

Liubov Zelenai

**INTERFACE DEVELOPMENT OF SCHEDULED TASK IN THE  
FORESTRY SYSTEM**

# **INTERFACE DEVELOPMENT OF SCHEDULED TASK IN THE FORESTRY SYSTEM**

Liubov Zeleniaia  
Bachelor's Thesis  
Autumn 2019  
Information Technology  
Oulu University of Applied Sciences

## ABSTRACT

Oulu University of Applied Sciences  
Degree Programme in Information Technology, Option of Internet Services

---

Author: Liubov Zelenaia

Title of the bachelor's thesis: Interface Development Of Scheduled Task In The Forestry System

Supervisor: Teemu Korpela

Term and year of completion: Autumn 2019

Number of pages: 39

---

The purpose of the thesis was to develop an interface of a scheduled task in the forestry system. The project was developed with the KMY registered association, which is the holder of regional PEFC group certificates in Finland. The Bitcomp Oy provided the API for the data fetch. This certification scheme has a huge collection of data containing information about the owners of forest areas, labels, and logos of users, and other information used for certification.

The aim of the developed task was to check the certificate property of the group of estates twice per year. The check was built as an automated scheduled task. The design of the interface and task were constructed in Visual Studio 2017 by the C# programming language, Visual Basic and .NET framework.

The result of the thesis project is the practical solution for the forestry system update of the Piimega Oy. The built interface and scheduled check improve the correctness and ensure relevant information about the estates of the company. The development can be proceeded anytime by the Piimega Oy development team in cooperation with the Bitcomp Oy and the KMY.

---

Keywords: PEFC, KMY, Bitcomp Oy, forest certificate, Visual Studio 2017, C# programming, Visual Basic, .NET framework

## **PREFACE**

The interface development was done at the Piimega Oy, a company working on software development for the forestry industry, during the trainee employment. The work was done with the help and guidance of the Piimega Oy's team leader, Jari Oksa.

I would like to give a great appreciation to Teemu Korpela, Jari Oksa and tutor Kaija Posio, for patient guidance and support during conducting the thesis. Moreover, I give gratitude to other teachers who have conducted studies of the Degree Programme in Information Technology at Oulu University of Applied Sciences for four years.

Oulu, 22.04.2019  
Liubov Zelenaiia

# CONTENTS

ABSTRACT	3
PREFACE	4
CONTENTS	5
VOCABULARY	7
1 INTRODUCTION	8
1.1 PEFC	9
1.2 KMY	13
1.3 Protection of the endangered species	15
2 DEVELOPMENT TOOLS	16
2.1 .NET Framework	16
2.1.1 User interface technology	17
2.1.2 Distributed systems technology	17
2.1.3 Programming languages	18
2.1.4 Microsoft SQL Server	18
2.2 Telerik	18
2.3 Software version control	19
3 CERTIFICATE DATA MANAGEMENT	22
3.1 Requirements	22
3.2 Scheduling	22
3.3 Service interface	23
3.4 OAuth 2.0	24
3.5 JSON	27
3.6 API	28
4 STORING AND USING DATA	30
4.1 JSON deserialization	30
4.2 Saving changes to the database	30
5 TESTING	31
5.1 Procedure	31
5.2 Error handling	31
6 ANALOG	33
7 CONCLUSION	36



## VOCABULARY

API	Application Programming Interface
CLR	Common Language Runtime
FCL	Framework Class Library
JSON	JavaScript Object Notation
KMY	Sustainable Forestry Association
MMC	Microsoft Management Console
PEFC	Programme for the Endorsement of Forest Certification Schemes (formerly the Pan-European Forest Certification scheme)
SQL	Structured Query Language
WCF	Windows Communication Foundation
XML	eXtensible Markup Language

# 1 INTRODUCTION

The policy of sustainable forest in Finland has a tradition and forests are managed due to principles of supportable forestry. The reason for such practices to be used is striking changes to the worse of the nature of forests in Finland. The protection of continuous supply of hardwood materials is the long-established aim of the policy, which has the priority in the management of forest in Finland equally with industrial, environmental, cultural and social aspects. Compared to Northern Finland, more than half of the forests in Southern Finland belong to individuals for non-industrial private purposes, which makes the implementation of commercial and environmental rules of forest conservation more difficult. Furthermore, the percentage of conserved forests in the north is almost ten times higher than in the south (1).

The widely accepted certification scheme was adopted by the Forest Industries Federation in Finland and is an efficient approach towards the authorized forest and log cutting. The practice of such system comprehensively boosts responsible forests and the usage of organic matter. One of the factors of quality and responsible management in corporations of forest business in recent years is the production of wood from forests which were certified. Forest certification is used to protect responsible forest management (2).

In essence, forest certification is the process of verifying forest compliance with the standard requirements. This is a simple concept, and certification is extensively used in all areas of production, providing independent confirmation of compliance with standards. Increasingly, forest processors, buyers, and managers are demanding for certification or buying certified products, therefore the interest in certification is growing. Regardless of the ease of the essential concept, forest certification becomes a controversial concern in the forest sector. The range of ways of using certification is expanding. Therefore, there are various aspects of certification such as (3):

- Forest and supply chain certification process



- Fundamentals of the theory of certification and design, operation of schemes
- The reality of existing schemes
- Political impacts
- Remaining problems and solutions

The increase in the diversity of ownership and management structures is caused by the development of forest standards. Examples of organizations created for developing and launching forest certification are The Forest Stewardship Council (FSC), The Programme for the Endorsement of Forest Certification Schemes (PEFC), The Sustainable Forestry Initiative (SFI) and Lembaga Ekolabel Indonesia (LEI). The dates of the establishment of the standards are presented in the Figure 1 (3).

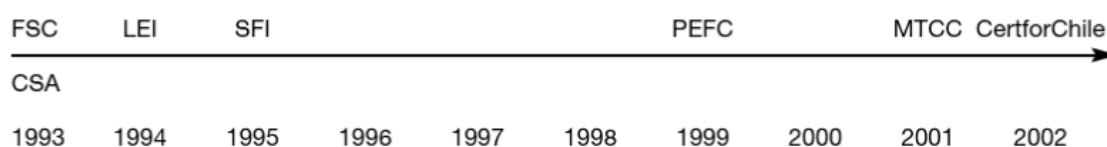


FIGURE 1. Dates of establishment

By using any of these approaches, the main object is the person who influences and controls the development and operation of the forest standard (3).

### 1.1 PEFC

In 1999 the Europe initiatives decided to set the Pan-European Forest Certification as mutual identification of national standards. Lately, at the end of 2003, the Pan-European Forest Certification expands its geographic area to a global framework to estimate and acknowledge forest certification standards. At present this scheme is used productively for mutual identification and was changed the name to the Programme for the Endorsement of Forest Certification (3).

The system can be developed to produce a related set of standards for any type of forest. The PEFC (Programme for the Endorsement of Forest Certification)

adopted the mutual recognition way. In this approach, the regional and national schemes are designed independently and then a system is defined for assessing the compatibility of standards. And the system used in this case for conducting mutual recognition assessments is very important. Requirements for compliance with the standard should be clearly defined (3).

The PEFC Council based in Luxembourg is a non-profit membership organization. There are two types of members: ordinary and extraordinary. The ordinary members are those, who are governing individuals maintaining forest certification in countries. The extraordinary members are international organizations which assist the PEFC council objectives. The general assembly is responsible for decision making by the majority of votes of national governing members and appoints directions of the organization (3).

The development process of the standard is a crucial matter because it is necessary to formulate decisions about how to handle incomplete data or contrary requirements. The final balance of the standard can be determined by the formation of the decision-making group as well as the method of decision-making. The content is equally important because it describes the requirements which will be carried out in a forest (3).

The PEFC Council has established that forest certification standards and certification schemes are developed in an open and transparent manner with the

participation of stakeholders. The organization of the PEFC council is presented in the Figure 2 (3).

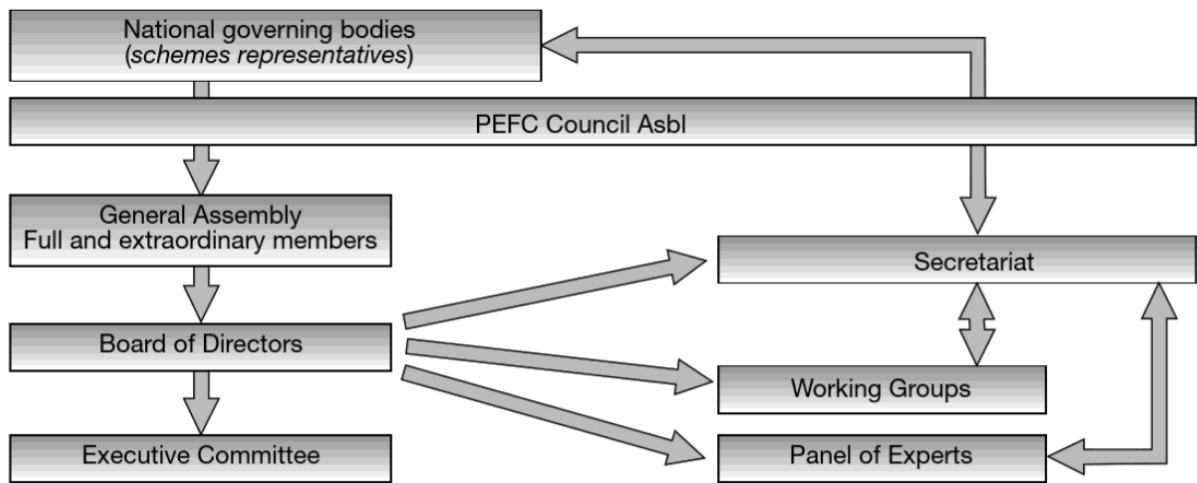


FIGURE 2. The organization of the PEFC council

The periodicity of reviewing the forest certification scheme should be at least once every five years in order to incorporate into the new experience and knowledge of the standards. The review process should be transparent and fair (3).

Traditionally, accreditation is granted by an existing national accreditation company in a particular country, and this approach is adopted by the PEFC scheme. Based on the general accreditation of systems, certification authorities are allowed to work up to four years. After that, the body will need special accreditation associated with forest production (3).

To ensure transparency, the provision of open information and consultation are important on the certification resolution. This contributes to increasing confidence in the scheme. As for PEFC, the consultation is included. However, there are no explicit requirements for information or consultation from the forest organization. The consultation is reviewed by the certification body. The only requirements for the information on certification are the name and location of the organizations (3).

Management and ownership of the world's forests are under the control of separate communities and individuals. Most of the time, these owners have a hectare of forest, which can be affected by authority and costs. However, the participation of small-scale forest owners and communities is very important. The reason for it

is that forests are extremely important for their livelihoods and because of a large impact on forests. PEFC is guaranteed to be available in the most economical way (3).

The rating of each standard can be measured including the number of forest certificates issued, the number of hectares certified, the global coverage of certified products or the number of issued supply certificates. The approximate number of certified hectares by schemes is presented in the Figure 3. The PEFC schemes may be considered the most used, however, there are no important parameters included (3).

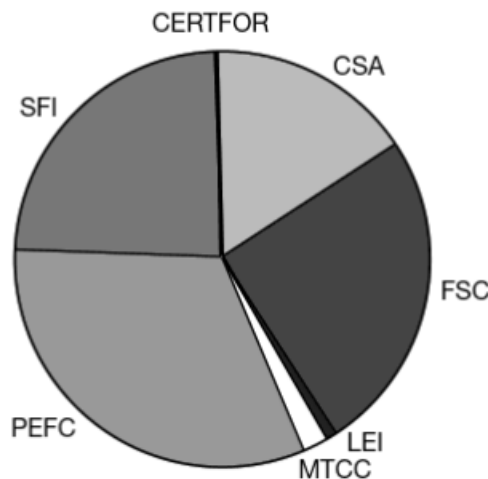


FIGURE 3. The relative number of a hectare of forest certified per scheme in December 2003

Forest certification was a very questionable issue and schemes are often questioned or disagree based on many findings in the report. Simultaneously, the PEFC standard is actively supported by small owners' Europe associations and many national governments of the industry. Expanding its capabilities to the global is also supported by many national schemes supporters that seek approval (4).

Main critics of certification generally believe that such a system has an insufficient impact to ensure forest maturation (4).

## 1.2 KMY

The Association for Sustainable Forestry (KMY) is the holder of the regional certificates of the PEFC group in Finland. KMY promotes sustainable forest management in Finland and develops forest certification (5).

The association maintains a register of participants in PEFC regional certification and provides information and advice on issues related to PEFC certification. The regional PEFC certification covers about two-thirds of the world's certified forests or about 260 million hectares. Based on this, about 85% of Finnish forests are PEFC certified (5).

Participation in PEFC certification is voluntary. Companies and entrepreneurs who provide services in PEFC forests must register for forest certification. There are two ways to register. Enterprises and entrepreneurs are registered to receive information as a member of the Forestry Workers Association or by registering with the Sustainable Forestry Association, which manages the regional certificates of the PEFC group. Registration covers the entire country, thus a registered business or an entrepreneur can offer their services to PEFC anywhere in the country (5).

The consent with PEFC certification terms is assessed annually in appraisal conducted by an external certification company. Forestry activities are assessed in each PEFC certification groups, both in the field and in offices. Forest owners, businesses and entrepreneurs are involved in regional certification. Such certification secures the diversity of the forest and maintains the social and economic sustainability of forestry (5).

PEFC Group Certification Areas in May of 2019 (5):

1. PEFC Southern Group Certification Area: Uusimaa, Kant-Häme, Päijät-Häme, Ostrobothnia, Aland Islands
2. Certification Area of the Western PEFC Group of Companies: Southwestern Finland, Pirkanmaa, Satakunta, Southern Ostrobothnia, Central Ostrobothnia

3. Southeast PEFC Group Certification: South Savo, South Karelia, Qimenlaakso
4. Eastern PEFC Group certification area: Central Finland, Northern Savo, North Karelia
5. Certification of the North PEFC Group: Northern Ostrobothnia, Kainuu, Lapland

The zones are presented in the Figure 4 (5).

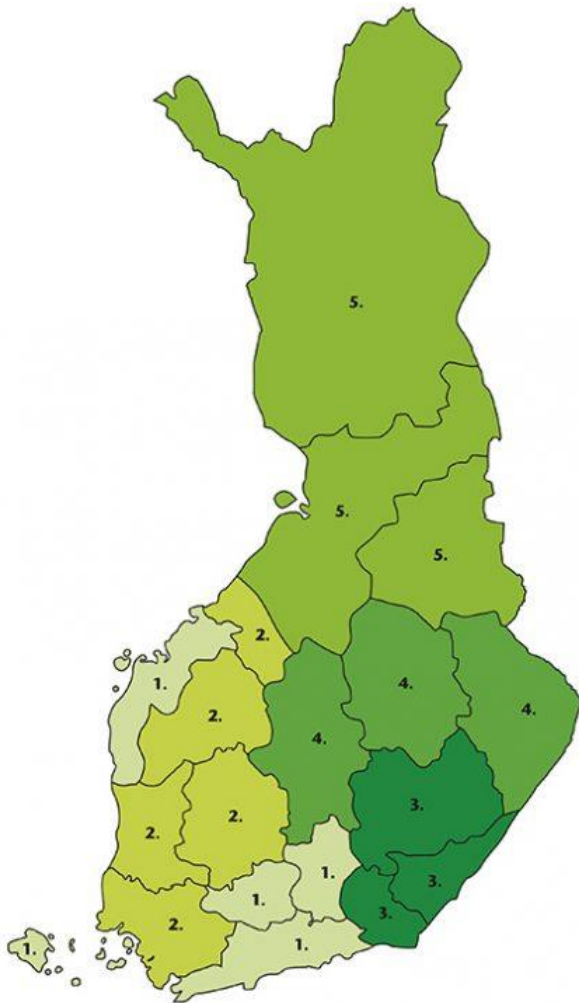


FIGURE 4. PEFC zones

### **1.3 Protection of the endangered species**

The criteria of the PEFC certification includes the protection of habitats of endangered species. In addition, it is necessary to protect habitats of other endangered species in accordance with the current model (5).

The protection project of endangered species in forestry activities was created in 2010 for forestry specialists and students as part of the protection of endangered forestry species. The project was developed by Tapio (now Tapio Oy) and the Finnish Environment Institute (SYKE) (5).

The purpose is to create a common set of rules for using the information and transmitting data on endangered species in forest use, nature management, and forest owners. The operational model includes guidelines and recommendations for material transfer and general procedures (5).

Information on the location of endangered species is stored in the Biological species section of the Hertta information system for environmental management. The system contains information on endangered species and plants, especially protected species, nature species, and birds directives, as well as some other species that are monitored in environmental management at both national and regional levels (5).

## 2 DEVELOPMENT TOOLS

### 2.1 .NET Framework

The .NET Framework features are available through a wide variety of managed types. These types are organized into hierarchical namespaces and packaged into a set of assemblies that build the .NET platform (6).

The platform consists of two parts. The basis is the Common Language Runtime (CLR) runtime environment, which can execute both regular programs and server applications. The second part is the Framework Class Library (FCL), which contains many components for working with databases, networks, input and output, files, and user interface. This allows the developer not to engage in low-level programming, but to use ready-made classes (6).

The components of the .NET framework are presented in the Figure 5 (16).

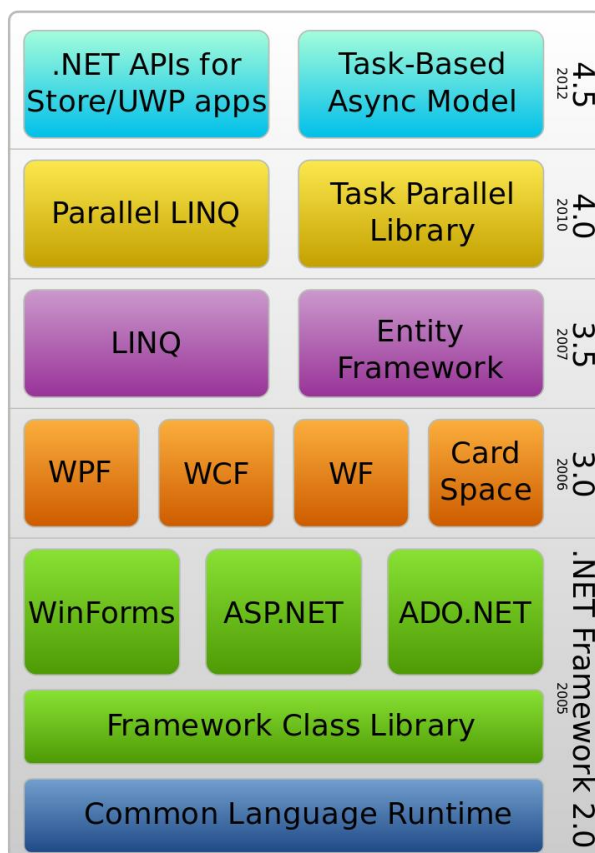


FIGURE 5. The .NET Framework components



### **2.1.1 User interface technology**

Windows Forms was used as a user interface technology for the thesis project. This is a relatively simple technology which offers features required for developing a Windows application. It is also important for supporting legacy applications. The Windows Forms infrastructure is relatively easy to learn and it is still widely supported in third-party controls (6).

### **2.1.2 Distributed systems technology**

Windows Communication Foundation (WCF) was used as a distributed system technology for the thesis project.

WCF is a complex communications infrastructure. It is flexible and fairly configurable. WCF implements the basic communication model described below between client and server applications (6).

1. On the server-side, the developer can specify which methods can be called by remote client applications
2. On the client-side, the developer can specify or output server-side method signatures, which should be called
3. On the server and client sides, the developer can choose the transport and communication protocol (in WCF this is done through binding).
4. On the client-side, the connection is established to the server.
5. The client calls a remote method which is transparently executed on the server.

WCF unleashes a client and a server even more through service contracts and data contracts. Instead of directly calling the remote method, the client sends the message to the endpoint service (6).

One of the advantages of this interchange is that clients have no dependencies on the .NET platform or on any proprietary communication protocols. The WCF infrastructure is highly configurable and offers the most extensive support for standardized messaging protocols. It allows for interacting with participants performing other software, perhaps on different platforms, and at the same time

support advanced tools as encryption. Another advantage of WCF is the ability to change protocols without the need to change other aspects of the client or server application (6).

### **2.1.3 Programming languages**

One of the basic ideas embodied in .NET is the compatibility of various parts of the application, which can be developed in different languages. For example, a program written in C# can access a method from a library written in Visual Basic .NET, or a class in Managed C ++ can be inherited from a class in Delphi .NET (6).

The Windows Forms part of the thesis project was developed in the C# programming language and the WCF part was developed in the Visual Basic language.

### **2.1.4 Microsoft SQL Server**

Microsoft SQL Server is a relational database management system (DBMS) developed by Microsoft. SQL (Structured Query Language) is a common database interface (7).

Microsoft and other companies produce a large number of software development tools which allow business applications to be developed using Microsoft SQL Server databases. Microsoft SQL Server also includes the Microsoft .NET Common Language Runtime (CLR), which allows implementing stored procedures and various functions for applications developed in .NET languages (7).

Piimega Oy uses Microsoft SQL Server to store and manage the data and the thesis project is connected to the company database to fetch and update the information.

## **2.2 Telerik**

Telerik is a developer of presentation-level and management tools for the Microsoft .NET platform. The flagship product of the company is the collection of Rad controls which provides organizations with all the components of .NET. Such components provide excellent automation of the development process with lower

project costs. Telerik's flexible project development, testing, and management tools enable companies of any size to create more stable software (8).

Telerik is used to create controls for the Windows Forms part of the application.

### **2.3 Software version control**

A version control tool has become critical for software developers, who usually spend time creating small changes in the application, some of which should be removed or checked the following day. It is extremely important to use control when a team is working simultaneously, and even on the same files, to prevent and manage potential chaos (9).

Piimega Oy uses Visual Studio for software development, and the TortoiseSVN was implemented as a version control system for projects. TortoiseSVN is a Windows Shell extension and a convenient Subversion client. It is free to use the software.

TortoiseSVN is a service with the tools to track edits and, if necessary, reanimate seemingly irretrievably deleted data. The principle of the application is to organize a special repository, which logs the history of all changes and creates the necessary copies (9).

The process is presented in the Figure 6 (9).

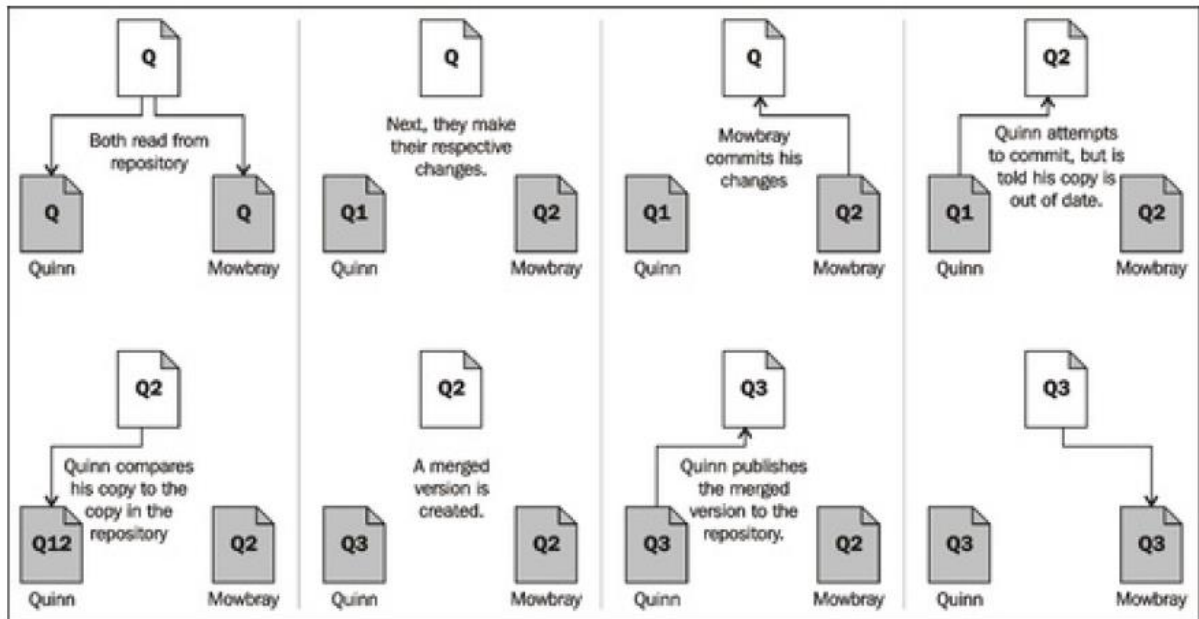


FIGURE 6. A diagram of the SVN process

Main characteristics of TortoiseSVN are as follows (10):

- organizing a centralized library of all changes;
- a convenient way to access Subversion commands;
- integrating into the context menu of the operating system;
- a unified method of information processing;
- versioning directories and metadata.

Since TortoiseSVN is actually an SVN client for accessing the Subversion repository, its main task is to organize a dialogue between the user and the database. After installing the program, the context menu of the Windows Explorer is supplemented with commands for performing the main tasks of Subversion - copying, editing, and merging. Using client tools it is possible to quickly create working copies of the necessary folders and files on the local computer (copy), create the necessary edits to the local copy (change) and merge them with the main document (merge) (9).

Advantages of TortoiseSVN are listed below (10):

- full implementation in Windows Explorer;
- access conflict notification;

- special functionality for files of binary formats;
- convenient access to reproduction.

Disadvantages of TortoiseSVN are the following (10):

- the service cannot guarantee the correct transmission of information from other users;
- methods of work may be incomprehensible to inexperienced users.

The application is popular among the developers of various software because it allows eliminating conflicts when opening the same parts of the project in progress.

## **3 CERTIFICATE DATA MANAGEMENT**

### **3.1 Requirements**

In 2016, the registration for the PEFC approval was set to be done on an individual basis. The properties are identified by a real estate number; therefore, the accuracy must be taken into consideration. The property code layout is a four-digit code (xxx-yyy-nnnn-zzzz) and can be found in extracts of property information (5).

The application allows checking properties validity to the PEFC certification with two developed ways:

1. Manually - with a button in the estate registers maintenance window of the Windows Forms for a single estate register check.
2. In a bulk – all estate registers from the SQL Server database are checked at once with the developed automated feature.

The estate register maintenance window has all the information about one particular estate. To check if the estate is PEFC certified, there is a button which calls the developed service of the WCF solution. In case of the bulk check, the application calls the service according to the schedule to check all estates registers. The service fetches the data of the selected estate or all estates from the endpoint (KMY) and the database update must arrange changes by writing the last update date.

The scheduled check is important because the KMY updates the information about estates twice a year, in the autumn and in the spring.

### **3.2 Scheduling**

The application of the .NET platform has the ability to execute the code on a schedule regularly. These tasks can be reminders, backups, or the creation of audit reports. There is no built-in solution for such problems, therefore such techniques need to be modeled. The only problem with the .NET is that the application can stop or restart work, due to which the task may not be completed or delayed.

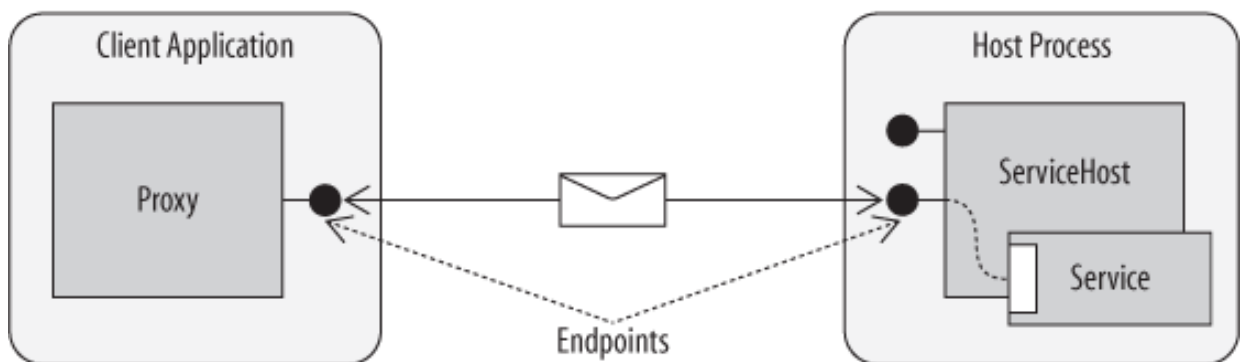
Therefore, application developers choose a more reliable solution, such as Windows services and task schedulers. (11)

The Windows Service has the ability to perform scheduled tasks while being installed on the server. Visual Studio has a service template which creates structures automatically and includes a class constructor, methods OnStart() and OnStop(). The service is a more reliable method of scheduling tasks because the task will be completed whereas the .NET application itself may restart due to files change or change in folders. Sometimes, such application turns off if there are no users (11).

There are two examples in which Timer.Elapsed and Thread.Sleep are used to create a task loop. This is useful when using the Windows Service. The Windows Service uses the OnStart method to initially create a loop with a thread or timer and to start the task. The OnStop method is used to stop the service and it is very useful for reporting the status of a task or errors. Class Timer is used to change the task schedule or stop the Windows Service (11).

### 3.3 Service interface

The service interface was developed in a WCF solution for a connection with the KMY data. The process of the calling endpoint is presented in the Figure 7 (12).



*FIGURE 7. ServiceHost exposes endpoints, and client proxies target a specific endpoint*

The project consists of the main files listed below:

1. IPEFCInterfaceService.svc.vb – a file with the service interface
2. PEFCInterfaceService.vb - a file with a class that implements the interface
3. IPEFC.vb – a file of business logic that creates client, calls the endpoint (KMY)
4. PEFC.vb – a file with the class that implements business logic
5. ResponsePEFC.vb – a class that stores the PEFC data retrieved from the endpoint

The Windows Forms solution has the WCF service reference to transfer fetched data to the Windows Forms solution and to use it afterward.

### **3.4 OAuth 2.0**

OAuth 2.0 is an authorization protocol which allows one service (application) the right to access user resources on another service. The protocol eliminates the need to trust the application username and password, and it also allows to issue a limited set of rights, but not all at once (13).

OAuth is an authorization protocol, which allows issuing rights to actions that a user can perform on behalf of the owner. At the same time the owner, after logging in, may not participate at all in the process of performing actions (13).

OAuth 2.0 is based on the use of basic web technologies: HTTP requests and redirects. Therefore, using OAuth is possible on any platform with access to the Internet and a browser: on websites, in mobile and desktop applications, browser plugins (13).

The general operation of an application using OAuth is:

1. obtaining authorization
2. access to protected resources

The result of the authorization is an access token — a key (usually only a set of characters), the presentation of which is a pass to protected resources. In the



simplest case, access to them occurs via HTTPS with an indication in the headers or as one of the parameters of the received access token (13).

The protocol describes several authorization options suitable for different situations:

1. authorization for server-side applications (most often, these are websites and web applications)
2. authorization for full client applications (mobile and desktop applications)
3. login and password authentication
4. restoring previous authorization

The authentication flow is presented in the Figure 8 (13).

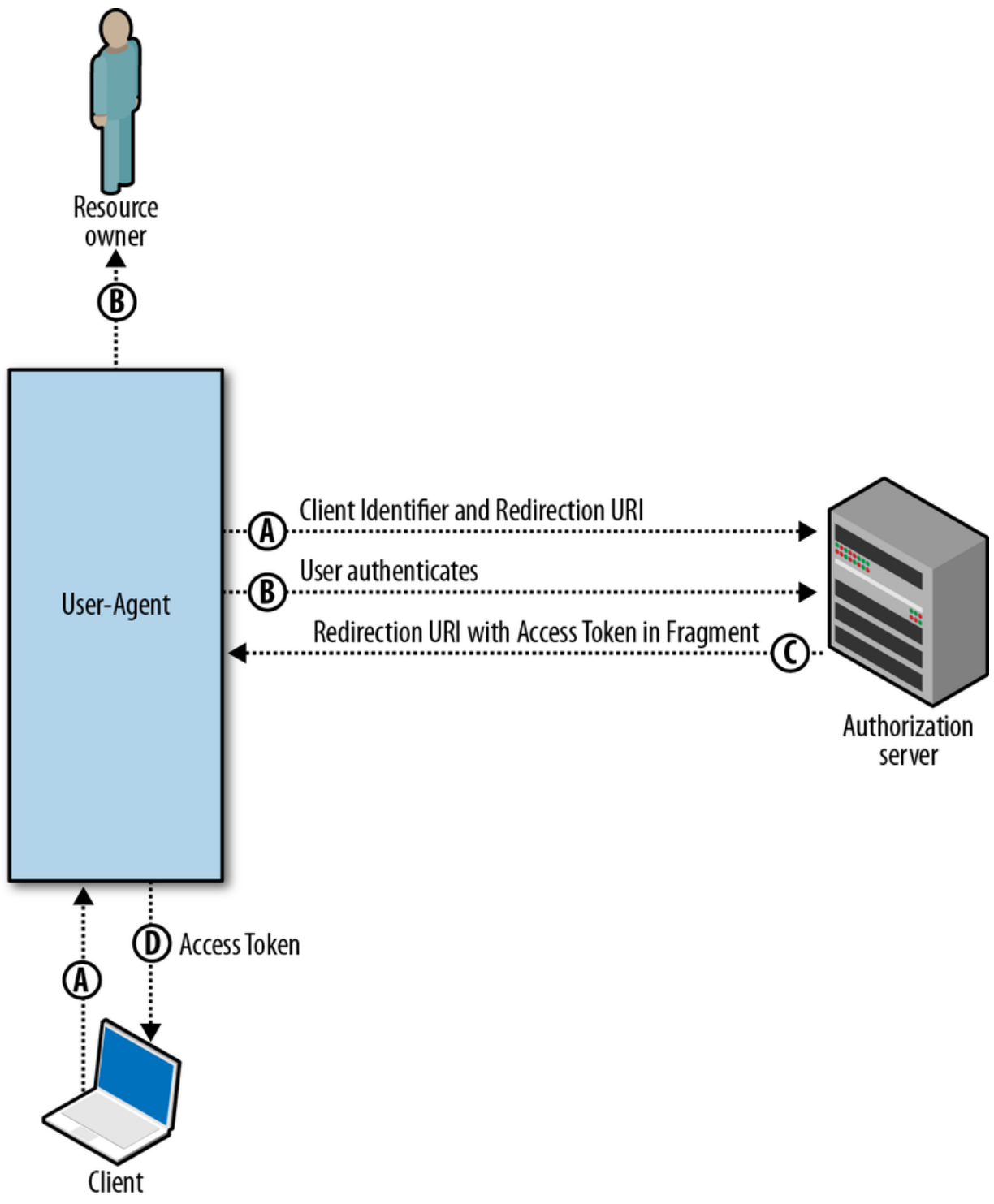


FIGURE 8. The Client-Side Web Applications flow: Step-by-step

This is the most difficult authorization option, but it only allows the service to unambiguously install the application requesting authorization (this happens during communication between the servers in the last step). In all other cases, authorization takes place entirely on the client and for evident reasons, it is possible to

disguise one application as another. This should be taken into account when implementing OAuth authentication in the services API.

### **3.5 JSON**

JSON is short for JavaScript Object Notation, a data transfer format. As the name implies, JSON comes from a JavaScript programming language, but it is available for use in many languages (14).

JSON itself uses the .json extension. When it is defined in other file formats, as .html, it appears in quotes as a JSON string or it can be an object assigned to a variable. This format is easy to transfer between the server and the client or browser (14).

JSON offers an effective alternative to XML due to easy to read and compact files, and it requires much less formatting. This informative guide will quickly help to understand the data that can be used with JSON and the basic structure with the syntax of the same format (14).

A JSON object is a data format with a key value, which is usually rendered in curly braces. When working with JSON, it is most likely to see JSON objects in the .json file, but they can also exist as a JSON object or string in the context of the program (14).

The format is basically set by two curly braces, which look as {}, and the data with key values is between them. Most of the data used in JSON is in JSON objects. Each pair of values is separated by a colon. Pairs of key values have a colon between themselves, such as here the “key”: “value”. The syntax diagram is presented in the Figure 9 (14).

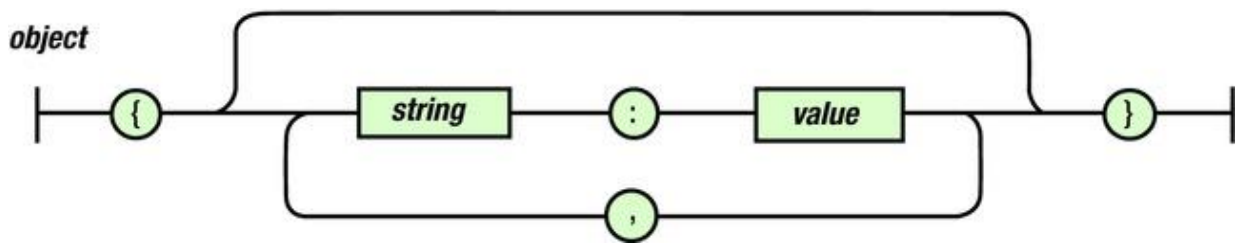


FIGURE 9. The syntax diagram of a string/value pair collection

### 3.6 API

An API (application programming interface) is a set of ready-made classes, functions, procedures, structures, and constants. All this information is provided by the application (or operating system) itself. At the same time, the user does not have to understand that this API technology ensures the interaction of the modules. The purpose of the information provided is the use of this data when interacting with external programs (15).

In general, this mechanism is used to combine the work of various applications into a single system. This is quite convenient for developers. Moreover, its internal mechanism does not matter - the developer may not even know about the API (15).

A sustainable forest economy company maintains forest estates register certification information in SERRE PEFC -certification registering the program. The company is maintained by BitComp Oy. The system contains a REST-service which has a public query interface for estates PEFC-group certifications validity check by estate registers number.

Credentials for using the service are obtained through the Forestry economics company. To use the interface, the user needs to identify themselves to the service as a registered user. This happens with OAuth 2.0. The valid time for 1 token is 24 hours. The validity time can be seen in the response messages variable; 'expires in' which is displayed as seconds. An example of the POST response as a JSON result is as follows:

```
{  
  
  "access_token": "...",  
  
  "token_type": "bearer",  
  
  "expires_in": 86399  
  
}
```

After the identification the user must add a header field and its value called 'access\_token' to the upcoming queries. The query interface contains a GET-method which takes the estates identification as a parameter and returns either a list of certified estates certificates with the HTTP-status code '200 OK' or a status '204 No Content' if the queried estate was not found or it did not belong to the regional PEFC group certification. An example of the GET response as a JSON result is shown below:

```
{  
  
  "identifier": "09241611123M001",  
  
  "identifierFormatted": "092-416-11-123-M001",  
  
  "regionId": 5,  
  
  "region": "Northern Ostrobothnia",  
  
  "certified": true  
  
}
```

## **4 STORING AND USING DATA**

### **4.1 JSON deserialization**

Serialization is the process of translating a data structure into a sequence of bits, or into another data structure that is convenient to store and transmit. Deserialization is the reverse process, the process of converting serialized data into a data structure.

After the WCF service has fetched the certificate data from the KMY endpoint in the JSON format, the data needs to be deserialized into the WCF service class. The class contains variables from the GET response. The Newtonsoft.Json library was used for the deserialization process.

The reference to a class is used in the Windows Forms solution, thus it is possible to check each parameter. Currently, the “certified” parameter is checked to discover whether the estate is PEFC certified or not.

### **4.2 Saving changes to the database**

The check was done either manually or as bulk automatically, and the information about the check date was written to the database, to the column “LastPefcCheck”. Also, if the certified property of the estate in the database, the column “Sertifiointi” did not match the same data in the KMY endpoint, the database update was done.

## **5 TESTING**

### **5.1 Procedure**

Code testing is done in several stages.

1. The developer
2. The supervising team of the Piimega Oy
3. Customers of Piimega Oy

The developer of the application should test how well the code works by debugging in Visual Studio 2017. Then they make changes for a better performance. Afterward, the supervising team checks the code along with the developer to give feedback, suggests some minor and major code rebuild. The approved project is used in the main Piimega Oy application and is distributed to the customers. These customers check if the interface works properly and give a notice when an improvement is needed.

### **5.2 Error handling**

The error control is made by the WCF solution and WCF error handling.

Exceptions that occur in a WCF service do not disrupt the operation of the host process, as well as the work of other running services and clients that are not related to these exceptions (12).

When an unhandled exception goes out of the scope of the service, the channel manager (the channel between the client's proxy and the service) intercepts and processes it, returning it in a serialized form in a SOAP message to the client. When the message reaches the intermediary (to the client's proxy), the latter initiates an exception on the client-side (12).

When accessing the service, the client may encounter three types of errors:

1. Communication errors. Occur, for example, when there is no connection to the network when the wrong address of the service is specified, or when

- the host process is not running. These exceptions are defined on the client-side by the CommunicationException exception class.
2. Channel status errors. Associated with the state of the channels created by the client's proxy with the service. Such an error could be, for example, an attempt to access an already closed proxy, which ends with the exception of the ObjectDisposedException class, or, for example, a mismatch between the contract and the level of security of the binding.
  3. Request Errors. Occur when the services work. The occurring CLR exceptions are not passed outside of services, they are converted to faults (faults) - Ex1 is neutral information for clients. To return to the client in a SOAP message, service failures are serialized.

The architecture is presented in the Figure 10 (12).

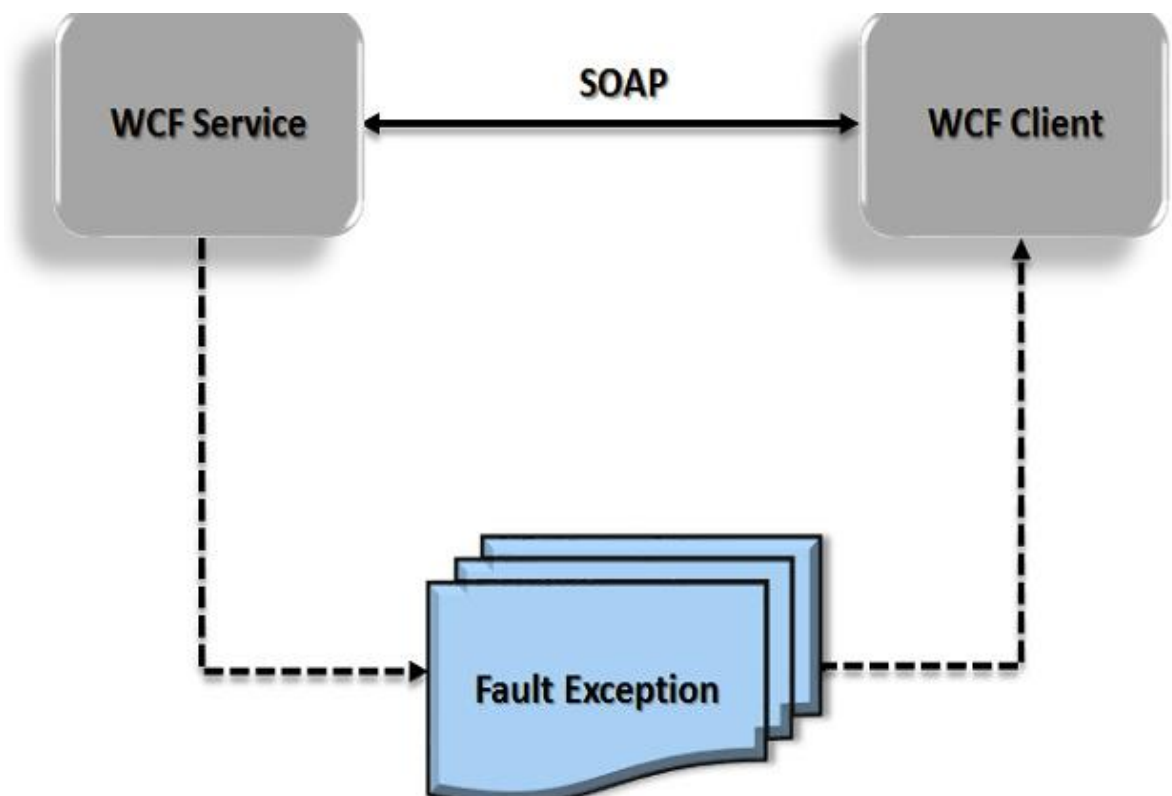


FIGURE 10. The exception handling in WCF



## 6 ANALOG

The analog example of the developed scheduled task is the Windows Task Scheduler. The administration often requires periodic maintenance tasks. Windows Task Scheduler is an MMC (Microsoft Management Console) tool for viewing and managing scheduled tasks (17).

Examples of tasks which are used to automate individual processes are as follows (17):

- creating recovery points automatically at specific times
- disk cleaning on specific days
- disk defragmentation at a certain time
- system diagnostic
- optimization of the computer

There are two main types of scheduled tasks in the Windows operating system (17):

1. The standard task is daily automatic tasks for performing system maintenance tasks. The user sees these tasks, and if necessary, can change them.
2. The hidden task for automating everyday system tasks. They are hidden from the user, and in most cases, it is not recommended to change them because the operation, maintenance, and security of the system depend on the performance of these tasks.

The thesis project was developed as a hidden scheduled task, which is run automatically two times a year.

The Windows Scheduler easily configures job execution scripts (17):

1. The task is performed only when the user is logged in, or vice versa, regardless of whether the user is logged in or not.
2. The task is performed on behalf of the user or with elevated administrative rights on behalf of the administrator.

Scheduled tasks have various components which are presented in the Figure 11 and include the following (17):

1. Triggers. The trigger sets the conditions for starting and completing various tasks. Tasks can be performed on schedule, when a user logs into the system or when the computer starts. It is possible to include events related to user actions in the task launch parameters. The use of event triggers greatly enhances process control capabilities.
2. Actions. The action task parameter defines the specifics of the running process. It allows a process to run programs, send emails, or display messages.
3. Principals. The principles monitor the safety of actions and determine which specific user is entitled to the task.
4. Settings. The settings describe conditions of the task management, such as the priority of the task or the state of the system in which the task is performed.
5. Registration Information. The information which is gathered when the task was created, such as the author of the task, the date, and the description.
6. Data. It is an additional information from the author of the task, which can be used by other users.

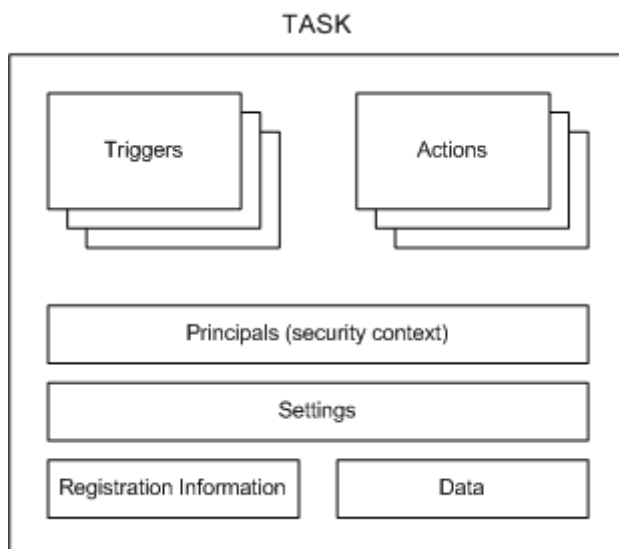


FIGURE 11. Task components

After the task is completed, the system can make changes, without notifying the user. In the same way, the thesis project scheduled task makes changes in the database and application without a notification.

## 7 CONCLUSION

Forest certification is the most important mechanism for forest conservation on the planet. Forests form the basis of life on Earth and are a source of wood, cellulose, food, and fuel.

Certification under the PEFC scheme provides for the assessment of forestry activities by an independent (third) party according to strict social, environmental and economic standards. Certified wood is tracked from the cutting area to the processing plant and then to the consumer. PEFC marking guarantees the consumer the legal origin of wood and products from it, as well as the fact that products come from responsibly managed forests.

PEFC certification includes ensuring the legality of timber production, the absence of irreparable damage to nature in their production, and no damage to the local population.

Sustainable development of forestry leads to satisfying the vital needs of the current generation of people without depriving such a possibility for future generations.

The aim of the thesis was to build an interface of the scheduled task for the forestry system at Piimega Oy. The company needed a system, which checks the company estate register certification twice a year and generates changes in the company database automatically. The changes in the project can be done anytime if a customer or a company will need further development or any problems solving.

The KMY was involved into the thesis project development, as the company owns the data about PEFC certificates. The API of the endpoint was done by the Bitcomp Oy. Thus, if this company will create any changes on the API side, the Piimega Oy will have to develop changes on the application as well for the proper working of the application.

During the thesis project development, I had to familiarize myself with the system of the forest certificates and also get acquainted with the .NET Framework, which is widely used for scheduled tasks by the company.

The work on the project went well during the whole process of the development. The company organized meetings with the company supervisor, the purpose of which was to check the progress, code and discuss further steps. The project was built according to Piimega Oy requirements, needs, and suggestions.

## REFERENCES

1. Vierikko, K., Pellikka, J., Hanski, K., I., Myllyviita, T., Niemelä, J., Vehkamäki, S., & Lindén, H. 2010. Indicators of sustainable forestry: The association between wildlife species and forest structure in Finland. Date of retrieval 27.04.2019 [https://www-sciencedirect-com.ezp.oamk.fi:2047/science/article/pii/S1470160X09001174?via%3Dihub](https://www.sciencedirect.com.ezp.oamk.fi:2047/science/article/pii/S1470160X09001174?via%3Dihub)
2. Forest certification promotes responsible forestry. Date of retrieval 27.04.2019 <https://www.forestindustries.fi/in-focus/environment-and-sustainability/forest-environment/forest-certification-promotes-responsible-forestry/>
3. Nussbaum, R., and Markku, S. 2004 The Forest Certification Handbook, Routledge. Date of retrieval 28.04.2019 <http://ebookcentral.proquest.com/lib/oamk-ebooks/detail.action?docID=430046>
4. Burton, P.J., et al. 2003. Towards Sustainable Management of the Boreal Forest, NRC Research Press. Date of retrieval 30.04.2019 <http://ebookcentral.proquest.com/lib/oamk-ebooks/detail.action?docID=228125>
5. KMY. Date of retrieval 28.04.2019 [www.kestavametsa.fi](http://www.kestavametsa.fi)
6. Lam, H. & Thai, T., L. 2001. .Net Framework Essentials. Date of retrieval 05.06.2019 <https://learning.oreilly.com/library/view/net-framework-essentials/0596001657/?ar>
7. Nielsen, P., & Uttam, P. 2009. Microsoft SQL Server 2008 Bible. Date of retrieval 08.06.2019 <https://ebookcentral-proquest-com.ezp.oamk.fi:2047/lib/oamk-ebooks/reader.action?docID=455849&query=microsoft%2Bsql%2Bserver>
8. Telerik UI. Date of retrieval 06.06.2019 <https://www.telerik.com/>
9. Tortoise SVN. Date of retrieval 06.06.2019 <https://tortoisesvn.net/>
10. Harrison, L., A. 2011. Tortoissvn 1.7 Beginner's Guide. Date of retrieval 24.05.2019 <https://ebookcentral-proquest-com.ezp.oamk.fi:2047/lib/oamk-ebooks/reader.action?docID=948574&query=tortoisesvn>

11. Cleary, S. 2014. Concurrency in C# Cookbook. Date of retrieval 24.05.2019 <https://www.oreilly.com/library/view/concurrency-in-c/9781491906675/ch12.html>
12. Bustamante, M., L. 2007. Learning WCF. Date of retrieval 08.06.2019 <https://learning.oreilly.com/library/view/learning-wcf/9780596101626/ch01s06.html>
13. Boyd, R. 2012. Getting Started with OAuth 2.0. Date of retrieval 08.06.2019 <https://learning.oreilly.com/library/view/getting-started-with/9781449317843/ch03.html>
14. Smith, B. 2015. Beginning JSON. Date of retrieval 08.06.2019 [https://learning.oreilly.com/library/view/beginning-json/9781484202029/9781484202036\\_Ch04.xhtml](https://learning.oreilly.com/library/view/beginning-json/9781484202029/9781484202036_Ch04.xhtml)
15. Kanjilal, J. 2013. ASP.NET Web API: Build RESTful web applications and services on the .NET framework. Date of retrieval 31.05.2019 <https://ebookcentral-proquest-com.ezp.oamk.fi:2047/lib/oamk-ebooks/reader.action?docID=1532007&query=API>
16. .NET Framework Wiki. Date of retrieval 31.05.2019 [https://en.wikipedia.org/wiki/.NET\\_Framework](https://en.wikipedia.org/wiki/.NET_Framework)
17. Windows Task Scheduler. Date of retrieval 22.07.2019 <https://docs.microsoft.com/en-us/windows/win32/taskschd/task-scheduler-start-page>