GitHub for Education

Alexey Zagalsky
University of Victoria
Victoria, BC, Canada
alexeyza@uvic.ca

Yiyun Zhao
University of Victoria
Victoria, BC, Canada
evianz@uvic.ca

Weiliang Wang
University of Victoria
Victoria, BC, Canada
weiliang@uvic.ca

Margaret-Anne Storey
University of Victoria
Victoria, BC, Canada
mstorey@uvic.ca

ABSTRACT

Software developers use GitHub to host and collaborate on software projects, however, GitHub provides not only a sharing space for the software projects, but a social meeting place for members involved or interested in the project; In-fact GitHub supports communities of interest and communities of practice. Additionally, GitHub is not limited to supporting collaboration for software projects alone, and it is being used in other areas as well, such as technical writing.

In this study we investigate *whether* and *how* GitHub is being used to support learning and teaching. We conduct a qualitative study, focusing on how is GitHub being used, what is the motivation, benefits and challenges of using GitHub for educational purposes. We found diversity in the usage of GitHub in the classroom, ranging from *"just another submission system"*, to a catalyst for improved collaboration and participation among students and educators. Furthermore, our findings indicate that GitHub can be used not only for software project-based courses, but for non-technical fields such as humanities as well. However, lack of support for widely used file types and formats, and Git's complex nature provide a challenge for adoption in the context of education.

We conclude by discussing the viability of GitHub to support learning and teaching, the limitations of our study, suggested practices for practitioners, and recommendations for GitHub itself. Finally, we provide an overview of future work.

Author Keywords

CSCW, Learning, Education, GitHub, Social Media, Qualitative Methodology

ACM Classification Keywords

K.3.1 Collaborative learning: Computer Uses in Education

CSCW'15, March 14–18, 2015, Vancouver, BC, Canada.

INTRODUCTION

GitHub is a popular web-based hosting service for software development that utilizes the Git revision control system. It has shown to be useful in several areas that require collaboration, especially in software development and technical fields (e.g. technical writing). We propose the use of GitHub for Education, and we're not the only ones to do so.

In 2011, Greg Wilson suggested[1] that GitHub could be used for learning materials -

> *Would it be possible to create a "GitHub for education"? Right now, I think the answer is "no", because today's learning content formats make* ***merging hard****. Whatever a "GitHub for education" would look like, it would not be yet another repository of open learning materials. There are lots of those already, but almost all their content is write-once-and-upload, ... rather than* ***sharing course content in a reusable, remixable way****.*

In 2012, he elaborated[2] that

> *"GitHub for Education" isn't necessarily, "Let's put educational materials in GitHub", but rather, "Let's* ***facilitate a culture of spontaneous-but-structured collaboration*** *and improvement".*

In 2014, GitHub initiated a campaign[3] with the purpose of promoting GitHub to higher education, and while it barely scratches the surface, it is a welcomed first step. GitHub's campaign has given stage to a discussion on the matter, and Jim Baker, a senior developer and a lecturer in Computer Science at the University of Colorado, has commented on Facebook (Fig. 1)

> *"We had a great experience using GitHub to support a collaborative workflow for the 70+ students in each of the two semesters of my CS course".*

When asked for more details he replied

[1] `http://software-carpentry.org/blog/2011/12/fork-merge-and-share.html`
[2] `http://software-carpentry.org/blog/2012/04/github-for-education.html`
[3] `https://education.github.com/`

Figure 1. GitHub education discussion on facebook.

> *"Pull requests (PR) are the heart of the GitHub workflow, and we took advantage of PRs, including task lists so that students could report on their work in progress and get over initial humps. Any merged PR got extra credit(!). Because the course had been improved in some way - this seemed like an interesting standard for giving out extra credit. Consequently, we mostly didn't merge PRs for labs, except for bug fixes, but we were always on the lookout for* ***better solutions than ours****. PRs were also merged for extra credit such as* ***corrections of my course notes****. Next fall we expect to have* ***autograding*** *implemented as a form of continuous integration, by running against the PRs through postcommit hooks".*

This research aims to examine whether and how GitHub can be used for educational purposes, while relying on its unique collaboration, social and awareness features. We aim to answer the following research questions:

- Is GitHub used in higher education courses to support learning?
- How is GitHub used to support learning?
- What is the motivation and benefits of using GitHub for education?
- What challenges are related to the usage of GitHub for education?

The contribution of this research is fourfold: (1) present up-to-date overview of related work, (2) investigate current use of GitHub *in the wild* (3) present challenges and problems related to using GitHub for education, and (4) provide recommendation and suggested practices on how GitHub could be used to support learning.

RELATED WORK

Before discussing GitHub and how its unique features might support education, we present a brief overview on technology in education, and the strong connection between Computer Supported Cooperative Work (CSCW) and education.

Technology in Education

Technology use in education began with educational films in the 1900s, and with Sidney Pressey's mechanical teaching machines [19] in the early 1920s. In the 1970s, learning material was augmented with graphics and multimedia, and in 1980s and 1990s computer based learning included the use of various technologies such as micro-worlds [17], simulations [23], and hypertext.

The early form of online learning (mid 1990s) was supported by Internet, eMail and Forums, and focused on interaction between the student and the computer. Modern form of online learning focuses on computer-mediated communications and the formation of a communities of practice [13]. It uses various technologies such as cloud computing [25], social media (e.g. Twitter [6, 20]), mobile devices [4], and MOOCs [18].

Technology had a major influence on education, and it will continue to do so (e.g. Khan Academy [26]). However, it is not trivial if and how various technologies should be used to improve learning.

Participants of the "2020 Panel", as described in the book "Technology in Education - Looking Toward 2020" [15], tried to envision the future of education and enrich people's understanding of technology's potential role in education, as well as opportunities, drawbacks and challenges. William Bossert, one of the panelists, proposes that technology should be used to encourage intellectual exchanges among students in the classroom. *Appropriately designed computer programs would make possible the solution of complex problems by a group of students, while helping "level the abilities" within a group so that less able students might take on tasks that would be viewed by others as meaningfully furthering the group activity*. Another panelist, Jim Minstrell, envisions the classroom of the future to be *full with electronic aids for teachers, helping with organizational and management tasks such as: keeping track of appointments and deadlines*. Our study suggest that the use of GitHub unique collaboration, social and awareness features might be the part of that vision.

CSCW and Education

Back in 1984, a workshop [8] organized by Paul Cashman and Irene Grief discussed the way people work and how technology could support it. The term "Computer Supported Cooperative Work" (CSCW) was first coined, while nowadays some researchers may also use the term "Computer Supported Collaborative Work". As Paul Wilson defined in his book [30] in 1991, CSCW [is] a generic term, which combines the understanding of the way people work in groups with the enabling technologies of computer networking, and associated

hardware, software, services and techniques. Since then, extensive research and experimentation has been conducted to explore CSCW, for instance the investigation of awareness by Dourish & Bellotti [2], issues in the design and evaluation [7, 14], or evaluation of collaborative applications or systems [10, 21], and more.

CSCW impact extends to education as well [1], by providing mechanisms to support learning. With educational applications and systems that are designed and evaluated, we are able to see how CSCW contributes and supports education in various scenarios, such as lectures, writing assignments or lab work [1]. Moreover, findings of three research programs conducted at the University of Strathclyde show that common guidelines can be developed in the form of procedural stages to facilitate the successful implementation and usage of CSCW technology cross-sectorally in education [12].

More specifically, as opposed to other domains where distance still matters [16], the distance in education is no longer an issue with the right tools and correct environment. Internet offers various possibilities for asynchronous as well as synchronous collaboration and plays a major role in distance education [22]. Remote education systems that are based on CSCW principles provide the learner various learning methods with effective learning experiences and at a low cost [29].

Furthermore, not only students can benefit from the application of CSCW in the classroom, but also teachers could take advantages of such systems, e.g. automating assignments or monitoring students' online behaviour.

GitHub and Education

GitHub offers several unique features that greatly facilitate user's collaboration. For example: Pull Requests are a way to initiate discussion. The discussion may involve code that is visible to everyone and shows what exact changes would be merged if it is accepted, or it may invovle other content beside code (e.g. screenshots) to provide a background for discussion. When a user wishes to contribute to someone else's project, he can either *clone*[4] the project, thus creating a full clone of the project in his local environment while changes commited would directly affect the original project. This is called a Shared Repository Model[5], where Pull Requests help start code review and conversation about proposed changes before they're merged into the master branch. Alternatively, the user can *fork*[6] the entire project, thus creating a parallel project where commited changes would not directly affect the original project. This is called a Fork & Pull Model, where Pull Requests will provide a way to notify the original project maintainers about the changes you'd like them to consider.

Users can not only follow other users or projects of interest, but also broadcast their activities to their followers. Furthermore, users can discover new projects by using the *explore* feature, or share snippets by using the *Gist* feature. GitHub also supports awareness by broadcasting updates to the user's news feed. The combination of these features supports and facilitates *"a culture of spontaneous-but-structured collaboration"*[7].

GitHub is not only a prime example of CSCW, but it's also a good data source to study how people collaborate. Wu *et al.* [31] analyzed rich GitHub timeline data and found that most of the *follow* activities on GitHub are not due to developers collaborations on GitHub but their interactions outside GitHub, such as on Twitter, Hacker News, and StackOverflow. Majumder *et al*. [11] conducted an investigation on how teams are formed in a social network based on millions of software repositories spanning a period of four years and hundreds of thousands of developers from GitHub. Tsay *et al*. [27] performed a quantitative analysis of project success of over 5,000 open source software projects hosted on GitHub. They used two measures, **developer attention** and **work contribution**, for projects success, and found that projects with a high level of developer multitasking tend to receive less developer attention, but greater work contribution.

Up until recently, GitHub has mainly been used for software development, but it can also be extended to other domains that involve collaborative works - Education is one of them [5]. In this different context, both GitHub and teaching techniques might require readjustment while some of GitHub features might be repurposed. For instance, student assignments should be modified to fit GitHub's transperancy and collaboration model.

Several educators have introduced the use of GitHub into their classrooms and had a good experience.

In 2010, Luis Felipe Borjas[8] posted about the use of github organizations[9], a way to simplify management of group-owned repositories, to manage the class projects. He also applied GitHub to exams: instead of waiting until the exam deadline to upload the exams, students could just *push*[10] as many times they wished and the last *push* they made before the deadline was going to be considered as their final response. Furthermore, he strongly suggested teachers to create exams/homeworks that would translate into private *git repos* to which the students could push.

In 2011, David Humphrey stated in his blog[11] that he asked his students to use Git/GitHub and highly recommended instructors to use GitHub in classroom. He claimed that although it was a little painful to learn Git/GitHub at the beginning, the payoff would be huge.

In 2013, Griffin & Seals have used GitHub in the classroom as a version control tool while leveraging the *branch* and *merge* features [5]. When students worked on programming assignments, it was easy to *merge* it back into the original project if

[4] https://help.github.com/articles/duplicating-a-repository

[5] http://guides.github.com/overviews/flow/

[6] https://help.github.com/articles/fork-a-repo

[7] http://software-carpentry.org/blog/2012/04/github-for-education.html

[8] http://lfborjas.com/2010/10/30/git-classroom-exams.html

[9] https://github.com/blog/674-introducing-organizations

[10] https://help.github.com/articles/pushing-to-a-remote

[11] http://vocamus.net/dave/?p=1358

their version worked, also pretty handy for them to abandon that branch if it's totally off course, without destroying the original project.

Apart from these testimonies of using *"raw GitHub"* in the classroom, there are also education targeted platforms based on the GitHub model. Coursefork[12], created in March 2013, is described as *"GitHub for course creation"*[13]. It is a platform for open-sourcing and collaborating on educational material, where educators can upload course material, allow others to create copies, modify or share them with others.

These examples show that although this idea has not been promoted widely, it is being practiced *informally* and has much potential to benefit education.

METHODOLOGY

Based on our research questions, we chose to employ Qualitative methodology [3, 24], consisting of three phases of data collection (online methods [28], interviews, validation survey), and iterative phases of data analysis. In this section we elaborate on our research questions, study design, study participants, data collection and data analysis.

Research Questions

This study strives to answer the following research questions:

Is GitHub used in higher education courses to support learning?

This study examines whether GitHub is being used (formally or informally) to support learning in higher education courses.

How is GitHub used to support learning?

In cases where GitHub is used to support learning, we investigate how it is used. Specifically, for what uses it is being used (e.g. file sharing, collaboration, time tracking). Moreover, we investigate what specific features of GitHub are being used to support learning.

What is the motivation and benefits of using GitHub for education?

We investigate the motivation and the possible benefits GitHub usage might bring to support learning, based on testimonies of the study's participants.

What challenges are related to the usage of GitHub for education?

Furthermore, we examine what challenges students and lecturers face when trying to use GitHub, or what problems prevent them from even trying.

Study Design

In the first phase of our study, we used online research methods, and specifically blog posts [28] describing personal experiences of educators with GitHub to support learning or teaching. This phase was exploratory in nature, and through it we found creative and successful examples of GitHub usage to support teaching and learning in the classroom. We also found discussions on the challenges and difficulties of using GitHub. This allowed us to refine our research questions, and guided us in the following phases of the study.

In the second phase, we interviewed 5 participants, beginning with interviewing one of the blog authors from the first phase. As opposed to the previous phase, at this phase we were able to thoroughly investigate the usefulness and potential of GitHub in the aspect of education. Through iterative data analysis of this phase, several themes have emerged indicating the motivation and challenges of using GitHub to support learning.

The set of themes that emerged informed the third phase of our research: a validation survey sent to interviewees from the previous phase, and to educators that have not been contacted thus far.

Participants

Our study targeted higher education lecturers and professors who have used or use GitHub to support learning and teaching. The inclusion criteria included any type of courses and any type of GitHub usage, as our study aims to investigate diverse usages of GitHub and it's usefulness for non technical courses.

Interview Participants

3 of the 5 interviewees have used GitHub for Computer Science or Software Engineering courses, and 2 interviewees used GitHub for Humanities courses. All of the interviewees have used GitHub for at least 2 semesters (max. 5 semesters). The class size ranges between 10 students and 130 students per course, where all the courses were taught as a frontal course (no online courses). Table 1 shows a summary of students' technical background, type of course (CS/Engineering vs. Non CS) and if a GitHub tutorial was provided.

Table 1. Was GitHub tutorial provided at the beginning of the course?

Interviewee ID	Course Type	Technical Bgd.	Tutorial
1	CS	Yes	No
2	CS	Yes	No
3	CS	Yes	Yes
4	Humanities	No	Yes
5	Humanities	No	Yes

Furthermore, one of the interviewees, provided us with anonymous statistical data based on a student survey conducted at the end of a software architecture course. The survey, receiving 26 responses, showed that most students (73%) had never used GitHub or just used GitHub once or twice before that course, and more than half of the respondents (62%) were satisfied with the process of getting used to the GitHub workflow. Majority of the students (81%) liked using GitHub for that course.

Validation Survey Respondents

Will be added in the full paper.

Data Collection

[12] `http://coursefork.org/`

[13] `http://opensource.com/education/13/9/coursefork-education-tool`

We started by conducting an initial literature review and multiple online searches, including through various social media channels. This led us to a few blog posts describing the experience with GitHub, and to a Facebook discussion on the topic.

For the interviews, we used email invitations sent to lecturers and professors that may have used or use GitHub to support learning or teaching. Potential participants were recruited in several ways: by contacting blog authors who shared their experience on using GitHub in the classroom; by posting an invitation on Twitter; and through snowball sampling, where interviewees would suggest other colleagues.

The interviews were conducted face-to-face (when possible), or with Skype. They lasted 18-60 minutes. During each interview, the interviewer was taking notes, while audio recording the interview as well. The interviews were semi-structured interviews that included 18 guiding questions, and the interviewer was allowed to *dig deeper* with additional questions. This supported the exploratory nature of our study, allowing to explore interesting and unexpected insights.

Data Analysis

Interview data analysis follows the qualitative data analysis guidelines [9, 24] and includes the following stages: (1) transcription of the data recorded, (2) organizing the data into easily retrievable sections, (3) familiarization with the data by reading and re-reading the data, making memos and summaries, (4) reading the data and labeling segments, i.e. coding, and (5) identifying themes or emergent concepts, through discussions among the researchers, and engaging in re-coding to develop more well defined categories.

FINDINGS

In following we address our main research questions and present main themes that emerged from the interviews.

RQ1: Is GitHub used in higher education courses to support learning?

At the beginning of our exploratory study, we truly weren't sure if GitHub can be or is being used to support learning and teaching (beyond the obvious usage for code hosting). This study reveals not only that it is being used in the context of education, but it will continue to be used by all the interview participants (5 out of 5) in the future. Moreover, all the interviewees showed great interest in our study and its findings, with the hope of improving or enhancing their experience.

Our study also indicates that GitHub can be used not only for software project-based courses, but for non-technical fields such as humanities as well.

However, our findings are limited as the inclusion criteria asked for educators that have used or currently using GitHub to support learning or teaching. We elaborate on this in the limitation section.

RQ2: How is GitHub used to support learning?

We found diversity in the usage of GitHub among the interviewees. The most basic case described in the interviews was to use it as a submission platform: *"GitHub was not a core part of the class, so it was just a project submission platform"* [P2]. However, even in this case, when asked to reflect, the interviewee reveals that it was also used for *"monitoring the progress of the students"*.

Several of the interviewees have used GitHub to hold additional course materials (e.g. class notes, assignments), and this allowed for the students, with the use of the Pull-Request mechanism, to provide improvement suggestions. If these suggestions, in the form of Pull-Requests, were to improve the existing version, they were accepted and the student would receive bonus points. One of the interviewees [P2] that didn't use GitHub to store class notes or assignments was pleasantly surprised when we shared it with him.

Another usage of GitHub, that we encountered in the interviews, was for sharing teaching material.

"Compared with educational platforms, it allows transfer of skills, ... You can use something from one project in another project... If you want to make a course site, I think GitHub is the fastest way to reuse materials. If I teach a course for a second time, I can create a site in 30 minutes for the new course, which is really quick. I also circulate my syllabus online. I have colleagues in University of New York, USC, Western University, Toronto, that are using versions of it" [P4].

RQ3: What is the motivation and benefits of using GitHub for education?

Motivation: Professional Experience and Goals

"...I thought this was important is that it gives a very nice way of collaborating, and the pull request mechanism where you give feedback is a wonderful way to giving feedback to each other." [P1]

For the interviewees who used GitHub for students to work on their projects, they would like students to have more experience with collaborative work in software development as well as the real industry [P3]. Meanwhile, GitHub provides repositories and it is a good platform to use the Git, which is what the interviewees need. The interviewees could be fully aware of what was going on in students' repositories and observe their behavior better [P2].

Motivation: Proposed By Students

"So actually, some students came to me, and they said - you should watch this video on how GitHub use GitHub to build GitHub" [P1]

It was the proposal from the students that inspired the interviewee to re-organize the course and use GitHub as a part of it. Therefore, the interviewee realized that the features in Github could benefit in education.

Motivation: Shared Space

"...to have a space where it was easy for students to share and build something in repositories or materials in the same space." [P4]

Students could easily share the course notes, references or other material in GitHub for the same class. From the aspect

of teaching, GitHub could be a place where professors could collaborate on teaching activities or pedagogical approaches [P4]. Teaching material and resources could be shared in a open repository for the professors of related courses.

Benefit: Easy Collaboration

"...is essentially for the teams to collaborate with each other, other than having let them email back and forward. That's how Github helps." [P3]

The Github is a suitable platform for collaboration where students could collaborate and communication. They had to explain their actions to each other on the projects they were working on. Therefore, the interviewees agreed that the Github collaboration mechanisms were useful. Besides, students were able to feel the activity in the projects that they were workin on, which leads to their awareness of the collaboration behavior in Github [P1].

Benefit: Software Development

"For many of them, it was the first time they were exposed to this Pull-Request based way of working, which is the modern way of doing software development." [P1]

The interviewees indicated that Github is the practice community for the software development. Students were able to gain more experience in software development as well as the experience of working with unfamiliar tools for those who were new to the Github [P3]. Moreover, it was helpful to use GitHub in the classroom so that students could learn the way of working for real industry world in Github [P1].

Benefit: Version Control

"I think just like the usual benefits of version control, which makes like merge, configs. It's easier to resolve them faster. It's easier to collaborate when have so many codes back and forward. It's easier to go back to history. It's easier to figure out what change cost bugs and so on." [P3]

Most of interviewees thought that version control brought one of the best benefits from Github. In a version control system, the interviewees could see the full history of how a document was built through different versions including all the issues and discussions [P1]. Under the situation that a group of people would like to build a repository, which could be versioned while they could do their work offline and discretely in the same time, one of the interviewees believed that the Github is the best choice with its features like push, pull, commit with comments and etc. On the other hand, it was easy for student to go back to previous version in a version control system [P4].

Benefit: Course Management

"If I have 40 students, like 20 groups that had to submit an exercise, an assignment, 3 times per semester, that would be a mess of documents that I had to organize. Now, on GitHub it was much easier. I didn't had to do anything basically. I just logged in the repositories, and saw the assignments." [P2]

It is easier and more convenient for the interviewees to use Github to manage the students in the class. They could have a centralized place for their students to upload or submit the exercises, assignments or projects [P2]. The interviewees could create repositories for each teams after assigning the students to different groups in the class. The whole process is streamlined and works well. It also saved time for the interviewees to go throught all the Emails from student of their projects and documents [P2, P3]. Other than that, the duplicates of the course website for different semesters were another usage from the instructor's view [P4].

In the aspect of teaching, interviewees also use Github to share materials with other teachers for new approaches to teach or new assignment ideas. Github automates the process rather than Email back and forward each other [P5]. Similarly, *"Github provides an accessible environment in connecting with the course materials [P5]."* The accessibility and easy usage are both important.

Benefit: Team Management

"...because I like firm deadlines, I could always see who went through the deadline, and I could penalize them. It's basically, I mostly used GitHub for the observability of that." [P2]

In Github, the interviewees were aware of the students' behavior, such as who was active or working on certain parts of a project, which team was handy or practical. The history of committing, emerging, commenting or pull requests were always available for interviewees, which was all recorded. It was easy to tell who was working or not. By monitoring how the team work was done, the credits would be given more fair to the students who worked more [P1, P5]. In another side, interviewees could observe the progress of the projects. For instance, the interviewee subscribed to all the students' repositories and followed the feed of them. Therefore, he would see how were the projects going and what approaches the students used for the projects since he didn't tell students to use any features specific [P2]. The interviewees could see everything, even in graph represented that the students were working on during the course semesters,[P4].

Benefit: Privacy settings

All of our interviewees used private repositories in Github while teaching in their class. Some of the interviewees used Github as an open repository for all the attending students, but not to public. Students could be able to see other groups of people's work. On the other hand, some interviewees preferred to keep all repositories private to avoid effect on academic results. Moreover, some interviewees allowed students to configure their repositories as they liked because for instance, team leader could have full access while other members might have to send a pull request to commit something [P3].

Benefit: Enhance learning, Exploring other projects

"...enjoyed some of the other open source projects and looking closer into them." [P1]

Our interviewees realized that Github is used by many more people in real industry, not only in academy. Compared to Moodle, Blackboard and other apps a lot of universities built, Github provides training of professional skills to some extent [P4]. It helps students to get touch with the real world as well as to be easier to get on a career path.

Benefit: Simple usage

"It doesn't have many fancy features, and you don't need them. So the simplicity is sort of elegant." [P1]

One of our interviewee thought Github was easy and straight forward to use. The highlight of features in Github was the Pull-Request mechanism which is obviously one of the strengths [P1].

RQ4: What challenges are related to the usage of GitHub for education?

We categorize the challenges we found into the following aspects: accessibility & usability, and external restrictions.

Accessibility & Usability

GitHub in its core is Git-based, and even though it provides a simple web interface, for tasks that involve collaboration, understanding of Git is required. *"To get the most out of GitHub, you need to understand Git. Even if you just use it to edit documents together, you will get conflicts once in a while. And if you want to use the review mechanism, then you want/should use the Pull-Requests. That requires some pretty deep knowledge of Git still. If you use it the right way it is simple, but somehow with Git you end up with conflicts, and if you don't understand it, it's magic"* [P1].

Improving accessibility for users with limited technical background was further reinforced: *"The problem is around accessibility for people who don't have lots of technical background. One way of doing that could be to support a minor variety of platforms using the client or to have more robust documentation, maybe like screencast documentation that explains very very basic things like setting up an account, getting the client, using the GitHub client...Largely, the biggest challenge is to lower the learning curve, not on the high end, but the very low end for people who are new to it to have much gentler learning curve"* [P5].

GitHub's file and format support is another main challenge mentioned by the interviewees, and specifically support for formats widely used in education, such as PDF, and Latex. The ability to view or render these formats directly on GitHub, similarly to Markdown rendering, would be very helpful. Especially, if it could support *diff* and review functionalities, with the ability to provide comments without altering the original file. For example, when an instructor wishes to mark the students' assignments or reports, he is forced to download and comment each file (e.g. PDF file) separately, which also alters the original file. Another option is to use the *issues* mechanism of GitHub to provide comments, however, this mechanism can't reference a specific place in the file. Support for slides was also mentioned: *"You can not easily view slides in GitHub, because the PDF is too large and you have to download it, it's a bit cumbersome... in a course you're always have presentation material, so you want some sort of integration with slides here. They (GitHub) have Speaker Deck, so I want Speaker Deck integrated with GitHub."* [P1]

External Restrictions

Interviewees also mentioned external restrictions that limit or prevent them from using GitHub for education. External restrictions can be local restrictions (e.g. a university's policy) or global restrictions (e.g. regional publishing licenses). *"We have LMS systems that we use, and in a way it's very important to use them because they're authenticated by the university. So things like students grades and stuff like that you don't (store on GitHub), because of the rules in US. You cannot store it anywhere else"* [P3].

"Ethically, I wish I could know where the server is, where I am pushing all my material" [P4].

Knowing the server's location plays a significant role when deciding whether you can use GitHub or not. Not only from the ethical aspects but from copyright aspects as well.

"Copyright is a big issue. For instance, we are working with a novel. In Canada, that novel is in public domains so it can be accessed online, but not in United States. So if the server is in US, it could probably be a problem... But if the server is in Canada, it would be OK...Where the server is located determines whether or not you can have it online" [P5].

VALIDATION SURVEY

In the previous section, we described the emerging themes from the interviews. In this section we describe the validation survey intended to validate our findings.

DISCUSSION

GitHub is a good example of a transition from a space to a place, as it transitioned from a hosting service that provides space for projects, to a sharing place. An interesting example that illustrates this was provided at one of our interviews:

"Last year, I introduced a number of deadlines, and the time of the deadline was always... on Friday evening. And the students didn't like it, so they opened an issue on GitHub (to change the deadline)" [P1].

Limitations

The exploratory nature of this study allows us to gain valuable insights to the possibility of using GitHub to support learning. However, this also prevents us from focusing on specific aspects to thoroughly investigate them or other possible applications. Furthermore, this qualitative study should be complemented by a quantitative study in the future.

From the data collection aspect, we collected data from the lecturer's or professor's point of view only, however, the student's point of view is also crucial. It may provide additional support to our findings or reveal new insights. Additionally, the participants of our study were recruited under the condition of already using GitHub to support learning, where it may imply that their experience had to be (somewhat) successful. This means that our study may have missed potential participants who failed to use GitHub to support learning, and the challenges they faced.

CONCLUSIONS

This research examines whether and how GitHub can be used for educational purposes, while relying on its unique collaboration and social features. The contribution of this research is fourfold: (1) present up-to-date overview of related work, (2) investigate current use of GitHub *in the wild* in the context of education, (3) present challenges and problems related to using GitHub for education, and (4) provide recommendations and suggested practices on how GitHub could be used to support learning. Findings of this research will help examine GitHub as tool to support learning, while focusing on potential advantages and disadvantages. Furthermore, it may provide a new perspective on collaboration in education, may lead to evolvement of GitHub from a software development hub into a collaboration platform, and guide practitioners and researchers in building new services and tools to support learning.

Our findings indicate that various educators use GitHub differently, even with similar environments (technical background of the students and the educator) and similar requirements (class size, course type, course goals). In some cases it may happen due to differences in requirements or personal preferences, however, in other cases it happens because GitHub's potential in the context of education is unrealized and educators aren't aware of the possible uses and advantages it may bring. In this sense, suggested practices should be formed and shared among educators, to better utilize GitHub's capabilities as a collaboration platform.

GitHub was initially designed as a version control system with software projects in mind, and it has shown to be useful in several areas that require collaboration, especially in software development and technical fields (e.g. technical writing). In this study, we found that GitHub can useful in the area of education, and provides many benefits. However, this is only the first step on investigating this phenomena. We foresee a rapid adoption of GitHub to support learning and promote collaboration among students and educators in the coming future.

ACKNOWLEDGMENTS

We are extremely grateful to all the interview and survey participants. We would also like to thank GitHub for their collaboration and willingness to help in this study.

REFERENCES

1. Dewan, P. A survey of applications of cscw including some in educational settings. *Proceedings of ED-MEDIA93* (1993), 147–152.
2. Dourish, P., and Bellotti, V. Awareness and coordination in shared workspaces. In *Proceedings of the 1992 ACM conference on Computer-supported cooperative work*, ACM (1992), 107–114.
3. Easterbrook, S., Singer, J., Storey, M.-A., and Damian, D. Selecting empirical methods for software engineering research. In *Guide to advanced empirical software engineering*. Springer, 2008, 285–311.
4. Gikas, J., and Grant, M. M. Mobile computing devices in higher education: Student perspectives on learning with cellphones, smartphones & social media. *The Internet and Higher Education 19* (2013), 18–26.
5. Griffin, T., and Seals, S. Github in the classroom: Not just for group projects. *J. Comput. Sci. Coll. 28*, 4 (Apr. 2013), 74–74.
6. Grosseck, G., and Holotescu, C. Can we use twitter for educational activities. In *4th international scientific conference, eLearning and software for education, Bucharest, Romania* (2008).
7. Grudin, J. Why cscw applications fail: problems in the design and evaluationof organizational interfaces. In *Proceedings of the 1988 ACM conference on Computer-supported cooperative work*, ACM (1988), 85–93.
8. Grudin, J. Cscw: History and focus. *COMPUTER-LOS ALAMITOS- 27* (1994), 19–19.
9. Lacey, A., and Luff, D. *Qualitative data analysis*. Trent Focus Sheffield, 2001.
10. Leland, M. D., Fish, R. S., and Kraut, R. E. Collaborative document production using quilt. In *Proceedings of the 1988 ACM conference on Computer-supported cooperative work*, ACM (1988), 206–215.
11. Majumder, A., Datta, S., and Naidu, K. Capacitated team formation problem on social networks. In *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '12, ACM (New York, NY, USA, 2012), 1005–1013.
12. McGregor, S., Thomson, A., and Ion, W. Cscw in industry and education: Transfering knowledge in engineering design. *proceedings of Concurrent Engineering, CE2000, Lyon, France* (2000).
13. McLure Wasko, M., and Faraj, S. it is what one does: Why people participate and help others in electronic communities of practice. *The Journal of Strategic Information Systems 9*, 2 (2000), 155–173.
14. Neuwirth, C. M., Kaufer, D. S., Chandhok, R., and Morris, J. H. Issues in the design of computer support for co-authoring and commenting. In *Proceedings of the 1990 ACM conference on Computer-supported cooperative work*, ACM (1990), 183–195.
15. Nickerson, R. S., and Zodhiates, P. P. *Technology in education: Looking toward 2020*. Routledge, 2013.
16. Olson, G. M., and Olson, J. S. Distance matters. *Human-computer interaction 15*, 2 (2000), 139–178.
17. Papert, S. Microworlds: transforming education. In *Artificial intelligence and education*, vol. 1 (1987), 79–94.
18. Pappano, L. The year of the mooc. *The New York Times 2*, 12 (2012), 2012.
19. Pressey, S. L. A machine for automatic teaching of drill material. *School & Society* (1927).

20. Reinhardt, W., Wheeler, S., and Ebner, M. All i need to know about twitter in education i learned in kindergarten. In *Key Competencies in the Knowledge Society*. Springer, 2010, 322–332.
21. Roseman, M., and Greenberg, S. Teamrooms: network places for collaboration. In *Proceedings of the 1996 ACM conference on Computer supported cooperative work*, ACM (1996), 325–333.
22. Roth, J. Cscw internet tools and environments for distance education. *University of Hagen Department for Computer Science 58084* (1998).
23. Rutten, N., van Joolingen, W. R., and van der Veen, J. T. The learning effects of computer simulations in science education. *Computers & Education 58*, 1 (2012), 136–153.
24. Seaman, C. B. Qualitative methods in empirical studies of software engineering. *Software Engineering, IEEE Transactions on 25*, 4 (1999), 557–572.
25. Sultan, N. Cloud computing for education: A new dawn? *International Journal of Information Management 30*, 2 (2010), 109–116.
26. Thompson, C. How khan academy is changing the rules of education. *Wired Magazine 126* (2011).
27. Tsay, J. T., Dabbish, L., and Herbsleb, J. Social media and success in open source projects. In *Proceedings of the ACM 2012 Conference on Computer Supported Cooperative Work Companion*, CSCW '12, ACM (New York, NY, USA, 2012), 223–226.
28. Wakeford, N., and Cohen, K. Fieldnotes in public: using blogs for research. *The Sage handbook of online research methods* (2008), 307–326.
29. Wang, C., Shao, M., Xia, J., and Chen, H. Research and design of remote education system based on cscw. In *Grid and Cooperative Computing*. Springer, 2004, 1038–1041.
30. Wilson, P. *Computer supported cooperative work:: An introduction*. Springer, 1991.
31. Wu, Y., Kropczynski, J., Shih, P. C., and Carroll, J. M. Exploring the ecosystem of software developers on github and other platforms. In *Proceedings of the Companion Publication of the 17th ACM Conference on Computer Supported Cooperative Work & Social Computing*, CSCW Companion '14, ACM (New York, NY, USA, 2014), 265–268.