



Adrian Schnedlitz, BSc

# **TimePunch - An Online Timetracking Tool for Education**

**MASTER'S THESIS**

to achieve the university degree of

**Master of Science**

Master's degree programme: Telematics

submitted to

**Graz University of Technology**

Supervisor

Univ.-Prof. Dipl.-Ing. Dr.techn. Wolfgang Slany

Co-Supervisor

Dipl.-Ing. Annemarie Harzl, BSc

Institute for Softwaretechnology

Graz, May 2016

## AFFIDAVIT

I declare that I have authored this thesis independently, that I have not used other than the declared sources/resources, and that I have explicitly indicated all material which has been quoted either literally or by content from the sources used. The text document uploaded to TUGRAZonline is identical to the present master's thesis dissertation.

---

Date

---

Signature

## Acknowledgements

First, I would like to express my gratitude to my supervisors, Annemarie Harzl and Wolfgang Slany, for their excellent support and assistance during the entire process of working on this thesis.

I am deeply grateful for my family, who has supported me during my entire studies.

My special thanks to Markus Hobisch and the whole Catrobat *Musicdroid* - Team who supported me during my testing phase, without their input this thesis would not have been possible.

Graz, May 2016



# Abstract

The way of teaching has changed in the past decades. Due to digitalization, traditional teaching methods are decreasingly practiced. The shift from in-class education to online education requires changes in teaching methods and learning methods for both teachers and students. This change in education implies the use of online tools which are developed especially for education. Most tools for educational purposes focus on dissemination of teaching materials, online examination, feedback functionalities for students and on enabling communication between students and teachers.

On the contrary the effort and time students are investing in their work is rarely a topic. Generally universities in Europe and the US use expenditure of time to determine the appropriate course credits or European Credit Transfer and Accumulation System (ECTS). This system enables students to estimate the expenditure of time for lectures and manage their course plans each semester. However many students have problems managing their time for courses, because ECTS do not match the real time students have to spend on the work in many cases. Moreover many studies have confirmed that effects such as stressors can have negative impacts on students' motivation or psyche and lead to Burnout syndrome.

Thus we argue that time tracking can help teachers to improve lecture expenditure of time estimation. In this thesis we present the design and implementation details of an Atlassian - JIRA add-on. The add-on contains the following input data: date, time spent, assignment identifier, category of work, and detailed description. Categories of work can of course be adapted to the needs of each course. The add-on provides automatic notifications, different data visualizations and data reviewing options for both students and teachers. The tool should enable course instructors to provide more accurate lecture expenditure of time information, which helps students to

manage their time better during a semester. Teachers can monitor if students spend their time accurately during lectures and improve their course or lecture materials based on the recorded data.

**Keywords**

Expenditure of Time, Time Tracking, Educational Time Tracking, Time on Task, Atlassian Timesheet, Timesheet Plugin, JIRA, TimePunch

# Contents

<b>Abstract</b>	<b>vi</b>
<b>1 Introduction</b>	<b>2</b>
1.1 Vocabulary	3
1.2 Motivation	6
1.3 Thesis Outline	8
<b>2 Related Work</b>	<b>10</b>
2.1 Commercial Time Tracking	10
2.2 Educational Methods	10
2.3 Learning Analytics	20
2.4 Stress and Motivation	24
<b>3 Feature Specifications</b>	<b>26</b>
3.1 Catrobat	26
3.2 Requirements	30
3.3 Roles	32
3.3.1 Administrator	32
3.3.2 Supervisor	33
3.3.3 Coordinator	33
3.3.4 User	34
3.4 Additional Requirements	36
<b>4 Existing Implementations</b>	<b>40</b>
4.1 1h - JIRA Timesheet Reports and Gadgets	40
4.2 TimePO - Time tracking and Timesheets	45
4.3 Tempo Timesheets for JIRA	50
4.4 Timesheets for Confluence	55
4.5 Self-Implementation	59

<b>5</b>	<b>Implementation</b>	<b>60</b>
5.1	TimePunch - An Online Timetracking Tool for Education . . .	60
5.2	TimePunch - Technical Details . . . . .	61
5.3	Administration Interface . . . . .	64
5.4	Coordinator Interface . . . . .	71
5.5	Supervisor Interface . . . . .	71
5.6	User Interface . . . . .	72
5.7	Problems and Decisions . . . . .	82
5.8	Results . . . . .	83
<b>6</b>	<b>Summary and Future Work</b>	<b>84</b>
6.1	Summary . . . . .	84
6.2	Future Work . . . . .	85
	<b>Bibliography</b>	<b>92</b>



# List of Figures

2.1	Teaching Methods - In-Class Teaching Model Example. Adopted from (Dang and Gajski, 2014)	12
2.2	Teaching Methods - Online Teaching Model Example. Adopted from (Dang and Gajski, 2014)	13
2.3	Teaching Methods - Cybergogy Model. Adopted from (Tan, M. Wang, and Xiao, 2010)	15
2.4	Teaching Methods - Hybrid Teaching Model Example. Adopted from (Dang and Gajski, 2014)	19
2.5	Learning Analytics - Indicator Based Learning Analytics (LA) Tool. Adopted from (Muslim et al., 2016)	22
3.1	Catrobat Logo	27
3.2	Pocket Code Logo	27
3.3	Google Document Timesheet Overview	31
5.1	TimePunch Overview	63
5.2	TimePunch Configuration - Administration	64
5.3	TimePunch - Administrator Toolbar	65
5.4	TimePunch Configuration - Team Menu	65
5.5	TimePunch Configuration - Category Menu	66
5.6	TimePunch Configuration - Category Management	66
5.7	TimePunch Configuration - Team Management	67
5.8	TimePunch Configuration - Email Configuration	68
5.9	TimePunch Configuration - Inactive Email Text	68
5.10	TimePunch Configuration - Import/Export	69
5.11	TimePunch - Background Jobs	70
5.12	TimePunch - Coordinator Timesheet Select	71
5.13	TimePunch - Timesheet Table	73
5.14	TimePunch - User Timesheet Selection	74

## List of Figures

5.15	TimePunch - Timesheet Entry Date Picker . . . . .	76
5.16	TimePunch - Timesheet Entry Category Picker . . . . .	76
5.17	TimePunch - Timesheet Entry Import Dialog . . . . .	77
5.18	TimePunch - Success Dialog . . . . .	77
5.19	TimePunch - Failure Dialog . . . . .	78
5.20	TimePunch - User Timesheet Summary . . . . .	79
5.21	TimePunch - User Time Visualization . . . . .	80
5.22	TimePunch - User Team Visualization . . . . .	81
5.23	TimePunch - User Timesheet Key Data . . . . .	82

# List of Tables

3.1	Catrobat Team Overview . . . . .	28
3.2	General Requirements . . . . .	31
3.3	Administrator Requirements . . . . .	33
3.4	Coordinator Requirements . . . . .	34
3.5	User Requirements . . . . .	34
3.6	Requirements - Timesheet Entry . . . . .	35
3.7	Data Visualization Requirements . . . . .	35
3.8	Import & Export Requirements . . . . .	36
3.9	User Privilege Requirements . . . . .	37
3.10	Team Requirements . . . . .	37
3.11	Category Requirements . . . . .	38
4.1	Existing Implementations - Add-On Overview . . . . .	40
4.2	1h Add-On - Comparison General Requirements . . . . .	42
4.3	1h Add-On - Comparison Feature Requirements . . . . .	43
4.4	1h Add-On - Comparison Administrator Requirements . . . . .	44
4.5	1h Add-On - Comparison Coordinator Requirements . . . . .	44
4.6	1h Add-On - Comparison User Requirements . . . . .	44
4.7	TimePO Add-On - Comparison General Requirements . . . . .	46
4.8	TimePO Add-On - Comparison Feature Requirements . . . . .	47
4.9	TimePO Add-On - Comparison Administrator Requirements . . . . .	48
4.10	TimePO Add-On - Comparison Coordinator Requirements . . . . .	48
4.11	TimePO Add-On - Comparison User Requirements . . . . .	49
4.12	Tempo Add-On - Comparison General Requirements . . . . .	51
4.13	Tempo Add-On - Comparison Feature Requirements . . . . .	52
4.14	Tempo Add-On - Comparison Administrator Requirements . . . . .	53
4.15	Tempo Add-On - Comparison Coordinator Requirements . . . . .	53
4.16	Tempo Add-On - Comparison User Requirements . . . . .	54
4.17	Timesheet Add-On - Comparison General Requirements . . . . .	56

List of Tables

- 4.18 Timesheet Add-On - Comparison Feature Requirements . . . 57
- 4.19 Timesheet Add-On - Comparison Administrator Requirements 58
- 4.20 Timesheet Add-On - Comparison Coordinator Requirements 58
- 4.21 Timesheet Add-On - Comparison User Requirements . . . . . 59
  
- 6.1 Requirements - Timesheet Future Work . . . . . 86





# 1 Introduction

Education has changed in the last decades, from mainly in-class teaching to Massive Open Online Courses (MOOCs). Today's technology enables educators to reach hundreds or even thousands of lecture participants. To do this they need supporting tools e.g. online platforms to provide learning materials, evaluation tools, online exams and platforms for feedback (Stuchlikova and Kosa, 2013). Topics like Hybrid Teaching (HT) (Dang and Gajski, 2014; King Chan, 2010; Tan, M. Wang, and Xiao, 2010), Massive Open Online Course (MOOC) (Stuchlikova and Kosa, 2013; Iqbal et al., 2014; Abedi and Beikverdi, 2012), web-based teaching, e-learning (Henno, Jaakkola, and Makela, 2014), m-learning (Jianjun et al., 2014) have been tackled already. Others like time tracking supported learning or generally subjects with time management or learners feedback through time tracking still have research needs (Taylor, 2002).

Teaching in practical university courses, regardless if it is done face-to-face, via distance-, or hybrid teaching, mostly focuses on reaching specific learning goals. Actual working hours of students are hardly ever measured (Taylor, 2002), nor is information collected how students spend their time. It is obvious why it is important to test, if students reached the learning goals of a course. We argue it is also necessary to measure the time students invest to reach these goals. This information can help students to increase their motivation and enable teachers to improve their course materials. If students experience too much stress this can have a negative impact on student's intrinsic motivation (Goto and Kusumi, 2015), performance (El-Ghoroury et al., 2012), health and well-being (Saklofske et al., 2012). This may be one reason why Europe and the US have course credits. Credits relate to working hours per semester and limit the amount of credits to a reasonable workload. This system enables students to estimate lecture expenditure of time. It would be easier for students to create course plans, which they can

## 1.1 Vocabulary

successfully pass, if course credits or ECTS are a realistic measure for the effort needed to pass a course. Teachers and students usually have certain expectations regarding how much time students will or should spend on a specific task, for example for understanding the assignments, researching the topic and the actual work. If time is not measured, it is impossible to know if teachers expectations and reality match. Thus teachers do not know if there is time left to teach additional topics or if the syllabus is just right.

To meet needs of both students and teachers a tool is needed which provides the necessary data, is easy to use and does not consume a huge amount of time. It should be adoptable to different courses. To support distributed online teaching an online tool would be an appropriate solution.

## 1.1 Vocabulary

### **Save Harbour Convention**

The Save Harbour Privacy Principles (SHPP)<sup>1</sup> are principles that ensure data protection of European Union and Swiss citizens in non-European countries. The SHPP were cancelled in October 2015<sup>2</sup>.

### **European Credit Transfer System**

The system for rating student performance in Europe is called ECTS<sup>3</sup>. An ECTS point is equal to 25 hours expenditure of time.

### **Free Open Source Software**

The source code of a Free and Open Source Software (FOSS) is openly available and persons have the freedom to apply any kind of changes, or share the source code legally<sup>4</sup>. The open source software license allows users to share or modify software and to use it freely<sup>5</sup>. Some popular open source licences are listed below:

- Apache License 2.0

---

<sup>1</sup><http://ec.europa.eu>, 2016.

<sup>2</sup><http://www.export.gov>, 2016.

<sup>3</sup>HELP.gv.at, 2016.

<sup>4</sup>GNU, 2016.

<sup>5</sup>Initiative, 2016.



## 1 Introduction

- MIT license
- BSD 2-Clause "Simplified" or "FreeBSD" license
- BSD 3-Clause "New" or "Revised" license

### **Online Teaching**

Represents a teaching style where lessons and course materials are provided online. Learners get the opportunity to handle their exercises without attending classes (Silva and Carys de Silva, 2014).

### **Distance Teaching**

Enables students to take part in courses that are geographically distributed (Abedi and Beikverdi, 2012).

### **Hybrid Teaching**

A combination of in-class and online teaching (Dang and Gajski, 2014)

### **Massive Open Online Course**

Providing lecture experience and teaching materials online to massive amounts of students (Abedi and Beikverdi, 2012).

### **Near Field Communication**

Near Field Communication (NFC) allows two devices to communicate with each other within a near field range (several centimetres) and is defined by several communication protocols<sup>6</sup>.

### **Pocket Code**

Is an application which enables users to easily create programs or animations in a visual programming language very similar to Scratch (Slany, 2016).

Pocket Code (PCo) is actually available for:

- Android
- iOS
- Windows Phone

### **Android**

An open source operation system for mobile phones from Google<sup>7</sup>.

---

<sup>6</sup>NFC, 2016.

<sup>7</sup>Android, 2016a.

### **Windows Phone**

An operation system for mobile phones from Microsoft<sup>8</sup>.

### **iOS**

An operation system for mobile phones from Apple<sup>9</sup>.

### **Smart Phone**

A mobile phone with high computation power, a larger display and the option to install applications<sup>10</sup>.

### **Jira**

A time and issue tracking tool from Atlassian<sup>11</sup> that enables several useful features for agile software developing. It is possible to create detailed project plans as well as explicit project reports, or visualize team performance. Jira can be extended by third party developed extensions and has been used for bug reports and issue tracking at the Catrobat project for more than two years.

### **Confluence**

A organization tool that offers wiki functionalities for company employees, work discussions and reviews for customers from Atlassian<sup>12</sup>. The software has been used as project calendar, project wiki, announcement platform and discussion forum at the Catrobat project for more than one year.

### **Jenkins**

Is a cross-platform application that can be used to build and run software tests continuously<sup>13</sup>.

### **Lightweight Directory Access Protocol**

An internet protocol for distributed directory access<sup>14</sup>.

---

<sup>8</sup>Microsoft, 2016.

<sup>9</sup>Apple, 2016.

<sup>10</sup>Scoop, 2016.

<sup>11</sup>Atlassian, 2016b.

<sup>12</sup>Atlassian, 2016a.

<sup>13</sup>Jenkins, 2016.

<sup>14</sup>IETF, 2016.

## 1 Introduction

### **Git**

A system that enables version control, is open source and freely available<sup>15</sup>.

### **GitHub**

Is a software project hosting platform<sup>16</sup>.

## 1.2 Motivation

The main focus of this thesis is to implement an online time tracking tool that provides recording, managing and evaluating expenditure of time in a comfortable way for students and educators. The tool should support lecturers to improve their university courses with practical exercises based on the collected students time data. The timesheet add-on should enable students to track their time on tasks for reaching course goals. Teachers' can use this information to evaluate the scope of the work items. Catrobat project members currently have to fill in their spent working hours into a Google Document Timesheet (GDTS) manually. Project participants must have a valid Google account, because it is required for accessing and sharing Google Drive documents.

Storing data on an external cloud service, as the Google Drive, enables service providers analysing the stored information on their servers. Since the European Court of Justice has revoked the Save Harbour Convention (SHC) in November 2015. Members of Graz University of Technology (TU Graz) are not allowed to store data that way any longer. Thus student timesheet data should not be stored on external servers.

The data is actually stored in Google Document (GD) timesheets, which do not provide any sort of automated background data verification or email notifications in case of issues. With the present solution project administrators are not able to generate a GDTS automatically. The following tasks have to be accomplished for every student:

---

<sup>15</sup>Git, 2016.

<sup>16</sup>GitHub, 2016.

## 1.2 Motivation

1. Log into the administration Google account
2. Switch to Google Drive and replicate the user timesheet template
3. Rename the GDTS
4. Fill in the required fields
5. Share the document with the user with read/write access

If a teacher wants to examine the progress of a student, the following steps have to be accomplished:

1. Request read-only or read-write permission from administrator, if the teacher is not the administrator
2. Wait until the administrator provides the permission
3. Log into his private Google account
4. Look into the user's GDTS
5. Check, if everything is correct (manually)
6. Notify the administrator, if the permission is not needed any more

The timesheet owner is responsible for keeping his timesheet up to date and formatting all entries correctly. The present time tracking solution does not provide any automatic data verifications. Students have to execute all of the following operations correctly:

- Copy required formula, e.g. calculating the total amount of working time for one task, from a field
- Paste required formula into a field
- Use correct date formatting
- Use correct time formatting
- Insert new row correctly
- Delete a row correctly

If students do not execute these steps accurately, *Copy & Paste* errors or wrong formatted data can cause broken formulas. This results in a corrupt student timesheet, because many formulas are connected with each other. Experience has shown that some students do not track their time promptly in the provided GDTS, they note their spent time on paper or in their own calendar instead. Incorrect, false or lost data are the result. The GDTS has also noticeable performance issues if a timesheet table grows and has more than 200 entries. Besides if Google applies any changes, adoptions or

## 1 Introduction

extensions to their documents, existing timesheet tables have to be updated manually.

Therefore a new time tracking solution should replace the GDTS, offer a more comfortable way for students to track their time and support teachers and administrators. The new solution should solve all problems from the current solution and reduce administrative workload of teachers. Students will still have to fill in their sheet entries manually, but date pickers, time pickers, field formatting checks and error notifications enable simple time tracking and prevent user mistakes in the future.

### 1.3 Thesis Outline

Chapter 1, contains the motivation for this master thesis and a short problem description. Chapter 2, includes commercial time tracking and related work results. A detailed problem description and an overview about software requirements is given in Chapter 3. Chapter 4, includes a comparison between existing time tracking add-ons and our requirements. In Chapter 5, the add-on implementation is described in detail. Chapter 6, contains the summary of this thesis and future work.



## 2 Related Work

This chapter intends to give an overview about commercial time tracking and different teaching methods.

### 2.1 Commercial Time Tracking

Time attendance systems are used to record the working time of company employees. This is usually done with RF-ID cards (Simao, Fonseca, and Santos, 2008). Within a conventional time attendance system, each of these cards is assigned to an employee (Simao, Fonseca, and Santos, 2008) who can sign-in, at the beginning of the working and sign-out, at the end of the working day. The card can be used to sign-out and leave the company area during breaks. However since employees are identified by these cards, it is possible to pass cards to other persons and manipulate working time. This is a major problem, using this sort of time tracking system (Simao, Fonseca, and Santos, 2008; Soewito et al., 2015). Long waiting lines can occur if a lot of employees arrive or leave at the same time (Soewito et al., 2015). This form of tracking working time is not feasible if an employee is working outside of the company building (Soewito et al., 2015) which happens frequently at the Catrobat project.

### 2.2 Educational Methods

Education has changed over the past decades. Traditional teaching is decreasingly practised at universities and online teaching methods are rising

## 2.2 Educational Methods

(Jayasinghe, Dharmaratne, and Atukorale, 2015). The online teaching revolution offers many different teaching methods such as:

- Distance teaching
- Online teaching
- MOOC
- Hybrid teaching

All teachings methods have pros and cons, some of them are applicable for large numbers of course attendees, others provide individual support for students and others enable education over wide distances. Online teaching methods must ensure that students are not left alone with the course materials and get the option to stay on track during lectures (Dang and Gajski, 2014).

### **Traditional In-Class Teaching**

Traditional in-class teaching or face-to-face teaching takes place in a classroom and learners have to be present during lessons. Students have to learn provided content from school books, write exams and solve exercises (Alhasani, 2015). As Dang and Gajski (2014) mentioned, this teaching method is applicable for about 15-20 lecture participants. A possible in-class teaching model which enables students to ask questions during discussion sessions is pictured in Figure 2.1.



## 2 Related Work

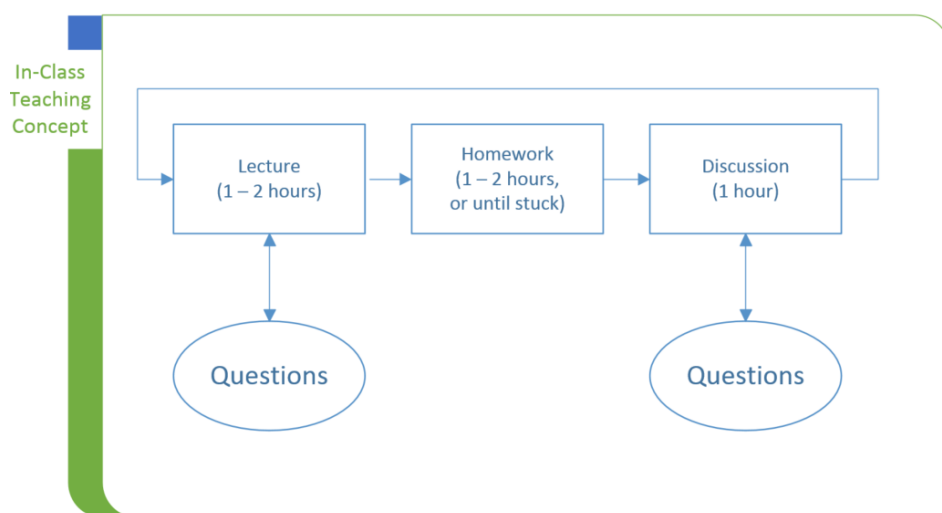


Figure 2.1: Teaching Methods - In-Class Teaching Model Example. Adopted from (Dang and Gajski, 2014)

In-class education offers great benefits for lectures with a small number of attendees. This teaching method enables teachers watching the behaviour of students, which is not possible in online teaching environments (Jayasinghe, Dharmaratne, and Atukorale, 2015). The direct interaction of learner and lecturer which is pictured in Figure 2.1 enables teachers to provide support and individual feedback for each student. This can have positive effects on student's learning process (Jayasinghe, Dharmaratne, and Atukorale, 2015). However location and time are important at in-class teaching (Jayasinghe, Dharmaratne, and Atukorale, 2015) and advantages, like personal contact, are diminished, when too much people participate (Dang and Gajski, 2014).

### Distance-Teaching

The popularity of Distance Teaching (DT) has grown over the past decades. It provides education over wide distances and enables students to interact with the teacher, other students or groups without location constraints (He and L. Wang, 2009). Course instructors have to face a great number of new challenges regarding online course or online lecture material design e.g. provide online tool interaction (Tan, M. Wang, and Xiao, 2010). The available

## 2.2 Educational Methods

technologies enable students to access learning materials from anywhere and at any time (He and L. Wang, 2009). For instance the School of the Air project offers distance education to students, who live far away from school in Australia (Crump, 2013). Lecture materials may include different media types, students can learn from independent sources.

Providing location-independent and flexible learning environments for students are great benefits of DT compared to traditional in-class education He and L. Wang (2009) identified challenges as:

- Problems caused by inexperienced instructors
- Insufficient technology explanations
- Problems caused by the technology itself

### Online-Teaching

As Jayasinghe, Dharmaratne, and Atukorale (2015) mentioned, the newer form of DT is called Online Teaching (OT). Figure 2.2 shows the work flow of a typical OT model, which assumes self-motivation by the lecture participants (Dang and Gajski, 2014). Just using an online environment for publishing course materials to transfer an in-class lecture to an online one is not enough (Dang and Gajski, 2014; Sieber, 2005).

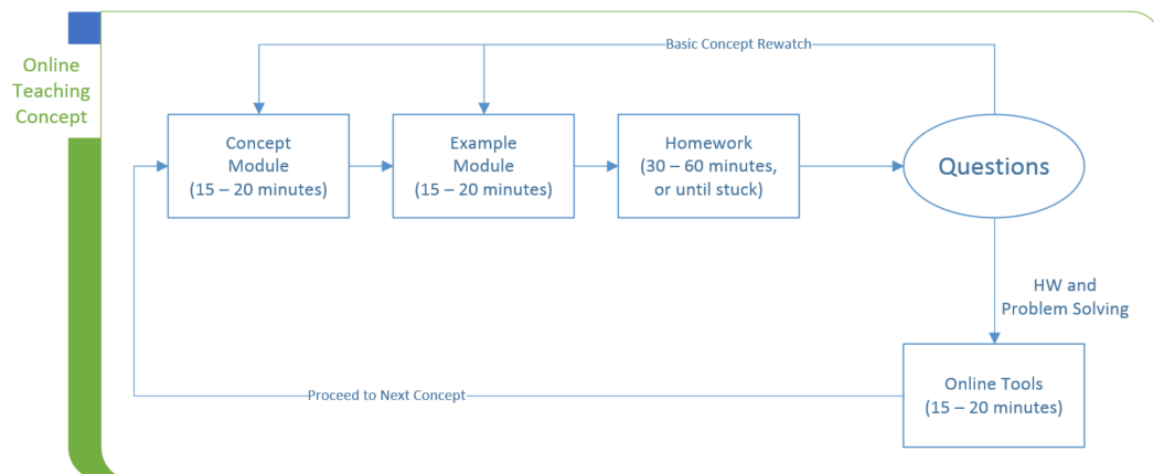


Figure 2.2: Teaching Methods - Online Teaching Model Example. Adopted from (Dang and Gajski, 2014)

## 2 Related Work

According to M.G. (1989) there exist three important types of basic interaction between students, course materials and the instructors, which take place in DT (Taylor, 2002). These interactions are listed below:

1. Student - Student
2. Student - Course material
3. Student - Instructor

It is possible to separate interaction into social interaction which represents interaction with other persons like friends, instructors or colleagues and individual interaction that is related to interactions with the course materials (Shu-ping and Hai-ying, 2013). There are five crucial principles for designing an OT course, as Tan, M. Wang, and Xiao (2010) mentioned in their article. These principles are listed below:

1. Meet the *Seven Principles in Undergraduate Education* (Chickering and Gamson, 1989)
2. Ensure instructor roles
3. Assure the Cybergogy learning model
4. Directness of instructors
5. Evaluation

The *Seven Principles in Undergraduate Education* (Chickering and Gamson, 1989) cover the following fields:

1. Connection between students and teacher
2. Teamwork among students
3. Focus on active learning
4. Provide immediate feedback
5. Attach importance on engaged time (time on task)
6. Convey high expectations
7. Acceptance of different properties and learning skills

Tan, M. Wang, and Xiao (2010) mentioned an online learning model for engaged learning. This Cybergogy model is shown in Figure 2.3 and highlights the importance of social, cognitive and emotive factors to provide a supportive online education environment.

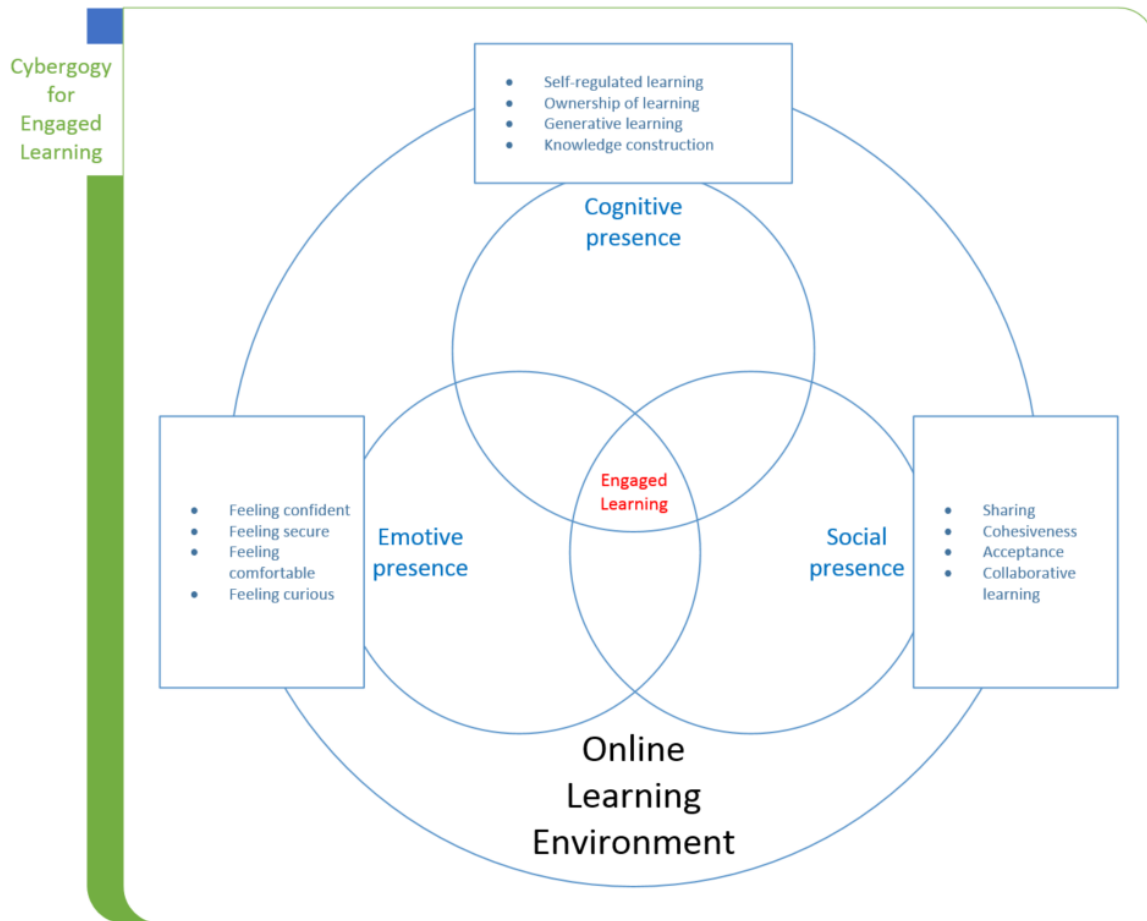


Figure 2.3: Teaching Methods - Cybergogy Model. Adopted from (Tan, M. Wang, and Xiao, 2010)

According to Tan, M. Wang, and Xiao (2010), it is the duty of a course instructor to ensure online teaching quality. For this reason the online teacher has to manage different roles and bring in social skills, because verbal as well as non-verbal communication can have positive effects on students learning (Tan, M. Wang, and Xiao, 2010). The four different online instructor roles are mentioned in the following list:

1. Pedagogic - Act as intermediary in education

## 2 Related Work

2. Social - Establish a friendly class environment
3. Managing - Manage all course relevant tasks (e.g.: organize information) (Tan, M. Wang, and Xiao, 2010)
4. Technical - Assure the used technology does not hinder students at learning

The technical role should additionally cover managing technical problems. Students should not take care about them and focus on learning instead (Tan, M. Wang, and Xiao, 2010).

According to Tan, M. Wang, and Xiao (2010), lecture evaluations provide feedback for the instructors. This feedback supports course instructors to adapt course materials to students' needs. It provides information if a technology needs to be changed or if students need help to accomplish their tasks. The California State University (CSU) Chico, has created some guidelines to ensure good quality for undergraduate online teaching, as (Tan, M. Wang, and Xiao, 2010) mentioned. Those guidelines highlight the importance of course evaluations and feedback options for attendees.

As mentioned before it is necessary to ensure that students stay on track during online courses. Students' perseverance depends partly on the quality of the lecture materials and the instructor skills (Dang and Gajski, 2014). If materials and instructor skills do not exceed a certain level of quality, students are more likely to drop out of a course. Therefore instructors have to spend a lot of time, providing good quality course materials, improving lecture materials and offering feedback to students. In large courses it is not possible to notice emotional states of lecture members (Jayasinghe, Dharmaratne, and Atukorale, 2015) impacting students motivation and therefore not possible to react accordingly (Tan, M. Wang, and Xiao, 2010). Students are left alone with motivational issues.

OT technology provides a flexible learning environment (Tan, M. Wang, and Xiao, 2010) and has no limitations related to location or time (He and L. Wang, 2009). OT provides education for a larger number of course participants which is not possible with traditional education methods (Dang and Gajski, 2014).

### **Massive Open Online Courses**

MOOCs are courses for great numbers of attendees who live geographically

## 2.2 Educational Methods

distributed and represent a recent form of OT (Stuchlikova and Kosa, 2013). Some MOOCs offer free online education providing lecture materials and supportive documents online for the lecture participants (Iqbal et al., 2014; Abedi and Beikverdi, 2012). Students get the opportunity to communicate, interact with other students and get in touch with the teaching staff by using course forums (Stuchlikova and Kosa, 2013) or mailing lists (Dang and Gajski, 2014). It is not the purpose of MOOCs to replace the conventional teaching method within classrooms (Stuchlikova and Kosa, 2013). This teaching principle should provide a new form of education for learners (Stuchlikova and Kosa, 2013). Stuchlikova and Kosa (2013) mentioned some MOOCs in their article:

- Khan Academy
- Google Course Builder
- Class2Go
- edX

According to Iqbal et al. (2014) the MOOC teaching method does not fit for every learner, because not all students are able to manage self-directed learning, as the high drop out rates of 90% show (Iqbal et al., 2014; Stuchlikova and Kosa, 2013). The popularity of MOOCs is rising at Universities since 2012 (Iqbal et al., 2014). The MOOC technology could revolutionize the world (Stuchlikova and Kosa, 2013) and provide reusable course content thus MOOCs can be cost effective (Abedi and Beikverdi, 2012). As there is no standardization for online education it is resulting in unsteady students' performance (Dang and Gajski, 2014). (Stuchlikova and Kosa, 2013) mentioned in their article this education method has to face challenges like:

- Quality of MOOC must be ensured
- Participants are responsible for learning by themselves
- Ensure course material quality and organization
- High drop out rate
- Cheating

### **Hybrid-Teaching**

HT is a combination of face-to-face and OT. As Dang and Gajski (2014) mentioned in their article, in-class education is not possible with a lot of

## 2 Related Work

participants and online teaching is not applicable if students are left alone with their problems. Figure 2.4 shows a possible model which is applicable for courses with several hundred participants. This model provides in-class sessions for students who need additional assistance in the form of discussions, exercises, tutoring or online tools as forum, mailing lists and additional materials (Dang and Gajski, 2014). These additional sessions can take place within classrooms or using remote conferencing tools, to get in touch with the teaching staff (Dang and Gajski, 2014). This enables personal feedback for students depending on their needs. The costs of this teaching method are lower compared to in-class teaching methods with a great numbers of course attendees, although additional teaching staff for tutoring sessions and online support is needed (Dang and Gajski, 2014).

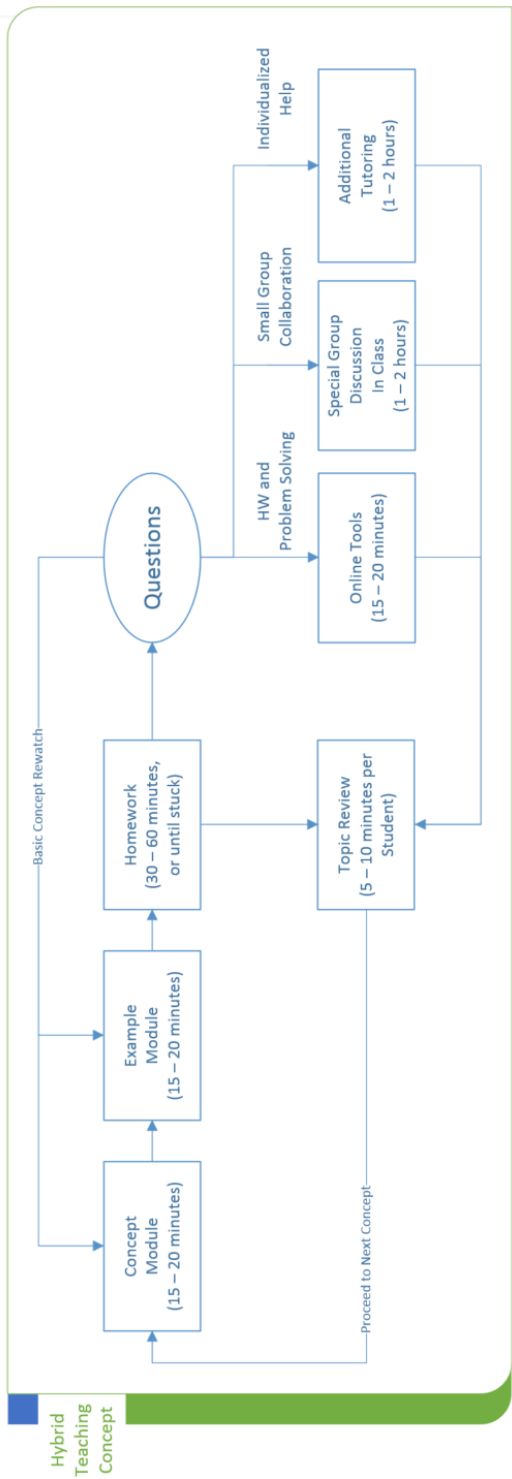


Figure 2.4: Teaching Methods - Hybrid Teaching Model Example. Adopted from (Dang and Gajski, 2014)

## 2.2 Educational Methods



## 2.3 Learning Analytics

LA is a challenging field which intends to improve teaching through analysing educational data (Muslim et al., 2016). This can be accomplished by tracking students learning behaviours or interactions with course materials (Muslim et al., 2016). Muslim et al. (2016) present a LA tool, that enables teachers to collect their data for analytics through rule-based indicators. Figure 2.5 shows a indicator based, LA tool by (Muslim et al., 2016), that involves users into the indicator definition process. The article also mentions, example indicators for an indicator based LA tool:

- Student activity tracking
- In-class presence
- Forum post rate
- Wiki post rate

## 2.3 Learning Analytics

## 2 Related Work

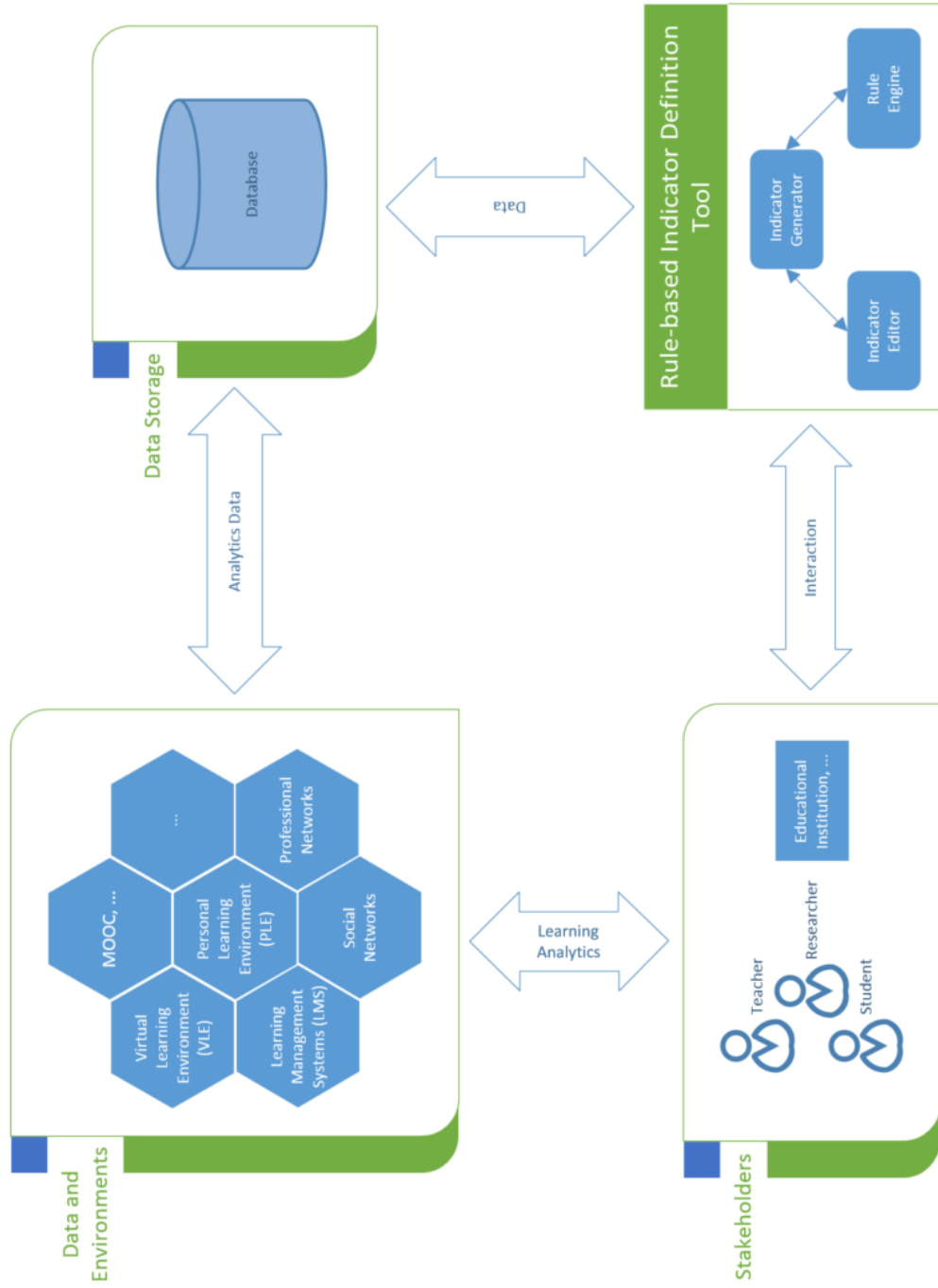


Figure 2.5: Learning Analytics - Indicator Based LA Tool. Adopted from (Muslim et al., 2016)

## 2.3 Learning Analytics

Teachers might imply, if students are able to reach a certain educational level, they are able to organize their time and motivate themselves (Taylor, 2002).

According to Taylor (2002) watching students posting activity in forums enables lecturers to estimate learners expenditure of time during course tasks. Liu (2015) found out that, research about the relationship of students' academic motivation and academic stress was only done by some surveys. Several studies discovered associations between negative emotional factors or negative dispositional factors and students health (Saklofske et al., 2012).

Thus it appears important to us to enable course instructors to examine if the time students have to spend on lectures exceeding the number of working hours, defined in the study plan.

### 2.4 Stress and Motivation

The measurement of expenditure of time during university classes is very important, because work overload can lead to stress, which may have negative influence on students motivation (Liu, 2015) or cause depression as well as other psychological responses (Duan et al., 2015). In Europe the required time for University courses is expressed by the ECTS<sup>1</sup>, whereas in the US the US Credit System (USCS) is applied. ECTS show the overall time students have to spent for a lecture compared to course credits<sup>2</sup> which express the required lecture hours instead. If a course is too much time consuming compared to its ECTS the students' performance might suffer (El-Ghoroury et al., 2012). Recording the spent time on tasks could enable students to arrange their available time more efficiently and reduce academic stress during the semester.

We argue that tracking students time on task is necessary to create a balance between course ECTS and the real expenditure of time. This might improve course materials due to students direct feedback (Tan, M. Wang, and Xiao, 2010). An appropriate syllabus could help minimize stress and burnout, thus improving students' health and motivation (Smith and Dillard, 2006).

---

<sup>1</sup>HELP.gv.at, 2016.

<sup>2</sup>ISAS, 2016.



## 3 Feature Specifications

In our opinion a manageable schedule is important for students' health and performance. A sustainable pace allows students to profit more from their education. They are more likely to engage in the courses, when they are not completely overworked and stressed-out. Teachers know if the syllabus is too ambitious, not challenging enough, or just right.

In the past we used GDTS to measure the necessary effort to pass our courses. Using GDTS proved to be error-prone and caused a huge administrative workload. Apart from the total amount of hours the GDTS did not provide any statistical data, as mentioned in Section 1.2.

Thus we want to implement a tool which is able to replace the currently used GDTS, offers more analysis options and automation of administrative tasks. The add-on will be tested with the help of the Catrobat project. The requirements for a new tool, role definitions and functionalities are mentioned in this chapter.

### 3.1 Catrobat

Catrobat<sup>1</sup> is a visual programming language that is used within the PCo<sup>2</sup> Android<sup>3</sup> application. PCo acts as an Integrated Development Environment (IDE) and allows first-time and casual users to develop their own applications on their Android smart phones or tablets. The application furthermore

---

<sup>1</sup>Catrobat, 2016a.

<sup>2</sup>Catrobat, 2016b.

<sup>3</sup>Android, 2016b.

### 3.1 Catrobat

allows to control several external hardware via Bluetooth<sup>4</sup> or Wi-Fi<sup>5</sup>, such as the Phiro robot<sup>6</sup>, Parrot drones<sup>7</sup>, Bluetooth Arduino boards<sup>8</sup> or Lego Mindstorms robots<sup>9</sup>. The Catrobat project was inspired by Scratch<sup>10</sup> and the main differences compared to Scratch are:

1. No other hardware than a smartphone is required by PCo
2. Programs have to be interpreted by clicking on the play button within PCo (player and IDE are displayed simultaneously within Scratch)
3. Device sensors like acceleration, compass, face-detection, are fully supported
4. Multiple simultaneously wireless connections are possible
5. Multi-touch mobile devices are fully supported

PCo is the application of the Catrobat FOSS project. The project was initiated by Wolfgang Slany, the head of the Institute for Software Technology (IST), at the TU Graz and the application is developed by students.



Figure 3.1: Catrobat Logo



Figure 3.2: Pocket Code Logo

#### Catrobat - Project Overview

The IST at the TU Graz offers students the possibility to take part in the Catrobat project. As shown in Table 3.1 there is a great diversity of available

---

<sup>4</sup>Bluetooth, 2016.

<sup>5</sup>WiFi, 2016.

<sup>6</sup>Phiro, 2016.

<sup>7</sup>Parrot, 2016.

<sup>8</sup>Arduino, 2016.

<sup>9</sup>LegoMindstorms, 2016.

<sup>10</sup>Scratch, 2016.



### 3 Feature Specifications

topics.

The Catrobat is a FOSS project. Students can do their bachelor thesis or master project within the scope of Catrobat. Catrobat uses a HT approach. Materials are provided online or shared between students and project tasks can be solved alone or using pair-programming sessions.

Name	Description
Catroweb	Catroweb team
Chromecast	Team for Google Chromecast support in Pocket Code
Core	Core team of Pocket Code
Design	Design team
Drone	Team for controlling the AR Drone via Pocket Code
HTML5	HTML5 team
iOS	Team that implements Pocket Code for smart phones with iOS
Jenkins	Team that administrates the test-server environment
Lego Robot	Team for controlling the Lego Mindstorm robots via Pocket Code
Musicdroid	Team for Musicdroid - playing music on your mobile phone
NFC	Team for NFC (Near Field Communication) support Pocket Code
NPO Marketing	Marketing team
Paintroid	Team for Paintroid - drawing pictures on your mobile phone
Phiro	Team for controlling the Phiro robot via Pocket Code
Physics Engine	Team for physic engine support in Pocket Code
RasperIno	Team for controlling the RasperyPi and Arduino with Pocket Code
Windows Phone	Team that implements Pocket Code for smart phones with Windows

Table 3.1: Catrobat Team Overview

Students can work at home or at the University. Students can get in touch with other members, if they struggle during tasks. Since project teams develop simultaneously on different tasks, it is necessary for separate member roles to ensure a regulated work flow. The different roles are listed below:

- Administrator
- Coordinator
- Developer

This kind of hierarchical role allocation keeps the project manageable and decisions where they belong, e.g. developers usually discuss problems

### 3.1 Catrobat

within the team and development progress with their team coordinator. Project coordinators forward only important information to supervisors or administrators during regular meetings.

#### Catrobat - Infrastructure

Catrobat project members have to use several different platforms for specific tasks, which are listed below:

- IRC<sup>11</sup> (Internet Relay Chat) - Is used to communicate with Catrobat members all over the world
- Google Drive - Is used to access GDTS, for time data recording
- Atlassian Confluence - Is used as project wiki
- Atlassian JIRA - Is used for issue tracking

Thus project participants have to remember several different platform credentials. This causes time delays during development, if students have to log on different platforms simultaneously. For this reason we want to reduce the number of different platforms and follow a *One Account for Everything - Philosophy*.

#### Catrobat - Confluence

Catrobat members can use the Atlassian Confluence web interface for:

- Team calendar
- Information pages of all Catrobat teams with descriptions of their topics
- Specific project team data like contact data, user names, coordinator and administrator names
- Meeting notes
- How-to-tutorials of important topics

This offers students a great platform where important information about the Catrobat project and upcoming events can be found.

---

<sup>11</sup>IRC.at, 2016.

## 3 Feature Specifications

### Catrobat - JIRA

JIRA<sup>12</sup> is used for issue tracking, bug reports and dividing work load between the developers. The JIRA Agile board provides information about task states and which member is currently working on project tasks.

## 3.2 Requirements

The tool requirement list was created in cooperation with the timesheet administrator, the core team coordinators and senior members of the project. The tool should provide helpful features, which are currently not available in GDTS. The add-on should be easy to use and look like the actual GDTS table, which is shown in Figure 3.3. Features as add, edit or remove timesheet entries are the most basic requirements. Additional features include user-specific timesheet settings, diagrams and automation e.g. email notifications. The tool should also collect more detailed information about the time students spend on tasks. For example, how many hours students spend on different categories like programming, researching the topic, discussing tasks. If students spend an unusual amount of time on researching and discussing the tasks this could be a sign that the task description needs some clarification. The tool should support Light Weight Access Protocol (LDAP), because we want to reduce the number of accounts per person. The solution should enable students who participate in projects or work in teams, to track their expenditure of time. Students should not be able to spy on the time other students' have spent, to avoid any form of competition. Thus access to other timesheets should be restricted to certain roles, which are described in Section 3.3.

The core plugin specifications are shown in Table 3.2, administrator features are listed in Table 3.3, coordinator features are mentioned in Table 3.4 and user features are presented in Table 3.5.

---

<sup>12</sup>Atlassian, 2016b.

## 3.2 Requirements

Stunden bisher:		Gesamtstunden letzte 30 Tage:	
	6:00:00	1,33%	
Stunden übrig:		444:00:00	98,67%
Monat (letzter Tag)	Stunden im Monat	%	% bisher
31.08.2015	0:00:00	0,00%	1,33%
31.07.2015	0:00:00	0,00%	1,33%
30.06.2015	0:00:00	0,00%	0,94%
31.05.2015	0:00:00	0,00%	1,33%
30.04.2015	0:00:00	0,00%	1,33%
31.03.2015	0:00:00	0,00%	1,33%
28.02.2015	0:45:00	0,17%	1,33%
31.01.2015	5:15:00	1,17%	1,17%

Figure 3.3: Google Document Timesheet Overview

Requirement	Description
Timesheet	Everyone in a certain (adjustable) group receives a timesheet automatically
Remaining Hours	View remaining hours
Completion Prediction	View a completion prediction
Add Course To Timesheet	Users can add course names to their timesheet
View Assigned Courses	User can view assigned lecture names + ECTS
Change Timesheet Hours	User can change the amount of target hours
Display Expenditure of Time	User can display work time (e.g.: average per month, each month, last 6 months)

Table 3.2: General Requirements

The major aspect of this master thesis is to provide an efficient online tool that enables students to track time they invest for courses. In our opinion

## 3 Feature Specifications

the tool should offer information to students and teachers and not serve as ranking list.

### 3.3 Roles

The tool should support some kind of rights management and user roles. These roles are specified in the list below:

- Administrator
- Supervisor
- Coordinator
- User

#### 3.3.1 Administrator

Administrators should be able to modify the tool configuration, define project teams, assemble team categories and assign categories to teams, as summarized in Table 3.3.

Administrators are responsible for the following tasks:

- Arrange members to teams
- Manage access to GDTS
- Add, Modify, Remove project user accounts

Requirement	Description
Add, Modify, Remove Teams	Administrator can add, modify and delete teams in the tool configuration
Add/Remove Categories	Administrator can add and delete categories in the tool configuration
Assign Category To Team	Administrator can assign categories to teams in the tool configuration
Assign User to Team	Administrator can assign users to teams in the tool configuration
Assign User to Coordinator Group for Team	Administrator can assign coordinators to teams in the tool configuration
Assign Supervisors	Administrator can assign supervisors in the tool configuration
Assign Tool Administrators	Administrator can assign administrators in the tool configuration
Team Activity	Visualize data of users, or teams
Define Email Notification	Define email notification text and when they occur
Edit Timesheet Entry	Permission to change user timesheet entries

Table 3.3: Administrator Requirements

### 3.3.2 Supervisor

This role allows members restricted access to all available student timesheets. Viewing data is permitted, but writing rights are not granted.

### 3.3.3 Coordinator

Coordinators are users who carry out organizational tasks within their team and assist Catrobat project members. Table 3.4, shows the coordinator

### 3 Feature Specifications

requirements. Members who are assigned to this role are able to display timesheet overviews of teams.

Requirement	Description
Display Team Member Timesheet	Coordinator can view timesheets of team members
Notifications	Coordinator receives email notifications related to his team members

Table 3.4: Coordinator Requirements

#### 3.3.4 User

Catrobat project members are assigned to the user role. This role receives access to all course materials which are required for developing. All project members obtain access to the Catrobat Confluence board, the project JIRA board, and their personal timesheet. Table 3.5, represents the user requirements.

Additional features as data representation, data visualization, team data visualization or time sheet entry behaviour are listed in Table 3.6 and Table 3.7.

Requirement	Description
Timesheet	Every user gets a timesheet with a unique ID
Edit entry	Possibility to edit an entry for a certain time
Visualization	Every user is able to visualize his data
Team visualization	Every user is able to visualize the data of his team
Notifications	Email notification if an administrator changes an entry

Table 3.5: User Requirements

Requirement	Description
Entry	User can create entries in his own sheet with several fields (e.g.: date, pause, category, ...)
Master Thesis Entry	User can create entries in his own master thesis sheet with several fields (e.g.: date, pause, category)
Change Entry Value	User can change his entries for a certain time
Change Entry Value Administrator	Administrator can change entries any time
Inactive Entry	User can define an inactive entry

Table 3.6: Requirements - Timesheet Entry

Requirement	Description
User Timesheet Entries Visualization	User can visualize his timesheet entries
User Timesheet Categories Visualization	User can visualize his spent time related to categories
Team Timesheet Visualization	User can visualize the timesheet entries of the team he is assigned to
Team Categories Visualization	User can visualize the spent time related to categories of the team he is assigned to

Table 3.7: Data Visualization Requirements

Table 3.8, shows additional, essential user requirements. The import function for existing timesheet entries from GDTS is one of the most important requirements for the new tool. The new tool's acceptance would suffer dramatically, if users need to import hundreds of existing entries manually.



### 3 Feature Specifications

Requirement	Description
Import From Google Documents	Import timesheet entries from an existing Google Document, or CSV table
Import All Timesheets	Import timesheets from a CSV file
Export Timesheet	Export the timesheet data of a user as CSV
Export All Timesheets	Export all available timesheets as CSV

Table 3.8: Import & Export Requirements

## 3.4 Additional Requirements

The new tool should provide different user roles with varying privileges. Table 3.9, highlights the administrator and user permissions, because they are necessary to ensure privacy.

### 3.4 Additional Requirements

Requirement	Description
Permission	Administrators are able to apply changes to the configuration
Timesheet Write Access User	Own timesheet
Timesheet Write Access Administrator	All timesheets
Timesheet Read Access User	Own timesheet
Timesheet Read Access Administrator	All timesheets
Change Timesheet Read Access	Administrator can give supervisors read access to all timesheets

Table 3.9: User Privilege Requirements

The tool should provide task categories. Administrators can define categories related to working tasks, which are usable within the timesheet. Users are able to select both their team and a suitable category, for tracking their *Time On Task*. The features, related to user teams and task categories, are shown in Table 3.10 and Table 3.11.

Requirement	Description
Team	User should get the possibility to select a team for his working hours entry if he is assigned to multiple teams
Team Visualization	Every team member is able to visualize the expenditure of time of his team
Team Member Overview	Shows which member is currently active

Table 3.10: Team Requirements

### 3 Feature Specifications

Requirement	Description
Category	User can select a category for his working hours entry which depend on team memberships
Create Category	Administrator can create categories
Delete Category	Administrator can delete existing categories
Assign Category To Team	Administrator can assign categories to teams
Pair Programming	User can choose team member(s) with a drop down menu for this category

Table 3.11: Category Requirements



## 4 Existing Implementations

Based on the requirements, identified in Chapter 3, we searched for existing tools. Self-implementation should only be considered if there are no appropriate solutions available, because it carries several risks which will be discussed in Section 4.5. Table 4.1 shows the search results of timesheet or time tracking add-ons for JIRA and Confluence. In general, all add-ons are constructed for commercial use and do not provide any task category support. The add-ons require the use of JIRA tickets or an external database for tracking the expenditure of time.

Name	Platform
1h	JIRA
TimePO	JIRA
Tempo	JIRA
Timesheet for Confluence	Confluence

Table 4.1: Existing Implementations - Add-On Overview

### 4.1 1h - JIRA Timesheet Reports and Gadgets

The 1h - JIRA Time Sheet Reports and Gadgets application<sup>1</sup> provides both gadgets and reports to visualize how much time the user spends on issues. The add-on enables users to display their working hours on the Confluence- or JIRA-dashboard. The tool offers filter options for the timesheet and project pivot report to show data of persons, groups or the whole project. The time users or groups spend on a project for a specific time period can

---

<sup>1</sup>Zhdanov, 2016.

## 4.1 1h - JIRA Timesheet Reports and Gadgets

be displayed and also exported. These reports can be used for billing project costs to customers. The plugin pricing depends on the number of users, but open-source projects can use it for free. All information was taken from the 1h plugin documentation (Reports and Gadgets, 2016a) and the product Youtube-Video<sup>2</sup>.

### Comparison - General Requirements

The 1h plugin does not support course management for users at all. It seems as if the add-on is constructed for industry needs and not for educational purpose. Table 4.2 shows which requirements the 1h add-on fulfills and which it does not. The timesheet entries are connected to Jira issues which would cause additional tickets for the Catrobat project. Every student would have to create an issue for every individual activity e.g.:

- Topic Research
- Meeting
- Pair Programming Session
- Programming Sessions
- Code Reviewing Sessions

To the best of our knowledge the 1h plugin does not provide categories for project tasks as listed in Section 3.2.

---

<sup>2</sup>Reports and Gadgets, 2016b.

#### 4 Existing Implementations

Requirement	1h
User Timesheet	Yes
Show Remaining Hours	Unknown
Add Course To Timesheet	No
View Assigned Courses	No
Change Timesheet Expenditure of Time	No
Display Expenditure of Time	Yes
Check Time Threshold	No
Check Timesheet Effort	Yes
Timesheet Entry	Time related to issue
Master Thesis Timesheet	No
Change Entry Value	Yes
Import Entry	Unknown
Change Entry As Administrator	Unknown
Inactive Entry	Yes

Table 4.2: 1h Add-On - Comparison General Requirements

In Table 4.3, the feature-requirements are compared to the 1h solution. The Jira implementation provides a timesheet overview with a built-in calendar, representing tickets that are assigned to a user, creating a project pivot overview showing the user expenditure of time or building a work time chart that summarizes the spent user time.

#### 4.1 1h - JIRA Timesheet Reports and Gadgets

Requirement	1h
Change User Permission	Unknown
Supervisor Role	Unknown
Administrator Can Access All Timesheets	Unknown
Team Management	Group
Team Visualization	Work Time Chart
Team Member Overview	Unknown
Category Management	No
Assign Category To Team	No
Pair Programming Feature	No
User Timesheet Entries Visualization	No
User Timesheet Categories Visualization	No
Team Timesheet Visualization	Project Report
Import From Google Document, or CSV	Unknown
Export Timesheet	Timesheet Report via Email
Export All Timesheets	No

Table 4.3: 1h Add-On - Comparison Feature Requirements

#### Comparison - Administrator Requirements

As shown in Table 4.4, the administrator is not able to assign categories to teams or manage categories at all. Users are able to subscribe to timesheet reports to receive regular report emails which is not as important as receiving notifications about students activity states, remaining project hours of team members or if administrators applied changes to student timesheets.



## 4 Existing Implementations

Requirement	1h
Add, Modify, Remove Teams	Unknown
Add, Remove Categories	No
Assign Category To Team	No
Assign User To Team	No
Assign Coordinators To Team	No
Define Email Notification	Unknown
Edit Timesheet Entry	Change JIRA issue

Table 4.4: 1h Add-On - Comparison Administrator Requirements

### Comparison - Coordinator Requirements

Assigning coordinators to project teams is one of the core requirements as mentioned in Table 4.5, but to the best of our knowledge the 1h plugin does not support any other role than administrator or user.

Requirement	1h
Display Team Member Timesheet	Unknown
Team Visualization	Project Pivot Chart

Table 4.5: 1h Add-On - Comparison Coordinator Requirements

### Comparison - User Requirements

As shown in Table 4.6, using the 1h plugin for educational purpose would cause problems sooner or later.

Requirement	1h
Timesheet	Yes
Edit Entry	Change Ticket
Visualization	Project Report
Automated Notifications	Timesheet Report via Email (requires subscription)

Table 4.6: 1h Add-On - Comparison User Requirements

### 4.2 TimePO - Time tracking and Timesheets

The Time Posting Online (TimePO) - Time tracking and Timesheets<sup>3</sup> implementation which is also provided for Jira, offers visualization features for spent time on tasks. The add-on enables employees to report their working hours in a timesheet view, based on created Jira tickets. Thus the tool enables managers to display the project progress. The application offers separated user interfaces for both managers and employees. Managers are additionally able to control the spent time on work for a team or project. Timesheets can be arranged on the JIRA dashboard and a team timesheet view is also provided. The task report is designed for commercial use just as the pivot report from the previously explained 1h plugin. Described details were taken from the TimePO plugin documentation (Siamenchykau, 2016).

#### Comparison - General Requirements

The timesheet data is represented as a table with detailed information about the time each employee has spent for an issue. It is possible to select a view with less details. The application provides reports for managers and employees to represent the overall spent time for a project or a team and timesheet table export features for managers. However the data representation looks similar compared to the 1h add-on.

---

<sup>3</sup>Ltd., 2016.

## 4 Existing Implementations

Requirement	TimePO
User Timesheet	Yes
Show Remaining Hours	No
Add Course To Timesheet	No
View Assigned Courses	No
Change Timesheet Expenditure of Time	No
Display Expenditure of Time	Yes
Check Time Threshold	No
Check Timesheet Effort	Yes
Timesheet Entry	Time related to issue
Master Thesis Timesheet	No
Change Entry Value	Change ticket, work log value
Import Entry	No
Change Entry Value Administrator	Unknown
Inactive Entry	Unknown

Table 4.7: TimePO Add-On - Comparison General Requirements

The TimePO plugin offers a team overview for managers, to review project team working hours. However some users complained that content representation gets very confusing, if a project has a large number of team members. Thus this tool might not be a good choice for educational purposes since courses can have hundreds of participants. In Table 4.7, the general timesheet requirements are compared to the TimePO solution.

The Jira add-on offers display features for timesheet data on the JIRA dashboard. A built-in calendar represents the time a user spent on an issue and report functions for both team timesheet and user timesheet report. As shown in Table 4.8 and Table 4.9 the plugin does not support categories, but provides different user permissions. Administrators do not have any restrictions and users can only access their personal work log. These permissions can not be adopted to our project needs as it does not satisfy the core requirements from Section 3.2 and the TimePO add-on does not provide team permissions, but offers different reports as shown in Table 4.10. Users are only able to assign their working log to Jira tickets and administrators can view this data afterwards.

## 4.2 TimePO - Time tracking and Timesheets

Requirement	TimePO
Change User Permission	Yes
Supervisor Role	Unknown
Administrator Access All Timesheets	Unknown
Team Management	Unknown
Team Visualization	Team Timesheet Report
Team Member Overview	Unknown
Category Management	No
Create Category	No
Delete Category	No
Assign Category To Team	No
Pair Programming	No
User Timesheet Entries Visualization	Timesheet Report
User Timesheet Categories Visualization	No
Team Timesheet Visualization	Project Report
Import From Google Document, or CSV	No
Export Timesheet	No, Report only
Export All Timesheets	No, Reports only

Table 4.8: TimePO Add-On - Comparison Feature Requirements

### Comparison - Administrator Requirements

The TimePO administrator privilege enables users to display time tracking logs from other users and create reports. This fulfills the coordinator requirements in Section 3.3.3.

## 4 Existing Implementations

Requirement	TimePO
Add, Modify, Remove Teams	No
Add, Remove Categories	No
Assign Category To Team	No
Assign User To Team	No
Assign Coordinators To Team	No
Define Email Notification	No
Edit Timesheet Entry	Unknown

Table 4.9: TimePO Add-On - Comparison Administrator Requirements

### Comparison - Coordinator Requirements

The TimePO add-on provides different user roles. The plugin administrator can be compared with the coordinator role for our educational needs.

Requirement	TimePO
Display Team Member Timesheet	Unknown
Team Visualization	Project Pivot Chart

Table 4.10: TimePO Add-On - Comparison Coordinator Requirements

### Comparison - User Requirements

As shown in Table 4.11, the TimePO implementation provides all core user requirements except email notifications, but informing users, coordinators and administrators about remaining expenditure of time, timesheet states or manipulated table entries, is essential and must be provided by the time tracking plugin.

## 4.2 TimePO - Time tracking and Timesheets

Requirement	TimePO
Timesheet	Calendar
Edit Entry	Change Work Log
Visualization	Team, User Timesheet Report
Automated Notifications	No

Table 4.11: TimePO Add-On - Comparison User Requirements

### 4.3 Tempo Timesheets for JIRA

The JIRA add-on Tempo - Timesheets for JIRA<sup>4</sup> is provided for both JIRA and JIRA Agile. The add-on is intended to reduce inaccurate customer bill and enable simple time tracking for employees. The time tracking is also connected to JIRA tickets. The tool is targeted for industrial use and can be commonly used with the following add-ons:

- Tempo Planner - provides visualizations for team & project planning
- Tempo Books - extension tool for Tempo Timesheets
- Tempo Folio - view and manage project costs and scope on portfolio level

The Tempo - Timesheets plugin provides functionalities like:

- User, project, team, account and advanced timesheet
- Different timesheet views - timesheet, calendar, report and list
- Export - printable, PDF and Excel
- Permissions
- Team management
- Billable time tracking
- Internal cost tracking
- Project capitalization measurement
- Budget for the future measurement

All information was taken from the plugin documentation (JIRA, 2016b).

#### Comparison - General Requirements

A comparison of the general and additional Tempo timesheet requirements is shown in Table 4.12 and Table 4.13. The plugin offers three different export functionalities. The Tempo add-on is also designed for companies as the employees work-time-tracking for customer billing shows. Furthermore it does not allow tracking expenditure of time on categories which is an essential requirement for educational use, as mentioned in Section 3.2.

---

<sup>4</sup>JIRA, 2016a.

### 4.3 Tempo Timesheets for JIRA

Requirement	Tempo
User Timesheet	Yes
Show Remaining Hours	Yes
Add Course To Timesheet	No
View Assigned Courses	No
Change Timesheet Expenditure of Time	No
Display Expenditure of Time	Yes
Check Time Threshold	No
Check Timesheet Effort	Yes
Timesheet Entry	Time related to issue
Master Thesis Timesheet	No
Change Entry Value	Change ticket value
Import Entry	Unknown
Change Entry Value Administrator	Unknown
Inactive Entry	Yes

Table 4.12: Tempo Add-On - Comparison General Requirements



## 4 Existing Implementations

Requirement	Tempo
Change User Permission	Yes
Supervisor Role	Team Manager
Administrator Can Access All Timesheets	Unknown
Team Management	Yes
Team Visualization	Team Timesheet
Team Member Overview	Team Timesheet
Category Management	No
Create Category	No
Delete Category	No
Assign Category To Team	No
Pair Programming Feature	No
User Timesheet Entries Visualization	Yes
User Timesheet Categories Visualization	No
Team Timesheet Visualization	Team Timesheet
Import From Google Document, or CSV	No
Export Timesheet	Yes
Export All Timesheets	Unknown

Table 4.13: Tempo Add-On - Comparison Feature Requirements

### Comparison - Administrator Requirements

With the Tempo add-on, users require an additional permission to get timesheet access. The plugin provides user, team manager and administrator roles.

### 4.3 Tempo Timesheets for JIRA

Requirement	Tempo
Add, Modify, Remove Teams	Unknown
Add, Remove Categories	No
Assign Category To Team	No
Assign User To Team	No
Assign Coordinator To Team	No
Team Activity	Unknown
Define Email Notification	Unknown
Edit Timesheet Entry	Unknown

Table 4.14: Tempo Add-On - Comparison Administrator Requirements

#### Comparison - Coordinator Requirements

The TimePO plugin satisfies almost all core requirements from Section 3.2, but it is not possible to track expenditure of time with this plugin without using JIRA tickets. As shown in Table 4.14, tracking expenditure of time based on categories is not possible, thus the use of this plugin is not suitable for our purposes.

Requirement	Tempo
Display Team Member Timesheet	Yes
Team Visualization	Yes

Table 4.15: Tempo Add-On - Comparison Coordinator Requirements

#### Comparison - User Requirements

Users are able to view the estimated-, remaining- and working time and overall spent time for all issues of a project which is useful at companies, but not necessary for educational purposes. As shown in Table 4.15 and Table 4.16, the Tempo add-on provides several different timesheet styles like:

- User Timesheet - Main timesheet for users

#### 4 Existing Implementations

- Project Timesheet - Helpful overview for project managers
- Team Timesheet - Approved users can select other timesheets from this page
- Account Timesheet - Provides an overview of accounts for users
- Advanced Timesheet - Views invalid worklog and available JIRA filter for timesheets
- Issue Timesheet - Shows a timesheet related to issues

Requirement	Tempo
Timesheet	Yes
Edit Entry	Change Ticket/ Change Effort
Visualization	Report, List View, Calendar View
Automated Notifications	Unknown

Table 4.16: Tempo Add-On - Comparison User Requirements

## 4.4 Timesheets for Confluence

Timesheets for Confluence<sup>5</sup> is available for Confluence instead of JIRA like the other three implementations. It enables users to track their expenditure of time via a My Structured Query Language (MySQL) database interface. The pricing depends on the number of users, similar to the 1h plugin and provides a burndown chart for visualizing estimated project time. The administrator is able to show timesheet data of all users and the implementation tends to be like the mentioned plugins for company needs. Besides it is not possible to define categories, different user roles or export timesheet data, as shown in Table 4.17, Table 4.18 and Table 4.19. All informations were taken from the plugin documentation<sup>6</sup>.

### Comparison - General Requirements

The plugin does not support any export since version *Confluence Server 5.1.3 - 5.9.9 2016-01-26*. To ensure data reliability either version control or export options for the plugin configuration and the user timesheet data must be provided.

---

<sup>5</sup>i4ware-Software, 2016a.

<sup>6</sup>i4ware-Software, 2016b.

#### 4 Existing Implementations

Requirement	Timesheet
User Timesheet	Yes
Show Remaining Hours	Yes
Add Course To Timesheet	No
View Assigned Courses	No
Change Timesheet Expenditure of Time	No
Display Expenditure of Time	Yes
Check Time Threshold	No
Check Timesheet Effort	Yes
Timesheet Entry	MySQL Database Entries
Master Thesis Timesheet	No
Change Entry Value	No
Import Entry	No
Change Entry Value Administrator	Unknown
Inactive Entry	No

Table 4.17: Timesheet Add-On - Comparison General Requirements

#### 4.4 Timesheets for Confluence

Requirement	Timesheet
Change User Permission	Unknown
Supervisor Role	No
Administrator Can Access All Timesheets	Yes
Team Management	Ticket Based
Team Visualization	Burndown Chart
Team Member Overview	Only Burndown Chart
Category Management	No
Create Category	No
Delete Category	No
Assign Category To Team	No
Pair Programming Feature	No
User Timesheet Entries Visualization	Only Burndown Chart
User Timesheet Categories Visualization	No
Team Timesheet Visualization	Burndown Chart
Import From Google Document, or CSV	No
Export Timesheet	No, Removed With Version 1.3
Export All Timesheets	No

Table 4.18: Timesheet Add-On - Comparison Feature Requirements

#### Comparison - Administrator Requirements

The Timesheet plugin provides only an administrator and user role. An external configured MySQL database is required and no other user is allowed to view other timesheets except the administrator who can not apply changes to existing timesheet entries. Thus the plugin does not meet our requirements from Section 3.2.

## 4 Existing Implementations

Requirement	Timesheet
Add, Modify, Remove Teams	No, MySQL Based
Add, Remove Categories	No
Assign Category To Team	No
Assign User To Team	No
Assign Coordinator To Team	No
Team Activity	Burndown Chart Only
Define Email Notification	No
Edit Timesheet Entry	No, Only View Entries

Table 4.19: Timesheet Add-On - Comparison Administrator Requirements

### Comparison - Coordinator Requirements

As mentioned in Section 3.3.1, the administrators must be able to change user permissions, manage categories and assign them to teams. As shown in Table 4.20, the Timesheets for Confluence add-on does not provide other roles except administrator and user.

Requirement	Timesheet
Display Team Member Timesheet	Only Administrator
Team Visualization	Unknown

Table 4.20: Timesheet Add-On - Comparison Coordinator Requirements

### Comparison - User Requirements

This plugin does not provide core requirements from Section 3.2, as shown in Table 4.21 and the burndown chart visualization is not able to represent single user timesheets.

## 4.5 Self-Implementation

Requirement	Timesheet
Timesheet	Yes
Edit Entry	Change MySQL Entry
Visualization	Burndown Chart
Automated Notifications	No

Table 4.21: Timesheet Add-On - Comparison User Requirements

## 4.5 Self-Implementation

We did not find any existing add-on for Confluence or JIRA that implements all requirements from Chapter 3. Using a JIRA ticket based time tracking tool was not an option since creating issues for every category for each task is not useful at all. An existing tool would have been preferable, because in-house development may cause problems as:

- The plugin must be maintained and updated by developers of the Catrobat project
- Problems caused by JIRA updates have to be fixed by developers of the Catrobat project
- Testing, bug fixing, deployment has to be done by developers of the Catrobat project

Therefore we decided to implement a tool ourselves, because of the following advantages:

- The developer knows exactly how the data is processed
- The developers decide where the data is stored
- The functionality can be pre-defined with a feature list / requirement list
- The features fit exactly our requirements
- Extensibility of the add-on is possible



## 5 Implementation

This chapter contains details about the TimePunch implementation, Graphical User Interface (GUI) and its provided features.

### 5.1 TimePunch - An Online Timetracking Tool for Education

As mentioned in Section 3.1, the Catrobat project follows the *one account for everything* philosophy, because remembering login credentials for different platforms especially within a FOSS project can be pretty annoying for its members. Since the wiki page was ported to Confluence and the issue tracking is managed by JIRA it was possible to reduce the number of required user accounts. A next step to reducing the number of accounts, was to move time tracking to a platform which supports LDAP and would fit seamlessly into the existing infrastructure. As mentioned in Section 4, no existing implementation was found that fulfilled all necessary requirements. Which is why the decision for a in-house developed tool was made. It was obvious to examine Confluence and JIRA as possible tool platform, because Catrobat was using these Atlassian products and these platforms provide built-in databases already. Due to database limitations, Confluence was excluded. JIRA represented the better choice, therefore we decided to develop a JIRA add-on. Thus developing an independent JIRA add-on, with access to the project LDAP data was decided.

## 5.2 TimePunch - Technical Details

### Development

Atlassian provides online support for third party developers including tutorials, forums and documentations for the JIRA RESTful API<sup>1</sup>. This simplified the developing process. Both Confluence and JIRA offer several built-in configuration possibilities like:

- Email Account - It is possible to configure incoming and outgoing email server settings
- Security - All connections are encrypted (https)
- External Data - Usage of external databases, like LDAP is possible
- User Management - User management and user account management
- Data Storage - Provided database
- Data Access - Interfaces to the internal database

### Server - Client Communication

The usage of RESTful web services<sup>2</sup> enables client - server communication that can be extended in the future. Well defined and clearly separated interfaces are provided. If a user requires data from the server a *HTTP-GET-REQUEST*<sup>3</sup> is created on client side within JavaScript and send to the server. If a valid client request is detected by the server, the required data is collected, manipulated or arranged to the requested format and send back to the client. In case of success the client receives a server response , otherwise it receives a *HTTP-RESPONSE-ERROR*<sup>4</sup>.

---

<sup>1</sup>Atlassian, 2016c.

<sup>2</sup>Oracle, 2016.

<sup>3</sup>W3, 2016a.

<sup>4</sup>W3, 2016b.

## 5 Implementation

### External Libraries

The *morris.js*<sup>5</sup> plugin was used to create the diagrams.

### Testing

The Server side Java code was tested using User Interface (UI) tests, according to the Agile Software Development (ASD) philosophy of the Catrobat project. The IntelliJ Maven plugin - Clover<sup>6</sup> was utilized using IntelliJ IDEA<sup>7</sup> as developing IDE, because it offers support for Java, HTML and JavaScript development.

### TimePunch - Workflow

The TimePunch add-on model is shown in Figure 5.1. The add-on can be accessed from everywhere, using the Internet. Data security is provided by JIRA, using encrypted data sessions which is achieved through *HTTPS-Connections*<sup>8</sup>. The data exchange between client and server, takes place as mentioned above. User data retrieved from the Catrobat JIRA server and the data processing is handled on the client side.

---

<sup>5</sup>morris.js, 2016.

<sup>6</sup>Atlassian, 2016d.

<sup>7</sup>Jetbrains, 2016.

<sup>8</sup>Technopedia, 2016.

## 5.2 TimePunch - Technical Details

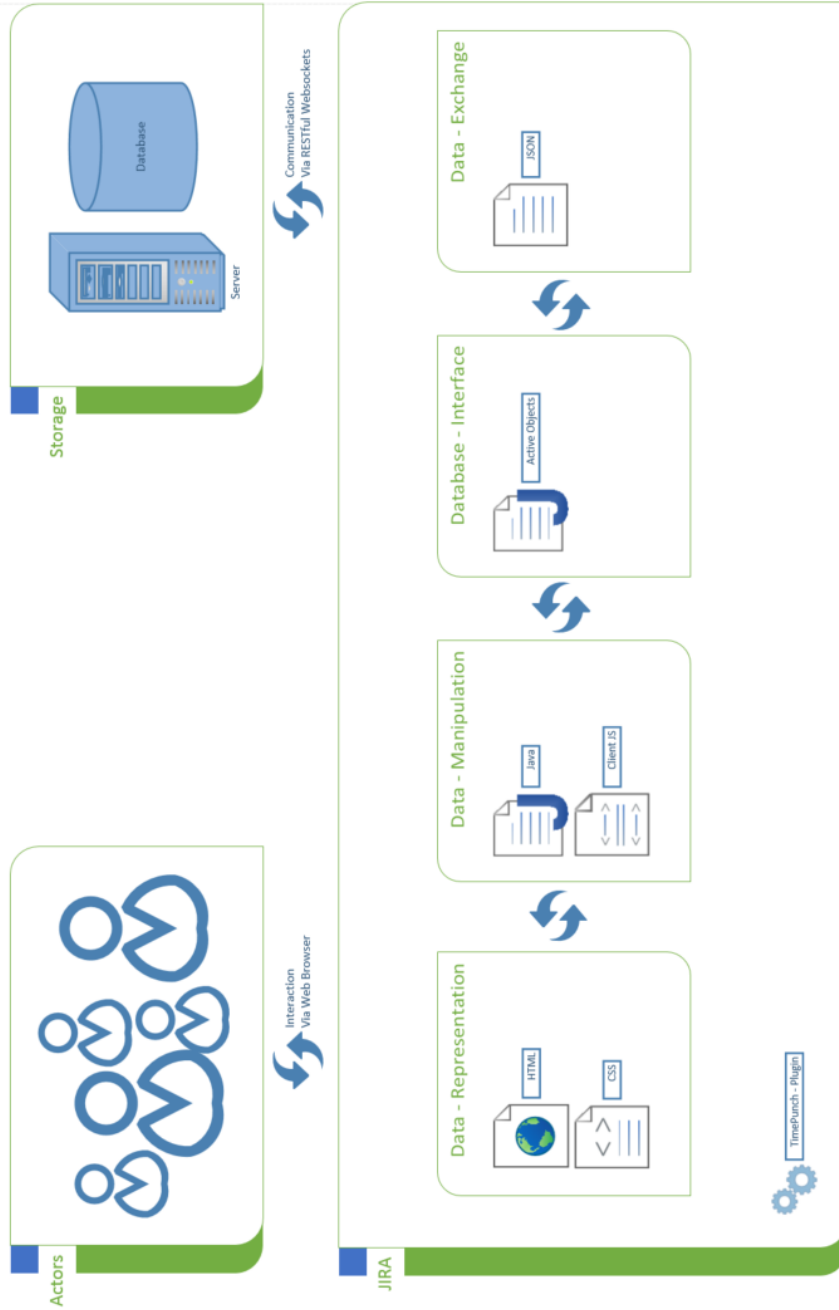


Figure 5.1: TimePunch Overview

## 5 Implementation

### 5.3 Administration Interface

#### TimePunch - Configuration

Users who are assigned to the *jira-administrators* LDAP group can grant other administrators or LDAP-groups access to the TimePunch - Configuration as shown in Figure 5.2.

##### Supervisor Timesheet Read Access

Supervisors

##### TimePunch Configuration Access

Allowed Groups/Users

- If nothing is given, every Jira Administrator may read/write
- If a group is given, every Jira Administrator within this group may read/write
- If a single user is given and this user is a Jira Administrator this user may read/write

Figure 5.2: TimePunch Configuration - Administration

If no user, or group is selected all users who are assigned to the *jira-administrators* LDAP group may access and edit the TimePunch - Configuration. If a user is assigned to the TimePunch - Administrator group, the add-on configuration can be accessed as shown in Figure 5.3.

## 5.3 Administration Interface

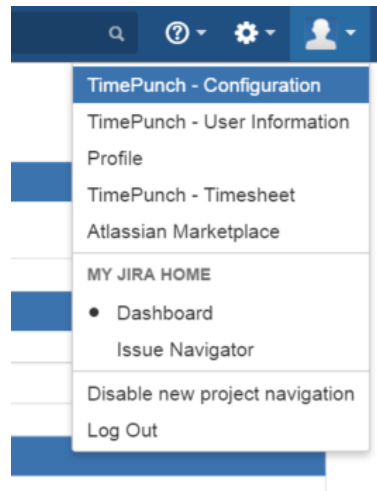


Figure 5.3: TimePunch - Administrator Toolbar

The TimePunch - Administrator is able to add, edit, remove or rename teams and assign users to them as shown in Figure 5.4.

### Team

Removing a team just effects the configuration and no user will be modified.

Team name <sup>\*</sup>

Figure 5.4: TimePunch Configuration - Team Menu

Add-on administrators are additionally able to add, remove or rename task categories, using the add-on configuration, as shown in Figure 5.5 and Figure 5.6.

## 5 Implementation

### Category

Removing a category does not modify any user.

Category name\*

Figure 5.5: TimePunch Configuration - Category Menu

### Category Management

**Important Note:** Do not forget to 'save' your changes.

Bug fixing

Bug fixing (PP)

Code Acceptance

Discussing issues/Supporting/Consulting

Inactive

Meeting

Figure 5.6: TimePunch Configuration - Category Management

TimePunch offers add-on administrators to assign task categories and users to available teams, as shown in Figure 5.7. This enables users to select those

categories for their expenditure of time. If users are assigned to the team coordinator group, they receive additional privileges.

### Team Configuration

**Important: Do not forget to 'save' your changes.**

Catroid  Edit Team Name

Coordinator	<input type="text" value="Select coordinators"/>
User	<input type="text" value="Select users"/>
Category	<input type="text" value="Programming x"/> <input type="text" value="Meeting x"/> <input type="text" value="Testing x"/>

Figure 5.7: TimePunch Configuration - Team Management

The TimePunch - Configuration provides adaptability of email notifications. Add-on administrators can define the outgoing email sender address and the outgoing sender name, as shown in Figure 5.8. It is furthermore possible to define both email subject and email text for all automated notification jobs, as shown in Figure 5.9. If the email text field left empty, a default message will be used. If an administrator applies any sort of changes to a user timesheet entry, an email with the following fields is send to both user and administrator:

- Reason – the administrator must enter a reason for manipulating the timesheet entry of an user
- Time stamp – the current time is added to the email notification if an administrator applies changes to an user timesheet entry
- Original entry – the original timesheet entry
- New entry – the changed timesheet entry



## 5 Implementation

### E-Mail Notifications

From Name   
The name of the sender which is shown in the welcome mail.

From   
The e-mail address of the sender which is shown in the welcome mail.

Figure 5.8: TimePunch Configuration - Email Configuration

Inactive

Subject   
The subject of the mail, if a student was not active since two weeks.

Body

The text of the mail. Following strings will be replaced by Confluence automatically:

- {{name}}: The name of the user
- {{date}}: The date of the users last entry

Figure 5.9: TimePunch Configuration - Inactive Email Text

### TimePunch - Backup

As data integrity was one of the major tasks, import and export functions for all TimePunch - User timesheets and the TimePunch - Configuration are provided as pictured in Figure 5.10. Users are able to import and export their timesheet data themselves.

### Export

Download 'All Timesheets' as [CSV](#).

Download 'Config' as [CSV](#).

### Import

Upload an existing 'TimePunch - Configuration' [here](#).

Upload existing 'TimePunch - Timesheets' [here](#).

Figure 5.10: TimePunch Configuration - Import/Export

## TimePunch - User Information Overview

TimePunch provides a user-information-overview which shows a summary of the following user fields:

- LDAP user information (user name and email address)
- LDAP user account status
- TimePunch timesheet status
- Last timesheet entry

This feature enables TimePunch - Administrators to verify the current state of TimePunch - Users, which is an immense benefit compared to the previously used GDTS.

## Background Jobs

The TimePunch implementation provides supportive features, which are triggered automatically or manually by a TimePunch - Administrator. The add-on offers an *Activity Verification* task which verifies the current timesheet

## 5 Implementation

state of all TimePunch - Users. The *Activity Notification* task notifies users, e.g. user, team coordinator or administrator, if a user timesheet is inactive, especially if the timesheet state is marked as inactive for a longer period. Additionally an *Out Of Time Notification* task which automatically informs timesheet owners, if their remaining expenditure of time is below a defined threshold, is provided.

All this tasks are triggered automatically by Linux cron jobs<sup>9</sup>. These jobs are sending periodical *HTTP-GET* requests to specific REST-URLs and the JIRA server triggers the corresponding methods. TimePunch - Administrators can force these jobs manually, as shown in Figure 5.11.

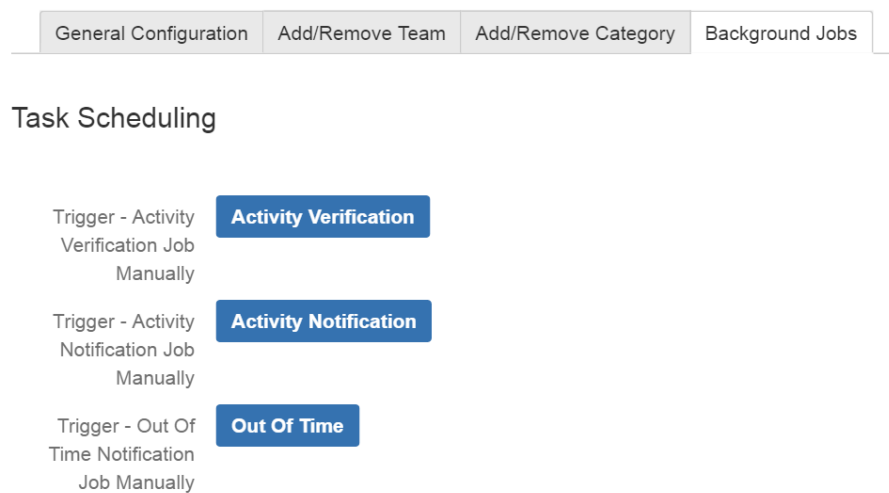


Figure 5.11: TimePunch - Background Jobs

### Timesheet - Manipulation

As listed in Section 3.3, TimePunch - Administrators should be permitted to apply changes to student timesheets. If an add-on administrator modifies a user entry, the timesheet owner and the administrator are informed with an email notification. This notification contains the original entry and the

---

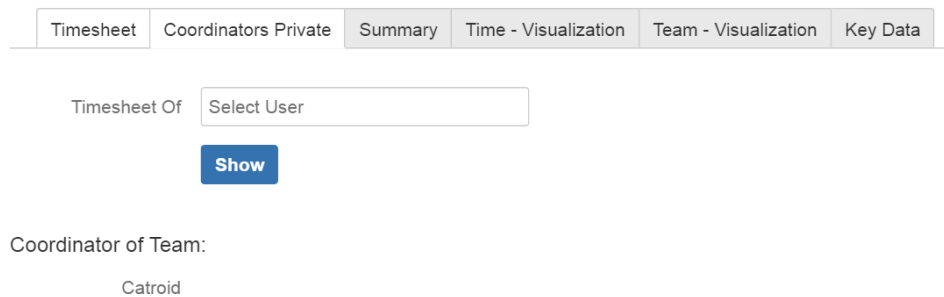
<sup>9</sup>ubuntuusers, 2016.

## 5.4 Coordinator Interface

changed one, because the tool currently does not support data version control. Thus any timesheet data manipulations by third party must be documented to ensure transparency.

## 5.4 Coordinator Interface

The TimePunch - Coordinator has the permission to display all timesheet entries and information of team members, as shown in Figure 5.12.



The screenshot shows a web interface for selecting a timesheet. At the top, there is a horizontal navigation bar with six tabs: 'Timesheet', 'Coordinators Private', 'Summary', 'Time - Visualization', 'Team - Visualization', and 'Key Data'. Below the navigation bar, the text 'Timesheet Of' is followed by a text input field containing the placeholder text 'Select User'. Below the input field is a blue button labeled 'Show'. Further down, the text 'Coordinator of Team:' is followed by the name 'Catroid'.

Figure 5.12: TimePunch - Coordinator Timesheet Select

## 5.5 Supervisor Interface

TimePunch - Supervisors are able to display all entries and information of available TimePunch timesheets. Both supervisors and coordinators can only display timesheet entries, only TimePunch - Administrators are able to manipulate entries.

## 5 Implementation

### 5.6 User Interface

The TimePunch add-on enables users to display their timesheet data as a list, as shown in Figure 5.13. This list is sorted by date, thus the latest timesheet entry is shown on top of the table. Users who are assigned to the *timesheet* LDAP group are able to access their personal TimePunch - Timesheet, as shown in Figure 5.14.

Timesheet										Import from Google Docs
Coordinators Private	Summary	Time - Visualization	Team - Visualization	Key Data						
Date	Inactive End Date	Start - End	Pause	Duration	Team	Category	Pair Programming	Jira Ticket ID	Task Description	
2016-04-18	yy-mm-dd	10:25 - 11:25	00:00		Music Droid	Theory (MT)				✓
2016-04-09		00:25 - 01:03		00:38	Music Droid	Refactoring		None	Test 1	
2016-04-08		17:10 - 22:15		05:05	Music Droid	Pair programming		None	Test 2	
2016-04-07		10:00 - 14:10		04:10	Music Droid	Theory		None	Test 3	
2016-04-06		20:27 - 21:30		01:03	Music Droid	Programming		None	Test 4	
2016-04-05		18:20 - 19:40		01:20	Music Droid	Pair programming		None	Test 5	
2016-04-04		11:00 - 14:20	00:30	02:50	Music Droid	Pair programming		None	Test 6	
2016-04-03		00:30 - 02:00		01:30	Music Droid	Pair programming		None	Test 7	
2016-04-02		16:50 - 20:00	00:15	02:55	Music Droid	Pair programming		None	Test 8	

Figure 5.13: TimePunch - Timesheet Table

## 5.6 User Interface

## 5 Implementation

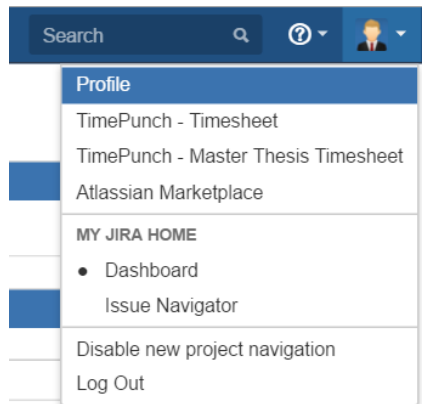


Figure 5.14: TimePunch - User Timesheet Selection

### User - Timesheet

Following actions can be performed by TimePunch - Users:

- Append entries to table
- Display timesheet entries
- Edit existing entry
- Delete existing entry
- Visualize personal timesheet data
- Import new entries from existing CSV, or GDTS
- Export timesheet data to CSV

Timesheet entries provide the following input options:

- Date – if the user clicks on the date insertion field a calendar view will pop up. The user is able to select the date he wants to insert into the field by clicking on it. This solution avoids wrong formatted user date inputs and also reduces the effort for creating a new entry. It is also possible to fill in the date manually. The date format is defined as YY-MM-DD. If a wrong date is given, it is replaced by the actual system date
- Inactive end date – enables a user to define an inactive period

## 5.6 User Interface

- Begin time – if the user clicks on this field a drop down list that allows to pick the start time is provided. It is also possible to insert the date using the keyboard. The required format of this field is pre-defined by the system as HH-MM. The default begin time is the actual time of the system and wrong user inputs are replaced with the current system date
- End time – this field is equal to the begin time field
- Pause time – the user is able to fill in his working pauses with this field. Per default 0 minutes are inserted and the format is similar to the begin time and end time field with HH-MM
- Team Select – if a user is assigned to than one team, a team selector is provided for each entry
- Category Select – users can select a category for their timesheet entry, depending on the assigned team categories
- JIRA ticket ID – this field is used for assigning Catrobat JIRA tickets directly to a timesheet entry
- Pair Programming select – users can append pair programming partners to their timesheet entry by clicking on this text field
- Description – the user has to insert a detailed description of his activity into this field


### User - Timesheet Data Input

The plugin provides input data checks and offers date pickers, as shown in Figure 5.15, team pickers, if a user is assigned to several teams and category pickers, as represented in Figure 5.16, to ease the input process.



## 5 Implementation

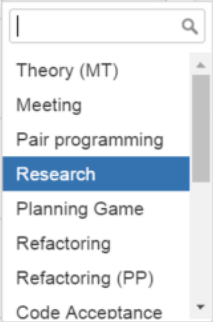
Date	Inactive End Date	Start - End
2016-04-23	yy-mm-dd	21:30 22:30



The image shows a date picker interface for the month of April 2016. The calendar grid displays days from Sunday to Saturday. The date 23 is highlighted in a dark blue box. Above the calendar, there are input fields for the date (2016-04-23), a format placeholder (yy-mm-dd), and time selection fields (21:30 and 22:30).

Figure 5.15: TimePunch - Timesheet Entry Date Picker

Team	Category	Pair Programming
Music Droid	Theory (MT)	



The image shows a category picker dropdown menu. The dropdown is open, showing a list of categories: Theory (MT), Meeting, Pair programming, Research (highlighted in blue), Planning Game, Refactoring, Refactoring (PP), and Code Acceptance. The background shows a table with 'Team' (Music Droid) and 'Pair Programming' columns.

Figure 5.16: TimePunch - Timesheet Entry Category Picker

### User - Timesheet Import

TimePunch offers users to import existing timesheet entries from their GDTS, as shown in Figure 5.17. The tool shows notification dialogues, if a

## 5.6 User Interface

performed action was successful, as shown in Figure 5.18. Users are also informed, if a performed action went wrong, as shown in Figure 5.19.

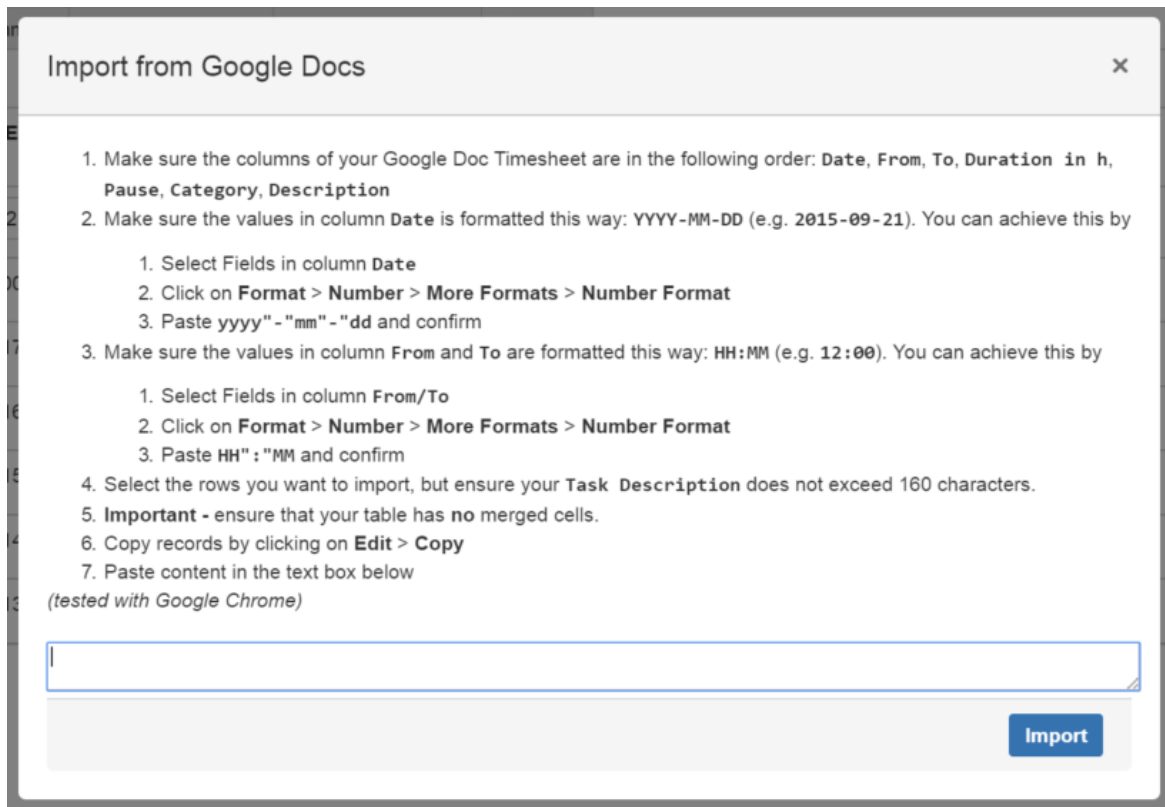


Figure 5.17: TimePunch - Timesheet Entry Import Dialog

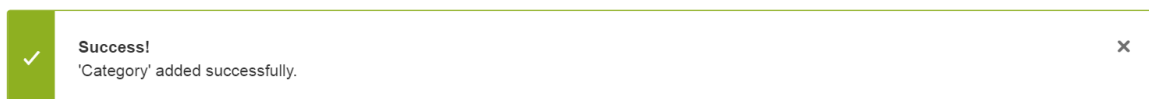


Figure 5.18: TimePunch - Success Dialog

## 5 Implementation



Figure 5.19: TimePunch - Failure Dialog

### User - Timesheet Visualization

The TimePunch add-on provides the following information-tabs:

- Timesheet - Shows the personal timesheet table
- Coordinators Private - if a user is coordinator/supervisor he can select other timesheets here
- Summary - Shows a table with the spent time each month
- Time - Visualization - Timesheet data visualization
- Team - Visualization - Team data visualization
- Key Data - Provides a lecture, ECTS and expenditure of time overview

### Summary

Figure 5.20, shows the summary overview of a TimePunch - User timesheet. This view represents how much time the user has spent each month, the total time the user spent, and the time the user spent within the last six month.

## 5.6 User Interface

Timesheet					
Timesheet	Coordinators Private	Summary	Time - Visualization	Team - Visualization	Key Data
Description			Value		
Time Spent: 2016-4			19hours 48mins		
Time Spent: 2016-3			210hours 9mins		
Time Spent: 2016-2			183hours 11mins		
Time Spent: 2016-1			132hours 14mins		
Time Spent: 2015-12			79hours 43mins		
Time Spent: 2015-11			26hours 29mins		
Time Spent: 2015-10			79hours 5mins		
Time Spent: 2015-9			43hours 44mins		
Time Spent: 2015-8			40hours 2mins		
Time Spent: 2015-7			6hours 4mins		
Total Spent Time			820hours 29mins		
Time / Month			82hours 2mins		
Overall Time Last 6 Month			558hours 42mins		



Figure 5.20: TimePunch - User Timesheet Summary

### Timesheet Data

The TimePunch implementation should not hinder users at their work, thus the add-on is designed as simple as possible. Figure 5.21 shows how TimePunch - Timesheets are visualized. This feature was not provided with GDTS. This sort of data representation enables users to notice their monthly expenditure of time immediately. This feature offers the possibility to define personal expenditure of time goals and a simple review mechanism.

## 5 Implementation

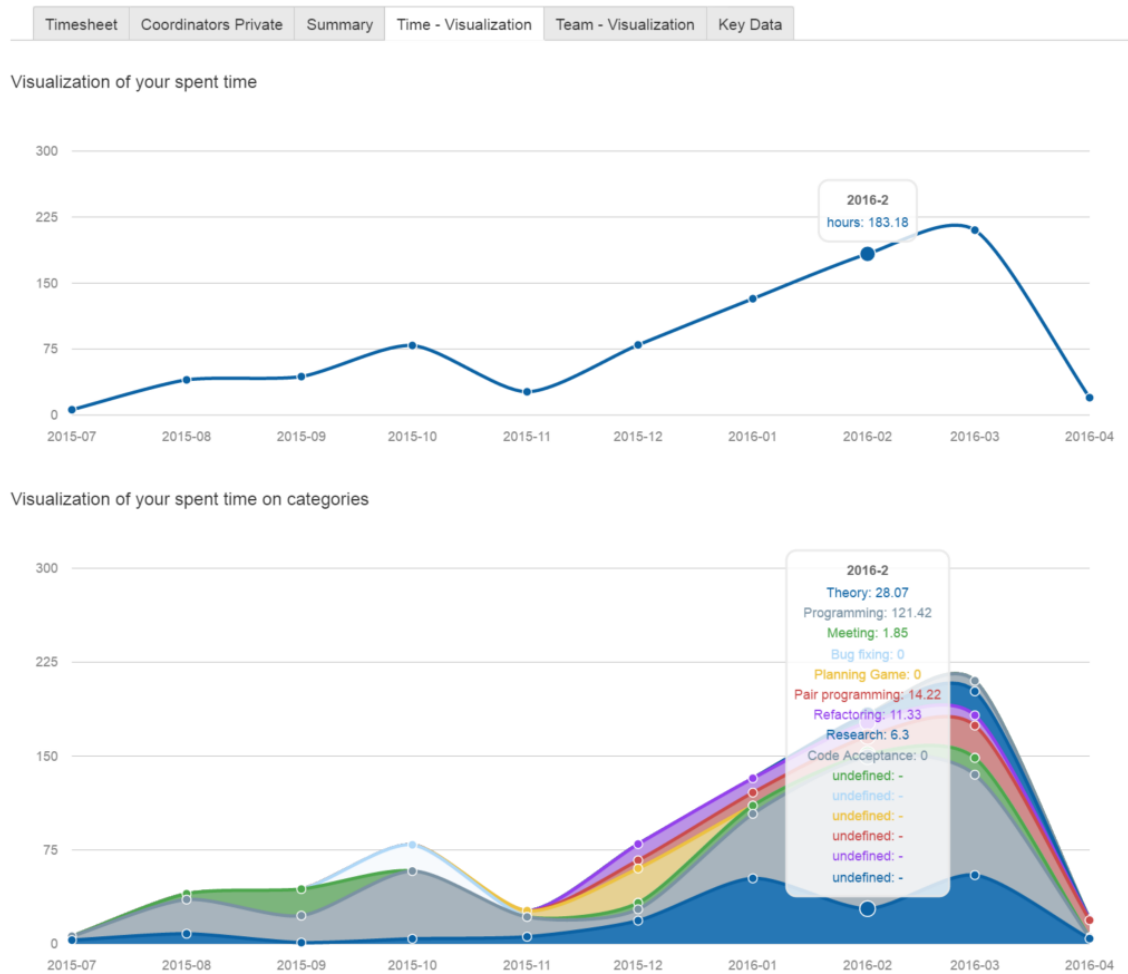


Figure 5.21: TimePunch - User Time Visualization

### Team Data

The add-on offers TimePunch - Users the possibility to display the spent time of their teams, as shown in Figure 5.22. This feature enables users to take a look at the overall team performance. Users are able to see the team performance and can examine if other members struggle - e.g. team has many hours for code reviewing, research or meetings

## 5.6 User Interface

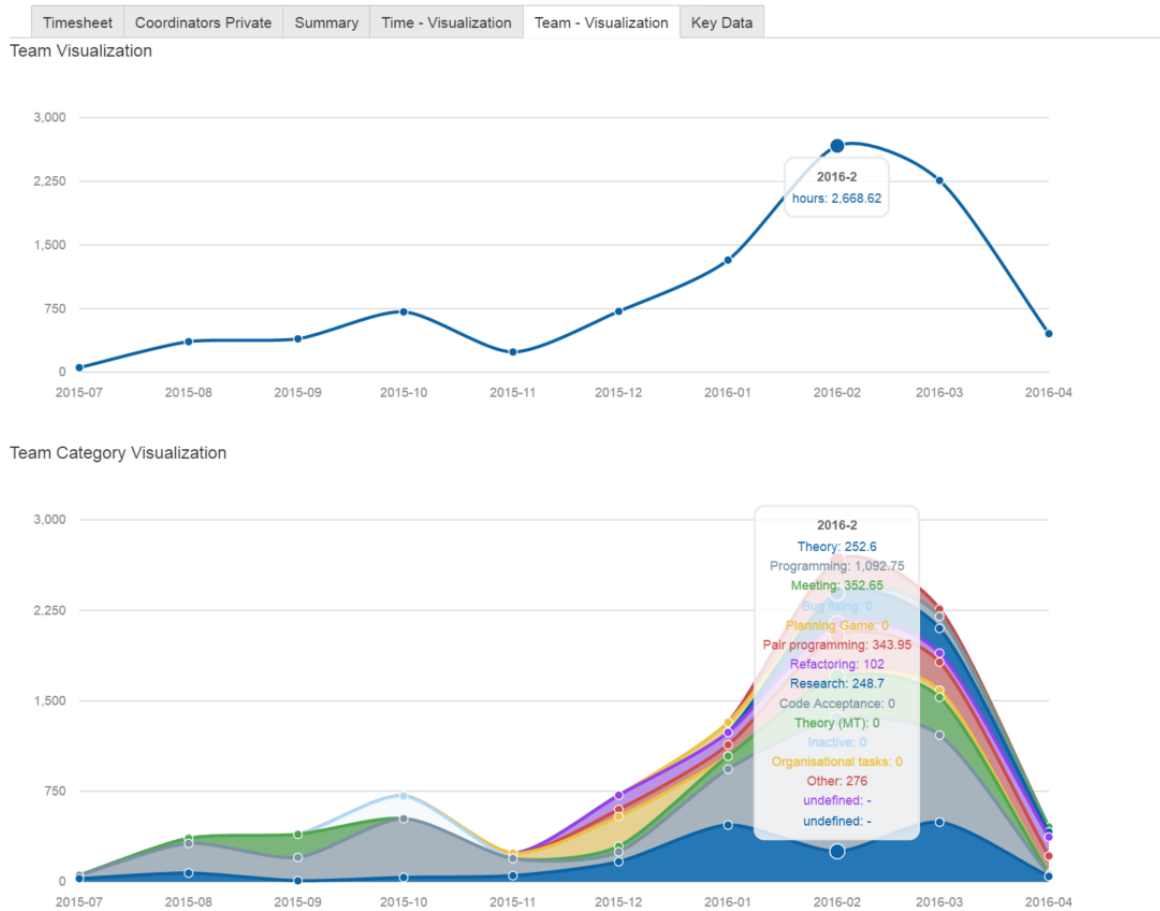


Figure 5.22: TimePunch - User Team Visualization

### Key Data

The TimePunch - Key Data tab provides information about the user's timesheet, as shown in Figure 5.23. Users can insert course names and course ECTS within this tab and get an overview about their current and remaining expenditure of time.

## 5 Implementation

Timesheet	Coordinators Private	Summary	Time - Visualization	Team - Visualization	Key Data
-----------	----------------------	---------	----------------------	----------------------	----------

### Export

Download 'Timesheet' as [CSV](#).

Remaining Timesheet Hours   
Shows your remaining timesheet hours.

Timesheet Hours   
Shows your timesheet hours.

Lectures   
Shows your lectures.

ECTS   
Shows your ECTS.

Figure 5.23: TimePunch - User Timesheet Key Data

## 5.7 Problems and Decisions

### Mozilla Firefox

At the early beginning of the plugin development a client-side data parsing bug was identified which only occurred using Mozilla Firefox<sup>10</sup>. It seems that the browser has issues with parsing JSON objects into strings, thus the *JSON.stringify()* method did not work. Further development and testing was accomplished using Google Chrome<sup>11</sup> and Opera<sup>12</sup>.

---

<sup>10</sup>Mozilla, 2016.

<sup>11</sup>Google, 2016.

<sup>12</sup>Software, 2016.

## 5.8 Results

As mentioned in Section 4.5, in-house plugin development carries some risks especially if the implementation is based on other software platforms. Thus if Atlassian applies any sort of changes to their software the TimePunch plugin has to be adopted. Students have to maintain the add-on in the future, otherwise its functionality is not guaranteed.

The latest TimePunch version includes all requirements from Section 3.2. Since there was not enough time for doing a test run with Catrobat team members, we can only refer to the direct feedback of the *Musicdroid* team members. The team did not see any previous implementation state of the add-on and their response was very positive. Since the GDTS problems are commonly known, by all Catrobat project teams, the expectations towards the TimePunch implementation and its provided features were very high. Users used their personal GDTS entries during the testing phase to ensure the import function and every tester was totally satisfied with the TimePunch add-on.



## 6 Summary and Future Work

This Chapter intends to give a summary and overview about the future work.

### 6.1 Summary

The defective, manual, annoying and slowly working GDTs was replaced through the TimePunch add-on. TimePunch improves data privacy, more data analysis functionalities and data visualization, compared to the previously used Google solution. The add-on enables course instructors to improve their teaching methods and lecture materials. Besides teachers can follow student's progress and respond quickly to problems. The TimePunch add-on enables students to manage their time on task in a comfortable way and instructors can contact students, if they do not show enough self-motivation. The add-on enables course instructors to verify if project groups work efficiently, related to their expenditure of time e.g. if teams spend much time on *Bug fixing*, *Meeting* or *Code Refactoring*, this might point to poor code quality or unexperienced teams.

The TimePunch implementation provides time tracking for students and reviewing options for course instructors. Students can watch their personal progress and how they spend their time. The add-on enables students to visualize the overall team progress.

## 6.2 Future Work

It was not possible to implement all of the defined requirements from Section 3.2, as the requirement list was extended several times during the development process. Future work could include the open requirements which are listed in Table 6.1. Additional functionalities could be:

- UX-Tests for the TimePunch user interface
- Implement user requirements derived from user feedback
- TimePunch version for mobile devices e.g.: Android application
- Combination with JIRA tickets

## 6 Summary and Future Work

Requirement	Description
Language Support	Support of multiple languages (e.g.: English & German)
Completion Prediction	User get a expenditure of time completion prediction related to their completed time each month
Team activity state	Coordinator can check the states of his team members
Viewing Permissions	Enable/Disable if coordinators are able to view whole timesheet data, or not
Extended Team Visualization	Supervisor, coordinator, administrator can select a team for visualization
Hard Reset	Empty all timesheet entries of a user timesheet
Additional Email Notification	Administrator can define a notification event on his own (e.g.: if a group is able to achieve something very fast)
Category Template	Administrator can define templates for categories e.g.: IT which consist other categories like Programming, Meeting, Refactoring
Export Group Data	Administrator can export timesheets of single teams, or groups
LDAP Team Group	Enable LDAP group selections for the team/coordinator fields
Deactivate Import	Administrator can enable/disable the Google Document Import of users
Version Control	Database history

Table 6.1: Requirements - Timesheet Future Work



# Appendix



The complete code of the **TimePunch - Implementation** can be found on **Github** - <https://github.com/Catrobat/TimePunch>





## Bibliography

- Abedi, M. and A. Beikverdi (2012). "Rise of Massive Open Online Courses." In: *Engineering Education (ICEED), 2012 4th International Congress on*, pp. 1–4. DOI: [10.1109/ICEED.2012.6779278](https://doi.org/10.1109/ICEED.2012.6779278) (cit. on pp. 2, 4, 17).
- Alhasani, H. M. A. (2015). "Design and precept of a flipped classroom style and its impact on traditional education." In: *Web Applications and Networking (WSWAN), 2015 2nd World Symposium on*, pp. 1–4. DOI: [10.1109/WSWAN.2015.7210350](https://doi.org/10.1109/WSWAN.2015.7210350) (cit. on p. 11).
- Android (2016a). *Android Developer Homepage*. visited on 2016-02-15. URL: <http://source.android.com/source/index.html> (cit. on p. 4).
- Android (2016b). *Android Homepage*. visited on 2016-02-13. URL: <https://www.android.com> (cit. on p. 26).
- Apple (2016). *iOS Homepage*. visited on 2016-02-15. URL: <http://www.apple.com/at/ios/whats-new/> (cit. on p. 5).
- Arduino (2016). *Arduino Bluetooth Board Homepage*. visited on 2016-02-13. URL: <https://www.arduino.cc/en/Main/ArduinoBoardBT?from=Main.ArduinoBoardBluetooth> (cit. on p. 27).
- Atlassian (2016a). *Confluence Homepage*. visited on 2016-02-15. URL: <https://www.atlassian.com/software/confluence> (cit. on p. 5).
- Atlassian (2016b). *Jira Homepage*. visited on 2016-02-15. URL: <https://www.atlassian.com/software/jira> (cit. on pp. 5, 30).
- Atlassian (2016c). *JIRA REST API Reference*. visited on 2016-04-18. URL: <https://docs.atlassian.com/jira/REST/latest/> (cit. on p. 61).
- Atlassian (2016d). *Maven*. visited on 2016-04-18. URL: <https://confluence.atlassian.com/display/CLOVER/Clover+for+IDEA+Installation+Guide> (cit. on p. 62).
- Bluetooth (2016). *Bluetooth Homepage*. visited on 2016-02-13. URL: <https://www.bluetooth.com/what-is-bluetooth-technology/bluetooth> (cit. on p. 27).

- Catrobat (2016a). *Catrobat Homepage*. visited on 2016-02-13. URL: <http://www.catrobat.org> (cit. on p. 26).
- Catrobat (2016b). *Pocket Code Homepage*. visited on 2016-02-13. URL: <https://share.catrob.at/pocketcode/> (cit. on p. 26).
- Chickering, A.W. and Z.F. Gamson (1989). "Seven principles for good practice in undergraduate education." In: *Biochemical Education* 17.3. cited By 15, pp. 140–141 (cit. on p. 14).
- Crump, S. (2013). "From radio, to satellite, to mlearning: Interactive distance education in Australia." In: cited By 0, pp. 217–220. URL: <https://www.scopus.com/inward/record.url?eid=2-s2.0-84944112031&partnerID=40&md5=b519b3e9531b59d924f4988a7362cc2e> (cit. on p. 13).
- Dang, Quoc-Viet and D.D. Gajski (2014). "Bringing in-class online - A hybrid solution." In: *Interdisciplinary Engineering Design Education Conference (IEDEC), 2014 4th*, pp. 12–17. DOI: [10.1109/IEDEC.2014.6784674](https://doi.org/10.1109/IEDEC.2014.6784674) (cit. on pp. x, 2, 4, 11–13, 16–19).
- Duan, H.a b et al. (2015). "Anticipatory processes under academic stress: An ERP study." In: *Brain and Cognition* 94. cited By 0, pp. 60–67. DOI: [10.1016/j.bandc.2015.01.002](https://doi.org/10.1016/j.bandc.2015.01.002) (cit. on p. 24).
- El-Ghoroury, N.H. et al. (2012). "Stress, coping, and barriers to wellness among psychology graduate students." In: *Training and Education in Professional Psychology* 6.2. cited By 0, pp. 122–134. DOI: [10.1037/a0028768](https://doi.org/10.1037/a0028768) (cit. on pp. 2, 24).
- Git (2016). *Git Homepage*. visited on 2016-02-15. URL: <https://git-scm.com> (cit. on p. 6).
- GitHub (2016). *GitHub Homepage*. visited on 2016-02-15. URL: <https://github.com> (cit. on p. 6).
- GNU (2016). *GNU Homepage*. visited on 2016-02-14. URL: <http://www.gnu.org/philosophy/free-sw.de.html> (cit. on p. 3).
- Google (2016). *Google Chrome*. visited on 2016-04-18. URL: <https://www.google.com/chrome/browser/desktop/> (cit. on p. 82).
- Goto, T. and T. Kusumi (2015). "The effects of regret on internalization of academic motivation: A longitudinal study." In: *Learning and Individual Differences* 37. cited By 0, pp. 241–248. DOI: [10.1016/j.lindif.2014.11.013](https://doi.org/10.1016/j.lindif.2014.11.013) (cit. on p. 2).
- He, M. and L. Wang (2009). "A New Model: Combination Technology-Based Distance Education and Traditional Teaching in Regular Undergraduate Program." In: *Education Technology and Computer Science, 2009. ETCS '09*.

## Bibliography

- First International Workshop on*. Vol. 2, pp. 80–86. DOI: [10.1109/ETCS.2009.280](https://doi.org/10.1109/ETCS.2009.280) (cit. on pp. 12, 13, 16).
- HELP.gv.at (2016). *Help.gv.at Homepage*. visited on 2016-02-14. URL: <https://www.help.gv.at/Portal.Node/hlpd/public/content/16/Seite.160120.html> (cit. on pp. 3, 24).
- Henno, J., H. Jaakkola, and J. Makela (2014). “From learning to e-learning to m-learning to c-learning to?” In: *Information and Communication Technology, Electronics and Microelectronics (MIPRO), 2014 37th International Convention on*, pp. 616–622. DOI: [10.1109/MIPRO.2014.6859641](https://doi.org/10.1109/MIPRO.2014.6859641) (cit. on p. 2).
- <http://ec.europa.eu> (2016). *Europa Homepage*. visited on 2016-02-14. URL: [http://ec.europa.eu/justice/data-protection/international-transfers/adequacy/index\\_en.html](http://ec.europa.eu/justice/data-protection/international-transfers/adequacy/index_en.html) (cit. on p. 3).
- <http://www.export.gov> (2016). *Export Homepage*. visited on 2016-02-14. URL: <http://www.export.gov/safeharbor/> (cit. on p. 3).
- i4ware-Software (2016a). *Timesheets For Confluence Plugin*. visited on 2016-02-15 Version 1.3. URL: <https://marketplace.atlassian.com/plugins/com.teknologiaplaneetta.confluence.grideditor.agile-application-stack/server/overview> (cit. on p. 55).
- i4ware-Software (2016b). *Timesheets For Confluence Plugin Documentation*. visited on 2016-02-20. URL: <http://confluence.i4ware.fi/display/i4ware/Timesheet%20for%20Confluence> (cit. on p. 55).
- IETF (2016). *IETF Homepage*. visited on 2016-02-15. URL: <https://tools.ietf.org/html/rfc4510> (cit. on p. 5).
- Initiative, Open Source (2016). *Open Source Initiative Homepage*. visited on 2016-05-06. URL: <https://opensource.org/licenses> (cit. on p. 3).
- Iqbal, S. et al. (2014). “On the impact of MOOCs on engineering education.” In: *MOOC, Innovation and Technology in Education (MITE), 2014 IEEE International Conference on*, pp. 101–104. DOI: [10.1109/MITE.2014.7020249](https://doi.org/10.1109/MITE.2014.7020249) (cit. on pp. 2, 17).
- IRC.at (2016). *IRC.at Homepage*. visited on 2016-05-07. URL: <http://www.irc.at/site/faq.php?lang=en> (cit. on p. 29).
- ISAS (2016). *ISAS Homepage*. visited on 2016-04-29. URL: <http://www.isasonline.org/international-students/what-is-the-usa-credit-system> (cit. on p. 24).
- Jayasinghe, U., A. Dharmaratne, and A. Atukorale (2015). “Students’ performance evaluation in online education system Vs traditional education

- system." In: *Remote Engineering and Virtual Instrumentation (REV), 2015 12th International Conference on*, pp. 131–135. DOI: [10.1109/REV.2015.7087277](https://doi.org/10.1109/REV.2015.7087277) (cit. on pp. 11–13, 16).
- Jenkins (2016). *Jenkins Homepage*. visited on 2016-02-15. URL: <https://wiki.jenkins-ci.org/display/JENKINS/Meet+Jenkins> (cit. on p. 5).
- Jetbrains (2016). *IntelliJ*. visited on 2016-04-18. URL: <https://www.jetbrains.com/idea/> (cit. on p. 62).
- Jianjun, Hou et al. (2014). "M-learning Course Development and Practice Based on Tutor-Directed Online Teaching." In: *Educational Innovation through Technology (EITT), 2014 International Conference of*, pp. 21–28. DOI: [10.1109/EITT.2014.12](https://doi.org/10.1109/EITT.2014.12) (cit. on p. 2).
- JIRA, Tempo for (2016a). *Tempo JIRA Plugin*. visited on 2016-02-15 Version 7.16.3. URL: <https://marketplace.atlassian.com/plugins/com.synergyapps.plugins.jira.timepo-timesheet-plugin/server/overview> (cit. on p. 50).
- JIRA, Tempo for (2016b). *Tempo JIRA Plugin Documentation*. visited on 2016-02-20. URL: <https://tempoplugin.jira.com/wiki/display/TEMPO/Tempo%20Timesheets%20Documentation> (cit. on p. 50).
- King Chan, ElsieSiu (2010). "Hybrid Learning: Teaching for Quality Learning at University." English. In: *Hybrid Learning*. Ed. by Philip Tsang et al. Vol. 6248. Lecture Notes in Computer Science. Springer Berlin Heidelberg, pp. 465–476. ISBN: 978-3-642-14656-5. DOI: [10.1007/978-3-642-14657-2\\_42](https://doi.org/10.1007/978-3-642-14657-2_42) (cit. on p. 2).
- LegoMindstorms (2016). *LegoMindstorms Homepage*. visited on 2016-02-13. URL: <http://www.lego.com/en-us/mindstorms/?domainredir=mindstorms.lego.com> (cit. on p. 27).
- Liu, Y. (2015). "The longitudinal relationship between Chinese high school students' academic stress and academic motivation." In: *Learning and Individual Differences* 38, pp. 123–126. ISSN: 1041-6080. DOI: <http://dx.doi.org/10.1016/j.lindif.2015.02.002> (cit. on pp. 23, 24).
- Ltd., ASynergy Apps (2016). *TimePO JIRA Plugin*. visited on 2016-02-15 Version 1.6.8.4. URL: <https://marketplace.atlassian.com/plugins/com.synergyapps.plugins.jira.timepo-timesheet-plugin/server/overview> (cit. on p. 45).
- M.G., Moore (1989). "Editorial: Three Types of Interaction." English. In: *American Journal of Distance Education* 3.2. cited By 522, pp. 1–7. ISSN: 08923647. DOI: [10.1080/08923648909526659](https://doi.org/10.1080/08923648909526659) (cit. on p. 14).

## Bibliography

- Microsoft (2016). *Windows Phone Homepage*. visited on 2016-02-15. URL: <http://www.windowsphone.com/de-at/how-to/wp8> (cit. on p. 5).
- morris.js (2016). *morris.js Github*. visited on 2016-05-13. URL: <http://morrisjs.github.io/morris.js/> (cit. on p. 62).
- Mozilla (2016). *Mozilla Firefox*. visited on 2016-04-18. URL: <https://www.mozilla.org/de/firefox/new/> (cit. on p. 82).
- Muslim, Arham et al. (2016). "A Rule-based Indicator Definition Tool for Personalized Learning Analytics." In: *Proceedings of the Sixth International Conference on Learning Analytics & Knowledge*. LAK '16. Edinburgh, United Kingdom: ACM, pp. 264–273. ISBN: 978-1-4503-4190-5. DOI: 10.1145/2883851.2883921 (cit. on pp. x, 20, 22).
- NFC (2016). *NFC Homepage*. visited on 2016-02-15. URL: [http://www.iso.org/iso/catalogue\\_detail.htm?csnumber=38578](http://www.iso.org/iso/catalogue_detail.htm?csnumber=38578) (cit. on p. 4).
- Oracle (2016). *The Java EE 6 Tutorial*. visited on 2016-04-18. URL: <https://docs.oracle.com/javaee/6/tutorial/doc/gijqy.html> (cit. on p. 61).
- Parrot (2016). *Parrot Homepage*. visited on 2016-02-13. URL: <http://www.parrot.com/de/> (cit. on p. 27).
- Phiro (2016). *Phiro Homepage*. visited on 2016-02-13. URL: <http://www.robotixedu.com/phiro.aspx> (cit. on p. 27).
- Reports, JIRA Timesheet and Gadgets (2016a). *1h JIRA Plugin Documentation*. visited on 2016-02-20. URL: <http://www.jiratimesheet.com/#/wiki/Overview> (cit. on p. 41).
- Reports, JIRA Timesheet and Gadgets (2016b). *1h JIRA Plugin Youtube Video*. visited on 2016-04-25. URL: <https://www.youtube.com/watch?v=ofocpgsf-AY> (cit. on p. 41).
- Saklofske, D.H.a et al. (2012). "Relationships of personality, affect, emotional intelligence and coping with student stress and academic success: Different patterns of association for stress and success." In: *Learning and Individual Differences* 22.2. cited By 28, pp. 251–257. DOI: 10.1016/j.lindif.2011.02.010 (cit. on pp. 2, 23).
- Scoop, Phone (2016). *Phone Scoop Homepage*. visited on 2016-02-15. URL: <http://www.phonescoop.com/glossary/term.php?gid=131> (cit. on p. 5).
- Scratch (2016). *Scratch Homepage*. visited on 2016-02-13. URL: <https://scratch.mit.edu> (cit. on p. 27).

- Shu-ping, Wang and Qiao Hai-ying (2013). "Innovations in the process of modern distance learning and teaching." In: *Computer Science Education (ICCSE), 2013 8th International Conference on*, pp. 1131–1134. DOI: [10.1109/ICCSE.2013.6554086](https://doi.org/10.1109/ICCSE.2013.6554086) (cit. on p. 14).
- Siamenchykau, Artsiom (2016). *TimePO JIRA Plugin Documentation*. visited on 2016-02-20. URL: <https://timepo.atlassian.net/wiki/display/TIMEPO/TimePO%20User%20Guide> (cit. on p. 45).
- Sieber, JoanE. (2005). "Misconceptions and realities about teaching online." English. In: *Science and Engineering Ethics* 11.3, pp. 329–340. ISSN: 1353-3452. DOI: [10.1007/s11948-005-0002-7](https://doi.org/10.1007/s11948-005-0002-7) (cit. on p. 13).
- Silva, E. de and E. Carys de Silva (2014). "Online teaching: A comparison of on-ground, online, facebook-linked teaching." In: *Interactive Mobile Communication Technologies and Learning (IMCL), 2014 International Conference on*, pp. 338–341. DOI: [10.1109/IMCTL.2014.7011159](https://doi.org/10.1109/IMCTL.2014.7011159) (cit. on p. 4).
- Simao, P., J. Fonseca, and V. Santos (2008). "Time attendance system with multistation and wireless communications." In: *2008 IEEE International Symposium on Consumer Electronics*, pp. 1–4. DOI: [10.1109/ISCE.2008.4559459](https://doi.org/10.1109/ISCE.2008.4559459) (cit. on p. 10).
- Slany, W. (2016). "Pocket Code: A Scratch-like integrated development environment for your phone." In: cited By 0, pp. 35–36. DOI: [10.1145/2660252.2664662](https://doi.org/10.1145/2660252.2664662) (cit. on p. 4).
- Smith, Jerry K. and Almond D. Dillard (2006). "Assessing and Conquering Burnout." In: *Proceedings of the 34th Annual ACM SIGUCCS Fall Conference: Expanding the Boundaries*. SIGUCCS '06. Edmonton, Alberta, Canada: ACM, pp. 374–377. ISBN: 1-59593-438-3. DOI: [10.1145/1181216.1181296](https://doi.org/10.1145/1181216.1181296) (cit. on p. 24).
- Soewito, B. et al. (2015). "Attendance system on Android smartphone." In: *Control, Electronics, Renewable Energy and Communications (ICCEREC), 2015 International Conference on*, pp. 208–211. DOI: [10.1109/ICCEREC.2015.7337046](https://doi.org/10.1109/ICCEREC.2015.7337046) (cit. on p. 10).
- Software, Opera (2016). *Opera*. visited on 2016-04-18. URL: <http://www.opera.com/de> (cit. on p. 82).
- Stuchlikova, L. and A. Kosa (2013). "Massive open online courses - Challenges and solutions in engineering education." In: *Emerging eLearning Technologies and Applications (ICETA), 2013 IEEE 11th International Conference on*, pp. 359–364. DOI: [10.1109/ICETA.2013.6674459](https://doi.org/10.1109/ICETA.2013.6674459) (cit. on pp. 2, 17).

## Bibliography

- Tan, Lennon, Minjuan Wang, and Jun Xiao (2010). "Best Practices in Teaching Online or Hybrid Courses: A Synthesis of Principles." English. In: *Hybrid Learning*. Ed. by Philip Tsang et al. Vol. 6248. Lecture Notes in Computer Science. Springer Berlin Heidelberg, pp. 117–126. ISBN: 978-3-642-14656-5. DOI: [10.1007/978-3-642-14657-2\\_12](https://doi.org/10.1007/978-3-642-14657-2_12) (cit. on pp. [x](#), [2](#), [12](#), [14–16](#), [24](#)).
- Taylor, J. (2002). "The use of principles for good practice in undergraduate distance education." MA thesis. Virginia Polytechnic Institute and State University (cit. on pp. [2](#), [14](#), [23](#)).
- Technopedia (2016). *Technopedia Homepage*. visited on 2016-05-08. URL: <https://www.techopedia.com/definition/5361/hypertext-transport-protocol-secure-https> (cit. on p. [62](#)).
- ubuntuusers (2016). *ubuntuusers Homepage*. visited on 2016-05-08. URL: <https://wiki.ubuntuusers.de/Cron/> (cit. on p. [70](#)).
- W3 (2016a). W3. visited on 2016-04-18. URL: <https://www.w3.org/Protocols/rfc2616/rfc2616-sec9.html> (cit. on p. [61](#)).
- W3 (2016b). W3. visited on 2016-04-18. URL: <https://www.w3.org/Protocols/rfc2616/rfc2616-sec10.html> (cit. on p. [61](#)).
- WiFi (2016). *WiFi Homepage*. visited on 2016-02-13. URL: <http://standards.ieee.org/about/get/802/802.11.html> (cit. on p. [27](#)).
- Zhdanov, Andriy (2016). *1h JIRA Plugin*. visited on 2016-02-15 Version 3.0.4. URL: <https://marketplace.atlassian.com/plugins/jira-timesheet-plugin/cloud/overview> (cit. on p. [40](#)).