INTRODUCTION

Our project is an innovative travel website designed to provide an easy travel planning experience for the user. It aims to alleviate users from the burdens of exhaustive research and planning fatigue, filling the needs of busy holiday goers who don't have time to plan in detail.

The primary target audience includes busy individuals or parents of younger children/early teens, first time travellers, specifically 18-22 year olds, and couples ages 22-35 seeking a romantic getaway, giving users the option to select which category they fall under.

At its core, our travel website offers a range of features to users. Users can expect recommendations for city breaks, activities, attractions and restaurants. Our website will make use of APIs to retrieve restaurant recommendations, taking an input city from the user, and returning restaurant suggestions for the user. The recommendations for activities and attractions include a map, enabling the user to easily see where each activity is in relation to each other. The resources page includes links to useful resources. For example, there will be links to important information such as Booking.com and Skyscanner. As foreign currency is needed for travels, our website also includes a currency converter web app so users can easily access foreign exchange rates.

The rest of the report will provide further details on what the project achieved and our processes for creating the project.

ROADMAP OF THE REPORT:

BACKGROUND
The background will discuss the specific details of our project, delving into the requirements for our travel website.

SPECIFICATIONS AND DESIGN
Specifications and design will discuss how we designed our project, looking at the Figma high-fidelity wireframes and the decisions behind our design. This chapter will discuss the architecture of our website, and user navigation. Finally, it will look into the requirements for running our website and for those we created the website for.

IMPLEMENTATION AND EXECUTION
The Implementation and Execution chapter will cover how the team went about creating the website and any potential challenges we faced, covering our approach and the tools we use. This chapter includes development approach and team member roles, tools and libraries, the implementation process, agile development and implementation challenges.

TESTING AND EVALUATION
In the Testing and Evaluation chapter, the report will explore any testing the team carried out, pointing out any limitations this may have uncovered, as well as our testing strategy.

CONCLUSION
As we reach the conclusion, the report will offer any final closing statements, summarising what has been discussed throughout the report.

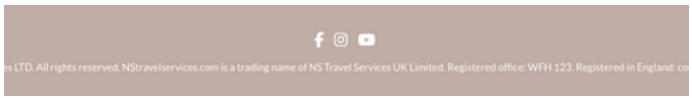
BACKGROUND

As a team, we chose this topic because we all love to travel, and it is something that we are all interested in. This chosen topic of travel helps people with passports enjoy their love of travelling. So that is where we come in and think of things you need to plan a holiday. Our website features places to eat, sites to visit, an itinerary, and helpful resources like a currency converter for the travelling nomad.

So, let us take a look at how it all works:
Our files are organized using folders, including components, pages and contexts. The website contains six main pages. For users to navigate around the website, we have implemented a navigation bar as a component on each page. As well as a footer for the bottom of the page. Show in these diagrams:

Diagram 1.1 NS Travel Services Header

Diagram 1.2 NS Travel Services Footer



Home: Let's start with the home page, which contains information and links to the city breaks pages; it also includes a photo carousel showcasing the sightseeing options, which are later shown on the sightseeing page. The last section on this page is a reviews section, which showcases how this website is helping improve their travel experiences.

City Breaks: The next page from our navigation bar is city breaks, containing image cards with different holidays, including Romantic Getaways, Family breaks, Independent Travel and Our Cities.

At the bottom of the image card is a 'Read More' button, which takes you to that page. These pages feature an image carousel and reviews on the right-hand side.

Sightseeing: The third main page is Sightseeing, which focuses on and contains sightseeing options from the two cities from our romantic getaways page. A quick description of the top sites in Paris and Edinburgh is followed by an interactive map using the zoom icon and displayed markers correlating to the sites described above.

Resources: The travel resources page includes a short description and four buttons linking to helpful travel websites: Been Travel Map, Booking.com, GOV: UK and Skyscanner. When you press those buttons, it takes you to the corresponding websites. Then below is a currency converter, automatically set to a hundred pounds to convert to Euros. But it is customizable to whatever the user needs to convert, with all the currencies and inputting the amount of money.

Restaurants: The next page on the navigation bar is restaurants, which uses an API to search for a city and generate restaurants in that city. It showcases a description of what type of restaurant it is, an image and a button that, when clicked on, takes you to that restaurant's website.

Itinerary: The final page of NS Travel Services is the itinerary page, which allows the user to generate a downloaded PDF itinerary by inputting your departure, destination, dates, duration, and any activities planned.

SPECIFICATIONS AND DESIGN

REQUIREMENTS TECHNICAL AND NON-TECHNICAL

Technical Requirements:
The main technical framework for our project to be coded in was React. Under the React umbrella, the technical requirements includes the following:

- Configuration and Routing
- Redux Implementation
- Hooks and Shared State
- Middleware and Services
- Testing

Non-technical Requirements:
The non technical requirements for our project have more to do with what the expectations are for our website outside technical needs. These include:

- The interface should be easy to use and friendly so our website can describe the services it provides clearly.
- Excellent uptime and availability for users to access whenever they need
- Help users understand what kind of travel experience they want to partake in
- The website should be responsive when a user increases or decreases the size of the screen, the content is still readable.
- Our site should present accurate and up to date information about the travel experiences users can engage in such as if restaurants are open or have closed down, they should not show up.
- Information regarding travel activities should be clearly displayed and easy to understand.
- Our use of APIs should be seamless and connect well with other platforms closely related to the services we display such as booking.com for hotels or gov.uk for travel advice

DESIGN AND ARCHITECTURE

Design:
The design we created for our website is aimed to be simple and aesthetic to the user so they are intrigued to explore each page.

Our prominently placed logo and brand name are positioned above a visible navigation bar, guiding users to labeled pages. A banner below clearly indicates the current page. Throughout, we employ Bootstrap's card and container components for a structured and easily digestible layout.

The home page uses images to show the possibilities of travel locations users can visit to evoke the desire to use our services.

The city breaks page presents four main options and a ‘read more’ button to allow the user to explore the options presented on the page and make an informed choice.

The sightseeing page includes an interactive map for the user to play around and delve straight into sightings within a given area.

A grid layout is used on the travel resources page to show resources the user can benefit from and includes a link for more details.

The search bar is presented at the top and placeholder text is used so the user is prompted to enter a city. The search button next to it guides users easily to find the results of their search.

Components for the destination and calendar information is placed at the top of the itinerary page to ensure each page flows cohesively.

Architecture:
Our app has a straightforward architecture, utilizing axios for backend/middleware and React for the frontend. Axios facilitates requests to external APIs, fetching information displayed on frontend pages for user viewing.

IMPLEMENTATION AND EXECUTION

DEVELOPMENT & TEAM ROLES

Our team embraced an agile development approach, conducting weekly meetings with predefined deadlines for subsequent tasks. During these sessions, team members assumed specific roles, fostering collaborative problem-solving and ensuring progress towards project milestones.

TOOLS & LIBRARIES

Html3pdf: This library allowed the user to create their personalised itinerary on our website. With this feature, users can compile their plans and download it as a PDF document.

Leaflet JS Library: Using Leaflet, we were able to add pins to the map, allowing users to conveniently visualise the locations of multiple attractions within a single map.

React-Datepicker: This provided the user with a user friendly interface to pick dates when planning their itinerary.

Fixer API: Utilised for a currency converter app, Fixer was used to retrieve real-time exchange rates. This functionality enriched the user experience by enabling dynamic currency conversions within the application.

TripAdvisor API: Integrated for restaurant search functionality, this API allowed users to explore and discover dining options based on their input location.

Bootstrap: This external dependency optimised key components such as cards and image carousel, ensuring efficient development.

TOOLS & LIBRARIES

GitHub: We leveraged GitHub as our version control system, promoting efficient collaboration and code management. Each team member created and worked on their own branch, allowing for parallel development without conflicts. Pull requests enabled comprehensive code reviews before merging into the main branch.

IMPLEMENTATION

Achievements:

- Successful integration of APIs (Fixer and TripAdvisor) and libraries (Leaflet, html3pdf and react-datepicker).
- Implementation of feature allowing users to download their itinerary.
- Creation of a colour theme toggle, providing the option to switch between dark and light modes seamlessly.
- This toggle functionality was achieved through the effective use of shared state, enhancing the overall user experience.

Challenges:

- Initial difficulty in finding a free API which accepts user-inputted cities and returns attractions.
- Encountered difficulty in devising shared state use case for all pages and implementing Redux.
- Unable to ensure responsiveness across a wider range of devices including smaller devices such as mobile phones.

Decision and Change:

- Adapted to the challenge by manually coding attractions on city breaks page.
- Subsequent decision to settle on the TripAdvisor API, specifically tailored to return restaurant information when a user inputs a city.

IMPLEMENTATION AND EXECUTION

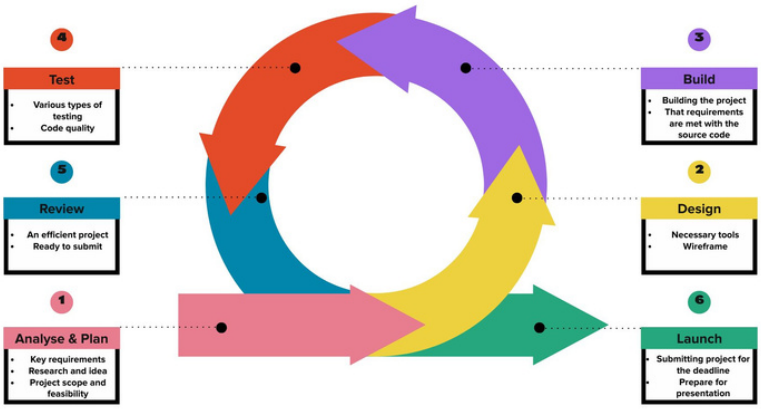
AGILE DEVELOPMENT

Agile development is a software development methodology that involves iterative development, collaboration, and user inputs. The practices we used as a team were the following:

- **Pair Programming:** Members of our team had Zoom meetings to pair programs. This helped promote collaboration and knowledge sharing and led to higher-quality source code.
- **Weekly Team Meetings:** These meetings provided an opportunity for us as a team to discuss our progress, identify any obstacles and synchronize any tasks.
- **Version Control:** Using Git helped us manage our source code and enable teamwork among us as a team.
- **Adaptability and Flexibility:** Agile is about responding to change as a team. We had to have the mindset of being able to adapt when code changed, which made changes to other pages on the website and learned from them.

Diagram 2 showcases our agile development process, which helps us to understand the overall project achievement as a team—from stage one, analysing and planning, to stage six, launching/submitting our project.

Diagram 2 - Agile Development Process



IMPLEMENTATION CHALLENGES

Having a successful implementation starts with a good and precise plan. But, of course, challenges did appear.

- **Understanding the scope of the project:** We needed as a group to focus on the website's purpose, who will be using it, users' expectations, the functionality, the problems we might face with languages and technologies and how it will benefit our target audience.
- I**mplementing new languages:** We have developed good coding knowledge in JavaScript and CSS as a team. We also used Bootstrap for the design and style of the website. However, we had some issues using Redux and could not use it in our project.
- **Project Time Limitations:** We had to deal with strict time constraints as we only had five weeks to complete this project. So we had to manage our time effectively as a team and prioritise what we wanted to achieve in this project.
- **The Complexity of the Project:** Working as a team, we knew we would work with many dependencies and interactions between different parts of the source code. So, we had to be careful what packages we used for each page.
- **Quality Assurance:** As a team, we had to be aware of reviewing our source code, as if we didn't, errors would start to appear, and that could use up valuable time that we don't have in this short period to complete our project.

TESTING AND EVALUATION

TESTING STRATEGY

The testing strategy employed for the NS Travel Services website focused on ensuring the functionality, usability, and reliability of the application. The following are key components of our testing strategy:

- **Unit Testing:** Individual components, such as the Currency Converter and Download Button, underwent unit testing to verify that each component's logic and functionality met the expected requirements. Jest, along with the React Testing Library, was used for these tests.

- **Integration Testing:** The interaction between different components and modules was thoroughly tested to ensure seamless collaboration. For instance, the integration of the ItineraryForm with localStorage and the functioning of the ThemeProvider and its impact on the application's theme were evaluated.

- **User Acceptance Testing (UAT):** Real users were actively invited to participate in a simple test. Engaging users in real-world scenarios, allowing them to explore the application's features. During UAT, participants provided valuable feedback on various facets of the system, including: visibility, functionality, and design aspects.

FUNCTIONAL TESTING

The functional and user testing focused on validating the core features of the NS Travel Services website and ensuring a positive user experience. The testing efforts covered the following aspects:

- **Currency Conversion:** The test simulated user input and verified that the conversion calculations were correct, providing users with reliable currency information.

- **Itinerary Management:** The ItineraryForm underwent testing to assess the functionality of adding and deleting days. This was crucial to guarantee that users can easily manage their travel itineraries, enhancing the overall user experience.

- **Download Functionality**: The DownloadButton was tested to confirm that the downloadPDF function is triggered appropriately when the button is clicked. This ensures that users can seamlessly download their itineraries as intended.

- **User Interface and Theming:** The Reviews component was tested to display reviewer names and review text correctly. Additionally, the ThemeProvider and useTheme hook were tested to validate the toggling of themes, allowing users to customise their viewing experience.

SYSTEM LIMITATIONS

Performance Testing Coverage: Our performance tests fell short in providing comprehensive coverage for all application units and components. Constraints in both time and knowledge impeded the execution of automated end-to-end testing.

Browser Compatibility: Although testing prioritized popular browsers, potential variations in rendering exist, especially in less common or outdated browsers. Further testing on a diverse range of mobile devices is recommended to ensure a consistently positive user experience.

External API Dependencies: The reliance on external APIs for travel-related information introduces a potential point of vulnerability. System performance may be impacted by issues such as API downtime or rate limiting.

CONCLUSION

The primary goal was to create a one-stop travel website, offering users comprehensive resources for crafting personalised trips and receiving tailored recommendations. This project advanced our React skills through tasks such as creating components, implementing routing and configuration, and conducting testing.

While working with React across multiple pages, we encountered technical challenges. Through testing and effective communication, we successfully addressed these issues. For instance, a few of the pages used Bootstrap components which meant that these classes had been assigned a default name. To prevent conflicts between components, we differentiated them by making necessary amendments.

We attempted to implement Redux as shown in the commented-out code in reduxSetup.js, encountered issues, and made efforts to resolve them without success. To improve code quality and maintainability we used folders to organise pages and components. We used created components to make development and readability easier. Our website maintained a minimal style across pages to avoid clutter and enhance readability. We ensured mobile responsiveness for cross-device along with the option of a dark mode for users to contribute to visual comfort.

To summarise, our coding project has successfully accomplished its objective of ‘creating a seamless user experience’ for travellers who prioritise the planning process but wish to alleviate the associated stress.

The collaborative efforts of our team, paired with invaluable resources, has resulted in a robust and fully-featured website. While we faced certain challenges during the project, successfully navigating through them has strengthened our abilities in collaboration and problem-solving.

Acknowledgements
This project would not have been possible without the collaborative efforts of Louise, Lauren B, Lauren H, Rainie, Remi and Princess. In particular, Louise has consistently demonstrated exceptional leadership skills throughout our project. She has been instrumental to our success significantly contributing to the positive outcomes we have achieved.