities.

3.1. Overview of the Model

Figure 1 depicts the proposed assessment model. Each activity is categorized within three dimensions: formative/summative assessment (x-axis), teacher/student involvement (y-axis), and group/individual assessment (represented with G, I, and G/I labels). For instance, *individual student interviews* is a summative assessment activity where both teachers and students are involved, but where the interviewees (the students) are in focus. This activity is only assessing an individual student (labeled I), whereas other activities may be group-based, meaning that the whole group is assessed as one entity (e.g., the *grading criteria* activity).

This assessment model may be seen as a *template* when designing assessment activities for a project-based course. For instance, when a teacher is designing a new capstone course, he/she can pick assessment activities that are relevant in the specific context[2]. One benefit of this model is that it emphasizes different aspects of the activities. For instance, if an activity should be used for grading purposes, an individual student report (partially summative) is more suitable than student coaching (only formative). It is important to select a wide range of activities, such that different aspects of assessments are covered. It is also essential that the assessment activities are aligned with the intended learning outcomes of the course [Biggs 1996]. For example, if oral communication skills are part of the learning outcome, oral feedback at student meetings may be more important than individual student reports.

The proposed model for assessing large student projects is general and may be applicable outside the field of software engineering. However, we find it particularly useful for handling the dynamics of this kind of projects that are characterized by changing requirements, flexible management needs, and multiplicity of roles.

We have applied and evaluated this model on a larger software engineering project course (see Section 4.1 and previous publication [Broman et al. 2012] for details of the course approach). The assessment activities have evolved and been refined over a seven-year period. In particular, we have used student feedback in the form of course evaluations, emails, and informal interviews to continuously improve the activities[3].

We contend that this model is general and may be applied to other larger project-based courses. We proceed describing the essence of each assessment activity of this model and motivates its categorization.

3.2. Assessment Activities

In this section we briefly explain the assessment activities presented in Figure 1.

3.2.1. Grading Criteria. The grading criteria activity is basically a list of requirements that a student group must fulfill to get a certain grade. This list should be available to all students before the course starts. Example of grading criteria are software engineering practices that shall be adapted (e.g., continuous integration and Scrum [Schwaber and Beedle 2001]) and artifacts that shall be produced (e.g., test plan and architecture). The grading criteria are formulated by teachers, but the student group decides on a target grade. Thus, students need to estimate and balance the workload and available resources (the number of members in a group, their skills, and available time). The grading criteria can either be evaluated as fulfilled/not fulfilled [Broman et al. 2012] or by using a performance scale and a weighting factor for the deliverables to obtain a numerical fulfillment measure of the requirements [Wikstrand and Börstler 2006].

Categorization: When grading students at the very end of a course, teachers should follow up the grading criteria to check if groups are meeting the requirements for a certain grade. Consequently, this activity is a tool for *summative* assessment where *teachers* are involved in the activity. On the

[2]Note that the collection of assessment activities that we present here is neither exhaustive nor complete. There are without doubt many more suitable assessment activities that may be categorized within this model.

[3]Some of the assessment activities were not present the first years that the course was given (e.g., individual student interviews and individual student reports), whereas some other activities have been gradually improved over the years.

other hand, grading criteria can also be used by students along the course as a checklist of things to do. Therefore, this activity also contributes to *formative* assessment that is characterized by *student* involvement. Finally, this activity is a *group* assessment; the requirements apply for the project group as a whole.

3.2.2. Oral Feedback at Student Meetings. Oral feedback is a simple and efficient assessment activity. To implement it, teachers need to physically attend student meetings, such as status, project, or review meetings. At these meetings, a teacher monitors group progress and contributions of individuals. Teachers should give oral feedback to the whole group or directly to individuals.

Categorization: The main purpose of oral teacher feedback is to guide students towards learning goals. Based on this feedback, students have the opportunity to take appropriate actions. Thus, such oral feedback is mostly a tool for *formative* assessment. However, the information about how students address and identify problems may also be useful when grading students. Therefore, this activity is also partially *summative*. Finally, oral feedback contributes to both *group and individual* assessment; feedback may be given to the whole group or to separate individuals.

3.2.3. Peer Assessment. The idea of peer assessment is to exploit students' (insiders') knowledge about individual student contributions. The purpose is to complement teachers' (outsiders') understanding of the actual contributions in the project [Clark et al. 2005; Lejk et al. 1996]. When peer assessment is conducted, each student assesses other peers' performance. For example, students can be asked to assign grades or distribute some extra points from a common budget [Wikstrand and Börstler 2006]. This data can then be used as an input for the final grading of students.

Categorization: As discussed in Section 2, peer assessment can be applied for both *summative* and *formative* assessments. Only *students* are involved in this process. The teacher's role is purely organizational. Finally, peer assessment is used to grade *individuals*, since, when peer assessment is carried out, each student assesses all other *individuals/peers* of his/her group.

3.2.4. Student Coaching. Student coaching accounts for all forms of peer assessment activities conducted by students within project groups. In particular, students are encouraged to assist each other by giving internal feedback. For example, students can give short tutorials, conduct hands-on sessions within their groups, or review produced artifacts.

Categorization: Similarly to peer assessment, student coaching includes evaluation and monitoring by peers. However, student coaching included in our assessment model is a tool for pure *formative* assessment. Thus, the goal of student coaching is to give/receive feedback during the course rather than to grade work of others. Student coaching is conducted without teacher intervention and is, therefore, categorized as a *student* only activity. Both groups and individuals can be targets of student coaching. Therefore, we categorize it as a tool for both *group and individual* assessment.

3.2.5. On-Demand Artifact Feedback. On-demand artifact feedback means that students may request, at any time during the course, feedback on their artifacts. Depending on the size of the artifact and timing in the course, the teacher may give oral feedback, written feedback, or both. The purpose of this activity is to give students early feedback, so that the quality of these artifacts can be improved during the course.

Categorization: As mentioned above, students are not obliged to request feedback on their artifacts. The final grade is neither affected by requesting feedback nor by the quality of intermediate versions of the artifact. This purely *formative* design of on-demand assessment technique is deliberate: by not including the summative aspect, the intention is to encourage the student to ask all kinds of questions at an early stage. If the early on-demand feedback also was summative, there is a large risk that students would hesitate to ask the question. As an alternative, an early artifact feedback can be compulsory (not on-demand) and summative. This kind of assessment task can also be useful, but we have not evaluated this approach in this work.

This activity, when using formative on-demand artifact feedback, requires both *teacher* and *student* involvement: it is the student who requests feedback, whereas the teacher gives the actual

feedback. Also, reviews can be requested by both individual students and their groups. Therefore, this procedure contributes to both *group* and *individual* assessment.

3.2.6. Individual Student Report. In this activity, each student prepares an individual report where he/she reflects on his/her own performance during the course. This report consists of two parts: (a) what the student claims he/she has contributed to the group, and (b) what are the most important things that he/she learned.

Categorization: An individual report is meant to be used by students as a self-evaluation tool. As a consequence, students have the opportunity to improve their own learning. Hence, a student performs *formative* self-assessment through an individual report. Additionally, this document is used by teachers to analyze individual student contributions, which is also used for student grading. Therefore, individual student reports are also used for *summative* assessment. Naturally, only *students* are involved in this activity because each student writes the report individually.

3.2.7. Individual Teacher Assessment. When a team of teachers is involved in the assessment of project groups, each teacher will have their own understanding of the different groups' performance and results. It is therefore important to take into account individual teachers' opinions when deciding on final grades. Therefore, we introduce an individual teacher assessment activity into the model, where each teacher first independently assesses groups and individual students. These individual assessments, including individual grading, are then discussed by all teachers together, so that a final grade can be agreed upon[4]. Note that this method encourages that several teachers are making the final assessment together, but that they initially do the assessment individually. If the course only has one teacher, there will be an individual assessment, but the assessment looses the benefit of having several different opinions and viewpoints from different teachers.

Categorization: Individual teacher assessment is carried out at the end of the course. Therefore, it contributes only to *summative* assessment where only *teachers* are involved. As previously mentioned, both *groups and individuals* are assessed in this activity.

3.2.8. Final Feedback Report. Written teacher feedback is an efficient way of communicating student performance [Smith 2008]. In our assessment model, this technique is represented as a final feedback report that is handed out at the last meeting. This report consists of two main components: a grading table and closing comments. The grading table consists of different grading areas, where each area is given a specific grade and a weight factor. This table is then used to calculate the final group grade. The closing comments include detailed feedback for each grading area. In particular, closing comments specify who performed the assessments (which teachers), what information was used (what documents and observations), positive and negative observations, pointers for further improvement, and rationale of the given grade with respect to other grades.

Categorization: The final feedback report is categorized as a tool for *summative assessment* because the received feedback (closing comments) cannot be used by students for improvements within the same course. Only teachers are involved in the assessment and the report concerns only the whole group, not individuals.

3.2.9. Individual Student Interviews. It is always challenging to evaluate individual students when they work in large groups. Direct evidence of student work is often invisible for teachers. Therefore, we introduce an additional assessment activity where individuals are the main focus. At this event, each student is briefly interviewed at an informal meeting (10-15 minutes) with respect to their role in the group. Typically, only a subset of all students is interviewed. The selection can be based on the individual reports or information received during student meetings. The main purpose is to judge if individual students have contributed more or less than the average student in the group.

Categorization: All interviews are conducted at the end of a course. Therefore, this activity is only meant to contribute to *summative assessment*. Both *teachers* and *students* are involved in the interviews as interviewers and interviewees, respectively. The main goal of this activity is to gain better

[4]This activity has similarities to the Delphi method, which was originally designed for expert group judgments.

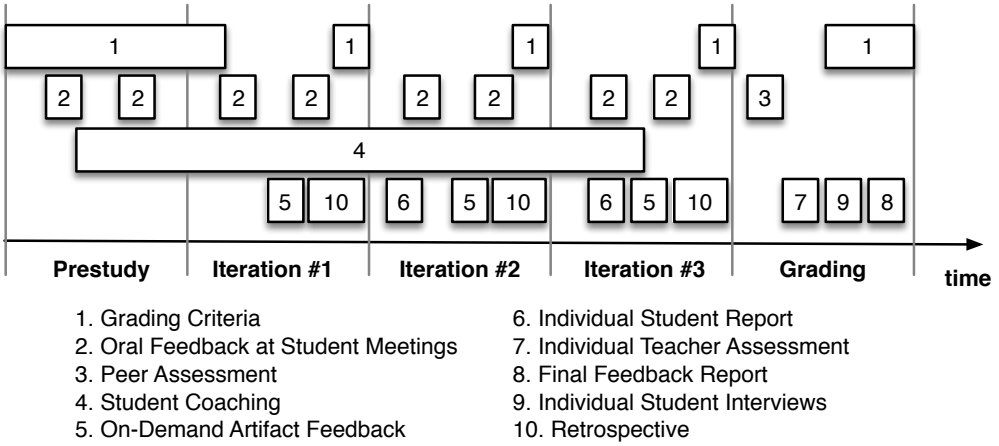

Fig. 2. Usage of assessment activity within the software engineering project course (schematic illustration)

understanding of individual student contributions. Therefore, this activity contributes to assessment of *individuals*.

3.2.10. Retrospective. If the project is organized into iterations (for instance by using an agile approach such as Scrum), an important assessment activity is to perform one or more retrospectives. In this activity, students discuss how they have been working during the last iteration. The main questions that should be discussed are: (1) what was good in the iteration? (2) what did not work well?, and (3) how can we improve? Retrospectives can also be used even if the project course is not based on iterations (for example by having the retrospective in the middle of the course).

Categorization: A retrospective is meant to be *formative*. It is important for the students to acknowledge what is wrong early in the project so that they can improve. Retrospectives are pure student activities, although teachers may help the students to organize their first meeting. The activity may be seen as a self-assessment of a group. In particular, retrospectives are meant to improve the process (the way of working) for the whole group, not assessing specific individuals.

3.3. Usage and Relationships

As mentioned earlier in this section, a teacher may select assessment activities that are relevant in a specific context. We argue that the main goals for a teacher, when selecting a subset of activities, are (1) to cover different aspect of assessment (formative and summative assessment, teacher and student involvement, and individual and group assessment), and (2) to select a set of activities that are aligned with the intended learning outcomes of the course. When applying the activities described above (or a selected subset), it is important to understand that they are not mutually exclusive, but rather interrelated and complement each other. Moreover, their particular usage is defined by their concrete implementation within a given course. In this section, we demonstrate an example of the usage and relations of the activities implemented in a large software engineering project course [Broman et al. 2012].

Usage: Figure 2 depicts a timeline for the software engineering project course divided into five periods: prestudy, three iterations, and the grading period. Each activity from the assessment model is placed along this timeline. Some activities are used several times, but only during short time period (e.g., oral feedback at student meetings). Other activities are used continuously during longer periods of the course (e.g., student coaching).

The grading criteria are introduced to students in the beginning of the course and are also used by students several times during the course. In particular, the grading criteria are heavily used by

students during the pre-study phase. At this point, these criteria help students to plan their work and to distribute responsibilities. When the first iteration begins, students still need to use the grading criteria to get started. We have also observed that students check the grading criteria at the end of each iteration. When the last iteration ends, students check with the grading criteria one more time to control whether the desired criteria are satisfied. Finally, the grading criteria are also used by teachers during the grading period.

The student coaching activity is used almost over the whole course period, except for the grading period. The absence of student coaching in the beginning of the pre-study phase and at the end of iteration 3 is explained as follows. At the very beginning of the pre-study period, students need some time to get to know each other to be able to understand where they can contribute with coaching. At the very end of iteration 3, students are already well prepared and do not need coaching.

The rest of the described assessment activities take place at specific time points. Students write and hand in their individual reflection reports twice: in the beginning of iteration 2 and at the end of iteration 3. Oral feedback at student meetings and on-demand artifacts feedback are conducted by teachers and students at several occasions during the course. According to our experience, the oral feedback is carried out approximately two times during the pre-study and iterations phases, whereas on-demand artifacts usually take place at the end of each iteration when students have prepared artifacts for review. As recommended by iterative agile methodologies, the retrospective meetings are scheduled at the end of each iteration.

Besides the grading criteria, there are four assessment activities that are used during the grading period: peer assessment, individual teaching assessment, individual student interviews, and final feedback report. The first step is to perform peer assessment[5]. Thereafter, teachers individually assess groups and students, followed by interview candidate selection. Based on this data, students are invited for interviews. Finally, all this information is used for writing final feedback reports.

Relationships: Figure 3 summarize our analysis of how the proposed assessment activities are related to each other. An arrow means that an activity influences another activity. For instance, an arrow from the grading criteria (GC) to retrospectives (R) means that the former provides some input for the latter, and, thus, GC influences R. Note that this analysis is not exhaustive since we have pointed out only the most distinct relations. Our goal is to demonstrate the tight interrelations

[5]Note that peer assessment is part of the proposed model, but currently not used in the course by [Broman et al. 2012].

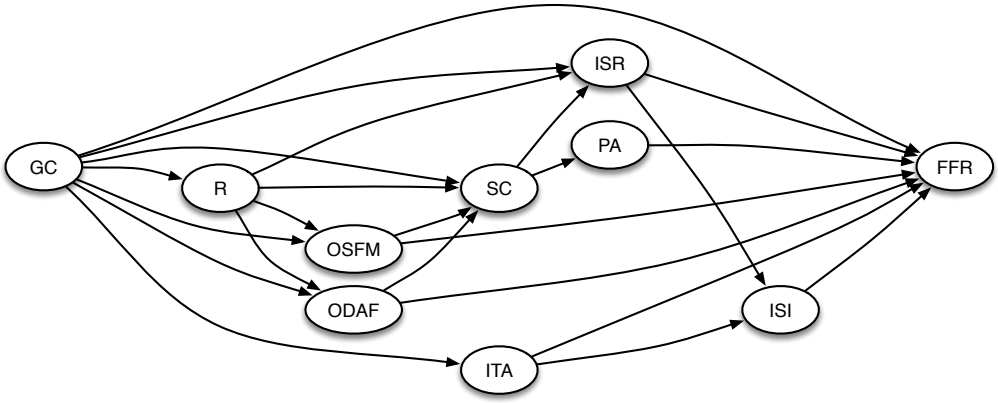

Fig. 3. Relationships between different activities. Note: Grading Criteria (GC), Oral Feedback at Students Meetings (OFSM), Peer Assessment (PA), Student Coaching (SC), On-Demand Artifacts Feedback (ODAF), Individual Student Report (ISR), Individual Teacher Assessment (ITA), Final Feedback Report (FFR), Individual Student Interviews (ISI), Retrospectives (R).

between assessment activities that should be accounted when they are used for assessment. Figure 3 shows that the grading criteria (GC) provides direct input a the largest number (7) of the activities. This is quite natural since GC are heavily used during the whole course by students and teachers (see Figure 2). At the same time, the final feedback report (FFR) activity relies on inputs from 7 different activities that capture diverse aspects of assessment. As a consequence, when the proposed assessment model is used, teachers are equipped with different sources of knowledge when conducting their final grading.

4. DESCRIPTION OF THE STUDY

In this section, we first present the course that is studied, followed by the design, data collection, analysis method for the evaluation of the assessment model. Finally, we describe the design and data collection process for the peer assessment experiment.

4.1. Context

The assessment model presented in the previous section is the result of a seven-year period of experimentation with a software engineering project course (see [Broman et al. 2012] for a detailed description of the course approach). The course is given as a master's level course. The students come from different curricula and countries. In 2013, 113 students participated in the course and about 13% of the students were females; in 2012, 90 students participated and about 15% of them were females.

As a part of the surveys, the students were asked to indicate their age, study program, and experience in software development and software industry. Among students of both years about 77% were 20 - 24 years old, 22% were of age 25 - 29 years, and only 1% of students were over 30 years. 76% of students were local master students; whereas 24 % were exchange students. Additionally, 83% of students had 6 months of experience in software development, 8% have 6 - 11 months of experience, and about 4% were more then 2 years experienced in software development. The experience of students in software industry was slightly different in 2013 and 2012 years. In particular, in 2012, 88% of students had less than 6 months of experience and 12% had 6 - 11 months of experience; whereas in 2013, 74% of students worked in industry less than 6 months, 11% worked in industry for 6 - 11 months, 11% had 1 - 2 years of experience, and 4 % had more than 2 years of experience in software industry.

The major goal of the course is for students to obtain fundamental understanding of a larger software engineering project and experience its challenges. Students should also learn organizational, process, and communication perspectives. Initially, students are divided by lottery into 3 - 4 groups of 20 - 35 students. One reason for having such large groups is that most functions and roles in a real company shall be present. All groups get the same task and the same customer representative. In both 2013 and 2012 the customer was an external product development company developing client devices for digital television. The course runs during an entire semester and comprises two phases:

(1) A prestudy phase where the students organize the group, assign roles, start education, interview the customer, and create a preliminary version of the architecture. The prestudy ends with a *tollgate* meeting where the continuation of the project is formally determined.
(2) A series of 2-week iterations for implementation, prototype evaluation, testing, documentation, and configuration management. The project ends with a demonstration and experience seminar with all groups.

The students are free to organize themselves. However, we recommend them to organize into smaller, cross-functional teams after the prestudy and use a SCRUM-based process framework.

In total, four teachers/teaching assistants are involved in the course. During the course, each group meets one teacher for approximately one hour at least once a week. Depending on their needs, some individuals and small groups also arrange additional short meetings with the teachers (e.g., to ask questions about project planning and requirements documents). During the course the teachers meet once a week to discuss the progress of the groups; very often these meetings result in

formulation of formative assessment to help the groups, or a particular group, to keep on going in the right direction. The teachers share a common virtual workspace where observations and students' presence in meetings are recorded. The summative assessment (grading and writing final feedback reports) is done in 2 - 3 consequent days where all teachers are involved. Performing all these activities may seem costly with a lot of communication overhead, but when all teachers know their role in the course our experience is that the course can be executed quite efficiently.

The artifacts created by the groups are handled in a revision control system on a server provided by the university. The teachers have full access to this. In addition, the students submit a weekly report on major activities in the groups. Individually, the students submit time-reports every week and individual reflection reports twice during the course. This information is used both for formative and summative assessment.

The course is, according to the official student evaluation, very popular and students are satisfied with their learning. As is the case with many different institutions, there are strong formal requirements on the examination of courses when it comes to fairness and grounding. It is considered especially important that all grading moments and the criteria are known by the students 4 months in advance, when the course selection has to be done.

4.2. Assessment Model: Design, Data Collection, and Analysis Method

The evaluation of the activities included in our assessment model is based on two questionnaire-based surveys with closed questions. The students answered each statement by selecting one of the six options: strongly disagree, disagree, neutral, agree, strongly agree, and N/A. The survey was conducted twice in December 2012 and in December 2013. The blanks with questions were handed out at the end of the course in both cases. In 2012, the answers from each group were collected in a closed envelope by an assistant not teaching the project and the group identities were randomly coded. In 2013, the answers were collected from all groups altogether in a common pile closed in an envelope. No identification of individual students was possible in both cases. The envelopes were first opened when the final grades had been already announced and the final feedback reports had been given to the students. 161 (72 in 2012 and 89 in 2013) questionnaires were collected in total that corresponds to approximately 81% (82% in 2012 and 79% in 2013) response rate.

To process the obtained data, we used descriptive methods of SPSS (Statistical Package for the Social Sciences). The answers of students were analyzed as nominal scale measurements. There were 39 questions in the survey of 2012, and 30 questions in 2013. In both surveys, 7 questions were about students' background (year of the course, age, sex, industrial experience, software development experience, curriculum, and result from theory exam). In the survey from 2012, 15 questions were related to the general course evaluation, and the remaining 17 questions were related to evaluation of the presented assessment model. In 2013, we did not reuse the 15 questions that were related to general course evaluation. We also added new questions to investigate further the findings from the previous year. In particular, in the survey from 2013, 13 questions were reused and 10 questions were added. Among all 42 questions (see Appendix A), 14 are detailed in this paper.

Peer assessment was not explicitly part of the course described in Section 4.1. Instead, to study applicability of peer assessment techniques in large groups project courses, we launched an experimental peer assessment at then end of the software engineering course. The main research questions are: (1) to what extent can student perform fair peer assessment in large project groups, and (2) are students' and teachers' assessments giving similar results? To answer these questions, in 2013, we conducted an experiment where students performed anonymous assessment of all students in the group they belonged to. Each student obtained a sheet of paper containing a list of all students in the group. For each student in the list, they were then asked to select one of the following:

— *Better* – I think this student has performed better than the average student of the company.
— *Equally* – I think this student has performed as an average student of the company.
— *Worse* – I think this student has performed worse than the average student of the company.
— *N/A* – I cannot estimate the performance of this student.

Similar, teachers were also asked to perform the same task for all student groups. Students conducted the experiment at the end of the course (the last mandatory meeting). Teachers carried out the experiment when the final grades and feedback reports had been already handed out to the students. In total, 99 forms were collected that corresponds to approximately 80% response rate. For both the surveys and the peer assessment experiment, it was clearly stated in the forms that this activity was optional, anonymous, and that it would not affect any grades.

4.3. Validity Threats

Runeson and Höst [Runeson and Höst 2009] identify four areas of validity threats for a case-study, such as ours:

— *Construct validity*. Are we measuring the phenomena we intend to measure? There is always a risk that the students misinterpret the questions in a survey and answer different questions than intended. We have tried to remedy this by asking several questions around the same assessment methods and in general the students are consistent in their answer. Asking many questions is a double-edged sword; it might lower the quality of the answers by fatiguing the subjects. To meet this we have tried to be careful in keeping the number of questions at a reasonable level. Our students are very focused on examination and grading and get informed on lectures, in meetings, in mail, and on the web. Questions of clarification are repeatedly asked during the course, so we believe that the meaning of the different assessment methods and grading levels in the questionnaire are well understood.
— *Internal validity*. Are the causal relationships true or are there external factors that can explain the observations? We have tried to eliminate bias by ensuring anonymity, giving enough time to answer the questions, and obtaining a fairly high response rate. In the peer assessment investigation, the survey was done before the students got their grades to avoid confounding the answers with too positive or too negative feelings about their learning. The questions in the evaluation of the assessment model are clear and straightforwardly posed about a certain method having a certain effect. So we are have good reasons to believe that the cause-effect data is true, but we are also humble to the fact we might have missed *additional* causes. In that case we have to rely on our experience and the care made when designing the questions. In the peer assessment study, the teachers' assessment might be biased against only observing factual data, not the social aspects, which were probably taken into account by the students. To some degree the observations can be triangulated by using the texts of the individual reports.
— *External validity*. To what extent is it possible to generalize the findings? For a study like this it is impossible to claim generality; the goal is to allow for analytical *transferability* of results in similar course settings. We have thus tried to describe the context of our study at a reasonable level of detail and also put some of the characteristics in relation to other work to facilitate knowledge transfer.
— *Reliability*. How dependent are the findings of the researchers? Would another team of researchers come to the same results? The survey questions are provided and are not difficult to re-use in another, similar course. The basic meaning of the different activities are described, but would require some work of adaptation if the study is replicated with another course. Working with our model for improvement of courses can be done without much adaptation once the reader have gotten our examples of how we classify different activities. We have an internal replication by asking several questions in two years, with a slightly different course organization. This does not demonstrate external replicability, rather the re-usability of the instruments used, which is a small, but necessary, prerequisite for reliability.

5. ANALYSIS AND DISCUSSIONS OF THE ASSESSMENT MODEL

In this section, we analyze and discuss the results of the surveys used for evaluation the assessment model. The results are presented as bar charts, depicted in Figures 4, 5, and 6. We discuss the results in terms of the previously introduced dimensions.

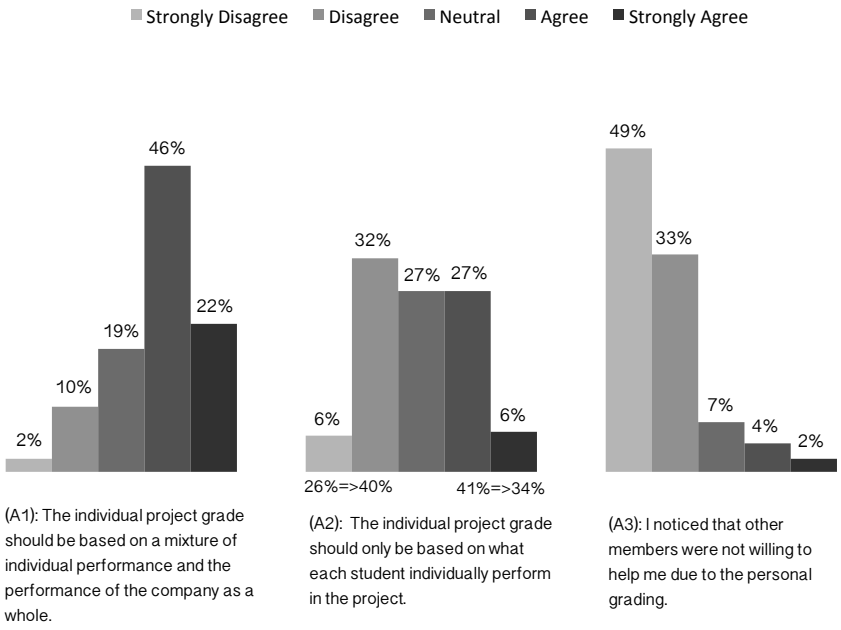

Fig. 4. Survey results for questions related to group and individual assessments. Missing and N/A answers are omitted for the sake of clarity. Each chart displays the total results for both 2012 and 2013. To show significant changes between the two surveys, the numbers below the bar charts show how the results have changed from 2012 to 2013, denoted as X% $\rightarrow$ Y%, where X and Y represents a sum of *strongly agree* and *agree* answers or a sum of *strongly disagree* and *disagree* answers for 2012 and 2013 respectively.

5.1. Individual and Group Assessment

Although students work together in group-based courses, each student must be given an individual grade at the end of a course. As previously described, the assessment activities in our model (Figure 1) contribute to both group and individual assessment. A natural question then arises whether individual or group performance should be emphasized when deciding on the final grade.

According to the results of statement A1 in Figure 4, a strong majority believes that a mixture of individual and group-based performance gives fair grading. In particular, 68% of the students agrees and strongly agrees to statement A1, whereas only 12% of the students disagree and strongly disagree.

The bar chart for statement A2 in Figure 4 shows that 33% of students think that individual project grades should only be based on their individual performance, whereas 38% disagree. These results show that there is a tendency among students to believe in mixed individual and group based assessments, although approximately one third of students (33%) still thinks that the assessment should be individual. Comparing student answers for A2 from 2012 and 2013, we observe a noticeable change among students' opinions. In particular, a majority of students in 2012 believed that individual grades should define their final grades, i.e., 26% disagreed with A2 against 41% who supported A2, whereas in 2013 a bigger portion of the students think that their final grade should not be based on their individual performance, i.e. 40% against 34%. A plausible, but not very strong explanation might be found in the answer for statement B3 in Figure 5 where the students of 2012 had higher confidence in the teachers' understanding of the situation in the groups. Thus, the stu-

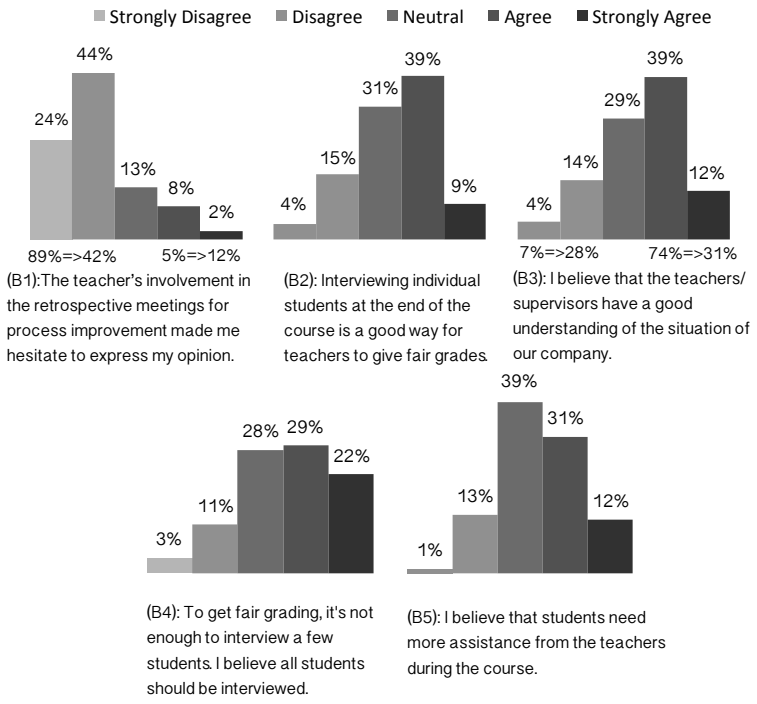

Fig. 5. Survey results for questions related to teacher and students involvement. The figure can be interpreted in the same way as Figure 4.

dents had a better feeling of being seen as individuals which would make it more worthwhile to go for individual grades only.

Collaboration among students is one of the primary learning goals in many group-based courses. Too strong focus on individual grading may, however, force students to focus solely on their individual performance, thus preventing fruitful collaboration. Fortunately, the results of this study do not point in this direction; 82% of the students confirm that such prevention of collaboration did not exist (statement A3).

In addition to the surveys, we have analyzed how much time students spend on either individual or group based work. The analysis is performed on 72 time reports that were submitted by the students during 2013. The rest of the reports (41) have been omitted due to incompleteness. For instance, students have not made any note about their collaborators. Our analysis shows that on average only 35% (the standard deviation is equal to 10%) of the student time is spent on individual work. The rest of the time is spent on collaboration with other students or group work (e.g., scrum teams, analyst teams, tester teams). These results confirm our intuition of the student surveys in that the final grade should be based on the mixture of both group and individual performance. Furthermore, according to our analysis, the amount of group work prevails the amount of individual work. As a consequence, it is hard to isolate individual contributions of students without taking the group's total contribution into consideration.

5.2. Teacher and Student Involvement

To enable fair and valuable student feedback, teachers need to be present at certain occasions where they can observe the students' work. In the project course of this study, teachers are involved in

student planning sessions and retrospective meetings, where students plan their work and discuss emerging problems.

A disadvantage with this approach is that students may hesitate to express their opinions when a teacher is present. Consequently, this could prevent students from improving their way of working and learning. Statement B1 in Figure 5 shows, however, that this threat is quite mild; 68% of the students state that the teacher involvement does not restrain them from honest discussions. Observe, that in 2012 students were more certain regarding statement B1 than in 2013. In particular, only 42% of students disagree with B1 in 2013, whereas 89% disagreed with the same statement in 2012. Thus, this question should be further monitored in forthcoming years.

When only teachers are involved in the assessment, it is extremely difficult to get a good understanding about individual students' contributions. This may lead to unfair summative assessment. To tackle this deficiency, our assessment model includes several activities with both student and teacher involvement (see Figure 1). In particular, we want to draw extra attention to the assessment activity of individual student interviews (see Section 3). The results of statement B2 clearly shows that individual interviews are appreciated; 48% of the students agree that this activity is a good way to assess and give fair grades (39% were neutral). Recall, that not all students go through the individual interviews: teachers select only those students whose performance in the course is unclear. For example, 12 students out of 113 were interviewed in 2013. As the result of these interviews, 5 students obtained the grade higher than the company grade and 2 students obtained the lower grades. The majority of students (51%) points out that all students should be interviewed (see statement B4). It is also desirable for teachers to be able to interview all students, however, this may not be feasible due to resource constraints.

Finally, the result of statement B3 shows that 51% of the students believe that the teachers have a good understanding of the situations within the group. Although the students *belief of teachers understanding* does not express *teachers' actual understanding*, we conclude that the teacher involvement in the course is vital and that the involvement suggested in this approach does not affect the team work negatively.

Note that the fraction of students who answer positively on B3 has significantly changed from 2012 to 2013. In particular, a much stronger majority of students (74%) agreed with B3 in 2012 compared to 2013 (only 31% of students agreed with B3 in 2013). This change may be caused by the increase in the number of students attending the course. In 2012, there were 3 groups, whereas in 2013, there were 4 groups. Both years, the same number of teachers were available (4 teachers). As a consequence, fewer teachers were attended student meetings in 2013, compared to 2012. However, we think that the reason for this serious change should be carefully investigated in further studies.

The majority of students in 2013 (43%) showed that they would like to have more assistance from teachers (B5). However, blindly increasing the teaching support may contradict the learning goal of training students to work independently.

5.3. Formative and Summative Assessment

One possible consequence of too much focus on summative assessments is that students focus on obtaining good grades rather than effective learning. Our survey shows, however, that 68% of the students think that doing a good job and learning are more important for them than getting good grades (statement C1 in Figure 6).

The individual student report is both a summative and formative assessment activity (see Figure 1). Our experience (as teachers) confirms that an individual report is a useful tool for teachers to get better understanding of individual student performance. As a tool for formative assessment, an individual report is meant to be an instrument for self-reflection, helping students to obtain insights of the group and their individual role. Such self-reflection may also help students improve their way of working. The results of our survey show, however, that this goal is not completely achieved. Only 18% of the students confirm that writing an individual report helps them to learn (statement C2 in Figure 6), whereas 53% of the students disagree with this statement.

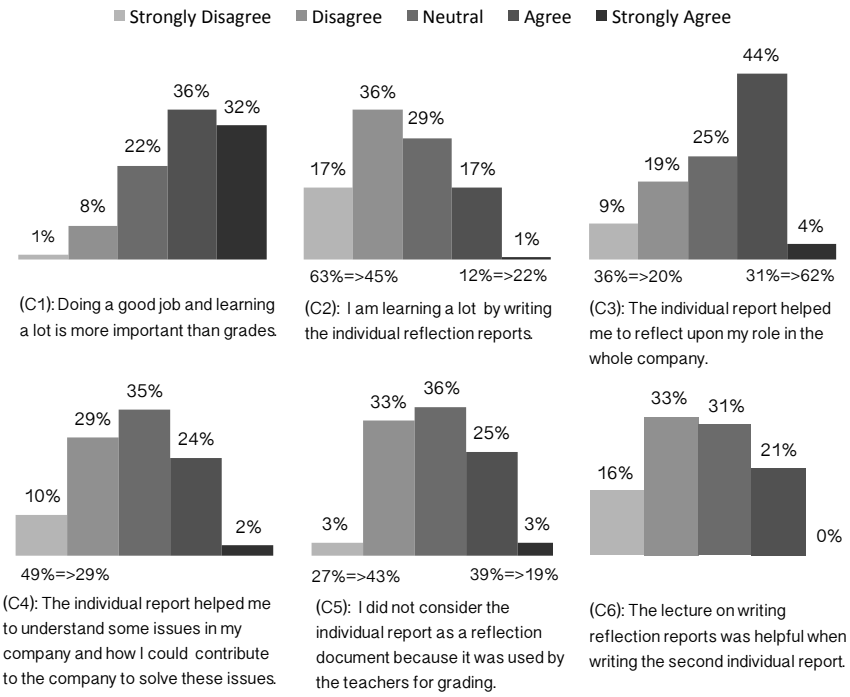

Fig. 6. Survey results for questions related to formative and summative assessment. The figure can be interpreted in the same way as Figure 4.

Statements C3 and C4 show that an individual report is more successfully used by students to reflect upon their own way of working (C3), compared to the whole group (C4). Indeed, our initial analysis of individual reports shows that students heavily reflect on their roles and shifts of their roles. Students also often reflect upon such aspects as communication insights, learned technologies, and the importance of planning. As examples, we show a few student reflections below:

"At the beginning, I was not happy with the role shift that was basically enforced by the company management. In the end, it was proven to be a good move that has actually benefited the company a lot . . . "

"In order to get the job done, I had to communicate more and better with the other developers in my scrum team."

"Something that has been quite difficult when working in the SCRUM team is how to work with different tasks that affect each other. Communication is extremely important in this situation because people sometimes understand things differently . . . "

"In the beginning of the project, we did not know where we were going, but in iteration 2 and 3 I felt that our project team was much more efficient than before. "

"I also learned that there was too much time wasted on meetings, which reduced the time left for real work, even though of course meetings are important to keep the project together . . . "

Still, some percentage (28% and 39% for statement C3 and C4, respectively) do not believe that individual reports are useful as a formative self-assessment activity.

Before we performed our survey, our hypothesis was that the use of an individual report for summative assessment may prevent students from giving honest reflections. This hypothesis is partly rejected. In particular, 28% of the students did not consider the individual report as a reflection document because it was used for grading (statement C5 in Figure 6), whereas 36% stated the opposite.

It is important to notice that the overall perception of individual reports as self-reflection tool has significantly changed from 2012 to 2013. In particular, students begin to appreciate the individual report activity more in 2013 than in 2012 (see the historical numbers below the charts for C2–C5). Comparing with 2012, this can be an affect of an additional lecture about the reflection process, which was given in the middle of the course in 2013. This lecture was given to better prepare students for self-assessment. However, the results for the corresponding statement C6 do not confirm completely our belief in the usefulness of this lecture. Only 13% of students answered positively on statement C6, whereas 31% disagreed.

Similar results about self-assessment have been reported by Hernandez [Hernández 2012] in a study of undergraduates in Hispanic studies. Both student and teacher perspectives were considered. When asked to rank the most frequent purpose of performing assessments students responded "grading students" in 24% of the cases, whereas teachers answered "grading student" in 13% of the cases. The study also investigated what students do with the assessment. Even though teachers gave feedback by marking weak points and suggesting improvements, many students just took part of the summative information of the assessment. To make the students learn more from assessments, the assessments needs "feedforward", that is, information about of what to do with the feedback. To some degree, we succeeded with the lecture introduced in 2013, but there is still more to be done. The inherent complexity of formative assessment is well described in the theoretical work of Yorke [Yorke 2003] who provides a multi-faceted tool for reasoning about course improvement. We are especially intrigued about making the students become independent learners as is required in their professional life.

6. ANALYSIS AND DISCUSSIONS OF THE PEER ASSESSMENT EXPERIMENT

In this section, we evaluate suitability of peer assessment techniques for summative assessment of students in large project courses. In particular, we address the two research questions introduced in Section 4.2.

6.1. To what extent can student perform fair peer assessment in large project groups?

Our hypothesis was that students would not have sufficient information to grade all peers within large groups (20 - 35 students). Thus, some students will be graded by too few peers in comparison with others. This might be problematic when deciding a fair grade based on the peer assessment.

In our experiment, the students were asked to selecte the *N/A* option when they could not grade a peer. Only 18% of all students' answers were *N/A*. There is no student who received *N/A* from all his/her peers. On average, each student was graded (i.e. obtained the *better*, *equally*, or *worse* grade) by 61% of his/her peers.

Thus, it seems like students believe that they are able to provide grades for their peers, that is, students think that they can perform peer assessment in large project groups. However, it is not possible to make a conclusion about the degree of "fairness" from these results. In the rest of this section, we analyze the fairness aspect by comparing peer assessment-based grading with grades assigned by teachers.

6.2. Are students' and teachers' assessments giving similar results?

To answer this question, we consider two aspects:

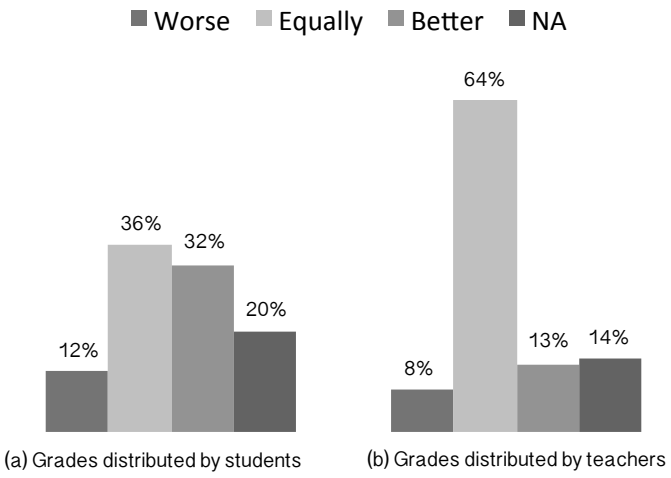

Fig. 7. Comparison of peer and teacher assessments

— The portion of *better*, *equally*, and *worse* answers given by students and teachers.
— The correlation between students' and teachers' grades.

Figure 7 shows the distribution of the *better*, *equally*, *worse*, and *N/A* answers given by students (the left chart) and teachers (the right chart). Figure 7 is built out of 2457 grades assigned by students to their peers and 342 grades assigned by teachers to students. This figure shows that comparable percentage of the *better* grade is assigned by students and teachers (12% and 8%), whereas there is a big difference in distribution of the *better* and *equally* grades. In particular, 64% of all grades given by teachers correspond to grade *equally*, whereas this portion is equal to 37% when assessed by students. At the same time, teachers assign grade *better* in 13% of all assigned grades, while this portion is equal to 33% among students. Hence, teachers tend to have a stronger opinion about giving out grades (fewer N/A) compared to the students, but students think that more students deserve either higher or lower grades than the company/group grade. The latter is not surprising; students ought to have more insights and opinions about how good other students in their group have performed.

To conduct correlation analysis we calculate average grades per each student assigned by teachers and students. Other aggregation functions could be considered [Lejk et al. 1996]. However, we think the average grade is representative enough for our analysis. Thus, the considered two correlated variables are average grades assigned by students and average grades assigned by teachers. The average grade assigned by all students to peer i is computed as follows:

$$\bar{G}^S(i) = \frac{Sw(i) + 2Se(i) + 3Sb(i)}{Sw(i) + Se(i) + Sb(i)}, \tag{1}$$

where

— $Sw(i)$, $Se(i)$, and $Sb(i)$ denote the number of students that assign the *worse*, *equally*, and *better* grade for student i, respectively, and
— $1, 2, 3$ are numerical encodings of grades *worse*, *equally*, and *better*, respectively.

We exclude the *N/A* answers. The similar formula, denoted as $\bar{G}^T(i)$, is used for computing average teachers' grades.

The scatterplot of the obtained results is shown in Figure 8. The location of each point represents an average student grade assigned by teachers (y-axis) and students (x-axis). Since the points on the scatterplot forms an ellipse-like shape, there is a weak correlation between our two variables. Indeed, the calculated value of the Spearman's rank order correlation is equal to $\rho = 0.15$ and the respective statistical significance value is equal to $p = 0.094$. Thus, we can only state that there is a really weak positive dependency between grades assigned by students and teachers. Additionally, since the calculated statistical significance value (denoted as p) is greater than 5% (it is equal to 9.4% in our case), the association between the two variables would not be considered statistically significant by normal standards. However, some interesting observations can be made by analyzing Figure 8.

We observe from Figure 8 that when teachers assign grade *equally*, the students assignments vary all the way from the *worse* to *better* grades. Moreover, only a few grades *equally* were assigned by students. This means that students tend to differentiate more their peers from the performance of a group as a whole than teachers do.

To see to what degree students are more positive or negative than teachers regarding their peers, we conducted the pairwise comparison of respective average grades. In particularly, we considered the difference between students' and teachers' gradings calculated as $I(i) = \bar{G}^S(i) - \bar{G}^T(i)$, so that $I(i) > 0$ means that students give a higher grade than teachers. This comparison shows that the grade assigned by students in 72% of cases is higher than the grade given by teachers, and in 21% of cases it is lower.

As a consequence, students definitely have a different view of student performance compared to the teachers. It is still not possible to give a certain answer to the question whether the students grading is fair enough for summative assessment. However, we argue that peer assessment results can serve as *input* when searching for over- and under-performing students. Peer assessments by students can be used as a tool in the assessment process. Further investigations (for instance by using reports and results of interviews) can then help the teachers to make final decisions of student grades.

6.3. Discussion in Relation to Related Work

Peer assessment has several strengths besides the potential use as input for teacher-based summative assessment, as discussed in previous section. Some of these strengths are:

— Identification of "free-riders". The ability to lower the grades for group members not contributing enough to the group is a main concern of many students [Farrell et al. 2012]. "Free-riders" normally receive relative low grades by their peers [LeJeune 2006; Clark et al. 2005].
— Accountability of the students. If the students know in beforehand that their individual contributions will be evaluated by the team members, they are more careful about how they spend the time in the project work.
— Focus on learning outcome. To provide a reliable assessment, students need a thorough understanding of the goals and evaluation criteria. Peer and self assessment contribute to the positive awareness of the learning outcome amongst students, especially the ability to reflect and give and receive feedback [Willey and Gardner 2010].
— More informed assessment. A common argument for peer-review is that teachers cannot always be present in the team-work and students have a much richer source of information. Several reports point out both that students have confidence in peer assessment [Farrell et al. 2012], and that the accuracy is acceptable [Herbert 2007; LeJeune 2006].

There are also inherent issues of peer assessment that need attention:

— Wrong focus in the course. There is a risk that students put more effort in pleasing the expectation of their peers instead of focusing on the customer requirements [Slivovsky et al. 2003].

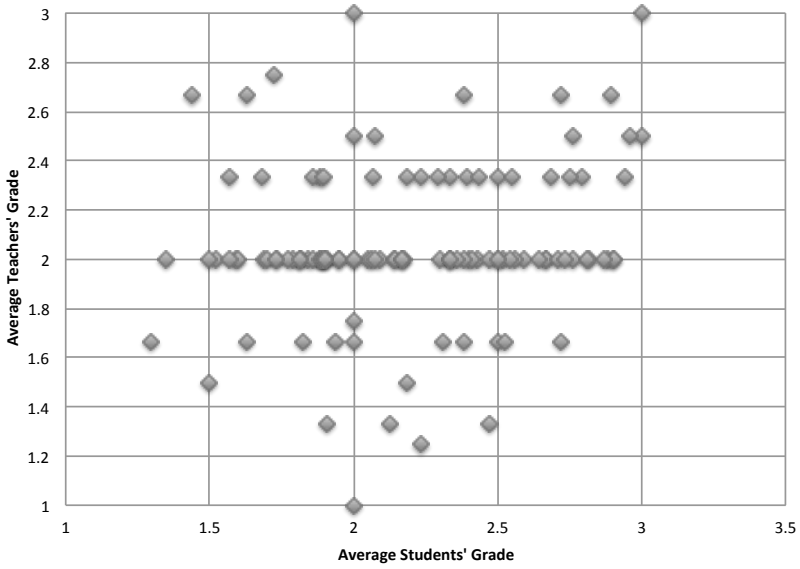

Fig. 8. Scatterplot for average grades assigned by students and teachers. Note: $1, 2, 3$ are numerical encodings of grades *worse*, *equally*, *better* respectively.

— Accountability of the teacher. In many institutions the final grade must be set by an accountable examiner. The students' competence in judging each other can always be questioned.
— Large overhead. Involving students in assessment takes time from the project work and makes the students less motivated in making a good assessment work.
— Under-rating oneself. It has been observed that good students have a tendency to under-estimate themselves in self-assessment, especially when it comes to giving the highest grade [Lejk and Wyvill 2001; Herbert 2007].
— Friendship. It is fully possible that groups of students form pacts where they rate themselves evenly and on a high level [Herbert 2007].
— Saboteurs. For many reasons students can misuse the peer assessment to try getting a better grade themselves. For instance, by intentionally over-assessing themselves and under-assessing their peers. Such patterns can be detected by analyzing the relation between self- and peer assessment [Lejk and Wyvill 2001].

LeJune [LeJeune 2006] compared student and teacher assessments and makes the following observation about student assessment: "The technique described here agrees remarkably well with the instructor observations of individual students performance." (p. 235). This could not be supported by our data, and there might be different reasons for this. First of all, the groups used in LeJune's study were much smaller, about 4 students, which means that they all are quite well familiar with each other, whereas our students cannot have a well-informed opinion about all 30 students in the group. We would probably need a more fine-grained measurement of the confidence in the assessment as in EPICS [Slivovsky et al. 2003]. Using just assess/not-applicable is not trustworthy; even though 20% of the assessments were non-applicable the average student assessed 24 members which is still a large group. Another reason might be that our teachers are conservative in deviating from the group grade. This might be due to the fact that the group grades were set first and thus the teachers were inclined to justify their decisions.

Still bearing in mind that the result is insignificant, we can observe from the scatterplot in Figure 8 that when both teachers and students suggest a deviation from the average they agree quite

often about students performing better than the average. One explanation for this might be that the students know what a good work is when they see it, but since the students have limited experience it is harder to determine what is wrong when some parts of a peer's accomplishment are insufficient. The competence of students to assess peers has always been questioned [Herbert 2007], but there is a potential to help them by asking them to assess a few well-defined aspects of the performance of their fellow students. This would be a more straight-forward task than assessing the overall performance in a single assessment. By taking more precise measurement, there is also a possibility that the noise in the data will be reduced and give a richer analysis.

Yet another strategy for noise reduction is to filter the raw data. SPARK^{PLUS} contains methods for detecting anomalies in the peer assessment, such as, students forming pacts to help each other and saboteurs that try to get a good grade themselves by under-rating others.

In spite of the insignificant result we now have a base-line for peer-assessment against which we can compare the result from future studies.

7. LESSONS LEARNED

In this section, we discuss some of the lessons we learned during the seven-year period used for developing this course. We divide the discussion into two topics: (1) the formative assessment aspect: how can we give continuous feedback that enables active learning? (2) the summative assessment aspect: how can we asses students' work and achieve fair grading?

7.1. Formative Assessment—Continuous Feedback for Active Learning

From Figure 1, we extract the assessment activities that are mostly formative: *oral feedback at student meetings*, *on-demand artifact feedback*, *retrospective*, *student coaching*, and *grading criteria*. For these assessment activities, we discuss the main lessons learned.

7.1.1. Oral feedback at student meetings. In general, oral feedback is a very good way to interact with students continuously during a course. Students tend to appreciate that senior teachers are present at student meetings, both for answering questions that appears during meetings, but also to give feedback and advice about the project. However, attending all student meetings are neither practically feasible, nor recommended. Teaching resources are limited, and for practical reasons, teachers need to prioritize the meetings they are attending.

We have found it very useful when senior teachers attend weekly status meetings, where all students in a group participate. By at least observing and giving feedback at these meetings, teachers can help the students to keep the focus and goal of the project clear. However, too much teaching involvement is also dangerous from a learning perspective. Teachers are giving good feedback by asking "the right questions", rather then providing answers and solutions. Typically, we are letting the students make minor mistakes during the course, without immediately interacting and changing the direction of the project. By doing so, students learn by their mistakes, something mentioned when students reflect on their learning during the course. It is, however, very important that the teacher is present at the student meetings and eventually helps the students to get back on track in the right direction.

Feedback and learning require trust between students and teachers. This trust relationship becomes especially challenging when the same teachers are both giving feedback and grading the students' activities. From a learning perspective, it is important that the teacher is more seen as a coach, a mentor, or a supervisor, than a teacher that is grading the student's performance. We have also learned that it is a good idea to use different teachers for giving feedback before the final presentation (at dry runs) and for assessing the final grade of the presentation.

Although feedback at a student meeting is mostly a formative assessment activity, it is also good for summative assessment, especially if the teachers try to hand out individual grades. We have found it helpful that different teachers sometimes visit the same group and that the teachers keep a written log that describes individual student performances. Such log book does not capture all

student performances, but can point out the extreme cases: both students that are excellent and students that are trying to fly under the radar.

7.1.2. On-demand artifact feedback. Giving feedback on artifacts during the course is essential for active learning. The on-demand aspect implies that feedback is provided by teachers when needed. Feedback on artifacts during the course is optional, initiated by the students. However, we have observed that a bit of encouragement is often required. We recommend that feedback sessions still are optional, but that some of the meetings are scheduled in advance.

Another recommendation is to assign specific artifacts to certain roles in the project group. By making this relation between artifacts and roles clear, it is easier to keep track of individual students' performance. It is, however, important to be quite flexible, not forcing to much structure on the students. This is especially important if student projects also include a transition into Agile development methodologies, as done in the company approach [Broman et al. 2012].

Although on-demand artifact is essentially a formative assessment activity, it still provides important information about individual student performances. As a consequence, we recommend that teachers provide as much feedback as possible on artifacts, including feedback on artifacts that are not explicitly mentioned in the grading criteria.

7.1.3. Retrospective. Retrospective meetings are pure formative assessment activities. The goal is that students run these meetings, including leading the discussions, take notes, and create action points. Although this is essentially a student activity, we recommend that a teacher is organizing the first retrospective meeting, to make the students aware of the main structure of such meeting. It is very important, however, that the teacher does not interfere in the actual discussion; he/she should only act as moderator or discussion leader, not as a teacher that is giving feedback.

We also suggest that the students themselves elect the discussion leader for the second and following retrospective meetings. Typically, we have suggested that the students should nominate and vote for a student in the group that they trust. Because this assessment activity should be purely formative, we also recommend that the teachers do not participate at all in the retrospective meetings that follows the first meeting.

7.1.4. Student coaching. Student coaching is a purely formative assessment activity where students help and coach each other. It is essential to empower the students to coach as well as encourage activities such as student teaching sessions, pair programming, and other learning activities. We strongly encourage student coaching in general, but should at the same time stress that these activities are hard for teachers to follow and assess.

To make such empowerment successful, it is vital that students get the roles that fit their interests and competences. One technique that we have found very useful for role selection is to start with a lecture where the teacher briefly presents available roles in the project (project manager, testers, developers etc.). Students then select the roles that they are mostly interested in, followed by a public voting among the students of the roles where there are more students than vacant roles. Voting ought to be closed and that only the winners are disclosed; the result and absent of votes for individual students can be emotionally very sensitive. Note also that students may change roles later during the course and that the actual roles can be less formally defined if agile methodologies are applied.

7.1.5. Grading criteria. Grading criteria are formally a list of requirements for achieving certain grades; it is not explicitly an activity. Nevertheless, we have found that such a list greatly influences the way students are working and prioritizing tasks. As a consequence, both the process of how students are using the list to plan their work and how teachers are using the list for assessing student accomplishments, can be seen as assessment activities.

We recommend that the list of grading criteria is divided into different levels, each level stating the minimal requirements for achieving a specific grade. We have applied the grading criteria at the group level and not at individual level. This means that students study the criteria together as a group and decide on the ambition of the project and parts that should be prioritized.

It is vital that each criterion is measurable, meaning that the teachers can, at the end of the course, clearly judge if a specific criterion was fulfilled or not. Moreover, we have realized the importance of encouraging the students to follow up and prioritize their work in relation to the grading criteria from the very beginning, so that teachers' feedback on, for instance, artifacts early on in the course becomes formative. We have also found that that grading criteria is a good way to communicate practices. Students invest time and try early in the course to understand what things are expected from them. This effort often generates relevant questions and may result in a good interaction between students and teachers.

Finally, we learned that it is important to divide the grading into two parts: process grades and result grades, describing how the student work to accomplish a specific goal and what the resulting product and artifacts are, respectively. We would also like to stress that the actual product domain (what the students produce) is of great importance; to be motivated and committed to the project, it is essential that students are interested in the product and the results.

7.2. Summative Assessment—Achieving Fair Grading

Next, we discuss lessons learned for achieving fair grading. From Figure 1, we select the assessment activities that are mostly summative: *individual student reports*, *individual teacher assessment*, *individual student interview*, and *final feedback report*. We do not provide any discussions of lessons learned for *peer assessment* because this activity has only been studied as an experiment, not as part of a real course.

7.2.1. Individual student report. Although we believe that the student report may be a good tool for self assessment and formative assessment, the survey shows that many students disagree. However, it has turned out that the individual student report is a very good summative assessment activity, in particular for individual grading.

When student explains what they have accomplished in the course, it is beneficial if they report this as a detailed time sheet. Such time sheet details when an activity occurred, what has been accomplished, and together with whom. When the teacher later reads all student reports, a fairly clear picture is shown on who collaborate with who, and if someone is trying to "fly under the radar" without accomplishing much in the course. In particular, we have found that the individual report is a good tool for giving input to the decision of which students that should be interviewed. In general, students tend to be very honest about their time report and what they have accomplished.

In our experience, it is, however, hard to have full control over the individual reports, especially if the course is large with many students. Since the students produce a lot of information about what has been done in the course, it is hard for teachers to sort out the relevant information in a reasonable time frame. As a consequence, the individual student report should be seen as an advice to request more information (for instance by student interviews), and not as a final summative assessment activity in itself.

7.2.2. Individual teacher assessment. Teachers observe individually during the course what different students accomplish. At the end of the course, teachers propose individually by themselves what grade different groups and individuals should have. By first giving an individual judgement, before discussing student performance with other teachers, we believe that fairer grading can be achieved. It is, of course, hard to measure how "fair" a grade is, but is important to not be biased by other teachers judgements too early in the process.

We have also found it particularly important to observe silent students, who might not say much at large meetings, but do significant useful work without speaking up and explaining it. Hence, it is very important that the individual student reports are analyzed when making the individual teacher assessments.

7.2.3. Individual student interviews. The individual student interview activity is purely summative, with the purpose of gaining more information about the individual students' performances. We

identify two main challenges of making this activity fair for grading purposes: (1) which students should be interviewed?, and (2) which questions should be asked during the interview?

We learned that this activity is heavily dependent on the success of the other assessment activities. In particular, both the individual student report and discussions at student meetings are important activities that can be used when selecting students for interview. Moreover, we also recommend that the individual teacher assessment activity is performed prior to the student interviews, so that teachers first individually create lists of students for interviewing, followed by a discussion between all teachers.

Another important thing is to be fairly prepared for the interview, meaning that the teachers know which the main questions/topics for the interview should be. For instance, if a student is interviewed because the teachers suspect that he/she has not contributed enough in the software development, questions about check-ins to the code repository may be prepared.

7.2.4. Final feedback report. At the end of the course, an activity for the teachers is to create a final feedback report, typically including grading. Most of the other summative assessment activities are input to this report, but this activity can also include further assessments of the student. In particular, we typically request the students to submit a bundle (zip-file) of all the artifacts that the students have produced during the course (including design documents, meeting protocols, code etc.). This bundle is then used when performing the final assessment of code, documents etc. Students may also submit a document describing how they have met the different grading criteria, including a statement saying what grade they are aiming for.

Another advice is that all involved teachers book several dedicated days for this activity. A good approach is to start this meeting by letting all teachers comment and discuss the performance of each student. Students that stand out in some way, either positively or negatively, are further discussed.

Another observation is that some roles in a project are easier to assess than others. For instance, developers tend to be fairly easy to assess, as long as teachers can inspect the code repository and study the content of individual students' commits. Also, it tends to be easier to assess students that have clear roles in the project. If students are switching roles during the course, individual judgements becomes significantly harder.

Making fair individual grading is in general hard. Typically, for a student to achieve considerably higher grades than the rest of the students in a project, they need to show both that their individual work is of high quality and that their work have large impact on the overall project result. Although giving significant work load for the teachers, providing short individual written feedback to the students seem to be highly appreciated.

8. CONCLUSIONS

In this paper, we propose an assessment model for assessing large software engineering project courses. The model consists of ten assessment activities that are evaluated by a questionnaire-based survey. We contend that this model can be useful when analyzing assessment activities in various software engineering courses. Moreover, we study two important aspects of peer assessment when applied in large projects. This analysis is based on a peer assessment experiment, performed at the end of a software engineering project course. Finally, we present and discuss lessons learned, which are the result of seven years experience of teaching and improving a course based on the proposed assessment model.

Acknowledgement

The authors would like to thank all the students who participated in the software engineering course.

REFERENCES

Adelina Basholli, Fesal Baxhaku, Dimitris Dranidis, and Thanos Hatziapostolou. 2013. Fair Assessment in Software Engineering Capstone Projects. In *Proceedings of the 6th Balkan Conference in Informatics (BCI '13)*. ACM, New York, NY, USA, 244–250. DOI: http://dx.doi.org/10.1145/2490257.2490268

John Biggs. 1996. Enhancing Teaching through Constructive Alignment. *Higher Education* (1996). DOI: http://dx.doi.org/10.1007/BF00138871

David Broman. 2010. Should Software Engineering Projects be the Backbone or the Tail of Computing Curricula?. In *23th IEEE Conference on Software Engineering Education and Training*. Pittsburgh, USA. DOI: http://dx.doi.org/10.1109/CSEET.2010.35

David Broman, Kristian Sandahl, and Mohamed Abu Baker. 2012. The Company Approach to Software Engineering Project Courses. *IEEE Transactions on Education* 55, 4 (2012), 445–452. DOI: http://dx.doi.org/10.1109/TE.2012.2187208

Nicole Clark, Pamela Davies, and Rebecca Skeers. 2005. Self and Peer Assessment in Software Engineering Projects. In *7th Australasian conference on Computing education - Volume 42 (ACE '05)*. Australian Computer Society, Inc., Darlinghurst, Australia, Australia, 91–100. http://dl.acm.org/citation.cfm?id=1082424.1082436

Vivienne Farrell, Graham Farrell, Paul Kindler, Gilbert Ravalli, and David Hall. 2013. Capstone Project Online Assessment Tool Without the Paper Work. In *18th ACM Conference on Innovation and Technology in Computer Science Education (ITiCSE '13)*. ACM, New York, NY, USA, 201–206. DOI: http://dx.doi.org/10.1145/2462476.2462484

Vivienne Farrell, Gilbert Ravalli, Graham Farrell, Paul Kindler, and David Hall. 2012. Capstone Project: Fair, Just and Accountable Assessment. In *17th ACM annual conference on Innovation and technology in computer science education (ITiCSE '12)*. ACM, New York, NY, USA, 6. DOI: http://dx.doi.org/10.1145/2325296.2325339

Sally Fincher, Marian Petre, and Martyn Clark. 2001. *Computer Science Project Work: Principles and Pragmatics*. Springer. DOI: http://dx.doi.org/10.1007/978-1-4471-3700-9

Nicole Herbert. 2007. Quantitative Peer Assessment: Can Students Be Objective?. In *9th Australasian conference on Computing Education (ACE '07)*. Australian Computer Society, Inc., Darlinghurst, Australia, Australia, 63–71.

Rosario Hernández. 2012. Does Continuous Assessment in Higher Education Support Student Learning? *Higher Education* 64, 4 (2012), 489–502. DOI: http://dx.doi.org/10.1007/s10734-012-9506-7

Peter T. Knight. 2002. Summative Assessment in Higher Education: Practices in Disarray. *Studies in Higher Education* 27, 3 (2002), 275–286.

Noel LeJeune. 2006. Assessment of Individuals on CS Group Projects. *J. Comput. Sci. Coll.* 22, 1 (Oct. 2006), 231–237.

Mark Lejk and Michael Wyvill. 2001. The Effect of the Inclusion of Selfassessment with Peer Assessment of Contributions to a Group Project: A Quantitative Study of Secret and Agreed Assessments. *Assessment & Evaluation in Higher Education* 26, 6 (2001), 551–561. DOI: http://dx.doi.org/10.1080/02602930120093887

Mark Lejk, Michael Wyvill, and Steve Farrow. 1996. A Survey of Methods of Deriving Individual Grades from Group Assessments. *Assessment & Evaluation in Higher Education* 21, 3 (1996), 267–280.

Fatma Meawad. 2011. The Virtual Agile Enterprise: Making the Most of a Software Engineering Course. In *24th IEEE Conference on Software Engineering Education and Training*. 324–332. DOI: http://dx.doi.org/10.1109/CSEET.2011.5876103

Iwan H. Purtro, Angela Carboneand, and Judy Sheard. 2014. Developing A Framework to Assess Students' Contributions during Wiki Construction. In *Sixteenth Australasian Computing Education Conference (ACE2014) (CRPIT)*, J. Whalley and D. D'Souza (Eds.), Vol. 148. ACS, Auckland, New Zealand, 123–131. http://crpit.com/confpapers/CRPITV148Putro.pdf

Per Runeson and Martin Höst. 2009. Guidelines for Conducting and Reporting Case Study Research in Software Engineering. *Empirical Software Engineering* 14, 2 (2009), 131–164. DOI: http://dx.doi.org/10.1007/s10664-008-9102-8

Ken Schwaber and Mike Beedle. 2001. *Agile Software Development with Scrum*. Prentice Hall.

Lynne Slivovsky, Leah Jamieson, and William Oakes. 2003. Evaluating Multidisciplinary Design Teams. In *American Society for Engineering Education Annual Conference & Exposition*.

Lois J Smith. 2008. Grading Written Projects: What Approaches Do Students Find Most Helpful? *The Journal of Education for Business* 83, 6 (2008), 325–330. DOI: http://dx.doi.org/10.3200/JOEB.83.6.325-330

Robert H. Todd, Spencer P. Magleby, Carl D. Sorensen, Bret R. Swan, and David K. Anthony. 1995. A Survey of Capstone Engineering Courses in North America. *Journal of Engineering Education* 84 (1995), 165–174. DOI: http://dx.doi.org/10.1002/j.2168-9830.1995.tb00163.x

David A. Umphress, Dean Hendrix, and James H. Cross. 2002. Software Process in the Classroom: the Capstone Project Experience. 19, 5 (2002), 78–85. DOI: http://dx.doi.org/10.1109/MS.2002.1032858

Maria Vasilevskaya, David Broman, and Kristian Sandahl. 2014. An Assessment Model for Large Project Courses. In *45th ACM Technical Symposium on Computer Science Education (SIGCSE '14)*. DOI: http://dx.doi.org/10.1145/2538862.2538947

G. Wikstrand and J. Börstler. 2006. Success Factors for Team Project Courses. In *19th Conference on Software Engineering Education and Training*. IEEE, 95–102. DOI: http://dx.doi.org/10.1109/CSEET.2006.34

Dawn E. Wilkins and Pamela B. Lawhead. 2000. Evaluating Individuals in Team Projects. In *31st SIGCSE technical symposium on Computer science education (SIGCSE)*. ACM, New York, NY, USA, 172–175. DOI: http://dx.doi.org/10.1145/330908.331849

Keith Willey and Anne Gardner. 2009. Improving Self- and Peer Assessment Processes with Technology. *Campus-Wide Information Systems* 26, 5 (2009), 379–399.

Keith Willey and Anne Gardner. 2010. Investigating the Capacity of Self and Peer Assessment Activities to Engage Students and Promote Learning. *European Journal of Engineering Education* 35, 4 (2010), 429–443. DOI: http://dx.doi.org/10.1080/03043797.2010.490577

Mantz Yorke. 2003. Formative Assessment in Higher Education: Moves towards Theory and the Enhancement of Pedagogic Practice. *Higher education* 45, 4 (2003), 477–501. DOI: http://dx.doi.org/10.1023/A:1023967026413

A. QUESTIONS USED IN TWO SURVEYS

The information in brackets at the end of each question shows the years when a question was included into the surveys (2012, 2013, or 2012-2013) and whether the results of a question were discussed in this paper. 7 questions that are related to the students' background (year of the course, age, sex, industrial experience, software development experience, curriculum, and result from theory exam) are not included in this list.

(1) We are the right number of people in this company (approximately 30) to learn practical aspects of software engineering. (2012)
(2) We are the right number of people in this company (approximately 30) to create a good product. (2012)
(3) I believe that our company has a good structure for producing quality software. (2012)
(4) I believe that the company approach in this course (i.e., that we are organized as a simulated company) makes our experience industry relevant. (2012)
(5) The teachers involvement in the retrospective meetings for process improvement made me hesitate to express my opinion. (2012-2013, B1)
(6) I think that I would have learned more if the customer was played by a course teacher, because the teacher understands better the course process than an external customer. (2012)
(7) An external real customer can make fair and correct assessment of the product. (2012)
(8) An external real customer can make fair and correct assessment of the customer communication process. (2012)
(9) I got an impression that the external real customer made the course chaotic in the beginning of the course. (2012)
(10) The presence of an external real customer makes me motivated to do a good job. (2012)
(11) It is easy to get clear answers from the customer when demonstrating prototypes. (2012)
(12) Numerical grading in this course (i.e., U/3/4/5 or F/C/B/A) makes me work harder compared to if the course had just two grades (i.e., passed/failed). (2012)
(13) The grading criteria influence our ways of working. (2012-2013)
(14) The individual project grade should only be based on what each student individually perform in the project. (2012-2013, A2)
(15) The individual project grade should be based on a mixture of individual performance and the performance of the company as a whole. (2012-2013, A1)
(16) The presence of an individual grade makes me work harder than if the grade was only based on the whole company's performance. (2012)
(17) I noticed that other members were not willing to help me due to the personal grading. (2012 - 2013, A3)
(18) I feel I have a fair chance to improve my own individual grade in the course. (2012)
(19) Interviewing individual students at the end of the course is a good way for teachers to give fair grades. (2012-2013, B2)
(20) Getting a good grade is the most important factor for me to make a good job in the project. (2012)
(21) Doing a good job and learning a lot is more important than grades. (2012-2013, C1)
(22) The result of the written exam in October 2012 made me more motivated for the project work. (2012)
(23) The feedback on the artifacts helped me to improve my work. (2012-2013)
(24) The competition between companies makes me motivated to do a good job. (2012)
(25) The competition between companies made us focus more on creating a good product than keeping good internal software engineering practices (e.g., creating testing procedures, writing architecture/design documents, and managing requirements). (2012)
(26) Honestly, I think that my reported hours correspond well to the actual number of hours worked in the project. (2012)
(27) I am learning a lot by writing the individual reflection reports. (2012-2013, C2)

(28) The individual report helped me to reflect upon my role in the whole company. (2012-2013, C3)
(29) The individual report helped me to understand some issues in my company and how I could contribute to the company to solve these issues. (2012-2013, C4)
(30) I did not consider the individual report as a reflection document because it was used by the teachers for grading. (2012-2013, C5)
(31) I think that I would have learned more by writing a common company experience report than writing individual reports. (2012)
(32) I believe that the teachers/supervisors have a good understanding of the situation of our company. (2012-2013, B3)
(33) The feedback from the teachers at the company meetings helped me to improve my work. (2013)
(34) I believe the retrospectives helped us to improve our performance. (2013)
(35) The grading criteria helped me to organize my work. (2013)
(36) To get fair grading, it's not enough to interview a few students. I believe all students should be interviewed. (2013, B4)
(37) I received sufficient feedback on the artifacts that I contributed to. (2013)
(38) The lecture on writing reflection reports was helpful when writing the second individual report. (2013, C6)
(39) I was coaching other students in my company, e.g. by giving tutorials, conducting hands-on sessions, and reviewing produced artifacts. (2013)
(40) I was coached by other students in my company. (2013)
(41) I believe that students need more assistance from the teachers during the course. (2013, B5)
(42) I think that coaching from other students helped me to improve my work. (2013)