Design and implementation of a wood measurement system

Daniel Zheng
dzheng@kth.se
073 - 585 23 45
Computer Engineering (TIDAB), Study program for bachelors Autumn 08
SCS, KTH, ICT

II121X Exam work within information and software systems

Abstract

This report presents the development of a **computer system** for calculating the **volume of wood**, with the help of **air pressure**, inside a closed and pressurized chamber. The work has been performed with the company **Celite AB** who is related to **Woodair AB**, a previous company who failed to complete the same task. The report also presents some information about the history, the current state, the future and the benefits of the project.

According to the **ideal gas law**, in a closed system with a **constant** amount of **gas**, the change in the **pressure**, the **volume** and the **temperature** of the gas has a **proportional relationship** with each other. By **measuring** the **difference** in pressure and temperature from pressurizing an airtight chamber **with an object** inside and **without an object** inside, it should be possible to determine the change in volume for the closed system and then also the **volume of the object** inside.

The **computer system** is written as a website using **PHP** and **HTML**. The motivation for using these high simplicity languages is simply that more is not needed and also that there is more graphical work to be done by the system than other computational work. The website runs on an **Apache** web server and is connected to a **MySQL** database. The website is organized in the **MVC** pattern. The website shows a page where a user can manually enter measurement values and other arguments. These values are then used to **calculate** the volume of the object. After calculation all values are **stored** in the database with a timestamp and id. The computed calculations in the database can then be listed and viewed.

Unfortunately the mini test model was **not complete** by the time I wrote this report. So instead we tested only the computer system. A so called "**sanity test**" of the computer system. A "sanity test" means that we test that the computer system responds reasonably to the arguments we feed it.

In the results of the "sanity test" we saw that the system responded **as predicted** in accordance to the current theory.

In the future will we test the computer system on a real scale mini model.

Sammanfattning

Den här rapporten presenterar utvecklingen av ett **datorsystem**, för beräkningen a **volymen av trä**, med hjälp av **lufttryck**, inuti en stängd förseglad kammare. Arbetet har utförts hos bolaget **Celite AB**, som har mycket historia med **Woodair AB**, det tidigare företaget som misslyckades att utföra samma uppgift. Den här rapporten presenterar också lite information om historian, det nuvarande tillståndet, framtiden och fördelarna med projektet.

Enligt den **ideella gaslagen** så har förändringen i **tryck**, **volym** och **temperatur** i gasen en **proportionell relation** med varandra, i ett stängt system med en **konstant** mängd **gas**. Genom att **mäta förändring** i tryck och temperatur från att trycksätta en kammare **med ett objekt** inuti och **utan ett objekt** inuti, borde det vara möjligt att bestämma förändringen i volymen för det stängda systemet och sedan också **volymen på objektet** inuti.

Datorsystemet är skriven som en hemsida i **PHP** och **HTML**. Motivering för användningen av dessa väldigt enkla språk är helt enkelt att mer inte behövs och att det är mer grafiskt arbete som ska göras av datorsystemet än annat beräkningsarbete. Webbsidan körs på en **Apache** webbserver och är kopplad till en **MySQL** databas. Webbsidan är organiserad enligt **MVC** mönstret. Webbsidan visar en sida där en användare manuellt anger värdena för mätningarna och andra argument. Dessa värden används sedan för att **beräkna** volymen på objektet. Efter beräkningen **lagras** alla värden i databasen med en tidsstämpel och ett id. Den beräknade beräkningen i databasen kan då bli listad och sedd.

Olyckligtvis var minitestmodellen **inte klar** vid tiden som jag skrev denna rapport. Så istället testar vi bara datorsystemet, ett så kallat "**förnuftighetstest**" av datorsystemet. Ett "förnuftighetstest" betyder att vi testar att datorsystemet svarar resonligt till argumenten som vi matar in i den.

I resultat av "förnuftighetstestet" såg vi att datorsystemet svarade **som förutsagt** och enligt den nuvarande teorin.

I framtiden så kommer vi att testa datorsystemet på en riktig skalenlig minimodell.

Preface

This degree project is my last finishing touch on my 3 year long bachelor education in **computer software engineering** at KTH. The report is written for people with good English and some knowledge in computer technology.

List of people that helped the project:

- **Christian Schulte** my examiner. Thanks for being the cheery and funny teacher as always.
- **Richard Nordberg** my teacher in the language and writing support course. Thanks for being very informative and helpful at the subject.
- **Thomas Ekman** and **Per Elenborg** my supervisors and coworkers.

Per and Thomas was previously the project interest finders for **Woodair AB,** so they have some history with the previous company. After watching the project fail, Per and Thomas thought that someone had to resume the task and so they started the small company **Celite AB**. Per and Thomas is again the interest finders and also helps in building as well as acquiring materials and people for the project.

Thomas and me has discussed a lot about how facility should work. Thomas with some help of Per has together built a smaller and simple test facility and has been running tests on it while keeping me informed.

I was promised royalties if the project ever was completed depending on how much I contribute, which I think is rather motivating. This project has been very interesting and I hope to continue working on it in my spare time.

Table of Contents

1 Introduction

In this section the reader will be introduced to the project.

This introduction text as well as the pictures taken is based on. [1] "Affärserbjudanden att teckna aktier I Wood Air AB".

1.1 History

Wood has since ancient time been used for fuel and construction material, to build homes and make paper and since the industrial age, wood has become a very large international export commodity. Wood has many specialized usage areas, that require different levels of **quality** and so we must make **differences** between different types of wood. In the Swedish wood industry, wood is used to make high quality paper, carton board, packaging and a long list of varieties.

The **research** to find a solid and accurate way to measure the quality of the wood has been going on for decades. A quality measurement of wood can be done by measuring the **raw density**. To measure the **raw density** of wood all that is needed is to measure both the weight and volume. However wood is never perfectly shaped and this makes measuring the volume rather difficult.

Currently in Sweden the volume of wood is measured in an old fashioned method by **inspectors** who use their eyes and measurement stick. These people work for an independent organization called **Virkesmätarföreningen (VMF).**

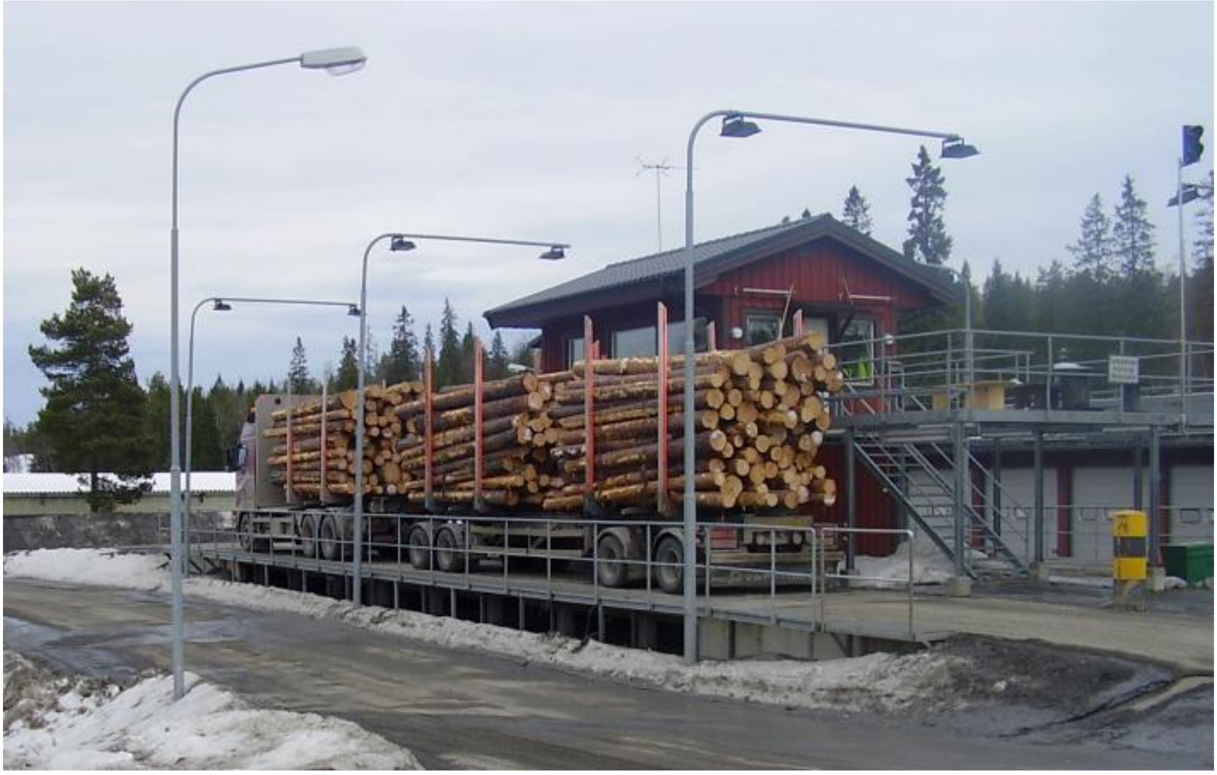

Figure 1.1. A traditional wood measurement station where traditional stick measurement is used.

1.2 Problem

A typical example would be a truck load of lumber logs. Before unloading, the truck has to stop by a measurement station. At this station the load is inspected by people from **VMF** and the volume of the load get measured (length, height, width). The precision of this way of measurement is not exact. At the same time the truck and the load is also weighed. Sometimes the **seller** complains about bad measurement and the truck load has to be inspected once more, which takes time.

The **quality** of the measurement differs, because of the **interference** from the seller and because there is a difference in **judgment** from inspector to inspector, so there is a large question mark if the numbers submitted for **statistics** are inaccurate. The **density** of wood is a measurement of quality, so it is important for both **seller** and **buyer** to get fair money for the correct quality of the wood.

There is also wood being sold as small **chips** stored in containers. The inspectors then judge the quality by inspecting the **top layer** of the chips. That means that the whole of the container load is being judged by the top layer quality, which is not totally accurate. **Cheating sellers** know that putting the best wood on top can give a higher quality rating and therefore more money. Cheating and measurement errors can cause a lot of arguments and confusion about the true value of the wood. This will in the end hurt **honest** buyers and sellers. Just the fact that there is an independent organization for this is proof that there is a high demand for an **impartial quality control**.

1.3 Solution

By using a large **air tight chamber** capable of fitting a truck load filled with logs, it is theorized that, when the air inside is **pressurized** there is a way to calculate the **volume of the wood** inside. Combine that with a weight measuring facility, it is hoped that the **quality measurement process** will become more **mechanized** and provide a higher accuracy, automation and speed. This could potentially increase the amount of wood **circuiting** and increase the **profit** for both sellers and buyers. This would make it harder for cheating sellers and better for honest people.

A typical example scenario would be a truck loaded full of wood. The truck would drive into the facility with the driver still inside. The facility would be sealed airtight and the air would be pressurized, but not so much that it can hurt the driver. A measurement would be taken and the facility would open up and the truck can continue its journey. The whole process should take about 4 minutes.

1.4 Previous attempt

Of course the idea of measuring volume using air pressure is not very new. A previous company, **Woodair AB** has built a first physical **test facility,** but unfortunately ran into technical difficulties. They ran into a debt of millions of Swedish kronor and therefore declared **bankrupt**. Therefore they **cannot complete** the task. But here is where **Celite AB** comes in, they simply want to complete the task. Allegedly the facility showed, before closing down, that the test-measurements were according the predictions. Apparently there already exist a previous fully functional software system but it will not be released until the engineers gets paid.

There exist a first test facility in **Husum** north of **Örnsköldsvik** in Sweden. The facility was complete but had technical difficulties that needed to be corrected. Unfortunately the company went into debt.

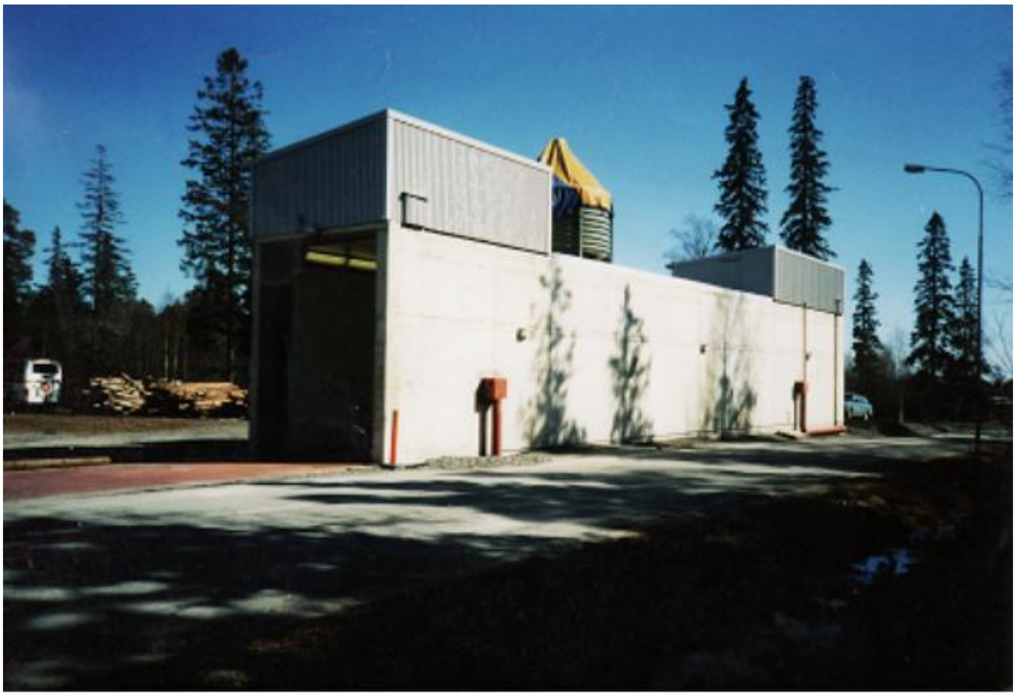

Figure 1.2. The first facility in Husum.

1.5 New attempt

Woodair AB who just went for building the **big scale facility** immediately of course ran into **technical difficulties** and then ran out of money. The thought was that **Celite AB** would finish making a fully functional, smaller, low cost, test model first. In other words **redo the project** correctly from the beginning and investigate the problem more thoroughly with a smaller start budget. **Woodair AB** did manage to complete a first test facility which is now **doing nothing** and it is built on borrowed land in the middle of **nowhere**. At the moment there is no plan for what to do with the facility and there is no point in tearing it down either, that would cost more than the ground it stands on. Here is where there is an amazing **opportunity** for a new company to just go in and buy the facility for almost nothing, then complete it and make lots of money.

1.6 Purpose

I will design and implement the **computer system** for the volume measurement facility. The computer system will have a mathematical algorithm that will be used to compute the volume of an object inside the facility from based on the readings from the air pressure and temperature sensors. The computer system will store and manage the calculated volume for statistical purposes. The computer system will either be manually entered or automatically fetched from the air pressure and temperature sensors. The computer system will have a graphical user interface that will make usage easy and fast.

2 Theory

In this section the theory of the project will be explained.

2.1 Mathematics theory

<u>Overview</u>

The **facility** have a large chamber that can seal in an object the size of a truck. On the top of the facility there is a large **cylinder with a piston** in it that can **compress the air** from the cylinder into the chamber and cause a high pressure inside the chamber. Se Figure 1.2 and 3.

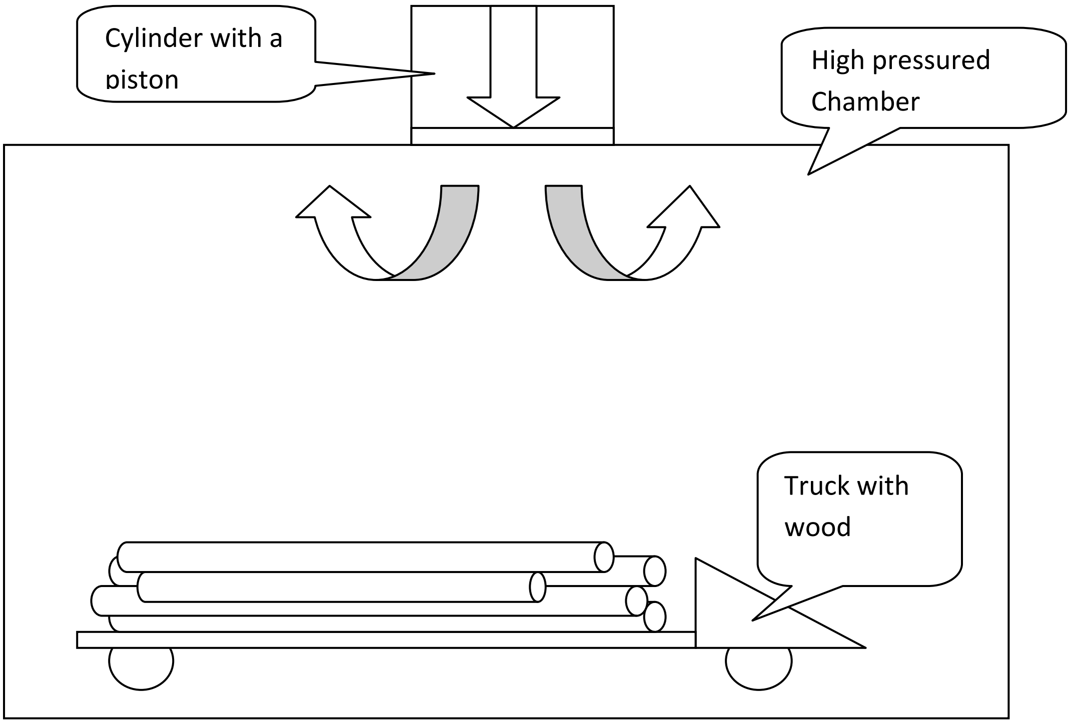

Figure 2.1. A model picture of the facility.

By **measuring** the **difference** in pressure and temperature from pressurizing an airtight chamber **with an object** inside and **without an object** inside it should be possible to determine the **volume of the object**.

Details

The **ideal gas law** is a physics law that describes the relationships between the pressure, volume and temperature of an ideal gas. All gases can be estimated to be ideal gases if the pressure is low enough and the temperature is high enough. This is because the intermolecular reactions will then have less effect.

The theory and effects from the ideal gas law is explained in this page:
[2] http://hyperphysics.phy-astr.gsu.edu/hbase/kinetic/idegas.html

This page from NASA is a more secure source:
[2.1] http://www.grc.nasa.gov/WWW/K-12/airplane/eqstat.html

The following list is an explanation of the variables:

- **VI**. The entire chambers maximum internal volume including the cylinder piston volume.
- **VC**. The cylinder piston volume, the volume of air that will be compressed into the chamber.
- **VO**. The volume of the object.
- **P**. The low pressure in the chamber.
- **V**. The volume of the gas inside the facility not counting the volume of the object.
- **T**. The temperature of the gas inside the chamber.
- **N**. The number of gas molecules.
- **k**. The Boltzmann's constant.
- **A.** The change in pressure of the gas at high pressure.
- **B.** The change in volume of the gas at high pressure.
- **C.** The change in temperature of the gas at high pressure.

According to the ideal gas law, the pressure, volume and temperature of a closed gas system is in the following relationship.

$$PV = NkT$$

Figure 2.2. The ideal gas law.

The volume of the gas inside the facility, is the total volume of the facility, excluding the volume of the unknown object with an unknown volume.

$$V = VI - VO$$

The changing volume of gas inside the facility has two different expressions. Where one is a subtraction of the cylinder volume and the other is a proportional change in the overall gas volume.

$$(V * B) = VI - VO - VC$$

$$(V * B) = (VI - VO) * B$$

When the closed gas system is at low pressure, the following formula applies:

$$PV = NkT$$

When the closed gas system has changed from low pressure to high the following formula applies:

$$(P * A)(V * B) = Nk(T * C)$$

From the low and high pressure systems we mathematically see that the base pressure, volume and temperature of both the systems can be reduced:

$$Nk = \frac{PV}{T} = \frac{(P * A)(V * B)}{T * C}$$

$$1 = \frac{A * B}{C}$$

$$A = C/B$$

$$B = C/A$$

$$C = A * B$$

We now see how the relationships between the change in pressure, volume and temperature fit to each other. An interesting things you can see form this relationship is that, in a closed gas system, when volume decreases the temperature decreases or/and the pressure increases. How much increase and decrease is however not predictable, but can be measured.

To find the volume of the object we start with the two different expressions for the change in gas volume.

$$0 = V * B - V * B$$

$$0 = (VI - VO - VC) - (VI - VO) * \left(\frac{C}{A}\right)$$

$$0 = (VI - VO)\left(1 - \frac{C}{A}\right) - VC$$

$$0 = (VI - VO) - \frac{VC}{1 - \frac{C}{A}}$$

$$VO = VI - \frac{VC}{1 - \frac{C}{A}}$$

Figure 2.3. The formula for the volume of the object.

As we can see the volume of the object depends on the internal volume of the facility, the cylinder piston volume, the change in pressure and the change in temperature. All of these dependencies can be measured. VI and VC could be measured using a simple measurement stick, C and A can be measured using special laboratory grade sensors build inside the facility chamber.

2.2 *System design theory*

The following text motivates the **design choices** I made for the computer system. The best general system design has a few properties that makes it better than other systems. The best system design is a system that meets the following **criteria's**:

- Has all the important functionality.
- Is easy and fast to develop. This reduce the time taken to develop the system and therefore the cost too.
- Is reliable and secure.
- Is maintainable.
- Has a parallelizable or scalable development process, so that the more people is working on the system the faster it will be completed.
- Is scalable, so that when more computing speed or storage is needed all that is needed is to add more hardware.
- Has room for future improvements, so that additional functionality can be added in the future.
- Is better than competing systems. Either the system is more cost efficient, faster or cheaper than other systems.

With the above list in mind I chose to develop the system as a **website** written in **PHP** (PHP Hypertext Preprocessor, the name is a recursive acronym) which is a dynamic scripting language for the server side of web browsers and **HTML** (HyperText Markup Language) a standard for displaying web pages.

The **website** runs on an **Apache** web server that is connected to a **MySQL** (My Structured Query Language) database. They are included in **XAMPP** (Cross Apache MySQL PHP Perl) a widely used web development package, that has preconfigured them to work with each other right after installation. Also included is **phpMyAdmin** which is a MySQL client written in PHP.

XAMPP and Apache is **open source**, meaning its free. MySQL is **free to try**, meaning it costs if it is used commercially.

The code syntax and functionality of the PHP language was found on:
[3] http://www.php.net/

Configuration support for the Apache server was found on:
[4] http://www.apache.org/

The terms for using the MySQL server was found on:
[5] http://www.mysql.com/

The installation download and configuration support was found on:
[6] http://www.apachefriends.org/en/xampp.html

The configuration support was found on:
[7] http://www.phpmyadmin.net/home_page/index.php

The following text motivates the choice of the design in every point of the above list of criteria's.

Functionality

The mathematics that will be used for **calculating** the volume of the object is simple **basic math** with floating point values, meaning its multiplication, division, addition and subtraction. PHP is very good at that. There is however a problem with **type safety** in PHP, so all values that are used must be checked to be of the **double** type. PHP uses the standard IEEE 754 double precision format. See **floating point precision** for other small details at their official website.

Every calculation needs to be **stored** in a database. PHP combined with MySQL is very good at this.

A **graphical user interface (GUI)** is needed to let the user interact with the computer system. Here is where HTML combined with PHP is great at. Graphical user interaction is exactly what websites are all about so developing the computer system as a website is by that thought a perfect idea.

Easy and Fast

I know by personal experience that these languages are smart and easy to work with and that there is a lot of online help available for them. The best part is that they are preconfigured to work each other, with an example website running, right after installation, using the XAMPP web development package.

Reliability and security

Apache and MySQL are both very stable and reliable software programs. If there is no **hardware failure** the computer system should be able to run day and night without slowdown.

Depending on if it is connected to the **internet** or not, the computer system could be read and manipulated by unauthorized people, so a security function will be needed for that. There is also the potential risk of a **denial of service attack**. Security problems are easily solved by simply unplugging the internet cord.

Maintainability

The computer system should be running fine as long as the database has storage space. If there is maybe 100 measurements per day then the database may need to be cleared maybe once every 20 years. A **backup** should be done every day in case of a **hardware failure** and this can be scheduled to run automatically. However the more entries there are in the database the longer it will take to do the backup so it would be smart to regularly **clear the database** every year or so.

Parallelizable development

The website is organized in the **MVC** (Model View Controller) pattern so a development team could parallelize the work in the model and the view, while keeping the controller as an interface. In this project however the only programmer is me.

The definition of MVC is found on:
[8] http://st-www.cs.illinois.edu/users/smarch/st-docs/mvc.html

Scalable

There should be one system running per facility. If more storage is needed another database server could be added. There is not much computations to be done, therefore adding more hardware will not improve computation time.

Future Improvements

When it comes to the graphical user interface there is plenty of room for future improvements. Simply add another view and connect it to the controller.

In the future there may be a need to for **automatic** fetching of the measurement values from the instruments. This requires hardware near programming that could be done using **C++**. PHP could then be used to **connect** it the C++ program, this is however not easy and rather tricky. That PHP is not a hardware near language is the one mayor flaw this system design has.

One of the key features of using a website as a solution is that in the future it is easy to connect to the **internet** so that any mobile or remote devices connected to the internet can be used to check up on the facility with ease. This of course saves travelling time for the maintenance personal and is also interesting for other people that has found an **interest** for the facility.

Competition

There is no competing system at the moment. Therefore this system wins when it is completed.

3 Computer system

In this section various things about the computer system will be explained.

3.1 Setup

The computer system consists of a database and a website running on a web server. The system was setup using the XAMPP package. The package included the three following:

- MySQL database
- Apache web-server that supports PHP
- phpMyAdmin a MySQL-client written in PHP.

3.1.1 Configuration

The programs are preconfigured to work with each other and there's also an example homepage from the start. After you have started the XAMPP-control panel you can see which services are running. They should all be green and running. You can see the default page by writing "http://localhost" or "http://127.0.0.1" in the web browser of the local system.

In the "C:\xampp\apache\conf" folder lies the "httpd.conf" file. Make a backup copy of it before you make changes in it. Open the file up in any simple text editor. In the file change the "DocumentRoot" to the project folder "C:\volmeasys". Also add directory access permission for the folder. It should look something like the following:

```
<Directory "C:\volmeasys">
   Options Indexes FollowSymLinks Includes ExecCGI
   AllowOverride All
   Order allow, deny
   Allow from all
 </Directory>
```

There is more security settings that can be done, but we will leave it unrestrictive for the moment.

3.2 Architecture

In this section the design of the computer system will be explained.

3.2.1 UML

The following **UML** (Unified Modeling language) diagram describe the **structure** of the computer system. The **arrows** indicate a direct **dependency**. The **name** of the computer system is **VOLMEASYS**, which is a contraction of the words; volume, measurement, system. Notice how the MySQL server and phpMyAdmin are **outside** of the website container, this is to indicate that I did not design them but that the computer system are using them.

The definition of UML is found on:
[9] http://www.omg.org/gettingstarted/what_is_uml.htm

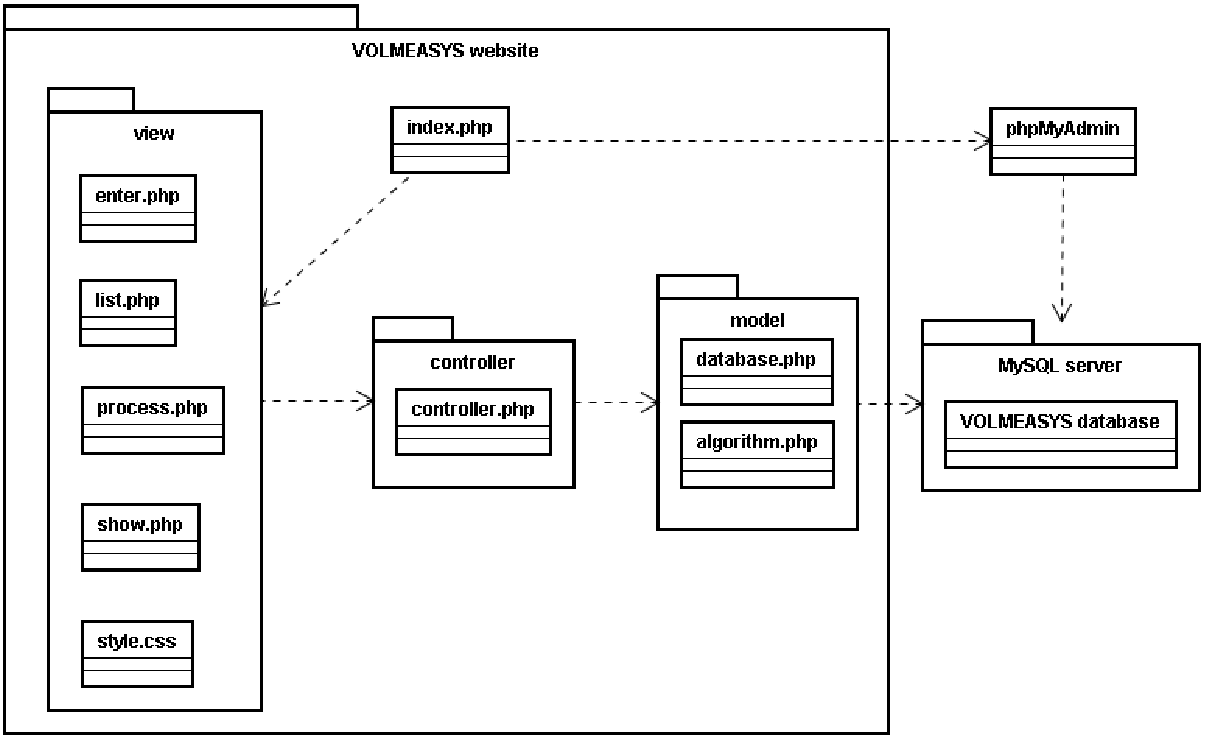

Figure 3.1. An UML diagram describing the structure of the computer system.

3.2.2 Database

The database and the tables should be generated automatically once the system starts for the first time.

The MySQL database require a database with the name "volmeasys_database" and in the database there should be two tables with the names; "measurement" that stores every individual measurement taken and "calculation" that stores every estimation of an object volume, pointers to which measurements where used in the calculation and the values of VI and VC.

The "measurement" table consist of the following columns; id, pressure, temperature.

The "calculation" table consist of the following columns; id, time, VO, VI, VC, M0, M1, M2, M3.

3.2.3 Website

The website consists of several PHP scripts that is structured up according to the MVC-pattern (Model View Controller). Every part in view must go through the controller to communicate with the model. This pattern good to use for future changes to the architecture. The website is written in object oriented PHP however all the methods are static. This is done to avoid name collisions.

In the **root** folder lie the model, view and controller folders. In the root folder lie the "index.php" file. The "index.php" file is a simple page with a few links pointing to all of the pages in the view folder and also links to the phpMyAdmin page.

In the model folder there is the following files:

- "algorithm.php". It has methods to calculate the volume of the object.
- "database.php". It has methods to communicate with the database. It can store a calculation, fetch the last calculation, list the most recent calculations and to fetch a specific calculation.

In the controller folder there is the following files:

- "controller.php". It has several methods that allow fetching results from the model. In all of the methods it checks whether the arguments are valid and throws exceptions if they are not.

In the view folder there is the following files:

- "enter.php". It is a page that has an input form this is where you can manually enter the measurements, VI and VC arguments. See mathematical theory section 2.1 . These arguments are submitted to the process.php file.
- "process.php". It is a dynamic page that communicates with the controller. It first uses the arguments received to calculate the volume of the object. Then stores the calculation in the database. Then displays a page showing all the arguments and the volume of the object, the time it was calculated and the id of the calculation. If there was an exception in calculating or storing it will show a stack trace of the problem.
- "list.php". It is a dynamic page that communicates with the controller. It fetches a list of the most recent calculations and shows them. Users can also click on them and be redirected to the more detailed view.php page with the calculation id as an get argument.
- "view.php". It is a dynamic page that views the details of a calculation. If no id is specified in the arguments it will show the most recent calculation.
- "style.css". It is a style that is applied to all the pages to improve its looks.

3.3 User guide

This section describes how to use the computer system once it is running.

After installing and configuring the computer system on a machine. Browse into **http://localhost** or the IP address for the website using any browser you want to. The index page should be showing a few links.

Click the **enter.php** link to view the enter page, where you may enter measurement values and submit them for calculation. If the calculation was successful, you will be redirected to the **process.php** page and here you will be shown the calculated **volume of the object** as well as the data the system used for the calculation. The calculation is automatically **stored** in the database. You will also see the time and id for the calculation in the database.

Click the list.php link to view a list of previous calculations.

Click the phpMyAdmin link to view, manage or erase the database using the powerful database client.

4 Test

We ran into difficulties with the physical mini test model. The main problem was pressure leakage. So unfortunately the mini test model was not complete by the time I wrote this report. So instead I will test only the computer system.

We will perform a so called "sanity test" of the computer system. A sanity test means that we will test that it responds reasonably to the arguments we feed it. To do that we first must investigate what is reasonable and what is unreasonable. In the mathematical theory section 2.1 we see that the end formula for the volume of the object is according to figure 2.3. Looking at the end formula the following **list of rules** are certain:

0. VI must be greater than zero. Otherwise we would not have a facility.
1. VC must be greater than zero. Otherwise no gas would be compressed.
2. VI-VC must be greater than zero. VC must fit inside the facility and leave room for the volume of the object.
3. VO must be between 0 and VI-VC. Because no object can have a negative volume or be greater in size than what the facility can contain. The empty facility can however have an object with a volume of zero.
4. A must be greater than 0. Because the gas pressure cannot change into nothing, if it does the result will be undefined.
5. C must be greater than 0. Because the gas temperature cannot change into nothing, if it does the result will be undefined.
6. B or its equivalent C/A must be less than 1.0 but greater than 0. Because the volume of the system is always compressed from VI to VI-VC.

If the arguments entered does not compute according to the above statements then the computer system should respond with an appropriate error message. We will test each statement above and focus on the limits.

5 Result

The following table is a table of the test results. The test was performed by feeding the computer system with arguments changing only one of the arguments at a time. The column "Expected" shows what response was to be expected. The columns VI, VC, Pressure and Temperature was the values that was input into the computer system. The column "Response" shows what the computer system responded.

nr	Expected	VI	VC	A	C	Response
0	exception VI	0	50	1.06	1	exception VI
1	exception VC	1000	0	1.06	1	exception VC
2	exception VI-VC	1000	1000	1.06	1	exception VI-VC
3	exception A	1000	50	0	1	exception A
4	exception C	1000	50	1	0	exception C
5	exception B	1000	50	1	1	exception B
6	close to 0	1000	50	1.05263157894736	1	0.000018999997678293
7	close to 1	1000	50	1.0526870389884	1	1.0000182207987
8	close to 500	1000	50	1.1111111111111	1	499.9999955
9	exact 949	1000	50	51	1	949
10	close to 950	1000	50	9999	1	949.9949989998
11	close to 0	1000	50	1	0.95	9.0949470177293E-13
12	close to 1	1000	50	1	0.94995	0.99900099900174
13	exact 500	1000	50	1	0.9	500
14	close to 949	1000	50	1	0.0196078	949.000002244
15	close to 950	1000	50	1	0.0000000001	949.999999995
16	exception VO	1000	50	1.05	1	exception VO

Figure 5.1. A table of the test results.

The following list is an explanation of each test number. Test 0 to 5 and 16 tests the rules from the **list of rules** in the previous section Test section 4. Test 6 to 15 tests the ranges and middle response values of VO. A rounding error occurs when the result is a mathematical fraction.

0. We test rule number 0. VI is set to 0 which is not legal according to the rule and therefore an exception VI was expected and an exception VI was responded.
1. We test rule number 1. VC is set to 0 which is not legal according to the rule and therefore an exception VC was expected and an exception VC was responded.
2. We test rule number 2. VI minus VC is equals 0 which is not legal according to the rule and therefore an exception VI-VC was excepted and an exception VI-VC was responded.
3. We test rule number 4. A is set to 0 which is not legal according to the rule and therefore an exception A was expected and an exception A was responded.
4. We test rule number 5. C is set to 0 which is not legal according to the rule and therefore an exception C was expected and an exception C was responded.
5. We test rule number 6. B equals 1 which is not legal according to the rule and therefore an exception B was expected and an exception B was responded.

6. We test the lowest range for VO changing only the A argument, with VI set to 1000 and VC set to 50. Because of rounding errors the lowest value of VO was expected to be a value close to 0, which is what the system responded.
7. We test the value 1 for VO changing only the A argument. Because of rounding errors the expected result is value close to 1, which is what the system responded.
8. We test the value 500 for VO changing only the A argument. Because of rounding errors the expected result is a value close to 500, which is what the system responded.
9. We test the value 949 for VO changing only the A argument. Because the arguments causes according to the mathematical formula a non-fractional result the expected value was exactly 949, which was what the system responded.
10. We test the highest range for VO changing only the A argument. The highest range is VI-VC which is in this case 950. For the A argument we pick a very high number. According to the formula the closer A reaches infinity the closer the value for VO changes to the highest range. The expected value for VO was a value close to 950 and this is also what the system responded.

11. We test the lowest range for VO changing only the C argument. Because of rounding errors the lowest value of VO was expected to be a value close to 0, which is what the system responded.
12. We test the value 1 for VO changing only the argument C. Because of rounding errors the expected result is a value close to 1, which is what the system responded.
13. We test the value 500 for VO changing only the argument C. Because the expected result is non-fractional the expected result is exactly 500, which is what the system responded.
14. We test the value 949 for VO changing only the argument C. Because of rounding errors the expected result is a value close to 949, which is what the system responded.
15. We test the highest range for VO changing only the argument C. The highest range for VO is VI-VC which is in this case 950. For the C argument we pick a very low number. According to the formula the closer C reaches 0 the closer the value for VO changes to the highest range. The expected value for VO was a value close to 950 and this is also what the system responded.

16. We test rule number 3. Setting A to 1.05 will result in a negative VO which is not legal according to the rule. Therefore and exception VI was expected and an exception VI was responded.

6 Conclusion

In figure 5.1 we see that every rule in the list of rules in section 4 has been followed. When an invalid argument was entered into the computer system that violates a rule from the list of rules an exception for that rule is received as expected. Therefore we can conclude that the computer system has passed the sanity test. The test shows that the computer system follows the current theory. However because we did not test with real arguments from the a test facility, therefore there is no guarantee that the computer system will calculate real arguments correctly.

This project is still incomplete and the next step for this project to build a mini test model and test the computer system on it.

7 Future

During the course of the project many areas have been identified as in need for further work. Because of the time schedule and priority requirements these have been left for the future instead. The following lists are some suggestions for improvements.

- Test the computer system on a real scale mini model.
- Automatic input of arguments, fetched directly from the measurement instruments.
- Export calculations in the format suggested by VMF.
- Add an automatic backup system and instructions for maintenance of it.
- Add a secure remote login, so that you can access and control the facility from any remote device.
- Investigate how VI and VC can be calculated using a test object and measuring the pressure and temperature changes.
- Investigate how air humidity affect the measurements and calculation.
- Round down calculation responses to more natural numbers.

8 References

[1]. Thomas Ekman. 2009-07-29. "Affärserbjudanden att teckna aktier I Wood Air AB". 2011-04-14. A short business proposition.

[2]. Ideal gas law. 2005-06-28. http://hyperphysics.phy-astr.gsu.edu/hbase/kinetic/idegas.html. 2011-05-08. A detailed description of the how the gas law works.

[2.1]. Ideal gas law. 2011-03-07. http://www.grc.nasa.gov/WWW/K-12/airplane/eqstat.html. 2011-05-08. A short description of the Ideal gas law from NASA Glenn Research Center.

[3]. PHP. 2011-04-14. http://www.php.net/. 2011-04-14. They manage the programmable scripting language.

[4]. Apache. 2011. http://www.apache.org/. 2011-04-14. Their main product is a versatile web server.

[5]. MySQL. 2011. http://www.mysql.com/. 2011-04-14. Their main product is a relational database management system.

[6]. XAMPP. 2011-01-27. http://www.apachefriends.org/en/xampp.html. 2011-04-14. Their main product is a web development package.

[7]. phpMyAdmin. 2011. http://www.phpmyadmin.net/home_page/index.php. 2011-04-14. Their main product is a MySQL client written in PHP.

[8]. MVC. 1997-03-04. http://st-www.cs.illinois.edu/users/smarch/st-docs/mvc.html. 2011-05-14. A description of the MVC design pattern.

[9]. UML. 2011-02-23. http://www.omg.org/gettingstarted/what_is_uml.htm. 2011-05-14. An introduction to UML.