## horizontal line



El-Care (Final Year Project)

09-05-2023

**─**

Amrita Giri

D00226038

Dundalk Institute of Technology

BSc (Hons) Computing in Cloud Computing

Supervisor : Tony Mc Carron

Second Reader : Dr Peadar Grant

# ACKNOWLEDGEMENTS

I would like to express my deepest gratitude to my supervisor, Prof. Tony Mc Carron and my second reader Dr. Peadar Grant, for their unwavering support, guidance, and invaluable insights throughout the process of writing this thesis and yearlong project.

Finally, I am forever indebted to my family , my classmates’ and close friends for their unwavering love, support, and understanding throughout my academic pursuits.

Best Wishes

Table of Contents

[horizontal line 0](#_Toc134554315)

[ACKNOWLEDGEMENTS 1](#_Toc134554316)

[ABSTRACT 5](#_Toc134554317)

[LITERATURE REVIEW 6](#_Toc134554318)

[Project Aim & Objective 6](#_Toc134554319)

[Market Research 6](#_Toc134554320)

[Market Analysis 6](#_Toc134554321)

[Competitor Analysis 6](#_Toc134554322)

[User Needs Assessment 7](#_Toc134554323)

[Industry Trends 7](#_Toc134554324)

[Regulatory Environment 7](#_Toc134554325)

[SWOT Analysis 7](#_Toc134554326)

[Market Entry Strategy 7](#_Toc134554327)

[Why Hosting the Application by Cloud Computing for Any Project Beneficial ? 7](#_Toc134554328)

[Cost Savings 8](#_Toc134554329)

[Security 8](#_Toc134554330)

[Flexibility 8](#_Toc134554331)

[Mobility 8](#_Toc134554332)

[Insight 8](#_Toc134554333)

[Rise in Collaboration 8](#_Toc134554334)

[Quality Management 8](#_Toc134554335)

[Disaster Prevention 9](#_Toc134554336)

[Loss Recovery 9](#_Toc134554337)

[Automatic Software Updating and Maintenance 9](#_Toc134554338)

[Competitive Edge 9](#_Toc134554339)

[Sustainability 9](#_Toc134554340)

[Scalability 9](#_Toc134554341)

[Reliability 9](#_Toc134554342)

[Global Reach 9](#_Toc134554343)

[Integration and Collaboration 10](#_Toc134554344)

[Models of Cloud Services 10](#_Toc134554345)

[IAAS (Infrastructure-as-a-Service) 10](#_Toc134554346)

[PAAS (Platform-as-a-Service) 10](#_Toc134554347)

[SAAS (Software-as-a-Service) 10](#_Toc134554348)

[FAAS(Function-as-a-Service) 11](#_Toc134554349)

[Types of Cloud 11](#_Toc134554350)

[Public 11](#_Toc134554351)

[Private 11](#_Toc134554352)

[Hybrid 12](#_Toc134554353)

[Cloud Use Cases 12](#_Toc134554354)

[ Edge computing and IoT 12](#_Toc134554355)

[ AI and machine learning 12](#_Toc134554356)

[ Serverless and event-driven architectures 12](#_Toc134554357)

[ Quantum computing 12](#_Toc134554358)

[ 5G and cloud-native networking 12](#_Toc134554359)

[ Decentralized and distributed cloud 13](#_Toc134554360)

[ Enhanced security and privacy 13](#_Toc134554361)

[ Green and sustainable cloud computing 13](#_Toc134554362)

[ Industry-specific cloud solutions 13](#_Toc134554363)

[Amazon Web Services 13](#_Toc134554364)

[Google Cloud Platforms 14](#_Toc134554365)

[Microsoft Azure 14](#_Toc134554366)

[Azure vs AWS vs GCP 14](#_Toc134554367)

[Market share 15](#_Toc134554368)

[Services 15](#_Toc134554369)

[Pricing 16](#_Toc134554370)

[Compliance and security 16](#_Toc134554371)

[Hybrid cloud capabilities 16](#_Toc134554372)

[Mobile Application Platforms 16](#_Toc134554373)

[Introduction 16](#_Toc134554374)

[Native Platforms 17](#_Toc134554375)

[Cross-platform Frameworks 18](#_Toc134554376)

[Hybrid App Frameworks 19](#_Toc134554377)

[Web Application Frameworks 20](#_Toc134554378)

[Conclusions 21](#_Toc134554379)

[Hybrid vs Native vs WebApp vs Cross-Platform 21](#_Toc134554380)

[Database Technology 23](#_Toc134554381)

[Introduction 23](#_Toc134554382)

[Why Firebase MySQL MongoDB are the most widely used ? 24](#_Toc134554383)

[Firebase (Realtime Database and Firestore) 25](#_Toc134554384)

[MySQL 25](#_Toc134554385)

[MongoDB 26](#_Toc134554386)

[Conclusion 26](#_Toc134554387)

[Frontend Technology 27](#_Toc134554388)

[Introduction 27](#_Toc134554389)

[HTML (Hyper Text Markup Language) 27](#_Toc134554390)

[CSS (Cascading Style Sheets) 27](#_Toc134554391)

[JavaScript 27](#_Toc134554392)

[Responsive Design 27](#_Toc134554393)

[CSS Frameworks 27](#_Toc134554394)

[JavaScript Libraries & Frameworks 27](#_Toc134554395)

[CSS Preprocessors 27](#_Toc134554396)

[Build Tools and Task Runners 28](#_Toc134554397)

[Version Control Systems 28](#_Toc134554398)

[Testing and Debugging Tools 28](#_Toc134554399)

[Performance Optimization 28](#_Toc134554400)

[Conclusion 28](#_Toc134554401)

[Backend Technologies 28](#_Toc134554402)

[Introduction 28](#_Toc134554403)

[Programming languages 28](#_Toc134554404)

[Web frameworks 29](#_Toc134554405)

[Caching systems 30](#_Toc134554406)

[Message brokers 30](#_Toc134554407)

[Server and deployment technologies 30](#_Toc134554408)

[Conclusion 30](#_Toc134554409)

[Visual Studio Code 31](#_Toc134554410)

[Introduction 31](#_Toc134554411)

[Conclusion 32](#_Toc134554412)

[PROPOSED SOLUTION 32](#_Toc134554413)

[Database (MySQL & Firebase) 35](#_Toc134554414)

[Firebase 35](#_Toc134554415)

[MySQL 35](#_Toc134554416)

[Conclusion 36](#_Toc134554417)

[Frontend Technologies (Bootstrap and HTML CSS) 36](#_Toc134554418)

[Bootstrap 36](#_Toc134554419)

[HTML, CSS, and JavaScript 37](#_Toc134554420)

[Conclusion 37](#_Toc134554421)

[Backend Technologies 37](#_Toc134554422)

[PHP 37](#_Toc134554423)

[JavaScript (Node.js) 38](#_Toc134554424)

[Conclusion 39](#_Toc134554425)

[Cloud Service and Connections 39](#_Toc134554426)

[PLAN FOR PROGRESSION 39](#_Toc134554427)

[REFERENCES 40](#_Toc134554428)

# ABSTRACT

I have been volunteering in old age homes and disability schools since a very young age and noticed a lack of proper technological systems for these two sectors. I believe that these sectors should also benefit from advancements in technology and the digital era. We have identified common problems faced by individuals who are unable to take care of their parents while away in another city or part of the world. To solve this problem, I propose a project called El-Care, which would act as a proxy for individuals to have everything their loved ones need in their hands.

The El-Care project consists of two distinct parts: the front-end and the backend. The front-end is a web application that communicates with clients and develops information from the backend into a comprehensible configuration. The backend is a cloud-supported project that can read and write any type of data to the website. This is justified using various technologies such as JavaScript, Visual Studio, AWS Cloud, PHP, GitHub, HTML/CSS, and Bootstrap for the development of this project.

# LITERATURE REVIEW

## Project Aim & Objective

The primary aim of this project is to develop a user-friendly, accessible, and efficient online service booking platform specifically tailored for elderly individuals. This website will provide a comprehensive range of services and by centralizing these services in one platform, the project seeks to streamline the booking process, enhance communication between service providers and elderly users, and foster a supportive community for seniors to easily access and manage essential services, ultimately contributing to their well-being and social inclusion.

1. To conduct thorough research and identify the essential services required by the elderly population, taking into consideration their unique needs and preferences.
2. To design a user-friendly and intuitive interface for the elderly service booking platform, prioritizing accessibility, and ease of navigation for senior users with varying levels of digital literacy.
3. To develop a secure and reliable platform that ensures the privacy and safety of users' personal information and provides an efficient booking process for various services.
4. To implement user feedback mechanisms with chat systems that allow continuous improvement of the platform's features and functionality, based on users' experiences and suggestions.

## 

## Market Research

### Market Analysis

1. **Target Market:** mostly elderly individuals aged 65 and above, as well as their family members or caregivers who may be responsible for managing their daily needs.
2. **Market Size and Segmentation:** analyze population data, aging trends, geographic location, income levels, living arrangements, specific service requirements, and projected growth of the elderly population in the region or country.

### Competitor Analysis

1. **Identifying Major Competitors :** in the elderly service booking industry, both online and offline. Assess their market share, strengths, weaknesses, and unique selling propositions.
2. **Investigating their Business :** their pricing strategies, range of services, target audience, and user experience to identify gaps or opportunities for differentiation.

### User Needs Assessment

1. **User Information Gathering :**Conducting surveys, interviews, or focus groups with elderly individuals and their caregivers to understand their specific needs, preferences, and challenges when it comes to accessing and booking services.
2. **Gather Insights :** into their level of digital literacy, accessibility requirements, and the factors that influence their choice of service providers.

### Industry Trends

1. **Analyzing Trends:** current and emerging trends in the elderly care market, such as the increasing demand for home care services, the integration of technology in care provision, and the growing emphasis on personalized care.
2. **Assessing Impact:** the potential impact of these trends on the elderly service booking industry and identify opportunities for innovation or adaptation.

### Regulatory Environment

1. **Research Regulations:** relevant regulations and guidelines governing the elderly care industry, such as licensing requirements, data privacy, and safety standards.
2. **Ensuring Law Amendment:** that the proposed platform complies with all applicable laws and standards, and factors in any potential regulatory changes when developing the website.

### SWOT Analysis

1. **SWOT:** Identify the strengths, weaknesses, opportunities, and threats related to the proposed elderly service booking website.
2. **Data Analysis :** Use this analysis to inform the platform's development strategy, marketing approach, and risk mitigation plans.

### Market Entry Strategy

1. **Developing Market Strategy:** a market entry strategy, including the selection of an initial target market segment, pricing strategy, promotional tactics, and distribution channels.
2. **Potential Partnership :** Consider potential partnerships with relevant organizations, such as senior centers, healthcare providers, or community organizations, to facilitate a successful market entry and foster long-term growth.

## Why Hosting the Application by Cloud Computing for Any Project Beneficial ?

Cloud is the amalgamation of virtualization resources at a cloud data center that is run at big infrastructure level by leading cloud service providers in the world. The resources can vary ranging from servers (including physical, virtual and hardware servers) , data collection and storage , operating system software and networking and security. These are the services that can be used for a monthly subscription fee or just charges for services used.

Whether it is work from home in a laptop or a mobile a recent survey showed that 92% of the organizations today have been seen using all the services provided by cloud server providers. So, do we at home unknowingly with google Gmail, google drive , Netflix or drop box. [22]

### Cost Savings

Paying for exactly the space and resources used and needed showed result in lower costs and higher returns.

### Security

Statistics of Rapid Scale said that there was a 94% of businesses improvement and meeting requirements in security after organizations changed to the cloud. Encryption of data used amped up in cloud made it less accessible for hackers to view data. [27]

### Flexibility

Extra bandwidth is easier in cloud rather than expensive update in local server IT Infrastructure proving significant difference to the overall efficiency of many organizations. InformationWeek mentioned that 65% said they were more able to meet the business demands of their clients. [28]

### Mobility

Cloud computing allows our billion mobile users to stay in the loop of accessing corporate data via their smartphones making work from home a very safe work environment

### Insight

Our ability to critically see our cloud analytics of our data with an eagle’s eye hence making tracking and customized report analysis easier.

E.g.: a beverage company increased profits by about $2 million a year but also were able to cut down $195,000 in their staff cost just by introducing cloud technologies in their company .[28]

### Rise in Collaboration

Working in a team is made easier with cloud services because of teamwork social gatherings to connect employees across your organization, therefore increasing their pique and work engagement.

### Quality Management

In an organization where the large teams and team worker must access all kinds of data in between projects and project works this kind of data handling can always result in human error and this can be solved by having a clear set of data in a document with a consistent template in the cloud. This also helps in maintaining a template that is of same consistency on the cloud to avoid any mixing of work or confusion or dilution of data

### Disaster Prevention

This industry is extremely fragile in cases where having even a downtime can have some bad effects on productivity, revenue, and brand reputation. Hence cloud managed to provide a massive disaster recovery to the extent that many companies got saved from running into massive losses or business failures with clients.

### Loss Recovery

The risk of losing data is huge in today’s day and age especially the ones that are saved locally but the risk is reduced with cloud for all the information that can be uploaded online remains safe and can be accessed by any computer or laptop that has internet connection

### Automatic Software Updating and Maintenance

The automatic updating and refreshing of software and avoiding spending time and money on external experts. It was also seen that half of the world population used less external hardware and software appliances after using Cloud resources as a benefit over them

### Competitive Edge

It is always an edge over the competitors when you manage to learn a technology that is relatively new cause you have an edge learning with experience by the time they catch up. The advantage of this approach showed that 77% had competitive advantage and significance. [28]

### Sustainability

Hosting on cloud resulted in carbon foot printing being reduced and there was a wastefulness at almost every level of a business proving to be more environment friendly

### Scalability

Cloud computing allows the website to easily scale its resources up or down based on demand.

### Reliability

Cloud service providers offer robust infrastructures with multiple layers of redundancy, ensuring high availability and minimizing downtime.

### Global Reach

Cloud hosting allows the website to be easily accessible from anywhere in the world, ensuring that elderly users and their caregivers can access the platform and book services regardless of their location.

### Integration and Collaboration

Cloud computing makes it easier to integrate the website with other software, tools, and applications, such as customer relationship management (CRM) systems, billing software, and data analytics tools. This can help streamline operations, improve communication with service providers, and enhance the overall user experience.

## Models of Cloud Services

### IAAS (Infrastructure-as-a-Service)

1. Infrastructure as a Service (IaaS) is a cloud computing service model that provides virtualized computing resources over the internet.
2. In an IaaS model, a cloud provider offers virtual machines (VMs), storage, networking, and other necessary computing infrastructure components to customers on a pay-as-you-go basis.
3. Some well-known IaaS providers include Amazon Web Services (AWS), Microsoft Azure, and Google Cloud Platform (GCP).
4. These providers offer a wide range of infrastructure services and tools that can be customized to suit the specific needs of a business or organization.

### PAAS (Platform-as-a-Service)

1. Platform as a Service (PaaS) is a cloud computing service model that provides a platform for developers to build, deploy, and manage applications without having to deal with the underlying infrastructure complexities.
2. PaaS offerings typically include tools and services for developing, testing, deploying, and maintaining applications.
3. Some well-known PaaS providers include Heroku, Google App Engine, Microsoft Azure App Service, and IBM Cloud Foundry.
4. These platforms offer various features and tools that cater to different programming languages and frameworks, making it easier for developers to build, deploy, and scale their applications.

### SAAS (Software-as-a-Service)

1. Software as a Service (SaaS) is a cloud computing service model that delivers software applications over the internet, eliminating the need for users to install and maintain the software on their local devices.
2. SaaS providers host, maintain, and manage the software, and users access it through a web browser, usually via a subscription model.
3. SaaS offers several benefits like Lower upfront costs, Easy access, Scalability, Automatic updates, Simplified integration
4. Some well-known SaaS examples include Salesforce (a customer relationship management platform), Microsoft Office 365 (productivity suite), Google Workspace (collaboration and productivity tools), and Slack (team communication and collaboration).

### FAAS(Function-as-a-Service)

1. Serverless computing, also known as Function as a Service (FaaS), is a cloud computing paradigm that allows developers to build and run applications without having to manage the underlying infrastructure, such as servers.
2. In a serverless environment, developers focus on writing individual functions or pieces of code that perform specific tasks, while the cloud provider takes care of the operational aspects like scaling, patching, and capacity management.
3. Popular serverless computing platforms include AWS Lambda, Google Cloud Functions, Microsoft Azure Functions, and IBM Cloud Functions.
4. These platforms support various programming languages and provide integration with other services within their respective ecosystems, enabling developers to build complex and efficient serverless applications.

## Types of Cloud

### Public

1. A public cloud is a cloud computing model where computing resources and services are offered over the internet to the public by a service provider on a pay as you go basis
2. Public cloud providers use a multi-tenant architecture where multiple users or organizations share the same underlying infrastructure and are offered as IaaS, PaaS, or SaaS based on the customer's requirements.
3. Some of the key benefits of public cloud include cost efficiency, scalability, flexibility, and reliability.
4. Well-known public cloud providers such as AWS, Azure, and GCP offer a wide range of services from computing and storage to machine learning and analytics, catering to various industries and use cases.

### Private

1. A private cloud is a cloud computing model where computing resources and infrastructure are dedicated to a single organization or user
2. They provide greater control, customization, security, and predictable performance.
3. Private clouds can be hosted on-premises or off-site by a third-party provider.
4. However, private clouds require higher upfront investment and ongoing operational expenses, have limited scalability, and increased management responsibility.

### Hybrid

1. A hybrid cloud is a cloud computing model that combines the use of public and private clouds, offering flexibility, scalability, cost efficiency, security, and improved business continuity.
2. In a hybrid cloud environment, data and applications can be shared between public and private cloud resources, connected via secure networking.
3. However, implementing a hybrid cloud strategy can also come with challenges such as increased complexity, data transfer and latency, and compliance and security.
4. Organizations should evaluate their specific requirements and workloads to determine the right balance between public and private cloud resources.

## Cloud Use Cases

As cloud computing continues to evolve, new use cases and applications will emerge. Here are some potential future cloud use cases:

### Edge computing and IoT

Edge computing will become more important to process data closer to the source as IoT devices grow, allowing for real-time data processing and analytics.

### AI and machine learning

Cloud providers will expand their offerings for AI and machine learning, providing access to powerful algorithms and tools to enable businesses to leverage AI.

### Serverless and event-driven architectures

Serverless computing and event-driven architectures will grow, enabling organizations to build efficient and scalable applications with minimal infrastructure management overhead.

### Quantum computing

Quantum computing may mature to allow cloud providers to offer quantum computing resources to customers, enabling them to solve complex problems currently intractable for classical computers.

### 5G and cloud-native networking

The rollout of 5G networks will facilitate faster data transfer and improved connectivity, enabling new cloud use cases.

### Decentralized and distributed cloud

Decentralized and distributed cloud solutions may gain traction as alternatives to centralized cloud providers, offering increased resilience, privacy, and control over data.

### Enhanced security and privacy

Cloud providers will develop new tools and services to help organizations protect their data and comply with regulations.

### Green and sustainable cloud computing

Cloud providers will focus on sustainable and energy-efficient solutions, such as utilizing renewable energy sources and developing energy-efficient data centers.

### Industry-specific cloud solutions

Cloud providers will develop more industry-specific solutions tailored to the unique needs and requirements of sectors such as healthcare, finance, manufacturing, and education.

Cloud computing will continue to play a critical role in enabling organizations to innovate, scale, and adapt to changing business environments as technology evolves.

## Amazon Web Services

1. Amazon Web Services (AWS) is a cloud computing platform offered by Amazon that provides a broad range of services and tools for organizations to build, deploy, and manage applications.
2. AWS offers Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS) to suit customers' specific needs.
3. It has a global infrastructure of data centers and availability zones across regions worldwide, enabling organizations to build highly available and scalable applications while minimizing latency for end-users.
4. AWS offers services such as computing, storage, databases, networking, security, identity, analytics, machine learning and AI, and developer tools, catering to various industries.

## Google Cloud Platforms

1. Google Cloud Platform (GCP) is a suite of cloud computing services that offers a range of infrastructure, platform, and software solutions for organizations to build, deploy, and manage applications.
2. GCP offers key services and features such as Google Compute Engine, Google Kubernetes Engine, Cloud Functions, Google Cloud Storage, Persistent Disk, Cloud SQL, Cloud Fire store, and Cloud Load Balancing.
3. GCP also provides a range of security and identity services such as Google Cloud Identity and Access Management, Cloud Key Management Service, and Cloud Security Command Center.
4. GCP is built on the same infrastructure that powers Google's own services, making it a reliable and powerful cloud computing platform.

## Microsoft Azure

1. Microsoft Azure is a cloud computing platform offered by Microsoft, which provides a wide range of services and tools for building, deploying, and managing applications.
2. Azure offers Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS) to cater to different customer needs.
3. It has a global infrastructure with data centers and availability zones in multiple regions, enabling organizations to build highly available and scalable applications.
4. Some of the key services and features provided by Azure include computing services, storage services, database services, networking services, security and identity services, analytics services, machine learning and AI services, and developer tools.

## Azure vs AWS vs GCP

|  |  |  |  |
| --- | --- | --- | --- |
| **Feature** | **GCP** | **AWS** | **Azure** |
| **Containerization** | Kubernetes (orchestration) | Relatively new | Offers platforms but not as good as GCP |
| **Market Position** | 3rd on market share | On top due to services provided | 2nd in market share |
| **Support Material** | Mastery possible with documentation/ videos | Good practical training needed to master AWS | No proper documentation hence self-study needed to master Azure |
| **Offerings** | 2nd option than AWS as both offer similar platforms | Services and support given to large organization | Companies only prefer azure for windows or can do it without outside help |
| **Global Marketing** | Global outreach is comparatively less as they don’t market as much | Due to marketing efforts, it known as the best cloud platform by large and small organizations | Azure is known but used less because AWS is a competitor. AWS policies are easier to follow than Azures |
| **Integration** | Gmail, YouTube and all other google services for seamless experience with cloud services | Users are happy with services also they are rendered | Microsoft integrated tools and software |
| **Open Sourced** | Not open sourced but some services are free. Provides portability of its services | Not open sourced need to pay as you use described accordingly | Open Sourced so users use it |
| **Tools Management** | Managed well but less services | Effective tool management and very vast services offered | Not proper management of tools |
| **Designing** | native environment design, offer/discounts/flexible contracts offered | For large storage and network usage | Big data and ML |
| **Focus Area** | DevOps with Docker and Kubernetes | different services on different platforms | Many features set mainly focusing on hybrid cloud |

AWS, GCP, and Azure are three leading cloud computing platforms, each with its own strengths and offerings. Here is a comparison of the three in terms of market share, services, and other factors:

### Market share

1. AWS is the market leader with the largest share and has been in the cloud computing space the longest, providing them with a more mature and extensive set of services.
2. Azure is the second-largest player, benefiting from Microsoft's existing enterprise customer base and integration with other Microsoft products.
3. GCP is the third-largest platform, known for its innovation and focus on developer-friendly tools, as well as its machine learning and data analytics capabilities.

### Services

1. AWS has the broadest range of services and features, which can be beneficial for organizations looking for the most extensive set of options.
2. Azure's tight integration with other Microsoft products and services can be advantageous for organizations already using Microsoft solutions.
3. GCP is particularly strong in data analytics and machine learning, leveraging Google's expertise in these areas.

### Pricing

1. All three platforms use a pay-as-you-go pricing model, where customers pay only for the resources they consume.
2. AWS and Azure offer similar pricing structures, while GCP tends to be slightly more cost-effective for some services, particularly in data storage and transfer.
3. All three platforms provide various cost optimization tools and options, such as reserved instances, committed use contracts, or savings plans.

### Compliance and security

1. AWS, GCP, and Azure all prioritize security and compliance, offering a range of tools and services to help organizations secure their data and applications.
2. All three platforms comply with numerous industry standards and certifications, but specific compliance offerings may vary by platform.

### Hybrid cloud capabilities

1. All three providers offer hybrid cloud solutions, but Azure's hybrid cloud capabilities are more mature due to its existing enterprise relationships and products like Azure Stack.
2. AWS Outposts and Google Anthos are the respective hybrid cloud solutions for AWS and GCP.

Ultimately, the choice between AWS, GCP, and Azure depends on an organization's specific requirements, existing infrastructure, and familiarity with the platforms. Many organizations opt for a multi-cloud strategy, using more than one cloud provider to take advantage of each platform's unique strengths and features.

## Mobile Application Platforms

### Introduction

Mobile application platforms are tools or environments that enable developers to build, test, and deploy mobile applications for various operating systems like iOS, Android, and sometimes even Windows or other platforms.

### Native Platforms

1. Native platforms refer to the environments and tools offered by operating system vendors that enable developers to create applications optimized for a specific platform or operating system.
2. For iOS (Apple), Xcode is the official Integrated Development Environment (IDE), which supports Swift and Objective-C programming languages.
3. For Android (Google), Android Studio is the official IDE, which supports Java, Kotlin, and C++ programming languages.
4. Native platforms offer advantages such as better performance, access to platform-specific features and APIs, and adherence to the platform's design and user experience guidelines.

### Cross-platform Frameworks

1. Cross-platform frameworks enable developers to create mobile applications for multiple platforms using a single codebase, reducing development time and maintenance efforts compared to native development.
2. Popular frameworks like React Native, Flutter, Xamarin, NativeScript, and Appcelerator Titanium allow developers to use popular programming languages like JavaScript, C#, and Dart and provide near-native performance and access to device features.
3. When choosing a framework, it's essential to consider factors like target platforms, required performance, developer familiarity with programming languages, and desired UI and UX.
4. Cross-platform frameworks save time and resources but may not always provide the same level of performance and platform-specific feature access as native development.

### Hybrid App Frameworks

1. Hybrid app frameworks allow developers to create mobile applications using web technologies such as HTML, CSS, and JavaScript that run inside a WebView on the target platform, which is a native container that displays web content.
2. Popular hybrid app frameworks like Apache Cordova (PhoneGap), Ionic, Framework7, Quasar, and Onsen UI provide a rich set of UI components and APIs to access native device features.
3. These frameworks offer ready-to-use UI elements and widgets optimized for mobile devices, and they can be used with popular web development frameworks like Angular, React, and Vue.js.
4. While hybrid app frameworks offer faster development and easier maintenance, they may not always provide the same level of performance and native-like user experience compared to native mobile app development.

### Web Application Frameworks

1. Web apps are applications that run in a web browser, built using web technologies such as HTML, CSS, and JavaScript so they are accessible across different devices and platforms, making them easier to maintain and more accessible than native applications.
2. Web apps can be simple or complex, depending on their functionality and purpose.
3. Single-page applications (SPAs) dynamically update the content of a single HTML page, providing a smooth user experience.
4. Progressive web apps (PWAs) combine the best features of web and native apps, offering a native-like user experience and offline functionality.
5. Server-rendered web apps are traditional web apps generated by the server, while static web apps consist of pre-built HTML, CSS, and JavaScript files.
6. Web-based mobile apps are designed specifically for mobile devices, built using responsive design principles.
7. When building web apps, developers must consider factors such as cross-browser compatibility, performance optimization, accessibility, and security.
8. Web apps offer easier updates and maintenance, platform independence, and reduced development costs.
9. However, they may not provide the same level of performance and access to native device features as native applications.
10. Choosing the right mobile application platform depends on factors such as target platforms, required performance, programming languages, and desired development speed.
11. Native platforms provide the best performance and access to platform-specific features, while cross-platform and hybrid frameworks allow for faster development and easier maintenance.

## Database Technology

Databases are organized collections of data, typically stored and accessed electronically from a computer system. They are essential for managing, storing, and retrieving large amounts of structured and semi-structured data efficiently. Databases enable quick access to relevant information, facilitating data manipulation and analysis.

#### Relational databases

1. They store data in tables with rows and columns, and relationships between tables are defined through primary and foreign keys.
2. Examples include MySQL, PostgreSQL, Oracle, and Microsoft SQL Server.

#### NoSQL databases

1. These databases are designed for handling unstructured or semi-structured data and do not rely on a fixed schema.
2. Examples include MongoDB (document-based), Cassandra (column-family), Redis (key-value), and Neo4j (graph).

#### Object-oriented databases

1. These databases store and manage data as objects, which can have attributes and methods, like object-oriented programming languages.
2. . Examples include ObjectDB and Db4o.

#### Time-series databases

1. These databases are optimized for handling time-stamped data, such as sensor data or financial data.
2. Examples include InfluxDB and OpenTSDB.

#### Graph databases

1. These databases are designed to store and manage data as nodes and edges in a graph, making them ideal for modeling complex relationships and interconnected data.
2. Examples include Neo4j and Amazon Neptune.

#### Distributed databases

1. These databases store data across multiple servers or nodes, providing fault tolerance, high availability, and horizontal scaling.
2. Examples include Apache Cassandra (NoSQL) and CockroachDB (relational).

Different types of databases are suitable for different use cases, and the choice of a database depends on factors such as data structure, scalability requirements, and the specific needs of an application.

### Why Firebase MySQL MongoDB are the most widely used ?

Firebase, MySQL, and MongoDB are widely used databases due to their unique features, ease of use, scalability, and flexibility in addressing various use cases. Here's a brief overview of each database and the reasons for their popularity

### Firebase (Realtime Database and Firestore)

Firebase is a cloud-based platform developed by Google that offers a suite of backend services, including Realtime Database and Firestore, which are NoSQL databases.

1. Realtime Database
2. Fire store

### MySQL

MySQL is an open-source relational database management system (RDBMS) that has been widely adopted for its performance, reliability, and ease of use. It uses SQL as the query language and is suitable for a wide range of applications. Some reasons for its popularity include:

1. Mature and well-established
2. Easy to set up and use
3. Cost-effective
4. Compatibility
5. Scalability and performance

### MongoDB

MongoDB is a popular open-source NoSQL database that stores data in a flexible, JSON-like format called BSON. It is designed to handle unstructured data and can scale horizontally across multiple nodes. Some reasons for MongoDB's popularity include:

1. Schema-less design
2. Scalability
3. High performance
4. Rich query language
5. Robust ecosystem

## Frontend Technology

Front-end technologies are the tools and frameworks used to create the user interface (UI) and user experience (UX) of a website or web application. They are responsible for the layout, design, and interactivity of a site.

1. HTML (Hyper Text Markup Language)
2. CSS (Cascading Style Sheets)
3. JavaScript
4. Responsive Design
5. CSS Frameworks
6. JavaScript Libraries & Frameworks
7. CSS Preprocessors
8. Build Tools and Task Runners
9. Version Control Systems
10. Testing and Debugging Tools
11. Performance Optimization

## Backend Technologies

Backend technologies refer to the server-side technologies that power web applications, mobile apps, and other software systems. They are responsible for processing, storing, and managing data, as well as ensuring smooth communication between the client-side (frontend) and the server-side (backend).

### Programming languages

#### Python

#### JavaScript

#### Java

#### PHP

#### Ruby

#### C#

1. A language developed by Microsoft, used mainly in the .NET framework for building web applications, APIs, and services.

### Web frameworks

#### Django (Python)

1. A high-level web framework that encourages rapid development and clean, pragmatic design.

#### Flask (Python)

1. A lightweight, micro-framework for Python, suitable for small to medium-sized web applications.

#### Express.js (JavaScript/Node.js)

1. A fast, minimalist web framework for Node.js, widely used for building RESTful APIs.

#### Spring Boot (Java)

1. A popular framework for building enterprise-level applications, simplifying the development process and reducing boilerplate code.

#### Laravel (PHP)

1. A modern web framework for PHP, emphasizing elegant syntax and modular design.

#### Ruby on Rails (Ruby)

1. A full-stack web framework that emphasizes convention over configuration, aiming for developer productivity and ease of use.

### Caching systems

#### Redis

1. An in-memory data structure store, used as a database, cache, and message broker.

#### Memcached

1. A high-performance, distributed memory object caching system, used to alleviate database load.

### Message brokers

#### RabbitMQ

1. A popular open-source message broker that implements the Advanced Message Queuing Protocol (AMQP).

#### Apache Kafka

1. A distributed event streaming platform designed for high-throughput, fault-tolerant, and scalable real-time data processing.

### Server and deployment technologies

#### Nginx

1. A high-performance HTTP server, reverse proxy, and load balancer.

#### Apache HTTP Server

1. A popular, open-source web server software.

#### Docker

1. A platform for developing, shipping, and running applications in containers, ensuring consistent environments and simplified deployment.

#### Kubernetes

1. A container orchestration platform for automating deployment, scaling, and management of containerized applications.

## Visual Studio Code

### Introduction

Visual Studio Code (VSCode) is a popular, lightweight, and powerful source code editor developed by Microsoft. It offers numerous benefits for coding projects, making it a top choice for developers across various programming languages and platforms.

#### Cross-platform compatibility

VSCode is available on all major operating systems, including Windows, macOS, and Linux, which makes it a versatile choice for developers working on different platforms.

#### Extensibility

The editor supports a wide range of extensions and plugins, allowing developers to customize and enhance their coding experience. These extensions can include syntax highlighting, code completion, debugging support, and more for a wide variety of programming languages and frameworks.

#### IntelliSense

VSCode offers advanced code completion, navigation, and understanding features through its IntelliSense feature. This helps developers write code more efficiently, catch errors more easily, and quickly navigate through complex projects.

#### Git integration

The built-in Git support in VSCode simplifies version control, making it easier for developers to track changes, collaborate, and manage their codebase.

#### Debugging capabilities

VSCode offers a powerful built-in debugger, which supports various programming languages and platforms. This allows developers to find and fix issues in their code quickly and efficiently.

#### Customizable UI

The editor's interface is highly customizable, allowing developers to tweak the layout, color scheme, and more to suit their preferences and improve their workflow.

#### Integrated terminal

VSCode comes with an integrated terminal that allows developers to execute commands, run scripts, and interact with their development environment without leaving the editor.

#### Language support

The editor supports a vast array of programming languages, making it suitable for many different types of coding projects.

#### Collaboration features

The Live Share extension enables real-time collaboration between developers, allowing them to share their workspace and code together, regardless of their physical location.

#### Active community and support

Visual Studio Code has a large and active community, which contributes to the development of the editor and its extensions. This ensures that the tool stays up-to-date, and developers can find support and resources when needed.

### Conclusion

In summary, Visual Studio Code provided a powerful, extensible, and user-friendly environment for developers working on various coding projects. Its wide array of features, support for multiple languages, and strong community backing make it a top choice for many programmers.

# PROPOSED SOLUTION

Introduction of what you planning to hist and basic of functionality brief

Sniipets of working code and some images of some major features db php etc

Diagram

Description automatically generated

Basic layout of the architecture for this project

Diagram

Description automatically generated

Technology layout of the architecture for this project

Logo

Description automatically generated

Technologies Used majorly

Graphical user interface, website

Description automatically generated

Opening Website Look

UML CLASS DIAGRAM

Diagram

Description automatically generated

This Diagram above gives a basic flow of class diagram involving the steps needed for the website. The flow and steps I plan during this project also showing the important roles and functions I will try to follow while coding the project

## Database (MySQL & Firebase)

Both Firebase and MySQL are popular databases, but they serve different purposes and are suited for different use cases

### Firebase

Firebase is a part of the Google Firebase suite, which includes other tools like Firebase Authentication, Cloud Functions, Cloud Messaging, and more. These services can be easily integrated with the Firebase database, streamlining development. Firebase is ideal for real-time applications like chat applications, gaming platforms, or collaborative tools, where data needs to be synced across multiple devices and users in real-time.

### MySQL

MySQL is suitable for applications that require complex data relationships, transactions, and querying capabilities, such as e-commerce platforms, content management systems, or business applications. MySQL is an open-source, standalone database system and can be hosted on-premises or in the cloud. It is not tied to a specific suite of services. MySQL is a relational database, which means it stores data in tables with predefined schemas. It is designed for structured data storage and supports transactions and complex queries

### Conclusion

I wanted to use both Firebase and MySQL together because my application has mixed requirements. For instance, I used Firebase for real-time features such as user authentication and chat application, while using MySQL for structured data storage and complex querying.

## Frontend Technologies (Bootstrap and HTML CSS)

Talking about HTML Rendering , environment setup , NPM, command line , rooting



### Bootstrap

Timesaving: Bootstrap is a popular CSS framework that provides pre-designed components and styles, which can speed up the development process by reducing the time spent on writing CSS from scratch.

#### Responsive Design

Bootstrap's grid system and responsive utilities make it easy to create responsive web designs that adapt to different screen sizes and devices.

#### Consistent Look and Feel

Using Bootstrap components ensures a consistent look and feel across your application, as well as compatibility with various browsers.

Customizable  
Bootstrap can be customized according to your design requirements, allowing you to change colors, fonts, and other styles while still leveraging its predefined components.

#### Large Community

Bootstrap has a large and active community, which means you can find plenty of resources, plugins, and tutorials online to help you learn and use the framework effectively.

### HTML, CSS, and JavaScript

#### Fundamental Technologies

HTML, CSS, and JS are the core technologies for building any web-based frontend. HTML is used for structuring content, CSS for styling, and JS for adding interactivity and dynamic content.

#### Browser Compatibility

Modern browsers support HTML, CSS, and JS natively, ensuring your website or web application will work across various platforms without requiring additional plugins or extensions.

#### Wide Range of Libraries and Frameworks

There are numerous libraries and frameworks available for HTML, CSS, and JS that can help you create complex and interactive frontend experiences. Examples include jQuery (JS library), React, Angular, and Vue.js (JS frameworks), and Sass or Less (CSS preprocessors).

#### Developer Familiarity

Most web developers are already familiar with HTML, CSS, and JS, making it easier to find and onboard developers for your project.

#### SEO and Accessibility

HTML, CSS, and JS, when used properly, can contribute to better search engine optimization (SEO) and accessibility for your website, making it more discoverable and user-friendly.

### Conclusion

By choosing Bootstrap combined with HTML, CSS, and JS for my front-end development, I can create responsive, visually appealing, and interactive web applications while leveraging a well-established and widely supported technology stack.

## Backend Technologies

### PHP

#### Mature Language

PHP has been around for a long time and has a well-established ecosystem. It is widely used for web development and has a large community that can provide support and resources.

#### Easy to Learn

PHP has a relatively low learning curve, making it accessible to developers with varying levels of experience.

#### Built for Web Development

PHP was designed specifically for web development, and its syntax and features are tailored for building web applications efficiently.

#### Integration with Web Servers

PHP can be easily integrated with popular web servers like Apache and Nginx, simplifying the deployment process.

#### Wide Range of Frameworks

PHP offers various frameworks like Laravel, Symfony, and CodeIgniter, which can help you structure your code and speed up development.

#### Cost-effective Hosting

PHP hosting is widely available and often more affordable than hosting for some other backend languages.

### JavaScript (Node.js)

#### Single Language for Frontend and Backend

Using JavaScript with Node.js allows you to develop both the frontend and backend of your application using a single programming language, which can streamline development and make it easier to find developers.

#### Non-blocking, Asynchronous I/O

Node.js uses an event-driven architecture with non-blocking I/O, making it well-suited for handling multiple concurrent connections and building scalable, high-performance applications.

#### Rich Ecosystem

JavaScript has a vast ecosystem with a wide range of libraries and frameworks available, which can help you build feature-rich applications quickly.

#### Real-time Applications

Node.js is an excellent choice for building real-time applications like chat applications, online gaming platforms, or live data streaming services due to its asynchronous nature and ability to handle real-time data efficiently.

#### Easy to Scale

Node.js applications are easy to scale, both horizontally and vertically, which can be important if you expect your application to grow and serve many users.

### Conclusion

By choosing PHP and JavaScript (Node.js) as your backend technologies, you can take advantage of the strengths of both languages, such as PHP's maturity and web development focus, along with JavaScript's versatility and performance capabilities

## Cloud Service and Connections

Complete AWS working , images code snippets and screenshots

# PLAN FOR PROGRESSION

* To establish partnerships with reputable service providers specializing in healthcare, personal care, home maintenance, leisure activities, and transportation, to offer a diverse and comprehensive range of services for the elderly users.
* To promote the elderly service booking website through targeted marketing campaigns, community outreach, and collaboration with relevant organizations, ensuring widespread awareness and adoption of the platform among the target demographic.
* To evaluate the impact of the platform on the quality of life and independence of elderly users, by tracking key performance indicators such as user satisfaction, frequency of service bookings, and reduction in barriers to access essential services.
* To create a supportive online community within the platform, where elderly users can engage with each other, share experiences, and receive peer-to-peer support, thus fostering social interaction and reducing feelings of isolation.

# REFERENCES

1. Verzeo Blogs. 2022. Which is the Best Operating System?? iOS vs Android vs Windows. [ONLINE] Available at: https://blog.verzeo.com/which-is-the-best-operating-system/#:~:text=Android%20is%20open%20source%2C%20while%20iOS%20and%20Windows,Apple%20Store%2C%20and%20Android%20has%20Google%20Play%20Store.. [Accessed 22 November 2022].
2. What is Cloud Computing? | IBM. 2022. What is Cloud Computing? | IBM. [ONLINE] Available at: https://www.ibm.com/cloud/learn/cloud-computing. [Accessed 22 November 2022].
3. Wikipedia. 2022. Amazon Web Services - Wikipedia. [ONLINE] Available at: https://en.wikipedia.org/wiki/Amazon\_Web\_Services. [Accessed 24 November 2022].
4. Wikipedia. 2022. Microsoft Azure - Wikipedia. [ONLINE] Available at: https://en.wikipedia.org/wiki/Microsoft\_Azure. [Accessed 24 November 2022].
5. WebsiteBuilderPoint |. 2022. What Is Client-Side in Web Development?. [ONLINE] Available at: https://www.websitebuilderpoint.net/what-is-client-side-in-web-development//. [Accessed 24 November 2022].
6. WebsiteBuilderPoint |. 2022. What Is Server-Side Development in Web Technology?. [ONLINE] Available at: https://www.websitebuilderpoint.net/what-is-server-side-development-in-web-technology/#:~:text=Server-side%20development%20is%20a%20process%20where%20the%20code,when%20the%20interface%20is%20not%20accessible%20to%20users.. [Accessed 24 November 2022].
7. Useful apps and websites for older people | Handicare News. 2022. Useful apps and websites for older people | Handicare News. [ONLINE] Available at: https://www.ageukmobility.co.uk/mobility-news/article/useful-apps-and-websites-for-older-people. [Accessed 24 November 2022].
8. Goodnet. 2022. 7 of the Best Apps for People with Disabilities - Goodnet. [ONLINE] Available at: https://www.goodnet.org/articles/7-best-apps-for-people-disabilities-list. [Accessed 24 November 2022].
9. Tunstall. 2017. 9 apps and websites for AT and disability services that could change your life - tunstallhealthcare. [ONLINE] Available at: https://blog.tunstallhealthcare.com.au/disability/9-apps-and-websites-for-at-and-disability-services-that-could-change-your-life/. [Accessed 28 November 2022].
10. Educba. 2022. Google Cloud Platform. [ONLINE] Available at: https://www.educba.com/google-cloud-platform/. [Accessed 28 November 2022].
11. Educba. 2022. GCP vs AWS vs Azure. [ONLINE] Available at: https://www.educba.com/gcp-vs-aws-vs-azure/. [Accessed 28 November 2022].
12. Focaloid Technologies. 2021. Types of Application Development Services. [ONLINE] Available at: https://www.focaloid.com/blog/types-of-application-development-services/#:~:text=Types%20of%20application%20development%20include%20desktop%20application%20development%2C,full%20form%20of%20API%20is%20Application%20Program%20Interface. [Accessed 28 November 2022].
13. Wikipedia. 2022. Firebase - Wikipedia. [ONLINE] Available at: https://en.wikipedia.org/wiki/Firebase. [Accessed 28 November 2022].
14. Olga Anoshyna. 2022. Why Choose Firebase as Cloud Backend for your Mobile Apps - Super Dev Resources. [ONLINE] Available at: https://superdevresources.com/why-use-firebase/. [Accessed 28 November 2022].
15. Willvick. 2022. Advantages of Using Bootstrap - Willvick. [ONLINE] Available at: https://willvick.com/advantages-of-using-bootstrap/#:~:text=Advantages%20of%20Using%20Bootstrap%201%20Open%20Source%20%E2%80%94,Bootstrap%20predefined%20design%20templates%20and%20classes.%20More%20items. [Accessed 28 November 2022].
16. Upwork. 2021. A Beginner's Guide to Back-End Development. [ONLINE] Available at: https://www.upwork.com/resources/beginners-guide-back-end-development. [Accessed 28 November 2022].
17. Upwork. 2021. Server-Side Scripting: Back-End Web Development Technology. [ONLINE] Available at: https://www.upwork.com/resources/server-side-scripting-back-end-web-development-technology. [Accessed 28 November 2022].
18. Jigsaw Academy. 2022. Top 15 IAAS Examples You Need To Know In 2021. [ONLINE] Available at: https://www.jigsawacademy.com/blogs/cloud-computing/iaas-examples/. [Accessed 29 November 2022].
19. Jigsaw Academy. 2022. Top 10 SAAS Examples. [ONLINE] Available at: https://www.jigsawacademy.com/blogs/cloud-computing/saas-examples/. [Accessed 29 November 2022].
20. Serverless Works. 2022. Serverless Computing Examples Explained | Serverless Works. [ONLINE] Available at: https://serverlessworks.com/examples#:~:text=Some%20of%20the%20examples%20of%20core%20serverless%20technologies,5%20Cloud%20functions%206%20Cloud%20pub%2Fsub%207%20Cloud. [Accessed 29 November 2022].
21. YourTechDiet. 2022. Why opt for FAAs? Here are 4 Real Life Examples. [ONLINE] Available at: https://yourtechdiet.com/blogs/function-as-a-service-faas-examples/. [Accessed 29 November 2022].
22. Salesforce.com. 2023. 12 Benefits of Cloud Computing and Its Advantages - Salesforce.com. [ONLINE] Available at: https://www.salesforce.com/products/platform/best-practices/benefits-of-cloud-computing/. [Accessed 23 January 2023].
23. Clutch.com. 2015. Mobile App Platforms: Hybrid, Native, Mobile Web - Clutch.com. [ONLINE] Available at: https://clutch.co/app-developers/resources/mobile-app-platforms-hybrid-native-mobile-web#:~:text=When%20building%20a%20mobile%20app%2C%20there%20are%20three,of%20each%20platform%20type%20and%20possible%20use%20cases.. [Accessed 23 January 2023].
24. Data Flair. 2022. Pros and Cons of JavaScript- data-flair.training.com. [ONLINE] Available at: https://data-flair.training/blogs/advantages-disadvantages-javascript/#:~:text=Advantages%20of%20JavaScript%201%201.%20Speed%20Since%20JavaScript,Functionality%20...%208%208.%20Versatility%20...%20More%20items. [Accessed 23 January 2023].
25. Sagar Sharma. 2023. 15 Top Front-End Technologies To Rule The Market. [ONLINE] Available at: https://credencys.com/blog/front-end-technologies/#:~:text=In%20simple%20words%2C%20it%20is%20a%20set%20of,up%20a%20website%2C%20web%20application%2C%20or%20mobile%20app.. [Accessed 27 January 2023].
26. Microsoft. 2023. What is IaaS?. [ONLINE] Available at: https://azure.microsoft.com/en-us/resources/cloud-computing-dictionary/what-is-iaas/ [Accessed 04 February 2023].
27. Managed cloud services (2023) RapidScale. Available at: https://rapidscale.net/ (Accessed: 09 May 2023).
28. InformationWeek, serving the information needs of the Business Technology Community (no date) InformationWeek. Available at: https://www.informationweek.com/ (Accessed: 09 May 2023).