

**Zentist**

Dental Web Application

By

Samoil-Bogdan Adascalului

Submitted to

**The University of Roehampton**

In partial fulfilment of the requirements

for the degree of

**BACHELOR OF SCIENCE IN COMPUTER SCIENCE**

Abstract

Many dental practices in the UK, still rely on using old-fashioned methods to manage their day-to-day activities. Spreadsheets for calculating costs, various applications for all the actions they need to perform such as email, online leads application, applications reminding them to call patients for second part of the treatment, managing costs applications, etc. Another problem is lack of security layers when dealing with sensitive information.

The purpose of this project is to develop a Web App that helps dental clinics go digital. The WebApp was designed on feedback received from dental professionals and customer consultation. This information received from the two sources will represent the backbone of this project. The application will be coded using a mixture of HTML, CSS, JavaScript, Node.js, MySQL and other tools and frameworks.

The WebApp consists in a practice presentation website, where customers can find information about the practice, services offered and get to know their doctors.

The second component of the WebApp is a login system that will redirect the users to the main application. The user-friendly interface will provide users with various information and includes features such as patient registration, appointment management, medical history, etc. It also adds a level of security and prevents “prying” eyes from accessing sensitive information. A role-based system will be set up , so different users have access to different areas of the application.

The webapp back-end is developed using Node.js and Express.js, with a RESTful API to communicate with the front-end. The database is developed using MySQL, with access controls in place to ensure data security.

The result of this project is a digital platform that streamlines clinic operations, improves efficiency, and enhances patient experience in dental clinics.

**Declaration**

I hereby declare that this project on the development of a web app for dental clinics is the result of my own work. All sources of information used in this project have been duly acknowledged. This work has not been submitted for any other degree or diploma, either in part or in full.

I understand that any act of plagiarism or academic dishonesty will result in disqualification of this project and may result in disciplinary action.

**Date:** 12/05/2023

**Samoil-Bogdan Adascalului**

Signed (apply signature below)

Acknowledgements

As I completed this project, I realized that I could not have completed this project without the help, knowledge, experience, and patience of many others. I would like to take this opportunity to thank those people who have been instrumental and contributed significantly to the success of this project.

First, I would like to thank the University of Roehampton for allowing my dream of becoming a Computer Science student come true and for the support I was provided throughout my undergraduate years.

I would like to also express my deepest and most sincere gratitude to the project supervisor, Lecturer Kimia Aksir, for her guidance and patience while monitoring my progress and providing invaluable support and information towards the completion of this project.

To all my friends and family who have stood by me during all those months when I worked on my project ideas and did not let me give up.

To my colleagues, Zakariya Oulhadj and Taylor Head who always believed in me and provided tremendous moral support.

Table of Contents

[1. Introduction 6](#_Toc134719802)

[1.1 Research Question or Problem that will be addressed. 6](#_Toc134719803)

[1.2 Aims 6](#_Toc134719804)

[1.3 Motivation 6](#_Toc134719805)

[1.4 Objectives 7](#_Toc134719806)

[1.5 Legal, Social, Ethical and Professional Consideration 7](#_Toc134719807)

[1.6 Background 7](#_Toc134719808)

[1.7 Report overview 8](#_Toc134719809)

[2. Literature and Technology Review 10](#_Toc134719810)

[2.1 Literature Review 10](#_Toc134719811)

[2.1.a Paper Medical Records 11](#_Toc134719812)

[2.1.b Computer stored medical records. 11](#_Toc134719813)

[2.1.c Conclusion 12](#_Toc134719814)

[2.2 Client requirements 12](#_Toc134719815)

[2.2.1 Customer Consultation & survey 13](#_Toc134719816)

[2.2.2 Results 13](#_Toc134719817)

[2.3 Technology review 15](#_Toc134719818)

[2.3.1 Version Control 15](#_Toc134719819)

[2.3.2 Text Editor 15](#_Toc134719820)

[2.3.3 Runtime Environment 15](#_Toc134719821)

[2.3.4 Template Engine 16](#_Toc134719822)

[2.3.5 Front-end development framework 16](#_Toc134719823)

[2.3.6 Database 16](#_Toc134719824)

[2.4 Summary Table 17](#_Toc134719825)

[2.5 Considerations 19](#_Toc134719826)

[2.5.1 Legal Consideration 19](#_Toc134719827)

[2.5.2 Ethical Considerations 20](#_Toc134719828)

[2.5.3 Professional Considerations 20](#_Toc134719829)

[2.5.4 Considerations conclusion 21](#_Toc134719830)

[3.Design 21](#_Toc134719831)

[3.1 Design objective 21](#_Toc134719832)

[3.2 User interface design 23](#_Toc134719833)

[3.3 Alternative Approaches 23](#_Toc134719834)

[3.3.1 Alternative User Interface Designs 23](#_Toc134719835)

[3.3.2 Alternative Development Methodologies 23](#_Toc134719836)

[3.3.3 Alternative security measures 23](#_Toc134719837)

[3.3.4 Alternative stylings 24](#_Toc134719838)

[3.4 System Architecture 24](#_Toc134719839)

[3.5 Wireframes 26](#_Toc134719840)

[3.6 Project Management Approach 27](#_Toc134719841)

[4.Implementation 29](#_Toc134719842)

[4.1 Implemented features 29](#_Toc134719843)

[4.2 Step by step implementation 29](#_Toc134719844)

[4.3 Evaluation 39](#_Toc134719845)

[4.4 Related Work 41](#_Toc134719846)

[5.Conclusion 43](#_Toc134719847)

[5.1 Reflection 43](#_Toc134719848)

[5.2 Future Work 44](#_Toc134719849)

[6.References 45](#_Toc134719850)

[7.Appendices 47](#_Toc134719851)

# **Introduction**

Working in an environment, such as a medical or dental practise, can be extremely stressful and demanding, especially if the practise has not yet opted to go fully paperless. Keeping up to date with all the paperwork that a practice needs to run smoothly, can put tremendous weight on someone’s shoulders. Calling patients, rescheduling appointments, booking new appointments, providing information to incoming patients, logging in to numerous different applications and remembering all login credentials are just a few of the tasks the staff of a medical practice needs to remember to do at the start of every day.

## 1.1 Research Question or Problem that will be addressed.

With all the problems to solve in such a brief time, is no wonder that sometimes medical practitioners do not have the best results at work. This project will attempt to address a few issues that all dental practises in the United Kingdom that have not gone digital confront on a regular basis. The first and most important one is the fact that many do not have a practice presentation website, a place where customers can visit and find out more about what, why and where of their medical practice. The second tackled problem is that a large number of these medical practices have their data unprotected. Sensitive information can be easily lost or copied by individuals with ill intentions. This can be solved by implementing a web application with a multi-layered login system for various positions in the medical practice.

## 1.2 Aims

The goal of this project is to create a user-friendly web application that will improve time management for the dental clinic's staff. The application will assist staff by providing all necessary information about a customer, an appointment, a doctor, and other relevant information without the need to search through several paper folders or juggle between a paper-based system and a crude, insecure spreadsheet system.

Another aim is to assist dental practises in saving money while becoming more efficient. Providing them free dental software will help them, and the money saved can be used in other areas of the clinic, such as staff training, salary rises, or system upgrades.

The third aim of this project is to improve the overall productivity of the dental practice staff members by allowing them to spend more time with the patients, hearing their concerns and needs, rather than spending hours to keep track of multiple spreadsheets, looking for files and so on .

## 1.3 Objectives

Several objectives must be completed to ensure that by the end of the project, all of the objectives I have set are met. They will also help me to address the problems exposed in the Chapter 1.1 (Research Questions)

a. UML Diagram – Activity diagram of the website (Chapter 4)

b. Customer consultation - identify key elements of a dental web application.(Chapter 2)

c. Login System – protect the customers data by adding an extra layer of security.(Chapter 5)

d. Analytics Section – give administrators insights about the revenue.(Chapter 5)

e. Contact us / Feedback Section – user feedback that can be used for multiple purposes.(Chapter 5)

## 1.4 Legal, Social, Ethical and Professional Consideration

In developing a dental clinic web app, various legal, social, ethical, and professional considerations that can affect the development, deployment and use of the application must be considered. A comprehensive consideration of all these can help identify potential issues and easily mitigate them during development. Among these are patient data confidentiality, informed consent, and data privacy and security. Additionally, it can aid in building credibility and trust with stakeholders and end users, which will have an effect on the application's and the development team's reputation. Given the rapid pace of technological development, failure to address the ethical, social, and legal considerations raised throughout the development of an application can result in serious harm to individuals, businesses, and society as a whole, according to The Ethics of Computing: A Survey of the Computing-Oriented Literature [[15]](#_6.References). This study will attempt to discuss these issues and how they were addressed during the development of the Zentist WebApp.

## 1.5 Background

In recent years, there has been an increasing demand for dental clinics to incorporate digital technology into their operations. Some clinics' traditional paper-based systems are inefficient, time-consuming, and prone to errors. Given recent developments, such as COVID-19, which has made clinics' duties even more difficult, they are faced issues they have never faced before, such as keeping social distance and limiting physical contact to a bare minimum. According to a study published in the International Journal of Science and Research[[14](#_6.References)], there was a considerable increase in the number of clinics that invested in office digital conversion during and after the COVID-19 pandemic. Furthermore, the studies and interviews I conducted for this research revealed that some clinics do not have a proper digital data storage system or that the current one needs to be updated.

Furthermore, this project is a suitable BSc project because it requires the integration of numerous technologies and abilities that are often taught in computer science programmes. The project involves the development of a prototype web application using Node.js, Express.JS, MySQL, HTML, CSS, JavaScript as well as knowledge about software engineering concepts such as agile development, user requirements analysis and project management. Through this project, I was able to gain practical experience in developing an application that solves a real-world problem.

## 1.6 Report overview

This report will have six sections and each one will tackle different areas of the Zentist project development.

* **Literature and Technology review**

Provides a review of the relevant literature and technology used in development of the Zentist WebApp, pros and cons of the transition from paper-based to computer-based systems, client requirements etc.

* **Design**

Describes the design stages of the dental application including features, functionality, and user interface. This section will discuss the design decision made during development.

* **Implementation**

Outlines the implementation of the dental application, including the software development process, testing and more. Also, in this section I will discuss various subjects such as implemented features, extra features, etc.

Evaluation subsection provides an assessment of the success of the dental application and recommendations for improving efficiency and customer satisfaction.

The related work section assesses the success of the dental application and makes recommendations to improve efficiency and customer satisfaction. This section will compare the Zentist with existing dental management systems and highlight the advantages and disadvantages. Also, it will highlight the work that needs to be done in order to keep Zentist competitive.

* **Conclusion**

Summarizes the main findings of the report, highlights the contributions and limitations of the research, and provides recommendations for future work.

* **References**

List of sources that were cited in the report.

* **Appendices**

Includes any additional materials such as screenshots, technical specifications, etc. that are relevant, but not essential to the main body of the report.

# **Literature and Technology Review**

## 2.1 Literature Review

With the majority of practices already transitioned from paper medical records to computer or cloud stored medical records, we still have many practices use the old-fashioned methods or still weighting the benefits of paper vs electronical medical records (EMR).

Studies like the ones conducted by J. Tsai[[9]](#_6.References) and J. Stausberg[[10]](#_6.References), revealed the pros and cons of each method, and researchers produced interesting conclusions. J. Tsai and G. Bond [[9]](#_6.References), concluded that on average, EMR were approximately 40% more complete and 20% faster than traditional paper-based records, while J. Stausberg and his colleagues[[10]](#_6.References) found that one of the main issues affecting EMR data quality is the “dual documentation practice”. Other studies have attempted to pinpoint the difficulties that dentists have on a daily basis when managing a dental practice, including time management, patient communication, appointment scheduling. According to a study by Viral [[7]](#_6.References), researchers found that practices have a waiting time varying between 13 minutes (five-star practices) and 34 minutes (one-star practices). This means that the staff need to spend more time doing administrative tasks (ex. Patient information sheets completed) and taking time away from patient care.

Chart, bar chart, histogram

Description automatically generatedAnother conclusion of the study is that 1 in 5 patients change their dentist due to the waiting times. The study discovered a direct corelation between the waiting times and star-rating of the practice. That means longer waiting times, less satisfied customers. Less satisfied customers, less revenue in the future. Peter Kriss (Senior Researcher at Medallia and Director of Research for Vizio Prize) states in an article for “Harvard Business Review” [[8]](#_6.References), that after examining two businesses in the same industry he concluded that customers who had the best customer experiences spent 140% more than those who had the worst experiences who were less likely to return.

Figure 1 Corelation between customer experience and loyalty

### 2.1.a Paper Medical Records

### Any patient medical information recorded on paper is referred to as paper medical records. Patient medical history, medical charts, medicines, and other information can be included. Despite the fact that they were commonly used 10-20 years ago, many practises have now gone digital. Although most medical practises have digitalized their medical information, some still question the benefits and drawbacks of each system.

#### A. Advantages

1. Reduced costs - The main advantage of the paper medical records are the reduced up-front costs. The practices need only an initial investment to start the business. Only a printer, paper, pens, and a filing cabinet are required to begin the business.

2. Ease of using them – Training needed to create, complete, and store these files is rudimentary and this is the main reason many practices still prefer this method. In a field where speed can make the difference between life and death, even a small learning curve can make a difference.

3. Easy to customize- another money-saving advantage of the paper medical records is that whenever you need a new design of the form you can just edit it in a text editor, whilst with the EMR, you need a developer to make the adjustments both in front and back-end of the application. This is both time and money consuming.

#### B. Disadvantages

1. Storage space – Storing the files take a lot of physical space that otherwise could be used for business purposes.

2. No back-up – One of the main disadvantages is that in case of loss there is no back-up solution, leading to total loss of important medical data.

3. Limited security – Lack or limited security is one of the reasons many practices choose the digital version.

### 2.1.b Computer stored medical records.

Computer stored medical records are the digital equivalent of the paper medical records. According to HealthIT.gov [[6]](#_6.References), using electronic health records has a few advantages and disadvantages such as:

#### A. Advantages

1. Fewer storage limits – Unlike paper medical records, the data can be stored in a local database or most EMR come with cloud storage possibility. Both will take minimum to no physical space.

2. Easier to understand – All files having the same layout and same writing will be easier to understand by the rest of the staff.

3. Secured – Compared to the paper medical records, computer stored medical records can come with a security protocol implemented. This way only the authorized staff can view the confidential information.

4. Possibility to integrate with e-prescribing tools - Can be integrated with e-prescribing platforms to improve patient’s experience. Mobility limited can attend phone appointments and receive their prescriptions online.

#### B. Disadvantages

1. High maintenance and upfront cost - Most of the EMR come with a one-time setup fee that can go up to a few thousand pounds. Also, they often charge monthly usage fees that can add up to a few hundred pounds.

2. Training Costs - Having an instructor to train the staff on the new systems could increase the total costs of implementing an EMR into the clinic. The result may be that for a period, the staff might not be able to quickly respond to patient needs as usual.

3. Dependence to internet- Most EMR need a high-speed internet band to quickly load the system information, but without one, the entire process is more frustrating than helpful.

### 2.1.c Conclusion

Finally, after weighing the pros and cons of both techniques, I concluded that the computer-stored solution is preferable. Even though it has higher implementation costs than the paper-based system, managers should consider that this is a one-time payment that will bring more efficiency to their practise and thus more revenue, and the benefits of the computer-stored system outweigh the disadvantages.

## 2.2 Client requirements

Understanding customers’ needs and requirements, is one crucial step when developing a CRM system. Knowing the final user’s needs will greatly improve the productivity (knowing from the start all the features of the application, will make planification easier), but also will enhance the time spent developing the system (not having endless communication about the features and how they should work).

According to C. Rees [[4]](#_6.References), there are four main ways to find the clients requirements:

* Customer Consultation
* Focus Groups
* Surveys
* Market Research

##### 2.2.1 Customer Consultation & survey

In the case of Zentist Dental Web Application, I used two methods to find the client’s requirements: direct customer consultation and surveys. The results gathered through both methods will bring great value to the final product by giving insights regarding the dental industry and how to better help practices to be more client-oriented.

Customer consultation involved face to face interviews with practice managers and dental clinics area managers to understand the needs and challenges they face when managing a practice. The aim of these interviews was to find key areas where a web application could bring value to the companies and to the services provided.

Online survey was used to gather feedback and obtain quantitative data from dental professionals that would reflect the user’s needs. The questions were specially designed to extract information about specific areas:

* The key features of a dental management system
* The security and reliability of the current system
* The usability and accessibility of the system

Participants were recruited through personal networks and professional associations, and they were required to have at least 2 years of experience working in the dental clinic. The data gathered through the customer consultation and surveys, was used to analyse and find common themes and requirements. The survey was hosted on freeonlinesurveys.com and it can be accessed using the URL <https://shorturl.at/npGJP> , also a template of the survey can be viewed in the Appendices section figure 21,22,23. All the data gathered was then used as guidance in the design and development stages of Zentist.

### 2.2.2 Results

The results of the customer consultation and survey revealed the features an application like Zentist needs to have in order to be useful for the workers in this industry.

According to the results, I have noticed that one of the key features, no matter the position they hold in the company, is a customer records management feature. Surprisingly 89% of the interviewed chose the CRM feature as the most important one.

This was followed closely by sales analytics, with 78% of responders indicating the importance of this feature. Knowing that amongst the interviewed we had doctors, managers, treatment coordinators that have commission-based contracts, this choice was easy to understand. Interestingly, some participants also expressed a desire for features such as messaging system, car parking validation, payment. This suggest that in the future the applications usability can be expanded.

Chart, bar chart

Description automatically generatedBellow you can find a list of features that a dental system needs to have, according to the survey.

Figure 2 Features requested by responders

One limitation of the survey is that it relied on self-reported data, which may be subject to bias or errors. Additionally, the sample size was small, which may limit the generalizability of the findings. However, the insights gathered from the survey can still be valuable guidance for the development of the dental clinic application. According to the survey I performed ([Appendix – figure nr 9](#_9.Survey_question_9)), the following five features were voted by the majority of the participants as essential for an application like Zentist.

* Customer Records Manager

The section allows dental practitioners to add, edit, view patient information, assign doctor, medical history, and previous appointments notes. The platform also includes a search function for finding specific patient records.

* Invoice manager

This feature allows the users to provide patients with an invoice for the services they received. They can supply invoices after each treatment performed in practice and they can give the customer a full detailed invoice of all the services, their cost, total cost, and the doctors that performed them.

* Data Security

This feature will be implemented by putting the entire application behind a login system. Users that do not have credentials will not be able to access any information. Based on the role assigned by the administrator of the website, certain sections will be available only to the administrators/managers of the clinic.

* Sales analytics

This section provides visual representations of the clinic’s revenue data. Using charts and graphs, the managers will be able to see and interpret the clinic’s data and analyse steps that needs to be taken.

## 2.3 Technology review

Having the experience of working in a four-persons Scrum team to tackle a project, clearly gave me a good insight of how things should be managed to have a good outcome. Bellow you will find a list of choices I had to make regarding the technology I will use for this project.

### 2.3.1 Version Control

Since the whole project is created having multiple milestones, having a version control option is vital. I had to consider three main options for this: Bitbucket, GitHub and GitLab. Knowing GitHub is the most used tool and also having prior experience on working with it on different projects, helped me with my choice.

### 2.3.2 Text Editor

While having multiple choices at my disposal such as Atom, Brackets, SublimeText, my choice of preference was VSCode, because it has the best extension system and has integration with many other editors. The fact that I am familiar with it from my previous projects, played a significant role.

### 2.3.3 Runtime Environment

When choosing the runtime environment, I had to question myself about the size of the project and scalability. After researching the right runtime environment, several advantages of NodeJS convinced me that this is the right environment .

1. Fast and efficient – uses an event-drive, Non-blocking I/O model that makes it fast and efficient, even when dealing with large amount of data.
2. Single programming language – developers cand use JavaScript on both the server-side and client-side.
3. Scalability – can easily manage large number of connections.
4. Easy to learn.

Since this application is only a prototype of what it could become, Node.js provides a very good infrastructure to easily create both a small-size prototype of a dental clinic application or a complete and performant dental clinic application with all the features necessary.

### 2.3.4 Template Engine

When choosing the right template engine, I looked for one that has integration with the runtime environment chosen above.

Pug (formerly known as Jade) is a high-performance template engine that simplifies the process of creating HTML markup making it easier to read and write.

a. Concise syntax – simple syntax that make it easy to read and write.

b. Faster rendering – generates highly optimized HTML markup that is faster to render in the browser.

c. Reusability – Pug supports template inheritances which allows users to define a base template and extend it in child templates.

### 2.3.5 Front-end development framework

While choosing the right tool for the front-end development, I had to choose between Bootstrap and Tailwind, both having different approaches. While Bootstrap is a framework that provides a large library of pre-built HTML, CSS, JavaScript components, Tailwind is a utility-first framework that provides a set of predefined classes that can be used to create custom designs.

In conclusion because this is a prototype and custom design can be created in the later stages of the development, I choose to go with the all-in-one solution, Bootstrap.

### 2.3.6 Database

While evaluating the possibilities, several alternatives were considered in the design stages of the project. A NoSQL database such as MongoDB was considered, however MySQL was chosen for several reasons such as, compatibility and familiarity. Not being familiar with MongoDB would have led to a lot of time being lost in the implementation process, so in the end I chose MySQL.

## 2.4 Summary Table

List of technologies that will be used within this project:

* Visual Studio Code (version 1.78.1) – Text Editor
* GitHub Desktop(Version 3.2.3) - app to interact with GitHub.
* GitHub - Code hosting platform for version control
* Node.js(Version 16.14.0) – runtime environment for executing JS code.
* PUG(version 3.0.2) – Template engine
* Express.js(Version 4.18.2) – Web Application framework
* Bootstrap(Version 5.0) – Front-end development framework
* Bootstrap Icons(Version 1.8.1) – Open-Source icon library
* Plotly.js(Version 2.22.0) – Charting library
* Html2Pdf.js(Version 0.9.3) – Html to PDF converter
* jQuery(Version 3.6.1) – JS Library
* Font Awesome(Version 4.7.0)- Open-Source icon library
* MySQL – Database
* Form Submit- form backend, API, and email service
* Docker(Version 4.4.4)- containerization platform

Bellow you can find a table with benefits, limitations, and the reason of use for some of the technology used. It includes the strengths of each application and how they contributed to the overall functionality of the web application.

|  |  |  |  |
| --- | --- | --- | --- |
| Technology | Benefits | Limitations | Why it was used |
| Visual Studio Code | Free, open source | Steep learning curve for beginners | Powerful text editor with features that can improve development productivity |
| GitHub Desktop | Simplifies GitHub interactions | Limited functionality compared to command line | Easy-to-use interface for interacting with GitHub and manage the version of codebase |
| GitHub | Code hosting and version control | Limited storage for free accounts | Allows multiple users to work or have access to same codebase and keep track of changes made. |
| Node.js | Cross platform and scalable | Asynchronous programming can be challenging | Powerful runtime environment for executing JavaScript code and build scalable web applications |
| PUG(Jade) | Easy to learn and use | Limited documentation | Templating engine designed for HTML page rendering in server-side tech like Node.js |
| Express.js | Lightweight and flexible web application | No built-in support for database management | Powerful web application framework for building scalable and customizable server-side applications |
| Bootstrap | Easy to use and customizable frontend design | Limited options | Comprehensive front-end development framework for building responsive web applications. |
| Plotly.js | Interactive charting library | Limited charting customizing options. | Charting library for creating interactive and dynamic charts and graphs. |
| MySQL | Stable and widely used | Can be challenging to scale to large datasets | Reliable and scalable database management system |
| HTML2Pdf.js | Easy to use HTML to PDF conversion tool | Limited options for advanced PDF features | Straightforward way to convert HTML pages into PDF format. |
| jQuerry | Simplifies DOM manipulation and event handling | Has performance issues with large-scale applications | Powerful JavaScript library for simplifying the process of DOM manipulation |
| Docker | Provides containerization for application deployment | Requires additional learning curve for docker specific tools | Enables efficient and streamlined application deployment |
| FormSubmit | Simple and user-friendly form backend service | Limited functionality compared to custom backend | Easy to use from backend and email service for handling form submissions. |

Figure 3 List of technologies used

## 2.5 Considerations

The development and implementation of a dental web application require careful consideration of legal, ethical, and professional standards. This chapter will explore the considerations in depth, outlining the various legal requirements and ethical principles that must be adhered to, as well as the professional standards that must be met.

### 2.5.1 Legal Consideration

When building an application that manages patient information, practice information and all sorts of confidential information, there are two legal aspects that need to be considered: privacy and confidentiality of patient’s details, and patient’s informed consent.

**Privacy and confidentiality of the patient’s details**

According to Richard Rognehaugh’s, *the health information technology dictionary* [[1]](#_6.References), privacy can be defined as an individual’s right to keep his personal information from being disclosed to anyone else. Patient details should not be issued to anyone without the patient’s consent. In cases where patients are unable to give their consent due to their mental incapacity of taking decisions, age, and information sharing decisions should be taken by the legal guardian or legal representative.

One of the main legal considerations for this project is data privacy and protection. The application will be handling sensitive patient information, including medical history, personal details, and appointment schedules. The key to preserving the privacy and confidentiality of the patients is to have a system that allows only authorized personnel access to this kind of sensitive information. We must make sure that the application complies with all relevant data protection regulations such as General Data Protection Regulation (GDPR) [[11](#_7.References)], to prevent unauthorized access, use or disclosure of such information. The first step that needs to be taken to accomplish this, is to authorize and give access to users according to the user’s position in the dental practice. The administrator should decide the amount of information any user should have access to and assign the relevant username and password. All the users should sign a GDPR agreement where they agree to comply with data handling procedures. This needs to be implemented into the application.

**Informed consent**

Is a core principle in medicine, that the patient needs to have all the information about the benefits and the risks of a procedure that he needs to be subject of. This will allow the patient to refuse or undergo the given procedure according to American Medical Association’s Code of Medical Ethics [[12]](#_6.References).

One crucial step in implementing this into our system is that the doctors need to explain in-depth all the ramifications of the procedure they are about to perform. After thoroughly explaining every single aspect of the procedure, the patient needs to sign the informed patient consent that will be added to the patient file. All of this will benefit the practice, but it will also protect the doctors from being sued for malpractice if patients are injured during treatment.

### 2.5.2 Ethical Considerations

In developing this application, we must also consider ethical considerations related to the handling and use of patient data. We must make sure that we handle patient data ethically, with respect to patient privacy and confidentiality. We must ensure that we use the data only for legitimate purposes, such as medical care and not for any other purposes, such as marketing or research, without the patient’s explicit consent. To implement this, every single employee must sign the practice ethical code of conduct.

### 2.5.3 Professional Considerations

As professional software developers, we must adhere to a set of professional standards and ethics. This includes designing and developing applications that are reliable, secure, and scalable. We must also adhere to a code of conduct that emphasizes honesty, transparency, and professionalism. This includes ensuring that the application is free of bugs, vulnerabilities and other defects that may compromise the security and privacy of patient data.

## 2.5.4 Considerations conclusion

Graphical user interface, text, application, email

Description automatically generatedAfter researching the legal, ethical, and professional considerations, I have decided to add a consent form to our application that needs to be read and signed by the patient. This way, he confirms that he understands how his personal data will be handled, what his treatment would entail as indicated by his doctor, and that he is willing to give certain information about his medical history. After signing the form, the user will save the file to the patient's folder in the root directory. Also, all users will have to agree to GDPR policies of personal data handling.

Figure 4 Part of the consent form

# **3.Design**

## 3.1 Design objective

When designing the application, I had in mind the following objectives and all the feature implemented had to consider these objectives:

* To provide dental clinics with a user-friendly platform that will facilitate digital management of the patients records and appointments.
* To reduce paper-based data storing system, leading to an eco-friendly work environment
* To streamline clinic operations, reduce human errors and improve efficiency.
* To provide features that facilitate communication between patients and dental professionals.
* To ensure that the web app is secure, dependable, and scalable.

Diagram

Description automatically generated with medium confidence

Figure 5 Application Activity Diagram

Diagram

Description automatically generatedTo implement these objectives into an application, several technologies had to be used such as those stated in chapter 2.4. Also, an activity diagram was created to better understand the way all components should interact.

Figure 6 “Show customers” activity diagram

## 3.2 User interface design

User interface design is a critical aspect of developing a successful web application like Zentist. The primary goal of the user interface is to ensure that the application is easy to use, intuitive and visually appealing. In general, the user interface design should make it easy for the user to access and use the features of the application making it easy for them to accomplish their daily tasks.

To accomplish this the following principles should be considered:

1. Consistency – the design should be consistent throughout, using consistent colours, typography, and layout. All this should lead to a better user experience.
2. Simplicity – the design should be kept simple, avoiding unnecessary complexity. Simple designs lead to a better user experience.
3. Clarity – The language used in the application should be simple and easy to understand
4. Accessibility – The design should consider at least some of the needs of the users with disabilities.

Following these principles, the user interface should be easy to use, intuitive and easy to work with. Various wireframes how they were used can be found in the wireframes section and [appendix](#_25._application-_user).

## Alternative Approaches

Alternative approaches were considered during the design phase of Zentist to ensure that the final design was efficient.

### Alternative User Interface Designs

Before deciding on the final version, various user interface concepts were investigated. The main considerations were the simplicity and ease of use for the end-user. Multiple layouts, colour schemes, and font options were investigated. Following extensive testing and assessment, a simple design was chosen for the application's final user interface design.

### Alternative Development Methodologies

Different development methodologies were considered to ensure that the project is delivered in time and to the required quality standards. Waterfall and Spiral were considered as a development methodology, but in the end, I decided that AGILE methodology fits the development of this particular project the best, because of its flexibility and rapid software development capabilities.

### Alternative security measures

Different security measures were considered in the design stage, to ensure the security of the application and also the confidentiality of the patient’s personal data. After evaluating some of them and knowing that the application is only a prototype, I decided to go with a simple login system, which involves querying a database and store the information into a session. The decision to use Docker in this project also improved the security by creating isolation layers between applications and limiting the potential impact of security breaches.

### 3.3.4 Alternative stylings

During the project's design stage, I had to pick what kind of styling I wanted to utilise for this project. I had to choose between external, internal, and inline styling. Since this is only a prototype and time management was essential, I chose to use inline and internal styling since is easier to keep track of, but also makes the application faster, since the order of execution is always inline -> internal -> external. Even though it has a few disadvantages, this can be changed in the future.

## System Architecture

System architecture refers to the overall design of the application, its components and how they interact. Bellow you will find a description of the three main components of the system.

1.Front end - This component will be responsible for the user interface and will be developed using HTML/CSS/JavaScript. The front-end will be designed with a clean and intuitive interface that is easy to use for both dental workers and patients.

2.Back end - The backend will be developed using Node.js and Express.js, that will be responsible for server-side scripting , server configuration and will allow frontend to communicate with the server. 3.Database - The database will be developed using MySQL and it will store all the patient records, appointments, users’ credentials, and other valuable information. The application’s database contains information regarding the users(username,password,email,role),staff(id,name,email,position,phone) and patients(name age, phone, email, treatments, costs, notes, etc.)

A picture containing table

Description automatically generated

Figure 3 Database tables

The interaction between the three components can be seen in the image bellow. The application fetches information from the database through the NodeJS + ExpressJS server that sends back packets of information as objects that can be accessed. The login / register process is done using the GET and POST methods that selects data from database or insert or update fields in the database. After that a comparison is done in the back-end and if the credentials inserted corresponds to the ones in the database, the login process starts.

Diagram

Description automatically generated

Figure 7 Diagram of system

## Wireframes

Wireframes are visual representation of the user interface design to help the developers better understand what the final product should look like. They are an essential part of the design process and provide a blueprint of the finished application design.

In the case of Zentist, wireframes were used to map put the various screens and features of the application. The wireframes were created using the online tool provided by Mockflow.com ([https://www.mockflow.com/).](https://www.mockflow.com/).%20) This helped a lot in in visualizing the application’s design before any code is written.

Graphical user interface, website

Description automatically generatedDiagram

Description automatically generated

Figure 9 Implementation

Figure 8 Index page wireframe

## 3.6 Project Management Approach

The project was managed using an Agile methodology. The development was done in sprints, with each sprint having approximately four weeks, this being one of the main ideas behind Agile (incremental delivery), according to www.agilealliance.org [[16]](#_6.References) . During each sprint, I focused on completing specific features and functionalities of the website. Regular weekly meetings were conducted with the project supervisor to discuss progress, identify any roadblocks and plan for the next sprints. The project was divided into smaller tasks and each task had a specific deadline.

Logo, company name

Description automatically generatedDiagram

Description automatically generated

Figure 9 Agile methodology

Figure 10 Agile incrementation

To ensure the quality of the prototype, regular testing and feedback from both customers and staff was received throughout the development process. Any issues or bugs identified were addressed promptly to ensure the prototype meets the aims and objectives of the project.

Also, a project management tool was used to help the participants better understand and plan the developing of the application. This involved the creation of a list of tasks required to complete the project. The application’s features had to be easy to understand and use, such as possibility to change the status of a task, manage delays, implement deadlines, etc. After consulting with several sources about the tool that has the most features, I reached to the conclusion that the app provided by Teamwork.com is the one that fits the project the best.

Timeline

Description automatically generatedFigure 11 Gantt chart of project management

# **4.Implementation**

The implementation stage of Zental Dental Web App was a crucial phase in the development process. The section presents an overview of the technical details and steps taken to accomplish the project’s objectives, but also how the tools in the previous sections were used. As stated in the design section, the implementation followed an agile methodology ,which allowed me to iteratively build and evaluate the application with help and feedback from professionals within the industry.

## 4.1 Implemented features

The project’s implementation was carried using the tools I choose in section 2 (Technology Review), and I have managed to implement the following features:

User Authentication: The application requires users to sign up and log in before accessing the information within the system. User authentication is implemented using session-based authentication, where user credentials are verified by the server and a session is created for the user.

Customer Management: Users can add, edit, and delete customer records, generate invoice, prescription, consent form. Customer records include basic information such as name, address, phone number, email address, but also information regarding the treatment such as date of treatment, type of treatment, payment status etc.

Analytics Dashboard: The application provides users with a dashboard that displays various metric related to customers and appointment data. Users can view charts and graphs that provide insights about the business performance.

Admin Dashboard: The application also provides and admin dashboard for users with such privileges. The admin dashboard allows admins to manage user accounts and access control.

Contact Agenda: The application will also provide the user with a section where he/she can find contact information of each employee of the clinic.

## 4.2 Step by step implementation

To achieve the desired outcome, first I had to produce a high-level step by step plan of the project:

1.Set up the development environment

2.Connect database to environment

3.Implement the CRUD operations

4.Implement the login feature

5. Analytics section

6. Contact us section

A picture containing text, screenshot, font, logo

Description automatically generated

Figure 11 Step by step implementation to fulfil the objectives of the project.

##### Set up the development environment

As stated in the technology review section, to start developing the application, first I had to set up the development environment. First step was to install NodeJS, which can be done by visiting official website (<https://nodejs.org/en/download>) and choose the desired package.

Text

Description automatically generatedSecond step was initialisation of the project. By using a simple command, such as we can quickly create an Node.js project. The “npm init” command, will create a package where the project dependencies will be stored. More information about the content of the package can be found in the [appendix](#_26._About_page) section. After the initialisation of NodeJS project, I had to install the dependencies using the npm( the world’s large software registry) and use it in the project.

Figure 12 Assign the dependencies

Text

Description automatically generatedThe third step when setting up the environment is to define the routes. This will allow the server to recognize requests for specific URLs and render the appropriate pages. More info about routing, what is and how is done can be found in [appendix](#_7.2_Routes) .

Figure 13 Create site routes

##### Connect Database to environment

Text

Description automatically generatedConnecting a database to the environment is a major step, when developing a web application. The environment is where the web application is hosted, and the database is where the data is stored. After installing MySQL “module” in NodeJS, I connected the database created in the section 4.4 System Architecture, with the newly created environment. This way we can use the data from the database in Zentist. For this project I chose to use createPool(), instead of createConnection() for speed related issues.

Figure 14 Link the database and the environment for data manipulation

Practically createPool() , allows the application to reuse connections and avoids establishing a new connection for each request.

##### Implement the CRUD operations

A green arrow pointing to a black background

Description automatically generated with low confidenceText

Description automatically generatedThe operations that CRUD consists of are the following four: create, read, update, and delete. They are a set of operations used in web application that use data management. This CRUD operations mixed with the routes that I defined in the previous step , helped me create, read, update, and delete information from the database.

Figure 16 Interaction between server and database through CRUD operations

Figure 15 Example of CRUD operations

After the two previous steps were implemented, connecting the MySQL with NodeJS and implementation of CRUD operations , I have managed to retrieve the desired information from the database

##### Implement the login feature

One of the key objectives of this project was to design the application with a login feature. This will provide an extra level of security against data attacks.

A screenshot of a computer

Description automatically generated with medium confidence

Text

Description automatically generatedFigure 17 Login feature, back-end implementation

Figure 18 Login feature, front-end implementation

Graphical user interface, application

Description automatically generated

Figure 19 Result

When I started to implement the login feature, I had to create a form that contains all the login information such as the one in the image above. In the back end, there are two different methods provided by Express.js that handles the information in the login page (app.get() and app.post()).

App.get () is a method that defines a route for handling HTTP GET requests. When a client sends a GET request to the login URL, the call-back function is executed using two arguments: “req”(the request object) and “res” (the response object). The call-back function can perform multiple actions such as, rendering a view, sending a response, or redirecting to another URL. In my case the call-back function is rendering the page “login” which can be found in the “views” folder, which the is set as the root folder by default.

App.post () is the second method that defines the route for handling HTTP POST request. The req.body object contains all the information in the request body. In this case it will get the information from the req.body about the username and password and it will crosscheck it against the information in the database. After the crosscheck, if the username or password does not exist in the database, the login page will be rendered along a message “User does not exist”. If the user does exist in the database, the server will redirect the client to the “results” page.

A role-based system was created so that not everyone has access to the specific areas of the web app based on their role. As the login system is role based, different users with different roles have different acces when they login.This role base-system can be greatly improved in the future, but as the application is a prototype a simpler implementation was chosen. A session was used to keep a user’s credentials stored and used when needed.

Graphical user interface, application

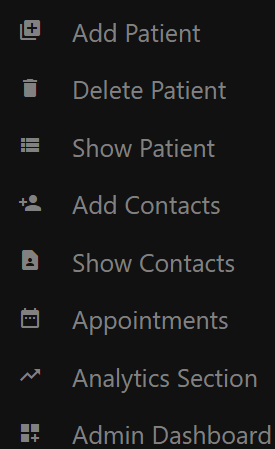
Description automatically generated

Figure 21 Staff view of the menu

Figure 20 Admin view of the menu

When a new user register in the application , the system will assign the role of “ new “ and he will not have access to any information. He needs to message the admin to gain access to different information.

A picture containing text, indoor

Description automatically generated

Figure 22 Page where all the new users with no role are redirected

##### Implement the analytics section

The analytics section is a crucial component of the application, as it provides information about the revenue of the clinic, bonuses to be paid, etc. For this section I used Plotly.JS, a popular JavaScript framework, to help with the graphs and charts.

First , I had to fetch the data I was interested in from the database using the appropriate queries and send it to the front-end where , with the help of HTML, CSS, JavaScript and Plotly.JS, I have managed to render the charts I needed.

A screenshot of a computer

Description automatically generated with low confidence

Figure 23 Example with the Analytics page

##### Implement the Contact us section

Getting back the feedback from customers is one important feature that almost every single web application has at the moment. Feedback can help the practice identify the parts of workflow , where they can improve themselves.

A screenshot of a computer

Description automatically generated with medium confidenceUsing FormSubmit, a form management backend platform, I have managed to handle the form submission, by redirecting the content to the company’s email. All the information written in the form will automatically be redirected to [contact.zentist@gmail.com](mailto:contact.zentist@gmail.com) . An example is shown below.

Figure 24 Contact us page that takes the input from users and redirects it to practice email

A screenshot of a computer

Description automatically generated with low confidence

Figure 25 Email received

##### Implement extra features

Other features such as Invoice generator, Prescription generator, Consent form, Admin Dashboard, Future Appointments were added and provides extra features to the ones planned at the beginning of the project and more information were added to the [appendix](#_29._edit_patient) section. For invoice and prescription generators, several new libraries were used, such as HTML2Pdf.js which is a library that easily converts parts of the HTML content into PDF documents. This library was used for both , Invoice Generator feature and for Prescription Generator. These two features will add extra functionality to the web app which will make the practices more willingly to use Zentist in their offices as the main customer management platform.

A screenshot of a computer screen

Description automatically generated with low confidence

Figure 26 Invoice example

A screenshot of a computer

Description automatically generated with medium confidence

Figure 27 Prescription example

## 4.3 Evaluation

The aim of the evaluation section of this project is to assess the success of the Zentist Web app implementation in meeting its objectives and requirements. As part of this process, it is also essential to analyse relevant software in the dental industry that provide the same range of services in order to discover best practises, features to be introduced in the future, and how Zentist compares to similar applications. The subsection of this section will discuss the success of the implementation (Customer Feedback) and also how the application compares to similar products on the market.

As part of the evaluation , me, and the customer, decided to showcase the application to the same professionals that gave me the insights for the project, and ask for their feedback through an online survey. Their responses will be very helpful for the future development of the application and will show me if the application is something they would use, how can I update the application ,what features needs to be added , what application integration needs to be added.

The survey consisted in five questions that will try to uncover the following information: user web applications usage background, web app workflow, users’ opinion about features that need to be implemented, application integration preferences and features to be implemented.

Chart, diagram

Description automatically generated with medium confidenceAccording to the survey, more than 70% of respondents said they would consider using the application as it is now, which, combined with the fact that 57% of them had never used a web application before, gives us a good idea of the application's engagement rate among dental practise professionals.

Figure 28 Graph with results of question 3 of the user feedback after they used the application

Another key component of the survey was to determine which additional platforms should be integrated into our application to make it more appealing to businesses, as some of them can bring notable features into the Zentist.

Chart, bar chart

Description automatically generated

Figure 29 Integration preferences

A picture containing text, font, logo, circle

Description automatically generated Also , I used Lighthouse ,Chrome’s automated tool that helped me improve the quality of the web pages. It runs a series of tests against a web page and generates a report that scores the pages against various metrics like , performance ,accessibility ,best practices and SEO. It also provides guidance on how to improve each metric. There are a few things that should be changed in the future , but for a prototype the results are good.

## 4.4 Related Work

One key area of the related work section is other dental practice management software. After researching the internet , the most used application on the patient management market, are the following : Dentrix, Eagle soft and Open Dental. After viewing their websites, I concluded that generally they offer the same features, including appointment scheduling , patient record management, billing , reporting and analytics. However, being longer on the market ,having a larger team overseeing the application and having a budget to spend on market analysis offer them more information about what are the daily needs of the practices. Some products, for example, are more focused on larger practises or have more extensive reporting and analytics tools, but as I indicated at the beginning of the report, this application is only a prototype that demonstrates an idea and what can be acquired with a solid development team behind it.

Another thing I discovered while investigating other applications is that in order to remain competitive and effective, the application need a team to provide periodic optimisation and maintenance. Bellow you will find a list of reasons why an optimization and maintenance are required:

* Performance:

As the user database and data volumes grow, the application’s performance can begin to slow down.

* Security:

As the cyber threats and data breaches become increasingly common , periodical optimization can prevent this.

* Bug Fixes:

Even the most rigorously tested application may still have bugs that requires fixing. Regular maintenance can prevent crashes and other issues.

* User Feedback:

As users interact with the application , they might provide valuable feedback on areas that needs to be improved.

In terms of application performance compared to other products on the market, I found that Zentist is competitive. The application offers a range of useful features including customer records management, invoice management, sales analytics, online pre-booking, online feedbacks, admin dashboard ,staff contact agenda. The interface is intuitive and easy to navigate and performs well in terms of speed and responsiveness. However , there are some areas where the app could be improved. For example, while the app provides some basic security features such as protection against SQL injection and cross-site scripting, more security measures must be built to secure that user's data in the future.

Finally, after evaluating the application by comparing it to other applications, receiving feedback from professionals, and also reviewing the list of objectives provided at the start of the project, I concluded that Zentist implementation is successful, meets all of the set objectives, and performs well in all key areas. A team of developers, however, is required to take over and move the application from the prototype stage to the deployment stage..

# **5.Conclusion**

## 5.1 Reflection

The development of the Zentist Web App has been a challenge and a rewarding project at the same time. Through this project, I have achieved this project’s aims and objectives by creating the UML diagram of the application, have a costumer consultation to identify the key elements of a dental web application and also creating a customer record manager, invoice manager, sales analytics, online pre-booking, online feedback form, the admin dashboard, login system in terms of application’s features.

Throughout the development of this project, I was able to expand my knowledge of several programming languages, including JavaScript, Node.js, Express.js, HTML, CSS, MySQL, and others, while also increasing my confidence in working as a full-stack developer.

When developing the application, I had to be aware of the legal, social, ethical, and professional considerations that can affect the development of a web application. I ensured that Zentist complies with data privacy laws and considered the ethical implications of its design and potential use.

While reflection on the development process, I recognize there are areas I could have done more and to improve the application, but as I said throughout this report, the application is in the prototype stage of development and needs a lot more attention to be a full-fledged CRM platform that can be used in dental practices. Even though this is only a prototype, considering the feedback received , the application has a lot of potential and it could be used to improve the workflow of a dental practice. Also, whilst working on this project , I had to discover some of the “ugly truths” of working as a full-stack developer.

1. **Planning before starting to code is particularly important**. Sometimes I was eager to start coding and see the results before I had a clear idea of what I wanted to create. This project taught me that I have to learn to walk ,before I learn to run.
2. **Not everything can be done in the given time**. As I was planning the steps that I had to finish this project, so many features came up to my mind and I was so eager to implement so many of them. But as the due to approached, I realised I am only human and putting too much stress on myself will not do any good and I might prevent from finishing the development of the application. So, I had to decide to work on fewer features, but put more focus into it.
3. **Asking for help**. As I started the development of this application, I was confident that I will be able to finish everything without any external help. Being stubborn and also having a strong personality that does not like to ask for help, I had to struggle a lot before I learned a very valuable lesson. As the time went by, I realised that the fastest way to fix / solve a problem is to ask for help. Teachers, colleagues , IT professionals they were all ready to help, I just had to ask for it.
4. **To grow you have to invest time**. When tackling different areas of the project, I realized I do not have the required knowledge to solve the problems, so I had to invest time into reading, codding, failing, and trying again in order to understand the concept and how to fix them. If one wants to succeed in this profession, he needs to learn to invest time without being frustrated. Growing requires time.
5. **Use less words to express ideas**. One of the unfortunes of working on a full-stack project is that when doing documentation or writing essays, there are so many areas of the project where you can go in depth and also so many things to write about each framework , library, coding style, etc. This project taught me understand how to say more with less words.

## 5.2 Future Work

Since the Zentist Web Application, is only a prototype, several features can be implemented in order to take to the next stage of development.

1. Integration of additional features:
   1. SMS patient reminder
   2. Online payment
   3. Appointment overlapping check
   4. Bi-lingual website
2. Mobile application development
3. Improved user interface
4. Expanding to other practices
5. Web deployment
6. Integration with other dental software
7. Improvement of data analysis section
8. Tele-dentistry
9. Car park registration
10. Treatment plan tracking

This is a small list of features that could be added to the application to make the work of dental practise employees easier and Zentist a more widely utilised application.

# **6.References**

[1] Rognehaugh, R. (1999) *The Health Management Information Technology Dictionary*. Gaithersburg, MD: Aspen Publishers.

[2] “Advantages and challenges of digitization in healthcare,” *Digital Salutem*, 30-Mar-2022. [Online]. Available: <https://digitalsalutem.com/advantages-and-challenges-of-digitization-in-healthcare/>. [Accessed: 21-Jan-2023].

[3] “Are paper or electronic medical records better?” *business.com*. [Online]. Available: <https://www.business.com/articles/paper-vs-electronic-medical-records/>. [Accessed: 21-Jan-2023].

[4] “Pros and cons of paper based medical records,” *TrueNorth ITG*, 01-Nov-2022. [Online]. Available: <https://www.truenorthitg.com/pros-and-cons-paper-medical-records/>. [Accessed: 21-Jan-2023].

[5] C. Rees, “A four step approach to capturing customer requirements,” *A Four Step Approach to Capturing Customer Requirements*. [Online]. Available: <https://sigmapro.co.uk/blog/7-a-four-step-approach-to-capturing-customer-requirements>. [Accessed: 23-Jan-2023].

[6] “What are the advantages of Electronic Health Records?,” *What are the advantages of electronic health records? | HealthIT.gov*, 08-Mar-2022. [Online]. Available: <https://www.healthit.gov/faq/what-are-advantages-electronic-health-records>. [Accessed: 22-Mar-2023].

[7] J. Finnegan, “Report outlines the business perils of Long Patient Wait Times,” *Fierce Healthcare*, 22-Mar-2018. [Online]. Available: <https://www.fiercehealthcare.com/practices/ppatients-switched-doctors-long-wait-times-vitals>. [Accessed: 20-Mar-2023].

[8] P. Kriss, “The value of customer experience, quantified,” *Harvard Business Review*, 05-Nov-2014. [Online]. Available: <https://hbr.org/2014/08/the-value-of-customer-experience-quantified>. [Accessed: 01-Apr-2023].

[9] J. Tsai and G. Bond, “Comparison of electronic records to paper records in Mental Health Centres,” *OUP Academic*,12-Dec-2007.[Online].Available: <https://academic.oup.com/intqhc/article/20/2/136/1785685>. [Accessed: 02-Apr-2023].

[10] Stausberg, J. *et al.* (2003) “Comparing paper-based with electronic patient records: Lessons learned during a study on diagnosis and procedure codes,” *Journal of the American Medical Informatics Association*, 10(5), pp. 470–477. Available at: <https://doi.org/10.1197/jamia.m1290>.

[11] “Guide to the UK General Data Protection Regulation (UK GDPR),” *ICO*. [Online]. Available: <https://ico.org.uk/for-organisations/guide-to-data-protection/guide-to-the-general-data-protection-regulation-gdpr/>. [Accessed: 03-Apr-2023].

[12] *Informed* *consent* *AMA*. Available at: <https://code-medical-ethics.ama-assn.org/ethics-opinions/informed-consent> (Accessed: April 12, 2023).

[13] Rees, D. (2020) *What is wireframing*, *Experience UX*. Available at: <https://www.experienceux.co.uk/faqs/what-is-wireframing/> (Accessed: April 17, 2023).

[14] Sathyendra, S. and Nahyan, M. (no date) *Digitalisation of Dentistry: A pandemic influence - IJSR*, *International Journal of Science and Research*. Available at: <https://www.ijsr.net/archive/v10i10/SR211012154200.pdf> (Accessed: March 7, 2023).

[15] Stahl, B.C., Timmermans, J. and Mittelstadt, B.D. (2016) “The ethics of computing,” *ACM Computing Surveys*, 48(4), pp. 1–38. Available at: <https://doi.org/10.1145/2871196>.

[16] *What is Agile Software Development?* (2023) *Agile Alliance |*. Available at: [https://www.agilealliance.org/agile101/](https://www.agilealliance.org/agile101/%20) (Accessed: 12 April 2023).

# **7.Appendices**

## 7.1 Motivation

The motivation behind this project, is to help dental professionals to improve their efficiency and their productivity by transitioning to computer-based systems. By providing a dental application that is user-friendly and free, I hope to make the transition easier and more accessible for dental practices of all sizes.

Also, personally having family members that work in the dental industry gave me a good insight about the struggles they need to face on daily basis. Their feedback and their colleague’s feedback gave me valuable information about the needs of the staff and what are the most important features such an application should have.

## 7.2 Routes

In Node.Js , routes are used to define different URLs or endpoints that the application can use. When user requests a particular URL, the application checks to see what code should be executed.

The ‘**req’**  and ‘**res’** objects are commonly used in Node.js to represent the request and the outgoing HTTP response. The ‘**req**’ object contains information about the incoming request such as the URL, headers or any data sent in the request body. The ‘**res**’ object is used to send back a response such as setting header, status code or data back in response body.

There are few methods used along res object such as redirect, send, render etc.

The ‘**redirect**’ method redirects the user to a specific URL. This is often used after a successful form submission or if user is trying to access a restricted area. The method does not accept other arguments than the page itself.

The ‘**render**’ method is used to render a specific URL but can also receive other objects as arguments to be sent to the front end. Arrays, objects, variables can be sent to the front end through this method.

## 7.3 TeamWork report

Bellow you can read the report generated by TeamWork to see details of how the project went.

Table 1 Double-click to see the full report

## 

## 7.4 Figures

### 1. Survey question 1

Chart, sunburst chart

Description automatically generated

Appendix-figure 1 Survey question no. 1

### 2.Survey question 2

A picture containing diagram

Description automatically generated

Appendix-figure 2 Survey question 2

### Survey question 3

Chart

Description automatically generated

Appendix-figure 3 Survey Question 3

A picture containing chart

Description automatically generated

Appendix-figure 4 Survey Question 4

### 5.Survey question 5

A picture containing diagram

Description automatically generated

Appendix-figure 5 Survey Question 5

### 6.Survey question 6

Chart

Description automatically generated

Appendix-figure 6 Survey Question 6

### 7.Survey question 7

A picture containing chart

Description automatically generated

Appendix-figure 7 Survey Question 7

### 8.Survey question 8

A picture containing graphical user interface

Description automatically generated

Appendix-figure 8 Survey Question 8

### 9.Survey question 9

Chart, bar chart

Description automatically generated

Appendix-figure 9 Survey Question 9

### TeamWork dashboard

Graphical user interface, text, application

Description automatically generated

Appendix-figure 10 TeamWork dashboard

### Gantt Chart

Timeline

Description automatically generated

Appendix-figure 11 Gantt chart of project management

### NPM dependencies

Text

Description automatically generated

Appendix-figure 12 Dependencies package

### Graphical user interface Description automatically generatedlogin page wireframe

Appendix-figure 13 Wireframe of the login page

### 15.Sitemap

Diagram

Description automatically generated

Appendix-figure 14 Initial sitemap

### 16. Final survey question 1

Chart

Description automatically generated with medium confidence

Appendix-figure 15 Project final survey question 1

### 17. Final survey question 2

Diagram

Description automatically generated

Appendix-figure 16 Project final survey question 2

### 18. Final survey question 3

A picture containing chart

Description automatically generated

Appendix-figure 17 Project final survey question 3

### 19.Final survey question 4

Table

Description automatically generated

Appendix-figure 18 Project final survey question 4

### 20 .Final survey question 5

Chart, bar chart

Description automatically generated

Appendix-figure 19 Project final survey question 5

### 21. Final survey question 6

Chart, bar chart

Description automatically generated

Appendix-figure 20 Project final survey question 6

### 22. customer consultation question part 1

Graphical user interface, application, table

Description automatically generated

Appendix-figure 21 Survey Form part1

### 23. customer consultation question part 2

Graphical user interface

Description automatically generated

Appendix-figure 22 Survey Form part2

### 24. customer consultation question part 3

Graphical user interface, application

Description automatically generated

Appendix-figure 23 Survey Form part 3

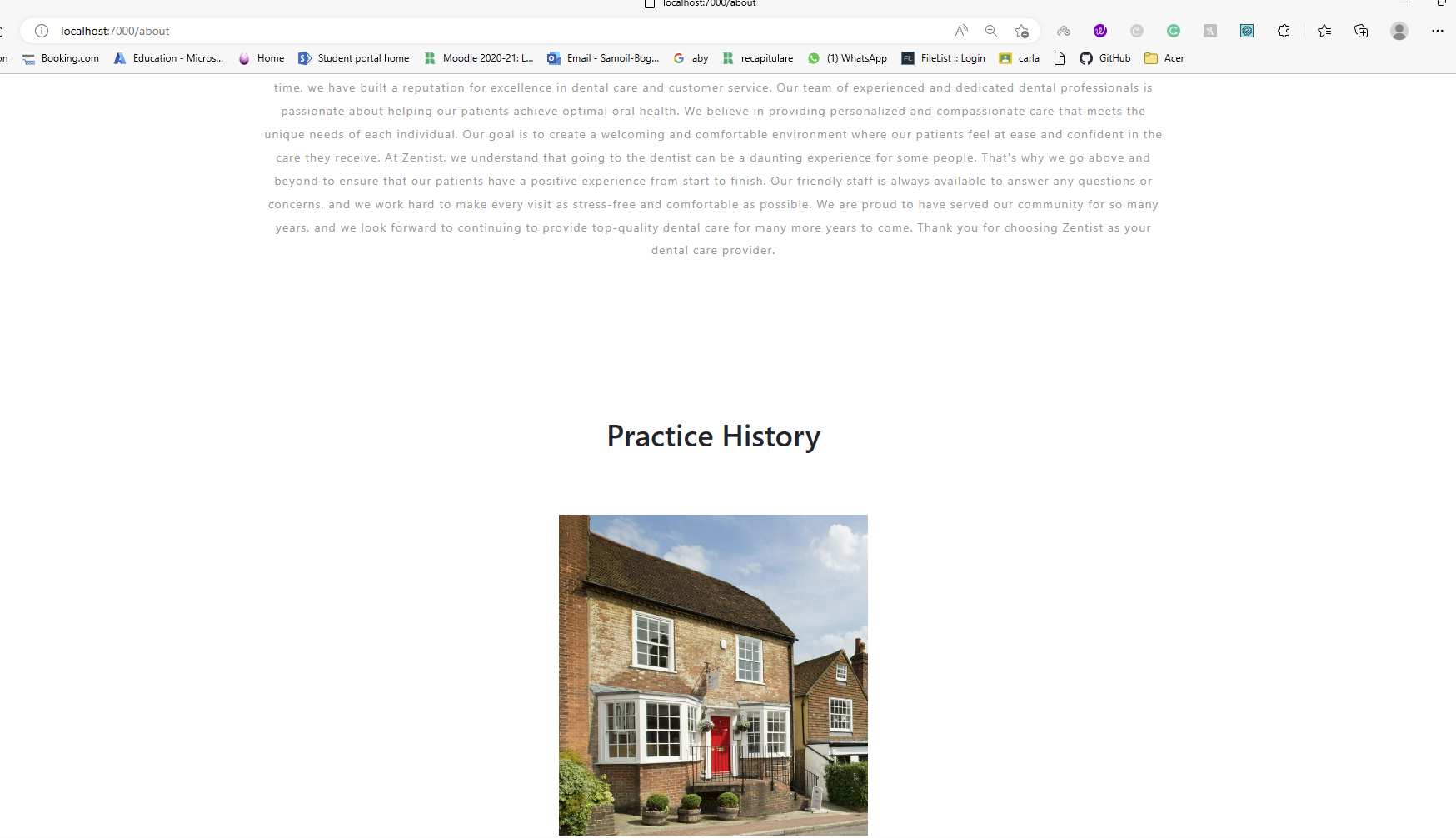
### 25. application- user dashboard wireframe

A screenshot of a computer

Description automatically generated

Appendix-figure 24 Wireframe of user dashboard

### 26. About page implemenation

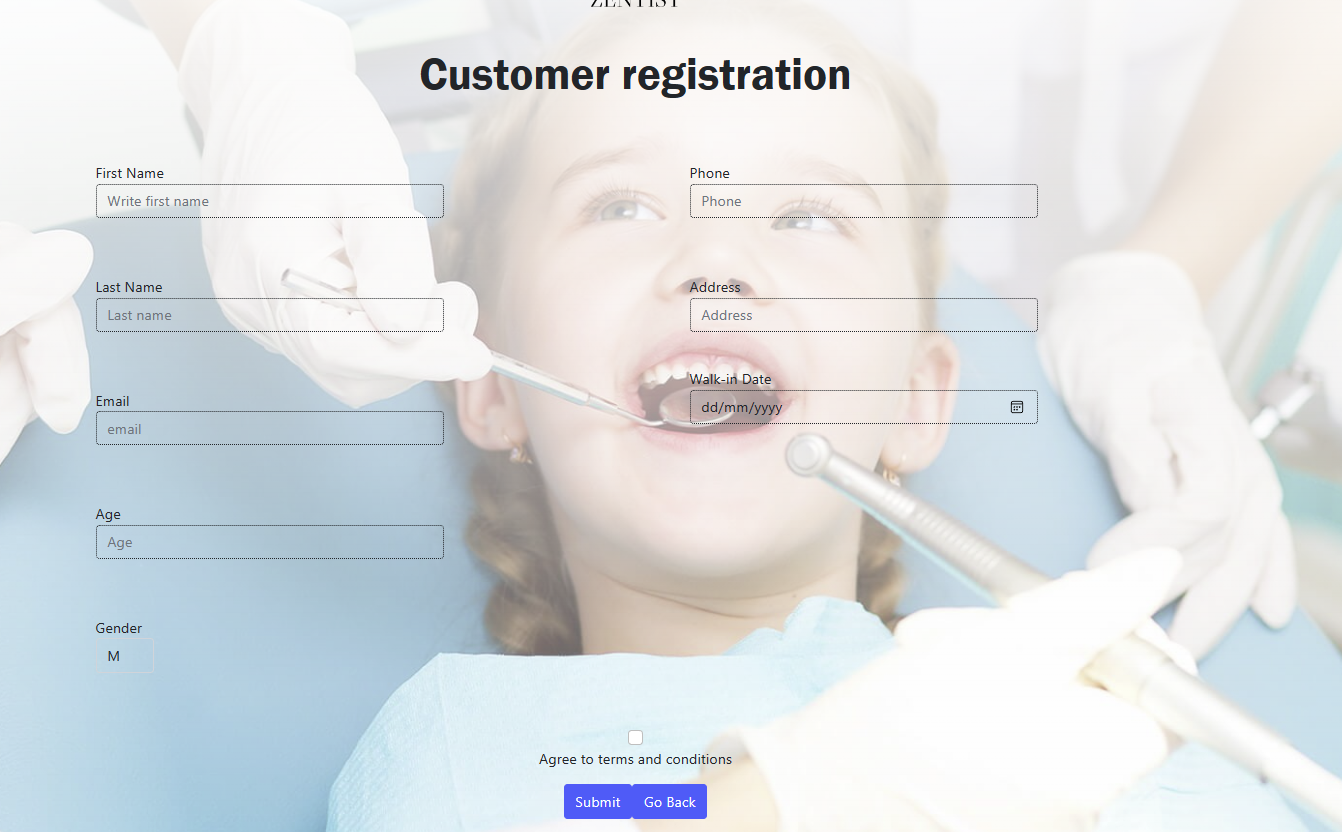


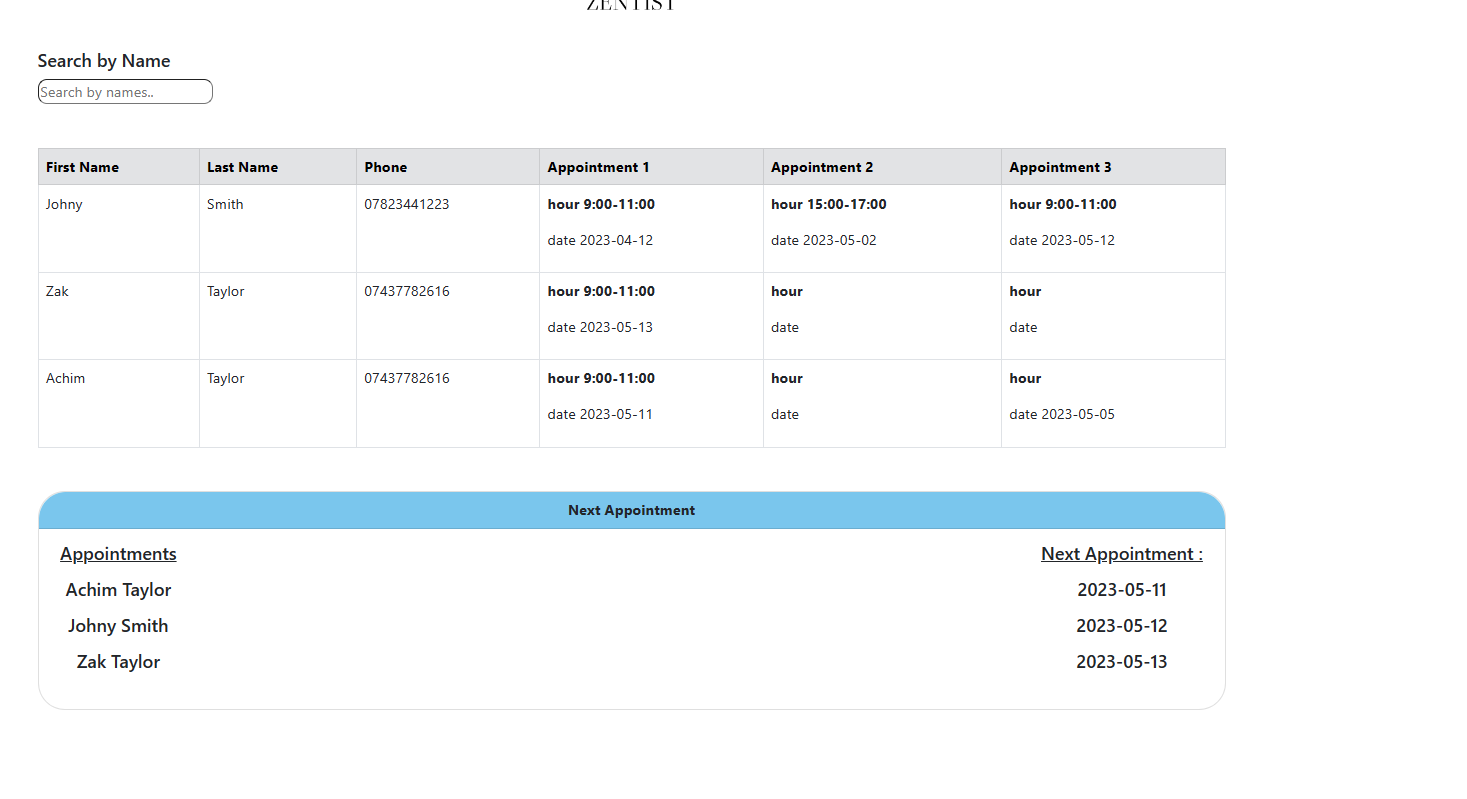
Appendix-figure 25 About section

Appendix-figure 26 add client backend

### 27. add client post backend

### 28. customer registration

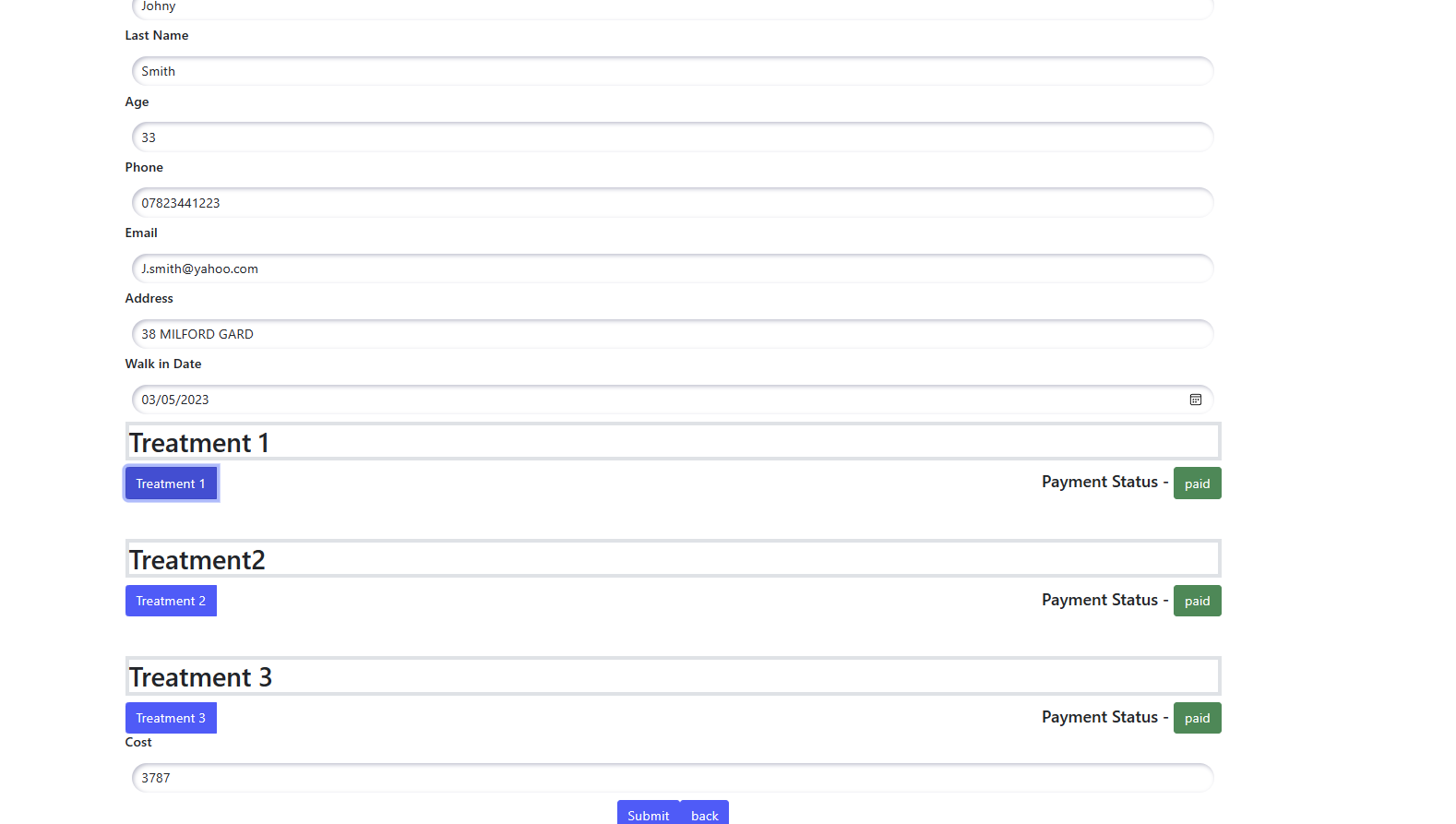


Appendix-figure 27 front end add client

Appendix-figure 28 appointments page

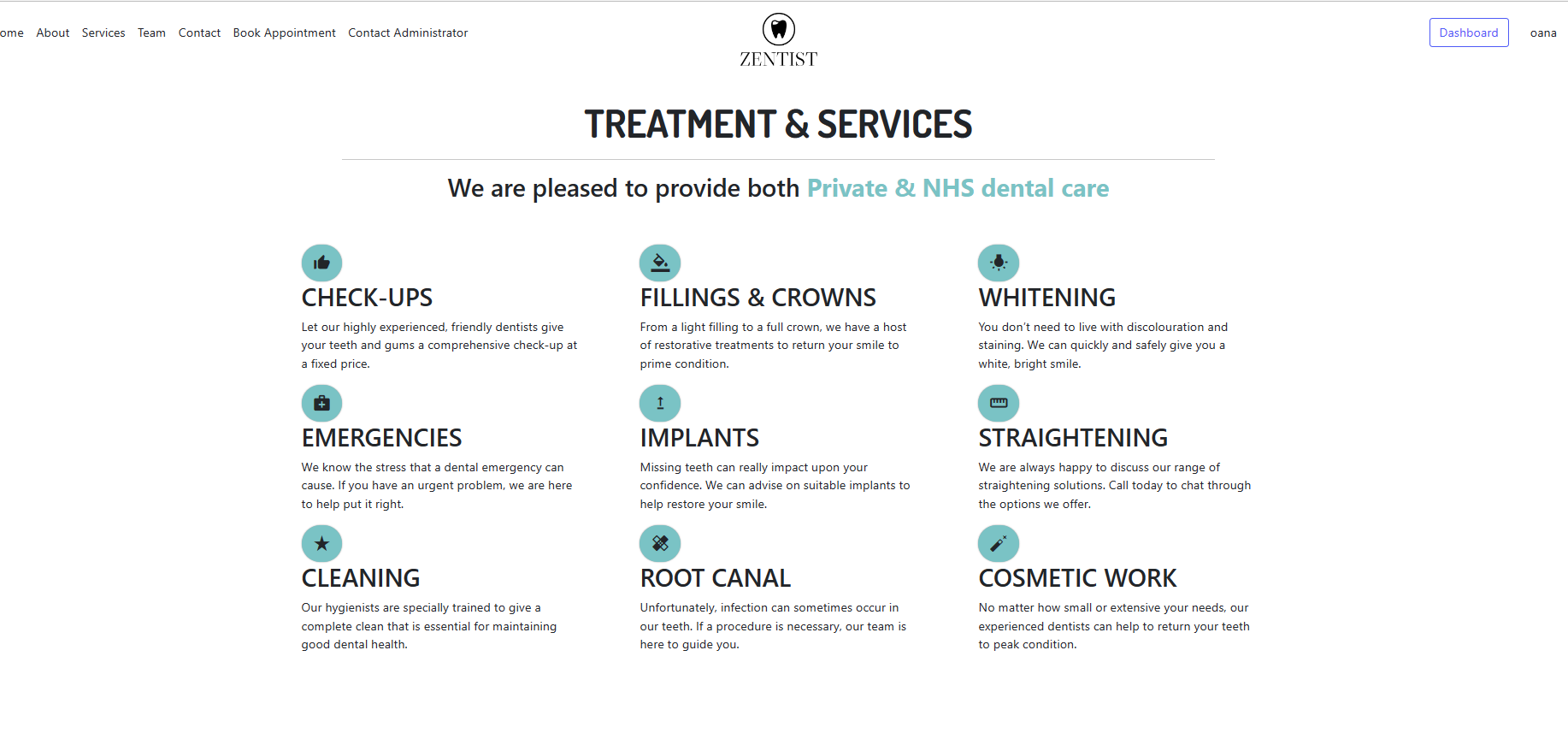
### 29. appointment section

### 29. edit patient



Appendix-figure 29 edit client

### 30. services page



Appendix-figure 30 Services page