

Pist.ee - Ski/Snowboard Review Web Application

Interactive Multimedia Design - 2015

Ryan Wilson - B00579628

PSG 10 - Giuseppe Trombino

Acknowledgements

Firstly, I would like to thank my mentor Giuseppe Trombino for guidance and support throughout my project. He has helped me develop some great ideas, and made me push myself to produce a project to the best of my ability. I would also like to thank George Moore for his support in lectures and course instruction.

Also thanks to my friends and family for your support throughout the project, notably Lesley Wilson, Brian Wilson, Lyndsey Reid, Stephen Martin, Shaun Steenson, Ben Hunter, Chris McCoppin & Adam Mcpeake.

Contents

Acknowledgements	1
Contents	2
1.0 Introduction	4
2.1 Idea Generation	5
2.2 Requirements Spec	5
2.3 Paper Prototyping	20
2.4 Feasibility Testing	22
2.5 Methodology Selection	23
3.1 UX Design	24
3.2 System Design	28
3.3 Logic Design	31
3.4 Data Design	32
4.1 Technology Selection	33
4.2 Technology Use	36
4.3 Notable Achievements	40
4.4 Notable Challenges	45
5.1 Testing Approach Solution	46

5.2 Testing Process	47
5.3 Test Results	48
5.4 User Survey Responses	50
6.1 Evaluation - Evaluation of Surveys	51
6.2 Evaluation - Project Outcomes	51
6.3 Evaluation - Methodology	52
6.4 Evaluation -The Plan	52
7.1 Conclusion - Report Summary	53
7.2 Conclusion - Reflection	54
7.3 Conclusion - Reflection of my Role	55
7.4 Conclusion - Future Work	55
8.1 References	56
9.1 Appendix 1 - Functional Requirements	59
9.2 Appendix 2 - Paper Prototyping	82
9.3 Appendix 3 - Feasibility Questionnaire	99
9.4 Appendix 4 - UX Designs	103
9.5 Appendix 5 - Functional requirement testing	122
9.6 Appendix 6 - User testing form	155
9.7 Appendix 7 - User testing result Summary	157

1.0 - Introduction

Every year 1000s of people like myself, head on Ski/Snowboarding Holidays across the world.

One problem my friends and I always come across is deciding where to go. Tripadvisor, is helpful when looking at accommodation, but really isn't designed for finding out information about resorts, with no such feature for searching criteria which we find important when choosing our destinations, such as slope information or the infrastructure of the resort.

Several newspapers, and different extreme sports sites publicise articles about where is best to go, but these are not the most reliable, as they are written by one person, who may have had a bad experience etc, or may want something completely different than yourself. Browsing through these reviews also takes a considerable amount of time.

When generating the idea for a suitable major project, building an application to address these problems was always my favourite option; to make a Ski/Board specific review site, which is simple to use, allowing people to leave reviews for each holiday they go on, and then search using criteria that interests them to find the best resort for them for their next holiday. The main aim for the project is to make it as simplified as possible. People will leave reviews by answering 12 short questions, which will have a 1-5 rating, with a short optional comment box at the bottom to leave a comment about their holiday, and a photo upload section. This will be quick to do, making more people keen to leave a review. The application should work on mobile devices, as they could leave a review when waiting on a plane etc. Photo upload isn't a necessary feature, but will improve the experience for people looking at reviews.

People can then search criteria to find resorts which are best suited to their needs, as well as the site generating top lists for different criteria such as "Best nightlife" or "Best

snow parks" etc. When a user makes an account, they have an option to add favourites, allowing them to return to the site and view them at a later date.

With the project now coming to an end, it has been very successful. Some parts of the project have changed along the way, due to design/development issues. However, the project still functions as originally forecasted, and simplifies the process of deciding where to go.

This report gives a brief insight into the work undertaken throughout the year, following the process from Idea Generation through to the completion of the project.

2.0 - Concept definition and testing

2.1 - Idea Generation

With the idea of the project now decided, it is time to develop the initial thought into an idea, which then can be designed and developed. One of the most important factors was branding. The name piste was simple, and after looking at domains, the domain pist.ee was available which works really well. Branding could now begin as well as the design of the site. The system design was the first task, so that the structure of the application could be decided. Creating a flowchart allowed all the pages of the application to be mapped out, and from this page content and the database design could be developed.

2.2 - Requirements Specification

An early process of the project was developing requirements. These requirements state exactly what the project should do, and how it should do it. They go into great detail, with each feature on each page of the site being a requirement. For instance a button to submit a form would be a requirement, an input field to input your name or password would also be a requirement. For the full requirements specification for this project see (Appendix 1) . For the format of the requirements specification a template called the

Volere Template was used. The Volere Template has 15 sections, some of which were not needed in this project. These sections cover things like Maintainability & Support, Usability & Humanity, Functional and nonfunctional requirements. The requirements are very useful when designing and developing, as they can be used as a checklist to make sure the application performs as initially expected, therefore simplifying the workflow of the project.

1.Purpose of the Project

1a.Background of the Project

The purpose of this project is to fill a gap in the market, to build a high quality review site for ski and snowboarding resorts/areas around the world. Currently there are some products, but all of these have serious setbacks, due to their design, functionality or data provided. The idea behind this project is to make a simple to use review site, where a user can leave a review in less than a minute, encouraging them to do so. Current solutions do not have all the information in one place, and have many unnecessary features. Most include key information about the resort, but do not include reviews. This means a resort can sound great on paper, but may have other problems which are not recorded such as large queues, crowded slopes or a bad infrastructure. With over 300 ski resorts in Europe alone, deciding where to go is always an issue. Being a snowboarder myself, I'm faced with this problem when deciding where to go every year, which is the main reason for the motivation to build this project.

TripAdvisor has been hugely successful with reviews/rating of popular holiday destinations, but does not include ski resorts. This leaves a massive gap in the market, forcing travellers to rely on review given by ski companies, which from experience, aren't particularly accurate, as they are not going to give bad reviews about somewhere they are trying to sell holidays to.

1b. Goals of the Project

One of the main goals is the usability of the site with the leaving and accessing of reviews being a quick painless task, unlike some of the existing review sites for different industries. Reviews will be easily accessed via a range of search options, allowing the user to find the resort that is most suited to them. As well as having reviews, key data about resorts all over the world will be included in results, as well as a photo upload system for users to add photos when leaving a review. Users will be able to login to leave reviews, and save favourites resorts.

2.Stakeholders

2a. The Client

The Client for this project is myself, as I came up with the concept behind the project. When the project is completed, I will be the main administrator of the page, and the decisions of what features are included in the site will be finalised by myself.

2b. The Customer

The customer for this project will be Skiers/Snowboarders. As the group of people involved with these sports can be extremely diverse, it is hard to aim the site at one group of people. The site will be aimed at all levels of skiers/snowboarders so the reviews have to be useful for all standards of skiers and snowboarders. This means the criteria will have to be carefully selected, so that the site is viewed as a useful resource by all, and not just a small demographic group. To help select the right criteria, I will ask a range of skiers/snowboarders what criteria they think are important when booking a ski holiday, and which are not.

2c. Other Stakeholders

Other stakeholders will include Testers, who will test both the functionality of the system, and make sure all features work correctly. As I have a large range of friends and family who have Skied/Boarded several times, they will be asked to help find any problems with the usability of the site. Another tester will be myself. I will make sure the site functions correctly under different constraints.

2d. Hands on Users

The hands on users for this project will be the final customers. This will be a wide range of age groups and backgrounds, potentially from different countries. Below I have outlined some typical users.

Students - First time skiing holidays with their friends. This group of people will usually be more interested on what's happening apres ski, than on the slopes. A resort with lively nightlife and cheap accommodation/food alcohol will be suited to them. Some students will never have skied/boarded before and some will so a range of slopes is important for the user, and knowing if a resort is good for beginners/advanced skiers/boarders.

Couples - Couples can be of any age, and any experience. From people who have been skiing all their life, to people who have decided to try something new and are complete novices. They have several criteria when booking a holiday, such as nice places to eat, possibly a quiet atmosphere, or maybe evening lively apres ski. Skiing they could be interested in any sorts of slopes, whether it be beginners blues, or terrifying blacks.

Families - Families usually have a mixture of skiing abilities, so being able to get lessons for their children is an absolute must. Also childcare may be a possibility, allowing parents to still have a good holiday on the slopes. Cost may also be a big issue for families.

Groups of friends - Like student, apres ski is very important for groups of friends of all ages on skiing holidays. Usually these groups have friends who have been going for many years, so are advanced in their field, so challenging slopes are important. Things like off piste, black slopes, snow parks and border cross will all be important to them.

Park Rats - Usually young and hip skiers snowboarders, these people will only care about the rails and jumps in the snowpark. Apres ski is also an important factor for them.

2e. Personas

To help test the usability of the project, I will develop several personas to cover all possible groups of users. This will help me realise if there is anything missing in my review system that would definitely be needed. Below is a structure of the table that will be used to develop my personas.

Name	John Smith
Age	54
Experience	30 years +
Interests	Off Piste, Gourmet Food, Steep Slopes
Travelling with	Wife who is also experienced
Budget	£3000 for week.
Computer Literacy	Confident

Several of these tables will be developed to help replicate different situations of users of the site.

2f. Priorities Assigned to users

Key User - I will be the key user of the site when testing is taking place. This will allow me access to all areas of the site, including the admin panel. Testers will not have access to the admin panel, unless under my supervision.

Secondary User- The secondary users of my project will be the testers at the early stage of the project, and the customers when the project is complete. This will allow me to gather feedback about the system, as the testers will have access to all areas a customer would have.

2g. User participation

As a tester of the website, the users will use real data to complete reviews on the site, and test it for its functionality. The user will need to have knowledge of a ski resort for leaving reviews, so that inaccurate data cannot be entered.

2h. Maintenance Users and Service technicians.

The only maintenance user of the system will be myself.

3. Constraints

3a. Solution Constraints

Description	The Website should work well on both desktop/mobile devices.
Rationale	Users can always enter reviews, whilst on the move. For instance when travelling.
Fit Criterion	The website will use a flexible framework to allow for the site to adjust to the screen its displayed on. HTML5 and CSS3 will be used, with help from jQuery for displaying of data and images.

Description	Users should be able to login securely with the option to use their facebook account.
Rationale	Users can login to leave reviews & save favourites, allowing them to have a better experience.
Fit Criterion	The site will have user accounts, allowing user to login, leave reviews, change details, and add favorites. Users can use their facebook page to login, allowing them to have an easier experience as less data will be filled in.

Description	The user should be able to search via several criteria.
Rationale	The site is easy to use, and users can find the information they want with minimal effort.
Fit Criterion	The search will be customizable, allowing the user to select between 1-10 criteria that are important to them. This will allow the data returned to be relevant to the user.

Description	Top lists of resorts to be included for each criteria.
Rationale	Users can look at the best rated resorts for each criteria. For example if nightlife is the most important option, they can select this and see all resorts ranked by this.
Fit Criterion	The site will have a page, which will allow the user to use buttons to select which top list to view. This will then be pulled from the database and be displayed in order via ranking. The results will display some data about the resort,

Description	To include detailed information about each resort
Rationale	Users can find all information they need, in one central location.
Fit Criterion	The resort pages of the site will include, reviews, key stats, weather & photographs.

Description	Easy to use admin panel
-------------	-------------------------

Rationale	So the site can be easily updated in the future, and reviews can be monitored.
Fit Criterion	The admin panel will allow resort info to be edited, reviews to be changed/deleted, and users to be edited.

3b. Implementation Environment of the Current System.

The system will be viewed in the browser on the worldwide web. The system will work on desktop and mobile devices. In the future, stand alone mobile applications could be made.

3c. Partner or Collaborative Applications.

There is no plan to have any partner or collaborative applications at this stage.

3d. Off the Shelf software.

MySQL- MySQL is a free database package that runs on a web server. Its fast reliable and easy to use and uses standard SQL to program it. This will be used to store any data for my application such as reviews, resorts and user data.

Laravel- Laravel is a PHP framework, allowing common task usually performed with raw PHP to performed easier, and with a lot less code. Several of its features will be used for security, forms etc.

jQuery - jQuery is a small lightweight javascript library, which makes manipulation of a HTML document much simpler. It also has great cross browser support, again simplifying the web development process.

3f. Schedule Constraints

The main schedule constraints of this project are to do with the deadlines of the modules. Below is an outline of the main project deadlines. For all deadlines, the deadline should be completed 3 days in advance to allow for last minute changes and testing.

21/11/14-Initial System Design and Technology Review.

12/12/14 - Initial user experience design.
16/01/15 - Risk Analysis and Functional prototype.
13/02/15 - Refined system design report
06/03/15 - Refined user experience design report
01/05/15 - Completed Solution
01/05/15 - Final Report
11/05/15 - Demonstration
12/05/15 - Viva Voce

The completed solution is for the 1/05/15, but to allow for testing and changes, this would really need to be completed a month in advance to this.

3g. Budget Constraints

The project has a relatively small budget, with the only costs being the domain name and hosting. This will be dependant on the brand, but current thoughts for a domain name are pist.ee which is around £55 per year. Hosting will be around £40 for the project.

4. Naming Conventions and Terminology

4a. Glossary of All Terms, Including Acronyms, Used by Stakeholders Involved in the Project

Term	Meaning
Boarder	A person who goes snowboarding
Skier	A person who goes Skiing
Piste	A Ski slope on a mountain
Off-Piste	Going off the Pistes, through unmarked forests, mountains etc.

Snow-Park	Part of a mountain with artificial jumps, rails etc for freestyle snowboarders and skiers.
Chair Lift	Used to transport skiers up the mountain.
Apres Ski	Nightlife and bars
Black Slope	Advanced slope, Usually ungroomed, steep, narrow with obstacles
Red Slope	Intermediate/Advanced slope, usually steep and narrow.
Blue Slope	Beginner/Intermediate slope, usually very wide and flat.
Green Slope	Beginners Slope, reasonably Flat
Grooming	How the pistes are looked after each night

5.Relevant Facts & Assumptions

5a. Relevant Facts.

- The 3 main competitors for the site would be [World Snowboard Guide On the Snow & Ski Club](#)
- Website will be built with a mySQL database, PHP, jQuery, HTML5 and CSS. Possible frameworks like bootstrap or laravel could be used.

5b Business Rules

Not Applicable until brand guidelines produced.

5c. Assumptions.

- *Each user can only leave one review per resort*
- *Reviews will be reviewed before going live.*

8. The Scope of Work

- To build a snowboard/ski review app which primary function is to provide a function to write/review basic reviews about snowboarding resorts around the world.
- the secondary function will be to include information about ski resorts, to give people all they need to know about the features of a resort in one place.
- The design of the site should be a very important feature, with usability being the number 1 criteria.

9. Functional Requirements

These requirements are what is needed to make the website function correctly. See below an example of my functional requirements snowcards. All 65 snow cards are included in appendix 1.

Number	2
Type	Functional
Description	Login form - Username
Rationale	Allows the user to enter their username.
Fit Criterion	Simple input form with placeholder username. Should work on all devices. Javascript validation so only valid characters can be entered. This then checks username with database, comparing with password
Dependencies	Login form - Submit button (4)

10. Non Functional Requirements

10a Appearance Requirements

Branding - The main logo for my site will be in the header of every page. When images are uploaded, they will also be watermarked with this logo. Brand guidelines will be made with a style tile for developing buttons, forms, lists, etc.

Colours - Colours have not been decided at this stage, but only 3 main colours will be used throughout the site. Initially my thoughts are a cyan/turquoise, grey and white.

10b Style Requirements

Tone of Voice- Within the brand guidelines, a guide will be produced so the same naming requirements are used throughout the site. The design of the site will also be consistent throughout.

Brand Style- The style of the brand will come across as a cool, non corporate brand, which has been built by skiers/boarders to help other skiers/boarders with their trust in the site.

11.Usability and Humanity Requirements

11a Ease of use requirements

One of the main constraints for the project is that it is easy to use. All navigation should clearly look like navigation whether it is a button (Should look like a button) or a link (Blue and underlined).

The site should work well on all screen sizes, with all features easily accessed.

If a form is completed and a user makes a mistake, the form should tell the user, without deleting any other data, so they do not have to retype all of their entry, as this would put them off using a review.

A user should be able to easily change their password.

11b Personalization and Internationalization requirements.

Users will be able to personalize their experience once they have signed up. They will be able to choose a username and password that suits them. Once they have this they will be able to add their own favourite resorts to a page, so they can look back at them later as well as leave reviews.

The native language for my site will be English, as this is the language the primary audience will speak. However, all markup will be valid HTML without slang which most browsers should be able to translate into other languages.

11c Learning Requirements.

The site will be designed in a way that guides the user through the site. A plugin like tour.js could be used to show the user the main features when they first visit the site.

11d Understandability and Politeness Requirements

The site will need to be precise, explaining what the review means, and how it has been compiled. This will mean telling the users how many reviews each resort has received. A possibility will be that reviews will not be counted until there has been at least 3 left, so an average can be taken.

11e Accessibility Requirements

The site should be able to be used by people with disabilities. The site will be fully valid html, which allows the use of screen readers etc. Careful consideration will be taken when choosing colours to allow for people who are colour blind.

12. Performance Requirements

12a Speed and Latency Requirements.

All pages should load immediately with the exception of when a search is performed. These results should still be returned in less than 3 seconds. When a button is clicked it should inform the user that it has been clicked without lag.

12b Safety Critical Requirements

There should be no flashing images on the site which could trigger epilepsy.

12c Precision or Accuracy requirements

- When a customer is leaving a rating it will be between 1-5.
- When a rating is shown to a customer, it will be given as a percentage.
- All information about ski resorts will be accurate such as number of lifts.

12d Reliability and Availability Requirements

All content should always be available, with all necessary maintenance periods happening between 1-6am once the site is live.

12e Robustness or Fault Tolerance Requirements

The system should be able to facilitate several users at once in its early stages.

12f Capacity Requirements

There will not be a limit on the number of reviews, users or resorts the system can have.

12g Scalability or Extensibility Requirements

The site should have the possibility to expand in the future. Features like advertisements, a shop, reviews for hotels could be considered.

12h Longevity Requirements

The product should have a life cycle of five years, but be updated periodically to keep with web standards.

14.Maintainability and Support Requirements

14a. Maintenance Requirements

- The database should be easily moved to another server.
- Users,Reviews and Resorts can be edited by admin.
- Style can be changed via FTP.

14c. Adaptability Requirements

The site is expected to run in all modern desktop and mobile browsers.

Eventually, native mobile apps could be made.

The site should be easily adapted to work on future technologies.

15. Security Requirements

15a Access Requirements.

- Until the project is released, it should only be able to be accessed via people with a valid password and username.
- Once the project is released, the application will be able to be accessed via the general public.
- I will be the only person with access to the admin panel in the near future.

15b Integrity Requirements

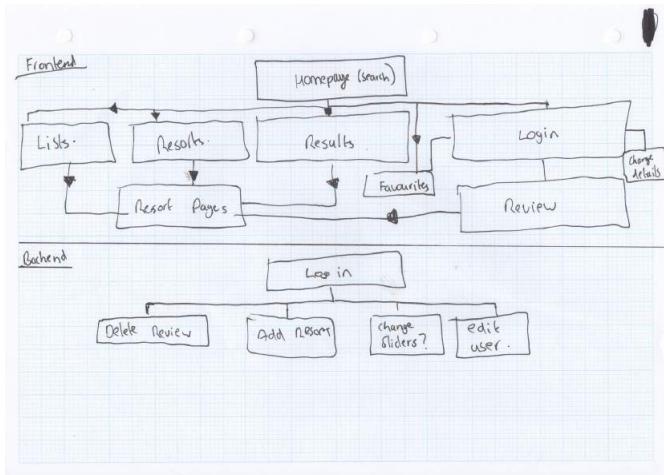
- Validation will stop incorrect data being entered into the forms.
- Captchas will be included on signup to stop robots/spam.
- All Passwords will be encrypted.
- Reviews shall be reviewed before going live, to check for slanderous comments.
Filters could also be included to flag harmful words.

15c Privacy Requirements

- Users will only be able to access their favourites.
- Email addresses of users will not be available to other users.

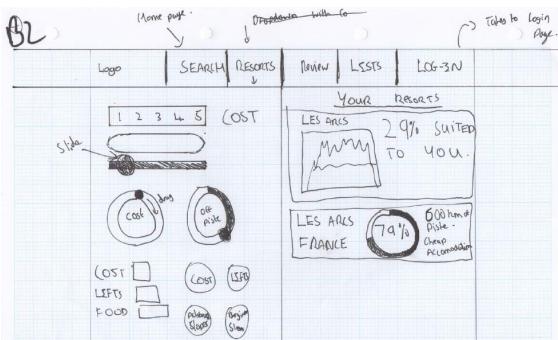
2.3 - Paper Prototyping

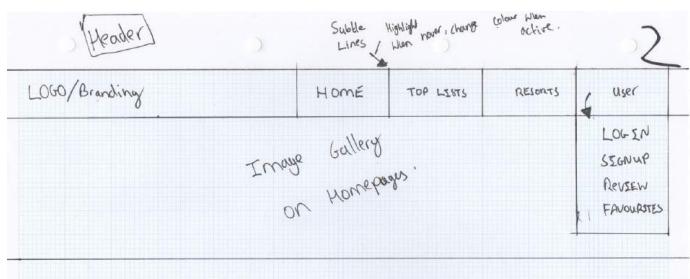
The first process for me was making a simple flow chart, to help decide how many pages the website would need, and how they would link back and forth to each other. This would help when designing navigation for the site.



Focusing on the frontend of the site, all pages end up linking to the individual resort page. This is because this is the key focus of the site, to help users find a resort which is suited to them.

The first step of the prototyping was sketching up some simple ideas, to get an idea of what could work with the project and what would not. These small parts of the site would also give ideas of how the final designs could look, simplifying the design process when making my final wire frames.





BS

③

RESORT NAME: IMAGE

Slider from uploaded images.

Category	Value	Score
Green	29	80
Pink	31	289km
Red	37	2
Blue	4	Food on top

Beginners: Shows stats (average rating from x reviews)

Intermediate: Shows stats (average rating from x reviews)

Advanced: Shows stats (average rating from x reviews)

Pro: Shows stats (average rating from x reviews)

Off-Piste: Shows stats (average rating from x reviews)

Nightlife: Shows stats (average rating from x reviews)

Charities: Shows stats (average rating from x reviews)

Food: Shows stats (average rating from x reviews)

Cost: Shows stats (average rating from x reviews)

Overall Score: 8.5/10

Bottom slope: 7500m

Top slope: 3500m

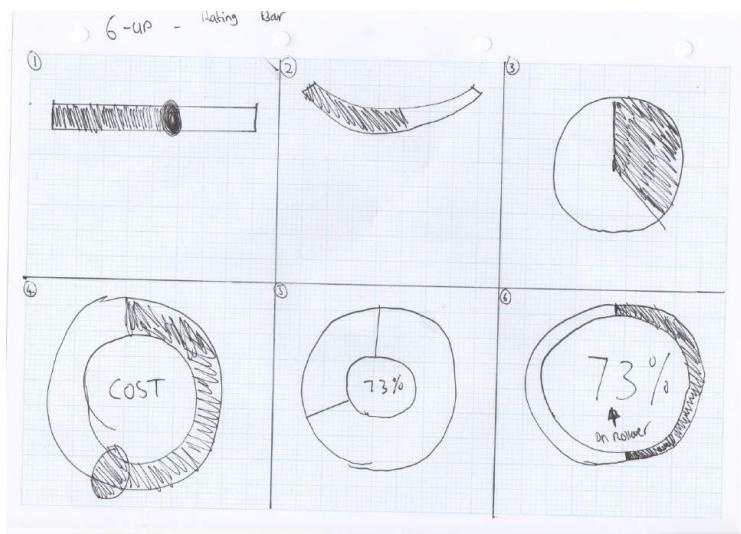
Drop: 7000m

Advanced Shiny nightlife.

Overall Score: 8.5/10

Best RESORT

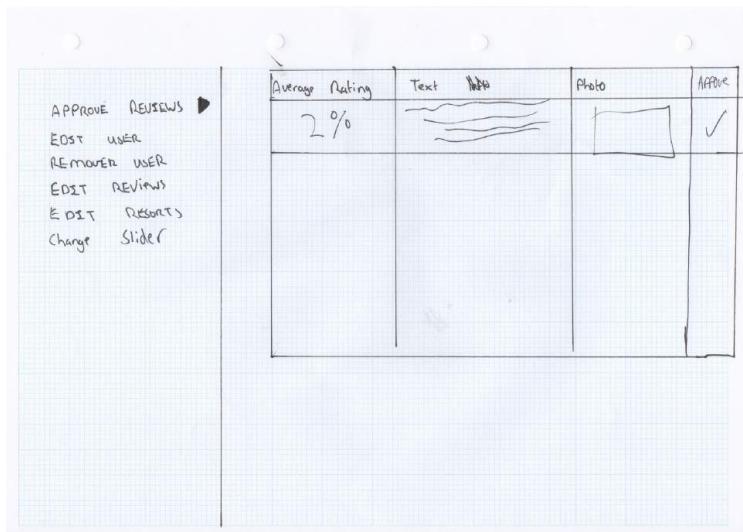
Show more ▶ opens short reviews & weather.



After spending some time designing the other parts of the site (*Appendix 2*) the next step would be to design the final wireframes, which could then be used as reference when making a photoshop mock up of the application.

Above are the designs for the homepage of the site, and the results page generated when a search is carried out.

The backend of the site will not be directly linked from the homepage. This will allow admin (myself for now) to log in and review/delete reviews, edit users and edit tables in the database. It will have a sidebar allowing the user to navigate through the different available options, which will be displayed on the right hand side of the page. These will allow tables to be searched and records edited. For the approve system, this will allow the admin to only allow reviews to be posted once submitted, allowing offensive material to be filtered.



2.4 - Feasibility Testing

Using the Volere Template was also a huge benefit when carrying out my feasibility testing. With the Volere Template, all aspects of the project are outlined. This allows each part of the project to be considered.

The first part of the feasibility testing was to carry out some research, to find out if there was a need for the project, and to discover what people wanted.

To do this I designed a short questionnaire which would highlight what people would like. (Appendix 3) . This allowed me to realise other people's thoughts and not just my own. The results of this confirmed the problem, that there was no comparison site that worked well for winter sports resorts.

When building the site, it is important that the developer will have the technical knowledge able to complete all the tasks. As the designer/developer are both myself, the technical knowledge needed to develop the site will be taken into account whilst designing it. However, challenges are always likely to evolve.

Once the site is built, the developer will need to measure how well the site meets the original requirements, and to see if it meets the final aim of the project.

Other types of feasibility which are not as important to my project are Economic & Legal. As I will not be using anyone else's data without permission, and there are no sensitive parts to the site so legal and ethical issues are not a worry.

2.5 - Methodology Selection

When deciding on a methodology to use for the project, agile seemed to suit the project best. Agile allows software to be developed in rapid stages, with tests happening at each stage of the project to allow quality to be kept to a maximum. In a larger team, agile would lead to a lot of communication between the team. What made agile best suited to this project was that it allows changes to be made at late stages of the project, and testing to begin at an early stage. This means several parts of the site can be worked on at once, and returned to if a problem arises. (*What is Agile model*)

3.0 - Design

3.1 - UX Design Evolution

Since the design of the initial UX design of the project, refinements have been made to how the project will work, which in turn will change the appearance and functionality of the site. In December 2014, a prototype of how the site would work was built, allowing problems with the site to be discovered, and changes to be made. The initial UX report didn't include every single aspect of the site, whereas this report will feature all aspects of the site. To see the full report of designs, refer to Appendix 4.

Another area which hasn't yet been focused on is branding. This was due to uncertainty of how the project would be branded. Now that the initial prototypes have been produced, it is an ideal time to work on branding for the project.

Branding Development

Initially the plan for this project was to develop the brand at an early stage. However, this didn't go to plan, as deciding on a final name for the project proved quite the challenge. The initial plan was to find a domain name which worked as part of the name. Finding domains which worked was quite the challenge. Pist.ee was always a favourite, as it's clean and simple, which suits the site as this is what the site is meant to be. A clean and simple site, which allows user to easily find the best resort suited to them.

The next process of developing the projects brand was producing a logo, and deciding on fonts and colours for the site.

For the branding of the project, it was necessary that the site felt like it was built to help people, and not for commercial reasons. This is one of the main problems with the competitors websites. It's almost too hard to find information, due to the vast amount of advertisements cluttering the pages. I understand if the site was commercialised,

advertisements would probably be necessary, but these would have to be implemented in a way not to ruin the feel of the site. For the image of the site, it would need to feel cool and modern. When designing the logo, inspiration was taken from a selection of skate, lifestyle & snowboard brands. Since the beginning of the project, I had an idea that a bumper sticker style logo would really suit the project, which led me to research different designs companies were using for them.



Supreme - <http://www.supremenewyork.com/>



I Love Bass - <http://shop.ilovebass.co.uk/product/rectangle-sticker>



Diamond -

<http://www.routeone.co.uk/diamond-supply-co-dmnd-sticker.html>



Grizzly -

<http://www.routeone.co.uk/grizzly-stamp-logo-sticker-6.html>

Development Stage



Instead of designing the logo against a white background, an image was used as this gives a better representation of what it will look like when included in the site. This first logo design made the decision that the Contrail One font would no longer be used for

the site. The square box didn't suit the style of the site, so the next designs will be made using a rectangular bounding box.

This logo is made using the Futura font face. The character spacing was adjusted so that the dot wouldn't affect the gap between the T and E, making it uniform with the rest of the characters. Between the other characters the character spacing was tweaked slightly so that all the letters are the same distance apart. Ideas to improve on this design are to possibly change the colour of the dot, and remove some padding around the edge.

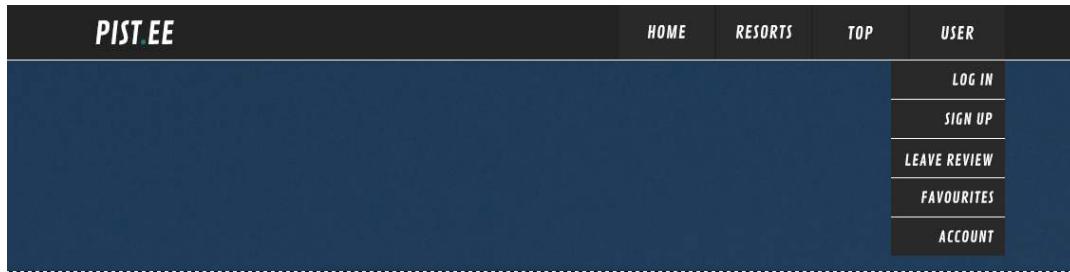


Above are some more concepts of the bumper sticker style logo. Experimenting with different colours has made re-considering the primary colours of the site a viable option, as the green and red backgrounds both look very well. However, for a snowboarding/ski sign the turquoise is best suited. This leaves the decision whether to go for white or light grey text, as the coloured dot doesn't work.

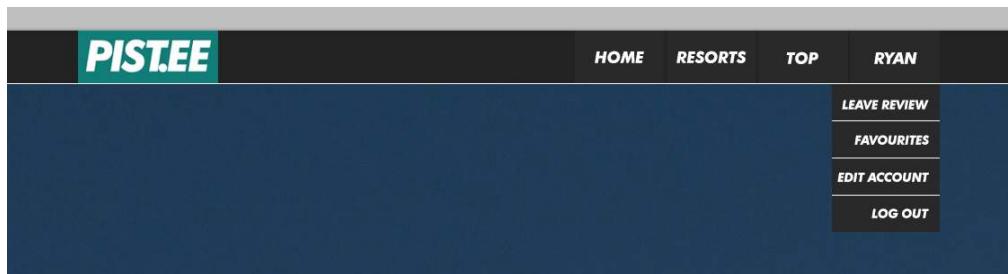
Now that a logo had been developed, this would allow me to start the refinement of my initial designs, with new branding and using both the feedback received from lecturers and testers, together with changing parts of the design to meet my new functional requirements, which had been developed when producing my refined system design.

Refinement of header

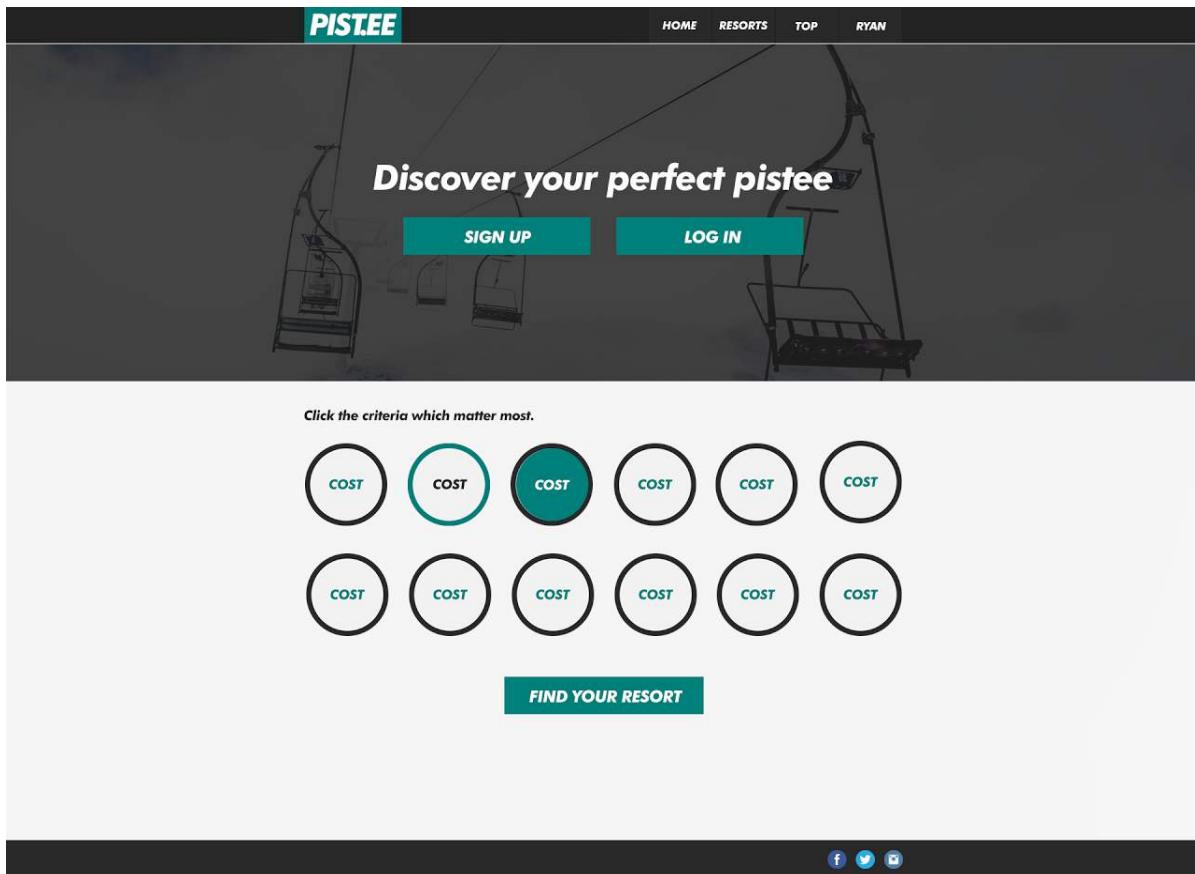
As the header is one of the most important parts of the site, as it appears on every page, this would be a good place to start with the refinement. Due to the refined functional requirements, there would be several changes to the drop down menu of the page. As the font face & logo are changing this would also have a major impact on the header. Below is the initial header, followed by the new header to give a comparison.



This is how the two versions of the header will look. One for when a user is logged in and one for when they are not logged in.



Home Page

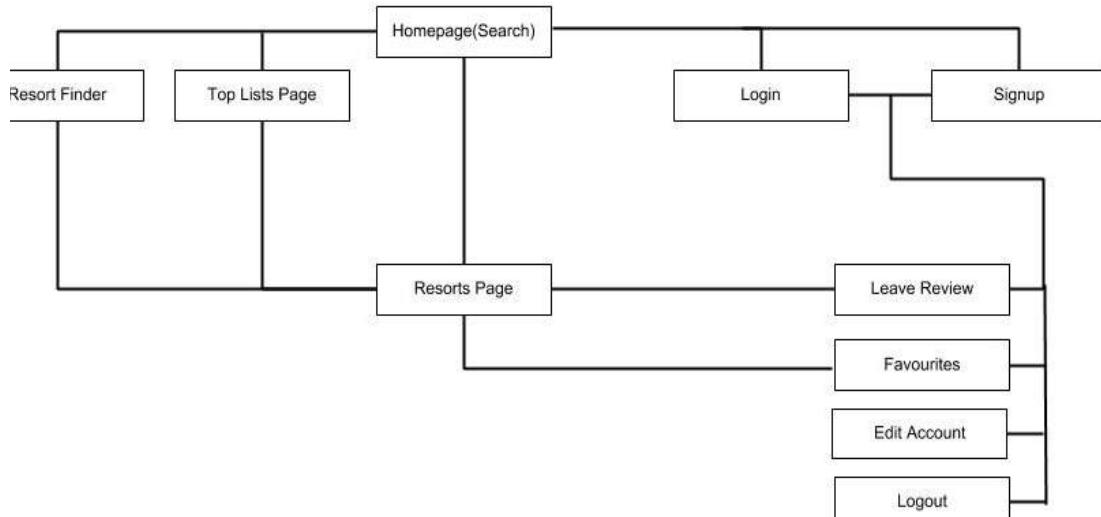


The main criteria of the homepage of the site hasn't changed. This has meant only minor updates have occurred with the homepage, such as updating to the new font and logo. Another change was in the original designs I'd planned on using 10 criteria, however with the site being responsive this proved a problem, as 10 can only be split evenly into two rows, whereas 12 can be split into 2, 3 or 4 rows. The footer of the site was also changed to a thinner design, as a large footer was not needed. To see the rest of the designs for my site, including mobile please refer to appendix 4.

3.2 - System Design

The structure of the front end of the system is relatively simple, with all pages linking back to the final resorts page, as the main purpose of the site is to find for the user their

ideal resort. The pages of the site give the user several options to find the resort suited to them, or to find a resort they have heard about by a simple country search.



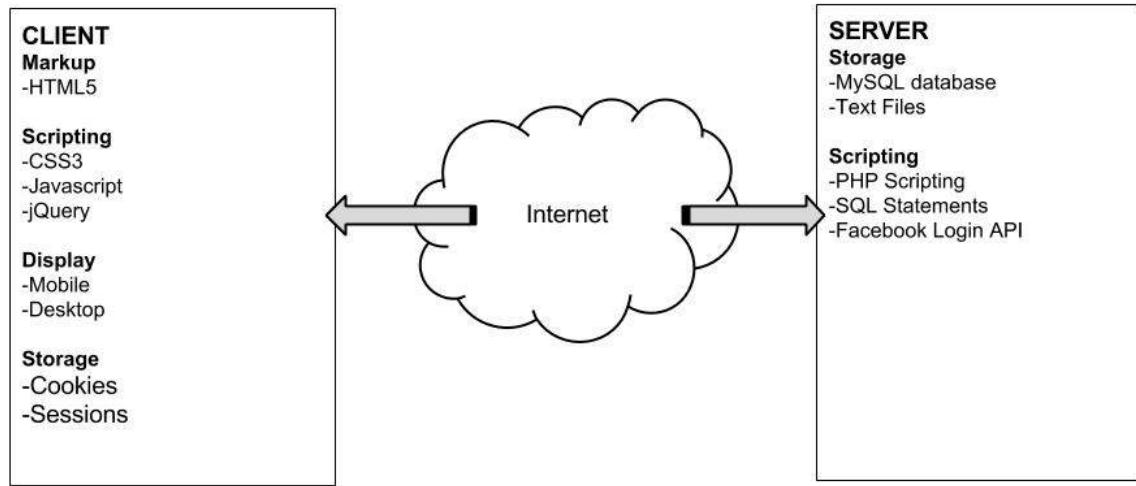
The most complex page of the site will be the main homepage, which uses a mixture of averages to search the database.

When the user logs into the site, they have 4 options which are:

- 1) to leave a review, 2) view their favourites, 3) edit their login details. 4)Logout

Client server model

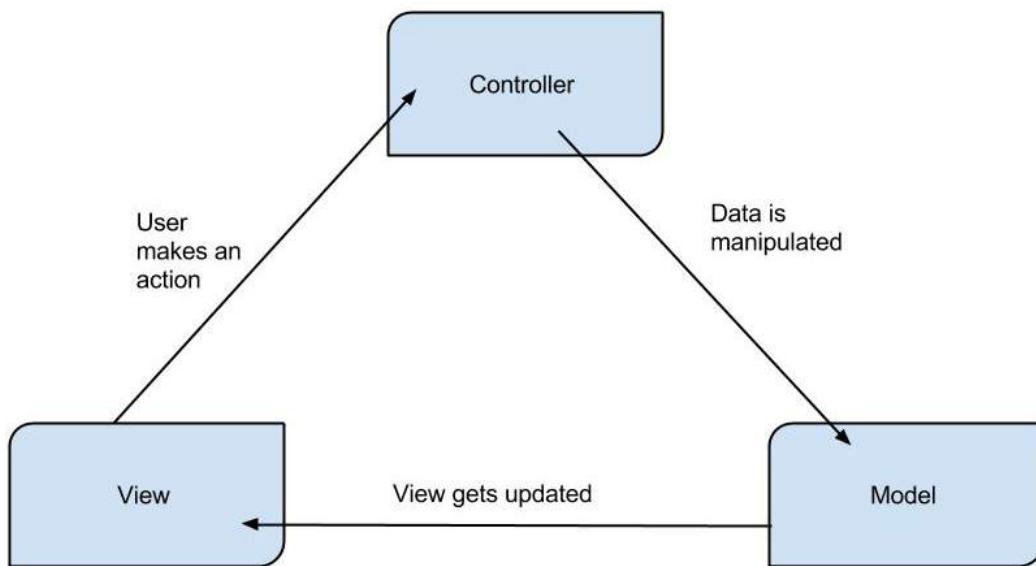
The client server model divides the application into two main parts. The Client Side and the Server Side. As the names suggest the client side is all the processes that occur on the end users (Client) machine. The server side is all the process which happen on the sites server, as well as other servers if a plugin or api is being used. Below is my client server model.



On the client side we have the markup, scripting, display and a small amount of storage. The main markup for this application will be HTML5. Without any scripting this would be rendered as plain text. The 2 scripting languages which will be used are CSS3 and Javascript. CSS3 will be used to render the main style and layout of the site, while jQuery, a Javascript framework will be used to manipulate the dom helping with tasks such as form validation, animation and loading data dynamically. Cookies and Sessions will be used to store user login details, so the user doesn't have to login every time they visit a new page.

On the server we have our main storage system which will be a MySQL database. This will store all of the data on the site in tables. This allows vast amounts of data to be stored, and easily searched. Text files could also be used to store information, but this isn't very secure and probably won't be used. The main scripting language will be PHP, allowing me to reuse files such as headers across the site, which will save time when editing the site. It will also be used along with SQL statements to search/query the database. The Facebook login API will be used to access the a Facebook server, to get user details.

3.3 - Logic Design



Above is a model view controller of how the application will function. The view is what is displayed in the browser. When a user makes an action like a search, or fills in a form, this is sent to the database (Controller). Then this data is manipulated via the PHP Scripting before loading back in the view so the user can see the result.

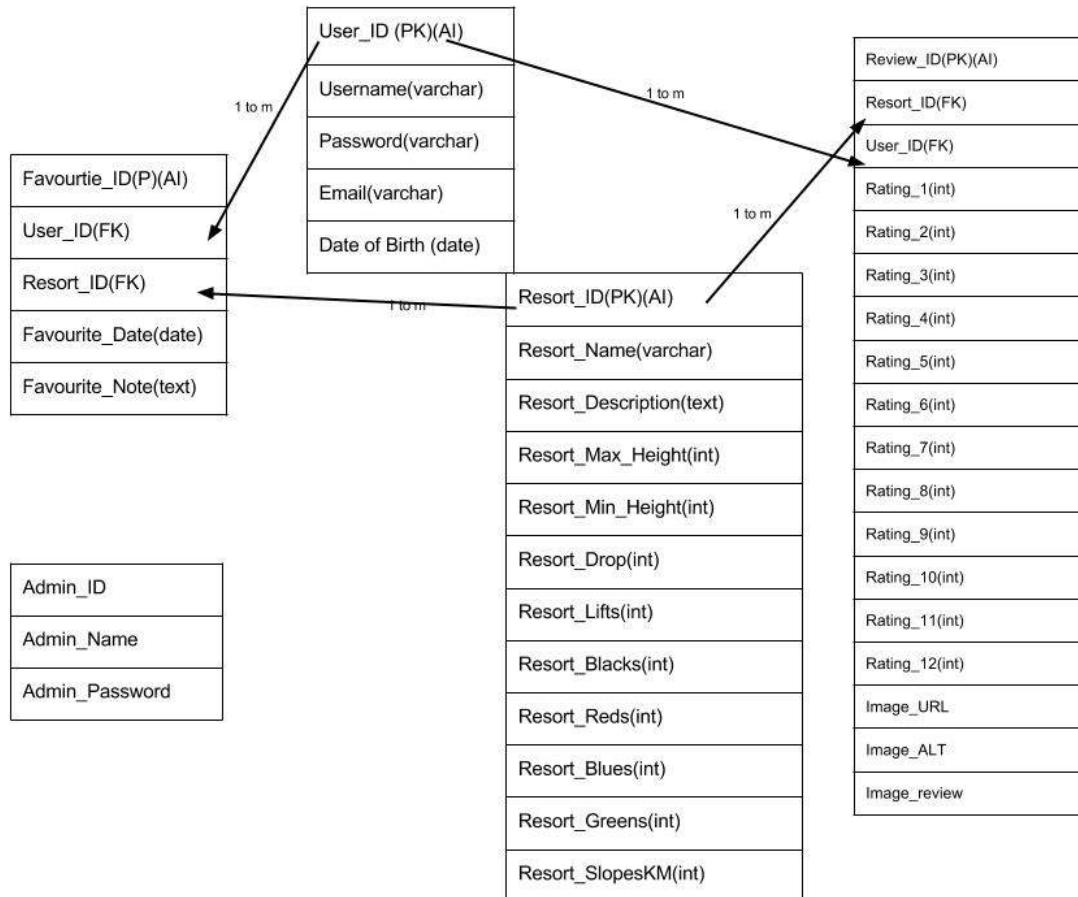
Several design patterns will be used throughout the application. This will save processing time, and help keep all the designs consistent throughout the site. Design patterns will be implemented via PHP, HTML and CSS.

The header and footer files for the site will be an include file. This means each page will call this file, and display it as part of the page. The benefits of this are, if something needs to be edited, it only needs to be edited on one page, instead of every part of the files. Include files could also be used for displaying results, keeping them consistent across the site.

Another design pattern will be the buttons and forms of my site. These will all use the same CSS, stored in an external file, meaning they will all have the same style throughout the site.

3.4 - Data Design

Below is the initial concept diagram of how the database system will look for the application. It consists of 5 tables, 4 of which are linked via primary keys, and one which is used to store the admins details, so it doesn't need to be linked into any of the other tables.



There is a table to store the user's details which has a 1 to many link to both the reviews and favourites tables. This is because one user can have many favourites and leave

many reviews, but each favourite and each review can only have 1 user. The resorts table also has the same links that the user table does to the review and favourites table. This allows the user to add a resort to their favourites, and to their reviews. These again are 1 to many relationships as each review can only have 1 resort, and each favourite will also be one resort.

The favourites table includes its primary key, and the two foreign keys to the user/resorts table, so it can decipher which user has left the review, and which resort it is. It also has a date which the item was made a favourite and a space for the user to leave a short note of why they made it a favourite.

The resorts table includes all the necessary data about each resort. This may also include the average rating for each resort in the future.

The reviews table includes its primary key, a foreign key to match each review to a specific resort, the 12 review values, a link to the image, an alt tag for the image and the user's review.

The database can then be used for the user logins and to control the rating/review system.

4.0 - Implementation

4.1 - Technology Selection

HTML5/CSS3

When starting the main choice was whether to make a native smartphone app or a desktop application with responsive design so it could be used on mobile devices.

The decision made was to use HTML5 & CSS3 to build a responsive web application. This would allow the application to be used on all modern devices. For building for the web, there isn't any viable alternatives to HTML5&CSS3.

In the future, dedicated mobile apps could be developed for Android & iOS but at this stage of the project it is not a necessity.

Mysql database

When selecting my database, there were two obvious choices. MySQL or MongoDB. Both are free open source databases, with MySQL being a relational database, and MongoDB being non relational. But what is the difference?

MySQL is a relational database and uses SQL statements. This makes it easy to query and allows a lot of data to be stored in each column. One problem with this is that they have to use a single server, unlike MongoDB.

Non Relational Databases can run across several servers which solves this problem. as the data field and value are stored together, unlike a standard relational database where everything is stored in columns of a table.

For this project I've decided to use MySQL, as I have a lot of previous experience with these databases. MongoDB also does not offer an easy way to join tables.

Facebook Login API

Users will be given the option to login with Facebook when signing up for my site, making the site more usable, saving the end user time. This will allow the user to skip authenticating that their email is correct, and entering their details into the userform. This will be great for the younger users of the site, as the majority of under 30's have a Facebook account.

PHP 5

I have decided to use PHP for my server side scripting language, as it suits the project, and works well with SQL statements for querying MySQL databases. It will also allow

the possibility to use a framework called Laravel, to simplify things like forms and security. It will allow the simplification of a lot of my code, and allow me to reuse several code segments.

JS/jQuery

Javascript and jQuery will perform the majority of the client side scripting. These will allow me make the site more dynamic, with features like animations and image galleries. It will also be useful to validate forms before they are sent to the server, to reduce the load on the server, and reduce the waiting time for users.

Frontend Framework

One of the decisions that needs to be made is if it is best to use a frontend framework of the site. Examples of these are Bootstrap, Kickstrap and foundation. What do they do? In short, they make building the front end of your website a quick and easy process. Frameworks are made up of a combination of HTML, CSS and JS files and images such as icons. There are several advantages and disadvantages of using a frontend framework.

The main advantages are that they are quick and easy to use, and are very flexible when it comes to cross browser compatibility. It also saves a lot of thinking time, for instance, when designing a button for your site, it may take a few hours to get the design correct, but with a framework these features are already included, and can be customised if needed.

However, frameworks also have many disadvantages. When customising your site, several problems can occur. One of these is that you are usually familiar with your own code, so finding what needs to be customised can sometimes be a long and painful job. When customising the code, the standard practice is to call a second stylesheet, which will overwrite the original code.

Another problem with frameworks is updating. When updating to a newer version, this could change the look and feel of your site, and possibly break it. The obvious idea would be not to update to the newer versions, but then your site would fall behind current standards.

For this project, a frontend framework will not be used. This is because a fully custom design will be used, and for this size of site it is not necessary.

4.2 - Technology Used

As predicted, the front end of the application was built using HTML5 & CSS3, with the help of Javascript & some jQuery plugins. These technologies allowed the application to control how each of the pages of the site was displayed. For the Database MySQL was used which allowed all the data needed for the application to be stored easily in tables and be accessed using PHP & SQL.

PHP allowed include files, which means parts of the site which would be reused throughout all the pages could be in one file. The 4 main include files used were the file which would get the database connection, header, footer and the template file to generate my results

```
<?php  
//database configuration  
$config['mysql_host'] = "localhost";  
$config['mysql_user'] = "B00579628";  
$config['mysql_pass'] = "xN46bQBn";  
$config['db_name'] = "B00579628";  
  
//connect to host  
$db = mysql_connect($config['mysql_host'],$config['mysql_user'],$config['mysql_pass']);  
//select database  
@mysql_select_db($config['db_name']) or die( "Unable to select database");  
?>
```

The above screenshot shows the database connection file. Every single page in the site needed to connect to the database, so having it as a separate file meant that if any info needed updated, it would need to be done once instead of several times. It also means

once a browser loads it, it can cache the file saving that code being reloaded every time a new page is loaded.

```
<?php
echo '
<section rel="'.$percentage.'" class="result-section">
<a class="sections" href="resort.php?page_ID='.$row['Resort_ID'].'">
    <div class="result-header">
        <h1>'.$row['Resort_Name'].'
        <h2>'.$row['Resort_Country'].'
    </div>
    <ul>
        <li class="head"> Slopes</li>
        <li> Blacks: '.$row['Resort_Blacks'].'
        <li> Reds: '.$row['Resort_Reds'].'
        <li> Blues: '.$row['Resort_Blues'].'
        <li> Greens: '.$row['Resort_Greens'].'
        <li> Lifts: '.$row['Resort_Lifts'].'
        <li> Liftpass Cost:&pound; '.$row['Liftpass_Price'].'
    </ul>
    <div class="result-avg">
        <h3> '.(int)$percentage.'%
    </div>
    <ul>
        <li class="head">Size</li>
        <li> Highest Slope: '.$row['Resort_Max_Height']. 'M</li>
        <li> Resort Drop: '.$row['Resort_Drop']. 'M</li>
        <li> Lowest Slope: '.$row['Resort_Min_Height']. 'M</li>
        <li> Resort Size: '.$row['Resort_Size']. 'KM</li>
    </ul>
</a>
</section>';
?>
```

The above PHP file is used in both the resorts and top lists page. It is called every time a while loop runs. This loop runs each time the SQL query returns a row in the db. This file produces the HTML to format the results.



Using SQL allows a variety of queries to be run on the database. These queries are necessary to return data. Some of the queries I used where relatively simple like the one below. It simply selects all the results from the resorts table.

```
" SELECT * FROM resorts;"
```

However more complex queries are needed when selecting data from a number of tables.

The below query is one of the more complex queries in the application. The query is used on the favourites page and selects from 3 tables! The first query puts a join between the resorts and favourites page. It then uses a where statement to get all the data from the favourites table where the user id is equal to the user id in the session variable. The join allows the information about the resort in each favourite to be displayed.

Once this query runs, it outputs the results using a while loop. Inside this loop, another SQL query takes place, which gets the average overall rating for each resort. This is then converted from a mark out of 60 (Total of the 12 rating options worth 5 each) into a percentage which is then displayed. A piece of javascript is then used to sort the sections into an order determined by their total overall rating.

```
$query = " SELECT favourites.*, resorts.*  
        FROM `favourites`  
        JOIN `resorts`  
        ON favourites.Resort_ID=resorts.Resort_ID  
        WHERE favourites.User_ID=$userid  
        ;";  
  
        $result = mysql_query($query) or die(mysql_error()); //note: use mysql_error() for development only  
        while($row = mysql_fetch_assoc($result)) {  
            $resortID = $row['Resort_ID'];  
            echo '<a class="sections" href="resort.php?page_ID='.$row['Resort_ID'].'" ><div class="result-section">  
                <div class="result-header">  
                    <h1>' . $row['Resort_Name'] . '</h1>  
                    <h2>' . $row['Resort_Country'] . '</h2>  
                </div>  
                <ul>  
                    <li class="head"> Slopes</li>  
                    <li> Blacks:' . $row['Resort_Blacks'] . '</li>  
                    <li> Reds:' . $row['Resort_Reds'] . '</li>  
                    <li> Blues:' . $row['Resort_Blues'] . '</li>  
                    <li> Greens:' . $row['Resort_Greens'] . '</li>  
                    <li> Lifts:' . $row['Resort_Lifts'] . '</li>  
                    <li> Liftpass Cost:&pound;' . $row['Liftpass_Price'] . '</li>  
                </ul>';  
            $averagedata = "SELECT AVG(Overall_Rating), Resort_ID, status FROM reviews WHERE Resort_ID=$resortID AND status=2";  
  
            $avgresult = @mysql_query ($averagedata); // Run the query.  
            $avgrow = mysql_fetch_array($avgresult);  
            $percentageoverall = ($avgrow[0] / 60) * 100;  
  
            echo '<div class="result-avg"><h3> ' . (int)$percentageoverall . '%</h3></div>';  
            echo ' <ul>  
                <li class="head">Size</li>  
                <li> Highest Slope:' . $row['Resort_Max_Height'] . 'M</li>  
                <li> Resort Drop:' . $row['Resort_Drop'] . 'M</li>  
                <li> Lowest Slope:' . $row['Resort_Min_Height'] . 'M</li>  
                <li> Resort Size:' . $row['Resort_Size'] . 'KM</li>  
            </ul>  
            <a href="delete-favourite.php?fav_ID=' . $row['Favourite_ID'] . '" class="delete-fav"></a>  
        </div></a>';  
    }  
>
```

jQuery & JS allowed the website be more dynamic and interactive for the user. The code below is used to search the table on the resort finder page. Every time a key is released after it is pressed in the search bar the function is ran to search through the table. Originally, the code would not work properly due to capital letters, but converting all text searched to lower case resolved this problem.

```
setInterval('swapImages()', 5000);

$("#search").keyup(function () {
  var value = this.value.toLowerCase().trim();

  $("table tr").each(function (index) {
    if (!index) return;
    $(this).find("td").each(function () {
      var id = $(this).text().toLowerCase().trim();
      var not_found = (id.indexOf(value) == -1);
      $(this).closest('tr').toggle(!not_found);
      return not_found;
    });
  });
});
```

Search						
Resort Name	Country	Lifts	Slopes	Size	Max Height	Favourite
Pas De La Casa	Andorra	129	97	216 KM	3200M	★
St Anton	Austria	86	134	260 KM	2811M	★
Bansko	Bulgaria	22	17	75 KM	2600M	★
La Plagne	France	137	242	425 KM	3250m	★
Les Arcs	France	120	130	425 KM	3145M	★

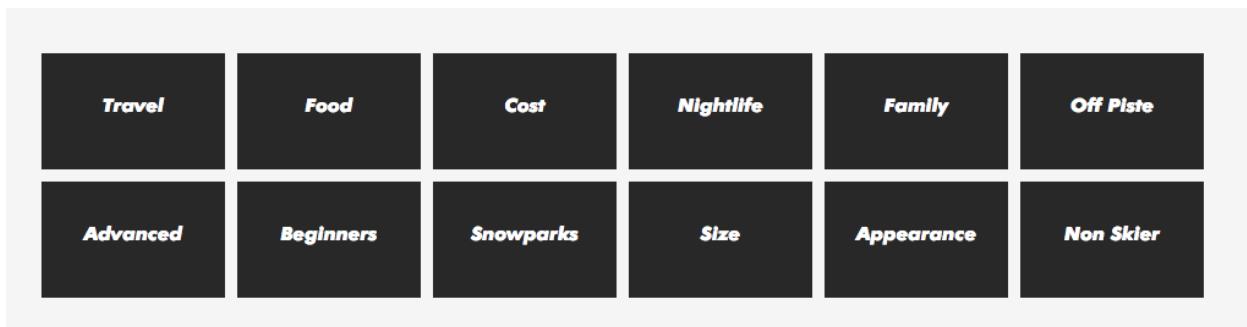
la p						
Resort Name	Country	Lifts	Slopes	Size	Max Height	Favourite
La Plagne	France	137	242	425 KM	3250m	★

4.3 - Notable Achievements

Throughout the project, many difficult challenges arose, as this was by far the most demanding project the developer had worked on.

One of the hardest parts of the project was building the search for the homepage of the site. This search would allow users to select between 0-12 options of what mattered to them when they went skiing/snowboarding. The results returned would then be an average of what they had picked and be returned in the correct order. At first this was an extremely daunting challenge, but once what needed to be done was on paper, it was just a matter of getting everything to work together.

The first step was building the front end. After some research using checkboxes in a form and editing the style would be the best option for the front end. Originally the plan was to have the buttons as circles, but with the varying lengths of type, this just didn't work. Therefore the design of the buttons were changed to rectangles.



To get the checkboxes edited to look like this, they were paired with labels and then the checkboxes themselves set not to display. This allowed custom styling to be applied.

```
input.CB{  
    display:none;  
}  
  
label {  
    display: block;  
    float:left;  
    width:150px;  
    height:60px;  
    margin-right: 10px;  
    margin-bottom: 10px;  
    background-color: #292929;  
    color:#fff;  
    padding-top: 35px;  
    color: #fff;  
    text-decoration: none;  
    font-size: 16px;  
    cursor: pointer;  
    font-family:'futura_md_btbold_italic';  
}  
  
input.CB:checked ~ label {  
    background-color: #0015fb;  
}
```

```
<input type='hidden' value='0' name='1'>  
<span><input class="CB" type="checkbox" name="1" value="1" id="travel"/> <label for="travel">Travel</label> </span>
```

Now that the front end of the checkboxes complete it was time to make them function like they should. To do this PHP & SQL . The input form method was set to GET so that the values of the form would be posted in the URL to the results page. The

next problem occurred as if a checkbox isn't selected the value isn't posted in the form. To overcome this, a hidden input was included before each checkbox with the same name with the value set to 0. This would then post the value 0 for that name, or if a checkbox was selected it would send a 1. Now that this information was posted to the results page using the get method it was time to start the complex script which would calculate the average selected categories. First `$_GET` is used to get the values needed from the form, and assigns them to variables.

```
$rating1 = $_GET["1"];
$rating2 = $_GET["2"];
$rating3 = $_GET["3"];
$rating4 = $_GET["4"];
$rating5 = $_GET["5"];
$rating6 = $_GET["6"];
$rating7 = $_GET["7"];
$rating8 = $_GET["8"];
$rating9 = $_GET["9"];
$rating10 = $_GET["10"];
$rating11 = $_GET["11"];
$rating12 = $_GET["12"];
```

The next section of code assigns some variables to 0, which will be used to store ratings, and a count of how many checkboxes have been selected. Then the database needs to be queried which works similarly to how the favourites page works. Getting the correct data is a little more complex however. The first select query simply selects all the rows from the resorts table and then starts a while loop. Now the average rating for each resort needs

to be selected. First we run a query to get all the averages from review table, where the resort ID = the current resort ID of the loop which is running.

```
$averagedata = "SELECT
AVG(Overall_Rating),AVG(Rating_1),AVG(Rating_2) ,AVG(Rating_3) ,AVG(Rating_4) ,AVG(Rating_5) ,AVG(Rating_6) ,AVG(Rating_7) ,AVG(Rating_8) ,AVG(Rating_9) ,AVG(Rating_10) ,AVG(Rating_11) ,AVG(Rating_12) FROM reviews WHERE Resort_ID=$resortID AND status=2;";
$savresult = @mysql_query ($averagedata); // Run the query.
$avgrow = mysql_fetch_array($savresult);
```

Then we need to calculate the average of the categories which have been selected. To do this, 12 if statements are used. Each variable `$rating1/2/3` etc will have a value of either 1 if the checkbox has been selected, or 0 if not. If the rating is 1 the script inside the if statement is ran. This selects the average for that row and turns it into a percentage and it assigns it to the rated variable. The counting variable adds one to the

variable each time an if statement is ran, getting the number of checkboxes selected.

```
if($rating1 ==1){
    $rated1 = ($avgrow[1]/ 5)* 100;
    $counting[0] = 1;
}
if($rating2 ==1){
    $rated2 = ($avgrow[2]/ 5)* 100;
    $counting[1] = 1;
}
if($rating3 ==1){
    $rated3 = ($avgrow[3]/ 5)* 100;
    $counting[2] = 1;
},
```

Once this section has ran, another if statement calculates the count on the count variable followed by the overall percentage of the ratings. If no checkboxes are selected an else statement runs which calculates the average overall rating of the 12 results. The data is then output to html using the template file attached.

```
if($count!=0){
    $count =count($counting);

    $percentage = ($rated1 + $rated2 + $rated3 + $rated4 + $rated5 + $rated6 + $rated7 + $rated8 + $rated9
+ $rated10 + $rated11 + $rated12)/$count;

}
else{
    $percentage = ($avgrow[0]/60)* 100;
}
include("include/result-template.php");
}
```

Another page which was difficult to build in the project was the top list page. The functionality of this page was similar to the main homepage, but would allow only one selection to be shown at a time. To build this page, the main links between each page linked back to the top-resorts.php page, with a different page id. This would allow the SQL SELECT to select the correct way to organise the resorts in the page dependent on the page ID.

```
<ul class="categorypage">
<li><a href="top-resorts.php?page_ID=1">Travel</a></li>
<li><a href="top-resorts.php?page_ID=2">Food </a></li>
<li><a href="top-resorts.php?page_ID=3">Cost </a></li>
    ...
```

If the page_ID is equal to 0 (This happens when the original top list button in the header is clicked), the overall percentage which is used in the DB is used to calculate the overall top results. If not, the correct average for the selected row got from the page ID is used.

```
if ($page_ID == 0){
    $percentage = ($avgrow[0] / 60)* 100;
}
else{
    $percentage = ($avgrow[$page_ID] / 5)* 100;
}
```

This ended up working extremely well due the page not reloading each time the links where clicked.

For the review system, one of the functional requirements was that the reviews would need to be approved before uploading. This was quite simple to implement, but worked very well. In the reviews column, an extra column called status was added which had a default value of 0.

<input type="checkbox"/>	17	Review_Name	varchar(64)	latin1_swedish_ci	No	None	Change Drop Primary U
<input type="checkbox"/>	18	Review	text	latin1_swedish_ci	No	None	Change Drop Primary U
<input type="checkbox"/>	19	status	int(11)		No	0	Change Drop Primary U

Now it was time to start work on the admin panel of the application. For the reviews section, a table was echoed using a while loop echoing all the results from the database. The table could also be searched to find a specific entry.

Edit Reviews						
<input type="text" value="Search"/>						
Review ID	Review Name	Review Description	Image Link	AVG Rating	Delete	Approve
31	Test Review	hello hello hello		66.66666666666667	Delete	Approve
30	Great for a lads holiday	Great skiing and even better nightlife. May as well be called Mandorla mind you!		86.66666666666667	Delete	Approved
29	Not much to do at night	Great skiing and weather but not much to do at night!		60	Delete	Approved
28	sd	Leave a short review.....		60	Delete	Approved
27	sd	Leave a short review.....		60	Delete	Approved

The table allowed individual records to be deleted or approved, as well as including a preview of the image uploaded, the AVG Rating given, the description and name. This will allow the reviewer to select if the review is genuine. If not the review can be deleted, and if it is it can be approved. This works by changing the 0 in the status column to a 1.

2. For all the select statements involving the reviews in the main site, a WHERE clause can now be added to only show when the status=2. This means unapproved reviews won't show.

Another challenge which rose while implemented the application is to do with the favourites system. Favourites are stored in their own table in the database. The problem was duplicate favourites could be added into the database from the favourite buttons throughout the site.

Max Height	Favourite
3200M	
2811M	
2600M	

There were a few ways which this could have been overcome. The method chosen was to hide or edit the buttons when the favourite already existed, stopping them from being able to be added. The first task was to see if a user was logged in using sessions. The if statement queries if a user is logged in or not. If a user is not logged in the user id gets set to 0.

```

if(empty($_SESSION['user'])){
    $userId=0;
}
else{
    $userId = htmlentities($_SESSION['user']['user_ID'], ENT_QUOTES, 'UTF-8');
}

```

The next step is to run a query which will select all the favourites from the favourites table for the current user logged in, and where the resort id equals the resort id of the selected row/resort. This is the first line of the below query.

```
<?php  
    $favquery = mysql_query("SELECT * FROM favourites WHERE User_ID=$userid AND Resort_ID=$resortID");  
    if(mysql_num_rows($favquery) == 0){  
        echo'<a class="fav" href="add-favourite.php?page_ID=' . $resortID . '></a>';  
    }  
?>
```

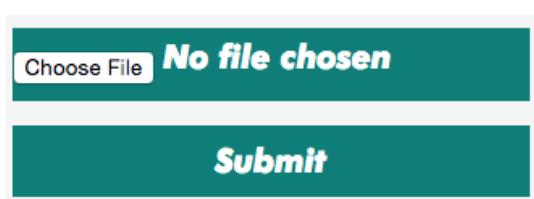
The query will return one result, which is added to the variable \$favquery. If no results exist the \$favquery will be 0. The if statement then echos the link to add the selected resort as a favourite if it doesn't exist.

4.4 - Notable Challenges

As well as having noticeable achievements in the project, there were also some challenges which were not resolved in the time frame available.

One of the requirements for the site was to use the Facebook login api. The plan was to implement this at a late stage of the project, but unfortunately this wasn't completed. This will hopefully be added in the near future, allowing more information to be gathered about the users. Luckily it isn't a necessity to have in the site and it still functions correctly without it.

Another challenge for my project which wasn't completed was the styling of the image upload button. Like the facebook login, this isn't a necessity, but if it was styled correctly the user experience would be improved. To overcome this project, the button would be



hidden, and then a new button added with JS linked to the old hidden button. This means a complete custom style or even an image could be used.

Testing

5.1 - Testing Approach Solution

Now that the application was fully implemented it was time to test it, to make sure everything worked correctly and as expected. The main tests that would be performed on the application would be to see if the functional requirements had been met correctly. This would decipher whether the application had met its requirements, aims and objectives. For these tests, edited versions of the snow cards from the functional requirements would be used. An original snowcard is shown below.

Number	12
Type	Functional
Description	Primary Menu - Top lists Button
Rationale	A button to link to the top lists page.
Fit Criterion	The button should look like a button, with the text easy to read on all devices.
Dependencies	Top lists page.

These snowcards decipher how every function on the site should work, so every feature should be included. Below is an edited snowcard which will be used to test my website.

Number	12
Type	Functional
Description	Primary Menu - Top lists Button
Rationale	A button to link to the top lists page.
Fit Criterion	The button should look like a button, with the text easy to read on all devices.
Expected Result	The button links the user through to the top list page
Actual Result	Pass/Fail

Once the test has been ran, the result can be recorded. If the result is a fail, the problem can be fixed, and then retested.

After the functional testing is completed, the next stage of the testing process is usability testing. Usability testing is when you get real users to test your problems, and see how they feel about using the system. To successfully user test, 5 different individuals will be used of different ages and digital abilities. I will get them to use the site, and fill in a short review about how they felt about the site.

Another type of testing that will need to be completed is compatibility testing. Compatibility testing will test if the site works across different browsers and devices. As the application needs to work across all devices, testing will occur on a desktop, laptop, tablet and smartphone

5.2 - Testing Process

The testing process for this project will be relatively simple. For the functional requirements the first step will be to edit the snow cards to include the tests. A test environment will then be made. The main test environment for my website will be using google chrome on an imac with a 1280px wide browser window. With the test environment set up, tests will run in the order of the functional requirements to simplify the process.

User testing will take place under the developer's solution, so instant feedback can be given as well as the questionnaire which will be completed by the user. The user will use the site for 5 minutes, and then fill out the questionnaire immediately. To see the example user form please refer to example 6. The test environment will be the same as the environment for the functionality testing.

The final tests will be compatibility tests. The goals for these tests are that the site works well on each site, and relates to the original designs. All functions should still be accessible. For each device, a short report will be made about how the application works as a whole on that device.

5.3 - Test Results

5.3.1 - Functional Requirement Test Results

The first tests which were ran were the functional requirement tests. This included 66 tests on different functions of the site. Out of the 66 tests 56 passed and 10 failed. This was a great result considering the majority of the tests which failed were due to changes in the design of the site, and did not affect the functionality. However some of these failures were important, and where things which will need to be included in the site in the future to improve usability. These were things like the requirement below which simply hadn't been built yet.

Number	9
Type	Functional
Description	Sign up with facebook
Rationale	Allows the user to use their facebook details to sign up, to save time.
Fit Criterion	User can connect with their facebook account to use as a signup method, to help save time.
Dependencies	Submit form- Submit button (10)
Expected Result	Should fetch details from users facebook.
Actual Result	Option does not exist
Pass or Fail	Fail

To refer to my Functional testing results please see Appendix 5.

5.3.2 - User Testing Results

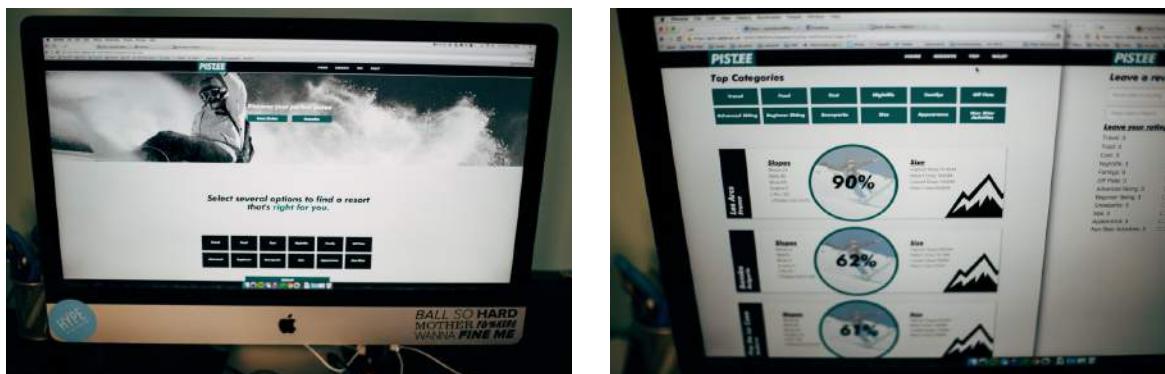
For my user test results to give an accurate report of how that sites worked, I needed to find a selection of candidates to test my site from different backgrounds. I tested a total of 5 people, of different age groups, computer knowledge and career. This give me a view of the site from other eyes apart from my own. The users accessed the site with no prior knowledge apart from knowing that the site was a ski/snowboard review application. They filled out a form whilst looking at the site. This can be found in Appendix 6.

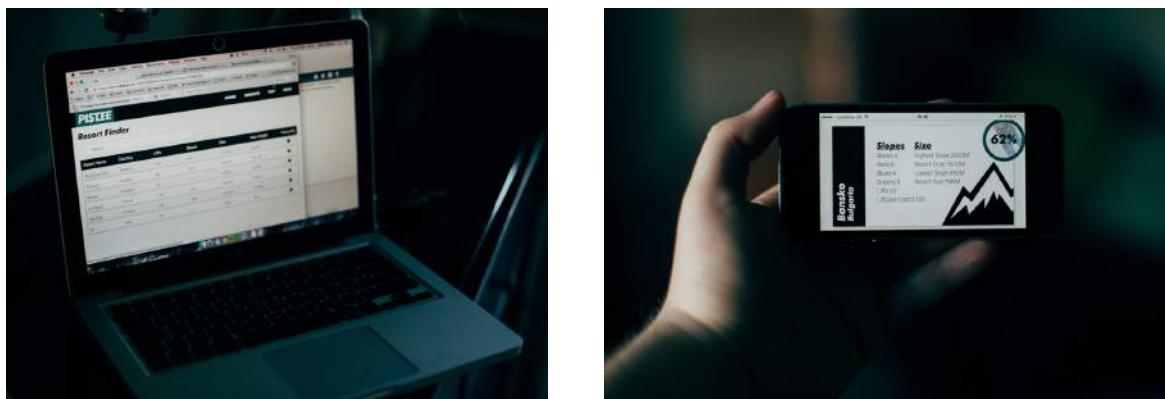
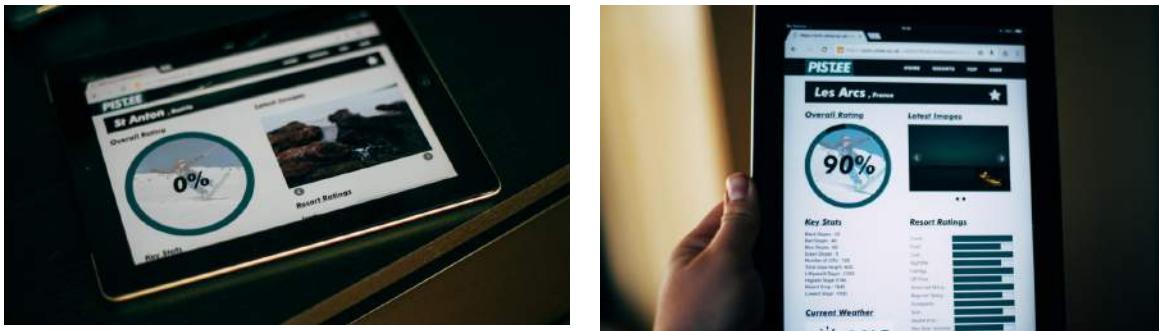


From the results, a few main trends were discovered. See a full report of the results in appendix 7. A lot of users didn't like the capital letters on forms, which is understandable, and can be easily resolved. All the users found the website easier to use than rival sites. This is a great result, but the site is a lot less complex than other sites, and doesn't include advertisements etc. Another feature that was mentioned was the favourites not being comparable which was originally a requirement of the site, which hasn't been implemented yet due to time constraints.

5.3.3 - Compatibility Testing.

Compatibility went very well for all devices, with the majority of media queries working as expected. The only device which had problems out of the 4 I tested was a portrait iPhone. This was because the media query for it was yet to be finished, which will be fixed in the near future. Below are some shots of the site across different devices.





The next steps for the functionality are getting it working seamlessly on mobile phones.

5.4 - User Survey Responses

From my user survey responses it has allowed new feedback on the completed version of the site. The user survey responses were very useful, but watching other people navigate throughout my site was also very useful. This allowed me to realise where users hesitated, which signified to me room for improvement. Below is a list of some improvements which could be made, due to watching users respond to the site.

- All caps seemed to confuse users in forms.
- People thought it would be a necessity to upload an image when leaving a review.
- The word skiing in buttons misled snowboarders into thinking the criteria wasn't suited to them.
- Redirect after sign up confused people as they assumed they'd be logged in.

- An introduction to what the site was could be useful.

These responses can now be used to plan future improvements to the website.

6.0 - Evaluation

6.1 - User Evaluation of Surveys

Overall, the general opinion of the site gathered from the user evaluation surveys was very good. All users said that the site felt easier to use than other travel sites which they had used, and felt the site would be useful. Having a range of IT abilities in my user test group was definitely important as depending on the level of someones ICT skills they picked up on different things.

Some of the users didn't notice bugs, and some did. From the review given in Appendix 7, the following changes will need to be made in the near future.

- Remove caps from forms.
- Show that it isn't necessary to upload an image.
- Change the word skiing on the buttons on the site.
- Automatically signed in after log in.
- An introduction added to the site.

6.2 - Project Outcomes

The main aim of the project was -

"To build a web application that will simplify the process of obtaining reviews about winter sports resorts. The application will allow users to make an account and enabling them to rate different resorts, search for resorts which suit them, upload photos, and save favourite resorts."

I feel this aim has been successfully met, due to the fact that the majority of the objectives and requirements of the site were finalised on time and completed to the quality that was initially planned. This means the project has been a success.

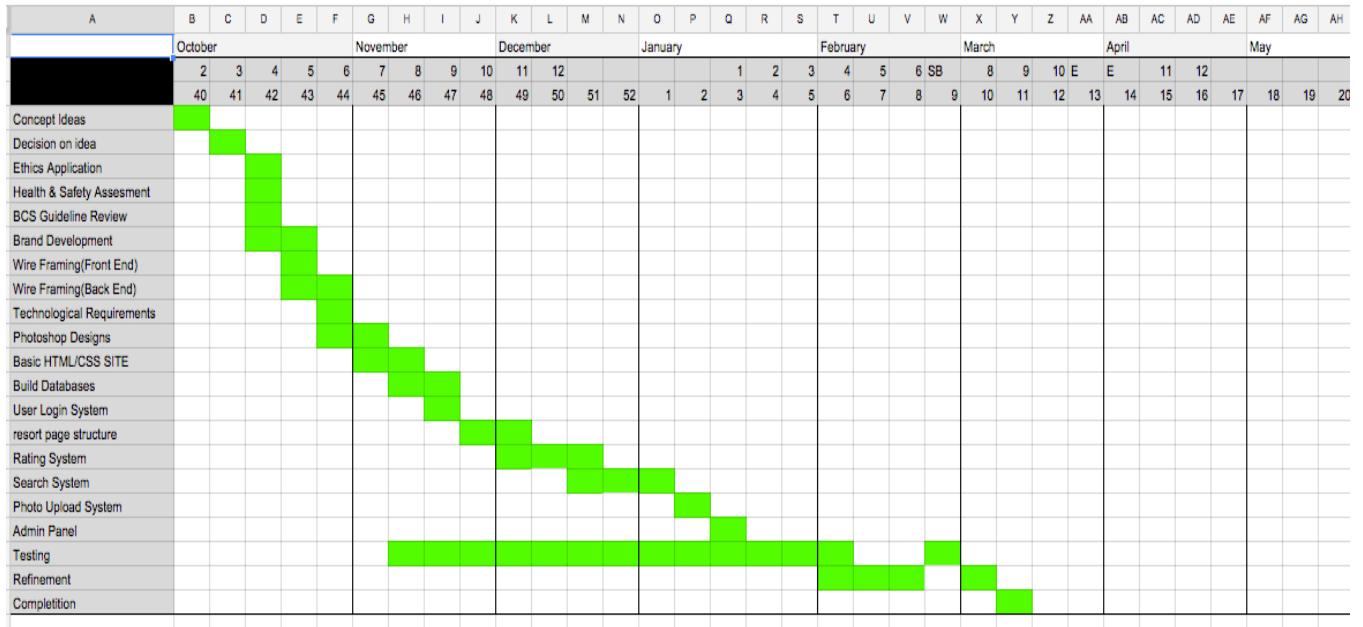
6.3 - Methodology

The methodology chosen for this project was Agile. After Reviewing the decision to use Agile at the end of the project, it was definitely a good decision to use it. This was mainly due to the style of testing which could be done. When a certain part of the project was completed, it could be tested and then redeveloped. Unlike a waterfall methodology, tasks could be started before another task has finished, allowing several aspects of the site to be worked on at one time. Even working as a single person team this was very useful, for example the rating system was very difficult to implement, so it could be left and returned to at a later date.

Another advantage of using agile was the limited planning needed to implement the project. This is due the fact changes can happen throughout the project, so the plan does not need to be 100% precise at the beginning of the project.

6.4 - The Plan

Time was always a very important constraint on the project; not only to meet the deadlines given, but to keep on top of the development process while still allowing sufficient time to allow for testing and improvements. To keep track of time a gantt chart was made, splitting the tasks up into different weeks. See attached gantt chart below.



Unfortunately, the expectations of the gantt chart were a little enthusiastic. This was due to the underestimation of the workload of both the major project and other modules. The schedule fell behind, but all deadlines were still met. The main issue caused by the schedule falling behind was unfortunately, I had insufficient time left at the end of the project for refinement.

In terms of resources, my project originally planned on using a snow report plugin. Unfortunately, the only one available for free wasn't suitable, due to the fact it could not be customised. Another part of the original plan that wasn't implemented was using frameworks for both the front and back end.

Apart from this, the project ran as planned and even though a few small setbacks were experienced I still judge the project as very successful.

7.0 - Conclusion

7.1 - Report Summary

Beginning with the idea generation and requirement specification of the project, I feel both these sections were completed extremely well giving me a good basis to start both

the system and visual design of the project. Paper prototyping was always going to be one of the weaker parts of my project. The paper prototypes may not be of a high standard, but they gave me the structure needed to go on and develop my project designs in Photoshop. In the design section, I am extremely satisfied with how the designs were produced, as these paired with the requirements specification simplified the implementation of my site greatly. Implementation of the site was also very successful, with very few features not working/included.

7.2 - Reflection

Overall I feel the outcome of the project has been a success. The final product is very similar to those in my designs, with only a few changes. Some of these changes were improvements and some were due to time/development constraints.

Time keeping was another issue with the project, I did find I experienced a few difficulties adhering to my predicted schedule. However, in spite of this all deadlines were met, which is the most important factor, but on hindsight I should have allocated more time to certain areas. For example, user testing took place on the 29th April which gave me a very limited time to make changes.

The project has been of great value in the improvement of my development skills, as seven months ago, when planning the project, I was extremely worried at my limited knowledge of PHP and scripting. Throughout the year, my development, design and management skills have improved greatly.

Basically, on reflection, if I was to amend my work schedule, I would definitely start earlier with the development of prototypes of parts of the site. I would also take time to implement a front end and back end framework (Bootstrap and Laravel). A frontend framework would make the process of making a responsive site much easier, as this was one of my main difficulties in the later stages of the project.

7.3 - Reflection of my Role

Working individually on a project of this scale is very challenging but rewarding. Managing, designing, developing and testing the project are the 4 main tasks which I undertook. This is quite a challenge, but it does allow things to get done efficiently as you aren't relying on other people to complete work which can have the effect of slowing down progress. It also helps you learn a great deal about your own capabilities.

However, I feel having only one person working on a project can lead to tasks simply getting overlooked, due to the fact the project is only being seen through one pair of eyes instead of several. This was picked up in my user testing which was the first time the site had been reviewed by anyone apart from myself.

I do feel I performed all the tasks to a high level, with only a very few select parts of the project causing some dissatisfaction.

7.4- Future Work

From the final evaluation, testing and user testing it has come apparent that some work needs to occur in the near future to make the application seamless. Some work could also be carried out in the distant future to improve the site further.

Work to be carried out immediately includes simple tasks found by the user requirements as shown in the list in section 6.1. Most of the jobs included in that list will be relatively easy to implement apart from the changing of the signup form. These changes will take place immediately so they are ready for the demonstration day.

Another piece of future work comes from the functional requirements. These are requirements which were set at the start of the project, which the project had to meet.

Below is a table showing the 10 requirements which were failed and how I plan to make them meet the requirements.

7. If Password is typed it enters in field as dots. If Password is not included login should fail.	Add a confirm password field in the signup form.
9. User can connect with their facebook account to use as a signup method, to help save time.	Add facebook login.
25. Show more button shows more results.	Show more button could be added, but is not necessary.
29. Key information on results section includes 3 highest ratings	No action needed as changed due to design problem with amount of data displayed.
32. Result ranking shown for country.	This could be added but again isn't necessary.
39. Written review max character input of 200 characters.	Add this feature to keep reviews short.
45. Key information on favourites section includes 3 highest ratings	No action needed as changed due to design problem with amount of data displayed.
47. User favourite can be compared to another favourite.	This feature should be added and be relatively easy to implement.
59. Snow conditions in table	No longer applicable.
65. Edit image sliders on admin table.	Future requirement, not necessary at this current stage.

From the compatibility testing, refinement is needed for the application when it is used on a smartphone.

References

What is Agile model – advantages, disadvantages and when to use it?. 2014. [ONLINE] Available at:

<http://istqbexamcertification.com/what-is-agile-model-advantages-disadvantages-and-when-to-use-it/>. [Accessed 8 October 2014].

Volere, n.d., Volere Template [Volere Template], [online]. Available:<http://www.volere.co.uk/template.htm> [30/04/2015].

jQuery, n.d., jQuery [jQuery], [online]. Available: <https://jquery.com/> [01/05/2015].

Meyer Web, n.d., CSS reset [CSS reset], [online]. Available:<http://meyerweb.com/eric/tools/css/reset/> [01/05/2015].

BX Slider, n.d., BX Slider [BX Slider], [online]. Available:<http://bxslider.com/> [01/05/2015].

Range Slider, n.d., Range Slider [Range Slider], [online]. Available:<http://andreruffert.github.io/rangeslider.js/> [01/05/2015].

James Fleet, n.d., Simple Weather [Simple Weather], [online]. Available:<http://simpleweatherjs.com/> [01/05/2015].

txt3rob, n.d., Private Secure Login System [Private Secure Login System], [online]. Available: <https://github.com/txt3rob/smsonetime> [01/05/2015].

Appendices

Appendix 1 - Functional Requirements.

Number	1
Type	Functional
Description	Logo
Rationale	The main Logo in the header.
Fit Criterion	The logo will show the main brand of the site, and be a link to the homepage of the site.
Dependencies	None

Number	2
Type	Functional
Description	Log in form - Username
Rationale	Allows the user to enter their username.
Fit Criterion	Simple input form with placeholder username. Should work on all devices. Javascript validation so only valid characters can be entered. This then checks username with database, comparing with password
Dependencies	Log in form - Submit button (4)

Number	3
Type	Functional
Description	Log in form - Password
Rationale	Allows the user to enter their Password.

Fit Criterion	Simple input form with placeholder password. This then checks username with database, comparing with password
Dependencies	Log in form - Submit button (4)

Number	4
Type	Functional
Description	Log in form - Submit
Rationale	Allows the user to submit their username/password combo, letting them login to the site.
Fit Criterion	Input button, which sends the username/password to database. If the combination is correct, the user will be logged in, allowing to access the additional pages of the site.
Dependencies	favourites page, change details page, leave a review page.

Number	5
Type	Functional
Description	Sign up form - Username
Rationale	Allows the user to enter username, so they have login details for the site.
Fit Criterion	Simple input with javascript validation to check if username is available, and if characters are correct.
Dependencies	Submit form - Submit button (10)

Number	6
--------	---

Type	Functional
Description	Sign up form password
Rationale	Allows the user to enter their desired Password, to sign into the site.
Fit Criterion	Javascript validation will be used to make sure password is 8 characters long, and uses letters and numbers.
Dependencies	Submit form- Submit button (10), confirm-password (7)

Number	7
Type	Functional
Description	Sign up form confirm password
Rationale	Allows the user to enter their desired Password again, so the password is definitely correct.
Fit Criterion	Javascript validation will be used to make sure password matches the previous password.
Dependencies	Submit form- Submit button (10)

Number	8
Type	Functional
Description	Sign up form email
Rationale	Allows the user to enter their email, so they can signup.
Fit Criterion	Simple input form with email placeholder. This then checks email with database, to make sure person hasn't already signed up. Javascript validation will be used to make sure email is correct format.
Dependencies	Submit form- Submit button (10)

Number	9
Type	Functional
Description	Sign up with facebook
Rationale	Allows the user to use their facebook details to sign up, to save time.
Fit Criterion	User can connect with their facebook account to use as a signup method, to help save time.
Dependencies	Submit form- Submit button (10)

Number	10
Type	Functional
Description	Sign up form submit
Rationale	Allows the user submit their details and make a user account.
Fit Criterion	Submit button adds details to database and makes an account. User will automatically be logged into the site.
Dependencies	Submit form- Submit button (10)

Number	11
Type	Functional
Description	Primary Menu - Home Button
Rationale	A button to link to the home page, allowing users to easily navigate back to the main page of the site.
Fit Criterion	The button should look like a button, with the text easy to read on

	all devices.
Dependencies	Home Page

Number	12
Type	Functional
Description	Primary Menu - Top lists Button
Rationale	A button to link to the top lists page.
Fit Criterion	The button should look like a button, with the text easy to read on all devices.
Dependencies	Top lists page.

Number	13
Type	Functional
Description	Primary menu - Resorts button
Rationale	A button to link to the resorts page.
Fit Criterion	The button should look like a button, with the text easy to read on all devices.
Dependencies	Resorts page

Number	14
Type	Functional
Description	Primary Menu - User button
Rationale	To link to the secondary drop down menu for the user
Fit Criterion	The button should look like a button, with the text easy to read on all devices.

Dependencies	Dropdown menu (15-18)
--------------	-----------------------

Number	15
Type	Functional
Description	Secondary menu - Log in button
Rationale	Takes the user to a login page, so they can have the ability to leave a review and add favourites.
Fit Criterion	Be part of drop down menu, and link to the login page.
Dependencies	Login page

Number	16
Type	Functional
Description	Secondary menu- Sign up button
Rationale	Takes user to a signup page/form, allowing them to make an account
Fit Criterion	Be part of dropdown menu, and link to the signup page.
Dependencies	Sign up page.

Number	17
Type	Functional
Description	Secondary menu - Review button
Rationale	Takes user to a review page. If not logged in takes them to login page.

Fit Criterion	Be part of a dropdown menu, and links to review page
Dependencies	Review page

Number	18
Type	Functional
Description	Favourites button
Rationale	Takes user to their favourites page. If not logged in takes them to login page.
Fit Criterion	Be part of dropdown menu, and links to favourites page.
Dependencies	Favourites page

Number	19
Type	Functional
Description	Secondary menu - Edit account button
Rationale	takes user to the edit account page. Only appears when logged in.
Fit Criterion	Be part of dropdown menu, and links to update account page.
Dependencies	Update account page.

Number	20
Type	Functional
Description	Find Resorts Page - Image Gallery
Rationale	Allows the main page to have a gallery, which could possibly be

	used for advertisements, or for general photos
Fit Criterion	This will be full width of screen, with an automated transition. Text/links will be able to placed in front of each image.
Dependencies	none

Number	21
Type	Functional
Description	Find Resorts Page - Select buttons
Rationale	Allow the user to select up to twelve criteria which are important for them.
Fit Criterion	Each button will have an image relating to the item, or a circular design with text. When selected, they will be highlighted.
Dependencies	Results page.

Number	22
Type	Functional
Description	Find Resorts Page - Submit button
Rationale	Submits the users request, and queries the database to find the ideal resorts in order of best rating.
Fit Criterion	Button will look like a button, and match the style of the site.
Dependencies	Results page.

Number	23
--------	----

Type	Functional
Description	Footer
Rationale	Footer of page will include primary menu, secondary menu and drop down menu as simple links, for ease of navigation
Fit Criterion	Links will have a simple style, such as underlined text.
Dependencies	Pages linked too.

Number	24
Type	Functional
Description	Results Page - Main layout
Rationale	The main way the results will be displayed in the page, so the user can easily navigate them
Fit Criterion	The results will be displayed in a single column page, with each result having its own section displaying brief information about the resort.
Dependencies	Key info for inside each resort, resorts page from link.

Number	25
Type	Functional
Description	Results Page - show more button
Rationale	Allows more results to be shown
Fit Criterion	Minimalist button, like an arrow.
Dependencies	Extra Results

Number	26
Type	Functional
Description	Results Page - key information - name
Rationale	The name displayed for each result.
Fit Criterion	Large name inside the section for each result.
Dependencies	None

Number	27
Type	Functional
Description	Results Page - rating
Rationale	So the user knows how suited the resort is to them as a percentage
Fit Criterion	Using a circular HTML5 Progress bar, with the percentage displayed in the middle.
Dependencies	None.

Number	28
Type	Functional
Description	Results Page -key information - Mountain Stats
Rationale	So the user gets a slight insight into the resort, before clicking into the full page
Fit Criterion	Be easy to read, with a mountain graphic for resort heights

Dependencies	none
--------------	------

Number	29
Type	Functional
Description	Results Page - key information-best for section
Rationale	Informs the user of the 3 highest ratings out of 12 for the resort.
Fit Criterion	Follow similar design to key info.
Dependencies	none.

Number	30
Type	Functional
Description	Resorts Page - review ratings
Rationale	Informs the user of each of the ratings for the 12 criteria of the site, allowing them to decide if the resort is suited to them.
Fit Criterion	Use Circular HTML 5 progress bars with the name below, and rating in the middle.
Dependencies	none.

Number	31
Type	Functional
Description	Resorts Page - key mountain stats
Rationale	Informs the user of the key stats about the mountain.

Fit Criterion	Be in a list form, with icons.
Dependencies	none

Number	32
Type	Functional
Description	Resorts Page - result ranking
Rationale	How the result is ranked against other resorts. Ie 14/26 in france.
Fit Criterion	Easy to read, and inline with design
Dependencies	none

Number	33
Type	Functional
Description	Resorts Page - written reviews
Rationale	Written reviews of the resort
Fit Criterion	Single column of reviews, giving name of reviewer, date, and the review.
Dependencies	none.

Number	34
Type	Functional

Description	Resorts Page - weather
Rationale	Allows users to view current weather for resort
Fit Criterion	Be displayed as tabular data, so its easily read.
Dependencies	none.

Number	35
Type	Functional
Description	Resorts Page - images
Rationale	An image gallery of images left by reviewers.
Fit Criterion	small image gallery, which expands when clicked.
Dependencies	none.

Number	36
Type	Functional
Description	Leave a review - Resort Select
Rationale	Allows the user to select what resort they are reviewing.
Fit Criterion	A select box which allows a user to select a country, followed by a resort name.
Dependencies	Reviews

Number	37
--------	----

Type	Functional
Description	Leave a review - Rating Sliders
Rationale	Allows the user review the resort using criteria. The review will be from the number 1-5
Fit Criterion	The sliders will be circular, and user can drag round to select. Not all selections have to be completed.
Dependencies	Reviews

Number	38
Type	Functional
Description	Leave a review - photo upload
Rationale	Allows the user to upload an image from there holiday, so other users can see what it would be like
Fit Criterion	The upload button should be clearly a button for photo upload, and a notification should be made when upload is complete.
Dependencies	Reviews

Number	39
Type	Functional
Description	Leave a review - written review
Rationale	Allow the user to leave a 200 character review about the resort.
Fit Criterion	Short text box with 200 character count down. Idea comes from the idea of short feedback like eBay use, so it is quick and easy to complete.
Dependencies	Reviews.

Number	40
Type	Functional
Description	Leave a review - Submit Button
Rationale	Submits review to database.
Fit Criterion	Standard button matching style of site.
Dependencies	Reviews table in database.

Number	41
Type	Functional
Description	User Favourites - Main layout
Rationale	The main way the results will be displayed in the page, so the user can easily navigate them
Fit Criterion	The results will be displayed in a single column page, with each result having its own section displaying brief information about the resort.
Dependencies	Key info for inside each resort, resorts page from link.

Number	42
Type	Functional
Description	User Favourites - key information - name
Rationale	The name displayed for each result.
Fit Criterion	Large name inside the section for each result.
Dependencies	None

Number	43
Type	Functional
Description	User Favourites - rating
Rationale	So the user knows how suited the resort is to them as a percentage
Fit Criterion	Using a circular HTML5 Progress bar, with the percentage displayed in the middle.
Dependencies	None.

Number	44
Type	Functional
Description	User Favourites -key information - Mountain Stats
Rationale	So the user gets a slight insight into the resort, before clicking into the full page
Fit Criterion	Be easy to read, with a mountain graphic for resort heights
Dependencies	none

Number	45
Type	Functional
Description	User Favourites - key information-best for section

Rationale	Informs the user of the 3 highest ratings out of 12 for the resort.
Fit Criterion	Follow similar design to key info.
Dependencies	none.

Number	46
Type	Functional
Description	User Favourites - delete button
Rationale	Allows the user to delete the favourite.
Fit Criterion	Small button with trash can icon for usability
Dependencies	favourites page.

Number	47
Type	Functional
Description	User Favourites - compare.
Rationale	Allows user to compare their favourites.
Fit Criterion	Small button with compare icon/word
Dependencies	none.

Number	48
Type	Functional
Description	Top Lists Page - Submenu
Rationale	Allows the user to select which of the 12 toplists to view.
Fit Criterion	Uses icons consistent with homepage. Loads list via ajax.
Dependencies	Lists.

Number	49
Type	Functional
Description	Top Lists - key information - name
Rationale	The name displayed for each result.
Fit Criterion	Large name inside the section for each result.
Dependencies	None

Number	50
Type	Functional
Description	Top Lists - rating
Rationale	So the user knows how suited the resort is to them as a percentage
Fit Criterion	Using a circular HTML5 Progress bar, with the percentage displayed in the middle.
Dependencies	None.

Number	51
Type	Functional
Description	Top Lists -key information - Mountain Stats
Rationale	So the user gets a slight insight into the resort, before clicking into the full page
Fit Criterion	Be easy to read, with a mountain graphic for resort heights
Dependencies	none

Number	52
Type	Functional
Description	Top Lists - key information-best for section
Rationale	Informs the user of the 3 highest ratings out of 12 for the resort.
Fit Criterion	Follow similar design to key info.
Dependencies	none.

Number	53
Type	Functional
Description	Resort Search Page - Search
Rationale	Allows users to select resorts using countries, so they can view all resorts in one country
Fit Criterion	Simple search box with select options for countries.
Dependencies	resort search table

Number	54
--------	----

Type	Functional
Description	Resort Search Page - Name
Rationale	Name of resort table column
Fit Criterion	Column of table with resort name
Dependencies	none.

Number	55
Type	Functional
Description	Resort Search Page - Height
Rationale	Resort height data so user knows heights of mountains.
Fit Criterion	Column of table with lowest and highest heights plus difference.
Dependencies	none.

Number	56
Type	Functional
Description	Resort Search Page - Size
Rationale	Size of resort table column
Fit Criterion	Column of table with number of KM of pistes, Number of runs, number of lifts.
Dependencies	none.

Number	57
--------	----

Type	Functional
Description	Resort Search Page - rating
Rationale	Rating of resort table column
Fit Criterion	Column of table with resort rating
Dependencies	none.

Number	58
Type	Functional
Description	Resort Search Page - Temperature
Rationale	Current temperature of resort table column
Fit Criterion	Column of table with resort temperature
Dependencies	none.

Number	59
Type	Functional
Description	Resort Search Page - snow conditions
Rationale	Snow conditions table column
Fit Criterion	Column of table with snow depths at low/high points
Dependencies	none.

Number	60
--------	----

Type	Functional
Description	Admin Page - Menu
Rationale	Menu with all links to Admin pages.
Fit Criterion	Allows the admin user to easily navigate through pages. Menu will be a sidebar, with links in a vertical list.
Dependencies	Admin pages

Number	61
Type	Functional
Description	Admin Page - Approve reviews
Rationale	Allows admin to approve reviews
Fit Criterion	Allows the admin to view reviews, and tick a box to approve them.
Dependencies	reviews.

Number	62
Type	Functional
Description	Admin Page - Edit users
Rationale	A page where admin can search for a user and edit their details/delete accounts.
Fit Criterion	Allows the admin user to easily navigate through pages. Menu will be a sidebar, with links in a vertical list.
Dependencies	Admin pages

Number	63
Type	Functional
Description	Admin Page - Edit reviews
Rationale	Allows admin to edit & delete reviews.
Fit Criterion	Allows the admin to edit reviews.
Dependencies	Reviews

Number	64
Type	Functional
Description	Admin Page - Edit Resorts
Rationale	Allows the user to select a resort, and make changes to it.
Fit Criterion	Easy to use interface, with resorts easily selected.
Dependencies	Resort pages.

Number	65
Type	Functional
Description	Admin Page - Edit Image sliders
Rationale	Allows the image sliders and the text associated with them to be easily edited.
Fit Criterion	Allows the user to upload new images, and change descriptions.
Dependencies	Admin pages

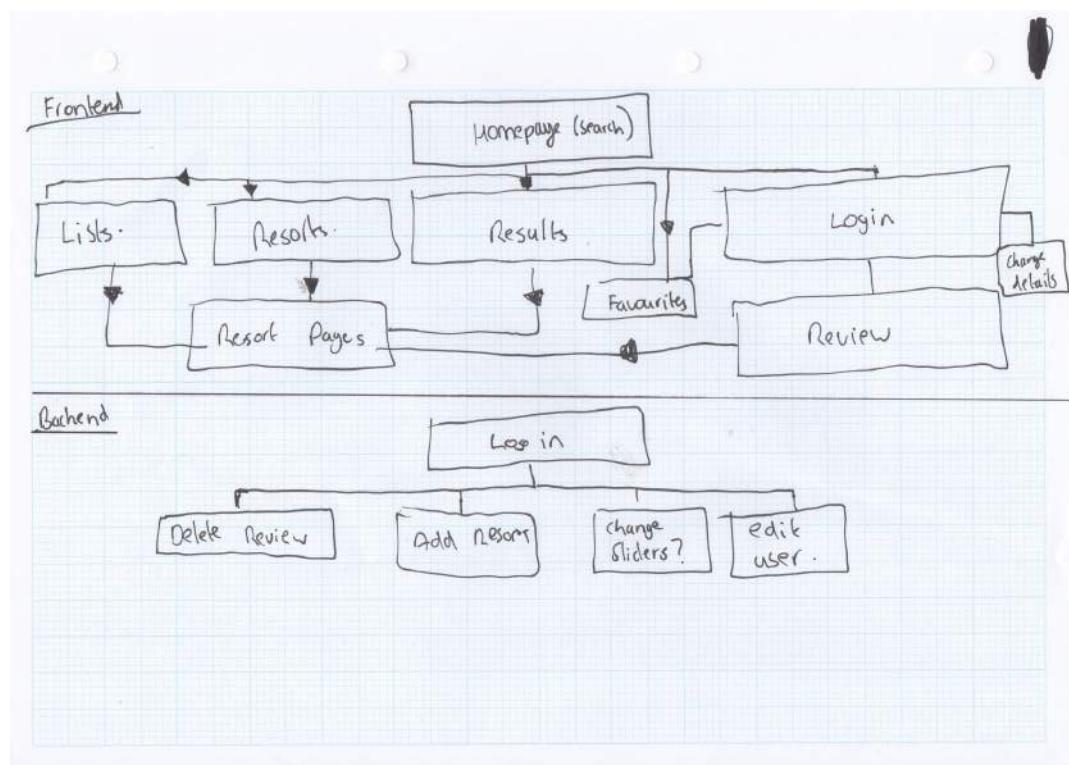
Appendix 2 - Paper Prototyping

Initial Thoughts

Designing a website always starts on paper. Jumping straight into photoshop or your code editor always leads to a longer process, usually leading to a less structured site, with less scope to become something modern. With paper prototyping, it takes a few minutes to sketch up some ideas for a page. This could be deciding where to place a header or a menu? Or seeing if something completely different will work. If this was done by experimenting in photoshop or CSS it could take hours, which in the industry can't happen due to tight deadlines and budgets. For me, Paper prototyping is all about deciding what layout to go for, as my drawing/designing on paper skills have never been great. Even though this is the case, it still helps when staring at a blank canvas on photoshop to have an idea where to start.

Flow Chart

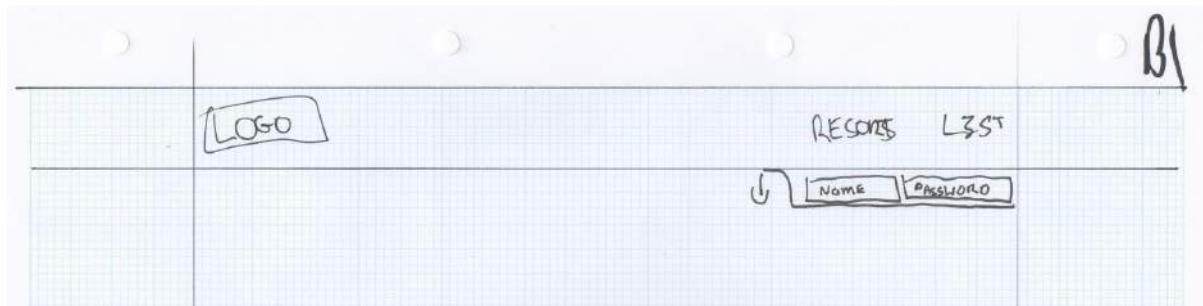
The first process for me was making a simple flow chart, to help figure out how many pages the website would need, and how they would link back and forth to each other. This would help when designing navigation for the site.



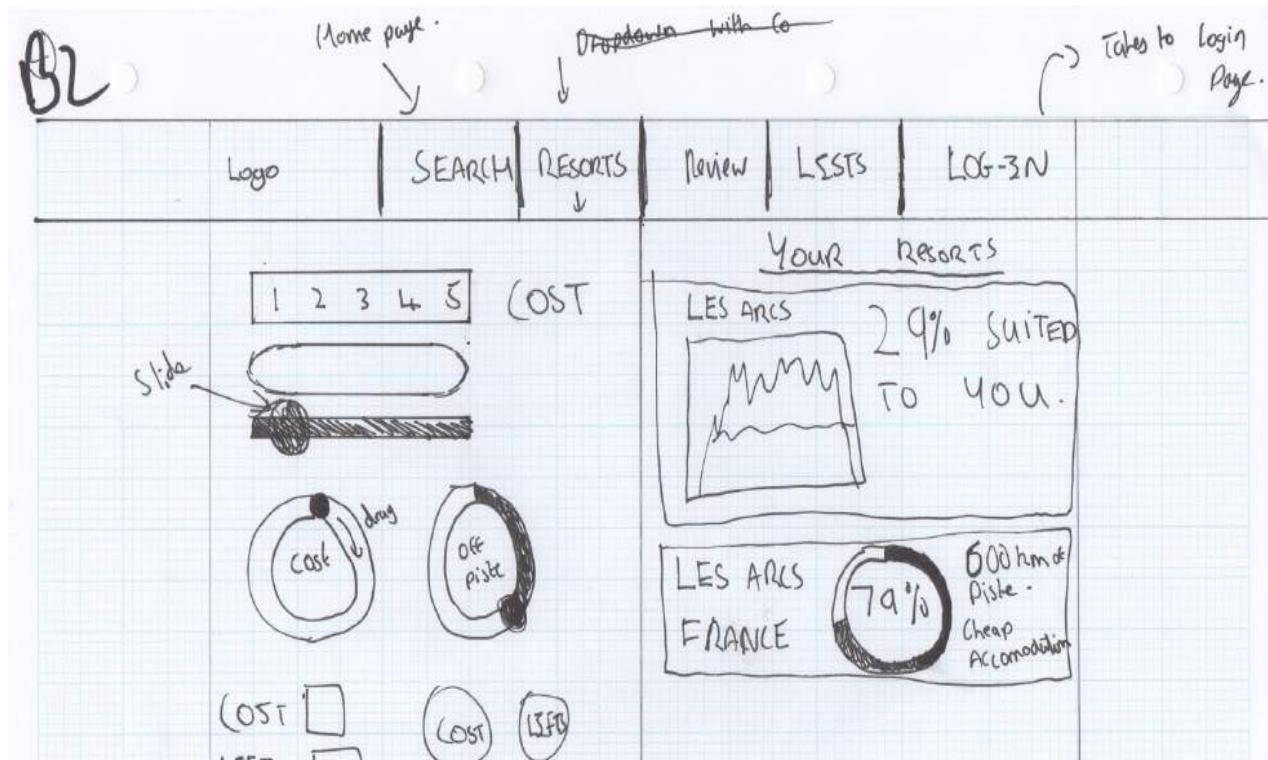
Focusing on the frontend of the site, all pages end up linking to the individual resort page. This is because this is the key focus of the site, to help users find a resort which is suited to them.

Initial Designs

The first step of the prototyping was sketching up some simple ideas. Below are a few of my sketches.



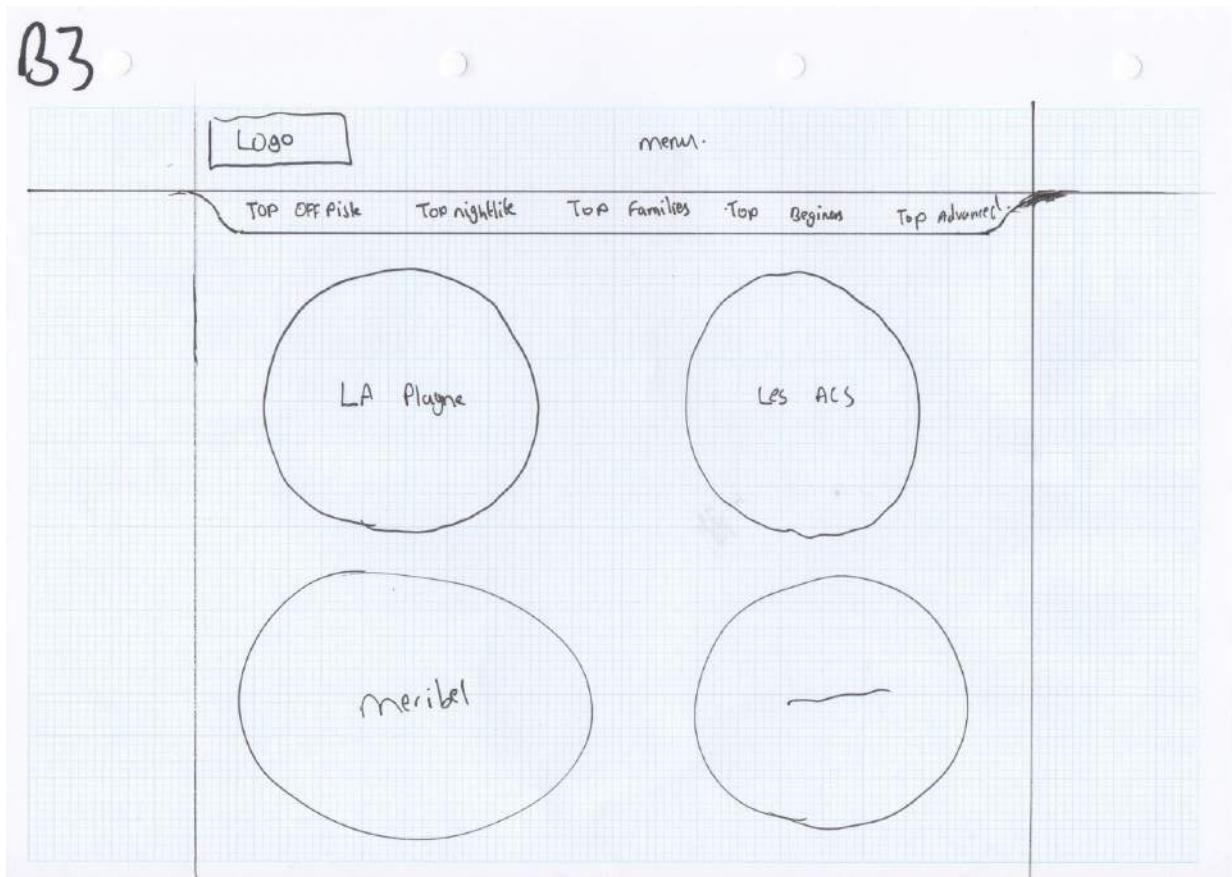
Above is a sketch for the header. This was sketched at a very early stage, before I had completed my flow chart. At this stage I hadn't realised that I'd have so many links in my primary navigation. It helped make the decision that the header would be a sticky header at the top of the page, as being able to navigate the site easily is one of my main criteria.



The above sketch was an idea of using a two column layout, with the search on the right column and the results in the left. This idea was ok, but was not used due to the fact that

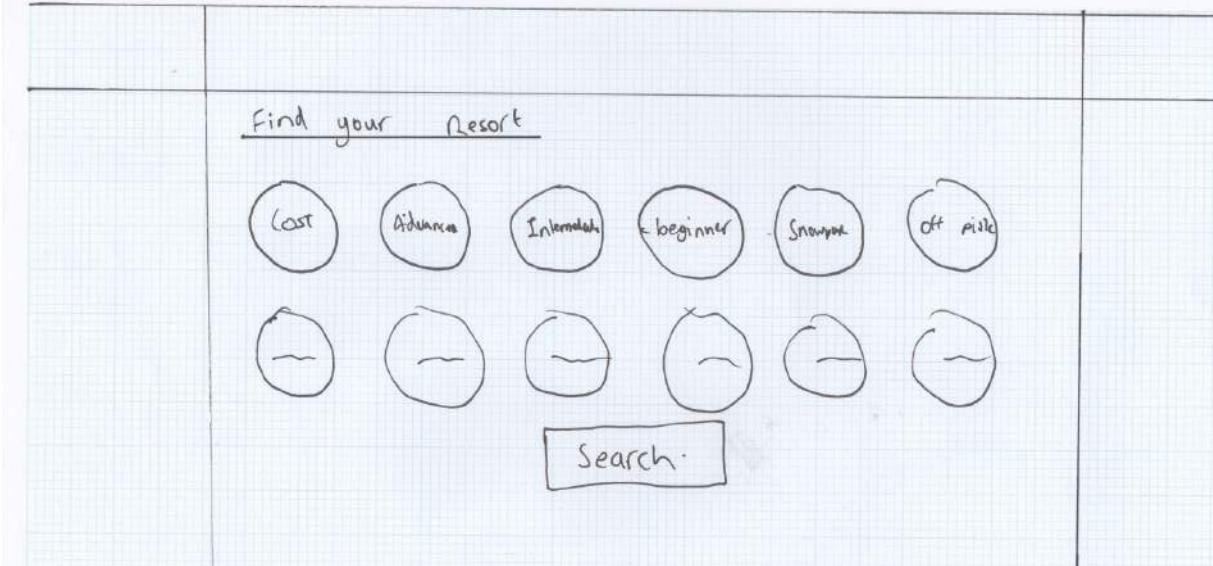
enough information wouldn't be displayed in the results. The left column was used to experiment with different methods for searching for results. The initial plan was to have a rating system here, allowing the user to rate which options were most important. When looking into building this, it appears that it would be too complex, so a checkbox option will be used instead. Out of the selection of input methods, the bottom right circles seem to be the favourite. Once clicked, the circles will be highlighted a different colour. 10 options will be available for the user, but not all will need to be selected.

The page was also used to experiment with the header. It was a lot more suited than the first option, but a drop down menu would be needed to house a secondary navigation for the user to select options such as logging in, signing up, reviewing and changing their details.



Above is an experiment for displaying information on the lists page. The main ideas for the site involve a lot of structures, but for the information needed to be displayed these were not ideal. The sub navigation also wouldn't work, as there would be ten options for organising the resorts by, so back to the drawing board for this page.

154



Above is a concept for the main page of the site. This takes the idea from B2 of the circular checkboxes. It's also helped to discover another problem which could be the number 10. Having 10 options makes sense for the rating algorithm, but for displaying 10 items on a responsive site they don't really respond well. Changing it to 12 options would allow 2 rows of 6 to be displayed on large monitors, and when on smaller monitors either rows of 2,3 or 4 could be displayed.

B5

(3)

LOGO

RESORT NAME

Slider from uploaded images

RESORTS TOP LISTS

[GREY]	2a	Lifts	80
[WHITE]	31	pistes	284km
[RED]	37	parts	2
[Black]	4	Food on/off	29.

Top Slope 3820m
Drop = 2000m
Bottom slope +600m

Advanced Shiny
nightlife
overall Score
7.5/10
10th Best RESORT

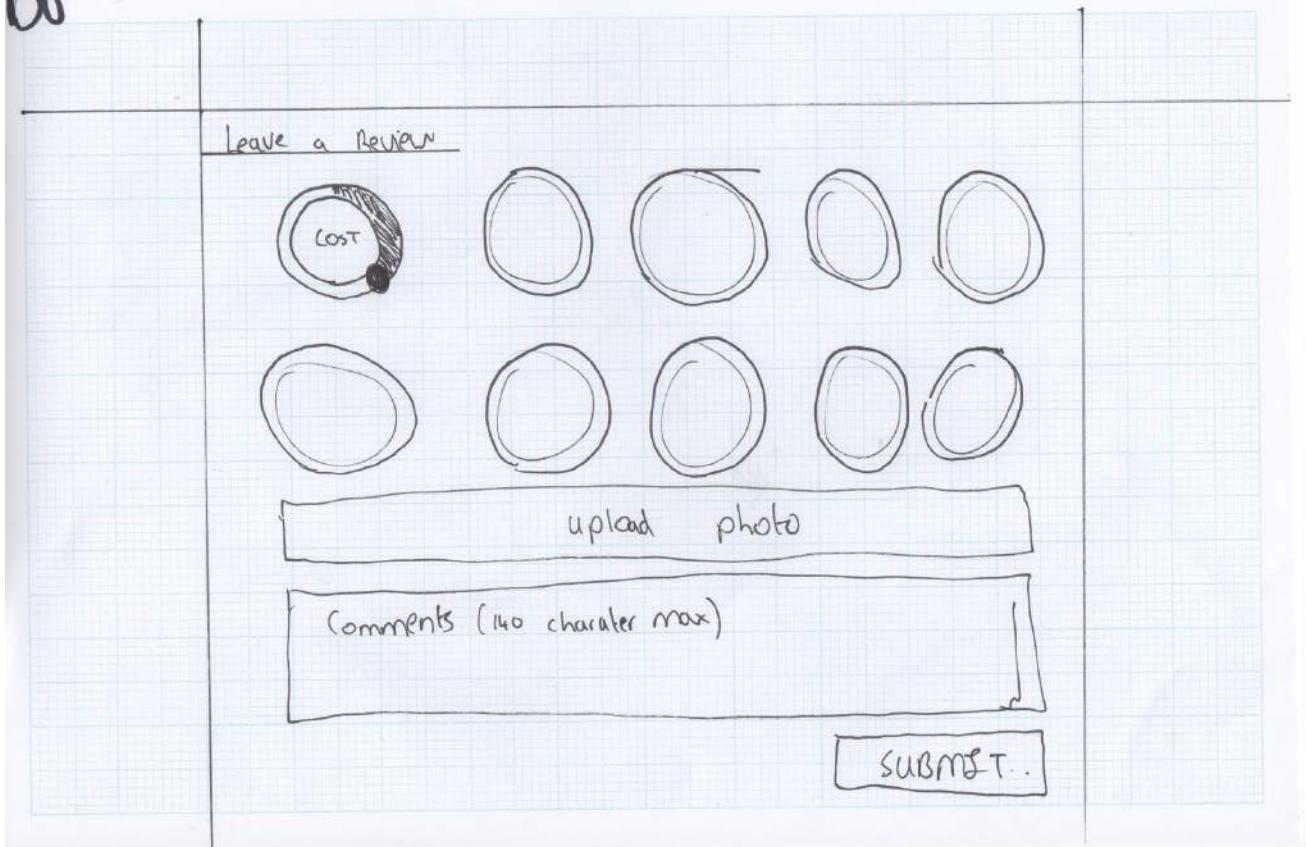
show more

opens shot reviews & weather

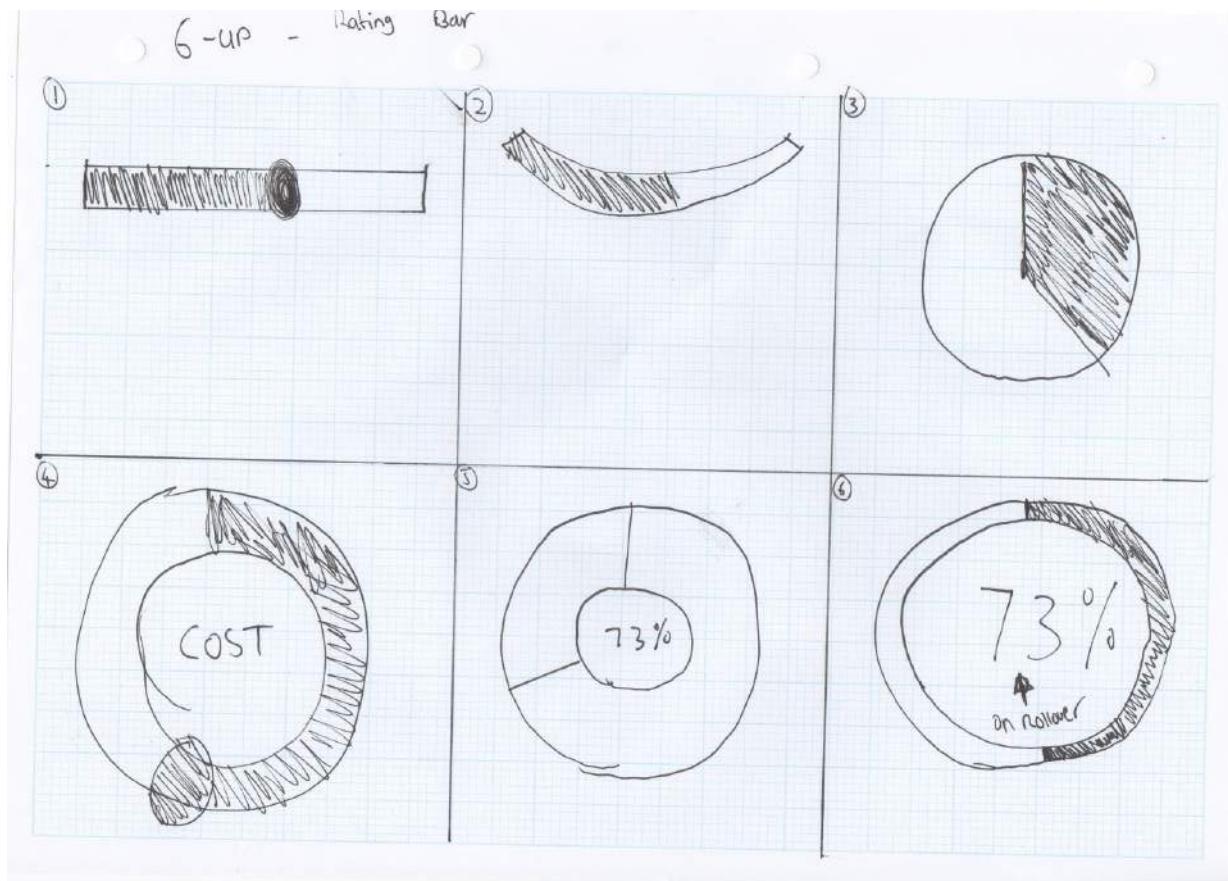
Above is a concept for the page which would show the information for the resorts pages. The two column layout works very well. However the page does not match the circle theme. It includes all of the information needed, once the show more button is pressed. The final layout could be very similar to this, but with circular progress bars for ratings.

B6

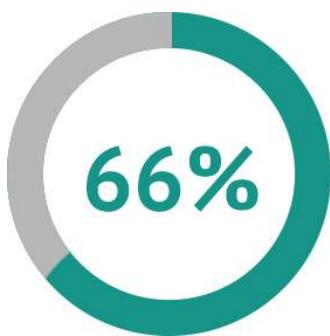
②



Above is a concept for the page for leaving a review. This uses the circular progress bars which could be dragged around by the user to select the value they want to give each rating. Another consideration for this page would be making it a two column layout for when on a normal computer, and one column for on mobile devices.

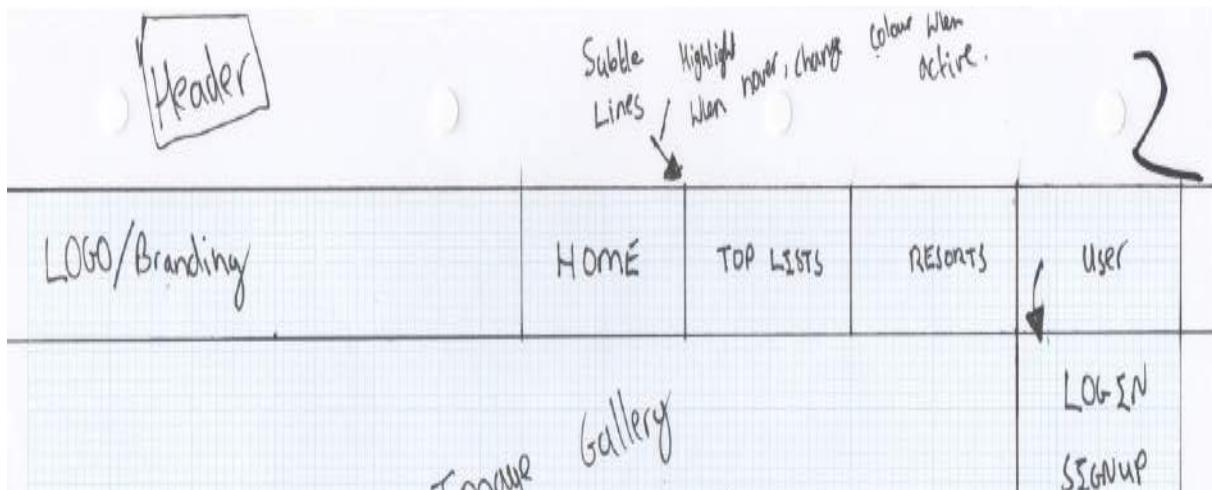


One of the most important factors for the site was the styling of the progress bar/slider. Originally the plan had been for a standard progress bar, but as my idea developed it made sense to use a circular bar to match the rest of the site. The Final product use a slider with data housed in the middle which will change on rollover. This slider style will be for displaying the data. For entering the data, a small circle will be included allowing the user to pull the slider.



Possibility for final design.

Final Wireframes



The first part of my design process was finalising how the header would look. This would house both the primary and secondary navigation for the site; allowing pages to be easily navigated, as the header would always be fixed to the top of the page. The Primary navigation allows the user to navigate to the three main ways of getting to the final resort pages, allowing them to find out information about their chosen destination in a variety of ways.

AI

Header

Switch to
Report Page

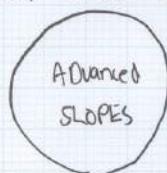
Image Gallery



Find your resort, Much what's important.



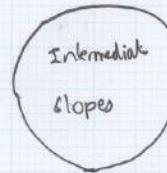
COST



Advanced
SLOPES



Beginner
Slopes



Intermediate
Slopes



Off
Piste



Food



Nightlife



APRES SKI
Activities



SHOPPING



FAMILY

HL

Search

Footer

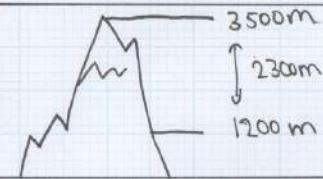
The main search page would consist of the 10 options (Possibly becoming 12). Any number of these could be checked, which will then decipher which resorts will be displayed. The image gallery could be used for advertising, or to link to important parts of the page.

HS

Page:

Header.

LES
ARCS

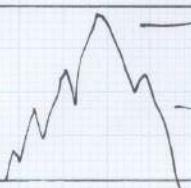
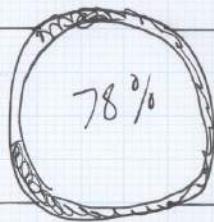


BLACKS	:	12
REDS	:	27
BLUES	:	53
GREENS	:	4
LBITS	:	112
SZE.	:	247 km ²

ALL Hyperlinked to Results page

BEST FOR
NIGHTLIFE
SHOWPARTIES
ADVANCED.

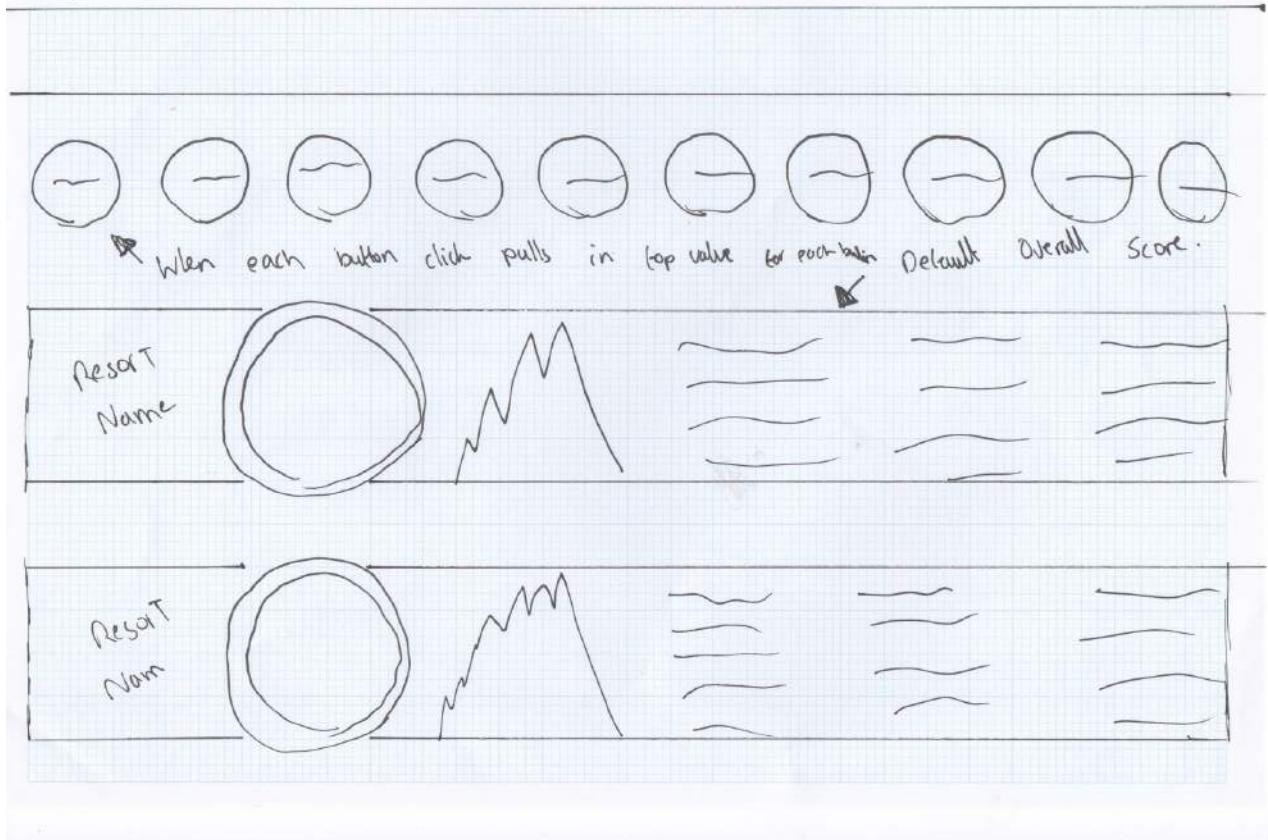
MERIBA



✓ ← MORE Load or Scroll.

When the user has searched the results will be returned in the above manner. Initially 4 sets of results will be displayed. When the entire section is hovered over, it will be highlighted allowing the user to then click through to that resorts page. The circles house how suited the resort is to the user which will be calculated using the options they have selected. If the user selects 3 options, and each option is worth 7,4,5 out of ten, an average of these numbers will be taken, and used to display the resorts which are best suited.

ff4



The above page is how the top lists page will look, with the consistent boxes to the results page. Only one button can be checked at a time, and once checked it will dynamically change the content. The circle will display the average rating for that category as a percentage.

AS

header

SELECT Country

or

Search

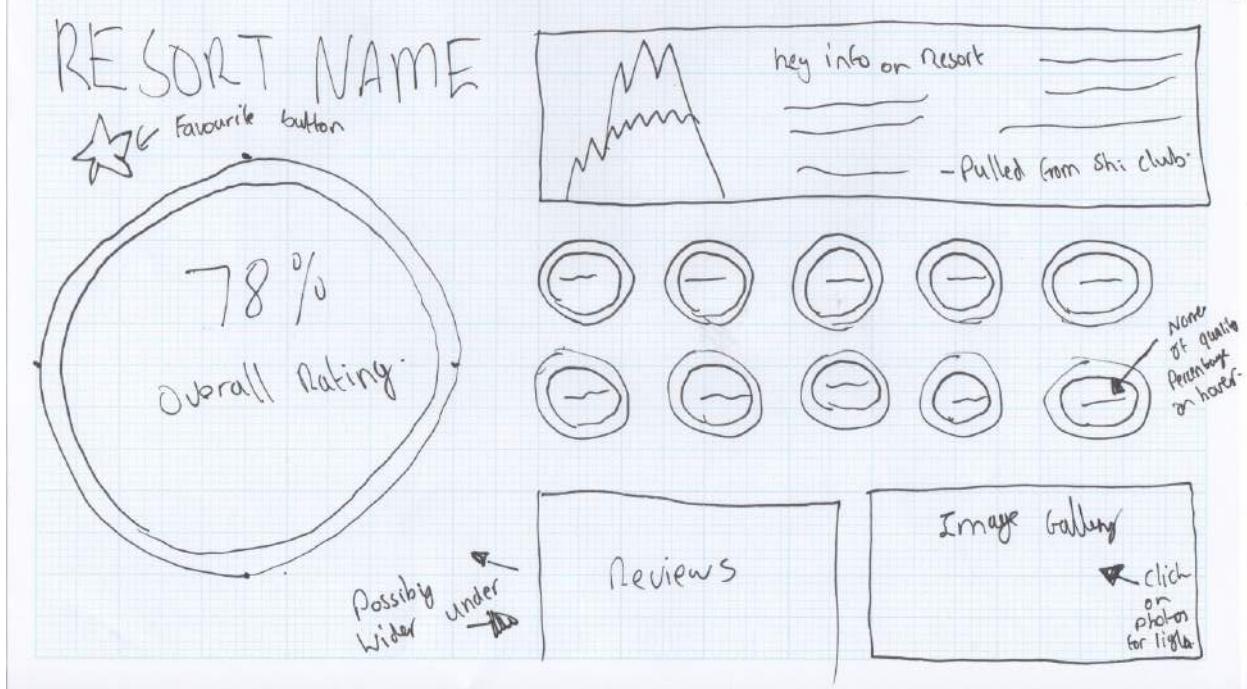
Alternating colors
↓
Resorts appear in table

NAME	HEIGHT	Size	Rating	Temp	Snow

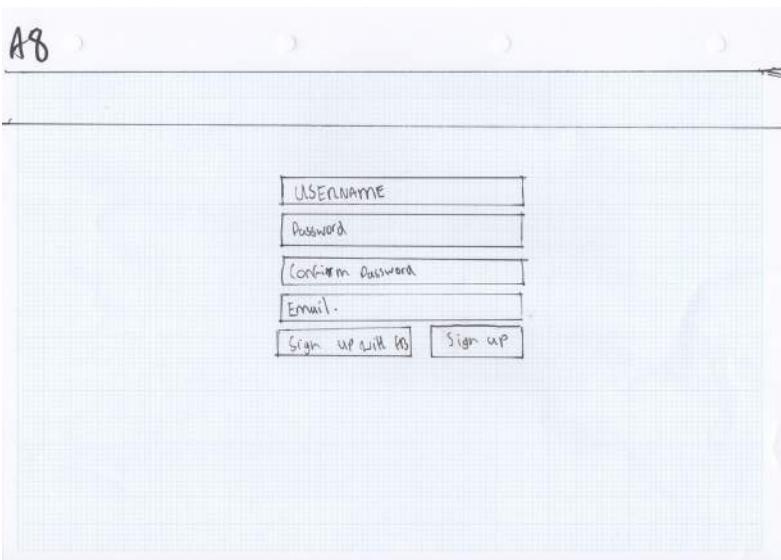
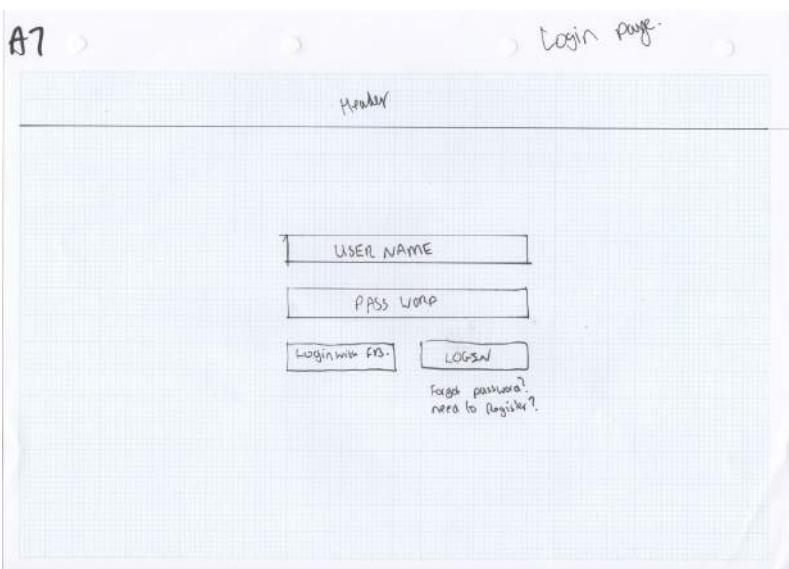
The above page is a simple page which will allow users to select a resort without searching via criteria. The select box will be pulled from the database, with the selected countries, or the database can be searched for resort names. Once the results are displayed, the individual options can be clicked, taking you to a full page about that resort.

A6

Resorts
Page



The above page is how the resort page will look. In the left column, the resort name will be displayed, as well as the overall rating. In the right column the key resort information such as the number of lifts, heights and slopes will be displayed, followed by the 10 individual ratings. Under this an image gallery of uploaded photos of the resort will be included, as well as the individual reviews. On a mobile device, the layout will change to a one column layout.



The login in /signup screen will be very simple, with the option to use a facebook login.

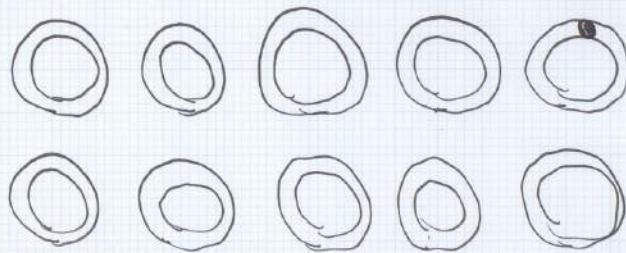
Aa

Header

1. Select Country

2. Select Resort.

3. Review



Describe in 200 characters.

UPLOAD Photo .

Submit.

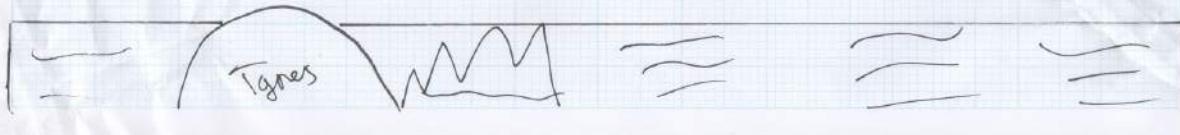
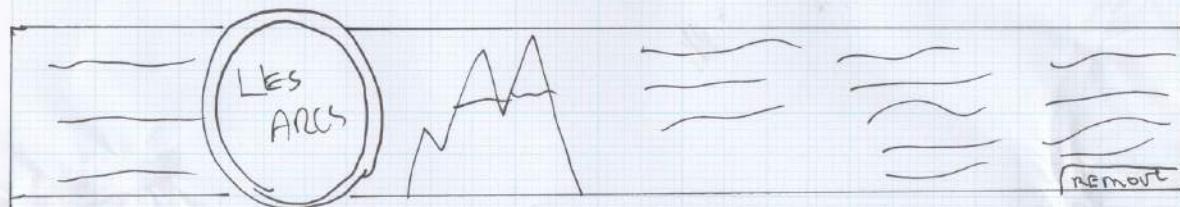
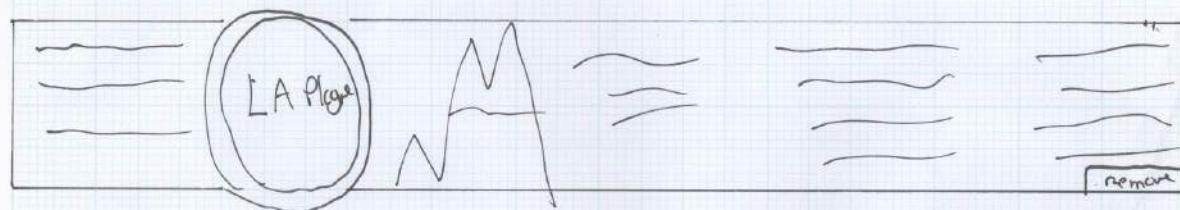
For the review page, a two column layout will be used unless the user is on a mobile device, or portrait tablet. The user will select the country/resort they wish to use which will be pulled from a database, then selected which ratings they want to give. They can then give a short 200 character review and upload a photo.

AD

Favourites

hidden

Favourites.



The favourites page will allow users to save resorts which are best suited to themselves. This page follows the same style as the other pages on the site.

The Backend

The backend of the site will not be directly linked from the homepage. This will allow admin (myself for now) to log in and review/delete reviews, edit users and edit tables in the database. It will have a sidebar allowing the user to navigate through the different available options, which will be displayed on the right hand side of the page. These will allow tables to be searched and records edited. For the approve system, this will allow the admin to only allow reviews to be posted once submitted, allowing offensive material to be filtered.

A hand-drawn wireframe diagram on grid paper. On the left, a vertical sidebar contains the following menu items: APPROVE REVIEWS, EDIT USER, REMOVE USER, EDIT REVIEWS, EDIT RESORTS, and Change Slider. To the right of the sidebar is a main content area featuring a table with four columns: Average Rating, Text, Photo, and Approve. The first row of the table has the value '2 %' in the Average Rating column, three wavy lines in the Text column, a small rectangle in the Photo column, and a checkmark in the Approve column. The second row of the table is completely blank.

Average Rating	Text	Photo	Approve
2 %	~~~~~	_____	✓

Appendix 3 - Feasibility Questionnaire

Snowboard / Ski Information/Review Site

A short form to gather information about the need for a better website/application to find out information about ski holidays.

Name

Age

How many times have you been Skiing/Snowboarding?

- 0
- 1-2
- 3-5
- 6+

Do you book your holidays online?

- Yes
- No

Do you use comparison sites to find out information?

- No
- Yes

Give up to 5 criteria which help you choose your holiday.

Cost, Slopes, Activities, Nightlife ETC

Do you feel it is hard to find out information about resorts in a central location?

Would you use a website which would help you find this information?

- Yes
- No
- Other:

Submit

Never submit passwords through Google Forms.

11 responses

[View all responses](#)

Summary

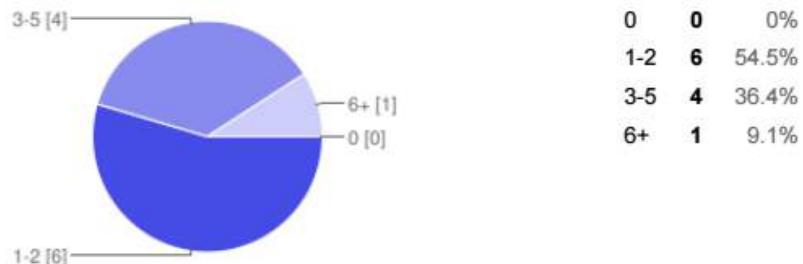
Name

Craig Reid
Natalie Leathem
Ivan Reid
Ann Adair
Lyndsey Reid
Alan Stewart
Ashleigh Baxter
Hannah Adair
Marty Rodgers
Matthew Breadon
Shaun

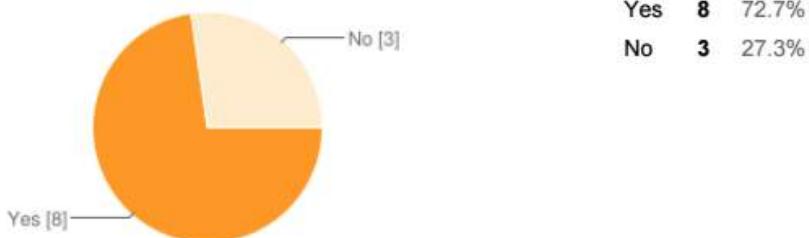
Age

24
25
26
19
21
20
Stenson
48
53

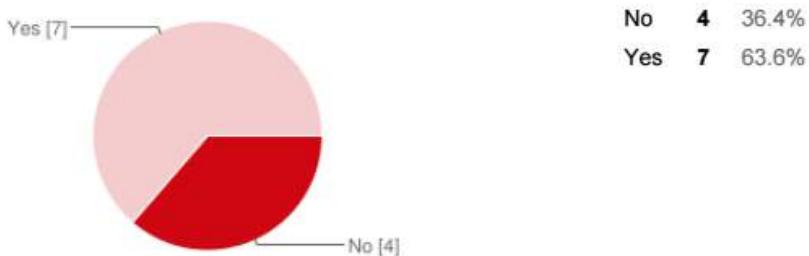
How many times have you been Skiing/Snowboarding?



Do you book your holidays online?



Do you use comparison sites to find out information?



Give up to 5 criteria which help you choose your holiday.

- Nightlife, Cost, Size of resort, good off piste
- Nightlife, Short Transfers, Size of ski area, Cost
- Snow quality Cost People Bars / nightlife Travel time
- Cost, the range of slopes for advanced, decent off piste, food, appearance of the town
- Cost, Good range of slopes, food, travel times, size of the town
- Food, Nice Hotels & Chalets,Good variety of slopes, Apres Ski
- nightlife, cost, nice food,
- Cost, a good range for beginners, food, nightlife, good range of other activities
- Cost, Good range of other activities, suitable for families, the slopes and size
- Family, Cost, Range of slopes, stuff to do when not skking
- Cost, snowfall, activities, nightlife

Do you feel it is hard to find out information about resorts on a central website

Yes, usually we just wing it!

Yes, sometimes they do not give enough depth into specific resorts

Yes, some booking sites give information, but its hard to find.

Not really

No, most information is easy to find, however if I want more specific information about a certain resort I would have to find a different site

Yes, it often involves nights of searching through various websites.

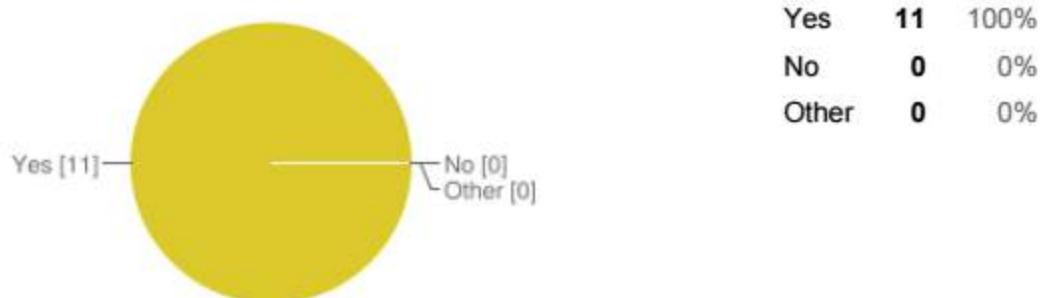
Trip advisor is where it's at

Yes, even booking sites lack information youd like.

Yes because they sometimes don't give enough depth of information about specific places.

Yes

Would you use a website which would help you find this information?



Appendix 4 - Designs.

Resort Search Page

The purpose of this page was so users could easily search and find resorts if they didn't want to use the search function. The data is displayed in tabular form as it is the easiest to read and navigate. From this page users can click on a resort and view it, or add it to their favourites to view at a later date. The two dropdown boxes at the top of the page allow user to either search or filter via country.

Top Lists Page

PISTEE

HOME RESORTS TOP RYAN

Select an option to reveal the top resorts for that criteria.

LES ARCS FRANCE/EUROPE

Slopes	Best For	Worst For
Blacks : 12 Reds : 16 Blues : 23 Greens : 4 Lifts : 46 Size : 263km	Advanced Nightlife Children Snowparks	Children Snowparks

68%

Highest Slope : 3245m
Resort Drop : 1500m
Lowest Slope : 1745m
Longest Slope : 14km

LES ARCS FRANCE/EUROPE

Slopes	Best For	Worst For
Blacks : 12 Reds : 16 Blues : 23 Greens : 4 Lifts : 46 Size : 263km	Advanced Nightlife Children Snowparks	Children Snowparks

68%

Highest Slope : 3245m
Resort Drop : 1500m
Lowest Slope : 1745m
Longest Slope : 14km

LES ARCS FRANCE/EUROPE

Slopes	Best For	Worst For
Blacks : 12 Reds : 16 Blues : 23 Greens : 4 Lifts : 46 Size : 263km	Advanced Nightlife Children Snowparks	Children Snowparks

68%

Highest Slope : 3245m
Resort Drop : 1500m
Lowest Slope : 1745m
Longest Slope : 14km

LES ARCS FRANCE/EUROPE

Slopes	Best For	Worst For
Blacks : 12 Reds : 16 Blues : 23 Greens : 4 Lifts : 46 Size : 263km	Advanced Nightlife Children Snowparks	Children Snowparks

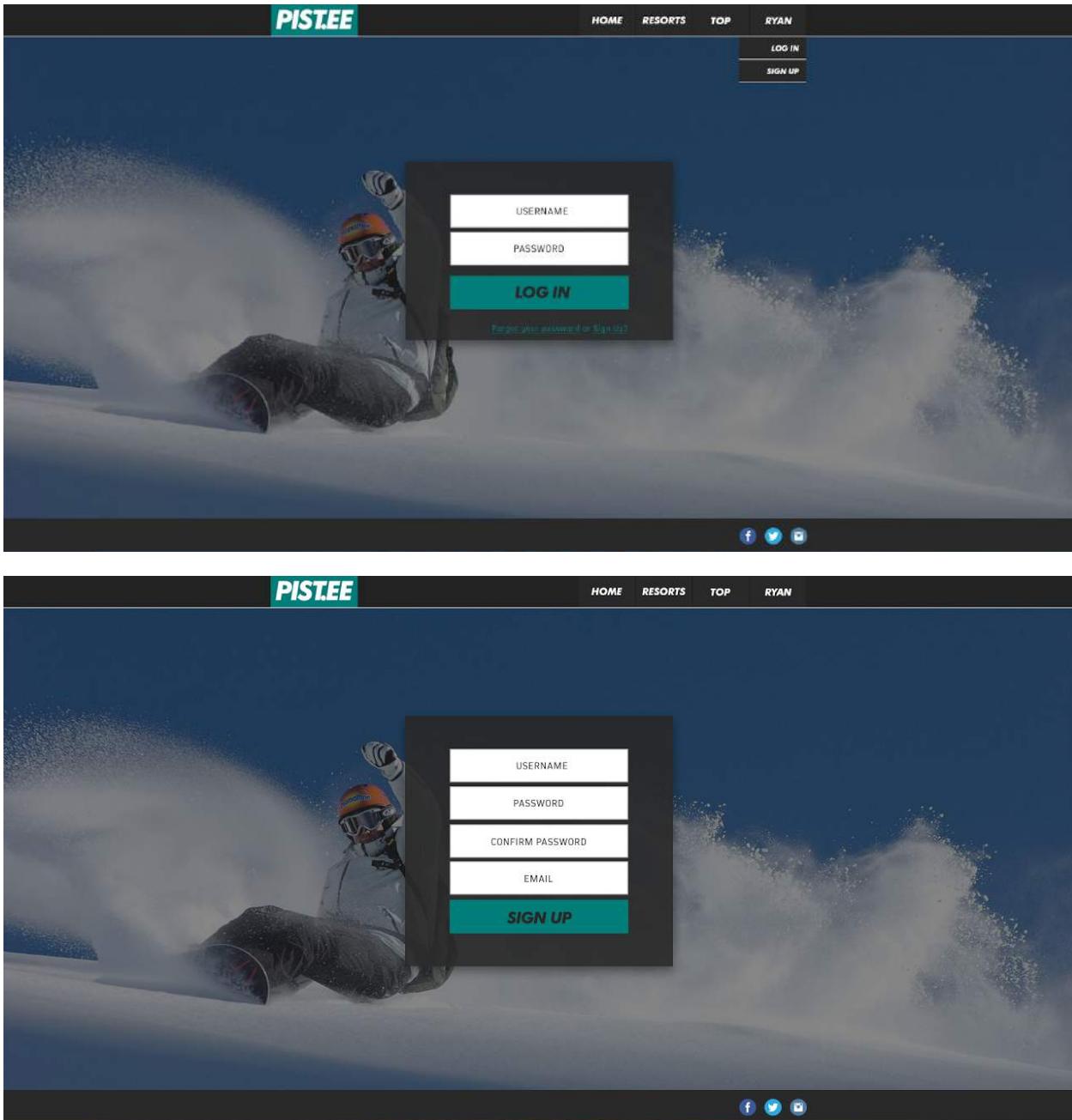
68%

Highest Slope : 3245m
Resort Drop : 1500m
Lowest Slope : 1745m
Longest Slope : 14km

Facebook Twitter LinkedIn

The purpose of this page of the site is so users can pick their most important criteria, and find out which resort is best rated for the criteria. The information for each resort follows a consistent structure across the site. When hovered over the user can click into the full resort page.

Login & Signup page



The user login page had to have a link to the signup page added, as this was added as a functional requirement. The form was also updated with new fonts.

Favourites Page

The screenshot shows a favourites page for the resort 'LES ARCS' (FRANCE/EUROPE). The page has a header with 'PISTEE' and navigation links for 'HOME', 'RESORTS', 'TOP', and 'RYAN'. Below the header, there's a section titled 'FAVOURITES'.

Each entry for 'LES ARCS' includes:

- Logo:** LES ARCS FRANCE/EUROPE
- Slopes:** Black : 12, Red : 16, Blue : 23, Green : 4, Lifts : 46, Size : 203km
- Best For:** Advanced, Nightlife
- Worst For:** Children, Snowpark
- Percentage:** 68% (displayed in a teal circle)
- Mountain Icon:** A stylized mountain peak icon with a trash can icon at the base.
- Resort Statistics:** Highest Slope : 3265m, Resort Drop : 1500m, Lowest Slope : 1745m, Longest Slope : 16km

At the bottom of the page, there are social media sharing icons for Facebook, Twitter, and LinkedIn.

The favourites page follows a similar structure to the rest of the results pages on the site, but has a delete button for the user to be able to remove a resort from their favourites.

Leave Review Page

The screenshot shows the 'Leave A Review' page of the PISTEE website. At the top, there is a dark header bar with the word 'PISTEE' in white. Below the header, the main content area has a light gray background. The title 'LEAVE A REVIEW' is centered at the top of the form. There are two input fields: 'SELECT COUNTRY' and 'SELECT RESORT', both with dropdown menus. To the right of these is a large text area labeled 'Leave a short review (Max 300 characters)'. Below the review area is a teal button labeled 'UPLOAD A PHOTO'. Underneath the photo upload button is another teal button labeled 'SUBMIT'. On the left side of the form, there is a section titled 'Leave your ratings.' containing a grid of 12 circular rating sliders. Each slider is labeled 'COST 4/5'. At the bottom of the page is a dark footer bar with social media icons for Facebook, Twitter, and LinkedIn.

The above page is a new design for the leave review page. The user can select a country then a resort from a drop down menu, then leave a rating using the slider for each of the criteria. A short review can then be left, and a photo uploaded.

Edit Account Page

The screenshot shows a dark-themed website for 'PISTEE'. At the top, there's a navigation bar with links for 'HOME', 'RESORTS', 'TOP', and 'RYAN'. Below the navigation, the main content area has a light gray background. In the center, there's a heading 'EDIT YOUR ACCOUNT' in bold capital letters. Below it is a form with four input fields: 'USERNAME', 'EMAIL', 'PASSWORD', and 'CONFIRM PASSWORD'. A large teal-colored button labeled 'SUBMIT CHANGES' is positioned below the input fields. At the very bottom of the page, there's a dark footer bar with three small social media icons: Facebook, Twitter, and YouTube.

EDIT YOUR ACCOUNT

USERNAME

EMAIL

PASSWORD

CONFIRM PASSWORD

SUBMIT CHANGES

The edit account page was kept relatively simple, allowing the user to change their username, email and password. In the future, more options could be added, but for now these are not necessary, and keeps the simple feel of the site.

Results Page

PISTEE

HOME RESORTS TOP RYAN

LES ARCS
FRANCE/EUROPE

Slopes	Best For
Blacks : 12	Advanced
Reds : 18	Nightlife
Blues : 23	
Greens : 4	
Lifts : 46	
Size : 26.9km	Children
	Snowparks

Best For
Advanced
Nightlife

Worst For
Children
Snowparks



Highest Slope : 3245m
Resort Drop : 1500m
Lowest Slope : 1745m
Longest Slope : 16km



LES ARCS
FRANCE/EUROPE

Slopes	Best For
Blacks : 12	Advanced
Reds : 18	Nightlife
Blues : 23	
Greens : 4	
Lifts : 46	
Size : 26.9km	Children
	Snowparks

SHOW MORE



Highest Slope : 3245m
Resort Drop : 1500m
Lowest Slope : 1745m
Longest Slope : 16km



LES ARCS
FRANCE/EUROPE

Slopes	Best For
Blacks : 12	Advanced
Reds : 18	Nightlife
Blues : 23	
Greens : 4	
Lifts : 46	
Size : 26.9km	Children
	Snowparks

Best For
Advanced
Nightlife

Worst For
Children
Snowparks



Highest Slope : 3245m
Resort Drop : 1500m
Lowest Slope : 1745m
Longest Slope : 16km



LES ARCS
FRANCE/EUROPE

Slopes	Best For
Blacks : 12	Advanced
Reds : 18	Nightlife
Blues : 23	
Greens : 4	
Lifts : 46	
Size : 26.9km	Children
	Snowparks

Best For
Advanced
Nightlife

Worst For
Children
Snowparks



Highest Slope : 3245m
Resort Drop : 1500m
Lowest Slope : 1745m
Longest Slope : 16km

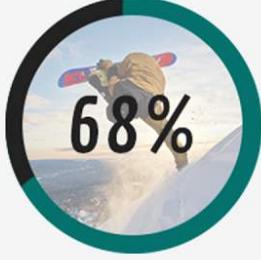
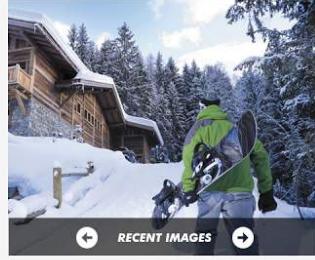


The results page follows the same structure as the favourites page and top lists page to keep the feel of the site consistent across all pages.

Resorts Page

PISTEE

LES ARCS FRANCE ★

KEY STATS

Blacks : 12	Highest Slope : 3245m
Reds : 16	Resort Drop : 1500m
Blue : 23	Lowest Slope : 1745m
Green : 4	Longest Slope : 14km
Lifts : 46	
Size : 263km	

WEATHER

La Plagne Snow Forecast for 2250 m

Outlook	Thu	Fri	Sat	Sun	Mon
Max/Min °C (2)	-4° -6°	-4° -8°	-4° -8°	-1° -8°	-2° -8°
Wind (km/h)	9	10	5	5	4
Freezing Level (m)	3400	3250	3200	3250	3400
Current Weather	7.0°C	calm			

RECENT IMAGES

RATINGS

REVIEWS

WONDERFUL RESORT WITH LOADS TO DO Jan 2015

We were there 3rd week of Jan and used Oxygene for the 10th year running; they have never let us down and always seem to have a good ski school for children. One year I suffered an injury (broke my leg) and they really looked after me. So we keep going back to them. That said I have never heard any complaints about any of the other schools! For fast lunch and good value for money with kids try Croq' Neige it is next to Oxygene ski in Plagne Centre.

WONDERFUL RESORT WITH LOADS TO DO Jan 2015

We were there 3rd week of Jan and used Oxygene for the 10th year running; they have never let us down and always seem to have a good ski school for children. One year I suffered an injury (broke my leg) and they really looked after me. So we keep going back to them. That said I have never heard any complaints about any of the other schools! For fast lunch and good value for money with kids try Croq' Neige it is next to Oxygene ski in Plagne Centre.

WONDERFUL RESORT WITH LOADS TO DO Jan 2015

We were there 3rd week of Jan and used Oxygene for the 10th year running; they have never let us down and always seem to have a good ski school for children. One year I suffered an injury (broke my leg) and they really looked after me. So we keep going back to them. That said I have never heard any complaints about any of the other schools! For fast lunch and good value for money with kids try Croq' Neige it is next to Oxygene ski in Plagne Centre.

PISTEE

[HOME](#) [RESORTS](#) [TOP](#) [RYAN](#)

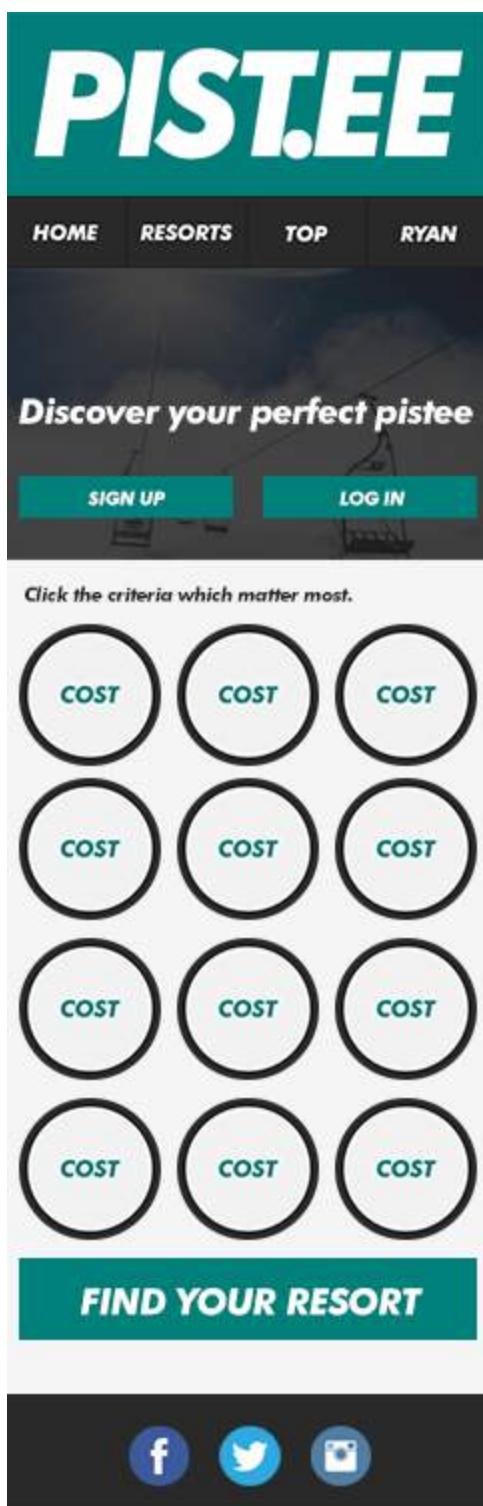
[Facebook](#) [Twitter](#) [Instagram](#)

The resorts page is the page which provides all the information about each resort. This includes the weather which is currently being pulled from snow-forecast.com. This will be updated before launch, as at the moment the weather widget is an i frame which can not be edited. A request has been made to get access to their XML weather feed, allowing me to build and design my own feed, once the data has been received.

Mobile Designs

The next stage of refinement was to produce new mobile designs for each page of the site. These would be activated using media queries when built, so had to be able to follow a similar structure to the mobile site.

Home Page



Resort Search Page

PISTEE

HOME RESORTS TOP RYAN

SEARCH

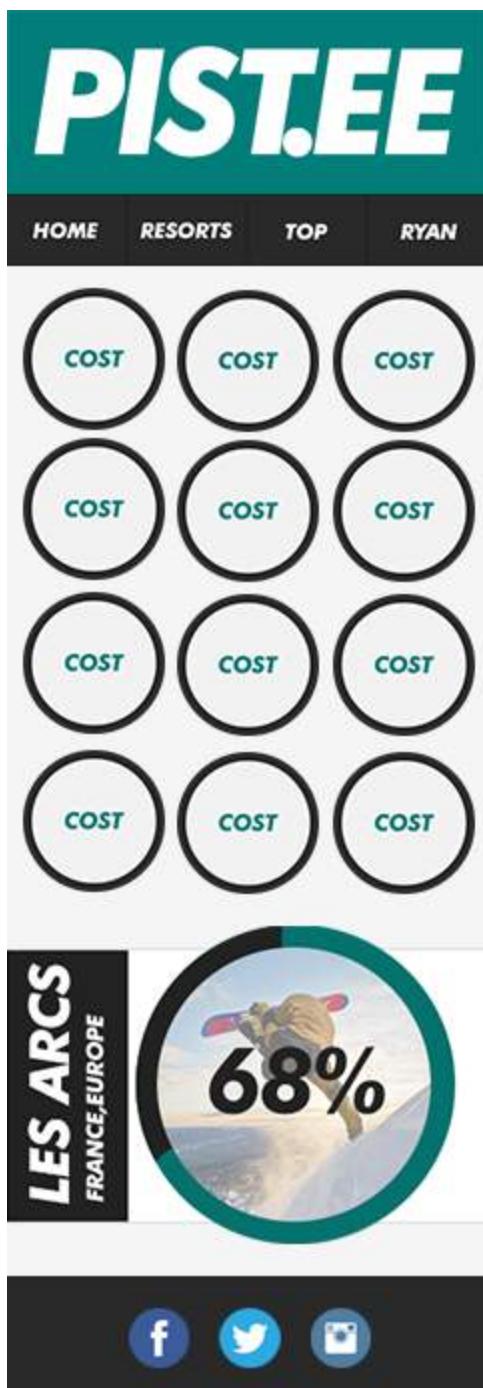
SELCT COUNTRY

Name

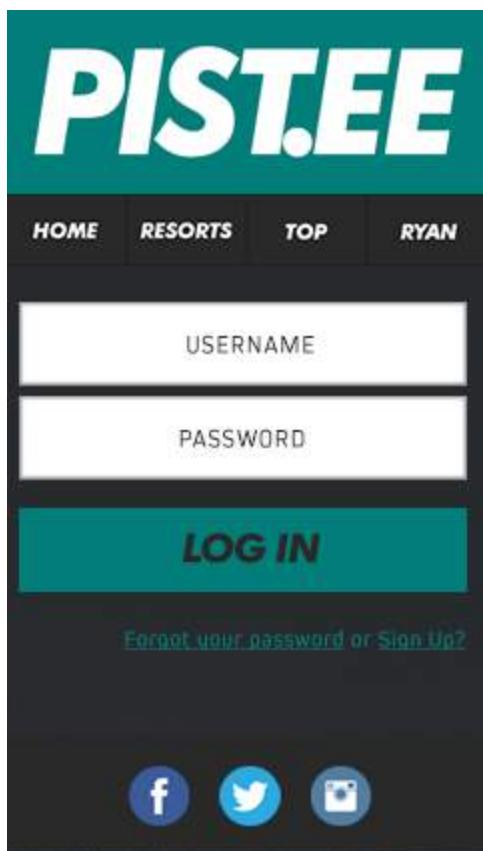
Pas De La Casa

f t i

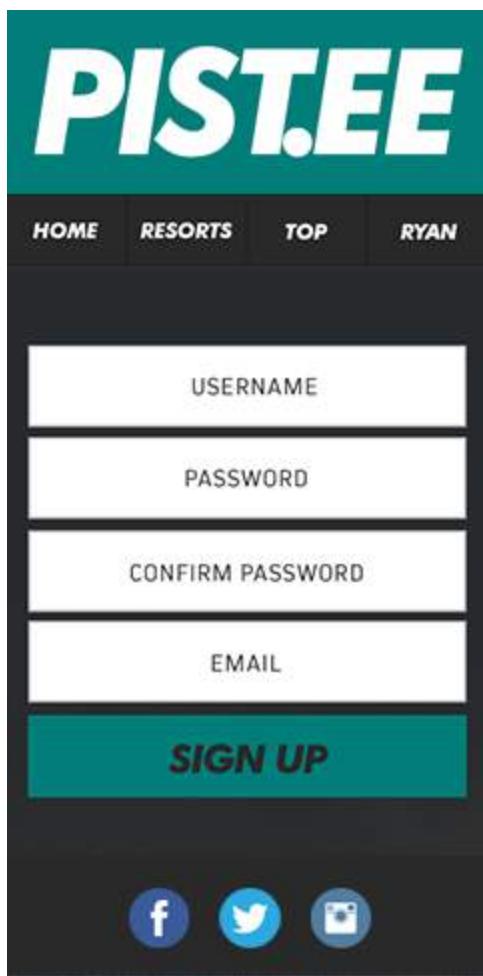
Top Lists Page



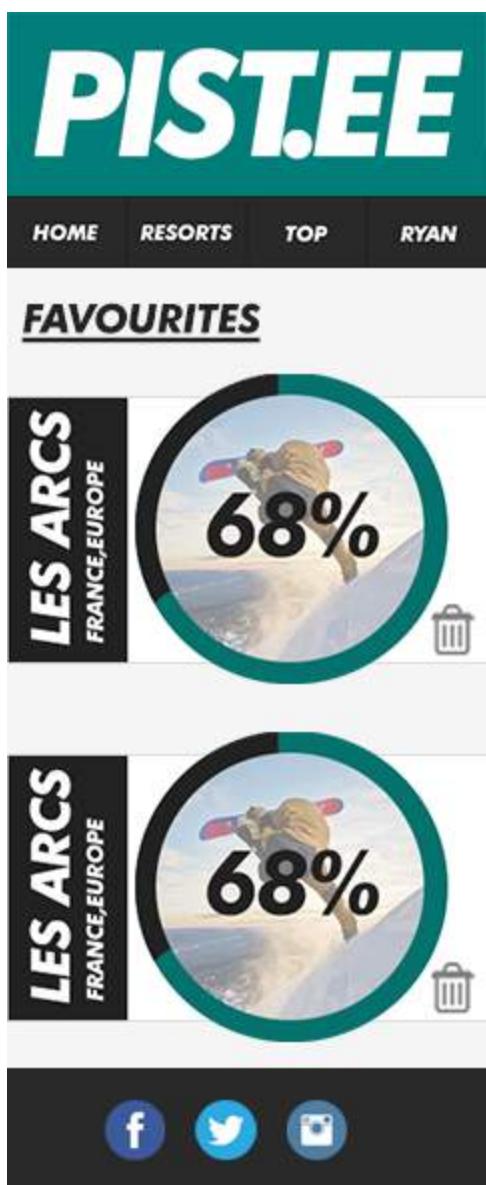
Login Page



Signup Page



Favourites Page



Leave Review Page

PISTEE

HOME RESORTS TOP RYAN

LEAVE A REVIEW

SELECT COUNTRY

SELECT RESORT

Leave your ratings.

COST 4/5	COST 4/5	COST 4/5	COST 4/5
COST 4/5	COST 4/5	COST 4/5	COST 4/5
COST 4/5	COST 4/5	COST 4/5	COST 4/5

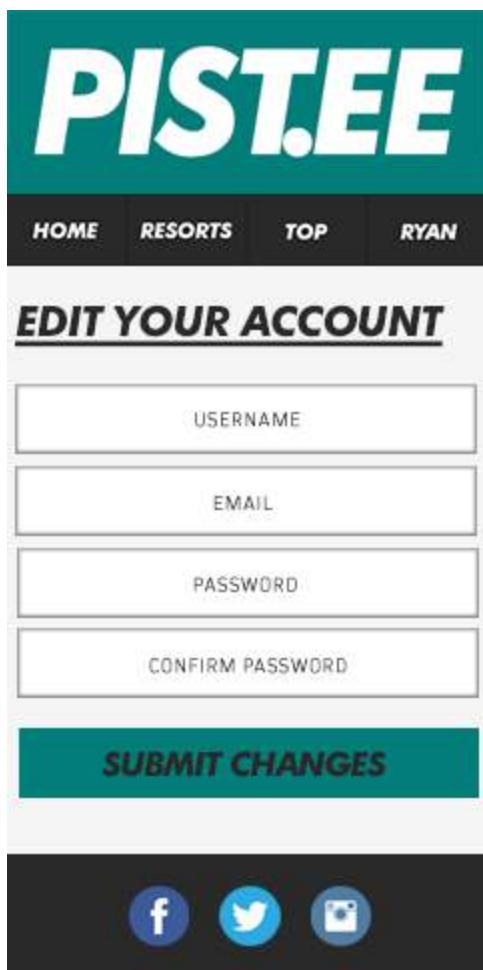
Leave a short review (Max 300 characters)

UPLOAD A PHOTO

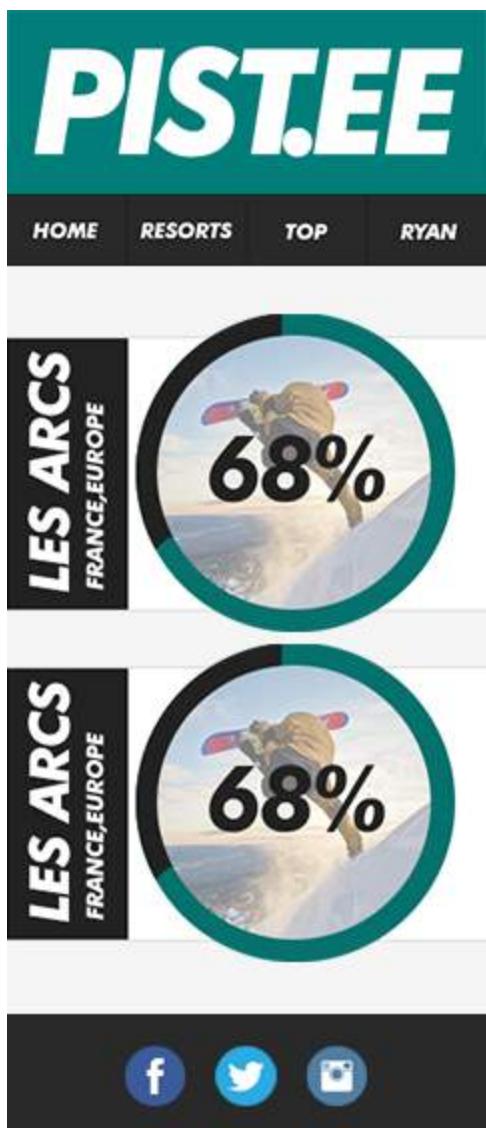
SUBMIT

Edit Account Page



Results Page



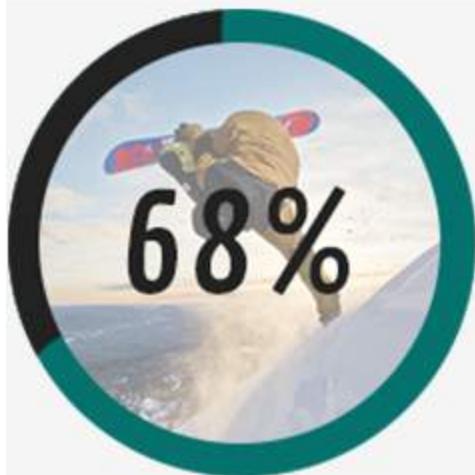
Resorts Page

PIST.EE

HOME RESORTS TOP RYAN

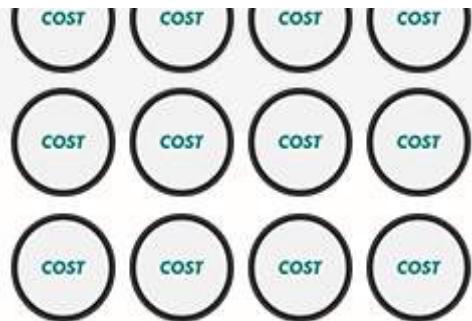
LES ARCS

FRANCE ★



RECENT IMAGES

RATINGS



KEY STATS

Blacks : 12 Highest Slope : 3245m
Reds : 16 Resort Drop : 1500m
Blue : 23 Lowest Slope : 1745m
Green : 4 Longest Slope : 14km
Lifts : 46 Size : 263km

WEATHER



REVIEWS

WONDERFUL RESORT WITH LOADS TO DO - JAN15

We were there 3rd week of Jan and used Oxygene for the 10th year running; they have never let us down and always seem to have a good ski school for children. One year I suffered an injury (broke my leg) and they really looked after me.

WONDERFUL RESORT WITH LOADS TO DO - JAN15

We were there 3rd week of Jan and used Oxygene for the 10th year running; they have never let us down and always seem to have a good ski school for children. One year I suffered an injury (broke my leg) and they really looked after me.

Appendix 5 - Functional Requirement Testing.

Number	1
Type	Functional
Description	Logo
Rationale	The main Logo in the header.
Fit Criterion	The logo will show the main brand of the site, and be a link to the homepage of the site.
Dependencies	None
Expected Result	Logo looks like brand guidelines, and is a link to home when clicked.
Actual Result	Logo looks like brand guidelines, and is a link to home when clicked.
Pass or Fail	PASS

Number	2
Type	Functional
Description	Log in form - Username
Rationale	Allows the user to enter their username.
Fit Criterion	Simple input form with placeholder username. Should work on all devices. Javascript validation so only valid characters can be entered. This then checks username with database, comparing with password
Dependencies	Log in form - Submit button (4)
Expected Result	If name is typed it enters in field. If form is submitted with empty username error comes up.
Actual Result	If name is typed it enters in field. If form is submitted with empty username error comes up.

Pass or Fail	PASS
--------------	------

Number	3
Type	Functional
Description	Log in form - Password
Rationale	Allows the user to enter their Password.
Fit Criterion	Simple input form with placeholder password. This then checks username with database, comparing with password
Dependencies	Log in form - Submit button (4)
Expected Result	If password is typed it enters in field as dots. If form is submitted with empty password error comes up.
Actual Result	If password is typed it enters in field as dots. If form is submitted with empty password error comes up.
Pass or Fail	PASS

Number	4
Type	Functional
Description	Log in form - Submit
Rationale	Allows the user to submit their username/password combo, letting them login to the site.
Fit Criterion	Input button, which sends the username/password to database. If the combination is correct, the user will be logged in, allowing to access the additional pages of the site.
Dependencies	favourites page, change details page, leave a review page.
Expected Result	If correct user will be logged in, if not notification to say login failed.

Actual Result	If correct user will be logged in, if not notification to say login failed.
Pass or Fail	PASS

Number	5
Type	Functional
Description	Sign up form - Username
Rationale	Allows the user to enter username, so they have login details for the site.
Fit Criterion	Simple input with javascript validation to check if username is available, and if characters are correct.
Dependencies	Submit form - Submit button (10)
Expected Result	If name is typed it enters in field. If form is submitted with empty username error comes up.
Actual Result	If name is typed it enters in field. If form is submitted with empty username error comes up.
Pass or Fail	Pass

Number	6
Type	Functional
Description	Sign up form password
Rationale	Allows the user to enter their desired Password, to sign into the site.
Fit Criterion	Javascript validation will be used to make sure password is 8 characters long, and uses letters and numbers.
Dependencies	Submit form- Submit button (10), confirm-password (7)
Expected Result	If Password is typed it enters in field as dots. If form is submitted

	with empty password error comes up.
Actual Result	If Password is typed it enters in field as dots. If form is submitted with empty password error comes up.
Pass or Fail	Pass

Number	7
Type	Functional
Description	Sign up form confirm password
Rationale	Allows the user to enter their desired Password again, so the password is definitely correct.
Fit Criterion	Javascript validation will be used to make sure password matches the previous password.
Dependencies	Submit form- Submit button (10)
Expected Result	If Password is typed it enters in field as dots. If Password is not included login should fail.
Actual Result	This field does not exist.
Pass or Fail	Fail

Number	8
Type	Functional
Description	Sign up form email
Rationale	Allows the user to enter their email, so they can signup.
Fit Criterion	Simple input form with email placeholder. This then checks email with database, to make sure person hasn't already signed up. Javascript validation will be used to make sure email is correct format.

Dependencies	Submit form- Submit button (10)
Expected Result	If email is typed it enters in field. If no email is submitted or an invalid one signup should fail
Actual Result	If email is typed it enters in field. If no email is submitted or an invalid one signup should fail
Pass or Fail	Pass

Number	9
Type	Functional
Description	Sign up with facebook
Rationale	Allows the user to use their facebook details to sign up, to save time.
Fit Criterion	User can connect with their facebook account to use as a signup method, to help save time.
Dependencies	Submit form- Submit button (10)
Expected Result	Should fetch details from users facebook.
Actual Result	Option does not exist
Pass or Fail	Fail

Number	10
Type	Functional
Description	Sign up form submit
Rationale	Allows the user submit their details and make a user account.
Fit Criterion	Submit button adds details to database and makes an account. User will automatically be logged into the site.

Dependencies	Submit form- Submit button (10)
Expected Result	User is added, and redirected to login to login
Actual Result	User is added, and redirected to login to login
Pass or Fail	Pass

Number	11
Type	Functional
Description	Primary Menu - Home Button
Rationale	A button to link to the home page, allowing users to easily navigate back to the main page of the site.
Fit Criterion	The button should look like a button, with the text easy to read on all devices.
Dependencies	Home Page
Expected Result	If button is clicked it links to homepage correctly
Actual Result	If button is clicked it links to homepage correctly
Pass or Fail	Pass

Number	12
Type	Functional
Description	Primary Menu - Top lists Button
Rationale	A button to link to the top lists page.
Fit Criterion	The button should look like a button, with the text easy to read on all devices.
Dependencies	Top lists page.
Expected Result	If button is clicked it links to top lists correctly
Actual Result	If button is clicked it links to top lists correctly

Pass or Fail	Pass
--------------	------

Number	13
Type	Functional
Description	Primary menu - Resorts button
Rationale	A button to link to the resorts page.
Fit Criterion	The button should look like a button, with the text easy to read on all devices.
Dependencies	Resorts page
Expected Result	If button is clicked it links to resorts correctly
Actual Result	If button is clicked it links to resorts correctly
Pass or Fail	Pass

Number	14
Type	Functional
Description	Primary Menu - User button
Rationale	To link to the secondary drop down menu for the user
Fit Criterion	The button should look like a button, with the text easy to read on all devices.
Dependencies	Dropdown menu (15-18)
Expected Result	If user login in read their username, if not read 'user'. When hovered, secondary menu should display
Actual Result	If user login in read their username, if not read 'user'. When hovered, secondary menu should display
Pass or Fail	Pass

Number	15
Type	Functional
Description	Secondary menu - Log in button
Rationale	Takes the user to a login page, so they can have the ability to leave a review and add favourites.
Fit Criterion	Be part of drop down menu, and link to the login page.
Dependencies	Login page
Expected Result	only exists if user is not signed in. Takes user to login page
Actual Result	only exists if user is not signed in. Takes user to login page
Pass or Fail	Pass

Number	16
Type	Functional
Description	Secondary menu- Sign up button
Rationale	Takes user to a signup page/form, allowing them to make an account
Fit Criterion	Be part of dropdown menu, and link to the signup page.
Dependencies	Sign up page.
Expected Result	only exists if user is not signed in. Takes user to Signup page
Actual Result	only exists if user is not signed in.Takes user to Signup page
Pass or Fail	Pass

Number	17
Type	Functional
Description	Secondary menu - Review button
Rationale	Takes user to a review page. If not logged in takes them to login page.
Fit Criterion	Be part of a dropdown menu, and links to review page
Dependencies	Review page
Expected Result	Only exists if user is signed in. Takes user to leave review page
Actual Result	Only exists if user is signed in. Takes user to leave review page
Pass or Fail	Pass

Number	18
Type	Functional
Description	Favourites button
Rationale	Takes user to their favourites page. If not logged in takes them to login page.
Fit Criterion	Be part of dropdown menu, and links to favourites page.
Dependencies	Favourites page
Expected Result	Only exists if user is signed in. Takes user to their favourites page
Actual Result	Only exists if user is signed in. Takes user to their favourites page
Pass or Fail	PASS

Number	19
Type	Functional
Description	Secondary menu - Edit account button
Rationale	takes user to the edit account page. Only appears when logged in.
Fit Criterion	Be part of dropdown menu, and links to update account page.
Dependencies	Update account page.
Expected Result	Only exists if user is signed in. Takes user to their account page
Actual Result	Only exists if user is signed in. Takes user to their account page
Pass or Fail	PASS

Number	19-A
Type	Functional
Description	Secondary menu - Logout Button
Rationale	Logs user out and redirects to homepage
Fit Criterion	Be part of dropdown menu, and links to update account page.
Dependencies	User session logs out.
Expected Result	Only exists if user is signed in. Takes user to homepage and ends session.
Actual Result	Only exists if user is signed in. Takes user to homepage and ends session.
Pass or Fail	PASS

Number	20
Type	Functional
Description	Find Resorts Page - Image Gallery
Rationale	Allows the main page to have a gallery, which could possibly be used for advertisements, or for general photos
Fit Criterion	This will be full width of screen, with an automated transition. Text/links will be able to placed in front of each image.
Dependencies	none
Expected Result	Image gallery to function with no bugs
Actual Result	Image gallery to function with no bugs
Pass or Fail	Pass

Number	21
Type	Functional
Description	Find Resorts Page - Select buttons
Rationale	Allow the user to select up to twelve criteria which are important for them.
Fit Criterion	Each button will have an image relating to the item, or a circular design with text. when selected, they will be highlighted.
Dependencies	Results page.
Expected Result	Button can be selected and deselected. between 0-12 can be selected.
Actual Result	Button can be selected and deselected. between 0-12 can be selected.
Pass or Fail	Pass - Design has changed.

Number	22
Type	Functional
Description	Find Resorts Page - Submit button
Rationale	Submits the users request, and queries the database to find the ideal resorts in order of best rating.
Fit Criterion	Button will look like a button, and match the style of the site.
Dependencies	Results page.
Expected Result	Button performs as expected, and when clicked takes user to results page with ratings.
Actual Result	Button performs as expected, and when clicked takes user to results page with ratings.
Pass or Fail	Pass

Number	23
Type	Functional
Description	Footer
Rationale	Footer of page will include primary menu, secondary menu and drop down menu as simple links, for ease of navigation
Fit Criterion	Links will have a simple style, such as underlined text.
Dependencies	Pages linked too.
Expected Result	Social media links can all be clicked
Actual Result	Social media links can all be clicked. Social media accounts dont exist.
Pass or Fail	Pass - No longer includes links due to styling change

Number	24
Type	Functional
Description	Results Page - Main layout
Rationale	The main way the results will be displayed in the page, so the user can easily navigate them
Fit Criterion	The results will be displayed in a single column page, with each result having its own section displaying brief information about the resort.
Dependencies	Key info for inside each resort, resorts page from link.
Expected Result	All resort info displayed, and the whole sections are clickable links.
Actual Result	All resort info displayed, and the whole sections are clickable links.
Pass or Fail	Pass

Number	25
Type	Functional
Description	Results Page - show more button
Rationale	Allows more results to be shown
Fit Criterion	Minimalist button, like an arrow.
Dependencies	Extra Results
Expected Result	Show more button shows more results.
Actual Result	Button does not exist.
Pass or Fail	Fail

Number	26
Type	Functional
Description	Results Page - key information - name
Rationale	The name displayed for each result.
Fit Criterion	Large name inside the section for each result.
Dependencies	None
Expected Result	Name displays in correct area with correct styling
Actual Result	Name displays in correct area with correct styling
Pass or Fail	Pass

Number	27
Type	Functional
Description	Results Page - rating
Rationale	So the user knows how suited the resort is to them as a percentage
Fit Criterion	Using a circular HTML5 Progress bar, with the percentage displayed in the middle.
Dependencies	None.
Expected Result	Average displayed wrapped in progress bar
Actual Result	Average displayed wrapped in progress bar
Pass or Fail	Pass

Number	28
Type	Functional
Description	Results Page -key information - Mountain Stats
Rationale	So the user gets a slight insight into the resort, before clicking into the full page
Fit Criterion	Be easy to read, with a mountain graphic for resort heights
Dependencies	none
Expected Result	Key info displayed in list
Actual Result	Key info displayed in list
Pass or Fail	Pass

Number	29
Type	Functional
Description	Results Page - key information-best for section
Rationale	Informs the user of the 3 highest ratings out of 12 for the resort.
Fit Criterion	Follow similar design to key info.
Dependencies	none.
Expected Result	Key information section includes 3 highest ratings
Actual Result	Does not exist.
Pass or Fail	Fail - Due to design change

Number	30
Type	Functional
Description	Resorts Page - review ratings
Rationale	Informs the user of each of the ratings for the 12 criteria of the site, allowing them to decide if the resort is suited to them.
Fit Criterion	Use Circular HTML 5 progress bars with the name below, and rating in the middle.
Dependencies	none.
Expected Result	Ratings displayed on page.
Actual Result	Ratings displayed on page.
Pass or Fail	Pass

Number	31
Type	Functional
Description	Resorts Page - key mountain stats
Rationale	Informs the user of the key stats about the mountain.
Fit Criterion	Be in a list form, with icons.
Dependencies	none
Expected Result	Key mountain stats displayed
Actual Result	Key mountain stats displayed
Pass or Fail	Pass

Number	32
Type	Functional
Description	Resorts Page - result ranking
Rationale	How the result is ranked against other resorts. Ie 14/26 in france.
Fit Criterion	Easy to read, and inline with design
Dependencies	none
Expected Result	Result ranking shown for country.
Actual Result	Result ranking doesn't exist.
Pass or Fail	Fail

Number	33
Type	Functional
Description	Resorts Page - written reviews
Rationale	Written reviews of the resort
Fit Criterion	Single column of reviews, giving name of reviewer, date, and the review.
Dependencies	none.
Expected Result	Written reviews exist with all info.
Actual Result	Name no longer included due to design but everything else included.
Pass or Fail	Pass

Number	34
Type	Functional
Description	Resorts Page - weather
Rationale	Allows users to view current weather for resort
Fit Criterion	Be displayed as tabular data, so its easily read.
Dependencies	none.
Expected Result	Weather info to be displayed for resort
Actual Result	Weather info to be displayed for resort. not tabular
Pass or Fail	PAss

Number	35
Type	Functional
Description	Resorts Page - images
Rationale	An image gallery of images left by reviewers.
Fit Criterion	small image gallery, which expands when clicked.
Dependencies	none.
Expected Result	Image gallery with photos from that resort.
Actual Result	Image gallery with photos from that resort.
Pass or Fail	Pass

Number	36
Type	Functional
Description	Leave a review - Resort Select
Rationale	Allows the user to select what resort they are reviewing.
Fit Criterion	A select box which allows a user to select a country, followed by a resort name.
Dependencies	Reviews
Expected Result	Country select box selects country, resort select box selects resort inside that country.
Actual Result	Country select box selects country, resort select box selects resort inside that country.
Pass or Fail	Pass

Number	37
Type	Functional
Description	Leave a review - Rating Sliders
Rationale	Allows the user review the resort using criteria. The review will be from the number 1-5
Fit Criterion	The sliders will be circular, and user can drag round to select. Not all selections have to be completed.
Dependencies	Reviews
Expected Result	Rating sliders allow ratings from 1-5 for all 12 resorts
Actual Result	Rating sliders allow ratings from 1-5 for all 12 resorts
Pass or Fail	Pass

Number	38
Type	Functional
Description	Leave a review - photo upload
Rationale	Allows the user to upload an image from their holiday, so other users can see what it would be like
Fit Criterion	The upload button should be clearly a button for photo upload, and a notification should be made when upload is complete.
Dependencies	Reviews
Expected Result	Photo can be uploaded. Then shown in correct resort
Actual Result	Photo can be uploaded. Then shown in correct resort
Pass or Fail	Pass

Number	39
Type	Functional
Description	Leave a review - written review
Rationale	Allow the user to leave a 200 character review about the resort.
Fit Criterion	Short text box with 200 character count down. Idea comes from the idea of short feedback like eBay use, so it is quick and easy to complete.
Dependencies	Reviews.
Expected Result	Written review with max character input of 200
Actual Result	Written review but no max input.
Pass or Fail	Fail

Number	40
Type	Functional
Description	Leave a review - Submit Button
Rationale	Submits review to database.
Fit Criterion	Standard button matching style of site.
Dependencies	Reviews table in database.
Expected Result	Submits review to DB. User redirected to home page.
Actual Result	Submits review to DB. User redirected to home page.
Pass or Fail	Pass

Number	41
Type	Functional
Description	User Favourites - Main layout
Rationale	The main way the results will be displayed in the page, so the user can easily navigate them
Fit Criterion	The results will be displayed in a single column page, with each result having its own section displaying brief information about the resort.
Dependencies	Key info for inside each resort, resorts page from link.
Expected Result	Results displayed as expected, and clickable as links
Actual Result	Results displayed as expected, and clickable as links
Pass or Fail	PASS

Number	42
Type	Functional
Description	User Favourites - key information - name
Rationale	The name displayed for each result.
Fit Criterion	Large name inside the section for each result.
Dependencies	None
Expected Result	Name is displayed and correct
Actual Result	Name is displayed and correct
Pass or Fail	Pass

Number	43
Type	Functional
Description	User Favourites - rating
Rationale	So the user knows how suited the resort is to them as a percentage
Fit Criterion	Using a circular HTML5 Progress bar, with the percentage displayed in the middle.
Dependencies	None.
Expected Result	Resort ratings displayed with progress bar. Results displayed in descending order.
Actual Result	Resort ratings displayed with progress bar. Results displayed in descending order.
Pass or Fail	Pass

Number	44
Type	Functional
Description	User Favouritese -key information - Mountain Stats
Rationale	So the user gets a slight insight into the resort, before clicking into the full page
Fit Criterion	Be easy to read, with a mountain graphic for resort heights
Dependencies	none
Expected Result	Key Information is displayed and correct
Actual Result	Key Information is displayed and correct
Pass or Fail	Pass

Number	45
Type	Functional
Description	User Favourites - key information-best for section
Rationale	Informs the user of the 3 highest ratings out of 12 for the resort.
Fit Criterion	Follow similar design to key info.
Dependencies	none.
Expected Result	top 3 ratings are displayed and correct
Actual Result	Ratings not displayed due to change in design
Pass or Fail	Fail - due to design change.

Number	46
Type	Functional
Description	User Favourites - delete button
Rationale	Allows the user to delete the favourite.
Fit Criterion	Small button with trash can icon for usability
Dependencies	favourites page.
Expected Result	User delete button exists and deletes favourite.
Actual Result	User delete button exists and deletes favourite.
Pass or Fail	Pass

Number	47
Type	Functional
Description	User Favourites - compare.
Rationale	Allows user to compare their favourites.
Fit Criterion	Small button with compare icon/word
Dependencies	none.
Expected Result	User favourite can be compared to another favourite.
Actual Result	Function does not exist
Pass or Fail	Fail

Number	48
Type	Functional
Description	Top Lists Page - Submenu
Rationale	Allows the user to select which of the 12 toplists to view.

Fit Criterion	Uses icons consistent with homepage. Loads list via ajax.
Dependencies	Lists.
Expected Result	12 options can be selected. Changes how the results are displayed depending on rating.
Actual Result	12 options can be selected. Changes how the results are displayed depending on rating.
Pass or Fail	Pass

Number	49
Type	Functional
Description	Top Lists - key information - name
Rationale	The name displayed for each result.
Fit Criterion	Large name inside the section for each result.
Dependencies	None
Expected Result	Name should be displayed correctly.
Actual Result	Name should be displayed correctly.
Pass or Fail	Pass

Number	50
Type	Functional
Description	Top Lists - rating

Rationale	So the user knows how suited the resort is to them as a percentage
Fit Criterion	Using a circular HTML5 Progress bar, with the percentage displayed in the middle.
Dependencies	None.
Expected Result	Rating displayed as expected.
Actual Result	Rating displayed as expected.
Pass or Fail	Pass

Number	51
Type	Functional
Description	Top Lists -key information - Mountain Stats
Rationale	So the user gets a slight insight into the resort, before clicking into the full page
Fit Criterion	Be easy to read, with a mountain graphic for resort heights
Dependencies	none
Expected Result	Key mountain statistics displayed
Actual Result	Key mountain statistics displayed
Pass or Fail	Pass

Number	52
Type	Functional
Description	Top Lists - key information-best for section
Rationale	Informs the user of the 3 highest ratings out of 12 for the resort.
Fit Criterion	Follow similar design to key info.
Dependencies	none.
Expected Result	3 highest rated categories display.
Actual Result	3 highest rated categories display.
Pass or Fail	Pass

Number	53
Type	Functional
Description	Resort Search Page - Search
Rationale	Allows users to select resorts using countries, so they can view all resorts in one country
Fit Criterion	Simple search box with select options for countries.
Dependencies	resort search table
Expected Result	Select box for countrys, search box to search
Actual Result	No select box but search box can also search countrys
Pass or Fail	Pass

Number	54
Type	Functional
Description	Resort Search Page - Name
Rationale	Name of resort table column

Fit Criterion	Column of table with resort name
Dependencies	none.
Expected Result	Name of resort with link.
Actual Result	Name of resort with link.
Pass or Fail	Pass

Number	55
Type	Functional
Description	Resort Search Page - Height
Rationale	Resort height data so user knows heights of mountains.
Fit Criterion	Column of table with lowest and highest heights plus difference.
Dependencies	none.
Expected Result	Max height in resort included in table.
Actual Result	Max height in resort included in table.
Pass or Fail	Pass

Number	56
Type	Functional
Description	Resort Search Page - Size
Rationale	Size of resort table column
Fit Criterion	Column of table with number of KM of pistes, Number of runs, number of lifts.
Dependencies	none.
Expected Result	Number of lifts, slopes and size included.
Actual Result	Number of lifts, slopes and size included.
Pass or Fail	Pass

Number	57
Type	Functional
Description	Resort Search Page - rating
Rationale	Rating of resort table column
Fit Criterion	Column of table with resort rating
Dependencies	none.
Expected Result	Column with rating.
Actual Result	Rating not present. Has been replaced with button to add favourites.
Pass or Fail	Pass

Number	58
Type	Functional
Description	Resort Search Page - Temperature
Rationale	Current temperature of resort table column
Fit Criterion	Column of table with resort temperature
Dependencies	none.
Expected Result	Temperature displayed.
Actual Result	Temperature no longer displayed
Pass or Fail	Pass - Due to room requirements in table.

Number	59
Type	Functional
Description	Resort Search Page - snow conditions
Rationale	Snow conditions table column
Fit Criterion	Column of table with snow depths at low/high points
Dependencies	none.
Expected Result	Snow conditions in table
Actual Result	Snow conditions not available
Pass or Fail	Fail - Due to room requirements in table. and getting snow conditions for each resort wasnt applicable

Number	60
Type	Functional
Description	Admin Page - Menu
Rationale	Menu with all links to Admin pages.
Fit Criterion	Allows the admin user to easily navigate through pages. Menu will be a sidebar, with links in a vertical list.
Dependencies	Admin pages
Expected Result	Menu has links to all admin pages
Actual Result	Menu has links to all admin pages
Pass or Fail	Pass

Number	61
Type	Functional
Description	Admin Page - Approve reviews
Rationale	Allows admin to approve reviews
Fit Criterion	Allows the admin to view reviews, and tick a box to approve them.
Dependencies	reviews.
Expected Result	Reviews are in table marked as unapproved until approved.
Actual Result	Reviews are in table marked as unapproved until approved.
Pass or Fail	Pass

Number	62
Type	Functional
Description	Admin Page - Edit users
Rationale	A page where admin can search for a user and edit their details/delete accounts.
Fit Criterion	Allows the admin user to easily navigate through pages. Menu will be a sidebar, with links in a vertical list.
Dependencies	Admin pages
Expected Result	User page has links to edit/delete user
Actual Result	User page has links to edit/delete user
Pass or Fail	Pass

Number	63
Type	Functional
Description	Admin Page - Edit reviews
Rationale	Allows admin to edit & delete reviews.
Fit Criterion	Allows the admin to edit reviews.
Dependencies	Reviews
Expected Result	Admin can approve, edit and delete reviews
Actual Result	Can approve and delete, but not edit.
Pass or Fail	Pass

Number	64
Type	Functional
Description	Admin Page - Edit Resorts
Rationale	Allows the user to select a resort, and make changes to it.
Fit Criterion	Easy to use interface, with resorts easily selected.
Dependencies	Resort pages.
Expected Result	Resorts can be edited and added/deleted.
Actual Result	Resorts can be edited and added/deleted.
Pass or Fail	Pass

Number	65
Type	Functional
Description	Admin Page - Edit Image sliders
Rationale	Allows the image sliders and the text associated with them to be easily edited.
Fit Criterion	Allows the user to upload new images, and change descriptions.
Dependencies	Admin pages
Expected Result	Images on homepage can be edited from here.
Actual Result	Feature does not exist
Pass or Fail	Fail

Appendix 6 - User testing example form.

Pist.ee - Your thoughts.

A short review to give feedback on the website.

*Required

Name *

Age *

Occupation *
If student state Degree Name

How computer literate would you describe yourself as? *

Above Average
 Average
 Below Average

What were your initial thoughts on the appearance of the site? *
Be truthful, and keep brief.

Did anything break/not work as expected?

What didn't you like about the site?

Would you say the website is easier to use than other travel websites you have used?

State Site if Possible

Would you class the website as useful?

- Yes
- No
- If I was interested in Skiing/Snowboarding

Submit

Never submit passwords through Google Forms.

 100%: You made it.

Powered by
 Google Forms

This content is neither created nor endorsed by Google.
[Report Abuse](#) - [Terms of Service](#) - [Additional Terms](#)

Appendix 7 - User testing Result Summary.

Summary

Name

Chris McCoppin
Ben Hunter
Lyndsey
Brian Wilson
Stephen Martin
Lesley Wilson

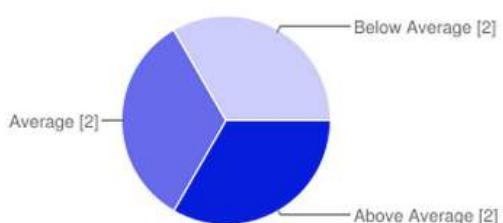
Age

22
23
24
19
55
54

Occupation

Interactive Multimedia Design
Sports and Exercise Science Student
Accounting
Machine Operator - Schrader Electronics
Mooring Master
Retired Secetary

How computer literate would you describe yourself as?



Above Average	2	33.3%
Average	2	33.3%
Below Average	2	33.3%

What were your initial thoughts on the appearance of the site?

Colourful and simple. Easy to find information.

It looked nice, and was easy to use.

Simple and easy to use. No cluttered adds etc.

Very appealing, easy on the eyes.

It has nice colours and layout.

The site was on brand, clean and visually impressive. The colours used, grid layout and theme of the site were consistent, usable and bold.

Did anything break/not work as expected?

When signing up it would be nice to not have to log back in.

No

Some of the ratings said 0%. Apart from that no.

No.

Everything worked perfectly.

Everything worked as expected.

What didn't you like about the site?

It would be nice to be able to compare favourites.

Some sections were unclear as to what certain choices were regarding. Capital letters at login and email entry threw me off slightly.

I didn't like the consistent use of capitals throughout the site, it didn't seem appropriate for all areas such as forms.

Capital letters in forms were confusing.

When signing up and logging in, you got logged back out and had to sign in again.

Would you say the website is easier to use than other travel websites you have used?

Yes, directski which i've booked with before is very confusing.

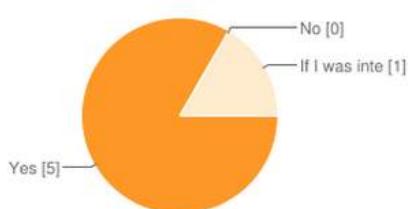
The website was a lot easier to use than other websites I've used such as, www.thomsonski.co.uk. The content was easy to find and appropriate, there was no unnecessary jargon, advertisements or external messages. Everything was neat, contained and useful.

Trip Advisor can be very confusing, but it does have more options

Yes, for finding out info it is a very efficient way of doing so.

Most of the websites for travel I would use let you book holidays, or link to places which you can book them.

Yes. Very straightforward and down to the point. Each step was progressive.

Would you class the website as useful?

Yes	5	83.3%
No	0	0%
If I was interested in Skiing/Snowboarding	1	16.7%

