

# Developing a financial simulation tool as a web application

Suraj Neupane



<b>Author(s)</b> Suraj Neupane	
<b>Degree programme</b> BITE10S	
<b>Report/thesis title</b> Developing a financial simulation tool, as web application	<b>Number of pages and appendix pages</b> 45 + 3
<p>“Kunnan Taitoa Oy”, a Finnish municipal financial expert, commissioned to upgrade its financial simulation tool from its current spreadsheet status to a web application. The principles of Open source served as the foundation of software development for a team of Haaga-Helia students who participated in the project ‘Taitoa’. The project aimed to deliver the working version of the web application. This thesis documents the process of application development of and the thesis itself is a project-based thesis. The paper highlights the benefit of using Agile development methodologies, Scrum over the traditional Waterfall model. Taitoa developed a routine to deliver its final product by the working increment of the software in three sprints. However, choosing backend or front-end technology from many possibilities as the need of application was the challenging set of tasks addressed at the beginning.</p> <p>While navigating through the technology dilemma, this study focuses on the advantages of using SPA (Single Page Application) technology for Taitoa. Deploying AngularJS as the frontend framework and PHP as the backend API language. Taitoa is powered by MariaDB relational databases.</p> <p>On the other hand, a monotonous UI (User Interface) of the spreadsheet needed to be transformed into more ‘Game-like simulation’ of a web application which was made possible with libraries like JQuery, Collapse.js, Bootstrap and C3 charts.</p> <p>Besides agile model used, a short requirement document was produced as a result of study the to give an overview of how the end user requirement was transformed to software as a solution and the software documentation is included in the paper to understand the core technology and its role in application.</p>	
<b>Keywords</b> Scrum, Open Source, AngularJS, PHP, Simulation tool	

## Table of contents

1	Abbreviation and Terms .....	1
2	Introduction .....	1
2.1	Background .....	1
2.2	Need for the application .....	2
2.3	Objective .....	2
2.4	Scope of the thesis .....	3
3	Software development technology .....	4
3.1	Open source .....	4
3.2	Closed source .....	4
3.3	Software development methodologies .....	6
3.3.1	Waterfall model .....	6
3.3.2	Scrum as agile development model .....	7
3.4	Choosing right technology .....	9
3.4.1	Backend .....	10
3.4.2	Frontend .....	12
3.4.3	Angular JS .....	15
4	System requirement specification .....	17
4.1	Current system .....	17
4.2	Purposed new system .....	18
4.3	Simulation scenario .....	19
4.4	Project phase and steps .....	20
4.5	Output of the project .....	20
5	Software documentation .....	22
5.1	System overview .....	22
5.1.1	Maria DB .....	23
5.1.2	PHP as backend language .....	23
5.1.3	AngularJS .....	24
5.2	Use cases .....	24
5.2.1	Actors .....	24
5.2.2	Use cases for general user .....	24
5.2.3	Use cases for administrative user .....	27
5.3	Data design .....	28
5.3.1	Entity relation diagram .....	28
5.3.2	Data Dictionary .....	29
5.4	User Interface Design .....	30
6	Implementation .....	33
6.1	Setting development environment .....	33

6.1.1	Proto server .....	33
6.1.2	Git repository .....	34
6.1.3	Pivotal tracker .....	34
6.1.4	Slack .....	35
6.2	Backend .....	35
6.3	Frontend.....	37
7	Conclusion and Result .....	39
8	Attachments.....	41
	References .....	43
	Appendices.....	47
	Appendix 1. Data Dictionary for City table present in database.....	47
	Appendix 2. Data Dictionary for Cityname table present in database.....	49
	Appendix 3. Data Dictionary for Population table present in database.....	49

# 1 Abbreviation and Terms

Terms	Description
JavaScript	Dynamic programming language also known as ECMAScript
HTML	Hyper Text Markup Language
JSON	JavaScript Object Notation
CURD	Create, Update, Read and Delete
JS	JavaScript
AJAX	Asynchronous JavaScript and XML
API	Application Program Interface
REST	Representational State Transfer is an architectural style, and an approach to communications that is often used in the development of Web services.
SCRUM	Agile software development methodology
Sprint	Work cycle time box in Scrum
AWS	Amazon Web Server
DOM	Document Object Model
UI	User Interface
ERD	Entity-Relationship Diagram
ER	Entity Relation
Front-end frame-works	Framework dedicated to manage front-end components
Frontend JavaScript Framework	Front-end framework built upon JavaScript
PHP	Server Side scripting language

## 2 Introduction

### 2.1 Background

Kunnan Taitoa Oy is a Finnish company with strong experience in municipal financial management along with other fields like human resources and financial services sector to support the business process of their client, especially municipal cooperation, supporting their need for purchase, sales, debt collection, payment, accounting, basic customer service etc.

The foundation of the financial simulation tool software was the outcome of Taru Project.

The project aims to create a solid ground for new innovative digital financial management services as well as automated financial processes.

Project title refers to a whole automated financial management process that combines individual financial data flows into a single data stream; corporate, municipal and public sector to take advantage of real-time and further refine the process of moving information. This helps to reduce the administrative burden and enable better management of the economy.

The project was commissioned by four companies: Administer Oy, Kunnan Taitoa Oy, Tieto Oyj and Tikon Oy. . The project is partly funded by Tekes: a two-year network project, and it works in close cooperation with academic institutions, governments and relevant stakeholders, such as the Federation of Finnish Finance, Finnish Financial Management Association and the Finnish Technology Industries. The project began in January 2014 and will end in December 2015. The project has a budget of EUR 2.44 million.

## **2.2 Need for the application**

Recently, the company felt the need to improve the current tools for decision making process into more dynamic and modular system. Basic excel sheets are used to input different attributes and simulate the future trend; however the company wants something more as enriched UI and dynamical representation of the graphs and charts.

Along with the modular application the functionality of the application should allow users to change certain variable like corporate tax rates, property tax etc. and should simulate the future trend. This should be reflected in charts and graphs with likely comparisons of different attributes.

Although the ministry of finance has already developed a similar tool in Microsoft excel, this project was established to transform the existing tools into a web application with a friendly UI. This project have been undergone for first sprint and first phase of the product has been delivered.

## **2.3 Objective**

### **Objective of the thesis**

The objective of the thesis paper would be to document the software development process, technology used and theory on open source technology and software development methodology. The need of the client as a web application by understanding the requirement. Transforming the current business process with the new web application and choosing the right technology through comparison of available skills and other resources.

## **2.4 Scope of the thesis**

The scope of the thesis paper would be to outline the software development process and its challenges, choosing the right technology and its benefits. The thesis paper would be a support material or guideline for developing a new system in the future. The paper would document the challenges faced during the development process.

The thesis itself is a product based thesis, so the research and theory would assist the development of the product and it's paramount of financial judgement. Thus the paper won't have any hypothesis to prove, rather it would focus on the feature of the new simulation tools and its footholds for its end users.

### **3 Software development technology**

Every software technology has strengths, weaknesses, history and culture. Wise choices will help make your project successful; poor choices will tend to cause difficulties (Art & Logic, 2015). Similarly, choosing a right foundation of software development would be even more challenging when it comes to the changing nature of technological evolution in such a fast phase.

However, the Open and Closed sourced technology have their own pros and cons. "In today's third-party, Internet-of-Things world, the only truth is that all software - be it Open or Closed-source - is inherently insecure" says Marissa Tejada (The internet of thing, 2015), specialized writer in platform technology.

#### **3.1 Open source**

Open source technology is defined as the production and development philosophy of allowing end users and developers to not only see the source code of software, but modify it as well (John Garger,2010). For illustration, the Linux operating system is one of the best-known examples of open source software technology paradoxes of the Microsoft approach to closed source where the end user ability to access the code is forbidden.

#### **Free source versus open source:**

As the topic indicates, free is equal to freedom, not price. There might still exist some confusion in the use of this term as some studies suggest that "open source" has re-placed the term "free software", while others still cannot see the relation between free and open. The ideology behind free software was first introduced within 1984 and 1985 by a system programmer Richard M. Stallman, who founded the free software foundation and created the "GNU software system". (Felicia G., 2013). Freedom according to Merriam-Webster dictionary is said to be "unrestricted use" and "thegnuproject" uses the word "unfettered" as the best suited definition of freedom and says all other alternatives as unfit for the definition of its use of "freedom" in this context (Richard Stallman, The GNU Project).

#### **3.2 Closed source**

Although, open source technology is more preferred over closed source, however, it is not the case that close source has only the drawbacks. For software, "the license is the product."(The License Is the Product, 1998). If this statement is correct, then it follows that software products cannot exist without licenses. The proposition that the license is the product flows, in turn, from the claim that software products are "intangible." Raymond



Nimmer, the principal drafter of the Uniform Computer Information Transactions Act, declares that the U.S. economy is experiencing “radical changes in the source of value and in how that value is commercially distributed.” (Phillips, Douglas E., 2009). Therefore, a proprietary software opens the possibility of a business model along with an integrated solution for the software, where fixing bug, version release, distribution and support are well organised and in-house.

### **Open versus close source:**

Open source software is recently gaining popularity among the development community. Due to the acceptance from a large variety of industries, demonstration of functioning equally good well as proprietary software, open source is the foundation for some of the most successful software platforms like android and iOS. Eric Raymond, and in his now famous book “The Cathedral and the Bazaar” he talks about the importance of Linux, as it was the very first time the open source developer community showed that not only complex and sophisticated software can be built in such way, but also that business models can be built around such way of software development and distribution.

Similarly, open source development has transformed its community with version control tools like GitHub, which allow diverse and distant developer to communicate and contribute to complex project. This approach of solving problem with source code shared with the developer results of the most secure and organized programming approach. Large OSCs (Open Source Communities), like the ones under the Apache Foundation with over 100 OSS (Open Source System) projects, have self-organized to produce complex enterprise-grade software. Most of the developers contributing to OSS projects are unpaid, geographically distributed, and yet integrated into highly organized and structured fabric of OSS projects. (Eric Raymond, 2001).

On the other hand, in-house software has lost its significant market share, this was not the case back in 1980s. The traditional model of software business has been standardized with sales of license, which principally oppose sharing of source code in public. This approach was fine till the business was the monopoly of few player, but its complexity grew with the competition and production cost skyrocketed. Likewise, there is also an interest not only in participating in an open source community, but to provide a software product to an open source community (FLOSS Research and Development FLOSS’07)

### **3.3 Software development methodologies**

Software development methods have evolved as the complexity of the project have increased and need for the rapid development created more value and opportunities for the software companies. Although, the traditional software development methods (TSDM) are still used for a specific kind of project, however agile software development methodology (ASDM) are more popular among the developer communities. There are different approach for software development which can be listed as follows:

Agile(Scrum, DSDM(Dynamic system development model), Crystal)

Spiral

RAD(rapid application development)

Waterfall(traditional method)

However, in this paper we are going to compare Scrum as Agile method where as waterfall as traditional methods. The comparisons will provide the insight for how the development methodologies should be selected for different projects according to their need.

Traditional methods were very disciplined and sequential. There were no testing phase before the product was ready. So a little fault had a strong and big impact because each phase is depends on the previous step and nothing was checked between the stages. Assessing the applicability of Traditional and Agile methodology to enhance success of software development projects.

On the other hand, ASDM is more flexible in the process and change needed to the project. Short documentation, rational task breakdown, collaboration with the customer and test driven development provide a strength in the development process. Despite, the efficient product testing, customer co-ordination and change adaptation this approach lacks the effectiveness in a big organization with considerable number of team members. According to VersionOne, Inc. 2014 survey, 10% agile projects failed because of the broader organization or communication problem. (VersionOne, Inc. 2014)

We are shortly going to discuss about Waterfall as TSDM methodology as well as Scrum as ASDM.

#### **3.3.1 Waterfall model**

The Waterfall model was created by Winston Royce in 1970 and considered as the classical approach for software development (Dooley 2011, 9). The sequential model, where project development is seen as flowing steadily downwards (like a waterfall) through the phases of software development requirements analysis, UI design, software implementa-

tion, project verification and software maintenance (The waterfall model, 2015). This method puts the solid customer need in project documentation as required. This document later works as the guideline for the team to navigate towards the goal achievement.

However, this methodology doesn't take into inevitable changes and feedback at the end of each phase which puts the whole project into risk. The other weakness of the model includes the possibilities of not being able to detect the wrong decision till the implementation or testing phase.

Figure 3.3.1 below shows the waterfall model of software development and list the different stages in the cycle, which starts with requirement and ends with the software maintenance phase.

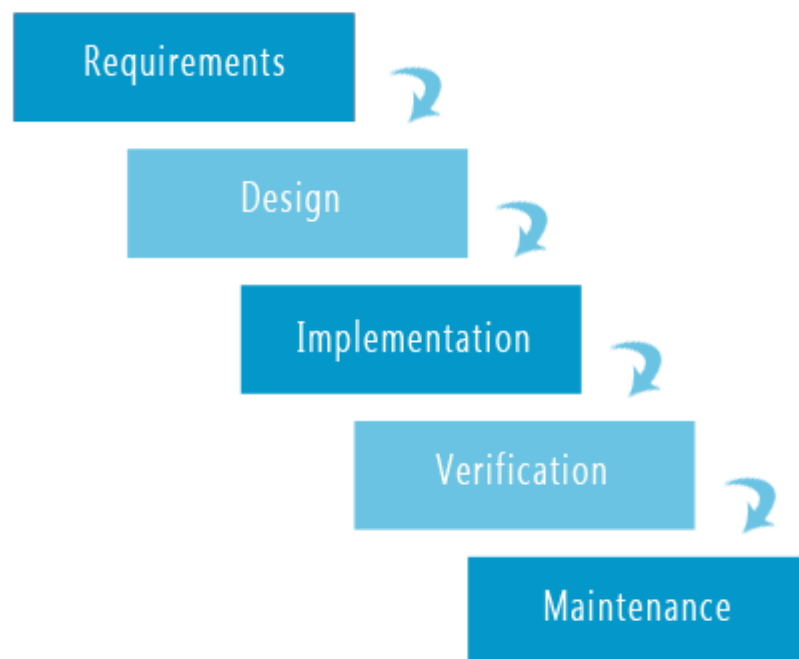


Figure 3.3.1 Waterfall Model of software development

### 3.3.2 Scrum as agile development model

The software project has grown so complex that advocating any single development model makes no sense. Neither agile nor conventional methodology can address the need of every project, nonetheless, Scrum stands frontline for addressing the need of most project. Scrum methodology is the iterative and incremental practices which use to manage complex software development process (Bushra Mostafa, 2015). Companies that build

software for a living usually sell methodologies (Schwaber, Ken, 2002, 23). The very reason why the Scrum works for different varieties of software projects.

The beauty of Scrum is that nothing is rigid everything can be changed according to the need of the customer even late in the development phase. Takeuchi and Nonaka, in their famous Harvard Business Review article [Takeuchi and Nonaka], analysis that ten of the competitive and innovative companies on the planet have a requirement of 'Built-in instability', meaning that team member are given the freedom to do research and creative but at the same time they are expected to produce up-to high standards. The same philosophy applied to the Scrum team as well.

### **The Scrum Team:**

Cross functional and self-organizing is the basic principal of the Scrum team. There are three main role in the team as follows:

- The Product Owner: The main role to be played as the owner of the product and items of the product backlog
- The Development Team: Plan, organize and develop incremental and functional deliveries of the backlog items.
- The Scrum Master: "The Scrum Master is responsible for the success of Scrum" (Schwaber & Beedle 2002, 31). He/ she would be finding the best way to manage the product backlog.

### **Scrum Events:**

As a software development methodology Scrum model has a collection of events which bring the project into more organized shape and facilitate team toward the goal formation process. The following events are the part of the Scrum:

- Sprint planning
- Sprint Review
- Daily scrum meeting and
- Sprint Retrospective

### **Scrum Process:**

Scrum process is a flow of customer need and desire through the product owner. The first event occurs is the sprint which is a timeline for a certain list of product backlog to be accomplished into a functional module of software. The sprints ends only if the desired set goal is met. During the sprint, no backlog can be changed and quality set aside by the sprint meeting can't be reduced. The product owner is always responsible for the backlog and can change it according to his need, however the changed backlog would be included

in the next sprint and have no effect on the ongoing one. The backlog can be broken into tasks in daily scrum or weekly scrum meeting, according to the development team needs.

The Scrum process figure 3.3.2 shows the process how the product backlog is allocated to a sprint backlog and break many. Product backlog can be chosen according to the need of product owner. A sprint length is determined by scrum master and adjusted according to the need of the project. Finally, the working increment of the software is delivered at the end of the final sprint meeting.

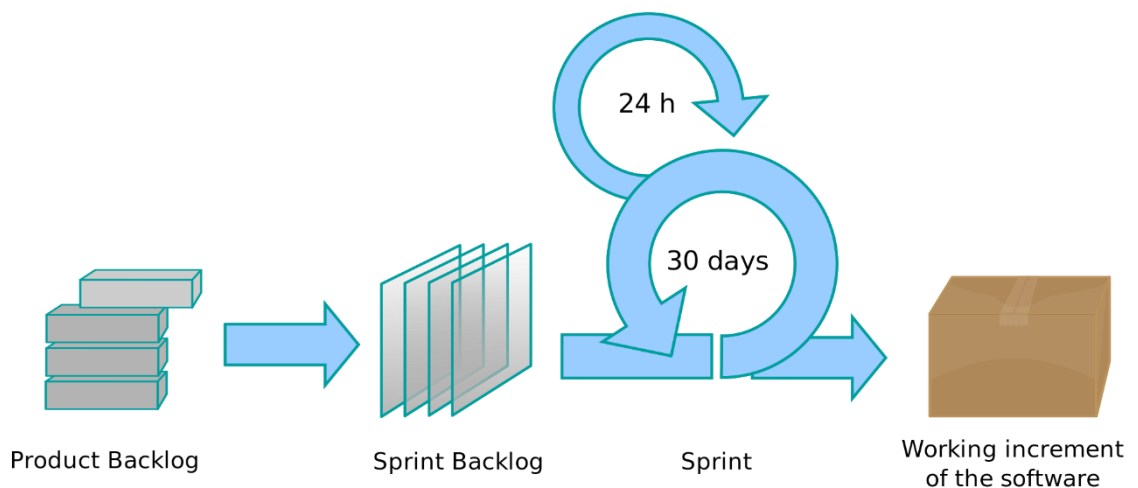


Figure 3.3.2 Scrum process with product backlog, sprint backlog and Sprint

### 3.4 Choosing right technology

Since about 2004, the World Wide Web has undergone great changes in perception, use and implementation, summarized under the vague term “Web 2.0”, made popular by Tim O’Reilly (T. O’Reilly, 2005).

Similarly, modern web application development has evolved from complex backend technology to client side JavaScript technology. Recent year lots of JavaScript framework and libraries have come into existence as need of the project. This dramatically improves the performance of the application, the processing power required and cost of running the server. The increased use of JavaScript in the browser prompted vendors to pay more attention to JavaScript execution performance and led to the development of new JavaScript engines and, in the case of Google’s Chrome, even an entirely new browser dedicated to support complex client-side JavaScript-driven web applications (Jan Varwig, 2009).

However, JavaScript along in frontend have no singular existence without the support of backend technology. Language like Java, Python, PHP and Ruby, etc. are more mature technology with the stable MVC (Model View Controller) framework and developer friendly IDE. With conventional web applications, user interface aspects are distributed over the server and clients, however, client is responsible primarily for accepting user interface events (Jan Varwig, 2009). These languages handle the communication between server, database and client's browser. Not long ago, backend were responsible for providing view data for HTML, however, this principle have been changed with the JavaScript framework like Backbone and AngularJS where the view is entirely manipulated in the client atmosphere. This evolution in technology has, scrutinizes the role of backend for just providing data and hence and forth from the database to browser.

Alongside with pros and cons of frontend and backend technologies it is still a hard choice to select the right technology like framework and programming language. There are several factors to be taken into account before selecting the right environment for the team. When a web development team develops a project, there are two important factors: how skillful the development personnel are, and what kind of tools they use (Changpil Lee, 2012).

### **3.4.1 Backend**

The first phase of development required a precise understanding of team awareness toward the technology. The backend development takes into account of framework and server side language for the minimum viable product (AS, 2014). On the other hand, further scaling, and optimizing the software takes a detail plan for reinventing the product.

However, there are basically three main areas of concern for implementing the right backend technology database, framework or language and production environment.

Firstly, understanding the relation between data and its structure for your application let you choose the right database, for illustration, relational database like MySQL, PostgreSQL or NoSQL database like MongoDB. The latter one will be easier to scale with increasing audience, whereas the other will be more prefer for relational data among the table of the database.

Secondly, choosing the server side language depends, among the development team. Almost all the language is well matured and has community support. A widely popular framework like spring is available for Java and PHP whereas Rails is a huge popular with Ruby. The above mentioned framework will provide common library and API for user au-

thentication and other general implementation as well as automated developing, building, testing and deploying. This makes the development work more secure, fast and easy because all the modules are tested by the large community and every security breach taken into consideration.

Finally, deployment is the final stage to bring your application to the live audience of the internet. Recently, cloud services like Amazon (AWS), Microsoft (Azure) and Google have entered into infrastructure business making a reality for a small group of developer to host their application on the cloud to reach to larger customer without owning any physical infrastructure of their own. "Take Snapchat, the photo-swapping service that recently turned down a multibillion-dollar takeover offer from Facebook. It processes 4,000 pictures a second on Google's servers, but is just two years old and has fewer than 30 employees. The company started out working with a Google service that helps young companies create applications and was chosen by Google to be an early customer of its cloud", writes Quentin Hardy, technology editor in the New York Times.

Thus, understanding the need of the application, the audience and development team determines the factor of backend like database, server side language or framework and cloud deployment instant.

#### **PHP as server side language:**

PHP (Hypertext Preprocessor) is a popular general-purpose scripting language that is especially suited to web development (The PHP Group, 2015). PHP dates back to 1995 created by Rasmus Lerdorf, since then most of the world's famous sites have been implemented this technology for the development of dynamic web application.

PHP is evaluated and processed only on the server and only the processed result is delivered for the client, meaning that no PHP script can be decoded on the browser. There are different combination of technology, which is used with PHP to form a full stack. Among them LAMP (Linux, Apache, MySQL and PHP) stack is considered the most reliable open sourced combined technology, which is capable to scale as the need of application.

Being the most popular programming language always gives the opportunity for the developer to choose the various frameworks according to their needs. PHP supports various different operating system and web server combinations. (Lengstorf 2009, 4; Trachtenberg & Sklar 2006, xv.) Not only this, there are numerous frameworks supporting PHP for the rapid development. "Using a framework also facilitates scalability and long-term

maintenance by complying with development standards, keeping your code organized and allowing your application to evolve and grow over time.” Grace Smith an experienced web designer and founder of Postscript5.

The figure 3.4.1 below shows the market share of the PHP and other frequently used programming languages. According to figures, although PHP leads the market share as programming language, but its increasing rate is way below Ruby and ColdFusion so we can conclude the fact that PHP is already in its phase of market precipitation.

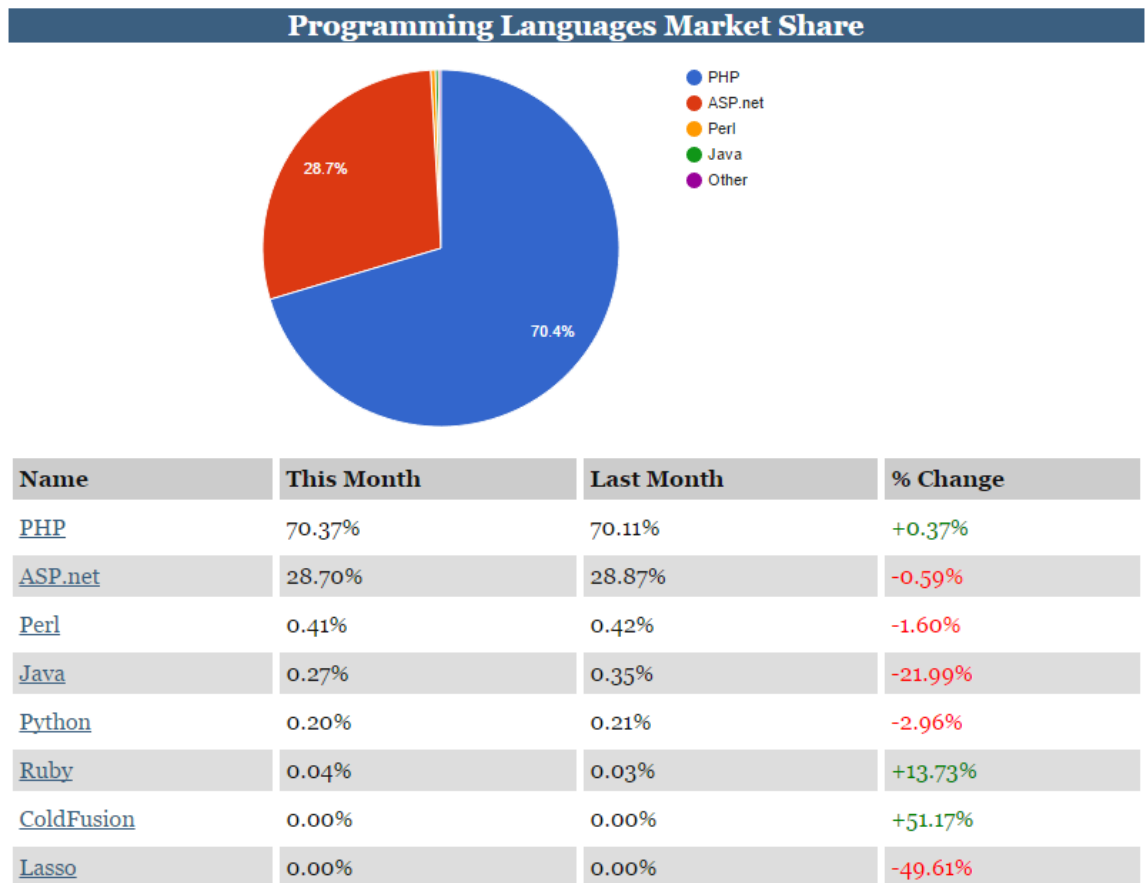


Figure 3.4.1 Market share of different programming language used today

### 3.4.2 Frontend

The growth of mobile application has segregated frontend as independent from the backend where data processing and business logic is maintained. A shift in consumer from traditional shopping behaviour to online have forced business to transform themselves into e-commerce entity. This transition took a whole new level when a mobile application came into existence. All the e-business were forced to provide an application for their costumer at least in two mobile platform iOS and Android. Back in 2012, Bill Ready, CEO of Braintree Payment Solution, forecasted in his article,” If Facebook, one of the



world's leading mobile sites, could be caught off guard by the shift to mobile, who else is poised to suffer a similar fate? One answer: the entire e-commerce industry" (Bill Ready, 2012).

Eventually, this transformation led to the development of independent frontend design for different devices like mobile and web which communicate to the API. API based application would serve the data from cloud making it possible for accessing the same data from different devices, however, this new architecture of the application needed a common frontend design which was fulfilled by the responsive mobile first frontend framework like Bootstrap.

Although, there are various solution for the UI design of the application, however, finding the right technology is first about deciding the application type; mobile, web or desktop app. The web application needs to be cross browser supportive; developed using the combination of HTML, CSS and JavaScript. On the other hand, targeting the specific app store for mobile (iOS, Android and Windows) require different approaches. Frontend design requires an intensive work to make it look simple, uncluttered and responsive (AS, 2014).

Figure 3.4.2 below describes the way how the modern API based application functions. RESTful API for cloud serves the JSON or XML data as the need of the user and frontend are responsible for their individual UI. This saves a lot of processing power of server which can be used in enhancing other capabilities of the application like scalability.

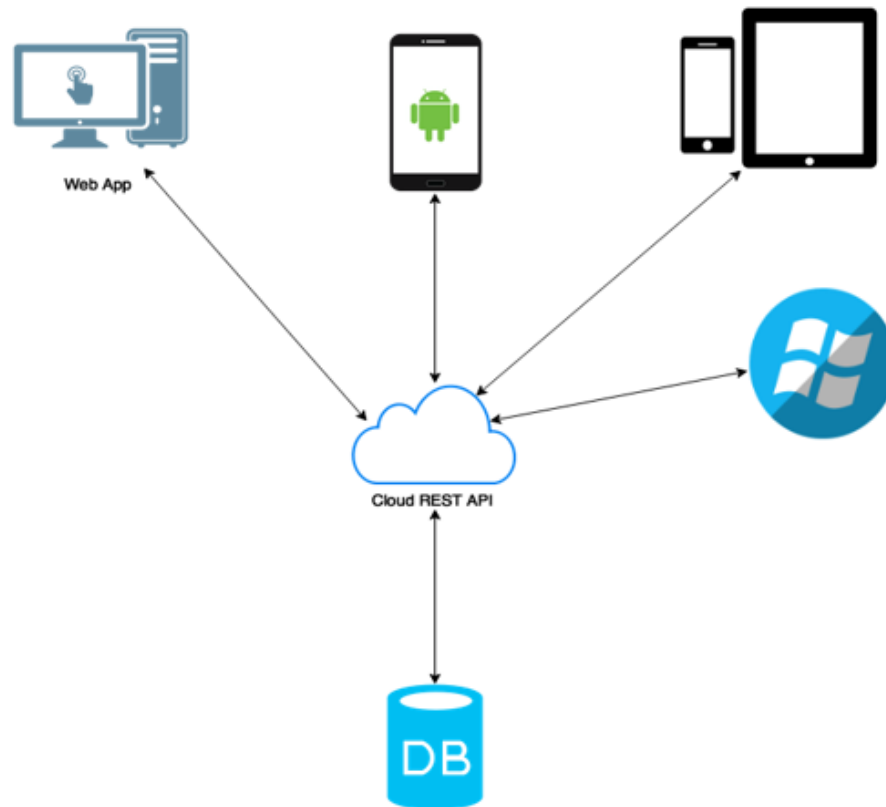


Figure 3.4.2 Modern RESTful API based application

### **Bootstrap as responsive UI:**

Bootstrap and other similar frontend framework are very popular among the developer community because of its elegant design and responsiveness; which works on both mobile and web. The framework itself shipped with all the required JavaScript dependencies for its plugins for different modules.

Any framework or library need to have a contributor, documentation and community for its regular update and enhancement; for this matter, bootstrap have come a long way with the current version of 3.4.3 and bootstrap version 4 alpha coming soon, meaning that a frequent update and improvements are made by its contributor. All the component is well documented and available in many languages and it is distributed under MIT license.

After its release bootstrap gained popularity because of its elegant design and easy to use by the beginners. Similarly, grid system on web was first introduced by bootstrap which empower the frontend developer by providing the tools to allocate and design the content with the right screen size independent of device type.

## Grid options

See how aspects of the Bootstrap grid system work across multiple devices with a handy table.

	Extra small devices Phones (<768px)	Small devices Tablets (≥768px)	Medium devices Desktops (≥992px)	Large devices Desktops (≥1200px)
<b>Grid behavior</b>	Horizontal at all times	Collapsed to start, horizontal above breakpoints		
<b>Container width</b>	None (auto)	750px	970px	1170px
<b>Class prefix</b>	<code>.col-xs-</code>	<code>.col-sm-</code>	<code>.col-md-</code>	<code>.col-lg-</code>
<b># of columns</b>	12			
<b>Column width</b>	Auto	~62px	~81px	~97px
<b>Gutter width</b>	30px (15px on each side of a column)			
<b>Nestable</b>	Yes			
<b>Offsets</b>	Yes			
<b>Column ordering</b>	Yes			

Figure: 3.4.3 Bootstrap grid system with scale in different size of screen

The Figure 2.5 above shows the screen size for multiple devices and how the CSS is fixed for the column width. The available number of column is 12 for each device but the class prefix change to xs, sm, md and lg respectively for the size of screen in increasing order.

### 3.4.3 Angular JS

AngularJS is a structural framework for dynamic web application. Angular is what HTML would have been, had it been designed for applications (AngularJS, 2015). In 2009, Google team of developer introduced to a new JavaScript framework which could manipulate the HTML code dynamically according to the need of the application. This had since then redefined a standard of developing SPA (Single Page Application).

As a framework, Angular is capable of managing all the front-end need like DOM manipulation, form validation, DOM event handling and two way data binding. On the other hand, routing for CURD (Create Update Read and Delete) application, back-end Ajax communication with the server and dependency injection are some out-of-the-box features that shipped with Angular.

Unlike JavaScript library, Angular is a full framework, which makes it learning curve steep and need a strong foundation of JavaScript knowledge already at the very beginning. It is backed by the Google as its flagship language of development which have interested a lot of developers and in its short period of evolution Angular have one of the most active

community and support documentation. Heavy traffic sites like YouTube is developed on top of Angular technology and it is shipped under the MIT license.

The figure 3.4.4 below shows the interest overtime for the existing big four frontend framework. The interest in Angular have risen significantly since 2012/2013 while other have a slight increase in software developer interest.

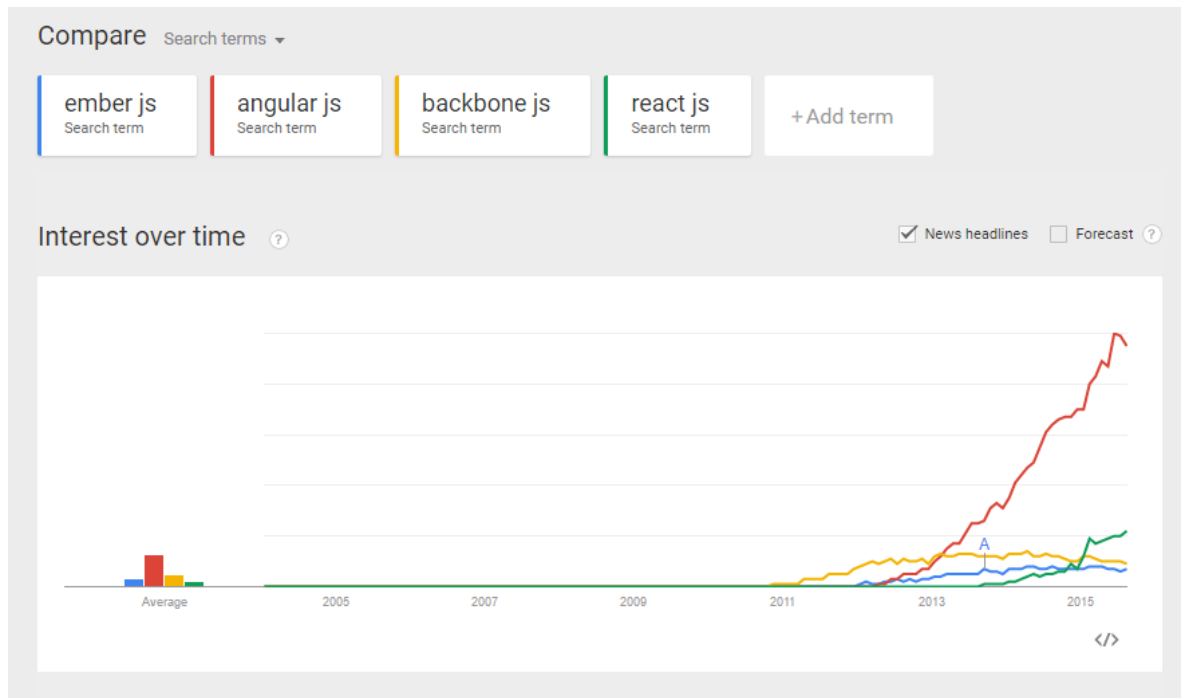


Figure 3.4.4 Interest over time for big four frontend framework

## 4 System requirement specification

In real life, developing an application starts with the need of the customer. Understanding the need of the customer and bringing the development environment parallel to the requirement of the end user is tricky and sometime challenging if the both party fails to understand the common ground of the problem. The well-known procedure to reduce the risk of this kind is to prepare the requirement specification mostly done by the member of end user and supported by the development team member.

“Although, writing a complete requirements document is time-consuming, there are many advantages to having one. For one thing, involving major stake holders in writing requirements helps to ensure that everyone agrees on what is to be done. This can avert misunderstandings down the road and save time that later might be wasted in rework. You can use a requirements document to help set expectations about what will and will not be accomplished in the current development cycle.”(Somerville, Ian & Pete Sawyer, 1997).

### 4.1 Current system

The current excel sheet is basic working model of simulation tool. It works fine to support decision making process, however, the disadvantage of the current simulation tool is UI that looks bit old fashioned, on the other hand, the graphical representation are limited with no more option to explore and compare the result with different prospect.

Similarly, the input method are very basic, only what the excel sheet provides, this could be more simplified with dynamically changing the attributes, but not saving them in system, only for the purpose of comparison and simulation process.

The figure 4.1 below shows the current working model of tool for Taitoa employee. The tool used is an excel sheet which creates graphs and values can be updated for the city selected. The current version of tool was created by finance ministry and Taitoa want to transform the current tool into a web application.

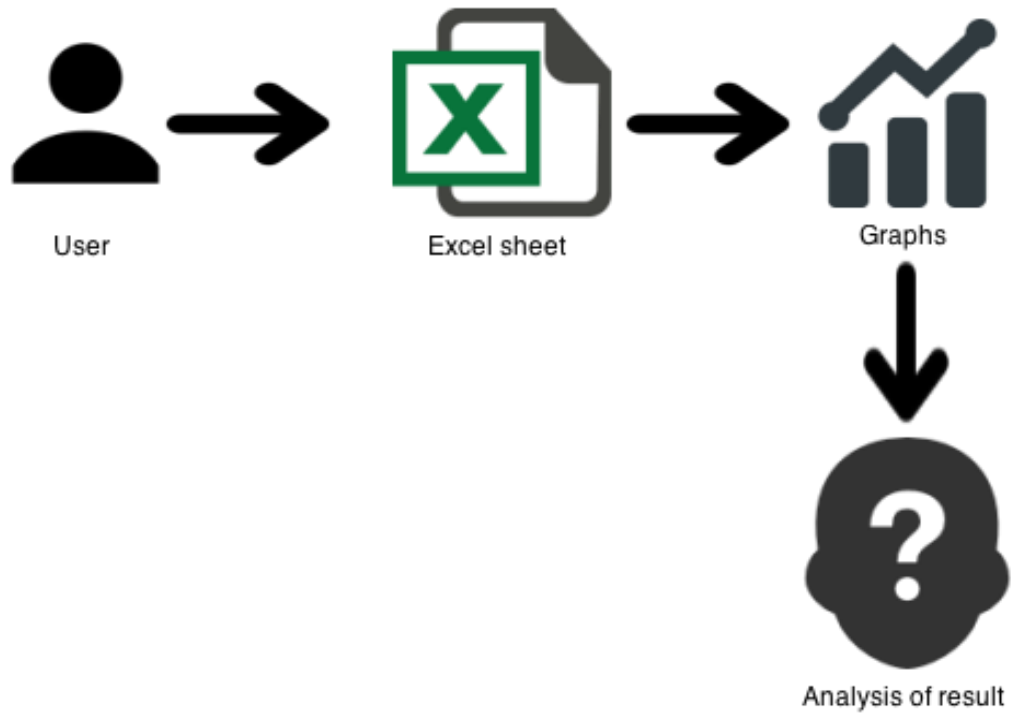


Figure 4.1: Current working model of application

#### 4.2 Purposed new system

The purpose of the new system is to provide the user friendly interface to interact with stored data to change certain attributes, and simulate the other relation dependencies. Along with manipulating data simulation tool will map trend and forecast in graphs and charts for visual representation.

The figure 4.2 below shows the purposed new system for Taitoa. There are three module simulation system, data import and financial statement which would interact to the database and provide a game like user experience with the new system. End User will be able to simulate the future data with the base data of past year can see the changes in graphs with more friendly UI.

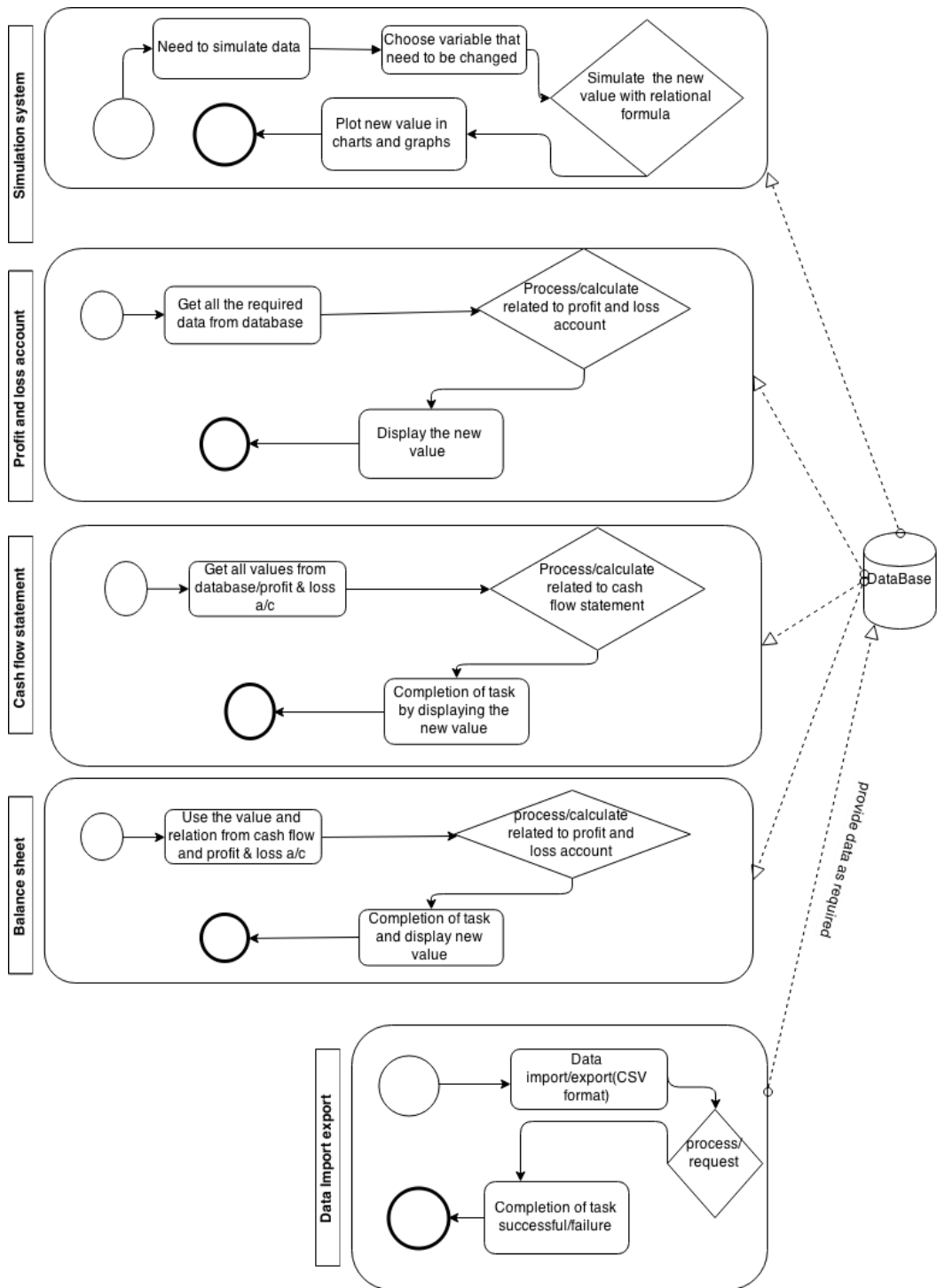


Figure 4.2: BPMN for purposed new system

### 4.3 Simulation scenario

The simulation of the financial data to forecast the future trend would basically require graphical interface, specially charts and graphs. Also the forecast attributes should be changed. At the first phase of project only two attributes would be changed named tax rate and loan amount. The simulation works as such that the tax rate changed would affect all the factor of profit and loss account and cash flow statements or balance sheet.

#### **4.4 Project phase and steps**

The project would be carried out in three different phase. The first phase would deliver the relation of entities from the excel sheet and data modeling would be produced. The data model prepared would be provided to the costumer and the final go ahead would take a project to second step.

The development team would at the first step work on the basis of data modeling document to provide the calculations of all the relation of data for the attributes of profit and loss a/c, balance sheet etc. In first stage only two of the attributes of the table should be allowed to change and this change should produce the difference or changed in relational calculation of the financial data.

The second stage of the project would be about two more attributes change and the import of the data from excel format. Taitoa as client, have no technical knowledge of software development. So, to overcome this problem and assist the first phase or data modeling phase of the purposed system would be done by accounting knowledge persons. This would provide a smooth transition of the requirement to the development phase.

#### **4.5 Output of the project**

The project would last for 12 weeks and the end of the project would deliver the following items:

- Final working software for simulation of financial data for future
- Software Requirement Document
- Software Documentation
- UI testing report (if applicable)
- System training document (if applicable)
- This new system would be more user friendly for Taitoa user with improved User Interface. The financial data would now be stored in remote database accessible via web rather than old model of data in individual excel sheet.
- The important transformation would be the graph and charts for the project. This would support the decision making process more efficient and fast than current version of tool.

The figure 4.3 below shows the overview of system from end user prospective. The three module will be accessible by the end user through the UI where they can read and update financial data, update and analysis chart and import data to database.



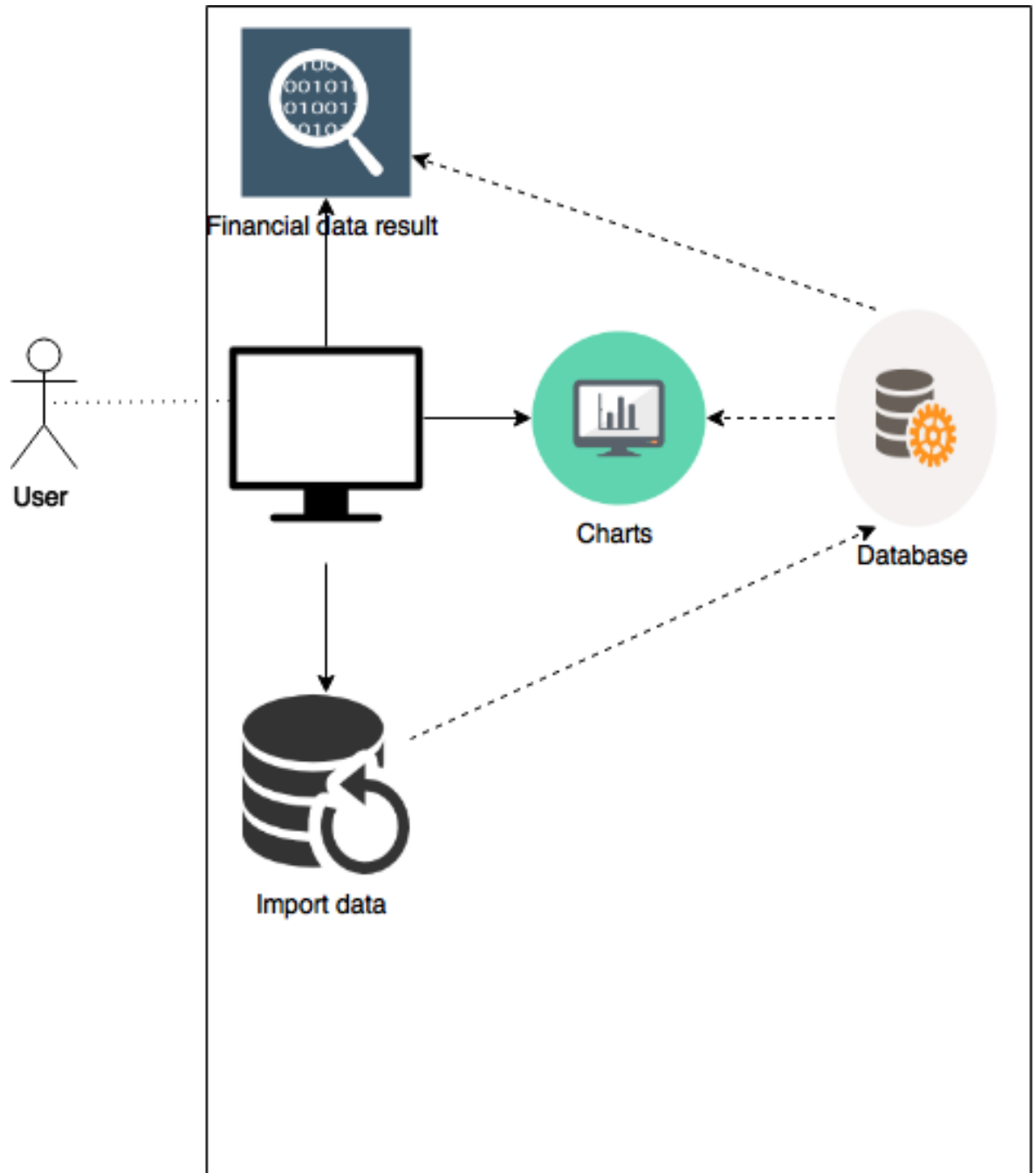


Figure 4.3 End user perspective of the system

## 5 Software documentation

Software design is a process by which the software requirements are translated into a representation of software components, interfaces, and data necessary for the implementation phase (IEEE Std. 1016-1998).

System design specification works for the development team as part of the guideline and translate the requirement document into more detail technical checklist primary reference for code development and, therefore, it must contain all the information required by a programmer to write code.

### 5.1 System overview

The core functionality of the application would transform the basic excel sheet into a more user friendly game like simulated modular app. However, the existing information and use of the excel sheet would be the business logic which would only be reinvented with the new UI design.

The new application would have three main functions and the module listed below:

- Graphical representation
- Financial simulation
- Financial data import

System overview of the technology used can be noted as:

System type: Web based application

- Database: Maria DB
- Language: PHP, JavaScript
- Front end library and framework: Bootstrap and JQuery

The system overview of the different model and layer of application can be seen in Figure 5.1 below. The presentation layer which includes UI communicates with the application layer where graphical representation module, data import module and financial simulation module are present. However, the presentation layer is buffered with application layer and no direct communication is possible with the database layer.

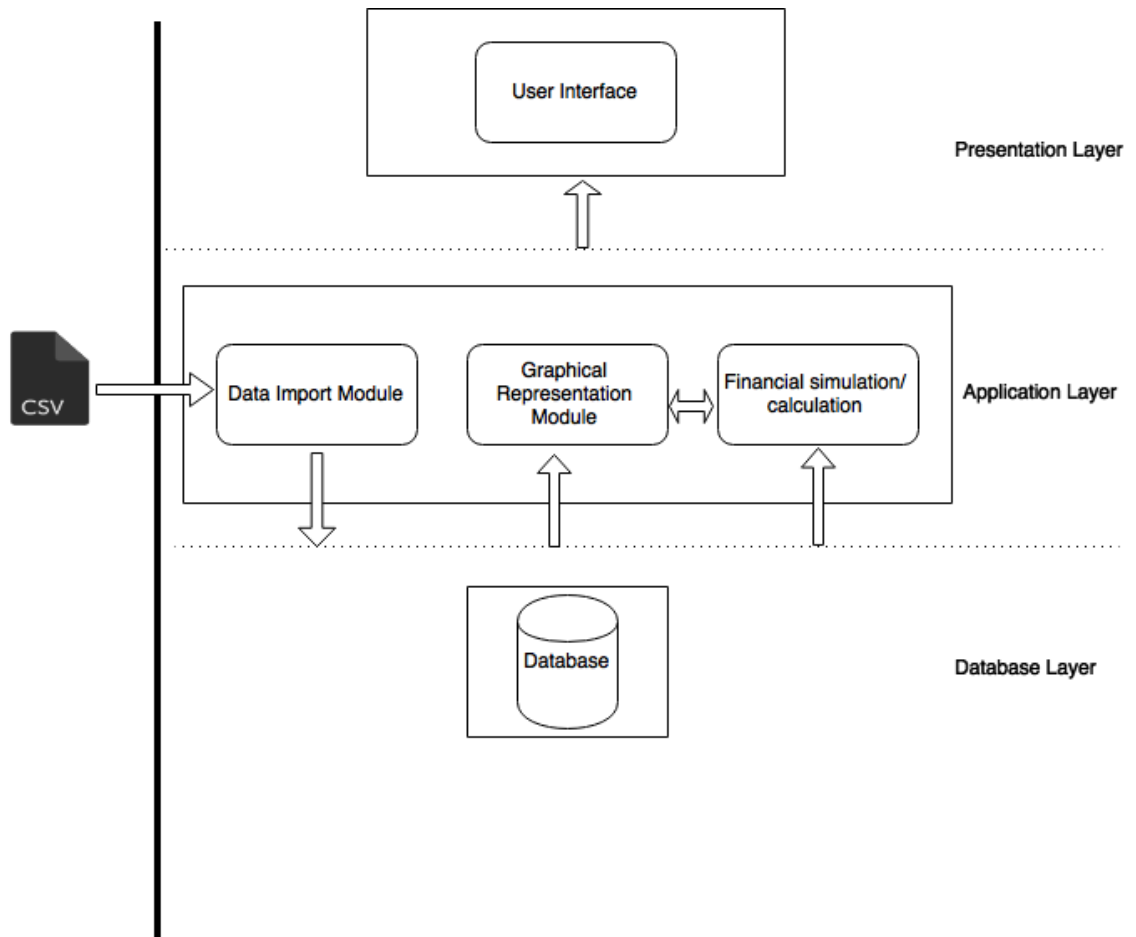


Figure 5.1 System Overview Modular Diagram

### 5.1.1 Maria DB

Maria DB is a drop-in replacement for MySQL. Maria DB strives to be the logical choice for database professionals looking for a robust, scalable, and reliable SQL Server (MariaDB, 2015). Taitoa is using Maria DB as the open source database system to store the financial data of the corresponding fiscal year. The current version available is 10.0 and the project will use the same version.

### 5.1.2 PHP as backend language

The server side programming language for Taitoa is PHP. The application is built with the SPA structure, thus PHP serves the JSON data as the API for the web application. The current version of PHP used in the project is 5.6. The role of the server side language is limited in SPA where the processing of data and view is generated by with the frontend framework.

### 5.1.3 AngularJS

Limiting the role of back-end and PHP, Taitoa project will implement SPA with the full stack use of AngularJS as a front-end framework. Apart from data calculation and presentation, Angular will be responsible for DOM manipulation with the integration of the Bootstrap for UI design. The application consist of input field from the user which will update or change the value of the other depending variable, so two way data binding for ng-model will be used to watch the change and calculate the updated variable.

## 5.2 Use cases

A use case is a methodology used in system analysis to identify, clarify, and organize system requirements (Margaret Rouse, 2007).It refers to the system activities performed by the user.

### 5.2.1 Actors

**General User:** The simulation financial web application would be accessible to the employee of Kunnan Taitoa. However, there is no authentication module in the application, meaning that the person aware of the link can access the application for their informative use. However, the data import module will be authorised for a certain user group.

**Administrative User:** The data import module of the application cannot be authorized for the general user. This module adds the financial data into the database so making sure the correct data gets into the system required the authentication and authorization for the user group which would be called an administrative user.

### 5.2.2 Use cases for general user

#### General User Use Case -Input for simulation:

<b>Use case name:</b> Input for simulation	<b>ID:</b> 001	<b>Priority:</b> High
<b>Primary actor.</b> General User	<b>Source:</b> Employee	<b>Use case type:</b> Business
<b>Interested Stakeholders:</b> Employee of Taitoa or the user of the application		
<b>Brief description:</b> This use case describe the input of the certain value from the user to input field. The		

goal of actor is to get the result of the dependent variables for that field.
<b>Goal:</b> The successful completion of calculation and display of new calculated value for dependent variable.

Table 5.2.1: General user input for simulation

**General User Use Case- Create graph and chart for input values:**

<b>Use case name:</b> Create graph and chart for input values	<b>ID:</b> 002	<b>Priority:</b> High
<b>Primary actor.</b> General User	<b>Source:</b> Employee	<b>Use case type:</b> Business
<b>Interested Stakeholders:</b> Employee of Taitoa or the user of the application		
<b>Brief description:</b> This use case describe the output of the graphs and charts from the system to input field for variable by user. The goal of actor is to get the result of the dependent variables for that field in graphical format.		
<b>Goal:</b> The successful completion of calculation and display of new calculated value for dependent variable along with graph and chart.		

Table 5.2.2: General user create graph for input values

**General User Use Case - Update fields for simulation:**

<b>Use case name:</b> Update field for simulation	<b>ID:</b> 003	<b>Priority:</b> High
<b>Primary actor.</b> General User	<b>Source:</b> Employee	<b>Use case type:</b> Business
<b>Interested Stakeholders:</b> Employee of Taitoa or the user of the application		
<b>Brief description:</b> This use case describe the update to input field by user. The goal of actor is to update the result of the dependent variables for that field .		
<b>Goal:</b> The successful completion of calculation and update calculated value for de-		

pendent variable.

Table 5.2.3: General User update fields for simulation

**General User Use Case - Update graph and chart for updated values:**

<b>Use case name:</b> Update graph and chart for updated values	<b>ID:</b> 004	<b>Priority:</b> High
<b>Primary actor.</b> General User	<b>Source:</b> Employee	<b>Use case type:</b> Business
<b>Interested Stakeholders:</b> Employee of Taitoa or the user of the application		
<b>Brief description:</b> This use case describe the update of graph and charts to input field by user. The goal of actor is to update the result of the charts for its corresponding dependent variables. .		
<b>Goal:</b> The successful completion of calculation and update graphs for updated dependent variable.		

Table 5.2.4: General User update graph and chart for updated values

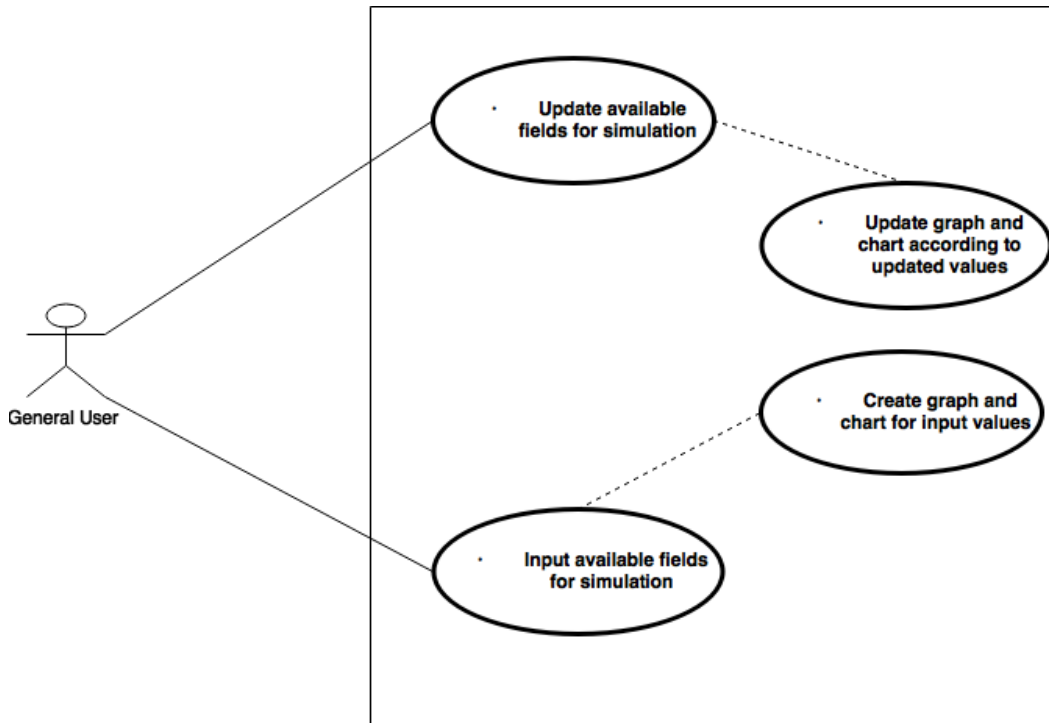


Figure 5.2.1: Use Cases diagram for general user

### 5.2.3 Use cases for administrative user

#### Admin User Use case -Import/Save new financial data to database:

<b>Use case name:</b> Import/Save new financial data to database	<b>ID:</b> 005	<b>Priority:</b> High
<b>Primary actor.</b> Admin User	<b>Source:</b> Employee	<b>Use case type:</b> Business
<b>Interested Stakeholders:</b> Employee of Taitoa as admin		
<b>Brief description:</b> This use case describe the import of past year financial data to the system for the simulation of the future. The goal of actor is to add the new financial year data to the system as the financial year pass by. .		
<b>Goal:</b> The successful completion of import data will create a new base year of financial data in the system.		

Table 5.2.5: Admin User -Import/Save new financial data to database

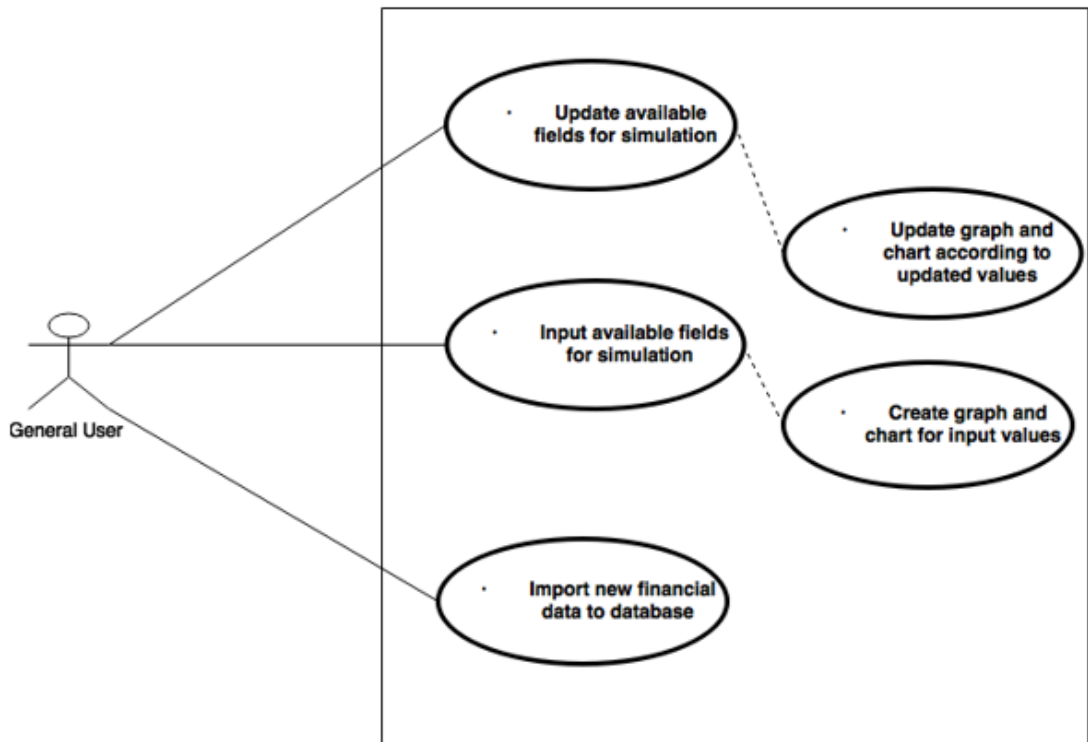


Figure 5.2.2: Use Cases diagram for Admin user

### 5.3 Data design

A simple definition for a database by a techterm.com states, "A database is a data structure that stores organized information" (Techterm, 2009). Data design is the core of the software development part. It provides a logical solution for the skewed data structure. Avoiding data redundancies, integration of table and data consistency can't be obtained without a proper data design.

#### 5.3.1 Entity relation diagram

An entity-relationship diagram (ERD) is a data modeling technique that graphically illustrates an information system's entities and the relationships between those entities. An ERD is a conceptual and representational model of data used to represent the entity framework infrastructure (Cory Janssen, 2010). The elements of an ERD are: Entities, Relationships and Attributes.



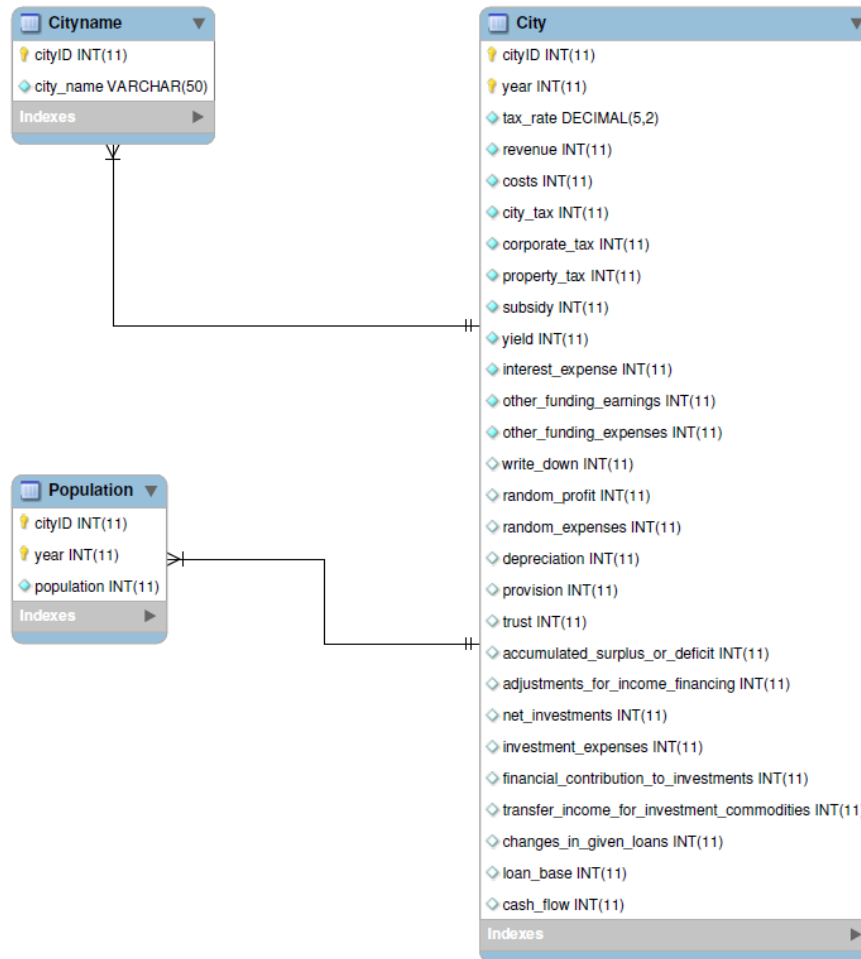


Figure 5.3.1: Entity Relation Diagram for existing database

The ER diagram (Figure 5.3.1) shows the three table existed in the database and their foreign key constraint with other table. The table in the database are: Population, City and Cityname.

### 5.3.2 Data Dictionary

The purpose of the data dictionary is to make a clear definition of the entities used during the development stage. It usually refers to metadata of the database table and its attributes. Technopedia defines the data dictionary as, “The data dictionary contains records about other objects in the database, such as data ownership, data relationships to other objects, and other data.”

The data dictionary for table City, Cityname and Population are attached in **appendix 1**, **appendix 2** and **appendix 3** respectively

## 5.4 User Interface Design

Bringing an idea of a website into its online status is not just a challenge of coding but also understanding the user themselves. There is no doubt that the backend is crucial and may involve user classified information. However, an end user is not really interested in this technical detail, what matters for them is only the presentation level of the application and its ease of usability.

The starting point of this is understanding the client need and interest. This need is transformed into the more matured prototyping with mock-up. There are three main UI views in the application.

- Welcome Page
- Game Simulation
- Data/Import

Firstly, the skeleton structure of game simulation is the main page where the business logic of the application is implemented. Its mock-up (Figure 5.4.1) looks as follows:

	[Logo] TALOUSPELI	
[Navbar]	Etusivu Kehikko	
	Valitse kaupunki <input type="text"/>	
	Taulukko <div style="border: 1px solid black; height: 150px;"></div>	
	----- <input type="text"/>	
	Kaavio <input type="text"/>	
[footer]		

Figure 5.4.1 Skeleton structure of UI for simulation page

Secondly, the simulation page consists of financial sheet and graphs as well, so the first skeleton structure should be transformed to more mature design where the financial data and graphs will be displayed. This view (Figure 5.4.2) will let the user select the municipality and all the financial data will be retrieved and calculated to display in this interface.

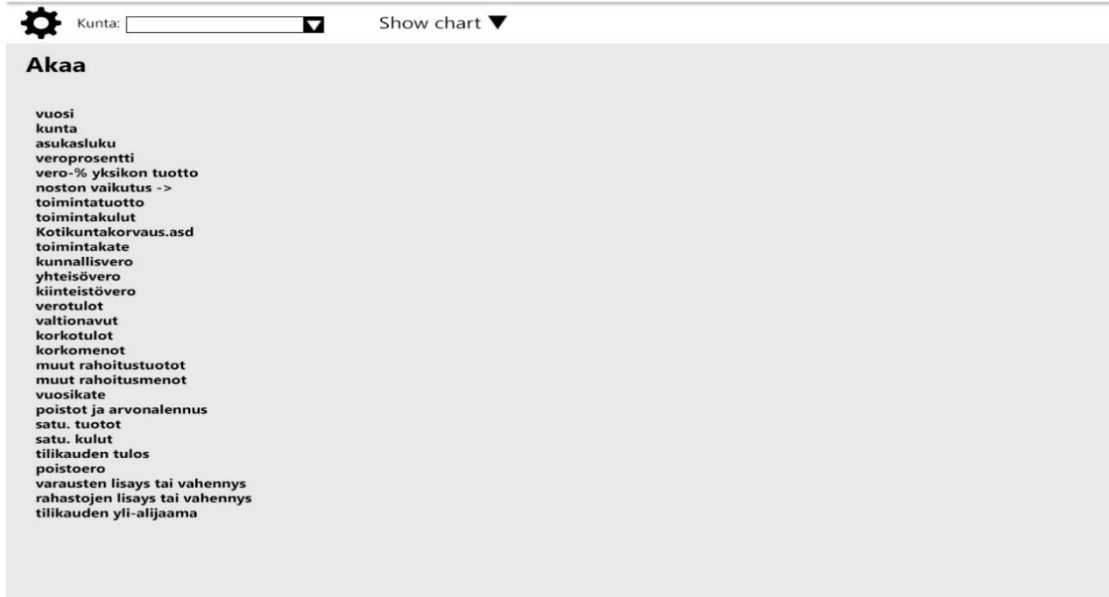


Figure 5.4.2 UI for selecting the municipality and displaying its financial data

Finally, when the municipality is selected the charts should be displayed along with the financial data. Thus the mock-ups for the charts in the game simulation page looks as figure 5.4.3.

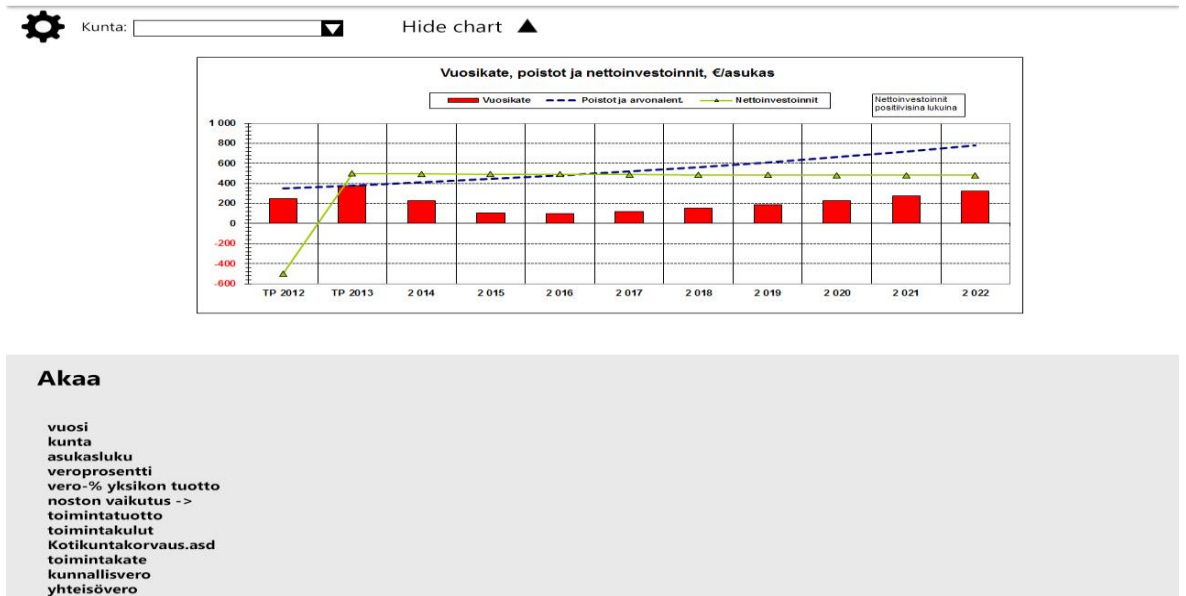


Figure 5.4.3 UI mock-up for chart in game simulation page

Along with the above mentioned interface design the application will consist of the welcome page where the user will have a brief information about the project, tools and com-

pany introduction. Also, at the end phase of the project development a new UI page for the data import will be added as well.

## 6 Implementation

This part of the paper will document the process of implementing the project “Taitoa”. The basic work will be transforming the current version of excel sheet provided by the finance ministry into the web version. The development process will follow a Scrum methodology, however, system requirement document and system design specification will also be prepared during the process.

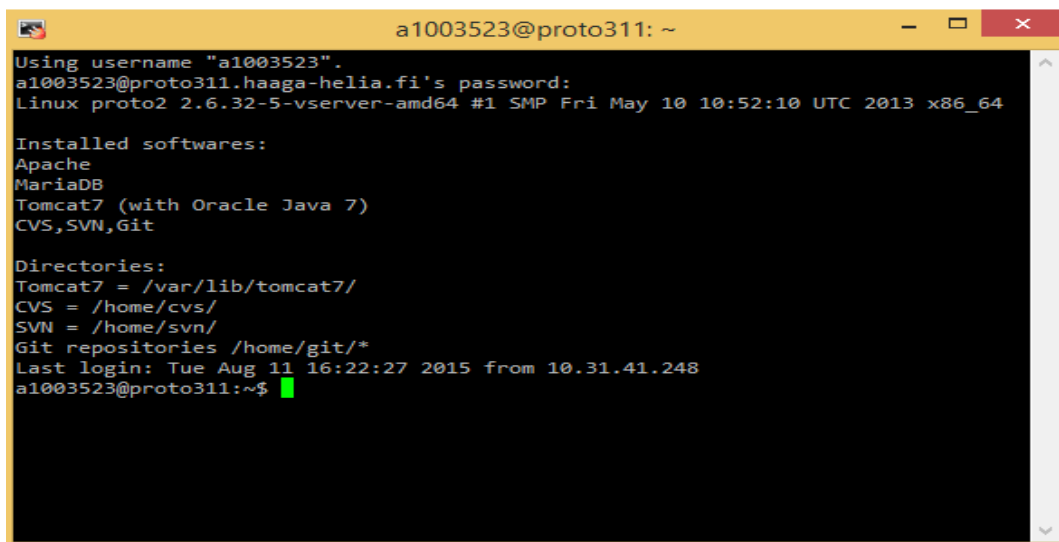
### 6.1 Setting development environment

Team work and project management are one of the most essential part of the software development which insure the goal achievement process. The first steps for the start of project is setting the development environment. The following tools were used during the project:

#### 6.1.1 Proto server

The project was done with the co-ordination of Haaga-Helia UAS which provided us with the proto server and access right given to all the members with a valid student number and password. The server could be accessed within the Haaga-Helia network as well as from the SSH connection out of the school environment.

The figure 6.1.1 below show how the console environment of proto server looks with the Putty tool for the user a1003523. The following figure also shows the installed software like the Apache web server, Maria DB relational database and Git version control.



```
a1003523@proto311: ~
Using username "a1003523".
a1003523@proto311.haaga-helia.fi's password:
Linux proto2 2.6.32-5-vserver-amd64 #1 SMP Fri May 10 10:52:10 UTC 2013 x86_64

Installed softwares:
Apache
MariaDB
Tomcat7 (with Oracle Java 7)
CVS,SVN,Git

Directories:
Tomcat7 = /var/lib/tomcat7/
CVS = /home/cvs/
SVN = /home/svn/
Git repositories /home/git/*
Last login: Tue Aug 11 16:22:27 2015 from 10.31.41.248
a1003523@proto311:~$
```

Figure 6.1.1 Console environment of Proto Server from Putty Tool

## 6.1.2 Git repository

'Taitoa' as web application project strongly advocates the power of open source software and its reliability to be used for commercial software development. With the similar fundamental git provides an environment among the open source community to coordinate and contribute the development work from any part of the world.

On the other hand, gits insures the committed project work can be reverted, code can be merged, undo mistakes and deletion. The git was installed in the server environment and all the project member were added as a user. The project can be cloned by members with the following command: `Git clone aXXXXXXX@proto311:/home/git/talouspeli.`

## 6.1.3 Pivotal tracker

Scrum or agile method of development usually has sprint, backlog and tasks as breakdown for simplicity, there are project management tools which support this process. In Taitoa project pivotal tracker was the official project management tool. The license was available for Haaga-Helia UAS.

This tool allows to create the sprint which would include different backlog. Backlog was a further breakdown into tasks and allocated to a different developer. The tasks could be tagged with different status like start, ongoing and deliver. The progress could be tracked with burn down charts. The figure 6.1.3 below shows the screenshot of pivotal tracker filled with backlog and tasks.

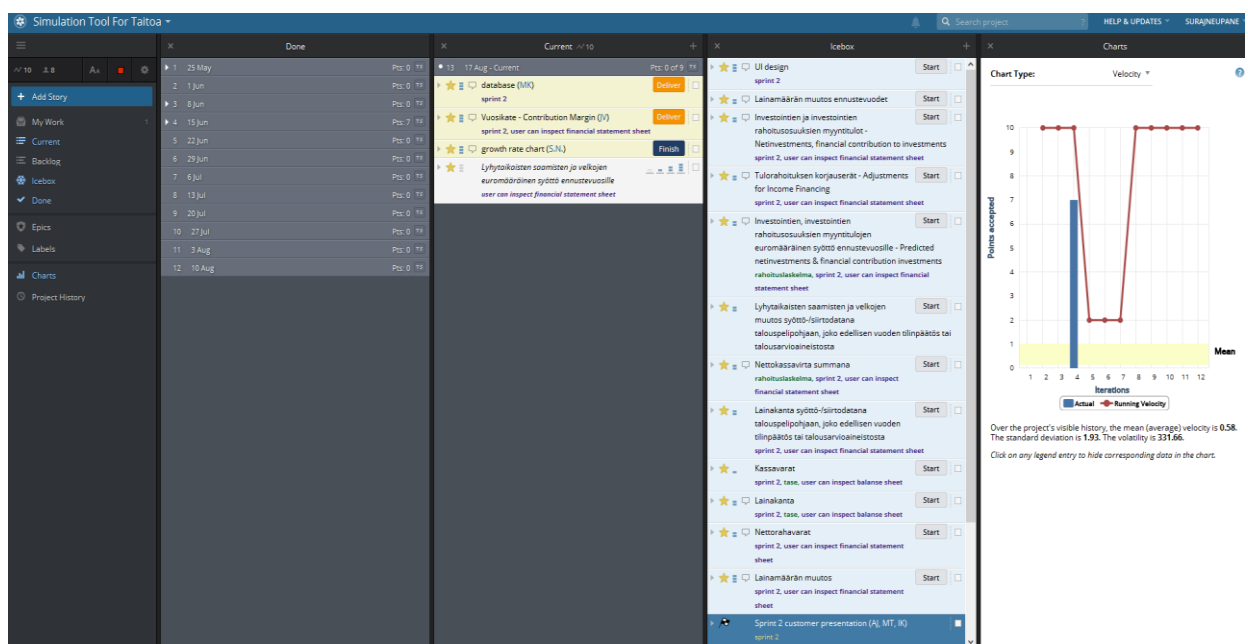


Figure 6.1.3: Pivotal tracker screen with burn down chart, tasks and backlog.

## 6.1.4 Slack

Communication between the team is the vital in any project and the tool used to exchange the information should be easy to use and reliable. Slack as the messaging application for the team had all the requirement for our project. A new domain `taitoa.slack.com` for our project was created and all the members were added to the project domain. The different topic like backend, frontend and general were created and the user could add themselves to the following forum. A team member was also able to send private message to another. The structure of different communication forum for `taitoa.slack.com` can be seen in the figure 6.1.4 below:

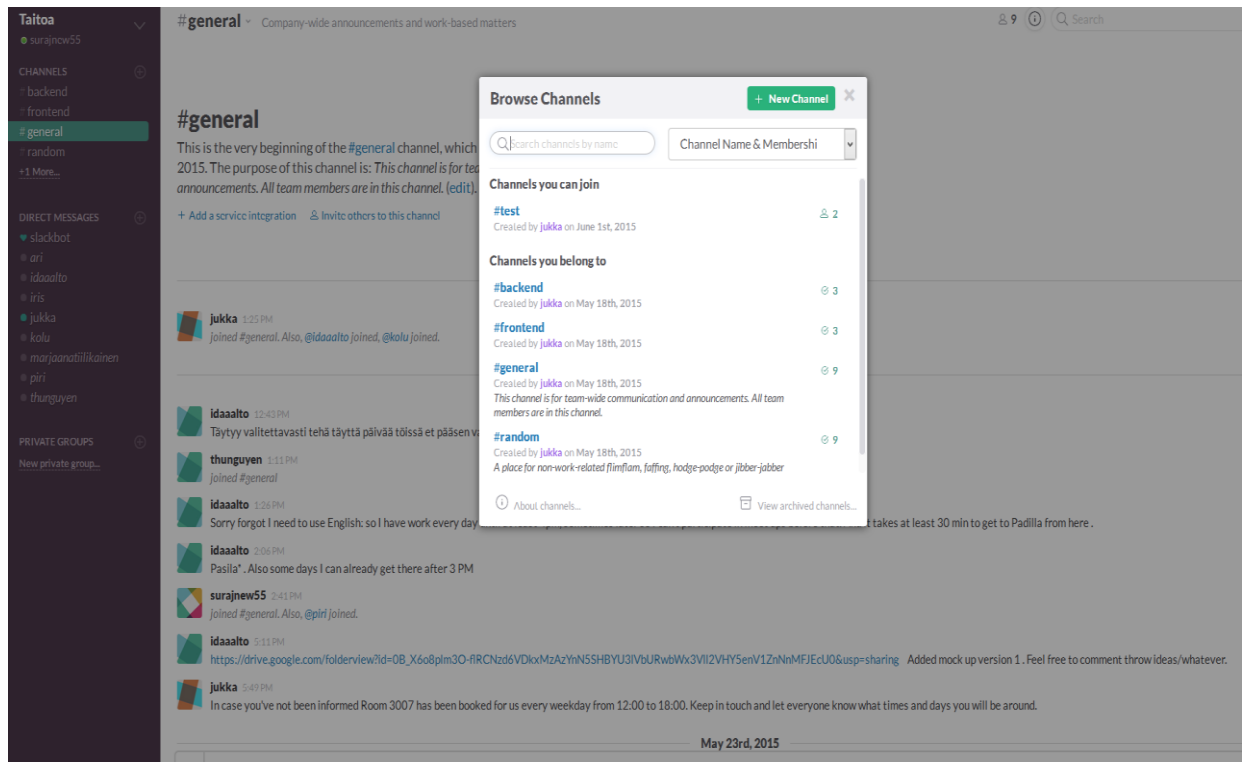


Figure 6.1.4 Screenshot for the team messaging application Slack with project domain `taitoa.slack.com`

## 6.2 Backend

The Taitoa web application is based on SPA (Single Page Application) technology. It only calls the server once for retrieving the data from the database. But, the rest of the frontend is handled by some kind of JavaScript library or framework. Similarly, in our application

AngularJS manages the frontend and calls the server for data at the very beginning. Figure 6.2.1 shows how the AngularJS code calls the backend via AJAX technology when the ng-app is initialized. If the server sends the response and the transaction is successfully the cities are populated on the view via \$scope data-model.

```
var app = angular.module('myApp', ['ui.bootstrap']);

app.controller('customersCtrl', function ($scope, $http, $timeout) {
    $http.get("http://proto311.haaga-helia.fi/json4.php")
        .success(function (response) {
            $scope.names = response.cities;
            $scope.population = response.population;
            $scope.cityname = response.citynames;
            $scope.current_year = response.current_year;
        });
});
```

Figure 6.2.1 HTTP call on the server via AJAX technology by AngularJS

The \$http get request on `http://proto311.haaga-helia.fi/json4.php` as shown in figure 5.4 triggers the connection to the database. `Jstone4.php` opens the connection by providing the authentication for Maria DB as shown in figure 6.2.2 below:

```
<?php
header("Access-Control-Allow-Origin: *");
header("Content-Type: application/json; charset=UTF-8");
$conn = new mysqli("localhost", "taitoa", "taitoa", "taitoa");
$conn->set_charset("utf8");
$sql = "SELECT
        cn.city_name, c.*,
        c.year, p.population
        FROM City AS c
        LEFT JOIN cityname AS cn ON c.cityID = cn.cityID
        LEFT JOIN Population AS p ON p.cityID = c.cityID AND p.year = c.year
        ORDER BY city_name";
```

Figure 6.2.2 Connection string for the database through PHP file

If the above connection is established and the SQL query is run on the database, it retrieves the city, population and city name with the proper join according to the relation defined. The successful response of this query and response for \$http.get request from angular looks as in figure 6.6 as below. As in figure 6.2.3 JSON array is as the response of the server on call to the json4 page which connects the database and query for all the cities with array of objects as in figure below:



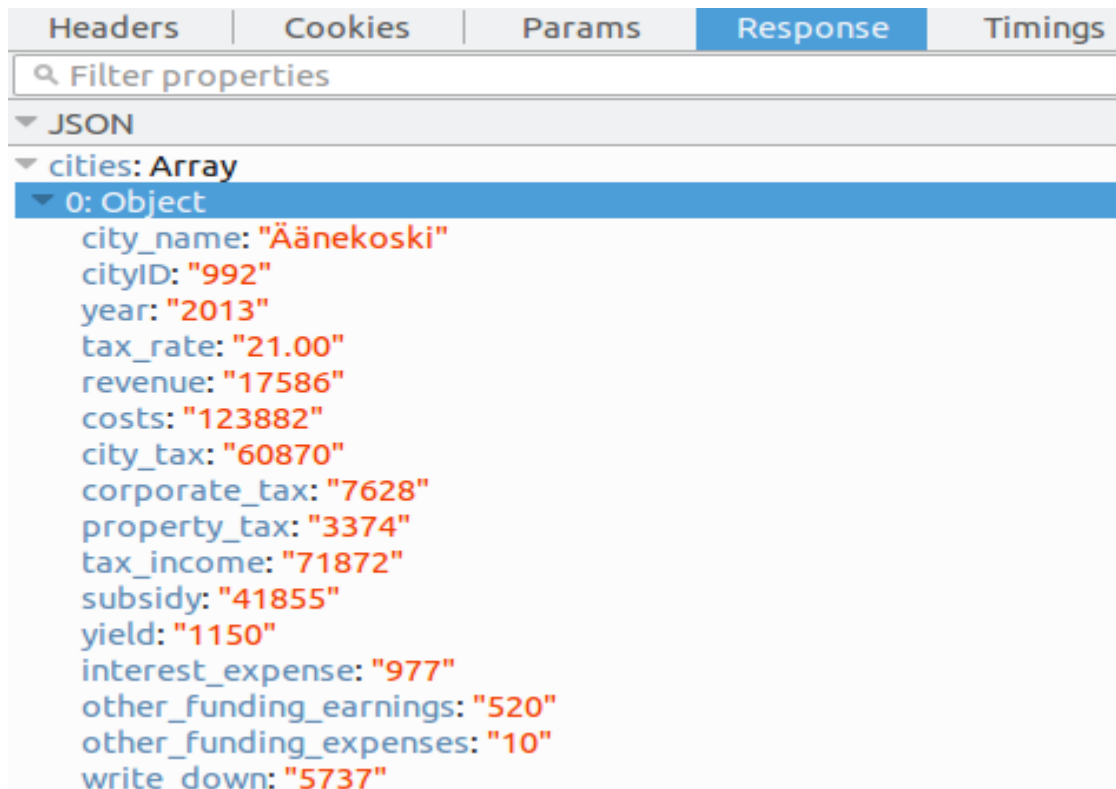


Figure 6.2.3 Successful response of server to the `$http` call from AngularJS.

### 6.3 Frontend

After the successful connection to the server and getting back the response with JSON array data AngularJS is responsible for managing the data and integrating it with HTML. For allowing code on HTML page define angular scope for myApp on view: `div ng-app="myApp" ng-controller="customersCtrl" class="container"`.

Also the different library or framework should be included in the project. In Taitoa the following library and framework have been used and included as a different CDN (Content Distribution Network) as in figure 6.3.1 below.

```
</script>
<script src="jquery-2.1.4.min.js"></script>
<!-- Angularjs -->
<script src="angular.js"></script>
<!-- Custom Javascript -->
<script src="app.js"></script>
<!-- Custom Stylesheet -->
<link rel="stylesheet" href="style.css">
<!-- Bootstrap -->
<link rel="stylesheet" href="bootstrap-3.3.4-dist/css/bootstrap.min.css">
<link rel="stylesheet" href="//maxcdn.bootstrapcdn.com/font-awesome/4.3.0/css/font-awesome.min.css">
<script src="bootstrap-3.3.4-dist/js/bootstrap.min.js"></script>

<!-- Charts -->
<script src="../bower_components/angular-chart/angular-chart.min.js"></script>
<script src="../bower_components/angular-bootstrap/ui-bootstrap.min.js"></script>
<link rel="stylesheet" href="../bower_components/c3/c3.css">
<script src="../bower_components/d3/d3.min.js"></script>
<script src="../bower_components/c3/c3.min.js"></script>
<script src="dom.js"></script>
```

Figure 6.3.1 all the library and framework included in project.

The further images of frontend development have been included as attachments.

## **7 Conclusion and Result**

Taitoa project will deliver the web application as the need of the client, as well as the project is the use of open source software which is the principal foundation of the system. Furthermore, the whole project will be done with the active involvement of Haaga-Helia UAS student.

The current version of the system works on the excel sheet stored on local computer and with no dynamic and modular feature. In contrast, the new system will be able to be accessed via internet as web application which results on the ease of accessibility, elegant UI design and modular features. On the same time, data import feature will be available to the user who can add the new fiscal year base data on the system.

Open source application development is the principal of the Taitoa project, meaning all the resources and technology needed for the project will be using the open source platform and tools. Recently, open source technology have matured enough to provide a strong base for the web development support. For illustration, LAMP stack was used to develop Facebook at its beginning stage which is one of the most visited site.

Likewise, education system are the backbone for the job market for preparing skilled worker, so the project like Taitoa helps university administration for ensuring the study curriculum are up-to-date with the need of market. Under the direct supervision of teachers Taitoa gave an opportunity for its member developer as experience of real working life environment.

Along with skills, project management and Scrum as Agile development was quite new experience for the team member. Scrum method of development works smooth with the experience of the methodology increases.

### **Result**

Till the end of first sprint, the development work have progressed to implement 70% of business logic and working level of basic UI design . There were some challenges during the start for choosing the right technology and fine tuning between the development members and evaluating their skills and understanding their role and contribution to the project.

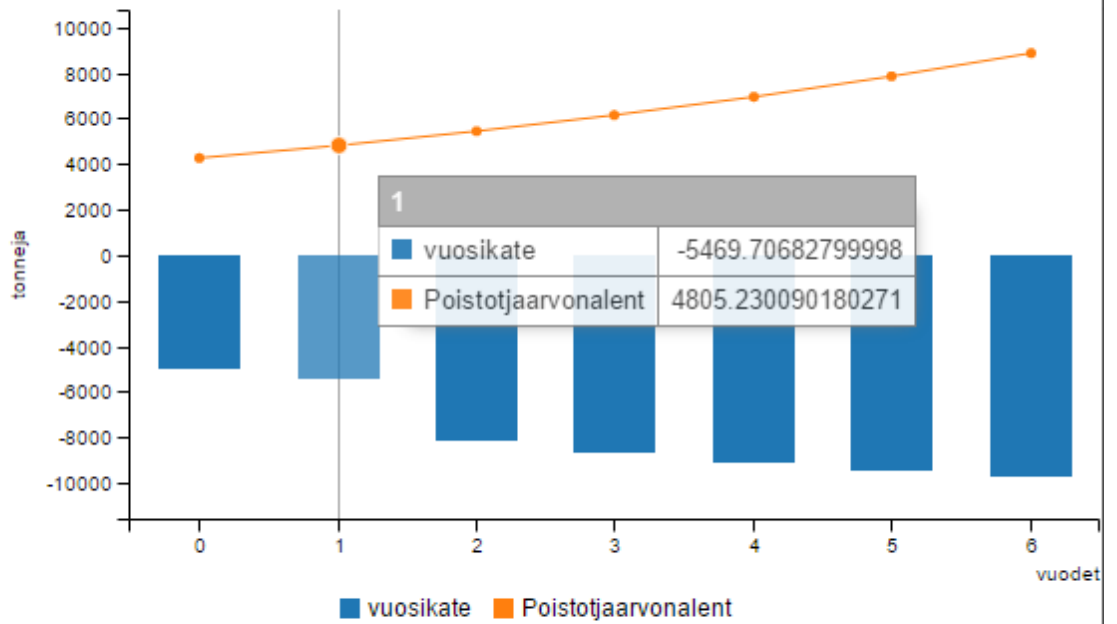
Despite the challenges, the second sprint is on its half way where the UI design and remaining part of the business logic will be implemented. The client demand for 'Game like

Simulation' of application will be implemented on this sprint. Along with the UI design, the testing team will start unit testing to ensure the integrity of the application.

Finally, the third and last sprint will ensure the testing part, fixing minor issue and giving the final touch for the application. The delivery of the application to the client will be done in this sprint.

## 8 Attachments

### Graphical Representation



Close

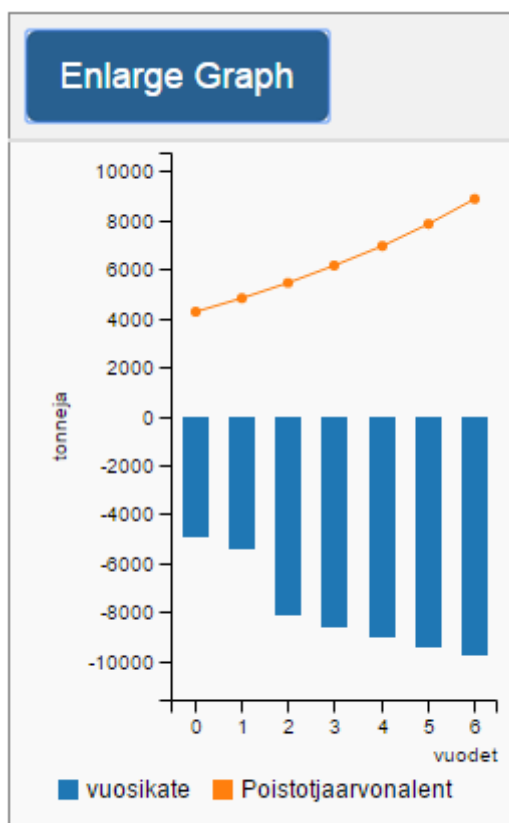
### Dashboard

Valittu kaupunki on:

Akaa

TULOSLASKELMA									
Vuosi	2013	2014	2015	2016	2017	2018	2019	2020	2021
Asukasluku	17 108	17 052	17 402	17 486	17 570	17 655	17 745	17 835	17 925
VEROPROSENTTI ↓	20,50	21,00	21,25	21,25	21,25	21,25	21,25	21,25	21,25
TOIMINTAKATE ↓	-85 910	-89 753	-93 289	-96 148	-98 330	-101 074	-103 744	-106 485	-109 246
VEROTULOT ↓	57 109	57 963	58 950	60 177	61 637	63 865	66 501	69 247	72 100
VALTIONOSUUDET	32 218	30 594	29 489	30 610	28 681	28 652	28 280	27 912	27 544
VUOSIKATE ↓	3 297	-1 304	-4 958	-5 470	-8 119	-8 665	-9 071	-9 433	-9 791
TILIKAUDEN TULOS ↓	-33	-5 067	-9 210	-10 275	-13 549	-14 801	-16 005	-17 269	-18 533
TILIKAUDEN YLI-ALIJAAMA ↓	1	-5 035	-9 178	-10 243	-13 517	-14 769	-15 973	-17 237	-18 501
TASEESTA									
kertynyt yli-/alijäämä	-207	-5 243	-14 421	-24 664	-38 182	-52 950	-68 923	-86 160	-104 651

Tuloslaskelman erien muutos prosentti							
vuosi	2015	2016	2017	2018	2019	2020	2021
Toimintatuotot	2.2	1.7	2.9	2.8	2.9	2.9	2.9
toimintakulut	3.3	2.8	2.4	2.8	2.7	2.7	2.7
kunnallisverot	1.8	2.8	2.3	3.6	4.2	4.2	4.2
yhteisöverot	-3.6	-10.1	5	4.7	4.3	4.3	4.3
kiinteistovero	8.5	1.2	2.4	2.4	1.7	1.7	1.7
valtionosuudet	-3.6	3.8	-6.3	-0.1	-1.3	-1.3	-1.3



## References

Art & Logic, 2015: Creation date March 5, 2015

URL: <http://artandlogic.com/how-to-choose-the-right-technologies-for-your-software-project/>, Accessed June 23, 2015.

The internet of thing, 2015: Author: Marissa Tejada, Created at 13/9/2014

URL: <http://www.theinternetofthings.eu/marissa-tejada-internet-things-convergence-technologies>, Accessed June 23, 2015.

John Garger, 2010: Updated at 11/9/2010

URL: <http://www.brighthub.com/computing/linux/articles/62137.aspx>, Accessed June 23, 2015.

Felicia G., 2013: Methodologies Used in Open Source Approach to Developing Software in Companies in Finland, Bachelor's thesis. Haaga-Helia UAS. Helsinki, URL:

[http://www.theseus.fi/bitstream/handle/10024/67601/Jaaskelainen\\_Felicia\\_Thesis.pdf?sequence=1/](http://www.theseus.fi/bitstream/handle/10024/67601/Jaaskelainen_Felicia_Thesis.pdf?sequence=1/), Accessed June 25, 2015.

Richard Stallman, the GNU Project: Author: Richard Stallman,

URL: <http://www.gnu.org/gnu/thegnuproject.html>, Accessed June 26, 2015.

The License is the product 1998: Author: Robert W. Gomulkiewiz, Article: The License Is the Product: Comments on the Promise of Article 2B for Software and Information Licensing, P.896. URL: <http://cyber.law.harvard.edu/property00/alternatives/reading1a.html>, Accessed June 28, 2015.

Phillips, Douglas E., 2009: Author: Phillips, Douglas E. Title: Software License Unveiled: How Legislation by License Controls Software Access. Cary, NC, USA: Oxford University Press, USA, 2009. ProQuest ebrary. Web. 29 June 2015.

Eric S. Raymond. Title: The Cathedral and the Bazaar. O'Reilly Media, 2001

FLOSS Research and Development FLOSS'07: Accessed June 29, 2015 URL:

<http://www.computer.org/web/csdl/index/-/csdl/fullIssue/citation/toc/asciitext/list/pfloss2007296100.txt>

VersionOne, Inc. 2014: 8th Annual State of Agile Survey 2014. VersionOne, Inc. URL: <http://www.versionone.com/pdf/2013-state-of-agile-survey.pdf>. Accessed: July 3, 2015

Bushra Mostafa, 2015: Topic: Assessing the applicability of Traditional and Agile methodology to enhance success of software development projects, Bachelor thesis, Haaga-Helia UAS, Helsinki. URL: [https://www.theseus.fi/bitstream/handle/10024/89299/Mostafa\\_Bushra.pdf?sequence=1](https://www.theseus.fi/bitstream/handle/10024/89299/Mostafa_Bushra.pdf?sequence=1), Accessed: August 17, 2015

Dooley, J. 2011: Title: Software Development and Professional Practice. Apress Publication. ISBN-13 (electronic): 978-1-4302-3802-7. URL: <http://it-ebooks.info/book/587/>, Accessed: August 17, 2015

Figure 3.3.1 The waterfall model, 2015: Created: February 5th 2014 URL: <http://www.oxagile.com/company/blog/the-waterfall-model/>, Accessed August 18, 2015.

Schwaber K. & Beedle M. 2002. Agile Software Development with Scrum. Pearson Education International. Prentice-Hall, Inc. Upper Saddle River. ISBN: 0-13-207489-3

T. O'Reilly, 2005: "What is web 2.0," Created: September 2005. Available: <http://oreilly.com/web2/archive/what-is-web-20.html>, Accessed July 7, 2015

Jan Varwig 2009, A Thin Web Application Framework based on Server-side JavaScript, Diploma Thesis, University of Rostock. Rostock, Germany. URL: <http://jan.varwig.org/wp-content/uploads/2009/10/diploma-thesis-jan-varwig-serverside-javascript.pdf>, Accessed: July 11, 2015.

Changpil Lee 2012, An Evaluation Model for Application Development Frameworks for Web Applications, Ohio State University, URL: [https://etd.ohiolink.edu/!etd.send\\_file?accession=osu1324663059&disposition=inline](https://etd.ohiolink.edu/!etd.send_file?accession=osu1324663059&disposition=inline), Accessed on 13 July, 2015.

AS, 2014: Article: How to choose the right Technology Stack for your Startup?, Available: <http://okyasoft.blogspot.fi/2014/04/how-to-choose-right-technology-stack.html>, Accessed 15<sup>th</sup> July, 2015.

Quentin Hardydec, 2013: Article: Google Joins a Heavyweight Competition in Cloud Computing. URL: <http://www.nytimes.com/2013/12/03/technology/google-joins-a-heavyweight->



[competition-in-cloud-computing.html?pagewanted=2&adxnnl=1&smid=tw-share&adxnnlx=1386083129-SZzrXO68IC136ObY2I9CSw&\\_r=0](http://www.php.net/competition-in-cloud-computing.html?pagewanted=2&adxnnl=1&smid=tw-share&adxnnlx=1386083129-SZzrXO68IC136ObY2I9CSw&_r=0)

The PHP Group, 2015: The official site for PHP, Copyright © 2001-2015 The PHP Group  
URL: <http://php.net/>, Accessed: 27<sup>th</sup> July 2015.

Lengstorf, J. 2009. PHP for Absolute Beginners. Apress. United States of America

Grace Smith 2014: Article: 13 PHP Frameworks to Help Build Agile Applications  
URL: <http://mashable.com/2014/04/04/php-frameworks-build-applications/>, Accessed: 14<sup>th</sup> July 2015.

Bill Ready, CEO of Braintree Payment Solutions, 2012. Article: How Mobile Devices Will shake Up the E-commerce Business. URL:  
<http://www.forbes.com/sites/ericsavitz/2012/10/03/how-mobile-devices-will-shake-up-the-e-commerce-business/>, Accessed 20<sup>th</sup> June 2015.

Sommerville, Ian & Pete Sawyer. (1997). Requirements Engineering: A good practice guide. West Sussex, England: John Wiley & Sons.

Figure 3.3.2: Scrum process with product backlog, sprint backlog and Sprint  
URL: [https://en.wikipedia.org/wiki/Scrum\\_\(software\\_development\)#/media/File:Scrum\\_process.svg](https://en.wikipedia.org/wiki/Scrum_(software_development)#/media/File:Scrum_process.svg)

Figure 3.4.1: Market share of different programming language used today Source:  
<http://www.w3cook.com/programminglanguage/php>

Figure 3.4.3: Bootstrap grid system with scale in different size of screen Source:  
<http://getbootstrap.com/css/#grid-options>

Figure 3.4.4 Interest over time for big four frontend framework URL:  
<http://www.google.com/trends/explore?hl=en-US#q=ember+js.+angular+js.+backbone+js.+react+js&cmpt=q&tz&tz>

AngularJS, 2015: Official site for AngularJS URL:  
<https://docs.angularjs.org/guide/introduction>, Accessed 14<sup>th</sup> August, 2015.

IEEE STD 1016-1998: URL: [http://www.atilim.edu.tr/~dmishra/se112/sdd\\_template.pdf](http://www.atilim.edu.tr/~dmishra/se112/sdd_template.pdf),  
Accessed: 20<sup>th</sup> June, 2015.

Maria DB, 2015: Official site for Maria DB, URL: <https://mariadb.org/en/about/>

Margaret Rouse, 2007: Used case definition,  
URL:<http://searchsoftwarequality.techtarget.com/definition/use-case>, Accessed 20<sup>th</sup> August, 2015.

Techterm, 2009: Definition of Database, URL: <http://techterms.com/definition/database>, Accessed 20<sup>th</sup> August, 2015.

Cory Janssen, 2010: Techopedia definition for entity relation, URL:  
<http://www.techopedia.com/definition/1200/entity-relationship-diagram-erd>, Accessed 19<sup>th</sup> August, 2015.

Rachel S. Smith, Senior Interface Designer, CSU Center for Distributed Learning,  
Available at: <http://www.cdl.edu/uploads/Qd/S6/QdS615B1DcnwRZInSuTDnQ/writing-requirements.pdf>

Opensource.org URL: <http://opensource.org/> Accessed June 23, 2015.

<http://www.apache.org/foundation/>, Accessed June 29, 2015

<http://www.apache.org/foundation/>, Accessed June 29, 2015

## Appendices

### Appendix 1. Data Dictionary for City table present in database

City table					
Column	Description	Data type	Required	Key	Special domain
<b>cityID</b>	Unique city identity	Integer	Absolute	Primary Key	Positive integer
<b>year</b>	Corresponding year for data	Integer	Absolute	-	Positive Integer
<b>tax_rate</b>	rate of tax for that fiscal year	Decimal	Absolute	-	Positive Decimal
<b>revenue</b>	revenue for the municipality	Integer	Absolute	-	Positive integer
<b>costs</b>	cost for the municipality	Integer	Absolute	-	Positive Integer
<b>city_tax</b>	city tax for that fiscal year	Integer	Absolute	-	Positive Integer
<b>corporate_tax</b>	corporate tax of the fiscal year	Integer	Absolute	-	Positive Integer
<b>property_tax</b>	property tax of the fiscal year	Integer	Absolute	-	Positive Integer
<b>subsidy</b>	subsidy provided by the city	Integer	Absolute	-	Positive Integer
<b>yield</b>	income return on an investment	Integer	Absolute	-	Positive Integer
<b>interest_expense</b>	interest on	Integer	Absolute	-	Positive Integer

	the loan	ger	lute		ger
<b>other_funding_earning</b>	earning from other source	Integer	Absolute	-	Positive Integer
<b>other_funding_expenses</b>	expenses on other different heading	Integer	Absolute	-	Positive Integer
<b>write_down</b>	loss in value of assets	Integer	-	-	Positive Integer
<b>random_profit</b>	profit from random sources	Integer	-	-	Positive Integer
<b>random_expenses</b>	expenses from random sources	Integer	-	-	Positive Integer
<b>depreciation</b>	reduced value of assets	Integer	-	-	Positive Integer
<b>provision</b>	expenses set aside for bad loan	Integer	-	-	Positive Integer
<b>trust</b>	assets for the benefit of third party	Integer	-	-	Positive Integer
<b>accumulate_surplus_or_deficit</b>	total of all the earning, profit, cost and loss	Integer	-	-	Positive/negative Integer
<b>adjustments_for_income_financing</b>	fund allocated for dependents of primary earner	Integer	-	-	Positive Integer
<b>net_investments</b>	total	Integer	-	-	Positive Integer

	amount of investment	ger			ger
<b>cash_flow</b>	total amount of cash register in a/c	Integer	-	-	Positive Integer

**Appendix 2. Data Dictionary for Cityname table present in database**

<b>Cityname table</b>					
<b>Column</b>	<b>Description</b>	<b>Data type</b>	<b>Required</b>	<b>Key</b>	<b>Special domain</b>
<b>cityID</b>	Unique city identity	Integer	Absolute	Primary key/Foreign key	Positive integer
<b>city_name</b>	name of city related to financial data	VarChar(50)	Absolute	-	String

**Appendix 3. Data Dictionary for Population table present in database**

<b>Population table</b>					
<b>Column</b>	<b>Description</b>	<b>Data type</b>	<b>Required</b>	<b>Key</b>	<b>Special domain</b>
<b>cityID</b>	Unique city identity	Integer	Absolute	Primary key/Foreign key	Positive integer
<b>year</b>	name of city related to financial data	VarChar(50)	Absolute	Primary key/Foreign key	Positive Integer