University of Glasgow | Department of Computing Science

A Website Translation Service

Alasdair Campbell
Andrei Mustata
Paul Moore
Stephen Hayton
Wei Zhang

Level 3 Project — 28th February 2012

Abstract

Since the writing of the earliest manuscripts and documents there has been a need for translation so that the people of the world could understand what had been written or said, in their own native language. Translation is just as relevant today as it was thousands of years ago, although the processes have modernised considerably. The focus of this project is to create an online presence and document delivery system for a Glasgow based translator who is creating her own document translation business. Our goal is to improve upon the translation sites currently available and to give a unique, modern and fresh feel to translation.

Education Use Consent

We hereby give our permission for this project to be shown to other University of Glasgow students and to be distributed in an electronic format. **Please note that you are under no obligation to sign this declaration, but doing so would help future students.**

Name: Alasdair Campbell Signature: _____

Name: Andrei Mustata Signature: _____

Name: Paul J Moore Signature: _____

Name: Stephen Hayton Signature: _____

Name: Wei Zhang Signature: _____

Contents

Chapter 1

Introduction

This article assumes a basic general computing knowledge from the reader. If any words or phrases are not understood, consult the glossary of terms (Section A) at the end of the document. All content is owned by the authors of this document or is otherwise referenced in bibliography (Section **??***)*

1.1 Background

As part of our degree in third year we are tasked with a team project. This relates to computing disciplines new to us and draws on knowledge gained from the preceding two years at University. Teams were randomly assembled at the beginning of semester and our team received the project to create a website providing a document translation service for a real client. It was a pleasing allocation mainly due to the latter part of the task: the fact we would be working with a real client. This bespoke web-based service builds on our client's existing business model, with the specific aims to bring about an expansion in her company, while also facilitating an improvement to her working practice as a whole.

There are many translation services available on the Internet already, a simple Google search for online translation returns around one hundred and fifty nine million results. We have looked into various different types of translation and interpretation websites during our research and have found pros and cons from each. This vast number of already available websites creates a desire of competing with what's already available, by trying to improve areas where other websites have fallen short. We believe one of the key factors that makes a modern website successful is minimalistic design: it offers simple and effective functionality and is aesthetically pleasing. The combination of these things means users are more likely to use the website after stumbling across it in a search, perhaps, and will ultimately give our client a larger customer base. To clarify, we are not trying to re-invent the wheel with our project. We have used frameworks and other free source components in the development of our website. The system is mainly built around the LAMP structure, i.e., Linux + Apache + MySQL + PHP. We believed that these choices would lower the costs of an eventual upgrade of the system, as opposed to having used some more exotic technologies that come at the cost of losing the robustness and stability that we needed. We aimed to maintain a user friendly feel and look and believe that is something we accomplished well. One of the things that encourages

this notion is the fact we have created a simple 3-stage process in which users can register, upload documents, and request languages to translate to.

1.2 Aims

To briefly summarise our task at hand: we are to develop a website for a free-lance translator. It should allow customers to upload documents, request one of the available languages to translate to, and submit a request for translation. The translator should then be able to review those submitted documents and send a quote to the customer for the job to be translated. If happy, the customer should then be able to pay for the job(s) via Paypal, and then receive their translated document after a specified period of time. Additional required features of the website will be discussed later. Listed below are a few important points on why using a software system to manage documents and to keep track of discussions and tasks has advantages over the current manual system:

- It makes **planning** deadlines easier, since computers are better than humans at taking large amounts of data and comparing, adding, etc.;
- It makes **file management** easier;
- It improves **productivity** - due to the two above points, the translator now has an organised system and has more time to spend doing the actual translation.

By having all the job details clearly stated and planned, the customers are encouraged to pay more attention and to take it more seriously. It also introduces the notion of accountability on both sides - customer and translator. Having a detailed log of all the documents delivered to customers makes keeping track of all the activities easier to do. By using an electronic payment system, the risk of having unpaid jobs is eliminated. It also gives the business a more professional look and makes our client less vulnerable to hacking attacks as opposed to other methods of payment. As part of the specification requirements, our website is required to support Paypal for payment of translated document. To provide a short brief of this service: ”PayPal allows any business or consumer with an email address to securely, conveniently and cost-effectively send and receive payments online ” - *www.paypal.com/about*. This setup provides the translator with a feature that will not only provide her with security but also will encourage her customers to use the site, since Paypal has a massive online presence with excellent reputation. It is seen as the medium of paying for goods online, and this helps contribute to a modern, reliable website that we can develop. The translator, therefore, does not need to be concerned with things like credit card verification, since this is all functionality covered by Paypal. This maintains a level of separation between herself and her customers, so that she absolutely knows if her work has been paid for before she commits to it.

1.3 Motivation

An essential part of our requirements gathering process was the first meeting with the translator, Joelle Cimatche. Our team had been forewarned by our supervisor that the translator was not very

”technically minded”, so we tried our best to prepare questions that did not assume she had much experience with a computer. It’s fair to say that after reading the initial brief for the project we made assumptions as to the needs of the translator, addressing the problem from our viewpoint. Realising we should expect to review these assumptions during the process of working on the requirements capture is a valuable lesson for future employment. Working with an individual or organization, firmly on the periphery of Internet technology adds it’s own challenges, particularly during the requirements capturing process. At the meeting, the translator explained she had a very basic understanding of word processing applications and the Internet, and not much else. This presented us with an additional challenge. We couldn’t simply relay technical jargon to the translator and expect her to provide useful feedback. Not only that, our team would have to create a very easy-to-use admin end to the website that she could master easily.

As we advanced the development of the website, we would have to be very clear and straightforward when updating her on our progress. We accomplished this mainly through emails containing concise, clear and jargon free status updates on the website progress. These updates had been sent at important stages of development such as the transition from the design inception to the early implementation. The reason for this is that we want to build something that the client wants, not something that we think the client wants. When we felt there was a need to ask the client about her opinion on some certain aspect of the website, we emailed her without hesitation and she was quite often happy to accommodate the new changes. It seemed that she was prepared to grant us some degree of trust with creating her website, which certainly helped, but we still sought her opinion of most changes that took place. Arguably then, communication with our client has been crucial to the design of this website. Our communication was put to the test when we had to get her to sign up for her own web server. We had to prepare a lengthy, detailed and broken down walk through of how to purchase website hosting. For an average user of the Internet, this task would have been far less demanding. However, our client’s expertise in this domain required far more patience. **Figure 1.1** somewhat illustrates the challenge we have. It is basically the ”adoption rate curve” for a percentage of users of new technologies. In other words, how quickly users are prepared to adopt a new technology after it is first released. It is known as Roger’s bell curve. Our client would be at the far right end of the spectrum, in the 16% of users described as ”Laggards” in the graph.

Another perspective to consider is how users actually come to accept and use a new technology, and the factors that influence such a decision. **Figure 2.2**, otherwise known as the Technology Adoption Model (TAM) provides a graphical realisation of this. [1] The model suggests that when users are presented with a new technology, a number of factors influence their decision about how and when they will use it, notably:

- Perceived usefulness (PU) - This was defined by Fred Davis as ”the degree to which a person believes that using a particular system would enhance his or her job performance”.
- Perceived ease-of-use (PEOU) - Davis defined this as ”the degree to which a person believes that using a particular system would be free from effort” (Davis 1989).

From our early meetings with our client, we believe we can make well informed assumptions about these two aspects. Since the client has especially contacted the University to ask for such a website

[1] http://en.wikipedia.org/wiki/Technology_acceptance_model

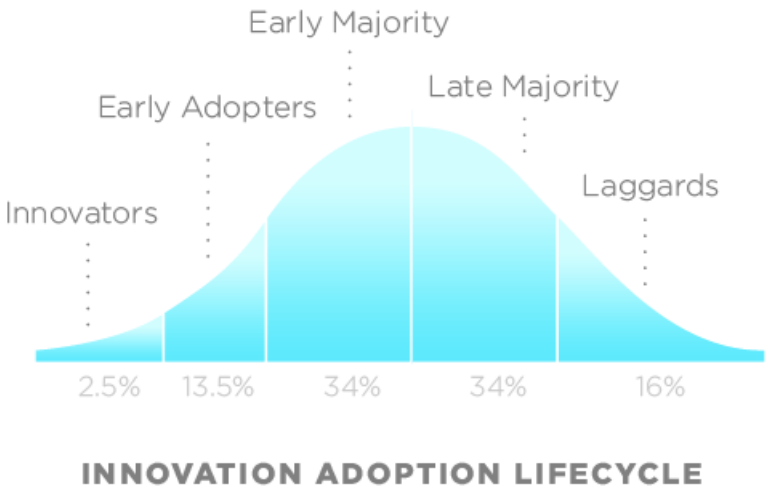


Figure 1.1: Roger's bell curve, source: http://en.wikipedia.org/wiki/File:DiffusionOfInnovation.png

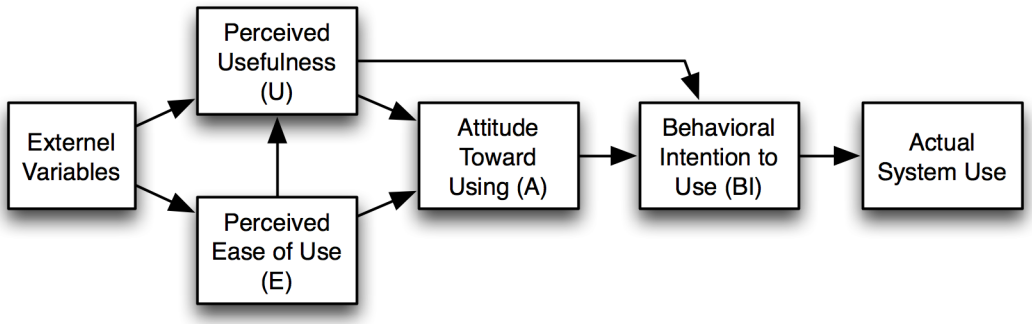


Figure 1.2: Technology Adoption Model, source: http://en.wikipedia.org/wiki/File:Technology_Acceptance_Model.png

to be implemented, and the fact she currently has no online presence already, it is our understanding that she believes this website or new system will definitely enhance her job performance. We have some reservations about her perceived ease of use however. As is explained earlier, she is very inexperienced with Internet technologies in general and may not feel entirely comfortable when faced with the new system for the first time. It is our main task to achieve simplistic, user friendly design that she can learn to use as quickly as possible and overcome any early doubts about the system. We believe that the usefulness of the website in the nature of her work greatly outweighs the challenge of having a novice user, although it is certainly a trade off we have to be mindful of.

Reflecting on our experiences in our project, working with the client has not been exceptionally easy. One of our main challenges is that our client is a novice computer user. Quite ironically, it has been a task for us to translate regular computing jargon into layman's terms in order for them to understand. This was critical in our requirement gathering process. In spite of this additional challenge that teams working with other projects might not particularly face, it is not necessarily

something that is discouraging for us. When we eventually graduate and face real world software projects, it won't always be technically minded people like ourselves we deal with, it will people who are more similar to our client. It will provide us something to drawn upon when we are asked to recount our experiences in future interviews. The opportunity to work with a real client will serve us as great preparation for working life.

Our project lives on the Internet. Ten years ago, the Internet was everywhere, but now it's even more so. It's a rapidly evolving area where new technologies and new techniques are developed very quickly. These have been made easy to use by anyone, regardless of: computer knowledge, nature of their device (mobile or desktop) or their operating system. All they need is an up-to-date web browser - arguably an easy requirement to fulfill, since most of them update automatically. So, there is a large audience that our project can reach, and that is considerably motivating for everyone in our team who worked on this project.

1.4 Preliminaries

To understand this report it is necessary to understand that we do not have to implement a translation algorithm. We are providing customers with an interface to send documents to be translated by a professional translator for a fee, and then returned in the chosen translated language using the same web interface. To understand the process that we have devised it would be advantageous to have a simple understanding of how a database works. As mentioned earlier, we have adopted several frameworks in our project development. These include Bootstrap (CSS, HTML), CodeIgniter and phpMyAdmin (all of which are discussed in Section 2 later). These frameworks provide advanced functionality which will, in most instances, not be necessary for our project and will not be utilised. Conversely, it allows us to demonstrate that we are professional software developers and that we are capable of software re-use.

1.5 Outline

The remainder of this report will go into more detail on the background research of our project, expand on our motivation and set out our group organisation and project plan. After this we will detail our design ideas and methodologies, before moving on to document our implementation, testing and evaluation. We will then discuss any problems encountered and the results of our evaluation before revealing the final status of the project, giving a detailed outline of the deployed site including any graphics and information relating to Bethel Translations.

Chapter 2

Design

2.1 Requirements Gathering

At the beginning of our project we met with the translator and our project supervisor to discuss her requirements. THe translator currently translates for an agency, on an ad-hoc basis, and is looking to create her own translation business. To this extent she wishes to have a website built to allow her to gain an online presence in translation and to digitize the process of receiving and sending translations. We were armed with a long list of questions and ideas for the meeting and a transcript is included as an appendix.

As software developers naturally would when creating anything from scratch, our team looked for similar websites already in existence. We identified common useful features of each, and also those features that were not so useful, and listed some we thought could be useful but simply did not exist in any of the sites we examined. One recurring theme we noticed in a majority of similar translation websites was that the home page was very cluttered and full of text. In other words, the process that the user had to follow to obtain some translation of a document was not extremely clear. Instead, they were met with various registration options, other services and annoying advertisements. **Figure 2.1** is a prime example of such bad design practices. It is cluttered, unclear and unattractive to potential customers. The most intruiging thing about the website in particular is that is ranked *first* in a Google search for the term "document translation". Rather alarmingly it seems this site has prioritised search engine optimisation over actual usability, or they have spent most of their budget paying to be rated first. Whilst being rated first is an advantage to the amount of business received, it should not detract from giving the user a usable and enjoyable experience. We therefore felt encouraged that there was certainly room on the online domain for a document translation service that was simpler, better designed and more intuitive to use.

From previous modules in our degree, namely IM2 and IS3, our team had experience of applying Jakob Nielsen's heuristics to obtain a successful user interface. We wanted to develop the idea that our website would display the minimal amount of information to a user by providing the registration, document upload and language selection elements on a single page, in a simple 3 step process. Our interface would then fall in line with the principle that user interfaces should have an aesthetic and minimalist design. [1] We believe this is an extremely important aspect of any modern website

[1] http://www.useit.com/papers/heuristic/heuristic_list.html

Figure 2.1: Online document translator, source: http://www.onlinedoctranslator.com/

based on the way that users make a decision of whether or not to use the services offered by the website. For example, imagine a user enters a Google search for “Translation service” and clicks our website in the results page. If the page they are met with looks too complicated or confusing in nature, the user simply clicks “Back” on their web browser, and goes to the next appropriate web page. If, however, the website looks clean, simple, and easy to use, the user would be more inclined to use it properly. We believe that our 3-step process found on the home page encourages anyone that requires a translation service for the provided languages to at least try for a quote, if not go through with the whole process.

With this approach in mind, we began to create some wireframes in order to form a solid idea of how this 3-stage process would physically look on our website. **Figure 2.2** illustrates our early attempt at doing so.

The layout is intuitive, flowing and simple to approach. In this early design we have split the page into two sections. In section 1 the user is presented with a short amount of precise text detailing the main function of the site. Subsequently, in Section 2 the user simply enters some details, adds the documents, sets the requirements and clicks a button to request a quote.
There is no daunting, large form based entry that some other websites encapsulate. It is minimal and it is undemanding. Obviously, the majority of the work done by our system would be when the user clicks “Get your quote”, and the system would have to register a user (pending email validation), upload their documents to the file store, associate a *jobid* and requirements, and place the job in our client’s pending work queue. The implementation of such functions are discussed later in Section 3.

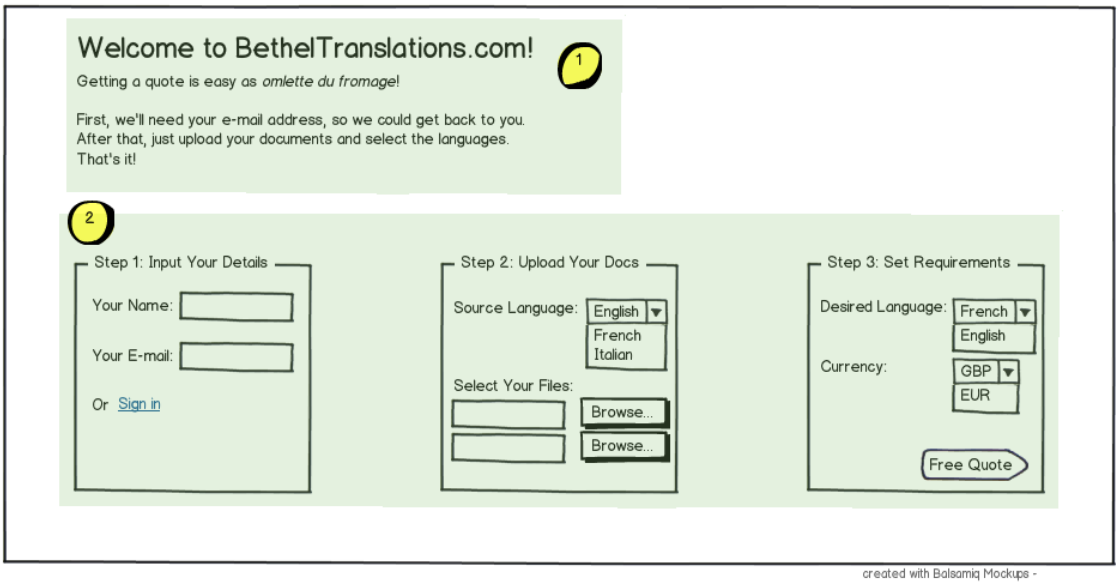


Figure 2.2: The 3-step process

2.2 User Process

With this 3 stage process as our main design factor of the website, we began to consider the main user groups of the website. We identified the needs of two main user classes: the **customers** and the **translator**

The clients are the users of the service, the visitors of the website. However, as far as the system is concerned, not all visitors are clients, because, in order for a visitor to become a client, he must register with the service. So we decided to have a visitor as a category with a registered user a subcategory. A registered user would then possess all the same abilities as a visitor with some specialised capabilities:

Visitor

1. Rationale: The visitor is just browsing. An anonymous visitor of the website.
2. Background: ”I need to get some documents translated. I came across this website and before I register or send any of my documents, I want to make sure that I’m dealing with a serious service.”

- He/She is a potential client, therefore the steps which he must make in order to become one must be as clear as possible.
- His/Her **goal** is to inform himself/herself about the service. In order for him/her to transition to being a customer, he/she must be convinced that the service provided is of great quality, so the system’s goal is to make itself trustworthy. Also, a clear privacy policy regarding e-mail addresses and the documents should be available, since they might contain sensitive data.

Customer

- Rationale: The customer is a registered user of the system.
- He/She has the same goals as the visitor, but, now that he/she is registered, he/she trusts the service a bit more. He/She has access to all the documents that he/she ever submitted for translation and can view each of the **job statuses** for which his/her documents are contained within. He/She can view documents he/she has paid for, and up to a certain period of time, download both the original copy and the translated copy. He/She can also view some other useful statistics on his/her previous jobs.

Administrator/Translator

- Rationale: The translator is the one answering all the translation requests.
- ”As a translator, I must review the documents that my clients send me and quote them. If they accept the quote, I must also translate them and let them know when the work is complete.”
- His/Her goal is to answer all of the client’s requests.

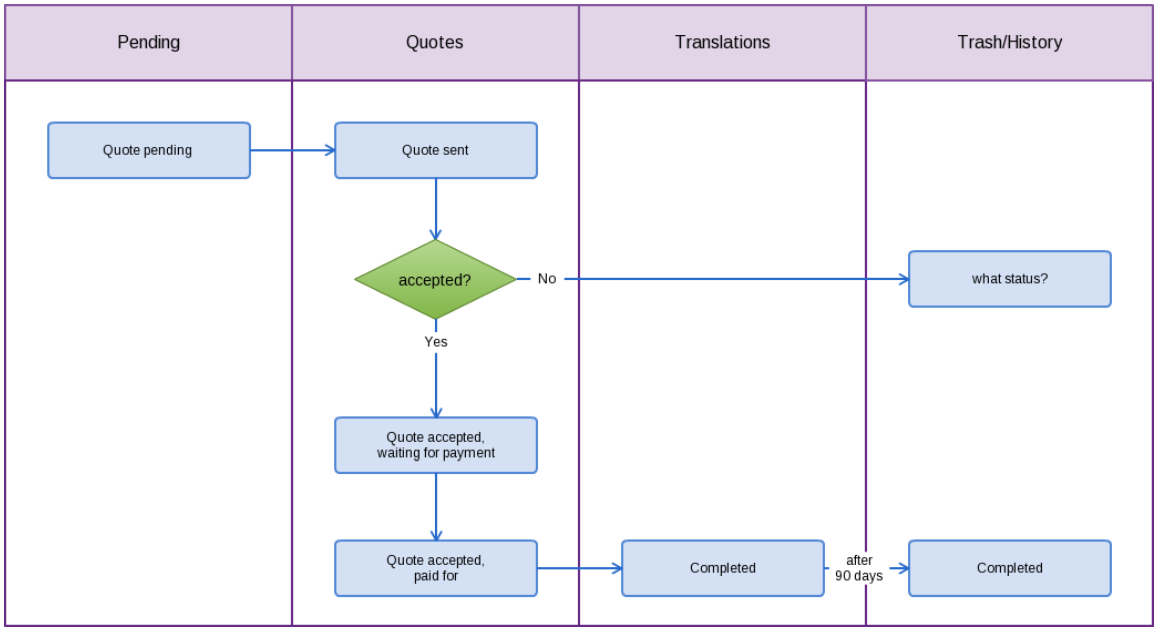


Figure 2.3: The transition of jobs through ”statuses”

Building upon the needs of these user classes, we needed to design a flexible system that provided functionality for all of these goals. Our main focal point would be the transition of **jobs**. In the context of our system, a job is what the website creates when the user uploads one or more documents for translation. Clients (registered users) would submit and pay for them. The translator would review them, download them and upload them. We considered the transition of jobs throughout the life cycle of a translation, as job statuses, and produced a sequence that is illustrated in **Figure 2.3**.

The diagram is much self-explanatory, but to summarise: Jobs that are submitted by the clients are placed in a **pending** work queue. The translator is able to review these documents and send some quote to the client. After a job has been quoted, it is placed in the **Quotes** queue. Quoted jobs are held in this queue until they are paid for via Paypal. When this has happened, they are moved to **Translations**, which is a breakdown of completed jobs. After a period of 90 days, jobs from this section are moved to **History** in order to reduce space on the server.

The transformation from this design plan to a viable user interface that is built upon these job statuses is described later in Section **??**.

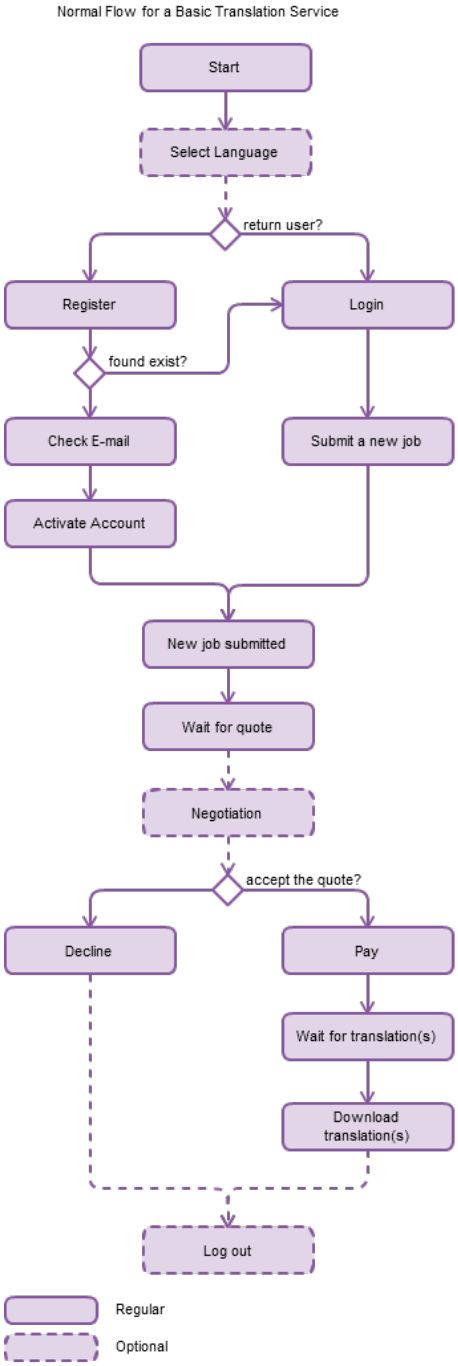


Figure 2.4: userProcessClient

First the client will have the option to switch among 3 displaying languages, namely English, French and Italian.

To use the translation service, one must first register by: inputting name and e-mail address, uploading one or more file(s), and setting translation requirements (source/desire language, currency and due date). Note that uploading is required so that only those who are actually going to use the service are allowed to register. Also, e-mail activation is required to finish the registration. If the system finds that the inputted E-mail address has been registered, the client will be asked to login.

Either by finishing the registration process or by submitting a new job (uploading file(s) and setting requirements) after login, a new job is submitted.

Meanwhile the translator will quote the job and a negotiation on details such as price or due date will possibly happen during this stage (it is optional, however).

If the client decides to accept the quote she/he will need to pay immediately through PayPal or could just leave it as quoted. Then the file(s) will (hopefully) be translated by the due date and a download link of the translation(s) will be offered in the client dashboard.

If the client decides to decline the offer, the translation service process will be canceled.

Other than those above, the client may:

Navigate the links on the main menu: read the About page (information about the service and the translator); read the Testimonials page; contact the translator in the Contact page through the contact form; request other services (Edit and Proofreading, Over-the-phone Interpreting and Video Remote Interpreting) in corresponding Service page through the contact form.

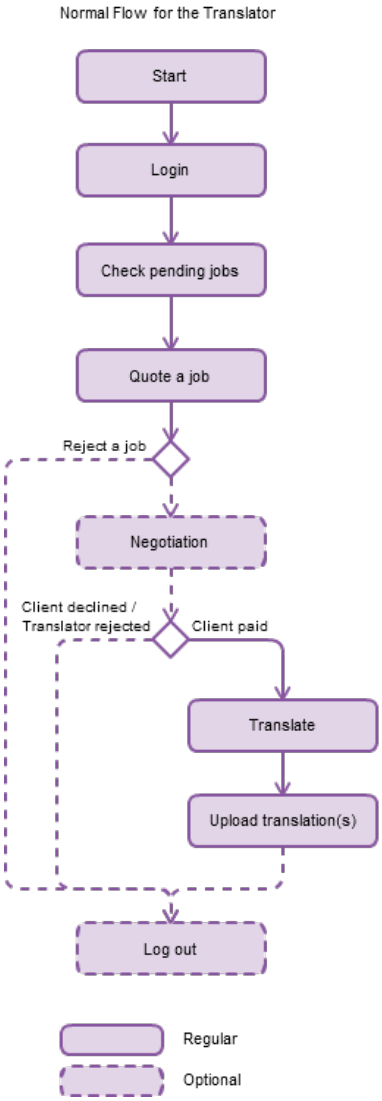


Figure 2.5: userProcessTranslator

First the translator will be directed to the admin dashboard, in which she/he will (probably):

Check all pending jobs, and pick one to quote.

The quoting process will include: check the job requirements, download the files uploaded in the job, read through the documents, set a quote and confirm it in the dashboard. During this process, the translator may reject a job within any stage.

A negotiation between the translator and a client may happen during or after the quoting stage. They may communicate via e-mails.

If no rejection or declination happens, the translator will then be waiting for the client to pay. Only after the payment is successful an actual translation process will start.

Finally the translator will upload the translated documents to the server through the admin dashboard.

Other than those above, the translator may:

Check processed jobs with different status in Quoted, Accepted, and Declined sections in the admin dashboard respectively. Also all recent translations can be downloaded through Translations and all past translation records can be found in the History section.

View the statistic data for the website in Site Statistics.

Navigate the links on the main menu: read the About page (information about the service and the translator); read the Testimonials page.

2.3 System Design and Wireframes

After laying out the process of the website we focused on organizing the content and on the wireframes.

Based on the user analysis discussed in section **??**, we split up the website ito two abstract sections:

- **Presentation pages**, which are available to all visitors, and describe the service, means of contact, and other such information.
- **User dashboard**, the section of the site available only to registered users, and which allows them to see the progress of their documents.

"presentation" - not the best word?

The **presentation pages**, as stated above, need to inform the visitor about the service. Next to "document translation", the translator also provided "video interpreting", "proofreading", and "over-the-phone interpreting". Since the first one is the main service, we decided that the front page of the website would present only an overview of all of these, while focusing on the document translation. The other services would have their own separate pages.

The main function of the **user dashboard** is to give a clear view of the statuses of the different jobs a customer has. With the status flowchart (Figure **??**) in mind, we thought that an appropriate structure to try would be one page per status, as shown in Figure **??**.

2.3.1 Information Architecture

In our first interview with the translator, we learned about her workflow and how she organizes her documents.

Based on the life-cycle of "jobs", which we discussed in Section **??**, we have

We analyzed the requirements and made a website structure diagram (Figure 2.6).

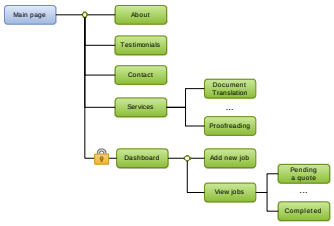


Figure 2.6: Website structure

2.3.2 Wireframes

Wireframes are blueprints of the website, representing every important page and their structure and behaviour. They focus more on what elements would a page contain, and their place on the page, rather than a refined design. Therefore, these sketches take form of rough drawings of the final design, usually keeping things like images or colours out, because they can distract the reader from his main purpose, which is analyzing the layout of that particular page, rather than the visual design.

Conventions used in the wireframes:

- **Green background** and/or **yellow labels** - are used to indicate specific groups of content that will be referenced in the document.
- **links** - blue text and underlined

By the nature of their content, the pages can be categorised into static and dynamic. Static pages display the same information for all users, regardless of their status (logged in or just visiting, client or administrator). About, Testimonials and Contact are such pages in the system described here. On the other hand, dynamic pages draw some data from the database and display it accordingly. In this case, the dynamic pages are the ones in the dashboard (both clients and administrators) since the information there is specific to every client.

Template

All the pages have this basic layout comprised of the following sections:

- Header
- Content area
- Footer

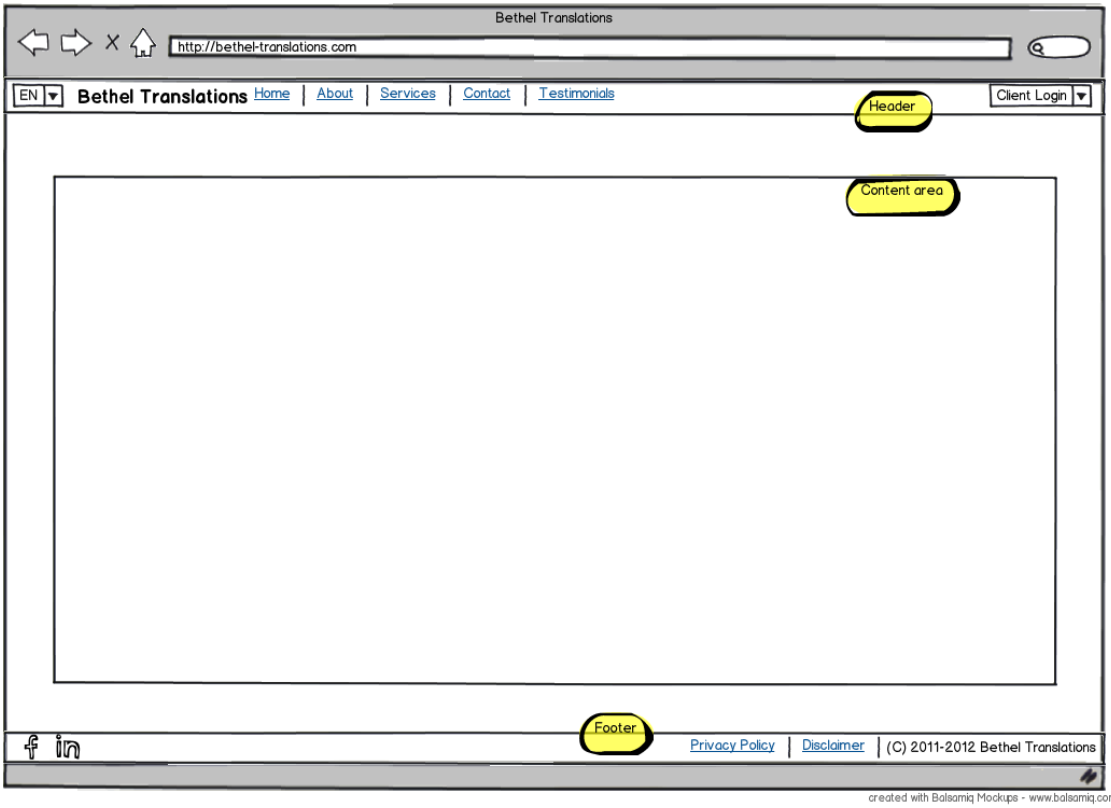


Figure 2.7: Main layout

The **header** and the **footer** sections will remain unchanged across all pages and the rest of the content goes in the **content area**. This is a good design practice because it keeps the entire website consistent. In terms of technological benefits this will increase the speed of loading the website due to lower bandwitdh requirements. The browser does not need to reload the header and footer every time, as it only loads the new sections. The browser can cache these sections also meaning that the site is more efficient. The rest of the sketches of the pages illustrated here will only show the content area.

The header contains:

- **the name and/or logo of the website.** This is key as it promotes the translators brand. Bethel Translations should be prominent to all clients so that they know what website they are on, they can use this to recall the services it provides and also recognise the quality of the brand at the same time.
- main navigation menu

The menu is a collection of links to the main areas of the website:

- Home
- About
- Services
- Testimonials
- Contact

- Login - this link must stand out, since it is of a higher importance and the page linked is of a different nature than the rest, so it is separated from the others.

Footer:

- copyright notice, if needed
- links to legal documents (e.g.: Privacy policy, terms of use, etc)
- awards or certifications (translation services related, PayPal certification)
- contact information (actual information or just a link to the 'Contact page)
- link to the business facebook page

Main page

From our earlier decision of creating a simple three step process and implementing this on our homepage, we took our wireframe and refined it to come up with the **Main Template** (Figure 2.7).

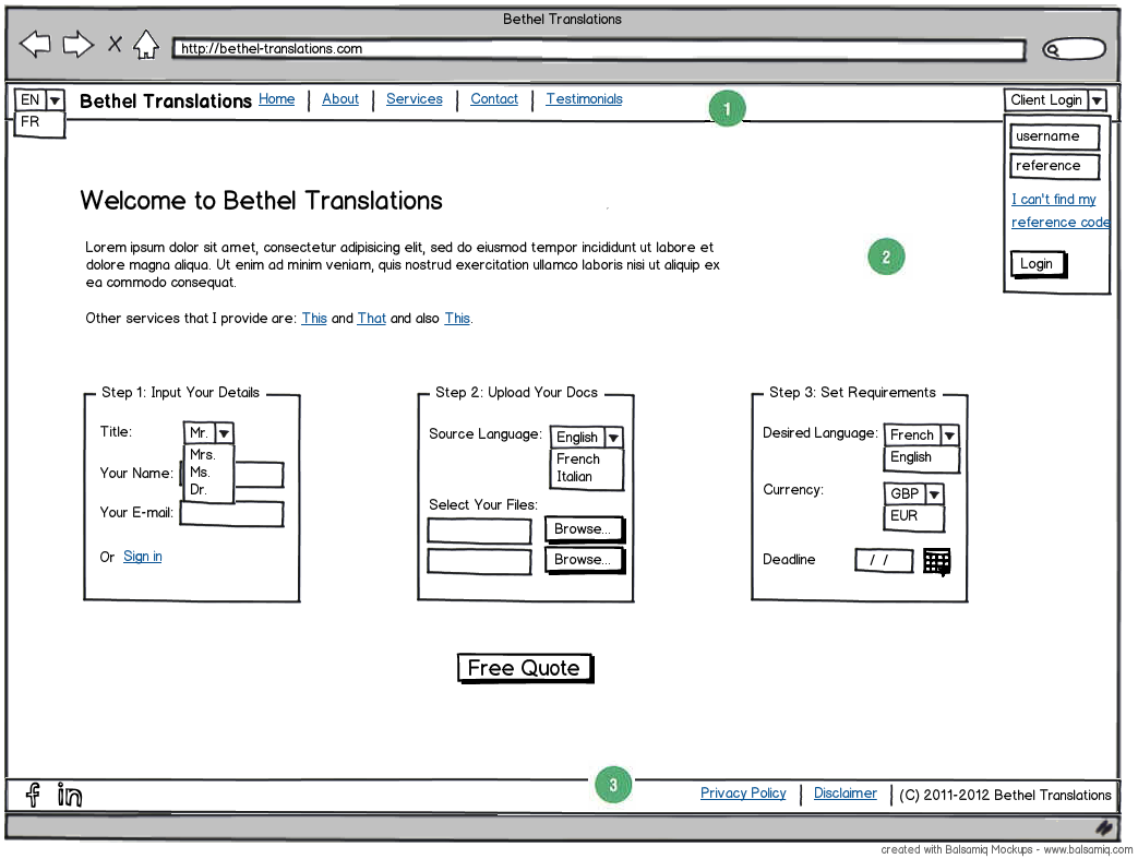


Figure 2.8: The Main Template

The site-wide header (labelled 1) and footer (labelled 3) can clearly be seen to be simple and uncluttered.

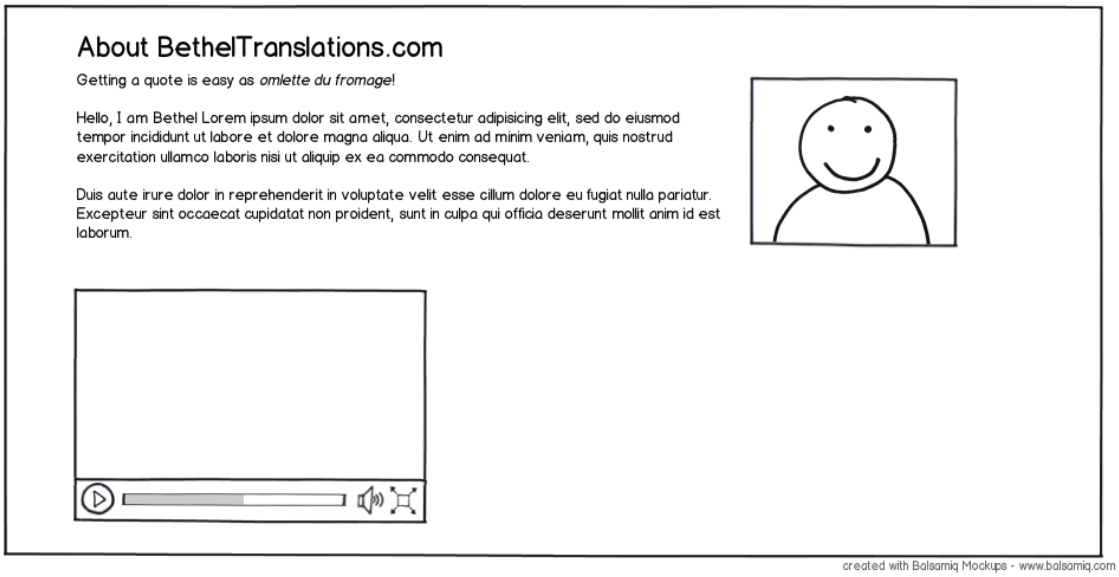


Figure 2.9: The About page

About

This is the page the users would go to in order to find out more about the service. Contains a few paragraphs that detail goals and accomplishments. The page needs to answer some possible questions that the users might have regarding the business:

- who is behind it?
- what are they doing?
- when did they start doing it?
- where are they?
- how are they accomplishing what they claim to do?
- Optionally an image, or even a short video, to enhance trustworthiness.

Testimonials

Having testimonials from happy customers also adds to the trust of the business. The business owner would ask her clients for feedback and permission to publish it on the website. Then she can pick which ones would suit her and post only a fragment on the website, along with some details of that client (name, company, occupation).

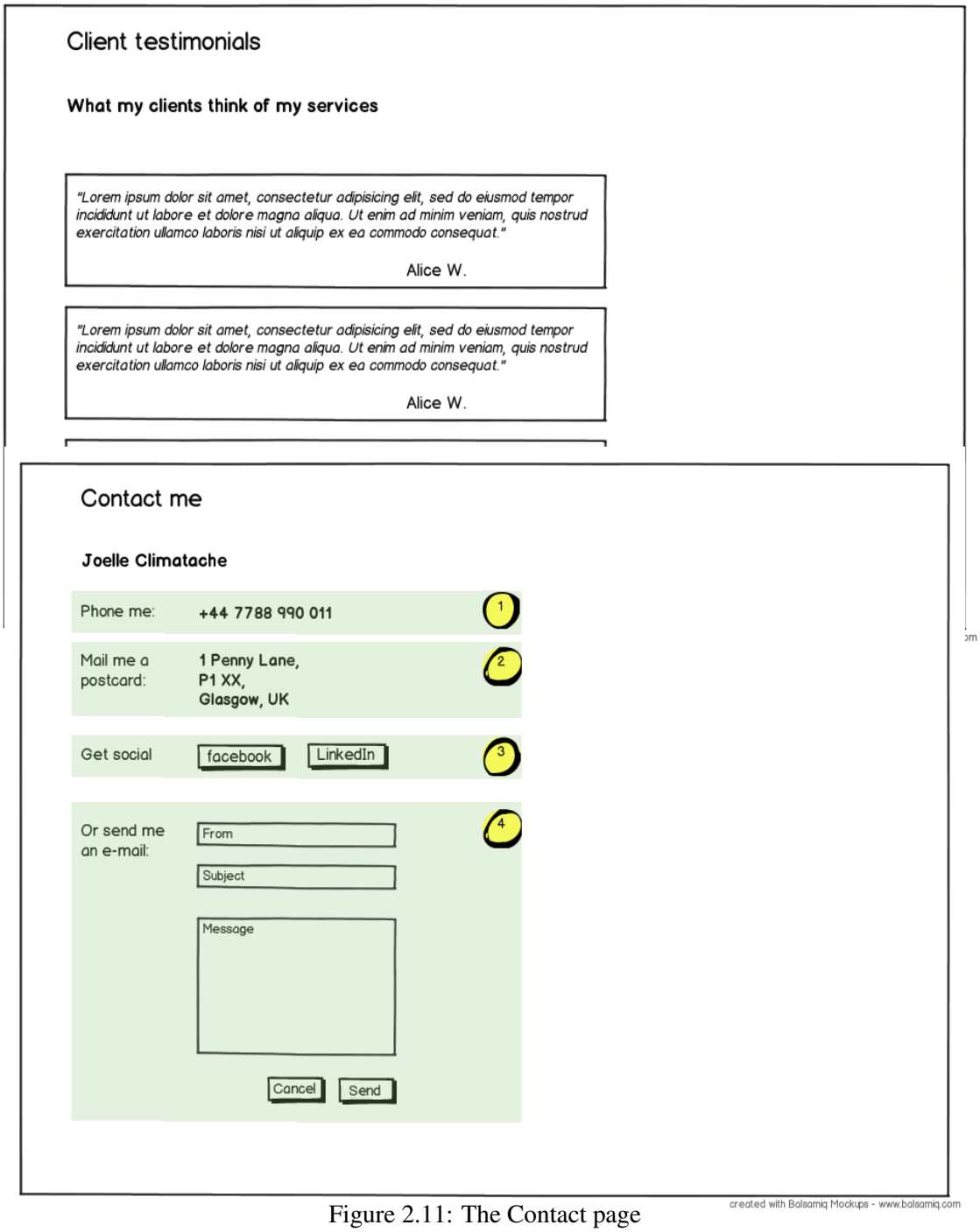


Figure 2.11: The Contact page

Contact

Having other contact methods (phone number, physical mailing address) also add to the credibility of the business. “A company with no address is not one you want to give money to.” [Jakob Nielsen]

1. work telephone number
2. physical address
3. social media pages
4. contact form (by e-mail)

2.3.3 Summary

From the images and diagrams in this chapter we have constructed the blueprints for a modern, stylish, and most significantly - user friendly website. Ultimately the website we want to build is one the client can fully use, without any reluctancy. The next chapter, **Implementation**, describes how transitioned from design to implementation, and the methods used in doing so.

Chapter 3

Implementation

In this chapter, we describe how we implemented the system from our design plan and detail the technologies used in doing so.

3.1 Development Environment

Programming for the project was split between several web-orientated languages. The central development language was PHP for developing the controllers and models. HTML5, JavaScript and jQuery were used in the views. As mentioned previously we employed the use of two frameworks, namely CodeIgniter and TankAuth. CodeIgniter provided the Model-View-Controller architecture PHP framework for the website, and TankAuth is an open source authentication library for CodeIgniter.

As we learned from the Distributed Information Management course we studied this year, web-development is ever-changing and to stay current developers must adapt to using new technology. As a result of this we have coded the views using the newest HTML5 standard. This will allow future developers to maintain the site with ease as adding new modern features will be simpler.

Another thing we learnt from our Distributed Information Management course is that it is desirable to separate out our concerns when developing in a web environment. This led us to using the open source Twitter Bootstrap `http://twitter.github.com/bootstrap/`. Twitter Bootstrap is simple and flexible HTML, CSS, and Javascript for popular user interface components and interactions. During implementation Twitter released an upgraded version, version 2.0, of Bootstrap and we upgraded to this version when it was released in early 2012.

For implementation purposes we set up an online SVN using Google Code. This version control system allowed us to keep track of changes, report and resolve issues, and maintain a wiki of useful pieces of information that we needed to keep and track.

Further to this we also set up a test server to allow us to view our site live on the web as we developed it and to test any changes as we made them.

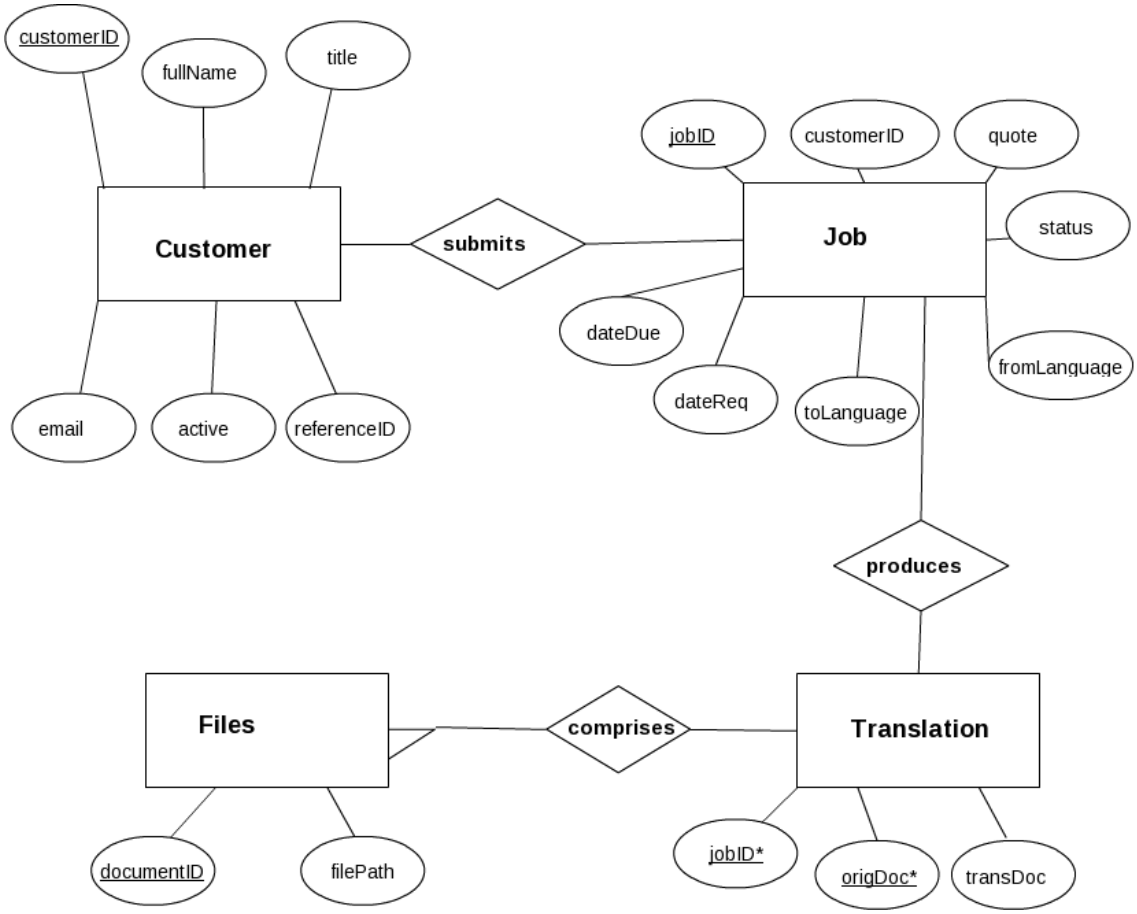


Figure 3.1: ER Diagram for Bethel Translations

3.2 Database Model

Our database structure was designed after we had thoroughly revised our user registration and job transition processes. We envisaged there being four separate entities: one for **customers** to become registered, one to represent **documents** being submitted, another for the **jobs** that are comprised of submitted documents and finally one for the documents once they have been **translated**. The attributes of each of these entities and the relationships between them is illustrated in **Figure 3.1**

The majority of activity for these tables occurs in the first three entities, as they are all populated in some way during the 3 stage process discussed earlier.

3.3 Database Schema

TBC

3.4 Prototype

TBC

3.5 User Interface

TBC

Chapter 4

Evaluation

First feedback with Karen incl wireframes etc - Jan 25
Second feedback session with Karen incl more complete website design - Feb 9th

4.1 Testing

4.2 Translator Evaluation

4.3 Client Evaluation

Chapter 5

Conclusion

Team photo goes here? :) A great project! etc..

5.1 Aims

TBC

5.2 Achievements

TBC

5.3 Future Work

TBC

5.4 Contributions

Alasdair done this... Andrei handled that... Stephen took responsibility for... Paul mainly done... Wei was responsible for...

Appendix A

Glossary of Terms

- **Free-lance** - Working for different companies at different times rather than being permanently employed by one company.
- **Customer** - A person or organization using the services of a professional person or company.
- **Translator** - A person or organization who offers the service of converting a document from one language to another.
- **Users** - A set of people who use or operate something, esp. a computer or other machine.
- **Requirement gathering** - Determining the needs of a client through any form of communication.
- **Software project** - Using the surrounding context, a software project aims to create application(s) using programming language(s) by adhering to project management principles.
- **Programming language** - A programming language is an artificial language designed to express computations that can be performed by a computer.
- **Web scripting language (*PHP, Javascript*)** - A scripting language is a programming language that allows control of one or more applications.
- **Website development** - The process of constructing and maintaining a website.
- **LAMP** - LAMP, (Linux, Apache, MySQL and PHP), is an acronym for a solution stack of free, open source software
- **Web application framework** - A software framework that is designed to support the development of dynamic websites
- **Open source** - Computer software for which the code is freely available

Appendix B

Project Plan

B.1 Introduction

B.1.1 Identification

This is the Project Plan for the Website for a Translator project by Team O.

B.1.2 Purpose and Description of Document

This project plan sets out all the details of the project and defines all the necessary work that must be undertaken to fulfill the project. It will be used as a reference guide by the team to make sure the project stays on time and that all the tasks that need to be undertaken happen according to schedule and in the correct order to acheive completeness.

B.2 Resources, Budgets, Schedules and Organisation

B.2.1 Work Breakdown Structure

Task 1	Group Organisation
Description	A document will be written following discussion with the team as to how the team will work and what our methods of communication, storage and backup will be.
Outcomes	The team will have clearly defined roles and each member will be able contact all members of the team in defined ways.
Deliverables	Team Organisation Document
Risks	Team members may be disappointed with role or may not agree on structure

Task 2	Scheduling and Planning Meeting
Description	A meeting will be held with the team to discuss our planning methods and scheduling of resources.
Outcomes	The basis of a project plan.
Deliverables	None
Risks	Certain events might be scheduled when a team member is unavailable.

Task 3	Project Plan
Description	A Project Plan will be written which the team can use to keep track of the state of the project. This will allow sufficient division of human resources to different aspects of the project.
Outcomes	Project Plan
Deliverables	Project Plan
Risks	N-A

Task 4	Research
Description	As a pre-requisite to requiremnets gathering we must reasearch the current trends in web-devlopment and research what works and doesnt work in a user friendly website.
Outcomes	The team will have an understanding of what current design practices are in use and what we must achieve to be competitive in the market.
Deliverables	None
Risks	N-A

Task 5	Requirements Gathering Interview
Description	An interview plan will be written, consisting of the objectives of the interview, questions to be asked, identification of the roles in the interview. During this task our initial research will be brought together, including the discussion of the use cases involved in the task. It will then be reviewed, edited and approved before going to interview the translator.
Outcomes	The team will have a firm understanding of what the translator wants the website to achieve and will be able to continue the process of development in a now fully informed manner.
Deliverables	None
Risks	If we do not ask the right questions we may not gather enough information about the system and will have a flawed understanding of what we must achieve.

Task 6	Specification and Requirements Document
Description	Following analysis of the interview writeup create a requirements document from the specification as set out by the translator.
Outcomes	Formalised requirements document
Deliverables	Requirements Document
Risks	Client may change their mind or team may have received innacurate requirements.

Task 7	Prototype
Description	Creation of a paper prototype to show to the translator to confirm that the requirements gathering process was correct and that we have a deep understanding of what the translator wants from the system.
Outcomes	Confirmation of correct requirements or list of ammendments needed to match requirements.
Deliverables	Paper Prototype
Risks	The prototype may consume too much time if the plan is not followed, therefore causing delay in meeting deadlines.

Task 8	Revision of Requirements
Description	From the Paper Prototype task the team will have recieved a list of things that need changed to match the translators requirements. These will be incorporated into the requirements document.
Outcomes	Requirements updated to what the translator wants. Team now has correct understanding
Deliverables	Revised Requirements Document
Risks	Translator may see this as an oppertunity to add further functionality to the specification.

Task 9	Static Page Implementation
Description	Implementation of the basic site, with only static pages and template.
Outcomes	Basic Website live on site
Deliverables	None
Risks	The implementation may consume too much time if the plan is not followed, therefore causing delay in meeting deadlines.

Task 10	Dynamic Page Implementation
Description	Complete implementation, including all document uploads, client and admin dashboards, paypal implementation and Contact forms.
Outcomes	Full version of website available online
Deliverables	None
Risks	The implementation may consume too much time if the plan is not followed, therefore causing delay in meeting deadlines.

Task 11	Testing
Description	Mmebers of the team with less involvment in the devlopment of the website will now fully test the site to check that the implemented functionality matches the requirements stated.
Outcomes	The team will have a list of known faults that need to be fixed and an understanding of what works well and what does not.
Deliverables	None
Risks	Although some members of the team will have had less involvement in the development, everyone has some understanding. The team have a biased view of what should and should not happen. This could lead to testing that is not 100 percent effective.

Task 12	Translator Evaluation
Description	The translator will be given a list of tasks and a questionnaire to complete in order to fully evaluate the site we have developed.
Outcomes	The team will learn what the client likes or dislikes about the website.
Deliverables	Evaluation Report
Risks	Ethics Approval

Task 13	Customer Evaluation
Description	Both potential and fake customers will be given a list of tasks and a questionnaire to complete in order to fully evaluate the site we have developed.
Outcomes	THe team will learn of what works and does not work about the site, getting feedback on why certain parts did or did not work.
Deliverables	Evaluation Report
Risks	Ethics Approval

Task 14	Dissertation
Description	A full report of the project, includind: design; implementation and evaluation conclusions. This is to allow our supervisor and reader learn about what we did, how we did it, and what did or did not work in our project.
Outcomes	Completed Dissertation
Deliverables	Dissertation
Risks	Dissertation may not be to acceptable standard, may be too long or too short or contain irrelelvant material.

Task 15	Project Presentation
Description	We will present a detailed description of how we went about taking this project from an idea to a completed piece of software. This will include a demonstration of the completed software.
Outcomes	Project Presentation
Deliverables	None
Risks	Not all team members are keen public speakers - this may detract from the quality of the presentation.

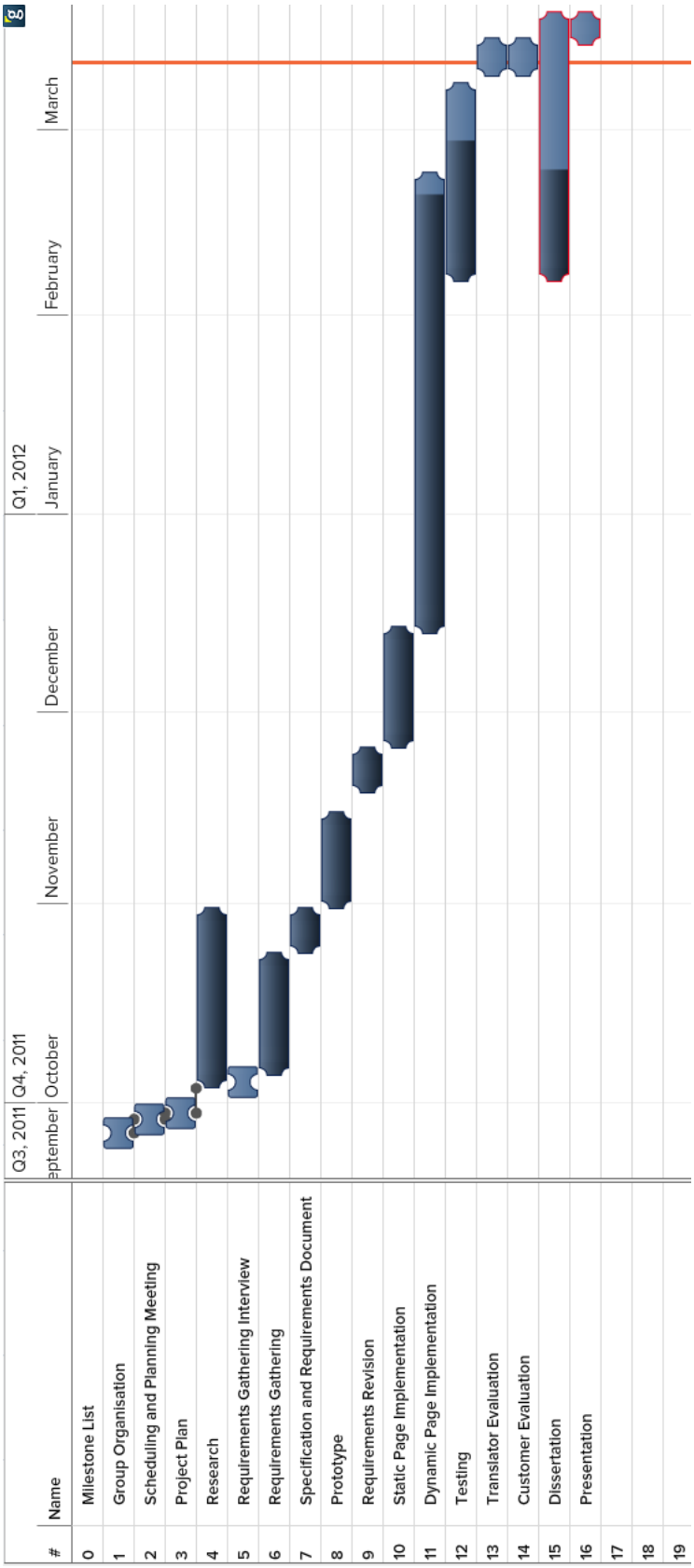
B.2.2 Resource Estimation and Allocation to WBS

We have five team members available to us. It is expected that each member spend around 5 to 8 hours a week on project giving us a sum total of 25 to 40 man hours spent on the project each week.

Other resources, as given to us by the Computing Science department, include a computer laboratory where we can undertake the necessary work to complete the project and access to a meeting room in Level 7 where we can have our individual team meetings.

Allocating these reources will be done on an ad-hoc basis in order to meet the deadlines and deal with the tasks according to the project schedule. One person will take overall responsibility, but everyone has an equal part in completing the tasks in the project plan.

B.2.3 Schedules



B.3 Organisation

TBC

B.4 Information Management

Our project data will be held on an SVN server with access rights given only to members of the team. This SVN server is hosted with Google, on Google Code. This means our project is open source. This is a requirement of using the CodeIgniter framework without having to pay for a license.

We also have a test server, to which we upload a second copy of everything from the SVN. This allows us to see the website as it will look on a browser viewed from a live web server. We have a script that we can run that will make sure everything that is on the server is also on SVN so that we do not loose work.

B.4.1 Communication

Our primary communication channel will be a mailing list, with every message archived on a server. Only members of the group can send messages to the team through the mailing list. Other methods of communication that we will use are:

- **Facebook** - Closed group page accessible by each member of the team, with ability to share multimedia content and instant messaging.
- **SMS/Phone** - Numbers were exchanged which proves a useful form of communication when a team member can't attend a meeting due to illness, for example.
- **Google Shared Calendar** - Every member can view and amend a group calendar set up on a gmail account.
- **Face to Face Meetings** - Every member will attend a weekly (or more frequent if needed) meeting to dicsuss progress.

B.4.2 Equipment, Materials, Facilities, and Other Resources

A wide variety of machines were used for the tasks in the project. The dissertation and reports were mainly written on lab machines in the Level 7 lab. The implementation was carried out mainly from home on team members individual personal computers and laptops.
During the supervised meetings notes were taken and distributed to the mailing list. The backups taken of documents were stored in various locations, such as on the university machines and at home. This provided a reliable backup and redundancy in case of media failure in one location. The backups of the test server were carreid out in a similar fashion in case of server failure.

B.5 Assurance Plan

From our Professional Software Devlopment course we have learnt that quality assurance is a major part of software development. From mistakes made in the past we now know that at least one member of the group should be appointed as Quality Assurer. This will minimise the risk of submitting poorly written code, documents or presentations and will greatly increase the standard of submitted work. This Quality Assurer should be a member of the team who is independant of the development team. This will allow them to report on quality issues without being influenced by the issues arising from software development. In a small team of five people it will be impossible to have a completly independant member being quality assurer however having people play to their strength means that one or two team members will be less involved with coding than others.

Quality Assurance will be carried out through all stages of the project: making sure that we adhere to set plans and deadlines, checking the standard of code submitted to the repository, and also spelling and grammar checking of the deliverables and dissertations. These reviews and inspections, when used along with software testing, will allow us to validate our requirements and verify that our software does what we intended it to. The document reviews will insure that we right a document that is fitting to properly explain and describe all that we have achieved.

B.6 Risk Management Plan

B.6.1 Risk Identification and Analysis

B.6.2 Monitoring

B.6.3 Avoidance

B.6.4 Mitigation

B.6.5 Review

B.6.6 List of Managed Risks

B.7 Configuration Management Plan

Appendix C

Evaluation Documents

C.1 Introduction and Consent

Bethel Translations Evaluation Consent Form
This evaluation will take about 20 minutes to complete. You may ask as many questions as you like before the evaluation starts. The tasks you have to carry out will be provided to you on another sheet. You are asked to circle YES if you successfully complete a task or NO if the opposite.
When uploading files please do not upload private or assessed documents lecture PDFs are an example of a good document to upload. All documents will be deleted after the evaluation and will not be opened or parsed.
When you have completed all tasks please tell the person in charge of the evaluation and they will direct you to the online questionnaire that has to be filled out to complete the evaluation. All results will be held in strict confidence, ensuring the privacy of all participants. No personal participant information will be stored with the data. Online data will be stored in a password protected computer account; paper data will be kept anonymous. Your participation in this experiment will have no effect on your marks for any subject at this, or any other university.
Please note that it is the website, not you, that is being evaluated. You may withdraw from the experiment at any time without prejudice, and any data already recorded will be discarded.
If you have any further questions regarding this experiment, please contact:
Team O: teamo@stbernadettes.co.uk
or
Karen Renaud (Team Supervisor): Karen.Renaud@glasgow.ac.uk

I have read this information sheet, and agree to voluntarily take part in this experiment:

Name: _____

Email: _____

Signature: _____

Date: _____ Age: _____

C.2 Task Sheet

Task Sheet: Client
Please navigate to www.betheltranslations.com then complete the tasks below.

Task 0: Navigate to all the pages of the site and write down the first thoughts you have about the site. Please write on the other side of this page.

Task 1: Please fill out the form on the home page and upload a doc. Please choose English to French translation.

Successful? YES NO

Task 2: Log in to your dashboard.

Successful? YES NO

Task 3: Check the status of your document and submit one more (French to Italian) Then accept one quote,

Successful? YES NO

Task 4: Download your translations

Successful? YES NO

Task 5: Contact Joelle to negotiate the price

Successful? YES NO

Task 6: Logout and visit the facebook page to leave some positive feedback.

Successful? YES NO