­­

Project Management. Simple.

MAJOR PROJECT REPORT (DRAFT)

INTERACTIVE MULTIMEDIA DESIGN  
Nathan Nelson // B00608772

**INTRODUCTION**

**OVERVIEW**

This report will cover all aspects of my progress throughout the Major Project, from the idea of a project management tool for freelancers, through to the design and development of the fully functional web application known as Workk.

**BACKGROUND**

In the United Kingdom alone, there are approximately 1.88 million people working freelance [crunch.co.uk, 2015]. This figure has been rising every year since 2011, when it was only 1.4 million. This shows just how many people are now working taking up work as a freelancer.

The concept of this major project is an online tool for freelancers to efficiently manage their time, and manage their various ongoing projects.

This was an idea that came to me through my own experiences. Since starting University, I have been working freelance, completing various web and graphic projects for clients across Northern Ireland. Whilst doing this, it was key that I kept track of everything, ensuring that nothing was missed. This was especially the case when working on more than one project at any given time.

**CONTEXTUALISE**

This concept will be aimed mainly towards those who work freelance within the multimedia industry. It is the perfect way for a freelancer to keep track of the projects they are working on, whilst also keeping on top of their workload.

This is a concept that will be created with the audience in mind, and coming from a similar background myself, I hope that this will help me to provide the perfect solution. Even though the initial aim of this concept is for use by those within the multimedia industry, there is opportunity for it to be used by absolutely anyone.

**SIMILAR SOLUTIONS**

This concept will differ from the solutions already out there, due to the high focus on user experience and design. By working in the industry, I know just how important it is that a user feels comfortable using a system, and if they cannot understand how to carry out a simple task, they will simply try another.

Before carrying out any development, I was keen to find out a bit about the similar solutions that already existed, and find out what I could do to ensure my concept met the user’s requirements.

Below are a few existing services that I found:

*Solo*  
<http://www.getsoloapp.com/>

Solo is an online project management tool designed specifically for freelancers. This makes it very similar to project I will be creating, so provides a good place to start in terms of existing products. Solo allows its users to create projects and set due dates, as well as create a client roster.

*Teamwork*

<https://www.teamwork.com/>

Teamwork is another online project management tool, but it is mainly targeted at large teams. This allows teams of people to work together, and track what each of them are doing within certain projects. This allows provides a platform for them to communicate with each other. This tool ensures that each person within the team knows exactly what he or she need to be doing on a particular day, within a particular project or piece of work.

*Basecamp*

<https://basecamp.com/>

Basecamp is probably one of the most well known online project management tools that there is. This is a tool that I personally have experience working with. Again, this tool is aimed mainly at managing large teams. Similar to the other services, Basecamp allows the users to create projects and assign tasks to particular team members. There is also a feature that allows clients to have access to particular projects, allowing them to view which tasks have been completed. The administrator sets the amount of detail that is shown to the client, therefore restricting what they can and cannot see.

From looking at existing solutions, I have found that they all have a lot in common, but still do not quite meet the requirements of a freelancer. ‘Solo’ is the only website that I found that aimed their services at freelancers. This website provides the users the opportunity to track each of their projects, complete set tasks and also upload files onto the system. Out of all of the websites that I found, this is the one that provides the closest solution to the problem I am aiming to solve.

**AIM**

The aim of this project is to create an online system for freelancers to keep track of their work. I aim to deliver this is in a simple, easy to follow manner, that features a beautiful front-end design. More specifically, this project will help freelancers to keep their clients up to date with project progress, as well as allow them to ensure they are managing their time correctly across multiple projects.

**OBJECTIVES**

The objectives for this project are listed as follows:

* Develop a clear user interface design, based on UX research and user requirements.
* Implement a feature that allows the user to create projects, assign tasks to them, and complete  them as they work their way through the project.
* Implement data visualization in order to show project progress and make up the dashboard.
* Create a user authorisation feature, allowing new users to register for an account, and existing users to log in.
* Develop a feature for storage of client information, including contact information and any associated projects
* Build a timing feature. This will allow the user to track their time, and log it against the corresponding project.
* Implement a profile feature. This will allow the user to log into their account using their selected username and password combination.
* Build a structured, secure database to store all of the application data.
* Focus on the UX of the website, ensuring it is accessible.
* Ensure the application is viewable across multiple screen sizes and devices.
* Test the application across multiple browsers

**SCOPE**

Time**:**

 Time is an obvious boundary with this project. I must work to meet all deadlines in order to stay on track, and manage my time accurately. In order to manage my time correctly, I must hard work on each stage of the process and ensure that everything is completed on time and to a high standard. Time is also limited, due to studying other modules. This again is a challenge on my time management skills.

 Cost**:**

 The cost of the project will be kept to the very minimum. The only cost involved in the development of this application will be for the purchase of the web domain. This will help the application to have an identity online and will fit closely with the branding. There is also the possibility of purchasing a hosting package to home the application. This could be used to further develop the application at a later stage, depending on the capabilities of the university servers.

 Quality**:**

Throughout my time in university, I have been working to develop professional and high quality projects for the web. Last year, I completed a one-year industrial placement as a Creative Developer and I hope to use the skills I learnt here, as well as those I have learnt during the course of my degree, in order to ensure the project is completed to a high standard.

Resources**:**

In terms of resources, there are many that I have at my disposal thanks to the University. As well as this, I also have access to a Macbook Pro, which I can use for development purposes, and also an iPhone which can be used for testing the mobile capabilities of the website. I also have access to text editor software, which will allow me to develop the website in a way that is familiar.

**RISKS**

When building this project, it is almost certain that I will run into some sort of problem, either small or much bigger. One of the obvious challenges from the outset is time. Time will very quickly run out and I just have to work hard to keep everything on track. This means meeting all deadlines, whether hard or soft. This will require me to ensure that my time management is perfect.

 Another challenge is ensuring that I implement every feature, but ensure that they work seamlessly together. This means ensuring that the development of each of the features is correct. This would include ensuring that the timing feature works alongside each project, and that the client dashboard is easily viewable and accessible by the client.  In order to ensure that the website works seamlessly, I will have to implement a database. This will be used to store all information associated with the application including profiles, project information and the dashboards.

A possible risk associated with this would be that certain features may not be possible to  implement within the University servers. This is something that I will not become aware of until the development process starts, although if it does occur, the option of purchasing external hosting would solve this issue.

Ensuring that the website works seamlessly across all devices will also be a challenge. This will involve developing a version of the website that is suitable for mobile viewing. This is a different user experience, but it must be made to feel familiar to the desktop version.

One final risk that could arise from this project is security. Due to dealing with user accounts, and also dealing with client information, I need to ensure that all information is secure and cannot be accessed by any unauthorized users.

If I can overcome these challenges along the way, then there is no doubt in my mind that I can achieve my initial aims and objectives.

**CONCEPT DEFINITION**

**IDEA GENERATION**

Before selecting an idea, I came up with a few different choices. This allowed me to analyse each of them, and choose the best project, and the one that I felt had the greatest potential.

Below are the ideas that I came up with during the initial idea generation stage, but decided against developing any further.

**Idea One – Fitness Game**

This concept is a mobile application that encourages people to stay active through the concept of an interactive game. When trying to achieve fitness goals, whether it is to loose weight, or gain muscle mass, it can be hard to stay on track. This application starts off by giving the user the option of a few characters to select from. Each character has their own set of goals. Once the user has selected their character, the game will begin.

As the user completes goals, they are awarded points, and therefore move their way through the levels. As they progress through the levels, their goals become more difficult, until they reach their end target.

The user can use the app to record their own activities. This will result in them being awarded with points for their character. These activities can be anything from walking, to weight lifting.

**Idea Two – Concert Alerts App**

This concept is a mobile application that allows music lovers to be alerted when their favourite artists are playing in their area. On first install, users are asked to store their favourite artists, but these can be added to or edited at any time.

The application picks up on the users location, and if there is a concert date released in their area that matches one of their favourite artists, then they will be notified via push notifications. They can then find out information on the concert, including where to find tickets and when they will be released.

Another feature of the application would be to store information on upcoming concerts that the user is attending. When the user buys tickets to an event, they can add it to their upcoming events. This feature would also provide the user with a live countdown.

**METHODOLOGY SELECTION**

When developing a project, the chosen methodology helps you to manage each stage of the process. There are a few different methodologies to be considered, some of which include:

**Waterfall –** A plan driven lifecycle model, which includes a clear set of requirements, and a clear schedule for development. This method allows for a detailed outcome, with only one component being completed at one time, before moving onto the next.

**Agile –** An iterative approach to development, where each component can be viewed as a mini project and are launched over shorter periods of time. This method ensures that a product is released quickly, but it relies a lot of each member of the team interacting well, in order for it to be successful.

**Prototyping –** A complex model, where a series of prototypes are used.Prototypes are designed, developed and reviewed until they are perfect, before being developed into the main application.

**CHOSEN METHODOLOGY**

After researching each of the methodologies, I chose to follow the **Waterfall** model. As a small University project, the waterfall model works well and is very easy to follow. The requirements of the project are developed in a way that fits the waterfall model, allowing me to develop each stage of the concept, before moving onto the next. This will allow me to ensure that each feature is coded well, and that each feature runs smoothly for the end user.

**REQUIREMENTS SPECIFICATION**

The requirements for this project were mapped out the Volere Template. This is a very popular template and guide used to produce requirements specifications. These are laid out in the form of Volare ‘Snow Cards’.

**Functional Requirements**

|  |  |
| --- | --- |
| **Requirement #: 1.0** | |
| Description | The product will have a login form |
| Rationale | To allow registered users to access their account |
| Fit Criterion | The user will gain access to their account, using the credentials that were set up during the sign up process |
| Dependencies | Requires the user to have previously signed up |
| Priority (5 being the highest) | 5 |

|  |  |
| --- | --- |
| **Requirement #: 1.1** | |
| Description | The product will have a sign up form |
| Rationale | To allow users to register for an account |
| Fit Criterion | The user will create a username and password combination, allowing them to gain access to their account |
| Dependencies | The user will be required to enter personal details and create a username and password |
| Priority (5 being the highest) | 5 |

|  |  |
| --- | --- |
| **Requirement #: 1.2** | |
| Description | The ability to log out |
| Rationale | To ensure that no unauthorized users can access the account |
| Fit Criterion | Once the user has clicked log out, they will be returned to the main product website, where they can log in again if required |
| Dependencies | User is required to be logged in |
| Priority (5 being the highest) | 5 |

|  |  |
| --- | --- |
| **Requirement #: 2.0** | |
| Description | The ability for a user to edit their own profile |
| Rationale | To allow users to keep their profile up to date, as well as controlling their login credentials |
| Fit Criterion | The user will be able to update their profile details, including their login credentials. |
| Dependencies | The user will need to have an account |
| Priority (5 being the highest) | 5 |

|  |  |
| --- | --- |
| **Requirement #: 2.1** | |
| Description | The ability for a user to change their password |
| Rationale | To ensure a user can access their account if they forget their password |
| Fit Criterion | The user will be asked to answer a security question and if correct will be able to reset their password |
| Dependencies | The user will need to be able to answer a security question, set during the sign up stage |
| Priority (5 being the highest) | 5 |

|  |  |
| --- | --- |
| **Requirement #: 2.2** | |
| Description | The ability for a user to delete profile |
| Rationale | To allow users to remove their accounts if necessary |
| Fit Criterion | The user will be able to remove their account, therefore removing all credentials from the system |
| Dependencies | The user will need to have previously owned an account |
| Priority (5 being the highest) | 4 |

|  |  |
| --- | --- |
| **Requirement #: 3.0** | |
| Description | The product will provide a dashboard for each user |
| Rationale | To allow users to keep track of and review their progress |
| Fit Criterion | The dashboard will be the first stage after logging into the website. This is where the user will be able to view all their projects, and review their progress. |
| Dependencies | The user will need to have added projects to their dashboard |
| Priority (5 being the highest) | 5 |

|  |  |
| --- | --- |
| **Requirement #: 4.0** | |
| Description | The ability to add projects |
| Rationale | To allow the user to keep track of that particular project, ensuring all work is completed |
| Fit Criterion | Adding projects will allow the user to record time against it, create specific tasks, and complete them accordingly. |
| Dependencies | No Dependencies |
| Priority (5 being the highest) | 5 |

|  |  |
| --- | --- |
| **Requirement #: 4.1** | |
| Description | The ability for a user to edit a project |
| Rationale | Allowing users the ability to edit a project, including the client name, project name and completion due date. |
| Fit Criterion | The user will be able to edit the information about a particular project. This will then be updated across the website. |
| Dependencies | A project must already have been created |
| Priority (5 being the highest) | 4 |

|  |  |
| --- | --- |
| **Requirement #: 5.0** | |
| Description | The ability to create tasks |
| Rationale | To outline each stage of the project, and allow the user to better review what stage the project is at. |
| Fit Criterion | Tasks are created and assigned to a particular project. This allows the user to mark them as complete when that particular stage is complete. |
| Dependencies | A project must have already been created |
| Priority (5 being the highest) | 5 |

|  |  |
| --- | --- |
| **Requirement #: 5.1** | |
| Description | The ability to remove a task |
| Rationale | Ability to remove a task if it no longer needs to be completed |
| Fit Criterion | User will be able to remove a task at the click of a button, therefore removing it from the to-do list |
| Dependencies | The user will need to have an account |
| Priority (5 being the highest) | 5 |

|  |  |
| --- | --- |
| **Requirement #: 5.2** | |
| Description | The ability to mark tasks as complete |
| Rationale | To ensure that the user can accurately review project progress |
| Fit Criterion | Once a particular task has been completed, the task can be marked as done. Thus, allowing them to move onto the next stage of the project |
| Dependencies | The task must have been previously created |
| Priority (5 being the highest) | 5 |

|  |  |
| --- | --- |
| **Requirement #: 5.3** | |
| Description | The ability to edit a task |
| Rationale | To provide further information on the task on hand, especially if the outcome has changed from when the task was added |
| Fit Criterion | The user will have the ability to edit a task, if the intended outcome has been changed by the client |
| Dependencies | The task must have been created |
| Priority (5 being the highest) | 5 |

|  |  |
| --- | --- |
| **Requirement #: 6.0** | |
| Description | Ability to log time against a project |
| Rationale | To allow users to record how long they spend on a particular project or task. |
| Fit Criterion | Time is recorded against a particular project, and total time completed appears on the main project dashboard |
| Dependencies | A project must have been created |
| Priority (5 being the highest) | 4 |

|  |  |
| --- | --- |
| **Requirement #: 7.0** | |
| Description | The ability to add client profile |
| Rationale | Allows user to keep store information on all of their clients |
| Fit Criterion | Information on clients is stored, including their contact information. Also allows clients to be directly linked to projects |
| Dependencies | No Dependencies |
| Priority (5 being the highest) | 5 |

|  |  |
| --- | --- |
| **Requirement #: 7.1** | |
| Description | The ability for edit a clients profile |
| Rationale | To ensure that client information is up to date |
| Fit Criterion | The user will be able to update the clients information, ensuring that all details are accurate |
| Dependencies | A client profile will need to have been created |
| Priority (5 being the highest) | 5 |

|  |  |
| --- | --- |
| **Requirement #: 7.2** | |
| Description | The ability to remove a client profile |
| Rationale | To ensure client list is up to date |
| Fit Criterion | The user will be able to remove client profiles that are no longer required |
| Dependencies | A client profile must have been created |
| Priority (5 being the highest) | 5 |

|  |  |
| --- | --- |
| **Requirement #: 8.0** | |
| Description | Ability to share project progress with client |
| Rationale | To allow clients to accurately review the progress being made on their project. |
| Fit Criterion | On the completion of major milestones in the project, the progress can be reported to the client |
| Dependencies | Client will need to have been added to the system |
| Priority (5 being the highest) | 4 |

**Non-Functional Requirements**

**Look and Feel**

*Appearance*

* The product will appear minimalistic
* The product will appeal to all users
* The product will have its own unique branding, helping it to stand out and be recognized.

*Style*

* The product will look professional
* The product will be easy to use

**Usability**

*Ease of Use*

* The product will easy to navigate for all levels of users
* The product will clearly display project information
* The product will clearly outline tasks to be completed
* The product will entice users to create an account and use the service
* The product will help users to accurately keep track of their project progress

*Personalisation*

* The users will be able to upload a profile photo to their account, which will be viewable on the Dashboard once they log in.

*Learning*

* The product will feature a website tour on first arrival. This will ensure that the users know exactly how to use each of the features.

*Accessibility*

* The product will contain alternative text where images are used
* The product will be built using valid HTML
* Careful consideration will be taken when it comes to colour to ensure that those who are colour blind do not have a hindered experience.

**Performance**

*Speed*

* Projects will be updated automatically across the website when edited or updated
* Tasks will appear instantaneously as completed as soon as the user selects ‘complete’
* All pages should load immediately
* All actions should be processed instantly.

*Reliability*

* The product will be available for use 24 hours a day, 365 days a year

**Operational & Environmental**

* The product will be accessible on all web browsers
* The product will be viewable on all screen sizes, including mobile devices
* The product will be used across all web compatible devices, including mobile phone, desktops, laptops and tablets.
* The product will be accessible online

**Security**

*Access*

* Only users who have created an account will be able to gain access to the features of the product
* As admin, I will have complete control over all access

*Integrity*

* Anyone who has not created an account will not be permitted to access the product
* All forms will contain correct validation to ensure that users details are stored correctly and safely
* All passwords will be encrypted

*Privacy*

* During sign up, users will be made aware of all information that they are required to provide
* Users details will not be available to other users

**Cultural**

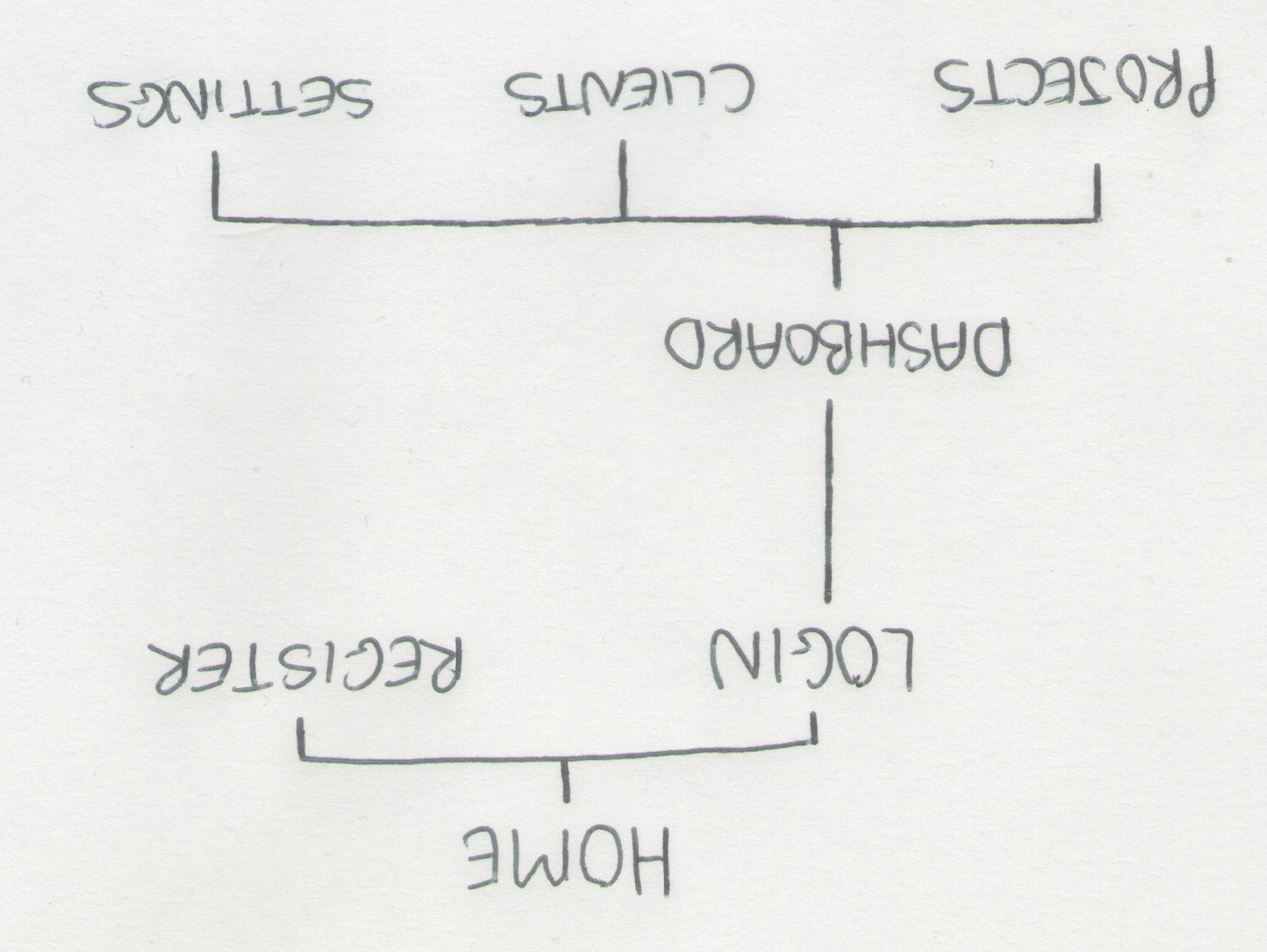
* The product will be created in a manner that does not cause offense to any religious or ethnic groups

**Legal**

* All users details will be obtained in accordance to the Data Protection Act.

**PAPER PROTOTYPING**

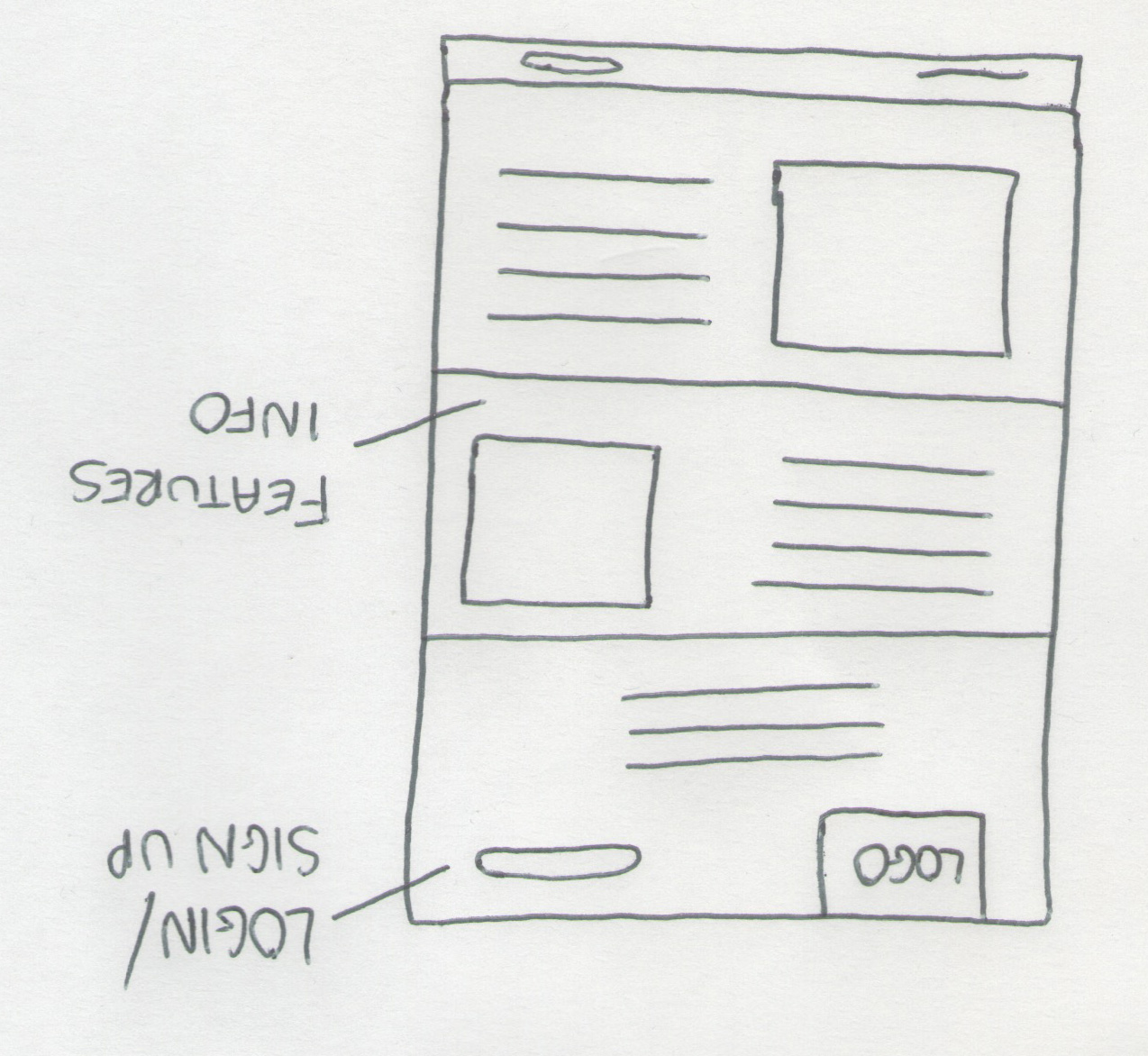
**SITE MAP**

****

Before starting on any layouts or designs for the website, I decided to think about how the site would be structured. This was done using a site map. This shows each of the pages, and how they are all linked. This included the main index page, explaining the product, and the project dashboard itself, which is accessible once the user logs in.

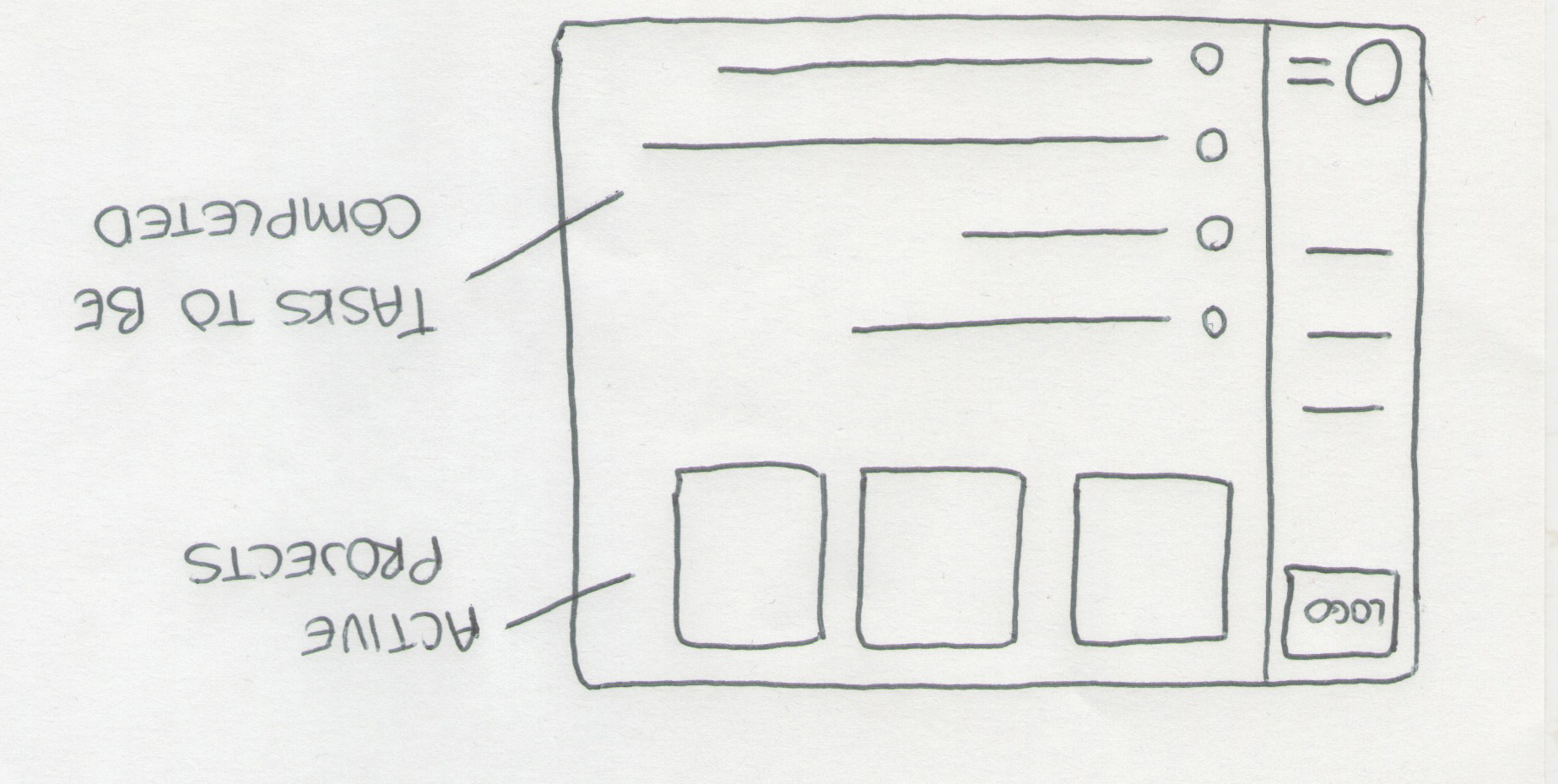
**HOME PAGE**

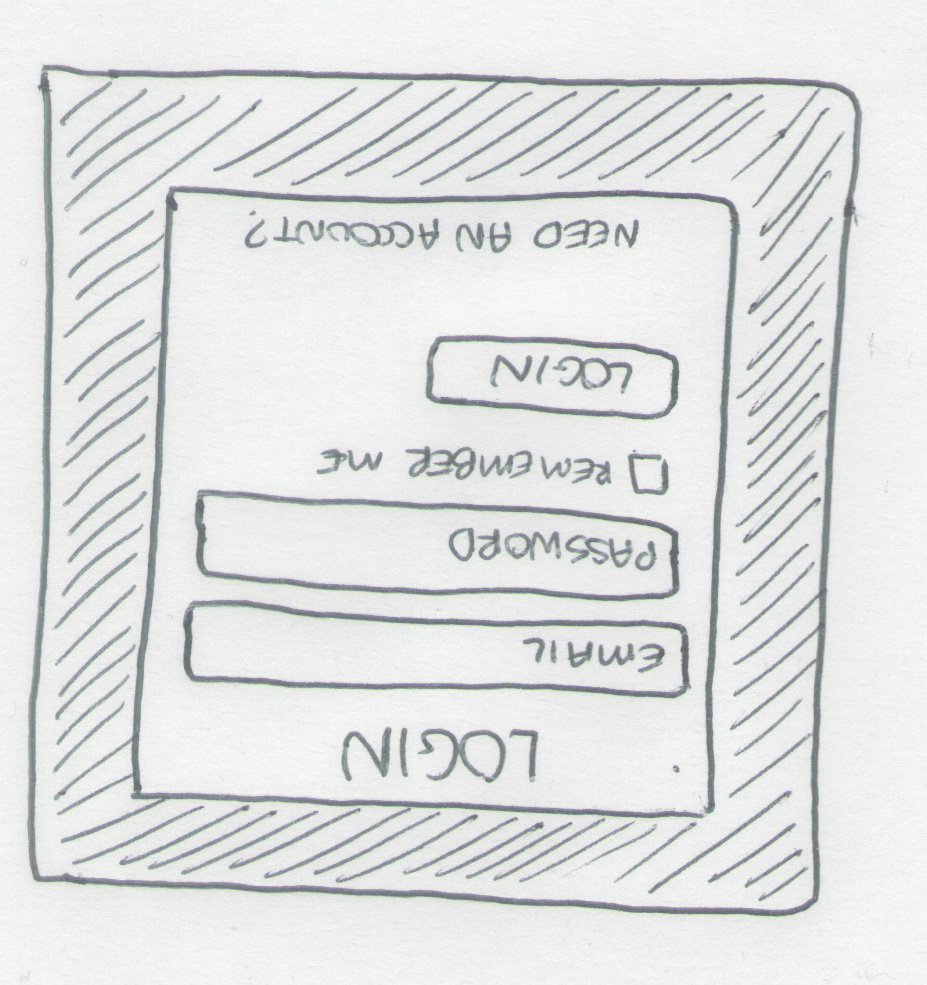
The first page that I sketched was the index page. This is the page that the user sees first when they visit the website. For first time users, this gives then an opportunity to find out a bit about the service, before signing up. For a user who already has an account, they can log in from here and gain access to their project dashboard. There is a heavy focus on design on this page, due to it being the first thing that people will see on visiting the site. This page will be developing with the idea of turning viewers into users.



**DASHBOARD**

Once the user has logged in they will gain access to their own personal dashboard. This will contain the projects they are currently working on, and tasks that need to be completed. Each page will include the Workk branding, as well as the user’s profile information and avatar.



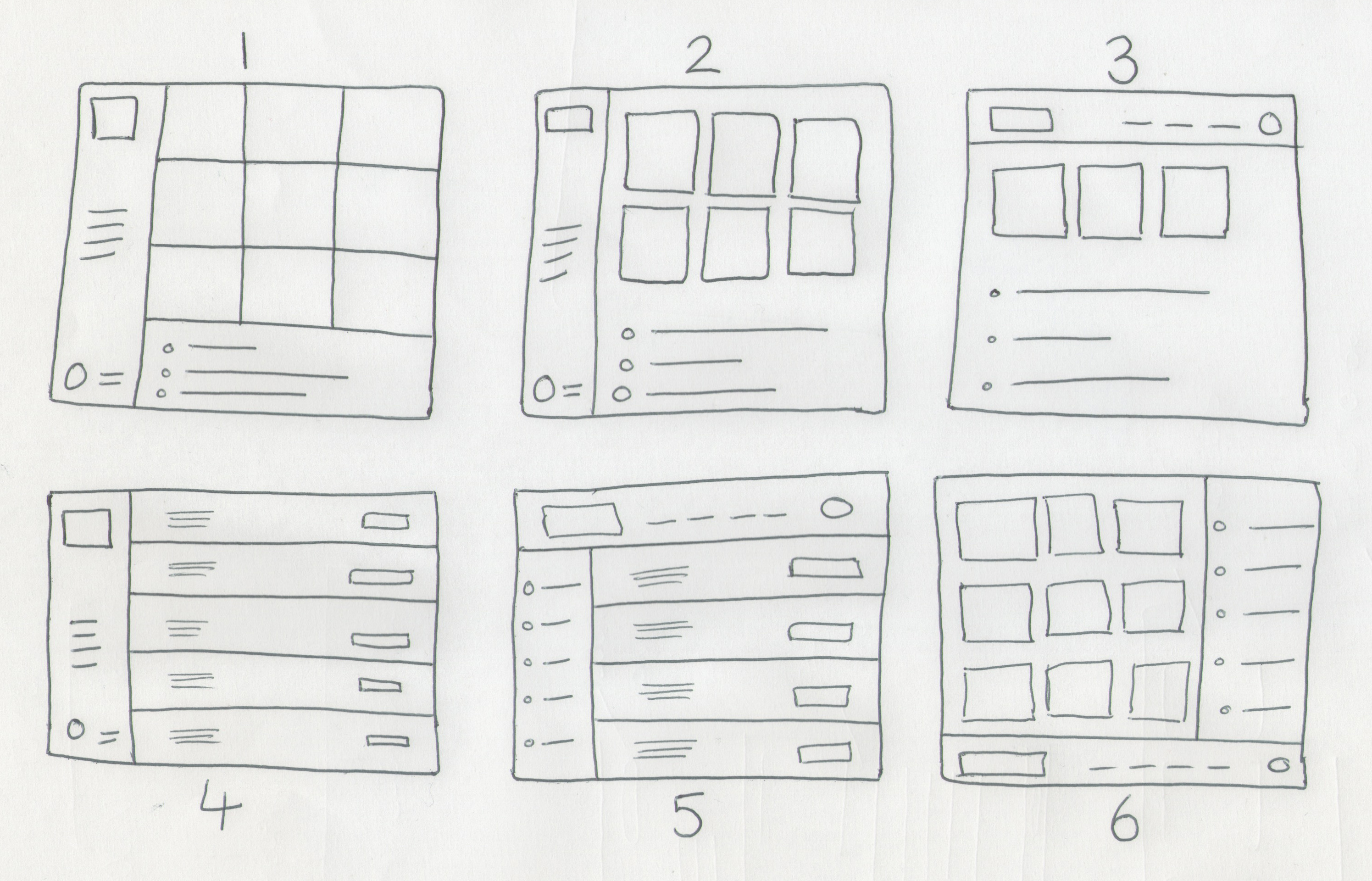


**LOGIN & REGISTER**

In order to use Workk, the user must create an account. This is accessible from the main home page. When registering, the user is asked to provide their email address, various personal details, and also create a password. Once they have registered, they can log into their personal dashboard, and begin using the features.

**6UPS**

After the initial sketches, I decided to rethink the dashboard a little, and refine how it would look. I thought that the best way to do this was to create 6-ups. After sketching the 6-ups, sketch number 2 was chosen as the layout that I would take further to review in more detail.



**WIREFRAMES**

The final iteration of the paper prototype was to create a set of detailed wireframes. These allowed me to refine a few features and show a more detailed version.

For the wireframe stage, I decided to focus more on how the mobile version of the website would look. Now that the functionality for the main site had been refined, this was a good time to think about the mobile experience. Wireframes allowed me to show how each of the main features of the concept would look across the website.

INSERT WIREFRAMES HERE

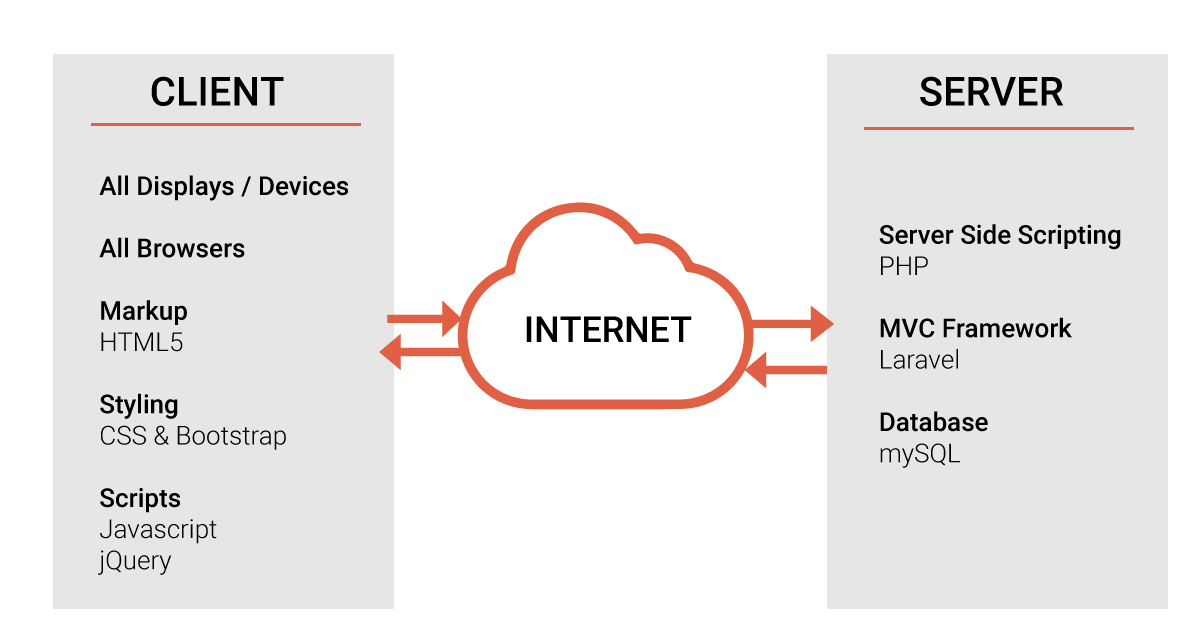
**DESIGN**

**SYSTEM DESIGN**

Before starting the design of the website itself, I began to look at how the system would work, enabling each of the features to work seamlessly on the front-end.

**CLIENT-SERVER MODEL**

The client-server model outlines the network between the client and the server. The model outlines the various components on the client side and shows how they interact with those on the server side. The client-server model for this project is outlined below:



**DATA DESIGN**

**DATABASE DESIGN**

Now that the wireframes had laid out how each of the features would be displayed on the front-end, I took some time to consider how the data would be interpreted in the back-end. This involved looking at how the database would come together.

In order to get a better feel for the system, and help me to better understand how the database would be set up, I started by deciding upon the tables that would be required.

The first table is the user table. This contains all of the information about each user, including their email address, password, and also some personal information that they provide during the registration process. Each user is also assigned a unique ID, which is used to uniquely identify them. This is given as the primary key of the table.

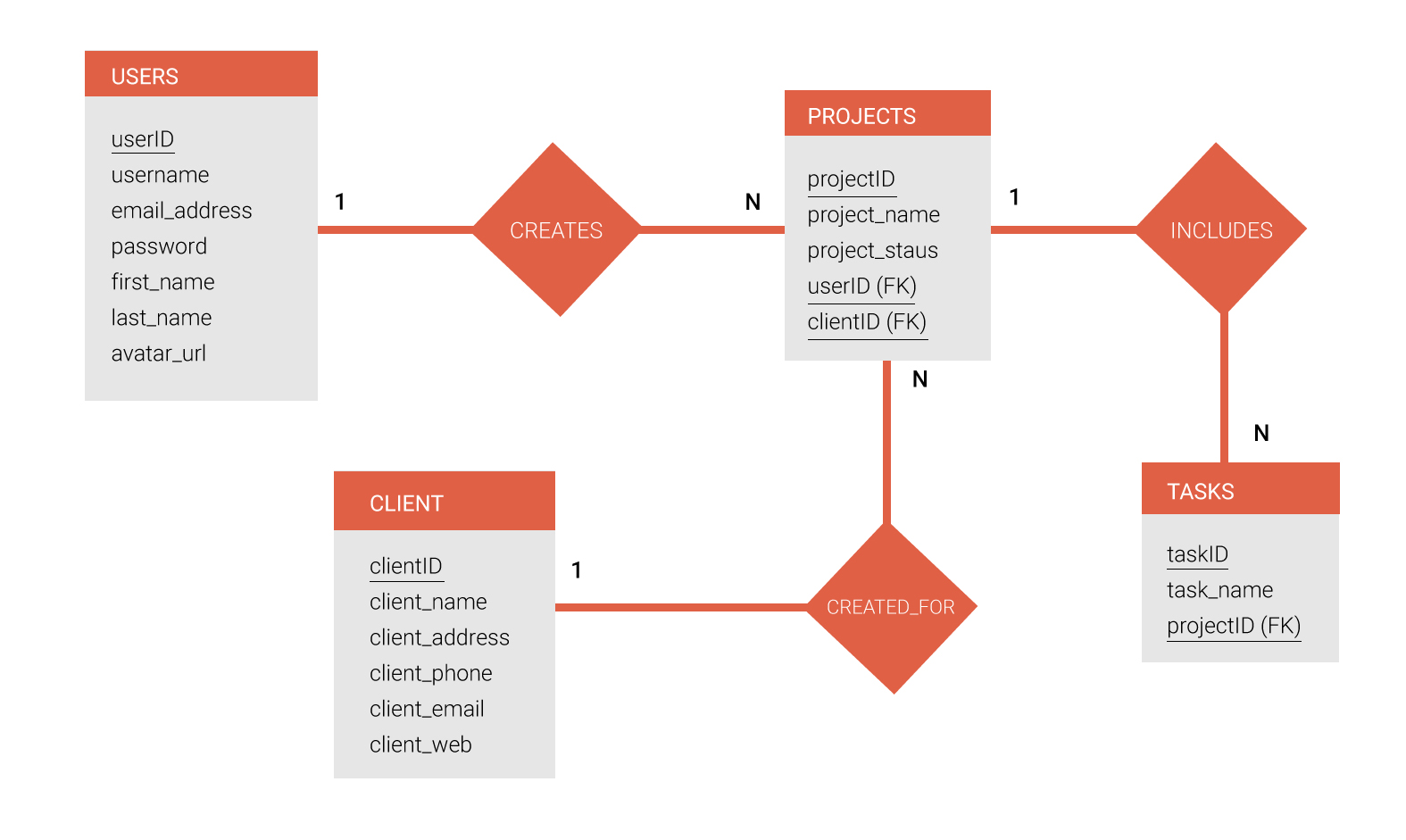
A table is also required for the projects. This will include a unique ID for each stored project, the name of the project, the project status and its due date. As well as this, the user ID is included as a foreign key. This ensures that the project is linked to the correct user and displayed within their personal dashboard.

The next table is the tasks table. This is a table which includes all of the individual tasks that the user sets for each project. The fields for this table include the ID and the task detail. The foreign key in this instance is the project ID. This links the task directly to a particular project, which in turn is linked to a user.

The final table is the clients table. This allows the users to collect a raster of clients, and link projects to specific clients. This table will include a unique ID, the client name and the clients contact information.

**ENTITY RELATIONSHIP DIAGRAM**

In order to get a feel for how each of the tables of the database would come together, an entity relationship diagram was created. This shows each of the relationships between the tables, and how the data is connected.



**IMPLEMENTATION**

**TECHNOLOGY SELECTION**

**SERVER-SIDE**

**Scripting**

Due to focus on my experiences being on front-end development, server side scripting is not something that I commonly work with. For the purposes of this project though, this is essential. In order to stick to what I know, I decided upon PHP. This is a server side scripting language that I am familiar with due to sitting a module in it. Due to this, this is a language that I can hopefully expand my knowledge on. PHP provides all the capabilities that I require to complete this project successfully.

**Framework**

After some consideration, I decided that the use of a server side MVC framework would be beneficial. This is something that I have had a little bit of experience working with, thanks to my time at Big Motive during my placement year. During my time there, I became familiar with the framework CodeIgniter, created by EllisLab. This a popular PHP framework that adapts the MVC model. After carrying out some research, I found that CodeIgniter was no longer the leading framework on the market. This is when I came across Laravel.

Laravel is a similar setup to CodeIgniter, as it also uses the MVC model, but it has rose in popularity in recent times. After carrying out some research into the framework, and coming across Laracasts.com, a website compiling of thousands of tutorial videos created by Jefferey Way, I decided this was the framework I would use. Due to the fact I was going to have to learn this framework from scratch, having never looked at it before, I knew that the learning resources available through Laracasts would be essential. This helped to shape my decision.

**Database**

For the database setup, I have chosen MySQL. This is a fast, secure and extremely popular database management system that is very easy to work with, and integrates perfectly with web applications. This is also a system that I am familiar with, so decided this was the perfect choice. Before coming to my decision, I did carry out some research into other alternatives. One of the alternatives I looked at was SQLite. This is a much smaller system, that is generally more suited to small single-user applications, therefore this was the reason I decided it was not the right choice for this project.

**CLIENT-SIDE**

**Markup**

In terms of markup, HTML5 was the obvious choice. HTML (HyperText Markup Language) 5 is the latest version of the language, which includes lots of new features and elements to help make writing your markup a lot easier. Introduction of tags such as <header> and <nav> save a lot of time when it comes to development.

**Styling**

The client-side styling will be handled using CSS, as well as some use of the new CSS3. There are new alternatives to CSS, such as SASS, but I decided to stick with what I am comfortable with. Due to having to learn a server side MVC framework from scratch, I didn’t want to have to teach myself something else too.

As well as CSS, I decided upon the use of Bootstrap. This is a front-end framework that allows users to create fluid, responsive layouts without the hassle of having to spend hours coding. As well as this, Laravel also supports the use of Bootstrap and it comes preinstalled with the package.