Requirements Specification
for
Beyond Limits Charity Website

16/11/2018
Version 1.0

Prepared by
Group 18

Nilsson Kristian
Mavropoulos Christos
Mortimer Jake
Redmond Matthew
Vasconcelos Francisco

Introduction

Brief:

Our brief is to design a website for the UK-based charity Beyond Limits. As their current website is fairly rudimentary, they do most of their publicity work using a Facebook Group. We are to take the elements that they already approve of from their original website (primarily the textual information) and create a new website which will provide all the functions they'll need to maintain their operations.

Justification:

Beyond Limits looks to help people who are "down on their luck". This includes families living in poverty, abuse victims, and people with learning difficulties. Beyond Limits bring these people to a safe environment in the countryside, where they can give them support and resources. The website is necessary in order to bring greater exposure and clarity to the charity and what they do. Doing so will benefit both the charity and, through them, the people who the charity help. It will also help those who want to support the charity, by making donating and volunteer application simpler and easier.

Overview of General Features:

1. The public will be able to donate money to the charity through the website.
2. Individuals will be able to apply for volunteering through information found on the website.
3. Anyone seeking help (with an email address) will be able to send a help/support request to the charity through an online form on the website.
4. Anyone interested will be able to learn more about the charity through the website, such as through the FAQ section, for example.
5. The charity will be able to more easily document and search who they are helping and what donations they have received, for the purposes of their annual reports.
6. The charity will be able to make edits to the website, as they grow and change.

Overview of the Remainder of the Document:

- Project Scope - A description of the scope of the project including boundaries, benefits, objectives, stakeholders and their needs.
- System Description - A description of the project's domain area, including research.
- Solution Requirements - Detailed and precise descriptions of both functional and non-functional requirements of the system.
- Open Issues - Identification and discussion of potential risks, hardware and software used and their limitations.
- Development Approach - Discussion of the chosen development approach and the alternatives considered, with justification.
- Terms and References - A list of all terms and references, with definitions for clarity.
- Project Schedule - A schedule containing both the soft and hard deadlines for the features needing to be implemented.

Project Description and Boundaries

The project is to develop a fully-functioning new website under the specifications of the charity Beyond Limits. The website should be simple to use for both the visitors and the administrators which are our clients. The project also involves creating a system for the administrators to be able to edit aspects of the website. That system will also incorporate a database in which the client will be able to store data about the people they help and potentially information that is essential for their annual report. The project doesn't involve creating a server to host the website or help them in other areas such as branding or marketing.

User Base (Stakeholders)

The main stakeholders are the general public and admins (with special privileges). The public will consist of:

1. People who seek help from the charity
2. Potential volunteers that need to find information and apply through the website
3. Volunteers who need to access the website for various reasons (eg.getting health and safety files)

4, The administrators; the 3 people who have access to restricted parts of the website containing confidential information, and tools to edit parts of the website.

Generally, we do not expect the user base to have technical expertise, as the charity revolves around helping underprivileged individuals that don't always have access to technology.

Overall, since their Facebook group has about 1000 members, we can expect at least that many unique visitors to be using the website.

Product Vision

A website that is modern and appealing while at the same time simple and minimalistic. Navigation should be intuitive and there shouldn't be any clutter. The admin page should follow similar rules so that the client would not be deterred when updating the website.

Project Deliverables, Features and Tasks.

Team 18 will be split into two sub-teams, one responsible for the front-end and the other responsible for the back-end of the website. The first team will oversee the design and style of the website. This design should be intuitive and functional so that navigation and general use of the different pages is simple enough for everybody to use.

The second sub-team is tasked to create a system on the website through which the administrators (our client) will be able to control and edit aspects of the website in order to customize existing content, add, and remove content. A login page will be created that has accounts for the administrators so that only they will be able to access the tools to edit the website. A database will also be created that the administrators can access. The database needs to have functions to store, sort and search information about all the people that the charity has helped. Furthermore, through the administrators' page, the client will be able to access and control the database through a series of forms and fields. Furthermore, the client will need to have a guide on how to use the features stated above so that they can effectively use their website and the tools provided to them.

Benefits and Overall Goals

The client is not using their current website as they feel it does not represent them and their vision, and it's hard for them to change it. The benefits of a new, easy-to-use website would be that the administrators would not find it difficult to create and post events or access the database which would encourage them to actively use the website. Thus, it will be easier for people to find out more about the charity which would extend the charity's reach.

Features for Future Releases

At a later stage in development, an archive feature can be implemented so that the administrators of the website can store information that will be useful when composing the annual

report. Also, the code for the website should be documented extensively and have comments to explain the different functions so that future development is easier, even by another group.

System Description

The website that we will be making for our client, Beyond Limits, which is a 'journey'. The website also has to be easily maintainable by someone who may not have a great knowledge of web design.

Some systems similar to this that we have done research into are: Christians against Poverty website, WordPress, and Facebook. I will give a brief description of each system, the reason for this choice of system to compare to, and discuss what is useful/not useful about these systems to our client:

1. **Christians Against Poverty(Front-End)**

   The client said that they really liked the Christians against poverty webpage because it is easy to find the content that you are looking for. They expressed that they were a fan of the drop-down menus because, once again, it reduced clutter and made the site easier to navigate. However, one concern they did highlight was the amount of scrolling that one had to do on the website. This is classic of websites which house too much information on each page. This can be easily remedied on the site we create by further decomposing each section; only a small, 'digestible' amount of information appears on each page.

2. **WordPress(WebPage Maintenance/Back-End)**

   WordPress is the framework that is currently used by the charity for their website. It is a PHP framework which is combined with the ability to edit content on the website at will. WordPress also provides statistics about the number of views to the site, etc. They also boast dozens of themes for your website(which need to be paid for), and guarantee that your site will be mobile-friendly.

   WordPress allows for the website to be altered on the fly; all the content, images, titles, etc. This is a very useful feature: for someone who doesn't know anything about how to make manual alterations to their site, if left the site would quickly become outdated. This is definitely something that we could improve upon: from their current site we can see that it has been rarely updated. This is apparently due to the fact that they don't feel comfortable editing the website. This has led them to be predominantly based on Facebook(which we will come to later).

   WordPress is very generic; it is not made for a specific client but tries to accommodate many possible clients. This is a simple, and quick website solution for many, however, the limited number of themes and lack of variety means that truly outstanding websites are hard to make. The client wants the website to be a 'journey', this is definitely not possible on WordPress. The number of themes and other customisation options are restricted by your plan; whether you are premium or standard user. Even if you have the maximum number of customisation options you are still limited in the number of design choices that you can make; this is due to the overwhelming minimalist design of WordPress themes.

3. **Facebook(Groups)**

   Facebook is being used as a research source because the client has expressed that they use it as a way to communicate with the community. Therefore, we would like to make use of the best features of Facebook groups, as highlighted by the client. The client said that Facebook was "easy to find and communicate" for organising events, etc. They also liked the way that they could delete items to avoid clutter. They also use Facebook for communicating with people who want to get in touch but don't want to feature on the website. We would like to make the events page, and blog page easier to update by making use of similar features to Facebook, and it may also be possible to create a feature to allow private communication with people who don't want to appear on the website itself. The pages would have a button to delete each post, as defined by the user's session; only Mrs Sadler and the other admins would be able to make these changes.

System Build

As regards the front-end (the design of the website) we are going to make the website very free-flowing; all pages will be transferred between with smooth transitions. This will help with the ethos of a ‘journey’. The first page that the user sees will be the home page; this will include links to all other sections of the site, as well as, potentially, an introduction to the site by the owner of the charity. Some more pages that will be included will be the testimonials page, volunteering page, donations page, contact us page, events page, blog page, pictures page, and a FAQ page. The testimonials page will include volunteer testimonials about their experience at the charity, as well as testimonials from people whose lives have been made better by the charity. It would be vital to include a testimonial from the charity's’ founder about her story and how she got to where she is: this would be very inspiring and would take any individual, who seeks the help from Beyond Limits, on a ‘journey’. The volunteering page would include a way for prospective volunteers to get in contact with the charity in order to apply for a role. It could include some notices for current volunteers as well. The donations page would be quite simple and would include an API to PayPal which would allow people to donate to the charity. It could also include a video explaining what the funds will be used for; why they are necessary.

The client was very keen on running competitions; the competitions will be part of the events page which can be updated from the admin panel but would include any competitions people can get involved in. The pictures page would include all the pictures provided by the client and would show the users of the site what a magical place the retreat is; once again taking them on a ‘journey’. The FAQ page would include questions and answers that the charity is asked a lot. This is necessary because the client stated that they are asked the same few questions quite frequently. The blog page will include a set of posts from Mrs Sadler or the volunteers about updates in the charity. This will be updated from the admin page, these posts can be created, edited, and deleted. In the footer of the website there will be links to other charities suitable for Beyond Limit's audience, an example of this being the Samaritans. The site will also feature links to social media; the users of the site can view activity on the charities social media accounts, and when the client adds to the blog their social media will be updated accordingly.

Regarding the back-end, we need to setup the server correctly to host the website and handle all requests from the front-end, and we will need to create an admin panel. The admin panel will be the majority of the work. This will be similar to WordPress in that the client will be able to edit and add elements on/to the site. The client, or someone working at the charity, will be able to add events; these will take on some template and the client can add in the required text. This will require correspondence with the client to determine the types of events they have which will dictate the templates that we make. Another function of the admin panel will be the ability to add to the blog page. This will add the content to the blog page as well as post directly to their social media to inform their followers of the new blog post. The entire back-end also requires us to make the admin panel secure; only providing access to a select few user accounts. We have been asked to make 3 user accounts as well as one account for the charity’s founder. These will have varying permission as to what they can change/add.

Some additional tasks that we would like to add to the system, if possible, are:

- A booking system for people to see if there is space to stay at the retreat overnight;
- An archiving system for annual reports; and
- A simple method of electronic data input for the charity to make the change into fully electronic data storage.

The booking system would be an additional page and would be connected to a separate data structure. This would hold the information as to whether there are rooms available and allow people to make a reservation by putting in their email and name. There would then be an automated confirmation email to make sure that they are a real person. Once the user had confirmed the booking the data structure on the server would be updated and thus the booking page would be updated accordingly.

The archiving system would most likely be a database/repository. We would make this look much nicer than a simple file storage system. The client would also have access to features such as create folders, upload, delete, etc. If this was to primarily be for annual reports then we could make this into a timeline on which the reports are attached along it by year. The client would need to be able to read/download these files as well. This would be behind a login screen which would use the same user accounts described above for the admin page. It is unsure at the moment whether the client has a clear format for each annual report; if this is not the case and this feature is implemented then one would need to be created. The client also made it clear that if they were to make the move to electronic data storage then all of the fields that are stored on the database would need to be editable, as well as being able to add fields. This would potentially be a feature for future releases because at the minute we are unsure as to how it would be implemented, and how complex it would be.

At the moment, the client stores all records on paper. We could attempt to make a data input section which was simple and easy to use. We would do this by, using the formats of documents provided by the client, making a simple form which when the requisite data is input, it will create a PDF with the information on the document. This PDF can then be stored on the server and can be accessed in some way. These documents would have to be backed up on local storage devices to prevent any risk of losing large amounts of critical information. This could be an option for future releases.

Functional requirements:

Website User Features:

ID, type and title	U1- Online help form
Description	A web-bound form filled in using text inputs (email address, name, age, comments) and tick boxes (for what help the user needs). Only one box may be ticked. The filled-in form will be automatically sent through email to the client when a “submit” button is pressed and an email confirming a successful submission will be sent to the email address given, if the email given is valid.
Priority	High
Dependencies	N/A
Expected results	A validly filled-in form (valid email address, name and help needed filled in) will be sent to the client through email and a submission success conformation will be sent through email to the user.
Exception handling	If the form is filled in erroneously (e.g. not a valid email address), the erroneous field will be highlighted red. All fields will remember the information input until a submission is successful or the user leaves the webpage.

ID, type and title	U2- Access/Menu Bar
Description	A collection of links at the top of every webpage, featuring hyperlinks to all main public webpages (e.g. Home, FAQ, How we can help, etc.). Hyperlink text will change shade/colour when hovered over.

Priority	High
Dependencies	N/A
Expected results	Each hyperlink will go to the appropriate webpage (e.g. clicking “Home” will take the user to the home webpage) when clicked on.
Exception handling	If the user clicks on more than one hyperlink, the last one clicked will be the one used.

ID, type and title	U3- Video feature
Description	Will play/pause a video when clicked on and will also feature sound controls. It is to be featured on the home webpage.
Priority	Low
Dependencies	N/A
Expected results	When clicked on, the intended video will play/pause.
Exception handling	If the video cannot be played after the video player is already loaded then display an error message, if the video player itself cannot be loaded then replace it with a block of text conveying the same meaning.

ID, type and title	U4- “More Info”
Description	On webpages with too much information to be fit onto the chosen height, this feature will automatically hide the excess information and replace it with the text “More Info” which, when clicked on, will reveal the excess information.
Priority	Medium
Dependencies	N/A
Expected results	When “More Info” is clicked, all hidden information will be revealed to the user.
Exception handling	If a problem is detected, it will display an error message instead of hidden information.

ID, type and title	U5- Animation
Description	When certain icons/links are clicked on/hovered over, an associated animation will play (e.g. fireworks or confetti).
Priority	Low
Dependencies	N/A
Expected results	Chosen icons will play an animation when hovered over or clicked on.
Exception handling	If an animation is taking greater than a specified time to load, it shouldn’t play (for those with slower computers).

ID, type and title	U6- Carousel
Description	Several images are cycled between, changing approximately every 10 seconds. There will be 2 arrow buttons, the left to go backwards and the right to go forwards.
Priority	Medium
Dependencies	N/A
Expected results	The carousel will display each image for approximately 10 seconds.
Exception handling	If the carousel cannot be loaded by a user, the first image should replace it as a static image.

ID, type and title	U7- Donation Button
Description	A button that when pressed will take the user to a donation menu allowing them to choose their method of donation (e.g. PayPal).
Priority	High

Dependencies	N/A
Expected results	The user will be able to donate to the client through clicking the button.
Exception handling	If an error is detected, the donation will be cancelled before the user has a chance to donate.

Admin Features:

Database Features:

ID, type and title	AD1- Add Record to Database
Description	The Admin will have the ability to add a new record to any table in the database.
Priority	High
Dependencies	AD2
Expected results	When they add a field, a blank record will appear at the top of the database table.
Exception handling	If a record is added by mistake, they'll be able to delete it.

ID, type and title	AD2- Delete Record to Database
Description	The Admin will have the ability to delete any record from either database.
Priority	High
Dependencies	N/A
Expected results	When they delete a record, that record shall no longer be stored on the database.
Exception handling	There will be prompt the make sure the user is not deleting a field by mistake.

ID, type and title	AD3- Database Table Search
Description	The Admin will have the ability to search in any given field, and will be presented with all records that match what they have searched for. They will be able to search using multiple table fields at once to make the search more specific.
Priority	High
Dependencies	N/A
Expected results	When a search is issued, all matching fields will be displayed.
Exception handling	If nothing matches the search, a message declaring such will be displayed.

ID, type and title	AD4- Add Field to Database Table
Description	The Admin will have the ability to add a new field (of a type of their choosing) to any table in the database.
Priority	High
Dependencies	N/A
Expected results	When they add a field, each record will now have that field and it will be blank (for the admin to input the data).
Exception handling	If a field is added by mistake, they will be able to undo it by typing Ctrl+z (or clicking "undo").

ID, type and title	AD5- Delete Field from Database Table
Description	The Admin will have the ability to delete a field (except for the Primary Key) from any table in the database.
Priority	High
Dependencies	AW8
Expected results	When they delete an field, each other field on it's right will move to the left to fill in the gap. When the admin tries to delete a field a message prompt will prevent them until they tell the prompt they are sure they want to delete the field.
Exception handling	If a field is deleted by mistake, they will be able to undo it by typing Ctrl+z (or clicking "undo").

ID, type and title	AD6- Database Table Sort
Description	The Admin will have the ability to sort based on any given attribute (increasing, decreasing, alphabetical…), and will be presented with all fields sorted in that order.
Priority	High
Dependencies	N/A
Expected results	When a sort is issued, all fields will be displayed in the order specified.
Exception handling	If two or more fields have the same value in the given attribute, the fields will default back to their original positions (whichever field was added to the table last higher up).

ID, type and title	AD7- Database Edit
Description	The Admin will have the ability to edit any record in the database.
Priority	High
Dependencies	N/A
Expected results	When an edit is applied, the value of the specified field in the specified record will be changed.
Exception handling	If the Admin puts in an incompatible value, they will be notified and required to change it.

Carousel Features:

ID, type and title	AC1- Carousel Editor Add
Description	When the Admin presses the "+" button on a carousel they will be able to add images to the carousel on a given webpage. When editing has been completed, an "Editing Complete" button is to be pressed to finalise the carousel.
Priority	Low
Dependencies	U6, AW8
Expected results	Any additions made by the admin will automatically be applied to the website once the "Editing Complete" button is pressed.
Exception handling	Given any sort of error, the website will not update and the admin will be informed through a message prompt.

ID, type and title	AC2- Carousel Editor Remove
Description	When the Admin presses the "X" button on a carousel they will be able to remove images from the carousel on a given webpage. When editing has been completed, an "Editing Complete" button is to be pressed to finalise the carousel.

Priority	Low
Dependencies	U6, AW8
Expected results	Any deletions made by the admin will automatically be applied to the website once the “Editing Complete” button is pressed.
Exception handling	Given any sort of error, the website will not update and the admin will be informed through a message prompt. If the carousel only has one image, it will not allow the admin to delete it.

ID, type and title	AC3- Carousel Timing Editor
Description	When the Admin views a carousel there will be a text box with the amount of time in seconds images are to be displayed for on the carousel. By typing into this they can change this time.
Priority	Low
Dependencies	U6, AW8
Expected results	When the time is changed and the “enter” key is pressed, the webpage will automatically update.
Exception handling	If the input is not a non-negative integer number, the carousel will not update on the webpage. The admin will receive a message prompt telling them their input was invalid.

Webpage Features:

ID, type and title	AW1- Webpage editor
Description	The admin will be able to edit the text on any given webpage by pressing an “edit” button (only accessible to them) and finalise their edits by pressing a “Submit” button.
Priority	Medium
Dependencies	AW8
Expected results	Any changes the admin makes will be finalised when the “Submit” button is pressed and a prompt ensuring they are sure they want to change the website is gotten through (they press “Yes”) and the website will update automatically.
Exception handling	If an error occurs, the website will remain as it was before any editing occurred.

ID, type and title	AW2- Add post (“upcoming events”)
Description	The Admin will be able to add posts to the “upcoming events” section of the website including text (necessary), image and video (if desired), and a date (necessary).
Priority	High
Dependencies	AW8
Expected results	When a new post is added, it will appear on the website in its appropriate place (based on date).
Exception handling	If an error occurs in submission, the admin will be informed through a message prompt.

ID, type and title	AW3- Delete Post (“upcoming events”)
Description	The Admin will be able to delete posts from the “upcoming events” section of the website. Each post will include an “X” icon that when clicked will cause a message prompt asking if the admin is sure they want to deleted the post with the buttons “Yes” and “No”. If they press “Yes” the post is removed from the website and otherwise the post is kept on the website.

Priority	High
Dependencies	AW8
Expected results	When a post is deleted, the admin should be asked if they are sure and if so the webpage should automatically update.
Exception handling	If an error occurs, the admin will be informed through a message prompt.

ID, type and title	AW4- Auto-delete (“upcoming events”)
Description	Each post must include a date after which point it will no longer be relevant and will be automatically be deleted from the webpage. The webpage will check all posts at the beginning of each day (e.g. at 00:01).
Priority	High
Dependencies	N/A
Expected results	Only the posts for events yet to occur will appear on the “upcoming events” webpage.
Exception handling	If a post has an incompatible date entered, it will not appear on the event page.

ID, type and title	AW5- Auto-sort (“upcoming events”)
Description	The posts will automatically be sorted by date, with those with the earliest date being placed higher on the webpage.
Priority	High
Dependencies	N/A
Expected results	All posts will appear on the “upcoming events” webpage in the order of what is coming up next.
Exception handling	If a post has an incompatible date entered, it will not appear on the event page.

ID, type and title	AW6- Add Question to Online Help Form
Description	The Admin will be able to add optional questions to the online help form, choosing from the answer types of text and tick-boxes (where only one can be ticked at a time). The website will update automatically.
Priority	Medium
Dependencies	U1, AW8
Expected results	When a new question is added, it will appear on the online help form at the bottom. It will include with it a way for a user to answer the question (e.g. a text box).
Exception handling	If an error occurs in submission, the admin will be informed through a message prompt. The online form will not be updated.

ID, type and title	AW7- Delete Question from Online Help Form
Description	The Admin will be able to delete optional questions from the online help form. Before deletion may occur, a message prompt will ask them if they are sure. Clicking “yes” on this message prompt will result in the question being deleted.
Priority	Low
Dependencies	AW8
Expected results	When a new question is deleted, all questions below it will move up the form to fill in the gap.
Exception handling	If an error occurs in submission, the admin will be informed through a message prompt. The online form will not be updated.

ID, type and title	AW8- Message Prompt
Description	When certain “high-risk” actions are being performed on the webpages, a message prompt will appear before they can proceed asking whether or not they are sure they want to proceed. Each Prompt will have two buttons; “yes” and “no”. By pressing “no” the activity is cancelled, by pressing “yes” the activity proceeds.
Priority	High
Dependencies	N/A
Expected results	When the admin tries to do something which could damage the website, a prompt will appear to make sure they know what they’re doing.
Exception handling	If the message prompt does not load, the action will not be carried out.

Non-Functional requirements

Performance	Every webpage needs to load with a low response-time. Preferably, around .5 seconds but no longer than 2 seconds. More resource intensive features (eg. carousel) should not hold back the loading of the rest of the page.
Manageability	The client will need to edit some of the webpages of the website by being able to post new events, delete old events, and make changes to other various pieces of information. They must be able to do that without needing to change the HTML code directly.
Usability	We expect the level of technical expertise of the user to be low as the charity deals with many underprivileged people who have restricted access to technology. Navigation and use of the website needs to be simple and intuitive. Webpages will be distinct and scrolling will be reduced to a minimum. Each webpage needs to explain its purpose, especially the ones that contain some sort of form which the user will need to fill out.
Security	Only members with admin privileges will be allowed to access records kept about the people that the charity helps (as it is confidential), and have the ability to edit parts of the website.
Reliability	The website always needs to be running, so that people who seek help can find the information they need when they need it. In terms of the data stored for the people who the charity helps, the website needs to be robust so the data will not be corrupted or lost.
Accessibility	The website should be simple enough that anyone can operate it. We will use headings to organise the structure of the content. Colour will be used to organize content but we have to make sure that colour-blind people can still view and distinguish the content.
Quality	The website should have no bugs and it is essential that the design is modern and minimalistic. Style and colours need to be consistent throughout the website and the webpages need to be concise. Images and video elements should be used to promote the charity’s message.

Capacity	Their Facebook group has almost 1000 members so the website needs to support around that many although this might change as the charity grows.
Scalability	This requirement is covered by the server that the website is hosted on and not the website itself.
Interface	Users will input data, whether this is in terms of filling out a form or getting data into a database. This data does not need to be parsed in such a way that a script could handle it. Such data will only need to be presented in a human readable format.

Open Issues

Vision	The client has shown keen interest in the idea of the website taking the user through a ‘journey’, this however is likely to make for a less intuitive user experience if taken too far and after further consultation with Beyond Limits we have decided to favour accessibility and ease of use in our approach and design a free-flowing and intuitive layout. There is a risk that the client will think the first prototype will be too impersonal and ordinary, setting back the front-end team as they may need to rethink their approach.
Manageability	The client will have to take charge of administration of the website once it has been completed. Tools will be provided to make this as easy as possible in the form of an administrators panel. However since the client isn’t technically proficient they may still have trouble updating the content of the website. To try and avoid this the first prototype should include an outline of the administrator’s page interface to get feedback as to how it can be improved.
Accessibility and Usability	The website will be from a variety of different devices. We may encounter problems with how webpages load on devices of different sizes and input methods. We will test on mobile and on desktop to ensure all features are usable on different screen sizes and different input devices (touchscreen, mouse and keyboard)
User Experience	All manner of people will be using the website, those seeking the charity’s help, volunteers and the administrators. None will necessarily have much if any IT expertise so we must get feedback from the client to ensure the website is well suited to it’s audience
Hosting	We will not be hosting the website ourselves so don’t have to worry about hardware. Instead we will be using a ‘cloud’ hosting service such as Amazon web services which is delocalised and boasts very high uptimes which would not be possible with local hosting.
Software and Data Security	If any security flaws are found in the underlying software (for example SSL’s Heartbleed) used to build the website after it is deployed, it may be left vulnerable as the client might not be able to update and apply security fixes. The client wishes to be able to save and access potentially sensitive documents containing personal and confidential information. This

	may have legislative implications due to for example, GDPR.
Development	There is a risk that work could be lost if code and documentation are not saved at regular intervals. Auto-saving will be used to minimise this risk as it is a feature on most development environments. Having multiple people working on the same codebase can mean it is difficult to track down where software breaks and bugs are introduced. Git version control will be used to track this and allow rollbacks.

Development approach

For this project, aesthetics and functions are key. We need the freedom and scalability in order to create a great user experience for the 2 main groups of people using this website, but we also need the system behind it to be robust and dependable.
For this we've decided to have an off-site piece of hardware to act as a server and host the website. An alternative would be to host the website on-site, but our clients have made it clear to us that there is no broadband connection near them and their data service is insufficient. The system will be a lot more dependable if we host it off-site, with a stronger connection and our clients can access it remotely.

To create the best possible user experience, we've elected to use high-end website development tools. Bootstrap is a front-end open-source framework for developing websites using HTML and CSS. One of the main features of Bootstrap relevant to this is its portability to mobile devices, this will allow us to do a lot with the limited knowledge of web development we currently possess. Instead of this, we could use standard HTML, CSS files with JavaScript to back it up but the website would not be nearly as interactive and visually interesting to fit the client's specifications and the user's needs.

To implement this software, we're going to need the best Integrated Development Environment. Microsoft Visual Studio is frequently updated, fully featured with everything we could possibly need to develop our back-end features. Our front-end development will use SublimeText as a text-editor in order to create our HTML and CSS files, and script in the Bootstrap library. Notepad or Notepad++ could be sufficient but SublimeText creates an easier workflow for our developers.

In order to maintain Version Control, we plan to use GitHub. This will allow us to upload our source-code and keep track of all changes using Git. BitBucket or SourceForge would be viable as a replacement but BitBucket isn't free for groups of 5 or more, so it wouldn't scale very well if the project was continued by a larger team and SourceForge is a lot less user-friendly and would slow down our development time for people who aren't as familiar with VCS.

We plan to use Test Files to as a testing approach to our server, making sure pages and resources load correctly as well as testing to see that our functional requirements are met. Postman is an alternative to this, a simpler alternative which would create a series of tests based of our own parameters. Unfortunately, we would have a lot less control over what we want to test and how we test it. Seeing as some of the features of this website might be less conventional, due to the nature of the client, we feel like a general approach wouldn't cover all the testing parameters of the project.

Terms

- **Admin**- Back-End – a user with higher access privileges. Can login and make changes to the site.
- **Attributes** – type of field.

- **Back-end** – the server and administrators webpage to edit the front-end.
- **BitBucket** and **SourceForge** – alternative to GitHub
- **Bootstrap** – framework for presentation of websites.
- **Bugs** – errors in code leading to unintended consequences.
- **Carousel** – a method of displaying images; works by scrolling through all the images horizontally.
- **Cloud** – hosted on the internet. Distributed across many servers which are connected to the internet.
- **Codebase** – all of our code.
- **CSS** – cascading style sheets. Used for styling.
- **Data** – information stored on a subject.
- **Database** – a persistent storage location for various data.
- **Deliverables** – discrete items which contribute towards the overall system.
- **Drop-down menu** – a type of menu which when a button is hovered over, drops down with a list of options.
- **Field(Database)** - pieces of data which corresponds to a record
- **Form** – a set of inputs for the user to put data into the system.
- **Framework** – a set of functions which can be used to simplify development.
- **Front-end** – website design which the user will see
- **GDPR** – Data protection regulations
- **Git Version Control** – See GitHub.
- **GitHub** – version control system for code.
- **Hardware** – e.g computers; physical machines.
- **HTML** – Hypertext Mark-up Language
- **Integrated Development Environment /IDE** – an application used for developing interactive applications, websites, etc.
- **Javascript** – scripting language used for automation and other effects.
- **Links** – routes to files, webpages, etc on the internet.
- **Off-site** – not on the local machine.
- **PHP** – programming language used on the back-end for servers.
- **Parse** – Edit data to fit a particular format which makes it easier to use.
- **Postman** – testing application for server.
- **Prototype** – a small section of the system, generally done in iterations.
- **Public page** - a page intended to be accessed by a user.
- **Record** – A collumn in a database

- **Response Time** – time taken for the page to load on the user's browser.
- **Scalability** – changes in resolution do not result in certain aspects of the page becoming unusable.
- **Server** – computer/set of computers which hosts the webpages.
- **SSL Heartbleed** – SSL(Secure Sockets Layer, now TLS) security vulnerability
- **SublimeText** – text editor
- **TLS** – Transport Security Layer.
- **UI** – User Interface
- **User base** – the set of all users.
- **VCS** – version control system.
- **Webpage** – a page which the user can view in their browser.

References

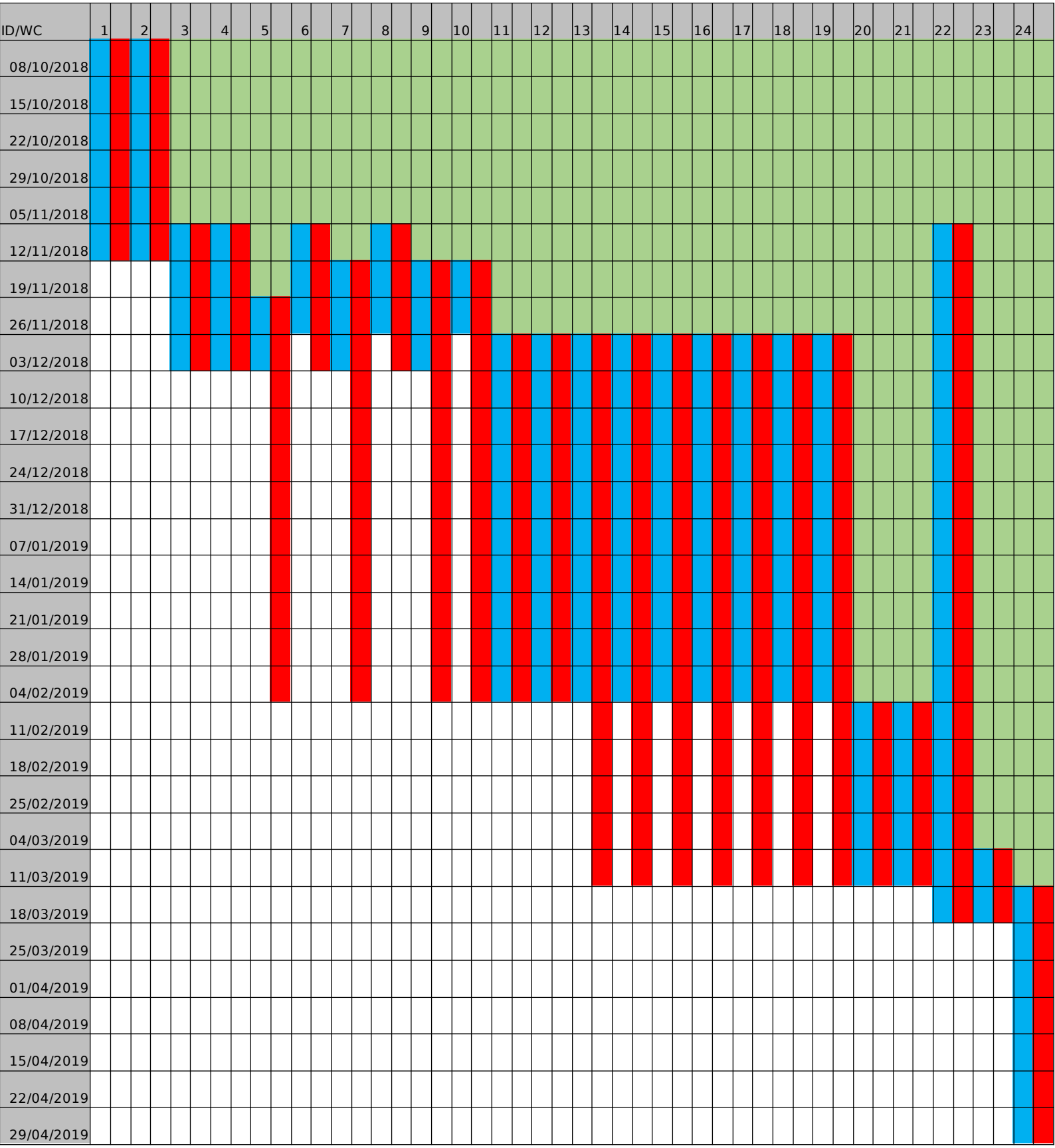
- Facebook, (**2004**). *Groups Pages*. [online] Available at: http://www.facebook.com/groups [Accessed 06 11 2018].
- WordPress, (**2003**). *Admin Side*. [online] Available at: http://wordpress.com [Accessed 06 11 2018].
- Christians Against Poverty, (**1996**). *Whole Website*. [online] Available at: http://capuk.org/ [Accessed 06 11 2018].
- SublimeText, (**2008**). *SublimeText Text Editor.* version 3.1.1, Software, Available at: http://www.sublimetext.com/3 [Accessed 09 11 2018]
- Visual Studio, (**2016**). *Visual Studio IDE*. version 1.28, Software, Available at: http://code.visualstudio.com/ [Accessed 09 11 2018]
- GitHub (**2008**). *VCS*. [online] Available at: http://github.com/ [Accessed 09 11 2018].
- SourceForge, (**1999**). *VCS*. [online] Available at: https://sourceforge.net/ [Accessed 09 11 2018].
- BitBucket, (**2010**). *VCS*. [online] Available at: https://bitbucket.org/ [Accessed 09 11 2018].
- Postman, (**Unknown**). *API Testing Environment*. [online] Available at: http://www.getpostman.com/ [Accessed 09 11 2018].

Project Schedule:

Requirement	Start Date	Soft Deadline	Hard Deadline
01-Requirements Specification	8/10/2018	15/11/2018	15/11/2018
02-Contribution Matrix 1	8/10/2018	15/11/2018	15/11/2018
03-Prototype 1 (non-functional)	15/11/2018	07/12/2018	07/12/2018
04-Contribution Matrix 2	15/11/2018	07/12/2018	07/12/2018
05-Online help form	01/12/2018	07/12/2018	08/02/2019
06-Base Webpages + Access/Menu bar	15/11/2018	23/11/2018	07/12/2018

07-Video feature	23/11/2018	07/12/2018	08/02/2019
08-Admin Webpage view	15/11/2018	23/11/2018	07/12/2018
09-Carousel	23/11/2018	07/12/2018	08/02/2019
10-Database interface	23/11/2018	01/12/2018	08/02/2019
11-Prototype 2	07/12/2018	08/02/2019	08/02/2019
12-Contribution Matrix 3	07/12/2018	08/02/2019	08/02/2019
13-“More Info” Feature	07/12/2018	08/02/2019	14/03/2019
14-Animation	07/12/2018	08/02/2019	14/03/2019
15-Donation button	07/12/2018	08/02/2019	14/03/2019
16-Carousel editor	07/12/2018	08/02/2019	14/03/2019
17-“Upcoming events” editing features	07/12/2018	08/02/2019	14/03/2019
18-Webpage editor	07/12/2018	08/02/2019	14/03/2019
19-Online Help Form editor	07/12/2018	08/02/2019	14/03/2019
20-Final Delivery	08/02/2019	14/03/2019	14/03/2019
21-Contribution Matrix 4	08/02/2019	14/03/2019	14/03/2019
22-Test report	15/11/2018	22/03/2019	22/03/2019
23-Contribution Matrix 5	14/03/2019	22/03/2019	22/03/2019
24-Team Project Sign-off	22/03/2019	29/04/2019	29/04/2019

Gantt Chart



WC – Week Commencing, Red – Hard Deadline, Blue – Soft Deadline, Green – From start date to deadline.